



# Team 6 Sprint 1 Planning Document

Cole Compton, Matt Whipple, Noah Johnson, Morgan Kaehr, Julian Ng

## **Sprint Overview:**

The overall goals for this sprint are to set up our development environments, implement the backbone of the website, and start implementing the social features within the application. For the first goal, the main focus will be making sure everybody has up to date versions of the technologies we will be using. This includes making sure everyone has the proper Java version, everybody has React, and so forth. We will also establish a directory structure for our repository. The second goal will involve setting up our web server and getting some semblance of the website's structure established. The third goal will be where most of our development will take place this sprint. We plan to implement account creation and log-in, friend list creation and management, and chats.

**SCRUM Master:** Cole Compton

**Meeting schedule:** Mondays, Wednesdays, and Fridays @ 5pm

**Risks and challenges:**

The challenges with this sprint will mainly revolve around the technologies. Many of us are only familiar in one or two of the technologies that we've chosen to use, so we are anticipating the setup and use of the technologies during this sprint to be a major hurdle. Another challenge will be in setting up our web server to handle requests and respond with the appropriate web page. This is somewhat of a risk as well: as of now, we aren't planning to use a framework, but as we work on the project we may find that significant revisions to our planned backend may be required. If this occurs it could set back our development by a few days.

# Sprint Plan

User Story 1: As a user, I would like to be able to register and create an account.

- Task 1 - Setup User class in backend
  - Time - 6 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 2 - Setup Database to store user accounts
  - Time - 3 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 3 - Setup Authentication security for Users (Firebase?)
  - Time - 3 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 4 - Design and create signup page for users
  - Time - 3 hours
  - Task Owner(s) - Julian/Morgan
- Task 5 - Connect backend and frontend work
  - Time - 5 hours
  - Task Owner(s) - Everyone
- Task 6 - Unit test with test emails and passwords in different browsers
  - Time - 1 hour
  - Task Owner(s) - Julian
- Acceptance Criteria
  - Given that account creation is functional, when users register a username and password, then an account should be made and the user should be able to login.
  - Given that the database is correctly set up and is connected to our server, when a newly created account is made, then the account will be added to the database.
  - Given that the authentication is functional, when a user creates an account, then access to the user's account should be private and their password encrypted.

## User Story 2: As a user, I would like to be able to login and edit my account preferences.

- Task 1 - Create login methods as well as functionality that allows users to edit their information in backend
  - Time - 3 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 2 - Design and create login page and edit account page for users
  - Time - 5 hours
  - Task Owner(s) - Julian/Morgan
- Task 3 - Connect backend and frontend work
  - Time - 10 hours
  - Task Owner(s) - Everyone
- Task 4 - Unit test with registered and non-registered emails and passwords in different browsers
  - Time - 1 hour
  - Task Owner(s) - Julian
- Acceptance Criteria
  - Given a functional login screen, when a user attempts to login to their registered account with a valid password, a successful login will occur.
  - Given a functional user class, when a user login occurs, then acknowledgement that the user has logged in by the server will occur so that it can keep track of users that are currently logged in.
  - Given a functional login screen, when a user attempts to login to their registered account without a valid password, a failed login will occur and the user will be asked to login again.

## User Story 3: As a user, I would like to be able to create a list of “friends” that can be easily accessible.

- Task 1 - Implement functionality and methods for adding/removing friends and having a friend list
  - Time - 6 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 2 - Modify database to account for friend lists for users
  - Time - 2 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 3 - Design and create friends list visuals
  - Time - 4 hours
  - Task Owner(s) - Julian/Morgan
- Task 4 - Unit test adding and removing friends in different browsers, checking to make sure the friends list is updated accordingly
  - Time - 2 hours
  - Task Owner(s) - Julian/Cole
- Acceptance criteria
  - Given a functional friends system, when users click the “add friend” button a friend request will be sent.
  - Given that the friends system is working properly, when users click the “remove friend” button the friend will be removed from their friends list.
  - Given that the database is implemented correctly, when users log in to their account they will see a correct representation of their friends list.
  - Given a functional friends system, when users look at their friends list, then they will see any friends who are currently logged in.

## User Story 4: As a user, I would like to be able to chat with my friends outside of and during games.

*(Note: For this sprint, we will just implement the chat outside of games, since we aren't adding gameplay until Sprint 2)*

- Task 1 - Create classes for representing chats and messages
  - Time - 3 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 2 - Add chat interface to website
  - Time - 5 hours
  - Task Owner(s) - Julian/Morgan
- Task 3 - Establish communication between client and server for updating messages
  - Time - 15 hours (also applicable to communication of other data later)
  - Task Owner(s) - All
- Task 4 - Test chat feature with 2 or more people, checking edge cases such as empty strings
  - Time - 3 hours
  - Task Owner(s) - Morgan/Noah
- Acceptance Criteria
  - Given a functional in-lobby chat, when a user enters a message into the chat, then every user in the lobby should receive the message immediately in the chat box.
  - Given a functional in-lobby chat, when a user enters a message into the chat, then the chat box should automatically refresh without refreshing the entire page.
  - Given a functional global chat, when any user enters a message into the global chat box on the home page, then every logged in user should see the message if they look at the global chat box.
  - Given a functional personal message chat, when a user messages a friend, then the person they message should see the message pop up with a username of who its from and a reply line to message back.

User Story 5: As a user, I would like to be able to check my game statistics. *(Note: For this sprint, the game statistics won't be updated, but they will be part of the profile and the users will be able to see them. These statistics will be updated once users can play games, hopefully during Sprint 2.)*

- Task 1 - Add functionality for game statistics into user class
  - Time - 3 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 2 - Modify database to account for game statistics (when a new user is created, set all the relevant fields to 0)
  - Time - 3 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 3 - Design the profile page to include game statistics
  - Time - 3 hours
  - Task Owner(s) - Julian/Morgan
- Task 4 - Establish communication protocol between client and server for displaying game statistics, as well as between server and database for retrieving information from database.
  - Time - 15 hours (also applicable to communication of other data, as mentioned in User Story 4)
  - Task Owner(s) - All
- Task 5 - Test game statistics page (since statistics won't be updated after games yet, we will have to manually edit information in the database)
  - Time - 2 hours
  - Task Owner(s) - Matt
- Acceptance Criteria (for this sprint)
  - Given that the game statistics component is implemented correctly, when users click on another user's profile, then they will be able to see their (blank) game statistics.
  - Given that the game statistics component is implemented correctly, when users check their own profile, then they will be able to see their (blank) game statistics.
  - Given that the game statistics are implemented correctly, when users are checking the statistics of themselves or others, then they will see (blank) entries for wins, losses, average score, etc.

## User Story 6: As a user, I would like to be able to compare my skills to those of others via a ranking system.

- Task 1 - Add functionality for ranking (for now, average score) into user class
  - Time - 2 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 2 - Modify database to account for rankings (again average score, which for now will be 0 since we haven't implemented gameplay yet)
  - Time - 1 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 3 - Design the leaderboard page
  - Time - 3 hours
  - Task Owner(s) - Julian/Morgan
- Task 4 - Create algorithm for creating leaderboard
  - Time - 3 hours
  - Task Owner(s) - Cole/Matt/Noah
- Task 5 - Establish communication protocol between client and server for displaying leaderboard, as well as between server and database for retrieving information from database.
  - Time - 15 hours (also applicable to communication of other data, as mentioned in User Stories 4 and 5)
  - Task Owner(s) - All
- Task 6 - Test leaderboard page (since player statistics won't be updated after games yet, we will have to manually edit information in the database)
  - Time - 2 hours
  - Task Owner(s) - Cole
- Acceptance Criteria
  - Given that the leaderboard system is correctly implemented, when users navigate to it, then they will see a sorted list of all users by their average game score.
  - Given that the database interface is functional, when users are looking at the leaderboard, then they will see up to date and correct statistics for all the players.



- Given that the leaderboard component is implemented properly, when users click on the leaderboards link, then they will be directed to the leaderboards page.

## Remaining Product Backlog:

- ~~1. As a user, I would like to be able to register and create an account.~~
- ~~2. As a user, I would like to be able to login and edit my account preferences.~~
- ~~3. As a user, I would like to be able to create a list of "friends" that can be easily accessible.~~
- ~~4. As a user, I would like to be able to chat with my friends outside of and during games.~~
- ~~5. As a user, I would like to be able to check my game statistics.~~
- ~~6. As a user, I would like to be able to compare my skills to those of others via a ranking system.~~
7. As a user, I would like to be able to access all of the different features of Barbu Online in a smooth and efficient manner.
- ~~8. As a player, I would like to be able to have a record of how many games I've won or lost.~~
9. As a player, I would like to have the option of choosing to join a private game or join a public server game.
10. As a beginning player, I would like to be able to read or play a tutorial on how each Barbu game is played.
11. As an advanced player, I would like to be able to play competitively against other advanced players.
12. As a player, I would like to be able to pick the game type when it is my deal.
13. As a player, I would like to be able to double other players before a hand is played.
14. As a player, I would like to be able to see a visual list of the cards in my hand.
15. As a player, I would like to be able to easily select a card to play when it is my turn.
16. As a player, I would like to have the score automatically calculated at the end of each game.
17. As a player, I would like other players' plays to be quickly and clearly displayed during the gameplay.
18. As a player, if I attempt to make a play that is not allowed by the rules of the game, I would like to receive a message explaining why I can't make that play and telling me to pick a different card.
19. As a player, I would like the current game and doubles to be displayed for my reference during the gameplay.
20. (If time allows) As a player, I would like to be able to play a single player mode that challenges me to get better.