

1. Brief introduction _/3

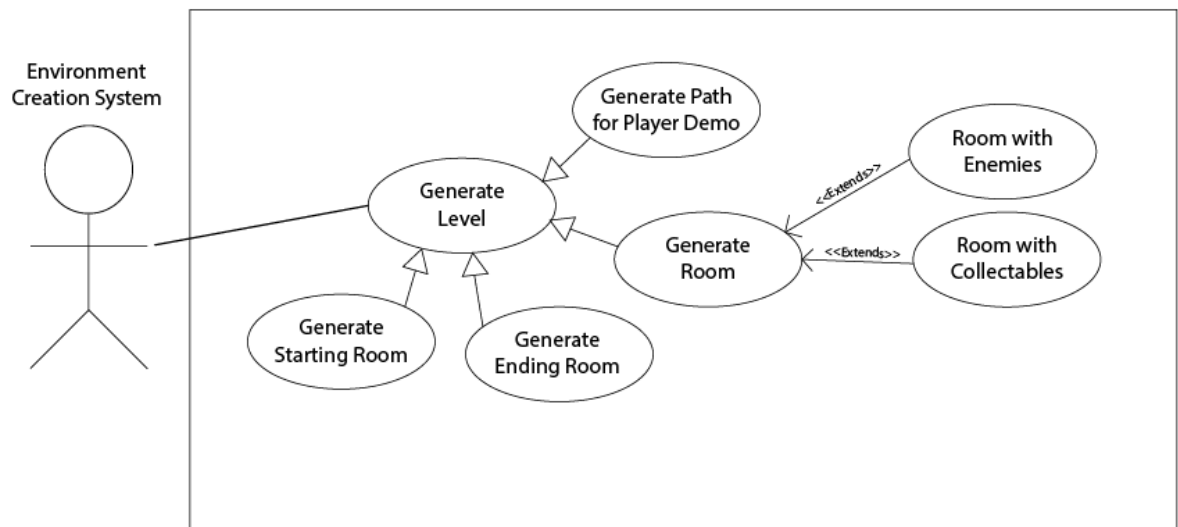
I am in charge of the level and environment design. I will be hand making tileable levels and implementing a system to procedurally connect them together. I will also be making a path for the Player AI to follow once in demo mode since I am a TL5.

Once the level has begun generating, the level will pick the starting room and ending room for the level so that it will be possible to complete the level. Then it will procedurally create more rooms in between and around existing rooms to make an immersive level with many things to do. Some rooms will also have the ability to spawn in enemies and collectables. Once the level is complete there will be a process that will calculate a path for the Player AI to follow once in demo mode. This process will be repeated for every level in the game to create unique experiences for each playthrough.

On top of this, I will also be in charge of making the game assets for the environments.

2. Use case diagram with scenario _/14

Use Case Diagrams



Scenarios

Name: Generate Level

Summary: The Environment Creation System first generates a starting room and an ending room, then it makes a specified number of other rooms that have a chance of spawning enemies and/or collectables. A path for the Player AI will then be calculated once the level has been generated.

Actors: Environment Creation System.

Preconditions: Individual Rooms have been hand made.

Basic sequence:

Step 1: Generate Starting Room.

Step 2: Generate Ending Room.

Step 3: Generate Standard Rooms for all other in-between rooms.

Step 4: Generate Player AI Path.

Exceptions:

Step 1: Based on random values the in-between room will be an Enemy Room.

Step 2: Based on random values the in-between room will be a Collectables Room.

Post conditions: Calculated value is displayed.

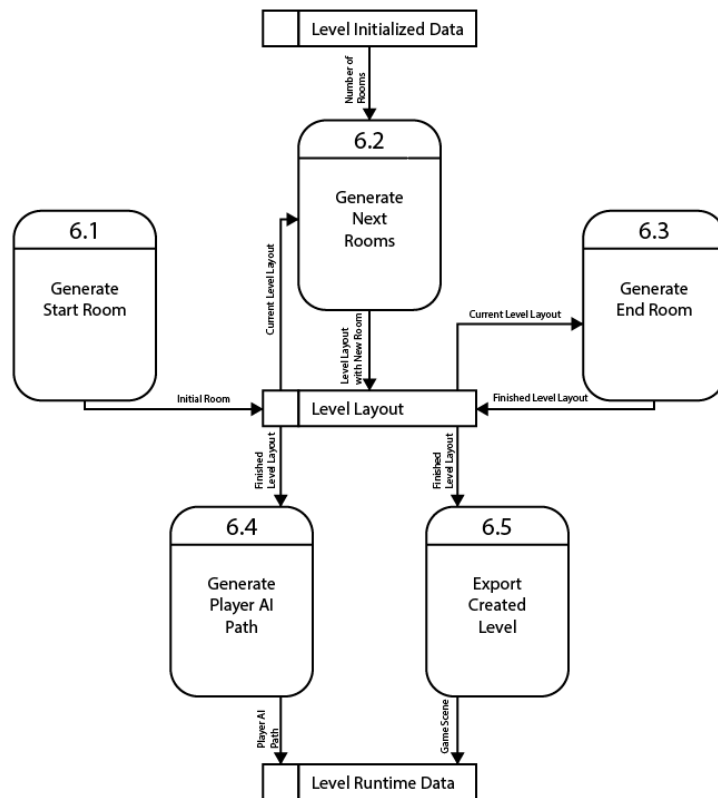
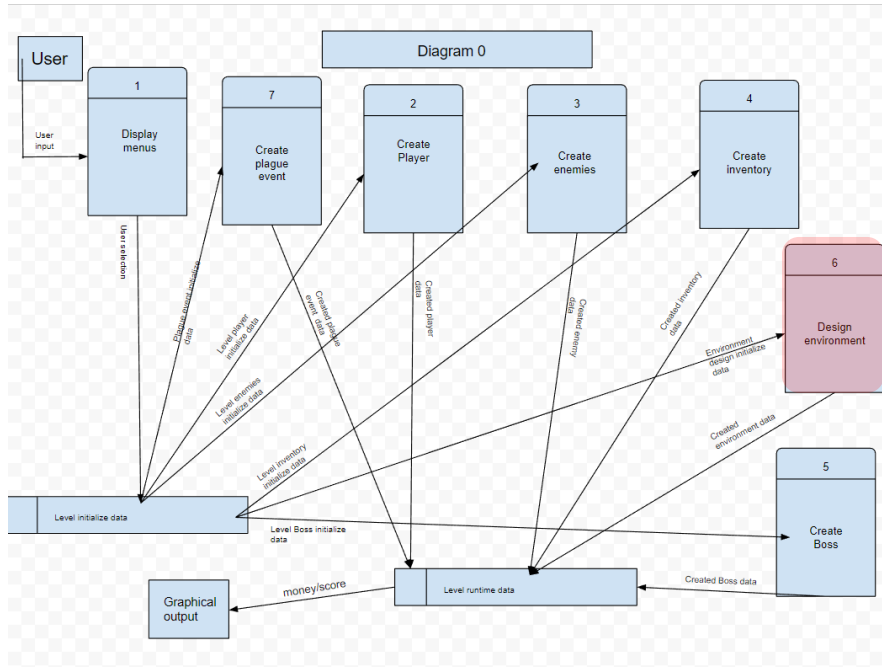
Priority: 1*

ID: E01

*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

Data Flow Diagrams



4. Acceptance Tests ____/9

This feature will have many predetermined elements, such as the Starting Room and Ending Room. The rooms themselves will also be premade, but there will be a big random factor in the order and inclusion of the rooms. There will also be a variable number of rooms that will be generated for each level.

The acceptance test will mainly test whether or not the player can indeed get from the Start Room to the End Room properly. It will also test how many of the rooms have enemies and collectables while trying to not exceed a certain percentage of all rooms. And lastly it will test to make sure that there are no duplicate rooms adjacent to each other.

The acceptance tests for these features are described below:

Room Generation:

Generate 1000 levels with the number of rooms generated being a random number between 5 and 30 rooms. The output will be sent to a file.

The output file will have the following characteristics:

- Levels must always completable.
- The % of Enemies in rooms must be between 50% and 70%.
- The % of Collectables in rooms must be between 20% and 30%.
- Adjacent rooms must not be identical.

Example for Room Generation feature

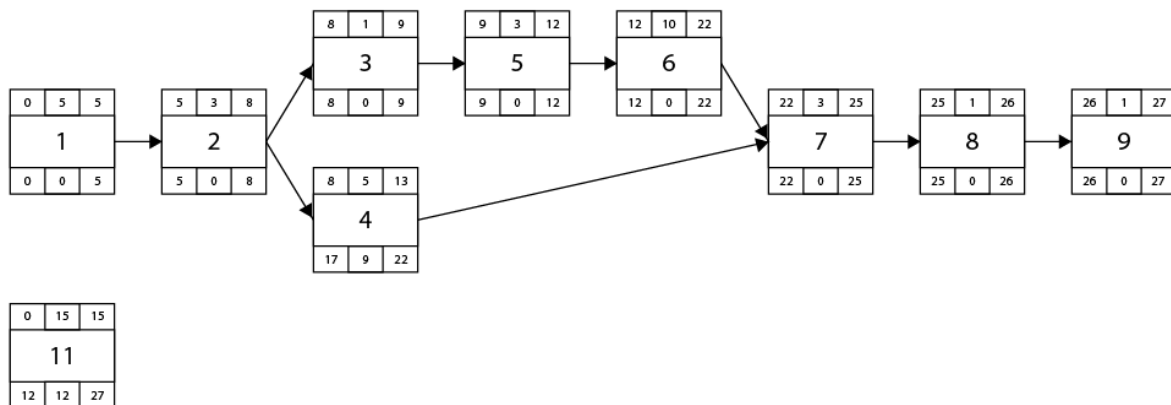
Number of Rooms	Completable	Number of Enemy Rooms	Number of Collectable Rooms	Identical Adjacent Rooms	% of Enemy Rooms	% of Collectable Rooms	Success
10	Yes	5	4	0	50%	40%	No
26	Yes	16	7	0	61.5%	26.9%	Yes
21	Yes	12	6	0	57.1%	28.6%	Yes
17	No	10	4	1	58.8%	23.5%	No
18	Yes	8	4	0	44.4%	22.2%	No
30	Yes	21	9	0	70%	30%	Yes
5	Yes	3	1	0	60%	20%	Yes

5. Timeline ____/10

Work items

Task	Duration (Hours)	Predecessor Task(s)
1. Hand-Make few Room Layouts	5	-
2. Empty Room Generation Programming	3	1
3. Implement Enemies and Collectables	1	2
4. Make Player AI Pathfinding	8	2
5. Test partial feature	3	3
6. Hand-Make rest of Rooms	10	5
7. Test full feature	3	4, 6
8. Documentation	1	6
9. Installation	1	7
10. Game Assets for Levels	15	-

Pert diagram



Gantt timeline

