

Final Report for algorithm programming

Project: Piano panic!



Lecturer:

JUDE JOSEPH LAMUG MARTINEZ, MCS

Muhammad Raihan Zulfi

2602229673

**Binus School of Computer Science Undergraduate Program
Universitas Bina Nusantara
Jakarta
2022**

I. Description (what our project is):	3
II. Inspiration-related work:	3
III. Problem Analysis (why we're doing our project):	3
IV. Storyboard & Mock-Ups:	4
V. Project Screen Shots & Link:	5
B. Link to GitHub Pages : https://wowcodesusah.github.io/hcimilestone/	6
C. Link to Github Link : https://github.com/WowCodeSusah/hcimilestone	6
VI. Plan of evaluation:	6

I. Description:

Name: Piano Panic!

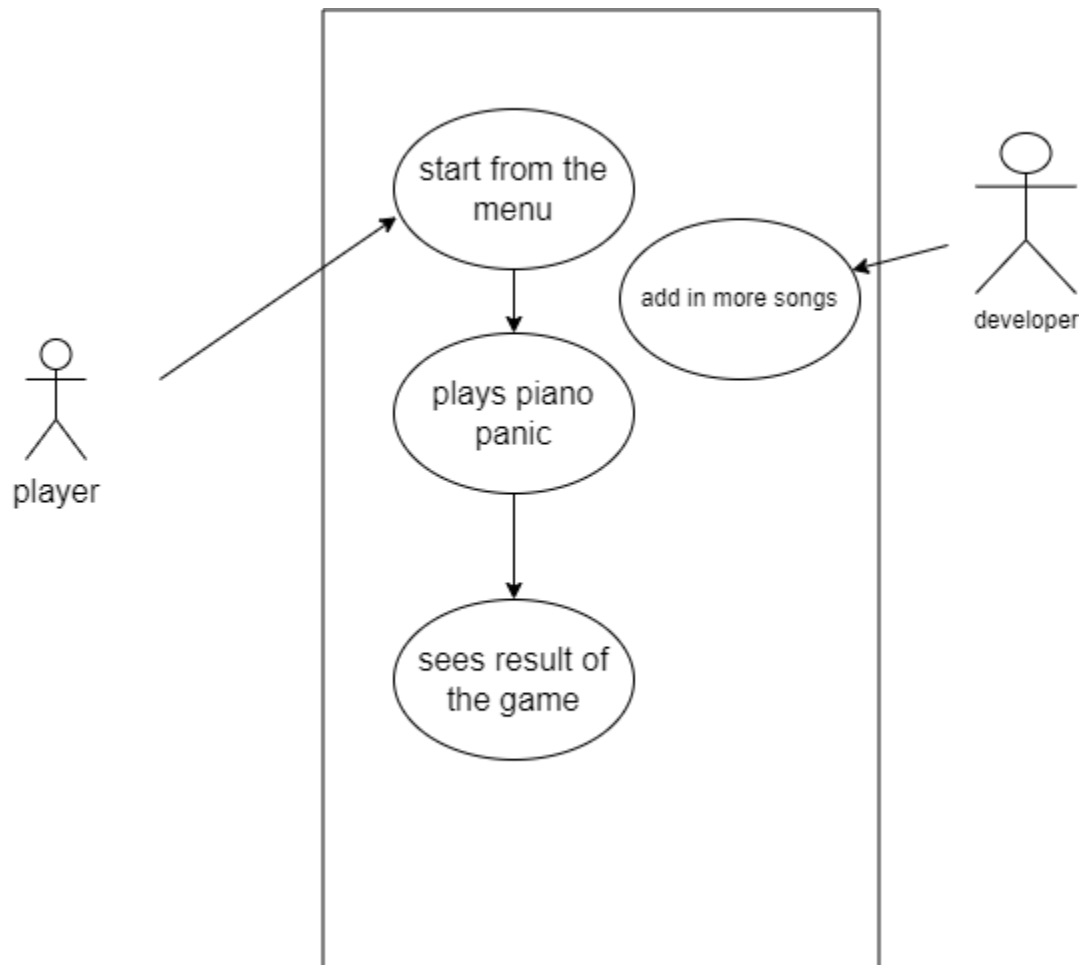
Similar to piano tiles, it's a rhythm where the objective is to click on the 4-bar rectangular box that falls down. With each press, the speed at which this box falls rapidly increases hence the name piano panic comes in.

Modules:

Pygame

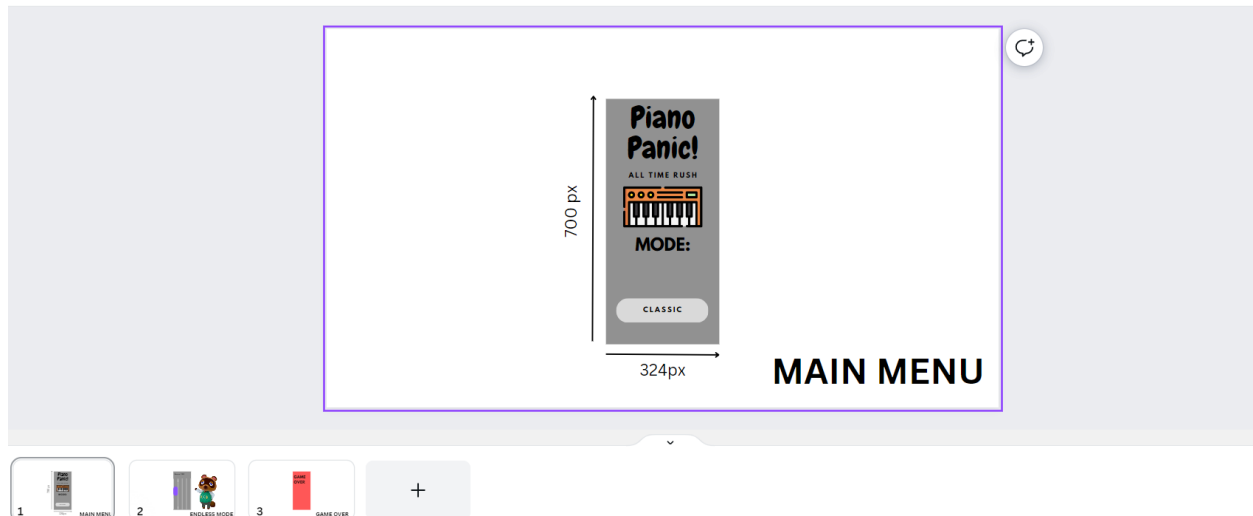
Json

II. Usecase diagram:



III. Design:

Used canva to design the screens:



IV. Project Screen Shots & Link:

A. Intiialsing

```
import pygame
import random
import json
from variables import Block

# Initializing Pygame and font
pygame.init()
clock = pygame.time.Clock()
score_font = pygame.font.Font('Fonts/Futura condensed.ttf', 32)

# Color
WHITE = (255,255,255)

# Size of the window application
screen = width, height = 324, 700
block_width = width // 4 # Creating 4 blocks for 4 Lanes
block_height = 130

# Displaying the game with no frame
window = pygame.display.set_mode(screen, pygame.NOFRAME)
pygame.display.set_caption("Piano Panic!")

# Images
bg_image = pygame.image.load('images/background1.png') # Background image
play_image = pygame.image.load('images/playbutton.png')
play_rect = play_image.get_rect(center = (width//2, height-80)) #becomes centered and a possible button
reply_rect = play_image.get_rect(center = (width//2,height-80))
gameover_image = pygame.image.load('images/gameover.png')
```

B. Inputs

```

running = True
while running:
    pos = None # When it loops complete, it'll reset due to the loop
    window.blit(bg_image, (0, 0))

    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False
        # Will stop the game if you press esc or quit
        if event.type == pygame.QUIT:
            running = False
        # will stop the game if you press esc or quit
        if event.type == pygame.KEYDOWN:
            if event.key == pygame.K_ESCAPE:
                running = False

        #if the mouse is being clicked
        if event.type == pygame.MOUSEBUTTONDOWN:
            pos = event.pos

```

C.

D. Different screens

```

if home_page:
    window.blit(play_image, play_rect)
    # If you click on play it go on to the game page
    if pos and play_rect.collidepoint(pos):
        home_page = False
        game_page = True

        pos = None #reset

        x = random.randint(0, 3) # It will randomly create a block from either 4 lanes
        b = Block(x * block_width, block_height, window) # Creates the block
        block_group.add(b)

        # you can change which song you want by changing the number depending on the notes file
        notes_list = notes_dict['2']
        notes_count = 0

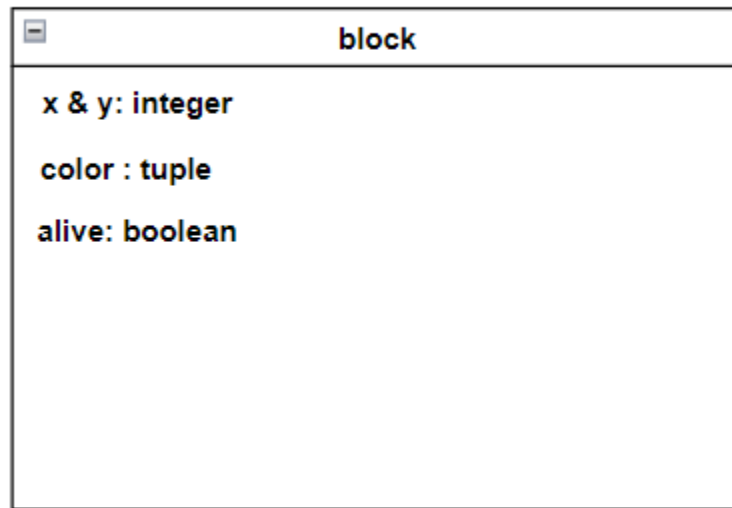
if game_page:
    pygame.mixer.music.set_volume(0.1) #Lower the background music when playing
    for block in block_group:
        block.update(speed)
        img1 = score_font.render(f'Score : {score}', True, WHITE)
        window.blit(img1, (70 - img1.get_width() / 2, 10))

    if pos: # it checks if the position of the mouse clicked the block
        if block.rect.collidepoint(pos):
            block.alive = False
            score += 1
            pos = None

            note = notes_list[notes_count]
            pygame.mixer.Sound(f'piano/{note}.mp3').play()
            notes_count = (notes_count + 1) % len(notes_list) # once it reaches at the end of the song the length will be the same amount of keynotes which means it leads back

```

V. Class diagram



consists of 1 class

VI. Lesson learned:

- VII. Creating a piano tile-like game was a challenging but rewarding experience for me. Throughout the process, I encountered several struggles that taught me valuable lessons about game development and programming.