November 22, 2015 v1.00

Team C

PostCardBuddy

Project Experiences

Authors of this document:
Emma Albertz
Caroline Brandberg
Linnéa Claesson
Billy Johansson
Johan Ju
Jacob Mejvik
Carl Rynegardh

Contents

1 Background

During the first week, and the work for release one, we have worked to figure out who our stakeholders are and what they would like to see from our system. We have used a lot of different elicitation techniques and tried to keep requirements at a fairly high level, goal- or domain-level. Finding requirements through different elicitation techniques have been the highest ranked priority for release one.

2 Methods and Techniques

2.1 Elicitation

To find relevant elicitation techniques, *Software Requirements - Styles and Techniques* by Soren Lauesen have been used as a guidance[1]. A first stakeholder analysis was conducted and the different stakeholders were then approached using different elicitation techniques.

The following elicitation techniques were used prior to release 1:

Brainstorming Used as a first step within the team to come up with basic ideas and functions. During the brainstorming session the functions specified by the key customer were also considered.

Questionnaire The questionnaire was sent out to people within the end user group. Questions from the brainstorming session were used. People answering where asked to grade functions with grade 0-5, where zero stood for not interesting and five for very interesting. An age field was added to see if there was a difference in interest of various functions between ages.

Interviews The key customer was interviewed.

Prototypes Three team members created one prototype each, independently of each other so as not to affect each other's ideas. It was decided to do this right away due to the time constraint upon this project. The prototypes are meant to be used for ideas to the graphical interface of the application. The use of prototypes is considered a suitable technique for this project since there are many easy to use and free programs available to make them and it gives not only the stakeholders but also the authors of the requirements a good idea of what it should look like and be able to do. It will be specifically be used when eliciting requirements from prospective end users.

Document study There is already a similar existing application on the market and it was used to further elicit functionality not already thought of and also to perhaps eliminate functionality that intervenes with the user experience. This was done after the initial brainstorming, to avoid making an identical application and not interfere with the team's creativity. This will be done more thoroughly for release 2.

Focus group

2.2 Specification

Context diagram A context diagram will be used since it is easy to use at the time for validation and verification. The diagram gives a good over-view of the system, both for the use of the client but also for the developers.

2.3 Validation

Prototypes The prototype gives the customer a unique opportunity to validate how the product matches their expectations. The prototypes will be contentiously adapted to the customers' needs and wants and new features will be added (or others removed) so that it becomes a good reflection on where the project is going.

2.4 Prioritization

3 Reflections

3.1 Elicitation

Prototypes A program was used for constructing the prototypes that has worked very well so far. It also proved to be of use for brainstorming new ideas and features, since the program itself offered a lot different options on how to do things.

Questionnaire Figure 1 presents the result of the questionnaire, which 38 persons answered. To get answers from that amount of people was no problem and it gave a start of what the users were interested in. The result of this is that the functionality "Share postcard on social media" was not important and "Suggestion for GPS-based images" was appreciated. This will be considered for release 2. The result also shows that the desired functionality did not change that much depending on the age. Using a questionnaire was interesting since it gave a good idea of the functions people are interested in. However, as the questionnaire was created it was desirable that it is quick to answer. Therefore, only ten questions were used to maximize the number of respondents and the quality of the answers. Afterwards it was realized that some interesting functionalities were missing. Knowing the interest of these functionalities as well could be of interest and might be investigated further prior the next release.

Document studies Document studies: The already existing app is easy to use and slim. It does not contain a lot of functions but there are enough. Most of the basic functions are already implemented. However, there are definitely some functionalities that could be of use that are not implemented. Also, the library of images is not very big and GPS based images depending on your localization only works in Sweden and Denmark. Thinking about this for release two and three would be a good idea.

Data model Creating the data model for the data requirements was in itself an exercise in elicitation. While gradually developing the ER-diagram new entities and relationships that were not easy to spot in the beginning started to emerge. This was largely due to dependencies between between different types of required data.

3.2 Specification

Context diagram The first context diagram created is presented in PMv2. The first diagram is very limited and contains too little information to understand the system. The updated diagram is presented in release 1 of the report System Requirements. The biggest problem creating a context diagram is that it should be big enough to present important details, but small enough to be able to get a overview of the system. Therefore it is very important to think through which components it should contain, and which should be left out. This

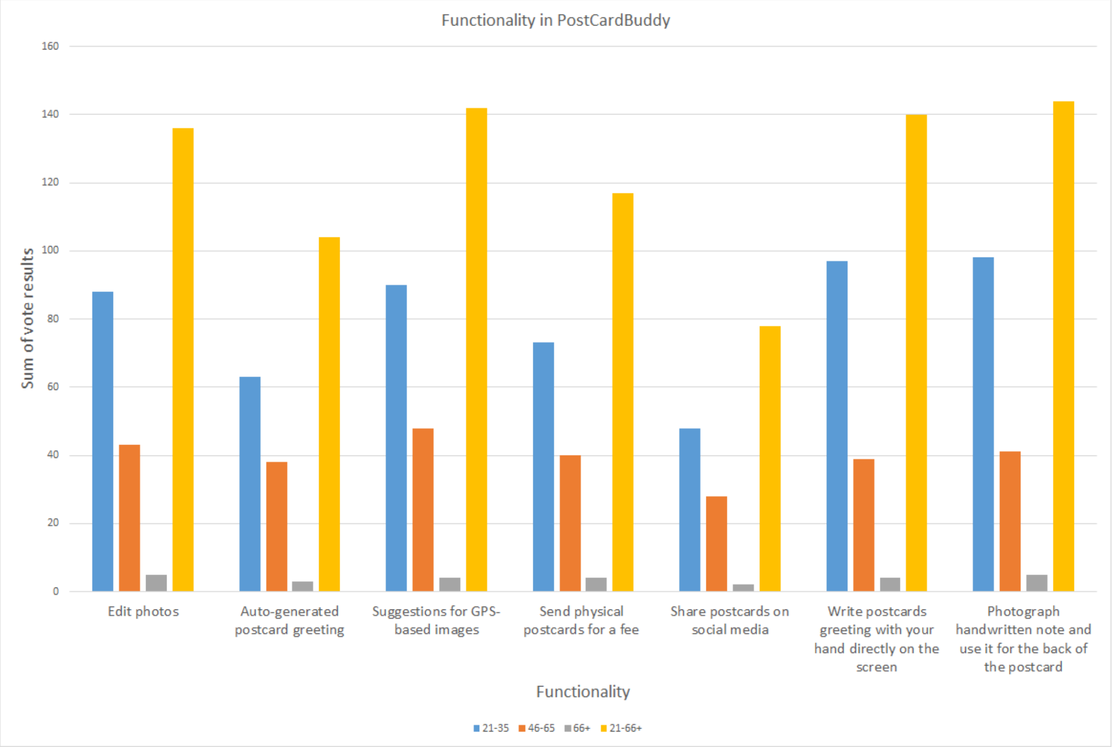

Figure 1: Result of the questionnaire on the desired functionality in PostCardBuddy

difference is often personal, which we noticed during the creation of release 1, which led to some discussion. The most time of the discussion were spend talking about if the back-end should be presented and how the functionality that is used within the mobile should be presented.

Data model The data model is a good tool to easily visualize dependencies of different systems and stakeholders. If done thoroughly it could be used as a good starting point for developers, and in particular database developers. But the more complicated the data model becomes, the harder it gets for non technical personnel to understand it. And in the same way it loses some of its value for developers if it is not thorough enough. This trade off probably means that this technique should be combined with some other tool, such as a data dictionary, to adequately satisfy technical as well as non-technical personnel.

3.3 Validation

3.4 Prioritization

4 Personal Statements

References

[1] Soren Lauesen, *Software Requirements - Styles and Techniques*, Pearson Education Limited, 2002