**Software Requirements and Design Document**

**For**

**Group <8>**

Version 2.0

**Authors**:

Megan Achurra

Dylan Giesler

Daniel Lukish

Miguel A. Perez Ojito

Thomas Rudolph

**Overview**

Django’s Fantasy is a fantasy soccer site focused on attracting newer players to the world of fantasy sports. We plan to offer exhilarating features such as: drafts, leagues, and a point system, but we will also lower the barrier to entry and soften the learning curve to help make things less daunting for new players. Tool tips and suggestions should help new players feel a little more informed, and comfortable when choosing a team or checking up on their progress. Knowing which data points are and are not relevant to a newcomer can be one of the most difficult tasks, so we plan to make a team information system that will help a person choose a team that is right for them by providing statistics along with context for those statistics. We will also be using a simplified point system where if your team wins a match you gain a point. That’s it, no frills. Just a simple, easy to use system that can help get your novice friends right into the action with more experienced players.

**Functional Requirements**

1. Create Leagues – High Priority

Creates the league owner based on who creates the league. Allows the invitation of other users to the league.

2. Invite To League – High Priority

League owners can invite any other user to join their league.

3. Drafts – High Priority

League owners can begin a draft at any point after they have filled their league with users. Then they can begin a draft where users in the league choose teams in a snake draft order.

4. Point System – High Priority

The metric by which users are compared to each other. You gain a point when your team wins or when you correctly guess which team would lose.

5. Login – High Priority

Allows access to the site after signing up. Required to get anything useful from the site.

6. Sign Up – High Priority

Allows the users to create an account, which will allow them to login with the same credentials in the future.

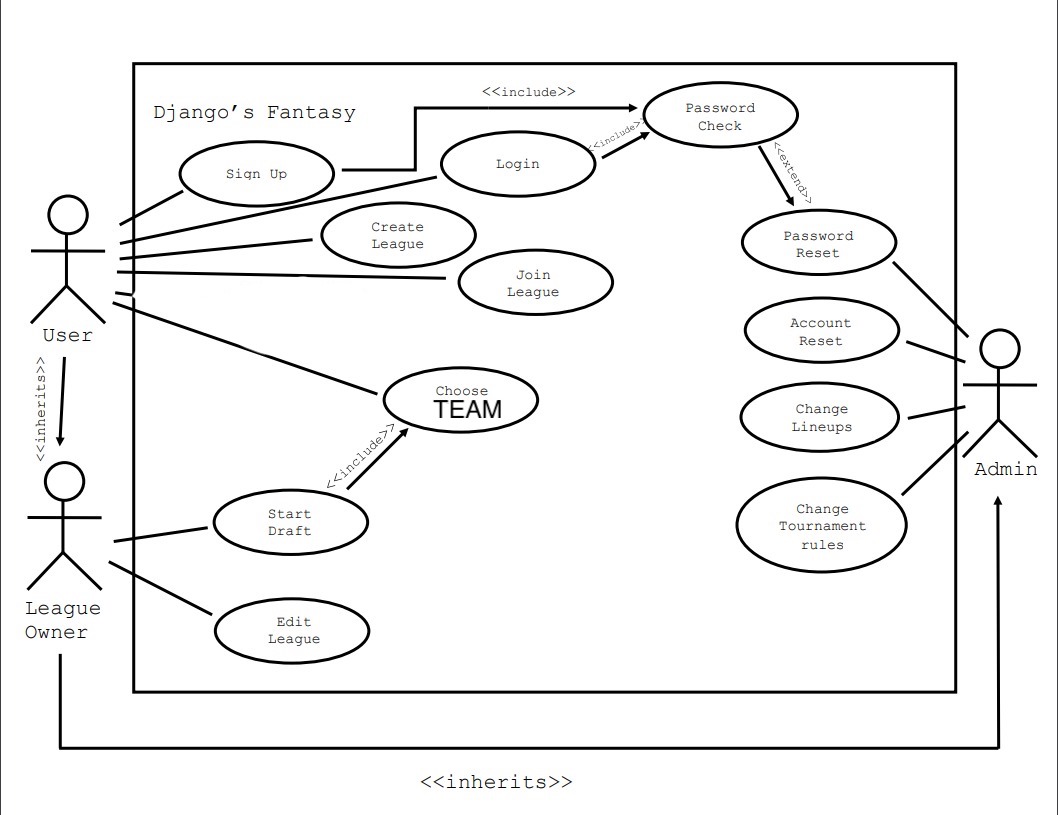
7. Team information system - Medium Priority

Allows the user to easily see a page full of statistics and explanations about all the teams in the premier league.

**Non-functional Requirements**

1. Cross Browser Compatibility: We want to ensure that all of our end users have the same experience no matter what browser their heart lead them to pick. To achieve this all that we needed to do was:
   1. Just load the site on a handful of browsers: Firefox, Chrome, Safari, and Edge
   2. use HTML features that are supported across all browsers, this was easy to do by simply looking at the compatibility of the HTML feature in w3schools
2. Mindful of Users with Accessibility Needs: Just tried our best to follow the WAI-ARIA roles, but this is really only accomplished by filling out the alt portions of HTML to a screen reader could interpret what is on the screen
3. Password Hashing: This just the right amount of salting these passwords will never hit our database in plaintext. This promises our users the peace of mind of using our website.
4. Automated data fetching for the API, using some kind of task scheduler.

**Use Case Diagram**

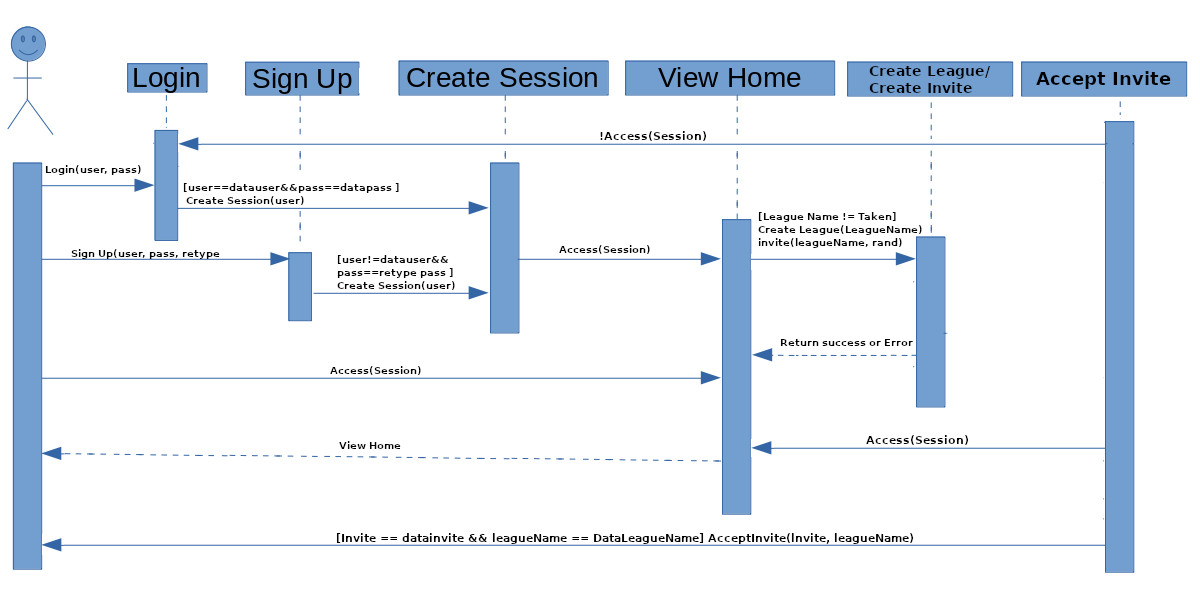


**Use Case Textual Descriptions:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sign Up | Login | Create League | Join League | Start Draft | Choose Team | Edit League | Remove  User | Change  Name | Password Check | Password Reset |
| User | User\* | User\* | User\* | League Owner\* | User\* | League Owner\* | League Owner\* | League Owner\* | User\* | Admin |
| Need a username and password | Must Have Account | Need Players to add to league | Must have an invitation | Must be the League Owner | Must be in the Draft and on draft page | Must be the League Owner | Must be the League Owner | Must be the League Owner | Must have entered a password into login | Request from User |
| 1.Enter username 2.Enter password  3. Renter password | 1. Enter username  2.Enter password | 1. Enter name  2. click create league | 1. Copy and paste invite link  2. log in again | 1. Click begin draft button | 1. Select available team | 1.Choose between remove user and change name | 1.Choose user that is to be removed | 1.Enter new name | 1.check entered password against stored hash  2. login or reset | 1.send link to user to reset password |
| Account created | Successful login or password reset | League is created and invitation to join are sent out | User is a member of the league | The draft will begin  and call choose team for the members of the league | A team will have selected and assigned to a user’s account | The owner will be taken to either remove user or change name | User is removed | League  Name is changed | Password is correct and login is successful or a password reset request is sent to admin | Link to reset password is sent to user |
| None | None | None | None | None | None | None | None | None | None | None |

\*: league owner and admin inherit from use - admin inherits from league owner

Class Diagram and/or Sequence Diagrams

**

**Operating Environment**

Many of the environments being used are local hosts on each person’s machine. All of us use different operating systems such as Linux, Windows, and Mac. Those of us testing and building the backend (php) need to run a webserver. The web server runs the php before sending the resulting code to the clients. Apache is the web server on our local host. We will create a system to officially host the site for testing, and display. It uses a hosting site, named Dreamhost. Dreamhost provides us with shell, sftp, phpmyadmin (This is an easy to use interface for databases), and MySql databases. This allows us to test our code on machine that we all can use, creating a uniform resource.

**Assumptions and Dependencies**

These have not changed since iteration 1.

One dependency is the use of embedded twitter frames. While it is unlikely for Twitter to change the way their frames work since it would break so many sites, it is still something we need to be aware of. Our home page also currently uses a recent games frame from sofascore.com. This could be something we do on our own in the future once our game database is working, but for now it works fine. Also on the home page, we currently have a stand-in chat system provided by minnit.chat, which is just standing in for the chat system we will make in the future. We are dependent on Sports Open Data API. However it has been running since 2016 so it is unlikely this will change.