

### HTML: A

To begin with, I have written the HTML files from scratch. There is a home page, a portfolio page, and a few supporting popup windows. I have made sure they are all being delivered as XHTML to ensure correctness of their structure. During the building process, I have gained experience with common issues and tricks using a great variety of tags, that helped me make the HTML content convenient for further CSS style. Both popup dialogs have some html properties set for either just a more user-friendly interaction or simply data validation. I have utilized a lot of IDs and classes, in addition to some specific element properties, I have also learnt how to link scripts, styles and other materials from third parties.

What is more, I have experimented with free HTML5 template from HTML5 UP, namely "Helios", and even though it was neat and simpler than the others available, it was still overcomplicated for the purpose of my project. Hence, I stood by the decision of making my own HTML and I believe that contributed to my better general understanding of how html works and how it is done properly.

### CSS: A+

Just like the HTML, I have written the CSS from scratch for all the pages and windows. I have touched on a lot of details such as positioning of elements, transparency, border styles, transitions, animations, margins, fonts, overflow, different z-indexes (aka layers), display, cursor and scrollbar elements and also made use of the viewport units for personal convenience and easier position calculation. Furthermore, I have designed a hidden class that protects some elements I did not want a user to see, unless intended. Another thing I managed to do is fit the pages exactly to the browser's window and tackled some current CSS issues such as scaling and browser differences in implementation and rendering of stylesheets. One thing that surprised me was that the same monitor resolution could have a different width and height measured in different browsers. I came up with some solutions to these and I have also tried to make them work on all browsers. When it comes to mobile version, the cross-platform issue became even more complicated as it involves a bigger scale and in fact this is one of the main problems CSS has to deal with nowadays, however, I am going to investigate it further and tackle it before a mobile-friendly public release.

### JavaScript: A+

For the functionality of my web pages I have used not only scripts but also integrated a third-party plugin and probably the most common JS library - jQuery. I started by creating the main functionalities - adding, editing, and removing a player. Via the jQuery dialogs (popup windows), it is quite easy and convenient to create, edit and store player information for later analysis. The *add\_player* window consists of the general player information and to proceed you are required to fill them all. I found it particularly interesting to integrate the country dropdown plugin. It was simple but in the same time I had to manually adjust it so it fitted to the project - had to find a way to extract the name of the country from the alpha3 code (i.e BGR to output Bulgaria) or edit some missing country information. *Edit\_player* dialog has a similar structure. Further use of JS is all the communication between the server and the database (queries, writes and removes), whereas I mainly used jQuery to easily grab elements and manually assign and/or remove classes to them when required as the regular JS method was compatible with all browsers.

A main feature of the analytical part of the platform is the factors trigger button. It highlights footballers in my portfolio. For example, a player under the age of 23 is considered young and therefore would be marked green (safe to hold as he has multiple years ahead of him), while on the other hand, a player above 30 years of age would be marked red and that suggests he has less time to increase in price and earn me money (in Football Index investing platform players tend to drop in price after 30 so it seemed a sensible pick). Similar technique is used to determine underbought and overbought players (less than 1% and over 10% of the whole portfolio value). This is used to determine how much your portfolio depends on a single player. The percentages I have picked as the most common factors throughout the football index community. There are also scripts that estimate the actual profit you have made AFTER commissions are applied (which is not available in the FI platform). All this helps me with decision-making when it comes to my strategy for a player.

## Web Tech Report - Football Index Insider ( [fi-sim.site](https://fi-sim.site) )

### SVG: A

With Inkscape software I manually created the football index logo to give it my own theme and merge it with a third-party footballer model. I started off by creating a polygon of suitable size with 6 edges. To round the edges, I used a small circle as a model to find the path difference and after that break that difference apart into different pieces. I then removed the outer pieces and merged all paths and nodes together with the union feature. Next, I duplicated the resulting polygon and scaled it down a little. Following the process above, I used that to produce the border of the logo. Further, I started with the thunder-like arrow. I created a rectangle and rounded its edges as I did previously. After some rotations, more edge cutting and some free hand drawing, I finished the figure with a triangle, and the result is visible at the home page.

I then went to import a footballer model - hence I traced its bitmap with multiple picture scans of the colors. This procedure generated a path for the image (with nodes), and I was therefore able to merge it with the logo and export the paths as a .png file (with a transparent background and ready to be imported).

### PNG: A

My general idea was to create a blurry background that is quite popular recently, thus I used GIMP to edit two different versions of my favorite stadium (Stamford Bridge) - for both home page and the portfolio page. To begin with the former, the first step I made was to apply a medium gaussian blur filter. Next, I measured a suitable distance via the helper lines, and applied the same gradient from the edges to the center of the image (black to transparent). This caused an overfitting issue at the corners that I managed to tackle via the airbrushing tool. Further, it was time to import the logo and the footballer. I placed them a layer above the image and added some transparency so you could "see" through. In addition, I added a few text fields for each line, and generated a nicely looking Legacy drop shadow, using the "alpha to selection" feature on another layer with no transparency. That made the text stand out from the blur. And lastly, I adjusted the brightness and the color saturation to produce an eye-pleasing result. About the portfolio background it turned out that a simple blurred version best fitted the presented content.

### Server: A+

About the server, I have set up a domain and a web-hosting provider from reg.com. For the domain itself I have enabled the UTF-8 encoding for greater input restriction, HSTS to protect against cookie attacks and outdated protocols and other http attacks. I have installed a Domain SSL certificate from Let's Encrypt (RSA 2048 bits SHA256withRSA) which has been approved, hence the domain automatically redirects to https (meaning it is secure). I have also manually restricted the domain from working with python and CGI scripts via the ISPManager because I have no intention of using such yet.

To further check how my domain performed when it comes to security, I went on sslabs.com and analyzed it. They have come up with an A+ grade with 100% security when it comes to Certificates and Protocol Support and 90% for Key Exchange and Cipher Strength with both TLS restriction and HSTS proving to be the main factors. It has therefore been tagged as "Trusted" for all browsers and mobile platforms.

Regarding the webhosting, I have also set it up to have quite limited access. It only allows HTTP, HTTPS, SSH and FTP requests (port numbers 80, 443, 22, and 21 respectively). I have further set up an automatic ban for multiple incorrect SSH and FTP access requests. The default password has been changed to a newly generated strong one to prevent cracking.

### Database: B

To securely store the information of my project, I have chosen to work with Firebase Real Time Database. For the main purpose of the web, namely storing player information and making numerous changes on them, I decided that real time updates were an important factor and a neat feature that I must include. However, Integrating Firebase with JavaScript was though - importing the Firebase module did not seem to be enough hence I have included further scripts to support the methods I required. Eventually I made the queries available and storing/editing values inside as well, which was a great step forward. I am glad to have gotten real time updates on the changes, but it would have been even more secure if I required authentication and required admin privileges to those actions.

To make this happen, I had to work with Firebase-admin which allowed authentication but is a different module and works with different methods and that would mean a major re-design which I did not want to commit to this late into the assignment. To avoid any major data loss, I have temporarily disabled the remove method of the database (until I provide a solution, remove is available for me directly from the database).

### **Dynamic Pages: A**

With regards to the dynamic page generation, I have done it wherever possible at the current state of project. There is no manually generated page apart from the home page and the templates, both of which have no changeable content anyways. I have implemented the dynamic generation via the html templates even though I am not using the template tags. This is for both *edit\_player* dialog and the main portfolio space on the portfolio page. I have created the so-called place holders (i.e divs) and I am inserting the data that comes in from a server request (and is also generated on the server-side) into them. This took a lot of effort to make but I can totally see why it is needed and how it affected my overall code readability and habits.

### **Overall outcome and general idea of this project:**

Well, to start with, Football Index is a platform that allows users to invest into shares of footballers. As I am quite active in there, I noticed it is a quite limited platform when it comes to analysis and legit profit calculations, thus I decided I would go on with this personal project. I did it on my own because it is a topic that I personally find enjoyable and I wanted to do the details myself and according to my taste. It was also easier to set up at the time.

The main purpose of my platform is to help with further analyzing football players and actually calculate profit accordingly (the main football index platform earns money from commissions but they are never mentioned until they have taken the money from you, hence you might end up with a loss while you believe the opposite). Not to mention that in order to find the actual amount of the commission taken from you, you would have to go through so many pages of all types of transactions registered on your account. In other words, it is a difficult and annoying job to find that value and it is quite easy to miss it out. Even then, you would have to make the calculations using a calculator - one by one.

So, my solution to this is quite simple and it shows the commission on the portfolio page and re-calculates the profit based on those (including dividends) should you sell your shares of the particular player at that exact point in time. It also offers me great background support for my transactions. A sort function to sort the portfolio on different columns I might include in the future but for the submission I focused on each category in the requirements list and it slowed down my overall progress with functionality. A filter function I will definitely include in the near future that would work on the factors and only display those players that the filter setting applies to.

A good time to mention about the Football index API - it was supposed to provide live data (buy and sell prices) - so it really motivated me to push the limits. However, due to current world situation it was delayed for later this year and I am yet to integrate it (it was in the initial plan and design). It was not to be for this submission. (Basically with the API, the current price that is shown on the portfolio would be live and not manually added/edited every time it changes. That would also force live updates on the other fields such as profit and commission. This is also why I chose to work with Firebase Real Time Database. That, however, I think is the major set-back in the current design of the project (along with database authentication issues).

In conclusion, I would like to say that am quite happy with the outcome I have achieved so far and this course has motivated me to experiment with many available opportunities, to publish the platform online and continue working on it afterwards because I firmly believe it could be useful, if not for other users, definitely for my own.