

Final Project for  
SW Engineering CSC648/848 FALL 2020  
DoReMeet  
Team 02  
Milestone 5

Nimiksha Mahajan ([nmahajan@mail.sfsu.edu](mailto:nmahajan@mail.sfsu.edu)) - Team Lead  
Meet Patel ([mpatel6@mail.sfsu.edu](mailto:mpatel6@mail.sfsu.edu)) - Backend & Database Lead  
Luke Chang-Knezevich ([lchangknezevich@mail.sfsu.edu](mailto:lchangknezevich@mail.sfsu.edu)) - Frontend Lead  
Mike Bolanos ([mbolanos1@mail.sfsu.edu](mailto:mbolanos1@mail.sfsu.edu)) - GitHub Master & Backend Engineer  
Jungsun Eoh ([jeoh@mail.sfsu.edu](mailto:jeoh@mail.sfsu.edu)) - Database & Backend Engineer  
Vincent Tse ([vtse3@mail.sfsu.edu](mailto:vtse3@mail.sfsu.edu)) - Server Admin & Frontend Engineer

**Product URL:** <http://ec2-13-52-247-220.us-west-1.compute.amazonaws.com/>

**Date:** 12/17/2020

**History Table**

<b>Version</b>	<b>Date</b>	<b>Comments</b>
M5V1	12/17/2020	Initial Submission
M4V2	12/15/2020	Submission updated with Feedback
M4V1	12/10/2020	Initial Submission
M3V2	12/05/2020	Submission updated with Feedback
M3V1	11/19/2020	Initial Submission
M2V2	12/05/2020	Submission updated with Feedback
M2V1	11/01/2020	Initial Submission
M1V2	10/11/2020	Submission updated with Feedback
M1V1	09/30/2020	Initial Submission

## **Table Of Contents**

1. Product Summary	2
2. Milestone Documents	4
a. Milestone 1	4
b. Milestone 2	37
c. Milestone 3	79
d. Milestone 4	108
3. Final Product Screenshots	145
4. Key Database Table Screenshots	169
5. Task Management System (Monday.com)	173
6. Team Member Contributions	177
7. Post Analysis	190

# 1. Product Summary

**Product Name:** DoReMeet

Our product, DoReMeet, aims to help artists build a community by providing them a platform where they can find collaborators, mentors, and/or friends. Artists, be it singers, dancers, painters, photographers or film makers, can use our product to find people that are close to their geographical location based on their personal preferences. They can choose to connect with fellow artists and collaborate on projects for work, fun or both.

The functions to be delivered in our final product are as follow:

1. Sign Up/Register - Unregistered users shall be able to sign up or register an account on DoReMeet from the landing page. To sign up, users will have to input their email and some other information like their name, date of birth, gender, art category etc.
2. Log In/LogOut - Registered users have the ability to log into their account and log out at any time using the credentials they signed up with.
3. View Potential Matches - Users will be able to see potential matches based on their preferences. They can choose to “Connect” with the user or “Pass” on them.
4. Filter Potential Matches based on Preferences - Users will be able to set their personal preferences for the art category, meeting in person or online, skill level, age and gender. The potential matches they see will be filtered based on these preferences, thus giving them more targeted search results.
5. Expand and Reduce Search Radius - If the user chooses to meet in person, they have the ability to set a radius limit for their potential matches. They can choose to expand or reduce their search radius, and will be shown potential matches only within the specified distance.
6. Matches - Users will be matched with people who they choose to connect with, if the other person also wants to connect with them. Once connected, the user has the ability to chat with the match and collaborate as they deem fit.
7. Chat - Users will be able to send and receive messages to/from the other users they've matched with.
8. Update Preferences - The user has the ability to update their preferences for art category, meeting (in person or online), minimum and maximum age, gender and

skill level. Updating their preferences will also update their potential match results to meet their new search criteria.

9. Search for Posts - Users also have the ability to search for posts on the Community page based on the Post Title and the Art Category

#### **Unique Features:**

1. Simple Connect / Pass Match Page - A simplified, not overwhelming space for users to pick and choose from artists nearby and around the globe. Unlike our competitors, we don't show users a bulk of fellow artists to pick from; instead we show that one person at a time and users get to directly see that individuals work. They can view the profile, and if interested choose to "Connect" with the user or they can simply "Pass" and will not be matched with the user.
2. Community Page - People will be able to see the Community page, with the recent posts made by the platform users and the Highlights of the month. Only users who have a registered account and are logged in will be able to post to the Community page to share their collaborations.
3. Community highlights - The community page will have a highlights section to show most liked posts for the month, based on user votes. Only registered users will be able to vote on the posts.

**Product URL:** <http://ec2-13-52-247-220.us-west-1.compute.amazonaws.com/>

## 2. Milestone Documents

SW Engineering CSC648/848 FALL 2020

DoReMeet

Team 02

Milestone 1

Nimiksha Mahajan ([nmahajan@mail.sfsu.edu](mailto:nmahajan@mail.sfsu.edu)) - Team Lead

Meet Patel ([mpatel6@mail.sfsu.edu](mailto:mpatel6@mail.sfsu.edu)) - Backend & Database Lead

Luke Chang-Knezevich ([lchangknezevich@mail.sfsu.edu](mailto:lchangknezevich@mail.sfsu.edu)) - Frontend Lead

Mike Bolanos ([mbolanos1@mail.sfsu.edu](mailto:mbolanos1@mail.sfsu.edu)) - GitHub Master & Backend Engineer

Jungsun Eoh ([jeoh@mail.sfsu.edu](mailto:jeoh@mail.sfsu.edu)) - Database & Backend Engineer

Vincent Tse ([vtse3@mail.sfsu.edu](mailto:vtse3@mail.sfsu.edu)) - Server Admin & Frontend Engineer

### History Table

Version	Date	Comments
M1V2	10/11/2020	Submission updated with Feedback
M1V1	09/30/2020	Initial Submission



## Table Of Contents

1. Executive Summary	2
2. Main Use Cases	4
3. Main Data Items and Entities	15
4. Functional Requirements	19
5. Non-Functional Requirements	22
6. Competitive Analysis	25
7. High-level System Architecture and Technologies	29
8. Team Contributions	30
9. Checklist	32

# 1. Executive Summary

Our product, DoReMeet, aims to help artists build a community by providing them a platform where they can find collaborators, mentors, and/or friends. Artists, be it singers, dancers, painters, photographers or film makers, can use our product to find people that are close to their geographical location and connect with them, to work on projects for work, fun or both.

Our vision is to extend the practice of instinctual decision-making by breaking down complex entities into a simple matter of yes or no via swipes to forums other than dating. We live a fast paced life, where practically everything is available at the touch of a finger. However, there are few reliable platforms that directly offer this provision for finding fellow artists online. With DoReMeet, users will have the ability to find people either within a 100 miles (150 miles for premium account users) radius of their current location or set up a ‘Meet Online Only’ account. They can “like” individuals (thus marking an individual as *interested*) if they want to connect with the person or “pass” (marking the individual as *not interested*) if they don’t think they’d be a right match.

The users, once registered, will be able to build their profile by adding photos, videos, links to their work, and a bio, which would be a little blurb about them, their work and what they are seeking on DoReMeet. Also, our product offers users’ the ability to add labels/tags to highlight what they are looking for, and this helps deliver more targeted results by filtering the potential matches to those that align with the users’ interests.

These tags break down the user base based on preferences, in terms of interaction criteria (meet in person or meet virtually), work type (passion project, professional work), skill level (novice, competent, experiences, expert), age, and art type (painting, dancing, music, photography, cinema). If the user also wants to emphasize anything in particular about what they are looking for (for instance if it is paid or unpaid work or if there is a timeline to the project) they can add custom tags on their profile. Once a user does find a match that they want to work with, they have the ability to interact with them via chat functionality, where they can share information, pictures and videos of their work. Honing in on the aim to build a community for artists, users can share the collaborations they do with people they meet on DoReMeet on its Community page and showcase their skills.

The source of income for DoReMeet will be advertisements. The advertising model is based on contextual advertising, which is considered to be the least intrusive form of advertising. When the website is opened, users will see small advertisements with keywords that are relevant to the ad underlined on a page. Visitors have the option of mousing over the word and clicking if they want to. We will integrate keywords into our content and based on the users’ preference, they’ll see more targeted ads. For instance, if someone is interested in music, violin in particular, they’d likely see ads from music stores that sell violins.

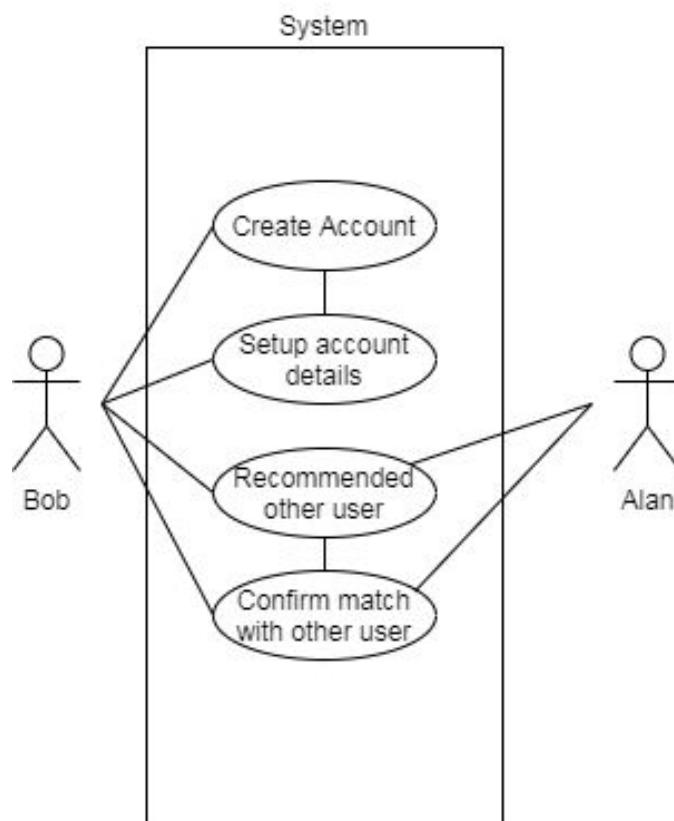
Additionally, to monetize the product we’d add certain paid features as a part of “Premium” membership. Members can choose to pay a small monthly fee to become Premium members; with the regular membership the number of likes per day is limited to a 100, but premium members will have unlimited number of likes. With this membership, users’ profiles will be highlighted,

signifying that they are serious in their search, and they'll get an increased size limit of 120mb for their portfolio (as opposed to the regular 80mb). They'll also be able to extend their search radius for upto 150 miles, giving them access to a larger pool of potential matches. Additionally, they'll be able to 'Rewind' on potential matches, if they skipped someone by mistake and would like to actually connect with them. Premium members will not see any ads, and will have a very customized, streamlined and uninterrupted experience.

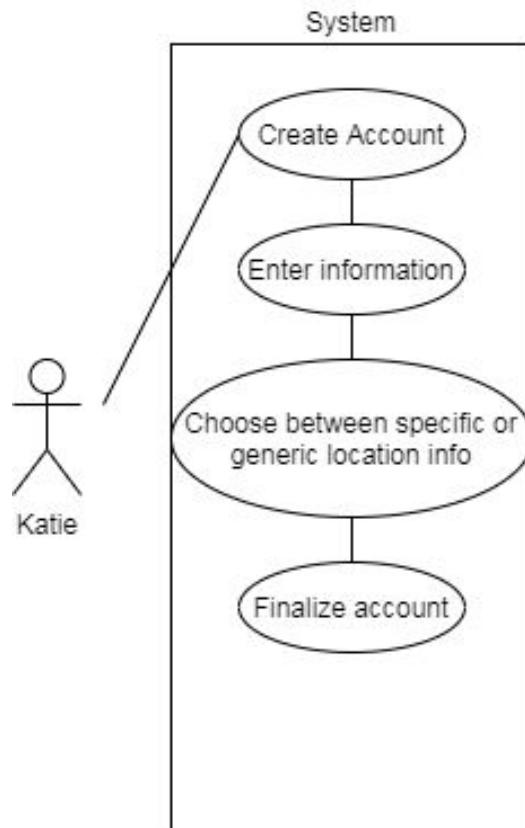
We are enhancing the model of connecting with other people by providing a platform geared towards artists that can not only be personalized based on their needs, but is also effective, safe, reliable, with a fun and creative touch to it. We will market our product based on these features and will appeal to the artist community by offering them a niche service. We will advertise our product on commonly used social media platforms like Facebook, Instagram, Snapchat and TikTok. Also, we'll use Google to advertise our website. In addition to advertising on these platforms, we'll advertise on university campuses by working with their Art, Visual Communications and Film departments, giving their students a platform to showcase their work and encourage collaborations. We'd also extend our platform to the general student body through student ambassadors, who through their own experience will be able to get their peers to join the platform. Our marketing strategies will target demographics between 15 - 25 year olds, and we'll leverage their experiences to build a product that can go "viral".

## 2. Main Use Cases

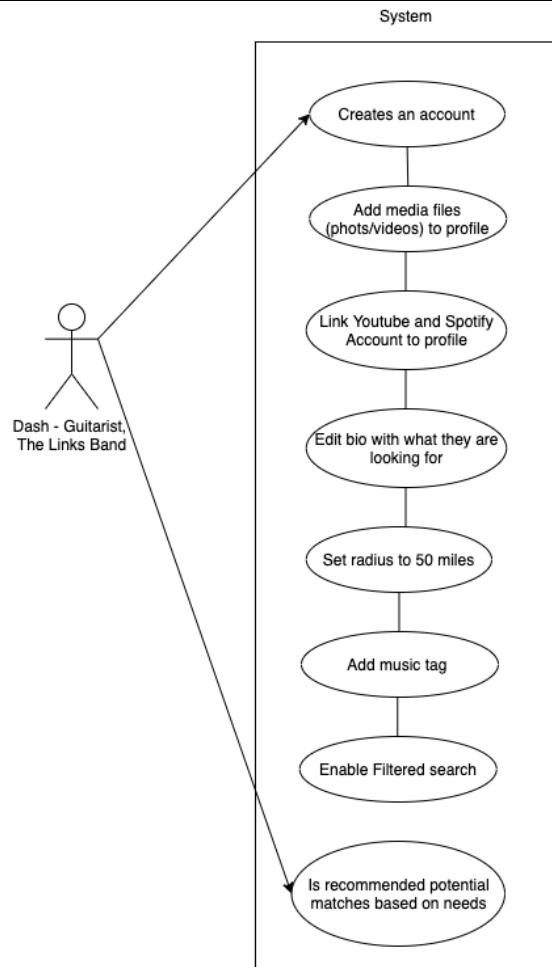
<b>Use Case Title</b>	Forming a duet
<b>Actors</b>	<p>Bob (Tuba player) - Bob is a 70 year old retiree who has recently begun playing the tuba again in his free time. Bob used to play the tuba in highschool and is now rediscovering his passion for the instrument.</p> <p>Alan (Trumpet Player) - Alan is a university student who recently began learning to play the trumpet. The world of music is new and vast to him and he is looking for others to experience it with and maybe even help guide him.</p>
<b>Description</b>	<p>After practicing a bit Bob decides he wants to form a duet with a trumpet player. He begins by creating his account on the site and selecting that he is interested in music. He specifies that he plays the trumpet, and is looking for someone who plays the tuba. Once Bob has finalized his account details he sees several potential matches with similar interests who meet the requirement that he specified. Bob chooses to match with Alan and similarly Alan chooses to match with Bob. Given their mutual wish to match, they are both notified that they have matched with each other and are given the option to chat and exchange details.</p>



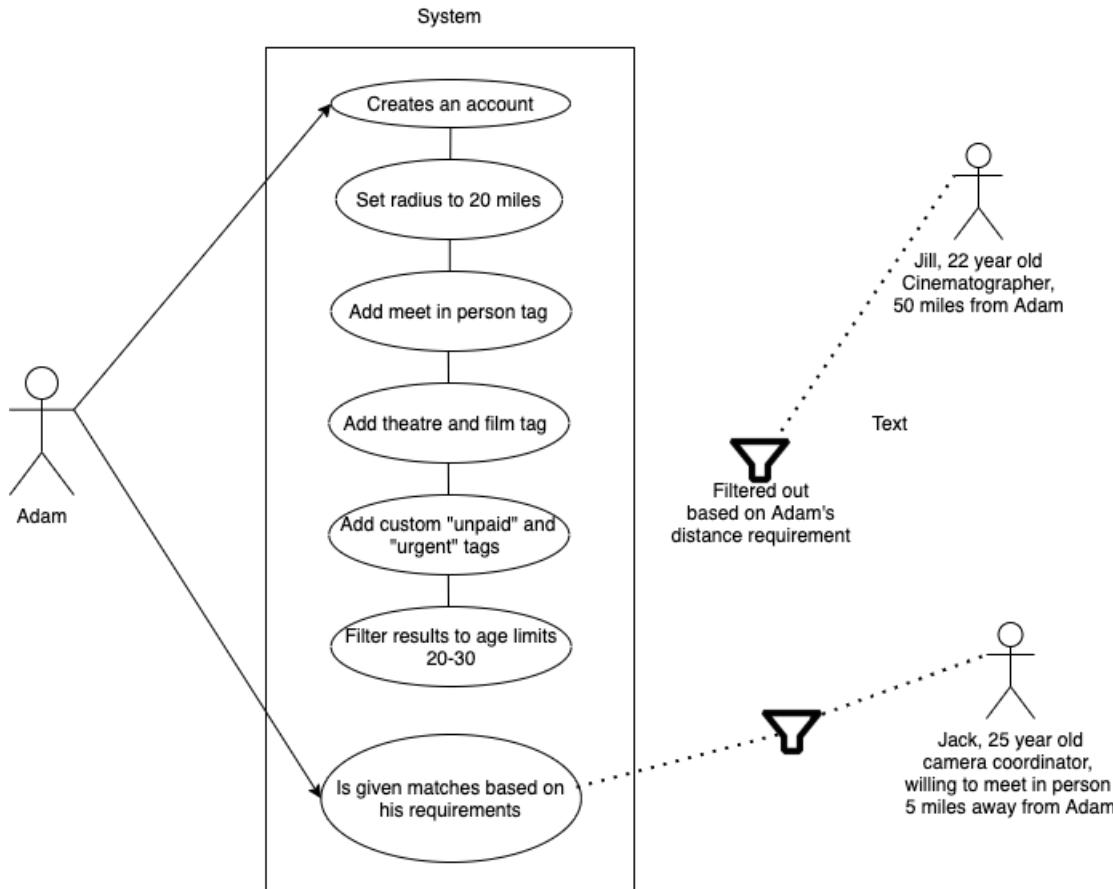
<b>Use Case Title</b>	Unwanted account details - sharing location
<b>Actors</b>	Katie (Painter) - Katie is a young adult living on her own who has a strong interest in painting. She has been painting ever since she was a kid and loves to do so whenever she can find the time to.
<b>Description</b>	<p>Recently Katie has been feeling the urge to do collaborative work with other painters so she decides to sign up for the site to meet new people. She begins the process of creating an account and everything is going smoothly until she gets to the part where she is prompted to enter her location details. Katie feels uncomfortable putting her location details onto the site and would instead prefer that this prompt be left blank.</p> <p>Possible solutions: Have a “meet online-only” category, Have a “general area or city/county category”</p>



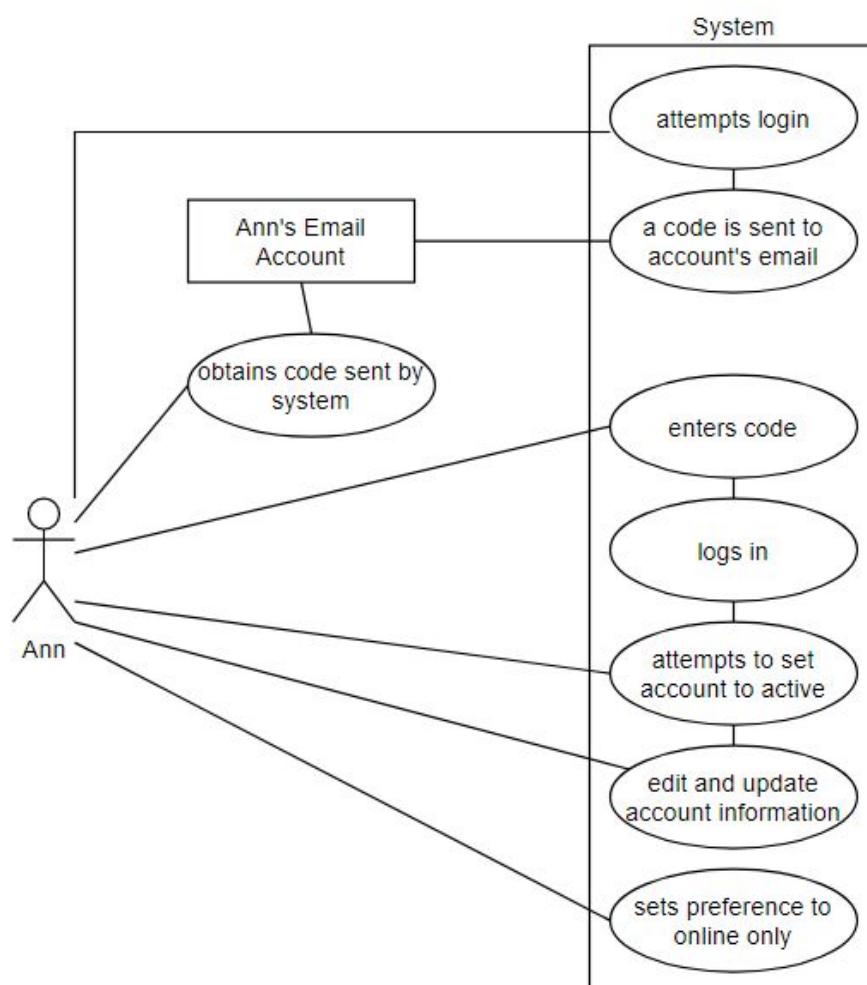
<b>Use Case Title</b>	Band Looking for Manager
<b>Actors</b>	The Links – A band of 4 musicians. Account made by Dash, the guitarist. The Links is an up and coming local rock band from the San Francisco Bay Area who have been doing shows randomly up until now. With the rising fame, they need help to organize their shows and more.
<b>Description</b>	With their rising popularity, the Links need a manager who can help them organize their shows, book more shows and help them build their brand. Given that they are still not as famous or rich, they want to use a free, yet reliable service that can help them find a band manager. They don't want to put the listing on Craigslist and at the same time, they feel that LinkedIn or Indeed will not be the right platform for them, and might not bring them the creative personalities that they are seeking. So they make an account on our website, and set up their profile, with pictures of the band members, photos from their shows, links to their YouTube videos and their Spotify. In their bio they list out what they are looking for, and a "job requirement" so to say for the role they are seeking to fill. They've chosen to restrict their matches to 50 miles, so they only see interested people who are close by and are familiar with the Bay Area. Also, they've set up filters so that they are seeing people who have an interest in music.



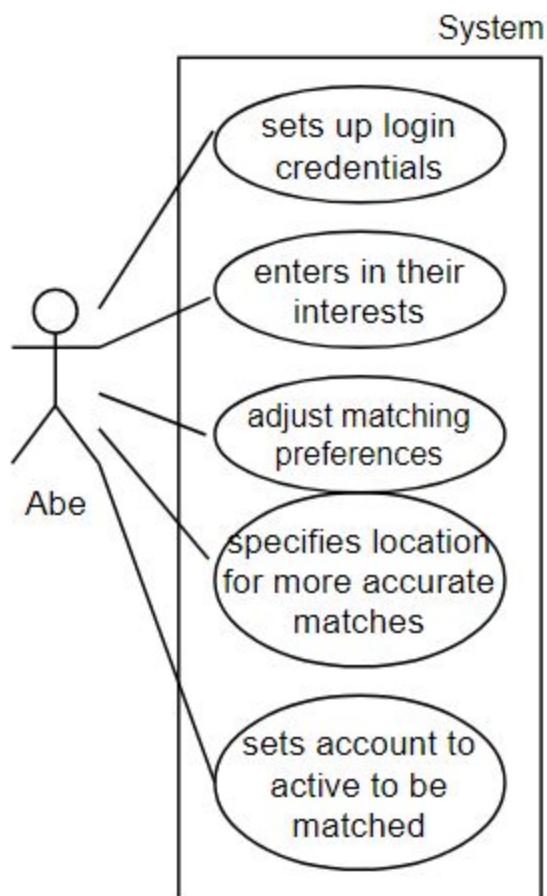
<b>Use Case Title</b>	Director looking to create movie for an Indie Film Festival
<b>Actors</b>	<p>Adam, 22, a senior at SF State University's School of Cinema with a passion for filmmaking.</p> <p>Jack, 25, a freelance cinematographer living in San Francisco, with experience working on Indie films.</p> <p>Jill, 23, a cinematography intern at Oakland Theatre Project and part-time film school student at CSU, East Bay.</p>
<b>Description</b>	<p>Adam wants to submit a short film for a Los Angeles Indie Film Festival but he needs a whole crew, from cameramen, to actors, to script writers and editors. He wants to expand his network and hence didn't want to just make a posting in his university, so he chose to create an account on our website. This gives him access to a bigger film and theatre community and also will help expand his knowledge of the subject. He chose to find people within 20 miles of his university housing so they can all easily meet up and work together. Also, he added tags like "urgent" and "unpaid" so that the people he matches with understand that it is unpaid work and collaboration for just credits and potential success from there on. Additionally, Adam was able to set filters to only get people between the age of 20 - 30, because he wants to make the movie for a niche demographic. Adam sees Jack, who is open to meeting in person and is 5 miles away from him as a potential match. He doesn't however see another cinematographer Jill, who is based 50 miles away from his location, based on the filters he set up.</p>



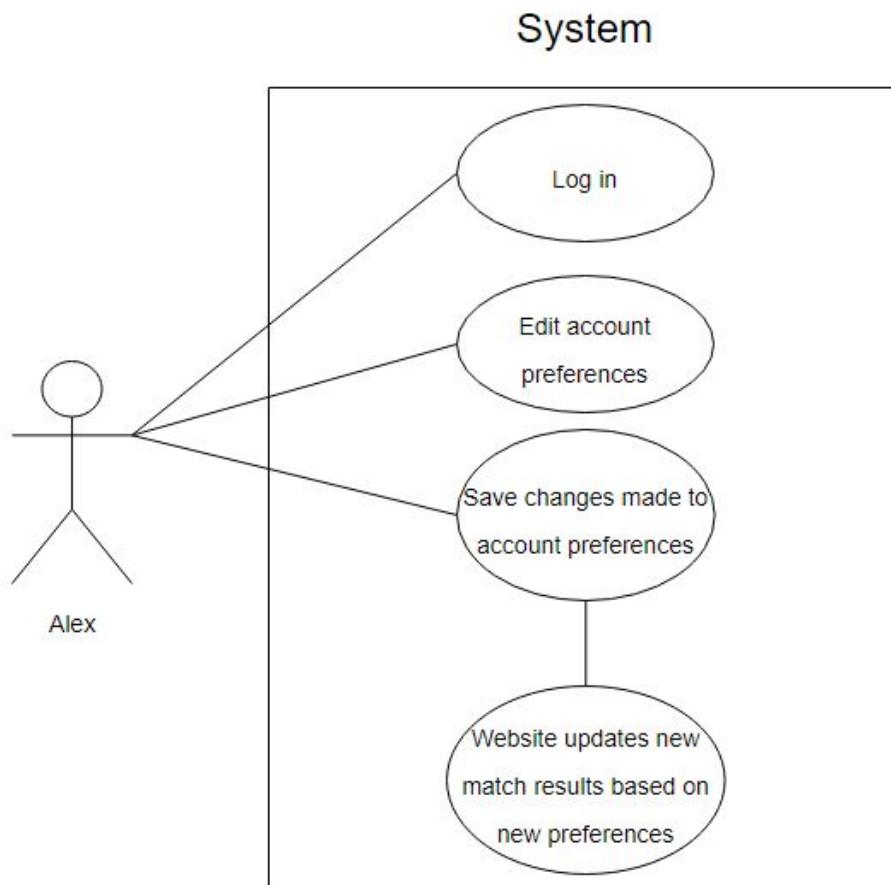
<b>Use Case Title</b>	Returning user
<b>Actors</b>	Ann, a returning registered user who found a new passion playing the bass.
<b>Descriptions</b>	Ann is a returning user. They enter their login credentials into the website. Ann is now given a prompt to enter a numeric code that was sent to their email. They successfully log into their account using the one-time use code and they set their account to active. Before doing that however, the website asked them to verify their current information and edit it as needed. Ann notices that there is an option to not share her location and opt for a meet online only setup. By doing so, Ann can now meet with other artists across the globe, who are not in close geographical vicinity. Also, Ann doesn't have to share information about their location, which makes her feel more secure.



<b>Use Case Title</b>	Account setup
<b>Actors</b>	Abe is an unregistered user looking to making an account, he is looking for other people to share his hobbies with
<b>Descriptions</b>	Abe is trying to find people in their local area that share the same hobbies/interests as them. They find our website that may be able to help them do so, and so make their account. They set their name, email, password, date-of-birth and current country. After this, Abe is now asked to set at least one tag for their account. They also find that they can turn on their location to see people in their area. With this, Abe can customize his search preferences even further. They can also customize their profile by adding a short bio. Abe has finished personalizing his account and sets his account to active.



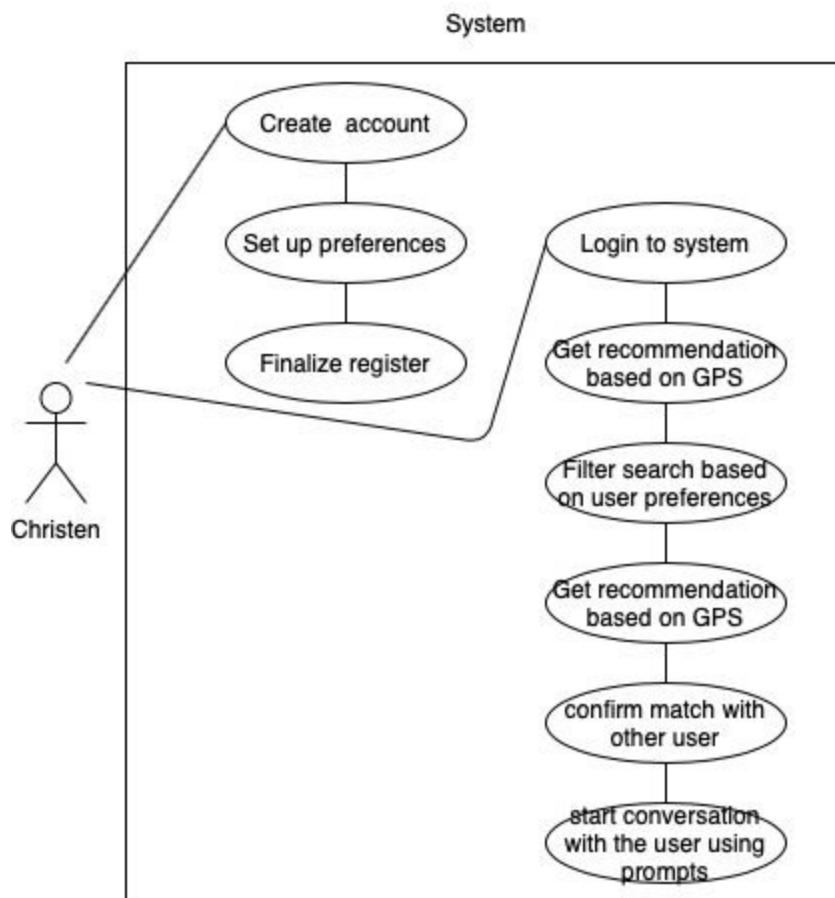
<b>Use Case Title</b>	Updating preferences
<b>Actors</b>	Alex is a long time registered user and has collaborated with other users on the platform many times. He has been collaborating with Pop artists, however, he wants to start exploring different types of genres.
<b>Descriptions</b>	Alex logs into his existing account and changes his potential match preferences from Pop to Open to all music genres. After Alex saves the changes, the website gives Alex new users to match with who have similar preferences to Alex's new preferences.



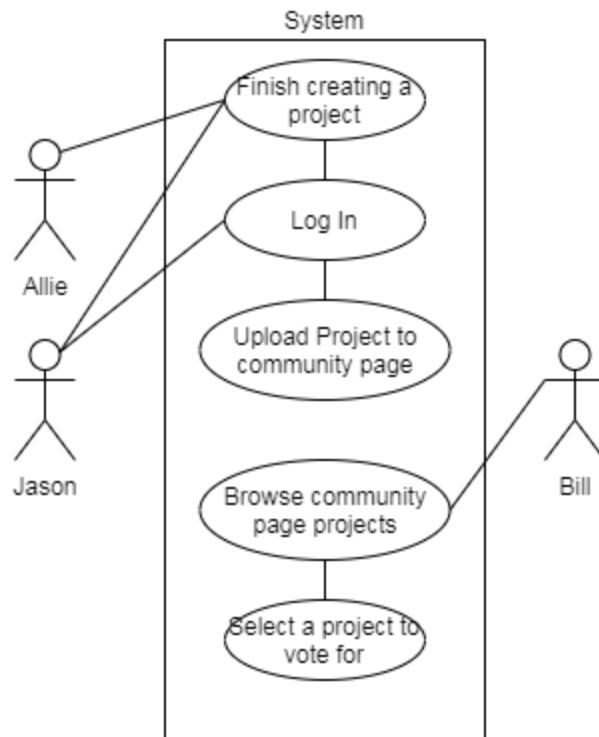
<b>Use Case Title</b>	Reporting a user
<b>Actors</b>	Alex (Registered User) - Alex is a producer and has worked with a few artists before on DoReMeet. He is excited to create more music with other artists. Alan (Registered User) - Alan is a new singer who has only released a couple of songs. Alan has only worked solo so he went on DoReMeet to find a collaborator. John (Admin) - John is a college graduate who has majored in communications. He has just started working at DoReMeet.
<b>Description</b>	Alex and Alan matched through the website and created a song together. However, Alan decided to release the song and did not give any credit to Alex. Alex reports Alan and an email is sent to the admin. John, an admin for the website, investigates Alex's report, and emails Alan giving him a chance to explain his side of the story. Alan responds, but given the lack of a viable justification from him and all the supporting evidence from Alex, John deletes Alan's account. Also, John sends Alan a copy of the website's terms and conditions which he accepted while creating his account, where it clearly states that there can be legal consequences if any party steals/doesn't give credit for any shared collaborations done with connections made on the website.



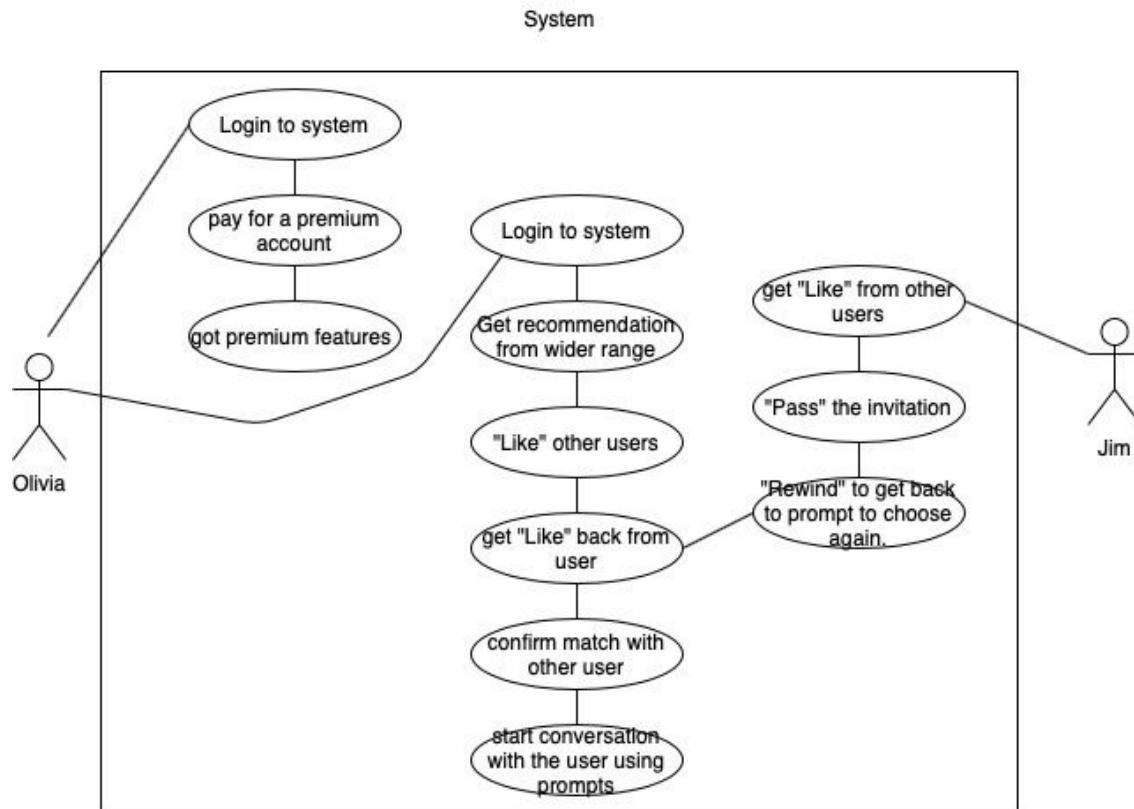
<b>Use Case Title</b>	Set skill level and Conversation Prompts
<b>Actors</b>	Christen: DoReMeet new registered user who is looking for a specific skill level Tango partner.
<b>Description</b>	Christen is interested in Tango, but she doesn't have any prior dance experience. She uses her account on the website to find a dance partner and sets her skill level as a Tango dancer to a novice. She only wants to meet beginner dancers, who she can learn with. She wants to practice her moves without feeling that she is wasting her partner's time. She filters her search based on people who are within 30 miles of her location and are novice, beginner Tango dancers. She also uses the website's conversation prompts to start a conversation with the dancers she matched with. This makes her feel less anxious and awkward about starting the conversation.



<b>Use Case Title</b>	Submitting a project to the community highlights page
<b>Actors</b>	Jason (Trumpet) Jason is a 34 year old accountant who enjoys playing the trumpet Allie (Clarinet) - Allie is a 27 year old beautician enjoys playing the Clarinet Bill (Violin) - Bill is a 60 year old lawyer who plays the violin and likes orchestral soundtracks
<b>Description</b>	Jason and Allie met each other on the site a few weeks ago. They have formed a duet as they both enjoy classical music and have been working on recording a song together. After finishing their song, Allie and Jason decide to post their work to the community highlights page. They navigate to the community page after one of them logs in their account and they begin the process of uploading their work. They upload the mp3 file of their song along with a description of the work. After they have finalized their details they submit the work and the project is uploaded. Bill is browsing the community highlights page and sees Allie and Jason's project. After listening to the mp3 they uploaded he is very impressed and decides to give their project a vote for the best project of the month.



<b>Use Case Title</b>	Premium features
<b>Actors</b>	Olivia: DoReMeet registered user who decided to pay a premium account. She is looking for a dance partner for a Latin dance competition. Jim: DoReMeet registered user who has a premium account, and is also a seasoned amateur dance competition participant.
<b>Description</b>	Olivia is a passionate amateur dancer. She loves all types of dances, but she's especially passionate about Latin dances. She wants to participate in a Latin dance competition. She already has a DoReMeet account, and using her regular account, she has been successful in finding suitable dance partners. She was planning to do it this time as well, but she quickly ran through her limited number of free "likes" for the day. She'd have to wait 24 hours to be able to access her next batch of free likes, but she doesn't want to wait that long. So, she decides to subscribe for a premium membership for the month. With the membership, her profile is highlighted to other users and it makes her stand out from other users. She believes that this will increase traction to her profile, and up her chances of getting likes back. Also, with the regular account, she would sometimes get annoyed by advertisements, which are completely removed from her new premium account. As soon as she pays her premium account, DoReMeet also lets her extend her search radius to 150 miles, so now she can use her unlimited likes for a wider range of users. She is happy with her decision to pay for the premium account. Jim is an amateur competitive dancer, and is also a long time DoReMeet premium account user. He's successfully found dance partners in the past, and is looking for someone to participate in a Latin dance competition with. He sees Olivia's profile and passes on her. But after he passed, he thought they'd be a good match, so he rewinds on her profile. Olivia too had liked Jim, and given they both liked each other, they are connected by the platform.



### 3. Main Data Items and Entities

1. **Registered User/User** - A user is any individual who has created an account on the website using their name, a valid email, date of birth and password.
2. **Unregistered User** - Any individual who has not created an account is an unregistered user.
3. **Landing Page** - The first page any individual, both registered and unregistered, will see when they visit the website. The page shall lead to a sign up or login page.
4. **Account** - An account is an entity associated with a user, and the site sees each user in terms of their valid account.
  - a. *Active/Activated account* - An active account is a user account that is in service and the information is stored on our database. An account is active when the user first creates and sets it up. Also, the account is activated after the user unpauses their account.
  - b. *Paused Account/Inactive Account* - Users have the ability to pause their account or deactivate it without deleting their information. This can be done if the user wants to take a break from using the website, but don't want to leave the platform all together.
  - c. *Disabled Account* - Users with an account have the option to delete their account and remove all their information from our database. Such an account is a disabled account.
5. **Premium Account** - A premium account is a paid account which in addition to all the features a regular account offers, offers additional features.
  - a. *Premium Features* - A premium account includes premium features including unlimited number of likes, extended search radius to 150 miles, increased portfolio size (120mb), highlighted user profile, ability to rewind on a passed potential match and no ads while using the website.
  - b. *Rewind* - As a part of their premium membership, users can "rewind" on their last potential match. This means that if they passed on a user, but didn't mean to, or they changed their mind, they have the ability to go back and look at the profile again, and like or pass as they wish.
6. **Profile** - A profile is the user's information on the website. They can set it up in any way they like, displaying whatever information they feel is relevant to them. The site provides means for them to curate their profile and optimize it to gather more traction.
  - a. *Bio* - A bio is a brief text description that the user can add to their profile to describe what they are looking for, or talk more about their work.
  - b. *Photos* - Users have the ability to add pictures of their work or anything relevant that showcases their interest or what they are looking for.
  - c. *Videos* - Similar to photos, users can add short videos to their profile showing their work as well. This is especially useful for musicians, dancers and filmmakers, who will get a chance to showcase a snippet of their work.

- d. *Links* - Users can add links to their website or our portfolio pages on their bio to endorse their work.
  - e. *Linked Accounts* - Users have the ability to link other accounts to our website. For instance, they can link their Instagram, Youtube or Spotify account to show more of their portfolio.
7. **Preferences** - Users can customize their profile by adding preferences to it. These preferences will make their account more personalized and will help them get more targeted potential matches. The users can add a variety of preferences from the given choices:
- a. *Art Categories* - The users will be able pick one of the five art categories - *painting, dancing, music, photography, cinema* - and their profile will get hits in that category only, i.e. dancers will see others dancers while singers will be put in a pool of other singers. This will help users get direct access to the community they are seeking to join.
  - b. *Work Type* - The site provides a platform for both passion projects and work. Users can filter their potential matches based on their preference and choose to work with others on serious projects or just collaborate on a passion project/hobby.
  - c. *Distance* - The users have the option to set distance preferences if they are open to meet other people in person. They can choose any radius between 1 mile to 100 miles (150 miles for premium account users) of their geographical location, and they will see potential matches only in the specified distance. Premium account holders can extend their radius upto 150 miles.
  - d. *Meet online only* - Users also have the option to set their profile preference to meet online only - i.e. they can opt to see only users who want to meet online and collaborate virtually. They don't have to share their location in this case, but will still have the option to filter what countries or cities they want to see the artists from.
  - e. *Gender* - Users will have the option choose artists based on gender (Male, Female, Trans)
  - f. *Age* - Users can also set an age range as a preference criteria for their potential matches. They can choose any range between 18 and 100 years of age. Users have the ability to set *minimum* and *maximum* age filters for their potential matches.
  - g. *Skill* - Users have the ability to set their skill level (novice, competent, experienced, expert) on their profile and can choose to only see potential matches based on their skill level. For instance, they can choose to learn with other novices or work with an expert.
8. **Tags** - Users will be able to add tags to their profile, which highlight key aspects about them, for instance a user is interested in art, portrait painting in particular and wants to meet fellow artists younger than 30 years in 50 miles of his radius. Also this is just a hobby for him, so he can add portraits, hobby, meet in person, as more such things as his tags, which help condense the information on his profile and get it more traction from other users looking for similar things.
- a. *Custom Tags* - Users can create their own custom tags to really highlight what they want. Some instances could be "unpaid", "urgent", "fun", "volunteer partner", "LGBTQ friendly" or anything that they think is relevant to their profile. The custom tags help share more about the user with potential matches and make more fitting connections.
9. **Potential Matches** - Potential Matches are the users that a user sees based on their preferences. The potential matches fit whatever criteria the user has set for art category,

distance, age, skill or gender. The user has the option to either pursue these matches or forego them. These are just people who the system thinks would be a good match with the user.

- a. *Waiting State* - Potential Matches that are liked by one user are in a waiting state till the other user either likes or passes on them.
  - b. *Filtered Results* - The potential matches can be filtered based on the preferences set by the user. For example, if a user sets distance limitations, all potential matches outside that radius will be filtered out.
10. **Likes** - Likes are the potential matches that the user wants to pursue. If the user does want to work with these potential matches and if they like them back, then the two of them will be connected. Regular account holders can like upto 100 people per day, whereas premium account users will have unlimited number of likes.
11. **Pass** - Passing are the potential matches that the user does not want to pursue, because they don't match their interest or they don't think would be a good match for them. Even if these potential matches, that the user passed, likes them back, they will not be connected, given it is not a mutual interest to work together.
12. **Matches/Matched users** - Matches are the potential matches that the user liked and who liked them back. Given their mutual interest in being connected, the site matched the 2 users and provided them a platform to interact with text messages, share photos/media etc and collaborate on whatever project they were both seeking to pursue.
13. **Community Page** - The community page holds joint projects that users have uploaded. Users who match with others on the site and go on to create something have the option to upload that project here and browse the projects that others have made. Users can vote on which project they think deserves to be the project of the month.
14. **Highlights of the Month** - The highlights of the month will be community voted projects. Each month the projects with the highest number of votes will be showcased as the highlights of the month for all users.
15. **Report** - A user has the ability to report any potential matches or matches if they engage in any illegal, violent or indecent activity. A reported user will be investigated and can have their account suspended if found guilty.
16. **Flag** - A user has the ability to flag any inappropriate content that they see from any potential matches or matches. The content will be reviewed, and if found in violation of the site's policies will be removed from the site. Also the user who posted it, will be notified with a warning regarding the content.
- a. *Inappropriate Content* - Content that violates our websites Community Guidelines, and/or aims to promote violence, hate or bias is considered inappropriate.

17. **Block** - A user has the ability to block any potential matches or matches if they feel don't want to engage with that person. Any blocked user will not be seen by the user and will not be able to connect with or contact the user.
18. **Unmatch** - A user can choose to unmatched with any of their matches, if they realize they don't want to pursue that connection, or don't want to stay connected for any reason. An unmatched user is not blocked, and can still appear in the users potential matches.
19. **Messaging/Chat Window** - When connected with a user (a potential match who the user likes and who liked them back), the user can connect with the match using a messaging or chat window, where they can share information via texts, photos or short videos.
20. **Conversation prompts** - The chat window gives initial conversion prompts when the users are first matched. This is to promote engagements and help users reduce the anxiety of coming up with a conversation starter. These prompts are suggested to promote engaging and productive conversations between the matches.
21. **Password** - The user needs to set a password when they create an account. The user then uses this password to log into the website from any device or browser. The password needs to meet certain security requirements (password length, special characters etc. as per the standard), to ensure that it is a strong password that cannot be easily cracked.
  - a. *Forgotten Password* - The user might forget their password, and they have the ability to say they forgot their password, at which point they'll be given a special link to reset their password via email.
  - b. *Reset Password* - The user can choose to reset their existing password (even if it is not forgotten) and will have the ability to do so.
22. **Administrator/Admin** - An administrator or admin is someone who has access to the site's backend is able to investigate any flagged or reported activities on the website. An admin has the ability to remove any content/user that is found in violation of the website's policies.
23. **Customer Service** - Customer Service is a support team for the website that can help the users with any technical or logistical difficulties and can help them navigate the website better. Also the customer service is able to support the user with any settings, password updates etc.

## **4. Functional Requirements**

### **Registered User (also referred to as user):**

1. Users shall be able to sign in to the website.
2. Users shall be able to sign out from the website.
3. Users shall be able to change their existing password.
4. Users shall be able to troubleshoot login problems.
5. Users shall be able to activate 2 factor identification.
6. Users shall be able to deactivate 2 factor identification.
7. Users shall be able to Like other users.
8. Users shall be able to Pass other users.
9. Users shall be able to view potential matches' profiles.
10. Users shall be able to view their matches' profiles.
11. Users shall be able to report other users.
12. Users shall be able to flag inappropriate content.
13. Users shall be able to block other users.
14. Users shall be able to vote on collaborations posted on the community page.

### **Unregistered User:**

15. An unregistered user shall be able to create an account using a valid email.
16. An unregistered user shall be able to only see the landing page of the website.
17. Unregistered users shall have to confirm their account registration via a valid email.
18. An unregistered user shall be required to enter a date of birth upon account creation.

### **Account:**

19. An account shall have one valid email linked to it.
20. An account shall be paused by the user anytime they want.
21. An account shall allow users to login while it is paused.
22. An account shall be disabled by users through the settings page.
23. An account shall be reactivated by the user anytime they want.
24. An account shall be active to allow users to continue matching with others.
25. An account shall allow its user to modify the password.

### **Premium Account:**

26. A premium account users' profile shall be highlighted.
27. A premium account users' shall have access to premium features.
28. A premium account user shall have to pay for premium features.
29. A premium account user shall be able to save their payment information.
30. A premium account user shall be able to update their payment information.
31. A premium account user shall be able to cancel their subscription.

### **Profile:**

32. A profile shall allow users to upload pictures.

33. A profile shall allow users to delete pictures.
34. A profile shall allow users to upload text descriptions as bios.
35. A profile shall allow users to update their bio.
36. A profile shall allow users to delete their bio.
37. A profile shall allow users to upload short videos.
38. A profile shall allow users to link their Youtube account.
39. A profile shall allow users to link their Spotify account.
40. A profile shall allow users to link their Instagram account.
41. A profile shall allow users to link any websites.
42. A profile shall allow users to change what is displayed publicly on their account.

**Preferences:**

43. Preferences shall allow users to filter potential matches based on art categories.
44. Preferences shall enable users to choose if they want to meet online.
45. Preferences shall enable users to choose if they want to meet in person.
46. Preferences shall enable users to share their location.
47. Preferences shall enable users to change their match preferences to online only.
48. Preferences shall enable users to change their match preferences to local only.
49. Preferences shall let users set a location radius for meeting in person.
50. Preferences shall let users filter by country if they choose to meet online.
51. Preferences shall enable users to filter by city if they choose to meet online.
52. Preferences shall enable users to adjust preference of minimum age for potential matches.
53. Preferences shall enable users to adjust preference of maximum age for potential matches.
54. Preferences shall let users to filter match preferences based on gender.
55. Preferences shall let users to set skill levels.
56. Preferences shall allow users to update their skill levels.
57. Preferences shall let users to filter potential matches based on skill levels.

**Community Page:**

58. The community page shall contain work that has been created by a collaboration of users.
59. The community page shall contain highlights of the month.

**Highlights of the Month:**

60. The Highlights of the Month shall contain one work that has the most votes for Uniqueness/Creativity.
61. The Highlights of the Month shall contain one work that has the most votes for Effort.
62. The Highlights of the Month shall contain one work that has the most votes for Quality.

**Admin/Administrator:**

63. Administrators shall be able to see a flagged user's account.
64. Administrators shall be able to see a reported user's account.
65. Administrators shall be able to delete a user's account.
66. Administrators shall notify a flagged user about any removed content.

**Password:**

67. Passwords shall be updatable via email in case the user forgets their password.

**Tags and Custom Tags:**

68. Tags shall be added to the users profile to highlight preferences.

69. Custom Tags shall allow users to list special requirements.

**Matches:**

70. Matches' profile shall be accessible to the user.

71. Matched users shall be able to communicate via text message.

72. Matches shall be able to share photos via chat.

73. Matches shall be able to share videos via chat.

74. Matches shall be able to search within their match list by first name.

75. Matches shall be able to search within their match list matches by username.

76. Matches shall be able to use default conversation prompts in chat.

77. Matches shall be able to unmatched their existing matches.

78. Matches shall be made when both users like each other.

**Potential Matches:**

79. Potential Matches shall be liked by both users to become a match.

80. Potential Match shall be removed if a user passes on them.

81. Potential matches shall be matched when both users have liked.

**Chat Window:**

82. Chat window shall show a list of all the conversations between matches.

83. Chat window shall allow users to sort matches by name.

84. Chat window shall allow users to sort matches chronologically.

85. Chats with each match shall have the option to interact via audio call.

86. Chats with each match shall have the option to interact via video call.

**Customer Service:**

87. Customer Service shall be able to chat with users.

## 5. Non-Functional Requirements

### **Security:**

1. Accounts shall optionally have 2 factor identification enabled.
2. User's password shall be encrypted in the database.
3. The site shall verify the users' credentials before successfully logging them in.
4. The website will have a SSL certificate (<https://>).
5. The website shall have OAuth 2.0 implemented.
6. Accounts shall only have one password. There shall be no master key.
7. The website shall warn users when clicking on links that take them to another website.
8. Premium account users' payment information shall be encrypted in the database.

### **Storage:**

9. Each account shall be able to store up to 80mb of associated content.
10. Each premium account shall be able to store up to 120mb of associated content.
11. A single media shall be less than or equal to 10mb.
12. User's data shall be saved into the MYSQL database.
13. Users data should be removed from the database when the user deletes their account.
14. User accounts that have not been logged in for 1 year shall be disabled.

### **Availability:**

15. The site shall be up and running at all times excluding system failures and scheduled maintenance.
16. The site shall notify of all scheduled maintenance, 1 week in advance.
17. The users shall be notified via email about all scheduled maintenance, 1 week in advance.
18. Any/all scheduled maintenance shall be scheduled past 10pm PST on Mondays.

### **Functionality:**

19. The website shall be hosted and deployed on the AWS server, EC2 instance.
20. Account details on the database shall be modifiable.
21. User details on the database shall only be modified by the user themselves.
22. Only flagged user details, that have been reviewed, shall be modified by the admin.
23. The user shall log in using either email or username.
24. The user shall be required to enter both password and email/username to log in.
25. The user shall need a valid email to create an account.
26. Users should be minimum 18 years of age to create an account.
27. The users' password shall be censored on the sign in page.
28. The users' password shall be censored on the sign up page.
29. The website shall be responsive to user input.
30. A premium account shall accept Visa, MasterCard and American Express.
31. A premium account shall accept GooglePay and ApplePay as a valid payment method.

**Fault Tolerance:**

32. Upon a system crash a global error page shall be displayed for all users trying to access the site.
33. There shall be a backup for the website before any update is pushed.
34. There shall be a backup for the database every week on Tuesday past 10:00pm PST.
35. Error messages shall be relevant to the user (ex. "Site is down at the moment" or "Your account has been suspended" etc.)

**Scalability:**

36. The number of accounts that the system shall be able to handle should be scalable based on consumer demand/traffic.

**Privacy:**

37. Developers and staff shall not have direct access to sensitive user information.
38. Anyone without a registered account shall not be able to view profiles of registered users.
39. The site shall accept cookies/cache only if the user explicitly allows it.
40. Users shall not be able to see other users' password.
41. Users shall not be able to see other users' email.
42. Users shall not be able to see other users' phone number (if provided).
43. Users data shall not be shared with any third-party applications/sites/organizations.
44. Users data shall not be sold to any third-party applications/sites/organizations.
45. The site shall not gather any unnecessary data from users.
46. Users' geolocation shall only be known by the website if they allow it.
47. Premium account users' payment information shall only be visible to that particular user.

**Compatibility:**

48. The site shall be compatible with Safari (version 10.0 - 14.0), Chrome (version 80 - 85.0) and Mozilla Firefox (version 78 - 81) and Edge (version 85).
49. The site shall be compatible with Mac OS (10.12.6 - 10.15), Windows (7 and 10) and Linux.
50. The site shall be compatible with mobile browsers on both iOS and Android.

**Look and Feel:**

51. The site shall maintain a core color scheme.
52. The website shall have a functional navigation bar on every page.
53. The website shall have a user settings page.
54. The website's logo shall direct the user to the main home page.
55. The website shall be mobile-friendly.

**Ease of Use:**

56. The site shall maintain less formal vocabulary and diction.
57. A preview of what the site has to offer will be displayed on the site landing page.
58. Website UI shall be simple and efficient.
59. Website UI shall be easy to use.
60. Website UI shall be intuitive.

61. Users shall be able to find what they need on the website in 5 seconds.

**Coding Standard:**

62. The whole site shall use the same coding standards related to code indentation.

63. All functions shall be modularized.

**Marketing Requirements:**

64. The website shall not make any political marketing campaigns.

65. The website shall have a logo on the navigation bar.

66. The website shall have a logo on the “tab” for site identity.

**Legal Requirements:**

67. The site shall explicitly not endorse any activity that is illegal, according to the U.S law.

68. Access to the website's policies shall be available at the bottom of every webpage.

69. The site shall protect all user data under the privacy laws in the U.S.

70. The site shall be transparent about the use of user data.

71. An unregistered user shall have to agree to the website's terms & conditions to successfully create an account.

**Performance:**

72. Match results shall be filtered in at least 1 second.

73. The website's response time shall be at most 1 second.

74. 2 factor identification code shall be sent to the user's phone in at most 5 seconds.

**Expected load:**

75. The system shall be able to handle up to 1000 users at once.

## 6. Competitive Analysis

Feature / Company	Instagram	Thumbtack	ArtistsNearby	Craigslist	Our Product (DoReMeet)
<b>Main Purpose</b>	Photo and video sharing social networking site	Service that matches customers with local professionals	Service that connects professional and amateur artists to a world-wide community of artists	Classified advertisements website with sections for wanted, services, community service	Platform dedicated for artists to find collaborators, mentors and/or friends.
<b>Major strength</b>	Expansive user base, monetary backing, brand name, easy to use	Expansive database, clean UI, reliable	Global community of artists	First movers advantage, vast user base	Simple model to help artists connect with other artists (without being bogged down by other professions), both for fun and work. Reliable and secure - connecting only those who are mutually interested in collaborating
<b>Weaknesses</b>	Can be generic, web traffic is self-contained	Limiting in terms of sub categories, limited to services for hire	Hard to complete searches, no custom searches/unable to filter results based on preferences	Bad UI, hard to navigate, unreliable and unsecure	Limited options at a time. Can get limiting based on preferences, so user might have to expand their search
<b>Backend/API</b>	Django Python, React Native	PHP with Go and Scala backend services, React	JavaScript/jQuery,	Perl	JavaScript, React. Location API - <a href="https://locationiq.com">locationiq.com</a>
<b>UI</b>	Scroll through photos, videos. Follow other users and view user profiles as a grid or in scroll view	See list of individuals providing services and select based on personal preferences	See list of individuals providing services and select based on personal preferences	See list of goods and services being sold/rented/ offered	See 1 profile at a time and choose between "liking" or "passing" them to continue. Simplified decision making model

<b>Target Audience</b>	General Public (generally younger people)	People providing and seeking a wide range of services	Artists, art-lovers, people looking to hire artists and art buyers and sellers	General Public (looking to buy and sell services and commodities)	Artists (painters, singers, dancers, photographers, filmmakers/theatre artists)
------------------------	---	---	--	---	---

## COMPETITIVE FEATURES TABLE

Feature	Instagram	Thumbtack	ArtistsNearby	Craigslist	Our Product (DoReMeet)
<b>Expand and Shrink Search Radius</b>	-	-	-	-	++
<b>Exclusively targeted towards artists</b>	-	-	-	-	++
<b>Collaboration Highlights</b>	-	-	-	-	+
<b>Community building</b>	+	-	-	-	+
<b>Targeted Matches</b>	-	++	+	-	++
<b>UI</b>	+	+	+	-	++
<b>Ease of Use</b>	+	++	-	-	+

+ feature exists; ++ superior; - does not exist

DoReMeet is a platform where we provide a service for the sole purpose of connecting artists, unlike our competitors which focus on a more general audience. Our platform is unique from our competitors because we offer a simplified, and not overwhelming space for users to pick and choose from artists nearby and around the globe. Unlike our competitors, we don't show users a bulk of fellow artists to pick from; instead we show that one person at a time and they get to directly see that individuals work. Most of DoReMeet's competitors require the user to go to a specific profile to view it, but we make this process simpler by showing the entire profile as a potential match and giving the user the resources needed to make an informed choice, quickly. Also, this limits any negative, unwanted traction to a users' profile, because they only see (and are similarly only shown to) people who match their personal preferences. DoReMeet aims to make targeted connections, based on shared interests, and thus reduces effort on the users' part to find a suitable match.

Additionally, we offer the ability to choose a custom distance to find a potential match i.e. we allow the users to set their search radius anywhere between 1 mile and 100 miles (150 miles for premium account users), and show them potential matches based on their preference. This ability to expand

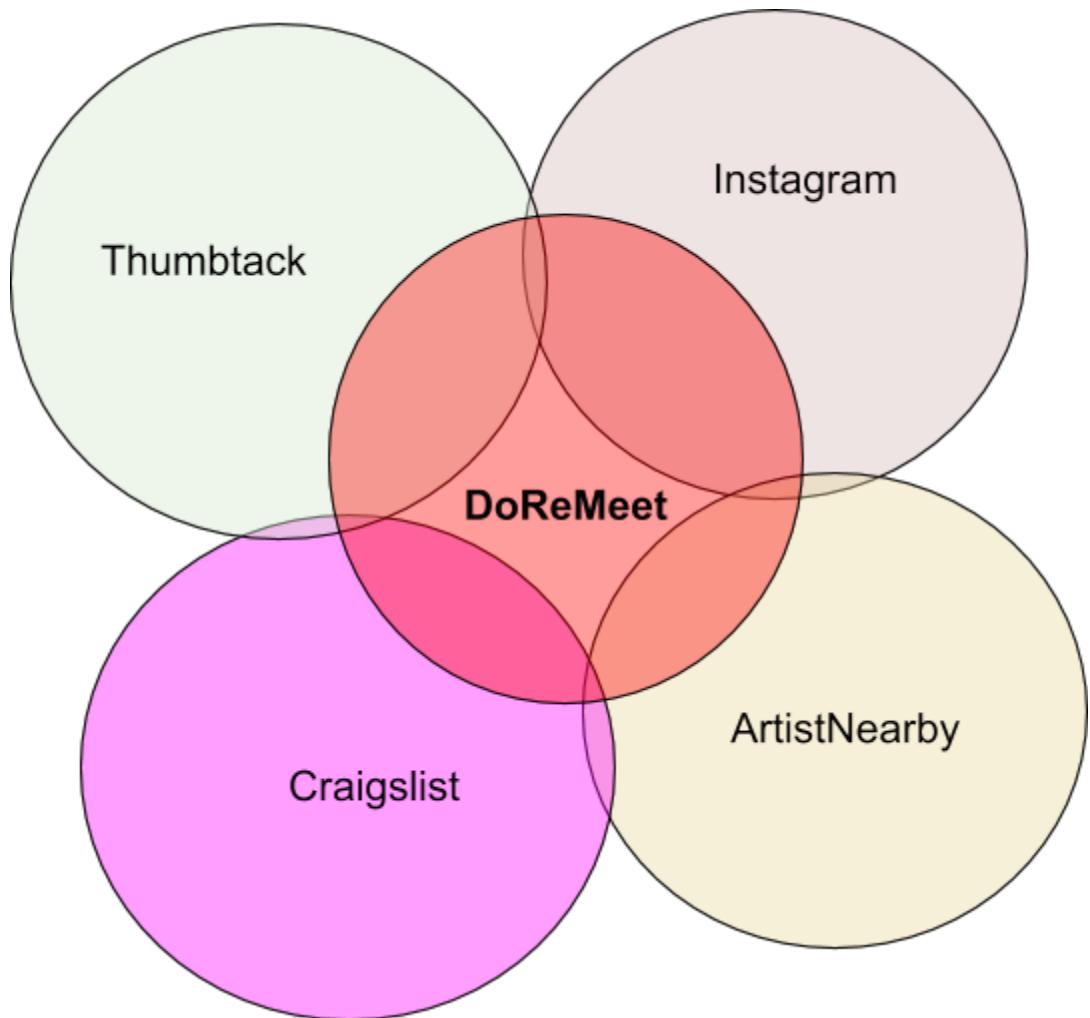
and shrink their search radius makes the matching process more sustainable and impactful. The closest our competitors come to offer distance based search is based on zip code, which generates rather generic results.

We will be using JavaScript to efficiently handle all web page requests, and this will also permit us to get to client input information (client data) more effectively. Any delicate user data will be encrypted for security and privacy. We will also implement OAuth 2.0 as a security measure to strengthen the integrity of a users' account. To implement custom distance, we will use a reliable Location API - [locationiq.com](https://locationiq.com) (services upto 5000 requests/day for free) to provide GPS info, and at no point will the user have to manually input their address/location.

The collaboration page will utilize a good UI and UX in order to provide a seamless and streamlined experience for users to browse the work of other creators. Viewing the uploaded work of others, users will be able to vote on the works that they like the best. These votes will be tallied up at the end of the month and the projects with the most votes shall be showcased accordingly. Utilizing React and JavaScript, this aspect of the site shall be easy to interact with and should build a community on the site that other applications/websites fail to do.

DoReMeet, unlike our competitors, offers a no pressure setting to connect and learn with/from people. The users range from experts to novices and users have the opportunity to find these matches based on their choice, for either a fun collaboration or for serious, paid work. We do not limit the user base to any category, and users are free to switch their preferences any time. In addition DoReMeet also offers a Community page to share collaborations, and give a chance to build an art community. Collaborations are rewarded as "Highlight of the Month" based on their popularity in the community.

In conclusion, while there are services where artists can connect with other artists/people seeking their services, there are no products like DoReMeet which offer quick, reliable and personalized service, specifically for artists.



## 7. High-level System Architecture and Technologies

- **Server Host:** Amazon Web Services (AWS Free Tier EC2 instance)
- **Operating System:** Linux: Ubuntu 18.04 Server
- **Database:** MySQL 8.0
- **Web Server:** Apache HTTP 2.4.46
- **Server-Side Language:** JavaScript
- **Additional Technologies:**
  - Web Framework: React.js, Angular.js, Node.js
  - Other frontend languages: HTML, CSS
- **SSH:** bash
- **SSL Cert:** Let's Encrypt
- **Web Analytics:** Google Analytics
- **IDE:** IntelliJ IDEA
- **Version Control & Collaboration Tool:** Github
- **Documentation:** Google Drive / Google Docs
- **Project Management Tool:** JIRA (Scrum Board)

## 8. Team Contributions

Member	Role	Contributions
Nimiksha	Team Lead and Documentation Lead	<p>Discuss product idea and features, and wrote the executive summary</p> <p>Wrote 2 use cases along with their diagrams</p> <p>Identified and defined the main data items and entities list</p> <p>Contributed to functional and nonfunctional requirements</p> <p>Sort and refine functional and nonfunctional requirements</p> <p>Identify competitors and features; wrote the competitive analysis</p> <p>Added System architecture details</p> <p><i>Post Feedback:</i> Updated Executive Summary, Use Cases, Data Entities, Functional Requirements and Competitive Analysis</p>
Meet	Backend and Database Lead	<p>Discuss product idea and feature to create executive summary</p> <p>Wrote 2 use cases along with the diagrams, however they were merged with others</p> <p>Helped identify main data items and entities</p> <p>Contributed to functional and nonfunctional requirements</p> <p>Sort and refine functional and nonfunctional requirements</p> <p>Identify unique features and do competitive analysis</p> <p><i>Post Feedback:</i> Updated Functional Requirements, Non-Functional Requirements and Competitive Analysis</p>
Luke	Frontend Lead	<p>Discuss product idea and feature to create executive summary</p> <p>Wrote 3 use cases along with their diagrams</p> <p>Helped identify and define main data items and entities</p> <p>Contributed to functional and nonfunctional requirements</p> <p>Sort and refine functional and nonfunctional requirements</p> <p>Identify unique features and do competitive analysis</p> <p><i>Post Feedback:</i> Updated Functional Requirements, Non-Functional Requirements, Use Cases and Competitive Analysis</p>
Mike	GitHub Master and Backend Engineer	<p>Discuss product idea and feature to create executive summary</p> <p>Wrote 2 use cases along with their diagrams</p> <p>Helped identify and define main data items and entities</p> <p>Contributed to functional and nonfunctional requirements</p> <p>Sort and refine functional and nonfunctional requirements</p> <p>Identify unique features and do competitive analysis</p> <p><i>Post Feedback:</i> Updated Use Cases, Functional Requirements and Non-Functional Requirements</p>
Jungsun	Database and Backend Engineer	<p>Discuss product idea and feature to create executive summary</p> <p>Wrote 1 use cases along with its diagrams</p> <p>Contributed to functional and nonfunctional requirements</p> <p>Sort and refine functional and nonfunctional requirements</p> <p>Identify competitors, unique features for our product and do competitive analysis</p> <p><i>Post Feedback:</i> Updated Functional Requirements, Non-Functional Requirements, Use Cases (added new use case) and Competitive Analysis</p>

Vincent	Server Admin and Frontend Engineer	<p>Discuss product idea and feature to create executive summary</p> <p>Wrote 1 use cases along with its diagrams</p> <p>Helped identify and define main data items and entities</p> <p>Contributed to functional and nonfunctional requirements</p> <p>Sort and refine functional and nonfunctional requirements</p> <p>Identify unique features and do competitive analysis</p> <p><i>Post Feedback:</i> Updated Functional Requirements, Non-Functional Requirements and Use Cases</p>
---------	------------------------------------	--

The team worked on all sections of Milestone 1 together during Zoom meetings, and collaboratively created this report.

*Post Feedback:* Similar to Version 1, all work was done collaboratively on Zoom meetings via a shared Google Document.

## 9. Checklist

TASK	STATUS
Team found a time slot to meet outside of the class	DONE
Github master chosen	DONE
Team decided and agreed together on using the listed SW tools and deployment server	DONE
Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing	ON TRACK
Team lead ensured that all team members read the final M1 and agree/understand it before submission	DONE
Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)	DONE

# SW Engineering CSC648/848 FALL 2020

## DoReMeet

### Team 02

#### Milestone 2

Nimiksha Mahajan ([nmahajan@mail.sfsu.edu](mailto:nmahajan@mail.sfsu.edu)) - Team Lead

Meet Patel ([mpatel6@mail.sfsu.edu](mailto:mpatel6@mail.sfsu.edu)) - Backend & Database Lead

Luke Chang-Knezevich ([lchangknezevich@mail.sfsu.edu](mailto:lchangknezevich@mail.sfsu.edu)) - Frontend Lead

Mike Bolanos ([mbolanos1@mail.sfsu.edu](mailto:mbolanos1@mail.sfsu.edu)) - GitHub Master & Backend Engineer

Jungsun Eoh ([jeoh@mail.sfsu.edu](mailto:jeoh@mail.sfsu.edu)) - Database & Backend Engineer

Vincent Tse ([vtse3@mail.sfsu.edu](mailto:vtse3@mail.sfsu.edu)) - Server Admin & Frontend Engineer

#### History Table

Version	Date	Comments
M2V2	12/05/2020	Submission updated with Feedback
M2V1	11/01/2020	Initial Submission
M1V2	10/11/2020	Submission updated with Feedback
M1V1	09/30/2020	Initial Submission

## Table Of Contents

Main Data Items and Entities V2	2
Functional Requirements V2	7
UI Mockups and Storyboards	11
High level database architecture and organization	25
- Database Organization	25
- Media Storage	32
- Search/filter architecture and implementation	32
High Level APIs and Main Algorithms	33
High Level UML Diagrams	35
High Level Application Network and Deployment Diagrams	36
Key risks	37
Project management	39
Team Contributions	40

## 1. Main Data Items and Entities V2

1. **Registered User/User** - A user is any individual who has created an account on the website using their name, a valid email, date of birth and password.
2. **Unregistered User** - Any individual who has not created an account is an unregistered user. An unregistered user is eligible to create a free or premium account.
3. **Landing Page** - The first page any individual, both registered and unregistered, will see when they visit the website. The page shall lead to a sign up or login page.
4. **Account** - An account is an entity associated with a user, and the site sees each user in terms of their valid account.
  - a. *Active/Activated account* - An active account is a user account that is in service and the information is stored on our database. An account is active when the user first creates and sets it up. Also, the account is activated after the user unpauses their account.
  - b. *Paused Account/Inactive Account* - Users have the ability to pause their account or inactivate it without deleting their information. This can be done if the user wants to take a break from using the website, but don't want to leave the platform all together.
  - c. *Disabled Account* - Users with an account have the option to delete their account and remove all their information from our database. Such an account is a disabled account.
5. **Account Type** - Every registered user can have a free account or a premium account. A free account is created by default, but the user has the option to opt for a premium account at a small fee, and access the premium features of our website. A user has the ability to switch between a free account and premium account, and vice versa.
6. **Free Account** - A free account is a regular account that every registered user gets by default. Unlike the premium account, the free account does not have paid features.
7. **Premium Account** - A premium account is a paid account which in addition to all the features a regular account offers, offers additional features.
  - a. *Premium Features* - A premium account includes premium features including unlimited number of likes, extended search radius to 150 miles, increased portfolio size (from 80mb in free account to 120mb), highlighted user profile, ability to rewind on a passed potential match and no ads while using the website.
  - b. *Rewind* - As a part of their premium membership, users can "rewind" on their last potential match. This means that if they passed on a user, but didn't mean to, or they changed their mind, they have the ability to go back and look at the profile again, and like or pass as they wish.

- 8. Profile** - A profile is the user's information on the website. They can set it up in any way they like, displaying whatever information they feel is relevant to them. The site provides means for them to curate their profile and optimize it to gather more traction.
- Bio* - A bio is a brief text description that the user can add to their profile to describe what they are looking for, or talk more about their work.
  - Photos* - Users have the ability to add pictures of their work or anything relevant that showcases their interest or what they are looking for.
  - Videos* - Similar to photos, users can add short videos to their profile showing their work as well. This is especially useful for musicians, dancers and filmmakers, who will get a chance to showcase a snippet of their work.
  - Links* - Users can add links to their website or our portfolio pages on their bio to endorse their work.
  - Linked Accounts* - Users have the ability to link other accounts to our website. For instance, they can link their Instagram, Youtube or Spotify account to show more of their portfolio.
- 9. Preferences** - Users can customize their profile by adding preferences to it. These preferences will make their account more personalized and will help them get more targeted potential matches. The users can add a variety of preferences from the given choices:
- Art Categories* - The users will be able pick one of the five art categories - *painting, dancing, music, photography, cinema* - and their profile will get hits in that category only, i.e. dancers will see others dancers while singers will be put in a pool of other singers. This will help users get direct access to the community they are seeking to join.
  - Work Type* - The site provides a platform for both passion projects and work. Users can filter their potential matches based on their preference and choose to work with others on serious projects or just collaborate on a passion project/hobby.
  - Distance* - The users have the option to set distance preferences if they are open to meet other people in person. They can choose any radius between 1 mile to 100 miles (150 miles for premium account users) of their geographical location, and they will see potential matches only in the specified distance. Premium account holders can extend their radius upto 150 miles.
  - Meet online only* - Users also have the option to set their profile preference to meet online only - i.e. they can opt to see only users who want to meet online and collaborate virtually. They don't have to share their location in this case, but will still have the option to filter what countries or cities they want to see the artists from.
  - Gender* - Users will have the option choose artists based on gender (Male, Female, Trans)
  - Age* - Users can also set an age range as a preference criteria for their potential matches. They can choose any range between 18 and 100 years of age. Users have the ability to set *minimum* and *maximum* age filters for their potential matches.
  - Skill* - Users have the ability to set their skill level (novice, competent, experienced, expert) on their profile and can choose to only see potential matches based on their skill level. For instance, they can choose to learn with other novices or work with an expert.
- 10. Tags** - Users will be able to add tags to their profile, which highlight key aspects about them, for instance a user is interested in art, portrait painting in particular and wants to meet fellow artists

younger than 30 years in 50 miles of his radius. Also this is just a hobby for him, so he can add portraits, hobby, meet in person, as more such things as his tags, which help condense the information on his profile and get it more traction from other users looking for similar things.

- a. *Custom Tags* - Users can create their own custom tags to really highlight what they want. Some instances could be “unpaid”, “urgent”, “fun”, “volunteer partner”, “LGBTQ friendly” or anything that they think is relevant to their profile. The custom tags help share more about the user with potential matches and make more fitting connections.
11. **Potential Matches** - Potential Matches are the users that a user sees based on their preferences. The potential matches fit whatever criteria the user has set for art category, distance, age, skill or gender. The user has the option to either pursue these matches or forego them. These are just people who the system thinks would be a good match with the user.
- a. *Waiting State* - Potential Matches that are liked by one user are in a waiting state till the other user either likes or passes on them.
  - b. *Filtered Results* - The potential matches can be filtered based on the preferences set by the user. For example, if a user sets distance limitations, all potential matches outside that radius will be filtered out.
12. **Likes** - Likes are the potential matches that the user wants to pursue. If the user does want to work with these potential matches and if they like them back, then the two of them will be connected. Regular account holders can like upto 100 people per day, whereas premium account users will have unlimited number of likes.
13. **Pass** - Passing are the potential matches that the user does not want to pursue, because they don't match their interest or they don't think would be a good match for them. Even if these potential matches, that the user passed, likes them back, they will not be connected, given it is not a mutual interest to work together.
14. **Matches/Matched users** - Matches are the potential matches that the user liked and who liked them back. Given their mutual interest in being connected, the site matched the 2 users and provided them a platform to interact with text messages, share photos/media etc and collaborate on whatever project they were both seeking to pursue.
15. **Community Page** - The community page holds joint projects that users have uploaded. Users who match with others on the site and go on to create something have the option to upload that project here and browse the projects that others have made. Users can vote on which project they think deserves to be the project of the month.
16. **Highlights of the Month** - The highlights of the month will be community voted projects. Each month the projects with the highest number of votes will be showcased as the highlights of the month for all users.

17. **Report** - A user has the ability to report any potential matches or matches if they engage in any illegal, violent or indecent activity. A reported user will be investigated and can have their account suspended if found guilty.
18. **Flag** - A user has the ability to flag any inappropriate content that they see from any potential matches or matches. The content will be reviewed, and if found in violation of the site's policies will be removed from the site. Also the user who posted it, will be notified with a warning regarding the content.
  - a. *Inappropriate Content* - Content that violates our websites Community Guidelines, and/or aims to promote violence, hate or bias is considered inappropriate.
19. **Block** - A user has the ability to block any potential matches or matches if they feel don't want to engage with that person. Any blocked user will not be seen by the user and will not be able to connect with or contact the user.
20. **Unmatch** - A user can choose to unmatched with any of their matches, if they realize they don't want to pursue that connection, or don't want to stay connected for any reason. An unmatched user is not blocked, and can still appear in the users potential matches.
21. **Message List** - The message list page shows a list of all the chat messages of a user with all their matches. The list has a little preview showing the users match's name and their text exchanges.
22. **Messaging/Chat Window** - When connected with a user (a potential match who the user likes and who liked them back), the user can connect with the match using a messaging or chat window, where they can share information via texts, photos or short videos.
23. **Conversation prompts** - The chat window gives initial conversion prompts when the users are first matched. This is to promote engagements and help users reduce the anxiety of coming up with a conversation starter. These prompts are suggested to promote engaging and productive conversations between the matches.
24. **Password** - The user needs to set a password when they create an account. The user then uses this password to log into the website from any device or browser. The password needs to meet certain security requirements (password length, special characters etc. as per the standard), to ensure that it is a strong password that cannot be easily cracked.
  - a. *Forgotten Password* - The user might forget their password, and they have the ability to say they forgot their password, at which point they'll be given a special link to reset their password via email.
  - b. *Reset Password* - The user can choose to reset their existing password (even if it is not forgotten) and will have the ability to do so.

25. **Administrator/Admin** - An administrator or admin is someone who has access to the site's backend is able to investigate any flagged or reported activities on the website. An admin has the ability to remove any content/user that is found in violation of the website's policies.
26. **Customer Service** - Customer Service is a support team for the website that can help the users with any technical or logistical difficulties and can help them navigate the website better. Aso the customer service is able to support the user with any settings, password updates etc.

## 2. Functional Requirements V2

### **Priority 1**

#### **Registered User (also referred to as user):**

1. Users shall be able to sign in to the website.
2. Users shall be able to sign out from the website.
3. Users shall be able to change their existing password.
7. Users shall be able to Like other users.
8. Users shall be able to Pass other users.
9. Users shall be able to view potential matches' profiles.
10. Users shall be able to view their matches' profiles.

#### **Unregistered User:**

15. An unregistered user shall be able to create an account using a valid email.
16. An unregistered user shall be able to only see the landing page of the website.
18. An unregistered user shall be required to enter a date of birth upon account creation.
25. An account shall allow its user to modify the password.

#### **Account:**

19. An account shall have one valid email linked to it.
24. An account shall be active to allow users to continue matching with others.

#### **Profile:**

32. A profile shall allow users to upload pictures.
33. A profile shall allow users to delete pictures.
34. A profile shall allow users to upload text descriptions as bios.
35. A profile shall allow users to update their bio.
36. A profile shall allow users to delete their bio.
37. A profile shall allow users to upload short videos.
38. A profile shall allow users to link their Youtube account.
39. A profile shall allow users to link their Spotify account.
40. A profile shall allow users to link their Instagram account.
41. A profile shall allow users to link any websites.
42. A profile shall allow users to change what is displayed publicly on their account.

#### **Preferences:**

43. Preferences shall allow users to filter potential matches based on art categories.
44. Preferences shall enable users to choose if they want to meet online.
45. Preferences shall enable users to choose if they want to meet in person.
46. Preferences shall enable users to share their location.
47. Preferences shall enable users to change their match preferences to online only.
48. Preferences shall enable users to change their match preferences to local only.

49. Preferences shall let users set a location radius for meeting in person.
52. Preferences shall enable users to adjust preference of minimum age for potential matches.
53. Preferences shall enable users to adjust preference of maximum age for potential matches.
54. Preferences shall let users to filter match preferences based on gender.
55. Preferences shall let users to set skill levels.
56. Preferences shall allow users to update their skill levels.
57. Preferences shall let users to filter potential matches based on skill levels.

**Community Page:**

58. The community page shall contain work that has been created by a collaboration of users.
59. The community page shall contain highlights of the month.

**Password:**

67. Passwords shall be updatable via email in case the user forgets their password.

**Matches:**

70. Matches' profile shall be accessible to the user.
71. Matched users shall be able to communicate via text message.
77. Matches shall be able to unmatched their existing matches.
78. Matches shall be made when both users like each other.

**Potential Matches:**

79. Potential Matches shall be liked by both users to become a match.
80. Potential Match shall be removed if a user passes on them.
81. Potential matches shall be matched when both users have liked.

**Chat Window:**

82. Message List shall show a list of all the conversations between matches.

**Priority 2**

**Registered User (also referred to as user):**

11. Users shall be able to report other users.
12. Users shall be able to flag inappropriate content.
13. Users shall be able to block other users.
14. Users shall be able to vote on collaborations posted on the community page.

**Account:**

20. An account shall be paused by the user anytime they want.
21. An account shall allow users to login while it is paused.
22. An account shall be disabled by users through the settings page.
23. An account shall be reactivated by the user anytime they want.

**Premium Account:**

26. A premium account users' profile shall be highlighted.
27. A premium account users' shall have access to premium features.

**Admin/Administrator:**

63. Administrators shall be able to see a flagged user's account.
64. Administrators shall be able to see a reported user's account.
65. Administrators shall be able to delete a user's account.
66. Administrators shall notify a flagged user about any removed content.

**Tags and Custom Tags:**

68. Tags shall be added to the users profile to highlight preferences.
69. Custom Tags shall allow users to list special requirements.

**Matches:**

72. Matches shall be able to share photos via chat.
73. Matches shall be able to share videos via chat.
74. Matches shall be able to search within their match list by first name.
75. Matches shall be able to search within their match list matches by username.

**Chat Window:**

83. Chat window shall allow users to sort matches by name.
84. Chat windows shall allow users to sort matches chronologically.

## **Priority 3**

**Registered User (also referred to as user):**

4. Users shall be able to troubleshoot login problems.
5. Users shall be able to activate 2 factor identification.
6. Users shall be able to deactivate 2 factor identification.

**Unregistered User:**

17. Unregistered users shall have to confirm their account registration via a valid email.

**Premium Account:**

28. A premium account user shall have to pay for premium features.
29. A premium account user shall be able to save their payment information.
30. A premium account user shall be able to update their payment information.
31. A premium account user shall be able to cancel their subscription.

**Preferences:**

50. Preferences shall let users filter by country if they choose to meet online.
51. Preferences shall enable users to filter by city if they choose to meet online.

**Highlights of the Month:**

- 60. The Highlights of the Month shall contain one work that has the most votes for Uniqueness/Creativity.
- 61. The Highlights of the Month shall contain one work that has the most votes for Effort.
- 62. The Highlights of the Month shall contain one work that has the most votes for Quality.

**Matches:**

- 76. Matches shall be able to use default conversation prompts in chat.

**Chat Window:**

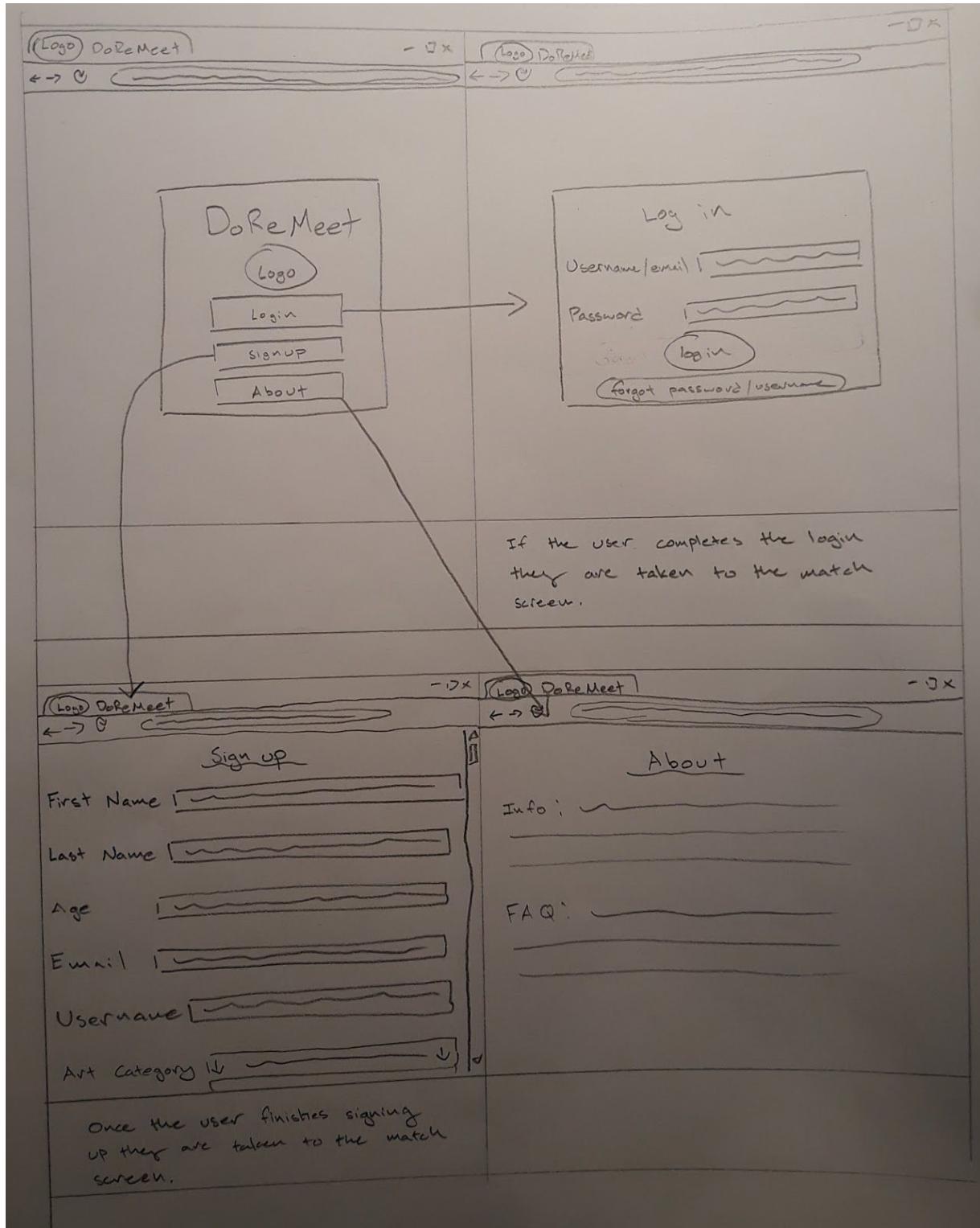
- 85. Chats with each match shall have the option to interact via audio call.
- 86. Chats with each match shall have the option to interact via video call.

**Customer Service:**

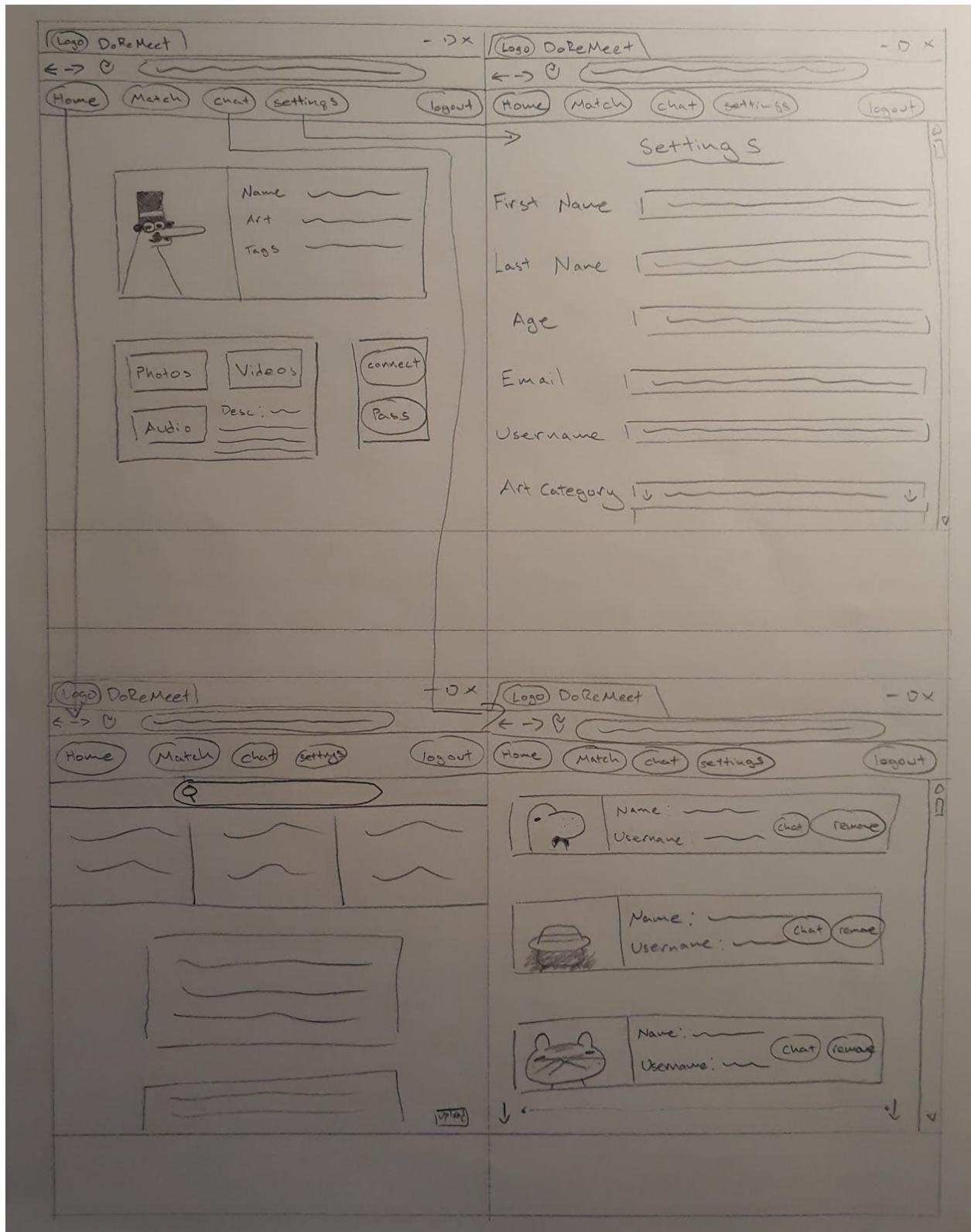
- 87. Customer Service shall be able to chat with users.

### 3. UI Mockups and Storyboards

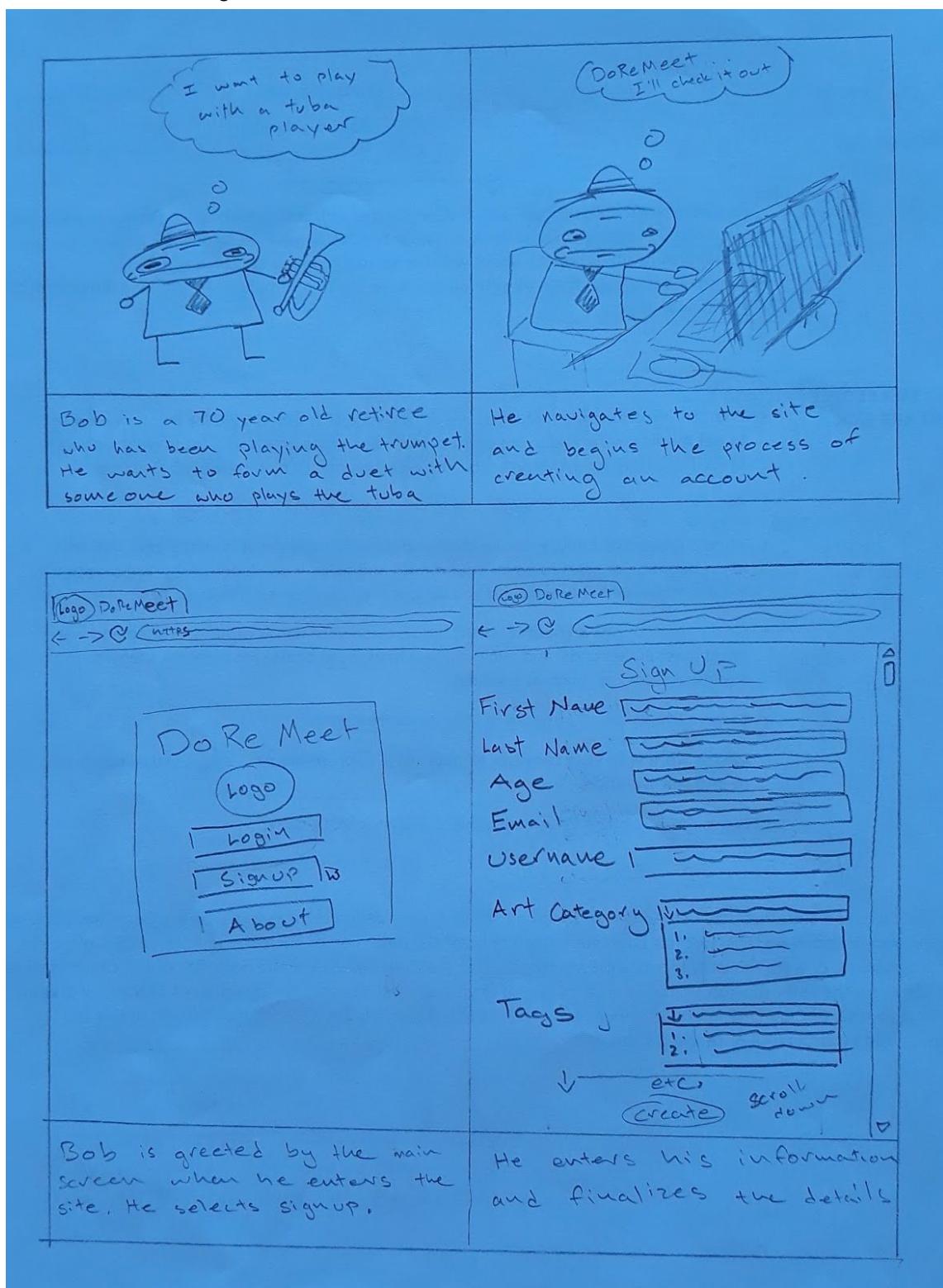
General Site flow (1)



## General Site flow (2)



## Use Case 1: Forming a duet



The wireframe illustrates the user flow from account creation to finding a match.

**Left Panel (Initial Account Creation):**

- Header: DoReMeet
- Navigation: Home, Match, Chat, Settings
- Profile Card: Name: Alan, Art: Music, Tags: Tuber, Desc: me
- Media Options: Photos, Videos, Audio, Desc
- Action Buttons: Connect, Pass

**Right Panel (Matching Process):**

- Header: DoReMeet
- Navigation: Home, Match, Chat, Settings
- Profile Card: Name: Alan, Art: Music, Tags: Tuber, Desc: me
- Media Options: Photos, Videos, Audio, Desc
- Action Buttons: Chat, Remove
- Profile Card: Name: Bob, Art: Music, Tags: Tuber, Desc: me
- Action Buttons: Chat, Remove

**Annotations:**

Bob is taken to the match page after his account has been created. He chooses to connect or pass with the people he matches with.

Bob has chosen to match with Alan, similarly Alan has matched with Bob. Bob clicks the chat button and finds Alan in his chat page.

The wireframe shows a message exchange between Bob and Alan.

**Messages:**

- Bob: Hi Alan want to form a duet? (Bob)
- Alan: Sure thing Bob!
- Bob: cool, lets meet up! (Bob)

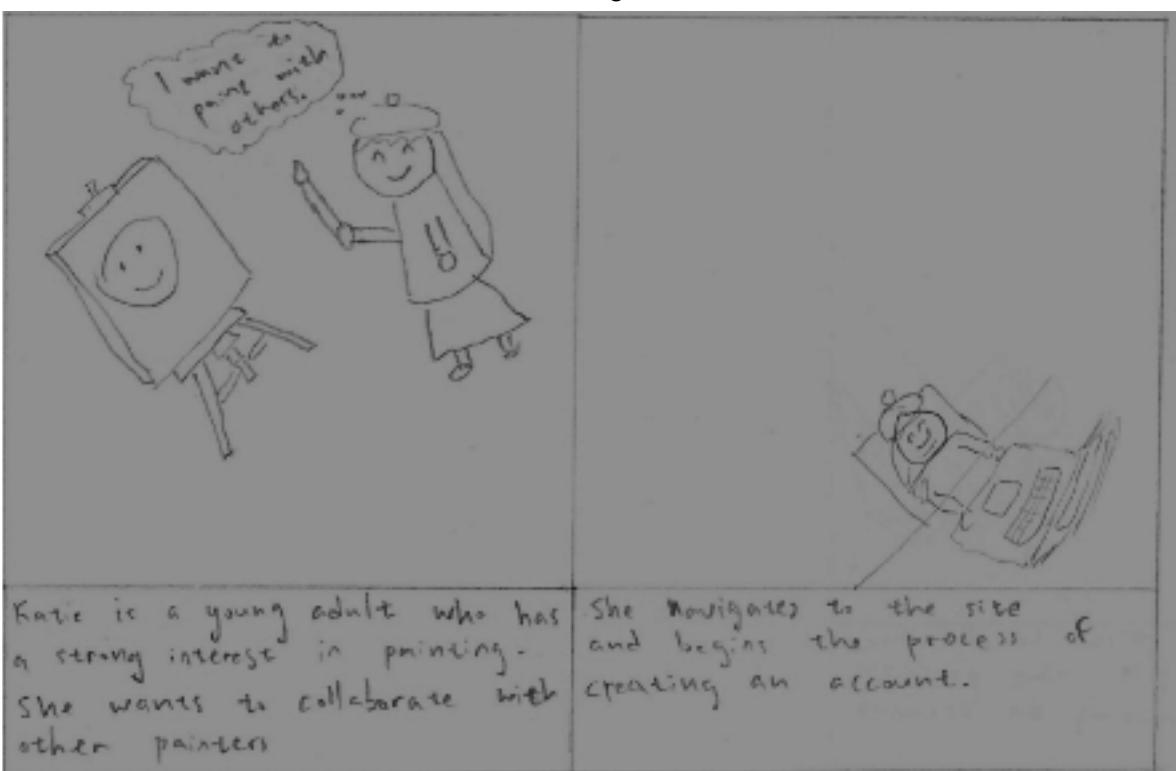
**Text Input:**

< Enter message here >

**Annotation:**

Bob messages Alan and the two of them agree on the details of their meet up.

## Use Case 2: Unwanted account details - sharing location



 DoReMeet	 DoReMeet
<p>Katie is greeted by the main screen when she visits the site. She selects sign up.</p>	<p>Katie enters in her information and leaves her location details blank. She finalizes her account.</p>

### Use Case 3: Band Looking for Manager

The storyboard consists of four panels illustrating a user case:

- Panel 1:** A cartoon illustration of three musicians. One says, "We need to find a manager. Let's use DoReMeet."
- Panel 2:** A "Sign Up" form for DoReMeet. It includes fields for Tags (with a dropdown menu showing 1, 2, 3, 4), skill level (dropdown menu), Photos (with upload and three photo slots), and a description section with two options: "For fun desc." (unchecked) and "For Work desc." (checked).  

The Links, a band of 4 musicians are looking for a manager who can help organize their shows. They choose to use DoReMeet in order to find a creative personality.
- Panel 3:** A continuation of the "Sign Up" process. It shows fields for linking YouTube, Spotify, and other accounts, and a "Search Radius" dropdown menu with options: 1. 10 miles, 2. 20 miles, 3. 30 miles, 4. 40 miles, 5. 50 miles. Below is a checkbox for "meet in person".  

They create their account, adding photos, links to their youtube and spotify, and that they are in the "work" category and a description of the job.
- Panel 4:** A "Match" screen showing account details (Name: [redacted], Art: [redacted], Tags: [redacted], Desc: [redacted]), and sections for Photos, Videos, Audio, Details, Connect, Pass, and Report.  

After entering their information they choose to restrict their search radius to 50 miles.

After finalizing their account details, the Links begin matching with potential managers.

## Use Case 4: Director looking to create movie for an Indie Film Festival

*I want to make a film!*

Adam wants to submit a short film for a LA indie film festival, but he needs a while crew.

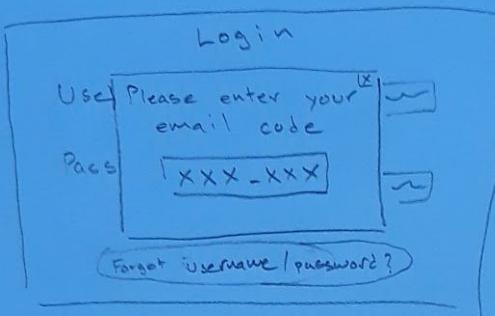
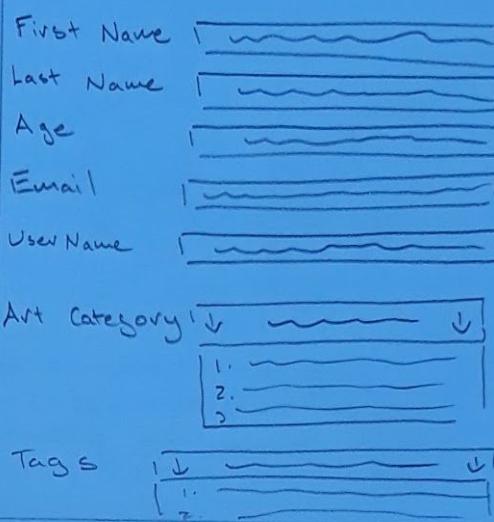
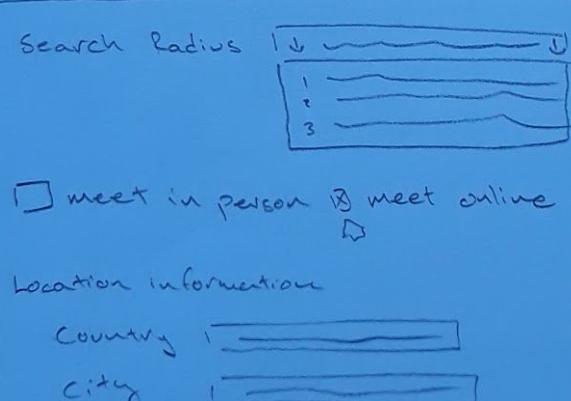
No navigates to the site and begins the process of creating an account.

Adam is greeted by the main screen when he enters the site. He selects sign up.

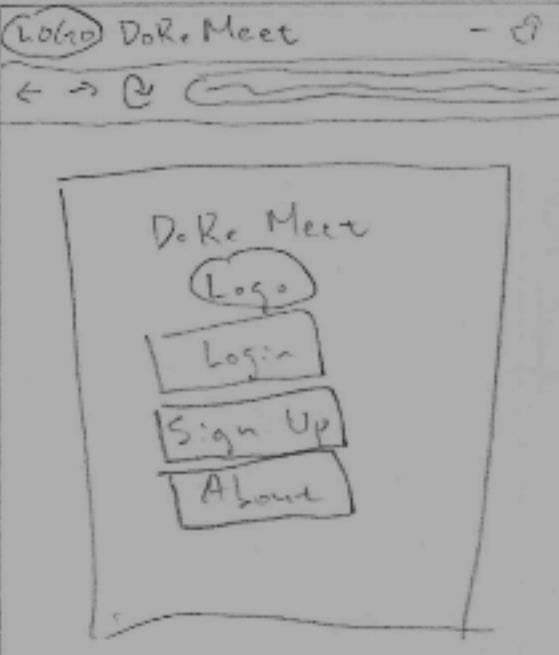
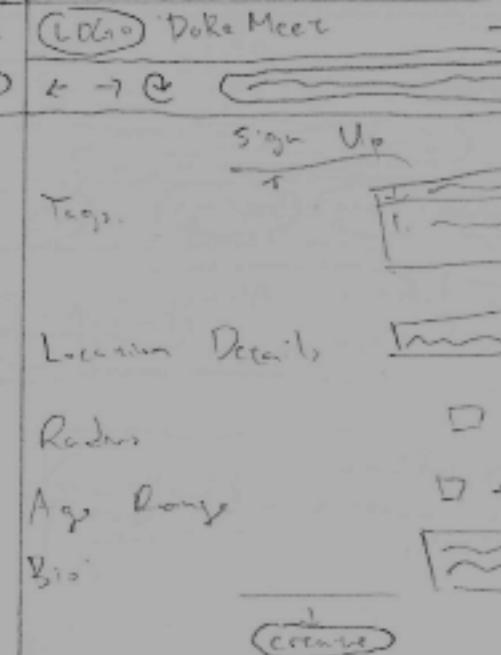
He adds tags "original" and "independent", sets his radius to 20 miles, sets the age range to 20-30 years old, and creates his account.

He sees Tom who is 5 miles away from him.

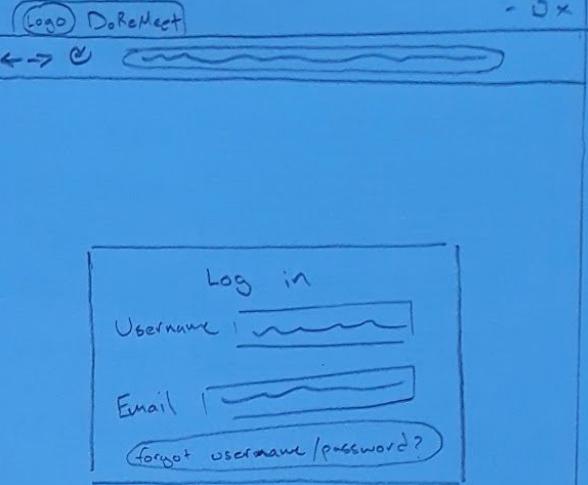
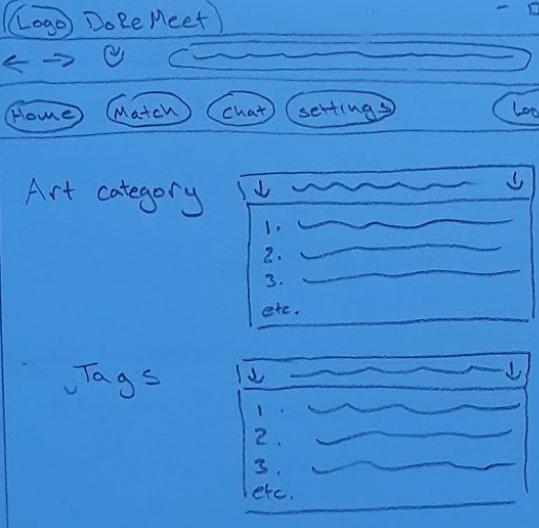
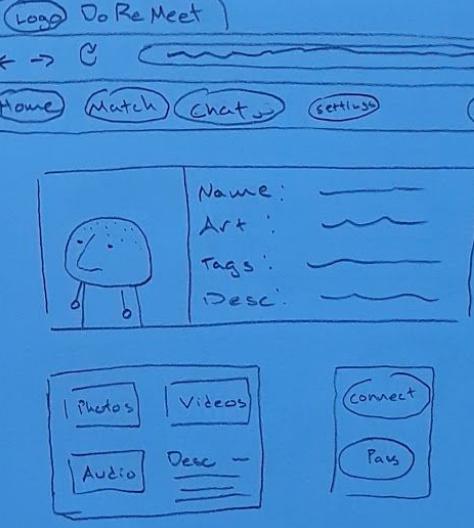
## Use Case 5: Returning user

	 <p>Ann is a returning user who has found a new passion for playing the bass. She attempts to log in and is asked to enter a code that has been emailed to her.</p> <p>Ann enters the code that she received via email and is let into her account. She is asked to verify her information.</p>
 <p><input type="checkbox"/> meet in person <input checked="" type="radio"/> meet online</p> <p>Location information</p> <p>Country: _____</p> <p>City: _____</p> <p>Ann notices that there is a meet online option. She selects it and now feels more comfortable that her location will shared.</p>	 <p>Name: _____</p> <p>Art: _____</p> <p>Tags: _____</p> <p>Desc.: _____</p> <p>Photos Videos</p> <p>Audio Desc.</p> <p>Connect Pass</p> <p>After finalizing her details she can then begin matching with other people who want to meet online.</p>

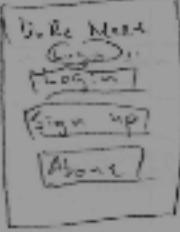
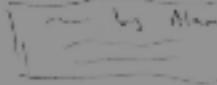
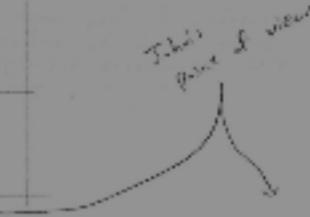
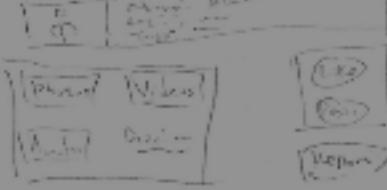
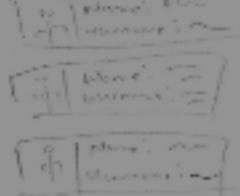
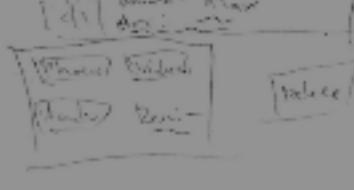
## Use Case 6: Account setup

	
<p>Alzo is trying to find people in his (local) area who share the same hobbies/interests as him.</p>	<p>He navigates to the site and begins the process of creating an account.</p>
 <p>DoRe Meet - ⚡ X ← → ⌂</p> <p>DoRe Meet Logo Login Sign Up About</p>	 <p>DoRe Meet - ⚡ X ← → ⌂</p> <p>Sign Up Tags: <input type="text"/> Location Details: <input type="text"/> Radius: <input type="checkbox"/> miles <input type="checkbox"/> 2.5 <input type="text"/> Age Range: <input type="text"/> Bio: <input type="text"/> <input type="button" value="Create"/></p>
<p>Alzo is greeted by the main screen when he enters the site. He selects sign up</p>	<p>Alzo enters in his information and adds a short bio. He creates his account.</p>

## Use Case 7: Updating preferences

	
<p>Alex is a long time registered user and has collaborated with many Pop artists. Now he wants to start trying different genres.</p>	<p>Alex logs into his account and navigates to the settings page in order to start changing his preferences.</p>
	
<p>Alex changes his art and tag preferences to view ones that he wishes to explore.</p>	<p>After Alex saves his new settings he can begin matching with new people who have similar interests.</p>

## Use Case 8: Reporting a user

		
<p>Alex and Alan watched through the website and created a song together.</p>	<p>Alex clicks login and enters his username and password.</p>	<p>The site takes Alex to the main page after logging in. Alex clicks on the Home button to see the homepage.</p>
		
<p>Alex sees that Alan has released their song on the community page without giving him credit. Alex clicks on the Chat button.</p>	<p>Alex clicks on Alan to see his profile.</p>	
		
<p>Alex reports Alan and an email is sent to John, an admin.</p>	<p>John reads the email and begins an investigation. He logs in to the site and clicks on Alan's profile. John clicks on Report.</p>	<p>After investigating, John clicks on Delete to delete Alan's account.</p>

## Use Case 9: Set skill level and Conversation Prompts

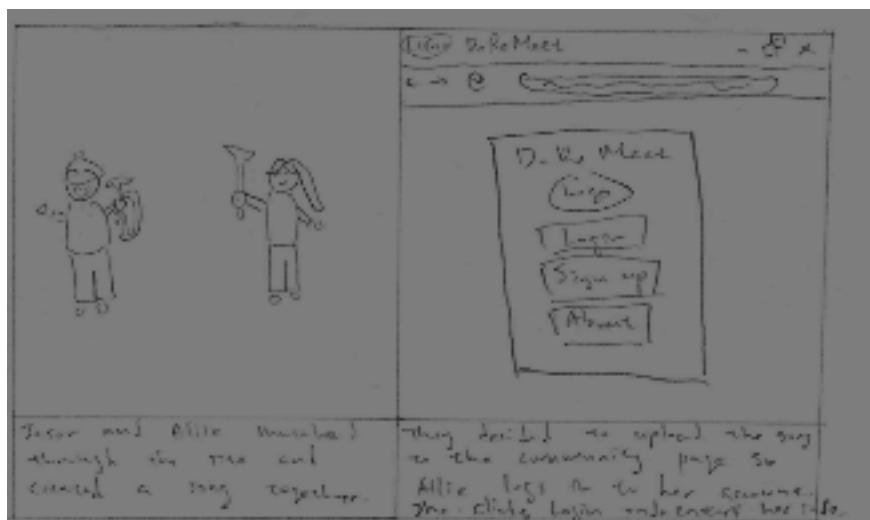
Christen has recently begun learning how to tango. She has no prior dance experience and wants to find a dance partner.

she sets her skill level to novice as she doesn't want to feel there is a large skill gap between her and her potential dance partner.

Christen is then matched with people of a similar skill level based on her preferences.

When starting a conversation Christen uses the conversation prompts to help her feel less anxious.

## Use Case 10: Submitting a project to the community highlights page



she is taken to the main page after logging in. She clicks Home.

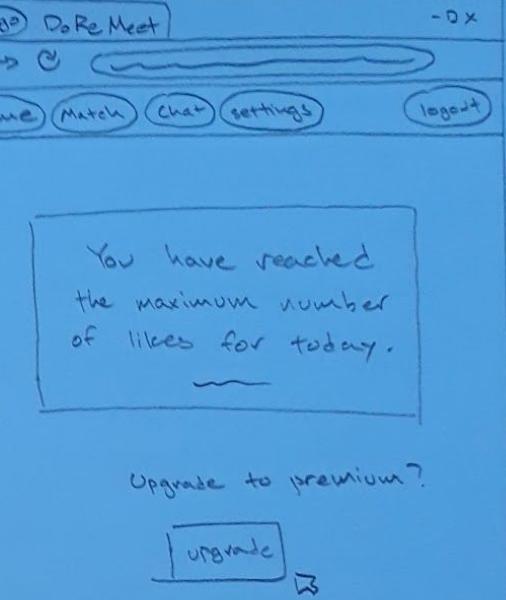
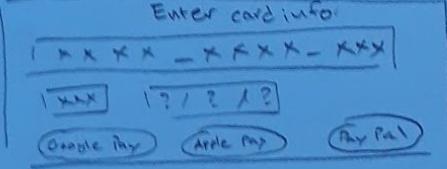
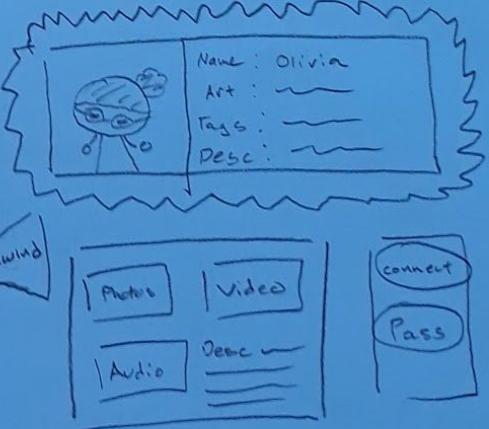
she is taken to the community page. She clicks Upload.

She uploads their mp3 file and enters in the information needed. She clicks Submit.

This is a user that is logged in and viewing the community page. He clicks on Alissa & Jason's song to listen to it.

This is very impressed and decides to vote Alissa & Jason's song for have another at the month.

## Use Case 11: Premium features

	
<p>Olivia is an amateur dancer and is also an avid Do Re Meet user.</p>	<p>Olivia wants to continue matching and finding people but she has used all her free likes for the day.</p>
 <p><input type="checkbox"/> One time subscription <input checked="" type="checkbox"/> monthly renewal subscription</p>	
<p>Olivia chooses to upgrade her account to premium. She enters her payment info and</p>	<p>Now Olivia can continue matching. Her profile also gets a custom border and she can "rewind" to change her match decision.</p>

## **4. High level database architecture and organization**

### **1. Database Organization**

#### **- Business Rules**

##### **A. Users (strong)**

- A new user shall be able to create one and only one account.
- A registered user shall be able to register as only one account type.
- A user shall have a unique id.
- A user shall have a first name.
- A user shall have a last name.
- A user shall have a unique email address.
- A user shall define gender.
- A user shall have a birthdate.
- A user shall have a phone number.
- A user shall have an art category.
- A user shall have at least one tag.
- A user shall have a skill level.

##### **B. User Address (strong)**

- A user address shall have a unique id.
- A user address shall have a street.
- A user address shall have a city.
- A user address shall have a state.
- A user address shall have a country.
- A user address shall have a zip code.

##### **C. Account**

- An account shall be available to one and only one user.
- An account shall have one unique id.
- An account shall have a username.
- An account shall have an account type.
- An account shall have user id.
- An account shall have one encrypted password.
- An account shall have one account creation date.
- An account shall have account status.

##### **D. Account Type**

- An account type shall have one unique id.
- An account type shall have one description.

## **E. File Paths**

- File paths shall have one unique id.
- File paths shall have a path for profile picture.
- File paths shall have a path for pictures.
- File paths shall have description for pictures.
- File paths shall have paths for videos.
- File paths shall have description for videos.
- File paths shall have a path for audio.
- File paths shall have description for audio.
- File paths shall have a path to their YouTube profile.
- File paths shall have a path to their Spotify profile.
- File paths shall have a path to their Instagram profile.
- File paths shall have a path to their website.

## **F. Preferences**

- Preferences shall have one unique id.
- Preferences shall have interest on minimum age range.
- Preferences shall have interest on maximum age range.
- Preference shall have interest on gender preference.
- Preference shall have interest in the meeting category.
- Preference shall have interest on minimum location range.
- Preference shall have interest on maximum location range.
- Preference shall have user's match interests on art categories.
- Preference shall have the user's interest on skill level.

## **G. Matches**

- Matches shall have one unique id.
- Matches shall have a username.
- Matches shall have a match\_status.

## **H. Chats**

- Chats shall have a unique id.
- Chats shall have client username.
- Chat shall have a creation date.

## **I. Chat Messages**

- Chat messages shall have a unique id.
- Chat messages shall have user\_id.
- Chat messages shall have client username.
- Chat messages shall have file id.
- Chat messages shall have a creation date.
- Chat messages shall have text.

## **J. Chat File**

- Chat files shall have a unique id.
- Chat files shall have a unique name.
- Chat file shall have a file path.

## **K. Community page**

- Community page shall have a unique id.
- Community page shall have a user's name.
- Community page shall have a post title.
- Community page shall have a post field.
- Community page shall have post total likes.
- Community page shall have a post creation date.

- **Entities, their attributes, relationship, and domains at the high level**

### **A. Users (strong)**

- **User\_id**: key, numeric
- **First\_name**: multivalue, alphanumeric
- **Last\_name**: multivalue, alphanumeric
- **DOB**: multivalue, timestamp
- **Gender**: multivalue, alphanumeric
- **Username**: multivalue, alphanumeric
- **Email**: key, alphanumeric
- **Phone**: multivalue, alphanumeric
- **Art\_category**: multivalue, alphanumeric
- **Tag**: multivalue, alphanumeric
- **Skill\_lvl**: multivalue, numeric

### **B. User Address (strong)**

- **User\_add\_id**: key, numeric
- **Street**: alphanumeric
- **City**: alphanumeric
- **State**: alphanumeric
- **Zipcode**: alphanumeric
- **Country**: alphanumeric

### **C. userAdd (weak)**

- **Users**: weak key, numeric
- **User\_address**: weak key, numeric

#### **D. Account (weak)**

- **User\_acc\_id**: key, numeric
- **Users**: weak key, numeric
- **Type**: weak key, numeric
- **Password**: multivalue, alphanumeric
- **Created**: composite, alphanumeric
- **Activate**: multivalue, alphanumeric

#### **E. Account Type (weak)**

- **User\_accType\_id**: key, numeric
- **User\_description**: multivalue, alphanumeric

#### **F. File Paths (weak)**

- **filePath\_id**: key, numeric
- **profile\_pic\_path**: multivalue, alphanumeric
- **pic\_path**: multivalue, alphanumeric
- **pic\_path\_desc**: alphanumeric
- **video\_path**: multivalue, alphanumeric
- **video\_path\_desc**: alphanumeric
- **audio\_path**: multivalue, alphanumeric
- **audio\_path\_desc**: alphanumeric
- **profile\_1**: multivalue, alphanumeric
- **profile\_2**: multivalue, alphanumeric
- **profile\_3**: multivalue, alphanumeric
- **profile\_4**: multivalue, alphanumeric

#### **G. Preferences (weak)**

- **preference\_id**: key, numeric
- **min\_age**: numeric
- **max\_age**: numeric
- **gender\_pref**: alphanumeric
- **min\_loc**: numeric
- **max\_loc**: numeric
- **art\_pref**: multivalue, alphanumeric
- **meeting\_pref**: multivalue, alphanumeric
- **skill\_lvl\_pref**: numeric

#### **H. Matches (weak)**

- **match\_id**: key, numeric
- **match\_client\_username**: weak key, alphanumeric
- **match\_status**: multivalue, alphanumeric

### I. Chats (weak)

- **chat\_id**: key, numeric
- **matches**: weak key, numeric
- **chat\_message**: weak key, numeric

### J. Chat Messages (weak)

- **Chat\_msg\_id**: key, numeric
- **Chat\_user\_id**: weak key, numeric
- **Chat\_msg\_client\_username**: weak key, alphanumeric
- **Chat\_msg\_file\_id**: weak key, alphanumeric
- **Chat\_msg\_creation**: multivalue, timestamp

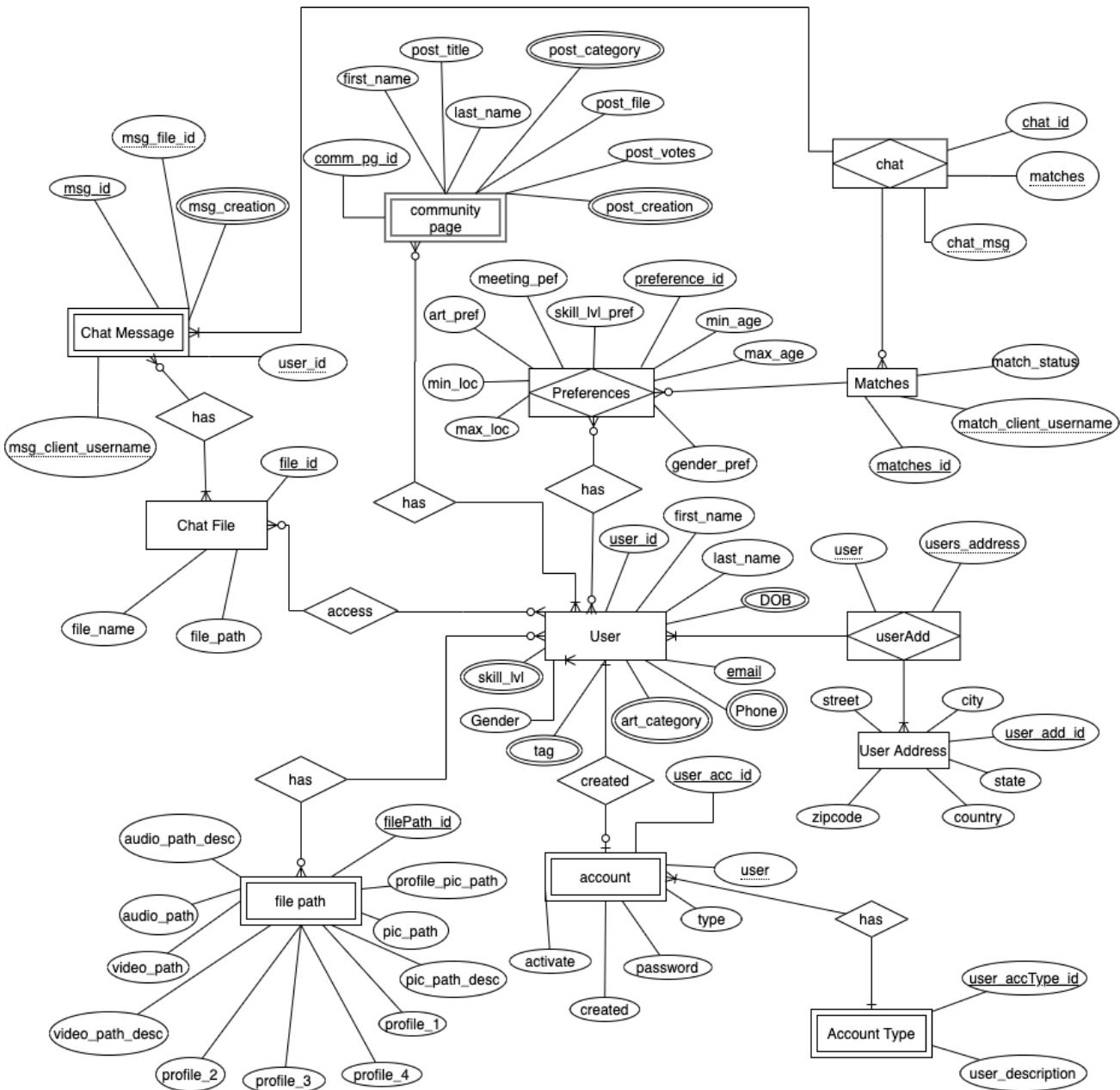
### K. Chat File (strong)

- **Chat\_file\_id**: key, numeric
- **Chat\_file\_name**: multivalue, alphanumeric
- **Chat\_file\_path**: multivalue, alphanumeric

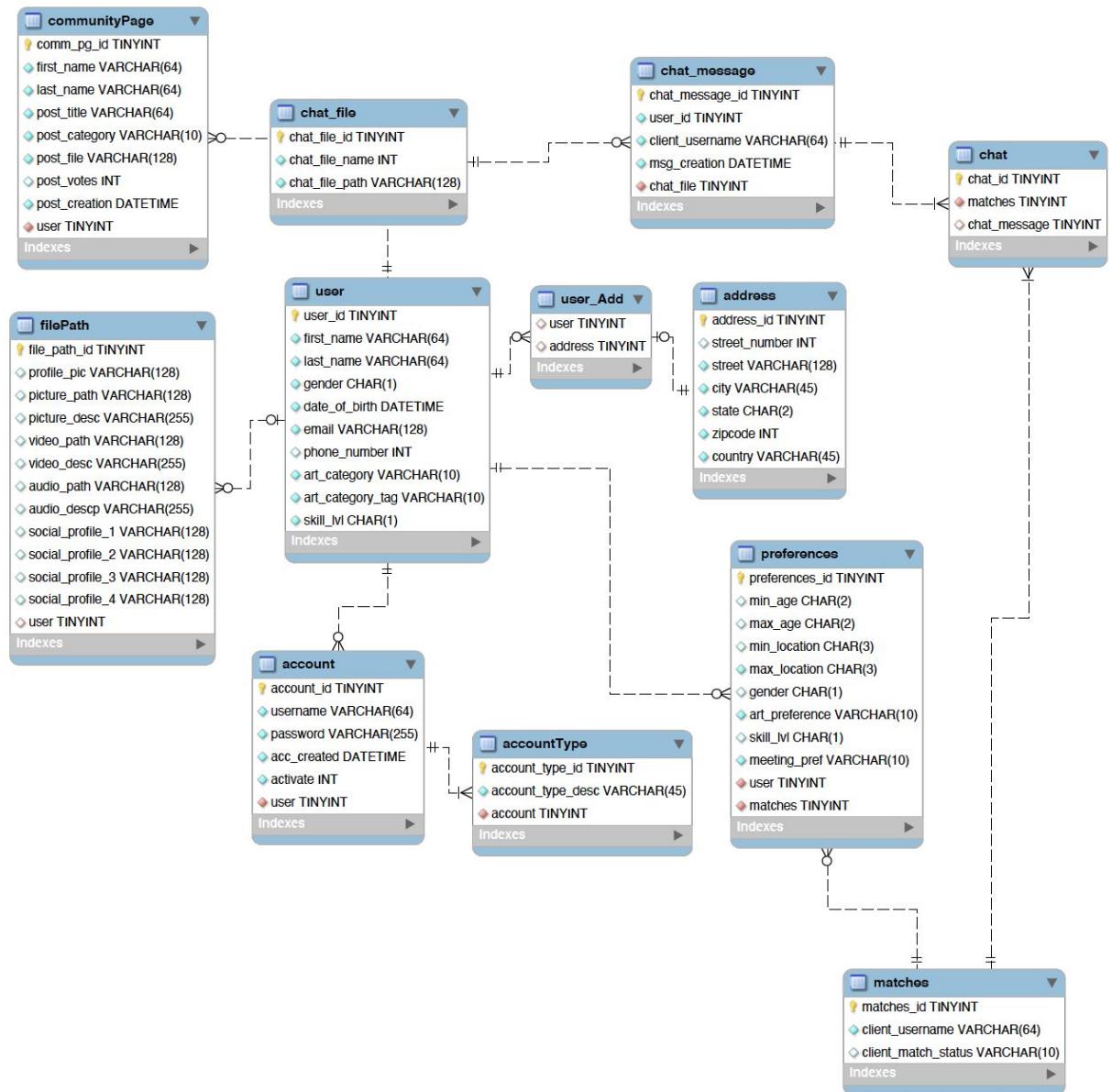
### L. Community page (weak)

- **Comm\_pg\_id**: key, numeric
- **First\_name**: multivalue, alphanumeric
- **Last\_name**: multivalue, alphanumeric
- **Post\_title**: multivalue, alphanumeric
- **Post\_file**: multivalue, alphanumeric
- **Post\_votes**: numeric
- **Post\_creation**: multivalue, timestamp

- **ERD Diagram**



- **EER Diagram (made using MySQL Workbench)**



- **DBMS**

We will be using **MySQL workbench** to create the database of our website, and there are many reasons to use MySQL workbench.

Actions will be directly carried out on the tables; it is also free and open source. It also has enhanced entity-relationship modeling support. Lastly, it is compatible with multiple operating systems, such as windows, Linux, and OS X.

## **2. Media Storage**

For the media/file storage we have decided to use/store them in a **file system**.

We made this decision as a team after doing some research on the media storage. Our website allows each user to upload multiple photos, videos and media files (upto 80mb for free account and 120mb for premium account) as a primary form of showing their work.

Storing media files in a database will cause load on the database, and negatively impact the performance, which we do not want so we chose to store media files in a file system.

In the database system, we will have a path stored which will take/help users to access the media storage. Using these file paths the user will be able to access their media files easily and much quickly.

## **3. Search/filter architecture and implementation**

Our product's searching system is fairly simple, wherein the website allows users to search the people they are connected to. With the help of a search algorithm, users will be able to search people in their match list only, by their names, or by their usernames.

In addition to letting people search the list of people they are connected with, users also have the ability to search the community page for posts based on the username of the person making the post or the art category they want to see the post in.

## 5. High Level APIs and Main Algorithms

### APIs to be used

**Location API:** For matching 2 registered users together based on their location, we will be using locationIQ (<https://locationiq.com/>) API, which will let us accurately present the user with valid matches that fits their location search criteria. For our website, we will have a search radius preference for matching two users together. Based on the radius specifications set by the user, we will present them with potential matches to pick and choose from.

**Encryption API:** Encryption should encrypt a user's password before it is stored in the database to secure the information. The password will be stored in a database in encrypted form, and will not be accessible to anyone. When a user tries to login a user's account, the Encrypt API will fetch the encrypted password, decrypt to see if it matches with what user types in. We will be creating our own API for encryption and not using any 3rd party libraries/APIs to uphold security standards.

**Authentication API:** Authentication should verify a user's identity using a unique identifier. A unique identifier is a set of a valid email-address as an User ID and associated password that user input when they registered the site. We are also planning to implement two factor authentication and OAuth 2.0 to enforce stronger security measures. Using these standard authorization practices, we will try to ensure that the users' account is secure and cannot be breached easily.

**Payment API:** We will also be implementing a payment API, since we will need to securely take payment information from the users' who opt to enroll in a premium account. We are still debating between using a 3rd party API vs building our own API. While third party APIs will already have the security aspect built in, it is a risk since we will be ultimately sharing sensitive user information with an outside company. Also most third party APIs are paid (we did find a free one: [Noodlio-Pay](#) and are testing its integrity) so we are not sure if we can go that route. With implementing our API, we need to ensure that it is secure, all the information is encrypted, along with making sure that the API is compatible for at least all the major cards/payment methods.

## Algorithms

**Sorting:** Given the expected amount of matches a user can have. We will rely on insertion sort for organizing the list of matches a user has. The amount of likes will also be sorted in this manner.

**Matching:** Matching will be done based mostly on preferences selected by the user. We will prioritize search results based on the meeting criteria first (in person or online). If it is a meet-in-person preference, we will then filter the results based on the minimum and maximum radius set by the user. Next, any tags that the user has listed will be prioritized and we will present results based on the custom tags/requirements set by the user. If a user has a premium account, they will have priority when being matched by other users, meaning that they will show up first and their highlighted profiles will gain more traction.

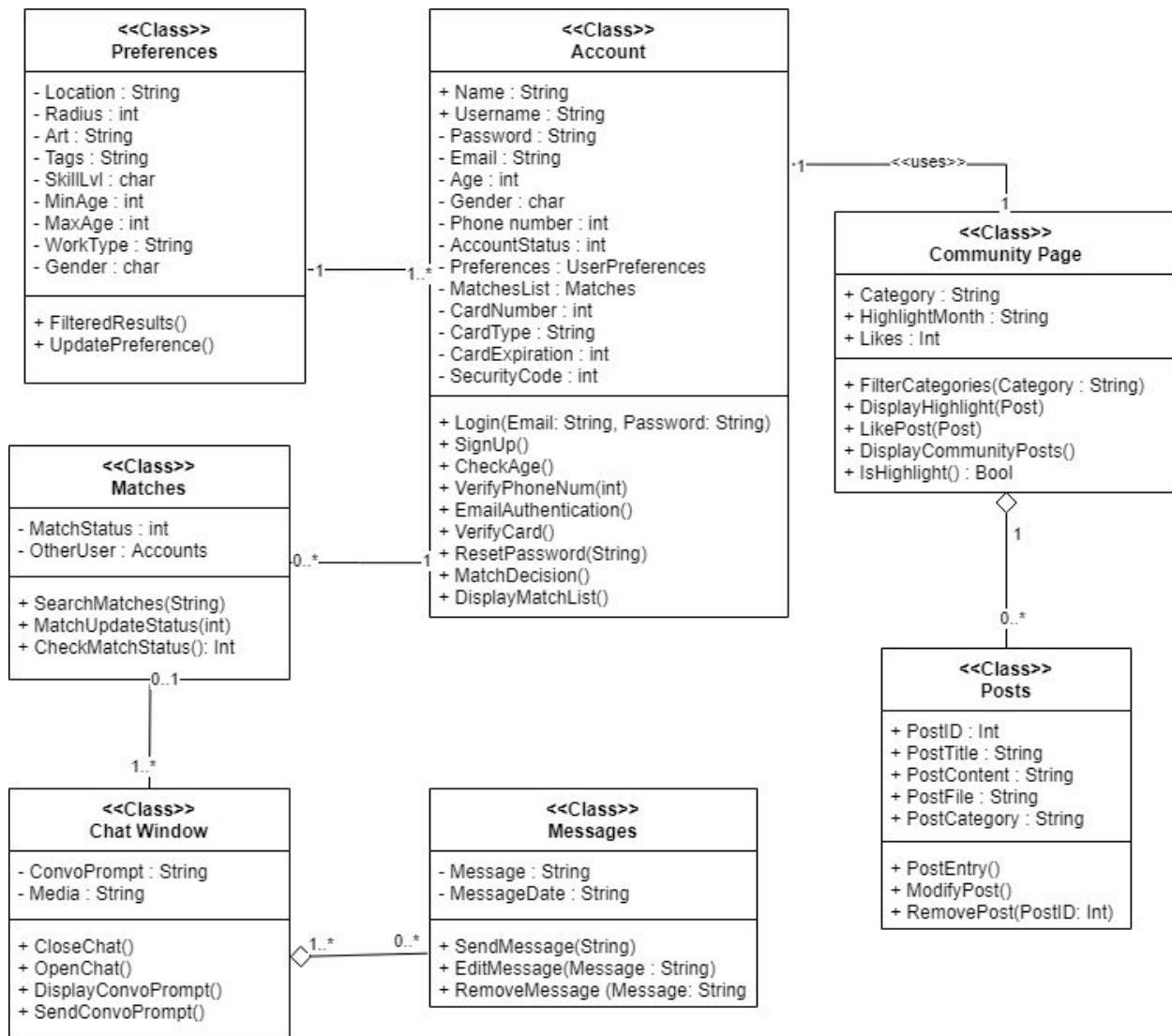
**Searching:** Our product provides users only relevant results without hassle. To provide these experiences, the product automatically filters out search results by user's predefined references such as, prefer art area or skill level of the art, and geographic information. Users will only get the result around the user's area. The only search users can do is within the list of people they are already matched with.

**Ranking:** Our website has a feature that highlights the most popular post that users have created with whomever they were matched with. The highlighted post will be chosen based on the amount of "likes" the post has. These "likes" will have a weight tied to them. The weights of these "likes" will degrade over time at certain time intervals.

## Changes to Stack:

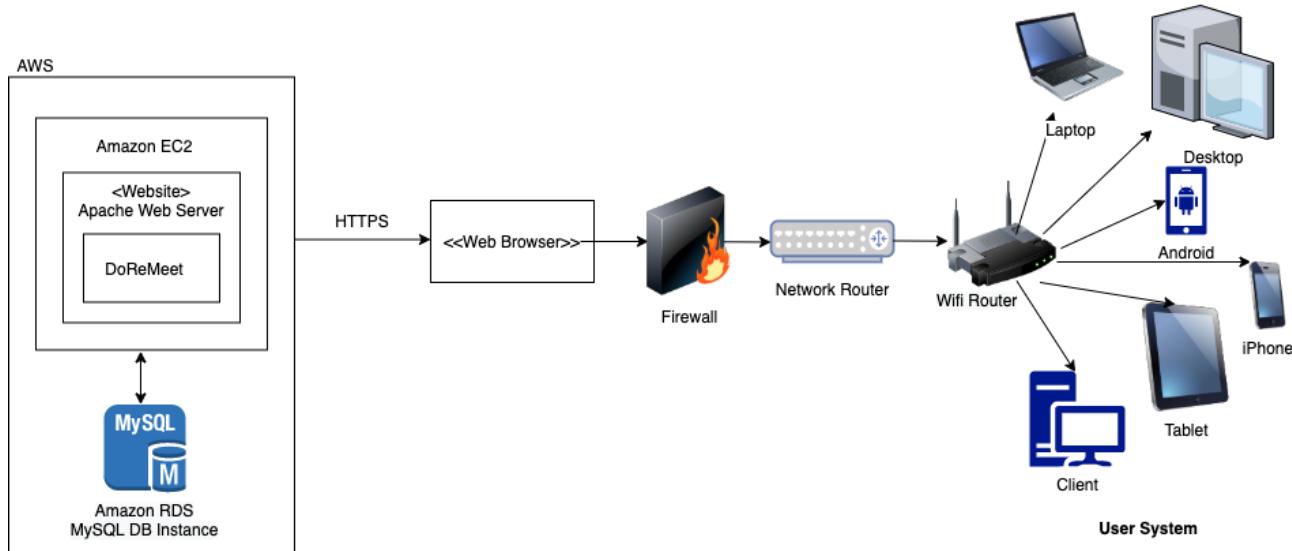
Instead of using React framework, we are using **Express** framework for our backend.

## 6. High Level UML Diagrams

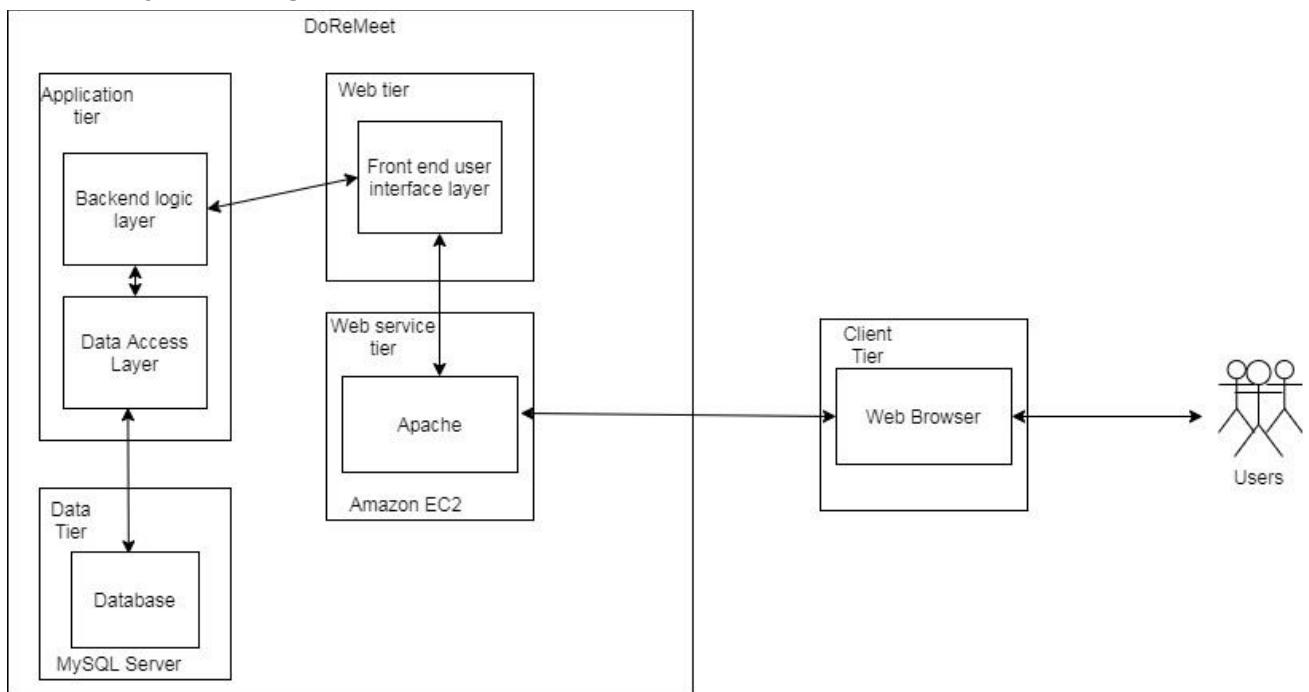


## 7. High Level Application Network and Deployment Diagrams

### Network Diagram



### Deployment Diagram



## 8. Key risks

### Skills risks

- Lack of experience in UI/UX design.

Our team is fairly new to building a whole product from scratch and the limited experience in terms of building good UI/UX design is a risk that will likely create roadblocks for us..

*Resolution:* We aim to utilize good practices and look to other sites for inspiration when it comes to site design.

- Lack of experience in the database and using API.

Similar to UI/UX, we have limited experience in implementing 3rd party APIs and in creating our own APIs. Also, there is only one person on our team who has some knowledge of databases, so navigating that and building an effective database will be a bit of a challenge.

*Resolution:* To overcome this limitation, we aim to learn in class from Professor Ortiz Costa, and utilize online learning materials (video, documentation, learning websites, etc.)

### Schedule risks

- Our goals for DoReMeet might be a bit lofty, and we may be overestimating our own abilities and may not be able to deliver on all of our priority 1 requirements.

*Resolution:* Properly manage our time and our team in order to maximize productivity and accomplish all the things we want to get done.

- Our timeline is too tight to implement our product's priority 3 level requirements, so we are worried if we'll get to test our product properly.

*Resolution:* We think we might need to come up with a few edge cases and start implementing tests at an early stage in our development process.

### Technical risks

- Relying on external API's might present issues in a lack of needed functionality or unforeseen bugs.

*Resolution:* Properly research all APIs before committing to using them, and using only stable versions to prevent any unforeseen bugs.

### Teamwork risks

- Loading too much work on a single teammate could lead to a backlog of work. *Resolution:* Make sure to properly distribute work and keep schedules in mind.

- We are doing a team project in a virtual environment. Remote situations give us more time to work, but at the same time, it makes it hard to focus on the task.

*Resolution:* Make sure to prioritize tasks and deload every work to every member. Use a task manager (Monday) to keep tracking small tasks and share the progress with other members. We are trying to prioritize communication within the team, to make sure that everyone is on the same page about all our work.

### Legal/Content risks

- We are handling user's location information, potentially even their credit card information (for premium users) and other personal information. There might be a chance people misuse our application with ulterior motivation, and try to cause harm to someone.

*Resolution:* We need to be extra careful with user's information, and we will give the users' ability to choose what information they want to share and not share.

## 9. Project management

For our product we are choosing to use monday.com.

We've set up our Project Board and use it to split tasks, assign work, and update the status of the work. Also, we are using Monday.com to keep track of deadlines and the timeline for our project, along with categorizing the work based on priorities.

This platform will help us keep track of our own progress, and see what stuff our team members are working on. It'll help us all communicate better and see the overall progress of the team.

The screenshot shows a project board on the Monday.com platform. On the left, there's a sidebar with options like 'Main workspace', 'Add', 'Filters', 'Search boards...', and a selected 'Project DoReMeet'. The main area displays a 'Milestone 1' board with one item: 'Milestone 2'. This milestone has a subitem 'Version 1' which contains several tasks:

Subitems	Owner	Status	Date	Notes
Main Data Items and Entities V2	N	Done	Oct-24	
Functional Requirements V2	N	Done	Oct-24	
UI Mockups and Storyboards	LC VT	Done	Oct-26	
High level database architecture and or...		Done	Oct-27	
High Level APIs and Main Algorithms	J	Working on it	Oct 27	
High Level UML Diagrams		Working on it	Oct 27	
High Level Application Network and De...	LC	Done	Oct-27	Needs Review
Actual key risks	N	Done	Oct-27	
Project management	N	Done	Oct-27	

Below this, another section labeled 'Vertical SW Prototype' is partially visible. The interface includes various icons for communication and management.

## 10. Team Contributions

Member	Role	Contributions
Nimiksha	Team Lead & Documentation Lead	<u>Documentation:</u> Main Data Items and Entities Functional Requirements Database architecture (Review only) High level APIs and Algorithms High Level UML Diagram (Design) Key Risks Project Management <u>Vertical Prototype:</u> Code Review
Meet	Backend and Database Lead	<u>Documentation:</u> Main Data Items and Entities Functional Requirements High level database architecture (Design and Implementation) ERD, EER Diagram, DBMS, Media Storage and Search/filter architecture High level APIs and Algorithms High Level UML Diagram (Design) Key Risks <u>Vertical Prototype:</u> Database implementation
Luke	Frontend Lead	<u>Documentation:</u> Main Data Items and Entities Functional Requirements UI Mockups and Storyboard (General Flow and 6 use cases) High level APIs and Algorithms High Level UML Diagram (Design) Network and Deployment Diagrams Key Risks <u>Vertical Prototype:</u> Frontend code Backend to Frontend connection
Mike	GitHub Master & Backend Engineer	<u>Documentation:</u> Main Data Items and Entities Functional Requirements High level APIs and Algorithms High Level UML Diagram (Design and Implementation) Key Risks <u>Vertical Prototype:</u> Backend Code (search from database and insert to database) Backend to Frontend connection
Jungsun	Database & Backend Engineer	<u>Documentation:</u> Main Data Items and Entities

		<p>Functional Requirements          High level APIs and Algorithms          High Level UML Diagram (Design)          Key Risks  <u>Vertical Prototype:</u>          Backend Code (connect to database, insert into database)</p>
Vincent	Server Admin & Frontend Engineer	<p><u>Documentation:</u>          Main Data Items and Entities          Functional Requirements          UI Mockups and Storyboard (5 use cases)          High level APIs and Algorithms          High Level UML Diagram (Design)          Key Risks  <u>Vertical Prototype:</u>          Connect backend to frontend          Search and Insert to Database</p>

The team worked on all sections of Milestone 2 together during Zoom meetings, and collaboratively created this report.

# SW Engineering CSC648/848 FALL 2020

## DoReMeet

### Team 02

#### Milestone 3

Nimiksha Mahajan ([nmahajan@mail.sfsu.edu](mailto:nmahajan@mail.sfsu.edu)) - Team Lead

Meet Patel ([mpatel6@mail.sfsu.edu](mailto:mpatel6@mail.sfsu.edu)) - Backend & Database Lead

Luke Chang-Knezevich ([lchangknezevich@mail.sfsu.edu](mailto:lchangknezevich@mail.sfsu.edu)) - Frontend Lead

Mike Bolanos ([mbolanos1@mail.sfsu.edu](mailto:mbolanos1@mail.sfsu.edu)) - GitHub Master & Backend Engineer

Jungsun Eoh ([jeoh@mail.sfsu.edu](mailto:jeoh@mail.sfsu.edu)) - Database & Backend Engineer

Vincent Tse ([vtse3@mail.sfsu.edu](mailto:vtse3@mail.sfsu.edu)) - Server Admin & Frontend Engineer

#### History Table

Version	Date	Comments
M3V2	12/05/2020	Submission updated with Feedback
M3V1	11/19/2020	Initial Submission
M2V2	12/05/2020	Submission updated with Feedback
M2V1	11/01/2020	Initial Submission
M1V2	10/11/2020	Submission updated with Feedback
M1V1	09/30/2020	Initial Submission

## **Table Of Contents**

1. Data Definitions V3	2
2. Functional Requirements V3	8
3. Wireframes based on Mockups/Storyboards V2 (detailed)	12
4. High level database architecture and organization V2 (detailed)	24
5. High Level Diagrams V2 (detailed)	25
6. List of Contributions	27

## 1. Data Definitions V3

1. **Registered User/User** - A user is any individual who has created an account on the website using their name, a valid email, date of birth and password. While registering their account, the user is also required to specify their Art preferences, their skill level in that category and their phone number (used for 2 factor authentication).
2. **Unregistered User** - Any individual who has not created an account is an unregistered user. An unregistered user is eligible to create a free or premium account. An unregistered user can also navigate to the Community page from the Landing page and browse through the recent posts and Highlights of the month. However, an unregistered user cannot make any posts to the community page.
3. **Landing Page** - The first page any individual, both registered and unregistered, will see when they visit the website. The page shall lead to a sign up or login page. Also the page will have information about the website, the product along with a link to the Community page, accessible to both registered and unregistered users.
4. **Account** - An account is an entity associated with a registered user, and the site sees each user in terms of their valid account.
  - a. *Active/Activated account* - An active account is a user account that is in service and the information is stored on our database. An account is active when the user first creates and sets it up. Also, the account is activated after the user unpauses their account.
  - b. *Paused Account/Inactive Account* - Users have the ability to pause their account or deactivate it without deleting their information. This can be done if the user wants to take a break from using the website, but don't want to leave the platform all together.
  - c. *Disabled Account* - Users with an account have the option to delete their account and remove all their information from our database. Such an account is a disabled account.
5. **Account Type** - Every registered user can have a free account or a premium account. A free account is created by default, but the user has the option to opt for a premium account at a small fee, and access the premium features of our website. A user has the ability to switch between a free account and premium account, and vice versa.
6. **Free Account** - A free account is a regular account that every registered user gets by default. Unlike the premium account, the free account does not have paid features. The user is capable of switching from a free account to a premium account and vice versa, as per their leisure.
7. **Premium Account** - A premium account is a paid account which in addition to all the features a regular account offers, offers additional features. The pricing model for the Premium account is set on 3 levels - monthly subscription, 6 months subscription or annual subscription. Premium account gives users a more customized experience and highlights their profile to gain more traction than a free account.

- a. *Premium Features* - A premium account includes premium features including unlimited number of likes, extended search radius to 150 miles, increased portfolio size (from 80mb in free account to 120mb), highlighted user profile, ability to rewind on a passed potential match and no ads while using the website.
  - b. *Rewind* - As a part of their premium membership, users can “rewind” on their last potential match. This means that if they passed on a user, but didn’t mean to, or they changed their mind, they have the ability to go back and look at the profile again, and like or pass as they wish.
8. **Profile** - A profile is the user’s information on the website. They can set it up in any way they like, displaying whatever information they feel is relevant to them. The site provides means for them to curate their profile and optimize it to gather more traction.
- a. *Bio* - A bio is a brief text description that the user can add to their profile to describe what they are looking for, or talk more about their work. The user has the ability to update their bio from their Profile page anytime.
  - b. *Photos* - Users have the ability to add pictures of their work or anything relevant that showcases their interest or what they are looking for. Users can add or delete photos anytime from their profile page.
  - c. *Videos* - Similar to photos, users can add short videos to their profile showing their work as well. This is especially useful for musicians, dancers and filmmakers, who will get a chance to showcase a snippet of their work.
  - d. *Links* - Users can add links to their website or our portfolio pages on their bio to endorse their work.
  - e. *Linked Accounts* - Users have the ability to link other accounts to our website. For instance, they can link their Instagram, Youtube or Spotify account to show more of their portfolio.
  - f. *Tags* - The tags a user chooses for their account, are also visible on the users profile page. These tags, like the other information, are editable. Tags represent keywords that show what the user is interested in/offering.
9. **Preferences** - Users can customize their profile by adding preferences to it. These preferences will make their account more personalized and will help them get more targeted potential matches. The users can add a variety of preferences from the given choices:
- a. *Art Categories* - The users will be able pick one of the five art categories - *painting, dancing, music, photography, cinema* - and their profile will get hits in that category only, i.e. dancers will see others dancers while singers will be put in a pool of other singers. This will help users get direct access to the community they are seeking to join.
  - b. *Work Type* - The site provides a platform for both passion projects and work. Users can filter their potential matches based on their preference and choose to work with others on serious projects or just collaborate on a passion project/hobby.
  - c. *Distance* - The users have the option to set distance preferences if they are open to meet other people in person. They can choose any radius between 1 mile to 100 miles (150 miles for premium account users) of their geographical location, and they will see potential matches only in the specified distance. Premium account holders can extend their radius upto 150 miles.

- d. *Meet online only* - Users also have the option to set their profile preference to meet online only - i.e. they can opt to see only users who want to meet online and collaborate virtually. They don't have to share their location in this case, but will still have the option to filter what countries or cities they want to see the artists from.
  - e. *Gender* - Users will have the option choose artists based on gender (Male, Female, Trans)
  - f. *Age* - Users can also set an age range as a preference criteria for their potential matches. They can choose any range between 18 and 100 years of age. Users have the ability to set *minimum* and *maximum* age filters for their potential matches.
  - g. *Skill* - Users have the ability to set their skill level (beginner, intermediate, expert) on their profile and can choose to only see potential matches based on their skill level. For instance, they can choose to learn with other beginners or work with an expert.
- 10. Tags** - Users will be able to add tags to their profile, which highlight key aspects about them, for instance a user is interested in art, portrait painting in particular and wants to meet fellow artists younger than 30 years in 50 miles of his radius. Also this is just a hobby for him, so he can add portraits, hobby, meet in person, as more such things as his tags, which help condense the information on his profile and get it more traction from other users looking for similar things. Tags will help filter a users' profile and help us share more customized potential matches.
- a. *Custom Tags* - Users can create their own custom tags to really highlight what they want. Some instances could be "unpaid", "urgent", "fun", "volunteer partner", "LGBTQ friendly" or anything that they think is relevant to their profile. The custom tags help share more about the user with potential matches and make more fitting connections.
- 11. Potential Matches** - Potential Matches are the users that a user sees based on their preferences. The potential matches fit whatever criteria the user has set for art category, distance, age, skill or gender. The user has the option to either pursue these matches or forego them. These are just people who the system thinks would be a good match with the user. Potential Matches are visible on the match page and a user can choose to either "Connect" with them or "Pass" which would indicate the users' interest or lack thereof in that particular potential match.
- a. *Waiting State* - Potential Matches that are liked by one user are in a waiting state till the other user either likes or passes on them.
  - b. *Filtered Results* - The potential matches can be filtered based on the preferences set by the user. For example, if a user sets distance limitations, all potential matches outside that radius will be filtered out.
- 12. Likes** - Likes are the potential matches that the user wants to "Connect" with. Pressing the green connect button, means a user liked the potential match. If the user does want to work with these potential matches and if they like them back, then the two of them will be connected. Regular account holders can like upto 100 people per day, whereas premium account users will have unlimited number of likes.
- 13. Pass** - Passing are the potential matches that the user does not want to pursue, because they don't match their interest or they don't think would be a good match for them. Pressing the red

pass button, means a user liked the potential match. Even if these potential matches, that the user passed, likes them back, they will not be connected, given it is not a mutual interest to work together. Once a user passes on a potential match, they will not see that profile for 4 week so that they get a more concise experience, but might see the same profile after, giving them the opportunity to pursue the missed match later.

14. **Matches/Matched users** - Matches are the potential matches that the user liked and who liked them back. Given their mutual interest in being connected, the site matched the 2 users and provided them a platform to interact with text messages, share photos/media etc and collaborate on whatever project they were both seeking to pursue. Matches can be seen under the Chat page, where the user is able to communicate with their match. A user can choose to “unmatch” with any match whenever they like. Also, matches are not lost if a user pauses their account, but will be lost if they delete their account.
15. **Community Page** - The community page holds joint projects that users have uploaded. Users who match with others on the site and go on to create something have the option to upload that project here and browse the projects that others have made. Users can vote on which project they think deserves to be the project of the month. The community page can be accessed from the Home page too, but a user has to be logged in, in order to make any posts. They can however view the recent posts and the highlighted posts even when they are not logged in. The community page shows the top 5 recent posts (based on chronology), though the user can see more if they keep scrolling, and the top 3 highlighted posts (based on number of user votes). Also, a user has the ability to search within the Community page for any particular posts from a genre.
16. **Highlights of the Month** - The highlights of the month will be community voted projects. Each month the projects with the highest number of votes will be showcased as the highlights of the month for all users. Only registered users have the ability to vote on the posts shared on the Community page, and the highlighted posts are picked.
17. **Report** - A user has the ability to report any potential matches or matches if they engage in any illegal, violent or indecent activity. A reported user will be investigated and can have their account suspended if found guilty. The reports are handled by the admin and are investigated in compliance with the websites Terms and Conditions and Privacy Policy.
18. **Flag** - A user has the ability to flag any inappropriate content that they see from any potential matches or matches. The content will be reviewed, and if found in violation of the site's policies will be removed from the site. Also the user who posted it, will be notified with a warning regarding the content.
  - a. *Inappropriate Content* - Content that violates our websites Community Guidelines, and/or aims to promote violence, hate or bias is considered inappropriate.
19. **Block** - A user has the ability to block any potential matches or matches if they feel don't want to engage with that person. Any blocked user will not be seen by the user and will not be able to

connect with or contact the user. A blocked account will not appear on the users' potential match list, ever.

20. **Unmatch** - A user can choose to unmatch with any of their matches, if they realize they don't want to pursue that connection, or don't want to stay connected for any reason. An unmatched user is not blocked, and can still appear in the users potential matches.
21. **Message List** - The message list page shows a list of all the chat messages of a user with all their matches. The list has a little preview showing the users match's name and their text exchanges. When clicked, the Chat page will show the entire message history between the two users.
22. **Messaging/Chat Window** - When connected with a user (a potential match who the user likes and who liked them back), the user can connect with the match using a messaging or chat window, where they can share information via texts, photos or short videos.
23. **Conversation prompts** - The chat window gives initial conversion prompts when the users are first matched. This is to promote engagements and help users reduce the anxiety of coming up with a conversation starter. These prompts are suggested to promote engaging and productive conversations between the matches.
24. **Password** - The user needs to set a password when they create an account. The user then uses this password to log into the website from any device or browser. The password needs to meet certain security requirements (password length, special characters etc. as per the standard), to ensure that it is a strong password that cannot be easily cracked.
  - a. *Forgotten Password* - The user might forget their password, and they have the ability to say they forgot their password, at which point they'll be given a special link to reset their password via email.
  - b. *Reset Password* - The user can choose to reset their existing password (even if it is not forgotten) and will have the ability to do so.
25. **Administrator/Admin** - An administrator or admin is someone who has access to the site's backend and is able to investigate any flagged or reported activities on the website. An admin has the ability to remove any content/user that is found in violation of the website's policies.
26. **Customer Service** - Customer Service is a support team for the website that can help the users with any technical or logistical difficulties and can help them navigate the website better. Also the customer service is able to support the user with any settings, password updates etc.
27. **FAQs** - Available on the footer, FAQs are the most frequently asked questions about the product and the website. The purpose of this section is to help the user navigate the website, without constantly having to contact the admins for small questions. It is a one stop place for most of the users' questions. If there are still questions, they can't find the answer to, they have the ability to go to the Contact page and send a message to the site admins.

28. **Privacy Policy** - Privacy policy, also available on the Footer on every page of the website, details how we collect and/or use the users' data. The purpose of this is to maintain transparency, and make users aware of how we are using/showing their data. Every user that creates an account, marks that they are agreeing to these privacy policies and will act in accordance with them. Anyone violating these policies can have their account temporarily suspended or even permanently deleted by the admins.
29. **Guidelines** - Guidelines are also available on the Footer section and share how users can make the best of the product without engaging in any unlawful, violent or hurtful manner. These guidelines are to help the user have a safe and enjoyable experience on DoReMeet. Failure to adhere to these guidelines may result in losing access to our platform.
30. **Terms and Conditions** - Terms are Conditions, also available in the footer area, is a contract between the company and the registered users, and aims to educate the users of both their rights and DoReMeet's rights in case of any potential future conflicts. Every user registering an account agrees to these Terms and Conditions, and if they don't, they will not be able to create an account on our platform. Additionally, non conformity to the terms and conditions can lead to the user losing access to our platform.
31. **Contact Us** - The Contact Us page allows users to communicate with the DoReMeet team to share any questions/concerns/suggestions they might have about the website. The aim of this page is also to get feedback from the user to improve our product to cater more to their needs. The user needs to provide their name and email, along with the message to contact the DoReMeet team, and that email will only be used to correspond with them. It will not be shared with any other users or third party apps.

## 2. Functional Requirements V3

### **Priority 1**

#### **Registered User (also referred to as user):**

1. Users shall be able to sign in to the website.
2. Users shall be able to sign out from the website.
3. Users shall be able to change their existing password.
7. Users shall be able to Like other users.
8. Users shall be able to Pass other users.
9. Users shall be able to view potential matches' profiles.
10. Users shall be able to view their matches' profiles.
14. Users shall be able to vote on collaborations posted on the community page.

#### **Unregistered User:**

15. An unregistered user shall be able to create an account using a valid email.
16. An unregistered user shall be able to only see the landing page of the website.
18. An unregistered user shall be required to enter a date of birth upon account creation.

#### **Account:**

19. An account shall have one valid email linked to it.
24. An account shall be active to allow users to continue matching with others.
25. An account shall allow its user to modify the password.

#### **Profile:**

32. A profile shall allow users to upload pictures.
33. A profile shall allow users to delete pictures.
34. A profile shall allow users to upload text descriptions as bios.
35. A profile shall allow users to update their bio.
36. A profile shall allow users to delete their bio.
37. A profile shall allow users to upload short videos.
38. A profile shall allow users to link their Youtube account.
39. A profile shall allow users to link their Spotify account.
40. A profile shall allow users to link their Instagram account.
41. A profile shall allow users to link any websites.
42. A profile shall allow users to change what is displayed publicly on their account.

#### **Preferences:**

43. Preferences shall allow users to filter potential matches based on art categories.
44. Preferences shall enable users to choose if they want to meet online.
45. Preferences shall enable users to choose if they want to meet in person.
46. Preferences shall enable users to share their location.
47. Preferences shall enable users to change their match preferences to online only.

48. Preferences shall enable users to change their match preferences to local only.
49. Preferences shall let users set a location radius for meeting in person.
52. Preferences shall enable users to adjust preference of minimum age for potential matches.
53. Preferences shall enable users to adjust preference of maximum age for potential matches.
54. Preferences shall let users to filter match preferences based on gender.
55. Preferences shall let users to set skill levels.
56. Preferences shall allow users to update their skill levels.
57. Preferences shall let users to filter potential matches based on skill levels.

**Community Page:**

58. The community page shall contain work that has been created by a collaboration of users.
59. The community page shall contain highlights of the month.

**Password:**

67. Passwords shall be updatable via email in case the user forgets their password.

**Tags and Custom Tags:**

68. Tags shall be added to the users profile to highlight preferences.

**Matches:**

70. Matches' profile shall be accessible to the user.
71. Matched users shall be able to communicate via text message.
77. Matches shall be able to unmatched their existing matches.
78. Matches shall be made when both users like each other.
74. Matches shall be able to search within their match list by first name.
75. Matches shall be able to search within their match list matches by username.

**Potential Matches:**

79. Potential Matches shall be liked by both users to become a match.
80. Potential Match shall be removed if a user passes on them.
81. Potential matches shall be matched when both users have liked.

**Chat Window:**

82. Message List shall show a list of all the conversations between matches.

**Priority 2**

**Registered User (also referred to as user):**

11. Users shall be able to report other users.
12. Users shall be able to flag inappropriate content.
13. Users shall be able to block other users.

**Account:**

20. An account shall be paused by the user anytime they want.
21. An account shall allow users to login while it is paused.
22. An account shall be disabled by users through the settings page.
23. An account shall be reactivated by the user anytime they want.

**Premium Account:**

26. A premium account users' profile shall be highlighted.
27. A premium account users' shall have access to premium features.

**Admin/Administrator:**

63. Administrators shall be able to see a flagged user's account.
64. Administrators shall be able to see a reported user's account.
65. Administrators shall be able to delete a user's account.
66. Administrators shall notify a flagged user about any removed content.

**Tags and Custom Tags:**

69. Custom Tags shall allow users to list special requirements.

**Matches:**

72. Matches shall be able to share photos via chat.
73. Matches shall be able to share videos via chat.

**Chat Window:**

83. Chat window shall allow users to sort matches by name.
84. Chat windows shall allow users to sort matches chronologically.

**Priority 3**

**Registered User (also referred to as user):**

4. Users shall be able to troubleshoot login problems.
5. Users shall be able to activate 2 factor identification.
6. Users shall be able to deactivate 2 factor identification.

**Unregistered User:**

17. Unregistered users shall have to confirm their account registration via a valid email.

**Premium Account:**

28. A premium account user shall have to pay for premium features.
29. A premium account user shall be able to save their payment information.
30. A premium account user shall be able to update their payment information.
31. A premium account user shall be able to cancel their subscription.

**Preferences:**

50. Preferences shall let users filter by country if they choose to meet online.
51. Preferences shall enable users to filter by city if they choose to meet online.

**Highlights of the Month:**

60. The Highlights of the Month shall contain one work that has the most votes for Uniqueness/Creativity.
61. The Highlights of the Month shall contain one work that has the most votes for Effort.
62. The Highlights of the Month shall contain one work that has the most votes for Quality.

**Matches:**

76. Matches shall be able to use default conversation prompts in chat.

**Chat Window:**

85. Chats with each match shall have the option to interact via audio call.
86. Chats with each match shall have the option to interact via video call.

**Customer Service:**

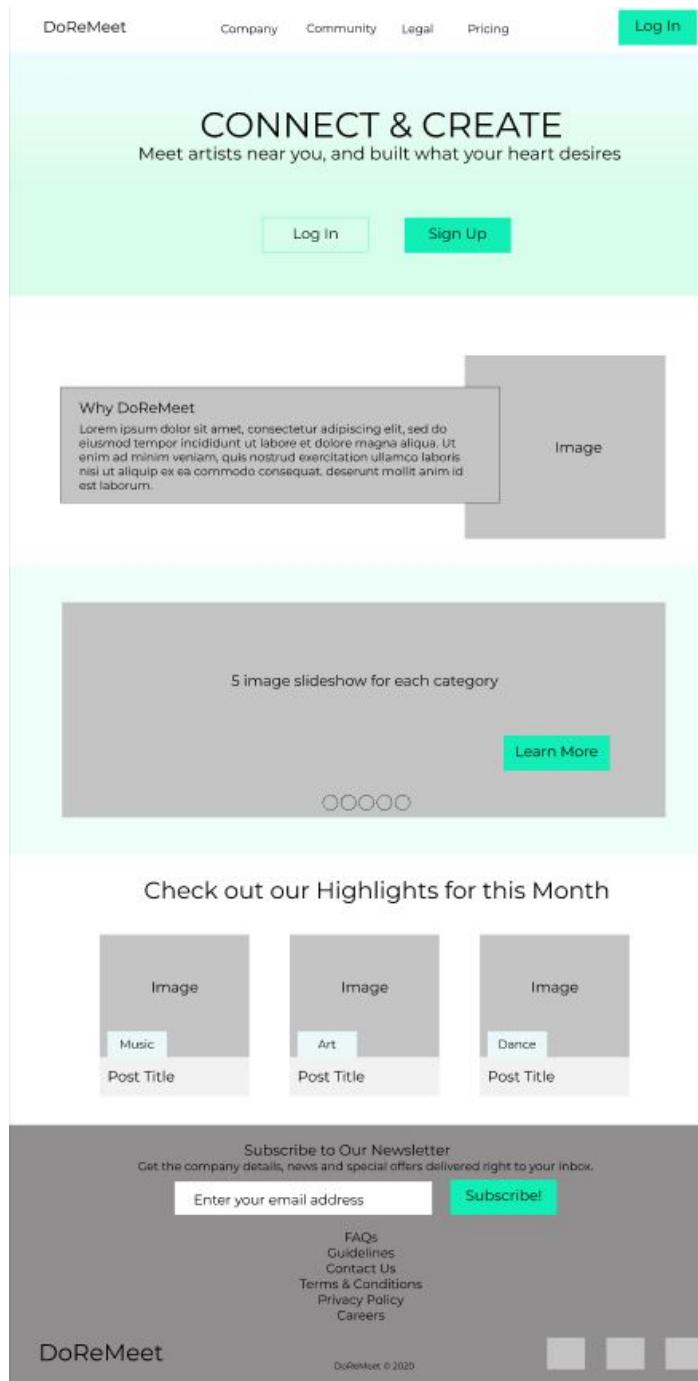
87. Customer Service shall be able to chat with users.

### 3. Wireframes based on Mockups/Storyboards V2 (detailed)

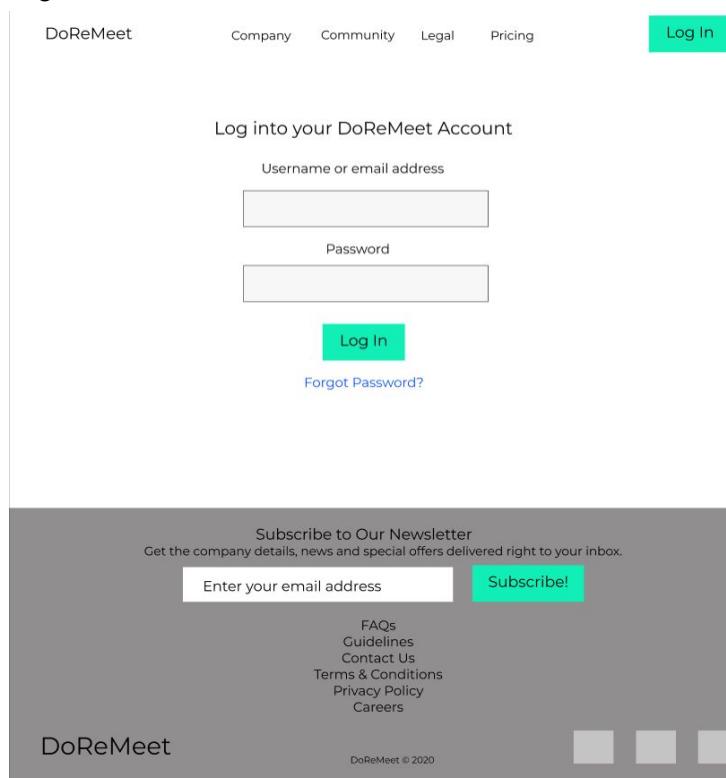
\***Update:** The Horizontal Prototype was updated to ensure that that site flow on the prototype is similar to that indicated in the wireframes

#### Pages:

##### Home Page:



## Log in:



The screenshot shows the DoReMeet login page. At the top, there is a navigation bar with links for "DoReMeet", "Company", "Community", "Legal", and "Pricing". To the right of these links is a green "Log In" button. Below the navigation bar, the text "Log into your DoReMeet Account" is displayed. There are two input fields: one for "Username or email address" and one for "Password". Below the password field is a green "Log In" button. Underneath the log in form is a link "Forgot Password?". At the bottom of the page, there is a dark grey footer section containing a newsletter sign-up form. It includes a placeholder "Enter your email address", a green "Subscribe!" button, and links for "FAQs", "Guidelines", "Contact Us", "Terms & Conditions", "Privacy Policy", and "Careers". The footer also features the "DoReMeet" logo and the text "DoReMeet © 2020" along with three small square icons.

Log into your DoReMeet Account

Username or email address

Password

Log In

Forgot Password?

Subscribe to Our Newsletter  
Get the company details, news and special offers delivered right to your inbox.

Enter your email address

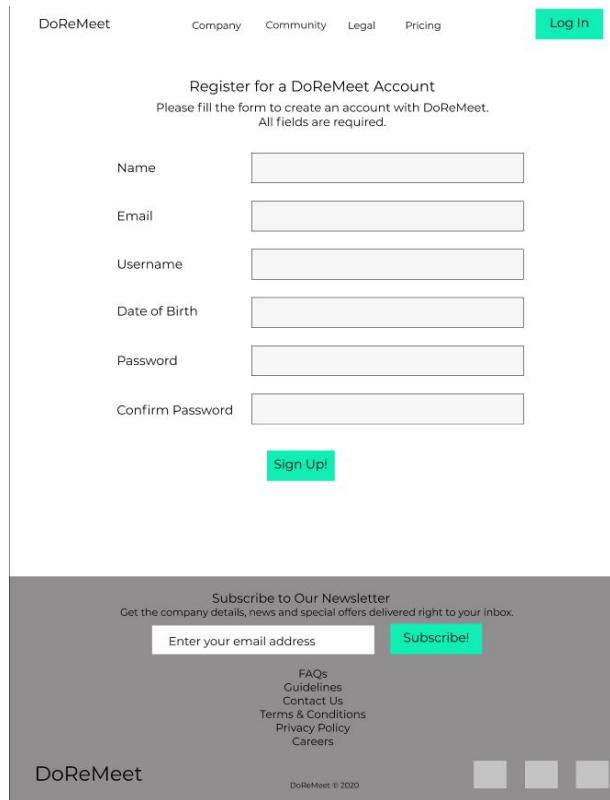
Subscribe!

FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy  
Careers

DoReMeet

DoReMeet © 2020

## Sign Up:



The screenshot shows the DoReMeet sign up page. At the top, there is a navigation bar with links for "DoReMeet", "Company", "Community", "Legal", and "Pricing". To the right of these links is a green "Log In" button. Below the navigation bar, the text "Register for a DoReMeet Account" is displayed, followed by the instruction "Please fill the form to create an account with DoReMeet. All fields are required." There are six input fields labeled "Name", "Email", "Username", "Date of Birth", "Password", and "Confirm Password". Below these fields is a green "Sign Up!" button. At the bottom of the page, there is a dark grey footer section containing a newsletter sign-up form. It includes a placeholder "Enter your email address", a green "Subscribe!" button, and links for "FAQs", "Guidelines", "Contact Us", "Terms & Conditions", "Privacy Policy", and "Careers". The footer also features the "DoReMeet" logo and the text "DoReMeet © 2020" along with three small square icons.

Register for a DoReMeet Account  
Please fill the form to create an account with DoReMeet.  
All fields are required.

Name

Email

Username

Date of Birth

Password

Confirm Password

Sign Up!

Subscribe to Our Newsletter  
Get the company details, news and special offers delivered right to your inbox.

Enter your email address

Subscribe!

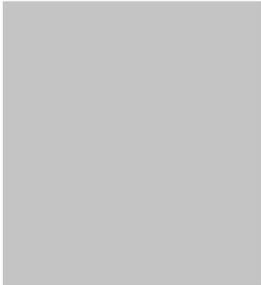
FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy  
Careers

DoReMeet

DoReMeet © 2020

## Matches:

DoReMeet    Community    Chat    Profile    Match    Settings    Log Out



**Robert, 26**  
**Bio:**  
 sddchb bchebc egcewch  
 ecic'hefc ed ceohfkfbc  
 ceeecnd eechrehcnd  
 chegecechcjcdjcmzyrgk. 😊

Dance   Salsa  
 Fun   Amateur   50 mi

Pass   Connect

All media files...

Report User   Instagram   Twitter   YouTube   Spotify

Subscribe to Our Newsletter  
 Get the company details, news and special offers delivered right to your inbox.

Enter your email address   Subscribe!

FAQs   Guidelines   Contact Us   Terms & Conditions   Privacy Policy   Careers

DoReMeet   DoReMeet © 2020

## Profile:

DoReMeet    Community    Chat    Profile    Match    Settings    Log Out



**Jack Smith**  
 Age: 26  
 D/o/B: 1/1/1994  
 Location: San Francisco, CA (Hidden)  
 Art Category: Film Making

Instagram   Twitter   YouTube

Indie   Director   Amateur   Fun   50 mi  
 Unpaid   Urgent

**BIO:**  
 sddchb bchebc egcewch ecic'hefc ed ghe;hc bciehcnbhebc  
 fchecjnbebc ceohfkfbc ceeecnd eechrehcnd  
 chegecechcjcdjcmzyrgk. 😊

All media files...

Community Posts

Post Title   Post Title

Subscribe to Our Newsletter  
 Get the company details, news and special offers delivered right to your inbox.

Enter your email address   Subscribe!

FAQs   Guidelines   Contact Us   Terms & Conditions   Privacy Policy   Careers

DoReMeet   DoReMeet © 2020

## Community:

The screenshot shows the DoReMeet community page. At the top, there's a navigation bar with links for DoReMeet, Community, Chat, Profile, Match, Settings, and Log Out. Below the navigation is a section titled "Community Page" with a sub-instruction: "Check out other peoples' art works or post some of your own amazing collaborations with fellow DoReMeet users." There's a search bar with placeholder text "Search for any post", a category button "Art", and a green "Search" button. A link "Share your work with us [here!](#)" is also present. The main content area displays several user posts in a grid format. Each post includes a placeholder image, a category label (e.g., Music, Art, Dance), a "Post Title" placeholder, and a "Dance" category label. One post is specifically labeled "Film". At the bottom of the page is a "POST" button.

**Community Page**  
Check out other peoples' art works or post some of your own amazing collaborations with fellow DoReMeet users.

Search for any post [Art](#) [Search](#)

Share your work with us [here!](#)

Image Post Title Music Post Title Art Post Title Dance Post Title

Dance Post Title

Film Post Title

[POST](#)

Subscribe to Our Newsletter  
Get the company details, news and special offers delivered right to your inbox.

Enter your email address [Subscribe!](#)

FAQs Guidelines Contact Us Terms & Conditions Privacy Policy Careers

DoReMeet DoReMeet © 2020

## Chat:

The screenshot shows the DoReMeet chat interface. At the top, there's a navigation bar with links for DoReMeet, Community, Chat, Profile, Match, Settings, and Log Out. The main chat area shows a conversation between two users: John and Adam. John's messages are on the left, and Adam's messages are on the right. Each message is preceded by a user profile picture and their name. Below the messages is a text input field with placeholder "Send a message" and a green "Send" button. A "Subscribe to Our Newsletter" section is located at the bottom of the page, identical to the one in the community section.

John lorem ipsum ...

Adam

Mary lorem ipsum ...

Adam lorem ipsum ...

Jane lorem ipsum ...

Send a message [Send](#)

Subscribe to Our Newsletter  
Get the company details, news and special offers delivered right to your inbox.

Enter your email address [Subscribe!](#)

FAQs Guidelines Contact Us Terms & Conditions Privacy Policy Careers

DoReMeet DoReMeet © 2020

## Settings:

The screenshot shows the 'Settings' page of the DoReMeet application. At the top, there are navigation links: DoReMeet, Community, Chat, Profile, Match, Settings, and Log Out. Below these are input fields for First Name, Last Name, gender selection (Male, Female, Other, Human), Email, Username, Password, Date of Birth, Phone Number, Art Category, Skill level, Minimum Age, Maximum Age, Gender Preference, Meeting Preference, Distance, and Tags (with a note about comma-separated values). A green button labeled 'Upgrade to Premium Account' is visible. Below the main form, there are 'Save Changes', 'Pause Account!', and 'Delete Account!' buttons. At the bottom, there's a newsletter sign-up section with a 'Subscribe!' button, and footer links for FAQs, Guidelines, Contact Us, Terms & Conditions, Privacy Policy, and Careers. The footer also includes the text 'DoReMeet © 2020' and three small icons.

## Premium Account Setup:

The screenshot shows the 'Premium Account Setup' page. At the top, there are navigation links: DoReMeet, Community, Chat, Profile, Match, Settings, and Log Out. A green box highlights the 'Upgrade your account' section, which states: 'and increase your chances of finding the right match!' followed by a list of benefits: Unlimited Likes, No ads, Highlighted Profile, Increased Search Radius, Rewind on Potential Matches, and Bigger Portfolio Size. Three subscription plan options are shown: '\$ 4.99 for 1 month', '\$ 49.99 for 6 month', and '\$ 119.99 for 12 month', each with a 'Select Plan' button. Below these are fields for Card Holder's Name, Card Number, Expiration Date (MM/YY), and CVV. A red box contains a note: 'You will be charged \$49.99 for one 6 month Subscription plan ending 06/2021. If you agree with the charges, please proceed with the upgrade. Enjoy your new features!' At the bottom, there are 'Upgrade to Premium Account' and 'End Premium Membership' buttons. The page also includes a newsletter sign-up section with a 'Subscribe!' button, and footer links for FAQs, Guidelines, Contact Us, Terms & Conditions, Privacy Policy, and Careers. The footer includes the text 'DoReMeet © 2020' and three small icons.

## Footer Pages:

**DoReMeet Guidelines**

These Guidelines and our Terms and Conditions are designed to ensure every user has a safe and enjoyable experience on DoReMeet. Failure to adhere to these guidelines may result in losing access to our platform.

**Upload only your own photos**

We take copyrights very seriously. If you don't own the rights to a photo or video, please don't post it.

**Respect all users**

We're a very diverse community. This means you should respect other people's beliefs, interests and property while on DoReMeet. DoReMeet takes a strong stance against hate speech, rude or abusive behaviour, bullying, and misogyny. You should behave the same way on DoReMeet as you would in real life. Additionally we encourage all of our users to report anyone who does not follow these behavioural guidelines. As a community rooted in kindness and respect, we expect all of our users to respect each other, themselves, and the DoReMeet staff.

**Subscribe to Our Newsletter**  
Get the company details, news and special offers delivered right to your inbox.

Enter your email address  Subscribe!

FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy  
Careers

DoReMeet © 2020

**DoReMeet** Company Community Legal Pricing Log In

**GUIDELINES**

**DoReMeet** Company Community Legal Pricing Log In

**FAQs**

**Need Help?**

Check the popularly asked questions, and if you still can't find the answer [Contact Us here](#)

Question 1   
Question 2   
Question 3   
Question 4   
Question 5   
Question 6

**Subscribe to Our Newsletter**  
Get the company details, news and special offers delivered right to your inbox.

Enter your email address  Subscribe!

FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy  
Careers

DoReMeet © 2020

**DoReMeet** Company Community Legal Pricing Log In

**PRIVACY POLICY**

**DoReMeet** Company Community Legal Pricing Log In

**Terms & Conditions**

**DoReMeet Terms and Conditions of Use**

Hey guys! Welcome to DoReMeet's Terms and Conditions of Use (these "Terms"). Our lawyers insist that we impose rules on users to protect all of our hard work. This is a contract between you and the DoReMeet Group (as defined further below) and we want you to know yours and our rights before you use the DoReMeet. Please take a few moments to read these Terms before enjoying the website, because once you access, view or use the website, you are going to be legally bound by these Terms (so probably best to read them first).

**DoReMeet RULES**

Before you can use our awesome website, you will need to register for an account ("Account"). In order to create an Account you must:

- be at least 18 years old; and
- be legally permitted to use the website by the laws of your home country.

You can create an Account via manual registration. For more information about what information we use and how we use it, please check out our Privacy Policy.

Unfortunately, we cannot allow you to use another person's Account without permission - that just wouldn't be fair! You'll have great fun on the website, but if you find that you've deleted or paused your Account or if you want to be removed from "Streaming" mode when you are logged in and clicking on the "Pause" or "Pause Account" link, Your Account will be deleted immediately, but it may take a little while for Your Content to be completely removed from the website. We will save your profile information in case you realise you miss us and you decide to restore your Account (which you can do within 30 days of de-activating your Account). If

**Subscribe to Our Newsletter**  
Get the company details, news and special offers delivered right to your inbox.

Enter your email address  Subscribe!

FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy  
Careers

DoReMeet © 2020

**DoReMeet** Company Community Legal Pricing Log In

**DoReMeet** Company Community Legal Pricing Log In

**DoReMeet** Company Community Legal Pricing Log In

# CONTACT US

Please fill out the form with any questions/concerns/suggestions you have, and we will get back to you as soon as possible.

Thank you for being a valued customer!

Name

Email

Message/Question?

[Send Message](#)

Subscribe to Our Newsletter

Get the company details, news and special offers delivered right to your inbox.

Enter your email address

[Subscribe!](#)

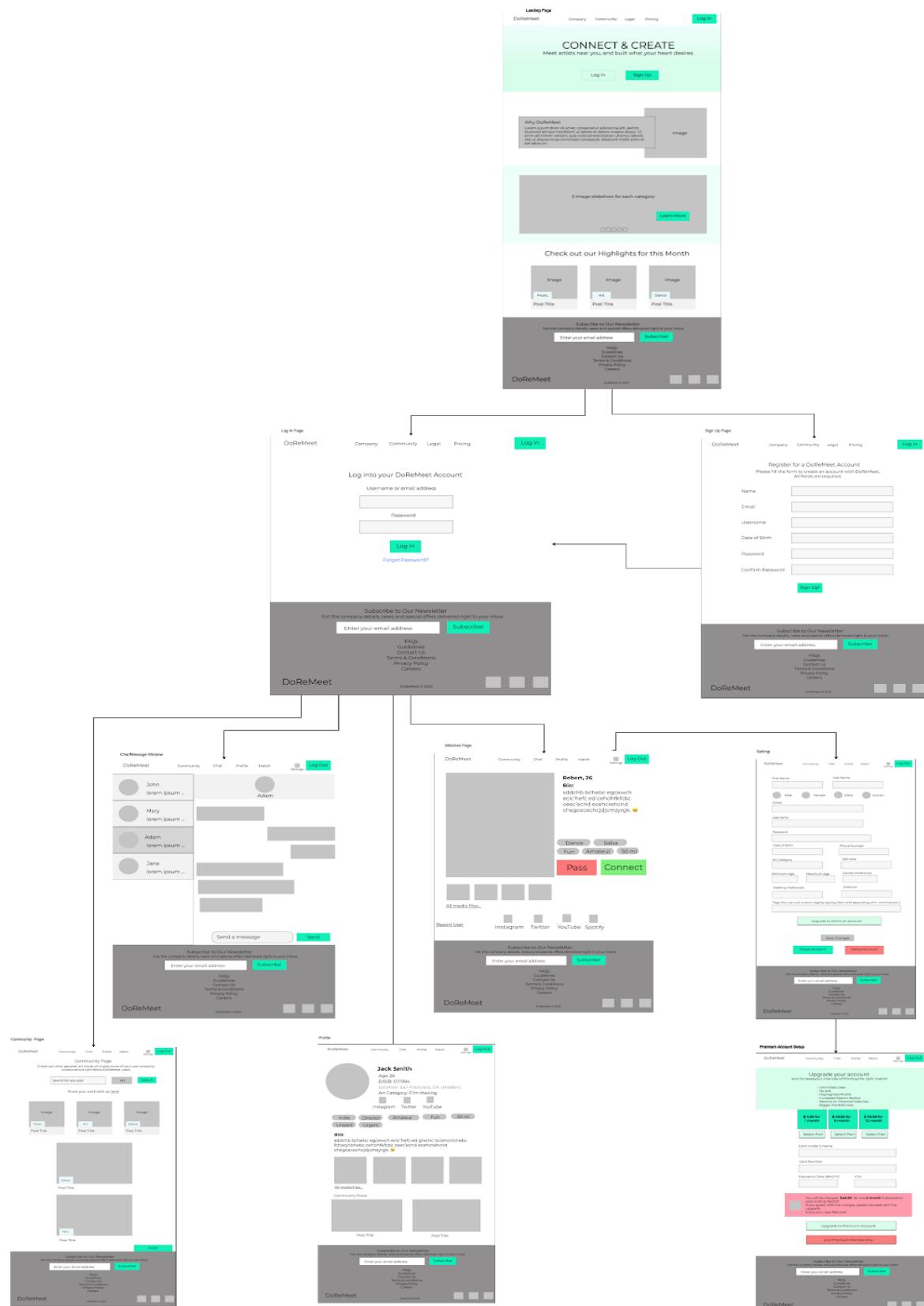
[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)  
[Careers](#)

DoReMeet

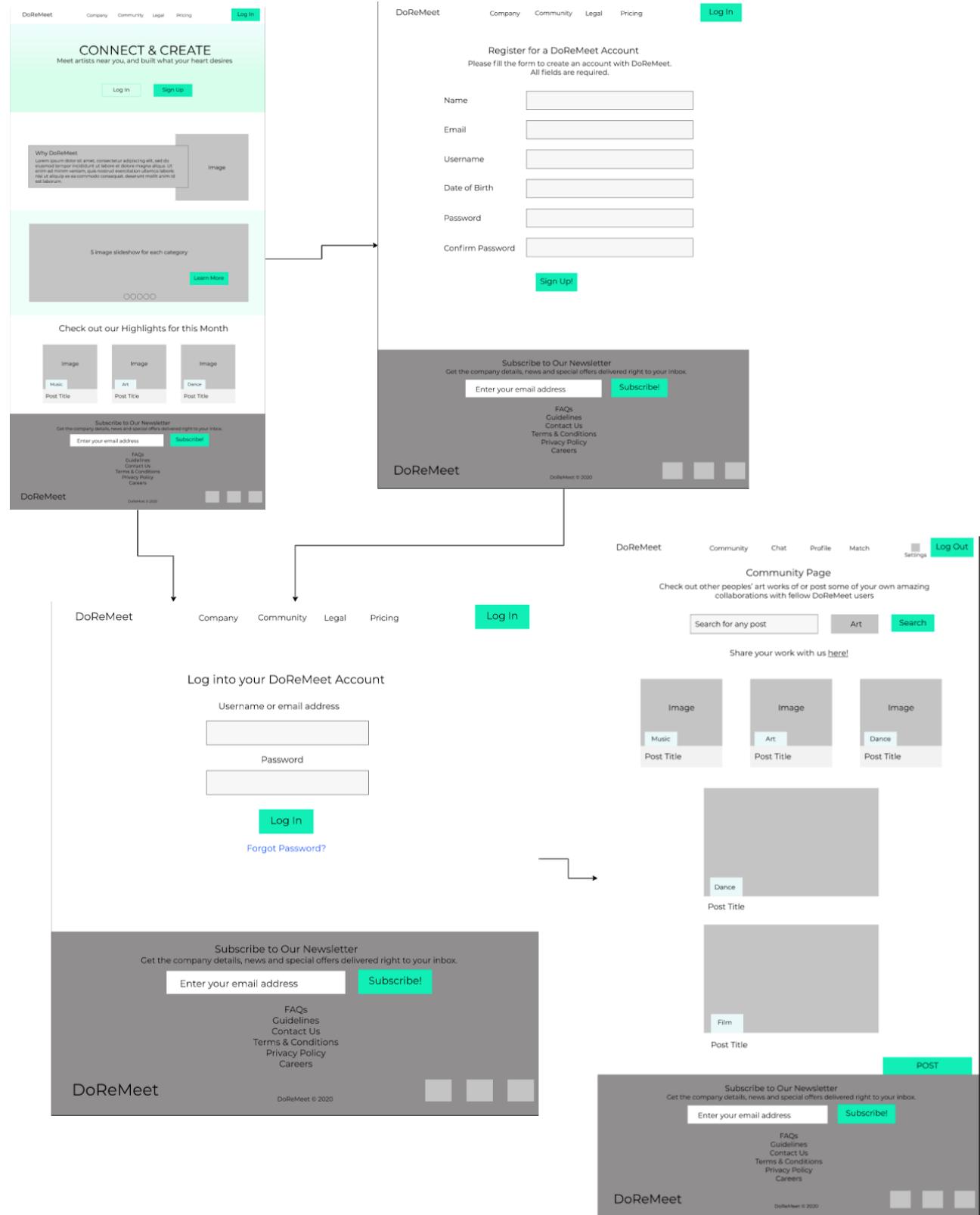
DoReMeet © 2020



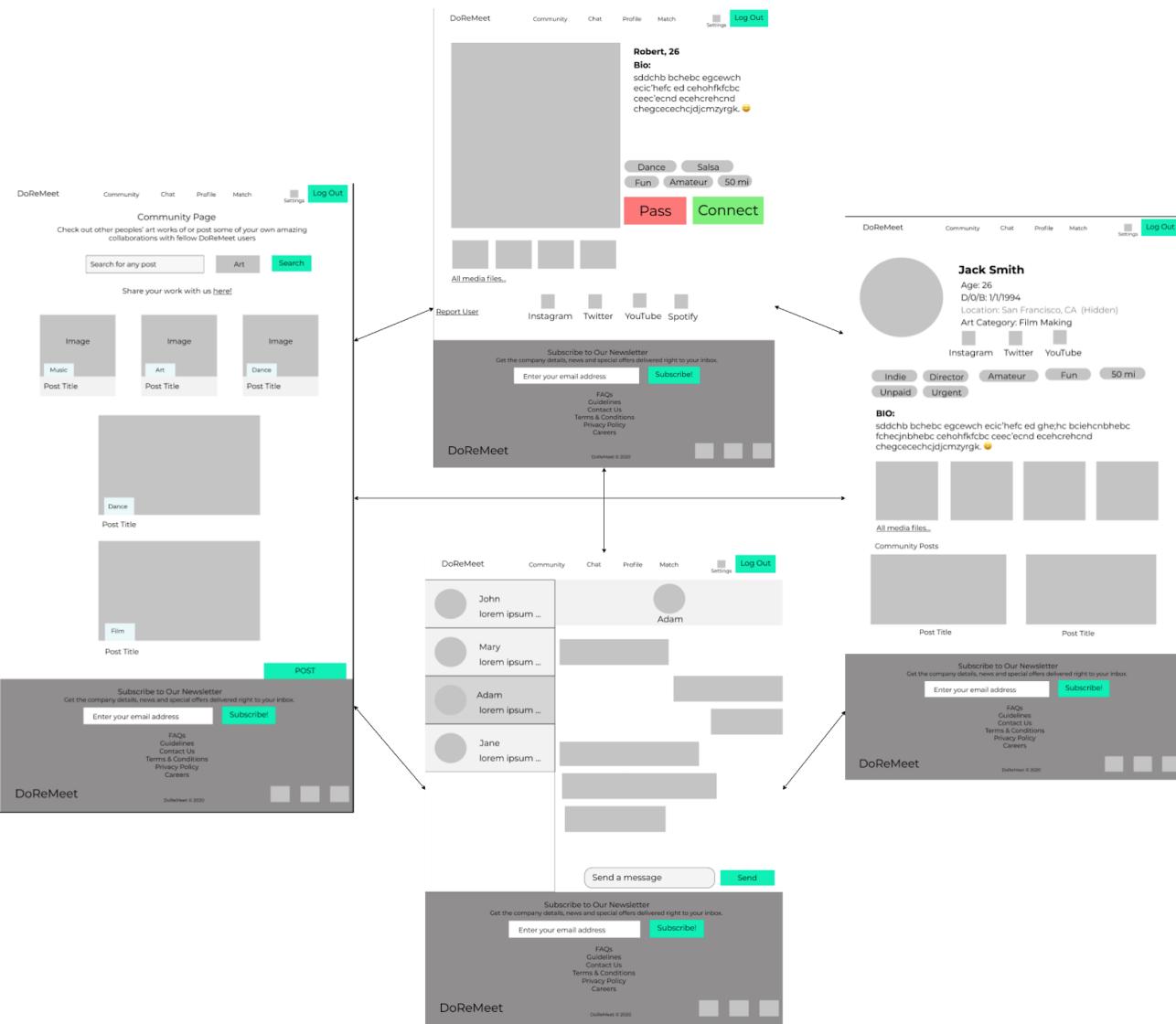
## General Flow:



## Sign up, Log In:



## User Pages:



## Settings:

The wireframe illustrates the user flow through the DoReMeet application, starting from the Community Page and ending with a newsletter subscription screen.

**Community Page:** Shows a search bar and placeholder cards for 'Image', 'Post Title', 'Dance', and 'Post Title'.

**User Profile View:** Displays a user's profile picture, name (Robert, 26), bio, and interests (Indie, Director, Amateur, Fun, 50 mi). It includes a 'Pass' and 'Connect' button, and links for 'All media files...', 'Community Posts', and social media integration (Instagram, Twitter, YouTube, Spotify).

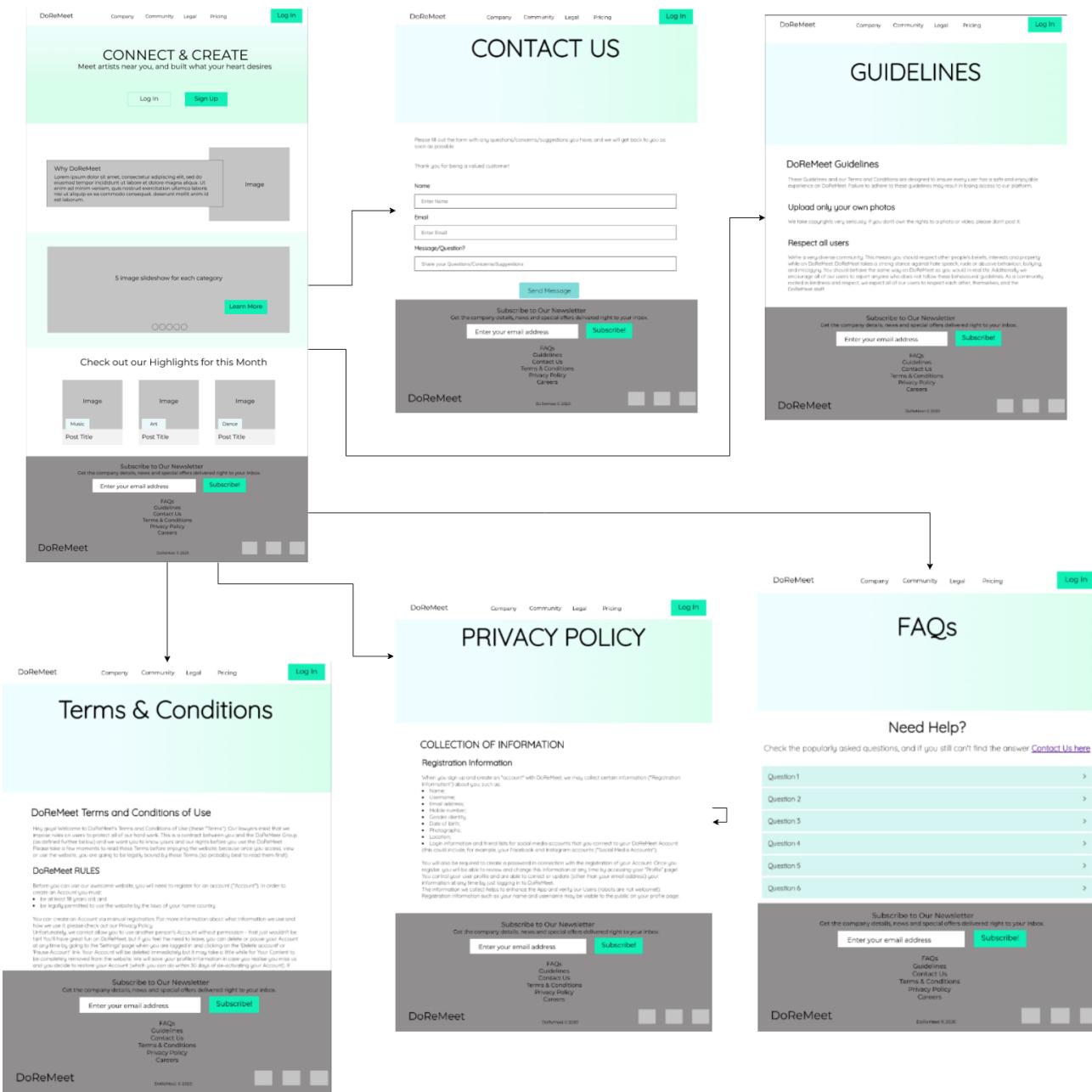
**Search Results:** Shows a list of users (John, Mary, Adam, Jane) with their names, profile pictures, and short bios.

**Account Settings:** Allows users to edit their profile information (First Name, Last Name, gender, email, username, password, date of birth, phone number, skill level, art category, maximum age, gender preference, meeting preference, distance, and tags). It also provides options to 'Upgrade to Premium Account', 'Save Changes', 'Paul Account!', or 'Delete Account!'. A 'Subscribe to Our Newsletter' section is also present.

**Upgrade your account:** Offers three subscription plans: \$4.99 for 1 month, \$49.99 for 6 months, and \$199.99 for 12 months. It includes a summary of benefits like 'Unlimited matches', 'No ads', 'Highlighted profile', 'Increased Search Radius', and 'Revised on Potential Matches'. It also shows a note about being charged \$49.99 for a 6-month subscription and a link to 'Upgrade to Premium Account'.

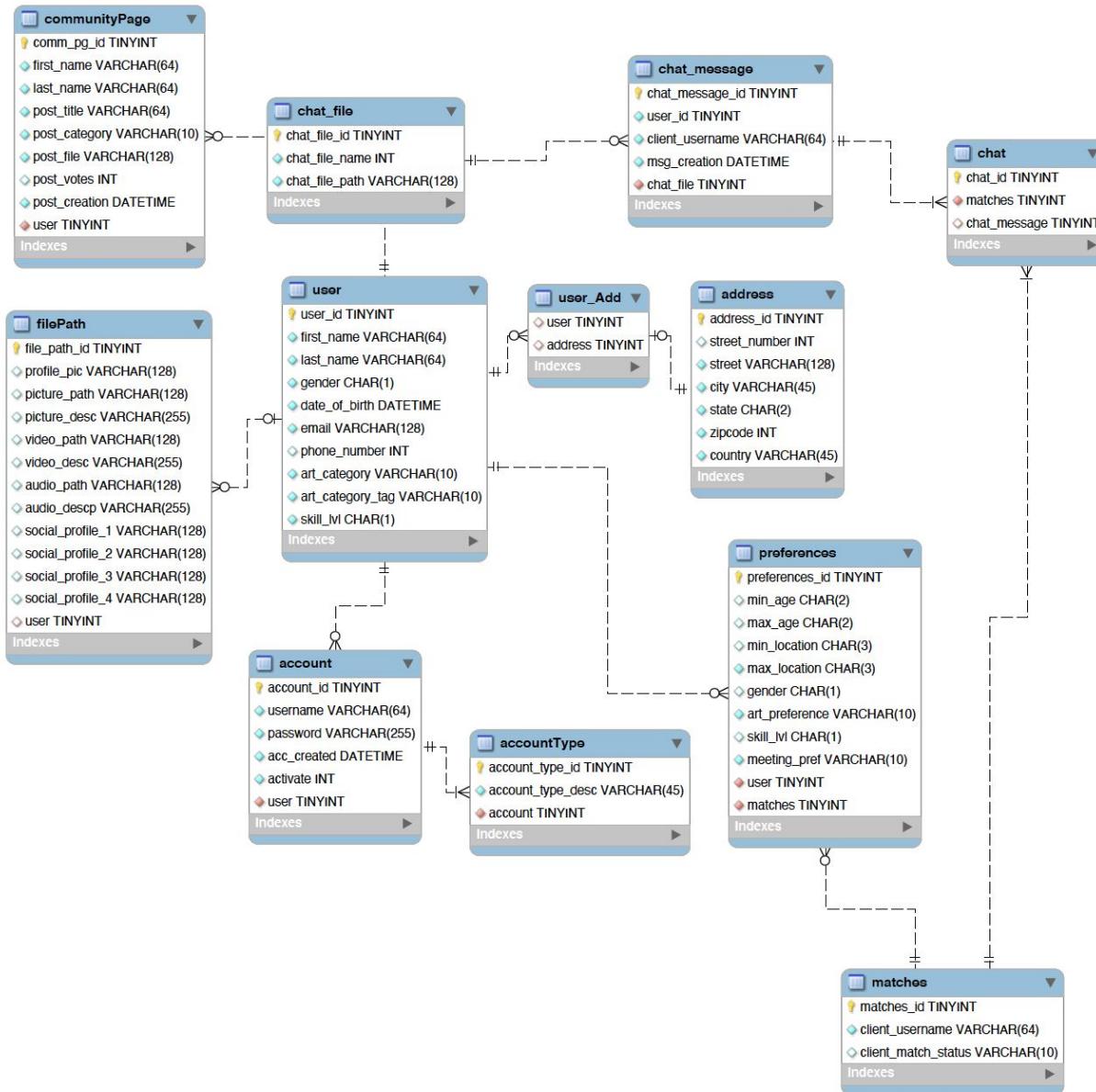
**Newsletter Subscription:** A final step where users can enter their email address to subscribe to the newsletter.

## Footer Links (available from all pages):



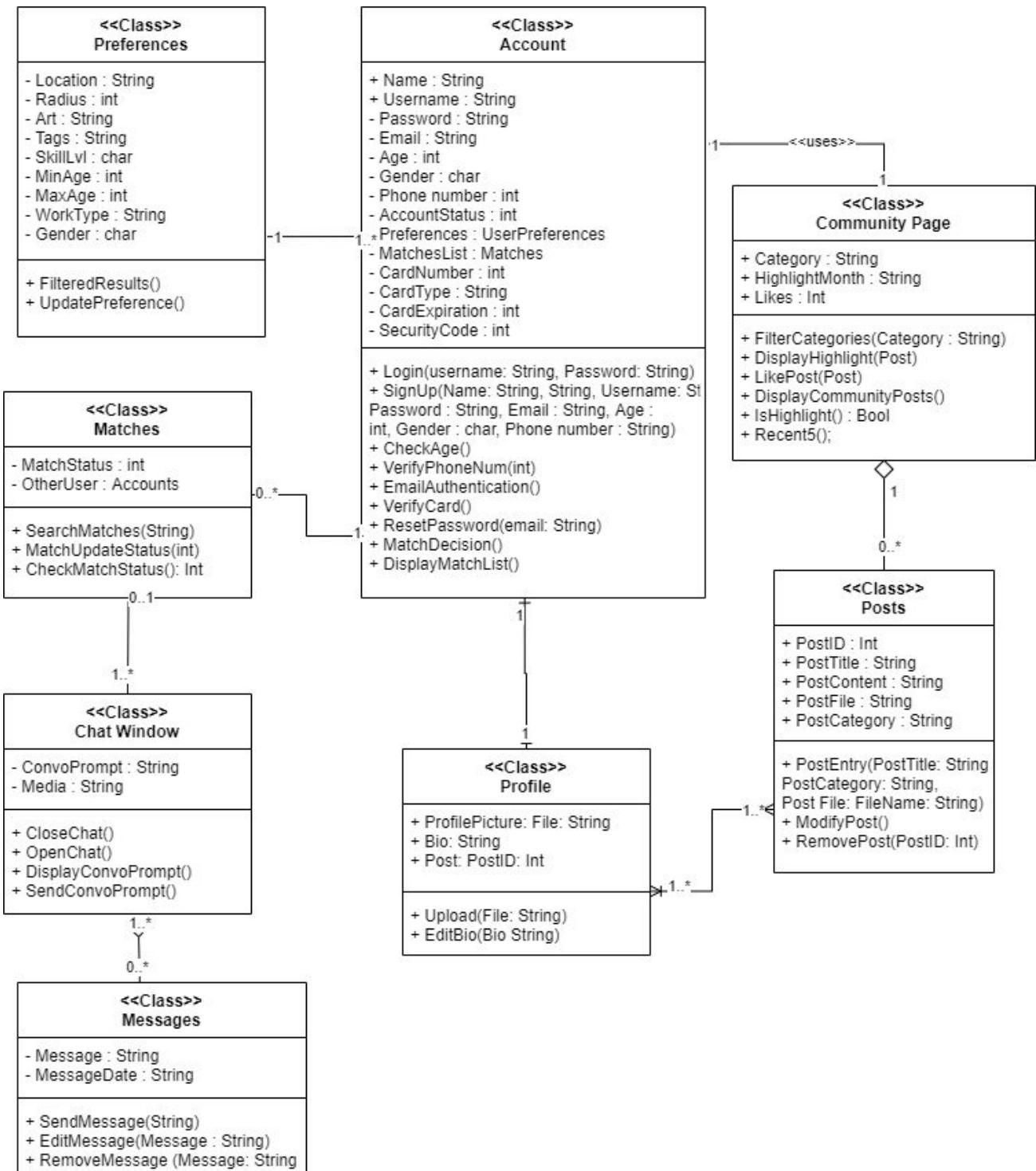
## 4. High level database architecture and organization V2 (detailed)

EER Diagram (made using MySQL Workbench)

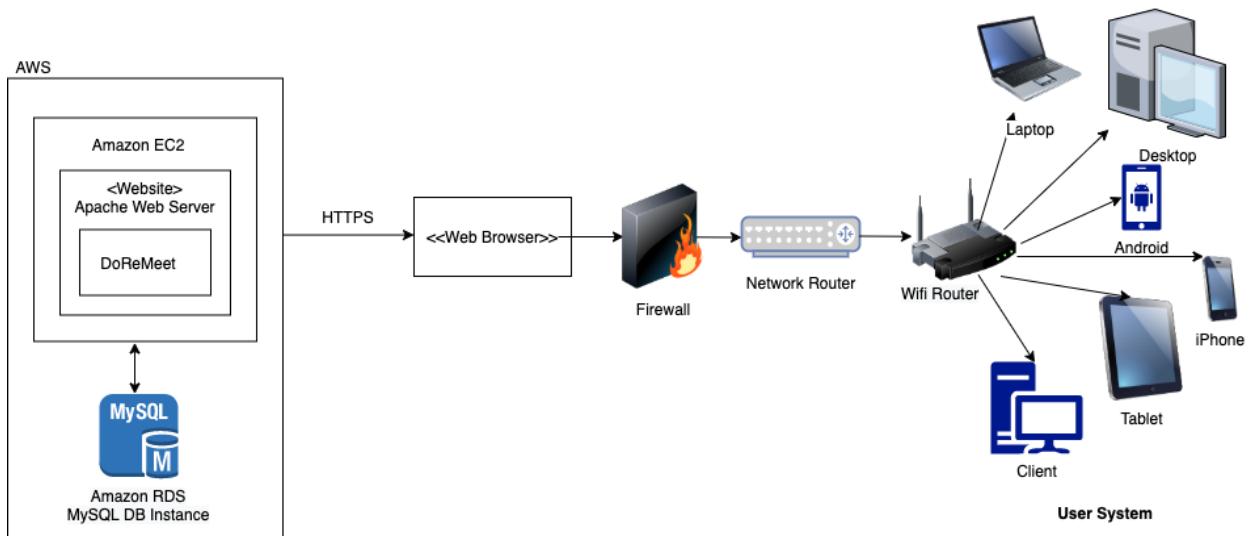


## 5. High Level Diagrams V2 (detailed)

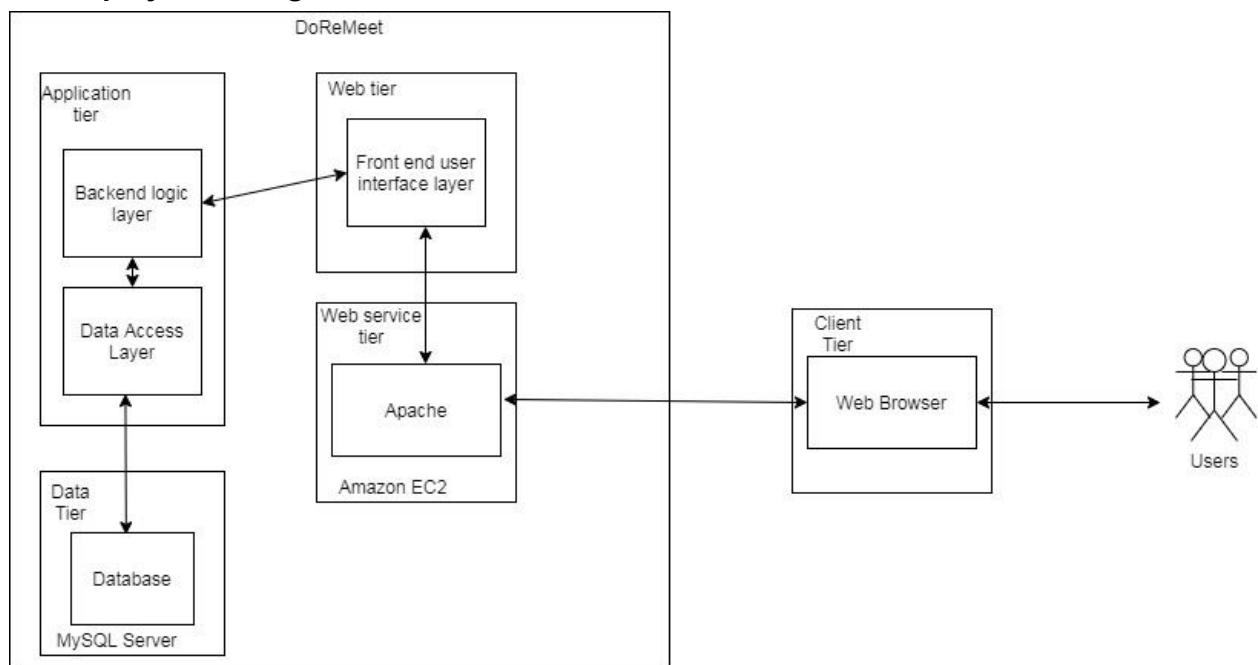
**UML Diagram**



## Network Diagram



## Deployment Diagram



## 6. List of Contributions

Member	Role	Contributions
Nimiksha	Team Lead & Documentation Lead	<u>Documentation:</u> Data Definitions Functional Requirements Wireframes Network Diagram <u>Horizontal Prototype:</u> Frontend for Landing Page, Footer Pages, Log In/Sign Up page Reusable site components - Card Elements, Footer, NavBar, Banner, Buttons Website responsiveness HT access configuration for the website on Apache Server (Research) Code Review and cleanup
Meet	Backend and Database Lead	<u>Documentation:</u> Data Definitions Functional Requirements EER Diagram <u>Horizontal Prototype:</u> Set up MySQL Database and connect it to backend Set up AWS RDS Database Connect RDS to local database SQL query planning and optimization Location API setup (to be completed in next milestone) Backend Configurations, Implement Router for Backend Sign Up Page configurations
Luke	Frontend Lead	<u>Documentation:</u> Data Definitions Functional Requirements Network and Deployment Diagram <u>Horizontal Prototype:</u> Frontend for Community Page, Profile Page, Setting Page, Premium Account Setup Connecting frontend to backend Host website on EC2 instance HT access configuration for the website on Apache Server (Implementation) Code cleanup and organization Documentation
Mike	GitHub Master & Backend Engineer	<u>Documentation:</u> Data Definitions Functional Requirements UML Diagram <u>Horizontal Prototype:</u> Connection between frontend, backend and database (sending and receiving queries/requests from the frontend, getting data from the database, sending

		<p>results back to frontend)</p> <p>Connect backend on EC2 instance</p> <p>HTaccess configuration for the website on Apache Server (Research)</p> <p>Login, Logout and Sign Up backend configurations</p> <p>Setting page configuration with backend and database.</p> <p>Get 5 most recent posts for Community page</p> <p>Post and Receive images from database</p> <p>Github management and cleanup</p>
Jungsun	Database & Backend Engineer	<p><u>Documentation:</u></p> <p>Data Definitions</p> <p>Functional Requirements</p> <p>Documentation Review</p> <p><u>Horizontal Prototype:</u></p> <p>Log in and Sign up backend configuration</p> <p>Location API setup (to be completed in next milestone)</p> <p>Connect RDS to local database</p>
Vincent	Server Admin & Frontend Engineer	<p><u>Documentation:</u></p> <p>Data Definitions</p> <p>Functional Requirements</p> <p><u>Horizontal Prototype:</u></p> <p>Frontend for Match Page, Chat Page, Community Page</p> <p>Recent post timeline setup on Community page</p> <p>Post and Receive images to and from database</p> <p>Frontend to backend connection</p> <p>EC2 instance maintenance</p> <p>Site responsiveness and cleanup</p>

The team worked on all sections of Milestone 3 together during Zoom meetings, and collaboratively created this report.

# SW Engineering CSC648/848 FALL 2020

## DoReMeet

### Team 02

#### Milestone 4

Nimiksha Mahajan ([nmahajan@mail.sfsu.edu](mailto:nmahajan@mail.sfsu.edu)) - Team Lead

Meet Patel ([mpatel6@mail.sfsu.edu](mailto:mpatel6@mail.sfsu.edu)) - Backend & Database Lead

Luke Chang-Knezevich ([lchangknezevich@mail.sfsu.edu](mailto:lchangknezevich@mail.sfsu.edu)) - Frontend Lead

Mike Bolanos ([mbolanos1@mail.sfsu.edu](mailto:mbolanos1@mail.sfsu.edu)) - GitHub Master & Backend Engineer

Jungsun Eoh ([jeoh@mail.sfsu.edu](mailto:jeoh@mail.sfsu.edu)) - Database & Backend Engineer

Vincent Tse ([vtse3@mail.sfsu.edu](mailto:vtse3@mail.sfsu.edu)) - Server Admin & Frontend Engineer

#### History Table

Version	Date	Comments
M4V2	12/15/2020	Submission updated with Feedback
M4V1	12/10/2020	Initial Submission
M3V2	12/05/2020	Submission updated with Feedback
M3V1	11/19/2020	Initial Submission
M2V2	12/05/2020	Submission updated with Feedback
M2V1	11/01/2020	Initial Submission
M1V2	10/11/2020	Submission updated with Feedback
M1V1	09/30/2020	Initial Submission

## **Table Of Contents**

<b>Product Summary</b>	<b>2</b>
<b>Usability Test Plan</b>	<b>4</b>
Test Objective	4
Test Description	6
Usability Task Description	7
Usability Test Table	8
Questionnaire	9
<b>QA Test Plan</b>	<b>10</b>
Test objectives:	10
HW and SW setup:	10
Feature to be tested:	11
QA Test Plan:	11
QA Test Browser Compatibility:	12
<b>Code Review</b>	<b>13</b>
<b>Self-Check on best Security Practice</b>	<b>27</b>
<b>Self Check: Adherence to Non Functional Specs</b>	<b>32</b>
<b>List of Contributions</b>	<b>35</b>

## **1. Product Summary**

**Product Name:** DoReMeet

Our product, DoReMeet, aims to help artists build a community by providing them a platform where they can find collaborators, mentors, and/or friends. Artists, be it singers, dancers, painters, photographers or film makers, can use our product to find people that are close to their geographical location based on their personal preferences. They can choose to connect with fellow artists and collaborate on projects for work, fun or both.

The functions to be delivered in our final product are as follow:

1. Sign Up/Register - Unregistered users shall be able to sign up or register an account on DoReMeet from the landing page. To sign up, users will have to input their email and some other information like their name, date of birth, gender, art category etc.
2. Log In/LogOut - Registered users have the ability to log into their account and log out at any time using the credentials they signed up with.
3. View Potential Matches - Users will be able to see potential matches based on their preferences. They can choose to “Connect” with the user or “Pass” on them.
4. Filter Potential Matches based on Preferences - Users will be able to set their personal preferences for the art category, meeting in person or online, skill level, age and gender. The potential matches they see will be filtered based on these preferences, thus giving them more targeted search results.
5. Expand and Reduce Search Radius - If the user chooses to meet in person, they have the ability to set a radius limit for their potential matches. They can choose to expand or reduce their search radius, and will be shown potential matches only within the specified distance.
6. Matches - Users will be matched with people who they choose to connect with, if the other person also wants to connect with them. Once connected, the user has the ability to chat with the match and collaborate as they deem fit.
7. Chat - Users will be able to send and receive messages to/from the other users they've matched with.
8. Update Preferences - The user has the ability to update their preferences for art category, meeting (in person or online), minimum and maximum age, gender and

skill level. Updating their preferences will also update their potential match results to meet their new search criteria.

9. Search for Posts - Users also have the ability to search for posts on the Community page based on the Post Title and the Art Category

#### **Unique Features:**

1. Simple Connect / Pass Match Page - A simplified, not overwhelming space for users to pick and choose from artists nearby and around the globe. Unlike our competitors, we don't show users a bulk of fellow artists to pick from; instead we show that one person at a time and users get to directly see that individuals work. They can view the profile, and if interested choose to "Connect" with the user or they can simply "Pass" and will not be matched with the user.
2. Community Page - People will be able to see the Community page, with the recent posts made by the platform users and the Highlights of the month. Only users who have a registered account and are logged in will be able to post to the Community page to share their collaborations.
3. Community highlights - The community page will have a highlights section to show most liked posts for the month, based on user votes. Only registered users will be able to vote on the posts.

**Product URL:** <http://ec2-13-52-247-220.us-west-1.compute.amazonaws.com/>

## **2. Usability Test Plan**

### Test Objective

DoReMeet aims to provide users a platform to find and connect with local (or global) artists to collaborate, by offering them a list of potential matches based on their personal preferences(based on art category, distance, skill level, age, gender). With our product we are extending the practice of instinctual decision-making via a quick, simple yes or no to the art community, and help them find other artists near them. Given this mission, usability testing is crucial for us, because we want to ensure that we are in fact giving the users a simple, not overwhelming, easy to navigate platform. We need some core functionalities implemented in our product, and need to ensure that they serve our product's purpose; the main feature we need usability testing for are as follows:

1. ***Match page shows potential matches to connect or pass on.***  
The very idea of our product is to offer a list of artists as potential matches to the user, based on their preferences. Therefore it is crucial to test the Match page to see that the users understand the idea of the page, and can instinctively navigate through it. We want to test that the users can view the potential matches' profile, see their work and then choose to "Connect" or "Pass" based on their interest.
2. ***The Community page shows Recent Posts made by DoReMeet users, and one can search for posts based on the Post Title and Art Category***  
Our product aims to provide a space for artists to build a collaborative community and the Community Page promotes that idea. We want to see that the users (both registered and unregistered) are able to see the Recent Posts, and see the different collaborations made by DoReMeet users, thus encouraging them to also use the platform and submit their own collaborations. Also, we are offering a search functionality on the Community page so that it is easy to sift through all the posts made. Via testing, we want to make sure users understand that they can view posts of a certain art category or look for a particular post by its title.
3. ***Users can send and receive messages to/from their matches on the Chat page.***  
The only way for people to actually connect with other artists and start collaborating is by interacting with them, and the Chat page provides that functionality. We want to check that the users understand they can only chat with people they match with. Also, the chat page needs to be tested to ensure that users are able to both send and receive messages to and from their matches.

**4. Users can update their *Profile* (bio, media, tags, preferences)**

DoReMeet's mission is to provide a customized experience for all its users, so we want the users to be able to update their own profile, whenever they want. Testing this page is important because users need to see that they can easily add/edit their bio, add/remove media files and the tags they choose and their preferences for the potential matches. Testing this feature is important because users should be able to easily navigate updating their profile/preferences, and should be able to do so whenever they want.

**5. Users will be able to get potential matches based on their *location preference* (expand and reduce the search radius)**

One of the key features of DoReMeet is that users can find people based on their location preferences. Users are able to adjust the distance for their potential matches. With testing, we want to make sure that the users know that they have the option to share their location with our product and set a minimum or maximum distance for their potential matches' preference. Also we need to make sure that the user knows they can adjust the radius anytime, or just stop sharing their location altogether if they like.

## Test Description

### *Usability Metric - Satisfaction*

System Setup: DoReMeet is compatible with Safari, Chrome, Mozilla Firefox and Edge. For usability testing purposes, we are getting the product tested by one of our teammates and by another person who was not involved in the development of the product, to ensure that our product in fact meets the satisfaction requirements. The tester will be given the product URL, and we will let them create an account, so that they can access all the features of our product. We'll see if they are able to update their profile, and set their preferences and location settings to get potential matches based on their interest. We'll observe how they use the Match page and if they are able to go through a list of potential matches by choosing to "connect" with or "pass" on them. We'll also check if they can navigate through the Community page, search for posts, and use the Chat page (send messages to anyone they match with).

Starting Point: The starting would be the landing page for DoReMeet (<http://ec2-13-52-247-220.us-west-1.compute.amazonaws.com/>) where the tester can start by creating an account, or they can go to the Community page to look at posts made by the users.

Intended Users: The intended users for this product are artists who are looking for fellow artists, in the Art, Music, Dance, Film or Photography category, to collaborate with. The users can be of any age over 18 and be of part of any gender, caste, race demographic. Also, the user can be a novice, amateur, expert or professional artist in their category, and can be looking for collaborators for work or for hobby/passion projects.

URL: <http://ec2-13-52-247-220.us-west-1.compute.amazonaws.com/>

What is to be measured: The test is supposed to measure how a user feels about our website, how they like the look of it and how easily they can navigate through the different pages. We want to measure on Likert scale how intuitive they find the Match page, navigating through potential matches, the Community pages along with all the other pages. We'll see if a user can recognize all the given functionalities and find out if they are satisfied with their experience, or if anything overwhelmed or confused them. Overall website friendliness and navigation will also be measured.

## Usability Task Description

Open the website <http://ec2-13-52-247-220.us-west-1.compute.amazonaws.com/> in any preferred browser that is supported by the product (Chrome, Safari, Mozilla Firefox or Edge) and create an account by signing up. Once you're logged in, test the following:

Task	Description
Matches	Navigate the match page, view the potential match's profile and if you like it, connect with them but if you don't pass on them
Community	Go to the Community Page and check out the Recent posts. Also, search for posts based on any art category. Post to the Community page under any of your preferred art categories.
Chat	Send a message to anyone you've matched with
Profile/Preferences	Update the profile bio, add a media file and update the art category preference. Also, update the skill level specified for the account.
Location (expand/reduce search radius)	Allow the product to use your location, and set the search radius for potential matches to be 10 miles. Check if the results displayed on the Match page are in fact within 10 miles

## Usability Test Table

Test / Use Case	% Completed	Errors	Comments
Matches	20%	Cannot see potential matches or connect/pass them	No matches based on the set preferences visible
Community	95%	Can't see all images	Only certain images are visible; not all uploaded images can be seen
Chat	50%	Can't receive messages from matches	If there are no matches, there is no message prompt as to why we can't see messages. Don't see any messages or directions
Profile/ Preferences	80%	Media files and users' Community posts not linked	Can't see the posts that the user made on the community page.
Location (expand/reduce search radius)	60%	Cannot expand or reduce search radius.	Can't adjust the desired match radius and there is no way to see how far a potential match is

## Questionnaire

Statements	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Match page shows potential match results based on your set preferences.	1	2	3	4	5
It is easy to either connect with or pass on a potential match.	1	2	3	4	5
It is easy to navigate through the community page and look for posts.	1	2	3	4	5

In this questionnaire, there will also be an optional *Comments section*, where the testers will have the ability to share any comments/suggestions they have.

We will use the results from usability testing to gauge the overall satisfaction with our product and get information about what we can potentially do or not do to enhance the user experience. This will help us not only make our product better, but also help with future marketing by giving us better insights on what the user needs.

### **3. QA Test Plan**

#### **Test objectives:**

For our QA testing we are testing some of the core functionalities of DoReMeet that are required for a smooth user experience. With this testing, we'll ensure that the user is able to access DoReMeet and navigate through at least the basic features that the website provides. We will be testing the functionality and effectiveness of some key non-functional requirements of the website, and make sure that these features work properly, as required for all users on all devices and browsers.

The first thing we are testing is browser compatibility, to ensure that the user is able to use DoReMeet on all major web browsers, and their most commonly used versions.

The next feature we are testing is that only eligible users are able to create an account on DoReMeet, in order to be in compliance with our legal terms and also to ensure that we are not exposing minors to any content that is not approved by their parents/guardians.

The third feature we are testing is our login functionality to make sure that registered users are able to access their account and enjoy the features offered by DoReMeet. Also to ensure that we are upkeep transparency and giving users a chance to make informed decisions, we'll be testing to make sure every time a user leaves the site by clicking on a third-party link, we give them an alert and let them know that they are leaving the DoReMeet space. Our website is not responsible for any cookies/cache collected by the third party and we want our users to be aware of that.

Lastly for the QA testing we'll also check to make sure that the website asks the users every time if they want to share their location, and check that no user information is being collected with the users explicit approval.

#### **HW and SW setup:**

Our product is accessible on any machine (MacOS, Windows or Linux) that is connected to the internet. It is also available on iPhones and Android devices. The website is compatible with

- Chrome, version 80.0 - 85.0
- Safari, version 10.0 - 14.0
- Mozilla Firefox, version 78 - 81
- Edge, version 85

No additional software/libraries need to be running on the client side.

### **Feature to be tested:**

To ensure that we are providing a service that is accessible, secure and functional, we are choosing to test the following critical functionalities.

1. The site shall be compatible with Safari (version 10.0 - 14.0), Chrome (version 80 - 85.0) and Mozilla Firefox (version 78 - 81) and Edge (version 85).
2. Users should be minimum 18 years of age to create an account.
3. The website shall warn users when clicking on links that take them to another website.
4. The user shall be required to enter both password and username to log in.
5. Users' geolocation shall only be known by the website if they allow it.

### **QA Test Plan:**

Test #	Test Title	Test Description	Test Input	Expected Output	Test Results
1.	Sign up - Must be at least 18 years of age	Any user trying to create an account must be at least 18 years old. If they are under 18, they should get a warning and not be able to create an account.	For Date of Birth. enter a date in 2003 or after	Error message letting the user know they do not meet DoReMeet's age requirement	PASS
2.	Leaving website message	The user should get a warning message if they click on a link that takes them to a third party website	Click on the Spotify link in a potential matches' profile	An alert message saying that the user is leaving DoReMeet and visiting a third party site	PASS
3.	Allow location use	The website shall explicitly ask user if they "Allow" the site to use their location, and do so only if the user agrees	Log into your DoReMeet account	Get an alert message asking if the user is allowing access to their location to get potential matches. Users can choose yes or no.	FAIL *not yet implemented on the actual product website

As per the CTO's suggestions, we went back for a brief session of QA testing and through iterative feedback, updated the UI of the features being tested.

For the Sign Up check for a user being at least 18 years of age, we collectively decided that along with just validating the age on the backend, we'll also display an error message on the frontend. We did a QA test run for it, any for any user under 18 trying to register, the website displays an appropriate error message.

As for the location use test, we unanimously decided that the user will have the option to allow location or deny access on the Login and Sign Up pages so they are in fact aware of the information they are sharing, and have the chance to make informed decisions. We did a QA test to ensure that the user gets a pop up message asking them about location access, and also ensured they can't proceed without making a choice, so that at no point the website accesses information that the user didn't approve of.

#### **QA Test Browser Compatibility:**

Test #	Browser	Compatibility (Yes/No)
1.	Chrome, version 80.0 - 85.0	Yes
2.	Safari, version 10.0 - 14.0	Partly. Some components do not display correctly
3	Mozilla Firefox, version 78 - 81	Yes
4.	Edge, version 85	Yes

## **4. Code Review**

### **a. Coding Style**

The coding style we've used is:

- Careful indentation and use of tabs instead of spaces
- Camel case for naming
- Descriptive names for variables and functions
- Code organized in folders categorized by their functionality (Auth, NavigationBar, Footer, card component etc.)
- /\* Comments \*/ as header for every file
- / inline comments within the code
- Modular functions

### **b. Code**

For the Code Review we chose to get our Community Page code reviewed since it is a core functionality of our product. The Community page is also a unique feature that we provide so that artists can share their collaborations and build a community with other DoReMeet users.

#### **Code Review request email:**

CSC648 Team 02 Code Review ➔ 

Luke Chang-Knezevich

to Nimiksha ▾

Hi Nimiksha,

Here's the code from our CommunityPage.js for the code review.

```
/*
**CSC 648 Team 02 DoReMeet
**File: CommunityPage.js
**Desc: Contains all the code needed for the community page. Calls on the backend through axios and
recieves json responses that it then displays accordingly on the page.
*/

import React, {useEffect, useState} from 'react';
import '../App.css';
import './CommunityPage.css';
import axios from 'axios';
import Navbar from '../components/Navbar/Navbar';
import Highlights from '../components/Highlights/Highlights';
import Footer from '../components/Footer/Footer';
import { BrowserRouter as Router } from 'react-router-dom';

const CommunityPage = (stateObj) => {
```

### Code to be reviewed:

```
/*
**CSC 648 Team 02 DoReMeet
**File: CommunityPage.js
**Desc: Contains all the code needed for the community page. Calls on the backend
through axios and receives json responses that it then displays accordingly on the
page.
*/

import React, {useEffect, useState} from 'react';
import '../App.css';
import './CommunityPage.css';
import axios from 'axios';
import Navbar from '../components/Navbar/Navbar';
import Highlights from '../components/Highlights/Highlights';
import Footer from '../components/Footer/Footer';
import { BrowserRouter as Router } from 'react-router-dom';

const CommunityPage = (stateObj) => {

    const onChange = e => {
        e.preventDefault();
        stateObj.setFile(e.target.files[0]);
        stateObj.setFileName(e.target.files[0].name);
    };

    //Handles the posting of projects to the community page
    const postHandler = async e => {
        e.preventDefault();
        const formData = new FormData();
        stateObj.setPostFile(stateObj.file);
        formData.append('file', stateObj.file);
        formData.append('post_title', stateObj.postName);
        formData.append('post_category', stateObj.postCategory);
        closePost();
        await axios.post('/makePost', formData, { headers: { 'Content-Type': 'multipart/form-data' } });
    };
}
```

```

//Sends the search terms to the backend and gets the response
const submitHandler = (event) => {
    event.preventDefault();
    alert("You are searching for " + stateObj.searchTitle + " " +
stateObj.searchCategory);

    axios.get('/searchPost', { params: { post_title: stateObj.searchTitle,
post_category: stateObj.searchCategory } }).then(response => {
        if(response.data.length > 0){
            console.log(response.data.length);
            let _html = "";
            _html += `<h1>Search Results</h1>`;
            _html += `<div class="RecentPostsFormat">
                
                <h2 class="PostTitle">${response.data[0].post_title}</h2>
                <h3 class="PostCategory">${response.data[0].post_category}</h3>
                <h4 class="PostVotes">${response.data[0].post_votes}</h4>
                <p class="PostDescription">Post Description</p>
                <p hidden class="PostID">${response.data[0].comm_pg_id}</p>
                <button id="PlusButton" type="button">+</button>
                <button id="MinusButton" type="button">-</button>
            </div>`;
            document.getElementById("search-post").innerHTML = _html;
            document.getElementById("recent-posts").innerHTML = '';
        }

        document.getElementById("PlusButton").addEventListener("click", voteplus);
        document.getElementById("MinusButton").addEventListener("click", voteminus);
    }
}
else{
    let _html = "";
    _html += `<h1>Search Results</h1>`;
    _html += `<div class="NoResult">
        <p>Sorry, we couldn't find anything</p>
    </div>`;
    document.getElementById("search-post").innerHTML = _html;
    document.getElementById("recent-posts").innerHTML = '';
}

```

```

        }).catch(function (error) {
            stateObj.setResultTitle("Not Found");
            stateObj.setResultCategory("Not Found");
            stateObj.setResultFile("");
        });
    };

const voteplus = () => {
    axios.post('/voteplus');
    console.log("vote test");
};

const voteminus = () => {
    axios.post('/voteminus');
    console.log("vote test");
};

//simply opens up the post box
const openPost = () => {
    document.getElementById("postform").style.display = "block";
}

//Closes the post box
const closePost = () => {
    document.getElementById("postform").style.display = "none";
}

//Upon entering the page the most recent 5 posts are displayed
useEffect(() => {
    axios.get('/recent5').then(response => {
        console.log(response.data[0]);
        let _html = "";
        _html += `<h1>Recent Posts</h1>`;
        response.data.forEach(post => {_html += `<div class="RecentPostsFormat">
            
            <h2 class="PostTitle">${post.post_title}</h2>
            <h3 class="PostCategory">${post.post_category}</h3>
            <p class="PostDescription">Post DescriptionPost DescriptionPost
DescriptionPost DescriptionPost DescriptionPost DescriptionPost
DescriptionPost DescriptionPost DescriptionPost DescriptionPost
DescriptionPost DescriptionPost Description</p>
            <button id="PlusButton" type="button">+</button>
            <button id="MinusButton" type="button">-</button>
        </div>`);
    });
});

```

```

        </div>`;})
    document.getElementById("recent-posts").innerHTML = _html;
    document.getElementById("search-post").innerHTML = '';
    document.getElementById("PlusButton").addEventListener("click",
voteplus);
    document.getElementById("MinusButton").addEventListener("click",
voteminus);
}).catch(function (error) {
    console.log('fail')
});
};

return (
<>
<Router>
<div className="App">
<Navbar />
<header className="App-header">
<div class="description">
<h2 align='center' top='30%'> Community Page </h2>
<p style={{fontSize: 22, marginLeft: 40, marginRight: 40 }}>
align='center'>See how other people are using DoReMeet to bring their artistic
dreams to life, or post some of your own amazing collaborations with fellow
DoReMeet users
<br />
<br />
Check out our <a href="#highlight-section"> Highlights section </a>
to see the most liked posts of the month.</p>
</div>
<div className="PageContainer">
<div class="SearchContainer">
<form class="search" onSubmit={submitHandler}>
<input class="searchBar" onChange={e =>
stateObj.setSearchTitle(e.target.value)} type="text" placeholder="Search" />
<select class="searchButtons" onChange={e => {
stateObj.setSearchCategory(e.target.value); }}>
<option value={"Music"}>Music</option>
<option value={"Dance"}>Dance</option>
<option value={"Art"}>Art</option>
<option value={"Cinema"}>Cinema</option>

```

```

        <option value={"Photography"}>Photography</option>
        {/* <option value={"M"}>Music</option>
        <option value={"D"}>Dance</option>
        <option value={"A"}>Art</option>
        <option value={"C"}>Cinema</option>
        <option value={"P"}>Photography</option> */}
    </select>
    <input class="searchButtons" type='submit' />
</form>
</div>
<div className="PostFormFormat">
    <label htmlFor="postbutton">Share your work with us here!</label>
    <button className="post" id="postbutton"
onClick={openPost}>Post</button>
</div>
<div class="post-popup" id="postform">
    <form class="post-container" onSubmit={postHandler}>
        <h2> Post your creative collaborations! </h2>
        <input type="text" onChange={e =>
stateObj.setPostName(e.target.value)} placeholder="Name" required />
        <div>
            <select onChange={e => {
stateObj.setPostCategory(e.target.value); }}>
                <option>Music</option>
                <option>Dance</option>
                <option>Art</option>
                <option>Cinema</option>
                <option>Photography</option>
            </select>
        </div>
        <label htmlFor="post-file"> Select file: </label>
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
        <input type='file' className='custom-file-input' id='customFile'
onChange={onChange} />
        <div>
            <input type='submit' />
        </div>
        <div>
            <button type= "button" onClick={closePost}>Close</button>
        </div>
    </form>
</div>

```

```
        </form>
    </div>
    <div class="RecentPosts" id="recent-posts">
        </div>
        <div class="SearchPost" id="search-post">
            </div>
            <p id="vote"></p>
            <div id='highlight-section' >
                <Highlights />
            </div>
        </div>
        </header>
    </div>
    <Footer />
</Router>
</>
);
}

export default CommunityPage;
```

## Code Review:



Nimiksha Mahajan <nimiksha98@gmail.com>

### CSC648 Team 02 Code Review

Nimiksha Mahajan <nimiksha98@gmail.com>  
To: Luke Chang-Knezevich <lchangknezevich@mail.sfsu.edu>

Mon, Dec 7, 2020 at 11:53 AM

Hi Luke,

Thank you for reaching out. Please find my review for the Community Page code.  
If you have any questions, please let me know.

```
<--- CODE REVIEW
This is good. It is helpful to have a header here
--->
/*
**CSC 648 Team 02 DoReMeet
**File: CommunityPage.js
**Desc: Contains all the code needed for the community page. Calls on the backend
through axios and receives json responses that it then displays accordingly on the
page.
*/
import React, {useEffect, useState} from 'react';
import '../App.css';
import './CommunityPage.css';
import axios from 'axios';
import Navbar from '../components/Navbar/Navbar';
import Highlights from '../components/Highlights/Highlights';
import Footer from '../components/Footer/Footer';
import { BrowserRouter as Router } from 'react-router-dom';

const CommunityPage = (stateObj) => {

<--- CODE REVIEW
Should add comments. Just onChange doesn't really give a good description of what
this is doing. Or consider more a descriptive name
--->
  const onChange = e => {
    e.preventDefault();
    stateObj.setFile(e.target.files[0]);
    stateObj.setFileName(e.target.files[0].name);
  };
}
```

```

<--- CODE REVIEW
Good comment and descriptive name
--->

//Handles the posting of projects to the community page
const postHandler = async e => {
  e.preventDefault();
  const formData = new FormData();
  stateObj.setPostFile(stateObj.file);
  formData.append('file', stateObj.file);
  formData.append('post_title', stateObj.postName);
  formData.append('post_category', stateObj.postCategory);
  closePost();
  await axios.post('/makePost', formData, { headers: { 'Content-Type': 'multipart/form-data' } });
};

<--- CODE REVIEW
It's helpful to have the alert for what the user is searching for, but considering from a UI/UX standpoint it might not be the most efficient thing to do. Alerts for every single search can be tedious/unnecessary.
But good work with comments and naming.
--->

//Sends the search terms to the backend and gets the response
const submitHandler = (event) => {
  event.preventDefault();
  alert("You are searching for " + stateObj.searchTitle + " " +
stateObj.searchCategory);

<--- CODE REVIEW
I think you've done a good job with naming everything, but I wonder if the code is the most efficient. Is using html components the most effective in terms of both frontend and backend? I feel you can use more modular code here, that will also improve the time to display search results
--->

  axios.get('/searchPost', { params: { post_title: stateObj.searchTitle,
post_category: stateObj.searchCategory } }).then(response => {
    if(response.data.length > 0){
      console.log(response.data.length);
      let _html = "";
      _html += `<h1>Search Results</h1>`;
      _html += `<div class="RecentPostsFormat">
        
      </div>`;
      document.getElementById('recentPosts').innerHTML = _html;
    }
  });
}

```

```

        <h2 class="PostTitle">${response.data[0].post_title}</h2>
        <h3 class="PostCategory">${response.data[0].post_category}</h3>
        <h4 class="PostVotes">${response.data[0].post_votes}</h4>
        <p class="PostDescription">Post Description</p>
        <p hidden class="PostID">${response.data[0].comm_pg_id}</p>
        <button id="PlusButton" type="button">+</button>
        <button id="MinusButton" type="button">-</button>
        </div>`;

    document.getElementById("search-post").innerHTML = _html;
    document.getElementById("recent-posts").innerHTML = '';
}

<--- CODE REVIEW
I'd suggest renaming the buttons, to be more indicative of their purpose - to Vote
up or Vote down
--->

    document.getElementById("PlusButton").addEventListener("click", voteplus);
    document.getElementById("MinusButton").addEventListener("click",
voteminus);
}

<--- CODE REVIEW
I think you should make the error message more descriptive, and display why the
search failed
--->

else{
    let _html = "";
    _html += `<h1>Search Results</h1>`;
    _html += `<div class="NoResult">
        <p>Sorry, we couldn't find anything</p>
    </div>`;
    document.getElementById("search-post").innerHTML = _html;
    document.getElementById("recent-posts").innerHTML = '';
}

}).catch(function (error) {
    stateObj.setResultTitle("Not Found");
    stateObj.setResultCategory("Not Found");
    stateObj.setResultFile("");
});

};

<--- CODE REVIEW
Naming - use camel case for better readability and consistency
--->

const voteplus = () => {

```

```

    axios.post('/voteplus');
    console.log("vote test");
};

const voteminus = () => {
    axios.post('/voteminus');
    console.log("vote test");
};

----- CODE REVIEW
The comments are good. Really helps the reader see what function is doing what.
Also helps with debugging and testing
--->
//Simply opens up the post box
const openPost = () => {
    document.getElementById("postform").style.display = "block";
}

//Closes the post box
const closePost = () => {
    document.getElementById("postform").style.display = "none";
}

//Upon entering the page the most recent 5 posts are displayed
useEffect(() => {
    axios.get('/recent5').then(response => {
        console.log(response.data[0]);
        let _html = "";
        _html += `<h1>Recent Posts</h1>`;
        response.data.forEach(post => {_html += `<div class="RecentPostsFormat">
            
            <h2 class="PostTitle">${post.post_title}</h2>
            <h3 class="PostCategory">${post.post_category}</h3>
`});
    })
}

----- CODE REVIEW
What is the purpose of all the Post Descriptions? I suggest adding comments and
making the code more modular instead of arbitrarily using 'Post Description'
--->
<p class="PostDescription">Post DescriptionPost DescriptionPost
DescriptionPost DescriptionPost DescriptionPost DescriptionPost DescriptionPost
DescriptionPost Description</p>
<button id="PlusButton" type="button">+</button>
<button id="MinusButton" type="button">-</button>
</div>`);}

document.getElementById("recent-posts").innerHTML = _html;
document.getElementById("search-post").innerHTML = '';
document.getElementById("PlusButton").addEventListener("click", voteplus);

```

```

        document.getElementById("MinusButton").addEventListener("click",
voteminus);
    }).catch(function (error) {
    console.log('fail')
});
});

}

<--- CODE REVIEW
The code is good, but I think it should be reformatted to be more readable.
--->
return (
<>
<Router>
<div className="App">
<Navbar />
<header className="App-header">
<div class="description">
<h2 align='center' top='30%'> Community Page </h2>
<p style={{fontSize: 22, marginLeft: 40, marginRight: 40 }}>
align='center'>See how other people are using DoReMeet to bring their artistic
dreams to life, or post some of your own amazing collaborations with fellow
DoReMeet users
</p>
</div>
</header>
<div className="App-content">
<h3>Community Page</h3>
<p>This page displays a grid of posts from other users. You can like and comment on them, and even collaborate with others by adding your own posts. Have fun!</p>
<hr/>
<div>
<h4>Recent Posts</h4>
<ul>
<li>Post 1</li>
<li>Post 2</li>
<li>Post 3</li>
<li>Post 4</li>
<li>Post 5</li>
</ul>
</div>
<hr/>
<div>
<h4>Search</h4>
<form>
<input type="text" placeholder="Search" />
<select>
<option value="Music">Music</option>
<option value="Art">Art</option>
<option value="Food">Food</option>
<option value="Tech">Tech</option>
</select>
<button type="submit">Search</button>
</form>
</div>
<hr/>
<div>
<h4>Collaborations</h4>
<ul>
<li>Collaboration 1</li>
<li>Collaboration 2</li>
<li>Collaboration 3</li>
<li>Collaboration 4</li>
<li>Collaboration 5</li>
</ul>
</div>
<hr/>
<div>
<h4>About</h4>
<p>DoReMeet is a platform for artists and creators to share their work and collaborate with others. We believe that art is a powerful way to connect people and inspire creativity. Join our community today and start creating together!</p>
</div>
</div>
</Router>
</div>
);
}

<--- CODE REVIEW
Should avoid using <br>. Instead set padding or margins.
--->
<br />
<br />
<--- CODE REVIEW
You should use a consistent naming format, and not use Camel case and '-'
alternatively. Since camel case is more prominent in the code, I suggest ensuring
that all variables/ids/functions are named using Camel case style.
--->
Check out our <a href="#highlight-section"> Highlights section </a>
to see the most liked posts of the month.</p>
</div>
<div className="PageContainer">
<div className="SearchContainer">
<form onSubmit={submitHandler}>
<input className="searchBar" onChange={e =>
stateObj.setSearchTitle(e.target.value)} type="text" placeholder="Search" />
<select className="searchButtons" onChange={e => {
stateObj.setSearchCategory(e.target.value); }}>
<option value="Music">Music</option>
<option value="Art">Art</option>
<option value="Food">Food</option>
<option value="Tech">Tech</option>
</select>
</form>
</div>
</div>
);
}

```

```
<option value="Dance">Dance</option>
<option value="Art">Art</option>
<option value="Cinema">Cinema</option>
<option value="Photography ">Photography</option>
/* <option value="M">Music</option>
<option value="D">Dance</option>
<option value="A">Art</option>
<option value="C">Cinema</option>
<option value="P">Photography</option> */
</select>
<input class="searchButtons" type='submit' />
</form>
</div>
<div className="PostFormFormat">
<!-- CODE REVIEW
Naming - use camel case for better readability and consistency
-->
<label htmlFor="postbutton">Share your work with us here!</label>
<button className="post" id="postbutton" onClick={openPost}>Post</button>
</div>
<div className="post-popup" id="postform">
<form className="post-container" onSubmit={postHandler}>
<h2> Post your creative collaborations! </h2>
<input type="text" onChange={e => stateObj.setPostName(e.target.value)} placeholder="Name" required />
<div>
<select onChange={e => { stateObj.setPostCategory(e.target.value); }}>
<option >Music</option>
<option >Dance</option>
<option >Art</option>
<option >Cinema</option>
<option >Photography</option>
</select>
</div>
<label htmlFor="post-file"> Select file: </label>
<!-- CODE REVIEW
This is not ideal. You should use styling to add the space instead of arbitrarily
using spaces.
Also code should be formatted in a hierarchy structure to be more readable.
-->
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
```

```
        <input type='file' className='custom-file-input' id='customFile'
onChange={onChange} />
        <div>
            <input type='submit' />
        </div>
        <div>
            <button type= "button" onClick={closePost}>Close</button>
        </div>
    </form>
</div>
<!-- CODE REVIEW
Should clean up this code, and condense it to avoid multiple divs. can us <div ....
/> instead.
-->
        <div class="RecentPosts" id="recent-posts">
        </div>
        <div class="SearchPost" id="search-post">
        </div>
        <p id="vote"></p>
        <div id='highlight-section' >
            <Highlights />
        </div>
        </div>
    </header>
</div>
<Footer />
</Router>
</>
);
}

export default CommunityPage;
```

## 5. Self-Check on best Security Practice

### 1. List major assets you are protecting

- Password
- Users' personal information (email, phone #, DOB)
- Media files (images) on server
- Users Location
- Database

### 2. Password encryption in database

```
app.post('/signup', (req, res) => {
  // user table

  data being sent to SQL

  checks the email and username in the
  database, if doesn't exists insert data into
  table

  // checks for email
  pool.query(todb_check_email, [email], (err, results) => {
    if(results && results.length == 0){
      // checks for username
      pool.query(todb_check_username, [username], (err, results) => {
        if(results && results.length == 0){
          var todb_user = 'INSERT INTO user (first_name, last_name, gender, date_of_birth, email, phone_number, art_category, skill_lvl)
          pool.query(todb_user, [first_name, last_name, gender, date_of_birth, email, phone_number, art_category, skill_lvl]
          user_id  first_name  last_name  gender  date_of_birth  email  phone_number  art_category  skill_lvl
          1  Test  one  M  1991-10-10 00:00:00  tone11@gmail.com  562743  A  I
          2  Test  two  M  1992-10-10 00:00:00  ttwo12@gmail.com  562743  A  I
          3  Test  three  M  1993-10-10 00:00:00  thre13@gmail.com  562743  C  I
          4  Test  four  M  1994-10-10 00:00:00  tfour14@gmail.com  562743  P  I
          5  Test  five  M  1995-10-10 00:00:00  tfive15@gmail.com  562743  D  I
          NULL  NULL  NULL  NULL  NULL  NULL  NULL  NULL  NULL
          hash the password of the user
          and save it into database

          console.log(hash);
          var todb_account = 'INSERT INTO account (username, password, acc_created, user) VALUES (?, ?, now(), ?)';
          pool.query(todb_account, [username, hash, user_id], (err, result) => {
            if(err) throw err;
            else{
              console.log(result);
            }
          }
        )
      
```

account_id	username	password	acc_created	activate	user
1	tone11	\$2b\$10\$6f6c359a00d4RdVvayM5cyOC3XqfD9AmPzBRHy1ddpCney20/ZK	2020-12-07 16:49:43	0	1
2	tone12	\$2b\$10\$8V.1vzCymGpnwqgHD9ICjSGDNyLjMymOlkfUjykm0ySV1dkh2	2020-12-07 16:55:32	0	2
3	tone13	\$2b\$10\$8c2LTlvrHvHq7QCeod9wed88Jf6l1QZSM1P1GJubmMsB8u	2020-12-07 16:56:16	0	3
4	tone14	\$2b\$10\$9P4hdC2sul0TfV73nQnhup.AeLgqfz2Z736fSewr3DUfTr9ED	2020-12-07 16:57:02	0	4
5	tfive15	\$2b\$10\$saTFcjOs8VsR/po1aRF9ehMogtSBVC6nQnahS8W./omf.X.sUO	2020-12-07 16:57:44	0	5
NULL	NULL	NULL	NULL	NULL	NULL

## Login

The diagram illustrates the login process. On the left, a screenshot of a login form is shown with fields for Username (tfinal00) and Password, and a Log In button. A red arrow labeled "taking action on click" points from the Log In button to the corresponding code in the middle. The middle section shows the React component code for the login form. A green box highlights the axios post request to '/login'. A blue box highlights the response handling logic. A red box highlights the onClick event handler for the Log In button. On the right, a screenshot of a database table named 'account' is shown with columns account\_id, username, and password. The table has six rows, with the last row showing account\_id 6, username 'tfinal00', and password '123'.

```

    login = e => {
      e.preventDefault();
      axios.post('/login', this.state).then(response => {
        if(response.data){
          this.props.history.push('/Community');
        }else{
          alert("Your username or password is incorrect");
        }
      });
    }

    <form onSubmit={this.handleSubmit}>
      <h1 className="heading-login">Log In</h1>
      <div className="input">
        <label htmlFor="username" className="sub-heading"> <b> Username </b></label>
        <input type="username" name="username" value={this.state.username} placeholder="Enter Username" required>
      </div>
      <div className="input">
        <label htmlFor="password" className="sub-heading"> <b> Password </b></label>
        <input type="password" name="password" value={this.state.password} placeholder="Enter Password" required>
      </div>
      &nbsp;
      <div className="input">
        <button className='btn' buttonStyle='btn--primary' buttonSize='btn--large' onClick={this.login} > Log In </button>
      </div>
      &nbsp;
    </form>
  
```

The diagram shows the backend login process. A red box highlights the axios post request to '/login' in the frontend code. A blue box highlights the database query in the backend code. A blue box also highlights the password '123' in the database table. A red box highlights the error message 'incorrect creds' in the frontend code. The database table on the right shows the same data as the previous diagram.

```

    login = e => {
      e.preventDefault();
      axios.post('/login', this.state).then(response => {
        if(response.data){
          this.props.history.push('/Community');
        }else{
          alert("Your username or password is incorrect");
        }
      });
    }

    app.post('/login', (req, res) => [
      console.log("_____start_____"),
      console.log(req.body),
      var todB = "SELECT * FROM account WHERE (username = '" + req.query.username + "')";
      pool.query(tdB, (error, result) => {
        if (result.length == 1) {
          // console.log(res.redirect('/'));
          req.session.username = result[0].username;
          req.session.userId = result[0].user;
          console.log(req.session);
          res.send(req.session);
          console.log("_____end1_____")
        } else {
          console.log("incorrect creds");
          console.log(error);
          console.log("_____0_____");
          console.log(req.session);
          res.send(null);
          console.log("_____end2_____")
        }
      })
    ])
  
```

account_id	username	password
1	tfinal00	123
2	tony12	42b3108f-1a9c-499c-891c-5d019a05455e
3	tony13	42b3108f-1a9c-499c-891c-5d019a05455e
4	tony14	42b3108f-1a9c-499c-891c-5d019a05455e
5	tony15	42b3108f-1a9c-499c-891c-5d019a05455e
6	tfinal00	123

## Sign Up

The screenshot shows a 'Sign Up' form on a website. The form fields are:

- Phone Number
- Password
- Confirm Password
- A checkbox for 'Remember Me'

The 'Sign Up' button is highlighted with a red box.

Annotations and code snippets:

- receiving data from the user:** Points to the input fields for Phone Number, Password, and Confirm Password.
- Sending data to backend:** Points to the 'Sign Up' button.
- receiving data to backend:** Points to the axios POST request in the code.
- doing action on click:** Points to the onClick event handler in the code.

```
<div className="input">
  <label htmlFor="phone" className="sub-heading"> <b> Phone Number </b> </label>
  <input type="phone" name="phone" value={this.state.phone} placeholder="Enter Phone Number" required onChange={this.handleChange}>
</div>

<div className="input">
  <label htmlFor="password" className="sub-heading"> <b> Password </b> </label>
  <input type="password" name="password" value={this.state.password} placeholder="Enter Password" required onChange={this.handleChange}>
</div>

<div className="input">
  <label htmlFor="confirmPassword" className="sub-heading"> <b> Confirm Password </b> </label>
  <input type="password" name="confirmPassword" value={this.state.confirmPassword} placeholder="Confirm Password" required onChange={this.handleChange}>
</div>

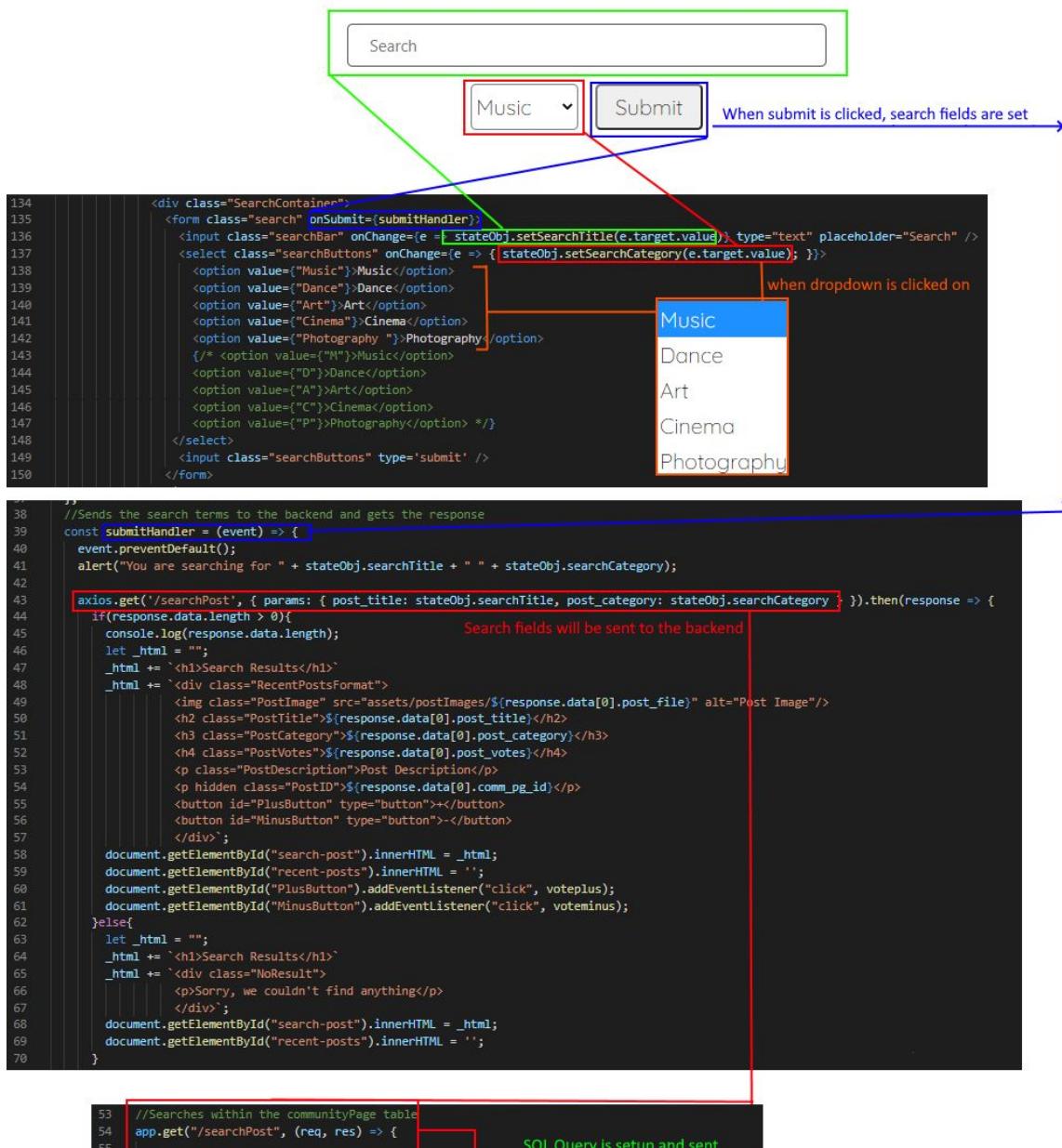
<div className="input">
  <button className="btn btn--primary" buttonSize="btn--large" onClick={this.register}> Sign Up </button>
</div>
```

```
handleSubmit = (e) => [
  e.preventDefault();

  console.log(this.state);
  axios.post('/signup', this.state)
    .then(response => {
      console.log(response)
      console.log(response.data);
      // window.location = "/Login"
    })
    .catch(error => {
      console.log(error)
    });
]
```

### 3. Input Data Validation

The screenshot shows the DoReMeet community page. At the top, there's a navigation bar with links for Community, Match, Chat, and Profile. On the far right, there's a gear icon for settings and a blue button labeled "Log Out". Below the navigation, the main content area has a light green header with the title "Community Page". The header includes a placeholder text: "See how other people are using DoReMeet to bring their artistic dreams to life, or post some of your own amazing collaborations with fellow DoReMeet users". Below this, another section says "Check out our [Highlights](#) section to see the most liked posts of the month."



```

56 var post_title = req.query.post_title;
57 var post_category = req.query.post_category;
58
59 var todb = 'SELECT * FROM communityPage WHERE post_title = ? AND post_category = ?';
60 pool.query(todb,[post_title, post_category] ,(err, result) => {
61   if (err || result == ''){
62     console.log(err);
63     console.log("searching fail");
64     res.send(err)
65   }else{
66     //console.log(result);
67     res.send(result);
68     console.log("searching pass");
69   }
70 })
71 });
72 });

    when there are no results
    comm_pg_id first_name last_name post_title post_category post_file post_votes
    NULL      NULL      NULL      NULL      NULL      NULL      NULL

    OR

    when result exists
    comm_pg_id first_name last_name post_title post_category post_file post_votes
    6          NULL      NULL      Curve      Art      2020-12-07_13-57-36.png 1

```

```

38 //Sends the search terms to the backend and gets the response
39 const submitHandler = (event) => {
40   event.preventDefault();
41   alert("You are searching for " + stateObj.searchTitle + " " + stateObj.searchCategory);
42
43 axios.get('/searchPost', { params: { post_title: stateObj.searchTitle, post_category: stateObj.searchCategory } }).then(response => {
44   if(response.data.length > 0){
45     console.log(response.data.length);
46     let _html = "";
47     _html += `<h1>Search Results</h1>
48     <div class="RecentPostsFormat">
49       
50       <h2 class="PostTitle">${response.data[0].post_title}</h2>
51       <h3 class="PostCategory">${response.data[0].post_category}</h3>
52       <h4 class="PostVotes">${response.data[0].post_votes}</h4>
53       <p class="PostDescription">Post Description</p>
54       <p hidden class="PostID">${response.data[0].comm_pg_id}</p>
55       <button id="PlusButton" type="button">+</button>
56       <button id="MinusButton" type="button">-</button>
57     </div>`;
58     document.getElementById("search-post").innerHTML = _html;
59     document.getElementById("recent-posts").innerHTML = '';
60     document.getElementById("PlusButton").addEventListener("click", voteplus);
61     document.getElementById("MinusButton").addEventListener("click", voteminus);
62   }else{
63     let _html = "";
64     _html += `<h1>Search Results</h1>
65     <div class="NoResult">
66       <p>Sorry, we couldn't find anything</p>
67     </div>`;
68     document.getElementById("search-post").innerHTML = _html;
69     document.getElementById("recent-posts").innerHTML = '';
70   }
}

```

## Results

Community Page

See how other people are using DoReMeet to bring their artistic dreams to life, or post some of your own amazing collaborations with fellow DoReMeet users.

Check out our [Highlights section](#) to see the most liked posts of the month.

Share your work with us here!

**Search Results**

Sorry, we couldn't find anything

Community Page

See how other people are using DoReMeet to bring their artistic dreams to life, or post some of your own amazing collaborations with fellow DoReMeet users.

Check out our [Highlights section](#) to see the most liked posts of the month.

Share your work with us here!

**Search Results**

  
Curve  
Art  
1 Post Description  
()

## **6. Self Check: Adherence to Non Functional Specs**

**\*\* Statements in red** -> These Non-Functional requirements are not implemented given time, resource, and knowledge limitations. Some of these (especially those pertaining to maintenance) cannot be implemented given that we will only have test users for M5, and there is not enough time to do actual maintenance. These requirements were defined for a product that is fully implemented and meets industry standards.

**\*\* Statements in Purple** -> These Non-Functional requirements have not been implemented because they are related to Priority 2 and Priority 3 Functional requirements. Given those Priority 2 and 3 functionalities have not been implemented, the corresponding Non-Functional requirements have also not been fulfilled.

### **Security:**

1. Accounts shall optionally have 2 factor identification enabled.
2. User's password shall be encrypted in the database. **DONE**
3. The site shall verify the users' credentials before successfully logging them in. **DONE**
4. The website will have a SSL certificate (<https://>).
5. The website shall have OAuth 2.0 implemented.
6. Accounts shall only have one password. There shall be no master key. **DONE**
7. The website shall warn users when clicking on links that take them to another website. **DONE**
8. Premium account users' payment information shall be encrypted in the database.

### **Storage:**

9. Each account shall be able to store up to 80mb of associated content. **ON TRACK**
10. Each premium account shall be able to store up to 120mb of associated content.
11. A single media shall be less than or equal to 10mb. **ON TRACK**
12. User's data shall be saved into the MYSQL database. **DONE**
13. Users data should be removed from the database when the user deletes their account. **ON TRACK**
14. User accounts that have not been logged in for 1 year shall be disabled.

### **Availability:**

15. The site shall be up and running at all times excluding system failures and scheduled maintenance. **DONE**
16. The site shall notify of all scheduled maintenance, 1 week in advance.
17. The users shall be notified via email about all scheduled maintenance, 1 week in advance.
18. Any/all scheduled maintenance shall be scheduled past 10pm PST on Mondays.

### **Functionality:**

19. The website shall be hosted and deployed on the AWS server, EC2 instance. **DONE**
20. Account details on the database shall be modifiable. **DONE**

21. User details on the database shall only be modified by the user themselves. **DONE**
22. Only flagged user details, that have been reviewed, shall be modified by the admin.
23. The user shall log in using either email or username. **DONE**
24. The user shall be required to enter both password and username to log in. **DONE**
25. The user shall need a valid email to create an account. **DONE**
26. Users should be minimum 18 years of age to create an account. **DONE**
27. The users' password shall be censored on the sign in page. **DONE**
28. The users' password shall be censored on the sign up page. **DONE**
29. The website shall be responsive to user input. **DONE**
30. A premium account shall accept Visa, MasterCard and American Express.
31. A premium account shall accept GooglePay and ApplePay as a valid payment method.

#### **Fault Tolerance:**

32. Upon a system crash a global error page shall be displayed for all users trying to access the site. **DONE**
33. There shall be a backup for the website before any update is pushed. **ON TRACK**
34. There shall be a backup for the database every week on Tuesday past 10:00pm PST.
35. Error messages shall be relevant to the user (ex. "Site is down at the moment" or "Your account has been suspended" etc.)

#### **Scalability:**

36. The number of accounts that the system shall be able to handle should be scalable based on consumer demand/traffic. **ON TRACK**

#### **Privacy:**

37. Developers and staff shall not have direct access to sensitive user information. **DONE**
38. Anyone without a registered account shall not be able to view profiles of registered users. **DONE**
39. The site shall accept cookies/cache only if the user explicitly allows it. **ON TRACK**
40. Users shall not be able to see other users' password. **DONE**
41. Users shall not be able to see other users' email. **DONE**
42. Users shall not be able to see other users' phone number (if provided). **DONE**
43. Users data shall not be shared with any third-party applications/sites/organizations. **DONE**
44. Users data shall not be sold to any third-party applications/sites/organizations. **DONE**
45. The site shall not gather any unnecessary data from users. **DONE**
46. Users' geolocation shall only be known by the website if they allow it. **DONE**
47. Premium account users' payment information shall only be visible to that particular user.

#### **Compatibility:**

48. The site shall be compatible with Safari (version 10.0 - 14.0), Chrome (version 80 - 85.0) and Mozilla Firefox (version 78 - 81) and Edge (version 85). **DONE**
49. The site shall be compatible with Mac OS (10.12.6 - 10.15), Windows (7 and 10) and Linux. **DONE**

50. The site shall be compatible with mobile browsers on both iOS and Android. **DONE**

**Look and Feel:**

- 51. The site shall maintain a core color scheme. **DONE**
- 52. The website shall have a functional navigation bar on every page. **DONE**
- 53. The website shall have a user settings page. **DONE**
- 54. The website's logo shall direct the user to the main home page. **DONE**
- 55. The website shall be mobile-friendly. **DONE**

**Ease of Use:**

- 56. The site shall maintain less formal vocabulary and diction. **DONE**
- 57. A preview of what the site has to offer will be displayed on the site landing page. **DONE**
- 58. Website UI shall be simple and efficient. **DONE**
- 59. Website UI shall be easy to use. **DONE**
- 60. Website UI shall be intuitive. **DONE**
- 61. Users shall be able to find what they need on the website in 5 seconds. **ON TRACK**

**Coding Standard:**

- 62. The whole site shall use the same coding standards related to code indentation. **DONE**
- 63. All functions shall be modularized. **DONE**

**Marketing Requirements:**

- 64. The website shall not make any political marketing campaigns. **DONE**
- 65. The website shall have a logo on the navigation bar. **DONE**
- 66. The website shall have a logo on the "tab" for site identity. **DONE**

**Legal Requirements:**

- 67. The site shall explicitly not endorse any activity that is illegal, according to the U.S law. **DONE**
- 68. Access to the website's policies shall be available at the bottom of every webpage. **DONE**
- 69. The site shall protect all user data under the privacy laws in the U.S. **DONE**
- 70. The site shall be transparent about the use of user data. **DONE**
- 71. An unregistered user shall have to agree to the website's terms & conditions to successfully create an account. **DONE**

**Performance:**

- 72. Match results shall be filtered in at least 1 second. **ON TRACK**
- 73. The website's response time shall be at most 1 second. **ON TRACK**
- 74. 2 factor identification code shall be sent to the user's phone in at most 5 seconds.

**Expected load:**

- 75. The system shall be able to handle up to 1000 users at once.

## **7. List of Contributions**

<b>Member</b>	<b>Role</b>	<b>Contributions</b>
Nimiksha	Team Lead & Documentation Lead	Product Summary Usability Test (shortlisting requirements and write up) QA Test plan (write up and ran all the QA Test) Code Review (gave review) Self-Check Adherence to Non-Functional Requirements Code Cleanup to meet Coding standards established M2, M3 Version 2 Documentation M4 Documentation Brief Iterative QA test with the team to incorporate CTO's feedback
Meet	Backend and Database Lead	QA Testing (for allow location feature) Fixed Log in password encryption in the database Self-Check on best Security Practices (Password Encryption) Self-Check Adherence to Non-Functional Requirements Configure the LocationIQ API to grab user's location Code the expand and reduce search radius feature for potential matches Documentation Review Brief Iterative QA test with the team to incorporate CTO's feedback
Luke	Frontend Lead	QA Testing (for sign up - Date of Birth Validation) Self-Check Adherence to Non-Functional Requirements Coded the Chat page to have a more friendly and intuitive UI Updated the Profile page to make the user information editable and add media files Built the global 404 Error page for the website Host updated version of the product on EC2 instance Documentation Review Brief Iterative QA test with the team to incorporate CTO's feedback
Mike	GitHub Master & Backend Engineer	Self-Check on best Security Practices (Data Validation) Self-Check Adherence to Non-Functional Requirements Configured the matching algorithm and logic Implemented the Match page to show potential matches to the user Fixed pathing errors and errors with uploading images to Community Page Updated Settings and Preferences backend to accept user changes Documentation Review Brief Iterative QA test with the team to incorporate CTO's feedback
Jungsun	Database & Backend Engineer	Self-Check Adherence to Non-Functional Requirements Worked on the Update password feature (user can safely update current password from Settings page) Password Recovery implementation Set up redirects for Log out Documentation Review Brief Iterative QA test with the team to incorporate CTO's feedback

Vincent	Server Admin & Frontend Engineer	<p>QA Testing - (for leaving website alert message)</p> <p>Self-Check Adherence to Non-Functional Requirements</p> <p>Worked on Match Page frontend (waiting for backend)</p> <p>Implemented Community page voting feature</p> <p>Upload Social media links on users' profile</p> <p>Documentation Review</p> <p>Brief Iterative QA test with the team to incorporate CTO's feedback</p>
---------	----------------------------------	--

## 4. Final Product Screenshots

### Landing Page

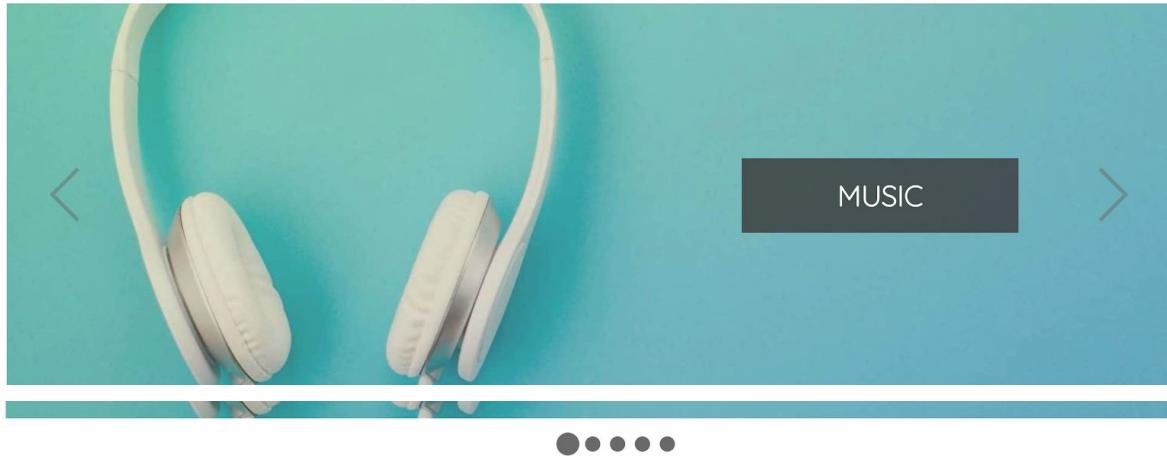
The screenshot shows a web browser window with the URL "Not Secure | ec2-13-52-247-220.us-west-1.compute.amazonaws.com". The page has a light blue header bar with the "DoReMeet" logo, navigation links for "Company", "Community", and "Pricing", and a "Log In" button. Below the header is a large, bold title "CONNECT & CREATE". A subtext below the title reads "Meet artists near you, and build what your heart desires". At the bottom are two buttons: "Log In" (white background) and "Sign Up" (blue background). The main content area is a light green color.

The screenshot shows a white rectangular box containing a teal-colored graphic of a crumpled yellow paper lightbulb with radiating lines. To the right of the graphic is a section titled "Why Do Re Meet?". The text describes DoReMeet as a platform for connecting and learning with people, ranging from experts to novices, without pressure. It mentions the ability to find matches based on choice, whether for fun collaboration or serious work. The text also notes that users can switch preferences and access a Community page for sharing collaborations and building an art community.

Why Do Re Meet?

DoReMeet offers a no pressure setting to connect and learn with/from people. The users range from experts to novices and users have the opportunity to find these matches based on their choice, for either a fun collaboration or for serious, paid work. We do not limit the user base to any category, and users are free to switch their preferences any time. In addition DoReMeet also offers a Community page to share collaborations, and give a chance to build an art community.

## Our 5 Popular Art Categories



## Check out our Highlights for this Month

The image shows three cards arranged horizontally, each featuring a small thumbnail icon of a person in a specific activity (Dance, Art, or Music). Below each thumbnail is a blue rectangular tag with the category name. The cards are set against a light gray background.

- Dance:** Amateur Salsa Dance at Local Competition
- Art:** Oil Painting
- Music:** Song inspired by everyday noises

## Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)

DaReMeet © 2020



## Login Page



Company    Community    Pricing

Log In

Log In

Username

Enter Username

Password

Enter Password

Log In

Remember Me

[Forgot Password](#)

Don't have an account yet?

[Sign Up!](#)

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

FAQs

Guidelines

Contact Us

Terms & Conditions

Privacy Policy



DoReMeet © 2020

## Sign Up Page



Company   Community   Pricing

Log In

### Sign Up

Please fill the form to create an account with DoReMeet.  
All fields are required.

First Name

Last Name

Email

Username

Gender  Male  Female  Other

Date of Birth

Phone Number

Art Category

Skill Level

Password

Confirm Password

By creating an account you agree to our [Guidelines](#), [Terms & Conditions](#) and [Privacy Policy](#).

Already have an account?

\*Footer not pictured for the Sign Up page (For reference, the footer looks similar to all the other pages)

## Community Page (Unregistered / Not logged in)

 DoReMeet

Company Community Pricing

Log In

### Community Page

See how other people are using DoReMeet to bring their artistic dreams to life, or post some of your own amazing collaborations with fellow DoReMeet users

Check out our [Highlights section](#) to see the most liked posts of the month.

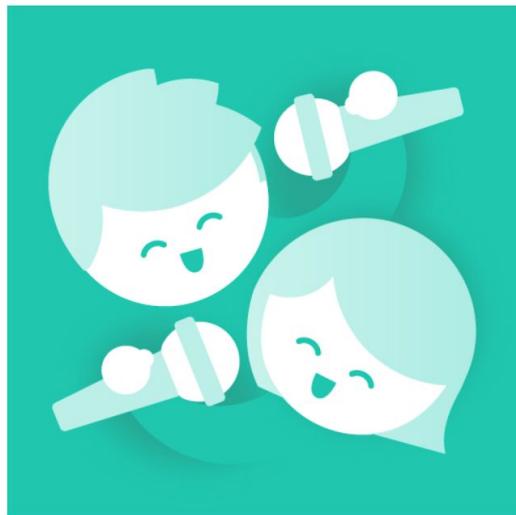
Log in or sign up to post a project to the community page!

Search

Music

Submit

### Recent Posts



#### R&B Cover

1

Music

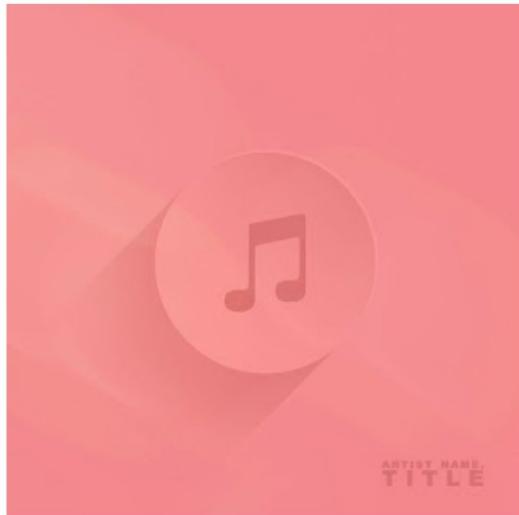
A collection of covers for famous 90s R&B songs



## Revive Instrumental

1  
Music

A collaboration of sounds from everyday life



## Pop Reimagined

1  
Music

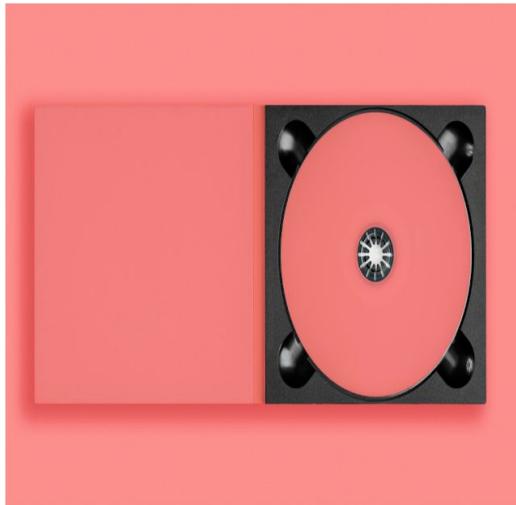
Famous pop songs mixed with indie tunes



## Jazz

1  
Music

classic jazz collaboration with budding musicians



## My Album

1  
Music

A music album for everyone

## Check out our Highlights for this Month



Dance

Amateur Salsa Dance at  
Local Competition



Art

Oil Painting



Music

Song inspired by everyday  
noises

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)



DoReMeet © 2020

## Pricing



Company   Community   Pricing

Log In

# PRICING

DoReMeet is a free service, and is open to all users. However, for a more customized experience, we offer DoReMeet Premium, where for a small monthly cost you get access to unlimited, unrestricted services, so that you can really find that right match for yourself.

With a Premium account you get:

- Unlimited Likes
- No Ads
- Highlighted Profile
- Increased Search Radius
- Rewind on Potential Matches
- Bigger Portfolio Size

Our pricing model is very simple and affordable, and aims to make our Premium services available to everyone.

\$4.99	\$49.99	\$119.99
for 1 Month	for 6 Months	for 1 year

Upgrade your account today and increase your chances of finding the right match!

To upgrade your account, log in and update your Settings

Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy



DoReMeet © 2020

## Match Page

DoReMeet

Community Match Chat Profile

Log Out

Welcome to the matching page!

- Make sure that your profile is all set up
- Check that your preferences are up to date
- Click on the button below to begin matching!

Click Here!

## Page on Clicking the button

DoReMeet

Community Match Chat Profile

Log Out

If you find someone you want to collaborate with, "Connect" with them, or else "Pass" to keep looking for the right match



Bobba Smith

aaaaaaaaaaaaaaaaaaaaaaa

Art: Dance

Tags: Salsa Hip Hop Hip Hop Hip Hop Hip Hop  
Hip Hop

Pass

Connect



YouTube



Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy

DoReMeet © 2020



## Chat Page

The screenshot shows the 'Chat Page' of a platform called 'DoReMeet'. At the top, there's a navigation bar with links for 'Community', 'Match', 'Chat', and 'Profile', along with a 'Log Out' button. On the left, a sidebar titled 'Matches' lists two users: 'John Tester' (with a YouTube icon) and 'Jane Tester' (with a Spotify icon). The main content area has a teal header that reads 'Welcome to the chat page' and a sub-instruction: 'On the left you can select a person to talk to and enter the message you want to send below'. Below this, there's a bulleted list of suggestions: 'Start off a conversation by introducing yourself!', 'Break the ice by discussing common interests.', and 'Find a project you're both interested in and begin collaborating!'. A small 'Matches' button is located at the bottom right of this section.

### Send and receive messages to/from Potential matches

This screenshot shows the messaging interface. On the left, a sidebar titled 'Matches' lists three users: 'ArtistGal456' (YouTube icon), 'Jane Tester' (Spotify icon), and 'MusicianMan123' (YouTube icon). The main area displays a conversation between 'MusicianMan123' and 'ArtistGal456'. MusicianMan123 sent the message 'Hi there this is a test message', and ArtistGal456 responded with 'Hi MusicianMan123, this is also a test message'. At the bottom, there's a grey input field with the placeholder 'Type your message here!' and a blue 'Send' button.

## Community Page (Logged in)

 DoReMeet

Community Match Chat Profile

 Log Out

### Community Page

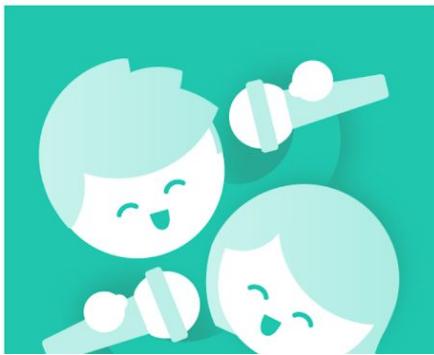
See how other people are using DoReMeet to bring their artistic dreams to life, or post some of your own amazing collaborations with fellow DoReMeet users

Check out our [Highlights section](#) to see the most liked posts of the month.

Share your work with us here!

### Recent Posts

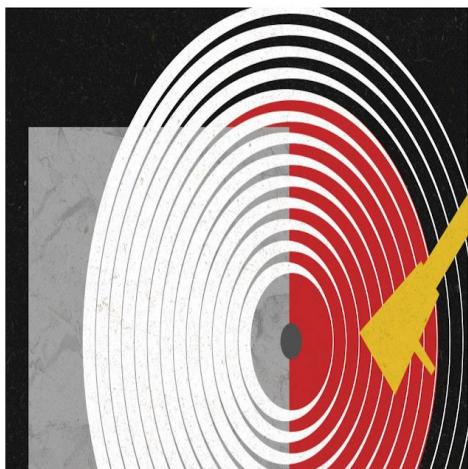


#### R&B Cover

1

Music

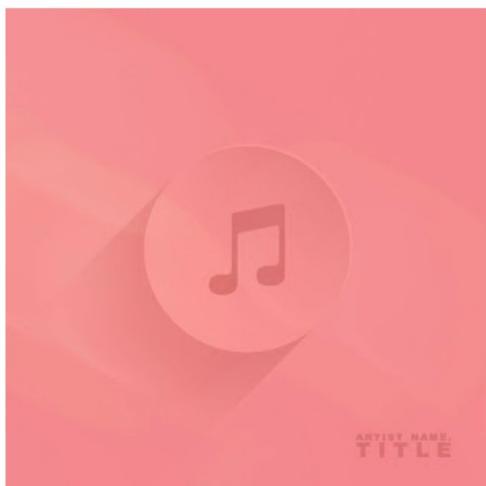
A collection of covers for famous 90s R&B songs



## Revive Instrumental

1  
Music

A collaboration of sounds from everyday life



## Pop Reimagined

1  
Music

Famous pop songs mixed with indie tunes



## Jazz

1  
Music

classic jazz collaboration with budding musicians



## My Album

1  
Music

A music album for everyone

## Check out our Highlights for this Month

Dance

Amateur Salsa Dance at Local Competition

Art

Oil Painting

Music

Song inspired by everyday noises

---

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)

DoReMeet © 2020



## Search and Voting (Community Page)

If Search result found -

Share your work with us here!

## Search Results



Jazz

Music

1

classic jazz collaboration with budding musicians

0 0

If Search result Not found -

Share your work with us here!

## Search Results

Sorry, we couldn't find anything

## Profile Page

DoReMeet

Community Match Chat Profile

Log Out

### Your Profile!



Edit

Name first last

Age:

Gender: f

Location: Placeholder Location (Hidden)

Art Category: Dance



Tags: +

Bio:

This is test user 1's bio (First Last)

Media:

#### Your Community Posts:



Art

Avocado Blob



Art

Egg Wobble



Music

Test

#### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

FAQs  
Guidelines  
Contact Us  
Terms & Conditions  
Privacy Policy

DoReMeet © 2020



## Settings/Preferences Page



Community Match Chat Profile

Log Out

# Settings

The settings you choose reflect your match preferences. Your potential matches will be based on these choices.  
You can update your settings anytime and get customized results.

[Edit Settings](#)

## Settings

First Name Currently:

Last Name Currently:

Your name will not be available for other users to view to protect your privacy

Gender Currently:  Male  Female  None

Date of Birth Currently:

 mm/dd/yyyy 

You must be 18 or older to use this site

Email Currently:

Your email is not viewable for other users for your privacy

Phone Number Currently: 123-456-7890

Required only for 2-factor identification

Username Currently:

Current Password:

New Password:

[Confirm Settings Changes](#)

## Preferences

Art Category Currently:

When switching categories make sure to update your skill level and content

Skill Level Currently:

Generally beginners have less than 5 years of experience and advanced has more than 7 years of experience

Minimum Age (18+) Currently:

The minimum age of people you will match with

Maximum Age Currently:

The maximum age of people you will match with

Preferred Match Gender Currently:  Male  Female  None

Your preferred gender for the people you will match with

Preferred Skill Level Currently:

Your preferred skill level for the people you will match with

Preferred Meeting Type Currently:

Your preference for meeting online or offline

Preferred Search Radius Currently (Miles):

Your preference for the maximum distance by which you'll match with people

[Confirm Preference Changes](#)

[Upgrade to Premium Account](#)

[Pause Account](#)

[Delete Account](#)

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)



DoReMeet © 2020

## Upgrade to Premium Account page



Community Match Chat Profile

Log Out

[Back to Settings](#)

### Upgrade your account

Increase your chances of finding the right match!

- Unlimited Likes
- No Ads
- Highlighted Profile
- Increased Search Radius
- Rewind on Potential Matches
- Bigger Portfolio Size

\$4.99 for 1 month



\$49.99 for 6 months



\$119.99 for 12 months



Card Holder's Name

Card Number

XXXX XXXX XXXX XXXX

Expiration Date (MM/YY) CVV

[Upgrade to Premium Account](#)

[End Premium Membership](#)

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

[Subscribe](#)

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)



DoReMeet © 2020

## FAQs

# FAQs

## Need Help?

Check the popularly asked questions, and if you still can't find the answer [Contact Us here](#)

Question 1



Question 2



Question 3



Question 4



Question 5



Question 6



### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)



DoReMeet © 2020

## Guidelines



Company   Community   Pricing

Log In

# GUIDELINES

## DoReMeet Guidelines

These Guidelines and our Terms and Conditions are designed to ensure every user has a safe and enjoyable experience on DoReMeet. Failure to adhere to these guidelines may result in losing access to our platform.

### Upload only your own photos

We take copyrights very seriously. If you don't own the rights to a photo or video, please don't post it.

### Respect all users

We're a very diverse community. This means you should respect other people's beliefs, interests and property while on DoReMeet. DoReMeet takes a strong stance against hate speech, rude or abusive behaviour, bullying, and misogyny. You should behave the same way on DoReMeet as you would in real life. Additionally we encourage all of our users to report anyone who does not follow these behavioural guidelines. As a community rooted in kindness and respect, we expect all of our users to respect each other, themselves, and the DoReMeet staff.

### Send original messages to other users

We strongly advise against copying and pasting the same message to every connection. We encourage our users to read profiles, learn about other people's interests, and send an appropriate, relevant message to each match. We promise this will increase your chances of engaging in interesting conversations.

### Don't pretend you're someone you're not

As previously stated, do not post photos that are not of you. We also recommend using our photo verification tool to let other users know that your profile is legit.

### DoReMeet photo guidelines:

- No kids on their own. They must be in the photo with an adult, and fully clothed.
- No photos in bikinis/swimwear indoors.
- No pictures in underwear.
- No Shirtless/underwear Mirror Selfies.
- Face must be clearly visible in all photos.
- No watermarks or text overlaid.
- No pornographic material.
- No graphic hunting photos.
- No guns.

\*Footer not pictured for the Guidelines page (For reference, the footer looks similar to all the other pages)

## Contact Us



Company   Community   Pricing

Log In

# CONTACT US

Please fill out the form with any questions/concerns/suggestions you have, and we will get back to you as soon as possible.

Thank you for being a valued customer!

Name

Enter Name

Email

Enter Email

Message/Question?

Share your Questions/Concerns/Suggestions

Send Message

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)



DoReMeet © 2020

## Terms & Conditions



Company    Community    Pricing

Log In

# Terms & Conditions

## DoReMeet Terms and Conditions of Use

Hey guys! Welcome to DoReMeet's Terms and Conditions of Use (these "Terms"). Our lawyers insist that we impose rules on users to protect all of our hard work. This is a contract between you and the DoReMeet Group (as defined further below) and we want you to know yours and our rights before you use the DoReMeet. Please take a few moments to read these Terms before enjoying the website, because once you access, view or use the website, you are going to be legally bound by these Terms (so probably best to read them first!).

### DoReMeet RULES

Before you can use our awesome website, you will need to register for an account ("Account"). In order to create an Account you must:

- be at least 18 years old; and
- be legally permitted to use the website by the laws of your home country.

You can create an Account via manual registration. For more information about what information we use and how we use it, please check out our Privacy Policy.

Unfortunately, we cannot allow you to use another person's Account without permission - that just wouldn't be fair! You'll have great fun on DoReMeet, but if you feel the need to leave, you can delete or pause your Account at any time by going to the 'Settings' page when you are logged in and clicking on the 'Delete account' or 'Pause Account' link. Your Account will be deleted immediately but it may take a little while for Your Content to be completely removed from the website. We will save your profile information in case you realise you miss us and you decide to restore your Account (which you can do within 30 days of de-activating your Account). If you delete your Account and try to create a new account within this time period using the same credentials, we will re-activate your Account for you. If you pause your account, we'll simply mark your account as inactive, and not show your profile to any of our other users. At the same time, you will not be able to see anyone else's account. You can however see and chat with the people you've already matched with and update your settings or 'Unpause' your account anytime.

We reserve the right at our sole discretion to terminate or suspend any Account, or make use of any operational, technological, legal or other means available to enforce the Terms (including without limitation blocking specific IP addresses), at any time without liability and without the need to give you prior notice. You may not access, tamper with, or use non-public areas of the website or our systems. Certain portions of the website may not be accessible if you have not registered for an Account.

## DISCLAIMER

Brace yourselves, this may look daunting but it is very important!

THE SITE, OUR CONTENT, AND MEMBER CONTENT ARE ALL PROVIDED TO YOU "AS IS" AND "AS AVAILABLE" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, FITNESS FOR A PARTICULAR PURPOSE, TITLE, OR NON-INFRINGEMENT.

SHOULD APPLICABLE LAW NOT PERMIT THE FOREGOING EXCLUSION OF EXPRESS OR IMPLIED WARRANTIES, THEN WE GRANT THE MINIMUM EXPRESS OR IMPLIED WARRANTY REQUIRED BY APPLICABLE LAW. NO ADVICE OR INFORMATION, WHETHER ORAL OR WRITTEN, SHALL CREATE ANY WARRANTY, REPRESENTATION OR GUARANTEE NOT EXPRESSLY STATED IN THIS SECTION.

ADDITIONALLY, WE DO NOT MAKE ANY WARRANTIES THAT THE APP OR SITE WILL BE UNINTERRUPTED, SECURE OR ERROR FREE OR THAT YOUR USE OF THE APP OR SITE WILL MEET YOUR EXPECTATIONS, OR THAT THE APP, SITE, OUR CONTENT, ANY MEMBER CONTENT, OR ANY PORTION THEREOF, IS CORRECT, ACCURATE, OR RELIABLE. YOUR USE OF THE APP OR SITE IS AT YOUR OWN RISK. YOU ARE SOLELY RESPONSIBLE FOR YOUR INTERACTIONS WITH OTHER MEMBERS. THE DOREMEET GROUP IS NOT RESPONSIBLE FOR THE CONDUCT OF ANY USER. DOREMEET DOES NOT CONDUCT CRIMINAL BACKGROUND CHECKS ON ITS MEMBERS.

NEITHER US NOR ANY OWNER WILL BE LIABLE FOR ANY DAMAGES, DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL, OR PUNITIVE, INCLUDING, WITHOUT LIMITATION, LOSS OF DATA, INCOME, PROFIT OR GOODWILL, LOSS OF OR DAMAGE TO PROPERTY AND CLAIMS OF THIRD PARTIES ARISING OUT OF YOUR ACCESS TO OR USE OF THE APP, SITE, OUR CONTENT, OR ANY MEMBER CONTENT, HOWEVER CAUSED, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), PROPRIETARY RIGHTS INFRINGEMENT, PRODUCT LIABILITY OR OTHERWISE.

THE FOREGOING SHALL APPLY EVEN IF WE WERE ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IF YOU BECOME DISSATISFIED IN ANY WAY WITH THE APP OR SITE, YOUR SOLE AND EXCLUSIVE REMEDY IS TO STOP YOUR USE OF THE APP AND SITE.

YOU HEREBY WAIVE ANY AND ALL CLAIMS ARISING OUT OF YOUR USE OF THE APP OR SITE. BECAUSE SOME STATES DO NOT ALLOW THE DISCLAIMER OF IMPLIED WARRANTIES OR THE EXCLUSION OR LIMITATION OF CERTAIN TYPES OF DAMAGES, THESE PROVISIONS MAY NOT APPLY TO YOU. IF ANY PORTION OF THIS LIMITATION ON LIABILITY IS FOUND TO BE INVALID OR UNENFORCEABLE FOR ANY REASON, THEN OUR AGGREGATE LIABILITY SHALL NOT EXCEED ONE HUNDRED DOLLARS (\$100).

THE LIMITATION OF LIABILITY HEREIN IS A FUNDAMENTAL ELEMENT OF THE BASIS OF THE BARGAIN AND REFLECTS A FAIR ALLOCATION OF RISK. THE APP AND SITE WOULD NOT BE PROVIDED WITHOUT SUCH LIMITATIONS AND YOU AGREE THAT THE LIMITATIONS AND EXCLUSIONS OF LIABILITY, DISCLAIMERS AND EXCLUSIVE REMEDIES SPECIFIED HEREIN WILL SURVIVE EVEN IF FOUND TO HAVE FAILED IN THEIR ESSENTIAL PURPOSE.

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)



DoReMeet © 2020

## Privacy Policy

# PRIVACY POLICY

## COLLECTION OF INFORMATION

### Registration Information

When you sign up and create an "account" with DoReMeet, we may collect certain information ("Registration Information") about you, such as:

- Name;
- Username;
- Email address;
  
- Mobile number;
- Gender identity;
- Date of birth;
- Photographs;
- Location;
- Login information and friend lists for social media accounts that you connect to your DoReMeet Account (this could include, for example, your Facebook and Instagram accounts ("Social Media Accounts").

You will also be required to create a password in connection with the registration of your Account. Once you register, you will be able to review and change this information at any time by accessing your "Profile" page! You control your user profile and are able to correct or update (other than your email address) your information at any time by just logging in to DoReMeet.

The information we collect helps to enhance the App and verify our Users (robots are not welcome!). Registration Information such as your name and username may be visible to the public on your profile page.

### Subscribe to Our Newsletter

Get the company details, company news and special offers delivered right to your inbox.

Enter your email here

Subscribe

[FAQs](#)  
[Guidelines](#)  
[Contact Us](#)  
[Terms & Conditions](#)  
[Privacy Policy](#)



DoReMeet © 2020

## 5. Key Database Table Screenshots

### 1. Account Table

```
CREATE TABLE IF NOT EXISTS `mydb`.`account` (
  `account_id` TINYINT NOT NULL AUTO_INCREMENT,
  `username` VARCHAR(64) NOT NULL,
  `password` VARCHAR(255) NOT NULL,
  `acc_created` DATETIME NULL,
  -- `activate` INT acc_created activate NOT NULL DEFAULT 0,
  `user` TINYINT NULL,
  PRIMARY KEY (`account_id`),
  INDEX `account_user_FK_idx` (`user` ASC),
  UNIQUE INDEX `username_UNIQUE` (`username` ASC),
  CONSTRAINT `account_user_FK`
    FOREIGN KEY (`user`)
    REFERENCES `mydb`.`user` (`user_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB;
```

saves username and password of the user  
Note password is encrypted

```
CREATE TABLE IF NOT EXISTS `mydb`.`accountType` (
  `account_type_id` TINYINT NOT NULL AUTO_INCREMENT,
  `account_type_desc` VARCHAR(45) NOT NULL,
  `account` TINYINT NOT NULL,
  PRIMARY KEY (`account_type_id`),
  INDEX `accountType_account_FK_idx` (`account` ASC),
  CONSTRAINT `accountType_account_FK`
    FOREIGN KEY (`account`)
    REFERENCES `mydb`.`account` (`account_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB;
```

identifies whether the account is general or premium

## 2. Match Table

```
DROP TABLE IF EXISTS `mydb`.`matches` ;  
  
CREATE TABLE IF NOT EXISTS `mydb`.`matches` (  
  `matches_id` TINYINT NOT NULL AUTO_INCREMENT,  
  `client_username` VARCHAR(64) NOT NULL,  
  `client_match_status` VARCHAR(10) NULL,  
  PRIMARY KEY (`matches_id`),  
  UNIQUE INDEX `client_username_UNIQUE` (`client_username` ASC))  
ENGINE = InnoDB;
```

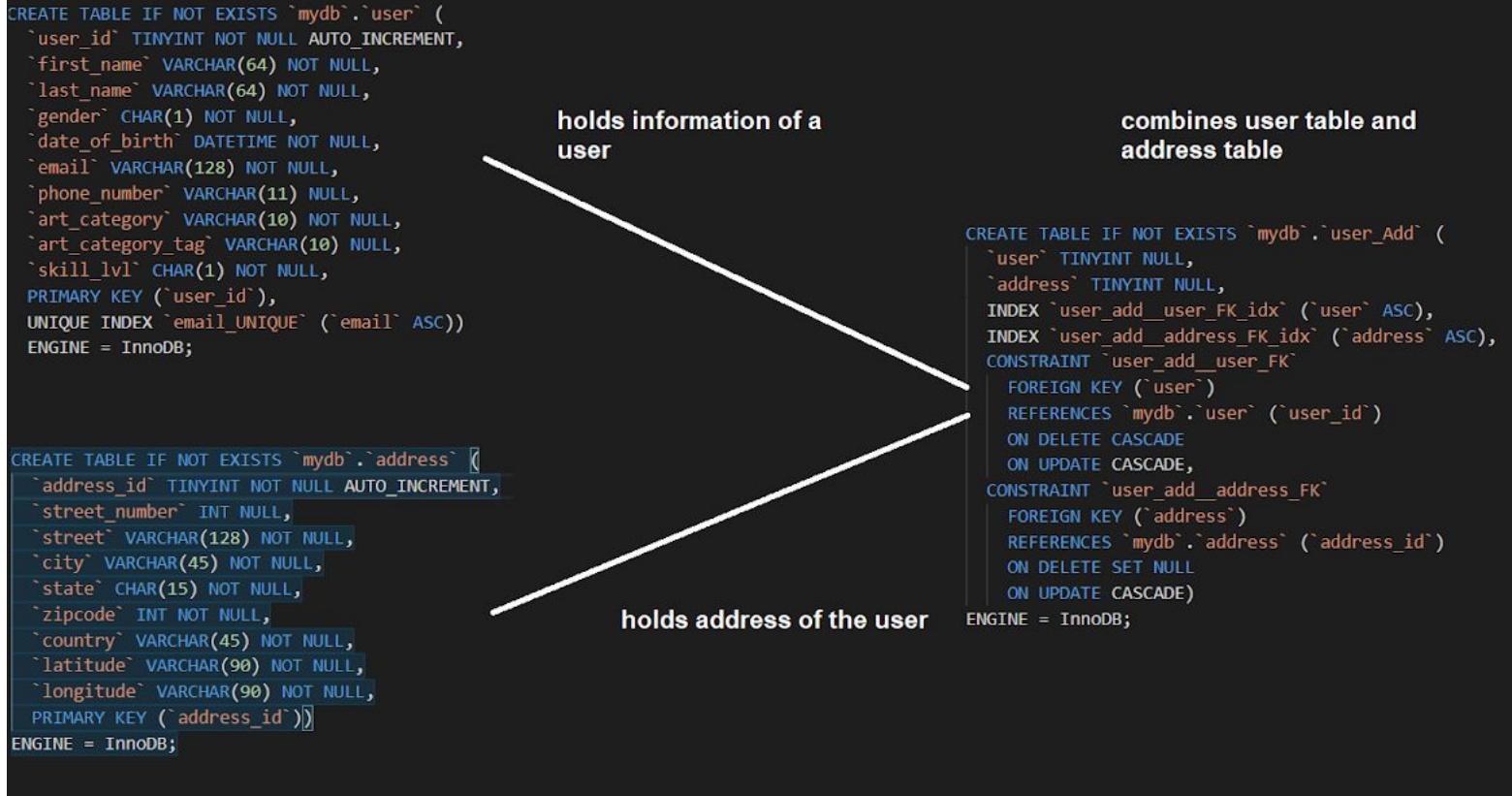
keep track of the matches

### 3. Preferences Table

```
CREATE TABLE IF NOT EXISTS `mydb`.`preferences` (
  `preferences_id` TINYINT NOT NULL AUTO_INCREMENT,
  `min_age` CHAR(2) NULL,
  `max_age` CHAR(2) NULL,
  `min_location` CHAR(3) NULL,
  `max_location` CHAR(3) NULL,
  `gender` CHAR(1) NULL,
  `art_preference` VARCHAR(10) NULL,
  `skill_lvl_pref` CHAR(1) NULL,
  `meeting_pref` VARCHAR(10) NULL,
  `user` TINYINT NOT NULL,
  `matches` TINYINT NULL,
  PRIMARY KEY (`preferences_id`),
  INDEX `preferences_user_FK_idx` (`user` ASC),
  INDEX `preferences_matches_FK_idx` (`matches` ASC),
  CONSTRAINT `preferences_user_FK`
    FOREIGN KEY (`user`)
    REFERENCES `mydb`.`user` (`user_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `preferences_matches_FK`
    FOREIGN KEY (`matches`)
    REFERENCES `mydb`.`matches` (`matches_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB;
```

saves preferences of users  
when matches are being made

#### 4. User and Address Table



## 7. Task Management System (Monday.com)

Milestone 1	Subite...	Owner	Priority	Status	Timeline	+
Version 1		N	High	Done	✓ Sep 23 - 24	
subitems	Owner		Status	Date	Notes	+
Executive Summary	N		Done			
Use Cases	V T J M E +2		Done			
Data Items and Entities	N		Done			
Functional Requirements	L C M B M F +2		Done			
Non Functional Requirements	J V T M F +2		Done			
Competitive Analysis	N		Done			
System Architecture and Technologies	N		Done			
Checklist	N		Done			
+ Add						
Version 2		N	High	Done	✓ Oct 1 - 11	
Update with Feedback	N				-	
+ Add						
	N					

Milestone 2	Subite...	Owner	Priority	Status	Timeline	+
Version 1	↳ 9	Q	High	Done	✓ Oct 15 - 29	
subitems	Owner		Status	Date	Notes	+
Main Data Items and Entities V2	N		Done	✓ Oct-24		
Functional Requirements V2	N		Done	✓ Oct-24		
UI Mockups and Storyboards	L C V T		Done	✓ Oct-26		
High level database architecture and or...	S		Done	✓ Oct-27		
High Level APIs and Main Algorithms	J M B		Done	✓ Oct-27		
High Level UML Diagrams	M B		Done	✓ Oct-27		
High Level Application Network and De...	L C		Done	✓ Oct-27 ✓	Needs Review	
Actual key risks	N		Done	✓ Oct-27		
Project management	N		Done	✓ Oct-27		
+ Add						
Vertical SW Prototype	↳ 1	N L C V T +3	High	Done	✓ Oct 27 - 29	
Version 2	↳ 1	N	Medium	Done	✓ Oct 27 - 31	
subitems	Owner		Status	Date	Notes	+
Update with Feedback	N		Done			

Milestone 3	Subite...	Owner	Priority	Status	Timeline	+
Version 1		5		High	Done	Oct 29 - Nov 19
subitems		Owner		Status	Date	Notes
Data Definitions			N	Done		
Functional Requirements			N LC MB MP VT J	Done		
Wireframes			N	Done		
Database Architecture			LC MP MB	Done		
High Level Diagrams			MB LC	Done		
+ Add						
Version 2		1		Medium	Done	Nov 19 - 22
subitems		Owner		Status	Date	Notes
Update with Feedback			N	Done		Were stuck
+ Add						
Horizontal Prototype		1		High	Done	-
subitems		Owner		Status	Date	Notes
Connect backend to frontend			MB LC MP	Done		
+ Add						
Milestone 4	Subite...	Owner	Priority	Status	Timeline	+
Version 1		6		High	Done	Nov 11 - Dec 11
subitems		Owner		Status	Date	Notes
Product Summary			N	Done		
Usability Test			N LC J MB VT MP	Done		
QA Test			N MP VT MB J LC	Done		
Code Review			N LC	Done		
Check on NF Reqs			N MP MB LC VT J	Done		
Check on Security Practices			N MP	Done		
+ Add						
Version 2		1		Medium	Done	Dec 11 - 17
subitems		Owner		Status	Date	Notes
Update with Feedback			N	Done		
+ Add						
Milestone 5	Subite...	Owner	Priority	Status	Timeline	+
Document		1		High	Done	Dec 10 - 17
subitems		Owner		Status	Date	Notes
Post Analysis			N	Done		
+ Add						
Send Emails				High	Done	Dec 16 - 17
+ Add						

Do Re Meet Website

	Subitem...	Owner	Priority	Status	Timeline
Set Up EC2		L C V T	High	Done	✓ Oct 1 - 29
Set up Apache2 Server		L C	High	Done	✓ Oct 1 - 29
Frontend		N L C V T	High	Done	✓ Oct 23 - Dec 17
subitems	Owner	Status	Date	Notes	+
Landing Page	N	Done			
Login Page	N	Done			
Sign Up Page	N	Done			
Match	L C V T	Done			
Chat	V T L C	Working on it		Sockets Set up	
Community	L C V T	Done			
Profile	L C V T	Done			
404	L C	Done			
Footer Pages	N	Done			
Settings	L C	Done			
Premium Account	L C	Done			

Backend

	Owner	Status	Date	Notes	+
Make Connections	M P M B	Done			
Send Receive Data from DB	M B M P	Done			
Set up axios calls	M P M B	Done			
Set up functions for Community	M B	Done			
Functions for chat	M B L C	Working on it			
Functions for community search	M B L C M P	Done			
functions for community highlights	M B	Done			
Voting - Community	M B	Done			
Settings	M B L C	Done			
Update Profile	M B V T	Done			
Login sessions	M B	Done			
User location and radius setup	M P	Working on it			

Database		Owner	Status	Date	Notes	
subitems	Owner					
Set up DB Tables	○	MF J	Done			
Connect to backend	○	J MP	Done			
Send information	○	MF MB	Done			
Store Updates	○	MF MB	Done			
Media File Path	○	MP	Done			
Password encryption	○	MF J	Done			
Location IQ API	○	MP	Done			
+ Add						

Deploying to Server		Owner	Status	Date	Notes	
subitems	Owner					
Run Node forever	○	LC	Working on it			
+ Add						

## 8. Team Member Contributions

### Combined Contributions

Reply all | Delete | Junk | Block | ...

#### CSC 648 Team 2 Team Contributions



Nimiksha Mahajan  
Thu 12/17/2020 11:40 AM

To: Meet Patel; Mike Bolanos; Vincent K. Tse; Jungsun Eoh; Luke Chang-Knezevich

Hi everyone,

Here are everyone's contributions to our team project throughout the semester. Please review the contributions for all the team members.

##### **Nimiksha Mahajan - Team Lead**

###### Documentation

- Milestone 1 - 5 documentation
- Identify Functional and Non-Functional requirements, along with other specs like Use Cases, Competitive Analysis and Product Summary
- Wireframes and Product Diagrams

###### Frontend

- Design landing page, auxiliary footer pages (5)
- Login and sign-up pages
- Reusable site components - Card Elements, Footer, Navbar, Banner, Buttons
- Site Responsiveness
- Browser Compatibility Check
- Website design and flow

###### Other

- QA Testing and Usability Testing
- Troubleshooting EC2 instance setup
- Code Organization and cleanup
- Code Review
- Minor git cleanup
- Project Management for DoReMeet
- Delegated Tasks based on roles and responsibilities
- Check in with team members to ensure we were on track or if they needed support
- Weekly check in with Backend and Frontend leads
- Led weekly meetings to check on product status and (re)prioritize tasks to ensure timely delivery of milestones
- General Troubleshooting with issues encountered
- Communication with class CTO

##### **Meet Patel - Backend & Database Lead**

###### Database

- Database creation
- ALL the work related to database including ERD, etc. (EER, Schema)
- Local connection between backend and Workbench
- Local connection between Ec2 and Workbench/rds
- Local connection between Workbench/rds and backend

« Reply all | ⚡ Delete ⚡ Junk ⚡ Block ⚡ ...

## CSC 648 Team 2 Team Contributions

- Local connection between wordendradius and Backend

### Backend

- Routing for backend and changing mysql to mysql2 (Dropped)
- Sign up
  - Connection of frontend and backend
  - Queries for inserting data into user table, account table, address table, and more
  - Encryption password
- Log in Done by Mike (Reformatted by Me/Meet)
  - Encryption password
  - Making sure user is able to log in after password is encrypted
- Location IQ
  - Sample Version/ test version of getting address using html/css and js
  - Sample version was implemented by Luck in frontend (react js)
  - Checking user is within radius or not
  - Implemented related work in backend
  - Checking radius – in progress
  - Query about sending location into database when user sign up
- Other
  - Reformat few backend code and query

### Luke Chang-Knezevich - Frontend Lead

- Contributed to documentation stuff (UI mockup drawings and storyboards, functional and nonfunctional reqs, use cases, data items & entities, high-level UML diagram, network and deployment diagram, key risks, general documentation review)
- General code contributions (Frontend codebase, community page, chat page, settings page, profile page, sign up/ log in page, match page, premium page, 404page, QA testing, code cleanup and organization)
- EC2 stuff (General EC2 setup, apache web server troubleshooting, frontend backend connection)
- Setup and planning of general front-end codebase
- Front-end design (logo, design/aesthetic)
- General front-end organization (meetings, planning, etc.)

## CSC 648 Team 2 Team Contributions

### Mike Bolanos - GitHub Master & Backend Engineer

Backend:

- Community Page
  - Searching, Posting, Voting user actions
  - Retrieve necessary data for this page (Recent post, Highlights)
- Users
  - Signup/Register/Login (no encryption) and Logout.
- Settings
  - Changing user information and account credentials
  - Updating search preferences
- Profile
  - Retrieving necessary data for this page (User , Profile , Community Posts)
  - Editing of user's profile (Images(Profile, Media), Bio)
- Match
  - Retrieving necessary data for this page (User, Match, Profile, Preferences)
  - Implementation of Pass, Connect, Who (will/will not) be matched
- Chat
  - Listing of users who are successful matches

Frontend:

- Minor corrections to variables and object placements
- Gathering of data needed to send to backend

Database:

- Revised database to fit with changes to project's requirements
  - corrected incompatible datatypes, tables, columns
- Adding of received sample data (Converting into database format)

Other:

- Connection Loop
  - assisted setup of connection loop for local machine and hosted website
    - Frontend-->Backend-->Database-->Backend-->Frontend
- GitHub
  - Branch Management
    - Sorting, Adding, Removing branches
    - Manual merges due to merge conflicts
  - General testing
    - Checking if a commit did not break anything
    - Testing compatibility for merging

## CSC 648 Team 2 Team Contributions

### Vincent Tse - Server Admin & Frontend Engineer

Frontend to backend connection (locally)

- Getting and sending back data, did this in a zoom meeting (axios calls and a couple backend app.js functions).
- Community Page
  - Recent posts
  - Search posts
  - Formatting of page
- Match Page
  - Formatting of page
  - Page Responsiveness
- Chat Page
  - First version of page (had to change after horizontal prototype review)
- Profile Page
  - Social media links
  - Bio
  - Tags (in progress)
  - Media (in progress)
  - Your community posts (in progress)
  - Page Responsiveness
- Frontend to backend connection (EC2 instance)
  - Did this in a zoom meeting, figured out we needed to add global axios settings.
- Milestone Documentation and Review

### Jungsun Eoh - Database & Backend Engineer

#### Backend and Database

- Worked on Password Recovery implementation
- Working on Password Recovery implementation (encrypted version)
- Set up Nodemailer API
- Password recovery frontend page
- Password recovery backend frontend connection
- Worked on the Update password feature
- Working on the Update password feature (encrypted version)
- Log in and Sign-up backend configuration
- Set up redirects to have better user experience.
- Backend Code (connect to database, insert into database)
- Review Connection RDS to local database
- Review Connection backend on EC2 instance

General documentation stuff(Data Definitions,

- Functional Requirements, Documentation Review, Main Data Items and Entities, High level APIs and Algorithms, High Level UML Diagram, Key Risks...)

I also want to acknowledge all of y'alls hard work as a part of this team. Everyone has been great about communicating and actively participating in the product development. You were all willing to step up to help whenever and wherever needed and truly worked as team.

I truly appreciate all your work. Thank you!

Best,  
Nimiksha

CSC 648 Team 2 Team Contributions

Vincent K. Tse  
Thu 12/17/2020 11:59 AM  
To: Luke Chang-Knezevich; Nimiksha Mahajan; Meet Patel; Mike Bolanos; Jungsun Eoh

The contributions look good to me as well, good job team.

...

Reply | Reply all | Forward

Meet Patel  
Thu 12/17/2020 11:59 AM  
To: Mike Bolanos; Vincent K. Tse; Jungsun Eoh; Luke Chang-Knezevich; Nimiksha Mahajan

Hello everyone, it was a pleasure to working with you all, I hereby confirm and I agree to the team contribution given above.

Sent from my Verizon, Samsung Galaxy smartphone  
Get [Outlook for Android](#)

...

CSC 648 Team 2 Team Contributions

Luke Chang-Knezevich  
Thu 12/17/2020 11:56 AM  
To: Nimiksha Mahajan; Meet Patel; Mike Bolanos; Vincent K. Tse; Jungsun Eoh

Hi Team,

The contributions look good to me. Thanks for all your hard work over the semester! It's been a pleasure working with y'all.

I wish you the best of luck in your future endeavors!

Regards,  
Luke Chang-Knezevich

...

CSC 648 Team 2 Team Contributions

Mike Bolanos  
Thu 12/17/2020 12:04 PM  
To: Nimiksha Mahajan

Yup, everything looks correct to me.

...

Are the suggestions above helpful? Yes No

Great, thanks! Great, thank you! Thanks for confirming!

Reply | Forward

CSC 648 Team 2 Team Contributions

Jungsun Eoh  
Thu 12/17/2020 1:35 PM  
To: Nimiksha Mahajan  
Cc: Meet Patel; Mike Bolanos; Vincent K. Tse; Luke Chang-Knezevich

Good work team!  
We could sleep tonight.

## Individual Member Contributions:

### Nimiksha Mahajan - Team Lead

Fw: CSC 648 Team 02 M5 and Product Contributions

 Nimiksha Mahajan  
Thu 12/17/2020 10:40 AM  
To: Jungsun Eoh

 Nimiksha Mahajan  
Thu 12/17/2020 10:00 AM  
To: Mike Bolanos; Meet Patel; Luke Chang-Knezevich; Vincent K. Tse

Hi everyone,

Here are my contributions to m5 and our product DoReMeet in overall.

#### M5:

- M5 Documentation
- Making site responsive - including all Landing pages, frontend pages, Sign Up, Settings
- Browser Compatibility check
- Communicating with team to ensure all Priority 1 functional requirements and non-functional requirements are completed
- Usability Testing (done independently to check website flow)

#### Overall:

- Milestone 1 - 5 documentation
- Identify Functional and Non-Functional requirements, along with other specs like Competitive Analysis and Product Summary
- Wireframes and Product Diagrams
- Frontend - Design landing page, auxiliary footer pages (5), login and sign-up pages
- Reusable site components - Card Elements, Footer, Navbar, Banner, Buttons
- Troubleshooting EC2 instance setup
- Code Review
- Code Organization and cleanup
- Project Management for DoReMeet
- Delegated Tasks based on roles and responsibilities
- Check in with team members to ensure we were on track or if they needed support
- Led weekly meetings to check on product status and (re)prioritize tasks to ensure timely delivery of milestones
- General Troubleshooting
- Communication with class CTO

**GitHub Contributions to date:** 61 (Includes commits for Code organization & cleanup, Milestone documents and other credential updates)

Thank you,  
Nimiksha

## Meet Patel - Backend & Database Lead

### Updated version contribution



Meet Patel

Wed 12/16/2020 10:45 AM

To: Nimiksha Mahajan; Luke Chang-Knezevich; Mike Bolanos; Vincent K. Tse; Jungsun Eoh

#### Database

- Local connection between backend and Workbench
- Local connection between Ec2 and Workbench/rds
- Local connection between Workbench/rds and backend
- Database creation
- ALL the work related to database including ERD, etc. (EER, Schema)

#### Backend

- Routing for backend and changing mysql to mysql2 (Dropped)
- Sign up
- Connection of frontend and backend
- Queries for inserting data into user table, account table, address table, and more
- Encryption password
- Log in
- Done by Mike (Reformatted by Me/Meet)
- Encryption password
- Making sure user is able to log in after password is encrypted
- Location IQ
- Sample Version/ test version of getting address using html/css and js
- Sample version was implemented by Luck in frontend (react js)
- Checking user is within radius or not
- Implemented related work in backend
- Checking radius – in progress
- Query about sending location into database when user sign up
- Chat
- In progress
- Other
- Reformat few backend code and query

Git hub commits 17

## Updated version contribution



Nimiksha Mahajan  
Thu 12/17/2020 1:39 PM  
To: Meet Patel



I agree to all these contributions. Thank you for all your hard work Meet!

It's been a pleasure.

Best,  
Nimiksha

...

---

**From:** Meet Patel <mpatel6@mail.sfsu.edu>  
**Sent:** Wednesday, December 16, 2020 10:45 AM  
**To:** Nimiksha Mahajan <nmahajan@mail.sfsu.edu>; Luke Chang-Knezevich <lchangknezevich@mail.sfsu.edu>; Mike Bolanos <mbolanos1@mail.sfsu.edu>; Vincent K. Tse <vtse3@mail.sfsu.edu>; Jungsun Eoh <jeoh@mail.sfsu.edu>  
**Subject:** Updated version contribution

Database

## Luke Chang-Knezevich - Frontend Lead

### CSC 648 Team 02 Milestone 5 Team Member Contributions

LC

Luke Chang-Knezevich  
Sun 12/13/2020 12:51 AM  
To: Nimiksha Mahajan; Meet Patel; Mike Bolanos; Jungsun Eoh; Vincent K. Tse



Hey team,

Here are my contributions for milestone 5.

#### Luke Chang-Knezevich (Frontend lead) Contributions:

- Contributed to documentation stuff (UI mockup drawings and storyboards, functional and nonfunctional reqs, use cases, data items & entities, high-level UML diagram, network and deployment diagram, key risks, general documentation review)
- General code contributions (Frontend codebase, community page, chat page, settings page, profile page, sign up / log in page, match page, premium page, 404page, QA testing, code cleanup and organization)
- EC2 stuff (General EC2 setup, apache web server troubleshooting, frontend backend connection)

Number of submissions/commits to GitHub: 63 (as of 12/13/2020)

#### Frontend lead specific stuff:

- Setup and planning of general front-end codebase
- Front-end design (logo, design/aesthetic)
- General front-end organization (meetings, planning, etc.)

I tried to keep things pretty general, let me know if there's anything that isn't right.

Thanks,  
Luke Chang-Knezevich

### CSC 648 Team 02 Milestone 5 Team Member Contributions

NM

Nimiksha Mahajan  
Thu 12/17/2020 1:40 PM  
To: Luke Chang-Knezevich



I agree to these contributions. Thank you for all your work Luke!

It's been a pleasure.

Best,  
Nimiksha

...

---

**From:** Luke Chang-Knezevich <lchangknezevich@mail.sfsu.edu>  
**Sent:** Sunday, December 13, 2020 12:51 AM  
**To:** Nimiksha Mahajan <nmahajan@mail.sfsu.edu>; Meet Patel <mpatel6@mail.sfsu.edu>; Mike Bolanos <mbolanos1@mail.sfsu.edu>; Jungsun Eoh <jeoh@mail.sfsu.edu>; Vincent K. Tse <vtse3@mail.sfsu.edu>  
**Subject:** CSC 648 Team 02 Milestone 5 Team Member Contributions

Hey team,

Here are my contributions for milestone 5.

## Mike Bolanos - GitHub Master & Backend Engineer

### CSC 648 Team 02 Milestone 5 Section 7 Team Member Contributions



Mike Bolanos

Tue 12/15/2020 10:33 PM

To: Nimiksha Mahajan; Luke Chang-Knezevich; Vincent K. Tse; Meet Patel; Jungsun Eoh; Mike Bolanos

Backend:

1. Community Page
  - a. Searching, Posting, Voting user actions
  - b. Retrieve necessary data for this page (Recent post, Highlights)
2. Users
  - a. Signup/Register/Login (no encryption) and Logout.
3. Settings
  - a. Changing user information and account credentials
  - b. Updating search preferences
4. Profile
  - a. Retrieving necessary data for this page (User , Profile , Community Posts)
  - b. Editing of user's profile (Images(Profile, Media), Bio)
5. Match
  - a. Retrieving necessary data for this page (User, Match, Profile, Preferences)
  - b. Implementation of Pass, Connect, Who (will/will not) be matched
6. Chat
  - a. Listing of users who are successful matches

Frontend:

1. Minor corrections to variables and object placements
2. Gathering of data needed to send to backend

Database:

1. Revised database to fit with changes to project's requirements
  - a. corrected incompatible datatypes, tables, columns
2. Adding of received sample data (Converting into database format)

Other:

1. Connection Loop
  - a. assisted setup of connection loop for local machine and hosted website
    - i. Frontend-->Backend-->Database-->Backend-->Frontend
2. GitHub
  - a. Branch Management
    - i. Sorting, Adding, Removing branches
    - ii. Manual merges due to merge conflicts
  - a. General testing
    - i. Checking if a commit did not break anything
    - ii. Testing compatibility for merging

GitHub Commits: 96

[Reply](#)

[Reply all](#)

[Forward](#)

## CSC 648 Team 02 Milestone 5 Section 7 Team Member Contributions



Nimiksha Mahajan  
Thu 12/17/2020 1:43 PM  
To: Mike Bolanos



I agree to these contributions. Thank you for all your hard work Mike!

It's been a pleasure!

Best,  
Nimiksha

...

---

**From:** Mike Bolanos <mbolanos1@mail.sfsu.edu>

**Sent:** Tuesday, December 15, 2020 10:33 PM

**To:** Nimiksha Mahajan <nmahajan@mail.sfsu.edu>; Luke Chang-Knezevich <lchangknezevich@mail.sfsu.edu>; Vincent K. Tse <vtse3@mail.sfsu.edu>; Meet Patel <mpatel6@mail.sfsu.edu>; Jungsun Eoh <jeoh@mail.sfsu.edu>; Mike Bolanos <mbolanos1@mail.sfsu.edu>

**Subject:** CSC 648 Team 02 Milestone 5 Section 7 Team Member Contributions

Backend:

## Vincent Tse - Server Admin & Frontend Engineer

### CSC 648 Team 02 Milestone 5 Contributions

VT

Vincent K. Tse

Mon 12/14/2020 4:56 PM

To: Nimiksha Mahajan; Mike Bolanos; Jungsun Eoh; Luke Chang-Knezevich; Meet Patel

These are contributions to milestone 5 as of 12/13 (# of github submissions = 33):

1. Frontend to backend connection (locally)
  - a. Getting and sending back data, did this in a zoom meeting (axios calls and a couple backend app.js functions).
2. Community Page
  - a. Recent posts
  - b. Search posts
  - c. Formatting of page
3. Match Page
  - a. Formatting of page
4. Chat Page
  - a. First version of page (had to change after horizontal prototype review)
5. Profile Page
  - a. Social media links
  - b. Bio
  - c. Tags (in progress)
  - d. Media (in progress)
  - e. Your community posts (in progress)
6. Frontend to backend connection (EC2 instance)
  - a. Did this in a zoom meeting, figured out we needed to add global axios settings.

[Reply](#)

[Reply all](#)

[Forward](#)

### CSC 648 Team 02 Milestone 5 Contributions

NM

Nimiksha Mahajan

Thu 12/17/2020 1:41 PM

To: Vincent K. Tse



I agree to these contributions. Thank you for all your work Vincent!

It's been a pleasure.

Best,  
Nimiksha

...

**From:** Vincent K. Tse <vtse3@mail.sfsu.edu>  
**Sent:** Monday, December 14, 2020 4:56 PM  
**To:** Nimiksha Mahajan <nmahajan@mail.sfsu.edu>; Mike Bolanos <mbolanos1@mail.sfsu.edu>; Jungsun Eoh <jeoh@mail.sfsu.edu>; Luke Chang-Knezevich <lchangknezevich@mail.sfsu.edu>; Meet Patel <mpatel6@mail.sfsu.edu>  
**Subject:** CSC 648 Team 02 Milestone 5 Contributions

These are contributions to milestone 5 as of 12/13 (# of github submissions = 33):

## Jungsun Eoh - Database & Backend Engineer

 Jungsun Eoh

Thu 12/17/2020 10:45 AM

To: Nimiksha Mahajan; Mike Bolanos; Vincent K. Tse; Meet Patel; Luke Chang-Knezevich

Git hub commit 22

Worked on Password Recovery implementation  
Working on Password Recovery implementation (encrypted version)  
Set up Nodemailer API  
Password recovery frontend page  
Password recovery backend frontend connection

Worked on the Update password feature  
Working on the Update password feature (encrypted version)

Log in and Sign up backend configuration  
Set up redirects to have better user experience.  
Backend Code (connect to database, insert into database)

Review Connection RDS to local database  
Review Connection backend on EC2 instance

General documentation stuff(Data Definitions,  
Functional Requirements, Documentation Review, Main Data Items and Entities, High level APIs and Algorithms, High Level UML Diagram, Key Risks...)

[Reply](#)    [Reply all](#)    [Forward](#)

### M5 contribution



Nimiksha Mahajan  
Thu 12/17/2020 1:38 PM

To: Jungsun Eoh



I agree to these contributions. Thank you for all your work June!

It's been a pleasure.

Best,  
Nimiksha

...

**From:** Jungsun Eoh <jeoh@mail.sfsu.edu>  
**Sent:** Thursday, December 17, 2020 10:45 AM  
**To:** Nimiksha Mahajan <nmahajan@mail.sfsu.edu>; Mike Bolanos <mbolanos1@mail.sfsu.edu>; Vincent K. Tse <vtse3@mail.sfsu.edu>; Meet Patel <mpatel6@mail.sfsu.edu>; Luke Chang-Knezevich <lchangknezevich@mail.sfsu.edu>  
**Subject:** M5 contribution

Git hub commit 22

## 10. Post Analysis

The design and development of DoReMeet has been a fun, yet challenging one. From coining this idea almost 3 months ago, we have come a long way in bringing it to fruition where our product is (almost) ready and functioning as we wanted. However there have encountered our fair share of challenges throughout this process. A few of our main challenges were:

- EC2 instance error - the first instance we created got corrupted, which we assume was because we tried to set up Elastic IP. We couldn't SSH into it. Also, it started exceeding the free tier limit and charged us, when it wasn't even running.
- Setting Up the EC2 instance - We faced some major issues setting up our EC2 instance successfully, and deploying the apache 2 server on the instance. We were facing some HTAccess issues, which prevented us from linking/redirecting our pages.
- Issues connecting backend and frontend on the server, due to permissions and virtual host settings
- Issues connecting to the database
- CORS Error
- Error tracking sessions when the user logs into the website, which caused the other functionality to not work as required
- Errors with building functionality for resetting the password
- Building the Chat page
- Sending and Receiving data to and from the database, backend and frontend

A lot of our challenges stemmed from our limited knowledge and expertise in developing and deploying a full stack application from scratch. While we had online resources, a lot of times we hit roadblocks where we couldn't resolve the error and had to spend hours or even days troubleshooting.

However, from these challenges, we learnt a great deal. One of them being recognizing our own shortcomings with managing time and resources. Even though we have most of the features working, we did feel the time crunch towards the end, where remaining tasks were piling up and the deadline was drawing close. Also, we maybe bit off more than we could chew, adding one too many features in Priority 1 which could have maybe taken a back seat. We also had an extensive list of Non-Functional requirements which while are needed for a successful product, were not entirely implementable in a product with such a small timeline. Also, we maybe didn't prioritize all the tasks quite as well as we should have, since some of the key functionality and unique features were the last one to be implemented.

From the lessons we learnt, here are a couple of things we would do better next time-

- Better organize the work and prioritize it so that the key functionality are implemented first
- Not use CSS framework like Reactstrap or react-bootstrap for our frontend. That would have given our application a much more clean look, and would have helped make it responsive.
- Use better coding practices to organize our code base, and use more consistent formatting and comments. Our current code base, while organized, is not the cleanest or uses the most

modular functions. We can definitely update our codebase to make it more understandable and efficient.

- Use our product management tool more frequently and effectively to keep track of our tasks.

In a nutshell, I see this project as a great learning opportunity where not only every single team member learned and grew as a Software Engineer but also learned to effectively work with a team. We had clear communication lines and active participation which helped us troubleshoot our errors and ensure timely delivery of all our milestones and the final product.

## Email

CSC 648 FA20 Team 02 DoReMeet Post Analysis

Nimiksha Mahajan  
Thu 12/17/2020 11:48 AM  
To: Meet Patel; Luke Chang-Knezevich; Mike Bolanos; Vincent K. Tse; Jungsun Eoh

Hey Team,

Please review the Post Analysis and take a moment to reflect on all the great work you've done this semester.

The design and development of DoReMeet has been a fun, yet challenging one. From coining this idea almost 3 months ago, we have come a long way in bringing it to fruition where our product is (almost) ready and functioning as we wanted. However there have encountered our fair share of challenges throughout this process. A few of our main challenges were:

- EC2 instance error - the first instance we created got corrupted, which we assume was because we tried to set up Elastic IP. We couldn't SSH into it. Also, it started exceeding the free tier limit and charged us, when it wasn't even running.
- Setting Up the EC2 instance - We faced some major issues setting up our EC2 instance successfully, and deploying the apache 2 server on the instance. We were facing some HTAccess issues, which prevented us from linking/redirecting our pages.
- Issues connecting backend and frontend on the server, due to permissions and virtual host settings
- Issues connecting to the database
- CORS Error
- Error tracking sessions when the user logs into the website, which caused the other functionality to not work as required
- Errors with building functionality for resetting the password
- Building the Chat page
- Sending and Receiving data to and from the database, backend and frontend

A lot of our challenges stemmed from our limited knowledge and expertise in developing and deploying a full stack application from scratch. While we had online resources, a lot of times we hit roadblocks where we couldn't resolve the error and had to spend hours or even days troubleshooting.

However, from these challenges, we learnt a great deal. One of them being recognizing our own shortcomings with managing time and resources. Even though we have most of the features working, we did feel the time crunch towards the end, where remaining tasks were piling up and the deadline was drawing close. Also, we maybe bit off more than we could chew, adding one too many features in Priority 1 which could have maybe taken a back seat. We also had an extensive list of Non-Functional requirements which while are needed for a successful product, were not entirely implementable in a product with such a small timeline. Also, we maybe didn't prioritize all the tasks quite as well as we should have, since some of the key functionality and unique features were the last one to be implemented.

From the lessons we learnt, here are a couple of things we would do better next time-

- Better organize the work and prioritize it so that the key functionality are implemented first
- Not use CSS framework like Reactstrap or react-bootstrap for our frontend. That would have given our application a much more clean look, and would have helped make it responsive.
- Use better coding practices to organize our code base, and use more consistent formatting and comments. Our current code base, while organized, is not the cleanest or uses the most modular functions. We can definitely update our codebase to make it more understandable and efficient.
- Use our product management tool more frequently and effectively to keep track of our tasks.

In a nutshell, I see this project as a great learning opportunity where not only every single team member learned and grew as a Software Engineer but also learned to effectively work with a team. We had clear communication lines and active participation which helped us troubleshoot our errors and ensure timely delivery of all our milestones and the final product.

Really good work team!

Thank you,  
Nimiksha

## CSC 648 FA20 Team 02 DoReMeet Post Analysis

LC

Luke Chang-Knezevich

Thu 12/17/2020 12:09 PM

To: Nimiksha Mahajan



Hi Team,

This semester flew by so fast! It feels like it wasn't long ago that we were just tossing ideas around for what our project could potentially be. I think that this experience was very fruitful for all of us and I'm grateful for all the work everyone put in. Even though we didn't get to finish everything that we originally planned I'm still proud of the product that we created.

I wish you all the best of luck in your future endeavors!

Regards,

Luke Chang-Knezevich

...

[Reply](#) | [Forward](#)

MP

Meet Patel

Thu 12/17/2020 12:05 PM

To: Luke Chang-Knezevich; Mike Bolanos; Vincent K. Tse; Jungsun Eoh; Nimiksha Mahajan



Hello everyone, it was fun working with you as a team, meanwhile we did encounter many issues and we were able to resolve it, yet there are few issue where we needed more time to resolve it, but I truly appreciate the work we have done as a team. Hope you guys have a good winter break!

Thank you,  
Meet Patel

## CSC 648 FA20 Team 02 DoReMeet Post Analysis

VT

Vincent K. Tse

Thu 12/17/2020 1:26 PM

To: Nimiksha Mahajan; Meet Patel; Luke Chang-Knezevich; Mike Bolanos; Jungsun Eoh



This project was definitely a challenge and a great learning experience with all the new concepts introduced. Most issues we had were fixed and we were so close to completing a few missing requirements.

...

[Reply](#) | [Reply all](#) | [Forward](#)

## CSC 648 FA20 Team 02 DoReMeet Post Analysis



MB

Mike Bolanos

Thu 12/17/2020 1:41 PM

To: Nimiksha Mahajan

Wow, reviewing all the setbacks we had had is painful. Even so, this has been a great and unique experience to learn and improve from. This was definitely fun. Who knows how much more we could have completed given the knowledge we have now. Thanks, everyone.

...

Thank you for your kind words.

Thanks for the feedback!

I totally agree!

! Are the suggestions above helpful?  Yes  No

Reply

Forward



JE

Jungsun Eoh

Thu 12/17/2020 1:38 PM

To: Nimiksha Mahajan

Cc: Meet Patel; Luke Chang-Knezevich; Mike Bolanos; Vincent K. Tse

The project was challenging, but great experience on everything, especially how to plan and excel actual product in real world. Wish we had one more week, but I guess this is how we learn how important to plan ahead before code. Thank you all for strong leadership and awesome team spirits, and thank you everyone who got my back every time I got error. Hope you all have fun and safe holiday!

Best,  
Jungsun Eoh.

