Project Milestone 2

As a Boilermaker pursuing academic excellence, we pledge to be honest and true in all that we do. Accountable together – We are Purdue.

(On group submissions, have each team member type their name).

Type or sign your names: Parker Bushey, Kaitlyn Roberts, Ashwin Senthilkumar

Write today's date: 9/20/2021

1. “Substantial” updates to your design (I leave this definition to your engineering judgment).

The project is now using the jnpm package to pull github repository urls from given npm modules, jcabi-github package to interface GitHub API with Java, and JGit to pull repository contents. These packages have been downloaded as jar files to minimize the need for installing dependencies.

Scoring calculation code is implemented/organized via classes for each, and a parent score class for shared functionalities. A main file will create instances of these classes for each url and call a function that returns the associated score value. There is also a url class that will hold each score instance and other data, such as the original url, the domain of the url and the name of the associated Git repository.

2. A statement of the tasks your team accomplished by this date.

While the milestone goals were not completed in their entirety, we believe that we have completed a solid foundation on which we can build the rest of what we intended to have done. We found and set up the modules needed to interact with each API and have successfully gotten test data from GitHub API.

Kaitlyn (Katie) researched which packages we would need to import to implement the project in Java, downloaded the correct jar files into our repository, and ran tests to ensure the packages were being imported correctly. At the time of writing, all packages imported successfully except the jnpm package. The import of a class called JNPMService is not working, which is an important class to make jnpm work. Katie also created the basic repository organization, the makefile, and env file. Finally, the parseURL function was made in the main file to pull the domain and repo names out of the urls given using a scanner to import the file and separate urls, then regex to find the domain and repo names. The function still needs functionality to implement calls to the APIs using those variables. Overall, researching the packages took about 5-6 hours, importing the packages and running the tests took 3-4 hours, and the rest took about an hour.

Parker built on the research done by Katie in choosing the API to interact with GitHub. He read through the documentation of the module and was able to access data that would be needed for future milestones using one of his own public repositories as a test. There was a large portion of time (3-4 hours) spent trying to set up the necessary modules needed for getting data from GitHub API. After successfully setting everything up and verifying it was working correctly, more time (2 hours) was spent reading the documentation for the jcabi library. Another hour was spent verifying that all the needed data from our specifications was reachable using the methods found in documentation.

Ashwin worked on researching on GitHub API and gathering information on what to include for the Bus Factor metric calculation. He spent the week before and this week setting up the environment and cloning the repo. He read through GitHubs documentation on using the

REST API to get stats on a repo's outside collaborators and traffic. He spent some time watching YouTube tutorials on how REST APIs can be used and learning more about npm. The total amount of time spent on knowledge gathering for this milestone was roughly 5-6 hours. Because most of the time was spent learning and gaining a better understanding of all the tools, there was not as much progress made on the programming part as expected for this milestone.

3. A comparison of what you accomplished vs. what you planned to accomplish. Are you on track? How were your time estimates?

For this milestone we had hoped to have had the function for getting all the data from a list of links completed along with two of the scores (licence score and bust factor). We fell short of the goals, only getting a start on the repository list function. The other two functions rely on the data gathered from the list function, so those were not started as of this milestone. Because of this we are behind schedule. This first milestone is taking significantly longer than we had anticipated mostly because of the large amount of research and learning that we needed to complete. After we finish creating our foundation for using the APIs, we predict our time will be spent more on accomplishing our initial design instead of heavy research.

4. Any changes in your planned timeline as a result of falling behind your initial plan. If you *deviate substantially* from your timeline, consider attending one of the course staff office hours to discuss the deviation.

To accommodate for falling behind, we will move the completion of the repo list function to the next milestone. We will also attempt to complete the two score function by week 3 or 4. We all plan to attend office hours on September 21st to discuss what we might do to improve our progress moving forward.