CEN 4010 Principles of Software Engineering, Summer 2021

Team 1, Gopher

Team #1

Summer Poissonnier [spoissonnier2019@fau.edu](mailto:spoissonnier2019@fau.edu)

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

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

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

July 1, 2021

**Milestone 3: More Detailed Requirements, Architecture, and Vertical Prototype**

**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 interest’s 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

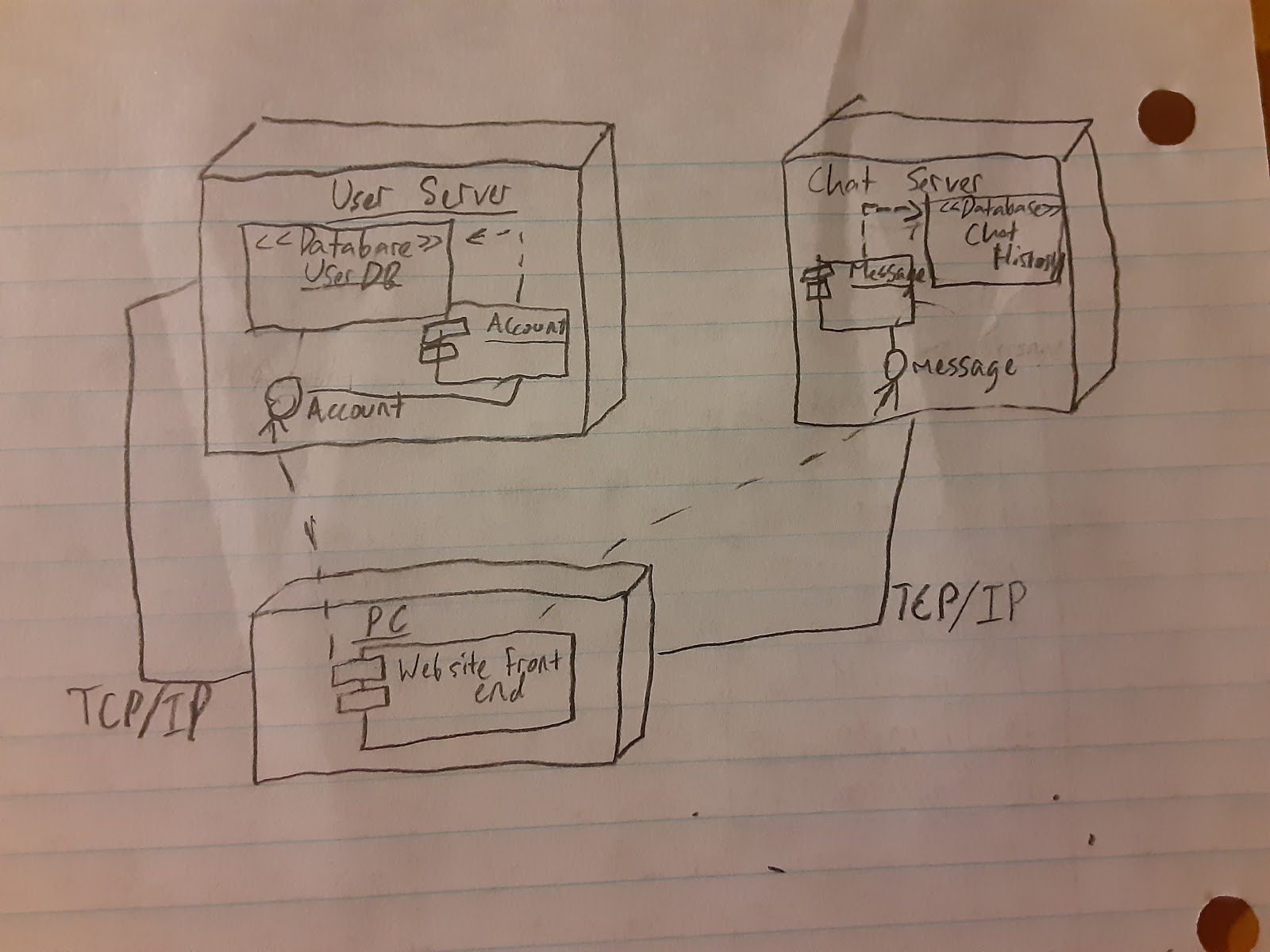
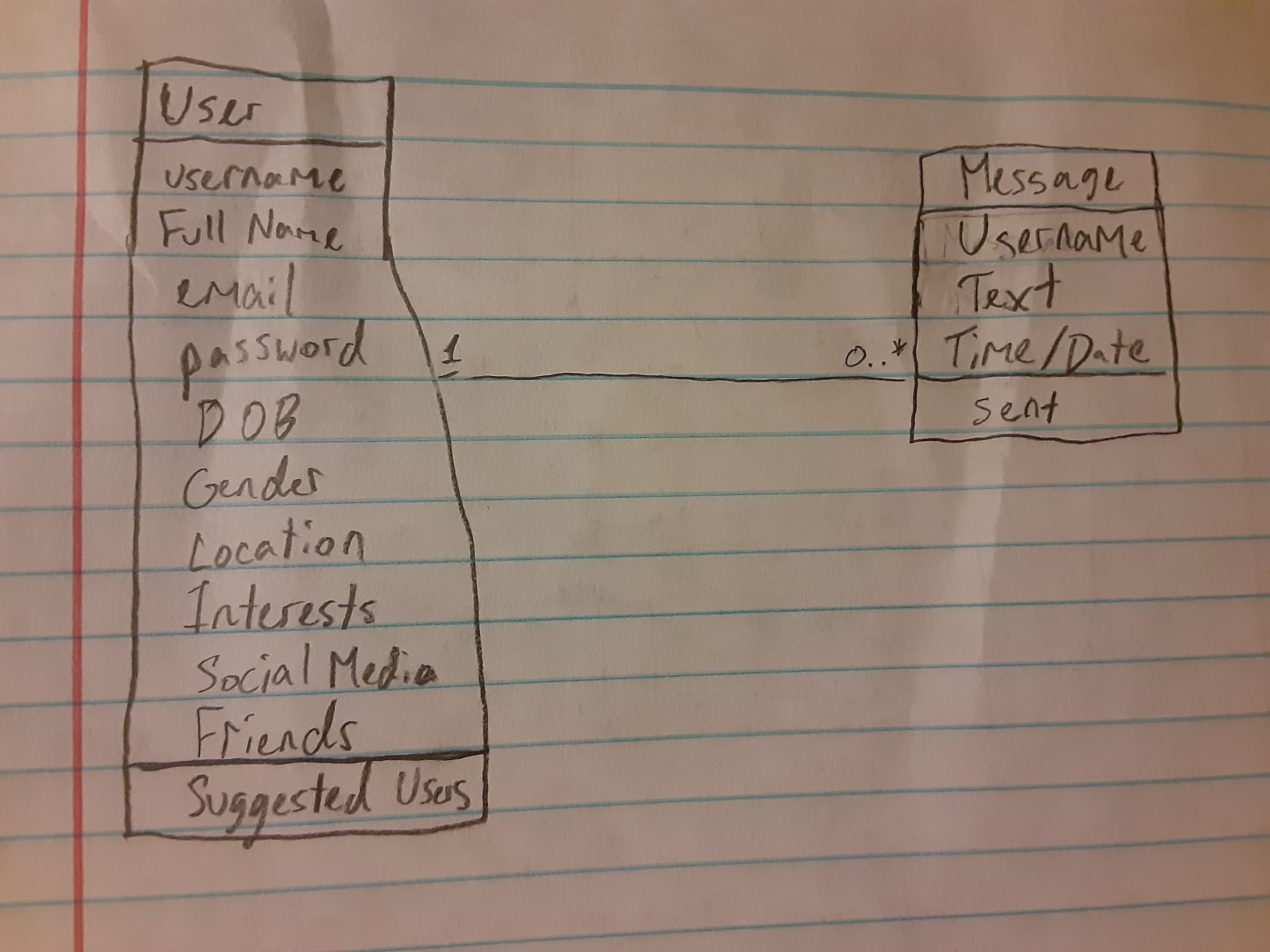
1. List of non-functional requirements

* Performance: The system will have a response time of 2 to 5 seconds
* Security requirements: Login username & password /User authentication
* 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 tables 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. Interests\_table (this table will store all of the available interests’ users can choose from)
     1. Cooking
     2. Traveling
     3. Video Games
     4. Gardening
     5. Pets
     6. Writing
     7. Reading
     8. Fishing
     9. Volunteering
     10. Television
     11. Sports
     12. Dancing
     13. Music
     14. Shopping
     15. Painting
  2. Users\_table (this table will store the user’s personal information)
     1. First and Last name
     2. Email
     3. Password
     4. Date of Birth
     5. Gender
     6. Location
  3. Social\_table (this table will store the user’s social media information)
     1. Instagram
     2. Twitter
     3. Facebook
     4. Snapchat
  4. Friends\_table (this table will store the friend’s list for each user)
     1. First and last name
  5. Suggested\_table (this table will store a list of suggested friends)
     1. First and Last name
     2. Interests
* Media Storage: Gopher only allows the upload of images for profile pictures. We will store these pictures in the database.
* Search/filter architecture and implementation: We are not going to allow users to search on Gopher. We will use the database to filter suggested friends and user’s will be able to click on their suggestions to see what friends Gopher recommends. The interests form where users can fill out their interests will be a page that allows the user to check what interests they have and store this in the database.
* 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