November 20, 2015 v0.05

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.

2 Methods and Techniques

2.1 Elicitation

We techniques we have used have been inspired by "Software Requirements Style and Techniques" chapter 8, written by Soren Lauesen. The figure 8.2, page 338, has been of great help in choosing appropriate elicitation techniques. For release one we used the techniques: Brainstorming, Questionnaires, ??Prototypes??, Focus groups, Similar companies, ask suppliers and stakeholder analysis.
Brainstorming: To come up with different functions we at first used brainstorming writing down the functions we could come up with. During the brainstorming session we also thought about, and extended, the specification and function given from the ??group that first came up with PostCardBuddy??
Questionnaire: The questionnaire contained functions that we came up with during the brainstorming session. Persons answering where asked to grade functions with grade 0-5, where zero stood for not interesting and five for very interesting. We also added a field for age to see if we could make out a difference in the interest of different functions between ages, stakeholders.
Prototypes: As we are very time constrained, having to come up with requirements in just three weeks, we decided to already create prototypes. Four of us made our own prototypes. As we are early in making requirements it was only about coming up with ideas for the graphical design of the app. In order not to affect each others ideas we designed them in isolation. Prototypes is a suitable technique for our product because of that we are specifying an application for end users. It helps the elicitation process by having uses say "Can I do that here" when they test the prototype.
Focus group:
Similar companies:
Stakeholder analysis:

2.2 Specification

Context diagram: A context diagram will be used because of its strengths to be easy to use at 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 our customer a unique opportunity to validate how our product match their expectations. We are trying to contentiously adapt the prototype to our customers needs and new features so that it becomes a good reflection on where the project is going.

2.4 Prioritization

3 Reflections

3.1 Elicitation

Prototypes: We found an easy to use program for constructing prototypes that have worked very well for us. When we made many individual prototypes we discoverer that it also worked as practical brainstorming where we found features in the prototypes.
Questionnaire: Figure 1 present the result of the questionnaire, which 38 persons have answered. To get answers from that amount of people was no problem, but it still 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 under release 2. The result also shows that the desired functionality didn't change that much depending on the age.

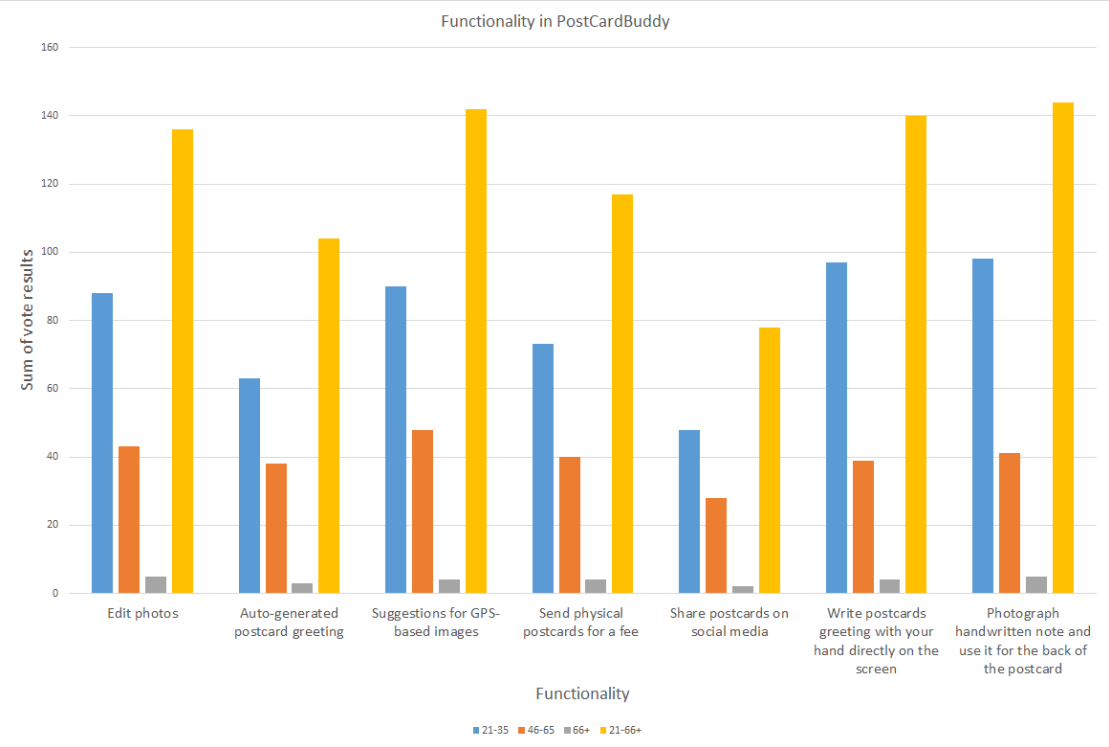

Figure 1: Result of the questionnaire on the desired functionality in PostCardBuddy

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 over-view of the system. Therefor it is very important to think through which components it should contain, and which should be left out. This 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.

3.3 Validation

3.4 Prioritization

4 Personal Statements