OOP Project Proposal

Project Title

THE THING: A Simple dino 2D Game Using Object-Oriented Programming Concepts

Group Members

1. Omer Shahid: 24K-0855
2. Asad Imran: 24K-0721
3. Affan Rasheed: 24K-0579

1. Introduction

• Background: This project focuses on creating a simple 2D game using Object-Oriented Programming (OOP) principles. The game is inspired by the classic "Dino Game" and involves a player controlling a dinosaur to avoid obstacles.

• Problem Statement: The project addresses the problem of designing a modular, reusable, and maintainable game structure using OOP concepts such as encapsulation, inheritance, and polymorphism.

• Objectives:
  - Build a functional 2D game with basic gameplay mechanics.
  - Demonstrate the use of OOP principles in game development.
  - Showcase modular design for scalability and maintainability.

2. Scope of the Project

• Inclusions:
  - Core gameplay mechanics (e.g., jumping, obstacle spawning).
  - Score tracking and display.
  - Pause functionality.
  - Game over and restart features.
  - Background elements like Day-night cycle and ground movement.
  - Sound effects for actions like jumping and scoring.
  - Advanced animations for the dinosaur and obstacles.
  - FPS counter for performance monitoring.

• Exclusions:
  - Multiplayer functionality.
  - Complex physics
  - Mobile or web platform support.

3. Project Description

• Overview: The project involves creating a 2D game where the player controls a dinosaur to avoid obstacles. The game uses OOP concepts like encapsulation (e.g., classes for Ground, Obstacles, Scores), inheritance (e.g., shared behavior for game objects), and polymorphism (e.g., different obstacle types).

• Technical Requirements:
  - Tools: Microsoft Visual Studio, SFML library for graphics and audio.
  - Programming Language: C++.

• Project Phases:
  1. Research: Study SFML and OOP principles.
  2. Planning: Design the game structure and class hierarchy.
  3. Design: Implement core classes like Ground, Obstacles, Scores, and Dino.
  4. Implementation:
    - Core gameplay mechanics.
    - Pause functionality and FPS counter.
    - Advanced animations and sound effects.
  5. Testing: Debug and refine the game.

4. Methodology

• Approach:
  - Use an iterative development process, starting with basic functionality and gradually adding features.
  - Modularize the codebase for easy debugging and future enhancements.

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

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

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

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

5. Expected Outcomes

• Deliverables:
  - A working 2D game with core gameplay mechanics.
  - A short report explaining the use of OOP concepts.
  - User instructions for playing the game.

• Relevance: The project demonstrates the application of OOP principles in game development, showcasing modular design, encapsulation, and reusability. It also highlights the use of SFML for graphics and audio in C++ programming.

6. Resources Needed

• Software:
  - Microsoft Visual
  - Use of msys2, a software distribution which making it easier to install, use, build, and port software on Windows.
  - SFML library for graphics and audio.
• Other Resources:
  - Online tutorials for SFML and C++.
  - Instructor guidance for debugging and optimization.
  - [SFML tutorial playlist](#)