PROJECT REPORT

PLANTS VS ZOMBIES

COURSE: OPJECT ORIENTED PROGRAMMING INTERNATIONAL UNIVERSITY – VIETNAM NATIONAL UNIVERSITY HCM School of Computer science and engineering

Declaration

We hereby declare the project report titled “PLANTS VS ZOMBIES” is submitted to Mr.Nghia . This report has been prepared on the basis of our own work where other published and unpublished source materials have been used, these have been acknowledged. This project work is submitted in partial fulfillment for School of Computer Science and Engineering, International University.

Group member:

Name	Student ID	Contribution
Nguyễn Đức Quốc Anh	ITITWE20018	33,33%
Lê Trọng Vinh	ITCSIU23045	33,33%
Nguyễn Phước Được	ITCSIU23005	33,33%

Link to project:

I.INTRODUCE TO THE GAME:

Plants vs Zombies was released on 5/5/2009 by the publisher Popcap. Designer George Fan and his collaborators have created a game about the battle between the undead and… trees. This is a really crazy idea. After more than 10 years, Plants vs Zombies has now become a classic game, it is impossible not to mention when talking about the development of the Tower Defense series.

Plants vs. Zombies is a tower defense video game developed and published by PopCap Games. The game was originally released for Windows and Mac OS X platforms, then ported to consoles, handhelds, and mobile devices. In Plants vs. Zombies, players will take on the role of a host in the midst of a zombie

apocalypse. When a horde of zombies begins to attempt to approach the house along parallel lanes, the player must protect the house by planting trees in the lane to shoot bullets that kill the zombies or harm them. The player collects a currency called "sun" to buy plants. If a zombie reaches the house by any lane, the level is considered failed, and the player will have to restart the level there.

Game interface overview:

Example:


650
100
50

2.1.Items:

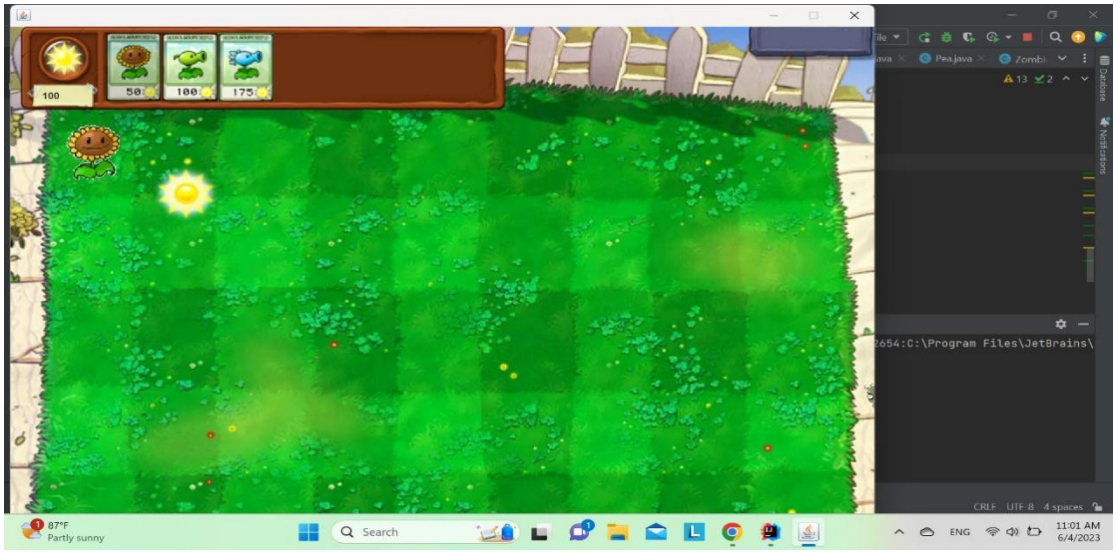
These are the items what we used in the game interface:

- Plant: 

- Zombie:

2.2 How to play:

- At the beginning of each level, players should try to develop themselves a lot of Sunflowers to be able to face the sun and earn money to buy weapons:



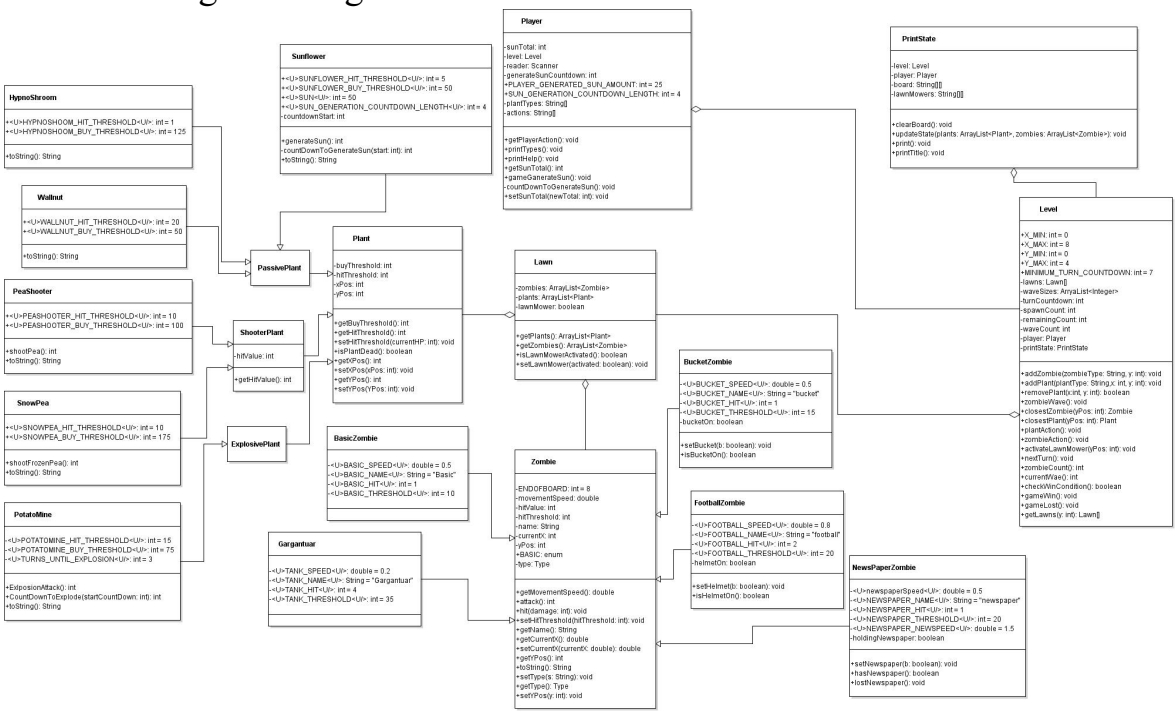
- Every 5s a sun will drop randomly at the top of the game and every 15s a sun will spawn from the sunflower. From there players will use sun to buy peashooter to quickly kill zombies:

- Because later, the difficulty will increase because there is a new stronger and stronger zombie, so we need to continue to accumulate sun to buy a new

type of plant that is freeze peashooter. freeze peashooter can slow down zombies and can buy more time for peashooters to shoot:

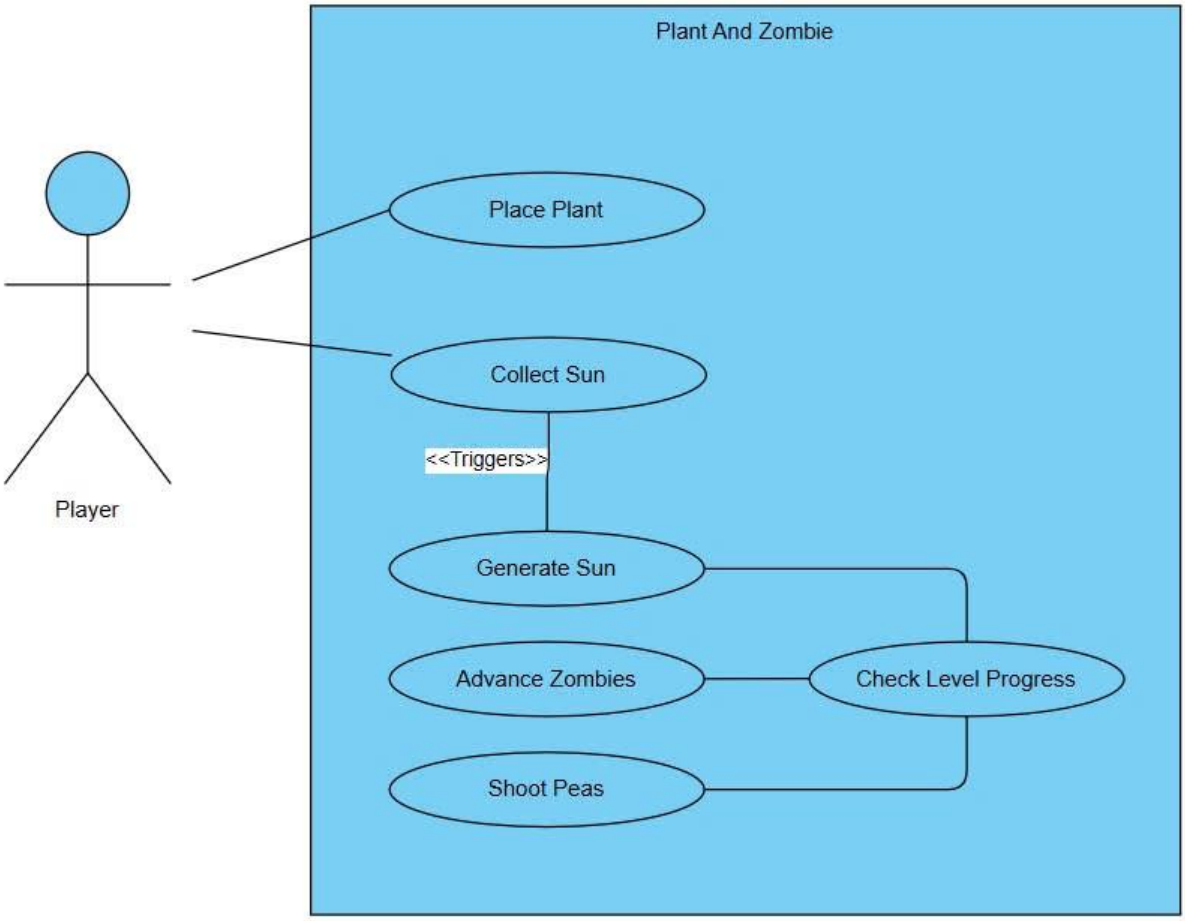
II.PROJECT ANALYSIS:

1. Class diagram design:

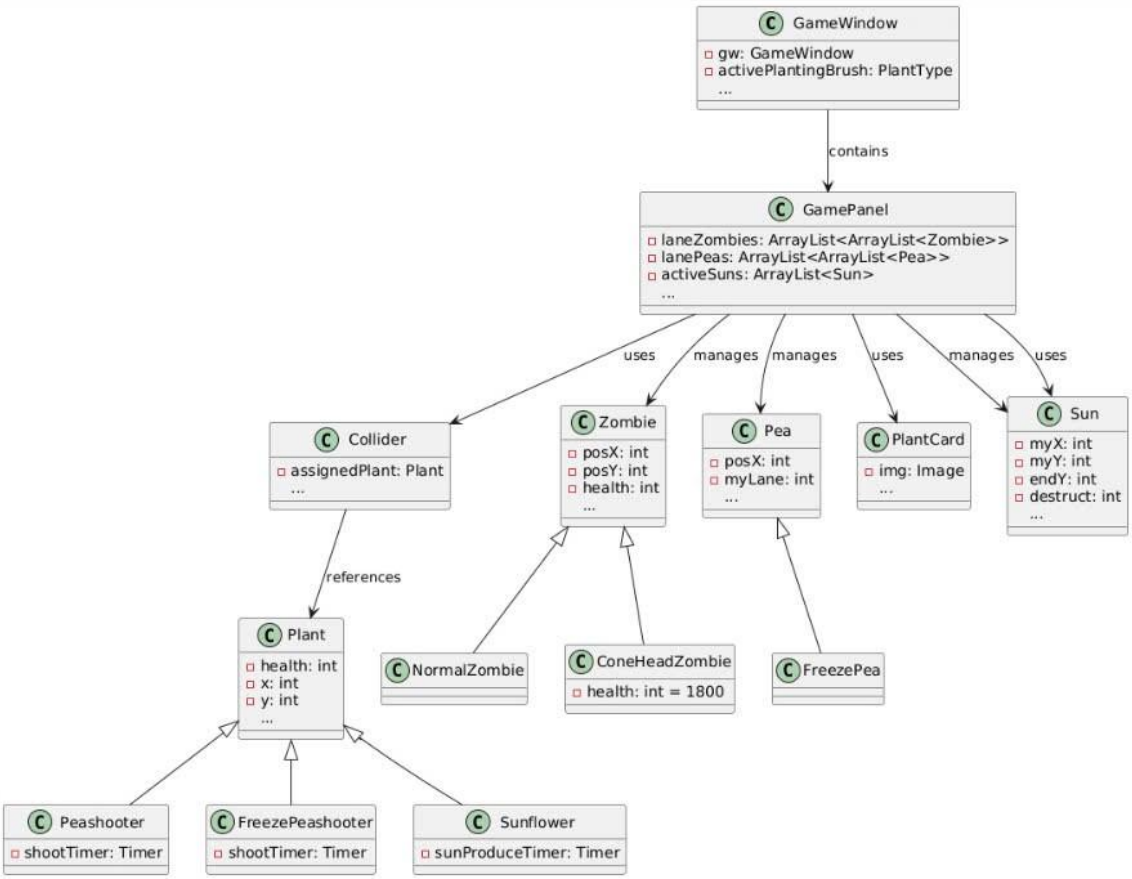


Link:https://www.google.com/url?sa=i&url=https%3A%2F%2Fgithub.com%2Fsirakberhane%2FSYSC-3110-Project-Plants-vs-Zombies-The-Puzzle&psig=AOvVaw3m2oIvIuUdMNYJKZXkpqR4&ust=1733461660076000&source=images&cd=vfe&opi=89978449&ved=0CBQQjRxqFwoTCMjah4fuj4oDFQAAAAAdAAAAABAE

2. Use case diagram



3. UML



4. Class analysis

4.1 GameWindow:

- GameWindow is the main class in the game Plants Vs Zombies. It extends from JFrame to display the game screen for user interaction.

- This class has several attributes and methods to manage the game state and actions.

Key attributes include:

- gw: a variable to represent the GameWindow class.
- PlantType: it represents all trees of enum data type such as: None, Sun Flower, Pea Shooter and Freeze Pea Shooter.

Methods include:

- GameWindow(): this method creates a game screen and it consists of several classes like: PlantCard, GamePanel.
- GameWindow(Boolean b): the method will create a menu screen for the user to select the game mode and it contains a Menu class.
- begin(): method will close the menu screen and start the level when the user clicks on “Adventure” mode.

4.2 GamePanel:

- This class implements MouseMotionListener to help the user use the mouse to interact with the plant tag and plant the tree.

Key attributes include:

- bgImage, peashooterImage, freezePeashooterImage, sunflowerImage, peaImage, freezePeaImage, normalZombieImage and coneHeadZombieImage: all of these properties are of the Image data type and they all represent the images of plants, peas, zombies, and gardens.
- colliders: is an Array to display the garden plot and it has a total of 45 squares. (5 rows and 9 columns).
- laneZombies and lanePeas: These properties use the ArrayList data type to represent zombies and peas that appear in 5 rows.
- activeSuns: is an ArrayList data type and it shows how the sun behaves like: falling to a random position, disappearing and appearing from the sun flowers.
- redrawTimer, advancerTimer, sunProducer, zombieProducer: is the Timer data type to set the time to repeat in the delay time. (Ex: time delay=5000, 5000/1000= 5 seconds)
- sunScoreBoard: is a JLabel data type and it show the sun score in the board.
- mouseX, mouseY: are an int data type, it represents the position of mouse.
- sunScore: is an int data type and it will update the sun score when the score used or collected.
- progress: is an int data type and it will count to 150 and stop to show the challenge are done. (1 zombie = 10)

Methods include:

- getSunScore(): will update the sun score.
- setSunScore(int sunScore): change the sun score when the user used or collected.
- GamePanel(sunScoreboard JLabel): create a sun scoreboard, 45 squares with a loop command and a garden to plant trees. It is also creating time loop for sun maker, zombie maker, redraw timer and advanced timer. Besides, this class accesses the image file to get an image of each of the above properties and creates a list of 5 arrays for zombie and bean sugar.
- advance(): to make the zombies and peas of 5 rows active at the same time.
- paintComponent(g Graphics): method to draw the appearance of plants such as: planting trees and zombies.
- mouseMoved(e: MouseEvent): support the user use the mouse to planting trees.
- setProgress(num: int): to display a notification when a player reaches a score equal to or greater than 150.

4.3 Level Data:

- The class develops a difficult game mode for players.

Key attributes include:

- Lvl: is a String data type and it equal to “1”. It means that this is the easy level.
- Level [] []: is a String data type and it is created by 2 Arrays. It contains two types of zombies such as: normal zombies, normal zombies and cone-headed zombies (the second array to ensure that the spawn rate of the cone-headed zombies is small and makes the game easier.)
- LevelValue [] [] []: is an int data type and it has 3 arrays to make normal zombies appear more than hat-headed zombies.

Methods include:

- LevelData(): create a file that stores the levels the player has played.
- write(String lvl): to update the data of players who have passed the stage and move on to the next stage.