

FRUIT NINJA

Description

Fruit Ninja is a simple game of slicing fruit. The player uses the mouse to slice the pieces of fruit that jump into the screen by dragging the mouse cursor across the thrown fruit. As the game progresses, more and more fruit come onto the screen, and players can slice multiple fruits at once to score combos. The more fruits the player slices the more points the person gets as slicing a normal fruit earns you a point while slicing a special fruit earns you 5 points to add to the total points earned per game.

User Guide

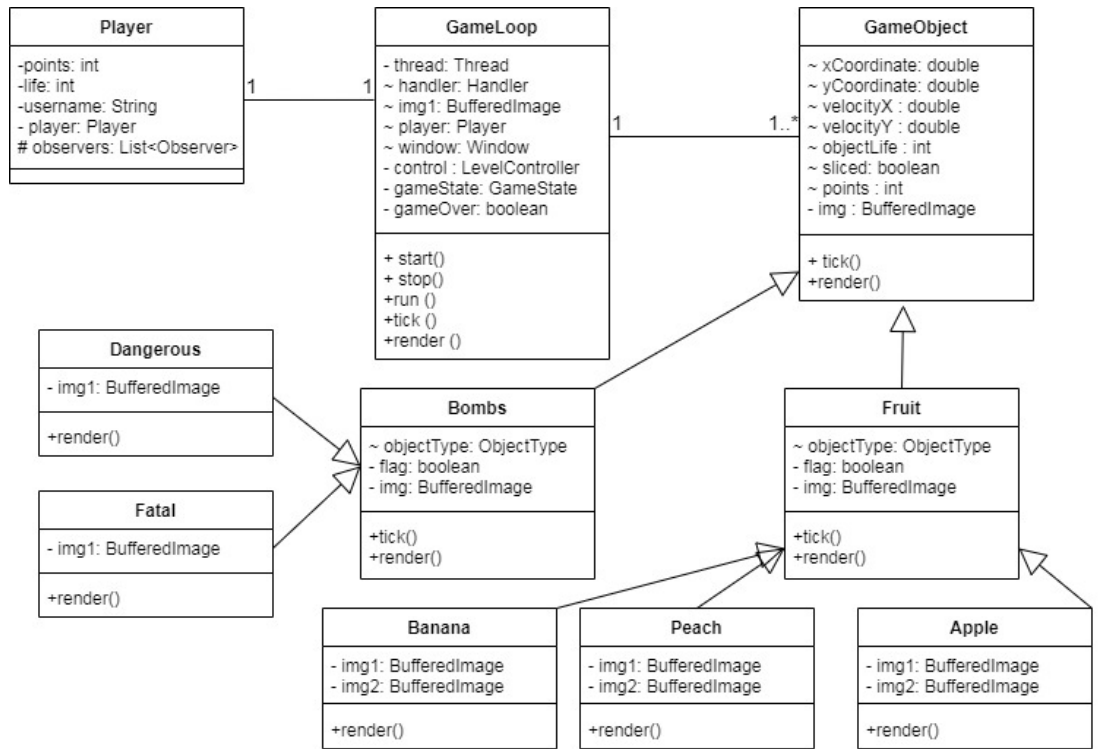
One is given the choice between 2 modes, classic and arcade, upon starting the game. In order to choose the desired mode, the player clicks on the logo of that mode which is found in the main menu

The classic mode grants the player 3 lives at the beginning of the game. If the player slices a bomb or a fruit goes out of bounds before being sliced a life is lost. Once the lives of the player reach 0 the game comes to a halt and the player has then lost the game. Fatal bombs appear from time to time, if these bombs were sliced then the lives of the player instantly reach 0 and the game ends.

The arcade mode allows the player to play for 60 whole seconds without losing lives neither by a fruit going out of bounds nor by slicing a bomb as bombs don't appear in the arcade mode.

To slice a fruit move the mouse cursor over the thrown fruit while also right clicking on the mouse to activate the blade built in the game which transforms the one piece of fruit thrown into 2 slices of that given fruit to give the effect of an actual sliced fruit.

Class Diagram:



GUI

Main Menu:



Arcade Mode(No lives):



Special Fruit:



Dangerous Bomb:



Fatal Bomb:



Design Patterns

Singleton is used to allow one instance of the Player to be created at a time.

Player's points is also set as Observable so that once setPoints method is called to change player's points, it calls notifyAllObservers method which loops through all observers implementing Observer updating their Points.

Observer is specifically implemented by Window to keep a live count of Player's points visible on the game screen, and LevelController which uses Strategy design pattern to generate objects in different difficulties according to points count.

