Flappy Bird Project Report (Manny Bird)

Title : Manny Bird

Name : Mann Sharma

Course : CSE

Date : November 2025

Registration Number : 25BAI10379

Introduction ~

This project is a remake of the popular mobile game flappy Bird using Python and the pygame library. The main idea of the game is very simple: the player controls a bird that must fly between pipes without hitting them. Even though the gameplay looks easy, it is actually very challenging and addictive.

functional requirements ~

- bird moves with spacebar
- pipes spawn with random gaps
- collision detection
- score tracking
- reset after game over

Non-functional Requirements ~

- Smooth gameplay at 60 FPS
- Responsive controls
- Minimal lag or glitches
- Simple UI and sound effects

System Architecture ~

The game uses a modular structure with separate files for main loop, assets, and logic. Pygame handles rendering, input, and timing.

- main loop: events + rendering
- bird class: position + image
- pipe class: position + scoring
- event system: key presses + timers

implementation details ~

Programming Language & Tools ~

- **Language:** Python 12.7
- **Library:** Pygame (for graphics, sound, input handling)

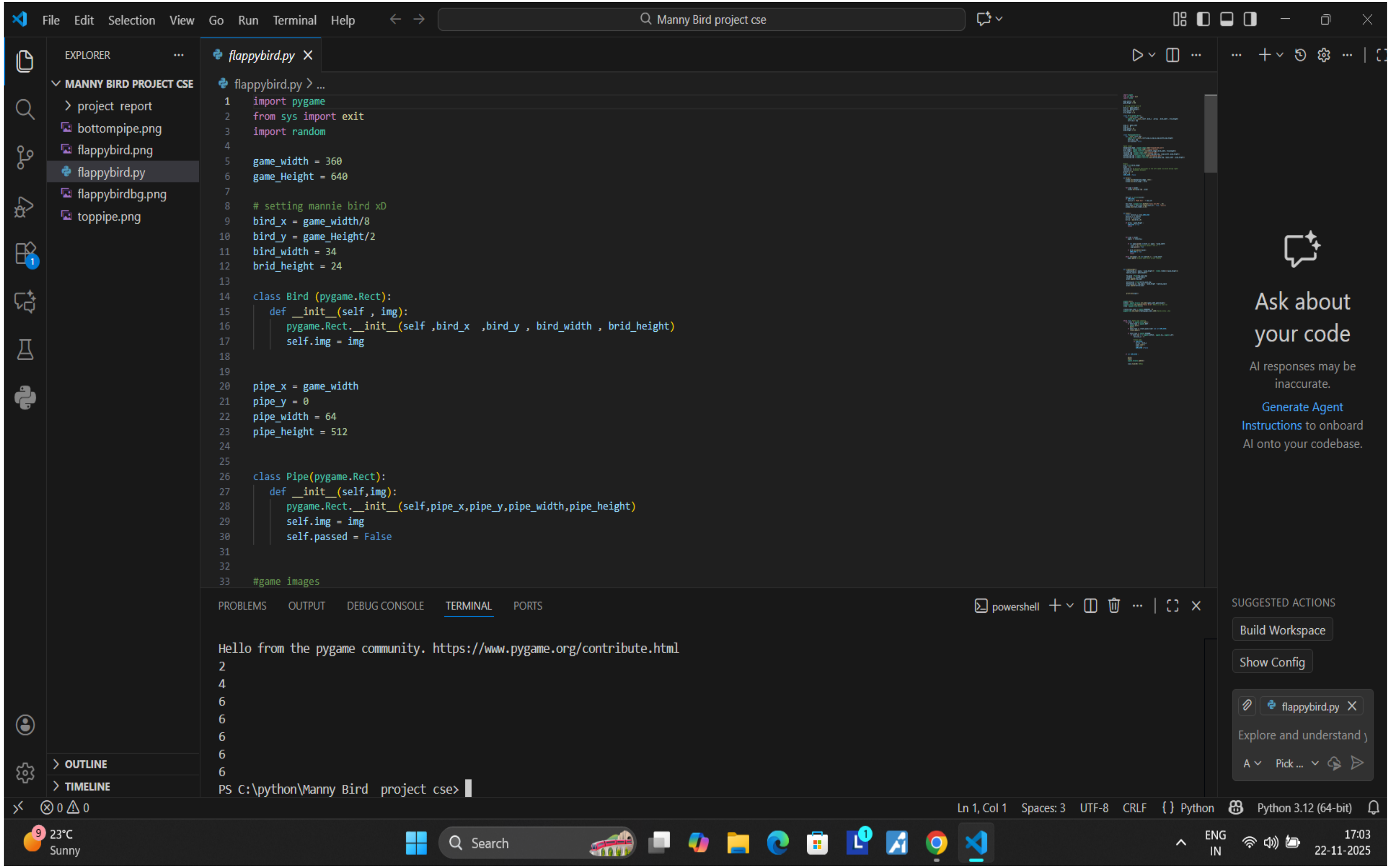
Game Logic -

- Gravity and jump physics for the bird
- Pipes move leftward; bird stays in place
- Collision detection using bounding boxes
- Score increases when bird passes a pipe

Assets Used -

- png sprites for bird, pipes, background
- Fonts for score display

screenshots & results -



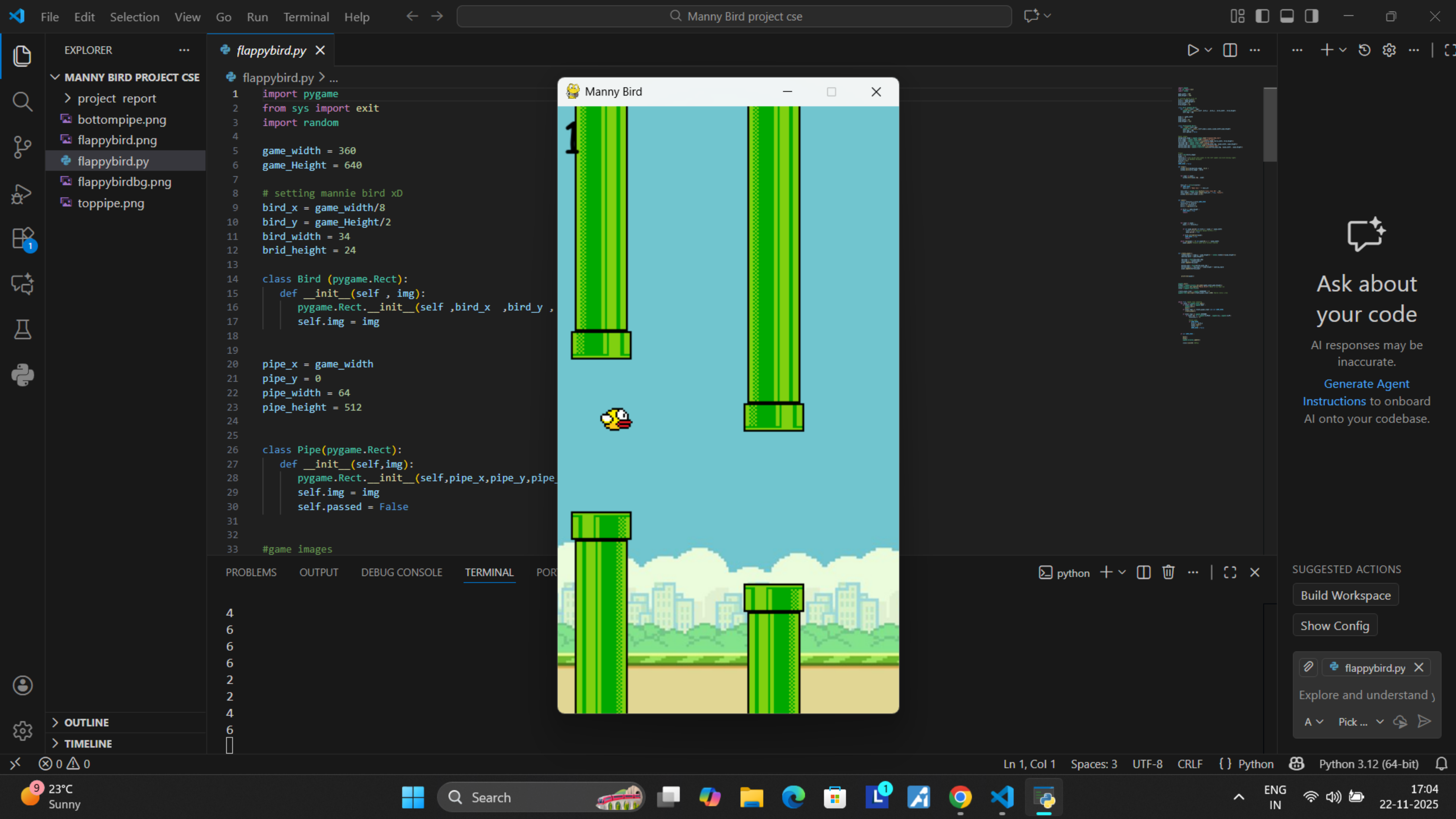
```python
import pygame
from sys import exit
import random

game_width = 360
game_Height = 640

# setting mannie bird xD
bird_x = game_width/8
bird_y = game_Height/2
bird_width = 34
brid_height = 24

class Bird (pygame.Rect):
   def __init__(self , img):
      pygame.Rect.__init__(self ,bird_x  ,bird_y , bird_width , brid_height)
      self.img = img


pipe_x = game_width
pipe_y = 0
pipe_width = 64
pipe_height = 512


class Pipe(pygame.Rect):
   def __init__(self,img):
      pygame.Rect.__init__(self,pipe_x,pipe_y,pipe_width,pipe_height)
      self.img = img
      self.passed = False


#game images
```

```
Hello from the pygame community. https://www.pygame.org/contribute.html
2
4
6
6
6
6
6
PS C:\python\Manny Bird  project cse>
```

Ask about your code

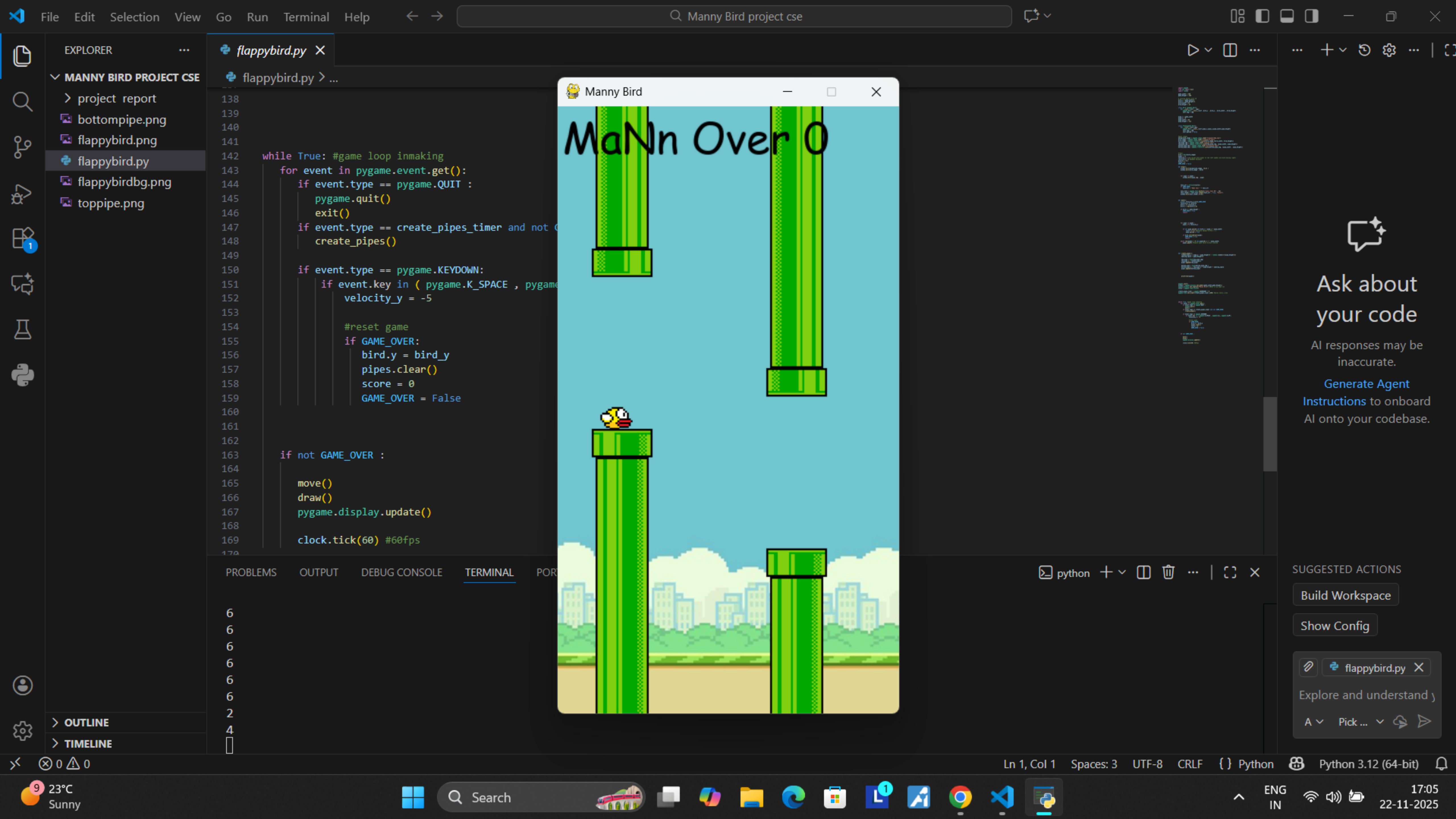
AI responses may be inaccurate.

Generate Agent Instructions to onboard AI onto your codebase.



```python
import pygame
from sys import exit
import random

game_width = 360
game_Height = 640

# setting mannie bird xD
bird_x = game_width/8
bird_y = game_Height/2
bird_width = 34
brid_height = 24

class Bird (pygame.Rect):
   def __init__(self , img):
      pygame.Rect.__init__(self ,bird_x  ,bird_y ,
      self.img = img


pipe_x = game_width
pipe_y = 0
pipe_width = 64
pipe_height = 512


class Pipe(pygame.Rect):
   def __init__(self,img):
      pygame.Rect.__init__(self,pipe_x,pipe_y,pipe_
      self.img = img
      self.passed = False


#game images
```

```
4
6
6
6
2
2
4
6
```



```
while True: #game loop inmaking
   for event in pygame.event.get():
      if event.type == pygame.QUIT :
         pygame.quit()
         exit()
      if event.type == create_pipes_timer and not (
         create_pipes()

      if event.type == pygame.KEYDOWN:
          if event.key in ( pygame.K_SPACE , pygame
              velocity_y = -5

              #reset game
              if GAME_OVER:
                 bird.y = bird_y
                 pipes.clear()
                 score = 0
                 GAME_OVER = False



   if not GAME_OVER :

      move()
      draw()
      pygame.display.update()

      clock.tick(60) #60fps
```

```
6
6
6
6
6
6
2
4
```

Testing approach -

Manually played it for some time tested every possible way and yeah initially there were some bugs like the score wasn’t increasing ,there was a time when it couldn’t detect if the bird touched the ground or not but used a bit of youtube and google and fixed it .

challenges faced -

Looping was very hard and settings pole were very hard to used a bit of youtube . It was quite challenging it took me like 2days to understand how pygame works but yeah with the help of yt and google I managed to get it done .

Learnings -

Learned how newtons is applicable in coding lol like gravity . Also learned about how to use pygame , how to use random function ,how to use include external png's , how to resize them etc.

future enhancements -

add sounds, menu, high scores, bird animation, mobile controls . This was a basic game in future I might make it 3d using unreal engines maybe

References- youtube tutorials and Github examples.