The ScholarCommons web app was a success and was a collaboration between our team of Surya, Afaq, Aatif, and Dhruv.

Aatif's Contribution: Designing the User Interface

Aatif played a pivotal role in designing the web app's user interface, ensuring that it was both visually appealing and highly functional. He conducted thorough research into design principles, color schemes, and typography to create a clean, modern, and intuitive interface. Aatif also worked closely with Dhruv to incorporate seamless navigation and user-friendly features such as filters, search bars, and pagination, which greatly improved the overall user experience. As the only senior in the group, he had lots of experience using Rutgers websites, and had great contributions to color schemes and layouts in particular. His experience with Front-end apps made him the go-to person for implementing any ideas that we had. He worked on the admin users, as well as a lot of the layouts that are used across the site.

Dhruv's Contribution: Website Layout and Design

Dhruv focused on the website's layout, working in collaboration with Aatif to ensure that the design was consistent, engaging, and easy to navigate. His main responsibilities included developing responsive grids, and incorporating interactive elements to the site - all while ensuring everything looks visually appealing. Dhruv's experience with front-end technologies such as HTML, CSS, and JavaScript enabled him to transform the designs into functional web pages that were accessible across various devices and screen sizes. He worked on the Customer Service pages for both the external users and the Customer Service admin users. Dhruv also contributed a lot in terms of testing the website for any functional bugs and reporting them to the team to look into.

Surya's Contribution: Adding New Features and Project Management

Surya demonstrated exceptional leadership skills in managing the overall project and keeping the team on track. He actively participated in feature planning, identifying key requirements, and prioritizing tasks to ensure that the project milestones were met on time. As someone with prior experience developing web-apps, he worked with everyone and contributed in both the frontend, as well as the back-end. Surya contributed to the web app's functionality by working on various features such as login, buy/sell, as well as the Blog. He contributed a lot to the websites’ backend core logic that the site is built on. As someone with a lot of experience with everything, he did a little bit of everything and worked with everyone else to make sure we stayed on track with time.

Afaq's Contribution: Back-End Development and Deployment

As the back-end developer, Afaq was responsible for developing some of the core functionality of the ScholarCommons web app. He worked with Surya and Dhruv to initialize the database to store and manage the users, sales, and all other relevant data. He wrote a Python script to scrape the web and find RateMyProfessor links, all the courses offered at Rutgers, as well as create GroupMe chats for all the courses. He worked on the RMP and GroupMe pages. He also worked on the deployment process on the Tomcat server.

Overall, all members contributed equally. Coming into the project, everyone had different different skills and experience - plus everyone had learning goals. We all came together and worked out what we could do so that everyone is happy and can contribute in their own way. We are satisfied with what we were able to accomplish as a team together.

In the beginning, we did not have a GitHub setup - and the way .war files work they were very difficult to set up with GitHub for us. For that reason, for most of the project we shared files over Discord and discussed everything in the groupchat. Surya was able to manage the code and what needed to be merged manually.

However, later on we got the GitHub setup and we started to use that for source code management. It still proved to be annoying with the .war files - but we were able to make it work for what we wanted. Overall since we were working on different modules and parts of the website - we did not have many issues merging and that allowed us to focus on our tasks instead of on resolving conflicts. Not all team members were able to set up GitHub due to issues that we were running into - so they stuck to the sharing through Discord method where the code would eventually be merged to GitHub. It was not ideal - but it worked for the group and it allowed us to focus on the code and implementation rather than getting sidetracked.