Project Reflection

Throughout the creation of the project, RateMyCourse, the team Course Warriors learned many things about different software development methods and tools. In the creation of the website, we utilized the Google Cloud Platform. After the website was created, Team Course Warriors used the tools XAMPP and Apache. Finally, for the team's communication, we made use of Trello and Slack.

The Google Cloud Platform provided the team with a powerful database management tool. Once an instance of the project was created on the platform, a MySQL database was created via command line. The team utilized four tables to store the University of Colorado - Boulder's FCQ data from the last ten years. These tables were CourseName, CourseDifficulty, CourseRating, and GradeDistribution. CourseName holds all courses with their subject, course number, and course title. CourseDifficulty stores the average grade, average hours spent, and the raw workload for each course. The course rating, professor name, and professor rating are held in the CourseRating table. And finally the GradeDistribution table stores the percents of students who received A's, B's, C's, or D's/F's. The team used a command line connection to the instance to create the tables, but then used the Google Cloud Platform to populate the tables.

The use of the Google Cloud to host the database proved to be very successful. The interface provided an easy-to-use tool, and the import feature allowed for easy database population. To populate the tables, the team compiled the FCQ data required for each table and stored that information as four separate CSV files (one for each table). The import feature on the platform then allowed for the data to be imported via those files. As database administrator, team member, Jackson updated the tables as necessary and granted team members access by adding their IPv4 addresses to the project's authorized networks. While this occasionally caused some delay in the development process, it did not prove to be a big obstacle.

Once our database was up and running, we needed a localhost connection to link our website to the information in our database. This was done by installing a local web server solution named Xampp. Xampp allowed us to run an Apache HTTP server locally from our machine so that when we typed the localhost's url on a local browser we could actually see the webpage and all of our code in action. Xampp also allowed us to run a MySQL server instance locally on our machines so we could check that the search engine and data visualization were being handled properly. The MySQL server allowed us to connect our webpage to the database hosted on the Google Cloud server and perform the necessary MySQL commands needed to search for any given class and pull all relevant information associated with said class.

Our project consisted of essentially two major parts, which were the ability to have user specified searches and data visualization. We used two tools to implement these functionalities, PHP and Google's data visualization API. PHP is a scripting language especially suited to be embedded in HTML-like pages; it is a hypertext pre-processor which process all the requests to our server and passes data to and from our 3 different web pages. The search engine was completed entirely using PHP; it handled posting the user's search query from our search page to the results page, where subsequently it performed a MySQL command to get all the matched courses from our database and displayed them as hyperlinks. Once a user clicked on one of these hyper links the course subject and number were then posted to the content page, and again MySQL commands were issued to get all relevant data needed for the data visualization. The information was then passed into the Google visualization API, which was written in javascript. This processed the information and created charts that would finally be displayed for the user's convenience.

Looking back on which project management tools our group was able to utilize the most, the instant messaging application Slack turned out to be the clear winner. We used Slack for almost everything: communication about project milestones, planning meetings, reporting issues with the

project and general discussion of solutions. The ability to compartmentalize our work from our personal lives and other classes as well as the instantaneous nature of the messaging made for an extremely helpful tool, especially when it came to authorizing different IP addresses. Slack became essential as one of us would simply message Jackson, our database administrator, and he would grant access within the hour and work could continue. This would likely be a much more difficult or tiresome issue if Slack were not used.

Trello on the other hand, a project planning application, was used much less frequently and in fact was completely abandoned after the plans were made for milestone 2. I think that the tool itself was useful and that it offered a lot of features that we didn't take advantage of due to the scope of our project. With such a relatively small project and only a couple of roles being doled out to the individuals of the group, having a large board with tasks, steps, and comments did not prove to be of much use. If our project had been much larger with many more features to develop, Trello would have made much more sense. But as for our that project, that was not the case.

There were a couple of examples where Trello could have maybe helped our project and made for a better product. For example, when deciding what metrics were to be implemented and visualized for our course rating system, a planning tool like Trello would have came in handy. There was some miscommunication about how certain metrics were going to be implemented, specifically the course difficulty metric. This metric was supposed to simply be a combined average of the students over all opinion of the difficulty of the course. Instead it was first implemented as the average GPA of the course. This was caught and resolved prior to the presentation. However, if we would have followed a system like Trello's where we had tasks, steps, and comments, these miscommunications would likely have been avoided and time saved to work on the project.

Throughout the semester, Team Course Warriors used many software tools to create RateMyCourse. These tools helped create the project, connect the website to the server, and helped with team communication.