SE 3XA3: Development Plan
Ultimate Tic Tac Toe

Team 3
Kunal Shah - shahk24
Pareek Ravi - ravip2

October 11, 2016

Contents

List of Tables

List of Figures

Abstract

This document describes the requirements for Ultimate Tic Tac Toe. The template for the Software Requirements Specification (SRS) is a subset of the Volere template (Robertson and Robertson, 2012).

1 Project Drivers

1.1 The Purpose of the Project

The purpose of this project is to redevelop an existing open source project. This is to de While creating proper documentation and following a modified waterfall design method.

1.2 The Stakeholders

The stakeholders include all parties that would have a vested interest in this projected. This includes, the client Dr Smith, the developers Pareek and Kunal,all the people who would be interested in playing the game and the teaching assistant Christopher.

1.2.1 The Client

The client of Ultimate Tic Tac Toe is Dr. Smith as he is the project manager. He requested this project. We are creating this game for him to distribute to the customer.

1.2.2 The Customers

The target customer for this game would be anyone who has a device connected to the internet and wants to play a game with a friend. This game i targeted towards young people, but it is open for anyone to play.

1.2.3 Other Stakeholders

Another stakeholder is the teaching assistant, Christopher, as is aiding Dr. Smith in project management during lab as well as the developers during the development process.

1.3 Mandated Constraints

There are 3 constraints of this game.

1. The internet

2. A device that can connect to the internet

3. A friend to play the game with

1.4 Naming Conventions and Terminology

The naming convention for winning an inner game will be called controlling a square. In order to win, a player must get a tic tac toe of controlled squares. If there is a draw in a square, it will be called a dumped square.

1.5 Relevant Facts and Assumptions

We assume that the users have a stable internet connection and are aware of the rules of regular Tic Tac Toe. It is important to note that the game will work better in large screens rather than a small display. We assume that the user is using a device that is either touch enables or has a mouse to click with.

2 Functional Requirements

2.1 The Scope of the Work and the Product

We investigated the original Ultimate tic tac toe game (kennycason, 2010) . In this game the developer has poor user interface and no game instructions. Its is also only supports single device with no AI options. Theses shortcomings are the features we would like to improve upon.

2.1.1 The Context of the Work

The central task is to create a game of Ultimate Tic Tac Toe that can be played on the internet with a friend. This will require a server to host the game, each user to connect to the game, and ensure a connection with the server. In order for the game to be played, the server will determine which player is to play, get that player's move record it in the virtual board, and then notify the other player of the first player's move. This process of sending the move made back and forth will be the role of the server. The server will also indicate to a player if they have won, lost or if that game has ended in a draw.

2.1.2 Work Partitioning

The development of this game is shared evenly between the two team members. The game logic is developed by both members. The graphical portion is mainly going to be done by Kunal, and the server portion will be done by Pareek.

2.1.3 Individual Product Use Cases

One of a possible use case is when a player might possibly disconnect from their internet connection, but when their internet resumes, they wish to reconnect to the game. Another case is if a player wished to change the device they are playing on during an on going game. There could be a case where both players agree that a player should be able to revert back by one move they made, in which case the game will undo the user's last move. The general use case if of course two users playing on their devices without any flaws from start to end.

2.2 Functional Requirements

Some of the functional requirements are that the game will display a game board on the user's device screen, the user's move will be registered, recorded and reflected on the game board. In addition, the game will determine when a player has taken control of a square, when the square square is dumped and when the game is won. The game logic will also determine in which square the next player will make their move based on the previous player's move. In order for the game to be played over the internet, the game would need to communicate to both devices through means of a server.

3 Non-functional Requirements

3.1 Look and Feel Requirements

In order to make the game easy to use, clear instructions will be provided to help them. With the use of unique but neutral colours, the aesthetics of the game will also make it an enjoyable experience. Computers using a browser with strong HTML5 support will be able to run the game easily. Interactive sounds will be played when a user makes a move and to notify them that their opponent has made their move. Smooth calming background music will also be played to make the experience enjoyable.

3.2 Usability and Humanity Requirements

The game must have an easy UI which is not difficult to use or learn. Both the touch interface and click interface should both work smoothly. There should a be a tab for instructions on both the rules of the game, but also how to play the game for those who might not know.

3.3 Performance Requirements

As this game will be played over the internet, it would be necessary for the speed of the information transfer to be very fast. It would not be acceptable if there was a long delay to record the move on another player's board. The game board would also need to be constantly updated. If the game logic knows the player's move, but it is not reflected on the board, that would be an issue. When the user is using the touch interface, it is important to ensure that the layout is such that the player does not accidentally make a move they did not intend to make. A certain level of precision is required to ensure this doesn't happen. Currently the game will be run on a local server, but in the future, a larger server would be required to to match the demand for the game and ensure that the server's capacity is sufficient. The server should be reliable and available at all times such that it does not crash leaving people without the ability to play the game.

3.4 Operational and Environmental Requirements

In order for the user to play this game, the system requirements are not very high. As long as they are using a web browser (preferably Google Chrome) and have a stable internet connection, the game should run smoothly. If the ping of a user's internet is very poor, it would cause some issues in communicating with the other player.

3.5 Maintainability and Support Requirements

The game should be easy to maintain as the only factor that would need constant care is the server. The game logic is not complicated and would be simple to fix if there were an error, but the server would be a challenge. If the server were to reach capacity, it would take some time to transfer to a larger server. It is currently impractical to use a large server for this product, but should the demand rise, larger servers can be prepared to transfer over to.

3.6 Security Requirements

Since this game is being run over the internet, there is always the matter of internet security. It is important that there be a certain level of encryption to ensure that the information being transferred is safe and protected.

3.7 Cultural Requirements

The product shall not be offensive to religious or ethnic groups. The game user interface should be language ambiguous. This means, even though the game is provided in English, it should be able to be navigated and played by someone who does not know how to read, write or speak English.

3.8 Legal Requirements

From a legal standpoint the main requirement is that we must not have access to any personal information from the users from when they connect to the internet. This game if it were to be rated by the ESRB, would have a rating of E for everyone.

3.9 Health and Safety Requirements

When the game is being played, to ensure there is no possible cause of epilepsy from the colors, very mild and neutral colors will be used to represent each player. Majority of the health and safety is on the owness of the user to ensure they are not walking and playing or are not playing the game for prolonged periods of time which could damage their health.

4 Project Issues

4.1 Open Issues

One issue we are concerned about is the adaptability of the user interface based on device the device they are on. Some issues could arise from touch or mouse input. Screen size of device could cause scaling problems such as having the part of the UI cut off. Lastly the user interface might not be the same based on where the game is accessed from because different browsers render things differently.

4.2 Off-the-Shelf Solutions

We will be utilizing four off the shelf solutions to assist this project

1. Moore server for web hosting
2. Gitlab for version control and issue tracking
3. Karma unit testing tool for verification
4. JsDoc for documentation generation

4.3 New Problems

Some new problems that have arisen during the development process are user problems such as miss click and poor internet connection. Additionally the planned server is not powerful enough to cope with our projected growth pattern.

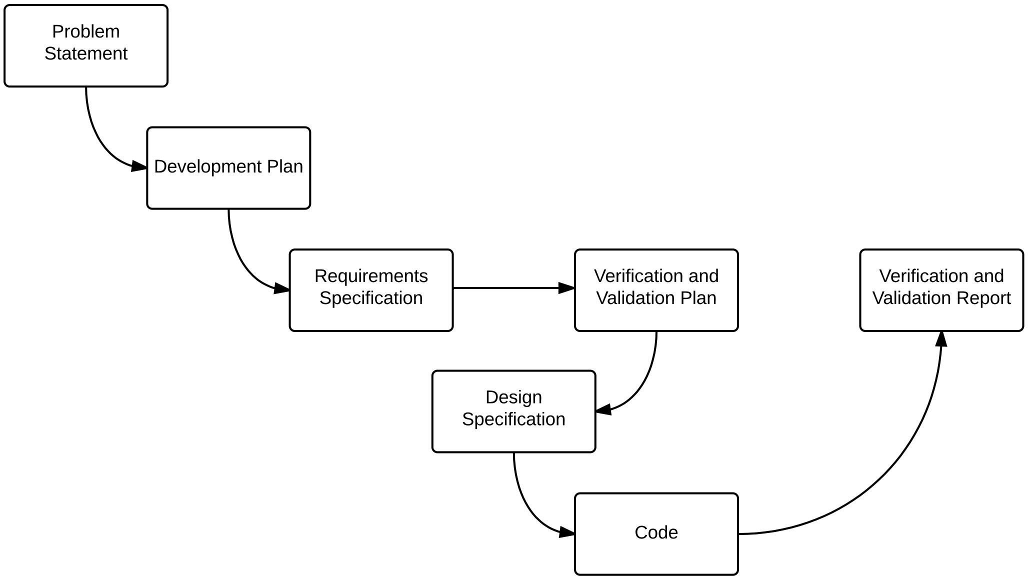
Problem Statement
Development Plan
Requirements Specification
Verification and Validation Plan
Verification and Validation Report
Design Specification
Code

Figure 1: Development Cycle

4.4 Tasks

The development cycle will follow the modified waterfall life cycle as detailed by Dr. Smith. Refer to Figure 1 (Smith, 2016). In verification and validation, a set of test cases to ensure that all the game features are working. Developing a thorough set of use cases as well as user testing, the game will be tested. The design of this game will follow the standard MVC game structure. The model will be located on the server, the view and controller will be on each individual device. The code has been broken down into various milestones. Please refer to the Gantt Chart. for further details. The final report will contain a detailed analysis of the result of the test cases and the reviews from the test users.

4.5 Migration to the New Product

The migration to this project will involve transferring the game logic to the server to allow for online game play if possible. The new product also will also change the graphics and the user interface. From a MVC standpoint, the view is the main change with the model shifting to the server. The controller will remain the same except making it friendly for mobile devices and all touch enabled devices.

4.6 Risks

There is a risk that the server that the game is hosted on could unexpectedly crash. There is also a security risk with a device being connected to the internet. We will ensure that only essential information is received and there are minimal permissions

4.7 Costs

We plan to complete this project with a zero dollar budget. All resources used to complete this project can be used free of charge. Additionally we plan to host this game on the McMaster Moore server which is also available to us for free.

4.8 User Documentation and Training

User documentation will be created in the form of game instructions. Comprehensive game rules have already already been created by mathwithbaddrawings.com (mathwithbaddrawings, 2016) . Our game instructions will be a version of theses rules with our own diagrams and explanations. There is no need for user training.

4.9 Waiting Room

After redeveloping the Ultimate Tic Tac Toe game, we plan to add Player vs Computer game mode. This AI should be able to intelligently perform moves based on the human player's previous move. Secondly User accounts which will be able to track player statistics such win ratio. Additionally we plan to add an online lobby to face strangers. Players will be matched based on their player statistics.

4.10 Ideas for Solutions

As discussed before we will be utilizing many off-the shelf solutions to help in the development process. We plan to Use CSS to style the game board.

To make the game language ambiguous we will be including many images into the game instructions. To make this game easy to maintain we shall be using Model-view-controller (MVC) software architectural pattern to make it modular.

References

kennycason. Ultimate tic tac toe, July 2010. URL https://github.com/kennycason/ultimate_tictactoe/.

mathwithbaddrawings. Ultimate tic tac toe rules, October 2016. URL https://mathwithbaddrawings.com/2013/06/16/ultimate-tic-tac-toe/.

James Robertson and Suzanne Robertson. *Volere Requirements Specification Template*. Atlantic Systems Guild Limited, 16 edition, 2012.

Dr. Spencer Smith. Lecture slides, October 2016. URL https://gitlab.cas.mcmaster.ca/smiths/se3xa3/.

5 Appendix

5.1 Symbolic Parameters

MAX_PLAYERS: Maximum number of concurrent players.

Table 1: **Revision History**

Date	**Version**	**Notes**
October 3	1.0	initial setup
October 5	1.1	updated Project Drivers content
October 6	1.2	added Functional Requirements
October 6	1.3	added Non-functional Requirements
October 7	1.4	updated Functional & Non-functional Requirements
October 7	1.5	Made notes for Project Issues
October 11	1.5	finished for Project Issues
October 11	2.0	Format document