VIETNAM NATIONAL UNIVERSITY HCM

INTERNATIONAL UNIVERSITY

School of Computer Science & Engineering

MEOW VS. ZOMBIES PROJECT

Course: OBJECT - ORIENTED PROGRAMMING

Group member

No	Full name	ID
1	Nguyễn Hoàng Hồng Ân	ITDSIU22151
2	Phạm Nguyễn Quỳnh Anh	ITDSIU22130
3	Đoàn Võ Thảo My	ITDSIU22138
4	Nguyễn Phúc Minh Quân	ITDSIU22163
5	Lê Bảo Trân	ITCSIU21114

https://github.com/nhhongan/meow-vs-zombie

Chapter 1. Abstract

Plants vs. Zombies is a tower defense video game developed and published by PopCap Games. Plants vs. Zombies was designed by George Fan, he initially had the idea that it would be a more defense-oriented game than the previous fish farming simulation game Insaniquarium (2001), from then on it gradually developed. evolves into a tower defense game featuring plants fighting against zombies. The game draws inspiration from titles such as Magic: The Gathering and Warcraft III: Reign of Chaos , along with the film Swiss Family Robinson.

In a tower defense video game, the player's goal is to prevent zombies from reaching the house. 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. Players collect 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 that level.

In this undertaking, our team created the game Meow vs. Zombies to bring players a version of Plants vs Zombies with many changes and improvements. The theme of the game is recreated based on the idea of a tower defense game with plants fighting against zombies. Along with the new idea, the basic rules of Plants vs Zombies remain the same, but the game rules have been refreshed and added some innovative features. New character, all plants change to cats. According to new game rules, the player must protect the pool by raising cats in the lane to throw wool that kill the zombies or harm them, and if a zombie falls into the pool, the level is considered failed and the player will have to restart that level.

The game encourages players to observe, strategize, and train the factory's defenses to overcome various challenges. This is the product of an effort to provide players with an entertaining game with lots of engaging content.

https://github.com/nhhongan/meow-vs-zombie

Chapter 2. Introduction

In today's rapidly advancing Software Technology industry, a higher level of programming skill is becoming increasingly essential. As a result, the traditional Procedural-Oriented Programming approach is no longer sufficient to meet all requirements. This has led to the development of a new method called "Object-Oriented Programming," following Alan Kay's principles, to address these challenges.

This project was specifically designed using the Java language, incorporating Object-Oriented Programming. This approach has effectively resolved several issues that commonly arise when using the traditional Procedural-Oriented Method:

- The code becomes more transparent, easily understandable, and concise.
- The project represents a cohesive logical system, achieved by combining numerous related classes.
- Each class contains multiple methods that perform distinct behaviors unique to that class.

Resources can be efficiently reused, enhancing overall efficiency.

The purpose of this project is to design a basic game by using OOP (Object-Oriented Programming) method. Therefore, our team decided to recreate a game named ‘Meow and Zombies’ that followed the four pillars of this measure. This project will go through the game overview and design, describing how the game is implemented and the programming functions and libraries used in the design.

Beside the requirements of the course, our team also wants to create this project on account of learning and practicing the techniques throughout the semester. On the other hand, we considered it as a good opportunity to have a special mark in progress of the future career as a programmer. Therefore, the final version is the effort and hard-work of all team members without much interference from other people or available source codes from the internet.

https://github.com/nhhongan/meow-vs-zombie

Chapter 3. Game overview

Meow vs. Zombies is an action strategy game where players must prevent enemy zombies from crossing their front lawn and entering the pool. Players strategically place meow throughout the yard in an effort to stop the forward progress of the zombies.

3.1 Game description

3.1.1 Character

- Meow

  The main defense against the advancing legions of zombies is Meow (3.1.1 Character - Zombie). Their diverse set of powers and traits contribute to the player's garden's defense. On the lawn that resembles a grid in the game, these plants are positioned carefully to combat the many kinds of zombies that are encountered.

  In order to facilitate the raising of meow in underground tiles, it is necessary to verify the status of the tiles to determine if they are empty or occupied. If a tile (3.1.2 Environment - Tile) is found to be empty, the specific properties of the meow, such as their ID, damage potential, and fish consumption. Subsequently, the program continues to assess the conditions, specifically checking if the current amount of available fish (3.1.2 Items - Sun) is greater than or equal to the fish consumption of the selected meow.

Not enough fish	

https://github.com/nhhongan/meow-vs-zombie

Enough fish	

Fig. 3.1.1 Character - meow (1)

The Meow (Meow and Ice-meow) are equipped with an alert system, which activates when a zombie appears within the same row as a meow. This alert is triggered through the function "alertMeow". When a meow receives an alert, it initiates an attack on the zombie, while if the meow successfully defeats all the zombies in its row, it returns to its idle state. The transition between alert and idle states is facilitated by the functions "setMeowDangered" and "setMeowIdle" respectively, allowing meow to perform their respective tasks based on the current situation.

Idle state	
Alert state	

Fig. 3.1.1 Character - Meow (2)

https://github.com/nhhongan/meow-vs-zombie

- Zombies

  Zombies in Meow vs. Zombies are the primary adversaries in the game. They are depicted as reanimated corpses with the objective of invading the player's garden and devouring the meow. Understanding the different types of zombies and their behaviors is essential for developing effective strategies to repel their attacks.

  The game is structured into waves (6.5 Wave/Level), with each wave presenting a new set of challenges. A wave consists of a fixed number of zombies that players must defeat to progress to the next level. The number and types of zombies in each wave vary, becoming progressively more difficult as the game advances.

  Zombies are randomly spawned across the lawn at the beginning of each wave. The specific locations where zombies appear are determined by the game's mechanics, ensuring a diverse and dynamic gameplay experience. This randomness adds an element of unpredictability, requiring players to stay alert and adapt their defensive strategies accordingly.

Fig. 3.1.1 Character - zombie (1)

https://github.com/nhhongan/meow-vs-zombie

3.1.2 Items

- Fish

  The fish plays a crucial role as a resource that players must manage effectively to grow and sustain their meow. Fish is represented by small fish icons that periodically fall from the pool or are generated by the bucket.

  Fish acts as the primary resource used to purchase and deploy meow. Each meow has a designated fish cost associated with it, indicating the amount of fish required to plant it. Players must accumulate enough fish to strategically select and place meow in their garden to defend against the encroaching zombie horde.

  Throughout the game, sun icons randomly appear on the game's lawn or are produced by specific fish-producing buckets. This meow includes Bucket. When players click on a fish icon, it is collected and added to their available fish count.

- Bullets

  The meow-shooter throws red wools as its primary bullet. These wool act as ammunition to damage and eliminate zombies. Each meow-shooter continuously produces and launches wools in a straight line towards the oncoming zombie wave.

  Bullets throwed by shootable meow travel at a moderate speed, allowing them to reach their intended targets within a reasonable timeframe. The range of the meow's bullets extends across a fixed distance in front of the shooter. This range determines the maximum distance the peas can travel before disappearing.

Fig. 3.1.2 Item - bullet

- Shark

  When zombies reach the range, the shark will appear at the row zombies touch then we will count how many times the shark appears. If it appears more than one time, the next time the zombie reaches the range , the shark will not appear.

  The range we will calculate based on the game screen width and height.

3.1.3 Environment

- Title

  The tile refers to the individual units or squares on the game's playing field or lawn where players can plant their defensive meow. The playing field in the game is divided into a grid system, with each grid representing a tile. The grid is typically organized in rows and columns, creating a matrix of tiles where players can strategically position their meow.

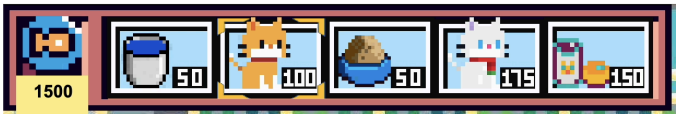
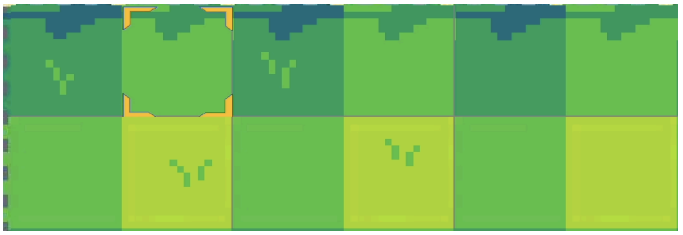
  Players can place their meow on the available tiles within the grid. Each tile can accommodate a single meow, and the placement of meow is subject to the player's strategic decisions. Proper placement is essential for maximizing meow effectiveness, creating defensive formations, and countering the approaching zombie horde.

- Select frame

  The game interface includes a selection frame for tiles and a meow bar. Both can be used by the mouse controller or keyboard controller (3.2 User manual).

  The selection frame of tiles refers to the highlighted area or border that appears around a tile when it is selected or targeted. This frame helps players identify and interact with specific tiles on the grid.

https://github.com/nhhongan/meow-vs-zombie

Bar's selection frame	
Tile's selection frame	

- Meow bar

  The meow bar is a user interface element that displays the available meow options for players to choose from during gameplay.

  The primary function of the meow bar is to allow players to select and choose meow for placement on the grid. It typically appears at the top or side of the screen and consists of a row or column of meow icons representing different meow types.

Fig. 3.1.3 Environment - meow bar

3.2 User manual

3.2.1 Mouse controller

Playing Meow vs. Zombies with a mouse is relatively straightforward. The general overview of how player can control the game using a mouse is listed below:

● Menu Navigation: Players can navigate through the game's menus using the mouse. Move the cursor over the desired option and click to select it.

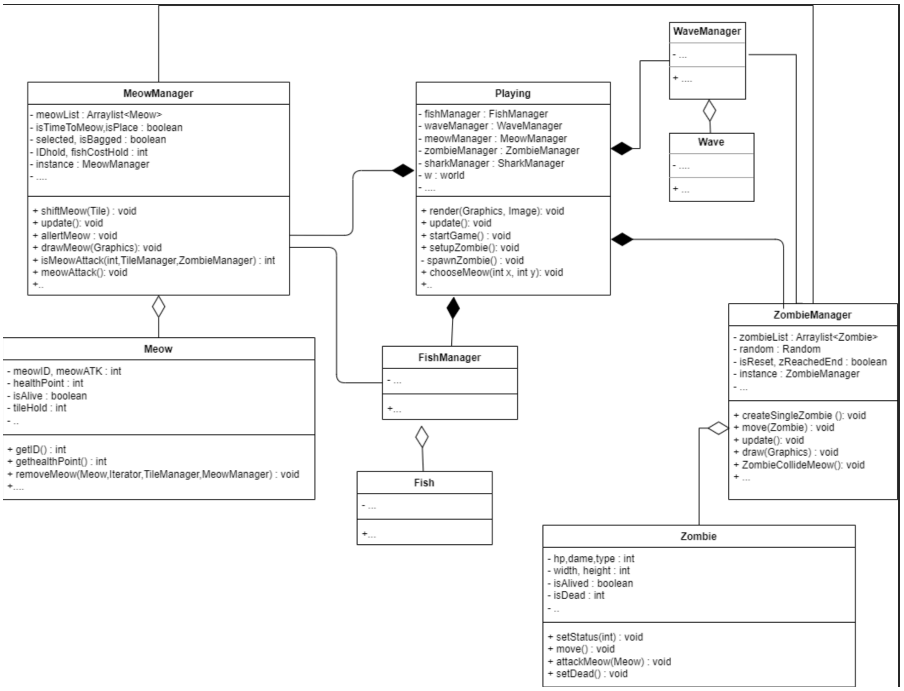
https://github.com/nhhongan/meow-vs-zombie

● Meow Selection: To select a meow, move the cursor over the meow icon located on the right side of the screen. Click on the desired meow to pick it up.

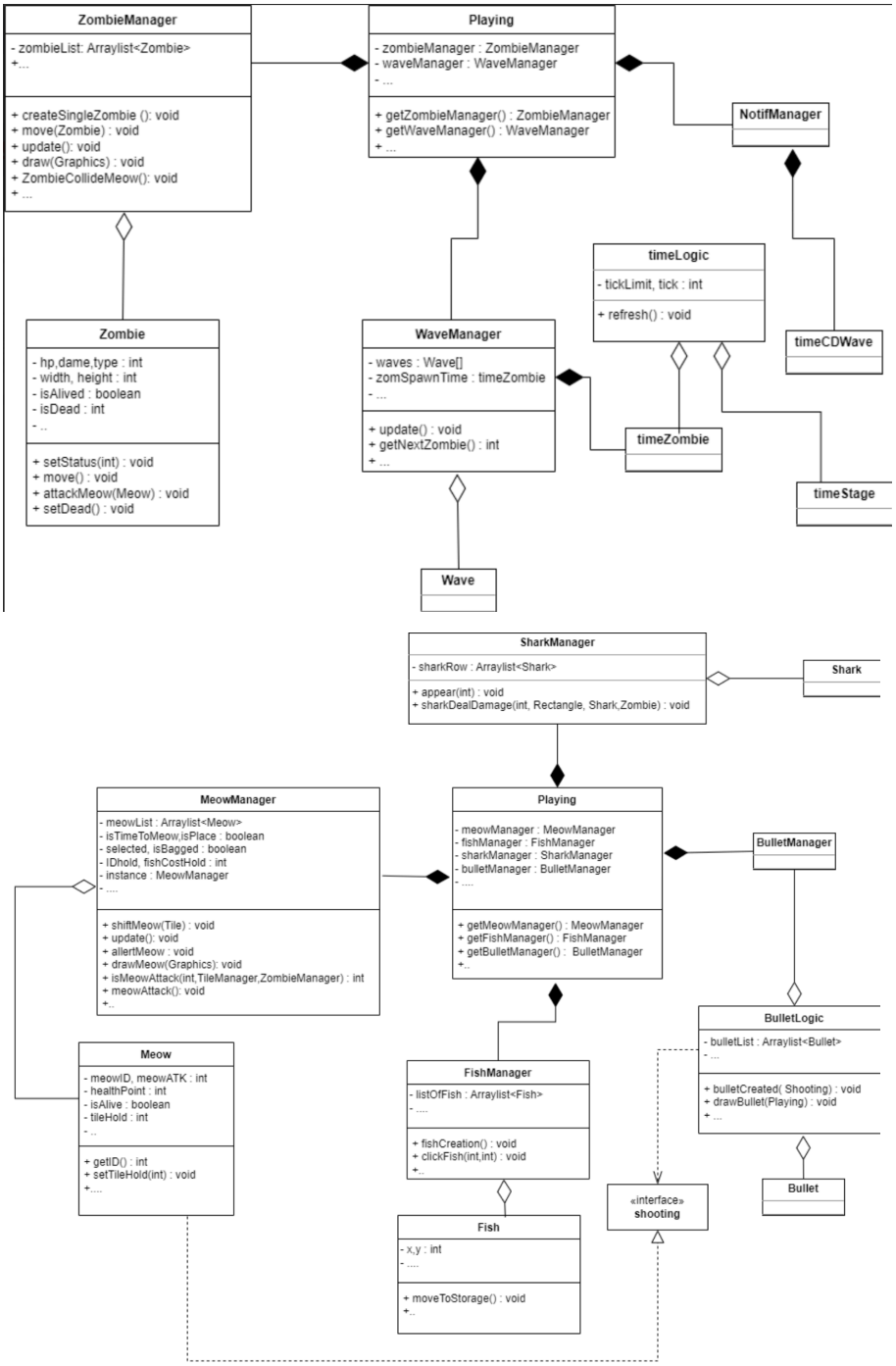
● Meow Placement: Once a player has selected a meow, move the cursor to the desired location on the grid. Left-click to place the selected meow in that spot.

● Fish Collection: Fish, which is the in-game currency, falls from the pool. Move the cursor over the fish to collect it. The collected fish can be used to purchase and place meow.

Chapter 4. UML



https://github.com/nhhongan/meow-vs-zombie



Chapter 5. Methodology

Before participating in the recent project, our team conducted research on the Object-Oriented Programming (OOP) methodology and its basic properties. However, we acknowledged that a thorough understanding of OOP alone would not be sufficient.

https://github.com/nhhongan/meow-vs-zombie

Our team also needed to address the challenge of collaborative coding. As a solution, we turned to Github, one of the most valuable tools for software developer teams. We primarily coded the game using Intellij IDEA and utilized Github as a code hosting platform for version control and collaboration. This enabled us to organize and manage the project more efficiently.

One month prior to starting the project, we conducted research on related games and followed how-to tutorials from various sources. Simultaneously, we engaged in practical exercises focused on OOP concepts and applied our knowledge using the Github platform.

Chapter 6. Game design

6.1 Objects

Objects play the most important role, bringing life and soul to this game. There are various types of objects in this game, and each type of object plays a specific role. Various objects in this game are used to create UI, thus making it interact-able by players. Therefore, players can use this opportunity to make many special formations, strategies or even performing experiments between objects

In addition, many objects have the ability to interact with other objects as well, making the game more consistent in terms of game logic. Meow, zombies, bullets, bar, and buttons are some typical objects of the game. Each of them will have different interactions with each other or with players in the game:

- Meow will use bullet to attack zombies
- Zombies will continuously move and attack meow if needed
- Bar and Buttons help players to interact with the game

6.2 Game scenes

Scenes are the collection of atmospheres in the game

Similar to other games, scenes in this game are also very important, as they can navigate players to make correct decisions

https://github.com/nhhongan/meow-vs-zombie

based on each scene. Not only that, they can also make the environment of the game become more colorful and lively.

Scenes also provide User Interface, allowing players to interact with them. Many UI are useful since they help players have a better experience of the game. There are total of 5 scenes in this game:

- Menu: initial scene when players turn on the game. It contains 2 buttons which allow players to play or quit the game.
- Playing: The main scenes of the game where players will interact with every main object and contents.
- Game paused: the scene can be found when players press the “pause” button. Not only can it temporarily stop the game, it also provides various functions such as Exit, restart, resume.
- Winning, Losing: the scenes which tell players the game has finished. When the conditions are met, either a winning or losing scene will appear, indicating that players have won (or lost) the game.

Fig 6.2.1: Game paused

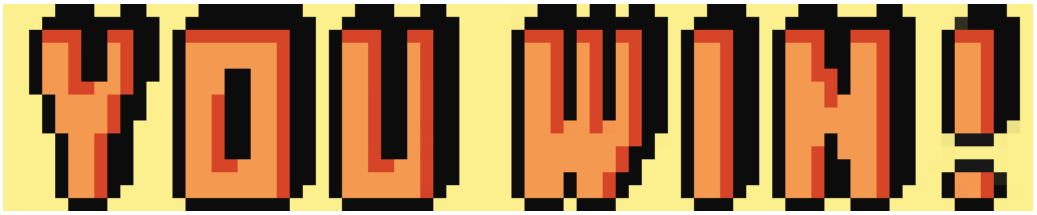


Fig 6.2.2: Win game

https://github.com/nhhongan/meow-vs-zombie

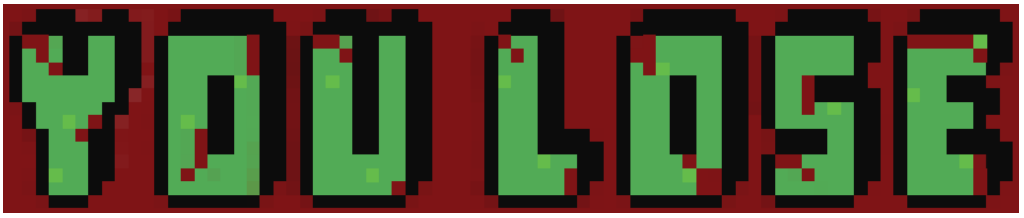


Fig 6.2.3: Lose game

6.3 Animation

At first, when making the game, the objects themselves did not have any animation, which makes the game look boring and unattractive. Therefore, in order to fix it, animation has come to live.

Using the definition of animation, the animations in this game are created in one of the most traditional yet effective ways: by fast-forwarding many picture frames of an animation, the objects are given a view that looks extremely like an animation.

Animations in this game originally come from many sources, but most of it comes from wiki-fandom or from the original game. The 2 types of objects use animation are meow and zombies:

- Meow: attack and idle animations
- Zombie: attack and move animation

6.4 Audio

Even if audio is not a must in this game, this feature still has a very significant role in order to bring to players a better experience. Without audio or sound, the game will become boring and players no longer have the interest to play the game.

Many objects of this game have their own audio, thus each audio will have its own unique properties. Each audio also brings to players different emotions and feelings, such as: calm, happy or intense feeling.

https://github.com/nhhongan/meow-vs-zombie

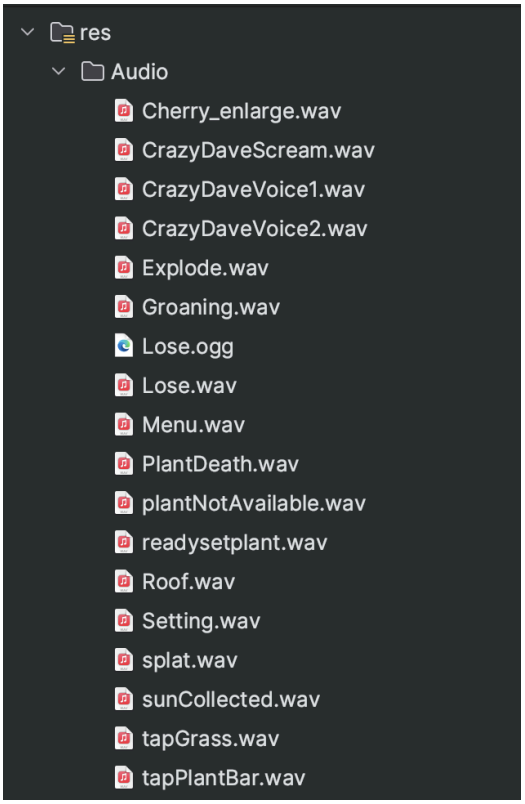


Fig. 6.4 Several audios used in this game

Chapter 7. Conclusion

The game is currently under development. The team's comprehension of the four OOP features, the SOLID concept, and the Design pattern principle has improved after finishing a game that had some new features not included in the initial edition. This has allowed the team to become more proficient in both the programming and game development processes. Therefore, the fundamental principles of OOP were strictly followed in the development of Meow vs. Zombies. The game code incorporates the SOLID principle, a design pattern that was learnt in class, and all four of the main OOP elements (encapsulation, inheritance, abstraction, and polymorphism).

Chapter 8. Futureworks

Unfortunately, the team was hoping to develop even more levels with different levels, game runs more smoothly, some other meow for players. However, due to a lack of time and the need to spend more time resolving some bugs, this group has not yet concluded the initial wish for development. Accordingly, in the future, some new meow and natural conditions such as day or night will be included to add more

https://github.com/nhhongan/meow-vs-zombie

resources to the player and evaluate the player's management skills. Therefore, any new commits are appreciated.

Chapter 9. Acknowledgement

We want to express our sincerest thanks to our lecturers and the people who have helped us to achieve this project's goals:

Dr. Tran Thanh Tung

MSc. Nguyen Quang Phu

Reference:

About Code:

[How to Make a 2D Game in Java #1 - The Mechanism of 2D Games]

How to Make a 2D Game in Java #1 - The Mechanism of 2D Games

[GUI]

Java GUI 🖼️

[Mouse listener]

Java MouseListener 🖱️

[SOLID principles]

Learn SOLID Principles with CLEAN CODE Examples

https://github.com/nhhongan/meow-vs-zombie

11. Our Github project:

https://github.com/nhhongan/meow-vs-zombie