Team Members

Akram Ali

Sabbir Ahmed

Rohith Suresh

Hanan Hussain

Pearl John

**Why Choose Our Team?**

Hey there! Choosing our team for the Pong game project? Well, we've got some solid reasons why we're the perfect fit:

**Diverse Skill Set:**

Our team brings a versatile set of skills, spanning mathematics, coding, and game development. We thrive on overcoming complex challenges and crafting innovative solutions.

**Efficient Communication:**

Communication is at the heart of our collaboration. We leverage platforms like WhatsApp and Discord for seamless interaction, ensuring clarity and understanding among team members.

**Dynamic Collaboration:**

Regular check-ins and a warm team atmosphere facilitate smooth onboarding for new members. Our in-person meetings are dynamic, fostering efficiency and an enjoyable working environment.

**Embracing customer needs:**

We're all ears when it comes to understanding what our clients and users need. Open communication and feedback are our secret sauce. Our secret sauce? It's the genuine interest we take in your experiences, challenges, and aspirations. We view every interaction as an opportunity to learn, adapt, and grow.

**Adaptability:**

New tools and tech? Bring it on! We're quick learners, staying ahead of the game in the fast-paced tech world. We're all about teamwork. Each of us brings our own flavor to the mix, making sure we communicate well and get things done. Our Pong game is not just a game; it's a result of smooth collaboration, making data management a breeze and customization a piece of cake.

**Pong Game Development For our Project 2**

**Project Overview**

So, we dove into the Pong game as out second group project, sprinkling it with a bit of math, physics, and coding love. The game features two paddles controlled by the players, a ball that bounces between the paddles, and a scoring system. We used Turtle for Python and spiced up the GUI with Tkinter, following a neat and organized approach to coding. Testing? You bet – we didn't want any bugs ruining the party.

**Components**

Screen Setup

The game window is set up using the Turtle graphics library. The screen is 640x480 pixels, with visible boundaries drawn to represent the playable area.

**Player Class**

The Player class defines the properties and behaviors of a player in the game.

**Attributes**

x: Initial x-coordinate of the paddle.

y: Current y-coordinate of the paddle.

speed: Speed of the paddle movement.

padWid: Width of the paddle.

padHei: Height of the paddle.

score: Player's score.

paddle\_color: Color of the paddle.

paddle: Turtle object representing the paddle.

score\_display: Turtle object for displaying the player's score.

Methods

update\_score\_display: Updates and displays the player's score on the screen.

scoring: Checks if the player's score has reached 10, indicating a win.

movement\_up: Moves the paddle upward within the screen boundaries.

movement\_down: Moves the paddle downward within the screen boundaries.

draw: Clears the previous drawing and updates the paddle's position.

**Ball Class**

The Ball class defines the properties and behaviors of the game ball.

**Attributes**

x: Current x-coordinate of the ball.

y: Current y-coordinate of the ball.

speed\_x: Initial speed of the ball in the x-direction.

speed\_y: Initial speed of the ball in the y-direction.

speed\_multiplier: Multiplier applied to ball speed after each score.

size: Diameter of the ball.

ball: Turtle object representing the ball.

**Methods**

reset\_position: Randomly resets the ball's position within the screen boundaries.

increase\_speed: Increases the ball's speed after each score.

movement: Updates the ball's position and checks for collisions with screen edges and paddles.

draw: Clears the previous drawing and updates the ball's position.

**Keyboard Bindings**

The screen listens for specific keypress events and triggers corresponding methods in the Player instances to move the paddles.

Player 1:

"w" key for moving the paddle up.

"s" key for moving the paddle down.

Player 2:

"Up" arrow key for moving the paddle up.

"Down" arrow key for moving the paddle down.

**Game Loop**

The main game loop (update\_game) continuously updates and draws the players and the ball. It also checks for scoring conditions and updates the screen. The loop runs every 10 milliseconds.

**Execution**

The game starts by creating instances of the Player class for both players and the Ball class. Keyboard bindings are set up to handle player input. The main game loop is initiated, and the game continues until a player wins. The Turtle graphics window is closed when the game is over.

**Technical Aspects**

Back-End Coding (Rohith):

Utilized Python classes and data structures for game mechanics.

Cooked up collision detection algorithms to ensure the ball doesn't sneak through paddles!

Testing (Akram):

Checked each part to ensure it works seamlessly. Took charge of identifying and fixing bugs, ensuring a smooth gaming experience.

Integrated everything in comprehensive integration testing, leaving no room for weird aftertastes.

Documentation (Sabbir):

Jotted down the recipe, detailing system architecture and coding standards.

Wrote down the work put on to the development by each member.

Support and Presentation (Hanan and Pearl):

Took charge of creating a PowerPoint presentation that not only conveys the technical aspects but also engages and captivates the audience.

The presentation showcases the journey of the “Pong” game development, highlighting key features, challenges faced, and the innovative solutions implemented.

**User and Coder Perspectives**

**User Experience:**

Made the game easy to play with Tkinter – simple and fun.

Gave clear instructions and feedback – no one likes a confusing game.

**Coder Insights:**

Kept the kitchen organized with a modular setup – easy to add new flavors (features).

Used Python, Turtle, and Tkinter for a delicious mix in game development.

**Conclusion**

So, why pick us? Because our Pong game isn't just a game; it's a feast of collaboration, creativity, and coding goodness. We're a bunch of university students who know how to turn a project into something awesome. Choose us, and let's make this game a party! 🎉