Lab 5 (Nov 5) #Team1

Sprint-5

Retrospective meeting: any updates since last meeting? any issues?

Planning meeting:

- what analytics feature(s) to implement next week?
- breakdown the goals into the actionable tasks
- define task completeness criteria (test cases if needed)

Feature:

- Analytics:
  - Map
  - Team Average Ratings

GUI:

Taskboard:

Done list of last sprint 4:

- small screen(1000px) view/desktop view (Hoda)
    - Acceptance criteria: Website looks pretty when the screen size is shrunk to less than 800px and no functionality of the website is hidden
- Create shared nav bar view (Hoda)
    - Acceptance criteria: nav bar works on all pages of the website, button links take you to each page
- Create frontend form for ratings (Hoda)
    - Acceptance criteria: looks pretty, clicking on the button receives data from database, confirms update in backend.
- Create MostCommonAge analytic function (Sprint 4) (Calvin, Mustafa)
    - Acceptance criteria: Return list of players with the most common age.
- Create TopandLowestRated analytic function (sprint 4) (Abraham, Evan)
    - Acceptance criteria: Return list of players who are highest and lowest rate (10 each)
- Create average rating calculation analytics function (sprint 4) (Abrahm, Evan, Calvin, Mustafa)
    - Acceptance Criteria: Returns list of teams with rating values for each team
- List Analytic functions on website (Mustafa Abraham)
    - Acceptance criteria: Get results from backend functions and display it on our website.

Todo list for next sprint:

-  Form validation for modify (Calvin)
    - Acceptance criteria: creates alert when some elements are not filled out/ not valid
-  Soccer field on homepage (Hoda)
    - Acceptance criteria: two soccer fields, one for analytics, one for features.
        - Each has buttons leading to the features
-  Soccer ball cursor(Hoda)
    - Acceptance criteria: a tiny soccer ball will take the place of the arrow
-  Common Age frontend(Hoda)
    - Acceptance criteria: looks pretty
-  Top and lowest rated frontend(Hoda)
    - Acceptance criteria: looks pretty
-  Average rating frontend(Hoda)
    - Acceptance criteria: looks pretty
-  Form validation for add (Calvin)
    - Acceptance criteria: creates alert when some elements are not filled out/ not valid
- Search bar on search page
    - Acceptance criteria: display a smaller search bar on results page
-  Create map and populate with player names and nationality analytic function (Mustafa Abraham Evan)
    - Acceptance Criteria: Map entity displays with all players
-  Create Analytic functionTeamAvg (Evan Abraham Mustafa)
    - Acceptance Criteria: Return list of top 10 highest rated teams.
- Create MostPopularPosition analytic function (Todo sprint 6)
    - Acceptance Criteria: Return top 3 most popular positions.
-  List Analytic functions on website for new analytics (Mustafa,Evan)
    - Acceptance criteria: Get results from backend functions and display it on our website.
- 

Test Cases:

- **Feature 1 (Insert): As a user I want to be able to add a new player to the Database with the information needed (name, nationality, team, player id).**
    - **Test case 1: User enters all the necessary information of a player into the website**
        - **Correct output: Website successfully takes in new information and updates the database with the player**
    - **Test case 2: User enters incomplete information about a player (eg. missing age, rating, etc.)**
        - **Correct output: Website throws an error requiring the user to enter all fields with information**
    - **Test case 3: User inputs a non alphanumeric character in any of the fields**
        - **Correct output: An error is thrown to notify the user that an illegal character has been inputted for information**

- **Feature 2 (modify): User want to update the players' information and keep the data up to date.(team, rating, age, position)**
    - **Test case 1: Users want to change the age for one of the players in the data. Users will find the players who need to be changed. Then,click the modify button and select the age. The textbox will show up and users are required to enter the correct age in the textbox and click the submit button.**
        - **Correct output: Server will be updated and show the age that users changed.**
    - **Test case 2: Users want to change the position for one of the players in the data. Users will find the players who need to be changed. Then,click the modify button and select the position. The dropdown menu will show up and users are required to select the position from the menu and click the submit button.**
        - **Correct output: Server will be updated and show the position that users changed.**
    - **Test case 3: Users want to change the rating for one of the players in the data. Users will find the players who need to be changed. Then,click the modify button and select the age. The textbox will show up and users are required to enter the correct rating in the textbox and click the submit button.**
        - **Correct output: Server will be updated and show the rating that users changed.**
        - **Correct output: Server will bring to user and show all the player list from the Club that users choose and the bottom of the list will show the average rating.**

  Feature 3 (Search): As a user I want to be able to search a player(s) from the Database with the information needed (name, nationality, team, age).

    - **Test case 1: User enters all the necessary information of a player into the website**
        - **Correct output: Website successfully takes the input and searches the qualified players from the database and displays them.**
    - **Test case 2: User enters wrong information about a player (eg. invaid age, rating, etc.)**
        - **Correct output: Website throws an error requiring the user to enter field with valid information**
    - **Test case 3: User leaves the field empty.**
        - **Correct output: An error is thrown to notify the user are required to fill the field**

- **Analytic 1 (World map): We want to display a world map that indicates where players are from each country**
    - **Test case 1: Display all players and country of origin**
        - **Correct output: World map is populated with 'points or dots' that shows each country and how many players are from there. Example Portugal has 14 players, so there will be 14 points on portugal.**
    - **Test case 2: Country with no players eg (Monaco)**
        - **Correct output: Monaco map will be displayed with no points as fifa 21 has not included any players in their database.**

- **Analytic 2 (Best Hits): User want to checkout which players have the best hits**
    - **Test case: User wants to see which player is most popular**
        - **Correct output: Returns list of players with most hits.**

- **Analytic 3: Users want to know the average rating for Club A. Users will type the Club name in the text box. Then,click the average button.Feature 3 (World map): We want to display a world map that indicates where players are from each country**
    - **Test case 1: Display all players and country of origin**

        - **Correct output: World map is populated with 'points or dots' that shows each country and how many players are from there. Example Portugal has 14 players, so there will be 14 points on portugal.**
    - **Test case 2: Country with no players eg (Monaco)**
        - **Correct output: Monaco map will be displayed with no points as fifa 21 has not included any players in their database.**

Feature 4 (Delete):User want to delete the players' information(team, rating, age, position)

Test case 1: Users wants to delete a user from the database

- **Feature (Delete): User want to delete a player from Database**
    - **Test Case1: Player enters 'Lionel Messi' to delete him from database**
        - **Correct Output: Player 'Lionel Messi' and all his attributes are deleted from Database**
    - **Test Case 2: User enters incorrect information filled with non alphanumeric characters**
        - **Correct output: The page throws an error that tells the user of incorrect input**
    - **Test Case 3: User enters a player name that does not exist (eg 'John Cena')**
        - **Correct output: 'Player does not exist in database'**

- **Analytic 4 (TopAndLowestRated): User want to checkout the top and lowest rated player**
    - **Test case 1: User will type 'top100' and it will return top 100 players from highest to lowest**
    - **Test case 2: User will type 'low100' and it will return low 100 players from lowest to highest**

- **Analytic 5 (Most common age): User want to checkout the Most common age group**
    - **Test case : Will return most popular age group ie '24' and return all stats of each player**
    - **Test case: If two age groups have exactly the same number of players then return players from the same age group ie '24' and '27' will return all players from both age group.**

- **Analytic 6 (Most common Position top 3): User want to checkout the Most common Positions**
    - **Test case : Will return top 3 most popular Positions eg "LW" "RW" "CDM". Will return all players from these positions and all their stats.**