Product Backlog (refined for sprint 2)

Task	Estimation	TODO	WIP	Completed	ID
As a user, I want to be able to select a match and see different representations of the match history. Estimation: 40 Priority: 2	4 hours		JS - use inputs to change between graphs and adapt data accordingly	HTML and CSS	1
As a user I want to be able to see key events in the match. Estimation: 20 (may remove) Priority: 3	2 hours		JS - use API and react to changes in the match as it progresses	HTML CSS	2
As a user, I want to be able to view the location and time of the match in progress. Estimation: 3 Priority: 2	20 minutes		JS - use data from API and backend	HTML CSS	3

As a user, I want to be able to change the statistic being measured (ie goals, red cards), so that I can find the player stats I want. Estimation: 3 Priority: 2	20 minutes		JS - use inputs to change/adapt on-screen representation.	HTML CSS	4
As a user I want to be able to change the season of the stats. Estimation: 8 Priority: 3	50 minutes	JS - use inputs to change/adapt on-screen representation.		HTML CSS	5
As a user I want to be able to select a stat mode (total or average or running-monthly-average) - average is dictated by a user input on a slider, specifying the games that are being averaged. Estimation: 13 Priority: 2	1 hour and 20 minutes		JS- use the API and JS to change the graphs as the user selects a different team	HTML CSS	6

As a user I want to be able to select a player Estimation: 1 Priority: 2	20 minutes		JS- using JS to update the graph with player stats accordingly.	HTML CSS	7
As a user I want a table that has the top 10 players in the league for a selected stat Estimation: 1 Priority: 3	20 minutes		JS - using JS to format and present data	HTML CSS	8
As a user, I want to input a period of time in the future to extrapolate relevant data. Estimation: 40 Priority: 4	4 hours	JS - use JS to create regression analysis and predict data in the period specified.		HTML CSS	9
As a user, I want to predict who will score the most, who will get the most assists. Estimation: 40 Priority: 3	4 hours	JS - use JS to create regression analysis and predict data in the period specified.		HTML CSS	10
As a user, I want to	2 hours	JS - use JS to create		HTML	11

predict the results of next year. Estimation: 20 Priority: 4		regression analysis and predict data in the period specified.		CSS	
As a user I want to be able to change the chart type so that I get the right representation. Estimation: 13 Priority: 2	1 hour and 20 minutes	JS - Change the chart variable inside the chart object. If pie charts are also included as an option some data manipulation will be required		HTML CSS	12
As a user I want to be able to select a team so that I can view their match history. Estimation: 13 Priority: 3	2 hours		JS - using JS to update this seasons match history and being able to use data from an API to display the team’s entire match history across seasons.	HTML CSS	13
As a user, I want to be able to view the entire league table so that I know the standings Estimation: 5 Priority: 4	30 minutes		JS- Using JS to get the data from the API’s and be able to present them in a table.	HTML CSS	14

As a user, I want to be able to see my favourite team's upcoming matches so that I am updated. Estimation: 8 Priority: 5	1 hour and 20 minutes	JS - use localStorage to remember specified favourite team, and present data dynamically depending on API calls.			15
Learning how to use the chosen APIs Estimation: 5 Priority: 1	30 minutes			Study how to query data and what data is given back JS - spike code to test calls and provide information.	16
HTML navigation Estimation: 5 Priority: 1	30 minutes		JS - transferring data persistently after navigation (ties in with id 20)	When button is pressed navigate to the correct page	17
Get localStorage working for static data (topScorers) Estimation: 5 Priority: 1	50 minutes			Use JS to initially retrieve static data for the user and put into localStorage.	18
Deployment of Website Estimation: 5	30 minutes	Learn how to host a website			19

Priority: 3					
Learn how to pass variables between pages Estimation: 3 Priority: 1 NEW	30 minutes		Create onclick event with navbar items and use localStorage to keep track of the currently selected league.		20

Sprint 1 Planning (1/09/21 -> 17/09/21)

What we are doing this sprint

Completing skeleton of the application - Sprint priority 1

- Finishing navigation through HTML pages - Sprint priority 1
- Completing CSS (design) - Sprint priority 2

Learning how to pass variables through pages - Sprint priority 3

What each person is doing

SCRUM MASTER: Antony

Antony:

Tasks 4-12 WIP: CSS, this is the CSS for the stats page, note that most of the CSS through the application is shared, therefore, this task has a lot of overlap with any other task including CSS.

Acceptance criteria/definition of done: The product owner and team are happy with the appearance of the application, everything in the application follows a consistent colour scheme, the application can be tested with some dummy data, and the application uses consistent UI elements

Suryadeep:

Tasks 1,2 and 3: Get the API functionality into the skeletal framework of Live Fixtures, and make sure requests work consistently as desired.

Acceptance Criteria/definition of done: The product owner and the team are happy with the API functionality in the skeletal framework of the application, data can be updated and follows through different web pages.

Max:

Tasks 16 and 18: Create Spike code for the API functionality, and make sure that requests work consistently as desired. Also, figure out how to store this data across the web application (IE. setting up database / local-storage / class structures) and create a working prototype.

Acceptance criteria/DoD: pull requests work, and data can be transferred across different pages of the app reliably.

Mursal:

(Task 17) Finalise navigation bar, get it on every page of the app(JS) - ties in with task 18, help out with CSS in other elements.

Acceptance criteria/DoD: Navbar matches colour scheme, works on every page and the links work correctly.

Eric:

Tasks 13,14 and help with 17(navigation): This is the CSS for the League Table page and match history page and will also overlap with some widgets too.

Acceptance criteria/DoD: Consistent colour scheme, Links work and the team is satisfied with the appearance of each page.

Also, allow space to add APIs later on.

Goal of sprint

The goal of this sprint is to complete the skeleton of the website. The skeleton of the website means all the design and navigation is finished, in the demonstration, the team should be able to show what the website will look like to users, this does not include data from the API, which shall be completed in the next sprint.

Sprint 1 Retrospective - 8/09/21

What went well?

1. We met the target velocity, all sprint backlog items were completed to the definition of done (passed specific AC).
2. In the next sprint, a slightly larger workload may be given, considering that the sprint was completed within the period.
3. No assessed risks have occurred yet, but the monitoring strategies have been in place. For example, meetings and scrum master outreach have ensured that scheduling issues are addressed before they become a problem.
4. We conducted a few live sharing coding sessions where a group of team members worked together on the same code. This worked well and increased productivity on tasks and alleviated many issues that would be faced if only one member was doing it. Implementing a form of pair programming into our processes could help increase velocity.

What could have gone better?

1. Could have set more tasks for each member this sprint, as we finished within the sprint period. However, estimating the tasks/user stories will always be imperfect, so will not add too much extra perceived workload for the next sprint.

2. We could have done more collaborative work to speed up task completion. A lot of tasks had overlap and could have benefited from team members working together to solve common problems e.g. both the stats page and live match page utilised charts.js, therefore, it would have been ideal if both team members researched and discussed the best way to implement the library.

3. We could have polished the league page more, instead of having buttons to transition to a page of widgets we could have a smaller widget that when clicked takes the user to the correct page, for example, there could be a small graph that when pressed takes the user to the statistics page which, includes a larger graph, league table and more customisable inputs. The page could also have a title

corresponding to the league that was selected e.g. when the user selects the premier league the title of the next page should be ‘Premier League’

What will we try next?

1. More live share sessions as groups, working together. Incorporating elements of pair programming into our team model. Allocating certain PBIs to be done in pairs potentially.
2. Learning session about how to use API’s and keep the data consistent throughout the pages of the App.

3. We will begin to implement the backend more substantially. Figure out which data is static (put into localStorage) and which data is dynamic (retrieve at run time from the API). Get data accurately retrieving and attempting to integrate into the front-end.

4. We will also try to increase the velocity of the sprint for the next one, which will include a heavier task workload for every member of the team. This again is always going to be an imperfect practice as individual user story estimations will not always be exact and could be overestimated or underestimated. As the project progresses, estimating how long tasks will take will hopefully become a bit easier as the team gains experience.

5. Added/changed a few user stories to account for the potential difficulties in implementing a working backend across all pages.

What questions do we have?

1. The team has a clear understanding of what needs to be done next and the tasks/user stories have been allocated accordingly, therefore, after this sprint, there are no questions.