Software Requirements Specification

SRS for

Tic Tac Toe Game

Prepared by: Sampada Bhatnagar

Institution: GZSCCET, MRSPTU

March 2017

**Table of Contents 1**

**1.** **Introduction 2**

1.1 Overview 2

1.2 Purpose 2

1.3 Scope 2

1.4 Project Functionality 2

1.5 Operations Performed 3

1.6 Acronymns and Abbreviations 3

**2.** **Functional Requirements 3**

2.1 Basic Inclusions 3

2.2 High Level Functions 3

**3.** **Non Functional Requirements 4**

3.1 User Interface 4

3.2 Software Interface 4

3.3 Hardware Interface 5

3.4 Communication Interface 5

3.5 Manpower 5

3.6 Performance and Security Requirements 5

3.7 Time Computaion 5

3.8 Software System Attributes 5

**4.** **Design Phase 6**

4.1 Use Case Diagram 7

4.2 Flowchart 7

**I. INTRODUCTION**

A. **OVERVIEW**

Tic-tac-toe (also known as noughts and crosses or Xs and Os) is an interactive two player game, X and O, who take turns to mark the spaces in a 3×3 grid. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game.In this desktop software,command prompt is opened on both client and server systems.Then their port and IP are inputted. After successful connection with the server, both users have the live gaming window opened and they begin to play. When 1 player wins and opponent loses,Terminal is closed or another game is started.Thus connection gets terminated and formed each time.

B. **PURPOSE**

Due to the simplicity of tic-tac-toe analogy; it is often used as a pedagogical tool for teaching the concepts of good sportsmanship , via experimental data collected from the players of the game, and in the branch of artificial intelligence that deals with the searching of game trees.

C. **SCOPE**

Xs and Os has several variants, namely:

(i) Quantum Tic Tac Toe: borrows three phenomena from quantum physics: superposition, entanglement and collapse.

(ii) Scrabble: Pick15 is isomorphic to noughts and crosses as it depicts the game by mapping it onto a magic square.

(iii) Treblecross:1 D version of the traditional play.

(iv) Misere Tic Tac: an inverse concept of losing the game exists.

(v) Notakto:It is an impartial or neutral variant and is often viewed as a disjunctive game.

(vi) Hollywood Squares: The only which doesn’t degenerate into some deterministic optimal strategy.

D**. PRODUCT FUNCTIONALITY**

(i) Allows player to play with AI i.e provision of a player and computer pitted against each other.

(ii) Permits the player to play against other player.

E. **OPERATIONS PEFORMED**

(i) Both players choose their respective symbols to mark.

(ii) Player 1 starts the turn by placing his/her symbol on any of the nine squares.

(iii) Player 2 makes his turn by placing his/her symbol on the empty squares.

(iv) Both players make their turns alternately.

(v) If any player gets the three respective symbols in a horizontal, vertical or diagonal row;he/she wins the tic tac toe game.

F. **ACRONYMS AND ABBREVIATIONS**

AI: Artificial Intelligence

D: Dimensional

SRS: Software Requirement Specifications

IDE: Integrated Development Environment

JDK: Java Development Kit

JVM: Java Virtual Machine

( ): Function

**II. FUNCTIONAL REQUIREMENTS**

1. **Basic needs of the game**
   1. A game board will exist in a 3x4 grid.
   2. The ally must be depicted in blue, thereby making enemy in red
   3. UnableToCommunicate = 1 in case of error
   4. Buffered image: Image resources are stored.
   5. Variable are made private i.e coding begins from the lowest visibility level.
   6. Text Antialiazing: Removes pixelation and used with Image Observer
2. **High Level Functions**
   1. ListenForServerRequest( ) : an integral modules for Input & Output stream of data while ensuring request from the server side socket is accepted.
   2. Painter( ) & LoadImages( ) : to change any Paint components i.e graphical aspects like border
   3. Runnable( ) & Run( ) : provides a common protocol for objects that wish to execute code while they are active.
   4. Bind( ) :assigns a local socket address to the port which is a communication point in networking.
   5. InitializeServer( ) : Responds to client request by initializing server side port.
   6. Connect( ) and Tick( ) : For Exception Handling and other connection establishment functionalities.
   7. CheckToWin( ) : Computes the winner out of the 2 players
      1. CheckForEnemyWin( ) : Determines whether enemy has won or not.
      2. CheckForTie( ) : Determines whether neither player won or not.

**III. NON-FUNCTIONAL REQUIREMENTS**

A. **User Interface**

The screens will be as follows:

1. Command Prompt: For textual input

1.1 The IP address of the console to be entered by players

1.2 The ports to be inputted in order to play the game

2. Game Screen: For Graphical Input and Output

2.1 The players see their respective boards on entering the gaming arena.

2.2 The basic interface will be same, with the name of the game being on top of

the screen, and their potential actions.

2.3 The remaining number of those actions are displayed in the form of symbols

Xs and Os as and when the game progresses.

2.4 Message Setup: Result that declares the winner and loser via dialogues.

B. **Software Interface**

|  |  |
| --- | --- |
| 1. OPERATING SYSTEM | Windows 98 & above |
| 2. TOOLS UTILIZED | Command Prompt |
| 3. IDE | Eclipse (Juno, Kepler versions) |
| 4.VIRTUAL MACHINES | JVM: J2SE 1.4.2, J2SE 5 |
| 5. KITS | JDK- compiler and interpreter |
| 6. INSTRUCTION SET | 32 bit, 64 bit |

C. **Hardware Interface**

A desktop/laptop having the following specifications:-

|  |  |
| --- | --- |
| 1. RECOMMENDED DISK  SPACE | 1 GB free |
| 2. PROCESSOR | Intel core i3, i5, i7 support |
| 3. HARD DISK DRIVE | 1 TB atleast |
| 4. GRAPHICS CARD | NVIDIA |
| 5. NETWORK CARD | Realtek Ethernet Integrated Controller (10/100 or 10/100/1000 NIC) |
| 6. CHIPSET | Mobile Intel HM65 Express |

D. **Communication Interface**

The intranet communication will be through TCP/IP protocol suite for this turn based game, as opposed to UDP for real time games, & the networking aspects cover IP addresses, ports and sockets.

E. **Manpower**

A single member was required in developing the above game i.e no requirement of a Team

F. **Performance and Security Requirements**

The system should respond to each user input within 2 seconds. Security will not be a concern for this project since there is no sensitive data being stored.

G. **Time Computation**

After the application is launched, it will take fewer than 5 seconds for the player to load

their board and begin playing the game. This ensures that users do not lose interest.

H. **Software System Attributes**

1. Usability: Familiar and easy to use interface for players

2. Reliability: The software will be able to run 99% of the time when launched.

3. Maintenability: Once installed, the game and data system will not need to be maintained at

scheduled intervals or any given stipulated time i.e regular maintenance is not needed.

4. Portability:Supportability exists for all systems.

**IV. DESIGN PHASE**

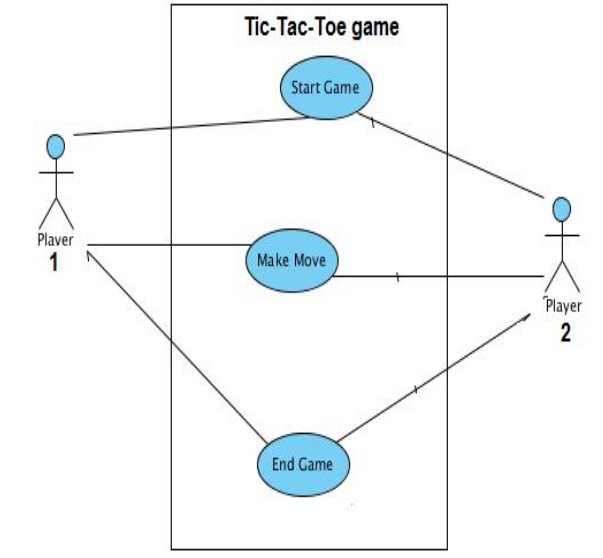


FIGURE 1: USE CASE DIAGRAM

Text Description

U1: Use Case 1: Player 1 wins

U2: Use Case 2: Player 2 wins

U3: Use Case 3: Results in a tie

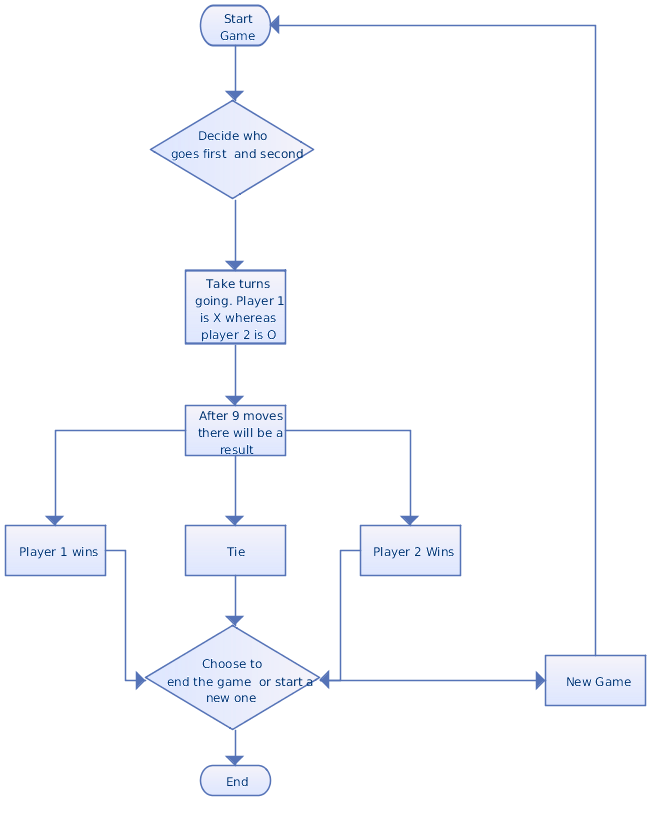


FIGURE 2: FLOWCHART