Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 11/03/2014 | 1.0 | Write specifications for the TicTacToe Project | Eren Sezener |
| 11/03/2014 | 1.1 | Write requirements for the TicTacToe Project | Deniz Sökmen |
| 11/03/2014 | 1.2 | Write requirement specifications | Deniz Sökmen |
| 12/03/2014 | 1.3 | Update user & system requirements | A. Emre Ünal |
| 12/03/2014 | 1.4 | Update system requirements specifications | Deniz Sökmen |
| 16/03/2014 | 1.4.1 | Update user requirements codes | A. Emre Ünal |
| 18/03/2014 | 1.5 | Add UML Diagrams | Eren Sezener |

**TABLE OF CONTENTS**

Revision History 1

1 INTRODUCTION 3

1.1 Document overview 3

1.2 System overview 3

1.3 Abbreviations 3

1.4 References 3

1.5 Conventions 3

2 REQUIREMENTS 4

2.1 User Requirements Specification 4

2.2 System Architecture 4

2.3 Use Cases and Usage Scenarios 5

2.4 System Requirements Specification 5

# INTRODUCTION

## Document overview

This document presents the software requirements specifications of TicTacToe game software development project.

It describes:

* Overview of software system to be developed
* Conventions to be used during the development of the project
* Requirements of functionalities
* Architecture of the software
* Use cases and usage scenarios,
* The compliance of requirements to user's needs
* The relative importance and precedence of requirements

## System overview

Tic-Tac-Toe game has been Turkey's national sport since Sir Muharrem Tic-Tac-Toe invented the game in around 500 BC. Although the game is popular in many Kahvehanes (Turkish Cafés), the game is rarely played on the Internet due to the lack of high quality software. TicTacToe Project aims to fill this gap by enabling users to play Tic-Tac-Toe against each other. The game will have peer-to-peer architecture and users will connect to each using their IP addresses.

## Abbreviations

The TicTacToe game software project: “the game” or “the software”

The TicTacToe game software project’s GitHub repository page: “the repo”

## References

|  |  |  |
| --- | --- | --- |
| # | Document Identifier | Document Title |
| [R1] | SDPv1.2 | Software Development Plan of the TicTacToe game, version 1.2. |

## Conventions

Requirements listed in this document are constructed according to the following structure:

Requirement Id

Requirement title

Requirement description

Requirement version

# REQUIREMENTS

## User Requirements Specification

SRS-REQ-101 LOCAL

Playing a local game

The user should be able to play a local game, with 2 players in one computer.

V1.0

SRS-REQ-102 JOIN

Joining a remote game

The user should be able to join a game through the network, to play with a remote opponent.

V1.0

SRS-REQ-103 HOST

Hosting a remote game

The user should be able to host a game through the network, to play with a remote opponent.

V1.0

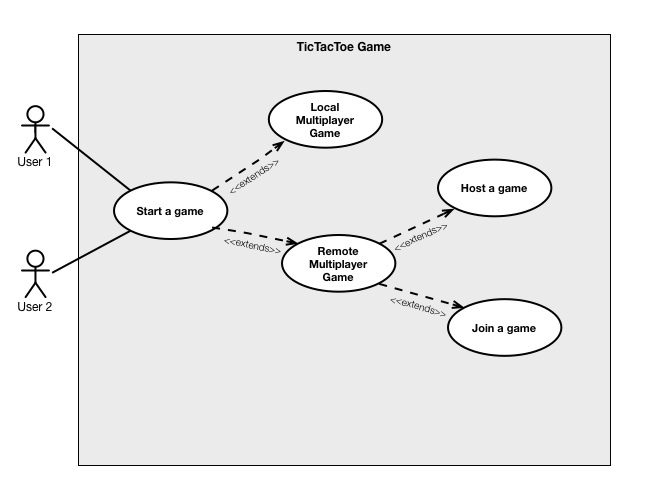
## System Architecture

In the main screen, there are two buttons: “Local Game” and “Network Game”. When the user clicks the local game button, TicTacToe game shall start directly and two players shall be able to play on the same computer without any internet connection.

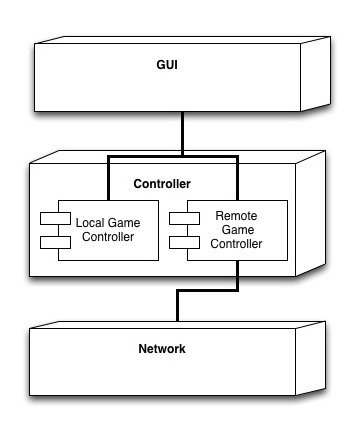
If the user chooses “Network Game”, another screen with two buttons shall be displayed and labelled as: “Join a Game” and “Host a Game”. If there is no internet connection, those buttons shall not be clickable. If the user chooses “Host a Game”, the game shall try to create a game over the network and wait for an opponent, until then the game shall be blocked in a screen with “Waiting for an opponent” message. If the user chooses “Join a Game”, there shall be a pop-up asking for the IP address and port of the host the user wishes to connect to. If the connection is successful, both the host and the client shall be redirected to TicTacToe screen.

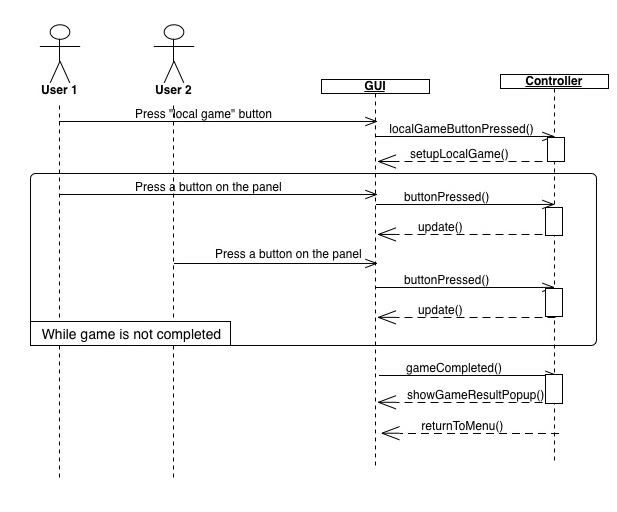
Finally, in the TicTacToe screen, there are 9 buttons indicating where a mark should be put and a label that gives information about whose turn it currently is. In both network and local game, whenever a button is clicked, the game shall check if the game is over. If the game is over, the players shall get a pop-up message announcing the winner and they shall be redirected to the main screen.

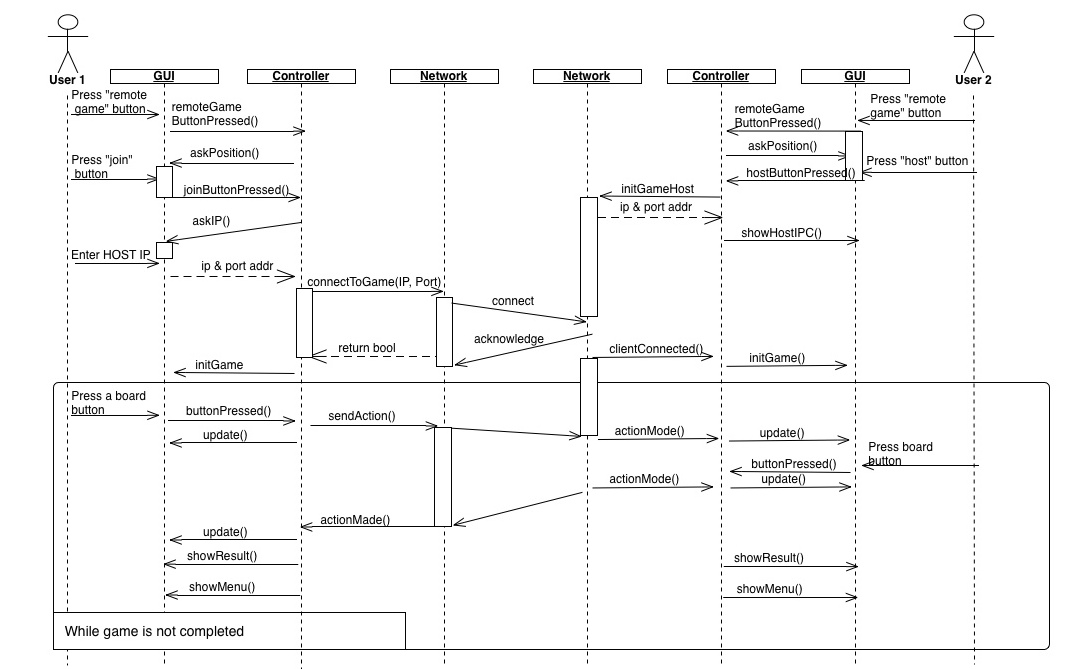
## Use Cases and Usage Scenarios



Use Case

System Architecture

Sequence Diagram for Local Game



Sequence Diagram for Remote Game

## System Requirements Specification

**Game Requirements**

SRS-REQ-001

Creating a Local Game

Whenever the user clicks the “Local Game” button, a game for two players on the same computer should be able to be created successfully without an internet connection.

V1.0

SRS-REQ-002

Game Rules

The game shall check for these conditions whenever a button is clicked:

* If it is the player’s turn
* If the game is over
* If the button is not occupied

V1.0

SRS-REQ-003

Ending the Game

If the game is over in both network and local game, both of the players shall get a pop-up message indicating the winner and shall be redirected to the main screen.

V1.0

**Network Requirements**

SRS-REQ-004

Hosting a Game

If the user clicks “Host a Game” button, the game shall first check the internet connectivity and then create a socket for listening connections. The game shall be blocked until there is a connection.

V1.0

SRS-REQ-005

Joining a Game

If the user clicks “Join a Game” and enters an IP address, the game shall try to connect to the specified address. After 5 seconds, if the connection is still not successful, the game shall drop the connection trials and send an error message to the user indicating the failure of connection.

V1.0

SRS-REQ-006

Playing the game over network

In the network mode, every action of the player shall be sent to the server through the network.

V1.0

SRS-REQ-007

Maintaining the Connection

Every time a game action is sent through the network, the game shall check if the action has been delivered successfully. Therefore, if there is an error with the connection, the game shall wait for 5 seconds and if there is still a problem, both of the players shall be redirected to the main screen with a pop-up message indicating the network problem.

V1.0