Software Project Management Plan (SPMP) for TicTacToe Project

*Baseline version 0.1*

*Issued on: October 30, 2014*

Issued by: Boardgames, Inc.

Issued for: Superior Company

# Signature

The following signature indicates approval of the enclosed Software Project Management Plan.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Superior Company Executive Committee representative

# Change History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Changes** |
| 0.1 | October 20, 2014 | M. Zaki Rasyadi / Aziis Yudha Adwitiya | First planning |
| 1.0 | October 28, 2014 | M. Zaki Rasyadi / Aziis Yudha Adwitiya | Project revised |
| 2.0 | October 30, 2014 | M. Zaki Rasyadi / Aziis Yudha Adwitiya | Final project planning |

# Preface

This document delivers the proposed plan to be taken by Boardgames, Inc. to meet the needs of software for Superior Company Client. In order to do so, it will deal only with the delivery of the software components in regards to the related project, which have dependencies on the hardware and network portions of the product, which Superior Company Client has been notified beforehand.

This document is created purposely for Superior Company Client executive committee. Prior to the acceptance from Superior Company Client executive committee, this document will be used by Superior Company Client’s project manager as well, and by project members and participants as a guidance to keep things accordingly.

***Important Notes for Soft-copy Viewing***

*At this part, you can mention any information for soft-copy readers. This information can include this document’s file type and its reader software (PDF is recommended). You can also mention special features that you have included for the soft-copy readers.*

# Table of Contents

[Signature 1](#_Toc402478338)

[Change History 2](#_Toc402478339)

[Preface 3](#_Toc402478340)

[Table of Contents 4](#_Toc402478341)

[List of Figures 5](#_Toc402478342)

[List of Tables 6](#_Toc402478343)

[INRODUCTION 7](#_Toc402478344)

[1.1. Project Overview 7](#_Toc402478345)

[1.1.1. Purpose, Scope, and Objectives 7](#_Toc402478346)

[1.1.2. Assumptions and Constraints 7](#_Toc402478347)

[1.2. Project Deliverables 8](#_Toc402478348)

[PROJECT ORGANIZATION 9](#_Toc402478349)

[2.1. Software Process Model 9](#_Toc402478350)

[2.2. External Interfaces 9](#_Toc402478351)

[2.3. Internal Structure 9](#_Toc402478352)

[2.4. Roles and Responsibilities 10](#_Toc402478353)

[2.5. Tools and Techniques 12](#_Toc402478354)

[PROJECT MANAGEMENT PLAN 13](#_Toc402478355)

[3.1. Work Activities 13](#_Toc402478356)

[3.2 Activity Network 14](#_Toc402478357)

# List of Figures

Figure 1…………………………………………………………………………………………..9

Figure 2…………………………………………………………………………………………..14

# List of Tables

Table 1…………………………………………………………………………………………..11

Table 2…………………………………………………………………………………………..14

**Chapter 1**

# INRODUCTION

## Project Overview

**Tic-tac-toe** (or **Noughts and crosses**, **Xs and Os**) is a [game](http://en.wikipedia.org/wiki/Paper-and-pencil_game) for two players, *X* and *O*, who take turns marking the spaces in a 10×10 grid. The player who succeeds in placing five respective marks in a horizontal, vertical, or diagonal row wins the game.

### Purpose, Scope, and Objectives

The purpose of this project is to analyze the requirements of, design, implement, and maintain the software for Superior Company Client, which is designated specifically to help in monitoring TicTacToe project according to the requirements specified by the client.

Any activity directly relates to the purpose is considered to be in scope, otherwise it is considered to be out of scope. For instance, hardware failure is out of scope of this project.

The objectives of the project are mentioned as follows:

* Complete the project by the due date
* Complete the project within the budget
* Provide all deliverables identified in section 1.1.3 by the due date
* Meet all the requirements that are mentioned in the SRS, which fall into one of these categories
  + Desktop application
  + Backend

### Assumptions and Constraints

Here is the list of all assumptions that are made:

* This project is a component of a larger project
* This project will deliver only the software components of the larger project
* The hardware is not included in this project, and is handled as a different project. Hardware is assumed to be ready at least by the end of this project.

Here is the list of all constraints that are made:

* Budget
  + Rp 450.000,-
* Time
  + 3 months.
* Staff
  + A representative from Superior Company will be required to assist in the requirements making. This representative will have the full authority in creating the final requirements, which will be evaluated and agreed by Boardgames, Inc.

## Project Deliverables

Here is the list of all items that will be available by the completion of the project.

* Software program, along with its environment and supporting libraries.
* Software documentation
  + Installation documentation
  + End-user documentation
* Installation of software program along with its environment and supporting libraries.
* Software training for users
* Project documentation
  + Software Project Management Plan (SPMP)
  + Software Requirement Specification (SRS)
  + Software Design Description (SDD)
  + Software Test Documentation (STD)
  + Software Test Plan (STP)

**Chapter 2**

# PROJECT ORGANIZATION

## Software Process Model

Software process model that we are going to use is Agile Software Development Model. There are some reasons why we choose this model. One of the reasons is because of the quality which has the principle to allow the product owner to make adjustments if necessary and gives the product team early sight of any quality issues. The other reason is that agile development principles encourage active user involvement throughout the product’s development and a very cooperative collaborative approach. This provides excellent visibility for key stakeholders, both of the project’s progress and of the product itself, which in turn helps to ensure that expectations are effectively managed. Besides that, there is also active involvement, cooperation and collaboration which makes agile development teams a much more enjoyable place for most people.

## External Interfaces

Each person has their own responsibility in process of making this TicTacToe project. There are three people; including the Project manager (also as a programmer), software architect and the analyst. Project manager took the main role with program and source code, manage and lead the project team also manage the coordination of the partners and the working groups, while the software architect did the design and built the structures for the project. Analyst job is to do some documentation, testers to verify the project, and making final report of the project.

## Internal Structure

In this project, we have three people who are involved in our team as project participants.

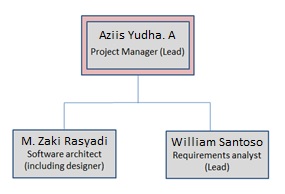


Figure 1. Internal structure

## Roles and Responsibilities

This is the table of Responsibility Assignment Matrix (RAM) with Work Breakdown Structure (WBS).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | WBS | Deliverables | CEO | Project Manager (Lead) | Requirement Analyst (Lead) | Software Architect |
| 1 | 1 | TicTacToe Project |  | L |  |  |
|  | 1.1 | Software Lifecycle Model Process | Aziis |  |  |  |
|  | 1.1.1 | Select project model | Aziis | L |  |  |
|  | 1.2 | Plan Project Management | Zaki | L |  |  |
|  | 1.2.1 | Create baseline WBS | William |  | L |  |
| 2 | 2 | Specification |  |  |  |  |
|  | 2.1 | Feasibility study | William |  | L |  |
|  | 2.2 | Requirements elicitation and analysis | William |  | L |  |
|  | 2.3 | Requirements specification | William |  |  |  |
|  | 2.4 | Requirements validation | William | L |  |  |
| 3 | 3 | Design and Implementation | Zaki | L |  |  |
|  | 3.1 | Architecture design | Zaki |  |  | L |
|  | 3.2 | Interface design | Zaki |  |  | L |
|  | 3.3 | Component design | Zaki |  |  | L |
|  | 3.4 | Acceptance design | Zaki | L |  |  |
| 4 | 4 | Validation |  |  |  |  |
|  | 4.1 | Development testing | William |  | L |  |
|  | 4.1.1 | Unit testing | William |  |  |  |
|  | 4.1.2 | Component testing | Wiliiam |  | L |  |
|  | 4.1.3 | System testing | Aziis | L |  |  |
|  | 4.2 | Release testing | Aziis | L |  |  |
|  | 4.3 | User testing | Zaki |  | L |  |
|  | 4.3.1 | Alpha testing |  |  |  |  |
|  | 4.3.2 | Beta testing |  |  |  |  |
|  | 4.3.3 | Acceptance testing | Aziis | L |  |  |
| 5 | 5 | Evolution |  |  |  |  |
|  | **Key** |  |  |  |  |  |
|  | A | Approval | 21 | 10 | … | … |
|  | L | Lead | 4 | 2 | … | … |
|  | S | Secondary | 0 | 4 | … | … |
|  | C | Contributor | 0 | … | … | … |
|  | R | Reviewer | 2 | … | … | … |

Table 1. WBS

## Tools and Techniques

TicTacToe program in this project is a strategic game with Artificial Intelligence (AI) based on the principle of Minimax Algorithm. Tools that have been used for this project are laptop, internet and printer. User Interface is designed with Java language programming. For Software interface, both user and developer is using command prompt to start the program.

**Chapter 3**

# PROJECT MANAGEMENT PLAN

## Work Activities

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **WBS** | **Task Name** | **Duration** | **Predecessors** | **Successors** |
| 1 | 1 | TicTacToe Project | 60 days | … | 2 |
| 2 | 1.1 | Software Lifecycle Model Process | 3 days | 1 | 3 |
| 3 | 1.1.1 | Select project model | 1 day | 2 | 4 |
| 4 | 2 | **Requirement Definition** |  | 3 | 5 |
| 5 | 2.1 | Feasibility | 2 days | 4 | 6 |
| 6 | 2.2 | Requirement Elicitation | 2 days | 5 | 7 |
| 7 | 2.3 | Requirement Analysis | 2 days | 6 | 8 |
| 8 | 2.4 | Requirement Specification | 3 days | 7 | 9 |
| 9 | 2.5 | Requirement Validation | 3 days | 8 | 10 |
| 10 | 3 | **System and Software Design** |  | 9 | 11 |
| 11 | 3.1 | System Analysis | 2 days | 10 | 12 |
| 12 | 3.2 | Architecture Design | 1 days | 11 | 13 |
| 13 | 3.3 | Interface Design | 2 days | 12 | 14 |
| 15 | 3.4 | Unit Design | 3 days | 14 | 16 |
| 16 | 3.5 | Overall Design | 8 days | 15 | 17 |
| 17 | 4 | **Coding** | 21 days | 16 | 18 |
| 18 | 5 | **Validation** |  | 17 | 19 |
| 19 | 5.1 | Debugging | 2 days | 18 | 20 |
| 20 | 5.2 | Requirement Validation | 1 days | 19 | 21 |
| 21 | 6 | **Integration** |  | 20 | 22 |
| 22 | 6.1 | Presentation | 1 days | 21 | 23 |
| 23 | 6.2 | Training | 3 days | 22 | 24 |

Table 2. Work activities

## Activity Network

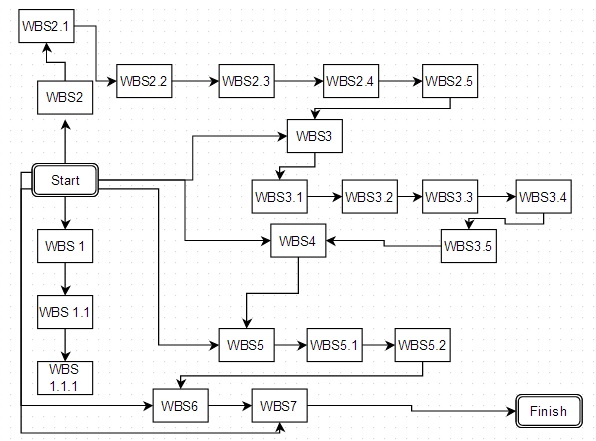


Figure 2. Activity Network