Visma Solutions - Summer Trainee 2023: Software Developer

Leevi Kämäräinen

For this programming task, I used JavaScript as my programming language.

In the task, I had to parse given URI and validate the parsed URI so that the given requirements are met. I made a strict assumption in that the URI must always match the following format:

<schema>://<path>?<parameter_key = parameter_value>.

This meant that any given URI which did not match the formatting, would not be a valid URI. This of course means that any typos in the given URI would result in invalid parsing. Here I also decided that even if the URI is not valid URI, the URI_identification object will still be created, but the variables for path and parameters would give information on what part of the URI is not valid. This decision was made because I thought that it would be more important that the objects are always created in case of the other apps/user requiring the objects to exist.

I did not have much experience with object-oriented programming with JavaScript, so the first challenge was getting more familiar with that. In the task itself, the biggest challenges were that how I would manage parsing the URI (the above assumptions helped with that), and how I would validate the paths and their parameters smartly. I decided to store the valid paths, and their parameters inside the objects functions, so that the parsed URI could be compared to those.

The actual parsing of the URI could be improved to be done in lesser steps so the code would be more clean. For example, using regular expressions or using split() -> join() function repeatedly I could have parsed the URI into an array in one line. However, since parsing the URI step by step is more intuitive, I decided to leave it as it is.