 Cairo university Faculty of Engineering

Embedded systems project:

TicTacToe

Presented by:

CodeCrafters

|  |  |  |
| --- | --- | --- |
| Name | SEC. | ID |
| Mohammed Mohsen | 3 | 9220737 |
| Mahmoud Omar | 3 | 9220787 |
| Mohammed Hashim | 3 | 9220748 |
| Mohammed Hisham | 3 | 9221105 |
| Abd ElRahman Badawy | 3 | 9220426 |

Contents

[1. Introduction 3](#_Toc170354100)

[2. Overall Description 3](#_Toc170354101)

[3. Specific Requirements 4](#_Toc170354102)

### 1. Introduction

#### 1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to detail the functional and non-functional requirements for the TicTacToe game application. This document is intended for developers, testers, and stakeholders involved in the project.

#### 1.2 Scope

This SRS covers the development of a TicTacToe game application, which includes user registration, login, profile management, game play, and history tracking functionalities.

#### 1.3 Definitions, Acronyms, and Abbreviations

* SRS: Software Requirements Specification
* GUI: Graphical User Interface
* AI: Artificial Intelligence

#### 1.4 Overview

The TicTacToe game application will allow users to register, login, manage their profiles, play TicTacToe games against an AI or another player, and view their game history.

### 2. Overall Description

#### 2.1 Product Perspective

The TicTacToe game application is an independent system designed to provide a digital version of the classic TicTacToe game with additional features such as user authentication, profile management, and game history tracking.

#### 2.2 Product Functions

* User Registration (Sign Up)
* User Login (Sign In)
* Profile Management
* Playing TicTacToe (Single Player vs AI and Multiplayer)
* Viewing Game History

#### 2.3 User Classes and Characteristics

* **Registered User**: Can sign in, play games, view and edit their profile, and see their game history.
* **Guest User**: Limited access, primarily for playing as a single-player without saving progress.

#### 2.4 Operating Environment

* The application will run on Windows, macOS, and Linux operating systems.
* The application will be developed using Qt and C++.

#### 2.5 Design and Implementation Constraints

* The application should use the Qt framework for the GUI.
* The application should be implemented in C++.

#### 2.6 Assumptions and Dependencies

* Users will have internet access for initial registration.
* Users will have basic knowledge of using computer applications.

### 3. Specific Requirements

#### 3.1 Functional Requirements

##### 3.1.1 User Registration (Sign Up)

* The system shall provide a registration page.
* The registration page shall include fields for username, password, age, gender, and phone number.
* The system shall validate the input fields.
* Upon successful registration, the user shall be redirected to the login page.

##### 3.1.2 User Login (Sign In)

* The system shall provide a login page.
* The login page shall include fields for email and password.
* The system shall authenticate the user based on the provided credentials.
* Upon successful login, the user shall be redirected to the main menu with profile, game, and history tabs.

##### 3.1.3 Profile Management

* The system shall provide a profile page.
* The profile page shall display the user's username, age, gender, phone number, games played, wins, and losses.
* The user shall be able to update their personal information.

##### 3.1.4 Playing TicTacToe

* The system shall provide a game page with options to start a new game or reset the current game.
* The game page shall allow the user to choose between single-player (against AI) and multiplayer modes.
* The game shall follow the standard TicTacToe rules (3x3 grid, win by completing a row, column, or diagonal).
* The game shall display the current result (win, loss, or draw).

##### 3.1.5 Viewing Game History

* The system shall provide a history page.
* The history page shall display a list of past games.
* Each past game entry shall show the final situation of the game (board state at the end).

#### 3.2 Non-Functional Requirements

##### 3.2.1 Usability

* The application shall have an intuitive and user-friendly interface.
* The application shall provide appropriate feedback for user actions (e.g., error messages, success messages).

##### 3.2.2 Performance

* The application shall load the main menu within 2 seconds after a successful login.
* The application shall respond to user inputs within 1 second.

##### 3.2.3 Reliability

* The application shall have an uptime of 99.5%.
* The application shall handle up to 100 concurrent users.

##### 3.2.4 Security

* The application shall use encryption for storing user passwords.
* The application shall validate all user inputs to prevent SQL injection and other attacks.

#### 3.3 System Behavior

##### 3.3.1 Game Rules

* The game shall be played on a 3x3 grid.
* Players take turns to mark a cell in the grid.
* A player wins by placing three of their marks in a horizontal, vertical, or diagonal row.
* The game ends in a draw if all cells are filled without a winning combination.

##### 3.3.2 System Flow

1. User opens the application.
2. User chooses to sign in or sign up.
3. If sign up, user completes registration and is redirected to the login page.
4. User logs in and is redirected to the main menu.
5. User navigates to profile, game, or history tabs as desired.
6. In the game tab, user chooses to play against AI or another player.
7. Game proceeds according to standard TicTacToe rules.
8. User can start a new game or reset the current game.
9. User can view past games in the history tab.

#### 3.4 Performance Requirements

* The system shall handle up to 100 concurrent users.
* The system shall have an average response time of less than 1 second for user inputs.
* The game page shall update the game board within 0.5 seconds of a move being made.