Imagine you are a senior software developer and you are spearheading the development of a new product that you hope will have an entirely cloud-native backend. However, this will require you and other analysts to perform a time-consuming market analysis on cloud platforms just so you can begin the debate of which one you should use. In the end, you decide on a platform but at the cost of many days that could have been spent wrapping up the design phase and beginning the development of the app. This is where our website, CloudFrontiers, comes into play.

CloudFrontiers offers ready-to-go listings of different cloud products and various details including what platform they are offered on, pricing, security info, storage, bandwidth, and the cloud service type. This allows the user to get a holistic view on how the product works and provides metrics by which they can compare the other products. Using these metrics and the results given, the user can then either use the results as their final market analysis, or can build off of it and expand the list with more products, using the products given by CloudFrontiers as a baseline to compare to and understand if the suggested products holds up to it or not. CloudFrontiers is not just aimed towards Product Managers or Analysts, but also to those who specialize in. CloudFrontiers focuses on user interactivity using JavaScript and static data pulled from a JSON file.

One of the key decisions made during the project planning phase was the choice of technologies. For the front-end development, we opted for HTML, CSS, and JavaScript to create a responsive and user-friendly interface, from this we derived our primary focus area to be our javascript so we could focus more on animations on the data and the interaction of the user. For the back end we initially decided on using a sql database and PHP to display the packages onto our web page. This is also where we decided what parts of the project we were going to work on, we split it up so that Sarvesh would be working on the HTML and Chris and Hayden would be working on the back end with the database, and jQuery. Here is our initial timeline: From the week of 3/13-3/17, the first product comparison page.

Primary tasks for this page include implementing the search feature on the first product comparison page with JQuery. The week of 3/24 to the beginning of the week of 3/31, the development of the second comparison tool page. Primary tasks for this page include implementing the drop-down menu to search the products to compare using Javascript and simple tests to pull information on the specified products from the database. The final two weeks of our project were allocated to connecting our front end with the backend. The back end development for our project involves managing the SQL database so that we can ensure that the database seamlessly interacts with our front end allowing our users to have a smooth and responsive experience.

We stayed true to our initial plan for the most part, we ended up spending more time than planned on the sql and PHP parts of our project that will be analyzed in further detail in the project challenges section. Though we had faced some obstacles, with our proper coordination and dynamic problem-solving, we were able to overcome those challenges and get our fully functional website completed.

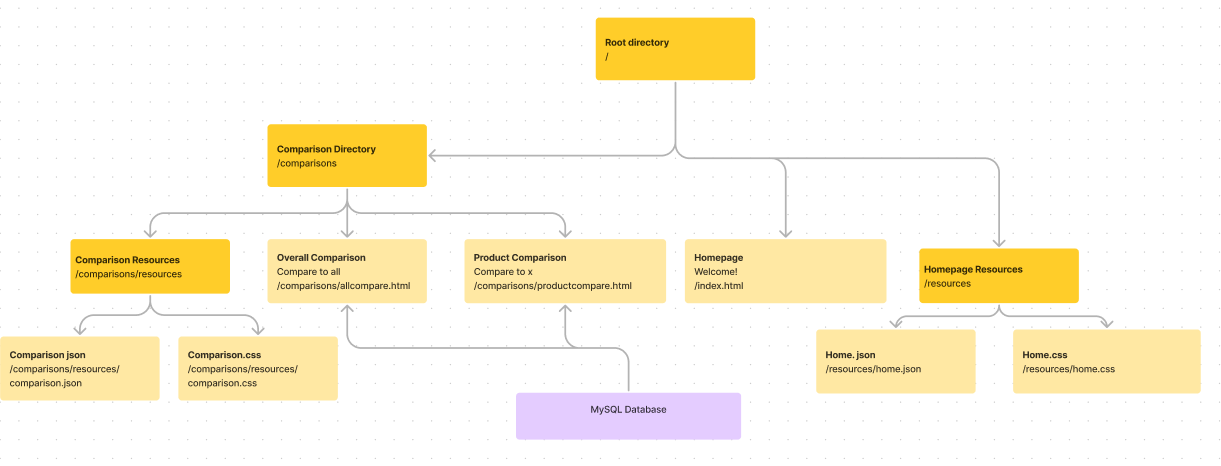
The architecture of our project is fundamental to its functionality and efficiency. The following model is an example of our finalized IA. 

Figure 1: Finalized IA

The yellow rectangles are files, while the bright yellow rectangles are folders. Specifically, we have a home page, an initial comparison tool so a user can compare different platforms for a specific product, and a specific product comparison tool to compare two different platforms’ offers. We also contained our database json and sql files within a back\_end folder to store our information. There were some changes we made throughout the project including removing an initial plan of making a catalog page, our addition of our file export page, and our change to use a single CSS for all of our pages.

During development, we ran into various issues with utilizing the SQL database. We encountered multiple problems using AJAX post requests to the PHP file that formed the connection to the database. The post request would be incapable of posting to the PHP file and would not have detailed information regarding errors, so it was nearly impossible to debug it. As a workaround, we created a JSON file that contained the same information as the database. We used AJAX in conjunction with that instead, refocusing our efforts on pulling data from a static source rather than the frustrating usage of the MariaDB. We also encountered issues combining the HTML code and our focus on JavaScript. The HTML is a necessary aspect of our project

Our results were slightly different from what we expected, as the website did have more functionality and webpages than initially planned, but we did not implement the SQL database we initially planned for. This project shows only the beginnings of the potential a service like this could offer. Additions that could be made if given more time could include a web-scraping feature to get real data, a larger effort on the HTML/CSS side to beautify the website, and adding recommendations for products based on specs, as well as links to their product pages. This website has a lot further to go if we want to make it an actual product for analysts to use, but I believe that what we have now are the building blocks that demonstrate its use and how valuable a resource like this could be.