## 1. Brief introduction __/3

As TL4, I will be working on visual aspects of the project. I will design backgrounds for 3 separate levels with their unique tiles and interactable environment elements. This will set the tone for each level and create a visually appealing design for the users.

My main responsibilities include sourcing or creating images for the project's atmosphere and setting of each level and importing tiles to design a structured terrain such as creating obstacles, traps, interactable elements (buttons, levers) to add depth to the gameplay. The aim is to encourage users to explore and experiment with the environment. Additionally, implementing realistic physics interactions—such as gravity, collisions, and object movements—to enhance immersion and gameplay mechanics to ensure engaging experience.

I will also be implementing a checkpoint feature to ensure a balanced experience. We would not want the player to progress through the stage just to fail very close to reaching the next stage and repeat it from the very beginning. Another goal is to create minimal cutscenes to enrich the storyline.

2. Use case diagram with scenario __14

Use Case 1:
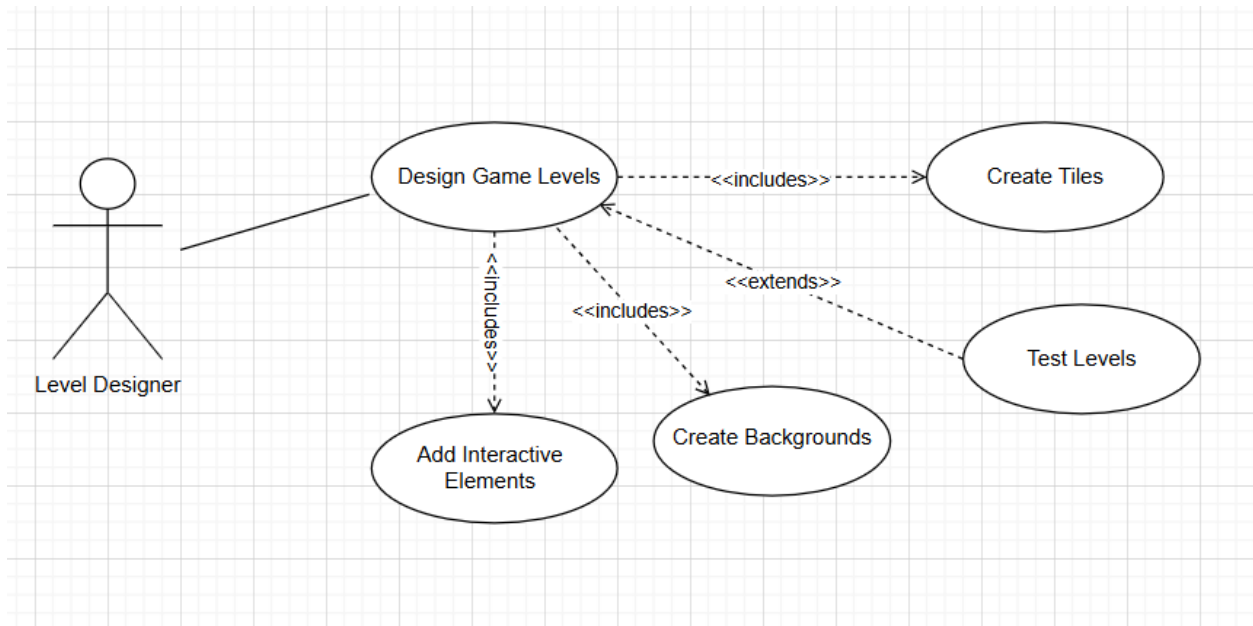
Scenario 1:

Name: Use case 1

Summary: The level designer creates and sets up a new level in the game including themes, tiles, and interactive elements.

Actors: Level Designer.

Preconditions: A design plan needs to be developed.

Basic sequence:

Step 1: Create a new scene.

Step 2: Add background theme.

Step 3: Create an empty canvas.

Step 4: Place tiles to form the terrain.

Step 5: Add interactive elements like traps, buttons, etc.

Step 6: Save progress and test the level.

Step 7: If the test is successful, the level is finalized and saved.

Exceptions:

Step 1: Unity fails to initialize objects: error message displayed on the log.

Step 5: An interactive element is missing required attributes: system prompts a correction.

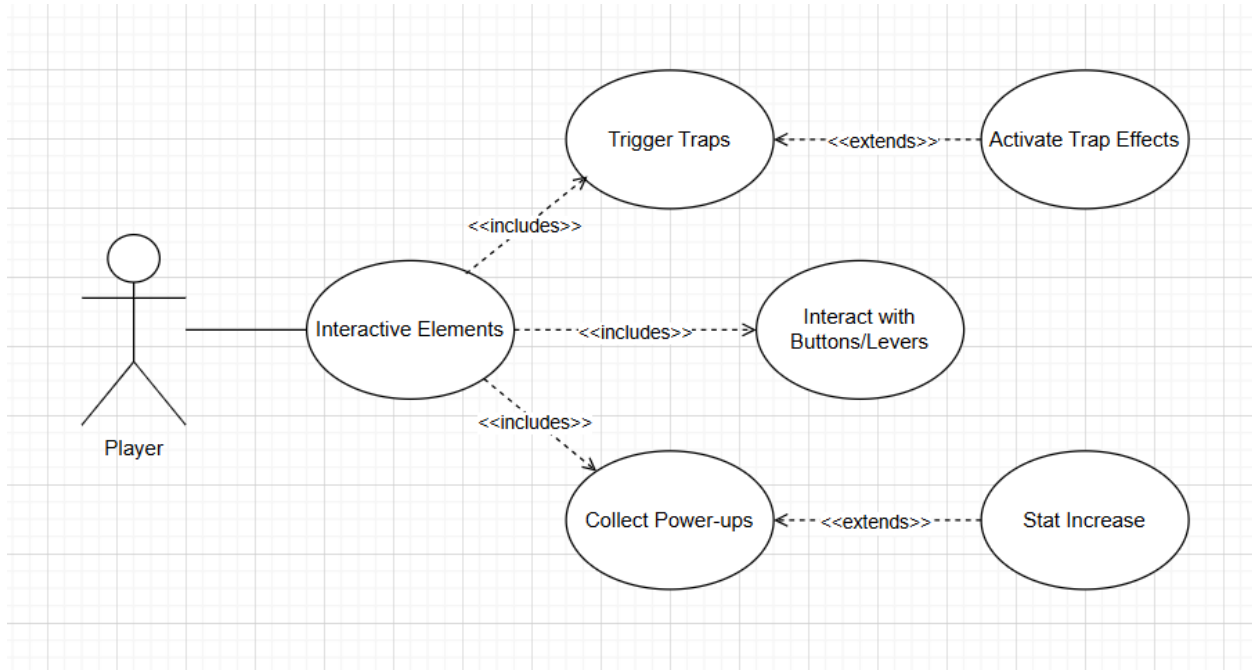**Step 6:** Level test fails due to an error (e.g. missing objective), designer receives feedback on the log.

Post conditions: The designed level is saved and ready for further testing and editing.

Priority: 1*

ID: LD01,

*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

Use Case 2:

Scenario 2:

Name: Use case 2

Summary: The player interacts with various environmental elements, including traps, buttons, levers, and power-ups, to navigate the level.

Actors: Player.

Preconditions: The player is within the range of an interactive object and the current level properly linked attributes to physics components.

Basic sequence:

Step 1: Move near an interactive element.

Step 2: Press the interaction button.

Step 3: The system identifies the type of interactive element.

Step 4: If it is a trap, the trap is triggered.

Step 5: If it is a button or a lever, the corresponding action is executed (e.g. opening a hatch or a door).

Step 6: If it is a power-up, the player collects it and gains benefits.

Exceptions:

Step 1: Player is out of range; the interaction is ignored.

Step 2: Interaction button is pressed, but there is no interactive object: nothing happens.

Step 4: Trap is already triggered; it cannot be triggered again immediately.

Post conditions: The interactive element is activated, changing the game state accordingly. If the player interacts with a power-up, their stats are updated.
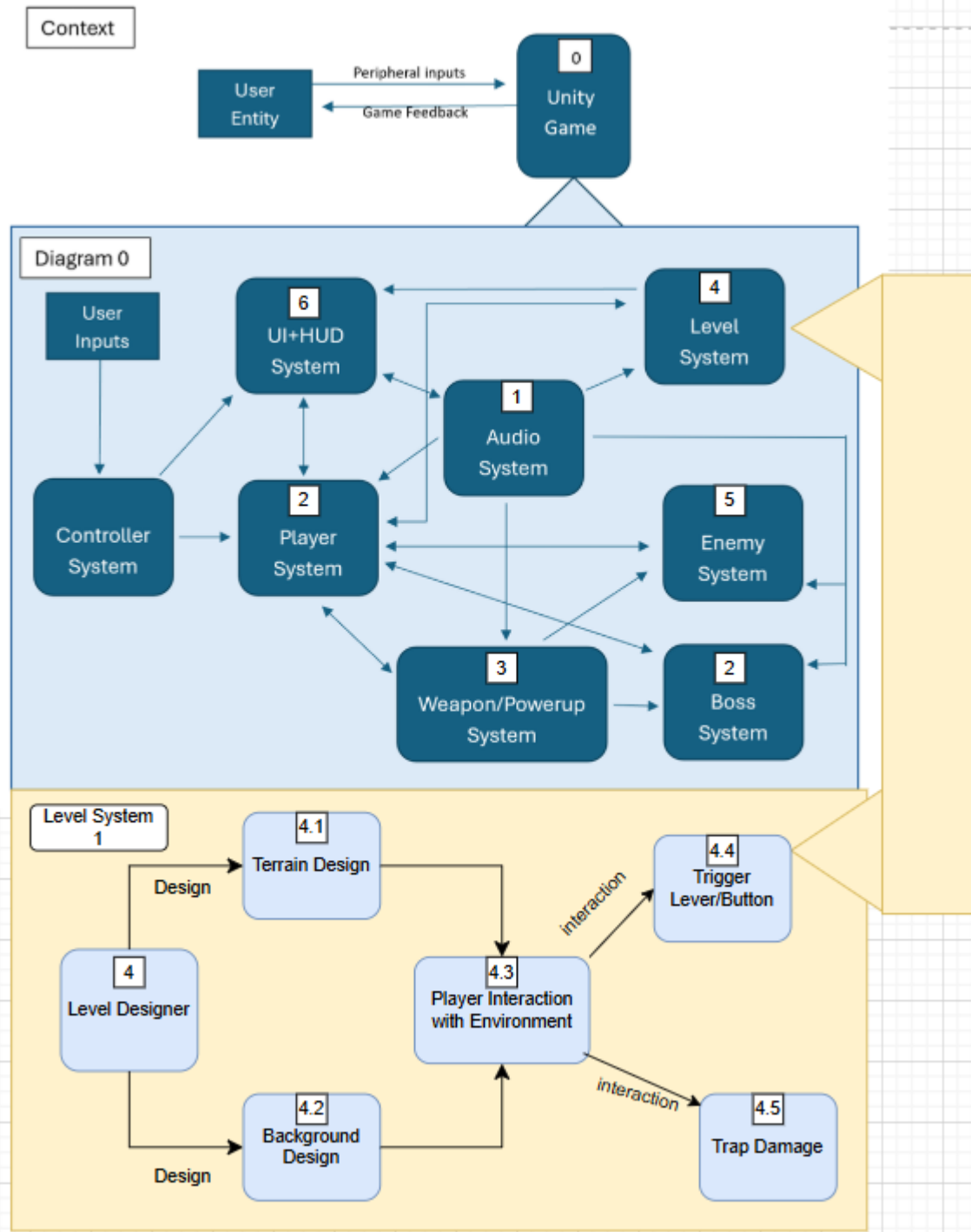
Priority: 2*

ID: IE01

*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

TechDown: Data Flow Diagram Abstraction Levels

Context

User Entity — Peripheral inputs → 0 Unity Game

Unity Game — Game Feedback → User Entity

Diagram 0

- User Inputs
- 6 UI+HUD System
- 4 Level System
- 1 Audio System
- Controller System
- 2 Player System
- 5 Enemy System
- 3 Weapon/Powerup System
- 2 Boss System

Level System 1

- 4 Level Designer — Design → 4.1 Terrain Design
- Design → 4.2 Background Design
- 4.3 Player Interaction with Environment
- interaction → 4.4 Trigger Lever/Button
- interaction → 4.5 Trap Damage

Process Descriptions

Level Begins:

IF player interacts with object:

        IF object type is lever && !lever.isFlipped

                Flip

                SetActive()

        ELSE IF object type is button && !isPressed

                Press

                SetActive()

        ELSE IF object type is trap

                IF trap is triggered by a player

                        activateTrap

4. Acceptance Tests _____9

Run each object 50 times to check functionality.

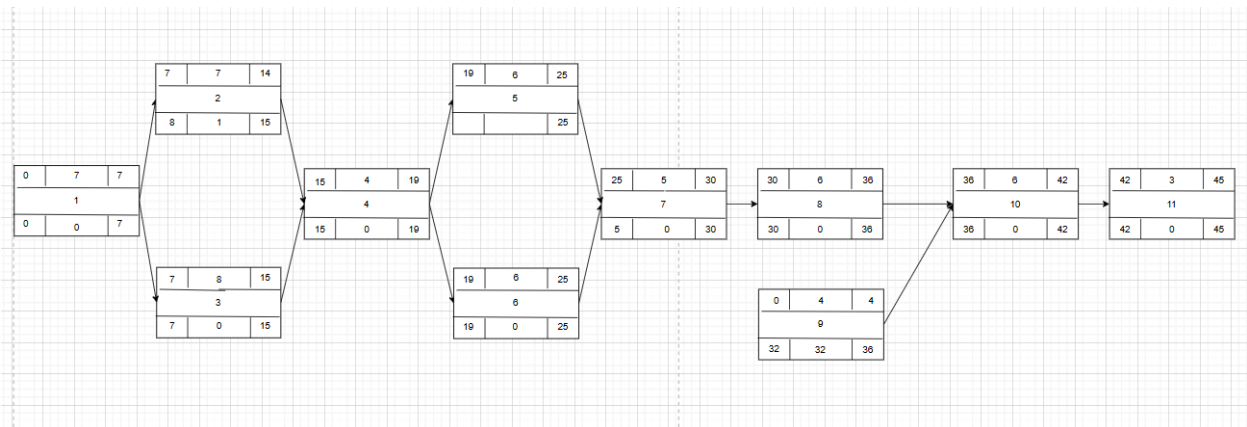The expected behaviors of the features are:

- Triggers interactable environmental elements correctly:
  - Does not activate randomly.
  - Does not fail to activate when interacted with.
  - Does not activate multiple times in a single interaction.
- Loading scenes correctly:
  - Does not fail to load or cause game crashes.
  - Does not load the wrong scene.
- Background loads correctly:
  - Does not fail to appear or load incorrect images.
  - Background layers are in correct order.
- Checkpoints activate properly:
  - Activates when the player reaches the correct position.
  - The game state is restored correctly.
- All these features are executed at a minimum of 49 times and at most 51 times.

5. Timeline _____/10

Work items

Work Items		
Task	Duration (hrs)	Predecessor Task(s)
1. Begin the level design process	7	N/A
2. Create and set up backgrounds for the level	7	1
3. Arrange tiles to form the map	8	1
4. Add physics components	4	2, 3
5. Add interactive elements	6	4
6. Add objectives	6	4
7. Create cutscenes	5	5, 6
8. Implement checkpoints	6	7
9. Documentation	4	N/A
10. Debugging and testing	6	8, 9
11. Installation and finalize	4	10

Pert diagram

Gantt timeline