Project methodology

1. Scrum

In order to make this project effective and agile, we are going to use Scrum as software development method. We are also using Scrum in order to practice and learn more about the methodology. For these educative purposes, we will switch some roles/characters during the process. Scrum is an iterative and incremental agile software development method for managing software projects and product or application development (Wikipedia, 2012).

This project have the following characteristics:

Product Owner:
Reading week 4: Rikard Andersson
Reading week 5: Oscar Brodefors
Reading week 6: Emil Nyström
Reading week 7: Filip Askviken
Reading week 8: Rikard Andersson

Represents the stakeholders and the business and is responsible for prioritize items and adds them to the product backlog.

Scrum master:
Reading week 4: Emil Nyström
Reading week 5: Filip Askviken
Reading week 6: Oscar Brodefors
Reading week 7: Rikard Andersson
Reading week 8: Emil Nyström

Ensures that the SCRUM-process is followed, removes impediments, and protects the team from disruption.

Development team:
Reading week 4: Oscar Brodefors, Filip Askviken
Reading week 5: Rikard Andersson, Emil Nyström
Reading week 6: Rikard Andersson, Filip Askviken
Reading week 7: Oscar Brodefors, Emil Nyström
Reading week 8: Oscar Brodefors, Filip Askviken

According to Deemer et al (2010), Scrum structures development in cycles of work called Sprints. These iterative sprints are

no more than one month each, and take place one after the other without breaks. The Sprints are time bound, a spring end on a specific date when the work has been completed or not. A sprint is never extended. In the beginning of each Sprint, a cross-functional team (the development team) selects items (requirements) from a prioritized list. The items needs to be finished at the end of the sprint and does not change during the sprint. After a sprint is finished, the team reviews the sprint with stakeholders and obtain feedback which can be incorporated in the next sprint. A sprint is done when the code is integrated, fully tested and potentially shippable (Deemer et al., 2010).

1.1 Starting Scrum

This project team started working with Scrum when the Product Owner formulated the product vision. Eventually, this evolved in a list of requirements and "to do's" called the product backlog. This backlog exists and evolves over time and is the road map for the project.

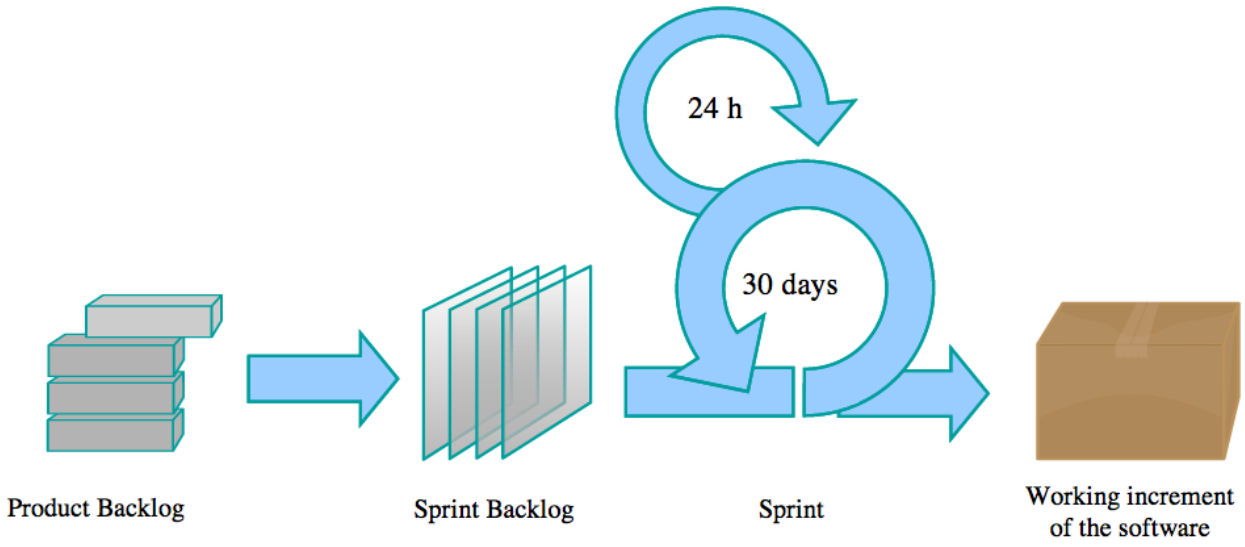

Figure 1 - Scrum project management method. Image is based on public domain graphics from Open Clip Art Library (openclipart.org), http://en.wikipedia.org/wiki/File:Scrum_process.svg

The parts in the product backlog is prioritized into different sprints of various length.

1.2 Daily Scrum

Every workday with the project will start with a short meeting of 15 minutes. The meeting aims to synchronize work and brief the others what you have done since last time. The meeting also aims to make sure that everyone is highly involved in the different parts of the project.

1.3 Sprint planning

Every achieved sprint should have tags.

Sprint one
Aim: have a base for an application made. Only a short text string showed in the application. All requirements written.
Purpose: Starting to use the Android-interface
Requirements: none
Sprint length: one day
Deadline: 2012-04-04
Achieved: 2012-04-04
Obtained feedback and future improvement: none

Sprint two
Aim: Have a start-GUI with four buttons: Search, Store, Add book, Settings. Have a list implemented in the Store. Have a database and table set-up.
Purpose: Starting to use the Android-interface to a higher extent
Requirements: NF1
Sprint length: 1 week
Deadline: 2012-04-16
Achieved: 2012-04-16
Obtained feedback and future improvement: The database needs to work in order to get the list to work.

Sprint three - functionality for uploading a book
Aim: Have the GUI working with functions and activities. Functionality for uploading a book finished. Working synchronization with the database.
Purpose: Maintain the work in progress and development
Requirements: 1.1.1, 1.1.3, 1.1.4, 1.1.5, 1.1.6, 1.1.7
Sprint length: 3 weeks
Deadline: 2012-05-12
Achieved: 2012-05-08
Obtained feedback and future improvement: Bugs: #10, #11, #12 are to be made.
Release: Version 0.1, date 12-05-08

Sprint four - bug fix
Aim: Solving bugs
Purpose: Maintain the work in progress and development
Requirements: 1.1.5
Sprint length: 4 days
Deadline: 2012-05-13
Achieved: 2012-05-13
Obtained feedback and future improvement: Bug #11 is to be made.
Release: Version 0.1.1, date 12-05-13

Sprint five - further functionality development - User details, ISBN-connection, well developed search function
Aim: Developing the functionality of the application
Purpose: Maintain the work in progress
Requirements:
2.1 - A buyer can search the database for available books
2.1.1 - In a search, a buyer can specify a book's title
2.1.2 - In a search, a buyer can specify a book's author
2.1.3 - In a search, a buyer can specify which course the book should be used in
2.1.5 - When obtaining answers on a search, a book can be pressed in order to read more details
2.1.6 - When obtaining answers on a search, the short list should present book title and price
3.1 - Working connection with LIBRIS-database
3.2 - A book can be uploaded by ISBN-number
4.1 - The user should be able to store his/hers user details on the internal memory
4.2 - The previously saved data should be loaded upon entering the settings-tab and autofilled into the fields
4.3 - The user should be able to delete previously saved user details.
Sprint length: 4 days
Deadline: 2012-05-15
Achieved: 2012-05-17 (two days late)
Obtained feedback and future improvement: This sprint includes a lot of new functionality. Bug #11 is to be made.
Release: Version 0.2, date 12-05-17

Sprint six - further functionality
Aim: Developing the functionality of the application by editing and delete a book from the market. Also, a more developed search function is to be made.
Purpose: Maintain the work in progress
Requirements:

1.2 - The seller can update information about a book or take a book off of the market
2.1.4 - A search can be sorted on any of the available attributes that a book can have. Originally it is sorted by price
2.2 - Buyer can upload a buy request
Sprint length: 1 day
Deadline: 2012-05-18
Achieved:
Obtained feedback and future improvement:
Release:

Sprint seven - Working buying request
Aim: Developing user functionality
Purpose:
Requirements: 2.2
Deadline:
Achieved:
Obtained feedback and future improvement:

2. Releases

Every release is connected to a sprint. A release should include new functionality to the application. Release 1.0 is a completed application ready for public use. A release with a odd number is a internal release, and a release with a positive number is a public release. A release with the digit-length of 0.1, 0.2, 0.3 et cetera indicated a change in a part of the application. A release with the digit-length of 0.1.1, 0.1.2 et cetera indicates a bug fix.

A new release includes the following documents:

- Changelog
- Test report
- Installation and new user instructions
- Apk-package

3. Test driven development

Test driven development (TDD) is a software development process which relies on a iterative process where a developer writes failing automated tests that defines a desired function, and then implement the code to pass the test afterwards (instead of

doing the other way around). TDD was not supposed to be used in this course and for this project. However, one of the tests in this project derived new functionality and therefore, TDD have been tried and used to some extent. The test for equals DataBook.java, with regards to search function with the LIBRIS database, gave us information that the equals-method did not do what is was supposed to do. This has to be solved by overriding the Java Object equals method.

4. Sources

Deemer, P, Benefield, G, Larman, C, Vodde, B (2010). *The Scrum primer*. Scrum Training Institute, version 1.2.

Sommerville, Ian (2007). *Software engineering*. Eight edition, Pearson Education Limited, Edinburgh Gate, England.

http://en.wikipedia.org/wiki/Waterfall_model

http://en.wikipedia.org/wiki/Scrum_(development)

http://en.wikipedia.org/wiki/Test-driven_development