Design process of an Android application for measuring and displaying road conditions.

Tobias Andersson and Tim Nilsson

DEPARTMENT OF DESIGN SCIENCES
FACULTY OF ENGINEERING LTH | LUND UNIVERSITY
2016

MASTER THESIS

Design process of an Android application for measuring and displaying road conditions

Tobias Andersson and Tim Nilsson

LUND UNIVERSITY

Design process of an Android application for measuring and displaying road conditions

Published by

Department of Design Sciences
Faculty of Engineering LTH, Lund University
P.O. Box 118, SE-221 00 Lund, Sweden

Subject: Interaction Design (MAMM01)
Division: Certec
Supervisor: Kirsten Rassmus-Gröhn
Co-supervisor: Günter Alce
Examiner: Joakim Eriksson

Abstract

The status of our roads and the possibility to monitor the conditions of them are becoming more and more important due to the steady increase of vehicles populating them. This thesis aims to improve a system for which purpose is to gather data about the status of roads, which the Swedish company Klimator develops. With more frequently measured roads salt- and plowing efforts could be optimized and individual consumers would be able to know if a road is suitable for driving.

The main component in Klimator's system is a laser sensor that measures a road's current status by directing three laser beams towards the surface and measuring their individual responses. Before this thesis the laser sensor was connected by physical wires to a logger box. The authors were to design and prototype an Android application to explore the possibility to replace this hardware, add Bluetooth capability and add a new layer of functions above, especially one which displayed the current slipping hazard.

All throughout the design and prototype process of the application interaction design was in focus, to provide it with an enhanced user experience. The main issue which was considered and managed, was how to give the driver information about the current slipping hazard without drawing the driver's attention away from the driving. Improvements on the old system were achieved and also reflections on how Klimator could potentially continue the development of the project, i.e. highlighting issues and unimplemented but interesting functions.

Keywords : Interaction design, Android, Internet of things, Slipping Hazard, Environment, Extreme programming

Sammanfattning

Statusen för våra vägar och möjligheten att övervaka deras förhållande blir allt viktigare på grund av den stadiga ökningen av fordon som trafikerar dem. Denna avhandling syftar till att förbättra ett system för vars syfte är att samla in data som beskriver tillståndet av våra vägar, som det svenska företaget Klimator utvecklar. Om vägarnas status kan mätas oftare så kan salt- och plognings ansträngningar optimeras och enskilda konsumenter skulle kunna veta om en väg är lämplig för körning.

Den viktigaste komponenten i detta system är en lasersensor som mäter en vägs nuvarande status genom att skicka ut tre laserstrålar mot ytan och mäta deras individuella reflektans. Innan denna avhandling anslöts lasersensorn med fysiska kablar till dedikerad hårdvara. Författarna har designat en Android-applikation för att undersöka möjligheten att ersätta denna hårdvara, lägga till Bluetooth funktionalitet och annan lämplig funktionalitet, speciellt en som kan visa upp den nuvarande halkrisken.

Under projektets gång så har interaktionsdesign var i fokus, för att kunna skapa en bra användarupplevelse. Den viktigaste frågan som hanterades var hur man skulle ge föraren information om den aktuella halkrisken utan att dra förarens uppmärksamhet från körningen. Förbättringar på det gamla systemet uppnåddes och även reflektioner kring hur Klimator skulle kunna fortsätta utvecklingen av projektet, dvs belysa problem som fortfarande existerar och lyfta fram funtioner som borde förbättra applikationen.

Nyckelord : Interaktionsdesign, Android, Internet of things, Halkrisk, Miljö, Extreme programming

Acknowledgements
We would like to thank the entire Klimator organization for the giving us the opportunity to do our Master's thesis on their behalf. Also Miljöbron and LTH's department of Design need to be thanked. Without our supervisors, Johan Edblad, Kirsten Rassmus-Gröhn and Günter Alce, this project would never have been successful. We also would like to thank our families and friends for their everlasting support.
Lund, June 2016
Tobias Andersson, Tim Nilsson

List of abbreviations and acronyms

- The box = The power converter and the Bluetooth adapter
- The system = the laser sensor, the Android tablet and the box
- The logging system = the Android tablet and the box
- The logger box = a large box containing hardware to function as the link between the laser sensor and the server
- IoT = Internet of things
- Bluetooth BR/EDR = Basic Rate/Enhanced Data Rate
- LMP = Link Management Protocol
- L2CAP = Logical Link Control and Adaptation Protocol
- SDP = Service Discovery Protocol
- HCI = Host controller interface
- RFCOMM = Radio Frequency Communications
- OS = operating system
- XP = extreme programming
- API = application programming interface
- RS232 = a standard for serial communication transmission of data
- IDE = Integrated Development Environment
- NDK = Native Development Kit
- Regex = Regular Expression
- XML = Extensible Markup Language

Contents

1 Introduction

It has been snowing ridiculous amounts the last week, but the temperature has been quite varying. It is morning and last night the temperature reached below zero degrees Celsius. As follows, you believe that some roads will be covered by ice, at least partially. But which roads?

1.1 Background

The benefits of being able to gather detailed information concerning the state of roads are both safety and environmental related. A road's slipping hazard could potentially create an accident prone situation, since the probable danger which lies within a road's slipping hazard might not be fully visible for the everyday driver. If a driver would be able to acquire the current state and/or the forecast of any road, it would be possible for the driver to determine if a certain road is suitable for the driver's upcoming route.

By gathering data regarding the state of roads, salt- and plowing efforts could be further optimized to decrease their environmental impact. No unnecessary salt deployment would occur, but any road/stretch of road with a too severe slipping hazard would be managed properly.

To allow for data gathering, the authors have been provided with an already functioning laser sensor. However, the sensor's current set up and necessary equipment, both in terms of cost and size, impede on its distribution.

The subject of interest for this Master's thesis was split in two, to aid in the optimization of salt- and plowing efforts by simplifying data gathering, and to display the measured data to a user. This Master's thesis was conducted at the department of Design at LTH and in conjunction with the company Klimator.

1.2 Aim of the thesis

The aim of this Master's thesis was to improve upon the system surrounding the the laser sensor that the company Klimator produces. The aforementioned sensor offers a service for measuring and approximating the road condition (i.e. ice, snow, wet or dry, but also a friction coefficient).

The setup before this Master's thesis included two components, the laser sensor and the logger box. The logger box was coupled to the laser sensor by physical wires and was able to map input data from the laser sensor to a gps location and then distribute this gathered data to a server, however the problem with it was its size, its clunkiness, its array of expensive hardware and that it also lacked the capability to display output to the user. Thus, the aim was to investigate the possibility to get rid of the logger box by designing and implementing an Android application and introducing Bluetooth functionality into the system through a RS-232-to-Bluetooth adapter.

The customer Klimator also presented an idea where they wanted to display the friction of a road to the driver while driving in four levels where level 1 was not slippery at all and level 4 was extremely slippery. This gave the authors a possibility to focus on the interaction design and how one needs to think in this specific situation.

The project was built around the following subtasks:

The authors:

- An investigation to determine RS-232-to-Bluetooth adapter and Android device to be recommended for an end product.
- Maintain the functionality that the logger box previously possessed.
- Design a user interface following known design guidelines and principles.
- Implement design suggestions for displaying the slipping hazard.
- Envision new functions that might be beneficial for the system and Klimator as a company.

Klimator:

- Design of a server for receiving and processing the laser sensor related data.

The metal tube, which makes up for the majority of the laser's length see figure 4, is actually only for protection against environmental hazards. The laser emitters and the photo diode is only situated in the snuffbox-like cylinder in the top. When the laser sensor is powered up it automatically starts its process of sending data packets via Bluetooth. The design of a data packet can be seen in figure 1.

Currently there exists no public framework on how to actually mount the sensor to a vehicle, aside from that the sensor is required to be set up as figure 4 entails to achieve optimal performance. A suggestion for attachment is to modify a bike holder to secure the sensor to. The box containing the Bluetooth adapter and power converter will have to be coupled to the sensor via a cable, e.g. in a car this box could conveniently be placed in its trunk.

```
S, 09237, 11080, 05291, 29
```

Figure 1: Laser data packet: State, L1, L2, L3, sensor's temperature

For other vehicles, which lacks a trunk, e.g. buses and trucks, a different solution had to be found. But buses and trucks possess a capacity for which a car does not, i.e. a lot of hull and undercarriage surfaces. This simplifies the issue massively, the sensor and the box could easily be bolted or attached to any of this excess surfaces without hindering the vehicle's abilities or its appearance.

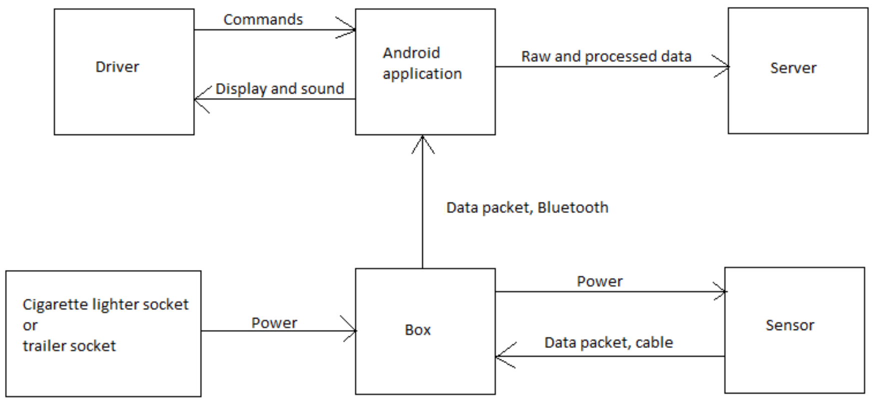

Figure 2: Set up relationships inside vehicle. Box includes a power converter and an RS-232-to-Bluetooth adapter

In all cases, the Android tablet will just be placed in the cab of the vehicle with a car mount. Figure 2 showcases the relationships of the different components and the driver for an example vehicle set up.

1.3 Stakeholders

To allow for the reader to gain a grasp of all the involved stakeholders, these are thus defined here and their respective relationship between each other.

1.3.1 The authors

The authors are both Master's thesis students at LTH within the field of Information- and Communications Technology.

1.3.2 Klimator

As described on their website, Klimator's main field of business is forecasts of road conditions [2]. The objective of the forecasts are to provide data, for which salt- and plowing efforts can be optimized. They are also the owners of the aforementioned laser sensor and all of its specifics.

1.3.3 Miljöbron Skåne

Miljöbron Skåne is a non-profit organization which focuses on linking companies with potential projects to students with the adequate knowledge to handle the aforementioned projects [3]. The organization only manages projects which have some form of relation to the environment and sustainable development.

1.3.4 Future users

Table 1 displays the possible future stakeholders which were identified at the starting point of the project by the authors and the main contact at Klimator, Johan Edblad and what kind of data that was of interest for each individual group. It was decided that Road maintenance drivers where of most interest, since they could both gather the data and use the data to improve their work. Hence the authors decided to focus on the company Svevia which already was an existing customer to Klimator.

Svevia Svevia are the leading company when dealing with maintenance of the roads in Sweden [4]. The maintenance includes aspects such as applying new asphalt and adjusting the current surface status e.g. salting and plowing.

Table 1: Future potential stakeholders and their individual needs

Users	**Unprocessed raw data**	**Friction**	**Verdict of processed raw data**	**Route planning**
Road maintenance drivers	-	x	x	x
Bus drivers	-	-	x	-
Cab drivers	-	-	x	x
Fleet owners	-	x	x	x
Private individual	-	-	x	x
Scientists	x	x	x	-

1.3.5 Stakeholder relations

The relations between the relevant stakeholders in the project are denoted A, B, C and D in figure 3.

- **A** Klimator focuses on distributing the laser sensor combined with a system enabling logging and transmission of data to a server. This end product was of interest for Svevia since this could improve their salting and plowing efforts around Sweden.
- **B** Klimator used the third party, Miljöbron, to find two Master's students with the adequate knowledge to improve the system surrounding the laser sensor.
- **C** Miljöbron recruited the authors and set up the contact between the authors and Klimator. Miljöbron also contribute through out the process by proofreading the authors report.
- **D** The authors and Klimator worked with the design process. The authors worked with the design and continuously got it revised by Klimator.

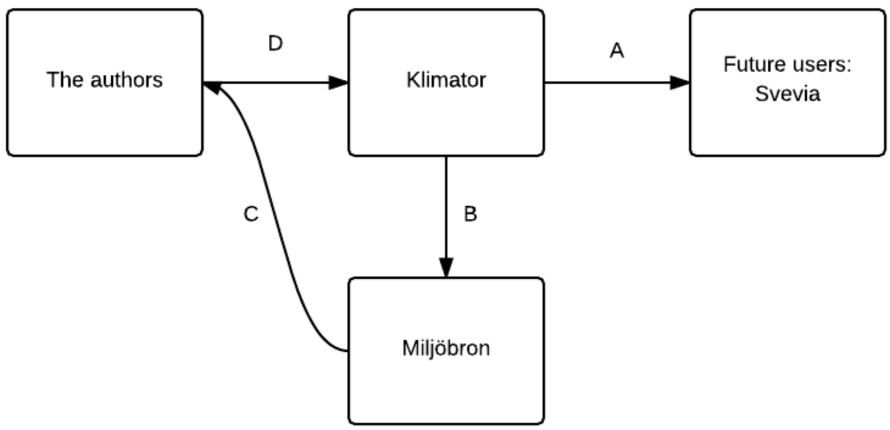

Figure 3: Displays the relations between each stakeholder

1.4 The laser sensor

Casselgren writes in the article, *Intelligent Road*, that the laser sensor emits three laser beams L1, L2 and L3, which each one has a different wavelength (L1=1550, L2=1310 and L3=980 (nm)) [1]. As the beams are reflected against the targeted surface the laser sensor measures their respective level of response with a photo diode. The targeted surface is approximately 2*2 cm. Four different states are used to describe the road condition: D, W, S and I (dry, wet, snow and ice). As of current, the laser sensor possess a sampling rate of approximately 20 Hz.

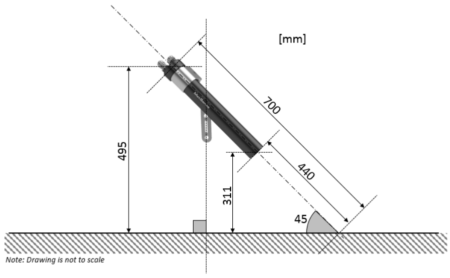

Figure 4: The optimal set up for the laser sensor

1.5 Technical background

Background knowledge regarding used tools and practices are presented here.

1.5.1 Bluetooth

Miller writes in *Bluetooth revealed: the insider's guide to an open specification for global wireless communication* that Bluetooth is a standard for wireless communication over a fixed distance and it is widely used in the IoT-field (Internet of things) [5]. Currently Bluetooth is categorized into two main groups, Bluetooth BR/EDR(Basic Rate/Enhanced Data Rate) and Bluetooth Low Energy. Bluetooth BR/EDR is primarily used in speakers and headphones, whereas Bluetooth Low Energy can be found in the type of devices which have a limited amount of electrical energy, e.g. fitness related wristbands.

The Bluetooth protocol defines some mandatory sub protocols which needs to be used for the Bluetooth functionality to work as intended. These are: LMP(Link Management Protocol), L2CAP(Logical Link Control and Adaptation Protocol) and SDP(Service Discovery Protocol) [6].

- LMP, the main focus of this protocol is to uphold and maintain the communication between two connected units [6].
- L2CAP, this protocol manages the segmentation of data which a sender wishes to send to a recipient but also the reassembly on the recipient side [6].
- SDP, which enables the crucial functionality of discovering Bluetooth units in the vicinity [6].

In addition two other protocols are widely implemented in the current Bluetooth units on the market, these are HCI(Host controller interface) and RFCOMM(Radio Frequency Communications).

- HCI, which handles the internal communication between the Bluetooth specifics on a unit with the actual host, i.e. the OS(operating system) of the host phone [6].
- RFCOMM, enables emulation of RS232-ports [6].

The distance between two Bluetooth units which wishes to communicate with each other differs, this is due to the different classifications of Bluetooth [6]. Currently there are three categories: class 1, class 2 and class 3. Class 1 has a theoretical max communicating distance of 100m, class 2 10m and class 3 1m. The majority of the products supporting Bluetooth uses class 2.

1.5.2 RS232

RS232 is a standard for serial communication in the field of telecommunications. This standard is a way of defining the characteristics of a signal which one wishes to transmit, e.g. how the timing of signals should be managed [7].

Using serial communication all bits which one wishes to transmit are sent in the correct order over the exact same instance across a link [7]. The contrast between serial communication and parallel communication is complexity. Parallel communication requires multiple channels and as followed also the need of synchronization, but the serial communication only requires one channel. The bite rates of serial communication will not be as good as with parallel, but the level of complexity is lower and which results in a cheaper and simpler technology.

1.5.3 Android

Android is an OS widely spread in products such as smartphones and tablets [8]. As of 2015 it controls 82.8% of the smartphone market [9]. The OS is developed by Google and is based upon a Linux kernel. In its true nature Android is open source, but in the real world each device possesses a fusion of open source and private code.

The framework which Android brings to application developers is the main factor of its success. Because, the array of ready made functionality and libraries is vast, and creates an environment which is easy, cheap and quick to develop in.

Android have a natural split of logic and layout, all interface designs in Android are defined in XML (Extensible markup language) sheets while the logic is defined in java classes and in activities/fragments controlling and changing the states of layout objects [10]. This invites developers to program their code in an even more object modulated manner.

1.5.4 Android Studio

An IDE (Integrated Development Environment), as described by JetBrains, is a tool to aid the use of programming by adding a user friendly interface with a multitude of built-in-tools, frameworks, debugging support, connecting plugins and etc, in short an IDE is used to maximize the development potential of a code [11].

The most commonly used IDE for Android development is Android Studio and it is based on Intellij IDEA software [12]. But even so, Android Studio has built-in-support, i.e. a NDK (Native Development Kit), for constructing Android applications with other native programming languages, e.g. C and C++. Before Google launched Android Studio in 2013, the IDE primarily used by Google for Android application development was Eclipse. Some of the most notable key points which Android Studio brings to the development world are a Gradle-based build system (build tool), NDK support, GitHub Integration and additional tools/frameworks.

1.5.5 Github

Version control is a concept to ease the development of software and GitHub can be used to satisfy this purpose [13]. GitHub is actually an additional layer of user friendly interaction above Git, whereas Git is the actual version control system.

The benefits with version control system in general are that developers are able to track the progress of the software which they work on [13]. There also exists the possibility for a development team to proceed in different branches, to roll back to an earlier version of the code and etc.

1.5.6 Programming structures

The website *Sourcemaking* argues that the reasons to use programming patterns and structures are to provide code with better readability, learn-ability and to speed up development [14]. Thus, someone unfamiliar with this particular source code, but familiar with programming, can easier grasp and take over the management of another developer's code. Also, major programming issues can be caught in the earlier stages of the development.

Singleton At times in programming it is desirable to only have one instance of a certain object [15]. The Singleton-pattern provides this functionality, to ensure that the Singleton-pattern is implemented as intended, the following checklist can be used:

1. Define each object which one wishes to be use as a Singleton as a private static-attribute.
2. Define a public static accessor method in the class.
3. Initiate Singleton-objects in accessor function.
4. Define constructors as protected or private.
5. Only the accessor method can be used to change the Singleton-object.

Observer-pattern The Observer-pattern is a part of the Model-View-Controller hierarchy where one class listens for changes in another [16]. A correct implementation of this pattern in Java is executed by letting a class denoted Model containing the model to extend the class Observable and a class denoted View that wants to use changed data in Model-class to implement the interface Observer. Then one have to bind the Model- and the View-object by the following syntax:

```
Model model = new Model();
View view = new View();
model.addObserver(view);
```

Inside the Model-class below where one wishes to notify the observing View-class the following two lines is needed:

```
setChanged();
notifyObservers();
```

Inside the View-class one has to overwrite a method update(), that is triggered when the notifyObservers()-method is called in the Model-class:

```
public void update(){
//handle changes in Model
}
```

Single Responsibility Principle-pattern Every single class should have one area of responsibility [17]. Responsibility in this case means handling all actions on one specific type of object. If a class have more than one responsibility area it generates fragility, meaning that it is hard to add or remove parts without it affecting the rest of the code. On the other hand, a class without any area of responsibility is redundant.

Regular expression In Android and Java, verifying the correctness of an input one can utilize a regular expression [18]. By correctness, both corrupted versions of the input are found and inputs which possesses illegal characters. This will allow for the code to avoid unnecessary exceptions and add a layer of safety.

The programmer defines a Pattern-object for which characters are allowed and at what locations in the input. An object of type Matcher is used to check if the given input matches the predetermined Pattern-object.

Location awareness One of the methods to achieve location awareness in Android is to implement the Google Map API (application programming interface) [19]. The community of Android development strongly advise one to use the Google API instead of the Android framework location API. This will require the application to be coupled with Google Play services, to enable the functionality, and the application to implement the necessary interfaces.

Parcelable To ease the sharing and management of an object from a custom built type in Android, this class can be made parcelable [20]. By making a class parcelable, the values of variables in that instant of the type are saved in a Parcel and then unpacked after the transfer. This allows the instant to maintain its state.

1.5.7 Crowdsourcing

Crowdsourcing is a concept when a work load is outsourced to a large crowd of people, instead of using a traditional crowd of employees [21]. The term can be applied in this project in the way that road conditions are crowdsourced, i.e. that road conditions are measured by a crowd of people, who are all automatically working towards a mutual goal. Any person who owns the laser sensor will not be an employee, but they will still add their sensor data to the community.

1.6 Related work

Klimator has two main competitors that also produce systems for measuring and displaying road conditions, these are Lufft and Teconer.

1.6.1 Lufft

The main difference between Klimator's sensor and Lufft's sensor MARWIS, is complexity [22]. An image of the application which is connected to the sensor can be seen in figure 5. Since, their sensor is constructed to measure more types of data than the laser sensor, Lufft's application will always need to be more complex than Klimator's.

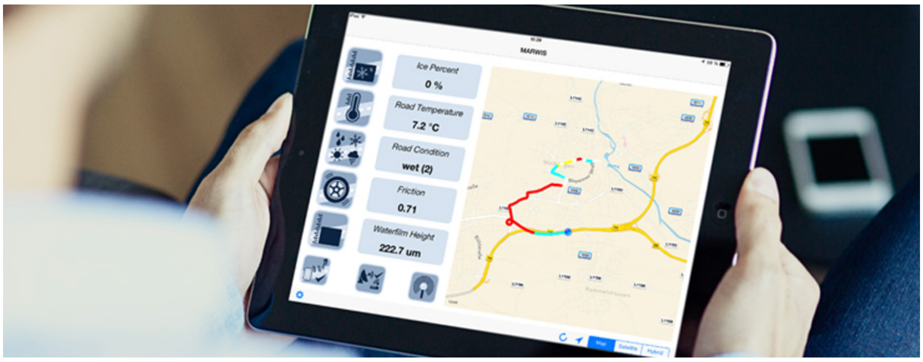

Figure 5: The application which is connected to the MARWIS sensor.

The MARWIS sensor is capable of measuring more types of data and its measure frequency lands at a 100 times per second as opposed to Klimator's system which measures 10 times per second. The array of measure types are temperature above road, relative humidity above road, surface temperature and road condition. The creators of MARWIS also boasts that their sensor is very robust and able to function even in extreme environment. But all these functions and the robustness makes the MARWIS sensor a much clunkier piece of hardware and by this more complex to attach to a vehicle then the laser sensor relative to this project as can be seen in figure 6.

Figure 6: Attachment of the MARWIS sensor to a car.

1.6.2 Teconer

The application which is coupled to Teconer's RCM411 sensor can be seen in figure 7. The figure shows that the application is meant for advanced users, since to grasp what happens on the screen one needs to know how to read the data that is on display [23].

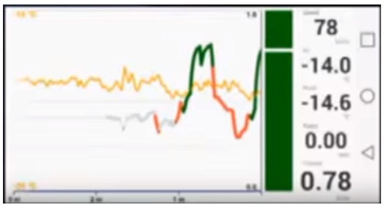

Figure 7: The application which is connected to the RCM411 sensor.

The RCM411 sensor, figure 8 displays the sensor, is very alike Klimator's laser sensor, both in appearance and function. As both the sensors utilizes a spectrum analysis, with an optical sensor, to evaluate the road condition through the reflective signals, i.e. the slipping hazard.

Figure 8: Attachment of the RCM411 sensor to a car.

2 Methods

The methods used in this project are stated and explained below. Figure 9 describes the main three phases of the project: the Pre-studies, the Implementation and the User testing.

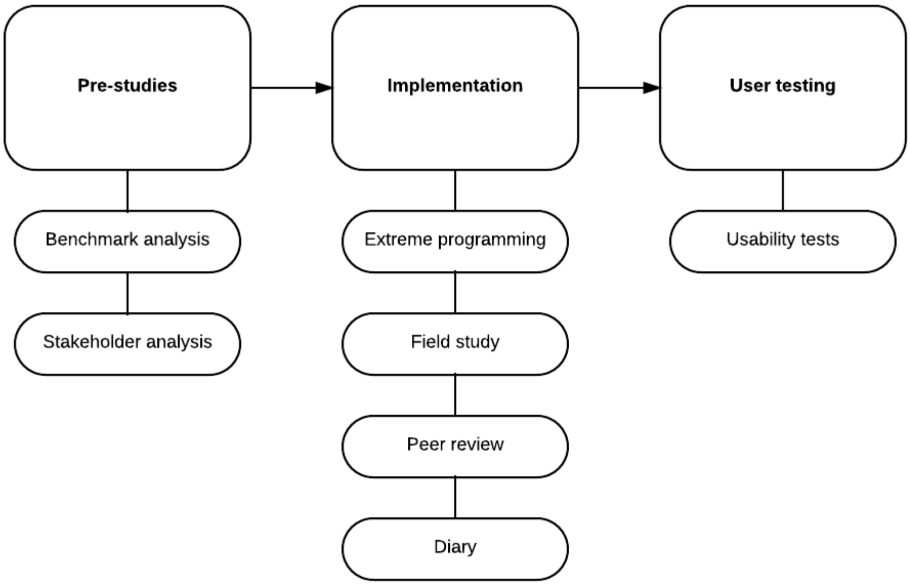

Figure 9: An overview of the main three phases of this work.

Firstly the authors handled the Pre-studies and when that was completed the Implementation phase was initiated, this was the most time consuming phase. Lastly the User testing was performed.

2.1 Theoretical background

To allow for the design of the application to reach a better experience for the user, methods for user-centered designs were applied to the project. To generate a high usability the authors applied a user-centered design process involving commonly known methods and principles.

2.1.1 Usability goals

Rogers, Sharp and Preece writes in the book *Interaction Design* that the usability goals are a way to optimize interaction [24, p 19-25]. They are usually stated as questions, thus inviting for the designer to think along the right lines of thought when designing. By using this method, conflicts and issues of the application can be found in an earlier state of the process.

The reason as of why the authors decide to utilize the usability goals was to allow the design of the application to reach a higher level of user experience. Thus, as the application was designed the authors answered these questions and evaluated how to proceed.

- **Learn-ability:** How much time does it take for someone to learn how to use the product's basic/advanced functions?
- **Effectiveness:** How well does the product achieve the purpose of which it was built for?
- **Efficiency:** How well is the product able to aid its users in executing their tasks?
- **Safety:** How safe is the product?
- **Utility:** How well suited is the product's array of functions in comparison to the product's intended purpose?
- **Memorability:** Once learned, how easy is it to remember how to use the product's basic/advanced functions?

2.1.2 Design aspects

The American cognitive scientist Donald Norman highlights the concept of design aspects or rather design principles [25]. If usability goals are used to find and define interaction, design principles are used to actually achieve the desired interaction.

With these aspects in mind, the authors were able to think from perspectives for which might not be too transparent initially. But when thought at, their importance shone through.

- **Affordance:** Defines the characteristic of an object for which determines the obviousness of said object's action.
- **Mapping:** Determines how well a user's mental model of the action corresponds to how the action really transpires.
- **Feedback:** Is the information which is presented to the user when an action has been triggered, e.g. to provide the user information about the progress of said action or the present the end result.
- **Consistency:** Determines how well an interface utilizes general rules through out its user experience, i.e. similar tasks should use similar operations.
- **Constraints:** To avoid the user for executing undesired actions an interface can establish constraints.
- **Visibility:** Determines how easy it is for a user to find the correct trigger for the desired action, both in aspect of the trigger's positioning and contrast to its immediate proximity.

2.1.3 Gestalt principles

A good practice is to apply the Gestalt principles on a project with focus on user-centered design. These are simple and easily applicable guidelines for achieving some of the design principles [26]. The Gestalt principles primarily used in this projects were the one of similarity and the one of proximity.

- Similarity, objects which have the same size, color and shape are generally perceived as connected by the user.
- Proximity, objects closer to each other are generally perceived as a group by the user.

2.1.4 Psychology of colors

Each color possesses an own set of psychology aspects for which it activates in the user, and these aspects can also vary for how the colors are combined together [27]. To know this in advance, the authors were able to choose appropriate colors schemes, so that the application's interface and interaction could be enhanced.

- **Green:** Tranquility and balance. There is no need for change, everything is as it should be.
- **Yellow:** Confidence, but it also possesses a chance of fear and anxiety.
- **Orange:** Physical comfort and warmth, but also deprivation and frivolity.
- **Red:** Can be warming and sensual, but also a sign of warning and aggression.
- **Blue:** Calmness, trust and logic. It possesses a natural feature of soothing.
- **White:** Hygiene and clarity, but also coldness and elitism.
- **Black:** Efficiency and sophistication, but also menace and oppression.
- **Grey:** Psychological neutrality, but could also be perceived as a lack of confidence and energy. Using purely gray can be perceived as depressing and invite on a instinctive level that it is time to hibernate.

An important aspect that needs to be addressed throughout a design process like this one is color blindness. According to statistics approximately 8 percent of the male and 0.5 percent of the female population of Northern European origin suffer from the most common color blindness, red-green color blindness [28]. Based on those numbers it is important keep this disability in mind throughout the entire process.

2.1.5 Sound

When trying to incorporate sound into an application one has to be wary, especially when putting it into an environment where the users might have specific routines, i.e. a professional driver. Dr Roy Patterson at the university of Cambridge published the article *Auditory warning sounds in the work environment* in 1990 where he evaluated the work environment of aircraft pilots and how one could streamline the use of sound inside the cockpit [29]. The article generated general guidelines consisting out of four important points when designing these types of incorporation of sound:

- What is the correct level for a warning sound? Meaning the level where the user is alerted however not excessively disturbed.
- What are the appropriate spectral characteristics? Meaning that the sound should be audible and easy to separate from other possible sounds.
- What are the short-term temporal characteristics that make a warning sound arresting without producing a startled response?
- What are the longer-term temporal characteristics that give the warning sound a distinctive and memorable rhythm?

2.1.6 View of multitasking

When driving a car, the driver needs to cope with an already heavy demand of multitasking and acute attention [30]. It is not only the driver's visual attention which is occupied with the driving, but also the driver's auditory attention, i.e. listening for overtaking vehicles, warning indications from vehicle horns, etc [24, pp 66-70]. Thus, the visual representation can only demand a slight use of the driver's focus and attention for the driver to grasp the visual output of the device. It might even be possible to argue that the driver should be able to understand the conveyed information from the device by using only his/hers peripheral awareness, to minimize the risk of pulling away the driver's focus from the more high priority tasks i.e. not needing the action of turning your neck or even glance your eyes' focus towards the device [24, pp 116-117].

2.1.7 Low-fidelity prototyping

By using low-fidelity prototyping the authors should in theory be able to find problems in their design early on, which might cause trouble in the long run. It also enables the option of trying a few different design ideas, since low-fidelity prototyping is cheap, blunt and quick.

One quite advantageous technique that can be used during the low-fidelity prototyping is storyboarding. As a design tool, storyboarding can be used to evaluate a human's reaction and the capability to use the product [24, pp 392-393]. It is a good technique to adapt when one wants to get a good grasp of the entire perspective which the product one wishes to implement might end up in. Another technique which can be used during low-fidelity prototyping is sketching [24, pp 392-394]. Sketching can be used to further explore the usability and interaction design, but also the feasibility, of a given design concept.

The low-fidelity used for this project consisted of a joint storyboarding followed by an individual sketching which then led to a discussion regarding future designs.

2.1.8 High-fidelity prototyping

The main difference between a low-fidelity prototype and a high-fidelity prototype lies in their respective complexity [24, pp 395-400]. A high-fidelity prototype is more complex and simulates the end product with more accuracy then a low-fidelity prototype. During this stage of the design process, the low-fidelity prototype will have its complexity and functionality enhanced, e.g. several visual representations of the slipping hazard is implemented in Android code. Thus, making it possible for the authors to simulate their proposed design concepts with more precision, before arriving to a conclusion for which suits and solves the problem best.

2.1.9 Framework for iterative usability testing

The danish human–computer interaction expert Jakob Nielsen recommends a sort of iterative test procedure, where one tests smaller functions at a time with a number of test persons set to five [36]. Research has shown that a test group of five persons would in average give insight to 85% of usability problems related to the tested system. So if one has access to 15 test persons he recommends that one should use them in groups of five that tests the system in different steps of the process to achieve more cost effective testing.

2.1.10 Google Material Design

Currently Google provides a wide range of guidelines and restrictions for developers intending to develop Android applications in their Google Material Design document [31]. This was introduced in June 2014 and as of 2015 most Android device uses the guidelines strictly. The guide brings up a lot of details to have in mind during the design process, e.g. which colors work well together and that icons and image should be designed as layers on top of each other instead of designing them three dimensional.

2.2 Benchmark analysis

The purpose of the technological considerations was to establish which hardware would suit the system's needs best. The two core components was an RS-232-to-Bluetooth adapter and an Android device and their respective most desired features were prioritized by Klimator. The authors started of browsing through different webpages selling electronic appliances such as global AliExpress and Swedish company NetOnNet and when suitable components were found they were judged against the categories related to the device.

2.2.1 RS-232-to-Bluetooth adapter

To pick the most suitable RS232-to-Bluetooth adapter for this project the authors conducted a small literature study to full-fill the customers needs. The customer and the authors discussed early on about which characteristics that were important for the RS232-to-Bluetooth adapter. These discussions led to the definition of four categories and these were used to judge the suitability of a specific adapter. Based on the categories the authors searched the market after adapters meeting the four categories Bluetooth-class, price, baud-rate and size of adapter.

They are given below in the order from most important to least important (based on the customer Klimator's wishes):

1. **Bluetooth-class**

   The Bluetooth-class affects the potential distance in which two Bluetooth connected units could communicate. Meaning a lower Bluetooth-class gives a more flexible system in terms of the location between two Bluetooth-devices which one wishes to pair.

2. **Price**

   It was important that the cost of the entire system was as low as possible, hence the price of every sub-part had to be taken into account.

3. **Baud rate, i.e. the symbol rate**

   The potential symbol rate which the adapter could transmit data was of importance since the adapter would have to transmit data constantly, i.e. the baud rate. The baud rate defines the potential number of bits which could be transmitted per second [32]. To transform baud to bits per seconds the following equation was used:

   $$bps = baud * number\ of\ bits\ per\ baud.$$

4. **Size of adapter**

   The main customer Klimator expressed that they where investigating if it was possible to place the adapter in a pre-designed holding containing other hardware-related details, hence it was of importance that the adapter would fit in this space. The size is presented in the following format:

   $$length * width * height$$

   and the unit is millimeters.

The result of the conducted study can be found in section 3.1.1

2.2.2 Android device

Concerning the choice of a specific android device, two core aspects existed which would be used to optimize and evaluate the design. Those where cost and screen size. Battery life was not relevant to take into account since the device could potentially be connected to a vehicle's cigarette lighter socket during the application's run time.

1. **Cost**

   Since the product was supposed to be sold including an Android device, it was highly important to choose a cheap device with the proper functionality, e.g. support of Bluetooth.

2. **Screen size**

   The process of choosing an adequate screen size was a difficult task. Firstly, it was tightly coupled with the price of the device. Secondly, the needed screen size was directly connected to the design of the interface and how the visual representation of the slipping hazard would be finalized, however since the process was iterative and that the design was continuously developed it was tough to decide this early on.

The result of the conducted study can be found in section 3.1.2

2.3 Extreme programming

The very core of the agile process XP (Extreme Programming) is to develop smaller functions or parts of functions through short iteration cycles, i.e. sprints [33]. These functions are based on user stories, which are formulated or revised in the beginning of a new sprint and during each sprint these user stories are locked, i.e. cannot be changed. The change of a user story can only come in effect to the next sprint. All of the user stories where derived from conversations with Johan Edblad at Klimator, the authors listened to his ideas and then translated those ideas into user stories. The full list of user stories can be found in Appendix A.

A key point when establishing user stories is that they are engineered in a way so that they are supposed to be feasible for that sprint's time span [33]. The working methodology for this thesis was divided in to four phases describing each individual sprint and their respective number of allocated days per sprint. The four phases were analysis (day 1), design(day 2-8), test planning(day 9) and testing (day 10).

- **Analysis**

  This phase consisted of analyzing the previous sprint and revising the end result of it. The data from that analysis was then applied on the user stories. In the end of this phase the authors picked which user stories that were to be implemented in the new sprint.

- **Design**

  During the design phase the authors initially discussed how to implement the user stories for that sprint. These discussions revolved all levels of design, i.e. source code and user interface. The phases then continued with that the authors did the actual implementation.

- **Test planning**

  The test planning phase was used to take the user stories for the sprint and translate them into concrete acceptance tests. An acceptance test is a test which aims to show that an implementation is working as intended in relation to a requirement or a user story [34]. Once an acceptance test passes that test cannot be allowed to fail again during any future iteration sprint.

- **Testing**

  Every sprint ended with that the authors applied the tests created in the test planning phase on the user stories to verify that the software met the minimum demands. When all the tests had passed the sprint was to be regarded as concluded.

The length of each sprint for this project was set to two weeks(10 workdays). Illustrated in Figure 10 is the life-cycle of each individual sprint with the four aforementioned phases.

The XP process also involves the method of regularly changing between tasks/-functions/modules [33]. This meant that the authors switched tasks once every day or possibly once every other day. This method ensured that the two authors got the opportunity to code, examine and give insight on every part of the software.

Rather then trying to predict which structures are to be used in the code, a team which implements XP instead starts to code with as simple designs and structures as possible [33]. And when a need for a more complex solution arises, the already developed structure is upgraded in complexity. In this way the code will never become more complex than it is requires. To avoid the decay of the code, the XP process demands that the code is frequently re-factored.

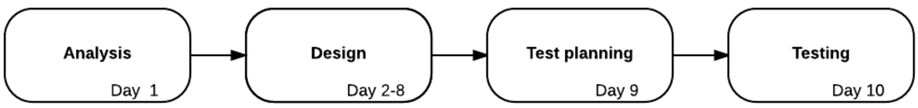

Figure 10: Displays the structure of each sprint

2.4 Peer review

The authors switched reports with another group of students, who also were working on their Master's thesis. The reports were exchanged every two weeks from week six until the end of the thesis. By utilizing this method new insights and suggestions of improvement from other perspectives were made possible.

2.5 Diary

To gain a better grip of the project's progress the authors kept a dairy on all daily events occurring in regards of the project. This provided the authors with both a means to keep track on everything that had transpired but also to be able to motivate themselves.

Mark K. Smith writes in his article, *Writing and keeping. A guide for educators and social practitioners*, that the benefits of keeping a journal/diary are plenty [35]. One of the mentioned benefits was that it provided the keeper of the journal being able to recapture a moment. But it should also grant the keeper with a more natural approach towards criticizing and thinking about past decisions.

2.6 Field-study

Since the project focused on a user-centered design process the authors decided together with the main customer Klimator to convey a field-study with Svevia, a potential future customer, approximately half way through the project. This was done in workweek 10 (24/3-2016). The aim behind this was to validate the design aspects which the authors had thought about and to test them in their end environment. The meeting was divided into two phases, given below:

1. In the first phase denoted the discussion phase the authors along with Johan Edblad from Klimator and Andreas Bäckström from Svevia discussed the product as of week 10, illustrated in figure 11. The discussion revolved around the product's design and it gave a possibility for Svevia to contribute with their insights. It also gave Svevia a possibility to describe their current routines so that the authors could get a better overview of the problems which the end product could solve.

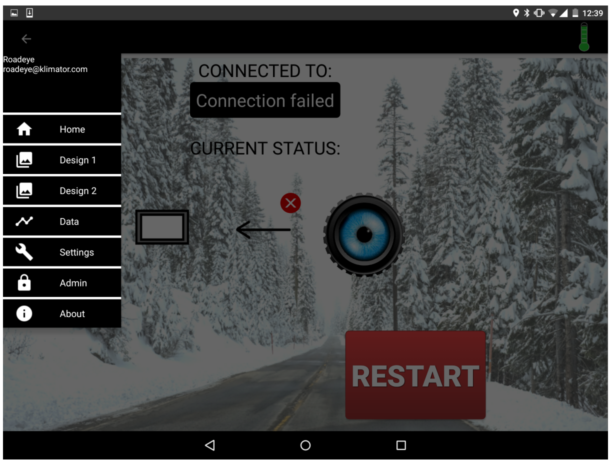

Figure 11: Design as of workweek 10

2. The second phase, denoted the observing phase, consisted of reviewing three of Svevia's operating vehicles. The authors photographed the dashboards and specifically from the driver's perspective with a Nexus 9-tablet's camera. To receive a more accurate view of each vehicles' sound

environment, the authors also measured the decibel levels with a Voltcraft sound level meter 33-2050 from inside the cabin when the vehicle's engine was started. Each vehicle was given approximately 10 minutes for a satisfying assessment.

2.7 Usability testing

As the process was user-centered it was decided that user tests would take place in the end of the project to establish the actual usability of the system. The test and the important aspects of it was discussed between the authors with approximately three weeks left of the project.

Based on the theory in section 2.1.9 the authors decided to convey user testing using five test persons where two worked as truck drivers. The idea was to apply similar testing after the end of this project by Klimator.

The setup of the each test was that the test person sat in the drivers chair, the test leader in the backseat with the sensor with him emulating road surface by directing the sensor at materials mimicking different surfaces. The tablet was positioned on the dashboard and the entire test session was documented with audio and video. Each test session was built as follows (the entire test and its structure can be found in appendix C):

- **Part 1: Introduction**

  During this phase the authors introduced the project and its purpose to the test person.

- **Part 2: Set-up**

  This was the first part of the actual test where the test person was given a few tasks to be executed with the vehicle not moving. They consisted of setting up the system in a way which drivers would.

- **Part 3: Driving**

  During the driving phase the user drove approximately 10 minutes(5 minutes for Design 1 and 5 minutes for Design 2) while conversing with the testleader. However everytime the user noticed a shift on the display he/she notified the testleader. The selected order of Design 1 and Design 2 was randomized to reduced the impact of the rotation. The respective order of Design 1 and Design 2 are changed for every test to reduced the impact of the rotation, this method is know as Counterbalance [37].

- **Part 4: Debriefing**

  During the debriefing phase the authors asked a few open questions to gain insight into the users thoughts and perception of the product. After that the user filled in a standardized SUS-questionnaire which can be found in Appendix D.

3 Results

All the results which were derived from the explained methods are presented here.

3.1 Stakeholder analysis

The results of the stakeholder analysis can be found in section 1.3.4, table 1 *Future potential stakeholders and their individual needs.*

3.2 Benchmark analysis

The result of the hardware considerations are based upon the result from the technological considerations.

3.2.1 RS-232-to-Bluetooth adapter

As discussed in the Methods-section the authors judged the RS-232-to-Bluetooth adapter on the following categories(in prioritized order): Baud-rate, class, size of adapter and price. The result can be seen in table 2.

Table 2: Potential RS-232-to-Bluetooth adapters to be used described by name, baud rate, bluetooth class, size and price.

Product/Manufacturer	**Baud rate(bps)**	**Class**	**Size (mm)**	**Price**
BL-819/Brainboxes [38]	1200-921600	2	75x34x19	$ 86.27
LM048/LM Technologies[39]	1200-921600	1	43.6x34x16	$ 81.33
BT578/IRXON[40]	4800-115200	2	78x34x16	$ 32.22

3.2.2 Android device

As discussed in the Methods-section the authors judged the Android devices on the following categories(in prioritized order): price and screen size. The result can be seen in table 3.

Table 3: Potential Android devices to be used described by name, price and screensize.

Device/Manufacturer	**Price ($)**	**Screensize (Inches)**
HTC 9/Nexus [41]	479	9
Quad-Core Q88/AllWinner [42]	35.29-40.59	7
Y360/Huawei [43]	67.76	4
Surftab Breeze 7.0/Trekstor [44]	73.94	7
Archos 70c Cobalt/Archos [45]	73.94	7
LG L40/LG [46]	73.94	3.5

3.3 Low-fidelity prototyping

The low-fidelity prototyping was executed in week one. The two authors first worked with a storyboard illustrated in figure 12 to establish a common understanding for the end environment for the product. Then the authors drew their own individual low-fidelity prototype, these were then compared and discussed to establish the direction which the authors aimed to pull the project in. The session generated two main concepts denoted Design 1 and Design 2 which the authors aimed to implement and test against each other.

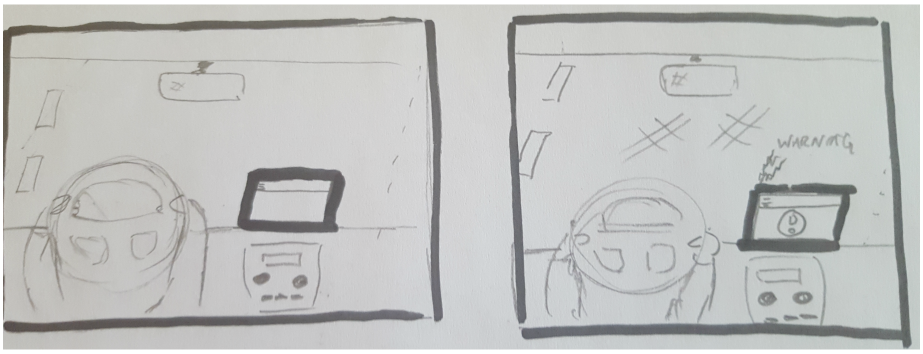

Figure 12: Low-fidelity storyboard

Design 1 Low-fidelity prototypes of design 1, where the focus was a color scheme progressing from green to red as the slipping hazard grew more severe illustrated in figure 13.

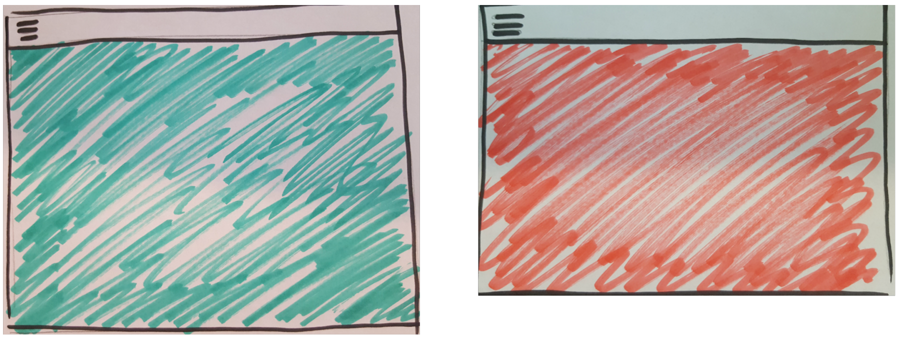

Figure 13: Low-fidelity prototype of Design 1

Design 2 Low-fidelity prototypes of design 2, where the slipping hazard was defined from the level of contrast between an image and the background, but also the size of the chosen image illustrated in figure 14.

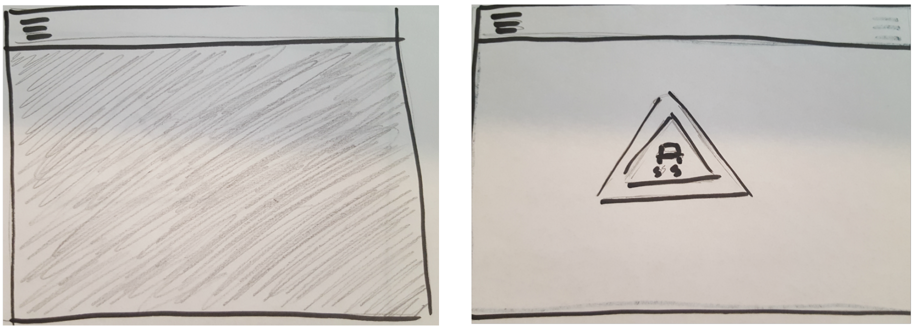

Figure 14: Low-fidelity prototype of Design 2

3.4 High-fidelity prototyping

The application's interface design will be presented in this chapter.

3.4.1 Home

The home screen, which can be seen in figure 15, is the very first screen which the user meets upon booting the application and after a short loading bar has finished. This fragment's only two tasks are to display which Bluetooth address the application is connected to and if the connection was successfully made. The left part of figure 15 displays a successful connection, while the right part shows an unsuccessful connection.

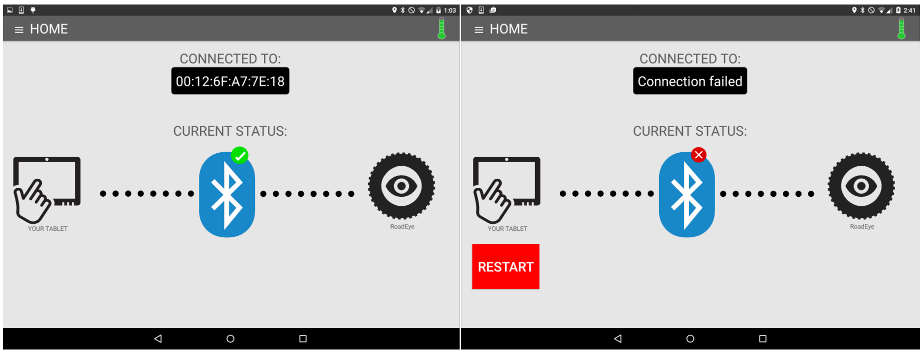

Figure 15: The home fragment, successful connection to the left and failed to the right

3.4.2 Design 1

A simple color scheme with four distinct levels, and which inherit the incremental changes as: Low slipping hazard - an intermediate slipping hazard - a high slipping hazard - a severe slipping hazard, see figure 16. An appropriate mapping of the color choices are, in order of severity (with red being the most severe): Green, yellow, orange and red.

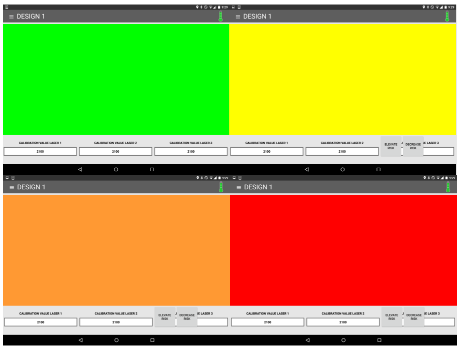

Figure 16: Design 1, the four levels

3.4.3 Design 2

The slipping hazard will be represented through the transition of the size and contrast of a chosen image. Figure 17 clearly depicts that the chosen image is that of a slipping car. As the slipping hazard grows more severe the size of the image becomes larger and the contrast against the background becomes sharper. When the slipping hazard arrives at its most severe level, the black border around the image is changed to red.

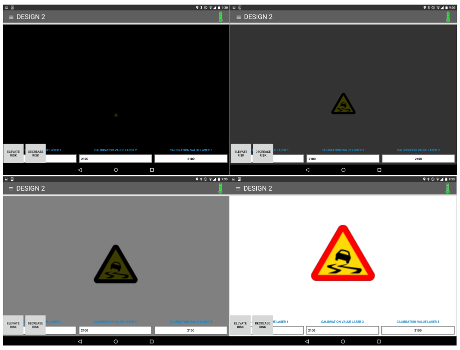

Figure 17: Design 2, all levels of slipping hazard

3.4.4 Data

When the application have been able to make a successful connection to the laser sensor's Bluetooth adapter, the data fragment prints all the relevant data as can be seen in figure 18.

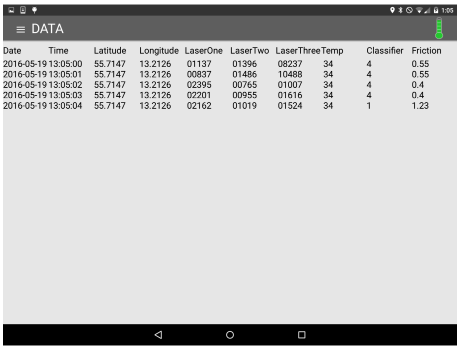

Figure 18: Data fragment, displaying its terminal-like print outs.

3.4.5 Settings

As the user enters the settings fragment, the view presented to them can be seen in figure 19. In this fragment the user can toggle on and off different aspects of the interface.

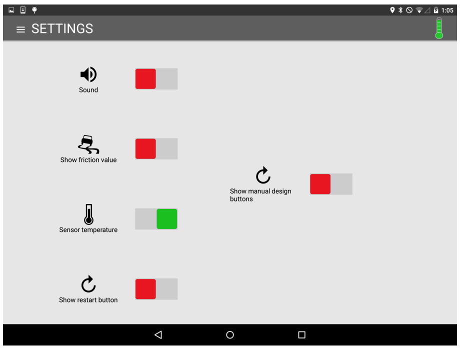

Figure 19: Settings fragment, displaying all the possible settings options.

3.4.6 Admin

The admin fragment has two phases, the log in prompt and when access has been granted. The log in prompt can be seen in the left part of figure 20 and the one when access has been granted to the right. The soft keyboard was activated as the user enters the admin fragment.

3.4.7 About

The about fragment, can be seen in figure 21, presents a short description of the company Klimator and also provides a clickable URL link, directing the user straight to Klimator's website.

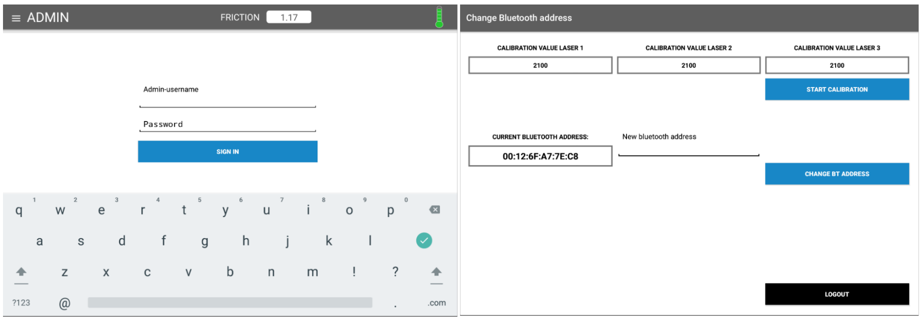

Figure 20: Admin fragment, before and after login procedure

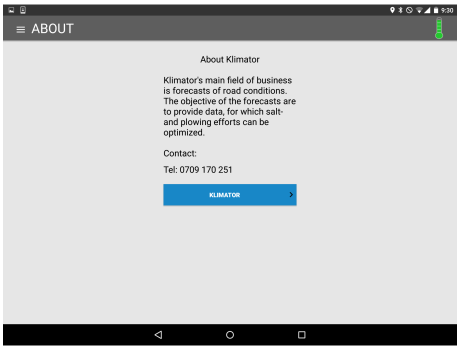

Figure 21: About fragment.

3.4.8 Drawer list

The drawer list is the list which presents all the possible navigational options for the user. When the user activates the drawer list, the change to the view will be as seen in figure 22. The drawer list will be put in focus and the current fragment will be put in the background.

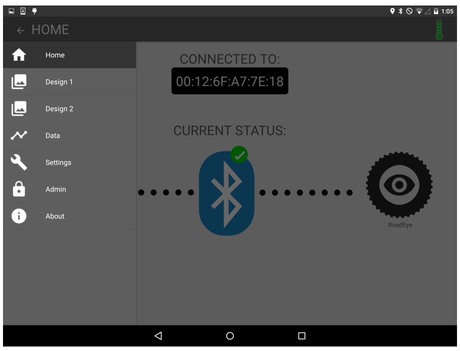

Figure 22: Drawer list

3.4.9 Sound

The current implementation, meaning both Design 1 and Design 2 uses a sound which alerts the driver when the friction is in the lowest interval(highest slipperiness). The sound is the same as most gps-devices uses to warn drivers of upcoming speed cameras.

3.4.10 Logotype

The authors designed a logotype for the application related to the concept of observing the road, due to this the logotype consists of an eye inserted to a tire. Figure 23 illustrated two stages of the logotype.

Figure 23: Logotypes of the application, the left one corresponds to the one used in an early stage and the right one to the logotype used for the end one.

3.5 Extreme programming

This section gives which user stories the authors worked with in each individual sprint.

3.5.1 Sprint 1: 25/1-5/2 2016

User stories worked with:

1. The user can see that the connection between the sensor and the tablet is working.

2. The user can see each single data entry meaning data from the sensor, its corresponding gps position and the time of the measure.

Reflection:
The user stories relevant to this sprint were deemed essential in an initial phase to create a solid foundation to establish connection and to retrieve the corresponding data. Klimator's main focus was to enhance their process of data gathering, hence the authors decided that it was important that this aspect was given the main focus initially. During the sprint the authors gain insight in their own coding and planning capabilities and this gave a possibility to refine the process for the upcoming sprint.

3.5.2 Sprint 2: 8/2-19/2 2016

User stories worked with:

3. The user can see the current estimated friction given between 0 and 1.

4. The user can see the current estimated friction given in a color scheme from green to red.

8. The customer Klimator can take part of old data, meaning lagging data due to network troubles.

Reflection:
As the foundation for the application had been established in sprint 1 it was time to move on to the more advanced user stories. The authors used their gained knowledge from sprint 1 and could maximize the time of sprint 2 in a more productive manner then sprint 1, however some problems arose which gave some unfinished tasks. This was mostly due to that the customer/supplier did not deliver the algorithms needed to finish some user stories. This sprint mainly focused on the interaction design as opposed to the first one where the data transmission was most crucial.

3.5.3 Sprint 3: 22/2-4/3 2016

User stories worked with:

3. The user can see the current estimated friction given between 0 and 1.

5. The user can see the current friction given in a color scheme from black to white along with an image at its biggest together with the white color.

7. The user can see when the sensor temperature exceeds 50 degrees Celsius.

8. The customer Klimator can take part of old data, meaning lagging data due to network troubles.

9. The user shall be able to change which sensor it desires to connect to.

Reflection:
The planning of sprint 3 involved some artifacts from sprint 2. The authors also chose some user tasks that had the potential risk of being too large for one sprint. This along with a sudden shift of approach generated mixed results. The reason behind the shift was an upcoming meeting with potential end customer Svevia. The authors and the representative of Klimator, Johan Edblad decided which aspects that were important to demonstrate for Svevia. These were Design 1 and Design 2 along with a finished home screen. The algorithm had not been delivered for this sprint either so the user tasks related to these had to be postponed once again.

3.5.4 Sprint 4: 7/3-18/3 2016

User stories worked with:

8. The customer Klimator can take part of old data, meaning lagging data due to network troubles.

9. The user shall be able to change which sensor it desires to connect to.

11. The user should be given information when the system is loading.

Reflection: During sprint 4 the main focus was to establish a few features to display at a meeting with Svevia the 24/3. Both design 1 and 2 needed to be working as optimal as possible to generate relevant feedback from Svevia. It was also decided that the home screen needed to be fully implemented. These three menus were fully implemented to present to Svevia. So the focus of the sprint was to give Svevia a glimpse in what this product could do. During this sprint the authors also received the algorithm which gave the authors a possibility to work with the previously mentioned user tasks which had been postponed.

3.5.5 Sprint 5: 21/3-1/4 2016

User stories worked with:

11. The user should be given information when the system is loading.

Reflection:
Since the meeting with Svevia and the Easter holiday occurred during this period, the authors decided to focus on stabilizing the application and not to inject any new functions. It was also decided that a few of the neglected user tasks were to be dealt with. The Svevia meeting gave good information for the authors to improve the software and mainly the user interface.

3.5.6 Sprint 6: 4/4-15/4 2016

User stories worked with:

8. The customer Klimator can take part of old data, meaning lagging data due to network troubles.

12. The system shall take pictures every 10 seconds of the road surface.

Reflection:
The authors continued to work with the user tasks which had been postponed during sprint 5. A lot of the time was also spent on cleaning up the code and making sure that future developers would be able to grasp and work with the code produced in this project. The user stories chosen in this sprint was not fully implemented, the problems were many and the projected workload was far greater then the one the authors had estimated initially. During this time the authors decided that all remaining time, both during this sprint and the upcoming one was to be spent on the report and to clean up the already existing software features.

3.5.7 Sprint 7: 18/4-29/4 2016

User stories worked with:

8. The customer Klimator can take part of old data, meaning lagging data due to network troubles.

Reflection:
All of the time during this sprint was spent on polishing the product, both on source code level and on design level. The aspect of stabilization was quite crucial during this part of the project, the authors deemed it important focus on quality rather then quantity in terms of functionality.

3.6 Field-study

As mentioned before the meeting was divided into two phases: the discussion phase and the observing phase.

3.6.1 The discussion phase

The information gained from the discussion phase was quite beneficial for the authors. Andreas Bäckström who is head of winter roads maintenance at Svevia in Gothenburg gave a lot of insight in the Svevia working procedures and what kind of road information that they were interested in.

Initially Andreas stated that according to Svevia's measurements approximately 10-15 % of the amount of salt which they spread on roads is redundant, meaning that it is not improving the salting efforts. Instead it affects the environment since the salt, which is quite soluble, pollutes subsoil water [47]. Currently Svevia sets up their salting routes before each winter and these routes are then managed by just a few drivers. Since Svevia's maintenance contracts are up for renegotiation every six year it is important that they take any indication of slippery roads serious, hence even though there is just a one percent risk that the roads are slippery, Svevia will still salt those roads.

When handling a salting route the drivers approach can differ according to Andreas. Either the driver's superior tells the driver to spread a fixed amount of salt continuously or the driver can use his own expertise to estimate where to distribute more or less salt. However, the amount of drivers with the right knowledge seems to decrease, meaning that an automated system for decision making regarding the amount of salt to spread would simplify Svevia's work. Inside the cab the driver can control both the amount of salt which he spreads but also how wide the spread should be. According to Andreas Svevia spreads an average of 10 g salt per m^2.

So the interest for Svevia in a product like the one produced during this thesis is to gain more accurate data for which segments of any given road are slippery and for what reason that segment is slippery, i.e. to know if it is because of snow, water or ice. This could then be applied to give indications of how much salt that is needed for each unique segment of a road. Which could be achieved by continuously providing the driver with information about the current friction of the road. The idea of taking pictures every n:th second of the road was also something that according to Andreas could help the people in charge of the salting routes to verify their decisions.

A great amount of the discussion was spent on discussing the subjects of " *Which is the target group for the end product and what kind of information are those individual target groups interested in?* Andreas could only speak from Svevia's point of view and as previously mentioned the friction value would be most interesting for them to handle their operations. Johan Edblad from Klimator brought up that they wish to spread approximately 100 units in the Gothenburg-area to gather a sufficient amount of data, this means that Klimator also needs to find other ways to distribute their system. One example of this could be any type of bus- or cab-services.

The authors and Johan described to Andreas how they would see the end product being put to use:

- For a professional driver to see the data which he himself gathered and data that other drivers collected.
- To increase the amount of data gathered.
- For a regular user to download an application that gives information about the current road status based on other driver's collected data.

3.6.2 The observing phase

After the discussion phase the authors were given a possibility to review three of Svevia's operating vehicles. In every vehicle the authors took pictures of the dashboard and measured the decibel level when the vehicle was started. As illustrated in figure 24, 25 and 26 the vehicles already contain a substantial amount of gadgets which makes the situation more tricky when designing an interface for that specific environment.

Vehicle 1 Decibel level measured: 88-92dB

Figure 24: Displays the cab of vehicle 1

Vehicle 2 Decibel level measured: 64-68dB

Figure 25: Displays the cab of vehicle 2

Vehicle 3 Decibel level measured: 62-65dB

Figure 26: Displays the cab of vehicle 3

3.7 Results from Usability testing

The usability testing was executed with five test persons where two worked as truck drivers and the testing showed that four out of five test persons preferred Design 1 over Design 2. They liked the simplicity of Design 1 and its non-attention seeking nature. The overall navigation tested in Part 1 where the test person did set up the tablet in a non-moving vehicle worked satisfyingly from the authors perspective. All tasks where fully executed by each test person, some were more troublesome then others (such as the calibration-feature) but this was something that the authors had taken into account pre testing.

The testing also gave good insights too how well the product worked with the end environment and which aspects the authors had missed. The test persons expressed a liking for the product but they also shared their thoughts regarding which use they see for the product. The three individuals that did not have the occupation as a truck driver expressed that they liked idea of getting warnings when the road was of slippery sort, they felt that this was more then enough information for them. Deeper information such as the friction value would probably just lead to confusion. The two truck drivers on the other hand liked the friction part since they already have to take that into account when securing their cargo, the friction value of a longer road segment could be quite useful for this endeavor.

The SUS-questionnaires generated good results, the average score was 80,5 on a scale of 1 to 100 where the average for systems using this scale normally end up around 68.

4 Discussion

The discussion will highlight all the aspects of the project and discuss the fallout of them. This sections will also provide the authors' reasoning and thoughts concerning future improvements.

4.1 Hardware evaluation

Under this headline the results from the hardware considerations will be discussed.

4.1.1 RS-232-to-Bluetooth adapter

Based on the investigation the conclusion was that the LM048 should be used throughout this project. It stood out in all the pre-defined categories. Its baudrate matched the one of its closest competitor (BL-819), but it differentiated itself by being both smaller and cheaper. Lastly it was also of Bluetooth class 1, whereas BL-819 was of Bluetooth class 2. The BT578 adapter produced by IRXON had a few good aspects, the trouble with it however was that it was quite big and that it had to be transported from China. For future endeavors when producing the system in a larger scale it could be worth using that adapter since its price is almost half of the other two relevant adapters.

4.1.2 Android device

Based on the authors' study it would be worth buying the Quad-Core Android Tablet Q88 sold at AliExpress for a price between $35.29 and 40.59. It had the sufficient functions, i.e. Bluetooth-support and was found at a low price. However due to the long-delivery time and the short amount of time of this project the authors decided to carry on with the Nexus 9 unit provided by the customer Klimator in the initial phase of the project.

4.2 Low-fidelity prototyping

This step gave the authors a possibility to compare their individual thoughts and ideas in an early stage of the process. The authors came up with two concepts that can be tracked to Design 1 and Design 2 implemented in the high-fidelity stage of the process. The benefit with this method is for a cheap price (in terms of time) one can establish low-level designs and create a common ground for the involved parties to work from.

4.3 High-fidelity prototyping

The following section will discuss the aspects which was previously mentioned in terms of interaction design and how and why the authors used these to develop their high-fidelity prototype.

4.3.1 General

The most crucial aspect design wise for the authors was to deliver a product where the user constantly had a good grasp of its options and possible actions.

To achieve this the authors consistently had the design principles mentioned in the Methods section in mind. Every new component added to the interface where judged on the design principles both independently, e.g. mapping and affordance, and in its environment, e.g. visibility.

Initially the authors discussed whether they should use a main layout built around the concept of tabs or if it was more suitable with a drawer-list layout where a list of possible sub menus could be presented on the left hand side of the interface. The authors chose the latter since it seemed like Google was more directed to this type of interaction tool based on the Google Material Design documentation. Since the application should be as easy as possible for the user to interact with it felt reasonable to use the most common way of interaction.

The drawer-list is supplemented with a toolbar where it is possible to display data such as the current friction value and the status of the temperature in the sensor. As previously mentioned a suggestion from Svevia was to constantly be able to observe the current friction, hence the placement of the friction value in the toolbar. The toolbar is visible in every menu except the admin mode after a successful log in, which adds to the design for a good overview of the entire navigation.

The main colors chosen for the overall design of the menu system and navigation were black, gray and white. Black and white both symbolizes efficiency, and gray provides a psychological neutrality. The color scheme is illustrated in figure 27. The gray color was a good foundation to both black and white, since it lies as a

Figure 27: Grey color scheme

transitional step between the two others and provides a sufficient contrast. Gray should however not be used alone, because it can invite depressing feelings and a lack of confidence. The black and white colors provide the greatest contrast that can be found, hence it was deemed suitable to use that certain combination together with the main color gray. Initially however, the authors worked with white and blue illustrated in figure 28 but to able to enhance the contrast, thus allowing the interface to be easier to use, the blue was replaced with gray. To provide some color injection into a quite sterile interface the authors decided to bring in a blue color that was applied on most buttons.

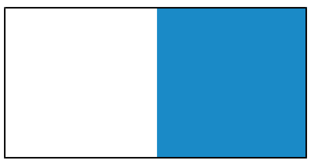

Figure 28: Blue-white color scheme

The interface has been implemented with a certain degree of constraints, to not give users too much access and power over the application and the laser sensor, i.e. the admin mode. The admin mode possesses all high level functionality which the ordinary user should not have access to. To access the admin mode, one must pass through a log in screen.

The application is not orientation sensitive, it has been locked into landscape mode. This is a physical constraint, as to minimize the confusion for which a change in orientation can bring.

4.3.2 Home screen

The home screen's purpose was to clarify to the driver if the connection between the tablet and the sensor was established and working as intended. Since it should be possible to change which sensor one connects to it was decided that it gave a good overview to print the address of the sensor which one currently had a connection to. The simplified image of the set up of the system was to enhance the user's understanding of how the different components interact with each other and to clarify the status of the connection. All of the icons used in this menu are tightly related to the Google Material Design advice regarding icons, i.e. they should be 2 dimensional, bold and geometric.

The button to the bottom left labeled RESTART with red background gives the user a possibility to restart the application and by this the entire connection. It is not desired however for the user to push this button if everything is working intentionally, hence this button is hidden while the connection is working. This was deemed as a necessary constraint to minimize the risk of an unwanted action from the user.

The background of the entire menu was decided to be just plain white. The authors played around with some images related to winter roads but after discussing internally and with other involved parties in the project it was deemed that a single white color was sufficient. This might add a little more dullness to the interface than an actual image would, however based on the probable end environment it was decided that the dullness was acceptable if the menu provided the sufficient information to the end user. The system is intended to be used in a situation where the main prioritization for the user is to drive and to keep an eye on the road, hence the system should be simplistic and easy to overview with the minimum amount of effort.

4.3.3 Data screen

This screen prints all of the data which comes in from the sensor, this feature was added to enable debugging of the data which the application's processes and by this giving a possibility of discovering any improper behavior from the sensor. To an untrained eye this information might be tough to grasp, however for a person with knowledge of the system it is quite simple to determine if the system is working accordingly just by observing this information. The main customer Klimator pushed on a feature like this so due to this the authors did give it a menu of its own instead of adding it behind the admin mode.

4.3.4 Settings screen

The settings fragment reached a point where it mainly served as a home for functions which only could quickly switch around the appearance of the user interface, e.g. turn certain symbols invisible, make the restart button always visible, display the friction coefficient and so forth. This allowed the authors to quickly display different interface set ups and level of abstracts, e.g. how would an application look if only advanced users (admins) would use it? How would it look if only normal consumers would to use the application? The word customization comes to mind. The customization aspects which the user can set the visibility of are sound, display of the current friction value, display of the current status of the temperature in the sensor and the display of the restart button. The design can be seen in figure 19.

4.3.5 Admin screen

All of the important features that could change anything with the data handling such as the connection that is being sent through or the actual calibration of the system has been put under an admin mode. First the user needs to give their credentials which in this case is an email and a password.

As previously mentioned the admin menu provides two main functionalities, one related to the calibration of the system and one related to changing the actual connections address which one wishes to connect to. The two have been divided by using the Gestalt principles of similarity and proximity. The calibration features have been placed closely to each other and the same for the connection address features. However the two features have been designed to look the same, which gives the general idea of two groups (similarity) divided by space (proximity). The hardest part with this has been the choice of a landscape orientation of the tablet. Since the width of the screen is far greater then the actual height it was quite important to get the distances between the two groupings correct.

The calibration group consist of four objects, three objects presenting the latest calibration value for each of the three sensors (if no calibration has been done each of these three is set to 2100 nm as requested by Klimator) and one button to initiate a new calibration. Originally the authors only intended to display the button however after discussing with Klimator it was deemed that it could be quite handy for a future operator to be able to control the calibration values in this menu. The calibration values was later added to Design 1 and Design 2 as well.

The connection address group consist out of three objects, the current address, an input field for a new address and a button to change the current address to the new address. This is a feature which enables easy transitions for the system to change hardware, meaning changing the sensor which one wishes to connect the tablet to.

Down in the bottom right corner a log out button has been placed and the reason behind this is that the authors wanted the user to grasp the other functions before it perceived the option of logging out. The whole idea of this is linked to the western way of reading, from left to right, top to bottom.

4.3.6 About screen

The about screen has not been given a lot of attention although it was worth mentioning that one existed. Currently it displays the basic information about Klimator and how one can contact them directly. This menu does also provide a button which directs the user to Klimator's homepage.

4.3.7 Design 1

Design 1 illustrated in figure 16 corresponds well with an already existing warning system, i.e. traffic lights. This design should provide a great level of learnability, since it is already understood by the general public. The purpose of these views' is to present the slipping hazard to a driver and granted by the simple method for which this purpose is conducted, the design's effectiveness should be good. By the nature of this design and its purpose it is difficult to analyze its efficiency, at least according to the book, *Interaction Design*, definition of efficiency.

The aspect of safety, in regard of this design, should amount to how much of the driver's attention is required to be angled towards the screen. A low amount results in a good level of safety, thus a high amount results in a bad level of safety. As of the design's simplicity, the only concept which has to be understood by the driver is the color of the screen at the current moment. Each color is also designed to cover a great majority of the screen's available surface. With these two elements in mind, this should provide the design with a good level of safety. As mentioned, regarding efficiency, it also difficult to judge the design's utility since it doesn't really possess any functions, at least not any functions which the user can utilize. All functionality is executed in the background. The design's memorability should be high, because the only feature necessary to remember is the color scheme and which in nature is self-explanatory.

Figure 29: A conceptual color wheel displaying the relationships of transition between colors [49].

The array of colors used in the design can be beheld from the figure 16. The green color symbolizes tranquility, and which makes it easy to grasp that there is nothing to worry about, since the slipping hazard is low or non-existent. The red color, which presents the most severe slipping hazard, symbolizes a warning of some kind. By doing a transition between green and red in the wheel of color see figure 29, clockwise rotation, it is clearly seen that before one arrives at red, yellow appears first and latter orange. This allows the user to understand that yellow displays a more severe slipping hazard than green and that orange displays a more severe slipping hazard than yellow.

It is also worth mentioning that the four colors are crisp and vibrant in nature, this allows for a sharper contrast between them. Hence, the user should not be able to accidentally mix up the colors.

As the color scheme is linked to an already existing system, i.e traffic lights, the level of affordance should be great. Resulting from the individual color's coupled psychology and also the relationship between them, the mapping for the user should be elementary. Since, the design doesn't really allow the user to perform any action, other than interpreting the colors, the quality of feedback cannot be evaluated. The design is consistent in all manners, which is inherited from its simplicity. The constraints in the design is high, which results in that the user cannot really do anything, aside from using back press and switch fragment from the drawer list. Also, the level of visibility is difficult to judge, since the user possesses no functions to use.

A problem which arises is: color blindness. The design could work very well for people which have normal eye sight, but it can grow cumbersome to try to adapt it towards a higher degree of accessibility. A problem which is negated almost entirely in design 2.

4.3.8 Design 2

The contrast between the background of the lowest level and the most severe level illustrated in figure 17 of the slipping hazard is very sharp. Since, the background initially begins as black and then as the slipping hazard grows more severe the background progressively changes to white. This allows for a greater change between the levels, thus it should be easier for the user to distinguish between them. This is further enhanced when the image's border color is changed at the most severe slipping hazard.

This design's learn-ability should be at a good level, as the aforementioned aspects provide a natural connection between the different levels of slipping hazard the screens are coupled to. It might not be as easy to recognize exactly which level the current view displays, but the relative difference from view to view makes it apparent that a change in the slipping hazard has indeed occurred and also if it is either for the better or for the worse. The design's purpose is to output the current slipping hazard for the user and with the design's focus being on changing the contrast, this should allow a good level of effectiveness.

As with design 1, there is not any real function of interaction from the user's side. Thus, the design's level of efficiency and utility are difficult to judge. The level of safety comes down to how easy it is for the user to grasp the content on the tablet screen when the user is driving. Since, the design has been constructed with the purpose that the user should be able to understand it without giving it it's main attention, the level of safety should be good. Because of the clear differences and changes that happens between the different views of slipping hazard, the level of memorability should be good.

The affordance between each level of slipping hazard should be very good, since the alterations which occurs to heighten the contrasts are distinct and should be obvious even in the user's peripheral field of vision. The chosen image is a slipping car, which maps naturally to the views' purpose of displaying the slipping hazard. And as the slipping hazard grows more severe the image's contrast to the background becomes sharper. Additionally, the border around the image is changed from black to red at the most severe level of slipping hazard, making the mapping even more obvious. Since, the user doesn't really possesses an active role, as the user only have to understand and view the displayed output, there is no way to evaluate the level of feedback.

The method of displaying the slipping hazard is achieved by heightening the contrast between the image and the background, this is consistently used throughout this design. Because there exists no real reason to provide the user with any functions and to simplify the algorithm's output to a mere slipping hazard, the level of constraint is high. This is coupled to make the experience for the user more pleasant and to avoid unnecessary confusion. The design's purpose is to display the slipping hazard with previously mentioned limitations and aim, e.g. the necessity of the user to switch its attention towards the device should not exist. This results that the design's visibility have been a focus during its entire development process.

Design 2 takes the aspect of color blindness into account a bit more then Design 1. Design 1 works between red and green, which as stated earlier is inseparable to approximately eight percent of the male population of Northern European ancestry.

4.3.9 Sound

After reviewing the work environment of a professional driver the authors drew the conclusion that the sound which was needed had to be clear but somehow not too disturbing. Early on the idea was to use sounds which somehow could be mapped to events as similar as possible to the slipping hazard. The authors had an idea early to use the same sound as a lot of the gps-devices use to warn the driver of upcoming speed cameras. This sound is a tone which is not too attention demanding but it serves its purpose quite satisfyingly. The natural instinct for a driver used to handling a gps is to slow down when hearing this sound, hence the authors felt that this could be something that could be taken advantage of. The authors deemed it appropriate to not use the sound on every changed verdict of the road surface, it was hence decided that it was easier to use it only when the road's slipperiness reached the highest level.

Related to the guidelines which Patterson stated in his paper *Auditory warning sounds in the work environment* the sound alerts the driver however it will not disturb the actual driving, it is audible but one could argue that it could interfere with the sound which the gps-device provides for warning of speed cameras. The authors' idea however is that regardless if its the slipperiness or an upcoming speed camera the favorable outcome for both occasions would be for the driver to slow down. The short-term temporal characteristics of the sound is not the type which causes a startled reaction and its long-term temporal characteristics are probably already known from using a gps-device.

4.3.10 Icons

The hamburger icon/button represents a collapsed list of more options, provides a good level of affordance. This icon/button is used widely in the world of digital interfaces, thus, its function should already be understood by the general public. The result being, that the user will not have to learn its function.

4.4 Extreme programming

Since the authors decided to apply a custom made scheme of the XP programming, it was ahead of time tailored to their specific need. Thus, their analyses and evaluation of the scheme will be biased.

By utilizing a strict process to program, the authors achieved a more rule based way of developing their code. The XP programming's timetable was divided into several two weeks long sprints. As the user stories were defined before each sprint, it was fairly easy for the authors to get a grip on the possible function scope as each sprint concluded. This provided the authors with the advantage to know approximately how many functions/user stories they would be able to finish by the end of the project. It is obviously worth mentioning that as time passed the authors' abilities to write code improved, thus this will as well be biased. But this is an inherent nature of all projects where skills are developed alongside with the project's progression.

One of the core traits with XP is that after each sprint, new test cases are established and tested. But also, the old test cases are re-tested and thus to ensure that not any glitches have made their way into the code during the latest sprint. The authors perceived a liking for this continuous testing, since it cemented and kept the code cleaner. In theory this sounded good but throughout the project the testing in the end of each sprint has been something that has been quite neglected. The test cases were somewhat constructed for all sprints in the end of the project.

This was the first time either of the authors have applied XP in a more professional project, thus to able to take full advantage of the scheme there existed a learning curve to overcome. This lead obviously to that the authors might not have grasped the concept fully at the beginning of the project and which lead to a false assessment, i.e. the authors stated that the XP's focus of task switching would be used. But in the end this practice was never employed, at not least to its fullest degree, because it felt unnecessary and clunky when the authors only consisted of a two-man team.

The process of constructing user stories needs to be improved in the future. The complexity of the user stories were quite varying. Some user stories took a couple of hours to handle, while some stretched over multiple sprints. The key would be to identify the bigger tasks and to find ways of dividing them into smaller parts.

Another key factor to mention is the dialog with potential suppliers in a working process such as this one. One problem with this arose when the authors presumed that the supplier (Klimator) could deliver the specific algorithms for this project in time for a specific sprint. This rendered somewhat of a lag in the entire implementation process. The authors managed to work around this, however for future endeavors it would be wiser to bring the customer/supplier into the planning process earlier on.

4.5 Peer review

It became clear at an early stage, to the authors, that the peer review method gave an added perspective on the paper, but also the project as a whole. The extra pair of eyes and minds, found and questioned matters which the authors had missed or unknowingly left aside. This allowed the project and the paper to reach a higher scientific level.

4.6 Diary

Keeping a diary have resonated well with the agile programming process for which the the authors have chosen to utilize, since it also contain the same focus of keeping track and being able to look back in time to watch what have happened, what the authors have been able to accomplish. It made it possible for the authors to see when, where and why some events during the project occurred, e.g. when did the authors stumble upon issues in the code? When and how were those issues resolved?

Every sprint in the coding process serves as an post in a diary. The benefit of logging the progress of each sprint are that one can record past occurrences and decisions on source code level. To be able to both track progress in a regular diary and in a source code diary has been quite beneficial for the authors. The authors also perceived it, just like Smith wrote in his articles, that it invited to a more natural approach to re-think about past decisions.

4.7 Field study

The field study gave a lot of insight in the potential end environment for the product related to this project. After observing three out of Svevia's vehicles the authors realized that these vehicles already contain a quite substantial amount of other gadgets, buttons and switches. The authors reflected over the situation and drew the conclusion that the interface had to be really clear, almost borderline dull to ensure that the drivers would interpret the information displayed by the interface in a satisfying way from the authors point of view.

The sound which was to be used when notifying the driver that the driving surface was of highest slipperiness was also determined too weak in the quite noisy environment that is the cab of the vehicles. The authors therefor focused on making the sound pop out more. Another aspect which the authors focused on was the friction value and Svevia's keen interest in it. As previously mentioned they expressed what their future hopes of the product and it became clear for the authors that they had almost overlooked this part and its importance for this end user.

4.8 Usability testing

The authors' idea regarding the use of five test persons was to set up a framework which could be applied by later developers of the system since the process was aimed to be iterative in as many aspects as possible. Looking back on the project the authors probably should have usability tested even smaller fractions of the system continuously, however this would have led to that a lot of time resources would be switched from the actual implementation to setting up the different tests.

The results of the usability testing gave the authors an indication of what worked and why. Design 1 and 2 were tested against each other and it was quite clear that Design 1 was preferred due to its simplicity. The test persons generally perceived Design 2 as to attention demanding, meaning that a lot of the attention needed for the actual driving had to be directed to the tablet instead. One reason behind this problem could be the sunny weather during the days when the tests were executed, the aspect of reflection in the screen became quite obvious.

Design 1 on the other hand could be observed in a satisfying manner just by using their peripheral vision. The test persons noticed this more rapidly and with less effort. One could of course look at it from another perspective and say that a clear transitions between every level of slipperiness could produce too much input for a driver to observe, the transitions of Design 2 are not as clear however it is easy to differentiate the lowest level of slipperiness from the greatest.

Another important factor to mention was that only two out of five test persons were truck drivers. This combined with that the product was directed primarily to truck drivers could generate misleading results, this was however not the case. The information that was deemed important from the authors perspective such as that the connection worked between the tablet and the sensor were satisfyingly interpreted by the test persons.

The overall navigation seems to be working in an acceptable manner. The test persons managed to complete the tasks of the test without any help after some exploration of the application. The concept of calibration was not the easiest to grasp and to find the menu where it was to be done. However this is something that the authors counted with. The authors vision is that one person at every company using the product should handle this function, this means that they will receive the appropriate training for such specifics.

As the authors see it the choice of test persons could generate some mixed results. This is mainly due to the relationships between the authors and all test persons. All five test persons were acquaintances of the authors which could lead to biased results, however the authors tried to listen and be attentive to the test persons' thoughts and troubles with the product. One could also question the choice of only two truck drivers as test persons when the product should be angled to specifically truck drivers, both this and the choice of close acquaintances as test persons comes down to the time limit. The usability testing should have been given more consideration earlier on and all the important aspects of it should have been considered wiser.

Regarding the set up one could argue that 5 minutes per design could generate mixed results. Especially when working with Design 2 multiple test persons stated that a lot of their focus was directed at the tablet. It might be more convenient to have some form of adaptation process which is not part of the test where the driver gets accustom with the tablet when driving.

The simulation of data is something that would need improvement. During the test the test leader simulated the road surface, this however rendered mixed results. Since the road status was quite good meaning now slipperiness, no rain and approximately 15 degrees Celsius it was redundant to direct the sensor towards the road since it would only generate the same result on the tablet. Instead it would be preferred if the authors created a simulation before the test independent of the results from the sensor just to control the design in a more efficient manner. This pre-defined pattern and the same route would give a more static test environment meaning that it would have been easier to compare the results of each test person.

4.9 Aspects of future development

To improve both the interest and feasibility of the laser sensor, and thus its distribution and coverage, an idea was to split the product into two implementations.

The first part would be the laser sensor in conjunction with the Android application which would be marketed (sold) to companies which manage the public transportation systems and public roads, see table 4. The vehicles in mind from a data gathering perspective are buses, since they share the same roads as ordinary inhabitants (traveling by car). Since the buses are already driving around in predetermined routes, thus covering a lot of highly trafficked roads, with the laser sensor attached to their hull it would be able to evaluate all those roads. The laser sensor would then need an Android application denoted **Application 1** which would handle the data transmission from the laser sensor to the server, much alike to the deliverable in this project. Other groups interested of this product would be road maintenance drivers and fleet owners.

The second part would be angled towards the individual and this product would only comprise of showing the gathered data from other users, meaning a small portion of Application 1. The individual would only have to purchase the application henceforth denoted **Application 2** and download it from the Play Store to be able to access the crowd sourced data, instead of having to buy, install and use the laser sensor also. This idea would make it simpler, cheaper and more inviting for the individual to purchase the application. And since the buses are already measuring a huge array of roads, there is really no need for the individual to also use the laser sensor. A few roads will not be measured, e.g. private roads and essentially all roads which are not covered by the buss routes. But the gain in distribution will be a huge advantage.

Table 4: Future customers and their respective recommended usage in response to the authors' suggestion of splitting up the product into two versions

Users	**Laser sensor + Application 1**	**Application 2**	**Raw data on server**
Road-maintenance drivers	x	-	-
Bus-drivers	x	-	-
Cab-drivers	x	-	-
Fleet owners	x	-	x
Private individual	-	x	-
Scientists	-	-	x

Another suggestion would be to have the laser sensor embedded directly into cars, i.e. have the sensor installed right at factory level. This would allow a major road coverage, much more than just having the buses outfitted with the sensor. The laser sensor would be included in all future purchases of new cars for that model, and possibly an adapted Android application variant could also be included in the car's own interface directly. This would enhance the distribution of the application and thus the awareness of the slipping hazard.

The authors have also observed that, when the application was running it drains heavily on the device's battery. The pinnacle of battery consumption have reached such a great height that even when the device's was plugged into a car's cigarette lighter socket it was still losing battery life. The reason for this heavy drain was mainly because of the high accuracy and high frequency of the GPS management, but also from an array of other processes for which accumulated drain exceeded the charging capabilities for the charger, e.g. the constant Bluetooth connection. To find a way around this issue was outside of this project's scope, but even so it was considered important to mention in the paper. And especially when the topic is aspects of future development, since if the application was to be used this issue have to be resolved.

The security of the application and the sensor have been put outside the scope of the project, but for future development it was deemed necessary to highlight the most exposed areas.

If no Android device is currently paired and coupled with the sensor's Bluetooth adapter, any device which can execute a scanning for Bluetooth devices is able to find the sensor's Bluetooth adapter. The only wall of security which hinders pairing is a four digit pin code, which yields merely 10 000 combinations. Also, the user has an unlimited amount of tries to input the code. With these two aspects in mind, a potential attacker could easily mount a modified brute force attack on the pin code, i.e. beginning by checking the most common pin codes, e.g. 1234 and 0000, then proceed to try all the other possible combinations. After pairing and when the sensor is powered up, the sensor spews out its data to the application. One possible solution might be to use a quick charger to try to keep up with the high battery drainage.

With the transmission of data between the tablet and the server another potential exposure are introduced. The HTTP-protocol and the two main methods used GET and POST are both quite simple to observe, the information sent in both are easy to read [50]. So if anyone would observe the communication between the tablet and the server it would be possible to see all data sent between the two. A potential solution to this that was discussed during the project between the authors and Klimator was to encrypt the data to be sent on the tablet side and to then decrypt it on the server side. However it was deemed that more urgent matters existed so this was set aside for further improvements.

So what does the problem introduced in the latest paragraph actually potential mean? Well if an attacker would be able to obtain both the raw values from the sensor and the calculated friction and classification, there lies a risk of backwards engineering the algorithm. The risk of actually being able to recreate the algorithm might be very slim, but still, the risk is theoretical.

A suggestion from one of the authors' supervisors was to enlarge every object of the interface, e.g. to give it a Doro feel. The reasoning supporting this concept was that the driver/user should have help to more easily target and press down buttons, access functions and navigation in the interface, since the application was intended for use when driving.

In the start of the project when the authors and their supervisor defined the scope of the project, there was a user story which did not make it into the scope. This was the user story of implementing a layer for Google Maps and to display this to the user. Also, the application would be able to take this information into account when the user inputted its intended destination. With this function the application would have better simulated an end product for the consumer, but as the product simply displays the current slipping hazard to the user, it might not be as useful.

5 Conclusions

The project have favored greatly from that it was conducted by not only one student, but two. Even though the authors have studied the same educational program at LTH, with some slight differences in focus and course selection, it have incited brainstorming and allowed for two different perspectives. Experimenting with contrasting ideas have provided with wider options for designing both the code and the interface. When the authors have stumbled upon a more harsh problem, it have been crucial to be able to quickly brainstorm between each other. Solutions have been found quicker and there have always been additional solutions to try.

Because of the unusual amount of supervisors attached to this project, the authors have always been able to ask questions and obtain answers from outside parties such as customers and supervisors at LTH. The authors feel that this aspect has benefited the project greatly.

In short, it is possible to achieve the project's mission statement defined in the chapter *Aim of the thesis*. Concerning the technical recommendations they might change if the application's interface layout changes, e.g. how much thus the user really benefit from a larger screen? Also, as earlier mentioned the application was draining the device's battery pretty roughly even though it was charging by the vehicle's cigarette lighter socket. The device's battery life should be more prioritized when electing one specific device.

An Android application designed with a user-centered experience in focus is adequate to solve the presented problem space. The logger box's purpose and functions were easily merged into Android code. This was because, a logger box's functions are quite rudimentary. The designing and implementing of a user-centered interface was achieved after accumulating guidelines and principles from the design world, and also actually following them. The authors envisioned quite a few functions which never made it into the application, the main reason being time constraints. But these are presented in the array of user tasks, for which might provide Klimator with additional incentive when moving forward with the development.

Even though the authors' final product is not ready to be released it provides a functional framework, a summary of adequate design guidelines and potentially inspiration for future developments. Concerns which should be addressed in further development are security, unimplemented functions, overall style and the possibility to split up the application into two parts.

References

[1] Johan Casselgren. *Intelligent road systems.*
http://tapahtumat.tieyhdistys.fi/site/assets/files/1284/casselgren_eng_nettiin.pdf (accessed 20 May 2016)

[2] Klimator. *Välkommen till Klimator.*
http://klimator.se/(accessed 11 April 2016)

[3] Miljöbron. *Om Miljöbron.*
http://skane.miljobron.se/om-miljobron/(accessed 11 April 2016)

[4] Svevia.
www.svevia.se(accessed 11 April 2016)

[5] Miller, Brent A. and Bisdikian, Chatschik. *Bluetooth revealed: the insider's guide to an open specification for global wireless communication*, 2 ed. Upper Saddle River, New Jersey: Prentice Hall PTR; 2001.

[6] Poole, Ian (2016). *Bluetooth radio interface, modulation, & channels*
http://www.radio-electronics.com/info/wireless/bluetooth/radio-interface-modulation.php(accessed 11 April 2016)

[7] Electronic Industries Association Engineering Department. *Interface between data terminal equipment and data communication equipment employing serial binary data interchange*, Washington: Electronic Industries Association Engineering Department. 1981.

[8] Android
https://www.android.com/(accessed 11 April 2016)

[9] *IDC.*
www.idc.com/prodserv/smartphone-os-market-share.jsp(accessed 11 April 2016)

[10] XML layout https://developer.android.com/guide/topics/resources/providing-resources.html(Accessed 23 May 2016)

[11] JetBrains. *IntelliJ IDEA*
https://www.jetbrains.com/idea/ (accessed 5 May 2016)

[12] Developers *Android Studios*
https://developer.android.com/studio/intro/index.html (accessed 5 May 2016)

[13] GitHub. *GitHub*
https://github.com/open-source (accessed 5 May 2016)

[14] Design Patterns https://sourcemaking.com/design_patterns(Accessed 23 May 2016)

[15] Singleton-pattern.

https://sourcemaking.com/design_patterns/singleton(Accessed 11 April 2016)

[16] Observer-pattern.

http://fileadmin.cs.lth.se/cs/Education/EDAF10/2015/lectures/Lecture5-handouts.pdf(Accessed 11 April 2016)

[17] Single Responsibility Principle.

http://fileadmin.cs.lth.se/cs/Education/EDAF10/2015/lectures/Lecture2-handouts.pdf(Accessed 11 April 2016)

[18] Regular expression

https://developer.android.com/reference/java/util/regex/Pattern.html(Accessed 23 May 2016)

[19] Google Maps API https://developers.google.com/maps/documentation/android-api/location(Accessed 23 May 2016)

[20] Parcelable https://guides.codepath.com/android/using-parcelable(Accessed 23 May 2016)

[21] Bratvold, David. *What is Crowdsourcing?.*

http://dailycrowdsource.com/training/crowdsourcing/what-is-crowdsourcing(accessed 11 April 2016)

[22] Lufft. *Mobile collection of weather data in real time with Marwis.*

http://www.lufft-marwis.de(accessed 11 April 2016)

[23] Teconer. *Winter maintenance of roads and runways.*

http://www.teconer.fi/en/winter.html(accessed 11 April 2016)

[24] Rogers, Yvonne and Sharp, Helen and Preece, Jennifer, *Interaction Design: Beyond Human-Computer Interaction*, 3 ed. Hoboken, New Jersey: John Wiley & Sons Ltd; 2011.

[25] Moveableonline. *The 6 Principles Of Design, a la Donald Norman.*

http://moveableonline.com/blog/2014/11/03/6-principles-design-la-donald-norman/(accessed 17 May 2016)

[26] Gestalt principles

http://facweb.cs.depaul.edu/sgrais/gestalt_principles.htm(accessed 19 May 2016)

[27] Colour Affects. *Psychological Properties Of Colours.*

http://www.colour-affects.co.uk/psychological-properties-of-colours(accessed 11 April 2016)

[28] Color blindness. *Facts About Color Blindness.*

https://nei.nih.gov/health/color_blindness/facts_about(accessed 17 May 2016)

[29] Patterson, Roy Auditory warning sounds in the work environment. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences* 1990; 327(1241): 485-492. https://www.ida.liu.se/ 729A15/mtrl/Patterson_and_Mayfield.pdf(accessed 20 May 2016)

[30] Aasman, Jans and Michon, John A, Multitasking in driving.(ed). *Soar: A cognitive architecture in perspective*, 1 ed. Springer Netherlands: Springer; 1992. pp 169-198.

[31] Google. *Material Design.*

https://www.google.com/design/spec/material-design/introduction.html(accessed 11 April 2016)

[32] What's The Difference Between Bit Rate And Baud Rate?

http://electronicdesign.com/communications/what-s-difference-between-bit-rate-and-baud-rate(accessed 25 May 2016)

[33] Martin, Robert C., *Agile Software Development, Principles, Patterns and Practices*, 1 ed. Upper Saddle River, New Jersey: Pearson;pp 11-17.

[34] Acceptance tests.

https://www.agilealliance.org/glossary/acceptance/(accessed 11 April 2016)

[35] Smith, Mark K. *Writing and keeping a learning journal. A guide for educators and social practitioners*

http://infed.org/mobi/writing-and-keeping-journals-a-guide-for-educators-and-social-practitioners/(Accessed 20 May 2016)

[36] Nielsen, Jakob. *Why You Only Need to Test with 5 Users*

https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users(Accessed 20 May 2016)

[37] Rubin Jeffrey ,Chisnell Dana. Handbook of Usability Testing: How to Plan, Design and Conduct Effective Tests, 2 ed. Indianapolis, Indiana: Wiley Publishing, Inc.; 2008.

[38] Brainboxes.

http://www.brainboxes.com/product/bl-819/bluetooth-to-rs232-serial-adapter-1-port-male(Accessed 11 April 2016)

[39] LM-Technologies.

http://lm-technologies.com/product/bluetooth-serial-adapter-class-1-lm048/(Accessed 11 April 2016)

[40] IRXON.

http://irxon.com/english/products/bt578_e.htm(Accessed 11 April 2016)

[41] Nexus tablet.

http://www.htc.com/us/go/buy-now-nexus-9/(Accessed 20 May 2016)

[42] AliExpress. *Cheapest 7inch Quad Core Android Tablet Q88 pro Allwinner A33 Android 4.4 Dual Camera WIFI OTG Bluetooth Capacitive Screen.*

http://www.aliexpress.com/(Accessed 11 April 2016)

[43] Netonnet. *4" smartphone med Quad Core-processor och stöd för dubbla sim-kort.*

https://www.netonnet.se/(Accessed 11 April 2016)

[44] Webhallen. *TrekStor SurfTab Breeze 7.0 Quad 7" / 512MB / 8GB / Android 4.4.2 / WiFi - Svart.*

http://www.webhallen.com/(Accessed 11 April 2016)

[45] Cdon. *Archos 70c Cobalt 7" 8GB Android Vit.* [Online].

http://cdon.se/(Accessed 11 April 2016)

[46] MediaMarkt. *LG L40 - Svart.*

http://www.mediamarkt.se/(Accessed 11 April 2016)

[47] What Happens to All the Salt We Dump On the Roads?

http://www.smithsonianmag.com/science-nature/what-happens-to-all-the-salt-we-dump-on-the-roads-180948079/?no-ist(Accessed 25 May 2016)

[48] Maniar, Nipan, et al. *The effect of mobile phone screen size on video based learning*, Journal of software 2008; 3(4), pp: 51-61.

[49] Wheel of colors. *Relations and transitions between colors.*

https://tysonrobichaudphotography.wordpress.com/2010/03/19/a-photographic-study-in-color-henri-matisse-style/(accessed 17 June 2016)

[50] GET and POST security

http://cowbelljs.blogspot.se/2011/12/get-and-post-security.html(Accessed 26 May 2016)

Appendix

A, the full list of user stories

1. The user can see that the connection between the sensor and the tablet is working.
2. The user can see each single data entry meaning data from the sensor, its corresponding gps position and the time of the measure.
3. The user can see the current estimated friction given between 0 and 1.
4. The user can see the current estimated friction given in a color scheme from green to red.
5. The user can see the current friction given in a color scheme from black to white along with an image at its biggest together with the white color.
6. The customer Klimator can see that data is gathered to their server.
7. The user can see when the sensor temperature exceeds 50 degrees Celsius.
8. The customer Klimator can take part of old data, meaning lagging data due to network troubles.
9. The user shall be able to change which sensor it desires to connect to.
10. The user shall be able to calibrate the system.
11. The user should be given information when the system is loading.
12. The system shall take pictures every 10 seconds of the road surface.
13. The user shall be able to see the road status of a longer stretch in form of a map view.

B, the full list of acceptance tests

The following tests are tests established in the end of each sprint to verify that the implemented functionality in that specific sprint are working as intended.
General:

- Start application, did the application start without crashing?
- Does the progress bar finish as the connection is established?

Bluetooth:

- Is the Bluetooth socket set up correctly?
- Is the Bluetooth socket connected correctly?
- Is the Bluetooth socket torn down correctly?
- Is the application able to connect to the laser sensor?
- Is the application able to receive the string value from the laser sensor?
- Can the application continuously receive the string value from the laser sensor?
- Can the Bluetooth handler be restarted?

Time stamp, date and GPS coordinates

- Is the application able to fetch the longitude value of the GPS coordinates, for the current position?
- Is the longitude value updated continuously?
- Is the application able to fetch the latitude value of the GPS coordinates, for the current position?
- Is the latitude value updated continuously?
- Is the application able to fetch the time and date values of the current time and today's date?
- Is the raw laser data synchronized correctly with time stamp, date and GPS coordinates?

Menu navigation

- Does the drawerList display properly?
- Does the drawerList vanish properly?
- Does all the fragments activate properly when the respective *link* has been pressed?
- Does the progress bar activate on boot?
- Does the progress bar's truck move almost alongside the progress bar's progress?

- Does the *sound off* function work?
- Does the *show temp* function work?
- Does the *show friction value* function work?
- Does the *show restart button* function work?
- Does the *show buttons for stepping through the designs* function work?
- Can a someone with a correct username and password log in?
- Does the *Start calibration* function work?
- Does the *Change BT-Address* function work?
- Is the change of a Bluetooth address displayed properly?
- Does the *logout* function work?

Visual/sound representation of the slipping hazard

- Design 1, can it be simulated manually step-by-step?
- Design 1, is the animation adequately fast/dramatic so that the driver is able to perceive the change in slipping hazard?
- Design 1, does the choice of colors possess natural mapping?
- Design 2, are the background colors showed correctly and timely?
- Design 2, is the image's animation executed correctly and timely?
- Design 2, are the contrast sharp enough?
- Design 2, does the animations possess an adequate time interval?
- Is the sound easily understood?
- Does the sound aid the driver's grasp of the change in slipping hazard?
- Does the sound annoy the driver after a prolong duration of use?
- Does the sound distract the driver?

Error managing

- What happens when the laser sensor is not powered up?
- What happens when the device doesn't have Internet access?
- What happens when the device doesn't have Bluetooth powered up?
- What happens when the driver locks the device's screen?
- What happens when the device runs out of memory?
- What happens when a data string received from the laser sensor is corrupt?

- What happens when a someone tries to log in in the admin fragment using a false username and/or password, alternatively enters a wrongly username and/or password?
- What happens if, during a logged in admin, tries to back press instead of logging out?

Application-to-server communication

- Is the application able to set up a connection to the server?
- Is the string and variable management properly executed, i.e. is the data string correct?
- Is the GET request *package* sent?
- Is the GET request *package* delivered?
- Is the server's database updated accordingly?
- Is the connection correctly finalized and taken down without exceptions?

C, Usability testing, instructions for test subjects.

Introduction

We are two students currently working with our Master's thesis at Klimator, a swedish company located in Gothenburg. Klimator's main focus are road conditions and how one can model them and determine the current road conditions. Our main focus for this thesis has been to develop an Android-application that connects to one of Klimators existing products via bluetooth and improves the process of data gathering. Early on in the project we decided to focus on truck drivers since we saw a great potential in this area for the product.

The test is divided into three parts:

1. During part 1 you will interact with the application while the vehicle is not moving. While executing this part I would like you to discuss your thoughts regarding your actions. This will take approximately 5 minutes.
2. During part 2 you will interact with two different designs illustrating the same thing. This will take approximately 5 minutes per design so 10 minutes all in all.
3. The third part will start of with a discussion between the test leader and yourself. Then you will fill in a SUS-questionnaire. After this the test is regarded finished.

We will record video and audio of the test, the recorded material will however only be used by the authors to summarize the results of the testing. It will be deleted afterwards and your credentials will not be compromised. Is this okey with you?

I will be able to answer your questions regarding the test and application however I would like you to be as independent as possible.

Pre-instructions

If login credentials are needed, use the following:
Username: **test@klimator**
Password: **test**

Part 1: Non-moving vehicle

Part 1 will take place with the vehicle started but **not** moving.

Pre-conditions Start the application on the desktop.

Tasks

1. Control that the Bluetooth-connection is working.
2. Control the current friction.
3. Control that data is streaming.
4. Change the connection address to **00:12:6F:A7:7E:18**

5. Calibrate the system
6. Change so that:
    - The friction value is shown in the interface
    - The sound is enabled
    - The sensor temperature is **not** shown in the interface.

Part 2: Moving vehicle

Part 2 will be executed with the vehicle in motion for approximately 2 x 5 minutes. The first five minutes the driver will drive around with Design x running, after that the application changes to Design y for another five minutes. The test is to be regarded ended after 10 minutes.
Start the vehicle and notify the observer every time you observe a change in the verdict.

Part 3: Debriefing

Open questions:

1. Spontaneous impression of the application?
2. Did the application fell coherent or did it feel sprawling?
3. What kind of information would you appreciate in an application like this one to benefit from in your current work environment?

D, SUS-questionnaire

Question	1 (Do not agree)				5 (Fully agree)
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					