

CS4530 Final Project Poster

What we added

We felt that the existing GameNite platform did not have the appropriate features to keep users engaged and keep them coming back to the site. The site lacked a shared resource that users could connect with each other through.

We implemented 3 user stories that helped solve this problem and connect the GameNite users together. First, we implemented a **ranking system** that reflects a user's skill level for a specific game type along with an overall rank which are all displayed on a searchable leaderboard.

Next, we implemented a **friend system** that allows users to connect with friends, and make new ones in the GameNite community. The friend system allows users to easily keep track of their friends' stats/rankings along with the functionality to play private games with their friends.

Lastly, we implemented public **user statistics** so that users could track their own performance along with other users on the platform. These are displayed on users' profiles.

Demo & Source Code

Live Demo Site: <https://spring26-project-team-408-1.onrender.com/>

Source Code: <https://github.com/neu-cs4530/spring26-project-team-408-1>

Stats

| | | | |
|--|--|---|--|
| Overall record 6W – 6L – 0T 12 played | Overall win rate 50% across all games | Overall rank 1277 Across all games | Last 5 games 3W – 2L – 0T 60% recent win rate |
|--|--|---|--|



| GAME | RECORD | WIN % | RANK |
|-------|--------------|-------|-------------------------|
| nim | 3W – 3L – 0T | 50% | 1277 best |
| guess | 3W – 3L – 0T | 50% | 1277 worst |

Leaderboard

| # | Player | Rank |
|---|--------|------|
| 1 | Kate | 1350 |
| 2 | Liam | 1344 |
| 3 | Mia | 1338 |

Incoming Requests

Bob (Friends with user0, pending to user1 (@user1)) Accept Decline Block

Outgoing Requests

No outgoing requests.

Friends

Frank (Friends with user0, pending from user7 (@user7)) Remove Block

Grace (Friends with user1 (@user1)) Remove Block

Blocked Users

No blocked users.

Future Additions

In the future, it would be nice to implement a more robust user search feature to allow for users to more easily find people to connect with. As well, building out the ranking system to be more advanced in the calculation - also displaying different tiers of ranks to give users a progression goal. Lastly, to expand the user statistics we could give individual game statistics, allowing users to look into user's most recent games.

Technical Design Choices

Ranks/Leaderboard:

- Every completed game triggers logic that calculates a new rank for each user involved in that game. The rank is based on opponent rankings, games played, and existing rank. Each users rank data is stored in a new rank repo in our DB.

Friend System:

- Friendship relationships are stored in a new repo and are maintained by our backend friendship service. There are new API endpoints and the front end service displays all friend information. When friends, users can start private games, hidden from public view.

User Stats:

- Game results are stored in a new repo in the MongoDB database with a record for each user in each game. This allows us to have specific access to each game result for future game statistics additions. The results are pulled in a react hook and sorted/filtered by user/game-types.