

# SoftDev P05

ChimiChangas

**DevoRoyale** by Victor Casado, Will Nzeuton, Mark Ma, Daniel Park

**TSD:** 2025-06-06

## Overview

DevoRoyale is a 2D online multiplayer game inspired by ZombsRoyale and Clash Royale. In DevoRoyale, devos will fight with weapons they have accumulated from looting boxes. The game will be in a 1 versus 1 format; Devo VS Devo. Last Devo standing wins.

## Program Components

### Frontend:

- Serves pages with Tailwind, HTML, CSS, and JS
- Displays all in-game graphics and animations as directed by Node.js

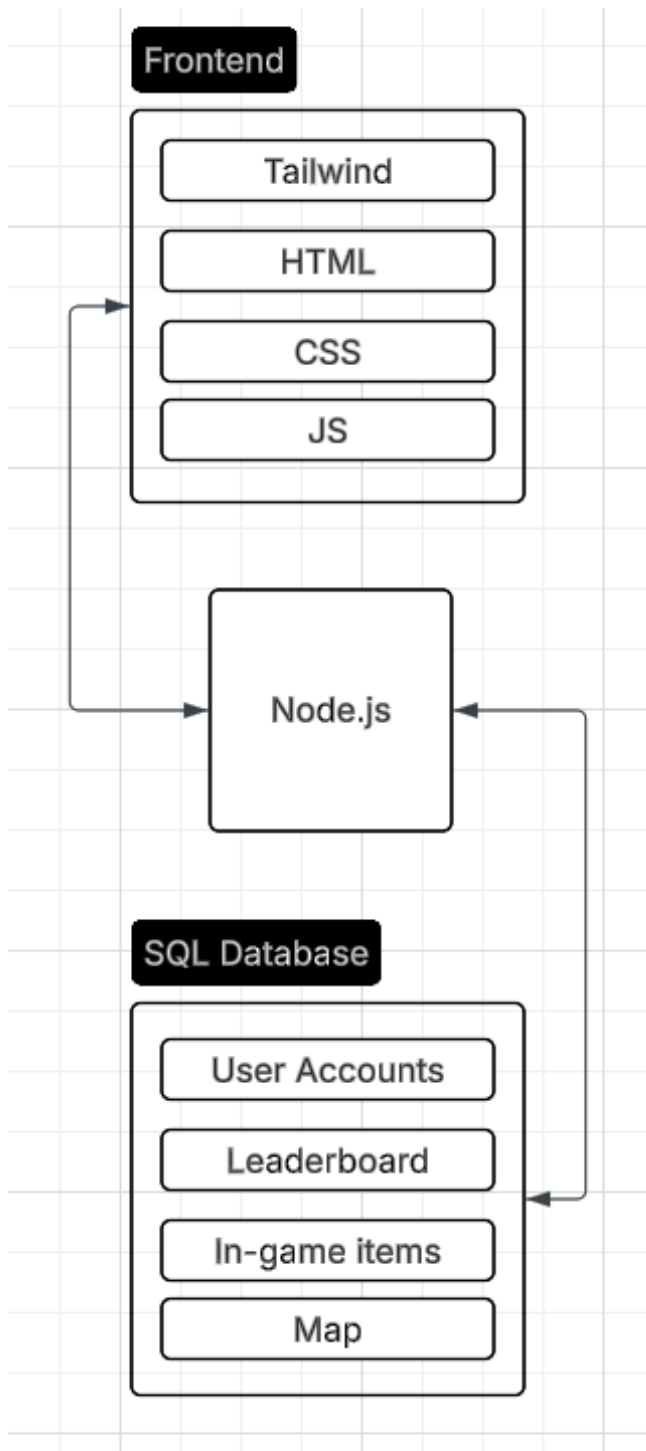
### Node.js:

- Serves as middleware server - brain behind the operations
- Handles asynchronous in-game logic as well as page routing
- Facilitates client to client communication
- Controls communication between backend and frontend

### SQL Database:

- Stores user account information
- Stores leaderboard information
- Stores game items information (healing items, guns, etc.)

# Component Relations + Visualization



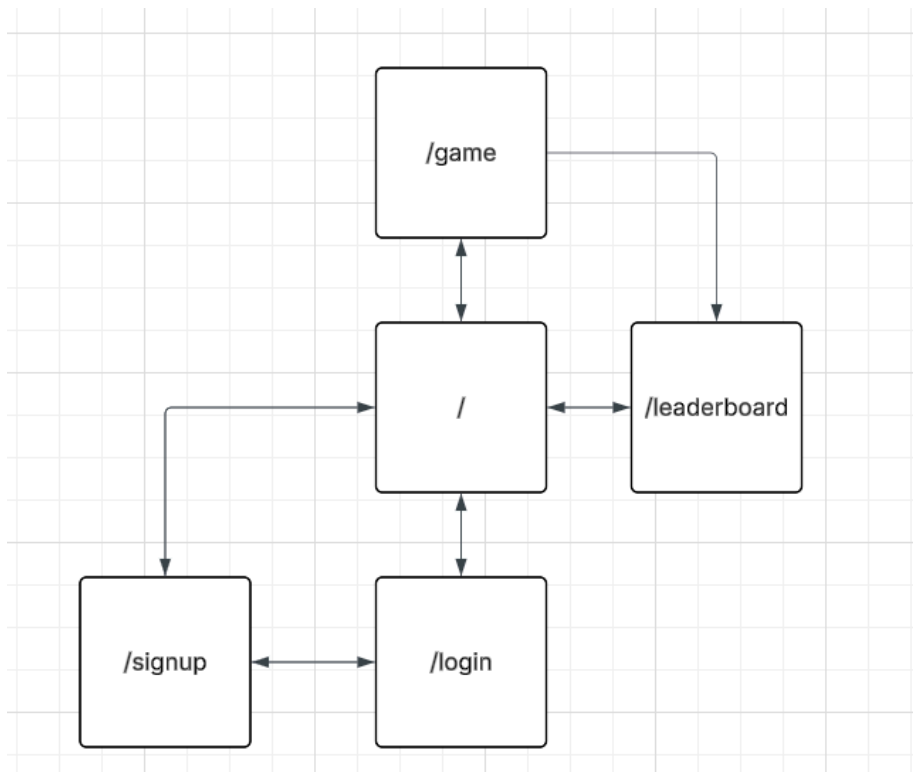
**Node.js** communicates with the **Frontend** in order to appropriately update graphics, ie:

- Health bar visually lowering upon taking damage
- Relaying other users' movements + actions to the screen

**Node.js** communicates with the **SQL Database** to both feed information to the **Frontend** and handle logical operations, ie:

- Database tells Node.js how much damage a user is to take for a certain gun
- Database tells Node.js how a gun is supposed to look, so that Node.js can tell the frontend.

## Site Map



## APIs

- We do not plan on using any APIs.

## Front-End Framework: Tailwind

- 1) Tailwind CSS's utility-first framework simplifies the web development process by reducing the need to write CSS or custom CSS.
- 2) Styles can be quickly applied to elements directly in the HTML code without needing to write custom CSS, saving significant development time.
- 3) Optimized and minimal CSS files are generated by Tailwind CSS, resulting in faster load times and improved overall website or application performance.

## Database Organization

### - User Table

Username (string)	Password (string)
Topher	*****
Mykolyk	*****
...	...

## Libraries/Packages/Modules

- `express` - A minimal web framework for Node.js that serves game client files and can handle routes like leaderboard data.
- `Express-sessions` - Storing session data on the client in a cookie. It works similar to sessions in Flask.
- `ws` - A WebSocket library that enables communication between the game server and connected players.
- `pixi.js` - Good for animation and rendering graphics/assets.

## Task Assignments

GENERAL TASKS	PMVC	DDP	DMM	DWN
Implement Database Interactions		X		

Construct Site-side Pages				X
GAME MANAGEMENT TASKS				
Game Map - Load initial obstacles and structures				X
Game Loop - Run core loop to update positions, physics, and game state (including storm function)	X		X	
Visual Experience - rendering graphics, animations, in-game UI		X		X
Input Handling - capture and process keyboard & mouse inputs and send to server			X	
Final Testing and Bug Fixing	X	X	X	X
ELO calculation and equations	X			X