

Champion document

Company: ScoSoft | Game: Scoto | Feature: Enemies and Flashlight mechanic

Name: Hayden Carroll | Date: 9/23/2021

1. Brief introduction

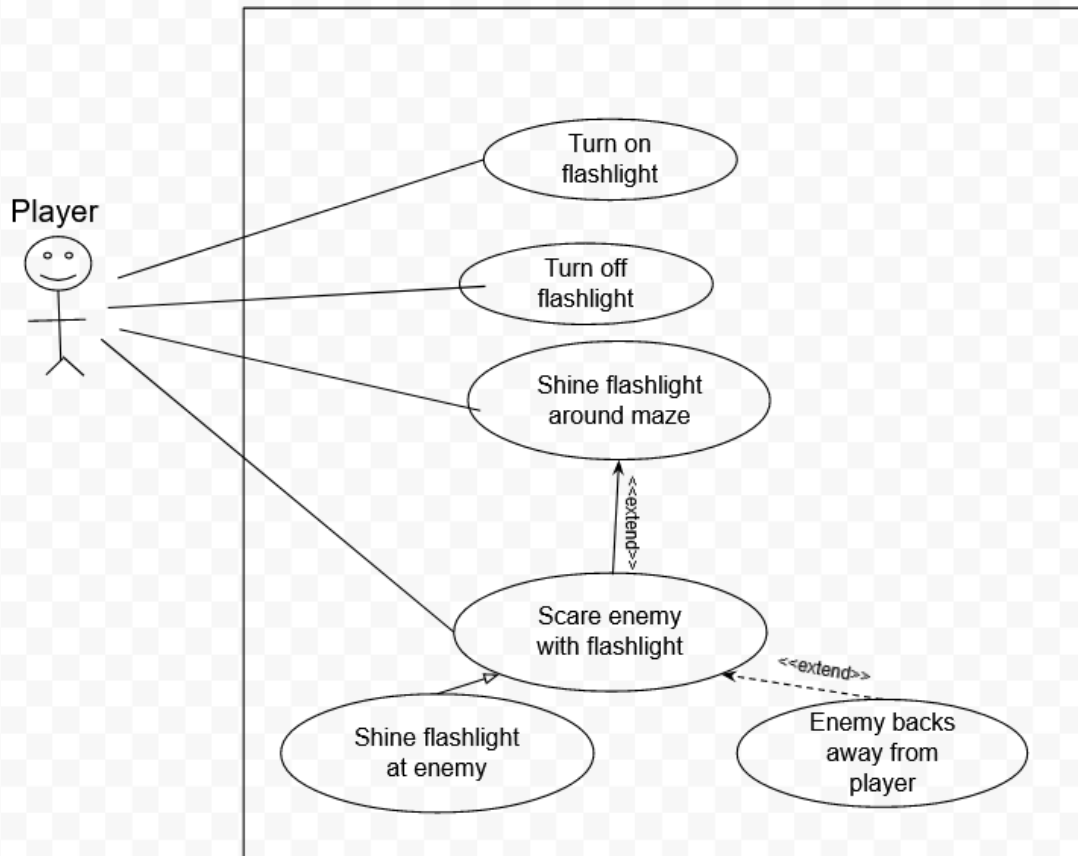
My feature for the Scoto video game is the enemy and enemy spawning, as well as the flashlight mechanic.

The enemies in Scoto are placed randomly throughout the map for the player to encounter.

There are two different types of enemies: Heavy enemies and light enemies. Heavy enemies are harder to kill and travel slower compared to light enemies. As a player, there are two main strategies to deal with these enemies. One can shoot the enemies with the guns they have obtained through Pandora's Box, or shine a flashlight at the enemies. Shining a flashlight at an enemy will cause the enemy to retreat backwards away from the player.

2. Use case diagram with scenario

Use Case Diagrams



Scenarios

Scenario 1 (first Use Case Diagram):

Name: Use flashlight

Summary: The player shines the flashlight around the maze, and can scare enemies with the flashlight.

Actors: Player

Preconditions: The player has just started a new game and is in the maze with a working flashlight.

Basic sequence:

Step 1: Player turns on the flashlight

Step 2: Player shines the flashlight around the maze

Step 3: If the player finds an enemy, the player shines the flashlight at the enemy.

Step 4: If the enemy is close to the flashlight beam, the enemy backs away.

Step 6: Load the start room and put the player in it.

Exceptions:

Step 1: If there is an enemy and the player shines the flashlight at it, the enemy may be scared.

Step 2: If the enemy is scared and is close to the player, the enemy will back away from the player.

Post conditions: The level is created and the player is currently in it.

Priority: 2*

ID: HC01

*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

Data Flow Diagrams

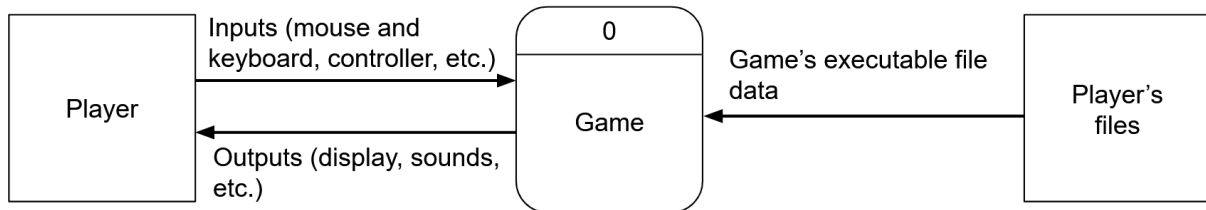


Figure 1.1: Context Diagram

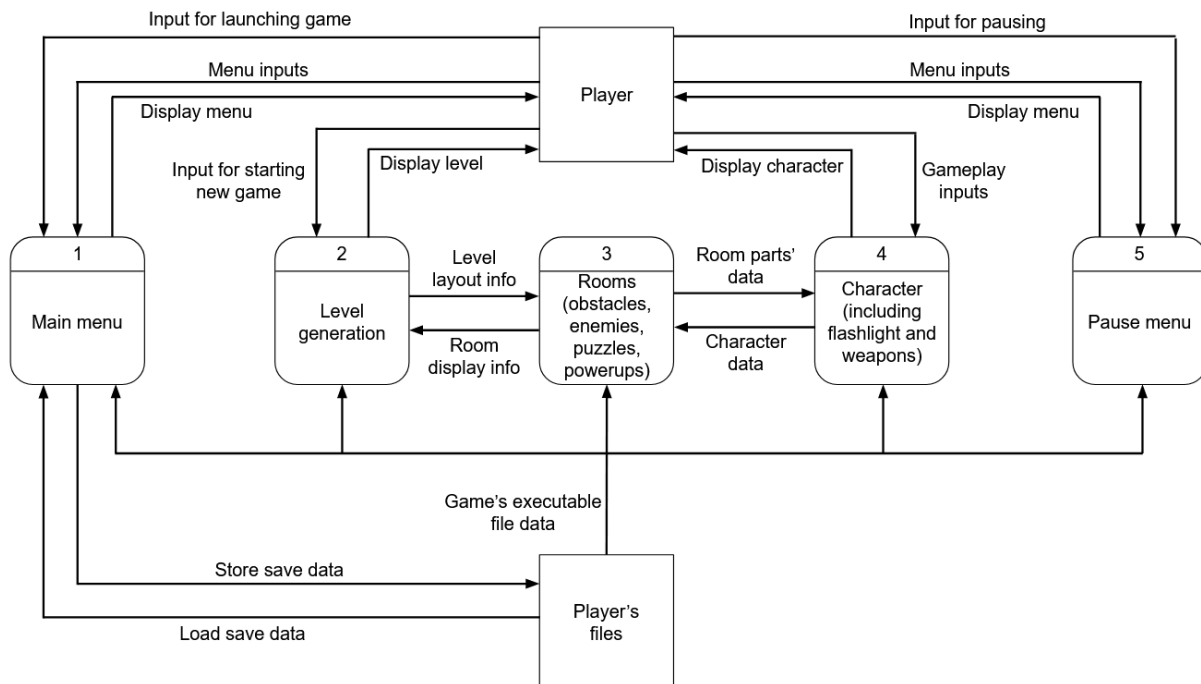


Figure 1.2: Diagram 0

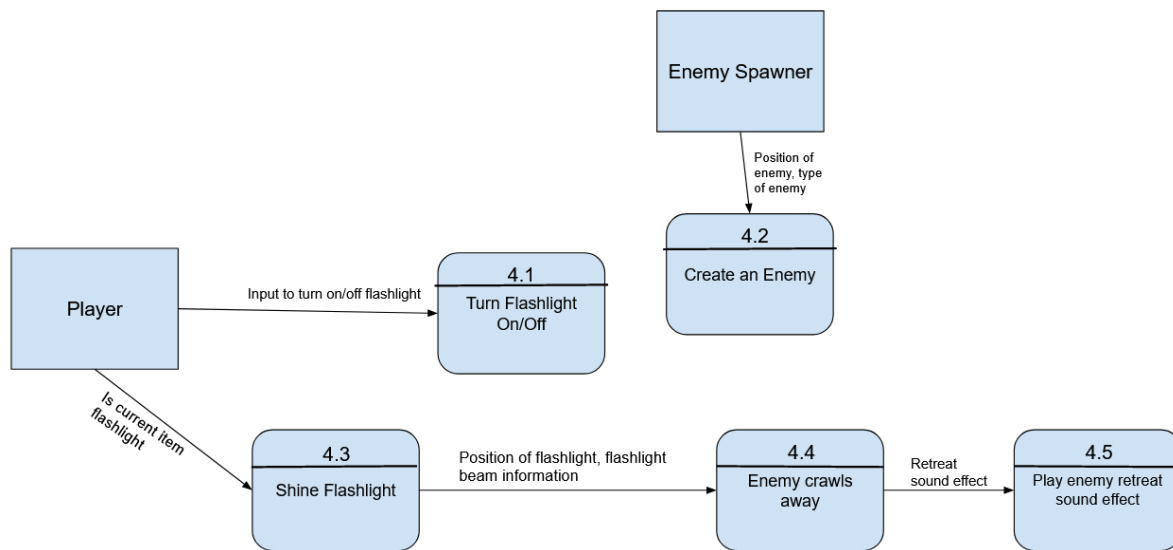


Figure 1.3: Enemy/Flashlight Diagram 4

Process Descriptions

[pseudocode/decision tree/decision table]

Structured English Process Description for 4.4 Enemy crawls away

IF Flashlight Beam hits Enemy

 IF Enemy is within flashlight beam radius

 Enemy Crawls away slowly from flashlight in a straight line

 Play enemy retreat sound effect

 ENDIF

 IF Enemy is not within flashlight beam radius

 Enemy continues walking toward Player

 ENDIF

ENDIF

IF Flashlight Beam does not hit Enemy

 Enemy continues walking toward Player

ENDIF

4. Acceptance Tests

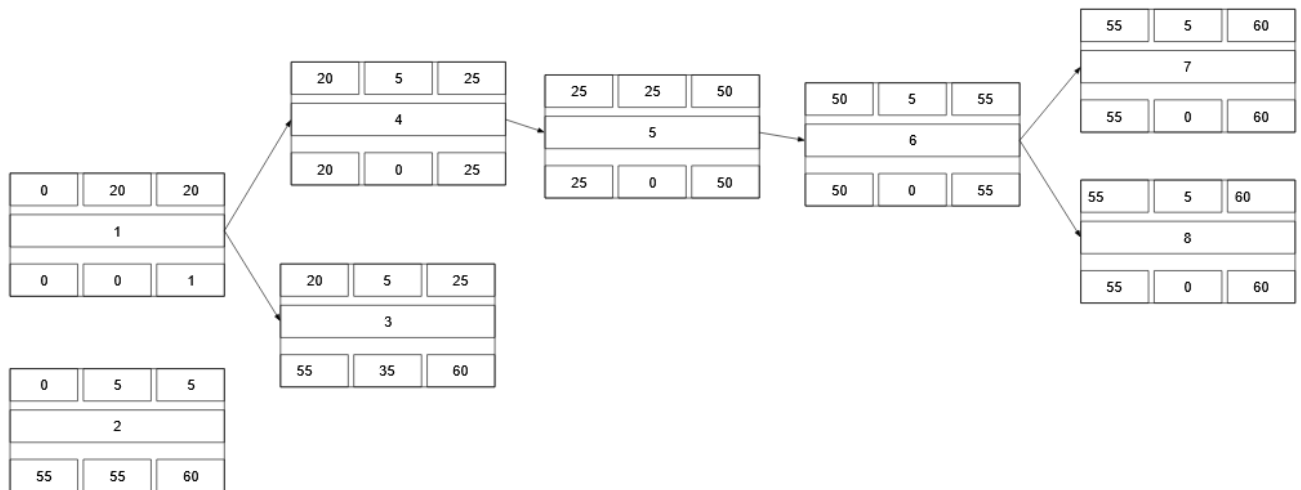
One thing to test is if the flashlight mechanic is working properly. The flashlight should correctly light up a dark space, as well as make enemies retreat. To test this, one could spawn a ton of enemies right in front of a player with an active flashlight and monitor if the enemies react. This would also test the ability of the enemies to move towards the player.

5. Timeline

Work items

Task	Predicted Duration (in hours)	Predecessor Task(s)
1 Create Flashlight class	20	--
2 Find 3D Assets	5	--
3 Integrate Flashlight class with Player	5	1
4 Create Enemy superclass	5	1
5 Create Light Enemy subclass	25	4
6 Create Heavy Enemy subclass	5	5
7 Integrate Enemy spawning with level generation	5	4,5,6
8 Integrate Enemy classes with weapons	5	4,5,6

Pert diagram



Gantt timeline

NOTE: To view the full Gantt chart, please visit the following link:

<https://docs.google.com/spreadsheets/d/1nWcPlpCmGG1wkVt009hMmXVpl7pcylWeZwahngLCbfc/edit#gid=0>

Key:

