CodeBuddy

Development of a programming assistance marketplace as a web application

Filip Renholm
Fredrik Möller
Isak Jansson
Klara Gustafsson
Leo Johansson Lara
Linus Bodeström Eriksson
Ludwig Thaung
Vendela Egondotter

Tutor, Dennis Persson
Examinator, Aseel Berglund
Copyright

The publishers will keep this document online on the Internet – or its possible replacement – for a period of 25 years starting from the date of publication barring exceptional circumstances.

The online availability of the document implies permanent permission for anyone to read, to download, or to print out single copies for his/her own use and to use it unchanged for non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional upon the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: http://www.ep.liu.se/.

© Filip Renholm
© Fredrik Möller
© Isak Jansson
© Klara Gustafsson
© Leo Johansson Lara
© Linus Bodeström Eriksson
© Ludwig Thaung
© Vendela Egondotter
Abstract

According to a study at Linköping University, students who study programming need additional assistance beyond the help provided in the course. According to the same survey, the majority of these students indicate that they would pay for private tuition in programming. At the same time, there are students who believe they have the competence to teach others in programming and could consider doing this against financial compensation. To meet these needs of students, a solution tested in this thesis resulted in the development of a web application to create a C2C platform where students can meet up to exchange knowledge within programming. The development of the web application is varied by design and functionality substantiated by scientific theories. During the project, user testing is performed to get opinions from the potential target group. The work mainly focuses on answering the research questions regarding consumer trust of the web application’s design. The report shows that it is possible to develop a web application according to the above criteria.
# Table of Contents

1 Introduction .................................................................................................................. 6  
1.1 Motivation .............................................................................................................. 6  
1.2 Aim ......................................................................................................................... 7  
1.3 Research Questions .............................................................................................. 7  
1.4 Delimitations ......................................................................................................... 7  

2 Theory ........................................................................................................................... 8  
2.1 Consumer Trust ................................................................................................... 8  
2.1.1 Recognition ........................................................................................................... 8  
2.1.2 Trust and Distrust of Web applications ................................................................. 9  
2.1.3 Consumer-to-consumer Trust ............................................................................. 10  
2.1.4 Rating system ..................................................................................................... 10  
2.1.5 Evaluating Consumer Trust ............................................................................... 11  
2.1.6 Usability ............................................................................................................. 11  
2.2 Aesthetics ............................................................................................................... 12  
2.2.1 Visual Aesthetics of Website Inventory (VisAWI) .............................................. 12  
2.2.2 Balance ............................................................................................................. 13  
2.3 Method Theory ..................................................................................................... 13  
2.3.1 Questionnaire ..................................................................................................... 14  
2.3.2 Prototype ........................................................................................................... 14  
2.3.3 Usability and User Testing .................................................................................. 15  
2.3.4 Continuous Agile Testing .................................................................................... 16  

3 Method ............................................................................................................................ 17  
3.1 Pilot Study ............................................................................................................. 17  
3.1.1 Demand Analysis ............................................................................................... 17  
3.1.2 Feature Analysis ................................................................................................. 18  
3.1.3 Prototype ........................................................................................................... 18  
3.2 Implementation ...................................................................................................... 18  
3.2.1 Modus Operandi ................................................................................................. 18  
3.2.2 System Overview ............................................................................................... 19  
3.2.3 Front End ........................................................................................................... 19  
3.2.4 Back End ............................................................................................................ 20  
3.2.5 Database ............................................................................................................ 20  
3.3 Testing .................................................................................................................... 21  
3.3.1 User Testing ....................................................................................................... 21  
3.3.2 Continuous Agile Testing .................................................................................... 22  

4 Results ........................................................................................................................... 23  
4.1 Pilot Study ............................................................................................................. 23  
4.1.1 Demand Analysis ............................................................................................... 23  
4.1.2 Feature Analysis ................................................................................................. 23  
4.1.3 Prototype ........................................................................................................... 24  
4.2 Implementation ...................................................................................................... 24  
4.2.1 General Design ................................................................................................. 24  
4.2.2 Start Page ......................................................................................................... 25  
4.2.3 Navigation Menu ............................................................................................... 27  
4.2.4 Registration ...................................................................................................... 29
5 Discussion ................................................................. 50
  5.1 Result ...................................................................... 50
    5.1.1 Pilot Study .......................................................... 50
    5.1.2 General Design ...................................................... 50
    5.1.3 Start Page .......................................................... 53
    5.1.4 Navigation Menu ................................................... 53
    5.1.5 Registration ........................................................ 54
    5.1.6 Login ................................................................. 55
    5.1.7 My Profile ........................................................... 55
    5.1.8 Search ............................................................... 55
    5.1.9 Teacher Profile ...................................................... 56
    5.1.10 Rating & Review ................................................... 56
    5.1.11 Booking of Teacher .............................................. 57
    5.1.12 My Bookings ......................................................... 57
    5.1.13 Payment Process .................................................. 58
    5.1.14 Chat Function ...................................................... 59
  5.2 Method .................................................................... 59
    5.2.1 Demand Analysis .................................................. 59
    5.2.2 Feature Analysis .................................................... 59
    5.2.3 User Testing ........................................................ 60
    5.2.4 Source Criticism .................................................... 61
    5.2.5 Validity ............................................................... 61
  5.3 Work in a broader context .......................................... 62

6 Conclusion .................................................................. 64
  6.1 Aim and Research Questions ...................................... 64
  6.2 Generalizability of the Study ....................................... 64
  6.3 Recommendation for Future Studies ............................ 65

7 References .................................................................. 66
Appendix A – Questionnaire .............................................. 69
Appendix B – Summary of Questionnaire ............................ 72
Appendix C – Prototype ................................................... 75
Appendix D – Product Backlog .......................................... 79
Appendix E – User Stories ................................................ 83
Appendix F – User Test .................................................... 87
Appendix G - CodeBuddy Marketing Plan .............................. 89
1 Introduction

The purpose of this project is to create a marketplace using a web application. The application is aimed towards students that need assistance or can give assistance within programming. The following chapter introduce the specific research question, explains why this application should be studied more closely and presents the purpose of this thesis. This chapter also presents the most important delimitations that have been deliberately made.

1.1 Motivation

Among students that have studied at least one programming class, 77% specify that they needed additional assistance in programming beyond the help provided in the course, see Appendix B – Summary of Questionnaire. Often, teachers and assistants are not able to assist to a sufficient extent in the programming courses that is preferred by the students. According to a survey, see Appendix B – Summary of Questionnaire, 84% of the students who indicated that they need additional help in programming are also willing to pay for private tutoring. At the same time, there are students who believe that they have the competence and knowledge to teach programming and who would like to assist other students with programming. Among the students who stated that they could consider teaching programming, 22% specified that they could do it without financial compensation and 78% with financial compensation.

Based on the needs described above regarding the lack of assistance in programming and the willingness to teach other students in the subject the idea of CodeBuddy emerged. The idea is to, by the use of a web application, create a marketplace where students can meet to gain help within, as well as give assistance in programming. Students who are willing to teach others will be able to post a profile on the site where they indicate which programming languages they possess knowledge within and when they are available to teach other students on the subject. This service would be provided in exchange for a given financial compensation. The student who is being taught receives the requested assistance for a reasonable price.

The web application will also make it possible to subscribe to the available timeslots that an assistant student has entered. Once it has been confirmed that a teaching session shall take place between two students, they will be able to contact each other through the web application to determine how and where the teaching session will take place.

In order to build trust between the service and the users, the web application should be designed in a way that generates consumer trust. As the use of the web application as a C2C (customer to customer)-tool, trust must be felt by both buyers and sellers. Confidence in the service can be built by influencing a person’s internal and external trust. [1]
1.2 Aim

The purpose of the project is to create a marketplace for students where they can meet to receive help within, respective, give assistance in different programming languages. In order for students to be willing to use the service, it is important to create a relationship with the user based on trust. This study will research if and how a web application regarding programming assistance can be developed to infuse confidence in the user.

1.3 Research Questions

How can a web application with the purpose of coordinating programming assistance be designed to generate consumer trust?

1.4 Delimitations

The service is designed to target students who studies programming at a college level. The web application will be designed in Swedish and the user base is then delimited to Swedish-speaking students.

During market research, only students who have previously studied programming will be taken into consideration. User tests will be delimited so that a test group with a number of individuals will evaluate the subjective experience of the web application during its development. Since the service is aimed toward students who are interested in programming, only students from that group will be used in the user tests.
2 Theory

This chapter will deal with the scientific theory acting as a foundation for the web application. The chapter include theory about consumer trust, aesthetics together with a method theory.

2.1 Consumer Trust

Trust is defined by Cambridge Dictionary as “to believe that someone is good and honest and will not harm you” or that “something is safe and reliable” [2]. For consumer trust this mean that the consumer perceive that a seller or location of commerce is safe and reliable to perform a purchase with. A study conducted in 2003 concluded a consumer’s perceived trust of a web application will largely affect whether or not named consumer will use the application or not [3]. Bearing this in mind, one can resolve the immense importance of trust, regardless of the purpose of the web application being to educate or to profit from.

2.1.1 Recognition

According to The Pennsylvania State University a consumer will have more trust in a web application if they have knowledge or experience about the web application pre-purchase. For users that have not yet encountered with the web application a strong positive impression is needed to gain credibility and trust for the web application. [4]

A similar result was found from a study done by P. B. Lowry as it also shows that brand awareness increases the initial trusting belief among customers. The study brings up how this can be gained from implementing a design on the web application or logotype that the customer recognizes features from well-known brands and websites. [5]

Similar to show well known features on the web applications, it is also important that the web application’s own logotype is clearly visible throughout the whole application to gain trust from individuals visiting the web application. [6] Further on the same study done by F. N. Egger shows that customers can increase their initial trust of a web application if there are any third party certifications that prove for the customers the e-shop’s validity. [6]

One important third-party certification in Sweden is the certification for trustworthy e-commerce called “Trygg E-handel”. The aim of the certification is to give consumers trust when visiting a web application that sell either goods or services. In the meantime, “Trygg E-handel” aim to educate consumers in their consumer rights during commerce over internet. [7]

There are fourteen key points a company need to fulfil to be given the “Trygg E-handel” certification on their e-shop. Below follow some key points that are relevant to a e-shop with services. [7]

1. The company’s logotype and full name should be easy to find together with the company’s contact information.
2. The company should be able to answer questions and complaints from customers within 48 hours during working days.
3. The customer has the right to get help with reclamation, right of regret and other help considering purchase even though the company of the web application do not directly provide the good or service.
4. The total price should be visible for the customer including taxes. There also need to be a clear product information stating what the customer gets from purchasing.
5. An order confirmation should be send by email or letter after purchase has been done.
6. The customers have the right to get the money back within fourteen days from the day the company go the order.
7. The customer should be informed about the terms and conditions for rights of regret and how it is done.
8. The company cannot enter a commerce with a minor (below 18 years old) without the approval of a guardian.
9. It should be clear in what ways payment can be done.
10. The web application should be designed in a way that is easy to understand and adapted to the targeted market. A tab or similar should include information about payment and terms and conditions.
11. If the purchase includes subscription of the goods or service, it should be clearly stated for how long the purchase is valid. It should also be stated when the customer can withdraw from the agreement.

2.1.2 Trust and Distrust of Web applications

The concept trust is important for sales. However, a study done by C. Xiaohuan Ou and C. Ling Sia shows that trust is not as critical on consumer decision making as distrust is on the users when considering purchase on an e-shop. This suggest that it may be important for companies owning an e-shop to consider preventing distrust issues prior to implementation of features and functionality that increases trust. The study shows that distrust tends to be raised if basic functionality requirements are missing while consumer trust is stimulated by design factors. [8] This is true also according to a study of 2 500 people done by B.J. Fogg at Stanford University as it states that the stand alone most important feature of a website to be viewed with credibility and trust is its design look. This include elements of the visual design, typography, layout, white space, images and colour schemes. [9]

Similar results were gained from another study done by P. Supavich and R. Sarathy which asked user to try four different version of the same unfamiliar website and collected data on how the trust for the website was affected. The four websites could either have high or low on the two aspects visual appeal and ease of use. The reveal of their findings was that high visual appeal in most cases would evoke trust among the users. It also showed that perceived visual appeal gave a significant increase in perceived ease of use for the users. [10]
A study done in 2014, however, shows opposite result. The study done by M. Seckler, with others, suggest that distrust is caused by graphical and design issues while trust on the other hand is enhanced by social factors such as recommendations by friends and reviews. This discovery has similar results as the study from Pennsylvania State University that states that a web application will gain more trust if the consumer has any knowledge about it from before. [11] [4]

Another study done by Princeton Survey Research Associates shows that gaining trust and credibility for a website has a strong correlation to what type of website it is. The study including 1 500 adult online users shows that small businesses [12], newspapers and television news are the type of websites that give the most trust to consumers while websites that sell either products or services have the lowest rate of trust among online users. The same study also shows that the two largest factors for deciding to visit a website is if the website is easy to navigate and if the consumer is able to trust the information on the website [13].

### 2.1.3 Consumer-to-consumer Trust

Consumer-to-consumer (C2C) e-commerce has been found to be more popular than business-to-consumer (B2C) e-commerce when it comes to web forums, online auctions and chat rooms. Differences have been found that say B2C e-commerce methods cannot be used in C2C e-commerce. In order for the consumer to overcome perceptions of uncertainties and risks of using online transactions, it is important to emphasize competence, responsibility and reliability. In C2C e-commerce, it is important that the web application conveys confidence to both buyers and sellers. This can be done by affecting the user's internal and external trust in the service. [1]

The internal trust, according to a study done by K. Jones and L. Leonard, consists of a person's inclination to rely on previously visited websites with C2C e-commerce transactions and the perception of the website’s quality. External factors are said to be opinions of other users regarding their trust in the web application as well as third-party recognition. [1]

The study tested how the two internal and two external ways of trust affect the consumer trust on a web application. The result of the study shows that only the internal factor of perception of website quality and the external factor of third party recognition could visibly improve the consumer trust in e-commerce. [1]

### 2.1.4 Rating system

Rating systems is a tool for e-commerce and e-shops to have a trust and reputation system for both buyers and sellers. The basic idea is to let both parties give a rating on each other, often after transaction of some sort, to describe satisfaction and trust. The rating system can act as a guideline for customers on which seller to trust when doing a commerce and vice versa. [12]
A positive natural side effect of using rating system is that the behaviour of customers and sellers improve together with the general quality of the product or service. This can also lead to an improved trust in the seller or whole e-shop. [12]

2.1.5 Evaluating Consumer Trust

The following four key principles can determine the level of trust perceived by the customer [14]:

1) **Design Quality.** To make the web site appear legitimate and professional are of great importance.
2) **Upfront Disclosure.** Customers prefer when sites are upfront and displays the information which is relatable the experience of the user.
3) **Comprehensive, Correct and Current.** The consumers appreciate when the site’s information is delivered with a sense of expertise, cunning as well as being up to date and easy to understand.
4) **Connected to the rest of the web.** The lack of third-party external sources linked to the web site makes it appear less trustworthy.

Taking stance from these key concepts when creating a survey, the trust generated by a web application can be evaluated. Focusing on the information the user is prompted to disclose and what information that is shown on the web site, an answer whether the web application is perceived as trustworthy or not can thereby be formulated. [15]

2.1.6 Usability

A key aspect to consider when designing a trustworthy web application is its usability. This is one of the conclusions of Marie Christine Roy’s study, which states that “the general usability of a web site has an impact on the establishment of trust” [16]. Another report, bolstering Roy’s findings, states that perceived usability indeed influences the degree of consumer trust shown in a website [17]. Since increased usability of a website enhances its credibility, it is important not to lose focus of usability when developing a web application meant to instil consumer trust.

According to Jakob Nielsen the usability of a web application is a complex and multi-dimensional measure. He suggests that a reasonable definition of usability consists of five elements: [18]

- **Learnability:** How easy a user can learn to use the system
- **Efficiency:** How efficient the system is to use
- **Memorability:** How easy it is to remember how the system works after some period of not having used it
- **Errors:** The system should lack of catastrophic errors and have very few minor errors
- **Satisfaction:** A users’ subjective opinion of using the system (e.g. if they like it)
2.2 Aesthetics

One does not have to be particularly imaginative to realize that a web application with appealing aesthetics might instil veracity with the user, and according to a recent study [19], a connection between perceived trust and the aesthetics of a website is likely to exist as stated in the previous section. This is further emphasized by another lucubration suggesting poor style (e.g. aesthetics) significantly “affected users’ perception of site quality”, which in turn has a major impact on perceived trust. [20]

2.2.1 Visual aesthetics of website inventory (VisAWI)

A study conducted with the aim of developing a measure of perceived visual aesthetics of websites resulted in the Visual Aesthetics of Website Inventory (VisAWI). By implementing the VisAWI in a series of tests the study could derive the four key elements of visual aesthetics of a website [20]:

- Simplicity
- Diversity
- Colourfulness
- Craftsmanship

2.2.1.1 Simplicity

Different definitions of simplicity have been used over the years. One of these is discussed by Ronald Arnheim who mentions that simplicity might be defined solely by the number of elements (e.g. forms and colours) an object consists of [21]. A previously mentioned study also states that additional features connected to simplicity are clarity, homogeneity and balance. [19]

Another report, favouring a mathematical approach, on the other hand defines simplicity as follows: “Simplicity is directness and singleness of form, a combination of elements that results in ease in comprehending the meaning of a pattern. Simplicity in screen design is achieved by optimising the number of elements on a screen and minimizing the alignment points.” [22]

2.2.1.2 Diversity

The diversity aspect of the visual aesthetics is meant to seize the visual complexity, dynamics and creativity of a webpage [20] [23]. As goes for simplicity, diversity has been widely recognized for being a vital parameter in website aesthetics [24].

For a website to gain diversity, the VisAWI study suggests five key factors to consider [20]:

- The layout is pleasantly varied.
- The layout is inventive.
- The design does not appear uninspired.
- The layout appears dynamic.
- The design is not uninteresting.

2.2.1.3 Colourfulness

The colourfulness mirrors the composition of colours used when designing the web page. It is suggested that the colour appeal of a website directs contributes to the user's perception of both trust and satisfaction [25].

A general although still concrete and clear guideline when composing a web application for commercial use has been found in a study conducted in 2004. It states that chromatic colours are preferable for creating a visually pleasing website: “Combinations involving the colour blue, and including two chromatic colours (e.g., light blue on dark blue) appear to be preferable to a combination with less contrast and including a chromatic colour (e.g., cyan on black) for promoting positive affect and behavioural intention.” [26]

2.2.1.4 Craftsmanship

Craftsmanship can be interpreted as the skill of putting all design elements together. The above-mentioned study, introducing the VisAWI-measurement, also implies that “A website needs to be harmoniously designed and the artistic ideas need to be implemented with skill and care” and furthermore states that “Craftsmanship reflects whether the site was designed with skill and care using modern technologies”. [20]

The study proposes four factors to regard when composing a website [20]:

- The layout should appear professionally designed
- The layout should be up-to-date
- The site should be designed with care.
- The design of the site does not lack a concept.

2.2.2 Balance

As previously stated a correlation between simplicity and balance is likely to exist. Thus, another often reoccurring feature in studies and reports regarding visual aesthetics of a website is balance [22], [27]. This is further emphasised by Altaboli and Lin, who states that balance indeed has a significant effect on perceived interface aesthetics [28]. Balance is not, although frequently misinterpreted as, a description of how well a pattern is represented on both sides of the display. Rather is it a design tool that correlates different shapes and colours with different amount of optical weight. A balanced screen can thus be created by evenly distributing objects with respect to their optical weight, meaning the ability of an element such as a graphic, text, headline, or subheading to attract the user's eye. [29]

2.3 Method Theory

In order to analyse the methods used in the project, underlying theory has been studied. This theory deals with questionnaires, prototypes and testing of the web application.
2.3.1 Questionnaire

During the collection of data, a questionnaire can be used. There are different ways to form a survey, for example it can consist of Likert-style questions or unipolar questions. Likert-style questions ask the respondent to choose between several alternatives where the alternative in the middle of the range often is a neutral answer like: not sure, neither or don't know. To the right of this neutral answer there are alternatives that in some way agrees with the statement or gives a positive response to the question. To the left of the middle alternative there are a number of alternatives that do not agree with the statement or gives a negative response to the question. It is the middle alternative that balances the positive and negative results, where the different alternatives often are numbered. When there is a middle alternative, the question usually consists of an odd number of alternatives like five, seven or nine. [30]

For evaluation questionnaires are usually Likert-style questions used. These are used to, for instance, examine customer satisfaction while the questions that are called unipolar more often are used to measure for example quantities like how much something is used. The strengths of the Likert-like questions are that they are used very broadly, and the respondents are then used to and comfortable with such an arrangement for the response alternatives. A weakness with Likert-like questions may be that the alternative in the middle of the range can be interpreted differently by the respondents. Some individuals can choose the middle alternative because they possess too little information about the topic, some do not want to give a negative answer and then chooses a neutral answer, and some select the middle alternative because it is their viewpoint. [30]

The Likert-like questions also has the advantages of handling large amount of data from a large testing group quickly, the interpretations made from the data can be validated through a wide variety of methods. The data can also easily be compared, combined or contrasted with other data-gathering techniques of qualitative characteristics such as interviews or open-ended answers. [31]

Nemoto, T and Beglar, D, argues that a four-point scale is preferable when the respondents are young or have low motivation of performing the questionnaire because it demands less effort and makes the questions easier to understand. Furthermore, they claim that a six-point scale is always desirable when possible because of the higher precision in measurement. [31]

2.3.2 Prototype

The idea of creating a prototype is to create an overview of the application's vision at an early stage. A prototype is a hypothetical version of a system that should be easy to modify during the system development [32].
There are different types of prototypes, one type is paper prototype that is a user interface prototype and the technology for creating such a prototype is called low-fidelity (lo-fi) prototyping. Lo-fi prototyping is an effective tool for easily demonstrating the behaviour of an interface at an early stage in the development. The technology enables multiple ideas to be generated in a shorter matter of time compared to high-fidelity prototypes. [33] Another type of low-fidelity prototype is the creation of a design scenario, for example, in the form of a storyboard. This kind of prototype can be used as an early and easy tool to get feedback from users [32].

It's useful to start by developing a horizontal prototype that effectively demonstrates the product's concept. A horizontal prototype provides a broad view of the features but due to lack of depth in the prototype, it can be seen as less realistic. During product development, the prototype should be divided into several vertical prototypes that are more detailed about the selected functions. [32]

### 2.3.3 Usability and User Testing

For a web application to be usable it is crucial to use a well-defined foundation and a clear process for user testing. The goal of testing usability is to see follow-up of functional development of the platform, whereas user testing puts the user in focus and aims to understand how the user perceives different aspects of the platform. These will be performed simultaneously in order to both review functionality and user-perception.

Usability has been defined with terms such as usable and useful; however, there is no common definition [34]. Dumas and Redish however described several important characteristics of every usability test: [35]

1. The goal is to improve the usability of a product. Tests become more specific with time.
2. The participants represent real users and do real tasks.
3. The test leader must observe and record what participants do and say
4. The test leader will analyse the data in order to diagnose the actual problems and recommend changes to be made.

Before every test a set of tasks are listed. These tasks are connected to the parts and functions of the platform recently developed and implemented, and will therefore be evaluated. There is a clear time-limit of the test in order to make all tests from different phases similar [36].

#### 2.3.3.1 Concurrent Think-Aloud Procedure

Think-aloud is a research method in which participants speak aloud any words in their mind as they complete a task. It is an effective way letting developers discover what users really think about their design. [37] If misconceptions are identified during the test, they can be turned into actionable redesign recommendations. It is also possible to analyse why users guess wrong about some parts of the UI and why they find others easy to use [38]. The method itself is cheap, robust, flexible, convincing and easy to learn.
2.3.3.2 Retrospective Probing

Retrospective probing is a verbal probing technique in which the interviewer administers the probe questions after the participants has completed the entire survey, system test or task. Effectively moderating user tests is critical to gaining insights and understanding about your users needs. Retrospective probing does not break the flow of an interview, but can result in participants forgetting what they were actually thinking about at the time as well as participants “inventing” problems they did not have. [39]

According to a short version of the previously mentioned VisAWI-model (see section 2.2.1) a website's aesthetic appeal can be evaluated by letting participants of the retrospective probing rate the website using a 1-7 Likert scale on four statements:

- Everything goes together on this site.
- The layout is pleasantly varied.
- The color composition is attractive.
- The layout appears professionally designed.

The developers of the model further states that "participants usually experience websites as rather positive starting from an overall evaluation value of 4.5." Furthermore, it is recommended to use at least 20 test users. [40]

2.3.4 Continuous Agile Testing

The following guidelines can be used in constructing a strategy for the continuous agile testing [41] [42]:

1. Define and execute a “just-enough” acceptance test.
2. Maintain a single source repository.
3. Automate the build.
4. Make your build self-testing.
5. Everyone commits to the mainline every day.
6. Keep the build fast.
7. Test in a clone of the production environment.
8. Make it easy for everyone to get the latest executable.
9. Everyone can see what’s happening.

To develop effective agile testing and integration, the elements of continuous integration is of great importance [41]. Implementing components of the practice results in rapid integrations and errors can be found and fixed quickly [42].
3 Method

This chapter will explain the method used in order to answer the research question. The methods used are well documented and are generally accepted as effective.

3.1 Pilot Study

Initial studies have been conducted to investigate whether there is a need for the web application and to map the desired functions for the application.

3.1.1 Demand Analysis

A demand analysis was performed to assure the need for an application based on the research question. The demand analysis was conducted by means of a survey where the target was to get 100 answers, of which 25 from students with data alignment.

In order to reach the desired target group, students who had at least once studied a programming class, the survey took place online. The survey was designed using Google Forms and it was distributed by posting a link to the questionnaire in social media groups for students and by sending the link to familiar students.

The survey, see Appendix A – Questionnaire, was designed with seven questions focusing on examining two aspects. One aspect was to investigate whether students were in need of further assistance in programming beyond the assistance provided in the programming classes and if these students were willing to pay for additional support. The second aspect was to investigate whether students who had the competence to teach programming were willing to teach other students in programming if they received financial compensation for it.

The questions in the survey were mostly not designed according to the Likert-like format, where there should be an odd number of response alternatives with a neutral statement for the alternative in the middle of the range [30]. The survey was designed without such types of questions to avoid the weakness of Likert-style questions, e.g. the neutral response option can create a misleading image in the data collection as individuals can select the middle alternative by different reasons. Beyond this, the survey was designed in a way that would simplify for the respondent to answer by highlighting a small number of questions. It was designed this way to prevent it from being perceived as burdensome and time consuming. With support of the compilation of the answers from the survey, see Appendix A – Questionnaire, the intended aim of the web application could be substantiated or deprecated.
3.1.2 Feature Analysis

When identifying possible features that the web application could contain, brainwriting was used. The different ideas that emerged were categorized into a list after whether the features were considered desirable or unnecessary. Functions that belonged together were then grouped into the categories. Within the different categories, the importance of the functions was graded by a three-tier scale to properly satisfy the purpose of the web application. This elaborated list became the product backlog and was also reworded as user stories.

3.1.3 Prototype

During the pilot study, each group member was tasked with designing a horizontal prototype. The different prototypes were designed differently, some were made on paper, other members used power points and some gained inspiration from other web pages. Lo-fi prototyping was used to generate many ideas in a short period of time, in line with theory [33]. The group discussed the various prototype proposals and a joint design scenario was created. The purpose of the first prototype was to give a broad view of the functions and therefore it was designed as a horizontal prototype. [32]

3.2 Implementation

This section describes the methods used for developing the web application. The process is broken down into the categories Modus Operandi, Front End, Back End, and Database.

3.2.1 Modus Operandi

A scrum-based methodology was used in creating the web application. A product backlog was created in the initial stages in the project, complete with priority ratings. The development team continuously evaluated the relevancy of user stories and these were moved from the product backlog to the sprint backlog before every sprint. A scrum master was responsible for coaching the team and coordinating the development with stakeholders.

The web application was built through three sprint cycles after an initial pre-work process referred to as the sprint 0. The zeroth sprint focused on market analysis and decision-making, sets the theoretical and practical foundation of the web application. The first sprint focused on implementing working basic navigational functionality, and describes the methods used for the ongoing development. The second sprint focused on adjusting and adding more specific functionality. The third sprint describes the methods used for finalizing development and user tests. Throughout the development measures were taken to ensure the project focused on the goals set in the beginning of the project.
3.2.2 System Overview

The application had different parts with different systems. The front-end ran through HTML/CSS and Bootstrap together with JavaScript to make it more interactive. In order for the user to communicate with the server jQuery with AJAX were used to create the connection between the back-end and the front-end. Through this connection, the user could retrieve data from the database and input data to the database. For example submitting a booking request or retrieving information about the teachers on the website.

3.2.3 Front End

The front end was developed using the tools HTML, JavaScript, Ajax, jQuery and CSS with additional features using the bootstrap framework.

HTML provides the structure of the web page, and CSS the layout. Bootstrap is used to prevent repetition between projects, utilize responsive design to allow the webpage to adapt to various screen sizes, add consistency to design and code between developers and ensure cross-browser compatibility.

JavaScript is used to add user friendly functionality because of the dynamic nature of the language that also supports an object-oriented approach. jQuery is a JavaScript framework implemented that makes working with the DOM easier by building lots of high level functionality.

AJAX is a method used for XMLHttpRequest from a web page to the server and send/retrieve data to be used on the web page. jQuery implements this interface abstractly, shielding the developers from the complexity of multi-browser support in making the request.

Through the usage of CSS and Bootstrap an overall visual design was developed for the whole website for every HTML file.

Development of the website were made in accordance to theory about how usage of familiar and well-known brands increases consumer trust[4][5]. Third party recognition were used to increase consumer trust, one example was the usage of Visa and Mastercard logotypes[1]. Inspiration from other websites, considered to be trustworthy by the public, were also taken to increase consumer trust.
3.2.4 Back End

The server was developed to function as a middle point between the front-end and the back-end sections of the application structure. Its main purpose is to process requests and deliver necessary content and data. At its core, Python is the main component as a building block for functionality. To further extend the flexibility and functionality of Python, the Flask framework was added using Werkzeug as a toolkit and Jinja2 for the template engine. Flask supports the web application with extended Python functionality such as form validation and various authentications. During development, the server is hosted locally using the port 5000 which is freely available and used by Flask in development environment. The server is also responsible for transferring data back and forth from the database and rendering it as HTML using Jinja2.

As trust issues comes from basic functionality requirements not being fulfilled [8], development of these requirements was prioritized, such as making sure the user can book a teacher. For example, as long as the function to book a teacher was unfinished the development of that functionality was always in focus.

3.2.5 Database

For managing the database, the Flask extension Flask-SQLAlchemy was used. It is a Flask extension and Flask friendly version of the extension SQLAlchemy for Python. During development, SQLite was used as a database engine because of its simplicity and the possibility to develop and test without the requirement of a database server. This setup enabled us to create, manage and transfer data from the database through Python.

An admin interface for the database was developed by implementing functionality from the Flask extension Flask Admin and HTTP Basic Auth. Flask Admin was implemented to obtain an easy managed admin interface and HTTP Basic Auth for safeguarding the application and protect chosen views. The interface can be used for a more efficient way of inserting and managing data complemented with a graphical view for a better overview.

The database was used to store the users, teachers, bookings, teacher types, and chats. Every teacher is a user, but a user does not need to be a teacher. Every booking is connected to a teacher and the category "teacher type" are tables for each programming language that lists every teacher that can teach in that language. SQLite is used through Python to create the database and all the necessary tables and relations between them. Through user interaction, data can be submitted into the tables by going through jQuery AJAX to Python Flask and finally to SQLite. For example, when a booking request is made the request will first pass through Python Flask where the server will control it and if it fulfills the requisites to be put into the database SQLite will be used to submit the request into the database.
3.3 Testing

To ensure that the platform performed according to desired criteria the methods continuous agile testing and user testing were used. The tests were developed with the ultimate goal of evaluating whether the platform was recognized as aesthetical appealing and therefore as trustworthy. This in an effort to build a base for later being able to answer the research question.

3.3.1 User Testing

The user tests were conducted during each development phase of the project. Since usability has been proven to instil trust with the user these tests were developed based on theory regarding both usability and user tests [34] [35] [36]. Before every test a set of tasks were listed. These tasks were connected to the parts and functions of the platform recently developed and implemented, and were therefore be evaluated. The tests were written and prepared on beforehand.

The platform is compatible on both computers and mobile devices. The user tests will, however, only be carried out on a computer.

An important aspect of the test is evaluating how the users interact with the platform, and see which decision they make and where they make them. This requires a test leader who continuously takes notes during the test and answers potential questions from participants [36]. In order to ensure complete coverage, the method Current Think-Aloud Procedure is used in which participants speak aloud any words in their mind as they complete a task [37].

After a conducted test the participants will conduct Retrospective Probing in which the interviewer administers the probe questions after the participants has completed the entire task. These questions will evaluate how well the users think the test went, what problems they encountered and the platform in its entirety. The same questions will be asked after the tests in every phase of the project. This will provide comparability and follow-up how the development of the web application is reflected in the user-experience, according to theory researched. [36] The probe questions were compiled with regard to the theory previously presented concerning usability. Of the five facets of usability suggested by Nielsen [18], the emphasis of the probe questions was put on design satisfaction as the main focus of the report is to instil trust by means of design. The error facet was also partly regarded since a user’s perceived trust of a web application rapidly decreases if basic functionality is missing or works in an improper manner [8].

The test participants are all independent with different backgrounds and levels of technical knowledge in order to reflect the customer-base of CodeBuddy.
Previously mentioned theory suggests that a user test consisting of the four VisAWI-statements should be conducted using at least 20 test participants per test [40]. However, it was decided to disregard these recommendations on test participants due to practical circumstances making it close to impossible for a project of this size to conduct multiple tests with at least 20 participants.

The tests were conducted at the end of sprint 2 and at the end of sprint 3 allowing for improvement of the application in between the two tests. All participants were asked to complete tasks presented to them in a task log, see Appendix F – User Test. According to previously mentioned theory regarding concurrent think-aloud procedure the participants spoke freely whatever came to their mind when conducting the tasks for the test leader to take notes of [36]. After completing the task log the participants answered eight motivating questions followed by five statements used for evaluating users’ subjective response regarding aesthetics and perceived trust, see Appendix F – User Test.

After a conducted user group testing the test leader analysed the data in order to diagnose the actual problems, and to recommend changes to be made.

3.3.2 Continuous Agile Testing

In order to effectively work across the development team, a set of guidelines was created to test the code continuously. Some practices of continuous integration was implemented [41] [42]. A single source repository was used to allow everyone to easily get the latest build and see the processes develop. New functionality was continuously integrated to the main application. It was crucial to prevent the integration of non-functioning code or code that could potentially damage earlier accepted functionality. To avoid this, every member went through a checklist to ensure that new functionality works along with the core application. To allow for delegating out tasks to members of the group Trello is used to put assignments and assign members to tasks. A “Ready for review” column was used to tasks to be put up for review to make sure everyone is in agreement that the task is completed.
4 Results

This chapter will present the results for the pilot study, implementation and user tests. The pilot study presents the results of the demand analysis, feature analysis and an early prototype developed at this stage. The implementation section presents the various pages of the web application with regard to their functionality and design. Furthermore, the results for the purchase and sales process together with the evaluation made using acceptance tests and user testing.

4.1 Pilot Study

The results of the pilot study contain the analysis of the requirements that potential users indicated they had, as well as a feature analysis of which functions that would be implemented on the web application to fill that need. The compilation of the demand analysis and the feature analysis is presented as tasks in the product backlog and as user stories. In addition to this, the prototype that is the basis for the web application is presented.

4.1.1 Demand Analysis

At the start of the project, a survey was designed to investigate the need for programming assistance as well as the demand for a service where students can offer and receive help within programming. The survey, see Appendix A – Questionnaire, gave 112 responses, see Appendix B – Summary of Questionnaire, of which 41 were from students studying computer science.

Among students that have studied at least one programming class, 77% specified that they would appreciate additional assistance in programming beyond the help provided by the university. According to the survey, 84% of the students who indicated that they need additional help in programming, are also willing to pay for private tutoring.

At the same time, there are students who believe that they have the competence and knowledge to teach programming and who would like to assist other students with programming. Among the students who were surveyed, 46% stated that they had the competence and knowledge to teach programming and 69% stated that they could consider teaching programming if they were to receive financial compensation.

4.1.2 Feature Analysis

During the pilot study, a brainwriting session was conducted to obtain a product backlog, see Appendix D – Product Backlog, with 88 articles as well as accompanying user stories, see provided in Appendix E – User Stories. The items in the product backlog and the user stories formed the foundation of the functionality implemented on the web application during the project. When the project was terminated, 44 of these 88 items had been implemented while the remainder was omitted.
4.1.3 Prototype

During the pilot study, a horizontal prototype was developed to provide a template for the implementation of functionality and design on the web application. The prototype, Appendix C – Prototype, consists of a design scenario combining designs from various existing web pages. Inspiration from existing web pages was taken to visually design the front page, user profile, search results and the navigation bar for the web application and create a target image for the future web application. The inspiration was collected from the web pages of Turo.com, LinkedIn.com and Codementor.io.

4.2 Implementation

The different subpages are implemented with functionality and design to enhance the users’ experience throughout the web application. Below follows the resulting general design, the formations of the different subpages, together with the results gathered from user testing.

4.2.1 General Design

The web application with its subpages follows a general design theme. This theme mainly consists of three colours; space grey, white and yellow as accent colour. There are also additional colours, such as a dark blue in the form of a background picture of stars-in-space and some light greys to enhance content.

At subpages, where information is displayed, the general theme consists of a dark header in a form of a picture and further content on white background at the bottom. In the dark header the main heading is included together with some essential information. Longer texts and information follows in the light body section below. The page ends with a dark footer to distinguish the body content and match navigation top menu. At subpages where the main focus is data input, the design theme consists of a whole background of space grey and white input boxes where text should be written in to clearly display where to write.

Text displayed on the web application are either white if written on dark backgrounds or black if written on light backgrounds. Throughout the subpages and different headings, there are different fonts and text-weights to differentiate content and headings. Larger headings are also in capital letters for further differentiation. If there are any hyperlinks embedded in the text, the text will turn yellow when hovering to indicate that users can press the link.

Other interactive design implemented throughout the web applications are popups. These carry a common design with light grey background and medium dark buttons, which has the same design as the yellow buttons. The popup carries a header which is bald with body text below, all in black. If there are input boxes in the popup these will have a white background to differentiate them from the background.
Similarly to the rounded corners on buttons, most of the objects displayed on the web application have rounded corners with the same radius. The rounded theme follows when displaying pictures as these are displayed as circle images.

### 4.2.2 Start Page

When a user first enters the web application, they are brought to the index page. For most users, when entering the index page, a header covers the whole page, although, depending on the size of the user's browser it might not fully cover the window. The header has a text welcoming the user, as well as two buttons, see Figure 1.

![Figure 1 Header on the index page](image)

The button on the right opens the window for registration when clicked, while the button to the left scrolls the page down to the section “Så fungerar det”, “How it works”.

In the section below the header, the vision of the web application is presented, and longer descriptions on how the application works for a user who wants to receive, or give, help. In the left column the application is explained for those who seek to receive help, while on the right it is explained for those who want to give help. See Figure 2.
The last section on the page is “How it works”. Here, presented in 4 steps, are instructions on how to use the application. Each step is accompanied with a short description, see Figure 3.
4.2.3 Navigation Menu

The navigation menu is composed of a top menu bar and a footer at the bottom. These two components are shown at every page of the web application.

4.2.3.1 Top Menu

The top menu is the main navigation tool and also where the company logo is displayed throughout the whole web application, see Figure 4. To the left, the company logo is set and this acts as a button to the start page. To the right, the main pages on the web application are displayed. The navigation top menu change content depending on if the user is logged in or not.
When not logged in, the top menu includes the tabs About Us, Find Teacher, Login and Register. Find Teacher is a drop-down menu with sub content that will be displayed when pressing “Hitta lärare” (“Find a teacher”), as seen in Figure 4. The drop-down menu includes Search, “Alla lärare” (“All teachers”) and five main programming languages, Java, Python, Ada, JavaScript and C++.

![Figure 4 Top navigation menu when not logged in](image)

When a user is logged in, more content will appear on the navigation menu and some will disappear. Login and register will disappear, while bookings, chat and my profile are added. My profile includes a drop-down menu that will drop down when pressed. This includes a link to the user’s profile and logging out.

When the user is at the top of the page the navigation top menu is transparent and show the page’s header background. When starting to scroll down the page, the top menu gets a dark grey background matching the web applications main colour theme.

### 4.2.3.2 Footer

The footer is placed at the bottom of the page and have the same layout throughout all pages. This includes information that is not needed as often as the information in the top menu. To the right there is an arrow that will bring the scroll the page back to the top, as seen in Figure 5. To the left on the footer there are two links, one to a contact form to the company and one to the terms of service.

Other information on the footer are in the middle where it states that the company have the rights of this page. To the right of the footer the logotype of the online payment process, Stripe, is shown together with the logotypes of the available pay methods. In this case, logotypes for MasterCard and Visa.

![Figure 5 Footer of the web application](image)
4.2.4 Registration

The function for registering a new user can be reached from three different locations; from the index page header's call-to-action register button titled “Registrera dig nu!” (“Register now”), from the hidden login page “Klicka för att registrera!” (“Click to register”), and from the navigation bar button “Registrering” (“Register”) showing up when not logged in. The registration form popup opens on top of the current page, as seen in Figure 6.

The user can either register/sign in with an already existing Google account or create a new account with the application. If the user presses "Sign in with Google", a separate popup which handles the Google sign in process will open.

![Figure 6 Registration popup](image)

All forms use placeholder text showing what should be filled in the fields. All fields are required to proceed from the register form through HTML form validation. For additional password security there are two required fields to enter the password. If the password fields do not match then HTML form validation will be performed automatically by the browser. If the user tries to create an account with a username or email that is already taken, another popup through server side validation is shown telling the user that that username, or email respectively, is already taken. When the user successfully finishes the registration, the user is automatically logged in.
4.2.5 Login

The user can access the login function from several locations. Most noticeable is the link located in the navigation top menu bar, which will display the popup shown in Figure 7, when clicked. The pop-up gives the user the option to sign in using a Google account or through a registered account. The Google login is designed to create a user with the Google account if it is not already linked to a profile on the server, otherwise it will perform a regular login.

![Login popup](image)

**Figure 7 Login-popup**

If the user has manually entered a username and password that is not linked to an existing profile on the server, server side validation will open an error message in the form of a pop-up. If the user has entered valid information, the user will be redirected to the index page with a welcoming message including the users name indicating that the login has been successful.

There is also a separate hidden page for logging in as seen in Figure 8. This is used when the user is trying to access pages which are restricted for users that are not logged in. At this page there is also the possibility to login with either a username or a Google account. If the user don’t have an account there is a link at the end of the page to register as a new user.
The user can log out from their profile by pressing the button “Logga ut” (“Log out”) on the navigation bar.

4.2.6 My Profile

Users can access their own profile page, my profile, through the navigation menu. In My profile the user can view their personal information both which is displayed for other users and information that is only connected to the account and not visible for others. It is also at this page the user can edit their profile, edit their account information and create a teacher profile.

4.2.6.1 User Information

At the top of the page there is a header which include the most basic profile information about the user. The header includes, as seen in Figure 9, the users profile picture, name, username and email. At the bottom of the header there are two buttons, one for profile settings and one to the user’s bookings.
It is also possible to change profile picture directly from my profile header by hover over the picture which will, when pressed, open up a popup asking to choose a picture from the user’s computer, as seen in Figure 10.

The username and email are only visible for the user itself. The name and profile picture are visible for others, regardless if the user is just a student or a have created a teacher profile. If the user has create a teacher profile there will be more detailed information below the header. This information includes the teacher’s rating, reviews, biography, education, work experience, programming language skills and how much the user want to be payed per hour during teaching lessons.

The average rating and number of reviews are displayed in a box together with a button which open up a review list, as seen in Figure 19. The review list will show all the reviews for a teacher. Each review includes title, body, and rating, name of who wrote the review and date and time.
The rest of the information is displayed under each other below the two boxes at the top, as seen in Figure 11. To display the user’s skills in programming languages the row for skills include badges for the programming languages the user can give out help in.

![Teacher information in my profile](image)

**Figure 11 Teacher information in my profile**

### 4.2.6.2 Profile Settings

The profile settings can be entered from a button in the user profile. At the profile settings the user can change their account information including name and email. The rest of profile settings will look different regarding if the user is only a student or have created a teacher profile.

If the user is only a student the profile settings will display name and email together with four buttons, as seen in Figure 12. The user can change their information by writing in the boxes and save with the left button at the bottom of the page. When changing information and saving a confirmation popup will display to inform the user that changes have been done to their profile. To change the password for the account the user can press down the middle button, and a new view will be shown where the user need to specify the new password and repeat it before saving the changes. There is also possible to change the profile picture by hovering over the picture at the top. If the user would like to go back to my profile there is a go-back-button at the downright of the page.
The upper button is for creating a teacher profile. If pressed, a popup will be shown with instructions and input boxes for further teacher information will extend below the current profile settings, as seen in Figure 13. These boxes are mandatory to fill in to create a teacher profile. When the new information is saved, by clicking the save-button, the user will now also be a teacher.
Most boxes in the profile settings form have a text input. The email is required an email format input and to specify programming language skills tick boxes are used for easy settings of skills, as seen in Figure 13. In this version there is five programming languages that the user can choose from.

The upper button at the profile settings as teacher is a button for stop being a teacher. This will remove all teacher properties from the user’s profile and the user will no longer be visible as a teacher. If the user decides to be a teacher again the old teacher information will be saved and pop up automatically as the user create a teacher profile again.

4.2.7 Search

The link to the search function is located in the navigation top menu. It has a dropdown menu where the user can choose to search for pre-selected filters to simplify the process, as shown in Figure 14. Upon pressing one of the pre-selected filters in the navigation menu, the user will be redirected to the search page where all the teachers matching the selected filter are displayed.
The search input at the search page lets the user find teacher by entering parts of the name into an input box. All the users matching the input will be returned and displayed as shown in Figure 15. The search can also be further filtered by applying both text input and a specific language to make the search more precise and tailored for the buyers’ preferences. Below the section with the search function and language option are buttons and information displayed related to the previous search. It gives the user the option to change in which order the buyers are shown. Pressing one of these buttons will rearrange the search result according to rating, highest price or lowest price.

Figure 15 Search page

Figure 15 also shows how a result element is displayed to the user. Every teacher is shown with their profile picture, name, rating, biography, price as well as all programming languages they know. The teacher’s name is also a hyperlink, which is highlighted on hover, and redirects to the teacher’s profile page. If the user has performed a search with parameters that does not match any existing teacher in the database, the user is informed of this.
4.2.8  Teacher Profile

The teacher profile is a page where users can view a teacher, see their skills and price, book a session and give a review on the teacher. This is one of the main pages on the web application as it is the place where bookings between students and teachers take place.

4.2.8.1  Teacher Information

At the top of the page there is a header which include the most important information about the teacher and buttons for different actions. The header includes a profile picture, the name of the teacher and the teachers rating displayed as stars. Under the rating, depending on if the user is logged in or have booked the teacher, buttons will be displayed, as seen in Figure 16. When a user is not logged in, or views their own teacher profile, no buttons are displayed. When the user is logged in, a booking-button will be displayed and after a booking has been carried out, two more buttons will be displayed - one to give a review and one to open up the chat.

![Figure 16 Teacher profile header depending on whether the user is logged in or booked the teacher](image)

Below the header follows more detailed information including price, number of reviews, a biography, education, work experience and programming language skills, as seen in Figure 17. At the top right, a button is displayed which opens up a list with all reviews written about the teacher. The button also states how many reviews that have been written. To the left, the price is displayed in a box. Below follows information written in text, except for skills where the language a teacher is able to teach in are displayed with badges.
At the bottom of the teacher profile page there is also a button for reporting the teacher, which will be displayed when the user is logged in. When pressed, a popup will be displayed and the user can write a complaint about the current teacher which is sent to the administrators of the web application.

4.2.8.2 Rating & Review

When a user clicks the button rate teacher button, “Betygsätt” (“Rate”), a popup emerges, as seen in Figure 18. In the popup the user can select a rating 1 to 5 for the teacher and write a review of the teacher, complemented by a review headline. Neither the headline or review field is required, meaning the user can submit a rating without any accompanying review.
If the user has already made a review of the teacher, the fields in the popup are filled with what was said in the last review. The user can, if they wish, change that text to something new and save the new review.

When a user posts or edits a review or rating, it updates list of ratings, the stars showing the rating, and the box showing how many reviews the teacher has received, without the user having to reload the page.

When a user presses the show reviews button, a list of all the review the teacher has received is shown, as seen in Figure 19. The list is sorted with the most recent review highest and the oldest one last. For every element in the list the user can see the rating, headline and review, as well as who wrote it, what date and time.

When clicked, the show reviews button changes to “Dölj omdömen” (“Hide reviews”) and if clicked again it will close the list of reviews.

Figure 18 Review popup
4.2.9 Booking of Teacher

When the user clicks on the booking button on a teachers profile it opens up a popup in the centre of the screen, see Figure 20. In the popup, the user can specify the different information for the booking. On the popup there are several select option fields and a text field in the bottom. The first option is to choose which of the programming languages known by the teacher to book for this occasion.
After choosing programming language the user can choose which year, month and day the occasion will take. The user can only choose dates that go up to one year into the future, but this parameter can be easily changed within the code of the web application. After choosing the date the user may select the starting time of the occasion, specifying the hour and minute, within five-minute intervals. After selecting starting time, the user has to choose the length of the occasion in terms of hours and minutes. The user may choose up to 4 hours and 55 minutes to book. This, as with the dates, is easily changed because of generalized code has been implemented for these functions.

When the user changes any of the options, except programming language and the length of minutes option, the window automatically updates the available options, in that way the booking won’t conflict with another booking request. Users are only presented with times that no one has booked. If the user chooses a date where all the hours during that day is already occupied by other bookings the window will show a “not available” option, showing to the user that it needs to change the date.

The last thing the user has to do, if they want to, is to send a message to the teacher with any other information regarding the booking, for example describing their specific programming problem.
When the user changes the amount of hours and minutes, the price at the bottom of the popup changes as well. At the end of the window there are two buttons. One for submitting the booking request, to the left, and one for cancelling the booking request, to the right. If the user presses submit, the application will check if all entries are valid. If they are, the booking request will be submitted, otherwise an error message popup will tell the user what’s wrong with their submitted booking request and what they should do.

If everything is correct the booking popup will close and a new booking confirmation popup tells the user they made a successful booking request. If the user clicks on the cancel booking button the booking won’t be submitted and the booking window will close with no further popups.

4.2.10 My Bookings

The My Bookings page consists of two parts, the header and the area in which the user can see and interact with the bookings belonging to them, as seen in Figure 21. The main purpose of the my bookings page is for the user to see and interact with their bookings. In the area located under the header, there are three or four dropdown links, depending on if the user is a teacher or not.

![Figure 21 My bookings page for a teacher](image)

The first dropdown link with the headline “Genomförda bokningar som elev” (“Made bookings as a student”) is used to display bookings the user has made as a student. The user can see bookings that are accepted by the teacher and booking requests that are waiting to be accepted. For each booking, the user can see the order number, teacher, language, date, time, price and additional information, as seen in Figure 22. The user is also able to cancel the booking by clicking a red cancel button. When the information or cancel button is pressed, a message will pop up displaying the relevant information for each action.
The second dropdown link with the headline “Bokningar på dig som lärare” (“Bookings as a teacher”) is only present if the user is also a teacher. For buyers, this option will not be present. When pressing this link, the user is able to see two tables containing accepted bookings and bookings waiting to be accepted. For each booking, the user can see the order number, student, language, date, time and additional information. For bookings waiting to be accepted, the user can choose to either accept the booking or cancel it. For accepted bookings, the user can cancel the booking and see if the booking has been paid or not. When the information, accept or cancel button is pressed, a message will pop up displaying the relevant information for each action.

The third dropdown link with the headline “Avbokade bokningar” (“Cancelled bookings”) displays all bookings that have been cancelled and displays information about the order number, student, teacher, language, date, time, price and additional information. When the information or accept button is pressed, a message will pop up displaying the information for this booking.

The fourth and last dropdown link is the button with the headline “Gamla bokningar” (“Old bookings”). Bookings that have expired are shown under this table. The information regarding order number, student, teacher, language, date, time, price and additional information is once again shown and when the information button is pressed, a message will pop up displaying the information for this booking.
4.2.11 Payment Process

The payment process has been implemented by integrating Stripe Checkout to help with authentication and security. The application acts as a middle hand in the payment and transfers the total amount, except the administration fee, to the teacher. The payment option is shown to the user when the teacher has accepted the booking. If the booking has not been paid the option to pay is displayed to the left in the table as a button, as seen in Figure 23. If the booking already has been paid the status is switched to payed and cannot be paid again.

![Figure 23 Payment at bookings](image)

When the payment button is clicked, Stripe’s checkout pop-up is launched, as seen in Figure 24. The pop-up displays, together with input boxes for information necessary for the payment, the application name, the administration fee and the total price. If the user enters valid information, the payment is processed and confirmed with a pop-up.

![Figure 24 Stripe payment pop-up](image)

When the buyers’ transaction has been confirmed, a receipt is emailed to the email address linked to the buyers’ application account, see Figure 25. The receipt shows which email has been used, to which teacher the payment has been made and the total price.
4.2.12 Chat Function

The chat page is split in two sections. For the complete chat, see Figure 26. On the left side of the page there is a list of users. These are users that the current user has started a chat with. The list is sorted by the most recent message, with the most recent at top and the oldest at bottom. For every user, its picture, name and a preview of the latest message in their chat is shown. The preview shows if the messages was sent by the logged in user or the other. If the message is a received message and the current user has not yet opened their chat, the preview is displayed in bold. Every user has a chat with administrators from the beginning. This gives the user the chance to interact with the chat, before using the chat with other users.
On the right side of the page the whole chat with the most recent message is displayed. The user can open whichever chat they desire, by clicking on another user in the list on the left side of the page. If the chat was read by the user, the preview text changes from bold to regular and does not change back until a new message is received. The chat that is shown has a header with the other user’s picture and name. If the other user is a teacher a link to their teacher profile is accessible by clicking their name or picture. Below the header, all messages are shown, as seen in Figure 26. The scroll starts at the bottom of the message section, showing the latest messages.

Each message is shown in a bubble. The message is inside the bubble while the date and time of the sent message is below the bubble. Received messages is aligned to the left of the chat, with a dark blue background and white text. Messages sent by the user is shown with black text on a soft grey background, aligned to the right.

At the bottom of the chat is a textbox. Any text in the box will be sent as a message if the user presses the button enter on the keyboard, while the box is selected, or by clicking the button. When a message is sent, the list of chats on the left side is updated.

4.3 User Testing

User Testing was conducted after sprint 2 and sprint 3. For each test there was a total of 12 people participating. The results presented will summarize conclusions from the concurrent think-aloud session, as well as the retrospective probing session.

Of the 24 test participants 62.5 % were men and 37.5 % women. 12.5 % were 18-21 years old, 83.3 % were 22-25 years old and 4.2 % were 26-29 years old. Further, all participants of the tests were students at Linköping’s University. However, 71 % were studying Industrial Engineering and Management, 8.3 % studied Information Systems Analysis, 8.3 % studied Energy-Environment-Management Engineering and 12.5 % studied Design and Product Development.

4.3.1 Concurrent Think-Aloud Procedure

During the concurrent think-aloud procedure the notes taken by the observers of the test-participants were widely varied in what the participants were saying, and therefore thinking, during different parts of the test.

One common denominator was that the colours and the general design theme was satisfying, and another that there were navigational problems with the web application. More specifically navigation regarding bookings, and how to create a seller profile in sprint 2, and navigation regarding profile information in sprint 3. In general participants had split opinions regarding the functionality, for example how the booking worked. For complete summary of notes from the concurrent think-aloud the session, see Table 1. The procedure also helped identify a lot of minor bugs, which proved helpful in finalizing the platform on time.
### Summary of notes

<table>
<thead>
<tr>
<th>Area</th>
<th>Sprint 2</th>
<th>Sprint 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive thoughts</td>
<td>• Stylish design in general</td>
<td>• Stylish design</td>
</tr>
<tr>
<td></td>
<td>• Quite easy to navigate</td>
<td>• Everything is easy to find</td>
</tr>
<tr>
<td></td>
<td>• Clear</td>
<td>• Clear message</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Seems reliable</td>
</tr>
<tr>
<td>Negative thoughts</td>
<td>• No confirmation when registering, logging in, etc.</td>
<td>• The concept of the application is not obvious</td>
</tr>
<tr>
<td></td>
<td>• Not knowing if logged in as teacher or student</td>
<td>• Format on inputs should be explicitly expressed</td>
</tr>
<tr>
<td></td>
<td>• Some pages lack visual design</td>
<td>• Start page does not seem scrollable</td>
</tr>
<tr>
<td></td>
<td>• Difficult to locate the chat and bookings</td>
<td>• Booking popup does not seem to be styled</td>
</tr>
<tr>
<td></td>
<td>• Teacher search functionality does not seem to work properly</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Scrawling design</td>
<td></td>
</tr>
</tbody>
</table>

Table 1 Summary of notes from concurrent think-aloud procedure

### 4.3.2 Retrospective Probing

Results from both the quantitative and qualitative aspects from the retrospective probing are presented and summarized below.

#### 4.3.2.1 Quantitative Probing

Results from both the sprint 2 test and sprint 3 test are presented in the Table 2. The result improved in all categories from sprint 2 to sprint 3, going from a 5.1 mean result to a 5.8 mean result. The biggest improvements can be seen in the questions “Everything adds up well on the web application” (5.3 to 6.2) and “I would feel comfortable entering credit card information on the website” (4.1 to 5.3), while the question “The colour-composition is perceived as attractive” barely sees any improvement (5.7 to 5.8).
<table>
<thead>
<tr>
<th>Statement</th>
<th>Average result (scale 1-7)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Sprint 2</strong></td>
</tr>
<tr>
<td>Everything adds up well on the web application</td>
<td>5.3</td>
</tr>
<tr>
<td>The layout is perceived as pleasantly varied</td>
<td>5.2</td>
</tr>
<tr>
<td>The layout is perceived as professionally designed</td>
<td>5.2</td>
</tr>
<tr>
<td>The colour-composition is perceived as attractive</td>
<td>5.7</td>
</tr>
<tr>
<td>I would feel comfortable entering credit card information on the web application</td>
<td>4.1</td>
</tr>
<tr>
<td>Mean</td>
<td><strong>5.1</strong></td>
</tr>
<tr>
<td>Confidence Interval for statement 1-4 (α = 0.05)</td>
<td>[4.6, 6.0]</td>
</tr>
</tbody>
</table>

Table 2 Quantitative probing results

### 4.3.2.2 Qualitative Probing

Summary of main questions from the qualitative probing from both sprint 2 and 3 can be found in Table 3.

In general, the question “What do you think about the platform in its entirety” received positive and similar feedback in both tests, while the questions regarding problems encountered and what could be improved had hugely varying answers. Problems encountered were more drastic in sprint 2, which including major navigational problems and unclear information/instructions on the website. In sprint 3 the problems encountered were usually minor bugs or text that was hard to read. Answers for what could be improved on the web application were also more drastic in sprint 2, where people wished for complete re-designs of some pages. In sprint 3 participants usually nit-picked about some of the details on the web application.
<table>
<thead>
<tr>
<th>Question</th>
<th>Summary of answers</th>
</tr>
</thead>
<tbody>
<tr>
<td>What do you think about the platform in its entirety?</td>
<td>• Good design&lt;br&gt;• Clear purpose&lt;br&gt;• Serious&lt;br&gt;• Professional</td>
</tr>
<tr>
<td></td>
<td>• Legit&lt;br&gt;• Great design&lt;br&gt;• Professional&lt;br&gt;• Overall good</td>
</tr>
<tr>
<td>What problems did you encounter?</td>
<td>• Navigational problems&lt;br&gt;• No confirmation pop-ups after actions on web application. Unclear if anything happens&lt;br&gt;• Unclear what a seller-profile is and when to make one</td>
</tr>
<tr>
<td></td>
<td>• Minor bugs&lt;br&gt;• Hard to read some text</td>
</tr>
<tr>
<td>What could be improved on the web application?</td>
<td>• Re-think “my bookings” tab to be intuitive&lt;br&gt;• The chat-function&lt;br&gt;• Some pages have no design&lt;br&gt;• More information to explain all steps</td>
</tr>
<tr>
<td></td>
<td>• Background image on header&lt;br&gt;• More info. FAQ, about us etc.</td>
</tr>
</tbody>
</table>

Table 3 Qualitative probing from sprint 2 and 3
5 Discussion

In the following chapter the results and methods are discussed.

5.1 Result

In this section the result of the implementation of the web application is discussed.

5.1.1 Pilot Study

Below is a brief discussion of the results of the demand analysis, the feature analysis and the prototype.

5.1.1.1 Demand Analysis

The target for the number of answers to the questionnaire given at the start of the project was reached by margin. This may be due, inter alia, to the survey being designed in a manner that would not considered to be burdensome to answer. Based on the results of the survey, it is possible to determine that there is a need for a service aimed at students who need further assistance in programming and students who can teach others in programming. The web application designed for this purpose was therefore relevant to develop.

5.1.1.2 Feature Analysis

To generate many ideas in a short period of time, brainwriting were used to generate items to the product backlog. It turned out that brainwriting was a very promising tool that generated many ideas. The group chose to write down the ideas in Excel instead of on paper, which is the traditional way, as this facilitated continued work and work with members who were at a distance.

5.1.1.3 Prototype

The prototype presented during the pilot study was used as an inspiration during the development of the web application. The idea of the prototype was that it should provide a broad picture of the functions and the designs desired on the web application. Therefore, the prototype was designed as a horizontal prototype [33]. This resulting in a prototype that was not so detailed, which meant that a large part of the design scenario was implemented on the web application.

5.1.2 General Design

This subsection will discuss the design of the web application in regard to simplicity, diversity, colourfulness, craftsmanship and balance, as these are key aspects of the design covering all pages on the web application.
There was a lot of focus on the design of the platform because of the research questions focus on generating consumer trust through design. The goal to research this connection, is derived from not only the clear correlation between user trust and design. This as can be studied in the theory presented, but also because of the nature of the platform. Since there are both buyers and sellers, each with different relevant pages and requirements, the platform is forced to have an easy-to-use design to function properly. This understanding was achieved by committing to user tests, where users emphasized the design aspects of the platform.

5.1.2.1 Simplicity

In accordance to what Ronald Arnheim described [21], the project initially set out to make the web application as simple as possible, both in regard to the amount of elements and amount of information on screen. After the user tests conducted, feedback was received that there was a lack of information on several pages. There was an internal discussion on the principles of simplicity versus clarity and what is more important. The conclusion of the discussion was that simplicity is more than just minimalism, which is what the project strived for initially. A study brought up during the discussion states that additional features connected to simplicity are clarity, homogeneity and balance [19], which helped with reaching the conclusion. Clarity and simplicity are not mutually exclusive, and after iterations of clarifying the content of the web application, a balance was achieved.

5.1.2.2 Diversity

Questions specifically targeting diversity were included in the user tests, in order to achieve and make sure sufficient diversity on the web application was achieved. Specifically fulfilling aspects, as having a dynamic layout, were considered during development [20]. This resulted in implementing functions such as jQuery Smooth Scrolling and buttons to jump between sections, or back to the top. In hindsight even more dynamic layout functionality could have been implemented in order to improve this further, for example animating entrance of content when scrolling down the web application.

5.1.2.3 Colourfulness

The general design theme tried to follow the core message of the platform, to be a meeting place for programming assistance. The web application therefore tried to follow suite, and give an impression that professional programmers were behind the platform, as the colour appeal of a web application direct contributes to the user’s perception of both trust and satisfaction [25]. This affected the design in several ways.

- When a lot of people started coding, there was a sense of "coolness" to use dark background.
- A lot of people often code in late hours, with the room rather dark. Darker colours reduce the contrast a little bit, and the eyes feel better.
• Darker colours are the default for several editors (e.g. vim, atom) or terminal, and a lot of people associate those colours to programming.

However, as a completely dark webpage does not appear as appealing, the compromise was to have a dark (space grey) header and footer on all pages – and white background on content in between.

All “call to action” buttons are in a deep yellow colour, as chromatic colours are preferable for promoting positive affect and behavioural intention. [26]

Other colour-combinations were tried on the web application, but it was decided that the deep yellow and space grey combination was the best. In hindsight a slightly darker yellow colour would increase visibility, especially on a white background.

5.1.2.4 Craftsmanship

As craftsmanship reflects whether the site was designed with skill and care using modern technologies [20], it was the intent of the developing team to make the web application look as modern as possible. As this was regarded as a crucial aspect of the platform, questions specifically targeting this were used in the user tests.

To achieve this aspect in the design there was a lot of focus on using the latest possible tools in development. Bootstrap 4, Font Awesome 5, jQuery Smooth Scrolling and animated popups are products of this. This goal was achieved relatively well, but could have been further developed by implementing additional functionality and animations for the page to appear even more modern.

5.1.2.5 Balance

There is a correlation between simplicity and balance [22], [27]. Therefore, even without thinking about it, the focus on simplicity made the web application relatively well balanced. Analysing optical weight to evenly distribute objects [29] was not one of the design-aspects in focus during development, even after initial plans to do so. Analysing balance is a more useful tool when forced to have a lot of elements on screen, without the possibility of spreading them out, which this web application did not require at any point, except for the My Bookings page.
5.1.3 Start Page

The start page is always the first impression of a web application for visitors. It is crucial for a web application’s success and efficiency. It is a starting point of the journey around the web application and a strategic asset for marketing goals. It is the first visual and emotional touch to the web application. The general idea was to focus on what would be visually pleasing for visitors, and to have content where to user could absorb the essence of it in seconds. Jakob Nielsen stated in an article: “Homepages are the most valuable real estate in the world. … The homepage is your company’s face to the world. Increasingly, potential customers will look at your company’s online presence before doing business with you — regardless of whether they plan to close the actual sale online.” [43].

There are three elements on the start page. The header, the “our offer” section and the “how it works” section. The idea is that after scrolling through the start page, you know exactly what the platform can offer you as a user, and a step-by-step process on how to achieve that.

The idea with the header was to have complete coverage of the screen, to be visually pleasing to look at, and to fit the general colour-scheme of the web application - forcing it to be a rather dark image. The initial image chosen was more of a better fit in the general theme of the platform, being a picture with coloured strings of code on a dark background. This however, did not fit into our requirements of simplicity on the web application, and was changed into the stars-in-space background used today. Having stars-in-space was also a good coloristic match to the space-grey background colour used in the header and footer.

Below the header is the “Vårt erbjudande”, or “Our offer” section that describes the use of the platform, for students and for teachers. This information was initially the vision of the platform, presenting what key aspects the developing team focused on. It was decided that even though such information is relevant to have on the web application, it should not be the first information a user sees. That information was moved to the “About us” page, and replaced with information that is far more crucial for a user to read initially, information about what the platform can offer the user.

The last section on the page is “Så fungerar det”, or “How it works”. The initial idea was to have a split timeline between students and teachers, students on the left and teachers on the right. This was decided against in early stages, but in hindsight it could have made information clearer, as students and teachers do not necessarily go through the same path on the web application.

The general design of the start page was developed early, and positive feedback was received through the user tests, resulting in no major changes throughout development.

5.1.4 Navigation Menu

The navigation top menu and footer are essential for the usability of the web application. These two have changed drastically through the implementation period, mainly after doing user tests.
In the early state of the implementation the navigation top menu contained different content. After doing user testing the result showed that the testers had a hard time finding the chat and bookings pages. This resulted in these pages moved up directly to the navigation top menu to increase the usability. The company logotype, however, stayed to the left of the navigation top menu since the start of the implementation. This based on our theory which states that trust for a web application increase if the logotype and full company name are clearly visible through every page on the webpage. [6] [7]

The footer content is mainly based on theory. The presence of third party certification and other connections to the rest of the web is very important in order for users to feel trust when visiting a web application [6] [14]. It is also important for user trust that it is clearly visible in what ways payment can be done [7]. Therefore, the logotype of the payment process Stripe, together with the logotypes for the pay methods, MasterCard and Visa, was clearly displayed at the footer to always be visible at all pages.

The terms of service and contact form in the footer are also implemented based on theory as it states that terms of service need to exist to give e-commerce trust for users. This is also one of the requirements for an e-shop to be given the trustworthy certification “Trygg E-handel”. The certification requirements also state that users should be able to send complaint to the company when there are problems with the orders regardless if the company of the webpage sell the actual service. Therefore, a contact form was implemented as an easy and quick way to get help. [7]

5.1.5 Registration

In the early stages of development of the web application, a page with a registration form was loaded for the user to fill in. Later, this was changed and substituted with a popup window. The main reason for this was to improve the visual appeal, but it also contributed to faster loading time since the page did not have to refresh. There is a clear connection between the visual appeal and the perceived trust of a web application [19] [20]. Thus, the design with a popup was preferred in an effort make the web application more trustworthy.

Furthermore, a button for register using a Google account was added above the form. The intention of this was to additionally increase the trustworthiness of the web application in accordance with theory regarding third party involvement on the web application. The Google login implements provided functionality and design developed by Google to the application. [5] [6]

The register popup was also provided with input controls to ensure the user entered valid values. All these features were implemented in the register form popup in an attempt to gain further trust with the user.
5.1.6 Login

In the early stages of the project there was a page designated for logging in, and if the user wanted to log in they had to go to the designated login page. In later stages of development, the login function was reworked to open as a popup. This was more visually appealing and did not require the whole page to reload, streamlining the process.

The login page was, however, not completely removed. Although all links for logging in on the web application opened the popup, users can still be redirected to the login page. In the case where a user tries to enter a page which requires the user to be logged in, for example the chat page, the user is instead redirected to the login page.

Users can choose to login using their Google account. The goal with this, except for being convenient, is to generate trust. Since Google is one of the most well-known brands in the world, displaying their logotype and implementing their functions will generate trust from the consumer. [5]

5.1.7 My Profile

The user profile is an essential page of the web application and makes the user able to view what information is shared with other users. This is especially important for users having created a teacher profile, as the information being displayed and edited under my profile is what the teachers use to market themselves to students with. The design on profile settings is differentiated from my profile information to prevent users from mixing up these two pages.

The user tests provided important feedback in developing the layout for the my profile page. The previous version of the page contained all information regarding the users’ bookings as well. This information was moved to a separate page for additional clarity regarding navigation.

During the user testing a demand for the ability to stop being a teacher and go back to only be a student was presented. Therefore, the button for removing a teacher account was implemented.

5.1.8 Search

The search functionality was designed to facilitate a smooth and simple filtering process for the users to quickly find a teacher matching their unique criteria. In regard to consumer trust, teachers rating is displayed among other information in the search results. To promote trustworthy teachers and to make it easy for users to overview the rating among teachers – functionality was added to sort the search results after rating, in a descending list. Displaying the rating clearly and letting it affect the display order can be assumed to have a positive effect on the consumers perceived trust for the web application. [12]
### 5.1.9 Teacher Profile

The teacher profile is one of the most essential pages on the web application as it acts as both selling page and booking page. This is both where teachers can show what type of help they can give and where students are able to read about different teachers' skills and reviews.

One out of four key principles for evaluating trust on a web application is to have an upfront disclosure. On the web application and at the teacher profile this was done by clearly show the price of the teacher in the profile, before a booking is done, in a separate box at the top. Therefore, if users can easily see the price displayed before taking any decision of purchase the trust of the web application increases. Hence the price was placed at the top of the teacher information. [14]

According to the trustworthy certification “Trygg E-handel”, one requirement for a web application to be seen trustworthy is if users can leave complaints or report to the web administrators if there are any problems with the service or a teacher [7]. Due to this the report button is visible at the bottom of the teacher profile page as soon as a user is logged in and views another teacher. This, to give the user possibility to directly report the teacher, but also not be able to report a teacher without any interaction between user and teacher. It was brought up for discussion whether the report button should be visible directly after login or after a booking was made. The result was that a user should be able to report a teacher as soon as the user visit the teacher profile if the teacher has written or showed any inappropriate information in their profile.

### 5.1.10 Rating & Review

As soon as a booking has been placed, between a user and a teacher, the user can make a review of the teacher. The teacher can also give the user a review, given that the user also is a teacher.

Implementing a rating system is a great tool to create trust between the users. Those ratings can then act as guideline for other users for what to expect from the teacher. As a result, this can lead to the users feeling more trust towards the web application. Theory also states that trust in a web application is gained by social factors such as recommendations and reviews. [11] [12]

In the early state of the implementation all users could leave reviews about any teacher. After implementation it was realised that this was not the best way to have the function. This meant that a user could leave reviews to improve, or worsen, the rating of a teacher without having any interaction with the teacher at all. It was discussed to have the user only being able to leave a review of a teacher after they had a lesson together.
It was later realised that, since users can start chatting as soon as a booking has been made between the two of them, the two users can have interaction with each other before a lesson. If a teacher were to act badly to their student before their lesson, the student would have to not only have the lesson it does not want, but pay for it, to be able to leave a review of the teacher.

Because of this the conclusion was made that it would be best if a user can leave a review about a teacher as soon as a booking has been placed and can change that review whenever they want in the future.

5.1.11 Booking of Teacher

When the booking system was first developed it was discussed whether having a calendar showing what times are occupied or having the occupied times removed when booked. As previously mentioned fewer elements on the screen increase the simplicity of the application which in turn might increase the trustworthiness of the application as well. [12] With this in mind the choice was made to implement a booking system where occupied times disappear. This was in part because it was deemed as more user-friendly but also easier to implement according to the knowledge and experience of the developers.

This system was implemented by sending all the bookings a teacher had to the user trying to book. Through this way the server could filter the different dates and times to remove occasions that were occupied. However, this may affect user experience as when the bookings start to pile up, more and more bookings need to be checked to filter options. When multiple users try to book at the same time, making the server go through a lot of demanding computations, this could potentially create a bottleneck for the system.

To solve this potential bottleneck another solution would have been to have a database table where the available times to make a booking request would be stored with all possible combinations. The downside is that the server would require larger memory if there would be an increasing number of teachers, due to much memory space being used to store data which is not useful.

This would mean that a size of $365 \times 24 \times 12 \times 4 \times 12 \times 7 \times 4 = 141,281,280$ bytes $= 141.3$ MB would be the worst-case table size stored per teacher, assuming every integer is 4 bytes. The downside is that the server would require larger memory if there would be an increasing number of teachers.

The server would still be constrained during insert as the demanding filtering process is conducted but the process would not be as demanding, as it would only need to filter out times connected to that booking request, compared to before where it could potentially compute filters for hundreds or thousands of bookings. Also, when the user opens the booking window the user would only request the available times and the server would deliver without doing any hard work as the filtering would already be done. This is important as it is in this interactive process that it should work as fluently as possible.
However as there was not any performance issues at the current stage of development this shift to a massive database was not developed but plans are there to develop it if it is deemed necessary in the future.

5.1.12 My Bookings

The My Bookings page is the primary tool for the users to view their bookings/booking requests without looking at their confirmation email. During early development, all booking tables were visible as soon as the user entered the My Bookings page. This created an overwhelming and confusing impression making it difficult to locate relevant bookings. After internal discussion of this issue, a decision was made to simplify the tables. The decision to simplify the design was made to improve both the user experience and to build more trust. As previously stated in, features connected to simplicity are clarity, homogeneity and balance. [14]

With simplicity in mind, a design was created which hides all tables until the user selects a table to view. The white background with a centred table adds to the homogeneity and balance of the layout, while the selected table pops out as the header changes from black to yellow and thereby provide more clarity.

5.1.13 Payment Process

The payment process can be perceived as one of the most critical events during a consumers’ experience of the web application. It requires the user to share sensitive information related to their personal finance. Due to this, several factors that increase trust was implemented, such as sending an order confirmation and clearly indicating which payment method that are available. [7]

Instead of implementing a payment method from scratch, API:s for payment such as PayPal or Stripe were discussed to simplify the process as well as generating consumer trust by the use of well-recognised brands. [5]

Initially, PayPal Express Checkout was researched and tested to integrate with the web application. However, due to a more complicated sandbox-testing environment, it was decided to use Stripe Checkout. Stripe’s API provided functionality such clearly indicating taxes, administrative fees, receiver, total price, authentication service and generating receipts which all are necessary components for generating consumer trust. [7]

The payment process requires both parties, both the teacher and the student have to accept the booking to initiate a payment. This functionality is both to simplify for the user and the web application administrators since it prevents unnecessary transaction to be completed in case the teacher denies a booking request.
5.1.14 Chat Function

The chat is the primary pace for users to communicate with each other. In earlier stages of development, the chat was visually designed as email. Users saw a long list of either all their received messages, or all their sent messages. The messages had a subject and a body and the user could only view one message at a time. After discussion in the team, we came to the decision to drop this layout, for a more favourable one.

As mentioned, users find websites more trustworthy if the design of the web application is similar to those of well-known web sites. Because of this the team chose to implement a layout that draws inspiration from several famous chat applications.

Furthermore, to build more trust, we tried to make it simpler. [19] The layout has messages colour coded, showing a clear difference between messages sent and received by the user. The placing of the messages differ as well, with sent messages placed to the right, and received to the left. Giving the user even more clarity in which message is which. The layout is the same whoever the user chats with, keeping a homogeneous structure.

5.2 Method

The following section will discuss the different methods used during the development parts of the project, source criticism and validity of the study.

5.2.1 Demand Analysis

One thing to note is that the pilot study lacks demographic questions, other than what the person is currently studying. At the time of the pilot study it was the project groups judgement that no further demographic questions were necessary. This might however somewhat limit the weight of the conclusions, as it hinders future replicability. Including questions such as age and sex would provide further replicability.

The methodology of conducting the pilot study can be regarded as sub-optimal. Even though the pilot study has good coverage when surveying, very little scientific studies regarding how to optimize surveys were studied before conducting the pilot study. Therefore it is clear that with better preparations the pilot study might have had better credibility.

5.2.2 Feature Analysis

The ideas that were presented during brainwriting session were divided into different categories in the product backlog to make it more orderly. It turned out that many items were modified, and a lot of things were removed completely during the course of the project, as there was no time to implement additional functionality. Among other things, the categories miscellaneous and marketing were completely removed.
The biggest advantage of having a product backlog was that it was easy to use when it was time to decide which tasks that would be taken care of next and to check what features that already have been implemented.

5.2.3 User Testing

The tests conducted provided great feedback and showed areas of improvement for the platform. Using both concurrent think-aloud procedure and retrospective probing proved to be a wise decision as the methods highlighted different aspects of the web application. The concurrent think-aloud procedure highlighted pain-points for the users, often during navigation. Misconceptions identified during the test turned into actionable redesign recommendations [38], which proved an effective way of solving the navigational problems of the web application. Retrospective probing followed up on functionality and design in a more general way, allowing to get an understanding of which expectations were met and which were not.

Since usability is of great importance for a website to be perceived as trustworthy [16] [17] this was one of the key aspects when developing the user tests. The five elements defining usability [18] were used as a foundation when generating the task log and qualitative part of the retrospective probing. This proved to be a wise decision given all feedback that later were used to improve the application immensely.

The lowest average result of all quantitative probing questions was “I would feel comfortable entering credit card information on the website”, with a result of 4.1/7 in sprint 2 and 5.3/7 in sprint 3. Even though it is disappointing receiving the lowest result on perhaps the most important question, it is worth noting that the result is not bad in any way. Using a Likert scale of 1-7, a rating of 5.3 translates roughly to “corresponds a lot” to the statement in question. With that in mind the overall ratings of all questions asked are very high.

The mean results of statements 1-4 of the quantitative probing after both sprint 2 and sprint 3 have a value of more than 4.5. Since 4.5 is the benchmark for when a user perceives a web application as aesthetical appealing it is likely that users experience the web application as rather positive with regard to the visual aesthetics. [40] This is further enhanced by the confidence interval calculated for statement 1-4 which suggests that the values indeed are higher than 4.5 on the level $\alpha = 0.05$ both after sprint 2 and sprint 3. Although, it is not possible to say whether the difference of the result of the quantitative probing after sprint 2 and after sprint 3 is significant on the level $\alpha=0.05$. For example, we cannot with certainty say that the visual designed was improved during the implementation phase in-between the user tests. However, as previously stated, the recommendations when using these four statements is to use at least 20 participants per test [40] and these recommendations were not followed, mainly due to lack of resources, which in contrary to the confidence intervals might imply that the tests are invalid.
The fifth and last of the five statements used in the quantitative probing did not concern the visual appeal of the web application but rather the perceived trust a user felt when using and navigating on the web application. The mean of the results of this statement after sprint 2 was 4.1 and 5.3 after sprint 3, which is a significant improvement on the level $\alpha = 0.05$. This implies that improvements and developments made in between the sprints did in fact make the web application more trustworthy.

The executions of the tests were in accordance to the characteristics of usability that Dumas and Redish established. [35] If the tests would have been complemented with additional points of measurement, using quantitative measurements on how the test participants navigated through the web application would have been a great tool in making a better, more easily navigated web application, but would not contribute to the research question in focus, to generate consumer trust.

A common trait in both user tests was that the test participants were chosen with laziness, which resulted in a majority of participants being friends to the developing team, causing some bias. Most participants are from the Industrial Engineering program, resulting in a homogenous testing group. Furthermore, the general IT-knowledge of these participants is probably higher than the average real user of the platform. Reviewing the participants, it is clear that the participating test groups should have been more heterogeneous and less biased in their testing.

5.2.4 Source Criticism

Some of the sources regarding perception on design are from the middle to late 1900s. As technology have developed extensively the latest 20 years the design and formatting on web application have changed and improved throughout the latest years. Therefore, the perception on web design may be different today in year 2018. It is also worth noting that the target group of the web application are in their twenties and therefore are raised through an IT saturated society. This may result in different needs and expectations, on what makes a web application feel trustworthy and have an appealing design, than what they were 50 years ago.

Furthermore, when discussing trust and visual design the perception may differ depending on personal preferences and experiences and therefore not always follow theory. As academic theory may be criticised for not take in consideration personal preferences, user testing may also show biased results. This is due to that only a few people do the user tests each time and can therefore not cover all different types of user experiences.

5.2.5 Validity

During development of the web application, the validity of the work was based on the user tests. There are two major factors that may affect in the results from user tests. These are how the test is structured, what the users are told to do, in what order, and how the questions are asked to the test user.
During testing, the users were given tasks to complete. When all tasks were done, the user had navigated all pages on the web application and by that also gotten a complete overview of the application, which is relevant for the questions asked later. While navigating the web application, test users are encouraged to say all their thoughts out loud, to give their unfiltered opinion. [37]

After the user finished the tasks given, they are asked a set of questions. The questions are of two different types; qualitative and quantitative. The qualitative questions are designed to be open, and the test user is asked to motivate their answers. While the quantitative questions ask a user to rate different aspects of the application, using Likert-style questions [30].

The research question of this report is focused on how to generate trust from the consumer. Therefore, the questions in user testing must reflect that. These questions mostly cover aspects connected to visual design, which as a big underlying factor to how trustworthiness in a web application is perceived [19] [20]. But some also ask users directly how they perceive the application as trustworthy. By keeping questions relevant for the research question, the validity is deemed to be high.

The validity of the results found from user testing could be questioned because of the number of participating test users. While it is recommended to use at least 20 users at testing, only 12 users were used. But the answers from test users where rather similar, showing users had similar experience. Even if there were more test users, the outcome of the tests would show similar results.

Another way to improve the validity would be to ask more focused questions, concentrating on singular aspects of the application. This would on the other hand take longer time, making it harder to motivate test users to be focused all the way through. Analysing the answers would have taken more time as well.

Another problem that can occur is that the more focused questions are, the easier they are to misunderstand. This could lead to the test user’s answers contradicting each other, making them somehow useless for analysing. Because of this issue raised, it was decided, that the questions and tasks given were the optimal choice.

5.3 Work in a broader context

As the web application aims towards college students and is connected to their education there are some ethical aspects that can be brought up for consideration. One large potential issue includes the possibility for students to help other students to cheat on projects connected to their education. By connecting students through the web application this also gives a platform for students to pay other students to do their assignments as this is not regulated by the administrators.
Another aspect of this is that students pay to get further help and education. As a university education in Sweden is financed by the Swedish government, this might bring up issues regarding students that cannot afford the service. Students that can afford the service of the web application can hence get an advantage in programming assistance, resulting in better knowledge and results.

Furthermore, the web application is aimed towards Swedish speaking students, which exclude international students from the application and service.
6 Conclusion

In this chapter the final conclusions of the study are presented.

6.1 Aim and Research Questions

The research question “How can a web application with the purpose of coordinating programming assistance be designed to generate consumer trust?” has been answered by implementing and evaluating the web application, using the theory as guidance.

Firstly, theory regarding how to achieve consumer trust was gathered and was then used as the foundation when developing the web application. It was found that several design factors are involved when it comes to how the user perceives trust.

Thereafter, the application was further developed, visually, with a focus on simplicity, diversity, craftsmanship, balance and colourfulness. Trust was generated by taking inspiration from the visual design of well-known web applications, that users already know and feel comfortable with. Furthermore, inclusion of key visual elements, was made, such as company logo and available payment options.

To further instil trust, a system for ratings and reviews between users were implemented, as well as APIs developed by well-known brands, such as Stripe Checkout and Google sign-up. Finally, a page to contact the company, as well as another page presenting the terms of the service were added.

Given the results from user tests, the application is deemed trustworthy and successful in generating consumer trust. Therefore, a conclusion can be drawn. By using theory regarding consumer trust as the foundation when developing a web application and by implementing the design elements stated above, a web application can be developed to generate consumer trust.

6.2 Generalizability of the Study

The study has been made by students at Linköping University, and all participants at the user testing sessions have been students at Linköping University as well. It is therefore the project groups’ perception that the results and conclusions are representative for students at the university. Considering the demographic similarities between university students at different universities in Sweden, it is highly probable that the results and conclusions are representative of all university students in Sweden.

However, to generalize the results and conclusions to non-students, students in other countries and older generation’s would be a big leap, as these demographic segments could have significantly different perceptions on consumer trust, and what is important in this regard.
6.3 Recommendation for Future Studies

Due to the limited resources and time, the user tests were somewhat unspecific regarding which aspects were key in generating consumer trust. For further studies, it is recommended to perform more user testing focusing on isolating each component and evaluating its contribution to distinguish and compare the components efficiency in generating consumer trust.

To get more input from the user tests in regard to consumer trust, more questions reflecting different aspects could be added. Currently, the trust perceived mainly focused on how the users felt giving out their credit card information. A wider perspective could be applied to evaluate their perceived trust of the web application.

As concluded, in order for further generalizability of the study, a more eclectic mix of participants can be recommended for future testing to get a higher degree of generalizability for the results and conclusions.
7 References


Appendix A – Questionnaire

Marknadsundersökning - Privat handledning inom programmering

En marknadsundersökning för kandidatarbete gällande privat handledning inom programmering för studenter.

*Obligatorisk

Vilken sektion tillhör du? *

- D - Datateknolog
- I - Industriell Ekonomi
- LING - Högskoleingenjör
- M - Maskinteknolog
- MatNat - Matematisk- Naturvetenskap
- SAKS - Systemvetenskap
- TBI - Teknisk Biologi
- Y - Teknisk Fysik
- Övrigt: ______________

Har du någon gång läst en kurs inom programmering? *

- Ja
- Nej
OM JA: har du någon gång behövt hjälp inom programmering utöver hjälpen som tillhandahålls i kursen?

○ Ja
○ Nej

Hur mycket skulle du vara villig att betala per timme (kr/h) för privat handledning? *

○ Mindre än 50
○ 51-100
○ 101-150
○ 151-200
○ 201-250
○ Jag är inte villig att betala

Har du kompetens att lära ut programmering? *

○ Ja
○ Nej

Skulle du kunna tänka dig att hjälpa andra studenter med programmering om du fick betalt? *

○ Ja
○ Nej
OM JA: hur mycket skulle du vilja ha betalt per timme (kr/h) för att ge ut privat hjälp inom programmering?

☐ Mindre än 50
☐ 51-100
☐ 101-150
☐ 151-200
☐ 201-250
☐ 251-300
☐ Över 300
☐ Jag behöver inte ta betalt
Appendix B – Summary of Questionnaire

Vilken sektion tillhör du?
112 svar

Har du någon gång läst en kurs inom programmering?
112 svar
OM JA: har du någon gång behövt hjälp inom programmering utöver hjälpen som tillhandahålls i kursen?
110 svar

Hur mycket skulle du vara villig att betala per timme (kr/h) för privat handledning?
87 svar

Har du kompetens att lära ut programmering?
112 svar
Skulle du kunna tänka dig att hjälpa andra studenter med programmering om du fick betalt?
112 svar

OM JA: hur mycket skulle du vilja ha betalt per timme (kr/h) för att ge ut privat hjälp inom programmering?
89 svar
Appendix C – Prototype

Front Page

Front page is info page, for easy access on how to start being a user. Based on Turo.com’s about page

Information on how to be a seller and buyer on the same page. Split the middle.

Button that redirect you to become seller/buyer, placed into start view.
User profile

Based on user profiles of LinkedIn.com and comentor.io

Picture and name centralised when entering page.
Rating shown, and review can be accessed, without having to scroll
Price listed in the open
Small bio where the user can describe themself
Areas which the user is proficient in is shown
Relevant work experience is shown
Relevant education is shown
Other relevant merits are shown
Search results
Based on search from codemento.io and turo.com

Picture, name and rating shown in result

Specified button for viewing profile

Search filter shown on search page
Navigation bar

Simplistic navigation bar based on Turo.com

Dropdown menu

Button for search on the navigation bar
# Appendix D – Product Backlog

<table>
<thead>
<tr>
<th>Area</th>
<th>ID</th>
<th>Prio</th>
<th>Description</th>
<th>Functionality</th>
<th>Design</th>
<th>Sprint 1</th>
<th>Sprint 2</th>
<th>Sprint 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Payment</td>
<td>1</td>
<td>1</td>
<td>Working system where teacher enters price per hours</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>User can pay after booking</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.1</td>
<td>2</td>
<td>User can view their history of booking payments</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>*</td>
<td>Automatic transactions from saved card info</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>*</td>
<td>Customer can demand refund if not satisfied with lesson</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>*</td>
<td>Teachers can offer volume discount</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>3</td>
<td>Transaction is reserved in case either buyer or teacher cancels meeting</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Booking</td>
<td>7</td>
<td>*</td>
<td>Teacher can post timetable with available times</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>1</td>
<td>Order confirmation on web site and by mail</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>1</td>
<td>Teacher gets order confirmation</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1</td>
<td>System for booking a teacher</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>*</td>
<td>Show number of free slots on group lessons (given group lessons are implemented)</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>2</td>
<td>Buyer can specify exactly what they want help at booking</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>3</td>
<td>Buyer can see their bookings in a table</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>*</td>
<td>Buyers can make reoccurring bookings</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>General</td>
<td>15</td>
<td>1</td>
<td>Database to store and manage users</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Functionality</td>
<td>16</td>
<td>*</td>
<td>Group lessons</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Communication</td>
<td>17</td>
<td>1</td>
<td>Buyers and teachers can send messages to each other after booking for the first time</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>*</td>
<td>Forum for help available</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>*</td>
<td>Skype, TeamViewer or similar software</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>*</td>
<td>Subscription mail with new teachers</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>4</td>
<td>Form for contacting administrators</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buyer</td>
<td>22</td>
<td>1</td>
<td>Database to store and manage buyers</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>*</td>
<td>Review system where buyers are graded</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>*</td>
<td>Profile with picture, description and skills listed</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>*</td>
<td>System for reporting other buyers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>*</td>
<td>Can block teachers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>*</td>
<td>Different layout after login for buyers and teachers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>*</td>
<td>Awards for teachers that register and book their first lesson</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>*</td>
<td>Reward system where users can collect point that can be used as currency</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>*</td>
<td>Free material to attract new user</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>*</td>
<td>Users can invite friends to earn website currency</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>*</td>
<td>Get one hour free when you make first transaction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>1</td>
<td>Different categories/tabs for different programming languages</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>Designated page for how the site works with interactive flowchart</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>2</td>
<td>User page different for buyers and teachers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>*</td>
<td>Teacher sorted by competence (language/classes)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>3</td>
<td>Teachers can be sorted by price</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>*</td>
<td>Show “Bestselling” teachers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>*</td>
<td>Web site available in both English and Swedish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>*</td>
<td>Recommendations based on bookings</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>1</td>
<td>Basic contact information is required for registration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>1</td>
<td>User has to enter name</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>1</td>
<td>User has to enter email</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>*</td>
<td>“not a bot” verification</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>*</td>
<td>General contact information is shown on teacher’s profile</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>1</td>
<td>Teachers has a biography that is shown on their profile</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>1</td>
<td>Profile for teachers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>2</td>
<td>Rating system, 1-5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48.1</td>
<td>3</td>
<td>Teacher's rating is shown with stars</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>2</td>
<td>Teacher can specify their knowledge in different programming languages</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>49.1</td>
<td>2</td>
<td>Teachers programming knowledge is shown in profile</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>2</td>
<td>Show the teachers work experiences</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>*</td>
<td>Teacher can block buyers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>*</td>
<td>Review system with a selection of pre-determined comments</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>3</td>
<td>Review system with custom comments</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>*</td>
<td>Buyers can save teachers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>3</td>
<td>Teachers have profile pictures</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>*</td>
<td>Teacher can list their works, for example link their Git</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>3</td>
<td>Show how many reviews a teacher has received</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>3</td>
<td>Report system for teachers</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>4</td>
<td>Users can specify their reasons for report</td>
<td>F</td>
<td>F</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>*</td>
<td>List different languages the teacher knows (not programming languages)</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Search**

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 61 | 1 | Search where you can find teachers | F | F | X |
| 62 | 2 | Sort teachers by average grade and price | F | F | X |
| 63 | 2 | Filter search for programming languages | F | F | X |
| 64 | * | Search recommendations of similar teachers’ classes etc. |   |   |   |

**Miscellaneous**

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 65 | 1 | User is told what is required to use the service | F | F | X | X |
| 65.1 | 1 | CodeBuddy's offer is presented on the index page | F | F |   |
| 65.2 | 2 | The Codebuddy team is shown on the website | F | F |   |
| 65.3 | 1 | Navigation bar has dropdown menus | F | F | X |
| 66 | 1 | An info page for how to become a teacher and what is required | F | F | X | X |
| 67 | * | Tutorial video that explains the website |   |   |   |
| 68 | 2 | Customer support | F | F | X | X |
| 69 | * | Grading system is divided in several categories |   |   |   |

81
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>70</td>
<td>*</td>
<td>Different info pages for teachers and buyers</td>
</tr>
<tr>
<td>71</td>
<td>*</td>
<td>Customer support on how to install Git</td>
</tr>
<tr>
<td>72</td>
<td>2</td>
<td>Form for contacting customer support</td>
</tr>
<tr>
<td>73</td>
<td>*</td>
<td>Recommendations on what software to use</td>
</tr>
<tr>
<td>74</td>
<td>*</td>
<td>Student mail or id is required to register</td>
</tr>
<tr>
<td>75</td>
<td>*</td>
<td>Buyers can post entries looking for teachers</td>
</tr>
<tr>
<td>76</td>
<td>*</td>
<td>Administrator review of users with bad reviews</td>
</tr>
<tr>
<td>77</td>
<td>*</td>
<td>Bonus points after completed lesson</td>
</tr>
<tr>
<td>78</td>
<td>*</td>
<td>Extended IT-support, help to set up computers etc.</td>
</tr>
<tr>
<td>79</td>
<td>*</td>
<td>Targeted towards upper secondary school students as well</td>
</tr>
<tr>
<td>80</td>
<td>*</td>
<td>Removal of cheating and plagiarism</td>
</tr>
<tr>
<td>81</td>
<td>*</td>
<td>Recommend help for coming classes, based on what classes are being taken</td>
</tr>
<tr>
<td>82</td>
<td>2</td>
<td>Teacher sets their price per hour</td>
</tr>
</tbody>
</table>
# Appendix E – User Stories

<table>
<thead>
<tr>
<th>Area</th>
<th>ID</th>
<th>User story</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Payment</strong></td>
<td>1</td>
<td>As a teacher I want to decide my own price</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>As a buyer I want to not be forced right away after booking</td>
</tr>
<tr>
<td></td>
<td>2.1</td>
<td>As a user I want be able to see all payment I have done on the website</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>As a buyer I want to not have to enter card information every time I book a lesson to save time</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>As a buyer I want to be able to get my money back if the lesson was not good</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>In order to secure several bookings at once as a teacher I want to be able to offer volume discount</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>In order to feel safe as a buyer I want my money to not be drawn until my lesson is complete</td>
</tr>
<tr>
<td><strong>Booking</strong></td>
<td>7</td>
<td>As a teacher I want to easily show when I am available</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>As a user I want to receive confirmation of my booking</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>As a teacher I want to be sure when I am booked</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>As a user I want to be able to book a teacher</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>As buyer I want to be able to see the available slots on a group lesson because I want to be prepared on how many people will be involved</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>As a buyer I want to be able to tell my teacher what I need help with because I want them to be able to prepare before lesson</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>As buyer I want my bookings listed in an easily understandable way</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>As a buyer I want to book reoccurring lessons in a given interval</td>
</tr>
<tr>
<td><strong>General Functionality</strong></td>
<td>15</td>
<td>As user I want my account details stored so that I do not have to make a new one every time</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>As a teacher I want to save time by showing several buyers at once</td>
</tr>
<tr>
<td><strong>Communication</strong></td>
<td>17</td>
<td>As user I want to send messages to the other part of my booking</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>As user I want a platform to discuss website and common problems</td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>As user I want recommended software to use</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>As teacher I want help with getting advertised. As buyer I want to see the new options new teacher give</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>As user I want to be able to contact website administrators</td>
</tr>
<tr>
<td><strong>Buyer</strong></td>
<td>22</td>
<td>As user I want my account details stored so that I do not have to make a new one every time</td>
</tr>
<tr>
<td></td>
<td>23</td>
<td>As teacher I want to see what to expect from a buyer</td>
</tr>
<tr>
<td>24</td>
<td>As teacher I want to see what to expect from a buyer</td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>--------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>As user I want to be able to contribute to the removal of bad users</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>As buyer I want to be able to block teacher because that makes me feel safe</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>As user I want the website tailored to the end of the service I am using</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>In order to be motivated as a teacher I want rewards</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>As user I want rewards because they will motivate me to use the service more</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>As potential user I want free material because it will attract me towards the service</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>In order to be motivated to invite friends I want rewards</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>In order to be motivated as a buyer I want rewards</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>As buyer I want the categories on different tabs because that makes it easier to navigate the website</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>In order to understand the website and service I want an info page</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>As user I want user profile pages designed differently because different things are important to show for teachers/buyers</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>As buyer I want pages with teacher with specific competence listed so that I can see all teacher that know what I want</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>In order to make a decision on cost I want to sort teachers by price</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>As teacher I want reward for being good. As buyer I want to see the teacher that are best</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>As user I want to be able to use the website even if I do not know Swedish</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>As buyer I want recommendations on which other bookings I could do</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>As user I want my contact information saved because I want to be contacted by the website</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>As user I want my name saved because having it listed will make other users relate to me more</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>As user I want my contact information saved because I want to be contacted by the website</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>In order to not overflow the website with fake accounts I want as a user a verification process</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>As buyer I want to know how to contact a teacher</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>As buyer I want to see who the teacher is</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>As buyer I want all the teachers information listed on a dedicated page because that makes the info easy to take in</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>In order to know what to expect I want as a buyer a rating of teachers shown</td>
<td></td>
</tr>
<tr>
<td>48.1</td>
<td>As a teacher I would like my rating to be easily comprehensible</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>As buyer I want to easily see what the teachers can help with</td>
<td></td>
</tr>
<tr>
<td>49.1</td>
<td>As a teacher I want my knowledge within programming to be shown</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>50</td>
<td>As teacher I want to show what I have done before</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>In order to feel safe as teacher I want to be able to block buyers</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>As buyer I want an easy system to make reviews</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>As buyer I want to be able to leave reviews telling exactly what I want</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>As buyer I want to safe my favourite teachers because I want to easily access them again</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>In order to feel safe as a buyer I want the teacher to show their face</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>As teacher I want to show what I have done before</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>In order to know how general a teacher’s grade is, I want to see how many reviews they have received</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>As user I want to be able to report another user for misconduct</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>In order to make a serious report I want to as a user be able to give a reason for reporting</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>As buyer I want teachers to list what languages the know because I do not feel comfortable with all languages</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>As user I want to be able to search for specific teacher</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>As buyer I want to be able to sort teacher by what I feel is most important</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>As user I want to be able to filter my search so that I easier find what I am looking for</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>As buyer I want to see results to similar searches because I might now have just the correct keywords</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>As user I want to know what is required of me</td>
<td></td>
</tr>
<tr>
<td>65.1</td>
<td>In order to understand the service, I want CodyBuddy to present their offer</td>
<td></td>
</tr>
<tr>
<td>65.2</td>
<td>In order to feel that the website is trustworthy I want to see those who works on it</td>
<td></td>
</tr>
<tr>
<td>65.3</td>
<td>As a user I would like the navigation of the site to be easy without have the navigation bar too big</td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>As potential teacher I want to know how the service works</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>As user I want to not have to figure out the website myself</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>As user I want help to be available so that I do not have to figure out issues myself</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>As user I want a more specific grading system so that I can more understand the user I am looking at</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>As user I want user profile pages designed differently because different things are important to show for teachers/buyers</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>As user I want help with installing Git because I know it is good to have for programming</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>As user I want a way to contact customer support on the website</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>As user I want to know what software is good to use</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>In order to know that all users are student I want as a user that users have to register using student mail or Id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td>In order to find help that is not listed I want as a buyer be able to post entries requesting for that help</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>As user I want administrators to review reports and users with bad reviews so that people may be removed for bad conduct</td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>In order to motivate me to keep using the service I want as a user rewards</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>As user I want help with all kinds of IT-related issues</td>
<td></td>
</tr>
<tr>
<td>79</td>
<td>As upper secondary school student I might also need help with programming</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>In order to feel that the service is serious I want a user for it to not encourage bad behaviour</td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>As user I want to know what more I could receive help with</td>
<td></td>
</tr>
<tr>
<td>82</td>
<td>As a teacher I want to set my price to what I feel is reasonable</td>
<td></td>
</tr>
</tbody>
</table>
Appendix F – User Test

Task log sprint 2:

- Scroll through the start page to get a first impression of the web application and the service offered.
- Register a user.
- Login.
- Update your biography and change password.
- Go back to the start page.
- Logout
- Login with your new password.
- Book a session with a teacher of your choice, who have experience in Python, the 23rd of February 2019 between 13:15 and 15:00.
- Send a message to the booked teacher in the chat.
- Give a rating on the booked teacher with a review and rating of your choice.
- Pay your booked session (You do not need to complete the payment process).
- Cancel your booked session.
- Report the teacher you booked with a motivation of your choice.
- Create a teacher profile.
- Change any information in your teacher profile.
- Send a message to the administrators/company.
- Logout.

Task log sprint 3:

- Scroll through the start page to get a first impression of the web application and the service offered.
- Find and navigate to the terms of service page.
- Register a user and login.
- Update your biography and change password.
- Go back to the start page.
- Logout
- Login with your new password.
- Book a session with a teacher of your choice, who have experience in Python, the 23rd of February 2019, at any time.
- Send a message to the booked teacher in the chat.
- Give a rating on the booked teacher with a review and rating of your choice.
- Pay your booked session (You do not need to complete the payment process).
- Cancel your booked session.
- Report the teacher you booked with a motivation of your choice.
- Create a teacher profile.
- Change any information in your teacher profile.
- Find the contact-us page and send a message to the administrators/company.
- Logout.
Qualitative probing:

- What problems did you encounter?
- What do you think about the design quality of the web application?
- How do you find the navigation of the web application?
- How would you rate the trustworthiness of the web application?
- Which aspects of the web application makes you believe so?
- What do you think about the platform in its entirety?
- Is there anything you lack on the web application?
- What could be improved on the web application?

Quantitative probing:

- Everything adds up well on the web application.
- The layout is perceived as pleasantly varied.
- The layout is perceived as professionally designed.
- The colour-composition is perceived as attractive.
- I would feel comfortable entering credit card information on the web application.
Appendix G
- CodeBuddy Marketing Plan

Filip Renholm
Fredrik Möller
Isak Jansson
Klara Gustafsson
Leo Johansson Lara
Linus Bodeström Eriksson
Ludwig Thaung
Vendela Egondotter

Tutor, Dennis Persson
Examinator, Aseel Berglund
Table of Contents

1 Introduction .................................................................................................................. 91
  1.1 NABC ...................................................................................................................... 91
    1.1.1 Need .................................................................................................................. 91
    1.1.2 Approach .......................................................................................................... 91
    1.1.3 Benefits ............................................................................................................ 92
    1.1.4 Competitors .................................................................................................... 92

2 Marketing Environment ............................................................................................... 94
  2.1 Macro Environment ............................................................................................... 94
    2.1.1 Political Factors .............................................................................................. 94
    2.1.2 Economic factors ............................................................................................. 94
    2.1.3 Sociocultural Factors ..................................................................................... 95
    2.1.4 Technological Factors ..................................................................................... 95
  2.2 Micro Environment ............................................................................................... 95
    2.2.1 Threat of New Entries ..................................................................................... 96
    2.2.2 Threat of New Substitutes ............................................................................... 96
    2.2.3 Bargaining Power of Customers ....................................................................... 96
    2.2.4 Bargaining Power of Suppliers ......................................................................... 96
    2.2.5 Industry Rivalry .............................................................................................. 96

3 SWOT Analysis ............................................................................................................ 97
  3.1 Strengths ................................................................................................................ 97
  3.2 Weaknesses ........................................................................................................... 98
  3.3 Opportunities ......................................................................................................... 98
  3.4 Threats .................................................................................................................. 98

4 Marketing Strategy .................................................................................................... 99
  4.1 Porter’s Generic Strategies .................................................................................... 99
  4.2 Ansoff Strategies .................................................................................................. 100
  4.3 Marketing Goals ................................................................................................... 101

5 STP ................................................................................................................................. 102
  5.1 Segmentation ......................................................................................................... 102
  5.2 Targeting ............................................................................................................... 102
  5.3 Positioning ............................................................................................................ 103

6 Marketing Mix ............................................................................................................. 104
  6.1 Product .................................................................................................................. 104
  6.2 Price ....................................................................................................................... 104
  6.3 Placement .............................................................................................................. 105
  6.4 Promotion .............................................................................................................. 105

7 References .................................................................................................................. 106
1 Introduction

The purpose of this project is to create a marketplace using a web application. The application is aimed towards students that need assistance or can give assistance within programming.

1.1 NABC

An executive summary of this marketing plan has been summarised in the categories Need, Approach, Benefits and Competition.

1.1.1 Need

The web application CodeBuddy is a platform developed to facilitate communication and support for programming students.

In a time where digitalization and IT-services continues to expand it gets even more important for students to develop their understanding in programming. Students learning programming during their time at university often spend a lot of time working on different kind of projects and lab series. Not rarely are these tasks heavy relative to the scheduled time with a tutor. Nor is it unusual for students to have to spend much time waiting for assistance since each tutor has numerous students to help. For many students this might create a need for external assistance which not can be provided by the university’s resources.

Naturally students can seek help through different kinds of web sites in order to answer their questions and solve their problems regarding coding. However, there are currently no service providing private tutoring exclusively for students in this field. Among students, although, there are many proficient programmers within different programming areas who might not always have the possibility to reach out to the students requiring extra guidance. This creates a need for a web application that join student to share knowledge and to get help in programming and IT-development which CodeBuddy can provide.

1.1.2 Approach

By easily being able to create your own buyer or seller profile you can, using CodeBuddy, show what kind of help that is provided and requested in a clear way. As a customer you can sort and filter among a wide selection of competent tutors to find just the help you are looking for. As seller you can show the competence you think is relevant for your customer group after a quick registration using a simple guide on the web page.

What differentiate CodeBuddy from already existing services is the unique target group. The platform is made by students - for students. Students have emphasized that support from other students is the most efficient way to quickly gain understanding to advance in their working process since the students helping out often have been in the same situation recently.
To secure the quality and economically protect our users the web application will act as financial intermediary and perform the transaction when both parts are satisfied.

Our approach when it comes to attract an initial customer base is to offer a bonus system for both buyers and sellers, that would make it economically attractive to start using the platform.

- The platform will through an intuitive design facilitate the search for competent support for programming problems
- The platform enables for students to moonlight and acquire relevant merits beside their studies in a field interesting for them.
- The platform will facilitate the communication between buyers and sellers as well as protects and quality secures.
- By dynamic and personalized mentorship effective learning is implemented.

1.1.3 Benefits

This service makes it easier for students to get help in a quick way by booking themselves on a free spot in a seller’s calendar on the website. The platform will connect students to create an expanding network outside their already existing network.

As a seller you have the opportunity to work as much as you wish by adding the time you are available in your personal CodeBuddy calendar. Not only does the seller get the opportunity to earn extra money, the seller also gets to maintain their programming skills and furthermore develops proficiency in teaching.

1.1.4 Competitors

There are numerous of competitors in this field of services. LearnJava, Stack Overflow and Khan Academy are just some of the services where the users can learn programming basics and get helped by other users. An advantage with these is that they are free to use and that they provide a large amount of information about frequently occurring programming problems. However, services like the ones mentioned above lack of a personal meet between buyer and seller where the seller provides personal tutoring as mentioned before, which is provided by CodeBuddy.

Another already existing way of getting extra support when taking programming courses is of course to ask fellow students, mail conversations with the examiner, available programming workshops and so on. A problem with this is that one might find it somewhat insufficient. As CodeBuddy is developed to easily access competent people at the university it would be the better long-term solution.
Yet another type of competitor are those selling whole courses online. Many of these sell courses as an alternative to studying at university, they focus on passing courses quickly. But they do not provide the personal contact like CodeBuddy, instead communication is provided via videos, texts and online chats. In addition, the price for taking these courses are often high. Since CodeBuddy is made by student - for students, our ambition is that CodeBuddy will be able to keep prices at a modest level.
2 Marketing Environment

Market analysis is a key part of a market plan since it enables a business to fully identify factors operating in its internal and external environment; and the success of an organisation is greatly depending on how it understands and operates in these environments.

The macro-environment contains external forces that an organisation can’t directly control and the micro-environment consists of factors that affect the company in its immediate area of operations.

2.1 Macro Environment

Analysing the macro-environment, including non-economic factors is crucial when looking to create a sustainable strategy. The traditional framework for analysing the macroenvironment is the PEST analysis. PEST is an analysis of political, economic, social and technological factors. The PEST analysis can be supplemented by doing a full PESTLE analysis, which complements earlier mentioned factors with legal and environmental analysis. Doing a full PESTLE analysis will not be necessary for CodeBuddy because these factors do not affect the business.

2.1.1 Political Factors

CodeBuddy will be active in Sweden and will be subjected to Swedish law, most importantly the tax regulations existing in Sweden.

The business will pay preliminary tax at source on the profit across the year, which amounts to ~30%. Because the application is a mediator the service will have to pay VAT of ~25% [1]. However, the inbound and outbound VAT difference is what will be paid to the Swedish Tax Agency.

Sellers on the platform will be subject to payroll tax of ~31% and varying wage tax depending on other sources of income [1].

2.1.2 Economic factors

The target group of CodeBuddy is students at Linköping University and therefore the economic situation of students needs to be examined.

Swedish Board of Student Finance grants ~3000 SEK per four weeks as well as a student loan of ~7200 SEK per four weeks, adding up to a total of 10 172 SEK [2]. Other sources of income common among students are housing allowance, part-time jobs, grants and economic support from family. Consensus is that the economic situation for students is rather good in Sweden.
2.1.3 Sociocultural Factors

The expectation of personalization is a sociocultural trend pressuring all companies today. Consumers live in a fast-paced environment where problems should be solved as quickly as possible and expectations increase year over year. 56% of all global consumers and 68% of all consumers aged 18 – 34 say they have higher expectations for customer service now than just one year ago [3].

The emergence and popularity of distance and online courses in programming create a vacuum in the need for personal programming assistance. CodeBuddy will be able to fill this void for some.

The ever-increasing popularity of online shopping create a trend where people are increasingly more willing to pay for service and goods over the internet [4].

Online education demand is on the rise: In a survey from BestColleges 98% of course administrators find that demand for online education has increased or stayed the same over the past few years [5]

2.1.4 Technological Factors

Access to internet and computers across the Swedish society is continuously increasing in Sweden. 2017 93% of the population in Sweden had access to computers, 95% have access to internet and 76% say that they use internet daily [6]. To use the newest technology is commonplace for the younger generations, especially students - who use their computers and internet for almost everything.

As computers and internet become a bigger part of our way of life, the amount of programming courses at universities naturally increases to meet this new demand. The amount of students studying IT related degrees are increasing fast [7]. However, more importantly this trend goes beyond traditional IT-degrees. As of 2017 most engineering degrees in Sweden involve one or several courses with a main or a part focus in programming. This trend also leads to student programs without a traditional focus on programming face an increasing amount of programming in their degree. To use the internet as a platform for a mediator where this increasing demand more easily can find a supply of help is therefore a good and logical choice.

2.2 Micro Environment

To analyse the micro environment porter’s five forces analysis is a method which can be applied. By using this tool, a wide range of different competitive sources can be highlighted and the company can better prepare for rivalry. The following sources are critically to properly analyse to form a well-based strategic positioning [8].
2.2.1 Threat of New Entries

If the entry barriers into the market are high, the company is better protected against the threat of new competition [8]. Our targeted market is defined by generally low entry barriers since our product is to provide the location and opportunity for buyers and sellers to meet. What’s important for our web application is to differentiate it with a simple and easy navigable user interface. Furthermore, it’s desirable to generate a large and loyal customer base to make the choice of switching platform harder for the users.

2.2.2 Threat of New Substitutes

CodeBuddy’s product would not substitute any existing service to the targeted segment. It would rather function as a complement to the tutoring offered by the university. A threat which could come to affect the demand of the services provided by the platform, is if the university decides to increase the resources on tutoring. There’s a number of web-sites offering programming tutorials and guides that might want to extend their service and also offer a place where private teachers can advertise themselves.

2.2.3 Bargaining Power of Customers

To make the service desirable, the percentile cut CodyBuddy will take from every transaction must be well measured. If the cut is too large the customers will be tempted to only use the platform as a place to make the initial contact and continue their business deal under the table. What’s crucial about our service is to provide an easy with an emphasis on enhancing economic and personal safety to our users.

2.2.4 Bargaining Power of Suppliers

CodeBuddy won’t have any suppliers since its function is to provide a meeting spot for buyers and sellers.

2.2.5 Industry Rivalry

In the existing market there’s a number of alternatives which can influence the effectiveness of our platform. Stack Overflow and similar forums is a place where quick and detailed information can be obtained by people who experienced the same difficulties. CodeAcademy and Udemy are examples of online platforms where on-demand tutorial videos and limited help can be obtained. The mentioned rivals work well for aspiring programmers who already got some experience in the area or want a basic introduction. Our platform is differentiating itself by offering a personal learning experience and a more effective method to gain a deeper knowledge via the continuous dialog with an experienced teacher. Web sites such as CodeMentor is a more in line with what our product is offering. However, CodeMentor is focusing exclusively on online tutoring while our product gives the users complete freedom on how the service can be performed.
3 SWOT Analysis

A SWOT analysis is conducted to summarize identified strengths, weaknesses, opportunities and threats.

<table>
<thead>
<tr>
<th>Strengths</th>
<th>Weaknesses</th>
</tr>
</thead>
<tbody>
<tr>
<td>- No direct competitors in the initial niche segment</td>
<td>- Easy to replicate the platform - increased risk of direct competitors</td>
</tr>
<tr>
<td>- Assurance through proximity between buyers and sellers</td>
<td>- Possible for users to circumvent platform</td>
</tr>
<tr>
<td>- Flexible to market trends</td>
<td>- Fragile to substitutes</td>
</tr>
<tr>
<td>- Real-time information update</td>
<td></td>
</tr>
<tr>
<td>- Cuts down on administration and planning for sellers</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Opportunities</th>
<th>Threats</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Programming assistance demand increasing</td>
<td>- Too few users</td>
</tr>
<tr>
<td>- Increasing usage of e-services</td>
<td>- Big gap in demand for buyers and sellers</td>
</tr>
<tr>
<td>- Existing systems not fulfilling all market needs</td>
<td>- Possible unwillingness to pay for premium service</td>
</tr>
</tbody>
</table>

3.1 Strengths

There are no direct competitors in the initial niche segment. CodeBuddy is offering a niche competitors are not - proximity between buyers and sellers.

The nature of the platform makes it so that it will naturally adapt to supply and demand for different types of services. This makes the platform naturally flexible to market trends.

The real-time information update is a differentiating factor between CodeBuddy and traditional real-life competition. For example, seeing a live schedule will enable a streamlined booking process.

One big advantage is that people who want to help with IT want to focus on exactly that - and not on the administration and planning that a traditional approach requires. Cutting down on administration and planning for sellers helps sellers achieve exactly that.
3.2 Weaknesses

The platform will be established in a market with low barriers for entry. It will be possible to replicate the platform rather easily - which increases the risk of direct competitors in the future. In order to be resilient to this weakness it is important to get a good amount of users - and that the service doesn’t present any problems and pain points. It is crucial not to give room to direct competitors.

It is possible for users to circumvent the platform by only using it for initial contact, and then planning their coaching through other media - and using, for example Swish, for private payments. This makes it crucial to have the platform not only be a mediator of information - but to create actual value by effectively reducing time for administration and planning.

The initial market survey conducted says that the CodeBuddy platform will be sustainable because of current demand of complementary tutoring. A threat which could come to affect the demand of the services provided by the platform, is if the university decides to increase the resources on tutoring.

3.3 Opportunities

Demand for IT-related degrees and courses is increasing and the amount of programming assistance required is increasing thereafter.

Capitalize on the ever-increasing usage of e-services.

Existing systems and platforms are either expensive or time-consuming. Therefore students often do not have an obvious choice for service.

3.4 Threats

One of requirements for a platform such as CodeBuddy to provide value for its users is by having an actual user base big enough to make looking for buyers and sellers convenient. It will require an initial marketing effort with possible sign-up deals in order to make the user base grow to the necessary size fast. If this effort fails, there is a threat that the platform won’t be able to provide the advantages needed to customers to choose this over a competing type of service.

If there is a big gap in demand between buyers and sellers it might create an unbalanced and unsustainable market. It is possible that this will be solved naturally by supply and demand, but the risk remains.

Our initial market survey shows possible willingness to pay for the service CodeBuddy will provide. However, when it comes to the moment of actually paying there is the risk of buyers being more willing to choose a free alternative whenever possible.
4 Marketing Strategy

In order to achieve the goals marketing goals and strategies Porters Generic Strategies, Ansooffs Strategies of Diversification and a STP analysis is used.

4.1 Porter’s Generic Strategies

Porter's generic strategies describe how a company pursues competitive advantage across its chosen market scope. There are four generic strategies, “Cost Leadership”, "Differentiation" (creating uniquely desirable products and services) and "Focus" (offering a specialized service in a niche market). The Focus strategy can then be divided into two parts: "Cost Focus" and "Differentiation Focus.” Porter's generic strategies are ways of gaining competitive advantage – meaning how the marketing strategy could look.

The greatest risk in pursuing a Cost Leadership strategy is that sources of cost reduction are not unique to one company, and that competitors copy cost reduction strategies. This makes it crucial to continuously find new ways of reducing costs.

Differentiation involves making your products or services different from and more attractive than those of your competitors. This usually requires constant adapting to the market.

Focus strategies concentrate use a narrow scope and focus on niche markets. By understanding the dynamics of that market and the unique needs of customers within it focus strategies develop uniquely low-cost or well-specified products for the market. Focus strategies involve achieving Cost Leadership or Differentiation within niche markets in ways that are not available to more broadly-focused players.

There are many types of competitors, but it is still clear that CodeBuddy will use a narrow focus on a niche market. It is impossible to use a cost focus - because the platform competes with free alternatives. Because of this, the only way for CodeBuddy to compete is to differentiate itself from its competitors - and therefore use a differentiation focus strategy.

![Porter's Generic Strategies Diagram](image-url)
Figure 1: Porters generic strategies

4.2 Ansoff Strategies

The Ansoff Growth Matrix describes four strategies for growth of companies, - Product development, Diversification, Market Penetration and Market Development.

*Product development strategy* (new products and current markets) suggests a company focus on current markets, but they seek to understand their underlying needs and wants better so they can see opportunities for new products.

*Diversification strategy* (new products and new markets) suggests that the company target customer areas in a market where the company is not operating and doing so with a new product. There are three levels of diversification: Diversification into related markets, diversification into unrelated markets using existing resources and capabilities, and diversification into unrelated markets which require new resources and capabilities.

*Marketing development strategy* (current products and new markets) suggests the company focus on attracting customer areas at a market where the company is not currently operating. There are several ways to do this. Opening up previously excluded market segments through pricing policies, new marketing and distribution channels, or entering new geographic markets.

*Market penetration strategy* (current products and current markets) will either try to reach out to new customers in an existing market or increase sales to current customers with an existing product. Market penetration strategy is the preferred route to growth for many businesses because it appears safe. The emphasis is on increasing market share through more marketing promotions, more effective marketing and by strengthening the offer by creating more customer value.

CodeBuddy is a start-up with limited financial resources. Further development of the platform could prove costly and time consuming. The market already exists, and therefore only a market penetration strategy is viable.
4.3 Marketing Goals

The goal with the strategies presented is to effectively create a market niche. It is clear from Porters’s Generic Strategies that CodeBuddy will use a narrow focus on a niche market. CodeBuddy will differentiate itself from its competitors - and therefore use a differentiation focus strategy. From Ansoffs Strategies point of view CodeBuddy will try to reach out to new customers in an existing market by more effective marketing and by strengthening the offer by creating more customer value, i.e use a market penetration strategy, although it is in a niche market.

**Marketing goal 1:** Establish a sustainable customer-base around the limited geographic region Linköping within one year of launch.

**Marketing goal 2:** Establish a position as a reliable service-providing platform, measured through customer surveys.
5 STP

The STP process is a concept that demonstrates the links between an overall market and how a business chooses to compete in that market. The goal of the STP process is to guide the business to the development and implementation of the marketing mix.

5.1 Segmentation

The market can be divided into different geographic segment and since the web application can be accessed world-wide, the scope is flexible. It can geographically be divided into varying sizes of nations or cities.

The market can also be segmented into students, university teachers and the general public. Students are more inclined to learning applications and can be assumed to demand the service as a complement to their courses.

University teachers might see the web application as an opportunity to further increase their pedagogic skills and their regular salary. However, compared to the student group, their economic situation and the fact that they’re already teaching probably make this group less prone to use the service.

The public might also host an interest of teaching or learning programming. This group is harder to define and reach and require more effort on the marketing for a smaller potential payoff.

The market also needs to be divided in to a seller and buyer segment, although these two can be presumed to be heavily overlapping. The sellers are assumedly more frequently students of higher grades with more experience. They also need to have some competence in the programming area, the ability to teach and the desire to make some extra cash in their spare time. The buyers are in line with the same logic more frequently found in the lower grades with the possibility to spend money on private tutoring and experiencing a lack of tutoring support provided by the university.

5.2 Targeting

Our marketing is initially primarily focusing on students located in Linköping and studying at Linköping University. The marketing can be expanded to other areas where there’s a large potential customer base consisting of students. This segment is clearly the most suitable for our service. Teachers and the general public are potential customers but shouldn’t be classified as the main target. The choice of group can be evaluated with the aid of the EMAAP-model. The model suggests that the chosen segment should fulfil the criteria of being effective, measurable, accessible, actionable and profitable.
It’s effective since the marketing only targets a homogenous group. The group can be defined by students with a strong demand of extra tutoring and the desire to have an extra income. According to our survey, 77.3% of the participants answered that they would appreciate more assistance and 68.8% answered that they were willing to provide tutoring if they were compensated.

It’s measurable, the segments size can easily be determined from statistics provided by Linköping University.

It’s accessible since they are located in the same geographical area and connected through networks on various social media platforms which can be used as a marketing channel for the web application.

It’s actionable, the initial market survey reveals that the range of the sellers preferred compensation and the buyers preferred price are overlapping to a great extent. A clear potential for a market where buyers and sellers both profit can be assumed.

It’s profitable from the perspective of the hosts of the web application because of the option to take a percentile cut or a fixed amount from each transaction made via the web sites payment system.

## 5.3 Positioning

To establish CodeBuddy on the market it’s important to take a position with a diversified strategy. 4C is a method to determine if a strategic position is effective and is evaluated by its clarity, credibility, consistency and competitiveness.

**Clarity** - A distinct message has to be sent to the students. With an emphasis on the safety and simplicity of the service, combined with the customized user experience and the effortless way of earning money/receiving competent help.

**Credibility** – It is crucial that the web application is credible to attract a customer base willing to use the web sites payment method instead of conducting their business transaction with the aid of other services. This can be achieved by getting well known influencers such as prominent teaching assistants and teacher to use or promote the web application.

**Consistency** - The message has to be consistent to make users trust the service and the benefits it provides. This means that the message described in “clarity” needs to be assured of its accuracy in delivery.

**Competitiveness** - Currently there’s no direct competitor for the chosen niche on the market. This gives CodeBuddy the first mover advantage and the possibility to undisturbed attract users for the service. It is crucial to keep the users from switching platform because of the relative simple task of replicating a competing platform.
6 Marketing Mix

To help define the marketing options in and how CodeBuddy will work to achieve the marketing objectives a 4P framework will be used. This will define analyse Price, Product, Promotion and Place, so that CodeBuddy’s offering meets the customer needs.

6.1 Product

CodeBuddy’s core product is to offer a web-based platform where sellers with competence in programming can offer a personalized learning experience to buyers in need of it. The service is in its first iteration going to focus on students at Linköping University. On one hand, the platform is a place where students with programming skills can advertise themselves as private teachers to access a wide customer base. On the other hand, it’s also a place where students in need of extra help can find accurate help delivered by people in their proximity.

To ensure customer satisfaction the web application comes with a rating- and review system. This is to promote well-performing “sellers” and let the customers base their choice of teacher from previous buyers’ experiences. The buyers will also be able to specify what type of help is wanted via a search system to filter and limit their alternatives to smaller range of options with the customized parameters. To further increase the trust between buyers and sellers the web application will be acting as a transaction intermediator.

What’s important to emphasize on this application is giving the user’s the freedom to customize the service to maximize their experience and perceived value.

6.2 Price

The price of the service is set by the seller. This is to let the seller decide what their service is worth. A seller with less experience, competence or need for compensation can thereby attract customers with a lower price. To give the sellers the option to dynamically price their service is crucial for meeting the market’s shifting demand. By implementing this functionality, the web application will probably offer quality education to the buyers and fair compensation to the sellers.

Costs related to the development of the web application would typically be server and connectivity cost as well as salary for the people assigned on the project. However, in this case Linköping University is hosting the server for free and the group working on the project are working for free since it is student group working on a university project.
6.3 Placement

CodeBuddy is a web application where all the necessary information about the sellers will be displayed on a profile page. The buyer and seller can make initial contact with a chat system implemented within the web application. When both parts have come to an agreement they can choose to continue their communication with the assistance of other communication channels. If the parts want to physically meet up or digitally carry out the service is for the parts to decide. The web application will be accessible from all regions, but the marketing will initially be directed at Linköping and more specifically students at Linköping University.

6.4 Promotion

To get CodeBuddy into the chosen market, it will take a distinct and efficient marketing to the targeted segment. By making use of advertisement on social media platforms like Facebook, Instagram, Twitter and Yodel some market penetration can be achieved. A starting offer for the users where CodeBuddy is giving the customer a cost-free lesson with the “teacher” of their choice will also be presented to the customer to gain an initial customer base. The sellers will also help advertise the website since it is in their own interest to further increase the number of users, so the word-of-mouth is probably going to be an effective marketing tool. Another way of creating an awareness of the web application is by getting the teachers at the university to promote the web site as an extra resource to enhance and extend their learning experience.
7 References


