Design Overview for <<Zombie Survival Game>>

Name: Jayden Kong
Student ID: 104547242

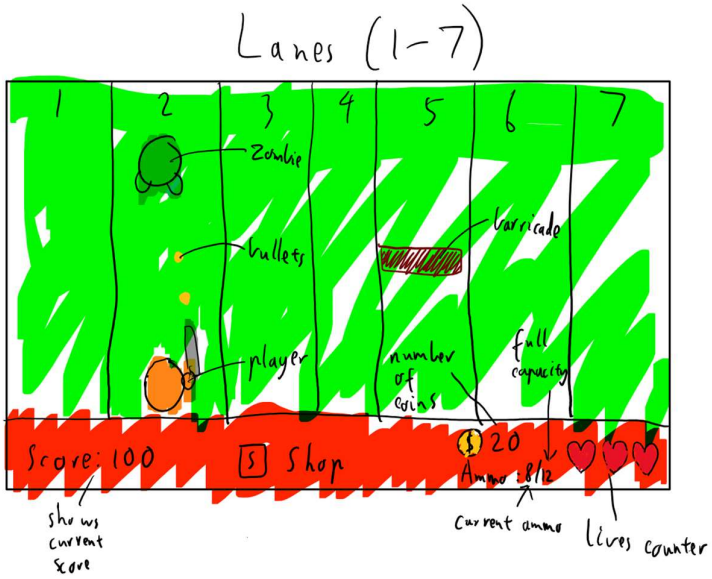
Summary of Program

Describe what you want the program to do… one or two paragraphs.

For my custom program, I would like to make a zombie survival game where the player must defend multiple lanes from zombies approaching a house. For this program, the player will control a player sprite which can shoot the zombies which move down from the top of 7 vertical lanes. For the first 5 minutes, the zombies spawn in predetermined numbers and lanes, allowing for the difficulty to be gradually increased and for gameplay elements to be gradually introduced (difficulty control). After these 5 minutes, the zombies will start spawning in random lanes at a random time interval between 5-10 seconds. There will also be 3 variants of zombies (normal, speedy, tanky) with different speeds, barricade break speed and health. The normal zombie will have average speed, slow barricade break speed and health. The speedy zombie will have fast speed, no barricade break speed (they climb over the barricades) and low health. The tanky zombie will have a slow speed, fast barricade break speed and high health. If any zombie manages to reach the house, one of the player's three lives will be lost. Once all the player's lives have been lost, the game will end and a screen will pop up displaying the player's final score.

The game will also have a shop system. Each zombie killed will drop 1 coin. These coins can be used in a shop where the player can buy things such as upgrades (e.g. increasing their reload speed), extra lives (which can only be bought twice), and as well as the barricade which can be placed directly in the middle of one lane.

Include a sketch of sample output to illustrate your idea.



Required Data Types

Describe each of the records and enumerations you will create using the following table (one per record).

Table 1: <<Zombie_stats>> details

Field Name	Type	Notes
Health	Integer	Influences number of bullets players need to kill the zombie
Speed	Integer	Indicates how fast zombies move down a lane
ASPD	Integer	Indicates how fast a zombie can break through a barrier

Table 2: <<Player_stats>> details

Field Name	Type	Notes
Speed	Integer	Indicates how fast the player can move from lane to lane
Reload_speed	Integer	Indicates how long it takes the player to reload the gun

Table 3: <<Zombie type>> details

Value	Notes

1: Normal 2: Speedy 3: Tanky	Used for accessing the zombie's stats in an array

I will definitely need more records/enumerations and I will figure that out as I go.

Overview of Program Structure

List the main functions/procedures you are going to need to create this program. For each function/procedure provide its name and a brief description of what it will do.

Don't spend too long on this at this stage. Focus on the main things you think you are likely to need and you can build on this as your program develops.

Include a structure chart (once you have your proposal approved by your tutor)

Create_zombies

- Sets up the zombie stats for each different type of zombie

Shoot

- Checks if ammo count is greater than 0
- If ammo count is greater than 0, it shoots a bullet and subtracts 1 away from the current ammo count
- If ammo count is equal to 0, a prompt will flash on screen to indicate to the player to reload

Shop

- Checks if the user buys any upgrades and applies these upgrades to their current stats
- Checks if the user buys any barricades and adds this to their inventory
- If the inventory already has 3 barricades, a prompt will flash on screen, telling the player they have too many barricades

Place_barricade

- Checks if the lane the player currently is in already has a barricade
- If there is none, it place a barricade in the lane
- If there is already one, do nothing

Zombie spawns

- Has exact number and types of zombies to be spawned in each lane for the first 5 minutes
- After 5 minutes have passed, it spawns a random amount of each specific zombie in random lanes at a random time within 7-15 seconds

Structure chart: