Game Proposal: Entrapped

CPSC 427 – Video Game Programming

Team: Team 16

- Yuntian Liu 18332528
- Murtaza Dibbo 49117154
- Jiayi Zhang 23367592
- Ray Huang 79217667
- Eric Chu 78205598

1. Story

The game is a 2D top-down horror roguelike shooter where the player will explore a secret laboratory with limited lighting and resources.

1.1 Background story

In 2040, the famous Dr. Smack, a brilliant scientist from the University of British Columbia, launched the world's first Mars immigration program, even before Elon Musk's SpaceX. After the first group of settlers has established a base on Mars, you are chosen to be the second group of immigrants - 100 of the brightest and most skilled people on Earth in various fields, such as hackers, engineers, doctors, soldiers, and scientists

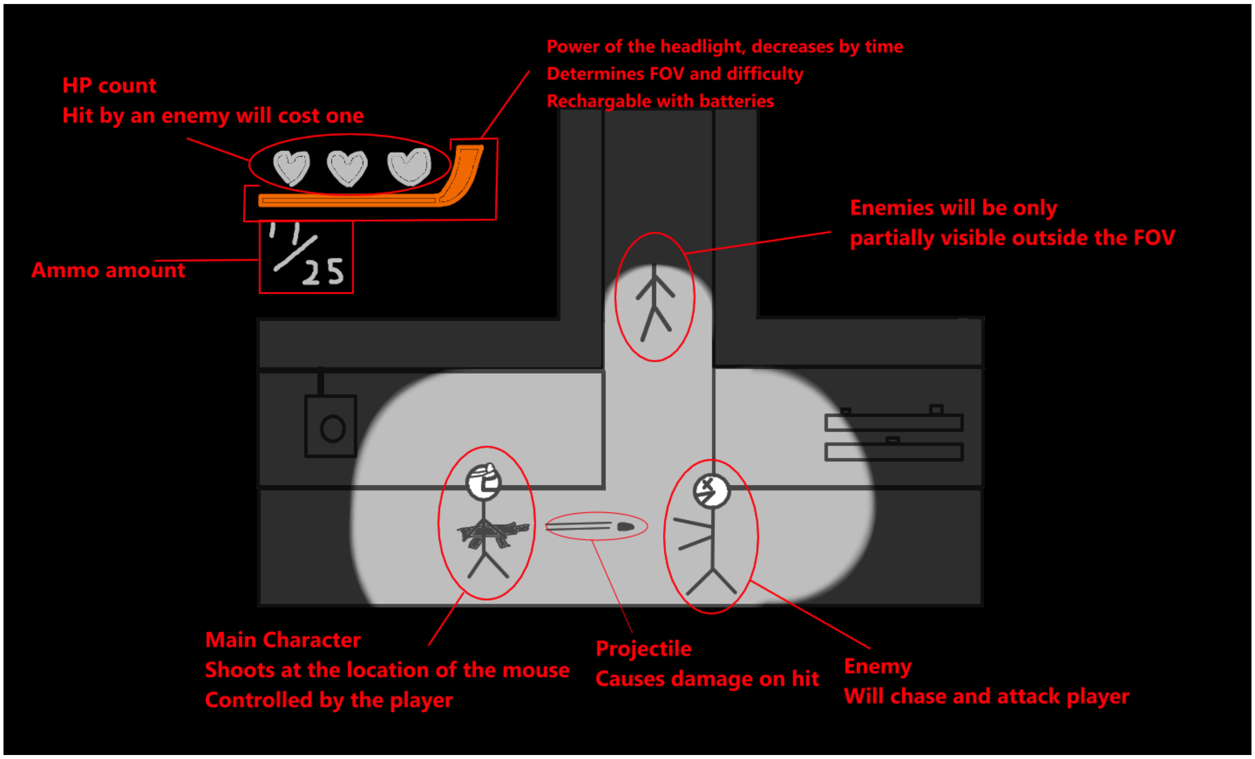
Upon arrival, you wake up in a room where you and your companion have been separated. Feeling that something is wrong, you decide to leave the room, find your companion, and try to figure out what's going on.

As an exploration, you realize that this place is a vast, maze-like facility. You are attacked by some zombies, you find the bodies of your companions, notes from the first settlers, weapons, strange laboratories, and records of past experiments in some of the rooms.

You discover the secret of the place: the fact that Dr. Smack is a Martian lurking on Earth, and that the entire Mars immigration program is a lie, it is designed to attract the best and brightest of Earth to Mars. Why would the Martians do this? It turns out that the Martians are at risk of extinction and degradation; they have a total population of less than a hundred, despite their high-end technology. So they selected the DNA of the highest quality humans for experimentation to breed a new species and guarantee the continuation of Martian civilization.

1.2 Game walkthrough

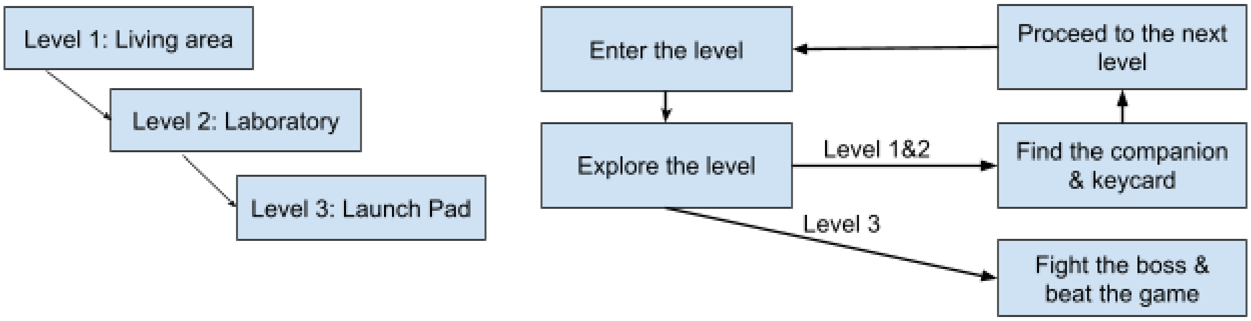
The following is a brief walkthrough of the game. For details of game mechanics, please refer to the *Scenes* and *Technical Elements* section.



HP count
Hit by an enemy will cost one
Power of the headlight, decreases by time
Determines FOV and difficulty
Rechargable with batteries
Ammo amount
11/25
Enemies will be only
partially visible outside the FOV
Main Character
Shoots at the location of the mouse
Controlled by the player
Projectile
Causes damage on hit
Enemy
Will chase and attack player

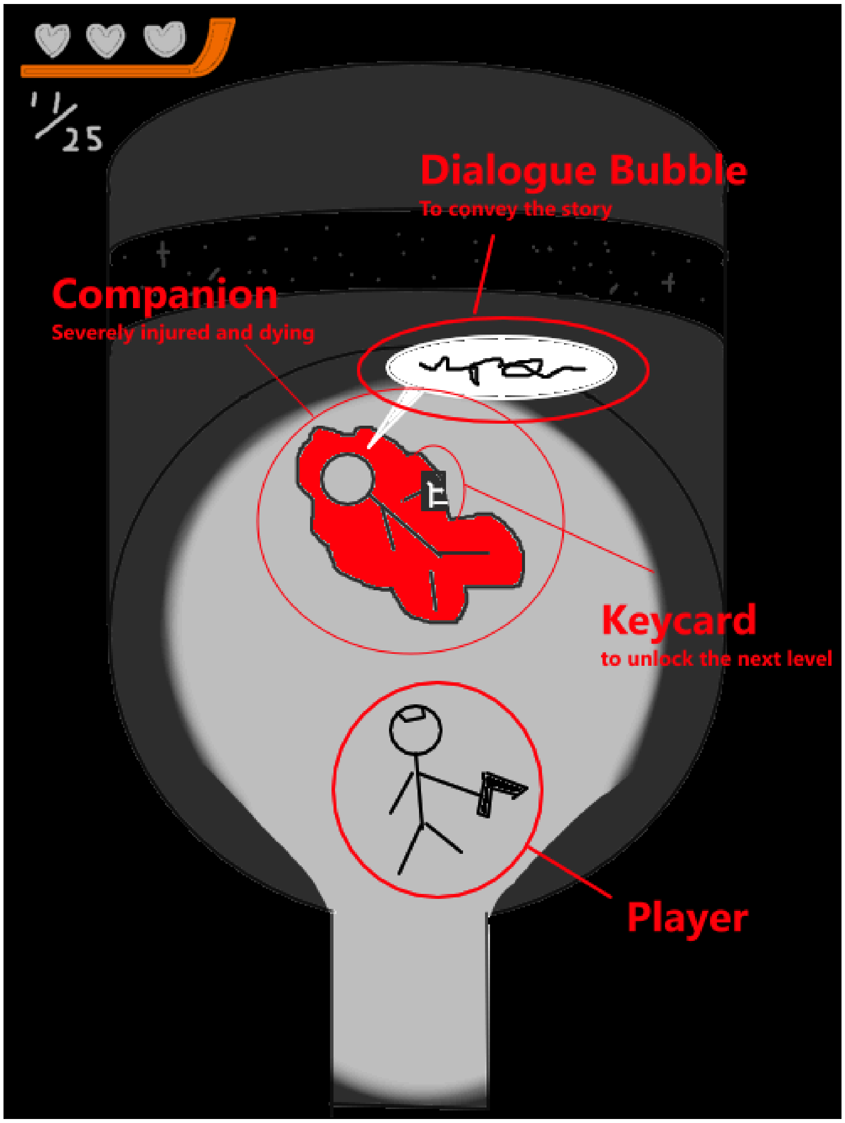
General Gameplay & HUD

The game will consist of three major levels. Each level will be an area of the facility and will have different visuals (tilesets) based on their name. As the level proceeds, the difficulty of the game also increases gradually. This is achieved by zombies of different difficulties. See 4.4.3 Enemy Spawning for detailed rules.



The player wakes up in the first level with an initial weapon, determined by the profession they chose before entering the game. The primary goal is to find their missing companions, as described in the background story. The player will need to find their missing companions in the maze-like level and retrieve their key cards to open the door to the next level.

Unfortunately, all the companions are killed. The player will be able to find a tape beside their body. The tape will play after the player interacts with it and will contain information about the background story. This is our primary way to tell the story. We will record our own voices for the tapes, and will include captions while it plays. See 3.2.3 Audio manager for more details.

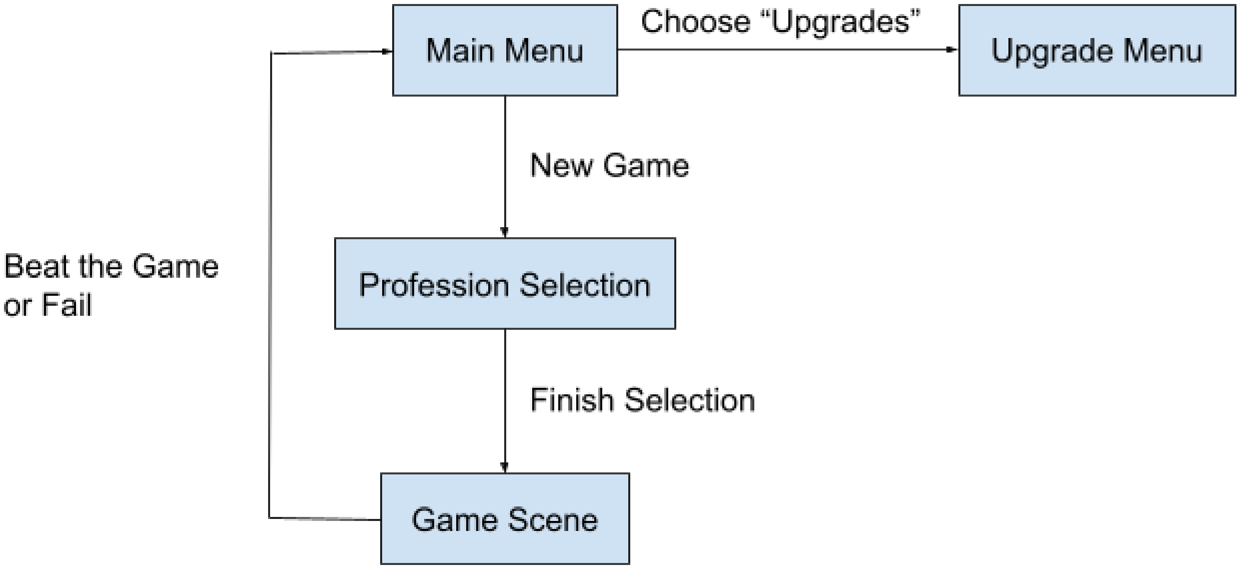


1.1 Encounter with Companion

Finally, by the end of the third level, the player meets the final boss: Dr. Smack, who will appear as a big zombie in a large room with higher than normal HP and damage. Once the player successfully defeats Dr. Smack, a brief cutscene will be shown to tell the player that they successfully escaped Mars.

2. Scenes

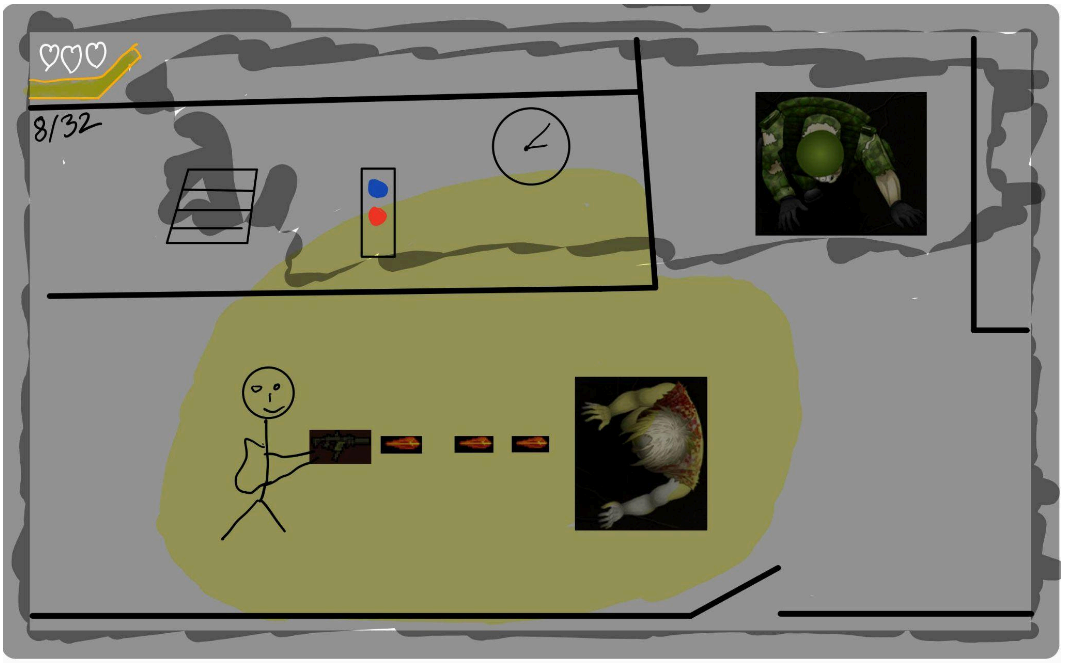
Following is a logic diagram showing how different menus and scenes can be accessed. In the following sections we will introduce what these scenes are.



2.1 Game Scene

2.1.1 Battle with Enemies

While exploring the maze, the player will encounter enemies and fight them. The player has a few options to defeat them:



Gun Fight: A scene showcasing a zombie approaching the player and the player shooting at the zombie.

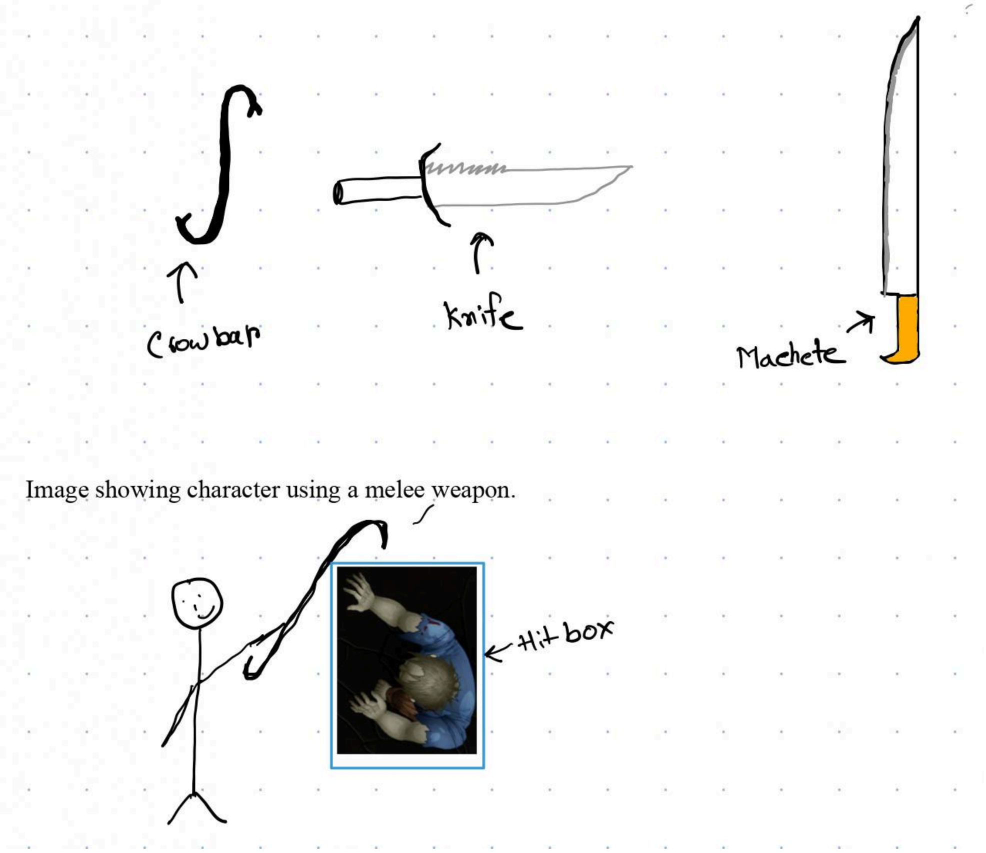
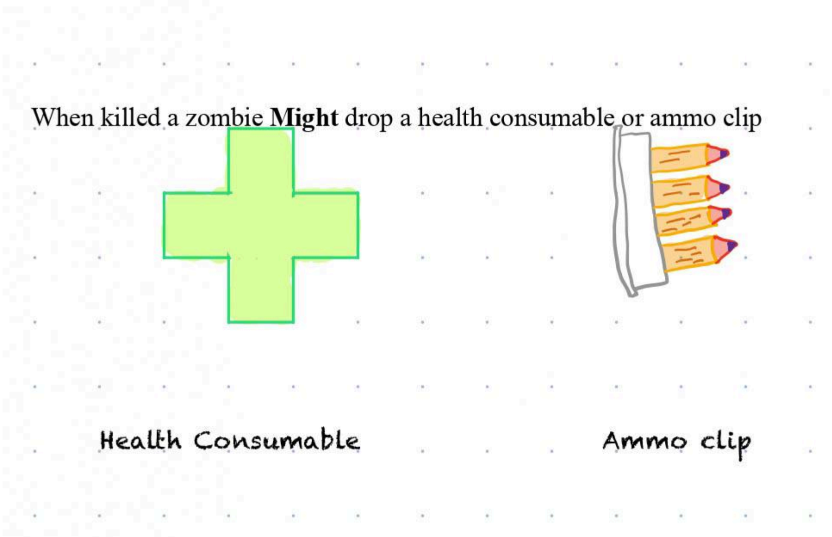


Image showing character using a melee weapon.

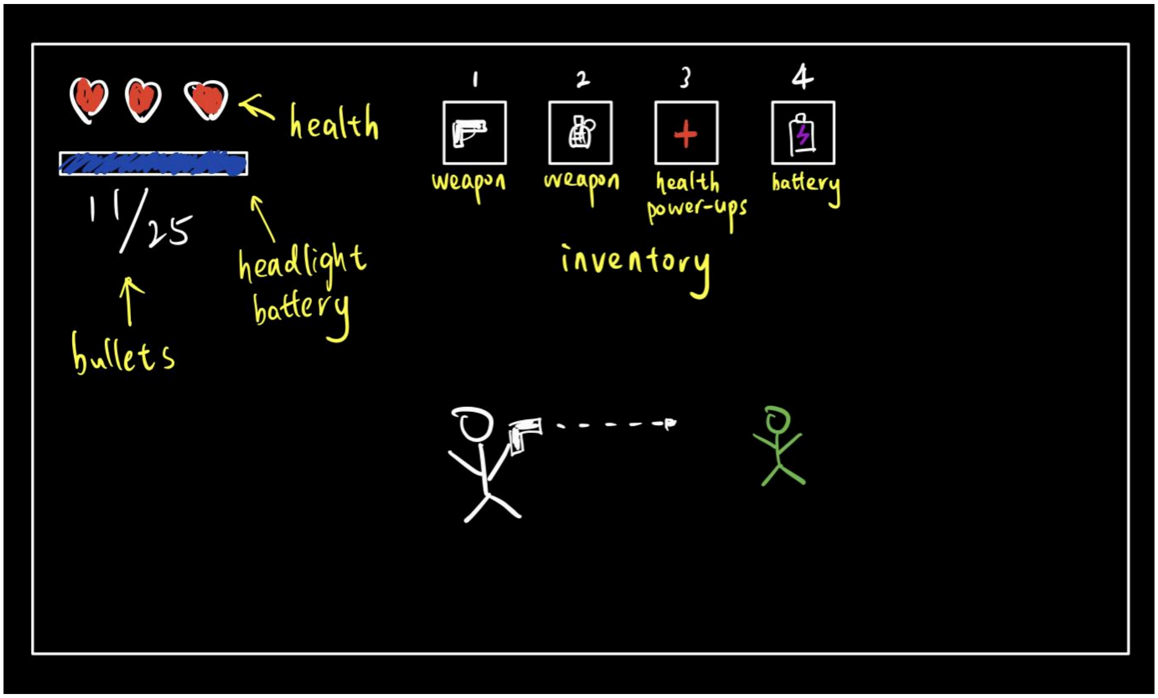
Melee fight: The player can use knife, crowbar, or machete to fight zombies, however the chances that players get bit will be much higher. Players are given an initial weapon at the beginning of the game based on their profession. See 4.5 Profession


When killed a zombie **Might** drop a health consumable or ammo clip
Health Consumable
Ammo clip

Reward drop: Sometimes when the Zombie dies, it will drop ammo clip/health consumable/……. or some sort of reward. This scene shows the reward. Health consumables heals the player and ammo clips increase the amount of ammo that the player has. The health consumable can be collected and used later, however the ammo clip will take effect on picking up. The drop chance is fixed for each type of item. Refer to 4.3 Consumables for more details.

2.1.2 Resource Management

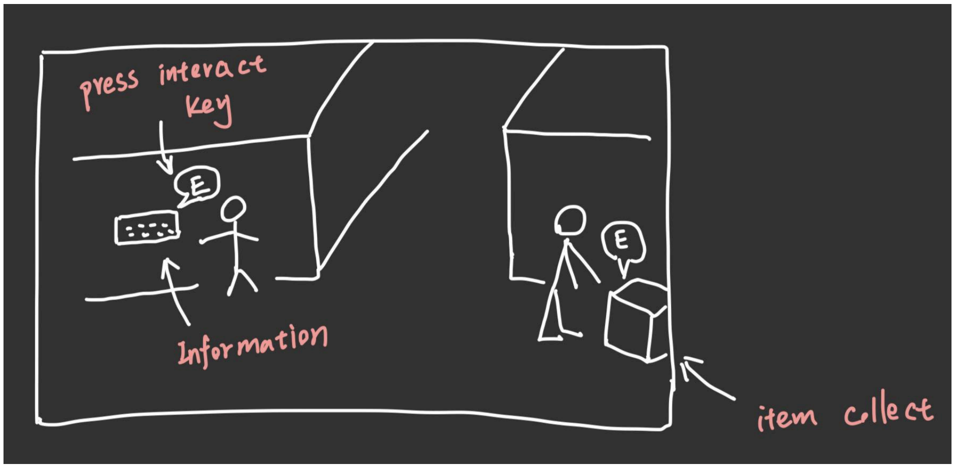
Following is a concept art of UI of the health bar, headlight battery, bullets remaining, and inventory. They constitute all the resources that the player needs to manage throughout the gameplay. Refer to 3.4.3.1 In-game states for more details of the stats of the player



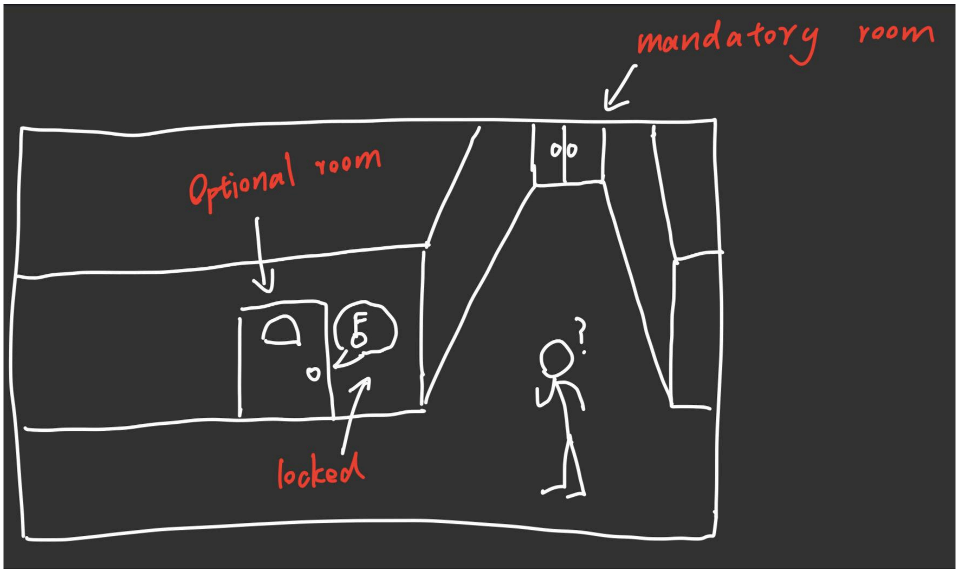
- Inventory system
    - Players can switch weapons, use health power-ups, and change the batteries of the headlight by pressing hotkeys (1, 2, 3, 4).
    - In addition, they will also collect grenades and consumables (health and battery) during the game. See 4.3 Consumables for what kind of consumables may be collected.
- Health
    - The player will die and return to the main menu if their health runs out.
- Headlight battery (Also called stamina in the following sections)
    - FOV decreases as the battery level decreases. It also determines how many zombies will respawn around the player every certain period.
- Ammo
    - Player’s ammo. If the ammo in the magazine runs out, a reloading will be required. If the ammo runs out completely, the player will only be able to use their melee weapon.

2.1.3 Exploration of the maze

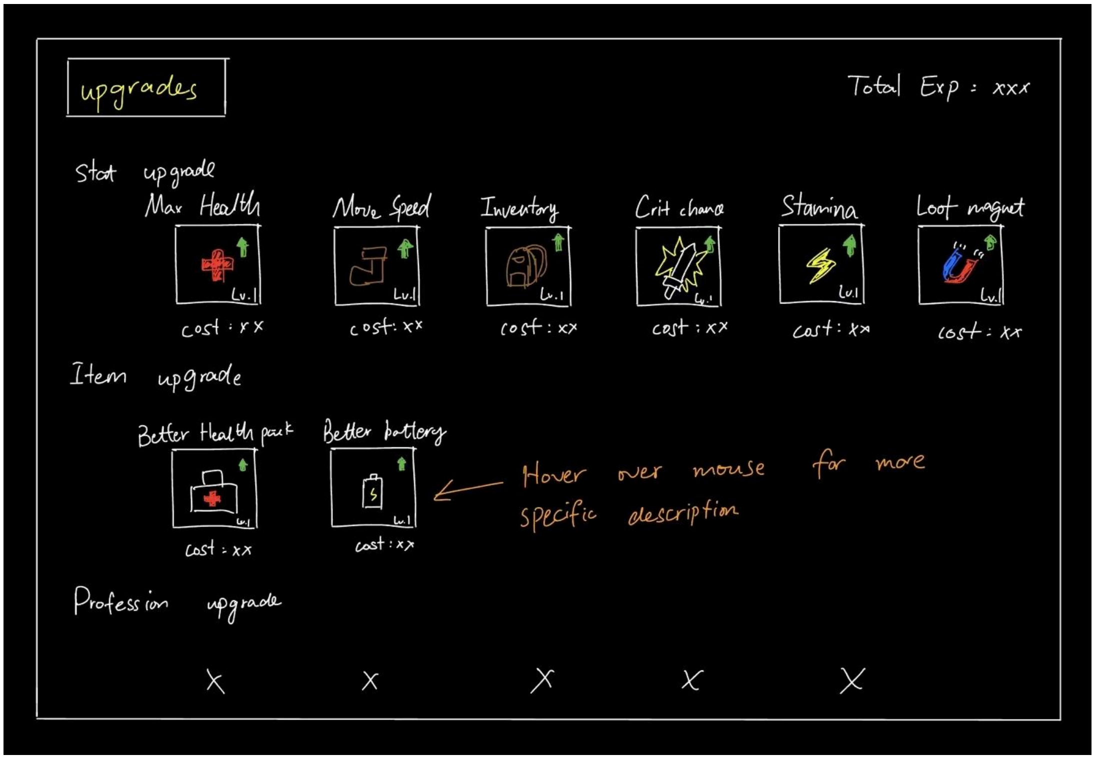
- **Enemy**: enemies are in the maze walking around and start chasing and attacking the player when they find the player. Their behavior is controlled by an AI system that we plan to implement. Refer to 3.4 Gameplay logic/AI for more details.
- **Interaction with the environment**: The player can interact and check the walls and objects in the maze to get clues or items (notes, information about the facility, headlight batteries, healing items, etc.)



- **Rooms**: There are many rooms with different functionalities in the maze. One type of room can be chosen to enter or not, while the other type of room must be entered, otherwise there will be no way out of the maze. However, some rooms are locked and require the player to find a key to open them. Refer to 4.4.1 Rooms for the kinds of rooms that will be in the game.



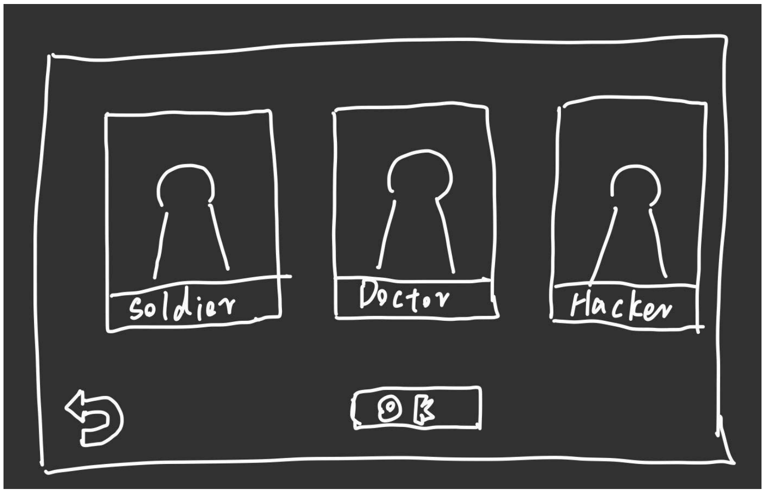
2.2 Upgrade Menu

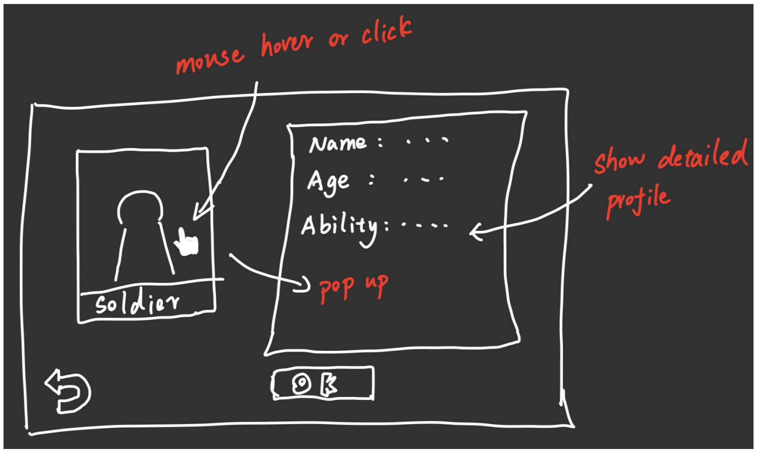


Hover over mouse for more specific description

From the main menu, the player can choose to enter the menu where they can purchase upgrades that will be effective when they start a new game. Refer to 4.2 Permanent upgrades for implementation and design details of the upgrades.

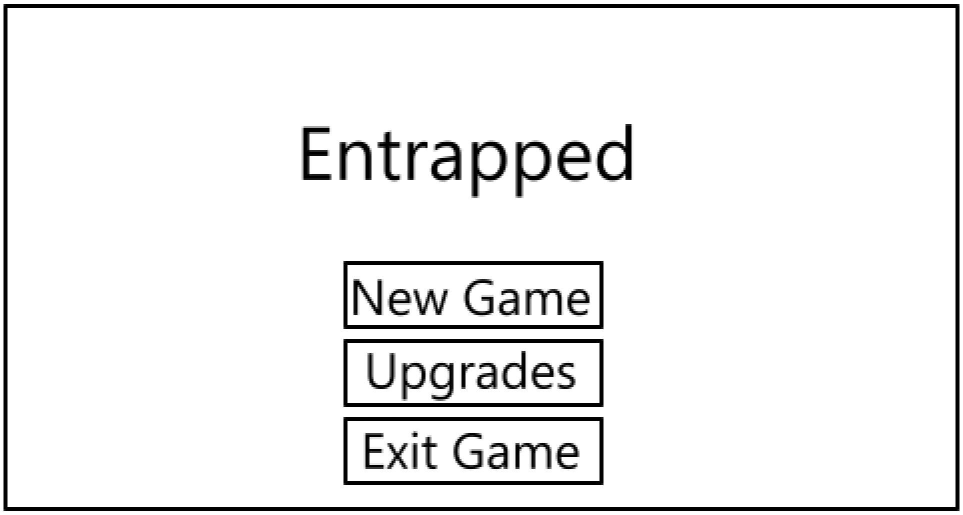
2.3 Profession Selection





Every time the player starts a new game run, a profession selection menu will appear to prompt the user to choose a profession. The player needs to select a profession to enter the game, which will then determine their initial weapon and special abilities. Refer to 4.5 Profession for more details.

2.4 Main menu



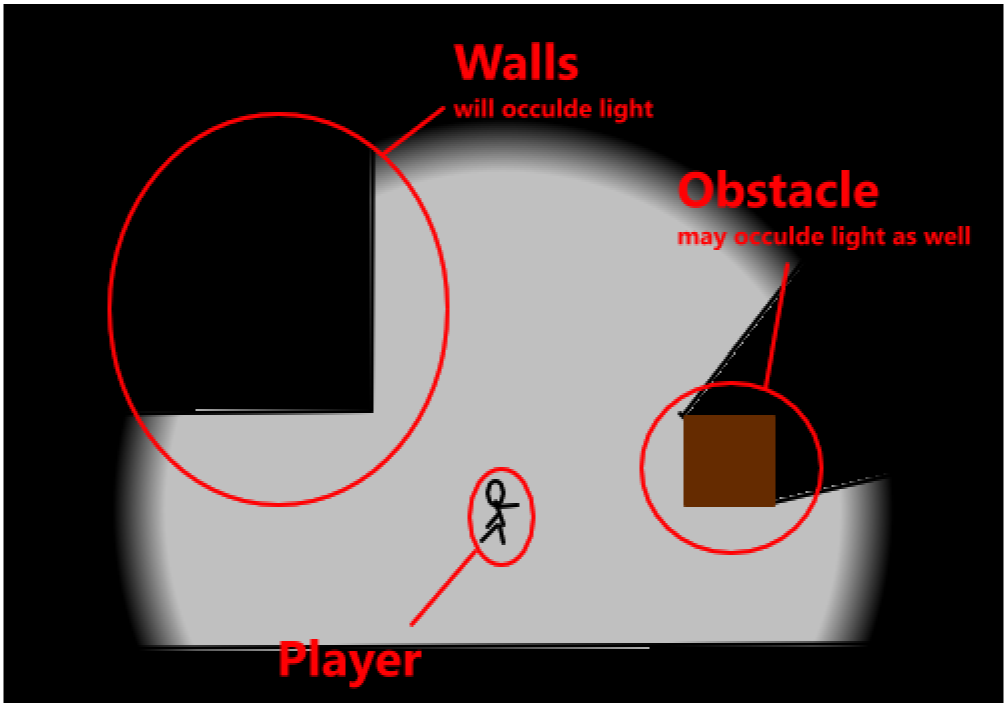
The first thing every player will see when they open the game.

- **New Game**: This will open the profession selection menu
- **Upgrades**: This will lead to the upgrade menu
- **Exit Game**: End the game process

3. Technical Elements

3.1 Rendering

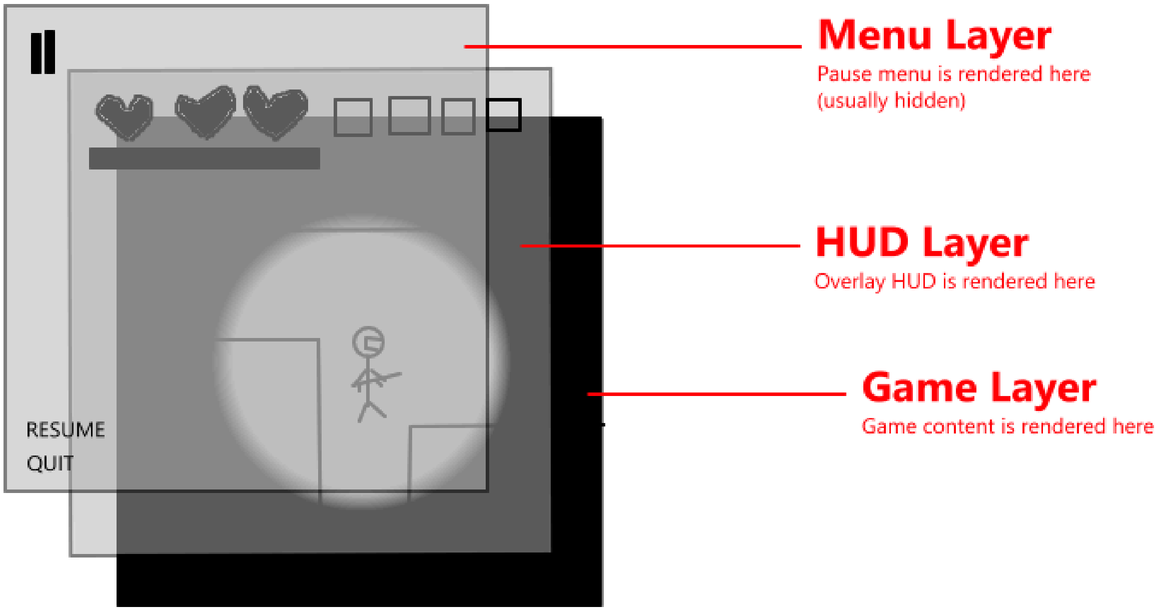
3.1.1 FOV



FOV

- Player could only see a certain radius away from the main character. Visually, it looks like a lighting mask
- The vision may be obscured by objects and walls and the player is unable to see anything behind
- Enemies outside of the FOV cannot be seen
- FOV's radius may be upgraded or impacted by the headlight battery. Hence it is a variable
- This is one of the core mechanics of our game and we will prioritize it.
- If we cannot fully implement the occluding effect, we will just allow the players to illuminate the circular area around them ignoring any obstacles.

3.1.2 UI Layer


RESUME
QUIT
Menu Layer
Pause menu is rendered here (usually hidden)
HUD Layer
Overlay HUD is rendered here
Game Layer
Game content is rendered here

UI layers

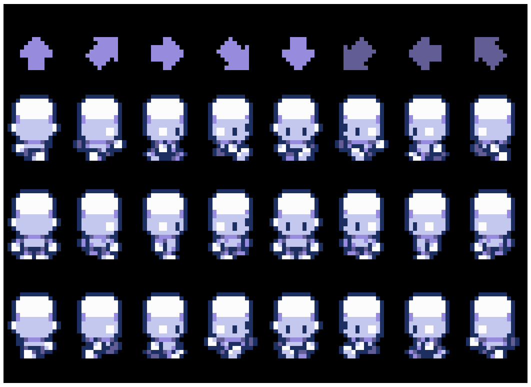
- While the game is going on, we will render visual elements to three layers
  - Menu Layer
    - This layer contains the pause menu. The player can quit the game from here.
    - Hidden unless the game is paused
    - Should contain mostly static UI elements
  - HUD Layer
    - This layer contains HP bar, battery power, ammo and consumable slots
    - Will be updated according to the state of the player
  - Game Layer
    - This layer contains most of the game sprites and FOV mask
    - Will be updated every single frame
- This is one of the core mechanics of our game and we will prioritize it.

3.1.3 Camera Following

- From the start of the game and throughout the entire game, the camera will be centered at the location of the player, unaffected by any input.
- We may implement this by manipulating the camera matrix with openGL.
- This is one of the core mechanics of our game and we will prioritize it.

3.2 Geometric/sprite/other assets

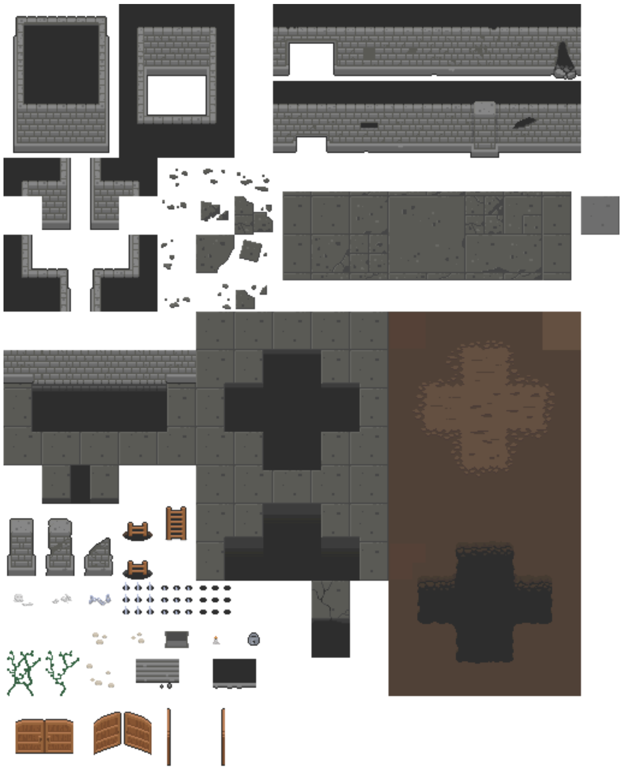
3.2.1 8-directional movement



Example 8-directional movement sprite, taken from https://axulart.itch.io/small-8-direction-characters

- Both the player and enemies will move in 8 directions.
- There will be a sequence of frames for animation corresponding to each direction.
- We will get free assets like this on Kenny.nl or itch.io. The example is obtained from itch.io.
- This is one of the core mechanics of our game and we will prioritize it.

3.2.2 Tileset and tile map



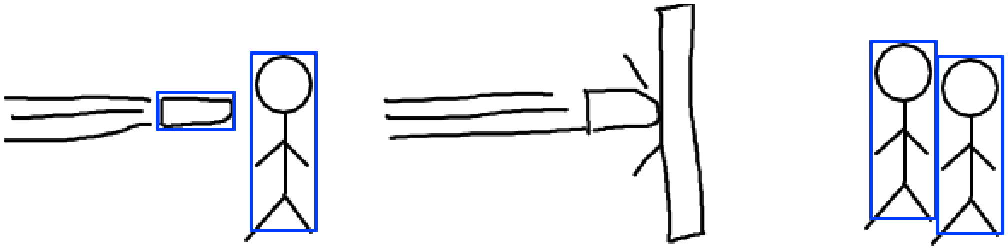
Example tileset, taken from https://elthen.itch.io/2d-pixel-art-dungeon-tileset

- We will use tilesets and tile maps to draw our map. This means that our map will be a plain 2d image that does not transform according to the player's perspective.
- A grid system will need to be implemented to render the map based on the tilemap: each tile in the asset will have a unique ID and the game will render each tile to their grid based on the ID we choose.
- Like the movement sprite, we will get free assets like this on Kenny.nl or itch.io. The example is obtained from itch.io.
- This is one of the core mechanics of our game and we will prioritize it.

3.2.3 Audio manager

- For our game, audio manager will be implemented for three types of audios
    - Background music
        - Simple, looping background music that is played infinitely.
    - Sound effect
        - Sound effect of interaction and battle, such as opening a door or death of an enemy.
        - Other systems will need to be able to call the audio manager to import and play the sound effect they desire.
    - Tape
        - Recorded messages to explain the story.
        - Captions will be rendered to the HUD layer while the tape is playing.
        - Will stop once the audio is over or the player chooses to proceed to another level.
        - If possible, we may implement a multi-channel audio player to play the tape and the background music at the same time.
        - If unable to implement it, we will pause the background music when the tape is playing
- This is one of the core mechanics of our game and we will prioritize it.

3.3 2D geometry manipulation: bullet and entity collision

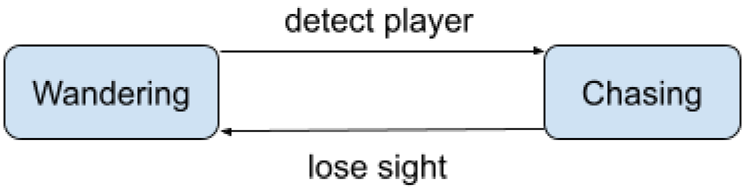


- There are four elements in our game that have collision: player, enemy, bullet and walls (obstacles)
    - Bullet:
        - Bullet will cause damage to the enemy if it overlaps with the enemy and then destroy itself.
        - Bullet will destroy itself if it hit the wall.

        - Bullet will pass through the player.
    - Player
        - Player will collide with enemies and walls.
        - When colliding with an enemy, the player will be damaged.
    - Enemy
        - Enemies will collide with other enemies, the player and walls.
        - When colliding with a player, the enemy will not be damaged.
- This is the core of our game and we will prioritize its implementation.

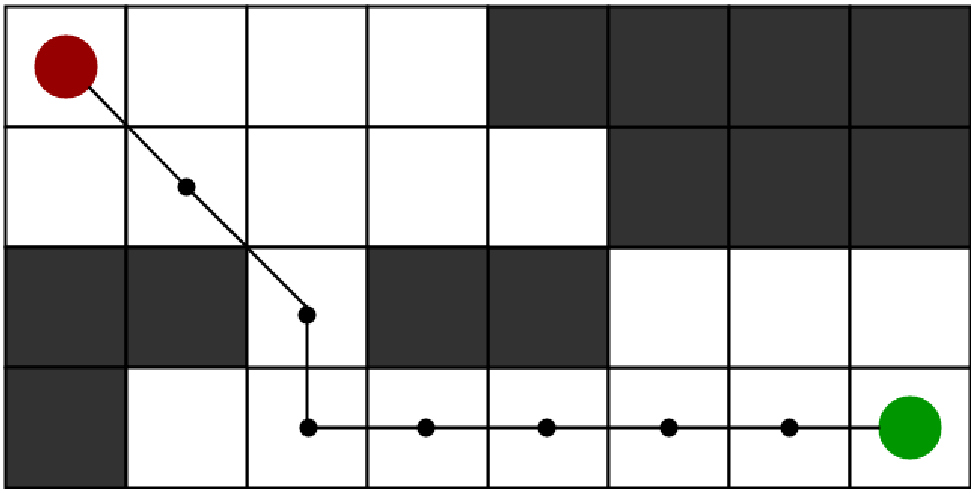
3.4 Gameplay logic/AI

3.4.1 Enemy behavior



- Most of the enemies in this game will be share similar AI logic
- There are two state of the enemy
    - Wandering
        - The enemy will randomly move around the level. For every given period, the enemy will attempt to move for a constant distance to one of the eight directions
        - After the player moves into a certain radius of the enemy, the enemy will transit to chasing state
    - Chasing
        - The enemy will keep moving towards the player
        - After the player stays outside of a certain radius of the enemy, the enemy will transit back to wandering state
- This is the core of our game and we will prioritize its implementation.

3.4.2 Path finding



A* pathfinding, taken from https://www.geeksforgeeks.org/a-search-algorithm/

- We will need to implement some path-finding algorithm in order for the enemy to navigate among obstacles to chase the player, such as A* search algorithm.
- The map will be divided into small grids. When the enemy is in chasing state, it will attempt to find a path to the grid of the player.
- The enemy will try to avoid obstacles and walls. If we have time to add a safe room, the zombie will not attempt to move into the safe room. See 4.4.2 Rooms
- Similar grid system may also be used to manipulate tilesets.
- If we are unable to fully implement the path-finding, we may use naive path finding. That is, the enemy will also move towards the location of the player ignoring the obstacles in between.

3.4.3 State management system

A systematic state management system will be implemented to track the player's in-game status as well the the save/load of the data

3.4.3.1 In-game states

In-game states will not be stored to disk and will be lost if the game process is terminated. These are data related to the current game session. We expect the player to complete the game without closing it like in many roguelike games such as Risk of Rain. Below introduced what data will be included in the In-game states

- Health
  - Player's primary resource. Once HP reaches zero, the player will die and be returned to the main menu.
  - Can be restored with consumables
- Headlight battery, or may also be called stamina

    - The Headlight battery level is proportional to the FOV of the player. It also determines how many zombies will respawn around the player every certain period. See 4.4.3 Enemy Spawning for more details
    - Zombies will respawn at an extreme rate if the power completely runs out. This drives the player to maintain their power level throughout the gameplay.
    - However, even if the battery runs out. The player will still be able to have a small FOV around the character instead of losing all the sight. The latter is too frustrating as the player will not be able to see anything.
- Consumables and power-ups
    - Amount and types of consumables and power-ups that the player obtained so far. Refer to 4.1 In game power-ups and 4.3 Consumables for details of these two game elements.
- Ammo
    - Amount of ammos that the player has. The player will need ammo to use ranged weapons or else they can only use melee weapons.
- Player stats
    - **movement speed**: The speed at which the player moves.
    - **critical hit chance**: There is a chance that the player’s attack may cause extra damage.
    - **critical hit damage**: The percentage of the extra damage that the player may cause on critical hit.
    - **loot magnet radius**: The Minimum distance that the player can pick up consumables and power-ups at.
    - Some of these stats may be removed if we do not have enough time to implement them.

3.4.3.2 Persistent data

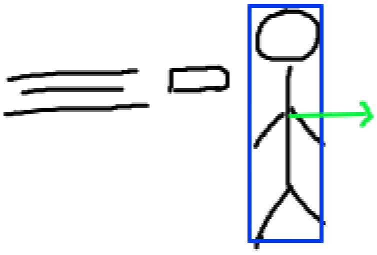
Persistent data refers to the data that will be stored to disk when the game process ends and loaded when the game process starts. Below introduced what data will be included in the In-game states.

- Experience point
    - The player will get experience points throughout their game experience. These points will be kept even if they lose or quit the game.
- Active permanent upgrades
    - The player can purchase permanent upgrades for their future games. These upgrades will not be lost as well.

Refer to 4.2 Permanent upgrades for more details on the rules of these two data.

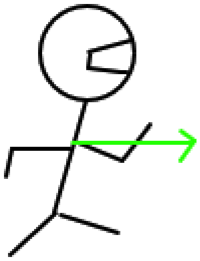
3.5 Physics

3.5.1 knockback



- Every time the enemy is hit by a projectile, it will be knocked back for a short distance i.e. a force vector will be applied for a short period of time.
- The direction of the knockback is always away from the player, no matter what angle the bullet hits the enemy with.
- If for some reason we cannot finish this functionality, we will instead just shift the zombie some distance away from the player on hit.

3.5.2 sprint



- Every time the player presses space, the player will attempt to sprint. This will give the player a momentum to the direction that he/she is moving towards.
- There is an internal cooldown for the sprint.
- If time does not permit us to finish this feature, we can just make the player move faster when the sprint button is pressed

4. Advanced Technical Elements

Following are more advanced technical elements specific to our game. We are planning to complete all of them, but we will focus on items of higher priority first.

4.1 In game power-ups (high priority)

Shield
block the next attack from the zombie.
Flare
have unobstructed vision for the next 15 seconds

Example power-ups

- While exploring the level, the player will find many power-up items that will benefit the player in various ways. These power-ups are only effective for the current run.
    - Spawning
        - Power-ups will be spawned in the chest room or puzzle room.
        - The chance of spawning for each power-up is based on their rarity. There are three levels of rarity: common, uncommon and rare. Common items have a 60% chance to spawn. Uncommon items have 30% chance and rare items have 10% chance.
        - Power-ups will have a stronger effect if they are of higher rarity.
        - Power-ups are not stackable. Each time a power-up needs to be spawned, the game will check if it has been obtained. Additionally, no identical power-up will be spawned at the same level.
    - Effect
        - Most of the power-ups are passive and will be effective throughout the gameplay once collected.
        - For each game frame, we will have a system loop through the power-ups to update their effect. Some power-ups will lose their effect after a certain period (e.g. flare in the example) and this system will make sure it happens
- We will adjust the number of power-ups that we will add to the game based on our progress. That is, we may add less power-ups if time does not permit.

4.2 Permanent upgrades (high priority)

Killing each zombie will give the player a certain amount of experience points. These rewards will persist across game sessions, allowing the player to upgrade in between runs. In the main menu the experience points can be used to upgrade the players. All the upgrades will take effect when the player starts a new game run. Below provided are some potential upgrades for the player.

- **A player can earn experience points by**:
    - Defeating enemies
    - Finding rooms
    - Interacting with key game element (e.g. solving puzzles)
- **General upgrades** across all professions (purchased with experience points):
    - **Basic stats** (max health, move speed, inventory expansion, critical hit chance, critical hit damage, loot magnet radius)
    - **Weapon upgrades** (max ammo per clip, damage)
    - **Item upgrades** (increase the health regenerated with health items, similarly with battery items)
    - Quality of life upgrades:
        - Map insight: Reveal portion of the map for a limited time to open up clear vision for the player at the start of each level.
        - Audio cues for enemy: Have visual indication for direction of enemy when they are near.
- **Special upgrades based on profession:**
    - **Soldier Upgrades**
        - **Increased Firepower**: Improve gun damage by 10% per upgrade (max 3 upgrades).
        - **Ammo Efficiency**: Reduce bullet consumption by 15% when firing (max 3 upgrades). (there is 15% chance that a shoot will not consume ammo)
    - **Doctor Upgrades**
        - **Advanced Healing**: Enhance health pack efficiency, restoring an additional 20% health (max 3 upgrades).
        - **Crafting Skills**: Unlock the ability to create health packs from found resources (max 1 upgrade).
    - **Hacker Upgrades**
        - **Enhanced Mini-Computer**: Allow one additional use of the mini-computer per level (max 1 upgrade).
        - **Stealth Protocols**: Temporarily avoid enemy detection for a limited time (max 2 upgrades).
        - **Battery Optimization**: Increase headlight battery life by 25% (max 2 upgrades).

4.3 Consumables (high priority)

There will be four types of consumables in the game. They may be collected throughout the gameplay in various places

- Grenades
  - Can be found in special rooms
  - Will cause damage to enemies in a area upon use
  - The area is a circle in front of the player
- Health consumable/Health pack
  - Can be found in special rooms and enemy dropped rewards
  - Will heal a certain amount of health upon use
- Battery
  - Can be found in special rooms
  - Will charge the headlight battery upon use
- Ammo clip
  - Can be found in enemy dropped rewards
  - Will increase player's ammo upon use
  - Will be used automatically upon picking up

If we cannot add all the consumables to our game, we may remove grenades as it is of less priority

4.4 Map generation and update cycle (high priority)

"Randomly" generated maps will be implemented to ensure that each round of the game offers a unique experience for the player. Every time when a player enters a map, there will be a set of preset map layouts that are randomly selected. This approach will add randomness to the game walkthrough, enhancing the overall gameplay experience.

When a map preset is selected, the game will try to spawn zombies and rewards on the map, according to the following rules.

4.4.2 Rooms

Each map preset will have a fixed amount of rooms, but the content of the room will be randomly generated when the map initializes. Following are some types of rooms that will be generated.

- **Safe room**: the player enters the room, their health is instantly restored to full HP. Enemies will not enter this room.
- **Power room**: the player enters the room, their headlight gets fully charged immediately.
- **Arsenal**: the player enters the room, their weapon's ammunition automatically reloads to its maximum capacity. The player may also find grenades in the room.
- **Lab room**: dead bodies of first immigrants and some companions, experiments notes, getting clues of the facility and the Martian's plan.
- **Boss room**: The space is more open for fighting, win the boss then the entrance to the next level opens. This room will not be generated unless in level 3

- **Puzzle room**: Once the player enters the room, the room will be locked down until the player solves the puzzle. Solving the puzzle can get a reward (power-ups, consumable or hint of the next puzzle room). A Hacker can use their mini-computer to skip the puzzle but won't get rewards.
- **Chest room**: The player will find a chest in this room. The chest contains a power-up or consumable.

If time does not permit us to implement all the rooms, we may focus on completing at least the chest room and 1-2 other rooms.

4.4.3 Enemy Spawning

At initialization and every certain period (we call this spawn checking interval), the game will attempt to spawn enemies

- We will set many spawn points on our map presets. Enemies will only spawn at the designated spawn points.
- Every time the map attempts to spawn an enemy, spawn points closer to the player will be prioritized and only a fixed number of enemies will spawn. However, enemies will not spawn within a certain distance of the player. This distance is slightly larger than the player's FOV. If we cannot implement this complicated spawning rule, we should just randomly spawn zombies on the map.
- The spawn rate of the enemy will increase as the player's battery runs out. This is achieved by making the spawn checking interval proportional to the player's headlight battery level. The spawn rate of the enemies will be maximized when the player completes.
- Different levels will spawn different enemies. We plan to have three types of zombies as enemies: easy, intermediate and hard.

- Level 1 will have 70% easy zombies and 30% intermediate zombies. Level 2 will have 40% easy, 40% intermediate and 20% hard. Final level should have 50% intermediate 50% hard. The difficulty of the zombie is determined by their health and damage. The harder the zombie, the higher their HP and damage is.

4.5 Profession (low priority)

At the start of the game, the player will need to choose a profession among the following. This will determine their initial weapon and passive ability.

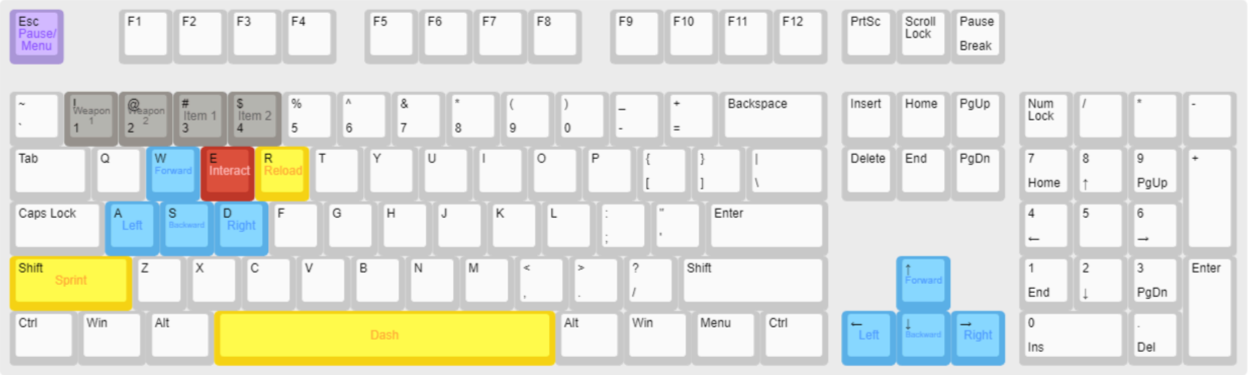
- Soldier: Good at fighting, stronger in physical strength
  - Faster sprint than other characters: more momentum
  - start with 1 grenade (other characters have 0)
  - Uses machete and rifle
- Doctor: Skilled in crafting healing items
  - start with 3 health packs (other characters have 1)
  - At the first death, the doctor can instantly revive on the spot once per level (Other characters return to the main menu).
  - Uses knife and pistol

- Hacker: Has a mini-computer, however, it can be used only once per level.
    - A mini-computer can help him skip a puzzle, but won't get the rewards normally obtained by solving it.
    - a mini-computer can open a locked door
    - Hacker's headlight lasts longer than those of other characters
    - Uses crowbar and SMG

Different melee weapons will have different hitboxes and visuals but their damage is identical. This is the same for ranged weapons: They will have different visuals, but they will have the same base damage. If we cannot complete all the professions in time, we should focus on one of them.

5. Devices

Keymap for keyboard control:



Mouse Control: cursor movement to aim, left click to shoot

6. Tools

- Asset preprocessing
    - Blender
        - We may use blender with shader to generate game sprites with open source 3d models.
    - Photoshop, Aseprite, Krita and other graphics editor
        - We may use graphics edition tool to preprocess the free assets
- Game window & events handling:
    - GLFW
        - GLFW provides a simple API for creating windows, contexts and surfaces, receiving input and events.
- Sound effects:
    - SDL
        - SDL provides low-level access to audio, keyboard, mouse, and graphics hardware via OpenGL.

7. Team management

7.1 Responsibility

The responsibilities are distributed among team members as following

- Murtaza
    - Graphics and rendering

        - FOV
    - Level design
        - Map design
        - Map mechanism
- Ray
    - Game logic
        - 2D collision
        - Physics: knockback and sprint
- Jiayi
    - Asset management
        - Different zombie movement sprites
        - Player movement sprites
        - Icons for power-ups, consumables
    - Sound Assets
        - Zombie attack; using weapons; using power-ups; player movements; BGM
- Yuntian Liu
    - Game logic
        - Enemy AI
        - Pathfinding algorithm
        - Grid system
        - State manager
    - UI design and implementation
        - UI layers
- Eric
    - General logic
        - Professions
        - Power ups
        - Consumables

7.2 Communication

Every week on Tuesday we meet in discord. Team members who do not show up for the meeting will be assigned more work. On the meeting we will primarily focus on the following topics

- The progress of the project
    - a show-and-tell of who did what since the last meeting
- The distribution of the workload
    - We will distribute tasks with the Trello board. If someone is unable to complete the task, we will assign the task to another team member
    - If a member feels their workload is unfair, they can voice their concerns, and the team will then redistribute the tasks as needed.

7.3 Policies

- If a team member cannot complete a task due to reasons other than medical emergency, he/she will need to complete more work next week.
- The internal deadline for the workload each week is Tuesday before the meeting.

8. Development Plan

Milestone 1: Skeletal Game

Week 1

- Basic collision system (Ray)
- Character movement (Eric)
- Barebone maze (Murtaza, Jiayi)
- Basic main menu (Yuntian)

Week 2

- Enemies without AI
  - Movement logic (Yuntian)
  - Sprite (Jiayi)
  - Collision (Ray)
  - Damage calculation (Eric)
  - Enemy types and design (Murtaza)
- Looping background music (Jiayi, Yuntian)

Milestone 2: Minimal Playability

Week 1 HP, battery and ammo. Basic shooting. Basic Physics. Damage system

Week 2 FOV. Interaction with the environment. Sound effect for player actions

Milestone 3: Playability

Week 1 Power ups and consumables. Replayable mechanism and upgrade.

Week 2 Tape. Special rooms. Profession

Milestone 4: Final Game

Week 1 Boss fight. Testing previous implementations

Week 2 Animation for the endings. Review for any space of improvement