Danish Travel Card(Rejsekort) Proposal

Jorge Y. Castillo Rodriguez
Hussein Salem
Aku Nour Shirazi Valta
Faisal Jarkass
Dhiraj Bikram Malla

IT University Copenhagen
http://jycr753.github.io/SSEQ-Report/

November 3, 2013

Collaboration Protocol

Purpose: Shared agreements on overall forms of team collaboration.

Team members:

Hussein Salem

Faisal Jarkass

Aku Nour Shirazi Valta

Dhiraj Bikram Malla

Xiaolong Tang

Jorge Castillo Rodrguez

1. **What are the time slots that you can set aside for the team work?** Wednesday: after lectures till 14:00 [12:30-14:30] At a later point we might change the time We could work longer time in the future if we need to.
2. **If you disagree with each other or think that the solutions/products of some of the others is not satisfactory, what is the tone of communication?** Group discussion / democracy
3. **What is the maximum length of postings?** 500 words
4. **What do you do if a team member remains silent?** One of the team members will talk with that individual. And we might talk about it with the TA
5. **Do you expect that team members ask for help if they recognize, they do not manage their tasks?** Yes. We should have an open dialog where our problems and concern can be discussed
6. **Which team conflicts might arise and how do you imagine dealing with them?** Deadline issues, help the one that couldnt do it or perhaps give him more time.
7. **How and where do you share files?*You might consider google docs, dropbox, or the BSCW.*** Google drive - File Sharing Google docs - Document editor Github - Latex Repository share
8. **Which tools for communication will you use for what purpose?** The main communication tool in our group will be Skype, since it is a free services that provides live chat, video and file transfer protocol if needed.
9. **how are you going to communicate asynchronously?** We decided to communicate through Skype and emails.

Prioritised List Of Software Qualities

Software Qualities	Points	Meaning
Safety	82	N/A
Security	**590**	Strong code structure in case hacking into system(F.x SQL injection) - Unit test -Errors report from users
Reliability	**560**	Computer programs ability to perform its intended functions and operations in a system's environment, without experiencing failure (system crash). IEEE-Std-729-1991: Software reliability is defined as the probability of failure-free operation for a specified period of time in a specified environment -Random testing -Redeveloping -Testing in different environments
Resilience	37	N/A
Robustness	210	N/A
Under-stability	210	N/A
Testability	270	N/A
Adaptability	310	N/A
Modularity	85	N/A
Complexity	120	N/A
Portability	50	N/A
Usability	**580**	The quality of user experience across websites, software, products, and environments. It is easy for users to understand and using the system. -Usability Design(UI) -Feedback from users
Reusability	82	N/A
Efficiency	150	N/A
Learnability	280	N/A

Development Model

In this section we will arguments the points for the chosen development methodology and how it can use to implement to develop the *Danish Travel Card*'s software while developing its software.

There is many developing software methodologies which describes the development life cycles and each its own importance in software development.

Waterfall Methodology

The ***Waterfall Method*** is one of the oldest and basic developments methodologies which still has a great influence in modern day software development and is follow by many software developments companies around the globe to organize their projects.

The Waterfall Method is a linear process where a sequential methodology is followed, progress is monitored and measured according to the completion of each phase.

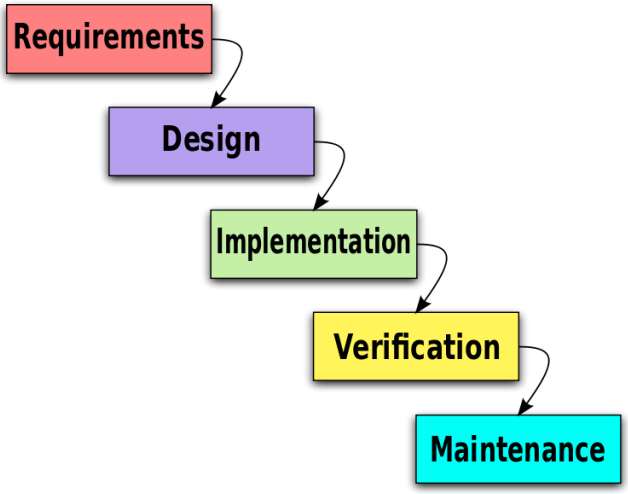

Figure 1: Waterfall Method

Waterfall Phases:

—**Requirements:** The first step in this methodology to start with requirement analysis and revise in the project is actually feasible with this methodology.

•**Design:** With the requirements and analysis acquire in the preview step, them a proper implementation strategy can be formulated according to the software environment. Further the design phase is divided in two sections. *system design and component design*. The system design contains details and specifications of the whole system and explains how each component of the system will interact with others. The component design contains specifications as to how each component will work separately.

•**Implementation:** In this phase is where actually some components start being created. The information collected in previews phases are applied in this step to create a functional software.

•**Testing:** This step is where the software developed in the preview phase is tested for any errors or discrepancies. In the Waterfall method the testing normally starts when the code is finished.

•**Installation:** After the software is tested, it can follow to the next phase where it gets installed in the computer or devices that it is build for.

•**Maintenance:** This final phase it could be stretched it for a few month or some years. With this follow up software bugs are fixed and adding/changing additional requirements or functionality to the software.

Advantages and Disadvantages

Advantages:

Since the waterfall method is the oldest and most widely used among software development companies there some advantages to this model:

1. Since it is a linear model, is very simple to implement.
2. The amount of resources to implement it are minimal.
3. Each stage is assigned to separate teams, this approach ensure project deadlines control.
4. Since the documentation is produced at every stage of the software development, this makes its understandably simpler.
5. After every major phase of software programming, testing is done tp the correct runtime.
6. Defined start and end points for each phase of the process, making it more easy to measured.

Disadvantages:

1. Fixed requirements difficult to change.
2. No final user visibility of the software until the development phase has been completed.
3. Clients are not very clear of what they want exactly from the software, when changes are mention in between it may cause a lot confusion.
4. Small changes or errors that may come up in the final software could cause a lot problems.

The waterfall works really well for commercial arrangements, but when working for internal customers its more difficult to implement.

Usage

TO BE DONE HERE