

CS 1530 g06 Requirements Spec

Software Requirements Spec

1. Product Overview and Summary

(From Section 1.0 - Scope - of the [software plan](#), but more detailed)

Quartz is a social media platform that enables users to connect and engage with one other on similar interests and hobbies. It is a site where users can create text posts and links on either (1) their personal page, or (2) a community page based on a specific topic (e.g. hockey, technology, music, etc). Posts on community pages can be voted up or down (through likes or dislikes), to determine how high it appears in the feeds of people who follow that community (similar to a web forum like Reddit). Users have the ability to share their opinions, thoughts and simply just catch up with old friends along with new ones. The sole purpose of this platform is to bring people together. Users will be able to create personal accounts that each have a user feed that keeps a record of all their previous commits. Users will also have the ability to assemble into communities based upon similar interests and engage with others through the use of a like/dislike system. Users can simply search for topics that they wish to engage upon and join communities where posts are visible to the entire community, accomplishing our goal of bringing people together. Users will also have the opportunity to share links within their posts for other users & communities to access. Although the upsides are tremendous, we account for the limitations of public posts as they may pose a certain level of threat to privacy. Private message functionality from user-to-user is still seen as a limitation to the general time length of implementation. Commenting on user posts has a certain degree of limitation as it doesn't allow users to properly engage with posts as the simple like/dislike system is quite rudimentary. Users will be able to respond through creation of new posts with inclusion of links to older posts. Media will not be uploaded directly unless through the use of links. Overall, Quartz has tremendous upside and will accomplish our goal of bringing people together.

Functions

Account Creation

- a. Users are able to register and create a secure account on the website (Devise API) with a unique name and password.

Account Pages

- b. Each user will have a page linked to their account. This page displays the user's posts and public information (name, bio, links).

Feed Pages

- c. This is the user home page after sign-in where all posts from people and communities followed will appear. This will implement the Recommended Content (6) function below.

Text Posts

- d. Users will be able to upload text which will be stored in our database (postgresql). These posts will appear on their profile pages and any community pages associated with the content (by hashtags).

Likes/Dislikes

- e. Users will be able to like/dislike a post.

CS 1530 g06 Requirements Spec

- f. The post will move up or down in the feed based on how many likes/dislikes it accumulates over a certain period of time

Recommended Content

- g. Each user will have a recommended content page based on their likes/dislikes
- h. Recommended content will be similar to tags the user has already liked/posted

Content Driven Communities

- i. Communities are pages specifically made for certain content. This will appear like a user account page, but it is public and anyone is able to post to the page.
- j. Community pages receive their own unique hashtags, which when someone posts using the hashtag, the post is automatically displayed on the community page.
- k. Anyone is able to create a community, and the community creator is considered its "owner"
- l. The owner is able to edit the pages bio and descriptions

Search

- m. The user has the ability to search for user/community pages, topics, and other posts.
- n. A feed is produced of matching users, pages, and posts.

Performance

- Every user will be able to post a maximum of 65536 (2^{16}) total posts to their page or to a community page
- Posts are limited to 10,000 characters
- Usernames can be up to 20 characters long

Limitations

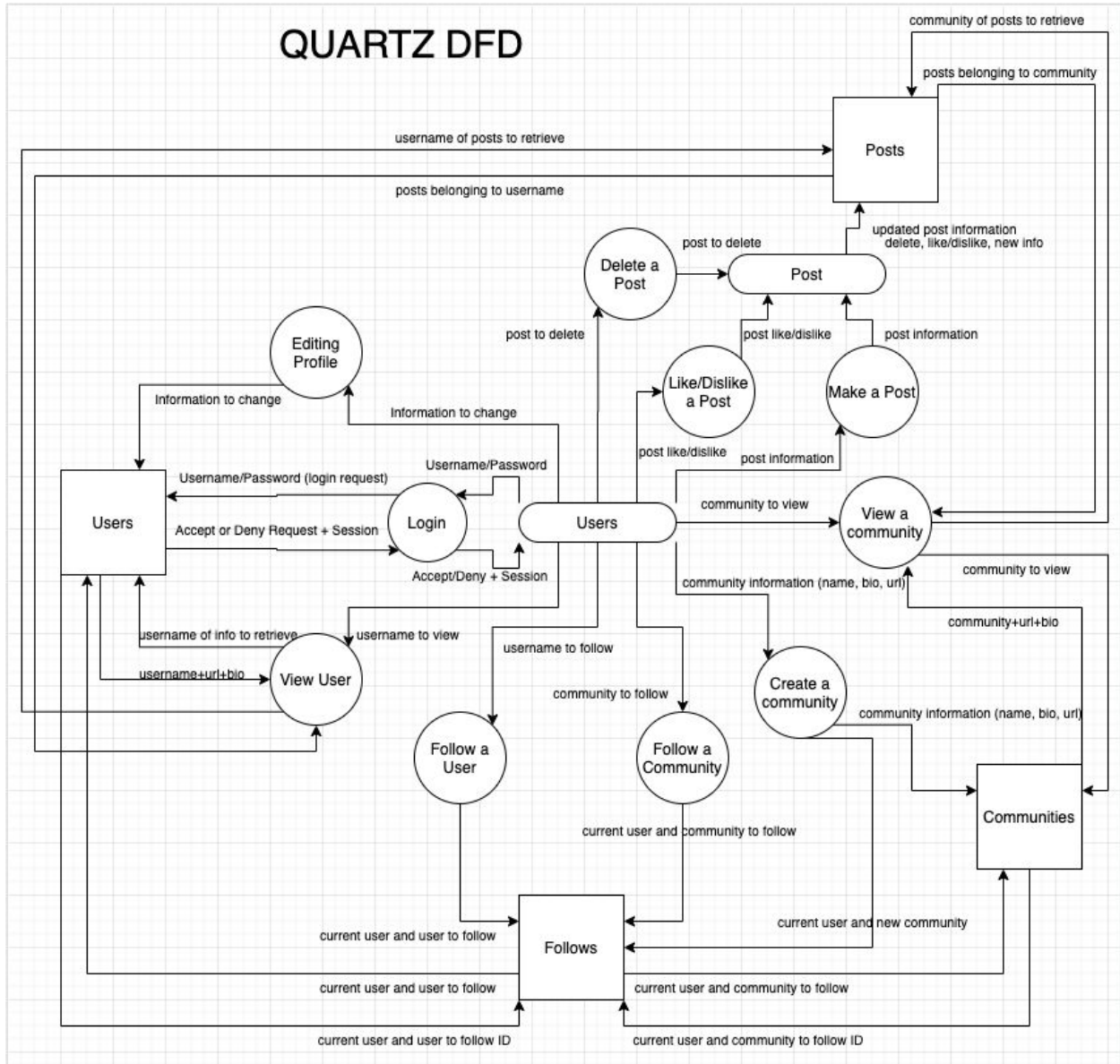
- Media like photos/audio/video can not be uploaded directly. However, links can be posted to this type of content hosted elsewhere
- There is no option to privately message other users
- All posts are public, there is no way to hide your page from certain people
- Users can not leave comments on posts. However, it is possible to create a new post in response to an existing post by linking to the original post

CS 1530 g06 Requirements Spec

2. Information Description

2.1. User interface ([A Preliminary User manual](#)) (See below)

2.2. High level data flow diagram



CS 1530 g06 Requirements Spec

2.3. Data structure (or object) representation

(what [data elements](#) are in the databases or objects?)

- users(id, email, encrypted_password, reset_password_token, reset_password_sent_at, remember_created_at, created_at, updated_at)
- communities(id, name, hashtag, description, created_at, updated_at, users_id)
- posts(id, post, hashtag, users_id, created_at, updated_at)
- follows(id, users_id, users_id, communities_id)

2.4. Data elements (or objects) dictionary

(compile all data elements into a dictionary)

- id: users, communities, posts, follows
- email: users
- encrypted_password: users
- reset_password_token: users
- reset_password_sent_at: users
- remember_created_at: users
- created_at: users, communities, posts
- updated_at: users, communities, posts
- name: communities
- hashtag: communities, posts
- description: communities
- users_id: communities, posts, follows
- post: posts
- communities_id: follows

CS 1530 g06 Requirements Spec

3. Functional Description

Each function is listed with:

- 3.1. Functions (you should specify functions using [IC cards](#))
- 3.2 Processing narrative
- 3.3 Design constraints
- 3.4. Diagrams

Objects:

IC Card

IC Name: Community

Description: A page centered around a certain topic, which all users can post to

Interaction Pattern:



Quiet State

My Task: Serve as a hub for people to make posts to. If a user makes a post with the community's hashtag, it is posted to the community

Time Critical Condition: NA

Name of Other IC: NA

Message to Other IC: NA

Other IC's Task: NA

Card 1 of 1 (If necessary please use several IC cards to describe an IC)

IC Card

IC Name: Post

Description: A text post, associated with a particular user

Interaction Pattern:



Quiet State

My Task: Stores text which was posted by a particular user.

Time Critical Condition: NA

Name of Other IC: NA


Message to Other IC: NA

Other IC's Task: NA

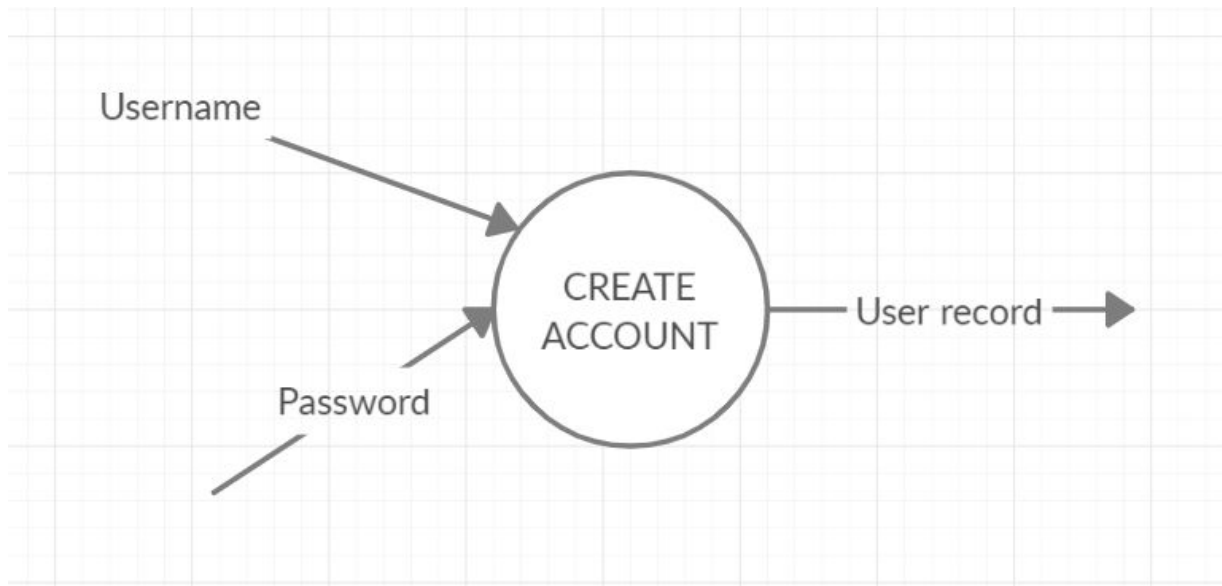
Card 1 of 1 (If necessary please use several IC cards to describe an IC)

CS 1530 g06 Requirements Spec

createAccount

IC Card	IC Name: createAccount
Description: Creates a new user account	
Interaction Pattern:	
	
By Myself no Interacton	
My Task: Allows a user to create a User account and page for themselves. Given a username and password, the account is created, along with a corresponding user page	
Time Critical Condition: NA	
Name of Other IC: NA	
Message to Other IC: NA	
Other IC's Task: NA	
Card 1 of 1 (If necessary please use several IC cards to describe an IC)	

- Allows a user to create a User account and page for themselves. The user enters a username and password, the account is created, along with a corresponding user page. The user page will display username, description, and posts from that user.
- Each username can only be used once.



CS 1530 g06 Requirements Spec

readHashtags

IC Card

IC Name: readHashtags

Description: Reads the hashtags of a post

Interaction Pattern:



By Myself with Interaction

My Task: Given a post, this function will read the hashtags of that post (words preceded by '#'). If a community exists for a hashtag, the post should be posted to that community page. This function will o

Time Critical Condition: NA

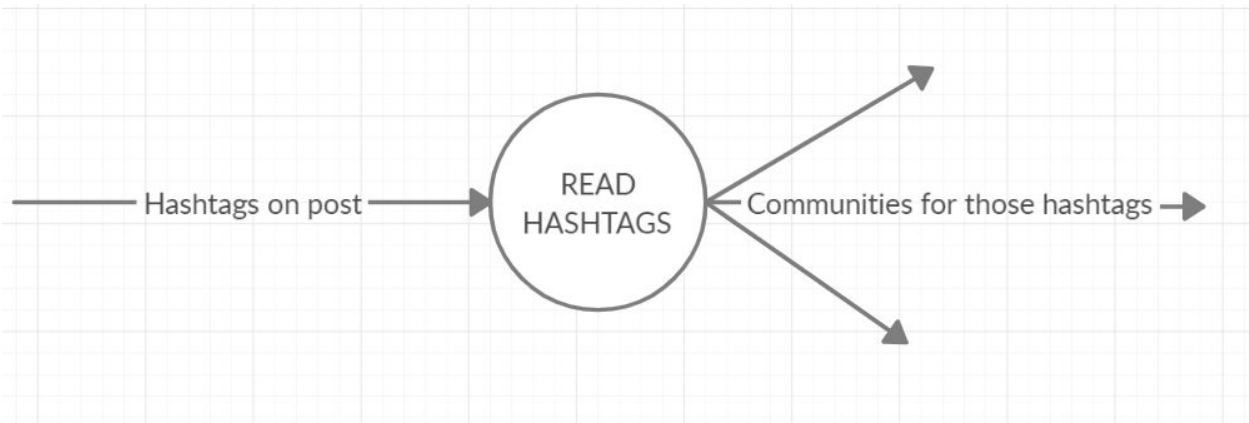
Name of Other IC: Post

Message to Other IC: Here are the communities this post belongs to

Other IC's Task: Give me the contents of the post, so I can read the hashtags

Card 1 of 1 (If necessary please use several IC cards to describe an IC)

- Given a post, this function will read the hashtags of that post (words preceded by '#'). If a community exists for a hashtag, the post should be posted to the feed for that community page.
- This function will only read hashtags for communities that already exist. If the community does not exist, nothing will happen.



CS 1530 g06 Requirements Spec

makePost

IC Card

IC Name: User

Description: makePost - create a new post

Interaction Pattern:



Mixed

My Task: A user can make a post to their user page. This function makes a new entry in the Post database.

Time Critical Condition: NA

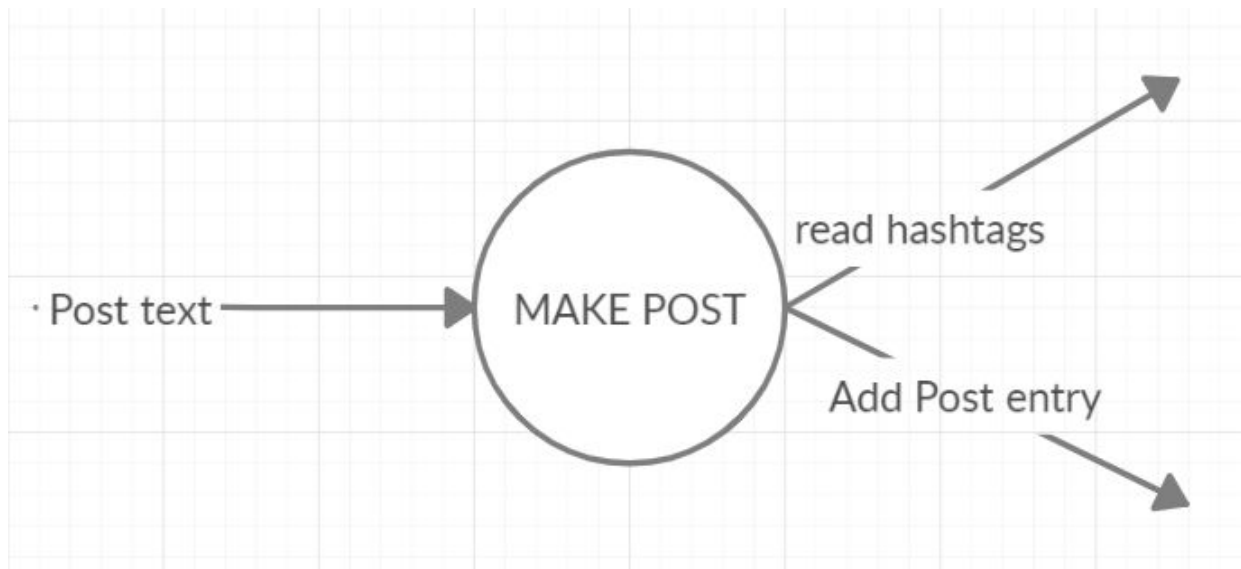
Name of Other IC: readHashtags

Message to Other IC: Heres a post, read the hashtags of this post

Other IC's Task: Read the hashtags and associate the post with the appropriate communities

Card 1 of 1 (If necessary please use several IC cards to describe an IC)

- A user can make a post to their user page. The user types in the text of their post into a text form, or types in a link to an image. This function also calls the 'read hashtags' function to determine which community pages the post should post to. This function makes a new entry in the 'Post' database.
- The user can only enter 1000 characters in a single post (including hashtags), and hashtags can be no more than 15 characters



CS 1530 g06 Requirements Spec

createCommunity

IC Card

IC Name: User

Description: createCommunity - create a new community

Interaction Pattern:



By Myself no Interacton

My Task: A user can create a community by specifying a name, hashtag, and initial description (the description can change later). The community must have a unique hashtag, which has not been created before. Th

Time Critical Condition: NA

Name of Other IC: NA

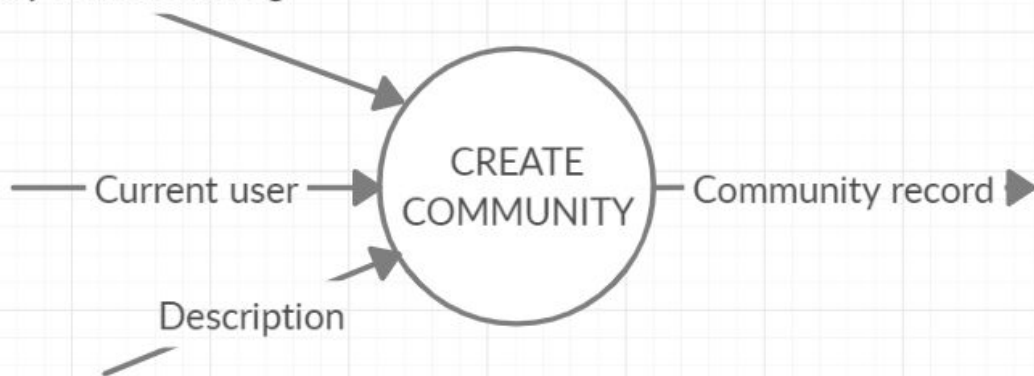
Message to Other IC: NA

Other IC's Task: NA

Card 1 of 1 (If necessary please use several IC cards to describe an IC)

- A user can create a community by entering a name, hashtag, and initial description (the description can change later). The community must have a unique hashtag, which has not been created before. The creating user will become the owner of that community. The community will be added to the Community database.
- There are no restrictions on creating communities, so one user is able to create and own as many community hashtag names as they can think of. This may prevent other users from creating communities which they want to own

Community name/hashtag




CS 1530 g06 Requirements Spec

deletePost

IC Card IC Name: User

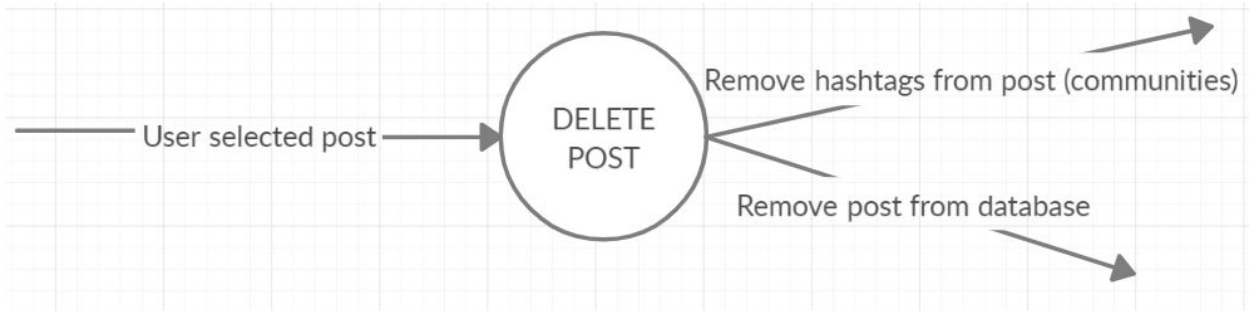
Description: deletePost - delete a post
Interaction Pattern:



By Myself no Interacton


My Task: A user deletes a post that they have created. The post is removed from the database.
Time Critical Condition: NA
Name of Other IC: NA
Message to Other IC: NA
Other IC's Task: NA
Card 1 of 1 (If necessary please use several IC cards to describe an IC)

- A user deletes a post that they have created. If the user is logged into their account, they will be able to press a button on each of their post which asks them to confirm deletion of that post. If the user confirms, the post is removed from the database.
- A deleted post can not be recovered after being deleted.

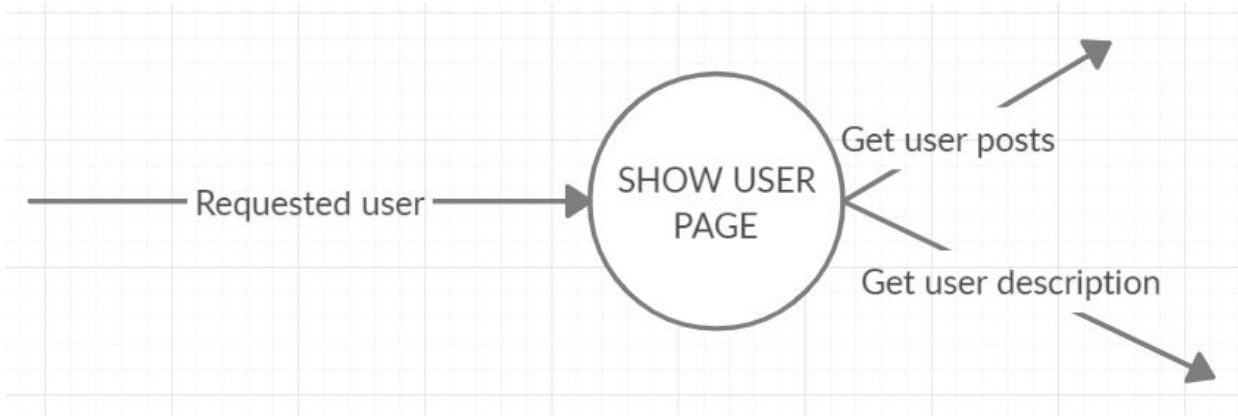


CS 1530 g06 Requirements Spec

showUserPage

IC Card	IC Name: showUserPage
Description: displays the page of a user	
Interaction Pattern:	
	
By Others no Interaction	
My Task: Ask the user to display their page, which contains their name, description, and a list of all posts that they have made, in reverse chronological order.	
Time Critical Condition: NA	
Name of Other IC: User	
Message to Other IC: Display the user page for this user	
Other IC's Task: Display the user page for that user	
Card 1 of 1 (If necessary please use several IC cards to describe an IC)	

- Anyone with an account can view a user's page, which is accessible from a link to that user's page. The user's page displays their name, description, and a list of all posts that they have made, in reverse chronological order.
- This function will only show user posts in reverse chronological order (for now). It will NOT be able to sort the posts in any other way, for example by the number of likes. Also, only 20 posts can be loaded at a time, more can be loaded by advancing to the next page.



CS 1530 g06 Requirements Spec

likePost/dislikePost

IC Card

IC Name: User

Description: likePost or dislikePost - add a like or a dislike to a post

Interaction Pattern:



By Myself with Interaction

My Task: Increases the number of likes/dislikes on a post by 1. Each user can only like a post once, so this function will do nothing if they have already liked or disliked a post.

Time Critical Condition: NA

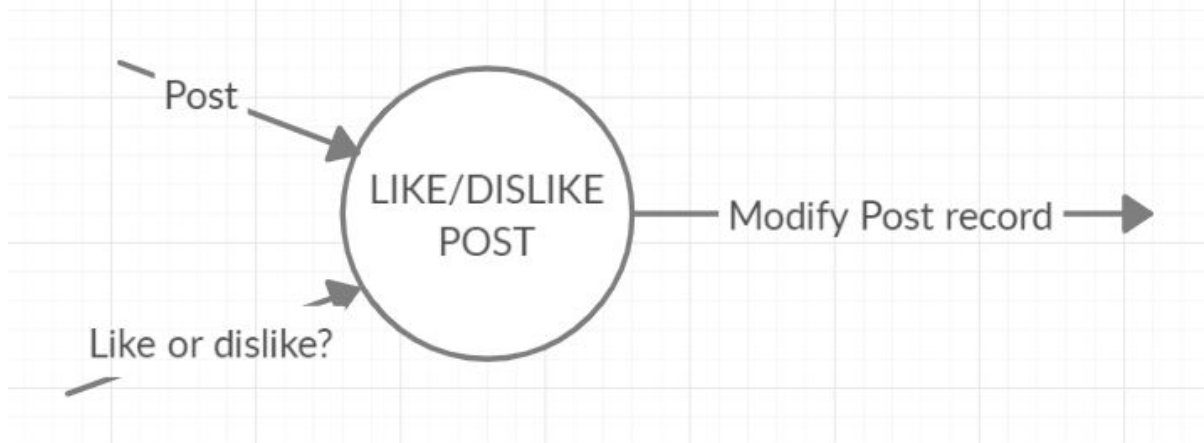
Name of Other IC: Post

Message to Other IC: X User liked or disliked this post

Other IC's Task: Increases the number of likes or dislikes on the post

Card 1 of 1 (If necessary please use several IC cards to describe an IC)

- Increases the number of likes/dislikes on a post by 1. The user can like or dislike a post by clicking on the corresponding button on the post. Each user can only like a post once, so this function will do nothing if they have already liked/disliked a post.
- Liking or disliking a post will not make it show up any higher or lower in any feed.



CS 1530 g06 Requirements Spec

showCommunityPage

IC Card

IC Name: showCommunityPage

Description: Displays the page of a community

Interaction Pattern:



By Others no Interaction

My Task: Displays the page for a given community. The community page shows information including the creation date, description, and owner. It also shows a list of recent posts in that community, and the user

Time Critical Condition: NA

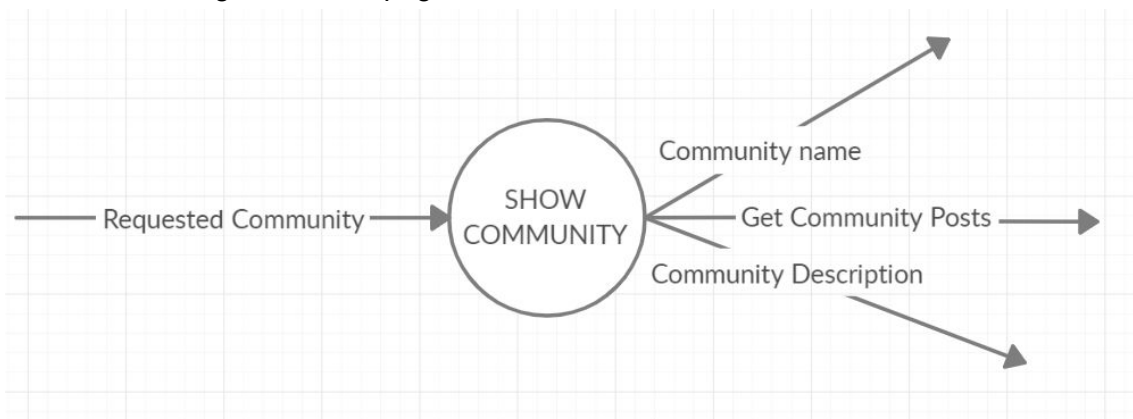
Name of Other IC: Community

Message to Other IC: Display your page

Other IC's Task: Use community information to display the page

Card 1 of 1 (If necessary please use several IC cards to describe an IC)

- Displays the page for a given community. The community page shows information including the creation date, description, and owner. It also shows a list of recent posts in that community, and the user who made each post. Posts are displayed in reverse chronological order.
- Similar to the user pages, posts can only be sorted by reverse chronological order (currently). Also, only 20 posts can be loaded at a time, more can be loaded by advancing to the next page.



CS 1530 g06 Requirements Spec

showFeed

IC Card

IC Name: showFeed

Description: Show the feed of the user logged in
Interaction Pattern:



By Myself with Interaction

My Task: When a user goes to their home page, they will see a list of posts in their feed. These posts come from the posts of users that the user follows, and the posts of communities that the user follows. Th

Time Critical Condition: NA

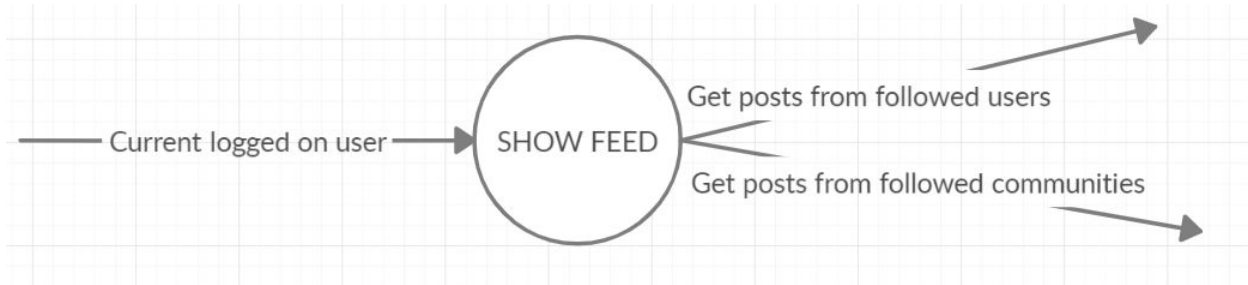
Name of Other IC: User

Message to Other IC: Send me a list of pages that the current User follows

Other IC's Task: Respond with a list of pages that the User follows


Card 1 of 1 (If necessary please use several IC cards to describe an IC)

- When a user goes to their home page, they will see a list of posts in their feed. These posts come from (1) the posts of users that the user follows, and (2) the posts of communities that the user follows. The posts are sorted in reverse chronological order.
- The posts in the feed can only be sorted in reverse chronological order. Also, only 20 posts can be loaded at a time, more can be loaded by advancing to the next page.

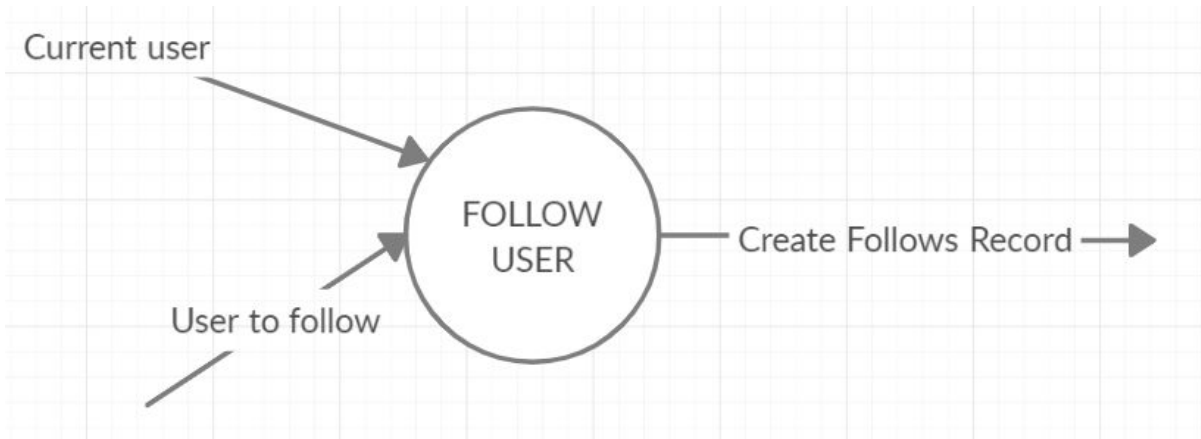


CS 1530 g06 Requirements Spec

followUser


IC Card	IC Name: User
Description: followUser Interaction Pattern:	
	
By Myself with Interaction	
My Task: Add another User to the current Users follow list	
Time Critical Condition: NA	
Name of Other IC: showFeed	
Message to Other IC: Add these users posts to the feed	
Other IC's Task: Add the new users posts to the feed	
Card 1 of 1 (If necessary please use several IC cards to describe an IC)	

- A user clicks on the "Follow" button on another user's page. That user will be added to the following users "following" list. This will cause posts by the followed user to appear in the feed of the following user.
- Following a user does not modify the feed of the followed user

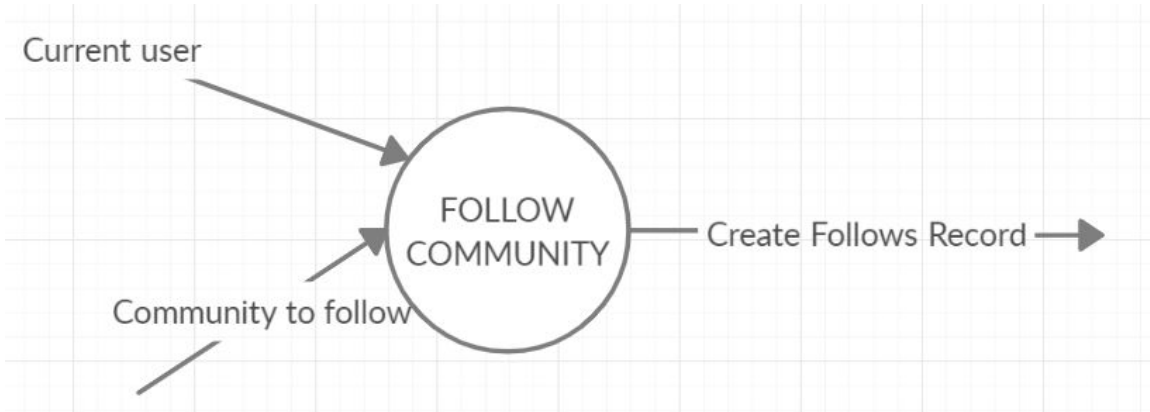


CS 1530 g06 Requirements Spec

followCommunity


IC Card	IC Name: User
Description: followCommunity	
Interaction Pattern:	
	
By Myself with Interaction	
My Task: A user chooses a community to follow. That community will be added to the following users following list. This will cause posts by the community user to appear in the feed of the following user.	
Time Critical Condition: NA	
Name of Other IC: showFeed	
Message to Other IC: Name of community to add	
Other IC's Task: Add posts from this community to the Users following list	
Card 1 of 1 (If necessary please use several IC cards to describe an IC)	

- A user clicks on the “Follow” button on a community page. That community will be added to the following users “following” list. This will cause posts by the community user to appear in the feed of the following user.
- Following a community does not give the user any additional privileges in that community.

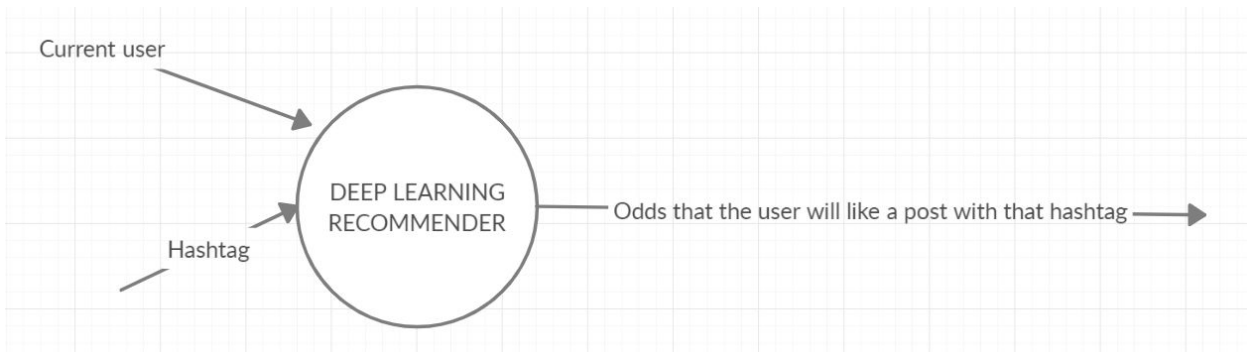


CS 1530 g06 Requirements Spec

Recommender

IC Card	IC Name: Recommender
Description: A deep-learning component to recommend hashtags to the user	
Interaction Pattern:	
	
Mixed	
My Task: The user will request and receive a list of 3 hashtags (communities) that they are likely to enjoy posts from. The recommendations come from a deep learning algorithm which is trained based on the lik	
Time Critical Condition: NA	
Name of Other IC: User	
Message to Other IC: Calculate the relative likelihood that a user will like a post of a given hashtag	
Other IC's Task: Train a deep learning model to predict the likelihood that a user will like a post with a specific hashtag, based on their like history	
Card 1 of 1 (If necessary please use several IC cards to describe an IC)	

- The user will request and receive a list of 3 hashtags (communities) that they are likely to enjoy posts from. The recommendations come from a deep learning algorithm which is trained based on the likelihood that a user will like a post for a particular hashtag.
- This function can only recommend posts based on their HASHTAGS, NOT based on the test content of the post



CS 1530 g06 Requirements Spec

4. Performance requirements

- Platform should allow at least one thousand concurrent users
- Platform should allow the creation of two hundred thousand users
- Platform should allow for the creation of posts in less than one second
- Platform should allow for the like/dislike functionality to process in less than one second
- Users can post a maximum of 65536 (2^{16}) total posts
- Posts are limited to 10,000 characters
- Usernames can be up to 20 characters long
- User Feed should update new posts in less than two seconds after user refreshes
- Follow user should be functional in less than one second
- Follow community should be functional in less than one second
- User Profile edits should be processed in less than one second

5. Exception conditions/exception handling

(For example, if a company has a crash job and suddenly needs to hire five hundred temp workers, how to deal with it?)

- More than 10,000 characters are needed
 - If there is an emergent situation where a user would need additional characters, and we find that the reasoning is justified, the user can reach out to the Quartz Team via email to make their case. In this situation, the extended post can be hard-coded onto the user's page or community by a member of the Quartz Team.
- User or Community needs to surpass performance limitation of 65536 posts per page
 - The Quartz Team will handle this on a case-by-case basis where the user or community would have to reach out themselves in order to increase the allotted capacity. If necessary, the Quartz team could either increase the allotted capacity of posts or implement a feature where older posts begin to delete as new posts come in.
- User needs support of photos/videos, private posts, or the ability to comment
 - The user can reach out to the Quartz Team and request any type of feature that they feel is lacking and we would be glad to let them know what we are working on. The integration of these features into Quartz are on the implementation wish-list and will be implemented given more time and resources. For the time being, they can simply post a link to their favorite photos and videos and reference other posts in order to imitate "commenting."

CS 1530 g06 Requirements Spec

6. Implementation Priorities

- Due to resource limitations such as having only five group members and one semester, we will first implement the core system supporting the following basic functions:
 - Secure user login and account creation
 - Create a Community
 - Make a post/Delete a post
 - Like/Dislike a post
 - User feed populated with user and community posts
 - Make user Profile with ability to edit
 - Follow a User
 - Follow a Community
 - Search for User/Community via search bar
- If there are additional resources, we will implement the following functions:
 - Post/content recommender component to recommend relevant posts to a user
 - Private posts
 - Photo/video upload
 - Commenting
- For the final product, we will implement all the basic functions plus some functions on the wishlist piece by piece. We will follow the agile incremental approach since each function is being implemented by iteration.

7. Foreseeable modifications and enhancements

- There are a number of foreseeable modifications that are only currently limited by the essence of time through the next few weeks.
 - The ability to comment on user's posts and provide feedback
 - Enhance the privacy scope of the posts to range from public to private
 - Inclusion of photo and video upload
 - Implement a direct messaging functionality from peer to peer
 - In addition to the like/dislike functionality, more emoticons can be used by users that are provided by most iPhones and Android devices
 - Create a potential marketplace to allow users to exchange goods for monetary value

CS 1530 g06 Requirements Spec

8. Acceptance criteria

Functional and performance tests

Documentation standards

(what tests will you do to accept this product? What documents will you deliver?)

- Basic functionality testing for:
 - Creating new accounts
 - Clicking on and viewing hashtags
 - Creating new posts
 - Creating and viewing communities
 - Show user feed
 - Follow functionality
 - Feed functionality
 - Deep learning algorithm
- Tests conducted using Ruby on Rails environmental tests
- Browser tests for:
 - Functionality of buttons
 - Functionality of account features
 - Theme and appearance
- Documents to be included:
 - Software plan
 - Requirements spec
 - User manual
 - Test plan
 - Test results
 - Design documents
 - Source code

CS 1530 g06 Requirements Spec

9. Sources of information

(This section includes documents from software vendors, outsourcing companies, and most importantly customers. For example the customer's company has a human resource document and our system is designed in compliance to that document. For the class project, you can cite the first kind of documents.) -- include version

- Milestones completed
 - Software plan
 - Requirements spec
 - Test plan
- Scope of the project is within the requirements set out on the class website
- Scope also within requirements outlined in the software document
- User manual, OOA/OOD document will also be submitted
- [Using Ruby on Rails API](#)

10. Revision history

(Each entry shows the date, short description and responsible person for each revision)

Sept 20

- Max: Added Functional Descriptions
- Gurmail: Added Product Overview and Summary, Schedule
- Jared: Added Preliminary User Manual
- Jake: User Manual and High Level DFD
- John: Determined cost

Oct 4

- Max: Detailed DFD
- John: Acceptance criteria and source of information
- Jake: Data Structure and Data Elements
- Jared: Exception handling and implementation priorities
- Gurmail: Foreseeable Modifications and Performance requirements

Oct 5

- Jake: Added screenshots to user manual

CS 1530 g06 Requirements Spec

User Manual

1. Product overview

(what this product is for and its main functions)

- a. Quartz is a social media platform where users create an account in order to produce and consume text-based content. The content that is produced can be shared with the user's followers or publicly on a relevant community page using the #hashtag feature. At the start, users will have the ability to create their own User Profile consisting of some personal information such as a unique bio as well as personal interests. There are two types of content pages that will be available to Quartz users; they are called personal and community pages. Additionally, the user will be able to perform several main functions within Quartz. These functions include but are not limited to making a private post for followers, making a public post to a community page, creating a community, following users and communities, and liking/disliking posts.

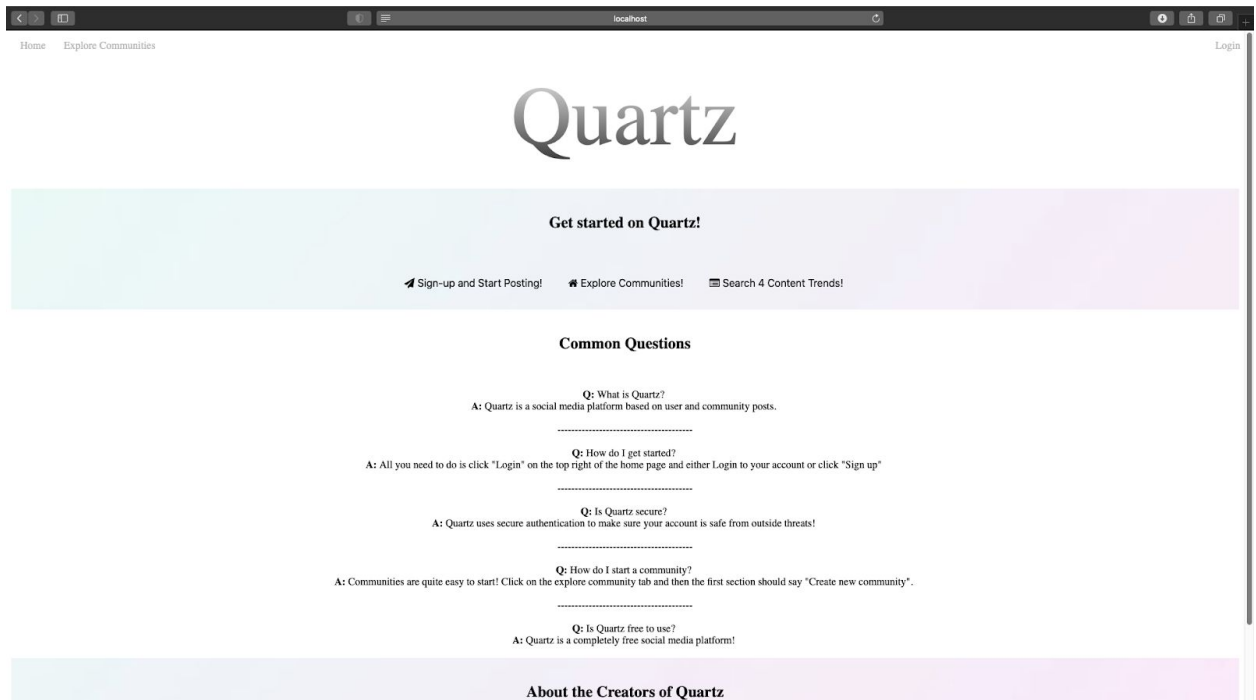
2. Getting started

(How can a user log in the system? What online help or other kind of help are provided? Sample runs.)

2.1. Log in

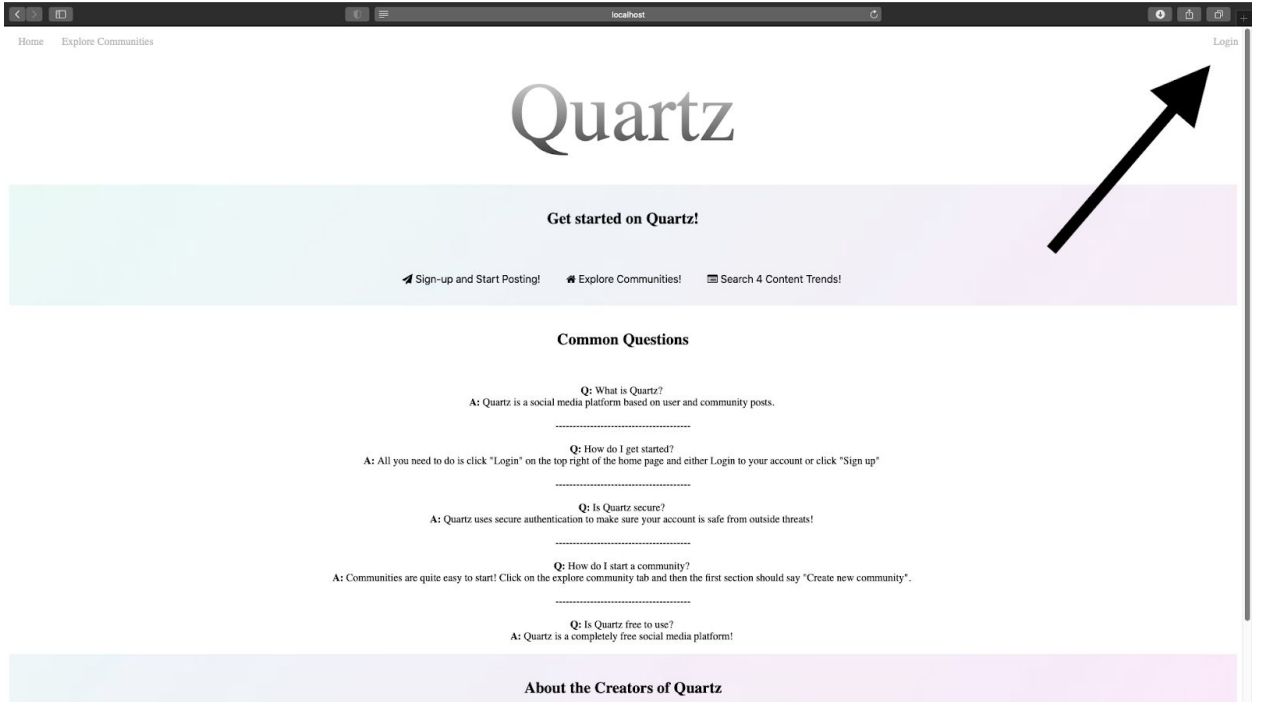
To log into the system, a user will:

1. Open the homepage of the website (root page '/')

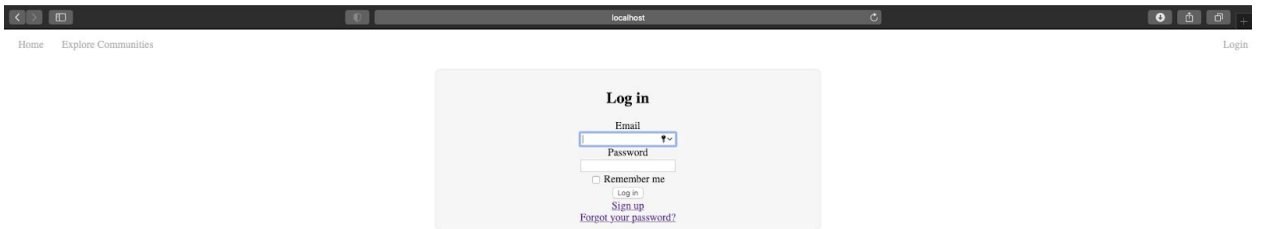


2. Move the mouse to the top right of the screen, hover over the navigation bar where it says "Login", and left click the mouse

CS 1530 g06 Requirements Spec

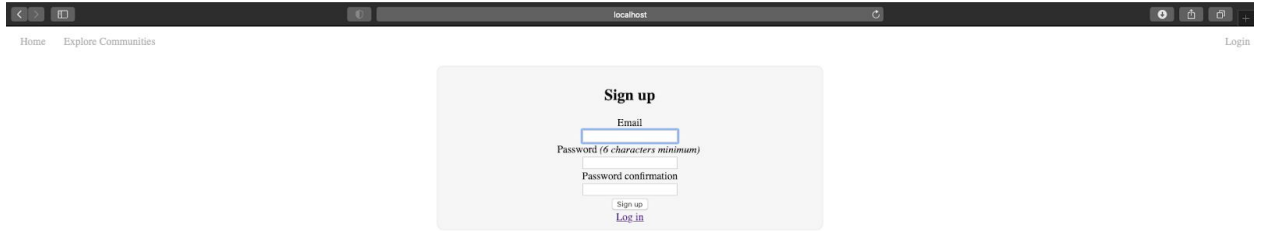


3. If an account is already made, type into the first box the email, second box the password, and hit login... Otherwise



- 3a. Click on the signup link below the login box
- 3b. Enter your email in the first box and password in the second and third
- 3c. Click the signup button below the boxes.

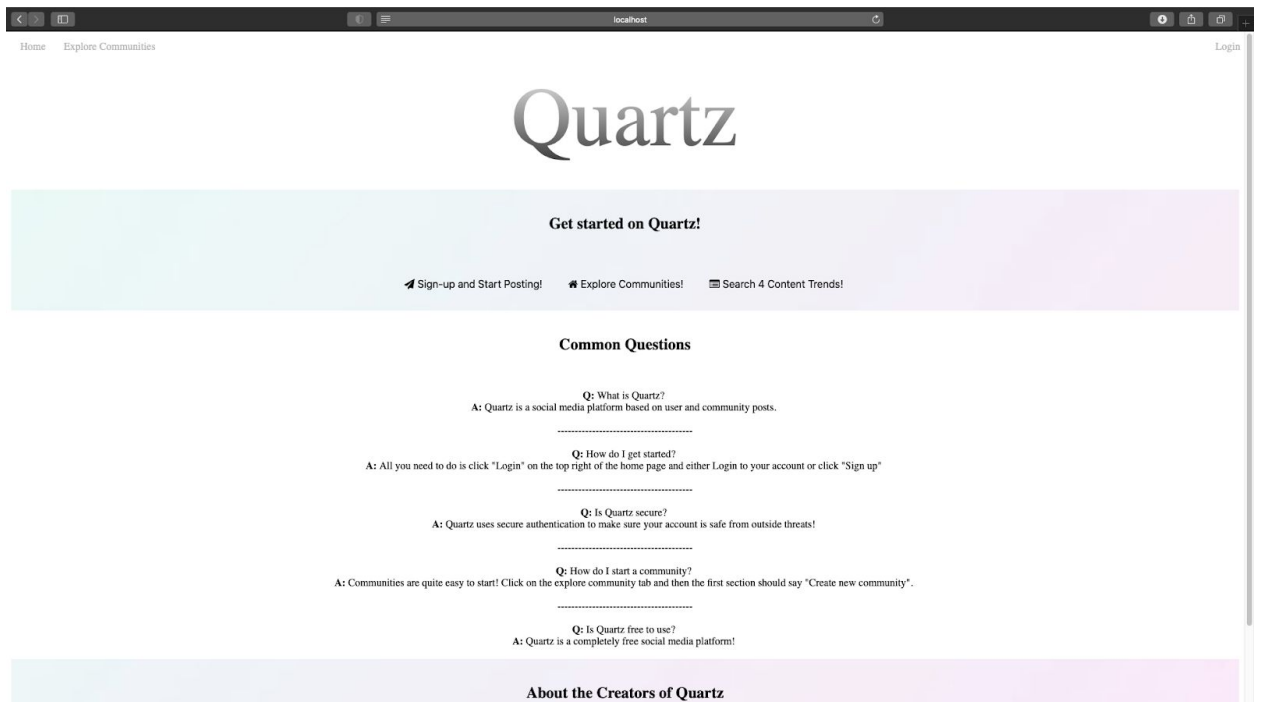
CS 1530 g06 Requirements Spec



2.2. Help mode

To get help on using the website, a user will:

1. Open the homepage of the website (root page '/')



2. Move the mouse to the top of the screen and hover over the navigation bar where it says "Help", and left click the mouse
3. The user will be redirected to the help page, where they can scroll through the user manual

CS 1530 g06 Requirements Spec

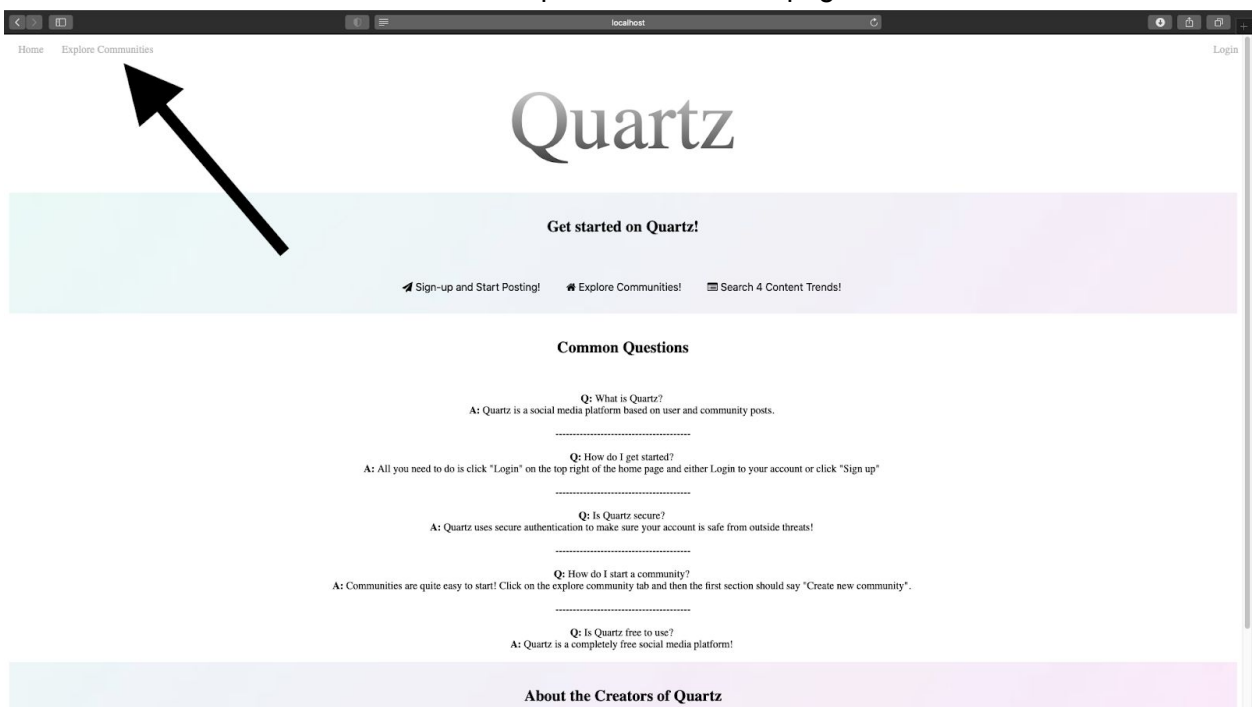
2.3. [Sample runs](#) and [example](#)

1. Making a Post

- a. To make a post, the user must first complete the login procedure outlined in section 2.1
- b. After logged in, the user will be looking at the home page (feed)
- c. The user will click “Profile” on the top right of the navigation bar (top right of page)
- d. The user will then click the button on their profile page labeled “Make a post”
- e. The user will then be displayed a text box, where they will be able to type anything within a 1000 character limit
- f. The user will click the button at the bottom of the text area called “Post” and the post will be sent live

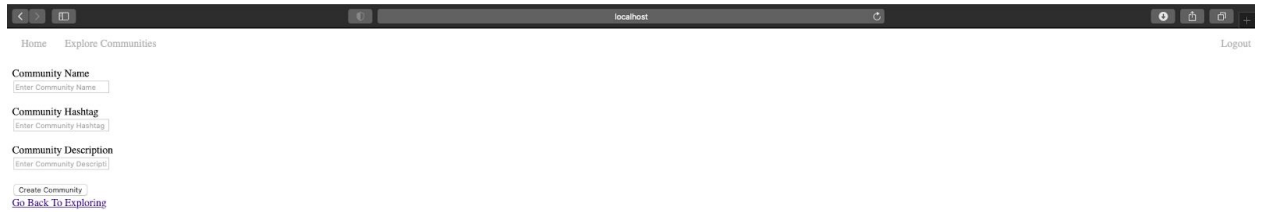
2. Creating a Community

- a. To create a community the user must first complete the login procedure outlined in section 2.1
- b. After logged in, the user will be looking at the home page (feed)
- c. The user will click the “explore communities” option in the navigation bar and be redirected to the explore communities page



- d. At the top of the page will be a button labeled “Create Community”. The user will click on this button
- e. The user will be redirected to a page that asks for info about the community (form). The user will input the community name in the first text box, community hashtag in the second, and community description in the third

CS 1530 g06 Requirements Spec



- f. The user will then Click the submit button below the form, and the community will be created
3. Viewing Feed
 - a. To view the feed the user must first complete the login procedure outlined in section 2.1
 - b. After logged in, the user will be looking at the home page (feed)
 - c. The user will be able to scroll through this home page, looking at all the posts generated on their feed
4. Viewing Profile
 - a. To view a profile, the user must first complete the login procedure outlined in section 2.1
 - b. After logged in, the user will be looking at the home page (feed)
 - c. To view...
 - i. their own profile, the user will move their mouse to the top right section of the navigation bar and left click on "Profile"
 1. They will be redirected to their profile page to view it
 - ii. Another user's profile, they can scroll through the current page (feed page) and left click on a user name (top left of post)
 1. They will be redirected to that user's profile page
5. Following a Community
 - a. To follow a community, the user must first complete the login procedure outlined in section 2.1
 - b. After logged in, the user will be looking at the home page (feed)
 - c. The user will then move the mouse to the top left of the page, and click on the "explore communities" navigation bar tab
 - d. The user will then be redirected to the explore communities page

CS 1530 g06 Requirements Spec

- e. On this page, the user will be able to either
 - i. Scroll through the page to find the community they want, and then click on that community
 - ii. Use the search bar to search for the community they are looking for
- f. When viewing the community page, the user will then move their mouse to the right top section of the screen and click on "Follow"
- 6. Following User
 - a. To follow a user, the current user must first complete the login procedure outlined in section 2.1
 - b. After logged in, the user will be looking at the home page (feed)
 - c. First, they will view another user's profile, so they can scroll through the current page (feed page) and left click on a user name (top left of post)
 - d. They will then be redirected to that user's profile page
 - e. When viewing the user's page, the current user will then move their mouse to the right top section of the screen and click on "Follow"
- 7. Delete a Post
 - a. To delete a post, the user must first complete the login procedure outlined in section 2.1
 - b. After logged in, the user will be looking at the home page (feed)
 - c. The user will click "Profile" on the top right of the navigation bar (top right of page)
 - d. From the profile page, the user will look at the post in the middle of their page they would like to delete
 - e. Once the post is found, the user will navigate the mouse to the right of the post, and click the button that says "delete"
- 8. Like or Dislike a Post
 - a. To like or dislike a post, the user must first complete the login procedure outlined in section 2.1
 - b. After logged in, the user will be looking at the home page (feed)
 - c. In their feed will be a collection of posts
 - d. The user will hover over one of the posts he/she intends to like/dislike
 - e. The user will then click either the "like" or "dislike" button at the bottom of the post
- 9. Editing your profile
 - a. To edit a profile, the user must first complete the login procedure outlined in section 2.1
 - b. After logged in, the user will be looking at the home page (feed)
 - c. The user will click "Profile" on the top right of the navigation bar (top right of page)
 - d. At the top right of the profile page, the user will then click on the "edit profile" button
 - e. The user will be redirected to an edit profile page, where they can type in a new name to the first text box, and a bio to the second text box

CS 1530 g06 Requirements Spec

3. Modes of operation

(How does a user operate the system? Does the user enter different types of commands? What are the commands? Does the user enter into a dialogue with the system? The sample dialogue can be incorporated into the sample runs discussed above in Section 2.)

Commands/dialogues/reports

- a. The modes of operation within Quartz that allow the user to interact with the site consist of both buttons and links. The user will navigate the system by clicking buttons/links that will take them to other pages.
- b. These modes of operation can be seen in the sample run in section 2.3. However, an example of these interactions within the home page is given below.
- c. Root/Home Page: Upon opening the root page,
 - i. Log in: There will be a login button at the top-right of the page where the user can click to either login or create an account. This will redirect to the login page.
 - ii. View User: Once logged in, the home page also acts as the “feed” page. Each post will consist of a user’s username which can be clicked to navigate to that user’s page
 - iii. Like/Dislike: Each post on the feed has a like/dislike button which can be clicked to like or dislike the post.
 - iv. Profile: There will be a profile button at the top-right of the page that when clicked on will take the user to their profile.
 - v. Explore Communities: There will be an “explore communities” button at the top left of the page that when clicked will redirect to the explore communities page.