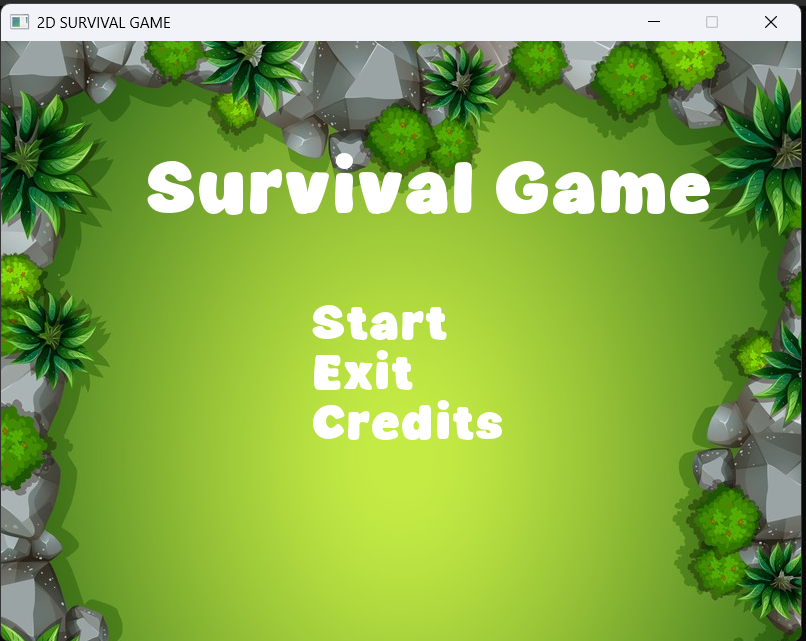
**2D SURVIVAL GAME**

**IN**

**C++ WITH SFML**



**Submitted by: David Rai**

**Submitted to: Aashish Acharya**

**Date of Submission: 04/21/2025**

**TABLE OF CONTENTS**

**Page no.**

**1. Introduction …………………………………………………………………3**

**2. Features………………………………………………………………………3**

**3. Game Logic Flow…………………………………………………………….4**

**4..Technical Details……………………………………………………………..4**

**5. Dependencies…………………………………………………………………5**

**6.Assest Requirement…………………………………………………………..5**

**7. Known Limitation……………………………………………………………6**

**8. Future improvement…………………………………………………………6**

**9. Game Screenshots……………………………………………………………(6-7)**

### **Introduction**

### **1.1 Project Overview**

A 2D survival game developed using **SFML (Simple and Fast Multimedia Library)** where the player navigates through progressively challenging levels, collects coins, avoids enemies, and manages health. The project demonstrates:

1)Object-oriented programming (OOP) in C++.

2)Game loop implementation.

3)Collision detection systems.

4)Asset management (textures, audio).

5)State handling (menus, levels).

### **1.2 Target Audience**

**Developers**: C++ programmers interested in game development with SFML.

**Players**: Casual gamers (if distributed).

**Educators**: For academic evaluation (e.g., as a coursework submission).

### **1.3 Tools & Technologies**

| **Component** | **Details** |
| --- | --- |
| **Language** | C++11 (or newer) |
| **Library** | SFML 2.5+ (Graphics, Audio, Window) |
| **IDE** | Visual Studio Code, GCC, or any C++-compatible environment |
| **Version Control** | Git ([GitHub Repository](https://github.com/DavidRaigithub/WORKSHEET-4" \t "https://chat.deepseek.com/a/chat/s/_blank)) |

## ****2. Features****

### **2.1 Core Mechanics**

| **Feature** | **Description** |
| --- | --- |
| **Player Movement** | WASD/Arrow Keys for 4-directional movement with sprite animation. |
| **Coin Collection** | Coins increase score; all coins required to unlock next level. |
| **Enemy AI** | Enemies follow the player in Level 3; collisions reduce health. |
| **Health System** | Player loses health on enemy contact; game over at 0 HP. |

### **2.2 Additional Features**

**Sound Effects**: Background music + event-triggered sounds (coins, collisions).

**Save/Load**: Persistent game state storage (level, score, health).

**Main Menu**: Start game, exit, or load saved progress.

## ****Game Logic Flow****

1)Start at main menu

2)Begin game (level 1)

3)Collect all coins to unlock next level

4)Progress through levels (1 → 2 → 3)

5)Complete level 3 to win game

6)If health reaches 0, game over

## ****4. Technical Details****

### #Collision Detection

1)Wall collisions based on grid system

2)Enemy collisions reduce health with cooldown

3)Decoration collisions block movement

### #Sound System

1)Background music for menu and game

* Sound effects for:

1)Coin collection

2)Enemy collisions

2)Game over

4)Game completion

### #Save/Load System

### **Saves:**

1)Player position

2)Health

3)Score

4)Current levelCoin positions

### #Rendering Order

1)Background.

2)Grid/walls.

3)Decorations.

4)Enemies.

5)Coins.

6)Player.

7)UI elements (score, health bar).

## ****5. Dependencies****

1)SFML 2.5+ (Graphics, Audio, Window modules)

2)C++11 or newer

## 6.Assest Requirements

The game expects the following assets in an assets folder:

Textures:

1)Player sprites.

2)Enemy sprites.

3)Coin sprite.

4)Background images.

5)Wall/stone texture.

6)Tree texture.

7)Shadow texture.

8)Fence texture.

9)Fonts.

10)Audio files.

## Known Limitations

1)Fixed window size (800x600).

2)Limited enemy AI (only follows in level 3).

3)Basic collision system.

4)No difficulty settings.

## Future Improvements

1)More levels.

2)Enemy variety.

3)Player abilities/attacks.

4)More sophisticated save system.

5)Configurable controls.

6)Particle effects.

## ****9. Game Screenshots( Level 1 - Level 3)****

