Individual Report – Aditya

For our EECS 395 QuickTutor project, I primarily worked on backend implementation. I first initialized and set up the SQLite3 database that housed all of our information for the project. Afterwards I wrote functions to capture user input from the frontend and populate them into the various databases needed. To take a break from the programming aspect, I also designed four different logos for our group, and the final logo which was decided by a vote. Next I continued on developing update functions to get and post requests from frontend, and also cleaned up our code by merging unnecessary files or code into a cleaner overall structure. I also kept charge of group meetings and made sure that everyone came together at least once a week to go over expectations and next steps, all to make sure we would continue to hit our milestones in time. Initially we had hoped to use a Google Maps API for our location, for which I wrote code to implement that into flask using *flask-googlemaps*, but later we decided that it was unnecessary for our web service to use this. The last few things I worked on was creating the poster and abstract, writing up some frontend HTML code for the UI, and work on sections for the group report.

Since Flask was new to me, most of my periodical work was achieved through the use of online tutorials, guides, or questions answered on Stack Overflow. Of course I also went to my groupmates for help, especially Kian since he was familiar with Flask beforehand. I knew how to work with SQLite3 and Python, so those parts of the programming I knew how to do already. For logo design I just used Photoshop and took in my team’s advice on a logo design and applied it. I used my previous experience in Photoshop for this. For the rest of the frontend request forms, I worked closely with Nathan and Kian to make sure everything was working as intended. For testing I created custom use cases and populated our database, checking to see if all requests were going through. Lastly I used Facebook Messenger to schedule all meetings and communication between the group and for gathering everyone together once a week.

Most of the issues I ran into during this project were based around getting frontend code to go to the backend. I have never worked on a full-scale development project like this before, usually only on either front or backend. So this was definitely a new learning experience for me. Issues mainly arrived when multiple people working on one thing would have different ways of naming variables. It took me over a week to figure out why one form wasn’t send data back; turns out it was due to mismatched variable naming. Other issues included some query functions not working properly, flask crashing when loading the web pages, and the listings not showing up correctly. All these bugs and issues we logged in our readme files and addressed them together as a group. We also kept track of how to solve some of these bugs so that if they came up again we would know what to do.

The lessons I learned throughout the project revolve around understanding how Flask works as a library and learning the complications of developing an application for both frontend and backend. On paper some of these tasks seemed quite easy, such as setting up a chat service or sending confirmation emails. But they become increasingly more complicated with more features that were added. Along with that, having multiple people edit the same file also causes problems too. I learned a lot about developing in a group and understanding how important version control is. I also learned my way around Flask and its various extensions, and overall how an online application like QuickTutor handles data and displays information.