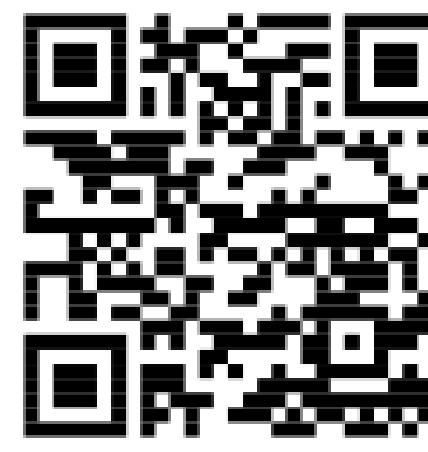




Motivation

Neon Strikers was motivated by a desire to explore real-time game systems development within a structured Agile environment. The team aimed to build a polished, fully playable product that demonstrates proficiency in Unity, C#, and collaborative software engineering practices while delivering a genuinely fun competitive multiplayer experience.

Problem Statement

Neon Strikers began as the passion project of our project owner, who had a vision for a competitive 2D physics-based game the entire team could build together and continue developing beyond graduation. Due to personal reasons, our project owner was unable to attend the showcase and had to step away from the project. The team committed to honoring that vision by seeing it through to a polished, presentable build.

Goals

Deliver
Playable
Build

Complete
7 Agile
Sprints

Polish &
Balance
Gameplay

Key Results

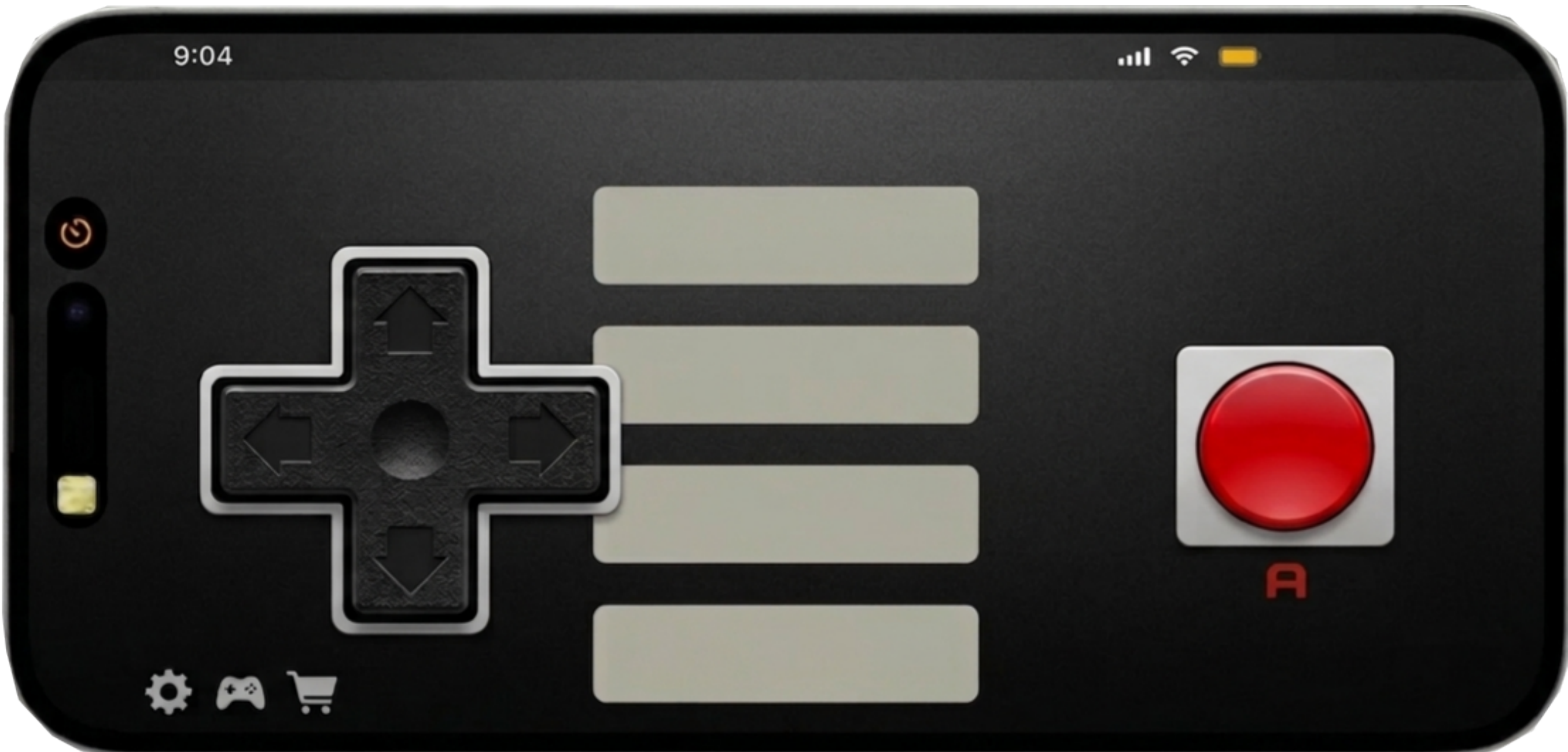
- 7 sprints completed on schedule
- 2 playable characters with distinct physics-based movesets
- 3 arena maps with unique layouts and hazards
- Local 2-player multiplayer verified through playtesting
- Full documentation: sprint reports, retrospectives, demo video

Global Design

Neon Strikers contributes to the growing body of student-developed open-source games and serves as a practical demonstration of component-based Unity architecture, Agile project management, and collaborative C# development, with skills directly transferable to professional software and game engineering roles.

Future Work

- Mobile version streamable to TV using smartphones as controllers
- Browser-based controller support via network connection (e.g. localhost.neonstrikers.us)
- Expand to 4-player support with dynamic map scaling
- Online multiplayer via Unity Netcode for GameObjects
- Downloadable release for Windows, Mac, and Linux for everyone to run locally



Acknowledgement

The team thanks the FIU School of Computing and Information Sciences and our capstone faculty for their guidance throughout this project.

Game Overview

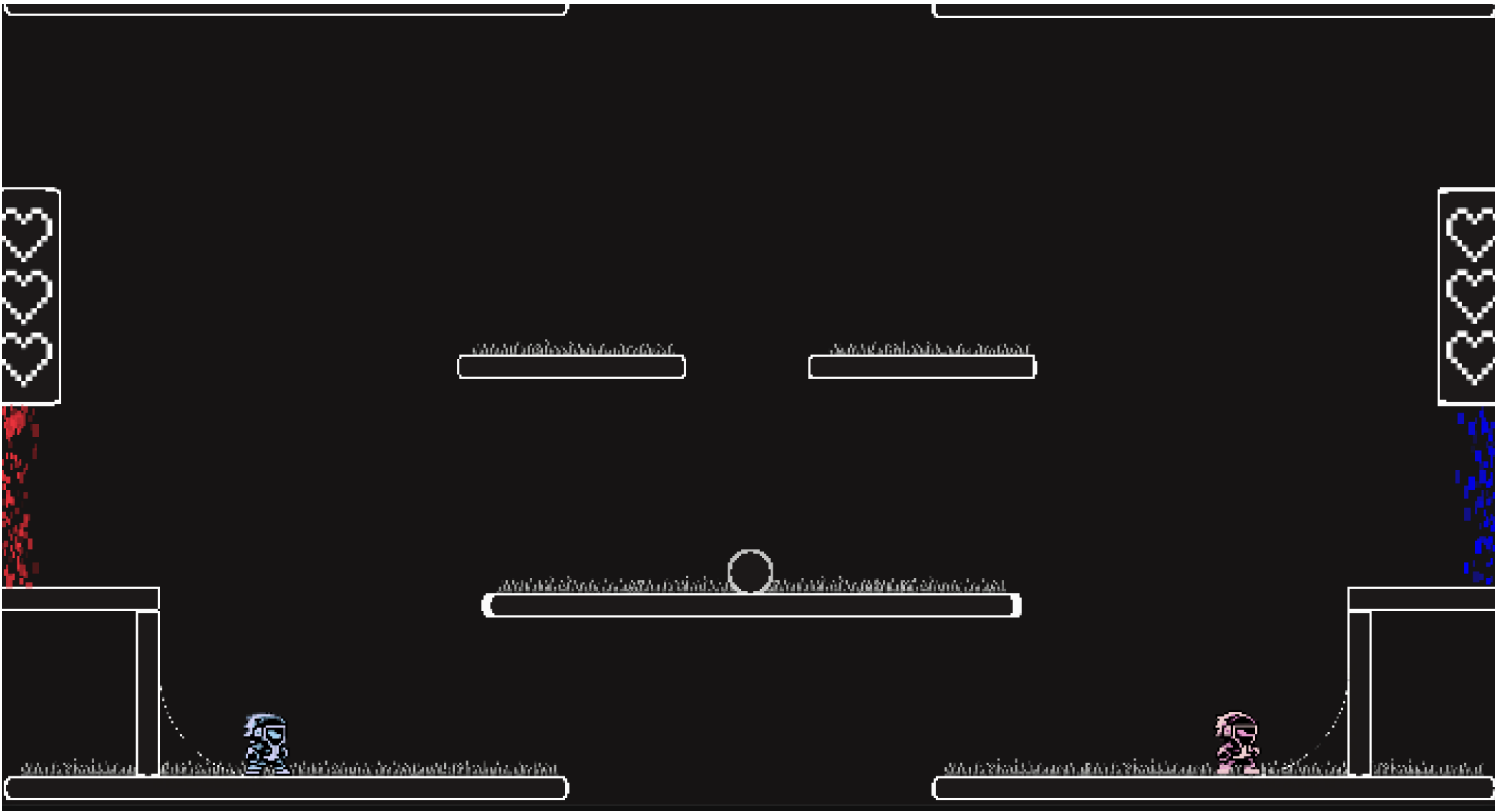
About the Game

Fast-paced 2D arena combat for 2 local players built in Unity 6000.3.3f1. Compete in neon-lit arenas using physics-based mechanics to outmaneuver your opponent.

Gameplay Loop

Select character → Choose arena → Kick off → Use physics to control and shoot the ball → Score in opponent's goal → First to reach the goal limit wins.

Game Preview · Prototype



Technical Architecture

 Unity / 6000.3.3f1	 C# / .NET / Mono	 GitHub / Version Control	 VS Code / IDE
 GitHub / Project Mgmt	 Windows / Build Target	 WebGL / Build Target	 Scrum / Methodology

System Design

PlayerController	Handles input, movement, and dash mechanics
CombatSystem	Manages hitboxes, damage calculation, and knockback
GameManager	State machine: Lobby → Match → Results
ArenaManager	Controls arena selection, ball spawning, and goal detection
UIManager	Drives HUD, score display, and menus

Challenges & Solutions

- Multiplayer Sync:** Deterministic input buffering reduced perceived latency in local split-screen modes.
- Team Continuity:** Mid-project team member departure handled by redistributing tasks and adjusting the sprint backlog.
- Physics Tuning:** Multiple playtesting passes refined velocity, friction, and knockback for balanced feel.
- Scope Control:** Online matchmaking deferred to ensure core gameplay loop was polished for final submission.

Development Process

Sprints 1-2 Core movement, collision, basic arena and player setup

Sprints 3-4 Physics ball mechanics, goal system, HUD, 2-player local multiplayer

Sprints 5-7 Polish, balance tuning, playtesting, final delivery