FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

ASSIGNMENT OF BACHELOR’S THESIS

Title:	Analysis of an IoT solution for assessment of the physical difficulty of tourist tracks using smart wearables
Student:	Zuzana Václaviková
Supervisor:	Ing. Jan Krákora, PhD.
Study Programme:	Informatics
Study Branch:	Web and Software Engineering
Department:	Department of Software Engineering
Validity:	Until the end of summer semester 2020/21

Instructions

The aim of this thesis is to analyze, design and implement a PoC IoT solution for difficulty estimation of tourist tracks. Using a mobile application, the user shall be able to inspect a track in advance and assess how much effort each specific part of the track would require from them, based on data collected from similarly fit users.

1. Analyze and compare similar existing solutions.
2. Analyze and design an IoT solution for data processing.
3. Design a mobile application for data visualization and track selection.
4. Create a PoC of the IoT platform and of the mobile application.
5. Demonstrate the solution’s functionality on a relevant use case, using real data collected from smart wearables.

References

Will be provided by the supervisor.

Ing. Michal Valenta, Ph.D.
Head of Department

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
Dean

Prague January 6, 2020

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Bachelor's thesis

Analysis of an IoT solution for assessment of physical difficulty of tourist tracks using smart wearables

Zuzana Václaviková

Department of Software Engineering
Supervisor: Ing. Jan Krákora, PhD.

June 4, 2020

Acknowledgements

I would like to thank my supervisor, Ing. Jan Krákora, PhD., for his guidance and encouraging words when I was out of my depth. Many thanks to Šimon Let for reviewing my work throughout the writing process and giving valuable advice. Finally, thank you to my partner David for providing endless support, and my family and friends, for never ceasing to believe in me.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as school work under the provisions of Article 60(1) of the Act.

In Prague on June 4, 2020 . . . .

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Václaviková, Zuzana. *Analysis of an IoT solution for assessment of physical difficulty of tourist tracks using smart wearables*. Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2020.

Abstrakt

Predmetom bakalárskej práce je softvérová analýza platformy Internetu vecí, ktorá odhaduje náročnosť turistických trás pre turistov na základe údajov o srdcovej činnosti zozbieraných z inteligentných hodiniek patriacich používateľom.

Výskumné kapitoly práce sa sústredia na moderné metódy hodnotenia kardiovaskulárnej kondície, ako aj štúdiu existujúcich aplikácií v oblasti fitness a e-health.

V práci je využitá metodika agilného vývoja. Funkcionalita riešenia je analyzovaná vo forme user stories, ktoré môžu byť priamo preberané ako zadania pre vývojárske tímy. Táto analýza je základom pre návrh front-endovej časti mobilnej aplikácie, ktorá podporuje navigáciu, integráciu inteligentných hodiniek a informované plánovanie túr samostatne alebo spolu s inými používateľmi aplikácie na základe ich fyzickej kondície. Celý design je vytvorený s dôrazom na smernice pre user experience.

Okrem toho je nadizajnovaná high-level architektúra riešenia a vyvinutý proof of concept platformy, ktorý je z dôvodu technických problémov nedokončený.

Kľúčové slová Internet vecí, náročnosť turistickej trasy, meranie tepu srdca, posudzovanie telesnej kondície, nositeľná elektronika, softvérová analýza, dizajn mobilnej aplikácie, užívateľské skúsenosti

Abstract

The subject of this bachelor's thesis is to perform a software analysis of an Internet of Things platform that estimates how difficult tourist tracks will be for people before they take them, based on heart rate data collected from their and other users' smartwatches.

The research chapters of the thesis focus on modern methods of cardiovascular fitness assessment and the study of existing applications in the domain of fitness and e-health.

Using methods of agile development, the functionality of the solution is analysed in the form of user stories, which can be directly taken as assignments for development teams. This analysis is the foundation for designing the front-end of a mobile application that supports navigation, smartwatch integration, and informed planning of hikes both alone and with other users based on all the participants' fitness. The whole design is built with an emphasis on user experience guidelines.

Subsequently, a high-level architecture of the solution is designed, and a proof of concept of the platform is developed, however, due to technical difficulties, it is unfinished.

Keywords Internet of Things, hiking track difficulty, heart rate monitoring, physical fitness assessment, smart wearables, software analysis, mobile application design, user experience

Contents

1 **Introduction** **1**

2 **Aims** **3**

3 **Fitness assessment** **5**

3.1 VO2 and VO2 max 5

3.2 The Performance Condition 9

3.3 Heart rate zones 10

4 **Existing solutions** **13**

4.1 komoot 13

4.2 endomondo 17

4.3 Strava 23

4.4 Garmin Explore and Garmin Connect 26

4.5 Summary 30

5 **Functionality** **31**

5.1 Heart rate monitoring 32

5.1.1 Manage devices paired with the App 32

5.1.2 Activate a device 33

5.1.3 Choose and integrate a wearable 33

5.2 Objective user fitness assessment 33

5.2.1 Implement support for VO2 max tests 34

5.2.2 Implement support for HR max and HR min tests . . . 35

5.3 Manual input of biometric data 35

5.3.1 Allow manual input of basic biometric values 35

5.3.2 Allow manual input of advanced biometric values 36

5.4 Routes and maps 36

5.4.1 Integrate a map 36

5.4.2 Plan a route . . . 37
5.4.3 History of hikes taken . . . 37
5.4.4 Saved routes . . . 38
5.4.5 Share routes . . . 38
5.4.6 Display planned route details . . . 39
5.4.7 Display old hike details . . . 39
5.4.8 Hike a route . . . 39
5.4.9 Change the hike date . . . 40
5.5 User profile . . . 40
5.5.1 Management of provided biometric information . . . 40
5.5.2 Profile picture management . . . 40
5.5.3 Log in options . . . 41
5.6 Hike with friends . . . 41
5.6.1 Adjust the calculations to friends' fitness . . . 42
5.6.2 Invite friends to hikes . . . 42
5.6.3 Refuse an invitation to hike with friends . . . 42
5.6.4 Accept an invitation to hike with friends . . . 43
5.6.5 Add friends . . . 43
5.6.6 Remove friends . . . 43
5.6.7 Accept a friendship request . . . 44
5.6.8 Dismiss a friendship request . . . 44
5.6.9 Share a link to a profile . . . 44
5.6.10 Joinable hikes . . . 44
5.6.11 Cancelling a hike . . . 45
5.6.12 Hike active on all participants' phones . . . 46
5.7 Notifications . . . 46
5.7.1 Notifications for friends tab . . . 46
5.7.2 Notifications for planned hikes . . . 46
5.8 Notes . . . 47
5.9 Considered features . . . 47

6 Mobile app UI design **49**
6.1 Set up screens . . . 51
6.2 Profile tab . . . 52
6.3 Map tab . . . 53
6.4 Plans tab . . . 57
6.5 Routes tab . . . 59
6.6 Friends tab . . . 59
6.7 Navigation mode . . . 60
6.8 Notifications . . . 61

7 Architecture **71**
7.1 Smartwatch application . . . 72
7.2 Smartphone application . . . 72

7.3 Server application 72

8 PoC implementation 75

8.1 Android application 75

8.2 Smartwatch application 75

8.3 Recommendations for future development 77

Conclusion 79

Bibliography 81

A Acronyms 89

B Contents of enclosed CD 91

List of Figures

3.1 With increasing speed of the runner, oxygen consumption increases linearly and plateaus around 18.5 kilometres per hour - the VO2 max [1]. 6

4.1 Route planning in the komoot app [2]. 15
4.2 Route elevation profile [3]. 16
4.3 Overview of the route's elevation [4]. 17
4.4 Overview of the route's surface [5]. 18
4.5 endomondo's main screen [6]. 20
4.6 endomondo's `More` tab, with main settings lost in other options [7]. 21
4.7 Statistics of a biking trip on endomondo [8]. 22
4.8 endomondo history example from the web application [9]. 22
4.9 Challenges feed on Strava [10]. 24
4.10 Strava segments near the user's location [11]. 25
4.11 Statistics of a run on Strava [12]. 26
4.12 My day - Garmin Connect's home screen displays statistics of the user's recent activity [13]. 27
4.13 Garmin's map tab [14] and route creation screen [15]. 29

6.1 Thumb-zone mapping for left- and right-handed users [16]. 50
6.2 Welcome screen with sign-up options and e-mail sign-up screen. . . 51
6.3 Welcome screen with sign-in options and e-mail sign-in screen. . . 52
6.4 The Profile screen 53
6.5 Fitness information 54
6.6 Fitness tests 55
6.7 6 Minute Walking Test 56
6.8 Devices 57
6.9 Map screen 58
6.10 Search for a place 59
6.11 Planning a route with waypoints 60

6.12 Picking a route based on its details. 61
6.13 The initial state of a planned hike. 62
6.14 Searching for friends to invite to a hike. A summary of already chosen people is on the bottom. 63
6.15 Overview of the hike. After pressing the `Finish` button, we are back to the `Map` screen. 64
6.16 All plans feed. Detail of a joinable hike. 65
6.17 My plans. Detail of a hike planned by the user, and a view of the invited and going people. 66
6.18 Detail of a historic hike. 67
6.19 Routes tab. Detail of a route. 68
6.20 Friends tab, finding new friends and the user profile of a friend. . . 68
6.21 Navigation mode. 69

7.1 Component diagram of the designed IoT solution. 71

8.1 Market share of smartwatch unit shipments worldwide from summer 2014 to spring 2020, by vendor [17]. 76

Chapter 1

Introduction

With the increasing popularity of smart wearables in the general public, a growing number of people have been able to benefit from adjustments in their behaviour according to the way their data gets processed and fed back to them. Runners can barely imagine not checking how fast and how efficiently they just ran their daily track or how much they have improved over the last month, people with sedentary jobs attempt to keep their daily step count above a currently recommended number, masses of people have been saving up for the Apple Watch 5 [18] in spite of the ECG health monitor being virtually unnecessary for those not at risk of cardiac disease [19], etc.

All of this data is collected from a variety of devices, such as smartwatches, textiles with integrated sensors, jewellery, or more invasive ones such as implants or tattoos, all of which are meant to be worn on the user's body and which measure and send data for analysis – hence the name *wearable technology*, or *smartwear* [20].

Thanks to the data collected from these devices, the users have been keeping track of their health and fitness, setting goals for themselves and tracking personal achievements. However, all of this is related to the past of each individual user, while the true potential of smartwear lies in collecting data from thousands of people, finding patterns and correlations in large datasets and using the results to increase precision and improve upon whatever the data is being used for by the virtue of prediction.

That being said, the result of this thesis will be a design for an IoT solution, mainly to be used by hikers who want to not only pick a track on a map and see how long it is going to take them to get from point A to point B (which generally isn't tailored to the individual's fitness and often doesn't account for the possible rapid changes in the track's elevation profile), but also an objective difficulty level of particular sections of said track, which may have additional benefits for people with cardiac issues or those trying to lose weight, who are interested in some more detailed insight into the exertion induced by the track in order to keep their heart rate in a specific range. All of this will be available to the user before stepping foot on said track, thanks to the data collected from users with various fitness levels and their own history, while the user's fitness will be presented more accurately as they continue to use the application.

I will study how fitness is assessed in regard to feasibility, availability for the general public, and precision, in order to provide fundamentals for further work.

Using methods of software engineering, I am going to analyse the use of wearable technology in existing applications and document the most common features as well as their outstanding traits and their use of IoT principles (or lack thereof). Based on this analysis, I will pick the most relevant features to include in my design and extend them with features of my own devising, which will set my solution apart from existing software.

I will design and implement a PoC IoT solution – a mobile application which will collect data from the smartwear sensor and an available GPS locator, along with a backend IoT platform to receive and process collected data, and the communication channels used between these two modules. I will neither design nor implement a full-fledged IoT platform as that's not in the scope of this thesis. I will design (not implement) the frontend of a mobile application that will not only allow a user to show their statistics and estimations, but will also include options to hike with friends.

I will examine the use of different device types with heart rate sensors and choose one with which I will build my PoC.

I will collect and perform suitable operations on the collected data in order to visualize it.

Chapter 2

Aims

The aim of this thesis is to analyse and design the functionality of an IoT solution for assessment of tourist track difficulty, based on the individual's fitness.

The early chapters of this thesis will be the analysis of the ways fitness is currently assessed and their relevance to my application, followed by a study of existing applications in the domain of fitness trackers and e-health, and an assessment of how important specific features are to the applications' main aims. I am going to apply the results of this research to my own approach when determining the required functionality for the system.

Based on these descriptions, I am going to design the front-end of a mobile application for track selection and data visualization. I am also going to design a high-level architecture of the IoT solution.

In the last part of the thesis I am going to focus on implementing a PoC of the designed solution and processing of collected data.

Chapter 3

Fitness assessment

Since the platform should estimate how a user will be affected by a hike, there needs to be some way of assessing their fitness to make the estimates legitimate instead of just guessing. This is why this chapter studies modern methods of fitness testing, the variables that need to be considered, and the accuracy and feasibility of such tests.

A physical fitness test usually evaluates multiple components of one's health, including cardiovascular endurance, muscular strength, muscular endurance, flexibility and body composition.

I will focus on familiarizing the reader with cardiovascular endurance testing since the heart, lungs and blood vessels are what gets exerted the most during hiking. There are other indicators that one can measure (such as the lactate threshold - a non-linear increase in blood lactate concentration [21]), but the most popular and easiest to measure with affordable sensors is VO2 max.

3.1 VO2 and VO2 max

The volume of consumed oxygen (VO2) is closely correlated with heart rate, respiration rate, and on/off-dynamics, all of which can be derived from RR interval data (time elapsed between two successive R waves – the tallest spikes on an ECG), with respiration rate being able to differentiate between metabolic (physical activity-induced) and non-metabolic (mental and non-exercise related physical stress) changes in heart rate, which makes models that take it into account highly accurate compared to those which only use heart rate [22].

VO2 max, the maximal value of VO2, is considered the most accurate metric of cardiovascular fitness.

Measured in millilitres of oxygen consumed per minute or millilitres of oxygen per kilogram of bodyweight per minute, it is the amount of oxygen the body is able to effectively use (transform to energy) during intense or maximal exercise. With the growing amount of oxygen the body can consume, it can perform better during strenuous activity and give improved results [23]. This is limited by the ability of the cardiorespiratory system to deliver oxygen to the exercising muscle, thus making it impossible for an athlete to operate above 100% of their VO2 max for extended periods [24].

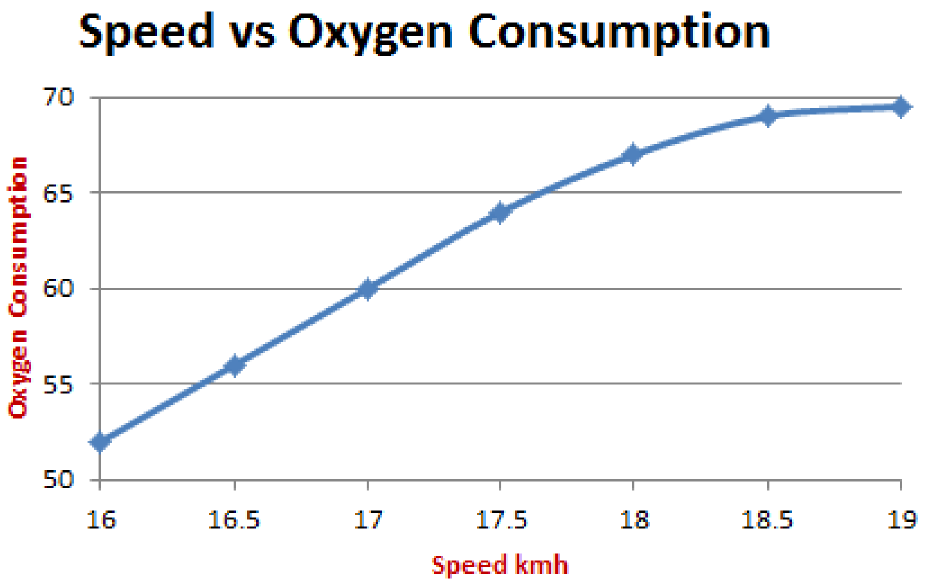


Figure 3.1: With increasing speed of the runner, oxygen consumption increases linearly and plateaus around 18.5 kilometres per hour - the VO2 max [1].

How to measure Vo2 max

Laboratory test

Typically, VO2 max is measured directly in laboratory conditions while wearing a respiratory mask, by analyzing inspired and expired breathing gases during maximal exertion [23], usually running on a treadmill or riding a stationary bike.

This method of determining VO2 max is highly accurate, but given the need for expensive equipment and trained staff, laboratory testing just isn't feasible for everyday population-wide testing. On the other hand, results from laboratory tests are often used as reference values for determining the accuracy of alternative methods.

Cooper test - Twelve-Minute Run-Walk

First suggested in the 1970's, Cooper's Twelve-Minute Run-Walk is an endurance test, where the main goal is to run (or walk) as long a distance as possible in twelve minutes. It was originally developed mainly for armies and police agencies, but it's also popularly (and unnecessarily [25]) used in schools on untrained pupils. It is inaccurate for people who do not train running and swim or bike instead, as their bodies are used to a completely different set of movements.

There is a high correlation between the distance an individual can run and their VO2 max value, which can be calculated thusly:
$VO2max = (22.351 \times distance\_in\_kilometers) - 11.288$ [26].

Multistage Fitness Test - Beep test

Introduced by a Canadian sports scientist Luc Léger in the 1970s, this aerobic test has consistently been found a reliable way of finding a person's VO2 max.

The original version of the beep test (the Track Test) had the participants running back and forth in the interval of two minutes while the running pace was increased, so they always had to run further than in the previous iteration.

This test was highly efficient, however, given the long intervals and spatial requirements, it couldn't be performed indoors, giving rise to modifications, such as the Twenty-Metre Shuttle Run Test. As the name suggests, two markers are placed twenty metres apart and again, the participants run back and forth between them, trying to reach the opposite marker before the next – shorter – stage begins, that is, before the next beep. Once the test subject fails to reach a marker, turns without touching the marked line, or starts running before the signal, after one warning, their test is over. They can also choose to stop when they have reached their maximum physical limit.

The test is divided into stages with recommended running speeds, each stage consisting of multiple shuttles - see the table at [27]. From the stage and shuttle reached, one can calculate the VO2 max using the formula

$$VO2max(\frac{ml}{kg * min - 1}) = 31.025 + 3.238X - 3.248A + 0.1536AX$$

where A is age of the participant and X is the speed in the final shuttle. This formula fits the commonly used version of the Beep test (starting at 8.0 km/h, jumping to 9.0 km/h and continuing to rise by 0.5 km/h per level) and may vary depending on the version used [28] [29].

This modified version (just like the original) has been recognized as a valid method to determine VO2 max of male and female adults, individually or in groups, on most gymnasium surfaces [29].

In 2017, Tomkinson et al. published a complex systematic study of nearly 1.2 million 9-17 years old children from 50 countries with data from beep tests as old as 1981, setting standardized norms for testing of fitness in the world's youth [30].

Six-Minute Walk Test

There are also some non-running tests which can be used to measure an individual's VO2 max.

The Six-Minute Walk Test, or 6MWT for short, originated from the Cooper test in the 1980's as a more feasible and less exhausting alternative [31].

The main objective is for the participant to walk as long a distance as they can in the provided six minutes on a 15- or 30-metre long track, making sharp turns at the end. Mänttäri et al. published a study in 2018 [32], aimed to develop a prediction model for VO2 max based on 6MWT results combined with heart rate at the end of the 6MWT and easy-to-measure anthropometric and demographic data (such as weight, height, age and gender). The different track lengths only showed negligible differences between distances walked (on average, 23.5 m longer on the 30-m course). *For men, the best predictors for VO2max were walking distance, age, BMI, heart rate at the end of 6MWT and height, and for women, walking distance, age and weight* – surprisingly, heart rate wasn't a significant factor for the female participants.

The formula they developed for men:

$VO2max = 110.546 + 0.062 \times distance - 0.25 \times age - 0.486 \times BMI - 0.42 \times height - 0.109 \times HR$

and for women:

$VO2max = 22.506 - 0.271 \times weight + 0.051 \times distance - 0.065 \times age$

where weight is body weight (kg), distance is distance walked in 6 min (m), age (years), BMI is calculated body mass index (kg/m2), height is body height (cm) and HR is heart rate at the end of the walking test (bpm).

The authors themselves conclude that their output should be cross-validated with new independent samples of the adult population, but they consider it a precise enough method to be used for public health monitoring in adults.

Another study from 2015 [33] instead used resting heart rate and arrived at this formula:

$VO2max = 70.161 + 0.023 \times distance - 0.276 \times weight - 6.79 \times sex - 0.193 \times resting_HR - 0.191 \times age$

using *distance* walked during 6MWT (m), *weight* (kg), *sex* – 1 for female, 0 for male, and $resting_HR$ for resting heart rate in beats per minute.

Sit-to-Stand test

Another test that doesn't require much in terms of space or equipment and is in fact a common daily activity, which is often used to test lower body strength and endurance in older adults, has been studied for correlation with VO2 max values. The Sit-to-Stand test, which consists of repetitive sitting down and standing up, has acquired a number of different forms just like most other tests used by the general public. The versions include but aren't limited to the incremental twelve-minute STS test – sit and stand in incrementally shorter periods of time, repeated for twelve minutes, often controlled by a metronome [34];

Having been closely studied by Nakamura et al., it is a potentially valid test (when done with arm support, to increase the complete exhaustion threshold, and the right chair height, among other factors), but more research needs to be carried out to find a suitable version [35] [36].

Non-exercise tests

There are also some tests that do not require exercise and instead, have the individual fill out a questionnaire, such as the Jackson test [37] or the George test [38], which ask questions on the topic of an individual's exercise habits, such as their perceived level of activity over the last six months. Every answer has a specific weight in the resulting equation, based on which the person's fitness is evaluated.

These tests, however, should not be used as a sole source of information especially in medicine, as the data is self-reported and may be influenced by bias or conscious falsehood.

3.2 The Performance Condition

A person's VO2 max is recognized as their baseline fitness level index, however, everybody has days when they perform better, and days when they perform worse, while the factors in play are the amount of sleep they get, being sore from previous workouts, and general physical and mental wellbeing.

That is why Firstbeat introduced the Performance Condition.

Firstbeat is a provider of physiological analytics for sports and well-being who have translated human physiology into mathematical models based on heart rate variability (see below). They provide their analytics engine as a service to a number of fitness device manufacturers, such as Garmin, Xiaomi, Honor and others.

The Performance Condition is a real-time index of a user's immediate fitness and fatigue level compared to their baseline fitness level.

This indicator is measured on a scale of -20 to +20, with each point representing roughly 1% of their Vo2 max. So if the user's current Performance Condition is +4, they can expect to perform outstandingly, but at the same time, during the course of a workout, this number will decrease as fatigue from the exercise gets closer [39] [40].

According to Firstbeat's support personnel, the formula behind the Performance Condition is protected information, but generally, it uses a combination of personal background data, internal and external workload data, and special guidance from the Firstbeat analytics engine [41]. Garmin's support website provides similar information: it is calculated based on the user's pace, heart rate and heart rate variability (HRV) [40].

HRV is the variability in intervals between cardiac cycles. It can be demonstrated, for example, by feeling one's pulse on the wrist while resting and breathing deeply - the interval shortens (heart beating faster) when breathing in, and lengthens (heart beating slower) when breathing out [42]. It's a complicated enough metric that to calculate it accurately, HRV should be measured using a chest strap, an intrusive heart rate monitoring method which generally isn't available to a runner on a daily basis.

Additionally, the only activities that are currently supported for determining the Performance Condition, are running and cycling, and the methods Firstbeat has developed to measure it has failed to satisfy a number of users, leading to them not paying much attention to the metric or outright ignoring the values [43] [44]. One of the reasons behind their dissatisfaction is the fact that the metric only connects the effort the user is making with the distance they are covering, without taking into account the steepness of the slope, the humidity, temperature, and other external factors and should be perceived more as an indicator of simply how much effort your body is making (as explained in BHerman's comment on a post [44] in Firstbeat's forum), while being marketed as a highly precise metric that shows you how your training is going and not really explained well to the users.

3.3 Heart rate zones

Everybody has a resting heart rate (for example, right after waking up from a good night's sleep), and a maximum heart rate (the highest number of beats per minute achievable). The range between HR min and HR max is commonly divided into five zones, based on the effort necessary and the effects the heart rate induces in a person:

- Zone 1 (Warm-up and very light exercise) – The trainee is relaxed at an easy pace and breathes rhythmically. Keeping one's heart rate in this zone helps with recovery from previous activities.

- Zone 2 (Easy, light exercise) – The trainee exercises at a comfortable pace, is able to hold a conversation and breathes slightly deeper. This zone is good for endurance practice.

- Zone 3 (Aerobic, moderate exercise) – At a moderate pace, it is more difficult to hold a conversation. Training in this zone helps improve aerobic capacity.

- Zone 4 (Threshold, hard exercise) – Training at a fast pace is somewhat uncomfortable, the trainee breathes forcefully. This zone is good for improving speed.

- Zone 5 (Maximum) – The trainee is at a sprinting pace, which is unsustainable for long period of time, and their breathing is laboured. This zone helps increase power in the trainee's body [45] [46].

Calculating HR min is an easy enough task - just measure your heart rate right after enjoying some good sleep; HR max is trickier.

Running methods to calculate HR max

There are some methods to get HR max by running, such as running towards a hill at a high intensity and then ascending the hill at as high intensity as possible, or, on flat ground, running 400 metres at high intensity and then increasing the intensity for another 400 metres. These tests are considered stress tests, and individuals who are not highly active are not advised to take them [47]. Another method is based on interval running: two intervals, four minutes each, in which the trainee is short of breath, interspersed by three minutes of active rest, and finished by a third interval, where two minutes in the trainee increases their speed as much as they can and run for as long as they can. HR max is the maximum heart rate achieved [48].

Age-based methods to calculate HR max

As the running methods are generally too difficult to do 'on the spot', there have been attempts to calculate HR max based on readily available data.

The most commonly used formula to get one's maximal heart rate, $HRmax = 220 - age$ also known as the Fox formula, has been time and time again found largely inaccurate. Originally intended as a rough formulation, with cross-sectional data from participants who are by no means representatives of a population [49] and based on unoriginal research [50], a number of researchers have tried to find similarly simple but more accurate formulas, most of them claiming to have found the right one. One of the popularly used is the cross-sectional research of Tanaka et al., who formulated the equation $208 - 0.7 \times age$. This formula was later validated by Gellish et al. in 2007 by conducting an analysis of longitudinal data from participants from various groups and developing a similar formula $207 - 0.7 \times age$, however, both Fox and Tanaka were disproved by Sarzynski et al. in 2014 as not precise enough based on standard error of estimate (12.4 and 11.4 bpm respectively) of the two formulas [51]. In that study, Sarzynski stresses *the importance of finding and validating other measures to be used in exercise prescriptions for the determination of intensity of exercise, the estimation of fitness levels, and as a criterion for achieving maximal exertion.*

The HUNT fitness study carried out on over three thousand people by Nes et al. found a more accurate formula $211 - 0.64 \times age$, with the standard error of estimate of 10.8 bpm, but couldn't find evidence of interaction with gender, physical activity, VO2 max or BMI [52].

It seems that computational models for peoples' fitness leave a lot to be desired, but this can all be remedied with further research.

Chapter 4

Existing solutions

There's a plethora of applications taking advantage of wearable tech, not all of them using the available bio sensors, ranging from basic real-time heart rate monitoring to progress tracking, to calorie measuring, and to social network sized fitness communities.

In this chapter I will research a few such applications and judge them based on following criteria:

- Use of IoT possibilities – use of available sensors, data collection from the community,
- User fitness assessment – how the user's fitness is assessed (attempt self-assessment, fill out a questionnaire, or take a physical test),
- Community – support of interaction between users,
- Extra features – interesting perks that make the platform stand out among others,
- User-friendliness – navigation around the mobile application - finding the general settings, creating and clearing a route, general user experience (the applications were tested on an Android phone),
- Availability – the platform's economical model and its suitability,
- Cross-platform – major operating systems supporting the mobile application and sensors, if any are used,
- Propriety – whether the solution is open-source, closed-source or else.

4.1 komoot

The cross-platform application for outdoor track suggestion can also be used as a tour planner, a map and a navigation system.

Use of IoT possibilities

Given its use of GPS sensors, komoot does qualify as a basic IoT system, however, it also relies heavily on user input for route rating. The smart watch only gets used for displaying of routes and navigation, not taking any advantage of the available biosensors.

User fitness assessment

When a user is creating a route, they can set its difficulty as one of five levels, describing the user's self-reported physical fitness and their current will to take up a challenge (or lack thereof): *Couch Potato*, *Average*, *In Good Shape*, *Athletic*, and *Pro*. This parameter is considered when the app is generating a suitable route – presumably by trying to adjust the elevation profile of the possible routes. The user's real fitness is not taken into account.

Availability

The region-based pricing model allows users who do not travel too much to use the app for free, since the first region is provided at no cost. In the other pricing options Single Region, Region Bundle and All Regions – the price-performance ratio seems to grow at a reasonable scale.

Community

With features like sharing and recommending of routes users have taken, following of other users, upvoting and commenting on their posts, the mobile application integrates a full-fledged social network.

Extra features

Komoot supports multiple sports, mainly hiking, running and multiple types of biking, to which it tailors the parameters of planned routes, such as mostly paved roads for road bikes and gravel for off-road biking. There are other activities the app supported by the app, which, however, do not offer the customized planning and can only be tagged later. These sports include skateboarding, skiing, climbing, and plenty of others. This only serves as a tag for the user with no real impact on the app's functionality.

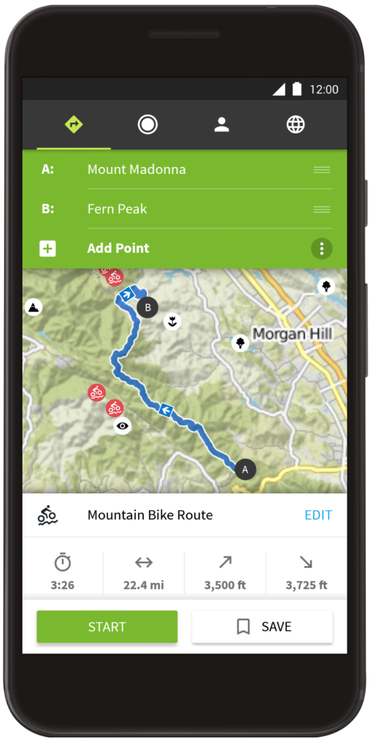
If a user decides to skip all planning and just go into the world, their route can be "recorded" for their use or the use of other komoot users, who can pick this route and either

a) follow it 'as is', with all the possible deviations from the known network of streets and paths with the risk of impassable roads, or

b) have komoot replan the off-grid segments and navigate through the known street network, or

c) edit the route according to one's liking.

Users can pick a starting point, a destination, as well as any number of waypoints in between for their route (see fig.4.1) Once the route is chosen, the user can go through the route's stats - the estimated time it will take to get from start to finish, its length, the elevation profile (see fig.4.2, 4.3) (uphill, downhill, highest and lowest points, estimated average speed) and the surfaces and their use in proportion to the route's length (see fig.4.4). All this information is delivered in easy-to-understand charts as well as interactive mappings – using a slider, the user can see which stat applies to which part of the route.



A: Mount Madonna

B: Fern Peak

Mountain Bike Route

3:26 22.4 mi 3,500 ft 3,725 ft

Figure 4.1: Route planning in the komoot app [2].

User-friendliness

The mobile app uses a bottom tab bar for navigation, however, the way the tabs are designed is not entirely user-friendly. The `Plan` and `Record` tabs are

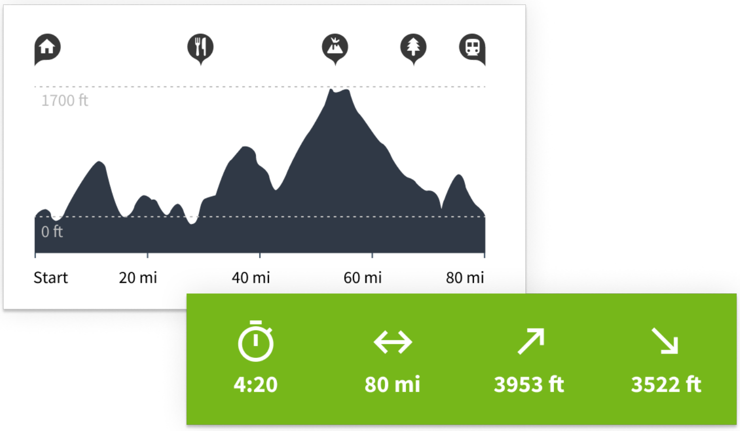


Figure 4.2: Route elevation profile [3].

clear enough, but the `Profile` tab contains a lot of information that does not seem entirely connected to the user's profile, such as people to follow and the app's settings. The last tab, titled `More`, only contains promotions for komoot offline maps and komoot premium, which is something a user does not expect under this tab. The navigation through the mobile app takes some time to get used to – it took me a while to find the Settings after I did not see them in the `More`menu on the bottom tab bar.

Once I created a route, the information provided was well-delivered and easy to read, however, there was no obvious way of completely cancelling the chosen route and picking another one. Instead, hiding in the Options of the route – which I had not noticed before – I found the "Reset route" option, which cleared the route. Overall, the app makers made some great UX decisions, but also some that do not seem natural to the user.

Cross-platform

The system is fully integrated with the Apple Watch and Samsung gear, and – at least limitedly – supports a number of other brands of smart watches and other Bluetooth-enabled devices. The mobile application runs both on iPhones and Android phones.

Propriety

The system is not entirely open-source, as a few of their repositories are public, but the core components remain proprietary [53].

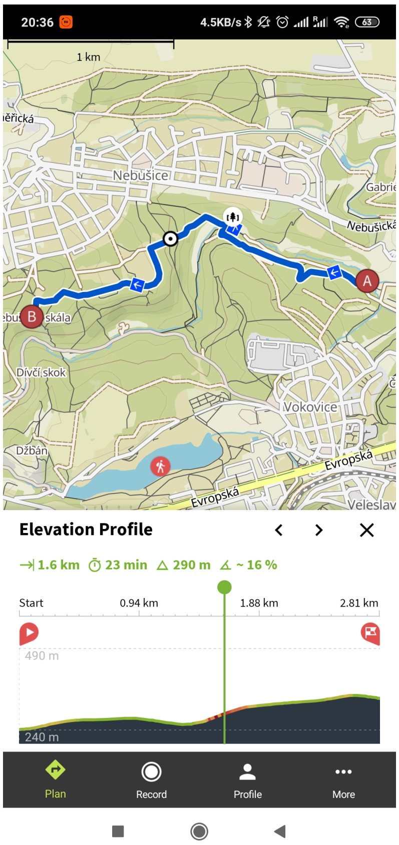


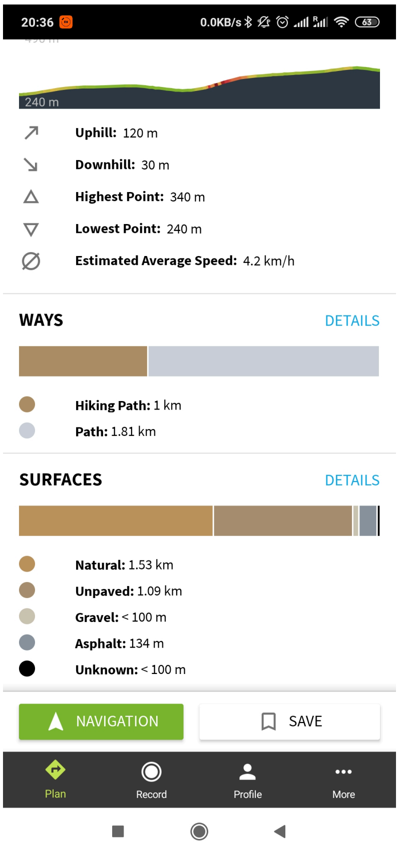
Figure 4.3: Overview of the route's elevation [4].

4.2 endomondo

Under Armour's endomondo application serves as a personal trainer for distance-based sports, such as running, cycling, or hiking. Their main focus is on the endorphin release people get from encouragement and motivation – this is why the app gives coaching advice, allows the users to send each other peptalks, and has over the course of its existence built a global fitness community.

Use of IoT possibilities

The application integrates with a wide range of watches and sensors, collecting data from GPS trackers, bicycle speed and cadence sensors, bike power sensors and heart rate measurement units. Some basic statistics based on this data (such as average heart rate) is presented to the user with a free plan, while the more interesting statistics displayed with interactive graphs are for premium users only – average heart rate history, heart rate zone analysis, even weather information collected from AccuWeather.com at the time of the workout.



Uphill: 120 m

Downhill: 30 m

Highest Point: 340 m

Lowest Point: 240 m

Estimated Average Speed: 4.2 km/h

WAYS DETAILS

● Hiking Path: 1 km
● Path: 1.81 km

SURFACES DETAILS

● Natural: 1.53 km
● Unpaved: 1.09 km
● Gravel: < 100 m
● Asphalt: 134 m
● Unknown: < 100 m

Figure 4.4: Overview of the route's surface [5].

The application used to have a feature called Peer Banchmark, which used other users' data to compare the user with the rest of the community, however, this feature was removed due to lack of use [54].

User fitness assessment

The premium version of the application promises personal exercise plans for running goals, and the heart rate zones are calculated using the Karvonen formula:

$$((maxHRrestingHR)\text{»}\%Intensity) + restingHR$$

However, endomondo's website does not mention how and when HRmax and HRmin are calculated, and the official support personnel does not have any additional information about what is written on their website [54]. In spite of that, the values for the Karvonen formula are pre-filled when a user accesses the page, so they probably use age-based formulas to estimate all the variables. In any case, they are all editable, so if the user already knows their HRmax for example from a laboratory test, they can use it. When creating a new training plan (that is, having endomondo calculate the “ideal” plan for how a person should train if they want to run a specific distance on a specific day in the future), one of the steps has the user choose a recent workout which will be used for calculating the individual’s fitness level and for the plan generation [55].

Availability

Endomondo comes in two versions – free and premium. The free version has basic functionality, which is so limited that some have opted for free alternatives with more useful features such as Strava or Runmeter [56]. It is supported by not too obtrusive ads on the bottom of the screen.

The premium version is ad-free, brings a number of necessary features such as heart rate zones, training plans, and others, and is based on a monthly subscription.

Community

Endomondo’s main focus in keeping their users on the platform is motivation, which is also demonstrated by the application’s slogan – “Free your endorphines”. One of their features is a real-time audio motivator, which makes comments on how well the user is doing compared to previous workouts, encouraging them if they are lagging behind or is informing them about the personal record they are be about to hit.

Competing against oneself, however, is not as much fun – and therefore as motivating – as getting to beat a friend or relative at the activity of choice. This is why the application also integrates a social network of millions of users with functions like sharing one’s activities, sending peptalks and challenges.

Extra features

After the end of the workout, you can answer a “How was your workout?”, however, since this is a plain text option, this data is only for the user’s later reading, and cannot be used in any machine processing.

A user can post photos in the description of the activity, as well as use tags which are useful on social media after sharing. With premium, one can filter based on these tags – useful for tracking the number of kilometres that the user has run in a specific pair of shoes, or biked with a specific chain. Other users can also be tagged, for example running partners or the person you are trying to beat in a challenge.

The premium version of the app also includes a few modes for interval training (periods of exercise alternating with periods of normal-to-high intensity training) as well as the option to build your own. There are many applications that support this feature – after all, it is just fairly basic time tracking – but it is nice to have all the functionality in a single app.

User-friendliness

The navigation in the mobile application is realized with a bottom tab bar, whose design is nicely done. The `More` tab hides a lot of functionality, effectively serving as an easy-to-reach hamburger menu.

Given the fact that that endomondo does not support planning of routes and merely follows where the user is headed, the `Workout` tab which contains a map is less crowded than in a navigation app (fig.4.5).

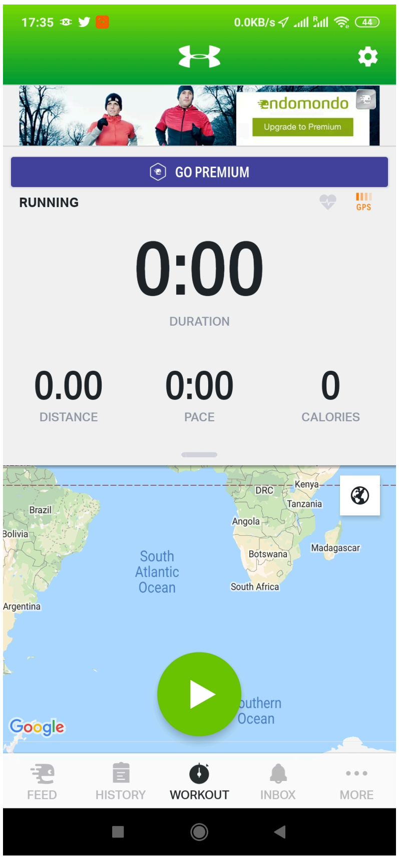


Figure 4.5: endomondo's main screen [6].

The main settings can immediately be found in the top-right corner of the main screen, or under the `More` tab, lost in between ten other options (fig.4.6). The application is easy to navigate and is fully made in, if somewhat blocky-feeling, material design.

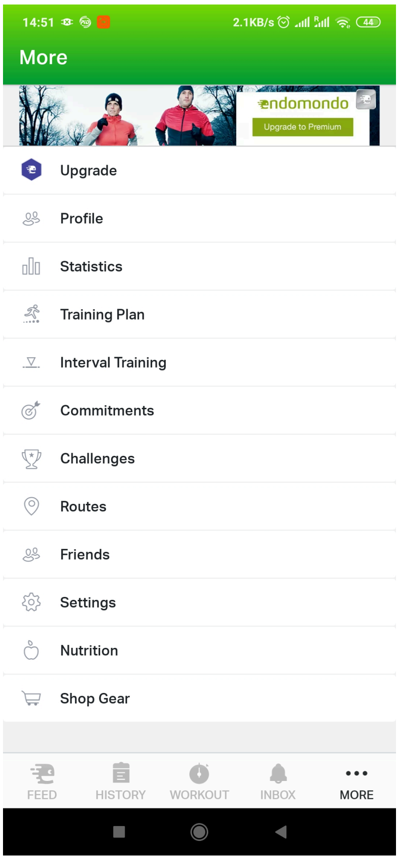


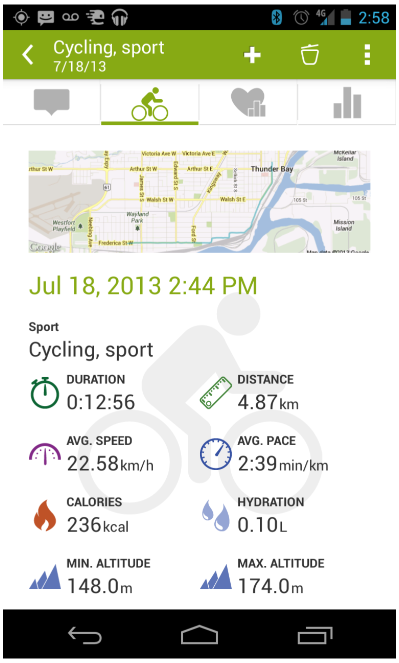
Figure 4.6: endomondo's `More` tab, with main settings lost in other options [7].

Cross-platform

All the features are available for iOS 9 and newer and Android 4.1 and newer. Endomondo also used to be available on Blackberry and Windows phones. Some features are only available in the web application.

Propriety

The whole endomondo application only has a single public repository on github, which is a fork of tapiriik – a tool for data synchronization between different fitness applications, such as Endomondo, Garmin Connect, Strava, Runkeeper, and others [57]. Otherwise, the whole application is completely proprietary.



Cycling, sport
7/18/13

Jul 18, 2013 2:44 PM

Sport
Cycling, sport

DURATION
0:12:56

DISTANCE
4.87km

AVG. SPEED
22.58km/h

AVG. PACE
2:39min/km

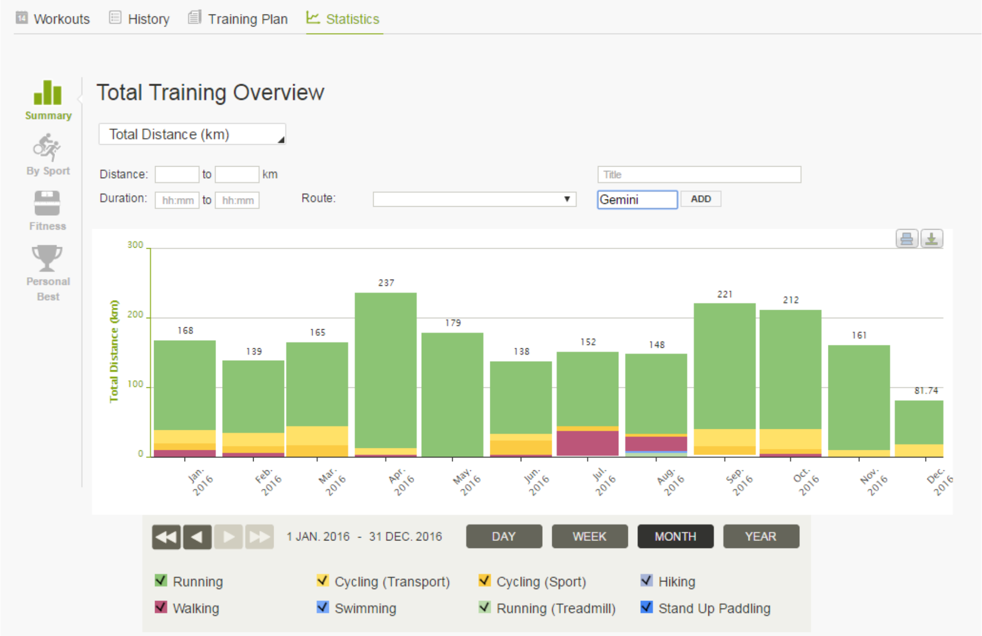
CALORIES
236kcal

HYDRATION
0.10L

MIN. ALTITUDE
148.0m

MAX. ALTITUDE
174.0m

Figure 4.7: Statistics of a biking trip on endomondo [8].



Total Training Overview

Total Distance (km)

Distance: to km

Duration: hh:mm to hh:mm Route:

Title

Gemini

1 JAN. 2016 - 31 DEC. 2016

☑ Running ☑ Cycling (Transport) ☑ Cycling (Sport) ☑ Hiking
☑ Walking ☑ Swimming ☑ Running (Treadmill) ☑ Stand Up Paddling

Figure 4.8: endomondo history example from the web application [9].

4.3 Strava

Strava is a social network for athletes that offers a large number of features mainly focused on tracking and analysis of users' activities, sharing, and competing [58]. The target sports of the platform are running and cycling, but it supports plenty of other activities. It has both a web and a mobile application, with some features shared and some only available in one of them.

Use of IoT possibilities

Strava makes good use of sensors in the users' devices – those for measuring heart rate, power, and GPS. The platform creates heatmaps of routes that are used the most, taking advantage of their tracking technology and the users' GPS sensors. They also use a novel way of determining the surface of routes, thanks to data not only from OpenStreetMap, but also from the users' bicycle frames.

User fitness assessment

The Strava platform includes plenty of user fitness-related components, such as the premium `Fitness & Freshness` feature, which shows the user how fit they are on a graph of past activities, and how fresh, based on the amount of exercise the user has recently done, all of this calculated using power data and heart rate or perceived exertion [59]. The platform also has heart rate zones, which are primarily calculated using an age formula, but the user can manually customize them.

Availability

Strava is available both for free and for an affordable monthly subscription.

Community

The platform has millions of active users who are connected in a large social network. It allows users to create clubs, such as groups of friends as well as racing teams. These clubs give the user the opportunity to keep track of the members' activity and exercise, with feeds and leaderboards.

For increased safety while outside, Strava Beacon is a premium feature that allows a user's friend to see their real-time location in case of an emergency.

Another community-building perk is the feed of challenges (fig.4.9), which help motivate the users with leaderboards and competitiveness.

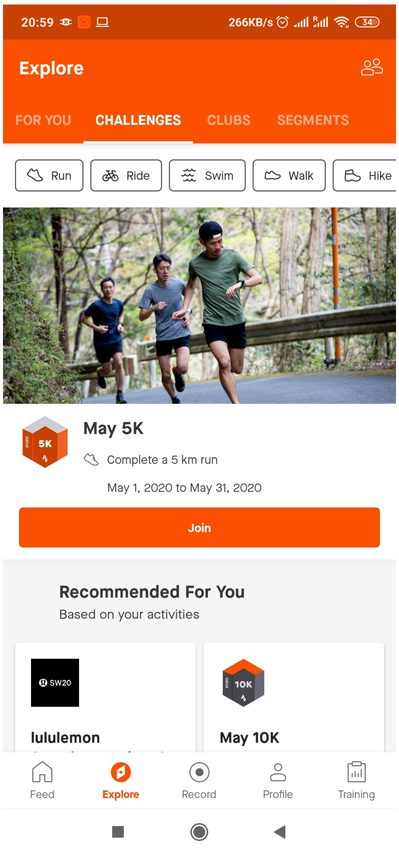


Figure 4.9: Challenges feed on Strava [10].

Extra features

One of the most popular premium features is called Segments (fig.4.10), where users can compare their times on specific portions of road or trail using leaderboards. This is supported by heatmaps of use frequency of specific routes. It also allows the user to input their pictures, gear, and perceived exertion to the activity session. The premium version of the platform offers a lot of features that are somewhat less connected to fitness but make it successful as a social network.

User-friendliness

The mobile app uses a tab bar for navigation, with nicely thought out tabs - `Feed` tab for posts from the community, `Explore` tab for challenges, clubs, and segments, the central `Record` tab for the actual activity, which is simple and clutter-free, with the options to change the sport, pick a route, and turn on the Strava Beacon. Another tab `Profile` keeps the user's account details

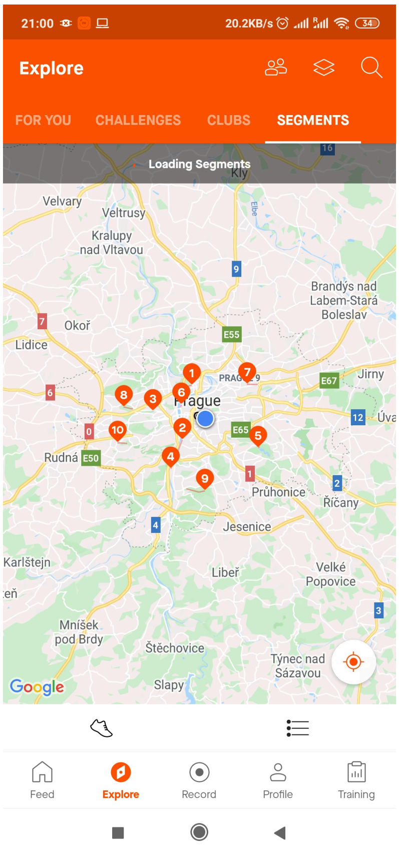


Figure 4.10: Strava segments near the user's location [11].

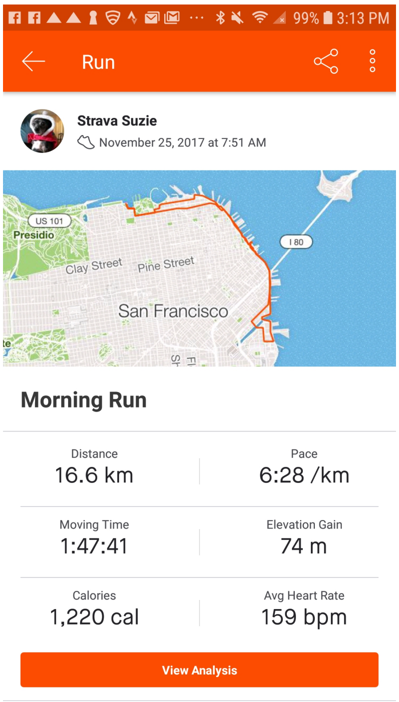
and fitness information, such as their stats, activities, and training log. The last tab `Training` contains the user's training plans, which are part of the premium subscription.

One thing that I missed was the option to create a route on the phone – this can only be done in the web app and then synchronized with the mobile app, making the mobile application less suitable for ad hoc navigation.

Another unusual design decision was putting the `Find friends` button only on some pages.

Cross-platform

The mobile app Strava is available both on Android and iOS, as well as providing a web application. It supports a variety of smart watches, power sensors, and other devices.



Strava Suzie

November 25, 2017 at 7:51 AM

Morning Run

Distance 16.6 km

Pace 6:28 /km

Moving Time 1:47:41

Elevation Gain 74 m

Calories 1,220 cal

Avg Heart Rate 159 bpm

Figure 4.11: Statistics of a run on Strava [12].

Propriety

The application is closed-source, but it offers a publicly available API for custom development.

4.4 Garmin Explore and Garmin Connect

The vast ecosystem Garmin has created for their users is filled with an assortment of wearables, radars, smart lights, navigations, customized maps, and plenty more, providing features which are presented to the user via modular mobile apps.

Here I will analyse the Garmin Explore application, which is targeted at hikers, in tandem with Garmin Connect, which functions as a fitness tracker.

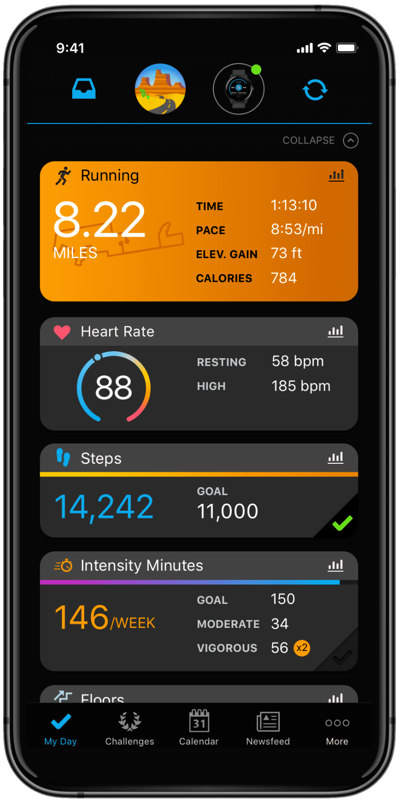
Use of IoT possibilities

Most fitness-focused Garmin watches do have a heart rate sensor and all of them include a step counter, however, this data is never compared with that of other users, keeping focus on the user's activity history with no prediction. The Garmin Connect app includes an "Insights" feature, which compares the user's simpler data, such as daily number of steps or hours of sleep, to that of all other users.

User fitness assessment

Some of the heart-rate-monitor enabled watches allow a user to set the zones (see chapter on fitness assessment) in which they would like to keep their heart rate depending on the activity they choose to do. The Connect app also allows users to manually input their VO2 max, but doesn't test the user on it.

As mentioned in the Fitness assessment chapter, they offer the Performance condition stat, which however, according to user feedback, isn't useful or accurate.



Running

8.22 MILES

TIME 1:13:10
PACE 8:53/mi
ELEV. GAIN 73 ft
CALORIES 784

Heart Rate

88

RESTING 58 bpm
HIGH 185 bpm

Steps

14,242

GOAL 11,000

Intensity Minutes

146/WEEK

GOAL 150
MODERATE 34
VIGOROUS 56 x2

Floors

Figure 4.12: My day - Garmin Connect's home screen displays statistics of the user's recent activity [13].

Availability

In general, Garmin's pricing model relies on the user buying one of their high-end smartwatches, whereupon the rest of the components (apps, basic maps, support, etc.) are free, with some exceptions, such as advanced maps. The company's smartwatches are considered premium quality, with a wide price range - target groups include *potential Rolex buyers* [60] as well as ordinary users [61].

Community

The Connect application handles the data from the point of view of a fitness tracker, with social features like groups, competitions, likes, comments, and badges of accomplishment [62].

Extra features

The Explore application is very simple, mostly just containing the bare necessities such as the map, route planning, and a history of activities. It doesn't even contain navigation options; it is hard for me as a user to understand what its use is. It allows the user to group waypoints, tracks, routes and activities into collections for each trip. The Connect application, as a fitness tracker, gives insights into the user's fitness, allows to track the lifespan of the user's gear, their sleep quality, water intake, menstrual cycle (as this can affect the user's stamina), and other factorss.

User-friendliness

The modularity of Garmin's apps allows for cleaner, easier-to-navigate interfaces.

The Garmin Explore app uses a tab bar for navigation; and while the four tabs do not have any inscriptions, the icons manage to communicate fairly clearly what the user can find in them. In the Map tab, it took me a while to find the button for creating a new route or waypoint (top-left corner in fig.4.13), since it does not stand out from the other UI components. The actual track creation process feels natural, visual and light-weight, with fast waypoint creation and editing, however, I lacked the option to search for a place that I would want to use as a waypoint. A large disappointment for me as a user was the fact that I did not get a planned route – only the polyline defined by the waypoints, which seems completely useless for navigation purposes. Once the route was saved and shown on the map, I could not figure out how to cancel it.

Another interesting decision is the Library tab, which contains the user's past activities, saved waypoints, routes, and tracks. The difference between the (otherwise synonymic) words "route" and "track" is that a route is the generated polyline, whereas a track is a set of breadcrumbs of a past hike.

The Connect app, as the main fitness app in the Garmin ecosystem, entails a larger number of features, resulting in a full tab bar as well as a hamburger menu, which I found a little confusing.

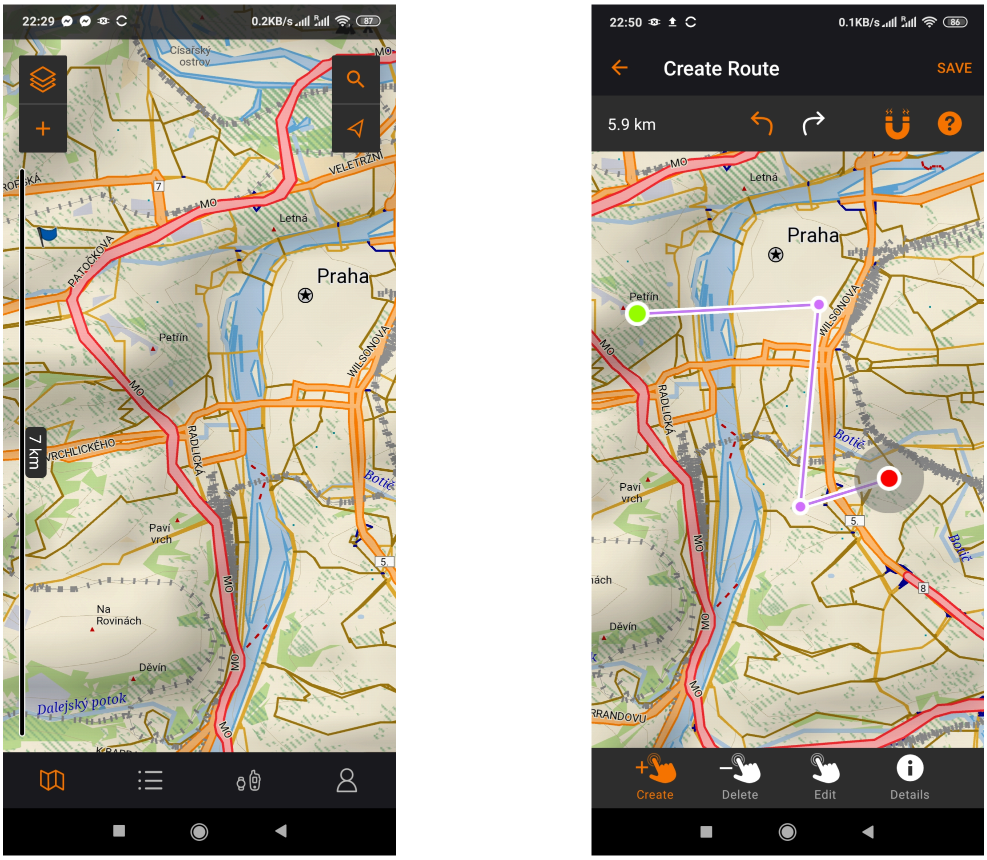


Figure 4.13: Garmin's map tab [14] and route creation screen [15].

Cross-platform

All of Garmin's applications can be installed both on Android and iPhone. However, they can only be paired with watches inside Garmin's ecosystem, which is a drawback for users who already own a smartwatch.

Propriety

Some of the Garmin applications are completely open-source, either due to the software that they are derived from, or simply to provide the general public with opportunity to customize their Garmin experience [63] [64]. These apps are mostly concerned with map handling, navigation and sound codecs, but also include some bits of functionality like widgets, low power watch apps and others.

While some of this code may be used in the Explore or Connect applications, these repositories don't contain their core functionality.

4.5 Summary

All of the studied platforms are used by a wide community of users, are either mostly or completely closed-source, base their maps off the OpenStreetMap, and have their own ways (or lack thereof) of integrating IoT principles in their design.

Komoot is a great app for navigation; its flexible way of route planning gives the user a good idea of what they can expect on their trip, and includes navigation. However, the information provided is not tailored to the user's actual fitness level.

Endomondo tracks the user's progress and motivates them. Since there is no route planning, it doesn't estimate the difficulty of outdoor tracks – instead it serves as a personal trainer.

Strava is great at tracking and analysing data from previous activity, and at providing new ways for athletes to connect and motivate each other. It gives the user a good insight into their data, and allows comparison between users on pre-defined segments. But again, there are no estimations done or tests to be taken.

Garmin Connect is a great fitness tracker, but the Explore app is not useful for route planning or navigation.

None of the apps fully customize their assessment of the user's fitness to estimate how they will do on their future hikes, which is a core feature of my application.

There is an abundance of community features in all of these apps; it is reasonable to consider how many of them are enough to keep the users entertained and how many become overwhelming.

Chapter 5

Functionality

I decided to construe the user-focused features as one would in a real-life agile project, as this is in my experience more straightforward and easier to alter when necessary than flooding the reader – for example, a developer – with a long list of elaborate use cases right away. The discussions that take place before reaching a specific decision should be well documented – brief enough and detailed enough at the same time to provide good value after a quick read. If the reader has access to a good transcript of these discussions, it is less likely that they will misunderstand the requested functionality; in the best-case scenario, if the reader is the potential developer, they are an active part of those discussions, so they have a substantial impact on the outcome, thus diminishing the "coding monkey" phenomenon.

The top-down approach of the agile methodology which is perhaps the most common and natural to grasp is creating epics, decomposing them into user stories, and with good knowledge of the project's architecture and after due consideration, adding technical subtasks in an issue tracker of choice.

Epics are complex, high-level characterizations of desired functionality, which usually take longer to deliver than a few weeks. An epic is usually comprised of description and multiple user stories, which cover the extent of the epic in higher granularity. Once all the stories are delivered, the whole epic is considered finished. Epics are sometimes formulated using the same format as user stories.

User stories (US) are short, simple descriptions of a feature told from the perspective of the person who desires the new functionality, usually a user of the system [65]. They are not system specifications or functional requirements. Rather, they are the beginning of a conversation that can lead to such specifications or requirements. However, they provide context and scope of the requested feature. Often they are written in simple, non-technical language, as most systems do not have tech-savvy users. The typical template for a user story is:

As a <type of user>, I want <some goal>, so that <some reason> [65], where the reason part is optional but recommended. Another important component of a US are *acceptance criteria*, which define the functionality which needs to be working for the story to be finished, often including not only the happy path, but a sad path as well. In general, acceptance criteria should be testable, which is why they are often used as starting points by testers, to get a better idea of what has been done, and provide them with some inspiration.

Based on the analysis of existing solutions, I will now describe the most important features that should be provided by my software.

The main focus of this chapter will be to document my internal analytical discussions. I will always start with a high-level description of the requested feature formulated as an epic or a user story and elaborate on its details incrementally, summing all of this up in acceptance criteria. Some stories will also contain hints toward completion for the potential developer, which will be based on research done in previous parts of this thesis. Since a part of this thesis will focus on implementing a proof of concept, some of the most imperative stories will also be fully processed and finished.

The platform will be referred to as the App.

5.1 Heart rate monitoring

The application should monitor a user's heart rate unobtrusively and reliably.

The user needs a method for getting their data to the mobile application and then to the backend, do it comfortably, preferably without any need for direct user—computer interaction.

5.1.1 Manage devices paired with the App

> As an App user, I want to pair and unpair my wearables with the App,
> so that the App can use the data of all devices that I currently own.

The app should provide the user with the list of the phone's paired devices, so that they can choose their fitness gadgets to use on their next hike. However, since the App can only support some wearable vendors, the ones that are not compatible should not be pairable. Also, if this is programmatically possible, non-wearables should not be displayed in the list, or at least should not be pairable, so that the user does not get confused by the presence or pairability of their Bluetooth speaker.

Acceptance criteria

- The user can pair any number of compatible wearables.
- The user can unpair any number of compatible wearables down to zero.

5.1.2 Activate a device

> As an App user, I want to mark a wearable I'm using as active, so that only its data is relevant to the statistics.

The user should be able to activate a device they want to use for monitoring of current activity or activity in the near future.

Acceptance criteria

- The user can mark a device as active.
- When a device is active, the platform can receive data from it.
- The user can mark a device as inactive.
- Only one device at a time is active; if activating a new one, the old device must be deactivated.

5.1.3 Choose and integrate a wearable

> As an App user, I want my heart rate-enabled wearable to work with the App, so that I do not have to buy a new one.

Based on the analysis of existing applications, the most often used type of wearable tech are smartwatches. While most applications support other devices, such as chest straps, watches are the most convenient and wide-spread gadgets to be used by the general public.

Vendors are discussed in the PoC implementation chapter.

Acceptance criteria

- A popular kind of wearable can communicate with the App in both directions.

5.2 Objective user fitness assessment

> As an App user, I want my fitness to be assessed, so that I can get relevant estimations of route difficulty.

Having analyzed the conventional ways of user fitness assessment in a previous chapter, the application should allow a user to use one of the multiple ways to get their fitness level. The outcome of the assessment should be the individual's VO2 max index and their heart rate zones – so that if they need to maintain their heart rate in a specific range, they can cross-reference it with the exhaustion they perceive and learn to recognize when they are in the desired zone.

If the user has no physical impairments, they can have their HR max assessed by a simple age-based formula, such as the one developed by Nes et al., or take one of the existing HR max tests knowing that they are dangerous when not done right (see the chapter on fitness assessment).

Users with physical impairments could self-evaluate how much their condition affects their fitness, and based on this their HR max can be set by using one of the standard formulas and subtracting a smaller or larger number of beats per minute (further research needs to be done on how many beats this should be). This self-assessed value should be corrected as more data is available about the user.

As far as objective assessment, I see plenty of potential for the use of machine learning and neural networks. The networks can learn from the data collected from a good number of users over a longer period, and then categorize the new users into fitness groups based on this data, which will then be used to predict the users' future hikes. More research is needed in cooperation with data scientists and physical therapy professionals, in order to develop reliable models for fast difficulty assessment of a route for a specific individual.

5.2.1 Implement support for VO2 max tests

> As an App user, I want to be able to take a guided test of my fitness in the App, so that I do not have to look for tests elsewhere.

The App should implement a guide to multiple tests of VO2 max, with instructions on what they should do, and signals to direct the user while taking the test. The instructions should also contain a description of the signals the user will receive. The initial instructions should be textual and possibly audible, so that the user understands the aim of the test before taking it. During the test, a watch vibration should alert the user about an incoming signal which will be displayed on the watch. This signal would contain directions like 'Turn in N seconds' where N is a countdown to 'Turn now!'. When no signals are occupying the device's screen, there should be pep talks like 'Keep going!' and 'Great job!'. At all times during the test, there should be the test status on the device's screen: remaining time, distance walked, and any other metric relevant to the test.

As it is the most accurate of the more feasible tests, the application should encourage the user to take the 6-Minute Walking Test, and let them take it using the app in both 15- and 30-metre-long variations. In order to calculate VO2 max, the App can use Mänttäri's formulas.

Acceptance criteria

- The user can take at least one VO2 max test using the App.
- During the test, the user gets directions on their smartwatch.

5.2.2 Implement support for HR max and HR min tests

> As an App user, I want to be able to take a guided test to find my heart rate zones in the App, so that I do not have to look for tests elsewhere and get as accurate heart rate zones as possible.

The implementation should be similar to VO2 max tests; if possible, both metrics should be measured by a single test to reduce the time a user has to spend setting up the App.

Acceptance criteria

- The user can take at least one HR max test using the App.
- The user can take at least one HR min test using the App.
- During the tests, the user gets directions on their smartwatch.

5.3 Manual input of biometric data

The users might not want to take an exhausting test at the time of setting up the App. This is why they should be allowed to input the data in other ways.

If a public universal database with all people's medical data existed, it would make sense to get the necessary information directly via an API. However, as no such thing exists (and for good reasons), we will let the user enter their biometric data manually.

5.3.1 Allow manual input of basic biometric values

> As an App user, I want to provide the App with my basic metrics, so that the results are relevant to me.

If the user has a smart scale, it could be integrated to get the most recent values, however, that seems like a bit of an overkill, considering that it's just one number - the weight.

The App needs the user's height, weight, biological sex and age. The age should be calculated based on their date of birth, so that it is always reasonably accurate. However, I myself have had issues with disclosing my exact date of birth to just any application that asks for it, and the accuracy would be good enough with even just the month and year. Yet again, it is reasonable to consider whether a user who does not have an issue giving out their heart rate and fitness data would show similar concern.

Acceptance criteria

- User can enter their basic metrics - height, weight, age, sex - manually.

5.3.2 Allow manual input of advanced biometric values

> As an App user who has recently done lab fitness tests, I want the App to use these values instead of taking the App's tests, so that I do not have to take fitness tests that are most probably less accurate than what I know.

A user should be able to manually enter the VO2 max, resting heart rate and maximum heart rate, as well as the severity of the effect a potential condition might have on the individual's fitness. The App should ask whether the user has such a condition and if there is a maximum heart rate (perhaps recommended by a doctor) the user should not exceed when active.

Acceptance criteria

- User can enter their advanced biometric data - VO2 max, HR max, HR min - manually.
- The App asks about the severity of any conditions that the user has and its impact on their fitness.
- The App asks about a recommended maximum heart rate the user should not exceed.

5.4 Routes and maps

5.4.1 Integrate a map

> As an App user, I want to use a map to visualize my hikes, so that I can follow them easily.

Since the App is meant to be used as a hike planner, the user needs an easy to follow aid to help them plan their routes. The whole App should be focused around the hikes a user takes and around their difficulty given the user's fitness level.

Acceptance criteria

- An interactive map is displayed in the App.

There is not a large variety of maps available for use by developers. While Google Maps would probably be a popular choice, Google decided to put their maps behind a paywall [66] in 2018. Therefore it makes sense to choose the open-source OpenStreetMap [67] and use one of its derivatives with a public API for route planning (such as OpenRouteService [68]).

It is important not to forget to credit the map's creator in their preferred way.

5.4.2 Plan a route

> As an App user, I want to plan a route from point A to point B.

Hikers often do not want the easiest, fastest, or shortest possible path from A to B – they look for a challenge and nice things to see. This is why they might make extra effort to take their hike through a specific spot, which might not be on the ideal route. Therefore, it should be possible to add such places to the route.

Since there are often multiple ways to get from A to B even through the chosen waypoints, the user should be able to choose which one they want to take.

Another option that is related to route planning is choosing whether or not to plan a round trip. Especially in hiking, when a lot of people come to the starting point by car, they want to return to the same point where they started. In order to keep the route interesting, the way back should be different than the way to the destination point, if possible.

Acceptance criteria

- User can pick a starting point from the map.
- The user's current location can be automatically found and set as the starting point.
- User can choose points on the map through which they want to pass.
- User can choose a destination point from the map.
- User can remove points from the planned route.
- The points in a planned route can be rearranged in any order.
- If the last waypoint is removed, the previous waypoint becomes the last point.
- If multiple different routes are available, the user can compare their attributes.
- If multiple different routes are available, the user can only choose one of them.
- Allow the user to do a round trip.

5.4.3 History of hikes taken

> As an App user, I want to see which routes I've hiked while using the App, so that I can remind myself of the impressions of the hike.

A user might want to remind themselves of a specific hike they went to – if it's because of fond memories, or just to make sure who went with them, or when it happened. This is why it's important to keep a log of the hikes the user has taken ever since logging in to the App.

Every route is planned over a network of geographical coordinates, which means one can create thumbnails from its projection on a map. This, along with the names of nearby prominent landmarks, such as the mountain range through which the user hiked, and the date of the hike, should be enough for the user to identify the route they were searching for. Since there is a number of attributes a route can have, it should be possible to filter by them. Filters should include the date on which the hike was taken, the time it took to complete it, the route's length, and other quantitative attributes.

Acceptance criteria

- Show a filterable list of hikes taken by the user.

5.4.4 Saved routes

> As an App user, I want to save some routes, so that I can do not have to create the same route again when I want to retake it.

Since the process of creating a route from scratch is somewhat lengthy, the user should be able to save it and find it later.

Acceptance criteria

- User can plan a brand new route and save it.
- User can choose an existing route made by their friend and save it.
- User can remove a route from saved routes.
- User can choose a saved route and hike it again.

5.4.5 Share routes

> As an App user, I want to link routes, so that I can discuss with my friends if we want to take it.

A group of friends is likely already using some sort of messaging, so sharing a link to the route and its details can make discussions about where to go faster.

Acceptance criteria

- User can get a link to a route.
- When the link is opened, the route's details are shown.

A group of friends is likely already using some sort of messaging, so sharing a link to the route and its details can make discussions about where to go faster.

Acceptance criteria

- User can get a link to a route.
- When the link is opened, the route's details are shown.

5.4.6 Display planned route details

> As an App user, I want to see interesting information about a route that has been planned for me, so that I can make an informed decision whether or not to take it.

When tapping a route, its details should be shown: its outline on a map, length, terrain in its segments, its elevation profile, colours of each official hiking route it uses, and interesting places to see along the route. This is also the place for information that is customized for the user's fitness – such as how long it will probably take them to hike it, how difficult its segments will be for the user in terms of perceived exertion and expected heart rate zones. This should take into account the limitations of the user's fitness, which they may have entered according to 5.3.2.

Acceptance criteria

- User can interactively see basic details of a planned route, including the route on a map, its length, terrain analysis, and elevation profile.
- User can see customized biometric details of a planned route, including the amount of time the hike will take them, the relative difficulty of the route's segments, and expected heart rate zones.

5.4.7 Display old hike details

> As an App user, I want to see interesting information about a route I've hiked, so that I can check it in a few days and show my friends.

Acceptance criteria

- User can see all the basic info as on a planned route.
- User can see the originally estimated values of biometric indicators and real values.
- User can see a graph of their heart rate during the hike.

5.4.8 Hike a route

> As an App user, I want to hike the route that I planned before.

Navigation is a core feature of this App since the planned hike is almost useless if the user cannot follow it in the App. But since it is not the only important feature, it should also be possible to temporarily escape the navigation mode, so that the user can do other things in the App.

Acceptance criteria

- User can start a planned hike on the day it was planned for.
- User can follow the directions of the App to stay on track.
- Once the destination is reached, navigation mode is automatically turned off.
- The navigation mode can be temporarily put in the background to use other parts of the App.
- Information about current segment is shown – the slope, the user's heart rate zone.
- While hiking, the user can see if they're following the plan – whether they're on time, or should go faster.

5.4.9 Change the hike date

As an App user, I want to replan my existing planned hike to a different day, so that if the weather is bad, I do not have to cancel it and plan all of it again.

Acceptance criteria

- User can change the date of a planned hike they have planned.

5.5 User profile

The user needs a place to review and change their provided information, as well as a possibility to take the provided tests.

5.5.1 Management of provided biometric information

As an App user, I want to be able to change the information I've entered into the App, so that the values are always up to date and my estimates remain accurate.

Acceptance criteria

- User can add and delete weight entries.
- User can edit the height entry, the recommended maximum heart rate, and the severity of potential health conditions.
- User can edit the advanced biometric data.

5.5.2 Profile picture management

As an App user, I want to be able to upload and delete my profile picture or use one from social media, so that my friends can recognize me in their friends list quickly.

Acceptance criteria

- If the user does not have a profile picture, there is a substitute placeholder.
- User can upload a profile picture.
- User can see the new profile picture once uploaded.
- User can delete their uploaded profile picture, which is replaced by a placeholder.

5.5.3 Log in options

> As an App user, I want the App to recognize me on different devices, so that my data is still connected to me when I change my phone.

People change their phones every few years, some even months, but want their new phone to have all the data as the old one. This can be done with manual backups. However, since our App has a server where most of the data is stored anyways, the data should be linked to the person's account and not the phone. The user should be able to sign up with e-mail, as well as social media platforms so that multiple user groups are targetted.

I considered allowing the user to use the App without an account, at the risk of them losing all their data, but it would not only be a nuisance for the user – a lot of processing power would have to be reinvested into the same user, and the previous data would not be as useful as they could be.

Acceptance criteria

- User can create an account using their e-mail.
- User can create an account using at least one social media platform.
- User can sign in to the same account on different devices and see all their data.

5.6 Hike with friends

Since hiking is generally an activity for small groups of people, it makes sense to be able to create groups in which everybody will know the hike's details and will be able to navigate.

Here I was able to discern between two main use cases – on the one hand, a user knows what is the group of friends they want at the hike and wants to just find a good route, and on the other hand, they want to hike a specific route and want the right people to join in. There is a third use case, the mix of the previous two, where the hike host knows about some people they want at the hike (for example their spouse) and would just like people to join in if they want to.

5.6.1 Adjust the calculations to friends' fitness

> As an App user, I want the estimations to be accurate even if I'm hiking with friends.

In a diverse group of people, someone's fitness level will likely be lower than that of the others, which means that the whole hike will take longer than if this person was not in the group. This is why the estimated time spent on the hike should be calculated based on the weakest link, and each participant's expected heart rate zones and all other biometric estimations should be adjusted accordingly.

The estimations can be adjusted either to the weakest participant's capabilities, or the system can acknowledge the motivation that they will feel when with others and slightly overestimate.

It would not be user-friendly to expose the name of the weakest individual, as they might feel ostracised, disheartened and less likely to go on hikes with their friends for fear of holding them back (and, by extension, stop using the App that made them feel that way). But at the same time, the aim of the solution is to provide accurate estimations for its users. This is why it should provide the information carefully.

Acceptance criteria

- When a hike is planned for a group of people, the predicted total time should be based on the time it would take the least fit user.
- The biometric values such as heart rate zones should be recalculated for all participants based on the total time the hike should take.

5.6.2 Invite friends to hikes

> As an App user, I want to invite friends to my hikes, so that we all have access to the same track when hiking it and preparing for it.

The invitation should include the proposed date of the hike and all the hike's details, including the time the hike will take if the user joins the group.

Acceptance criteria

- When user A is planning a hike, they can invite their friend user B to join.
- When user A invites user B to a hike, user B receives an invitation with the hike's details.

5.6.3 Refuse an invitation to hike with friends

> As an App user, I want to refuse invitations to hikes that I do not want to attend, so that the hike organiser knows I will not come and my invitation list is not cluttered.

Acceptance criteria

- When user A receives an invitation from user B, user A can refuse the invitation.

5.6.4 Accept an invitation to hike with friends

> As an App user, I want to accept invitations to hikes that I want to attend, so that I get updates and plan around the event.

Acceptance criteria

- When user A receives an invitation from user B, user A can accept it. From then on, user A is an active participant of the hike.

5.6.5 Add friends

> As an App user, I want to add new friends, so that I can invite them to hikes.

The only people that can be added need to be users of the App. In order to add friends, the App can use user profile links (primarily textual, but QR codes could be a popular option) and social media. A nice-to-have way would be using the phones' proximity if the two people are close to one another, for example with Bluetooth or NFC.

Acceptance criteria

- User A can send a friendship request to user B via a link to user A's profile.
- User A can send a friendship request to user B via popular social media.
- If the request is accepted, user A can invite user B to hikes and vice versa.

5.6.6 Remove friends

> As an App user, I want to remove people I no longer meet with from friends, so that they cannot invite me to hikes anymore.

Acceptance criteria

- If user A and user B are friends, both can remove the other user from their list of friends.
- If user B is removed, neither user A or user B can see each other's profiles or invite each other to hikes.
- If user B is removed, the hikes (and related data) user A and user B took together remain in their respective histories.

5.6.7 Accept a friendship request

As an App user, I want to accept a friendship request that someone sent me, so that we can invite each other on hikes.

Acceptance criteria

- When user A sends a friendship request to user B, user B can accept the request.
- When user B accepts a friendship request from user A, they can both invite each other for hikes.
- When users are friends, they cannot send new friendship requests to each other.

5.6.8 Dismiss a friendship request

As an App user, I want to dismiss a friendship request when I do not know the sender or do not want to hike with them, so that my request list is not cluttered.

Acceptance criteria

- When user A sends a friendship request to user B, user B can dismiss the request.
- When user B dismisses a friendship request from user A, neither of them can invite the other one for hikes.
- When user A sends a friendship request to user B, user B cannot send a friendship request to user A; they can only accept or dismiss the existing request.

5.6.9 Share a link to a profile

As an App user, I want to share my and my friends' profiles, so that we can connect more easily.

Acceptance criteria

- User A can share a link to their own profile.
- User A can share a link to their friends' profiles.

5.6.10 Joinable hikes

As an App user, I want to let my friends join my hikes, so that if I do not know who exactly I want there, my friends can decide for themselves if they want to come and I do not have to text everybody myself.

When planning a hike, not always do we know who might want to come, and sometimes we really want to try out a specific route. This is why the App should allow the user to not only invite people, but also allow the user's friends to join if they want to. The user's friends need to find out about the upcoming plans, this is why the App should have a feed of the community's plans.

At the same time, if the hike is already meant to be long, the user will not want someone to simply join because they want to, if it should mean extra two hours added to the total time. That is why the potential participant's request should be reviewed by the hike's host. This request should let the host know how much time this new person would add to the hike.

To look at this from the side of the requester, they should also be able to tell if the hike is going to be too hard for them, and whether them joining would make the hike ridiculously long.

Another point to consider is the capacity of such a hiking trip – the hike's host should be able to stop the joining requests when they've decided there are enough or the right people, and receive them again if somebody cancels.

Acceptance criteria

- When planning a hike, it can be marked as joinable.
- User can see their friends' upcoming plans and their difficulty, relative to this user's fitness.
- User A can ask to join user B's joinable hike.
- When user B receives a request from user A to join one of their hikes, they can see how this changes the hike's total time so that they can make an informed decision.
- A hike can be marked as `not joinable` after being joinable and vice versa.
- When a hike is marked as `not joinable`, all pending joining applications should be dismissed.

5.6.11 Cancelling a hike

> As an App user, I want to let my friends know that I will not be able to go to the hike, so that their metrics are accurate when I'm not there.

When different plans come up, a user will want to cancel their participation on a hike. This can happen both to an invited attendee and the hike's host. This is why it should be possible to appoint a new host from the invited friends if the original host needs to cancel – there should always be a host.

Acceptance criteria

- When user A accepts an invitation from user B, user A can let user B know that they have changed their mind and will not attend. The total time and the attendees' biometric estimates should be recalculated.
- A hike's host can cancel, but they have to appoint a new host.

5.6.12 Hike active on all participants' phones

As a guest on a hike, I want to be able to navigate, so that if the host's phone battery dies, there's someone to keep track of the route.

Acceptance criteria

- When a hike is started, it is started on the phones of all attendees.

5.7 Notifications

Everything important that happens on the App should have notifications.

5.7.1 Notifications for friends tab

As an App user, I want to receive notifications about friend requests.

Acceptance criteria

- If user A sends user B a friendship request, user B receives a notification.
- If user A sends user B a friendship request and user B accepts, user A receives a notification.

5.7.2 Notifications for planned hikes

As an App user, I want to receive notifications about my planned hikes.

The hike host should receive notifications about activity of this hike: join requests, accepted and refused invitations.

A hike guest should receive notifications about activity of the hike: change in attendance, change of date.

Everybody who attends the hike should get a notification about the hike's planned date coming up soon (for example, in a week, and on the next day).

Acceptance criteria

- If user A requests to join user B's hike, user B receives a notification.
- If user A sends user B a join request and user B accepts, user A and user C (another hike guest) receive a notification.

- If user A as the host of the hike changes the date of the hike, all attendants except for the host should receive a notification.
- If user A's hike is planned to seven days from now, every attendee receives a notification.
- If user A's hike is planned next day from now, every attendee receives a notification.

5.8 Notes

Since the platform needs data from similar users to make estimations on routes, all features except for navigation require an active Internet connection. Navigation is working with already existing plans and reception in mountainous regions isn't ideal, which is why maps for planned routes should be downloaded.

5.9 Considered features

One feature that could help make the platform into a social network is messaging. I decided against it because everybody is already using their messaging apps of choice to communicate and it would be distracting the user from planning their hikes. However, letting people see their friends' previous hikes and letting them react to all that is happening could be a good community feature.

Some other features that would be interesting to see in the application are

- integration with geocaching [69] for gamification,
- showing the weather forecast for the planned day in the details of the hike,
- user groups for users who always hike together, so that they would not have to search for their friends every time,
- multiple hike hosts, so that there are multiple people who can oversee the hike's activity,
- collecting data of non-planned, spontaneous hikes,
- cancelling the hike once it has started, so that if the user decides they do not actually want to go, the App does not force them,
- pausing the hike so that the attendees can have a lunch break,
- optional notifications for when the user is not following the plan (when being too slow or wandering off-route).

Chapter 6

Mobile app UI design

Based on the desired functionality of the platform described in the previous chapter, I was able to create a design for the user interface – UI of the mobile application.

In the beginning of this UI design project, I first looked at what would be the platform the project should run on. Since it is a mobile application, the range of options was slimmed down to two – either Android or iOS, given their prevalence on the market (Android on the first place with 70.68% of the market share, iOS second with 28.79% as of April 2020) [70]. It is partially due to Android's crushing lead on iOS devices, but also because of my personal experience with the more popular OS, that I decided to design the app for Android.

One of the most important decisions to make when designing an application is the navigation in it, and from there there's only a small step to information architecture (IA) – the *science of organizing and structuring content of the [. . .] mobile applications* [71]. The IA of an application helps the designer decide which concepts in the application are high-level and discrete enough to be used as "umbrella terms" for navigating to other, deeper concepts, while feeling intuitive to the user.

However, when reading up on design guidelines for either of the systems, I realised that Android's instructions seem outdated. Backed by plenty of discussions online [72] as well as my own user experience with the native applications, the typical hamburger menu (three stacked horizontal lines) has gone out of style.

One of the reasons is the increasing size of handheld devices. A hamburger menu (or *navigation drawer*) located in the top left corner is nearly inaccessible for the majority of users, who are right-handed and use their thumb to operate the device [73], as showed by fig. 6.1.

Another reason is the fact that the hamburger menu decreases discoverability of features, as they are hidden by default [72].

Lastly, using the hamburger menu often tempts the designer to just put

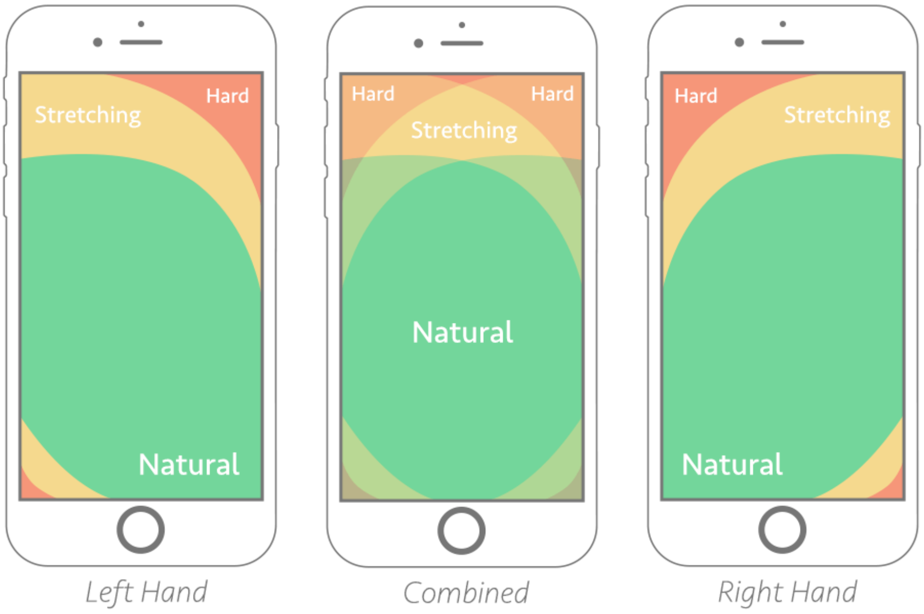


Figure 6.1: Thumb-zone mapping for left- and right-handed users [16].

everything there, without giving too much consideration to the use of the features.

One of the most popular alternatives is the *tab bar* – or *bottom navigation* in Android documentation. This type of navigation is mainly implemented by Apple's designers, but is fast spreading to Android-powered devices. It's a bar on the bottom of the screen containing three to five labelled icons representing *tabs* where each leads to a different location in the app. It enforces the idea that a user should always know where in the application they are, what they can do there and where they can go because it's always on the screen (except when the keyboard is being used). Another one of its advantages is the possibility of inobtrusive notifications using icon badges – little red circles in the top right corner of the icon, often containing a number to show the number of notifications. Other navigation options include dropdown menus, scrollable menus, single page dot navigation, and others [74].

While designing the application, I followed the material design guidelines.

When describing the screens I will refer to the user stories which the discussed UI component covers.

I chose the tab bar with five tabs: `Plans`, `Routes`, `Map`, `Friends`, and `Profile`. However, when installing the APP, the user needs to set the APP up.

6.1 Set up screens

The welcome screen contains all the options for signing up, with a button to switch to the sign-in screen if the user already has an account (US 5.5.3). Signing up with e-mail creates a new account for the user based on their e-mail (fig.6.2), while using a social network account takes the user to a sign-in screen of the specific social network. Similarly, the user can sign in to their existing account (fig.6.3).

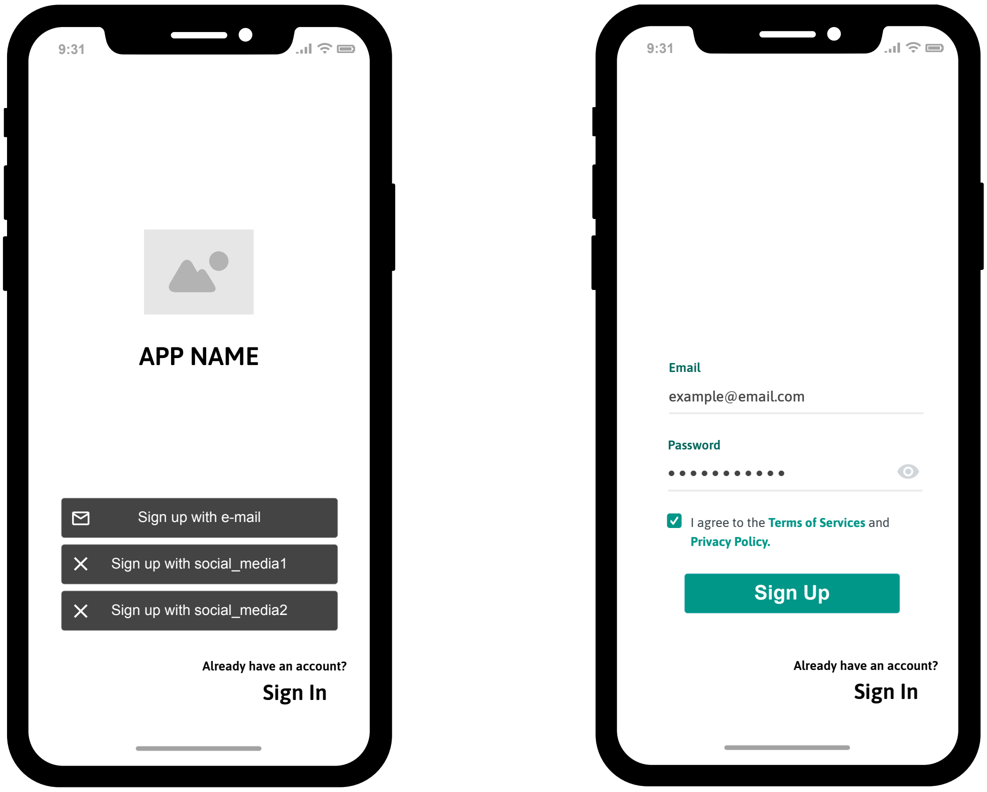


Figure 6.2: Welcome screen with sign-up options and e-mail sign-up screen.

After signing up and creating a new account, there is a sequence of screens asking for the user's personal data, such as height, weight, biological sex, birth date, and whether they know about a condition that means their heart rate should be limited, and if so, how much (on a scale of 1 to 5) (US 5.3.2, 5.3.1). Another screen asks the user to pair a watch with the APP, showing a list of Bluetooth devices paired with the phone (US 5.1.1). The last one asks the user to take a fitness test, so that their fitness can be assessed (epic 5.2). This step can be skipped, but there's a warning about the inaccuracy of methods based on readily available data compared to taking a fitness test. At the same time, the user is assured that the test can be taken later.

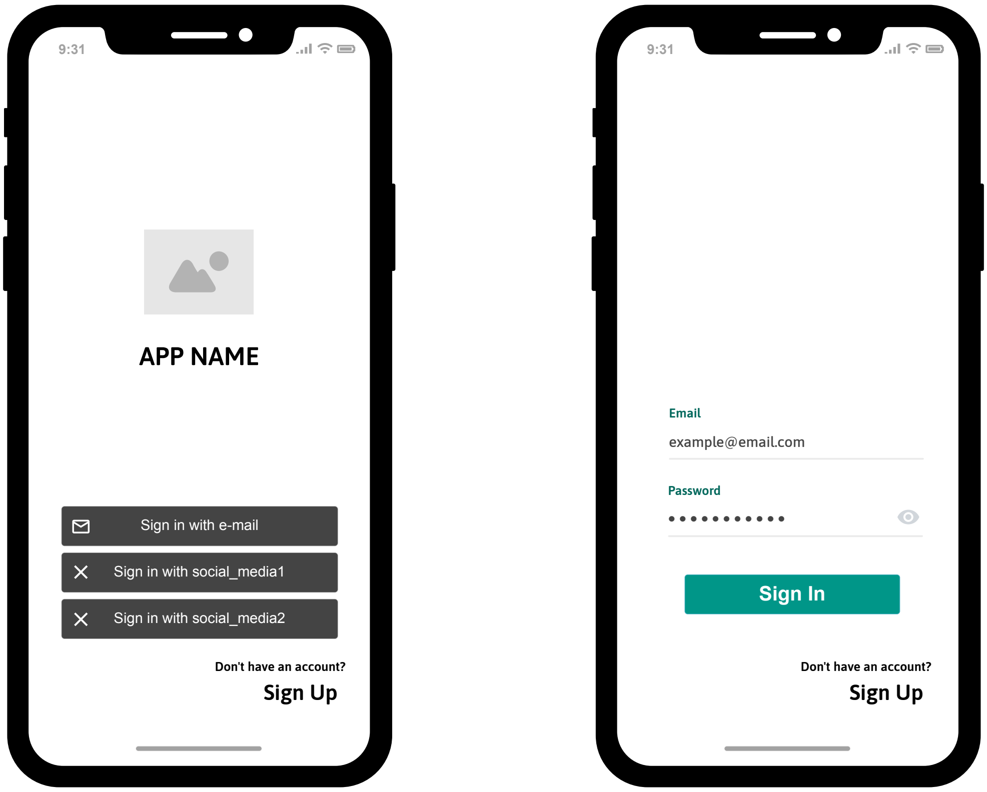


Figure 6.3: Welcome screen with sign-in options and e-mail sign-in screen.

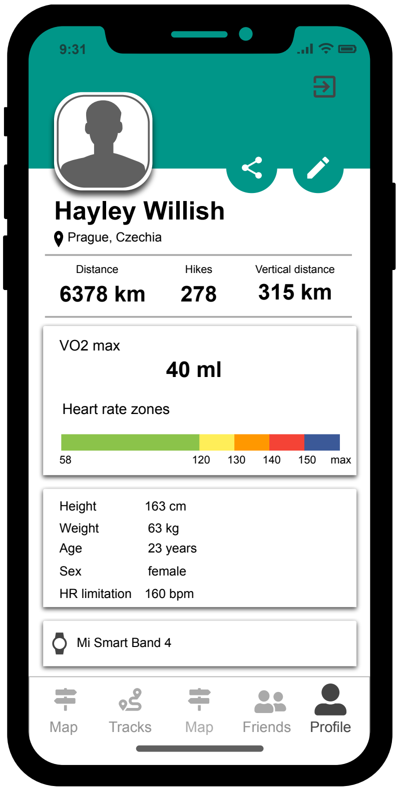
If the user decides to take the test, they are rerouted to the screen with fitness tests (without the bottom navigation bar). Again, it is possible to quit this screen.

6.2 Profile tab

After setup, the APP opens on the `Profile` tab (fig.6.4), displaying all entered and calculated information on cards, as well as the option to log out, and the currently active device (US 5.5.1, 5.5.3, 5.1.2).

The card with fitness opens to show the user's fitness data – VO2 max, heart rate zones, and minimum and maximum heart rate. The heart rate zones are color-coded; these colors occur in the whole application to signify difficulty. There is an icon to get more information about the metrics (excerpts from the Fitness Assessment chapter), and the option to take a fitness test (fig.6.5). The pictures are placeholders for charts of development of the metric over time.

On the `Fitness tests` screen (fig.6.6), the user can choose between multiple tests, each of which has a tag saying what it tests (epic 5.2). As an



Hayley Willish

Prague, Czechia

Distance	Hikes	Vertical distance
6378 km	278	315 km

VO2 max

40 ml

Heart rate zones

Height	163 cm
Weight	63 kg
Age	23 years
Sex	female
HR limitation	160 bpm

Mi Smart Band 4

Figure 6.4: The Profile screen

example, the `6 Minute Walking Test` screen has directions on what the user should do, and after the test, there is a screen with results (fig.6.7). Again, the pictures will contain the charts of the metric's values over the course of the test.

When we come back to the `Profile` screen (fig.6.4), we can also manually edit all the information the user has provided. The profile picture, basic data like height etc., as well as the fitness information (US 5.3, 5.5.2).

Another option from here is to see the devices by expanding the card with the active device. The `Devices` screen (fig.6.8) contains the active device which can be deactivated (US 5.1.2, 5.1.1), as well as other paired devices, which can be either activated or unpaired. The user can also pair a new device with the app (opening a list of devices paired with the phone). Tapping the `Activate` button will deactivate the currently active device.

6.3 Map tab

The default tab on which the application opens is the `Map` tab (fig.6.9). This is where the user can look at the integrated map (US 5.4.1), as well as search

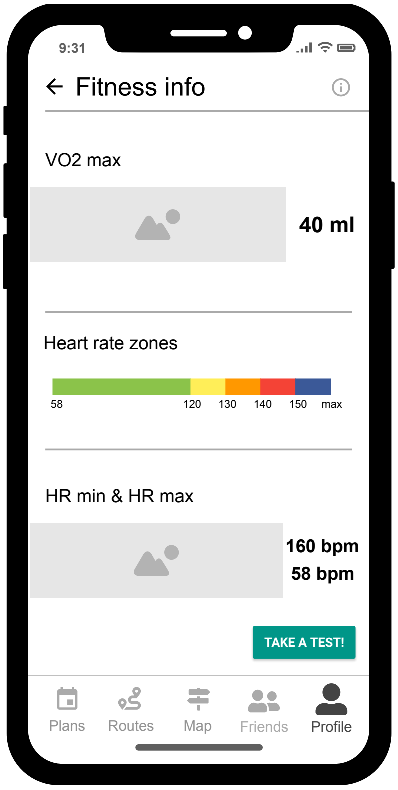


Figure 6.5: Fitness information

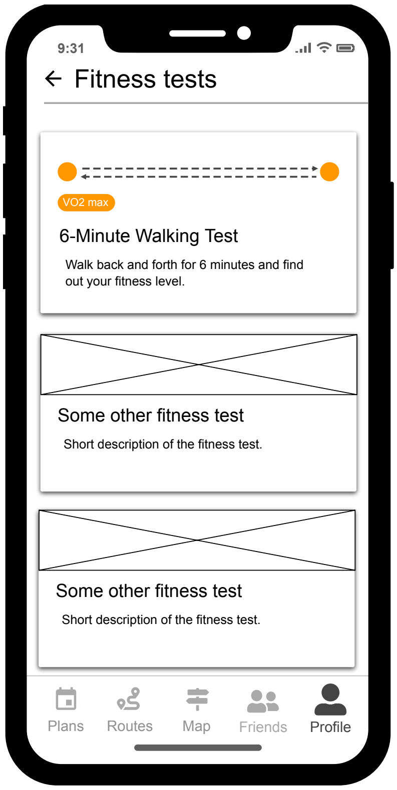
for specific places (fig.6.10) and add them as waypoints to the planned route (US 5.4.2). The waypoint will be added to the route as the next waypoint.

Planning a new route can also be started with the floating button on the bottom right of the screen (fig.6.9).

The planning process consists of three steps:

1. waypoints,
2. pick a route, and
3. plan a hike.

In the `Waypoints` step fig.6.11), the user picks where they want to start their route, the destination, and all the points through which they want to pass. Each of the points can be picked directly from the map with a long press, or by searching for a place (fig.6.10). The current location can also be used as a starting point. The intermediate points can be added using the + button, and all the waypoints can be rearranged by drag'n'dropping each line by the menu icon on the left. The user can also remove individual waypoints using



Fitness tests

VO2 max

6-Minute Walking Test

Walk back and forth for 6 minutes and find out your fitness level.

Some other fitness test

Short description of the fitness test.

Some other fitness test

Short description of the fitness test.

Figure 6.6: Fitness tests

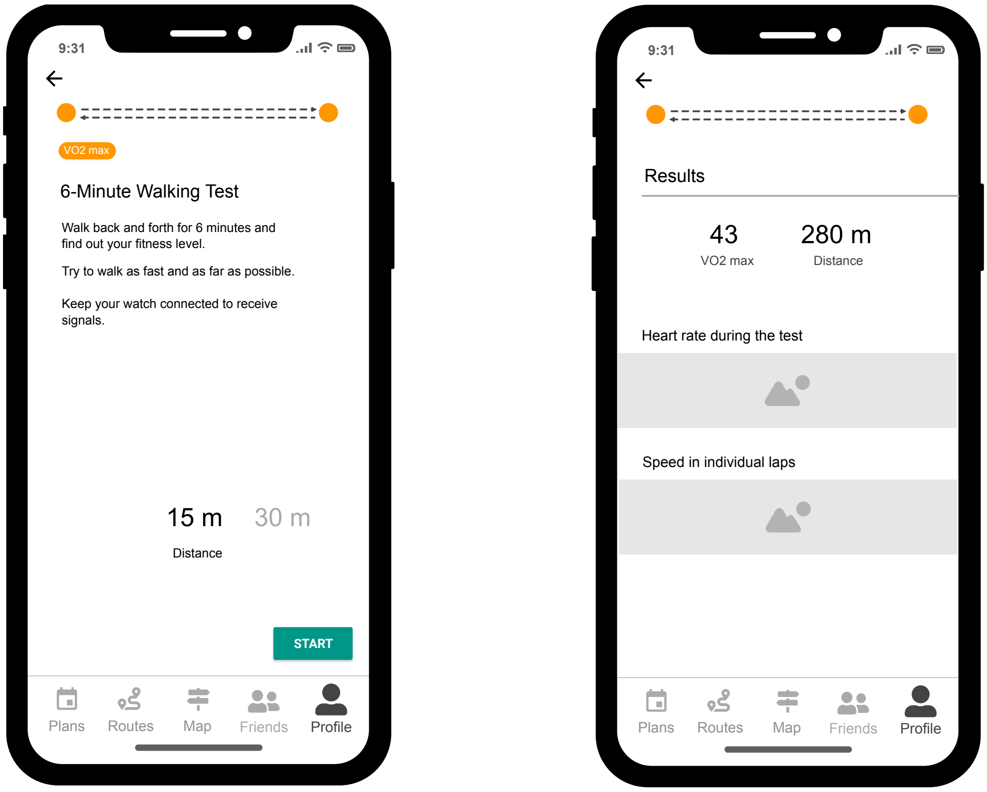
the X icon on the right: when removing the starting or destination point, only the text is removed, since every route needs two points.

Another option is to choose a return trip, where the platform should offer routes that start in the starting point, go through the destination, and either give a different route to go back if possible, or has the user retrace their steps.

When the user is satisfied with the waypoints, they tap the `Find a route` button. In this step the planning UI pulls up higher, covering a larger part of the map, while the visible part of the map contains all the route options on an appropriately zoomed-in map. The options are colour-coded based on their difficulty. The chosen option is highlighted, while the others are visible but translucent. The options can be switched between by tapping on the map.

The middle of the planning UI now contains the details of the chosen route – the time it will get the user to hike it if they go alone, its elevation, length, prevalent surface type, and a difficulty chart (US 5.4.6).

The bottom part of the planning UI has information on individual segments of the route. The picture is a placeholder for the route's elevation profile, which has two interactive sliders to pick the segment whose details the user wants to know. These details include its elevation, surface, and a



6-Minute Walking Test

Walk back and forth for 6 minutes and find out your fitness level.

Try to walk as fast and as far as possible.

Keep your watch connected to receive signals.

Figure 6.7: 6 Minute Walking Test

difficulty chart.

The route can be saved and unsaved by tapping the bookmark icon, and shared via link by tapping the share icon.

If the user wants to just start the hike without planning for a different day or inviting friends, they can press the `GO` button and the APP navigates them, or they can continue to the next step: Plan a hike.

Planning a hike means choosing the date with a date picker (US 5.4.9), and the people to join the user (fig.6.13; US 5.6.2). These can be searched for among the user's friends, using the `Invite people` button (fig.6.14). While searching for friends, their avatars are added to the summary on the bottom and the counter is incremented.

Once the friends are chosen, the user can see a summary of the hike, with the difficulty chart adjusted to the group's common fitness (fig.6.15; US 5.6.1), the number of people who are coming or invited, and how long it is going to take them.

This screen also contains a checkbox to allow friends join the hike – whether the hike should be marked as joinable when it is shared to the friends' feed of `All plans` (fig.6.16).

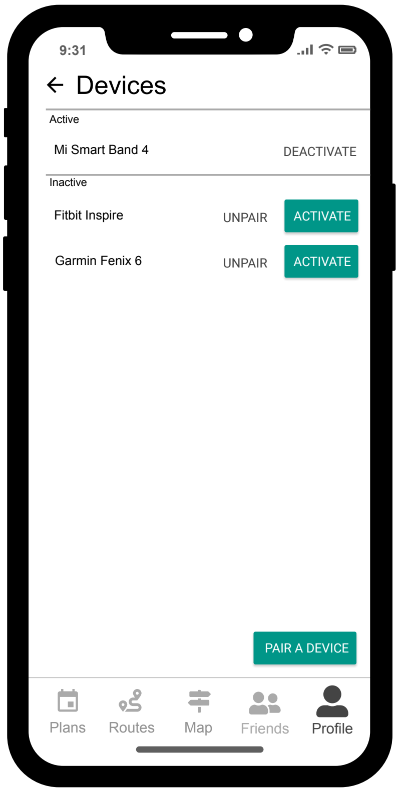


Figure 6.8: Devices

Once the planning is finished, we return to the basic `Map` screen, with a snack bar notification telling the user where they can find the planned hike and how many people were invited (fig.6.15).

The whole planning can be stopped at any point with the `Cancel` button, which will trigger a confirmation dialogue, as the user is about to delete all their progress, and after confirming, we return to the `Map` tab.

6.4 Plans tab

The `Plans` tab is divided into three sections – `All plans`, `My plans` and `History`. The default `All plans` section (fig.6.16) is a filterable feed of the users' and their friends' upcoming plans (US 5.6.10). This is also where the user's invitations to friends' hikes are displayed; these can be either dismissed or accepted (US 5.6.4, 5.6.3).

A collapsed hike card contains basic information, the route on a map (represented by the picture placeholder) such as who hosts it, when it is happening and how long it will take, whether it is joinable and how difficult it would be for the user.

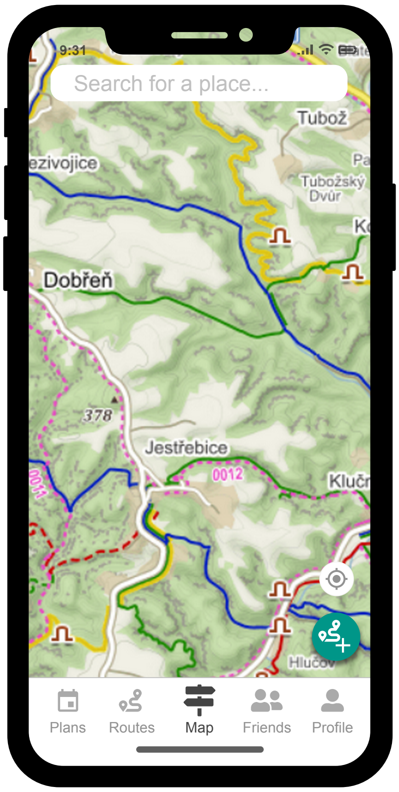


Figure 6.9: Map screen

When tapping on the card, the card expands to show more of the hike's details: who is coming, how long it is currently planned to be plus how much longer it will take if the user joins, and all of its information recalculated to the user's fitness.

If the hike is joinable, there is an `Ask to join` button (US 5.6.10); if not, the user can only go back.

Again, there are the options to save (US 5.4.4) or share the route (US 5.4.5).

The `My plans` section (fig.6.17) contains the hikes that the user has planned or joined. The detail of a hike planned by the user is different, compared to looking at hikes planned by other people, since there have to be options to administer the hike. Requests to join are shown in this detail; by tapping at the `People` section the user can manage the people connected to the hike, as well as allow or disallow people to join the hike (US 5.6.10). The route can be cancelled by the red trash icon, and the user can also let others know that they cannot go (US 5.6.11). If the user is the host of the hike, they have to appoint a new host from the people who have accepted the invitation.

The `Start` button will become active on the day of the hike. When it is pressed, the route will be activated on the phones of all the attendants,

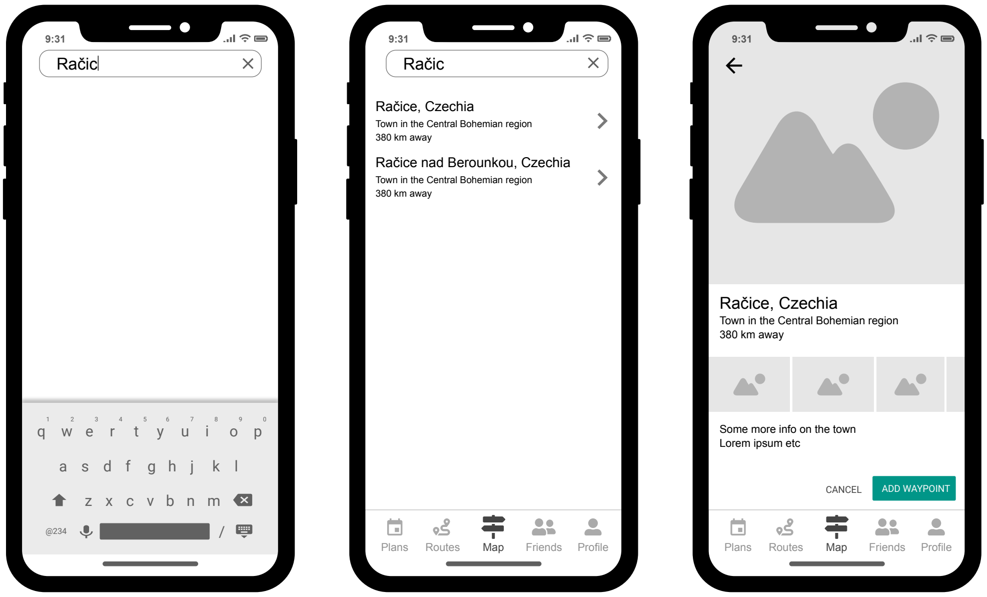


Figure 6.10: Search for a place

allowing to switch to navigation mode (US 5.6.12).

Tapping on the `Route` section will show the route's details, just like in fig.6.12.

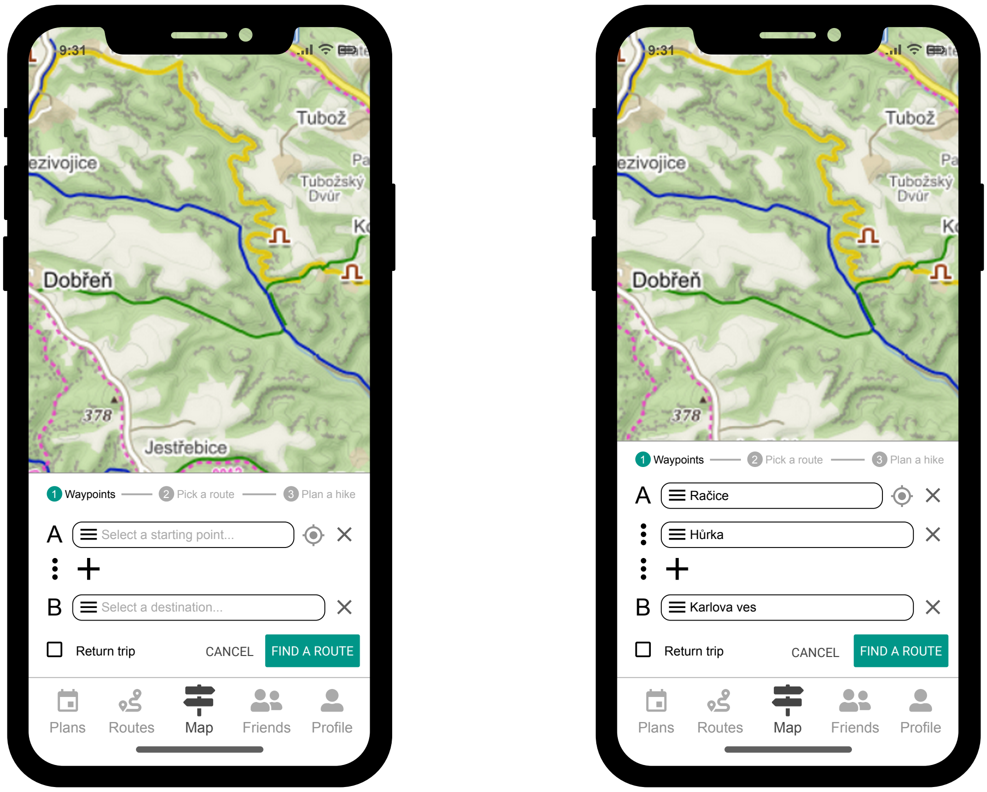
The `History` section (fig.6.18) of the `Plans` tab is the same as the `My plans` section, except all the hikes are in the past (US 5.4.3). Each hikes' map contains the planned route, and the real route in case the user did not precisely follow the plan. This hike can be repeated by pressing the `Plan a hike` button and choosing whether to use the originally planned or the actually taken route. Then the user is again taken to the planning UI.

6.5 Routes tab

The `Routes` tab (fig.6.19) contains a filterable list of saved routes (US 5.4.4). The only difference between a hike card and a route card is the fact that routes are not planned, so they do not include a host, people who are going, or the date they are planned for.

6.6 Friends tab

Finally, the last tab is the `Friends` tab. It contains a view of all the user's friends, as well as any pending friendship requests (US 5.6.7, 5.6.8). The detail of a user's profile is similar to the user who is signed in, but contains only basic information. The profile is shareable (US 5.6.9), and the friend



A Select a starting point...
B Select a destination...

A Račice
Hůrka
B Karlova ves

Figure 6.11: Planning a route with waypoints

can also be removed from the user's friends list (US 5.6.6). This is confirmed by a confirmation dialogue, warning the user about the fact that they will be removed from one another's plans.

By pressing the add button, the user is navigated to the Add friends screen (fig.6.20) can also add new friends using a link or social media (US 5.6.5).

6.7 Navigation mode

The application allows a user to navigate a planned route using a map and written directions (fig.6.21; US 5.4.8). Once a hike is started, the view is switched to the `Map` tab. The route is visualized on the map in colours of respective hiking trails and the user's position is highlighted. There's also an overview of the difficulty of the current part of the route, and the user is notified if they are lagging behind the plan. If the user wants to see something else than the navigation mode, they can use the `Close` button on the top right, or just switch between tabs. The navigation mode stays alive and accessible through a permanent notification.

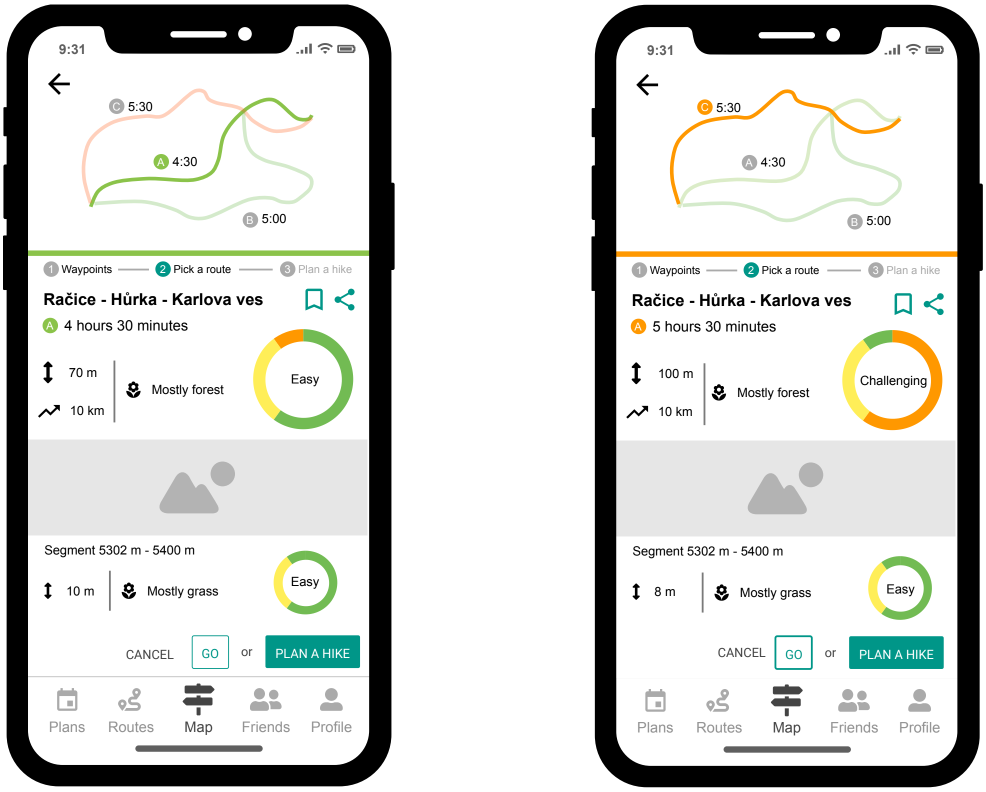
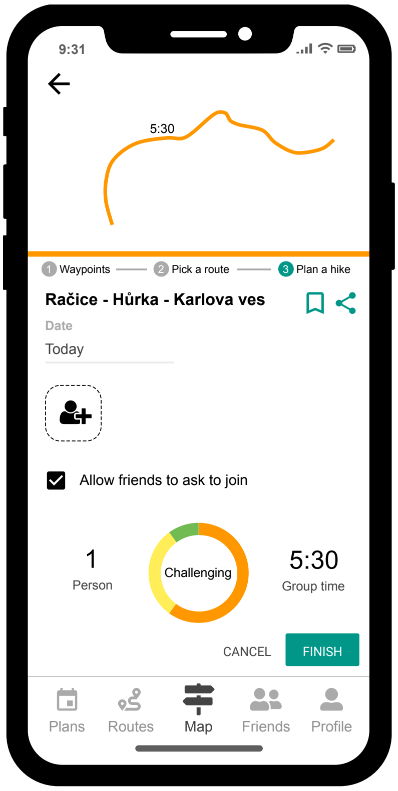


Figure 6.12: Picking a route based on its details.

6.8 Notifications

Notifications are communicated to the user via icon badges on the bottom tab bar. When not using the app, the user receives notifications through the phone's notifications bar.



Račice - Hůrka - Karlova ves

Date
Today

☑ Allow friends to ask to join

Figure 6.13: The initial state of a planned hike.

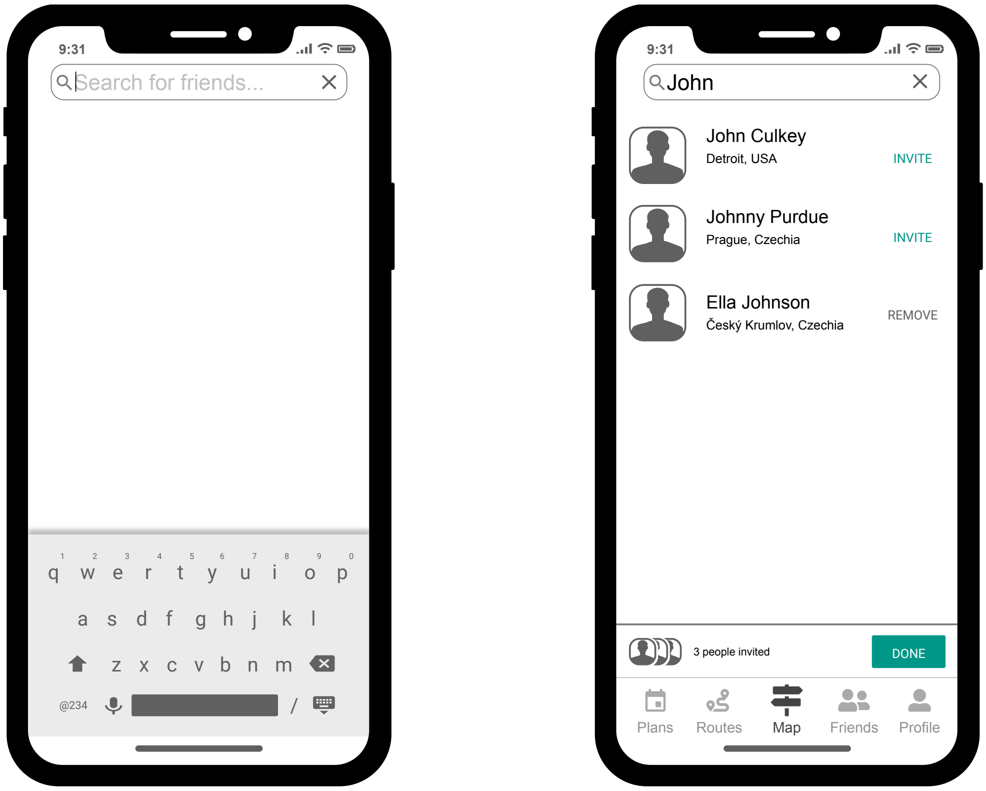
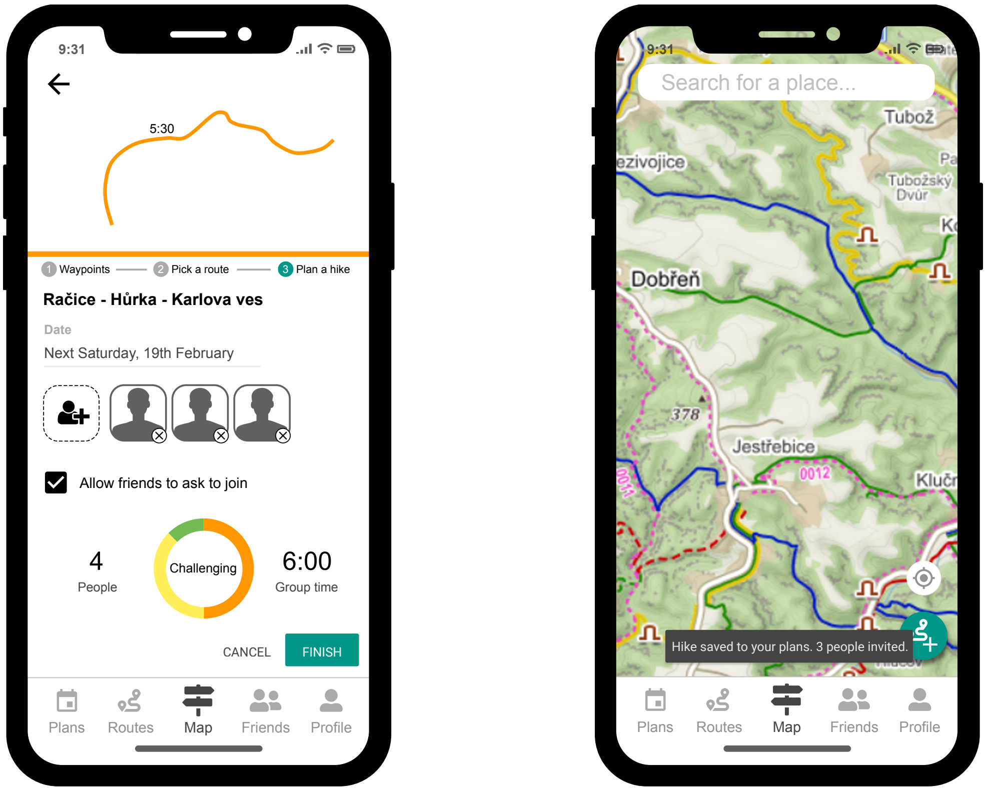


Figure 6.14: Searching for friends to invite to a hike. A summary of already chosen people is on the bottom.



Račice - Hůrka - Karlova ves

Date

Next Saturday, 19th February

☑ Allow friends to ask to join

4 People

Challenging

6:00 Group time

Hike saved to your plans. 3 people invited.

Figure 6.15: Overview of the hike. After pressing the `Finish` button, we are back to the `Map` screen.

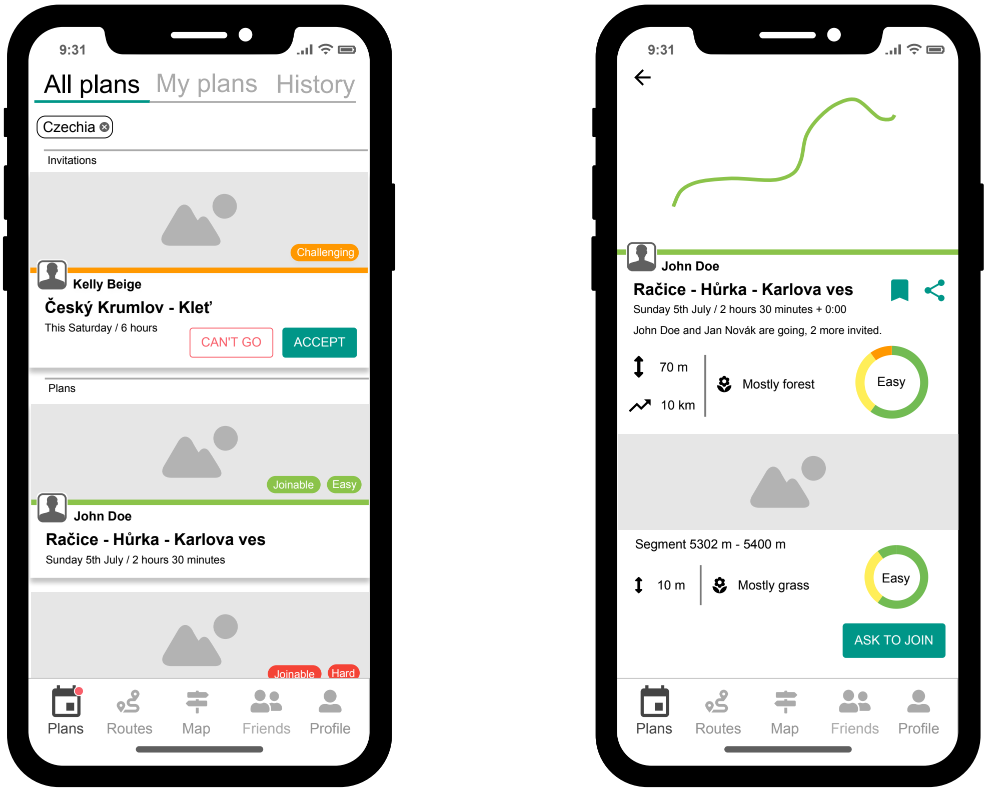


Figure 6.16: All plans feed. Detail of a joinable hike.

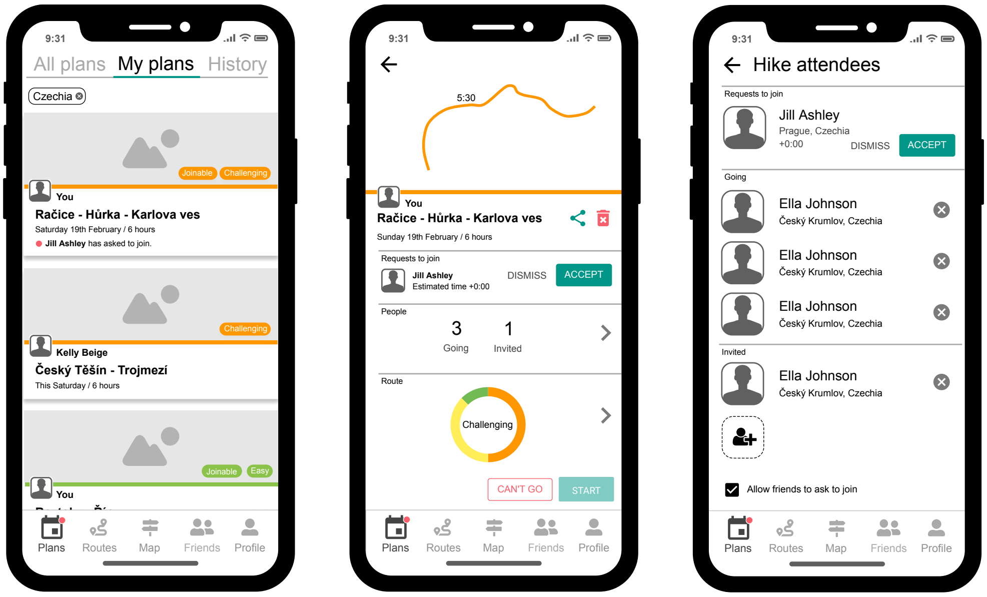
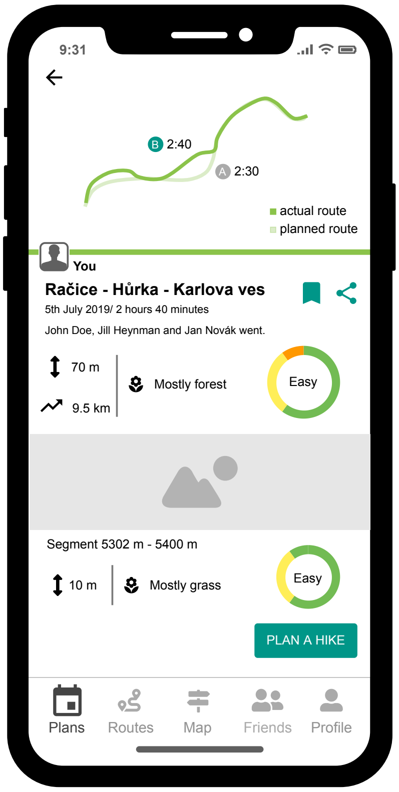


Figure 6.17: My plans. Detail of a hike planned by the user, and a view of the invited and going people.



You

Račice - Hůrka - Karlova ves

5th July 2019/ 2 hours 40 minutes

John Doe, Jill Heynman and Jan Novák went.

70 m

9.5 km

Mostly forest

Easy

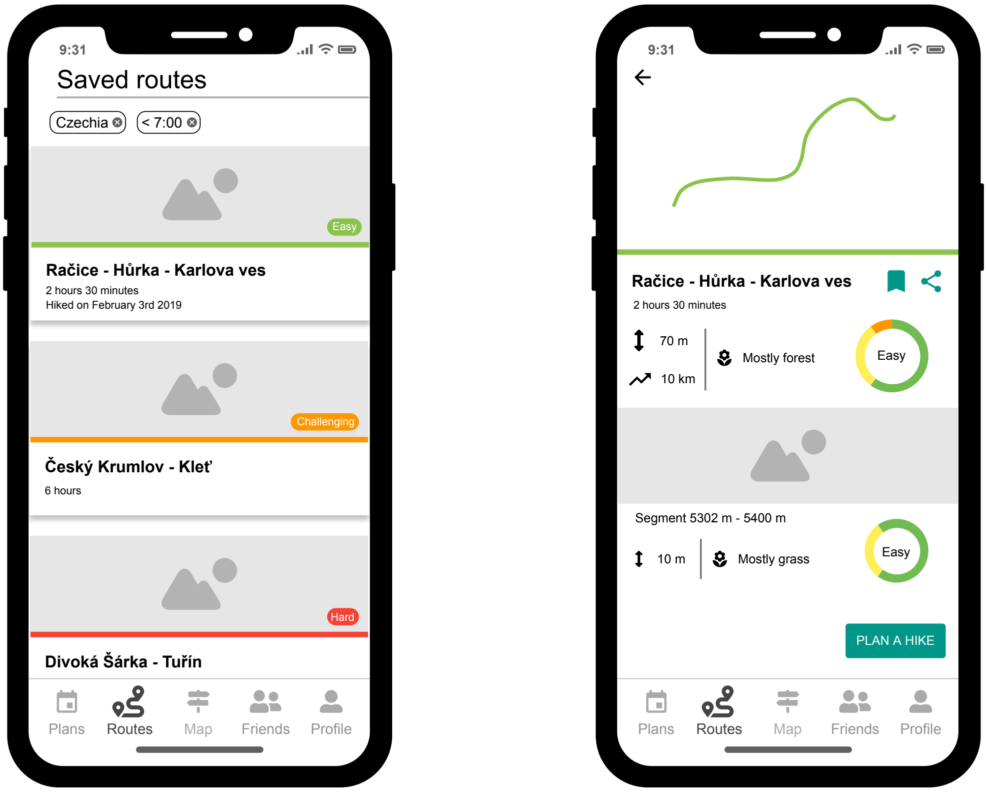
Segment 5302 m - 5400 m

10 m

Mostly grass

Easy

Figure 6.18: Detail of a historic hike.



9:31

Saved routes

Czechia ⊗ < 7:00 ⊗

Easy

Račice - Hůrka - Karlova ves

2 hours 30 minutes

Hiked on February 3rd 2019

Challenging

Český Krumlov - Kleť

6 hours

Hard

Divoká Šárka - Tuřín

9:31

Račice - Hůrka - Karlova ves

2 hours 30 minutes

70 m

10 km

Mostly forest

Easy

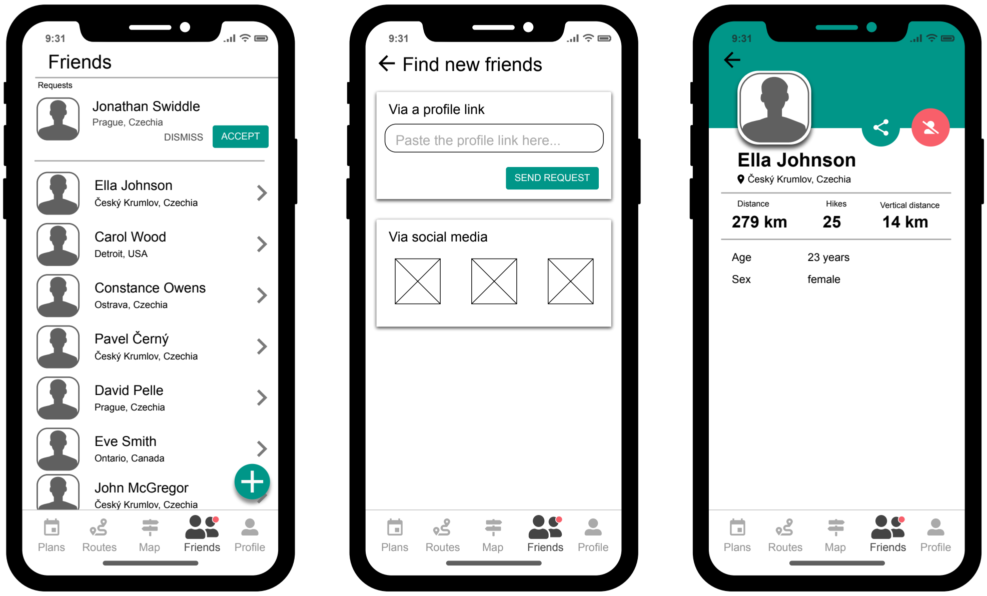
Segment 5302 m - 5400 m

10 m

Mostly grass

Easy

Figure 6.19: Routes tab. Detail of a route.



9:31

Friends

Requests

Jonathan Swiddle

Prague, Czechia

DISMISS ACCEPT

Ella Johnson

Český Krumlov, Czechia

Carol Wood

Detroit, USA

Constance Owens

Ostrava, Czechia

Pavel Černý

Český Krumlov, Czechia

David Pelle

Prague, Czechia

Eve Smith

Ontario, Canada

John McGregor

Český Krumlov, Czechia

9:31

Find new friends

Via a profile link

Paste the profile link here...

Via social media

9:31

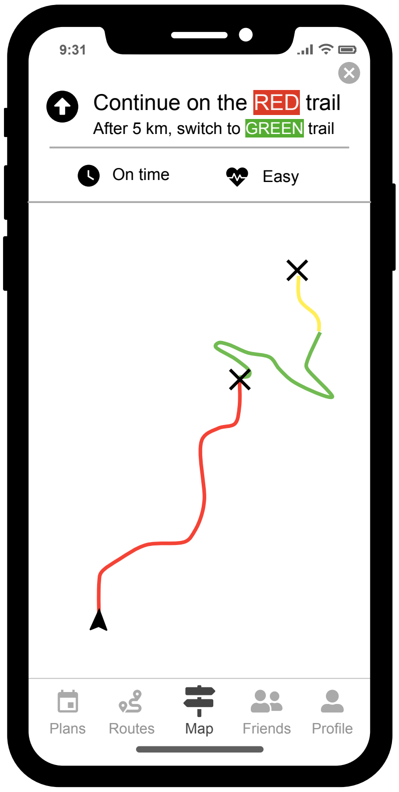
Ella Johnson

Český Krumlov, Czechia

Distance	Hikes	Vertical distance
279 km	25	14 km

Age	23 years
Sex	female

Figure 6.20: Friends tab, finding new friends and the user profile of a friend.



9:31

Continue on the RED trail

After 5 km, switch to GREEN trail

On time

Easy

Figure 6.21: Navigation mode.

Chapter **7**

Architecture

In this chapter I will describe the high-level architecture of the IoT platform, based on the component diagram.

Internet of Things, as a system, requires outlying units "on site", which measure the data and send it to the processing centers. These units are in my case both the smartwatches and mobile phones.

The smartwatch requires an application to propagate the data from the heart rate sensor to the mobile application through the `hr_data` interface. Once the raw data is received, it is combined with GPS data from the phone and forwarded through the `data_access` interface to the IoT platform, where it gets processed and saved. When the smartphone application requires it, it requests the processed data through the same interface and displays it.

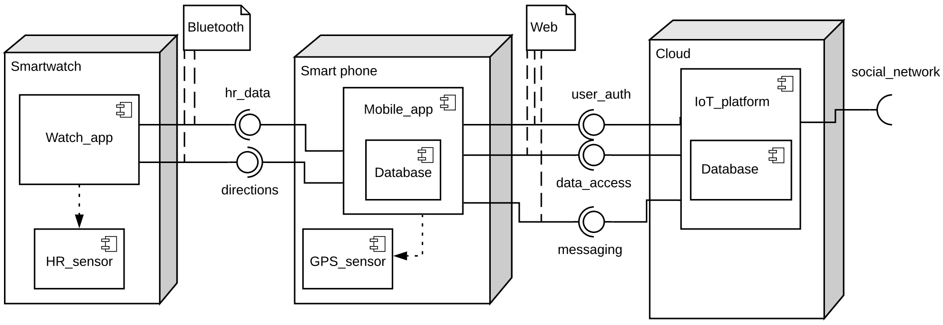


Figure 7.1: Component diagram of the designed IoT solution.

Depending on the number and complexity of the features in the future, it is also possible to introduce a web client in order to make the smartphone APP lighter and easier to navigate.

The watch communicates with the phone application over Bluetooth, with the phone application acting as a server, waiting for the watch application to connect and send its data. The phone application connects to the server using the phone's Internet connection.

7.1 Smartwatch application

Since the vendors of smartwatches generally use proprietary operating systems, every integrated smartwatch type needs an application of its own. This is distributed together with the phone application.

Generally, it should be as light-weight as possible, to spare the limited battery resources. Without any need for persistence, the application layer should collect the data from the heart rate sensors and regularly send them to the phone in batches. The presentation layer displays data from the phone (such as instructions for a fitness test or navigation directions) received through the `directions` interface. This is why it requires a bi-directional channel for Bluetooth communication.

7.2 Smartphone application

The phone application needs to be able to communicate with the watch via Bluetooth using the `hr_data` and `directions` interfaces, as well as with the APIs for data access, notification and user authentication offered by the server.

The user is authenticated via `user_auth`.

The `Database` also contains the user's plans, so that the user does not have to be online when in navigation mode.

When navigating, the `Mobile_app` collects data from the available `GPS_sensor` and from the `Watch_app`, and tries to send it to the server via the `data_access` interface if there is an active Internet connection. Otherwise this is stored in the local `Database` and sent when the phone next connects.

The `Mobile_app` is also subscribed to a messaging service such as Firebase [75] provided by the server to handle notifications.

7.3 Server application

The `IoT_platform` application communicates with the `Mobile_app` via the aforementioned interfaces, and also requires a public interface from every supported social network, so that the users can log in using their accounts from those networks and add new friends that they already know.

The server application has a simple front-end client for administration purposes.

The `IoT_platform` is hosted in a cloud, since this infrastructure provides room for scalability and performance [76], which should work perfectly in combination with my application's need for storing of detailed route definitions and computing tailored route difficulty estimates.

The offered `user_auth` interface is used for authentication. The user's data should be as secure as possible given its very personal nature, while maintaining a comfortable level of usability for the user; this is why authentication should be done with OAuth, as opposed to HTTP Basic Auth, combined with a One Time Password for logins from unrecognized devices.

Chapter 8

PoC implementation

In this chapter I describe the process of implementing the PoC.

A proof-of-concept is a quick and dirty solution that just works [77]. The source code does not need to be pretty, or designed well, the main point is the functionality.

8.1 Android application

I started by implementing a simple Android application that acts as a server for incoming Bluetooth connections from paired devices. This was fairly straightforward since Android's documentation is up to date and there is a large community of developers for this OS.

My application is written in Java.

8.2 Smartwatch application

The next step was to choose and integrate a smartwatch.

Initially I meant to use the Xiaomi Mi Band 4 since I already owned one. However, Xiaomi watches do not pair directly with the phone; they use a proprietary application and server-based pairing. There is no way to access the data from the phone other than to reverse engineer the protocol, root the phone, and access the pairing keys in the app's database [78].

I looked at multiple vendors' popularity based on market share and affordability, their documentation to see how good their development support is and whether or not they have public APIs for development.

According to Statista.com (fig. 8.1), the most popular are Apple watches, followed by Samsung, Fitbit and other vendors.

While Apple has a crushing lead on all other vendors, buying a unit just for development is not an option for me because of their high prices. This is why I chose the Samsung Galaxy Watch Active.

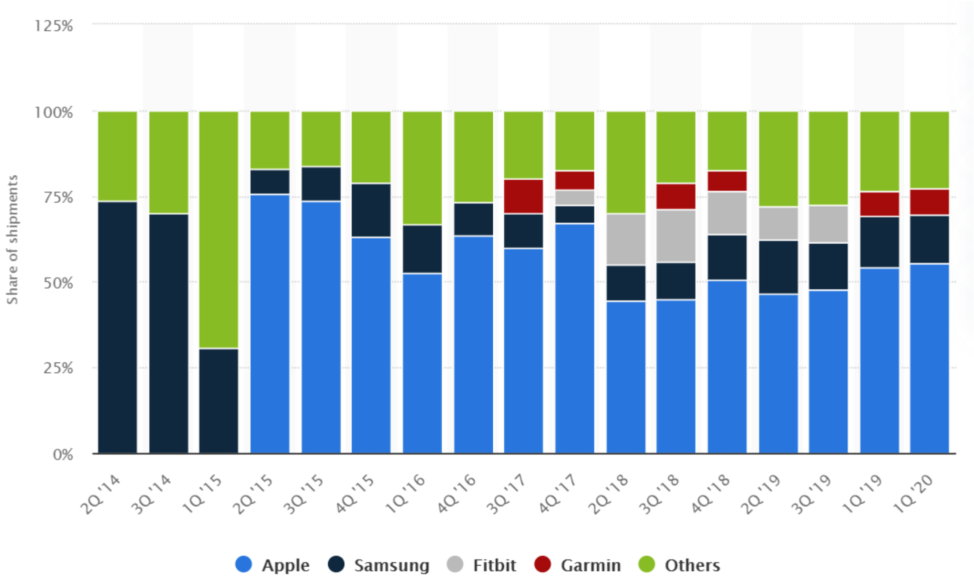


Figure 8.1: Market share of smartwatch unit shipments worldwide from summer 2014 to spring 2020, by vendor [17].

Samsung watches run Tizen OS, which has public APIs, extensive documentation, tutorials and a large community, with native applications written in the C language. This is why I was surprised to find out how difficult it is to develop for this watch. The watch needs to be connected over the local network to the Tizen Studio IDE; the tutorial for this [79] omits the important step of enabling developer options when working with Bluetooth, so that the device does not disconnect when trying to use the Bluetooth API. The tutorial also doesn't mention that Bluetooth should not only be disabled, but has to be turned off, on, and off again in order to connect.

Working with Bluetooth itself was tricky as well since their guide [80] explicitly says that it should be implemented in the main thread, but the app refused to connect to the phone when it was implemented that way. When I finally found a working sample application [81], its Bluetooth functionality was divided into different threads.

When implementing the watch's communication with the Android phone, another surprise was the tutorial's missing step of asking for permissions to access the heart rate sensor [82].

When my application could finally both measure heart rate and send string data to the Android phone, I attempted to send the heart rate data, which the application successfully performed and then crashed, without an accessible crash log, stack trace, or any log that could shed light on the issue at hand. After spending a large amount of time on trying to find a solution to this problem without any results, even trying to run my app on a different watch (Samsung Galaxy Watch Active 2), I figured out that the problem was bigger than I initially expected and decided to leave the implementation unfinished.

8.3 Recommendations for future development

When it comes to wearables, I recommend limiting the supported devices to those that do not use server-based pairing, as this would cover most of the currently available gadgets, or take advantage of (and contribute to) existing open-source software for wearables integration, such as Gadgetbridge [83].

In addition to these restrictions, continuous heart rate monitoring causes an extreme drain of the watch's battery, which is fine for short activities (sprints, swimming, cardio workouts), but can be a considerable issue while hiking. This is why I recommend supporting mostly devices which have a large capacity in this regard.

Conclusion

The aim of this thesis was to analyse and design the functionality of a PoC IoT solution for assessment of tourist track difficulty, based on the individual's fitness.

To support my research with knowledge in the fitness domain, I studied the modern methods of cardiovascular fitness assessment, judged them based on their feasibility, accuracy and relevance to my platform. This gave me a good insight into measurable attributes of fitness and what is required of my application for it to be user-friendly when it came to fitness testing while remaining accurate.

Thanks to this research I could evaluate the suitability of fitness estimation models used by existing applications in the domain of fitness and e-health. I also studied their use of IoT principles and opportunities, their community features, user-friendliness of their mobile applications, the supported platforms and propriety. With this inspiration, I saw which features are crucial for the applications in the domain, and how I can make mine stand out. One of the most important such features is the option to use similar users' data to predict the exertion of a specific user.

With this in mind, I took to figuring out the functionality that the platform ought to support, using the approach of agile methodology. The result of this effort is a chapter of epics and user stories, ready to be handed over to an agile development team. A number of features was suggested for future consideration.

All these requirements were manifested in the front-end design for an Android mobile application, which is also ready to be implemented and tested with users, and greatly exceeded the original aims of this thesis.

Another step was designing a high-level architecture for the IoT platform, and finally, the development of a PoC. Development efforts were hindered by technical issues of the chosen smartwatch's operation system, misguiding documentation (or lack thereof) and absence of examples for this particular problem.

Due to these issues I was not able to collect the data which I needed to demonstrate the PoC.

More research needs to be done, in coordination with data scientists and physical therapy professionals, in order to develop reliable models for fast difficulty assessment of a route for a specific individual.

Bibliography

[1] Training 4 Endurance. Vo2 max running. no date, [Online; cited 27th January 2020]. Available from: https://training4endurance.co.uk/wp-content/uploads/2013/07/V02max-running.png

[2] komoot. No Internet? No Problem. [Online; cited 28th January 2020]. Available from: https://www.komoot.com/features

[3] komoot. Know Exactly What To Expect. no date, [Online; cited 28th January 2020]. Available from: https://www.komoot.com/features

[4] komoot. Route elevation detail. 2020, [Screenshot by author]. Available from: https://www.komoot.com/

[5] komoot. Route surface overview. 2020, [Screenshot by author]. Available from: https://www.komoot.com/

[6] endomondo. Screenshot of the endomondo main screen. [Screenshot by author].

[7] endomondo. Screenshot of the endomondo More tab. [Screenshot by author].

[8] endomondo. Endomondo to get G+ integration? [Online; cited 28th January 2020, screenshot by Ryan Moore]. Available from: https://www.androidcoliseum.com/2013/07/endomondo-to-get-g-integration.html

[9] endomondo. History Page Example. [Online; cited 28th January 2020]. Available from: https://support.endomondo.com/hc/en-us/articles/202228448--Tagging-Statistics

[10] Strava. Strava Challenges screenshot. [Screenshot by author].

[11] Strava. Strava Segments screenshot. [Screenshot by author].

[12] Strava. Viewing Activities. [Online; cited 28th January 2020]. Available from: https://support.strava.com/hc/en-us/articles/216919157-Viewing-Activities

[13] Garmin. My day screen. no date, [Online; cited 27th January 2020]. Available from: https://connect.garmin.com/static/screen-phone-myday-717fec02a4d8eea2690241774363b357.png

[14] Garmin. Screenshot of Garmin Explore Map screen. [Screenshot by author].

[15] Garmin. Screenshot of Garmin Explore route creation screen. [Screenshot by author].

[16] Ingram, S. Thumb zone mapping. 2016, [Online; cited 1st May 2020]. Available from: https://cloud.netlifyusercontent.com/assets/344dbf88-fdf9-42bb-adb4-46f01eedd629/496f7bc0-4c6c-4159-b731-ec3adcf91105/thumb-zone-mapping-opt.png

[17] Statista. Market share of smartwatch unit shipments worldwide from the 2Q'14 to 1Q '20*, by vendor. 2020, [Online; cited 10th May 2020]. Available from: https://www.statista.com/statistics/524830/global-smartwatch-vendors-market-share/

[18] Apple. Proactive Health Monitor - Always in sight. Always insightful. September 2019, [Online; cited 26th November 2019]. Available from: https://www.apple.com/apple-watch-series-5/health/

[19] Curry, S. J. Screening for Atrial Fibrillation With Electrocardiography. *JAMA Cardiology*, June 2018, [Cited 26th November 2019]. Available from: https://jamanetwork.com/journals/jama/fullarticle/2695678

[20] Kiana Teherani, A. M. Wearable Technology and Wearable Devices: Everything You Need to Know. [online]. *Wearable Devices Magazine*, March 2014, [Online; cited 26th November 2019]. Available from: http://www.wearabledevices.com/what-is-a-wearable-device/

[21] Ghosh, A. K. Anaerobic Threshold: Its Concept and Role in Endurance Sport. *The Malaysian journal of medical sciences*, January 2004, [Online; cited 27th January 2020]. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3438148/

[22] Firstbeat Technologies Ltd. VO2 Estimation Method Based on Heart Rate Measurement. March 2012, [Online; cited 7th February 2020]. Available from: https://www.firstbeat.com/wp-content/uploads/2015/10/white_paper_vo2_estimation.pdf

[23] Firstbeat Technologies Ltd. *Automated Fitness Level (VO2max) Estimation with Heart Rate and Speed Data (white paper)*. Firstbeat Technologies Ltd., November 2014, [Online; cited 19th January 2020]. Available from: `https://assets.firstbeat.com/firstbeat/uploads/2017/06/white_paper_VO2max_30.6.2017.pdf`

[24] Bassett Jr., D. R., Howley, E. T. Limiting factors for maximum oxygen uptake and determinants of endurance performance. *Medicine & Science in Sports & Exercise*, January 2000, doi:10.1097/00005768-200001000-00012, [Cited 23rd January 2020].

[25] Hubert, G. Evaluation des COOPER-Tests aus leistungsphysiologischer Perspektive - Kann über den COOPER-Test die aerobe Ausdauerleistungsfähigkeit abgeschätzt werden? [Evaluation of the Cooper test from a performance-pysiological perspective - Can aerobic performance and fitness be rated through the Cooper test?]. August 2006, [Online; cited 23rd January 2020]. Available from: `http://www.ub.uni-heidelberg.de/archiv/8585`

[26] Cooper, K. H. A Means of Assessing Maximal Oxygen Intake Correlation Between Field and Treadmill Testing. *JAMA Network*, January 1968, doi: https://doi.org/10.1001/jama.1968.03140030033008, [Online; cited 23rd January 2020].

[27] Wood, R. Beep Test Calculations Table. *Topend Sports Website*, 2008, [Online; cited 27th January 2020]. Available from: `https://www.topendsports.com/testing/beep-table.htm`

[28] Wood, R. Beep Test Variations and Modifications. *Topend Sports Website*, 2008, [Online; cited 27th January 2020]. Available from: `https://www.topendsports.com/testing/beep-variations.htm`

[29] Léger, L. A., Lambert, J. A maximal multistage 20-m shuttle run test to predict VO2 max. *European Journal of Applied Physiology*, 1982, doi: https://doi.org/10.1007/BF00428958, [Online; cited 27th January 2020].

[30] Tomkinson, G. R. et al. International normative 20 m shuttle run values from 1 142 026 children and youth representing 50 countries. *British journal of sports medicine*, 2017, doi:https://doi.org/10.1136/bjsports-2016-095987, [Online; cited 27th January 2020].

[31] Ubuane, P. O. et al. The historical evolution of the six-minute walk test as a measure of functional exercise capacity: a narrative review. *Journal of Xiangya Medicine*, volume 3, no. 0, 2018, [Online; cited 12th February 2020]. Available from: `http://jxym.amegroups.com/article/view/4832/5603`

[32] Mänttäri, A. et al. Six-minute walk test: a tool for predicting maximal aerobic power (VO2 max) in healthy adults. *Clinical Physiology and Functional Imaging*, May 2018, doi:https://doi.org/10.1111/cpf.12525, [Online; cited 10th February 2020].

[33] Burr, J. F. et al. The 6-Minute Walk Test as a Predictor of Objectively Measured Aerobic Fitness in Healthy Working-Aged Adults. *The Physician and Sportsmedicine*, March 2015, doi:https://doi.org/10.3810/psm.2011.05.1904, [Online; cited 12th February 2020].

[34] Nakamura K. et al. Effect of Different Seat Heights during an Incremental Sit-To-Stand Exercise Test on Peak Oxygen Uptake in Young, Healthy Women. *Journal of Sports Science & Medicine*, August 2016, [Online; cited 21st February 2020]. Available from: http://europepmc.org/article/PMC/4974853

[35] Nakamura K. et al. The Effect of Different Standing up Frequencies in Sit-to-stand Exercise on Oxygen Uptake. *Journal of Physical Therapy Science*, volume 26, 2014, [Online; cited 21st February 2020]. Available from: https://pdfs.semanticscholar.org/1986/75677b4d4b2f8b8111cd1a4a57df01615467.pdf

[36] Nakamura K. et al. Validity and Reproducibility of an Incremental Sit-To-Stand Exercise Test for Evaluating Anaerobic Threshold in Young, Healthy Individuals. *Journal of Physical Therapy Science*, December 2015, [Online; cited 21st February 2020]. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4657412/

[37] Jackson, A. S. et al. Prediction of functional aerobic capacity without exercise testing. *Medicine and Science in Sports and Exercise*, December 1990, doi:10.1249/00005768-199012000-00021, [Online; cited 12th February 2020].

[38] George, J. D. et al. Non-exercise ˙VO2max estimation for physically active college students. *Medicine and Science in Sports and Exercise*, March 1997, doi:10.1097/00005768-199703000-00019, [Online; cited 12th February 2020].

[39] Firstbeat Technologies Ltd. Real-Time Performance Condition. 2020, [Online; cited 23rd January 2020]. Available from: https://www.firstbeat.com/en/consumer-feature/real-time-performance/

[40] Garmin. What is the Performance Condition Feature on My Garmin Fitness Device? 2020, [Online; cited 23rd January 2020]. Available from: https://support.garmin.com/en-US/?faq=A28UA4k16v1qjjGuvSFgo8

[41] Elli, Firstbeat Support. private communication, January 2020.

[42] Firstbeat Technologies Ltd. What Is Heart Rate Variability (HRV) & Why Does It Matter? [Online; cited 23rd January 2020]. Available from: https://www.firstbeat.com/en/blog/what-is-heart-rate-variability-hrv/

[43] rbuder. Garmin/Firstbeat Performance Condition Metric Observations. January 2019, [Online; cited 23rd January 2020]. Available from: https://forum.trainerroad.com/t/garmin-firstbeat-performance-condition-metric-observations/10276

[44] Heikoo. Performance Condition - not very useful. 2019, [Online; cited 23rd January 2020]. Available from: https://forums.garmin.com/sports-fitness/running-multisport/f/forerunner-945/210619/performance-condition---not-very-useful

[45] Garmin. What Are Heart Rate Zones and How Are They Used with My Garmin Fitness Device? 2020, [Online; cited 19th January 2020]. Available from: https://support.garmin.com/ms-MY/?faq=s3HqdKNtWV1NYrK16eFcc7

[46] Polar.com. HEART RATE ZONES | THE BASICS. April 2016, [Online; cited 20th May 2020]. Available from: https://www.polar.com/blog/running-heart-rate-zones-basics/

[47] Mackenzie, B. Maximum Heart Rate Stress Tests. 1997, [Online; cited 20th May 2020. Available from: https://www.brianmac.co.uk/hrm2.htm

[48] ntnu.edu. Heart Rate Calculator - CERG - NTNU. 2014, [Online; cited 30th March 2020]. Available from: https://www.ntnu.edu/cerg/hrmax

[49] Gellish, R.L. et al. Longitudinal Modeling of the Relationship between Age and Maximal Heart Rate. *Medicine & Science in Sports & Exercise*, volume 39, May 2007, doi:10.1097/mss.0b013e31803349c6, [Online; cited 8th February 2020].

[50] Robert Robergs, R. L. The surprising history of the ”HRmax=220-age” equation. *International Journal of Online Engineering*, volume 5, April 2002, ISSN 1097-9751, [Online; cited 8th February 2020].

[51] Sarzynski, M. A. et al. Measured Maximal Heart Rates Compared to Commonly Used Age-Based Prediction Equations in the Heritage Family Study. *American Journal of Human Biology*, September 2013, doi: 10.1002/ajhb.22431, [Online; cited 8th February 2020].

[52] Nes, B.M. et al. Age-predicted maximal heart rate in healthy subjects: The HUNT Fitness Study. *Scandinavian Journal of Medicine & Science in Sports*, February 2012, doi:https://doi.org/10.1111/j.1600-0838.2012.01445.x, [Online; cited 27th March 2020].

[53] komoot. komoot Github page. [Online; cited 28th January 2020]. Available from: https://github.com/komoot

[54] Tim, Endomondo Support. private communication, March 2020.

[55] Ozogán, M. Endomondo – tréninkové plány. November 2015, [Online; cited on 2nd March 2020]. Available from: https://michalozogan.cz/endomondo-treninkove-plany/

[56] Duffy, J. Endomondo (for iPhone) Review. May 2015, [Online; cited 24th February 2020]. Available from: https://www.pcmag.com/reviews/endomondo-for-iphone

[57] Fair, C. tapiriik. 2013, [Online; cited on 2nd March 2020]. Available from: https://github.com/Endomondo/tapiriik

[58] Strava. [Online; cited 19th February 2020]. Available from: https://www.strava.com/

[59] Meg, Strava support personnel. Freshness & Fitness. [Online; cited 19th February 2020]. Available from: https://support.strava.com/hc/en-us/articles/216918477-Fitness-Freshness-Summit-

[60] Miller, D. So Who Exactly is Buying the New (and Expensive) Garmin Smartwatches? *Reviews.com*, March 2019, [Online; cited 19th January 2020]. Available from: https://www.reviews.com/blog/who-is-buying-the-garmin-smartwatches/

[61] Hall, C. Best Garmin watch 2020: Fenix, Forerunner and Vivo compared. *Pocket-lint*, January 2020, [Online; cited 19th January 2020]. Available from: https://www.pocket-lint.com/fitness-trackers/buyers-guides/garmin/143061-best-garmin-watch-fenix-forerunner-and-vivo-compared

[62] Garmin. Garmin Connect. 2020, [Online; cited 19th January 2020]. Available from: https://connect.garmin.com/

[63] Garmin. Open Source Software for Garmin Devices. 2019, [Online; cited 19th January 2020]. Available from: https://developer.garmin.com/open-source/overview/

[64] Garmin. garmin/connectiq-apps - A collection of Connect IQ apps and libraries. 2019, [Online; cited 19th January 2020]. Available from: https://github.com/garmin/connectiq-apps

[65] Mountain Goat Software. User Stories. [Online; cited on 2nd March 2020]. Available from: https://www.mountaingoatsoftware.com/agile/user-stories

[66] Sears, J. Yep, Google Maps Really is Charging Everyone. May 2018, [Online; cited 13th April 2020]. Available from: https://mapifypro.com/yep-google-maps-really-is-charging-everyone/

[67] OpenStreetMap. [Online; cited 13th April 2020]. Available from: https://www.openstreetmap.org/about

[68] OpenRouteService. [Online; cited 13th April 2020]. Available from: https://openrouteservice.org/

[69] Geocaching. [Online; cited 18th May 2020]. Available from: https://www.geocaching.com/play

[70] StatCounter. Mobile Operating System Market Share Worldwide. [Online; cited 20th May 2020. Available from: https://gs.statcounter.com/os-market-share/mobile/worldwide

[71] Tubik Studio. Information Architecture. Basics for Designers. May 2017, [Online; cited 20th May 2020]. Available from: https://uxplanet.org/information-architecture-basics-for-designers-b5d43df62e20

[72] Sebastian, F. Tab Bars are the new Hamburger Menus. May 2018, [Online; cited 1st May 2020]. Available from: https://uxplanet.org/tab-bars-are-the-new-hamburger-menus-9138891e98f4

[73] Ingram, S. The Thumb Zone: Designing For Mobile Users. September 2016, [Online; cited 20th May 2020]. Available from: https://www.smashingmagazine.com/2016/09/the-thumb-zone-designing-for-mobile-users/

[74] Agente Content Team. Hamburger Menu Examples and Alternatives. June 2018, [Online; cited 1st May 2020]. Available from: https://agentestudio.com/blog/hamburger-menu-examples-and-alternatives

[75] Google. Firebase Cloud Messaging. [Online; cited 3rd June 2020]. Available from: https://firebase.google.com/docs/cloud-messaging/

[76] Nowak, P. The Role of Cloud Computing in the IOT (Internet of Things). February 2018, [Online; cited 3rd June 2020]. Available from: https://www.nopio.com/blog/proof-of-concept-mvp/

[77] Nowak, P. Proof of Concept vs Minimum Viable Product. April 2018, [Online; cited 3rd June 2020]. Available from: https://www.nopio.com/blog/proof-of-concept-mvp/

[78] tojaad. July 2019, [Online; cited on 22nd March 2020]. Available from: https://codeberg.org/Freeyourgadget/Gadgetbridge/wiki/Mi-Band-4/

[79] Tizen. Testing Your App on Galaxy Watch. [Online; cited 3rd June 2020]. Available from: https://developer.samsung.com/galaxy-watch-develop/testing-your-app-on-galaxy-watch.html

[80] Tizen. Tizen Bluetooth guide. [Online; cited 3rd June 2020]. Available from: https://docs.tizen.org/application/native/guides/connectivity/bluetooth/

[81] Tizen. Tizen Bluetooth tutorial – sample application. [Online; cited 3rd June 2020]. Available from: https://docs.tizen.org/development/sample/native/Network/%28Tutorial%29_Bluetooth

[82] Tizen. Accessing Heart Rate Monitor (HRM) Sensor data for Native applications. [Online; cited 3rd June 2020]. Available from: https://developer.tizen.org/community/tip-tech/accessing-heart-rate-monitor-hrm-sensor-data-native-applications?langswitch=en

[83] Shimokawa, A., Pfeiffer, C., Gobbetti, D. et al. Gadgetbridge. [Online; cited 28th May 2020]. Available from: https://codeberg.org/Freeyourgadget/Gadgetbridge/

[84] Oxford University Press. Lexico.com. [Online; cited 6th November 2019]. Available from: https://www.lexico.com/en/definition/internet_of_things

[85] Rouse, M. internet of things (IoT) [online]. *TechTarget Network*, 2017, [Online; cited 28th November 2019]. Available from: https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT

[86] Oxford University Press. Lexico.com [online]. 2019, [Online; cited 26th November 2019]. Available from: https://www.lexico.com/en/definition/proof_of_concept

Appendix **A**

Acronyms

IoT Internet of Things – *The interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data* [84] without requiring human-to-human or human-to-computer interaction. [85]

PoC Proof of Concept – *Evidence, typically deriving from an experiment or pilot project, which demonstrates that a design concept, business proposal, etc. is feasible.* [86]

HRV Heart Rate Variability – The variability in intervals between cardiac cycles.

Appendix B

Contents of enclosed CD

Available online

The contents of the attached medium are also available at:
https://github.com/faedrielle/fit-bachelors-thesis-data.git

Visualization

`readme.txt`	the file with CD contents description
`exe`	the directory with executables
`src`	the directory of source codes
`tizen-app`	smartwatch app sources
`android-app`	smartphone app sources
`thesis`	the directory of LaTeX source codes of the thesis
`text`	the thesis text directory
`thesis.pdf`	the thesis text in PDF format