

Tech Stacks

Frontend

1. Framework:

- **React.js:** For user interface elements like game selection screens, profiles, and leaderboards.
- **Alternative: Vue.js** (simpler syntax, quicker to learn).

2. Styling:

- **Tailwind CSS:** For rapid UI development using utility classes.
- **Alternative: Bootstrap** for pre-styled components.

3. Game UI and Integration:

- Use **Phaser.js** for building 2D arcade-style games directly within the browser.
- Phaser is JavaScript-based, which integrates naturally with a React or Vue frontend.

Backend

1. Framework:

- **Node.js with Express.js:** For managing APIs (leaderboards, game stats, etc.).
- **Alternative: Django** or **Flask** for teams comfortable with Python.

2. Database:

- **Firebase Firestore:** Ideal for real-time leaderboard data.
- **Alternative: MongoDB** or **PostgreSQL** depending on project needs.

Game Development

Option 1:

1. Game Engine:

- **Godot Engine:** Best for 2D arcade-style games, using **GDScript** (similar to Python) for scripting.

2. Game Features:

- **HTML5 Export:** Use Godot's built-in export feature to generate games that run directly in a web browser.
- Integrate dummy data for testing leaderboards using Firebase or a custom API.

Option 2:

1. Game Library:

- **Phaser.js:** Lightweight and easy-to-learn JavaScript library specifically for 2D games.
 - Integrates directly into the web app.
 - Great for arcade-style mechanics (e.g., progression, collisions, animations).
 - Export as a JavaScript module and plug it into the frontend.

2. Game Features:

- Use Phaser's physics engines (Arcade, Impact) for game interactions.
- Build simple games and load them dynamically through the frontend.

Other Tools

1. Version Control: Git/ GitHub

2. Prototyping:

- **Figma** for UI/UX wireframes and design blueprints.

3. Deployment:

- **Vercel/Netlify** for hosting the frontend.

- **Firebase Functions** or **Heroku** for backend APIs.

4. **Development Tools:**

- **VS Code** for coding.
- **Phaser Editor 2D** (optional) for Phaser-specific tooling.

Workflow Overview

1. **Frontend Team:**

- Build the UI for game selection, leaderboards, and profiles.
- Integrate Phaser games into the React (or Vue) components as embedded modules.

2. **Game Development Team:**

- Develop games directly in Phaser.js.
- Share JavaScript modules for frontend integration.

3. **Backend Team:**

- Create APIs for leaderboard management and player stats.
- Use Firebase for real-time syncing.