

Software Engineering CSC648/848 Spring 2019

BetterHome

Spring 2019

Team 43

Taso Grigoriou (team lead - agrigori@mail.sfsu.edu)

Henok Kassegn (frontend lead)

Sawyer Nixon (backend lead)

Cole (Michael) Tormey (github master)

Austin Sy-Velasco (backend / DB manager)

Liwang Gao (frontend)

Milestone 2

04/19/2019

History Table (revisions):

Revisions	Date
M2V2	04/19/2019
M2V1	04/02/2019
M1V2	03/20/2019
M1V1	03/17/2019

Table of Contents:

1. Data Definitions V2
2. Functional Requirements V2
3. UI Mockups and Storyboards
4. High-level Architecture, Database Organization
5. High-level UML Diagrams
6. Identify actual key risks for your project at this time
7. Project Management

1. Data Definitions V2:

Admin:

User from BetterHome team that can maintain use information and ensure listings posted on our website are valid.

Client (Registered User):

User that has registered with the site, can access all listings on the site, can message the listing owners, and can post their own listings.

Guest User:

User can see the listings on the site without making an account, but cannot message listing owners nor access their contact information.

Privacy policy:

Policy to maintain and protect our user information.

Map View:

A list of all available home/apartment listings in the designated area.

Handicap:

Indicator on listings to show that they have handicap accessible pathways.

Listing:

- Photo:

Some photos to showcase the house/apartment.

- Price/Rate:

A general idea of how much each listing is.

- Address:

Area for where the listing is located.

BART Accessible:

Within a close proximity to a BART Station.

Availability:

Indicator to show that if the listing is available (could signify that an admin should take down the listing OR can just scrap this and have the owner take down the listing themselves).

Reseller:

A user that is planning to purchase houses/apartments not for their own use, but to improve and then resell back on the market.

Handicap User:

A user that would like special accommodations with their house: wheelchair accessible (elevator to apartment, ramp) or within a relatively close distance to a hospitable or care unit.

Landlord:

A user that will be posting listings on the website in order to get a buyer's attention.

Rooms:

Amount of rooms available in the property listing.

Parking:

How much parking comes with the property.

Profile:

A place on our website that will show personal information of the user.

Help:

A brief explanation of how our website works.

2. Functional Requirements V2:

1st Priority:

For Admin:

1. Admin shall be able to update/maintain user information.
2. Admin shall be able to delete Landlord and Clients.
3. Admin shall be able to lock and unlock Landlord and Clients.
4. Admin shall be able to find Landlord and Clients.
5. Admin shall be able to view gallery.
6. Admin shall be able to delete photos.
7. Admin shall be able to reset password for both Landlord and Clients.
8. Admin shall be able to access username and email of Landlord and Clients.

For Landlord:

1. Landlord shall be able to browse the website without login.
2. New Landlord shall be able to register.
3. Registered Landlord shall be able to post property descriptions.
4. Registered Landlord shall be able to post the location of the property.
5. Registered Landlord shall be able to post the price of the property.
6. Registered Landlord shall be able to post the city and zipcode of the property.
7. Registered Landlord shall be able to select if the property is a house or apartment.
8. Registered Landlord shall be able to select if the property is wheelchair accessible.
9. Registered Landlord shall be able to login.
10. Registered Landlord shall be able to add photos.
11. Registered Landlord shall be able to delete photos.

For Client:

1. New clients shall be able to register.
2. Registered clients shall be able to login.
3. Registered clients shall be able to sign out.
4. Registered clients shall be able to modify their profile.
5. Registered clients shall be able to add the listing as favorite .
6. Registered clients shall be able to revisit the favorite .
7. All clients shall be able to browse the website without login.
8. All clients shall be able to view the sale listing.
9. All clients shall be able to view the sold listing.
10. A free text search box shall be displayed to all clients.
11. All clients shall be able to search the listing by city.

12. All clients shall be able to search the listing by price range.
13. All clients shall be able to search the listing by property type.
14. All clients shall be able to modify the last search.
15. The search results shall be able to be sorted by relevance.

2nd Priority:

For Admin:

1. Admin shall be able to filter data by property type, status, or date posted.
2. Admin shall be able to review flags, and take the appropriate action.
3. Admin shall be able to run data analysis on user data to understand how the application is being used.

For Client:

1. The default display in the free text search box shall be “address, neighborhood, zip”.
2. All clients shall be able to filter the search result by relevance.
3. All clients shall be able to view the search results in a map view based on current location.
4. All clients shall be able to contact the landlord.
5. All clients shall be able to contact the admin.
6. Listings that are similar to a viewed listing should be displayed to all clients
7. A Q&A page shall be provided for all clients.
8. All handicap users shall be able to search for listings by accessibility.
9. All resellers shall be able to repost the listing onto the site.
10. The application shall keep track of registered user history search.
11. Registered clients shall be given a list of checkbox to select and search (Advanced search).

For Landlord:

1. Registered Landlord shall be able to contact registered clients.
2. Registered Landlord shall be able to view gallery.
3. Registered Landlord shall be able to contact admin.
4. Registered Landlord shall be able to change the property status to available, pending, and sold.
5. Registered Landlord shall be able to update price, location, and pictures.
6. Registered Landlord shall be given a list of checkbox to select if their property has access to wheelchair, BART, parking, etc.

For guest User:

1. Guest users shall be able to register.
2. Guest users shall be able to search listing by city.
3. Guest users shall be able to search listing by price range.
4. Guest users shall be able to search listing by property type.
5. Guest users shall be able to modify the last search.
6. Guest users should be able to view the search results in a map view based on current location.

For Apartment Searcher:

1. Apartment searcher shall be able to search apartments only available for rent
2. Apartment searcher shall be able to search the apartment by number of bedrooms
3. Apartment searcher shall be able to search the apartment by monthly payment.
4. Apartment searcher shall be able to sort by the results by relevance.
5. Apartment searcher shall be able to filter the results by relevance.
6. Apartment searcher shall be able to view the apartment features.

3rd Priority:**For handicap:**

1. Handicap users shall be able to flag their profile, denoting a need for handicap-specific accessibilities.
2. Handicap users shall be able to specify in their profile the exact nature of their accessibility needs.
3. Handicap users shall be able to specify in their search that they require a ramp.
4. Handicap users shall be able to specify in their search that they require a ground floor unit.
5. Handicap users shall be able to search the listing by number of floors.

For reseller:

1. Resellers shall be able to access the original post for the property they purchased.
2. Resellers shall be able to modify and repost the original listing with updated information.
3. Original listing shall be linked to new listing so other users may view and verify improvements and/or alterations to property.

Non-Functional Requirements:

Security:

1. Login shall be required for Clients and Admins.
2. Username shall be the Client's email.
3. Password shall be encrypted when stored.
4. Client's session shall end upon leaving the site.
5. Client's session shall only end by code design.

Performance:

1. Loading time for site shall be less than 3 seconds for any screen.

Capacity:

1. The total data storage allowed by the web site shall not exceed of 80 % of the server capacity for this site.
2. The web site shall be prepared to support scalability for adding future new features.
3. The web site shall be capable to handle at least 50 Clients simultaneously.

Reliability:

1. Downtime for maintenance shall be less than 3 hours per month.
2. Downtime for maintenance shall not affect the main functionality of the site.
3. In all cases, downtime for maintenance shall be informed to the Client through email.

Recovery:

1. In a total failure case, the whole site should be put down to revision.
2. If broken, the mean time to recovery shall not exceed one day.

Data Integrity:

1. Database tables shall be backed up every day.
2. Administrator shall be able to execute a recovery when needed.
3. Image Sizes shall be limited up to 1 megabyte.
4. Images shall be uploaded in correct format (jpg, jpeg, or pdf) to the server.
5. Links to images on the server shall be uploaded to the database.

Compatibility:

1. The site shall be compatible with the last version of Microsoft Edge browser (44.17763).
2. The site shall be compatible with the last version of Safari browser (12).
3. The site shall be compatible with the last version of Firefox browser (65.0.2).

4. The site shall be compatible with the last version of Chrome browser (7.3).
5. Third party applications shall not be able to modify any content that may affect the site compatibility.
6. The site shall be ready to support with any or minimal changes any other compatibility that may be added in future versions.
7. The site should be compatible to escalate to new relational databases.

Conformance with Coding Standards:

1. Architecture and design standards shall meet all the requirements listed under the High Level Architecture section of this document.
2. Only working code that meets all the code standards shall be submitted to the project repository.
3. Any working code shall be tested and debugged before being considered working code.
4. Any internal errors or exceptions returned by the code shall be stored in a log.
5. Any error that may affect the functionality of the site shall be reported to the Client.
6. Any error shall be handled in a way that does not affect the functionality of the site.
7. The whole production cycle of this site shall be finished 2 weeks before the delivery date.
8. This site shall not be launched without all the priority one features completed and tested.

Look and Feel Standards:

1. The application and its layouts shall look professional.
2. The site shall be simple enough to handle by all the parties involved.
3. Elements on screen shall have the correct density to meet the compatibility standard of the browsers.
4. Elements on screen shall have rich and beautiful colors for Client delight.
5. The site shall be able to work correctly without mouse interaction.
6. The site shall be able to work correctively without keyboard interaction.
7. Elements in screen shall be resized automatically without Client interaction when being loaded in all the different platforms supported by the site.

Internationalization / Localization Requirements:

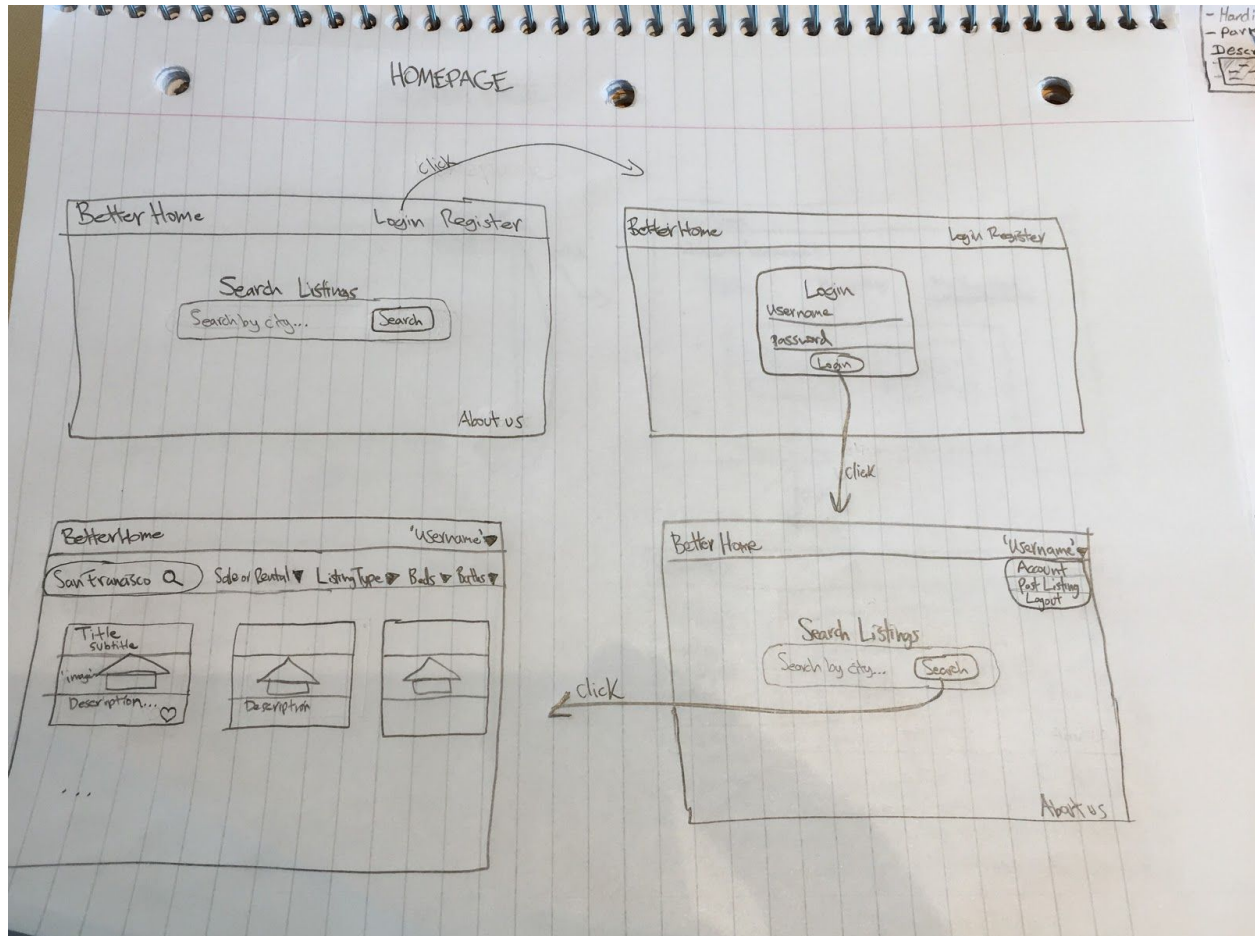
1. Default language shall be English.
2. The site shall support scalability to add other languages.
3. The site shall support geolocation in order to show listing locations.
4. Any copyrighted material shall be immediately be taken down upon reception of an official DMCA takedown request.

Website Policies:

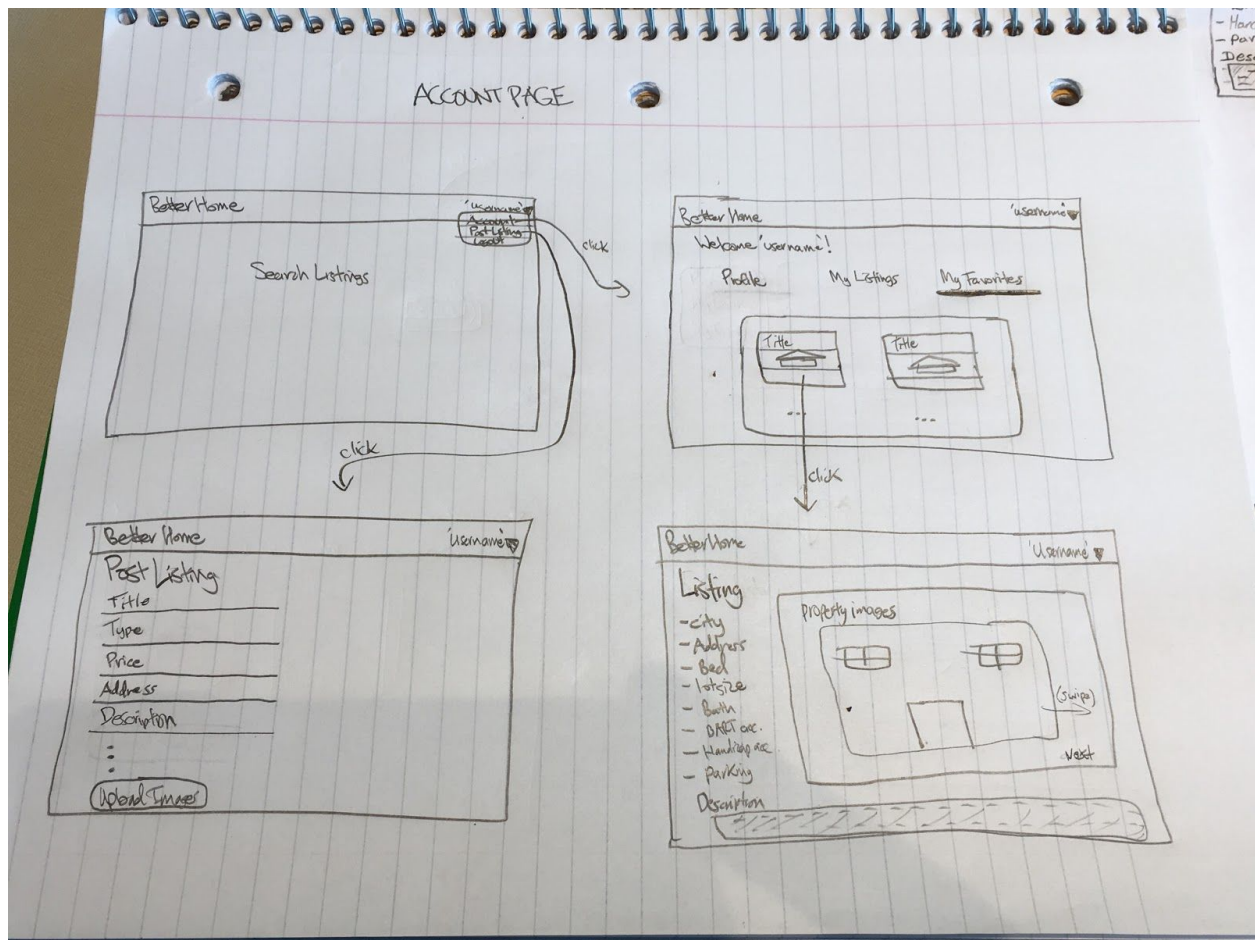
1. A link to the policies of this site shall be always visible in all its pages to be accessible by all the parties.
2. Clients' data shall not be sold to third parties.
3. Clients and Landlords data that do not add any functionality to the system shall not be collected.
4. Clients that post inappropriate listings(false listings/ copyright images/ sexual images) shall have their postings taken down.

3. UI Mockups and Storyboards

Login flow:



Account Flow:

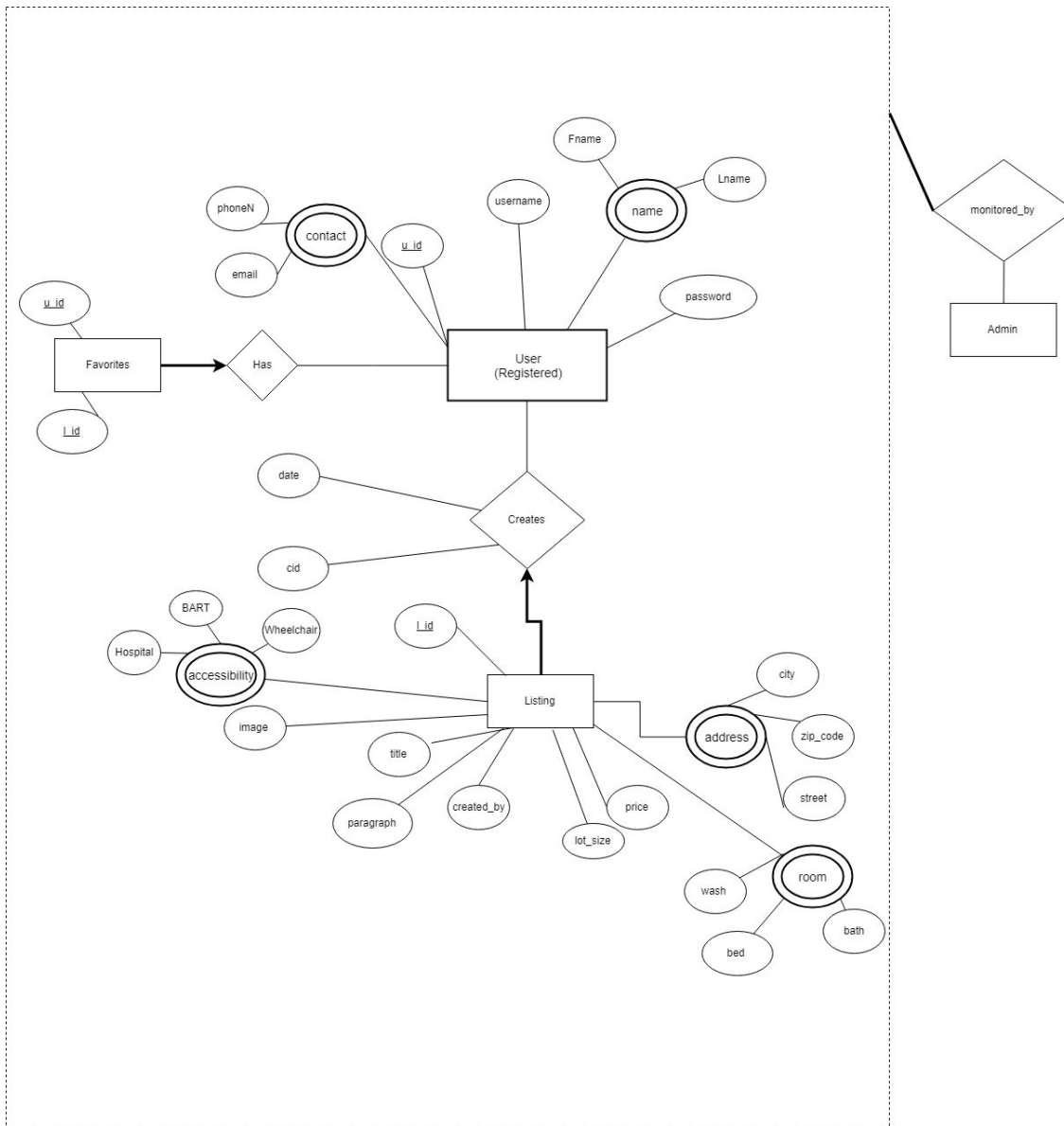


4. High-level Architecture, Database Organization:

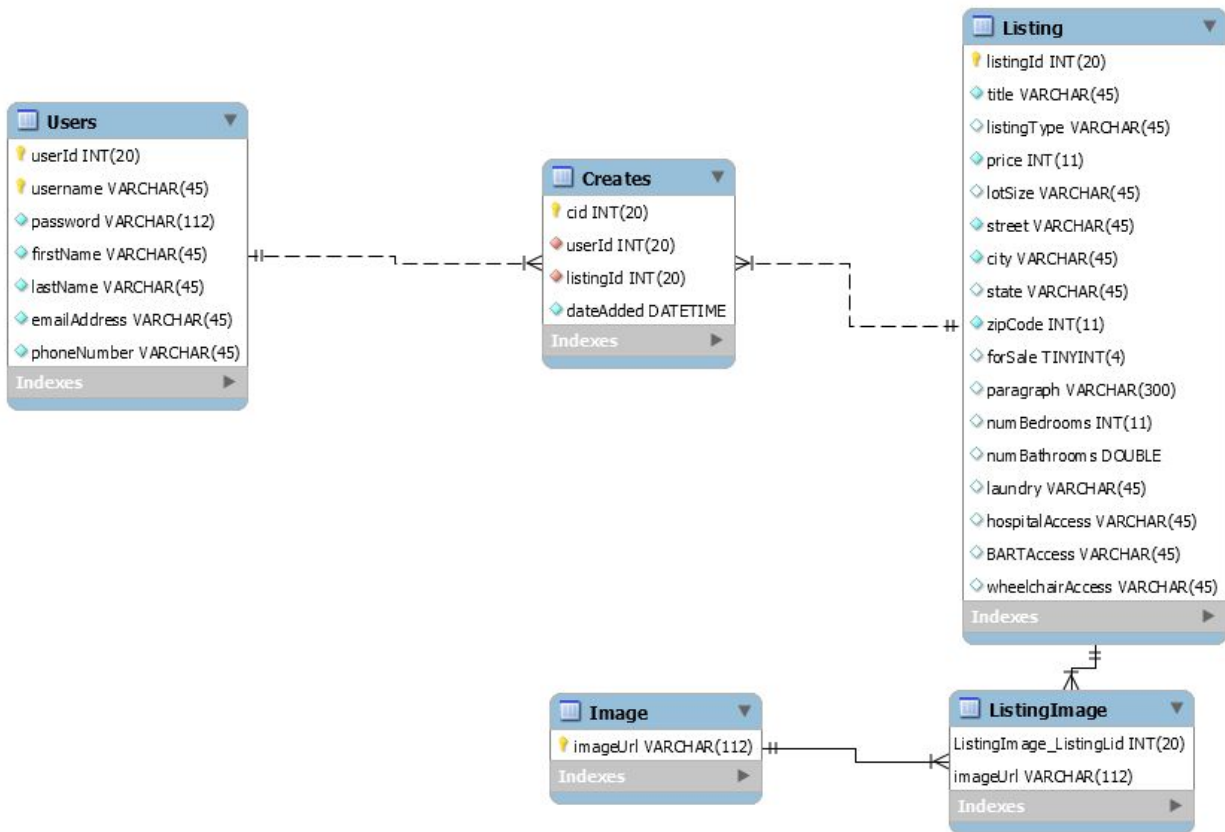
Business Rules:

1. Users can create multiple listings on the website.
2. Users can view multiple listings with a specific amount of rooms.
3. A listing must have only one description describing its characteristics.
4. A listing can be displayed on multiple listing boards.
5. Admins must monitor all listings posted by all users.

ERD Diagram:



DB Diagram:



DBMS:

We chose MySQL as our Database Management System since it provides great support alongside MySQLWorkbench and the workflow is very intuitive and efficient.

Media Storage:

Images (.jpeg, .png, .tiff, .bmp) will be the only form files that the user can input into the database. They will be stored as URLs on the “Image” table and be a primary key in “ListingImage” that will connect it to the “Description” table.

Search/Filter Architecture and Implementation:

Our search functionality requires a city name in the free text search box from the frontend. We utilize the SQL LIKE clause in this format ‘%{city}%' which finds all Listing.city values that have {city} in any position, provided by the request body.

The listing type, number of bedrooms and bathrooms are also (optionally) provided in the search query in order to filter out Listings which don’t match the query using the WHERE clause.

All together the search query looks something like this (JavaScript/SQL):

```
let sqlQuery = `SELECT * FROM Listing WHERE city LIKE '%${req.body.cityName}%'
AND listingType = '${req.body.listingType}' AND numBedrooms = ${req.body.numBedrooms}
AND numBathrooms = ${req.body.numBathrooms}`;
```

API's:

We have several endpoints provided on our backend service that allow the angular services to make HTTP requests in the form of CRUD operations (create, retrieve, update, delete). They all begin with `/api` so as to not point to the static files in our public directory. Some examples are:

- **POST `/api/search`**
 - Pass in a search query into the request body and retrieve an array of Listing objects that make the given query
- **GET `/api/listing/:id`**
 - Pass in a Listing id to the GET request header to retrieve a single Listing from the database
- **POST `/api/login`**
 - Pass in a username and password request body, the backend queries the DB for the User with the provided username. Then, compare the encrypted password with password from the request body and send a success/failure response back

Non-Trivial Algorithm:

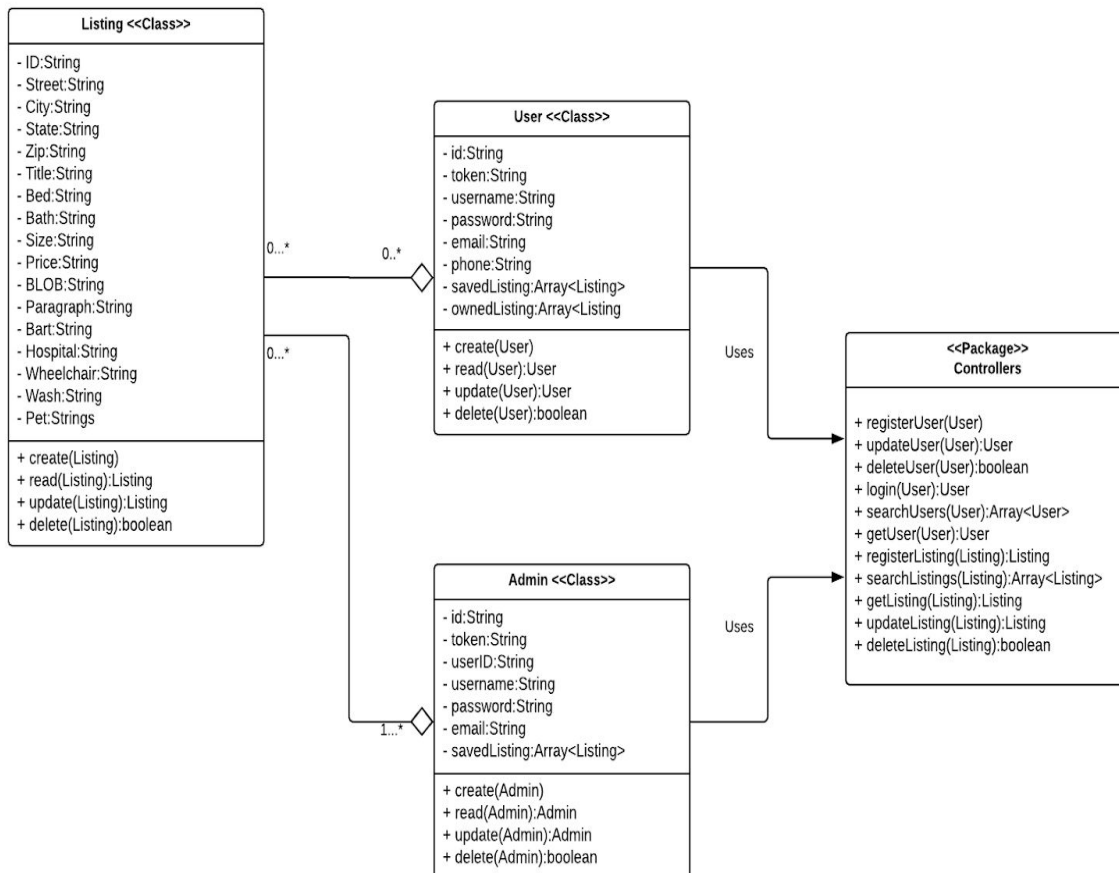
No non-trivial algorithm is required nor has been divized as of yet for our project.

Software Tools and Frameworks:

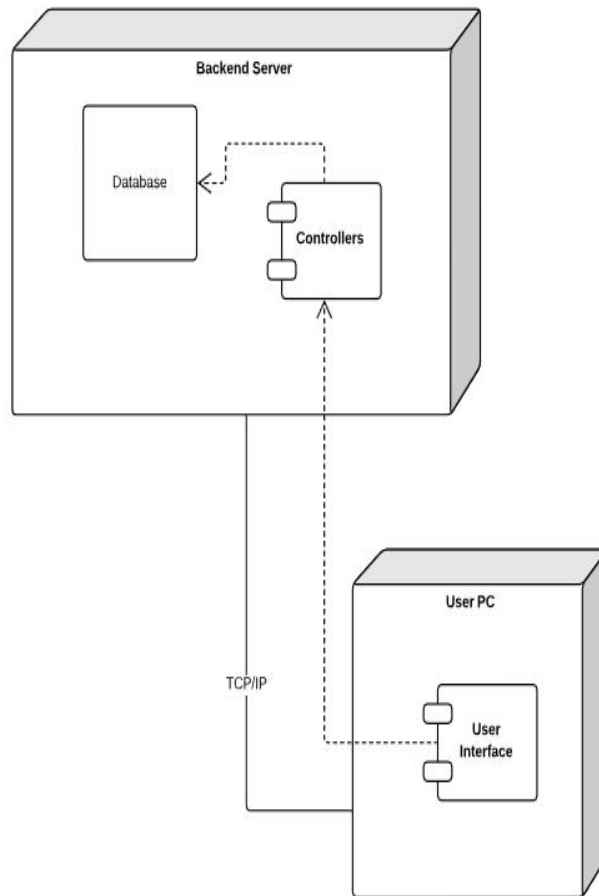
We have not added any additional SW tools or frameworks for our project.

5. High-level UML Diagrams:

Better Home UML Class Diagram



Deployment Diagram



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

- Skills risks (do you have the right skills):
 - As a whole, the frontend and backend teams are developing the necessary skills in order to take the database and UI mockup design and learn the frameworks in order to implement the requirements in code
- Schedule risks (can you make it given what you committed and the resources):
 - Due dates for the milestones so far have felt tight, however with proper time management and getting started early as we have, we should be in good shape
- Technical risks (any technical unknowns to solve):
 - With the architecture and structure provided in the codebase by the team lead, we don't foresee any legitimate technical issues during development
- Teamwork risks (any issues related to teamwork):
 - The team has well defined roles (frontend / backend) and is operating well on their own, however there could be more time spent to meet up outside of lecture in order to sync up
- Legal/content risks (can you obtain content/SW you need legally with proper licensing, copyright):
 - There are risks of copyrighted images being uploaded to our server and how we will address this, however this is the only major hurdle.

7. Project Management:

We have decided to use Trello for agile development and project management. We have split up the work by categorizing them into user stories inside the following sections:

- Defined
- In Progress
- Completed
- Testing
- Accepted

You may view and track the progress of our user stories by visiting our Trello board by clicking the link below.

<https://trello.com/b/8fSVt9YJ/csc-648-betterhome-team43-board>