**Development Log**

First of all, we gathered up as a team, and started a little bit of a brainstorming, generating different ideas of things we wanted to accomplish with this project, thinking about what kind of blog we wanted to do, and somehow a general path of how to do it, while also choosing the tools we would be using.

Following this, we connected remotely using Liveshare, a Visual Studio extension that allows the user to share a project’s files to the team so they can all collaborate and work together. We started the project by installing the node modules, and creating the main folders and files for our content, such as the header and footer template, index, the javascript file for the server and the css file.

Afterwards, we worked on setting the server up, writing the necessary code to get the node integration for our landing page and started adding base content for the blog. We designed how it should look like. As we continued with the development we faced some problems, the first one was getting the data from the API: NodeJS tried to parse the data before it was fully extracter, it took us some time to find a solution to this. In addition, suddenly the mouse was not scrolling. It took us some time to solve the issue, checking HTML, CSS and JavaScript logic to no avail. Restarting the server was enough to fix this. Another problem worth highlighting is that one of our Post functions was not working properly, we tried to get information from a paragraph element. After we decided to take a break, and after researching a bit, we found out that the body parser does not get information from non-input elements.

One of the biggest issues we found was to make the blog editable, we solved this using an array with dictionaries which contain the data to be displayed per blog. At first, everything was working fine: it was relatively easy to display the updated text, but this was not the case with the image files. After some research, we discovered that with the library express-fileupload we could save the image on a local directory and then we could display the file the same way we did with the text.

Finally, when the web page was completely functional, we took care of smaller details, such as changing and adapting the feel and look of the pages. We chose not to have a specific topic for the blog, giving the project a focus similar to that of Blogger, for example. Each one of us wrote an initial post for the blog about different topics, while the user can write about anything they want.