# Leaderboard and Matchmaking Team

# Timeline

**High priority**

* Create leaderboard class and methods
  + Implement logic for retrieval of player rankings
  + Integration of leaderboard with Authentication and Profile
* Create method to sort top players
  + Implement sorting logic
    - Method to sort by rating points
    - Method to sort by win loss ratio
    - Method to sort by number of wins
* Integration of leaderboard with GUI
  + Leaderboard GUI creation
  + Leaderboard GUI menu options (e.g. filtering, etc.)
* Integration of leaderboard with Networking
  + Storing data in database
  + Fetching data in database
* Implement match result processing (e.g. Win/loss updates, rating updates, etc.)
* Allow player interaction through GUI
* Create matchmaking class and methods
  + Create queue data structure
  + Implement sorting algorithm on the queue data structure for matching opponents
  + Integration of Matchmaking class with Player (Display opponent information)
  + Deliver players into the correct game lobby
* Implement Timeout method and integration with GUI (Penalty for leaving in-progress games)

**Medium priority**

* Create method retrieve searched player
  + Handle cases where player is not found
* Create method to rank player’s friend list
* Add sorting order toggling functionality
  + Integrate with GUI
  + Integrate with Networking
* Unit testing

**Low priority**

* Display search results dynamically for leaderboard player search
* Ensure sorting efficiency of sorting logic
  + - Optimize method to sort by rating points
    - Optimize method to sort by win loss ratio
    - Optimize method to sort by number of wins
* Integrate network manager with leave match method

# Deadline logbook

[**Project**](https://d2l.ucalgary.ca/d2l/le/content/649707/viewContent/6914688/View) **deadlines**

Iteration 1: March 7th at 23:59

Iteration 2: March 21st at 23:59

Iteration 3: April 11th at 23:59

**Project group deadlines**

March 5th at 15:00: Complete diagrams corresponding to your development role and merge to main.

March 7th at 12:00: Review/critique parts of other teams.

March 7th at 21:00: Project iteration 1 submission. Leaves a 3 hour buffer in case something goes wrong.

**Team deadlines**

March 2nd: Complete use case descriptions

March 5th at 15:00: Complete use case diagrams and/or additional diagrams

March 5th at 16:30: Complete class structure diagrams

# Meeting logbook

February 28th at 16:00

Discuss workload distribution. Harry will work on leaderboard. Yash will work on matchmaking. Complete use case descriptions by Sunday (March 2nd) and plan to work on class structure diagrams together.

March 4th at 21:00

Discuss dispute resolution within team. Start working on class structure diagrams together. Push completed tasks to git. Plan to work on class structure diagrams together. Use time to complete dependencies.