

SW Engineering CSC648-848 Summer 2021  
dropsell.gq, Jose's Angels

Team 03

Name	Email	Roles
Mitchel Baker	<a href="mailto:mbaker3@mail.sfsu.edu">mbaker3@mail.sfsu.edu</a>	Team lead
Krina Bharatbhai Patel	<a href="mailto:kpatel11@mail.sfsu.edu">kpatel11@mail.sfsu.edu</a>	Frontend lead
Charmaine Eusebio	<a href="mailto:ceusebio1@mail.sfsu.edu">ceusebio1@mail.sfsu.edu</a>	Frontend engineer
Rowena Elaine Echevarria	<a href="mailto:rechevarria@mail.sfsu.edu">rechevarria@mail.sfsu.edu</a>	Frontend engineer
Michael Schroeder	<a href="mailto:mschroeder@mail.sfsu.edu">mschroeder@mail.sfsu.edu</a>	Backend lead, Github master
Kenneth N Chuson	<a href="mailto:kchuson@mail.sfsu.edu">kchuson@mail.sfsu.edu</a>	Backend engineer
Jamie Dominic Walker	<a href="mailto:jwalker5@mail.sfsu.edu">jwalker5@mail.sfsu.edu</a>	Backend engineer

Milestone 2  
July 8, 2021

History Table

Version	Date	Notes
M1V1	06/22/2021	
M1V2	07/02/2021	
M2V1	07/08/2021	
M2V2	07/20/2021	

## Table of Contents

<b>Data Definitions</b>	<b>3</b>
<b>Prioritized Functional Requirements</b>	<b>8</b>
Priority 1	8
Priority 2	12
Priority 3	15
<b>UI Mockups and Storyboards</b>	<b>16</b>
<b>High Level Database Architecture and Organization</b>	<b>37</b>
Entity Relationship Diagram	42
Database Model	43
<b>High Level APIs and Main Algorithms</b>	<b>45</b>
<b>High Level UML Diagrams</b>	<b>50</b>
<b>High Level Application Network and Deployment Diagrams</b>	<b>51</b>
Application Network	51
Deployment diagrams	52
<b>Identify actual key risks for your project at this time</b>	<b>53</b>
<b>Project Management</b>	<b>54</b>
<b>Detailed list of contributions</b>	<b>55</b>

## 1. Data Definitions

1. Unregistered User: A user who can visit the website, explore the products displayed in the marketplace, and checkout any public facing pages. Unregistered users need to register to use all the features of the website related to viewing auctions, commenting and rating products, messaging buyers/sellers, and purchasing products.
  - a. A general user shall be able to create an account.
  - b. A general user shall choose the option of creating an account as buyer/seller/both.
  
2. Registered User: A registered user is able to access the website with all features. Registered users need to login to the website for buying or selling products.
  - a. User login
    - i. username
      1. Must enter username to login
    - ii. password
      1. Must enter a valid password, encrypted with Bcrypt library
  
  - b. User account details
    - i. user\_id
    - ii. email
    - iii. first\_name
    - iv. last\_name
    - v. phone
    - vi. street
    - vii. city
    - viii. state
    - ix. zip
    - x. created\_at
    - xi. is\_active
    - xii. Payment details
      1. Attributes
        - a. CC number
        - b. Expiration date
        - c. 3 digit code
        - d. Zip
      2. Payment details should be handled by Stripe.js. We may store payment details in the database for multiple payment options in the future.
    - xiii. User is a buyer? Seller?
      1. is\_buyer
      2. is\_seller
  
  - c. User conversations
    - i. conversation\_id
    - ii. sending\_user\_id (FK to sending user)
    - iii. receiving\_user\_id (FK to receiving user)
    - iv. Messages
      1. message\_id
      2. conversation\_id (FK to conversation id)

- 3. message\_timestamp
- d. User meetups
  - i. meetup\_id
  - ii. buyer\_id (FK to buyer id)
  - iii. seller\_id (FK to seller id)
  - iv. meetup\_time
  - v. meetup\_location
- 3. User session
  - a. Users should have an active session created in order to keep track of the date and time they logged in. This information will be used to keep track of the length of the user's session.
  - b. When the user logs in, their session data should be updated in the database.
  - c. If the user's session has expired, then the user should be required to log in again.
  - d. If, for example, the user's session is still active and they decide to refresh the page, then the user should stay logged in.
  - e. Attributes
    - i. session\_id
    - ii. session\_expires
    - iii. session\_data
- 4. Buyers: can browse products and buy them.
  - a. shopping\_cart
    - i. shopping\_cart\_id
    - ii. buyer\_id (FK to buyer id)
    - iii. subtotal
    - iv. shopping\_cart\_products
      - 1. product\_id
      - 2. shopping\_cart\_id (FK to shopping cart id)
      - 3. title
      - 4. price
      - 5. quantity
- 5. Sellers: can upload product information and sell them.
  - a. Seller Ratings
    - i. seller\_rating\_id
    - ii. seller\_id
    - iii. seller\_rating (5-star rating system, from 1-5)
  - b. Products: The items which are uploaded by sellers, and purchased by buyers.
    - i. product\_id
    - ii. seller\_id
    - iii. title
      - 1. At Least 5 word Title or name of the product
    - iv. description
      - 1. At Least 10 word Description of the product
    - v. image
      - 1. At least 1 image with size of (600x600 or 2x2) for product visibility

- vi. price
    - 1. The price for product (For sale and auction only)
  - vii. category
- c. Product Comments
- i. product\_comment\_id
  - ii. product\_id (FK to product id)
  - iii. creator\_id (FK to user id)
  - iv. comment\_timestamp
  - v. comment
- d. Product Ratings
- i. product\_rating\_id
  - ii. product\_id (FK to product id)
  - iii. creator\_id (FK to user id)
  - iv. product\_rating (5-star rating system, from 1-5)
- e. Product Refunds
- i. product\_refund\_id
  - ii. product\_id (FK to product id)
  - iii. buyer\_id (FK to buyer id)
  - iv. seller\_id (FK to seller id)
  - v. refund\_amount
6. Auction Products
- a. product\_id
  - b. seller\_id (FK to seller id)
  - c. starting\_bid
  - d. auction\_duration
7. Top-Purchased Products
- a. product\_id
  - b. seller\_id (FK to seller id)
  - c. total\_purchased
  - d. added\_at
8. Daily Deal Products
- a. product\_id
  - b. seller\_id (FK to seller id)
  - c. deal\_duration
9. Shipping Products
- a. product\_id
  - b. buyer\_id (FK to buyer id)
  - c. seller\_id (FK to seller id)
  - d. shipping\_from
  - e. shipping\_to
  - f. transaction\_total

## 10. Redux related data definitions

### a. Login

#### i. Actions

1. setUsername(username)
  - a. Action type: 'USER\_SET\_USERNAME'
2. setPassword(password)
  - a. Action type: 'USER\_SET\_PASSWORD'
3. loginUser()
  - a. userData(username, password)
4. redirectUserAfterLogin(loggedIn)
  - a. Action type: 'USER\_IS\_LOGGEDIN'

#### ii. Reducer

1. INITIAL\_LOGIN\_STATE
  - a. Username
  - b. Password
  - c. loggedIn

### b. Register

#### i. Actions

1. setUsername(username)
  - a. Action type: 'USER\_SET\_USERNAME'
2. setPassword(password)
  - a. Action type: 'USER\_SET\_PASSWORD'
3. setConfirmPassword(confirmPassword)
  - a. Action type: 'USER\_SET\_CONFIRM\_PASSWORD'
4. createUser()
  - a. userData(username, password, confirmPassword)
5. redirectUser(registered)
  - a. Action type: 'USER\_IS\_REGISTERED'

#### ii. Reducer

1. INITIAL\_REGISTER\_STATE
  - a. Username
  - b. Password
  - c. confirmPassword
  - d. Registered

### c. Products

#### i. Actions

1. setTitle(title)
  - a. Action type: 'PRODUCT\_SET\_TITLE'
2. setDescription(description)
  - a. Action type: 'PRODUCT\_SET\_DESCRIPTION'
3. setPrice(price)
  - a. Action type: 'PRODUCT\_SET\_PRICE'
4. setImage(image)
  - a. Action type: 'PRODUCT\_SET\_IMAGE'
5. setSuccess(isSuccess)
  - a. Action type: 'PRODUCT\_SET\_SUCCESS'
6. setCategory(category)

- a. Action type: 'PRODUCT\_SET\_CATEGORY'
  - 7. setCategories(categories)
    - a. Action type: 'SET\_CATEGORIES'
  - 8. changeDropdownText(text)
    - a. Action type: 'CHANGE\_DROPDOWN\_TEXT'
  - 9. createProduct()
    - a. formData(title, description, price, category, file)
  - 10. getProducts(products)
    - a. Action type: 'GET\_PRODUCTS'
- ii. Reducer
- 1. INITIAL\_PRODUCT\_STATE
    - a. Title
    - b. Description
    - c. Price
    - d. Category
    - e. File
    - f. filePreview
    - g. isSuccess
    - h. Products
    - i. Categories
    - j. dropdownText
11. Search data definitions
- a. Query
  - b. searchQuery/setSearchQuery()
  - c. filterProducts(products, query)

## 2. Prioritized Functional Requirements

### Priority 1

#### Marketplace

1. Unregistered and registered users shall be able to query the database for products by interacting with a search bar.
2. Sellers shall be able to put their products up for auction or list them in the marketplace
3. Buyers shall be able to propose buy it now or best offer options for a seller's product
4. Users should be offered a base price for automatic reject or accept
5. Buyers shall be able to access the location information of sellers through an interactive map
6. Unregistered and registered users shall be shown ads of products similar to their search histories
7. Buyers shall be given a clear breakdown of shipping options to choose from
8. Buyers shall be able to check their account settings to determine if a product they've purchased has been confirmed, processed, shipped, or returned
9. Buyers shall be given a tracking number to stay up to date with their product's delivery status
10. Products shall be displayed as out of stock if the supply has run out
11. Buyers shall be contacted by email if they opt to be notified when a product comes back in stock

#### Website Features

12. Sellers shall be provided with consignment operations to get their products delivered to buyers
13. Buyers shall be given a price matching tool, which compares products either already posted on our app or compares other products from other websites (amazon)
14. Buyers shall be provided with daily deals on the home page
15. Users shall be able to zoom in or out of product images with a magnifying glass feature
16. The checkout page shall calculate tax automatically based on the buyer's location

#### Buyers

17. Buyers shall sign a purchase agreement; they must agree to conduct business according to DropSell's rules prior to using the marketplace
18. Buyers shall be able to send inquiries of interest to a product's seller
19. Buyers shall save products they wish to purchase by adding them into a shopping cart
20. Buyers shall be able to remove an item from their shopping cart and the total list of products should be updated accordingly

21. Buyers shall be able to return to the main shopping menu to look for other products if they are not finished shopping
22. Buyers shall have the option of canceling or modifying their order
23. Buyers shall receive a detailed receipt of their purchase through email after checking out
24. Buyers shall receive shipping information through email if their purchase involves shipping products to their specified address
25. Buyers shall be able to add their full name, email, phone number, and delivery address if they haven't inputted this data prior to checking out
26. Buyers shall have the option of choosing an existing payment option or adding a new one
27. Buyers shall be notified by final invoice of their purchase's pickup/delivery time, expected time of arrival, name of seller, location to meet them at, their contact info, and the grand total to pay
28. Buyers shall be able to rate and star products
29. Buyers shall be able to interact with all of their starred products in their buyer settings
30. Buyers shall be able to choose from various UPS delivery methods
31. Buyers shall be able to view all data related to a seller's review page
32. Products shall have their reviews displayed when listed in the marketplace
33. Buyers shall be permitted to submit reviews of products they purchase
34. Buyers shall add auction products to their watch list to keep track of the auction's most recent updates
35. Top-rated products shall be displayed to buyers after they purchase a similar product

## **Sellers**

36. Sellers shall be able to create new products with a title, description, price, category, and image
37. Sellers shall be able to edit the title, description, price, category, and image of their products
38. Sellers shall sign a contract before being granted the privilege of posting products on DropSell
39. Sellers shall agree to a small listing fee for each product they sell
40. Sellers shall be able to adjust the quantity of a product they've listed on DropSell
41. Sellers shall be able to delete products they've listed
42. Sellers shall have the option of selecting multiple products to delete
43. Sellers shall be shown all their listed products under the seller settings section of their profile
44. Sellers shall have the option of relisting their products for sale
45. Sellers shall get an email confirmation after one of their products has sold

## **Auction**

46. Buyers shall be able to filter the products they are searching for, based on minimum/maximum price, location filtering, or type of shipping (pickup/delivery)

47. Sellers shall have full control over their auction settings, from choosing the average starting bid, the time when the auction begins, to the total duration of the auction
48. Buyers shall be displayed the current price of an auction, the amount of time remaining, and any bids from other buyers
49. Buyers shall be able to bid on products with one click.
50. Buyers and sellers shall see a live countdown of the auction's remaining time
51. Sellers shall be able to set the duration of their auction for as long as 30 days or as short as 1 hour
52. Buyers shall not be able to retract a bid they place on a product
53. Buyers shall be notified immediately by email that they've won an auction

### **Messaging**

54. Buyers shall have the ability to contact sellers directly through the DropSell website
55. Buyers shall have the ability to contact sellers via email or phone, if they choose to do so

## Priority 2

### Marketplace

56. Buyers shall be given an advanced search bar for detailed searches, including product sizes, price ranges, and colors
57. Sellers shall be able to publish different styles of their products, such as different colors or fabrics
58. Buyers and sellers shall be able to schedule and edit meetup times
59. Buyers shall be refunded in full if they purchase multiple products which causes the product to be sold out
60. Unregistered and registered users shall be able to share products with friends through email, social media, or a shareable link
61. Registered users shall have the option of flagging products as inappropriate, a scam, or submit a ticket to the Dropsell team for further review
62. Unregistered and registered users shall be able to go back to products they have viewed previously on their user feed
63. The marketplace shall show the maximum time length for shipping a product
64. The marketplace shall show the minimum time length for shipping a product
65. The marketplace shall perform monitoring on unregistered and registered users
66. The marketplace shall keep track of the ID's of posts unregistered and registered users click on which will be saved in the database under most-interacted products
67. The marketplace shall contain a currency converter for entering international markets

### Website Features

68. Sellers shall be provided with a daily product forecast
69. Sellers shall be provided with a profitability test for determining how much it costs to ship a product on Dropsell as opposed to other ecommerce services
70. Sellers shall be provided with an algorithm which keeps track of how many interactions their products have
71. Products shall contain hashtags to help with categorization
72. Unregistered and registered users shall be able to search products using hashtags
73. Buyers shall be shown products which are similar in price or category to the product they are currently viewing
74. Buyers shall be shown comparisons between product specifications
75. Sellers shall be able to see statistics related to how many buyers have purchased their products
76. Sellers shall be able to keep track of the median price ranges of products they are viewing

## **Buyers**

77. Buyers shall be able to click on product listing images to zoom in and see more details
78. Buyers shall be provided with a list of their purchase histories
79. Buyers shall be able to save products they'd like to buy in the future to a wishlist
80. Buyers shall be rewarded with discount codes if they are frequent buyers
81. Buyers shall be provided with a promotion code box for entering their discount code when checking out
82. Buyers shall be able to generate a referral URL which they can send to friends
83. Buyers shall be able to subscribe to a specific seller so they can be notified when a product's price is updated, the product has been restocked, or if the product has been removed
84. Buyers shall be notified about products they have viewed previously
85. Buyers shall be asked if they are still interested in the products they've previously viewed

## **Sellers**

86. Sellers shall allow their data such as user details, products, profile picture, to be publicly displayed to buyers
87. Sellers shall be able to advertise their products to specific buyers with a send offer feature
88. Products shall have a review page where buyers send feedback about its quality
89. Sellers shall have a review page where buyers can provide structured feedback
90. Sellers shall have a rating system based on a five-star system
91. Sellers shall be required to display "illegal item" on illegal products
92. Sellers shall be required to explain their illegal item and their reasons for listing it
93. Seller products shall be marked as out of stock if its supply runs out
94. Seller products shall be marked as "Last 1 Available" so buyers know that it is the last product remaining
95. Sellers shall participate in a striking system if they choose to perform misconduct or violate the terms of service.
96. Sellers shall have 3 strikes before their account faces a possible suspension or blacklist

## **Auction**

97. Buyers shall bid on products as many times as possible before the remaining time runs out
98. Buyers shall be able to keep track of auction statistics
99. Buyers shall be able to see how many other buyers are bidding on the same product
100. Buyers shall be notified with a 5 minute warning before an auction finishes
101. Buyers shall be notified when the auction finishes
102. Buyers shall be notified after they win an auction

103. Buyers shall have the option of choosing from multiple payment options, such as Paypal, ApplePay, or GooglePlay

## Priority 3

### Marketplace

104. Sellers shall be able to create a wedding registry with a product wishlist for their special day
105. Buyers shall be shown other buyers who have made the same purchases
106. Buyers and sellers shall have the time length of shipping a product hidden if their location is closer than 1 or 2 miles

### Buyers

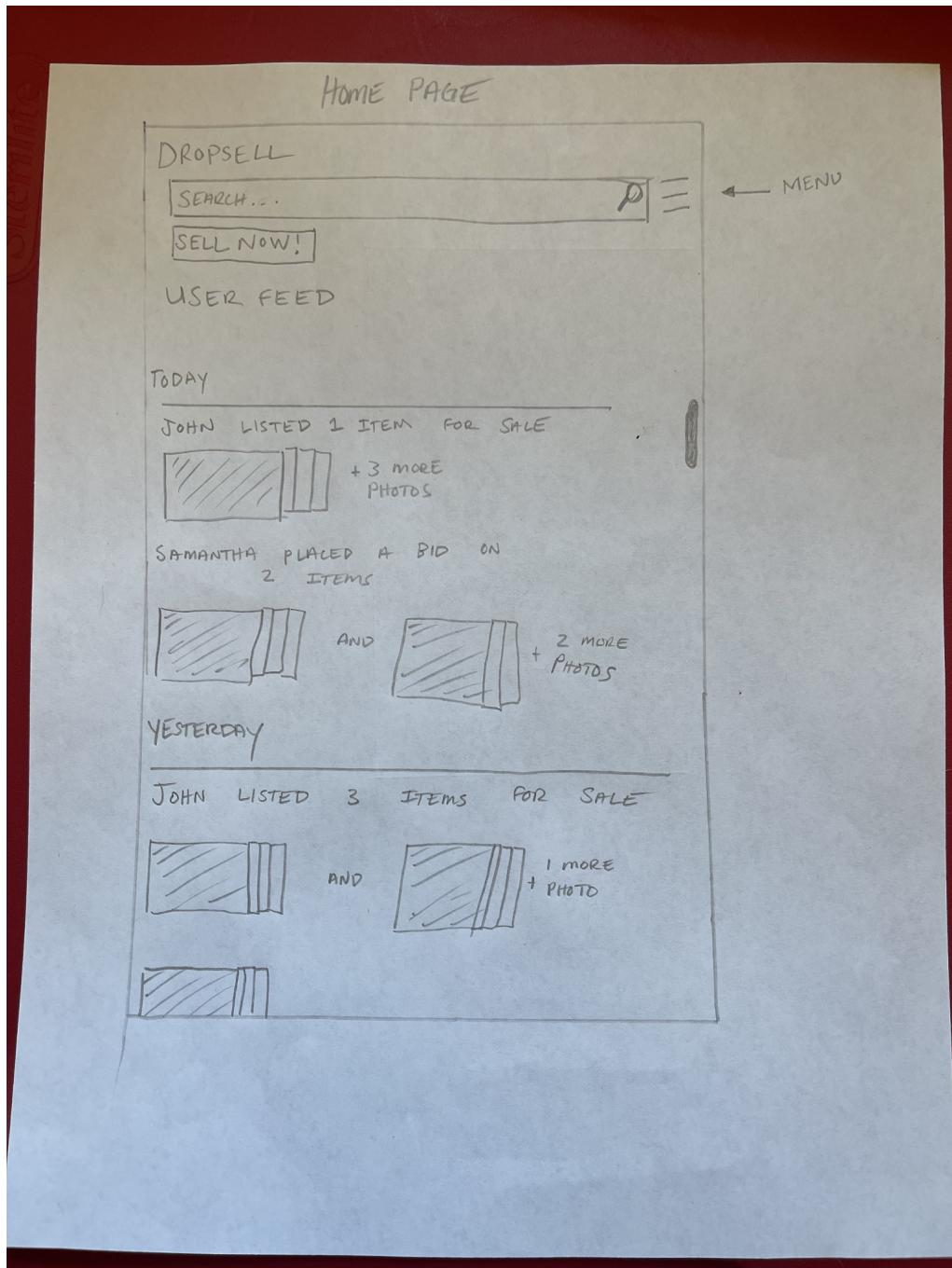
107. Buyers shall be awarded with a random product if they purchase products deemed as lucky
108. Buyers shall be rewarded with a promotion code or a random product for their birthday

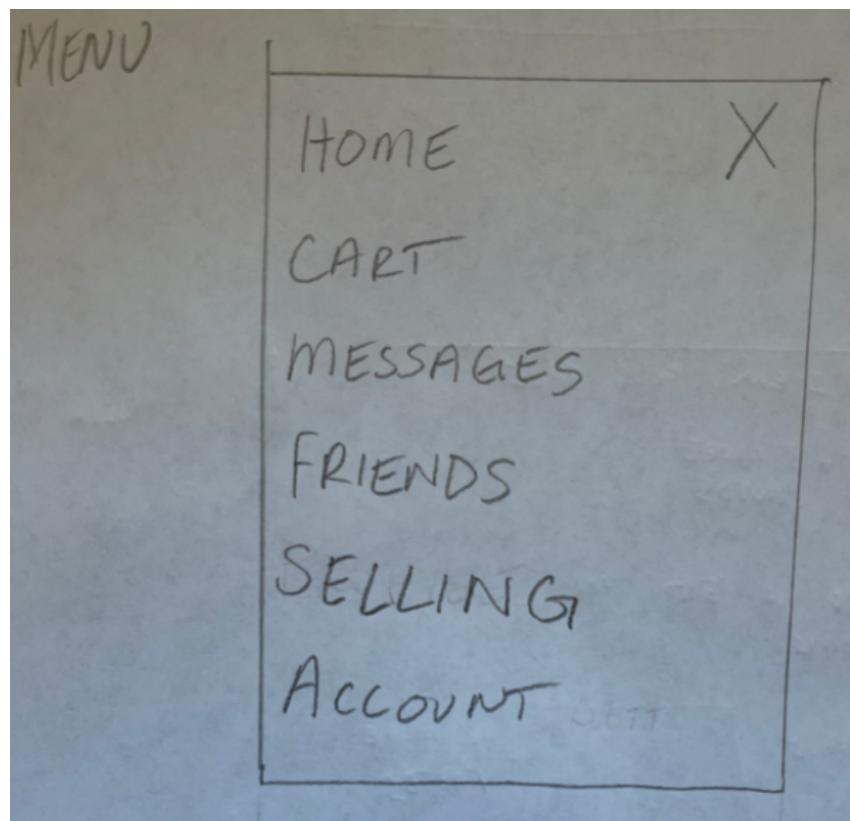
### Sellers

109. Sellers shall have background checks if they intend to sell on DropSell
110. Sellers shall have their products hide ratings if they have 5 stars

### 3. UI Mockups and Storyboards

#### Home Page



**Menu**

## About Page

### About DropSell

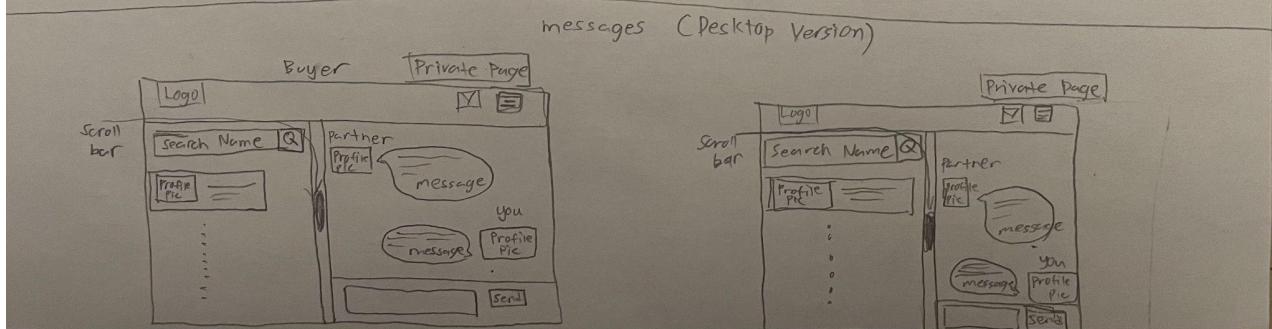
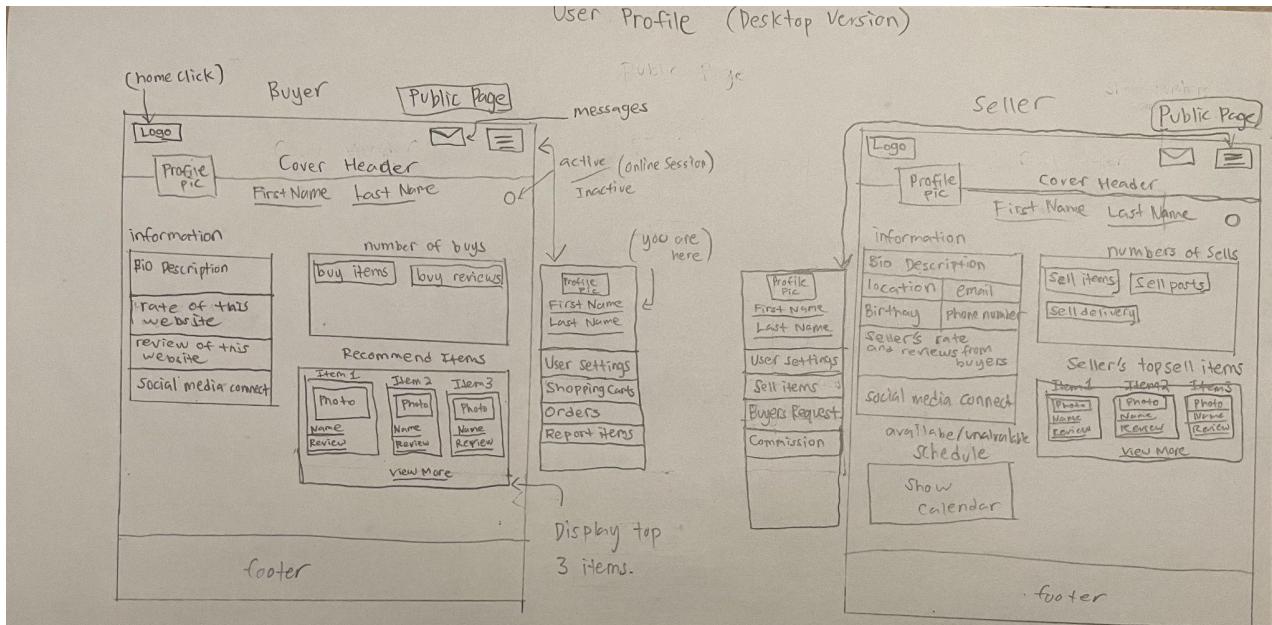
DropSell is a new digital marketplace created by seven senior students at San Francisco State University. We focus on our customers' safe and secure selling and buying experience.

DropSell provides all sellers and buyers to make transactions locally and globally, with little to no fees.

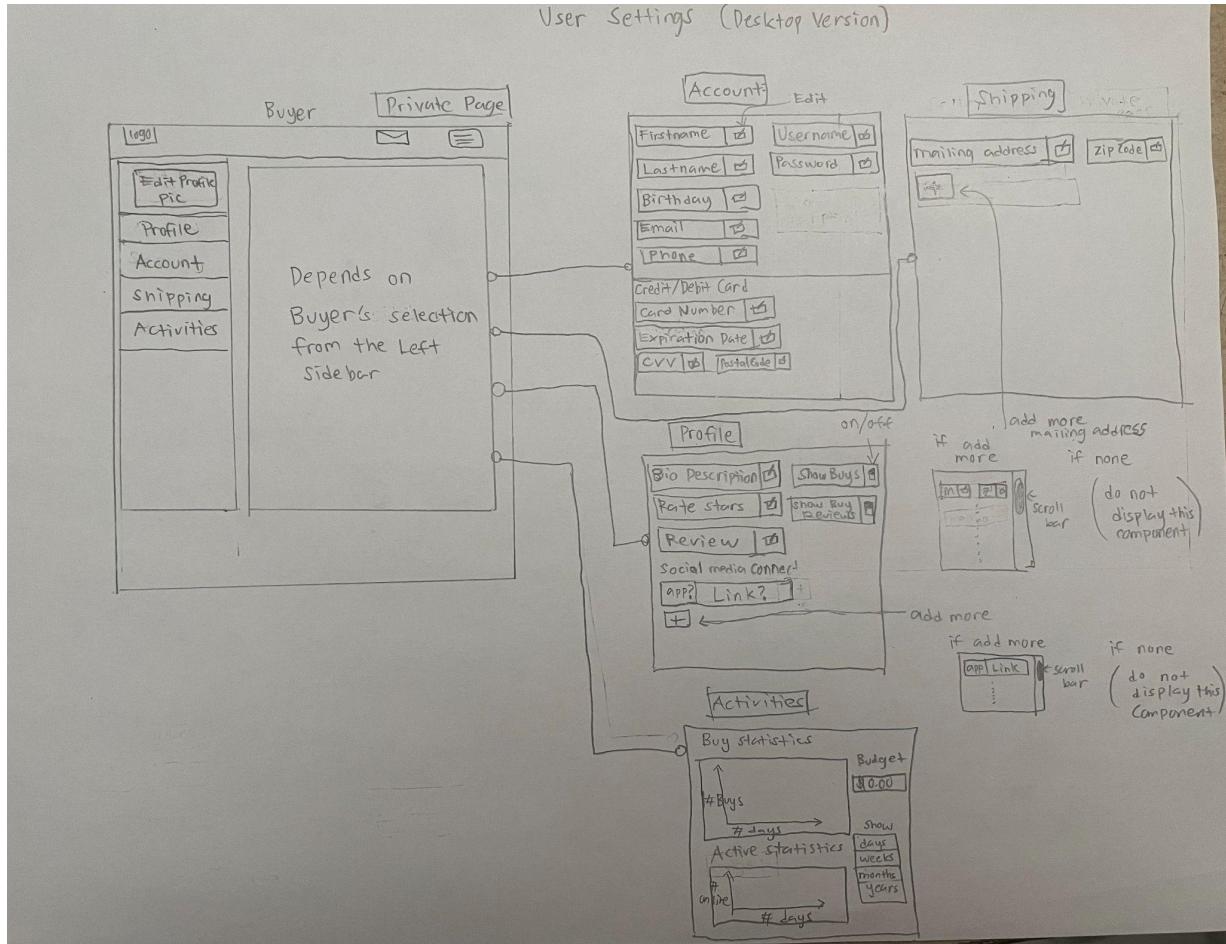
## User Sign up / Login

<p><b>Log In!</b> or <u>create an account</u></p> <p><input type="text"/> Email or Username</p> <p><input type="password"/> Password</p> <p><input type="button" value="Log In"/></p> <p>or</p> <p><input checked="" type="checkbox"/> Continue with Facebook</p> <p><input checked="" type="checkbox"/> Continue with Google</p> <p><input checked="" type="checkbox"/> Continue with Apple</p>	<p><b>Sign Up!</b></p> <p><input type="radio"/> Buyer   <input type="radio"/> Seller   <input type="radio"/> Both</p> <p><input type="text"/> First Name   <input type="text"/> Last Name</p> <p><input type="text"/> Email</p> <p><input type="password"/> Password <input type="checkbox"/> Show</p> <p><input type="text"/> Confirm Password</p> <p>(if signing up as Seller/Both)</p> <p><input type="text"/> Driving License ID</p> <p><input type="button" value="Sign Me Up"/></p> <p>or</p> <p><input checked="" type="checkbox"/> Continue with Facebook</p> <p><input checked="" type="checkbox"/> Continue with Google</p> <p><input checked="" type="checkbox"/> Continue with Apple</p>
--	--

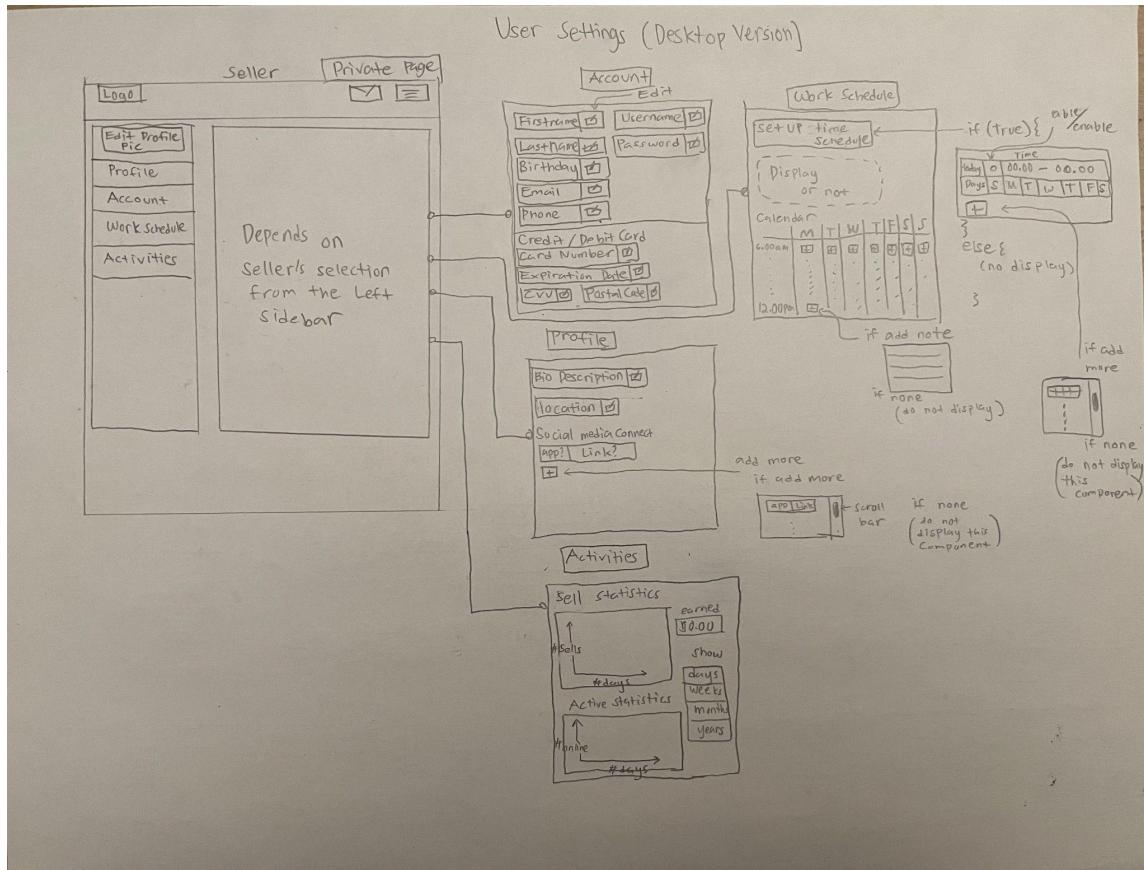
## User Profile



## Buyers Settings



## Sellers Settings



### Starred/Watching Product Listing, Starred/Watching Product List

STARRED / WATCHING PRODUCT LISTING

BNIB WIRELESS DUALSHOCK PS4 CONTROLLER



CONDITION: NEW  
QUANTITY: 3  
PRICE: \$40.00

[BUY IT NOW](#)  
[ADD TO CART](#)  
[WATCHING](#)

SHIPPING: FREE  
DELIVERY: JUL 12 - JUL 20  
PAYMENTS: PAYPAL  
SELLER: JANEDOE  
[CONTACT SELLER](#)

STARRED / WATCHING PRODUCT LIST

WATCHING	
BNIB WIRELESS DUALSHOCK PS4 CONTROLLER	<a href="#">UNWATCH</a>
MINT CONDITION 1ST GEN CHARIZARD	<a href="#">UNWATCH</a>
PRE-OWNED DOMINION BASE DECK	<a href="#">UNWATCH</a>
USED SAMURAI CHAMPOO DVD BOX SET	<a href="#">UNWATCH</a>
NEW JEFFREY CAMPBELL LITA SIZE 7	<a href="#">UNWATCH</a>

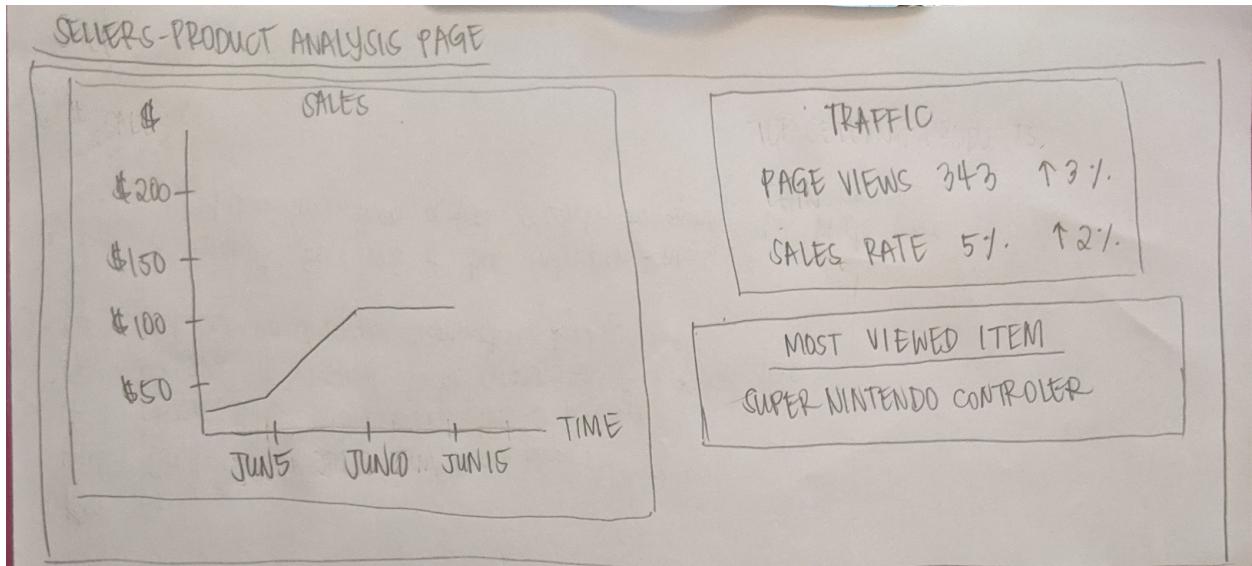
### List of Auctions and Products for Sale

CURRENT AUCTIONS BUYERS ARE PARTICIPATING IN		DAY / HOUR / MIN / SEC
⊕ YOUR BIDS		
JEFFREE STAR BLUT BLUV PALETTE		TIME LEFT: 00:00:02:46
HERMÈS BIRKIN 35 HAND BAG		TIME LEFT: 00:04:03:17
BNIB TAMAGOTCHI WONDER GARDEN		TIME LEFT: 00:08:01:09
NEW TEKKEN 7 FOR PS4		TIME LEFT: 02:00:00:00

SELLERS - LIST OF PRODUCTS FOR SALE	
⌚ SELLING	
PHOTO	TITLE
<input checked="" type="checkbox"/>	PRE-OWNED IPAD MINI 4TH GEN
	PRE-OWNED RAZER BLADE 15 LAPTOP
	NEW LULULEMON ALIGN LEGGINGS S
	NEW NARS IGNITED EYESHADOW PALETTE
FORMAT	PRICE
AUCTION	\$100
AUCTION	\$1000
AUCTION	\$40
BUY-IT-NOW	\$35

## Seller Analytics and Create Product Listing Page



SELLER - CREATE NEW PRODUCT LISTING

CREATE LISTING

**DETAILS**

TITLE: [ ]

CATEGORY: [ ] ▾ ▾

CONDITION: [ ] ▾

FORMAT: [ ] ▾

PRICE: [ \$ ]

QUANTITY: [ ]

DESCRIPTION: [ ]

SHIPPING: [ ] ▾

**FEES:**

wavy lines

**BOOST YOUR LISTING**

[ ]

[ ]

[ ]

[ ]

[ ]

[ ]

## User Checkout Experience

### Receipt Info

1.) RECEIPT INFO

APP LOGO

ENTER INFO FOR RECEIPT DETAILS

First Name  Last Name

Phone Number

Email

Pickup  Delivered

IF PICKUP

Pickup <product-name> from  
<Seller - location>

USE EXISTING DELIVERY ADDRESS

ADD A NEW DELIVERY LOCATION

DELIVERY ADDRESS

ADDRESS  CITY

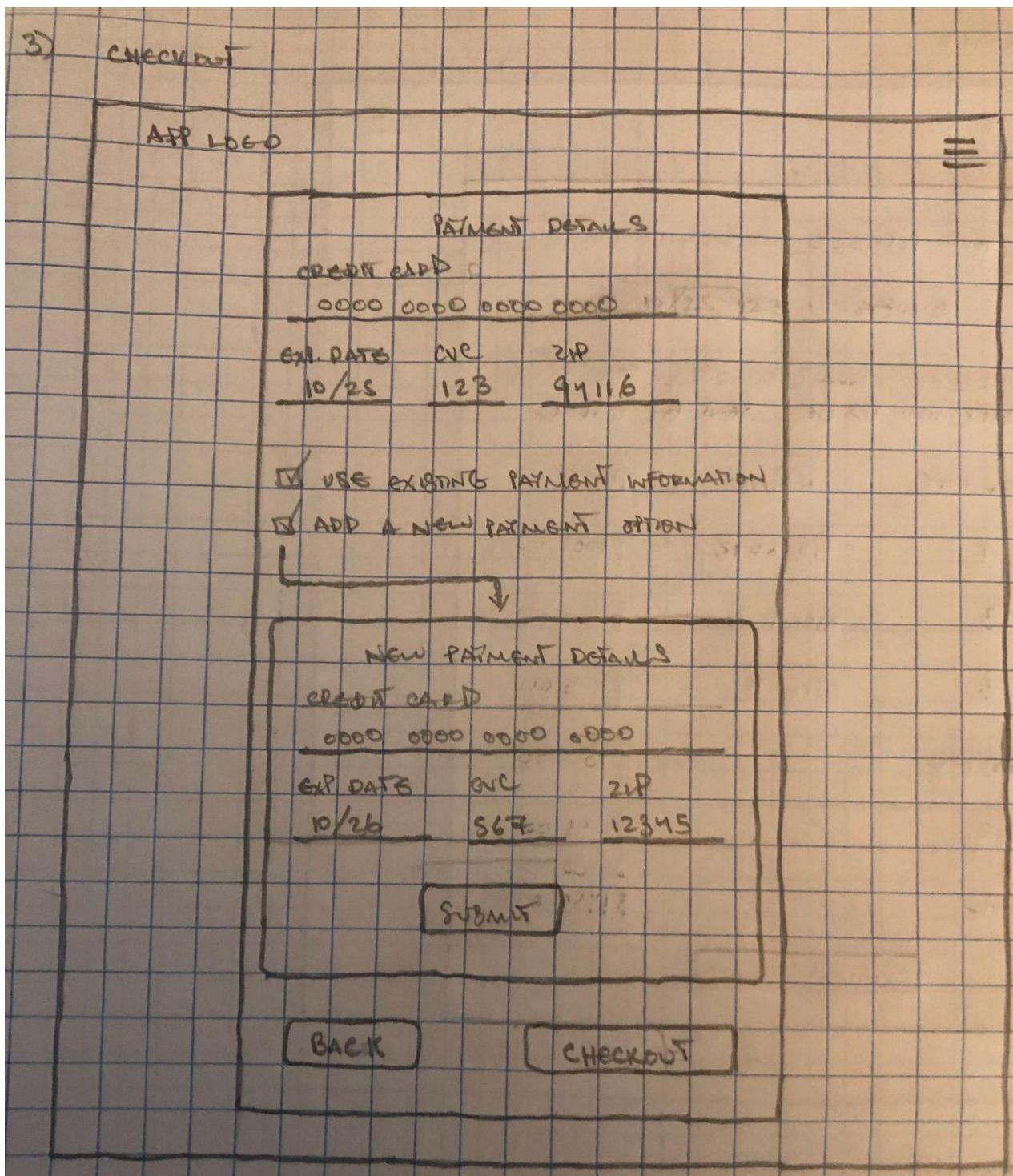
STATE  ZIP  UNIT

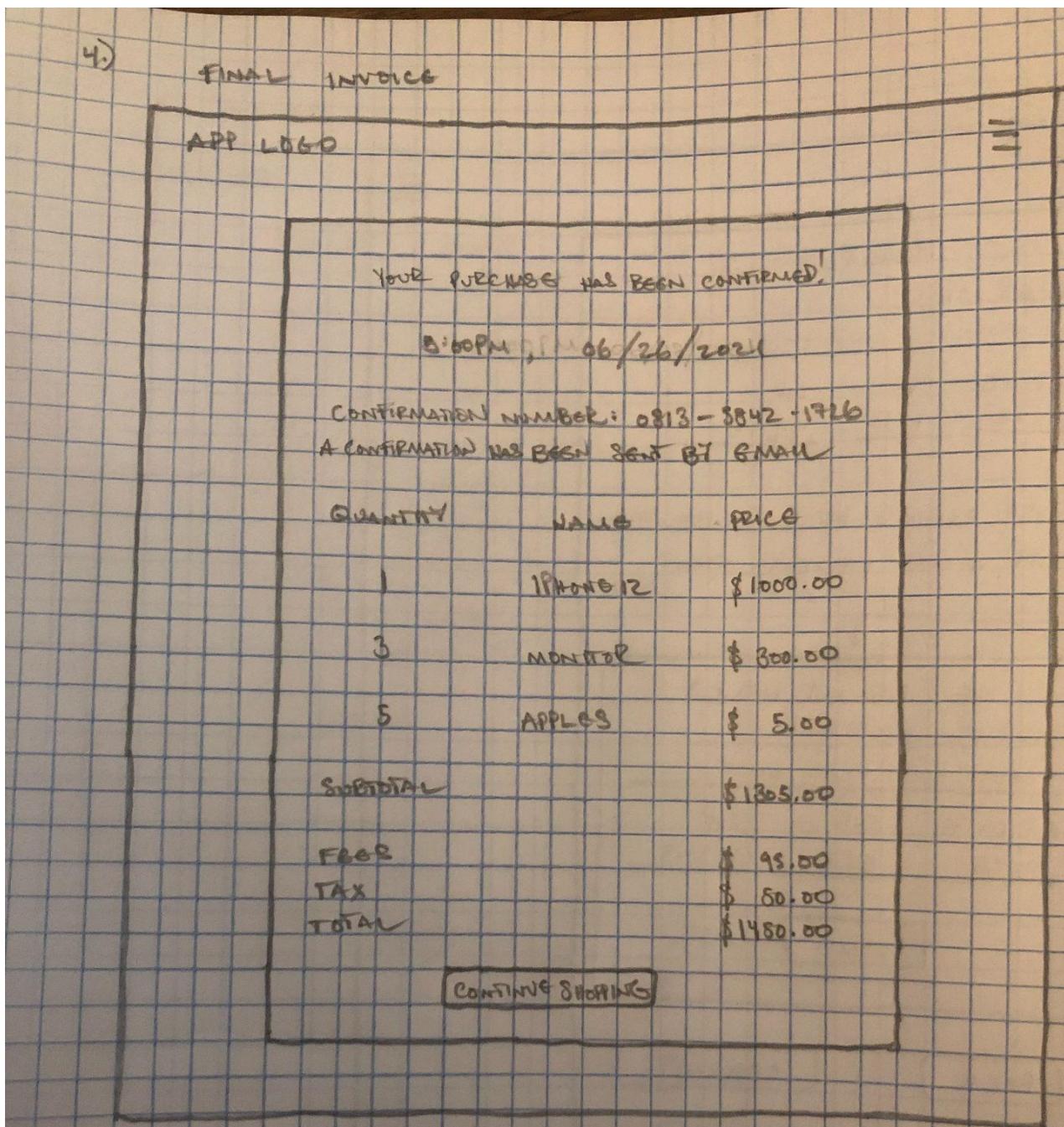
MODIFY ORDER  CONTINUE

- IF USER HAS NOT YET FILLED OUT THEIR RECEIPT INFO, THEN GATHER THIS FROM THEM FOR FUTURE USE
- IF A BUYER HAS PURCHASED BEFORE, AUTOMATICALLY FILL THEIR INFORMATION
- DELIVERY OPTIONS EXPANDED IF DELIVERY CHECKBOX IS CHOSEN.
- IF THEY'VE ADDED THEIR ADDRESS BEFORE, AUTOMATICALLY FILL IT IN WITH THEIR EXISTING INFO

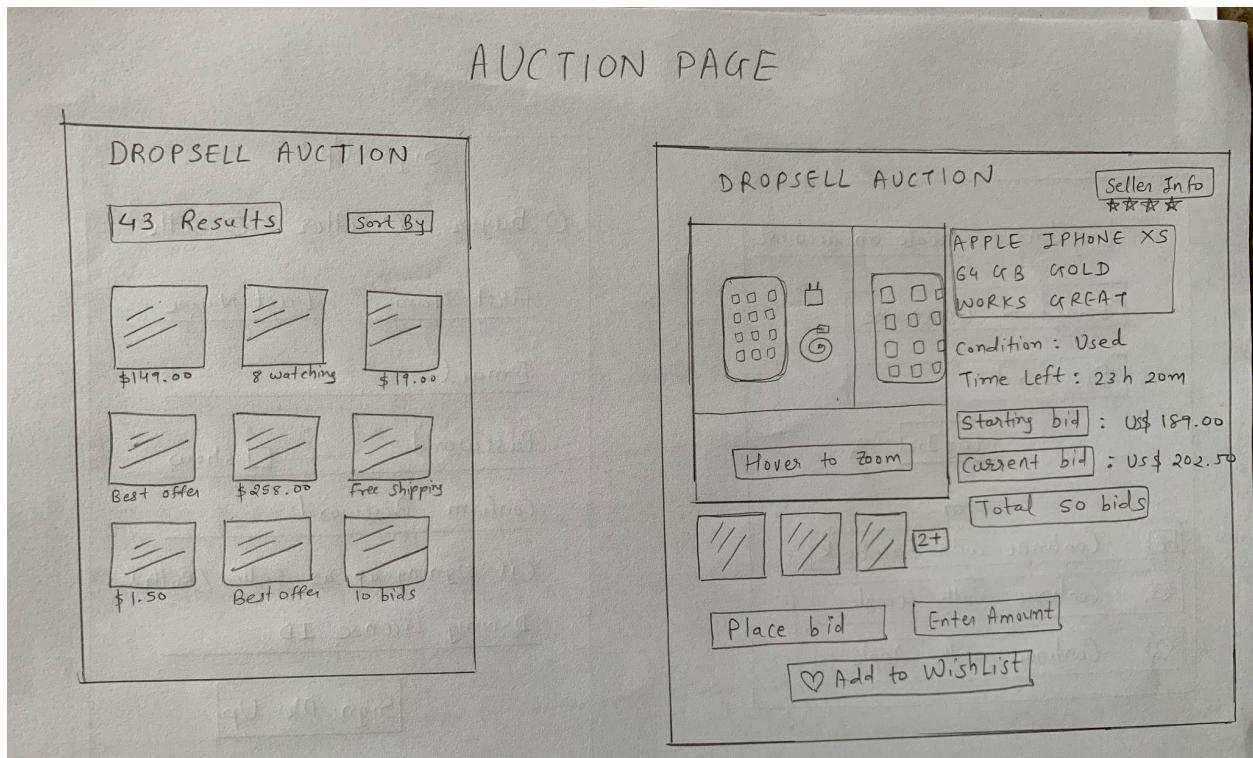
## Summary

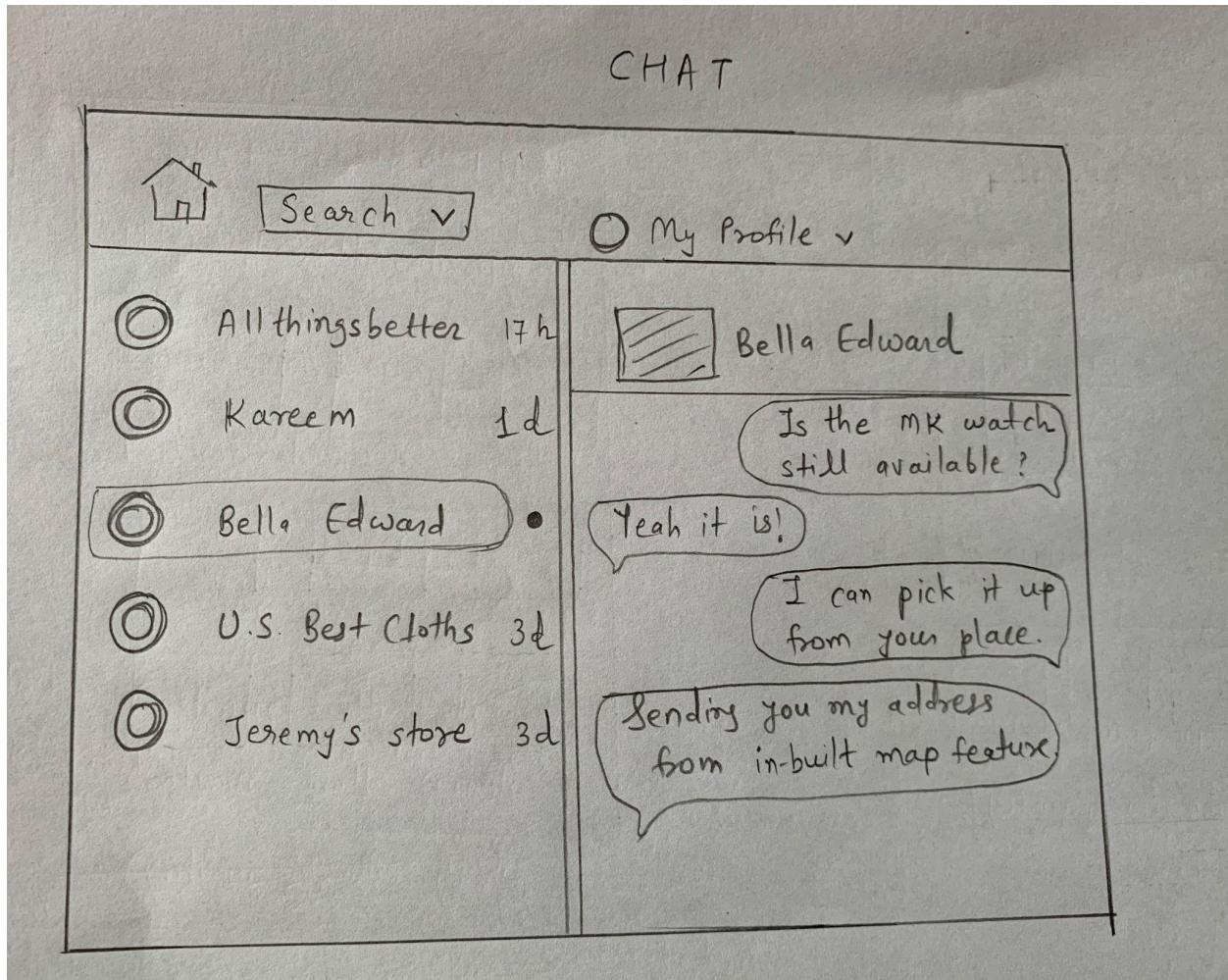
2.7) Summary			
APP LOGO			
Quantity	Product	Name	Price
1	IMAGE	MONITOR	\$ 1000.00
3	IMAGE	MONITOR	\$ 300.00
5	IMAGE	APPLES	\$ 5.00
Subtotal			\$ 1305.00
Fee			\$ 95.00
Tax			\$ 50.00
Total			\$ 1450.00
<a href="#">BACK</a>		<a href="#">CONTINUE</a>	

**Checkout**

**Final Invoice**

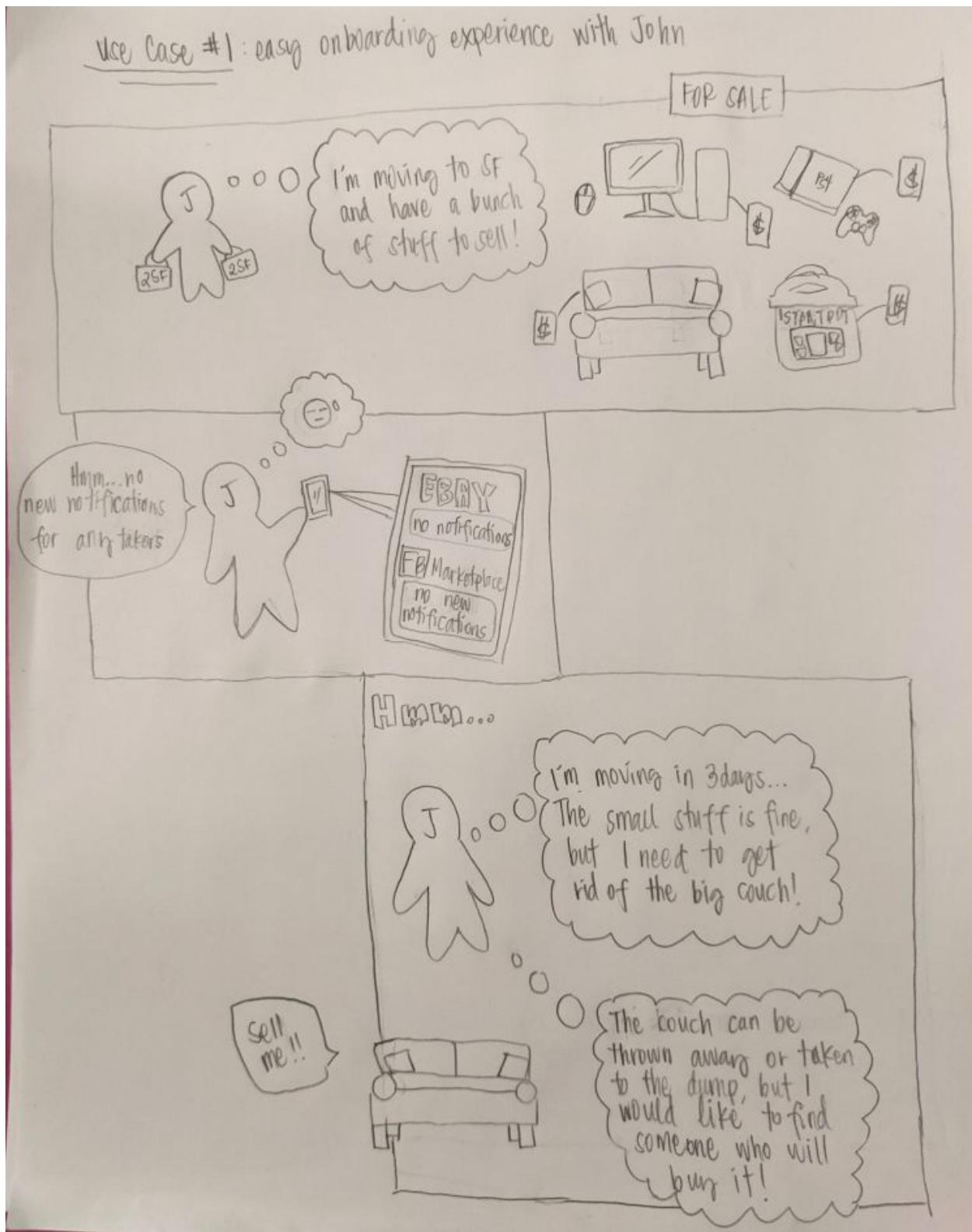
## Auction

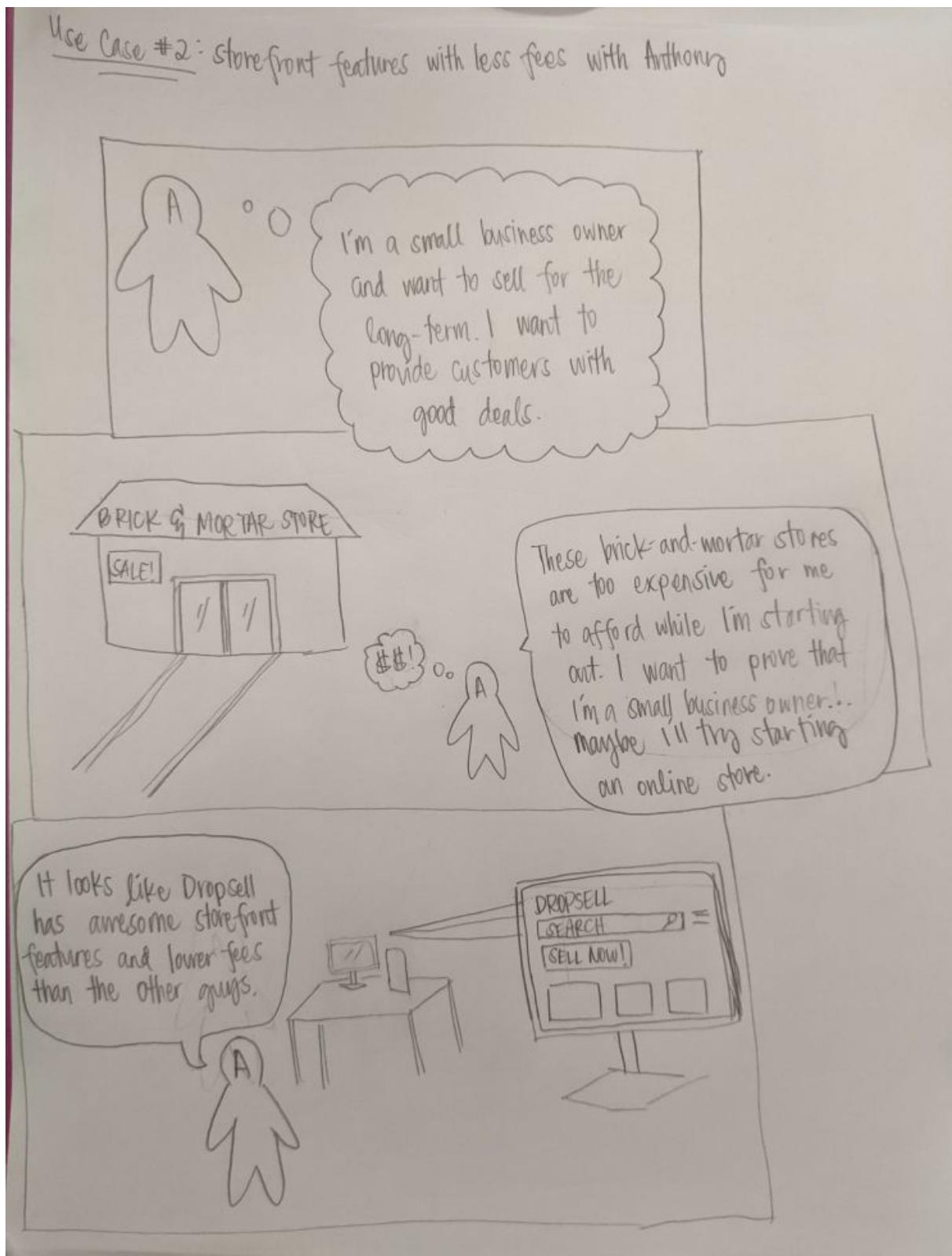


**Buyer/Seller Chat**

## Storyboards

### Use Case #1



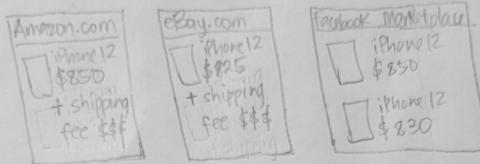
**Use case #2**

**Use case #3****Use Case III : Price Matching Feature**

Sandra wants a new iPhone 12, so she decided to check different websites for prices. So she decided to check different websites for prices.



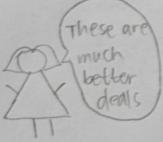
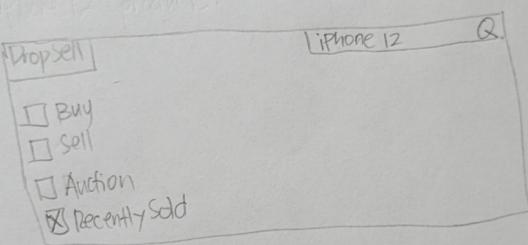
This is very time consuming



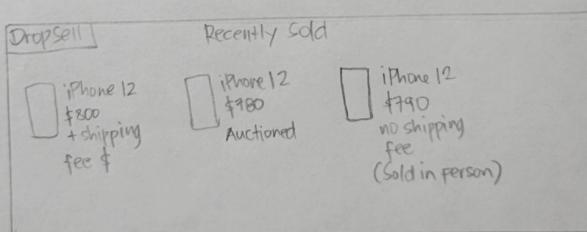
Most of them has a high shipping fee, if not, the products are overpriced.



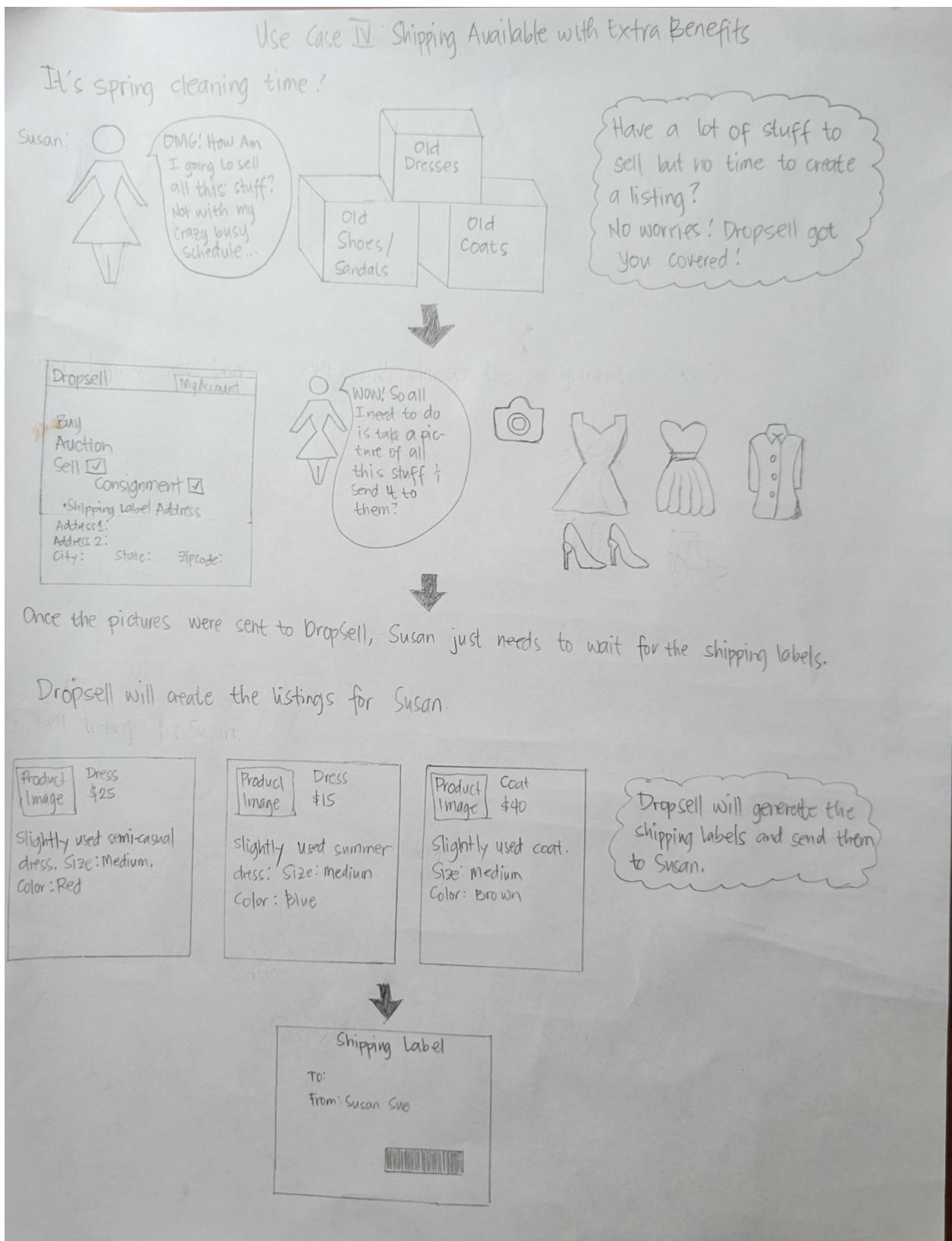
Sandra then went to DropSell website to compare the recently sold iPhone 12 products.



These are much better deals

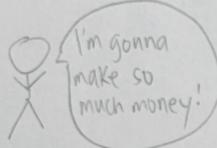


DropSell shows the most recently sold products + fees.

**Use case #4**

**Use case #5****Use Case II : Flagging System to Avoid Fraud Deals/Scams**

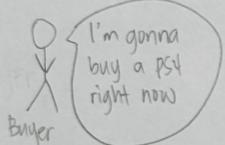
James uses DropSell to sell his products. However, his PS4 controller products are all fake.



**DropSell** Create a Listing

PS4 Controller  
\$  
Color: Black  
Brand new PS4 Controller,  
Qty: 10

Once the product has been listed, it will available in the DropSell website right away.



**DropSell** PS4 Controller

PS4 Controller  
\$  
Color: Black  
Brand new PS4 Controller,

+ - Add to cart  
Wishlist  
Bid  
Share

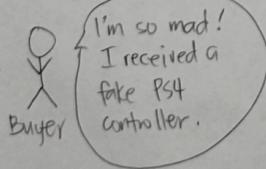
Once the buyer receives the product, he/she can rate the product & the seller (James).

Rate Product :

Rate seller :

Leave a Review: "Do not buy from this seller. I received a fake PS4 controller!"

Report Seller :



Once the buyer reported James, DropSell will flag the item as a scam & will be removed from the listing.

## 4. High Level Database Architecture and Organization

### *Business Rules*

1. One registered user can create zero, one, or many products. A registered user shall upload at least one image, many images are optional.
2. One registered user can have zero, one, or many seller ratings. Registered users can optionally keep track of the rating data to display in seller analytics.
3. One registered user can have one session assigned to them. User sessions can optionally store session data related to the user, in addition to mandatory session id and expiration values.
4. One registered user can have one shopping cart. It is optional for registered users to add products into their shopping cart, it is not necessary for records to exist in a user's shopping cart before a user is registered.
5. Many registered users can create zero, one, or many product comments. Registered users can optionally edit or delete their comment data.
6. Many registered users can create zero, one, or many conversations. It is optional for registered users to have conversations with other registered users, they should still have full functionality of the website.

### *Entities, attributes, relationships, domains*

1. Registered user
  - a. Attributes
    - i. user\_id
    - ii. username
    - iii. email
    - iv. password
    - v. first\_name
    - vi. last\_name
    - vii. phone
    - viii. street
    - ix. city
    - x. state
    - xi. zip
    - xii. is\_active
    - xiii. created\_at
    - xiv. is\_buyer
    - xv. is\_seller
  - b. Relationships (Other tables)
    - i. sessions
    - ii. shopping\_cart
    - iii. conversations
    - iv. messages
    - v. meetups
    - vi. seller\_ratings
    - vii. products
2. Sessions
  - a. Attributes
    - i. session\_id

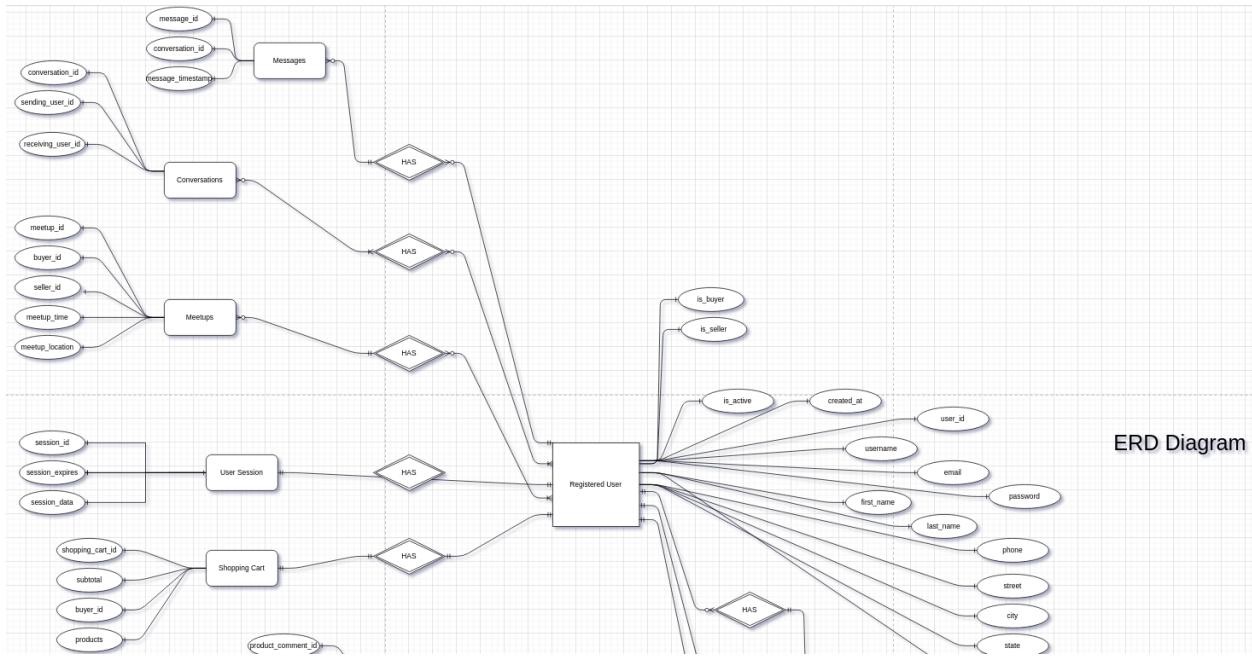
- ii. session\_expires
  - iii. session\_data
- b. Relationships
    - i. users
- 3. Conversations
    - a. Attributes
      - i. conversation\_id
      - ii. sending\_user\_id
      - iii. receiving\_user\_id
    - b. Relationships
      - i. users
      - ii. messages
- 4. Messages
    - a. Attributes
      - i. message\_id
      - ii. conversation\_id
      - iii. message\_timestamp
    - b. Relationships
      - i. Conversations
- 5. Meetups
    - a. Attributes
      - i. meetup\_id
      - ii. buyer\_id
      - iii. seller\_id
      - iv. meetup\_time
      - v. meetup\_location
    - b. Relationships
      - i. users
- 6. Shopping cart
    - a. Attributes
      - i. shopping\_cart\_id
      - ii. buyer\_id
      - iii. subtotal
    - b. Relationships
      - i. user
- 7. Marketplace Products
    - a. Attributes
      - i. product\_id
      - ii. seller\_id
      - iii. title
      - iv. description
      - v. price
      - vi. images
      - vii. category
    - b. Relationships
      - i. users
      - ii. product\_comments
      - iii. product\_ratings
      - iv. product\_refunds

- v. top\_purchased\_products
  - vi. daily\_deal\_products
  - vii. shipping\_products
8. Product comments
- a. Attributes
    - i. product\_comment\_id
    - ii. creator\_id
    - iii. product\_id
    - iv. comment\_timestamp
    - v. comment
  - b. Relationships:
    - i. products
    - ii. users
9. Product ratings
- a. Attributes
    - i. product\_rating\_id
    - ii. product\_id
    - iii. creator\_id
    - iv. product\_rating
  - b. Relationships:
    - i. products
    - ii. users
10. Product refunds
- a. Attributes
    - i. product\_refund\_id
    - ii. product\_id
    - iii. buyer\_id
    - iv. seller\_id
    - v. refund\_amount
  - b. Relationships:
    - i. products
    - ii. users
11. Auction products
- a. Attributes
    - i. product\_id
    - ii. seller\_id
    - iii. starting\_bid
    - iv. auction\_duration
  - b. Relationships:
    - i. products
    - ii. users
12. Top-purchased products
- a. Attributes
    - i. product\_id
    - ii. seller\_id
    - iii. total\_purchased
    - iv. added\_at
  - b. Relationships:
    - i. products

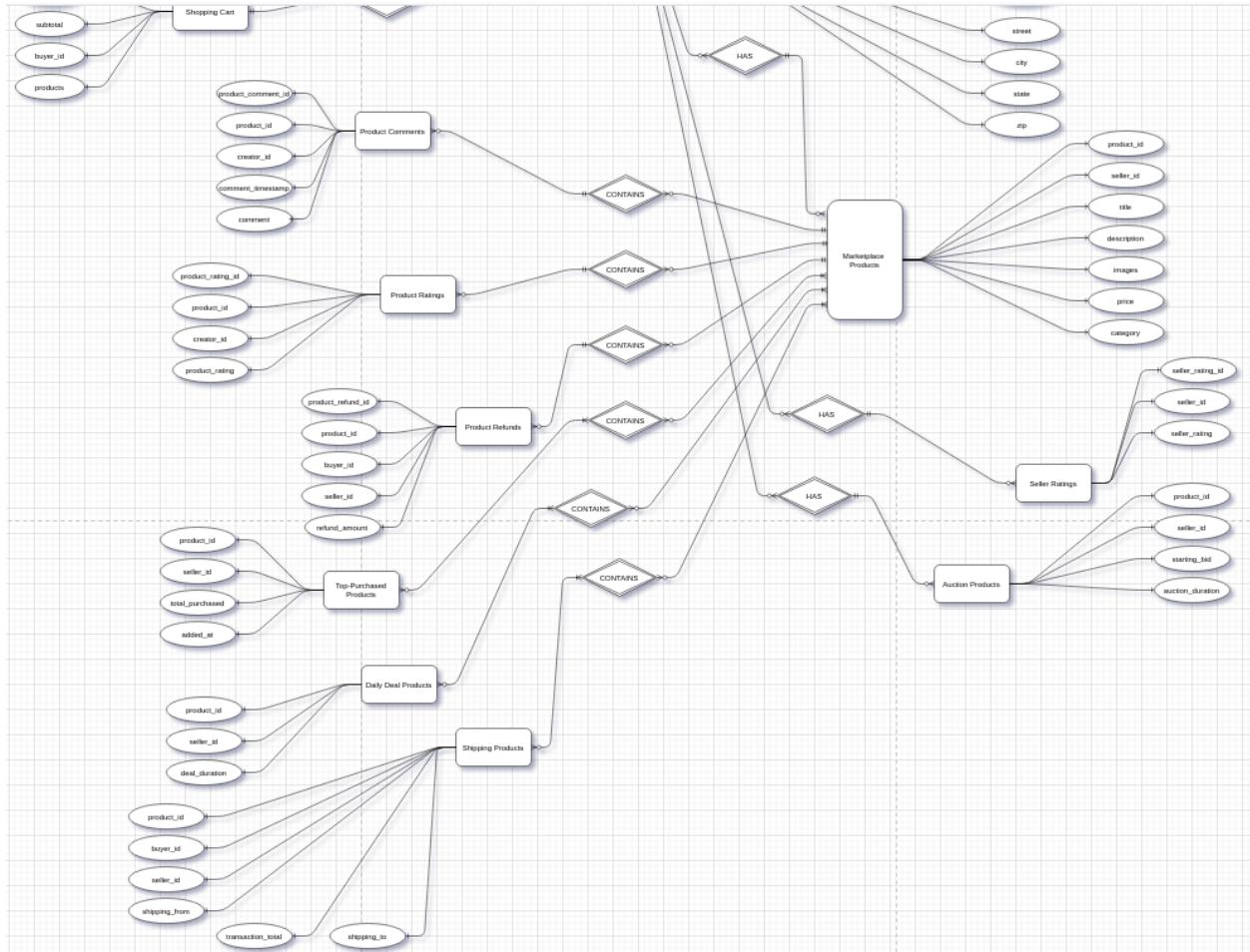
- ii. users
- 13. Daily deal products
  - a. Attributes
    - i. product\_id
    - ii. seller\_id
    - iii. deal\_duration
  - b. Relationships
    - i. products
    - ii. users
- 14. Shipping products
  - a. Attributes
    - i. product\_id
    - ii. buyer\_id
    - iii. seller\_id
    - iv. shipping\_from
    - v. shipping\_to
    - vi. transaction\_total
  - b. Relationships
    - i. products
    - ii. users

### Entity Relationship Diagram

1. <https://drive.google.com/file/d/1OmkYbqwzamnWdM2J0xKMSGNLJ2dQBWrc/view?usp=sharing> (top of document)

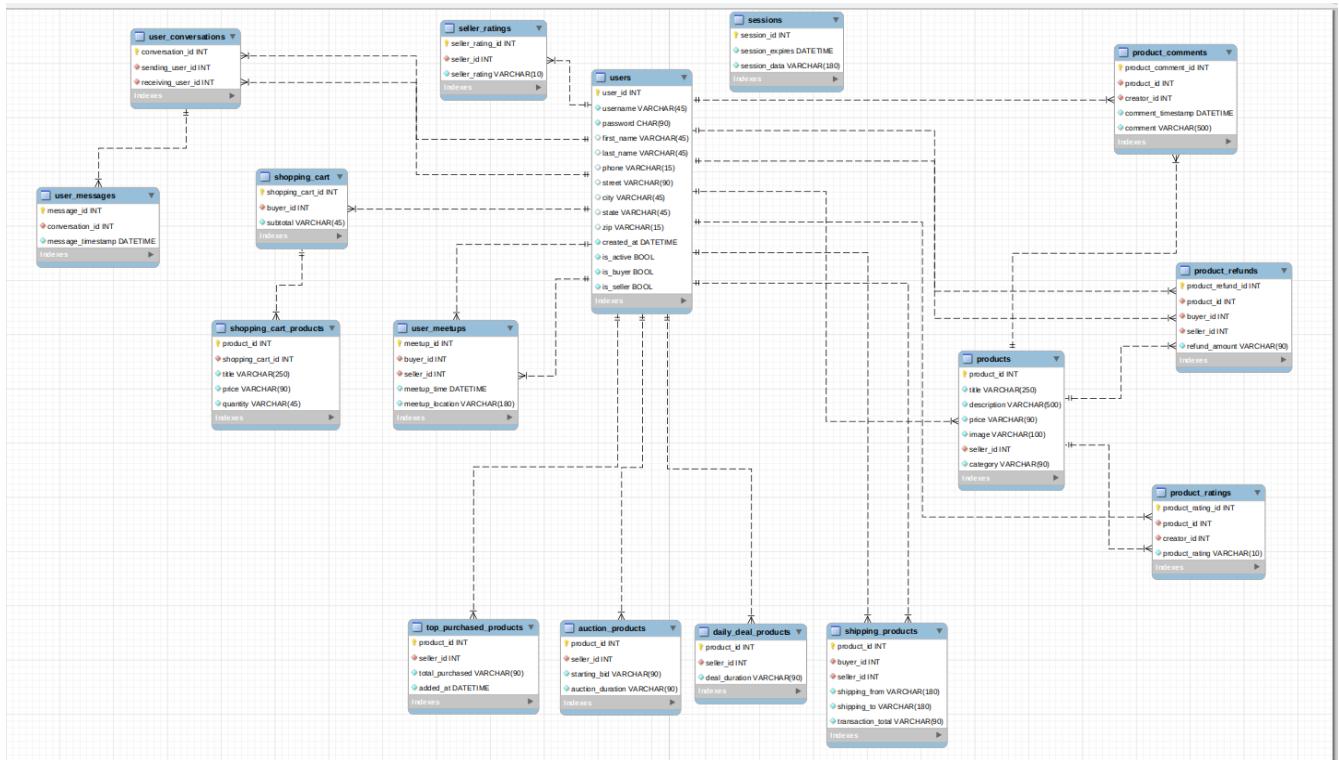


### Entity Relationship Diagram



## Database Model

1. [https://github.com/sfsu-joseo/csc648-848-sw-engineering-sum21-T03/blob/development/application/server/db/mysql\\_ja\\_model.mwb](https://github.com/sfsu-joseo/csc648-848-sw-engineering-sum21-T03/blob/development/application/server/db/mysql_ja_model.mwb)



## DBMS Decision

1. We will be using MySQL Workbench as our DBMS, since the software comes out of the box with features such as writing custom SQL statements, and creating/managing models and schemas. MySQL Workbench is also an easy-to-use interface which makes creating new tables, setting public/foreign key relationships, and creating new columns simple to explain to other team members.

## Media Storage

1. Images and video/audio files will be stored in our project's file system. We will only be storing the image and video/audio file names in our database in order to save space. Additionally, we have decided to store these media files in our project's file system because we can keep them in one consistent location. As a result, this makes lookup times and references to these media files easy and efficient. Since we are storing only the media file names in the database, we can simply prepend the proper path behind the image when we have to display these media files to the client. A good example of prepending the proper path would be with an `<img>` tag: `src={`/uploads/${product.image}`}`. When we load the `src` attribute, we can add `/uploads` to the front of the file name and we're done.

*Search/filter architecture and implementation*

1. Our search algorithm involves the use of a URLSearchParams object which is created when the user clicks the 'search' button of the search bar. We have implemented a filterProducts(products, query) function, which takes the current Array of products and the search query parameter to filter with. filterProducts() then returns an updated Array with the products corresponding to the specified search parameter. By default, our Home page will load all of the products saved in the database. This is done with a 'SELECT \* FROM products' SQL query.
2. In order to implement the category filtering for our search bar, we have implemented an initial array of categories to choose from: Clothes, Shoes, and Electronics. When a user opens the category dropdown menu, they can choose from these options to filter products. When they click one of the category options, an axios request is sent to our Node API which takes the category they clicked as a query parameter. On the backend, the '/api/product-categories' route will fetch all products from the database which correspond to the category the user clicked on. This is done with a 'SELECT \* FROM products WHERE category = ?" statement, where ? is filled in with the specified category.
3. Products will be created dynamically inside our ProductCreationForm.js component. With this component, users can specify the title, description, price, category, and image of their new product. After a user creates a new product, the product will be displayed on the Home page along with any other products.

## 5. High Level APIs and Main Algorithms

- a. API's
  - i. Creation of accounts(adding to DB)
    1. The user inputs all of their new account information, such as: username, email, password, and confirm password.
    2. Once the user passes all of the client-side validation, they can then register their account with the credentials they have provided.
    3. When the user clicks the register button, this will trigger a post request to our Node API server. The user's credentials will be sent in this post request using the request body object.
    4. After the post request is sent to the server, there is server-side validation performed, which checks that all of the user's credentials are valid.
    5. After the user's credentials are valid, this data will be added into an async/await call or a Promise object which will trigger the SQL statement which adds the new user into the database.
    6. If there were no errors when creating the new user, then the user should be directed to the login screen to login.
  - ii. Logging into accounts(accessing DB/verification)
    1. Input username/password, after clicking login then send post request to node api server, check to see if the user exists in the database, if it doesn't return error message to user, if it does, create a new session for the user, and then redirect them to their Home page
  - iii. Similar products
    1. When a user is browsing the application, store a history of the last 10 products they viewed. Use this information to display suggestions to the user.
  - iv. Buyers reviews, ratings.
    1. When users create reviews for products that review is stored in the database and linked to the appropriate item
    2. Ratings are based on a 5 star system.
      - a. As a registered user rates another registered user they leave a review that gets stored in the database and linked to the registered user.
      - b. Registered users can also leave a rating between 1-5 for that registered user that will be stored in the database linked to the appropriate user.
      - c. All reviews and ratings will be attached to profiles and when any user loads a profile this information will be retrieved from the database and displayed.
  - v. Shopping cart
    1. When a user adds an item to their shopping cart that item is reserved for them for a period of 10 minutes(can be adjusted or tied to a user's session).
      - a. OPTION A, when placed in a shopping cart that item is reserved to that user for a period of time so that another user can't accidentally buy the product before them if they check out faster.
      - b. OPTION B, we inform the user how many users concurrently have that item in their shopping cart and establish(and

- importantly inform the user) that their item is not secured until they start the actual check out/payment process. This means another user can accidentally steal another users item if they check out first.
- vi. Messaging.
1. Array of objects, with each object containing the ID of the message poster, ID of the message itself, and the time the message was sent.
- vii. Tag system/price checking
1. When an item is created the seller will enter all appropriate tags to the item in a fashion similar to hashtags. For example a PS5 might have the tags; #Electronics, #Game, #Sony (# used for reference).
  2. Using tags we can program different algorithms such as daily deals, categories, price checking.
  3. This tag will be stored with the item in the database,
  4. Users must enter a minimum number of tags.
- viii. Listing products and retrieving product information: title, description, image, price, brand, condition(new/used/etc), misc/notes, and importantly any tags that apply to the product.
1. Creating a new product, after checking that all required sections are filled in(title, description, image, price, brand, condition, misc/notes, and tags.), create a database entry linked to the user.
  2. If the request to see a product is valid, retrieve the product information and display it in the appropriate manner(ie seller/buyer history, auction listing, etc)
  3. When a product is created for sale, before submitting the item to the database/market place run the price checking algo and display suggested prices and similar product information.
- ix. Buyer Receipt details for delivery or checkout (first name, last name, phone, email)
1. If user elected for peer-to-peer deliver simply store the transaction information in the users history
  2. If a user opted for a consignment service then, like the peer-to-peer service, the transaction information will be stored in the user's history. Then a shipping label will be provided to the buyer.
- x. Inappropriate language detection
1. Maintain a list of words/phrases that are deemed inappropriate or that violate terms and conditions.
  2. When a user creates content(i.e. comments on a post, leaves review, creates a product, etc) parse the text into tokens and check against our list of inappropriate language.
  3. Handle the inappropriate comment based on our terms/condition.
- b. Significant non-trivial Algorithm/process
- i. Price Matching against similar products(Superior service)
    1. Using the Tag api to get product tags.
    2. Compare the tags listed on the product with similar tagged items
      - a. When comparing tags we will look at similar items that have been listed in the past(time period to be adjusted, up to 3 months).

- b. Based on the tags of the item compared to current and past tags of similar items we generate a list of information and present that to the user.
  - c. This information will show simple statistics of sales related to the item.
  - d. Display the picture of the item so the seller can see how relatable their item is to the suggested ones.
- 3. When a seller posts an item our price checker will display listings that have sold in the past, are currently listed, and products that are not selling and all associated price points of those objects.
  - 4. This price checking allows a new or inexperienced seller to get a feel for the correct amount they should list their item for.
    - a. Example 1, a user wants to sell an item fast.
      - i. They list the item, our price checking algorithm displays history of sells and current listings
      - ii. The user sees that the item averages 100\$ over the past 3 months.
      - iii. The seller sees that current listings of that item range from 85-90\$ and that items priced over 105\$ generally stay on the marketplace for 2 weeks.
      - iv. Seller values time over efficiency and lists the product for 75\$ to make the fastest money.
    - b. Example 2, a user wants the most money and is patient,
      - i. User has an item sitting around in their storage and wants to make the most money they can, and are willing to wait
      - ii. They list the item, our price checking algorithm displays history of sales and current listings.
      - iii. The user sees that the item averages 500\$ over the past 3 months
      - iv. The user sees that over the past 3 months items placed over 575 take an average of 3 weeks-1 month to sell.
      - v. The user also sees that items placed over 600\$ rarely sell.
      - vi. The user is patient and lists the item for 580\$ to get the biggest return.
    - c. Example 3, a user is new and has an unique item they don't know how to price.
      - i. The user has never sold anything second hand before and has a high quality painting they wish to sell.
      - ii. They list the item, our price checking algorithm displays history of sells and current listings
      - iii. The user sees similar paintings for this artist but does not see the exact match for his painting.
      - iv. Based on the information return the user can see an average cost of items of similar nature.
      - v. Using this information he knows that if he lists the painting for under 200\$ he will most likely be losing

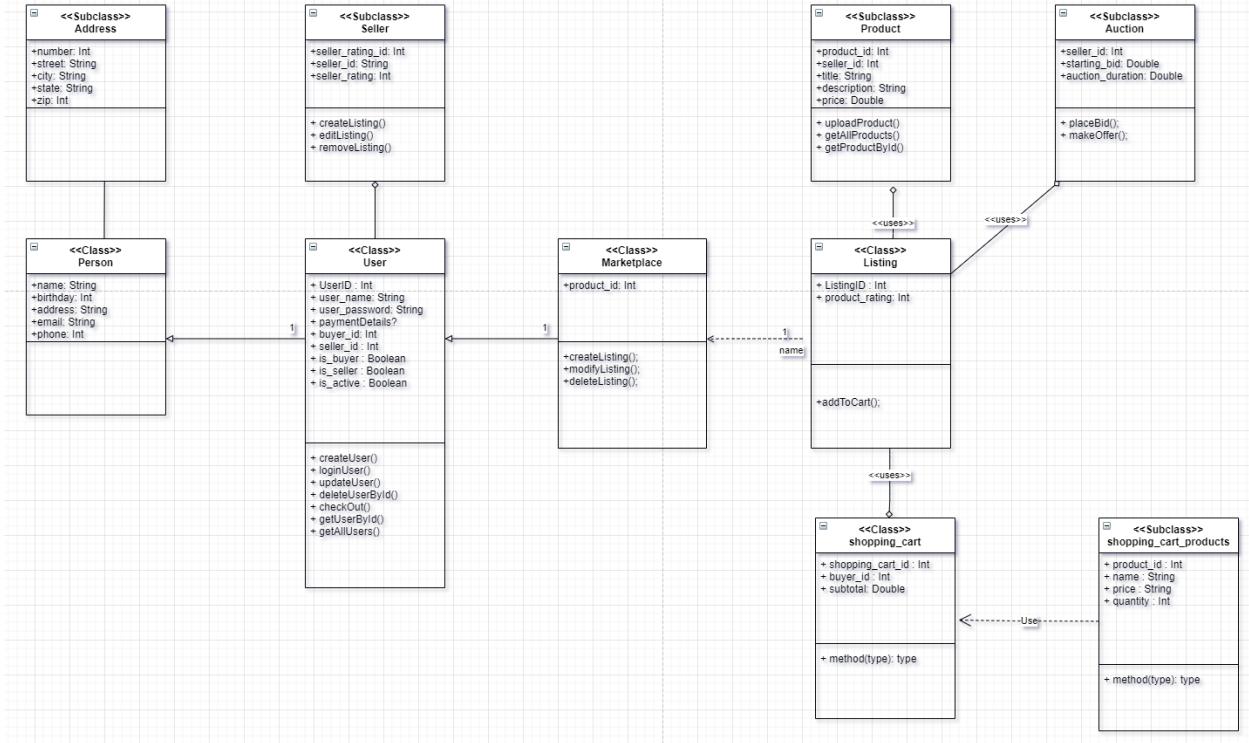
- money and that if he lists the painting for over 700\$ he may never sell it.
- vi. The user lists the painting for 400\$.
- ii. Ordering reviews and rates
1. This will help to organize reviews and rates
- i. When buyers purchase their items, they are able to review and rate their items and post them on a public page.
- ii. After their review and rate items end, then it will post them to the public in alphabetical order to make sure it is easy to find and organize for reviews and rates items.
- iii. Alerting if most of the items have bad reviews and rates
1. Let the sellers know that their selling items must be satisfied to the buyers.
- i. This can have feedback from the buyers, and making sure that when the sellers sell their items, they have to improve and make better items of what they are selling.
2. If the sellers cannot resolve this issue, then let the business management know about this issue.
- i. Having bad reviews and rates can cause down business selling items.
- ii. Sellers should be very responsible to make sure that their items are good enough.
- iv. Detecting inappropriate comments
1. When the buyers put comments on selling products, this algorithm will avoid inappropriate comments as possible.
- i. This is useful because when young children visit this site, then they should NOT read any inappropriate comments.
2. Use API Calls from the list of inappropriate words, to see if one or more words matches the comments, then the comments should not be allowed to post them in public.
- v. Recommend best meetup time
1. When the buyers and sellers decide to schedule meetup times, then recommend the best meetup times will show the best of what days and times to meet.
- i. Just in case the buyers or sellers are too lazy to set up their scheduled meetings. They can just click "Show best meeting times".
2. Use a graph algorithm to optimize the best meetup times between the seller and buyer by analyzing their schedule planner.
- vi. Calculate the mean and standard deviation of the number of selling, buying, refund, and return products.
1. Keeping track of a fair number of selling, buying, refund, and return products.

2. This will help to manage business and keep track of customers and sellers actions.
- vii. Alerting if most of the items have many reports and complaints from buyers.
1. If most of the buyers report their items, then the sellers and business management let know that they should be aware to resolve this issue.
- viii. Match categories for all products
1. Making sure what kind of products belong to which categories
    - i. For example, if the seller sells a pencil and the categories are school, work, home, and etc. The algorithm will figure out what pencil does relate to one of these categories. Pencil item should be in the school category.
  2. This helps to organize and search for buyers' wanted items.

## 6. High Level UML Diagrams

1. [\(Bottom of document\)](https://drive.google.com/file/d/1OmkYbqwzamnWdM2J0xKMSGNLJ2dQBWrc/view?usp=sharing)

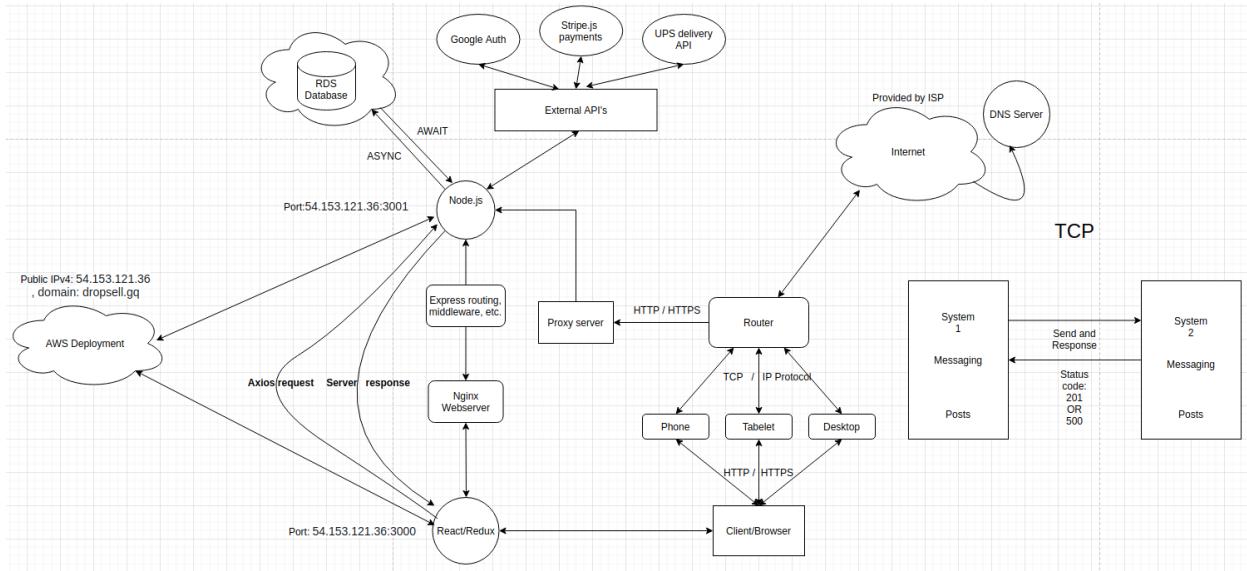
UML Diagram



## 7. High Level Application Network and Deployment Diagrams

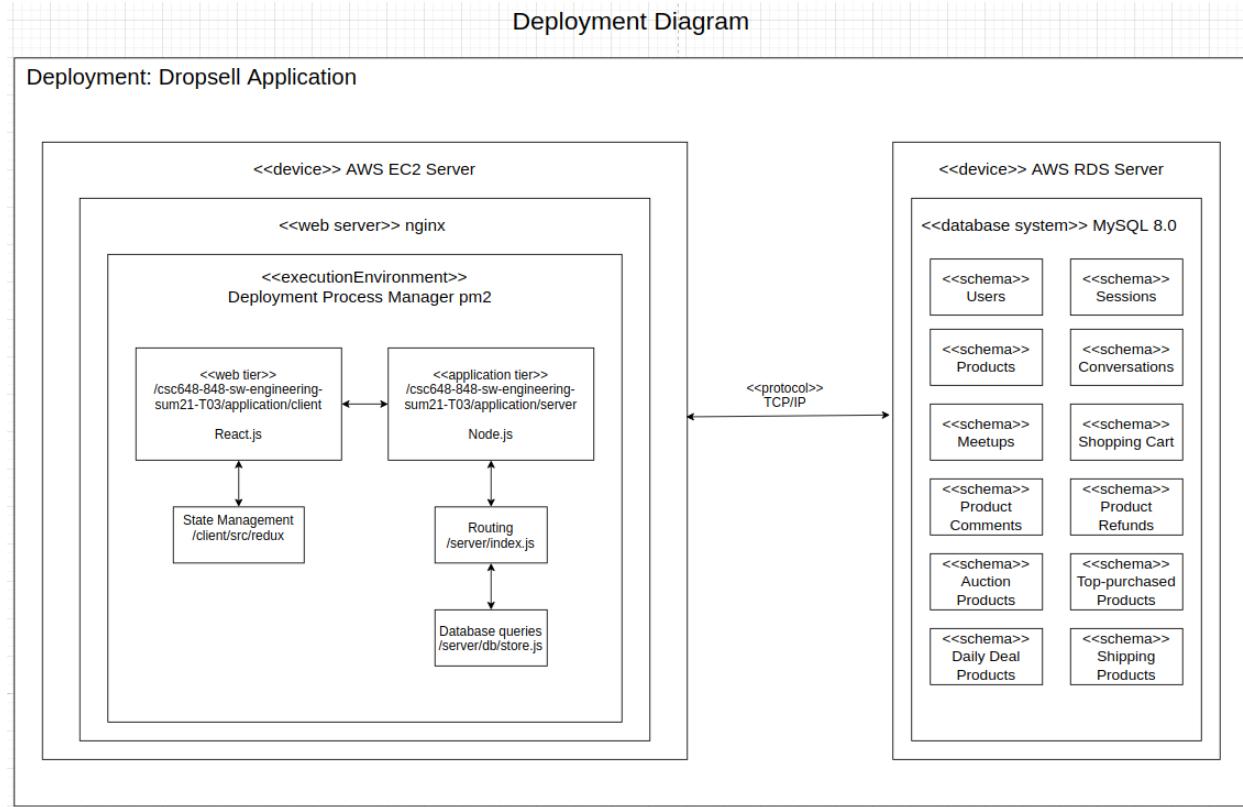
### *Application Network*

1. <https://drive.google.com/file/d/1XLbonkJqFnO7ZeOMQqstFPeZm8cr7UxX/view?usp=sharing>  
(Top of document)



### Deployment diagram

1. [\(Bottom of document\)](https://drive.google.com/file/d/1XLbonkJqFnO7ZeOMQgstFPeZm8cr7UxX/view?usp=sharing)



## 8. Identify actual key risks for your project at this time

- c. Schedule risks
  - i. There are conflicting schedules when trying to set up group meetings. Our group will mitigate this by working both asynchronous and synchronous when possible.
- d. Legal risks
  - i. This is related to either copyright laws when saving images or copying another company's idea. We will resolve this by either making our content as unique as possible, or obtaining the appropriate copyrights and/or licensing.
- e. User risks
  - i. This is related to scammers who wrongfully use the application. To resolve this issue, we will add security such as captcha, email verification, two-factor authentication, and a report button which users will be able to utilize accordingly.
- f. Selling items risk
  - i. This unique risk comes with many businesses set up for users to sell their products. Some products can possibly be dangerous and illegal to sell. We will resolve this by labeling such products as dangerous and/or illegal.
- g. Skills risk
  - i. As with any group work, there will be different skill levels amongst the members. We will mitigate this by sharing information, teaching each other, and working together in a way that best benefits what each member is the most capable of doing.

## 9. Project Management

As a team, our group decided the best way to complete this milestone was to take a divide and conquer approach. We worked together synchronously as much as possible, asynchronously when we were not able to meet, and communicated through Discord throughout. During meetings, we discussed how we would prioritize our functional requirements and categorized them accordingly. We split up the UI mockups and storyboards initially, then split up work on the database, UML and network diagrams, storyboards and editing the milestone document. We messaged each other in Discord concerning updates. We had set meetings for any questions, and discussed backlogs (upcoming lists of tasks we want done), new features, running tasks (tasks under development, tasks that were fixed and/or upgraded, tasks ready to move to production, and tasks that were labeled as done and/or deployed.) We set up and utilized the Trello application for managing all these tasks as a group. Overall, the methods that our group has been using to complete the milestones have worked really well so far. We will continue to use Trello, have scheduled meetings to manage current and future tasks, as well as maintain daily communication on Discord.

## 10. Detailed list of contributions

<u>Student Name</u>	<u>Contributions</u>
Mitchel Baker	<p>Mitchel started off M2 working on the UI mockups for our user checkout experience, which included: receipt info, summary, checkout, and final invoice pages. He collaborated with Michael to complete the high level database architecture requirements, ranging from our business rules, ERD diagram, database model, DBMS of choice, media storage decision, and the search/filter architecture. Mitchel also worked with Kenneth in order to research/design our application network diagram, while also creating the deployment diagram, list of contributions, search/filter implementation, and the vertical SW prototype. Lastly, Mitchel wrote up the data definitions and filled in necessary details.</p>
Charmaine Eusebio	<p>Charmaine created a few of the most important UI/UX web pages for our site. She wrote up from scratch the seller analytics (key feature), the create product page, list of auctions, products for sale, starred/watching product and starred/watching list. She also played a pivotal role with the storyboards which reflected our use cases 1 and 2. Lastly, Charmaine identified many of the key risks which she found sufficient solutions for, while also discussing our project management style.</p>
Kenneth N Chuson	<p>Kenneth did a tremendous job with the amount of detail he put into the profile and user settings pages. You can see the amount of work he put into them as a result of his mockup's quality. Kenneth also played a huge role with the write up of our high level API's and main algorithms. He has brought up</p>

	numerous ideas involving data analytics and machine learning which will really make our application stand out. Lastly, Kenneth conducted the majority of the research for creating our application network diagram.
Krina Bharatbhai Patel	Krina jumpstarted our progress on Milestone 2 by creating our initial set of data definitions. She laid the groundwork for the future additions added into our data definitions section. Krina also demonstrated her creativity with the login, register, auction, and chat mockups. Her mockups display all of the key features which we want to implement for our application, while showing off a solid understanding of UI/UX fundamentals. Krina was also responsible for prioritizing our functional requirements. Despite not being an easy task, she took to it quickly and showed that she can deliver quality work with efficiency. Lastly, Krina was also a big part of creating the vertical prototype by implementing a couple of the main UI components for our home page.
Michael Schroeder	Michael was assigned the creation of our initial home and menu page mockups. He took to these tasks with diligence and demonstrated his ability as a quick learner. Michael also collaborated with Mitchel on the ERD diagram portion of the high level database architecture requirements, while also implementing from scratch the UML diagram for our project. Despite not having prior experience writing UML diagrams, he demonstrated an eagerness to learn new skills which is a rare skill to find in people.
Rowena Elaine Echevarria	Rowena was in charge of writing up the about page for our UI mockups. She also did a tremendous job with the creation of her storyboards for use cases 3, 4, 5. She incorporated a ton of

	<p>detail into her storyboards, which demonstrates her attention to detail, especially in regards to UI/UX. As a result of the detail put into her storyboards, our Milestone 2 gives readers the opportunity to see first-hand examples of how DropSell solves problems for users who are looking to either buy or sell products online.</p>
Jamie Dominic Walker	<p>Jamie was in charge of the high level API and main algorithms section. He demonstrated an eagerness to learn more about API's by researching and sharing videos/articles about the topic. Jamie is someone who knows how to apply his critical thinking skills to real-world applications. When it comes to explaining his thoughts and ideas, Jamie takes an implementation-based approach to explaining our main algorithms which was extremely helpful. He explored various edge cases related to our API implementations, while also bringing up thought-provoking analysis during our meeting discussions.</p>