**Object Oriented Programming Report Final Project: SNAKE using Java**

A picture containing graphics, screenshot, logo, graphic design

Description automatically generated

**Lecturer: Jude Joseph Lamung Martinez, MCS**

**Prepared By:**

Farrel Fari Rahman - 2602158191

**Binus University 2023**

**Table of Contents**

1. **Brief Description**
2. **UML Diagram**
3. **How it Works**
4. **Screenshots**
5. **Sources and References**

**Brief Description**

In this project I am making the Snake game using Java programming language. My resources for this project are Java swing and ActionListener. For Java swing is a GUI toolkit for Java. It is used to produce a graphical user interface for java programs. For ActionListener is an interface in Java, it is used so whenever an action is performed in the program it will give a notification to the program itself.

**UML Diagram**

A screenshot of a computer

Description automatically generated with medium confidence

**How it Works**

in this program the code are divided into 4 classes and there are Main, GamePanel, GameFrame and MainMenu.

In the Main Class:

A screenshot of a computer program

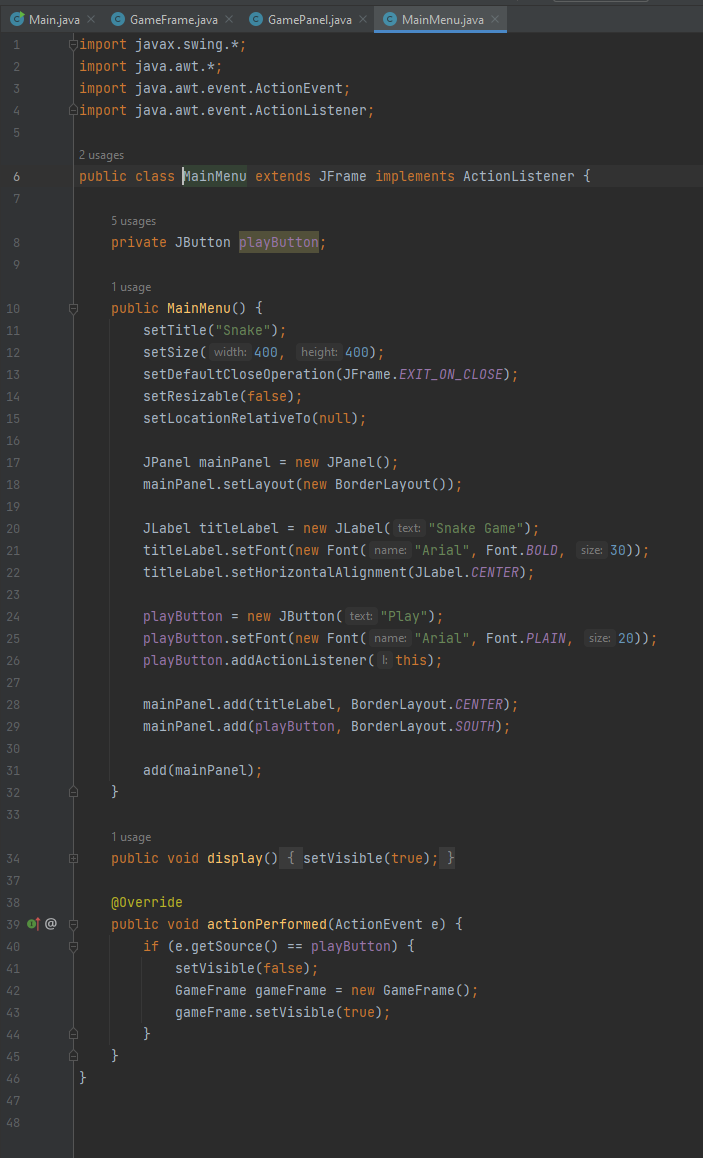
Description automatically generated with medium confidence

In this class it contains the main method and the starting point of the application itself. If the class is executed then the main menu frame will show and display as the starting point of the application.

In the GameFrame Class:  
  
A screen shot of a computer program

Description automatically generated with medium confidence

In this class, it extends JFrame and it represents the main game window. Inside this class are the properties of the frame.

In the MainMenu Class:  
  


In this class it extends JFrame and it represents the main menu window. The display method in the class will be called to show the main menu. The actionPerformed handle the play button, after it was clicked it will hide the main menu while creating and showing the game frame.

In the GamePanel Class:  
  
Part 1.

A screenshot of a computer program

Description automatically generated with medium confidence

In this part the class extend to JPane that represent the main game panel. In this part of the class we can see that there are various of variables that set the rules and setting of the snake game.

Part 2.



In this part there are the start, paintComponent and the draw method. For the start method is to initialized the necessary variables and set up the timer when it is executed. Second for the paintComponent method, this method is used to render the game objects and graphics. Last method is the draw method, this method is used to draw the game objects and the game’s interface such as the snake, food, score and high score.

Part 3.

A screen shot of a computer program

Description automatically generated with low confidence

In this part of the class there are newFood, move, checkFood and checkCollision method. First newFood method, in this method it will generate random coordinate for the food object. Second is the moce method, in this method it will update the snake position based on the current direction. Third the checkFood method, in this method It will check whether the snake has collided with the food or not, if yes then it will record and add the score by the number of the food. Finally the check collision method, in this method it checks for collisions with the snake's own body or the game boundaries and ends the game if a collision is detected.

Part 4.

A screenshot of a computer program

Description automatically generated with medium confidence

In this part of the code there are gameOver, resetGame and saveScore methods. For the gameOver method, the method displays the game over screen with the final score and a retry button. For the resetGame method, the method resets the game state and starts a new game. Finally the saveScore method is the method saves the current high score to a file.

Part 5.

A screen shot of a computer program

Description automatically generated with low confidence

In this final part there are loadScore, actionPerformed method and MyKeyAdapter, RetryButtonListener Class. For the methods, the load score method is where it load the high score from a file. For the actionPerformed method is where the method is implemented from the ActionListener interface and handles the game updates triggered by the timer. For the classes, the MyKeyAdapter class is a nested class that extends KeyAdapter and listens for key events to change the direction of the snake and for the RetryButtonListener class is a nested class that implements ActionListener and handles the retry button click event.

**Screenshots of the Program**

Main Menu Window:  
  
A screenshot of a computer

Description automatically generated

Snake Game Window:  
  
A screenshot of a game

Description automatically generated with medium confidence

Game Over with retry button window:  
  
A screenshot of a computer screen

Description automatically generated with medium confidence

The highest score recorder will be shown in the next game after pressing the retry button:  
  
A screenshot of a game

Description automatically generated with medium confidence

# Resources

* https://www.youtube.com/watch?v=bI6e6qjJ8JQ&t=901s

# GitHub Link:

* https://github.com/WHEL154/SNEK