Beaver Bets

Kaden Fugate, Luke Frugia, Derek Casini, and Samuel Rome

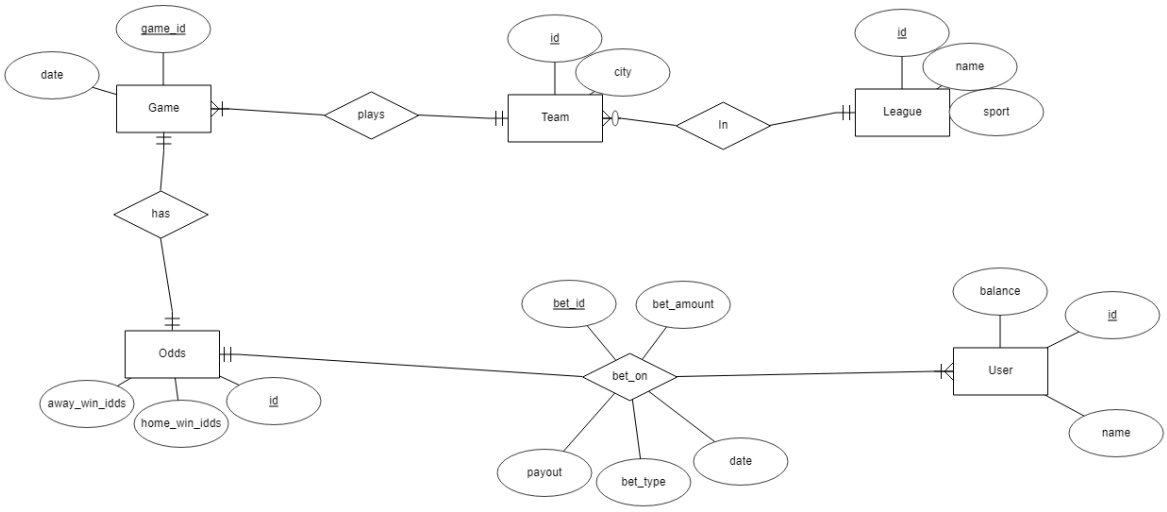
1. Introduction

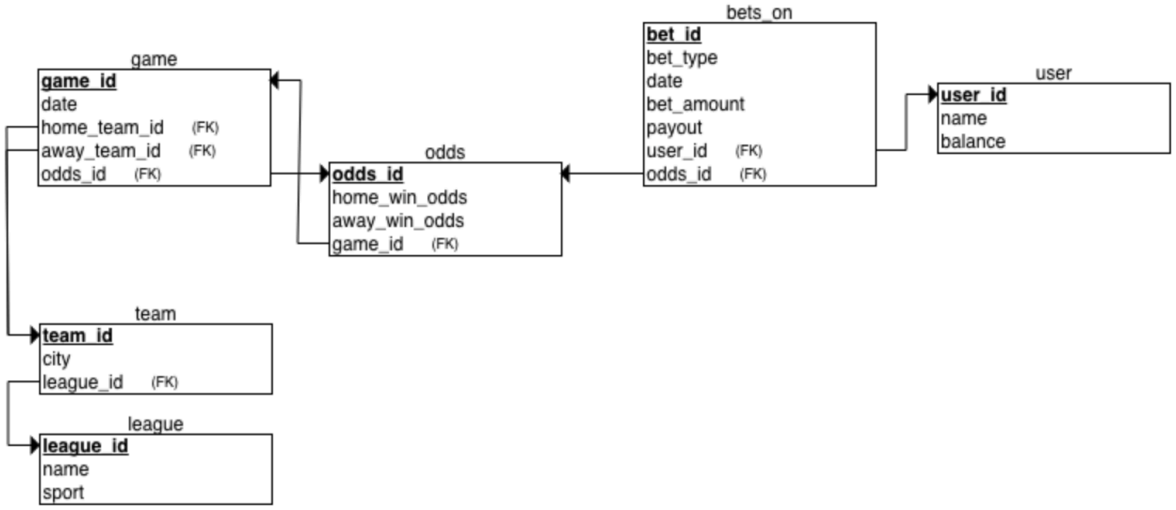
This application is for sports betting users. It will allow them to sign up and make various bets on different sports games given the odds of the game. The main use of this application will be to track the users bets and depending on the odds of the game and whether or not the user chose to bet on the home team or the away team the user will win or lose. The average user will be those that are interested in sports and betting on who is going to win in particular.

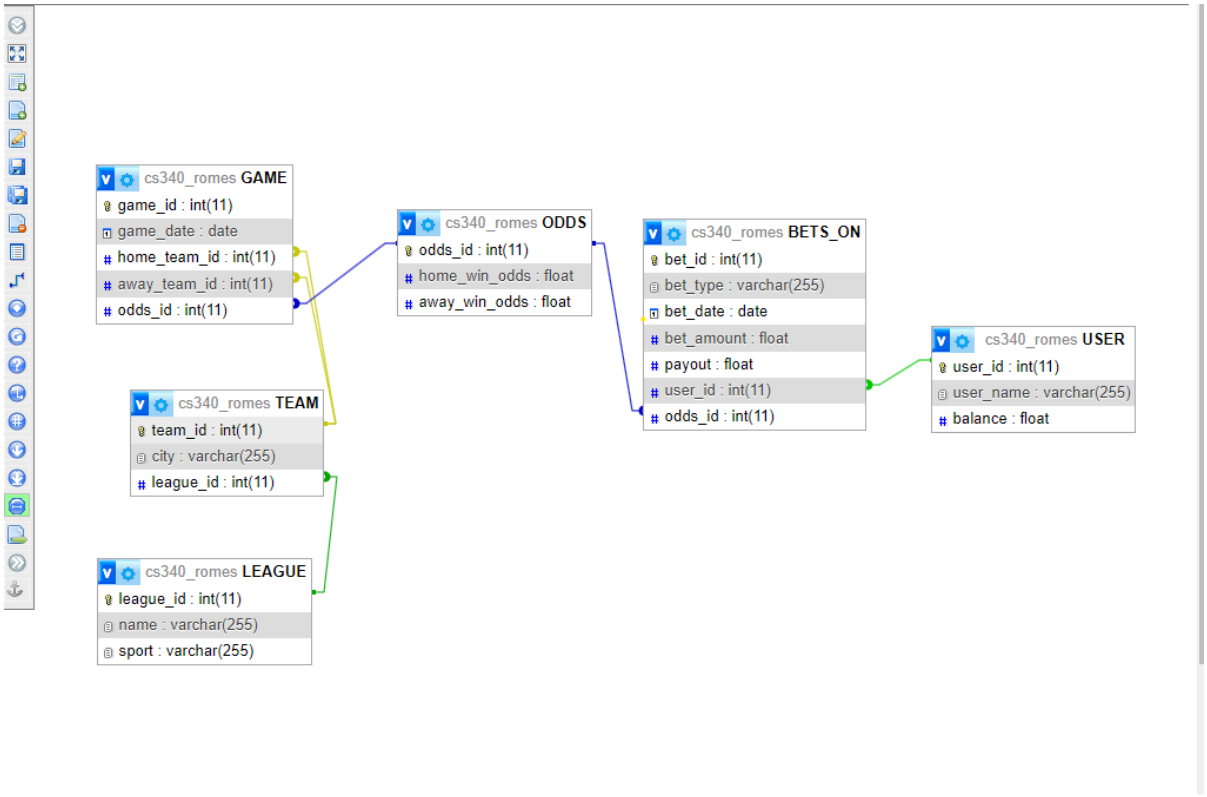
2. Detailed functionality and requirements

Our app allows users to place bets on certain games in various leagues. Before placing a bet, the user must place a bet on who they think will win a game, in what league. Users will see home and away winning odds displayed before they place their bet. After placing bets the app will wait till it receives the results of the game and pay out to winning users. Users must make a positive non-zero bet.

3. Database Design







4. Website Design

Website Layout:

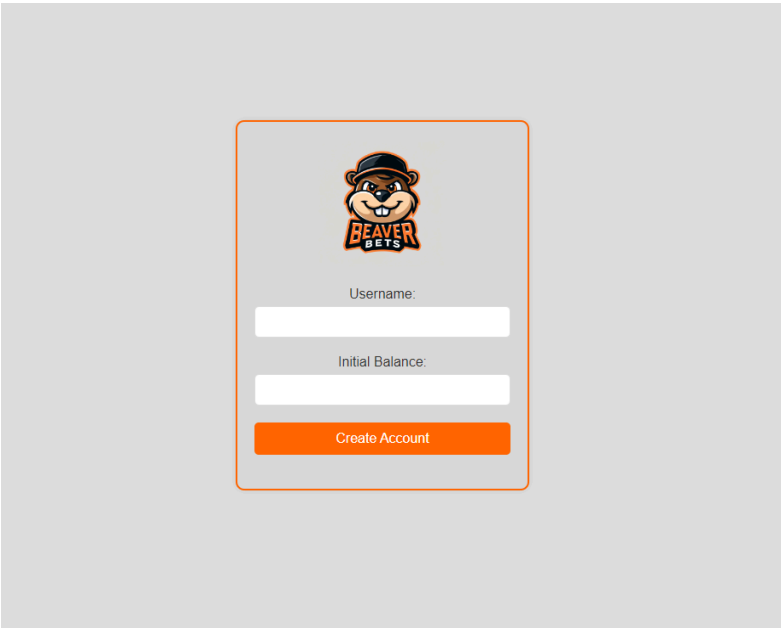
The webpages in our website consist of a pretty straight forward structure. You initially start off on the signup page where you are able to signup and add an initial balance.

You then proceed to your user profile where your balance is displayed and you can see all of your betting information.

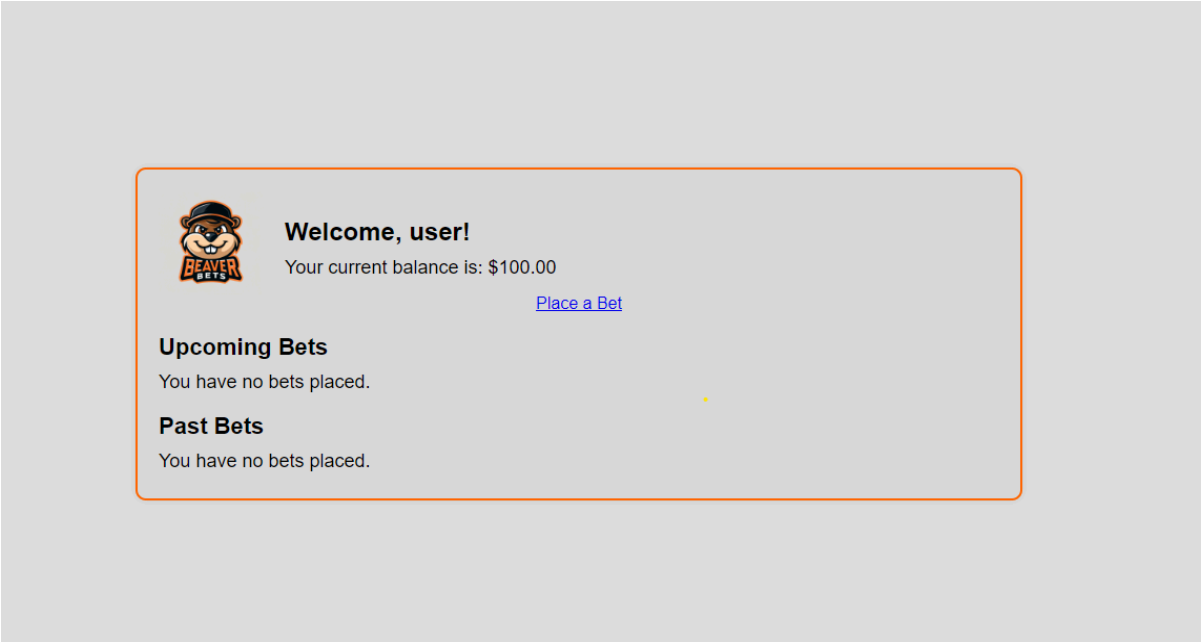
You can then proceed to place a bet where it takes you to a betting page for all available bets on different teams and the odds for those games.

After placing bets you are able to return to your profile page and see all of your new betting information and also allow you to double down or simulate the bet. This will either double your money you spent or it will simulate what the bet would look like after the game is done using a randomly generated number.

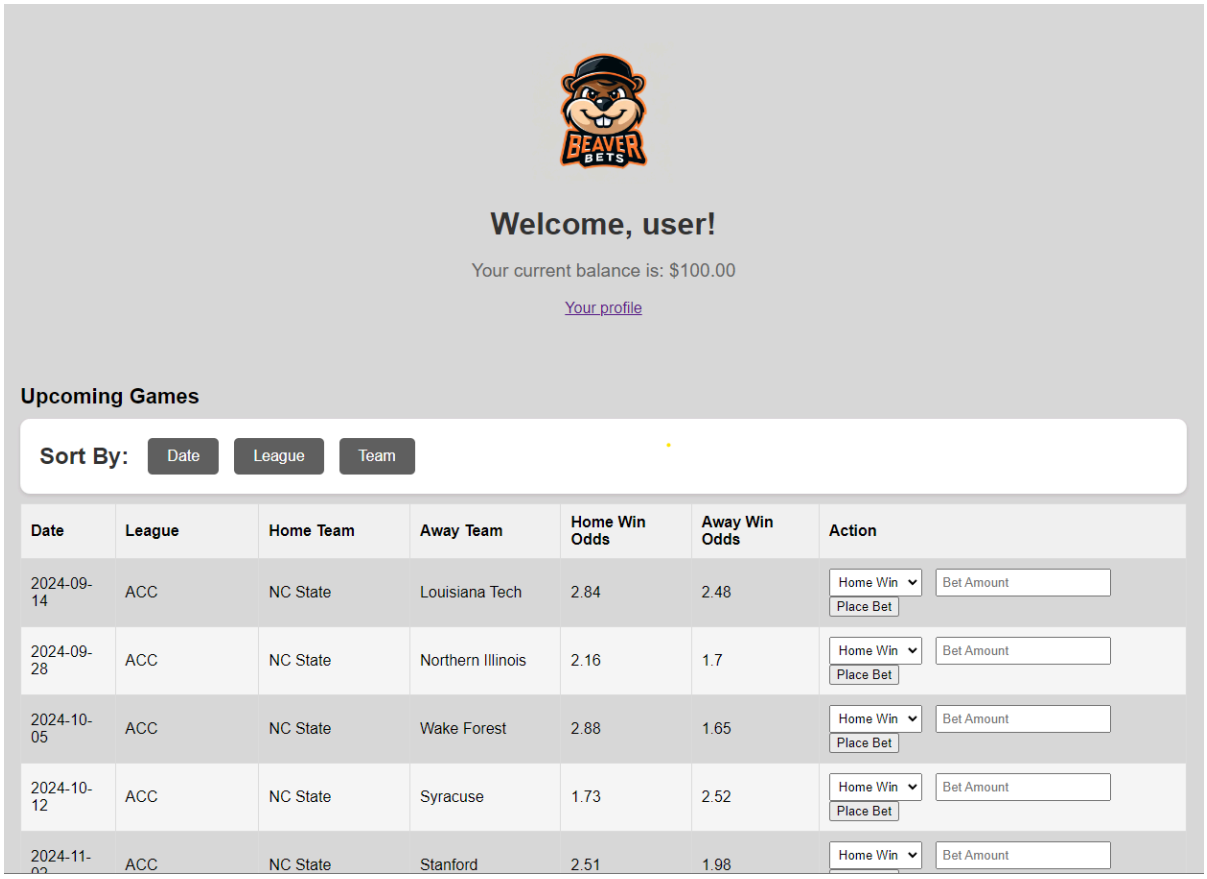
User interface and instructions



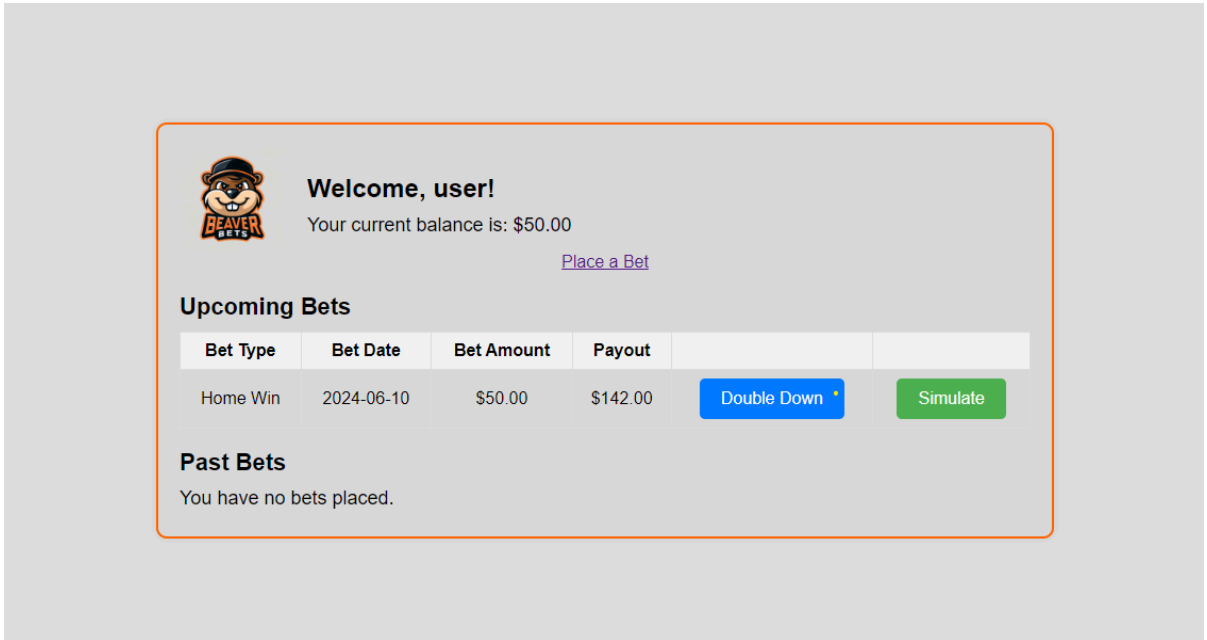
The first operation and the purpose of this page is to be the initial login page where users will be able to input an initial balance and login with their username.



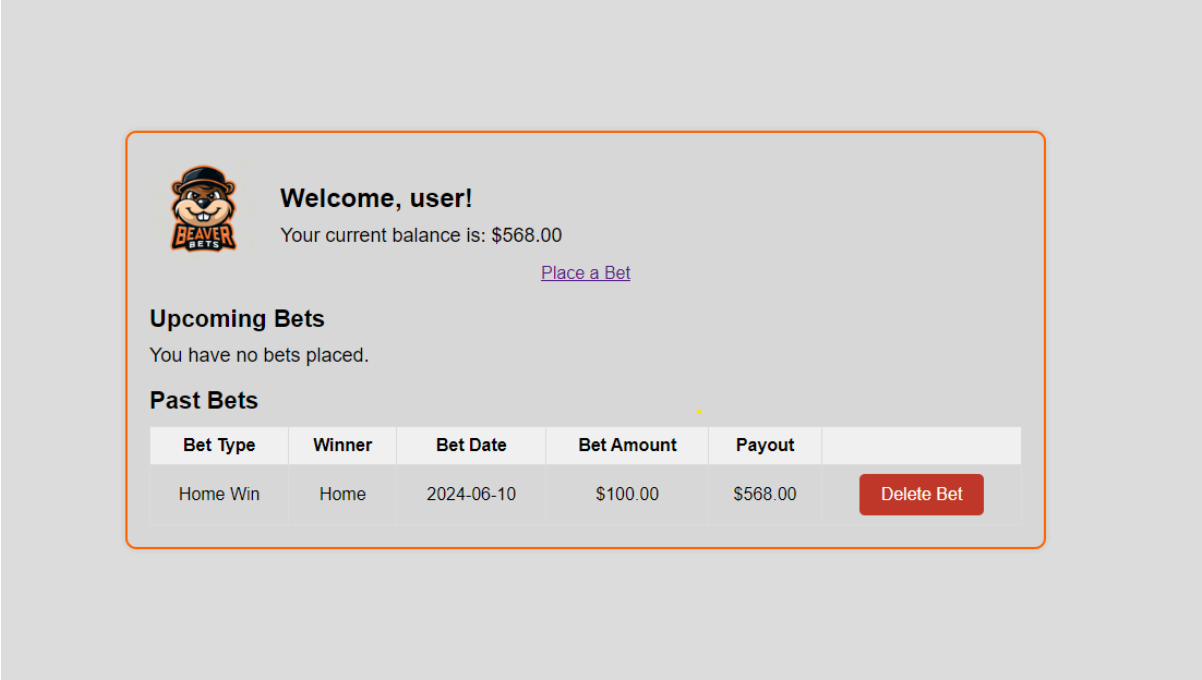
You will then be able to use this page as the main dashboard/profile for the user that will show all upcoming bets and past bets and will also allow you to click the “place a bet” button as an operation.



This page will be used after you hit “place a bet" and will allow the user to look at all the available games they can bet on and add a bet as an operation and it will also allow sorting for various games as another option to perform to see various games and odds.



This is the profile page that you reach after clicking on “place bet” on the previous page and going back to your profile. It takes you back to the profile/dashboard view and shows you your betting information and gives you the option of doubling, and seeing what your bet would look like. As 2 other options for betting operations.



This is the page that you will hit after you click the operation to simulate bets to allow you to see what the bet looks like in the past after the game was over and you won your bet.

How to run the code:

Choco install php

Go to php install (mine was C:\tools\php83)

Find the line: ;extension=mysqli (mine was in the folder php.ini) and remove the “;” which will uncomment it.

You can run it with this command

php -S localhost:8000

And then access the website for example with:

localhost:8000/login.php

If that doesn't work you can also use this link to check it out if you can’t run it locally: https://web.engr.oregonstate.edu/~fugateka/cs340/final/signup.php

5. Application Implementation

Our use of HTML/PHP consists of 3 main files: bets.php, user.php, login.php, and signup.php.

Bets.php contains: HTML that displays the user information and upcoming games with a form to place their bets. The PHP code starts a session and connects to the database and proceeds to fetch and display details about the user and upcoming games and odds on those games. It will also handle bet submissions and calculate payouts and insertion into the database of bets. On top of this the user can choose different methods for sorting. These sort requests will re-query the bets in a different order.

User.php contains: HTML regarding Information and bets, with a link to bets.php in order to place a new bet. The PHP code starts a session to track the state of the user and connects them to the database. It also retrieves both the upcoming and past bets from the database by username. The user can also simulate the game or double down on their bet on this page. Simulation will produce a random number based on the odds and add a winner id to the game activating the GAME tables trigger (updating all of balances of users who had a correct bet on the game). The double down will simply double the users bet amount, calculate a new payout, and subtract the new amount from the users balance.

Login.php contains: HTML that displays a form for the user to input their username. The PHP code starts a session and connects to the database, and then proceeds to validate the username and redirects you if successful to user.php.

Signup.php contains: HTML for the form to get info from the user. The information to retrieve is the username as well as their initial balance. The php will check the username does not already exist and that the balance is a positive non-zero number. Sign up will redirect to user.php

Our use of CSS consists of 3 more files: bets.css, login.css, and user.css. These 3 pages are used to style their corresponding pages to make them more visually appealing to the user.

We also use Javascript to fetch data about college football teams, games, and odds. It processes the data to generate SQL statements, and outputs these statements to be used for populating a database with a league, team, game, and odds data. Ultimately to help automate the creation of a sports betting application database.

We have many main SQL queries we use to set up the tables and our database. We create tables: LEAGUE, TEAM, GAME, ODDS, BETS_ON, and USER in order to have unique identifiers for each aspect of sports betting. We also use SQL queries to insert information into leagues, teams, and games in order to allow users to bet on games from various teams in a sports league.

The function calculate_payout is used to calculate the given payout for a bet based on the amount of the bet and the odds. It does this by taking the bet_amount and multiplying it by the odds of that team. This will then return a payout that the user receives.

The procedure determine_payout is used to update the bets_on table with the payout and adjust the users balance accordingly depending on the bet that is calculated. It does this by retrieving the home win or away win odds and using the calculate_payout function in order to update the payout in the bets_on table and the balance of the user.

The Trigger after_bet automatically decreases the users balance by the bet amount when a new bet is placed. It does this by getting triggered after a new row is inserted into bets_on it will automatically update the users table by decreasing the users balance by the bet amount.

The trigger after_winner_id_set automatically calls the procedure pay_out_bets with the game id of the game that was updated.

The procedure pay_out_bets updates users balances who made a bet on the same game with a correct prediction. The update will add the payout value of the bet they made to their balance.

How to run the code (on windows):
Choco install php

Go to php install (mine was C:\tools\php83)

Find the line: ;extension=mysqli (mine was in the folder php.ini) and remove the ";" which will uncomment it.

You can run it with this command (in the submitted code's directory)

```
php -S localhost:8000
```

And then access the website for example with (while connected to osu wifi or on vpn):

```
localhost:8000/signup.php
```

(On Mac):

Execute `brew install php`
Execute `php -S localhost:8000
Access in the browser of your choice at `localhost:8000/signup.php` while connected to osu wifi or vpn.

If these don't work, we're also hosting the webpages on the OSU web server at this link: https://web.engr.oregonstate.edu/~fugateka/cs340/final/signup.php

6. **Future Work and Lessons learned**

Beyond this class we could implement an actual money deposit system to complete the functionality of the betting app. In the initial setup process, we struggled a little to get the app running but had little to no trouble once that part was figured out. To address that issue, we just needed to do a little more research. We feel we completed the project how we wanted, so we don't believe we would change anything if done again.

Appendix - Team Report
Division of labor summarized:

Project Report, Grade Sheet and assistance: Derek Casini, Samuel Rome

Coding and assistance: Luke Frugia, Kaden Fugate