CEN 4010 Principles of Software Engineering, Summer 2021

Team 1, Gopher

Team #1

Summer Poissonnier: Team Lead/Development Team [spoissonnier2019@fau.edu](mailto:spoissonnier2019@fau.edu)

Ania Schulz: Scrum Master [schulza2019@fau.edu](mailto:schulza2019@fau.edu)

Joshua Walsworth: Product Owner [jwalsworth2018@fau.edu](mailto:jwalsworth2018@fau.edu)

Rodrigue Bichotte: Development Team [rbichotte2018@fau.edu](mailto:rbichotte2018@fau.edu)

Carlos Alvarado: Development Team [calvarado2020@fau.edu](mailto:calvarado2020@fau.edu)

July 27, 2021

<https://lamp.cse.fau.edu/~cen4010_su21_g01/Gopher/>

Final demo: <https://www.youtube.com/watch?v=J6e7cejWmbQ>

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Revision** |
| 1 | 6/14/2021 | N/A |
| 2 | 7/22/2021 | We took out the functions: Friends list and Suggested Friends list. |
| 3 | 7/29/2021 | Updates were made to the UML Diagrams and Product summary as specified by the instructor. |

**Milestone 5: Final Project Delivery and Demonstration**

**Project Summary**

1. Gopher is a social website aimed at teens and adults ages 16+.
2. Gopher has many interesting functions that allow users to interact:
   1. You will be able to view and post comments on the main forum of the site
   2. You will be able to view and post messages within a discord link to a gopher discord.
   3. You will be able to create and edit your profile on Gopher
3. There are so many similar sites out there that there are no functions Gopher has that are entirely unique. However, there is no site that has the exact combination of functions that Gopher does. Gopher has a forum where users can chat, a chatroom which links to a discord server, and user settings/account information. Gopher is the perfect site for those wanting to remain anonymous but also wanting to interact with others online.
4. <https://lamp.cse.fau.edu/~cen4010_su21_g01/Gopher/>

**Milestone 3 Documents:**

**Vertical Demo:** <https://lamp.cse.fau.edu/~cen4010_su21_g01/vertical_demo/>

**YouTube Link:** <https://youtu.be/v8R9stt1TAs>

1. Executive Summary

In essence, this is a website, called Gopher, which connects people based on their common interests. The key advantages of Gopher are: upon signing up, you will be able to choose your interests. Then, the website will offer suggestions for people you might be interested in meeting based on your interests. When it recommends a person, it shows that person’s profile image, name, age, contact information (e.g., Twitter, Instagram, Facebook, phone #), location, and a short bio.

When a user creates an account, the website will require that he or she creates a username and password, which he or she will use when signing in to the website. This is important because we do not want our users to be concerned about someone logging into their account without their knowledge/consent.

Once you have signed into your account, you can either view and edit your own profile or view suggested connections. This is an advantage Gopher provides because users are able to connect with people based on their common interests.

1. Competitive Analysis

|  |  |
| --- | --- |
| **App Name** | **Features** |
| Gopher | * Sign up (username & password) * Fill form with list of popular interest * Suggest people * Ability to have a short bio, keep link of social media, contact information, name, age and location (you can only see those from people on suggested list) * Keeps track of suggested people |
| Unblnd | * Connect with like-minded people (based on hobbies, interests) * Anonymous * Smart group matching * Automatically suggest groups * Integrated chat * Based on location |
| Meetup | * Groups people with similar interests * Join scheduled events * You can make plans yourself to go to events etc. * Integrated messenger * Advance analytics * Branding (allows you to create logo, put links etc.) |
| We3 | * Connects you with like-minded people * Private profiles (only your tribe can see you) * Groups of only 3 * Allows you to track/manage your mutual interest/traits * All 3 people in the group must be the same gender * Integrated chat |
| Tinder | * Login using Facebook or create an account * Create a profile with information about yourself * Swipe if you don't like person and click heart if you do * Meet with users and get to know them |
| Bumblebee | * Edit your profile, including your advanced filters. * Add new photos. * Verify your profile. * Message your matches. * Backtrack on accidental left swipes. * View your Beeline. * Super Swipe folks you're interested in. * Subscribe to Bumble Boost. |

Gopher is similar to those apps listed above in many ways. All those apps match you to people with similar interests. Gopher has the same purpose but it differs in the sense that it keeps track of past suggestions and if it runs out of current suggestions it will provide you a list of past suggestions. Gopher also allows you to get in touch with as many people as you want as long as you have similar interests.

1. Data Definition

* **Gopher:** the name of the software product.
* **HTML:** Hypertext Markup Language which provides a web designer the ability to tell the web browser what to do.
* **CSS:** Cascading Style Sheets which go hand in hand with HTML as a set of rules that determine the visual appearance of the webpage.
* **JavaScript:** A language allowing users to create interactive web pages.
* **Responsive Design:** A design that allows the website to adapt to the user’s device.
* **Frontend:** The client-side of the website such as what the user sees and interacts with.
* **Backend:** The server-side of development where everything behind the scenes goes on. This powers what happens on the front-end. Main components of the backend are: the server, the database, the software, and the operating system.
* **MySQL:** An open-source database management system.
* **SQL Server:** A software that is a database management system allowing the storage and retrieval of data.
* **Database:** A set of data stored in a computer.
* **API:** Application Programming Interface that enables two programs to interact and communicate with each other.
* **Bootstrap:** A free open-source framework for users to download for web design.
* **Git:** A version control system allowing users to store and edit their code.
* **GitHub:** A cloud interface for git. GitHub is a host for source code.
* **UI Design:** User Interface Design, allowing screens and interactions to make up the website or app.
* **PHP:** Hypertext Preprocessor, a server-side scripting language.
  + phpMyAdmin: a free tool allowing the administration of MySQL on the web.

1. Overview, scenario, and use cases

The use scenario of Gopher is very simple: a person (who has a basic understanding of how a website works) would like to expand his or her social network and find other people with similar interests. So, that user creates his or her account. To do this, the user must create an account with a username and a password, their date of birth, their name, and their gender. Once that has been done, the user can fill in the extra details of his or her profile. The required fields are name, age, location (zip code), bio, picture, and interests. To fill out the interests’ field, the user picks at least 3 from a large list of interests. These are general categories that are fairly common interests such as music, movies, food, sports, video games, etc. In the bio, the user can describe his or herself for any connections to see. In the contact info field, the user may put a link to different social media platforms such as: Instagram, Twitter, Facebook, Snapchat, etc. as well as other methods of communication such as their phone number and their email address so that when someone else connects with him/her, they can go on a platform they have in common and chat.

Now that the user has created a profile, he or she can begin connecting with other users. There will be a button to connect with someone, which when pressed will lead to the profile of another user. All of the discovered user’s account information (except the password obviously) will be displayed, so the user will learn all about that person. If the user would like to begin communicating with the other user, he or she must only click one of the social links (or the email address) and it will lead directly to the other user’s profile on that social media platform.

The main page of Gopher will be a thread with the topic of the day that users can comment on. If users want to discuss more detailed topics such as their interests they can click on ‘chat’ in the tab bar or the ‘interests’ thread’ on the main page that will allow them to choose which interests chat room they would like to enter.

There will be a friends and suggested friends tab that the user will be able to click on to find new connections or previous connections. There will also be a link the user can select to view/edit his or her own profile at any time.

1. High-Functional Level Requirements

1. Basic account and password security/encryption: 1

2. Website should be able to sort through people and match the user up with others based on interests (a filtering algorithm): 1

3. 1 to 1 chatting, and a “public forum” styled homepage: 2

4. Users must be able to alter their interests, profile picture, biography and be able to link to their social media profiles if so desired: 1

5. The website must be able to recommend people the user has previously chatted with as friends: 2

6. Website must be able to pick a “topic of the week” by means of linking to another site

in order for users to discuss said topic: 2

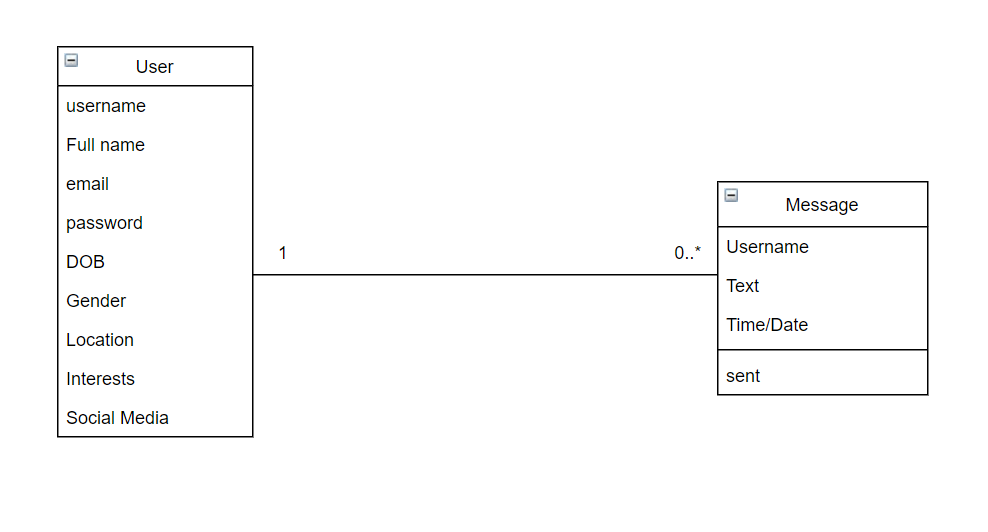
1. List of non-functional requirements

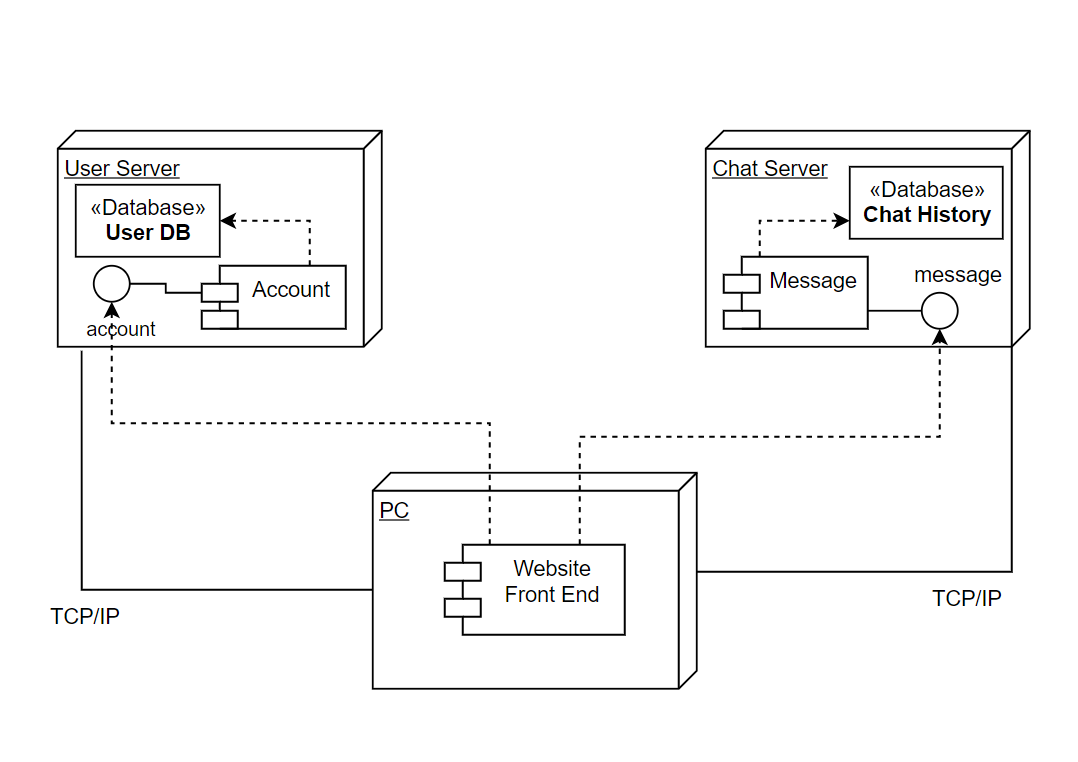
* Security requirements: Login username & password /User authentication
* Performance: The system will have a response time of 2 to 5 seconds
* Usability: It’s very user friendly, very easy for the user to learn/know what to do.
* Availability: The website is going to be available at any time that the user may need it unless it's going through maintenance. Which means it would be available to the user 98% of the time.
* Accessibility: The website will be accessible from any major web browser, and from laptops as well as tablets and phones.
* Expected load: This software should not have a bigger level of implementation than at a university level, no more than a few thousand users. Chat rooms should be able to manage a few hundred people inside of them at a time. (Numbers not final.)
* Storage: Usernames, e-mails, and passwords will be stored in a server database.

1. High-level system architecture and database organization

* Bootstrap (open-source framework license for the UI design)
* MySQL Language (for the database)
* HTML Language (for the UI design/front end)
* CSS Language (for the UI design/front end)
* JavaScript Language (for the functionality of the website/front end)
* Brackets open-source code editor (for programming)
* Google Chrome (supported browser)
* Safari (supported browser)
* Balsamiq (a tool to design the software)
* SQL Server (to host the database)
* PHP (language for the back-end development)
* DB Organization: The Database will be organized on phpMyAdmin and is called cen4010\_su21\_g01. The tables on the database will consist of:
  1. Forum\_posts (this table will store the forum post text information)
     1. Post id
     2. Topic id
     3. Post text
     4. Creation time
     5. Post owner
  2. Forum\_topics (this table will store the topic information)
     1. Topic id
     2. Topic title
     3. Creation time
     4. Topic owner
  3. users (this table will store the user’s personal information)
     1. First and Last name
     2. Username
     3. Password
     4. Age
     5. Gender

* Media Storage: Gopher will not allow media storage.
* Search/filter architecture and implementation: We are going to allow users to register and insert their name, username, password, age, gender, and a bio about themselves. They will be able to navigate through Gopher to see their account profile, home page, chat, settings, and logout. Users will be able to post to a forum and reply to other users.
* We will most likely not be using any APIs.
* There will likely be no non-trivial algorithm or process.

1. High-Level UML Diagrams 



1. Key Risks for Project

1. Skills Risks: The team skills are not all the same. Some people took intro to internet computing, database structures, and others did not. We are playing to our team's strengths by dividing and conquering based on who can do what.

2. Schedule Risks: We might have a schedule risk since we have some skills that we need to work on. We might not have enough time to finish everything as we have to learn some skills as we go. Our solution is to try to stay ahead so we don’t fall behind.

3. Technical Risks: Our team is not at all comfortable with the backend. We are trying to figure it out by watching tutorials so we can learn as much as we can to be able to get the project done. We also could run into issues with the software as we begin to develop it.

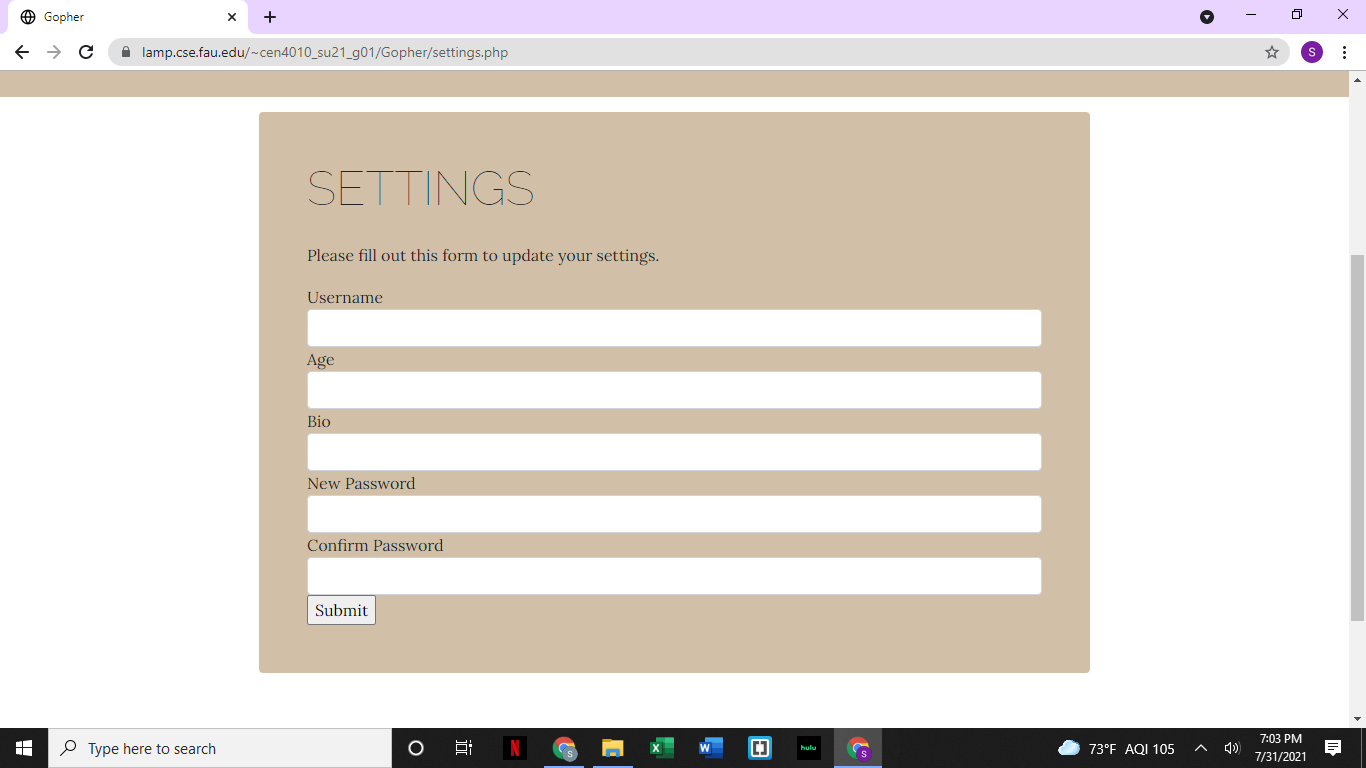
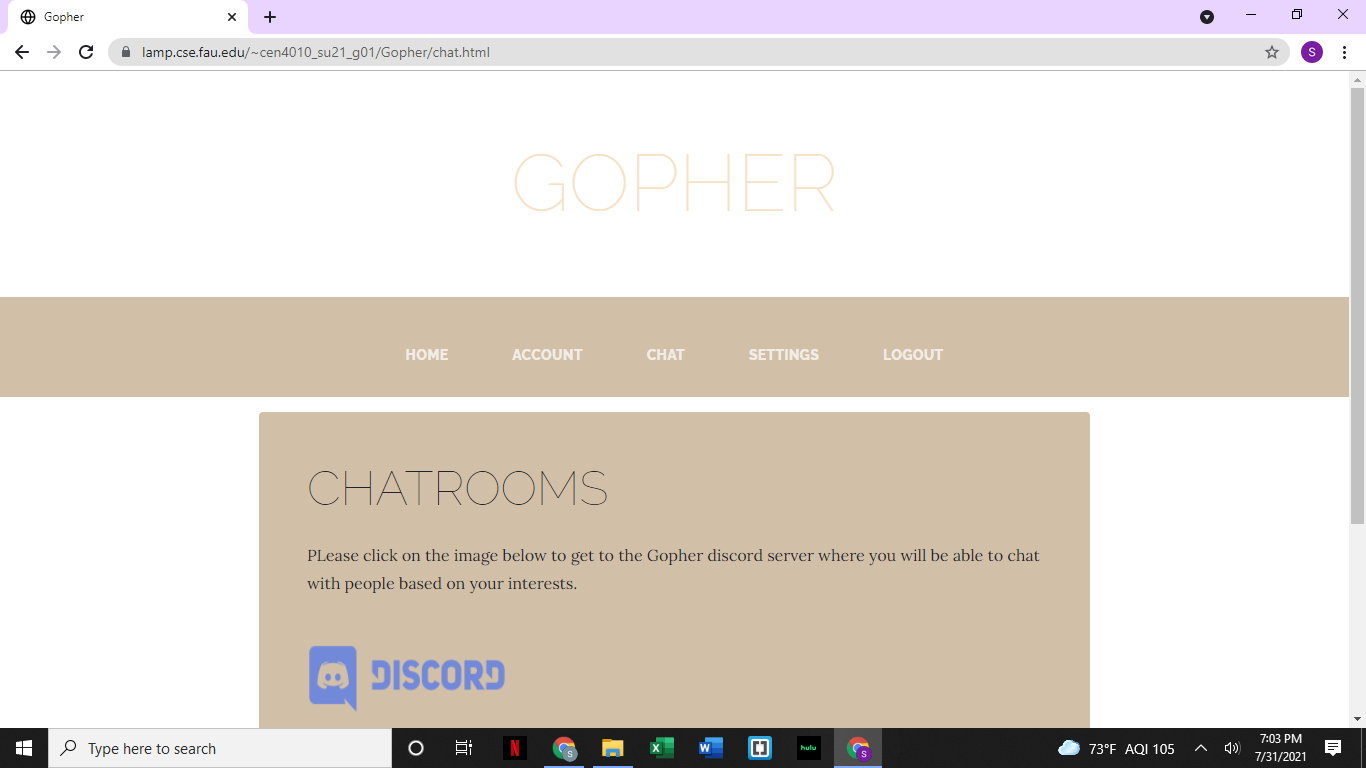
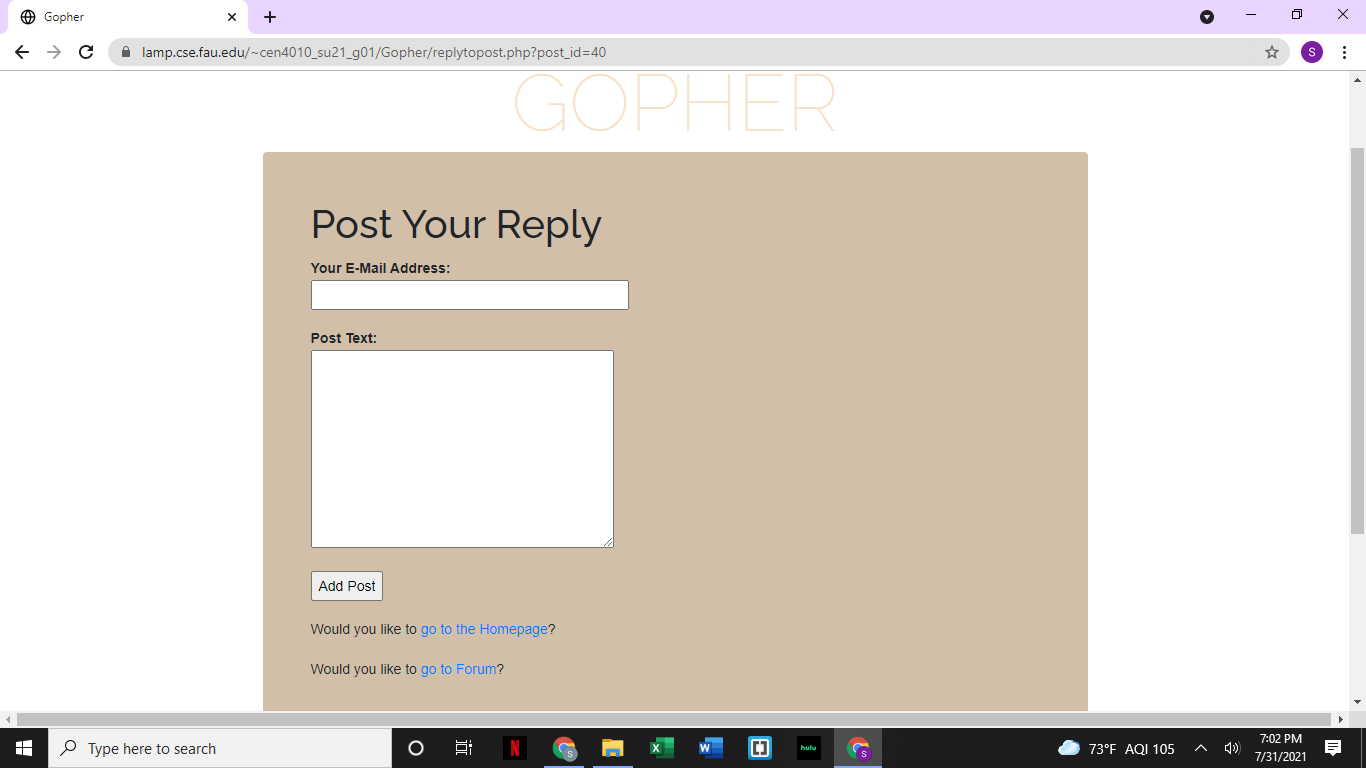
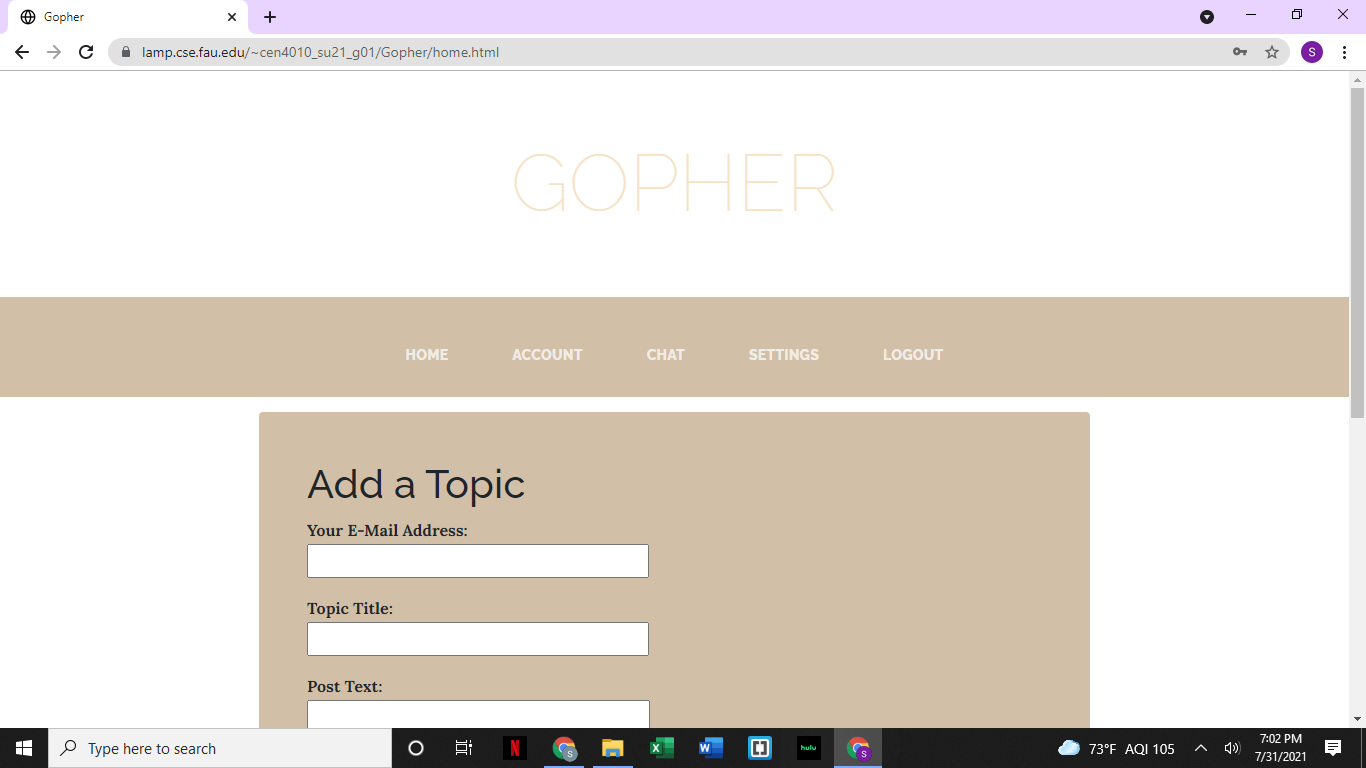
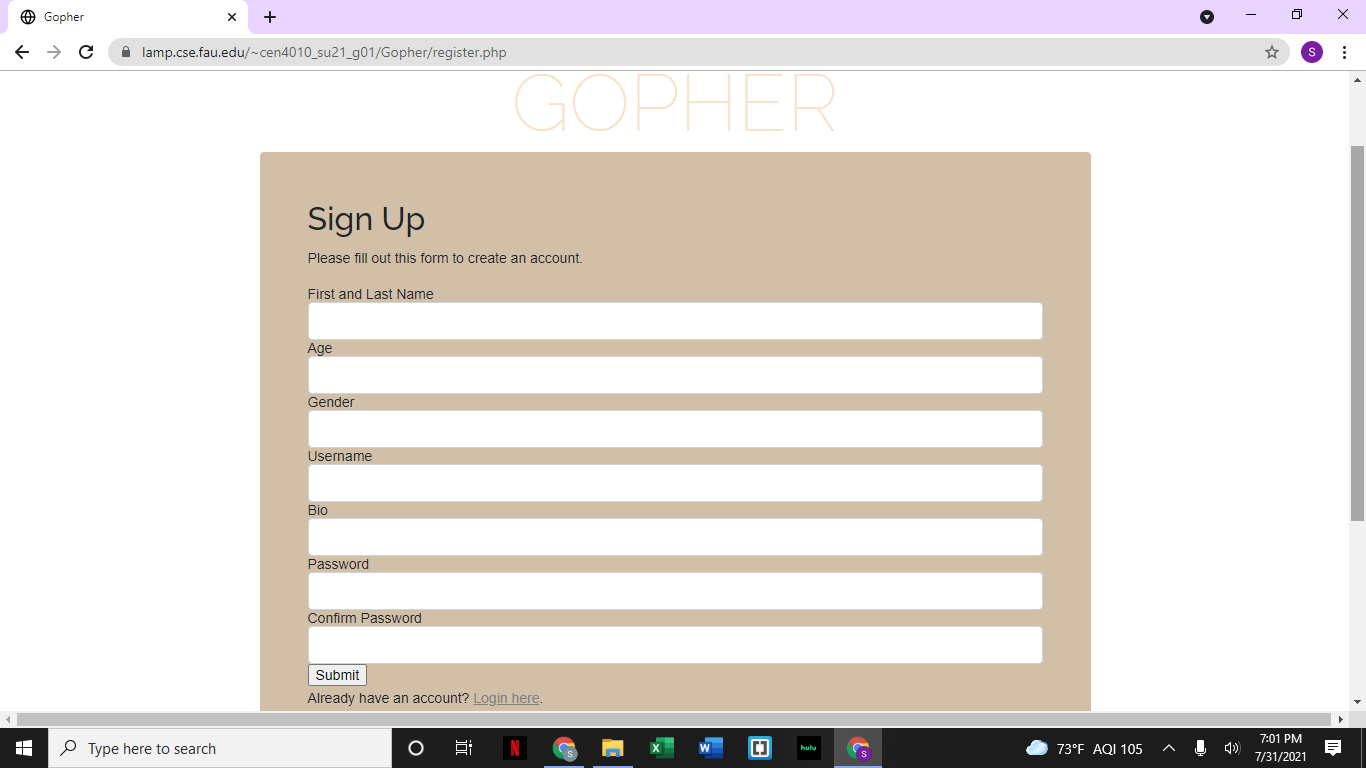
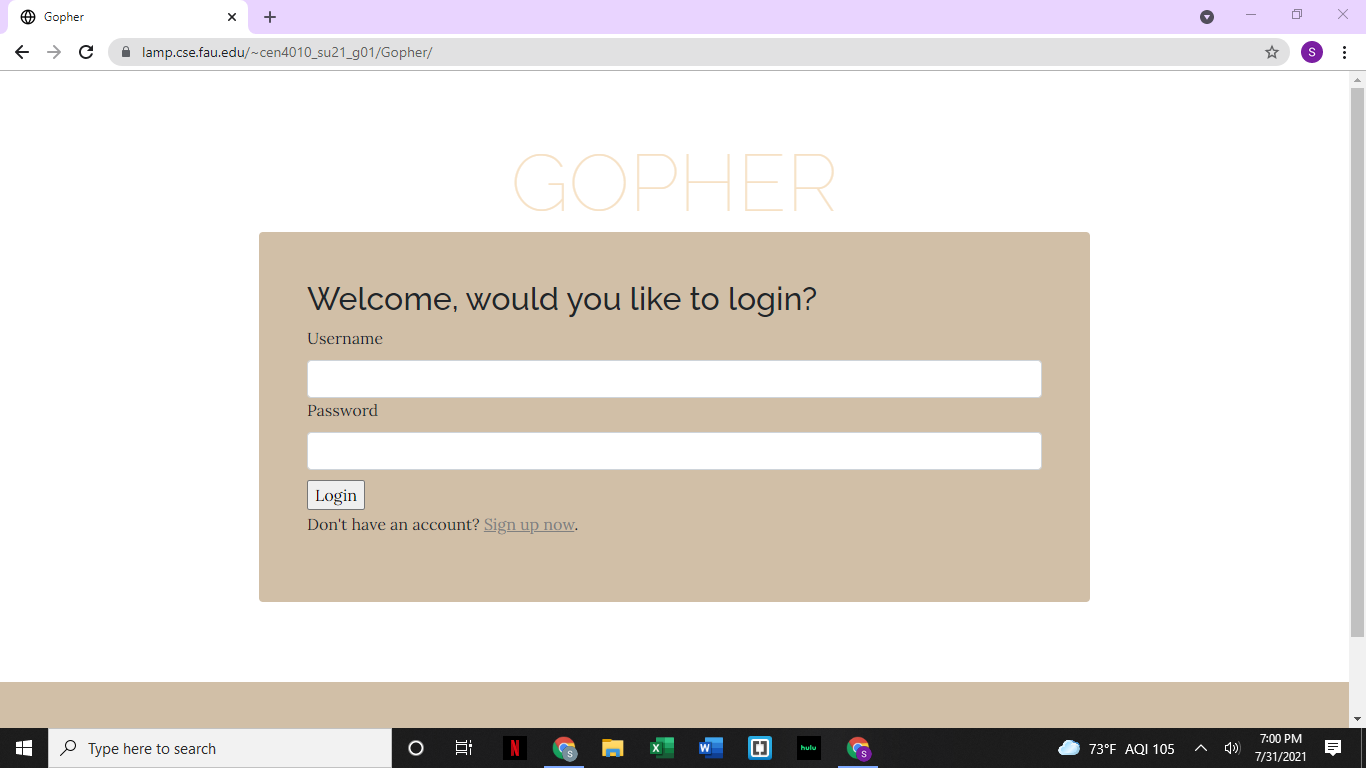
4. Teamwork Risks: We do have teamwork risks because we are not at full capacity. A member in our team hasn’t been responding. Our solution is to divide the work into 4 instead of 5.

5. Legal/Content Risks: N/A

**Instructor’s feedback for milestone 3 and 4:**

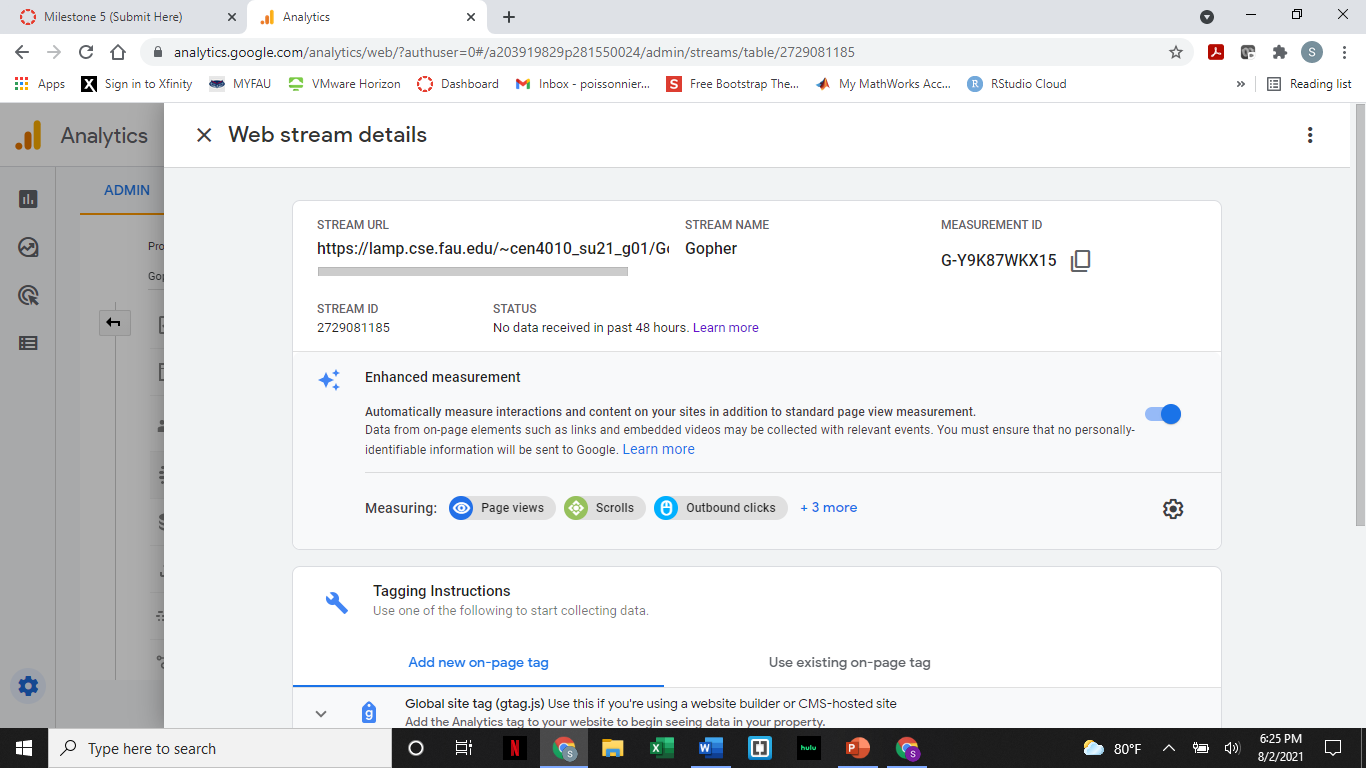
Milestone 3 lacked the changes that needed to be made in Milestone 1. The executive summary needs to provide a short description with advantages of the software and not each step that will be taken in using the software. The competitive analysis is also confusing and needs to be redone where the products have their own columns. Milestone 4 had the main issue of the product summary. The product summary looked like a list of answers rather than an attraction for users. The committed functionality was also not up to the rubric. The usability test plan was also inconsistent with the test objectives and the questionnaire.

**Screenshots of Final Product:**

****

**Google Analytics Plot:**

We were not receiving data for some reason.



**Team Member Contribution:**

1. Team Member Contribution to Project:
   1. Summer
      1. I developed the entire backend for the website and contributed to each write up of each milestone. Ania and I created the final draft for the design of the website in milestone 2. I was in charge of GitHub so I pushed each milestone into GitHub for the team to have access to. I made sure to submit each milestone for the group on time. I created the tables in phpMyAdmin and I tested our project each time on the lamp server. I also made every demo video that was required in the milestones. **Score: 27.5/20**
   2. Ania
      1. I created the group webpage for milestone 0. I contributed to the balsamiq frames layout for the website with Summer, Carlos and Joshua but Summer and I finalized the design. I contributed to the paperwork of every milestone. I was the scrum master so I always asked everyone which task they wanted to do in every meeting so I could then assign it to them. I developed the whole front end for Gopher because I was put in that position. Summer and I tested everything for the website to make sure it was working properly. I also created the Gopher server on discord. Score: **27.5/20**
   3. Josh
      1. I came up with the idea for the site, and throughout the project I led the discussions regarding what the site’s functions should be. I developed the preliminary version of the site, which Ania and Summer redesigned and implemented their own version of. Also, I did my share of the group paperwork. Summer and Ania took the initiative to do most of the work themselves. Which is great, except that I had no knowledge of the latest updates to the site, so I couldn’t contribute as much as I would have liked to. **Score: 20/20**
   4. Carlos.
      1. I contributed to the layout of the site with balsamiq frames and helped with the encryption of user data. I came up with the 1st proposal for encryption (which involved a cipher to encrypt user data), however the team ended up going for a simpler solution instead. I helped define the high-level functional requirements and non-functional requirements and ensure they were implemented. I have no knowledge of php and html, so my contributions to the project were not as crucial or impactful to its development as I would have hoped. **Score: 20/20**
   5. Rodrique

I contributed to the website by completing the analysis on what our website is capable of doing and the features it comes along with, also I made a comparison on the website with other dating websites. In addition, I was told a logo was needed for the website I completed but I guess it wasn't needed anymore. The more in-depth parts I was trying to help with but due to my lack of experience in software development I could only do so much. I did the best I could and helped to the best of my ability and truly appreciate my teammates for doing the best they did to get this project done. **Score: 5/20**

1. Contribution to teamwork:
   1. Summer
      1. I contributed to the team by scheduling meetings, sending zoom links for said meetings, and making sure the work got done on time. I also helped the team out by making sure to submit the milestones before they were due and push them to GitHub for everyone to have access to.
   2. Ania
      1. I joined every meeting, discussed what needed to be done and what everyone wanted to do so that as the scrum master I could assign everyone their chosen task.
   3. Josh
      1. I initiated many of the team zoom meetings, in which we discussed the tasks at hand and delegated the responsibilities among the team members.
   4. Carlos
      1. I helped the team with brainstorming in general and with the site layout. I took up all of the leftover tasks for the milestones and tried to help with everything (no matter how small) that my skill set allowed for.
   5. Rodrique

i. I joined in on the meetings when I was able to join and talk to my team members, on the project and what needs to be done, and when I didn't attend, I would ask what was talked about in the meetings.

1. Number of submissions to GitHub:
   1. Summer
      1. 7
   2. Ania
      1. 1
   3. Josh
      1. 1
   4. Carlos
      1. 1
   5. Rodrique
      1. 1

**Post-Project Analysis:**

**Team Lead: Summer Poissonnier**

The development of Gopher was definitely a lengthy and challenging process. There were many issues that Ania and I ran into when developing Gopher. The main challenge that we faced with this project was not every team member had experience with creating websites and doing backend/database work, so the workload was not divided evenly (for this reason, it was challenging to get the website done in time as Ania and I were the ones that had to take on that task). Another challenge for me was figuring out how to create the backend since I had never done something like that before. I ran into many issues when testing Gopher due to issues with the backend. Another challenge was time management because Ania and I had to learn how to do backend and some frontend things that we have not yet learned how to do and we had to do this in addition to doing our part for the write up in each Milestone and developing Gopher. Next time I would get a head start on developing the website so all of the original functionalities will be able to make it. I would also do more research and learn more about front end development and back-end development ahead of time. The features we managed to finish despite our challenges were: the login page, the forum page, the chat page, and the account and account settings page. I would have liked to finish the uploading of an image portion of our website and the connecting with friends’ portion of the website. The reason we could not get these functionalities finished is because we did not have time, even contribution, and skill set to do everything we originally planned. We gained the knowledge of how to develop the backend and how to work with php, sql, and databases. This is very important as this skill will be useful in future careers and projects. We learned from this project that it takes a lot of work, effort, and time to develop software; especially when not everyone has the same skills.