Snake Game - Project Documentation

1. Project Overview

1.1 Introduction

The Snake Game is a simple yet classic arcade game where a player controls a snake that moves in four directions to eat food and grow in length. The objective is to avoid collisions with the walls and the snake's own body while achieving the highest possible score.

This project is developed as part of a software development course, focusing on **Agile development (SCRUM framework)**, **JIRA for project management**, and **basic software testing**. The game is built using **JavaScript**, stored in a **GitHub repository**, and deployed online for accessibility.

1.2 Project Goals

Develop a **functional Snake Game** with simple yet engaging gameplay.

Apply **Agile project management** practices using JIRA.

Implement **basic unit tests** to validate key functionalities.

Document all aspects of the project, including requirements, design, and implementation.

2. Documentation

2.1 Technologies Used

Programming Language: JavaScript

Development Tools: JIRA, GitHub, VS Code

Testing Framework: Jest (JavaScript)

Diagram Tool: draw.io

2.2 Project Structure

Source Code: Contains the main game logic, UI, and assets.

Documentation: Contains README, diagrams, and requirement specifications.

3. Functional Requirements

3.1 Core Game Features

1. **Game Controls**: The player can move the snake using arrow keys or touch controls.
2. **Food Collection**: The snake eats food to grow longer.

3. **Score Tracking**: The game keeps track of the player's score in real-time.
4. **Game Over Condition**: The game ends when the snake collides with itself or the game boundaries.
5. **Restart Option**: Players can restart the game after a game-over event.
6. **Pause & Resume**: The game can be paused and resumed at any time.
7. **High Score Saving**: The highest score achieved is stored and displayed.
8. **Leaderboard**: Store and display top scores.

4. Non-Functional Requirements

4.1 Performance Requirements

The game should run **smoothly without noticeable lag** on both desktop.

Game response time should be less than 100ms per keypress.

4.2 Usability Requirements

The game should be **easy to understand and play** without requiring additional instructions.

Controls should be **intuitive** and responsive.

UI should be **visually appealing and accessible** on different screen sizes.

4.3 Reliability & Maintainability

The game should not crash during normal operation.

Bug-free gameplay after final testing before deployment.

4.4 Security Requirements

Ensure **safe deployment** with no exposed vulnerabilities.

5. Deployment & Testing

5.1 Deployment Plan

The project will include a **README file** with instructions on how to play.

5.2 Testing Plan

1. **Unit Testing**: Validate game mechanics (movement, collision detection, score tracking).
2. **Integration Testing**: Ensure UI components interact correctly with the game logic.
3. **Performance Testing**: Measure frame rates and responsiveness on different devices.

6. Conclusion

This project provides an excellent opportunity to apply **Agile development methodologies** in a real-world scenario. By developing a classic **Snake Game**, we have explored **project management with JIRA, version control with GitHub, software testing, and deployment**. The project is structured for easy extension, allowing future improvements such as leaderboards, different game modes, and multiplayer features.

Project Members Credentials

Ilhan Dag, 37689