Game Proposal: Bullet Brawl

CPSC 427 - Video Game Programming

Team Members:

Seraj Abo Sabbah (62850656)
Danny Hu (94336237)
Ishraq Mostafa (58930181)
Jason Wei (26637397)
Leo Yu (23803794)
Akshat Singla (????????)

Story:

This game is inspired by fighting games such as Super Smash Bros. and draws heavily upon its gameplay mechanics and objectives, however, rather than melee combat Bullet Brawl will focus on ranged combat via guns.

Bullet Brawl is an action-packed multiplayer game designed for 2 (but can support up to 4 players). Each player starts with three lives and a basic weapon, but the real excitement begins as they fight to knock each other off the platforms to reduce their opponent's lives to zero. When players get hit with bullets they are knocked back and the goal is to knock your opponents off the platforms using well-placed shots while being careful not to fall off yourself and remain the last one standing to win. Weapon mystery boxes also spawn at random times around the map and these boxes yield new and exciting weapons. Power-ups also appear, adding an extra layer of strategy to the chaos.

Technical Elements:

Rendering: We will have platform meshes that will be destroyed by the player's throwables. The sprites will be rendered at the start of the game at specific spawn locations, and weapon and power-up crates will be randomly spawned as the game progresses.

Geometric/Sprite/Other assets: We will have sprites for the textures for players, platforms, weapons, throwables, power-ups etc. There will also be assets for animations for player jumping, moving horizontally, moving up and down platforms, shooting, throwing objects, dying etc.

2D Geometry manipulation: Players sprite will be able to move and change directions. Throwables will have a projectile path

Gameplay logic: Need logic to move the player around and allow it to jump. Also will need logic doing something similar to ray casting in order to determine whether interactable objects are in the player's line of sight.

Physics: Acceleration on sprint movement and deceleration. There is also acceleration on the bullet as it travels through the platform. Limited jumping vertically and gravity effect. The player can jump to a different platform and stay there, or drop from the current platform with a key.

Advanced: AI will be used for adding extra enemies to the game to increase the pace and action in the arena. **Suggestion:** Ensure AI-controlled enemies use strategic tactics and decision-making for engaging gameplay.

Audio: Gun shots and kills will have a specific audio. Also we will have a pop sound when the player picks up power-ups or weapons

Advanced Technical Elements:

AI enemy players

- The player can choose to compete with AI to add more players to the battle
- The AI will attempt to shoot in the direction of the player and try to knock the player off the platform
- The AI will attempt to collect power-ups and mystery boxes so the player cannot get them
- **Impact:** No way to play the game without two human players
- **Alternative:** Do not implement AI or implement a very simple predetermined path that the NPC's follow as AI is not crucial to the game since it is designed with player vs player in mind.

Destruction

- Platforms can be destroyed partially by explosives/explosive weapons (rpg, grenades etc.). Can be fully destroyed if the platform keeps getting shot at.
- **Impact:** Less dynamic gameplay as maps are all static, players can play on the same platform for extended periods of time
- **Alternative:** The entire platform breaks when hit with an explosive

One-way collisions for platforms

- Ability to jump to platforms above and drop to platforms below without colliding
- **Impact:** Decreases the verticality of the game, players may be stuck without an escape path
- **Alternative:** solid platforms that players cannot jump through, with map design that balances this

Dynamic Camera

- A dynamic camera like the one in Super Smash which follows players around and keeps both players in view so the action is more clearly seen
- **Impact:** For larger maps it may be more difficult to see the action when the players are up close
- **Alternative:** have a static camera that shows all of the map

Moving Platforms

- Platforms that move around the map and carry players around
- **Impact:** Gameplay less dynamic as this would add another way to move around the map
- **Alternative:** Do not implement the feature for consistency across all maps

Devices:

Can use either keyboard or mouse for menu selection

Player 1:

Up	w
Down	s
Left	a
Right	d
Shoot	g
Block	h

Player 2:

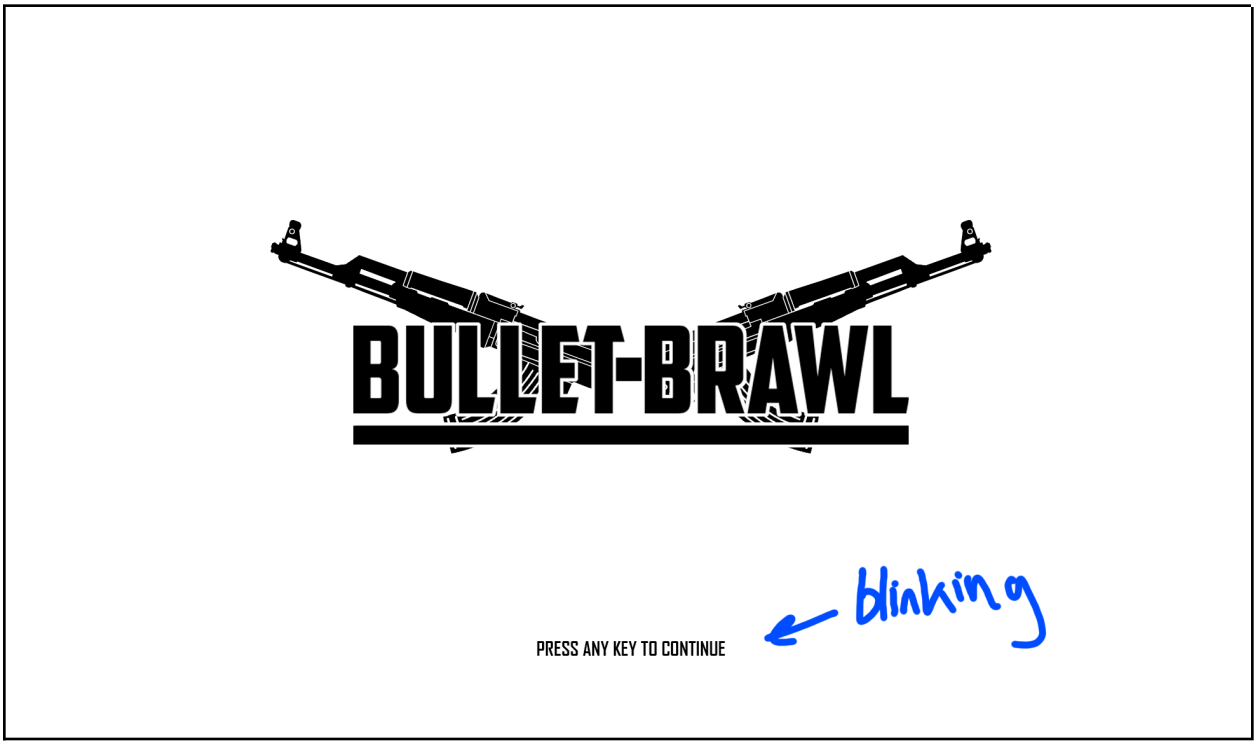
Up	Up arrow
Down	Down arrow
Left	Left arrow
Right	Right arrow
Shoot	Numpad 1
Block	Numpad 2

Others:

Return to main menu	Esc

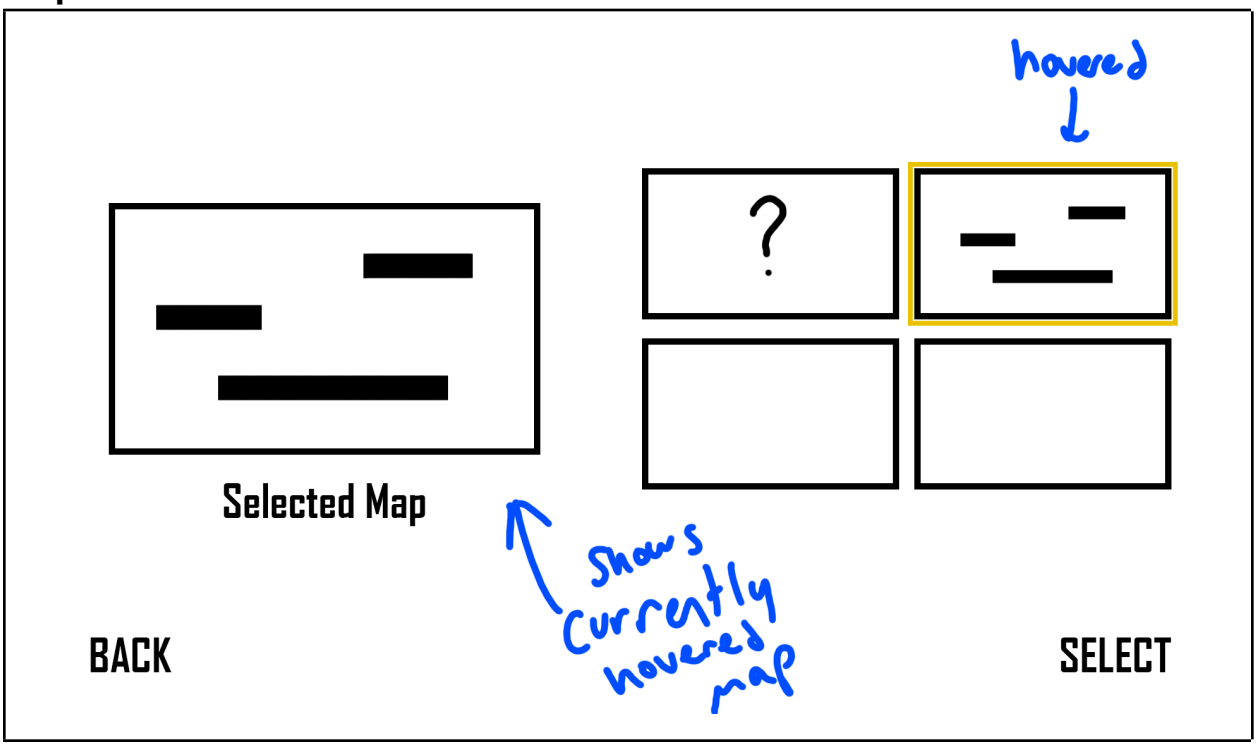
Concepts:

Start screen

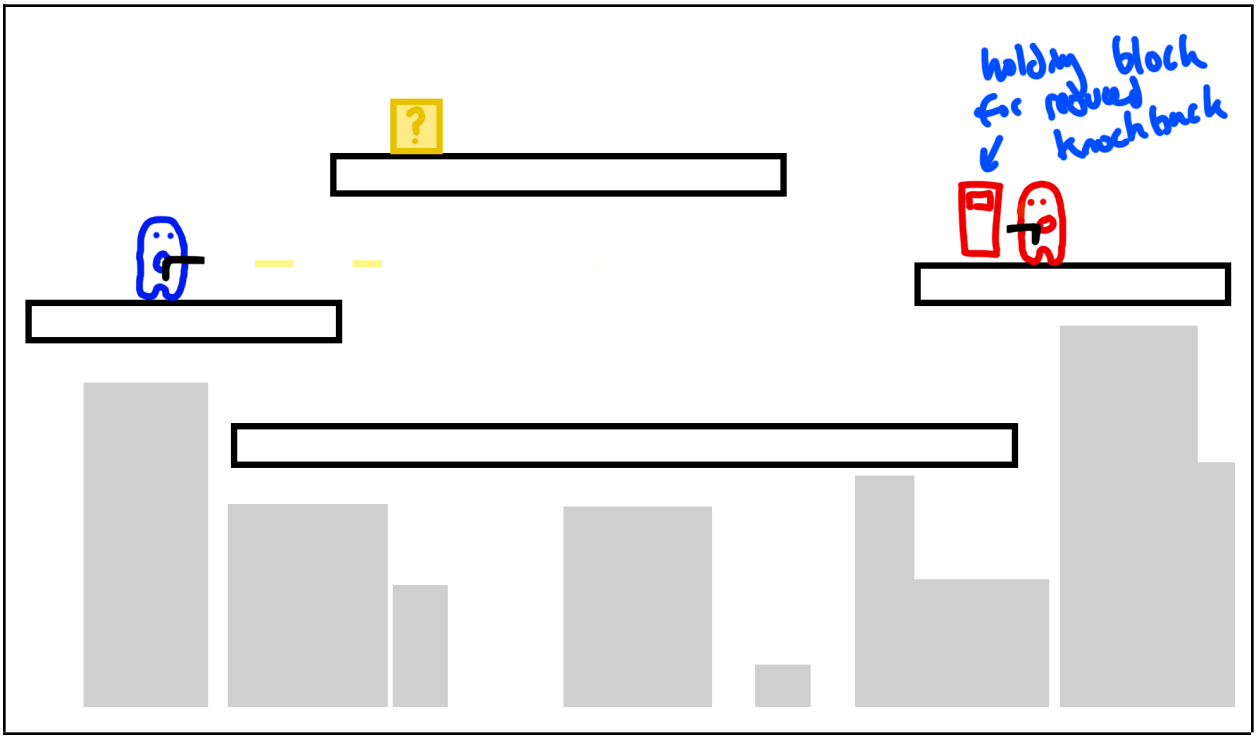


Main menu

Map selection



Gameplay



Tools:

We will use Adobe Photoshop and Illustrator to create our art assets.

Team management:

Use GitHub issues to track tasks, and assign group members to the tasks. These can also be attached to PRs

Development Plan:

Skeletal Game Due Oct 12th:

Week 1 September 25

- Basic collision detection (player and platform)
- Basic player sprite movement
- Movement controls for one player
- Implement test map
    - Single long platform

Week 2 October 2

- Projectile sprite movement
- Horizontal shooting of a projectile originating from player
- Defined game boundaries
    - Kill player if boundary is touched
    - Respawn player at spawn point
- Designs for more maps

Week 3 October 9

- Implement textures for platform
- Basic physics
    - Gravity
    - horizontal acceleration and deceleration
    - Vertical acceleration (jumping)
- Basic assets
    - Player model

Minimal Playability Due Oct 30th:

Week 1 October 16

- Projectile collisions with players
    - Projectiles knockback players
    - Knockback is reduced with distance the bullet travels
- Implement the ability to block projectiles
- The second player made available
- Sprite sheet animation for walking
- Player life counters
    - Track the number of deaths for each player
- Background art

Week 2 October 23

- Parallax backgrounds
- Add tutorial level and instructions
- Reset the game when a player's life count reaches 0
- Add more complex maps, and multi-platforms to support longer gameplay
- Advanced movement
    - Double jump

Playability Due Nov 20th:

Week 1 October 30

- One-way collision platforms
    - Can jump to platforms above player
    - Can drop off a platform the player is currently standing on
- Designs plus art and assets for 2 more maps
    - New maps must include platforms that are destroyable and moving platforms
- Designs for different weapon classes
    - Assault rifle
    - Submachine gun
    - Sniper
    - Grenade launcher
- Random weapon crate spawns
- AI players simple path finding

Week 2 November 6

- Gun models for the new weapons
- Power-up designs and implementation
    - Speed boost
    - Triple jump
    - Reduced knockback
    - Extra life
- Sound effects
    - Walking sound
    - Shooting
    - Bullet hit player
    - Death
    - Jumping
- Start screen
- AI advanced pathfinding and shoots toward the player

Week 3 November 13

- Destructible platforms for one map
- Different AI difficult levels
- Wrap up all art and assets
- Level selection
- Complete player animations
- Menu and options

Final Game Due Dec 4th:

Week 1 November 20

- Balance weapons and movement
- Add background music

- Victory screen

Week 2 November 27

- Polish game
- User testing, play testing