Name: Daravath Saathvik
Module: Game Design I
Course: Digital Game Design

राष्ट्रीय डिज़ाइन संस्थान
NATIONAL INSTITUTE OF DESIGN

Introduction

This project focuses on creating a Pong Game — a digital version of one of the earliest and most iconic arcade games.
The game is designed using **JavaScript (p5.js)**, which provides powerful tools for handling real-time rendering, animation, and user interaction.

However, this version of Pong is not just a technical recreation.
It integrates the philosophical concept of **Voluntarism**, where gameplay and design reflect the idea that **willpower dominates logical reasoning**.
The project aims to merge **philosophical thought** with **interactive design**, allowing players to experience a unique twist on decision-making and adaptability.

Objective

The objectives of this project were to:

1. Understand and implement the fundamental concepts of **game physics**, including ball motion, collision detection, and rebound mechanics.

2. Develop **AI-controlled paddles** that adapt intelligently to the ball's movement using prediction and reaction logic.

3. Introduce a philosophical layer by embedding **Voluntarism** into the game's design, challenging traditional control logic.

4. Utilize **AI assistance (ChatGPT)** not for coding, but for **logic refinement**, idea exploration, and debugging guidance.

Tools and Technologies Used

Tool	Purpose
p5.js	For rendering graphics, animations, and handling input.
HTML / CSS	To structure and style, the game interface.
JavaScript (ES6)	To implement game logic, physics, and AI behaviour.
Visual Studio Code	As the development environment for coding and testing.
ChatGPT (AI Assistant)	Used for conceptual and logic support, debugging insights, and algorithm improvement ideas.

Development Process

Step 1: Game Setup
A 1080×720 canvas was created using p5.js.
Key variables were defined for the ball, paddles, and boundaries.
The function rectMode(CENTER) was used for easier positioning of paddles and walls.

Step 2: Ball Mechanics
A dedicated Ball class was developed with attributes for position, velocity, and radius.

- The update() function handled continuous movement.
- The checkCollision() method ensured realistic paddle and wall rebounds.
- A reset() method re-centered the ball after scoring.

Step 3: Paddle Design
Two classes were implemented — PaddlePlayer for manual control and PaddleAi for automated response.
AI paddles were enhanced using **smooth-follow** logic and small noise offsets for natural, human-like motion.

Step 4: AI Implementation
The AI's behaviour is modelled using a **Proportional Controller (P-controller)** approach.

- Predictive targeting estimates the ball's future position.

Step 5: Testing and Optimization

- Fine-tuned ball speed, paddle reaction rates, and collision precision.

- Solved the “ball sticking” issue by repositioning the ball post-collision.
- Reduced frame drops from Gaussian blur by optimizing draw cycles and lowering filter intensity when necessary.

Problems Faced and Solutions

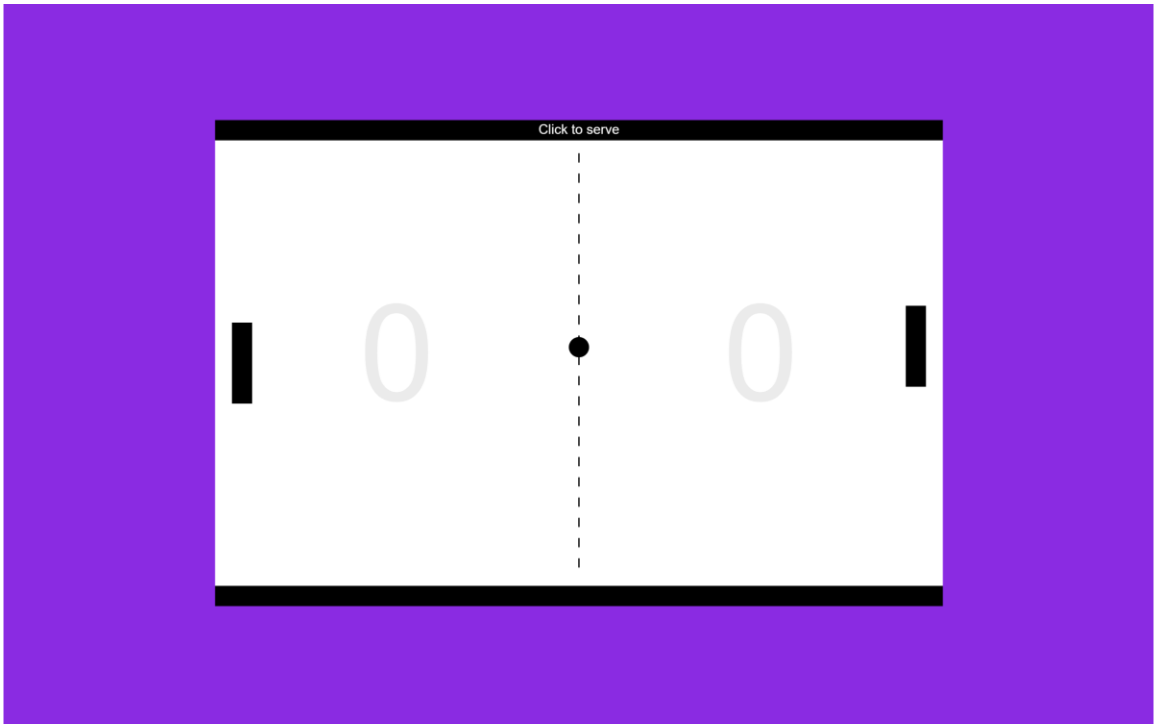
Problem	Description	Solution
Ball Sticking to Paddles	The ball occasionally overlapped and stuck during collision.	Repositioned the ball slightly outside the paddle immediately after collision.
AI Paddles Too Synchronized	Both AI paddles mirrored each other’s movement.	Introduced noise, prediction offsets, and independent timing intervals.
Illogical Controls for Voluntarism Concept	Needed a mechanic to represent willpower over logic.	Reversed the player’s controls (left ↔ right).
AI Behaviour Too Robotic	The AI’s reactions were overly precise.	Added delay intervals, noise, and random “misses” to mimic human error.

Role of AI in the Project

AI (ChatGPT) served as a **logic and design assistant**, not as a code generator.
It was used to:

- Explain complex mechanics such as **collision handling** and **P-controller behaviour**.
- Suggest structural improvements for smoother gameplay logic.
- Offer debugging insights without writing the actual code.

This approach aligns with the theme of **Voluntarism** — AI provided reasoning support, while the actual coding, decision-making, and implementation depended on my **willpower, understanding, and experimentation**.



Click to serve