

# CHALMERS



## WishLister

-Conceiving, Designing and Developing an iPhone Application

*Master of Science Thesis in the Programme Interaction Design*

CHRISTOFER LEYON  
NATHALIE NILSSON

Department of Computer Science and Engineering  
CHALMERS UNIVERSITY OF TECHNOLOGY  
Göteborg, Sweden, December 2010

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

WishLister

Conceiving, Designing and Developing an iPhone Application

CHRISTOFER LEYON

NATHALIE NILSSON

© CHRISTOFER LEYON, December 2010.

© NATHALIE NILSSON, December 2010.

Examiner: MORTEN FJELD

Department of Computer Science and Engineering

Chalmers University of Technology

SE-412 96 Göteborg Sweden

Telephone + 46 (0)31-772 1000

Cover:

WishLister application icon

Department of Computer Science and Engineering

Göteborg, Sweden, December 2010

## **Acknowledgements**

The authors would like to thank the following people for encouragement, inspiration and for taking time out to help us with this project. Johan Berndtsson and Peter Erhard at inUse for invaluable tips and insight about everything from interaction design to marketing, throughout the project. Johan Carlberg for development support and help, which allowed the authors to actually finish this master thesis. We can't thank you enough. Morten Fjeld at Chalmers University of Technology for insight and supervision throughout the course of the project and it's many ups and downs. Last but not least all the people that participated in the numerous user tests conducted throughout the project.

## **Abstract**

The iOS platform, today, has a market share of 28% for smartphones. Apple's App Store has, since it launched, had over seven billion downloads. The touch screen paradigm introduced by Apple with the initial launch of the iPhone in 2007, to many revived the market of touch screen interfaces. Needless to say, the iOS platform provides exciting development possibilities in the area of interaction design.

This master thesis in the Interaction Design Programme at Chalmers University of Technology aims to combine methods and knowledge gathered from Chalmers with the work flow and process used at inUse AB, primarily the use of effect managing, which is a method developed by inUse for ensuring a project achieves its desired effect. The goal of the thesis was that of producing a commercial app for Apple's iOS platform and to release it on Apple's App Store.

Through the use of iterative development methods, an app was conceived, designed and developed throughout a seven month period. User tests were conducted throughout this process, to ensure a high usability standard for the developed app. The app, WishLister, aims to simplify the process of creating and sharing wish lists with friends and family, as well as easing the purchasing of gifts for friends and family.

Some of the functionality of WishLister still remains to be implemented and this will be done outside the scope of this master thesis. The release of the app is set at sometime during late winter, early spring of 2011.

inUse AB is the largest consulting company in the field of usability and strategic design in Scandinavia. This master thesis was conducted at inUse's Gothenburg office.

# Table of Contents

<b>1 Introduction</b>	<b>1</b>
<b>1.1 Background</b>	1
<b>1.2 Aim</b>	1
<b>1.3 Delimitations</b>	2
<b>2 Methodology and Theory</b>	<b>3</b>
<b>2.1 Literature study</b>	3
<b>2.2 Agile Methodology</b>	3
2.2.1 Scrum	4
2.2.2 Pair Programming	5
<b>2.3 Project Risk Analysis</b>	5
<b>2.4 Concept Generation</b>	6
2.4.1 Brainstorming	6
2.4.2 Fly on the Wall	6
2.4.3 Random Words	6
2.4.4 Randomised Webpages	7
2.4.5 Focus Groups	7
2.4.6 Media Scan	7
<b>2.5 Concept Evaluation</b>	8
2.5.1 Functional Analysis	8
2.5.2 Expert Evaluation	8
<b>2.6 User Interaction Scenario</b>	8
<b>2.7 Competitor Analysis</b>	8
<b>2.8 User Analysis</b>	9
2.8.1 Defining User Group	9
2.8.2 Interviews	9
<b>2.9 Effect Mapping</b>	10
<b>2.10 User Requirement Specification</b>	10
<b>2.11 Interaction Design</b>	11
2.11.1 Fundamental Design	11
2.11.2 Functional Design	11
2.11.3 Detail Design	11
<b>2.12 Design Evaluation</b>	12
2.12.1 Expert evaluation	12
2.12.2 User Test	12
2.12.2.1 Paper Prototypes	12
2.12.2.2 Static images on device	13
2.12.2.3 Developing interactive scenarios	13
2.12.2.4 Pilot Session	13
<b>3 Time Schedule</b>	<b>14</b>

<b>4 Concept Phase</b>	<b>15</b>
<b>4.1 Concept Generation</b>	15
<b>4.2 Concept Evaluation</b>	15
4.2.1 Functional Analysis	15
4.2.2 User Survey & Expert Evaluation	15
<b>4.3 Pitching to Businesses</b>	16
<b>5 WishLister</b>	<b>17</b>
<b>5.1 WishLister Concept</b>	17
<b>5.2 Competitor Analysis</b>	18
5.2.1 Web-based services	18
5.2.2 App Store	19
<b>5.3 User Analysis</b>	22
<b>5.4 Effect Map</b>	24
5.4.1 Aim	24
5.4.2 Target User Groups	24
5.4.3 Usage Goals	27
5.4.4 User Requirement Specification	28
<b>6 WishLister Development</b>	<b>32</b>
<b>6.1 Paper prototypes</b>	33
6.1.1 Sprint 1	33
6.1.2 Sprint 2	33
<b>6.2 Static images on device</b>	35
6.2.1 Sprint 3	36
6.2.2 Sprint 4	37
<b>6.3 Interactive scenarios</b>	38
6.3.1 Sprint 5	38
6.3.2 Sprint 6	40
6.3.3 Sprint 7	41
6.3.4 Sprint 8	42
6.3.5 Sprint 9	44
<b>6.4 Further Development</b>	46
<b>7 Result</b>	<b>47</b>
<b>7.1 The colours of WishLister</b>	47
<b>7.2 The technical solutions of WishLister</b>	47
<b>7.3 Launching WishLister</b>	48
<b>7.4 The five tabs of WishLister</b>	49
<b>7.5 Creating a wish</b>	49
7.5.1 Textfields	50
7.5.2 Adding a photo	50
7.5.3 Adding to wish list	51

<b>7.6 Wishes</b>	<b>52</b>
7.6.1 Moving and tapping wish pads	52
7.6.2 Deleting a wish pad	53
7.6.3 Moving a wish into a wish list	54
<b>7.7 The Wish Lists</b>	<b>54</b>
7.7.1 Navigating wish lists	55
7.7.2 Viewing a wish list	55
7.7.3 Deleting a wish list	56
7.7.4 Organising a wish list	57
<b>7.8 Sharing a Wish List</b>	<b>57</b>
7.8.1 Share view	58
7.8.2 Sharing a wish list	58
7.8.3 Inviting friends	59
<b>7.9 Working with a friend's Wish List</b>	<b>59</b>
7.9.1 Viewing friend's wish lists	60
7.9.2 Reserving a wish	60
<b>7.10 Using the Cart</b>	<b>61</b>
<b>8 Discussion</b>	<b>63</b>
<b>8.1 Status of WishLister</b>	<b>63</b>
<b>8.2 WishLister's Aim and Measurement points</b>	<b>63</b>
<b>8.3 Usability and Interaction Design</b>	<b>63</b>
<b>8.4 Future work</b>	<b>64</b>
<b>8.5 Work Process</b>	<b>64</b>
<b>9 Conclusions</b>	<b>66</b>
<b>References</b>	<b>67</b>
<b>Appendices</b>	<b>70</b>

# 1 Introduction

This is a master thesis at the Interaction Design Master Programme at Chalmers University of Technology, done by Christofer Leyon and Nathalie Nilsson. Mr. Leyon has a bachelor degree in Product Design Engineering and Ms. Nilsson has a bachelor degree in Software Engineering, both from Chalmers. The thesis will be in collaboration with inUse AB and focused on the iOS platform.

inUse is the biggest consulting company, in the field of usability and strategic design, in Scandinavia. It was founded in 2002 and today have over 50 consultants with offices in Gothenburg, Malmö and Stockholm. Their main goal is to “[...] change the way the world views IT”. (inUse, 2010)

The iOS platform is a relatively new and definitely exciting platform for developing content. It is used by most of Apple’s mobile devices (iPhone, iPod touch and the iPad) as well as Apple TV and has a market share of 28% for smartphones (Kellogg, 2010). The touchscreen functionality that these mobile devices implement has become more or less mainstream in mobile devices today, due in large to the success of the iPhone. It is an interesting albeit quite old technology (the first patent for a touch sensor was filed in 1971) that opens up new interaction design possibilities. It is these possibilities that this master thesis aims to explore.

## 1.1 Background

The first iPhone was released in June 2007 and in terms of innovation and originality it was a huge leap forward for what is today known as smartphones. According to Apple it combined three products into one, namely a mobile phone, an iPod and an internet capable hand-held device (Apple, 2007). Three years later and with the release of the iPhone 4 the market is brimming with devices, where many of them, in some way, mimic the interaction paradigm set by the initial release of the iPhone three years ago.

Despite the initial success of the iPhone, the device did not come into it’s own until the release of the App Store, which allowed iPhone users to download applications directly to it. Today there are over 300 000 apps in the App Store and there have been over seven billion apps downloaded since the App Store opened and with a small sum payed for a developer license anyone can create and sell apps on the App Store (Apple, 2010a).

## 1.2 Aim

The aim of this master thesis is to combine methods and knowledge gathered from studies at Chalmers University of Technology with the work flow and processes used at inUse. Through the use of this knowledge an app for the iOS platform, that incorporates inUse’s strong focus on usability and interaction design, will be developed. An app is defined as an application available at Apple’s App Store which is designed to work on devices compatible with the iOS platform. This to see what commercial value there is to gain, for inUse, from establishing a presence on the iOS market. However, the resulting app is not meant to produce any substantial monetary value for inUse or the authors but rather work more as a tool for exposure, leading hopefully to opportunities for other projects in the future. As such, usability, interaction design and ease of use are of paramount importance when designing the app, whereas number of features are not. A goal is to release the app on Apple’s App Store, following the completion of the thesis and approval from inUse.

### **1.3 Delimitations**

Since the aim of the project is to release an app on the App Store, any necessary subsequent updates or changes to the app following this release is outside the scope of the master thesis and will not be included in this report.

Apple provides a document outlining the human-interface guidelines (Apple, 2010b) that should be followed when developing an app for the iOS platform. This document can be downloaded from Apple's website and as recommended reading for any considering developing an app for release on the App Store. There is also a limitation on the functionality of the app since Apple most likely will not approve an app that competes directly with one of Apple's own apps. Also, an app may be rejected if there is already too many similar apps readily available for download on the App Store (Apple, 2010c).

The limited time of the project coupled with the limited knowledge and experience of the authors, when it comes to developing for the iOS platform, needs to be considered when deciding what app to develop.

Any type of marketing done by the authors or inUse in conjunction with the release of the app will not be presented in this report as it would take away from the focus of the report, which is on interaction design and usability.

## **2 Methodology and Theory**

inUse has a distinct way of working in their projects, much due to the method known as Effect Mapping, which is a method developed by inUse themselves. As such, the methodology used in this project was based on some key elements of how the inUse project process looks. That is not to say that the process was copied right off as some modifications were made and were necessary to make the process fit this master thesis. It is important to note that as the project started out the authors had no clear idea as to what iOS app was going to be developed. This was deemed an important part of the project, with concept development and evaluation playing a big role in the project, alongside design and development.

### ***2.1 Literature study***

A literature study was an integral part of the project as it gave the authors the opportunity to learn and digest the methods and theory that is the basis for how inUse tackle any new project. During the start of the project seven books were recommended by inUse for use as a way to gain insight, not only into the way inUse work, but also to gain more in depth knowledge of usability and interaction design trials and tribulations with respect to the iOS platform. The eighth book (Jävla skitsystem!) was read at the end of the project and was used as a supportive document to the decisions made throughout the design process. The books are listed below in alphabetical order and more information about them can be found in the reference list.

- Användbarhet i praktiken
- Designing the iPhone User Experience
- Effect Managing IT
- Jävla Skitsystem!
- Pragmatic thinking and learning
- Tapworthy
- The iPhone Developer's Cookbook
- The Smashing Book.

### ***2.2 Agile Methodology***

Agile methodologies are typically used mainly within software development serves as a foundation for managers and their teams. Agile methodologies are based upon iterations in contrast to sequential methodologies such as the waterfall model. Winston Royce, who was the first to outline sequential methodologies, in 1979, described it as a process where every phase must be completed before the next one. Sequential methodologies make it hard to go back change project requirements, should that be needed. As such, agile methodologies were developed to alleviate this problem. When working with agile methodology, focus is on the repetition of abbreviated work cycles, which enable continuous updates of project requirements and design throughout the entire span of the project. The big advantages with agile methodology are the reduced development costs and a shorter time to market. (Agile Methodology, 2008) According to Eriksson (2007) it is easier to find more requirements through using agile methodologies, due to the fact that it is almost impossible to find all requirements in the beginning of a project, and if possible would take extremely long to do.

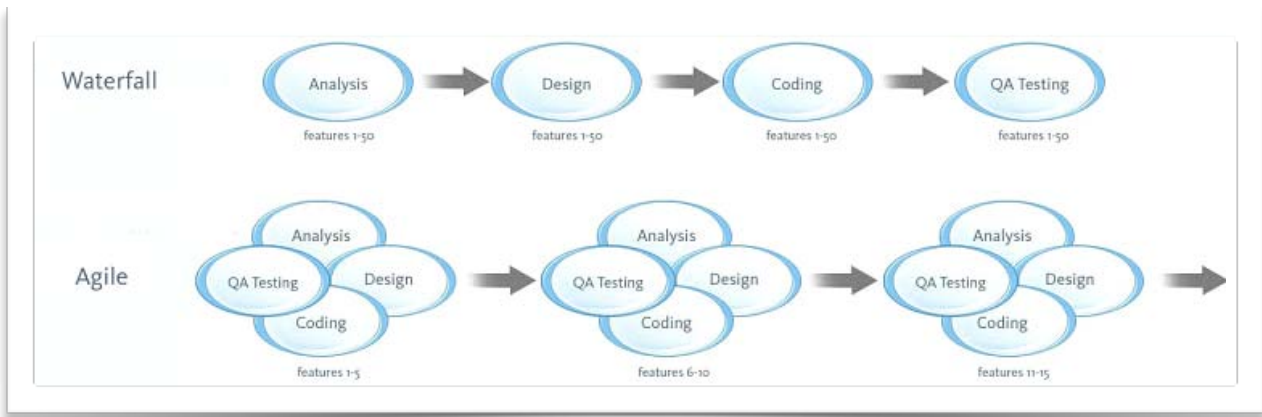


Figure 1 - Waterfall model contra agile methodology (Sy, 2007).

### 2.2.1 Scrum

Scrum is one of many agile methodologies, with the word Scrum being a term used in rugby. In this project inspiration from Scrum has been used in order to create a modified development process, see Figure 2. The first phase, called Concept phase, consists of idea generation, evaluation and user analysis.

Scrum is an iterative process where the project team works towards a set of fixed goals during a one to four week period, which in Scrum is called a Sprint. This iterative workflow will take part after a concept has been defined along with project requirements. The project requirements will be prioritised as well as be subject to change throughout the project. Before each Sprint the requirements with the highest priority will be moved to a Sprint list. The requirements in the Sprint list will be designed, developed and evaluated during a three week period. During the Sprint it is not possible to change, add or delete any requirements. This is done after each Sprint before starting the next one.

The team will also meet up every morning for a daily meeting, where each member will be answer the following questions:

- What did you do yesterday?
- What will you do today?
- Are there any impediments in your way?

(Sutherland and Schwaber, 2007)

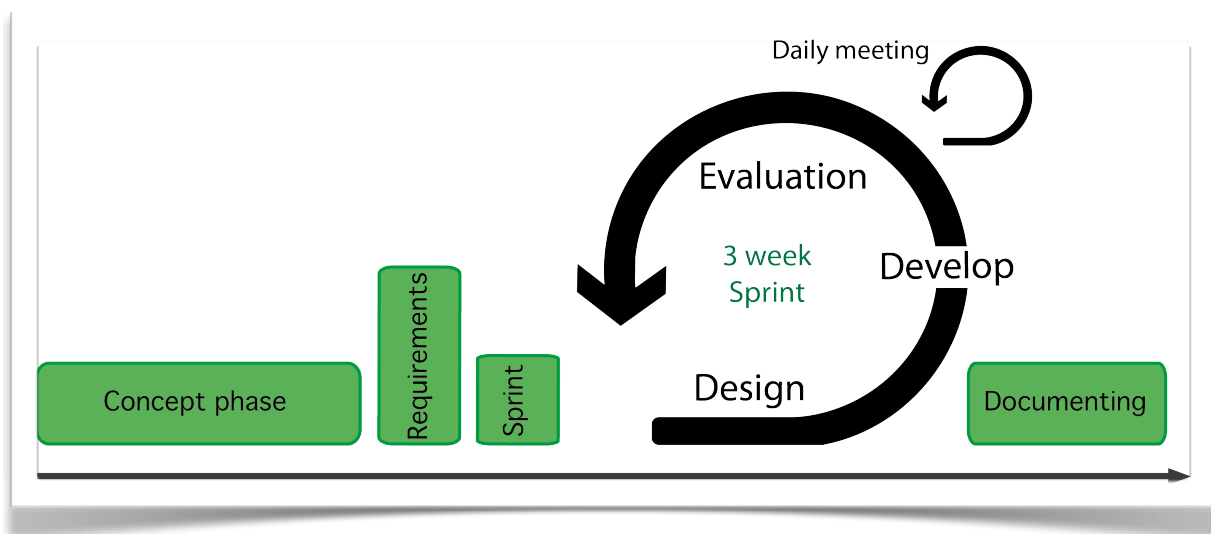


Figure 2 - The projects development process.

Before a new Sprint can begin, the entire team including the client will meet up for a reflection meeting where the current state of the project is discussed. Finally, the last phase (documenting) will be devoted to report writing and an oral presentation.

The process described here differs from the one that is presented in the project's Planning Report and Project Plan (see Appendix A and B), mainly because these documents were created before the project started and in the spirit of agile methodology the process and plan were revised.

### **2.2.2 Pair Programming**

The old idiom "Two heads are better than one" sums up the concept of pair programming well. The idea that two people working together on a problem will solve it faster than if they were working separately on one problem each is a fact that few people would refute. Pair programming in a nutshell is the sharing of a workstation by two programmers. The two programmers will work together at one workstation as opposed to each of them sitting and working by themselves. The purpose of pair programming is to increase communication, with the end result being higher quality code than if the programmers were to sit on their own and write. Like many collaborative methods pair programming is not for everyone and can be quite challenging, especially for people who are used to working on their own. However, the benefits clearly outweigh the difficulties. (Beck, 2000)

## **2.3 Project Risk Analysis**

A project risk analysis document is intended to be used to support the development process of the project. The method used aims to create a holistic view of potential risks that may occur during the project, along with measures on how to avoid them or limit their impact. For this project, the Lichtenberg method, a risk analysis method developed by Steen Lichtenberg, was used to identify possible risks. By identifying each risk and its impact, the damage done, should the risk occur, will hopefully be less severe than if the risk was an unknown.

The first step in the Lichtenberg method is to identify possible risks and evaluate each of them, with the probability that a specific event will occur and the consequence if it should occur, being assessed. The probability of occurrence and the impact of the risk occurring (consequence) will be valued separately on a scale of one (low) to five (high), with one meaning a low probability of occurrence, and a small impact on the project respectively. Moreover, to obtain a risk value the probability and the consequence grade, per risk, will be multiplied giving a value between 1-25.

Risks with a value higher than 12 will be given more attention than risks with a value lower than 12. One should keep in mind that identifying a risk doesn't necessarily mean it can be avoided, even if that might be the case sometimes as measures to reduce possible risks are also discussed and identified in this analysis. Finally, a description of counter-measures in the event that a risk does occur are also compiled. This data will be logged in a Risk log (see Table 1 below), which will be updated throughout the project. (Lichtenberg, 2005) The updated risk log can be found in Appendix C.

<i>Risk Description</i>	<i>Probability</i>	<i>Impact</i>	<i>Risk</i>	<i>Measures to eliminate/reduce risk</i>	<i>Measure if a risk occur</i>

**Table 1** - Risk Log (Risk coefficient: Low 1-9, Medium 9-16, High 16-25).

## **2.4 Concept Generation**

In order to facilitate the generation of a plethora of ideas from which a final one could be chosen, a number of idea generation techniques were used in this project. Some of them are methods that have been around for a long time and a few make use of new technology as a way to gain inspiration from which an idea of epic brilliance might spring up.

### **2.4.1 Brainstorming**

The object of brainstorming is the generation of solutions to a given problem. Brainstorming without a clear problem or issue is possible but would probably be less effective than had a clear goal been set. The core of brainstorming is the utilisation of a group's combined imagination and experience with the goal of finding the best solution to a given problem. The idea is that bouncing ideas off other people will yield more and better solutions than what can be achieved by a single individual. The method is not without its pitfalls however. One very important aspect of a successful brainstorming session is the mantra that no critique is allowed. In essence this is to allow the free flow of thoughts to fill up the room and not let critiques and discussions about specific details bog down the session. There is also a social aspect as people might feel attacked or become apathetic or even somewhat depressed if their ideas get shot down repeatedly. (Landqvist, 2001)

A typical brainstorming session employs no more than seven people, as this is a good number for allowing everyone to get a word in edgewise. The session can last for as long as one deems necessary but after 45 minutes energy is typically quite low amongst the participants and if the session is to continue a long break is usually required. The result of a brainstorming session is more or less a list of ideas that solve whatever issue or problem was the focus of the brainstorming session. (Landqvist, 2001)

### **2.4.2 Fly on the Wall**

The fly on the wall method, at its core, is about observing without interfering. The object of the method is to watch potential users' behaviour, without the users' knowledge, within a certain context as a way to gain information that might not be possible to extract through means where the user knows he or she is being observed. This is especially true for certain types of subconscious behaviour or behaviour which might carry any form of social stigma or tabu. A typical person performs a multitude of actions each day that are more or less lost on them, should a question be asked regarding said action. As such it is sometimes necessary to observe people's behaviour rather than ask them about it. (IDEO, 2003)

### **2.4.3 Random Words**

The use of words as a stimulus for generating ideas is what the random words method is all about. The participants write down random words on small pieces of paper. The number

of words is not critical to the quality of the result. These words are then shuffled together whereupon a participant draws a word and considers the word in the context of a specific issue or problem. An idea is hopefully formed and is written down as the next participant in line draws a random word and does the same thing. The words work as catalyst for forming associations that might not have been formed otherwise and can be a useful tool in seeing a problem from a different angle or other point of view. The random words session usually lasts until there are no more words left. (Straker and Rawlinson, 2003)

#### **2.4.4 Randomised Webpages**

The site stumbleupon.com allows its users to surf the internet in a somewhat haphazard way. By randomising websites, users can stumble upon new content (hence the name) that they would not have found otherwise. This aspect of the site inspired the team to use Stumbleupon as a way to generate ideas. The thought was that during twenty minutes, a number of sites would be stumbled (much the same as googling is to Google) and ideas would spring up from the inspiration the websites provided. The session should be focused around a topic or problem so as to help the participants have something to relate the random websites to. Ideas that spring up during the session are logged on paper. Note that the link of an idea to a website is not necessary to log, since the website itself serves no other purpose than that of a muse.

#### **2.4.5 Focus Groups**

Focus groups, much like brainstorming sessions, make use of a collective. In a focus group, the participants discuss openly a predetermined topic and focus is on the interaction between the participants and not between participants and interviewer, whose role is more that of a moderator. Focus groups are useful for gathering data on people's experiences or feelings towards, for example; a commercial product, and also for exploring and generating ideas. The participants of a focus group are chosen depending on the wanted outcome and topic of the specific focus group session. For example a focus group concerning the topic of driving a car probably should not contain people without an actual driving license. A focus group should consist of six to ten participants and should last one to two hours. During a focus group a moderator will be present to ensure that the discussion stays on track and moves in the right direction. The moderator will ask open-ended questions or probe a participant to elaborate on statements or thoughts expressed. However, the moderator should try to stay as inconspicuous as possible while still maintaining a authoritative stance and not get personally involved in the discussion. (Gibbs, 1997)

#### **2.4.6 Media Scan**

A media scan is an information and inspiration gathering method developed by the project team. It serves the purpose of finding content, either online or in paper form, that can inspire new ideas. Similar to the random websites method explained above a media scan is more in depth as it requires the researcher to read articles or watch movie clips. As such, the media scan is not strictly a concept generation tool as much as it is an inspiration gathering tool, whereupon the collected inspiration will hopefully lead to new insights and subsequently new concepts and ideas. The media scan is not necessarily a method that is conducted in short sessions but more in short bursts (fifteen minutes reading the newspaper in the morning) spread across a longer time period. These time periods can last for the entire concept generation phase.

## **2.5 Concept Evaluation**

Following concept generation, resulting in many ideas, a structuring, grading and subsequent choice of concept or concepts, depending on project type, is necessary. This process can be helped by the use of evaluation methods that serve to add a certain objectivity to something that might otherwise become mostly about personal preferences. That is, the project team may be tempted to think of themselves as potential users instead of considering who the actual users are (Berndtsson and Ottersten, 2002).

### **2.5.1 Functional Analysis**

A functional analysis is used to grade concepts in respect to certain predefined criteria. The criteria are chosen to cover the aspects of the project that are deemed the most important. Needless to say choosing the right criteria is of critical value for the functional analysis to work properly and for it to give the project team useful data to use in decision making. The practical execution of a functional analysis makes use of a matrix. The matrix has concepts on one axis and criteria on the other. The project team then goes about grading each concept in respect to each criteria. For example, one important criteria for a design project was for it to allow one-handed use. One proposed concept involves using one hand to hold the product with the other hand used to manipulate it. Obviously this concept scored low on the criteria of one-handed use. However it might score higher on another criteria which is why the criteria are often weighed as well as graded to allow for a high grade in an important criteria to be worth more than a high grade in a criteria which is not as important. The grades are eventually summed up, whereupon each concept gets a numbered score. This score should represent each concept's ability to be successful and as such the best concept should have the highest score. (Landqvist, 2001)

### **2.5.2 Expert Evaluation**

By using an expert evaluation it is possible to collect feedback on a concept without requiring a major undertaking. The evaluation of an expert results in a list of possible flaws which can then be addressed before conducting large user studies. In essence the expert evaluation is a cost and time saving method, as it can allow the project to make use of knowledge within the entire company and not just knowledge within the project team. (Berndtsson and Ottersten, 2002)

## **2.6 User Interaction Scenario**

A user interaction scenario involves at least one person and at least one action and the outcome describes the behaviour and the experience of the person who acts. User interaction scenarios are used to describe a user's activities. A scenario often includes high-level and low-level goals where the former represents the main goal and the latter represent one or more sub goals. Scenarios is often used to help and analyst to focus on peoples attention on assumptions when they perform a specific task. (Rosson and Carroll, 2001) A scenario does not need to be in text form. However, it is the most common way of representing a scenario, though storyboards, video mockups, prototypes can also be used. (Rosson and Carroll, 2001; Carroll, 1995)

## **2.7 Competitor Analysis**

According to Armstrong, Kotler, Harker and Brennan (2009) it is important that the product being developed provides greater customer value and satisfaction than its competitors. To succeed on the market it is necessary to have an strategic advantage, it is not enough to only concentrate on customer's needs. A method used for this is to study competitor products. For example, in the automotive industry, car makers buy cars of other brands to take apart and study. (Armstrong, Kotler, Harker and Brennan, 2009) In this project

competitors have been defined and studied for future analysis. The goal is to collect inspiration and ideas to implement in the project, as well as to learn about the market and what works and what does not. Focus has also been on finding and understanding important factors on the App Store to identify users' needs for iOS applications. From this, the project team may be able to avoid making mistakes that competing products have already made.

## **2.8 User Analysis**

The main goal of a user analysis is to collect requirements from potential users of the product. Even requirements that might seem obvious to the project team are collected and defined. A method suggested by Berndtsson and Ottersten (2002) has been used in order to define user groups and collect requirements. First, the team and the client attempt to identify user groups based on their own subjective perspective. Two people from each of the defined user groups will then be interviewed. The interviews are then used as foundations for the creation of personas for each of the user groups.

### **2.8.1 Defining User Group**

According to Berndtsson and Ottersten (2002) a user analysis is the basis for building a great product. It is important to have knowledge about a product's potential user group for obvious reasons. A well-conducted analysis will often result in significant benefits, for example: shorter lifecycle costs.

By answering generic questions about the users, the project team and client will get a better understanding for who the potential users are. The questions answered cover the users' knowledge, experience, tasks, frequency of usage, age, gender and also the potential value of the product for the user. (Berndtsson and Ottersten, 2002)

### **2.8.2 Interviews**

When developing a product the user group's requirements plays an important role. At an early development stage, this is a problem because eliciting the right requirements requires that the developer and the user understand each other and are able to communicate efficiently. It is also important that the interview method used to collect data is chosen carefully to match the specific purpose, for example: quantitative versus qualitative data. Another important factor that will affect the outcome of an interview is the choice of environment. No matter what environment an interview takes place in, the interviewee will be affected by it. Usage of a mediating tool can also be helpful, where a mediating tool is an artefact used to facilitate communication.

The questions used during the interviews were of a semi-structured nature, with the intention that the answers move beyond a mere yes or no. When using semi-structured questions the interviewer does not need to ask the questions in the same order or formulate the questions in exactly the same way during each interview. (Engelbrektsson, 2010) By using open-ended questions, the interviewer can invite the user to a dialogue which in turn makes for a deeper discussion. Probing was also used to clarify information during the interview and to repeat without interrupting to gather more detailed answers (Schmidt and Conaway, 1998; Karlsson, Kaulio, Hampf and Sperling, 1998). Each interview should last approximately one hour with all of the interviews being voice-recorded for later analysis. Finally, according to Berndtsson and Ottersten (2002) it is enough to interview at least two persons per user group.

## 2.9 Effect Mapping

Effect mapping serves to help the team define what benefits the project will give, who will create them and what needs to be done to accomplish it, by creating an effect map. According to Ottersten and Balic (2007) "An effect map describes the desired effect and provides a hypothesis regarding how these are to be achieved". The method consists of four steps. The first step is to explain *why* the product is being developed by defining its core purpose. The target groups are defined by determining *who* will create the desired effect, that was defined in step one, during step two. Each *target group* is analysed and their wants and needs are formulated as usage goals. This is the third step. Finally, the product and operations design are described in *actions* where the target groups needs, expectations, values and driving forces are met. (Ottersten and Balic, 2007) See Figure 3 for a graphical overview of an effect map.

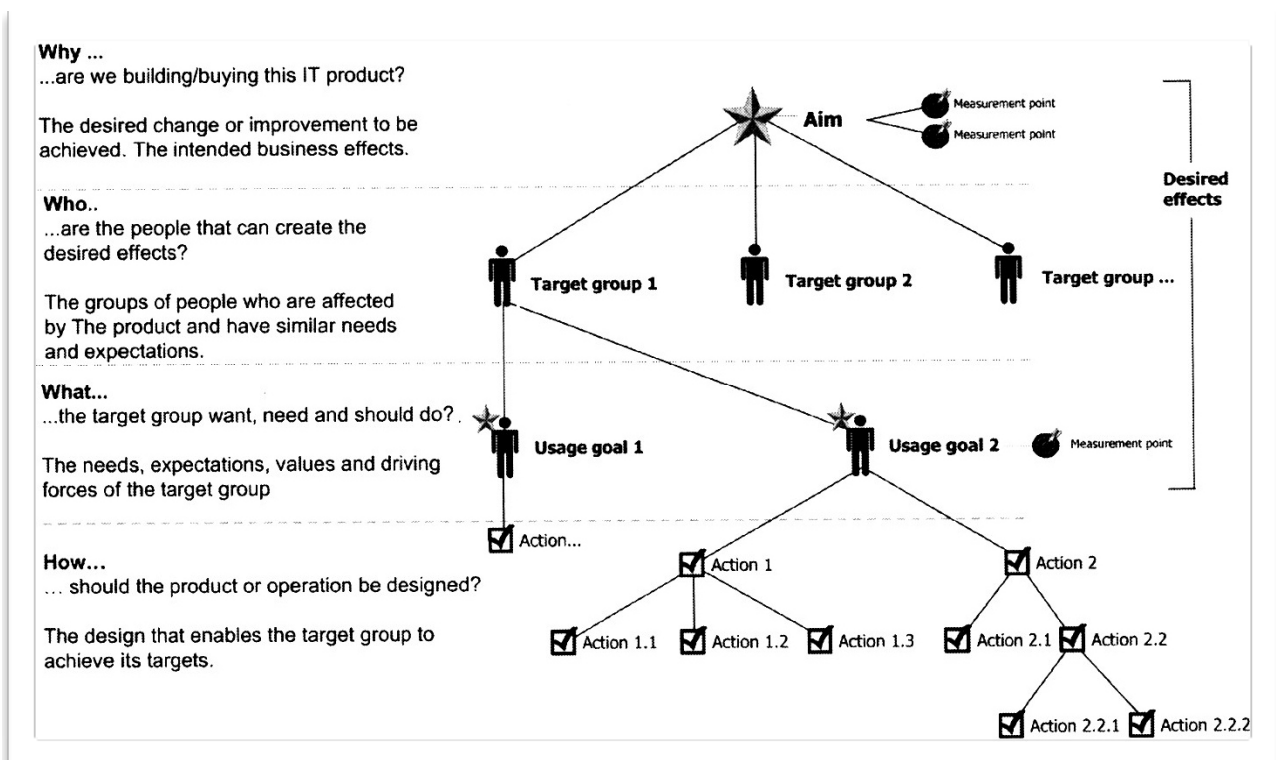


Figure 3 - Schematic figure of an Effect Mapping (Ottersten and Balic, 2007).

The resulting effect map can play different roles in a project. For this project the method is used in three different areas: when managing the project so that the desired effects will be achieved, to ensure that the product's design matches the expectations (supporting data for product design) and finally to structure what has already been completed. (Ottersten and Balic, 2007)

## 2.10 User Requirement Specification

With the usage goals in the effect map have been defined it is possible to specify the products requirements (actions in Figure 3 above). A user requirement specification contains the requirements for the product being developed (Berndtsson and Ottersten, 2002) and is divided into five sub requirement categorisations; Functionality, Usability, Reliability, Performance and Supportability (FURPS+). These are in turn divided into three categories where each of the requirements will be specified, described and graded on a scale of one to five, where a one is of low priority. These are:

- Functional Requirements
- Nonfunctional Requirements (including: usability, reliability, performance, supportability)
- Design Constraints.

A functional requirement describes *what* the system should do and a nonfunctional requirement describes *how* the system should work. Design constraints, on the other hand, is a directive for the developed product, for example: the choice of programming language depends on what is actually being developed. (Eriksson, 2007)

## **2.11 Interaction Design**

An interaction design process can be divided into three stages; fundamental design, functional design and detail design. Each of these stages provides more and more detail towards the final interaction design of the product. Berndtsson and Ottersten (2002) look at it like a funnel where the fundamental design is at the top where the funnel is wide and where detail design is at the bottom where the funnel is narrow, with functional design being in between. This funnel can also represent the focus of the project and the ability to make radical changes. At the top of the funnel radical changes are easily implemented whereas they get harder and harder (as well as more expensive) to implement the closer to the bottom you get.

Throughout this project it is likely that ideas concerning detail design pop up during the first two stages. Here Berndtsson and Ottersten (2002) recommend writing these ideas down and putting them aside for later so as not to get bogged down in details within a process that is meant to be as broad as possible.

### **2.11.1 Fundamental Design**

The fundamental design is a broad strokes approach. The goal for this process is to define and identify the key elements of the product and how they interact with each other and also to weigh the importance of these different elements. A fundamental design describes the patterns which aids the user in completing the task he or she has set out to do. This stage allows the interaction designer to go a little crazy with her ideas as the simplicity of the designs emanating from this process allows a lot of ideas to be considered. The fundamental design is also a good way to communicate the general concept or vision to other parties, such as programmers, clients or market analysts. (Berndtsson and Ottersten, 2002) This will make sure that the grand vision of the interaction designer is also a feasible one as well as one that fits with the identified users and their goals.

### **2.11.2 Functional Design**

Once a fundamental design has been established the project moves on to the functional aspects of the interaction design process. Note, however, that these two processes (fundamental and functional) may overlap somewhat during the design process. Functional design describes the product in more detail, for example: how the user navigates through the product or how feedback is handled throughout the system (Berndtsson and Ottersten, 2002). If the fundamental design focused more on *what* the user should be able to do, the functional design focuses on *how*, where the how is explained as features and functionality within the product. The result of this process is a design that describes all the key elements' different behaviours.

### **2.11.3 Detail Design**

Moving down the funnel, the project will eventually arrive at the detail design stage. The detail design stage takes into consideration things like fonts, colours and diction. Note that

these are not decisions based on aesthetic aspects but rather on cognitive and psychological aspects. That is to say that a recommendation for a colour stems from a reason other than that it looks good. Depending on the project, this process is either very elaborate or not. The more people that are set out to work on finalising the product, such as programmers and graphical designers, the more detailed the detail design should be and vice versa (Berndtsson and Ottersten, 2002).

## **2.12 Design Evaluation**

Evaluations should be performed as soon as it is possible to interact with the product through, for example: prototypes or mockups. Two evaluations will be performed, namely an expert evaluation and five user tests, at the end of a Sprint to bring knowledge of what should be changed for the next one.

### **2.12.1 Expert evaluation**

An expert evaluation should be performed on an interaction designer. It is important that the one who carries out the evaluation has not had any earlier influence on the design of the product. An expert evaluation can go on from a couple of hours up to two weeks.

In this project the shorter evaluation time has been used due to time constraints, and each evaluation was held on one of inUse's interaction designers. The evaluation follows four steps: the first one is for the expert to define the product's aim and decide its extent. Second, the project team will demonstrate the product's content and design. Some form of prototype was used as a mediation tool during the demonstration. Third, the expert answered some questions of the type must be fixed or should be fixed, with regards to issues identified. Finally, the data collected is documented and presented to the project team. (Berndtsson and Ottersten, 2002)

### **2.12.2 User Test**

A user test aims to observe real world usage of the product as a way to find possible flaws design or execution. A potential user performs a number of assigned tasks under the supervision of the project team. Any problems discovered during the test are documented and analysed in order to improve the product.

User test can be performed in many different ways using many different methods. What most of the methods have in common is that they centre around user or users trying to complete a number of tasks. The user is often asked to think aloud when performing the tasks, which makes it easier for the project team observing to understand the user's motivations and feelings. (Berndtsson and Ottersten, 2002)

According to Nielsen (2000) "The best results come from testing no more than 5 users and running as many small tests as you can afford." which is the practice followed during the user testing for the development of the application.

The following methods will be used during the user tests depending on where in the project the team is. They will be used as tools to accomplish the test and to explore, evaluate and communicate design ideas (Ginsburg, 2010).

#### **2.12.2.1 Paper Prototypes**

To define fundamental key design elements, paper prototypes have been created and evaluated with user tests. Paper prototypes can be used as a tool during user tests to eliciting requirements from the user. With use tests performed with paper prototypes it is

necessary for the test leader to act as a "computer" (manage the interface elements), since the prototypes do not allow for any interaction in themselves. (Ginsburg, 2010) Paper prototypes were created by visualising the functionality of the developed iPhone application.

### **2.12.2.2 Static images on device**

When the general interaction flow of the application has been determined through, for example: paper prototypes, static images can be used on the device to elicit for detailed design and functional requirements. By visualising static images on a screen it is possible to elicit visual design detail such as size, colour, labels and text. Another benefit is that it is easier to follow a common thread since no one need to act as the "computer". (Ginsburg, 2010)

Static images were created with Adobe Illustrator and uploaded to an iPhone with the help of the Dropbox app. These images were then used for conducting structured user tests.

### **2.12.2.3 Developing interactive scenarios**

Interactive scenarios are created based on the fundamental and functional design requirements defined. The scenarios make use of actual working code on an iOS device. Each scenario represents a certain function or feature within the app, but without any actual context, instead it is presented detached from the rest of the app. Missing requirements can be captured during user tests using the scenarios. (Ginsburg, 2010)

Scenarios based on the static images were created and implemented. The functionality, design and the interaction of a small part of the application was subsequently tested and evaluated.

### **2.12.2.4 Pilot Session**

A pilot session should be performed when the fundamental design, functional design and detail design requirements of the scenarios have been integrated. The tests of this session could be compared to an overall review of the app where everything has been put together. (Ginsburg, 2010)

The idea is that this should be done in the project when all parts of the application works and feedback is needed.

### 3 Time Schedule

The time schedule is more of an overview of the project than a detailed plan. It is divided into four development phases; Concept, Design & Testing, Development and Report. During the first phase, concept phase, the main goal is to generate a lot of ideas and settle on one for further development. Phase two and three are done in parallel to each other. Each Sprint starts with one week design followed by one week development and ends with one week of evaluation. This process will be performed three times, before the fourth phase take over where the project will be documented and presented orally. The time schedule is presented in a graphical format below (see Figure 4).

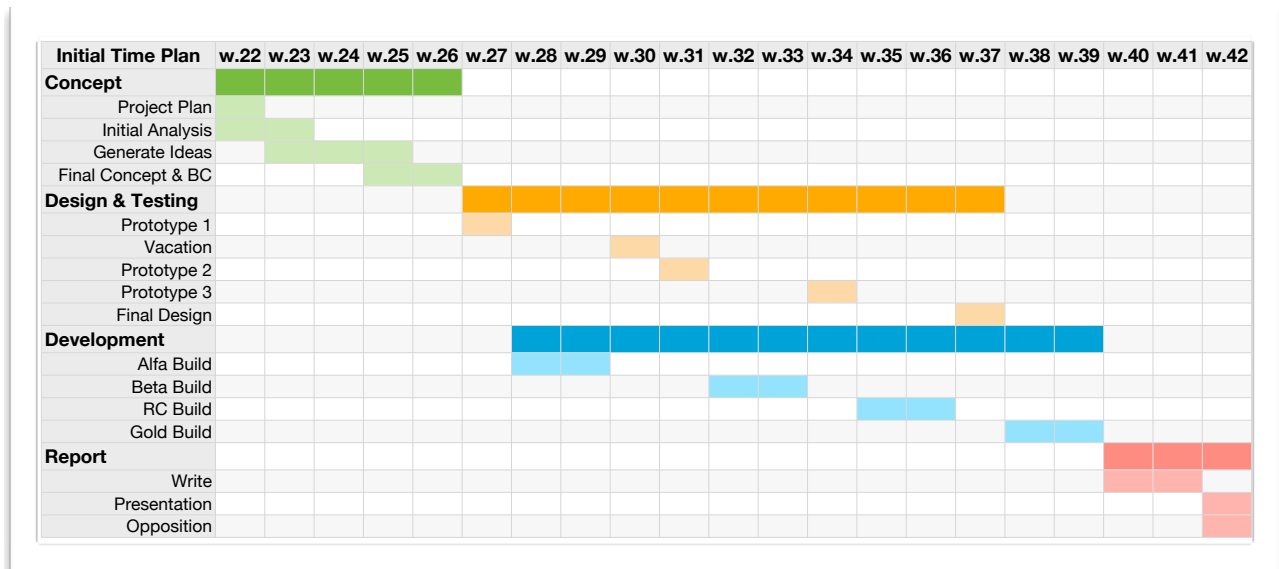


Figure 4 - Initial time schedule for the project.

## **4 Concept Phase**

Five weeks were set aside for the concept phase. The goal of this phase was to generate and evaluate concepts, whereupon one final concept would be chosen to be realised into an App Store app. Note that concept in this case refers to an idea and not to any specifics concerning the idea. Any specification was to be carried out in the design and test phase where the idea would be evolved into an actual product concept and eventually a finished product.

### **4.1 Concept Generation**

At the start of the concept phase a document was set up to store any and all ideas and concepts generated by the project team. This was an online document that the whole project team could edit and add to. During a three week period the methods outlined and described in section 2.4 were used to generate a number of ideas, which were added to the online document. These ideas were in no way filtered, instead care was taken not consider if a concept was already available as an app or how feasible it was in conjunction to the scope of the project. All of the methods used were conducted as planned sessions throughout the concept generation phase by the project team, with the exception of the focus group where six experienced iPhone users took part (see Appendix D for questions discussed during the focus group). Some methods, such as brainstorming and randomised webpages were used more than once over this three week period, as opposed to the focus group which was used once. The media scan was conducted throughout the process as is the nature of the method, since it relies on continuous observation during a period of time to find patterns and common denominators. Following this period of extensive creativity the online document of ideas contained around 150 concepts.

### **4.2 Concept Evaluation**

Considering the amount of concepts generated (around 150) it became necessary to filter these and narrow them down so as to be able to use certain evaluation methods. The list of concepts was analysed by the project team and each member, purely subjectively and independently, narrowed the list down from over 150 to a more manageable number of forty concepts. These forty concepts were then narrowed down further as the project team sat down and cross-referenced lists. This cross-reference, along with a basic market analysis, which meant searching through Apple's App Store for possible already existing apps based on the same concepts, resulted in a list of thirty concepts.

#### **4.2.1 Functional Analysis**

The thirty concepts decided upon by the project team were put through a functional analysis where they were inserted into a matrix and graded in respect to a number of criteria. The criteria were chosen as to make sure that the concept gaining the highest score in the analysis would also be the concept with the highest chance of succeeding within the limitations that were set up surrounding the project. Some of the criteria were lifted from Göransson and Landin (2009) and Rebernik and Bradac (2010). Having identified the criteria necessary the project team set out to grade the thirty concepts. The result from the grading can be seen in Appendix E and F where the top ten concepts are listed.

#### **4.2.2 User Survey & Expert Evaluation**

The top ten concepts, along with a short description of each one, were sent out to the participants of the focus group. They were asked to mark four favourites among the concepts and send the information back. Along with this an expert evaluation was also carried out in regards to the top ten concepts with two senior interaction designers at

inUse. Out of the ten concepts, five were excluded based on the user survey and on recommendations by the experts at inUse, which left five concepts for continued work. These were: Posten, H&M, IKEA, Wish List & Groceries.

### ***4.3 Pitching to Businesses***

Having narrowed down the concept list to five concepts from well over a hundred, the project team started to pitch the concepts. Out of the businesses contacted, two showed interest in collaborating. Scenarios were written for these businesses (see Appendix G). Note, these scenarios are in Swedish since the companies contacted were Swedish-based and were used to explain and clarify the concepts further. The Wish List was only pitched internally to inUse as it does not make use of an external partner or an already established business or brand. A short deadline was set by the project team as a way to make sure not too much time was spent writing up a deal with an external partner, which would take time away from development. Also, since the project was conducted during the summer, there was an inherent risk that contacts would go on vacation before a deal could be finalised. As it turns out this was also what eventually happened. None of the businesses were able to finalise a deal with the project team before the deadline, with one not being able to start discussing a deal before August of 2010. Because of this the project team decided to focus their efforts on the Wish List concept as it, as mentioned before, required no external partner to be realised. This is not to say that the Wish List concept was deemed not as good as the other concepts, only that the project team, from an learning and experience standpoint, were interested in working with an external partner as this is usually how inUse conducts their business. The Wish List concept also scored the highest on the functional analysis matrix, making it an excellent concept for the goals of the project.

## 5 WishLister

The WishLister concept stemmed from the project team's personal experiences as related to gift buying. It was a concept that was conceived early on in the project and which survived the scrutiny of the project team as well as the scrutiny of inUse. Having been chosen as the concept to realise as an app, the project team set out to analyse the WishLister concept with competitor and user analysis and by creating an Effect Map.

### 5.1 WishLister Concept

The name "WishLister" was chosen by sending out a survey where participants were asked to choose amongst four names, their favourite (see Figure 5). These names were chosen by the project team and approved by inUse, by virtue of them being good potential brand names. WishLister got 45% of the votes and was therefore chosen as the official name of the concept.

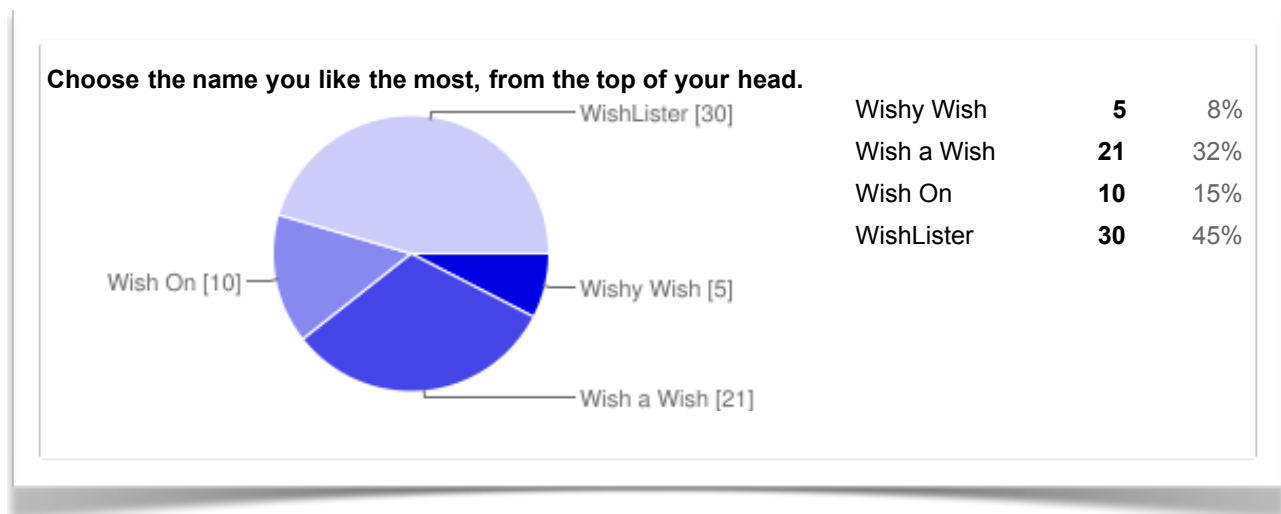


Figure 5 - Result of name survey.

The WishLister concept is meant to make gift buying easier by using the availability of a hand-held device on the iOS platform. The idea is that of creating, and most of all, sharing of wish lists through an iOS app with the help of Facebook's iOS SDK. Creating a wish list should be straightforward as well as distributing it, which is all the owner of the wish list really needs to worry about. For friends and family bound on buying gifts there are a few difficulties that could be made more streamlined and efficient. First and foremost is the tracking of items actually already bought off of the list. Most people have probably, at least once, experienced buying the same gift for Christmas or birthdays as someone else. This is avoidable by making sure to track what everyone else is buying and crossing those things off of the list. Keeping a wish list up to date as to who as bought what can be quite cumbersome however when only making use of e-mails or phone calls. Keeping a centralised list where everyone crosses off items as soon as they are bought would make this process less of a hassle. For weddings and other bigger events where guests may want to get together and buy a gift, the WishLister can make the process of splitting a gift amongst several people a breeze.

### Scenario of WishLister

A scenario for the concept WishLister was created. The first scenario written for WishLister was called Camille's Birthday Bash and can be seen below.

*Camille's thirtieth birthday is coming up and she's planning on having a huge party. To help the guests with their gift purchases Camille has just set up a new wish list for her birthday, and party, in the WishLister application on her iPhone. Over the course of the last few months Camille has been on the look out for potential gifts. Having found a potential gift she has added it to the WishLister app, which now has 12 potential gifts listed. She starts a new wish list, for her birthday, by adding seven of the 12 potential gifts. As she passes by a store, a beautiful vase catches her eye as it reflects the sun into her face. She quickly opens up WishLister on her iPhone and navigates to her birthday wish list. She adds a new item to the list and names it "Kick-ass vase", whereupon she takes a photo of the vase as well as adds its price and where she found it and saves it to the wish list. As she gets on the bus on her way home, Camille shares her wish list with her friends, by logging into Facebook through WishLister and selecting the friends which have been invited to the party.*

*Camille's friend Maya is sitting by her desk reading today's newspaper when her iPhone starts vibrating violently on the table. She picks it up and a push-notification from WishLister tells her that Camille has invited her to see her birthday wish list. Maya scrolls through the items in the list and finds a perfect gift for her to buy, breathing a sigh of relief as she has been wondering what she should buy Camille, since she does not know her that well. She tags the item to make sure no one else buys it before her and puts her iPhone down. At the end of her workday Maya sets off to buy a gift for Camille. Using the information provided by Camille she has no problem finding the store which sells the item. As she pays for the item she picks up her iPhone and marks the gift as bought.*

*Being a captain on a fishing boat means Ben is out a sea for weeks at a time. As such, he has no opportunity to buy Camille a gift before the party as he will dock the same day as the party is scheduled to take place. Because of this predicament Ben loves the functionality built into WishLister which allows him to bid on a gift. Ben scrolls through Camille's birthday wish list and bids 50 dollars, out of the 100 dollar price tag, for a satin scarf. A few days later Ben receives a push-notification from WishLister where Camille's friend Fred has bid the remaining 50 dollars for the scarf. Fred was also willing to buy the scarf, wrap it and write a card.*

*Camille's birthday party was a smashing success and she loved all the gifts she received, not surprisingly as she was the one who picked out most of them.*

## **5.2 Competitor Analysis**

A number of potential competitors were identified and reviewed. The existing wish list services were all web-based, as no apps were found on the App Store (as of July 2010) that would qualify as potential competitors. That is, no apps were found whose functionality in any way resembled that of the WishLister concept. Therefore, a more generic study was conducted of Apple's App Store, which looked at what apps tend to populate the top ten lists and also what apps receive high star ratings.

### **5.2.1 Web-based services**

There exists a couple of websites focusing on the aspect of creating and sharing wishes and wish lists. These are both US-based and Sweden-based. In terms of iOS compatibility none of the websites looked at work well on an iOS device. Nonetheless it was important to try out these services and see what great functionality they offered as well as what not so great functionality they offered, so that these could be implemented and avoided respectively. The competitor analysis consisted of real world usage throughout the concept

and design phase as a way to gather information and inspiration, to facilitate the definition of a target user groups, the creation of an effect map as well as the creation of a requirement specification.

The competitors studied were:

- onskelistan.nu
- wishlist.com
- amazon.com/wishlist
- presentlista.nu
- dreamofthis.com.

### 5.2.2 App Store

Apple's App Store was launched in 2008 and since then over 7 billion downloads have been registered and the App Store today contains over 300 000 apps (Apple, 2010a). Needless to say the App Store is a very interesting market place for developers. The App Store contains top charts, that display the top two hundred apps in three different categories. These are: Top Free Apps, Top Paid Apps and Top Grossing Apps. Exactly how these lists are generated are known only to Apple themselves.

#### Top Chart Analysis

To gain insight into the possible differences between free apps and apps that cost money, the position of the top 25 apps on the Top Free and Top Paid charts were documented during a 12 day period. The first day saw the documentation of what apps were to be analysed, by virtue of them being the top 25 apps on that day. Subsequently, each day the position of these 25 apps was documented, resulting in Figure 6 and 7 seen below.

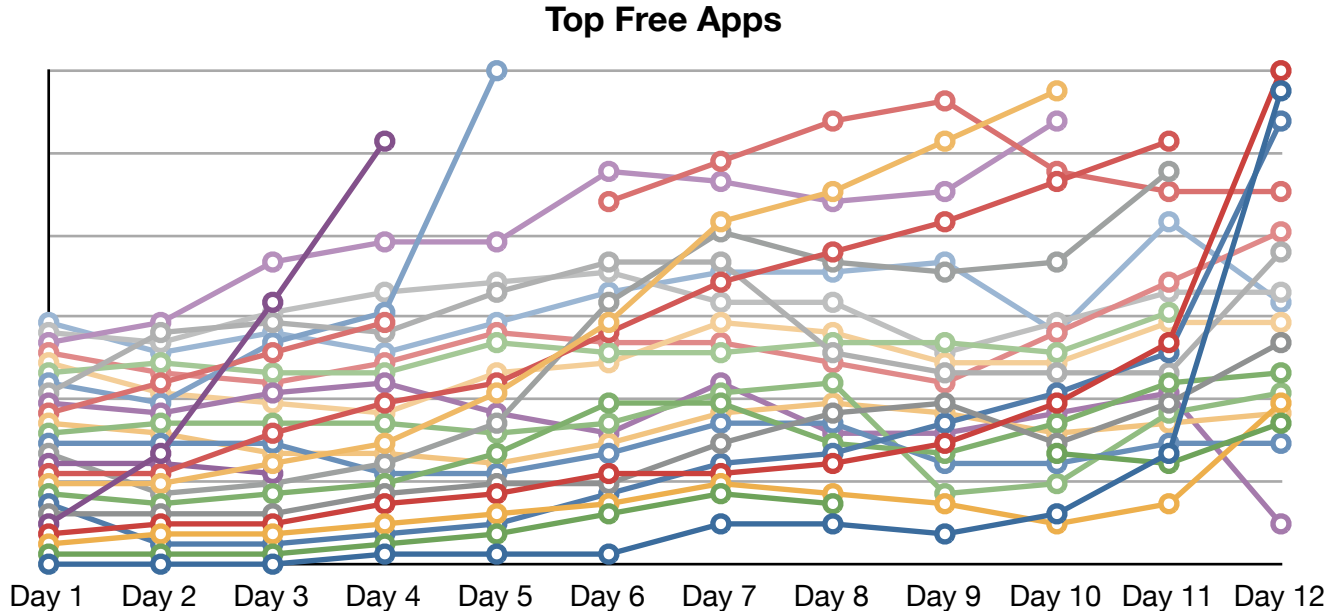


Figure 6 - Top Free Apps analysis.

## Top Paid Apps

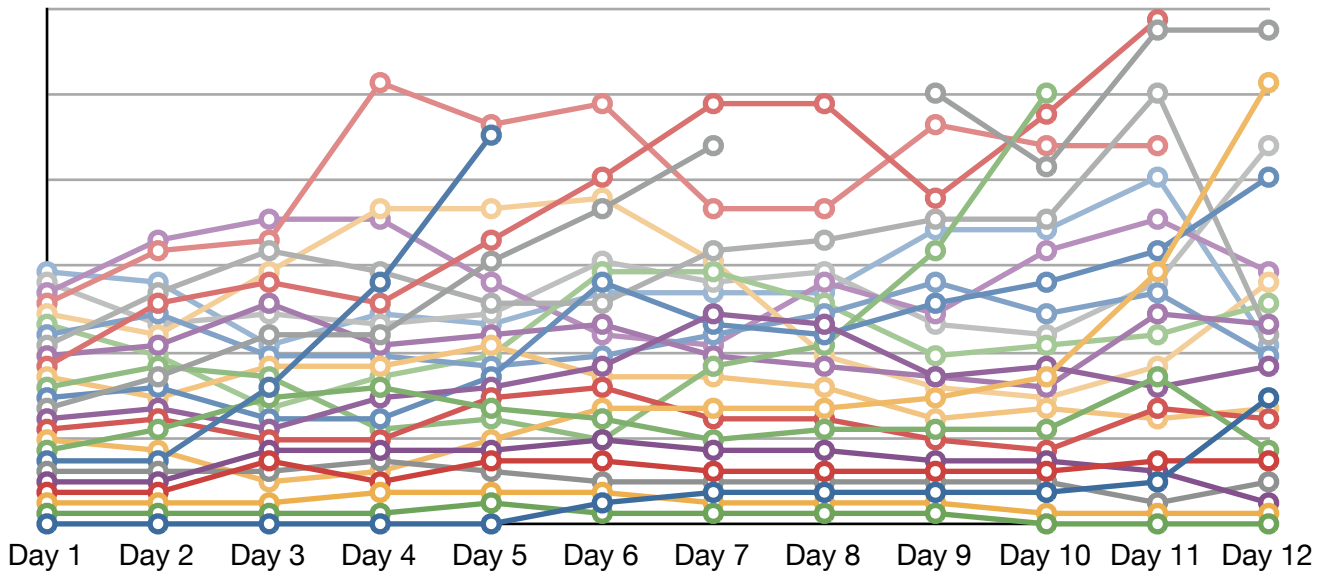


Figure 7 - Top Paid Apps analysis.

As the charts clearly show, the top paid chart is much less prone to extreme fluctuations than the top free chart. Out of the 25 apps on the Top Paid chart, only a few ever lost their place amongst the top 25, as most of the changes to the chart were reorganisations of the apps already there, with only a few new apps ever being introduced. This is radically different from the Top Free chart where changes were more drastic and at the end of the 12 day period most of the top ten apps of the Top Free chart were new apps that hadn't been there when the study began. It is, however, somewhat difficult to draw conclusion from these findings due to the fact that the exact nature by which the top charts are created and maintained is known only to Apple.

### Genre Analysis

Over the same 12 day period as the top chart analysis, the genres of the apps populating the top 20 spots of each top chart (Top Free, Top Paid and Top Grossing) were documented. This was done to see if there was any prevalence of any of the genres to be overrepresented and to see where the WishLister app could fit into the market. As it turns out, games are by far the most popular genre, with navigation apps claiming that spot for Top Grossing apps only. At the other end of the spectrum, productivity apps, weather apps and books are not even present on the chart, showing either a lack of quality products or a lack of user interest (see in Figure 8).

### iPhone Genre Comparison

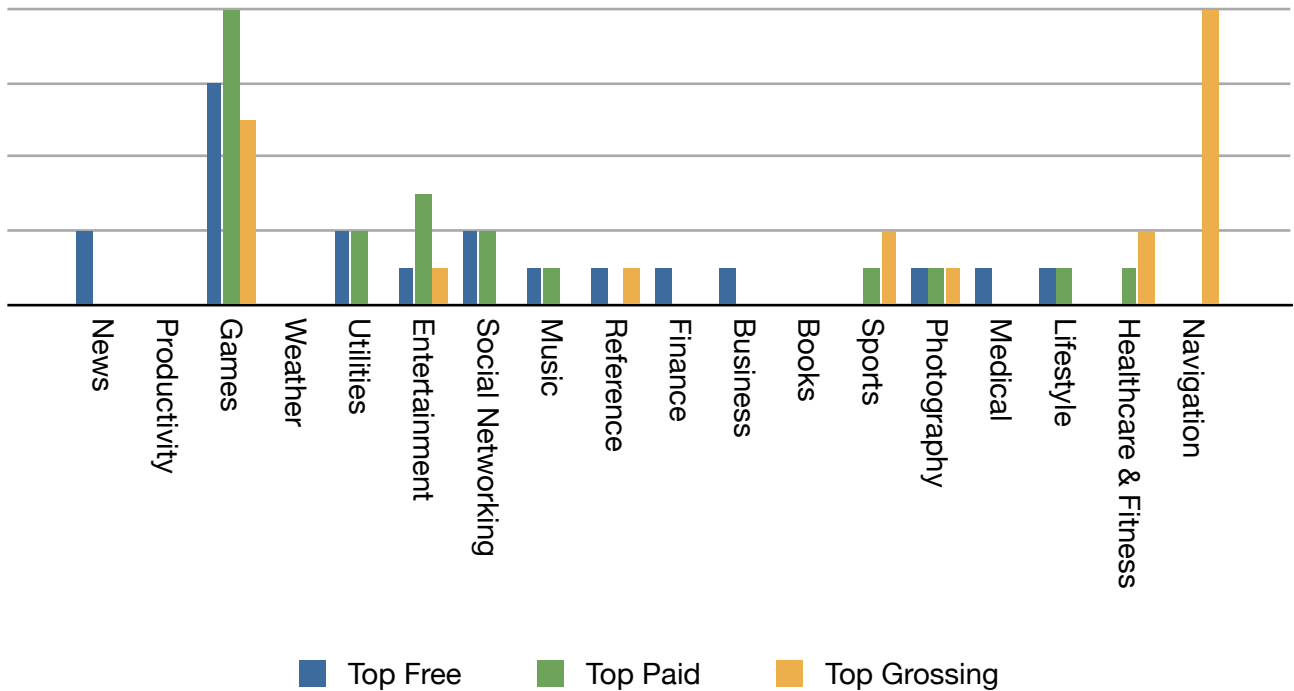


Figure 8 - iPhone genre comparison.

### Star Ratings

The goal of analysing the star ratings of the top 50 apps of each top chart (Top Free, Top Paid and Top Grossing) was to identify any pattern as to how an app is graded. Considering the fact that people pay money for some apps, are these more prone to bad reviews or good reviews, or is the difference negligible compared to free apps? The star rating system of App Store consists of a scale between one and five where users can grade an app at any time from any device running iOS or iTunes. To grade an app a user must have actually downloaded it to their device. The analysis points to an interesting correlation between star ratings and the money paid for an app as is shown in Figure 9.



Figure 9 - Star rating comparison.

The data shows that customer satisfaction (as shown by an app's star rating) varies between the top charts. The Top Free applications had an average star rating of 2.5, Top Paid applications beat that with an average of 3.3 whereas Top Grossing came out on top with an average star rating of 3.8. What might be the reason for this correlation? Some possible factors include:

- users are more inclined to download free apps, as they are free, making the turnover on free apps much greater than that of more expensive apps, as user's download free apps just to try them out. More expensive apps, however might only be downloaded following some form of contemplation on the part of the customer
- the quality of free apps might not be as high as for more expensive apps. Developers might for instance release free apps to try out new functionality
- user's might not judge an app as quickly if they have spent money on it as opposed to having got it for free.

### **5.3 User Analysis**

It is important that WishLister meets the project team's and inUse's goals as well as the users' needs. To identify which user groups exist, a number of questions about potential users were answered. The questions are related to the users current knowledge, experience, tasks, frequency of usage, age, gender and the potential value of the product for the user.

#### **Knowledge**

The product is intended to be distributed internationally, and as such there might be a lot of differences between target users in different countries. One main difference is language. However, most educated people in the world speak or read english to such a degree that using english as the main or only language for the app is not seen as a major flaw or limitation in the design of the product. Providing the product in different languages or perhaps the largest ones in the world is not considered to be major priority but rather something that will be implemented should there be enough time.

The iOS platform is a very homogenous platform and what can be developed for it is quite limited, which means one can draw conclusions about usage of a product from the general guidelines and functionalities of the platform itself.

#### **User experience**

The international market for WishLister should take notes from the cultural and social norms within the country where the it is used. The ability to find out how gifts are bought and for what occasions is hard and a certain amount of generalisation is needed. Specifically the different categories, or occasions, where the product could be used to keep a wish list run the risk of becoming too many if one is to consider any and every event where a gift might be purchased.

Speaking from experience, most kids in Sweden, and probably in Europe and North America as well, use wish lists either to send to Santa for Christmas or to give to their parents for birthdays. These lists can be quite long in younger years, sometimes spanning several pages, and usually become subsequent shorter the older you get. As a kid, one is blessed with parents whose responsibility it is to buy you gifts (looking at it from a child's point of view). As one gets older the focus on gift buying is more on friends and siblings than merely on parents. The logistics of spreading and organising a wish list becomes a lot more complicated as an adult than it was a child for this very reason.

## **Tasks**

The object of WishLister is to help in the organisation and execution of gift buying for any occasion, for example: birthdays, Christmas or weddings. WishLister is designed to help the user not only find what to buy but also where to buy it and to make sure no one else has already purchased the gift in question, eliminating the risk of having bought the same gift as someone else.

Usage of the product can be divided in two categories where one is the creation of a wish list and the other is the actual usage of the wish list by sharing it. The former is focused on the finding and logging of potential gifts, either for a specific event or occasion, or alternatively for one's own usage. The latter is focused on harvesting the wish list for information which can then be used to procure a gift or present for a specific event or occasion. As such the creator of a wish list might not be the actual user of it.

## **Frequency of Daily Usage**

The frequency of usage is extremely dependent on the amount of users. That is, WishLister is at its best when the users' friends and family also use it. The amount of celebrated occasions may also differ between countries and families within a country, which will have effects on the amount of daily usage.

## **Age**

WishLister is designed to suit all ages. For example, toddlers can have access to WishLister through their parents using an iOS device to create and share a wish list for them, until they are old enough for their own iOS device. The goal of WishLister is that people of all ages should be able to use it, thus increasing the number of potential users.

## **Gender**

Statistics concerning the iPhone demographic shows that the division between men and women is approximately 60/40, whereas for the iPod touch the numbers are closer to 50/50 (AdMob, 2010). However, the expectations for WishLister is that both genders will make use of it to the same extent.

## **Values**

Users will choose WishLister based on the increased availability it offers and because it makes gift buying a whole lot easier, as well as making sure that the gift is accepted wholeheartedly by the receiver.

## **User Groups**

From the information collected eight user groups were identified and defined, whereupon two interviews per user group were conducted to create the foundations for the target groups. Seven of these eight user groups were used throughout the project, where the one omitted being The Outsider, whose lack of interest in a product such as the WishLister made that user group irrelevant.

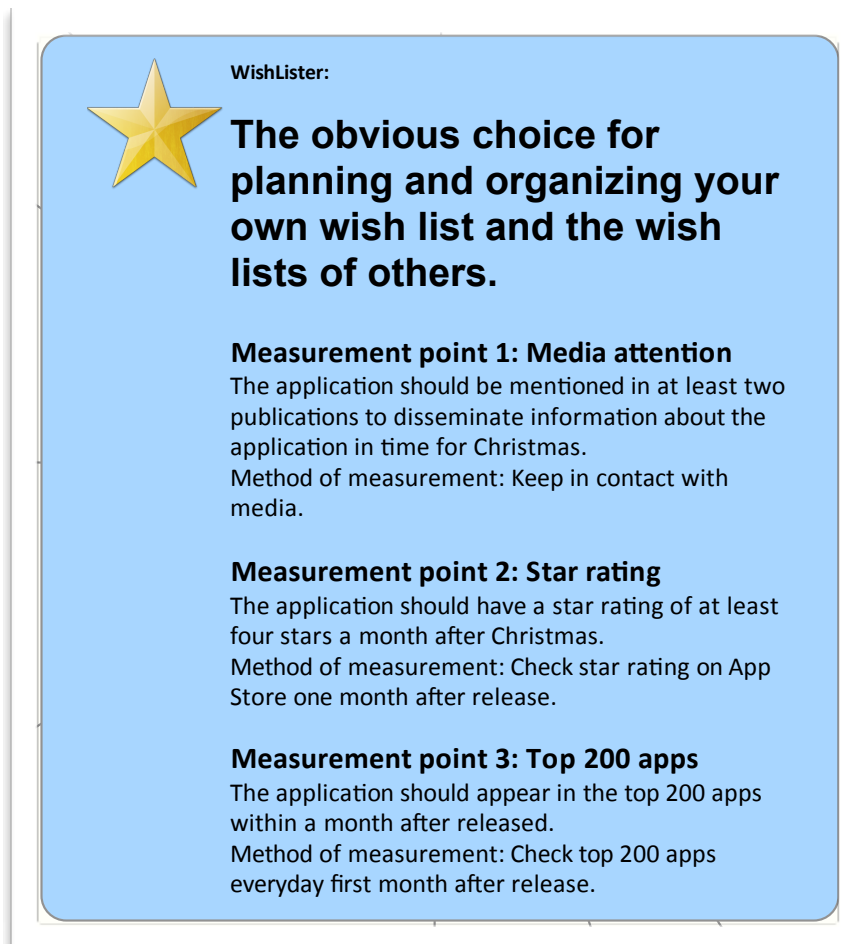
- *The Dreamer* - save wishes for use later
- *The Caretaker* - share her children's wishes
- *The Artist* - wants to know what other people are interested in
- *The Pricerunner* - always searching for the best price
- *The Wishlister* - create and share wishes
- *The Organiser* - wants to store all wishes in a shopping-list for future purchase
- *The Careerist* - wants to find something to buy in an accessible way
- *The Outsider* - not interested in the product.

## 5.4 Effect Map

The Effect Map is used to define WishLister's purpose, target user groups, usage goals and actions. The whole Effect Map can be found in Appendix H.

### 5.4.1 Aim

The aim of the project is also the whole reason for the project to exist in the first place. Bearing this in mind, defining the aim is critical for the project's ability to be successful as well as being the lifeline of the Effect Mapping method. The aim of WishLister was defined as becoming "*The obvious choice for planning and organizing your own wish list and the wish lists of others*". This is the overall aim of the application when it has been released and it is followed by three measurement points which provide measurable data to make sure that the project is moving in the right direction (see Figure 10).



**WishLister:**

**The obvious choice for planning and organizing your own wish list and the wish lists of others.**

**Measurement point 1: Media attention**  
The application should be mentioned in at least two publications to disseminate information about the application in time for Christmas.  
Method of measurement: Keep in contact with media.

**Measurement point 2: Star rating**  
The application should have a star rating of at least four stars a month after Christmas.  
Method of measurement: Check star rating on App Store one month after release.

**Measurement point 3: Top 200 apps**  
The application should appear in the top 200 apps within a month after released.  
Method of measurement: Check top 200 apps everyday first month after release.

Figure 10 - WishLister aim and measurement points.

### 5.4.2 Target User Groups

Fourteen semi-structured interviews were conducted with the goal of eliciting values and thoughts surrounding gift buying and gift giving. These interviews were conducted at inUse in Gothenburg and spanned from half an hour up to an hour and were recorded for later analysis should that be deemed necessary (see Appendix I for the questions used during the interviews). The findings of these interviews laid the groundwork for defining potential target user groups for WishLister. The resulting target user groups were fleshed out using user interaction scenarios and defined in more detail, the result of which can be seen in Figure 11-17 below.



### The Dreamer

*"I want to save wishes for later use."*

**Driving force:** Avoid forgetting wishes.

The Dreamer tends to forget what to put on his wish list when the time comes. Bearing this in mind, the Dreamer wants to be able to save a wish when he comes across it, be it writing down the name of a book seen in a bookstore or photographing a nice vase at a friend's house, the Dreamer wants an easy way to log the information for use at a later time.

Figure 11 - Target Group: The Dreamer.



### The Caretaker

*"I want to organize and share my children's wishes amongst family and friends"*

**Driving force:** To make it easier for family and friends to buy her children gifts.

The Parent wants to be able to organize and share her children's wish lists, so as to make it easier for family and friends to buy them gifts. An application that aids the Caretaker in logging, tracking and sharing her children's wish lists is an application the Caretaker will use consistently.

Figure 12 - Target Group: The Caretaker.



### The Artist

*"I want to buy gifts that surprises the receiver, in a good way"*

**Driving force:** Wants to surprise the person receiving the gift.

The Artist doesn't want to pick an item of a list when buying a gift, instead she wants to tap into the person's inner being and buy them something that they did not know they wanted or needed. There is an art to buying gifts, an art that the Artist wants to perfect. As such she wants as little help as possible but can accept the notion of finding out what a person's interests are, as an aid.

Figure 13 - Target Group: The Artist.



### The Pricerunner

*"I want to get as much gift as I can for my money"*

**Driving Force:** Get her money's worth.

The Pricerunner is very price conscious, which is not to be confused with being cheap, as the Pricerunner can spend a lot of money on a gift but makes sure to get the best price for that specific gift. Having that in mind, the Pricerunner is not loyal to any specific brand or store, instead she shops around to find the best deal. Getting her money's worth also means the Pricerunner likes to band together with friends to buy gifts beyond her budget. The Pricerunner wants an application that can show her the best deals available for a specific product as well as streamline the process of buying a gift together with friends.

Figure 14 - Target Group: The Pricerunner.



### The Wishlister

*"I know what I want"*

**Driving force:** Getting what she needs.

The Wishlister knows what she needs and what she wants and makes sure to write that down. Being the quintessential consumer, she uses wish lists not just for special events but in her daily life as a way to help her friends and family buy her the perfect gift.

Figure 15 - Target Group: The Wishlister.



### The Organizer

*"I plan all of my purchases so that I won't forget anything"*

**Driving force:** Remember to buy gifts.

The Organizer likes to organize things and keep them structured for the sake of not forgetting. The Organizer writes down what to buy and what he has bought and checks things off a list every time he buys something. The perfect application for the Organizer is an application that lets him structure information from other people's wish lists and keep them organized so he can make sure he has bought all the gifts he has planned.

Figure 16 - Target Group: The Organiser.



### **The Careerist**

*"I'm too busy to shop around for gifts"*

**Driving force:** Save time.

The Careerist is a busy man, very busy. As such, he wants to easily find what to buy so that he can go out and buy it and not have to spend time shopping around, considering what to buy. To some extent the opposite of the Artist, the Careerist likes the idea of picking a gift off a wish list.

An application that displays a person's wish list along with information on where to buy said gift is a perfect application for the busy Careerist.

**Figure 17** - Target Group: The Careerist.

### **5.4.3 Usage Goals**

"Usage goal describes what the different groups of users need or expect, and what should happen in order for the desired effect to be achieved" (Ottersten and Balic, 2007). For each target user group, a number of usage goals were identified by the project team and inserted into the effect map (the entire effect map can be found in Appendix H). These usage goals sometimes overlap for different users, while no two users have exactly the same usage goals. For the more general usage goals which are shared by all the users, a specific category was created called 'all users'.

#### **The Dreamer**

- Wants to register sporadic gifts on the fly
- Wants to be able to specify detailed information about his wish
- Must use the application every time he purchases a gift.

#### **The Caretaker**

- Wants to be able to get feedback on what has been bought off of a wish list
- Wants to be able to share wish lists with friends and family
- Must use the application every time she purchases a gift.

#### **The Artist**

- Wants to see friend's wish lists to get inspiration.

#### **The Price Runner**

- Needs to be sure that she has found the best price
- Wants to be able to bid on a gift
- Wants to be able to see who else has bid on a gift
- Must use the application every time she purchases a gift.

#### **The Wishlister**

- Wants to receive gifts from her wish list
- Wants to be able to share wish lists with friends and family
- Wants to be able to categorise wish lists depending on event
- Must specify detailed information about her wishes
- Must make sure her friends use the application
- Must use the application every time she purchases a gift.

### The Careerist

- Need feedback every time a friend's wish list is updated
- Wants to see detailed information about his friend's wishes
- Must make sure his friends use the application.

### The Organiser

- Needs to be able to organise his gift shopping list
- Wants to make sure he has bought all the gifts he is supposed to
- Must use the application every time he purchases a gift.

### All Users

- Simple, uncluttered and easy to use interface
- Bug free
- Know who the owner of a wish and wish list is.

### 5.4.4 User Requirement Specification

In the Effect Mapping method the usage goals leads to the creation of actions as a means of solving and fulfilling the needs of the users. In this section the actions that were defined based on the usage goals will be described and ordered into four main categories; functional requirements, nonfunctional requirements, design constraints and inUse expectations (see Tables 2 through 5). Each requirement will be graded with a number between one and five, where one is low priority and five is high priority, with regard to the implementation of the functionality.

Table 2 - Functional Requirements.

NO	Requirement	Description	Priority
1	Logo	The application should have an easily recognisable logo	5
2	Name	The application should have an easily recognisable name	5
3	Create wish	A user shall be able to create a wish	5
4	Add wish to wish list	A user shall be able to add a wish to a specific wish list	5
5	Create wish list	A user shall be able to create a wish list	5
6	Specify name of wish	A user shall be able to specify a name for a wish	5
7	Specify name of wish list	A user shall be able to specify a name for a wish list	5
8	Delete wish	A user shall be able to delete a specific wish	5
9	Delete wish list	A user shall be able to delete an entire wish list	5
10	Navigation	A user shall be able to navigate between different views	5

<b>NO</b>	<b>Requirement</b>	<b>Description</b>	<b>Priority</b>
11	Share a wish list	The user shall be able to share a wish list with friends and family	5
12	Save wish information	A user shall be able to save information about a wish on the device	5
13	Delete wish information	A user shall be able to delete information about a wish from the device	5
14	Reserve friend's wish	A user shall be able to reserve a friend's wish in a wish list	5
15	Reserved wish	The owner of a wish shall not be able to see if a wish has been reserved	5
16	Specify picture for wish	A user shall be able to use the iPhone camera to save a photo together with a wish	5
17	Facebook Connect	The application shall allow the user to login to Facebook via Facebook Connect	5
18	Multiple wish lists	A user should be able to navigate between wish lists	4
19	Usage without Facebook	A user should be able to use the application without logging in to Facebook	4
20	Shopping list	Reserved wishes should be placed in a shopping list	4
21	Edit wish	A user should be able to edit the information of a wish	4
22	Edit wish list	A user should be able to edit the information of a wish list	4
23	Accept invitation	A user should be given a choice about who to interact with through the app	4
24	Push-notification	Subscribers to a wish list should be able to receive notifications about news and changes concerning a specific wish list	3
25	Universal wish list	A user should be able to save wishes without adding them to a specific wish list	3
26	Specify store of wish	A user should be able to specify a store for a wish	3
27	Specify price of wish	A user should be able to specify a price for a wish	3

NO	Requirement	Description	Priority
28	Specify notes of wish	A user should be able to specify notes for a wish	3
29	Comment on wish	A user should be able to comment on a wish	2
30	Invites	A user should be able to give invites to the application to non-users	2
31	Partial purchased	A user should be able to get together with a friend and reserve a wish	2
32	Sort wish list	A user should be able to sort wishes within a wish list	2
33	Sort shopping list	A user should be able to sort intended purchases in the shopping list	1
34	Login to Facebook	The application should keep the user logged in to Facebook until the user manually logs out	1
35	Categorisation	It should be possible to categorise wishes and wish list to specific categories as Christmas, birthday etc	1

**Table 3 - Non-Functional Requirements.**

NO	Requirement	Description	Priority
36	Usability	The application shall follow Apple's iPhone human interface guidelines	5
37	Availability	User's shall be able to use the application anytime	5
38	Security	The application shall not hand out personal information without permission from the user	5
39	Dependability	The user created information shall never be deleted without user participation	5
40	Compatibility	The application shall be compatible with other devices	5
41	Reliability	The application shall have a high fault tolerance	5
42	Flexibility	It should be easy to make future development changes in the application	3
43	Speed	The application should not be unreasonably slow	3

**Table 4 - Design Constraints.**

NO	Requirement	Description	Priority
44	Apple's iPhone guideline	Follow Apple's iPhone human interface guidelines	5
45	Xcode	Use Apple's IDE for development	5
46	Ownership	The user should always know who's which list/view she's looking at	5
47	PHP	Use PHP scripts to allow communication between the database and its web server	4
48	JSON	Use JSON to allow communication between the device and PHP, which in turn allows communication between the device and database	4
49	Colours	The application should make use of no more than 3-4 main colours	4
50	iPhone components	Make use of Apple's own components to facilitate future updates	4
51	Illustrator	Make use of Illustrator to visualise prototypes	4
52	Colour blindness	The application should make use of colours that allow colour blind people to use it	3
53	Categorisation	Each category should be represented visually	3

**Table 5 - inUse Expectations.**

NO	Requirement	Specification	Priority
54	Media attention		5

## 6 WishLister Development

The following chapter contains words specifying certain standard interface components of a typical iPhone user interface. For this reason, it is recommended that anyone not familiar with the iOS nomenclature read through Appendix J before reading through this chapter. This chapter provides an overview of the changes and feedback in each Sprint, however it does not contain an overall walkthrough of the WishLister app. Instead, a walkthrough is provided for the final result in Chapter 7.

With a clearly defined concept along with a competitor analysis, a user analysis and an Effect Map, work started on developing the idea of the WishLister application. A pre-study was conducted to identify core functionality and how that should be structured in the application. Five main areas were identified and their tasks were listed in a hierarchical diagram (see Figure 18). From this, work started on constructing prototypes for testing evaluation. The idea was to focus the first two Sprints on paper prototypes, moving on to static images on device (iPhone and an iPod Touch) for the next two Sprints while finally moving on to interactive scenarios for the remaining five Sprints, ending up with a finalised version used in a pilot session. This outcome does not match the time schedule presented in Chapter 3. Why this is, is discussed in more detail in Chapter 8. Also, time constraints lead the project to be finished before a pilot session could be carried out within the scope of this master thesis (see section 6.4).

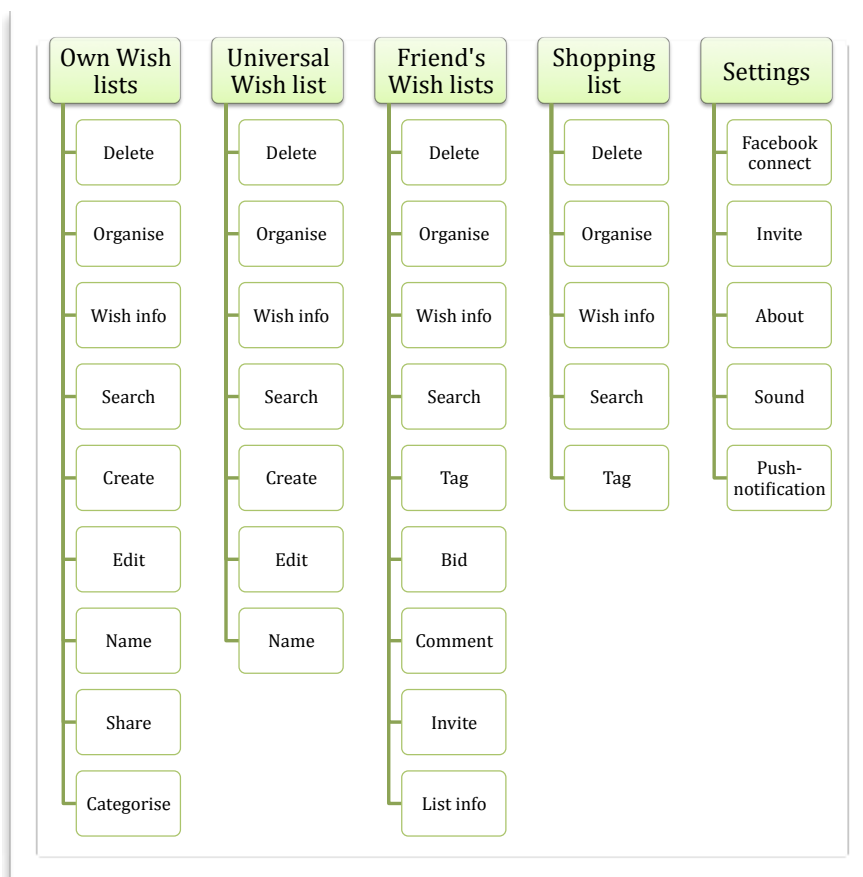


Figure 18 - Core functionality of WishLister.

## 6.1 Paper prototypes

Through the use of paper prototypes the fundamental requirements were outlined and defined. It was important to keep the first Sprint as broad as possible, so as to leave room for quick changes. Focus was instead on producing many ideas and concepts around the central theme and requirements that had been defined during the concept stage.

### 6.1.1 Sprint 1

The first Sprint made use of paper prototypes and whiteboard prototypes. The paper prototypes were conceived through the use of an iPhone stencil sheet where the project team could get a feel, from the start, for the limitations of the iPhone's screen. A large iPhone stencil was used together with a whiteboard (see Figure 19) to help see the developed prototypes in a different light.



Figure 19 - Whiteboard prototype.



Figure 20 - Paper prototype.

Two expert evaluations were conducted with interaction designers at inUse and made use of the paper prototypes developed. The prototypes were enclosed in an iPhone case (see Figure 20) so as to simulate real world usage, whereupon the experts were given a few simple tasks to complete. The evaluation brought to light strengths and weaknesses of the prototypes, along with tips on solutions.

### 6.1.2 Sprint 2

Taking all the good things about the prototypes from Sprint 1 and cutting out the bad parts, a new prototype was developed. The prototype made heavy use of inspiration from other iOS apps and was developed with focus on the fundamental aspects of WishLister and its uses. Starting out as sketches on a whiteboard then sketches on paper (see Figure 21), before being transferred into Illustrator where it was cleaned up and refined (see Figure 22), the prototype was readied for user testing. The five tabs present in this prototype lived on throughout the entirety of the project and will be referenced in later Sprints. The five tabs, as seen in Figure 22 were defined as, from left to right and top to bottom:





Figure 22 - Sprint 2 Illustrator prototype.

## 6.2 Static images on device

The Sprints in this section were focused both on the fundamental and the functional design aspects when developing new prototypes. The prototypes developed were evaluated through the use of static images displayed on either an iPhone or an iPod Touch, where simple functionality was illustrated by switching between images. Adobe Illustrator was used in the development of the prototypes and they made use of ready-made templates for standard iOS components as well as custom made graphics.

### 6.2.1 Sprint 3

Based on new elicited requirements from the user tests in Sprint 2, a new prototype was developed. Functionality and design were improved upon while fundamental parts of the earlier prototypes were mostly retained. Aspects of the interaction design concerning functionality, such as how to navigate and handle user feedback, were now prioritised in order to capture these requirements. Presented below is a list of changes made.

- Implemented ability to edit wishes after they have been created by navigation user back to add wish tab. Editing a wish also allows for placing or moving the wish into a list (Figure 23, left)
- Simplified the functionality of deleting a friend's wish list by adding a deletion cross that is always visible (Figure 23, middle)
- Added friend's name for each wish in the shopping list to show who the wish is for (Figure 23, right)
- The shopping list is no longer sorted alphabetically, to give users the ability to reorganise the list themselves (Figure 23, right)
- Reordering wishes is now possible, in the user's own wish lists and the shopping list, by pressing, holding and dragging them (Figure 23, right)
- New tab bar icons implemented (Figure 23).

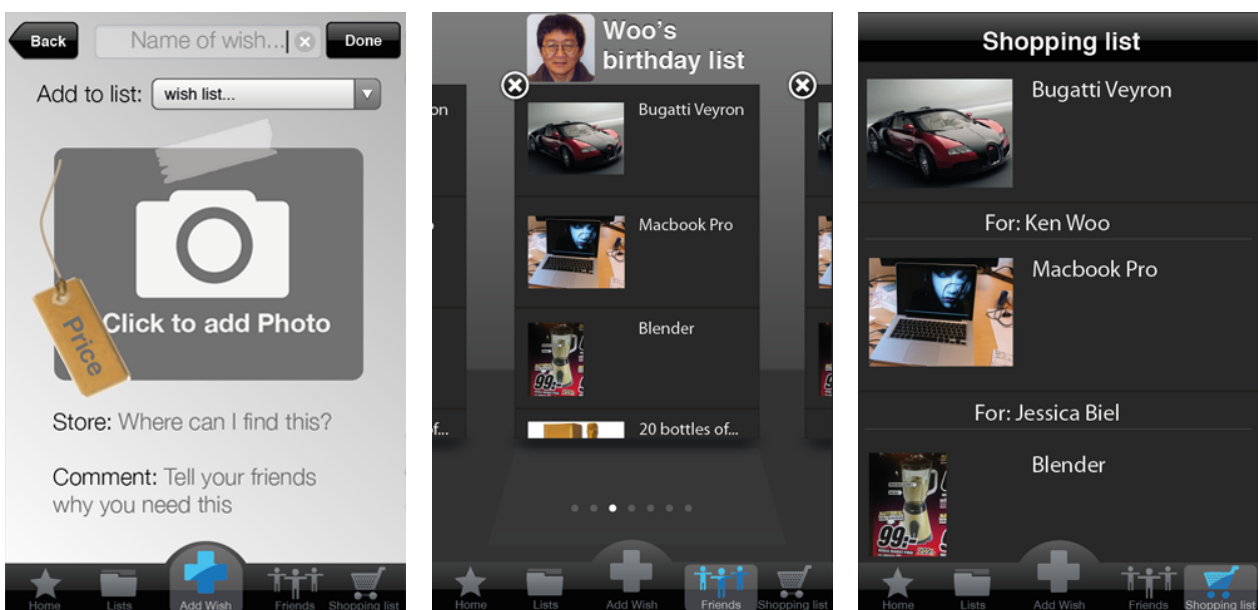


Figure 23 - Images on device from Sprint 3.

The evaluation of the prototype was this time concerned with both fundamental and functional design aspects. Results of the five user tests conducted are listed below. Along with these five user tests, an expert evaluation with an independent iOS developer was also conducted and the results from this evaluation are also present in the list below.

- Deleting a wish should present the user with more feedback
- Edit wish list button was mistaken for standard edit button but is in fact a create new button according to Apple's official documentation
- How to share a list is unclear
- Deleting a wish from a wish list might be too cumbersome
- Having a deletion cross visible all the time, for wish lists, inconsistent with behaviour in other parts of the app
- Reserving a wish lacks user confirmation and is undoable.

### 6.2.2 Sprint 4

This Sprint contains the last static image prototype as following Sprints concern the actual implementation of WishLister on the iOS platform. Therefore, it was important to capture fundamental and functional design requirements that had so far been missed. A requirement for proceeding onwards with the development was approval from inUse that the WishLister met their standards in terms of functionality and interaction design. Changes made due to user feedback from Sprint 3 are listed below.

- Deleting a wish pad now displays a dialogue box requesting user confirmation for the action (Figure 24, top left)
- Edit button changed to a standard button labeled Edit (Figure 24, top middle)
- The functionality of sharing a list was implemented in a new button labeled Share (Figure 24, top middle)
- Edit button in opened wish list omitted, swipe delete retained (Figure 24, top right)
- Deleting a friend's wish list is now done with a button labeled "Delete", and also requires user confirmation (Figure 24, bottom left)
- Reserve wish functionality unchanged as technological limitations were unknown at the time, however the reserved by: box in friend's list changed to be clearer (Figure 24, bottom middle).

A few other changes were made that did not not relate to user feedback but instead related to the existing requirement specification.

- Labels in dialogue box when deleting a wish list or a friend's list changed to correspond to behaviour when deleting a wish pad (Figure 24, top left).
- User prompted to accept or deny invitation from friends to access their wish lists, so as to give the user a choice about whose wish lists to view (Figure 24, bottom right)
- Colour used to differentiate between functionality concerning a user's own lists and a user's friend's lists (green and blue respectively) (Figure 24)

The results of the five user tests as well as the expert evaluation of an interaction designer at inUse are shown below.

- No option to clear input fields when creating a wish
- How and where to enter a wish name is unclear
- Add wish screen is too cluttered
- Dots used at bottom of a wish list, to show there is more content, confusing to user.

Following the expert evaluation and user tests, WishLister was approved by inUse for further development. InUse was however concerned with how the application would be distributed, in other words; should it be free, free with advertisements or should it cost money? Following a dialogue with inUse the decision was made that WishLister should be distributed as a free app containing advertisements using Apple's iAd platform.



Figure 24 - Images on device from Sprint 4.

### 6.3 Interactive scenarios

Following approval from inUse, development started on interactive scenarios for WishLister. Interactive scenarios are here defined as pieces of functionality taken from previous prototypes and implemented on the iOS platform and that could be run as an application on an iPhone or iPod Touch. The implementation work process made use of pair programming. Focus here was on detail design as well as on identifying missing functionality that had been lost in earlier Sprints. Note that only the first three tabs (Home, Lists and Add Wish) are mentioned throughout the Sprints in this section, and this is because the basic functionality of the last two tabs (Friend's Lists and Shopping List) is more or less the same as that of the wish list tab. Therefore, it was deemed unnecessary to implement these until the design and functionality of the first three tabs had reached a mature level.

#### 6.3.1 Sprint 5

Due to the lack of experience in working with the iOS platform, the first prototype worthy of user testing took four weeks to develop and contained basic functionality, thus making this

Sprint a lot longer than previous and following Sprints. The prototype was based on the static images tested during the previous Sprint and was focused on the creation and deletion of wishes and also how they were presented to the user. Some changes were made based on user feedback from Sprint 4.

- Creating a new wish list was not possible in earlier prototypes. A functionality that apparently fell through the cracks. The functionality was implemented with inspiration taken from Apple's app store (Figure 25, top left & top middle)
- Creating wishes and wish lists with the same name or with no name should not be possible, due to technical limitations, and as such functionality was implemented to prevent this from happening (Figure 25, top right)
- Undo button implemented when adding wish so as to clear out fields in case of mistakes (Figure 25, bottom left)
- Add wish screen reorganised and restructured (Figure 25, bottom left).

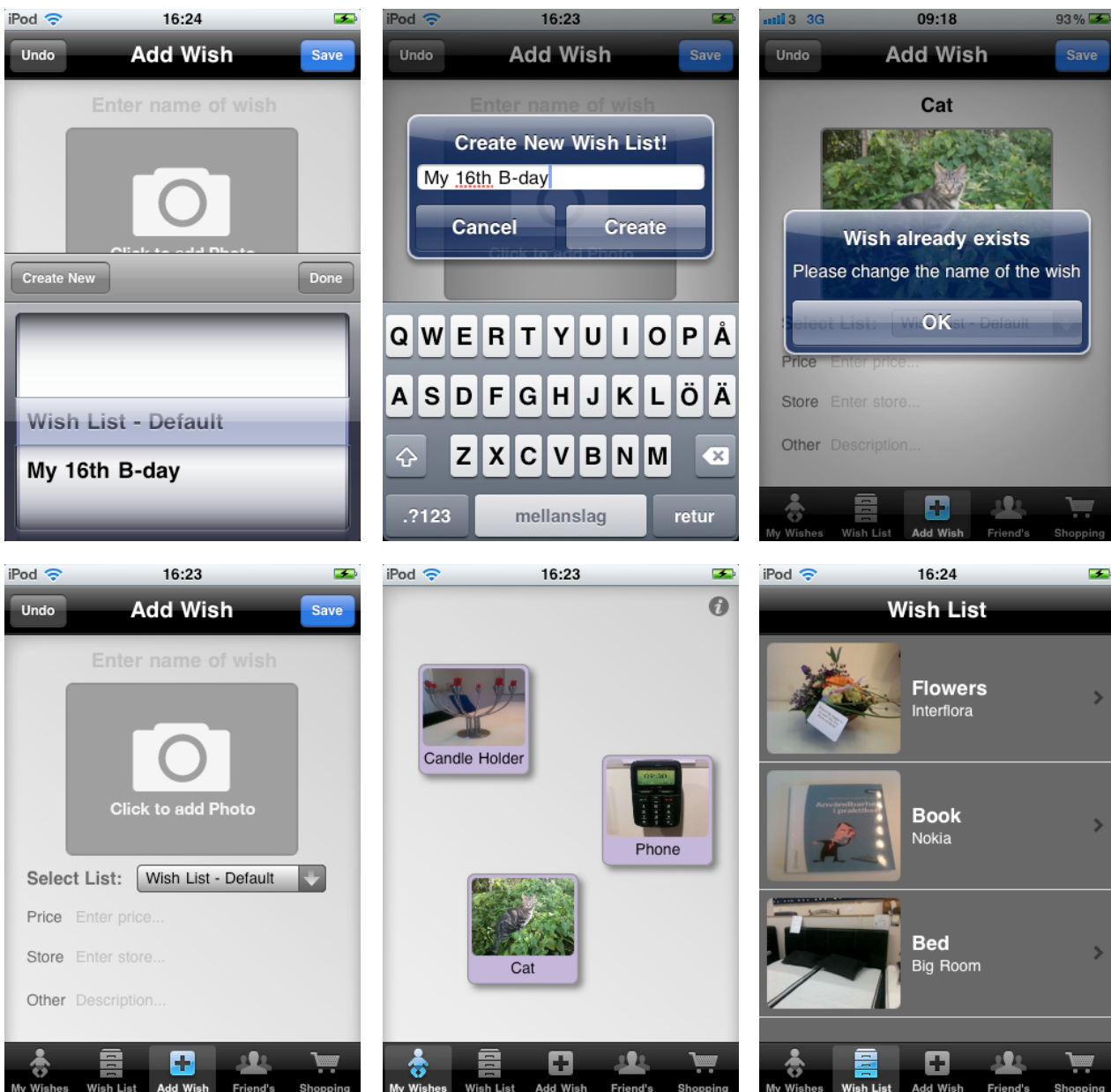


Figure 25 - Screenshots from Sprint 5.

This Sprint provided the first user tests conducted on the iOS platform by the use of an iPhone. The five user tests focused mostly on adding, viewing and deleting wishes. Since focus was so limited the results were also quite sparse compared to previous testing sessions. The main points from the tests are listed below.

- The icons for the tab bar need to be clearer in communicating their function
- The view for creating a new wish is somewhat cluttered and unclear
- Interaction with the wish pads, moving, pressing, pressing and holding, was not as intuitive as expected.

### **6.3.2 Sprint 6**

Having implemented basic functionality, this Sprint focused on refining this basic functionality, mostly in terms of how wishes are represented and also how the interaction with multiple wish lists is managed. Changes for this Sprint are shown below.

- Tab bar icons changed to better communicate their function (Figure 26)
- Backdrop for wish pads added (Figure 26, top left)
- Graphics of wish pads changed in an attempt to make interaction more intuitive (Figure 26, top left)
- Creating multiple wish lists implemented (Figure 26, top middle)
- The screen for creating a new wish was completely overhauled in an attempt to make the interaction with more intuitive (Figure 26, top right)
- iAd placeholder implemented in detail wish view (Figure 26, bottom left)

One interaction designer at inUse and five potential users were used in testing the application. The goal of the tests was to get a feel for how the entire life cycle of a user created wish worked and where potential problems may lay.

- Consider the amount of taps and views used. Edit mode for wishes maybe redundant
- Price tag and name input fields are both unclear when creating a new wish
- Interaction with wish pads still somewhat unclear
- Swipe delete in opened wish list unclear. No user figured out how to delete a wish without help, despite this being standard iPhone behaviour
- Reorganising wishes in list not possible. A functionality that also seems to have slipped through the cracks
- Share button for wish list unclear. Users were unsure about which list the share button belonged to
- How to delete a wish pad unclear. All users tried pressing and holding but failed to bring up the delete dialogue box.

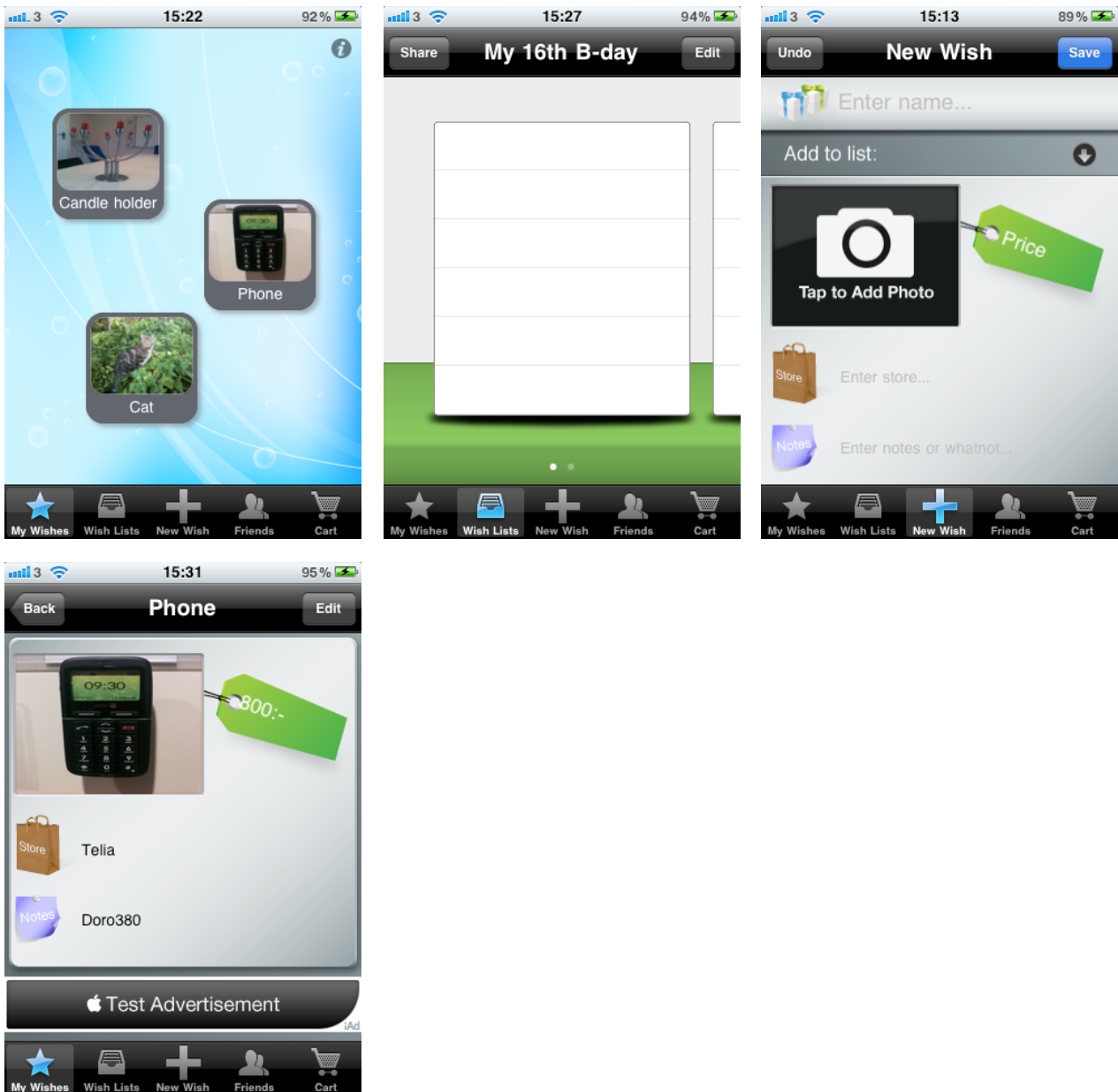


Figure 26 - Screenshots from Sprint 6.

### 6.3.3 Sprint 7

From expert evaluation and user tests conducted it became obvious that WishLister contained some matter of redundancy. Bearing this in mind, this Sprint focused on optimising the amount of steps and interactions (changes can be seen in the list below) needed to complete a specific task in an attempt, to keep the application lean and to maintain high usability.

- Wish pads redesigned (Figure 27, top left)
- New wish screen tweaked to allow for faster wish creation (Figure 27, top middle)
- Edit mode for wishes was removed in favour of editing wishes directly when viewing them
- Edit button added in opened wish list view to make deletion of wishes more clear as well as allowing for organising and sorting of wishes (Figure 27, top right & bottom left).

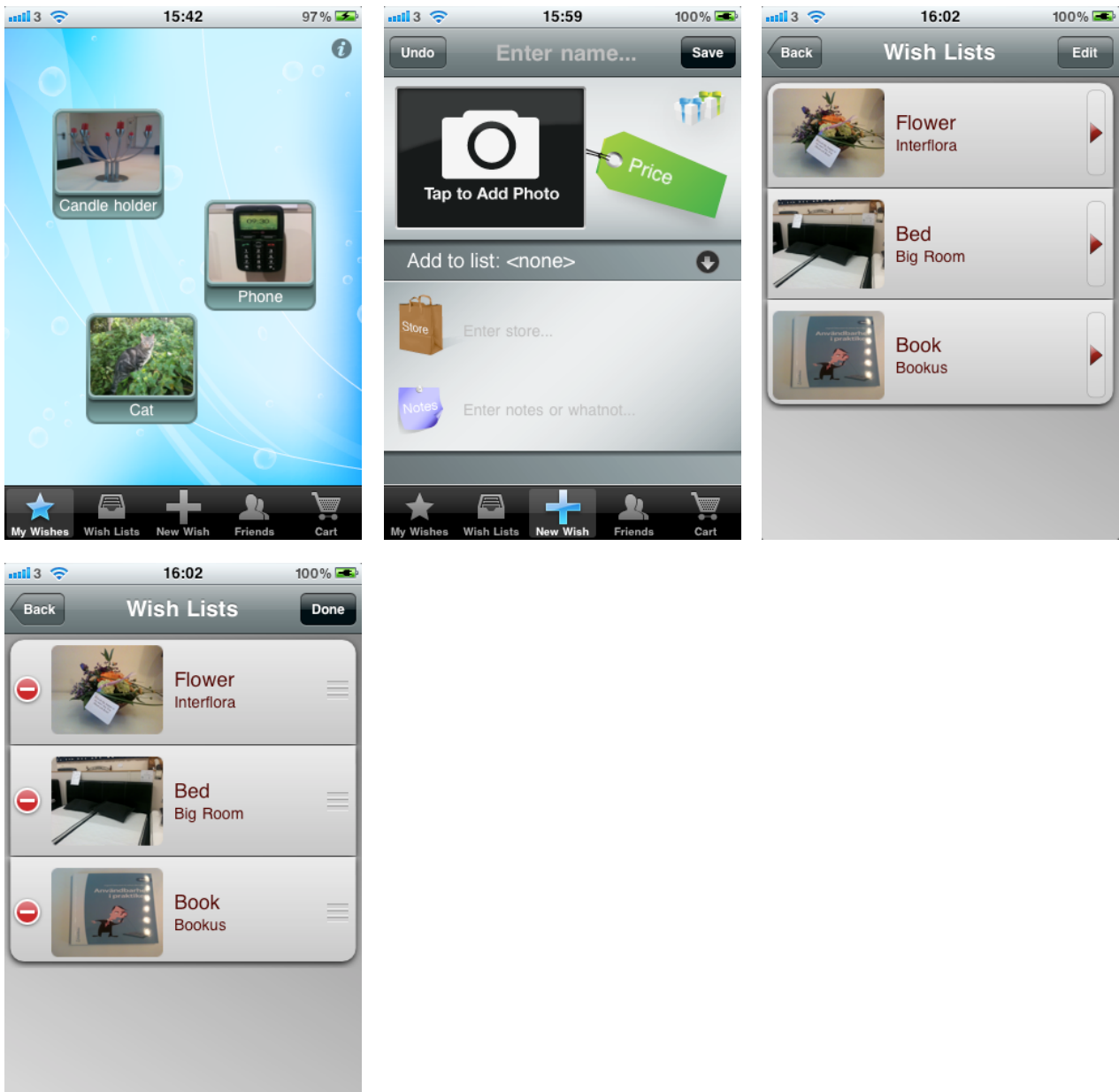


Figure 27 - Screenshots from Sprint 7.

The result from the expert evaluation and the five user tests made it clear that omitting the edit mode gave users a better understanding of the interaction with the application. The changes to the wish lists also meant that no user failed in deleting a wish or sorting a wish list. The expert evaluation was done with a graphics designer, who suggested a number of changes and gave tips on how to improve the graphical aesthetics of the application.

### 6.3.4 Sprint 8

Requirements gathered from the user tests in Sprint 6 and 7 were evaluated and implemented during this Sprint, since Sprint 7 only focused on optimising usability. From what was brought in previous user tests, this Sprint focused on both optimising and improving existing functionality. The following changes were implemented:

- The delay from pressing and holding a wish pad to the delete dialogue box appearing was reduced from one second to 0.4 of a second (Figure 28, top left)
- Added fade on wish pads to show when they are being acted upon (Figure 28, top left)
- Changed background image for my wishes (Figure 28, top left)

- Removed the navigation bar in wish lists view (Figure 28, top middle)
- Changed the edit button in wish lists view to a delete button, as that was its only usage when name change was moved into opened wish list view (Figure 28, top middle)
- Changed the design of the share button and delete button for wish lists (Figure 28, top middle)
- Tweaked the graphics for wish lists (Figure 28, top middle)
- A dialogue box will appear when pushing the implemented delete button in the wish list view (Figure 28, top right)
- It should be possible to change the name of a wish list (Figure 28, bottom left)
- Tab bar hidden in wish detail view and in opened wish list, to acquire more screen real estate (Figure 28, bottom left)
- Changed colour of navigation bar in opened wish list to green (Figure 28, bottom left)
- It is now possible to login to Facebook and see friends in a table view when sharing a wish list (Figure 28, bottom middle)
- Changed the name of undo button from undo to clear (Figure 28, bottom right)
- Changed the name of done button when editing wish from done to save (Figure 28, bottom right)
- Changed colour of the navigation bar to green in new wish (Figure 28, bottom right)
- Changed the graphic for new wish view to help users find wish name input field (Figure 28, bottom right)
- Miscellaneous graphic tweaks (Figure 28).

Five user tests were conducted on potential users. The focus during these tests were on the input of data and on bringing up new possible design constraints and missing functionality. One minor problem, which had been discovered in earlier user tests, was that the new wish view was still somewhat hard for users to understand. In response to that, the design and the text input field placeholders were evaluated and improved upon for the whole application to be more consistent and to increase usability, in the following Sprint. From the user tests the following feedback was noted:

- The text input field placeholders and overall graphics in the new wish views needs to be clearer, as they tend to confuse the user.
- Problem to find where to enter the name of a wish
- Using <none> to signify that a wish will not be placed in a wish list can generate confusion for users
- The application needs to follow a common thread and be more consistent throughout, e.g colours, functions.

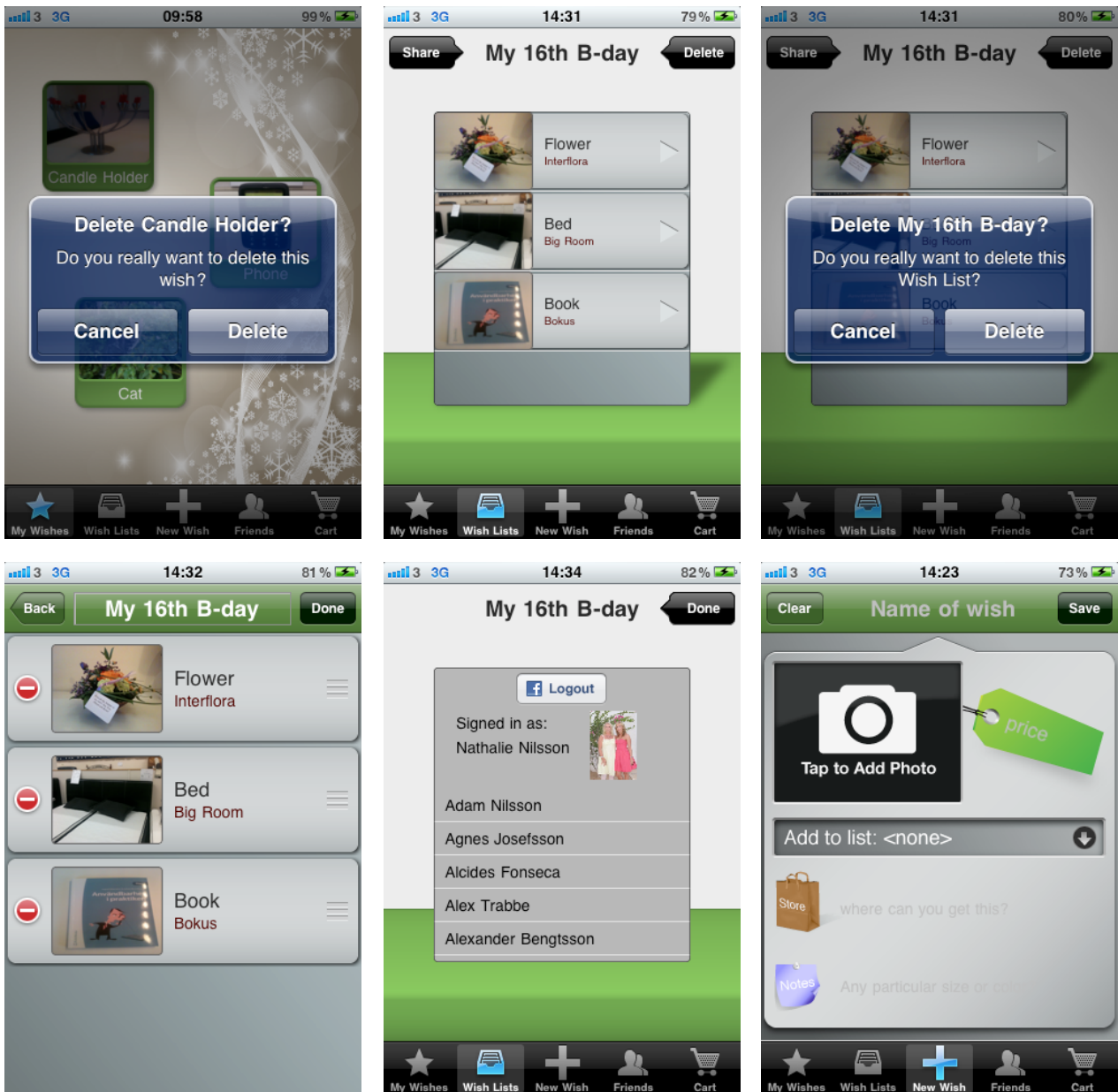


Figure 28 - Screenshots from Sprint 8.

### 6.3.5 Sprint 9

With feedback received in Sprint 8, Sprint 9 concentrated on the consistency of the application. The implemented changes and tweaks are listed below:

- Changed name of tab my wishes to wishes (Figure 29, top left)
- Miscellaneous graphic tweaks in wishes view (Figure 29, top left)
- Refined background for wish lists (Figure 29, top middle)
- Changed design of wish list table view cells (Figure 29, top middle)
- Reworked Facebook login implementation (Figure 29, top right)
- Moved Facebook login/logout button (Figure 29, top right)
- Implemented invite friends button which allows the user to send an e-mail to invite friends to download the app (Figure 29, top right)
- Redesigned new wish view (Figure 29, bottom left)
- Changed the functionality of how a new list is created (Figure 29, bottom middle)
- Changed tab bar icons (Figure 29)
- Designed a logo for WishLister (Figure 29, bottom right).



Figure 29 - Screenshots from Sprint 9.

An expert evaluation meeting was held with both supervisors from inUse and from Chalmers. The goal was to get feedback on the functionality and the design of WishLister. No user tests were performed in this Sprint, see section 6.4. The expert evaluation brought up the following issues:

- A photo taken in portrait mode should be cropped and not scaled to fit into a landscape aspect ratio
- The user should be able to see the image of a wish in a larger format (zoomed in)
- iAds can be added in more places throughout the app
- The share and delete buttons are not standard buttons and may confuse the user
- Next and previous buttons on the keyboard for jumping between input fields when editing a textfield would be nice
- Consider the possibility of allowing the user to delete a wish in the new wish view
- Miscellaneous graphical tweaks.

## **6.4 Further Development**

Following the completion of Sprint 9, it became clear that the project would not meet its deadline, which was a release of WishLister on App Store in time for Christmas 2010. The main issue was the sharing of wishes and how the transfer of data of this functionality would be handled. Having discussed the matter with both supervisors from Chalmers and inUse the decision was made that the remaining work done on WishLister, to ready it for a release sometime after Christmas, would be conducted outside the scope of this master thesis. Therefore, the results as seen in the next chapter will present the functionality of the app as it sits at the moment of writing but also present the functionality that will be implemented outside this master thesis, as the graphics and the interaction have been designed for this.

## 7 Result

This chapter contains a walkthrough of all of the functionality that WishLister has to offer. In an effort to make, what is easily shown in a live presentation or on an actual device, more readable, the functionality will be presented first as real-world scenarios, and then described, discussed and motivated with the help of screenshots. These screenshots are presented chronologically, so the reader should follow them from top to bottom and from left to right.

### 7.1 The colours of WishLister

The main colours used within WishLister are green blue and brown. Throughout WishLister there are other colours in play but the most prominent and widely used colours are these (see Figure 30). These three colours were all chosen for different reasons, apart from the fact that they work well together and are harmonious. The colour scheme was chosen in an attempt to create a neutral colour scheme in terms of its relation to holidays, where a colour scheme of white and bright red would be too much reminiscent of Christmas, for example.

- The green colour was chosen for the calming effect that green has, as well as to ride the bandwagon of environmentally-friendly concepts, where green represents sustainability and growth. It is used within WishLister as a cue for things created or editable by the user.
- The blue colour was chosen as a compliment to the green colour, to some extent because of its prominent use on Facebook. Since the app was to make use of Facebook as its social network of choice, it was natural to choose a colour to represent this part of WishLister that would be familiar in some way, if only unconsciously.
- The brown colour is used as a filler colour to the other two main colours. The colour relates to the colour of a parcel or of unbleached paper. It serves the purpose of providing an organic component to WishLister, to give the app an overall organic and human feel.



Figure 30 - The main colours used in WishLister.

### 7.2 The technical solutions of WishLister

Facebook is used all over the world and have more than 500 million active users (Lorica, 2010). More than a third of the population of Sweden uses Facebook and the number rises steadily (Bjerre, 2010). Developers can tap into this enormous potential market by using Facebook's free and readily available APIs. The APIs provides information and services, that can be accessed through third-party applications and sites, about the users of

Facebook. There exists many advantages of allowing users to log in with their Facebook accounts on a third-party implementation, some of these are:

- Increase conversion rate of users from visitors to members
- Increase the time visitors spend on the site or application
- Increase the frequency of usage
- Reliable, Facebook is a trusted brand name
- Facilitate marketing opportunities.

(Bjerre, 2010; Krohn, 2009)

By using Facebook, the user does not need to create a new account for the application and provides the user with the ability to share wish lists with Facebook friends that also have the application on their iOS device.

WishLister requires data to be transmitted between users through the application, when sharing a wish list. The reason for this is that a user's shared wish list needs to be stored in a centralised location so as to allow other users access to the information at any time. WishLister stores these shared wish lists in a database. The database, in turn communicates with WishLister via JSON and PHP.

### **7.3 Launching WishLister**

Launching WishLister is done from the Springboard, which is the iOS equivalent of a computer desktop. The WishLister icon depicts a gift that has just been opened with light emanating from it. The gift is placed on a green and brown background. The idea behind the icon is that the gift in this case is a representation of a wish from WishLister and the light emanating is the joy one feels when receiving that perfect gift. All the colours used in the icon are consistent with colours used within the app. As can be seen in Figure 30 (left) the icon stands out enough amongst other icons for the user to easily identify it.



Figure 31 - WishLister icon.

## 7.4 The five tabs of WishLister

WishLister makes use of five tabs, situated in the tab bar of the app (see Figure 31). These tabs have been consistent throughout the project and their function has been presented earlier (see section 6.1.2) but a recap may be necessary. The tabs navigate the user between the main views of WishLister.

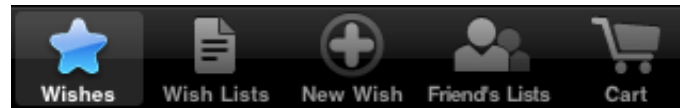


Figure 32 - The tab bar in WishLister.

- **Wishes.** This view is where a user stores wishes that have yet been organised into a wish list, before deciding what occasion (e.g. birthday or Christmas), if any, and subsequent wish list they should be added to. The idea here is that quick creation of wishes should be possible without the need for creating a wish list. The falling of a star brings with it the opportunity to make wish, as the legend goes. WishLister uses a star as the representation of a wish, which much like a falling star, is unbounded.
- **Wish Lists.** This view shows all of the wish lists created by the user and is where a user will edit and share his or her wish lists with friends. The idea behind allowing the creation of multiple wish lists stems from the fact that the user may have different wish lists for different occasions but also that the user might want to share different wish lists with different friends. The icon for the Wish Lists tab is plain and simple in that it represents a paper wish list.
- **New Wish.** This is where the user creates his or her wishes and places them, either in the Wishes tab, or in the Wish Lists tab. The act of creating a wish can be seen as adding something, which is why the icon for the New Wish tab is a plus sign.
- **Friend's Lists.** This view shows the user his or her friend's wish lists. The idea here is that the user should be able to reserve a wish, to stop anyone else from buying the same gift. Reserving a wish means that everyone else, except for the owner of the wish list, will see it being reserved and will also be unable to reserve it for themselves. The owner of a wish list can not see if a wish is reserved or not. Two busts are used as the icon for this tab. The reason for one bust being behind the other is to represent that the user can only view one wish list at a time.
- **Cart.** Wishes that the user reserves are copied to the Cart, which allows the user to have a single shopping list of gifts. The idea is to help the user remember what gifts to buy and also to provide easy access and planing for his or her gift purchases. A shopping cart is used as the icon for this tab as it represents the ease by which something can be added or removed and also that the contents is not yet acquired by the user.

## 7.5 Creating a wish

*Perhaps you remember Camille? She had a birthday bash a few pages back and used WishLister to share her birthday wish list with her friends. Now though, she is more interested in gifts relating to Christmas, which is just around the corner. Having arrived at her local IKEA store, she is now scanning the bed department in search of a new bed. You see, she has just moved into a new apartment with her boyfriend Matt and they need a bigger bed, among other things. She finds a bed that she really likes and decides to add it to her wish list. Once again, she turns to WishLister for her wishlisting needs. She creates*

a wish list, called *Christmas 2010*, and adds the bed to it. Smiling to herself she then heads off to the restaurant to eat one of those famous Swedish meatball sandwiches.

In order for Camille to complete the task of adding the bed she found, to her newly created Christmas 2010 list, there are a few steps she must complete. The steps described below can be performed in any order, and are listed here in the most common order by which they are performed. There are no functional requirements for what information needs to be input. However, a new wish must have a name, and can not have the same name as an already existing wish. This also applies when creating wish lists. In short, the only thing a user must do when creating a new wish is to give it a name.

### 7.5.1 Textfields

The new wish view provides the user with four textfields, all designated for a particular piece of information (see Figure 32). All the textfields have placeholder texts, helping the user in understanding what to fill in. At the top of the view is the textfield for the name of the wish, which is the only field the user is required to fill in. Below that is the price tag, which is represented by a yellow price tag, so as to give the user a way of remembering easily what that field is for. At the bottom are the store and notes input fields. The store field is meant to be populated with information about where the wish can be bought and the paper bag is a representation of a shopping bag that one might receive when buying an item. The notes field is supposed to be used for more miscellaneous information, such as garment size or preferred colour. The icon for it is a stack of post-it notes representing the randomness of the information that can be provided for this field.

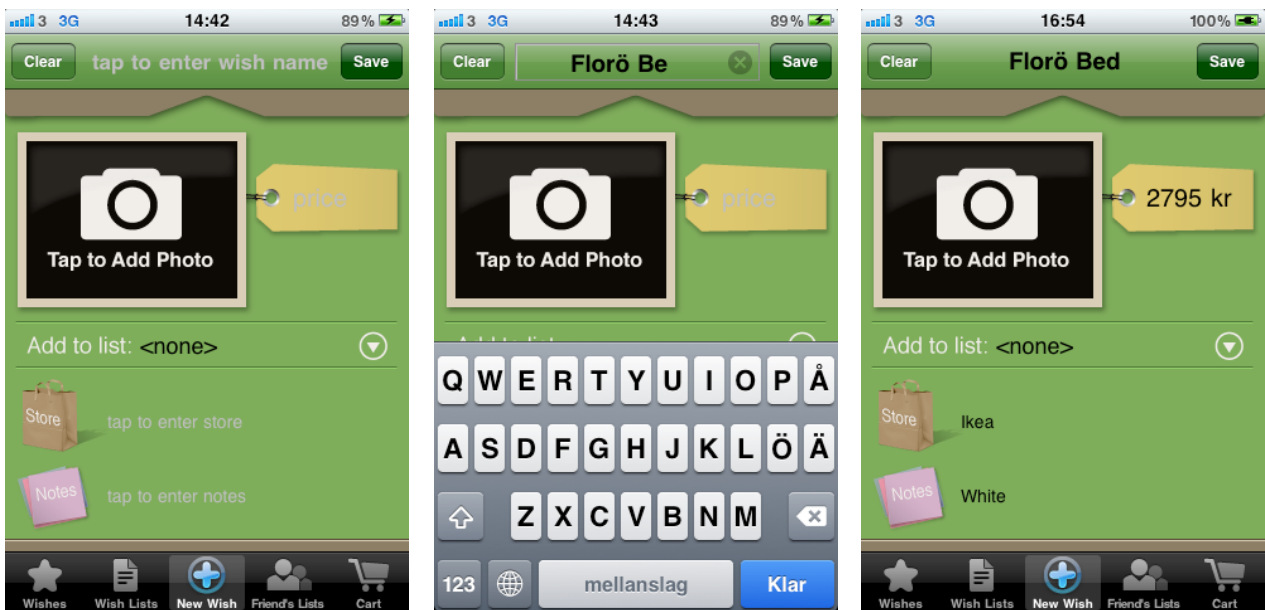


Figure 33 - Entering text in textfields.

### 7.5.2 Adding a photo

To help the user remember and also to provide the user's friends with more detailed information, a photo can be added to a wish. The black square situated near the middle of the view (see Figure 33) provides a deep contrast to the bright surroundings meaning it will not go unnoticed. The reason for this is that photos are seen as extremely important when it comes to providing detail information (a picture says more than a 1000 words), so the app encourages the adding of a photo and provides an easy way for the user to do so. A photo can be added by either choosing an existing photo from the user's iPhone photo library or by taking a new photo, should the device sport a built-in camera.

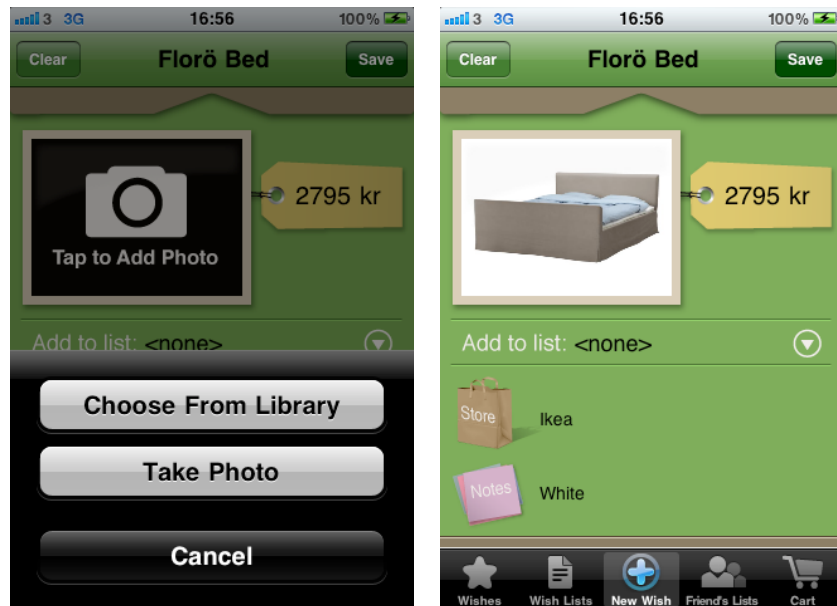


Figure 34 - Adding a photo.

### 7.5.3 Adding to wish list

Last but not least, the user has a choice of adding the wish to an existing wish list, to a new wish list or to no wish list (<none>, which means it will be placed in the Wishes view). The default here is that the wish will be created in the Wishes view. The reason being that the user should not have to make a choice, and also since the Wishes view is supposed to work almost like temporary storage, it felt necessary to allow the user to create a wish quickly and on the go, and not be boggled down with unnecessary choices. As can be seen in Figure 34 the user can select an existing wish list from the list or tap on the textfield to create a new wish list. The button on the right hand side will reflect if the user has selected an already existing wish list (Done) or if the user is creating a new wish list (Create). The name of the list is then shown in the main view.



Figure 35 - Selecting or creating a wish list.

## 7.6 Wishes

While munching on a sandwich, that was far from what it was hyped up to be, Camille looks through her wishes. For the past couple of weeks she has added wishes to WishLister as she has come across them, at shops, at friend's apartments or online. As she peruses, and moves them around the screen, she realises that she actually bought a pink sweatshirt a few days ago. Since her wish has come true she decides to delete the pink sweatshirt from WishLister. She also goes through the rest of the wishes and decides what to do with them. A couple of them she adds to her Christmas 2010 list, along with the bed she added just moments before, and others are deleted. As the last meatball finds its way into her mouth, she chews it precariously as she phones up Matt. Matt has been working all day and is just about ready to go home. Camille says she will meet up with him at the train station in 20 minutes. Reluctantly, as it only takes him three minutes and 16 seconds (he measured it once, for kicks), Matt agrees to wait for her.

The less than satisfying sandwich aside, Camille used all the functionality available to her in the Wishes tab, to achieve her desired outcome. The Wishes view, as has been noted earlier, is used as a temporary storage of random wishes that might not fit into a specific list. For example, the user could use a wish as a reminder for what wine he or she was supposed to bring to the dinner party. The view itself is, for all intents and purposes, a blank canvas which serves as a desktop of sorts for wish tabs (see Figure 35).



Figure 36 - Empty Wishes view.

### 7.6.1 Moving and tapping wish pads

The icon-like square, depicting the name and photo of the wish, is known as a wish pad (See Figure 36). These pads can be moved about in the view freely, with the idea being to support the notion that they are placed here at a whim and are not organised in any way. As the user interacts with a pad, its colour will fade to give visual feedback, to the user, that the pad is being acted upon. Tapping the pad will in turn navigate the user into a detail view of the wish. This view is the same as the new wish view with the exception that the tab bar has been hidden. Instead, there is an iAd placeholder. The reason behind hiding the tab bar was not only to provide room for the iAd but also to limit the interaction possible to the user. A limitation might not always be a good thing but it can help a system be more robust by making sure that the user can not perform certain intricate actions. Inside the

detail view of a wish, editing takes place by virtue of tapping on the desired field, just like when creating a wish. As the view works the same as the new wish view, it gives the user a sense of familiarity while also giving the user an easy any way for making changes.



Figure 37 - Moving and tapping a wish pad.

### 7.6.2 Deleting a wish pad

In the scenario, Camille decides to delete the pink sweatshirt, as she realises that she has already bought one. She does this by pressing and holding on the sweatshirt pad. By pressing and holding a dialogue box will appear, asking the user if he or she really wants to delete the wish. Pressing cancel means the user will be navigated back to the main view with no changes made. Pressing delete means the user will be navigated back to the main view, but with the sweatshirt no longer present. As far as the wish pads go, deleting, opening and moving are the ways in which the user will interact with them. However, there is one more action that will affect this view, namely moving a wish into a wish list.

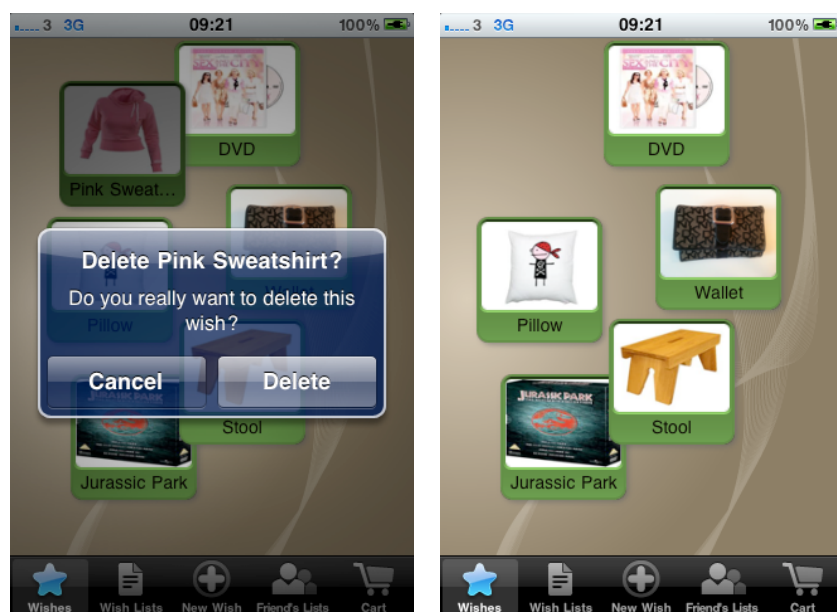


Figure 38 - Deleting a wish pad.

### 7.6.3 Moving a wish into a wish list

Camille moves some of her wishes into a wish list. She does this by opening the wish and pressing the 'Add to list:' field in the detail wish view (see Figure 38). In doing so she is given a choice to move the wallet into one of her existing wish lists or to create a new one. She chooses to move the wallet into her Christmas 2010 list. Having chosen the list and tapping save, the user is navigated back to the Wishes tab, and can see that the wallet pad is gone. The reason for navigating the user back to the main view, instead of navigating the user to the wish list, is to show him or her that the pad has been moved, by virtue of the fact that it is no longer present in the view. This is done to avoid any confusion about whether a wish is copied or just transferred when moving it to a list.



Figure 39 - Moving a wish into a wish list.

## 7.7 The Wish Lists

As Camille arrives at the train station, she finds Matt sitting on a bench outside basking in the sun. Blocking his sun until he sees a need for opening his eyes, she gives him a hug and kiss before they venture inside the station towards their train. Luckily, the train is on time and leaves in 10 minutes. As they take a seat on the train, Matt remarks that he threw out the orchid, that Camille got for her birthday, as it looked dead. Camille, having more knowledge about orchids than Matt, scolds him for throwing it out as it was not dead at all, even though it might have appeared that way. Pouting and with her arms crossed, Camille sulks while Matt apologises. He remarks that maybe she could put a new orchid on her Christmas list, to make up for the one she just lost. Camille, still sulking, opens up WishLister and navigates to the list she created for her birthday. On that list is the original orchid she wanted, and got, for her birthday. She moves the orchid to her Christmas list and decides to delete the old birthday list. As she opens her Christmas list, Matt notices the bed she added before and asks about it. Camille tells him that she found it at IKEA and that it would fit perfectly in their new bedroom. Matt disagrees, and following a heated argument and more sulking Camille agrees to delete it from the list. She also moves the orchid to the top of the list and renames it from Christmas 2010, to simply Christmas. She decides to share the list with her friends as it is only a few weeks until Christmas. Alas, the train hits the brakes before she can share the wish list and she and Matt exit the train and start walking home, hand in hand.

### 7.7.1 Navigating wish lists

The Wish Lists view is by far the most complex view, in terms of functionality. It allows the user to browse, delete, reorganise and share his or her wish lists. The view is centred around the concept that the user's wish lists are shown as small thumbnails, showing off the content of the wish list but not allowing for any interaction to take place between the wish list and the user. The user may scroll between his or her lists by swiping their finger across the screen (see Figure 39). The view will scroll through the wish lists much like pages, where the user will swipe his or her finger across the screen which will shift the screen across to the next wish list. At the bottom of the view there are a number of small dots. These dots represent the current page (or wish list) that the user is currently viewing and also show how many wish lists the user has created, helping the user keep track of this, without having to scroll through and count all the wish lists.



Figure 40 - Wish Lists navigation.

### 7.7.2 Viewing a wish list

By tapping a finger on a wish list, the user will, in effect, open the list whereupon it will grow to fill the entire screen and also allow the user to interact with it, just as he or she would any standard table view (see Figure 40). Pressing the back button on the navigation bar will shrink the view down again to its original size. To navigate within the table view a user can scroll up and down and also tap each cell to open that specific wish up in a detail view. This view is the same as the one users are presented with when tapping a wish pad in the Wishes view. Any modifications to the wish within this view will be reflected on the table view at the moment the user taps save. Should the user move the wish to another wish list, the wish will disappear from the list the user is currently viewing and be added to the wish list the user added the wish to.

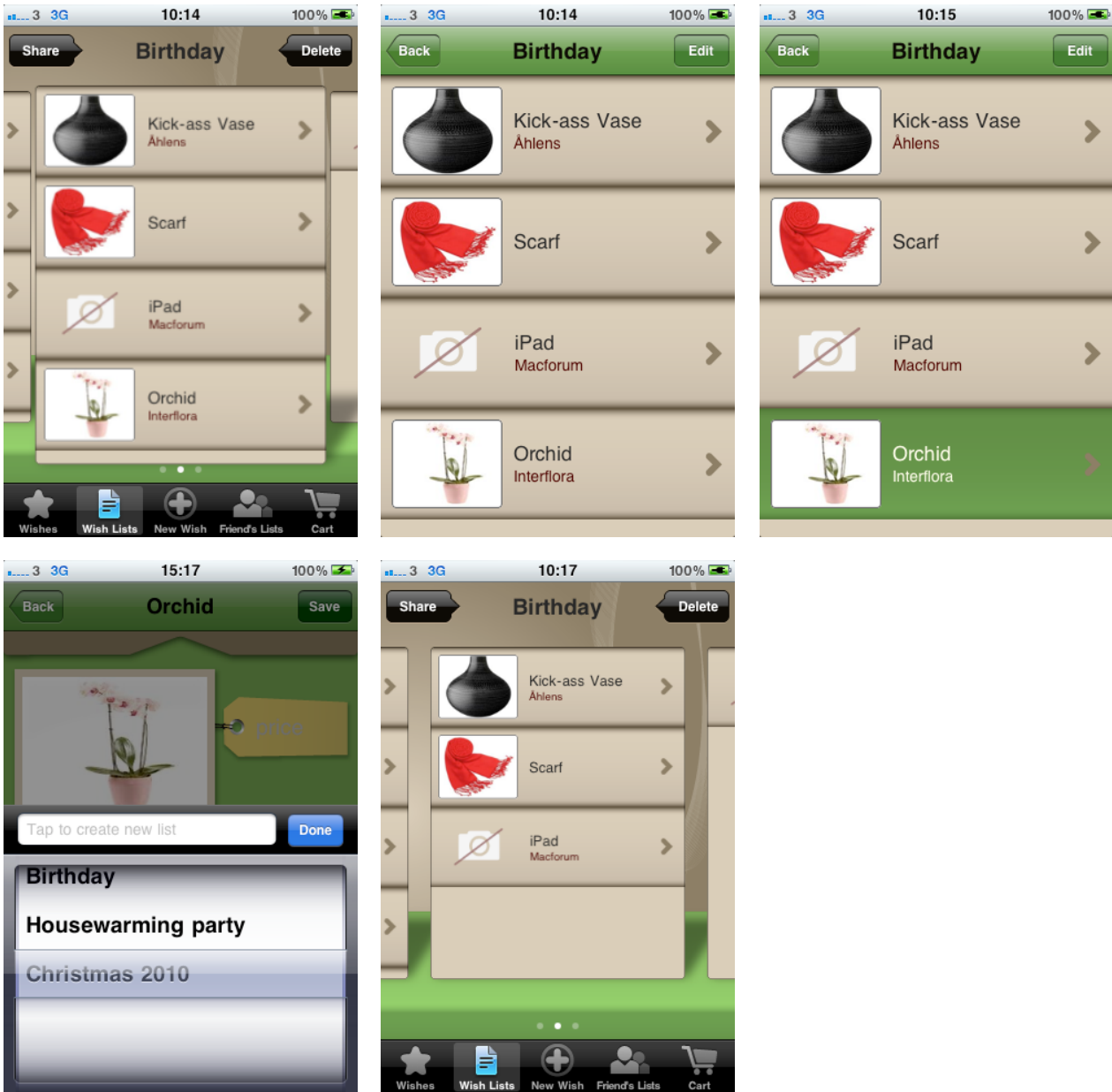


Figure 41 - Viewing a wish list.

### 7.7.3 Deleting a wish list

Should a user want to delete an entire wish list, he or she can easily do so by tapping the delete button next to the wish list name. A dialogue box will then appear asking for confirmation, whereupon the list will be removed from the view (see Figure 41).

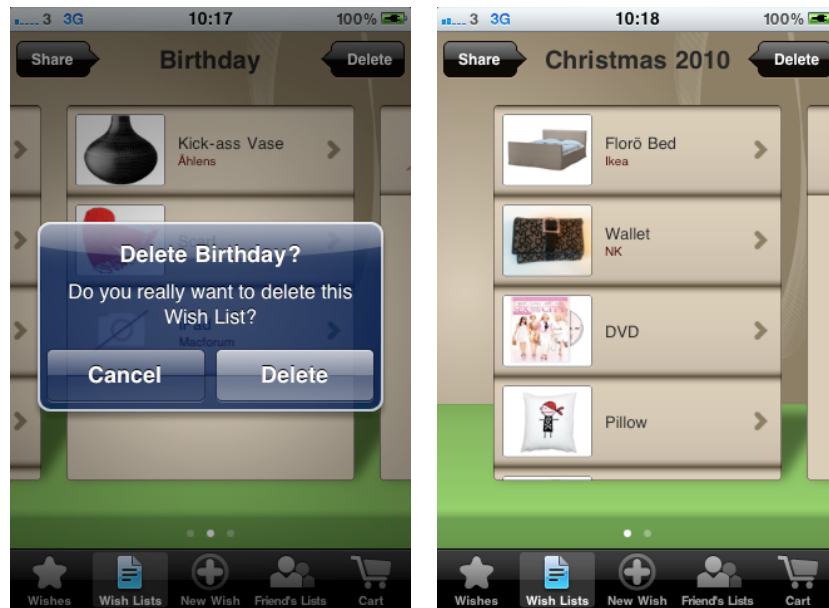


Figure 42 - Deleting a wish list.

### 7.7.4 Organising a wish list

It is also possible for a user to reorganise, rename or delete a single wish from within a wish list. This is done by pressing the edit button (see Figure 40, top right). Having tapped on the edit button the user can reorganise the wishes by dragging them around. Delete wishes, by pressing the red button at the far left of the cell and then pressing the delete button. The user can also rename the wish list by pressing the textfield at the top of the view. See Figure 42.

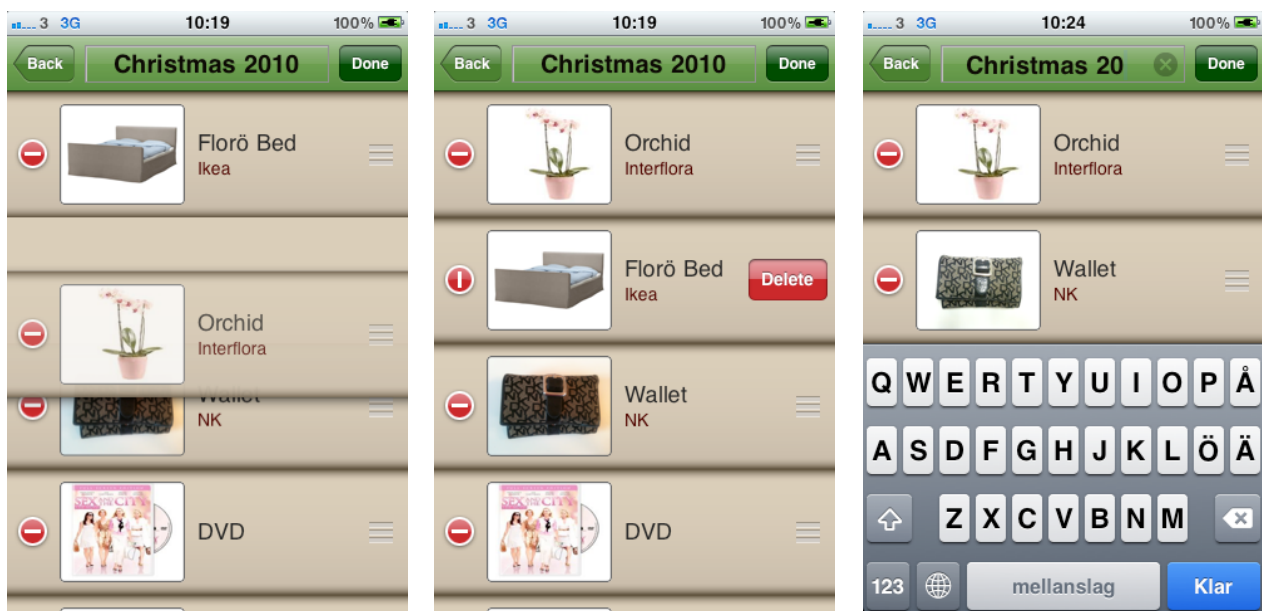


Figure 43 - Organising a wish list.

## 7.8 Sharing a Wish List

*Well at home, and while Matt takes care of dinner, Camille tries again to share her Christmas Wish List with her friends. Having logged in to Facebook, she shares her list with her friend Christofer, and also considers inviting her brother Bo to download the app. She shouts a question in Matt's general direction, if he thinks Bo would be interested in*

using an app such as WishLister. Matt, in middle of a grand culinary experiment, shouts back that Bo, as an early adopter to anything iPhone related, would probably want to give WishLister a go. Camille sends off an e-mail from within WishLister, recommending Bo to download WishLister so that she can share her Christmas wish with him. Following a minor incident in the kitchen, Matt comes into the living room coughing and remarks that maybe the two of them should order takeout instead.

### 7.8.1 Share view

Sharing a Wish List in WishLister is a simple process. Pressing the share button next to a Wish List will spin the wish list around 180 degrees (see Figure 43). This helps the user understand that the share view presented at the back of the wish list belongs to that wish list. Having spun around the user is unable to scroll the view and all other wish lists are hidden to minimise distractions.



Figure 44 - Share view.

### 7.8.2 Sharing a wish list

Having pressed the share button the user is presented with the share view, where the share button has been replaced by a Facebook login button which is easily recognisable and a standard component in any Facebook-enabled app (see Figure 44). Upon pressing the button the user is navigated to the Facebook app where login is handled automatically, alternatively, if the user does not have the Facebook app installed, he or she is navigated to Safari where they can log in manually. Having logged in, the user is brought back into WishLister. The login button has now become a log out button, the user's profile picture is seen in the top left of the view and all the user's Facebook friends, who have also logged in through the app are listed in a table view. It should be pointed out that having a name on the list only means that the person has logged in to Facebook from the app at one time or other. It does not necessarily mean that the user still has the app installed or is using it. This is a limitation of Facebook and not something that is easily modified. Clicking on a name in the table view will place a checkmark next to the name, signifying that this person can access the Wish List on their own device.

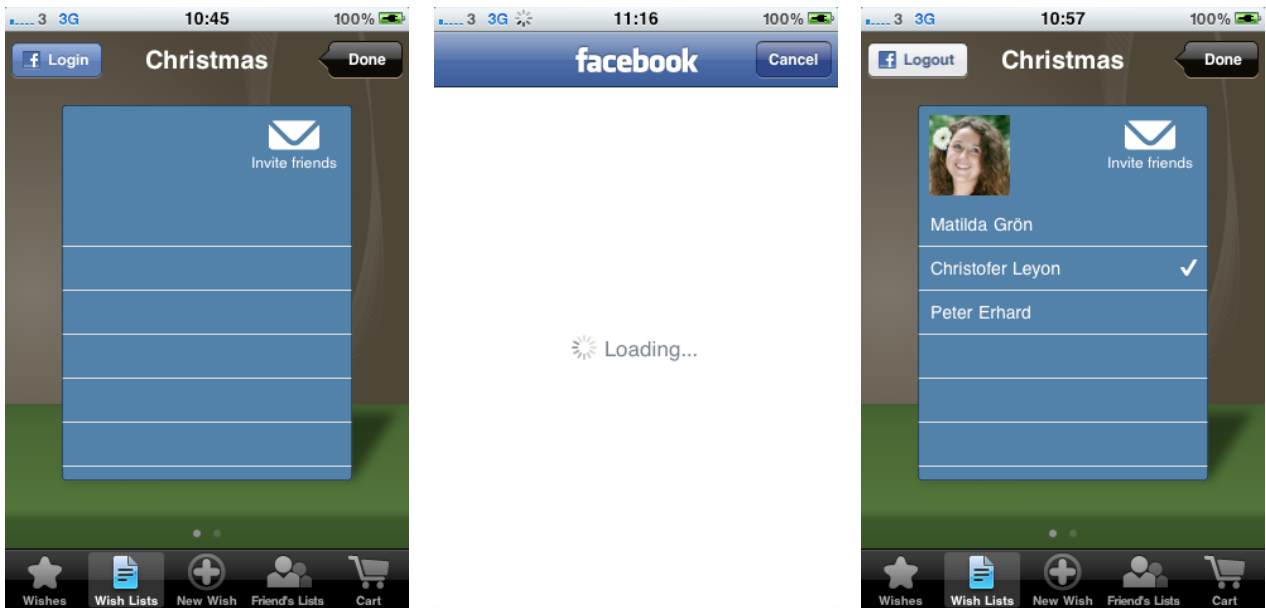


Figure 45 - Sharing a Wish List screenshots.

### 7.8.3 Inviting friends

There is also a simple way to invite friends to download WishLister for their own devices, namely through the invite friends button (see Figure 45). Tapping this button will bring up a form e-mail which the user can either edit, if they want to make it more personal or just send it as is. The mail contains a link to the WishLister app on App Store, making it easy for potential users to download the app. Tapping the done button after having completed these tasks, will spin the wish list back around 180 degrees and allow the user to once again navigate his or her wish lists.

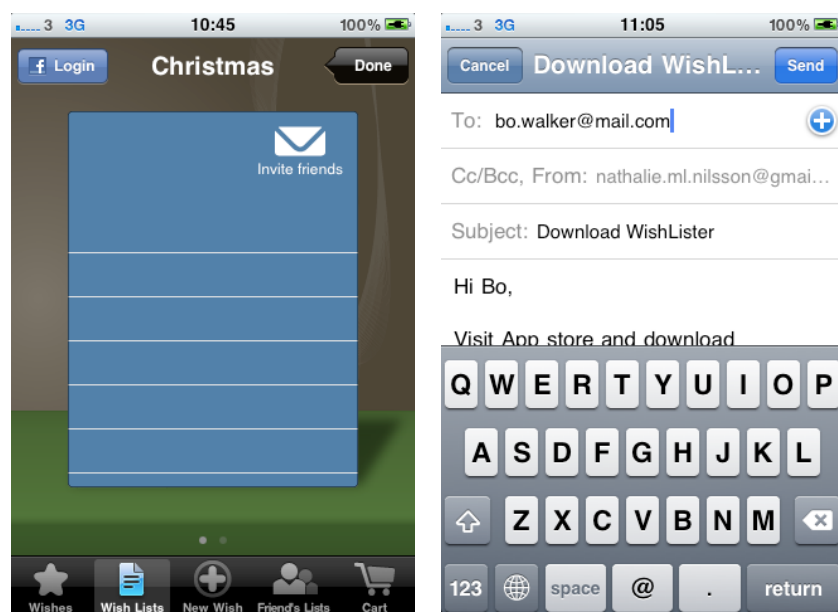


Figure 46 - Inviting friends.

## 7.9 Working with a friend's Wish List

*Bo Walker eased himself into his comfy chair. Everything was set, the snacks were on the table, along with the drinks, His best friends were there, with their respective partners, the kids were sleeping and throughout the house there was anticipation in the air. His wife lunged herself into his lap, almost causing him to loose his breath as she caught him off*

guard. They were gathered to watch the final *Lost* episode, something that Bo had looked forward to for a long time, mostly for the closure he hoped it would bring. His iPhone buzzed in his pocket. Contemplating whether to leave it or check it, Bo decided to ignore it for now. As the credits rolled and his friends made their way home, Bo remembered his buzzing iPhone and picked it up out of his pocket. He had received an e-mail from his sister Camille. Apparently, she wanted him to download some app or other, so that she could share her Christmas wish list with him. Not being a stranger to new and potentially exciting prospects, Bo downloaded WishLister off App Store and launched it. Immediately he noticed a small red badge at the bottom of the screen. He tapped it and was then asked to accept Camille's invitation to see her wish list, and he did. He chuckled to himself at the sight of the orchid on the list and thought that plants were never Camille's strong suit and she probably killed the orchid he gave her for her birthday. Apparently, someone had already bought a DVD for Camille so that left Bo with the choice of buying either another orchid, a wallet, a pillow or a stool. He decided to get her the wallet and reserved it. A little red badge with a '1' on it popped up at the bottom of the screen. Curious as to what this new badge meant he pressed the button next to it.

### 7.9.1 Viewing friend's wish lists

As a user shares a Wish List with friends, these friends will notice a small badge next to the Friend's Lists icon, in the tab bar, showing that invites are awaiting confirmation (see Figure 46). Navigating to the Friend's Lists view will bring up a dialogue box asking the user to either confirm or ignore the invitation from another user. Should the user ignore the invitation, it will be brought up again the next time he or she launches the app, but not before. Tapping accept means that the friend's wish list will be added to the view. The Friend's Lists view works in more or less the same way as the Wish Lists view, where the major difference is the fact that the user can not modify another user's wish list. The view replaces the green colour of the Wish Lists tab with a blue colour signifying that the user is now looking at friend's lists and not their own. This is important, as the views are very similar and their functionality almost identical.

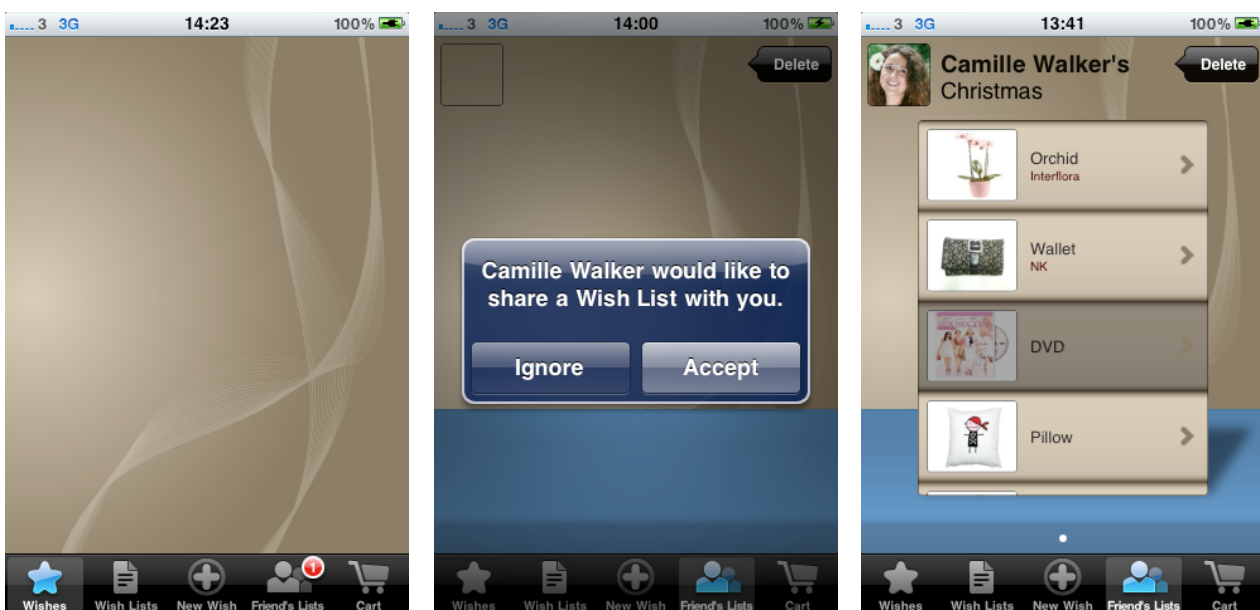


Figure 47 - Receiving an invitation.

### 7.9.2 Reserving a wish

A wish can only be reserved by one person, however anyone can view the wish, even if they can not reserve it. Wishes reserved by the user have a blue glow around them, while

wishes reserved by other people show up with a grey overlay, in the friend's wish list. Any wishes that have not been reserved by the user or by other people show up in the familiar golden brown colour. Opening a wish will present the user with the, by now, familiar detail wish view, albeit in blue. The user is able to look at the view only and can not interact with it in any way except for the tapping of the reserve button at the top right corner of the view. Tapping the button will reserve the wish, changing the button from a reserve button to an unreserve button, while also displaying a badge next to the Cart tab, signifying that the wish has been copied to the Cart. Note that the interaction between a user's list and his or her friends is completely unknown to the user themselves. That is, as the user invites people to view his or her wish list, any reservations made by the user's friends will not be visible to the user. Knowing what everyone was buying you for your birthday would not make for a good selling point for the app.



Figure 48 - Reserving and unreserving a wish.

## 7.10 Using the Cart

*As Bo taps on the Cart icon in the tab bar he is presented with a view, showing the wallet he just picked out for Camille. He taps on the wallet and quickly realises that as he reserves more wishes the Cart is going to let him keep track of what items he has decided to buy for his loved ones. Bo sends off a quick e-mail to Camille thanking her for showing him the app, as he knows it will come extremely handy. On Christmas day, Camille opens up her gift from Bo, finding the wallet she wanted inside layer after layer of wrapping paper. She jumps up and hugs Bo, all the while thanking him for the beautiful gift. Bo jokes that he should have bought the whole family iPhones so that they can all use WishLister, now that he is hooked. As the snow falls gently outside, the laughs of the Walker family fill the air.*

The idea behind the Cart is that, as the user comes to reserve a couple of wishes, there is a need for keeping these wishes organised somewhere. Otherwise the user would have to navigate to each friend's list and look at what he or she did reserve, if they should forget. The Cart consolidates all the user's reserved wishes into one view, along with supplying the name of the friend, for whom the wish is intended. As the user goes out to make his or her gift purchases, he or she can easily keep track of what they were supposed to buy. As far as functionality goes, the Cart is just like every other table view in WishLister. The blue colour scheme shows that these are friend's wishes that are listed. A

user can tap on the wish to get more information, should they forget where they were suppose to buy it, delete the wish or reorganise the wishes in wish list in the same fashion as in his or her own wish lists. The user can remove a wish from the cart in two ways, namely by tapping unreserve in the detail wish view, or by pressing edit and deleting the wish from the list. They both serve the same outcome, which is to remove the wish from the cart which also unreserves it. Unreserving a wish means it is once again available for other friends to reserve.



Figure 49 - Using the cart.

## **8 Discussion**

The discussion will touch on some the major points concerning why the deadline could not be met, the overall achievements of the project and also a brief contemplation on the work process and its highs and lows. Possible future work for WishLister will also be discussed.

### **8.1 Status of WishLister**

The status of WishLister, as of the writing of this report, is that functionality concerning the sharing of wishes, namely the interaction between the app and the database set up to hold the share wish lists, has not been implemented. The functionality, as seen by the screenshots in chapter 7, exists in terms of design, only lacking some execution for it to be ready for release. This functionality proved to take longer than expected to implement and was therefore not completed in time for the deadline of the project. As such, no user testing was conducted on this critical part of the app, except for the tests conducted with paper prototypes early on in the development process. The functionality concerning sharing wish lists is similar in many respects to functionality that does exist in WishLister as it sits today. However, even though the authors do draw some conclusions about the usability and ease of use of the design, they would like to point out that these are no substitutes for conducting user testing on a finished product.

### **8.2 WishLister's Aim and Measurement points**

In the developed Effect Map (see appendix H) the WishLister's aim was defined as: "*The obvious choice for planning and organising your own wishes and the wish of others*". To find out if the aim was achieved or not, three measurement points were defined. Since the application will be completed outside the scope of this master thesis, it is not possible to measure if the aim has been met using the measurement points, as such it can said that it was not reached at the time this report was written. However, the measurement points will be evaluated as soon as the app is finished and released on the App Store.

### **8.3 Usability and Interaction Design**

Despite the fact that WishLister never reached release-ready status, the authors are confident that the parts of the app that were developed in full, have a high level of usability and good interaction design. The agile methodology used throughout the project, which made use of user tests and expert evaluations in every Sprint, resulted in 35 user tests and 8 expert evaluations being performed during the development process. The performing of all these tests have made sure that the app is virtually bug-free, as it sits today, and also that the interaction with the app is a smooth and intuitive experience.

The bulk of the testing of WishLister was done on iPhone owners. It can be argued that, to get a fair picture of usability problems, more testing should have been done on potential users that do not own an iPhone. However, as the app is not set to persuade any potential user of WishLister into buying an iPhone, and even if it did that undertaking would seem somewhat futile, it was thought useful only to consider users who already owned an iPhone and for whom the usage of the app would be as simple as downloading it from App Store.

In terms of experts, most of them are employes at inUse and therefore share a common framework. Experts from a number of different usability, and interaction design businesses, may have given a broader, more nuanced view, however, finding and recruiting such experts was deemed to be too time consuming, as to provide a good ratio between effort and reward.

## **8.4 Future work**

Before any future work can be considered for WishLister, the app needs to be finished and be approved by inUse as well as Apple. When WishLister is released on App Store, the project team can then start working on future updates.

One idea that did not make it into the app was that of bidding on a wish. The idea was that WishLister could facilitate the somewhat strenuous task of buying a gift together with someone else. This is quite often the case at bigger parties, or weddings when guests want to buy something grand that may not be within their budget. The solution is to get together and pay a little each. WishLister would make this process easy, by allowing a user, not only to reserve, but also to bid on a wish. Essentially, the user would state how much money he or she wanted to put in towards reaching the price, and other users would be able to do the same. When the price was met, it would be easy to organise the acquiring of the gift at hand.

The idea is that WishLister could be available in multiple languages. The first app will be released in English as well as Swedish. A future goal is to translate the app into other languages as well, for example in Japanese.

Another feature considered for future implementation is to use the built-in GPS functionality of an iOS device, for locating wishes geographically. In other words, the user would specify exactly where the wish can be bought, which can then be accessed by the user's friends, making it even easier for them to find it.

Since WishLister integrates with Facebook, it is quite natural that the development of the product should make more use of this integration. One such idea would be the creation of a web-based Facebook app, that would work essentially the same way as its iOS counterpart, with the exception of not requiring an iOS device to use. This would increase the number of potential users dramatically, and also provide better overall availability for existing users.

PriceRunner is a web-service where it is possible to compare prices for a product in different stores. An integration with PriceRunner and WishLister would help users to find the best price for a specific wish.

The concept of WishLister could be sold as a generic technology that could be branded by a company, to serve as a means for marketing oneself or one's products. Big department stores could, for example, keep a branded version of the app on the App Store that would provide the user with special offers or discounts when they visit the store as well as provide local information about the store.

## **8.5 Work Process**

The work process, outlined at the beginning of the project, served to provide the authors with a good foundation from where to argue for their decisions. It was not without its problems however, as the result and lack of a finished product clearly shows. The following paragraphs discuss some of the main points of interest in discussing the work process used in this project.

In the initial time schedule, four Sprints were planned, where each Sprint contained one week of design, one week of development and one week of evaluation. It turned out in the end, however, that this time schedule did not fit the project's work process. Each sprint

proved to be unique in its own way and it was hard to determine exactly how much time the different parts of each Sprint would take. The first Sprints took longer, due to lack of experience with the iOS platform as a lot of time was spent learning the language, compared to the later Sprints. It also came to light that a lot more time was needed to be spent on developing, as opposed to design and evaluation, since this proved to be the most difficult part for the authors.

Most of the (around 150) concepts conceived in the concept generation phase, somehow related to a collaboration with a company or business. In the end, it turned out that it was difficult and time consuming to try and pitch the author's proposals to businesses during the summer period, since most people had already gone on vacation and as such interested in a collaboration in September at the earliest, which was too late to fit this project.

Starting the project, the authors had no experience of developing for the iOS platform. As it were, actually starting to develop WishLister, was a slow and process filled with questions with dubious answers given, on internet forums and articles. However, nearing the end of the project the authors felt confident that the development of WishLister was possible for them, however, not feasible with the time left before the project deadline. If the development process was restarted today, the result that came out of the four months of development could probably be achieved in less than two months time. A testament to how much the authors have learned.

To this end, it can be questioned if the project started out on ambitions and goals set way too high. That would not be an unfounded assumption. Many of the decisions concerning functionality were based on experience using iOS devices, but not developing for them. In retrospect, a more structured development process may have been able to recognise that the project was falling behind earlier, allowing the authors to reevaluate the concept at what it should include.

In the beginning of the project a risk analysis was performed. As it turns some of these risks did occur during the project. However, even if the risks were identified, in terms of probability and impact, along with countermeasures, it still proved difficult for the authors to manage these risks. One problem was to discover if a risk had actually occurred, as some were quite subtle, in time to counteract the impact the risk had on the project.

## 9 Conclusions

Relating to the aims and goals defined at the beginning of the report, the following chapter will present the aims and goals that have and have not been met in the project.

One goal of the project was *"[...] to combine methods and knowledge gather from studies at Chalmers University of Technology with the work flow and process used at inUse."*

This goal can be seen as achieved. Throughout the project 17 different methods along with inUse's Effect Map were combined into a unified work process. The knowledge gathered from Chalmers combined with inUse's own method have given the authors a broader view of how methods can be combined and be used together in a product development environment. That the methods have also been used in a real situation project has been very instructive and rewarding for the authors.

Another goal was to *"[...] incorporates inUse's strong focus on usability and interaction design."*

By making use of inUse method of Effect Mapping, along with a number of user tests and expert evaluation, the project definitely incorporated inUse's strong focus on usability and interaction design into the entire work process of the project.

Another goal was *"[...] to release the app on Apple's App Store."*

The only goal that was not reached in the end, was the release of the app on Apple's App Store. However, as this was a goal that was tightly linked to inUse's goal for the project and since the app will be released, only outside the scope of this project, the goal will be accomplished, ultimately. However, as time and time management is an important aspect in product development, the authors maintain that the goal was not reached inside the scope of the project.

An internal goal amongst the authors was to learn to develop for the iOS platform, by learning the language and usage of the tools. As the result clearly shows, the authors did achieve this goal in a very profound way.

## References

- AdMob, 2010. *AdMob Mobile Metrics Report*. [online] Available at: <metrics.admob.com/wp-content/uploads/2010/02/AdMob-Mobile-Metrics-Jan-10.pdf> [Accessed 2010-07-20].
- Agile Methodology, 2008. *Agile Methodology*. [online] Available at: <agilemethodology.org> [Accessed 2010-06-10].
- Apple, 2007. *iPhone Premieres This Friday Night at Apple Retail Stores*. [online] Available at: <www.apple.com/pr/library/2007/06/28iphone.html> [Accessed 2010-11-24].
- Apple, 2010a. *Apple Special Event*. [online] Available at: <events.apple.com.edgesuite.net/1010qwoeiuryfg/event/index.html> [Accessed 2010-11-10].
- Apple, 2010b. *iOS Human Interface Guidelines*. [online] Available at: <developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html> [Accessed 2010-07-07].
- Apple, 2010c. *App Store Review Guidelines*. [online] Available at: <developer.apple.com/appstore/resources/approval/guidelines.html> [Accessed 2010-06-29].
- Armstrong, G. Kotler, P. Harker, M. and Brennan, R., 2009. *Marketing: An Introduction*. Italy: Pearson Education Limited.
- Beck, K., 2000. *Extreme Programming Explained: Embrace Change*. Boston, MA.: Addison-Wesley.
- Bjerre, L., 2010. *Dra nytta av Facebook Connect*. [online] Available at: <internetworld.idg.se/2.1006/1.326046/dra-nyttan-av-facebook-connect> [Accessed 2010-07-02].
- Berndtsson, J. and Ottersten, I., 2002. *Användbarhet i praktiken*. Lund: Studentlitteratur.
- Carroll, J., 1995. *Scenario-Based Design*. Hoboken, NJ.: John Wiley & Sons.
- Clark, J., 2010. *Tapworthy: Designing Great iPhone Apps*. Sebastapol, CA.: O'Reilly Books.
- Engelbrektsson, P., 2010. *Question based methods*. (Personal communication 2010-01-26).
- Eriksson, U., 2007. *Kravhantering för IT-system*. Malmö: Studentlitteratur.
- Gibbs, A., 1997. *Focus Groups: University of Surrey: Social research Update Issue #19*. [online] Available at: <sru.soc.surrey.ac.uk/index.html> [Accessed 2010-06-14].
- Ginsburg, S., 2010. *Designing the iPhone User Experience*. Boston, MA.: Addison-Wesley.

Göransson, O. and Landin, K., 2009. *M.I.G for iPhone and iPod Touch - A Physical Quiz Game Ported to an Application Designed for Attention*.

Hunt, A., 2008. *Pragmatic Thinking and Learning: Refactor Your Wetware*. Illustrated ed. Raleigh: The Pragmatic Programmers.

IDEO, 2003. *Method Cards: 51 ways to inspire design*. San Francisco, CA.: William Stout.

inUse, 2010. *Om oss*. [online] Available at: <[www.inuse.se/om](http://www.inuse.se/om)> [Accessed 2010-11-24].

Jonas, S., 2010. *Jävla skitsystem!: Hur en digital arbetsmiljö stressar oss på jobbet - och hur vi kan ta tillbaka kontrollen*. Stockholm: Jonas Söderström.

Karlsson, M. Kaulio, M. Hampf, J. and Sperling, L., 1998. *Eliciting customer requirements: product representations as mediating objects in focus group interviews*. In: *EIASM, 5th International Product Development Management Conference*. Como, Italy, June 1998.

Kellogg, D., 2010. iPhone vs. Android. *Nielsen Wire*, [blog] 4 June. Available at: <[blog.nielsen.com/nielsenwire/online\\_mobile/iphone-vs-android/](http://blog.nielsen.com/nielsenwire/online_mobile/iphone-vs-android/)> [Accessed 2010-11-23].

Krohn, A., 2009. *Facebook Connect, del 1 - fördelar och nackdelar*. [online] Available at: <[www.mashup.se/api/facebook-connect-del-1-fordelar-och-nackdelar](http://www.mashup.se/api/facebook-connect-del-1-fordelar-och-nackdelar)> [Accessed 2010-12-03].

Landqvist, J., 2001. *Vilda idéer och djuplodande analys om designmetodikens grunder*. 2nd ed. Stockholm: Carlssons Bokförlag.

Lichtenberg, S, 2005. *How to avoid overruns and delays successfully: Nine basic rules and an associated operable procedure*. [online] Available at: <[www.lichtenberg.org](http://www.lichtenberg.org)> [Accessed 2010-06-10].

Lorica, B., 2010. *Where Facebook's half a billion users reside*. [online] Available at: <[radar.oreilly.com/2010/07/facebook-reaches-half-a-billion.html](http://radar.oreilly.com/2010/07/facebook-reaches-half-a-billion.html)> [Accessed 2010-07-02].

Nielsen, J, 2000. *Alertbox: Why you only need to test with 5 users*. [online] Available at: <[www.useit.com/alertbox/20000319.html](http://www.useit.com/alertbox/20000319.html)> [Accessed 2010-06-14].

Ottersten, I. and Balic, M., 2007. *Effect Managing IT*. Stockholm: Liber AB

Rebernik, M and Bradac, B, 2010. *Module 4: Idea Evaluation methods and techniques*. [online] Available at: <[www.creative-trainer.eu/index.php?id=download&L=1](http://www.creative-trainer.eu/index.php?id=download&L=1)> [Accessed 2010-06-14].

Rosson, M. and Carroll, J., 2001. *Usability Engineering*. San Francisco, CA: Morgan Kaufmann Publishers

Sadun, E., 2009. *The iPhone Developer's Cookbook: Building Applications with the iPhone SDK*. Boston, MA.: Addison-Wesley.

Schimdt, W.V. and Conaway, R.N., 1998. *Results-Oriented Interviewing: Principles, Practice and Procedures*. Boston, MA.: Allyn & Bacon, Inc.

Smashing Media GmbH, 2009. *The Smashing Book*. Germany: DE Druck Europa GmbH

Straker, D and Rawlinson, G, 2003. *Random Words*. [online] Available at: <[www.creatingminds.org/tools/random\\_words.htm](http://www.creatingminds.org/tools/random_words.htm)> [Accessed 2010-06-28].

Sundström, T., 2005. *Användbarhetsboken*. Lund: Studentlitteratur AB.

Sutherland, J. and Schwaber, K., 2007. *The Scrum Papers: Nuts, Bolts and Orgins of an Agile Process*. [online] Available at: <[citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.814&rep=rep1&type=pdf](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.814&rep=rep1&type=pdf)> [Accessed 2010-05-28].

Sy, D., 2007. Adapting Usability Investigations for Agile User-Centered Design. *Journal of Usability Studies*, 2(3), pp.112-132.

Tyco Electronics, 2010. *History of Elo*. [online] Available at: <[www.elotouch.com/AboutElo/History/default.asp](http://www.elotouch.com/AboutElo/History/default.asp)> [Accessed 2010-11-23].

## Appendices

<i>Appendix A - Planning report</i>	I
<i>Appendix B - Project plan</i>	III
<i>Appendix C - Project Risk Analysis</i>	IV
<i>Appendix D - Focus group questions</i>	V
<i>Appendix E - 30 ideas</i>	VI
<i>Appendix F - 10 ideas</i>	VII
<i>Appendix G - Scenario - IKEA &amp; Coop</i>	VIII
<i>Appendix H - Effect Map</i>	IX
<i>Appendix I - Interview Questions</i>	X
<i>Appendix J - Basic overview of iOS graphical user interface</i>	XI

## ***Appendix A - Planning report***

### **Planning report**

Master Thesis in Interaction Design Program at Chalmers University of Technology

#### **Background**

The iPhone/iPad platform is a relatively new and definitely exciting platform for developing content which this project plans to explore. The touch technology that has now become more or less mainstream, due in large to the success of the iPhone, is an interesting albeit quite old technology that opens up new interaction design possibilities.

#### **Aim**

The aim of the project from the student's point of view is to learn, through experience, the complete process for defining, designing and ultimately realising an application for the iOS4 platform. The application should be released commercially in Apple's App Store.

The aim of the project from inUse's point of view can be seen more or less as a marketing campaign. They want the project to result in media exposure, as a way to promote inUse as a partner for IT-solutions.

#### **Goal**

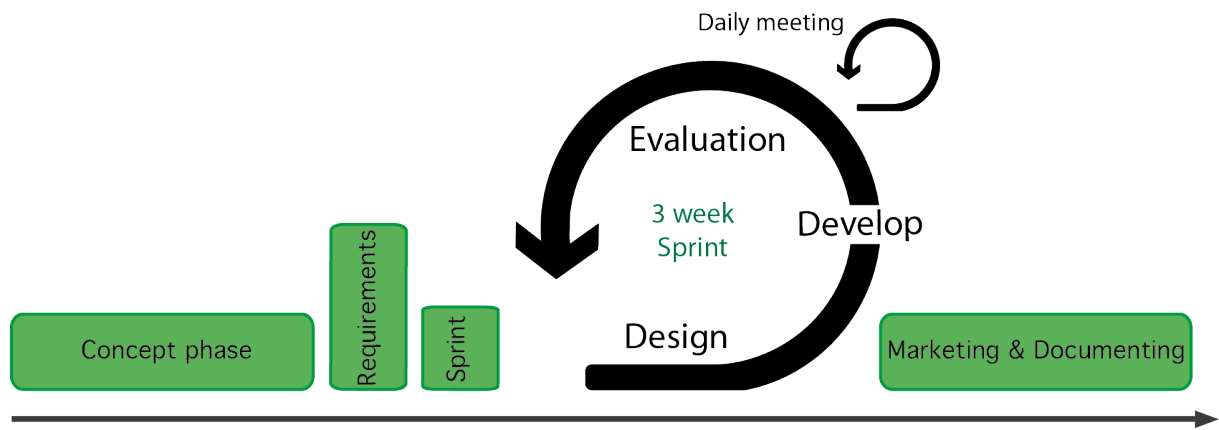
The goal of the project is a commercial product that will be available through Apple's App Store.

#### **Method**

Our development process will focus on an iterative progression where design and development go hand in hand. In the beginning of the project focus will be on generating a plethora of ideas, without any form of filtering. The aim is to get as many ideas as possible to begin with, with their respective qualitative aspects considered later.

Having generated enough ideas these will then be refined and reviewed with the goal of finding an idea that will form the framework for the project. This idea might be connected with an external company in which case a deal with the company in question will be made, considering copyright, patents and other legal aspects.

The idea will then be designed and developed through the use of a multitude of methods but the process will follow the general structure of the image seen below. At the end of the development phase, time has been allocated for marketing as understood by inUse's aim for the project. A more detailed time schedule can be seen further down on this page.



## Scope

The scope of the project is limited to the development of a gold master version of the application at hand. No further development within the context of this master thesis will be done, when the app has been commercially released.

The focus of the project will be on interaction design. As such, some technical aspects concerning programming may be handled by people other than the student, if it is deemed necessary to finish the project at hand. The goal here is not for the students to learn as much about software programming for the iPhone (even though the students will aim for programming as much of the apps as the are capable of) but rather on the design of the actual interaction a user has with the application.

## Time Schedule

Initial Time Plan	w.22	w.23	w.24	w.25	w.26	w.27	w.28	w.29	w.30	w.31	w.32	w.33	w.34	w.35	w.36	w.37	w.38	w.39	w.40	w.41	w.42	
<b>Concept</b>																						
Project Plan																						
Initial Analysis																						
Generate Ideas																						
Final Concept & BC																						
<b>Design &amp; Testing</b>																						
Prototype 1																						
Vacation																						
Prototype 2																						
Prototype 3																						
Final Design																						
<b>Development</b>																						
Alfa Build																						
Beta Build																						
RC Build																						
Gold Build																						
<b>Marketing</b>																						
<b>Report</b>																						
Write																						
Presentation																						
Opposition																						

## **Appendix B - Project plan**

### **Concept phase**

21 May – 2 July (5 weeks)

- Project plan
  - Generate ideas
- Goal: Final Concept
- } Evaluation throughout the phase

### **Design phase**

5 July – 6 August (4 weeks + 1 week holiday)

- Prototype 1
  - Prototype 2
  - Prototype 3
- Goal: Final Design
- } Evaluation throughout the phase

### **Development phase**

9 August – 1 October (8 weeks)

- Alfa build
  - Beta build
  - RC build
- Goal: Gold build
- } Evaluation throughout the phase

### **Marketing**

4 October – 22 October (3 weeks)

### **Report**

4 October – 22 October (3 weeks)

- Write
- Presentation
- Opposition

## Appendix C - Project Risk Analysis

No	Risk description	Probability (1-5)	Impact (1-5)	Risk (1-25)	Measures to eliminate/ reduce risk	Measures if a risk occur
1	Unable to finish all the programming in time	3	5	15	Use free time to learn X-Code, get help from other developers	Take time out of writing the report to finish coding
2	Company not interested in collaborating	3	4	12	Well thought out sales pitch	Look at other companies or change concept
3	Unable to contact company due to vacation	3	4	12	Contact the chosen company before 23/6	Look at other companies or change concept
4	Developed app rejected by Apple	2	5	10	Make sure that the application follows Apples's requirements	Update application
5	Lost of data due to hardware failure	2	5	10	Use Dropbox to create a backup	Use Dropbox to reclaim lost data
6	Collaboration between team members not working	1	5	5	Cookies and hugs for everyone!	Try counseling to resolve problems
7	Lost of time due to injury	1	4	4	Avoid potentially hazardous activities	Rework project plan to make up for lost time
8	Unable to do proper user testing due to lack of available participants	1	3	3	Plan ahead	Do user tests with friends and family, use participants several times

## ***Appendix D - Focus group questions***

### **Focus group questions**

Fokusgrupp är en intervjuform där en liten grupp åsiktsfyllda människor samtalar runt ett förbestämt ämne som deltagarna har gemensamt. Detta möte kommer att fokuseras på iPhone och dess applikationer för att finna nya idéer kring utvecklingsmiljön.

#### **Fråga 1**

Vad använder du din iPhone till?

Skriv ner användningsområdena...

#### **Fråga 2**

Finns det någonting annat som din iPhone skulle kunna hjälpa dig med i vardagen?

Öppen diskussion...

#### **Fråga 3**

Be deltagarna att dokumentera deras fritidsintressen iPhone-lappen...

Har du någon gång saknat en app i samband med din fritid?

#### **Fråga 4**

Kan iPhone hjälpa dig med någonting på jobbet eller i skolan?

Öppen diskussion...

#### **Fråga 5**

Finns det någon app som du saknar till din iPhone?

Öppen diskussion...

#### **Fråga 6**

Övrigt!!!

#### **Material:**

- Mediating tool 3 iPhone prototyper på papper
- Antecknings-(l)app
- Olika sorters pennor
- Snacks o Dricka
- Recorder

## Appendix E - 30 ideas

Questions	Posten	Tamagotc	Postorder	Finanser	Träningsp	SJ	Lodestar	Job Hunt	Hemkansk	Charader	LFV	Tyda	Resturang	Önskelista
Ability to be independent	4	5	3	4	4	4	4	4	3	5	4	5	3	5
Ability to be found	5	2	5	4	4	5	2	3	3	4	4	4	1	4
Ability for media attention	5	1	4	5	2	3	5	4	2	3	3	2	5	5
Ability to attract repetitive use	3	4	4	5	4	3	4	5	3	2	4	3	2	3
Possibility of no legal issues	5	3	5	3	5	5	5	5	4	5	5	5	5	5
Dependency on external partner	1	5	1	2	1	1	5	4	3	4	1	1	1	5
Dependency on secure info transfer	3	5	2	1	4	4	3	5	5	5	4	5	5	5
Device compatibility	5	5	5	5	5	5	5	5	5	5	5	5	5	5
Team's ability to realize concept	2	4	3	1	3	3	4	3	3	5	3	5	5	5
Team's attitude towards concept	4	2	4	5	3	3	3	4	3	2	3	2	3	4
Size of target group	4	3	4	3	3	4	1	4	3	3	4	3	1	3
Competition	5	3	4	3	5	2	4	3	3	3	3	3	5	5
Does the concept fit within a trend	1	1	4	3	3	2	5	2	2	1	1	1	1	1
Does it fill a need	3	1	5	4	4	3	4	5	2	2	3	3	4	4
Is it seasonal	4	5	5	5	5	5	5	5	5	5	4	5	5	3
Is it part of a fad	5	5	5	5	4	5	3	5	5	5	5	5	5	5
National/International	1	5	3	3	1	1	1	4	4	1	1	2	3	5
Score:	60	59	66	61	60	58	63	70	58	60	57	59	59	72

IKEA	Trafikinfo	Liseberg	Boplats	Ferries/	onlinepizz	Handla	Annotera/	Keep it	Antatomy	Pharmacy	Cats vs	Maträtt	Infovis	Mecenat	Fotbollsför
3	4	4	4	4	4	4	5	5	5	4	3	3	3	1	4
5	3	5	5	3	4	3	4	3	3	3	3	3	3	4	4
5	3	3	4	3	1	4	3	3	2	3	4	2	3	4	3
3	4	1	3	2	3	5	4	5	3	2	4	3	3	4	4
5	5	5	5	5	5	5	5	5	5	4	5	5	5	3	5
1	3	1	1	1	4	1	5	5	3	3	5	2	3	1	1
3	5	5	3	5	4	3	5	5	5	5	4	3	5	4	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
4	5	5	3	5	4	4	3	5	5	4	4	4	5	3	4
5	3	2	4	1	2	4	3	3	3	2	3	1	3	2	3
4	4	1	4	2	2	5	2	3	3	3	3	3	3	2	3
5	4	5	5	3	3	5	4	3	2	4	1	4	3	3	4
1	1	1	1	1	1	4	1	3	2	1	3	2	3	1	1
5	5	3	4	3	2	5	3	1	2	3	1	2	2	4	3
5	5	1	5	4	5	5	5	5	5	5	5	5	5	5	1
5	5	5	5	5	5	4	5	4	5	5	4	5	4	5	4
5	1	1	1	1	1	1	5	5	1	5	5	3	5	1	1
69	65	53	62	53	55	67	67	68	59	61	62	55	63	52	55

## Appendix F - 10 ideas

Questions	Posten	H&M	Finance	Job Hunt	Wish List	IKEA	Traffic	Boplats	Groceries	Keep steady
Ability to be independent	4	3	4	4	5	3	4	4	4	5
Ability to be found	5	5	4	3	4	5	3	5	3	3
Ability for media attention	5	4	5	4	5	5	3	4	4	3
Ability to attract repetitive use	3	4	5	5	3	3	4	3	5	5
Possibility of no legal issues	5	5	3	5	5	5	5	5	5	5
Dependency on external partner	1	1	2	4	5	1	3	1	1	5
Dependency on secure info transfer	3	2	1	5	5	3	5	3	3	5
Device compatibility	5	5	5	5	5	5	5	5	5	5
Team's ability to realize concept	2	3	1	3	5	4	5	3	4	5
Team's attitude towards concept	4	4	5	4	4	5	3	4	4	3
Size of target group	4	4	3	4	3	4	4	4	5	3
Competition	5	4	3	3	5	5	4	5	5	3
Does the concept fit within a trend	1	4	3	2	1	1	1	1	4	3
Does it fill a need	3	5	4	5	4	5	5	4	5	1
Is it seasonal	4	5	5	5	3	5	5	5	5	5
Is it part of a fad	5	5	5	5	5	5	5	5	4	4
National/International	1	3	3	4	5	5	1	1	1	5
Score:	60	66	61	70	72	69	65	62	67	68

## **Appendix G - Scenario - IKEA & Coop**

### **Scenario för IKEA**

Maria sitter hemma och surfar på IKEA.se. Hon behöver förutom ett antal mindre saker, en ny sängstomme och tillhörande hurts. Hon sätter ihop en inköpslista och laddar över den till sin iPhone och IKEA appen. Sedan tar hon bussen över till IKEA Bäckebo. På vägen kommer hon på att hon glömt lägga till värmeljus på listan så hon tar upp sin iPhone, öppnar IKEA appen och bläddrar igenom alternativt söker i katalogen och hittar ett paket värmeljus som hon lägger till i sin lista.

Väl framme vid IKEA stiger Maria in i butiken, tar en påse, går upp en våning och tar upp sin iPhone. Hon stöter på en bunge med värmeljus som hon lägger i påsen och bockar av på inköpslistan. Hon rör sig mot sängavdelningen för att titta på en sängstomme och hurts. Vid sängavdelningen hittar hon en sängstomme som hon tycker om, Malm, och tar således kort med iPhones kamera på streckkoden som tillhör sängstommen. Hon får möjlighet att välja vilken variant (träslag, färg) som hon är intresserad av, varpå sängstommen läggs till i inköpslistan. Hon hittar sedan en hurts, i samma serie, som hade passat perfekt och tar kort på dennes streckkod. Hon väljer sedan variant och får reda på att den varianten för närvarande är slut i butiken. Hon får då möjlighet att välja en annan variant för att se om någon annan IKEA butik i närheten har den i lager eller sätta produkten under bevakning. Maria vill ha hurtsen i samma träslag som sängstommen och väljer därför att kolla om den finns på IKEA Kålleröd. Olyckligtvis är den slut där med så Maria sätter hurtsen under bevakning. Då produkten åter finns i lager kommer hon få en push-notifikation om detta.

Maria plockar på sig övriga småsaker som hon har på sin inköpslista och bockar av dem vartefter. Hon hittar även några småsaker som hon inte fanns med på listan som hon plockar på sig, varpå hon tar kort på deras streckkod för att få med dem på listan. Maria beger sig sedan till lagret för att hämta sängstommen. Lagerplats ser hon i appen och hittar lätt fram till rätt hylla, tar med sig en sängstomme och vandrar mot kassan.

Vid kassan registreras de varor Maria bockat av på sin iPhone automatiskt och hon drar sitt VISA-kort, betalar, packar sina varor och åker hem för att sätta ihop sin nya säng.

### **Scenario för COOP**

Johan sitter på bussen på väg hem från jobbet. Trött efter arbetsveckan planerar han att inhandla all mat för helgen på Coop Forum inte långt från sin bostad. Med sin iPhone sitter han och bläddrar bland recepten i Coops iPhone-applikation. Plötsligt råkar han på ett recept som fångar hans uppmärksamhet och som han vet skulle passa utmärkt en varm, solig fredagskväll som denna. Med hjälp av applikationen skapar han automatiskt en inköpslista med ingredienserna för receptet i fråga. Johan fortsätter sedan bläddra och lägger successivt till ett antal andra recepts ingredienser för resten av helgens middagar. Han lägger dessutom manuellt till ett antal basvaror i inköpslistan som han vet att familjen behöver. Fokuserad på sin uppgift missar Johan nästan sin hållplats men hoppar i sista stund av bussen precis som dörrarna stängs bakom honom. Han blickar upp mot den molnfria himlen tar en vagn och beger sig in på Coop Forum. Inne i affären plockar Johan på sig de varor som finns uppskrivna på inköpslistan och med hjälp av kameran på iPhones skannar han varornas streckkoder varpå de registreras. När alla varor på inköpslistan blivit registrerade, tillsammans med några spontana köp som Johan lockats av, beger han sig mot kassan. Väl vid kassan checkar han enkelt ut via iPhones, drar sedan sitt MedMera-kort, betalar, får sitt kvitto och går tillbaka ut i solen. Han tar på sig solglasögonen och börjar traska hemåt samtidigt som han ler mot solen och åt tanken på en härlig middag på terrassen.



## ***Appendix I - Interview Questions***

### **Interview Questions**

**1. For what occasions do you normally buy gifts?**

Birthday, Christmas, Name day, Father's/Mother's day, Baby shower, Graduation, Wedding, Engagement party, Easter, Halloween,

Other: \_\_\_\_\_

**2. How do you plan your gift purchases for birthdays, christmas etc?**

**3. a. How often, if at all, do you write a wish list?**

**b. If so, how do you write it? (on a piece of paper, document on PC, notes in cellphone)**

**4. How do you go about spreading your wish list with friends and family?**

**5. Are you ever worried that you will buy the same gift as someone else for a friend?**

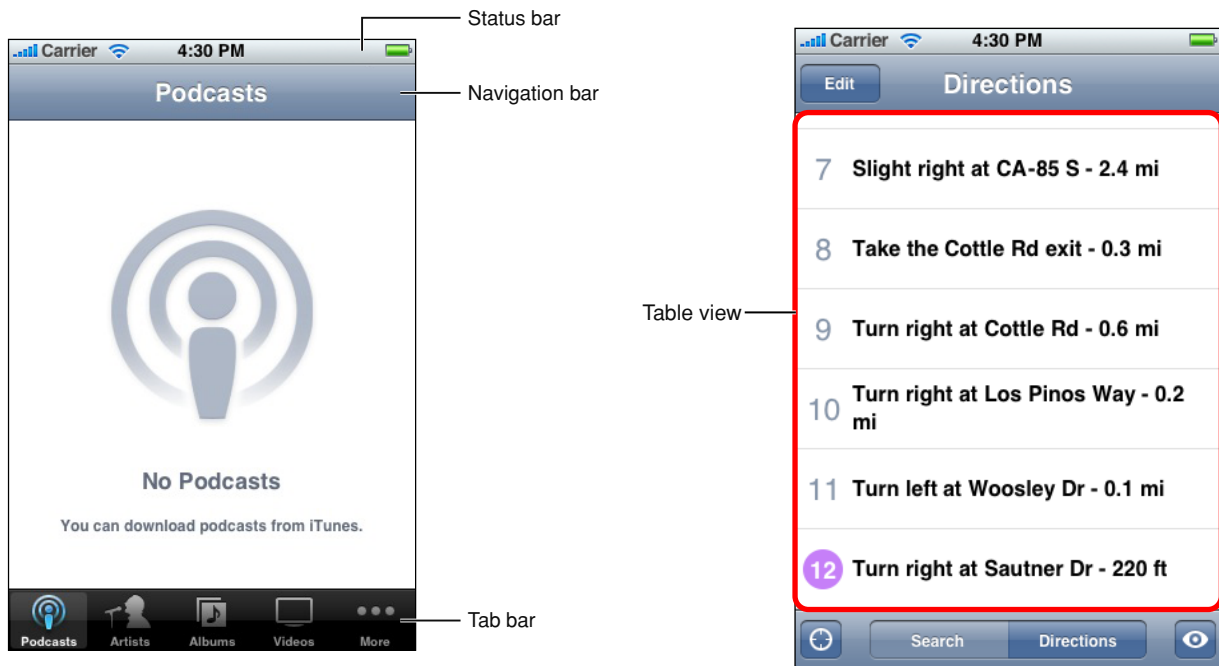
**6. a. Do you ever get together with friends to buy a more expensive gift for someone?**

**b. If so, how do you go about planning that?**

**7. Have you ever found something that you would like for Christmas only to forget to tell anyone or forgot to write it on your wish list?**

**8. Would an app that does, all that our app will, be interesting to you?**

## Appendix J - Basic overview of iOS graphical user interface



iOS GUI images (Apple, 2010b)

- **Status bar.** The status bar provides the user with information not necessarily relevant to the current app but instead provides overall device information such as battery charge, current time and signal strength. The status bar can be hidden when using an app, however this is not recommended
- **Navigation bar.** The navigation bar gives the user feedback on where he or she currently location within an app. It usually provides a means to navigate back to where you came from, by use of a back button on the top left corner. It also provides optional buttons for managing items in a view, for example: edit, save or delete
- **Tab bar.** The tab bar is a means for having multiple views inside an app and provides a simple way to navigate between them, by use of buttons at the bottom of the screen. Pressing a button will switch the application view to the one related to the button pressed. The tab bar buttons do not provide functionality for interacting with items inside a view but only for the navigation between them
- **Table view.** A table view is basically a long list of cells, where each cell can be interacted with and provide some type of functionality. The most common interaction of tapping a cell is that of a detail view showing information about the cell that could not be shown in the limited space of a table view cell.

(Apple, 2010b)