



***SHIV ASHISH SCHOOL***

**COMPUTER PROJECT**

**ON**

**“PYGAME”**

**TOPIC :** ***Pygame***

**GROUP :** ***Dev Radadia* *and* *Dheer Banker***

**CLASS :** ***XII (Science)***

**BATCH : *2020-21***



***Index***

* Certificate
* Acknowledgement
* Python Introduction
* System Requirements
* Game Logo
* Project Overview
* Procedural Design
* Modules Used
* Functions Used
* Source Code
* Screenshots
* Limitations
* Bibliography

### ***Project Overview***

PingPong - A project made by Dev Radadia and Dheer Banker

The aim of the project is to create a ping-pong game with two game-modes – ‘1 Player’ and ‘2 Players’ with a smooth, intuitive gameplay.

* ***Salient Features* :**

1. The project is divided into 6 packages:
   1. ***ai*** – Contains the code governing the AI used in 1 player game-mode
   2. ***image*** – Contains the images used in the project
   3. ***r*** – Contains the screen-wise strings, resources and font styles used in aaathe project
   4. ***screens*** – Contains the code governing every screen in the project, aaaaaaaaawith a module for every screen
   5. ***sound*** – Contains the sounds used in the project
   6. ***sprites*** – Contains the different pygame drawables that have been aaaaaaaaused in the project
2. All the GUI Elements in the project are developed solely using pygame from scratch, in order to keep the GUI consistent throughout the application.
3. The project also includes various sounds in it, which are played when a button is clicked, or when the ball bounces with the paddle or the wall, etc.
4. There are a total of 6 screens in the project :
   1. ***About*** – Tells the user about the developers and the basic controls of aaaaaaaathe game
   2. ***Main*** ***Menu*** – The main screen that has options to go to the other aaaaaaaaaaaaascreens
   3. ***PlayerNames*** – The screen where players can enter their names and aaaaaaaaaaaaaaachoose their paddle colours
   4. ***Game*** – The game screen, where the players can play
   5. ***Pause*** – The screen which comes up when the game is paused, by aaaaaaaapressing ***Pause button*** or by pressing ***P***
   6. ***EndGame*** – The screen which declares the winner of the game that aaaaaaaaaaa was just played
5. Finally, all the screens are bound together by the controller code “***The Game.py***”.

***Procedural Design***

1. To start the application, click on **The Game.py** or run **The Game.py** via a Python console.
2. The main screen appears, which has three options :
   1. **Start** : Takes the user to the **Player Names** screen (details mentioned aaaaaaabelow).
   2. **About** : Takes the user to the **About** screen, which contains aaaaaaaainformation about the project and its basic controls.
   3. **Quit** : Exits the application.
3. **Player Names** screen : Here, the players can set their respective names and aaaaaaaaaaaaaaaaaaaaachoose their colours (default White). From here they aaaaaaaaaaaaaaaaaaaaacan either **Return to Main Menu** or press **Enter** to aaaaaaaaaaaaaaaaaaaaamove to the **Game** screen.
4. **Game** screen : After an initial countdown from 3 to 1, the game starts.

Players can :

* + - * 1. move their paddles using the respective controls (mentioned in **About** screen)
        2. pause the game using the ***Pause button*** or by pressing ***P***
        3. go to the main screen by pressing ***Esc***

1. **Pause** screen : Shows the current scores and has the options to **Resume, aaaaaaaaaaaaReturn to Main Menu** or **Quit**.
2. **Endgame** screen : The screen which follows when the game ends, showing aaaaaaaaaaaaaaaaathe winner’s name, and the options to **Play Again, aaaaaaaaaaaaaaaReturn to Main Menu,** or **Quit**.

***The possible screens and transitions are shown in the image below***

***Game Logo***



***Screenshots***













***System Requirements***

* **Hardware Requirements**
* *Operating System* : Supports Linux, Windows (95, 98, ME, 2000, XP, Vista, 64-bit Windows), Windows CE, BeOS, MacOS, Mac OS X, FreeBSD, NetBSD, OpenBSD, BSD/OS, Solaris, IRIX, and QNX
* *RAM* : 500MB
* *Graphics Card* : AMD Radeon HD 5450
* *CPU* : Intel Pentium 4 2.00GHz
* *HDD Space* : 50MB
* **Software Requirements**
* *IDE* : A Python IDE (eg. Python IDLE, Spyder, Pycharm, Atom, etc.)
* *Python Modules* : Pygame, Numpy, Random and Math

***Modules Used***

*User-defined Modules*

1. r.about
2. r.colors
3. r.endgame
4. r.font\_size
5. r.game
6. r.main
7. r.pause
8. r.playernames
9. screens.about
10. screens.endgame
11. screens.game
12. screens.main\_menu
13. screens.pause
14. screens.playernames
15. sprites.Ball
16. sprites.Border
17. sprites.Button
18. sprites.Label
19. sprites.Paddle
20. sprites.PauseButton
21. sprites.Textbox

*Built-In Modules*

1. pygame
2. random
3. math

***Functions Used***

1. main
2. start\_menu
3. player\_details
4. start\_game
5. pause\_game
6. launch\_endgame
7. launch\_about
8. \_\_init\_\_
9. show\_menu
10. showAbout
11. setAboutText
12. setDisplay
13. PlayerNames
14. handleColorClick
15. getPlayer1Name
16. getColor1
17. getPlayer2Name
18. getColor2
19. setColorButtons
20. playersReset
21. gameReset
22. collides
23. play
24. countdown
25. setGameObjective
26. setMovables
27. setPaddleMargin
28. setPaddleSpeed
29. setBallResetMargin
30. setBounceBias
31. setPlayer1Name
32. setPlayer2Name
33. setBounceAccelera ation
34. setPlayerColors
35. getWinnerColor
36. getWinnerName
37. getScores
38. pause\_game
39. setScores
40. setWinnerColor
41. setWinnerName
42. showEndScreen
43. bounce
44. same\_dir
45. crossed
46. ballReset
47. update
48. setResetMargin
49. setBallSpeed
50. getXSpeed
51. moveUp
52. moveDown
53. rectangle
54. draw
55. test\_surface
56. image
57. rect
58. setHighlightable
59. stayHighlighted
60. staysHighlighted
61. getTextRgb
62. handle\_event
63. define\_event
64. getText
65. setText

***Limitations***

The limitations of the projects are :

1. No option to change the Theme of the project
2. No option to turn off the sounds
3. The project screen cannot be resized
4. No leader-board maintained
5. Joysticks not supported
6. No settings screen present
7. No other game modes like Infinite Play, Timed Play, etc.
8. To be continued…

***Bibliography***

1. <https://www.pygame.org/docs/>
2. <https://docs.python.org/3/>
3. <https://www.geeksforgeeks.org/>
4. <https://pythonprogramming.net/pygame-python-3-part-1-intro/>
5. NCERT Class-XI Computer Science Textbook
6. NCERT Class-XII Computer Science Textbook