Project Report: *2 Player Angry Birds*

Shivansh Nair

April 29, 2025

Abstract

This project aims to create an Angry Birds style game designed for two players in a turn-based PvP format. Players take turns slingshotting birds onto their opponent's structures to destroy them and win the game.

Contents

1 Modules

External Libraries Used

- **pygame**: Used for game development including rendering graphics, handling user input, and managing game loops and events.
- **random**: Utilized for randomizing block types and game elements to add variability.
- **math**: Used for mathematical calculations such as physics and trajectories.
- **math**: Used for mathematical calculations such as physics and trajectories.

2 Directory Structure

```
main.py             # Entry point of the game; initializes and launches the main men
game.py             # Game logic and loop
menu.py             # Menu system and UI
utils/
   bird.py          # Bird class: behavior, physics, and special abilities
   block.py         # Block class: structure, damage, and destruction logic
assets.py           # contains bird and background images
README.md           # Project documentation and instructions
```

File Functions

- `main.py`: Launches the game, handles initial setup, and routes to menu or game screens.
- `game/`: Contains the main game loop and core gameplay logic.
- `menu/`: Implements the menu system, including navigation and user interface.
- `utils/bird.py`: Defines the `Bird` class, including movement, physics, and special abilities.
- `utils/block.py`: Defines the `Block` class, including hit detection, damage, and destruction.
- `assets/`: Contains all bird sprites and background images

3 Running Instructions

1. **Install Pygame ce:**

   ```
   pip install pygame-ce
   ```

2. **Run the Game:**

```
python3 main.py
```

3. **Controls:**

   - **Mouse Click & Drag**: Click and drag the bird from the slingshot to set angle and power.
   - **Mouse Release**: Launch the bird towards the opponent's structure.
   - **Keyboard (e.g., Spacebar)**: Use the special ability of the currently launched bird (if available).

4 Features

4.1 Basic Features

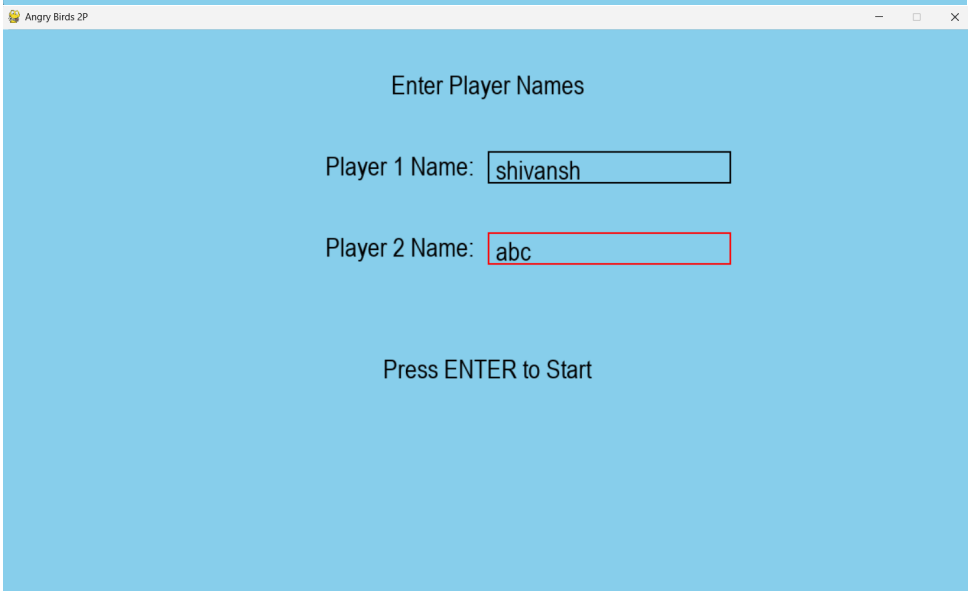
- **Turn-Based PvP Gameplay**: Two players alternate turns launching birds to destroy each other's structures. Both players have the same structure to make it fair.
- **Multiple Bird Types**: Four bird types (red, yellow, blue, black) each with unique properties and special abilities. Each bird type does differing amounts of damage to different blocks.
- **Physics-Based Projectile Motion**: Bird trajectories are calculated using velocity, gravity, and collision physics.
- **Destructible Blocks**: Structures made of different block types (wood, ice, stone) that respond to bird impacts and can be destroyed.

4.2 Advanced Features

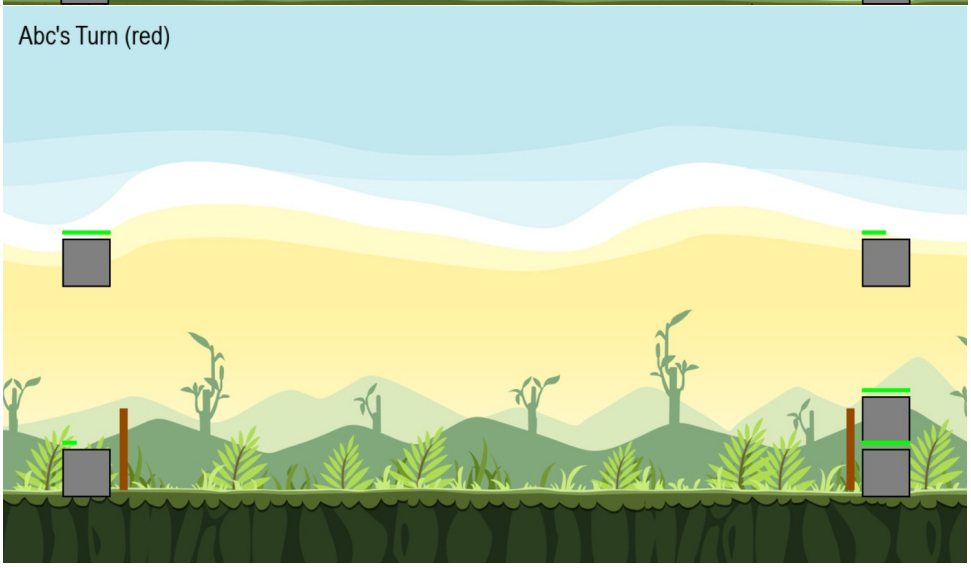
- **Special Bird Abilities**: Birds can activate special powers mid-flight (triggered by keyboard) to affect blocks differently or spawn additional birds. For example, blue can duplicate and bomb can explode.
- **Dynamic Launch Arc Visualization**: While dragging the bird, a predicted trajectory arc is drawn to assist aiming.
- **Collision Response and Bounce**: Birds bounce off blocks with velocity adjustments depending on the collision side and block state.
- **Win Detection and Turn Management**: The game automatically detects when a player's blocks are destroyed and declares the winner, managing turns accordingly.

4.3 Gameplay Images

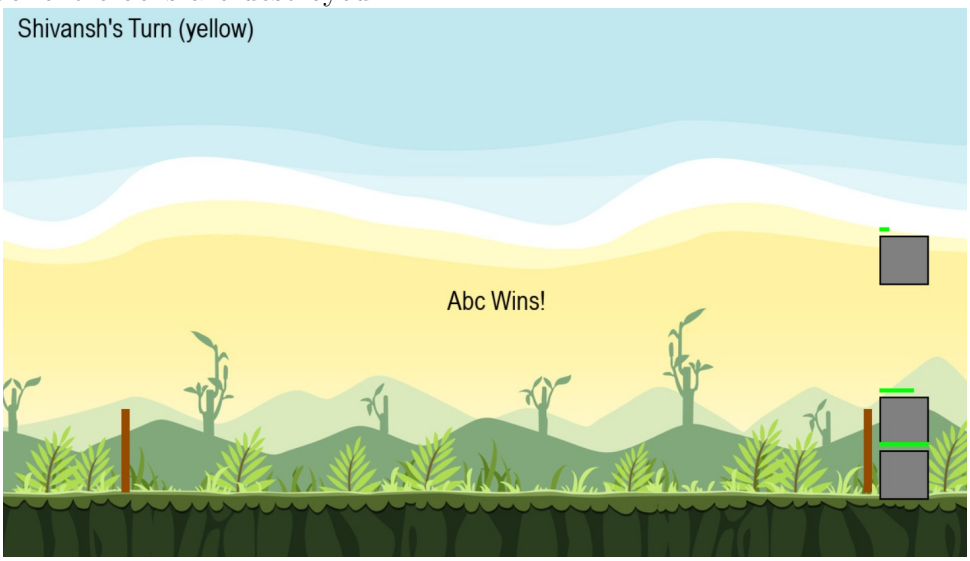
First when the game is launched, the players need to input their names and press enter to begin.



Then, the game menu is opened. players must turn by turn drag the bird from the slingshot and target enemy bases. Health bars of each block is displayed above it.



As the game goes on, blocks keep getting destroyed and someone wins when all their opponent blocks are destroyed.



5 Project Journey

Key Learnings

- Implementing realistic physics for projectile motion and collision detection enhanced understanding of kinematics.
- Managing turn-based multiplayer logic within a single game loop improved skills in state management.
- Using Pygame for graphics and input handling provided practical experience with game development frameworks.

Challenges and Solutions

- **Challenge:** Handling accurate collision detection between moving birds and blocks.
  Solution: Implemented bounding box checks and stored last hit blocks to prevent repeated hits in the same frame.
- **Challenge:** Balancing bird launch velocity and gravity to create a natural and fun trajectory.
  Solution: Tuned physics constants such as gravity and velocity limits through iterative testing.
- **Challenge:** Visualizing the predicted launch arc dynamically during mouse drag.
  Solution: Calculated future positions using physics equations and rendered the arc with Pygame lines.

6 Bibliography

1. Pygame Documentation: `https://www.pygame.org/docs/`
2. Chat GPT: `https://chatgpt.com/`