Chapter 2 of Team Term Project Report

Introduction

Since our previous Deliverable, our group decided to switch the HFOSS project we focused on. We decided to switch to OpenAQ because STEM included many problems for our group including:

- Issues finding all packages of the source code.
- Building what little of the source code we actually could obtain.
- Not being able to successfully build the project on multiple machines running Ubuntu Linux.

With a lack of running software to develop or run tests on, our group could not continue making progress within the overall scheme of the project. Therefore our group has decided to work with OpenAQ as our new #1 candidate HFOSS project. OpenAQ includes:

- All source code needed to properly build from scratch.
- Guides to build the project from the source code.
- Project being written in JavaScript.
- Successfully built and inspected on a few machines already (more on this later).

Accomplishments

During project development for Deliverable 2, our group has accomplished writing a test plan, which includes the hierarchy of how we are to proceed with testing different aspects of OpenAQ throughout the rest of the semester. Our team has also written how we would test certain operations within OpenAQ , 5 out of 25 is what we wrote so far.

Learning Outcomes

Our team has learned how to develop a test plan for a detailed method of testing different parts of the software from bottom to top. The work done helped develop a further understanding of how to write test cases for a complex software which includes multiple components and interfaces.

Problems Experienced

Even though the guides for building OpenAQ appear to be well written, the steps to successfully building OpenAQ sometimes appear to be out of order. At particular steps, a software package would not install because prerequisite software were instructed to be installed afterwards. Installing all software components when needed resolved many of these issues.

Going forward

The next goal is to keep uncovering more information of how the software functions in order to develop further relevant test cases. As our team learns more software engineering principles to apply, we can further strategize our resources efficiently towards completing future Deliverables for OpenAQ.