University of Tartu

Faculty of Mathematics and Computer Science

System Modelling

Mancala Project

Project Plan

Authors: Simo Peterson

Mart Sein

Margus Sellin

Raigo Kodasmaa

Coordinator: Dmitri Danilov

Tartu, 2011

Version Control

Date	Changes	Comments	Author
07.11	Introduction; team; schedule; details; progress log.	About the project. Roles of team members. Deadlines and details about tasks. Happened events.	Simo
11.11	Progress log.	Adding some events into the progress log.	Simo
13.11	Progress log; conclusion.	Completing progress log. Writing the conclusion of the project.	Simo

Table of Contents

Introduction

The aim of this document is to give an overview of the Mancala project: roles of team members, details about tasks and results of the project. The project is and assignment in the "System Modelling" course. The project is hosted on Google Code: http://code.google.com/p/mancala/.

Team

Simo Peterson – Project manager, analyst, tester, presenter.

Mart Sein – Programmer, tester.

Margus Sellin – Programmer, tester.

Raigo Kodasmaa – Documenter, modeller.

Schedule

Task	Deadline	Time spent (hours)	Authors
Documentation			
Project plan	07.11		Simo
Requirements analysis	07.11		Simo
Interview with customer	08.11		Simo, Raigo
GUI mock-ups	08.11		Mart
Storyboards	09.11		Raigo
Object diagrams	09.11		Raigo
UML class diagrams	10.11		Raigo
Installation guide	12.11		Simo
User Manual	12.11		Simo
Testing Report	13.11		Simo
Coding			
Understanding and improving automatically generated code	12.11		Margus, Mart
Implementing the requirements	12.11		Margus, Mart
Running testing	12.11		Margus, Mart
Final testing	13.11		Simo
Presentation			
Slides	15.11		Simo
Presenting	15.11		Simo

Team does not have scheduled meetings. All discussion takes place in a Skype conversation. If any of the team members at some point sees the need for a meeting, it is then organised and carried out as soon as possible.

Details

1. Documentation

- Project plan – Consist of project introduction, roles of team members, project schedule, details about tasks, progress log and project results.
- Requirements analysis – A list of functional and non-functional requirements with detailed descriptions.
- Interview with customer – A discussion between customer and one team member covering 3-5 features that were missed before. Recorded on video.
- GUI mock-ups – Simple pictures to give an understanding of how user will interact with the applications and what is the difficulty level of accessing full functionality.
- Storyboards – A detailed story covering most of the functionality supported by object diagrams. An additional storyboard covering GUI supported by mock-ups.
- Object diagrams – Modelled visualisations to paragraphs of story.
- UML class diagrams – Generalisation of object diagrams covering full functionality. Convertable to Java code.
- Installation guide – How to get the application running?
- User Manual – What can be done with the application?
- Testing Report – Results of final testing. How well does the application perform compared to initial requirements?

2. Coding

- Understanding and improving automatically generated code – What kind of modifications are needed to start implementing the requirements?
- Implementing the requirements – Achieving full functionality of the application.
- GUI – A simple, but convenient graphical user interface.
- Running testing – Reduction of bugs while implementing the requirements.
- Final testing – Eliminating as much unfound bugs as possible.

3. Presentation

- Slides - Solved and unsolved problems. How was the work organised? SDM method. Demo.
- Presenting – Preparing at home and presenting to others in class.

Progress log

Date	Event	Comments	Participants
06.11	Initial Skype conversation.	How will we carry out the project and what tasks are preferred by team members?	Team
06.11	Creating repository.	Sharing repository information with team.	Mart
07.11	Meeting.	Initial outline for the project plan and confirmation that everyone agrees to the schedule.	Simo, Mart, Raigo
07.11	Project plan.	Detailed description of the project. Running updates will be made in the document.	Simo
07.11	Requirements analysis.	List of all requirements with detailed description. Minor changes still acceptable.	Simo
09.11	Requirements analysis.	Reviewing the functional requirements and adding non-functional requirements.	Simo
08.11	User story.	Writing the user story that will be used in storyboards.	Raigo
09.11	Object diagrams.	Modelling the object diagrams that will be used in storyboards.	Raigo
10.11	Storyboards.	Splitting the story into paragraphs and adding object diagrams. Storyboards completed.	Raigo
11.11	Interview script.	Writing the script for interview.	Simo
11.11	Recording the interview.	Performing and recording the interview.	Simo, Raigo
11.11	Requirements analysis.	Reviewing all the requirements after the interview.	Simo
11.11	UML Class diagram.	Source for automatically generated Java code.	Raigo, Mart

11.11	GUI mock-ups.	Visualisation of user interface in different states according to storyboards.	Mart
11.11	Project plan	Updating progress log.	Simo
12.11	Coding.	Main functionality of the game.	Mart
13.11	Coding.	Adding extra functionality to the application.	Mart, Margus
13.11	Running testing.	Elimination of bugs.	Mart, Margus
13.11	Final testing.	Running the application and writing down notes on its performance.	Simo
13.11	Testing report.	Comparing the application performance to the initial requirements.	Simo
13.11	User manual.	Guide for beginner user for using the application.	Simo
13.11	Installation guide.	Guide to get the application running.	Simo
13.11	Project plan.	Updating progress log and writing conclusion of the project.	Simo
13.11	Finalising documents.	Language proofing and formatting. Generating PDF-s.	Simo
13.11	Submitting project.	Creating a tag in repository. Sharing the location of the repository with coordinator.	Simo

Conclusion

To sum it up it can be said that this project was an interesting, but also quite difficult challenge for our team. We started working exactly one week before the deadline. All tasks were spread out nicely and by doing something every day the project progressed quite smoothly. Mostly team members completed tasks assigned to them on time. Last day was very busy for the whole team – testing and fixing minor bugs took a lot more time than we expected. Never the less we got all the major mistakes fixed and also most of the minor ones. The application is running correctly. It was developed using story driven modelling method. All team members contributed to the completion of the project.