

SW Engineering CSC648/848 Fall 2018

GatorTrader

Team 01

Team Lead - Marcus Mertilien
Email: marcusmertilien@gmail.com

Front-End Lead - Michael Phan
Athena Javier
Alex Ha

Back-End Lead - Albert Shevchuk
Raul Serrano
Daniel Martinez

Milestone 2
October 18, 2018

Revision Date	Notes

1. Data Definitions V2

1. **Guest User**- A user without an account, can still browse the web.
2. **Item**- Books, Calculator, Furniture, etc. that is on sell.
3. **Messages**- Guest users are able to contact the sellers about the item posted for sell.
4. **Registered User**- User with an account can message other sellers about items.
5. **Email**- An .edu email address to show they're SFSU students to buyers.
6. **Image**- Help view what/how/look the item on sell
7. **User record**- Registered users are able to keep track of what they posted.
8. **Admin**- User with special privileges; have the ability to remove posts from the site and generally enforce the Terms and Conditions of the site.
9. **Category**- Guest and registered users are able to search items by category.
10. **Drop Down Category**- Let users search easier by selecting by category.
11. **Search Engine**- Users are able to search any item on the search engine. Such as, "books", "Calculator", etc.
12. **Account**- Buyers shall register to complete a transaction.
 - a. *First Name and Last name*- the users' Full name
 - b. *Birthday*- must be 18 years old and older to make a transaction on GatorTrader.
 - c. *Email*- must have a .edu email
 - d. *Phone Number*- Shall be included for easy and smooth transaction between buyers and sellers.
 - e. *Agreement*- Shall agree to GatorTrader policy to complete transaction.

2. Functional Requirements

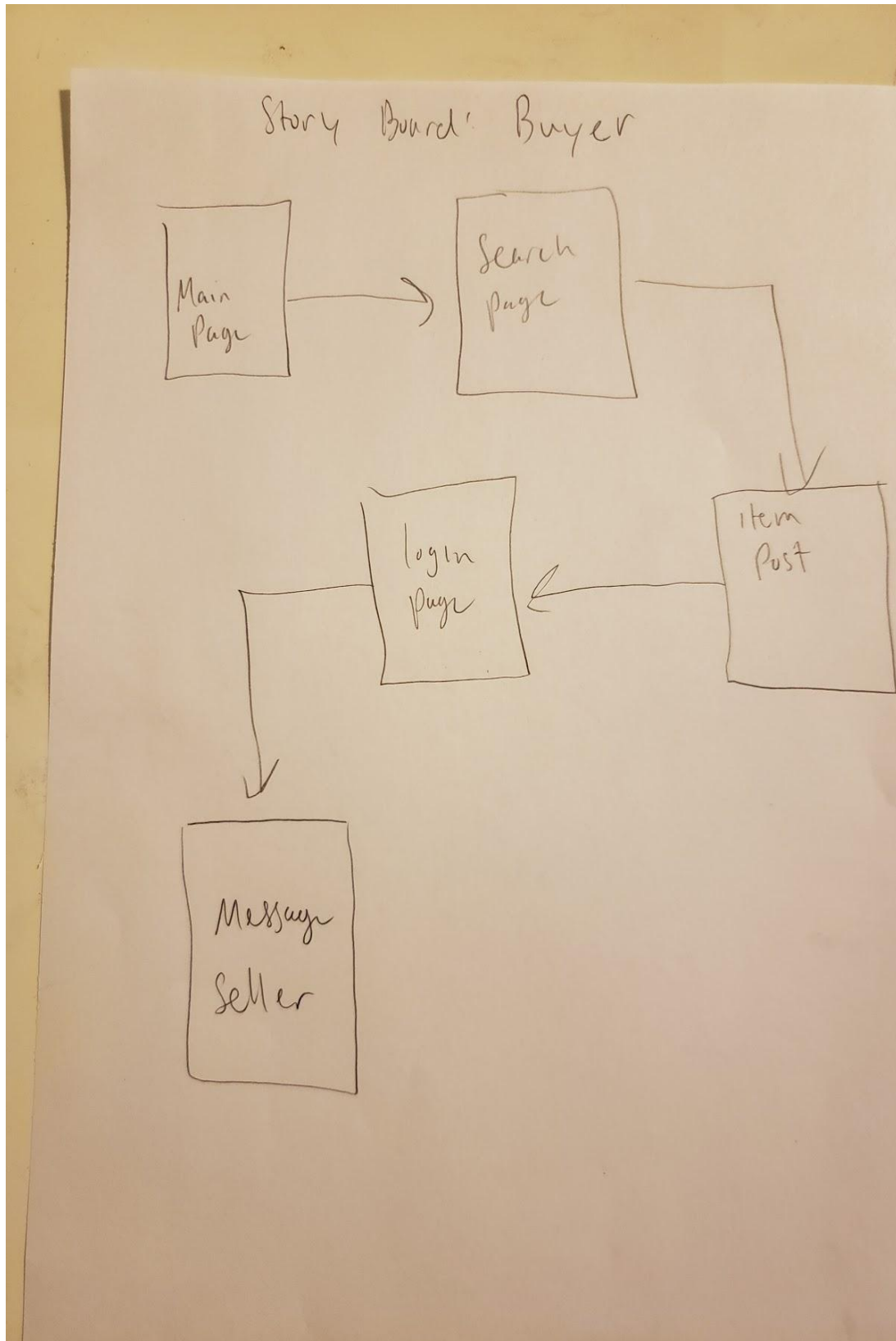
- Priority 1
 - Post items
 - Login & Registration Page
 - Search by Category
 - Admin Dashboard
 - Detailed Search Page
 - Seller Dashboard
 - Password Encryption
 - Google Analytics
 - **Home Page**
 - **Lazy Registration**
 - **Image Upload**
 - **Website Disclaimer**
 - **Messaging Page**
 -
- Priority 2
 - Search Field Validation
 - Thumbnails
 - New Tabs for Detail Pages

Features	Amazon	eBay	Craigslist	Our Future Product
Search	++	+	+	+
Selling	+	+	+	++
Delivery	++	+	+	++
Browse	+	+	+	+

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

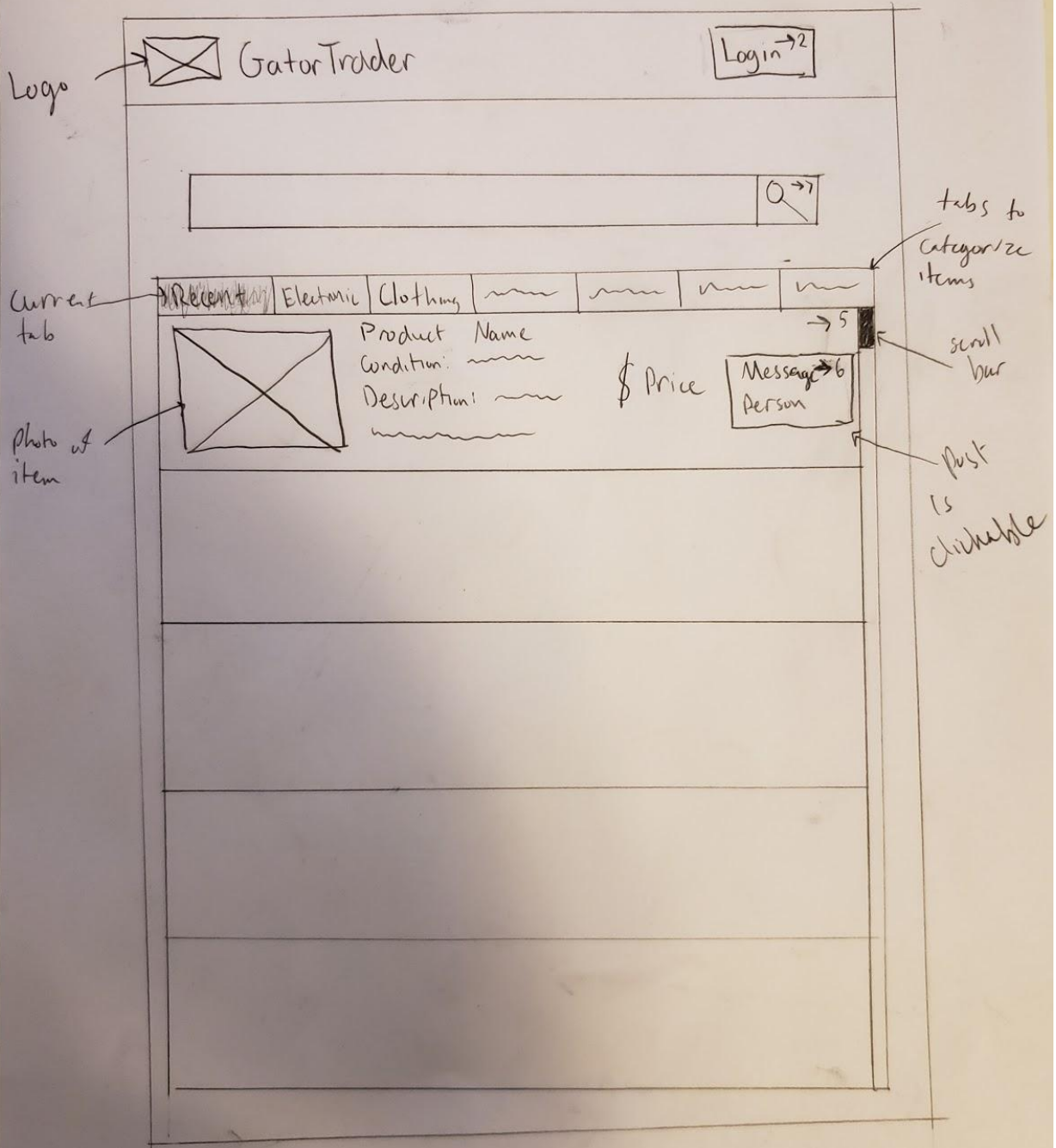
Our product will offer some of the features other sites already have like Search, Browse, Sell, etc. However, our product has an advantage when it comes to Selling and Delivery because the communication is directly between seller and customer which will give customers the advantage of asking for a price reduction, or other accommodations that our competitors will not have. In addition, our Delivery system is faster than our competitors because the product will be delivered directly to the customer by the seller which means that the product can be delivered within ours and even minutes and the customer will not have to wait the usual two days or five days shipping time like in other companies. Registration establishes a level of security and safety for both parties. An admin has the ability to disclose communication messages between both parties to the law enforcement or the party itself.

3. UI Mockups & Storyboards




Unauthorized Landing Page

1




Search Results Page

7

 Gator trader


Search Results for:

	Product Name Condition: <u> </u> Description: <u> </u>	\$ Price	<div>message <u>5</u> Person</div>

scroll
bar

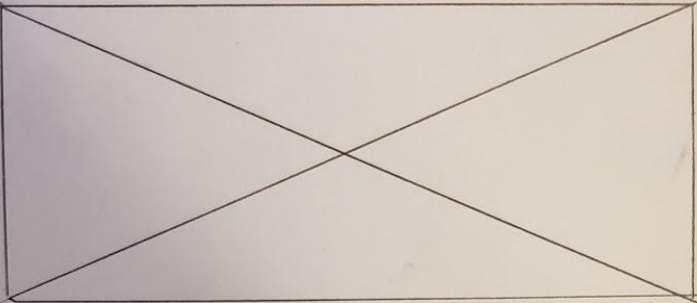
Item Post Page

5

 GatorTrader

username

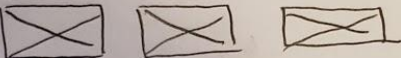
Product Name



button
to
scroll
through
photos

button
to
scroll
through
photos

Picture
gallery
of item



Description: ~~~~~
~~~~~  
~~~~~

Condition: ~~~~~

\$ Price

Message → 6
Person

Login Page

2



GatorTrader

Login

→4

→3

Messaging Page

☒ Gator Trader

username

MessagingPerson

Username: _____

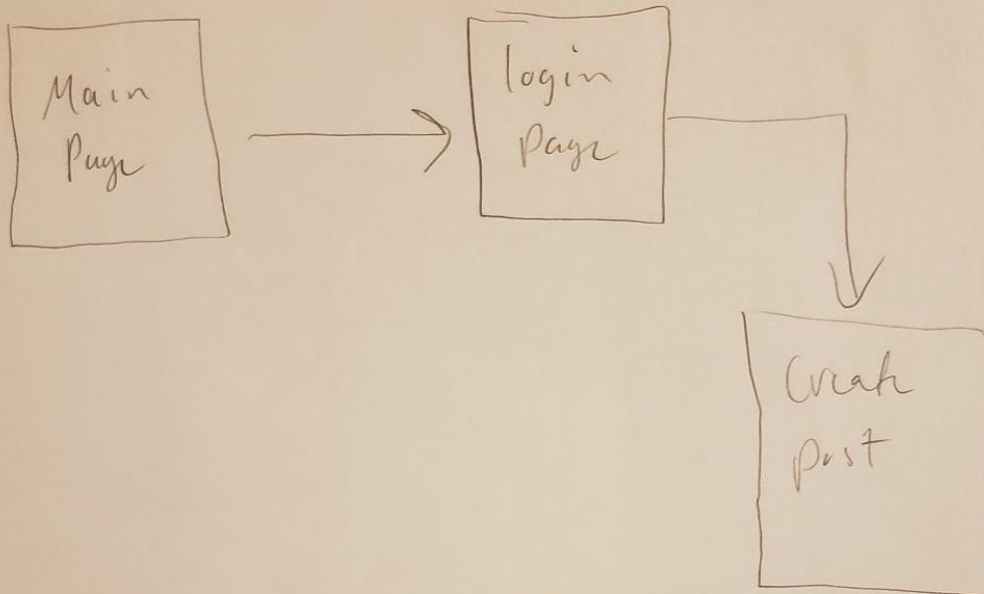
Person: _____

Person: _____

Username: _____

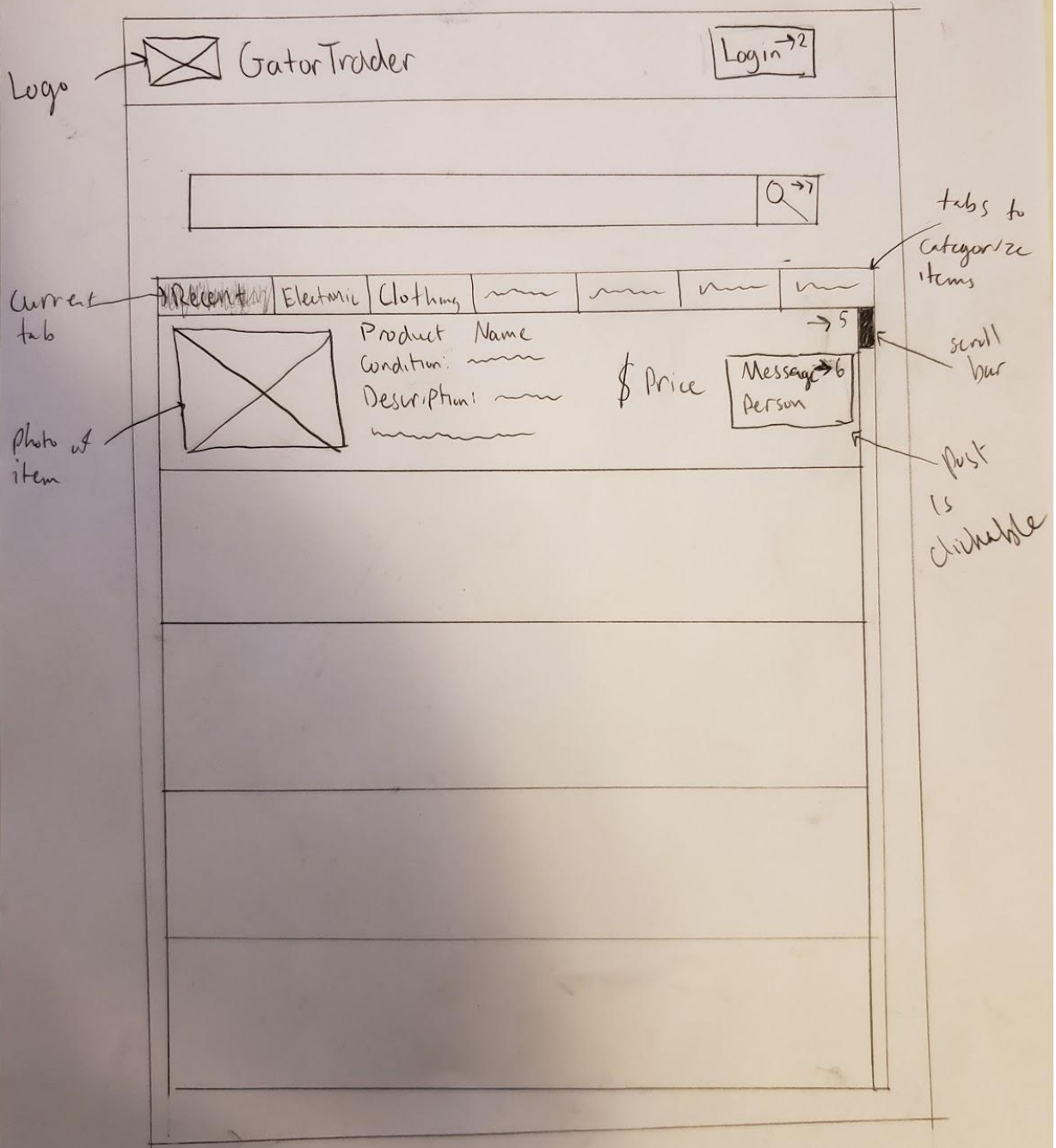
Send message button

Story Board: Seller



Unauthorized Landing Page

1



Login Page

2



GatorTrader

Login

Create Post Page

8



GatorTrade

Username

Create Post

Product name

Condition

price

Description

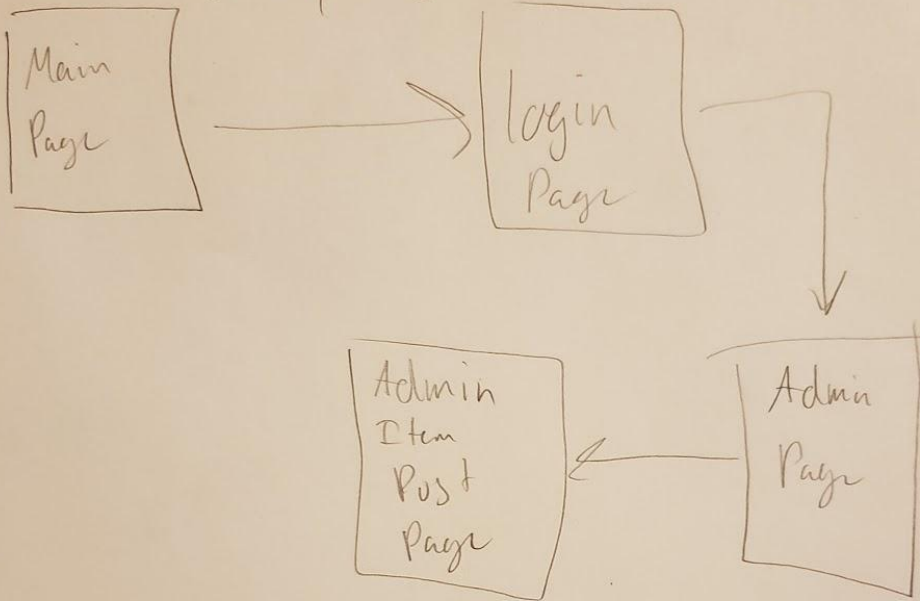
Upload
Photo

clickable

Create Post

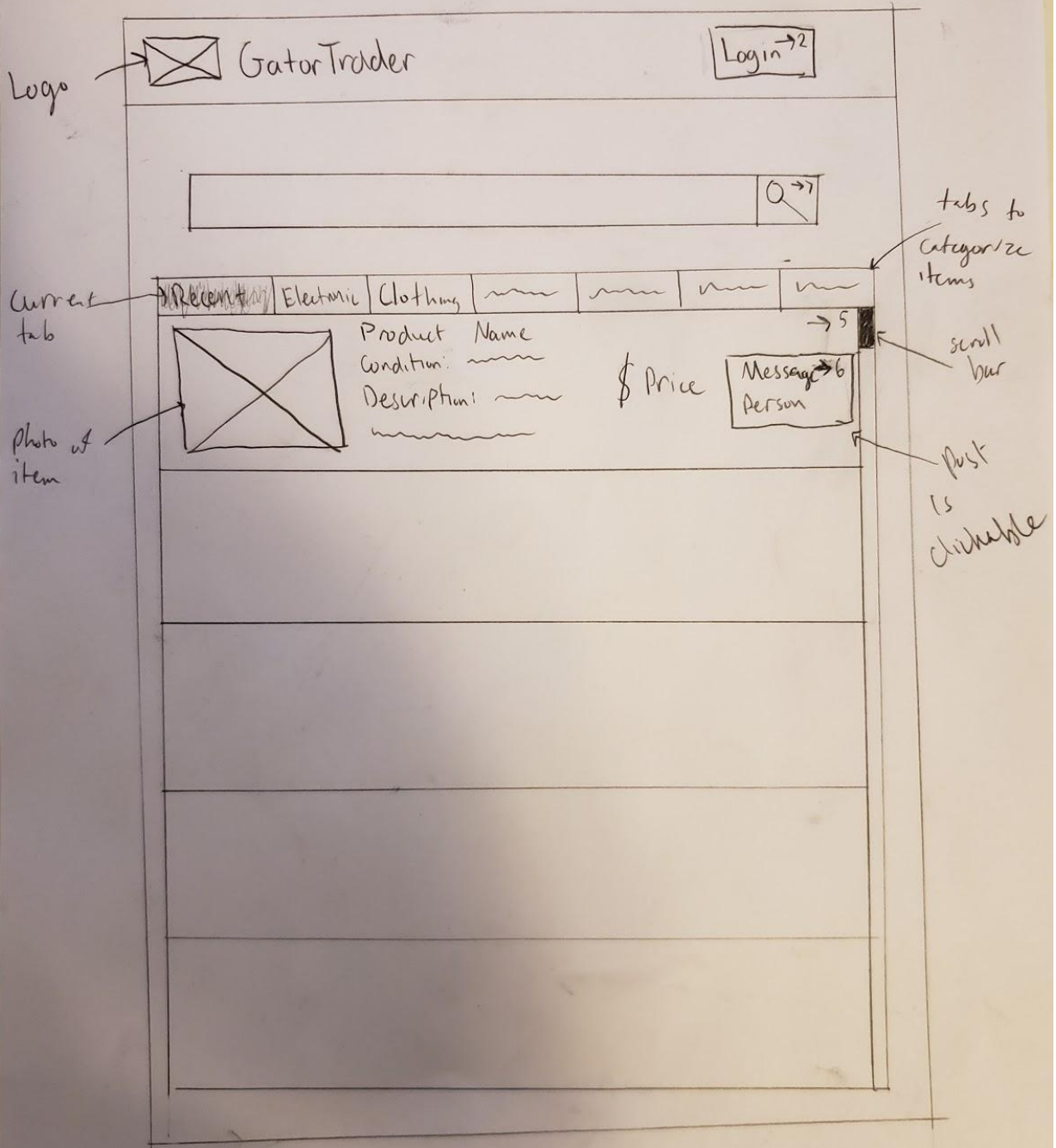
clickable

Story Board : Admin



Unauthorized Landing Page

1



Login Page

2



GatorTrader

Login

→4

→3

Admin landing Page

picture
of item

Admin Item Post Page

☐ GuterTrader

☐ Admin

Product Name

Buttons
to
scroll
through
photos

Picture
gallery

Description: ~~~~~
~~~~~  
~~~~~

Condition: ~~~~~
\$ Price

accept

reject

4. High level Architecture, Database Organization

DB organization:

All the information will be in a database named users_database which contains 3 tables in the following configuration.

Table 1: user_info : this table will be used to store all users registration information

1. user_id
2. first_Name
3. last_Name
4. student_id
5. userName
6. user_password
7. date_created

Table 2: images: table to save photos of items each user is selling.

1. item_id // this serves as a unique id for each image
2. display_name // this will be the named displayed on the page
3. item_category // this will be the category for the image
4. item_price
5. date_updated
6. item_name // this is the file name for the image
7. item_type // this represents the image extension. example .png or .IMG
8. item_size //this has the size of the image file
9. foreign_id // this will connect user to image

Table 3: user_chat: we will use this table to save messages between seller and buyer.

1. message_id_number
2. message
3. sender
4. foreign_id // this will connect user to chat

Table 4: Search by %like

1. item_id // this id will be connect to the image key to connect the mashup item to the image information.
2. item_mashup //the description title and category will be saved here as one to look for “Contains” or %like.
3. foreign_id // this will connect user to image

Media Storage:

- the images will be store in a folder in the server and then access with the information from the image table. also in some cases the image might be converted to save space

Search/filter architecture and implementation:

1. The user will be able to choose a category by name and/or input text to search for items that contain similarities (%like). for example the user will be able to just choose a category like Furniture and all items in that category will be displayed or the user can choose no category and just write a word like “bed” and it will display all item with the word bed in the title or description.

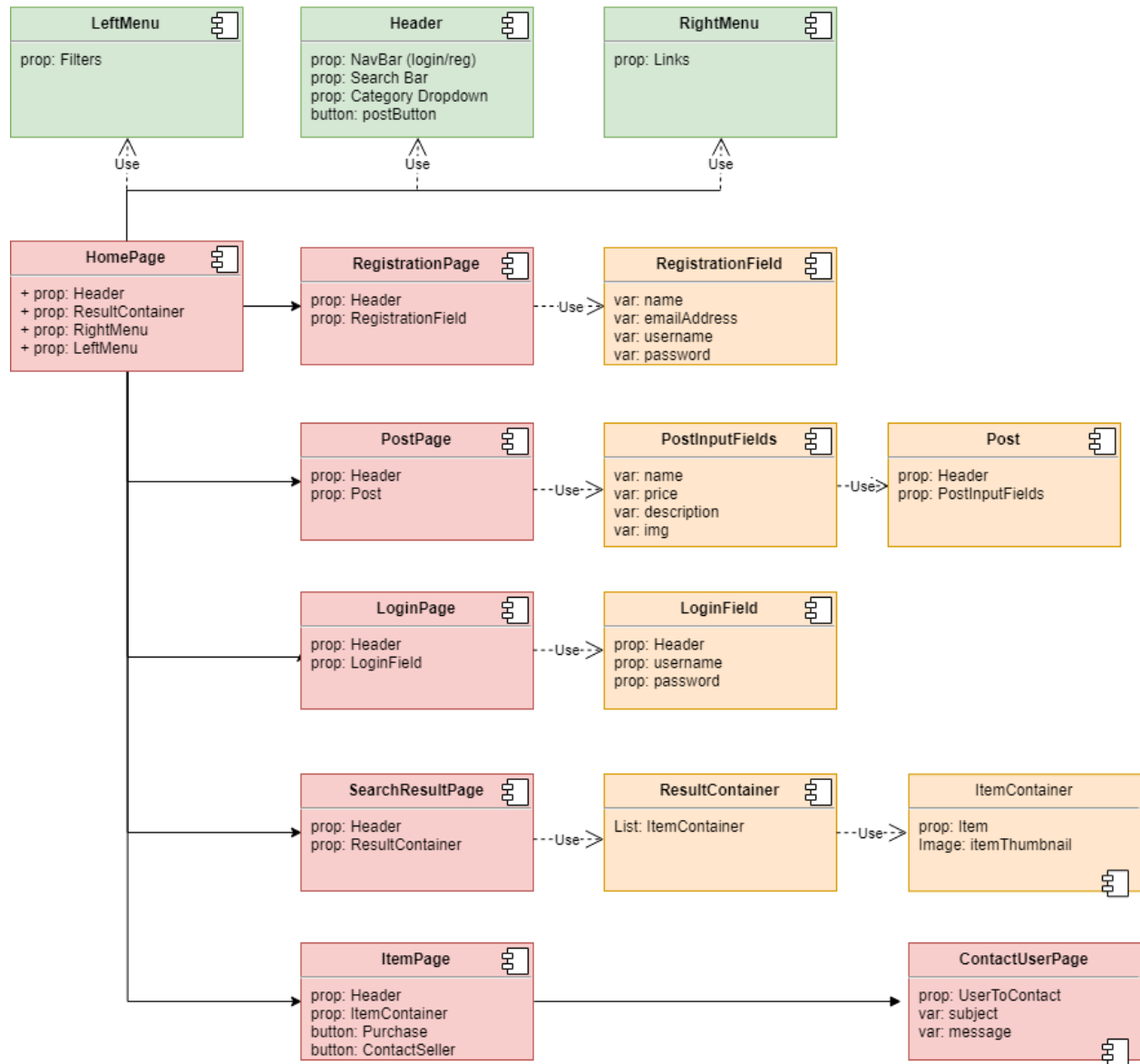
2. The user will be able to sort the search by newest to oldest, and by price.

The database will be coded & organized as follows:

The first table will contain the user registration information and it will contain an id number. The id number will auto increment for each new user. This id will also serve as a foreign key to connect images with users. the second table will contain all images from each user and it will also have an item_id to identify each picture. In addition, it will be connected with foreign keys to user_info table and Category table for easy access and retrieval. To add and get information from SQL we will use nodeAPI to connect to the database, we will use express to connect nodeAPI and the view code, and finally for our view code we will use react.

5. UML Diagram

GatorTrader Website



LEGEND:

Page linking to page →

Page

Components
used in all pages

Components used
in parent page

* Front-end framework: ReactJS

6. Key Risks

- **Scheduling Risk:** At this moment Team 01's members are unable to meet outside of class as a team. Due to scheduling conflicts it is not possible to find a time in which all members of the team can be present for planning and group coding sessions.
 - **Mitigation:** In order to solve this issue, team lead Marcus Mertilien, will rely heavily on front and back end leads to schedule meetings between their respective members. Team leads will then have a separate meeting in order to coordinate further efforts and keep the project on task.
- **Skills Risk:** Our frontend framework React poses a potential risk since it is a relatively new framework. While our project depends on a stable build, React is different in the way it interfaces with the Document Object Model (DOM) and may be a challenge for the team to implement best practices. Asynchronous function calls would be an example.
 - **Mitigation:** Team lead Marcus Mertilien, is most familiar with React, and his shared experience will help the team to overcome any potential roadblocks. React while new, is very well documented. Weekly check-ins will ensure team members stay on track.