Michael Flatley

COMP 412

**Open Source Project Evaluations**

For this project I have chosen to Discuss the following open source projects:

* Food Inspections Evaluation from the City of Chicago (<https://github.com/Chicago/food-inspections-evaluation>)
* Prometheus (<https://github.com/prometheus/prometheus>)
* Simulation Craft (https://github.com/simulationcraft/simc)

The Food Inspections Evaluation project is one that was created to assist the Chicago Department of Public Health by finding patterns in their public data to reduce the time it takes to find restaurants with critical food violations. Since its conception this project has helped reduced the number of days it takes to find these violations by seven days.

Since this project was added to Github, in 2014 there have only been six contributors indicating a lack of developer interest. This may have to do with how specialized the domain of this project is, the use of the R programming language, or the extensive amount of data testing that is required before contributions can be added to the project.

Even though there weren’t a lot of developers who contributed to this project, there is a significant amount of use for users within the Chicago food inspecting domain. This is made evident because since this project was first published for use the overall time to find critical health code violations was reduced by seven days.

Developers who would be interested in working on or reviewing this project will be able to do so easily since the documentation for this project is very thorough. The ReadMe file contains sections that describe what libraries are used, how the files in the project are structured, and what data was used to develop the data model. The project also has a file titled CONTRIBUTING.md which is used to describe to developers what they will need to do to contribute to the project including a measure of improvement, a description of the changes that they have made, and a clear definition of what data needs to be used to perform tests on their changes.

The work put into the documentation detail is also evident when looking at the individual files in the project. Each .R file maintains the same type of indentation and uses camel case for the naming conventions. The purpose of each method and section is also clearly defined in plain English by the developers’ use of comments. The use of these software engineering practices helps to improve the readability for developers who are familiar with R programming and for those that are new to it; which is an important quality for Open Source projects.

Being an open source project, there are opportunities for others to contribute to the project through Github. However, it appears that the only contribution that this project needs would be input on ways to improve their current algorithm; the base functionality would not need to be changed. There are also four items listed under the “Issues” tab that need working on, but several of those are over a year old and it isn’t clear if those issues have been cleared up at all. With all of the opportunities for people to contribute, it seems that the project is has hit a standstill and I don’t perceive there being many more updates.

The Prometheus project provides a monitoring service to organizations that are interested in observing and receiving reports on systems or services and then sending alerts based off of user set rules. Unlike the previous project this one has much more developer interest; this is evident from the one hundred one contributors. Another factor that may account for the developer interest is that the service doesn’t have a specified domain, which allows it to be used by a wide variety of organizations.

This project does have evidence of user interest. The Github repo links directly to the project’s website which clearly displays over twenty organizations, such as SoundCloud and Ericsson, which use the service provided by this project. The project has even been included in O’Reilly media’s book *Site Reliability Engineering: How Google Runs Production Systems*. The Prometheus monitoring service has garnered enough developer and user interest over the past several years that the project hosted a convention in Berlin this summer.

The documentation of the Prometheus project is very clearly composed and informative. The ReadMe file contains a link to the Prometheus.io website for users to access released versions of this software, along with configuration information for the project and libraries required to run the project. There is also specific information about what kind of environment is needed to run the project. This project also has a Contributing.md file that describes how one can report bugs in the project, or contribute to improve on the project. The instructions on how to report bugs are very clear, since Prometheus has a team of developers whose job it is to maintain the project, but the instructions on how one can contribute are severely lacking. Users are advised to discuss ideas in their mailing list before working on the idea.

The file structure for this project is a bit hectic making it difficult for new contributors or code reviewers to determine where code will need to be modified to either add functionality, optimize code, or remove bugs. This project has over 15 folders in the main folder of the service, and while minor descriptions have been added to these folders in Github, it would still be difficult for developers without knowledge of the system to navigate through. As far as the files themselves, there appears to be a lot less commenting on this project which may be the result of contributors who do not follow all of the coding guidelines described in the Contributing.md file. Despite the lack of comments the contributors appear to all conform to the same stylistic requirements requested of this project.

The Prometheus project has a lot of opportunities for new developers to contribute to the project. If users find bugs, the Prometheus team has outlined ways that these can be reported. The bugs are then added to the “Issues” tab where one can see the issues that have been resolved and reported. The project leads also appear to be constantly updating the category tags on these requests, and marking some with a Help Wanted tag. With issues being reported and resolved within the past week, it is plain to see that this is still very much an active project that could still use developers to help fine tune their published service.

The last project