Final Project Report

1. In our final project, we have accomplished most of the goals in the proposal. In our final work, our website is able to search a Gened course based on the categories, distinguish if a course has an online section or not, and is able to help the students create a list of easy Gened courses.

2. Based on the usefulness part of our proposal, we provide more detailed information for each course: professor's name, section, course number, department, A rate, etc. We also categorize the courses even by years, semesters and sections, which can help students better make the decision. However, we did not provide visualization for the general education courses' grade, we achieved all of the rest of the usefulness we stated in the proposal.

3. We didn't change the source of data to use. In the original schema design, we only considered what attribute will be useful to our application. Once we gathered all the real world data, we had to make proper changes to our database to better support the data import. Also, there were few edge cases that we did not cover in our initial table design, and we were able to fix them by closely examining the data.

4. Despite the attribute changes, we did not do much modification on the relations. For the attribute changes, we divide the "grade" attribute into letter grades, and each letter grade has its count; We also make different assumptions on both of the Grade and Course table in which we import all UIUC course grades into the Grade table while the Course table only contains the Gened courses.

5. As described in the first part, we removed the function for searching the course based on the professor's name. As we think our application focuses on helping students find the easiest Gened courses, searching by professor name is not a necessary functionality. Also, we give students a chance to view and leave comments for various courses.

6. There are a few different aspects that our application benefits from the advanced database program. First and foremost, we are able to monitor the usage and have it hosted on GCP rather than starting up the database in our local machine. Secondly, we are able to organize the data and query the data in an organized database which saves huge programming effort by using relational database.Lastly, our stored procedure is able to update the A rate table with the latest needed information once it is executed.

7. Dongming Liu: We had a difficult time connecting the frontend and the backend because we were not too familiar with the functions, libraries, and APIs that establish the connection between the frontend and the backend. The recitation section about connecting frontend and backend applications was not that useful because the recitation discussed using vanilla javascript, but we used React JS for our frontend and python for our backend. We ended up searching online and found out that there was an awesome

library called “Axios”, which was used to set up the connection from frontend to backend. In addition, it was very easy to use. If teams in the future decide to use React JS as their frontend library, they should check out “Axois”.

Owen Xu: We have a hard time learning using python and actually writing code using python.
Michael Fadillah: Local machine was missing some dependencies with the frontend module even with all the dependencies installed.
Xinyi Wei: I had a hard time learning Python, how to execute query using Python, and how to properly extract request and return data on the server side using Python. We also had a hard time preprocessing the data, as there were some format issues with the original data.

8. In our final project, we added some features in the login part. In our latest version, our application is able to check the username and password of the input. If the username does not exist or the password does not match, the login will be denied. We also provide students the chance to leave comments and see the ones from the others. The comments are sorted by the timestamp.

9. We can develop filters and advanced searching options in our search course function in the future. For example, we can filter the courses by specific section, specific A rate of the class, etc. For the advanced searching options, we can include searching by any combinations of courses, GPA ranges, professors. This will require us to write more generic and multifunctional SQL queries. We can also develop a visualization section for the GPA distributions in each semester. This will show what courses the users might want to choose when they look at the results of the searching.

10. Frontend: Dongming Liu
    DB: Michael Fadillah Wong
    Backend: Owen Xu, Xinyi Wei, Michael Fadillah Wong, Dongming Liu
    Various Report: Owen Xu, Xinyi Wei, Michael Fadillah Wong, Dongming Liu

    All teammates are on top of their task, and all teammates have their task completed when we meet.