Design Document

for

NCAA Picks

Caleb Funk

Jack Legate

John Nguyen

Ryan Nguyen

Table of Contents

1.1 Executive Summary

Making predictions is an integral part of being a college football fan, whether it’s the result of the game, or how a team is going to perform on a given gameday. Unfortunately, the college football prediction landscape is predominantly influenced by gambling platforms, which often overshadow a fans opportunity to engage in non-monetary competition. NCAA Picks bridges this gap with a user-friendly predictions website that is focused on the social environment of a shared passion, college football.

Participants will earn points based on the accuracy of their predictions which will reflect on their position in the group leaderboard. What makes NCAA Picks unique is the level of customization it offers. Group administrators will be able to tailor their groups experience to their wants and needs through a customizable environment, including the ability to pick which teams to follow each week. Whether you're a casual fan or a college football superfan, the app is designed to be user-friendly and accessible, making it easy to dive in and enjoy the game.

Besides the fun predicting, NCAA Picks serves a bigger purpose, by promoting collaboration and friendly rivalry. Having a zero monetary risk environment is the key to the success of this application. This not only enriches the fan experience but also serves as a socially responsible project that encourages positive interactions among football enthusiasts. We are excited to create a safe social environment where fans can enjoy the sport we all love.

1.2 Background and Technical Requirements

Idea Space

One existing challenge within the sports prediction market is the large number of gambling-focused platforms. In regards to the problems this creates, many sports enthusiasts are often left without a safe space to enjoy friendly competition without monetary involvement. Our project is needed because it offers a web-based application where users can make weekly predictions on college football games without any inclusion of real money. This solves the problem of wanting to engage in sports predictions without the risks associated with gambling. Users who would benefit from this include sports fans, casual enthusiasts, friends, families, and communities seeking a fun and risk-free way to compete and connect over their love of the game.

Similar Ideas

- One existing platform within the current market is ESPN's College Pick'em. In regards to its features, it allows users to predict the outcomes of college football games and compete on leaderboards. One advantage of this platform is its integration with ESPN's vast sports resources, providing users with easy access to game statistics and expert analysis. But a disadvantage is that it doesn't offer the ability to form private groups with customizable settings, limiting social interaction among friends and family.
- Another similar platform is Yahoo Sports' Fantasy Football Pick'em. It offers features like public and private leagues, game predictions, and overall rankings. An advantage here is the option to create private leagues, but a disadvantage is the complexity of the interface, which can be overwhelming for casual users.
- A third existing platform is CBS Sports' College Football Pick'em. In regards to its features, one of the advantages that can be found with this platform is its straightforward interface, making it easy for users to participate without much hassle. But a disadvantage to this is the limited customization options for private groups, which might not satisfy users looking for a more tailored experience. Besides this, another feature that gives it an advantage is the availability of expert insights and analysis to inform user predictions. And one last disadvantage could be the minimal social features, as it lacks interactive components that encourage community engagement among users.

Our idea differs by focusing on simplicity and user experience. We provide an easy-to-use platform where users can form private groups, customize their experience by choosing specific teams to follow, and enjoy a social component through interactive features. Unlike existing platforms, our project emphasizes non-monetary competition and a user-friendly interface, making it a better option for those seeking casual yet engaging sports prediction activities without the risks associated with gambling.

Required Technology

Being that our application is a web based application there are a couple key software requirements that will be needed to ensure that it works properly. A database like MongoDB will be required to ensure that large amounts of data can be stored and retrieved in a timely manner. The database should be easy to access and work well with others as there is lots of data flowing through (player data, game data, group settings, social aspects, etc). Java Spring Boot will be at the core of the web application as the backend of the web application. Since Spring Boot is a microservice application meaning that it works in components. This adds lots of benefits for a web application with many parts such as a sports prediction app, since components are smaller and do not need every piece to function properly. Giving the users a simplistic yet full interaction, it makes the longevity extend as the parts are maintained by component and has good libraries for security and API calls.

With all the active and moving data, a front end such as Angular will be utilized to ensure that users get that competitive sports fantasy feel. Angular is easy to use and will provide lots of options for customizable aspects as well as functionally profound components. With Java Spring Boot being the back end branching off the Spring framework, there will not be a need to implement any technologies as the Spring framework has all the libraries and functions needed to achieve the sports prediction app.

Deployment Plans:

- Possible Future utilization of AWS

Assets and Engines

As previously mentioned in the section above, lots of the technology that is needed to complete this project already exists and has documentation that help guide its process. Being that most if not all of the project will be in Java Spring Boot there may be bits and pieces that the team will have to come together to implement and create, but Spring Boot will handle the API calls as well as some of the security sign in that will be required for users to log in. This resource as combined with Angular and Mongo will be at the forefront of the operation, and these resources can be accessed online.

Software/Hardware Requirements

User

Users will only need a web browser to access the NCAA Picks platform. Any modern browser such as Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge will work seamlessly. The app will be fully responsive, so users can enjoy it on desktops, tablets, or smartphones.

Server Setup

In addition to the frontend and backend services talked about above, we will need some software to manage the server.

- Apache HTTP Server: We've chosen Apache to host our Angular frontend because it's a well-established, reliable option. Apache makes it easy to serve static files (like the ones generated by Angular), and it has great features for forwarding requests from users to our backend services. Plus, it comes with built-in support for secure connections (SSL/TLS), which will help protect user data and keep everything running smoothly.
- Docker (Optional): We're considering using Docker to make deployment easier. Docker lets us bundle everything—backend, frontend, and database—into containers, which are like lightweight packages that can run anywhere. This will ensure that no matter where we deploy the app, it works the same way, simplifying management and reducing the risk of configuration issues.

1.3 Requirement Analysis

System Architecture

Frontend (Angular)

The Frontend (Angular) component serves as an integral part of the user experience for the NCAA Picks web application. Built with the Angular framework, the frontend is designed to be intuitive, responsive, and engaging. The component aims to ensure that users ranging from casual fans to dedicated college football enthusiasts can seamlessly navigate and interact with the application. The frontend will be divided into five portions: Login/Sign Up, Home, In-Group, Group Customizations, and Profiles.

- Login/Sign Up: Interface for new users to create accounts and existing users to log in.
- Home: The central dashboard where users can view the global leaderboard, view their current groups, create or join groups, and navigate to other key features of the platform.
- In-Group: Space for users within a specific group to make weekly game predictions, view upcoming games, view their group leaderboards, and access the customization screen.
- Group Customizations: An interface that enables group administrators to tailor group settings, select teams to follow, define prediction categories, and manage members.
- Profiles: Place where users can customize their profile and comment on other profiles.

Backend (Java Spring Boot)

The backend of our system will utilize Rest communication to ensure single purpose CRUD operations to keep the structure simple and easy to use. This connects both the frontend and the data holding applications (database/External API's) that we will be using. The backend will be divided into four portions: User Authentication, Database Management, Data Retention, and Score Calculations.

- User Authentication: Authenticate user login/signup as well as store user data securely.
- Database Management: Maintain the mongoDB database with user information, group information, and football information.
- Data Retention: Every once in a while, get the data/stats about teams and games and update the database accordingly.
- Score Calculations: Calculate the scores for each player every week and update the database with those scores based on the results of the football week.

Database (MongoDB)

MongoDB is ideal for web applications due to its flexible schema, allowing for dynamic data structures that adapt to changing requirements. Its ability to scale horizontally through sharding ensures high performance even as data volumes grow, while the JSON-like document format facilitates easy integration with JavaScript and front-end technologies.

- Flexible Schema Design: MongoDB's schema-less structure allows for easy adaptation to evolving application requirements, making it ideal for a growing prediction platform.
- High Scalability: MongoDB supports horizontal scaling, ensuring the database can handle large volumes of data efficiently as the application and user base grow.
- Structure: MongoDB's document format aligns well with JavaScript, facilitating seamless integration with the Angular frontend and supporting intuitive data storage and retrieval.
- Real-Time Data Management: MongoDB's efficient data handling enables fast updates, crucial for managing up-to-the-minute football data, predictions, and leaderboard results.

Server (Apache HTTP)

The Apache HTTP server will host the Angular frontend, managing the traffic to the site, and forwarding all requests to the backend. It will serve static Angular files and ensure secure communication between the frontend and backend. By handling incoming requests and securely directing them to the backend, this component plays a crucial role in maintaining a seamless, reliable, and secure experience for users. The server will handle several key functions:

- Serve Angular Application: Apache will host the files generated by the Angular frontend, ensuring quick access and responsiveness for users as they navigate the platform.
- Secure Communication: To protect user data, Apache will enable SSL/TLS, providing secure transmission between the frontend and backend components.
- Load Management and Request Forwarding: Apache will facilitate load balancing and enhance performance and reliability during high traffic.

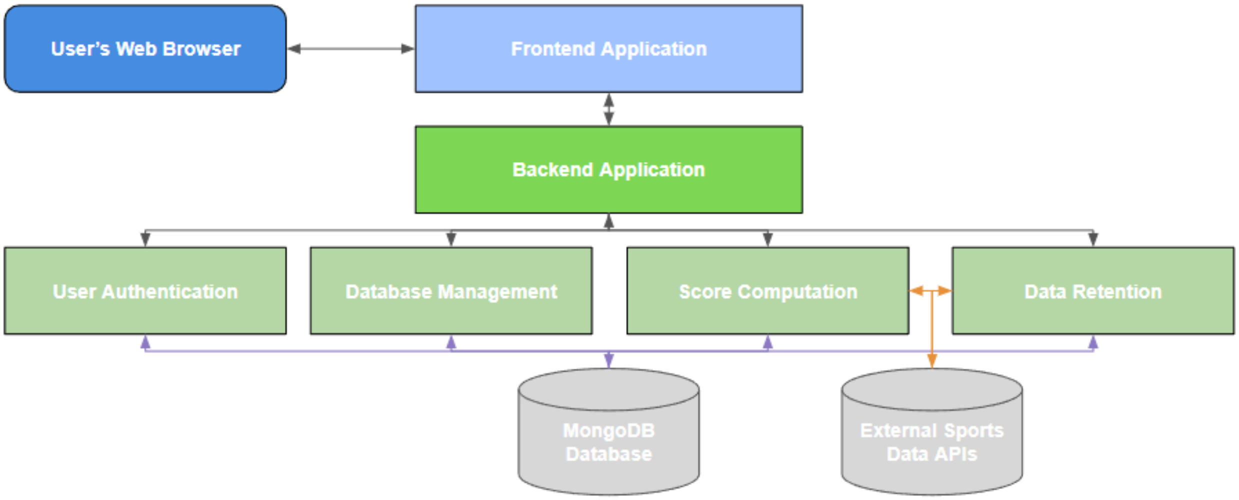


Figure 1: Project Overview Diagram

Personnel

Frontend		Backend	
Caleb Funk	**Ryan Ngueyn**	**Jack Legate**	**John Nguyen**
• In-Group Page • Login/Signin Page • Settings Page • Confirm Join a Group Page	• Home Page • Create Group Page • Profile Page • Header/Footer • Error Page	• Backend endpoints • Rest API Integration • Integration Testing • External API data collection	• Set up Mongo DB • Singular Group Endpoint • Get Teams Endpoint • Predictions Endpoint

- Ryan Nguyen: Primary responsibility will be the development of the NCAA Picks frontend, ensuring seamless integration with the backend and delivering a user-friendly application. His versatility allows him to support other aspects of the project as needed.
  - Develop and Maintain the Angular Frontend Application
  - Implement UI/UX Designs
  - Collaborate with Backend Developers to Connect Frontend Elements

- John Nguyen: Will work on the backend component of the project Java Spring Boot will be at the forefront of the operation. John's fullstack internship has offered him the necessary skills in backend programming, as well as front end skills when necessary.
  - Collaborate with jack to create RestAPI Endpoints
  - Create sql script to generate data in the database
  - Utilize classes and models to organize data

- Jack Legate: Is responsible for the communication and project infrastructure, due to his experience in RestAPI's. Jack will also be a helping hand for everyone else if they need due to his experience in the tech stack that we are using.
  - Develop/Test Backend/API Endpoints
  - Maintain System Infrastructure (MVC)
  - Collaborate with Database Developer

- Caleb Funk: Will help lead the design and implementation of the project's user interface. He is responsible for designing a responsive UI, developing API endpoints for efficient frontend-backend communication, and managing the web server.
  - Facilitate communication of data to the frontend
  - Develop the Angular frontend
  - Collaborate with backend developers in running the server

System Features

Rank 1: Infrastructure and Core Functionality

- *1.1: Core database interaction*
  - Game Data: Fetch game schedules and results.
  - Automatic Updates: Update leaderboards and user points based on scores.
- *1.2: User Login/Sign up*
  - Login Functionality: Users can log in with their credentials.
  - Sign Up Functionality: Users can register to receive an account and profile.
- *1.3: Group Creation*
  - Create Groups: Users can create private groups with custom names.
  - Invite Members: Generate unique codes or links to invite others.
- *1.4: Weekly Predictions*
  - Game Schedule Display: Show upcoming games for selected teams.
  - Prediction Interface: Users can make predictions on game outcomes.
- *1.5: Group Leaderboard*
  - Leaderboard: Display rankings within each group based on points.
  - Weekly and Total Rankings: Show standings for each week and cumulatively.

Rank 2: Planned Features

- *2.1 Group Followed Teams*
  - FBS Teams: Groups choose custom sets of teams to follow throughout the season.
- *2.2 Group Prediction Options*
  - Choose Predictions: Groups can customize the predictions they make each week.
  - Predictions: Outcome, score, over/under or more niche options like weather.
- *2.3 Stat Screen*
  - Stat Retrieval: Utilize a RestAPI that will return specific game stats.
  - Implementing on top of a SQL query that will partition the data as needed.
- *2.4 Global Leaderboard*
  - Seasons Played: Highlight users with the most participation over seasons.
  - Search Functionality: Users can search for other users by username.
- *2.5 Point Distribution*
  - Point Equation: Use an equation for each of the prediction options to calculate the points earned for every accurate choice.
- *2.6 Profile Messages*
  - Direct Message: Users can search for others and directly message them.

Rank 3: Advanced Individual Features

- *3.1 Profile customization*
  - Backgrounds Colors: Users can customize the look of their profiles.
  - User Icons: Users can customize what icon they want for their account icon.
  - About Me: Users can use this section to talk about themselves and their interests.
- *3.2 Expanded Social Interactions*
  - Friend System: Add friends and see their status.
- *3.3 Statistic Visualizers (Graphs)*
  - Personal Stats: Track personal prediction accuracy over time.
  - Group Stats: View overall group performance and trends.

1.4 Software Engineering Tools and Techniques

1. Agile vs. Traditional

We have chosen an Agile approach to ensure flexible and iterative development. We have defined high-level goals to meet early on, which allows us to continually assess and adapt based on feedback and progress. While we each have gravitated towards individual tasks and roles, we aim to have each sprint last 2 weeks in order to break down the workflow. Agile allows us to deliver value incrementally and be responsive to change, which makes it well-suited for this collaborative, user-centered project.

2. Tools

2.1 IDEs

- **IntelliJ IDEA:** Provides a user friendly architecture with integrated tools and provides it as the ultimate tool for a web application.

2.2 Main Libraries

- **Angular**: A front-end web framework for building dynamic and responsive applications with a component-based architecture.
- **JDK**: A software toolkit that provides tools, libraries, and a runtime environment for developing and running Java applications.
- **Maven**: A project management and build tool for Java applications that simplifies dependency management, builds, and testing processes.

2.3 Postman

We will use postman to test out any endpoints that are created through the backend. This tool will be used to guarantee that the backend endpoints are working like expected on the network.

2.4 Testing Plans and Tools

- **Backend:** Endpoints will be tested with integration tests with JUnit testing to test the expected input and output. Postman will be used to test if the endpoints are going to be able to be reached on the network.
- **Frontend:** The frontend will be tested manually by the developers as well as our expected user base. We will be testing the web applications by first using it as intended, then we will try to break it. We want to ask users to test it because they may see something that we may not see.

2.5 Bug Tracking

- **Gitlab:** Use the issue board to track bug occurrences by adding it as an “Incident” so that we can remember to address it.

2.6 Versioning

- **GitLab:** Provides a built in github styled interface where the developers are able to see commits, branches, merge requests and more.

2.7 Issue Tracking

- **Gitlab:** Issue board similar to a Jira styled issue board that allows developers to assign themselves, give time estimates, label tasks, and to organize issues to get completed

3. Code Reviews

We have elected to impose single signoff code reviews as our team will generally be split with half focusing more on frontend development and the other half focusing on backend development. This ensures that work is still properly reviewed by the right set of eyes and that it doesn’t get locked behind multiple developers trying to fit a review into their schedule.

4. Documentation

- **Global Documentation:** (GitHub Wiki) A centralized repository for project-wide documentation, including architectural diagrams, design decisions, and meeting notes.
- **Local Documentation:** (Code Comments) Developers will follow the "comment as you code" principle, ensuring that code is accompanied by meaningful comments for clarity.

5. Team Communication

- **Discord:** Primary tool for instant messaging, quick discussions, and real-time collaboration.
- **GitHub:** For code-related discussions, issue tracking, and project management.

6. Team Meetings

- **Weekly Meetings:** Scheduled conferences (using Discord) every week to discuss progress, plan sprints, and address any challenges.

- **Ad-Hoc Meetings:** Additional meetings may be scheduled as needed for pair programming, design discussions, or urgent issues.

1.5 Timeline

Alpha

Alpha Phase	Caleb Funk	Jack Legate	John Nguyen	Ryan Nguyen
Week One	Setup user sign in and account creation page	Get endpoints for all stats needed for display	Create Profile endpoint to return profiles	UI and Queries for Profile Messages
Week Two	Coordinate with backend to store user account information	Setup single point external api data grabbing	Set up DB to contain info per profile	Profile Customization
Week Three	Integrate user accounts with Ryan's work on profiles	Create an admin board to change time etc.(mock system)	Create a model that represents a profile and list of connected profiles	Profile Users Implementation
Week Four	Allow users to join and create different groups	Prioritize help towards finishing backend for first deployment	Ensure all endpoints are created	Integrate Completed UIs with Backend APIs

Beta

Beta Phase	Caleb Funk	Jack Legate	John Nguyen	Ryan Nguyen
Week One	Display dynamic stats about each matchup	User testing, giving the product to users	Endpoint to retrieve home data	UI and Queries for Home
Week Two	Users should be able to make predictions that save to the database	Add Different Prediction Options	Update Group endpoint to handle multiple groups with users	UI and Queries for Create Group
Week Three	Prediction options odds need to be pulled from	Add equation balancing of predictions	Update/create users endpoints to handle different tasks	Home / Create Group Users Implementation

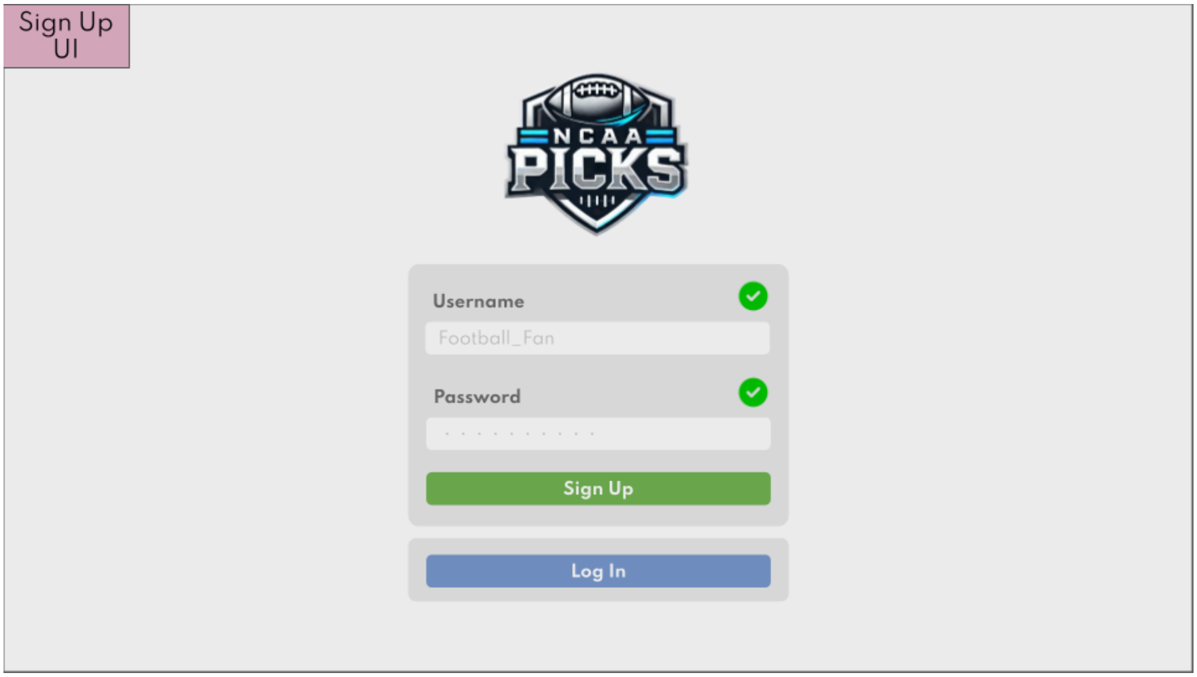
	backend or calculated			
Week Four	Users need to be awarded points for their predictions	Live System of week to week data	Ensure groups, users, and profile endpoints are complete	Integrate Completed UIs with Backend APIs

Final

Final Phase	Caleb Funk	Jack Legate	John Nguyen	Ryan Nguyen
Week One	Setup settings page to see and edit account information	Give Bells and Whistles endpoints to frontend	Connect backed to server and have it online	Check All Pages Function Accordingly and Bug Fix
Week Two	Finalize UI enhancements and ensure responsiveness	Ensure that server stays active unmonitored	Ensure that server stays active unmonitored	Finalize UI Enhancements and Ensure Responsiveness
Week Three	Complete UI and frontend test suites	Prepare the product for demo and showcases	Have Users create accounts and select teams to ensure DB has the correct data	User Acceptance Testing and Presentation Preparations
Week Four	Final bug fixes and Presentation Preparations	Final bug fixes and Presentation Preparations	Final bug fixes and Presentation Preparations	Final Bug Fixes and Presentation Preparations

1.6 Appendix A: UI Sketches

UI Sketch One

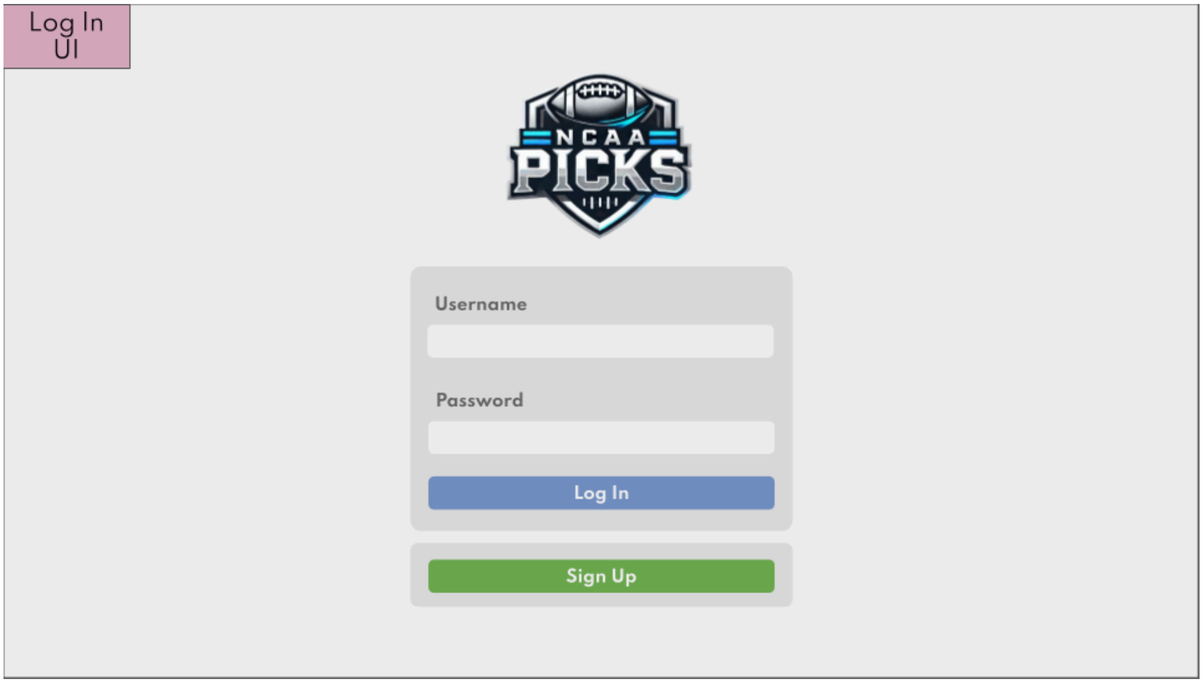


Username

Football_Fan

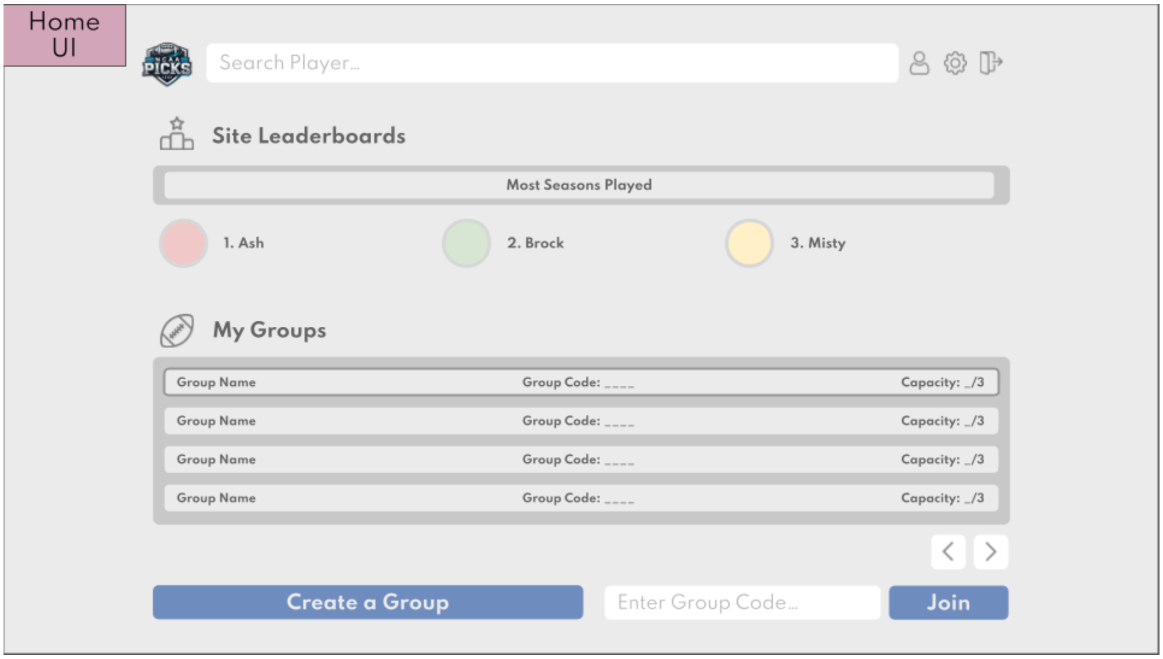
Screen that user would see when they are creating an account for our site

UI Sketch Two



Screen that user would first see on the site, prompting them to login to an existing account.

UI Sketch Three

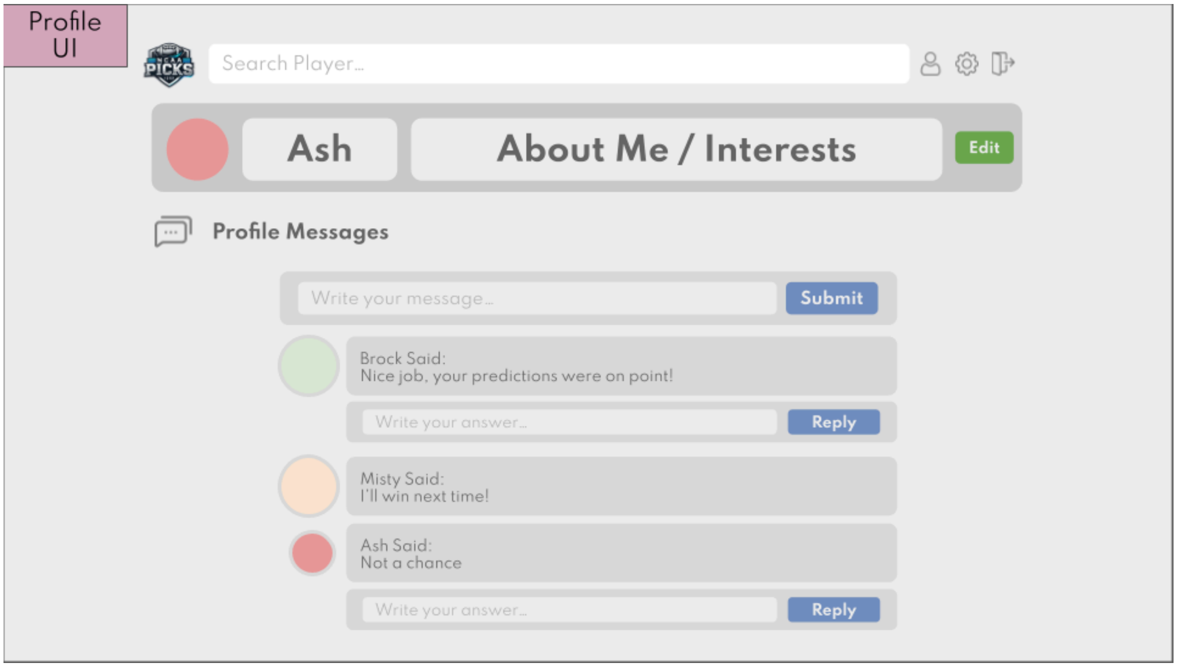


1. Ash
2. Brock
3. Misty

Group Name	Group Code: ____	Capacity: _/3
Group Name	Group Code: ____	Capacity: _/3
Group Name	Group Code: ____	Capacity: _/3
Group Name	Group Code: ____	Capacity: _/3

Home page and first thing the user sees after logging in. Main place to view and manage all groups the user is in, and see some of the site's top users.

UI Sketch Four



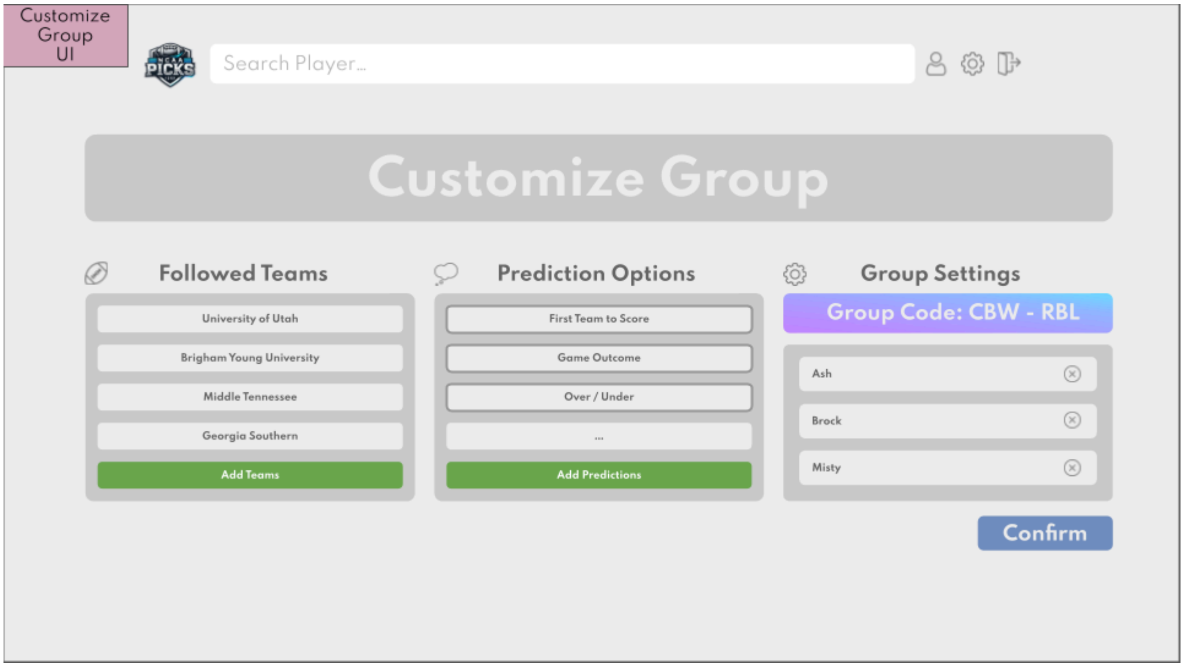
Brock Said:
Nice job, your predictions were on point!

Misty Said:
I'll win next time!

Ash Said:
Not a chance

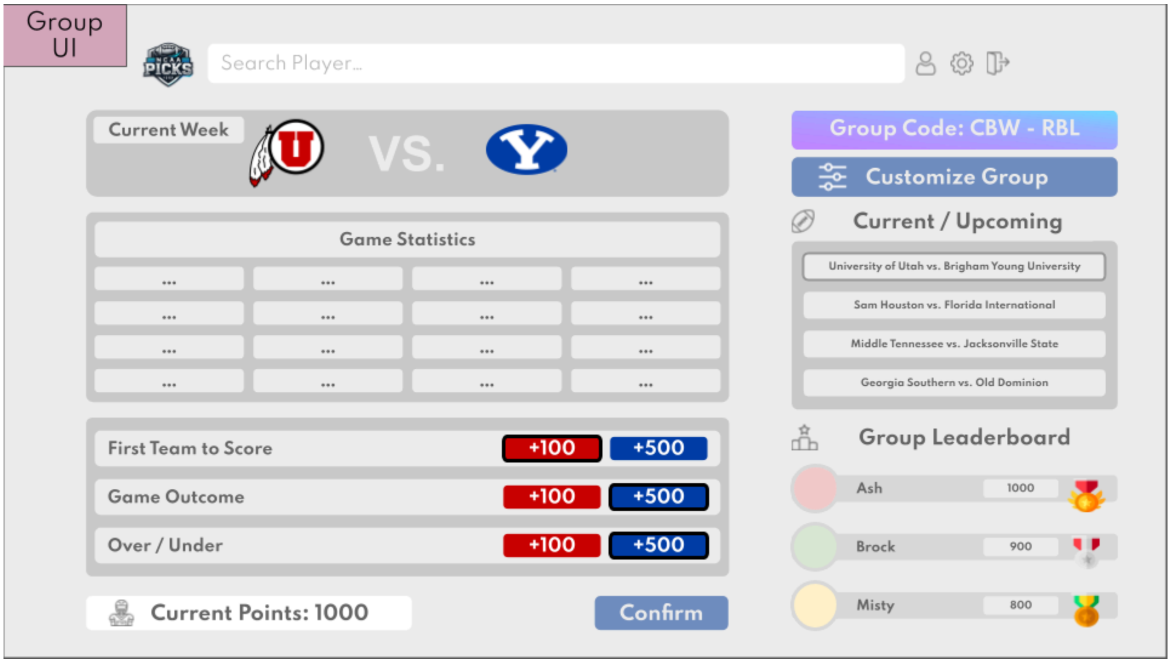
Profile UI screen so users can update information about themselves and see any messages they have received from other users.

UI Sketch Five



Screen for an eligible user to manage group settings including all the teams they follow, the predictions they make, and all members of the group.

UI Sketch Six



Primary screen for users to interact with to make their predictions and see group leaderboard.

1.7 Appendix B: Use Cases

Use Case One

Number	**Use Case 1**
Title	**Create an Account / Sign Up**
Preparer	Jack Legate
Actor/User	New User
User Story	As an aspiring new user, I want to create a new account so that I can participate in NCAA Picks with my friends.
Course of Events	1. Open the NCAA Picks website. 2. Press the ‘sign up’ button on the non-user home page. 3. Get rerouted to a form for the necessary account creation information. 4. Fill out information, as well as a password that has certain rules to it. 5. Press the 'Create Account’ button. 6. Get rerouted to the user home page as the new user.
Exceptions/ Alternates	● ‘Login’ button right next to ‘signup’ button if user already has an existing account; see use case 2
Related UI	**See Sketch One**

Use Case Two

Number	**Use Case 2**
Title	**Login to an Existing Account**
Preparer	Jack Legate
Actor/User	Existing User
User Story	As an existing user, I want to login with my confidential NCAA Picks account credentials so that I can participate in NCAA Picks with my friends.
Course of Events	1. Open the NCAA Picks website. 2. Press the ‘login’ button on the non-user home page. 3. Get rerouted to a form for entering in login credentials. 4. Fill out login credentials. 5. Press the 'login’ button. 6. Get rerouted to the user home page as the logged in user.
Exceptions/ Alternates	● ‘Sign up’ button right next to ‘login’ button if the user is new to the website; see use case 1 ● Have a forgot credentials button to help user remember their credentials with an email or sms link

Related UI	**See Sketch Two**

Use Case Three

Number	**Use Case 3**
Title	**Create a group**
Preparer	Jack Legate
Actor/User	Group Administrator
User Story	As a member of a friend group, I can create a NCAA Picks group for my friends to join with the ability to customize my groups experience, so that my friend group can have a unique experience with NCAA Picks
Course of Events	1. Before the season starts, an aspiring group admin presses the create group button. 2. Users are rerouted to the group creation page where they can write out a description, group name and custom pick options. 3. Custom pick options will be the weekly decisions each team member will make. 4. After the group is created, a copyable link will be present for the group admin to distribute around.
Exceptions/ Alternates	● None
Related UI	**See Sketch Five**

Use Case Four

Number	**Use Case 4**
Title	**Join an existing group**
Preparer	Caleb Funk
Actor/User	Registered User
User Story	As a registered user I can join an already existing group using a unique code given to me by the administrator or somebody within the group.
Course of Events	1. Open the NCAA Picks website. 2. Login to your account. 3. Request group code from a member of the group you want to join. 4. Enter code into the input box. 5. Click ‘Join’.
Exceptions/ Alternates	● None

Related UI	**See Sketch Three**

Use Case Five

Number	**Use Case 5**
Title	**Make Weekly Game Predictions**
Preparer	Caleb Funk
Actor/User	Group Member/Playing User
User Story	As a member of a group, each week I can make predictions on the outcome of the games for the teams that my group follows.
Course of Events	1. Open the NCAA Picks website. 2. Login to your account. 3. Click on your desired group. 4. For each matchup, analyze statistics to make your choices in each category.
Exceptions/ Alternates	● None
Related UI	**See Sketch Six**

Use Case Six

Number	**Use Case 6**
Title	**Customize a Group**
Preparer	Caleb Funk
Actor/User	Group Administrator
User Story	As the administrator of a group I can add teams to follow as well as change the predictions that we have to make for each matchup.
Course of Events	1. Open the NCAA Picks website. 2. Login to your account. 3. Click on your desired group. 4. Click ‘Customize Group’. 5. Click ‘Add Teams’ to search out and add teams to your list. 6. OR Click ‘Add Predictions’ to change the selection of active predictions to make on each matchup.
Exceptions/ Alternates	● None
Related UI	**See Sketch Five**

Use Case Seven

Number	**Use Case 7**
Title	**View Group LeaderBoard**
Preparer	John Nguyen
Actor/User	User within a group
User Story	After an exciting or disappointing week of college football, I can't wait to see how accurate I was with my predictions for the week.
Course of Events	1. Open the NCAA Picks website. 2. Press the sign in button to login. 3. Navigate to the group that you want to check the scores for. 4. Click on "Weekly Outcome" to display results.
Exceptions/ Alternates	● This may be expand where the the results could be displayed on their own page to make it simpler or to have more data returned
Related UI	**See Sketch Six**

Use Case Eight

Number	**Use Case 8**
Title	**Search User Profile**
Preparer	John Nguyen
Actor/User	Active User
User Story	Like the coach of the program you may want to scout other players to check their stats and maybe get some insight into what picks they are choosing.
Course of Events	1. Open the NCAA Picks website. 2. Press the sign in button to login. 3. A user can either find a user through the search player feature found above, or the user can select their own profile. 4. Then click on the user that they want to view.
Exceptions/ Alternates	● In the future we may want to expand this feature into its own page so that the user can see mutuals to add people that they may know
Related UI	**See Sketch Three/Four/Five/Six**

Use Case Nine

Number	**Use Case 9**
Title	**View Global Leaderboard (Most Seasons Played)**
Preparer	Ryan Nguyen
Actor/User	Existing User
User Story	As an existing user, I want to see other possible users to socialize with and also be able to see highlighted players who have the most seasons played.
Course of Events	1. Open the NCAA Picks website 2. Fill out login credentials. 3. Press the 'login' button. 4. Get rerouted to the user home page as the logged in user. 5. Global leaderboard is viewable and displays players who have played the most seasons
Exceptions/ Alternates	● If no games have been played yet, display a message indicating that rankings will be available after the first set of games.
Related UI	**See Sketch Three**

Use Case Ten

Number	**Use Case 10**
Title	**Profile Customization**
Preparer	Ryan Nguyen
Actor/User	Existing User
User Story	As an existing user, I want to be able to customize my account in order to be more unique and express myself via my profile.
Course of Events	1. Open the NCAA Picks website 2. Fill out login credentials. 3. Press the 'login' button. 4. Get rerouted to the user home page as the logged in user. 5. Select the profile button to be rerouted to a designated profile page. 6. Click an 'edit' button and be able to customize profile picture, background, and about me section.
Exceptions/ Alternates	● If the user attempts to navigate away without saving changes, display a warning to prevent loss of data.
Related UI	**See Sketch Four**

Use Case Eleven

Number	**Use Case 11**
Title	**Chat and Leave Messages on Other Users' Profiles**
Preparer	Ryan Nguyen
Actor/User	Existing User
User Story	As an existing user, I want to socialize with other users by having the option to leave a message on their profile and be able to leave/get responses.
Course of Events	1. Open the NCAA Picks website 2. Fill out login credentials. 3. Press the 'login' button. 4. Get rerouted to the user home page as the logged in user. 5. Select the profile button to be rerouted to a designated profile page or go to another user's profile page. 6. Click the 'write your message' or 'write your answer' textbox in order to leave a message or reply to one. 7. Click the 'submit' or 'reply' button to confirm the message you wrote.
Exceptions/ Alternates	● If the user has set their profile to disallow messages, display a notice like "This user is not accepting messages." ● Users can delete messages they've posted.
Related UI	**See Sketch Four**

Use Case Twelve

Number	**Use Case 12**
Title	**Invite Friends/Family/Strangers**
Preparer	John Nguyen
Actor/User	Active User
User Story	Making predictions on any college football brings the excitement to another level, but nothing beats the satisfaction of beating of your friends and being right
Course of Events	1. Open the NCAA Picks website. 2. Press the sign in button to log in. 3. Join a group or request a friend/family to give you a group code to join.
Exceptions/ Alternates	● It might be beneficial to have this search bar feature be present on every page so that the user can search for a user anytime
Related UI	**See Sketch Five/Six**

Use Case Thirteen

Number	**Use Case 13**
Title	**View Weekly Outcome**
Preparer	John Nguyen
Actor/User	Active User
User Story	After placing all of the predictions that the user chooses, they can then login and check the group leaderboard to see their current points and place.
Course of Events	1. Open the NCAA Picks website. 2. Press the sign in button to log in. 3. Click the group that they want to check. 4. There will be the results posted in the group leaderboard.
Exceptions/ Alternates	● Considering the large amount of data that is represented, it may be better to have the results be on a different page, or even allow the user to choose what results they want returned.
Related UI	**See Sketch Six**

1.8 Revisions Appendix

General

- Title numbering
- Making font all the same throughout document

Executive Summary/Introduction/Abstract

No changes

Background and Technical Requirements

- Slight paragraph adjustment
- Possible deployment plan listed

Requirements Analysis

- Refine Project Framework Diagram
- Condensed system architecture and personnel sections to fit page requirements
- System features numbered appropriately (1.1, 2.1, etc.)

Software Engineering Tools and Techniques

No changes

Timeline

No changes

UI / Use Case Appendix

- Text/Captions added for each UI sketch