**This revision date: 9/28 revision time: 4:48 pm**

CEN4010 – Principles of Software Engineering. Fall 2021

Milestone 1 – Project Proposal

Group 10 – Emanon

**Maxon Corvil** [mcorvil2018@fau.edu](mailto:mcorvil2018@fau.edu),

**Ashley Davis** [davisashley2020@fau.edu](mailto:davisashley2020@fau.edu),

**Patrick Messina** [pmessina2017@fau.edu](mailto:pmessina2017@fau.edu),

**Michael Niebauer** [mniebaue@fau.edu](mailto:mniebaue@fau.edu),

**Andrew Sexton** [asexton2020@fau.edu](mailto:asexton2020@fau.edu)

Due September 28, 2021

Possible project names: Chitter-Chatter, Pigeon Post, Relay, Emanon

Initial project overview/summary

This project is a social media site that provides high-quality connections between its users. Intended as a method of keeping in touch and maintaining a human connection between friends and family when physical contact is not possible.

**Some Details:** Users are able to foster a more in-depth connection than other social media sites through live chatrooms with their friends that allow them to have real-time conversations whenever they feel the need to talk. If users simply want to share something with people they know, they can make posts about whatever they need to share. Users should also feel comfortable using the site with privacy levels that let users specify who is able to see their posts.

**Chatrooms**– text chat minimum, voice chat/calling as a stretch goal. Should be private and support a few people in one room.

**Posts** – short “tweet” style posts as they default, optional checkbox to expand the post length. Expanded posts have additional Teaser field where users can specify a preview to appear on other user’s feeds/pages. If not specified, longer post is truncated.

**Privacy levels** – A minimum of 3 privacy levels can be specified for a post:

* Public- where anyone can see a post
* Private/friends – where only other users that are at the friend level or higher can see the post,
* Privileged/best friends – where only best friends/privileged friends can see this post.

Two users become friends after one user send the other a friend request, and the recipient accepts it. Each user can set the “friendship level” of each of their friends to either:

* Normal/friend – default friendship level for two users that have become friends through an accepted friend request
* Privileged/best friend – higher friendship level that can see all posts.

Once two users are friends, each one can independently set the friendship level for the other. For example, users A and B just accepted a friend request, by default, user A can see user B’s public and private posts, but not extra private posts (same for user B seeing user A’s posts). If user A raises B’s friendship level to privileged, user B can now see all of user A’s posts, but user A can still only see user B’s public/private posts. User B does not receive any indication that user A has raised their privacy level (or lowered it), and user B has no indication when looking at one of A’s posts whether it is a public, private, or extra-private posts. Thus, user B (or any user) has no way of knowing what user A has set for their privacy level.

1. **Executive Summary -Andrew**

Keeping in touch with friends and family can be difficult nowadays, especially when people are no longer able to regularly meet to socialize in person. This is an issue that has become extremely apparent since the start of quarantines during the Covid-19 pandemic. Several social media sites that existed prior to the pandemic provide some method for staying connected to friends and family, but most of them fail to come close to emulating the experience of in person socialization.

Emanon is a new social media site that intends to improve upon existing platforms by creating higher quality connections between its users. Emanon is a tool for users to keep in touch and maintain a strong connection with their family and friends whenever in person meetups are not possible, such as during lockdowns or when friends move long distances. It is intended to be a one stop site for users to share details about their day-to-day lives with those they care about.

Emanon will provide users with the expected features of a social media site, posts, a friends list, image, and video sharing. Users will also be given enough control over their posts so that they can feel comfortable sharing any detail of their lives and knowing that only the intended recipients will see the news. Additionally, Emanon lets users have dynamic meetups where they can socialize in real time, coming closer to a real, in-person meetup than other social media sites. Unlike other social media sites that have become a tool for advertisers, Emanon is focused on connections between people, not businesses. Therefore, accounts are restricted to only real people. This is one of the many ways that Emanon hopes to help its users maintain a deep connection with those they care about.

1. **Competitive Analysis -Andrew**

In this section Emanon is compared to some of its competitors on the following metrics: the types of posts available to each site’s users, privacy options for posts, the implementation of a live chat feature, implementation of an activity, ease of use, and account security. Each metric will be rated on a 5-point scale where 1 is bad, 2 is subpar, 3 is decent, 4 is good, and 5 is extremely good. Five competitors were chosen based on their potential for connecting users.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Emanon** | **Twitter** | **Tumblr** | **Facebook** | **Snapchat** | **Instagram** |
| **Post Types** | **4** | **2** | **5** | **5** | **3** | **3** |
| **Post Privacy** | **5** | **3** | **1** | **5** | **5** | **4** |
| **Live Chatrooms** | **4** | **3** | **2** | **4** | **5** | **4** |
| **Activity Status** | **5** | **1** | **2** | **4** | **3** | **2** |
| **Ease of Use** | **5** | **4** | **2** | **1** | **5** | **4** |
| **Account Security** | **3** | **4** | **4** | **5** | **3** | **4** |

**Relay (4.33)** <https://lamp.cse.fau.edu/~cen4010_fa21_g10/>

Relay will have default posts of 200 characters with an option to create longer posts with a 200 character teaser that will appear when seen on a feed posts will also support videos and images in posts. Relay will allow users to specify the privacy level of each of their posts individually, restricting whether each post can be seen by anyone, only their friends, or only a subset of their friends. Relay will also have live chatrooms which provide a space for users to socialize with their friends in real time. Relay will also have activity statuses that will let a user’s friends know when they are online and available, busy, or offline. Relay will have a straightforward design that makes it easy for users to make posts, see their friends’ posts, or join chatrooms. Relay plans to implement an optional 2-Factor Authentication (2FA) for users to secure their accounts.

**Twitter (2.83)** <https://twitter.com/>

Twitter does not provide posts longer than 280 characters by default, but it does let users make multiple posts at once that are linked in a thread. Posts may also include images and video. Twitter has a binary, account wide setting for post privacy where all posts by an account are either public or restricted to their followers. Twitter does not have live chatrooms, but their direct messages/group conversation feature is similar as it provides private conversations between groups of users. Twitter does not have built in method for tracking who is online, even for followers. Users can still estimate who is active by checking timestamps on recent posts. Twitter has a well-designed interface with a post feed and simple search bar. Twitter has 2FA and the ability to require a phone number or email address to reset a password.

**Tumblr (2.67)** <https://www.tumblr.com/>

Tumblr’s text posts do not have a character limit, and may include images, video, audio, quotes, and Chats which resemble a script for a movie or play. Tumblr posts are viewable by anyone using the website, the exception is private posts which are only viewable by their creator. Tumblr has a direct messaging feature that allows two users to have a private conversation but does not have group chats. Tumblr has also has a simple green dot to indicate when a user is active. Making posts and searching for content are straightforward, however, long posts are not shortened when they appear in a feed, which can be an annoyance. Tumblr also provides 2FA and will send emails when there are logins on new devices.

**Facebook (4)** <https://www.facebook.com/>

Facebook posts have a roughly 60,000 character limit and truncates long posts to roughly 500 characters on a user’s feed. Posts can contain a variety of media including images, video, and audio. Facebook has a strong post privacy feature that allows users to specify whether a post is public or friends-only and specify a subset of friends to include or exclude. Facebook allows users to have private group chats where they can share messages and media or have conversations over voice or video. Facebook shares users’ activity status with their friends by default but gives users the option of manually managing which friends can see their status or hiding it entirely. Facebook is not easy for new users learn how to navigate and use. It has many features that clutter the screen and make navigating the site difficult for new users. Facebook has many security features including 2FA and login alerts for new devices, it also has optional encryption for emails from Facebook.

**Snapchat** **(4)** <https://www.snapchat.com/>

Snapchat is a phone app for sharing images and video that are deleted after being viewed. Each video is up to 10 seconds long, but videos can be grouped up to one minute in length. Users can select which friends are sent their posts on a case-by-case basis, or post to their Story which can be configured so that anyone can see, or just friends. Snapchat also lets users privately message each other in groups of up to 32 users, supporting text, video, and voice. Snapchat does not let users see which of their friends are currently using the app by default, but in group chats users can see when the other members are typing. Additionally, users are able to see whether their friends have viewed their posts or not. Snapchat has a UI similar to a camera app with a navigation bar along the bottom for accessing its features. Snapchat also has 2FA for accounts.

**Instagram (3.5)** <https://www.instagram.com/>

Instagram allows users to post images and video, with an optional caption of up to 2,200 characters. There is no post-by-post privacy control for default posts except for making an account private so that only followers can see the account’s posts and new followers must be approved. For more privacy control, users can create a Story post which lets them choose between sending the Story as a direct message to one or more users, posting it to their Story, or posting it to their Close Friends list. Followers can be manually removed from the Story to hide it from them and must be added to the Close Friends list to see Close Friends Stories. Instagram also has group chats where two or more users can have a conversation via text, voice, or video. By default, Instagram also allows users to see the activity status of their followers, or people who they receive a message from. If users can turn this feature off, their activity is hidden from everyone, but they can no longer see the activity of anyone else. Instagram is simple to use for both making posts and searching for content. Some features, like the chatroom, are only accessible in specific places on the app. Instagram supports optional 2FA and maintains a list of official emails sent from Instagram going back 14 days for each account.

**Planned Advantages**

Emanon’s goal is to be more focused on connecting its users than any other social media site. Many other sites make it possible for users to have conversations with each other, but few emphasize users actively talking to each other at the same time. Emanon will feature live chatrooms that only allow users who are currently online to participate in, strengthening the connection between users as they know other participants are active in the conversation. Emanon will also aim to make it easier for users to share details about their life than other sites by allowing a wide variety of posts. We also aim to give users enough control over their posts so that they feel comfortable sharing news with their friends and family.

1. **Data Definitions -Ashley**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Meaning | Usage | Description |
| User | Actor | Use Case scenario | General term for person navigating the site |
| Non-member | Actor | Use Case scenario | User who does not have an account registered with the site |
| Member | Actor | Use Case scenario | User with a registered account |
| Friend | Actor | Use case scenario | Member that is followed by the account owner |
| Non-friend | Actor | Use case scenario | Any member the account owner does not follow |
| Account | Data | Use case scenario | Information stored in the website database in connection to a user |
| Long post | Data | User generated content | User post containing 200 characters or more. May contain image or video |
| Short post | Data | User generated content | User post containing less than 200 characters. May contain image or video |
| Home page | User Interface | Default page | First page the user encounters. Option to log in, sign up, or view global posts |
| Website | User Interface | User interface | Front-end display for user interaction |
| Feed | User Interface | Use case scenario | Page to view posts from members and interact with those posts |
| Log in | User Interface | Use case scenario | Web page for signing into account. Requires user credentials (e.g., email and password) |
| Friendship level | Feature | Use case scenario | A rank among friends. Can be set at best friend or friend. |
| Filter | Feature | User service | User can sort posts based on relevancy and date |
| Follow | Feature | User service | Add member to friends group. Displays member’s content in feed |
| Delete | Feature | User service | Remove data |
| Search | Feature | User service | User can look up posts and members related to the keywords inserted |
| Like | Feature | User service | Allows user to react positively to a post with one click of a button |
| Comment | Feature | User service | Allows members to reply to posts |
| Photos | Feature | User service | Allows member to upload pictures |
| Videos | Feature | User service | Allows member to upload videos |
| Private | Security feature | Privacy | Minimal level of access to shared content. Non-friends can only see one’s name and profile photo. All other content is hidden |
| Privileged | Security feature | Privacy | Access to exclusive content shared by the account owner. Only selected friends can view the content |
| Public | Security feature | Privacy | Content is shared to any user on the site with no restriction |
| Activity status | Feature | Member status | Indicates whether user is online, busy, or offline |
| Chatroom | Feature | Live message room | Private messenger between two or more users that allow real time connection |
| System | Hardware platform and services | Use case scenario | Front-end design, back-end services, MySQL database, and all code |
| Emanon\* | Domain | Use case scenario | Represents the website and all associated web pages |

1. **Overview, Scenarios, and Use Cases -Maxon / Patrick / Ashley**

**Use Case – Homepage:**

The user goes to the home page and wants to see global posts and has not logged on or created an account.

1. **Description:** Use case describes the first thing that the user sees when going to the website.

2. **Actors**:

2.1 Existing user that is not logged in or new user

2.2 System

3. **Preconditions**:

3.1 User has an active internet connection

4. **Primary Flow of Events**:

4.1. User arrives on web page.

4.2. User has access to public / community posts

4.3. User can either create an account or login from this page

5. **Alternate Flows**

5.1 **If the user attempts to interact with posts**

If user tries to interact with posts, they will be forced to either login or create an account.

1. Web site notifies user that they need an account to proceed

2. Return user back to previous post

**Use Case – Sign Up:**

When a user goes to the homepage, they will be shown public posts, but they will need to login or create an account if they want to interact with posts or create a post. If they do not have an account, they can click on sign up to create a new account.

1. **Description:** Use case describes the process of the user creating an account.

2. **Actors**:

2.1 New user

2.2 System

3. **Preconditions**:

3.1 User has an active internet connection.

3.2 User has an email not connected to another account.

4. **Primary Flow of Events**:

4.1 User arrives on web page.

4.2 User clicks on create account.

4.3 User enters an email, password, and first and last name.

5. **Alternate Flows**

5.1 **If the user attempts to use an email that is already used**

If user tries to use an existing email in step 4.3

1. Web site notifies user that the email Is already being used

2. gives user the option to change email or login.

**Use Case – Login:**

When a user goes to the homepage, they will be shown public posts, but they will need to login or create an account if they want to interact with post or create a post. If they have an account, they can just sign on with their credentials.

1. **Description:** Use case describes the process of the user logging into an account.

2. **Actors**:

2.1 Existing user

2.2 System

3. **Preconditions**:

3.1 User has an active internet connection.

3.2 User has an existing email connected to an account.

3.3 User isn’t already logged in

4. **Primary Flow of Events**:

4.1 User arrives on web page.

4.2 User clicks on sign in

4.3 User enters an email and password

4.4 User is taken to their personal feed

5. **Alternate Flows**

5.1 **If the entered credentials aren’t found on the database**

In this situation, the user entered credentials that aren’t found on the database.

1. Web site notifies user that one or both of the fields need to be changed

2. Gives user the option to reenter the credentials or to create an account.

**Use Case – Create Short Post:**

After a user has already verified their account, they get the ability to create posts, when creating a post, the user can select post type (long or short if under 200 characters). Additionally, user can add images or videos. Finally, user gets the option of who can view the post so. It can be either public, friends only, or selected individuals/close friends.

1. **Description:** Use case describes the process of the user creating a post, customizing it, and selecting who can view it.

2. **Actors**:

2.1 Existing user

2.2 System

3. **Preconditions**:

3.1 User has an active internet connection.

3.2 User has already logged in

4. **Primary Flow of Events**:

4.1 User is already logged on

4.2 User clicks on create new post

4.3 User customizes post using 200 characters or less

4.4 User selects the privacy settings for this post

5. **Alternate Flows**

5.1 **If the post is over 200 characters**

In this situation, the user enters over 200 characters. Website notifies user that the post has become a long post and they can either send the post as is, delete it, or change it into a long post

1. If this is a long post, the user has to option to create a snippet/teaser for the post that follows the short post restrictions. There is no cap for the length of the long post.
2. If the user wants to post as is then the post will be sent as a short post
3. If user deletes the post, then nothing will be saved or posted

5.2 **if the user cancels the post**

In this situation, the user opts to not create a post at all. Website asks if they are sure and they can delete the post permanently or send it to drafts.

1. If user deletes the post, then they will be returned to dashboard
2. If the user saves to drafts, then it will be saved in the data base but not viewable by anyone.

**Use Case – Create Long Post:**

After a user has already verified their account, they get the ability to create posts, when creating a post, the user can select post type (long or short if under 200 characters). This case covers when the user selects “Long Post”. Additionally, user can add images or videos. Finally, user gets the option of who can view the post so. It can be either public, friends only, or selected individuals/close friends.

1. **Description:** Use case describes the process of the user creating a long post, customizing it and selecting who can view it.

2. **Actors**:

2.1 Existing user

2.2 System

3. **Preconditions**:

3.1 User has an active internet connection.

3.2 User has already logged in

3.3 User has surpassed 200 characters or user has already opted for a long post

4. **Primary Flow of Events**:

4.1 User is already logged on

4.2 User clicks on create a new post and surpasses 200 characters or opts to make it longer

4.3 The user will be given two text fields to customize,

4.3a One field is optional and allows the user to enter up to 200 characters. This will be the snippet/teaser for this post and will appear in other users’ feeds

4.3b The other field has no limit and contains the main content of the long post

4.4 User selects the privacy settings for this post

4.5 The user finishes customizing their post by submitting it

5. **Alternate Flows**

5.1 **If the user submits without creating a custom snippet.**

In this situation, the user opts to not create a snippet for their long post. Website notifies user that the snippet will be automatically made from the first 200 characters. Users may either accept or change the snippet.

1. If user accepts, then the system will automatically use first 200 characters of the main content as the snippet
2. If the user declines, then they can create their own teaser

5.2 **if the user cancels the post**

In this situation the user opts to not create a post at all. Website asks if they are sure, and they can delete the post permanently or send it to drafts.

1. If user deletes the post, then they will be returned to dashboard
2. If the user saves to drafts, then it will be saved in the data base but not viewable by anyone.

**Use Case – Search:**

After a user has already logged into their account, they get the ability to search. The search is one search and will return users and public posts with the query inside of the post.

1. **Description:** The search function allows the user to find other users by name and posts based on content. One search will return both results on the same page and the user can interact with the results.

2. **Actors**:

2.1 Existing user

2.2 System

3. **Preconditions**:

3.1 User has an active internet connection.

3.2 User has already logged in

3.3 User has a query in mind

4. **Primary Flow of Events**:

4.1 User is already logged on

4.2 User enters search terms and sends them in

4.3 The search returns the results and user can interact with them

5. **Alternate Flows**

5.1 **If the user search returns no results**

In this situation, the users search returned no results, so there is nothing for them to interact with.

1. The User is shown a “no results” screen
2. The user can either search again or return to their personal feed.

**Use Case – Add friend/follow:**

After a user has already logged into their account, they get the ability to follow/add other users. This will populate a user’s personal feed with the post of users they follow/add and give them the ability to interact with the other person.

1. **Description:** Adding a user can be done from either the search screen or from the public post screen and it adds the users who you are following’s posts to your personal feed.

2. **Actors**:

2.1 Existing user

2.2 Another existing user

2.3 System

3. **Preconditions**:

3.1 User has an active internet connection.

3.2 User has already logged in

3.3 User has found another user

4. **Primary Flow of Events**:

4.1 User is already logged on

4.2 User finds another user they want to add/follow

4.3 The followed user’s posts get added to the follower’s personal feed.

5. **Alternate Flows**

5.1 **If the user is already added/followed**

In this situation, the two users are already friends. If this happens then the system alerts the user of this, and they can either remove that person as a friend or cancel.

1. The user is alerted by the system to tell them that they are already friends

**Use Case – Like/comment on a post:**

The user has the option to like or add comments on posts. This requires that the user is logged into their account. When they like the post, the count will go up with an animation. If they comment on the post, comments will appear under the post.

1. **Description:** This feature will create the feeling of community as it gives people the ability to interact with others posts. They will also have the option to react with preselected emojis.

2. **Actors**:

2.1 Existing user

2.2 Another existing user’s post

2.3 System

3. **Preconditions**:

3.1 User has an active internet connection.

3.2 User has already logged in

3.3 User found a post they want to like or comment on

4. **Primary Flow of Events**:

4.1 User is already logged on

4.2 User finds another user’s post

4.3 the likes on the post go up by 1 if they like it and the comment appears as the newest comment if they comment.

5. **Alternate Flows**

5.1 **If the user deletes or unlike the post**

In this situation, the user either deletes their comment or unlike the post.

1. The user is alerted by the system questioning if they are sure
2. If they proceed then the total likes will go down by 1 or the comment under the post is removed

**Use Case – Chat :**

Users have the ability to chat with other users. This requires that the user is already logged into an existing account and both users are currently online. All chats are saved on the database.

1. **Description:** This feature will create the feeling of community as it gives people the ability to message each other. Chats will take place in real time but for users’ security only friends can message each other.

2. **Actors**:

2.1 Existing user A

2.2 Another existing user B

2.3 System

3. **Preconditions**:

3.1 Both users have an active internet connection.

3.2 Both users are logged in

3.3 User initiating the chat is friends with the user they want to message

4. **Primary Flow of Events**:

4.1 User A is already logged on

4.2 User A is already friends with user B

4.3 User A sends a chatroom request to user B, along with a short message.

4.4 Chatroom request appears on user B’s screen, and user B accepts the request.

4.5 A message box appears on both users’ screens, allowing them to send messages to each other. Sent messages will appear as a stack with newest being closest to the bottom.

5. **Alternate Flows**

5.1 **If the user tries to start a chat with someone who is not their friend**

In this situation, the user attempts to chat with someone who is not on their friend list.

1. The user is alerted by the system with a message that states they need to be friends first.
2. The message also gives them the ability to send a friend request or cancel.

5.2 **If user B declines the chat request**

In this situation, user B declines the request and may send an optional message to user A

1. User B declines user A’s chat request, and may optionally enter a message to send to user A explaining why.
2. User A receives a notification that user B declined their chat request, along with any message sent by user B
3. **Initial List of High-level Functional Requirements -Everyone**
4. **Create Account**
   1. Users should be able to create an account by supplying an Email, Password, FirstName and LastName to the system. This information will be stored and used for future logins and search results. Users will not be able to create an account if any of the Email, Password, Re-enter Password, FirstName, or LastName fields are not filled out. Account creation will be denied if the provided email is in use by another account, the Password and Re-enter Password fields have different contents, or the password does not meet minimum requirements of 8 characters, and at least one of each of the following: uppercase letter, lowercase letter, number.
   2. **Response Sequence**
      1. User enters their Email
      2. User enters a Password
      3. User re-enters their Password
      4. User enters their FirstName and LastName
      5. System checks if the Email is used by another account
      6. System verifies that both Password fields match and meet the minimum requirements
      7. System stores Email, FirstName, and LastName
      8. System will alert the user that the account was created
      9. System will provide a button to redirect the user to the main page
5. **Create Post**
   1. Users will be able to submit posts to the database when they are logged into an account. Users must provide text content and optional media content that includes images or video. Posts longer than 200 characters will provide the user with an optional Teaser field where they can enter up to 200 characters. The Teaser will appear in other users’ feeds or when searching for posts. Users may also create Teasers at any time by selecting an optional Long Post checkbox. When the user submits the post, it will be saved to the database.
   2. **Response Sequence**
      1. User opts to create a post
      2. User enters text into the Content field
         1. If the Content field exceeds 200 characters, or user selects the optional Long Post checkbox, the user is provided with an additional Teaser field
         2. User enters up to 200 characters in Teaser field
      3. User provides an optional Image or Video with their post
      4. User finalizes and submits their post
      5. System stores the post in the database and notifies the user’s Followers and Friends
6. **Chatrooms**
   1. The system will provide a method for two or more users that are friends, and currently online, to create a private chatroom where they can send text messages to the other participants in real time. When all participants leave the chatroom, all messages exchanged will no longer be available to the participants
   2. **Response Sequence**
      1. One user sends a chatroom request to one or more friends
      2. At least one friend accepts the chatroom request
      3. The system places the user and all accepting friends into a private chatroom and provides all participants with an interface to share messages, send chatroom invites to additional users, and leave the chatroom
      4. Participants share messages to the chatroom
      5. The system temporarily stores these messages
      6. When all participants leave the chatroom, it is closed and the system deletes the messages.
7. **List of Non-functional Requirements -Everyone**

* **Expected Load:** The system will be designed to support up to 30 concurrent users.
* **Response Time:** The maximum response time allowed to the system for transactions will depend on both the number of concurrent users, and the type of transaction. At expected load of 5-10 users, all transactions should take no more than 50 milliseconds. At maximum capacity of 30 concurrent users, creating a new account, logging in, and creating a post should take no more than 500 milliseconds, searching for posts should take no more than 250 milliseconds, both sending and receiving chatroom messages should take 125 milliseconds for a total delay of 250 milliseconds between when a user sends a message, and the other participants receive it. Beyond 30 users, response times are not guaranteed.
* **Reliability:** The final system will allow no more than 30 minutes mean downtime per month. This will be used for fixing issues and making large updates that cannot be done while the system is functional.
* **Privacy:** The system will comply with all privacy regulations in the locations it is intended to operate
* **Ease of use:** The site should be designed to be intuitive and require a few minutes at most for users to become familiar with its features.

1. **High-level System Architecture**
   1. LAMP Server - <https://lamp.cse.fau.edu/~cen4010_fa21_g10/index.html> The FAU lamp server will host both our team site and our project site.
   2. Hyper Text Mark-up Language - will be used to display the application in the web browser.
   3. Cascading Style Sheets - will be used to style the HTML.
   4. JavaScript - will be the language used for client-side functionality.
   5. Bootstrap - will be used for additional styling web page structure.
   6. React – will be used for to further enhance the user experience with an interactive UI.
   7. Python Flask – will be used for server-side programming.
   8. MySQL – will be used for the database to maintain user posts/images/user accounts.
   9. Socket IO –Will be used for real time bi-directional communication between client/server.
   10. Visual Studio Code - will serve as the primary code editor.
   11. Notepad++ - will be used to hold code snippets or command line commands.
   12. Photoshop – will be used to edit or create images as needed for the webpage
   13. GitHub – will be used to maintain code and version control
   14. Jira- will be used to track development as the project progresses.
   15. Whatsapp – will serve as the primary communication tool for the team.
   16. Webex – will be used to host meetings.
   17. Web browsers – various web browsers will be used throughout the development of the project.
2. **Team Roles**
   1. **Team lead, Github Manager, Scrum Master, Front & Back End support**Andrew Sexton
   2. **Front-end developers**

Maxon Corvil

Patrick Messina

* 1. **Back-end developers**

**Back-end lead:** Michael Niebauer

Ashley Davis

1. **Checklist**

|  |  |
| --- | --- |
| Team decided on basic means of communication | **X** |
| Team found a time slot to meet outside of the class | **X** |
| Front and back end team leads chosen |  |
| Github master chosen | **X** |
| Team ready and able to use the chosen back and front end frameworks |  |
| Skills of each team member defined and known to all | **X** |
| Team lead ensured that all team members read the final M1 and agree/understand it before submission |  |