Component Code Description

Overview

This report outlines the development and testing processes for a 2D shooter game implemented in C++. The game includes fundamental elements like a player character, enemies, and bullet mechanics, utilizing OpenGL for rendering and incorporating basic collision detection.

Code Description

The game's architecture is structured around several key classes: `Player`, `Bullet`, `Enemy`, and `Game`. Each class is responsible for specific aspects of the game:

Player Class:
- Manages player attributes like position, health, and dimensions.
- Handles shooting mechanics, allowing the player to fire bullets.
- Includes collision detection to check if the player is hit by enemy bullets.

Bullet Class:
- Represents bullets with properties like position, speed, and damage.
- Includes a method to move bullets and a drawing function for rendering.

Enemy Class:
- Similar to the Player class, but represents enemy characters.
- Contains attributes for position, health, and dimensions.
- Includes a method for checking collision with player bullets.

Game Class:
- Central class that manages the game state, including score and time.
- Handles initialization of enemies and updates game elements.
- Responsible for rendering all game objects and managing collisions.

The main function (`main()`) creates a game loop that continuously updates the game state, processes user input (mouse movements), and renders the game frame