**Requirements Document**

**Table of Contents**

[Modification History 3](#_Toc444352035)

[Domain Knowledge 4](#_Toc444352036)

[Glossary 4](#_Toc444352037)

[Interview with Client 5](#_Toc444352038)

[Functional Requirements 6](#_Toc444352039)

[Use Cases 6](#_Toc444352040)

[Use Case 1 6](#_Toc444352041)

[Use Case 2 6](#_Toc444352042)

[Use Case 3 7](#_Toc444352043)

[Use Case 4 7](#_Toc444352044)

[Use Case 5 8](#_Toc444352045)

[Use Case 6 9](#_Toc444352046)

[Use Case 7 9](#_Toc444352047)

[Use Case 8 10](#_Toc444352048)

[Use Case 9 10](#_Toc444352049)

[Use Case 10 11](#_Toc444352050)

[Use Case 11 12](#_Toc444352051)

[Use Case 12 12](#_Toc444352052)

[Use Case 13 13](#_Toc444352053)

[Use Case 14 13](#_Toc444352054)

[Non-Functional Requirements 15](#_Toc444352055)

[Cost Constraints 15](#_Toc444352056)

[Reliability 15](#_Toc444352057)

[Time Constraints 15](#_Toc444352058)

Modification History

* 2/23/2016 Original product design and documentation was modified to keep track of each player’s history and display upon user’s request.
* Modified the PowerPoint presentation which has to be re-designed due to lack of understanding and clarifying the material.
* 02/26/2016 Edited for consistency and clarity
* 3/26/2016 All documentation was modified in order to make it align with all the changes made to the program prototype. New use cases were implemented as new features were added, and the no longer applicable cases were deleted as well.
* 4/1/2016 All sequence diagrams for normal execution and exceptional scenarios were finished.

# Domain Knowledge

## Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| 6x6 Tic Tac Toe | A game that is played with two players, X and O, who take turns marking spaces usually in a 6x6 grid. |
| Graphical User Interface | Human-Computer Interface that uses windows icons and menus which can be manipulated by user input. |
| Platform | The hardware and support software which a program is intended to operate. |
| Client/Server System | The relationship between processes running on separate machines. A client initiates the dialog by sending a request to the server. |
| 4-in-a-row | When a player succeeds in placing four marks in a horizontal, vertical, or diagonal row. |
| MySQL | An application database that manages data and allows fast storage and retrieval of that data. |

|  |  |
| --- | --- |
| **Acronym** | **Meaning** |
| 3T | Tic-Tac-Toe |
| GUI | Graphical User Interface |
| SPMP | Software Project Management Plan |
| SQA | Software Quality Assurance |
| 3T FC | Tic Tac Toe Flow Chart |
| TC | Test Cases |
| AI | Artificial Intelligence |

## Interview with Client

**Location:** S708OneMainStreet

**Date:** 2/8/2015

**Time:** 1:30 PM

**Attendees:** Yuan Shengli

Project Management Committee (Eddie, Carrie, Jason, Ibra)

**Description:**

*Question 1*: How is the application going to work?

*Response*: The application will be a computer 6x6x4 tic-tac-toe game. The game will be played on a 6x6 grid, two players (one may be a computer) will take turns to place a game piece of their choice. The winner is the player with the most 4 stones in a horizontal, vertical, or diagonal row. A user can either play as a guest or as a registered user. The computer will keep track of each of the players score only if they are both registered users. If requested, the computer will display the history of the selected registered players. Users may select the level of difficulty and who goes first.

*Question 2*: What steps did you take to develop it?

*Response:* I took the classic Tic-tac-toe model, so there was not much of a concept to develop, but I extended it to be a board with six rows instead of three.

*Question 3:* What are the parameters of the game?

*Response:* Due to the nature of the 6x6 game board, there are more spaces available to be played on, but the rules of the 3x3 game board are still in play. By having this relationship between the game board and rules, it allows for the player to strategically plan his/her moves in more new ways. The winner is determined by the greatest amount of stone line- ups.

*Question 4:* What problems do you believe you’ll encounter while creating the application?

*Response:* I believe that there might be an issue in the process of deployment, however, I also believe that this it will not be a major issue there is still the possibility. Other than that, I do not know of any foreseeable issues.

# Functional Requirements

## Use Cases

### Use Case 1 (Register User)

|  |  |
| --- | --- |
| Goals of actor | Register new account. |
| Task | Give the user the option to set up new user name and password for new user to log into the game.  Prompt the user to create a security question and secret answer. |
| Preconditions | The information given must be new information.  User must not be an existing user |
| Exceptions | User enters existing username information  User enters not matching security answers. |
| Variation of action interactions | Display error message if username is already taken.  Display error message if both security answers do not match.  User looks at the help instructions |
| System change/production | Take the user to the main menu window so he/she can select from all the initial options |

### Use Case 2(Log in)

|  |  |
| --- | --- |
| Goals of actor | Log in to the game. |
| Task | Enter the user name and password to log into the game. |
| Preconditions | User has to be registered in order to log in. |
| Exceptions | .User name forgets log in information  User forgets security question with answer |
| Variation of action interactions | Display error message if user name or password is invalid  Give the option to reset password  User looks at the help instructions |
| System change/production | User decides to reset password, security must be answered. |

### Use Case 3 (Play as Guest)

|  |  |
| --- | --- |
| Goals of actor | Play a game as a guest. |
| Task | Start a new game without logging in. |
| Preconditions | User cannot already be logged in. |
| Exceptions | User is already logged in and tries to play as guest |
| Variation of action interactions | Display error message if user is logged in ( are we gonna give the option to log user out or log user out automatically )  User looks at the help instructions |
| System change/production | User chooses to log out and play as guest. |

### Use Case 4 (Reset Password)

|  |  |
| --- | --- |
| Goals of actor | Reset password. |
| Task | Reset user password. |
| Preconditions | User must already be registered. |
| Exceptions | User is given an error message of invalid username given  If answer to security question is invalid display another message |
| Variation of action interactions | Username recovery option is given in case user has forgotten it.  (Create new account if security question is not answered correctly.??)  User looks at the help instructions |
| System change/production | Retrieve user’s information from database when and if the security question is answered correctly.  Take user to main window for the option to log in again. |

### Use Case 5(View Statistics)

|  |  |
| --- | --- |
| Goals of actor | View play statistics. |
| Task | Displays user Win/Loss record. |
| Preconditions | User must be logged in. |
| Exceptions | New user with no play history requests to view statistics.  Display error message |
| Variation of action interactions | User play the game without looking the statistics.  User chooses the help option to receive instructions  User looks at the help instructions |
| System change/production | User is given option to end or continue game. |

### Use Case 6 (Player vs. Player)

|  |  |
| --- | --- |
| Goals of actor | Select game mode. |
| Task | Choose to play the game against another player. |
| Preconditions | User has to be a guest or logged in.  Accept input for choosing the play as guest or registered user mode from player one |
| Exceptions | User two attempts to change play mode |
| Variation of action interactions | User is given an error message and given the option to end game and start new one instead.  User chooses the help option to receive instructions |
| System change/production | End the current game and start new one. Let player two become player one to choose mode. |

### Use Case 7 (Player vs. AI)

|  |  |
| --- | --- |
| Goals of actor | Select game mode. |
| Task | Choose to play the game against AI. |
| Preconditions | User must either log in or choose to play as a guest. |
| Exceptions |  |
| Variation of action interactions | User chooses to exit the game which is taken as a loss.  User chooses the help option to receive instructions |
| System change/production | Current window is closed and score is updated by adding winner. |

### Use Case 8 (Difficulty Level)

|  |  |
| --- | --- |
| Goals of actor | User must choose from three modes of difficulty |
| Task | Adjust the AI difficulty level. |
| Preconditions | User has to be already logged in or be a guest player |
| Exceptions |  |
| Variation of action interactions | User chooses to exit the game which is taken as a loss.  User looks at the help instructions |
| System change/production | Current window is closed and score is updated by adding winner. |

### Use Case 9 (Select Game piece)

|  |  |
| --- | --- |
| Goals of actor | Select between X or O. |
| Tasks | Give the user the option to choose between X or O. |
| Preconditions | User must be logged in, game mode and difficulty level must have been chosen already. |
| Exceptions |  |
| Variation of action interactions | User chooses to exit game before starting to play to change stone and this is taken as loss  User looks at the help instructions |
| System change/production | Current window is closed and score is updated by adding winner. |

### Use Case 10 (Select First Player)

|  |  |
| --- | --- |
| Goals of actor | Select which player goes first. |
| Tasks | Let X or O choose who is doing the first move in either play mode. |
| Preconditions | AI must play as X by default and user as O, the only selection to be made is who goes first. |
| Exceptions |  |
| Variation of action interactions | User chooses to exit game to change piece.  User looks at the help instructions |
| System change/production | Current window is closed and score is updated by adding winner. |



### Use Case 11(Exit )

|  |  |
| --- | --- |
| Goals of actor | To exit current window. |
| Task | Give the user the option to close current window at any moment |
| Preconditions | Main window has to opened |
| Exceptions |  |
| Variation of action interactions | User exits from all windows to go back to the main window.  User looks at help instructions in current window |
| System change/production | User is taken to previous windows |

### Use Case 12(Log Out)

|  |  |
| --- | --- |
| Goals of actor | Logout |
| Tasks | Log user out so another user can log in, or to allow user to play as a guest. |
| Preconditions | Must be currently logged in. |
| Exceptions | User tries to log out while he/ she is not logged in. |
| Variation of action interactions | Display an error message to user and give the option to log in.  User looks at the help instructions |
| System change/production | Log out current user to let new user log in.  Let current user log in. |

### Undefined use cases //for now

### Use Case 14

|  |  |
| --- | --- |
| Goals of actor | Quit  Like use case 11 this case is present in previous uses cases. And the quit option is only one action and if we have to implement it how it should look like? Unless we have to create some scenarios where players are present and want to quit. And as you can see on the precondtions the game cannot be in progress. |
| Task | Exits the application |
| Preconditions | A game cannot be in progress. |
| Exceptions | Can’t quit the application if there is a game in process. Will request confirmation to terminate game. |
| Variation of action interactions |  |
| System change/production |  |

### Use Case 15

|  |  |
| --- | --- |
| Goals of actor | Select game mode. |
| Task | Choose to play the game against another player. |
| Preconditions | User must either log in or choose to play as a guest. |
| Exceptions | Can be used in a multi-player setting or against the AI |
| Variation of action interactions |  |
| System change/production |  |

# Non-Functional Requirements

## Cost Constraints

No cost constraints are foreseeable in the near future. Due to this lack of constraint, we are sure no cost will be generated and also no downtime will be expected for this project.

We are using GitHub and Drobox to keep track of our documentation and source code. A log file is generated to show all of changes and commits on the master branch. The branch has been replicated as a mirror branch where developers can make changes and submit their code to be reviewed before being merged with the main branch.

## Reliability

The software should be usable, without any operational faults, for the period of time between its deployment and the following three years. Based on the schedule and following phases that we have use, the product should be able to be completed at the respected date.

## Time Constraints

Most meetings took place either after CS 3420 class or in Google Hangouts, Skype calls, and email in order to accommodate all group members with a schedule where we could all meet without running into any time constraints whilst building the application and filling out the required documents for the project.