

CS473 Final Report - TUG

URL for Video

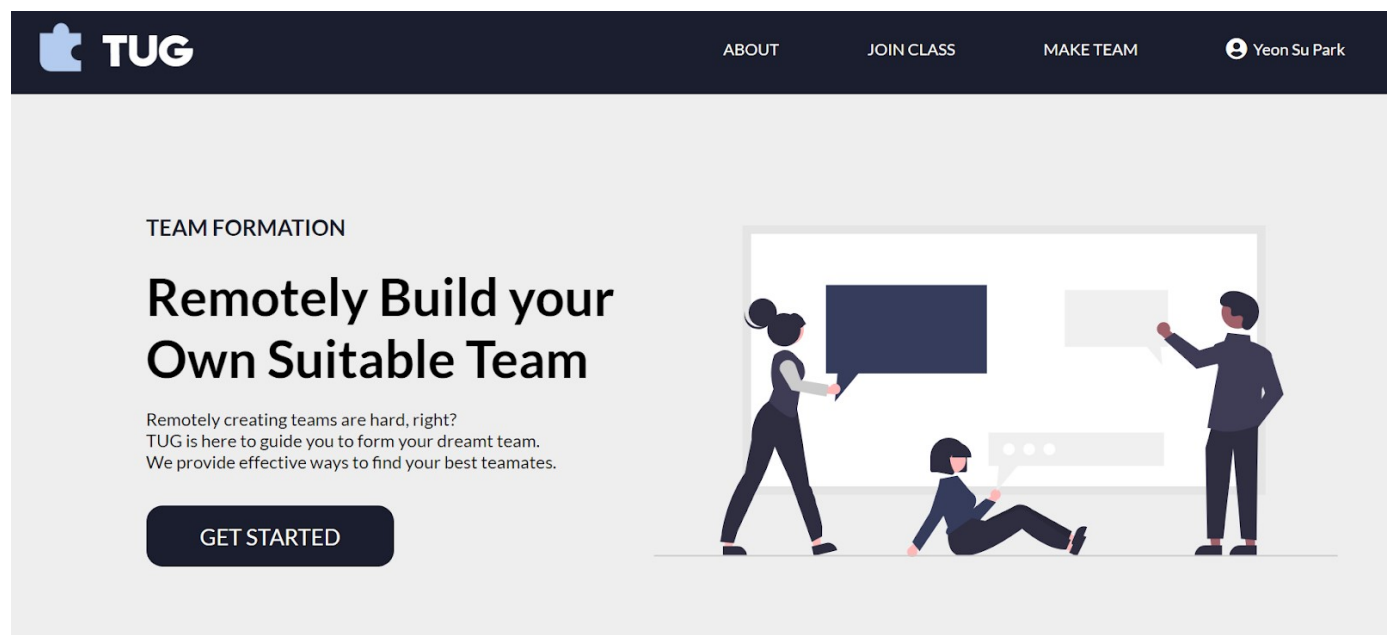
<https://youtu.be/6-1Vq-p6qCI> (<https://youtu.be/6-1Vq-p6qCI>)

Subtitles were created using YouTube subtitle feature. Please turn it on when watching our video.

Representative Screenshots

Main Page

User can sign in to the website and start remotely building their team by clicking the main "GET STARTED" button.



Profile Setting

User need to set the 3 tags to show the ability and share the fun facts about themselves, respectively.

HOMEJOIN CLASSMAKE TEAM Yeon Su Park

Create Your Profile

Write **3 Tags** to Show Your Abilities! Be Creative!

Add Your Ability Tags

react_master frontend_designer firebase

Write **3 Tags** to Share Interesting Facts about You!

Add Your Interesting Tags

cat_lover violinist horror_movies<3

Submit

Essential Quiz session

At the first round of quiz, users would need to answer to the questions which are important to build a team.

QUIZ TIME!

It's Quiz Time!

Q1. Are you familiar with the presentation?

Check if this question is important to you!
☐

Yes

☐

☐

☐

☐

☐

No

Q2. Which communication style do you prefer?

Check if this question is important to you!
☐

Fun Quiz session

From the second round to the final round, users would need to answer to the fun quizzes. The result is reflected in the randomness parameter of team building algorithm.

 QUIZ TIME!

It's Quiz Time!

Q1. Do you like pineapple pizza?

Check if this question is important to you! ☐

Never

Maybe

Absolutely

Q2. Do you like mint-chocolate?

Check if this question is important to you! ☐

Never

Maybe

Absolutely

Chat session & past credits

After the quiz session, users are divided into the potential team and chat with the potential teammates. They can check the profiles and credit history of other anonymous teammates.

 CHATTING TIME!

Moderator

Hi, Mare, Tiger, Snake, and Nyancat! Ready to have some chat?

Chat with your potential group members. Click each chat bubble to see profiles.

Addax

hello

Moderator

You have three minutes before voting!

One minute left! Share your last comments to people.

Ten seconds left! Get ready to vote!

♦♦ 3 minutes 20 seconds left!

Get to know your teammates!

Addax

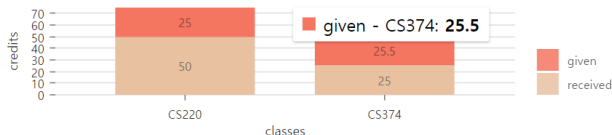
Skill-Sets

html # css # javascript

Interesting Facts

game lover # computer mania # fast worker

Credit History



Class	given	received
CS220	25	50
CS374	25.5	25

Voting page

Users would need to vote after the chat session is over whether to finalize with this team.



CHATTING TIME!

Moderator

Hi, Mare, Tiger, Snake, and Nyancat! Ready to have some chat?

Chat with your potential group members. Click each chat bubble to see profiles.

You have three minutes before voting!

One minute left! Share your last comments to people.

Ten seconds left! Get ready to vote!

Finished! Please wait for votes...

Discussion Time has Passed!

Do you want to make team with current members?

Yes

Try Again (2/2)

Please wait until other teammates to vote! 😊

Credit system on the past project

When the project is done, each user gives the credit to the past teammates. This is reflected to the credit history and shown in the future team building process.



HOME

JOIN CLASS

MAKE TEAM

 Cheryl Siy

Credit

Submitting honest reviews will help you and other classmates in selecting a team mate with a good fit.

Remaining points to distribute: 24

Uxer Ham

Yeon Su Park

Juan Mail

submit

Profile page of the user

Users can check their profile and credits.

HOMEJOIN CLASSMAKE TEAM9 Yeon Su Park

YOUR CLASS

ACTIVE TEAM

PAST TEAM

PROFILE



Yeon Su Park
Email: yspictures16@gmail.com
Credit: 300
You are level 4 Nubjuki!

**Level 1**
Newcomer

**Level 2**
Hard-Worker

**Level 3**
Average Worker

**Level 4**
Proficient Worker

**Level 5**
Teamwork Master

Quality Arguments

Fun Quiz and Tags helps in Starting team building

"The idea of conducting a survey(quiz) before chatting was good. I was able to organize what I wanted from my team members and expect similar people to match." -P1

"Of course, it would be more convenient for the assistant to make it(teams) randomly, but I think it was good that I had a choice(in accepting members)." -P2

QUIZ TIME!

It's Quiz Time!

Q1. How comfortable are you with your English skills?

Check if this question is important to you! ☐

Not Confident ☐ ☐ ☐ ☒ ☐ Very Fluent

Q2. What teammate do you prefer? An enthusiastic person or a relaxed person?

Check if this question is important to you! ☒

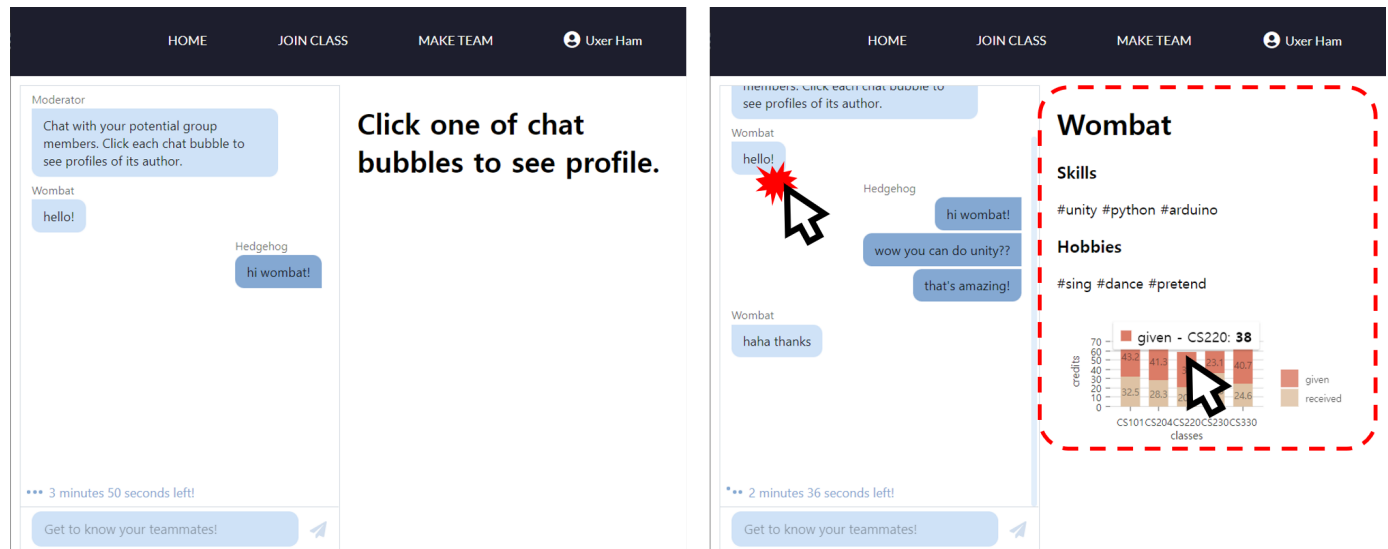
Very Enthusiastic ☐ ☒ ☐ ☐ ☐ Very Relaxed

It's difficult to meet new people every semester. Remote work has stopped people from creating new bonds and has made people more siloed [1].

To kickstart the team building session, we designed a fun quiz to group people into predetermined teams. The quiz asks users' preference in teammates such as language compatibility, personality likeness, and even random questions which can be used as a conversation starter later on.

Anonymous Teambuilding Chatting Sessions helps in increasing interaction

"Anonymity, information about people, and being able to advertise myself more easily were better with the platform. " -P3



Anonymous Identities

Anonymity allows users to freely chat without risk of knowing their identity. A study by [2] has proven that anonymous chatting has increased participation in an educational setting. We seek to use the same method in increasing student interaction and reduce the social pressure in accepting or rejecting a team during the team building session.

Tags

User input tags are shown during chatting sessions. This helps people to find teammates that will match the team's needs while allowing people to find a common ground at a glance.

Credit History

Credit history of members are shown during the chatting session. This will give users further information in assessing teammates that are diligent workers and free riders.

Moderator Bots

Discussions are more productive and efficient in an online chat discussion when a predefined discussion structure is established. Similar to SolutionChat [3], a real-time moderator support chat bot is used in the team-making session interface to provide structure. Our moderator chatbot lets users know how many people are in the room, how much time is left, and suggest topics users can discuss.

Credit System helps build up trust and accountability

"When I used this system, it was very nice to know a little more about other people. I thought I could trust my team members a little more because I could see what kind of experience they have now and what they have shown in other projects. Of course, you have to do the project yourself to know the exact details, but I think it will help build trust that you can get more information when you make a team." -P3

To reduce free riders, a credit system is implemented in the system. After finishing a project, each can distribute 100 points to rate each teammate's contribution. This scheme is similar to a crowd research study to ensure a fair evaluation in crowd based work [4]. Credits will be displayed on each person's profile during chatting sessions for other users to observe and decide on their reliability.

[1]Yang, Longqi, et al. "The effects of remote work on collaboration among information workers." *Nature human behaviour* (2021): 1-12.

[2]J. Yang and C. Choi, "Enhancing interactiveness of problem-based programming course using anonymous chatting service," *2017 16th International Conference on Information Technology Based Higher Education and Training (ITHET)*, 2017, pp. 1-6, doi: 10.1109/ITHET.2017.8067828.

[3]Sung-Chul Lee, Jaeyoon Song, Eun-Young Ko, Seongho Park, Jihee Kim, and Juho Kim. 2020. *SolutionChat: Real-time Moderator Support for Chat-based Structured Discussion*. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. DOI:<https://doi.org/10.1145/3313831.3376609> (<https://doi.org/10.1145/3313831.3376609>)

[4]Rajan Vaish, Snehal Kumar (Neil) S. Gaikwad, Geza Kovacs, Andreas Veit, Ranjay Krishna, Imanol Arrieta Ibarra, Camelia Simoiu, Michael Wilber, Serge Belongie, Sharad Goel, James Davis, and Michael S. Bernstein. 2017. *Crowd Research: Open and Scalable University Laboratories*. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. Association for Computing Machinery, New York, NY, USA, 829–843. DOI:<https://doi.org/10.1145/3126594.3126648> (<https://doi.org/10.1145/3126594.3126648>)

Evaluation

Since the evaluated results were too long, we decided to move the results into the discussion section.

Research Goals

We set our research goals as follows:

- RQ. 1 - help students to meet new people while forming teams
- RQ. 2 - ensure the reliability of individual users
- RQ. 3 - minimize social pressure of students forming groups online

Study Design

We recruited 4 undergraduate students (average age 22.0, 3 males, 1 female) from a local university. Every subjects had experience in online team formation for a class. No subject received monetary rewards for participation.

We collected prior experiences in online team-formation from every subjects and set those as our baseline. We measured qualitative and quantitative data of the baseline before subjects use our system to minimize contamination when reporting their experience.

Procedure

The experiment is conducted in a video conference. Each participant joined a Zoom session and received a link to online forms. The participant first reported their prior experience of online team formation for classes. Then, the participant filled a Likert scale 7 point questionnaire to measure their baseline experience. After finishing it, the participant is guided to sign into TUG, join a designated class, and make a team. We designed our experiment as a controlled Wizard of Oz experiment. Each team has a participant and three wizard experimenters; chat is fully anonymized and participants did not know about this. Experimenters used predefined skill tags and coversation algorithms to control their dialogues; experimenters are allowed to answer participants if there is any questions but not allowed to write any creative conversations. After finishing the chat session, the participant was guided to visit a link of post-experiment online forms. The participant then finished the same Likert scale survey they did for the baseline, and to compare experiences of TUG with those of the baseline. The participant was able to ask questions while completing forms; but required to stop sharing their screen while filling both Likert scale questionnaires.

Discussion

Survey

Prior Online Team-Building Experience

Subject had used diverse online platforms to make teams online; real-time collaborative documents (N=2), text-based communication platform (N=2), and real-time video conference (N=1). Most subjects obtained informations of other teammates in texts (N=3);

real-time image (N=1) and video (N=1) were rarely observed.

Every subjects (N=4) mentioned that online teams would be more prone to free riders than offline teams. Half of subjects said they would feel uncomfortable to reject teams online (N=2; N=3 when forming teams in a video conference). Interestingly, most subejcts felt unsatisfied when finalizing their online teams (N=3), even though most subjects said they made teams with people they already knew (N=3). By observing these answers, we concluded that our research goals to help (1) team formation with new people, (2) reliability of individual users, and (3) minimizing social pressure are well defined.

TUG Experience

After subjects using TUG, we asked them reasons for accepting or rejecting their teams. Every subjects (N=4) mentioned what they observed from chat dialogues. Some subjects mentioned credit history (N=3), and skill tags (N=2). We concluded that our design provided subjects useful information to validate their teams.

Qualitative Feedback

We asked subjects to compare TUG with their prior experience (baseline) in respect to (1) reliability of teammates, (2) convinence, and (3) overall satisfaction.

Every subjects (N=4) said they felt higher reliability of teammates. For their reasons, they mentioned credit history (N=3), skill tags (N=2), and getting know each other with chats (N=2).

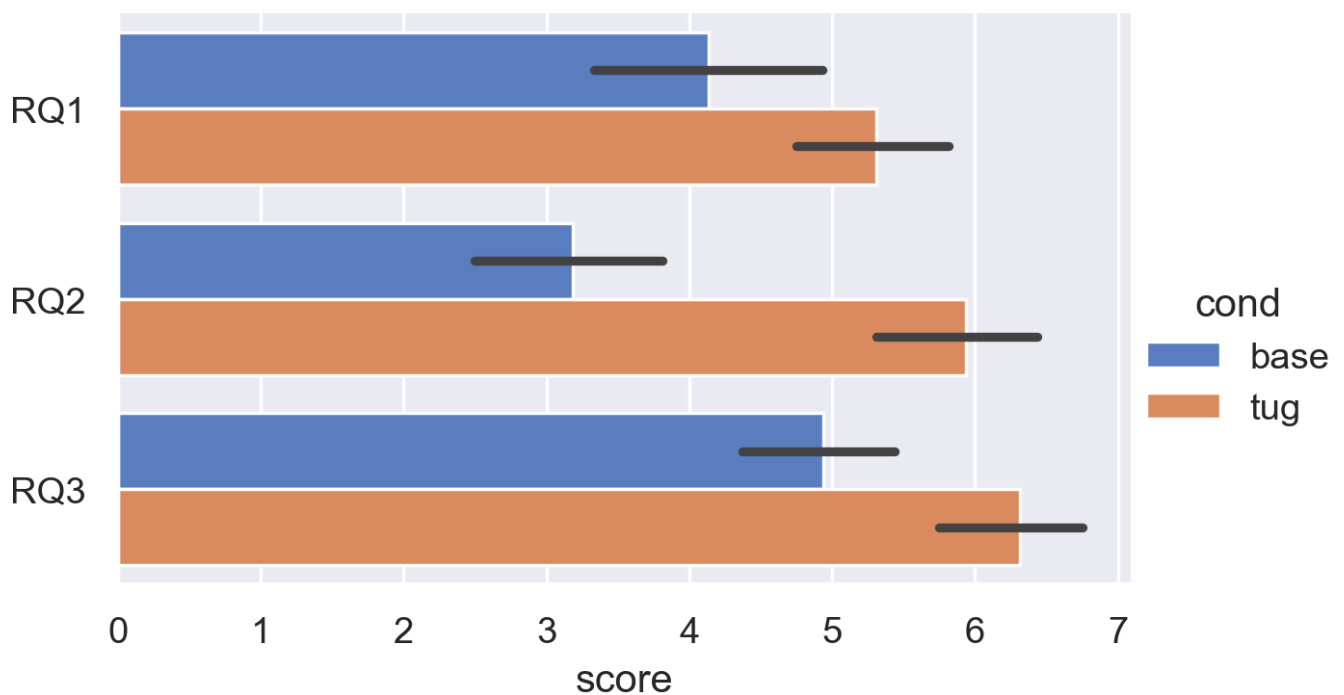
In case of convinence, every subjects (N=4) preferred TUG over the baseline. Interestingly, P1 mentioend conveninece of our system of "automatically matching with desired participants," while P2 mentioned conveninece of "having an option to accept (the team)." Some subjects mentioned discomfort from requiring more time (N=2), or requiring active conversation (N=2).

In case of overall satisfaction, every subjects (N=4) preferred TUG over the baseline. P1 said our system made team-building easy because it "automatically match people with similar tendencies." P3 even mentioned that anonymous system of TUG had advantage compared to offline team building sessions.

Questionnaire

We designed a 7-scale Likert questionnaires and compared the scores of TUG and prior experiences of subjects in respect to our research goals.

RQs	Questions
RQ1	i feel team-building process easy to follow
	i feel i successfully advertised myself to others
	i feel comfortable introducing myself to others
	i feel attached to the team
RQ2	i feel i gained enough amount of informations about teammates
	i feel i know what to expect from other teammates
	i feel i can plan team's goal
	i feel i can distribute tasks to myself and others
RQ3	i feel i can easily turn down (reject) teams if i want to
	i feel my opinions mattered as much as others while forming a team
	i feel i would turn down (reject) teams if i find someone as a free rider
	i feel the team will not be finalized without my consent



Scores for RQ1 were 4.13 (sd=1.68) for baseline, and 5.31 (sd=1.20) for TUG. Scores for RQ2 questions were 3.19 (sd=1.42) for baseline, and 5.94 (sd=1.12) for TUG. Scores for RQ3 questions were 4.94 (sd=1.12) for baseline, and 6.31 (sd=1.07) for TUG. Wilcoxon signed rank test showed significant difference between baseline and TUG for every RQ groups; RQ1

($Z=10.5$, $p<0.01$), RQ2 ($Z=3.5$, $p<0.05$), and RQ3 ($Z=8.5$, $p=0.05$). We concluded that our system outperforms in regards of every research questions compared to conventional ways of online team formation.

Individual Reflection

Seonghye Cho

What part of the implementation did you contribute to?

First, I **proposed a specific flow of our service** on the basic concept and HMW solutions: In the first round, the page asks about the information needed for team building, and from the next round, it suggests fun quizzes that people can easily enjoy the team building process. Also, I applied the importance check button concept in the quiz session.

Second, I **implemented the functional parts** of the quiz session (quiz_information page, random_quiz.jsx and question_candidates.jsx files), room dividing algorithm(kmeans.jsx, team_building_algorithm.jsx), voting component(voting.jsx), and several other pages(waitingjoin page, activeteams component).

What worked well and not in your team? How did you overcome any hurdle in teamwork? What lesson about teamwork did you learn that you might apply to your next team project?

Communication was smooth and positive in the overall teamwork process. Chatted actively and frequently in kakao chat. If they had questions about each other, they asked or answered without hesitation, and thanks to those active chatting, we could finish our work positively and comfortably. All the team members did not wait for others to do it first, but they came forward first to schedule and helped others easily join (ex. making a zoom link or google docs file).

It was **not easy to test** the whole system alone because our project required at least four people to access at the same time. We could test only up to eight people, and the test in large number of 20 or more could not be conducted. It would be nice to create a demo code for the test or the automatic testing process. In our case, we usually had to wait three minutes to enter the quiz session, but it was hard to wait 3 minutes for every test. I made a demo code to set the page wait only 15 seconds and Auejin also decreased the chatting time to 1min 30secs.

I learned some **useful methods** to effectively and meticulously proceed the work. This had a decisive influence on communication. In particular, Google Docs and Kakao Talk notification functions were frequently used. We collected the materials what we have done in Google Drive, so we could refer to those previous works easily when doing the next project. Through Google Docs' check box function, we could easily check whether we accomplished each grading criterion, and how far we did and how much work left.

Through the team-based design project experience, what did you learn about social computing and web-based GUI implementation?

'Social computing' continued to remind me of **focusing on people**, not the computer algorithm itself. I thought that if we made a team making algorithm well, wouldn't it be okay to subtract the chatting session between people? However, looking back under the theme of social computing, it was the chatting session or credit system between team members that complemented the algorithm that had limitations in the end.

I learned how to cope with Web browser's asynchronous processing method. Web browser asynchronously rendered javascript code for faster service, resulting in a problem that the component was shown as a null value before it loaded data from DB. I learned the **'useState'** method which makes 'state' available in function components, so that the component is initially loaded as 'loading', and then automatically updated with the data from DB.

Cheryl Siy

What part of the implementation did you contribute to?

I had the task of creating the "Credit" page wherein we ask users to distribute points to their teammates according to their contribution. The task can be divided into 3 main parts, the Dynamic Form, UI computation, and Storing and handling data. The Dynamic Form takes in the current teammates data from the database, where the credit, name, and id are stored. It then gets processed, formatted, and passed on to display in the UI dynamically.

Once users input data, a dynamic text is displayed on top to show how many points are remaining for distribution. The value is then computed and displayed real-time in the UI to let users know the remaining credits they have to distribute. This also accounts for error handling such that if the distributed credits are greater than 100, the form wouldn't submit.

Once the form is submitted, the database in Firebase is updated. There are 2 items being stored and handled - the users' inputs and the teammates' credits which are stored in each of the user's data. User's average credits are also computed every time someone submits a credit rating. This is to ensure that the database is always updated. The current teams' list is then effectively moved to the past teams' list.

What worked well and not in your team? How did you overcome any hurdle in teamwork? What lesson about teamwork did you learn that you might apply to your next team project?

Establishing a predetermined structure right from the start worked in our remote team. We have a scrummaster who is in charge of moderating the discussions and tasks. Our discussions were efficient and straight to the point, which made working together easier. Meeting regularly to catch up and share problems also helped in solving hurdles that team members are experiencing.

Having a mix of graduate and undergraduate students in the team worked well for our team. We have diverse backgrounds and expertise which helped in brainstorming ideas and creating a valuable product.

The only difficulty I can think of is finding a common time slot to meet. We have a mix of graduate and undergraduate students so the schedule is highly varying. However, we still manage to meet regularly by using when2meet and usually scheduling meetings late at night.

Through the team-based design project experience, what did you learn about social computing and web-based GUI implementation?

Through the team-based design project experience, I have learned that it takes a lot of brainstorming and time to create a social computing and web-based GUI implementation. Also, users should be the main focus in designing the social computing interfaces.

It's my first time to work heavily in ReactJS and Firebase so I learned a lot about the integration of the front and back end implementation.

The format in the class is like a hackathon that pushes everyone's limit to learn and deliver fast. From the start, the team has little knowledge about web-based GUI implementation. In just a matter of weeks, I am amazed at how the team has pushed through and delivered a working prototype of the TUG platform.

Yeon Su Park

What part of the implementation did you contribute to?

I mostly contributed to the aggregation of people's work and producing a platform with unified design of UI. I have designed the main color scheme, fonts, buttons and logos to enhance readability of our interface. I often worked by getting raw functions from my teammates, and build a UI based on the functions. Adding on to this, I worked to aggregate other members' work and functions, considering our end-to-end scenario. I connected different functions together, tested them and fixed errors if necessary.

I have also made contributions to various functional parts of our website. One of them is the login function. I connected the authentication function of the firebase to our interface so that when the user logs in via their google account. I also connected Firebase to our website, so that we can collect user's data into our database.

Profile page and the tagging system was another part I worked on. Especially for the profile page, I have designed a level system (gamification) so that the user's avatar changes according to the credits they have earned. I have also worked on various dynamic features like joining classes, listing enrolled classes, etcetera.

What worked well and not in your team? How did you overcome any hurdle in teamwork? What lesson about teamwork did you learn that you might apply to your next team

project?

Through the team-based design project experience, what did you learn about social computing and web-based GUI implementation?

Auejin Ham

What part of the implementation did you contribute to?

I contributed to the implementation in two major ways; (1) I created real-time chat rooms, and (2) converged data structures made by other teammates to visualize into tags and plots. For the real-time chat rooms, I had to make sure that messages, chat room timers, and anonymous names by uids are updated in real time and safely separated for different rooms and rounds. Also since this page is what users are going to gather information for to bet over their ideal teams, I designed some social-computing elements (e.g., a moderator bot to promote engagement and recommend conversation topics, bar plots to compare each user's contribution to their past teammates) that can promote more not only temporarily denser communications but also wiser decisions for each user.

What worked well and not in your team? How did you overcome any hurdle in teamwork? What lesson about teamwork did you learn that you might apply to your next team project?

Each of our members could learn web development skills and successfully collaborate to implement our prototype. However, we often struggled in communication since not all of us had the same first language. Subsequently, bandwidth for real-time communication was limited since all of us were not able to meet offline. Therefore, as a scrum master, I tried to make meeting session as frequent as possible. We met at least once a week via Zoom even if there is no urgent task to finish. While preparing for the DP milestones, we met once or twice a day. For each Zoom call, I also guided our team to list up microtasks to do until next meeting. I learned that frequent feedback between teammates could quickly resolve language barriers, and even prevent potential bugs that could be occurred due to miscommunication. Subsequently, listing up tasks can not only make us keep track of the milestones, but also promote motivations to continue the long journey of CS473.

Through the team-based design project experience, what did you learn about social computing and web-based GUI implementation?

This is my first time to develop a dynamic web service and I had to learn numerous techniques and tools to maintain the correctness of both database and rendered pages for real-time interaction of multiple users. What I find most interesting is about how easy to use modules made by others. While implementing my parts, I used Chatscope for a chat room interface, and Nivo for credit score visualization. Without the strong ecosystem of React, I would not be able to implement all of these. Finally, while designing the user study, I was able to obtain some useful skills to read Google Forms data into Pandas.