

Tokyo Olympics Schedule viewer & Metrics Manager - (Olympicstop.com)

Contents

- 1. Describe what data is stored in the database. (Where is the data from, and what attributes and information would be stored?)*
- 2. What are the basic functions of your web application? (What can users of this website do? Which simple and complex features are there?)*
- 4. Project Title*
- 5. Project Summary: It should be a 1-2 paragraph description of what your project is.*
- 6. Description of an application of your choice. State as clearly as possible what you want to do. What problem do you want to solve, etc.?*
- 7. Usefulness. Explain as clearly as possible why your chosen application is useful. Make sure to answer the following questions: Are there any similar websites/applications out there? If so, what are they, and how is yours different?*
- 8. Realness. Describe what your data is and where you will get it.*
- 9. Description of the functionality that your website offers. This is where you talk about what the website delivers. Talk about how a user would interact with the application (i.e., things that one could create, delete, update, or search for). Read the requirements for stages 4 and 5 to see what other functionalities you want to provide to the users. You should include:*

Functionality

Data

Functions

9.1 A low-fidelity UI mockup: What do you imagine your final application's interface might look like? A PowerPoint slide or a pencil sketch on a piece of paper works!

9.2 Project work distribution: Who would be responsible for each of the tasks or subtasks? List of the person responsible for which exact functionalities in section 6. Explain how backend systems will be distributed across members. Be as specific as possible as this could be part of the final peer evaluation metrics.

Ajit Ravichandran (ajitr2)

Kavya Karthi (kkarth3)

Sneha Mohan (sneham5)

Sam Kremerman (skreme3)

1. Describe what data is stored in the database. (Where is the data from, and what attributes and information would be stored?)

- The data is from the Tokyo Olympics dataset provided in the PT1 Datasets for extra credits: <https://www.kaggle.com/datasets/arjunprasadsarkhel/2021-olympics-in-tokyo>.
- The provided dataset contains the following attributes present in the following files (the entities list crosses 11k rows)
 - ==> *Athletes.csv* <==
Name, NOC, Discipline
 - ==> *Coaches.csv* <==
Name, NOC, Discipline, Event
 - ==> *EntriesGender.csv* <==
Discipline, Female, Male, Total
 - ==> *Medals.csv* <==
Rank, Team/NOC, Gold, Silver, Bronze, Total, Rank by Total
 - ==> *Teams.csv* <==
Name, Discipline, NOC, Event
- The provided dataset contained 22 overlapping columns in it, and it only provides a little leeway as we do not have other information that would allow us to branch away from Stats oriented application.
- We merged another dataset to get more information on Scheduling, etc., to add more functionality. So we are looking at <https://www.kaggle.com/datasets/llui85/tokyo-2021-olympics-complete-grouped-by-type>, which offers additional columns we can leverage.
 - "Aggregate,"
 - "Ceremony,"
 - "Competitor,"
 - "Discipline,"
 - "Event,"
 - "EventUnit,"
 - "Individual,"
 - "Medal,"
 - "MedalCount,"
 - "Organisation,"
 - "Participant,"
 - "Phase,"
 - "Result,"
 - "ScheduleItem,"
 - "ScheduleSession,"
 - "Stage,"
 - "SubEventUnit,"

"Venue"

- Data will be cleaned from this dataset and merged with other relevant tables during DB design.

2. What are the basic functions of your web application? (What can users of this website do? Which simple and complex features are there?)

Basic Functionality

- The website will have a login workflow.
- It will allow us to view the events' results by allowing us to filter various categories. (Athletes, sports & countries)
- CRUD operations on entities
- Search view to find results, venues
- Will display stats by country, player, sport
- Detailed view to express more data
- Option to add events, view results, update results - delete results (Role/Privilege based)

Complex Features

- Planning to add analytics data from the backend
- Based on feasibility & time constraints, the potential to book event spaces (Role/Privilege based)

3. What would be an excellent creative component (function) that can improve the functionality of your application? (What is something cool that you want to include? How are you planning to achieve it?)

- Multiple fields to custom search sort specific entities (Backend API)
- UI Visualization (Python Plotly)

4. Project Title

- Tokyo Olympics Schedule viewer & Metrics Manager (Olympicstop.com)

5. Project Summary: It should be a 1-2 paragraph description of what your project is.

Any sport would require a website in this modern age where the fans or people can go and look up details about their favorite athletes or venues in general. We propose a solution that targets this use case and provides users with a tool to access the same information by allowing multiple search/filter features that will enable folks to quickly gather the information they want to review about their favorite sportsperson. In addition, we will let users see global-level stats and individual contributions (athlete, country, event, and more). This solution is proposed as a one-stop solution for folks who would like to gather more information quickly.

We plan to design a web application with both a front-back end to offer this functionality. We will use various components to develop this application which will be listed below.

6. Description of an application of your choice. State as clearly as possible what you want to do. What problem do you want to solve, etc.?

The Olympics is the pinnacle of sports, with all the countries coming together with their best of the best representatives competing. Still, more focus is needed on this event compared to other popular sporting events. One key factor in this modern digital age is the presence of a good website as having a good website. So we thought about an excellent way to engage people with the available data. And a perfect way to engage people is to show the current results competitively. Everyone would love it if they or their country won any competition regardless of their personal engagement with the sport. On the other hand, we would also want to quickly find details about our country, sport, and athlete in general.

The proposal aims to solve a fundamental problem by allowing users to engage more in the events even if they're not physically present in and around the stadium or do not have a subscription to watch the event live. We are a free-to-use website that lets users quickly glance or deeply review the stats they want to check by allowing them to filter on various parameters that suit their needs and have a summarised or detailed view.

We provide a clean interface that lets users quickly get what they want. Let's say I am User A - "I want to see who won in Archery as I used to do archery prior, and I want to see someone excel in that field." All I have to do is click on the sport and filter it - I will get a list of winners in the sport of archery, along with players who interact with the sport. Let's say another User B - "I want to see my country win everything" - He can search for his country, and all the players and their win tally would pop up, which would want to motivate their country more.

Also, in an application, we will have certain moderator users who have privileges to add more information or rectify or delete certain information regarding the events to keep the updates as accurate as possible so everyone can review the results without bias.

We let users quickly view all this information in a clean seamless way, no matter what kind of fan you're. We gotcha!

7. Usefulness. Explain as clearly as possible why your chosen application is useful. Make sure to answer the following questions: Are there any similar websites/applications out there? If so, what are they, and how is yours different?

Who would want to use something other than an application that provides information about an event that is the actual pinnacle of sport? Of course, everyone would like to win at the end of the day. No one would enjoy losing, and even if we cannot win, we would want someone who represents us to win, and we provide a platform to view all this information quickly. We offer an easy-to-view platform to review the various data, statistics, or significant achievements performed by athletes individually or in the scope of a country or the sport.

The website most similar to the use case is the website from which the original dataset is scraped, which is <https://olympics.com/en/olympic-games/tokyo-2020>. The website focuses more on multiple things, which prevents users from getting what they want quickly. We focus on

providing a user view that lets people view high-level details of various categories sorted by multiple parameters. We allow these views, allowing users to get whatever information they require quickly. We provide high-level visualizations that provide aggregated views with stats that provide visualizations that let users view multiple fields.

8. Realness. Describe what your data is and where you will get it.

- The data is from the Tokyo Olympics dataset provided in the PT1 Datasets for extra credits: <https://www.kaggle.com/datasets/arjunprasadsarkhel/2021-olympics-in-tokyo>.
- The provided dataset contains the following attributes present in the following files (the entities list crosses 11k ~ 11086 rows)
 - ==> *Athletes.csv* <==
Name, NOC, Discipline
 - ==> *Coaches.csv* <==
Name, NOC, Discipline, Event
 - ==> *EntriesGender.csv* <==
Discipline, Female, Male, Total
 - ==> *Medals.csv* <==
Rank, Team/NOC, Gold, Silver, Bronze, Total, Rank by Total
 - ==> *Teams.csv* <==
Name, Discipline, NOC, Event
- The provided dataset contained 22 overlapping columns in it, and it only provides a little leeway as we do not have other information that would allow us to branch away from Stats oriented application.
- We merged another dataset to get more information on Scheduling, etc., to add more functionality. So we are looking at <https://www.kaggle.com/datasets/llui85/tokyo-2021-olympics-complete-grouped-by-type>, which offers additional columns we can leverage.
 - "Aggregate,"
 - "Ceremony,"
 - "Competitor,"
 - "Discipline,"
 - "Event,"
 - "EventUnit,"
 - "Individual,"
 - "Medal,"
 - "MedalCount,"
 - "Organisation,"
 - "Participant,"
 - "Phase,"
 - "Result,"

"ScheduleItem,"
"ScheduleSession,"
"Stage,"
"SubEventUnit,"
"Venue"

- Data will be cleaned from this dataset and merged with other relevant tables during DB design.

9. Description of the functionality that your website offers. This is where you talk about what the website delivers. Talk about how a user would interact with the application (i.e., things that one could create, delete, update, or search for). Read the requirements for stages 4 and 5 to see what other functionalities you want to provide to the users. You should include:

Functionality

Data

We have different kinds of data, workflows, and access controls

- ❖ We provide the initial set of data, and this is the total stats of the Olympic games provided by us from the dataset after cleaning and manipulation to fit the use case
- ❖ User data for authentication, authorization, and access control. It would include attributes such as username and password
- ❖ We design the schema of workflows that end-users can leverage.

Functions

Any user has the following abilities that they can leverage from our website.

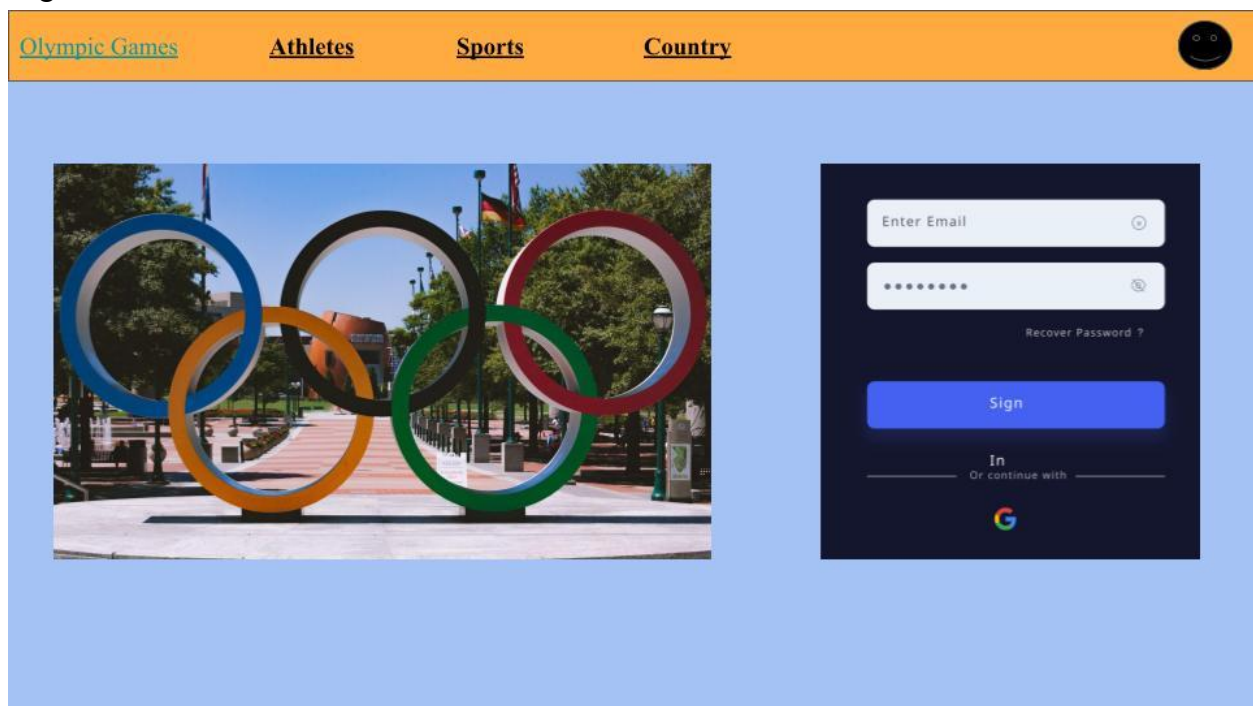
- ❖ One can log in to our website, and we process authentication and authorization.
- ❖ Anyone can leverage the search sort functionality of our website.
- ❖ Anyone can leverage the visualizations on the website.
- ❖ Anyone can request certain events/information to be added, updated, or removed.
- ❖ The moderator can add events, player information, countries, etc.
- ❖ Data enforcement for certain fields such as countries, for example, so random non-existent countries can't be added.
- ❖ Moderator can update data along with data enforcement applied.
- ❖ Moderator can delete data as needed.
- ❖ Moderators can view data to validate insertion, updation, and deletion
- ❖ Have Stored Procedures and triggers to update stats across multiple tables when an athlete wins a medal across multiple dependant tables (country stats) or update individual sum of medals across categories.

9.1 A low-fidelity UI mockup: What do you imagine your final application's interface might look like? A PowerPoint slide or a pencil sketch on a piece of paper works!


Slides are Linked below, and additionally, images are posted below.


Here: [Olympicstop.com](https://olympicstop.com)


Login Screen




Landing Page

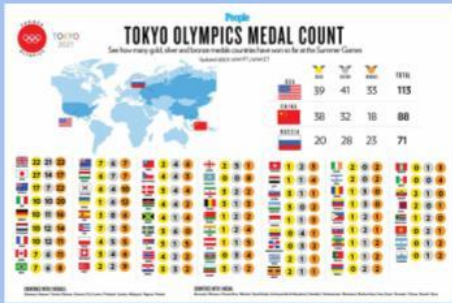

[Olympic Games](#)



[Athletes](#)


[Sports](#)


[Country](#)

- You can click on Athletes to search sort filter by athlete name, country, sport
- You can view various sports and current stats associated with the sport
- You can click on a country to view more information of its current win tally etc
- Or you can view an individual player details or see high level stats here below

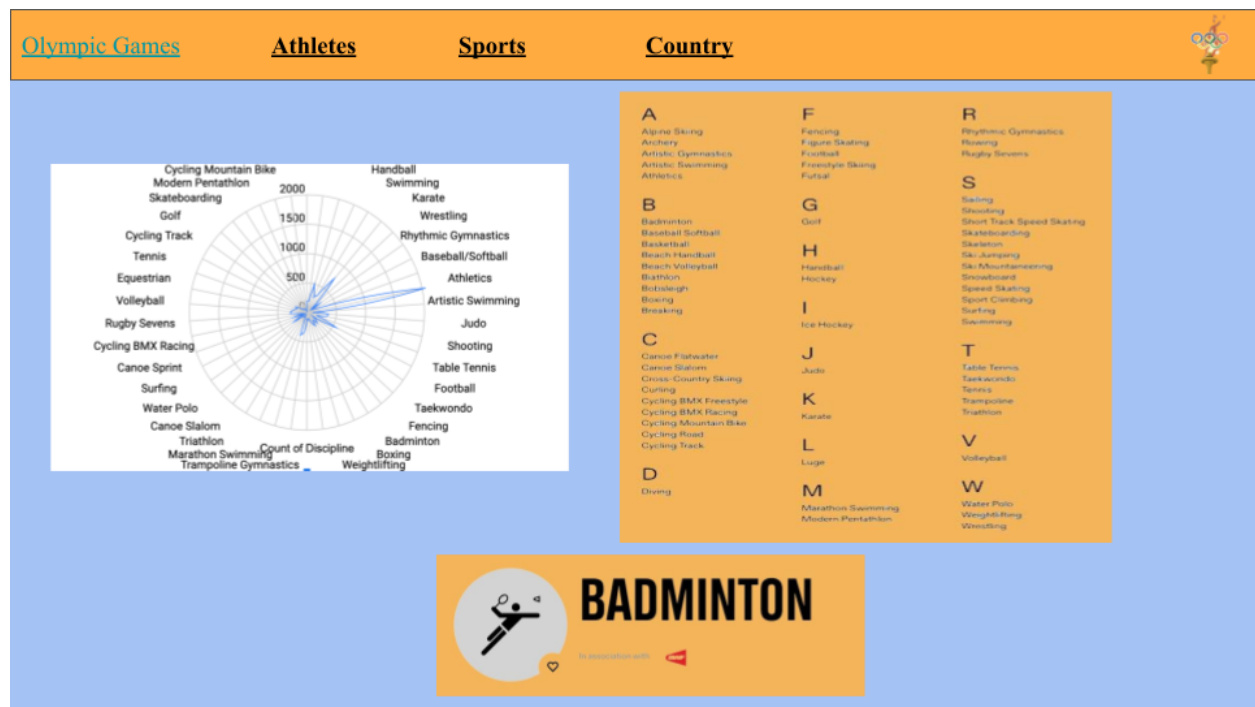




View by Athlete

[illegible]

View by Sport



View by Country



Admin user who can perform CRUD operations

9.2 Project work distribution: Who would be responsible for each of the tasks or subtasks?

List of the person responsible for which exact functionalities in section 6. Explain how backend systems will be distributed across members. Be as specific as possible as this could be part of the final peer evaluation metrics.

The idea is to design the front end on React (either JSX/TSX), Have visualizations on the data performed using Plotly.

We are leaning towards Python's Flask for the backend; for the database, it would be a MySQL implementation on GCP.

Essentially the team will collaborate with each other as the components are tied together. But as for individual contributions, we have decided the following:

Ajit Ravichandran (ajitr2)

Will work primarily on the DB design, implementation, deployment hosting, and servicing of the databases as needed during the monitoring or development phase. (Local hosting of MySQL till Cloud Hosting of the same server and DB design)

Will work on data visualization components that can be passed to the front end

Will additionally assist teammates in the process of UI design / API framework development.

Kavya Karthi (kkarth3)

Will work on the list view and sort view for all the pages on the front end (Player Page, Country Page, Sports Page)

Will leverage the APIs built to create a seamless UI for the users to leverage and quickly navigate through the website. Will be leveraging React (either JSX/TSX)

Sneha Mohan (sneham5)

Will work on the Detailed view and Filtered view for all the pages on the front end (Player Page, Country Page, Sports Page)

Will leverage the APIs built to create a seamless UI for the users to leverage and quickly navigate through the website. Will be leveraging React (either JSX/TSX).

Both the folks working on the front end (Kavya & Sneha will leverage the visualizations provided to the front end and will enclose the same and display these metrics on screen)

Sam Kremerman (skreme3)

Will work primarily on the backend leveraging backend tools and technologies to design interfaces through which data can be manipulated on the backend. Will work on designing APIs for the front end.