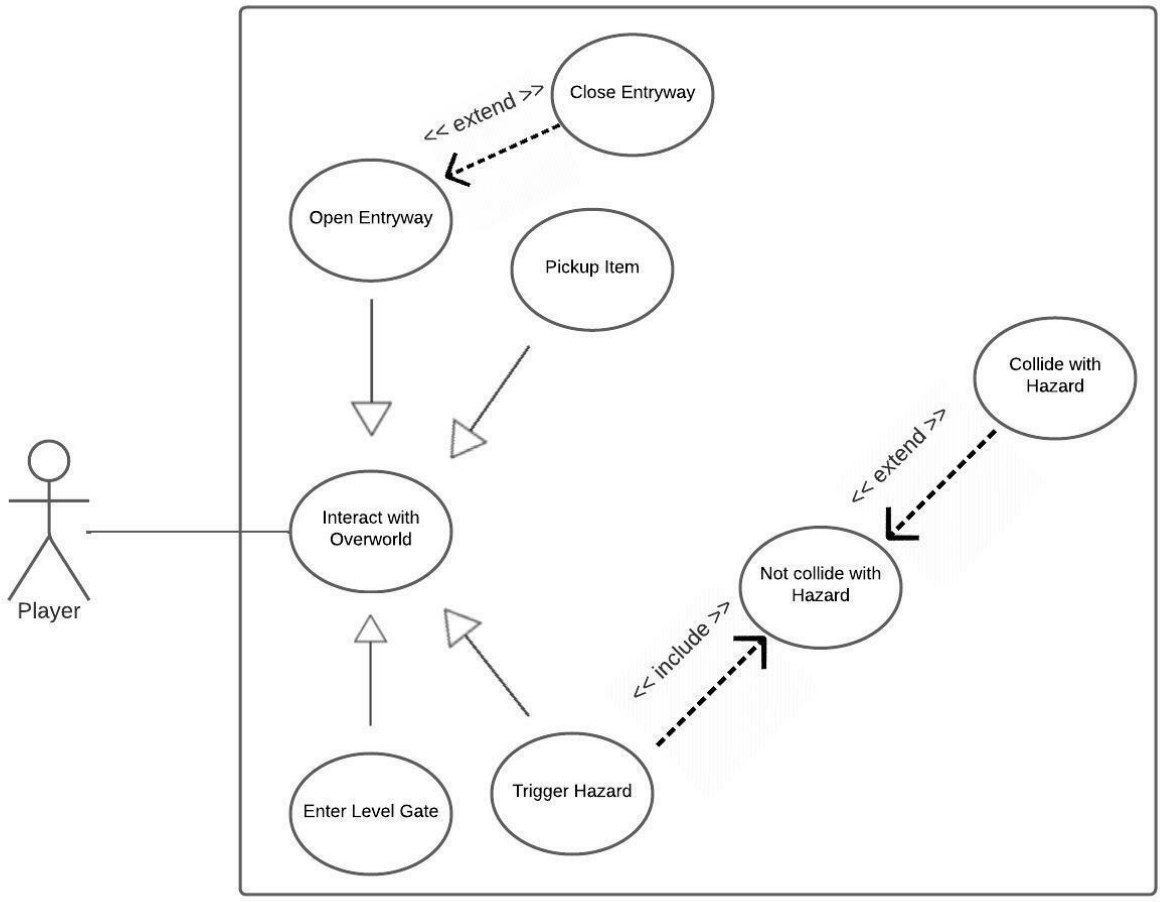
Name_____Seth Cram_____ Mark _____/50

## 1. Brief introduction _/3

My feature for “Dungeon Jump” is the Overworld Environment Builder. In our game, we plan on having two playable perspectives: 2D top-down and 2D side-scroller. These two perspectives will be linked and switchable through player-interactable entryways called “Level Gates”. I’m responsible for the level design of the 2D top-down perspective referred to as the Overworld.

## 2. Use case diagram with scenario _14

### Use Case Diagrams




### Scenarios

**Player Scenario**

**Name:** Player interaction with the Overworld.

**Summary:** The player interacts with the overworld through entryways, items, hazards, and level gates. They respond in an appropriate manner.

**Actors:** The player of the game.

**Preconditions:** Overworld has already been generated.

**Basic Sequence:**

**Step 1:** Player spawns into Overworld.
**Step 2:** Player walks around.
**Step 3:** Player interacts with the overworld in the form of picking up an item and adding it to their bag. .
**Step 4:** Player walks around.
**Step 5:** Player interacts with the overworld in the form of opening an entryway.
**Step 6:** Player walks around.
**Step 7:** Player interacts with the overworld in the form of entering a level gate.
**Step 8:** Player leaves overworld. (despawns)
**Step 9:** Player re-enters overworld. (respawns)
**Step 10:** Player walks around.
**Step 11:** Player interacts with the overworld in the form of triggering a hazard.
**Step 12:** Player doesn't collide with hazard.

(most steps don't necessarily have to be in this order besides 7-9. Although, we do need to walk around in-between essential steps and the player must spawn as step 1.)

**Exceptions:**

**Step 5:** The entryway is already open, so the player closes the entryway.
**Step 12:** If the player collides with a hazard, the player loses health.

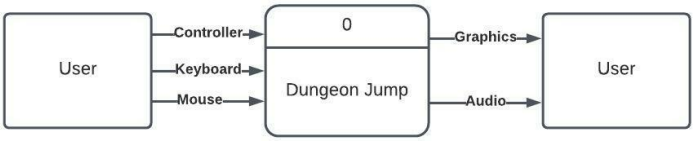
**Post Conditions:** The player interacts with the overworld environment in a sensical manner.
**Priority:** Overall 2* (but level gate interaction is 1*)
**ID:** PO1
*The priorities are 1 = must have, 2 = essential, 3 = nice to have

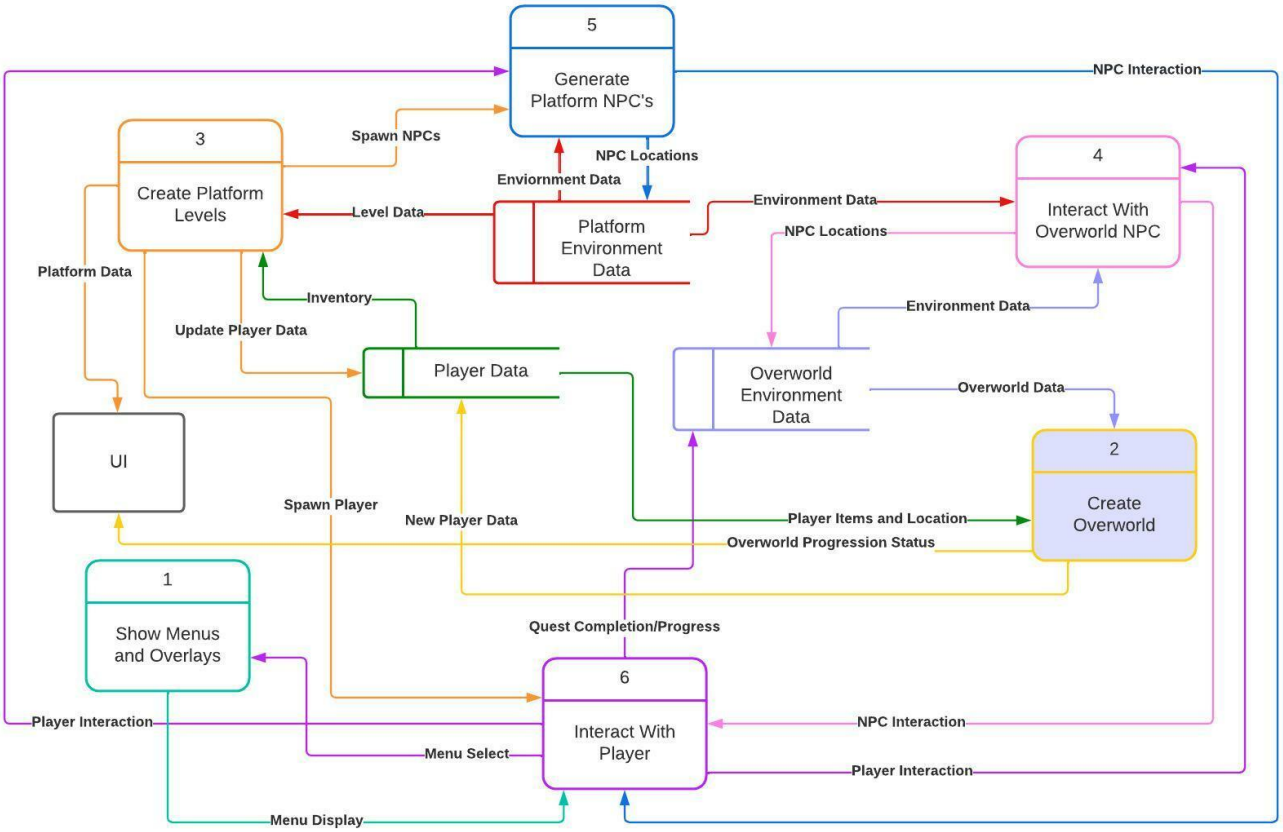
## 3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

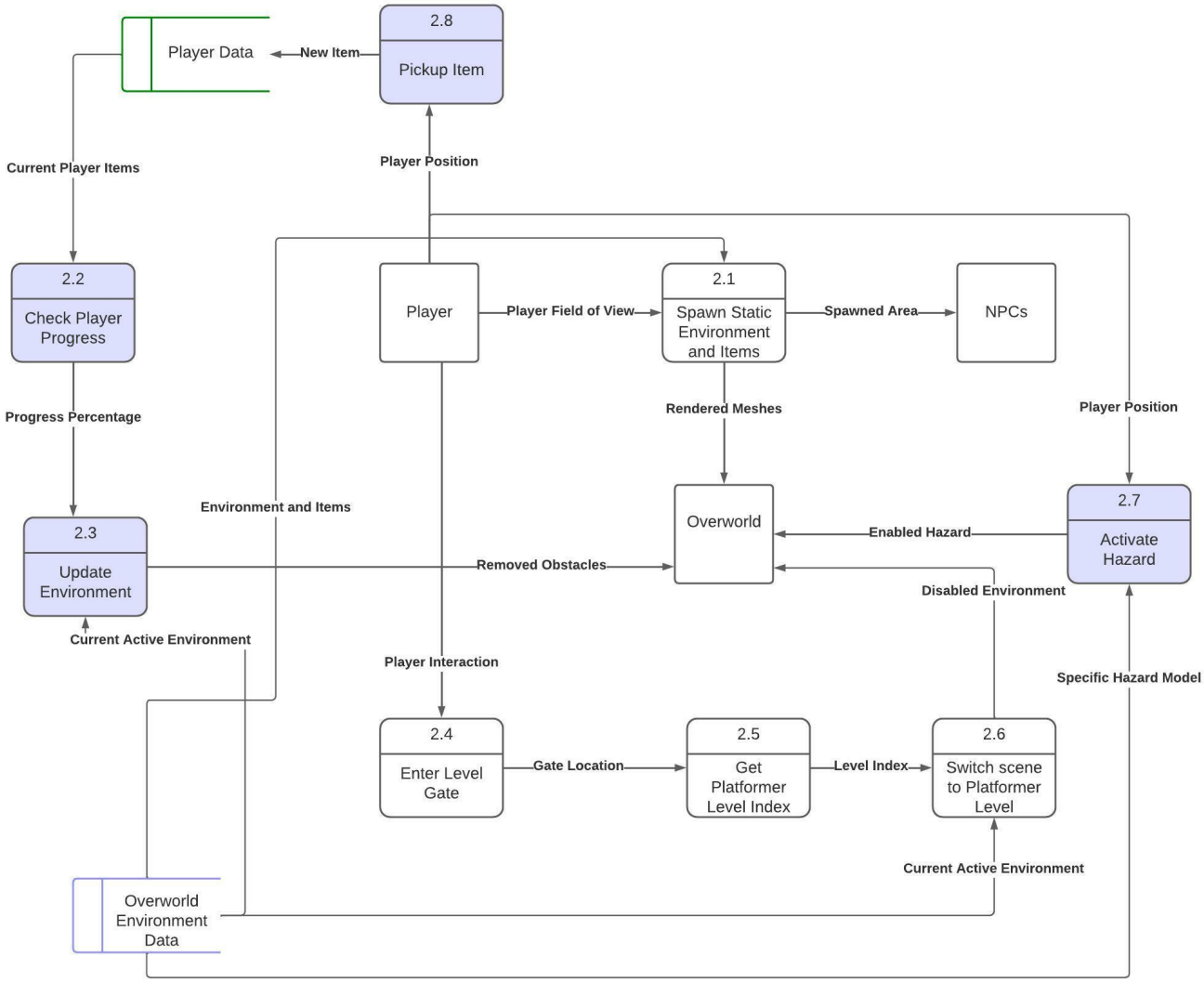
[Get the Level 0 from your team. Highlight the path to your feature]

In the data flow diagrams below, I'll highlight the Overworld Environment feature in its entirety, through using the "Create Overworld" process and multiple sub-processes, going all the way down to process descriptions for multiple sub-processes.

### Data Flow Diagrams


User
Controller
Keyboard
Mouse
0
Dungeon Jump
Graphics
Audio
User


5
Generate Platform NPC's
NPC Interaction
3
Create Platform Levels
Spawn NPCs
NPC Locations
Enviornment Data
Level Data
Platform Environment Data
Environment Data
4
Interact With Overworld NPC
NPC Locations
Platform Data
Inventory
Update Player Data
Environment Data
Player Data
Overworld Environment Data
Overworld Data
UI
2
Create Overworld
Spawn Player
New Player Data
Player Items and Location
Overworld Progression Status
1
Show Menus and Overlays
Quest Completion/Progress
6
Player Interaction
Interact With Player
NPC Interaction
Menu Select
Player Interaction
Menu Display




## Process Descriptions

The process description for process 2.7, “Activate Hazard”, is shown below through structured english:

if player position == hazard’s trigger position
    instantiate hazard model into the overworld

The process description for process 2.8, “Pickup Item”, is shown below through structured english:

if player position == item position
    Add to player inventory
    Delete item from the world

The process description for process 2.2, “Check Player Progress”, is shown below through structured english:

progress percentage = 100 * Count( current player items ) / Count( total player items )

The process description for process 2.3, "Update Environment", is shown below through structured english:

```
if progress percentage >= 25 && area1Complete == false
        area1Complete = true
        remove blocking obstacle(s) to area 2
if progress percentage >= 50 && area2Complete == false
        area2Complete = true
        remove blocking obstacle(s) to area 3
if progress percentage >= 75 && area3Complete == false
        area3Complete = true
        remove blocking obstacle(s) to area 4
if progress percentage >= 100 && area4Complete == false
        area4Complete = true
        remove blocking obstacle(s) to victory area
```

## 4. Acceptance Tests _____9

The acceptance tests will go through the testing of the main features of the overworld: Level Gates, Item Pickup, and Damaging Hazard Encounters.

**Level Gate: expected input**

I'll create a test script to wait 1 second after creating the overworld, then teleport the player on top of a Level Gate. The waiting is necessary to make sure everything is initialized correctly. The input will be the player character's collider and the level gate trigger, and the corresponding output should be a change in scene to a platformer level. In order to automate this testing, we'll have to override the function being called to change the scene, or include a "testing" variable within the change scene script. Otherwise, we won't be able to test for output in an automated fashion, since the scene will change and our script won't be able to restart the test. So, now that the test doesn't change the scene, we can restart the scene after another 1 second of waiting and repeat 100 times. If the platformer level transition function is called, increment a static variable. The expected output should be a variable with 100 as its value. Assuming the variable is initialized at zero. I'll make the number of tests variable public so we can change it to do less tests since waiting 200 seconds is 3 minutes.

**Level Gate: unexpected input**

I'll do the exact same operations as described in the "expected input" case for the Level Gate, except I'll teleport an NPC to the level gate instead. Theoretically, nothing should happen since an NPC shouldn't be able to trigger a level change. So, the input is the NPC's collider and the Level Gate trigger, and the output is a variable with a value of 0. Assuming the variable is initialized at zero and only increments when the change scene function is called.

**Trigger Hazard**

I'll create a test script to wait 1 second after creating the overworld, then teleport the player on top of a hazard. First, I'll store the player's current health in a variable. I'll then wait 0.5 seconds and if the player's health is lower than it originally was, I'll increment a static counting variable. I'll then restart the scene. I'll repeat the process 100 times with a cumulative wait time of around 2 minutes. The input is the NPC's collider and hazard collider, and the output is a counting variable with a value of 100 if successful. I'll make the hazard field public so tests can be run with different hazards too.

**Pickup Item: expected input**

I'll create a test script to wait 1 second after creating the overworld, then teleport the player on top of an item. I'll wait 0.5 seconds and if that item shows up in the player's list of items, I'll add one to a counter variable. Regardless, I'll then restart the scene. I'll do this 100 times with a single item. I'll make the item field public so tests can be run with different items too. So, the input is the player's collider and the item trigger, and the output should be a variable with a value of 100.

**Pickup Item: unexpected input**

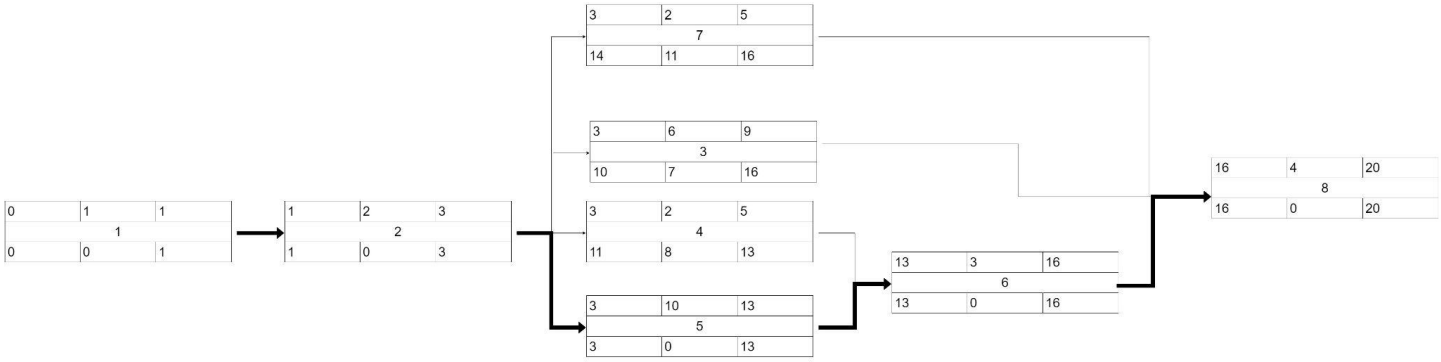
I'll do the same as the "expected input" case for Pickup Item, but I'll instead teleport the item on top of an NPC. So, the input is the NPC collider and the item trigger, and output should be a variable with a value of 0.

## 5. Timeline _____/10

### Work items

| Task | Duration (Hrs) | Predecessor Task(s) |
|---|---|---|
| 1. Research Unity Store assets for general overworld layout | 1 | - |
| 2. Integrate layout into the game | 2 | 1 |
| 3. Create Level Gates and link them to Platformer Levels | 6 | 2 |
| 4. Make/find and integrate removable obstacles to stop player progression | 2 | 2 |
| 5. Integrate overworld hazards, items, and buildings. | 10 | 2 |
| 6. Check Player for items and update world accordingly | 3 | 4,5 |
| 7. Documentation | 2 | 2 |
| 8. Testing | 4 | 3,6, 7 |

## Pert diagram




## Gantt timeline

Key:

Work Hours

Slack

Task Number