Project Title: THE THING

Group Members

- Omer Shahid: 24K-0855

- Asad Imran: 24K-0721

- Affan Rasheed: 24K-0579

Submission Date: April 24, 2025

1. Executive Summary

• Overview: This project implements a 2D side-scrolling game inspired by the classic "Dino Game." The game uses Object-Oriented Programming (OOP) principles and SFML (Simple and Fast Multimedia Library) for graphics, audio, and input handling. The main tasks included designing game mechanics, implementing player interactions, and creating a visually appealing interface.

• Key Findings:

- Successfully implemented core game mechanics such as jumping, obstacle spawning, and collision detection.

- Integrated audio and visual elements to enhance the user experience.

- Demonstrated the use of OOP concepts like encapsulation, inheritance, and polymorphism.

2. Introduction

• Background: The "Dino Game" is a popular browser-based game that serves as a fun and engaging way to demonstrate OOP principles. This project was chosen to explore how OOP concepts can be applied to game development, focusing on modularity, reusability, and maintainability.

• Project Objectives:

- Develop a 2D game using SFML.

- Implement game mechanics such as player movement, obstacle generation, and scoring.

- Demonstrate the use of OOP principles in a real-world application.

3. Project Description

• Scope:

- Included: Player character (Dino), obstacles, scoring system, background elements, and audio effects.

- Excluded: complex animations.

• Technical Overview:

- Tools Used: Visual Studio Code, SFML library, C++.

- Technologies: Object-Oriented Programming, SFML for graphics and audio.

4. Methodology

• Approach:

- Weekly planning sessions to define milestones.

- Iterative development with regular testing and debugging.

- Agile methodology to adapt to challenges and refine features.

• Roles and Responsibilities:

- Omer Shahid: Implemented the player character, audio and overall game logic.

- Affan Rasheed: Developed the obstacle generation, scoring system, fps system and filing.

- Asad Imran: Background elements, restart functionality, pause functionality and ground generation

However, much of the implementation was carried out collaboratively through discussions on WhatsApp and Discord.

5. Project Implementation

• Design and Structure:

- Classes: `Dino`, `Obstacle`, `Ground`, `Scores`, `SoundManager`, `GameState`.

- Game Loop: Handles input, updates game state, and renders graphics.

• Functionalities Developed:

- Player movement and jumping mechanics.

- Dynamic obstacle spawning and collision detection.

- Scoring system with increasing difficulty.

- Background music and sound effects.

• Challenges Faced:

- Collision Detection: Adjusting bounding boxes for accurate detection.

- Audio Integration: Synchronizing sound effects with game events.

- Performance Optimization: Managing multiple objects efficiently.

6. Results

• Project Outcomes:

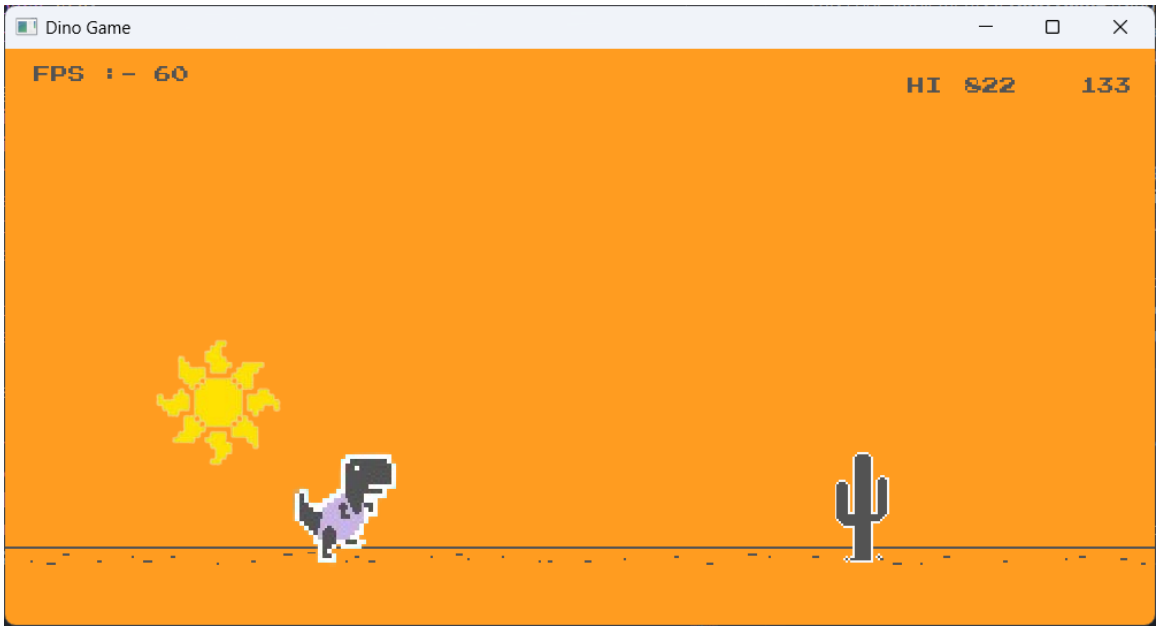
- Fully functional 2D game with smooth gameplay and responsive controls.

- Modular codebase demonstrating OOP principles.

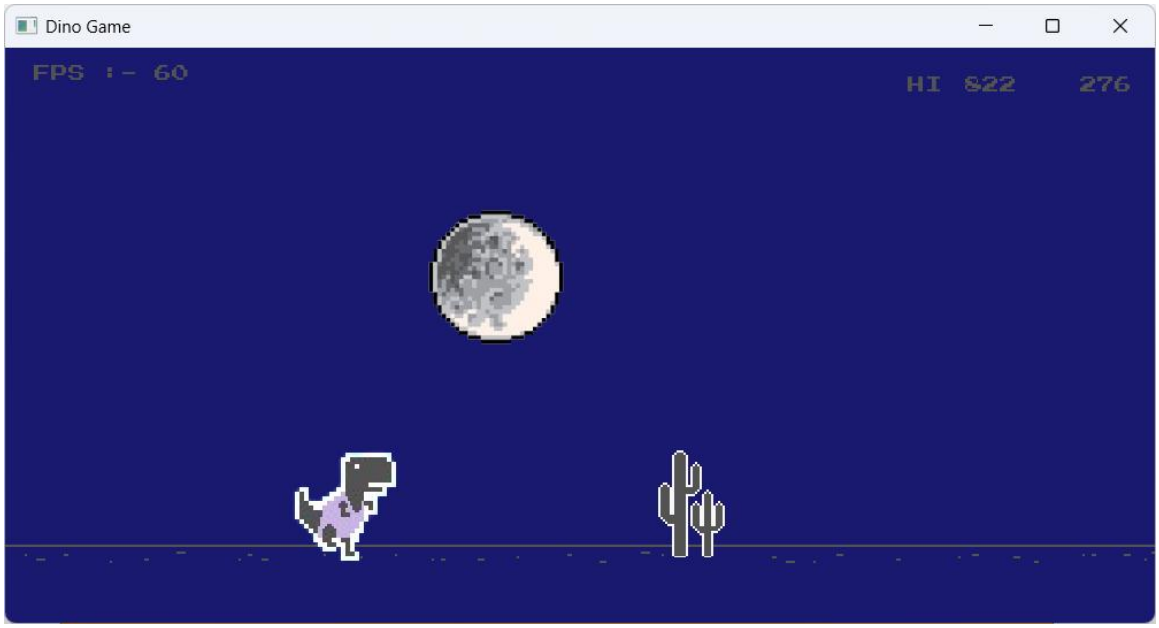
• Screenshots and Illustrations:

- Game Interface:

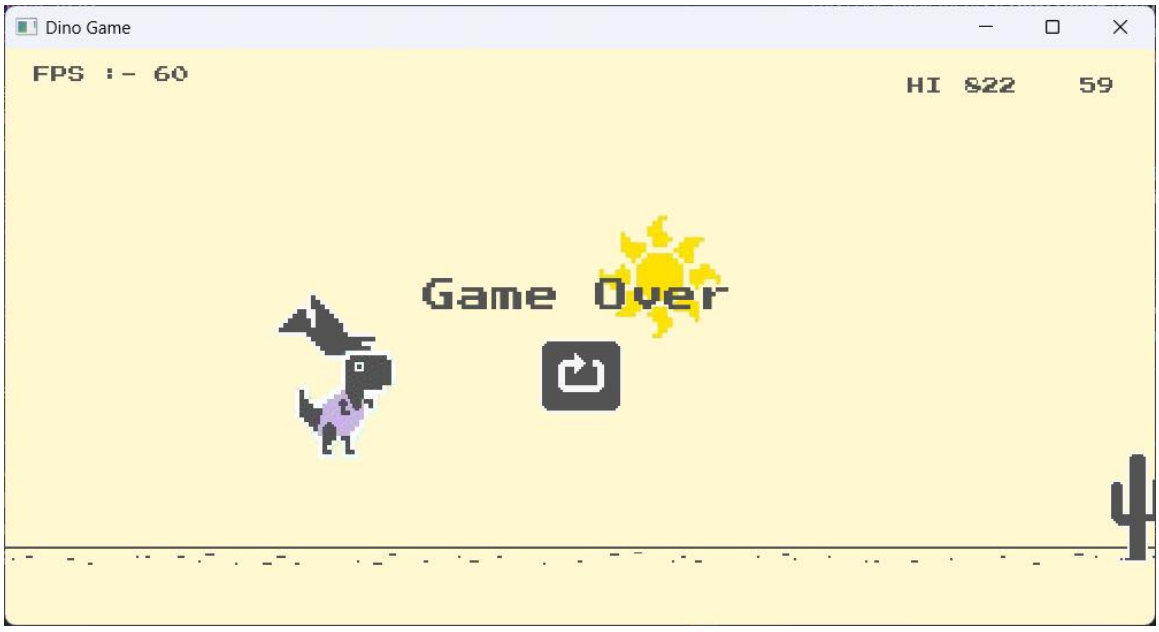
DAY TIME



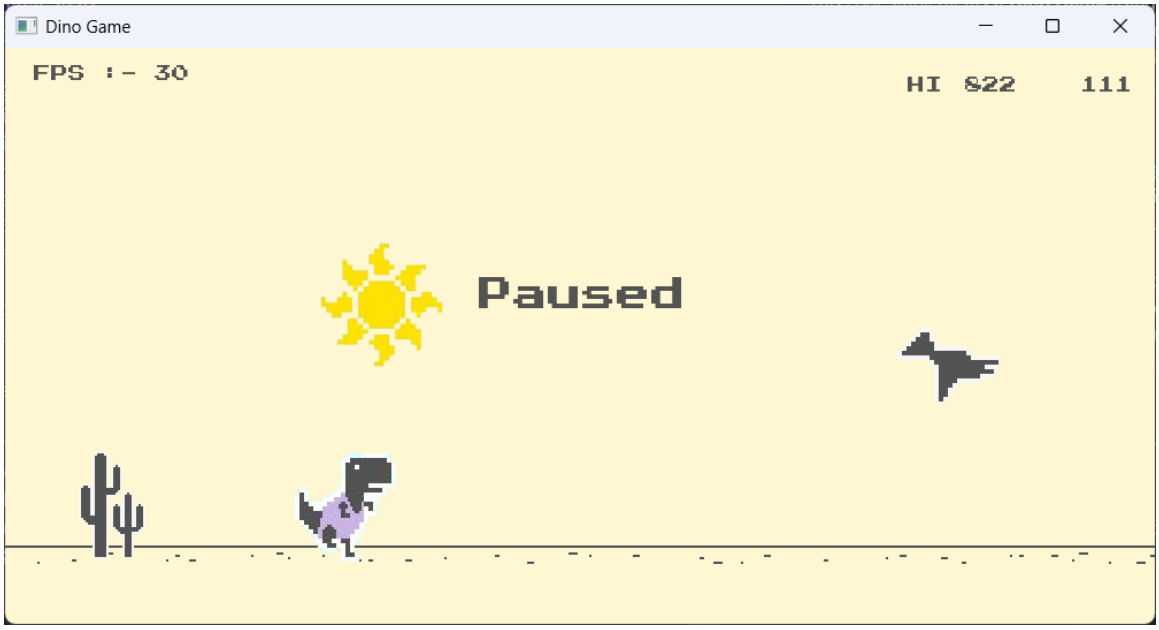
NIGHT TIME



- Game Over (Collision detection)



- Game paused (esc sequence)



• Testing and Validation:

- Tested on multiple systems for performance and compatibility.

- Validated gameplay mechanics through feedback from other peers.

7. Conclusion

• Summary of Findings: The project successfully demonstrated the application of OOP principles in game development. Key accomplishments include a modular design, engaging gameplay, and a polished user experience.

• Final Remarks: This project provided valuable insights into game development and the practical use of OOP. Future improvements could include additional features like power-ups, multiplayer mode, and enhanced animations.