Process Book - FIFA20 Dashboard

1. Overview and Motivation

While searching for data on sports, we came across Soccer player ratings (FIFA20) on Kaggle. Though FIFA 20 is a soccer simulation video game developed by EA, ratings within the game are considered as the best, even among professional soccer players. These ratings showcase the players in the game based on their performances from the past year of world soccer. Since we all are passionate soccer fans, we immediately decided to work with this data.

With this data, we would like to visualize player information from different perspectives.

1. **Player Rooster** This would give the list of all players playing for a given country, club, position and their individual ratings.
2. **Player Comparison** This would let us compare the ratings of any two chosen players.
3. **Story View** This would highlight parts of our visualization and link them to established facts and predict outcomes

The objective is to identify the most effective visualization for each of these perspectives.

2. Links

2.1 Source Code

All the code for this project can be found on GitHub.

2.2 Website

The website can be visited at https://laknaren16.github.io/fifa20-dashboard/index.html

3. Team

- Lakshmi Narayanan Ramasamy
  UID: u1201500
  Email: lakshminarayanan.ramasamy@utah.edu
- Rohit Singh
  UID: u1210167
  Email: singh.rohit@utah.edu
- Sachin Boban
  UID: u1210407
  Email: sachin.boban@utah.edu

4. Development Build

To generate a development build we make use of Node.JS Package Manager or NPM. NPM shipped with Node.js. NPM is one kind of middle main which connects projects with dependencies. NPM is written in Javascript. More information on NPM can be found here. Our application is composed of a combination of coding frameworks and languages including JavaScript, HTML, CSS, ReactJS and D3JS.

4.1 Installation

Clone the repo and install dependencies using the following commands:

```
$ cd datavis-fifa20-dashboard/
$ npm install
```

4.2 Usage

1. Start the node server using the following command:

   ```
   $ npm start
   ```
2. Navigate to localhost:3000/ on your browser to view the app.

5. Data

All the player ratings as well as league information are stored as `csv` files in Github (`src/data/`).

5.1 Player Ratings

FIFA 20 Player Ratings data was obtained from Kaggle. This data contains ratings for around 18K professional soccer players. Each player has around 70 attributes. This data would require some cleanup as we will not be needing all the attributes. Missing values in the dataset need to be marked with some flags (e.g. NA/UNK).

5.2 League Information

In order to provide league-wise visualization of the data, we need a mapping between club name and the league which it is a part of. However, only club names are available in Kaggle dataset. Further details about the club like ratings and league information was obtained by scraping FIFA Index website.

This scraping was done using python script. The script can be found here.

6. Project Proposal- (25th Oct 2019)

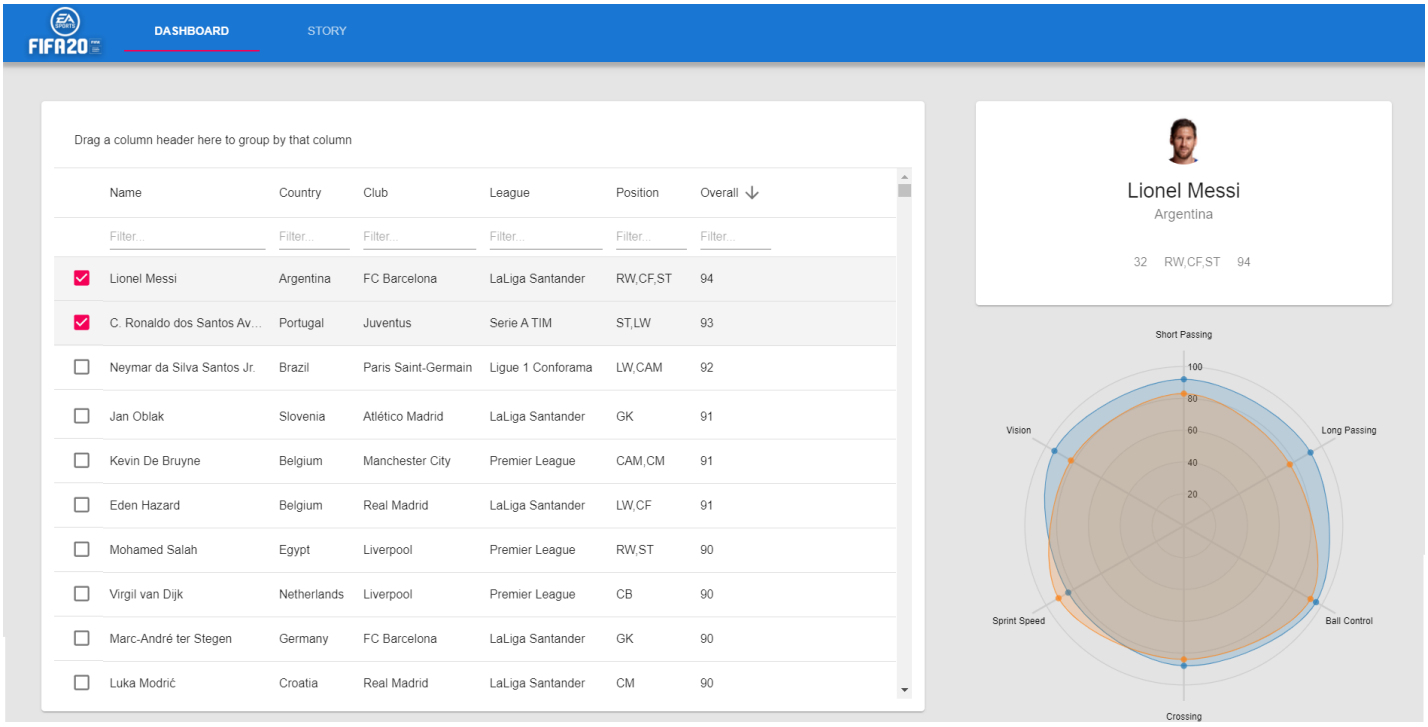
The document describing our initial project proposal submitted on 25th October 2019 can be found here.

7. Project Milestone- (8th Nov 2019)

The project process book as on the milestone submitted on 8th Nov 2019 can be found here. The project milestone release is available here.

8. Project Milestone Meeting- (22nd Nov 2019)

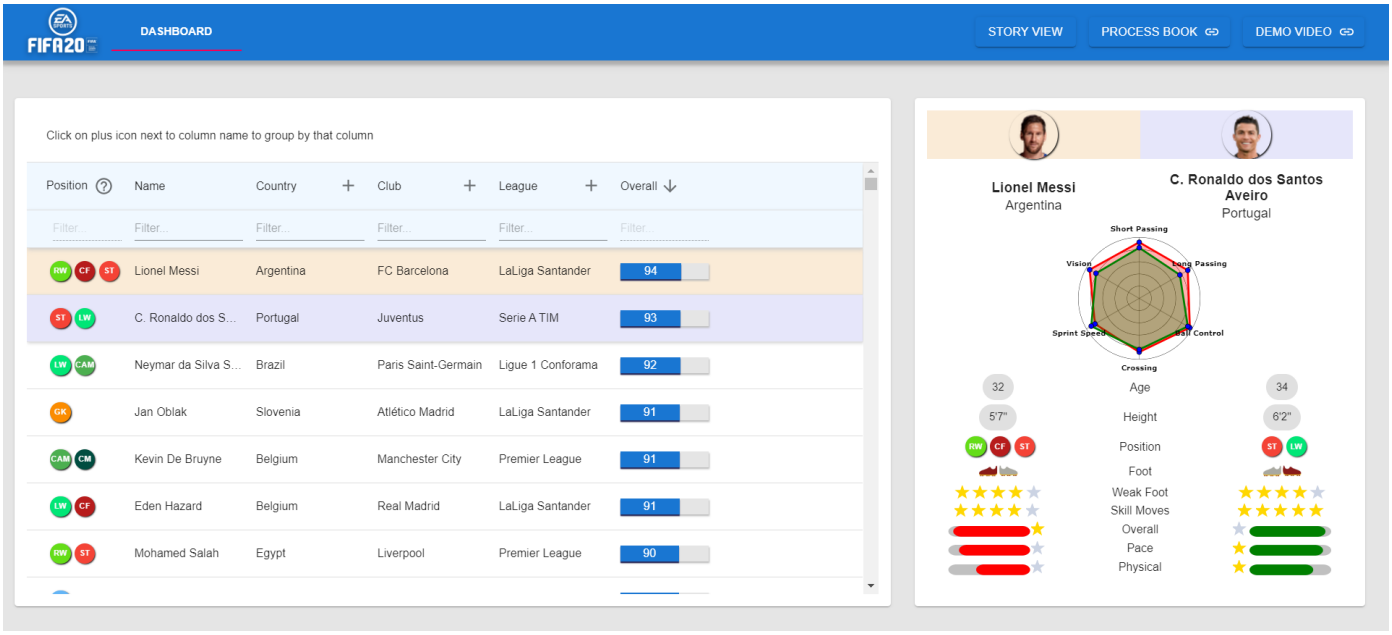
Here is a view of our application as presented to Kiran during the meeting:



With a clear functional prototype in place, we were able to have a through discussion on each aspect we were trying to represent. We received several suggestions from Kiran which helped us focus our efforts on better weighed designs and well specified implementation goals in the last and most productive phase of the development.

9. Final Visualization Design

Our final design deviates significantly from our initial design proposal in order to make the visualization experience more intuitive, user friendly and better suited to its goal. We have improvised based on the experience we gained on the way, the peer feedback we received from Braeden and the continuous guidance from our TA- Kiran Gadhave. Here is a peek into our final design:



10. Design Motivations

The following section give a brief summary of the various considerations and motivations behind the ideas we discarded and the final design implementations we adopted for each of view:

10.1 Player Table

The Player Table lists out all players in our data with options for filtering, sorting and grouping all set into one concise window. If you investigate our initial proposal here (which we now realize wasn't as refined as we would have liked it to be) you will find that we had planned our first view or tab to be driven as per the following flow:

Select Country -> Select League -> Select Team -> Get Team Player Table -> Select Player

And then we had planned to have a second tab to allow direct player selection for comparison.

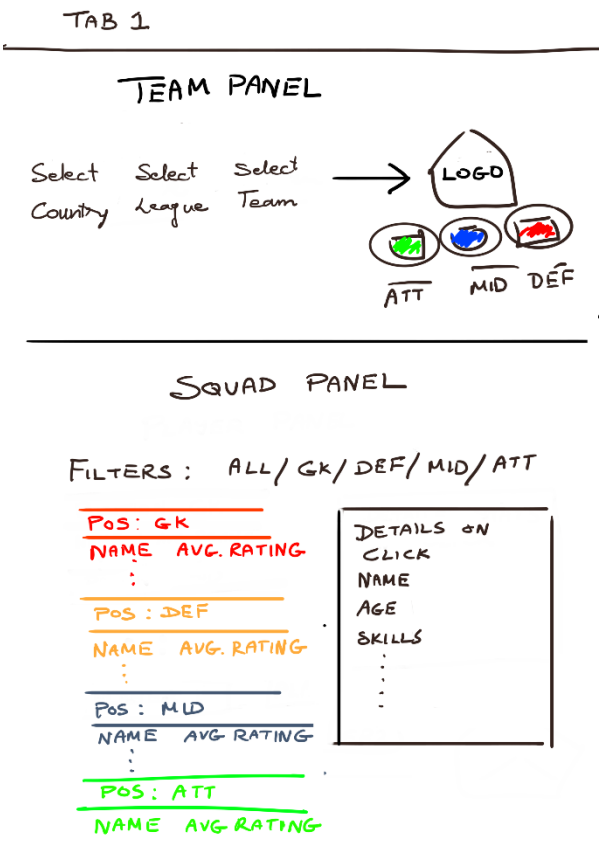


Figure 1: Initial Proposal for TAB1

The figure on the right depicts our original plan for team and player visualization as our first view. However, we realized later that this approach is way too much user selection driven than is required. So, we have adopted a more concise approach. This shift in design evolved from a feedback we received from Braeden during the peer feedback. Braeden said we should add filters for direct search in addition to allowing selection by team. This got us thinking on whether our selection flow is too restrictive, and would the user spend more time navigating between players than viewing the statistics. So, we came up with a design for the selection component that captures more and uses less resources. In our current implementation selection is driven through a table which presents a full list of players on load. The user can directly dive into viewing player statistic. The user can filter/search within any of the attributes in our list through the easily accessible selection fields on the table header.

This table with filter-based approach also does away with using a separate second tab to allow selection of players for comparison. We combined both views into one window. The table gives an overview and scope for selection and filtering. Selection then drives viewing of player details and comparison. Our design is in line with the:

Visual Information Seeking Mantra

(Shneiderman, 1996)

Overview first,

Zoom and Filter,

Then details on demand

The table uses the most appropriate visual cues to represent the data for each of its columns. Name, country, club and league are shown as text. A player's overall rating is encoded using a bar graph.

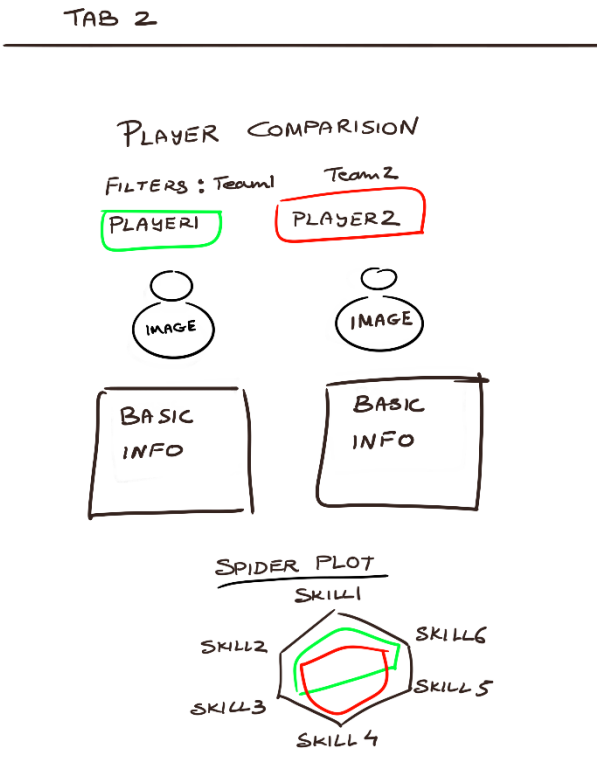
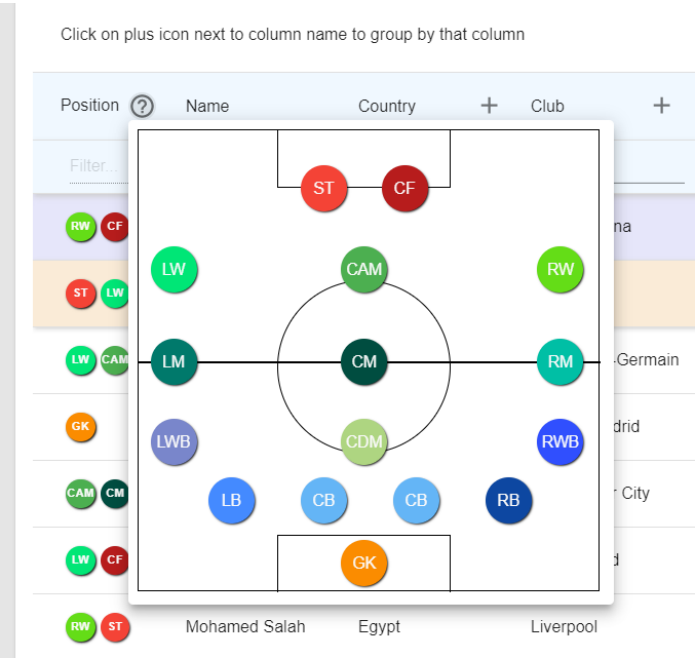
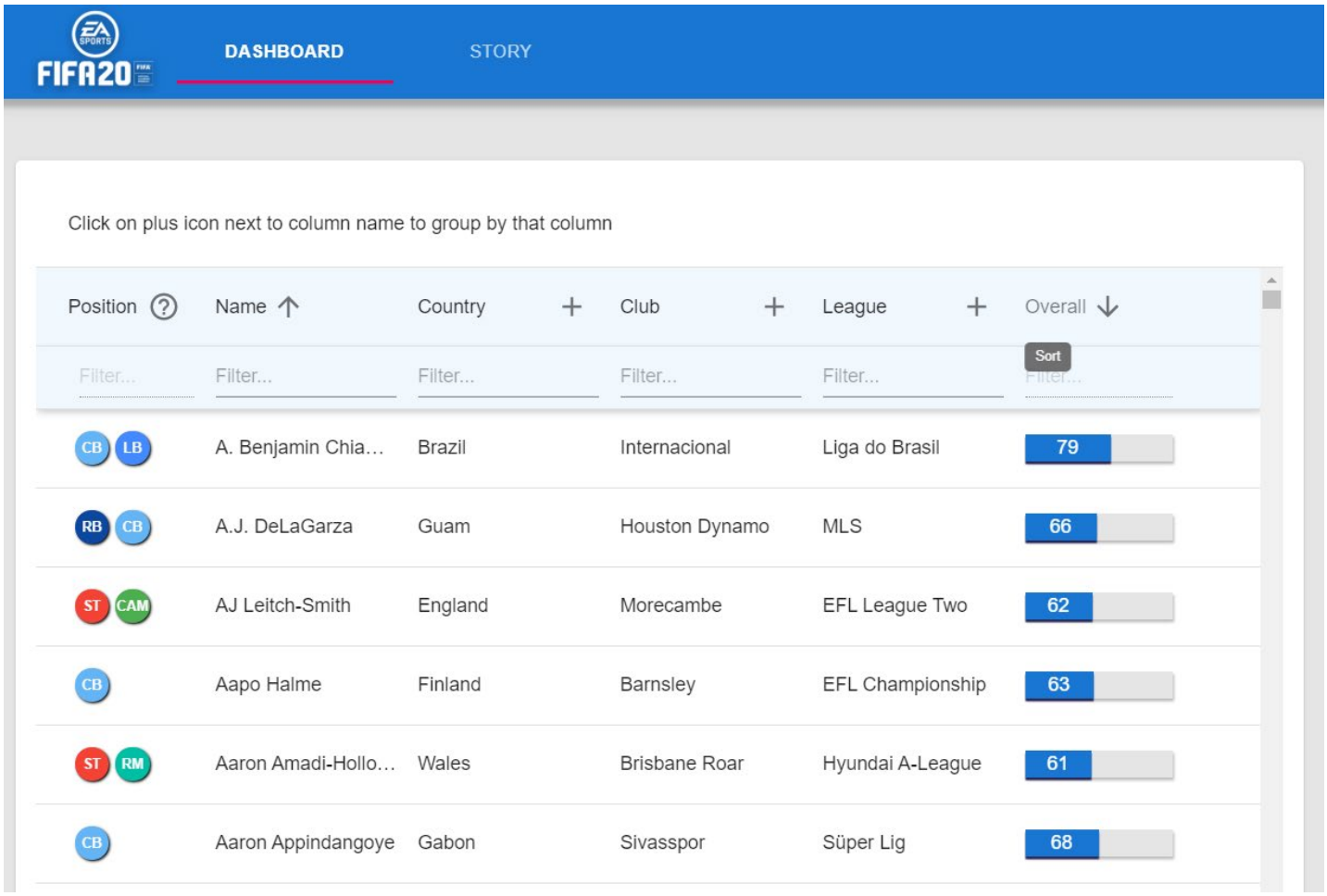


Figure 2: Initial Proposal for TAB2

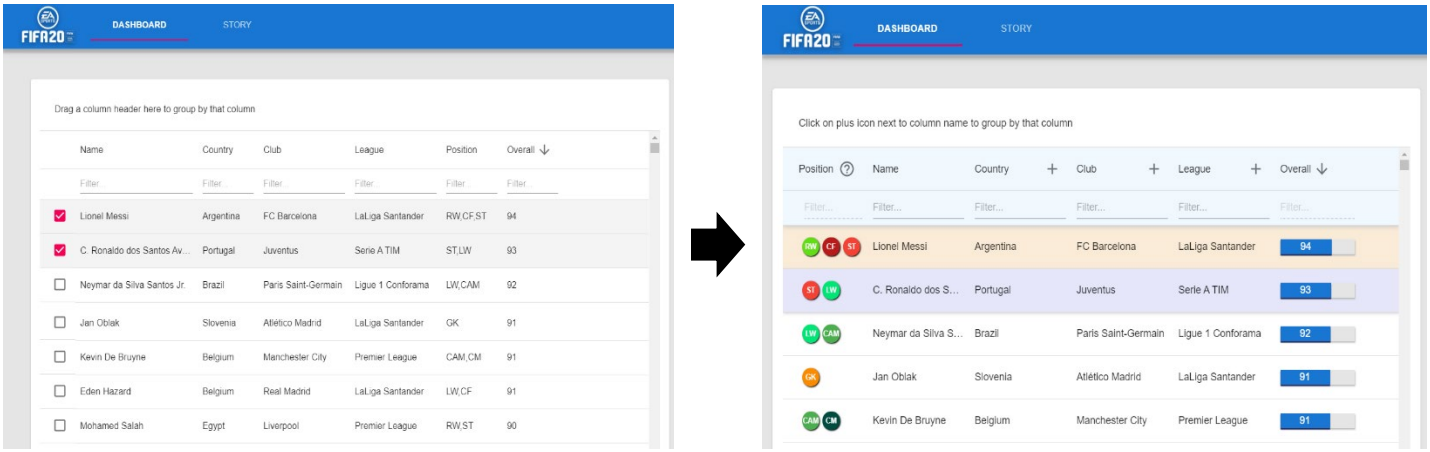
Position being a categorical data is rightly represented using color encodings. Color Intensity denoted subcategory for position. The position legend is available on hover:



Here is a closer look at the table. The fields with + symbol support grouping while fields with ↓ symbol support sorting. FILTER allows the user to directly query based on column value:



If you compare against the table as on the Project Milestone meeting the difference and enhancement in visual appeal is apparent.



10.2 Player Card

The Player Card shows details for player or players(two) selected on the player table. The player card also houses a radar plot for comparison of two players but let's talk of it as a separate component altogether. The key challenge to designing the player card was to settle upon the following questions about the layout:

1. What attributes to show for a player in single player mode?
2. What attributes to show for players in double player or comparison mode?
3. Visual encoding for the selected attributes?
4. How to show link between selection on table and corresponding update on player card?

10.2.1 Single Player Card

This final state of this view is a result of several cycles of planning and work and re-planning and rework. Here is its journey:

Lionel Messi

Argentina

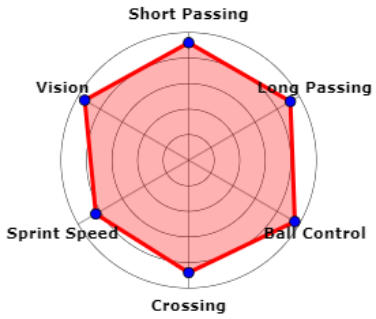
Age: 32
Height: 5'7"
RW,CF,ST

Weak Foot ★★★★☆
Skill Moves ★★★★☆

94	Overall
87	Pace
92	Shooting
92	Passing
96	Dribbling
39	Defending
66	Physical

Lionel Messi

Argentina



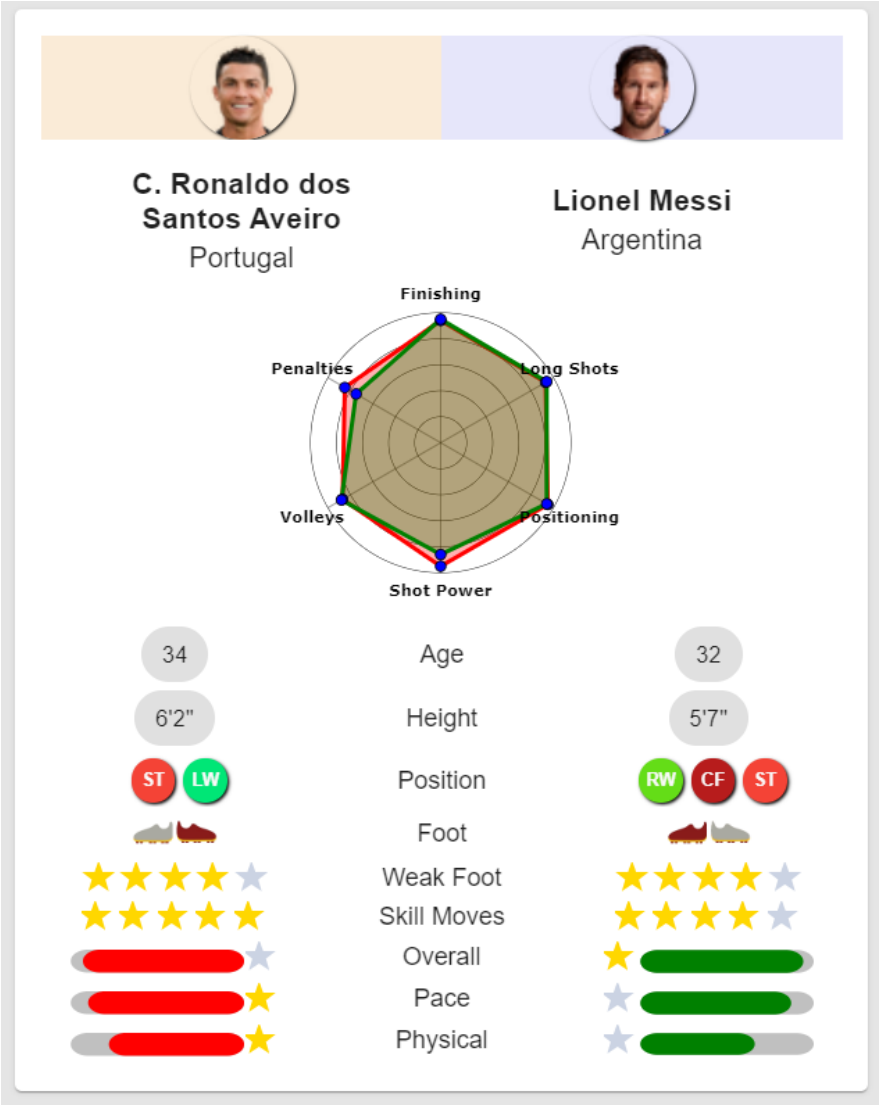
Age: 32
Height: 5'7"
RW CF ST

Weak Foot ★★★★☆
Skill Moves ★★★★☆

94	Overall
87	Pace
92	Shooting
92	Passing
96	Dribbling
39	Defending
66	Physical

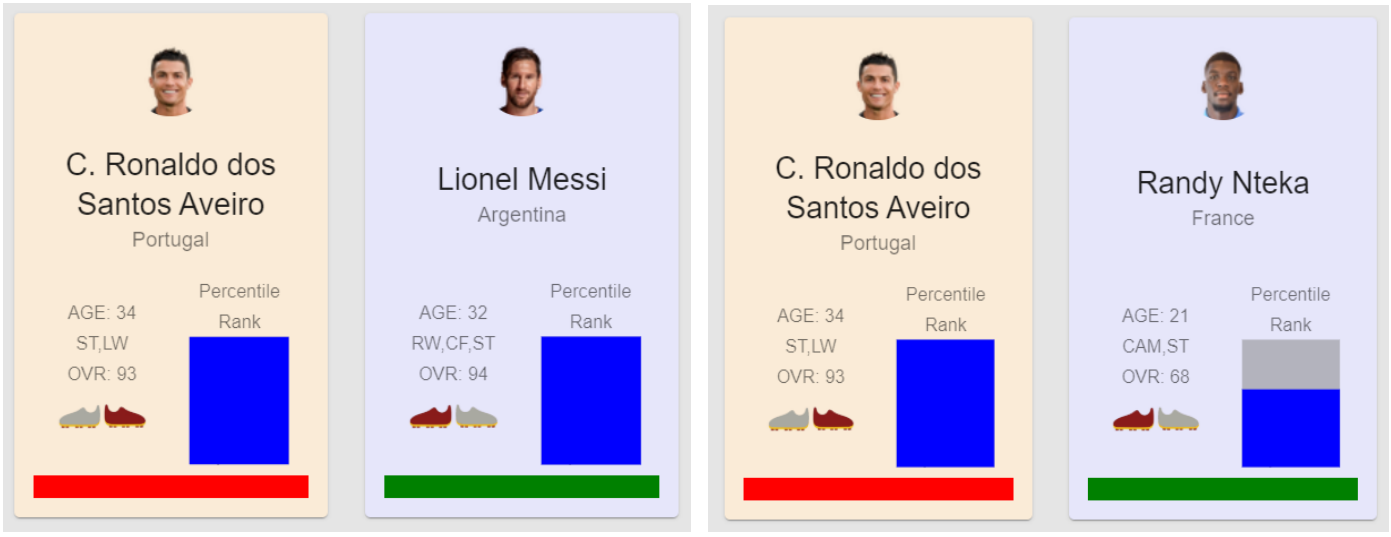
10.2.2 Player Comparison Card

This state borrows heavily from the single player card but has been designed to be different where it's needed. Here is how the idea evolved over multiple iterations:



10.2.2.1 The Percentile Rank (a promising but discarded idea)

In the initial designs we were considering including a rank on each player card based on percentile as in how the players Overall Attribute compares to all others in the list.



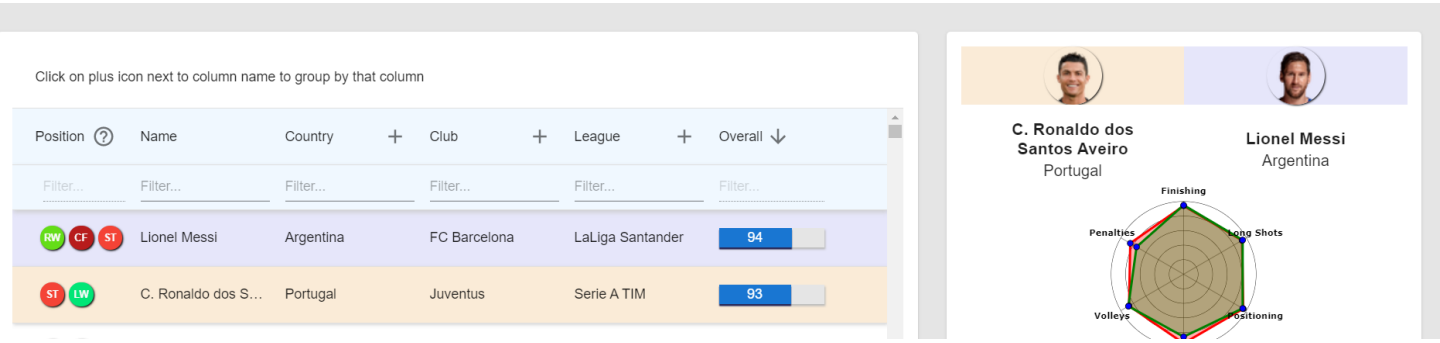
While we believed that this is a good representation of a player's overall skill, we did not go on to feature this in the final design because of space constraints on the card.

10.2.2.2 Common Labels (a space optimization technique)

As you may notice that in the final player comparison card designs, we have merged the cards into one and have used same attribute labels for the two players. This turned out to be a simple but effective technique to save space while giving the visualization a truly comparison.

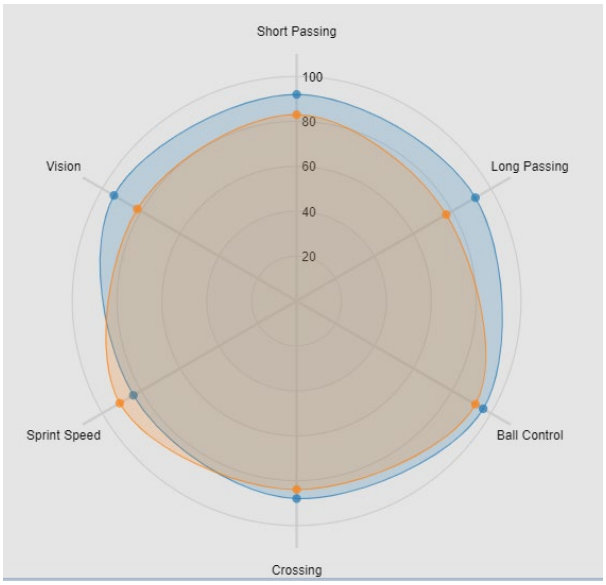
10.2.2.3 Color Encodings for Selection (a link to Player Table)

The effectiveness here is in simplicity of the choice. While white clearly looks like a better choice of color for the body, adding a strip of color to the head not only enhances the visual appeal but establishes a clear correspondence to the selections made on the table.



10.2.3 Radar Plot

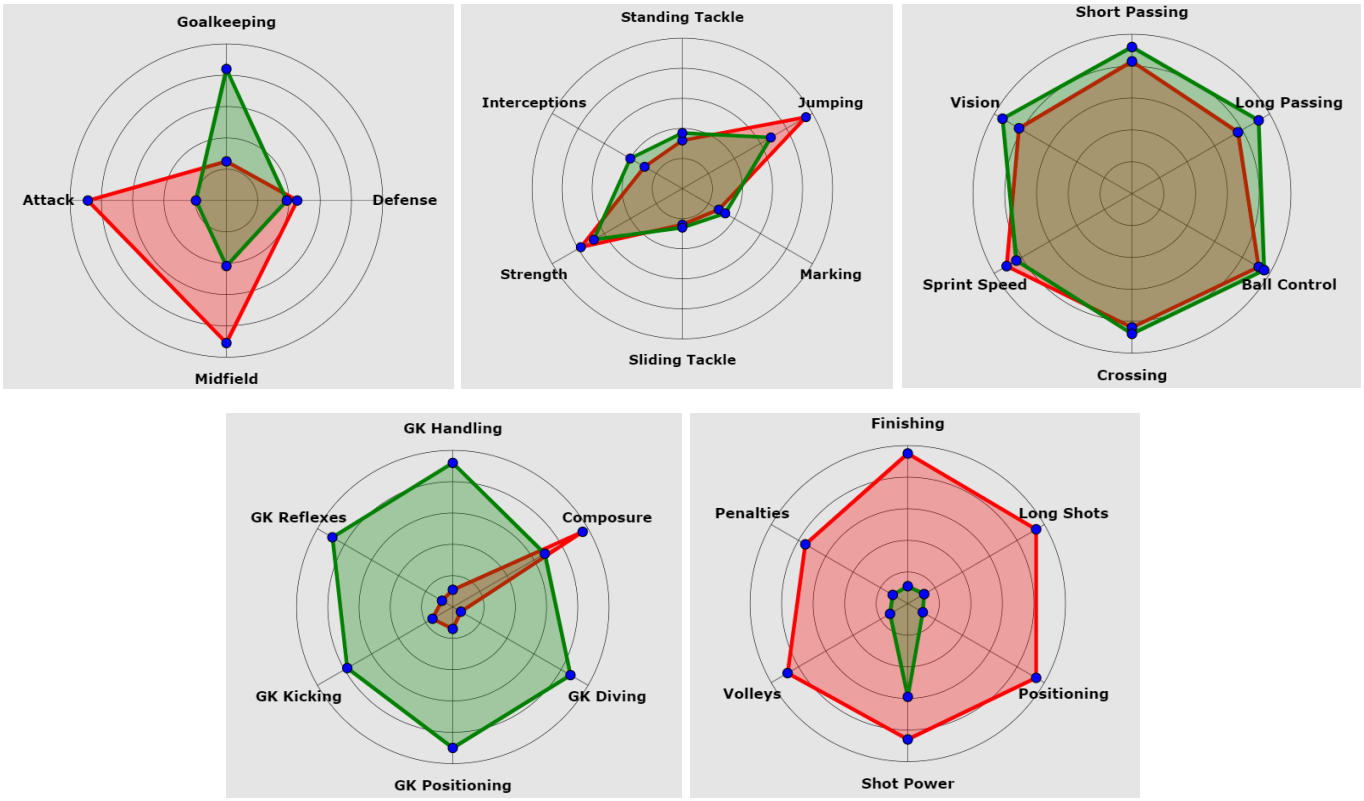
This is one of those design choices that has been able to successfully make the difficult journey from our initial proposal to the final implementation. However, the final product is way more detailed than what we had initially thought of. We first started off with a static radar plot with a fixed set of 6 chosen attributes for each of the 4 base positions- GK, DEF, MID, ATT. Here is the first version of our Radar Plot:



Although this looks appealing, it has some fundamental design flaws. One that Kiran rightly pointed out in the milestone meeting and we rectified it over later versions was that curved lines represent gradual change. Also, we completely agree with what Braeden had pointed out in the peer feedback review. A static radar plot with 6 fixed attributes is not very useful for a deep dive comparison. His suggestion was that we allow the user to choose attributes through a dropdown. We gave that a try, but it turned out to be obtrusive and complicated. So we implemented an alternative that is simple yet allows space for detailed investigation into attributes.

10.2.3.1 Switch Attribute Mode (attribute selection made simple)

In our current implementation a user is able to switch between goalkeeping, defense, mid field and attack attributes for any player giving him a total of 24 attributes for player comparison on the radar plot instead of the initially proposed 6 while maintaining the clarity of view on the plot.

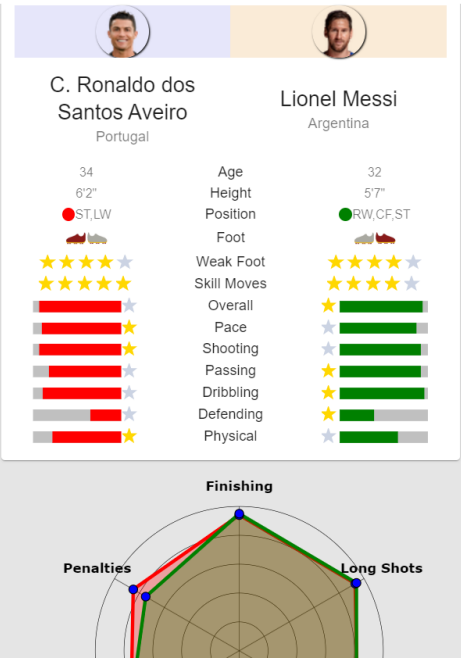


10.2.3.2 Radar Plot Position (another decision for space optimization)

Ensuring most optimized usage of screen space has been a major factor governing our decisions. We were initially considering rendering the radar plot as a separate component displayed below the player card. While this worked well for a single player view when the card was shorter in length, the comparison view would push the radar plot down the screen. This would require the user to constantly scroll up and down the screen and thus hamper experience. So, in the final design we made the radar plot a part of the player card to give the user one unified view and a seamless comparison experience. The following screenshots clearly show the problem we were facing and how our final design solves it:

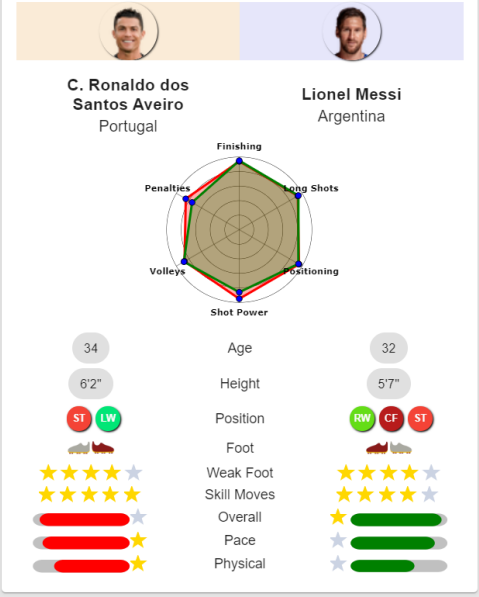
Click on plus icon next to column name to group by that column

Position	Name	Country	Club	League	Overall
RW CF ST	Lionel Messi	Argentina	FC Barcelona	LaLiga Santander	94
ST LW	C. Ronaldo dos S...	Portugal	Juventus	Serie A TIM	93
LW CAM	Neymar da Silva S...	Brazil	Paris Saint-Germain	Ligue 1 Conforama	92
GK	Jan Oblak	Slovenia	Atlético Madrid	LaLiga Santander	91
CAM CM	Kevin De Bruyne	Belgium	Manchester City	Premier League	91
LW CF	Eden Hazard	Belgium	Real Madrid	LaLiga Santander	91
RW ST	Mohamed Salah	Egypt	Liverpool	Premier League	90



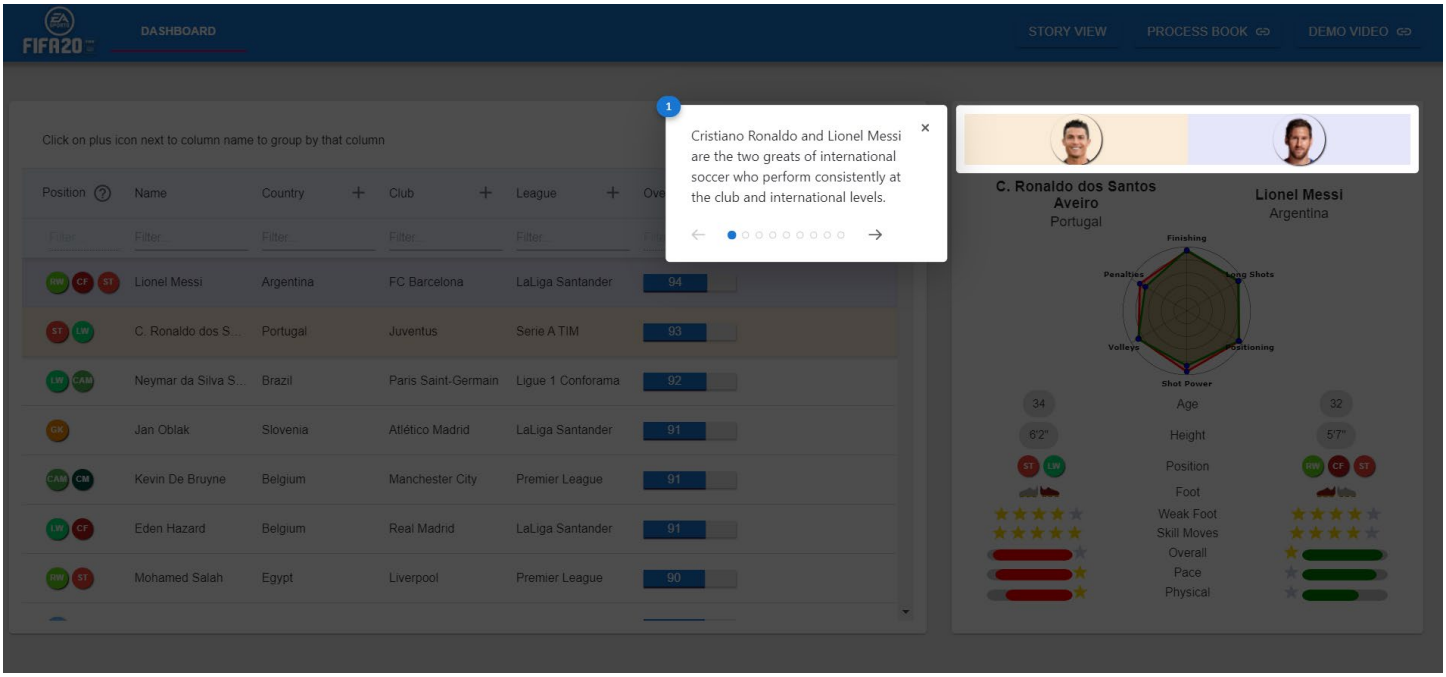
Click on plus icon next to column name to group by that column

Position	Name	Country	Club	League	Overall
RW CF ST	Lionel Messi	Argentina	FC Barcelona	LaLiga Santander	94
ST LW	C. Ronaldo dos S...	Portugal	Juventus	Serie A TIM	93
LW CAM	Neymar da Silva S...	Brazil	Paris Saint-Germain	Ligue 1 Conforama	92
GK	Jan Oblak	Slovenia	Atlético Madrid	LaLiga Santander	91
CAM CM	Kevin De Bruyne	Belgium	Manchester City	Premier League	91
LW CF	Eden Hazard	Belgium	Real Madrid	LaLiga Santander	91
RW ST	Mohamed Salah	Egypt	Liverpool	Premier League	90



11. New Feature Implemented- Story View

A visualization is serves it purpose if it can be stringed together to tell a story through the data. We decided to combine all the data we were representing in multiple views to depict something that would appeal to all soccer fans. So, we decided to pull up performance stats for the two top players of the current generation and see how our data backs their on-field history. The final product can be seen through our story view.



12. Feature Not Implemented- Wage Age View

While we have followed through and delivered more than what we had proposed in our initial plan there was one idea which did not fit well with what our final representation and so was left out as a carefully considered design decision and discussion with Kiran. The view in question is this:

13. Styling, Transitions and Animations

Our application has been tweaked to enhance user experience through applying transitions and animations where the data representation demands it. For example, the radar plot does not abruptly change states. The points and the lines move gradually to depict a gradual flow of information. Also, transition has been applied to the player attribute bar graphs to encode ordering. We have also placed tooltips wherever applicable. Here is a how styling helped give our final design the modern feel and appeal we were looking for:

14. Cross Platform Compatibility

We have ensured that our application is responsive and maintains its structure on the following screen resolutions-

- 2560 by 1600
- 1920 by 1080

We have also tested our website for compatibility on the following web browsers:

- Google Chrome Version 78.0.3904.108
- Firefox 70.0.1
- Safari 13

Since the rest of our application deals only with player skills and on field attributes, adding a wage related visualization would just feel out of place and give an incomplete feel.

15. Checklist on Proposed Must-Have Features

- ✓ Having the correct player list for a club is crucial. An incorrect list would make it impossible for the user to find the player he/she is looking for.
- ✓ The club-league mapping must be done correctly for the same reason stated above.
- ✓ User should be able to select any Club/League to see player ratings
- ✓ While comparing any two players, the points of comparisons must be determined based on the position of the first player. Comparing any two players with pre-determined attributes does not make any sense. For e.g., the important attributes of a goalkeeper are completely different from that of a striker. Once displayed, user should be able to modify any of the selected default attributes.