**Team Members**
Project Title: Disharmony
Members:

- Maaz Hassan - 30087059
- Dawood Afzaal - 30063067
- Hasan Nassar - 30099862
- Hassan Sohail - 30092431

**Schedule:**

Our initial schedule for the project was to begin with the UI design and mockups, which we would then follow up by implementing the design using React/TypeScript and Tailwind CSS. The next phase after this would be to work on creating and managing the database using MongoDB, and defining a WebSocket messaging protocol. After this we would move on to work more heavily on the backend of our application, while concurrently building out the UI as backend features were completed. The final phase would then be to link the backend of the application to the front end, build and link the mobile version of the UI, and do the finishing polish. As we began the project, we stayed on course with our schedule up until the backend creation - the UI creation was going smoothly up until this point. However, once we started the backend, we realized we severely underestimated the complexity of the backend. Our entire messaging protocol had to be updated, as we realized we would need a lot more API endpoints to implement all the features for our application while keeping everything real-time. Overall the backend was very time consuming, so instead of staying on schedule, the group decided we had to carve out more time to work on it and push back the rest of the UI implementation, mobile implementation, and polish stages. This was a struggle with the already limited development time, and some more experienced members had to work much more in order to meet the deadline.

**GUI Design Evolution:**

Overall the design of our graphical user interface was extremely similar to the proposed design mockups. We were able to implement the design nearly identical to what we had planned. The only changes we had in the design were minor changes in the button placements and polishing of some of the pages and modals. We added these minor changes to make the design of the interface look better and easily accessible. We made sure to change some of the button placements for adding friends and text channels in the mobile version as we wanted this version to have high ease of browsing for the user.

**Unexpected Challenges:**

The main unexpected challenge we faced was underestimating the complexity of the backend. Initially, we didn’t think our backend would be as complex as it eventually turned out to be, and we ended up having to add a lot more endpoints than were initially planned. This significantly increased the amount of development time and stress on the group. As mentioned before, we overcame this challenge by rescheduling our initial timeline, and devoting more time to developing the backend, utilizing our more experienced members.

Another unexpected challenge we faced was working together as a group at times. This was mainly due to two reasons that we had not foreseen. The first reason was the major differences in opinions that each of the group members had with the project. Initially, everyone was on the same page about what the project will be and the general outlook of it. However, once we began working on the project itself,  as we were working on the UI design and implementation, each of the group members was having different opinions and inputs on how the design should look. Especially with the smaller details of the design, each member had a different idea to incorporate into the project. This made progressing with the design much more difficult than expected as it was taking much longer to complete the design of the interface than expected. We were able to overcome this issue during our group meetings by having the members reach a compromise in which pieces of each of the members' ideas were incorporated into the design. The second issue we faced was finding the time to work on the project together as a group. Initially we figured that the group would be able to meet together at least a couple times a week to discuss and work on the project. However, what we didn't expect was how difficult it would be to have these gatherings, due to each member's schedule being different and it ended up being a challenge to find a time each for the whole group to meet together. We decided to address this issue by splitting off into smaller groups, having the group members with similar schedules and class timings meet together and also do some pair programming/code reviews. This resulted in fewer meetings with the whole group as a whole, however even with the smaller groups we were still able to progress strongly with the project. Following this, when the group would then meet up as a whole, we would fill each other in on the ideas that the smaller groups had been working on.

**Knowledge Gaps:**

The main issue we had was working with TypeScript. A few of the group members  had some experience working with TypeScript, but the group as a majority didn't have much knowledge of the technology. This made developing the project more difficult and time consuming as we were learning a new technology while we were working on the project. We knew we wanted to use TypeScript as our project was likely to be quite large, and using TypeScript is generally considered a good principle when working on a large group project. As we were reflecting on this, the group as a whole really wished that we had more prior experience with TypeScript as the whole project would have gone a lot smoother. The only solution we could find to address this issue was for each of the group members to first devote some time to learning the technology by using the internet. Second, the group made sure to try to work together as much as possible, especially with the parts using TypeScript so that we could learn together and also provide guidance for each other.

Another issue the group faced was working with WebSockets.. A majority of the group had some experience in working with websockets, however lacked enough experience to implement a full solution using them. This is mainly due to the group not having done much work with WebSockets and because some of the group had not used Python in a long time either.. To address this issue, similar to our solution with TypeScript, we spent as much time as possible relearning Python fundamentals and WebSockets, and tried to work together whenever we got the chance so that we could help each other out. This was easier to do than the previous issue as with this we were mostly just relearning material that we had worked with previously, and so as we kept practicing and coding the knowledge and skills were coming back to us, as opposed to learning something completely new with TypeScript.

**Success and Shortcomings:**

Something that we noticed worked well during the project was setting a deadline for each part of our project. Every single component of the project that we wanted to implement during the project, we set a tentative deadline and would try our best to meet that deadline. This worked well for us as this helped to keep the group on track. With these deadlines we were able to consistently keep on track with our work, and everytime we missed a deadline we would set a new deadline and try to meet it.

We also used a Trello board to create and assign tasks to ourselves, as a way to keep track of who was doing what. This really helped us stay organized with our tasks, and was a valuable collaborative tool.

Something that we also noticed worked well, which was mentioned in one of the previous sections, was having the group split off into smaller groups so that there would be better communication between the members. This also helped tremendously because we were also able to pair up the members with similar skill sets together, allowing them to work well with each other and make tremendous progress on the project in the form of pair programming.

Something that we noticed that didn’t work out too well was our initial planning of the project. As time went on, we realized that our initial plan for the project, including the coding aspects and the communications aspects were not planned out as accordingly as we needed them to be. This shortcoming made it difficult to progress throughout the project and the group had to work together to address this issue.

Another shortcoming that we noticed was that the scope of the project initially expanded beyond our control. There were initially components that we wanted to implement into the project that we figured would be possible, but as we were working on the project we realized they were difficult to implement and it was becoming time consuming to figure them out. Examples of this are the Giphy feature, which would allow the user to integrate with the Giphy API to send gifs in the chat, as well as the Google Maps feature, which would allow the user to send a map of their location in the chat. We included these in our initial proposal of the project but as we were working on implementing the rest of the project, we realized we were running out of time and had to scratch them. We believe that if we had better managed the scope of the project, and had more development time in general, the development process would have been much smoother.

**Overall Project Rating:**

On a scale of 1-10, we decided to give our project a rating of 7. The main reason for this is because we realize that our project was not as unique as required. The application that we created is similar to other chatting platforms, and we believe that this hinders the rating of our project. We believe that we should have created a project idea that would be more unique, however we still believe that the chatting platform we decided to go with is a great technical project to work on as it still requires many difficult and unique technical challenges to be overcome, such as figuring out the real-time communication aspect. Another reason why we don't rate our project higher is because we weren't able to implement all of the features that we had initially planned to make our project unique, like the Giphy integration and Google Maps integration. As mentioned before too, there were many struggles during the process as well that hindered our ability to make the project as great as we planned initially. However, all critique aside, we still believe that we were able to make a great project, hence the rating of 7. We were able to create a fully functional chatting platform that can be used to communicate with people across the internet. We believe we created a sleek and functional interface that is intuitive for users. We were also able to learn many things during this project as we had to work with technologies that were completely new to many of us, and also had to relearn many aspects from previous work. This made the process of working on the project very beneficial for the group as there were several learning outcomes. This is why we believe a rating of 7 is fair for the project that we have created.