SOFTWARE PROJECT MANAGEMENT PLAN FOR

Dots & Boxes Game



Version 1.0

3/19/2013

Prepared for:

IT 484   
Minnesota State University - Mankato

Prepared by:

Russell Walters, Chisomo Banda, Lindiwe Hove, Matt Ruppert

[**Table of Contents 2**](#_Toc531719853)

[**1** **Project Overview 3**](#_Toc531719854)

[1.1.1 Purpose, scope and objectives 3](#_Toc531719856)

[1.1.2 Assumptions and constraints 3](#_Toc531719857)

[1.1.3 Project deliverables 3](#_Toc531719858)

1.1.4 [References 4](#_Toc531719859)

1.1.5 Definitions, acronyms, abbreviations 5

[**2** **Project organization 5**](#_Toc531719862)

[2.1 External interfaces 5](#_Toc531719863)

[2.2 Roles and responsibilities 5](#_Toc531719864)

[**3** **Managerial process plans 5**](#_Toc531719865)

[3.1 Management objectives and priorities 5](#_Toc531719866)

[3.2.1 Staffing plan 6](#_Toc531719868)

[3.3 Reporting plan 6](#_Toc531719869)

[3.5.1 Risk identification 6](#_Toc531719872)

1. **PROJECT OVERVIEW**

**1.1.1 PURPOSE, SCOPE, OBJECTIVE**

This term project is to create a software system to play the game of Dots and Boxes, using the Unified Process for software development. The software system being considered for development is referred to as Dots & Boxes. The customer providing specifications for the system is Professor Schilling. The ultimate customer, or end-user, of the system will be avid and casual gamers. This is a new project effort, so the version under development is version 1.0.

The purpose of the system under development is to create a fun and challenging one or two person strategy game.

The original *Dots and Boxes* is a paper and pencil game that we propose to migrate to a computer form. Given a square or rectangular array of dots, two players take turns joining two adjacent dots with a horizontal or vertical line. When such a move adds the fourth side of a box, the player who did the deed claims the box (marking it as hers in some fashion) and must take an extra turn. A player who can complete a box is not obliged to do so. The game ends when all the boxes are taken. The player who closed more boxes is declared the winner.

There are two versions of the game:

* The first version will allow two humans to play against each other.
* The second version will allow one human to play against the software game.

**1.1.2 ASSUMPTIONS AND CONSTRAINTS**

Several assumptions are made:

* If the user is playing a two player game, both users are seated at the same computer and alternate turns.
* Games are not saved and closing a browser will quit the current game.
* The development team will provide the tools (hardware, software, etc) to complete this project.
* The development team will learn and work together to meet the deliverables of this project.
* The client will respond in a timely manner to all questions and requests for information.

There have been no imposed constraints from our client. The application is not critical so its availability is not a concern. However, the following constraints may impose limits on this project:

* Time availability for this project is constrained by other commitments the development team has.
* Additional resources (financial or human) are not available for the project.

**1.1.3 PROJECT DELIVERABLES**

Friday, February 15

* A statement of the project scope
* Risk assessment and a plan for risk mitigation
* An initial draft of the requirements
* Use cases
* General description of pre-implementation quality assurance activities

Friday, March 1

* Second draft of the requirements
* Initial draft of the overall architectural design
* Initial draft of the software requirements specification document

Friday, March 22

* Completed architectural design
* Function point analysis
* Detailed module design
* Design specification document
* Completed software requirements specification document
* Completed software project management plan
* Testing plan for the implementation

Friday, April 26

* Beta version of the project code and initial user manuals

Monday, April 29

* Results of beta testing another group’s project returned to other development team and client

Friday, May 10

* Final version of project, manual, and testing summary to client.

**1.1.4 REFERENCES**

Professor Schilling, Susan, “Dots and Boxes Project Description”, DB-SE Project.doc, February 2013.

Professor Schilling, Susan, “Project schedule for IT 484/584, Spring 2013”, Project schedule Sp2013, February 2013.

Dots and Boxes implementation available on the web at the following URL: <http://www.math.ucla.edu/~tom/Games/dots&boxes.html>

The lightweight Play Framework for Java and Scala development available at the following URL: <http://www.playframework.com/>

The Open Source Processing Language web site available at the following URL: <http://www.processing.org/>

The Open Source Processing Language for the Web – Processing.JS web site available at the following URL: <http://processingjs.org/>

## DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

|  |  |
| --- | --- |
| **Term or Acronym** | **Definition** |
| Alpha test | Limited release(s) to selected, outside testers |
| Beta test | Limited release(s) to cooperating customers wanting early access to developing systems |
| Final test | aka, Acceptance test, release of full functionality to customer for approval |
| DFD | Data Flow Diagram |
| SDD | Software Design Document, aka SDS, Software Design Specification |
| SRS | Software Requirements Specification |
| SSRS | System and Software Requirements Specification |
| DAB | Dots & Boxes |
| PLAY Framework | PLAY framework is a framework used to create applications in Java |
| Processing Language | Open source programming language and environment for people who want to create images, animations, and interactions |
|  |  |

## 2 Project organization

## External interfaces

The client for this project is Professor Schilling. Formal communication between the client and the team will be done by the Project lead. Necessary interaction can be done through anyone on the team, but all discussions and deliverables to the client will be documented clearly for records.

## Roles and responsibilities

**Project Lead:** This person will be in charge of keeping track of tasks and deadlines for the team. The project lead also acts as a liaison between the development team and the client.

**QA & Testing:** Is designated to be in charge of monitoring the development and performing integration and system testing.

**Developer:** Conducts the research, designing, implementing, unit and integration testing of the software.

1. **Managerial process plans**

**Management objectives and priorities**

The development team’s primary objective is to ensure the successful completion of the project. To be considered successful, the team must perform the following:

1. Use good software engineering methods to develop the product.

* Apply the methods learned in classes.
* Experience a new way of doing things.
* Practice reflective learning.

1. Deliver a quality product that meets the requirements agreed upon in the SRS.

* Deliver a product that is stable and relatively defect-free.
* Deliver a system that addresses the client’s needs at the client’s satisfaction.

1. Honor its commitments.

* Meet client and team deadlines.

1. Display professionalism.

* Value the time of team members and the client.
* Accept and support team decisions.
* Communicate openly and frequently.
* Take responsibility for the success of the project.
* Be proactive.

1. Make efficient use of all available resources.

* Learn from each other.
* Experiment with existing tools and processes.

**Staffing plan**

Project Lead – Matt

QA & Testing – Chisomo

Developer – Russell and Lindiwe

**Reporting plan**

Internal reporting for team members will be relatively informal. Each team member will provide a status report to the team during the weekly team meetings. This information will be used to update the project plan.

External reporting will be more formal. At the end of each semester, the team will give a presentation of the project to the IT484 class.

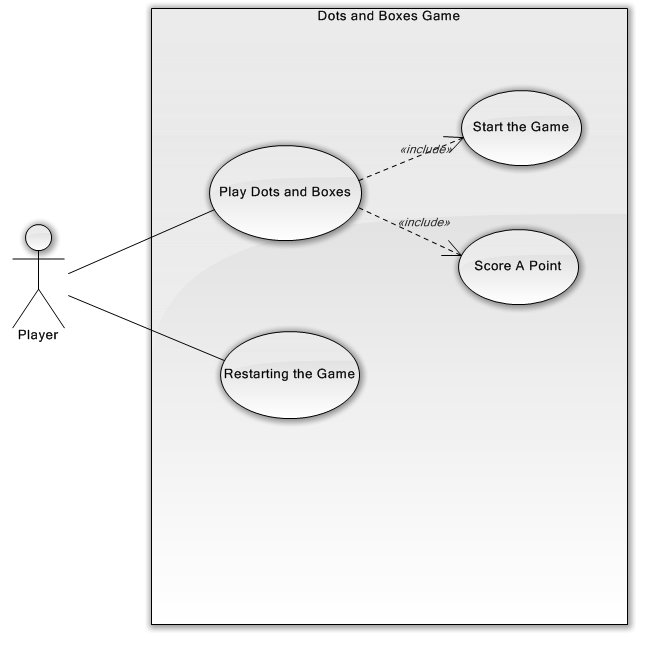
**Risk Identification**

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Probability**  **(1-10)** | **Impact**  **(1-10)** | **Mitigation** |
| Time constraints | 9 | 9 | Build artifacts in an iterative and incremental fashion. |
| Ability of the team to get together | 8 | 8 | Use of tools for communication and version control such as Git, GitHub and Heroku. |
| Team skill set | 7 | 9 | Team members on their will refresh Java and web programming skills. |
| Changing scope | 3 | 8 | Maintaining adherence to our project scope in/out list. |
| Misunderstanding of requirements | 3 | 8 | Clarify requirements with Professor. |
| Communication among the team | 2 | 8 | Meet every day after class and have regular status updates. |

**Project schedule:**

**Inception**

**Production of a Use case model**



**Business Case**

1. **Is the proposed software product cost effective?**

**Yes.**

1. **How long will it take to obtain a return on investment?**

**5 weeks to the end of semester grade.**

1. **What will be the cost if the proposed software product is not developed**

**A low grade and/or failing the class**

1. **If the software product is to be sold in the marketplace, have the necessary marketing studies been performed?**

**N/A**

1. **Can the proposed software product be delivered on time?**

**Yes**

1. **What will be the impact if the proposed software product is delivered late?**

**A low grade and/or failing the class**

1. **Are software tools needed?**

**Yes**

1. **Are they currently available?**

**Yes. See references.**

1. **Do they have all the necessary functionality?**

**Yes**

**Risks**

1. **Technical Risks**

**Design, implementation, interface, maintenance and verification problems**

1. **The risk of not getting the requirements right?**

**Low. Mitigated by performing the requirements workflow**

1. **The risk of not getting the architecture right**

**The architecture may not be sufficiently robust**

**Proof-of-concept prototype is to be built to test the feasibility of constructing part of the software product. To ensure that the requirements have been accurately determined, the test workflow also commences at this stage.**

**Elaboration**

**The initial requirements are refined. Monitoring of risks and refining their priorities. Refining the business case and producing the project management plan.**

**Construction**

**Production of the first operational quality software version of the Dots and Boxes game.**

**Implementation and Testing to be conducted:**

* **Unit testing of modules**
* **Integration testing of subsystems**
* **Product testing of the overall system.**

|  |  |
| --- | --- |
| **Time Frame** | **Activity** |
| **March 21 – April 5** | **2 player mode development** |
| **April 6** | **2 player mode ready for testing** |
| **April 8 – 12** | **2 player mode testing** |
| **April 8 – 19** | **1 player against computer mode developed** |
| **April 22 – 26** | **Final testing and bug fixing** |

**Transitioning**

**The aim of this phase is to ensure that the client’s requirements have been met:**

* **Faults in the software product are corrected**
* **All final versions of documentation are complete**
* **Attempts are made to discover any previously unidentified risks**
* **Enhancement for performance and usability.**