

Gator Trader - Milestone 2

Edit History

Oct 12, 2018 -

Oct 15, 2018 - Ryan added info to data definitions (not sure if i need to fill this out?)

1. Data Definitions V2

Gator Trader - The name of our product

User - Any individual using our website.

- **Non-registered User** - Permission to search the site, open / read description and pictures on posted items. Allowed to click on “create a posting” but will be prompted after trying to submit post to either sign in or create an account.
- **Registered User** - Permission to search the site, open / read / create a posting and pictures. Allowed to create a personal profile. Should be a college student but students outside of SFSU are encouraged.
- **Administrator** - Granted permission to accept and remove posts. Ability to ban a user if they feel it's appropriate. Inherits permissions of registered users.

User Profile - Allows a user to display information describing themselves, such as which college they are attending, where they are located, etc. A picture of themselves can be posted if they wish to do so.

Administrator Page - Interface where an administrator has the ability to perform their roles.

Registration Page - A form which requires an individual to input certain information regarding their identity. This can include their name, location, and even a short description about themselves.

User Agreement - A *not real* legal document between the website and registered user. In order to use the website's services, the user must "agree" with the website's terms of conditions and policies before becoming a registered user.

CAPTCHA - A security check to confirm if a user is a real-person or a computer bot. The security check can consist of re-typing a displayed word to clicking the requested images.

Bookmark - Registered users are able to save and favorite posted items they may be potentially interested in. This provides for quicker access as well as the ability to store all interested items in one location.

Search Function - Allows the user to search for an item by directly typing in the item they are interested in. This allows for quicker access versus searching through each post. A search field will be available on every webpage.

Post - Actual listing of a product posted by a registered user which contains information regarding the item they would like to sell or purchase. This can include a picture of the item followed with a description. The meetup location and contact information can also be found here.

- **Category** - Sorts each post by the specific item posted to create an easier method to find an item. A user can then choose a category from a drop-down menu specified by a keyword.
- **Title** - Name of the product listed.
- **Description** - Thorough explanation describing the product listed. This can include the age of the product, condition, manufacturer, etc.
- **Image** - A minimum of 1 image clearly showing the specific item listed.
- **Post status** - Displays if a registered users post was accepted or declined by an administrator.
- **Seller** - The User who made this post.
- **Status** - The state of the post. Can either be Pending Approval, Approved, or Removed
- **Price** - The amount of money the user who made the post is asking for the product they are trying to sell / buy.

- **Meetup Location** - Location where the selling and buyer both agree on to conduct the transaction of buying or selling an item.
- **Contact Information** - A method in which a buyer or selling can contact the owner of a specific post.

Message - Communication system between two registered users on Gator Trader to discuss a certain post.

Notification - Message sent by Gator Trader to notify a registered user that they received a message as well as confirmation on whether their post got approved or declined.

Posting Section - A page which contains all of the current postings listed on Gator Trader.

Sort Function - Allows the user to order the postings from the most recently posted, most expensive to least expensive posts, and least expensive to most expensive posts.

2. Functional Requirements V2

1 - Required Functions

Non-Registered and/or Registered User

1.1.00 Non-Registered and registered users shall be able to search by category

1.1.01 Items can be divided under certain subsections. For example, a sofa/table would be listed under furniture.

1.1.02 We will have 14 categories. These categories consist of Furniture, Books, Computers, Phones, Video Games, Traditional Games, Food, Makeup, Bicycles, Skates and Skateboards, School Supplies, and Artwork.

1.2.00 Non-Registered and registered users shall be able to search by keyword

1.2.01 The search engine will use like words and exact words to match the search engine to its results. For example, “Java Programming Book” for sale could have several keywords attached to the search result - programming and book.

1.3.00 Non-Registered user shall register to contact seller

1.3.01 Non-Registered users will begin the registration process if they want to contact anyone on the site.

1.3.02 This is to prevent spammers, and to keep the registered-user accountable for their actions on their account.

1.4.00 Non-Registered user shall register to post

1.4.01 Non-Registered users will have to create an account before being able to post so their seller profile can be displayed.

1.4.01 They will fill out the required fields such as name, city (and major) as well as choosing a login and password.

Registered User

1.5.00 Registered user shall be able to login with their credentials after registering

1.5.01 Registered user will need to login with their login and password to be able to view their listings, messages, and any other notifications on their account.

1.6.00 Registered user shall have their own profile page

1.6.01 Registered users will have their own profile page where they can check their listings and transaction history.

1.6.02 Registered users will be able to check their messages with other users here.

1.7.00 Registered user shall be able to post listings

1.7.01 The registered user will fill out required information regarding their post such as name, price, category, condition, and description.

1.7.02 Registered user will be able to add tags to make their product more visible to the appropriate searches.

1.7.03 Registered user will not be required to upload a picture to post.

1.7.04 Registered user will be notified that it will take 24 hours before it becomes visible to the public.

Administrator

1.8.00 All products shall require administrator approval before posting

1.8.01 After a registered user has posted an item, it will be sent to a moderator for approval.

1.8.02 If the post is accepted, it will then be posted for public view.

1.8.03 If it is rejected, a notification/message from the administrator will be sent to the poster listing what requirements the post failed to meet.

1.9.00 Registered user shall be banned by the administrator for posting anything inappropriate

1.9.01 To increase the positive experience for our non-registered and registered users, we do not allow any profanity, nudity, etc. to be posted in messages or listings.

1.9.02 We don't allow registered users to use language that we consider to be defamatory, offensive, or that could be seen as abuse or harassment.

1.10.00 Registered user shall be banned by the administrator for any phishing or scamming attempts

1.10.01 An administrator may ban a registered user if there is any attempt or intention of scamming/phishing another registered user.

2 - Desired Functions

Non-Registered and/or Registered User

2.1.00 Non-Registered and registered users shall have access to contact information to a help desk/administrator

2.1.01 An email listed on a contact us page or the bottom of every page to help resolve any issues or questions that may come up.

Registered User

2.2.00 Registered user shall be able to delete their own listings

2.2.01 If the registered user have sold the product elsewhere or no longer wish to sell, they have the option to delete the listing on the site.

3 - Opportunistic

Non-Registered and/or Registered User

Registered User

3.0.00 Registered User shall be able to rate other registered users

3.0.01 Registered users can rate other registered users they have bought from.

3.0.02 Registered users can rate other registered users they have sold to.

3.0.03 Registered users will have their rating on their profile page

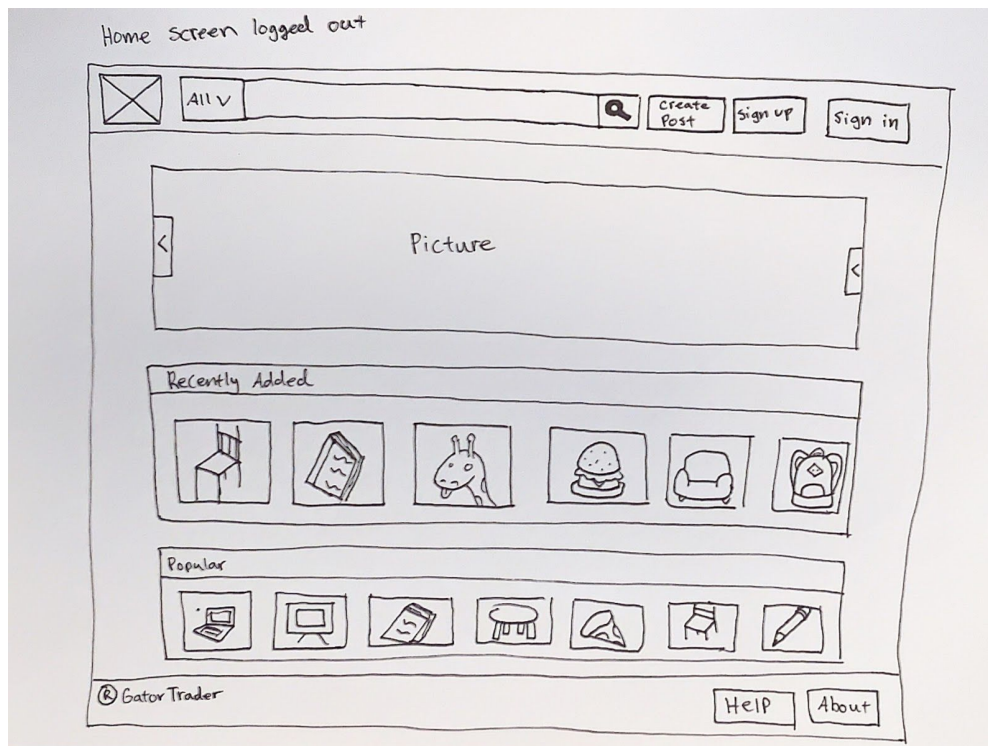
3.0.04 Registered users will have the amount of transactions they have had on the site next to their name.

3.1.00 Registered user shall have postings removed if they become banned

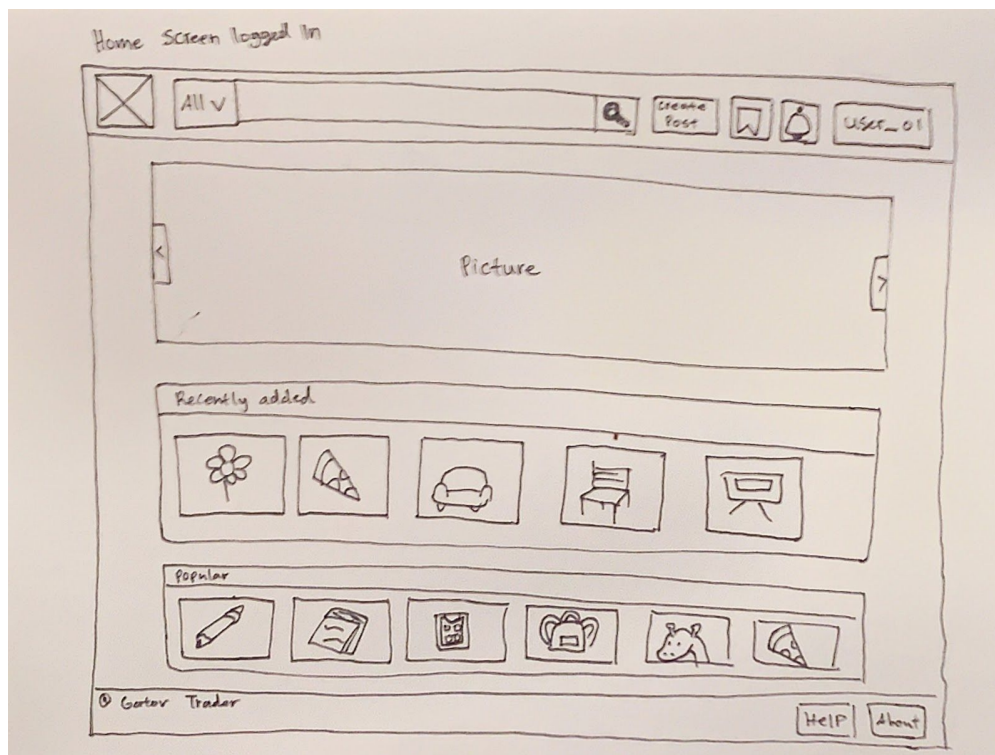
3.1.01 If a registered user had postings listed and becomes banned, their postings will be automatically be removed from the site.

3. UI Mockups and Storyboards (high level only)

Home Screen Page (Logged out):



Home Screen Page (Logged in):



Create Post Page:

Create Post Page

☐ All ☒ V

Create Post

User_01

Title:

Photo

Price

Condition

New ☒

Category

Books ☒

Description

Warning Post may take time

Post Cancel

Search Results Page:

Items for Sale Page

☐ All ☒ V

Giraffe

Create Post

User_01


1 2 3 ... 1,000,000 >>

Filters ☒


Results 1-10 of 1,000,000

Sort by ☒


Item title 1


\$20


Item title 2


\$3

Item title 3


Free

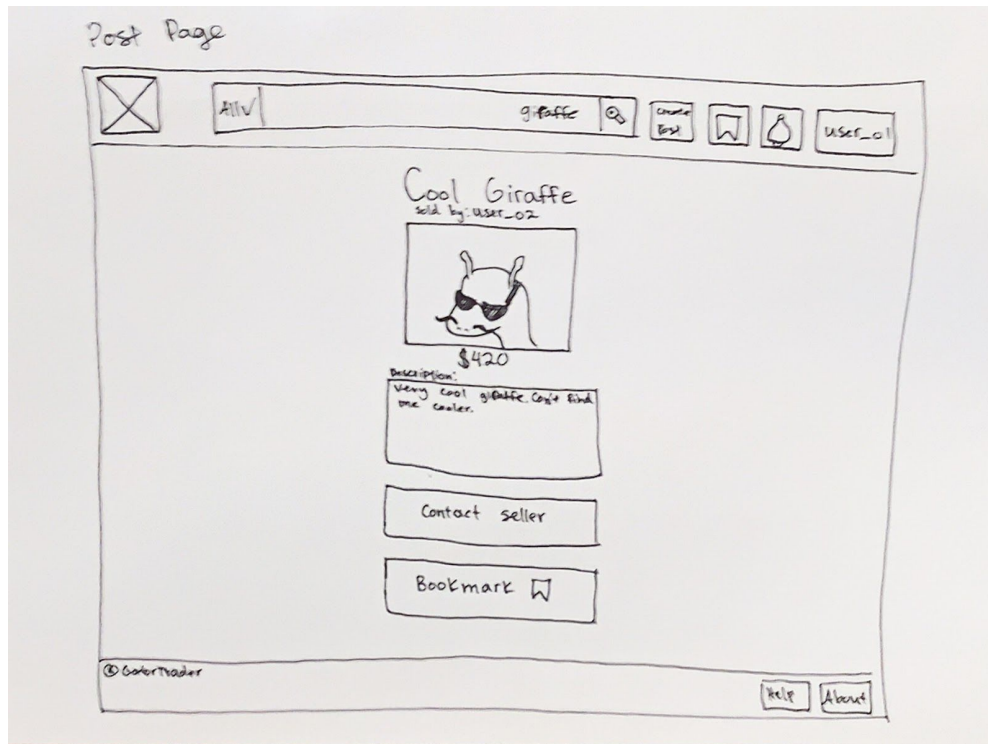
Item title 4


\$420

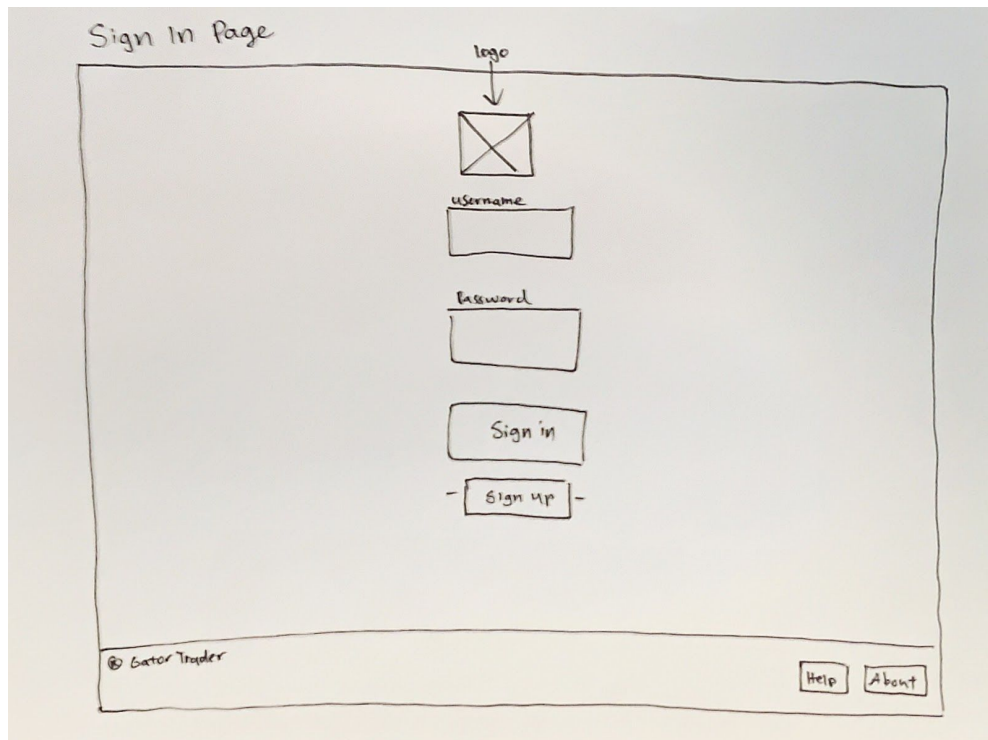
Giraffe Trader

Help About

Post Page:



Sign In Page:



Sign Up Page:

Sign up Page

Logo

username

password

password

Captcha ☒

☒ I Agree

Sign up

Sign in

© Gator Trader

Help About

Message Page:

Message Page

ALL

create Post

User 01

Previous Messages:

User 01

Admin 1

User 02

send to:

User_04

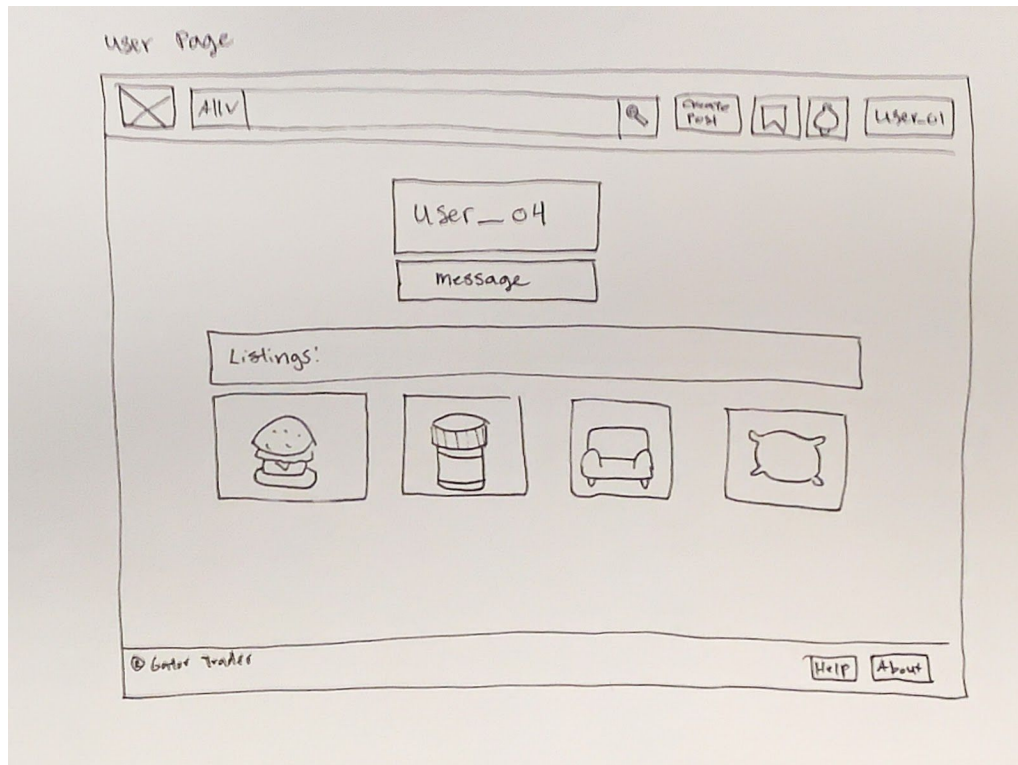
cancel

send

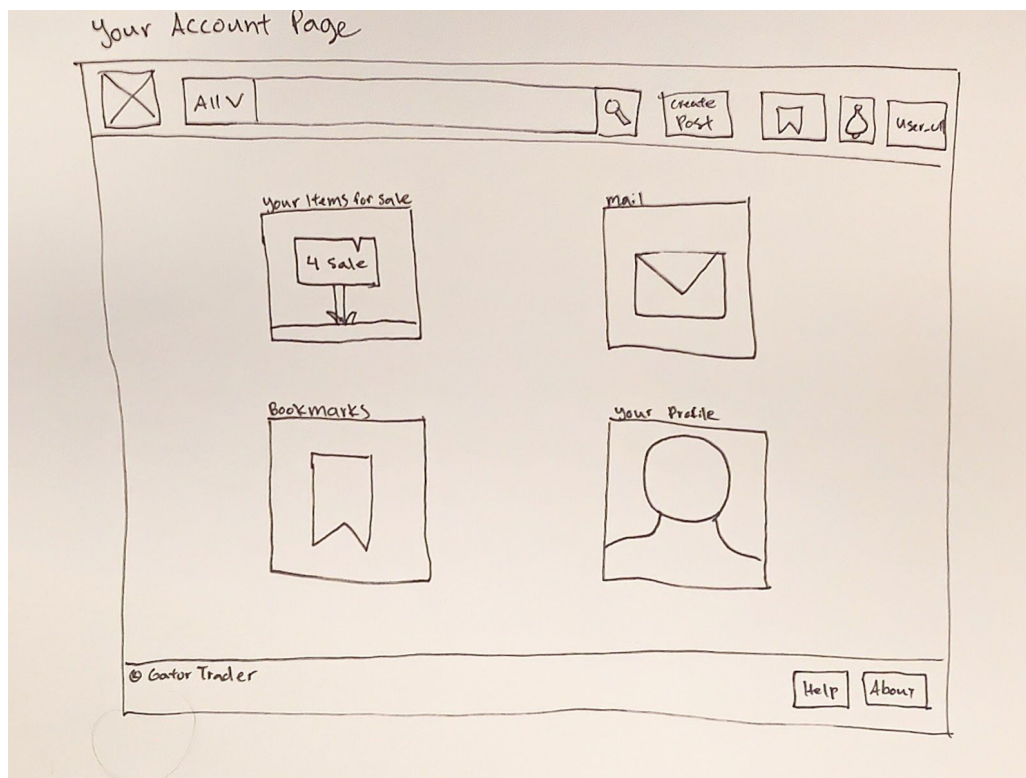
© Gator Trader

Help About

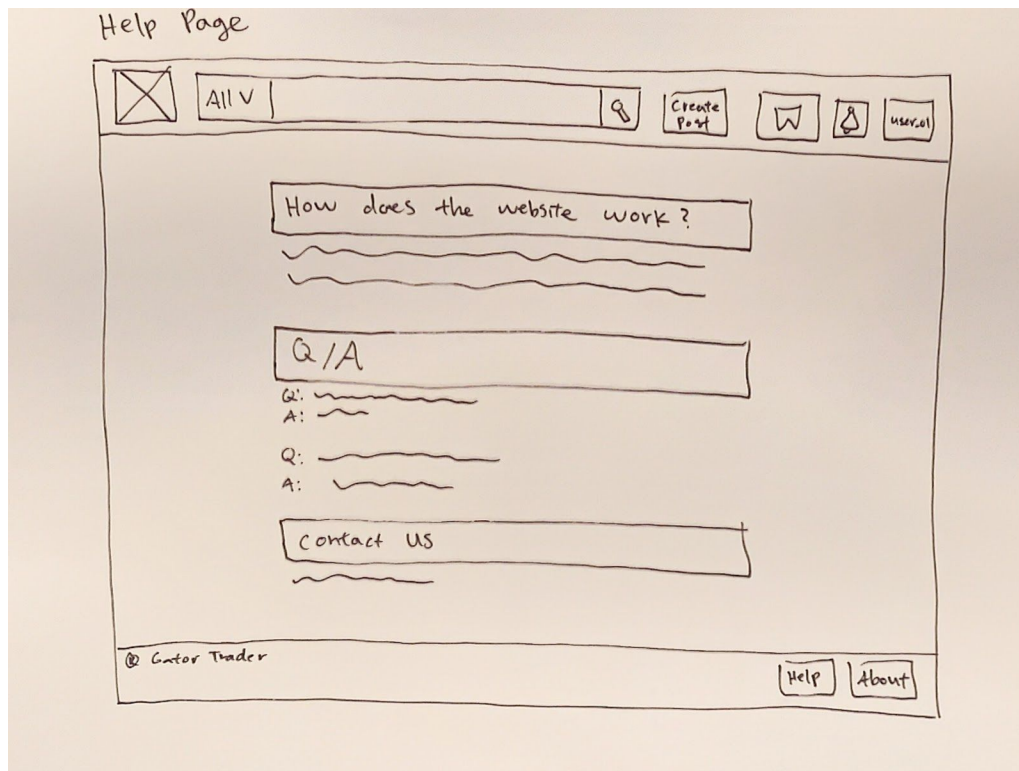
User Page:



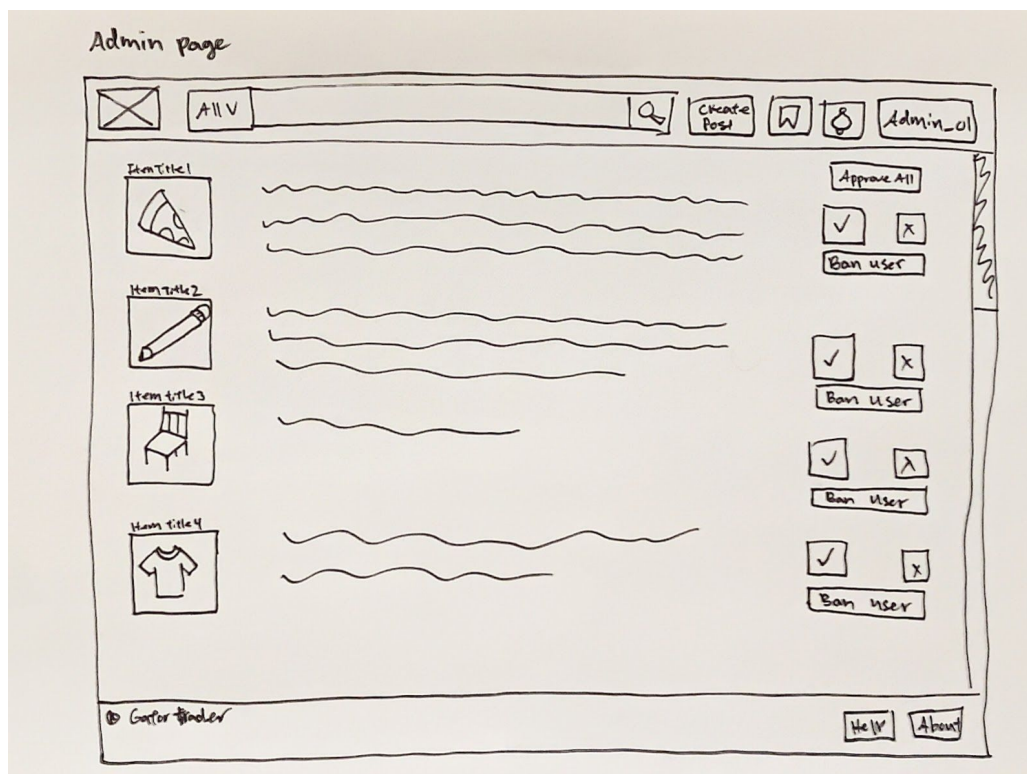
Your Account Page:



Help Page:



Admin Page:



4. High level Architecture, Database Organization

Database Schema

post

Attribute	Type	Key
id	Number	Primary
user_id	Number	Foreign
category_id	Number	Foreign
create_date	Date	
post_title	Varchar(50)	
post_description	Text	
post_status	Char(1)	
price	Number	
is_price_negotiable	Char(1)	
last_revised	Date	

post_image

Attribute	Type	Key
id	Number	Primary
post_id	Number	Foreign
image	Varchar	

registered_user

Attribute	Type	Key
id	Number	Primary
first_name	Varchar(50)	
last_name	Varchar(50)	
email	Varchar(100)	
login_password	Varchar(20)	
is_banned	Boolean	

notification

Attribute	Type	Key
id	Number	Primary
user_id	Number	Foreign
message	Varchar	
read	Boolean	

category

Attribute	Type	Key
id	Number	Primary
category_name	Varchar(20)	
parent_category_id	Number	Foreign

message

Attribute	Type	Key
-----------	------	-----

id	Number	Primary
sender_id	Number	Foreign
recipient_id	Number	Foreign
sent_date	Date	
message	Text	
post_id	Number	Foreign

Media Storage

Images will be stored in the file system. The image directory path or file name will be kept in a table. Only support Images of type jpeg.

Search/Filter architecture

Algorithm / Software Used for Search: MySQL

Post Search:

Search bar input will query the post table for a match in title or description, or if the input matches a category all the items from that category will be displayed in a category page.

Example: search for a phone

```
SELECT *
FROM post p
WHERE p.post_title LIKE '%phone' OR
p.post_description LIKE '%phone%'
```

Filtering: By category and by price range.

Sorting: Posts can be sorted by increasing or decreasing price, and by newest first.

APIs

/api/post/create

Creates a new post in the database. The post's status will be inactive until an admin approves.

Type: POST

Data: title, description, category, image

/api/post/search/:name/:category/:page/:sort

Returns all posts that contain name in the title. It will return a maximum of 50 results.

Request: GET

Data: name, category, page, sort

/api/post/self

Returns all the current logged in user's posts.

Request: GET

/api/post/:id

Returns the full details of a single post based on its id.

Request: GET

Data: id

/api/message/read

Returns all the messages for any active posts for the currently logged in user, ordered chronologically.

Request: GET

/api/message/read/:postid

Returns all the messages for a specific post that the user owns.

Request: GET

Data: postid

/api/message/send/:postid

Returns all the messages for a specific post that the user owns.

Request: POST

Data: postid, message

/api/notification/

Returns all notifications for the currently logged in user.

Request: GET

/api/notification/unread

Returns all unread notifications for the currently logged in user.

Request: GET

/api/notification/read/:ids

Flags specified notifications as read.

Request: GET

Data: ids

/api/categories

Returns all the categories

Request: GET

/api/register

Logs the user into the site and stores their session as a cookie

Type: POST

Data: username , password, captcha

/api/login

Logs the user into the site and stores their session as a cookie

Type: POST

Data: username , password

/api/bookmarks

Returns a list of bookmarked items by user

Request: GET

/

The home page.

Request: GET

/about

Request: GET

/admin

Displays the administrator interface

Request: GET

/user

Displays users page , such as own post and bookmarks

Request: GET

/user/:id

Displays a specified user page

Request: GET

Data: id

/login

Presents a login screen for a user to type in a username or password

Request: GET

/register

Displays to sign up for account

Request: GET

/createpost

Create post page for a user to create a new posting.

Request: GET

/postconfirm

Confirms creation of POST, and informs the user that they must wait for approval

Request: GET

/search/:name/:page/:sort

This will display a list of search results from a category.

Request: GET
Data: name, page, sort

/product/:id

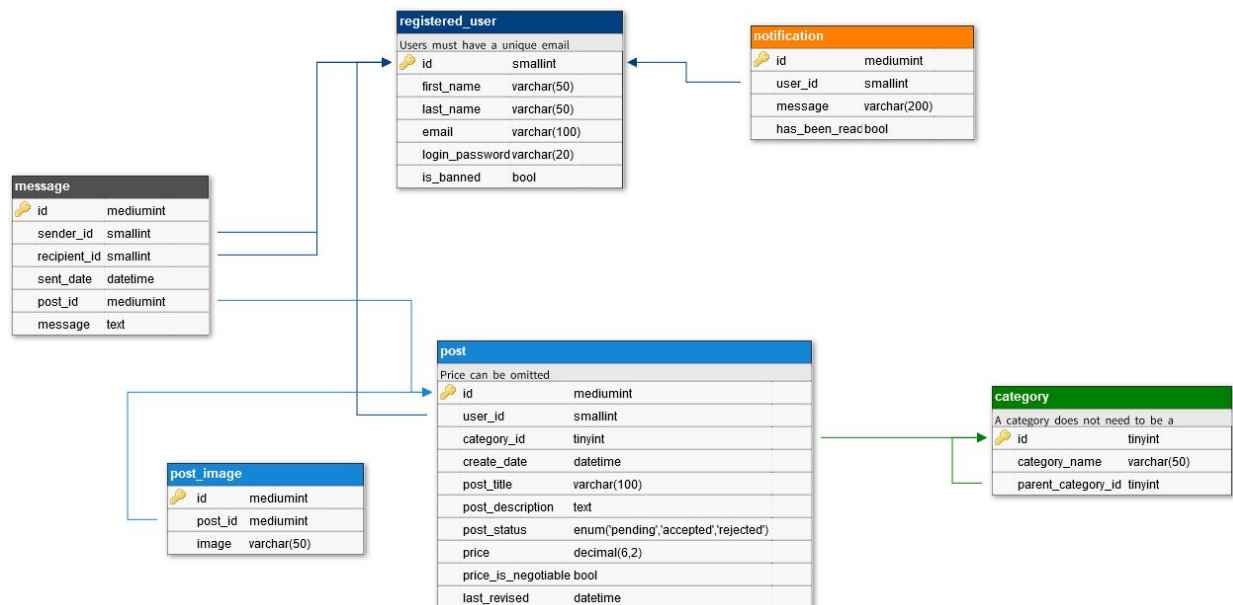
Displays product clicked on
Request: GET
Data: id

Significant non-trivial algorithms

Algorithm to determine what posts to display in the home page, based on their popularity. For example, display the posts that have been viewed the most in the current week, or display posts that contain keywords that are searched the most.

5. High Level UML Diagrams

Database



6. Identify actual key risks for your project at this time

Skills

1. Backend Team has no prior experience in backend web design.
2. Frontend Team has no prior experience in AJAX

Schedule

1. Achin has work every day until 5pm, except Mondays and Tuesdays he gets off at 2pm
2. Achin has a 90 minute commute to school
3. Anthony has a 50 minute commute to school
- 4.

Technical

1. Storage space on AWS for images not adequate

Teamwork

1. No one on team has experience in collaborative coding

Legal

1. Use of unauthorized images. One of the team members could use an image found on a search engine or image repository that isn't authorized for reuse.