

# YourSports

Dynamic Debugger (Team 02)

## Team & Backend Lead

Kshitiz Sareen: [ksareen@sfsu.edu](mailto:ksareen@sfsu.edu)

## Frontend

Lead Kevin Islas

Developer Shamar Ireland

## Functional

Lead Sabur Saigani

Assistant Jonathan Ip

## Database Administrator

Wenye Guo

## Github Master

Mathew O Abiola

## Project URL

<http://34.136.124.189:3000>

## Date

August 2nd, 2022

## Milestone 5

Milestone	Date
M5	August 2nd, 2022
M4V2	July 31st, 2022
M4V1	July 27th, 2022
M3V2	July 22nd, 2022
M3V1	July 18th, 2022
M2V2	July 14th, 2022
M2V1	July 7th, 2022
M1V2	June 30th, 2022
M1V1	June 9th, 2022

## Product Summary

Name

**YourSports**

Major Committed Functions

Regular user

1. Regular users shall be able to create an account.
2. Regular users shall be able to sign up as a creator account.
3. Regular users shall be able to access the homepage.
4. Regular users shall be able to view the live scores of a game.
5. Regular users shall be able to view the duration of a game.
6. Regular users shall be able to view updated statistics of players.
7. Regular users shall be able to search for articles through a search bar.
8. Regular users shall be able to filter articles they would like to read by keywords.
9. Regular users shall be able to filter articles they would like to read by different types of sports.
10. Regular users shall be able to search articles by heading of the article.
11. Regular users shall be able to view the article heading in the search results for articles.
12. Regular users shall be able to view the article author in the search results for articles.
13. Regular users shall be able to view the article postdate in the search results for articles.
14. Regular users shall be able to view the article image as a thumbnail in the search results for articles.
15. Regular users shall be able to search games by the teams playing in the search bar.
16. Regular users shall be able to filter games by the sport type.
17. Regular users shall be able to filter games by their date.
18. Regular users shall be able to view the game score in the search results for games.
19. Regular users shall be able to view the game date in the search results for games.

- 
- 20. Regular users shall be able to view the game location in the search results for games.
  - 21. Regular users shall be able to search for player statistics according to the player name in the search bar.
  - 22. Regular users shall be able to filter player statistics by the sport type.

#### Registered user

- 23. Registered users shall have the same privileges as regular users.
- 24. Registered users shall be able to log in.
- 25. Registered users shall be able to log out.
- 26. Registered users shall be able to reset their password.
- 27. Registered users shall be able to delete their account.
- 28. Registered users shall be able to post in the discussion forum of the game.
- 29. Registered users shall be able to delete their posts in the discussion forum.
- 30. Registered users shall be able to comment on articles.

#### Basic Account

- 31. Accounts shall be able to be deleted by users.
- 32. Accounts shall contain the name of the users.
- 33. Accounts shall contain the username of the users.
- 34. Accounts shall contain the email of the users.
- 35. Accounts shall contain the password of the users.

#### Creator Account

- 36. Creators shall have the same privileges as registered users.
- 37. Creators shall be able to upload articles to the website.
- 38. Creators shall be able to view Dashboard.

#### Admin Account

- 39. Admin shall have all the privileges.
- 40. Admin shall be able to create discussion forums for upcoming games.

#### Articles

- 41. Articles shall be posted by creator users.
- 42. Articles shall be able to be filtered by sports type.
- 43. Articles shall be able to be filtered by keywords specified by the user.
- 44. Articles shall contain a heading.
- 45. Articles shall contain a subheading.
- 46. Articles shall contain an introduction.



47. Articles shall contain contents.
48. Articles shall contain a conclusion.
49. Articles shall contain an image.

## Games

50. Games shall be filtered by date.
51. Games shall be filtered by sport type.
52. Games shall be filtered by keywords specified by users in the search bar.
53. Games shall have an individual discussion forum.
54. Games shall be able to show live scores of corresponding teams.
55. Games shall be able to show the date of the game.
56. Games shall be able to show the location of the game.

## Discussion Forum

57. Discussion forum shall be able to be viewed by all users.
58. Discussion forum shall be able to show posts from users.

## Dashboard

59. Dashboard shall be able to show the number of views of the article.
60. Dashboard shall be able to show the heading of the article.
61. Dashboard shall be able to show the posted date of the article.

## Backend Service

62. The backend service shall be able to validate the email for duplicates when users try to sign up.
63. The backend service shall be able to check for duplicate accounts when users try to sign up.
64. The backend service shall be able to create an account in the database.
65. The backend service shall be able to store posts by a user in the discussion forums.
66. The backend service shall be able to update player statistics in the database.
67. The backend service shall be able to filter articles specified by the user.
68. The backend service shall be able to filter games specified by the user.
69. The backend service shall be able to filter player statistics specified by the user.

## Website

70. The website shall be able to check if the name is not empty when users try to sign up.

- 
- 71. The website shall be able to check if the email is not empty when users try to sign up.
  - 72. The website shall be able to check if the password is at least 8 characters before the users try to sign up.
  - 73. The website shall be able to check if the password is a maximum of 20 characters before the users try to sign up.
  - 74. The website shall be able to check if the terms of service have been accepted when users try to sign up.

#### Comment

- 75. Comments shall be able to be viewed by regular users.
- 76. Comments shall be posted by registered users.
- 77. Comments shall be able to be deleted by poster.

#### Unique Features

Our product will offer countless services to users who are interested in sports information and are sports fans. Users will be able to share their opinions with each other through news articles. Other users can read news articles, and they can also react to news articles through comments, likes, and dislikes from other users. Users who have uploaded their articles can also view different statistics like the number of likes, number of dislikes, and number of views that their article has received. These statistics will give the Author of the article, how their article is being perceived by other users of the product.

Our product also allows users to interact with other users who are watching the same sports game through the discussion forums provided for each game. Users can share their opinions on the game, and interact with other users who have the same interests. This will serve as a great way to build a community of users with similar interests through sports.

Finally, our product keeps up to date with player statistics, so that users can access the latest information on their favorite players at their fingertips.

#### URL

<http://34.136.124.189:3000>



# Milestone Documents

# YourSports

Dynamic Debugger (Team 02)

---

## Team & Backend Lead

Kshitiz Sareen: [ksareen@sfsu.edu](mailto:ksareen@sfsu.edu)

## Frontend

Lead Kevin Islas

Developer Shamar Ireland

## Functional

Lead Sabur Saigani

Assistant Jonathan Ip

## Database Administrator

Wenye Guo

## Github Master

Mathew O Abiola

## Milestone 1

Milestone	Date
M1V2	June 30th, 2022
M1V1	June 9th, 2022

## Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Executive Summary</b>	<b>2</b>
<b>Use Cases</b>	<b>3</b>
Actors	3
Cases	6
<b>Main Data Items and Entities</b>	<b>13</b>
<b>Functional Requirements</b>	<b>14</b>
Registered Users	14
Unregistered Users	15
Backend Service	16
Website	16
Creators	17
Dashboard	17
Group Owner	17
<b>Non-functional Requirements</b>	<b>18</b>
System Requirements	18
Performance Requirements	18
Storage, Security and Environmental Requirements	18
Marketing and Legal Requirements	19
Content	19
Privacy	19
<b>Competitive Analysis</b>	<b>20</b>
<b>Technology Specifications</b>	<b>22</b>
<b>Checklist</b>	<b>23</b>
<b>List of Team Contributions</b>	<b>24</b>

## Executive Summary

The product that we are trying to build is called YourSports. Today, many sports applications are focused on providing news and real-time updates to live games, however, most of these products don't focus on the information that a particular user is looking for, but instead provide information as a whole, leaving the user to scrape through all the data, to find any information that is relevant to their interests. This leaves the user unsatisfied, hence they stop using these products. Our product is a one-stop solution for all the sports fans, as our application focuses on personalizing information for the users, by providing data that is focused on their favorite teams and players.

YourSports is a web application that offers countless services to its customers, from real-time trends to fantasy betting, that will keep users engaged and updated with real-time information. Our product will allow users to access live Twitter feeds that relate to their favorite teams and players, provide information on any games that they would like to watch, allow users to easily access player statistics, provide real-time updates of live games, and have group chats where users can interact with fans of similar interests. Our product will also keep users engaged through fantasy betting. The sports industry involves a lot of betting, where fans have lost enormous amounts of money. Our product will allow fans to bet with the same amount of energy, and also not waste any money through fantasy betting. Instead of betting with money, users can bet through points, and they will win or lose according to the outcome of the games they have betted on. These points can be used to win prizes. This is one of our application's unique services that will help users by disrupting the betting industry.

Our product is designed to cover wider aspects of contents without overflowing users with unnecessary information, and we believe that this attracts more consumers. Since it is very interactive, it can generate revenue by providing enormous amounts of data to different companies that can use this data to advertise their products on our platform. This application can also be expanded to a talent acquisition company, that will help new and talented players enter the industry. Our product is the solution to all the problems that existing products in the sports industry face, and it has a lot of potential for growth.

## Use Cases

### Actors

Steward (Student):

Characteristics, skills, and pain points:

- Likes to keep track of sports
- Very busy with his studies
- Very impatient
- Very straightforward
- Gets irritated easily
- He does not like browsing through many websites to be updated with sports news.

Goals:

- He wants to be updated with sports news without browsing multiple websites.
- He would only like news that caters to sports

Mathew (Gambler):

Characteristics, skills, and pain points:

- He loves to take risks
- He loves to travel a lot
- He is addicted to gambling
- He has lost a lot of money from gambling

Goals:

- He wants real-time updates on the bets he places while traveling so that he is not anxious about whether he has won or lost money
- He wants to continue gambling, without losing actual money
- He wants a platform that can help him practice betting with virtual money, instead of real money, so that he is better prepared when placing bets with real money.

Wendy (Cashier):

Characteristics, skills, and pain points:

- Wendy has a child who she needs to take care of.
- She is a big fan of the warriors.

- 
- She is very busy with her family.
  - She cannot watch games due to time constraints.

Goals:

- She wants real-time updates of games in which the Lakers play through text broadcast and commentary as she cannot watch live games.
- She also wants notifications instead of checking through multiple platforms for information.
- She would also like to be involved in the game by sharing her opinion on live games.

**John (Engineer):**

- Characteristics, skills, and pain points:
- John has no interest in sports.
- John's friends are interested in sports, and he feels left out because he cannot have a conversation with them when they talk about sports.
- He is very social.
- He likes to make new friends.
- He is very busy with his work

Goals:

- He wants easy access to sports information so that he can learn more about sports.
- He wants to make new friends who are into sports and have conversations with them.
- He wants to follow basketball and he does not want to watch live games, however, he wants notifications of any updates in a game.

**Jane :**

- Characteristics, skills, and pain points:
- Jane is a big fan of basketball.
- She is a big fan of the Lakers.
- She is a big fan of LeBron James.
- She has no interest in reading news that is not related to Lakers or LeBron James.
- She gets distracted easily.
- She uses Twitter a lot.

Goals:

- 
- She wants news that is only related to the Lakers or Lebron James.
  - She does not want to miss any tweets related to Lebron James or the Lakers.
  - She does not want to open Twitter because she gets distracted by tweets from other users and topics.
  - She also wants updated stats of the players in a season.

#### Maxey (Student):

- Characteristics, skills, and pain points:
- Maxey loves basketball.
- Maxey loves going out with his friends to watch basketball games.
- He is really busy with studies and work.
- He works in low-income jobs and cannot afford to watch games that fit his budget.
- He is impatient.
- He does not like checking through multiple websites to find nearby games, where his favorite teams are playing, and they fit his budget.

#### Goals:

- He wants information on games that are not far away, have his favorite teams playing, and fit his budget.
- He wants notifications of any games that fit his criteria.
- He wants real-time updates of any games through text that he is not able to attend.

#### Julia (Sports Journalist):

##### Characteristics, skills, and pain points:

- Julia creates news articles about sports.
- She creates articles that cater to users of specific interests.
- She does not like publishing her articles on every sports website.
- Her articles don't get many views because they are not available to users that the article is intended for.

##### Goals:

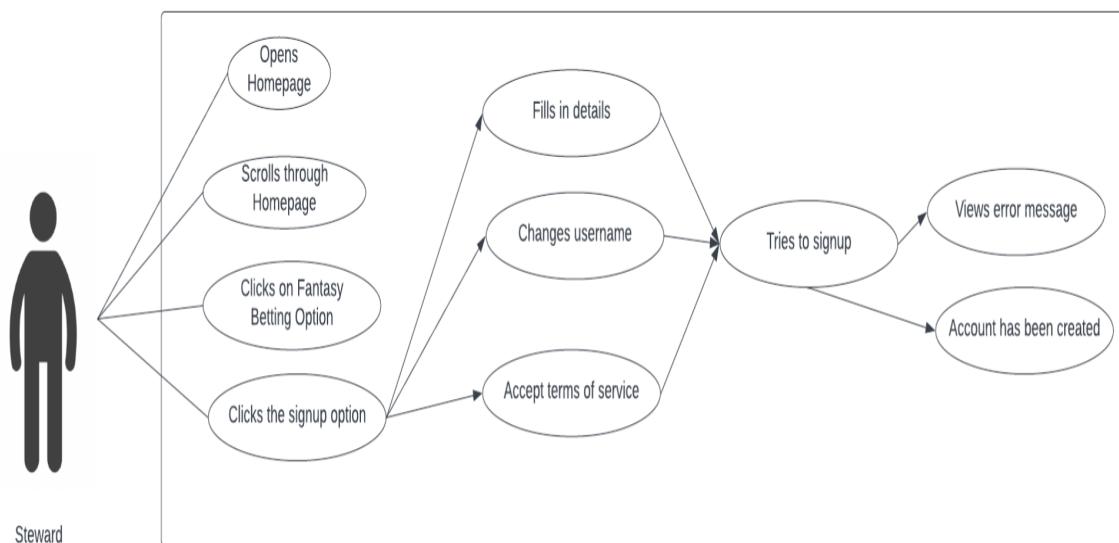
- She wants to publish her articles on websites that cater to the specific interests of users.
- She wants her articles to be read by users who have the same interest in the information presented in her article.

## Cases

### Signing up

Actors: Steward

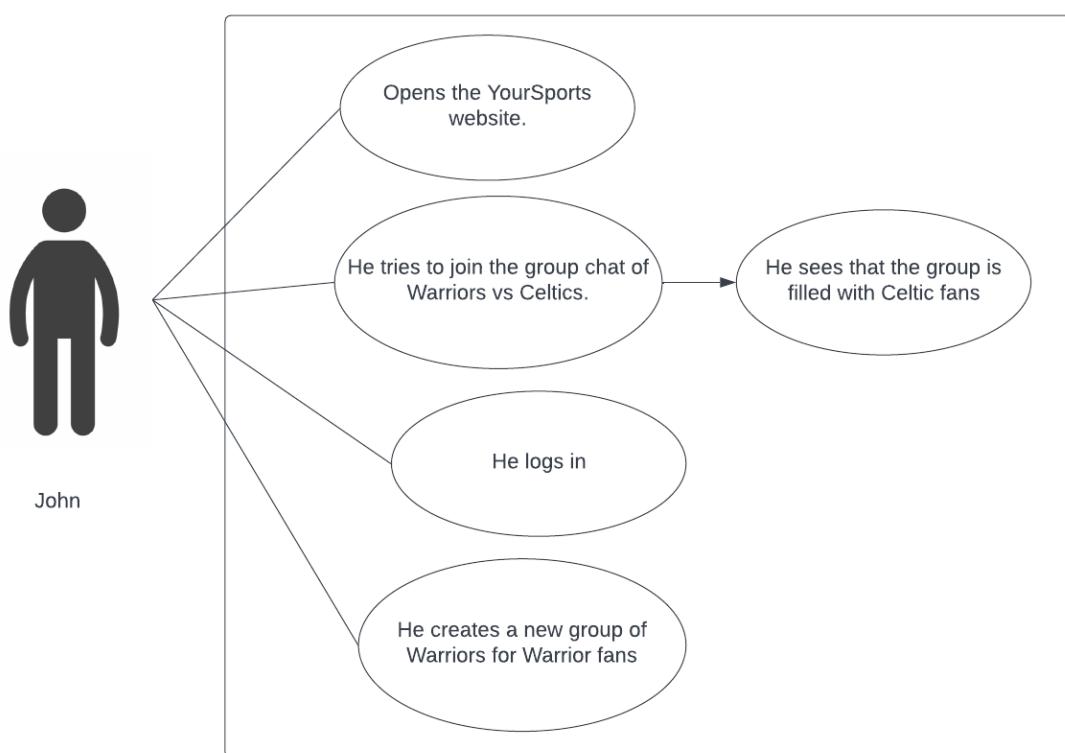
Steward makes a Google search on the latest sports news. The third search result shows him the YourSports website. He opens the website and scrolls through the homepage. The homepage has all the latest news and it shows a lot of features like searching games, and fantasy betting. He clicks on fantasy betting however it states that he must be signed up to use this feature. He is intrigued by fantasy betting and he wants to sign up. He clicks the sign up option, where he is prompted to fill in details like first name, last name, username, email, password and date of birth. He fills in all the details and clicks sign up. An error message popped up stating that a user with the same username already exists. He changes the username and tries to sign up again. It fails again because he has not yet agreed to the terms of service. After agreeing to the terms of service, he tries again and this time the sign up is successful.



## Creating Group Chat

Actors: John

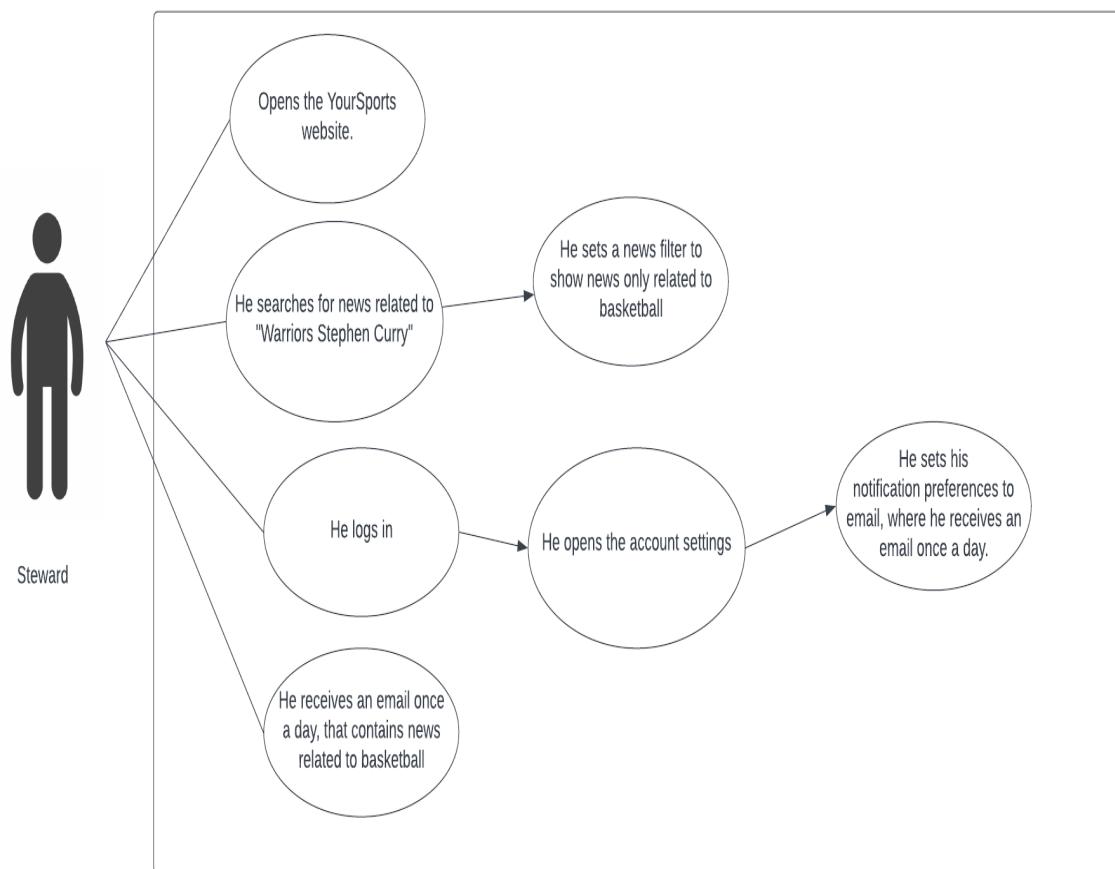
John wants to follow the game of Warriors vs Celtics. Unfortunately, he is traveling, and he has no one to watch the game with. He does not want to watch the game alone. He opens the YourSports website to follow the game. He notices he can join group chat to discuss with others. He logs in his account and joins the chat named Warrios vs Celtics. But he finds that they are all fans of the Celtics. He creates a new group chat named Warrior and invites some of his friends who are fans of Warrior to this chat.



## Checking Sports news

### Actors: Steward

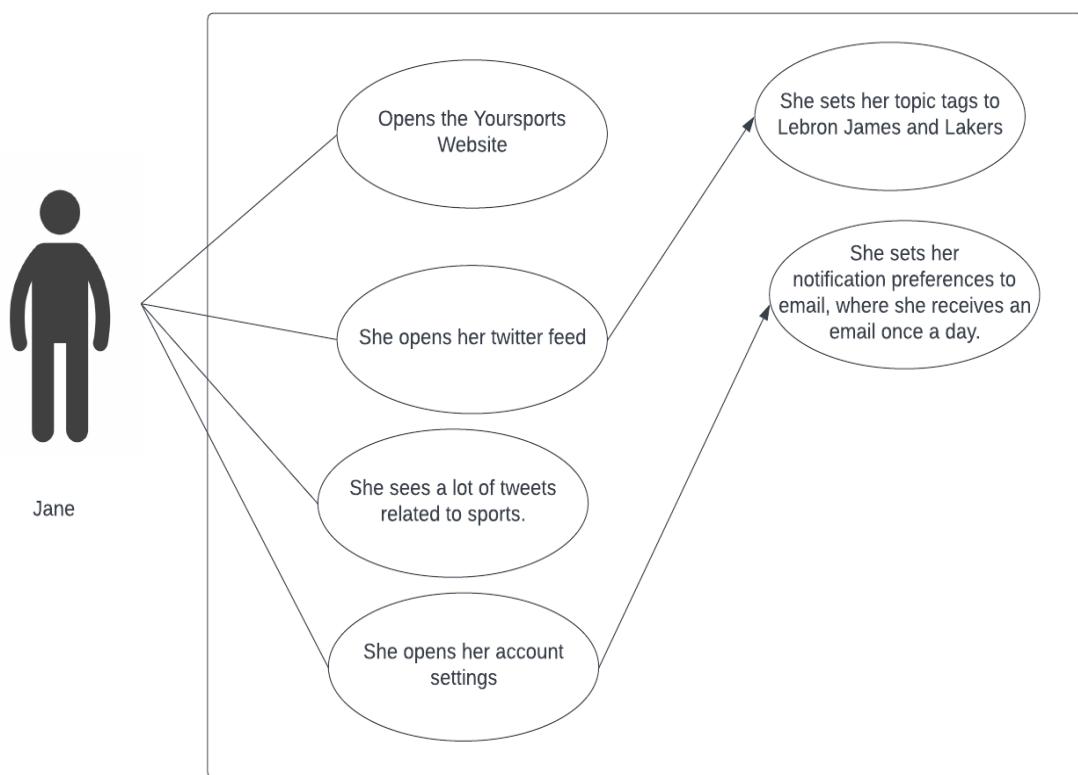
Steward likes to keep track of sports. His favorite sport is basketball. He has subscribed to many sports websites and mobile apps that send him updates on sports news via email and notifications. But he only wants to read news that is related to basketball. He goes to YourSports and searches by teams and players e.g. "Warriors Stephen Curry" then reviews the results. He finds that the information provided by YourSports is detailed and latest. He decides to use this site to follow his favorite sport. He logs in, and from the account settings, he chooses notification news to only be related to basketball, sets his preferences of receiving emails on basketball news only once a day.



## Checking Twitter Feeds

### Actors: Jane

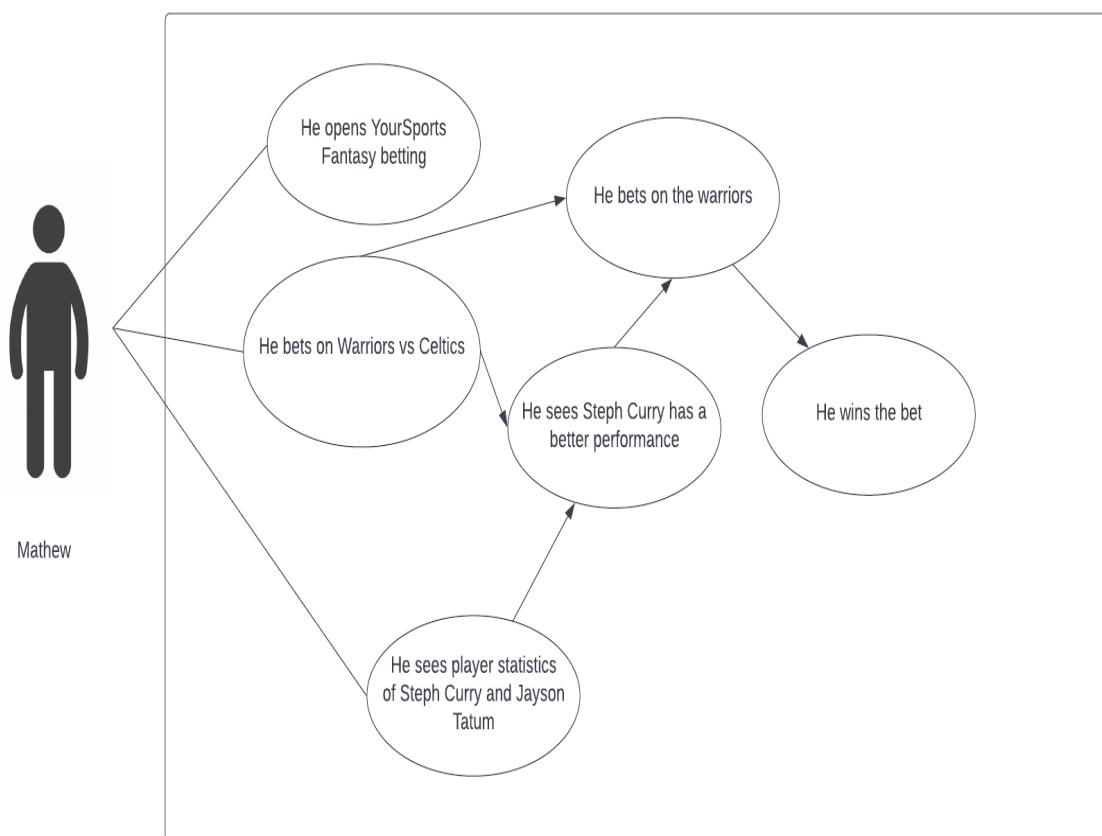
Jane is a big fan of Lebron James and the Lakers, and she regularly tweets about them. She likes to know what other people are tweeting about Lebron James and the Lakers. She does not want to miss a single tweet, but she does not want to open twitter to avoid distraction. She opens the YourSports website to see live Twitter feeds on sports. She chooses her target topics as Lebron James and the Lakers to only see twitter feeds related to these two teams. She also enables the option from her account settings, to receiving emails once a day from YourSports regarding any new tweets that are related to the topics she follows.



## Fantasy Betting

### Actors: Mathew

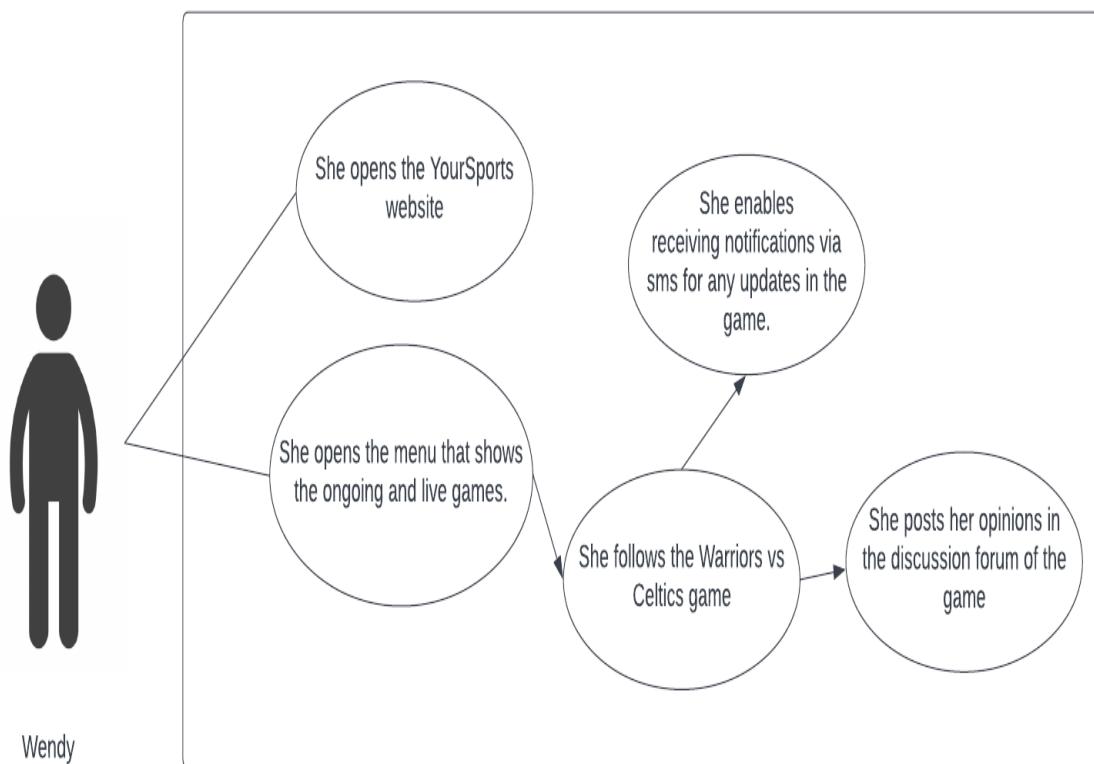
Mathew is very addicted to gambling and sports betting. He has lost a lot of games previously and is in a lot of debt. He is trying to find a platform where he can practice sports betting without using real money. His friend tells him about YourSports where he can bet on any live and upcoming games, with YourSports coins instead of real money. He decides to bet on the game of the Warriors vs Celtics. He sees the player statistics of Steph Curry and Jayson Tatum. He sees that Steph Curry has a better performance, and therefore Warriors have a higher chance of winning. The updated statistics of Steph Curry helped Mathew make a more calculated decision on who to bet on. His decision was fruitful as Warriors won the tournament. He realized he enjoyed fantasy betting as much as betting with real money, and he does not wish to bet with real money anymore.



## Following Sports Games

### Actors: Wendy

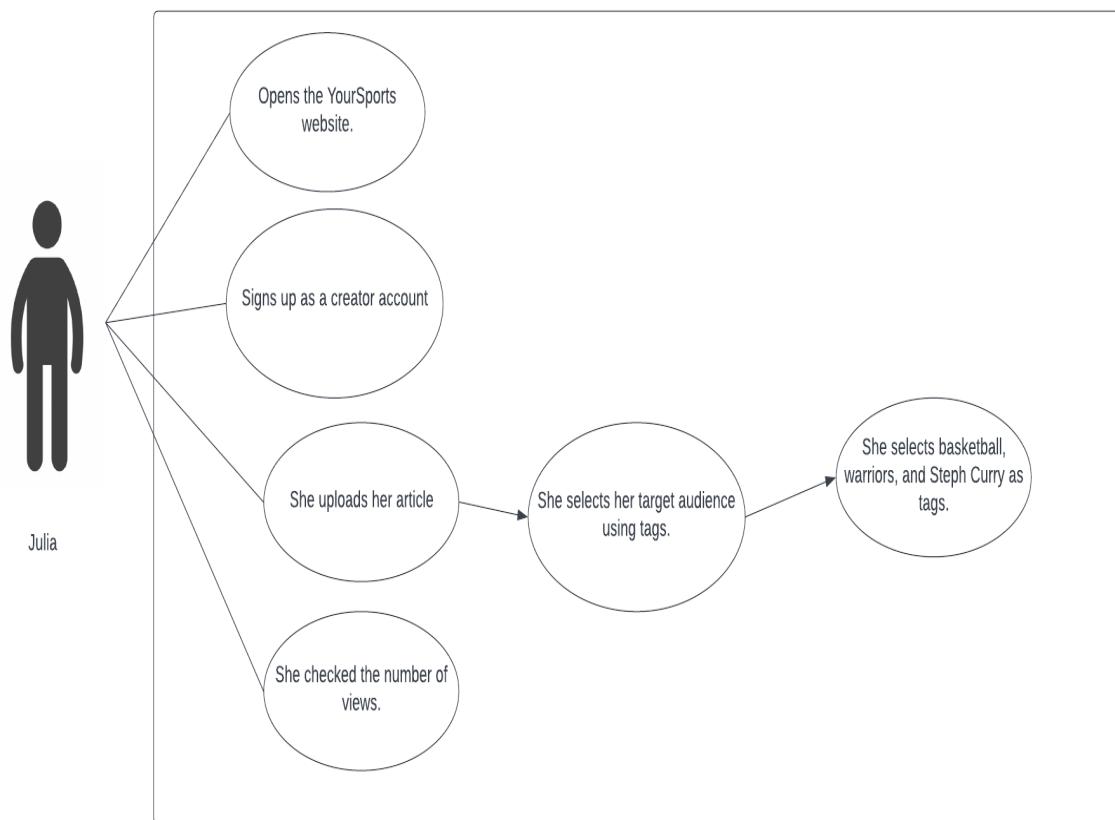
Wendy is very busy with her work and she also has a child to take care of. Due to her busy schedule, she cannot watch the games of the Warriors vs Celtics. She cannot watch the videos and would like to read text commentary about the game. She opens YourSports website, and chooses to follow the Warriors vs Celtics game, and set receiving notifications via SMS on her phone, for any update in the game. From time to time, she also opens the YourSports website and posts her views on the current game in the discussions forum of the game.



## Posting news articles

### Actors: Julia

Julia is new in the sports journalism industry, and a lot of her articles are not being read by her target audience. She opens YourSports website, and signs up as a creator account. She uploads her article on how the Warriors won the NBA championship, and how all of San Francisco rejoiced their win. While uploading her article, she selects her target audience from the tags. She selects basketball, Steph Curry, and the Warriors as tags. She checked the number of views few days later, which were over 5000. This is the highest number of views she has ever gotten on her articles.



## Main Data Items and Entities

**Users :** Users will be referring to the set of people who will use our application. Users will be given wide access to features like searching news, access to live Twitter feeds, following live games, posting opinions in the discussion forum of a game, and viewing player statistics. Users will be characterized in two ways : Registered and Unregistered.

**Account :** Each registered user will have an account that contains all the details and keys that are required to identify a user.

**News :** Since our website involves collecting and giving information to users, we will refer to this information as news.

**Games :** Most of the information in our website will be related to games, and most of our features deal with sport's games.

**Backend Service :** The backend service will relate to all the services that perform operations from an external source.

**Website :** Since our product is a web application, we will refer to it as a website.

**Group :** Group will refer to all the private group chats and discussion forums that allow users to interact with other users.

**Message :** Message will refer to the content a user provides to other users in a group.

**Database :** Database will refer to the system that stores all the information our website requires.

## Functional Requirements

### Registered Users

1. Registered users shall be able to log in to the website.
2. Registered users shall be able to access the homepage.
3. Registered users shall be able to filter the news they would like to follow according to teams.
4. Registered users shall be able to filter the news they would like to follow according to players.
5. Registered users shall be able to filter the news they would like to follow according to different types of sports.
6. Registered users shall be able to choose whether they want to enable receiving emails that contain sports news.
7. Registered users shall be able to choose at what rate they want to receive emails that contain sports news.
8. Registered users shall be able to see live Twitter feeds.
9. Registered users shall be able to type the topics they would like to follow on Twitter.
10. Registered users shall be able to remove topics they are currently following.
11. Registered users shall be able to receive emails of any new tweets.
12. Registered users shall be able to view updated statistics of players.
13. Registered users shall be able to follow sports games.
14. Registered users shall be able to enable receiving notifications by email.
15. Registered users shall be able to post text in the discussion forum of the game.
16. Registered users shall be able to unfollow games they are currently following.
17. Registered users shall be able to search for specific comments in the discussion forum.
18. Registered users shall be able to mark posts to be read so that they don't receive notifications.
19. Registered users shall be able to see the live scores of a game.
20. Registered users shall be able to receive responses to their posts in the discussion forum.
21. Registered users shall be able to see the duration of a game..
22. Registered users shall be able to comment on articles.
23. Registered users shall be able to mark as read on articles.

- 
- 24. Registered users shall be able to search for articles through a search bar.
  - 25. Registered users shall be able to reset their password.
  - 26. Registered users shall be able to change their profile pic.
  - 27. Registered users shall be able to delete their account.
  - 28. Registered users shall be able to view profiles of other users.
  - 29. Registered users shall be able to choose if they want to enable receiving sms that contain sports news.
  - 30. Registered users shall be able to invite their friends to use the YourSports website.
  - 31. Registered users shall be able to be redirected to a live broadcast of the game.
  - 32. Registered users shall be able to delete their posts in the discussion forum.
  - 33. Registered users shall be able to switch between light and dark mode.
  - 34. Registered users shall be able to subscribe to a premium version of YourSports website.
  - 35. Registered users shall be able to enable receiving notifications by messages on phone.
  - 36. Registered users shall be able to red flag other registered users for any posts in the discussion forum.
  - 37. Registered users shall be able to send messages in the group chat.
  - 38. Registered users shall be able to receive messages in the group chat.
  - 39. Registered users shall be able to view messages in the group chat.
  - 40. Registered users shall be able to create a group chat.
  - 41. Registered users shall be able to see other online users in the group chat.
  - 42. Registered users shall be able to delete messages they previously sent.
  - 43. Registered users shall be able to edit messages they previously sent.
  - 44. Registered users shall be able to bet on live and upcoming games.
  - 45. Registered users shall be able to get YourSports coins by signing up.
  - 46. Registered users shall receive 100 YourSports coins when signing up.
  - 47. Registered users shall get cash by selling their YourSports coins.
  - 48. Registered users shall win YourSports coins if they win a bet.
  - 49. Registered users shall lose YourSports coins if they lose a bet.
  - 50. Registered users shall be able to buy YourSports coin using actual money.
  - 51. Registered users shall be able to view their current number of YourSports coins in their account.

## Unregistered Users

- 52. Unregistered Users shall be able to create an account in our application.
- 53. Unregistered Users shall be able to access the homepage.

- 
- 54. Unregistered Users shall be able to search games without signing up.
  - 55. Unregistered users shall be able to see live twitter feeds.
  - 56. Unregistered users shall be able to search for specific comments in the discussion forum.
  - 57. Unregistered users shall be able to see the live scores of a game.
  - 58. Unregistered users shall be able to see the duration of a game.
  - 59. Unregistered users can sign up as a creator.
  - 60. Unregistered users shall be able to be redirected to a live broadcast of the game.
  - 61. Unregistered users shall be able to switch between light and dark mode.

## Backend Service

- 62. The backend service shall be able to validate the user email by checking if it is unique when they try to sign up.
- 63. The backend service shall be able to check accounts with the same username when the unregistered user tries to sign up.
- 64. The backend service shall be able to create an account in the database.
- 65. The backend service shall be able to store messages sent by a user.
- 66. The backend service shall be able to filter news according to the filters specified by the user.
- 67. The backend service shall be able to filter tweets according to the topics a user follows.
- 68. The backend service shall be able to create a new forum for every upcoming game.
- 69. The backend service shall be able to fetch updates from a game every 5 seconds.
- 70. The backend service shall be able to send a verification email to the user's email so that the user can verify their account.

## Website

- 71. The website shall be able to check if the first name is at least one character before the Unregistered user tries to sign up.
- 72. The website shall be able to check if the last name is at least one character before the Unregistered user tries to sign up.
- 73. The website shall be able to check if the email is at least one character before the Unregistered user tries to sign up.
- 74. The website shall be able to check if the password is at least 8 characters before the Unregistered user tries to sign up.

- 
- 75. The website shall be able to check if the difference between present day and date of birth is at least 13 years before the Unregistered user tries to sign up.
  - 76. The website shall be able to check if the terms of service has been accepted before the Unregistered user tries to sign up.

## Creators

- 77. Creators shall have the same privileges as the Registered users.
- 78. Creators shall be able to upload articles to the website..
- 79. Creators shall be able to select their target audience.
- 80. Creators shall be able to view Dashboard.

## Dashboard

- 81. Dashboard shall be able to show the number of views of the selected article.
- 82. Dashboard shall be able to show the comments on the article.
- 83. Dashboard shall be able to show the redflags of an article.

## Group Owner

- 84. The Group's Owner shall be able to change the group chat name.
- 85. The Group's Owner shall be able to invite other users to the group chat.
- 86. The Group's Owner shall be able to block any users in the group chat.

## Non-functional Requirements

### System Requirements

1. The system shall be hosted on a Google Cloud Compute Engine instance of 1 cpu core, 1gb RAM and 10GB storage size.
2. A MySQL relational database shall be hosted in Google Cloud to store the data.
3. NodeJS and express shall be used to build the backend server.
4. React shall be used to host the front-end server.
5. The front-end service shall run on port 3000.
6. The back-end service shall run on port 8080.
7. The Google Cloud compute engine shall use an Ubuntu 18.04.6 LTS.
8. The code repository shall be hosted in Github.
9. The front-end and backend service shall accept traffic from all ip-addresses.
10. The Virtual Machine hosted in Google Cloud shall allow both HTTP and HTTPS traffic.
11. The website shall be resizable and compatible across different screen sizes and operating systems.

### Performance Requirements

12. If a registered user enters the wrong password, the system shall respond within 3 seconds of submitting their details for signup.
13. All inserts to the database should happen within 5 seconds.
14. The homepage shall open within 10 seconds.
15. Search Results should be displayed within 5 seconds.
16. The website shall show updates in a game every 5 seconds.

### Storage, Security and Environmental Requirements

17. Unregistered Users shall only be able to register if their username is unique.
18. Unregistered Users shall only be able to register if their email is unique.
19. The statistics of the player will be updated every 24 hours.
20. The YourSports API shall not accept more than 10 requests per 10 seconds from the same IP address.
21. Passwords of registered users shall be encrypted with MD5 hash function.

## Marketing and Legal Requirements

22. Unregistered Users shall be able to accept the terms of service before signing up.
23. Articles uploaded by a creator user, shall need to follow community guidelines provided by the website.

## Content

24. The password shall be hidden when typed when trying to sign up.
25. The password shall be a minimum of 8 characters when trying to sign up.
26. The registered user shall be redirected to the homepage if the user has logged in.
27. Registered Users shall be prompted to log in if an account with the same email already exists.
28. Unregistered users shall have to fill in the name with at least one character while signing up.
29. Unregistered users shall have to fill in the username with at least one character while signing up.
30. Unregistered users shall have to fill in the email with at least one character while signing up.
31. Unregistered users shall have to fill in the password with at least one character while signing up.
32. If a Registered User has not put any filters in the news articles they are searching for, they shall receive all the news articles in the database.
33. Posts in the discussion forum shall have a minimum 1 character limit.
34. Comments in the Article view shall have at least one character.
35. The website logo shall be on the top left of every page.
36. Text inputs should have a minimum of 1 character and the input should be trimmed before processing.

## Privacy

37. The email, username and password shall be collected and only used for authentication.
38. Registered users shall have their log in details stored.

## Competitive Analysis

Feature	Yahoo! Sports	Thescore	Flashscore	ESPN Sport	BBC Sport
Strengths	Have a good and wide range of sports news, the site's visual presentation is precise and attractive. Navigation bars have most sports types as individual tabs and will extend into more detailed categories when the cursor hovers on it. Scoreboards located right below the Nav-bar, and the most attractive news and information are enlarged and grouped on the top of the site. Also changes quick access teams according to your location data.	News available on a variety of sports, Easy to use, Easy to access information on latest games	live scores and lineup updated in real time based on different categories, live commentary by text, detailed statistics, odds comparison, Less ads	One bus stop for multiple news platform including SEC network, Access to ESPN insider, Live update of different sports on home page, Video highlights of random sport games, Cable channels, Search bar, Can stream games	Has lots of news and blog post about different sports, It also uses social media to advertise its app.
Weaknesses	Advertisements are everywhere and distracting. Too much news that is not directly related to the sports or players like the comments of some politicians and celebrities. The site advertises their own betting system but with very technical terms which makes it hard to understand.	No personalization for users, A lot of advertisements on the website, Bad user interface, No Subscription feature	Favorite team limit 100, Small available date range, Some sports updates are not accurate	False advertisement of service, Bias commentary, one sided coverage, Hard to navigate and glitches, Poor service review by users	Is not organized, Some of the news and blog post are not updated, Has lots of adds
Pricing	Generally free, can add money to fantasy betting through Paypal	No subscription fees, All revenue generated is from advertisements	Free, bet using one of advertised sportsbook partners	Free for guest/unregistered users	No Subscription/ Free
Social Media	Facebook, Twitter	Facebook, Twitter, Instagram	Facebook, Twitter, Instagram	Facebook, snapchat, twitter, instagram	Twitter, Facebook, Instagram, Youtube
Onboarding Experience	Most information on site is very straightforward. Just have to follow the names on the navigation bar and you will be able to find information that you are looking for. However, there will be information that may be too technical for newcomers to understand. There is also no tutorial for the fantasy betting system, and the betting system is very confusing without any explanation.	No support from the website, A moderate number of steps are required to understand the website.	Easy to find info, Seamless onboarding	Overwhelming with multiple random sports updates, highlights. Not easy to navigate	Doesn't help you with your first step, Really hard to navigate, Takes a while to get used to.



Feature	Yahoo! Sports	TheScore	Flashscore	ESPN sport	BBC Sports	YourSport
Access to Live Twitter Feeds	+	-	-	-	-	+
Search by filter for news articles	++	-	-	+	+	+
Discussion Forum	-	-	-	-	-	++
Text Commentary	-	+	+	+	++	++
Access to Player Statistics	+	-	++	++	-	+
Fantasy Betting	+	-	+	++	+	+

Does not exist: -    Exist: +    Superior: ++

With the above table, we can see that a lot of competitors in the industry have text commentary, and information on player statistics. Most of them even offer news to their users. However, most of the competitors rely on journalists and news articles that are not intended for most of the users. Our focus is to use the same features, but with an emphasis for personalization of the user, where they can access all the information they want to read from Twitter, and news articles that cater to their interests. Lastly, we can see that a few competitors allow sports betting, however our betting system takes a more jovial approach, where users bet for fun and experience, without losing any money. The focus of our fantasy betting feature is to keep users engaged the same way they are engaged when betting with real money, and hence disrupt the betting industry. The lack of features and functionalities from our competitors like no discussion forums and no access to live Twitter feeds is where our product comes in, and serves as a competitor in the market.



## Technology Specifications

### Server Host

Google Compute Engine e2-micro (2vCPU)

### Memory Limit

1GB Memory

### Storage Size

10GB

### Operating System

Ubuntu 18.04.6 LTS

### Database Management

MySQL 8.0

### Server Environment

Node.js 14.0.0

### BackEnd Framework

express 4.18.1 (JavaScript)

### FrontEnd Framework

React

### IDE

Visual Studio Code

### Web Analytics

Google Analytics



## Checklist

Team found a time slot to meet outside of the class. **(Done)**

Github master chosen. **(Done)**

Team decided and agreed together on using the listed SW tools and deployment server. **(Done)**

Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing. **(Done)**

Team lead ensured that all team members read the final M1 and agree/understand it before submission. **(Done)**

Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.). **(Done)**

## List of Team Contributions

### Kshitiz Sareen (Team Lead, Backend Lead)

Installed and Hosted the MySQL database on Google Cloud.

Installed the Node JS server on the Google Cloud Virtual Machine.

Hosted the React front-end Application on the Google Cloud Virtual Machine.

Did research on Thescore for the competitive analysis.

### Kevin Islas (Front-end Lead)

Contributed in team meetings by building use cases.

Helped in finalizing the functional and non-functional requirements.

Created the react environment for the front-end application.

### Shamar Ireland (Front-end Developer)

Helped develop the team website by creating the buttons that point to each team members about page.

Created the template that other team member's used for their personal about page.

Helped in building the functional and non-functional requirements.

### Wenye Guo (Database Administrator)

Helped with the use cases and requirements.

Conducted research on Flashscore competitor for the competitive analysis.

### Sabur Saigani (Functional Lead)

Contributed in building the documentation.

Helped in building the use cases.

Helped in creating the functional and non-functional requirements.

Did Research on BBC Sport for the competitive analysis.

### Mathew O Abiola (Github Master)

Contributed by building use cases in team meetings.

Did research on ESPN for competitive analysis.

### Jonathan Ip (Functional Assistant)

Contributed by formatting and paging the documentation.

Did research on Yahoo Sports for the competitive analysis.

# YourSports

Dynamic Debugger (Team 02)

---

## Team & Backend Lead

Kshitiz Sareen: [ksareen@sfsu.edu](mailto:ksareen@sfsu.edu)

## Frontend

Lead Kevin Islas

Developer Shamar Ireland

## Functional

Lead Sabur Saigani

Assistant Jonathan Ip

## Database Administrator

Wenye Guo

## Github Master

Mathew O Abiola

## Milestone 2

Milestone	Date
M2V2	July 14th, 2022
M2V1	July 7th, 2022
M1V2	June 30th, 2022
M1V1	June 9th, 2022

## Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Data Definition</b>	<b>3</b>
Users	3
Regular User	3
Registered User	3
Account	3
Basic Account	4
Creator Account	4
Admin Account	4
News	5
Tweets	5
Articles	5
Comment	6
Games	7
Group	7
Discussion Forum	7
Group Chat	8
Message	8
Dashboard	9
Backend Service	9
Website	9
Database	9
<b>Prioritize Functional Requirements</b>	<b>10</b>
Priority 1	10
Priority 2	13
Priority 3	14
<b>UI Mockup and Storyboards</b>	<b>17</b>
Signing Up	17
Checking Sports news	18
Posting news articles	19
Following Sport Game	20
Checking Player Statistics	21
<b>Database Architecture and Organization</b>	<b>22</b>

Database Organization	22
Regular User	22
Registered user	22
Games	22
Discussion Forum	22
Admin Account	22
Business Rules	22
Entities	23
Entity Relationship Diagram (ERD)	25
Database Model	26
Database that will be used	26
Media Storage	26
Search architecture and implementation	27
<b>High-Level API and Main Algorithm</b>	<b>29</b>
YourSports API	29
<b>UML Diagram</b>	<b>30</b>
<b>Application Network and Deployment Diagram</b>	<b>31</b>
Application Network Diagram	31
Deployment Diagram	32
<b>Current Key Risks</b>	<b>33</b>
Skills	33
Schedule	33
Technical	33
Teamwork	34
Legal/Content	34
<b>Project Management</b>	<b>35</b>
<b>List of Team Contributions</b>	<b>36</b>

## Data Definition

### Users

Users will be referring to the group of people who use our application. Users will be given wide access to features like searching news, access to live Twitter feeds, following live games, posting opinions in the discussion forum of a game, and viewing player statistics.

There will be two types of users in our application:

#### Regular User

These are the users who have not logged in. Regular users will only be able to watch scores of live games and search for news and tweets.

Regular Users will have the following attributes:

##### Tracking ID

> This refers to a unique id given to the unregistered user.

##### IP address

> This refers to the IP address of the device that the user uses to access the website.

#### Registered User

These are users who have an account. Registered users will have access to all features our application provides. Registered users inherit from regular users.

Registered users will have the following attributes:

##### User

> This refers to the tracking ID of the user.

##### Account

> This refers to the account of the user.

### Account



Each registered user will have an account that contains all the details and keys that are required to identify a user.

There will be three types of accounts in our application:

### Basic Account

A Basic account will have access to features like searching news, access to live Twitter feeds, following live games, posting opinions in the discussion forum of a game, and viewing player statistics.

The Basic Account has the following attributes:

#### Name

> This refers to the name that the user used to sign up, and it is usually the legal name of the user.

#### Username

> This refers to the username that the user used to sign up for, and it will be used to distinguish a user from other users.

#### Email

> This refers to the email that the user used to sign up. The email will be used to associate their account with their email, and also serve as a communication method between the user and the application.

#### Password

> This refers to the set of characters that a user used to sign up, and it will be used to authenticate a user along with their email.

### Creator Account

A Creator Account will have access to all the privileges and properties that a basic account has with the additional privilege of publishing articles to the website. Every user that has a creator account will have a basic account. A creator account will inherit from a basic account.

The creator account will have the following properties:

#### Account

> This refers to the account of the user.

### Admin Account



An Admin account is used to regulate and ensure proper usage of the application by other accounts. An Admin account has all privileges in the application along with access to the database and backend service to manipulate, change, add and remove data. There will only be one admin account. The admin account will also inherit from the basic account.

The admin account has the following properties:

#### Account

- > This refers to the account of the user.

## News

News will refer to the information that users can access from our application.

Our application will provide different types of news:

### Tweets

These are the tweets collected from Twitter according to users' preferences and interests.

Tweets will have the following properties:

#### Username

- > This is the username of the user who posted the tweet. The username will follow twitter data conventions.

#### Content

- > This is the content of the tweet.

### Articles

These are the articles posted by users who have a creator account. Articles can be searched and filtered by date posted and which sport is it used to talk about.

Articles have the following properties:

#### Post Time

- > This is the date and time the article was posted.

#### Image

> This is the image that will be used as the thumbnail for the article. The acceptable image formats are png and jpeg, with a maximum size of 5 MB.

### Heading

> This is the title of the article.

### Subheading

> This is the subheading of the article.

### Introduction

> This is the introduction part of the article.

### Content

> This is the main content that contains all the detailed information of the article.

### Conclusion

> This is the conclusion of the article.

### Author

> This is the Author of the article.

### Sport

> This is the type of sport which the article targets. E.g. Basketball and Baseball.

## Comment

Registered users will be able to post comments on articles to express their views and opinions on the information given in the article.

Comment will have the following properties:

### Article

> This is the Article for which the comment was posted.

### Author

> This is the user who has posted the comment.

### Post Time



> This is the date and time where the comment was posted.

### Content

> This contains all the information that a user wants to convey.

## Games

This refers to the past, current, and future sports games that users can follow. Users can follow the scores, and post any opinions they have about the game.

Games will have the following properties:

### Player One Score

> This is the score of the home team.

### Player Two Score

> This is the score of the opponent team.

### Game Location

> This is the location of the game.

### Team One

> This is the home team.

### Team Two

> This is the away team.

## Group

This refers to all the groups like group chats and discussion forums that allow users to interact with other users.

There will be different types of groups:

### Discussion Forum

The discussion forum will refer to the forums of every sports game where users can post their opinions about the game. This discussion forum will have no owner and every registered user who follows a game will be able to join the discussion forum.

Discussion forums will have the following properties:



### Forum Name

> This is the name of the forum.

### Game

> This is the sports game for which the forum has been created.

### Account List

> This is the list of all the accounts of the users that are part of the forum.

## Group Chat

The group chat will refer to private groups of sports fans who share similar interests. The private group chats will have an owner and every registered user will only be able to join a private group chat upon the owner's approval.

Group chats will have the following properties:

### Group Chat Name

> This is the name of the group chat.

### Group Chat Owner

> This is the user who owns the group.

### Group Account

> This is the list of all the accounts of the users that are part of the group chat.

## Message

This refers to the message a user posts in a group. All communication between users of the application will happen between the users.

Message has the following properties:

### Account

> This is the account of the user who has posted the message.

### Content

> This is the content of the message.



## Group

> This is the group in which the user has posted a message.

## Dashboard

This refers to the Dashboard of every user that has a creator account.

Dashboard has the following properties:

### Article List

> This is the list of all the articles that the creator has uploaded.

### Content

> This is the number of views for each article.

### Creator

> This is the creator account of the user who owns the dashboard.

## Backend Service

The backend service will be used frequently to refer to all the operations that will happen on the server-side like filtering and searching for articles, filtering and retrieval of tweets, fetching live updates from games, and handling messages from users, updating and retrieving player statistics.

## Website

The Website will be used frequently to refer to all the operations that will happen on the client-side like signing up and logging in, and verification and validation of data.

## Database

The database will be used frequently to refer to all the operations that involve the storage of data. It will be used to store the news, account information of users, information on sports games, information on players, and all the messages that a user posts in a group.

## Prioritize Functional Requirements

### Priority 1

#### Regular user

1. Regular users shall be able to create an account.
2. Regular users shall be able to sign up as a creator account.
3. Regular users shall be able to access the homepage.
4. Regular users shall be able to view the live scores of a game.
5. Regular users shall be able to view the duration of a game.
6. Regular users shall be able to view updated statistics of players.
7. Regular users shall be able to search for articles through a search bar.
8. Regular users shall be able to filter articles they would like to read by keywords.
9. Regular users shall be able to filter articles they would like to read by different types of sports.
10. Regular users shall be able to search articles by heading of the article.
11. Regular users shall be able to view the article heading in the search results for articles.
12. Regular users shall be able to view the article author in the search results for articles.
13. Regular users shall be able to view the article postdate in the search results for articles.
14. Regular users shall be able to view the article image as a thumbnail in the search results for articles.
15. Regular users shall be able to search games by the teams playing in the search bar.
16. Regular users shall be able to filter games by the sport type.
17. Regular users shall be able to filter games by their date.
18. Regular users shall be able to view the game score in the search results for games.
19. Regular users shall be able to view the game date in the search results for games.
20. Regular users shall be able to view the game location in the search results for games.

- 
- 21. Regular users shall be able to search for player statistics according to the player name in the search bar.
  - 22. Regular users shall be able to filter player statistics by the sport type.

### Registered user

- 23. Registered users shall have the same privileges as regular users.
- 24. Registered users shall be able to log in.
- 25. Registered users shall be able to log out.
- 26. Registered users shall be able to reset their password.
- 27. Registered users shall be able to delete their account.
- 28. Registered users shall be able to post in the discussion forum of the game.
- 29. Registered users shall be able to delete their posts in the discussion forum.
- 30. Registered users shall be able to comment on articles.

### Basic Account

- 31. Accounts shall be able to be deleted by users.
- 32. Accounts shall contain the name of the users.
- 33. Accounts shall contain the username of the users.
- 34. Accounts shall contain the email of the users.
- 35. Accounts shall contain the password of the users.

### Creator Account

- 36. Creators shall have the same privileges as registered users.
- 37. Creators shall be able to upload articles to the website.
- 38. Creators shall be able to view Dashboard.

### Admin Account

- 39. Admin shall have all the privileges.
- 40. Admin shall be able to create discussion forums for upcoming games.

### Articles

- 41. Articles shall be posted by creator users.
- 42. Articles shall be able to be filtered by date.
- 43. Articles shall be able to be filtered by sports type.
- 44. Articles shall be able to be filtered by keywords specified by the user.
- 45. Articles shall contain a heading.
- 46. Articles shall contain a subheading.
- 47. Articles shall contain an introduction.
- 48. Articles shall contain contents.

49. Articles shall contain a conclusion.

50. Articles shall contain an image.

## Games

51. Games shall be filtered by date.

52. Games shall be filtered by sport type.

53. Games shall be filtered by keywords specified by users in the search bar.

54. Games shall have an individual discussion forum.

55. Games shall be able to show live scores of corresponding teams.

56. Games shall be able to show the date of the game.

57. Games shall be able to show the location of the game.

## Discussion Forum

58. Discussion forum shall be able to be viewed by all users.

59. Discussion forum shall be able to show posts from users.

## Dashboard

60. Dashboard shall be able to show the number of views of the article.

61. Dashboard shall be able to show the heading of the article.

62. Dashboard shall be able to show the posted date of the article.

## Backend Service

63. The backend service shall be able to validate the email for duplicates when users try to sign up.

64. The backend service shall be able to check for duplicate accounts when users try to sign up.

65. The backend service shall be able to create an account in the database.

66. The backend service shall be able to store posts by a user in the discussion forums.

67. The backend service shall be able to update player statistics in the database.

68. The backend service shall be able to filter articles specified by the user.

69. The backend service shall be able to filter games specified by the user.

70. The backend service shall be able to filter player statistics specified by the user.

71. The backend service shall be able to fetch updates from a game every 5 seconds.

## Website

- 
- 72. The website shall be able to check if the first name is not empty users try to sign up.
  - 73. The website shall be able to check if the last name is not empty when a user try to sign up.
  - 74. The website shall be able to check if the email is not empty when users try to sign up.
  - 75. The website shall be able to check if the password is at least 8 characters before the users try to sign up.
  - 76. The website shall be able to check if the date of birth is at least 13 years old when users try to sign up.
  - 77. The website shall be able to check if the terms of service has been accepted when users try to sign up.

#### Comment

- 78. Comments shall be able to be viewed by regular users.
- 79. Comments shall be posted by registered users.
- 80. Comments shall be able to be deleted by poster.

## Priority 2

#### Regular user

- 1. Regular users shall be able to view the player statistics through a pie chart.
- 2. Regular users shall be able to be redirected to a live broadcast of the game.
- 3. Regular users shall be able to search for specific comments in the discussion forum.

#### Registered user

- 4. Registered users shall be able to like an article.
- 5. Registered users shall be able to dislike an article.
- 6. Registered users shall be able to change their profile picture.
- 7. Registered users shall be able to view profiles of other users.
- 8. Registered users shall be able to red flag other registered users for any posts in the discussion forum.
- 9. Registered users shall be able to follow sports games.
- 10. Registered users shall be able to unfollow games they are currently following.
- 11. Registered users shall be able to search for specific messages in the discussion forum.

12. Registered users shall be able to mark posts as read so that they don't receive notifications.

13. Registered users shall be able to mark as read on articles.

14. Registered users shall be able to receive responses to their posts in the discussion forum.

### Creator Account

15. Creator shall be able to view the article statistics through a pie chart.

16. Creators shall be able to see the number of likes an article has received.

17. Creators shall be able to see the number of dislikes an article has received.

18. Creators shall be able to select their target audience.

### Dashboard

19. Dashboard shall be able to show the comments on the article.

20. Dashboard shall be able to show the redflags of an article.

### Backend Service

21. The backend service shall be able to send a verification email for users to verify their account.

### Discussion forum

22. Discussion forum shall be sorted with specific words.

## Priority 3

### Regular user

1. Regular users shall be able to read live twitter feeds.

2. Regular users shall be able to switch between light and dark mode.

### Registered user

3. Registered users shall be able to bet on live and upcoming games.

4. Registered users shall receive 100 YourSports coins when signing up.

5. Registered users shall get cash by selling their YourSports coins.

6. Registered users shall win YourSports coins if they win a bet.

7. Registered users shall lose YourSports coins if they lose a bet.

8. Registered users shall be able to buy YourSports coin using actual money.

9. Registered users shall be able to view their current amount of YourSports coins in their account.

- 
10. Registered users shall be able to enable/disable receiving emails about sports news.
  11. Registered users shall be able to choose how frequently they receive emails about sports news.
  12. Registered users shall be able to read live Twitter feeds.
  13. Registered users shall be able to type the topics they would like to follow on Twitter.
  14. Registered users shall be able to unfollow topics they are currently following.
  15. Registered users shall be able to receive emails of any new tweets.
  16. Registered users shall be able to enable receiving notifications by email.
  17. Registered users shall be able to choose if they want to enable receiving sms that contain sports news.
  18. Registered users shall be able to enable receiving notifications by messages on the phone.
  19. Registered users shall be able to invite their friends to use the YourSports website.
  20. Registered users shall be able to send messages in the group chat.
  21. Registered users shall be able to receive messages in the group chat.
  22. Registered users shall be able to view messages in the group chat.
  23. Registered users shall be able to see other online users in the group chat.
  24. Registered users shall be able to delete messages they previously sent in the group chat.
  25. Registered users shall be able to edit messages they previously sent in the group chat.
  26. Registered users shall be able to create a group chat.

#### Group Owner

27. The Group's Owner shall be able to change the group chat name.
28. The Group's Owner shall be able to invite other users to the group chat.
29. The Group's Owner shall be able to block any users in the group chat.

#### Message

30. Messages shall be able to be edited by poster.

#### Tweets

31. Tweets shall be pushed to users.
32. Different topics of tweets shall be pushed to users.

#### Notification

- 
- 33. Notification shall be sent by email.
  - 34. Notification shall include new tweets or/and sport news.
  - 35. Notifications shall be sent by messages.

#### Backend service

- 36. The backend service shall be able to filter tweets by the topics a user follows.

#### Group chat

- 23. Group chat shall contain multiple group members.
- 24. Group chat shall have one group owner.
- 25. Joining a specific group chat shall require permission of the group owner.
- 26. Group members of a group chat shall be able to post messages.

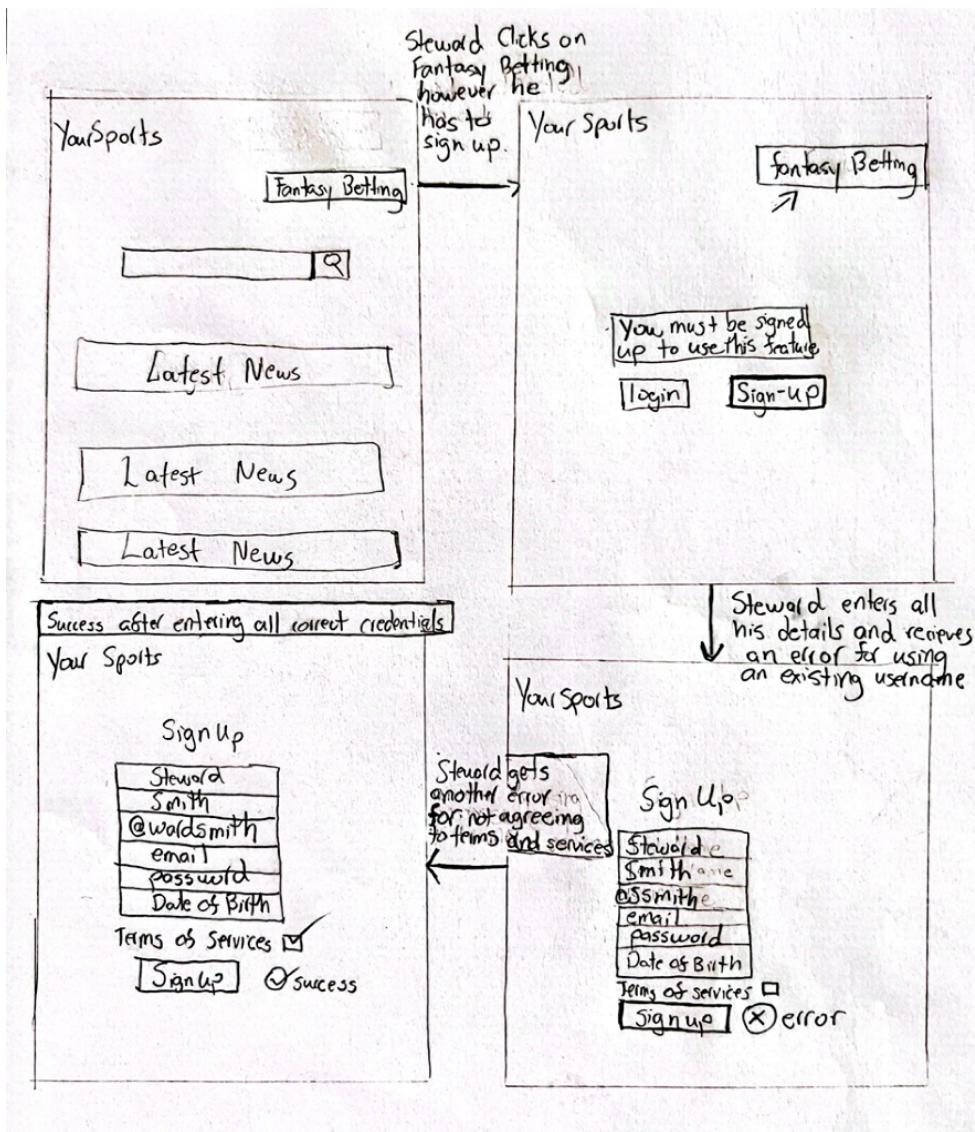
#### Message

- 27. Messages in the group chat shall be posted by registered users.
- 28. Messages in the group chat shall be able to be deleted by poster.
- 29. Messages in the group chat shall be able to be marked using a red flag.
- 30. Messages in the group chat shall be responded to by other registered users.

## UI Mockup and Storyboards

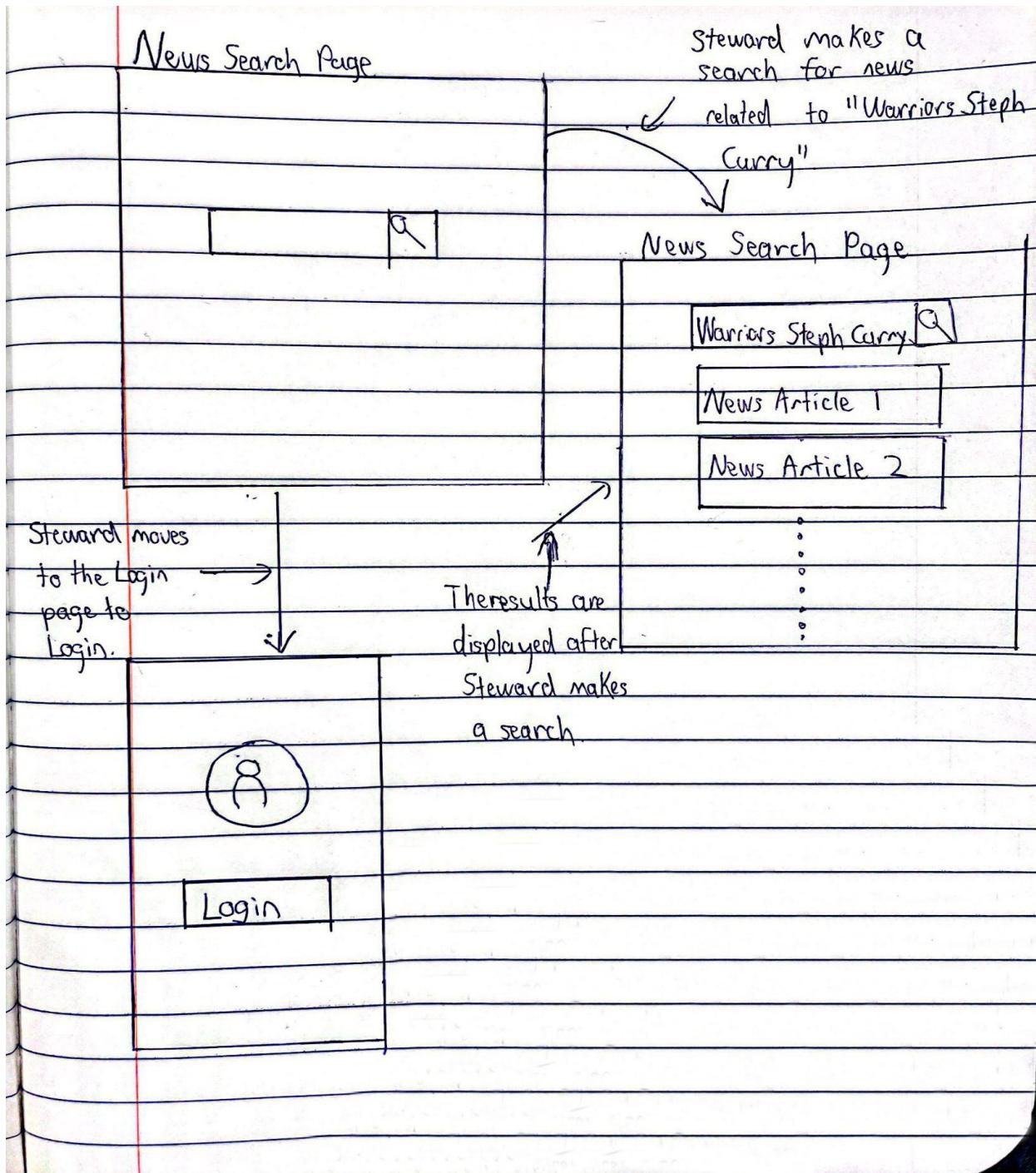
### Signing Up

Steward wants to participate in fantasy betting however he learns that he has to be signed up. He proceeds to sign up by entering all his credentials. He thinks he's in the clear but when he clicks the sign up button at the bottom of the form he gets an error. This is so because he forgot to agree to the terms and conditions and used a username that already existed. He attempts to do this again and he is successful once he enters a username that's not already taken and agrees to the terms and conditions.



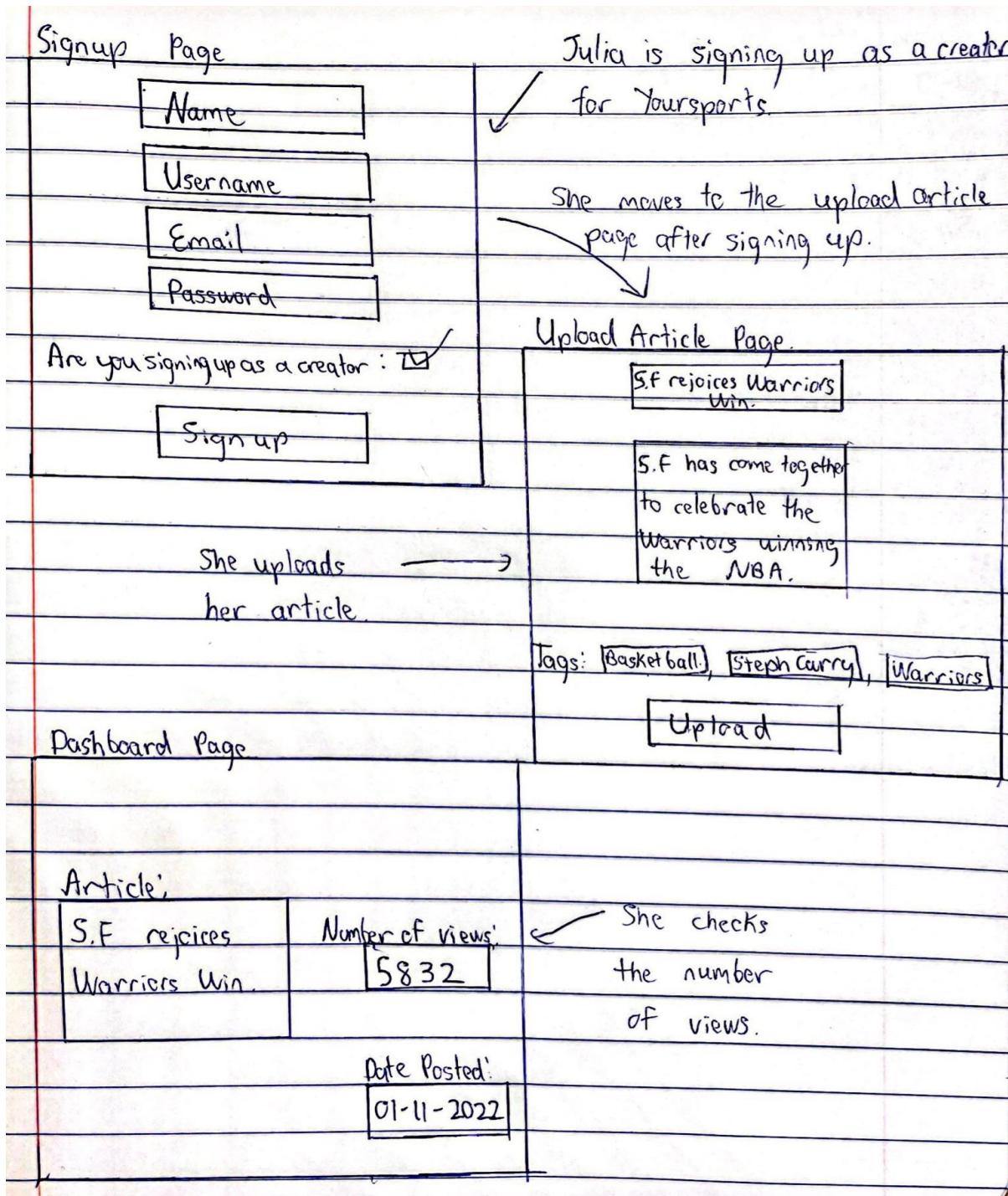
## Checking Sports news

Steward makes a search for news related to "Warriors Steph Curry". The results for all the news related to "Warriors Steph Curry" are displayed on the news search page.



## Posting news articles

Julia is signed up as a creator for YourSports. After signing up she uploads her article. While uploading her article, she sets her audience tags as Basketball, Steph Curry, Warriors. She uploads her article. After a few days, she checks the number of views on her article.

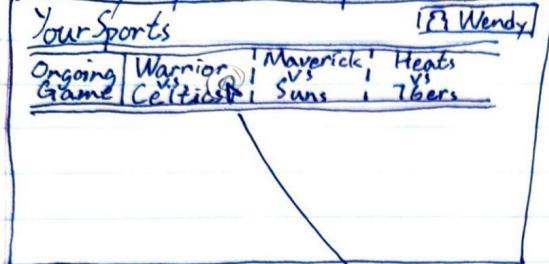


## Following Sport Game

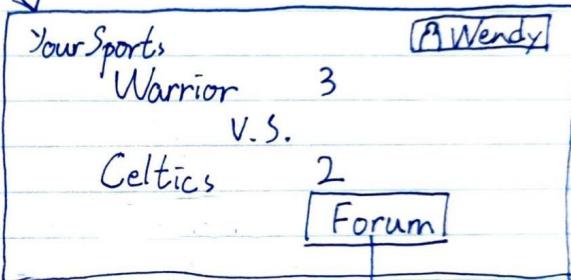
Wendy, a registered user, wants to be more involved with the Warrior vs Celtics game, so she comes to the YourSports site and opens the game's page. While she is on the site, she also went on to post her opinions on the game's forum where everyone else is also posting on.

Following Sports game

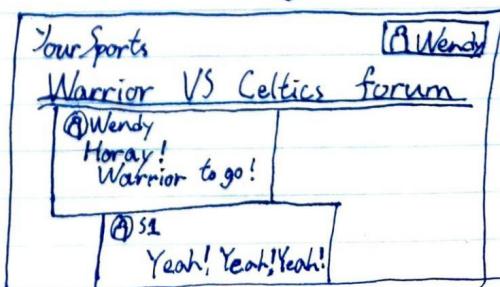
Wendy opens YourSports site



Wendy open menu that show ongoing and live games

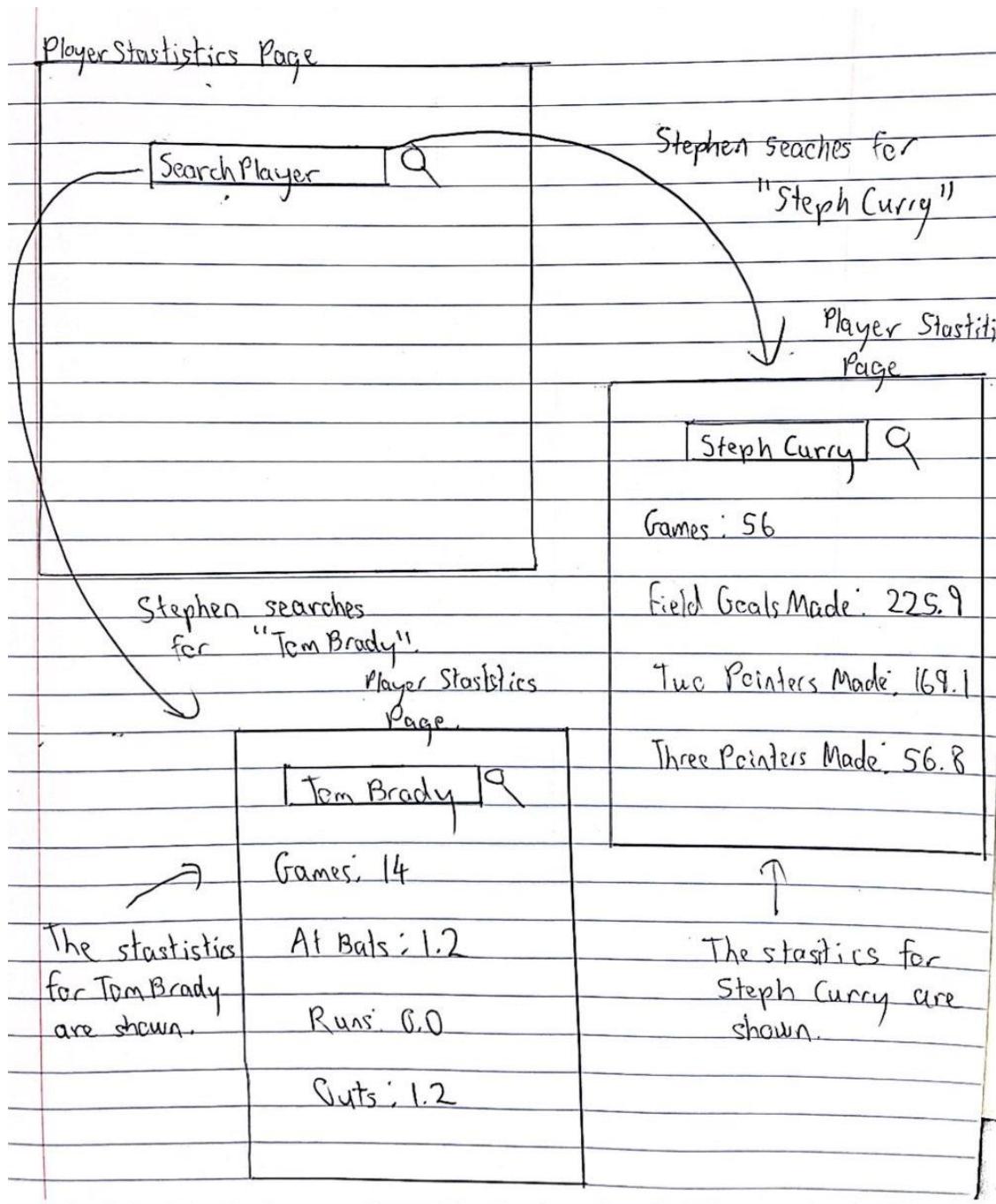


Wendy post her opinions in the discussion forum of the game



## Checking Player Statistics

John wants to check the updated statistics of Steph Curry and Tom Brady. He visits the player statistics page in the YourSports website. He first searches for Steph Curry, and sees the statistics like games Played, field goals made, two-pointers made, and three-pointers made. He then searches for Tom Brady, and views his statistics.



## Database Architecture and Organization

### Database Organization

#### Regular User

1. Regular users shall be able to view the live scores of a game.
2. Regular users shall be able to view the duration of a game.
3. Regular users shall be able to search games by the teams playing in the search bar.
4. Regular users shall be able to filter games by the sport type.
5. Regular users shall be able to filter games by their date.
6. Regular users shall be able to search for specific comments in the discussion forum.

#### Registered user

1. Registered users shall have the same privileges as regular users.
2. Registered users shall be able to post in the discussion forum of the game.
3. Registered users shall be able to delete their posts in the discussion forum.

#### Games

1. Games shall have an individual discussion forum.
2. Games shall be able to show live scores.
3. Games shall be able to show the date of the game.
4. Games shall be able to show the location of the game.

#### Discussion Forum

1. Discussion forum shall be able to be viewed by all users.
2. Discussion forum shall be able to show posts from users.

#### Admin Account

1. Admin shall be able to create discussion forums for upcoming games.

### Business Rules

1. A regular user shall be or not be a registered user.
2. A registered user shall be able to use many devices to login the account.

- 
3. A registered user shall be able to post many posts in game discussion forums.
  4. A discussion forum shall contain many posts.
  5. A post shall be able to be replied to many times.
  6. An account shall be either a creator or an admin.
  7. A game shall have at most one discussion forum under the games page.
  8. An admin shall be able to manage many discussion forums.
  9. An admin shall be able to manage many games.

## Entities

Regular User (Strong)

`Tracking_id` (INT, PK, Not null, AI)

`IP_Address` (INT, Not null)

Account (weak)

`Account_ID` (INT, PK, Not null, AI)

`Name` (VARCHAR, Not null)

`Username` (VARCHAR, Unique, Not null)

`Email` (VARCHAR, Unique, Not null)

`Password` (VARCHAR, Not null)

Admin (weak)

`Admin_ID` (INT, PK, Not null, AI)

`Account_id` (INT, Not null, Unique, FK to Account(`Account_ID`))

Registered user (Weak)

`Tracking_id` (INT, Not null, FK to Regular User(`Tracking_id`))

`Account_id` (INT, Not null, Unique, FK to Account(`Account_ID`))

Game (Strong)

`Game_ID` (INT, PK, Not null, AI)

`Game_Home_Team` (VARCHAR, Not null)

`Game_Away_Team` (VARCHAR, Not null)

`Game_Location` (VARCHAR, Not null)



Home\_Team\_Score (INT, Not null)

Away\_Team\_Score (INT, Not null)

Duration (INT, Not null)

Discussion forum (Weak)

Forum\_id (INT, PK, Unique, Not null, AI)

Game\_id (INT, Unique, Not null, FK to Game(Game\_ID))

Forum\_Name (VARCHAR, Not null)

Posts (Weak)

Post\_id (INT, PK, Unique, Not null, AI)

account\_ID (INT, Not null, FK to Account(Account\_ID))

Content (VARCHAR, Not null)

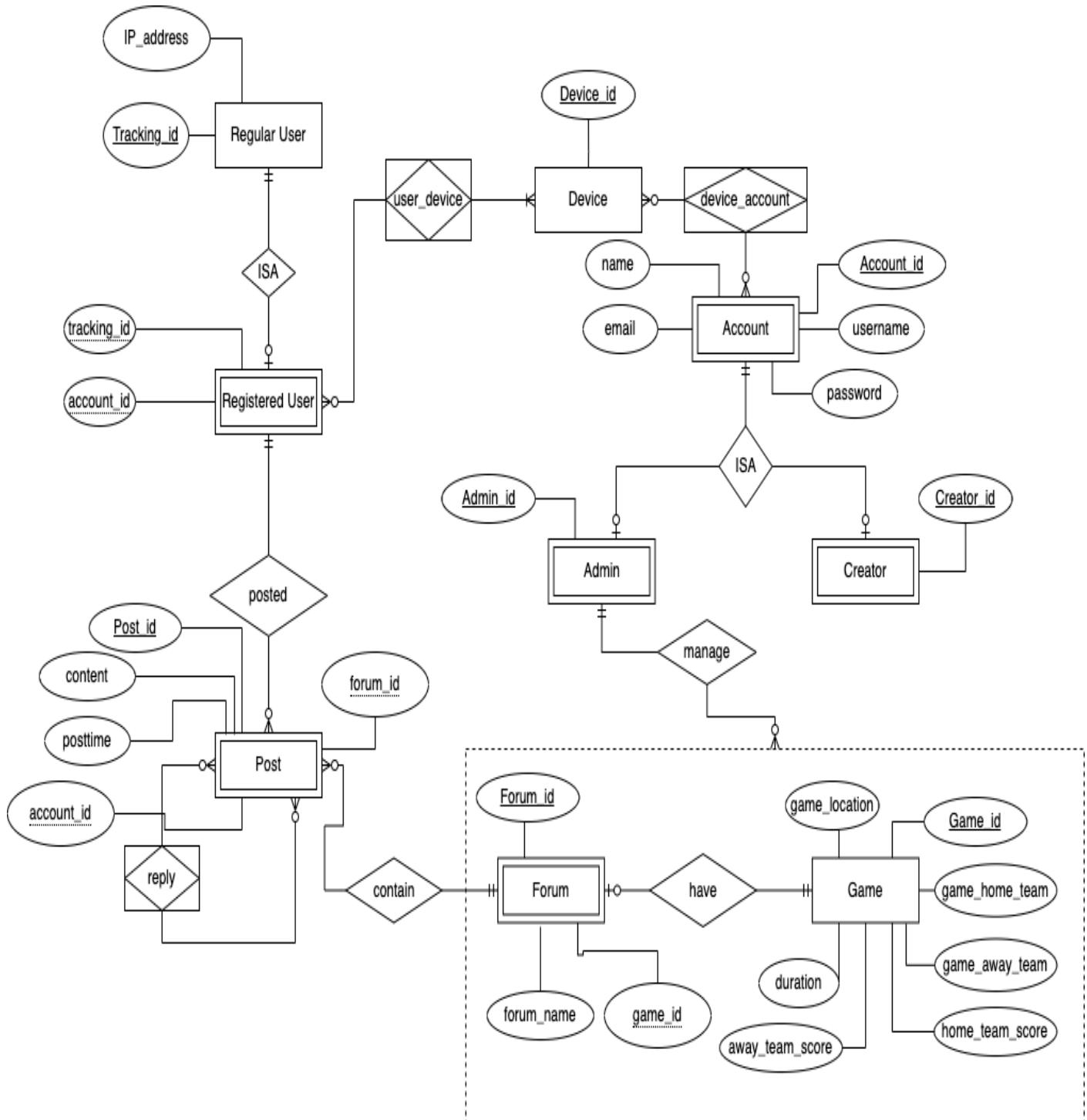
forum\_id (INT, Not null, FK to Discussion Forum(Forum\_id))

Post Time (DATE, Not null)

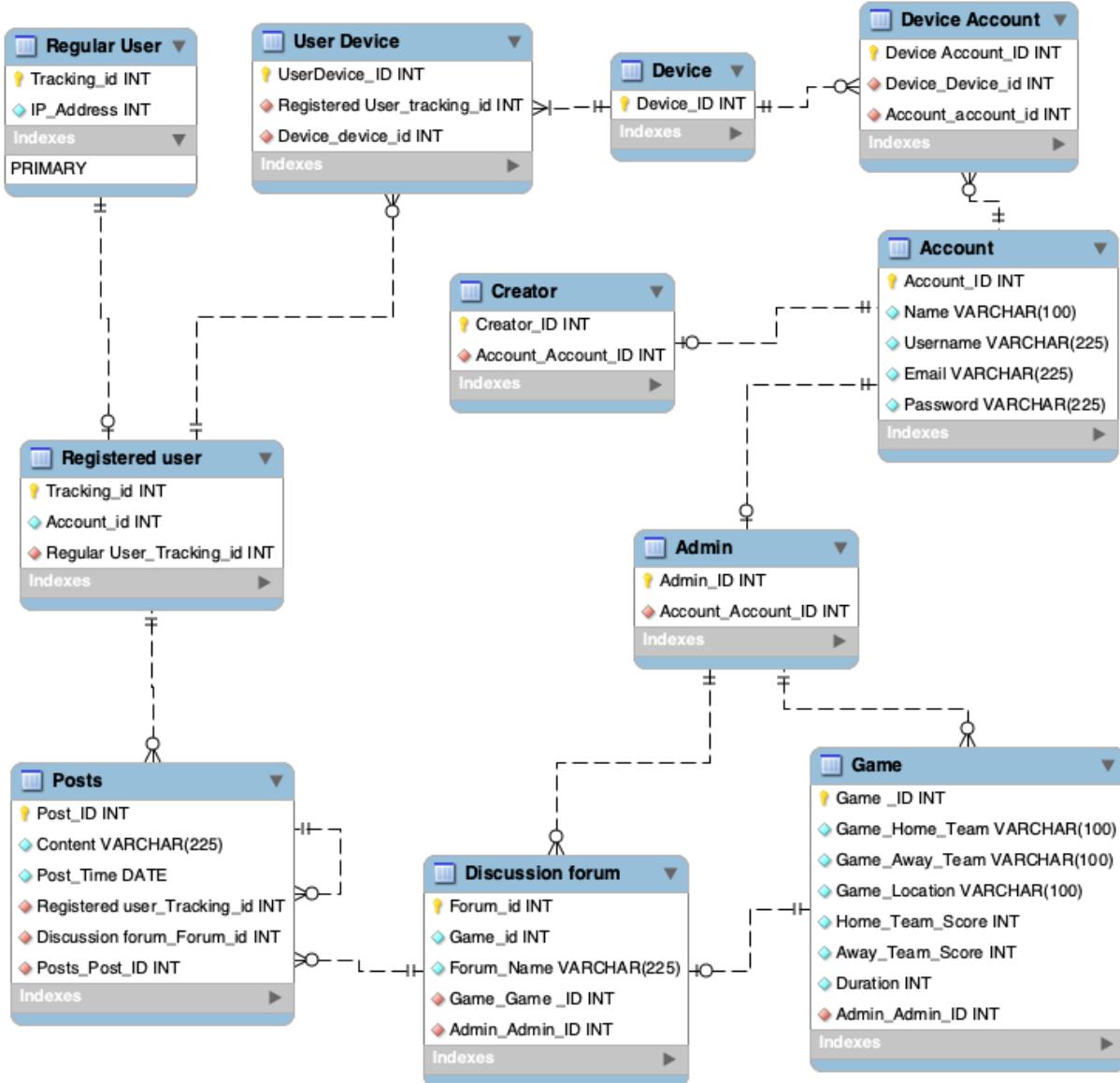
Device (Strong)

Device\_ID (INT, PK, Not null)

## Entity Relationship Diagram (ERD)



## Database Model



## Database that will be used

We will use MySQL database as it supports relational databases, and our database administrator is proficient with MySQL.

## Media Storage

Our application will only deal with images, and text formats of data. The images will be stored in Google Cloud Storage. Each image will only be in a JPEG or PNG file format, with a maximum size of 5MB. The only entity that will require images are

the Articles. Each record in the Article table will point to the URL of the image that is stored in the Google Cloud Storage bucket of our project. The Article table will consist of a field that has a datatype of VARCHAR (255). The field will store the URL of the thumbnail of the article. When the front-end requests data from the database, the URL of the image will also be returned, as it is stored in the table. The query to return the articles, that contain the URL of the image is:

```
SELECT * FROM Article, Account WHERE Article_ID = inArticleID AND fk_Author_ID = Account_ID;
```

Since we are returning everything (\*), the URL of the image will also be returned. Here is a sample record of an article thumbnail URL stored in the database.

Article_ID	PostDate	Image_url	Heading
► 42	2022-08-01	<a href="https://firebasestorage.googleapis.com/v0/b/yoursports-352701.appspot.com/o/LeBron%20James%20is%20the%20greatest%20Basketball%20Player.jpg?alt=media&amp;token=4f3a2a2d-0a2e-4a20-9a20-0a200a200a20">https://firebasestorage.googleapis.com/v0/b/yoursports-352701.appspot.com/o/LeBron%20James%20is%20the%20greatest%20Basketball%20Player.jpg?alt=media&amp;token=4f3a2a2d-0a2e-4a20-9a20-0a200a200a20</a>	LeBron James is the greatest Basketball Player

## Search architecture and implementation

Our search architecture will be implemented for searching articles.

To search our articles, we will use three constraints:

### 1. Search text specified in the search bar

If a search text is specified, the articles will be filtered according to the text provided in the search bar. To filter articles according to search text, we will use the like operator in our MySQL database to find articles that have a heading that have the same pattern specified as the text that is provided in the search bar.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM
(SELECT image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article
WHERE (heading LIKE CONCAT('%',inSearchText,'%') OR inSearchText is NULL) ) AS
Articles JOIN RegisteredUser ON user_id = Articles.fk_authorid ORDER BY
posttime DESC;
```

### 2. Date filter provided in the filter panel

If we only filter articles by date, we will return all the articles that have been posted after and including the date that has been specified in the filter.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM
(SELECT image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article
WHERE (posttime >= inPostdate OR inPostdate is NULL)) AS Articles JOIN
RegisteredUser ON user_id = Articles.fk_authorid ORDER BY posttime DESC;
```

### 3. Sport type filter provided in the filter panel

If we only filter articles via the type of sport, we will return all the articles that talk about the sport specified in the filter.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM
(SELECT image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article
WHERE (sport = inSport OR inSport is NULL )) AS Articles JOIN RegisteredUser ON
user_id = Articles.fk_authorid ORDER BY posttime DESC;
```

Combining the conditions for all the three filters will result in a search query like the one below.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM
(SELECT image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article
WHERE (posttime >= inPostdate OR inPostdate is NULL) AND (sport = inSport OR
inSport is NULL ) AND (heading LIKE CONCAT('%',inSearchText,'%') OR
inSearchText is NULL)) AS Articles JOIN RegisteredUser ON user_id =
Articles.fk_authorid ORDER BY posttime DESC;
```

If we don't specify any filters, then all articles in the database will be returned.

All articles returned from the backend service to the user will be organized according to the date they were posted in descending order. This basically means that users will be able to view the latest articles on the website.

The properties returned from the database for each article are the:

**image\_URL**

> This is the thumbnail image of the article.

**heading**

> This is the title of the article.

**subHeading**

> This is the subHeading of the article.

**posttime**

> This is the date and time the article was posted.

**sport**

> This is the sport that the article is providing information for.

**Author**

> This is the name of the author who created the article.

## High-Level API and Main Algorithm

### YourSports API

Our Backend service will be hosted on this URL: <http://34.136.124.189>. The Backend service will provide access to the YourSports API. We will include the following APIs:

#### 1. Upload article

Through this API, the creator shall create an article. To do so, we required a heading, a subheading, an instruction, a content, a conclusion and the sport type. Backend will add some more information including postdate and author name. This data will be sent and stored into our database. Then this article can be searched and viewed by other users.

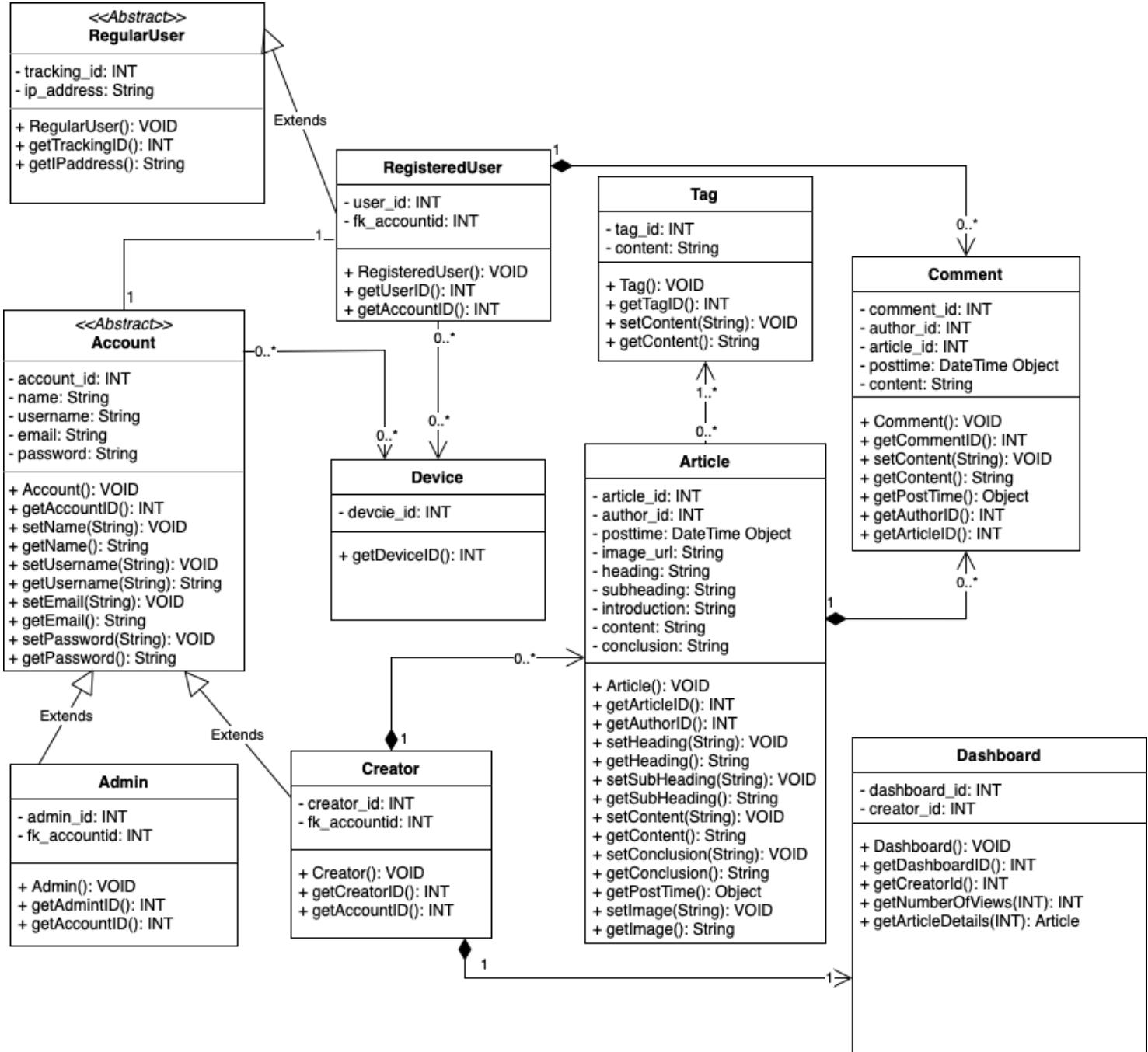
#### 2. Interact with article

Through these APIs, the reader who logged in can interact with articles in multiple ways. These interaction methods include like, dislike and comment. If users click the like or dislike button, the frontend will send a request to the backend. Then the information will be stored into the database. Similarly, users can also leave comments. Once the comment was submitted, it will be stored into the database. Backend also tracks how many times the article has been viewed and automatically updates this information in the database.

#### 3. Post in discussion forum

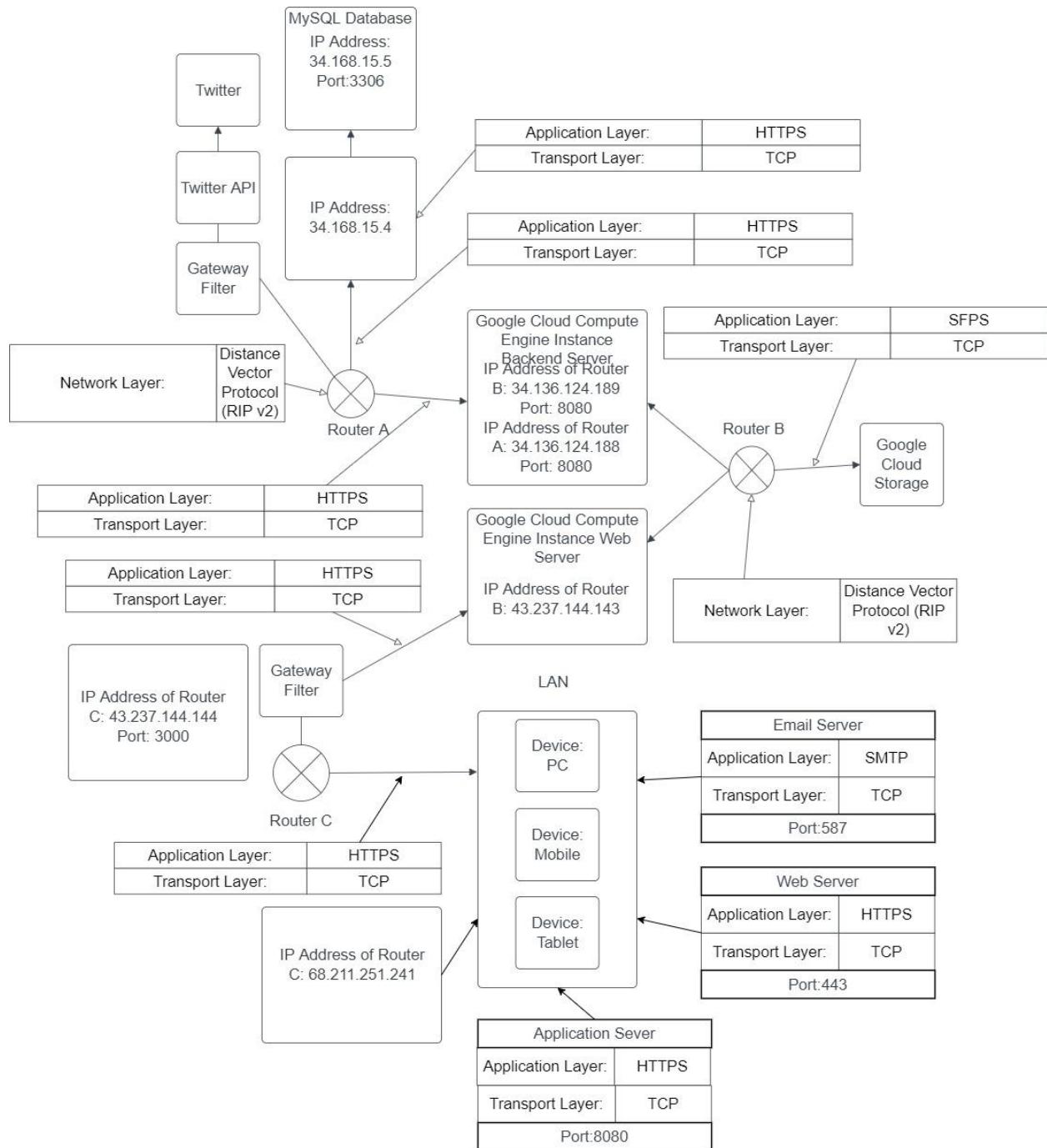
Through this API, the users who logged in can post messages in the discussion forum of any game. Once the message is submitted, it will be stored into the database. Other users shall be able to read this message. Users can also delete their posts by clicking the delete button. The frontend will send a request to the backend, and the backend will update the database.

## UML Diagram

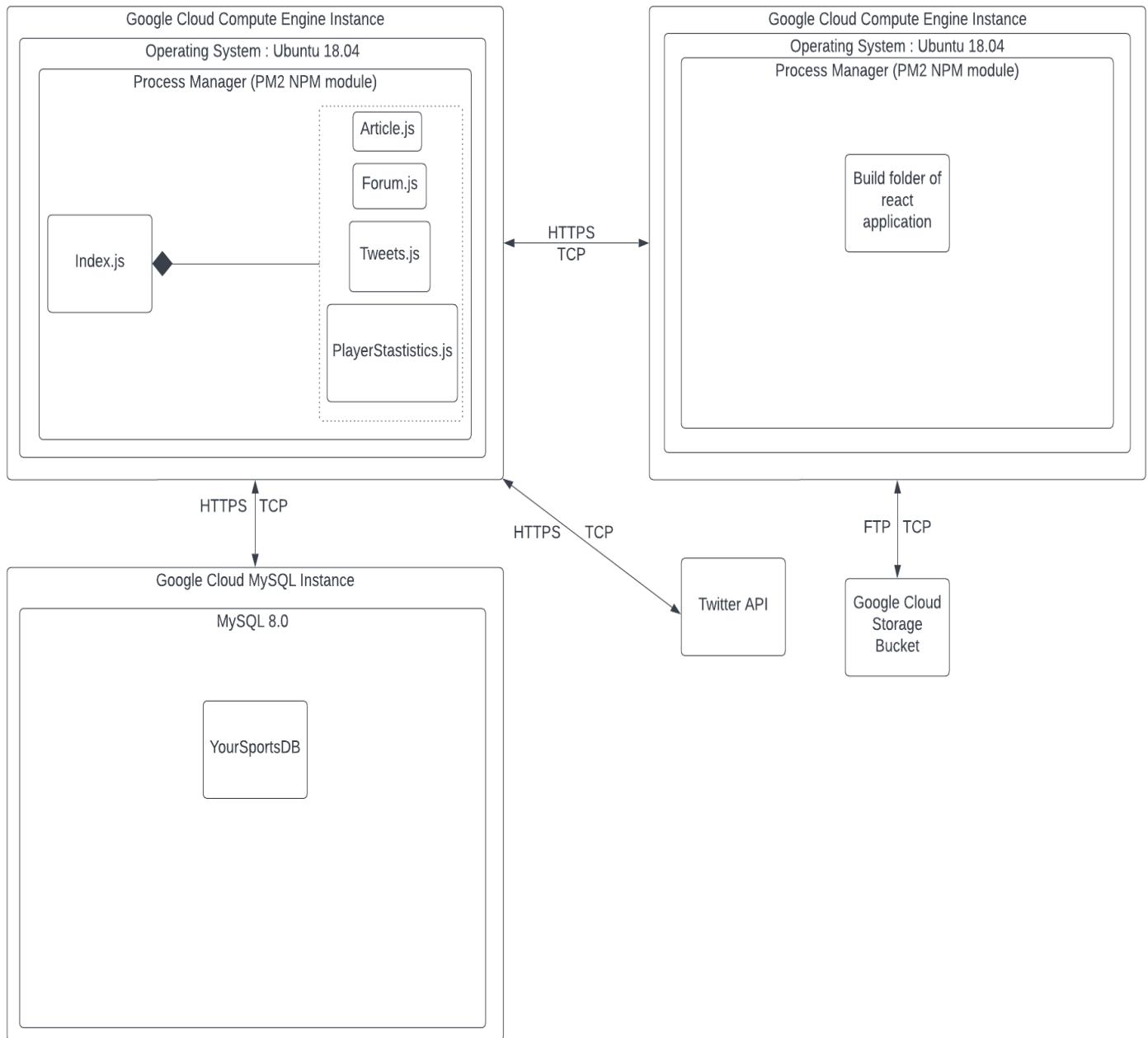


## Application Network and Deployment Diagram

### Application Network Diagram



## Deployment Diagram



## Current Key Risks

### Skills

#### Risks

- Most of the team does not have experience with cloud technologies
- Team members have varying skills, hence it is difficult to make decisions regarding technologies that fit everyone's requirements.

#### Proposed Solution

- Give team members enough time to learn the required technologies for our product.
- Make sure at least 3 members of the team can work with the cloud technologies that will be used in our project.

### Schedule

#### Risks

- The priority one functional requirements may be difficult to complete by the end of the semester.

#### Proposed Solution

- Set internal deadlines to complete priority one requirements related to the features of our application.
- Delegate work according to the skillset of a team member.

### Technical

#### Risks

- Many team members are not familiar with SQL which will be used to implement all of our search algorithms.

#### Proposed Solution

- Make sure every team member understands the functions implemented in the front-end and backend.
- Use the internet to search for optimal solutions that we will face when building our project

- 
- Limit the number of different technologies used in our project.
  - The team lead will make sure to review all the code from each branch before being pushed to the master branch.

## Teamwork

### Risks

- There is a possibility of conflict between team members who are working together on the same tasks.

### Proposed Solution

- The team lead should delegate tasks to team members who have a similar set of skills, and make sure to resolve any misunderstandings between team members.

## Legal/Content

### Risks

- Users who are creators might upload inappropriate content in their articles.
- Users might use media/content that is not owned by them when they are trying to publish their articles.
- Setting up a terms and conditions agreement.

### Proposed Solution

- Use an administrator account to monitor the content uploaded by creators.
- Create a legal document that states behavioral guidelines.
- Create a legal document that users who are creators agree to use only work owned by them when they are uploading articles.
- Use a ready made terms and conditions document that can be changed as per needed.



## Project Management

Our project will be composed of three main components: front-end, back-end, and database system. Each component will be assigned to developers. The team lead will assign tasks to the people with the desired skill-set and set internal deadlines. The team lead will offer 24/7 support to make sure the tasks are completed by their given deadlines. The team lead will regularly pull from the Github repository, to make sure the developers are following proper coding standards, and making sure everything is going as per the requirements of the project. The front-end and back-end leads are responsible for setting their own deadlines, and plans of action to complete their given tasks. Trello will be used to track the tasks, the resources of the project will be accessed through Trello, and all the communication will be done through Discord. The team lead will set realistic deadlines, and consult with the front-end and back-end lead before assigning tasks and setting deadlines, to make sure realistic and achievable goals are set. The entire team will attend the scheduled meetings unless otherwise instructed, and update and ask for support from other team members to make sure tasks are completed by their deadline. The team lead will also assign tasks related to the documentation if they can be done individually, or complete the tasks together in a team meeting. Finally, the team lead will consult the rest of the team members before making any major decisions related to the project.

## List of Team Contributions

### Kshitiz Sareen (Team Lead, Back-end Lead)

Constructed the back-end API for searching news articles, and signing up. Deployed the front-end and backend applications on the server. Managed team meetings, and tasks progress of individual team members. Draws the “checking sport news” mockup storyboard.

### Kevin Islas (Front-end Lead)

Contributed in designing the home page and the display for the articles. Implemented the search API front-end call function as well as the display of the data accordingly. Contributed in displaying information received from the sign up API and implemented the navbars. Draws the “posting news articles” mockup storyboard.

### Shamar Ireland (Front-end Developer)

Contributed in designing the home and signup page as well as the formatting of the data displayed below the search bar. Contributed in implementing the signup API and prioritizing the use cases. Draws the “signing up” mockup storyboard.

### Wenye Guo (Database Administrator)

Contributed in drawing the high level UML class diagram, ERD, database model, and in database organization. Draws the “creating group chat” mockup storyboard.

### Sabur Saigani (Functional Lead)

Contributed in drawing the networking diagram, high level UML class diagram, ERD, database model, and in database organization. Contributed in checking the documentation’s grammar. Draws the “twitter feeds” mockup storyboard.

### Jonathan Ip (Functional Assistant)

Contributed in formatting and organizing the documentation’s layout as well as checking the documentation’s grammar. Draws the “following sport game” mockup storyboard.

### Mathew O Abiola (Github Master)

Contributed in prioritizing the use cases. Draws the “fantasy betting” mockup storyboard.

# YourSports

Dynamic Debugger (Team 02)

## Team & Backend Lead

Kshitiz Sareen: [ksareen@sfsu.edu](mailto:ksareen@sfsu.edu)

## Frontend

Lead Kevin Islas

Developer Shamar Ireland

## Functional

Lead Sabur Saigani

Assistant Jonathan Ip

## Database Administrator

Wenye Guo

## Github Master

Mathew O Abiola

## Milestone 3

Milestone	Date
M3V2	July 22nd, 2022
M3V1	July 18th, 2022
M2V2	July 14th, 2022
M2V1	July 7th, 2022
M1V2	June 30th, 2022
M1V1	June 9th, 2022

## Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Data Definition</b>	<b>2</b>
Users	2
Regular User	2
Registered User	2
Account	2
Basic Account	2
Creator Account	2
Admin Account	3
News	3
Tweets	3
Articles	3
Comment	4
Games	4
Group	4
Discussion Forum	4
Group Chat	4
Posts	5
Dashboard	5
Backend Service	5
Website	5
Database	5
<b>Prioritize Functional Requirements</b>	<b>6</b>
Priority 1	6
Priority 2	7
Priority 3	8
<b>Wireframes</b>	<b>10</b>
Signing Up Wireframes	10
Checking Sports News Wireframes	11
Posting News Articles Wireframes	12
Following Sports Games Wireframes	13
Checking Player Statistics Wireframes	14
<b>Database Architecture and Organization</b>	<b>15</b>
Business Rules	15
Entities	15
Entity Relationship Diagram (ERD)	17
Database Model	18
Database that will be used	18
Media Storage	18
Search architecture and implementation	19
<b>High Level Diagram</b>	<b>20</b>
UML Diagram	20
Application Network Diagram	21
Deployment Diagram	22
<b>List of Team Contributions</b>	<b>23</b>
<b>Horizontal prototype feedback</b>	<b>24</b>
Home page	24
Game page	24
Signup	24
Login	24
Player page	24
Upload articles	24
Setting	24

## Data Definition

### Users

Users will be referring to the group of people who use our application. Users will be given wide access to features like searching news, access to live Twitter feeds, following live games, posting opinions in the discussion forum of a game, and viewing player statistics.

There will be two types of users in our application:

#### Regular User

These are the users who have not logged in. Regular users will only be able to watch scores of live games and search for news and tweets.

Regular Users will have the following attributes:

##### Tracking ID

> This refers to a unique id given to the unregistered user.

##### IP address

> This refers to the IP address of the device that the user uses to access the website.

#### Registered User

These are users who have an account. Registered users will have access to all features our application provides. Registered users inherit from regular users.

Registered users will have the following attributes:

##### User

> This refers to the tracking ID of the user.

##### Account

> This refers to the account of the user.

### Account

Each registered user will have an account that contains all the details and keys that are required to identify a user.

There will be three types of accounts in our application:

#### Basic Account

A Basic account will have access to features like searching news, access to live Twitter feeds, following live games, posting opinions in the discussion forum of a game, and viewing player statistics.

The Basic Account has the following attributes:

##### Name

> This refers to the name that the user used to sign up, and it is usually the legal name of the user.

##### Username

> This refers to the username that the user used to sign up for, and it will be used to distinguish a user from other users.

##### Email

> This refers to the email that the user used to sign up. The email will be used to associate their account with their email, and also serve as a communication method between the user and the application.

##### Password

> This refers to the set of characters that a user used to sign up, and it will be used to authenticate a user along with their email.

#### Creator Account

A Creator Account will have access to all the privileges and properties that a basic account has with the additional privilege of publishing articles to the website. Every user that has a creator account will have a basic account. A creator account will inherit from a basic account.

The creator account will have the following properties:

#### **Account**

> This refers to the account of the user.

#### **Admin Account**

An Admin account is used to regulate and ensure proper usage of the application by other accounts. An Admin account has all privileges in the application along with access to the database and backend service to manipulate, change, add and remove data. There will only be one admin account. The admin account will also inherit from the basic account.

The admin account has the following properties:

#### **Account**

> This refers to the account of the user.

### **News**

News will refer to the information that users can access from our application.

Our application will provide different types of news:

#### **Tweets**

These are the tweets collected from Twitter according to users' preferences and interests.

Tweets will have the following properties:

#### **Username**

> This is the username of the user who posted the tweet. The username will follow twitter data conventions.

#### **Content**

> This is the content of the tweet.

#### **Articles**

These are the articles posted by users who have a creator account. Articles can be searched and filtered by date posted and which sport is it used to talk about.

Articles have the following properties:

#### **Post Time**

> This is the date and time the article was posted.

#### **Image**

> This is the image that will be used as the thumbnail for the article. The acceptable image formats are png and jpeg, with a maximum size of 5 MB.

#### **Heading**

> This is the title of the article.

#### **Subheading**

> This is the subheading of the article.

#### **Introduction**

> This is the introduction part of the article.

#### **Content**

> This is the main content that contains all the detailed information of the article.

#### **Conclusion**

> This is the conclusion of the article.

#### **Author**

> This is the Author of the article.

## Sport

> This is the type of sport which the article targets. E.g. Basketball and Baseball.

## Comment

Registered users will be able to post comments on articles to express their views and opinions on the information given in the article.

Comment will have the following properties:

### Article

> This is the Article for which the comment was posted.

### Author

> This is the user who has posted the comment.

### Post Time

> This is the date and time where the comment was posted.

### Content

> This contains all the information that a user wants to convey.

## Games

This refers to the past, current, and future sports games that users can follow. Users can follow the scores, and post any opinions they have about the game.

Games will have the following properties:

### Player One Score

> This is the score of the home team.

### Player Two Score

> This is the score of the opponent team.

### Game Location

> This is the location of the game.

### Team One

> This is the home team.

### Team Two

> This is the away team.

## Group

This refers to all the groups like group chats and discussion forums that allow users to interact with other users.

There will be different types of groups:

### Discussion Forum

The discussion forum will refer to the forums of every sports game where users can post their opinions about the game. This discussion forum will have no owner and every registered user who follows a game will be able to join the discussion forum.

Discussion forums will have the following properties:

### Forum Name

> This is the name of the forum.

### Game

> This is the sports game for which the forum has been created.

### Account List

> This is the list of all the accounts of the users that are part of the forum.

### Group Chat

The group chat will refer to private groups of sports fans who share similar interests. The private group chats will have an owner and every registered user will only be able to join a private group chat upon the owner's approval.

Group chats will have the following properties:

#### **Group Chat Name**

> This is the name of the group chat.

#### **Group Chat Owner**

> This is the user who owns the group.

#### **Group Account**

> This is the list of all the accounts of the users that are part of the group chat.

## **Posts**

This refers to the message a user posts in a group. All communication between users of the application will happen between the users.

Posts have the following properties:

#### **Account**

> This is the account of the user who has posted the message.

#### **Content**

> This is the content of the message.

#### **Group**

> This is the group in which the user has posted a message.

## **Dashboard**

This refers to the Dashboard of every user that has a creator account.

Dashboard has the following properties:

#### **Article List**

> This is the list of all the articles that the creator has uploaded.

#### **Content**

> This is the number of views for each article.

#### **Creator**

> This is the creator account of the user who owns the dashboard.

## **Backend Service**

The backend service will be used frequently to refer to all the operations that will happen on the server-side like filtering and searching for articles, filtering and retrieval of tweets, fetching live updates from games, and handling messages from users, updating and retrieving player statistics.

## **Website**

The Website will be used frequently to refer to all the operations that will happen on the client-side like signing up and logging in, and verification and validation of data.

## **Database**

The database will be used frequently to refer to all the operations that involve the storage of data. It will be used to store the news, account information of users, information on sports games, information on players, and all the messages that a user posts in a group.

## Prioritize Functional Requirements

### Priority 1

#### Regular user

1. Regular users shall be able to create an account.
2. Regular users shall be able to sign up as a creator account.
3. Regular users shall be able to access the homepage.
4. Regular users shall be able to view the live scores of a game.
5. Regular users shall be able to view the duration of a game.
6. Regular users shall be able to view updated statistics of players.
7. Regular users shall be able to search for articles through a search bar.
8. Regular users shall be able to filter articles they would like to read by keywords.
9. Regular users shall be able to filter articles they would like to read by different types of sports.
10. Regular users shall be able to search articles by heading of the article.
11. Regular users shall be able to view the article heading in the search results for articles.
12. Regular users shall be able to view the article author in the search results for articles.
13. Regular users shall be able to view the article postdate in the search results for articles.
14. Regular users shall be able to view the article image as a thumbnail in the search results for articles.
15. Regular users shall be able to search games by the teams playing in the search bar.
16. Regular users shall be able to filter games by the sport type.
17. Regular users shall be able to filter games by their date.
18. Regular users shall be able to view the game score in the search results for games.
19. Regular users shall be able to view the game date in the search results for games.
20. Regular users shall be able to view the game location in the search results for games.
21. Regular users shall be able to search for player statistics according to the player name in the search bar.
22. Regular users shall be able to filter player statistics by the sport type.

#### Registered user

23. Registered users shall have the same privileges as regular users.
24. Registered users shall be able to log in.
25. Registered users shall be able to log out.
26. Registered users shall be able to reset their password.
27. Registered users shall be able to delete their account.
28. Registered users shall be able to post in the discussion forum of the game.
29. Registered users shall be able to delete their posts in the discussion forum.
30. Registered users shall be able to comment on articles.

#### Basic Account

31. Accounts shall be able to be deleted by users.
32. Accounts shall contain the name of the users.
33. Accounts shall contain the username of the users.
34. Accounts shall contain the email of the users.
35. Accounts shall contain the password of the users.

#### Creator Account

36. Creators shall have the same privileges as registered users.
37. Creators shall be able to upload articles to the website.
38. Creators shall be able to view Dashboard.

#### Admin Account

39. Admin shall have all the privileges.
40. Admin shall be able to create discussion forums for upcoming games.

#### Articles

41. Articles shall be posted by creator users.
42. Articles shall be able to be filtered by date.
43. Articles shall be able to be filtered by sports type.
44. Articles shall be able to be filtered by keywords specified by the user.
45. Articles shall contain a heading.

- 46. Articles shall contain a subheading.
- 47. Articles shall contain an introduction.
- 48. Articles shall contain contents.
- 49. Articles shall contain a conclusion.
- 50. Articles shall contain an image.

#### Games

- 51. Games shall be filtered by date.
- 52. Games shall be filtered by sport type.
- 53. Games shall be filtered by keywords specified by users in the search bar.
- 54. Games shall have an individual discussion forum.
- 55. Games shall be able to show live scores of corresponding teams.
- 56. Games shall be able to show the date of the game.
- 57. Games shall be able to show the location of the game.

#### Discussion Forum

- 58. Discussion forum shall be able to be viewed by all users.
- 59. Discussion forum shall be able to show posts from users.

#### Dashboard

- 60. Dashboard shall be able to show the number of views of the article.
- 61. Dashboard shall be able to show the heading of the article.
- 62. Dashboard shall be able to show the posted date of the article.

#### Backend Service

- 63. The backend service shall be able to validate the email for duplicates when users try to sign up.
- 64. The backend service shall be able to check for duplicate accounts when users try to sign up.
- 65. The backend service shall be able to create an account in the database.
- 66. The backend service shall be able to store posts by a user in the discussion forums.
- 67. The backend service shall be able to update player statistics in the database.
- 68. The backend service shall be able to filter articles specified by the user.
- 69. The backend service shall be able to filter games specified by the user.
- 70. The backend service shall be able to filter player statistics specified by the user.
- 71. The backend service shall be able to fetch updates from a game every 5 seconds.

#### Website

- 72. The website shall be able to check if the first name is not empty users try to sign up.
- 73. The website shall be able to check if the last name is not empty when a user try to sign up.
- 74. The website shall be able to check if the email is not empty when users try to sign up.
- 75. The website shall be able to check if the password is at least 8 characters before the users try to sign up.
- 76. The website shall be able to check if the date of birth is at least 13 years old when users try to sign up.
- 77. The website shall be able to check if the terms of service has been accepted when users try to sign up.

#### Comment

- 78. Comments shall be able to be viewed by regular users.
- 79. Comments shall be posted by registered users.
- 80. Comments shall be able to be deleted by poster.

## Priority 2

#### Regular user

1. Regular users shall be able to view the player statistics through a pie chart.
2. Regular users shall be able to be redirected to a live broadcast of the game.
3. Regular users shall be able to search for specific comments in the discussion forum.

#### Registered user

4. Registered users shall be able to like an article.
5. Registered users shall be able to dislike an article.
6. Registered users shall be able to change their profile picture.
7. Registered users shall be able to view profiles of other users.
8. Registered users shall be able to red flag other registered users for any posts in the discussion forum.
9. Registered users shall be able to follow sports games.
10. Registered users shall be able to unfollow games they are currently following.

- 11. Registered users shall be able to search for specific messages in the discussion forum.
- 12. Registered users shall be able to mark posts as read so that they don't receive notifications.
- 13. Registered users shall be able to mark as read on articles.
- 14. Registered users shall be able to receive responses to their posts in the discussion forum.

#### Creator Account

- 15. Creator shall be able to view the article statistics through a pie chart.
- 16. Creators shall be able to see the number of likes an article has received.
- 17. Creators shall be able to see the number of dislikes an article has received.
- 18. Creators shall be able to select their target audience.

#### Dashboard

- 19. Dashboard shall be able to show the comments on the article.
- 20. Dashboard shall be able to show the redflags of an article.

#### Backend Service

- 21. The backend service shall be able to send a verification email for users to verify their account.

#### Discussion forum

- 22. Discussion forum shall be sorted with specific words.

### Priority 3

#### Regular user

- 1. Regular users shall be able to read live twitter feeds.
- 2. Regular users shall be able to switch between light and dark mode.

#### Registered user

- 3. Registered users shall be able to bet on live and upcoming games.
- 4. Registered users shall receive 100 YourSports coins when signing up.
- 5. Registered users shall get cash by selling their YourSports coins.
- 6. Registered users shall win YourSports coins if they win a bet.
- 7. Registered users shall lose YourSports coins if they lose a bet.
- 8. Registered users shall be able to buy YourSports coin using actual money.
- 9. Registered users shall be able to view their current amount of YourSports coins in their account.
- 10. Registered users shall be able to enable/disable receiving emails about sports news.
- 11. Registered users shall be able to choose how frequently they receive emails about sports news.
- 12. Registered users shall be able to read live Twitter feeds.
- 13. Registered users shall be able to type the topics they would like to follow on Twitter.
- 14. Registered users shall be able to unfollow topics they are currently following.
- 15. Registered users shall be able to receive emails of any new tweets.
- 16. Registered users shall be able to enable receiving notifications by email.
- 17. Registered users shall be able to choose if they want to enable receiving sms that contain sports news.
- 18. Registered users shall be able to enable receiving notifications by messages on the phone.
- 19. Registered users shall be able to invite their friends to use the YourSports website.
- 20. Registered users shall be able to send messages in the group chat.
- 21. Registered users shall be able to receive messages in the group chat.
- 22. Registered users shall be able to view messages in the group chat.
- 23. Registered users shall be able to see other online users in the group chat.
- 24. Registered users shall be able to delete messages they previously sent in the group chat.
- 25. Registered users shall be able to edit messages they previously sent in the group chat.
- 26. Registered users shall be able to create a group chat.

#### Group Owner

- 27. The Group's Owner shall be able to change the group chat name.
- 28. The Group's Owner shall be able to invite other users to the group chat.
- 29. The Group's Owner shall be able to block any users in the group chat.

#### Message

- 30. Messages shall be able to be edited by poster.

#### Tweets

- 31. Tweets shall be pushed to users.

32. Different topics of tweets shall be pushed to users.

#### Notification

33. Notification shall be sent by email.

34. Notification shall include new tweets or/and sport news.

35. Notifications shall be sent by messages.

#### Backend service

36. The backend service shall be able to filter tweets by the topics a user follows.

#### Group chat

23. Group chat shall contain multiple group members.

24. Group chat shall have one group owner.

25. Joining a specific group chat shall require permission of the group owner.

26. Group members of a group chat shall be able to post messages.

#### Message

27. Messages in the group chat shall be posted by registered users.

28. Messages in the group chat shall be able to be deleted by poster.

29. Messages in the group chat shall be able to be marked using a red flag.

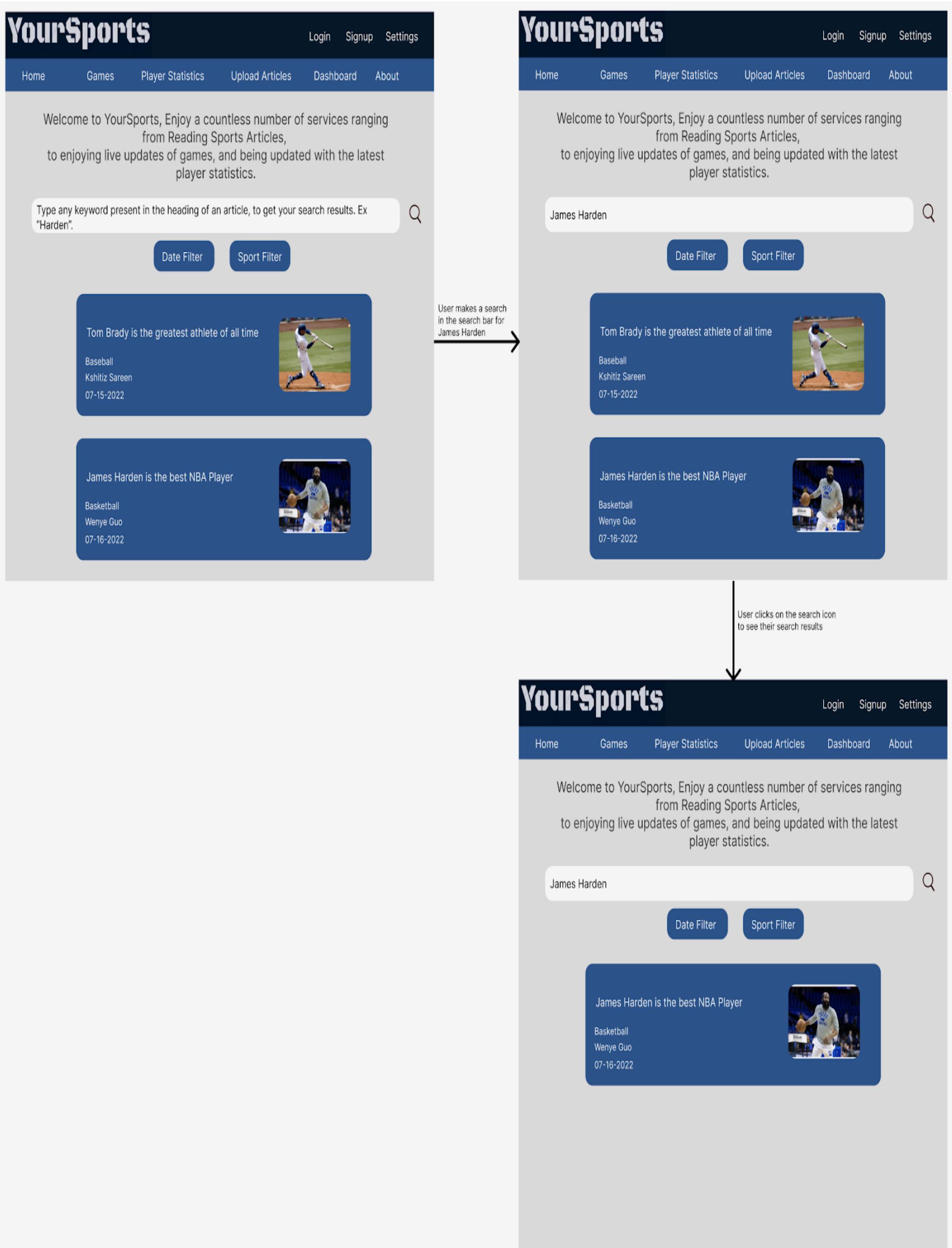
30. Messages in the group chat shall be responded to by other registered users.

## Wireframes

### Signing Up Wireframes



## Checking Sports News Wireframes



## Posting News Articles Wireframes

This wireframe shows a news article titled "Tom Brady is the greatest athlete of all time". The article has a view count of 5653 and was uploaded by Kshitiz Sareen on 07-15-2022. It includes a thumbnail image of Tom Brady swinging a bat.

User clicks on Dashboard, to view the articles they have uploaded.

This wireframe shows the dashboard where users can upload new articles. It displays two recent uploads: one about Tom Brady and another about James Harden.

Once the user clicks on the Ok button in the Article Uploaded popup, the user is redirected to the previous page, which in this case is the home page

This wireframe shows a confirmation message "Article Uploaded" with an "Ok" button. Below it is the main news article content about Tom Brady.

User clicks on Upload Button, and receives a confirmation on whether article has been uploaded

This wireframe shows the "Upload Articles" page. It includes fields for "Heading", "Subheading", "Image", "Introduction", "Content", and "Conclusion". An "Upload" button is at the bottom.

User fills in on the details of the article that they want to upload

This wireframe shows the published version of the news article. It includes the heading, subheading, image, introduction, content, conclusion, and an "Upload" button.

## Following Sports Games Wireframes

### YourSports

Home Games Player Statistics Upload Articles Dashboard About

Welcome to YourSports, Enjoy a countless number of services ranging from Reading Sports Articles, to enjoying live updates of games, and being updated with the latest player statistics.

Type any keyword present in the heading of an article, to get your search results. Ex "Harden".

Date Filter Sport Filter

**Tom Brady is the greatest athlete of all time**

Baseball  
Kshitiz Sareen  
07-15-2022



**James Harden is the best NBA Player**

Basketball  
Wenya Guo  
07-16-2022



### YourSports

Home Games Player Statistics Upload Articles Dashboard About

Search Games By the teams playing

Date Filter Sport Filter

**Warriors 103 - 90 Celtics**

06-13-2022 Chase Center, San Francisco

User clicks on the Games Page, to see all the latest games going on

### YourSports

Home Games Player Statistics Upload Articles Dashboard About

**Warriors 103 - 90 Celtics**

Duration Chase Center, San Francisco

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Go Warriors !!!**

The User presses enter and their message is sent in the discussion forum of the game.

### YourSports

Home Games Player Statistics Upload Articles Dashboard About

**Warriors 103 - 90 Celtics**

Duration Chase Center, San Francisco

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Input Text**

### YourSports

Home Games Player Statistics Upload Articles Dashboard About

**Warriors 103 - 90 Celtics**

Duration Chase Center, San Francisco

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Test Username**  
This is a Test Post, Warriors is playing really well, and its going to win the finals.

**Temp Username**  
Go Warriors!!

**Input Text**

## Checking Player Statistics Wireframes

Welcome to YourSports, Enjoy a countless number of services ranging from Reading Sports Articles, to enjoying live updates of games, and being updated with the latest player statistics.

Type any keyword present in the heading of an article, to get your search results. Ex "Harden".

Date Filter Sport Filter

**Tom Brady is the greatest athlete of all time**

Baseball  
Kshitiz Sareen  
07-15-2022

**James Harden is the best NBA Player**

Basketball  
Wenye Guo  
07-16-2022

Search Player By Their Name

Sport Filter

Steph Curry

Mookie Betts

User clicks on the Player Statistics Page, to see all the updated statistics of Players

Search Player By Their Name

Sport Filter

Steph Curry

Mookie Betts

Games: 14  
At Bats: 1.2  
Runs: 0.0  
Hits: 0.0  
Singles: 0.0  
Doubles: 0.0  
Triples: 0.0  
Home Runs: 0.0  
Runs Batted In: 0.0  
Batting Average: 0.0  
Outs: 1.2

Search Player By Their Name

Sport Filter

Steph Curry

Mookie Betts

Games: 56  
Field Goals Made: 225.9  
Field Goals Attempted: 431.0  
Two Pointers Made: 169.1  
Two Pointers Attempted: 272.9  
Three Pointers Made: 56.8  
Three Pointers Attempted: 158.1

## Database Architecture and Organization

### Business Rules

1. A regular user shall be or not be a registered user.
2. A registered user shall be able to use many devices to login the account.
3. A registered user shall be able to post many posts in game discussion forums.
4. A discussion forum shall contain many posts.
5. A post shall be able to be replied to many times.
6. An account shall be either a creator or an admin.
7. A game shall have at most one discussion forum under the games page.
8. An admin shall be able to manage many discussion forums.
9. An admin shall be able to manage many games.

### Entities

Regular User (Strong)

`Tracking_id` (INT, PK, Not null, AI)

`IP_Address` (INT, Not null)

Account (weak)

`Account_ID` (INT, PK, Not null, AI)

`Name` (VARCHAR, Not null)

`Username` (VARCHAR, Unique, Not null)

`Email` (VARCHAR, Unique, Not null)

`Password` (VARCHAR, Not null)

Admin (weak)

`Admin_ID` (INT, PK, Not null, AI)

`Account_id` (INT, Not null, Unique, FK to Account(`Account_ID`))

Registered user (Weak)

`Tracking_id` (INT, Not null, FK to Regular User(`Tracking_id`))

`Account_id` (INT, Not null, Unique, FK to Account(`Account_ID`))

Game (Strong)

`Game_ID` (INT, PK, Not null, AI)

`Game_Home_Team` (VARCHAR, Not null)

`Game_Away_Team` (VARCHAR, Not null)

`Game_Location` (VARCHAR, Not null)

`Home_Team_Score` (INT, Not null)

`Away_Team_Score` (INT, Not null)

`Duration` (INT, Not null)

Discussion forum (Weak)

`Forum_id` (INT, PK, Unique, Not null, AI)

`Game_id` (INT, Unique, Not null, FK to Game(`Game_ID`))

`Forum_Name` (VARCHAR, Not null)

Posts (Weak)

`Post_id` (INT, PK, Unique, Not null, AI)

`account_ID` (INT, Not null, FK to Account(`Account_ID`))

`Content` (VARCHAR, Not null)

`forum_id` (INT, Not null, FK to Discussion Forum(`Forum_id`))

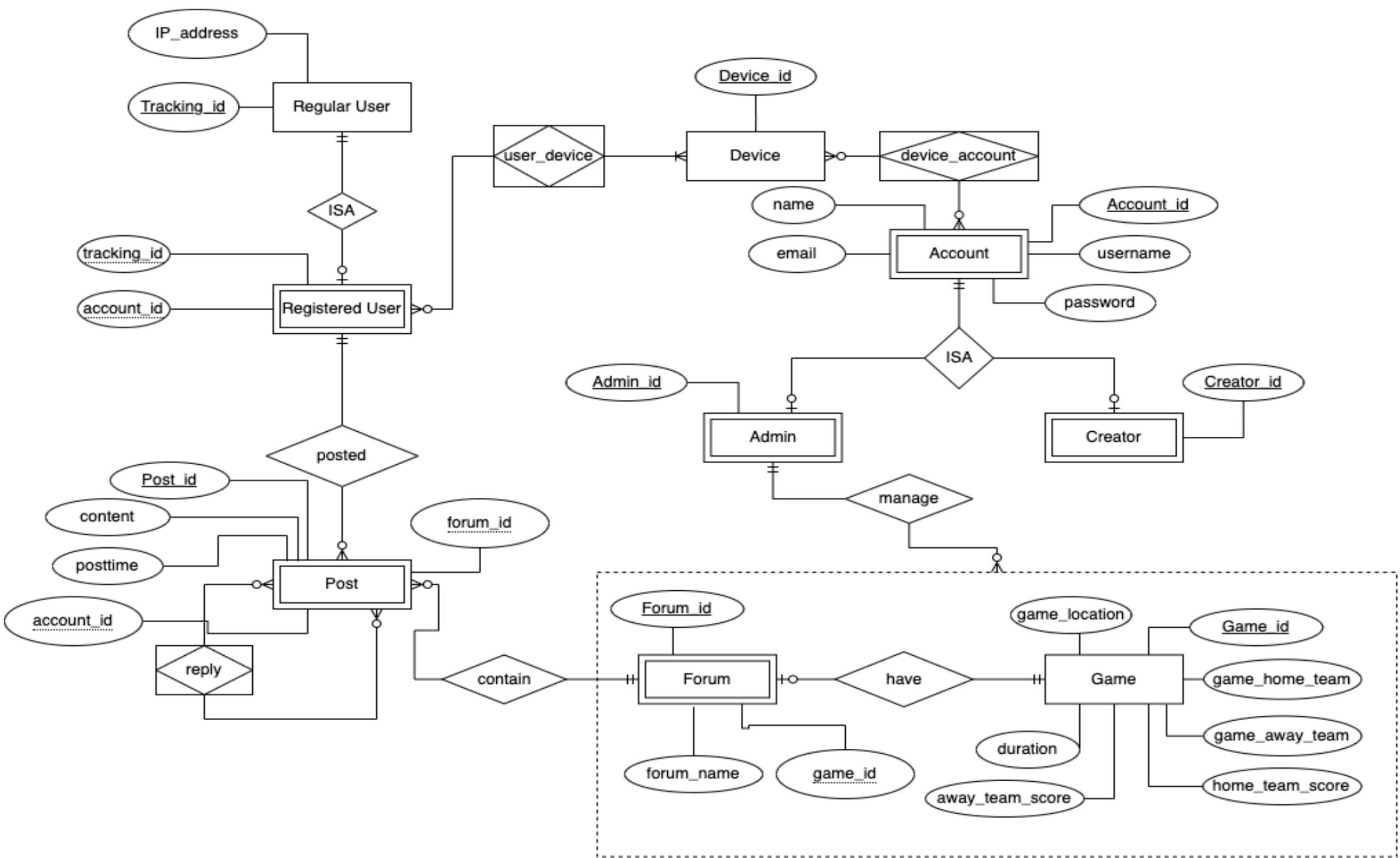


Post Time (DATE, Not null)

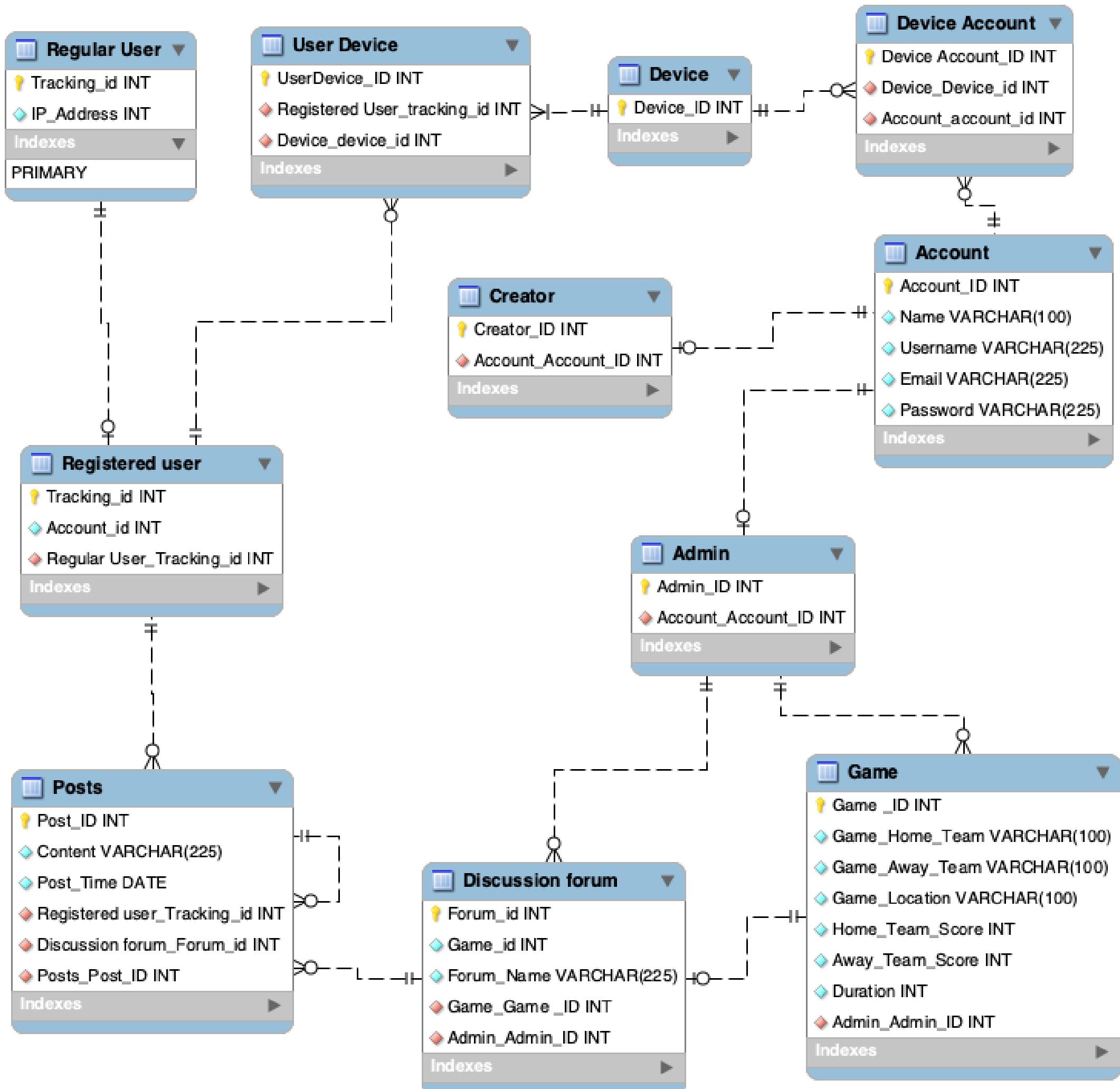
Device (Strong)

Device\_ID (INT, PK, Not null)

### Entity Relationship Diagram (ERD)



## Database Model



## Database that will be used

We will use MySQL database as it supports relational databases, and our database administrator is proficient with MySQL.

## Media Storage

Our application will only deal with images, and text formats of data. The images will be stored in Google Cloud Storage. Each image will only be in a JPEG or PNG file format, with a maximum size of 5MB. The only entity that will require images are the Articles. Each record in the Article table will point to the URL of the image that is stored in the Google Cloud Storage bucket of our project. The Article table will consist of a field that has a datatype of VARCHAR (255). The field will store the URL of the thumbnail of the article. When the front-end requests data from the database, the URL of the image will also be returned, as it is stored in the table. The query to return the articles, that contain the URL of the image is:

```
SELECT * FROM Article, Account WHERE Article_ID = inArticleID AND fk_Author_ID = Account_ID;
```

Since we are returning everything (\*), the URL of the image will also be returned. Here is a sample record of an article thumbnail URL stored in the database.

Article_ID	PostDate	Image_url	Heading
42	2022-08-01	https://firebasestorage.googleapis.com/v0/b/yoursports-352701.appspot.com/.../LeBron%20James%20is%20the%20greatest%20Basketball%20Player	LeBron James is the greatest Basketball Player

## Search architecture and implementation

Our search architecture will be implemented for searching articles.

To search our articles, we will use three constraints:

### 1. Search text specified in the search bar

If a search text is specified, the articles will be filtered according to the text provided in the search bar. To filter articles according to search text, we will use the like operator in our MySQL database to find articles that have a heading that have the same pattern specified as the text that is provided in the search bar.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM (SELECT
image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article WHERE (heading LIKE CONCAT('%',inSearchText,'%')
OR inSearchText is NULL) ) AS Articles JOIN RegisteredUser ON user_id = Articles.fk_authorid ORDER BY posttime DESC;
```

### 2. Date filter provided in the filter panel

If we only filter articles by date, we will return all the articles that have been posted after and including the date that has been specified in the filter.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM (SELECT
image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article WHERE (posttime >= inPostdate OR inPostdate is
NULL)) AS Articles JOIN RegisteredUser ON user_id = Articles.fk_authorid ORDER BY posttime DESC;
```

### 3. Sport type filter provided in the filter panel

If we only filter articles via the type of sport, we will return all the articles that talk about the sport specified in the filter.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM (SELECT
image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article WHERE (sport = inSport OR inSport is NULL )) AS
Articles JOIN RegisteredUser ON user_id = Articles.fk_authorid ORDER BY posttime DESC;
```

Combining the conditions for all the three filters will result in a search query like the one below.

```
SELECT image_URL,heading,subHeading,posttime,name as `Author`,sport FROM (SELECT
image_URL,heading,subHeading,fk_authorid,posttime,sport FROM Article WHERE (posttime >= inPostdate OR inPostdate is
NULL) AND (sport = inSport OR inSport is NULL) AND (heading LIKE CONCAT('%',inSearchText,'%') OR inSearchText is
NULL)) AS Articles JOIN RegisteredUser ON user_id = Articles.fk_authorid ORDER BY posttime DESC;
```

If we don't specify any filters, then all articles in the database will be returned.

All articles returned from the backend service to the user will be organized according to the date they were posted in descending order. This basically means that users will be able to view the latest articles on the website.

The properties returned from the database for each article are the:

**image\_URL**

> This is the thumbnail image of the article.

**heading**

> This is the title of the article.

**subHeading**

> This is the subHeading of the article.

**posttime**

> This is the date and time the article was posted.

**sport**

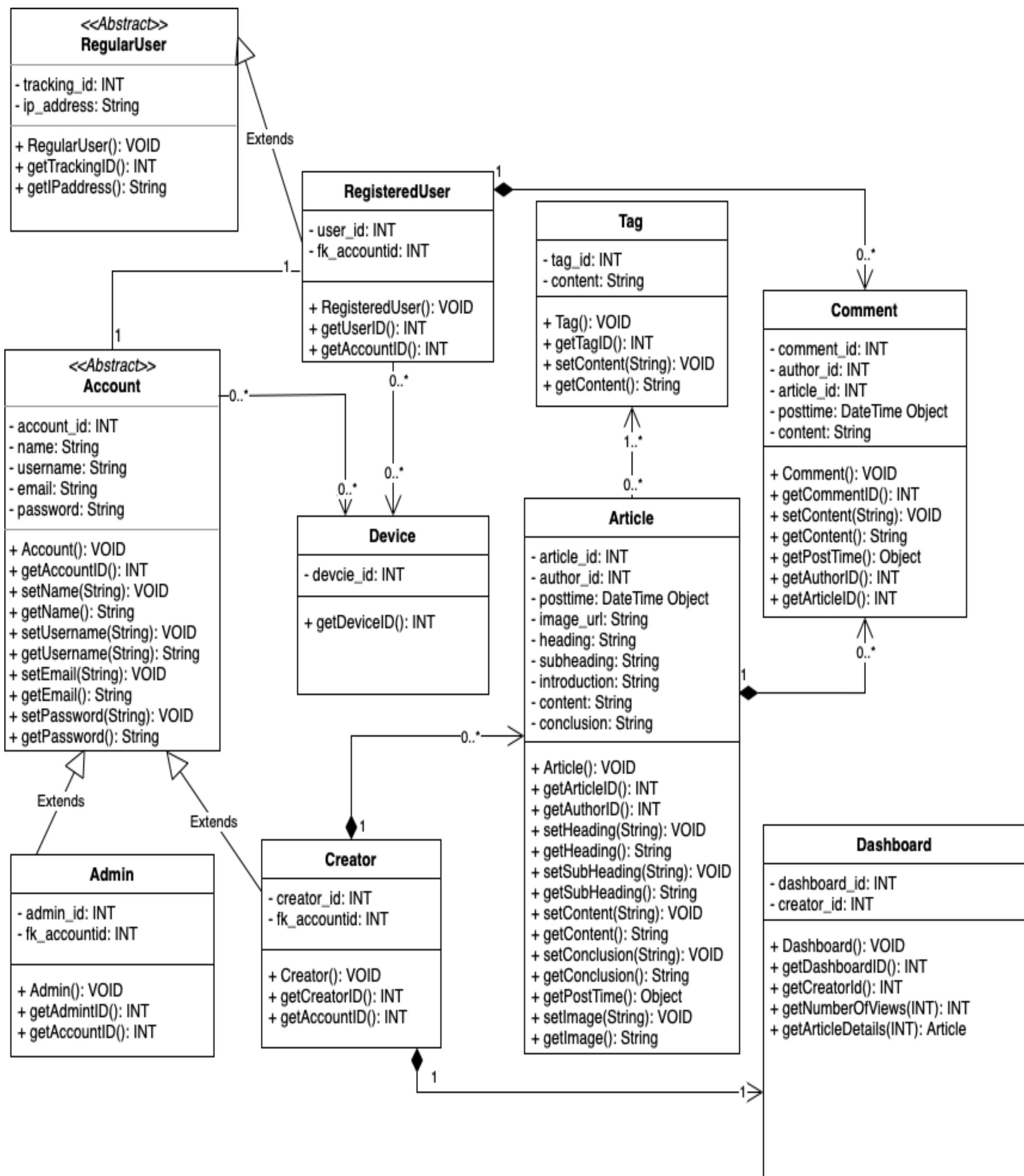
> This is the sport that the article is providing information for.

**Author**

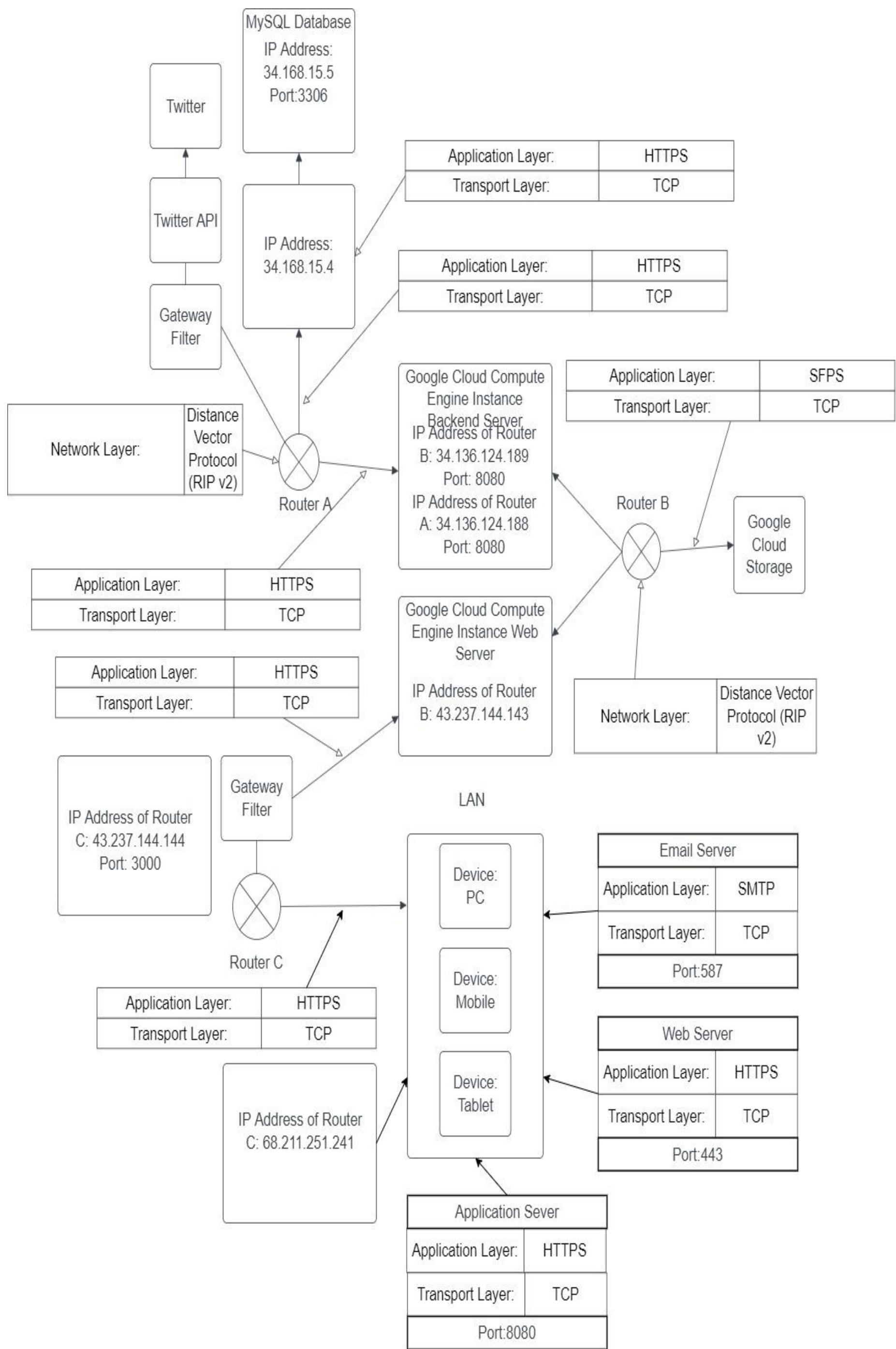
> This is the name of the author who created the article.

## High Level Diagram

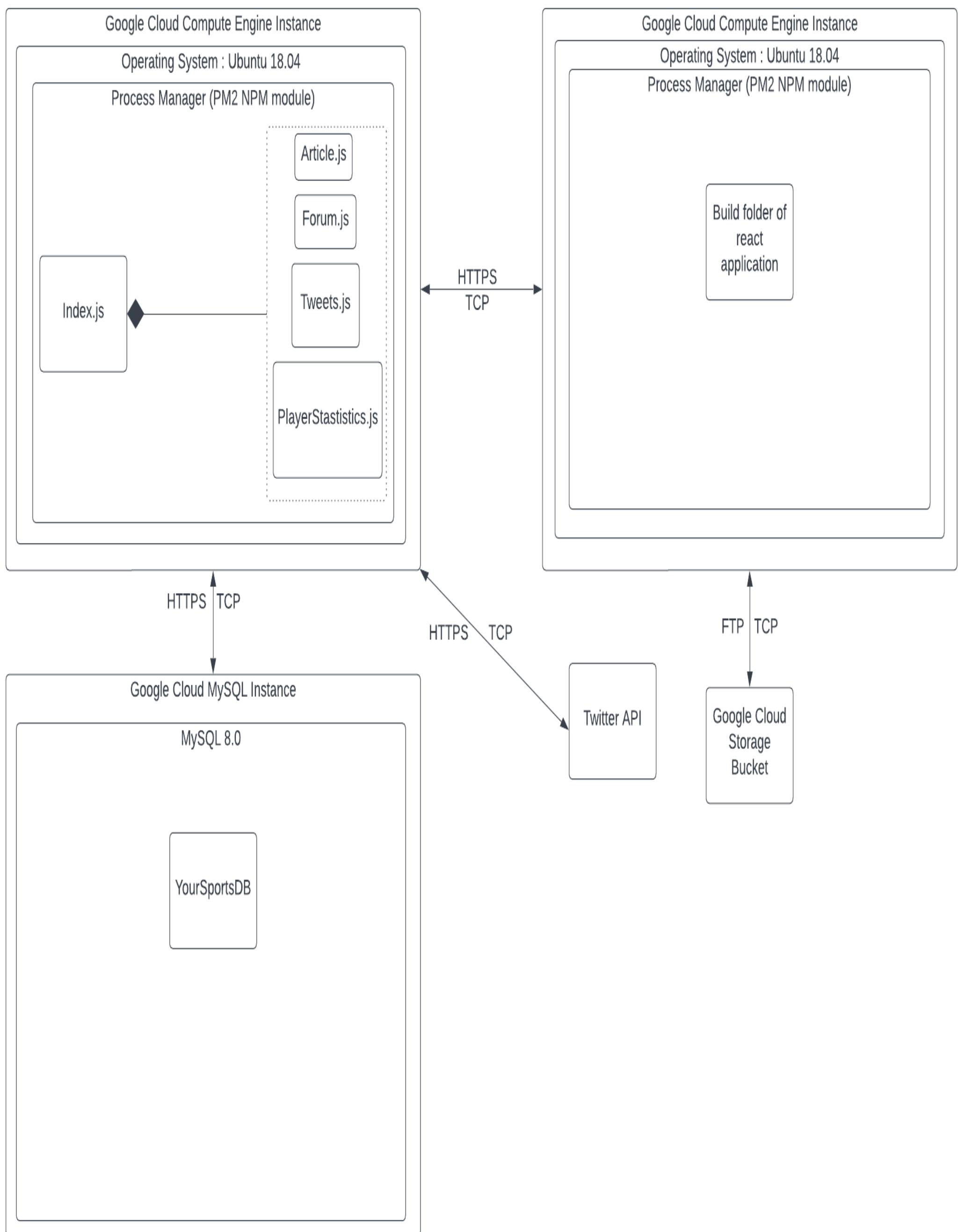
UML Diagram



## Application Network Diagram



## Deployment Diagram





## List of Team Contributions

### Kshitiz Sareen (Team Lead, Back-end Lead)

Helped in creating the Posting News Articles wireframes, Following Sports games wireframes, and Checking Player Statistics wireframes. Organized team meetings and set team agendas

### Kevin Islas (Front-end Lead)

Helped in implementing the front end for the games feature and player statistics. Provided insight on the course of the individual webpages.

### Shamar Ireland (Front-end Developer)

Helped in implementing new front end features such as Uploading Articles and Dashboard. Also helped to update the login, signup and account settings page.

### Wenye Guo (Database Administrator)

Helped in adding new requirements, and in high level database architecture and organization.

### Sabur Saigani (Functional Lead)

Helped in adding new requirements, and in high level database architecture and organization

### Jonathan Ip (Functional Assistant)

Helped in rearranging functional requirements. Also helped in formatting and organizing the documentation's layout as well as checking the documentation's grammar.

### Mathew O Abiola (Github Master)

Helped in creating the Sign Up wireframes, and Checking Sports News wireframes.

## Horizontal prototype feedback

### Home page

1. Font size: Welcome part bigger, nav bar smaller
  2. Filter part bigger
  3. Filter Sports type show selected result, same as date
  4. search meaningless word, need feedbacks
- Your search - ... - did not match any documents.

Suggestions:

- Make sure all words are spelled correctly.
  - Try different keywords.
  - Try more general keywords.
- )

### Game page

1. Navbar suggestion need change
2. Detailed info position is incorrect
3. Add some explanation about this page to guide user, same as "welcome" in home page

### Signup

1. password/username... add requirements
2. Terms of service should be mandatory(services should be a link to another page)
3. add a link to login

### Login

1. Add "forget password", a link to reset password page
2. Add "create a new account", a link to signup page

### Player page

1. Need to change the layout

### Upload articles

1. All put in the Left, waste space

### Setting

1. Reset password page need to implement
2. Add footer(links of social media or some contact info)

# YourSports

Dynamic Debugger (Team 02)

---

## Team & Backend Lead

Kshitiz Sareen: [ksareen@sfsu.edu](mailto:ksareen@sfsu.edu)

## Frontend

Lead Kevin Islas

Developer Shamar Ireland

## Functional

Lead Sabur Saigani

Assistant Jonathan Ip

## Database Administrator

Wenye Guo

## Github Master

Mathew O Abiola

## Milestone 4

Milestone	Date
M4V2	July 31st, 2022
M4V1	July 27th, 2022
M3V2	July 22nd, 2022
M3V1	July 18th, 2022
M2V2	July 14th, 2022
M2V1	July 7th, 2022
M1V2	June 30th, 2022
M1V1	June 9th, 2022



## Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Product Summary</b>	<b>3</b>
Name	3
Major Committed Functions	3
Priority 1	3
Unique Features	6
URL	6
<b>Usability Test Plan</b>	<b>7</b>
Test Plan	7
Purpose	7
Problem Statement and Objectives	7
Task List	7
User Profile	8
Method (Test Design)	8
Test Environment	8
Test Monitor Role	8
Evaluation Measures	8
Task Description	9
Usability testing for effectiveness and efficiency	12
Usability testing for Satisfaction	13
Questionnaire	13
User 1	13
User 2	14
User 3	16
User 4	17
User 5	18
<b>QA Test Plan</b>	<b>20</b>
<b>Code Review</b>	<b>28</b>
Coding style	28
Code Peer Reviews	28
Reviews from the same team	28
Reviews from the different team	29
Explanation to our choice of code	30
<b>Self-check on Best Practices for Security</b>	<b>31</b>
Major Assets under protection	31



Encrypted Passwords in the DB	31
Input Data Validation	31
<b>Adherence to Original Non-functional Specs</b>	<b>33</b>
System Requirements	33
Performance Requirements	33
Storage, Security and Environmental Requirements	33
Marketing and Legal Requirements	34
Content	34
Privacy	35
<b>List of Team Contributions</b>	<b>36</b>

## Product Summary

Name

**YourSports**

Major Committed Functions

Priority 1

Regular user

1. Regular users shall be able to create an account.
2. Regular users shall be able to sign up as a creator account.
3. Regular users shall be able to access the homepage.
4. Regular users shall be able to view the live scores of a game.
5. Regular users shall be able to view the duration of a game.
6. Regular users shall be able to view updated statistics of players.
7. Regular users shall be able to search for articles through a search bar.
8. Regular users shall be able to filter articles they would like to read by keywords.
9. Regular users shall be able to filter articles they would like to read by different types of sports.
10. Regular users shall be able to search articles by heading of the article.
11. Regular users shall be able to view the article heading in the search results for articles.
12. Regular users shall be able to view the article author in the search results for articles.
13. Regular users shall be able to view the article postdate in the search results for articles.
14. Regular users shall be able to view the article image as a thumbnail in the search results for articles.
15. Regular users shall be able to search games by the teams playing in the search bar.
16. Regular users shall be able to filter games by the sport type.
17. Regular users shall be able to filter games by their date.
18. Regular users shall be able to view the game score in the search results for games.

- 
- 19. Regular users shall be able to view the game date in the search results for games.
  - 20. Regular users shall be able to view the game location in the search results for games.
  - 21. Regular users shall be able to search for player statistics according to the player name in the search bar.
  - 22. Regular users shall be able to filter player statistics by the sport type.

#### Registered user

- 23. Registered users shall have the same privileges as regular users.
- 24. Registered users shall be able to log in.
- 25. Registered users shall be able to log out.
- 26. Registered users shall be able to reset their password.
- 27. Registered users shall be able to delete their account.
- 28. Registered users shall be able to post in the discussion forum of the game.
- 29. Registered users shall be able to delete their posts in the discussion forum.
- 30. Registered users shall be able to comment on articles.

#### Basic Account

- 31. Accounts shall be able to be deleted by users.
- 32. Accounts shall contain the name of the users.
- 33. Accounts shall contain the username of the users.
- 34. Accounts shall contain the email of the users.
- 35. Accounts shall contain the password of the users.

#### Creator Account

- 36. Creators shall have the same privileges as registered users.
- 37. Creators shall be able to upload articles to the website.
- 38. Creators shall be able to view Dashboard.

#### Admin Account

- 39. Admin shall have all the privileges.
- 40. Admin shall be able to create discussion forums for upcoming games.

#### Articles

- 41. Articles shall be posted by creator users.
- 42. Articles shall be able to be filtered by sports type.
- 43. Articles shall be able to be filtered by keywords specified by the user.
- 44. Articles shall contain a heading.

- 
- 45. Articles shall contain a subheading.
  - 46. Articles shall contain an introduction.
  - 47. Articles shall contain contents.
  - 48. Articles shall contain a conclusion.
  - 49. Articles shall contain an image.

## Games

- 50. Games shall be filtered by date.
- 51. Games shall be filtered by sport type.
- 52. Games shall be filtered by keywords specified by users in the search bar.
- 53. Games shall have an individual discussion forum.
- 54. Games shall be able to show live scores of corresponding teams.
- 55. Games shall be able to show the date of the game.
- 56. Games shall be able to show the location of the game.

## Discussion Forum

- 57. Discussion forum shall be able to be viewed by all users.
- 58. Discussion forum shall be able to show posts from users.

## Dashboard

- 59. Dashboard shall be able to show the number of views of the article.
- 60. Dashboard shall be able to show the heading of the article.
- 61. Dashboard shall be able to show the posted date of the article.

## Backend Service

- 62. The backend service shall be able to validate the email for duplicates when users try to sign up.
- 63. The backend service shall be able to check for duplicate accounts when users try to sign up.
- 64. The backend service shall be able to create an account in the database.
- 65. The backend service shall be able to store posts by a user in the discussion forums.
- 66. The backend service shall be able to update player statistics in the database.
- 67. The backend service shall be able to filter articles specified by the user.
- 68. The backend service shall be able to filter games specified by the user.
- 69. The backend service shall be able to filter player statistics specified by the user.

## Website

- 
- 70. The website shall be able to check if the name is not empty when users try to sign up.
  - 71. The website shall be able to check if the email is not empty when users try to sign up.
  - 72. The website shall be able to check if the password is at least 8 characters before the users try to sign up.
  - 73. The website shall be able to check if the password is a maximum of 20 characters before the users try to sign up.
  - 74. The website shall be able to check if the terms of service have been accepted when users try to sign up.

#### Comment

- 75. Comments shall be able to be viewed by regular users.
- 76. Comments shall be posted by registered users.
- 77. Comments shall be able to be deleted by poster.

#### Unique Features

Our product will offer countless services to users who are interested in sports information and are sports fans. Users will be able to share their opinions with each other through news articles. Other users can read news articles, and they can also react to news articles through comments, likes, and dislikes from other users. Users who have uploaded their articles can also view different statistics like the number of likes, number of dislikes, and number of views that their article has received. These statistics will give the Author of the article, how their article is being perceived by other users of the product.

Our product also allows users to interact with other users who are watching the same sports game through the discussion forums provided for each game. Users can share their opinions on the game, and interact with other users who have the same interests. This will serve as a great way to build a community of users with similar interests through sports.

Finally, our product keeps up to date with player statistics, so that users can access the latest information on their favorite players at their fingertips.

#### URL

<http://34.136.124.189:3000>

## Usability Test Plan

### Test Plan

#### Purpose

The purpose of these tests is to test the usability of the superior feature of our product, where regular users can sign up using a creator account and upload their own articles, and how other users interact with those articles. A creator account has the privilege of uploading its own articles that contain sports information. Other users can search these articles, and users can read the articles, and based on their perception of the information given in the articles, they can leave comments, likes, and dislikes. The author who has signed up as a creator can view the comments the article has received, as well as check the number of views, likes, and dislikes their article has received. These statistics will give the author an idea of how other users are reacting to the information given in the article. The comments that other users leave will give detailed information to the author.

#### Problem Statement and Objectives

The problem is to find out how users find the usability of uploading articles, reading, and reacting to articles. The objective to test the task list mentioned above is as follows:

1. How did regular users find the usability of signing up as a creator.
2. If a registered user who has a creator account wants to upload their article, how smooth was their experience.
3. How satisfied was the user when they searched for articles.
4. How satisfied was the user with how the information given in the article was presented and were they able to express their opinion through the controls given by the website.
5. Were registered users who have creator accounts, easily able to get detailed and easy-to-understand statistics on how other users are reacting to their articles.

#### Task List

1. Regular users can sign up as a creator.
2. Registered users who have a creator account can upload their own articles.
3. Users can search articles.



4. Users can read, like, or dislike and add comments to the article.
5. Registered users who have a creator account, can view the different statistics of their article, like number of likes, number of dislikes, and number of views.

## User Profile

The profile of our user is as follows:

1. The user does not use technology regularly. They only use it when required, or when they access information.
2. The user uses a computer to stream through different websites that are commonly used like Google, Facebook, or any other social media website.
3. The user does not understand scroll through any other website, because they don't like to be overwhelmed with information.

All the users will start from the homepage of the website.

## Method (Test Design)

Each of the five functions in the task list will be tested for effectiveness, efficiency, and user satisfaction.

## Test Environment

All the users will test the product in the comfort of their own homes. Once they are done testing, they will send their feedback via email.

## Test Monitor Role

We won't have a test monitor however the team lead will regularly check with the intended users, on how much they have tested their app. The team leader expects the intended users to finish their testing and receive feedback within 48 hours of asking users to volunteer as intended users of the product.

## Evaluation Measures

Each of the five functions in the task list will be tested for effectiveness, efficiency, and user satisfaction.

The effectiveness will be measured with the following metrics:

1. How useful is the feature.
2. How easy is it to navigate.
3. Does it serve any purpose to the target audience.

The efficiency will be measured with the following metrics:

1. How much time did it take to complete the task.



2. How many pages did they have to go through to complete the task.
3. How helpful was the website when they tried to complete the task.

The satisfaction will be measured with the following metrics:

1. How the user felt after completing the task.
2. Did the user have any more expectations from the product.

## Task Description

### Task 1

Task	Description
Regular Users will sign up as a creator	A Registered user will try to sign up as a creator.
System Setup	Users will access the website on a laptop.
Starting Point	Users will be on the homepage
Intended Users	The intended users are regular users who have not signed up to the YourSports website.
The URL of the system that will be tested is	<a href="http://34.136.124.189:3000/SignUp">http://34.136.124.189:3000/SignUp</a>
What is to be measured	<ol style="list-style-type: none"> <li>1) Did the user find errors while trying to sign up as a creator.</li> <li>2) How much time did it take, and did they find the information asked during the signup useful.</li> <li>3) How much time did it take to complete the entire task.</li> </ol>

### Task 2

Task	Description
Registered Creators can upload their own articles	Users will upload an article to the YourSports Website.
System Setup	Users will access the website on a laptop.
Starting Point	Users will be on the homepage
Intended Users	The intended users are registered users who have signed up as a creator.
The URL of the system that will be tested is	<a href="http://34.136.124.189:3000/UploadArticle">http://34.136.124.189:3000/UploadArticle</a>
What is to be measured	<ol style="list-style-type: none"> <li>1) Did the user find errors while trying to upload an article.</li> </ol>



	<p>2) Were they able to fill in all the information they wanted to convey, and were they able to upload the image of their choice.</p> <p>3) How much time did it take to complete the entire task.</p>
--	---

### Task 3

Task	Description
Users can search articles	Users will search for an article.
System Setup	Users will access the website on a laptop.
Starting Point	Users will be on the homepage
Intended Users	The intended users are regular users who want to access sports information.
The URL of the system that will be tested is	<a href="http://34.136.124.189:3000/Home">http://34.136.124.189:3000/Home</a>
What is to be measured	<p>1) Was the user able to perform the search in the right search bar.</p> <p>2) Was the user able to use the filters provided easily.</p> <p>3) Did the user know what to do if they got 0 search results.</p> <p>4) Were the users able to access information to their liking and did they face any errors when performing the search.</p> <p>5) How much time did it take to complete the entire task.</p>

### Task 4

Task	Description
Users can read, like, or dislike and add comments to the article.	The user will add a comment and like the article.
System Setup	Users will access the website on a laptop.
Starting Point	Users will be on the homepage
Intended Users	The intended users are registered users who want to read the information given in the article.
The URL of the system that will be tested is	<a href="http://34.136.124.189:3000/ArticleView">http://34.136.124.189:3000/ArticleView</a>
What is to be measured	<p>1) Was the user able to open an article from the search results.</p>

	<p>2) Was the information given in the article visually appealing.</p> <p>3) Was the user able to find the like/dislike button easily.</p> <p>4) Was the user able to find the comments section easily.</p> <p>5) Was the user able to post a comment or like an article if they were not logged in.</p> <p>6) Did the user face any errors when trying to like/dislike the article or add a comment.</p> <p>7) Was the user satisfied with the visual structure of the information given in the article, the like and dislike button, and the comments section.</p> <p>8) How much time did it take to complete the entire task.</p>
--	---

### Task 5

Task	Description
Registered Creators can view all the statistics of their article; such as, number of likes, dislikes, and views	Registered users will be able to see the statistics on the articles they have uploaded.
System Setup	Users will access the website on a laptop.
Starting Point	Users will be on the homepage
Intended Users	The intended users are registered users who have uploaded articles to the website and want to check the statistics.
The URL of the system that will be tested is	<a href="http://34.136.124.189:3000/Dashboard">http://34.136.124.189:3000/Dashboard</a>
What is to be measured	<p>1) Was the user able to find the Statistics of their uploaded articles easily.</p> <p>2) Were the statistics visually appealing. And if the statistics are useful.</p> <p>3) Did they face any errors when checking the statistics.</p> <p>4) How much time did it take to complete the entire task.</p>



## Usability testing for effectiveness and efficiency

Test/ Use case	% Completed	Errors	Comments	No. of Pages	Time taken to complete the task
sign up	100	Found no errors	I don't know what's the benefit of registering as a creator. I didn't choose that part.	2	2 minutes
upload articles	70	Cannot change status to creator after signing up	Not allowed to change to a creator after I signed up as a regular user, I need to sign up a new account to upload articles.	2	7 minutes
search articles	90	Articles cannot found by author	Doesn't support search articles by author name	1	30 seconds
View/Review articles	90	Cannot edit posted comment	It is not easy to find that I can post comment until I scroll down and get to the bottom of the view article page	2	2 minutes
View self article statistics	100	Found no errors	It is easy to get the statistics of my published articles	2	1 minute



## Usability testing for Satisfaction

### Questionnaire

Place a tick in the box, for every question in the questionnaire. For the column of Any extra feedback, add your feedback using text.

User 1

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Any extra feedback
I found it easy to find the sign up page.				✓		
I found the information required for the sign-up was required.				✓		
I found it easy to select the option that requires me to sign up as a creator.			✓			The checkbox was there but I wasn't sure what it meant to be a creator
I found it easy to find the Upload Article Page.				✓		
I found it easy to fill in all the information.				✓		
I was able to select the image and it looks good with the whole article.		✓				The image to the side of the article felt a bit out of place
I was able to read the text typed in the search bar easily.			✓			
I found the filters convenient and useful when searching for relevant sports information.				✓		It is nice to filter between sports



I was able to navigate through the search results easily.			✓			
I think the like and dislike buttons were clearly visible.				✓		They were visible but they were a bit offset
I think the comments were clearly visible at first glance.		✓				
I was able to read through the article easily, without straining my eyes.				✓		
I was easily able to find the statistics page.		✓				I didn't realize dashboard had the statistics
I was able to see the statistics of the articles clearly, and they were presented in an organized manner and descriptive manner.				✓		
I found the statistics useful.			✓			It's nice to see but they don't really matter to me

## User 2

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Any extra feedback
I found it easy to find the sign up page.				✓		
I found the information required for the sign-up was required.				✓		
I found it easy to select the option that requires me to sign up as a creator.			✓			Check box was a bit small and almost missed it



I found it easy to find the Upload Article Page.					✓	
I found it easy to fill in all the information.				✓		
I was able to select the image and it looks good with the whole article.			✓			Image was a bit stretched on some articles
I was able to read the text typed in the search bar easily.			✓			Text was small at times
I found the filters convenient and useful when searching for relevant sports information.				✓		
I was able to navigate through the search results easily.				✓		
I think the like and dislike buttons were clearly visible.				✓		
I think the comments were clearly visible at first glance.					✓	
I was able to read through the article easily, without straining my eyes.				✓		
I was easily able to find the statistics page.					✓	
I was able to see the statistics of the articles clearly, and they were presented in an organized manner and descriptive manner.					✓	
I found the statistics useful.			✓			A bit useful but not necessary



## User 3

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Any extra feedback
I found it easy to find the sign up page.					✓	
I found the information required for the sign-up was required.				✓		
I found it easy to select the option that requires me to sign up as a creator.				✓		
I found it easy to find the Upload Article Page.				✓		
I found it easy to fill in all the information.				✓		
I was able to select the image and it looks good with the whole article.			✓			
I was able to read the text typed in the search bar easily.				✓		
I found the filters convenient and useful when searching for relevant sports information.				✓		
I was able to navigate through the search results easily.				✓		
I think the like and dislike buttons were clearly visible.					✓	
I think the comments were clearly visible at first glance.					✓	
I was able to read through the article easily, without straining my eyes.				✓		
I was easily able to find the statistics page.			✓			
I was able to see the statistics of the articles clearly, and they were presented in an organized manner and descriptive manner.			✓			



I found the statistics useful.					✓	
--------------------------------	--	--	--	--	---	--

## User 4

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Any extra feedback
I found it easy to find the sign up page.					✓	
I found the information required for the sign-up was required.					✓	
I found it easy to select the option that requires me to sign up as a creator.				✓		It is easy to select, but maybe a little bit too easy. Everyone can be a creator then why wouldn't we all be creator
I found it easy to find the Upload Article Page.			✓			
I found it easy to fill in all the information.				✓		
I was able to select the image and it looks good with the whole article.	✓					
I was able to read the text typed in the search bar easily.		✓				
I found the filters convenient and useful when searching for relevant sports information.	✓					Having filters are great, but being unable to see what filter is currently on is not useful
I was able to navigate through				✓		



the search results easily.						
I think the like and dislike buttons were clearly visible.			✓			They do look clear, but it stands out too much
I think the comments were clearly visible at first glance.		✓				
I was able to read through the article easily, without straining my eyes.			✓			
I was easily able to find the statistics page.				✓		
I was able to see the statistics of the articles clearly, and they were presented in an organized manner and descriptive manner.		✓				Par chart is cool, but the information on it is not representing informantions in a useful way
I found the statistics useful.	✓					Views count should be separated from like and dislike

## User 5

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Any extra feedback
I found it easy to find the sign up page.					✓	
I found the information required for the sign-up was required.				✓		
I found it easy to select the option that requires me to sign up as a creator.				✓		



I found it easy to find the Upload Article Page.					✓	
I found it easy to fill in all the information.					✓	
I was able to select the image and it looks good with the whole article.			✓			
I was able to read the text typed in the search bar easily.				✓		
I found the filters convenient and useful when searching for relevant sports information.			✓			
I was able to navigate through the search results easily.				✓		
I think the like and dislike buttons were clearly visible.			✓			
I think the comments were clearly visible at first glance.		✓				
I was able to read through the article easily, without straining my eyes.			✓			
I was easily able to find the statistics page.					✓	
I was able to see the statistics of the articles clearly, and they were presented in an organized manner and descriptive manner.					✓	
I found the statistics useful.				✓		



## QA Test Plan

- Search Results should be displayed within 5 seconds.

### Test Objectives

The objective of this test is to test how fast search results are retrieved from the database and displayed when a user performs a search on the website.

### Hardware and Software Setup

The hardware setup required for this test is a basic computer where the website will be run and the tests will be performed. In our case, we will be using a MacBook Pro 13-inch to run our website.

The Software setup will be the website with the base URL  
<http://34.136.124.189:3000>.

The URL where we will perform our tests is <http://34.136.124.189:3000/Home>.

### Feature to be tested

The feature to be tested is the efficiency and speed of our search results.

### QA Test Plan

No.	Description	Test Input	Expected Output	Pass/Fail
1	Testing User will type "Steph Curry" in the search bar that is present on the homepage.	"Steph Curry" (type: String)	3 search results, where each article heading contains "Steph Curry". The results should be displayed within 5 seconds.	Pass
2	Testing User will type "Harden" in the search bar that is present on the homepage.	"Harden" (type: String)	18 search results, where each article heading contains "Harden". The results should be displayed within 5 seconds.	Pass
3	Testing User will type " " in the search bar that is present on the homepage.	" " (type: String)	37 search results, where all articles are returned from the database, with the message saying that, since the search provided was empty, all articles were returned. The results should be displayed within 5 seconds.	Pass



2. Text inputs should have a minimum of 1 character and a maximum of 250 characters, and the input should be trimmed before processing.

### Test Objectives

The objective of this test is to test if the input data provided by a user is validated before processing from the backend.

### Hardware and Software Setup

The hardware setup required for this test is a basic computer where the website will be run and the tests will be performed. In our case, we will be using a MacBook Pro 13-inch to run our website.

The Software setup will be the website with the base URL

<http://34.136.124.189:3000>.

The URL where we will perform our tests is

<http://34.136.124.189:3000/ArticleView>.

### Feature to be tested

The feature to be tested is the security and validation of data before processing.

### QA Test Plan

No.	Description	Test Input	Expected Output	Pass/Fail
1	Testing User will type "Warriors win again" in the input text box of the comments section that is present on the article view page of an article.	"Warriors win again" (type: String)	A new comment will appear in the Comments section that has the content "Warriors win again".	Pass
2	Testing User will type " " in the input text box of the comments section that is present in the article view page of an article.	" " (type: String)	No comment will be added to the Comments section.	Pass
3	Testing User will type "Warriors" 250 times with a space between each word in the input text box of the comments section that is present on the article view page of an article.	"Warriors" (type: String). The input will be typed 250 times, with a space in between each word.	No comment will be added to the Comments section.  An error message will pop up stating "Max number of characters is 250".	Fail



3. Articles uploaded by a creator user shall need to follow community guidelines provided by the website.

### Test Objectives

The objective of this test is to test if the input data provided by a user is validated and checked by the backend, to ensure compliance with community guidelines.

### Hardware and Software Setup

The hardware setup required for this test is a basic computer where the website will be run and the tests will be performed. In our case, we will be using a MacBook Pro 13-inch to run our website.

The Software setup will be the website with the base URL

<http://34.136.124.189:3000>.

The URL where we will perform our tests is

<http://34.136.124.189:3000/UploadArticle>.

### Feature to be tested

The feature to be tested is the security, validation, safety, and compliance of input data provided by a user.

### QA Test Plan

No.	Description	Test Input	Expected Output	Pass/Fail
1	Testing User will type Sports Information in the Heading, subHeading, Introduction, Content, and Conclusion section.	Heading: "Test Heading" (type: String).  SubHeading: "Test SubHeading" (type: String).  Introduction: "Test Introduction" (type: String).  Content: "Test Content" (type: String).  Conclusion: "Test Conclusion" (type: String).	A new article will be uploaded.	Pass
2	Testing User will use profane language in the Heading,	Heading: "**** *** Heading" (type: String).	A new article will not be uploaded.	Fail

	subHeading, Introduction, Content, and Conclusion section.	SubHeading: "**** *** SubHeading" (type: String).  Introduction: "**** *** Introduction" (type: String).  Content: "**** *** Content" (type: String).  Conclusion: "**** *** Conclusion" (type: String).  "**** ***" is a profane word.	A message will pop up, stating the information given in the article does not follow community guidelines.	
3	Testing User will use an NSFW image as a thumbnail for the article.	The input image is an NSFW image.	A new article will not be uploaded.  A message will pop up, stating that the information given in the article does not follow community guidelines.	Fail



4. The website shall be resizable and compatible across different screen sizes and operating systems.

### Test Objectives

The objective of this test is to test if the website is resizable and compatible across different devices with different screen sizes.

### Hardware and Software Setup

The hardware setup required for this test is a basic computer or phone where the website will be run and the tests will be performed. We will be using a MacBook Pro 13-inch, an Iphone12 Pro, and an Alienware x14.

The Software setup will be the website with the base URL  
<http://34.136.124.189:3000>.

The URL where we will perform our tests is <http://34.136.124.189:3000/Home>.

### Feature to be tested

The feature to be tested is the compatibility and resizability of the website.

### QA Test Plan

No.	Description	Test Input	Expected Output	Pass/Fail
1	<p>Testing User will open the homepage, with some search results.</p> <p>The page will be tested on a MacBookPro 13-inch screen that has a macOS operating system.</p>	<p>Screen: 13-inch Macbook Pro            Operating System: macOS Sierra</p> <p>Page URL:  <a href="http://34.136.124.189:3000/Home">http://34.136.124.189:3000/Home</a></p> <p>Search Bar Input: "Steph Curry"</p> <p>Number of Search Results: 3</p>	<p>3 search results are displayed which contain the heading, author, article post date, and the thumbnail of the article.</p>	Pass
2	<p>Testing User will open the homepage, with some search results.</p> <p>The page will be tested on a 34- screen that has a Windows operating system.</p>	<p>Screen: 34-inch screen            Operating System: Windows 11</p> <p>Page URL:  <a href="http://34.136.124.189:3000/Home">http://34.136.124.189:3000/Home</a></p> <p>Search Bar Input: "Steph Curry"</p> <p>Number of Search Results: 3</p>	<p>3 search results are displayed which contain the heading, author, article post date, and the thumbnail of the article.</p>	Pass



3	<p>Testing User will open the homepage, with some search results.</p> <p>The page will be tested on a phone screen that has an IOS operating system.</p>	<p>Screen: iPhone 12 Pro, 2.82x5.78 inches.</p> <p>Operating System: IOS 15.5</p> <p>Page URL: <a href="http://34.136.124.189:3000/Home">http://34.136.124.189:3000/Home</a></p> <p>Search Bar Input: "Steph Curry"</p> <p>Number of Search Results: 3</p>	<p>3 search results are displayed which contain the heading, author, article post date, and the thumbnail of the article.</p>	Fail
---	--	--	---	------



5. The YourSports API shall not accept more than 10 requests per 10 seconds from the same IP address.

### Test Objectives

The objective of this test is to test if the YourSports API can protect itself from overloading of network traffic, and prevent starvation of the resources provided by the virtual machine from a single user.

### Hardware and Software Setup

The hardware setup required for this test is a basic computer where the website will be run and the tests will be performed. In our case, we will be using a MacBook Pro 13-inch to run our website.

The Software setup will be Postman. Postman will be used to test the API.

The URL where we will perform our tests is

<http://34.136.124.189:8080/api/searchnews/search>.

### Feature to be tested

The feature to be tested is the security, availability, and rate-limiting feature of the YourSports API.

### QA Test Plan

No.	Description	Test Input	Expected Output	Pass/Fail
1	Testing User will make 5 API calls to the YourSports API within 10 seconds from one device.	Testing Application: Postman API URL: <a href="http://34.136.124.189:8080/api/searchnews/search">http://34.136.124.189:8080/api/searchnews/search</a> Request Type: POST Number of calls: 5 Duration: 10 seconds Number of devices: 1	All API calls will succeed, and the requested results will be returned.	Pass
2	Testing User will make 14 API calls to the YourSports API within 10 seconds from two different devices.	Testing Application: Postman, API URL: <a href="http://34.136.124.189:8080/api/searchnews/search">http://34.136.124.189:8080/api/searchnews/search</a>	All API calls from device one will return the requested results.	Pass



	<p>Device one will make 7 API calls within 10 seconds.</p> <p>Device two will make 7 API calls within 10 seconds.</p>	<p>Request Type: POST</p> <p>Number of calls: 14</p> <p>Duration: 10 seconds</p> <p>Number of devices: 2</p>	<p>All API calls from device two will return the requested results.</p>	
3	<p>Testing User will make 15 API calls to the YourSports API within 10 seconds from one device</p>	<p>Testing Application: Postman,</p> <p>API URL: <a href="http://34.136.124.189:8080/api/searchnews/search">http://34.136.124.189:8080/api/searchnews/search</a></p> <p>Request Type: POST</p> <p>Number of calls: 15</p> <p>Duration: 10 seconds</p> <p>Number of devices: 1</p>	<p>The API will return the requested results until the 10th API call.</p> <p>On the 11th call, the API will return an error message stating "Too much traffic from the same IP address, please try again later".</p>	Fail

## Code Review

### Coding style

We separate the responsibilities for different functions in different files. We used longer function names to make it much easier to understand as later we debug or enhance the code. Proper alignment helps us easily to find which statements are controlled by which control structures. For the statements, we also group statements together and placing a blank line between groups.

### Code Peer Reviews

Reviews from the same team

#### Code Review Request #10

Open wenyeguo opened this issue 2 hours ago · 1 comment

wenyeguo commented 2 hours ago

Please review the sql function which used for search news articles:  
Branch: database  
File name: StoredProcedureGetNewsArticles.sql  
Link to the file: <https://github.com/sfsu-joseo/csc648-848-01-sw-engineering-summer-22-team02/blob/database/application/database/StoredProcedureGetNewsArticles.sql>

wenyeguo added the **help wanted** label 2 hours ago

wenyeguo assigned Kevin-Islas 2 hours ago

Kevin-Islas commented 6 minutes ago

- The logic has no issues.
- Adding inline comments would make it easier for others to understand.
- Formatting is consistent and easy to follow.

Assignees	Kevin-Islas
Labels	<b>help wanted</b>
Projects	None yet
Milestone	No milestone
Development	Create a branch for this issue or link a pull request.
Notifications	Customize <a href="#">Unsubscribe</a> You're receiving notifications because you authored the thread.

#### Front-end Code Review #12

Open Kevin-Islas opened this issue 4 minutes ago · 1 comment

Kevin-Islas commented 4 minutes ago

Review requested:  
Please review the search and filter components in InputSubmissions.js  
Branch: front-end  
Path to file to be reviewed:  
csc648-848-01-sw-engineering-summer-22-team02/front-end/application/front-end/src/InputSubmission.js

wenyeguo commented 1 minute ago

1. The overall structure is clear and the naming is easy to understand.
2. The format looks good and consistent.
3. It will be better to add a header to introduce what is this file used for.
4. Add more inline comments to make the code more readable.

Assignees	No one—assign yourself
Labels	None yet
Projects	None yet
Milestone	No milestone
Development	Create a branch for this issue or link a pull request.

## Reviews from the different team

```
application > front-end > src > JS InputSubmission.js > ...
1   import React, { useState, useEffect } from "react";
2   import axios from "axios";
3   import "./Navbar.css";
4   import ArticleView from "./ArticleView";
5   import "./Footer";
6
7   /*Function receives user input and calls api to return articles*/
8   function InputSubmission() {
9     const [data, setData] = useState("");
10    const [InputText, setInputText] = useState("");
11    const [date, setDate] = useState("");
12    const [sport, setSport] = useState("");
13    const [length, setLength] = useState("");
14    const [searchApplied, setSearchApplied] = useState(false);
15
16    function handleSearch() {
17      if (InputText === "" && sport === "") {
18        setSearchApplied(false);
19      } else {
20        setSearchApplied(true);
21      }
22      var config = {
23        method: "post",
24        url: "http://34.136.124.189:8080/api/searchnews/search",
25        data: {
26          sport: sport,
27          searchText: InputText,
28          date: date,
29          length: length,
30        },
31      };
32      axios(config)
33        .then(function (response) {
34          setData(response.data);
35          console.log(response.data.sport);
36          console.log(response.data.length);
37          console.log(response.data);
38        })
39        .catch(function (error) {
40          console.log(error);
41        });
42    }
}
```

### Team 06's review:

"At the beginning of the file, the comment "/\*Function receives user input and calls api to return articles\*/" was extremely helpful as it was a clear and concise comment that explained exactly what the following function does.

Another great comment is "/\*When no filters applied all articles will be displayed \*/", for similar reasons to what was stated above, as it helps give the reader context on what the following code does.

While this is technically not required, our team believes that having a "header" comment with a brief description of the purpose of the whole file itself, will help readers have a better understanding of the following code as it will be put into context.

While our team is familiar with react code, others reading the code that have never used react before may be confused when it comes to react specific lines of code. For example the use-state hooks. Some clarifying comments would be helpful.

Overall indentation, organization and formatting of code is very well done.

The logic of the functions make sense and the way your team is displaying data to the front end checks out. Great job on these aspects of the code."

```
application > database > StoredProcedureGetNewsArticles.sql
1  Use YourSportsDB;
2
3  DELIMITER $$ 
4  CREATE PROCEDURE `GetNewsArticles`
5  (IN inPostdate VARCHAR(255),
6   IN inSport VARCHAR(255),
7   IN inSearchText VARCHAR(255))
8
9  BEGIN
10
11  SELECT Image_url,Heading,SubHeading,PostDate,Name as `Author`,Sport FROM
12  (SELECT Image_url,Heading,SubHeading,fk_Author_ID,PostDate,Sport FROM Article WHERE
13  (PostDate >= inPostdate OR inPostdate is NULL)
14  AND
15  (Sport = inSport OR inSport is NULL )
16  AND
17  (Heading LIKE CONCAT('%',inSearchText,'%') OR inSearchText is NULL)
18  ) AS Article JOIN Account ON Account_ID = Article.fk_Author_ID ORDER BY PostDate DESC;
19
20  END$$
21  DELIMITER ;
```

Team 06's review:

"The first thing our team noticed regarding the code is it does not have any in-line comments, nor an overall description to explain the purpose of this file. We recommended commenting on the code even though it is a .sql file, as the comments would help those who are unfamiliar with SQL code, such as a front-end developer, or someone learning the technology.

The indentation, spacing, formatting of this SQL code is excellent, and makes it easy to follow along for the reader.

We could not find any problems with the logic/process of the SQL code."

### Explanation to our choice of code

This is the most commonly used function. We believe that these codes are the most representative of our code style.

## Self-check on Best Practices for Security

### Major Assets under protection

Passwords - by storing encrypted passwords

Image assets - by setting the Google Cloud Storage bucket to public read-only

User data - by not collecting any data on the client and server side, and making sure the user's password is always encrypted.

### Encrypted Passwords in the DB

We encrypt passwords using the MD5 hash function that is provided in MySQL. An MD5 hash is created by taking a string of any length and encoding it into a 128-bit fingerprint. Encoding the same string using the MD5 algorithm will always result in the same 128-bit hash output. MD5 hashes are commonly used with smaller strings when storing passwords, credit card numbers, or other sensitive data in databases such as the popular MySQL.

Here is a screenshot, where we encrypt passwords provided by the user in the backend itself.

```
query = 'INSERT INTO RegisteredUser (name,username,email,password) VALUES ("'+name+'", "'+username+'", "'+email+'", MD5("'+password+'"));';
con.query(query, (error, results, fields) => {
  if (error) {
    res.json(error);
  }
  res.json("Account Created Successfully");
});
```

### Input Data Validation

Data Validation during registration

**Name** - It should contain at least 1 character that is not a space.

**Username** - It should contain at least 1 character that is not a space.

**Email** - It should contain at least 1 character that is not a space and it should follow the regular expression rules of an email

**Password** - It should be 8-20 characters long.

**Terms of Service** - It should be validated and checked if it is ticked, before registration.

We have converted all of our input data to lower case for safety and accurate results, however, that is not required since MySQL automatically handles the upper case and lower case checking when searching for data.

```

let name = req.body.name == null || req.body.name == "" ? null : req.body.name.trim().toLowerCase();
let username = req.body.username == null || req.body.username == "" ? null : req.body.username.trim().toLowerCase();
let email = req.body.email == null || req.body.email == "" ? null : req.body.email.trim().toLowerCase();
let password = req.body.password == null || req.body.password == "" ? null : req.body.password;
let isCreatorAccount = req.body.isCreatorAccount == null || req.body.isCreatorAccount == false ? null : true;
let termsOfServiceAgreed = req.body.termsOfServiceAgreed == null || req.body.termsOfServiceAgreed == false ? null : true;

if (termsOfServiceAgreed==true)
{
if (name!=null && username!=null && email!=null && password!=null)
{
if (email.match(
|^(([^<>(){}\\.,;:\\s@"]+\\.(^<>(){}\\.,;:\\s@"]+)*|(.+"))@((\\[[0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}\\.)|(([a-zA-Z\\-0-9]+\\.)+[a-zA-Z]{2,}))$|
))
{
if(password.length>=8 && password.length<=20)
{

query = 'INSERT INTO RegisteredUser (name,username,email,password) VALUES ("'+name+'","'+username+'","'+email+'",MD5("'+password+'"));';
con.query(query, (error, results, fields) => {
    if (error) {
        res.json(error);
    }
    res.json("Account Created Succesfully");
});
}
else
{
    res.json("Please adhere to the password requirements")
}
}
else
{
    res.json("Please enter a valid email");
}
}
else
{
    res.json("Please fill in all the details");
}
}
else
{
    res.json("Please accept terms of service");
}
}

```

## Data Validation during search

**Input Search Text** - It should contain at least 1 character that is not a space. If it does not contain at least 1 character that is not space, it will be set to NULL. If all the inputs provided to the search bar are NULL, all the data from the database will be returned.

We have converted all of our input data to lower case for safety and accurate results, however, that is not required since MySQL automatically handles the upper case and lower case checking when searching for data.

Here is an example

```

let sport = req.body.sport == "" || req.body.sport == null ? 'NULL' : "'"+req.body.sport.trim().toLowerCase()+"'";
let searchText = req.body.searchText == "" || req.body.searchText == null ? 'NULL' : "'"+req.body.searchText.trim().toLowerCase()+"'";
let query = 'CALL getNewsArticles('+date+', '+sport+', '+searchText+');';

```

## Adherence to Original Non-functional Specs

### System Requirements

1. The system shall be hosted on a Google Cloud Compute Engine instance of 1 cpu core, 1gb RAM and 10GB storage size. (DONE)
2. A MySQL relational database shall be hosted in Google Cloud to store the data. (DONE)
3. NodeJS and express shall be used to build the backend server. (DONE)
4. React shall be used to host the front-end server. (DONE)
5. The front-end service shall run on port 3000. (DONE)
6. The back-end service shall run on port 8080. (DONE)
7. The Google Cloud compute engine shall use an Ubuntu 18.04.6 LTS. (DONE)
8. The code repository shall be hosted in Github. (DONE)
9. The front-end and backend service shall accept traffic from all ip-addresses. (DONE)
10. The Virtual Machine hosted in Google Cloud shall allow both HTTP and HTTPS traffic. (DONE)
11. The website shall be resizable and compatible across different screen sizes and operating systems. (DONE)

### Performance Requirements

12. If a registered user enters the wrong password, the system shall respond within 3 seconds of submitting their details for signup. (DONE)
13. All inserts to the database should happen within 5 seconds. (DONE)
14. The homepage shall open within 10 seconds. (DONE)
15. Search results in the homepage when searching for articles should happen within 7 seconds. (DONE)
16. The website shall show updates in a game every 5 seconds. (ISSUE: We need to set a listener in the website that fetches live updates every 5 seconds. The team is not familiar with Pub/Sub systems that will be used to implement this requirement)

### Storage, Security and Environmental Requirements

17. Unregistered Users shall only be able to register if their username is unique. (DONE)

- 
18. Unregistered Users shall only be able to register if their email is unique. (DONE)
  19. The statistics of the player will be updated every 24 hours. (ISSUE: We need to set a listener in the website that fetches live updates every 24 hours. The team is not familiar with Pub/Sub systems that will be used to implement this requirement)
  20. The YourSports API shall not accept more than 10 requests per 10 seconds from the same IP address. (DONE)
  21. Passwords of registered users shall be encrypted with MD5 hash function. (DONE)

## Marketing and Legal Requirements

22. Unregistered Users shall be able to accept the terms of service before signing up. (DONE)
23. Articles uploaded by a creator user, shall need to follow community guidelines provided by the website. (Issue: Very difficult to check if content uploaded by a user follows community guidelines. We will need to implement Natural Language technologies, and high end neural networks that can check if the information uploaded by the user is compliant with the community guidelines)

## Content

24. The password shall be hidden when typed when trying to sign up. (DONE)
25. The password shall be a minimum of 8 characters when trying to sign up. (DONE)
26. The registered user shall be redirected to the homepage if the user has logged in. (DONE)
27. Registered Users shall be prompted to log in if an account with the same email already exists. (DONE)
28. Unregistered users shall have to fill in the name with at least one character while signing up. (DONE)
29. Unregistered users shall have to fill in the username with at least one character while signing up. (DONE)
30. Unregistered users shall have to fill in the email with at least one character while signing up. (DONE)
31. Unregistered users shall have to fill in the password with at least one character while signing up. (DONE)

- 
- 32. If a Registered User has not put any filters in the news articles they are searching for, they shall receive all the news articles in the database. (DONE)
  - 33. Posts in the discussion forum shall have a minimum 1 character limit. (DONE)
  - 34. Comments in the Article view shall have at least one character. (DONE)
  - 35. The website logo shall be on the top left of every page. (DONE)
  - 36. Text inputs should have a minimum of 1 character and the input should be trimmed before processing. (DONE)

## Privacy

- 37. The email, username and password shall be collected and only used for authentication. (DONE)
- 38. Registered users shall have their log in details stored. (DONE)

## List of Team Contributions

### Kshitiz Sareen (Team Lead, Back-end Lead)

Organized team meetings and set team agendas.

### Kevin Islas (Front-end Lead)

Did the code review for team 06, defined the coding style, and selected the files that were to be sent for code review.

### Shamar Ireland (Front-end Developer)

Decided the tasks that used for the usability tests.

### Wenye Guo (Database Administrator)

Decided the questions asked in the questionnaire. Assessed the current status of the non-functional requirements. Performed the internal code review. Added the rules and code screenshots that validated our input data.

### Sabur Saigani (Functional Lead)

Added the unique features in the product summary, and helped in adding the functional requirements.

### Jonathan Ip (Functional Assistant)

Contributed to the finalized list of P1 requirements that were to be implemented. Fully responsible in formatting and organizing the documentation's layout as well as checking the documentation's grammar.

### Mathew O Abiola (Github Master)

Added the assets we would be protecting and all the encryption method used for the passwords.

## Product ScreenShots

### Article Upload page

Article Upload page

Select Sport

Evaluating James Harden 13 season career.

What pushes James Harden to his best performance in every season.

Choose Thumbnail

Choose File jameshardennew.jpeg



In this Article, we will talk about what keeps James Harden at the top and why he will go down as one of the greatest in the NBA.

On this date in 2019, James Harden went off for a career-high and franchise-record 61 points in a home win against the San Antonio Spurs. In 37 minutes, he also recorded 7 rebounds and 3 steals while shooting 19-of-34 shooting from the field, 9-of-13 from beyond the arc, and 14-of-17 from the free-throw line.

With this game, he joined the likes of Wilt Chamberlain, Michael Jordan, Kobe Bryant to have multiple 60-point games in a single season.

Upload

Sign up as a creator Page

Kshitiz

Sareen

kshitizsareen709@gmail.com

•••••••••••••

Password must be between 8-20 characters

Are you signing up as a creator:

Do you agree to the terms of service:

SignUp

[Have an account already? Login here!](#)

## Article Search Page

Welcome to YourSports, Enjoy a countless number of services ranging from Reading Sports Articles,

to enjoying live updates of games, and being updated with the latest player statistics.

LeBron



Filter Article By Sport

1 search results



LeBron James is the greatest Basketball Player

## Article View Page

kevin islas

2022-7-31



### Why Shohei Ohtani is already better than Babe Ruth

The Great bambino would not be calling any shots against this new phenom.





## Article Comments Page

Comments

**wenye**  
Good job!

**shamar ireland**  
This is by far the best thing I've heard today

**wenye**  
I like this article

**shamar ireland**  
cant disagree

**kevin islas**  
I believe Lebron is the greatest athlete to ever play basketball but I'm not sure if I would say he is the greatest ever

**jonathan**  
The greatest!!!

## Article Statistic Page

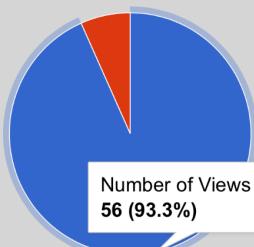
[Write New Article](#)

**My Articles**



**Yordan hand hits IL baseball**  
2022-7-31

**Aritcle Stats**



Category	Value
Number of Views	56 (93.3%)
Number of Likes	4 (6.7%)

- Number of Views
- Number of Likes



## Database Tables

### Account Table:

Account_ID	Name	Username	Email	Password	isCreator
28	kshitiz sareen	sareen	kshitizsareen709@g...	f45761459f8863468...	1
29	shamar ireland	shamarireland79	shamarireland79@h...	f849618fac31084ff0...	1
30	jonathan	ipchjoanthan	jip3@mail.sfsu.edu	25f9e794323b45388...	1
31	kevin islas	kevinislasm	kevin.islas04@gmail...	76a180b36896a0a...	1
32	wenyue	wenyeguo	wenyeguo@outlook...	25d55ad283aa400af...	1
33	sabur saigani	sabur	sa@gmail.com	29f6195b555c21fd7...	0
34	kshitiz	ksareen	kshitiz@gmail.com	f45761459f8863468...	0

### Article Table:

Article_ID	PostDate	Image_url	Heading	SubHeading	Introduction	Content	Conclusion	Sport	NumberOfViews	NumberOfLikes	NumberOfDislikes	fk_Author_ID
42	2022-0...	https://...	LeBro...	LeBron i...	LeBron...	Some p...	All in all h...	basketball	76	5	0	29
43	2022-0...	https://...	Sports	Have yo...	It has b...	Basketball	Basketbal...	basketball	50	2	3	30
44	2022-0...	https://...	Yorda...	Astros'...	Alvarez'...	Alvarez...	The Astro...	baseball	54	4	1	32
45	2022-0...	https://...	Why...	The Gre...	This ima...	Times h...	The realis...	baseball	50	3	1	31
46	2022-0...	https://...	Jame...	Lets talk...	Let me i...	James...	With ever...	basketball	28	4	1	28
47	2022-0...	https://...	dsjak...	ndsk S...	dnsksd...	nsdxknsd...	baseball	6	0	0	0	28

### Comment Table:

Comment_ID	Content	PostDate	fk_Article_ID	fk_Author_ID
60	Good job!	2022-08-01	42	32
61	This is by far the bes...	2022-08-01	42	29
62	I like this article	2022-08-01	42	32
63	cant disagree	2022-08-01	42	29
64	I believe Lebron is th...	2022-08-01	42	31
65	I need to get up off t...	2022-08-01	43	29
66	Basketball seems int...	2022-08-01	43	29

### Player Table:

Player_ID	PlayerName	PlayerType
1	Jaylen Brown	0
2	Steph Curry	0
3	AI Horford	0
4	Steven Adams	0
5	Yordan Alvarez	1

### Game Table:

Game_ID	TeamOne	TeamTwo	GameDate	GameLocation	TeamOneScore	TeamTwoScore	HasForum	SportType	teamOneURL	teamTwoURL
1	MIL	BKN	2021-10-19	Fiserv Forum	59	48	1	0	https://upload.wikim...	https://upload.wikim...
2	LAL	GS	2021-10-19	Crypto.com Ar...	53	56	1	0	https://upload.wikim...	https://upload.wikim...
3	CHA	IND	2021-10-20	Spectrum Cen...	57	57	1	0	https://upload.wikim...	https://upload.wikim...
4	DET	CHI	2021-10-20	Little Caesars...	41	44	1	0	https://upload.wikim...	https://upload.wikim...
5	NY	BOS	2021-10-20	Madison Squa...	64	62	1	0	https://upload.wikim...	https://upload.wikim...
6	TOR	WAS	2021-10-20	Scotiabank Ar...	39	45	1	0	https://upload.wikim...	https://upload.wikim...
7	MEM	CLE	2021-10-20	FedExForum	61	56	1	0	https://upload.wikim...	https://upload.wikim...

### Post Table:

Post_ID	Content	PostDate	fk_Account_ID	fk_Game_ID
35	Great, cannot wait to...	2022-08-01	29	208
36	Can't wait to see Da...	2022-08-01	31	208
37	cannot wait to see it	2022-08-01	32	208
39	i am really excited	2022-08-01	33	208

# Project Management

This screenshot shows a Trello workspace titled "CSC 648 Project Workspace". The left sidebar lists "Boards", "Members", "Settings", "Workspace views", "Tables", "Calendar", "Your boards", and "Project Tasks". The main board has sections for "Project Resources", "Milestone 1 resources", "Milestone 2 resources", "Milestone 3 resources", "M4 Resources", "Questions For Next Meeting", and "To Do".

- Project Resources:** Contains a "YourSports" document and a "CSC648-848 Summer 2022 Milestone3 Team02.pdf" file.
- Milestone 1 resources:** Contains a "Milestone 1 PDF" document.
- Milestone 2 resources:** Contains "Data Definitions", "UI Mockups", and "Wireframes".
- Milestone 3 resources:** Contains "Prioritized Functional Requirements for M3", "Product Summary", "Usability Tests", "QA Tests", and "Self-Check on Best Practices for Security".
- M4 Resources:** Contains "M4 reference", "Product Summary", "Usability Tests", "QA Tests", and "Self-Check on Best Practices for Security".
- Questions For Next Meeting:** Contains a "Front-end changes" card.
- To Do:** Contains cards for "Front-end changes", "Database Changes", "Screenshots of DB Tables", and "Screenshots of Trello Workspace".

This screenshot shows the same Trello workspace after some updates. The "Project Resources" section now includes a "Questionnaire" document. The "Milestone 3 resources" section has been renamed to "Milestone 3 resources" and contains "Prioritized Functional Requirements for M3" and "Wireframes". The "M4 Resources" section has been renamed to "M4 Resources" and contains "Product Summary", "Usability Tests", "QA Tests", and "Self-Check on Best Practices for Security". The "To Do" section has been renamed to "Pending" and contains cards for "Front-end changes", "Database Changes", "Screenshots of DB Tables", and "Screenshots of Trello Workspace". A new "Blocked" section has been added, containing a single card for "M4 Documentation".

**Database Changes**

in list To Do

Members: WG +

Description: Edit

Change Datatypes in Stored Procedures.  
Change Datatypes in Tables.  
Make changes to GetPosts Stored Procedure.

Activity:

Write a comment...  
Wenye Guo yesterday at 7:07 PM  
done

Add dropdowns, text fields, dates, and more to your cards.  
Start free trial

Power-Ups: + Add Power-Ups

Automation: + Add button

Actions: → Move, Copy, Make template, Watch, Archive, Share

**Adding new functional requirements**

in list Done

Members: JI, WG + Due date: Jul 13 at 5:00 PM Complete

Description: Edit

Priority function requirement

Activity:

Write a comment...  
JI

Add dropdowns, text fields, dates, and more to your cards.  
Start free trial

Power-Ups: + Add Power-Ups

Automation: + Add button

Actions: → Move, Copy, Make template, Watch, Archive, Share

**Done**

Documentation Grammar Check  
UI Mockup for Signing up Use case  
UI Mockup for Creating Group Chat use case  
UI Mockup for Checking Sports news Use case  
UI Mockup for Checking Twitter Feeds Use Case  
UI Mockup for Fantasy Betting Use Case  
UI Mockup for Posting News Articles  
Building ERD and EER  
Adding new functional requirements  
Changing the Mockups  
Building the front-end  
Creating wireframes  
Code Review Task For M4  
MS Cover Page and Product Summary  
+ Add a card



## List of Team Contributions

### Kshitiz Sareen (Team Lead, Back-end Lead)

Contribution Score: 10

Helped organize team meetings, and set team meeting agendas. Delegated tasks related to the backend and front-end and set deadlines. Helped in building the YourSports API, that connects the backend and the front-end

### Kevin Islas (Front-end Lead)

Contribution Score: 10

Helped implement APIs into the front end. Continued to make UI changes to further improve product usability. Improved scalability in different parts of the website. Helped test product usability.

### Shamar Ireland (Front-end Developer)

Contribution Score: 10

Helped assist with connecting the backend to the front end specifically with local storage of the user when they logged in and with the signup itself with displaying the data the user entered to create an account. Also helped with some last minute UI changes that we picked up on that needed to be fixed

### Wenye Guo (Database Administrator)

Contribution Score: 10

Works mainly on the database, update database procedures according to actual demands, helped to test the product usability

### Jonathan Ip (Functional Lead)

Contribution Score: 10

Fully responsible in formatting and organizing the documentation's layout as well as checking the documentation's grammar.

### Sabur Saigani (Database Developer)

Contribution Score: 7

Search and add games and players data to database, helped to test the product usability

### Mathew O Abiola (Github Master)

Contribution Score: 4

## Post Analysis

There are three main challenges that we as a team faced

1. Time Constraint: One of the biggest problems we faced as a team was to complete this entire project within two months. While our team was dedicated, and efficient, we faced numerous problems, when it came to managing time and deadlines, because of the enormous amount of tasks that we had to complete in each milestone. We started off with a lot of functional requirements, thinking that we could implement all of them including the priority two and priority three requirements. However, as time went along, we started facing issues in development and prioritization of the requirements. The prioritization process took us a long time. Finally, some team members had very busy schedules as a whole, and were not able to contribute as much as required to the project. To solve these issues, we started reducing our priority one requirement with each milestone, and the team as a whole was involved in making the decisions on which requirements were possible to implement by the end of the semester and how much they contribute to our project.
2. Front-end Challenges: One of the major issues we faced were multiple aspects in front-end development. Due to lack of experience, we were not sure how to build a website that prioritizes usability. This was one of the major reasons we lost a significant amount of points, during the horizontal prototype feedback. However, the feedback from the professor pushed us in the right direction, to build a website that has a good user experience. We started testing our websites in different browsers and different screen sizes, along with making sure each and every page, had enough information to guide the user on how to use the website. There are some parts of the website that still need improvement, however this project has definitely pushed the team to build better products. These new found skills will set us apart in the industry.
3. Learning curves: Another major issue that we faced as a team, was the lack of technical knowledge, and the major learning curves that we had to go through to implement the entire project. Only four out of seven members were technically sound, and they did most of the development work. Two other team members were completely focused on the documentation, because the other members were overwhelmed with development tasks. To solve this, the whole team decided at the start of every milestone, what tasks they would like to work on, and how much time they would require to implement these tasks. One member of the



team was always on call, to help the other team members if required. While this may not have been the best way to complete tasks, it did help us to complete the project.

We as a team think that, before building any large scale product, we should do a thorough research of what we are actually trying to implement, who the competitors are, what would the target audience be and how would this product serve our users. From there on we decide what features we need to implement, and alongside each feature we calculate how much time would be required to build this feature, and what technologies would be best suited to build this feature. Finally, we evaluate the skill set, and work schedule of each member and decide who is the best person to complete this task. Once we gather a list of all the requirements, we decide which ones can be implemented in the given time constraint.

As a team lead, I feel I could have done a better job in setting deadlines, and delegating tasks according to each member's skill set. With that being said, I am very grateful to be the leader of this team, as all of my team members have put in their best efforts to see this project succeed, supported me all along the way, and in the process taught me a lot about myself. I have definitely learnt a lot as team lead, and I am happy our team could complete this project successfully.