

USNO KARMAKAR: Player & Core Controls (Main Character System)

Responsibility: Everything related to the player character

Features to Implement

1. Movement Controls

- A → Move left
- D → Move right
- Smooth movement using velocity instead of instant jumps.

2. Jumping Mechanism

- W → Jump
- Implement gravity (player falls back down).
- Prevent double jumping (jump only when on ground or block).

3. Ducking Mechanism

- S → Duck
- Reduce player height when ducking.
- Used to avoid flying enemies.

4. Player States

- Normal
 - Jumping
 - Ducking
 - Dead (game over)
-

SADIA FARDEEN MAHMOOD: Environment, Background & Platforms

Responsibility: World movement and level illusion

Features to Implement

1. Moving Background

- Background scrolls left when player moves forward.
- Creates illusion that the character is moving.
- Can be parallax (multiple layers moving at different speeds – optional).

2. Floating Platforms / Blocks

- Randomly generated blocks at different heights.
- Player can jump and land on them.
- Blocks move with the background.

3. Coins System

- Coins placed above floating blocks.
- Detect collision between player and coin.
- Increase coin counter when collected.

4. Random Generation Logic

- Platforms and coins spawn randomly ahead of the player.
 - Old objects are removed once they move off-screen.
-

MD. FARHAT ANJAM: Enemies, Obstacles & Collision Logic

Responsibility: Game difficulty and danger mechanics

Features to Implement

1. Enemy Type 1: Shooting Enemy

- Stays at fixed position.
- Shoots bullets at intervals.
- Bullet collision with player → death.

2. Enemy Type 2: Zombie

- Walks on ground toward the player.
- Touching player → death.

3. Enemy Type 3: Flying Enemy

- Moves in air at player height.
- Touching player → death.
- Player can **duck (S)** to avoid collision.

4. Collision Detection System

- Player vs enemies
 - Player vs bullets
 - Player vs spikes
 - Simple rectangle-based (AABB) collision.
-

FARZANA HASAN SHOMI: Ground, Spikes & Player

Responsibility: Game difficulty and danger mechanics

Features to Implement

1. Ground System

- Continuous ground that moves with the background.
- Player must stay above ground.

2. Obstacle: Spikes

- Placed on ground.
- Collision with player → player dies.

3. Player Model (2D Sprite / Shape)

- Draw the main character using OpenGL (rectangle / textured sprite).
 - Maintain player position (x, y), velocity, and state.
-

Shared Responsibilities (All Members)

These should be discussed and implemented together:

- Game loop (update → draw → input)
 - Keyboard input handling
 - Score system (coins collected)
 - Game over screen & restart option
 - Code integration & testing
-