



Game Designing

Presented By :-

Deependra Rathod(GL)

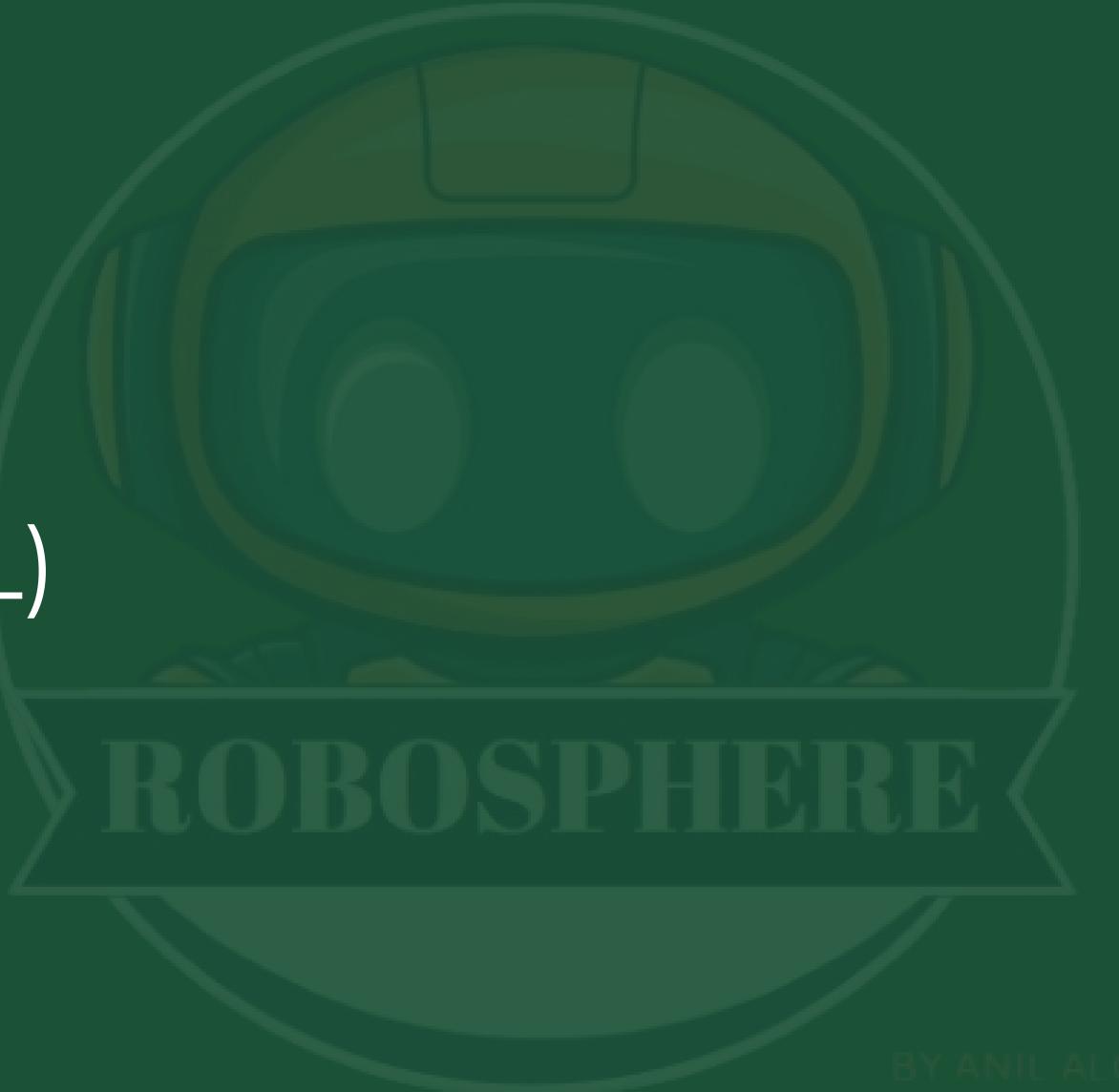
Rishabh Yadav

Govind Singh Rajput

Anil Alok

Guided By :-

Dr. Ankita Kumari (Asst. Prof)



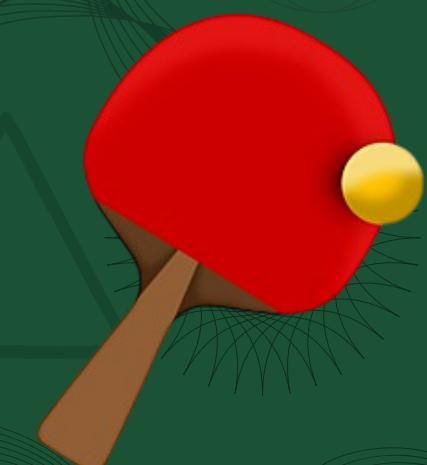
Introduction - What is Game Designing?

- Game Designing = **process of creating rules, visuals, and interactions** for a game.
- Involves **storytelling, creativity, coding, and graphics**.
- Related to our subject → uses **Computer Graphics (visuals, animations)** and **Multimedia (sound, interaction)**.
- Example: Designing our own Brick Breaker game.



History of Gaming

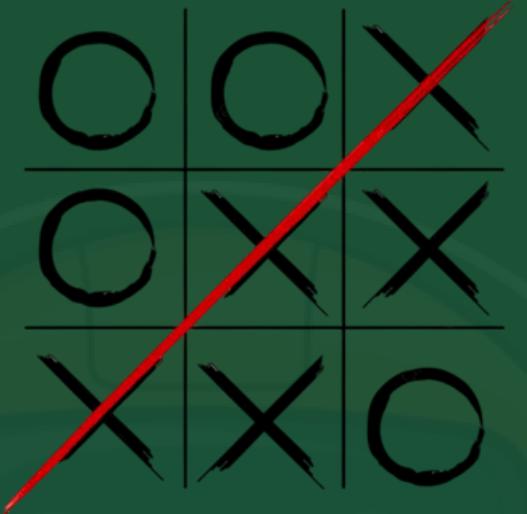
- **1970s:** Text-based games (e.g., Pong, Space Invaders).
- **1980s:** Arcade & console revolution (Mario, Pac-Man, Tetris).
- **1990s:** Handheld consoles + early 3D games (Pokémon, Doom).
- **2000s:** PC/Console boom – GTA, Counter-Strike.
- **2010s:** Online + Mobile gaming rise (PUBG, Clash of Clans).
- **Now:** Cloud gaming, VR & AR, AI-powered gaming



Types of Games

- **Simple/Classic Games:**

- Tic-Tac-Toe
- Snake Game
- Brick Breaker



- **Advanced PC/Console Games:**

- PUBG, Valorant, GTA, FIFA

- **Modern Immersive Gaming:**

- VR/AR (Virtual Reality & Augmented Reality)
- Full body motion sensors & VR suits
- Metaverse-based multiplayer experiences



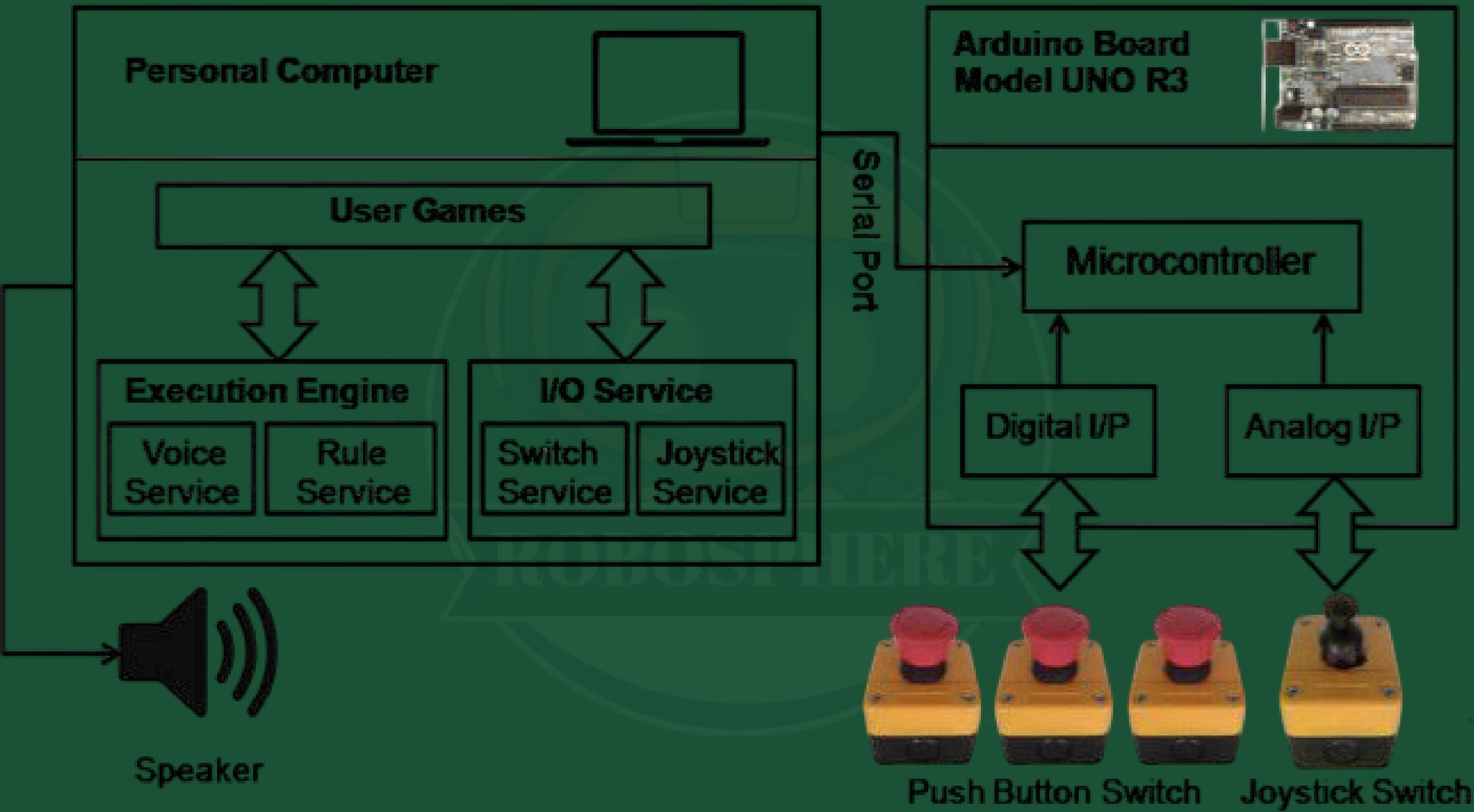
Requirements for Game Development

-  **Coding Skills** – logic, algorithms, problem-solving.
-  **Physics & Math** – collision, speed, bouncing (e.g., Brick Breaker).
-  **Graphics/UI-UX** – attractive visuals, smooth gameplay.
-  **Sound & Multimedia** – music, sound effects, voiceovers.
-  **Creativity & Storytelling** – engaging user experience.
-  **Testing & Debugging** – fixing bugs, improving performance.

ROBOSPHERE

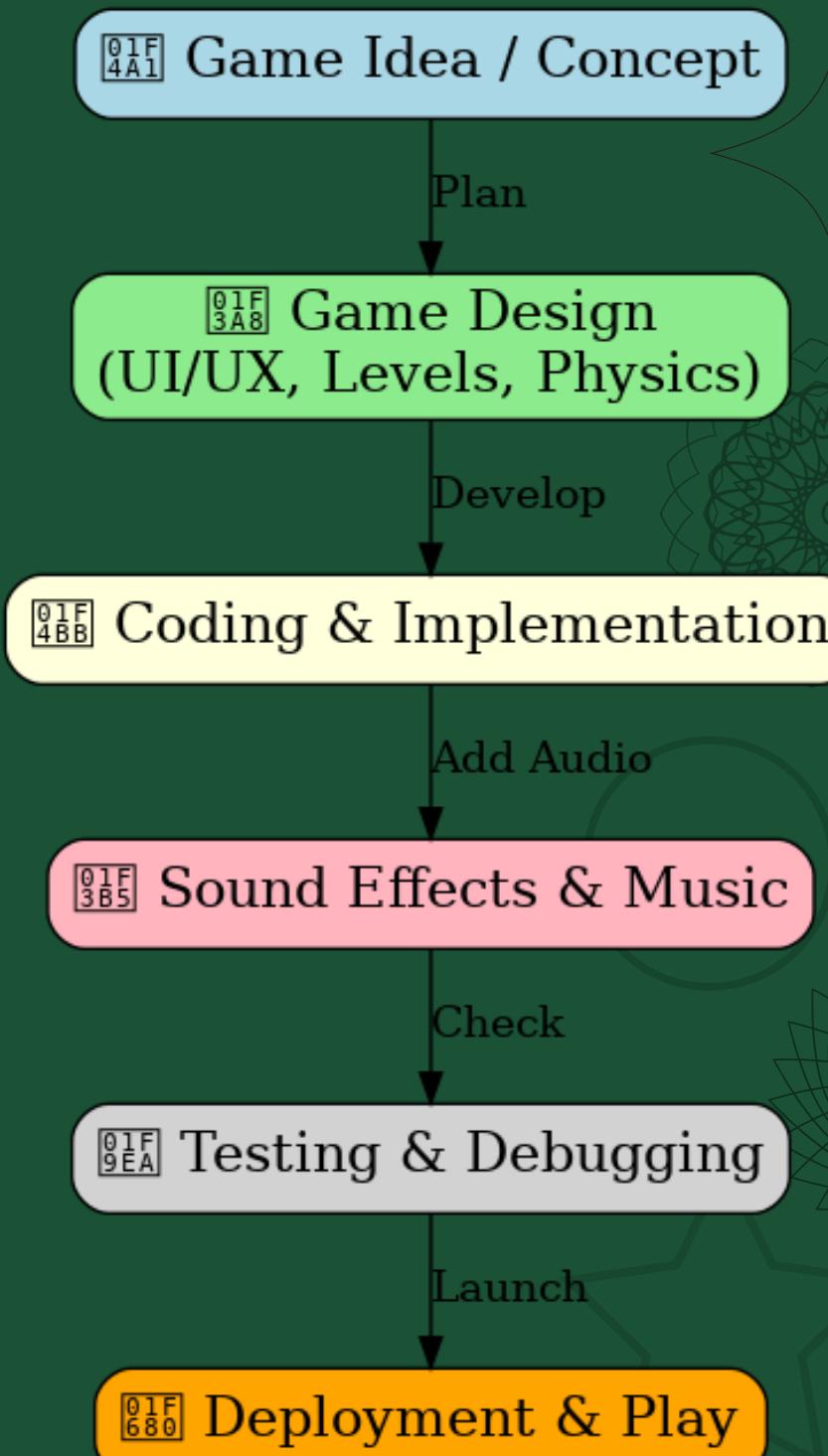
BY ANIL ALOK

The Architecture of game Execution Platform



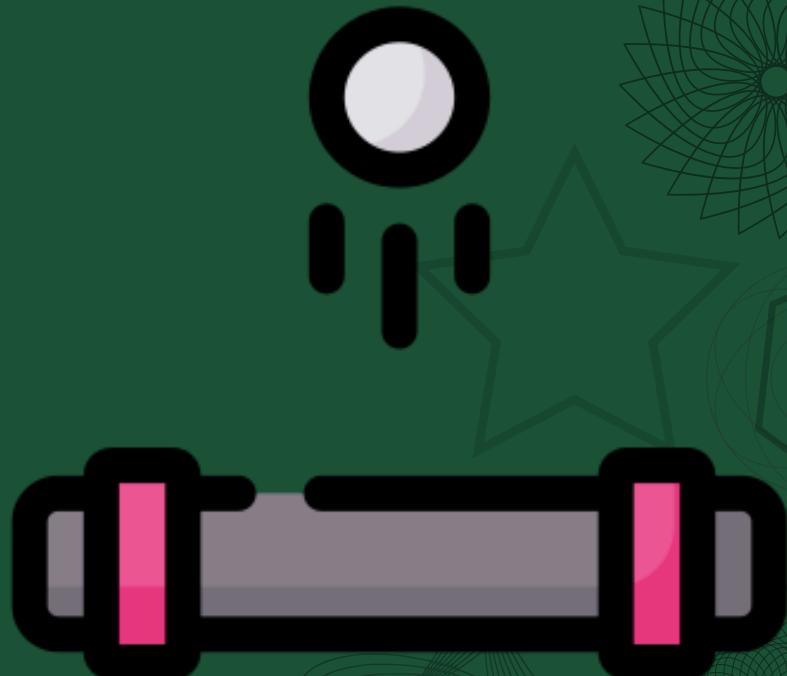
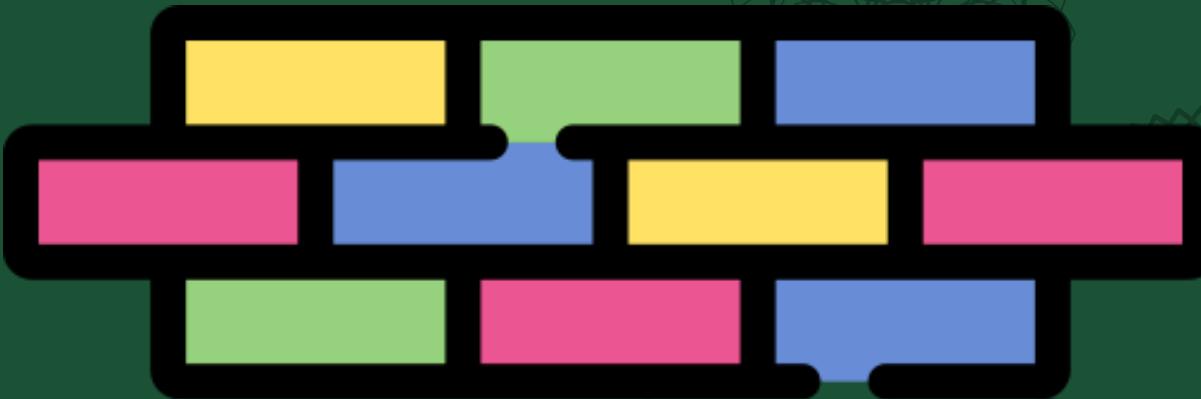
Flow of Game Development

- **Concept & Idea** – decide what type of game.
- **Design** – characters, objects, levels.
- **Coding/Implementation** – programming logic.
- **Integration** – graphics + sound + input devices.
- **Testing** – check for bugs.
- **Final Deployment** – ready to play.



Brick Breaker – Our Game

- **Classic Arcade Game** where a ball breaks bricks using a paddle.
- Objective: **Destroy all bricks without losing lives.**
- Elements:
 - **Paddle** (controlled by mouse).
 - **Ball** (with bouncing physics).
 - **Bricks** (score points when broken).
- Added **custom sound effects** for better experience.



Game Designing in Brick Breaker

- **Working & Designing**

- Paddle at bottom, ball bounces, bricks at top.
- Mouse controls paddle movement, left-click launches ball.

- **Math & Physics**

- Angle of ball, velocity, collision detection.

- **Sound Integration**

- Custom recorded **.wav** sounds:
 - Background Music
 - Ball hitting paddle/brick
 - Life lost / Game Over



Versions of Our Game

- **Version 1:**
 - Paddle controlled by keyboard.
 - No sound effects.
- **Version 2:**
 - Paddle controlled by mouse.
 - Ball launched by left click.
 - Custom sound effects added.
- **Version 3 (Final):**
 - Background music + sound effects integrated.
 - Limitation: Multiple sounds couldn't play simultaneously, solved with minimal setup.

Let's Have a Demo !!!

ROBOSPHERE

BY ANIL ALOK

Challenges & Learnings

- **Challenges:**

- Linking graphics.h in modern compilers.
- Handling collision detection properly.
- Sound integration issues (PlaySound limitations).

- **Learnings:**

- Basics of Computer Graphics (drawing shapes, animations).
- Integration of sound and user input.
- Practical understanding of game physics and logic.
- Teamwork, debugging, and problem-solving skills.

Conclusion

- Understood **how games are designed and developed.**
- Learned **graphics, physics, and multimedia integration.**
- Future Scope:
 - Add levels, power-ups, multiplayer mode.
 - Improve graphics with modern frameworks (**SFML/Unity**).

Thankyou



Any Questions?

