SW ENGINEERING CSC 648/848 FALL 2022

EZRent
By CyberDesign
Team 4

Members:

Team Lead : Devon Dy-Liacco ddyliacco@mail.sfsu.edu

Github Master: Praise O Eubany

Frontend Lead: Youssef Hammoud

Backend Lead: Issa Shihadeh

Database Master: Tung Nguyen

Frontend Engineer: Ricardo Lopez

Milestone 2

10/19/2022

History Table:

Version	Latest revision
M2V1	10/19/2022
M2V2	11/10/2022

Table of Contents

1. Data Definitions

User
- Anyone who uses the site
- No data items

Registered User
- A user who has an account
- email, password, phone number, name

Landlord
- A Registered User who can post listings and be reviewed by other Registered Users
- landlord rating
- Post listings
- Edit listings

Renter
- A Registered User who can post reviews about Landlords or Property Listings
- renter rating

Prospective Renter
- A Renter who advertises themselves to Landlords
- list of listing attributes that they're looking for

Listing
- **(**Also referred to as Property Listing) a post containing a property's information
- listing rating, images (max 20), landlord, property

Comment
- A posted field of text posted to a Listing or another Comment
- author, message, number of likes, number of dislikes

Direct Message
- Messages between two Registered Users
- sender, time sent, whether it was read or not, message

Profile:
- A webpage containing information about a Registered User
- bio, profile picture, rating

Search Result
- A list of Listings or Profiles outputted by the search engine
- rating, associated attributes

Review
- A post to a Listing or Landlord
- description, rating

Rating:
- A way to analyze a person's experience with a Landlord by choosing 1 out of 5 stars (5 being the best)
- decimal average of all ratings posted about a particular profile or listing

Property
- Housing owned by a Landlord
- price, rating, rooms, description, location, size

Block List
- A list of Registered Users that a particular Registered User has blocked

Moderator
- Someone who handles reports and poor behavior
- Access to Registered Users comments/ratings

Banned List
- A list of Registered Users that are not allowed to use the site's services

Report
- A message sent to Moderators indicating that a Registered User doesn't comply to the site's community guidelines
- includes a reason, and a link to the violation

Help Page
- Teaches Users how to use the app's services and allows them to give feedback about the site

Site Review:
- User's feedback about the site
- comments, rating, experience

Filter:
- Search engine parameters
- price, location, rooms, rating

Login:
- An account's way of accessing a Profile
- email, password

Create account
- The ability to make a profile with a first name, last name, email and password.
- landlord, renter

Map
- The view and area of the listings on a visual diagram
- addresses, links to available properties

Save list
- A list of saved properties where a User can go back to view the same listings that were being viewed.
- property listings

2. Prioritized Functional Requirements

Priority 1

Criterion	Requirements
1. User Registration	1.a. Users must be able to register an account to become a Registered User 1.b. The system must allow Users to agree to a Privacy Policy 1.c. Users must be able to browse the website without an Account 1.d. The system must allow Users to provide their personal information 1.e. The system must allow Users to create a username for their account 1.f. The system must allow Users to change their personal information 1.g. The system must allow Registered Users to become a Landlord
2. User login	2.a. The system must allow Registered Users to log in their account. 2.b. The system must allow Registered Users to reset their password. 2.c. The system must allow Registered Users to browse the website 2.d. System must allow Registered Users to log out
3. Website Header	3.a. The Website Header must include the search feature 3.b. The Website Header must be able to link to the Registration Page 3.c. The Website Header must be able to link to the Login Page 3.d. All Registered Users must be able to navigate to their Profile 3.e. All pages must include the Website Header
4. Home Page/Navigation	4.a. The Home Page must be accessible to all Users 4.b. Users must be able to navigate to the Home Page from any page
5. Website Footer	5.a. The Website Footer must allow Users to see the publisher's information 5.b. All pages must include the Website Footer
6. Search	6.a. Users must be able to search for Property Listings 6.b. Users must be able to search for Landlords 6.c. Users must be able to search Listings by address

	6.d. Users must be able to search Landlords by name
7. Search Filters	7.a. Users must be able to filter their searches 7.b. Users must be able to filter Property Listings by price 7.c. Users must be able to filter Property Listings by address 7.d. Users must be able to filter Property Listings by Amenities 7.e. Users must be able to filter Property Listings by number of rooms 7.f. Users must be able to filter Property Listings by number of bathrooms 7.g. Users must be able to filter Property Listings by size 7.h. Users must be able to filter Property Listings by rating
8. Search Results	8.a. Users must be able to visit Property Listing pages from the search results page 8.b. Users must be able to visit Landlord Profile pages from the search results page 8.c. Users must be able to sort their search results 8.d. Users must be able to sort their Listing search results by price 8.e. Users must be able to sort Listing search results by rooms 8.f. Users must be able to sort Listing search results by distance 8.g. Users must be able to sort Listing search results by rating 8.h. Registered Users must be able to sort Renter search results by rating
10. Ratings	10.a. Renters must be able to rate Properties 10.b. Renters must be able to rate Landlords 10.c. Users must be able to see Ratings

11. Reviews	11.a. System must allow Renters to write Reviews 11.b. System must allow Landlords to write Comments to Reviews 11.c. System must allow Renters to reply to Comments 11.d. System must allow Registered Users to rate Comments 11.e. System must allow Registered Users to rate Reviews 11.f. System must allow Landlords to reply to Comments 11.g. System must allow Users to read Reviews 11.h. System must allow Users to read Comments 11.i. System must allow Users to view Ratings 11.j. System must allow Registered Users to change their ratings to Reviews 11.k. System must allow Registered Users to change their ratings to Comments
12. Property Listing	12.a. Landlords must be able to post Listings 12.b. Landlords must be able to delete Listings 12.c. Landlords must be able to mark Listings as currently rented 12.d. Landlords must be able to unmark Listings as currently rented 12.e. Landlords must be able to post images to Listings 12.f. Landlords must be able to write descriptions to Listings 12.g. Landlords must be able to set locations to Listings 12.h. Landlords must be able to set a price to Listings 12.i. Landlords must be able to set a size for Listings 12.j. Users must be able to see Listings 12.k. Users must be able to find the Landlord page of the one who posted the Listing 12.l. Users must be able to post Reviews to Listings 12.m. Registered Users must be able to post Reviews to Listings 12.n. Renters must be able to save Listings to favorites
13. Landlord Profile	13.a. System must provide a Profile for Landlords upon Account creation 13.b. Landlords must be able to set a bio to their Profile 13.c. Users must be able to see a Landlord Profile 13.d. Users must be able to see Reviews about a Landlord on the Landlord Profile 13.e. Users must be able to see a Landlord’s rating on their Profile 13.f. Users must be able to see a Landlord’s contact information 13.g. Users must be able to see a Landlord’s name 13.h. Users must be able to see a Landlord’s email 13.i. Registered Users must be able to post Reviews to Landlord Profiles

14. Renter Profile	14.a. System must provide a Profile for Renters upon Account creation 14.b. Renters must be able to set their Account to Prospective Renter 14.c. Renters must be able to set a bio to their Profile 14.d. Users must be able to see public Renter Profiles 14.e. Renter Profiles must show Renter ratings 14.f. Renters must be able to edit their Profile 14.g. Renter Profiles must show Renter history

Priority 2

Criterion	Requirements
1. User Registration	1.a. The system must send an email to confirm when the user created the account. 1.b. The system must provide a way for Users to verify their Account
2. User login	2.a. The system must provide a way for Registered Users to log in to their account if they forgot their password
7. Search Filters	7.a. Users must be able to control the amount of search results displayed to them
8. Saved Listings	8.a. Registered Users must be able to view all saved Property Listings 8.b. Registered Users must be able to save Property Listings to interact with later 8.c. Registered Users must be able to remove saved Property Listings 8.d. Registered Users must be able to visit Property Listing pages from the Saved Listings page
10. Ratings	10.a. Landlords must be able to rate Renters
11. Reviews	11.a. System must allow Renters to edit Reviews 11.b. System must allow Renters to edit Comments 11.c. System must allow Landlords to edit Comments 11.d. System must allow Renters to delete Reviews 11.e. System must allow Renters to delete Comments 11.f. System must allow Landlords to delete Comments
12. Property Listing	12.a. Landlords must be able to edit Listings 12.b. Landlords must be able to edit fields for Listings to be filtered through the search engine 12.c. Renters must be able to compare multiple Listings 12.d. Renters must be able to comment under Listings
13. Landlord Profile	13.a. Landlords must be able to edit their Profile 13.b. Users must be able to see a Landlord's phone number 13.c. Registered Users must be able to Direct Message a Landlord from their Profile
14. Renter Profile	14.a. Renters must be able to edit their Profile 14.b. Renter Profiles must show Renter history

16. Storage	16.a. System must store Direct Messages 16.b. System must store Saved Searches 16.c. System must store Banned Users 16.d. System must store Blocked Users
17. Moderation	17.a. Registered Users must be able to report Reviews 17.b. Registered Users must be able to report Comments 17.c. Registered Users must be able to report Listings 17.d. Registered Users must be able to report other Registered Users 17.e. Moderators must be able to confirm a report 17.f. Moderators must be able to send warnings to users 17.g. Moderators must be able to delete Reviews 17.h. System must allow Moderators to log in 17.i. System must allow Moderators to log out
19. Direct Message	19.a. Registered Users must be able to send Direct Messages to other Registered Users 19.b. Registered Users must be able to edit Direct Messages 19.c. Registered Users must be able to delete Direct Messages 19.d. Registered Users must be able to forward Direct Messages 19.e. Registered Users must be able to reply to a Message 19.f. Registered Users must be able to react to a Message 19.g. Registered Users must be able to pin a conversation 19.h. Registered Users must be able to delete a conversation 19.i. Registered Users must be able to search for a Direct Message 19.j. Registered Users must be able to hide a conversation 19.k. System must show that a Direct Message has been sent 19.l. System must show that a Direct Message has been read 19.m. System must allow Registered Users to turn off read receipts

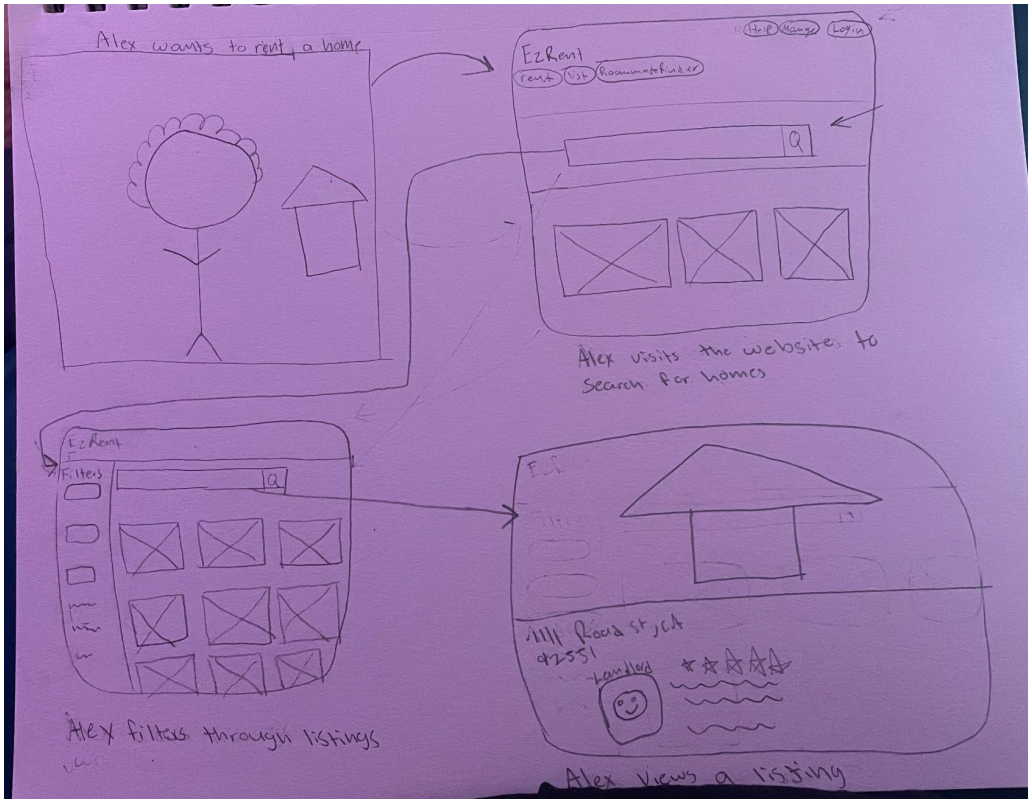
Priority 3

Criterion	Requirements
4. Home Page/Navigation	4.a. Users must be able to navigate to previously visited pages
6. Search	6.a. Landlords must be able to search for Renters
7. Search Filters	7.a. Users must be able to filter Property Listings by Landlord 7.b. Users must be able to filter Property Listings by Landlord properties 7.c. Users must be able to filter Landlords by rating 7.d. Users must be able to filter Landlords by name 7.e. Users must be able to filter Landlords by experience 7.f. Landlords must be able to filter Prospective Renters by rating 7.g. Landlords must be able to filter Prospective Renters by name 7.h. Landlords must be able to filter Prospective Renters by experience 7.i. Users must be able to save a collection of filters 7.j. Users must be able to filter Property Listings within range of a City 7.k. Users must be able to filter Property Listings within range of a zip code. 7.l. Users must be able to filter Landlords by number of reviews
8. Search Results	8.a. Users must be able to compare Property Listing search results 8.b. Users must be able to share search results 8.c. Landlords must be able to visit Prospective Renter Profile pages from the search results page 8.d. Users must be able to sort Landlord search results by rating 8.e. Users must be able to sort Landlord search results by experience 8.f. Registered Users must be able to sort Renter search results by experience
12. Property Listing	12.a. Users must be able to zoom in on images
14. Renter Profile	14.a. Landlords must be able to see private Renter Profiles 14.b. Renters must be able to set their Profile to private 14.c. Registered Users must be able to Direct Message a Renter from their Profile
15. Prospective Renter	15.a. Prospective Renters must be able to set requirements for Listings they are looking for 15.b. Prospective Renters must be able to set their Account back to a regular Renter 15.c. Landlords must be able to view Prospective Renter Profiles

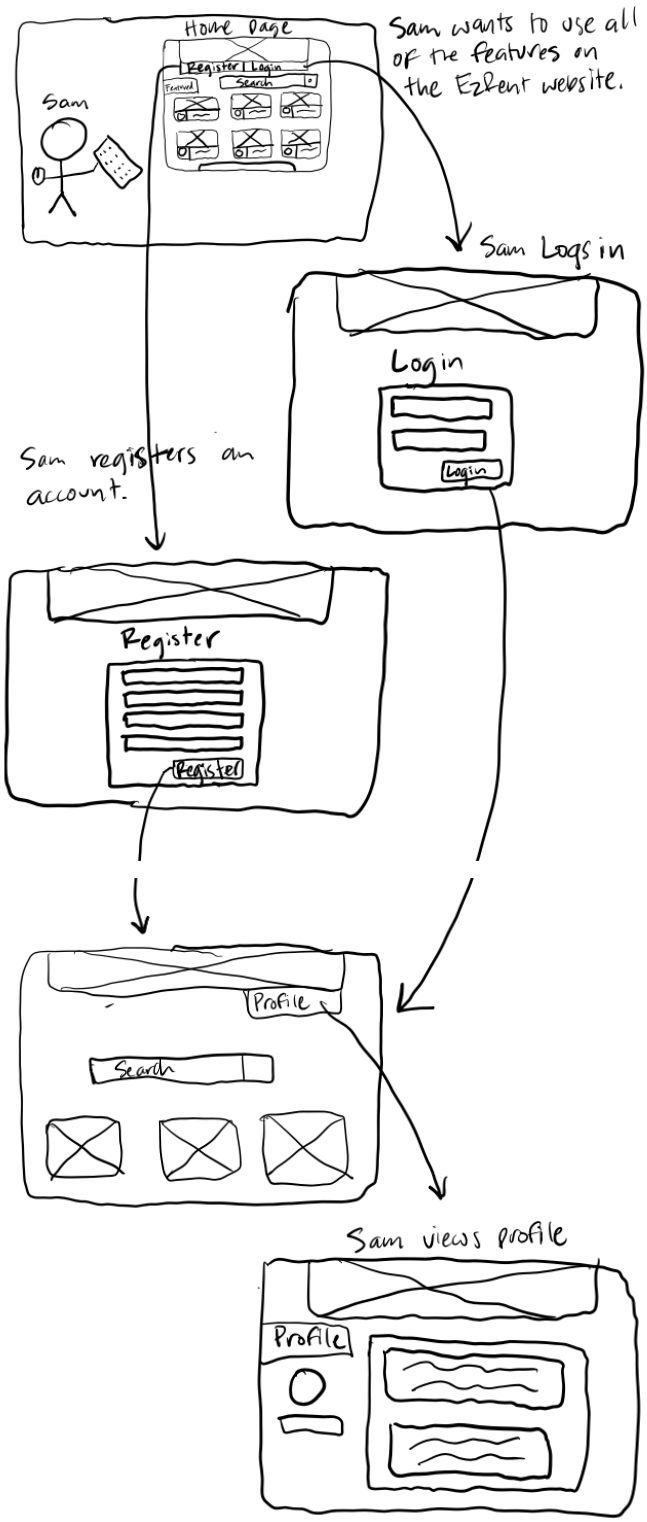
17. Moderation	17.a. Registered Users must be able to block other Registered Users 17.b. Registered Users must be able to unblock other Registered Users 17.c Moderators must be able to send warnings to users 17.d. Moderators must be able to revoke a Rating 17.e. Moderators must be able to ban Registered Users 17.f. System must allow Admins to add new Moderators 17.g. System must allow Admins to log in 17.h. System must allow Admins to log out 17.i. System must prevent Blocked Users from commenting on Blocker's Listings 17.j. System must prevent Blocked Users from commenting on Blocker's Profile 17.k. System must prevent Blocked Users from seeing Blocker's Profile
18. Help page	18.a. Help Page must include a tutorial of how to use its services 18.b. Help Page must include a way to contact Customer Service 18.c. Registered users must be able to rate experience while using the site
20. Map View	20.a. Users must be able to see a Map of Listings 20.b. Users must be able to interact with a Map of Listings 20.c. Users must be able to select an area of the Map 20.d. System must show search results on a Map 20.e Users must be able to view listings near their area 20.f. Users must be able to view landlords within their area

3. UI Mock-ups and Storyboards

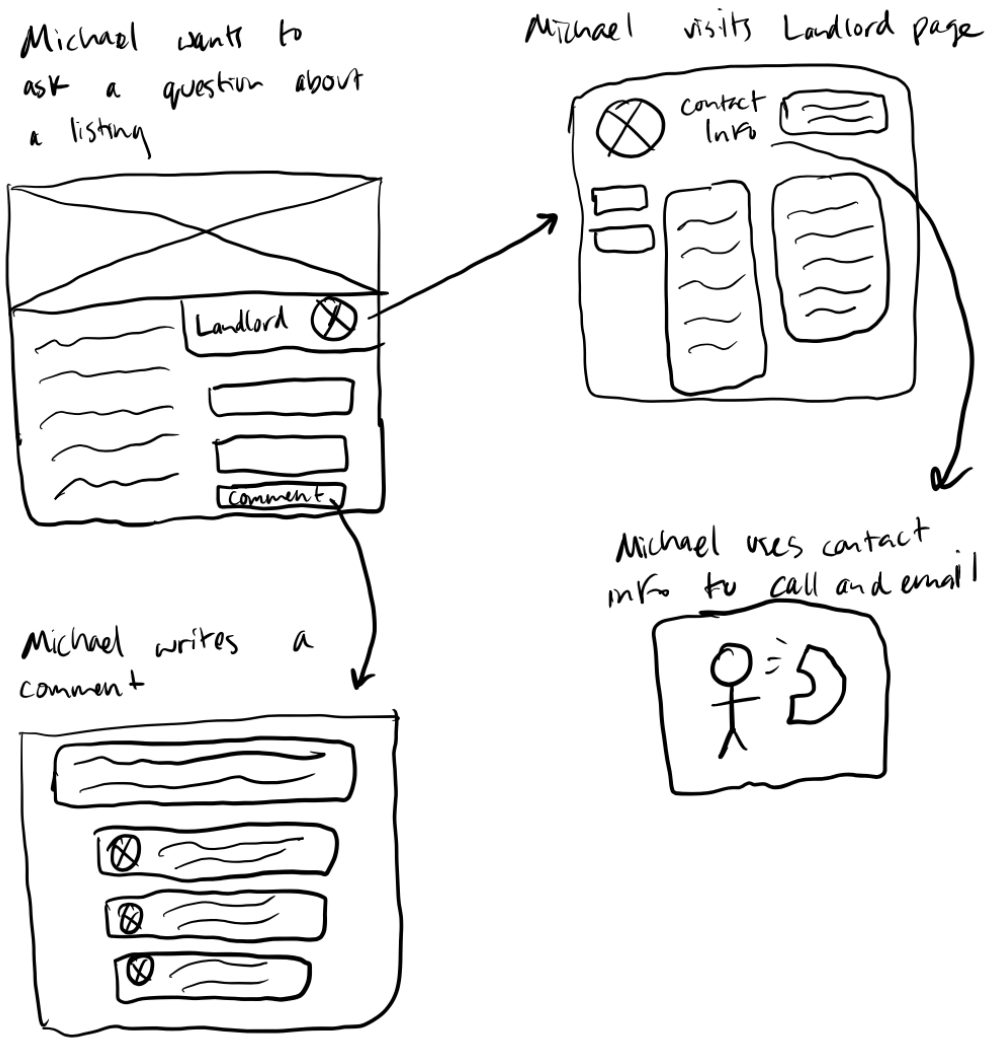
1. Search



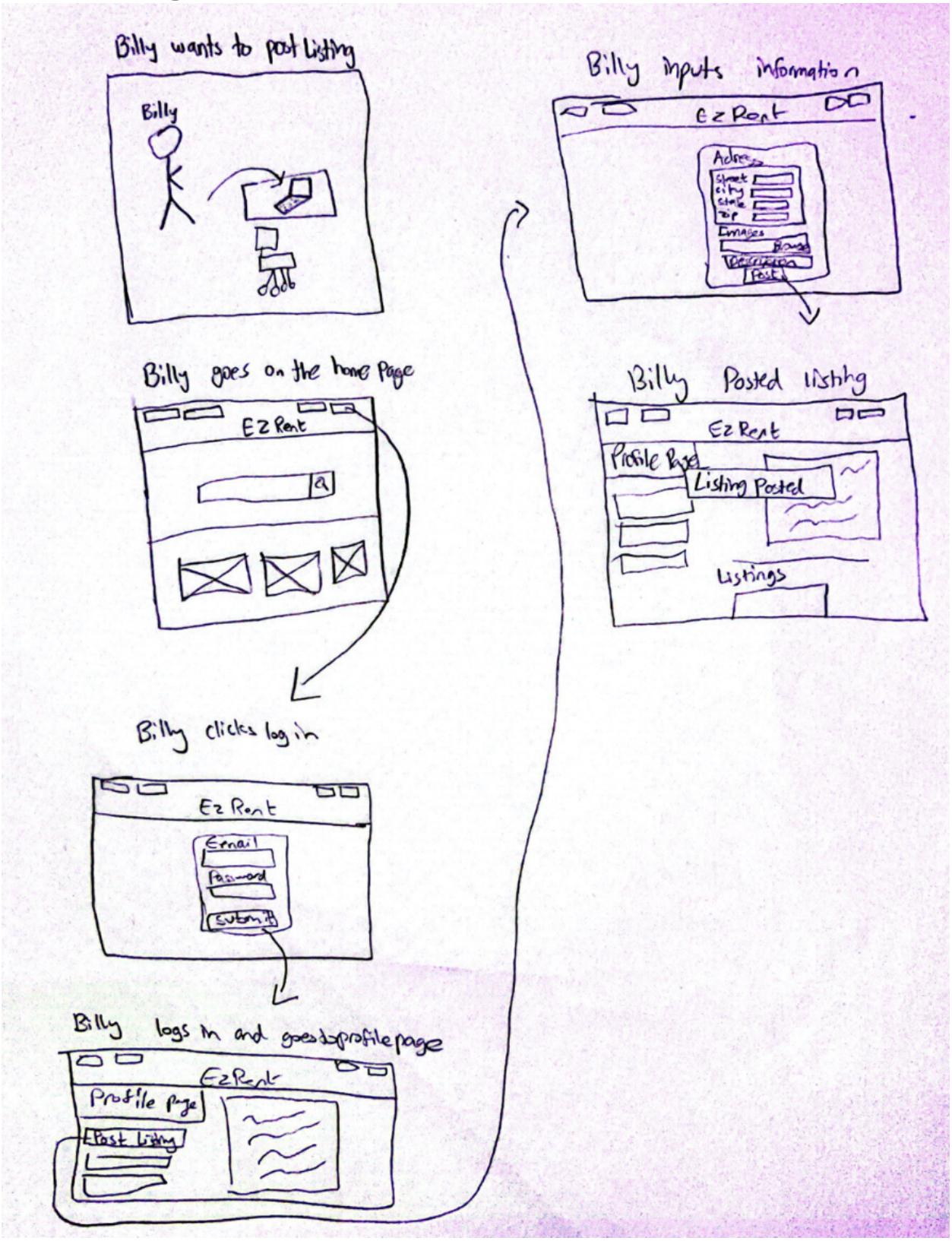
2. Login


Home Page
Register | Login
Featured
Search
Sam
Sam wants to use all of the features on the EzRent website.
Sam Logs in
Login
Login
Sam registers an account.
Register
Register
Profile
Search
Sam views profile
Profile

3. Inquiring User



4. Post Listing



4. Database Architecture and Organization

Database Requirements

1. **Registered User**
   - **1.1.** A registered user shall post zero many reviews
   - **1.2.** A registered user shall post a review for many listings
   - **1.3.** A registered user shall post a review for many landlords
   - **1.4.** A registered user shall participate in zero or many conversations
   - **1.5.** A registered user shall send many messages
   - **1.6.** A registered user shall have one full name
   - **1.7.** A registered user shall have one profile picture
   - **1.8.** A registered user shall have one email
   - **1.9.** A registered user shall have one phone numbers
   - **1.10.** A registered user shall have one password
   - **1.11.** A registered user shall file many reports
   - **1.12.** A registered user shall block many registered users
   - **1.13.** A registered user shall have many saved listing lists
2. **Landlord**
   - **2.1.** A landlord is a registered user
   - **2.2.** A landlord shall have many listings
   - **2.3.** A landlord shall be included in many landlord reviews
   - **2.4.** A landlord shall write many comments
3. **Moderator**
   - **3.1.** A moderator is a registered user
   - **3.2.** A moderator shall settle many reports
   - **3.3.** A moderator shall have an alias
4. **Listing**
   - **4.1.** A listing shall have one address
   - **4.2.** A listing shall be posted by one landlord
   - **4.3.** A listing shall have many images
   - **4.4.** A listing shall have one description
   - **4.5.** A listing shall have one price
   - **4.6.** A listing shall have a number of bedrooms

4.7. A listing shall have a number of bathrooms

4.8. A listing shall have a size

5. Review

5.1. A review shall be posted by one registered user

5.2. A review shall have many reports

5.3. A review shall have one rating

5.4. A review shall have one description

5.5. A review shall have many comments

5.6. A review shall have many pictures

6. Landlord Review

6.1. A landlord review is a review

6.2. A landlord review shall include one landlord

6.3. A landlord review shall include at most one listings

7. Listing Review

7.1. A listing review is a review

7.2. A listing review shall include one listing

8. Conversation

8.1. A conversation shall have two registered users

8.2. A conversation shall have many messages

9. Message

9.1. A message shall be sent by one registered user

9.2. A message shall have a text message

9.3. A message shall belong to one conversation

10. Picture

10.1. A picture shall be posted by one registered user

10.2. A picture shall have a name

11. Comment

11.1. A comment shall be posted by one registered user

11.2. A comment shall reply to zero or one comments

11.3. A comment shall have a number of likes

11.4. A comment shall have a number of dislikes

12. Review Comment

12.1. A review comment is a comment

12.2. A review comment shall be written under one review

13. Listing Comment

13.1. A listing comment is a comment

13.2. A listing comment shall be written under one listing

14. Saved Listing List

14.1. A saved listing list shall have many listings

14.2. A saved listing list shall belong to one registered user

14.3. A saved listing list shall have a name

15. Report

15.1. A report shall be sent by one registered user

15.2. A report shall be settled by one moderator

15.3. A report shall refer to one review

15.4. A report shall have one reason

15.5. A report shall have one verdict

Entities and Attributes

RegisteredUser

- reg_user_id: int, primary key
- time_created: datetime
- name: varchar(100)
- email: varchar(100)
- phone: varchar(45)
- password: varchar(100)
- profile_picture_fk: int, references Picture, foreign key
- bio: varchar(255)
- role: int

Landlord

- reg_user_fk: int, primary key, references RegisteredUser
- rating: decimal(2,1)
- num_reviews: int

Moderator

- reg_user_fk: int, primary key, references RegisteredUser
- username: varchar(100)

Listing

- listing_id: int, primary key

- landlord_fk: int, references Landlord
- price: decimal
- description: text
- street_number: int
- street: varchar(100)
- city: varchar(100)
- state: varchar(2)
- address_line_2: varchar(16), null
- zip_code: int
- rooms: int
- baths: int
- pets: bit
- size: int, null
- type: varchar(45), null
- time_created: datetime

Review
- review_id: int, primary key
- reg_user_fk: int, references RegisteredUser
- rating: int
- description: text
- time_created: datetime

LandlordReview
- review_fk: int, primary key
- landlord_fk: int, references Landlord

ListingReview
- review_fk: int, primary key
- listing_fk: int, references listing

Comment
- comment_id: int, primary key
- author_fk: int, references RegisteredUser
- reply_fk: int, references Comment, null
- message: varchar(100)
- thumbs_up: int
- thumbs_down: int
- time_created: datetime

ReviewComment
- comment_fk: int, primary key
- review_fk: int, references Review

ListingComment
- comment_fk: int, primary key
- listing_fk: int, references listing

SavedListing
- saved_id: int, primary key
- reg_user_fk: int, references RegisteredUser
- listing_fk: int, references listing
- time_saved: datetime

Block
- block_id: int, primary key
- reg_user_fk: int, references RegisteredUser
- blocked_user_fk: int, references RegisteredUser

Ban
- ban_id: int, primary key
- mod_fk: int, references Moderator
- reg_user_fk: int, references RegisteredUser
- reason: text

Report
- report_id: int, primary key
- poster_fk: int, references RegisteredUser
- review_fk: int references Review
- description: text
- time_created: datetime
- verdict: enum:(warning, ban, okay)
- moderator_fk: int, references RegisteredUser
- time_settled: datetime, null
- time_created: datetime

Picture
- picture_id: int, primary key
- file_name: varchar(100)
- poster_fk: int, references RegisteredUser

Profile Picture

- picture_id: int, primary key, references Picture
- reg_user_fk: int, references RegisteredUser

Listing Picture

- picture_id: int, primary key, references Picture
- listing_fk: int, references Listing
- order: int

Review Picture

- picture_id: int, primary key, references Picture
- review_fk: int, references Review
- order: int

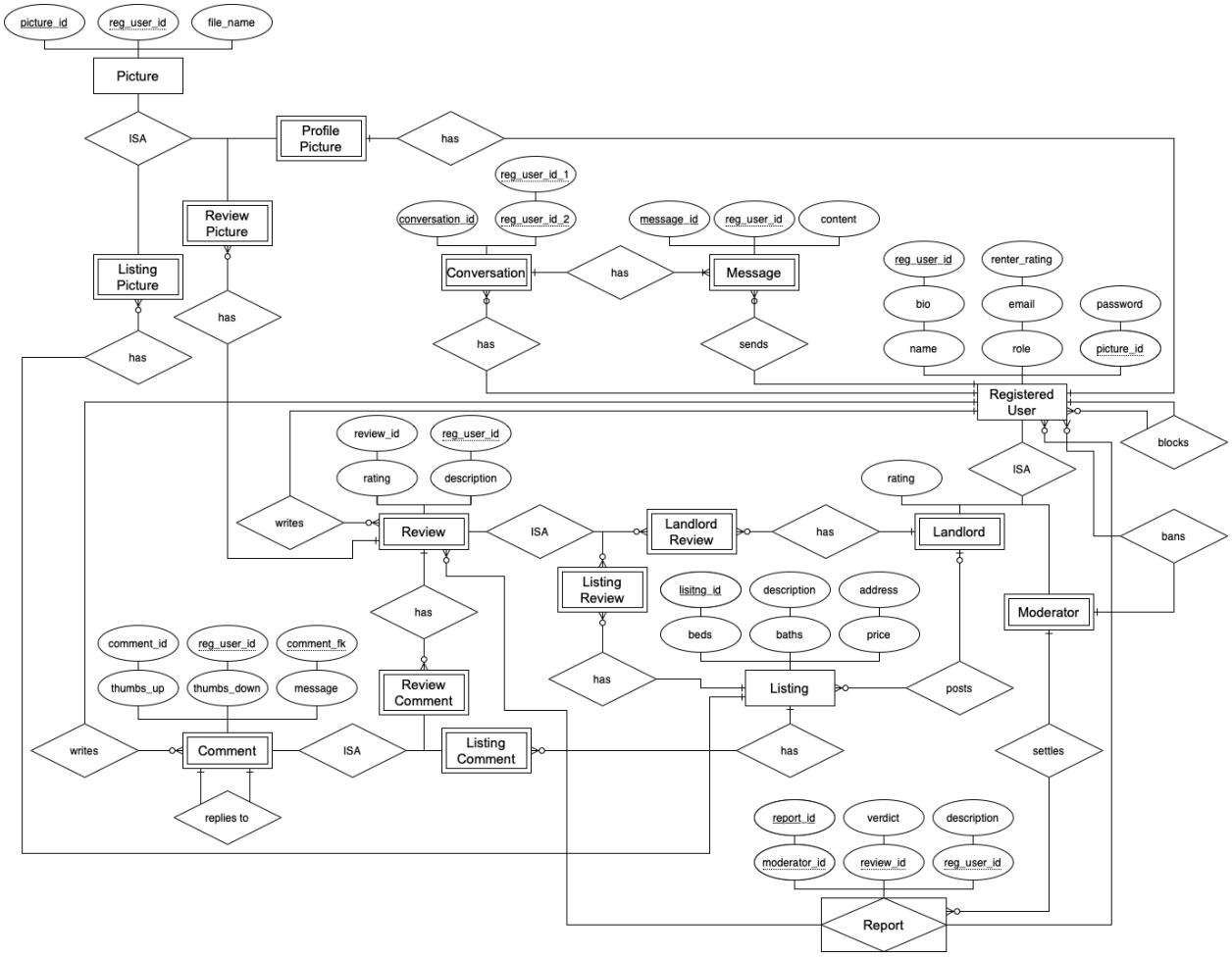
Conversation

- conversation_id: int, primary key
- reg_user_1_fk: int, references RegisteredUser
- reg_user_2_fk: int, references RegisteredUser

Message

- message_id: int, primary key
- conversation_fk: int, references Conversation
- author_fk: int, references RegisteredUser
- reply_fk: int, references Message, null
- content: text
- time_sent: datetime

Entity Relationship Diagram



DBMS

MySql is a type of structured database which stores only structured data. We chose MYSQL for our project because of Node. js performs well with relational databases like MySQL. Because we work on relation data and MYSQL is very flexible, so MYSQL will be the best for learning.

Media Storage

Images will be kept in our file structure. And then the file names and locations of those images will be stored in the database. We will use the node.js module Multer to store these files.

Search/Filter Architecture and Implementation

Listings
Users will search listings by address, by zip code, by location, or by city. The default search results will be sorted by rating. If multiple ratings have the same rating, they will be sorted by the newest listings. If the listings were posted at the exact same time, they will be sorted by address. If nothing shows up, the site will offer other recommended listings.
Users will then be able to apply multiple filters to their search:

Minimum Price	Users will be able to set a minimum price, which will exclude any listings below said filter
Maximum Price	Users will be able to set a maximum price, which will exclude any listings that exceed said filter
Listing Rating	Users will be able to apply a rating filter which will portray listings of desired rating and above.
Landlord Rating	Users will be able to apply a rating filter which will display landlords of specific ratings and above. Listings shown will exclude ratings below the applied filter.
Beds	Users will be able to filter out bedrooms based on desired amount. Listings shown will exclude any amount of bedrooms below desired quantity.
Baths	Users will be able to filter out bathrooms based on desired amount. Listings shown will exclude any amount of bathrooms below the desired quantity.

Users can also be able to order their search results one attribute at a time:

Price	Users can order the results by most expensive or by least expensive.
Listing Rating (default)	Users can order the results by highest rated listing.
Landlord Rating	Users can order the results by the highest rated landlords.
Time Posted	Users can order the results by the most recent listings or by the oldest listings.
Number of Reviews	Users can order the results by the most number of reviews.

Landlords

Users will search Landlords by location including address, city, zip code or by name. The top rated Landlords will be the first results the user will see; this will be our default to make sure the user has the most optimal experience.
Users won’t be able to apply filters to this search.

5. APIs and Main Algorithms

APIs

Get filters: will get the settings from the filters sidebar and feed it to the the listing search engine
Get listing search results: will get the search terms from the search bar and feed it to the search engine
Get landlord search results: will get the search terms from the search bar and feed it to the search engine
Get listing page: will get a listing page from the database and return a page with its data
Get landlord page: will get a landlord page from the database and return a page with its data
Get renter page: will get a renter page from the database and return a page with its data
Post listing: will post a listing to the database with the parameters that a landlord sets
Delete listing: will delete a given listing from the database
Post listing review: will post a review to the database with the parameters that a renter sets
Post landlord review: will post a review to the database with the parameters that a renter sets
Post listing comment: will post a comment to the database with the parameters that a registered user sets
Post review comment: will post a comment to the database with the parameters that a registered user sets
Post account: will post a registered user to the database with the parameters that a user sets
Log in to account: will get the user information from the database and create a session
Log out of account: will delete a session
Post saved listing: will post a saved listing to the database from a listing that a renter chooses

Algorithms

Most Relevant Search

1. Check empty search
2. Check if address format
3. For each element
    a. If 2 chars, checkState
    b. If number
        - If array[itl] is string, street # - SELECT street# return
          else zip code - return
    c. SELECT* FROM Listing WHERE- foreach column(address)
4. For each 2 elements
    a. SELECT* FROM Listing WHERE- foreach column(address)

Featured Landlords/ Top 3 Landlords

1. Sort Landlord Table by Rating
2. Pick first 3
    a. If same rating, pick random

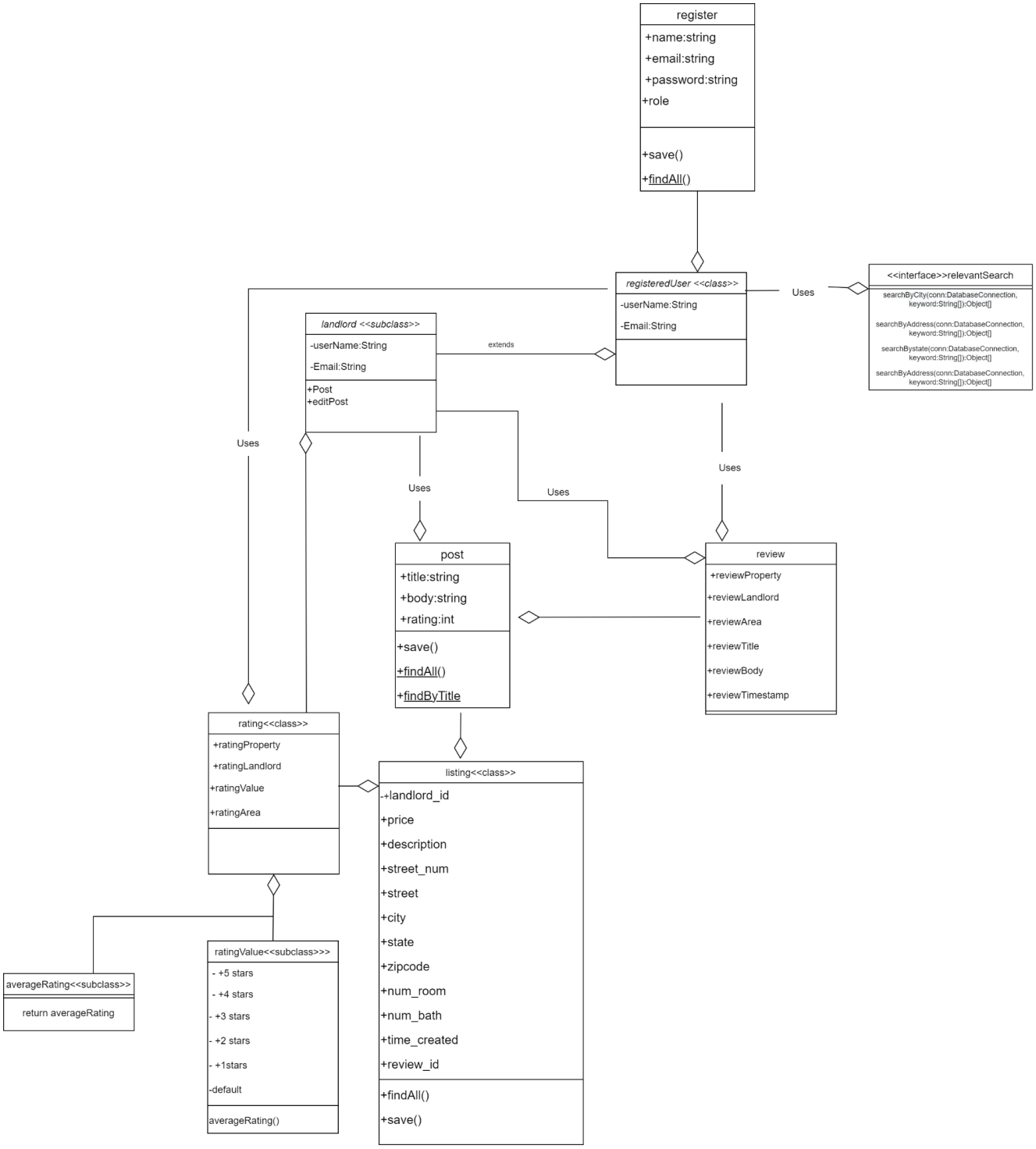
Top Reviews/ Landlord Page

1. Sort reviews by rating
2. Pick first 3
    a. If same rating, pick random

Overall/ Average rating calculator

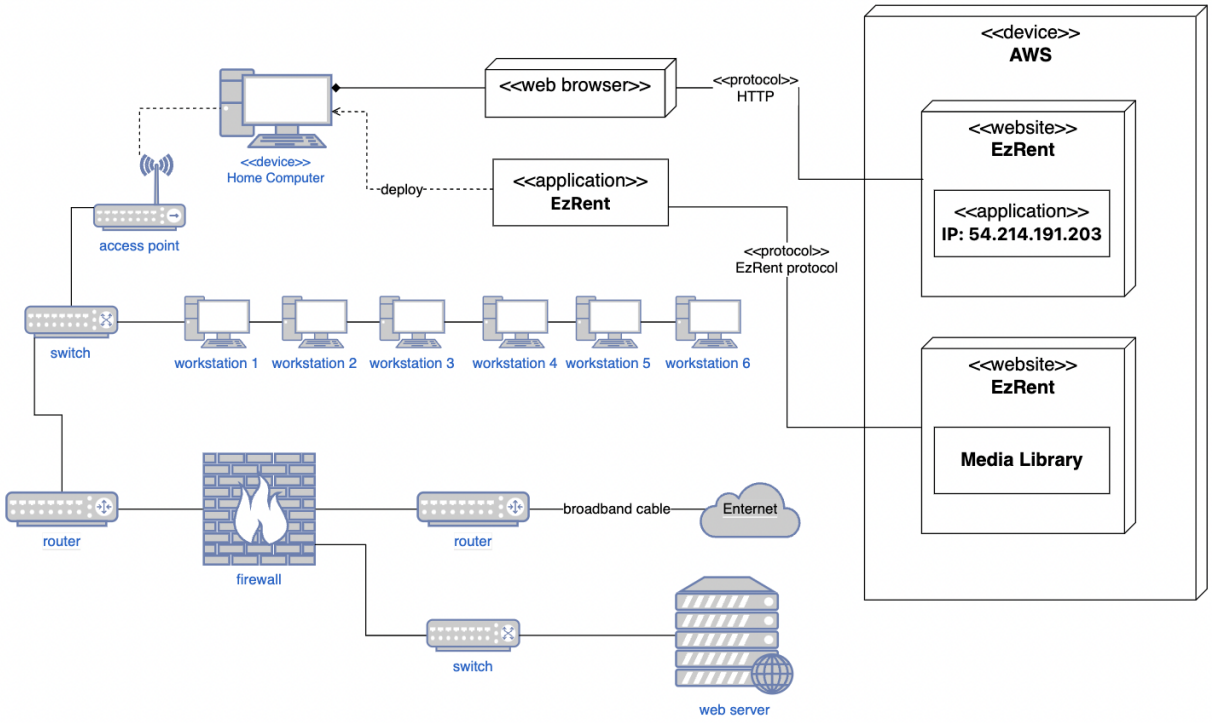
1. Add all ratings
2. Divide by number of ratings

6. UML Diagram

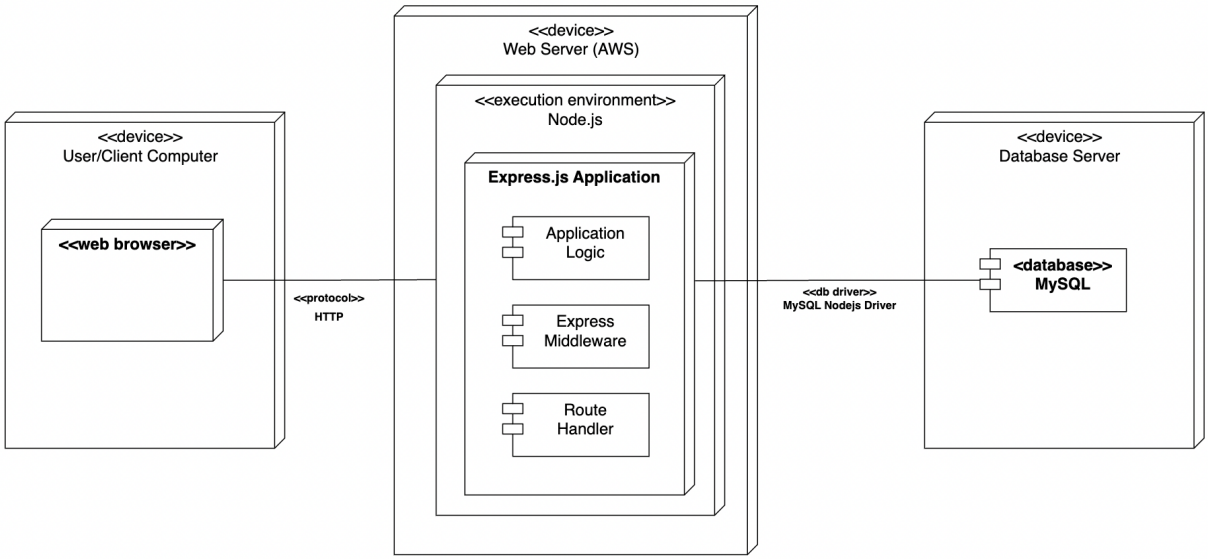


7. High Level Application Network and Deployment Diagrams

Network Diagram:



Deployment Diagram:



8. Key Risks

- Skills Risks
    a. It's our first time working in a software engineering environment. Software engineering is different from software development, so we might get certain steps wrong in the process of making this product.
        - We will address this issue by researching the proper ways of engineering a product. In addition, we will regularly give each other feedback so we can iterate on better designs.
- Schedule Risks
    a. We have a lot of functional requirements and we might not get to implementing all of them in time. If we are in the middle of implementing something that we don't have enough time for then we might have to drop it at the risk of affecting other parts of the project.
        - We can solve this by strategically prioritizing our requirements. We can also focus on one function at a time, not rushing things, so that other requirements aren't stepped on if one happens to be dropped.
- Technical Risks
    a. For our database server we are using AWS. As a team, this is our first time using this technology, so we are all pretty unfamiliar with it.
        - We can reduce this risk by researching how to use AWS and its contents. We can also watch tutorials and ask specialists for help when we are stuck.
    b. Using GitHub as a team is a risk because crucial lines of code can be lost if conflicts aren't merged properly. A rare possibility, but a very dangerous one is that our repository might be deleted.
        - We can reduce this risk by communicating at all times and informing all team members before committing or deleting any changes. This way, all team members will look over the changes that need to be made and see if they could affect the entire repository before actions are made.
- Teamwork Risks
    a. Not being able to meet up due to other responsibilities/priorities.
        - We can solve this by prioritizing this class over other classes due to the amount of work needed to be done in this class. We can also communicate and inform the team lead beforehand if one can't attend a meeting. This way, the team lead could then reschedule a meeting time where all members could attend.
    b. Miscommunication can cause work to be overridden or accidentally deleted. We can be either working over somebody else's work or creating

more work for ourselves by working on the same tasks. It can also cause deadline issues if scheduling is not communicated properly.
    - Our communication skills must be at a high level in order for us to not run into any issues. This is why we must have meetings and ways to communicate at all times when needed. We want to make sure we ask for help whenever it is needed.
- Legal/Content
    a. For the website, due to our services being offered within the realestate atmosphere, it is imperative to have quality pictures which will further more represent our overall motive.
        - With this being said, many pictures online are copyrighted so it would be ideal to actually take our own pictures instead rather than use licensed pictures. Another way around this would be to use royalty free pictures found on certain sites. Otherwise, we need to ask for permission, from the picture owner, to use the copyright image that we want to use.
    b. Being a site that allows users to post pictures from their personal computers, there is a risk that they will post copyrighted photos.
        - The way our site will address this issue is by having a team of moderators identify copyrighted content and allowing them to remove it from the site.

9. Project Management

For this milestone, we used ClickUp for its project management tools. The team lead created tasks and set deadlines within the app. For tasks that could be split into subtasks, each team member assigned themselves to a subtask. For example, each team member assigned themselves to work on at least one storyboard for each function of our website. The team lead assigned all other tasks manually.

Whenever someone started a task they would set the task to “in-progress”. This would let the team lead know which tasks still need to be assigned. When they complete a task, they would set the task to “review” so the team lead could suggest changes to the work to meet the standards of the project. The team lead would then set the task to “revise” so the original assignee could make the necessary changes. After everything looks good, the task would be set to “closed” to signify that the team can move on to a new task.

In addition, the team used Discord to coordinate meetings to talk about who does what, and to do group tasks such as doing certain documentation sections together.

10. List of Team Contributions

Devon Dy-Liacco	Documentation: ● Delegated tasks and set deadlines ● Storyboard #3 and #6 ● Proofreading ● Project management
Score: 8/10	Vertical Prototype: ● Delegated tasks and set deadlines ● Proofreading
Praise O Eubany	Documentation: ● Prioritized functional requirements ● Storyboard #1 ● Key risks ● Search/filter architecture ● UML Diagram ● Database entries
Score: 8/10	Vertical Prototype: ● Github management
Youssef Hammoud	Documentation: ● Prioritized functional requirements ● Database requirements ● Storyboard #4 ● Key risks ● Network diagram ● Deployment diagram
Score: 9/10	Vertical Prototype: ● Initial set up ● Home page (frontend) ● Search results page (frontend/backend) ● Filters (frontend) ● Styling
Issa Shihadeh	Documentation: ● Database entities and attributes ● Database ERD ● Storyboard #7 and #8 ● Proofreading
Score: 7/10	Vertical Prototype:

	● Routing
Tung Nguyen	Documentation: ● Database entities and attributes ● Database requirements
Score: 8/10	Vertical Prototype: ● Search results page (backend) ● Database tables ● Models and controllers ● Github management
Ricardo Lopez	Documentation: ● Created document outlines ● Data definitions ● Prioritized functional requirements ● Storyboard #2 and #5 ● Key risks ● Proofreading ● Search/filter architecture ● UML set up ● Main algorithms ● Database entries
Score: 8/10	Vertical Prototype: ● Initial set up ● Filters (frontend)