SW Engineering CSC648/848 Section 02 Fall 17

Real Estate Website

"Agent 007"

Group 07

Local

Team Members

Dillan Brosnan (Lead)

-San Francisco State University

(dillonbrosnan11@gmail.com)

Roy Anguiano

-San Francisco State University

Indra D Gairhe

-San Francisco State University

Vikram Grewal

-San Francisco State University

Anirudh Mohan

-San Francisco State University

Darryl Raveche

-San Francisco State University

Milestone 2

October 24, 2017

Revision	Changes
10/23	Initial state of the Milestone 2 documentation
10/27	Revision after Petkovic's SCRUM notes
10/29	Final Revision FROZEN

1. Data Definition

Listing:- Properties listed for sale. Only agents will be allowed to list to the listings. Browsers, buyers and sellers will have the ability to see the listings. Each will include the following features:

- **Address -** physical location of the listing**.** Address will include the followings required data:
    - Street number – may contain letter or number of combination.
    - Street name -
    - City
    - State- abbreviated two letter name of the state listing is located.
    - Postal code – a 5-digit number assigned by USPS.
- **Pictures –** Digital image of house providing view of features of the property
- **Number of rooms** – includes living rooms and bedrooms
- **Number of bathrooms** – includes full bath room and fractional bathroom. A full bathroom will have sink, toilet, shower and bathroom, and half size bathroom will include sink and toilet. Size will be displayed in decimal number. For example a house with 3 bathroom and half will be described as to contain 3.5 bathrooms.
- **Square footage** – multiple of width and length of house
- **Year built** – the year the house was finished in usable condition. Does not include the renovation completion date.
- **Lot_size** – size in square footage of a listing's outer boundary.
- **HOA**: Binary value where the listing is managed under **H**ome Owners Association. The value is either True or False.
- **Type**: Type of property listed and can be of the following:
    - House
    - Townhome
    - Manufactured Home
    - Condos
    - Mobile Trailers
    - Commercial - property used strictly for business.

Price: Advertised price of the listing. Agents can opt to not advertise the price or state that the price is negotiable if stated or firm price.

Listing ID: a unique number assigned to a listing.

Account:- Arrangement to each user/agents under which information of each each is stored as provided by the registering individual; identified by userID for users and agentID for agents.

Browsers:- An Individuals visiting website, who have not yet registered. Browsers will not be able to contact agents without registering.

Users:- Individuals who have successfully completed the registration process by providing required information. Users will have the following information.

- UserID: an Unique number assigned to each user
- First Name : User entered mandatory first name, can contain special character.
- Middle Initial: option middle name, that can be one letter or multiple strings.
- Last name: User entered mandatory Last name, may contain characters like hyphen (-), apostrophe (‘) etc.

Agents:- individuals who has real estate property to sell. Agents will have following mandatory data:

- AgentID: unique id assigned to the agents.
- First Name
- Middle Name
- Last Name
- Street Number: street number of address of the agent doing business from.
- Street Name: Street name of the business address of agent.
- City; City where the address is located
- State: two letter state of the state
- Postal Code: 5 digit number of the business address
- Phone: 9 digit number of the phone number that users and admin can contact to.
- License Number: professional real estate license number of the individual.

Admin:- Administrator or individual employee of the company who is authorized to

- Access accounts
- Delete accounts
- Contact account holder and modify as reasonable.

Key Word: text or number used to search the listing that users, buyers, agents enter in text areas.

Posted date: Date the listing is posted for sale.

2. Functional Requirement:

1 Browsers

Priority 1

1.1 Shall be able to see the listings, including those that were sold recently or listed for sale when visiting the website.
1.2 Shall be able to filter listings by Zip Code of address, Price of house, or Number of Bed/Bath Rooms in the house.
1.3 Ability so sign up for account and contact admin.
1.4 Shall be able to order the search result by such as price.
1.5 Shall be able to see where, on Google Maps, the selected listings are located.

Priority 2

1.1 Shall be able to zoom in the map with relevant information such as home address, price displayed in such as text area or information window (infowindow).
1.2 Able to see photos and extended information of the listing a browser is interested in.

Priority 3

1.1 Shall be able to view nearby businesses.
1.2 Shall be able save favorite listings without having to register.

2 Users

Priority 1

2.1 Shall have all the capability of Browser.
2.2 Shall be able to contact Agents of the interested listings by clicking the house listing and get detail on the web about the house.
2.3 Shall be able to update account information.
2.4 Shall be able to view messages replied by agents or admin.
2.5 Shall be able to register as agent
2.6 Shall be able to eply to agent.
2.7 Shall be able to save favorite listings.

Priority 2

2.1 Shall be able to create a list of listing in order of priority.
2.2 Compare different listing in one GUI.

3 Agents

Priority 1

3.1 **Shall be able to create account as an agent.**
3.2 Shall be able to enter and update listing information.
3.3 Shall be able to add and update photos and videos of listings.
3.4 Shall be able to respond to Users.
3.5 Shall be able to contact admin.
3.6 Shall be able to view their Dashboard, containing Sellers listings and /or messages between agents and buyers.

Priority 2

3.1 Shall be able to import file for listing
3.2 Shall be able to schedule sale promotion (for future date).
3.3 Shall be able to update the price and advertise such as “Reduced Price” , “Nice Neighbothood”, “Top High School District”, “24 Hours Security” etc.

Priority 3

3.1 Shall be able to contact past sales history with the current listings of the agent based on criteria
3.2 Shall be able to list view past visits and dash board by users and browsers

4 Admin

Priority 1

4.1 Shall be able to create account as employee
4.2 Shall be able to contact/response users, and agents
4.3 Shall be able to view who is registered with the site.
4.4 Shall be able to delete or modify accounts and save the state.

Priority 2

4.1 Shall be able view web analytics such as visit number, time spent on the website.
4.2 Shall be able send promotional email or alert to users in the selected area based on past search area.

3. Mockups and Use Cases

1. Browser:

Betty wants to buy house. She visits the website agent 007. Form the home page, Betty is met with many options as to what to do during her time on agent07. Predominantly placed are showing of features listings. Betty is directed to a page, where all the homes available for sale as well as featured homes are displayed. Also featured in the web page is that she can enter key word such as zip code, city etc. to narrow the listings she can focus on. Once she enters the search option, the result will be displayed with information and about the house along the location of the house in google maps.


Home
Buy
Sell
Favorites
Sign in
Enter City, zipcode...
Search
850K New listing
3 bath|2b
Pinole, CA
2bath|3 bedr
Richmond, CA
Sold
2bed2bath
Static lane
Multipine Lane
$500K Great
About
Careers
Investors Relations
More houses

Home Buy Sell

Sort options

bedroom
Price
build year
location

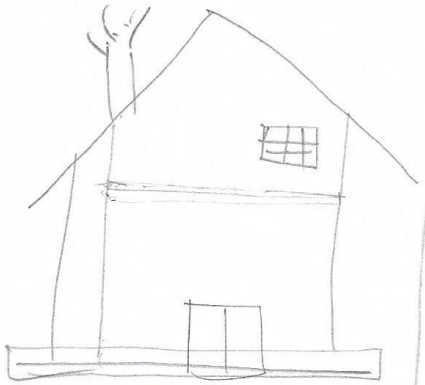
Contact agent

Google map

name 1

name 2

name 3



Price
year
bed/bath

Contact

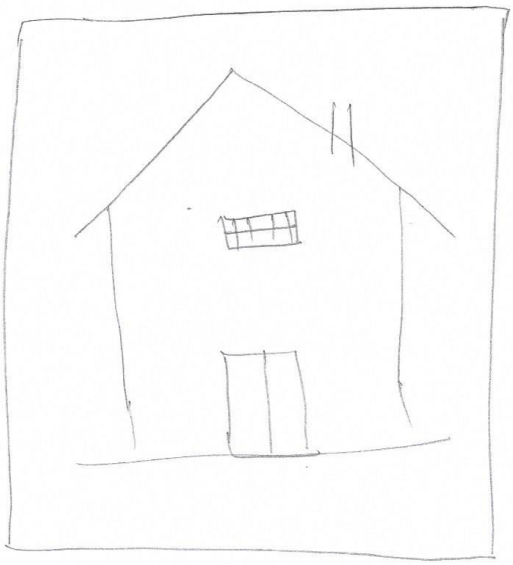
More houses

2. Buyers:

Betty tellers her friend, Brian, about our website Agent 007. Brian is a very eager buyer, and is shopping around homes for him and his family. Brian starts the same way out as Betty, simply browsing the site, until he comes across one he is very interested. He clicks at the picture to get more information. He will be directed to next page where he will find more information. After reviewing the information, he is more interested and wants to contact agent. But he must sign in got contact. Since he has not registered before, he is directed to sign in/register page. He will be directed to the page where he can register by entering the necessary information. He agrees to terms and condition and verifies the CAPTCHA. He will chose register option.


Home
Buy
Sell
Favorites
Sign in
Enter City, zipcode...
Search
250K New listing
3 bath | 2b
Pinole, CA
2 bath | 3 bedn
Richmond, CA
Sold
2bed 2bath
Multiplane Lane
Strathie Lane
$500K
Great Finish
of
About
Careers
Investors Relations
More
More houses

Home Buy Sell Favorites Signin



Picture 1

Picture 2

Detail information about the house . . .

Picture 3

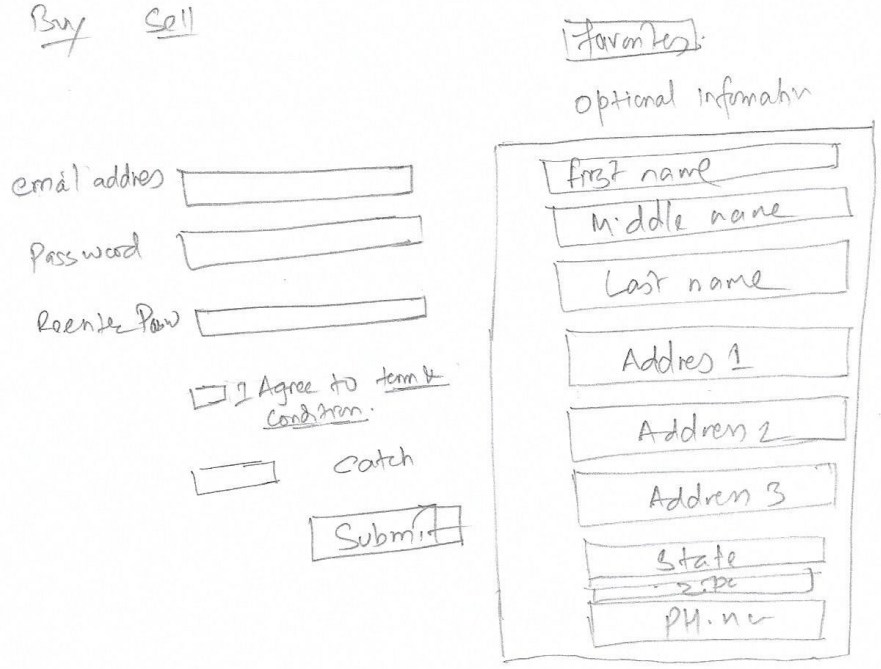
Contact seller

More pictures

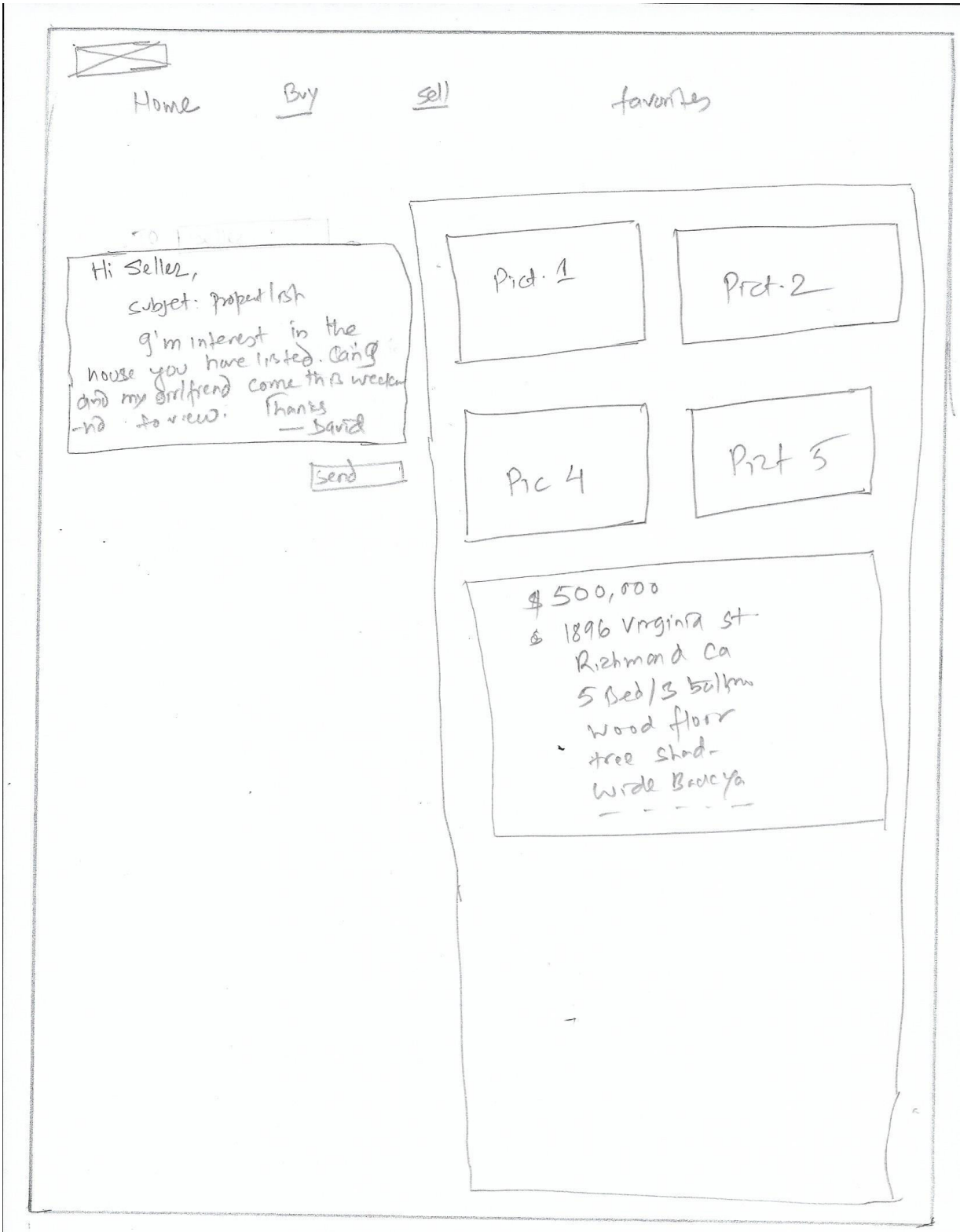
About Careers More

17 Annual

Home Buy Sell



About — — —


Home
Buy
sell
favorites
Hi Seller,
subject: propertlsh
g'm interest in the house you have listed. Can g and my girlfriend come this weeken -nd to view. Thanks
— David
Send
Pict. 1
Pict. 2
Pic 4
Pict 5
$500,000
1896 Virginia St
Richmond Ca
5 Bed/3 bathm
Wood floor
tree shad-
Wide Bouc ya

3. Agents

Sam is an up and coming realtor. She is looking for new places to list her homes for potential buyers to see. After finding our website, Agent 007, Sam wants to post her house that is for sale. Registering as a seller is much like registering as a Buyer, except Agents can post houses and Buyers can contact agents. After registering as an Agent, Sam is directed towards the Seller Dashboard, which contains items like messages from Buyers, and ways to browse Sam's listings. She wants to update the listings.


Home
Buy
Sell
Favorites
Sign in
Enter City, zipcode...
Search
250K New listing
3 bath/2b
Pinole, CA
Richmond, CA
Sold
2bed2bath
Static Lane
Multipane Lane
$500K
Great
More houses
About
Careers

Home Buy

log out

Edit Profile

messages

my listings

Settings

web visits

other information

New listing Alert

Status update

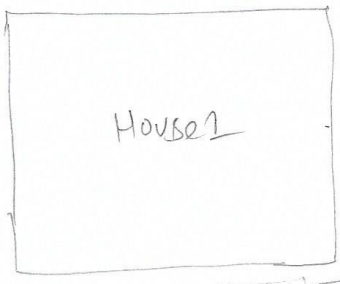
New sale alert

Sale status

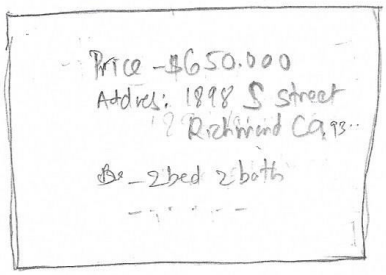
Home Buy Logout

1. Edit

Date listed

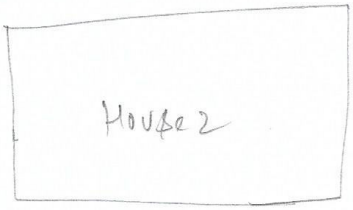


02/17/17

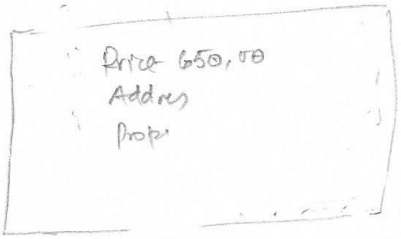


Set Promoting Updat. Delet

2.

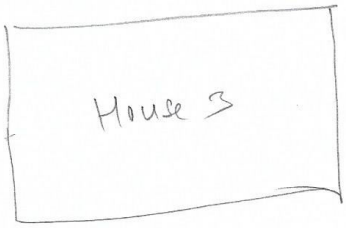


03/18/17

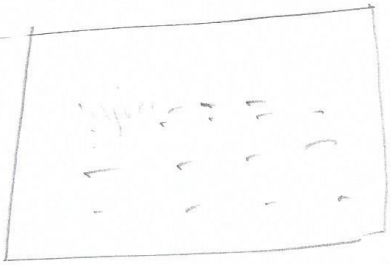


Set Promoting update dele.

3.



03/19/17



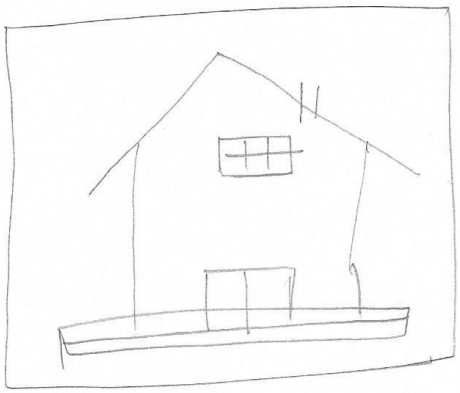
Set Promotion update delete

4.

Home Buy

favorites log out

Current Shop



Advertise delete

Pict delete

Pict 2 delete

Pict 3 delete

Pict 4 delete

upload New Picture

Editabe text area

500,00

Addren 1

Addres 2

update

4. High Level Architecture, Database Organization

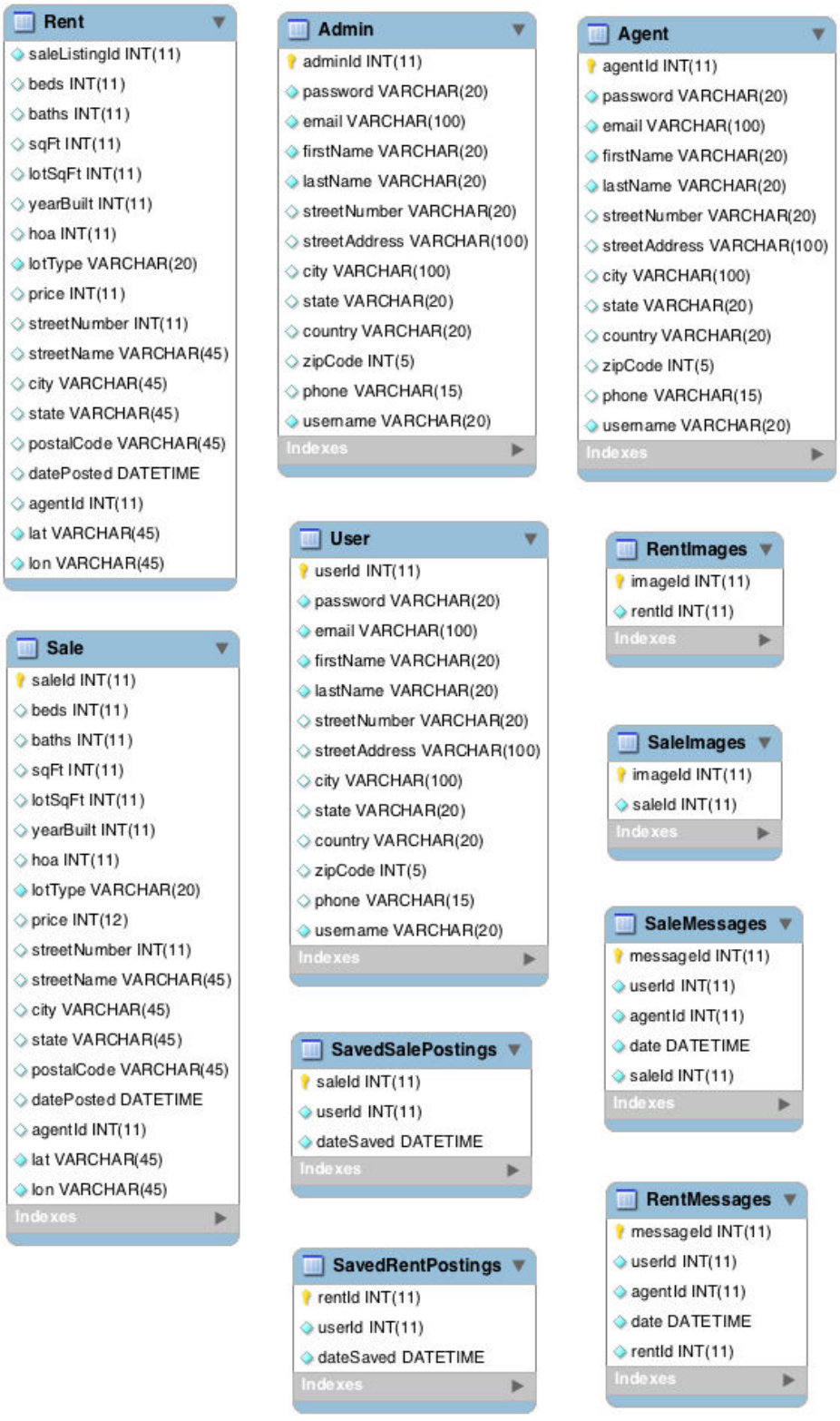
High Level Architecture

The website consists of multiple routes including the sell, rent, login, signup, and dashboard routes for all actors. It uses the MVC design pattern in which for the models we have all of our database queries including database schemas for the each one of the following routes. This allows us to separate the logic out of the view and controller. There are multiple pieces of information that must be fetched within each route depending on what parameters are passed within the URL or in the request to that page. This logic is know as are controller, but in the case of express the routing takes care of the controllers. Express takes in a number of requests, but it essentially matches to that of a CRUD design since it uses an http server. The server is constantly listening for POST, GET, UPDATE, and DELETE requests, so we can listen to each URL on the website for what kind of server request happens. These routes allow us to validate either on the view or in the logic of the route. If the information is compliant to that of what needs to be inserted, deleted, or fetched in the database, we can perform the required database query. All of this logic is defined in the model as stated before. Express is also taking care of rendering all the views on the website using our EJS view engine. The model and controller are both responsible for rendering views since once the information changes for the database it must also be rendered by the view. The routes that were defined earlier each have a purpose for rendering views. Sometimes it is unnecessary to render views with information since the user on the website has not requested any additional information with the view. If this is the case, we have separate logic for views created with data fetched and separate logic for views that are rendered without additional data.

DB Organization

The website uses a MySQL DB for all of its information. It is organized into multiple tables which include: Sale, Rent, Admin, User, Agent, SavedPostings, ExteriorDetails, InteriorDetails, Messages, and Flag postings. Our Sale and Rent tables are identical in data columns, but it allows us to separate the logic of price vs price per month for each of the houses. The models which include our database queries fetch for certain information depending on when the user requests houses for sale vs houses for rent. Both of these tables hold all necessary information for basic details on whatever type of property that is being sold (sqFt, lotSize, beds, bathrooms, street number, street name, city, state, zip code, lat, lon, and a primary key of saleId vs rentId, and an agentId). The id of the property will reference different tables including the SavedPostings, Messages to the agents regarding certain properties, and also the agentId allows us to go fetch the contact information of the agent. The admin, user, and agent tables all hold the same piece of data columns too. Each has different permissions for the website while using the website, so it is necessary to separate the entities from each other in the database. As stated before, each of these users once logging in will be fetching their id information from the correct database table when logging in. This will allow us to route the user to the correct dashboard

without accidentally displaying information to certain users that should not have permission to access that field of the website. This is a basic database schema of our website as of now.


Rent
saleListingId INT(11)
beds INT(11)
baths INT(11)
sqFt INT(11)
lotSqFt INT(11)
yearBuilt INT(11)
hoa INT(11)
lotType VARCHAR(20)
price INT(11)
streetNumber INT(11)
streetName VARCHAR(45)
city VARCHAR(45)
state VARCHAR(45)
postalCode VARCHAR(45)
datePosted DATETIME
agentId INT(11)
lat VARCHAR(45)
lon VARCHAR(45)
Admin
adminId INT(11)
password VARCHAR(20)
email VARCHAR(100)
firstName VARCHAR(20)
lastName VARCHAR(20)
streetNumber VARCHAR(20)
streetAddress VARCHAR(100)
city VARCHAR(100)
state VARCHAR(20)
country VARCHAR(20)
zipCode INT(5)
phone VARCHAR(15)
username VARCHAR(20)
Indexes
Agent
agentId INT(11)
password VARCHAR(20)
email VARCHAR(100)
firstName VARCHAR(20)
lastName VARCHAR(20)
streetNumber VARCHAR(20)
streetAddress VARCHAR(100)
city VARCHAR(100)
state VARCHAR(20)
country VARCHAR(20)
zipCode INT(5)
phone VARCHAR(15)
username VARCHAR(20)
Indexes
Sale
saleId INT(11)
beds INT(11)
baths INT(11)
sqFt INT(11)
lotSqFt INT(11)
yearBuilt INT(11)
hoa INT(11)
lotType VARCHAR(20)
price INT(12)
streetNumber INT(11)
streetName VARCHAR(45)
city VARCHAR(45)
state VARCHAR(45)
postalCode VARCHAR(45)
datePosted DATETIME
agentId INT(11)
lat VARCHAR(45)
lon VARCHAR(45)
Indexes
User
userId INT(11)
password VARCHAR(20)
email VARCHAR(100)
firstName VARCHAR(20)
lastName VARCHAR(20)
streetNumber VARCHAR(20)
streetAddress VARCHAR(100)
city VARCHAR(100)
state VARCHAR(20)
country VARCHAR(20)
zipCode INT(5)
phone VARCHAR(15)
username VARCHAR(20)
Indexes
RentImages
imageId INT(11)
rentId INT(11)
Indexes
SaleImages
imageId INT(11)
saleId INT(11)
Indexes
SaleMessages
messageId INT(11)
userId INT(11)
agentId INT(11)
date DATETIME
saleId INT(11)
Indexes
SavedSalePostings
saleId INT(11)
userId INT(11)
dateSaved DATETIME
Indexes
SavedRentPostings
rentId INT(11)
userId INT(11)
dateSaved DATETIME
Indexes
RentMessages
messageId INT(11)
userId INT(11)
agentId INT(11)
date DATETIME
rentId INT(11)
Indexes

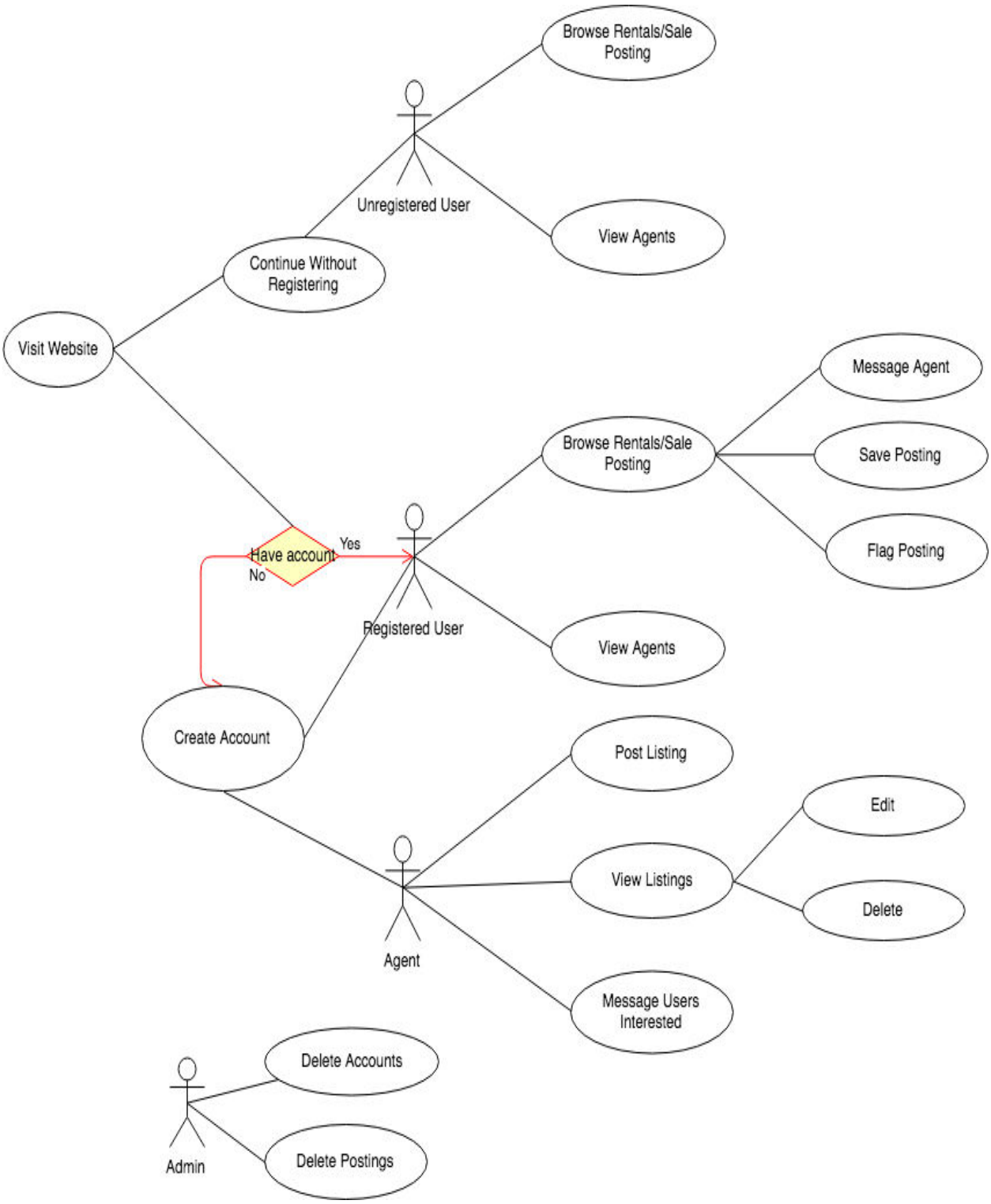
Media Storage

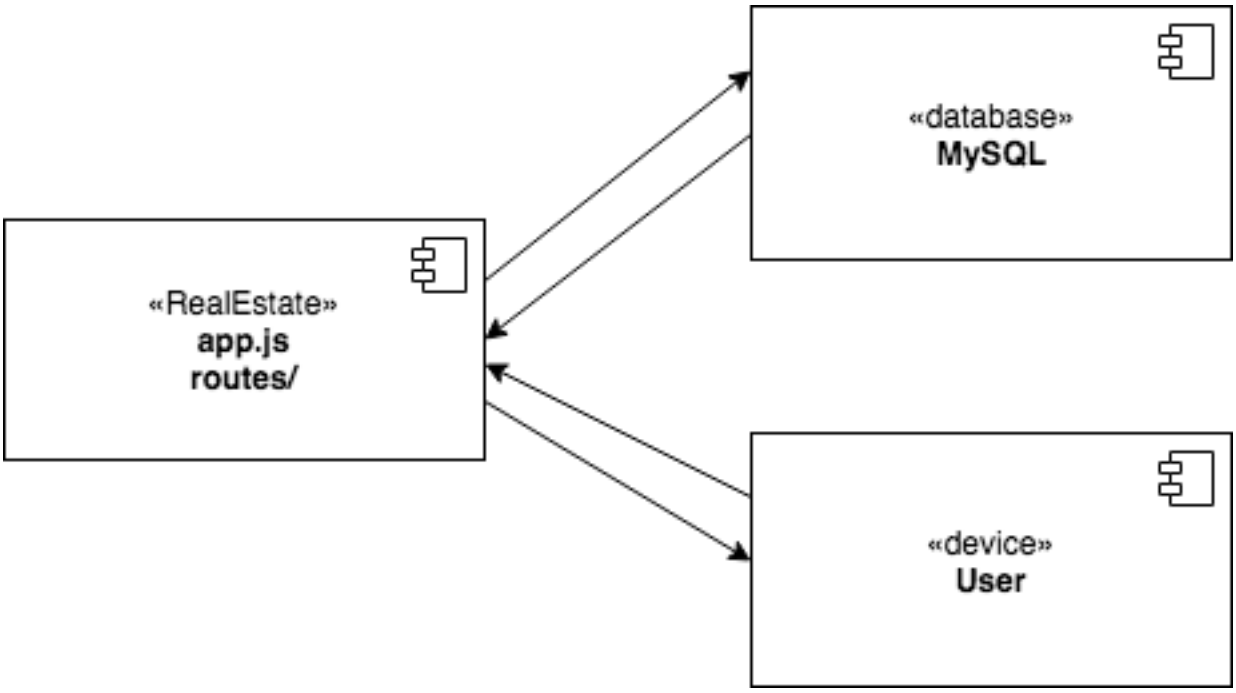
For media storage, we will not be using DB Blobs. Instead we will be using the express-file-upload middleware. This will allow us to save the images on our file system without storing it in the database. Instead of saving the image itself in the database, we will be storing the imageId, (saleId/rentId) which in turn will allow us to go fetch the image from the file system. We will be able to limit sizes for the files along with the file types by checking what accepting 3 mimetypes (jpg, png, and gif). Each posting will be limited to a certain amount of files on the file system that way one posting doesn't over use our file system.

Search/Filter Architecture and Implementation

For the search functionality of the sale/rent postings, we will be picking a start location that is taken from a autocomplete form using Google Places API. This location has a lat/lon which will be compared with each of our postings in our database. We are using a rhumb line to calculate the distance between both of these locations. The error in the miles is about 5% which is extremely low for distance. The search will allow users to search within a certain distance of their search point using our mathematical formula and lat/lon points. As for the filter, I believe it would be easier to do basic filtering using the database (bedrooms, sqFt, lotSize, lotType), but when it comes to more advanced keywords that will be taken care of using the DOM. This will allow us to take some stress off the server during advanced queries. These queries will be inserted into the URL using parameter queries.

5. High Level UML Diagrams




«database»
MySQL
«RealEstate»
app.js
routes/
«device»
User

6. Key Risks

Skills: Our team at this point has the necessary skills to complete this website. While no body is perfect in every area, what one member lacks, another makes up for. We also possess the most powerful skill of all, the ability to learn, and learn quickly.

Schedule Risks: The entire team is very busy, with other projects and personal matters that must be attended to as well. I see no problem in our team being able to prioritize and maximize our time spent together and apart, working on this website.

Technical Risks: For some of the team, this is the first time building a functioning website with so many moving parts and intricacies. Our tool belt contains many tools unseen to some of us before building this site, but with help from each other and guidance from the CTO and CEO we will learn to use these technical tools to our full advantage.

Teamwork Risks: Since beginning this website there have been two main teamwork risks that have been observed, lack of communication and procrastination. Either can be very risky to a project to this size with so many people working. After speaking with the parties concerned, the team lead addressed the issues at group level, and asked for better communication within the group, as well as better time management. The team lead also plans to foster communication for a better working environment for all.

Legal/Content Risks: We as a team see no reason there should be any legal or content risks involved with this project.