**Requirements Document**

*Team NULL Terminators*Usman Tamanna – Team LeadBryan Nafager – Secretary  
Joe Kanter – SQA Representative   
Arubaleze Ugochukwu   
Mervyn Cabio  
Purja Bishal

**Table of Contents**

[Document Overview & Modfication History 3](#_Toc246755789)

[Domain Knowledge 4](#_Toc246755790)

[Glossary 4](#_Toc246755791)

[Interview with Client/Questionear 5](#_Toc246755792)

[Functional Requirements 6](#_Toc246755793)

[Use Case 1 6](#_Toc246755795),7

Use Case 2……,,,,…………………………………………………………………8, 9

[Non-Functional Requirements](#_Toc246755801) 10

[Cost Constraints](#_Toc246755802) 10

[Reliability](#_Toc246755803) 10

[Time Constraints](#_Toc246755805) 10

**Document Overview**

This is the requirement documentation for {NULL Terminators} 5x5 TicTacToe game, it covers what the game is planned to do in functional requirement; also it examines other nonfunctional requirements which are cost constraints, reliability and time constraints. As well as interviews with team members and use cases with descriptions.

**Modification History**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Date/Author** | **Reviewed By** | **Changes** |
| 1.0 | 2-01-2013 Arubaleze Ugochukwu | Usman Tamanna | Added document overview Updated glossary |
| 1.1 | 2-07-2013 Arubaleze Ugochukwu | Usman Tamanna Bryan Nafegar | Added the interview to the document Updated the glossary |
| 1.2 | 2-12-2013 Usman Tamanna | Bryan Nafegar Mervyn Cabio | Added functional and nonfunctional requirements of the game |
| 1.2.1 | 3-05-2013 Arubaleze Ugochukwu | Joe Kanter Usman Tamanna Purja Bishal | Updated glossary Fixed errors in documentation |
| 2.0 | 4/7/2013  Arubaleze Ugochukwu | Usman Tamanna Bryan Nafegar  Mervyn Cabio  Purja Bishal  Joe Kanter | Added Correct Use Cases for 1 and 2 players with Description |

# Domain Knowledge

## Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| C# | ‘C sharp’ programming language used to write, compile and run the Game. |
| Algorithm | A method by which the program can spot a winning combination and alert the user and end the game when one appears. |
| Hardware | The physical components of a computer |
| Software | The program(s) used to direct the operation of a computer. |
| Strategy Game | A game in which the players' decision-making skills have a high significance in determining the outcome. |

|  |  |
| --- | --- |
| **Acronym** | **Meaning** |
| AI | Artificial Intelligence – The computer created algorithm for single player games. |
| GUI | Graphic User Interface – sums up the user view which enables them to interact with the game. |
| F.I.I.F | Full Implementation in Future |
| P1 | Player 1 |
| P2 | Player 2 |
| TTT | Tic-Tac-Toe |
| UML | Unified Modelling Process |

## Interview with Client/Questioner/…

**Location: Library RM 430**

**Date: 2/5/2013**

**Time: 2:40pm**

**Attendees: Usman Tamanna, Bryan Nafegar, Joe Kanter**

**Description:**

*Question 1*: How do you want players to provide input?

*Response*: The players will provide input by clicking on the grid which will then display the user’s choice of X or O (depending on their initial pregame selection).

*Question 2*: Are the AI be dependent on the history of the play, and not just on the positions of the tokens before and after the move?

*Response:* The AI can be dependent on the history of the play but it is not required. Whichever algorithm makes smart and fast decisions should be implemented.

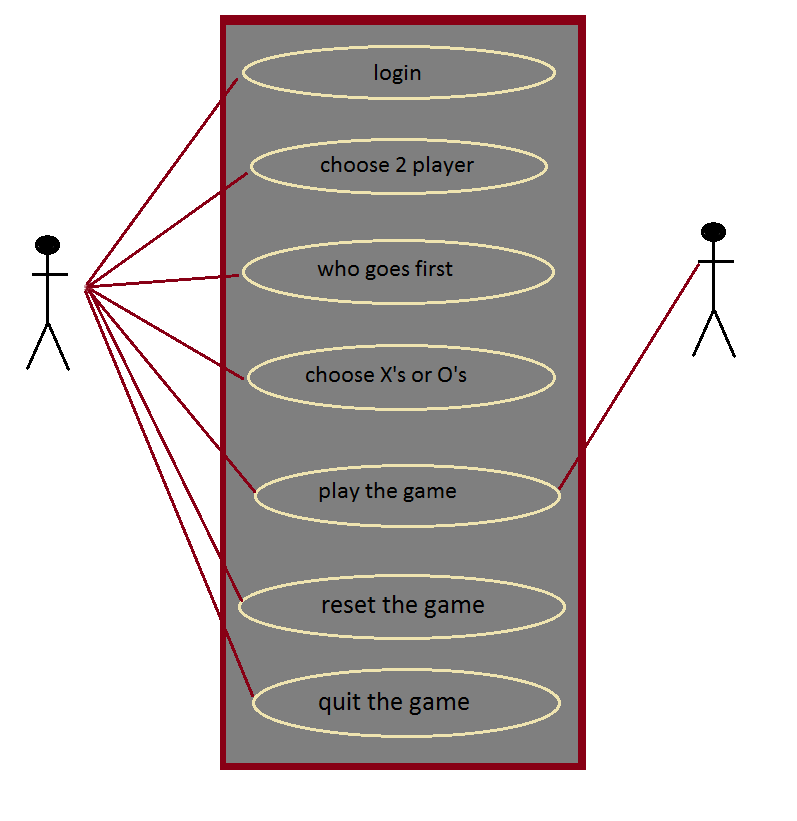
*Question 2*: What happens after the game ends?

*Response:* The game should show a game over screen also the game should increment the user’s wins or losses in the history. In addition the game should reset and present the user with a new game.

# Functional Requirements

## Use Cases

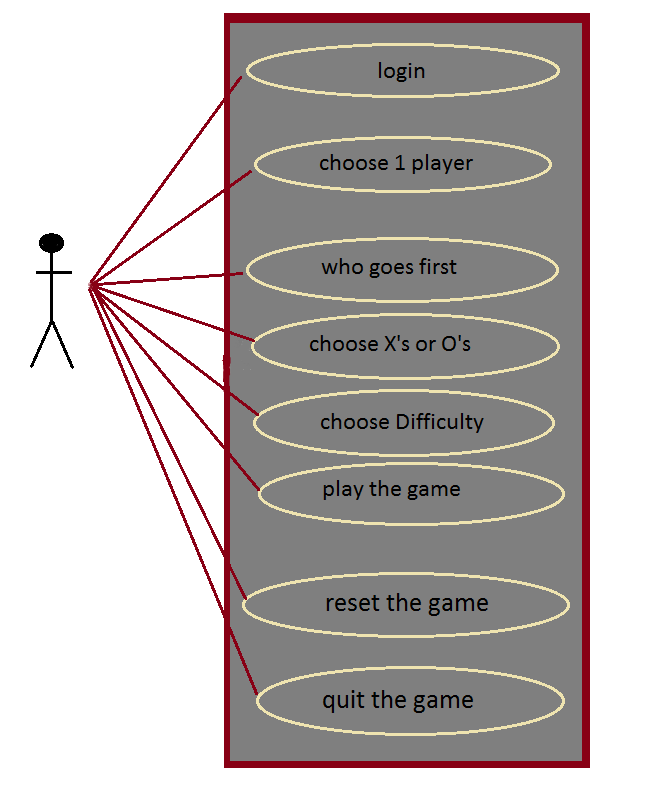
### Use Case 1 for 2 player



## 

|  |
| --- |
| Description of the use case 1 for 2 players |
| Login to the game with a username and password (user can create an account if he/she does not have one) |
| Choose the 2 player use case, notifies the game that a second player will be playing. |
| Choose who goes first allows the player to decide who gets the first turn |
| Choose between X’s or O’s lets the players choose their preferred marking shape |
| Play the game allows P1 and P2 to see the game board and commence playing |
| Reset the game allows the user to reset the game and start a new one |
| Quit the game terminates and closes the game state |

## Use case 2 for 1 player



|  |
| --- |
| Description of the use case for 1 player |
| Login to the game with a username and password (user can create an account if he/she does not have one) |
| Choose the 1 player use case, notifies the game that the AI will be playing. |
| Choose who goes first allows the player to decide who gets the first turn |
| Choose between X’s or O’s lets the player chose his/her preferred marking shape |
| Choose Difficulty allow the player to set the AI difficulty |
| Play the game allows P1 to see the game board and commence playing |
| Reset the game allows the user to reset the game and start a new one |
| Quit the game terminates and closes the game state |

# Non-Functional Requirements

## Cost Constraints

There are no Cost Constraints or expenses expected for the duration of this project.

## Reliability

The code will be tested meticulously for any bugs or problems. The proposed 2 player version of the game will work 100% bug free.

## Time Constraints

We expect the game to run and respond in a timely manner with no response lag.