Norwegian University of Science and Technology

Department of Computer and Information Science
Faculty of Information Technology, Mathematics and Electrical Engineering

Autumn Project
TDT4501 Computer Science, Specialization Project

Empirical Study of Timetable Retrieval, and Bus Location Approximation

Authors:
Ole Kristian Nakken,
Sofia Nascimento Bakke

Supervisor:
Rune Sætre

March 16, 2015

NTNU – Trondheim
Norwegian University of
Science and Technology

Abstract

In cold Norway it is not an uncommon occurrence for people to stand at the bus stop for 40 minutes, in the middle of winter, waiting for a bus that is never coming because it is stuck trying to climb an icy hill. Such information would certainly be more helpful than of a simple time estimate that is indicating that the bus is five minutes away, for 40 minutes. The bus service in Trondheim has in recent years become increasingly digitized, with arrival estimation through GPS and mobile ticketing. An open web API allows anyone to gather these estimates, and this has been done by numerous mobile applications. However these applications do not use this data for anything but simply displaying the estimates, while through this paper we propose a proof-of-concept that utilizes these estimates to approximate the locations of buses in Trondheim and displaying these in a map on smart phones.

To establish whether this will help travelers feel less frustrated, wait less, or at least perceive that they wait less, a web survey was conducted together with research on how digitized public transport has been utilized around the world.

Through this survey, it was found that 81 % of users thought that knowing the location of the bus, would help reduce their wait time. Surprisingly the users of mobile applications believe they spend 5 minutes waiting for their bus, while the average user of traditional methods spend only 4.9 minutes waiting. This minuscule difference contradicts some of the previous research done on this field, that found mobile users spent two minutes less than others.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This report describes the work conducted as a preliminary study for Ole Kristian Nakken and Sofia Nascimento Bakke's master thesis. The paper includes theory and background of the project, as well as testing of technologies through prototyping. During the project a literature study and a user survey was conducted. To test whether the idea for a mobile real-time bus map could work in practice, technologies were tested, and a demo application created for an Android phone.

1.1 Motivation

A lot of time and effort has gone into various FURIOUS projects over the years (further explained in Section 2.1), but there has been little user testing of the products developed. It was uncertain how BusTUC[1] (also known as The Bus Oracle), AtB's real-time tracking[2], and applications utilizing these services, have had an impact on the waiting time and daily commute of passengers. Similar tests in Seattle [Watkins et al., 2011] show that commuters using real-time applications wait almost two minutes less than users of traditional information.

After AtB released their real-time API to developers numerous applications, both mobile and web based, have been created utilizing the API in various ways. Many of these applications combine AtB's real-time tracking with the BusTUC. One of the most popular examples of this is "bartebuss" (Mustache Bus)[3], which combines a map of bus stations and BusTUC lookup.

Winter is cold in Norway, and with ice, snow and hilly terrain, vehicles often have problems getting from A to B. It is a common occurrence that passengers continue to wait for a bus that is stuck simply because they do not know it is stuck. The system estimating arrival times does not take this into account, and will simply continue to state that the bus is a couple of minutes away. To inform people of this, and help reduce wait time, showing the locations of the buses in Trondheim on a map was proposed. One example of such a map, "busskartet" (The Bus Map)[4], calculates current bus locations based on the timetables, and displayes them on a map. However this map does not utilize the real-time tracking system from AtB, as the developers found it too unreliable at the time. With the continuous improvements to AtB's real-time systems, it might now be able to support such a map view. Nettbuss, an intercity bus company, has already

[1] `http://busstuc.idi.ntnu.no`
[2] `https://www.atb.no/aapne-data/category419.html`
[3] `http://bartebuss.no//`
[4] `http://www.busskartet.no/`

started showing all their regional buses in a simple map view on their site[5]. These two maps were the main inspiration behind the idea of an interactive real-time map including all buses in Trondheim.

1.2 Goals and Research Questions

To lead the development of the project, two research questions were proposed, and a goal was set to partially answer these during this project. The questions will carry over to the master thesis, where they will be fully answered.

Goal Conduct a user survey to gather the travelers' opinion of the bus today, and gather information from research already conducted to create a mobile application that based on the information gathered.

Research Question 1 How can a mobile application help travelers spend less time waiting for their bus to arrive?

Research Question 2 What is the best way to present route information in a mobile application in order to reduce waiting time and frustration?

The ultimate goal was to create a prototype application containing a map with animated bus routes using real-time data provided by AtB, and support this through relevant research. A simple prototype was planned for the autumn project, which would allow additional research to be done on the concept. Further work on the functionality, user interface, and usability, as well as user testing of the application will be the primary focus in the spring project of 2015. During the spring project tests will be done to see if the application can help people spend less time waiting, and with that get less frustrated. If this application can help make people want to travel by bus rather than car, it might create a ripple effect, meaning this may help the environment and the traffic in Trondheim [Mane and Khairnar, 2014].

1.3 Research Method

During the development of this Autumn project, research went into what already existed of applications and research done on travelers world wide. A lot of helpful information came to the surface which can be used when actually implementing the application. The research consisted of four phases:

1. Project definition
2. A user survey and analysis of the results
3. Literature study and inspecting already existing applications
4. Development of a prototype with map and animated buses

Documenting all these phases was done along side the development. Everything is documented in this report divided into different chapters: Theory and Background, Method, Result and Discussion & Conclusion.

[5] `http://www.nettbuss.no/sanntid`

1.4 Thesis Structure

This section is a description of what the different chapters contain.

Chapter 2 Theory and Background This chapter contains information on what has been done earlier with BusTUC, which is part of the FURIOUS project, and other applications that already exist on the market. As well as a look at technologies related to AtB's real-time system and application development.

Chapter 3 Method This chapter contains information on how the research was conducted during the project.

Chapter 4 Result Results from the project are presented in this chapter. The results are findings in the literature study and the user survey as well as the development of a prototype application.

Chapter 5 Discussion & Conclusion To finish off the report, there is a discussion chapter discussing how the development of the project went, and a conclusion. At the end there is a section on future work containing plans and visions for the future of the idea.

Appendix The appendix contains all the data gathered during the survey as well as a dictionary on abbreviations and other words used in the report.

Chapter 2

Theory and Background

This chapter contains some background information on the project, and already existing applications. Additionally it goes through different approaches of relaying bus information to developers, looking at what AtB is currently doing and their future plans.

2.1 BusTUC and FURIOUS

BusTUC was first developed by Tore Amble at NTNU [Amble, 2000] in 1997. It is an application that allows written natural-language interaction to gather bus departure and arrival times. This means that users can ask questions about bus schedules in plain text into a search field, and get the result printed to them correctly. BusTUC is available at NTNU's web page[1], and supports both Norwegian and English queries. AtB, the public transport service in Trondheim, added BusTUC, or the Bus oracle as they call it, to their web-page[2] in 1999, and has been there since.

After Amble's work, numerous improvements have been made, including many master theses. The TaleTUC [Andersstuen and Marcussen, 2012] master thesis incorporated speech recognition into the mobile application TABuss [Marcussen and Eliassen, 2011], which was another master thesis. Other improvements include incorporating train timetables into the system and extending the functionality to work in Oslo as well. All these theses and extensions are part of the FURIOUS, the Future's Ultimate Intelligent Route-Organizing System, project.

2.2 AtB's Real-Time System

In February 2011, AtB released their real-time system [AtB, 2011], tracking all their buses. This system is developed by a third party company named Swarco [Swarco, 2009]. Figure 2.1 shows the communication flow between buses and the system.

The system is used to track all buses using the route network in Trondheim and Klæbu [AtB, 2014a]. Because Swarco also delivers the traffic light systems in Trondheim, the system includes a signal prioritization [Swarco, 2014]. This means that the bus is prioritized when programming traffic lights, and the bus can travel more quickly [Statens Vegvesen Region midt, 2011].

By calculating the buses' arrival, and making the traffic lights work in the buses' favor, this system is constructed to make the public transportation system more efficient and convenient

[1] `http://busstuc.idi.ntnu.no/`
[2] `https://www.atb.no/spoer-bussorakelet/category1160.html#oracle`

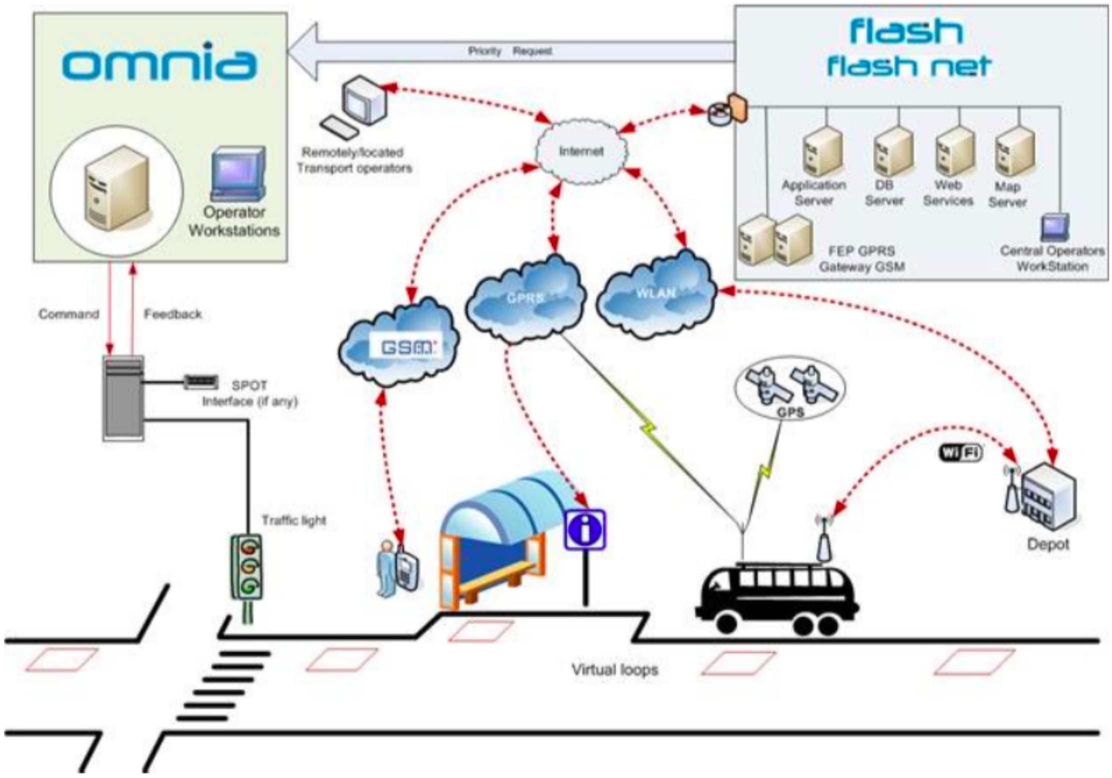

Figure 2.1: AtB's Real-Time System [Torfinn Utne, ICT Manager AtB, personal correspondence, 2014-09-22]

for travelers. This prioritization might help reduce wait time for buses by as much as 40 % or 10-20 seconds at each of the integrated intersections [Swarco, 2014].

The information about the buses' arrival time is available on screens mounted on bus stops, AtB's web site, and mobile applications, thereby making it easy to access. This information is also available to the public as an open source API[3] that developers can get access to by emailing AtB. The only information AtB wants in return is your name, what you want to do with the data, expected load, type of application, and your email address.

The API has limited functionality, and only allows fetching of the data that is available on the screens and the locations of bus stops. It does not provide direct access to the positions of the buses yet, but AtB says that the GPS data of buses is something they will launch 2015 (See Section 2.2.2).

2.2.1 Interaction with the Real-Time System

In order to gather data from the real-time system, the developer has to ask AtB for access via email, as earlier mentioned. Communication with the system is over SOAP, and is returned with a JSON file containing requested information. After sending a request for the two next buses

[3]API can be reached at `http://st.atb.no/infotransit/userservices.asmx`

coming to a bus stop, using the bus stop ID, your developer ID, and password, the server returns the JSON shown in Listing 1 below.


```
{
    "total":2,
    "timeServer":"2014-12-10 13:29",
    "InfoNodo":[{
        "nome_Az":"AtB",
        "codAzNodo":"16011472",
        "nomeNodo":"Strin",
        "descrNodo":"Strindheim",
        "bitMaskProprieta":"0",
        "codeMobile":"Strindheim",
        "coordLon":"10.455689",
        "coordLat":"63.435553"}],
    "Orari":[{
        "codAzLinea":"9",
        "descrizioneLinea":"9",
        "orario":"10.12.2014 13:31",
        "orarioSched":"10.12.2014 13:28",
        "statoPrevisione":"Prev",
        "capDest":"Munkegata M2",
        "turnoMacchina":"80",
        "descrizionePercorso":"71"
    },{
        "codAzLinea":"6",
        "descrizioneLinea":"6",
        "orario":"10.12.2014 13:39",
        "orarioSched":"10.12.2014 13:37",
        "statoPrevisione":"Prev",
        "capDest":"Munkegata M2",
        "turnoMacchina":"51",
        "descrizionePercorso":"34"
    }]
}
```


Listing 1: Example JSON Returned from AtB's Online API

As shown in Listing 1, the descriptions are in Italian. This is because the API is developed by Swarco, which is an Italian company. The most central points are translated and described in Table 2.1.

2.2.2 Interaction with SIRI

AtB is currently planning to deploy SIRI VM (Service Interface for Real Time Information: Vehicle Management), during 2015 [AtB, 2014b]. This will allow access to positions of buses in and around Trondheim, through an API similar to the one AtB is currently using, described in Section 2.2.1.

Table 2.1: Translation of Bus Stop JSON

Element	Translation
InfoNodo	General information about the bus stop
codAzNodo	Unique bus stop ID
descrNodo	Name of bus stop
coordLon	Longitude coordinates of bus stop
coordLat	Latitute coordinates of bus stop
Orari	List of incoming buses
codAzLinea	Line number
orario	Estimated arrival time
orarioSched	Scheduled arrival time
capDest	Destination
turnoMacchina	Unique bus ID

There are many implementations of SIRI VM that are available, some of which are completely open. Sending a request for information about all buses to `http://data.itsfactory.fi/siriaccess/vm/json` (Accessed 2014-12-11), returns a long list of buses containing, among other things, ID, destination, origin, location and bearing for every bus. A portion isolating information about one bus can be seen in Listing 2. In Tampere, Finland, where this bus is located, they have used this information to implement an interactive map. This map is described in Section 2.4.1. SIRI is a well established standard, and therefore AtB's implementation should be quite similar to this example.

```
{"Siri":{
    "ServiceDelivery":{
        "ResponseTimestamp":1418312800851,
        "ProducerRef":{"value":"IJ2010"},
        "Status":true,
        "MoreData":false,
        "VehicleMonitoringDelivery":[{
            "VehicleActivity":[{
                "ValidUntilTime":1418312830027,
                "MonitoredVehicleJourney":{
                    "LineRef":{"value":"9"},
                    "DirectionRef":{"value":"2"},
                    "FramedVehicleJourneyRef":{
                        "DataFrameRef":{"value":"2014-12-11"},
                        "DatedVehicleJourneyRef":"1730"
                    },
                    "OperatorRef":{"value":"paunu"},
                    "OriginName":{
                        "value":"Keskustori O",
                        "lang":"fi"
                    },
                    "DestinationName":{
                        "value":"Annala",
                        "lang":"fi"
                    },
                    "Monitored":true,
                    "VehicleLocation":{
                        "Longitude":23.8712225,
                        "Latitude":61.4779837
                    },
                    "Bearing":102.0,
                    "Delay":"-P0Y0M0DT0H1M36.000S",
                    "VehicleRef":{"value":"paunu_155"}
                },
                "RecordedAtTime":1418312800027
            }],
            "version":"1.3",
            "ResponseTimestamp":1418312800851,
            "Status":true
        }]
    },
    "version":"1.3"
}}
```


Listing 2: A SIRI VM Description of a Bus in Tampere, Finland

2.3 Application Technologies

To allow anyone to gather bus information on-the-go, the natural platform choice is either smart phone application or web. However, when developing an application for smart phones one must consider the divide between Android and iOS in the user base. The use of native applications on each platform would feel the most natural to users, but developing one application for each platform would be time consuming. On the other hand, there are multi-platform alternatives out there, like HTML5, CSS, and JavaScript wrapped with PhoneGap, which would allow deployment on web, Android, iOS, and other mobile devices in one package.

2.3.1 Google Maps

The natural choice of map service is Google Maps[4], as it supports all considered platforms, both native and web based. Google has made individual API's for web, android and iOS because of the differences in the platforms. However, these differences are mostly based on programming language and the implementations contain most of the same functionalities.

Google Maps does require an API key in order to use the map data, but this is free and easily obtainable from the Google Maps website.

2.3.2 Android vs. iOS

Android OS is a mobile operating system based on the Linux kernel and developed by Google since 2003 [Elgin, 2005], and it is a widely popular operating system around the world. This is because of the advantage of Android OS being available on several smart phones in different price ranges, and thereby makes it capture a wider range of smart phone users [Mahapatra, 2013]. iOS is developed by Apple and is the OS used on iPhones and iPads, and it is currently in its 8^{th} iteration with iOS 8[5]. Even though Mahapatra states that iOS is the most popular operating system in Norway at 56 %, Android is not far behind at 41,41 % at the end of 2013. This is a wide user base, and a reason for the application in this project to be developed for Android.

For Android development, Google recommends Android Studio [Google, 2014a]. Android Studio is a new Android Development Environment based on IntelliJ IDEA, taking over for the older Eclipse based Android Development Kit. Android Studio is the environment used to develop the prototype application described in Section 4.3. This environment includes tools to use when developing Android applications including an Android Emulator for running the application, Android SDK tools and a Device Manager.

Android development is done using the programming language Java. This is a widely popular object oriented programming language, which is well documented and easy to use. Because Android is Java based it is more approachable than iOS development to the development team [Goadrich and Rogers, 2011], as they have a lot of previous Java experience, but little to no experience with iOS' Objective-C or Swift.

One of the major differences is the cost of development for each platform. The development team has both Android and iOS phones available, so the cost of devices will not be discussed. However, while the license cost for deploying on Apples App Store is $99/year[6], Android only requires a one time fee of $25[7]. This difference makes a great impact on the cost of development, and maintenance.

[4] `https://developers.google.com/maps/`
[5] `https://www.apple.com/ios/`
[6] `https://developer.apple.com/programs/start/ios/`
[7] `http://developer.android.com/distribute/googleplay/start.html`

2.3.3 Web

Using a tool such as PhoneGap[8], developing applications for multiple platforms can be done all at once. It uses HTML, CSS and JavaScript to allow seasoned web developers instant access to the mobile application market. By using a technology like PhoneGap, an application for every platform, including web sites, could be created from the same code. A problem with these types of applications is that it is harder to make a powerful application tailored for a given platform.

From personal experience an application made using web techonlogy may feel slow. Another reason for not choosing this type of application development was the fact that both the authors are familiar with Java development, and just one with the HTML, CSS, JavaScript development.

2.4 Existing Bus Maps and Applications

There exist a lot of different mobile and web based applications already displaying timetables and maps indicating locations of public transportation systems today. This section is going to describe some of these that inspired the idea for this project.

2.4.1 Bus Maps

There exist many different simulated and real-time maps of buses today. One example is, the earlier mentioned, busskartet[9], which currently is the only one covering AtB's routes. However, busskartet does not utilize the potential of AtB's real-time system, described in Section 2.2. Instead it predicts the bus movements according to the timetables, and is thereby rarely accurate.

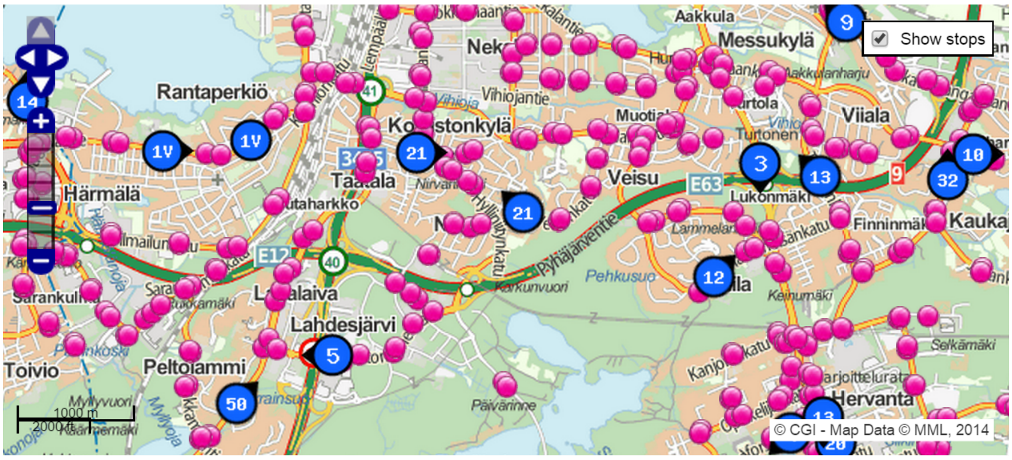

Figure 2.2: Tampere Public Transport's Traffic Monitor

To approximate locations, busskartet combines the timetables with Google's Directions API[10], which has many limitations. The Directions API is made for regular drivers who are not allowed to use "bus only" roads, while sometimes the trams do not follow the road at all. In addition to this, inaccurate positions of bus stops sometimes results in bus stops being placed on the opposite side of the road resulting in Google Directions taking massive detours to turn around.

[8] http://phonegap.com/
[9] www.busskartet.no
[10] https://developers.google.com/maps/documentation/directions/

These issues sometimes result in displaying buses taking long detours at extreme speed, on roads too tight for bus traffic.

An example of a map that uses real-time tracking would be Nettbuss' real-time map[11]. This map updates about once every two minutes, but as this map concentrates on intercity buses, accuracy is not as important as in a city where a bus can pass multiple bus stops in two minutes.

Perhaps the best example of real-time tracking of buses is deployed in Tampere, Finland. Here they have successfully deployed SIRI VM (Section 2.2.2), and created a map[12] displaying all buses and bus stops in and around Tampere. In contrast to nettbuss' map, this map updates about every 5 seconds. A picture of the map in action is displayed in Figure 2.2. The dots on the map are bus stops, and the numbered circles are buses.

2.4.2 Applications

There are many different applications on the market today, trying to help travelers to find the public transport timetable as easily and convenient as possible. These are important for our development and inspiration when creating a new application. They may contain both good and bad features, making it easier to know what works in practice and what does not.

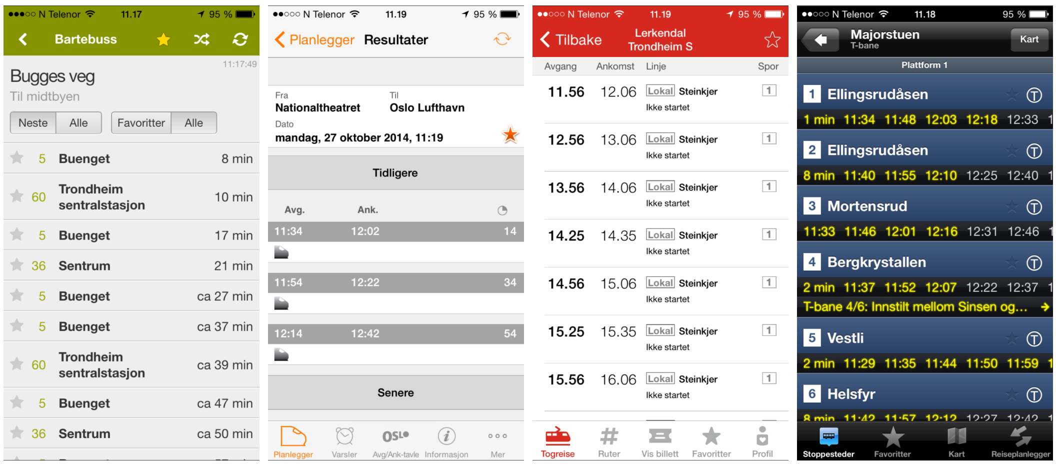

Figure 2.3: Existing Applications
From the Left: Bartebuss, Flytoget, NSB and Ruter Reise

Figure 2.3 shows four applications that already exist. The leftmost application is the one called Bartebuss[13]. This application is the one that was most popular among the participants in our survey, described in Section 4.1. Bartebuss utilizes the real-time system showing when the next bus will arrive at a bus stop, in this case "Bugges veg". Other features included is the Bus Oracle, earlier mentioned in Section 2.1, and a map displaying all bus stops, but no buses. The application can also be reached on the web[14].

Flytoget[15] and NSB[16] are both train applications. Flytoget is a more narrow application than any of the others. This is due to the airport express train just traveling between Oslo Airport

[11]http://nettbuss.no/sanntid
[12]http://lissu.tampere.fi/?lang=en
[13]Bartebuss: https://play.google.com/store/apps/details?id=com.runemartin.bartebuss&hl=en
[14]http://bartebuss.no/favoritter
[15]Flytoget: https://itunes.apple.com/no/app/flytoget/id456924023?l=nb&mt=8
[16]NSB: https://itunes.apple.com/no/app/nsb/id439655098?l=nb&mt=8

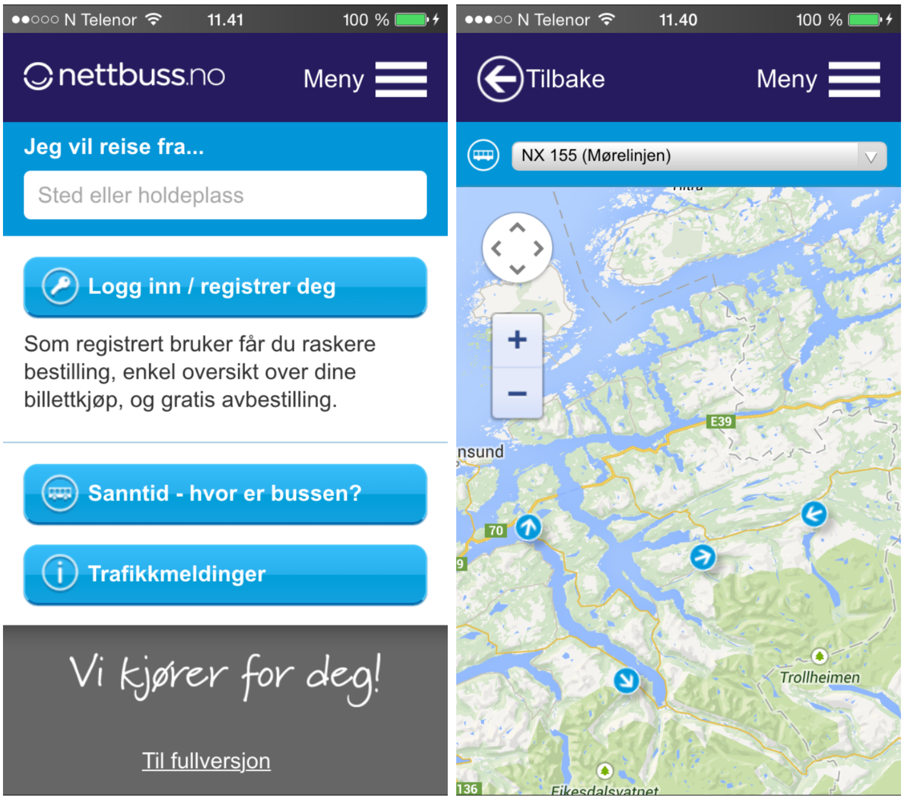

Figure 2.4: Nettbuss Real-Time Map

Gardermoen and the city of Drammen. This means that the application dos not need to render a lot of data, and therefore looks a lot simpler than the others. NSB is an application used by travelers all over the country, traveling by train.

The Ruter Reise[17] application is used throughout the eastern region of the Norway. It is a very popular application, and in contrary to Bartebuss, which is made by a third party, is an official application for the public transportation system there, called Ruter.

The earlier mentioned Nettbuss real-time map is also implemented as a mobile application[18], shown in figure 2.4. The arrows on the map shows where the buses are, and which direction they are traveling. This application inspired many of the ideas for a map utilizing the real-time system of AtB in Trondheim.

[17]Ruter Reise: `https://itunes.apple.com/no/app/ruterreise/id299318111?l=nb&mt=8`

[18]`https://itunes.apple.com/no/app/nettbuss.no/id843705088?l=nb&mt=8`

Chapter 3

Method

Based on the research questions presented in Section 1.2, a method using both literature review, and a survey was created. This way a thorough picture of bus travel both inside and outside the small city of Trondheim, Norway could be created, as well as a focus towards our research questions not presented in previous works. The project focuses on improving public transportation in Trondheim. Plans were created to develop a mobile route retrieval application based on AtB's real-time data.

3.1 Information Retrieval

In order to gather as much relevant information as possible, a literature study was conducted. The goal of this study was to find general information about accuracy of bus tracking, how it is solved elsewhere, and the impact on waiting time produced by mobile and web based applications, both timetable based and real-time. Researching existing technologies and solutions resulted in new ideas and inspiration leading the mobile application in new directions, reflected in the research questions.

3.1.1 Gathering Articles

Different articles and papers were needed for the background of this project to be complete and thorough. A lot of details about public transportation systems in different cities around the world has been documented, which means that there are no lack of papers to read. In order to find the papers that were the most relevant for the project, queries were executed on both Google Scholar[1], and Oria[2]. The idea behind using multiple databases is to avoid the problems of their ranking algorithms [Beel and Gipp, 2009], and thereby finding more relevant works. The literature study mostly happened in the months of September and October 2014. Many of the queries consisted of combinations of the words: "GPS", "bus", "timetable retrival", "public transportation", "tracking", "mobile", "ticketing", "mapping", "survey", and "technology acceptance model". The problem with a pure web based search is that many old articles are not available on the web, but as the scope is set to mobile applications old non-digital articles could be omitted. The articles span from 1972 to 2014.

In order to answer the two research questions in Section 1.2, two different scopes for the reviews were made: one broad to deal with route and information retrieval for bus travellers,

[1] http://scholar.google.no/
[2] http://www.oria.no

and a narrow scope for mobile route retrieval applications in public transportation. All the articles used in this project are listed in the bibliography at the end of this report.

3.1.2 Selecting Articles

As the queries found through the previous section produced numerous results, a selection routine was necessary to make sure the articles were as relevant as possible. Articles containing bus tracking using the GPS on the travellers' phones were excluded, as they are too inaccurate and are not in widespread use by the bus companies themselves. In addition many articles describing the development of mobile ticketing applications were excluded as well. Because traveller satisfaction and waiting time were essential to the research, many of the chosen articles were case studies, with either anonymous traveller monitoring, interviews, or both. The main attributes looked for were traveller satisfaction, waiting time (both perceived and actual), and how they found their bus schedules. A lot of the articles thus has a focus on empirical studies regarding travellers. It is important the mobile application developed in this project is useful to travellers, and thus the articles found were incredibly important for the research.

3.2 User Survey

In order to gather user information and feedback on already existing systems and the users' thoughts and travel habits, a web survey was conducted. It was shared with users in different cities, mostly Trondheim and Oslo. Face-to-face interviews were tested before creating the web survey. This gave bad results since the people asked waited for their bus, and the bus usually arrived before finishing the questionnaire. Additionally the interviews where too time consuming and inefficient for the amount of questions given. This was why the web survey where created. By creating the web survey, the survey could be expanded to include other cities in addition to Trondheim. This meant that information about how timetable retrieval in other cities worked could be taken into account when analyzing the survey. This would allow a broader set of existing applications to be examined, and thereby inspire the development of the map presented in this project.

Ten questions were asked, with five follow up questions if the interviewee's primary way of gathering route information was through a phone application. Most of these questions were multiple choice, but also contained an "other" alternative where the users could input their own special cases if needed. The only exception being question 10e where the user was asked to input their personal experiences with their application of choice. All the questions asked are in the following list, and the results are found in appendix B.

1. How many times a week do you commute by bus?
2. What is your work status? (Voluntary)
3. How old are you? (Voluntary)
4. In which city do you utilize bus the most?
5. In what context do you commute by bus?
6. How long do you usually wait for the bus?
7. Do you think knowing the exact location of the bus would be helpful to reduce the time spent waiting for the bus?

8. On a scale from 1 to 10 (10 being extremely frustrated), how infuriated do you become by waiting for the bus?
9. If you arrive late for your bus, what is the reason?
10. What service do you primarily use to find route information? If the answer was phone application, a new set of questions were asked in addition
    (a) Which phone do you use?
    (b) Which application do you primarily use?
    (c) On a scale from 1 to 10 (10 being super happy), how satisfied are you with the application of your choice
    (d) How did you hear about the application?
    (e) Please elaborate on your experiences with the application (Voluntary)

Chapter 4

Results

This chapter includes all the results from the work in chronological order. Starting with the user survey earlier mentioned, moving on with some findings in the literature study, and finishing with the demo application developed.

4.1 User Survey

Most of the data gathered was generated through a web based user survey created using Google Forms, and distributed on Facebook. The survey also included four face-to-face interviews with people on the street using the same questionnaire. All the data gathered is included in Appendix B.

4.1.1 Waiting Time

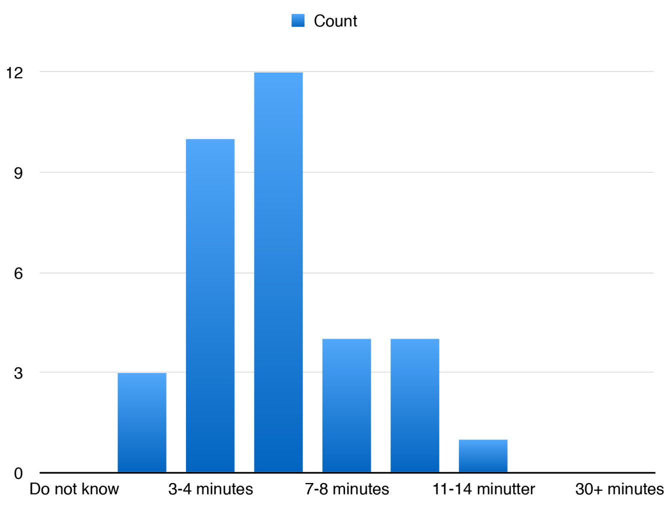

Figure 4.1: Perceived Waiting Time in Trondheim

The overall picture shows that most of the people answering the survey, 81 %, thought that having the opportunity to see the bus' location would decrease their waiting time (Figure B.4). This result confirmed that the idea of a mobile map application might be desired, and with that a useful and a unique addition to the myriad of available mobile applications in the Trondheim area. If the participants had shown resistance to the idea, it would have to be reworked. It is worth mentioning that the waiting time data gathered are the participants own perceived waiting time, and may not reflect the actual time spent waiting. As shown in [Watkins et al., 2011], the perceived waiting time and the measured waiting time may vary with as much as two minutes. Since the focus of our application is to make travelers less frustrated, and wait less, the focus is on their perceived waiting time, since this affects the frustration.

Table 4.1: Reasons for Missing the Bus in Trondheim

Reason	Count
I am never too late	3
The bus was ahead of schedule	2
Got wrong route information from the app minutes	5
Real-time was inaccurate and showed the wrong time minutes	5
Miscalculation of walking time to the bus stop	12
Did not use the route information	3
The bus goes so often that I do not care if I'm one minute late	3
Other	1

Of the people traveling with bus in Trondheim, 65 % thought they waited more than five minutes for the bus to arrive (numbers in Figure 4.1), but the reasons for this varied. Some buses were late, and passengers came too late for the bus. One of the questions where why people ran late for the bus, and the results in Trondheim are described in Table 4.1. Even though there is a large variation in the answers, the one that sticks out is that people have a hard time calculating how long it takes to get to the bus stop. Taking this into account a new feature suggestion appeared: make it possible for people to get a notification when it is time to go to the bus stop.

Table 4.2: Average Percieved Waiting Time by Route Retrieval Method

Retrieval method	Number of answers	Mean time (min)	Mean trips/week
All	55	5.2	3.9
Web application	6	5.7	1.8
Mobile Application	26	5.0	4.0
Other	11	4.9	7.1

As depicted in Table 4.2, the average perceived waiting time of travelers using modern mobile applications are about the same as the traditional travelers. However the users of web applications believe they wait an average of 0.7 minutes (42 seconds) longer than the other users. This increase in waiting time might be connected to this group being the least active bus users, with only 1.8 trips per week on average, and they might therefore arrive earlier at the stop than the more experienced travelers. In addition, those that travel less by bus, might not want to go through the hassle of installing an application, or finding a timetable, for just one query. Either way the frustration level seems to be about the same for all sorts of travelers. This is shown in Table 4.3.

In the questionnaire, the frustration level when the bus was too late was answered on a scale from 1 to 10, where 10 was extremely frustrated. The total result form the survey is visualized in Figure 4.2. This figure shows that the frustration level looks like a normal distribution where most of the answers were "5". Every participant in the survey was at least a little bit frustrated with the bus being late: no one answered “1”.

Table 4.3: Average Frustration by Route Retrieval Method

Retrieval method	Mean frustration level
All	5.8
Web application	5.5
Mobile Application	6
Other	5.8

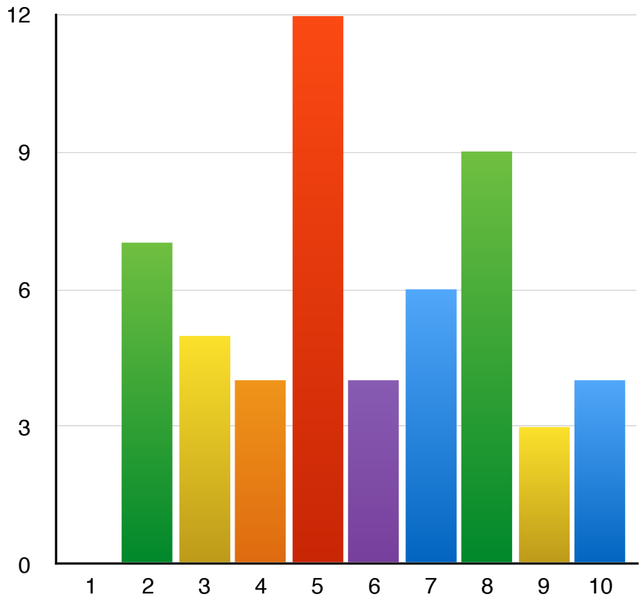

Figure 4.2: Frustration Level. All answers

4.1.2 Application Usage Feedback

The section of the questionnaire about application usage gave some feedback on people's application habits: likes and dislikes. This gave some inspiration for the application about to be developed.

Of all the participants in this survey (63), 25 people answered that they mostly used applications to find the route information (question 10, Figure B.5). The most popular applications are Bartebuss in Trondheim and RuterReise in Oslo. These are two really good and different applications with different highlights and features. By combining the best features from these two, the application described in this project, will have a fair chance to compete in the market.

In addition some people wrote a little comment on what feature they liked about the application, or what they missed and wished was there. About RuterReise, people commented that the application needed the ability to show the whole trip in a map, that it is a great application everybody should have, and that the application just shows when the next bus will pass, and not the entire route from A to B. Even though the last comment is false, it was still something someone answered. It is possible to get the travel from A to B in the RuterReise application. This makes it seem like the feature is not visible enough for users to notice it. By the looks of the feedback on Bartebuss, the application tend to not update the time of arrival correctly so that the buss suddenly arrives before the real-time suggested. Another person commented that it is hard to find the timetable for a given bus route. It should be possible to click on the bus route one desire, and not just the stops. An example is being able to click on bus number 5, and see its entire route.

4.1.3 Survey Conclusion

The results from the user survey showed the differences in waiting time according to where people traveled with public transportation and how they found the bus schedules. The result and feedback received from the user survey, including waiting time, frustration level, and feedback on the application they are currently using, were taken into account when developing the prototype application. In addition to making it easier and better for bus travelers in Trondheim to get timetable information, one should research whether the application makes the whole trip more satisfying by making the waiting time shorter, and reduce frustration. Several participants in the survey thought the map idea proposed sounded like a good idea, which is a great starting point for the application. Creating an application that already has had negative feedback before creation seems unnecessary. The confidence that this idea might work as a mobile application remains intact, and it will be interesting to test its impact on travellers once it is complete.

4.2 Research on Travelers

Over the years, a lot of research has gone into public transportation systems. This chapter denotes some of the findings made during the literature study.

4.2.1 Waiting

The article [Golob et al., 1972] documents a research of consumer preferences of public transportation systems. The research contains data on more than 700 responses from face-to-face questioning. Some of the more important characteristics of the transportation system, according to the participants were "less waiting time" and "shorter travel time". Some other characteristics were "arriving when planned" and "less walk to pick-up". By this one might assume that people using public transportation want the experience to take as little time as possible. Regarding travel time and pick-up points, there is not much we are able to improve from our standpoint. Something we can try to change is the waiting time by making it possible for travelers to locate the bus at a given time.

The article [Watkins et al., 2011] describes a user test done by measuring the wait time of passengers on bus stations in the proximity of The University of Washington, in Seattle. In addition to measuring their waiting time, an interviewer asked the passengers various questions about how long they had waited, how aggravated they became because of the waiting, and how they found their route information. Watkins et al. hypothesized that users of real-time route acquisition services (in their case, the mobile application OneBusAway), would decrease actual waiting time, and have it converge with the actual waiting time. This in turn, would decrease the aggravation level of the passengers. Through this test Watkins et al. showed that users of real-time information wait almost two minutes less than users of traditional route information (timetables, trip planners or just showed up at the station). However this decrease in wait time did not reduce the level of frustration produced by waiting for the bus. One limitation of the article is that it focuses its attention on a small area of Seattle, and might have produced different results elsewhere. This is something necessary to test in Trondheim, and maybe an application helps here.

4.2.2 Mobile Application Acceptance

The article [Mallat et al., 2008] focuses on mobile ticketing applications. The document states that this field of mobile commerce has been quite successful. Even though our focus is not on ticketing systems, but rather on timetable systems, the two systems go hand in hand. Mallat et al. states that usage of mobile applications has become more and more natural for the average person. People do not have to stop by a newsstand to get the latest news, buy a CD at the music shop, or pay for their ticket on the bus with cash any more. Mobile applications are easier for users to reach on the move, since everything is available on their mobile phone. This makes it easier to check the bus' timetable while walking to the bus stop, and decide whether it is time to start running in order to catch the bus.

During the development of this project a web survey was conducted (Section 4.1 and Appendix B). The results from this also showed that participants mostly used mobile applications to check the bus' timetable. The second most used platform to check the timetable was webpages.

In order to test the systems acceptance, articles were read during the literature study to find methods and models to help structuring such an analysis. Two types of models were found interesting. These are described in the next two sections.

Technology Acceptance Model

The Technology Acceptance Model was originally proposed by Fred Davis in 1989 [Davis, 1989]. The model has been refined and extended for different uses since then. As described by [Chuttur, 2009], the initial model focuses on three factors that explain the user's motivation: *Perceived ease of use*, *Perceived usefulness* and *Attitude Toward Using*. Later these have been a little refined, but still leaving perceived ease of use and perceived usefulness as two very important factors explaining user acceptance.

Davis defines, in [Davis, 1989], perceived usefulness to be:

> "The degree to which a person believes that using a particular system would enhance his or her job performance"

, and perceived ease of use to be:

> "The degree to which a person believes that using a particular system would be free of effort"

Davis has then later modeled scale items which users need to answer in order to measure the acceptance of the system. There are, after two rounds of reduction, six items per factor. These are shown in Table 4.4 and Table 4.5. These tables are taken from [Chuttur, 2009].

Obviously these are not questions directly usable for the project. This means the questions have to be reformulated to fit with the application and its usage. Davis' Technology Acceptance Model focuses on a system being used to make someone's work easier and better. Since our application is not to be used with work, the model has to be reformulated to fit this too.

Using this model, it is possible to measure the travelers' acceptance by testing the application on a selection of travelers. The results will hopefully give an indication on the applications future.

Table 4.4: Revised 6 Items Scale for Perceived Usefulness

Item No.	Candidate item for psychometric measures for perceived usefulness
1	Using the system in my job would enable me to accomplish tasks more quickly
2	Using the system would improve my job performance
3	Using the system in my job would increase my productivity
4	Using the system would enhance my effectiveness on the job
5	Using the system would make it easier to do my job
6	I would find the system useful in my job

Table 4.5: Revised 6 Items Scale for Perceived Ease of Use

Item No.	Candidate item for psychometric measures for perceived ease of use
1	Learning to operate the system would be easy for me
2	I would find it easy to get the system to do what I want to do
3	My interactions with the system would be clear and understandable
4	I would find the system flexible to interact with
5	It would be easy for me to become skillful at using the system
6	I would find the system easy to use

Several extensions to the application have been proposed, and the Mobile Services Acceptance Model is one of them [Gao et al., 2014]. This model is extended to fit an Acceptance measurement on a mobile information system, and might be a better fit for the project. The model adds three additional factors to TAM: Trust, Context and Personal Initiative. In addition to these three factors, the model uses Characteristics to examine students' perception of mobile information services.

Another extension was a theoretical one [Venkatesh and Davis, 2000], which uses the second proposed Technology Acceptance Model by Davis, with a focus on organizations. Testing it on different employees in an organization, it is shown that the TAM2 is strongly supported across the organizations tested.

System Usability Scale

System Usability Scale, shortened to SUS, is a "quick and dirty" usability scale proposed by John Brooke [Brooke, 1996]. The scale was created to utilize simplicity and speed, consisting of a questionnaire with only ten questions. The questions are as follows [Sauro, 2011]:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.

9. I felt very confident using the system.

10. I needed to learn a lot of things before I could get going with this system.

The questions are to be graded by users on a scale of five, where five is “Strongly agree” and one is “Strongly disagree”. By having a strict questionnaire like this, it is possible to calculate a score, that in turn can tell if the application is useful. Sauro [Sauro, 2011] show step by step application of the System Usability Scale. Calculating the SUS-score is done by following these steps:

- For odd items: subtract one from the user response.
- For even-numbered items: subtract the user responses from 5
- This scales all values from 0 to 4 (with four being the most positive response).
- Add up the converted responses for each user and multiply that total by 2.5. This converts the range of possible values from 0 to 100 instead of from 0 to 40.

The output value can be interpreted on a scale and may be defined as perceived usability of the application. The scale is not linear, but rather a curve as displayed in Figure 4.3.

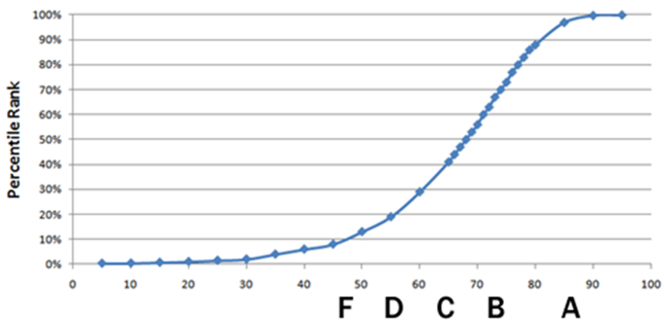

Figure 4.3: SUS Score [Sauro, 2011]

This is a quicker usability measurement, but it is also not quite as thorough as the TAM. TAM calculates perceived ease of use as well as perceived usefulness, and is therefore a more descriptive model, which gives a more thorough result.

Seeing as the SUS was created to provide a quick and easy way to measure usability, one might think it is not very accurate. According to [Bangor et al., 2008] this might not be exactly correct. The article presents findings worth nearly ten years of System Usability data. Based on this data, the ten questions is analyzed one by one for its validity, and compared to the other nine questions. Originally, the SUS was only used once on a software project, but Bangor et al. suggests that the scale may be used iteratively, to better see the evolution of the application. There are also other ways suggested that the SUS can be used to supplement usability testing. These are listed below.

- Providing a point estimate measure of usability and customer satisfaction
- Comparing different tasks within the same interface
- Comparing iterative versions of the same system

- Comparing competing implementations of a system
- Competitive assessment of comparable user interfaces
- Comparing different interface technologies

[Bangor et al., 2008] concludes with the SUS being a robust tool in helping assess the quality of a broad spectrum of user interfaces.

4.3 The Application

The idea behind this project was to create an application that would display all the buses in Trondheim on an interactive map which, unlike busskartet, would use the real time data provided by AtB.

During the project a small proof-of-concept Android application was produced to find how well such an application would perform. As the idea was laid down, the accuracy of the real-time service was uncertain, and a prototype had to be developed in order to confirm that it was achievable.

Every bus in Trondheim is equipped with a GPS, and is continuously sending its current position to a server. Here the position of the bus, together with bus routes and bus stop locations are combined to estimate when the bus will arrive at each stop on its route. As shown in Figure 4.4, this estimate is available through an online API, and is the same information used by the information screens at the bus stop.

AtB does currently not provide direct access to positions of their buses, but this is planned for release in 2015 [AtB, 2014b] through a SIRI VM system [Knowles, 2008] developed by Trapeze (previously known as KiZOOM). If released in time to be included in the next iteration of this project, it would most likely be the method used to gather positions, as it will most certainly surpass the accuracy of the estimation method described in this section.

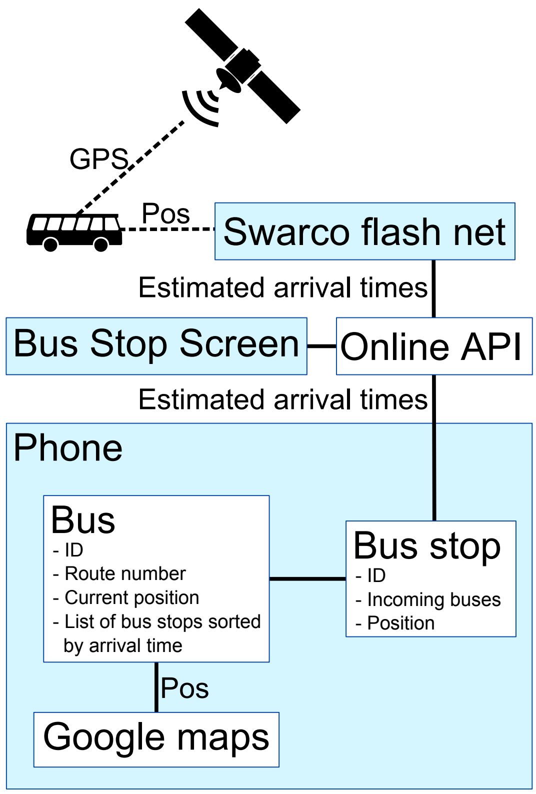

Figure 4.4: Flow of Positional Data from Bus to Map

However AtB does allow anyone to fetch real-time estimated arrival times for buses on each bus stop, and the conversion of this data back into coordinates is shown in Figure 4.4. The information received through the API contains a list of incoming buses with an ID, route number, and an estimated arrival time. A description of how this data is transferred and what it contains is shown in Section 2.2.1. This application only utilizes the bus stop ID, its name and location together with the route number of each bus, its estimated arrival time, and ID.

To improve the scalability of the system shown in Figure 4.4, moving the position approximation out of the phone to a server somewhat similar to the Adapter pattern [Gamma, 1995] was proposed, leaving only Google Maps and a connection to this server on the phone itself. This would heavily decrease the traffic on AtB's servers, as only one device would query their servers instead of hundreds of individual phones. When the goal is to continiously gather data from all bus stops in Trondheim, the impact such an architecture would be huge. A centralized platform like this would also allow easier deployment to any platform, be it Android, web or iOS. In addition it would allow removal of unnecessary overhead and would allow querying for a single bus instead of by an area. It would also allow more rapid changes to the algorithm if needed, as it would not have to go through Google or Apple for application evaluation, but instead be deployed on a private server.

Using information from all bus stops, the application creates a list for each bus in route containing upcoming bus stops for that bus, and corresponding arrival times. By continuously updating this list, the application can know when a bus stop has been passed, as that stop is simply removed from the list. Finding the next stop on a bus' route is as simple as sorting this list by arrival time. Afterwards the system will look up the position of that stop through the same API. Now that the application knows where each bus will be, and when it will be there, it will animate a path between the bus' current position and the next bus stop's position, arriving at the stop at the estimated time. These position approximations are then drawn on a map on the users phone.

Because the system only uses data gathered from the online API, the need for timetable data is eliminated and with it a lot of maintenance work of constantly updating and parsing these disappears.

The proof-of-concept limited its gathering to nine different bus stops in north-eastern Trondheim. These stops follow one of the most heavily trafficked roads for buses in Trondheim, and carries over ten different routes in each direction. Because of this the application could draw a lot of buses by only querying a small set of stops, and avoid causing too much traffic to the AtB servers with this poorly optimized gathering solution.

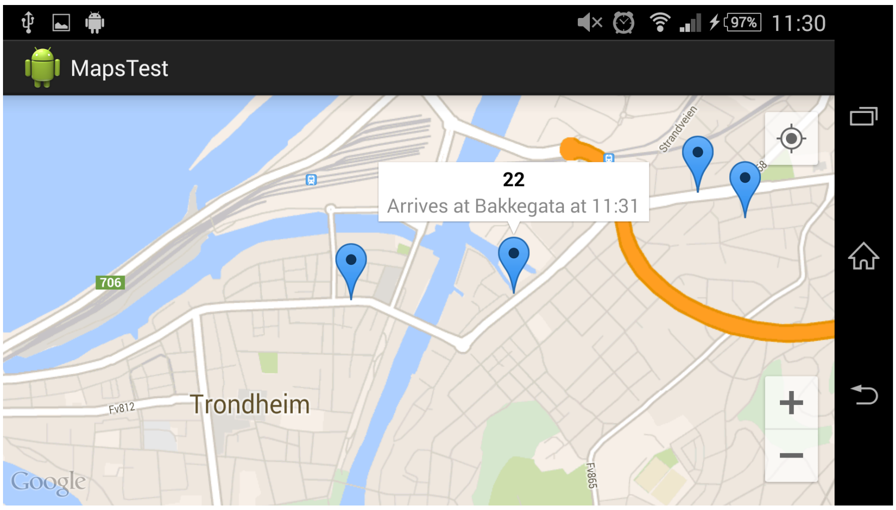

Figure 4.5: Prototype Example Screen

The functionality of the application is quite limited, but as shown in Figure 4.5 (displaying bus 22 just departing from Solsiden, heading for Bakkegata) it allows users to view arrival times,

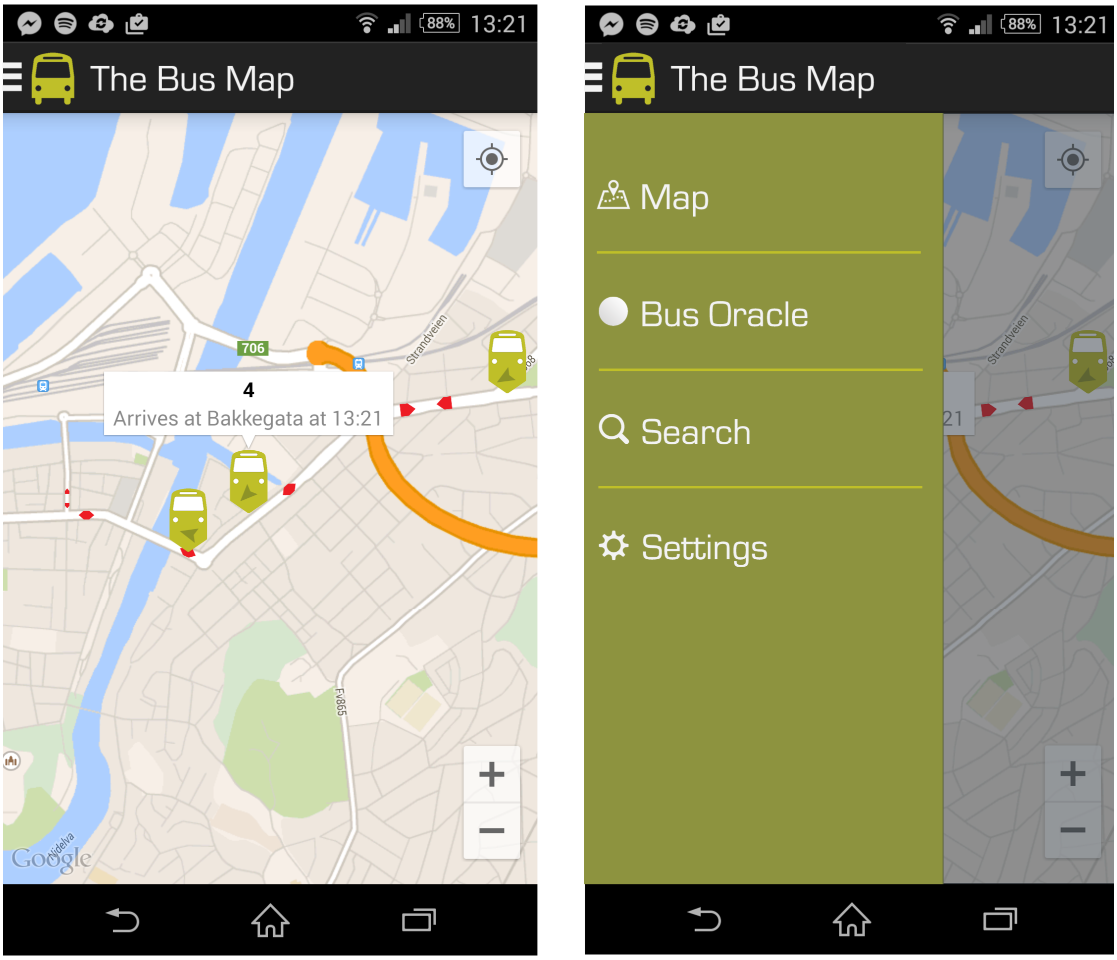

Figure 4.6: Visualized Plans for the Application

line number and position of a selected bus. It does not display where the stops are on the map, and the bus will not follow the road, but instead follow a straight line between each stop. All these features were considered unnecessary for such a crude prototype, but will be reconsidered for the full release.

To better be able to describe the future plans for the application, Figure 4.6 was created. It displays two screens for a future applications. The applications main feature is the map with the moving buses. This is displayed on the screen to the left, and shows three buses, their bearing, and bus stops. All icons used are in Figure 4.7, for better viewing. The application will have a menu which is reachable from the three lines at the top of the application, or by sliding a finger from left to right. The menu is visible at the right in Figure 4.6.

All, but one icon used in the menu are known icons from other applications: a gear for settings, three bars for menu, a map for map, and a magnifying glass for search. The icon for the Bus Oracle was created specially for this application since such an icon does not exist, and is inspired by a crystal ball used by oracles to predict the future.

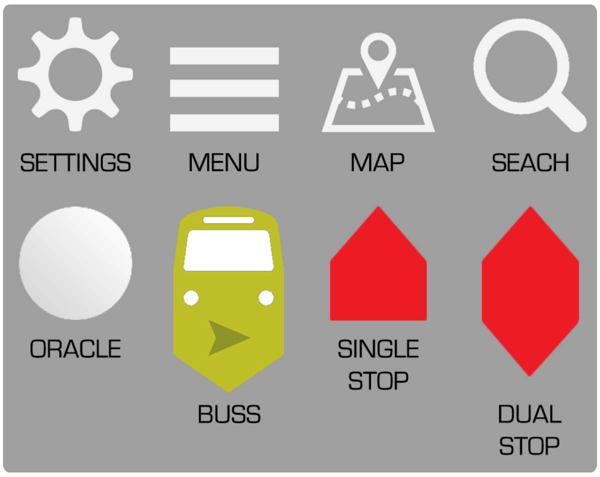

Figure 4.7: Icons Used in the Visualized Plans for the Applications

The icons on the map were created to better display the correct information on the map. The buses have their own icon, a green bus with an arrow inside, showing the bearing of the given bus. All stops are visualized as red arrows on the road. There is a distinction between a single stop and dual stop. The difference on the map is that there is an arrow on each side of the stop for the dual stops, rather than just on one side, like it is for the single stops. In practice the difference depends on where each stop is positioned. Some bus stops will have its opposite stop (the stop that is for the other direction) directly on the other side of the road, these are displayed as dual stops. For stops that are more spread, the stops are displayed as single stops. This is for better seeing where the stop actually is. Some stops are only one direction, and then there will only be one single stop for this place.

As for colors, the light green is inspired by AtB's logo. The stops are red to symbolize "stop", and it is easy to see on the map. All menu icons are white because it is comfortable to look at when the background is the darker green. The dark green was found on the AtB homepage as the darker color of a button. All the color codes are found in Table 4.6.

Icon	Color Code
White	#f3f3f3
Light Green	#bfbf28
Dark Green	#8d9327
Red	#ed1c24

Table 4.6: Color Codes

Chapter 5

Discussion & Conclusion

This chapter contains the discussion and the conclusion of the project. There is also a section of further work and some visions for the master thesis.

5.1 Discussion

During this autumn project's time span, a literature study and user survey was conducted. There was a lot of relevant information revealed during these steps. From the survey results, one can clearly see that the passengers desire such a bus map to lower their wait time. Therefor the development of the application will be continued in the spring project. Reading papers about waiting time, it was assumed that travellers using the mobile application to look up bus routes would wait less than those using anything else. This was proven to be wrong during the user survey. There was very little difference in waiting time based on our answers. However, based on the results from the survey, it is believed that the map application might help reduce the waiting time. This is something to work with during the master thesis, and it will be exiting to see the results from the research by the summer of 2015.

This project was heavily inspired through literature studies of research done on waiting time and frustration level, as well as technologies and mobile application impacts. Based on personal experience, the buses in Trondheim have a tendency to be late. Assisting people with information on where the buses are on a map might help with planning when to leave for the bus stop, and by that hopefully not decrease their wait time.

During the research on evaluation methods of usability and user acceptance, Technology Acceptance Model was discovered. The model seems like a great tool for acceptance testing on users, but might require a different approach to fit a mobile application for travelers. The model can give responses helping the development to move in the right direction. Another easier model, or scale, one might use is System Usability Scale. This is a much quicker and easier assessment of usability, which might also be useful to the project.

In Addition to the literature study and the survey, a prototype map application was created. The development of this prototype is strongly linked to the responses from the survey, where 81 % of the users believed such a app would help reduce wait time. The prototype was developed to see if our ideas for a real-time map were realizable, which was proven to be true. A lot of work needs to be done to make the map more useful than the prototype is, and much of this work is detailed in Section 5.3.2.

5.2 Conclusion

A lot of background information and research already conducted was unveiled during this autumn project. This helps with the project's approach and future development. A map application will be developed during the master thesis in the spring of 2015. The development will be based on the findings from this project, and hopefully will answer the research questions defined in Section 1.2.

As of the goal of this autumn project:

> *Conduct a user survey to gather traveling people's opinion of the bus today, and gather information from research already conducted to create a mobile application that answers to the information gathered.*

This goal was achieved and gathered a lot of background data for the development of the application, and further research on the completed map. All the information gathered is described in this report, mostly in chapter 4, and some background information in chapter 2.

With the help of the gathered information, application prototype turned out well, and will be implemented with many improvements in the next iteration of the project. Some plans for the future work on the application are denoted in the next section (Section 5.3), and a proposed look of the application is displayed in Figure 4.6. It is believed that all the plans and proposed features realizable during the spring of 2015, and that the application can be tested on users to see if the idea works in practice.

The focus feature of the application will be the map and making it work perfectly. Iterative feedback from users will be taken into account to make it usable. For the application to be more desirable, the bus oracle will be added, together with some search functions to find the desired bus on the map. Other ways of finding the bus will also be looked into, but this will be planned in the spring. We believe it is better to have some features that work perfectly, than a myriad of mediocre features.

5.3 Future Work

There were developed plenty ideas and visions for the future, during the project, and these are described in this section. Since the project is not finished yet, just the autumn project, this section contain mostly ideas for the work during the master thesis.

5.3.1 User Acceptance and Testing

As earlier mentioned, there exist a lot of different testing models, and in this project two were detailed in Section 4.2.2. For the spring work, the models need to be further researched, and modified to fit the project. There is also a need to see which one will work best, or if a combination of the two might work. All of this will be worked on in the master thesis. Either way, model is great to assess the quality of the application's user friendliness, and usefulness. It is very important for us that the application works in practice, and is usable by the travelers.

5.3.2 Application

In this project a small proof-of-concept android application, was developed (See Section 4.3). It successfully proved that such a real-time bus tracking map is within reach, but would require a lot of optimization on both technical and GUI aspects. As the application is to be released on Android it should also follow the official Google design guide [Google, 2014b].

As of now each instance of the application queries the AtB servers for nine different stops, but by expanding this to all stops in Trondheim and multiple users, this will without a doubt cause a lot of strain on their servers. Therefore an intermediary server should be in place, which will constantly update via the AtB API, and distribute this to the users. This will not only reduce traffic on AtB servers, but also for the individual user as the intermediary can remove the unused information returned, and only send out the information necessary to display bus positions.

When AtB opens up their SIRI VM to the public [AtB, 2014b], it should be easy to replace the current estimation method, with this simpler and more accurate system. AtB plans to release this system in 2015.

In addition to these technical improvements, tracking every bus introduces many possibilities for future features. There are also many other features, not directly related to bus locations, that should be available in order to compete with the current market of bus applications in Trondheim.

Locating just one bus As the intermediary will have estimated the positions of every bus, a user should be able to just ask for the position of one of them. This would not be as easy without an intermediary as the phone application would have to gather information from every stop on that bus' route to estimate its location, while the server constantly does this.

Integrating BusTUC The user should be able to query the Bus Oracle from within the application.

Displaying BusTUC response on a map By parsing the response from BusTUC, it should be possible to find which bus it recommends, and then displaying the location of this bus on the map.

Website With all bus positions available on a server, it should not be hard to implement a web based map. This map could, in a pinch, be used as a base for an iOS version.

Indicating positions of bus stops Presenting where bus stops are in relation to the buses in traffic and perhaps the location of the user.

Showing direction of buses Reporting the bearing of the buses could be useful when buses are moving slowly or standing still, as then the animation alone might not be enough to decipher its direction.

Bibliography

[Amble, 2000] Amble, T. (2000). BusTUC - A Natural Language Bus Route Oracle. In *Proceedings of the sixth conference on Applied natural language processing*, pages 1–6. Association for Computational Linguistics.

[Andersstuen and Marcussen, 2012] Andersstuen, R. and Marcussen, C. J. (2012). TaleTUC: Automatic Speech Recognition for Bus Route Information System. http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A603563&dswid=-4058.

[AtB, 2011] AtB (2011). AtB Årsrapport 2011. https://www.atb.no/getfile.php/Filer/Diverse%20informasjon/AtB_arsplan_2011.pdf. Accessed: 2014-10-27.

[AtB, 2014a] AtB (2014a). Er alle bussene i Trondheim koblet opp i sanntidssystemet? https://www.atb.no/sanntid/er-alle-bussene-i-trondheim-koblet-opp-i-sanntidssystemet-article9691-395.html. Accessed: 2014-10-27.

[AtB, 2014b] AtB (2014b). Åpne data for sanntidsinformasjon. https://www.atb.no/aapne-data/category419.html. Accessed: 2014-12-06.

[Bangor et al., 2008] Bangor, A., Kortum, P. T., and Miller, J. T. (2008). An Empirical Evaluation of the System Usability Scale. *Intl. Journal of Human–Computer Interaction*, 24(6):574–594.

[Beel and Gipp, 2009] Beel, J. and Gipp, B. (2009). Google Scholar's Ranking Algorithm: An Introductory Overview. In *Proceedings of the 12th International Conference on Scientometrics and Informetrics (ISSI'09)*, volume 1, pages 230–241.

[Brooke, 1996] Brooke, J. (1996). Sus-a quick and dirty usability scale. In Jordan, P. W., Thomas, B., McClelland, I. L., and Weerdmeester, B., editors, *Usability evaluation in industry*, chapter 21, pages 189–194. London: Taylor & Francis.

[Chuttur, 2009] Chuttur, M. (2009). Overview of the Technology Acceptance Model: Origins, Developments and Future Directions. *All Sprouts Content*, 9. Paper 290, http://aisel.aisnet.org/sprouts_all/290.

[Davis, 1989] Davis, F. D. (1989). Perceived Usefulness, Perceived Eease of Uuse, and User Acceptance of Information Technology. *MIS quarterly*, pages 319–340.

[Elgin, 2005] Elgin, B. (2005). Google Buys Android for Its Mobile Arsenal. http://www.webcitation.org/5wk7sIvVb. Accessed: 2014-10-29.

[Gamma, 1995] Gamma, H. (1995). OO Design Patterns. http://www.cs.cornell.edu/Info/Courses/Spring-98/CS211/lectureNotes/10-DesignPatterns.pdf.

[Gao et al., 2014] Gao, S., Krogstie, J., and Siau, K. (2014). Adoption of Mobile Information Services: An Empirical Study. *Mobile Information Systems*, 10(2):147–171.

[Goadrich and Rogers, 2011] Goadrich, M. H. and Rogers, M. P. (2011). Smart Smartphone Development: iOS Versus Android. In *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education*, pages 607–612. ACM.

[Golob et al., 1972] Golob, T. F., Canty, E. T., Gustafson, R. L., and Vitt, J. E. (1972). An Analysis of Consumer Preferences for a Public Transportation System. *Transportation Research*, 6(1):81–102.

[Google, 2014a] Google (2014a). Android Studio. https://developer.android.com/sdk/installing/studio.html. Accessed: 2014-12-04.

[Google, 2014b] Google (2014b). Google Design Guidelines. http://www.google.com/design/spec. Accessed: 2014-12-06.

[Knowles, 2008] Knowles, N. (2008). *SIRI Handbook & Functional Service Diagrams*. Kizoom Limited.

[Mahapatra, 2013] Mahapatra, L. (2013). Android vs. iOS: What's the Most Popular Mobile Operating System in Your Country? http://www.ibtimes.com/android-vs-ios-whats-most-popular-mobile-operating-system-your-country-1464892. Accessed: 2014-10-29.

[Mallat et al., 2008] Mallat, N., Rossi, M., Tuunainen, V. K., and Öörni, A. (2008). An Empirical Investigation of Mobile Ticketing Service Adoption in Public Transportation. *Personal and Ubiquitous Computing*, 12(1):57–65.

[Mane and Khairnar, 2014] Mane, M. P. S. and Khairnar, V. (2014). Analysis of Bus Tracking System Using GPS on Smart Phones. *IOSR Journal of Computer Engineering*, 16.

[Marcussen and Eliassen, 2011] Marcussen, C. J. and Eliassen, L. M. (2011). TABuss: An Intelligent Smartphone Application. Technical report, Department of Computer and Information Science, NTNU.

[Sauro, 2011] Sauro, J. (2011). Measuring Usability With the System Usability Scale (SUS). http://www.measuringu.com/sus.php. Accessed: 2014-12-06.

[Statens Vegvesen Region midt, 2011] Statens Vegvesen Region midt (2011). Tilleggsutredning miljøpakke Trondheim, kollektivtransport. http://miljopakken.no/wp-content/uploads/2012/02/Kollektivtransport-Tilleggsutredning.pdf. Accessed: 2014-10-27.

[Swarco, 2009] Swarco (2009). Public Transport Management, Norway, Trondheim. http://www.swarco.com/en/References/ITS-REFERENCES-LIST/PUBLIC-TRANSPORT-MANAGEMENT,-Norway,-Trondheim-AtB. Accessed: 2014-12-06.

[Swarco, 2014] Swarco (2014). Aktiv prioritering. http://www.swarco.no/Hva-kan-vi-bidra-med/Moderne-byer/Trafikkstyring/AKTIV-PRIORITERING. Accessed: 2014-12-06.

[Venkatesh and Davis, 2000] Venkatesh, V. and Davis, F. D. (2000). A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies. *Management Science*, 46(2):186–204.

[Watkins et al., 2011] Watkins, K. E., Ferris, B., Borning, A., Rutherford, G. S., and Layton, D. (2011). Where Is My Bus? Impact of Mobile real-Time Information on the Perceived and Actual Wait Time of Transit Riders. *Transportation Research Part A: Policy and Practice*, 45(8):839–848.

Appendix A

Terminology

This appendix chapter contains definition of abbreviations used in the report.

API Application Programming Interface

AtB The bus service in Trondheim

BusTUC Bus: The Understanding Computer

FURIOUS Fremtidens Ultimate Intelligente Ruteopplysningssystem (the Futures Ultimate Intelligent Route-Organizing System)

GPS Global Positioning System

GUI Graphical User Interface

HTML HyperText Markup Language

JSON JavaScript Object Notation

NSB Norges Statsbaner AS (Norwegian State Railways)

SDK Software Development Kit

SIRI VM Service Interface for Real Time Information: Vehicle Management

SOAP Simple Object Access Protocol

UTM Universal Transverse Mercator coordinate system

Appendix B

User Survey

This part of the appendix contains the results from the user survey conducted. The data is visualized through bar charts displaying the percentage each answer got, sorted by questions. The questions were asked in the order the bar charts appear in. Not all participants answered all the questions, as this depended on how they answered their questions. For instance, people who do not take the bus are immediately sent to the 'finished" page. The survey consisted of three pages: the introductory page consisting of only one question, the page about bus usage, the page about waiting and finally the page about mobile applications. To get to the last page, the participants had to answer "App" on how they found the bus schedule.
The actual analysis on this survey is in section 4.1. This appendix just contains the number of answers for each question displayed in bar charts.

Introductory Page

The introductory page only consisted of one question to weed out the ones that did not ride the bus.

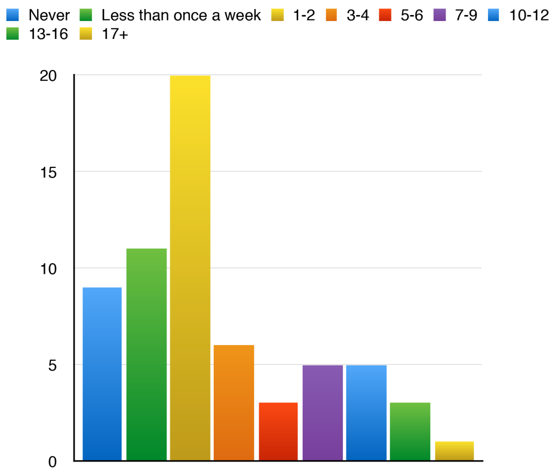

Figure B.1: Question 1
How Many times a Week Do You Ride the Bus?

Bus Usage

This page focused on the specific bus habits of the users and consisted of six questions.

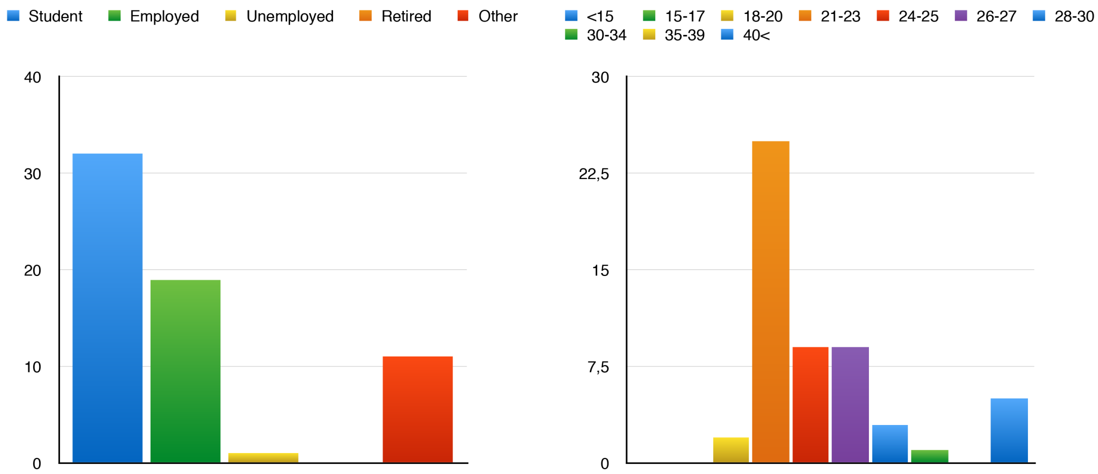

Figure B.2: Question 2 and 3
1) What Is Your Working Status? 2) What Is Your Age?

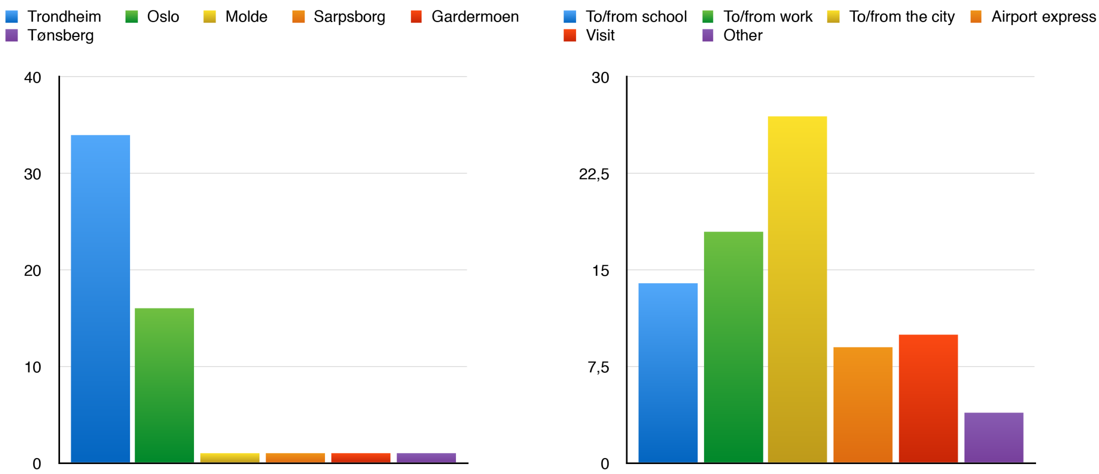

Figure B.3: Question 4 and 5
1) In What City Do You Ride the Bus? 2) In What Context Do You Take the Bus?

The following table (B.1) contains the replies for both the introductory question and the bus usage questions.

Table B.1: Introductory and Bus Usage Questions

Answer nr	**How many times a week do you ride the bus?**	**What is your current working status?**	**How old are you?**	**In which city do you most often take the bus?**	**In what context do you most often take the bus?**
1	1-2	Student	18-20	Trondheim	To/from school, To/from the city, Visit
2	3-4	Student	21-23	Trondheim	To/from school
3	3-4	Student	21-23	Trondheim	To/from the city
4	1-2	Student	21-23	Trondheim	To/from the city, Airport express, Visit
5	1-2	Student	26-27	Trondheim	To/from school
6	3-4	Unemployed	21-23	Trondheim	To/from the city
7	7-9	Student	24-25	Trondheim	To/from school, To/from work, To/from the city
8	1-2	Employed	21-23	Oslo	To/from work
9	1-2	Student	24-25	Trondheim	To/from the city, Airport express
10	1-2	Student	21-23	Oslo	To/from school, To/from work, Visit
11	1-2	Student	21-23	Oslo	To/from the city
12	3-4	Student	21-23	Trondheim	To/from school, To/from work, To/from the city
13	5-6	Employed	24-25	Oslo	To/from work, To/from the city
14	1-2	Student	21-23	Trondheim	To/from the city, Airport express, Visit
15	1-2	Student	18-20	oslo	To/from school
16	1-2	Employed	28-30	Oslo	To/from work
17	7-9	Employed	40<	Oslo	To/from work
18	1-2	Running a business	40<	Oslo	To/from the city
19	1-2	Student	24-25	Trondheim	To/from the city, Airport express
20	1-2	Student	26-27	Trondheim	To/from the city
21	10-12	Student	21-23	Tønsberg	To/from school, To/from the city
22	13-16	Student	21-23	Oslo	To/from school
23	7-9	Employed	26-27	Oslo	To/from work, To/from the city, Airport express, Visit
24	10-12	Employed	40<	oslo	To/from work
25	7-9	Student	21-23	Trondheim	To/from school, To/from the city
26	3-4	Student + part time job	24-25	Trondheim	To/from work, To/from the city, Airport express
27	1-2	Student	21-23	Trondheim	Airport express
28	Less than once a week	Student	26-27	Oslo	To/from work

29	1-2	Student	24-25	trondheim	Visit
30	10-12	Employed	24-25	Oslo	To/from work
31	10-12	Employed	24-25	Oslo	To/from work, To/from the city, Visit
32	1-2	Student	21-23	Trondheim	To/from the city
33	10-12	Student	21-23	Trondheim	To/from school
34	Less than once a week	Student	21-23	Trondheim	To/from the city, Visit
35	17+	Employed	40<	oslo	To/from work
36	Less than once a week	Student	24-25	Trondheim	To/from the city
37	1-2	Student	21-23	Trondheim	To/from the city, Airport express
38	5-6	Employed	26-27	Trondheim	To/from work, Visit
39	Less than once a week	Student	21-23	Oslo	Home to parents every other week
40	1-2	Student	21-23	Trondheim	Workout
41	Less than once a week	Employed	28-30	Trondheim	To/from the city
42	5-6	Student	21-23	Sarpsborg	To/from school, To/from the city
43	Less than once a week	Employed	30-34	Trondheim	To/from the city
44	13-16	Employed	26-27	Trondheim	To/from work
45	Less than once a week	Student	21-23	Trondheim	To/from the city
46	1-2	Employed	26-27	Trondheim	To/from the city
47	13-16	Student	26-27	Trondheim	To/from school
48	3-4	Employed	26-27	Trondheim	To/from work
49	Less than once a week	Employed	28-30	Gardermoen	Airport - Hotel
50	1-2	Student	21-23	Trondheim	Workout
51	Less than once a week	Employed	21-23	Trondheim	Visit
52	7-9	Student	21-23	Trondheim	To/from school
53	Less than once a week	Employed	21-23	Trondheim	Airport express
54	Less than once a week	Employed	40<	Molde	To/from work
55	Less than once a week	Employed	18-20	Trondheim	Airport express

Waiting on the Bus

This page focused on the waiting. Since we want to solve, or improve peoples waiting time, the survey had five questions that focused on this.

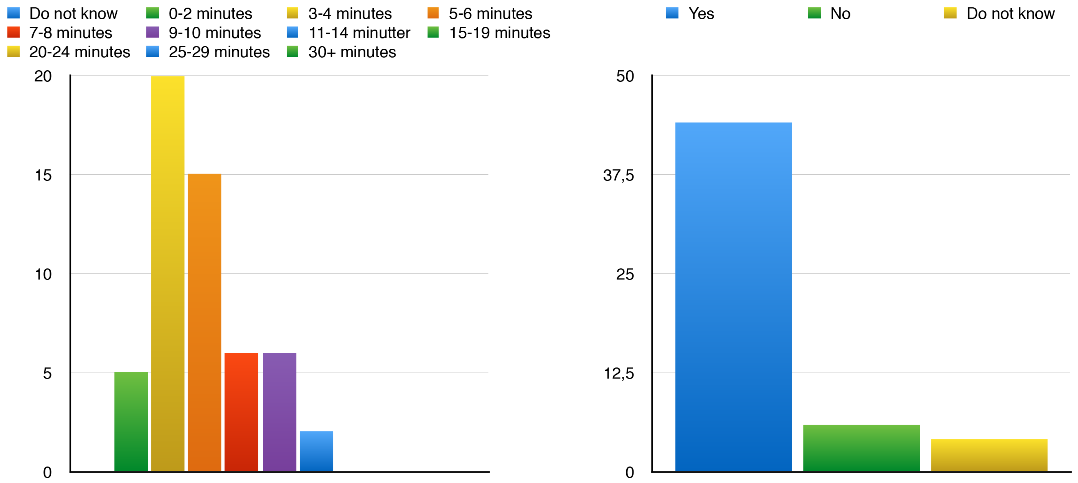

Figure B.4: Question 6 and 7
1) How Long to Do You Wait for the Bus, on Average? 2) Do You Think Knowing Where the Bus Is Located Will Help You Accomplish Shorter Waiting Time?

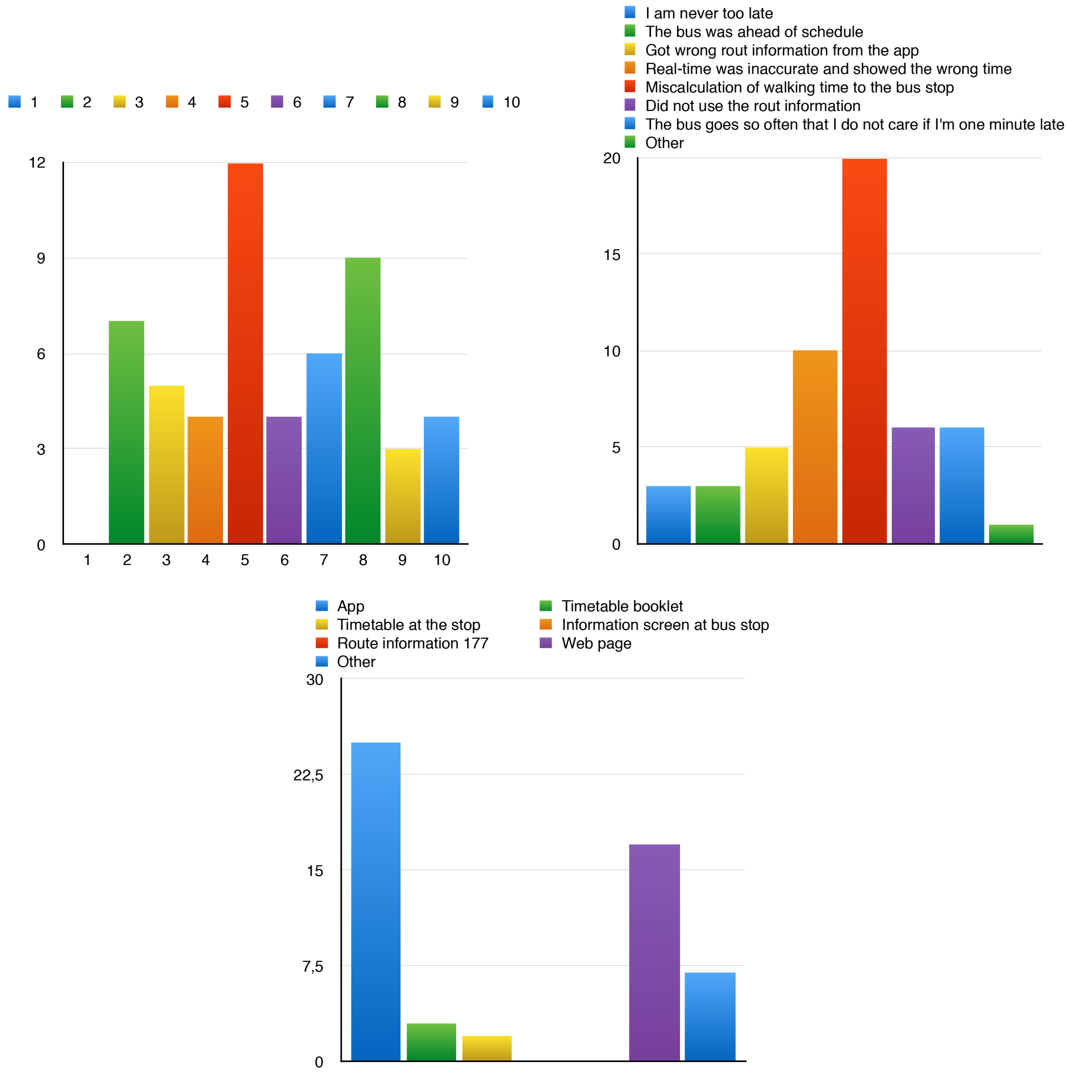

Figure B.5: Question 8, 9 and 10
1) on a Scale from 1-10, Where 10 Is Super Annoyed, How Annoyed Does You Get Waiting on the Bus? 2) If You Are Too Late for the Bus, What Is the Reason?, 3) How Do You Retrieve the Bus Schedule?

The following table (B.2)shows what the participants answered on these questions.

Table B.2: Waiting on the Bus

Answer nr	**How long do you tend, on average, to wait for the bus?**	**Do you think information on where the bus is located, can help you achieving shorter waiting time?**	**On a scale of 1 to 10, where 10 is very irritated, how annoyed are you of waiting for the bus?**	**Which service do you use primarily to find your bus route?**	**If you are late for the bus, what is the reason?**
1	5-6 minutes	Yes	6	App	Got wrong route information from the app
2	0-2 minutes	Yes	8	Route map at stop	The bus departs so often that I do not care if I'm one minute late
3	3-4 minutes	Yes	7	App	I am never too late
4	7-8 minutes	Yes	9	App	Miscalculation of walking time to the bus stop
5	5-6 minutes	Yes	8	App	Real-time was inaccurate and showed the wrong time
6	5-6 minutes	Yes	3	Route map at stop	Did not use the route information
7	5-6 minutes	Do not know	5	App	Miscalculation of walking time to the bus stop
8	3-4 minutes	No	2	Information screen at stop	The bus departs so often that I do not care if I'm one minute late
9	9-10 minutes	Yes	3	Information screen at stop	Did not use the route information
10	5-6 minutes	Yes	10	App	Real-time was inaccurate and showed the wrong time
11	3-4 minutes	Yes	4	App	Real-time was inaccurate and showed the wrong time
12	3-4 minutes	Yes	5	App	Miscalculation of walking time to the bus stop
13	3-4 minutes	Yes	5	App	Real-time was inaccurate and showed the wrong time
14	7-8 minutes	Yes	4	App	Got wrong route information from the app

15	3-4 minutes	Yes	2	Web page	Real-time was inaccurate and showed the wrong time
16	3-4 minutes	Yes	2	App	Miscalculation of walking time to the bus stop
17	5-6 minutes	No	2	App	Miscalculation of walking time to the bus stop
18	9-10 minutes	Yes	8	App	Real-time was inaccurate and showed the wrong time
19	3-4 minutes	Yes	2	Web page	The bus departs so often that I do not care if I'm one minute late
20	9-10 minutes	Yes	8	Web page	Real-time was inaccurate and showed the wrong time
21	7-8 minutes	Do not know	5	App	Did not use the route information
22	3-4 minutes	Yes	3	Web page	Did not use the route information
23	3-4 minutes	Yes	8	Web page	Miscalculation of walking time to the bus stop
24	0-2 minutes	Yes	5	App	The bus departs so often that I do not care if I'm one minute late
25	3-4 minutes	Yes	5	App	Got wrong route information from the app
26	7-8 minutes	Yes	8	App	Got wrong route information from the app
27	7-8 minutes	Yes	7	App	The bus was ahead of schedule
28	0-2 minutes	No	3	Web page	The bus departs so often that I do not care if I'm one minute late
29	5-6 minutes	Yes	7	Web page	Miscalculation of walking time to the bus stop
30	3-4 minutes	Yes	5	App	Miscalculation of walking time to the bus stop
31	3-4 minutes	Yes	4	App	Miscalculation of walking time to the bus stop

32	5-6 minutes	Yes	7	App	Real-time was inaccurate and showed the wrong time
33	0-2 minutes	No	8	App	Real-time was inaccurate and showed the wrong time
34	9-10 minutes	Yes	9	Web page	Got wrong route information from the app
35	7-8 minutes	Yes	10	Timetable booklet	Miscalculation of walking time to the bus stop
36	3-4 minutes	Yes	6	App	I was slow out the door
37	5-6 minutes	Yes	4	App	Miscalculation of walking time to the bus stop
38	5-6 minutes	Yes	6	Information screen at stop	Miscalculation of walking time to the bus stop
39	11-14 minutes	No	9	Web page	Miscalculation of walking time to the bus stop
40	3-4 minutes	Do not know	6	Web page	The bus was ahead of schedule
41	11-14 minutes	Yes	5	Web page	I am never too late
42	9-10 minutes	Yes	10	Web page	The bus was ahead of schedule
43	5-6 minutes	Yes	8	Web page	Miscalculation of walking time to the bus stop
44	3-4 minutes	Yes	5	Information screen at stop	Miscalculation of walking time to the bus stop
45	3-4 minutes	Do not know	2	Web page	Did not use the route information
46	5-6 minutes	Yes	7	Timetable booklet	Miscalculation of walking time to the bus stop
47	3-4 minutes	Yes	7	Information screen at stop	Real-time was inaccurate and showed the wrong time
48	9-10 minutes	Yes	10	App	I am never too late
49	5-6 minutes	Yes	5	Information screen at stop	Did not use the route information
50	5-6 minutes	No	3	App	The bus departs so often that I do not care if I'm one minute late
51	5-6 minutes	Yes	2	Web page	Miscalculation of walking time to the bus stop

52	3-4 minutes	Yes	8	Timetable booklet	Miscalculation of walking time to the bus stop
53	0-2 minutes	Yes	5	Web page	Miscalculation of walking time to the bus stop
54	3-4 minutes	Yes	5	Web page	Miscalculation of walking time to the bus stop
55	Do not know	Yes	10	App	The bus was ahead of schedule

App Questions

If the user answered that he or she used mobile application to find the bus schedule, he or she had to answer this page as well. By making these questions a picture could be made around how happy people where with the current situation, and how the applications worked for them. With this information, it might be possible to create an application that is even better than the existing ones. When finishing the survey, the users got the opportunity to write what else they thought of their application of choice. This information will be essential when drawing inspiration from other applications.

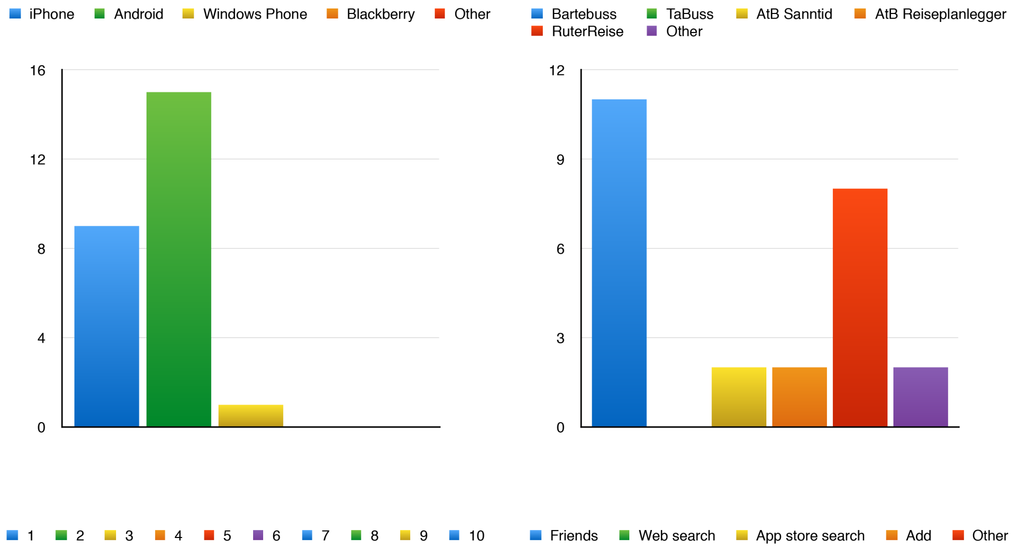

Figure B.6: Question 11 and 12
1) What Phone Do You Use? 2) What Application Do You Use?

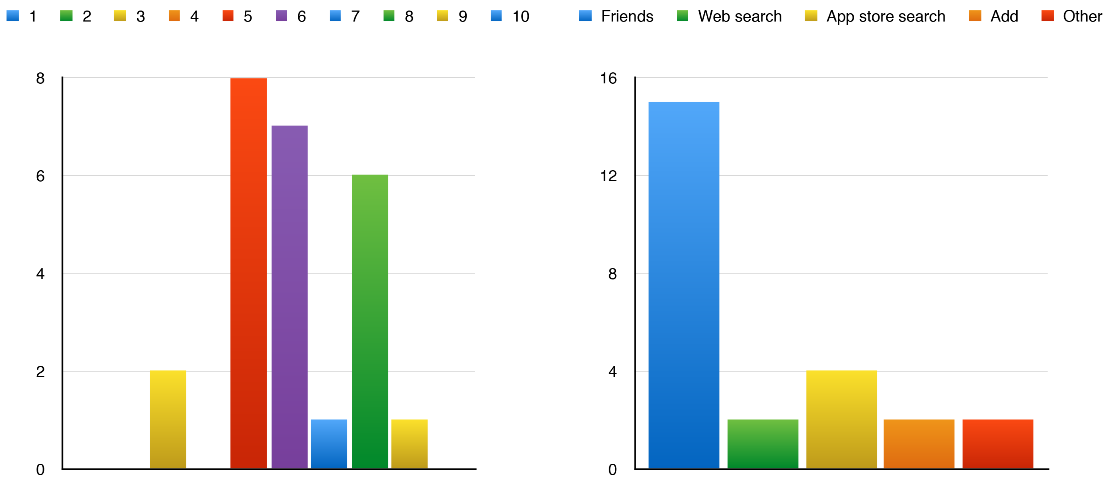

Figure B.7: Question 13 and 14
1) On a Scale from 1-10, Where 10 Is Super Happy, How Happy Are You with the Application?
2) How Did You Discover the Application?

The following table (B.3) show how the participants, who answered they used the application the most, replied on the application questions.

Table B.3: App Questions

Answer nr	**What phone do you have?**	**Which application do you use the most?**	**On a scale of 1 to 10, where 10 is super satisfied, how satisfied are you with the app?**	**How did you hear about the app?**
1	iPhone	Bartebuss	8	Friends
3	iPhone	Bartebuss	5	Friends
5	iPhone	Bartebuss	6	Friends
7	iPhone	Bartebuss	6	Friends
10	Android	RuterReise	7	Do not know
11	iPhone	Bartebuss	3	Friends
12	Android	Bartebuss	5	Friends
13	iPhone	RuterReise	5	Appstore search
14	Android	RuterReise	5	Friends
16	Windows phone	Trine i farta	8	Friends
17	Android	RuterReise	8	Friends

18	Android	RuterReise	5	Web search
21	Android	VKT	6	Web search
24	Android	RuterReise	9	Appstore search
25	Android	AtB sanntid	5	Add
26	Android	Bartebuss	8	Friends
27	iPhone	Bartebuss	6	Friends
30	Android	RuterReise	3	Appstore search
31	iPhone	RuterReise	6	Add
32	iPhone	Bartebuss	8	Friends
33	Android	Bartebuss	8	Friends
36	Android	AtB reise-planlegger	6	Appstore search
37	Android	AtB reise-planlegger	5	?
48	Android	AtB sanntid	6	Friends
50	Android	Bartebuss	5	Friends
55	iPhone	Bartebuss	4	Friends

Some participants answered an additional step: **(Optional) Describe your experience with the app**. These are the answeres we got from the numbered participants.

- **#10 RuterReise** It needs a lot more features, such as being able to see the entire bus route (with stops) when you press a route.
- **#17 RuterReise** When you scale from 1 to 10, it should say what is worst, best, etc.. Such as the previous question. Eg. the one about furstation could easily be clearified.
- **#24 RuterReise** Very good! Definitely a "must have" app.
- **#26 Bartebuss** Very good experience with the app, particularly fond of the UI and the way information is displayed. Could have been better at updating time when it comes to major delays, but expect that some of this lies with AtB's real tables ...
- **#30 RuterReise** The way developers think I use the app is quite banal. For example cumbersome to find the bus from A to B, but easy to find when the next bus passes. When the next bus passes I do not care if I have to wait 20 minutes at the bus exchange.
- **#33 Bartebuss** The app frequently change the time the bus will arrive when it is approaching, so it comes sooner than you think. This makes it difficult to calculate when to go home.
- **#55 Bartebuss** Cumbersome to identify the timetable for a single bus, should have been a function to select the bus you want and not just a menu for bus stops you want.