CONCEPTION AND IMPLEMENTATION OF A MOBILE APPLICATION WITH FITNESS TRACKERS AS SUPPORTIVE TOOLS FOR COMPUTED STRESS DETECTION

Marcel Heil

Bachelor Thesis

Aufgabensteller  Prof. Dr. François Bry
Betreuer  Prof. Dr. François Bry, M. Sc. Yingding Wang
Abgabe am  04. August 2016
Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst habe und keine anderen als die angegebenen Hilfsmittel verwendet habe.

München, den 04. August 2016

Marcel Heil
Abstract

Stress is a common problem in our contemporary society, especially among students and professionals. The pressure and demand to deliver excellent results in their studies or at their work places is constantly growing. To overcome this pressure, plenty of them are using wearables as smart watches or fitness trackers along with their smartphones to monitor their daily activities and to measure physiological data. These data can provide valuable outcome to detect and prevent from possible stressors, but there is a lack of mobile applications unifying the variety of measurements to an user-friendly and self-supporting tool.

This bachelor thesis presents the conception and implementation of a mobile application, which uses heart rate data provided by state-of-the-art fitness trackers to derive the computed stress, a term that will be elaborated later in this thesis. Initially, the accuracy and usability of several fitness trackers has been examined in a pilot study and afterwards evaluated to derive the heart rate variability score, which is a well-known and accepted measurement for stress detection according to the best knowledge of the author of this thesis.

The proposed mobile application in this work presents the computed stress in an user-friendly interface and gives the users the ability to merge their activities and self-assessment of stress levels with the computed stress. This fusion provides a modern approach for stress detection using current technologies while it focuses on the real-life and not on laboratory scenarios. The latter case is conducted by many of the former scientific studies.

Diese Bachelor Arbeit präsentiert die Konzeption und die Implementierung einer mobilen Anwendung, welche die von aktuellen Fitness Trackern gemessenen Herzdaten verwendet um den Computed Stress herzuleiten, welcher später in dieser Arbeit genauer erläutert wird. Zu Beginn wurde die Genauigkeit und die Nützlichkeit von verschiedenen Fitness Trackern in einer Pilotstudie getestet und danach evaluiert um die Herzraten Variabilität herzuleiten, welche nach dem besten Wissen des Autors ein bekannter und angesehener Messwert zur Stresserkennung darstellt.

I want to thank Prof. Dr. Bry for giving me the opportunity to write this bachelor thesis. Prof. Dr. Bry always supported me with valuable feedback on my work and showed great interest in it. The regular meetings with him and M. Sc. Yingding Wang produced new ideas and guided this bachelor thesis into the right direction.

M. Sc. Yingding Wang has been my supervisor through the project. He has been my first counterpart and always available to answer any of my questions via mail, our communication tool Slack or personally. M. Sc. Wang helped me to get insight into the professorships project Backstage and offered lead when needed. He assisted me to structure the project with weekly meetings and has been always one step ahead.

I also want to thank everyone who participated in the pilot study and the second user study. The participants have been very motivated during the studies, they immediately reported problems with the devices or apps and gave important feedbacks, which helped to create new ideas and to optimize the proposed app. Their feedback may also help to improve future versions of CStress.
## Contents

1 Introduction .............................................. 1
  1.1 Motivation ........................................... 1
  1.2 Overview ........................................... 2
  1.3 Computed Stress ...................................... 3
  1.4 Objective ........................................... 5

2 Conception & Design ..................................... 7
  2.1 Pilot Study .......................................... 7
    2.1.1 Methods ........................................ 7
      2.1.1.1 Participants ................................ 7
      2.1.1.2 Devices ..................................... 7
      2.1.1.3 Procedure ................................... 9
      2.1.1.4 Data handling and analysis .................. 9
    2.1.2 Findings ....................................... 10
      2.1.2.1 Pilot Study ................................ 10
      2.1.2.2 Post Pilot Study Survey .................... 11
  2.2 Computed Stress Model ............................... 12
    2.2.1 Measuring the heart rate ....................... 12
    2.2.2 Heart Rate Variability ......................... 12
    2.2.3 HRV Score ...................................... 13
    2.2.4 Computed Stress ................................ 14
  2.3 Design ................................................ 15
    2.3.1 Mockup ......................................... 15
    2.3.2 Design Process ................................ 16

3 Implementation & Evaluation ............................ 17
  3.1 Tools ................................................. 17
    3.1.1 Android Studio .................................. 17
    3.1.2 SQLite Browser .................................. 18
    3.1.3 Slack ............................................. 18
  3.2 Features .............................................. 18
    3.2.1 Functional Features ............................. 18
      3.2.1.1 Download of the Heart Rate Data ........... 18
      3.2.1.2 Computed Stress Chart ....................... 19
      3.2.1.3 Activity List ................................ 19
      3.2.1.4 Upload of the User Data .................... 19
    3.2.2 Non-Functional Features ....................... 20
CHAPTER 1

Introduction

1.1 Motivation

In the last couple of years, a working group of the Institute of Informatics at the Ludwig-Maximilian University (LMU) in Munich developed the learning platform Backstage\(^1\). On the one side, Backstage strives to help students actively to engage, even in large lecture classes as common at the LMU. On the other side, it enables the lecturer to become aware of students thoughts and to figure out which topics students find difficult. But in the future, Backstage will go further. Lecturers should be able to get information about stress levels of students during lectures, while studying and even during exams. There can be many reasons for students to be stressed:

- Too much learning material
- Complex topics
- Complex or too many exercises
- Number of exams
- Tight timetables
- Private problems (family, partner, friends)
- Financial problems
- Side job(s)

Those and many more possible individual stressors in their daily life’s need to be considered. Adjusting the teaching to students stress levels may not be the solution for all of these stressors, but could help them to be more successful in their studies. The lecturers could also take advantage, as they would be able to align their teaching to the students skills and capabilities.

\(^1\)URL: http://backstage.pms.ifi.lmu.de
1.2 Overview

Traditional studies about stress detection are focused on the self-assessment of attendees or on observing them during the execution of specific tasks. Clinical studies are conducted under laboratory conditions and usually only ascertain short or long term physiological data. Studies with fitness trackers are often focused on very specific use cases such as improving the performance of sport athletes. There are just a few studies connecting the self-assessment of attendees with the physiological data gathered by fitness trackers [19], but almost none of them attempts to observe students or professionals during their everyday life [24].

Recently, technology and lifestyle companies are pushing a huge amount of new wearables in the form of smart watches and fitness trackers onto the consumer market. They are available in many different styles, colors and with varying sensors and functions. Especially fitness trackers are broadly becoming accepted among the society. They can be spotted on the wrists of digital natives but also on those of digital immigrants. Digital natives is a term for the people that grew up with computers, video games, the internet, smartphones and social media [30]. In opposite to digital natives, which are accepting this technologies as natural, digital immigrants had their first experiences with modern technologies as adults [30]. Furthermore, the number of low-budget smart watches and fitness trackers is increasing rapidly, what also drives the sales figures [17]. For the average user, there is no need to buy professional and expensive devices from popular manufacturers like Garmin² or Polar³.

The wearables are used both as accessories as well as supporting tools to record daily activities and to measure physiological data. As fashion items, they can be seamlessly integrated into the lifestyle. Especially trending companies like Apple, Samsung or Fitbit are advertising their smart wearable devices as fashion statements, not only as geeky gadgets. Activities such as moving, eating or sleeping are recorded and physiological data like steps, calories, heart rate and more is collected with the devices sensors. The sensors can be integrated into sports equipment, be attached to someones clothes or be worn on the body as bracelets [17] or even as earbuds, as the startup Bragi does with their first wireless smart earphones Dash⁴.

Together with the steadily evolving mobile applications (apps), provided by the wearables manufacturers or third-party developers, and popular health platforms such as Google Fit or Apple Health, users are able to overview the collected set of information easily and in real-time. However, the big number and variety of fitness apps or platforms can also lead to a desperate search for a high quality app [17]. Using the app developed by the devices manufacturer is usually the best choice, but there are also negative examples like the Mi Fit app for the Xiaomi Mi Band 1s, as it lacks important features such as continuous heart rate monitoring.

The collected data can help to maintain someones health, or to even improve it. Professional sports athletes are using smart wearable sensors to improve their performances [15], and for people fighting against obesity or heart diseases, they can play an important role for tracking their healing progress [4]. Even so the sensor data of wearables has already been used in a broad range of scientific studies, there is a lack of studies that are focusing on detecting stress.

Stress is a cause of many diseases [24] and a huge problem for many people in our fast-paced society. Especially students, who are continuously under pressure to deliver excellent

²URL: https://buy.garmin.com/de-DE/DE/c10002-pl.html
³URL: http://www.polar.com/de/produkte
⁴URL: http://www.bragi.com
results, and professionals, who are suffering because of their enormous workload and tight schedules, are more frequently becoming victims of stress.

As many people are wearing their fitness tracker the all day, why not take advantage of this? Most of the devices provide a lot of valuable data [11] which could be connected to the users self-assessment [24] and to their daily activities, but has not been done yet. Before doing this, it has to be proven that the measurements of fitness trackers are accurate enough [6]. There are already some studies that address the accuracy of popular fitness trackers or smart wearable devices from Fitbit, Apple, Samsung, Garmin and many others. Those studies mainly focus on readings like the heart rate or steps [4], as those sensors are typically implemented into the devices. The overall results of this studies are positive and show that most of the fitness trackers are doing a good job [4]. Of course they are not as precise as clinical devices, but together with the right algorithms they are able to provide meaningful results [6].

Even so it is already proven that fitness trackers can deliver valuable data, the two devices used for this bachelor thesis (Figure 1.1) needed to be tested in terms of accuracy and usability. The Charge HR from the Californian company Fitbit and the Mi Band 1s from the Chinese company Xiaomi are devices in different price classes and may return different results in their measurements (a comparison between the two fitness trackers can be found in section 2.1.1.2, p.7). Also it has not been clear if they both provide the functionality needed for the app which has been developed for this bachelor thesis.

To test the fitness trackers and to collect user data, which is needed for the creation of a computed stress (section 1.3, p.3) model and for the conception of the afterwards implemented application, a pilot study has been realized right at the beginning of the work for this bachelor thesis. The methods and findings of the pilot study are presented in section 2.1.

1.3 Computed Stress

Before introducing the computed stress, the term stress itself has to be specified. Selye described stress as “the non-specific response of the body for any demand made upon it”

5Picture source: https://www.fitbit.com/de/chargehr
In the introduction of his book “Stress without distress”, he wrote a beautiful and extensive definition:

All living beings are constantly under stress and anything, pleasant or unpleasant, that speeds up the intensity of life, causes a temporary increase in stress, the wear and tear exerted upon the body. A painful blow and a passionate kiss can be equally stressful. The financier worrying about the stock exchange, the laborer or the baseball player straining his every muscle to the limit, the journalist trying to meet a deadline, the patient fighting a fever, all are under stress. But so is the baseball fan who merely watches an interesting game, and the gambler who suddenly realizes that he has lost his last cent or that he has won a million dollars. Contrary to widespread belief, stress is not simply nervous tension nor the result of damage. Above all, stress is not something to be necessarily avoided. It is associated with the expression of all our innate drives. Stress ensues as long as a demand is made on any part of the body. Indeed, complete freedom from stress is death! [23]

Selye’s definition illustrates two major facts:

- Stress can be either a positive or negative feeling. A human’s bodily reactions indicate if someone is under stress or not, but this objective information has to be combined with the context.

- Stress should not be completely avoided! It is a natural reaction of the body and keeps humans alive. Without stress, there would be no life.

Sharma et al. allocated stress into different parts. For them, it is a complex reaction of the body that has several different components such as psychological, cognitive or behavioral ones [24]. Stress is influencing the health of people in many ways, but as humans are all not equal, everybody reacts differently to any event that may cause stress [24].

But even if stress is a very individual measure, there are bodily reactions that appear in similar patterns. Clinical studies with electrocardiograms show that attendees have a higher heart rate during stress conditions than in rest [19]. Though, the pilot study for this paper pointed that the heart rate alone is not appropriate enough to use it as an indicator for stress, but it can be derived to more meaningful values.

Through the literature review, the heart rate variability (HRV) has been found as a valid indicator for mental stress [16]. The HRV measures the intervals between two consecutive heart beats (RR) and considers the variations of them. HRV is commonly used in scientific research and also in clinical studies [16]. The HRV can be further calculated to a HRV score, which represents “the strength of the autonomic nervous system at a given time” [5]. In other words, it represents how good a human body can adapt to any demand for change. A lower HRV or HRV score indicates a higher amount of stress and vice versa [16]. More details about the HRV and HRV score are presented later in section 2.2.2, p.12 and section 2.2.3, p.13 of this thesis.

For this bachelor thesis, the term computed stress is defined as the inverted HRV score (for more details, see section 2.2.4, p.14). But why not use the HRV score itself? Because the human intuition would correlate a higher score with a higher amount of stress [16]. Inverting the HRV score to the computed stress supports the humans understanding and prevents from misinterpretations. This concept is especially important for the visualization of the computed stress in the app named CStress. This app has been conceptualized and implemented for this bachelor thesis. Sharma et al. described computed stress as “stress computationally derived from instantaneous measures of stress symptoms obtained
1.4. OBJECTIVE

by non-invasive method” [24]. This definition shows similarities to the process how the term computed stress has been developed for this thesis. It also focuses on instantaneous measures (the heart rate) obtained by non-invasive methods (fitness trackers).

1.4 Objective

To date, self-assessments through questionnaires have been a good measure of stress. But now, as fitness trackers with different kinds of sensors are becoming widespread available, it is time to bring objective methods into the game. This could help to gather more qualified data [24]. It is still rare to measure stress in real-life conditions [19], but fitness trackers provide the possibility to monitor the physiological reactions of a body continuously. Moreover, as almost everybody owns and uses a smartphone, the collected data can be processed in real-time and visualized in apps.

But there are some challenges that have to be tackled. The accuracy of fitness trackers may be perfect, but if the app provides a bad user interface (UI) and tends to predetermine the physiological data instead of leaving the interpretation up to the user, it would be useless [4]. Furthermore, the usage of digital devices to recognize stress can again lead to stress for the user [22].

This bachelor thesis approaches to conceptualize and to implement an app that offers solutions for aforementioned challenges. Chan et al. stated in their paper, that a real-time stress monitor could provide subjects with continuous feedback about their stress levels [2]. Once connected with the users activities and their self-assessment, and presented in an user-friendly interface, this feedback could support the early detection of stress and help to improve human life [8].

Especially students and professionals could benefit from this approach, as they would be able to detect possible stressors in their fast-paced everyday life.
This chapter presents the methods and findings of the pilot study, elaborates the term computed stress and introduces the design concept for the proposed app CStress.

2.1 Pilot Study

2.1.1 Methods

A pilot study has been conducted to gather usable heart rate data from the fitness trackers and to test the devices usability and accuracy. Five participants have been wearing the fitness tracker continuously for the time of one week. During the study, participants have recorded their activities and their stress levels at an hourly rate. After the pilot study, the participants have been asked to complete a survey, which has been used to collect feedback about their satisfaction with the fitness trackers and to improve further studies.

2.1.1.1 Participants

Five participants attended the pilot study. Two are students (one of them is the author of this thesis), one is a trainee at a magazine publisher. The other two participants are scientists at the LMU. All of the participants are male.

2.1.1.2 Devices

Two different fitness trackers have been used to collect the participants activity data during the pilot study. Fitbit’s Charge HR and Xiaomi’s Mi Band 1s. A comparison of them and their features can be found in table 2.1 on page 8.

Fitbit Charge HR

The Fitbit Charge HR is one of the premium activity trackers currently available on the market. With a manufacturer’s suggested retail price of 149,95 Euro, it is a high-end product. It offers all of the well-know functions that users expect in that price sector. Those are heart-rate monitoring, step, track, calorie and automatic sleep tracking, automatic wrist
detection as well as call notifications. It also has a display, which can automatically show
the time when turning the arm and some other information like the current heart rate, steps
taken or burned calories when clicking through with the side button.

The Charge HR can be synchronized with the so called Dashboard on Fitbit’s website\(^1\)
and/or with the Fitbit application on iOS or Android, but Fitbit lacks the integration to the
platforms Google Fit (Android) and Apple Health (iOS). There are a couple of apps which
can be used instead of Fitbit’s app. Those applications are using the Fitbit API, which gives
developers the ability to read and write methods and collect the data from the trackers.

**Xiaomi Mi Band 1s**

The Xiaomi Mi Band 1s is the successor of the chinese companies first ever fitness tracker
Mi Band, which was introduced in July 2014. It adds a heart rate sensor to the following
features. These are a step and sleep tracker, a sleep-cycle smart alarm, water resistance (IP67)
and vibration alerts for calls and notifications. With an price range of 15,- to 25,- Euro it is
a low-end and entry-level product. The Mi Band 1s does not include a display, it just has
three LEDs on top of the sensor. Those can indicate the charging level of the tracker or give
feedback on activity milestones.

The Mi Fit application, provided by Xiaomi, is the first choice to synchronize the data
of the Mi Band 1s. It has an user-friendly interface, but lacks some kind of information
that third party applications provide. The Mi Band Notify & Fitness\(^2\) app, which has been
used for this pilot study, is able to monitor the heart rate continuously and provides more
options for customization. Xiaomi does not provide a public API for third party developers.
Still some independent developers have reverse-engineered an API to access to the devices
sensors.

<table>
<thead>
<tr>
<th>Device</th>
<th>Fitbit Charge HR</th>
<th>Xiaomi Mi Band 1s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price</td>
<td>130,00 - 149,95 Euro</td>
<td>15,00 - 25,00 Euro</td>
</tr>
<tr>
<td>Weight</td>
<td>unknown</td>
<td>5.0g (sensor only)</td>
</tr>
<tr>
<td>Display</td>
<td>Yes (OLED)</td>
<td>No (only LED’s)</td>
</tr>
<tr>
<td>Color Options</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Bluetooth</td>
<td>v4.0 (required)</td>
<td>v4.0 (required)</td>
</tr>
<tr>
<td>Water Resistance</td>
<td>sweat, rain, splash</td>
<td>IP67 certified</td>
</tr>
<tr>
<td>Temperature</td>
<td>-20°C to 45°C</td>
<td>-20°C to 70°C</td>
</tr>
<tr>
<td>Heart Rate Monitoring</td>
<td>Yes, optical</td>
<td>Yes, optical</td>
</tr>
<tr>
<td>Google Fit Integration</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Apple Health Integration</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mobile App</td>
<td>Fitbit (iOS &amp; Android)</td>
<td>Mi Fit (iOS &amp; Android)</td>
</tr>
<tr>
<td>App Quality</td>
<td>Good</td>
<td>Average</td>
</tr>
<tr>
<td>Sleep Tracking</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Smart Sleep Alarm</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Smartphone Notifications</td>
<td>Call Notifications</td>
<td>Call Notifications</td>
</tr>
<tr>
<td>Body Placement</td>
<td>Wrist</td>
<td>Wrist</td>
</tr>
</tbody>
</table>

\(^1\)URL: https://www.fitbit.com/login
\(^2\)URL: https://play.google.com/store/apps/details?id=com.mc.miband1&hl=en
2.1. PILOT STUDY

2.1.1.3 Procedure

At the beginning of the pilot study, the fitness trackers have been synchronized with the participants' smartphones. Anonymous user accounts have been prepared for every participant to allow access to the collected data after the study. All of the participants preferred to use the Fitbit app for iOS or Android for their Fitbit Charge HR. The app has worked fine for two of them, but one participant has experienced problems while synchronizing the fitness tracker with his phone. He has only been able to use Fitbit’s web interface. It offers the same functionalities as the app, but lacks the automatic synchronization feature, so it has been necessary to connect the fitness tracker with the computer from time to time. Xiaomi does not provide a web interface, but also developed an Android and iOS app called Mi Fit. Mi Fit provides the expected basic functionalities, but it lacks continuous heart rate monitoring. This is why another app has been used for the study. The participants have been using the third party app Mi Band Notify & Fitness as it provides the possibility to monitor the heart rate continuously.

<table>
<thead>
<tr>
<th>Timeframe</th>
<th>...</th>
<th>16:00 - 17:00</th>
<th>17:00 - 18:00</th>
<th>18:00 - 19:00</th>
<th>19:00 - 20:00</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main activity during the given timeframe</td>
<td>Studying scientific papers</td>
<td>Driving the car</td>
<td>Sports (running)</td>
<td>Relaxing on the sofa</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stressed (0-2)</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comment</td>
<td>I was interrupted through some messages on Facebook...</td>
<td>traffic jam...</td>
<td>way more positive stress</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2.1: Example of the form for the pilot study

Subsequently, the form (exemplary in figure 2.1) has been introduced to the participants. The form has one column for every hour during a day. For each hour, they have been asked to record the activity and their self-assessed stress level on a scale from zero to two. Zero stands for no stress, one for a moderate stress level and two for a higher stress level.

During the execution of the study, the survey has been prepared. The goal of this survey is to receive feedback about the fitness trackers and their inherent apps, as well to improve further studies. This survey consists of 16 questions in total, starting with asking for problems the participants encountered, following up with questions about the best and the worst features of the fitness trackers, as well as the participants satisfaction with the devices and the apps. In addition, the participants have been asked for any behavior changes they have made because of the fitness tracker and if they are willing to buy one of them for personal use.

2.1.1.4 Data handling and analysis

The forms of all the participants have been collected to analyze the data after one week. Some difficulties occurred while extracting the heart rate data from the devices. Fitbit offers an export of the activity data to a comma-separated value (CSV) file or to an Excel file, but the heart rate data is exclusive and can not be downloaded by the average user. This is
why the open source Fitbit Powershell Module by Collin Chaffin\(^3\) has been used - it offers the possibility to extract the heart rate data on a day by day basis to an Excel file. Though, before using the program, a developer account for Fitbit’s API has to be created because Fitbit only allows the download of heart rate data to apps for personal use.

The third party Mi Band Notify & Fitness app already offers the heart rate data export to a CSV file. It also supports automated integration and synchronization with Google Fit. This feature makes it possible to easily receive the hourly average for the heart rate of every participant using the Xiaomi Mi Band 1s.

Up next, all the data from both devices have been processed to a uniform format. For this purpose, a tiny program called CSV Parser has been written in Java. It calculates a unique timestamp for every reading and pairs it together with the belonging heart rate. A second program called HRV Calculator has also been created with Java. HRV Calculator takes the CSV file with the unified data as input, calculates the RR intervals, the RMSSD, the \(\ln\text{(RMSSD)}\) and most importantly, the HRV score (More about this terms in section section 2.2.3, p.13). The algorithms for these programs are also used for the implementation of CStress.

### 2.1.2 Findings

#### 2.1.2.1 Pilot Study

Stress cannot be measured solely with the raw heart rate data. It has to be processed to more meaningful values. This is the most important finding of the pilot study. The pilot study has also shown that Fitbit’s fitness tracker has some advantages towards Xiaomi’s Mi Band 1s, especially in terms of accuracy. Fitbit’s Charge HR and Xiaomi’s Mi Band have tracked ten-thousands of readings through the one week period. After calculating hourly average values and combining them with the participants self-assessment and their activities, it was clear that more work needed to be done.

The pilot study has also revealed some technical limitations of the fitness trackers. To start with the worst, Xiaomi’s Mi Band 1s does not provide continuous heart rate monitoring, at least not the device itself. In order to do this, it needs the control of a third-party application. Xiaomi’s in-house app Mi Fit lacks this feature - it only allows periodically measurements at night. On the contrary, the already introduced Mi Band Notify & Fitness app implements continuous heart rate monitoring. For continuous heart rate monitoring, users can choose among different interval settings, for example every second, five seconds, ten seconds, and so on. The problem is, that the application needs to run in the background while doing this. In addition, the smartphone has to be close to the Mi Band as they are connected via bluetooth. Background tasks and bluetooth both drains the battery of the smartphone - this is why the Samsung Galaxy S2, which has been used for the study, needed to be re-charged at least two times every day.

Fitbit’s Charge HR offers a smarter solution. It measures the heart rate independently and automatically in five to ten second intervals. While workouts, this intervals can even shrink to one second. The Charge HR does not require a continuous connection to a smartphone. To synchronize the data, it just needs to be connected to a smartphone or a computer time by time. Even if Fitbit’s tracker is doing all the work on its own, the experience through the pilot study has shown that the battery still lasts up to five days.

\(^3\) URL: https://github.com/CollinChaffin/psFitbit
Another very critical limitation is the accuracy of the heart rate sensors. At this point, the Mi Band has brought up more weaknesses. Due to its tiny appearance, the fit of the wristband has always been too loose or to strong. For that reason and because of possible technical limitations of the sensors, the Mi Band has regularly produced reading errors. The heart rate readings are generally too high and the records have also shown regular and unexpected swings. Fitbit’s tracker instead produces constant and reliable results. Those experiences and also the fact that Xiaomi does not provide an API to access the sensor data (Fitbit does) lead to the decision, that the Mi Band is not integrated into the CStress. Instead, the app seamlessly integrates Fitbit’s API and uses the sensor data of their fitness trackers for the computed stress model.

2.1.2.2 Post Pilot Study Survey

The goal of the survey was to figure out how and if the participants had been satisfied with the fitness trackers and their inherent apps. Participants have also been asked to give feedback about their favorite and non favorite features. In total, the survey contains 16 questions. To run the survey, the web tool Lime survey assisted. Lime survey is an open source web tool and provides the possibility to run the whole survey process automatically. After the survey had been completed by all participants, Lime survey extracted the results to an Excel file.

The survey results display several advantages and disadvantages of both fitness trackers. Fitbit’s Charge HR and the Fitbit app have received better feedback than its cheaper competitor Xiaomi Mi Band 1s. Especially Xiaomi’s lack of an useful and user-friendly app caused some problems during the pilot study. The participants had to use the previously introduced Mi Band Fitness & Notify app for Android instead, because it is way more useful and provides a lot more features than Xiaomi’s own app.

One problem emerged during the setup of a Fitbit wristband. One of the participants was not able to synchronize his fitness tracker with the smartphone, but he finally could connect it to his laptop and synchronize it via cable. A similar problem occurred while connecting the Mi Band to the smartphones as they normally not support Bluetooth v4.0 (which is required by the Mi Band 1s), but at the end a solution has been found to solve this issue.

Both of the fitness trackers have shown very good results for the battery life. The Fitbit Charge HR’s battery lasts up to five days - this is very impressive as it has been continuously tracking the heart rate and saving it in its internal storage. Xiaomi’s Mi Band 1s has an even longer battery life, but the problem is that it does not track the heart rate itself - the app needs to control the sensor in order to do this. This strongly drained the battery of the smartphones and caused them to turn off several times.

The rated favorite features of both fitness trackers are the sleep tracking and their possibilities to track workouts. Participants also mentioned that the wearing comfort of both devices have been great, especially because they are both very small and light. The lack of different sizes for Xiaomi’s Mi Band 1s can be an issue to adapted individual needs.

In general, all of the participants have been satisfied with their fitness trackers, but their opinions about the apps have differed. Fitbit’s app offers way more functionalities and have made a better impression than the third party app Mi Band Notify & Fitness. Quite contrary to this, the price point of Xiaomi’s Mi Band 1s is way better, as it costs just between 15 and

URL: https://www.limesurvey.org/de/
CHAPTER 2. CONCEPTION & DESIGN

25 Euro - Fitbit’s tracker charges between 130 to 150 Euro. But most important is the fact, that both systems are motivating the users to enhance their personal behavior.

2.2 Computed Stress Model

The term computed stress has already been presented in the introduction (section 1.3, p.3) of this paper. It is modeled as the inverted heart rate variability (HRV) score. Though, to derive the HRV score from the heart rate data, there are some important steps and terms that need to be elaborated.

2.2.1 Measuring the heart rate

How do the fitness trackers measure the heart rate from the wrist? They are measuring the blood volume pulse (BVP). “Blood volume is the amount of blood in a blood tissue during a certain time interval. BVP measures the amount of light that is reflected by the skin’s surface” [24]. Less reflected light - which is detected by sensors on the device - means a higher blood volume and vice versa. Fitness trackers such as the Fitbit Charge HR or the Xiaomi Mi Band 1s are using this procedure to measure the heart rate. The pilot study has shown that the accuracy of the measurements is highly correlated with the way how users have been wearing their wristbands.

Is the wristband too loose or overtightened, the overall quality of the measurement suffers. Movements can also have negative effects, especially during sport activities. But the technology required to measure someone’s pulse with light has been dramatically improved during the last years. Hospitals are already using pulse oximeters to measure the amount of oxygen in the blood in a similar way [2]. The technology companies have developed intelligent algorithms which remove a lot of “noise” generated by the movements. This allows the sensors to measure the heart rate reasonably even when people are not resting.

Certainly fitness trackers cannot replace clinical methods in terms of accuracy, but instead they are able to monitor heart rate changes during the day - and every day. They can deliver a huge amount of readings which can be further processed to more meaningful values such as the HRV or HRV score.

2.2.2 Heart Rate Variability

The human heart is, while resting, beating between 50 and 80 beats per minute. A heart rate of 60 means that a heart is beating 60 times in a minute, so the average time between two consecutive heart beats is 1 second or 1000 milliseconds. The Heart Rate Variability is a measure of the variations in heart rate and the intervals between consecutive heart beats (RR) [16]. Advances in technologies (small and more accurate mobile devices), especially through the last couple of years, allow us to measure RR intervals with pulse watch systems [21] as they are build in fitness trackers such as the Fitbit Charge HR or Xiaomi Mi Band 1s.

One could think that while a human organism is increasing its HRV, stress is increasing too. But healthy human hearts are continuously changing the distance between heart beats to adapt to cognitive and/or physical loads [3]. Humans are experiencing stress if a heart is not able to adjust the HRV appropriately to their current load [3]. The master thesis “Eustress and Distress Detection from Physiological Data using Supervised Learning”[4] of Patrick Hagen further investigates the relation between the HRV and stress. He divides stress into three areas:

5Project website: http://stila.pms.ifi.lmu.de
2.2. **COMPUTED STRESS MODEL**

- **Positive stress**: (Eustress) Irregular heart rate with regular changes among the consecutive RR intervals

- **No stress**: Irregular heart rate with irregular changes among the consecutive RR intervals

- **Negative stress**: (Distress) Regular heart rate with regular changes among the consecutive RR intervals

HRV can be a reasonable indicator of health, but it is also a very individual measure. It depends on a set of factors, namely endogenous, exogenous and constitutional ones. Endogenous factors are the non influenceable physiological parameters such as the age or gender [21]. Exogenous factors are influenceable lifestyle factors which can be the physical fitness, sports activities, someone’s body weight or the ingestion of drugs [21]. Constitutional factors are for example the genetics or circadian rhythms [21].

All those factors can have positive and negative influences on the HRV, so all of them have to be considered when analyzing and interpreting it. The computed stress model developed for this bachelor thesis aims to support users to derive their own interpretations. They should be enabled to connect their own feelings with the visualized computed stress levels.

### 2.2.3 HRV Score

The HRV score is a one-dimensional variable that can be derived from the raw heart rate data. It is calculated from the root mean square of the successive differences (RMSSD), which can be seen as a snapshot of the Autonomic Nervous System [5].

To receive the RMSSD and the final HRV score, a sequence of calculations need to be done. This sequence can be broken down into the following steps:

\[
RR = \frac{60}{Heart \ rate} \quad (2.1)
\]

\[
RMSSD = \sum_{i=1}^{n} (RR_i - (RR_{(i-1)}))^2, \quad n := Number \ of \ heart \ rate \ readings \quad (2.2)
\]

\[
\ln(RMSSD) \quad (2.3)
\]

\[
HRV_{score} = \ln(RMSSD) \times 20 \quad (2.4)
\]

- RR (2.1): Represents the interval between two consecutive heart beats. An RR interval is calculated by dividing 60 through the heart rate [5].

- RMSSD (2.2): Represents the root mean square of the successive differences. This value is a traditional and well researched measure of Autonomic Nervous System activity [5].

- Applying a natural logarithm (ln) to the RMSSD (2.3): The RMSSD behaves logarithmically. This makes it difficult to conceive the scale of its variations. In order to create a number that behaves in a more linear way, the ln is applied [5].

- Expanding the ln(RMSSD) (2.4): Normally the values of the ln(RMSSD) are in a range of 0 to 6.5 [5]. In order to create a more accurate scale, it is expanded with the factor 20 [5]. This is the final HRV score, which typically ranges between 0-100.
As previously reported, the HRV is a very individual measure. This is why the personal HRV score should not be compared to the score of others. Instead, users should analyze personal trends and use it to detect possible stressors.

Higher HRV scores usually correlate with a younger (biological) age, regular sports activities, better fitness, eating healthy food and many other factors. Changes in lifestyle can deeply affect the various factors and so the HRV score, but users should not be worried if their scores are lower on any given day. It is more important to observe the long term changes and to improve the HRV score over the time.

### 2.2.4 Computed Stress

Computed stress has already been specified as the inverted HRV score. The HRV score is not directly used for the visualization in CStress, because it could lead to misinterpretations [16]. A higher computed stress level represents a lower HRV score and vice versa. This makes it easier for users to understand the chart.

Most of the time the HRV score ranges between 50 and 80 [5]. In some cases it can even climb above 80 or come below 50. The users of the app Elite HRV have an average HRV score of 60 and the top ten percent of its users are scoring above 74 [5]. Elite HRV is one of the most popular apps working with HRV scores, so it can be taken as a good reference for CStress’s computed stress model.

\[
Stress_{\text{computed}} = 110 - HRV_{\text{score}}
\]  

(2.5)

To calculate the computed stress, the HRV score is subtracted from the constant 110 (2.5). This constant value is great enough to intercept even the highest HRV scores, so all of them fit into a scale from 0 to 100. Since the HRV scores are mostly located between 50 and 80, the aforementioned subtraction (2.5) transforms most of the HRV scores into computed stress values from 30 to 60.

For the computed stress model of this thesis, the threshold of determining whether a user is stressed or not is defined as a computed stress value of 50. As the intuition would expect, a lower computed stress value than 50 means that less bodily stress is experienced and a higher computed stress value than 50 indicates a higher stress level. CStress presents these values to the user in a graphical representation (a chart). As an opportunity for improvement, in future versions of CStress the threshold shall be adapted to the users’ individual conditions such as age, gender, body size, body weight and etc.

The HRV score and the computed stress is calculated for every ten minutes. This ensures that there is a sufficient amount of heart rate inputs. Fitbit’s Charge HR tracks the heart rate every five or ten seconds, so for a time period of 10 minutes there are between 60 and 120 heart rate measurements available. Traditionally the HRV is measured with clinical devices such as electrocardiography (ECG) [8], where the heart rate is tracked continuously and not in five or ten second intervals as the Fitbit Charge HR does. This procedure normally delivers 60 or more readings per minute. While developing the computed stress model for CStress, it has been experimented with calculating the HRV score and the computed stress value for every five minutes. But the results showed that a ten minute period is less error-prone.
2.3 Design

2.3.1 Mockup

To generate a mockup (figure 2.2, p.15) of the application’s user interface, a storyboard which shows possible application screens and their lifecycle has been created via paper prototyping.

When starting the application, users land on the main screen. The main screen provides an overview of the current day, with the computed stress and the activities below. Different colors indicate the different computed stress levels in the chart. Activities are ordered chronologically and are visualized with icons and the activities names.

New activities are addable with a long click on the chart. The long click opens a new screen, where users can choose an activity out of a pre-defined list. If an activity has been chosen, users are able to adjust the time range for it. Once the time range has been assigned to the activity, the new activity is added to the main screen.

Of course, users can also access historic data within the application. It is possible to choose any date in the past to review the computed stress and the activities. Though, the visual integration of historic data has not been concerned at the beginning. Moreover, it has been developed during the implementation phase.

The storyboard also shows a navigation menu, from where the users can access the different screens such as the main screen (“Start”), screens to review the historic data, or screens such as a help or about screen. The menu is accessible from almost any place inside the
application via a hamburger button in the top left corner of the screen.

2.3.2 Design Process

A storyboard helps to visualize possible user interactions with the app. However, it is just a draft and the first ideas normally need to be refined several times through the implementation process. Asking different people for feedback is the best way to improve the design process. This is the reason why the app design has been discussed multiple times before finalizing it.

The app has been presented to friends, family, fellow students and researchers periodically. Their reviews unveiled bugs, developed new ideas and helped to keep the design focus on the user experience. This was especially important as CStress has been developed solely by one person, namely the author of this thesis.

Developing an app alone can sometimes be helpful, as the developer is aware about every aspect of the program and is able to make changes or correct errors quickly. But it can also lead the developer to loose the focus, especially when talking about design. Paying attention to classic principles of good design and guidelines like Google’s Material Design for Android helps to prevent oneself from this sacrifice.

Switching to the technology side of the implementation process, there are similar challenges to be tackled. It is important for the developer to use common design patterns and to organize the project. Design patterns are helpful for structuring the code and ease the reuse of code fragments. Organizing the project in hierarchies such as packages, folders and classes makes it accessible for the developer himself and for others while reviewing it.

The Integrated Development Environments (IDE) helps to take a lot of workload from the designers and developers. Activities or fragments can be designed and previewed in the IDE and integrated into the code. They also enable easy and quick testing of the app on an emulator or, if available, on a test device. With this possibilities, the changes in code and design can be reviewed instantly.
The main task of this bachelor thesis is to implement a mobile application for Google’s mobile operating system (OS) Android. This application visualizes the computed stress and supports the users to record and overview their daily activities. On one hand, the users are able to relate their self-assessments to the computed stress presented in the chart. On the other hand, it is possible for the professorship to access the collected data from the users via a REST-API (for more details, see section 3.4.4, p.28). This data can be valuable for further studies of the professorship and to improve the mobile application in upcoming updates.

3.1 Tools

To develop CStress, the following tools have been used:

3.1.1 Android Studio

Android Studio 2.1.2\(^1\) is the latest release of Android’s official integrated development environment (IDE). It is based on JetBrains’ IntelliJ IDEA software\(^2\). Android Studio provides helpful features such as code highlighting, auto completion of code and the possibility to include (open source) libraries and extensions. Developers can also easily design activities in XML and preview them inside the IDE before testing them live on a device or in the emulator. However, the emulator is not the best choice to test Android apps. Compiling an app takes a significant higher amount of time and there are some functions that cannot be tested with it. Testing on real devices always delivers more accurate results and is the only way to be sure that a given feature works on a given device.

Unfortunately there has been only one device model available through the implementation of CStress, a Samsung Galaxy S2 with Android 5.1.1 (APK 22). This fact caused a problem especially at the release of the app. Installing the app on other devices worked fine, but some participants of the second user study have not been able to start the app properly, because the system language has not been set to English as it has been on the test device. The system language automatically determines the date format. If the system language

\(^1\)URL: https://developer.android.com/studio/index.html
\(^2\)URL: https://www.jetbrains.com/idea/
has been set to German (or other languages), the app could not parse the date and crashed. To resolve this problem and to protect the app from such failures, the code needed to be adjusted. This has been a good example for the fact, that especially Android apps should be tested on many different devices and systems before releasing it.

### 3.1.2 SQLite Browser

SQLite Browser is an open source app which provides the possibility to create, modify and delete SQLite databases. As CStress implements several SQLite databases, this app helped especially with the creation and modification of those.

### 3.1.3 Slack

Slack is a communication platform especially made for development teams, but it is also used for the communication in large companies. The standard version can be downloaded for free and already provides a lot of features [25]. Conversations can be organized in different channels to represent different projects or topics [25]. Users can also communicate via private messages [25]. The app allows the sharing of almost any kind of files and the integration of many tools [25].

Slack has been used for the communication between the author of this bachelor thesis, the supervisor M. Sc. Yingding Wang and Patrick Hagen, the author of the master thesis “Eustress and Distress Detection from Physiological Data using Supervised Learning”. The two works are part of a research project at the institute of informatics at the LMU. Therefore, it has been necessary and very helpful to exchange ideas and thoughts between each other through Slack.

### 3.2 Features

#### 3.2.1 Functional Features

The app provides several functional features. These are the parts that relate to the function of the application - in other words: what the application does.

#### 3.2.1.1 Download of the Heart Rate Data

The most important feature is the download of the heart rate data (figure 3.1) recorded with the Fitbit wristbands. Users can start the download automatically with a click on the refresh button in the top right corner of the monitor screen for the current day’s data. To download data from the Fitbit servers, the application needs to access the Fitbit API. The synchronization process is explained in section 3.4.2. If users want to download data from any day in the past, they have to choose a date in the history tab. They are forwarded to a separate screen, visually equal to the today tab. The download process there is the same as mentioned before. Once the data has been downloaded, the application automatically refreshes the computed stress chart with the new data. If there is no data available, the application informs the users with a message.
3.2. FEATURES

Figure 3.1: Download of Fitbit data

3.2.1.2 Computed Stress Chart

The computed stress is visualized within a line chart (Figure 3.2, p.20). This chart is created with an open source library called MPAndroidChart\(^4\). MPAndroidChart offers plenty of different kinds of charts such as bar or bubble charts, but the line chart fits best to the visualization concept of CStress and can be modified appropriately. In CStress, the line chart uses gradient colors to indicate the different stress levels as defined in section 2.2.4 on page 14. A green color indicates a lower stress level, white stands for an average stress level and red for a higher stress level. It is important to notice that this visualization does not pay any attention to the fact that stress can be either positive or negative.

3.2.1.3 Activity List

Users are able to add activities to the activity list below the computed stress chart (figure 3.3, p.21). It is always possible to view the list and to add new activities, even if there is no chart data available. Users can choose out of a set of pre-defined activities or add a custom activity. For every activity, it is necessary to choose a feeling (neutral, positive or negative) and the time range. The feeling represents how the user felt during a certain activity, the time range represents how long an activity was accomplished. Depending on the feeling chosen by the user, CStress displays the activity with a green (positive feeling), white (neutral feeling) or red (negative feeling) background in the activity list. Activities can also be deleted with a long click.

3.2.1.4 Upload of the User Data

To evaluate the recorded user data, it shall be uploaded to a server (Figure 3.4, p.22). CStress sends a HTTP Post request with arrays containing the profile, heart rate, HRV, computed stress and the activity data in a JSON object. These data is not only necessary to proof the objectives of this bachelor thesis, they are also important for further studies. Especially the contextual data such as the profile and activity data in combination with the computed stress could help to detect stress in a more accurate and specific way. For example, Patrick Hagen’s master thesis “Eustress and Distress Detection from Physiological Data using Supervised Learning”, which is currently still in progress, focuses on the detection of eustress and

\(^3\)URL: http://sqlitebrowser.org

\(^4\)URL: https://github.com/PhilJay/MPAndroidChart
Those terms are dividing stress into positive (eustress) and negative (distress) directions. However, the detection of eustress and distress, especially with non-invasive methods, needs far more research than the detection of stress itself. Especially eustress is a weakly researched field.

### 3.2.2 Non-Functional Features

The non-functional features are the parts of the app that do not directly relate to the function of the app.

#### 3.2.2.1 Fragments

CStress uses fragments to build a multi-pane user interface. Fragments can be described as a modular section of an activity, in CStress the main activity. When the activity is paused or destroyed, the fragments take on the same behavior. As long as the activity is running, fragments can be added or removed independently. They are also added to a back stack managed by the activity, which allows to return to a fragment, for example with a back button. Though fragments are dependent on the activity’s lifecycle, they have their own lifecycle and define their own view layout.

#### 3.2.2.2 Stability

To provide a stable and flawless user experience, CStress handles possible exceptions and informs users via dialogs or toasts (Android’s system messages) about events or problems. There are several exceptions which have to be caught. The authorization process, the retrieving of access tokens and the download of user data from Fitbit require an internet connection via the mobile network or a wireless access point. If there is no connection to the internet, the user gets informed and cannot accomplish the requested task. When there is no
3.3. USER INTERFACE

Figure 3.3: Adding new activities to the activity list

data available to download, for example when a user did not wear the fitness tracker during a certain day or time, CStress informs the user with a message. The app also prevents the users from unintended actions, as it prompts them to confirm commands like deleting an activity or disconnecting from the Fitbit account.

3.2.2.3 Security

As CStress connects with the users Fitbit accounts and computes sensible data such as heart rate or sleep data, security is one of the most important layers. Fitbit’s API (section 3.4.1, p.26) already requires the usage of high security standards, as it works with the OAuth 2.0 protocol (section 3.4.2, p.27). This leads to a secure authorization process and token management. CStress stores the raw heart rate data in databases, computes further values and forwards them to the protected back end via the REST API. Other data such as the activity or profile data are stored in SQLite databases inside the application and are also forwarded to the back-end.

Once a user disconnects her Fitbit account from the application, all the data stored locally inside the application are deleted. This ensures that successive users of the app are not able to access data from previous users.

3.3 User Interface

CStress’s user interface has been developed around the latest Android material design guidelines. A main focus through the development is to provide an easy to use and self-explaining UI which supports users and protects them from unintended actions.
Google itself describes material design as “a visual language for our users that synthesizes the classic principles of good design with the innovation and possibility of technology and science” [9]. Inspired by the study of paper and ink, material design focuses on realistic surfaces and edges which can be easily recognized and understood by users [9]. Lights, surfaces and animations are elementary to transmit how elements move, interact and exist next to each other [9]. Print-based design, in other words elements such as typography, grids, space or color are guiding users through the interface [9]. They help to detect hierarchies and importances between elements [9]. User actions are triggering motions which can lead to changes in the interface, but these changes never destroy the continuity of the users experience [9].

Material design can be regarded as a three-dimensional environment whose elements are light, shadows and the material itself [9]. All elements have x-,y- and z-axis dimensions and a z-axis position in the 3D world [9]. There are two different kind of lights, which are creating different kinds of shadows [9]. The key light creates a directional shadow, while the ambient light creates a soft shadow from all angles [9]. In Android, shadows are created by the elevation difference between overlapping materials [9].

CStress is using material design elements such as toolbars, a navigation drawer, raised and flat buttons or sliding tabs. In the profile fragment and the fragment where users can add activities, CStress also uses parallax effects for scrolling. The application design pays attention to classic principles of good design, such as the layout structure, typography, color schemes, icons, animations and so on.
3.3. USER INTERFACE

3.3.2 UI Elements

3.3.2.1 Colors

CStress focuses on a classic and clean black and white design (Figure 3.5, p.23). Black and white are two of the most emotive visuals in the nature. They meld together in a contrasting design, which creates a dynamic that is impossible to ignore for the users. Many applications are focused on colorful themes, representing colors of brands or trying to attract users. CStress takes advantage of this fact and differentiates itself from the majority. The black and white design provides endless opportunities, which other color schemes can not manage to generate.

Even so CStress focuses mainly on black and white colors, it uses a small set of muted colors to promote important elements of the user interface (figure 3.5, p.23). Based on Fitbit’s brand color, users can find shades of a turquoise green throughout the apps elements. Such elements are background or text colors of buttons, highlighted lines of tabs, the computed stress chart or background highlights for list elements. In opposite to the turquoise green, CStress uses a quiet red to highlight increased stress levels and activities with a negative feelings.

3.3.2.2 Typography

The typography is one of the leading elements that guides users through the interface. With the right typography, they are able to recognize hierarchies in the design. A carefully chosen size and style balances the contents density and increases the reading comfort. CStress uses font sizes from 12 scaleable pixels (sp) up to 24sp and different font thicknesses from regular
to medium to highlight text (figure 3.5, p.23).

3.3.2.3 Icons

Icons have to be differentiated in two categories: system icons and custom icons. System icons are there to symbolize a command, file or directory or to represent actions like delete, go back or refresh. Custom icons are there to support interface elements such as texts or buttons with a visualization. CStress uses system icons in toolbars, the navigation drawer or for buttons. Colored custom icons are used to support activities with graphical representations (figure 3.6, p.24). The icons are reduced to their minimal forms and appear simple, modern and friendly.

3.3.2.4 Animations

Visual feedback on user actions is important for guidance and can signal hierarchical and spatial relationships between elements. CStress implements a couple of simplified animations in its UI. The monitor provides a sliding tab layout where users can easily switch between today’s and history’s tab. Parallax animations in the profile and activity fragment (figure 3.6, p.24) are providing feedback to the users when scrolling. While opening the navigation drawer, popups or dialogs, the background is highlighted by darkening it.
3.3. USER INTERFACE

3.3.3 Usability

Gould et al. already recommended three principles of design for an useful and easy to use system in the year 1985. Those principles are still a high standard for usability, even in mobile apps.

- **Early focus on users and tasks:** “First, designers must understand who the users will be. This understanding is arrived at in part by directly studying their cognitive, behavioral, anthropometric, and attitudinal characteristics, and in part by studying the nature of the work expected to be accomplished.” [10]

- **Empirical measurement:** “Second, early in the development process, intended users should actually use simulations and prototypes to carry out real work, and their performance and reactions should be observed, recorded, and analyzed.” [10]

- **Iterative design:** “Third, when problems are found in user testing, as they will be, they must be fixed. This means design must be iterative: There must be a cycle of design, test and measure, and redesign, repeated as often as necessary.” [10]

The development of CStress has been focused as much as possible on satisfying those principles. As the app has been developed for a scientific research project, the target group is well defined. The project focuses especially on students, which are normally in an age between 18 and 30 depending on the degree they are studying for. This age group grew up with digital devices and modern technologies and their expectations in terms of usability and functionality are usually high. CStress has been regularly tested by fellow students and scientific assistants through the implementation process to detect possible weaknesses in design and functionality. When errors in code or in the design appeared, they have been resolved as soon as possible. The app has been tested after incremental changes to ensure that no problem immersed.

3.3.4 How to use CStress

On the first start of CStress, users are encouraged to login to their Fitbit account. This connection is required to use the app. The login directs users to Fitbit’s login page, which asks them for the account email and password. After the successful login, the users are redirected to CStress and are able to use the app (section 3.4.2, p.27). Upon this point, users can download their heart rate data from Fitbit’s Servers and add their personal activities.

CStress’s concept works best when it is used on a daily base. Especially the activities need to be recorded as early as possible, because the self-assessment of the time range of and the feeling through an activity is more precise the earlier it is done. The situation is different with the computed stress, as it is not so important to review it immediately. Comparing the computed stress and the own activities can be done later. However, there is nothing negative about reviewing it regularly to monitor the own progress.

Users have the possibility to turn notifications on and to set the time for it (figure 3.7). Once allowed, the notifications in CStress remember them to update their activities and to download the recent heart rate data. They can also complete their profile (figure 3.7), which adds valuable information like age, gender, body size or weight to the data which is uploaded via the REST-API (section 3.4.4, p.28). Upcoming questions about the app’s functionalities or other topics may be answered in the About section. Users can find the FAQ and links to project Stila’s and Fitbit’s websites there.
3.4 Technologies

3.4.1 Fitbit API

The Fitbit API [7] provides the possibility to access data from the Fitbit fitness trackers. Third party applications can access profile, location, nutrition, activity, sleep and weight data, food logs, and most important for this application, the heart rate data. Fitbit offers two different kinds of heart rate data. There are the heart rate time series, which contain values such as the minimum or maximum heart rate or the resting heart rate of a day [7]. But this application uses the heart rate intraday time series, which return the heart rate values for the requested day and detail level [7]. A request has the following structure:

GET https://api.fitbit.com/1/user/-/activities/heart/date/[date]/1d/[detail-level]/time/[start-time]/[end-time].json

- [date]: the requested date, can be either today or a date in the past
- [detail-level]: number of data points to include, for example 1s or 5s (CStress uses a value of 1s for the detail-level)
- [start-time]: start of the period, format HH:mm
- [end-time]: end of the period, format HH:mm

The response of a Fitbit API request is a JSON object, which in turn contains different datasets as arrays of JSON objects. This inner JSON objects contain the time, for example
00:10:20 and the associated heart rate measure, for example 64. Fitbit wristbands normally record the heart rate in intervals of five or ten seconds, but can be more accurate for sports activities.

### 3.4.2 OAuth 2.0

Getting access to Fitbit’s API requires the usage of the OAuth 2.0 protocol. OAuth in general can be described as an open protocol, which allows the secure authentication to API services from web, mobile and desktop apps [18]. OAuth 2.0 is the next evolution of the OAuth protocol created in the year 2006 and is already the required authentication protocol of many API’s [18].

There are plenty of open source libraries for different programming languages which implement the OAuth 2.0 workflow. CStress uses the Scribe for Java library. It provides all the methods needed to connect to an API service and just needs some customization to get access to Fitbit’s API. There are two possible ways of access through the OAuth 2.0 flow, the Authorization code grant flow and the Implicit grant flow [7]. As CStress uses the Authorization code grant flow, let us see how it works:

- CStress redirects the user to Fitbit’s authorization page with the following request:

```
https://www.fitbit.com/oauth2/authorize?
response_type=code&client_id=XXXXXX&redirect_uri=CStress://fitbit/callback %2Ffitbit_auth&scope=heartrate
```

Figure 3.8: Redirect URL example

The request contains the `client id` (the Fitbit API application id), the `response type` (here: `code` for the authorization code grant flow) and the `scope` (a list of permissions that should be requested). CStress currently just requests the heart rate data.

- Fitbit redirects the user back to CStress via the redirect URL (`CStress://fitbit/callback`) with the authorization code as a URL parameter.

- CStress exchanges the authorization code for an access token and a refresh token. Example request:

```
POST https://api.fitbit.com/oauth2/token
Authorization: Basic Y2xpZW50X2lkOmNsaWVudCBzZWNyZXQ=
Content-Type: application/x-www-form-urlencoded

client_id=XXXXXX&grant_type=authorization_code&redirect_uri=CStress://fitbit/callback %2Ffitbit_auth&code=1234567890
```

Figure 3.9: Token request example

- CStress stores the access token and refresh token in a SQLite database locally on the phone. The access token is used to make requests to the Fitbit API. The refresh token is used to obtain a new access token when the access token expires, without re-prompting the user to login.
To be independent of expired access tokens, CStress obtains a new access token and refresh token with every API request.

### 3.4.3 SQLite

SQLite is a software-library that implements a self-contained, server-less, zero-configuration and transactional SQLite database engine [26]. SQLite is free to use for any purpose. It is the most widely developed database in the world and very popular among applications [26]. It provides a large set of advantages and features to developers:

- SQLite does not need a separate server process [26].
- It reads and writes directly to ordinary disk files [26].
- A complete SQLite database is contained in one single disk file [26].
- The database file format is cross platform [26].

CStress uses several SQLite databases to store the data for the computed stress charts (heart rate, HRV, computed stress), for activities, for the profile and even to manage the access and refresh tokens needed for Fitbit’s API requests. The implementation is based on a database model and a database handler. The model represents a database with its tables and the tables’ columns. The handler manages the creation and upgrade of tables as well as removing them and provides methods for all CRUD (create, read, update, delete) operations.

### 3.4.4 REST API

REST (REpresentational State Transfer) defines a model or an architecture for the communication between or from/to web servers [20]. It is one of the most popular architectures for cloud-based API’s [20]. REST typically uses the Hypertext Transfer Protocol (HTTP). The interactions between clients and servers are limited by a number of verbs (operations), which are GET, POST, PUT and DELETE [20]. Each of the verbs has its own meaning, which in turn produces clarity about the purpose of the request [20]. Requests gain more flexibility by assigning nouns (resources) [20].

In CStress, REST is used to provide server-side access to the users data. These data can be either profile, heart rate or activity data as well as the derived HRV scores and the computed stress. To send the data to the server, CStress implements a HTTP POST request. The request needs nothing more than the servers URL, one or more headers and a body with the data. “Content-Type” is a header and its value defines the kind of resources that are send with the request. In this case the value of Content-Type needs to be “application/json”, as CStress is sending a JSON object (as a string) to the server. The string with the JSON object is send as an entity in the request’s body.

Users of CStress’s beta version can decide whether they want to upload their data to the server or not. As the collected data are very personal and need to be protected from unauthorized access, the server requires a password (“secret”) to allow communication between the client and itself. This secret is attached to the JSON object as a key-value pair.

In future versions of CStress the server upload will, once allowed by the user, work automatically. The current beta version implements the upload process with a button blick in the settings of the app. Users need to enter at least their matriculation number before uploading to the server. This number allows the clear assignment of the users data on the server-side.
3.4. TECHNOLOGIES

3.4.5 Libraries

CStress is using a set of open-source libraries for Android. Those libraries are adding additional functionalities to Android’s software development kit (SDK). Libraries can be integrated into the app in different ways. CStress integrates them in the easiest and most common way by adding them to the dependencies in the build.gradle file. A build represents a version of an app [27]. Each app can have different versions with different configurations (builds). For example, the app can exist in different versions for different devices or API levels. Gradle uses “a specific set of rules to combine settings, code and resources configured in your build types” [27].

As mentioned, libraries are integrated into the app by adding them to the “dependencies” section in the build.gradle file with the “compile” statement. For example, the chart library “MP Android Chart” can be integrated with the following statement:

```
compile 'com.github.PhilJay:MPAndroidChart:v2.2.4'
```

The compile statement instructs the system to integrate the MPAndroidChart module in the version 2.2.4 from github when building the app.

In total, CStress integrates eight open source libraries. The libraries and their functions are listed below.

- **AppCompat**[^5]: This library allows the implementation of Android material design elements on pre-lollipop (Android 4.3 or lower) devices. Android’s version 5.0 features new UI widgets and enables material design. Older platforms are not directly supporting material design, so they need to implement it manually.

- **Chrome Custom Tabs**: There are generally two options for developers to open a URL on a mobile device: They can launch a browser or build their own in-app browser. When launching a browser, the app stops and a context-switch takes place. In addition, the browsers UI cannot be modified and disturbs the users experience with the app. In-app browsers instead can be customized, but “don’t share state with the browser and add maintenance overhead” [14]. Chrome Custom Tabs are providing a solution for this problems. They are customizable and “make transitions between native and web content more seamless” [14]. The toolbar, their functions and enter/exit animations can be customized. Moreover, Chrome Custom Tabs even allow developers to pre-start the browser and to pre-fetch content which produces a better user experience [14].

- **Android Design Support**[^6]: Similar to the AppCompat library, Android Design Support allows to integrate a set of Android material design elements into all devices with Android 2.1 or higher. Such elements in CStress are the navigation drawer, raised and flat buttons.

- **Scribe for Java**[^7]: This library implements the previously presented OAuth 2.0 protocol. Scribe already supports a lot of major OAuth API’s such as Facebook, Google or Twitter out-of-the-box. To connect with Fitbit’s API in CStress, Scribe needs to be customized. Some library methods need an override and the request has to be aligned to Fitbit’s requirements.

- **MPAndroid Chart**[^8]: MPAndroid Chart provides eight different chart types such as

[^5]: URL: [http://android-developers.blogspot.de/2014/10/appcompat-v21-material-design-for-pre.html](http://android-developers.blogspot.de/2014/10/appcompat-v21-material-design-for-pre.html)
[^6]: URL: [http://android-developers.blogspot.de/2015/05/android-design-support-library.html](http://android-developers.blogspot.de/2015/05/android-design-support-library.html)
[^7]: URL: [https://github.com/scribejava/scribejava](https://github.com/scribejava/scribejava)
[^8]: URL: [https://github.com/PhilJay/MPAndroidChart](https://github.com/PhilJay/MPAndroidChart)
line, bar or pie charts. They can be easily implemented and adjusted in their appearance and already support useful features like touch-gestures (pinch-to-zoom), animations or value highlighting.

- **Observable Scroll View**: The library observes scroll events and adds animations to provide feedback for users touch gestures. The scroll view interacts with the toolbar introduced with Android material design and improves the look and feel of the user interface.

- **Nine Old Androids**: Nine Old Androids is just needed for some animations and transitions of the Observable Scroll View.

- **Apache HTTP Client**: Apache’s HTTP client implements the HTTP protocol in Java. In CStress, the protocol is only needed to make POST requests to the server in order to upload the user data. The requests to Fitbit’s API are made through Scribe’s OAuth library.

### 3.5 Second User Study

#### 3.5.1 Methods

CStress has been released as a closed beta version in Google’s Play Store after about two months of implementation work. A second user study has been conducted to evaluate the app and to collect data for further studies such as Patrick Hagen’s aforementioned master thesis (see section 3.2.1.4, p.19). Participants of this study have a couple of tasks to accomplish:

- wearing the Fitbit Charge HR during the examination phase
- using the CStress for Android app on the personal smartphone
- monitoring the daily computed stress levels in the app
- tracking the activities and self-estimations of feelings in the app

After the study they have been asked to complete another post study survey (appendices B.2). The goal of this post-study survey is to evaluate the usability and design of CStress and to proof the users awareness of the computed stress.

#### 3.5.1.1 Participants

Ten participants have joined the second user study, all of them study at the LMU. They have been gathered through an announcement on the projects website and through announcements in lectures. The diversification of genders has been half female and half male. All of the students are studying informatics or media informatics.

---

9 URL: https://github.com/ksoichiro/Android-ObservableScrollView
10 URL: https://github.com/JakeWharton/NineOldAndroids
11 URL: https://hc.apache.org/httpcomponents-client-ga/
12 URL: http://stila.pms.ifi.lmu.de
3.5. SECOND USER STUDY

3.5.1.2 Procedure

The participants have received their Fitbit fitness tracker and the instructions (appendices B.1) at the beginning of the study. Similar to the pilot study, the Fitbit user accounts have already been prepared with anonymous mail addresses. Every participant has installed the Fitbit and CStress app on his or her smartphone and has connected the fitness tracker with the anonymous user account. After this procedure, the participants have connected CStress to their Fitbit account and could start using the app.

The second user study is not only important for this bachelor thesis, it also delivers data for the aforementioned master thesis of Patrick Hagen (see section 3.2.1.4, p.19). For this reason the study is still in progress while this bachelor thesis is already completed. The study participants have been asked to wear the fitness tracker to monitor their stress development as often as possible, especially before, during and after exams. In addition, they have received a survey where they have been asked to self-assess their feelings before and after every exam. The post study survey for the second user study has been conducted after the participants used the app for about two weeks. It contains 20 questions divided into five categories:

- overall satisfaction (1 question)
- usability (7 questions)
- system & performance (4 questions)
- awareness of computed stress (5 questions)
- general feedback (3 questions)

The survey for the second user study has been created with Google forms. It is a similar system as Lime Survey, which has been used for the survey after the pilot study, but looks more modern and user-friendlier. The creation of survey questions is way easier with it. Google forms are build upon Google’s Material Design guidelines, so the look and feel is familiar to the study participants as they have already gotten to know material design on their Android smartphone and in CStress.

Almost all of the survey questions, except the three questions about the participants general feedback, have been asked to be answered on a linear scale from 1 (disagree) to 5 (agree). It is easy to compute the mean figures for all the responses and for every question group with that kind of scale. The final average scores represent the overall level of accomplishment or attitude toward the subject matter. The general feedback questions have been asked to be answered in textual form, because the goal of this questions is to get insights into the participants favorite and non-favorite features and to collect ideas for future improvements of CStress.

3.5.1.3 Data handling

As soon as all participants have responded to the survey, their answers have been analyzed. Google forms provides the download of the results in different formats such as CSV or excel. The excel file has been used to calculate means for the different questions and categories. Excel also has been used to generate a spider diagram (figure 3.10, p.32) to illustrate the results for the categories. The feedback in textual form has been analyzed by summarizing the answers into several categories.

13URL: https://docs.google.com/forms/u/0/
3.5.2 Findings

The goal of the survey was to evaluate CStress in several categories such as the participants overall satisfaction with the app, the usability, the performance of the system, the participants awareness of the computed stress and to receive general feedback.

- **Overall Satisfaction:** The mean for the participants overall satisfaction with CStress is 3.4. This score is especially interesting as it is lower than all of the individual means of the different questions groups. 6 of 10 participants rated CStress with a 3 (neutral), the others rated the app with a 4 (partly satisfied). However, the tendency is more positive than negative, which is proofed by the results of the following categories.

- **Usability:** This category achieves the highest score of all categories with a mean of 4.24. The result absolutely represents the focus on the UI of CStress during the implementation. CStress is especially easy to learn (mean 4.4) and to reuse (4.4) and can be used almost effortless (4.1). The important features in CStress are easily accessible (4.4). The UI is recognizable throughout the whole app (4.2) and characters and texts are well readable (4.3). Only the structure of the presented information could be way better (3.9).

- **Performance & System:** The performance and system category receives an overall score of 3.5. CStress shows some weaknesses in terms of feedback on user actions (3.3) and receives very different ratings at the point “CStress asks for input if needed” (3.2). However, CStress scores better in terms of speed and reliability (3.9) and requires almost the fewest possible steps to accomplish a task (3.6).

- **Awareness of Computed Stress:** This category proofs the participants awareness of the computed stress as proposed in this thesis and evaluates its representation in CStress. The mean for this category is 3.52. Most of the participants understand the term computed stress (3.6) and the graphical representation of the computed stress (the chart) is easy to interpret (4.5) for almost all of them. The chart also fairly reflects
the stress levels of the participants (3.6). Indeed, most of the participants can not easily make a connection between their self-assessed feelings through activities and the computed stress (2.8). Thus, possible stressors in their everyday life’s can not always be found (3.1).

There are three questions in the general feedback category. The study participants have been asked what they like and dislike in CStress and which missing features could improve the app in the future. Half of the participants have mentioned the computed stress chart as one of their favorite feature. The chart is presented in a comprehensible way and easy to interpret. It enables stress tracking, especially through the sleeping hours. The participants also like the simple UI and the good design of CStress (for example the colorful and simple activity icons) as well as the quick synchronization with Fitbit (more precisely the download of the heart rate data).

Those are the favorite features in CStress, but which things do the study participants dislike? Many participants have experienced problems while interacting with the computed stress chart. CStress switches to the add activity screen on a long click which is annoying as it can happen when scrolling through the chart. Other problems with the chart are that there is no scale for the computed stress levels and that the timeline on the x-axis does not show half and full hours as standard. Some participants have experienced app crashes or issues while syncing especially at the beginning of the study, but most of that problems have already been fixed by app updates.

CStress is still in a early stage and can be improved in several areas. These improvements should be aligned to the users needs, so they have been asked which features they would like to see in future versions of the app. The most frequently requested feature is automated activity tracking, for example with activity data (sleeping, workouts) from the fitness tracker. Other common requests are a more powerful chart with integrated activities, feelings and more interactions and the “gamification” of CStress with achievements, more notifications and tips for the users. Participants also would like to reuse custom activities, to track activities easier (for example with a start/stop feature) and to remove the aforementioned problem when long clicking on the chart.
CHAPTER 4

Discussion

This chapter summarizes the key findings of this bachelor thesis, presents limitations and conclusions which have been drawn from the research as well as it gives recommendations for future works.

4.1 Summary

CStress is one of the first apps that explores the possibilities of stress detection with state-of-the-art fitness trackers. Heart rate data recorded by Fitbit’s fitness tracker Charge HR is used to derive the computed stress as presented in this research initiative. The heart rate readings are surely not as accurate as when recorded with clinical devices such as ECG’s, but the pilot study has revealed that they are meaningful enough to approximately derive the HRV. However, it shall be proofed that the inverted HRV score (the computed stress) can also be used as an indicator for stress in everyday life scenarios, as it is normally only ascertained under laboratory conditions or for short time periods.

The conception of CStress is based on two components, namely the computed stress as a representation of the HRV and the self-assessments of CStress’s users. Through the research for this bachelor thesis, the HRV has been found as the ideal indicator for stress. Combining it with the self-assessment of users through their activities has been assumed as the possible key to build a realistic stress detection system.

CStress’s UI has been developed around this idea. The computed stress is presented in minimalistic charts, which leave most of the interpretation up to the users. Users are able to record their activities together with their personal feelings. These are presented together in activity lists right below the computed stress charts, so users are enabled to review their computed and personal stress levels and to recognize relations between them.

A second user study with students of the LMU has been conducted to proof this conception. All of the study participants have received a Fitbit Charge HR fitness tracker to record their heart rates and have installed CStress on their Android smartphone to review their stress development and to record their activities. At the end of the study, they have been asked to complete a survey about their satisfaction with CStress, the usability and per-
formance of the app, their awareness of the computed stress and to gather general feedbacks.

The results of the survey show that CStress is able to provide a good overview over individual stress development and that the UI is well-designed and user-friendly. However, CStress is still in its early stage and has some weaknesses, which need to be overcome. The graphical representation of the computed stress is currently basic and could be more interactive. Future versions of the app may also include gamification features and a easier way to track activities.

4.1.1 Limitations

While working at this bachelor thesis, some limitations in different areas have been found. These are divided into technical, personal and model limitations and presented below.

4.1.1.1 Technical

Initially the two fitness trackers Fitbit Charge HR and Xiaomi Mi Band 1s have been considered for the heart rate measurements. Especially the entry level fitness tracker Mi Band 1s has shown some weaknesses during the pilot study. Its measurements are not accurate enough to be used for the calculations of the computed stress in CStress. Accessing data from the Mi Band 1s is another problem, as Xiaomi does not provide an API for third-party developers yet. The only possibility for accessing this data is to use third-party apps like Mi Band Notify & Fitness, but these third-party apps need to be constantly connected to the fitness tracker to receive heart rate data.

Fitbit’s Charge HR instead offers solutions where the Mi Band 1s weakens. It can automatically record the heart rate and saves the data locally on its chip until it is synchronized with the smartphone or the computer. Fitbit also provides their own API where third-party developers can access the data. However, it requires the explicit approval of Fitbit to receive the heart rate data.

During the second user study, CStress has been used on different Android devices for the first time. Before it has only been tested on a rooted Samsung Galaxy S2 with Android 5.1.1. Some of the participants have experienced bugs with the date and time format in CStress. This bugs only appeared on devices with a system language that is not set to English. The problem has been solved very quickly, but it has revealed one of the greatest challenges in Android programming: the number of different devices. According to Google’s developer console, CStress is possibly working on 6668 Android devices, all with different configurations. Developers are not able to test their apps on all of them, so it is strongly recommended to develop the apps as universal as possible.

4.1.1.2 Personal

The quality of the fitness tracker may be perfect, but an incorrect handling of users can produce irreversible errors in the measurements. One of the most common problems is the right fit of the wristband. If there are different size options available, it is important to choose the right size when buying the fitness tracker. Normally the trackers are also coming with an adjustable strap which enables the users to adjust the wristband into the right position. The heart rate sensors in fitness trackers typically require a tight but still comfortable fit to diminish error rates.

Some of the fitness trackers can also have skin-irritating substances which can lead to
itches or cause inflammations. These can make it impossible to wear a specific fitness tracker. It is recommend to review the substances used in the production of the preferred fitness tracker before buying it.

In order to produce the most accurate results, users are normally asked to provide information such as their age, gender, body weight or body size and their health condition. If these are wrong, the fitness tracker or the app may calculate a wrong step size or an incorrect amount of burned calories. Moreover, without these information the HRV score is almost meaningless, as its interpretation is dependent of this personal factors.

4.1.1.3 Model

CStress is dependent of accurate heart rate measurements by the Fitbit Charge HR. If the measurements are incorrect, the calculations of the HRV score and so the computed stress may be flawed. This errors can only be partly covered by good algorithms.

A common problem among fitness trackers is that most of them are limited to a few specific sensor types. They usually include basic sensors such as accelerometers, which track movements in every direction, and gyroscopes, which measure orientation and rotation. These sensors allow step and activity tracking and a rough estimation of the calories someone has burned. Some fitness trackers also include altimeters to measure how many stairs one has climbed. GPS is only available in more expensive devices or the fitness tracker needs a continuous connection to a smartphone to use a GPS signal.

Not all of the fitness trackers include heart rate sensors or sensors to measure the galvanic skin response (GSR), but those are mandatory for developers to produce more meaningful results. Combining the data from different sensors with good algorithms is the key to success for every stress detection system. CStress is able to remove faulty heart rate records, but can not influence the quality of measurements. This dependency is CStress’s greatest weakness.

It is important to wear the fitness tracker regularly without many breaks, because it helps to create a better overview of any given day and enables users to review the short- and long-term trends in their stress development. Once a user is not wearing it regularly, she is not able to interpret her own stress development correctly.

One of the greatest challenges with smart wearable devices is to track activities. Fitness trackers are able to detect movements like turning around or walking, but they have problems to recognize more specific activities such as eating, cycling, driving the car or many others. Because of this lack, CStress does not try to automatically detect the activities of its users. Instead, users are asked to track activities on their own, but there are some problems which have been found alongside this approach:

- it is a lot of work for users to document all the activities during a day
- users may forget to track activities
- users sometimes don’t remember certain activities

These problems may be solved in future works (4.3, p.38).
4.2 Conclusion

Current fitness trackers are in general relatively accurate in tracking physical activity. They are good in step counting or in estimating burned calories. Some of the fitness trackers, such as the Fitbit Charge HR, even provide the possibility to monitor the heart rate continuously. The research of this bachelor thesis revealed that it is possible to derive the computed stress with heart rate readings from fitness trackers, such as the ones from Fitbit. Furthermore, the computed stress as proposed in this thesis fairly reflects the stress levels of CStress’s users during their activities. Future versions of CStress may be more accurate in their stress measurements and be able to differentiate between eustress and distress with the usage of more sensors.

CStress provides a clean and friendly UI, which leaves most of the interpretation of the computed stress chart up to its users. The feedback of the second user study showed that the design approach is proofed to be appropriate. However, CStress needs to be more interactive and could even include gamification features in the future. The computed stress chart shall display more information such as the activities and the users self-assessments and provide more possibilities of interaction.

One of the greatest challenges in CStress and with fitness trackers in general is activity tracking. CStress implements activity tracking in a sophisticated way. Users need to manually enter their activities, the time range and their feelings. Future versions of CStress instead shall automatically track activities such as sleeping (for example with data from the fitness tracker) or suggest activities depending on the users movements, their physical reactions and time patterns.

4.3 Future Work

There are many possibilities to extend CStress in the future. As already mentioned in the limitations (section 4.1.1, p.36), automated activity tracking is still the most difficult task for fitness trackers and their inherent apps. Future versions of CStress may put a focus on accomplishing this task. The usage of devices with better and more sensors could help to achieve this goal, but this is not and should not be the only solution. Software-side solutions like more intelligent algorithms (machine learning!) and simplified UI’s may be the better way.

For example, users could receive activity recommendations on their smartphones. They could be asked to confirm or to deny the recommendation, which would help the system to adapt to its users and to learn from them. An exemplary situation would be a football player who has his practice every Tuesday from 7pm to 9pm. A “silly” system would recognize this as a walking or running activity. But if the system asks the football player for his actual activity and recognizes the weekly pattern, it could automatically make suggestions in the future.

Another possible improvement for CStress would be the integration of more and different devices from different manufacturers. CStress is yet only working with Fitbit’s devices, but there are many other fitness trackers with heart rate sensors on the market. Examples are the lately released Samsung Gear Fit 2\(^1\), Pebble’s Watch 2\(^2\), the Jawbone UP 3\(^3\) (which is yet

\(^1\)URL: http://www.samsung.com/global/galaxy/gear-fit2/
\(^2\)URL: pebblewatch2
\(^3\)URL: https://jawbone.com/up
4.3. FUTURE WORK

not able to monitor the heart rate continuously) or Garmin’s Vivo-Fitness lineup\textsuperscript{4}. Some of the latest trackers, for example Jawbone’s UP 3 and the Microsoft Band \textsuperscript{5}, are also capable of measuring the skin conductance. The next years could bring more fitness trackers with even more new sensors onto the market.

In the area of stress detection, CStress could be enabled to automatically detect positive and negative stress patterns. The aforementioned master thesis of Patrick Hagen is exactly working into this direction. His assumptions, if validated through the results of his thesis, would improve the stress detection in CStress dramatically. Users would be able to precisely review their bodily stress reactions and the system could automatically relate the extended computed stress to their self-assessments.

This extended stress detection system could also help to improve teaching at universities. The learning platform Backstage has already been introduced in the motivation (1.1, p.1) of this thesis. Lecturers want to be able to receive information about their students stress levels during exams, lectures or practical lessons. With the decreasing prices of fitness trackers and a reliable stress detection system, the lecturers could be enabled to adjust their teaching to the students stress levels.

\textsuperscript{4}URL: https://explore.garmin.com/en-US/vivo-fitness/
\textsuperscript{5}URL: https://www.microsoft.com/microsoft-band/en-us
List of Figures

1.1 The Fitbit Charge HR and the Xiaomi Mi Band 1s ............................................. 3
2.1 Example of the form for the pilot study ................................................................. 9
2.2 The Mockup of CStress’s UI .................................................................................. 15
3.1 Download of Fitbit data ......................................................................................... 19
3.2 The Computed Stress Chart .................................................................................. 20
3.3 Adding new activities to the activity list ............................................................... 21
3.4 Upload the user data to the server ......................................................................... 22
3.5 Classic black and white design with turquoise shades of green and muted shades of red. Different text sizes and styles ....................................................... 23
3.6 Colorful but clean icons and parallax animation ................................................... 24
3.7 The profile and the settings .................................................................................... 26
3.8 Redirect URL example ............................................................................................ 27
3.9 Token request example .......................................................................................... 27
3.10 Results of the post study survey .......................................................................... 32
List of Tables

2.1 Comparison between the fitness trackers ........................................ 8
Bibliography


on Personal Data Meets Distributed Multimedia (New York, NY, USA), PDM ’13, ACM, 2013, pp. 31–34.


Appendices

A Pilot Study
A.1 Task sheet

Preliminary study for the bachelor thesis of Marcel Heil

INTRODUCTION

This preliminary study has the goal to gather the activity data of a user, for example monitoring the heart-rate, counting the steps or tracking the sleep behavior. Collecting this data may help us to find stress-relevant signs or other signals, focused especially on learning activities.

YOUR TASK

The only two things you need to do during this study are first, wearing the given activity tracker, a FitBit charge HR band or the alternative Xiaomi Mi Band 1s, for five to seven days (and possibly nights) and second, fill the form you receive with this paper every day.

To monitor yourself you may use the FitBit Dashboard on www.fitbit.com or for the Xiaomi Mi Band, the Mi Fit and Mi Band Notify & Fitness (Pro Version) on Android. You'll need to log-in with an user account on both platforms. You'll receive a personal log-in with this study, which will also be used anonymously for our data analysis later.

USER ACCOUNT

Use this account data to log-in to the FitBit Dashboard on www.fitbit.com or to the Mi Fit Application / Mi Band Notify & Fitness Application on Android:

Participant: Max Power
User No.: XXXX
User Name: User Two
Email: XXXXXXXX@gmail.com
Password: XXXXXXXX

Please note that this informations shouldn't be communicated to other people.

For any help or issue please contact me under my email: marcel.heil@campus.lmu.de

TABLE EXAMPLE

Before starting with the form, write down your name and the current date on top of the table.

The table has a column for every hour of day, called Timeframe. In the second row you should specify the main activity you've been doing during the given timeframe. Main activity means that you spend at least 50% of the time in an hour with this activity.

We would be pleased if you give us feedback on your stress level during an activity, simply write 0 if you weren't stressed, write 1 if you were moderately stressed and 2 if you were very stressed during an activity.

Bachelor Thesis of Marcel Heil

LMU München - 2016
Please note: Stress can be either positive or negative, for example stress during a run can be positive, stress during the working hours can be negative. You don’t need to specify this, but you may pay attention to this when filling the form.

There’s also a comment section where you can write down any comment you have, for especially an explanation why you had stress during an specific activity (this is optional but may help us). Here’s an example how it could look like:

<table>
<thead>
<tr>
<th>Timeframe</th>
<th>16:00 - 17:00</th>
<th>17:00 - 18:00</th>
<th>18:00 - 19:00</th>
<th>19:00 - 20:00</th>
<th>…</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main activity during the given timeframe</td>
<td>Studying scientific papers</td>
<td>Driving the car</td>
<td>Sports (running)</td>
<td>Relaxing on the sofa</td>
<td></td>
</tr>
<tr>
<td>Stressed (0-2)</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Comment</td>
<td>I was interrupted through some messages on Facebook...</td>
<td>traffic jams...</td>
<td>way more positive stress</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**IMPORTANT**

This study collects sensible personal data like heart-rates or sleeping behaviors. It is mandatory to inform you that this data will be used anonymously in the following thesis.

Our university is following the rules of the *Asilomar Convention for Learning Research in Higher Education*, you can find these rules under [www.asilomar-highered.info](http://www.asilomar-highered.info).

If you're interested in the results of this study, please contact me personally under my email above.

Thank you very much for your participation in this study!
<table>
<thead>
<tr>
<th>Timeframe</th>
<th>00:00 - 01:00</th>
<th>01:00 - 02:00</th>
<th>02:00 - 03:00</th>
<th>03:00 - 04:00</th>
<th>04:00 - 05:00</th>
<th>05:00 - 06:00</th>
<th>06:00 - 07:00</th>
<th>07:00 - 08:00</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main activity</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>during the given</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>time frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Stressed (0-2)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Comment</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Timeframe</strong></td>
<td>08:00 - 09:00</td>
<td>09:00 - 10:00</td>
<td>10:00 - 11:00</td>
<td>11:00 - 12:00</td>
<td>12:00 - 13:00</td>
<td>13:00 - 14:00</td>
<td>14:00 - 15:00</td>
<td>15:00 - 16:00</td>
</tr>
<tr>
<td><strong>Main activity</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>during the given</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>time frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Stressed (0-2)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Comment</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Timeframe</strong></td>
<td>16:00 - 17:00</td>
<td>17:00 - 18:00</td>
<td>18:00 - 19:00</td>
<td>19:00 - 20:00</td>
<td>20:00 - 21:00</td>
<td>21:00 - 22:00</td>
<td>22:00 - 23:00</td>
<td>23:00 - 00:00</td>
</tr>
<tr>
<td><strong>Main activity</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>during the given</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>time frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Stressed (0-2)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Comment</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bachelor Thesis of Marcel Heil  
LMU München 2016
Post-study survey for the bachelor thesis of Marcel Heil

The last couple of days you were wearing the activity tracker and filling the form that you received with it. Now, as this part of the study is over, I ask you to reflect about the whole study. For this purpose I would be pleased if you answer me a couple of questions about the study in the following survey.

Hello. Just go through the questions and answer them as accurate as possible. I would be very thankful if you leave some comments.

There are 16 questions in this survey

General Questions

<table>
<thead>
<tr>
<th>1 [1]</th>
<th>Did you experience problems during the setup of the activity tracker? *</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Please choose only one of the following:</td>
</tr>
<tr>
<td></td>
<td>☐ Yes</td>
</tr>
<tr>
<td></td>
<td>☐ No</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2 [1.1]</th>
<th>If you answered &quot;Yes&quot;, which problems did occur?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Please write your answer here:</td>
</tr>
</tbody>
</table>

Please describe.

<table>
<thead>
<tr>
<th>3 [2]</th>
<th>Did you wear the tracker at night during your sleeping hours? *</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Please choose only one of the following:</td>
</tr>
<tr>
<td></td>
<td>☐ Yes</td>
</tr>
<tr>
<td></td>
<td>☐ No</td>
</tr>
</tbody>
</table>
4 [2.1] If you answered "Yes", how did it feel to wear it at night?
Please choose only one of the following:
- It felt comfortable
- I needed some time to get used to it
- It was really uncomfortable

5 [3] How often did you charge the tracker? *
Please choose only one of the following:
- Every day
- Every second day
- Every three or more days

6 [4] Overall, how satisfied from 0 (not satisfied) to 5 (very satisfied) have you been with the activity tracker? *
Please choose only one of the following:
- 0
- 1
- 2
- 3
- 4
- 5

This question is just about the device, there will be an extra question about the application.
7 [4.1] Describe your favorite feature of the activity tracker: *

Please write your answer here:

8 [4.2] Now, describe which feature of the activity tracker which was worst for you: *

Please write your answer here:

9 [5] How satisfied from 0 (not satisfied) to 5 (very satisfied) have you been with the application that synchronizes and visualizes the collected data of your activity tracker? *

Please choose only one of the following:

- 0
- 1
- 2
- 3
- 4
- 5
10 [5.1] Describe your favorite feature of the application: *
Please write your answer here:

11 [5.2] Now, describe the feature of the application which was worst for you: *
Please write your answer here:

12 [6] Did you change any behavior because of the feedback you received from the activity tracker / the application? *
Please choose only one of the following:

☐ Yes
☐ No
13 [6.1] If you answered "Yes", please describe the behavior that you changed:

Please write your answer here:

14 [7] Would you buy this (kind of) activity tracker for yourself? *

Please choose only one of the following:

- Yes
- No

15 [7.1] If you answered "Yes", how much money would you spend for it?

Please choose only one of the following:

- 10-50 €
- 50 - 100 €
- 100 - 200 €
- > 200 €
Comments

Here you can leave any comment you like, for example if you have ideas for improvements of further studies.

16 [8] Here is space for your comments:

Please write your answer here:
Thank you for participating in this study! The results will be essential for my work so I'm really thankful that you took part.

01.01.1970 – 01:00

Submit your survey.
Thank you for completing this survey.
B Second User Study
INTRODUCTION

Stila is a research project of the Institute of Informatics at the Ludwig-Maximilian University in Munich. We aim to use physiological data of the study participants to detect stress development of each individual while doing exams at the end of SS16. The outcome of this study may have beneficial effects to all students at LMU. We really appreciate that you want to help us with your participation!

More information about the project can be found on the project website: http://stila.pms.ifi.lmu.de

TASK & AGREEMENT

For a limited time, depending on your exam dates, you receive a Fitbit Charge HR wristband.

You agree to wear this fitness tracker as often as possible during this period of time (so as to make it possible to gauge the tracking system) and in any case during your exams (so as to perform the study).

Fitbit's tracker will automatically record your heart rate and also analyze your sleep. You will complete an online questionnaire of about 15 questions an hour before an exam and another online questionnaire of about 15 questions an hour after an exam. It will only take you about 10 minutes for each. A reminder to complete the online questionnaire will be sent to you via mail 24 hours before an exam.

In addition, you agree to use the Android application CSstress, which was developed for this research project, regularly to monitor your stress levels and to record your activities and the self-estimations of your feelings. You will receive an additional questionnaire about your experiences with the application during the study.

At the end of this period of time, at a day and time agreed upon with you, you will return the fitness bracelet to us in good condition.

USER ACCOUNT

Use this account (not your own account) details to either login to your Fitbit mobile application or to the Fitbit Dashboard on www.fitbit.com.

Participant: Max Power

User Name: User One
Email: XXXXXXXX@gmail.com
Password: XXXXXXXX

Please note that you are not allowed to communicate account details to other persons. You should not change the email or password of the user account.
IMPORTANT

This study collects sensible personal data such as activities, heart rates or sleeping behaviors. It is mandatory to inform you that these data will be used anonymously through the evaluation.

Our working group is following the rules of the Asilomar Convention for Learning Research in Higher Education, you can find these rules under www.asilomar-highered.info.

For any question, feedback or if you encounter problems with the application or the device, feel free to contact us via mail: stila@pms.ifi.lmu.de

Thank you very much for your participation in this project. We wish you success in your exams!

I agree to the terms and conditions of this project and to return the fitness tracker at the (date) __________________ in good condition.

You can return the fitness tracker at the office of Yingding Wang - yingding.wang@ifi.lmu.de - Room L103 Oettingenstraße 67, or at the office of Martin Josko - josko@lmu.de - Room L111 Oettingenstraße 67. Please contact Yingding and Martin per mail before returning the fitness tracker.

Location, _______________________________ Signature _____________________________

Date

Stila - a research project of the Institute of Informatics at the LMU Munich - July 2016
CStress survey
The survey’s goal is to receive feedback about your satisfaction with CStress, the usability, the performance and stability of the app and your awareness of the term computed stress. In addition, your suggestions for improvements may help us to enhance CStress in upcoming versions.

Your answers will be analyzed and presented in the bachelor thesis "Conception and Implementation of a Mobile Application with Fitness Trackers as Supportive Tools for Computed Stress Detection".

Take your time and reflect about CStress before answering the questions.
This survey will take you about 10 to 15 minutes to complete.

*Satisfaction with CStress
1. How satisfied are you with CStress? *
   Mark only one oval.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
   1  | 2  | 3  | 4  | 5  |
   Not satisfied | | | | | Very satisfied

Usability questions
2. CStress is easy to learn *
   Mark only one oval.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
   1  | 2  | 3  | 4  | 5  |
   Disagree | | | | | Agree

3. I can use CStress effortlessly *
   Mark only one oval.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
   1  | 2  | 3  | 4  | 5  |
   Disagree | | | | | Agree
<table>
<thead>
<tr>
<th>4. I can easily reuse CStress *</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Agree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5. The important features in CStress are easily accessible *</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Agree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. CStress presents the information well structured *</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Agree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7. CStress characters and texts in their form and size are well readable *</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Agree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8. CStress UI has a recognizable style throughout the whole app *</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Agree</td>
</tr>
</tbody>
</table>

**Performance and System**

<table>
<thead>
<tr>
<th>9. CStress is fast and reliable *</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Agree</td>
</tr>
</tbody>
</table>
10. **CStress requires the fewest steps to accomplish a task** *  
   *Mark only one oval.*
   1 2 3 4 5  
   Disagree ☐ ☐ ☐ ☐ ☐ Agree  

11. **CStress provides feedback while interacting** *  
   *Mark only one oval.*
   1 2 3 4 5  
   Disagree ☐ ☐ ☐ ☐ ☐ Agree  

12. **CStress asks for input if needed** *  
   *Mark only one oval.*
   1 2 3 4 5  
   Disagree ☐ ☐ ☐ ☐ ☐ Agree  

### Awareness of computed stress

13. **I understand the term computed stress used in CStress** *  
   *Mark only one oval.*
   1 2 3 4 5  
   Disagree ☐ ☐ ☐ ☐ ☐ Agree  

14. **The graphical representation of the computed stress (the chart) is easy to interpret** *  
   *Mark only one oval.*
   1 2 3 4 5  
   Disagree ☐ ☐ ☐ ☐ ☐ Agree  

15. **The computed stress presented in CStress reflects my own stress levels during a day** *  
   *Mark only one oval.*
   1 2 3 4 5  
   Disagree ☐ ☐ ☐ ☐ ☐ Agree
16. The computed stress presented in CStress helps me to better understand my feelings through activities *
*Mark only one oval.

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>Agree</td>
</tr>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

17. CStress can help me to find possible stressors in my everyday life *
*Mark only one oval.

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>Agree</td>
</tr>
<tr>
<td>Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

General feedback

18. What do you like in CStress? *

...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................

19. What do you dislike in CStress? *

...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................

20. Which missing features could make CStress better in the future? *

...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................
...........................................................................................................................................................................................................................................................................................................................................................................................................

Thank you for participating in this survey!