

SW Engineering CSC648/848 Spring 2022

GatorBay

Team 6

Team Lead: Bobby Roots broots@mail.sfsu.edu

Backend Lead: Michael Davichick

Backend Member: William Rattan

Frontend Lead: Michael Wang

GitHub Master: Michael Almeda

Milestone 2

Date of submission: 03/19/2022

1. Executive Summary

San Francisco State University is located in the most ideal place with very comfortable weather year round. SFSU is famous for being located in the Bay which is one of the most visited spots in the country by tourists. It is also famous for being only a stone's throw away from the world's tech hub Silicon Valley and fostering an environment in which a student can feel comfortable and therefore more able to learn. One of the greatest unfortunate downsides of attending a University surrounded by some of the best minds this world has to offer is the cost of Attendance. The average college student attending has student loans and credit card bills. Students are looking for any way to help reduce their financial burden while attending SFSU in any way they can. The leading cause of student financial debt at SFSU are the required books and electronics needed for attendance and furniture for new students.

Thanks to the ingenious thinking from team 6 those debt ridden individuals will have a way to save more money. GatorBay is a web app that would allow students, staff, and faculty to purchase and sell books, electronics, furniture and clothing from other students, staff and faculty. In many cases the student will have to pay nothing due to Gator Bay's exchange option. The ability to receive these items any time day or night is an added bonus for people in a rush.

All of us from Team 06 acknowledge that Students, Faculty, and Staff could use a website that caters more to them. That is why Team 6 has decided to build GatorBay. GatorBay will offer the exchange of goods such as Furniture, Clothes, Books, and electronics. What makes GatorBay so unique is that it is exclusively for SFSU Students, Faculty, and Staff. That along with our 24/7 operating hours at safe locations throughout SFSU we help to assist in the ease of making transactions in a non-harmful environment. Team 06 is dedicated to making this dream a reality.

2. List of main data items and entities (expand as necessary)

- a. product - A product serves as a placeholder for all items sold on the site.
There are as follows:
 - i. books
 - 1. Textbooks
 - 2. Magazines
 - 3. Novels
 - ii. electronics
 - 1. Laptops
 - 2. Cell Phones
 - 3. TVs
 - iii. furniture
 - 1. Tables
 - 2. Chairs
 - 3. Couches
 - iv. clothing
 - 1. Shirts
 - 2. Pants
 - 3. Jackets
- b. unregistered_user - Unregistered users are able to browse posts but are required to log in / register when attempting to purchase, exchange or sell.
- c. registered_user - Registered users can not only browse but post sell orders, purchase and exchange. Contact information regarding the poster is only visible to registered users as a safety precaution.
- d. admin - can do everything a Registered User can do and can approve or delete posts from other Registered Users.
- e. buy - Purchase products that Registered Users post.
- f. sell - Sale products that registered users post.
- g. exchange - Exchange products that registered users post.
- h. time - time the product is to be exchanged
- i. location - Location the product is to be exchanged
- j. u_name - name of registered user.
- k. p_name - name of product
- l. password - password of registered users
- m. phone_number - phone number of registered user
- n. r_id - unique registered user id
- o. a_id - unique admin id
- p. description - details about the product being displayed

- q. transaction_id - unique id of product transaction
- r. awaiting - waiting for registered user who posted product to accept or decline transaction
- s. accepted - registered user who posted product has accepted proposal to transfer product to another registered user
- t. pending - transaction has an interested reg
- u. decline - transaction has been declined by registered user posting product

3. Functional Requirements - prioritized

a. Priority 1

- i. Unregistered:
 - 1. Unregistered users shall be able to browse posted items.
 - 2. Unregistered users shall be able to create an account.
 - 3. Unregistered users shall be able to log in.
- ii. Registered:
 - 1. Registered users shall inherit everything unregistered users have
 - 2. Registered users shall be able to buy displayed products.
 - 3. Registered users shall be able to post products
 - 4. Registered users shall be able to sell products
 - 5. Registered users shall be able to see available messages on a personalized dashboard.
- iii. Admin:
 - 1. Admin users shall inherit everything registered users
 - 2. Admin users shall be able to approve post request
 - 3. Admin users shall be able to disapprove post request
 - 4. Admin users shall be able to remove posts

b. Priority 2

- i. Unregistered:
 - 1. Unregistered users shall also be able to search for specific posts.
- ii. Registered:
 - 1. Registered users shall be able to delete their account.

2. Registered users shall be able to select the time of exchange.
 3. Registered users shall be able to select drop off locations:
 - a. Library
 - b. Cesar Chavez Student Center
 - c. SFSU Police Headquarters
 - iii. Admin:
 1. Admin users shall be able to ban registered users
- c. Priority 3
 - i. Unregistered:
 1. Unregistered users shall be able select dark mode.
 - ii. Registered:
 1. Registered users shall be able to view history.
 - iii. Admin:
 1. Admin shall be able to message registered users.
 2. Website shall display recent posts.

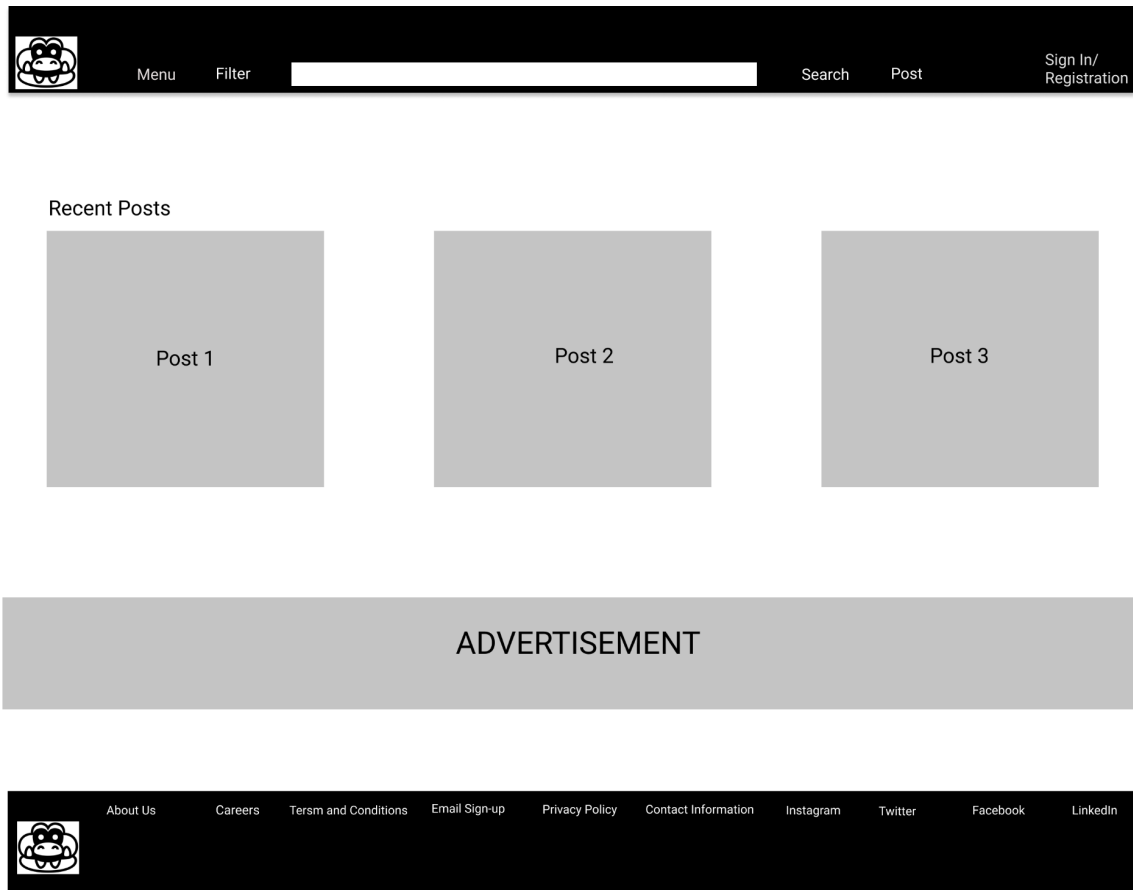
4. UI Storyboards for each main use case (low-fidelity B&W wire diagrams only)

Use Cases:

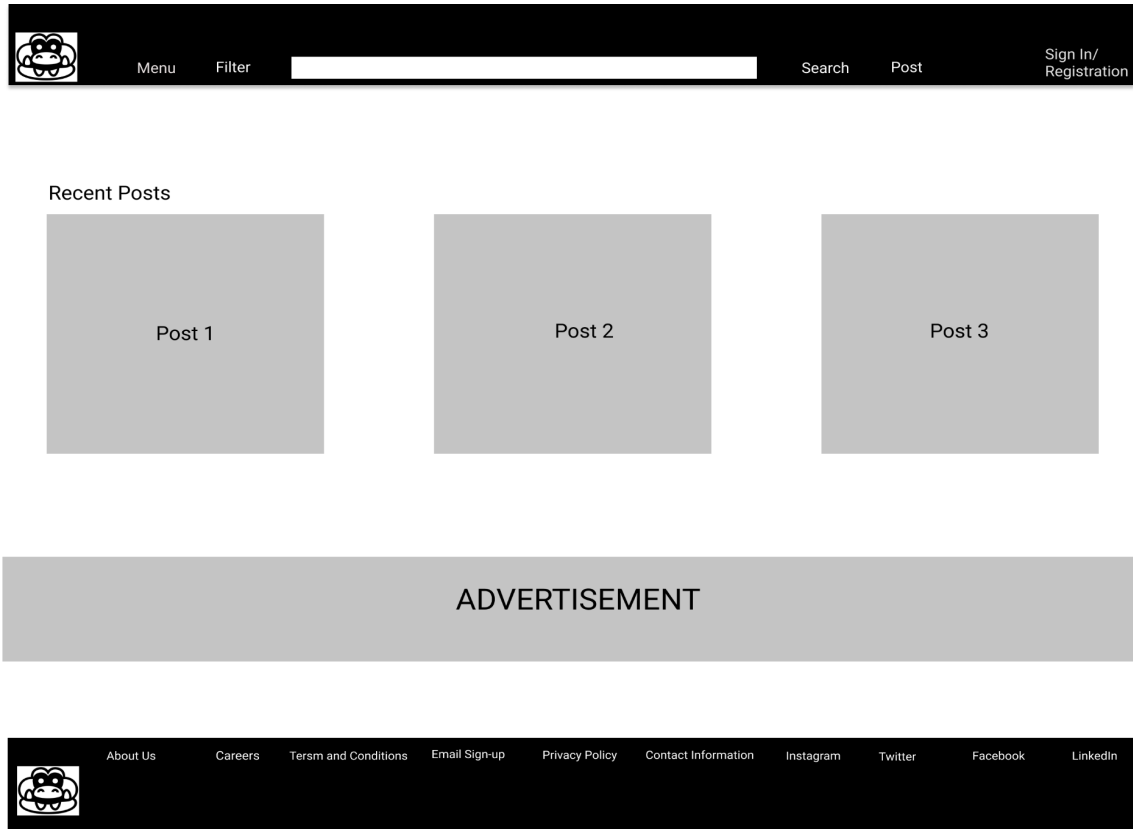
- 1. Lisa is looking to exchange a book for another book**
- 2. Alex is looking to purchase furniture.**
- 3. Samatha wants to sell a book**
- 4. John is the system admin.**

Use Case One

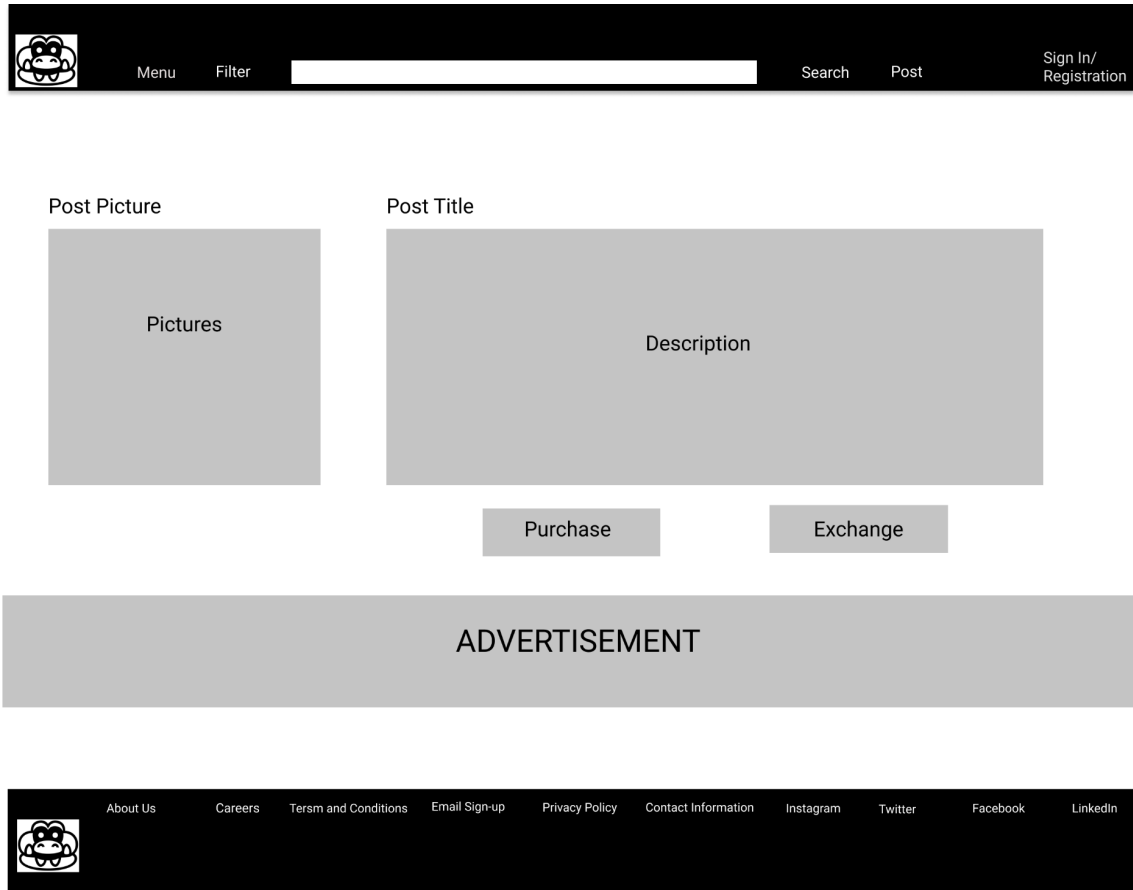
Homepage: Lisa navigates to the homepage of Gatorbay to find the books she needs for class. Lisa easily finds the filter because it's next to the search bar where she expects the filter option to be because of her experience with other websites. She selects the books filter to filter the products to only books.



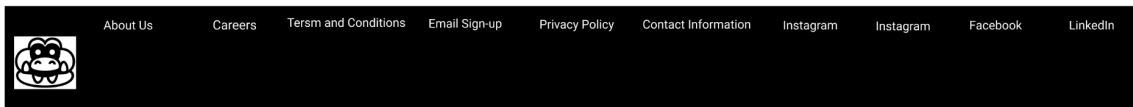
Filter: Lisa looks through the posts on books and finds the books she is looking for. She clicks on the book thumbnail to see the post in detail.



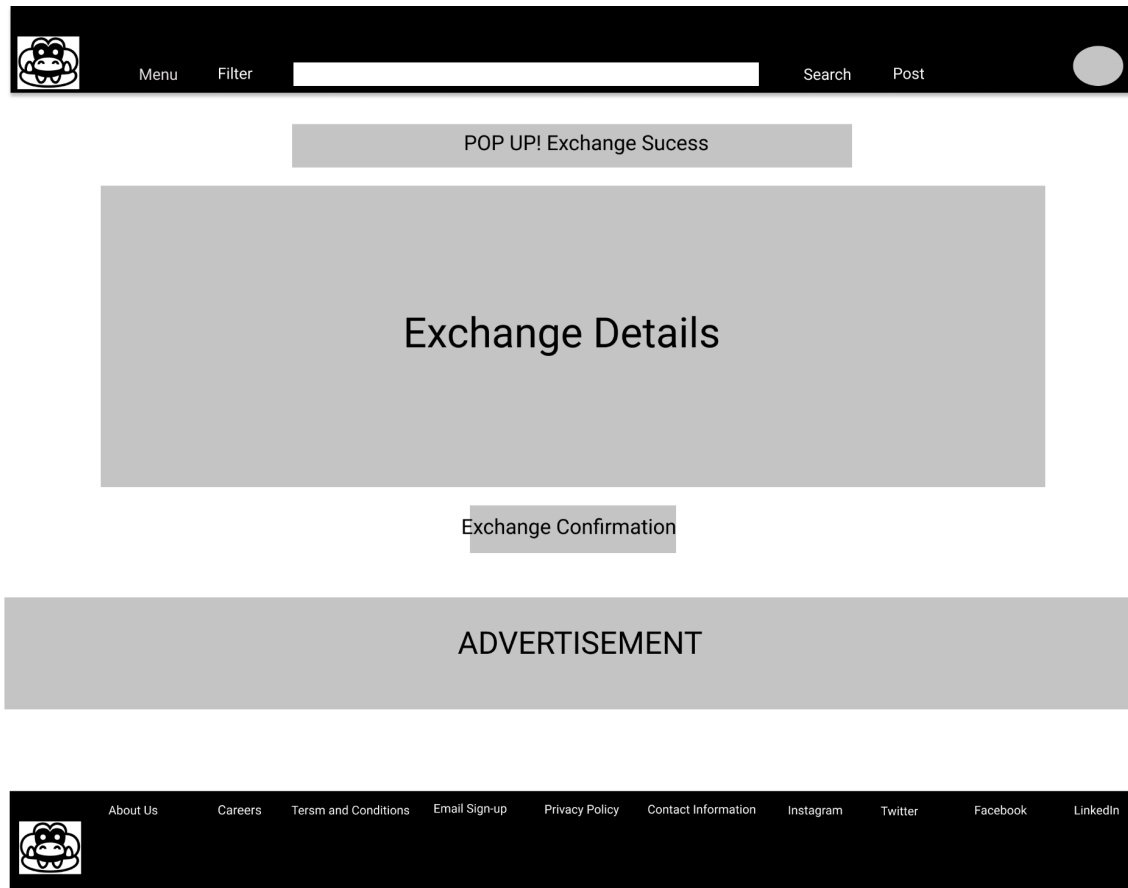
Product Page: Lisa sees the book's pictures and description, and decides she wants to purchase the book. Lisa clicks on the purchase button.



Login: Lisa is prompted to login. After filling out the necessary information Lisa selects Sign In.

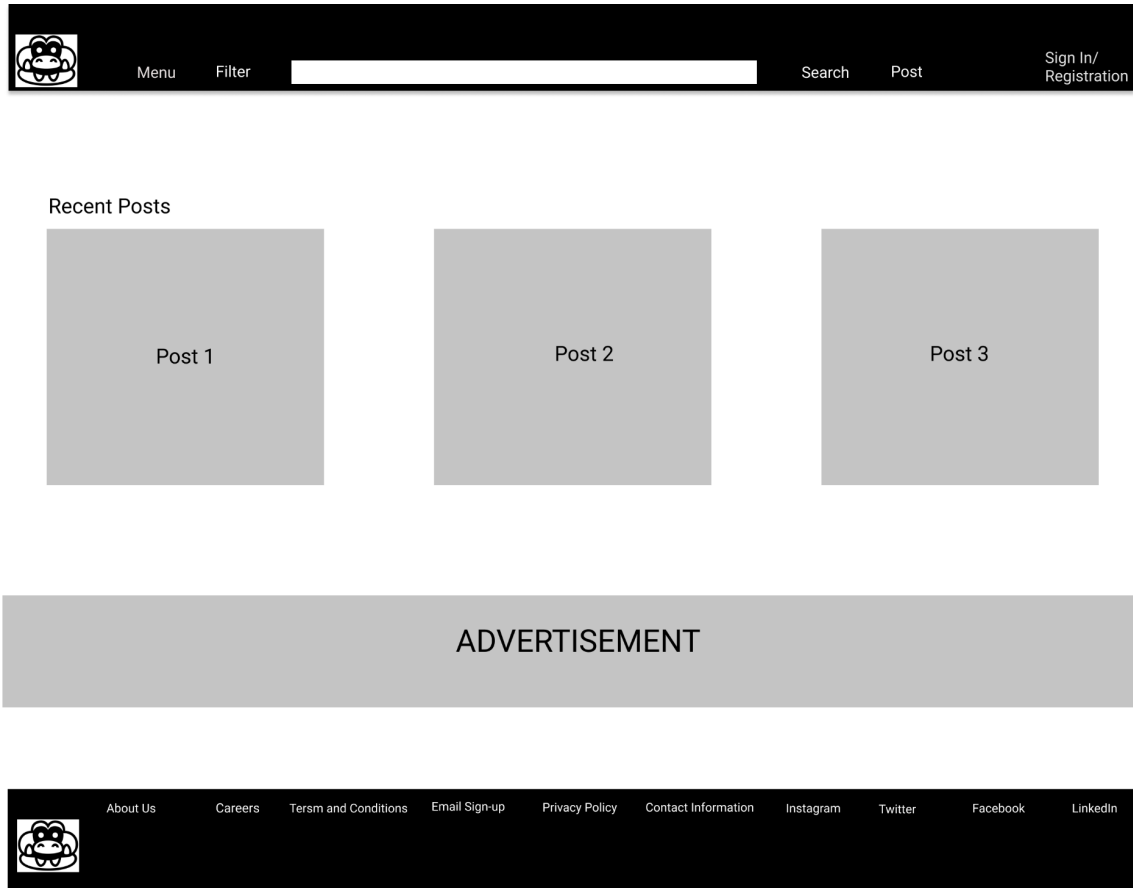
The image shows a light gray rectangular login form. Inside the form, there are two white input fields. The first field is labeled "Email" in gray text to its left. The second field is labeled "Password" in gray text to its left. Below the password field is a gray button with the text "Sign In" in white. Underneath the button, there are four lines of gray text: "Cant Log in?", "Forgot Password", "Not a member yet?", and "Create Account".

Confirmation: Lisa sees the Exchange Confirmation button. After selecting it Lisa sees that her exchange has been processed successfully. The exchange details list out everything that transpired.

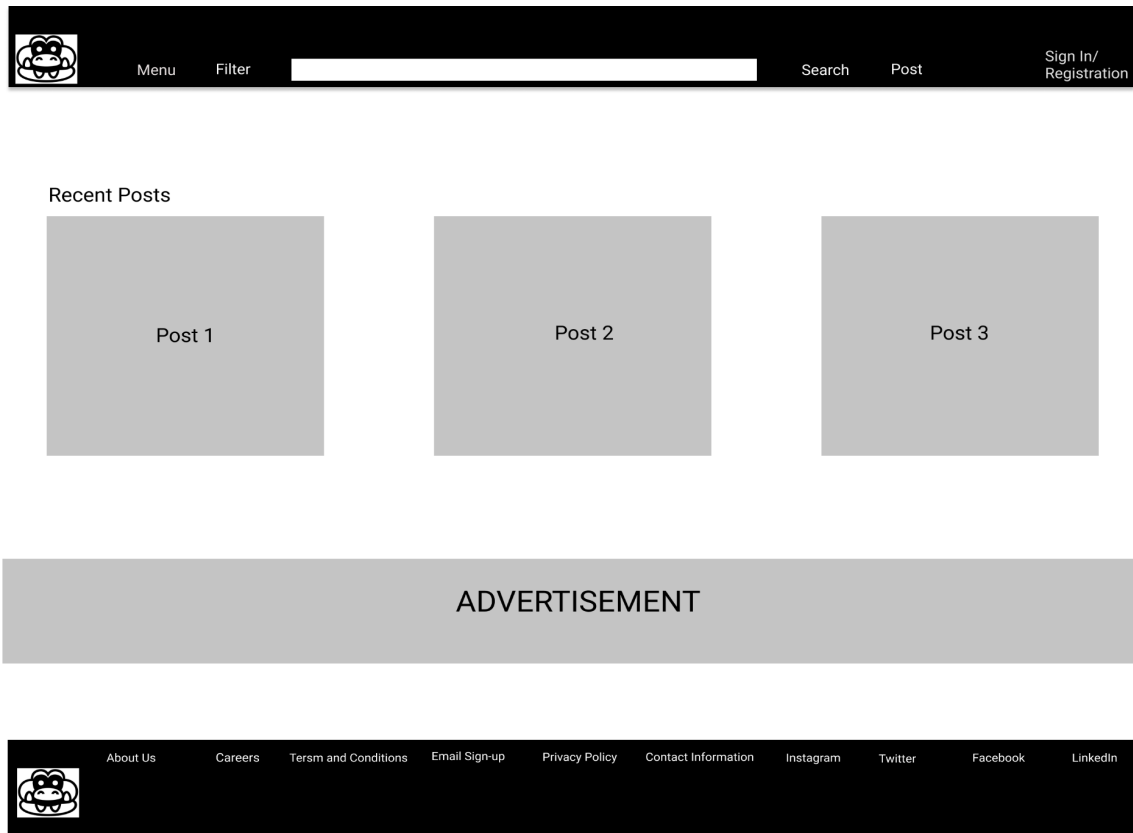


Use Case Two

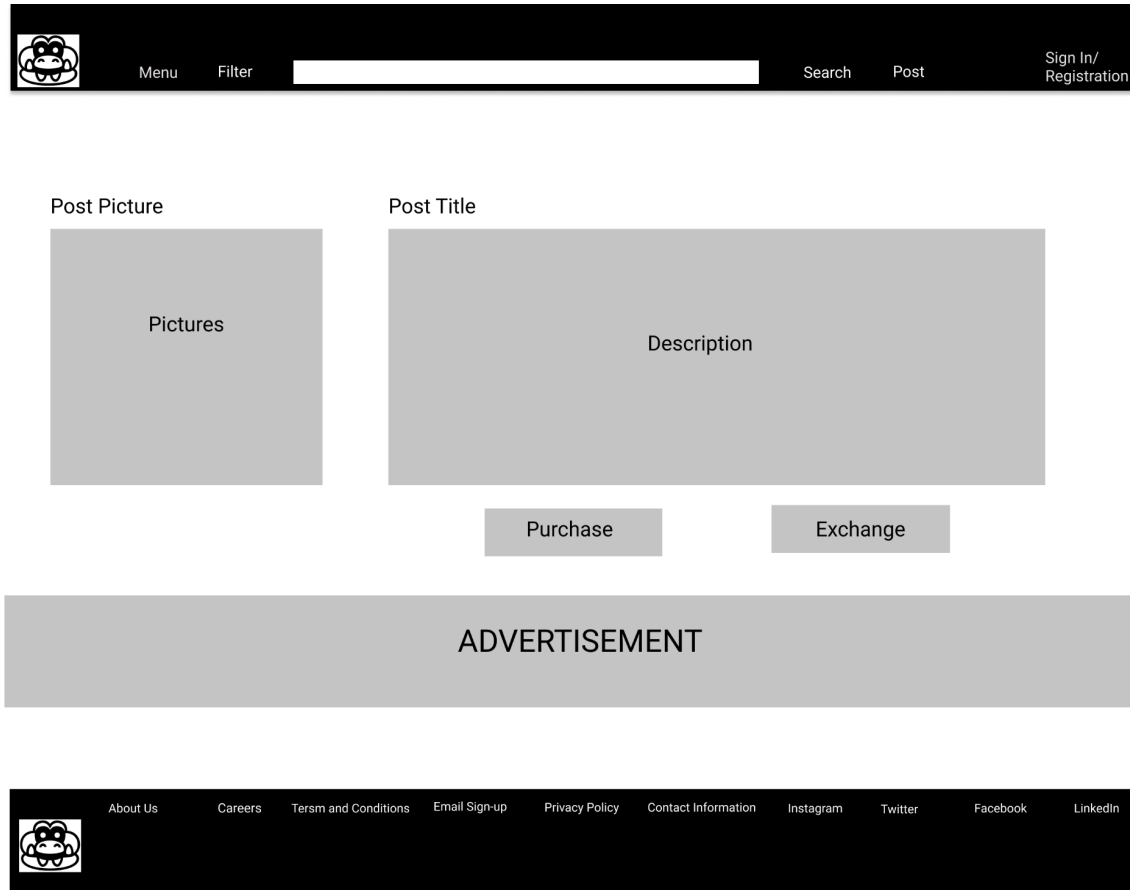
Homepage: Alex navigates to the homepage of Gatorbay to find furniture he needs for his room. Alex easily finds the filter button because it's next to the search bar where he expects the filter option. He selects the furniture filter to filter the products to only furniture and selects the search button.



Filter: Alex sees the selection of furniture and decides to select the chair that he wants.



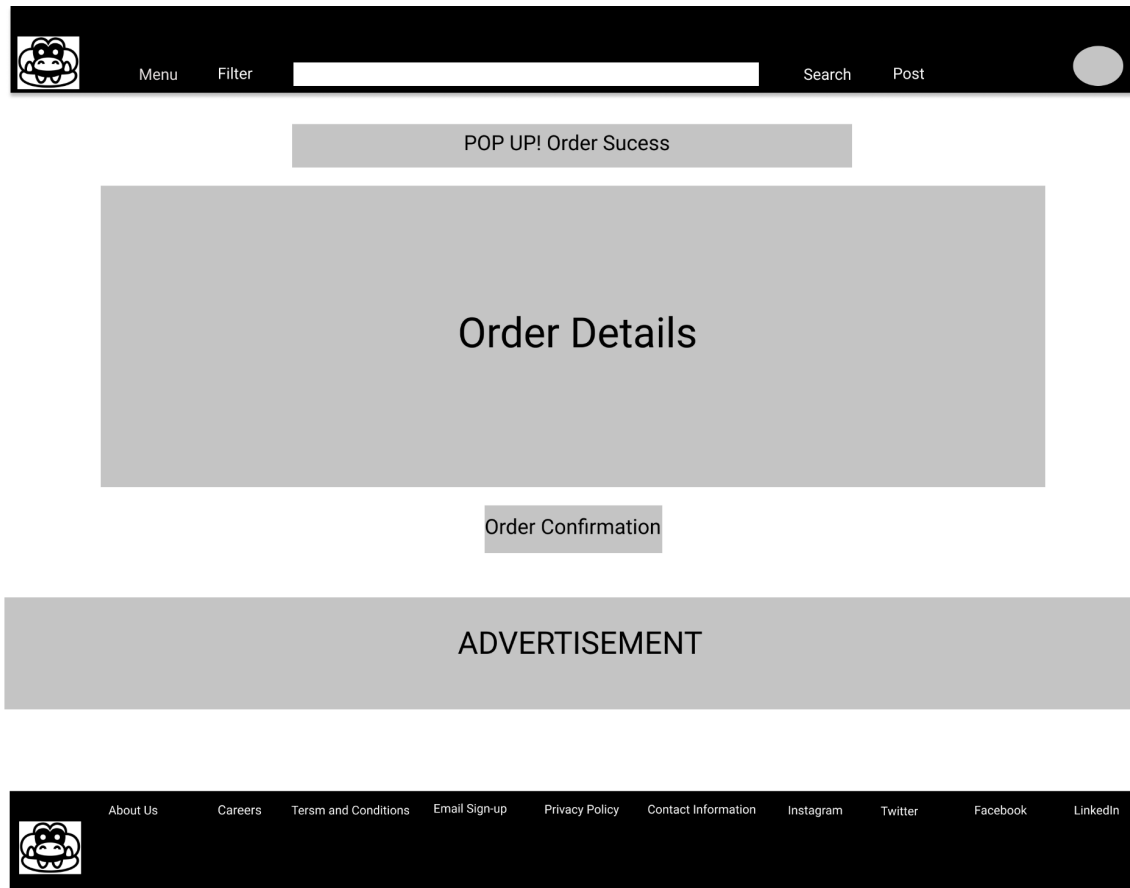
Product Page: Alex sees the chair's pictures and description, and decides he wants to buy the chair that he sees. Alex clicks on the purchase button.



Login: Alex is prompted to login. After filling out the necessary information Alex selects Sign In.

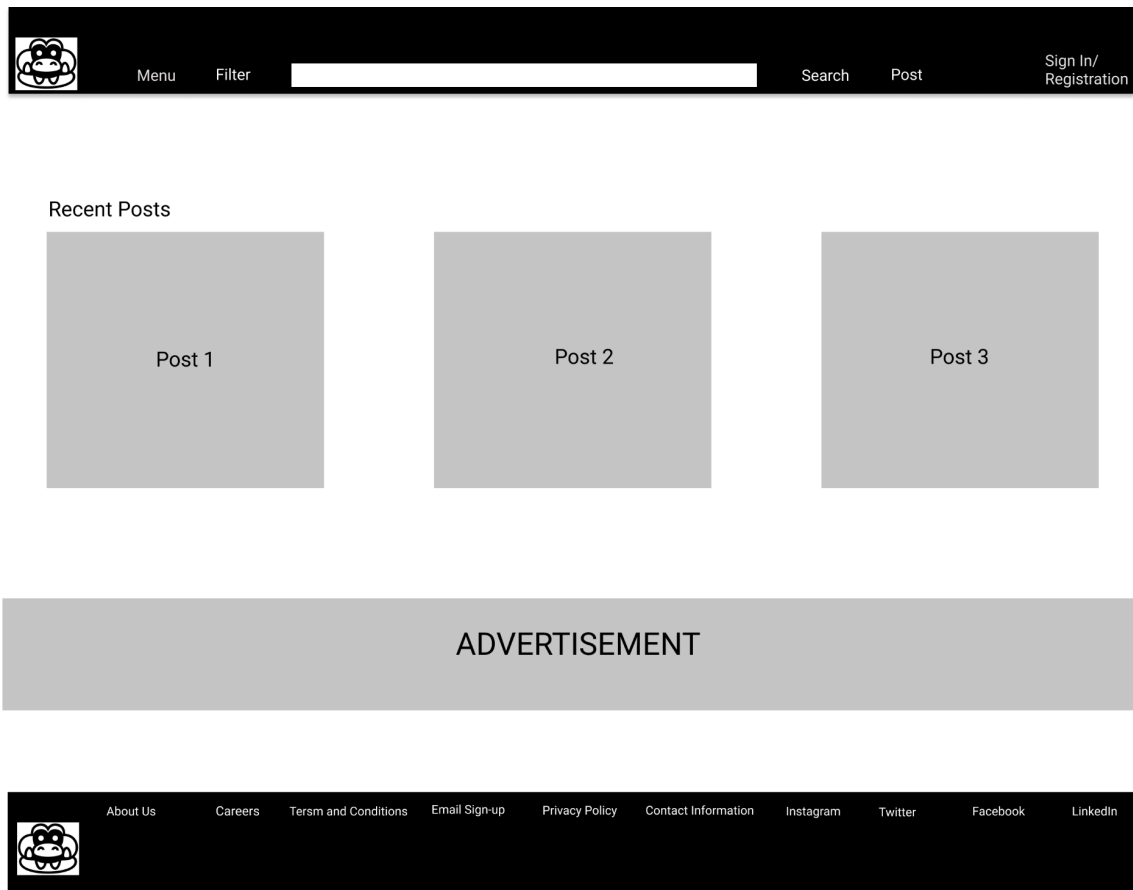
A light gray rectangular box containing a login form. At the top left is the label "Email" in black, followed by a white rectangular input field. Below this is the label "Password" in black, followed by another white rectangular input field. Centered below the password field is a gray button with the text "Sign In" in white. Below the button are four lines of text: "Cant Log in?", "Forgot Password", "Not a member yet?", and "Create Account".

Confirmation: Alex sees the Order Confirmation button. After selecting it Alex sees that his exchange has been processed successfully. The exchange details list out everything that transpired.




Use Case Three

Homepage: Samantha navigates to the homepage of Gatorbay to prepare to create a post to sell her textbook. Samantha sees the Post button on the homepage and clicks on that button.



Posting: Samantha sees the different categories she needs to fill out to create a post, so she links some pictures of her book, then fills out the form and clicks submit.

[Menu](#)[Filter](#)[Search](#)[Post](#)[Sign In/Registration](#)

Photos


Title

Description

Category

Submit

ADVERTISEMENT

[About Us](#)[Careers](#)[Term and Conditions](#)[Email Sign-up](#)[Privacy Policy](#)[Contact Information](#)[Instagram](#)[Twitter](#)[Facebook](#)[LinkedIn](#)

Login: Since Samantha does not have an account, she selects the Create Account button.



Email

Password

[Sign In](#)


[Cant Log in?](#)

[Forgot Password](#)

[Not a member yet?](#)

[Create Account](#)

Registration: Samantha registers for a Gatorbay account by filling out all the fields that are required to be filled in. After filling out the form she clicks the Join Us button. A pop up appears with the words, "Registration Success". After a few seconds Samantha is redirected to the Post Confirmation page.

MenuFilterSearchPostSign In/
Registration

POP UP! Registration Success

First name

Last name

Student ID


Email

Phone #

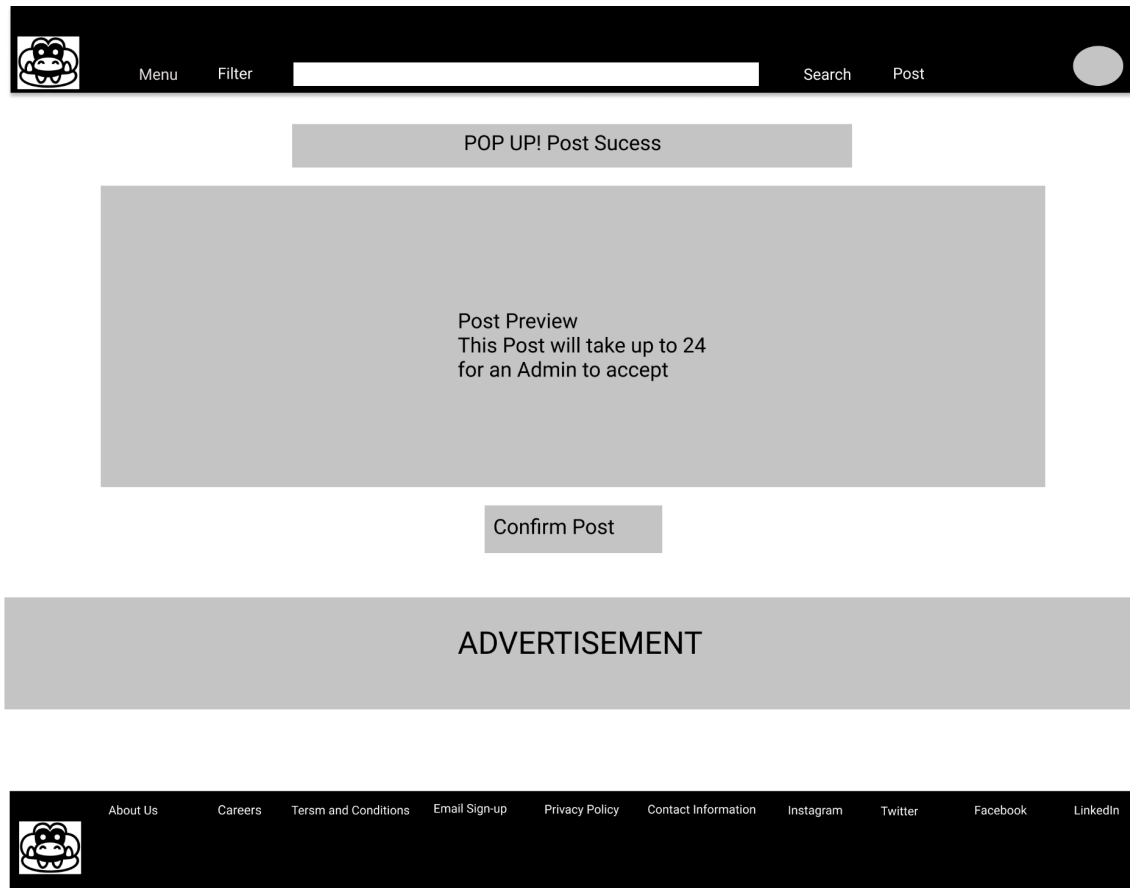
Password

☐ By clicking, you are confirming that you agree to this sites Terms and Conditions.

Join Us

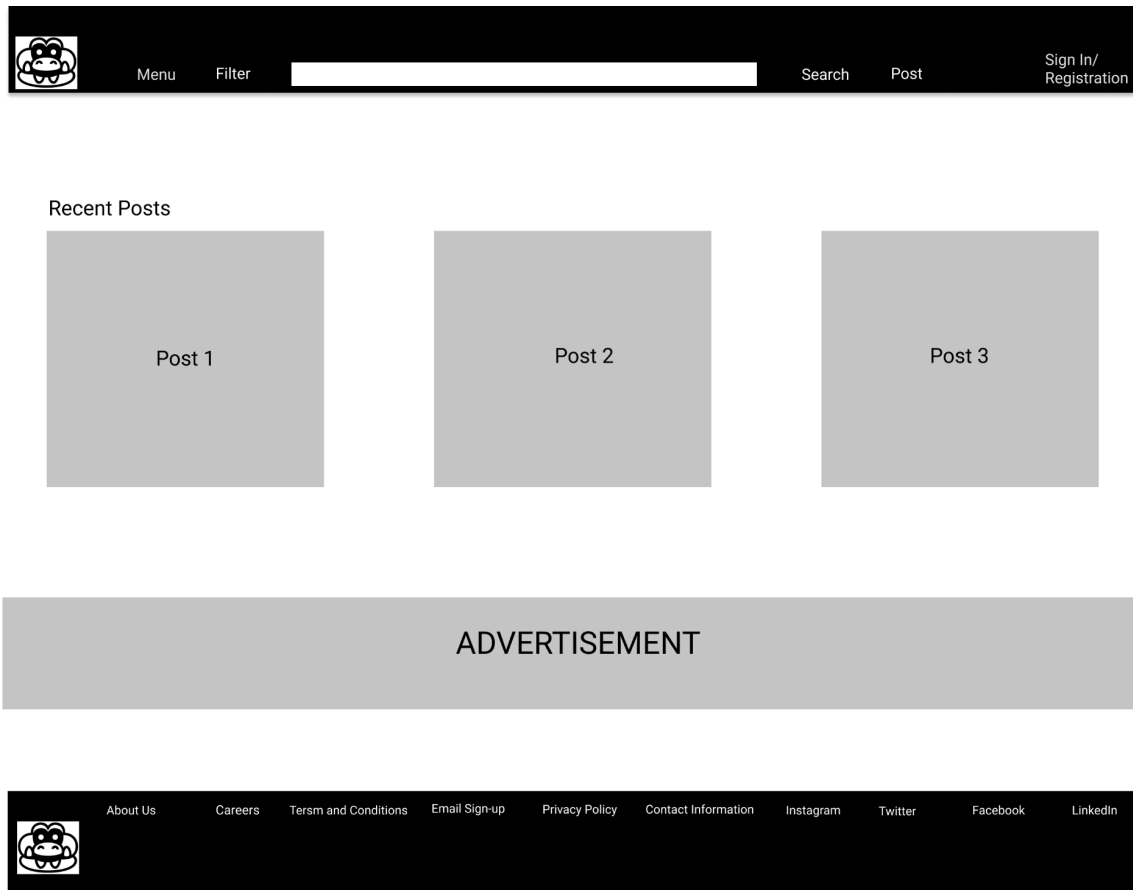
About UsCareersTerm and ConditionsEmail Sign-upPrivacy PolicyContact InformationInstagramTwitterFacebookLinkedIn

Confirmation: Samatha sees the Confirm Post button. After selecting it Samatha sees that her post has been processed successfully. The post details list out everything that transpired.



Use Case Four

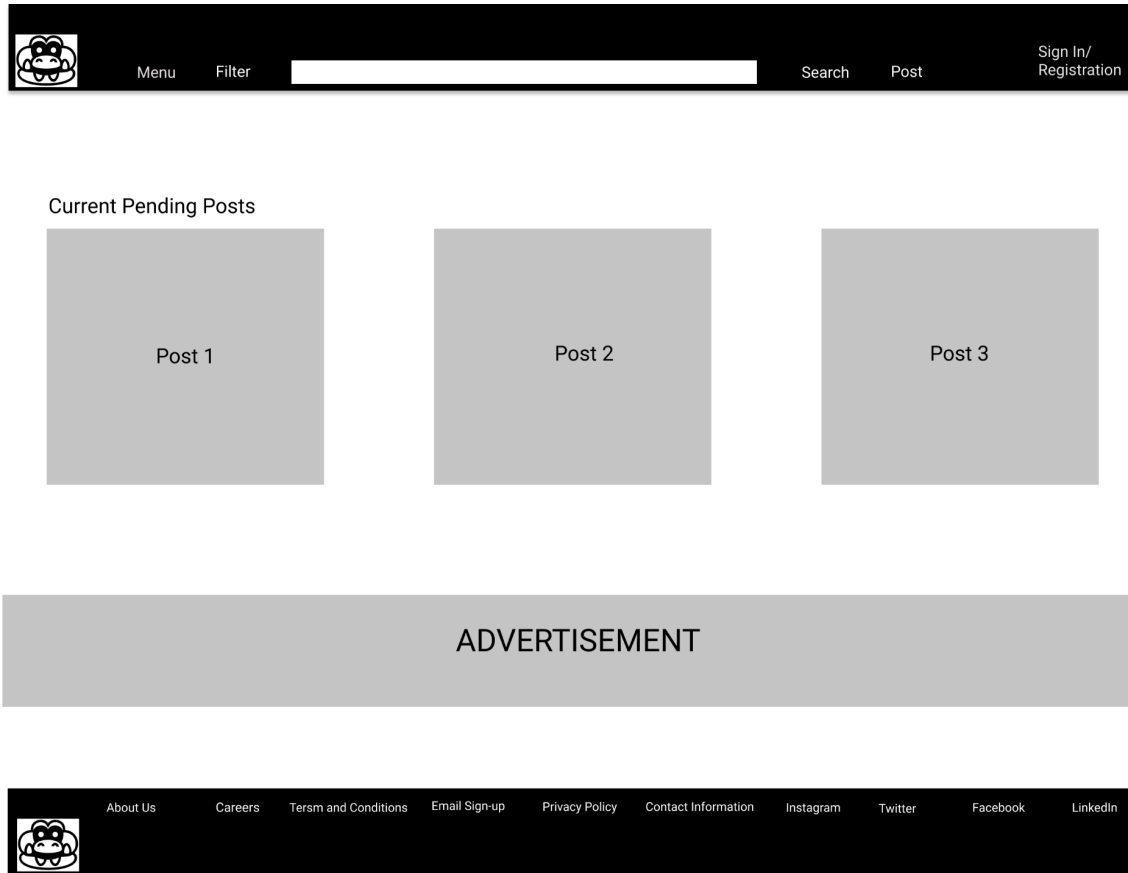
Homepage: John is the system admin. He navigates to the homepage and decides to sign in as the admin.



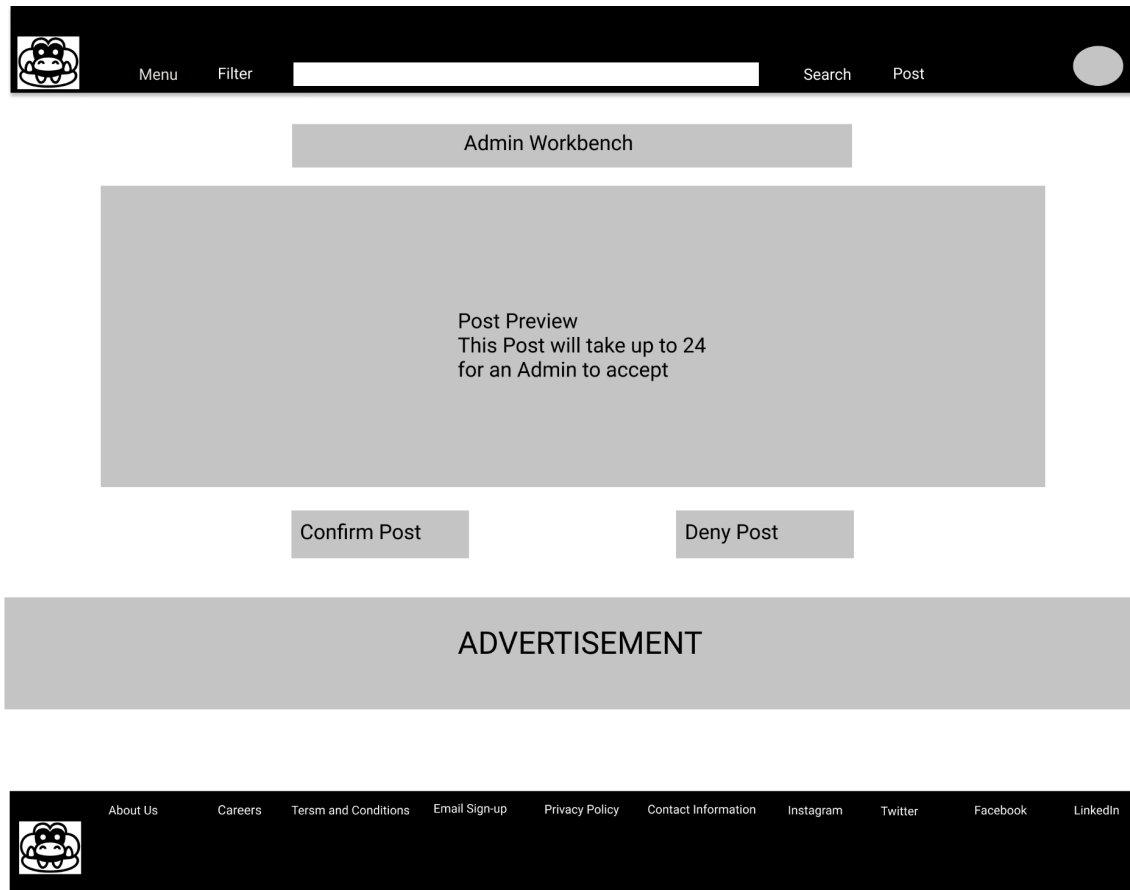
Login: John is prompted to login. After filling out the necessary information John selects Sign In.

A light gray rectangular box containing a login form. At the top left of the box is the word "Email" in a small, dark font, followed by a white rectangular input field. Below this is the word "Password" in a small, dark font, followed by another white rectangular input field. Centered below the password field is a light gray rectangular button with the text "Sign In" in a small, dark font. Below the button are four lines of text, each preceded by a small dark square bullet point: "Cant Log in?", "Forgot Password", "Not a member yet?", and "Create Account".

Pending Post: John sees the display of the current pending posts. He then selects one of them.



Post Confirmation: John reviews the post and decides to confirm it using workbench.



5. High level Architecture, Database Organization summary only

Libraries: React (Framework for app)
 Axios (Used for API calls from front end / back end)
 Bootstrap (Templates used for front end)
 React-Router (Used to route our pages)

Front end runs on React
Back end runs on Node, Express

Database Organization Summary

Users Table

INT <u>uid</u> (uq, ai, nn)	VARCHAR(225) uname (nn)	VARCHAR(225) password (nn)	INT phone_number (nn)	VARCHAR(225) email (nn)
--------------------------------	-----------------------------------	--------------------------------------	---------------------------------	-----------------------------------

- **uid**: user ID; unique to distinguish one user from another.
- **uname**: user name; each user has their own user name.
- **password**: each user creates a password so only they can log into their account.
- **phone_number**: every user provides their phone number for contact information.

Products Table

INT <u>pid</u> (uq, ai, nn)	VARCHAR(225) pname (nn)	VARCHAR(225) description (nn)	VARCHAR(255) ptag (nn)
--------------------------------	-----------------------------------	---	----------------------------------

- **pid**: Products ID: unique to identify different products from one another.
- **pname**: The name of a product.
- **description**: The description of the product.
- **ptag**: Product tag; determines if the product is a book, furniture, or electronic.

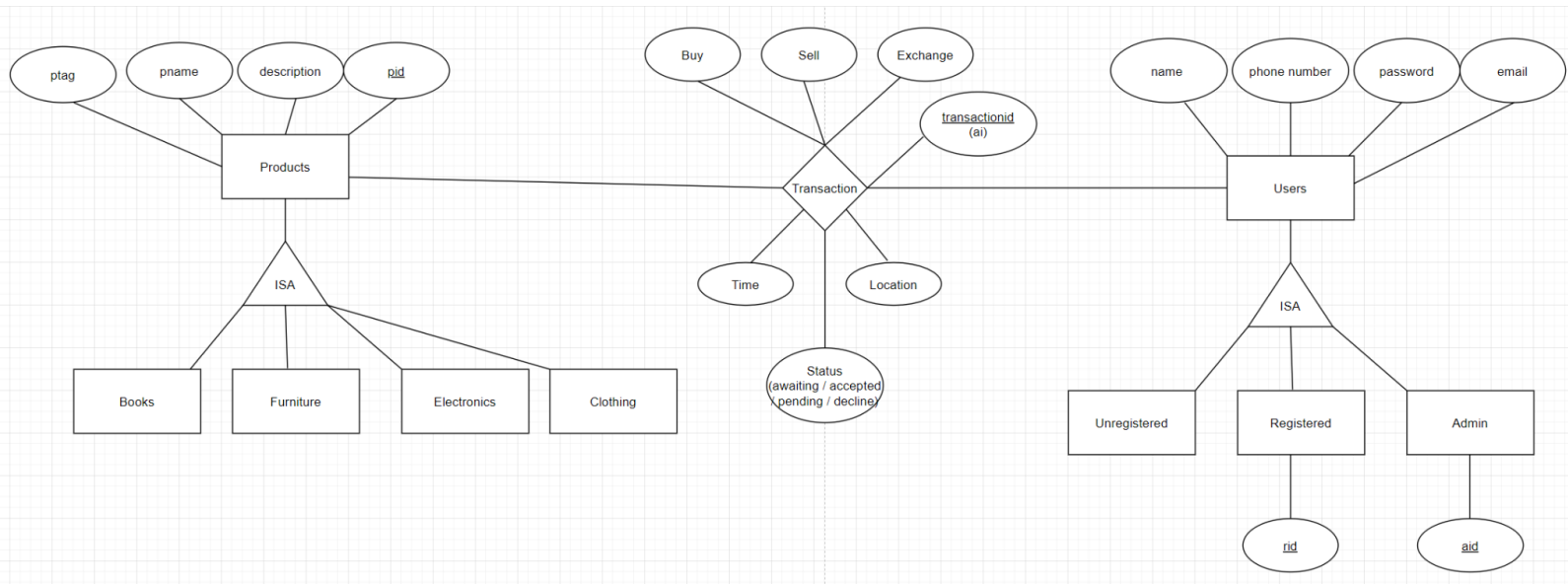
Transaction Table

INT <u>tid</u> (uq,ai,nn)	VARCHAR(225) ttag (nn)	DATETIME time (nn)	VARCHAR(225) location (nn)	VARCHAR(225) status (nn)
------------------------------	----------------------------------	------------------------------	--------------------------------------	------------------------------------

Foreign keys of Transaction Table

INT <u>pid</u> (fk)	INT <u>uid</u> (fk)
------------------------	------------------------

- **tid:** transaction ID; unique to identify differing transactions from one another.
- **ttag:** transaction tag; determines if the transaction is a buy / sell / or exchange.
- **time:** stores the time for any pending / completed transaction.
- **location:** location of various approved safe zones for transactions.
- **status:** keeps track of a post's status; if a buyer moves forward with a transaction and the exchange is pending, the status will reflect that.



Media Storage

- The only media being used in our application are JPG files for product images and our group's profile pictures. All of this media will be stored in DB BLOBs.

Search / Filter Architecture and Implementation

- The search consists of two attributes; Product Tag (ptag) and Product Description (description) that are saved and updated in the React State. The Product Tag (ptag) exists in the form of a drop down on the side of the search bar and contains all of the product types stored within the database (Books, Furniture, Electronics, Clothing). The Product Description is stored via an onChange function that auto updates every time the user enters a character into the search bar. When the user hits search, both the drop down value (ptag) and the search value (description) are sent within an API function to the back end.
- When filtering, the back end grabs these two values (ptag, description) and sends a query to the database using SELECT. The ptag attribute uses the WHERE operator to grab the exact match, while the description attribute uses

the LIKE operator followed by “%” to return any product that matches the regular expression value of description.

- If the user selects “none” from the drop down, requesting that no specific category is selected, the SQL statement will instead pull all categories from the database. The LIKE operator is still implemented in this functionality, allowing the user to search and filter requested parameters as intended.

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

- skills risks:
 - Team lead shall learn more about Bootstrap and React.
- schedule risks:
 - Due to only having five people in the team the schedule will always be a pressing matter.
- technical risks
 - None
- teamwork risks
 - None
- legal/content risks:
 - None

7. Project management

None at this time.