

NOT YOUR ORDINARY PLATFORM GAME

Name : Muhammad Erizky Suryaputra
ID : 2201797052

INTRODUCTION

This game is a game that is not like your any ordinary platform games, hence the name, you will find and encounter many things that is way out of your expectations, granted this game is not easy but it is fun nonetheless.

PROBLEM

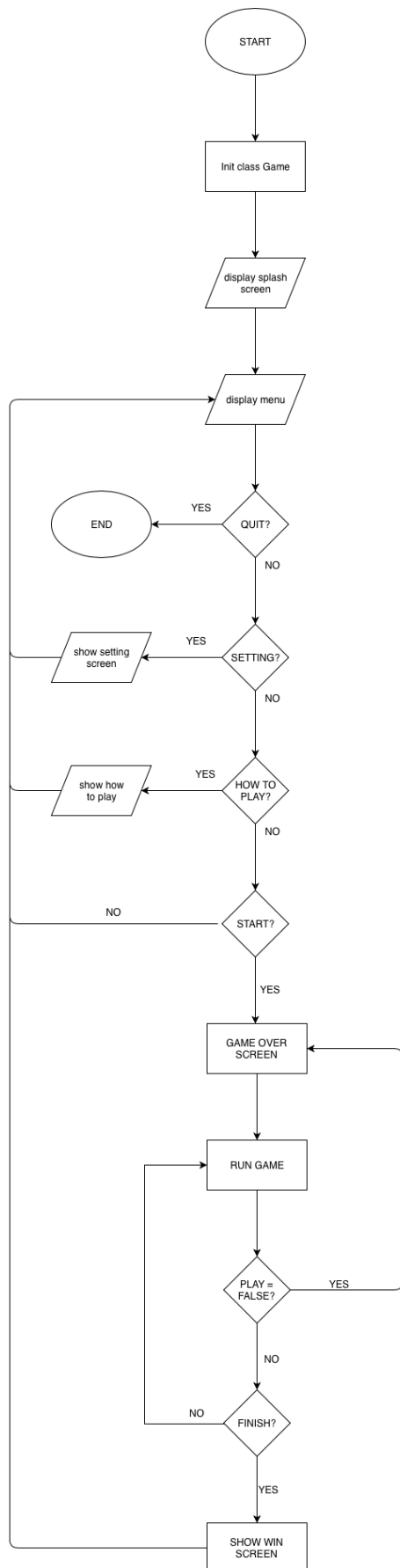
Why I choose to create a game instead of an app is because I want to test my skill, how capable I'm in creating a video game and also to increase my knowledge in video game making. To test my skill, I was going to recreate the one game who started it all which is '*Super Mario Bros*' using pygame. Later, after hearing people thoughts about the idea, they said the game was uninteresting and suggested to make it more challenging. Thus I scrapped my original plan and instead recreated the knock-off more harder version called '*Cat Mario*' which explain why the game use the '*Super Mario Bros*' sprite.

The purpose of creating this game is to gives the player a satisfied feeling when they beat the game, because the harder the difficulty of the game, the more satisfied when they beat it. And also to give a fun immersion when they played the game.

WHAT API DID I USE?

Pygame. The module that's being used for the core of the game. Every single thing from this game is being done solely using this single module only.

DESIGN



Player
+ self.image + self.rect + self.vel + self.acc + self.pos + self.walking + self.jumping + self.dead
+ images + jump + jump_cut + update + anime

Enemies
+ self.image + self.rect + self.vel + self.pos + self.dead
+ images + update + animate

Objects
+ self.image + self.rect

Game
+ pygame.init() + self.screen + self.clock + self.highscore + self.run + self.lives + self.checkpoint + self.finish
+ load + new_game + run_the_game + events + update + draw + splash + menu + game_over + win + wait + draw_text

- Gameplay

The main concept of the game is to survive from all the obstacles that get in your way and find your way to the finish line. The player can walk left or right but not both at the same time and also jump high or low depending how long you held the jump button to avoid the obstacles. If the player collide with the obstacle they die, if the player fell off the screen they die. After finishing the level, player is given an instruction about how to get the second mode called '*SPEEDRUN*' mode. In this mode the player is expected to have memorized all the event that is going to happen and will countdown how fast they can beat the game with that in mind. There's also a settings to change the window mode, show fps, and mute all sounds.

- How it Works

The whole game is being put in a loop so that it will be able to restart if the player made change to the window mode setting. In the loop it will first initialized the class Game and everything on it, then it will show the splash screen, after that the player will be greeted with the menu screen.

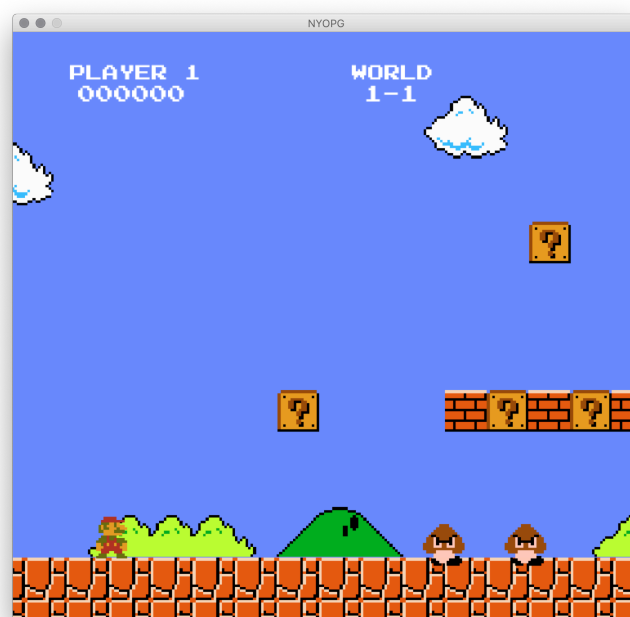
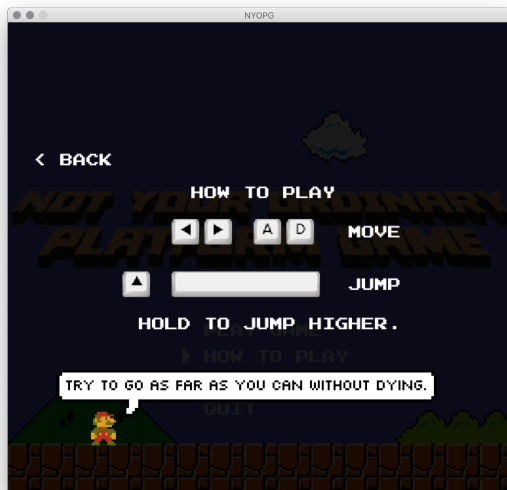
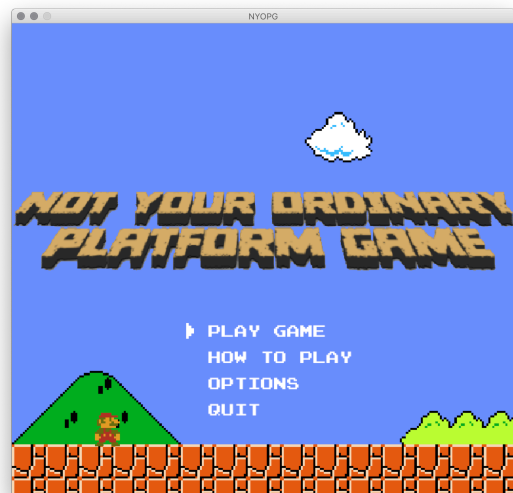
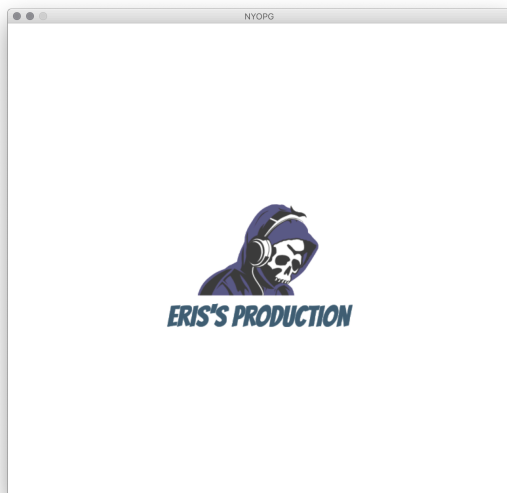
If the player choose the play game, it will move to another loop so when the player die they didn't go back to the menu. If the player choose how to play, it will show the how to play screen while still inside the menu. If the player choose options, it will show the setting screen while still inside the menu. If the player choose quit, it will break out from all the loop and ending the game.

While playing, the player move depending on what key the player input, for example: if the player press the left key, the player x position will be decreased so that it will move to the left and vice versa. If the player press the jump key, it will give the minimum y position for the player to achieve when jumping. And if the player is holding the jump key it will give the corresponding y position until it reach the maximum height, so that the player start to fall even if the player is still holding the jump key.

When the player position is on the middle of the game, it will start to 'scroll' everything back when the player 'move forward', creating the effect as if the player is moving forward. This can be achieved by decreasing the whole (except the player) x position when the player is moving forward while the player is on the middle of the screen and keeping the player in that exact middle position.

While colliding with the enemies, the game must check first if the player is running straight into them or if the player is jumping from above. This checking is needed in order to achieve the ability to stomp on the enemies and directly killing them. For example: if the player is running straight into the enemies (to the right), that means the player velocity is greater than 0 and the player y position is the same or less than the enemies, if the checking found both situation as True then that mean the player is indeed running straight into them and will proceed to give death to the player.

RESULT



REFERENCE

<http://www.cat-mario.com/> (The game I'm recreating and music)

<https://github.com/justinmeister/Mario-Level-1> (For images and sound)

<https://www.youtube.com/watch?v=uWvb3QzA48c&list=PLsk-HSGFjnaG-BwZkuAOcVwWldfCLu1pq> (Template of the game)

<https://www.youtube.com/watch?v=H2r2N7D56Uw&feature=share> (Splash screen)