Project Report

Flappy Bird: Horror Edition

1. Game Overview
2. **Genre:**

Flappy Bird: Horror Edition is a 2D Platformer game

1. **Concept:**

The player as the flappy bird, flaps it’s wings to avoid zombies and demonic dragons coming towards their way, while also collecting coins as a currency to purchase in the shop.

1. **Visual:**

The art style of an 8-bit 2D Platformer game, most often seen in 1980 platformer games.

1. Inspiration
2. **Flappy Bird:**

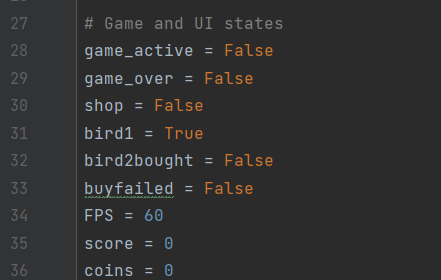
Flappy Bird is a [mobile game](https://en.wikipedia.org/wiki/Mobile_game) developed by [Vietnamese](https://en.wikipedia.org/wiki/Vietnam) [video game artist](https://en.wikipedia.org/wiki/Video_game_artist) Dong Nguyen. The game is a [side-scroller](https://en.wikipedia.org/wiki/Side-scrolling_video_game) where the player controls a bird, attempting to fly between columns of green pipes without hitting them. The game was released in May 2013 but received a sudden rise in popularity in early 2014 Flappy Bird received poor reviews from some critics, who criticized its high level of difficulty and alleged plagiarism in graphics and game mechanics, while other reviewers found it addictive

1. **Sonic.exe:**

Sonic.exe is part of a genre of online horror stories known as [creepypasta](https://www.dictionary.com/meaning/creepypasta).

In August 2011, user JC-the-Hyena submitted a story on the Creepypasta Wiki about a strange CD-ROM the narrator received from a friend. The CD had SONIC.EXE written on it (.exe is file extension designating an executable file). It centered around the video game franchise and character Sonic the Hedgehog. A prominent feature of the story was the look of its murderous incarnation of Sonic, who had been given black, bleeding eyeballs with glowing red pupils. The creator later explained that the creepypasta had been inspired by an edited screencap of the character. Sonic.exe grew in popularity, and a game based on Sonic.exe was created in August 2012 by MY5TCrimson, along with other tributes and parodies. The game and story both gathered tributes and parodies. It’s not clear how much of the creepypasta’s appeal was due to its scariness or an ironic appreciation of its amateurish writing and clichéd style, which later got it removed from the Creepypasta Wiki.

1. Gameplay Mechanics
2. **Game states:**



A list of states and variables

Game\_active -> To display game UI

Game\_over -> To display game over UI

Shop -> To display shop UI

Bird1 -> If bird skin 1 is equipped, if not then bird skin 2 is equipped

Bird2bought -> If bird skin 2 is bought, will display different UI, explained in IV. Interface – ii. Shop menu

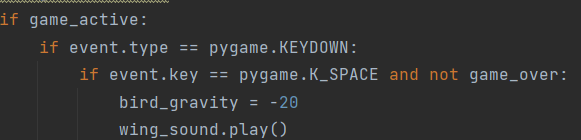
Buyfailed -> If insufficient balance to buy bird skin 2, will display “Insufficient Balance!” explained in IV. Interface – iii. Shop

FPS -> Frames Per Second, used for the last line clock.tick(FPS)

Score -> In game collected coins for each round

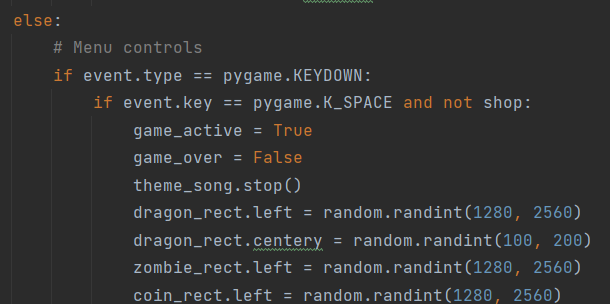
Coins -> Overall collected coins, displayed in the main menu and shop

1. **Controls**



Located in while True loop – for event in pygame.event.get() – if game\_active

Making it able for the bird to “flap” in to a lower bird\_rect.centery value if the player presses space. Only available while the game\_active = True and game\_over = False

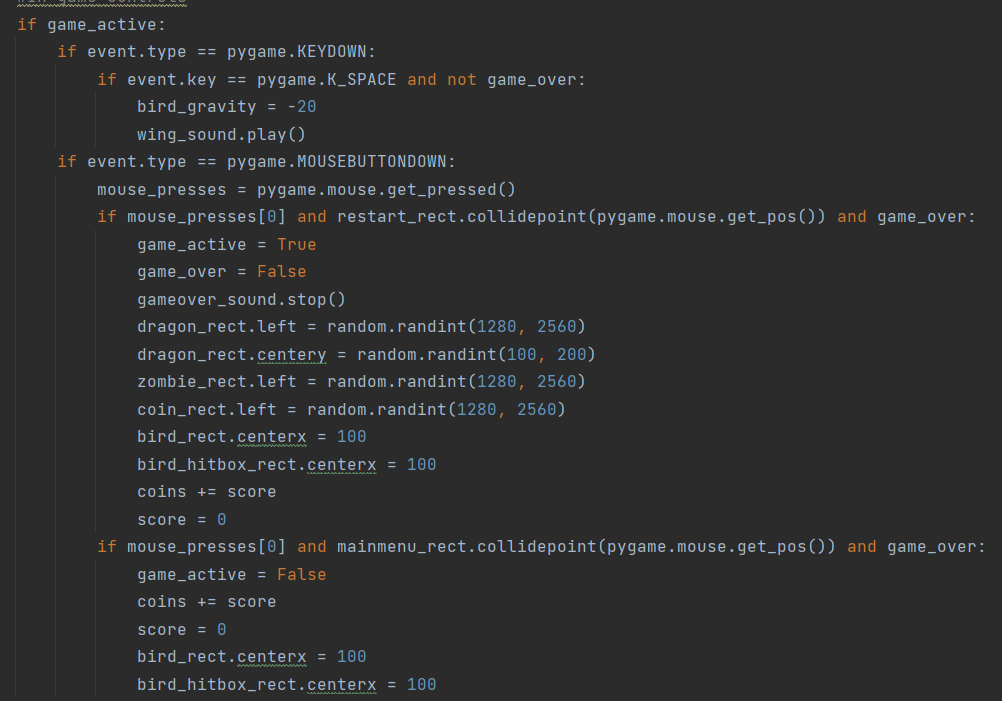


Located in while True loop – for event in pygame.event.get() – if game\_active’s else statement

Making it able for the player to start the game from the main menu, by pressing space.

Only available while the game\_active = False and shop = False

1. **Pressing UI surfaces**



Located in while True loop – for event in pygame.event.get()’s else statement – if game\_active

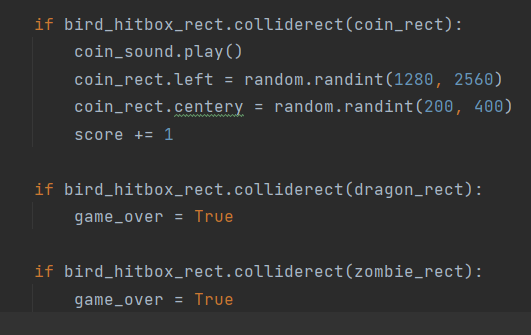
Making it able for the player to press restart, and goes straight back to the game again (game\_active = True, game\_over = False), having their score added to the initial coins variable, and having their score reset.

Resets the bird’s death animation (explained in IV. Interface – IV. Game over UI), resets the demon dragon and zombie’s randomized spawn point.

Also making it able for the player to press main menu, where it goes straight to the main menu UI (game\_active = False), having their score added to their initial coins variable, and having their score reset.

Only available if the game is still running (game\_active = True)

1. **Collisions**

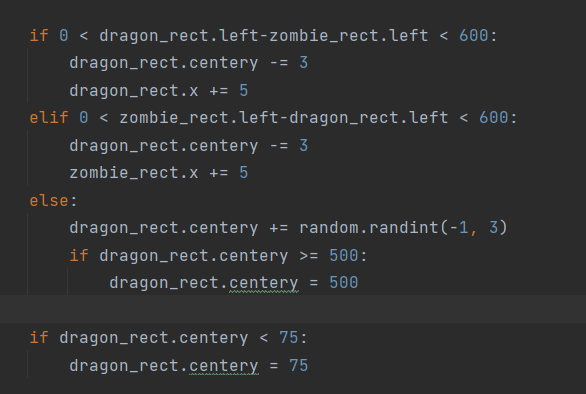


If bird hitbox’s rectangle collides with coin’s rectangle, coin’s rectangle will reposition in to a new position, giving the illusion of getting “eaten” and spawns another one. 1 will be added to the initial score value.

If bird hitbox’s rectangle collides with demon dragon’s rectangle, it will display game over UI.

If bird hitbox’s rectangle collides with zombie’s rectangle, it will display game over UI.

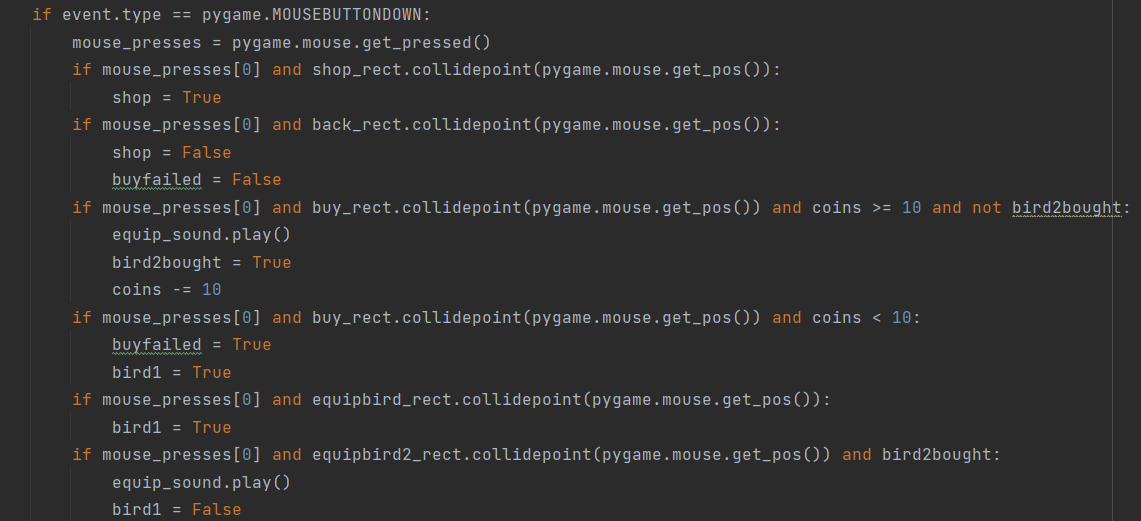
1. **Enemy repositioning and move set algorithm**



An if else statement that repositions or changes the value of zombie\_rect.x (zombie’s X position), dragon\_rect.centerx (dragon’s X position), and dragon\_rect.centery (dragon’s Y position).

Does not permit the dragon to fly above the sky limit.

Adding a feel of randomized enemy move set and repositioning.



Located in while True loop – for event in pygame.event.get()’s else statement – if game\_active’s else statement

A range of if else statements that makes the player able to visit the shop UI by pressing the shop button, go back to main menu UI by pressing the back button, and an algorithm to decide the game states.

If coins is sufficient, bird skin 2 hasn’t been bought yet, and player presses buy: bird2bought = True, 10 coins deducted from balance

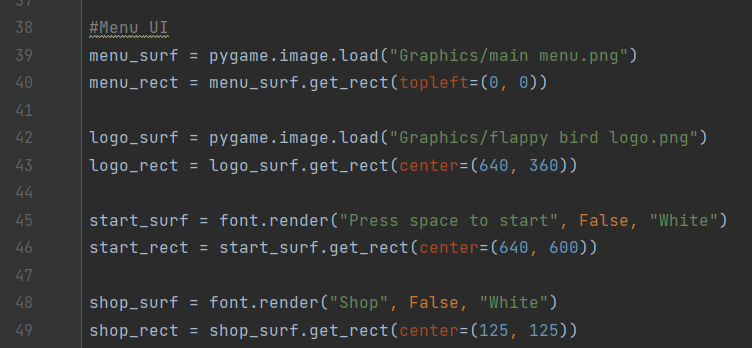
If coins isn’t sufficient, and player pressed buy: buyfailed = True, bird1 = True

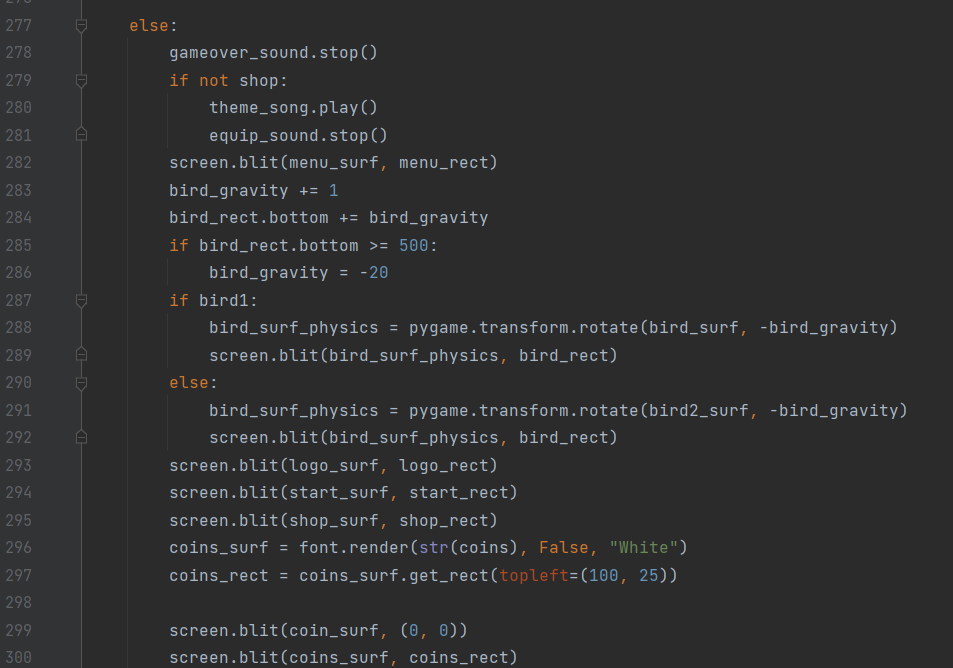
If player presses equip bird skin 1 : bird1 = True

If player pressed equip bird skin 2 and bird skin 2 has been bought: bird1 : False

Only available if the game isn’t running (game\_active = False)

1. Interface
2. **Main menu:**



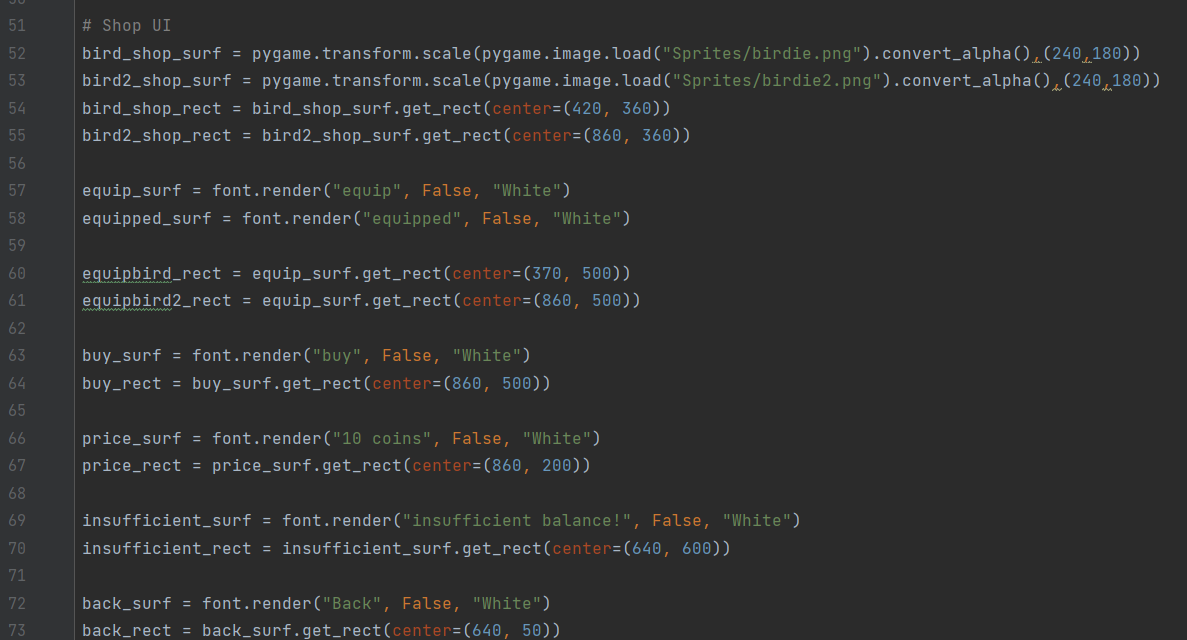


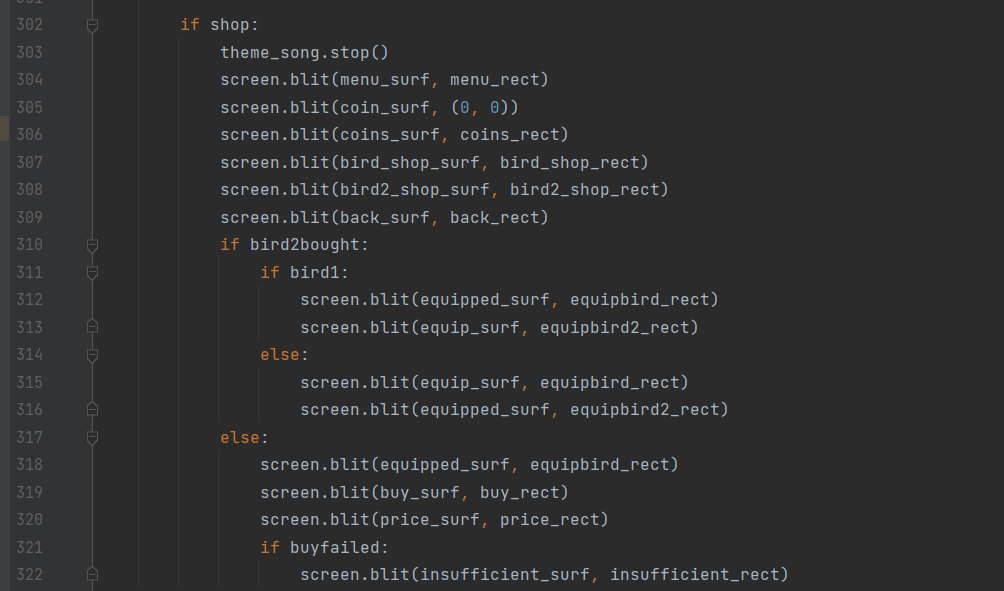
Located in while True loop – if game\_active’s else statement

Main menu’s UI, that displays menu background, Flappy Bird Logo, start text, shop button,

Coins that are attained from the game display, and the bird’s flapping animation (with it’s surface and rectangle written in iii. game UI).

1. **Shop menu:**





Located in while True loop – if game\_active’s else statement – if shop

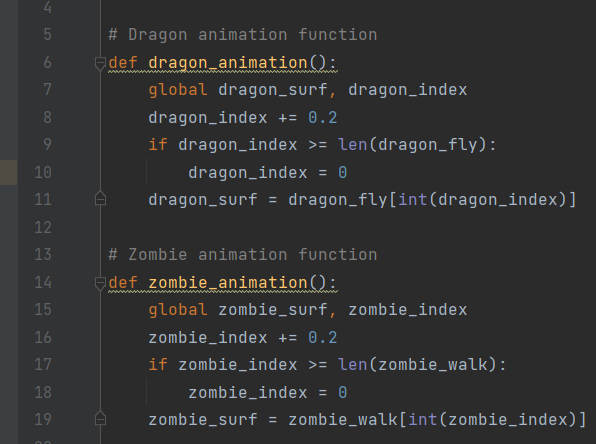
Shop menu’s UI, that displays main menu background, total coin’s display, back button to go back to main menu, two of the bird skins that you could buy.

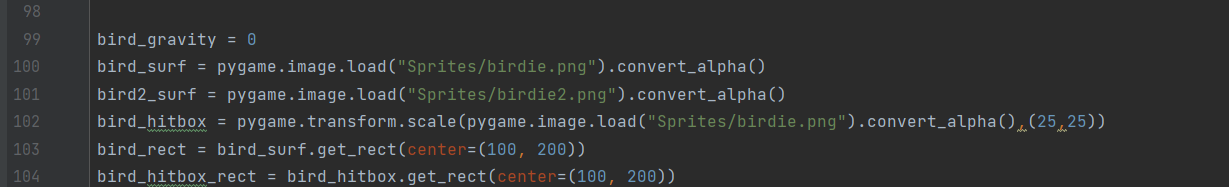
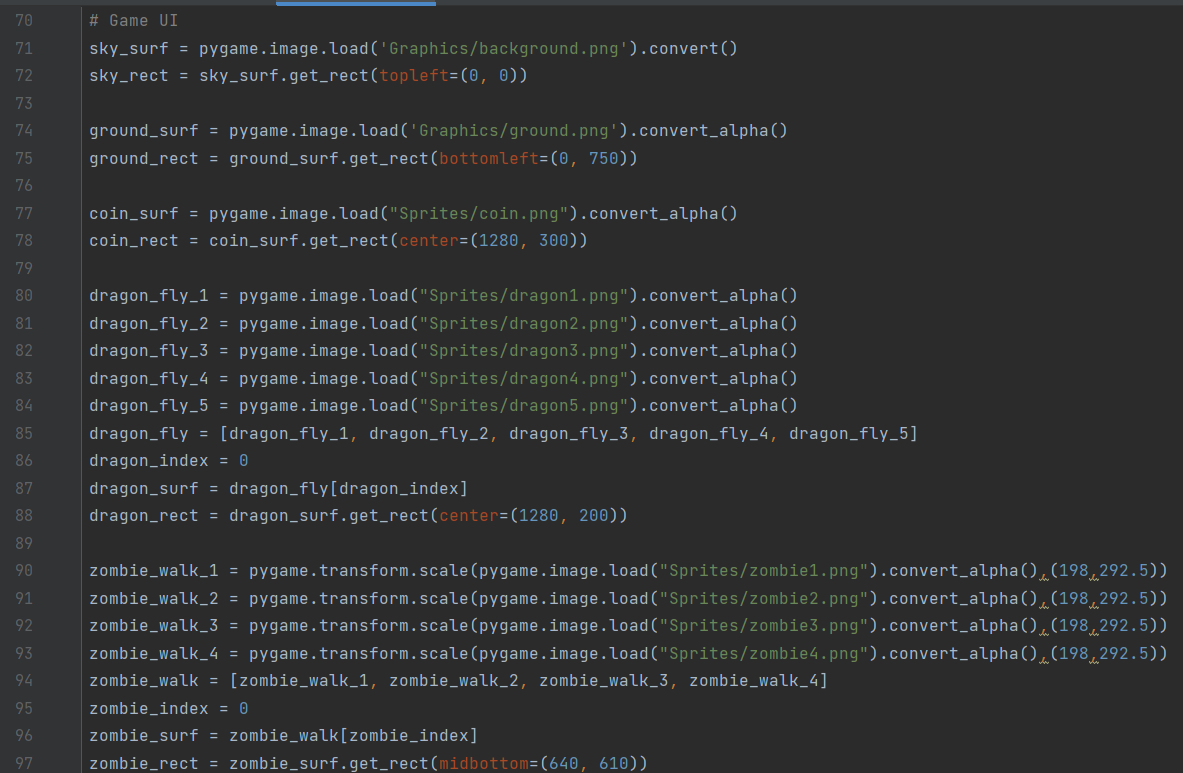
Also has an if else statements that controls the display of the texts and game states:

buy button and price tag displays if bird2 hasn’t been bought (bird2bought = False), insufficient text displays if bird2 isn’t able to be bought (buyfailed = True), equip button to equip for both the bird skins, and equipped text if equip is pressed for both the bird skins.

Bird2bought and buyfailed states are explained in Mechanics – Game states

1. **Game UI**

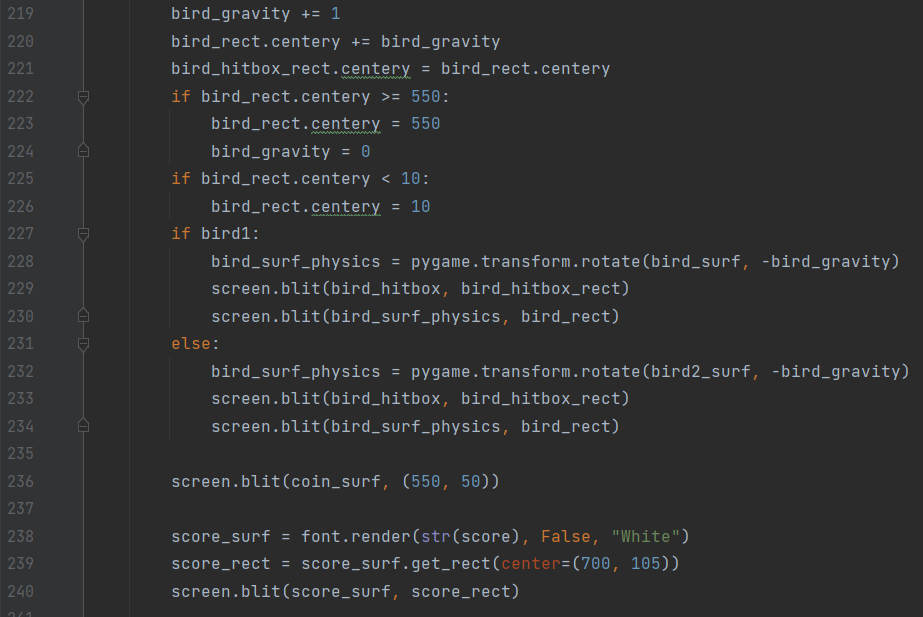
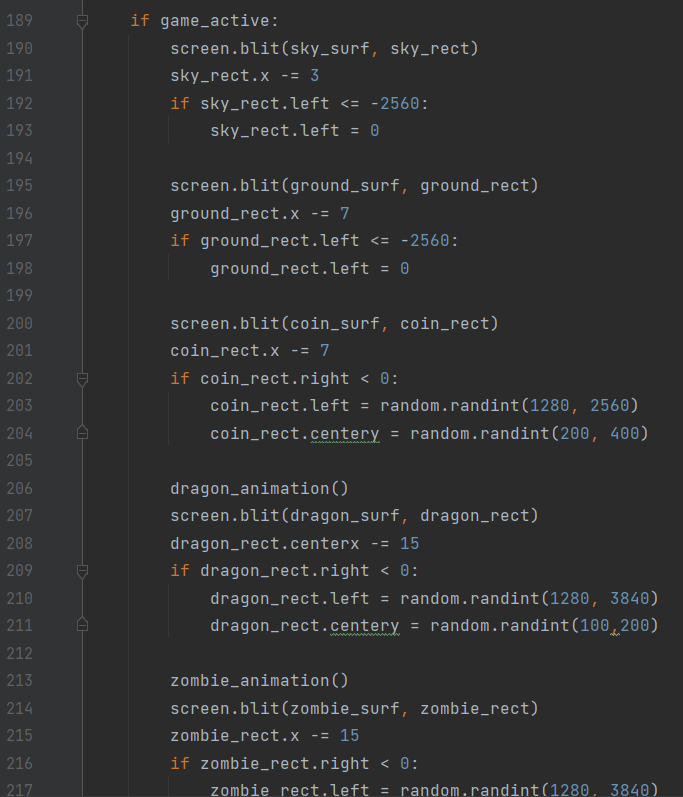




All the surfaces and rectangles for the game UI, including:

The sky, the ground, collectable coins, animated dragon, animated zombie, the bird, the bird’s hitbox.

Animated dragon and zombie controlled within def dragon\_animation() and def zombie\_animation()

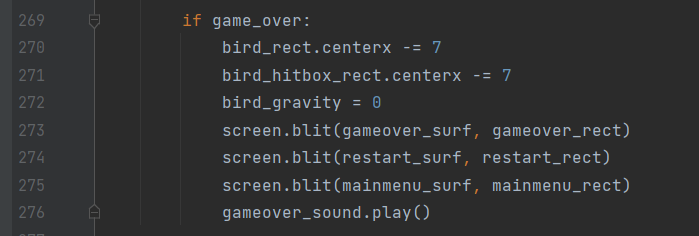
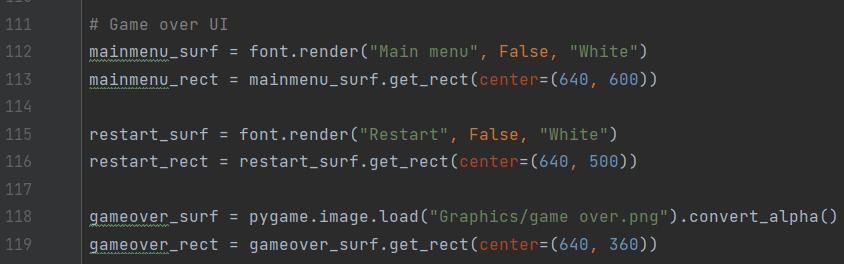


Located in while True loop - if game\_active

Displays the moving sky, the moving ground, the moving coins, the animated and moving dragons, the animated and moving zombies, the bird including it’s overall physics, an if else statement whether bird skin 1 or bird skin 2 is equipped to display either, the coin symbol for the score display, and lastly the score display

Bird1 state is explained in Mechanics - Game states

1. **Game over UI**

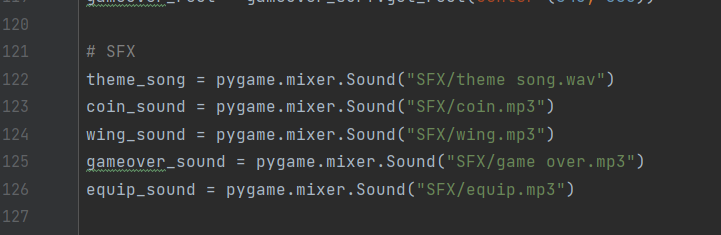


Located in while True loop - if game\_active – if game\_over

Displays the game over surface, the restart button, and the main menu button.

Since the game is still active, the game still plays in the background. Though having the statement in 270 and 271 will make the bird slowly fade away to the left, as it would look like it’s being dragged down by the monster.

1. Music and SFX (Sound Effects)

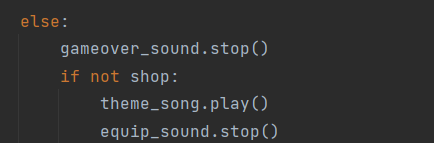


Music and sound effects are taken from Sonic.exe’s and Flappy Bird’s soundboard

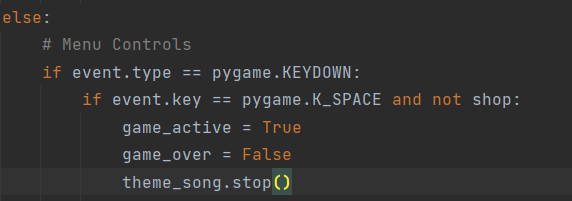
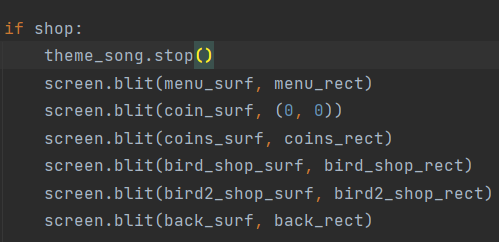
The list of sound effects that is used:

* Theme song.wav (Sonic.exe’s Green Hill Theme Song)

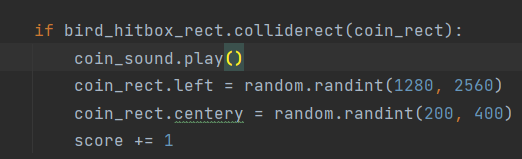
Plays in the main menu (game\_active = False, shop = False)



Stops when in game (game\_active = True), stops when in shop UI (shop = True)

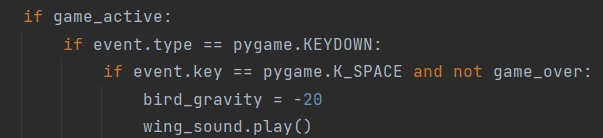


* Coin.mp3 (Original Sonic’s coin collected)



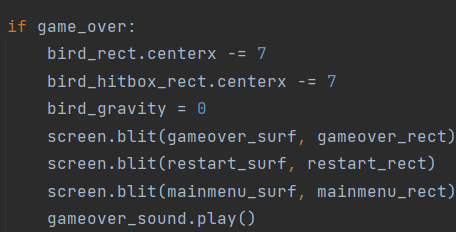
Plays if bird hitbox’s rectangle collides with coin (game\_active = True, game\_over = False)

* Wing.mp3 (Flappy bird’s wing sound)

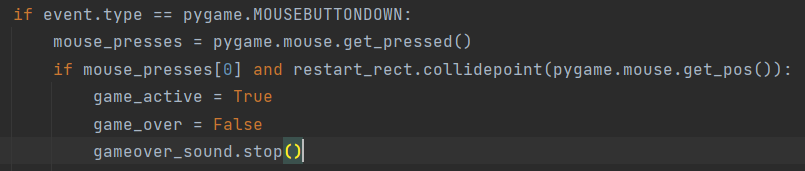


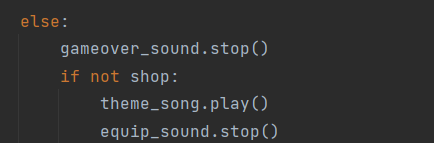
Plays if game\_active = True, game\_over = False, player presses space key

* Game over.mp3 (Sonic.exe’s game over sound)



Plays if game\_active = True, game\_over = True

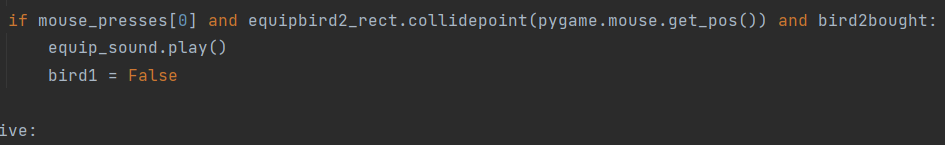




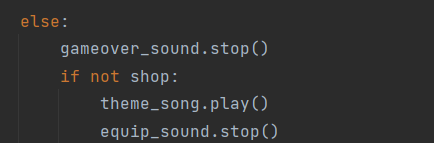
Stops if game\_active = True, and game\_over = False (Player press restart on game over UI)

Or if game\_active = False

* Equip.mp3 (Sonic.exe’s evil laugh)



Plays if game\_active = False, Bird skin 2 is bought (bird2bought = True), and player presses equip bird skin 2 button



Stops if player goes back to main menu (game\_active = False, shop = False)

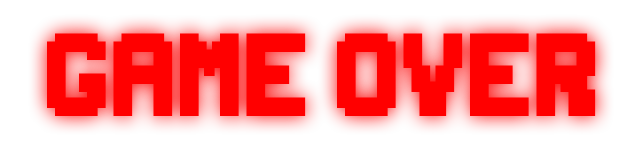
1. Sprites
2. Main menu background



1. Flappy Bird logo



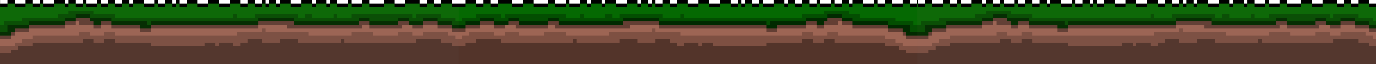
1. Game over logo



1. Game UI background



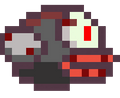
1. Game UI ground



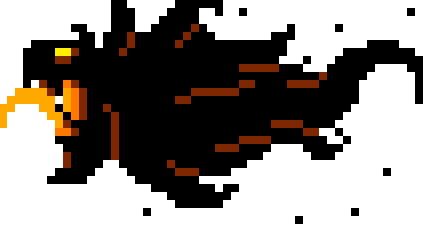
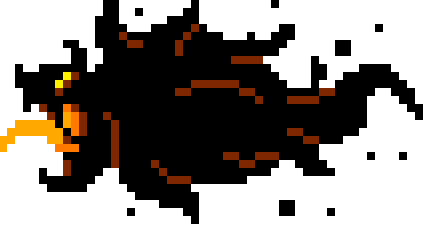
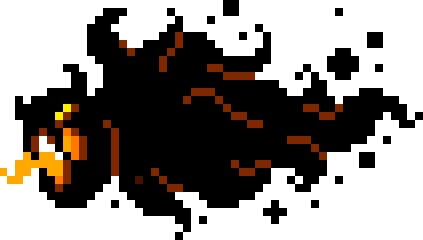
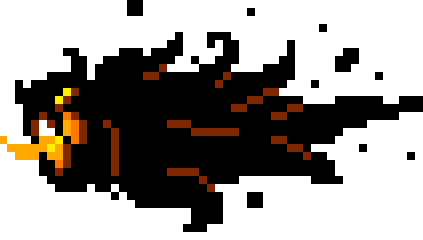
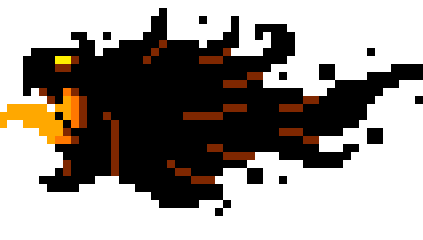
1. Bird (skin 1)



1. Bird (skin 2)



1. Demon dragon (animated)



1. Zombie (animated)





1. Coin

