Kyle Cullion

M08747336

CS Senior Design II

Professor Annexstein

Final Self-Assessment

Part A:

Among my team of fellow developers on Heimdall, I made several contributions throughout the lifespan of our project. My contributions were in several different areas of the development process. I helped in the composition of the schema for the PostgreSQL database instance used for Heimdall and made continual necessary updates throughout the development cycle. The updates to the schema were made after discussions of metrics added to the original set of collected metrics. I was involved in the discussion of which metrics we should aim to expand our agent to collect and as a result helped with making several of the table additions and setup. Along with updating and maintaining certain tables and relationships within the database, I aided in the creation of some of the associated back-end bookshelf library files for the tables and the creation of the data to be sent within the API code. Along with development in the back-end portion of the application, I contributed to some front-end developments as well. I mainly worked on the disk info usage visualization when it came to the front end. I aided in displaying the related information to the user such as path of partition, used disk space, and free disk space. Also, I added in the dynamic changing of the progress bar's width in accordance with the used disk space and the alternating color according to space used and other design choices with this section of the dashboard. Working within these areas of the development led to me expanding my knowledge base by a substantial amount.

The various areas I listed in my initial self-assessment were mostly all areas that I expanded my working knowledge base in. First in my initial self-assessment, I mentioned my coursework in computing systems and my previous experience with the internal working of computers and the various hardware. This development on Heimdall has vastly increased my knowledge about computing metrics. Working with Zack, who led agent development, allowed me insight into what metrics would be insightful into the low-level insight into the system's health. Our discussions pushed Heimdall to further aggregation than we ever thought we would accomplish initially. Researching the various system metrics and their corresponding aspects between Linux and Windows systems were key factors in guiding my understanding of computing architecture and allowed me to play a bigger role in the development process than I could've played prior to this project. Having worked on Linux systems primarily I was not familiar with all the corresponding system components and their place within the Windows architecture. Thus, I had to delve into the different architectures to really aid the development process of Heimdall on my part. Another skill that was greatly honed during our project was in terms of the software engineering production process. I had previous coursework and co-op experience in this realm, however I had only ever experienced one method during my

development career. Our Heimdall team's implementation of the development process shined a light on how much faster and more efficient software development can be with a small team. Our ways of divvying out modules tailored towards each person's strength allowed us to achieve our goals quickly and fully. No unfair compromises were made since the discussions we held on a regular basis during meetings were open and honest. The approach to airing out grievances freely altered my perspective on software development as most of my development experience came from large, corporate teams. Along with the development methods that I learned, and my teammates expanded my knowledge on, I also learned many new technological skills. I was somewhat familiar with Golang, PostgreSQL, and NodeJS coming into our project. Yet, these three technologies are the cornerstones of Heimdall as an application. My understanding of these technologies and their underlying frameworks are now what I would consider solid enough for me to seek out future development with these technologies. Vivek, one of the other developers of Heimdall, enlightened me on the power of leveraging NodeJS as a back-end framework for an application and taking away the major headaches of working with JavaScript. My successes were helping implement the schema in PostgreSQL relatively seamlessly and easily navigating the framework that NodeJS supplied Heimdall to successfully route the metric data smoothly, quickly, and accurately. There were obstacles I encountered throughout my development on Heimdall. My little AngularJS expertise gave me trouble when dealing with the front end of the Heimdall application. My front-end development during this project turned into me having to research the AngularJS documentation thoroughly to ensure my contributions on this front were worthy of being a part of Heimdall. This was one of my struggles. Another struggle was just understanding the necessary aggregation to make for a smooth user experience when viewing the Heimdall dashboard for any of their registered agents. The discussions we had to remedy loading times and handling the large amounts of data being moved were tough for me to suggest a solution and I had to rely upon my teammates to devise the solution for that specific problem. Even though I myself had obstacles during Heimdall's development, I was aided by being part of such a great team that could overcome such issues together.

Part B:

The group that I was a part of during Heimdall's development accomplished most of what we set out to do. Although we did not accomplish all our goals, we accomplished all our main goals. The objectives we did not meet were more stretch goals than key components of the application we set out to create. Our product is an application that we as developers are proud of creating. Our product is now a simple to use, open-source alternative to the proprietary options in the field of metric monitoring and visualization. This whole experience with my group has shone a light on group work that was somewhat dim from previous development work in my past. As I stated before, my experiences for prior application development were in a large, corporate setting for the most part. These corporate experiences, whether it was the company or team's fault specifically, left me leaning towards individualism in my projects. However, this group showed me the power and influence of team choice. Being comfortable with team members has enlightened me to how much more enjoyable group work can be than individualism in software development. Working with this group showed me that software development, while just as challenging with a team, can be more fun and as a result more creative. This dynamic of

the group was one of the more successful aspects of our process. Our discussions and meetings were key in ensuring an easy development. The group's rule of honest and open discussion led to faster and more accurate development of our ideas. This was excellent in leading to all of us making compromises that were never made of guilt or annoyance. The product was better for the group airing out grievances freely and not being afraid of what one another may constructively criticize. Along with these successes though, we ran into an issue or two with such an open process. Sometimes, changes would be made and one of the members of the group would take advantage of the trust level between all of us on a development level. This led to a few instances where there were somewhat complicated conflicts having to be resolved by the group. Another issue was that most of our meetings were virtual for the middle to end of the development process. Initially, this was due to work and school schedules conflicting, however the pandemic of novel COVID-19 was the main factor. Due to the pandemic, we have held our meetings purely virtually for a month and a half. This caused some strain in terms of our process, but we adapted successfully in my mind. The very few issues never blocked any of the group from making impactful contributions towards the development of Heimdall.

As part of the Heimdall development group, I believe I contributed everything that I set out to achieve and was allotted in terms of work during Heimdall's development. Through my work across the database schema, some API back end development in terms of data transfer, and some front end work, I completed all of my assigned modules and helped out when other areas needed work or a fix that I was able to devise a solution for to submit. Ideally, with a team of four the work would be slit up 25/25/25/25, however due to the differing strengths of the team the work was not split entirely evenly. My fellow team members all pulled their weight in terms of expected and unexpected contribution to the development of Heimdall. Michael was instrumental in building the UI framework for Heimdall to allow the design to communicate the metric visualization effectively to the user. Michael's sleek and user-friendly UI created the best user experience we were hoping for when using Heimdall's web interface. Zack was the lead developer on the monitoring agent and without Zack Heimdall would be useless. His developed agent smoothly collected the metrics with barely any notice on both Linux and Windows host systems and quietly and consistently sent the expected metrics to be visualized. Zack also heavily helped in other facets of the project such as devising an aggregation that allowed for extremely quick updating of the dashboard view for such a large amount of polled data. Vivek was also very important in his contributions to Heimdall's development. Vivek's expertise in PostgreSQL allowed us to easily compose an effective schema that stayed largely the same throughout the entire project with only the slightest of modifications being made to accommodate stretch agent metric collection goals. His previous experiences in setting up NodeJS back-end frameworks were crucial to alleviating the pains of usual back end development. His method of splitting up the back end into the library, routes, and utilities directories that comprise the API and database connection and configuration led to our success in creating Heimdall. So, while I contributed what was expected of me and a little more, Heimdall would be nothing without the work of Zack, Vivek, and Michael to help create and refine the application to meet our goals!