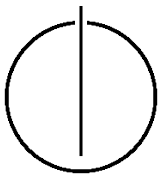
DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

Implementation and Analysis of a Gamification Approach for User Integration into a Mobility Application via Crowdsourcing

Robin Otto



DEPARTMENT OF INFORMATICS

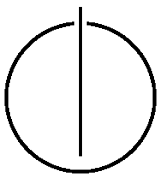
TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

Implementation and Analysis of a Gamification Approach for User Integration into a Mobility Application via Crowdsourcing

Implementierung und Analyse eines Gamification Ansatzes für Benutzer Integration in eine Mobilitäts-App durch Crowdsourcing

Author:	Robin Otto
Supervisor:	Prof. Dr. rer. nat. Florian Matthes
Advisor:	M. Sc. Anne Faber
Submission Date:	15.09.2016



I confirm that this bachelor's thesis in informatics: games engineering is my own work and I have documented all sources and material used.

Munich, 15.09.2016 Robin Otto

Acknowledgments

First of all, I would like to thank Prof. Dr. Florian Matthes for granting me the elaboration of this thesis.

I also would like to thank my advisor Anne Faber for her continous feedback and encouragement throughout the thesis.

A thank-you goes to iteratec GmbH, especially to Michael Westrich and Stefan Rauch, who took the time to support me on all points whenever help was needed.

I would also like to thank Adrian Hernandez-Mendez who supported me by providing multiple ideas and suggestions on different subjects.

Furthermore, a special thanks goes to my colleagues and friends and all other interviewees, who helped me by being available on demand and by giving precious feedback.

Without any of your support, it would have been considerably more difficult to elaborate this thesis.

Abstract

To establish a successful mobility ecosystem and influence users' mobility behavior, it is necessary to provide users with real time data regarding the different transportation modes. One source to gain real time data is by integrating a critical quantity of end users as data providers via crowdsourcing. Thereby the motivation of the crowd to participate is crucial for the success of crowdsourcing initiatives.
The goal of this bachelor's thesis is to motivate end users to provide information about the crowdedness of public transportation in Munich. The collected information should be integrated into further aspects of a mobility ecosystem, in the TUM LLCM context, for added value and should be accessible in real time and for statistical purposes. Existing applications with such features will be analyzed and included in the approach to benefit from successful and unsuccessful applications.
To achieve these goals an application will be implemented, which will be accessible for end users. To motivate the users a gamification aspect will be implemented and evaluated in regard of its success for a continuous involvement of users. The application will be "easy to use", as this is one focus during the implementation.

Contents

1 Introduction

This thesis deals with the implementation and analysis of a gamification approach for user integration into a mobility application via crowdsourcing. This chapter presents the motivation for this work followed by the research questions and objectives arising from the motivation.

1.1 Motivation

According to *World urbanization prospects*, published by the United Nations in 2014 [43], a global urbanization is recognizable. Whereas in 1950 only 30% of the world's population lived in urban areas, the percentage increased to over 50% at the end of 2014. The prospect indicates that it will probably increase even further to about 66% in 2050. It is furthermore stated that already 73% of Europe's population lives in urban areas [43]. Additionally, "as the world continues to urbanize, sustainable development challenges will be increasingly concentrated in cities [...]. Integrated policies to improve the lives of both urban and rural dwellers are needed" [43]. In other means, a solution is needed to cope with the urbanization and the challenges coming along, like managing public transit and living conditions. Specifically the part of solving the public transit problem for Munich, as the city has to cope with urbanization as well, will be regarded in this work. The following statistical data has been extracted from the official evaluation of the statistics agency of Munich [37]. The city of Munich has approximately 1.4 million residents and the provider of public transportation, the Münchner Verkehrsgesellschaft (MVG) [32], offers 21 lines to handle about 555 million commuters annually. That makes an average of 1.5 million people using the local modes of public transportation per day, most of them traveling at business hours. While the capacities of the MVG do not even remotely countervail against the masses of commuters using public transportation on their way to or from work, many trains are hardly used between a certain time interval. The problem coming along with this discrepancy, namely overcrowded trains at 9 a.m. and empty ones after 11 a.m., leads to delays, cancellations and unhappy clients. Therefore a solution is needed to provide the commuters with the information of when it is most intelligent to leave by train in matters of crowdedness and the chance to have a seat.
Especially since the modes of public transportation are not crowded equally over time, it would be interesting to know, how travelers would behave if they would know that the next train going in their dedicated direction is empty. In fact, investigating research papers and studies and participating in a commuter's daily routine have shown that

the information about the commuters' behavior is strongly interesting for companies providing connected mobility platforms in Munich, like Moovel and Moovit but also for MVG itself. Supplementary, it is motivating to implement an approach and to evaluate whether the suggestion of other modes of public transit to the commuters due to overcrowded trains would satisfy their needs. Additionally into consideration comes the migration of the collected information from this work into the context of the TUM Living Lab Connected Mobility (TUM LLCM), a mobility ecosystem. "The research project contributes to the design and implementation of open, provider-independent digital mobility platforms" [41].
Further motivation for this work is the way the desired information is collected. The data about the crowdedness of trains doesn't need to be gathered actively, but is submitted by users in real-time who are sitting in a particular train. In order to realize the submission by users, an application will be implemented combining elements from crowdsourcing and gamification. While crowdsourcing makes data available that couldn't be acquired otherwise, the purpose of the gamification feature is to motivate all users to keep on submitting. By implementing that approach, user specific impressions about certain information can be collected, in this case, the crowdedness of a particular mode of public transit. To ensure that the users are actually going to submit their impressions, a gamification feature is implemented. Additionally if there is the possibility to measure the degree of crowdedness by the users themselves in combination with a validation system for the different votes, evaluation of the collected data is not needed and thereby resource costs are reduced.
In summary, two crucial arguments motivate this work: The evaluation of the possibilities to influence commuters' travel behavior by providing certain information about the crowdedness in public transportation as well as an analysis of the approach to bring users to provide data about how full a certain mode of public transportation is. Although there are already several applications, like *Moovit* and *Waze*, trying to realize such features, a successful approach on providing data about how many people are on a train and making users to submit this very information based on public transportation in Munich is yet to find.

1.2 Derived Research Questions and Objectives

The derived **research questions** are depicted below:

1. *Would commuters change to a less crowded mode of public transport for a few minutes trade-off?*
2. *How would the information about particular train cabins satisfy their needs?*
3. *Is it possible to make commuters provide this exact same information for others inside an application?*
4. *What are the limitations that are involved?*

The **main objectives** that have been carved out from those questions are as follows:

1. *Elaborate the concept of crowdsourcing and gamification considering the state of the art, in order to apply them to the application.*

2. *Develop a mobility application serving people with the information about the crowdedness of a particular mode of public transport and include a combination of crowdsourcing and gamification in order to motivate the users to submit their impression.*

This work is structured referring to the completion of the objectives. For the first objective it is inevitable to further investigate, what the plan of crowdsourcing and gamification is. A detailed definition of the general concepts as well as their utilization and process can be found in section 2.1 and section 2.2. With the two concepts and their meaning in mind, several gamification elements and crowdsourcing approaches are analyzed, evaluated and compared. The crowdsourcing and gamification approaches can be found in chapter 3. For an analysis of the potential gamification elements, see subsection 2.1.4. Subsequent to the outline of the different foundations and the appliance of the user-centered design thinking, chapter 4 describes the final implementation with an approach for a solution concerning the two objectives and all of the problems coming along.
To fulfill the second objective, it is necessary to analyze several competitors and to outline the general concepts of usability and user experience. After figuring out how usability and user experience can improve the application in general, some approaches by competitors originating from the field *connected mobility* are considered. The corresponding parts in this work are section 2.3 and section 3.1. Based on usability and as a foundation for a good user experience, a scheme for user-centered design thinking and its appliance in this work can be found in section 4.1. With the extracted concepts and approaches, it is possible to build a mobility application that provides a good usability in combination with user experience. The implementation is depicted in chapter 4.
In addition to this, chapter 5 contains a conclusion based on the definition of the foundations as well as on the breakdown of the competitors. This chapter also gives an outlook of what could be accomplished to further enhance this work considering the limitations. For a chronological outline of this work, see the structure below.

Foundations

The foundations chapter deals with the basic principles, which are needed for the comprehension of this work would be hardly possible. It deals with gamification in general, how it will be used and which elements could potentially be implemented. Furthermore the concept of crowdsourcing is explained together with an illustration of its process. In addition to crowdsourcing and gamification, this part deals with the basic concept of usability and user experience as well as their dependency.

State of the Art

In order to improve the overall application it is very important to analyze the

different approaches of existing applications. All considered applications originate either from the subject matter connected mobility or productivity and implement important features for this word, for instance *gamification and crowdsourcing*.

Implementation

A scheme for user-centered design thinking is applied as a foundation for user experience. After the conceptual design, which depicts how the expectations and performance requirements are met by this work, a detailed comparison of different hybrid app frameworks as well as mobile backend services is illustrated. Additionally an explanation why which tools and frameworks have been used for this work can be found. The last section deals with the actual documentation of the implementation stating why different elements have been used and how.

Conclusion and Outlook

A general conclusion with reference to the research questions and objectives as well as additional nice-to have features that did not fit in the scope of this work are depicted in this chapter.

2 Foundations

To understand important parts of this work, this abstract deals with the fundamentals used to reach the desired goal. In the first place, a definition of the term gamification in general as well as its main components is provided. Secondly, the concept of crowdsourcing is explained together with its elements. Last but not least the fundamentals of usability and user experience are illustrated.

2.1 Gamification

First of all, it is important to outline the difference between serious games, simulation and gamification. Serious games as well as simulations are games which are not primarily focused on entertaining the player, though they contain elements to do so, by way of example: A racing simulator for Formula 1 drivers. However, gamification is not a stand-alone game itself, but rather describes a process: cutting out pieces from games and apply those elements to non-game environments [24]. Furthermore, it is of importance to distinguish between gamification and games for learning, which represent games with particular learning goals, and game-based learning, comprising all games instilling the process of learning to its users. For a clearer differentiation of gamification see Table 2.1.

	Game	**Serious Game/Simulation**	**Games for Learning (G4L)**	**Game-Based Learning (GBL)**	**Gamification**
Basic Definition	includes all other categories except gamification	game designed for purposes other than pure entertainment	game designed specifically with learning goals in mind	process and practice of learning using games	use of game elements in non-game context
Purpose	any purpose	change in behaviour, attitude, understanding, knowledge	connected with educational goals	Not a game - this is an approach to learning	used to drive motivation, but also to make something more playful

Table 2.1: Classification of gamification based on [3]

2.1.1 General Concept

In general, there are two types of gamification: structural and content gamification [24]. The first one signifies, as the name states, changing the structure around the content by applying "game elements to propel a learner through content with no alteration or changes to the content" [24]. A common implementation would be to take the score elements from games and apply them to the application.
The definition of content gamification is pretty intuitive: add "game elements and game

thinking to alter content to make it more game-like" [24], for instance by adding a story or characters.
These two concepts apply in different ways to the user experience when used. The content gamification will keep the user attracted due to the fact that he or she wants to know how the story evolves, how the characters develop or more generalized: what happens next. Structural gamification affects the application and the user experience in two ways. On the one hand it serves as a fun factor for usually tedious activities and processes, where the user of the application now has a game element or technique which makes it more interesting to finish certain tasks, for example completing TO-DO lists. On the other hand, there is often a competitive element, which engages the users to advance by scoring points, so that they see themselves in front of their colleagues or any other users on the leaderboard. Since it is the easier and faster method, most developers use the structural part as they just have to add the game elements to the structure of the application without the need to alter any functionalities which are already up and running [24]. A *gamified* application can contain either one or both of those two types.
In comparison to apps that use gamification elements, no matter if structural or content gamification, in the current state of non-gamified apps, the user's attitude towards completing tasks is, if existent, rather disinterested. This lack of interest can lead to a loss of quality or quantity of the result of the completed tasks. However, gamification aims at bringing users into a flow and giving them the feeling of ownership while doing these tasks, trying to keep the motivation at a high level. The goal is to provoke a possible change from being demotivated to complete the task to getting the feeling of personal achievement while doing such out-of-favor activities. Several approaches have come up over the last years gamifying different applications. For a comparison and analysis of the state of the art see chapter 3.

2.1.2 Utilization

A state of the industry report by Spil Games [36] highlights why gamification plays an important role. According to this report, more than 1.2 billion people were expected to be playing video games at the end of 2013. With a former world population of about 7.1 billion people, every sixth person in the world was a gamer by that time. The number is expected to be even higher today.
Games succeed to bring the players into a flow between boredom and anxiety through the perfect balance between challenges and skills since the early 1960s and thereby made it to a medium that addresses billions of people in today's life.
This flow is exactly what developers aim at when using gamification. Figure 2.1 depicts how people get into the flow. The expressions *A1 - A4* display the most common appearance in successful games. After starting over with meeting the initially low skills of the player with easy tasks, many games leave the flow for a certain interval. This happens by either assigning the player with challenges that are to difficult, what results in anxiety, or by giving tasks that are not hard enough to meet the player's skills, which results in boredom. The successful games sooner or later manage to get back into the

flow by matching the player's skills with the difficulty of the tasks assigned.
The information about the amount of people having experience with games and furthermore know how it feels to be in the flow, in combination with the knowledge about how gamification aims to bring users in this flow, validates that using the concept of applying game elements in non-game environments is a good approach.

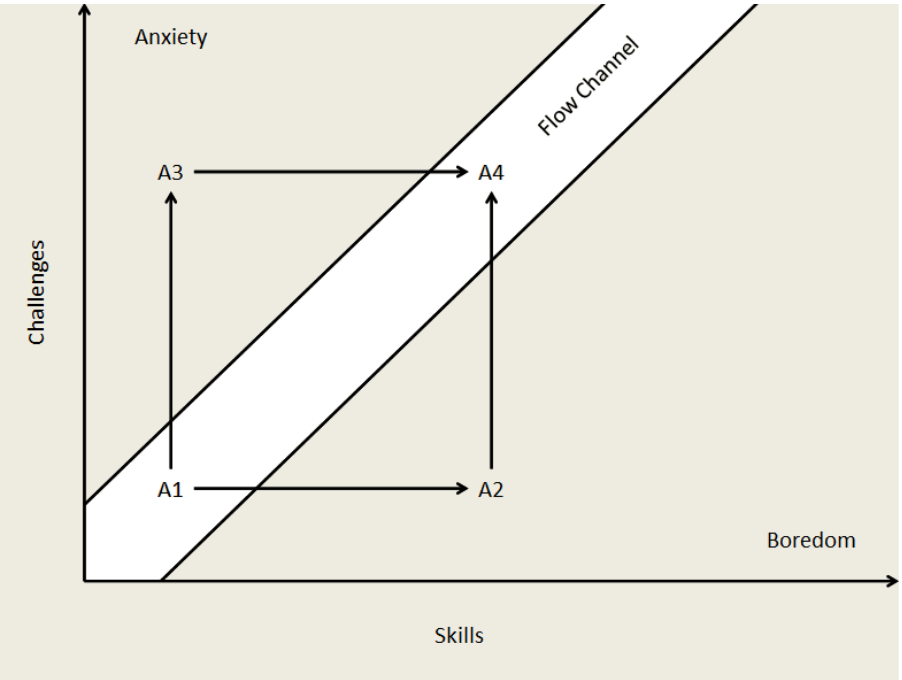


Figure 2.1: Illustration of a gamer's flow based on [8]

2.1.3 Octalysis

Octalysis is a gamification framework created by Yu-Kai Chou [7], the Gamification Guru of the year 2014 and 2015. It is based on eight core drives, which, if applied, introduce different types of motivation to an application. Yu-Kai Chao indicates that every good game has different ways to engage its user, all of them originating from one of those eight core drives [7]. Figure 2.2 illustrates multiple instances of each core drive. The definition of the core drives will be of importance as the analysis of the state of the art in chapter 3 is also accomplished considering the way the competitors make use of gamification.
The eight core drives are:

1. *Epic Meaning & Calling*: represents the feeling of doing something greater than one was supposed to do

2. *Development & Accomplishment*: stands for any form of progress that is made or rather challenges that have to be overcome

3. *Empowerment of Creativity & Feedback*: motivates users to express their creativity & provides feedback for it.

4. *Ownership & Possession*: feeling of owning something, is triggered by advancing to achieve items

5. *Social Influence & Relatedness*: includes all social elements that drive people, normally either social responses or competition.

6. *Scarcity & Impatience*: represents "the drive of wanting something because you can't have it" [7].

7. *Unpredictability & Curiosity*: human desire to investigate processes with unpredictable outcome or events of interest.

8. *Loss & Avoidance*: users try to avoid producing any negative outcome.

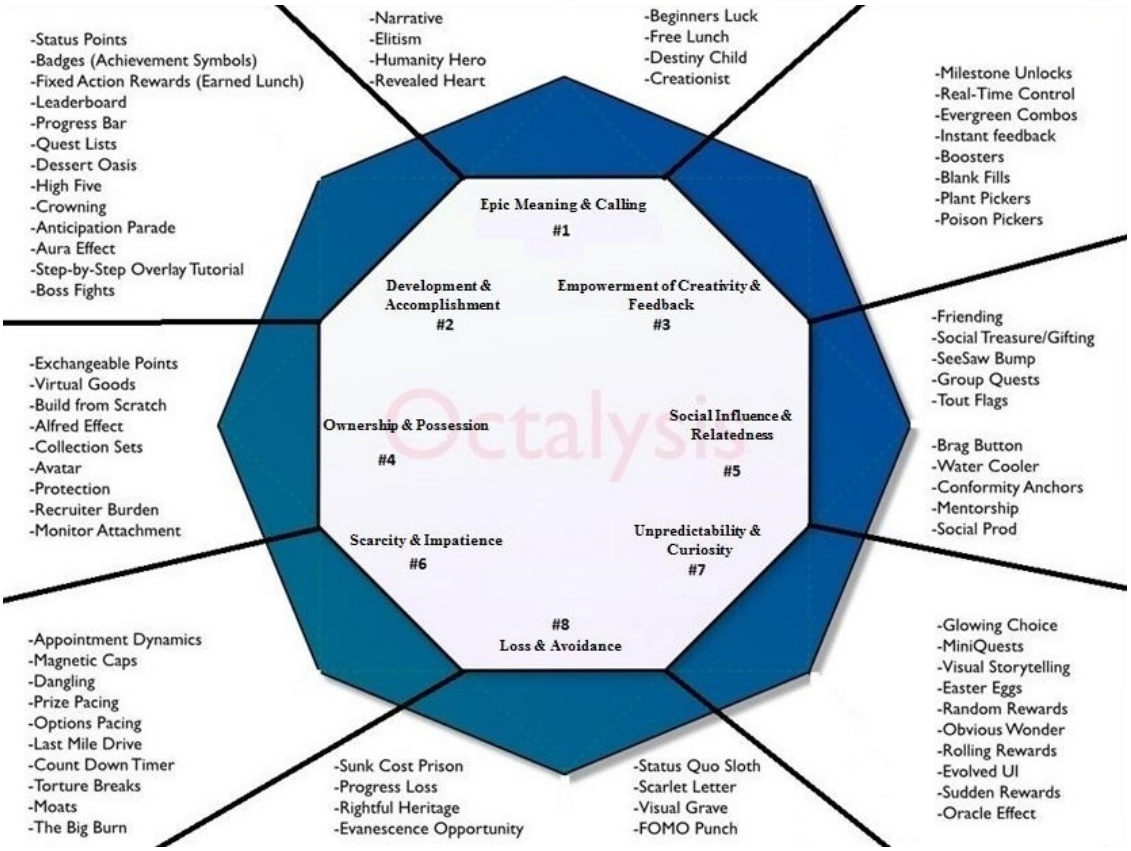


Figure 2.2: The eight different core types based on the gamification framework Octalysis [7]

2.1.4 Potential Gamification Elements

The six most common gamification elements are *Points*, *Levels*, *Challenges*, *Virtual Goods*, *Score Boards* and *Gifting and Charity*. In Table 2.2, based on an article by Barry Kirk [25], these elements are listed and compared according to several attributes applicable to all of them. The criteria used are *reward, status, achievement, self expression, competition* and *altruism* as all of them are "based upon satisfying fundamental human needs" [25].
Some important elements in Table 2.2 are:

- **Points**: reward given to the user for the termination of a particular challenge or task. Through rewarding a user with points a feeling of *status*, *achievment* and *competition* amongst others arises.
- **Virtual goods**: for task completion the user is provided with in-app objects that can be bought for virtual purposes like online communities.
- **Gifting and charity**: offers the opportunity "to send earned gifts to friends or donate them to charity" [25].

Elements	Reward	Status	Achievement	Self Expression	Competition	Altruism
Points	●	●	●		●	●
Levels		●	●		●	
Challenges	●	●	●	●	●	●
Virtual Goods	●	●	●	●	●	
Score Boards		●	●		●	●
Gifting & Charity		●	●		●	●

Table 2.2: Comparison of gamification elements based on [25]

The following interpretation of additional gamification elements is adapted from an article written by Giancarlo Fortino et al [13]. One possibility beside the elements compared by Barry Kirk, would be the utilization of an avatar. An avatar serves as a self-expression of the user and can be styled individually in means of choosing its clothes and its skin or hair. To actually be attractive for users there has to be a reward offering new clothing opportunities, for instance a new hat for submitting new votes. The avatar is reckoned as a user's self-representation inside the application. Through dressing the avatar in *late-game* outfits, one can express his increased higher in-app social status and therefore feels to be reputed to be a more experienced user. A possible realization of an avatar is shown in Figure 2.3.
On top of this, there is the leaderboard element (see *Score Board* in Table 2.2). Every user has an account- or username and is ranked in relation to other users in a table. Ranking can be realized through different attributes, in the majority of cases the unit of measurement is either points or levels while leveling up often depends on the received points. As this implementation suits best the purpose of generating some kind of competition between the users, it is the preferred candidate for the final implementation.

Figure 2.4 illustrates how a potential implementation could look like.
The third alternative, which is used fewer times than the other two and also not included in Kirk's comparison, is to include medals in the application. These medals give the user the opportunity to express their advance in terms of the submission of votes. Furthermore, by having multiple medals, the users are able to define their profile. As shown in Figure 2.5, medals could be realized by using a progress bar in combination with the reward.

Figure 2.3: Avatar [14]

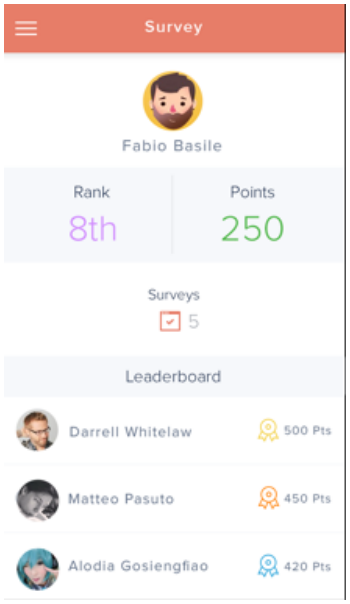


Figure 2.4: Score board [11]

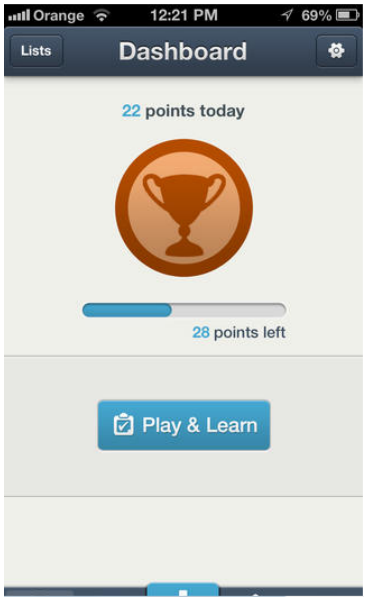


Figure 2.5: Medals [1]

2.2 Crowdsourcing

Crowdsourcing is the second big concept in this work which needs to be defined in order to later apply it. After depicting the general concept, the process of crowdsourcing is described. In addition to this follows a section with an illustration of what the advantages and limitations for this work are.

2.2.1 General Concept

Jeff Howe defines the expression as follows. "Simply defined, crowdsourcing represents the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call. This can take the form of peer-production (when the job is performed collaboratively), but is also often undertaken by sole individuals. The crucial prerequisite is the use of the open call format and the large network of potential laborers" [21]. Thus

it appears that using crowdsourcing can be very powerful in terms of information gain. Due to its potency it is one of the dedicated fundamentals for this work. The expression crowdsourcing itself depicts the process of several anonymous people working together on the web, most commonly to provide information. The so called *crowd* thereby denotes a digital community, which works together to reach certain goals. Crowdsourcing originates from the economical expression outsourcing, meaning to distribute work to other companies mostly in order to reduce costs. The difference lies in the kind of distributing work. While in crowdsourcing, enterprises use the knowledge of private people often without having additional costs, outsourcing is often related to paying a rather high amount of money to the sub-company in order to delegate some tasks, like the production of certain elements. On top of that, crowdsourcing can also lead to an information gain of knowledge that could not be acquired otherwise also because it target group is unproved, in contrary to outsourcing. One example for that would be the collection of data about daily grind activities [22].

2.2.2 Process

Further into detail, crowdsourcing can be divided in its main parts:

- the crowd,
- the initiator and
- the process.

The *crowd* defines who participates, what they have to do and what the participants get in return. The *initiator* defines the person or company who sets the whole process in motion and what his or her dedicated outcome is. The *process* consists of a type, for instance the mapping of urban conditions or transmit data about crowdedness, a type of open call, either open or restricted, and a certain medium, for instance television or World Wide Web [13]. According to Geiger et al. [15] the process of crowdsourcing can be divided into four steps as shown in Figure 2.6:

- pre-selection of contributors,
- accessibility of peer contribution,
- aggregation of contributions and
- renumeration for contributions.

Fortino et al. [13] interpret these steps as follows. First of all, the enterprise has to do a preselection of potential contributors, who will take part in the process of crowdsourcing. Furthermore it is necessary to define how peer contributions can be accessed. After the receipt of the contributions they have to be merged in order to create a final result. In the end, the participants often receive some kind of compensation for their interest.

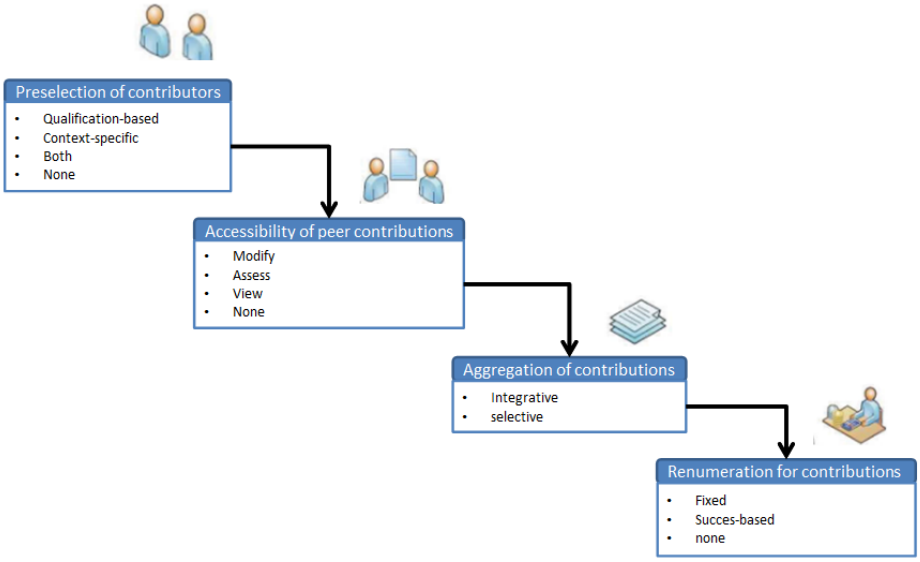


Figure 2.6: Crowdsourcing process based on [13]

2.2.3 Advantages and Limitations For This Work

Jeff Howe's definition of crowdsourcing depicts what advantages come along when using the concept. With the use of crowdsourcing, there is the opportunity to gain information from a huge knowledge pool without having to pay for it, in comparison to outsourcing. An additional benefit is the receipt of such information in real-time, what exactly satisfies the needs in this case. It is most likely that users actually will use the feature as this implementation offers a win-win situation. On the one hand, the app creators gain access to information they can analyze and evaluate. On the other, there is the crowd that instantly gains access to the degree of crowdedness of their dedicated train.
Although there are multiple approaches to receive information about the degree of crowdedness of a chosen mode of public transportation exclusive of crowdsourcing, for instance by evaluating video recordings, its concept is still the most adequate in this case.
Evaluating video recordings is excluded from any further investigation for mainly two reasons: time and complexity. In order to automatically analyze and collect data extracted from recordings of train stations, some kind of machine learning system would have to be implemented. Additionally, accessing the recordings of a train station requires certain permissions, whereat the process of acquiring such permissions is not easy and there is no guarantee that such rights will be granted.
In a nutshell, the advantages of using the power of the crowd are on the one hand the timeliness of the data, on the other the relief of the available resources. But naturally occurring when talking about the diversity of users, there are several problems that have

to be taken care of. Namely the analysis of the submitted data to determine whether it is fake or not and what to do if no data is available or rather how is it possible to increase the reliability. These two problems are yet to solve and represent the biggest handicaps when using crowdsourcing.

2.3 Usability and User Experience

The following paragraph deals with several aspects of usability and user experience in general in order to later apply these concepts according to the individual needs. Both concepts are important in order to ensure that the gamification feature will succeed. One major focus of a gamification element is to enhance the user experience and to increase performance and productivity of the user. Granted that an application would have a bad user experience, a gamification feature would not turn it into a successful competitor. Thus, before gamification can be used, it has to be ensured that the user experience is already high enough. Therefore usability and user experience are analyzed and concepts coming along, for instance from Travis Lowdermilk's book [27], are applied to this work. Figure 2.7 illustrates a leitmotif of steps that developers have to take to ensure that the resulting application will provide the user with a great user experience.

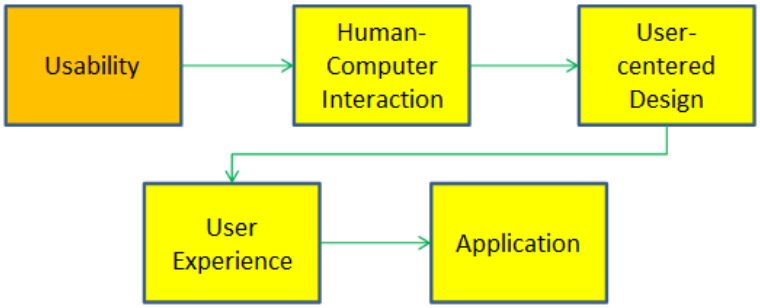


Figure 2.7: Relation between usability, user-centered design and user experience based on [27].

2.3.1 Usability

As shown in Figure 2.7, usability is the first step to take when developing an application. According to Travis Lowdermilk, usability is the cornerstone of application development and it should be taken care right in the beginning of the planning and development process. The International Organization for Standardization defines usability in paragraph §9241 as follows: "Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction

in a specified context of use." [4]. To make this definition more readable it is broken down: When talking about the usability of a product, each one has a context of use. This context of use consists of the following five components as shown in Figure 2.8.

- The *user* is defined as the person, which interacts with the product, for instance the application.
- The *task* is the activity that needs to be accomplished by the user to reach a certain goal.
- The *equipment* represents all parts of hardware and software that the user has in his inventory.
- The *environment* can be split into physical and social surroundings. Physical surroundings could be either dependent from the workplace and the local temperature, whereas social surroundings could represent colleagues or the boss.
- The *product* itself is the part of the equipment for that the usability has to be defined and evaluated.

The interaction resulting from the five components can be measured according to the usability units of measurement. The units are effectiveness, efficiency and satisfaction (see Figure 2.8). According to the International Organization for Standardization, the effectiveness of a product represents the "accuracy and completeness with which users achieve specified goals" [4], in other words:

Is it possible to solve the problem with this product?

Furthermore, the efficiency is defined as "the resources expended in relation to the accuracy and completeness with which users achieve goals" [4], formulated as a question:

How much effort do I have to put into the product in order to solve the task?

The last unit of measurement, satisfaction, stands for the comfort and acceptability of use, which means:

How satisfied am I with the result that I get from the product?

The usability goals for an application are related to the units of measurement as they are rated according to the extent of effectiveness, efficiency and satisfaction with which the goal has been reached. However, the goals are also related to the five components of usability, as these components originate from the aspiration to reach certain goals (see Figure 2.8).
One more characteristic that is required to determine whether an application is useful is the utility. Utility is measured by "whether it provides the features you need" [33] to reach your goal.
Summing up, the composition of a useful application can be illustrated as shown in Figure 2.9.

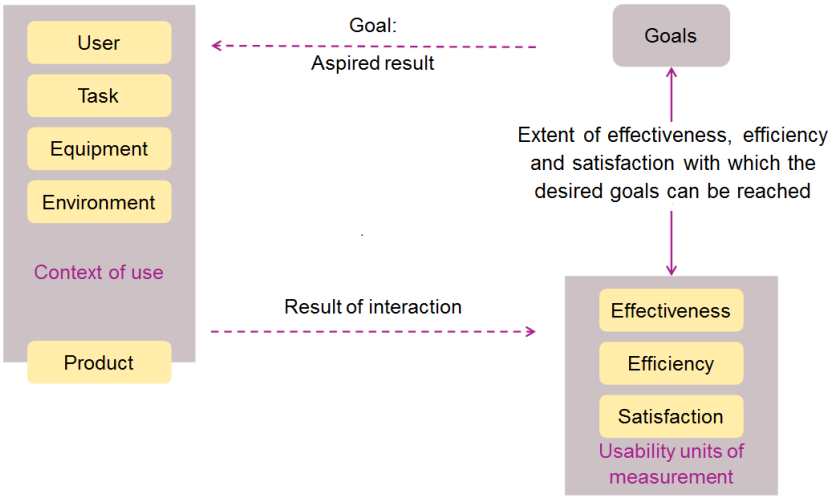


Figure 2.8: The components with impact on the usability of an application, based on [10]

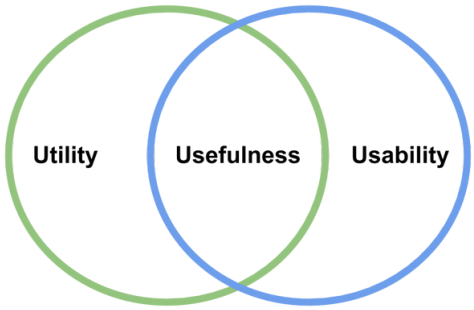


Figure 2.9: Usefulness := Utility + Usability [34]

2.3.2 User Experience

The second important attribute of a successful application is its user experience. Tom Tullis explains user experience as follows: "Usability is usually considered the ability of the user to use the thing to carry out a task successfully, whereas user experience takes a broader view, looking at the individual's entire interaction with the thing, as well as the thoughts, feelings and perception that results from that interaction" [40]. According to Tullis, that means user experience comprises all perceptions and reactions, which:

- the product evokes before actually using it,
- result from the actual or expected use of the product and
- the perceptions and reactions that affect the client after using the product.

Furthermore it embraces feelings, preferences and expectations of the user and is strongly dependent on brand image, representation, functionality, performance and

its interactive behavior, as well as the context of use. As Figure 2.10 illustrates, user experience surrounds amongst others usability but still consists of components which are disjoint from usability, namely the look of the application and the feeling of interacting with the system.

Figure 2.10: The components of user experience [6]

3 State of the Art

Since technologies become more and more diverse and resources are limited, many enterprises outsource multiple tasks, as Juho Hamari tells in his paper about gamification and crowdsourcing [31], "to be carried out by the crowd; a mass of people reachable through the internet". Furthermore, "at least 50% of organizations have gamified some of their processes by 2015" [31]. Concluding from this, there are many applications and other tasks, which combine elements from gamification and crowdsourcing, to be precise, too many to include all of them in this analysis. Therefore the most interesting and suitable applications implementing those fields have been extracted. In section 3.1, three of the existing applications, namely *Moovel*, *Moovit* and *Waze*, will be analyzed precisely. These three originate from the same subject matter: Connected Mobility. This subject matter is also the context of this work and hence makes these competitors specifically interesting. The section 3.2 gives an overview of how the productivity apps *Habit RPG*, *Super Better* and *To-Doist Karma* make use of gamification.

3.1 Connected Mobility

As this work is contemplated to be integrated into the context of a connected mobility platform, applications from this subject matter are analyzed.

3.1.1 Moovel

The first application, Moovel, owned by the Moovel Group GmbH, lets the user choose between many modes of transportation, like bike, car rental, car-sharing, trains, long distance rails, taxi. The reason that Moovel outperforms most of the other connected mobility applications is the fact that the Moovel Group corporates with many service providers out of the different fields of transportation and therefore can make it possible for users to pay all wanted services in-app with one account. Although Moovel does neither use crowdsourcing nor gamification, it implements a simple and intuitive navigation and thereby meets one main focus of the implementation of this work, that is an easy to use application. Therefore it is analyzed and its navigation manner are considered when implementing this work.

Features

The features for the analysis have been taken from Moovel's website [29]. The application provides a simple and intuitive layout with only three tabs containing three features.

First of all, on the start screen (see Figure 3.1) there is a map with the user's current location and a settings button for choosing which modes of transport to display on the map. The user simply enters the preferred destination and then sees a page with all available alternatives from the current location to the destination and a badge with the price for that route (see Figure 3.2). After selecting the optimal route one is prompted with the in-app payment. This is possible due to the fact that Moovel collaborates with many providers of different modes of transportation, most of them offering an interface to their payment system. In addition to this, Moovel offers the registration of many familiar payment methods, including different types of credit cards and PayPal.
The second tab displays a list of booked trips.
Apart from a third tab showing account information, that is the basic functionality (see Figure 3.3). Moovel doesn't provide and furthermore doesn't want to include more features as a quick use is their self-declared must-have [29]. Since Moovel does not only offer services from numerous public transportation providers, but furthermore includes the in-app payment, their application contains all features a connected mobility application needs.

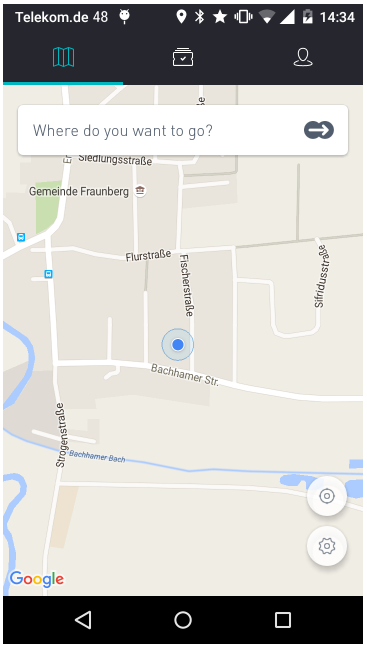


Figure 3.1: Moovel: Start screen

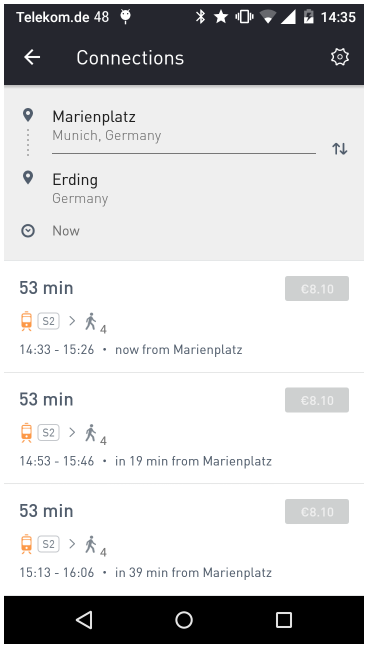


Figure 3.2: Moovel: Connections

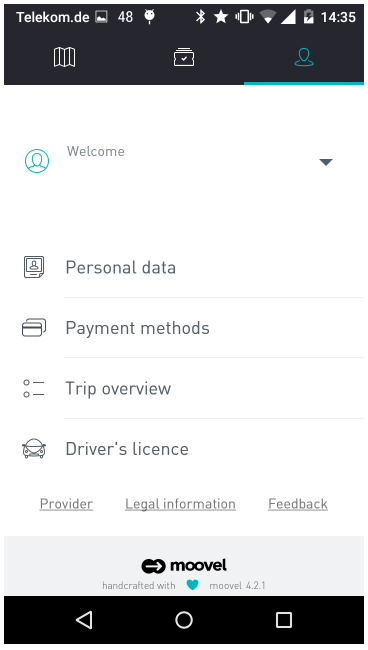


Figure 3.3: Moovel: Account settings

3.1.2 Moovit

The concept of the second application, namely Moovit, is the very opposite of the other candidate: this implementation supplies the user with many features and is thereby more complex. The main business consists of providing their clients with real-time public

transit information and GPS navigation for the different modes of public transportation. Furthermore they offer live community service and advisories. The following analysis of the functionalities is based on the description on Moovit's website [30].

Features

The following analysis of the features is based on the description on Moovit's website [30]. On the start screen the user is directly prompted with five tabs, a search bar, and all available lines leaving from nearby stations. Each tab consists of one core feature.
On the first tab, there is the so called *Omnisearch* and as the name already states, it provides the user with a full search capability, including addresses, stations, places, and the ability to compare routes with detailed informations. By default, the app recommends a preferred line. What distinguishes the search in the Moovit app from the one implemented by Moovel are the available transportation modes. While Moovel includes multiple modes like bikes, taxis and car-sharing, Moovit focuses on public transportation.
Secondly, there is the *Near Me* feature with the possibility to explore what's nearby and also to find the closest stations together with current arrival times for a minimum waiting time.
One of the two core functionalities is included in the *Favorites* tab, where users may enter their preferred addresses for home, work and other locations of interest through which the application automatically creates best routes and generates a push up notification whenever it is the best time to leave for work or home.
The other core feature is contained in the third tab and is called *Live Ride* (see Figure 3.4). Here the user can choose among different lines currently passing the location and set one of those lines as the one the user is currently on. After doing so he receives notifications when to get off the bus or the train and what the estimated time of arrival is. When clicking on the line, a new window opens, where the user is now in charge of reporting several issues concerning his current situation (see Figure 3.5). This contains information about whether his train is crowded, dirty, not on time or even a status report about the station he is waiting at. In addition to this, if somebody reported the station or a particular line, there is the opportunity for other users to view such information. The disadvantage is the misplacement of this feature as there is no obvious indication that another user reported something. In order to access the crowdsourced report, the user has to navigate to an additional screen without being prompted that further information exists.
The forth tab is called *Alert* and contains information about extraordinary events. One example would be a construction site coming along with cancellations or heavy delays of trains.
Last but not least there is the *More* tab, where account information are displayed. Especially interesting is one fragment, which is responsible for the app becoming important for this analysis. The effect of the rewards the user receives when submitting status reports about a certain line or station, a ranking system. The users are categorized

in several ranks dependent on how many points the user already got from reporting information. A beginner starts with the rank *Newbie* and can level up until becoming an *Aviator*, which means the user is now among the best 5% in the country (see Figure 3.6). Superficially examined, this sounds like an awesome gamified feature to motivate the crowd to submit their impression. However, by taking a closer look and investigating the submission, the app not only reveals that there is no check on whether the reporter really is on or close to the line or station and that one can submit multiple votes. Furthermore, after only a few submissions a potential user is already among the top 5% of users in Germany which does not really speak in favor of Moovit's implementation of the gamification approach. Concluding from this, the approach is either not accepted by the users or the user group in Germany is not yet big enough.

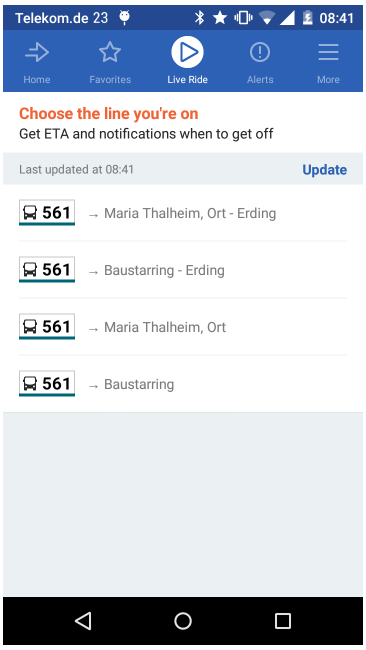


Figure 3.4: Moovit: Live ride

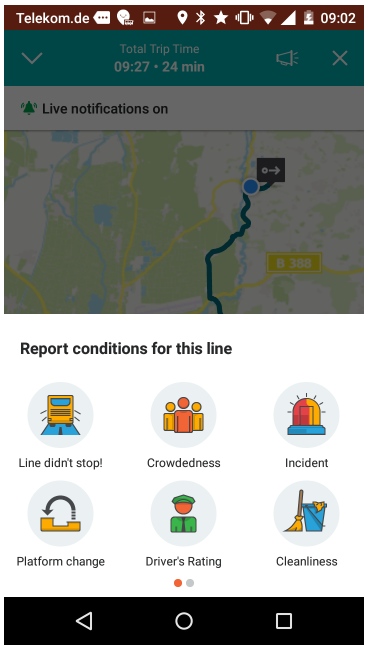


Figure 3.5: Moovit: Report a line

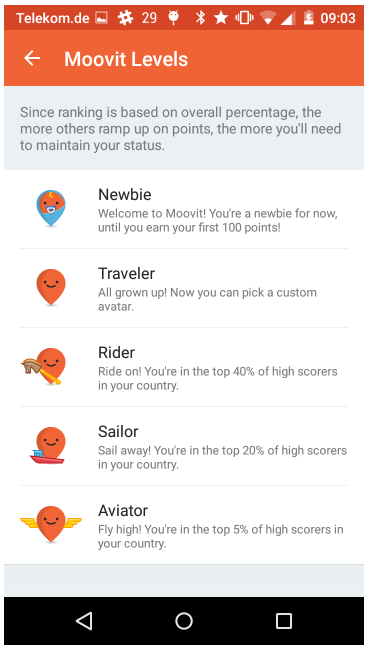


Figure 3.6: Moovit: Ranking system

3.1.3 Waze

The third and last competitor that will be analyzed is Waze. Waze is a turn-by-turn navigational application that uses crowdsourcing to get information in real-time about traffic and road conditions. Though it has been developed by an Israeli developer team called Waze Mobile, since 2013 the application is in possession of Google. The following analysis of Waze is based on *How Waze built its Craze through Gamification* by Christine Yee [44].
The article provides a detailed breakdown of the application "by applying an Octalysis analysis to its design and mechanics" [44]. See Figure 2.2 for a visualization of all eight

core drives of the gamification framework Octalysis. For a recap on the definition of each core drive see subsection 2.1.3.

Features

According to the article, standard GPS systems focus mainly on two core drives. The first one, *Development & Accomplishment* expresses itself inside GPS systems through users that arrive at their chosen destination with a high efficiency. The second core drive is *Loss & Avoidance*. In terms of the standard GPS systems that means that users avoid the negative experience of losing the way by keeping the directions easy.
Waze takes it even further: it introduces a social element to create stationary communities. Although Google owns the rights for the map, the navigation system works based on the data collected through crowdsourcing. The road itself is continuously updated through drivers submitting data as soon as they enter the car. The submitted data can contain information about road conditions, traffic, or police presence. An extract of Waze's features and functionalities is shown in Figure 3.7. The extract contains a screenshot of the typical navigation screen together with Waze's crowdsourcing contribution system.

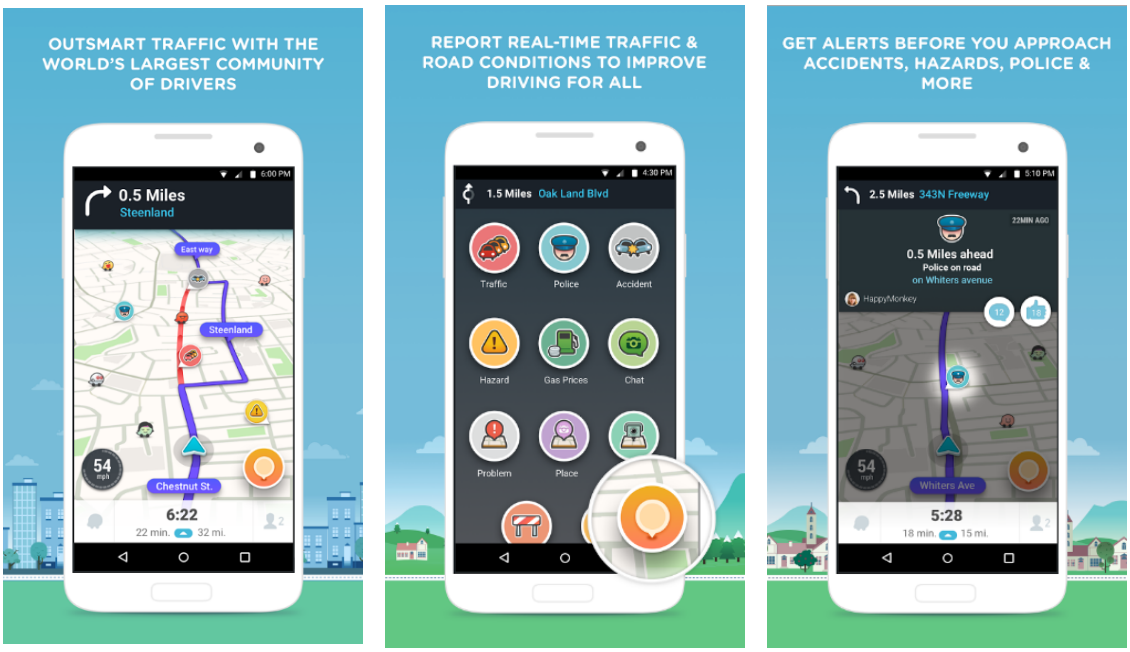


Figure 3.7: Waze navigation and contribution system [42]

According to Christine Yee, Waze's user journey can be divided into different phases: onboarding, scaffolding and endgame phase.
The onboarding phase starts with the first time start of the application. As soon as a user gets to see other contributors and senses their information as useful or as an improvement for the experience, the user encounters core driver number one, *Epic Meaning & Calling*.

By receiving thank notes from other drivers for contributing for the first time, users get in touch with core drive number five: *Social Influence & Relatedness*. Furthermore, all users are rewarded with points for their contribution increasing their motivation.
Following the onboarding phase, the user moves on to the scaffolding phase. This happens once he or she has a basic understanding of how to work with the system. The user is now quite engaged to take the experience even further. By collecting individual data of each client, the application is able to learn unique driving behavior, to understand the user's habits and hence is able to guess the dedicated destination in combination with a recommended route. Waze thereby creates a whole new form of personalized relationship between the app and the client. During the scaffolding phase, another core drive implementation appears: *Unpredictability & Curiosity*. Translated to the Waze app, the core drive manifests itself whenever a rare occasion is contributed by a user, for instance police presence. Once other users recognize this contribution, they are curios to see if it still exists when they get there. Christine Yee furthermore states in her article that a crowdsourcing driven application is not faultless but equalizes all disadvantages through unique experiences created through crowdsourcing.
According to Yee's article, during the third and last stage there are additional core drives that keep the user motivated. The phase is called end game phase and triggers amongst others the *Ownership & Possession* and *Scarcity & Impatience* core drives. All in all, according to Yee's article, the Waze application succeeds in almost all eight core drives except for *Creativity & Feedback*. Accordingly, the application is almost complete in matters of gamification. The confirmation that the developers of Waze Mobile did a great job is given through the massive amount of people using the application as well as through the consistent correctness of the provided routes, even when using it in small cities or villages.

3.2 Productivity Apps

The large variety of applications using gamification in matters of productivity apps motivates to include this subject matter in the state of the art. The analysis of the chosen productivity apps is based on the article *Top Ten Gamified Productivity Apps that will Boost your Life* by Christine Yee [45].

3.2.1 Habitica

The first application is called *Habitica*. The application aims at bringing fun to the users while completing to-do lists. In contrast to common to-do lists, *Habitica* rewards the user with experience points for achieving tasks. Thereby it makes use of the core drive *Ownership & Possession*. Through the proposal of quests and the level-up capability, *Habitica* makes use of core driver number two, *Development & Accomplishment*. Through a punishment for tasks that stay on the list for too long the user's stats decrease. This loss of points or the like represents the core drive *Loss & Avoidance*. In addition to this,

by providing a multiplayer capability to the users, another core drive is used, *Social Influence & Relatedness*. The different categories of *Habitica* are habits, dailies, to-dos and rewards as shown in Figure 3.8.

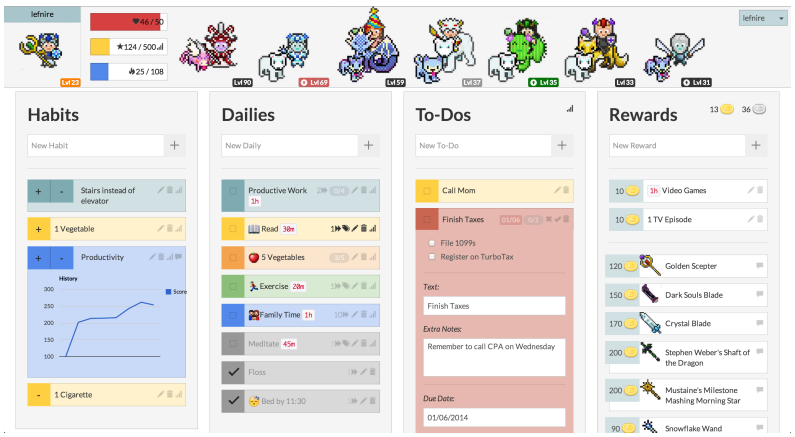


Figure 3.8: Habitica [19]

3.2.2 Super Better

The next competitor, *Super Better*, is also a productivity app but, in comparison to Habitica's to-do lists, aims at winning over bad habits like smoking or unhealthy nutrition. As well as *Habitica*, it provides quests and challenges and thus makes use of the core drive *Development & Accomplishment*. Furthermore, the application offers a individual *Secret Lab* where progress can be tracked, just as how to improve one's performance for certain tasks. This feature is a representation of core drive number three, *Creativity & Feedback*. Figure 3.9 shows how *Super Better* realizes achievements and what the progress tracker for performance improvements looks like.

3.2.3 To-Doist Karma

The last application analyzed in this subject matter, *To-Doist Karma* is rather simple. For completing tasks in time the user is rewarded with good karma and if not the penalty is negative karma. By collecting karma, the user can rank up from beginner over expert to enlightened. Finally the progress and the results are visualized in a graph and thereby giving feedback about how a user performs. By combining these features, *To-Doist Karma* makes use of the core drives *Development & Accomplishment* and *Loss & Avoidance*. A screenshot of the completion history and a karma diagram is shown in Figure 3.10.

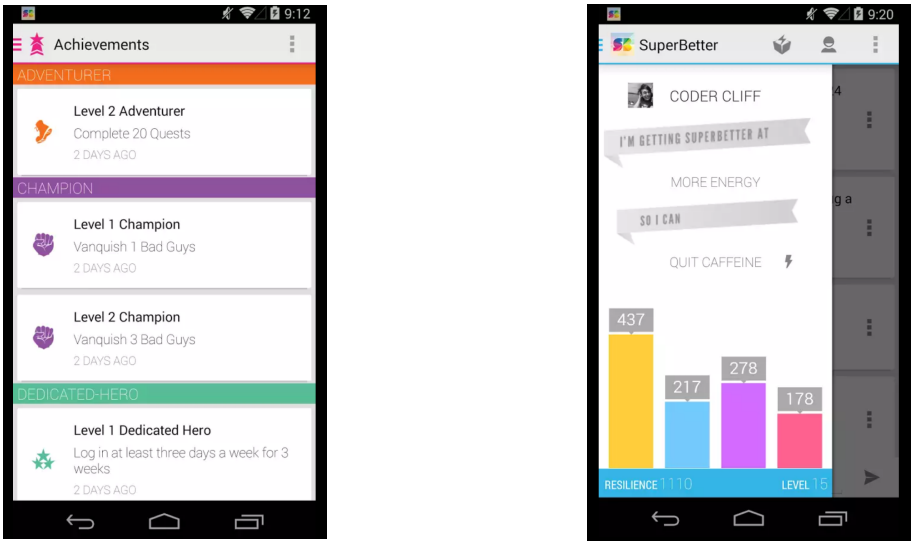


Figure 3.9: Super Better [38]

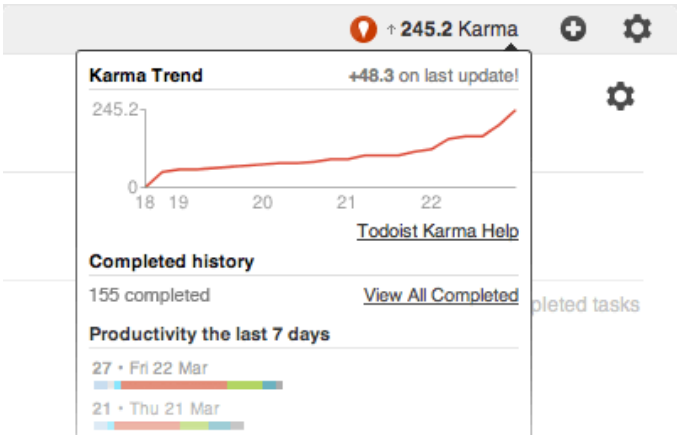


Figure 3.10: To-Doist Karma [9]

3.3 Conclusion

The variety of applications implementing the concepts of crowdsourcing and gamification is huge. Similarly big is the way the concepts are used inside the different applications. As outlined in subsection 2.1.1 two different types of gamification exist, namely structural and content gamification. In the scope of the state of the art analysis of this work, both types can be recognized. As Waze's as well as Moovit's gamification element changes the structure around the original content in terms of rewarding the user for reporting information, structural gamification is implemented. In comparison, content gamification can be identified in the implementation of the competitors from the subject matter productivity. These applications introduce quests and partially even role play games, elements taken from content gamification, to keep the users motivated.

The core features which remain important for this work are the *Near Me* as well as the gamified submissions feature for crowdedness. These features are the desired ones due to the fact that the purpose of the developed app is not to waist the user's motivation on choosing the preferred line but to focus on providing the maximum user experience. Furthermore the focus is to enhance crowdsourcing through the use of gamification elements. Both features are provided by Moovit. It is important to think of an alternative solution compared to what the developers of the related work did, specifically having some kind of verification in mind to check whether the submitted vote is valid.
For a feature comparison of the applications, as well as further, not analyzed applications (shown grayed) see Table 3.1. The applications are rated according to several attributes. Furthermore it is depicted, which core drives were used according to the Octalysis gamification framework.

Competitors	Easy To Use	Crowdsourcing	Gamification	Core Drives
Moovel	●			None
Moovit		●	●	#2, #4
Waze	●	●	●	All except #3
Habitica	●		●	#2, #4, #8
Super Better			●	#2, #3
To-Doist Karma	●		●	#2, #8
Epic Win			●	#1, #2, #5
Chore Wars	●		●	#2, #4, #5
Fitocracy			●	#2, #5, #8

Table 3.1: Feature comparison

4 Implementation

Based on the foundations and considering the outcome resulting from the examination of the state of the art, this chapter deals with the appliance of the acquired knowledge. After applying the concepts theoretically, the implementation of the application is visualized.

4.1 User-centered Design - The Plan

To guarantee that our application will provide users with a great user experience, user-centered design thinking is necessary. Before implementing any user-centered design practices, several aspects have to be elaborated. In his book *User-centered design* [27] Travis Lowdermilk offers a formula, which will be followed to complete this process.
For a visualization of the user-centered design process see Figure 4.1. The user-centered design process is structured into five parts. After having an *innovative idea*, a lot of research is needed in order to *understand* the requirements of potential *users* correctly. Subsequent to collecting user and functional requirements, the developers have to *define the interaction* including use-cases and personas. Additionally, interviews have to be accomplished and first sketches and *prototypes* have to be shown to the clients. Furthermore, before finally beginning the development process, the developers should *design the user interface* together based on more detailed prototypes.
According to the author the first step is to construct a plan including the following sections.

4.1.1 Mission Statement

The point of a mission statement, according to Lowdermilk, is to "remind the team of its purpose" [27]. Where *it* is related to the project a specific team is working on. In other words, by having a mission statement the probability to get lost on particular features is reduced tremendously, since it always reminds the team what the actual goal is. In this work, the mission statement has to contain its *simplicity* as well as its subject matter, *connected mobility*. In addition to this, the mission statement has to outline the proposition of the information about the crowdedness of several public transport modes. Concluding from this, the mission statement for this work could be:

Design a simple and intuitive connected mobility solution providing people with the crowdedness of different modes of public transportation.

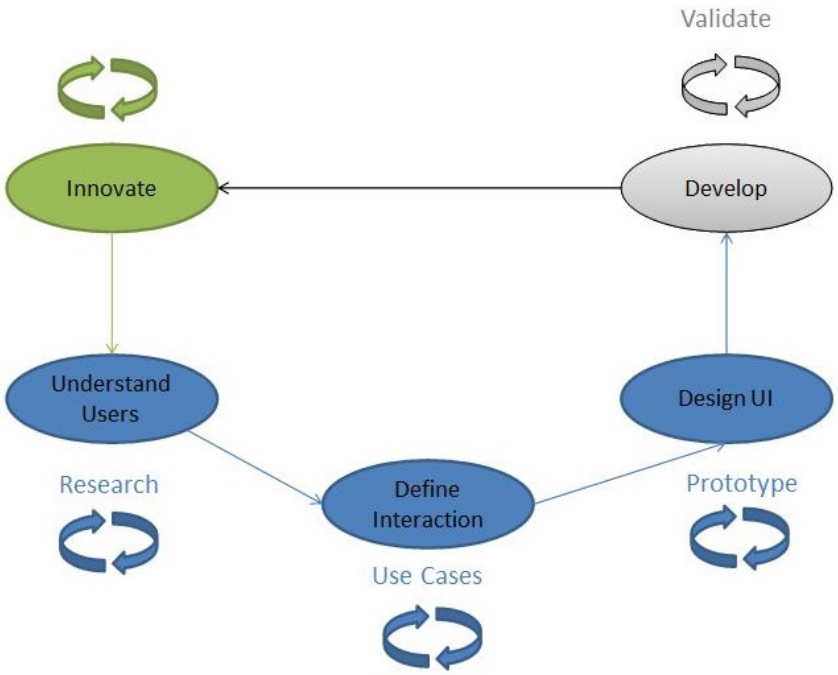


Figure 4.1: User-centered design process based on [20]

4.1.2 Project Details

According to Lowdermilk [27], the project details include:

- an attractive title,
- a brief description,
- the list of stakeholders,
- an impact assessment.

The dedicated title for this work is *CrowdednessIndicator* as it contains information about the application's main focus, the information about the crowdedness of certain modes of public transit. An accurate description could be:

Involve users into submitting data about how crowded their current mode of public transport is and provide those users with a departure board containing this very information using crowdsourcing and gamification.

A possible stakeholder for this application is the Moovel Group GmbH, as a provider of a connected mobility application including public transportation in Munich, but lacking any feature about its crowdedness. Furthermore, a stakeholder for this project could be the TUM LLCM, as the project aims at the realization of a manufacturer independent connected mobility platform [41]. In addition to this, there is the Münchner Verkehrsgesellschaft as the operating company of the public transport in Munich. Last but not

least probably also the city of Munich as one could possibly use the collected data to create statistics, for instance for an annual review.
Concerning the impact assessment there is, on the one hand, one positive effect that comes along with our application, namely that the commuters in Munich gain new knowledge about their chosen mode of public transportation. On the other, there are two negative side-effects that have to be considered, first the fact that fake submissions could initiate a boarding manner, which could lead to empty trains to remain empty, second that the provided data is sensible, but still available to the public.

4.1.3 User Requirements

The scheme in Travis Lowdermilk's book [27] recommends as a next step to define the user requirements for the application. According to Lowdermilk, collecting user requirements builds the basis for all following steps and hence is the most important component. The process of collecting user requirements describes the gathering of needs outlined by potential users and iteratively showing them what the resulting requirements are in order to avoid misunderstandings. Looking at the process of user requirements, the difficult part is to translate the requests, which are often abstract or not suitable, into significant requirements.
Therefore, several interviews with potential users have been accomplished. In the first stage, with the help of initial sketches, the users have been asked what requirements they could probably think of when using the application. Later on, based on the initial sketches, the first prototype has been shown to the users and feedback has been applied accordingly. The results of the first prototype iteration are displayed in Figure 4.2 to Figure 4.4.
Figure 4.2 illustrates the start screen for the first prototype. The user is prompted with a drop-down menu as well as the current location in latitude and longitude coordinates. Beside the opportunity to choose the closest stations around the user through the drop-down menu, it is also possible to change to the leaderboard screen or to open a map window displaying the user's position. After choosing a dedicated line from the scrollable list, the user is directed to the details screen with further information about the estimated departure time and a potential delay. Additionally, the details screen contains a button leading to the feedback screen (Figure 4.4). The user measures the crowdedness of the train according to the provided pictures. After measuring the crowdedness, the user is then directed to the gamification element.
Among many aspects that came up during the interviews and after showing the first prototype, there were some points that appeared multiple times: The argument mentioned most frequently by the potential users was the requirement to use the application while walking. Many stated that having to type text would be a constraint that would make the app unattractive for them. Users want to receive information with no further interaction needed. Furthermore, the interviewees indicated that the application should display the received information in a way that the users can immediately compare the different trains arriving at the station in a certain time interval, especially having in mind the

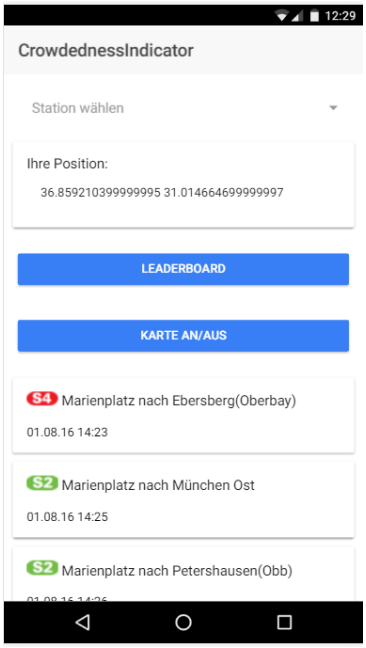


Figure 4.2: Start Screen

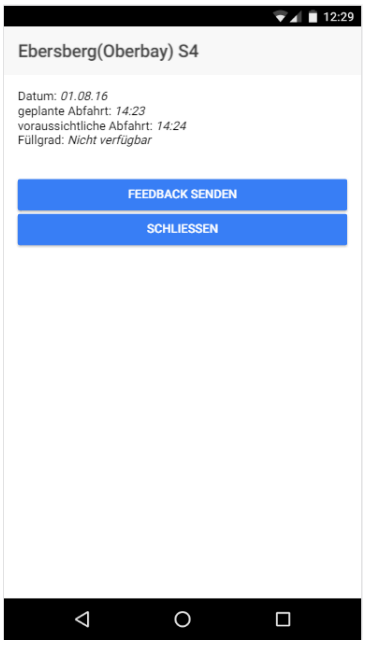


Figure 4.3: Details Screen

Figure 4.4: Feedback Screen

crowdedness attribute of each line. If they would need to focus on the display to extract the desired information, the distraction could be to big while walking down stairs or entering a train station. In addition to this, a few people mentioned the probability to forget to vote as soon as they enter the train and would therefore appreciate a reminder to submit their impression after their train has departed. Furthermore, users stated that one feature would make the application an enhancement of a commuter's everyday life: the possibility to determine which cabins of an incoming train are crowded, in order to being able to take a less packed cabin if needed.

4.1.4 Functional Requirements

After figuring out the user requirements, the next step is to determine the functional requirements, according to Lowdermilk [27]. Functional requirements are, in contrast to user requirements, the technical specification of an application. In other words, how the user requirements are planned to be implemented from a technical view. As the technical solution is not the major focus of this work, the description will be rather brief. The first requirement is to provide the user with a reminder to submit his vote. Whenever the user decides to choose a particular train, he should be alerted that there is the possibility to give feedback concerning the crowdedness of the train by setting the degree on a scale from one to four and pressing the vote button. This is necessary because many users seem to forget to submit their impression as soon as they enter their dedicated train. Therefore a push notification is needed after the user's train of choice departs from his train station. The next point is to determine the nearest train station and

to display the proper departure board. When any user opens up the app there should be no further interaction required to see the incoming trains for the train station where the user is closest to. Since the users are most likely not capable or not willing to type text into a field while using the application, it is more likely to succeed if they only have to open it to see the dedicated information. That is why it is of importance to receive the correct GPS data so that the nearest station is determined correctly. Coming along with the determination of the correct train station, it is further necessary to identify the trains arriving at and leaving from this particular station. For the realization of this feature, mobile data usage is necessary, as a request to the interface offering a departure board for several stations has to be made.

4.1.5 Dataflow and Database Diagram

To be aware of all possible steps required by a user to reach certain goals on their way through our application, Travis Lowdermilk recommends in his plan [27] to create a dataflow as well as a database model.

Dataflow diagram
The dataflow model gives every developer the opportunity to reproduce every possible step to be taken by the client and may uncover mistakable instructions or navigation processes. In this work, the dataflow model is rather simple. After opening the application the first activity is to display the start screen. There it has to be differentiated between users that are close to a train station, in this work that means closer than 500 meters, and users which are too far away. In case a user is close enough when opening the application, the departure board is visualized on the start screen. When the user interacts with one of the lines, the next action is triggered. Details are displayed and the possibility to submit feedback is provided. If the user chooses to submit a vote, a reward is given and the user afterwards gets directed to the leaderboard. Alternatively, if the user is more than 500 meters away from the next train station, an alert arises and the application directs to the leaderboard.
The entire dataflow diagram is depicted in Figure 4.5.

Database diagram
A database diagram is a diagram illustrating on how the data in the project is structured and organized and how it can be accessed. This is especially useful if the structure of the project's database is complex and very large. As in the scope of this work, the use of a mobile backend service together with the hierarchy is flat and easy to understand, the database model is not as useful as it would be in major applications. The data that is actually stored in the backend contains information about lines for which feedback has been submitted as well as the users who submitted this very feedback. For lines, the stored information is composed of the crowdedness, recieved by a user's vote, the

date, its destination and line number as well as departure time from the station the user entered the train. Each line has a unique identifier, which, in combination with the date, makes the stored data about it recognizable and furthermore reusable. Each user element, separated by their mail addresses, contains the date of the last login, points, a username and the number of votes. For the purpose of this work, no additional data needs to be stored, hence the database diagram is omitted.

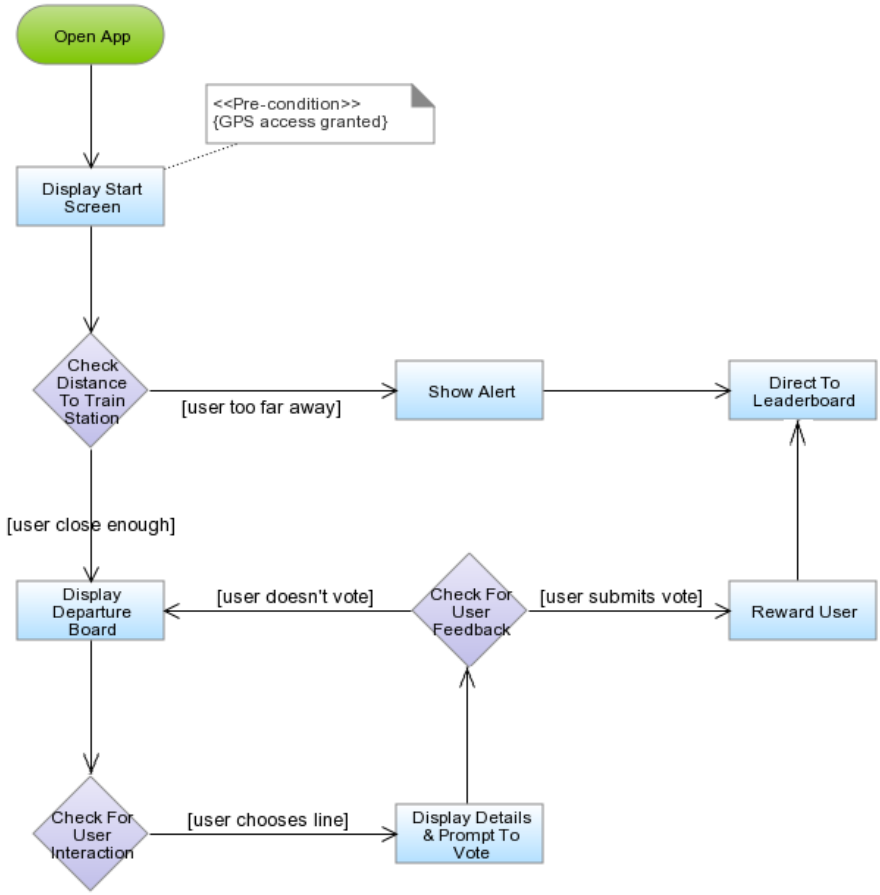


Figure 4.5: Dataflow model

4.1.6 Personas and Scenarios

In order to avoid misinterpretations on who the target group is, Lowdermilk [27] recommends to create personas and scenarios, where the target group participates in living an example situation where they are in need of this work, is necessary. The different personas with their diverse characteristics are depicted in Figure 4.6. Three personas were examined together with an exemplary scenario for each of them.

The first persona, Christian, represents the daily commuter who has a normal work time and therefore is not able to choose at which time he wants to go to work. In his case, the scenario would be that he is interested in which cabin is least crowded in order to maximize his chances to get a seat. That is why the access to this very information would be an advantage for Christian.
The second persona, Lisa, is a around 30 years old and the mother of two children. The scenario of Lisa illustrates a potential user with an urgent need of an empty train. Whenever she uses a mode of public transport, she is on her way to a zoo or the like with her kids. Therefore, she wants the train to be empty enough that she is still able to pay attention to both kids without being distracted. Furthermore, Lisa needs enough space for her baby carriage. To put her plan into practice, Lisa needs the information about the train's crowdedness in advance and hence our application would grant her an additional benefit.
The third and last persona, Maria, depicts a senior citizen at an advanced age, whose capability to walk is already hindered. Maria's scenario is that she needs to take the train from time to time as she does neither have a driver's license nor is any grocery store close enough to her location of residence. Fortunately, she is able to take the train to a grocery store but thus has to know upfront whether all the seats are taken or not due to her walking disability. Our application would provide the needed information to make her shopping routine a much more comfortable trip.

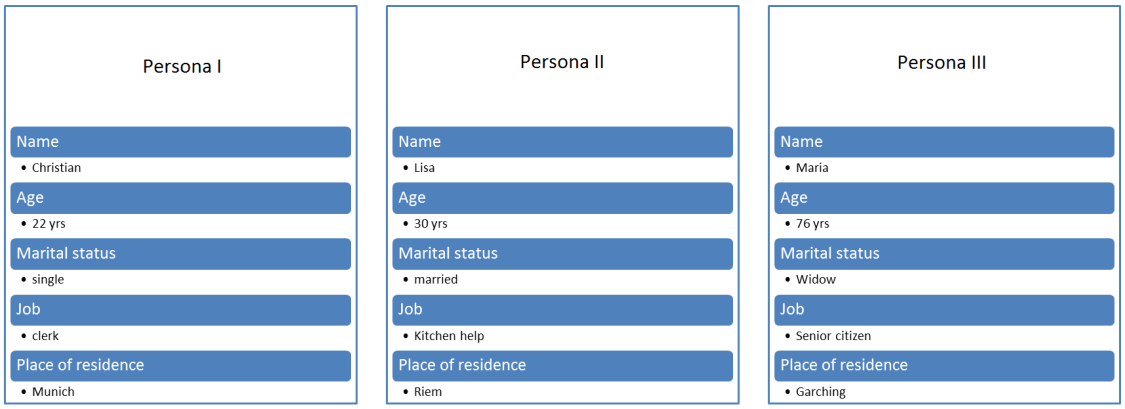


Figure 4.6: Personas for the CrowdednessIndicator

4.2 Conceptual Design

Together with the user-centered design thinking, the conceptual design has to be elaborated in order to gain a clear sight on what shall be achieved and how. The conceptual design is defined as follows: "Description of how a new product will work and meet its performance requirements"[5]. The core points of this definition are:

- functionality of the product,
- performance requirements and
- how the functionalities meet these requirements.

Based on these points, the conceptual design is elaborated.

Functionalities of the product
First of all, in order to ensure proper functioning of the application, it is necessary that the user grants access to the device's GPS, the mobile data usage and the current mail account. These are the requirements for using the application. The App provides the user with a departure board of the nearest train station, determined through the GPS sensor of his or her mobile phone. On this departure board the user can find information about the estimated time of arrival, the delay, the line and the destination of all trains leaving from the current train station. Through an interface to the Münchner Verkehrsgesellschaft (MVG), an up to date, complete and correct time plan is guaranteed. Furhtermore, the application ensures the correctness of the current train station by requesting all nearby stations from a Google Maps application programming interface (API) [17] and using the closest one as the one the user is currently at.

Performance requirements
The first requirement is that the application should provide all functionalities also with low data. Even if the user is inside a building, the current position has to be determined in order to measure the distance to the nearest train station. The application should work on all devices, hence the implemented features may not exceed the capabilities of the low-grad devices.

How the functionalities meet the requirements
As every HTTP call to the different APIs are based upon simple JSON receiving, even users with low connection speed, no matter if temporary or permanent, are able to use the app and see all the data they wish to. On top of all the information provided by different interfaces, our application solves an additional problem many commuters in Munich have to cope with on a daily basis. The aim is to give the commuters the ability to request the crowdedness of a train before entering by simply navigating to the details of a certain line inside our application. Since for the scope of this work it is not possible to access any statistics or evaluations concerning the degree of crowdedness of public transportation, the aim is to motivate the users who want the data, also to share their impression whenever they are using modes of public transportation. The problem coming along with requesting information from the client is the motivational part. In order to encourage the user to submit his or her view about the number of people in a train, a gamification element is introduced, responsible for instilling a feeling of competition and the intention to always make progress so that the user keeps on

using the application. Thereby, the app is able to collect a sufficient amount of data. Furthermore, through including drop downs and scales for voting and selecting, the app stays intuitive and easy to use and thus the necessary interaction is limited on making a few clicks whereas input fields filled with normal text would be inappropriate.

4.3 Technologies

As the specific goals in this work are fixed, technologies that support achieving these have to be found. As the application implemented in the scope of this work should be device independent, a *Hybrid Mobile App Framework* is necessary for development. In order to store the collected data as well as the different users with focus on frontend, the use of a *mobile backend service* is the preferred way. This section deals with the evaluation of front-end and backend technologies in order to determine which suits best for this work.

4.3.1 Hybrid Mobile App Framework

"A hybrid application (hybrid app) is one that combines elements of both native and web applications. Native applications are developed for a specific platform and installed on a computing device. Web applications are generalized for multiple platforms and not installed locally but made available over the Internet through a browser" [35]. With this definition the reason why a hybrid app is the preferential product to create in this work is elaborated.
First of all, during development, there is less limitation on testing the application due to its independence of the device it will later be deployed to. The developer is neither in need of having a real device for each operating system nor is there any necessity of learning multiple programming language. The code for every platform is written in HTML, CSS and Javascript and later compiled to the dedicated platform. Furthermore, since the target group is not yet determined, the developer stays independent and does not have to rewrite any code if the application should only be available on another operating system. The major trade-off coming along with using a framework made for hybrid app development is that there is no possibility to make use of all native features that could be implemented with for example Android Studio or XCode. In addition to this, if it is desired to use the native features provided by Ionic [23], it either doesn't offer the same user experience or it requires a lot of work to accomplish something that can be compared to using an integrated development environment for native apps. See Table 4.1 for a more detailed comparison of the features of native, mobile web and hybrid apps.
The providers of hybrid app frameworks that came into consideration are depicted in Table 4.2.
The Ionic framework version 2 in combination with typescript and Angular 2 is the dedicated development framework in this work. However, Ionic 2 is still in beta phase.

The following abstract is dedicated to outline the different technologies that have been used.
First of all, Ionic 2 provides the easiest and most intuitive way to deploy the source code, which is IOS and Android compatible, on the dedicated platform. All that is needed to do so is to enter the command *ionic run <platform of choice> <device or emulator>*. After that, the build process for the app together with the deployment automatically triggers. When the build has finished, the app is immediately deployed and runs on the dedicated platform or emulator. Similar to this feature, multiple commands come along with the Ionic Framework automating many processes, like page or provider generation, project creation or serving the files on the local machine for testing. Furthermore, Ionic 2 uses known frameworks for the development, namely Angular 2, CSS and their own components for the graphical user interface, built upon regular HTML5. Thanks to the awareness of the programming languages and frameworks through previously implemented web applications and also because of all the available features, a lot of time was saved, otherwise needed for investigating the tools and how they work and instead more resources could be spend on focusing on the actual task. Despite the fact that there are several frameworks that provide similar features as Ionic 2, for instance the Meteor.js framework, which are already released, Ionic 2 is still the favorite for this work.

4.3.2 Mobile Backend As A Service

As a major prerequisite for the reusability of data about the crowdedness and information concerning the gamification feature, we need some kind of persistent storage. Since the focus of this work is not to provide the application with a backend, an alternative to implementing a backend is needed here. The final decision was to use a cloud storage provider, a so called mobile backend as a service. The mobile backend as a service offers logic for saving data, a definition of access rules in order to prevent data abuse and quick and easy integration in any application. These technical specifications fulfill the

Features	**Native Apps**	**Mobile Web Apps**	**Hybrid Apps**
Performance	High	Low	Medium
Offline Mode	Supported	Not Supported	May be supported
Distribution	App Store	Mobile Browser	App Store
Cross Plattform Support	No	Yes	Yes
Device Level Access	High	Low	Medium
User Interface	Good	Average	Good
Development language	Native only	Web only	Native and Web
Development Time	High	Low	Medium
Development Cost	High	Low	Medium
Code Portability	Low	High	High
Manintenance	High	Low	Medium

Table 4.1: Native vs. Mobile Web vs. Hybrid Apps based on [12]

Provider	Advantages	Disadvantages
Ionic 2	Works with predefined components Great community Command Line Interface with lots of useful features	Need to know Angular 2 to do anything complex
Onsen UI	Works with predefined components Excellent documentation with examples	PhoneGap/Cordova builder not included, but supported No support for Material Design (yet)
Framework 7	Simple to use Relies only on HTML, CSS and JavaScript Good performance Combine with any JavaScript framework of choice	PhoneGap/Cordova builder not included, but supported
React Native	Native-like performance Huge community	Steep learning curve Development tools only work on OS X

Table 4.2: Comparison of different Hybrid App Frameworks based on [28]

requirements of the scope of this work and hence a mobile backend as a service is used. This analyzation is based on Ben Teese's article: *Firebase, Meteor & Cognito: Just Enough Database for a Mobile World* [39]. The three biggest providers are Amazon with their product Cognito, the Meteor Development Group with Meteor, and Firebase, which is owned by Google. We evaluated those services respective to the features that are important for our application, see Table 4.3 for reference. One item to be examined is the capability to share data in real-time. Firebase offers an easy and intuitive way to make an additional step: whenever data is shared, the developer has the option to send a push-notification to all clients, so users instantly receive a notification when someone gave feedback about the crowdedness. Concerning storage manners and accessibility of data, Firebase also provides the best solution with the possibility to define a set of rules on who is allowed to access the stored data, as well as the opportunity to customize the structure. The only downside concerning storage manners is that the client always receives the full node when requesting certain attributes. So that requests to the database do not use too much data, it is important to flatten the structure of the data. With this prerequisite in mind, the full node retrieval hardly affects the performance and has little impact on the mobile data usage. For Meteor, the capabilities in the fields accessibility and real-time data are nearly equivalent, plus there is the option to fully configure the server-side. Beside the fact that the official documentation on how to include Meteor in Angular 2 is lacking some important information, creating a big puzzle for people who are trying to complete the tutorial, in the current state Meteor is not compatible with Ionic 2 as Ionic 2 is still in beta phase. Meteor is therefore not considered any further.
The last candidate, Cognito is a web services provided by Amazon and offers direct database access without the need to go through an application layer. Furthermore it is quick and easy to integrate and to access as well and it has a good documentation. Although it has many advantages, Cognito does neither provide an automatic synchronization of data, nor is there an easy way to integrate it with Ionic. In order to ensure that a submitted vote is immediately distributed to all clients, our application relies on that feature. Though the downsides would be tolerable so far, the Amazon web service is not for free and this work is a prototype. Hence it is excluded from further investigation.

Criteria	Firebase	Meteor	Cognito
Real-time data	- share data - push-notifications	- support for live queries	- trigger data synchronization
Storage manner	- always receive full node - minimize depth	- MongoDB instance - key-value storage - application layer	- direct database access - no application layer
Accessibility	- firebase service - define rule language	- client- and server side - flexibility	- offers Amazon Web Services - quick and easy
Support	- good documentation - mail access	- documentation lacks information - application server and MongoDB - learn entire framework	- good documentation - not for free

Table 4.3: Comparison of Mobile Backend Providers based on [39]

4.4 App Development

The new knowledge, acquired by analyzing the concepts of *crowdsourcing*, *gamification* and the *state of the art*, is now applied to the application developed in the scope of this work. First of all, the question was to determine which gamification element should be used. Having in mind the numerous possibilities compared in Table 2.2, one constraint has to be considered, which reduces the variety of gamification elements to choose from. As outlined in the abstract, one focus during the implementation is to keep the application *easy to use* and therefore too complex gamification elements are not considered any further. Concluding for this work, the remaining elements that fulfill the constraint are either a leaderboard, points or levels. In order to satisfy as many criteria as possible a combination of a leaderboard and points is used.
After weighing up the different available hybrid mobile app framework providers and the mobile backend services, having the user requirements in mind and knowing what the dataflow should look like, the implementation can start.
In order to offer the user a native feeling when using the application, only common elements are used for the different parts to ensure that the user is already familiar with the interaction elements.
Furthermore all items that support a functionality when pressed, should be recognized as such. Hence, some design rules according to Hao Loranger's article *Beyond Blue Links: Making Clickable Elements Recognizable* [26] are used. Aiming at avoiding any potential confusion, it is still necessary to prevent to give non-clickable items the look or feel of an interaction element. As the focus during development is to make the application *easy to use*, as few pages as possible are provided in the application.
The start screen displays a departure board of the lines leaving from the train station the user is the closest to. As the main point of the departure board is to compare the incoming trains according to attributes like departure time, destination and degree of crowdedness, each attribute of the lines has to be displayed in a different column.
The most intuitive approach for implementing a comparable list is the table component. Ionic 2 provides a table component out of the box with individual row and column elements. To give the table a native feel, some styling is needed as the initial component

looks makeshift. In addition to this, the table header has to be fixed as the user should always know which attribute he is comparing while scrolling down. The departure board, arranged as a table, is one of the biggest changes resulting from collecting user requirements after showing the first prototype (shown in Figure 4.2). Having a look at the start screen, it does not only display a departure board of the incoming trains, but also contains a tab bar making it possible to change between the start screen and the gamification element, in this case the leaderboard (or *scoreboard*), without further navigation. Another feature on the start screen is the possibility to toggle the visibility of the user's current position on a map with all stations around the user highlighted by a marker. The screenshot of the final start screen is shown in Figure 4.7. Whenever the

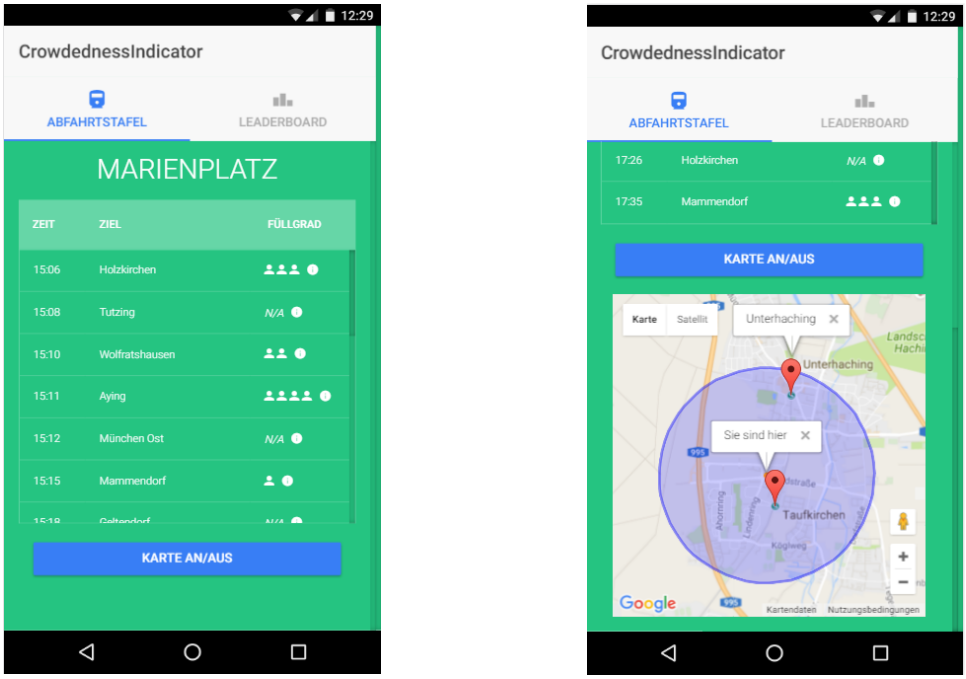


Figure 4.7: Start Screen with map

user chooses the dedicated line, a new screen becomes active including more detailed information about this very line. Among the information about departure time and platform of the preferred line, also the potential delay is displayed, as shown in Figure 4.8. Furthermore, the user is prompted with a group of radio buttons with which the current degree of crowdedness can be determined in case the user is inside this train.
The aim of using radio buttons is to realize information collection through *crowdsourcing*, the first important concept of this work. Looking forward to ensure that the functionality is used, the second big concept is implemented on the details screen.
When the user submits his vote inside the application, a reward with a particular number of points is given, which advances the user on the score board. The underlying concept is *gamification*. By using this reward in means of *gamification*, the user experience of the application is increased significantly. As shown in Figure 4.9, the application prompts

the user with a message stating that a reward has been given and that the user's effort can be seen on the leaderboard page. Furthermore, after submitting a vote, the user is directed to the start screen showing the departure board. The submitted vote is included in the departure board immediately to visualize the effect of a vote so that the user recognizes the changes made through the submission. This decision results from the analysis of the state of the art where other competitors, especially Moovit (subsection 3.1.2), do not show any outcome of a user's report of a line. The user doesn't know whether the vote is available to other users or stored for other purposes.
To navigate to the leaderboard, shown in Figure 4.10, two possibilities exist. The user is either too far away from a train station and hence is directed to the leaderboard or the user clicks on the leaderboard tab in the tab bar of the start screen. The leaderboard displays several information about each user. The attributes are:

- Rank: depicts which place the user takes, measured by the number of points,
- Username: the name of the user,
- Points: the number of points acquired by each user through contributions,
- Votes: the number of votes a user has submitted.

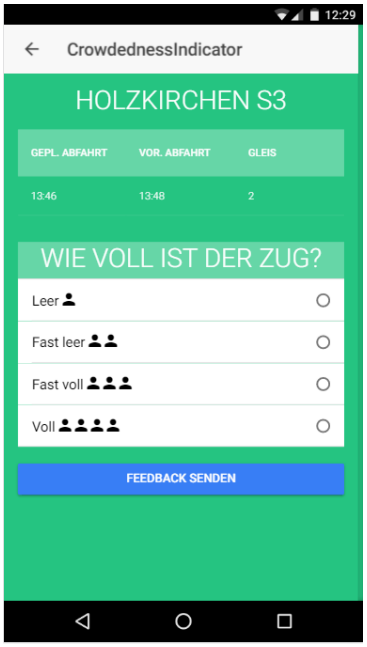


Figure 4.8: Details Screen

Figure 4.9: Reward Toast

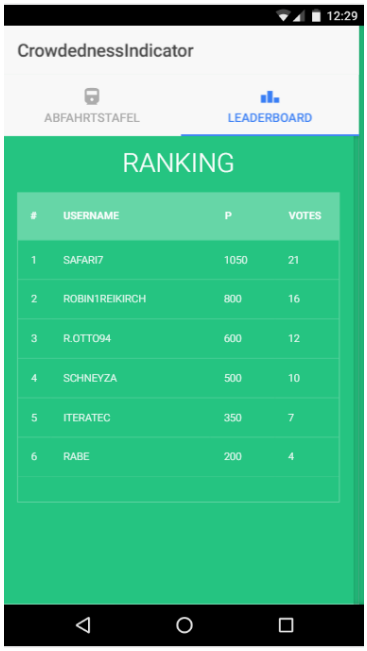


Figure 4.10: Leaderboard

5 Conclusion and Outlook

This chapter provides a closing summary referencing the research questions depicted in section 1.2. Following the conclusion, an outlook is illustrated, including additional nice-to-have features that did not fit in the scope of this work.

5.1 Conclusion

Deducing from the outcome of the implementation and the afterward interrogation of potential users, by elaborating the content of this work it is possible to answer the derived research questions from the introduction.
For the scope of this work, potential users were interrogated by prompting them with the final implementation. The aim of this poll was to receive feedback in order to be able to answer the research questions.

Would commuters change to a less crowded mode of public transport for a few minutes trade-off?
Concluding from the poll, the potential users could imagine to take another train, if it was less crowded. In general, the opinions differed concerning the trade-off the users would have to take. Some stated that they would not be able to wait more than ten minutes for the transfer to their dedicated location, especially on the way to work. Others indicated that they would even accept 15 minutes if a guarantee could be given that they could have a seat. Important for all interviewees was the accuracy of the provided information as well as its timeliness. In general, deducing from the overall feedback, the application would be more useful if it would include the subway, as Munich's infrastructure provides subway transport covering the trunk route at a few minutes interval.

How would the information about particular train cabins satisfy the user's needs?
According to the potential users, this information would be of great value as they would not have to wait for the next train. The problem with offering such kind of information is that it goes to much into detail about a particular line. Hence it is not possible to display this information at the very beginning when opening the application. By not providing this information on the start screen, interviewees stated that they would tend to ignore this information as the receipt would be too time-consuming. They furthermore mentioned that it would be too seducing to fake this information as one could mislead many by doing so. Despite these negative aspects, if it would be possible to display the crowdedness of particular train cabins accurate and reliable, it would be

of high-value to the commuters.

Is it possible to make commuters provide this exact same information for others inside an application?
The aim was to encourage users to contribute information about the crowdedness by introducing the concept of gamification. Whenever a user contributes, a reward is granted. During the poll, the potential users were asked to use the application implemented in this work. The interviewees tried to measure how high the grade of encouragement is through the use of gamification. The feedback differed between the users. Some stated that they were attracted by the reward and could imagine to contribute in order to advance on the leaderboard. Others revealed that they were not as attracted as they would be by another gamification element. In general, the crucial prerequisite for the potential users concerning the intention to contribute was that the underlying system already provides information about the crowdedness. Concluding from this, most of the interviewees would only contribute if there were already some votes as they would see that their submission was valuable.

What are the limitations that are involved?
Another important part of this work was to elaborate the constraints existing when combining crowdsourcing with gamification, specifically considering the inhabitants of Munich as the main target audience. As the exclusive right to an interface providing the details about the public transport infrastructure in Munich belongs to the Münchner Verkehrsgesellschaft (MVG), it was not possible to access this information. In the scope of this work, a makeshift interface has been used, self-made from parsing information from different digital departure boards. The problem coming along with the interface was that it only granted information about the S-Bahn services. Hence, in this work it was not possible to access any other modes.
Another limit that appeared, was the constraint of not being able to test the implementation in production environment. The reason for this was that a rollout did not fit in the scope of this work. That is why potential users have been interviewed and the results from this poll have been applied accordingly.

Summary
In summary, the research questions were answered. Although the level of detail of the results cannot be compared to any outcome from evaluating the feedback of a distribution in the Google Play Store [18] or the Apple Store [2], the answers were accurate enough to draw a deduction. If a partner could be found providing information about multiple modes of public transport, it would make sense to include the functionalities implemented in the scope of this work in the partner's application. As such partner was not available during the elaboration of this work, the evaluation had to be done basing on interviews and a prototype.

5.2 Outlook

Due to the fact that the time to elaborate this work was limited, many features would be nice to have but did not fit in the scope. In the following lines, some of these features are depicted. As shown in Figure 4.7, the train station is determined immediately when the user starts the application. For future changes, it would be preferable to provide a possibility to manually specify the user's current train station. This feature would primarily be of interest whenever the GPS doesn't function normally.
In addition to this, the line chosen by the user should be saved and the application should send a push notification reminding the user to contribute the impression about the crowdedness. As users tend to forget to submit their vote, this improvement could be of great value.
One important add-on for this work is to include multiple modes of public transport. By realizing that, a broader variety of alternatives could be provided. The advantage would be that the chances would increase to find the desired mode. Additionally, if the application would include multiple modes, it would attract more users.
In order to further enhance the user experience, a more complex account registration should be enabled. While in the current state, the username is extracted from the device's active account, the user should be able to specify the username, aiming at an increase of the user's identification with the corresponding account. Coming along with this feature would be the necessity to add a screen displaying details about the account.
Another future task that could also increase the user experience would be the implementation and test of multiple gamification elements in combination with user studies for each element. This could reveal the element with the highest acceptance by the users and hence would maximize chances of the application to succeed. Furthermore it could be imaginable to not only test other gamification elements but to implement a combination of multiple elements as for instance Waze does.
Stated by interviewees, it would be advantageous to have a foundation for the crowdedness of several lines in order to motivate users to further contribute to the system. Therefore alternatives to crowdsourcing as an information source should be considered for the beginning. Possible alternatives would be either to analyze video recordings of particular train stations or to check whether an interface to Google Analytics [16] would provide the desired knowledge.
Last but not least, the application implemented in this work is contemplated to extend an existing system by integrating its functionalities. The application could be integrated in the context of the TUM Living Lab Connected Mobility [41] as the features of the application enhance a connected mobility platform through user integration in the process of collecting information. Generally speaking, the integration in any connected mobility platform lacking a crowdsourcing feature is put on record.

List of Figures

List of Tables

Bibliography

[1] ACAR GmbH. URL: http://www.acar.ch/solutions/Pages/Mobile-App-Quiz-Manager.aspx (visited on 08/30/2016).

[2] Apple Store. URL: http://www.apple.com/de/ (visited on 08/23/2016).

[3] K. Becker. *Games vs Game-based Learning vs Gamification – My Version*. 2015. URL: http://minkhollow.ca/beckerblog/2015/06/21/games-vs-game-based-learning-vs-gamification-my-version/ (visited on 07/05/2016).

[4] N. Bevan, B. Sherwood-Jones, and J. Earthy. *ISO standards for user centered design and the specification of usability - submitted*. URL: http://www.nigelbevan.com/papers/ISO%20standards%20for%20user%20centered%20design%20and%20the%20specification%20of%20usability%20-%20submitted.pdf (visited on 07/01/2016).

[5] Business Dictionary. URL: http://www.businessdictionary.com/definition/conceptual-design.html (visited on 07/11/2016).

[6] K. Butters. *An Introduction To UX Design*. 2016. URL: http://www.springmediadesign.uk/blog/introduction-ux-design/ (visited on 08/11/2016).

[7] Y.-K. Chao. *Octalysis – complete Gamification framework*. 2015. URL: http://yukaichou.com/gamification-examples/octalysis-complete-gamification-framework/ (visited on 07/06/2016).

[8] M. Csikszentmihalyi. *Flow: The psychology of optimal experience*. [Nachdr.] Harper Perennial Modern Classics. New York: Harper and Row, 2009. ISBN: 9780061339202.

[9] To-Doist Karma. URL: http://www.todo-software.de/news/todoist-karma (visited on 08/12/2016).

[10] M. Dörflinger. *IN EN ISO 9241-11 – Anforderungen an die Gebrauchstauglichkeit – Leitsätze*. 2015. URL: http://www.digital-management.at/din-en-iso-9241-11/ (visited on 08/11/2016).

[11] R. Durso. URL: https://dribbble.com/shots/2459527-Leaderboard (visited on 08/30/2016).

[12] eVideoTuition. 2014. URL: http://www.slideshare.net/videotuition/mobile-application-landscape-native-hybrid-mobile-web-apps-introduction-comparison (visited on 07/15/2016).

[13] G. Fortino, W. Shen, J.-P. Barthès, J. Luo, W. Li, S. Ochoa, M.-H. Abel, A. Guerrieri, and M. Ramos, eds. *Designing a Gamification Mechanism to Encourage Contributions in a Crowdsourcing System: May 6-8, 2015, Calabria, Italy*. Piscataway, NJ: IEEE, 2015. ISBN: 9781479920013.

[14] Freeze Pro Software. URL: https://contractiq.com/Entertainment-App-Development-4118-from-Freezepro-software#prettyPhoto/0/ (visited on 08/30/2016).

[15] D. Geiger, M. Rosemann, and E. Fielt. "Crowdsourcing information systems: A systems theory perspective." In: *Proceedings of the 22nd Australasian Conference on Information Systems (ACIS) 2011*. Sydney, Australia, 2011.

[16] Google Analytics. URL: https://www.google.com/intl/de/analytics/ (visited on 08/23/2016).

[17] Google Maps. URL: https://developers.google.com/maps/documentation/javascript/ (visited on 06/25/2016).

[18] Google Play Store. URL: https://play.google.com/store?hl=de (visited on 08/23/2016).

[19] Habitica. URL: https://habitica.com/static/features (visited on 08/12/2016).

[20] F. Hofmann. *„Das ist ja einfach!"* 2014. URL: http://de.blog.cubeware.com/das-ist-ja-einfach/ (visited on 07/15/2016).

[21] J. Howe. *Crowdsourcing: A Definition*. 2006. URL: http://crowdsourcing.typepad.com/cs/2006/06/ (visited on 07/01/2016).

[22] J. Howe. *The Rise Of Crowdsourcing*. 2006. URL: http://www.wired.com/2006/06/crowds/ (visited on 07/01/2016).

[23] Ionic. URL: http://ionicframework.com/ (visited on 08/30/2016).

[24] K. M. Kapp. *The gamification of learning and instruction fieldbook: Ideas into practice*. Ed. by L. Blair and R. Mesch. San Francisco, CA: San Francisco, CA: J. Wiley & Sons, 2014. ISBN: 9781118674437.

[25] B. Kirk. *Gamification: Are we there yet?* 2011. URL: http://www.dmnews.com/digital-marketing/gamification-are-we-there-yet/article/217676/ (visited on 07/08/2016).

[26] H. Loranger. *Beyond Blue Links: Making Clickable Elements Recognizable*. 2015. URL: https://www.nngroup.com/articles/clickable-elements/ (visited on 07/12/2016).

[27] T. Lowdermilk. *User-centered design*. First edition. Beijing and Sebastopol, California: O'Reilly, 2013. ISBN: 978-1-449-35980-5.

[28] D. Markov. *Comparing The Top Frameworks For Building Hybrid Mobile Apps*. 2015. URL: http://tutorialzine.com/2015/10/comparing-the-top-frameworks-for-building-hybrid-mobile-apps/ (visited on 07/01/2016).

[29] Moovel. URL: https://www.moovel.com/de/DE/app (visited on 07/10/2016).

[30] Moovit. URL: http://moovitapp.com/de/ (visited on 07/10/2016).

[31] B. Morschheuser, J. Hamari, and J. Koivisto. "Gamification in Crowdsourcing: A Review." In: *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 4375–4384. DOI: 10.1109/HICSS.2016.543.

[32] Münchner Verkehrsgesellschaft. URL: https://www.mvg.de/ (visited on 08/25/2016).

[33] J. Nielsen. *Usability 101: Introduction to Usability*. 2012. URL: https://www.nngroup.com/articles/usability-101-introduction-to-usability/ (visited on 07/01/2016).

[34] N. Parente de Sousa, A. K. Carvalho, and M. A. Holanda. *Ventilator vs. healthcare professional vs. patient, time to talk!* 2016. URL: https://xlung.net/en/articles/14 (visited on 08/11/2016).

[35] M. Rouse. *hybrid application (hybrid app)*. 2011. URL: http://searchsoftwarequality.techtarget.com/definition/hybrid-application-hybrid-app (visited on 07/01/2016).

[36] spilgames. *State Of Online Gaming Report*. 2013. URL: http://auth-83051f68-ec6c-44e0-afe5-bd8902acff57.cdn.spilcloud.com/v1/archives/1384952861.25_State_of_Gaming_2013_US_FINAL.pdf (visited on 07/04/2016).

[37] Statistisches Amt München. *Die Münchner Verkehrsgesellschaft mbH*. 2015. URL: https://www.muenchen.de/rathaus/Stadtinfos/Statistik/Verkehr.html (visited on 07/01/2016).

[38] Super Better. URL: https://www.superbetter.com/ (visited on 08/12/2016).

[39] B. Teese. *Firebase, Meteor & Cognito: Just Enough Database for a Mobile World*. 2015. URL: https://dzone.com/articles/firebase-meteor-cognito-just (visited on 07/01/2016).

[40] T. Tullis. *Measuring the user experience: Collecting, analyzing, and presenting usability metrics*. Amsterdam [u.a.]: Amsterdam [u.a.]: Elsevier [u.a.], 2008. ISBN: 9780123735584.

[41] TUM Living Lab Connected Mobility. 2016. URL: http://tum-llcm.de/ (visited on 08/24/2016).

[42] Waze Android. URL: https://play.google.com/store/apps/details?id=com.waze&referrer=utm_source%3Dgmm%26utm_campaign%3Dgmm_android (visited on 08/23/2016).

[43] *World urbanization prospects: The 2014 revision : highlights*. New York: United Nations, 2014. ISBN: 978-92-1-151517-6.

[44] C. Yee. *How Waze built its Craze through Gamification*. 2014. URL: http://yukaichou.com/gamification-examples/an-octalysis-look-at-the-waze-craze/ (visited on 07/06/2016).

[45] C. Yee. *Top Ten Gamified Productivity Apps that will Boost your Life*. 2014. URL: http://yukaichou.com/lifestyle-gamification/the-top-ten-gamified-productivity-apps/ (visited on 06/25/2016).