



Team 208: Matthew Li, Manos Papanidis, Ethan Man, Ronald Nguyen

PROJECT OVERVIEW

Multiplayer Gaming, Fully Social

NetPlay is a web-based multiplayer gaming platform designed to bring players together with a rich social infrastructure. Over Sprints 1–3, the team implemented a comprehensive suite of social features: user profiles, a friends system, direct messaging, game invitations, teams, and both individual and global leaderboards. Every feature was engineered for real-time responsiveness. Players see friend activity, game invites, and chat messages the moment they happen, no refresh required.

FEATURES

Profile Customization

Profile for The Knight Of Games
Total Points: 0
[Add Friend](#)

- Username: user0
- Account created 2 months ago
- Team: No Team

Bio
I am the greatest of all time

Friends

Friends

[My Friends](#) Requests Sent **1** Find Friends

Filter by name... A–Z

Sénior Dos @user2 Last seen: 4 days ago [Invite to Game](#) [Remove](#)

Teams

Team: NetPlayers [Administrator](#)

Roster [Stats](#)

Team Stats

7 Wins	6 Losses	1.17 W/L Ratio	13 Games Played	0.54 Avg Score
--------	----------	----------------	-----------------	----------------

Member Contributions

Yao @user1	4W / 3L	7 games	4 pts
Sénior Dos @user2	3W / 3L	6 games	3 pts

Leaderboards

Global Leaderboard (all-time)

Global ▼ All Time ▼

[Individual](#) [Team](#)

RANK	USERNAME	TEAM	POINTS
1 ▲	Yáo (@user1)	NetPlayers	4
2	Sénior Dos (@user2)	NetPlayers	3
3	Frau Drei (@user3)	No Team	1
4	test2 (@test2)	No Team	1

Direct Messages

Direct Messages user2

[Start](#)

user2 I've invited you to a game!...

user2 user2 invited you to a nim!
[JOIN GAME](#) [DECLINE](#)
Apr 9, 2:52 PM

user2 How are you doing?
10:01 AM

user2 I'm good, how are you?
10:13 AM

user1 Great!!
10:33 AM

user1 Let's play
10:34 AM

TECH STACK + DESIGN DECISIONS

We extended the existing GameNite codebase, a TypeScript monorepo consisting of a React + Vite frontend, an Express + Node.js backend, and a shared types package used by both. New REST endpoints handle friend requests, DM conversations, team management, and leaderboard aggregation, while Socket.IO broadcasts real-time events: friendRequest, dmNewMessage, dmNotification, and leaderboardUpdated, to connected clients. Notification state is shared across components using React Context, avoiding prop drilling between the always-mounted layout and individual page components. Our CI pipeline runs TypeScript checks, ESLint, 86 Vitest unit tests, and 61 Playwright end-to-end tests on every pull request, with the application deployed to Render.com.

FOR THE FUTURE

Looking ahead, the most exciting opportunity is expanding the game library. With the full social infrastructure in place (friends, teams, leaderboards, and real-time invites), adding games with more strategic depth, such as a card game, chess, or backgammon, would give the competitive features far more meaning and better showcase the platform's potential. We would also prioritize implementing persistent authentication, as the current in-memory session is lost on page refresh, which is the most impactful usability improvement we could make. Also, a friend activity feed showing when friends finish a game, join a team, or climb the leaderboard would deepen social engagement and give users a reason to keep coming back to the platform. Lastly, we would love to look into implementing a mobile-friendly (iOS/Android) version of NetPlay in the future.