

**Software Engineering CSC648/848 Spring 2019**

**~ Project Pegasus ~**

Team Number 104

Brenna Tirumalashetty - brennag@mail.sfsu.edu

Omar Alaniz

Zachary Prince

Quan Lu

Adan Guanlao

Fatma Khan

Wafi Mohamed

**Milestone 2**

*March 25th, 2019*

<b>Revision Number</b>	<b>Date</b>	<b>Notes</b>
1	March 25th, 2019	Original submission
2	April 4th, 2019	Incorporating 1st feedback
3	May 17th, 2019	Final

## 1 - DATA DEFINITIONS

- Student: a student at San Francisco State University who has been assigned an SFSU email address. Students can be current users of the Pegasus website (either registered or unregistered), future users of the website, or non-users of the website that we would like to attract. A student can be an undergraduate or a graduate, enrolled part time or full time. Student website registration status determines their access to Pegasus services.
- Domicile: a residence, which can be an apartment, house, room, garage, half-room, or other rentable dwelling. A Domicile has the following properties: address, type of property, price, amenities, parking, laundry, gym, number of occupants, etc.
- Landlord: The owner of a Domicile, which can be a private homeowner or a property management company, who does not reside at the domicile. They have an account and are logged in. They use the website to browse listings, post listings, and receive messages from registered users. Landlords have an account, but need not have a profile. Landlords can contact domicile or residence for maintaining property.
- Tenant: A Student that is a current resident of a Domicile. They have an Account and Profile, as well as the address of their current Domicile. They shall have access to the Student Neighbor Network.
- Star Tenant: A Tenant who creates and has a Listing to rent out portions of their Domicile. They may or may not share their current Domicile with other Tenants. They may also own the Domicile as a Landlord. Star Tenants have an Account, Profile, and address of their current Domicile. They shall have access to the Student Neighbor Network if they are also a Student. Their Profile shall indicate whether or not they are a Student.
- Unregistered User: a user who does not have an account or is not currently logged in. Unregistered Users can browse listings and use the features of the geographical overlay, such as viewing nearby services and calculating commute times, but may not contact any Landlord, Tenant, or Star Tenant. They shall not access any information pertaining to the Student Neighbor Network or profiles of Registered Users. They shall not access information about current residents of a domicile. Unregistered Users may, if they are Students or Landlords, create an account. Once an account has been created and verified by the administrator, the Unregistered User becomes a Registered User.
- Registered User: a user who has an account and is currently logged in. They use the website to browse listings. Registered users may be Landlords, Students, Tenants, or Star Tenants. Registered Users have an account and a profile. A registered user has access to the services that Project Pegasus has to offer. The user is able to contact any Landlord, Tenant, or Star Tenant in contrast to an unregistered user. A register user must be only SFSU students, unless, they are registering as a landlord.
- Administrator: a user who has elevated privileges including access to the website database. They use the website to perform administrative tasks and to moderate Listings and Registered Users. An administrator is granted admin rights by another admin and/or the company itself. The administrator is a user but they do not fall under registered or unregistered.

- **Student Neighbor Network**: A network of Registered Students that provides a platform for Registered Students to communicate with each other, find roommates, post information to neighborhood and community pages, and view information posted by others. The Student Neighbor Network may be accessed only by Registered Students. The Student Neighbor Network is made up of pages. Each page is a group of registered student users that can interact together on the page, by posting comments and photos, classifieds, and other information. Registered student users can view each other's profiles and contact each other through the pages. Pages created in the Student Neighbor Network may contain text descriptions, photos, comment boxes, links, and contact information. Users are Registered Students that can view and post to pages. Users have a profile that can be viewed by other users.
- **Account**: Landlords, Star Tenants, Tenants, and Registered Students, must create an account to use the website. An account shall contain information pertinent to each user type: Students: name, phone number, SFSU email address, user profile; Star Tenants: name, phone number, email address; Tenants: name, phone number, email address, user profile; Landlords: name, property management or real estate company, office phone number, number of listings created. An account can either be verified or unverified. Any account, when created, shall be considered unverified. We plan to verify accounts based on their university-provided email address. Once a user requests their account be activated, they shall then be sent a secure and unique link to click at the email they provided. Once that link has been clicked, their account shall be considered verified. An unverified account shall not have access to post or apply for listings, or access the social network portion of the site. Furthermore, if a user opts to deactivate their account, their account shall be considered disabled. When an account is disabled, the end user is given a secret key with which they can use to reactivate their account if they choose. After 1 week, any disabled account shall be permanently deleted.
- **Profile**: A profile is a special view of an account's attributes. For example, a "profile" shall consist of a verified user's name, major, interests, bio, etc. There shall be a special page that shall be set up for all users that acts as a way to view their 'profile'.
- **Listing**: the entity that represents a property for rent. Listings contain information on the domicile (location, property amenities, number of bedrooms, number of bathrooms, price, etc.), information about the landlord, current Star Tenant and Tenants (when applicable), and embedded calendar for open-house sign-up. Individual Registered Students shall also see their roommate compatibility score with the current Tenants (if applicable) on the listing. Listings are posted by Landlords or Star Tenants and are available to view by all users. Listings may be moderated by the administrator.

## 2 - FUNCTIONAL REQUIREMENTS

**KEY:** Reference #

Ex. PL1.2

Post Listing - PL

Item.SubItem - 1.2

**Priority 1. Post listings.** Reference # PLX.X.

**P1.** PL1.0 - Landlords and tenants shall be able to create listings with information about their domicile:

- P1.** PL1.1 - Price
- P1.** PL1.2 - Square footage
- P1.** PL1.3 - Type of domicile
- P1.** PL1.4 - Number of occupants
- P1.** PL1.5 - Pet policy
- P1.** PL1.6 - Amenities
- P1.** PL1.7 - Disability accessibility
- P1.** PL1.8 - Pictures
- P2.** PL1.9 - Length of commitment
- P2.** PL1.10 - Utilities included
- P2.** PL1.11 - Parking
- P2.** PL1.12 - Furnished or unfurnished
- P2.** PL1.13 - Individual or group leasing
- P2.** PL1.14 - Security
- P3.** PL1.15 - Move-in specials
- P3.** PL1.16 - Lease renewal policy
- P3.** PL1.17 - Subletting policy
- P3.** PL1.18 - Number of each type of room
- P3.** PL1.19 - Renovation status
- P3.** PL1.20 - Noise policy
- P3.** PL1.21 - Age of building

**P1.** PL2.0 The site administrator shall approve all listings prior to posting.

**Relevant Stories**

*L1, L2, L3, A1*

**Priority 1. Search and Filter.** Reference # SFX.X.

**P1.** SF1.0 - Registered and unregistered users can search available domiciles using filters for:

- P1.** SF1.1 - Size and type of domicile (private bedroom in an apartment or house, shared bedroom, private apartment)
- P1.** SF1.2 - Price range
- P1.** SF1.3 - Roommate compatibility (for Registered Users only)

- P1.** SF1.4 - Amenities
- P1.** SF1.5 - Utilities included
- P1.** SF1.6 - Disability accessibility
- P2.** SF1.7 - Number of occupants
- P2.** SF1.8 - Proximity to university
- P2.** SF1.9 - Pet friendliness
- P2.** SF1.10 - Furnished or unfurnished
- P2.** SF1.11 - Length of commitment
- P2.** SF1.12 - Individual or group leasing
- P3.** SF1.13 - Parking
- P3.** SF1.14 - Rent control policy
- P3.** SF1.15 - Provided security type
- P3.** SF1.16 - Rent payment methods
- P3.** SF1.17 - Noise level
- P3.** SF1.18 - Renters insurance requirements
- P3.** SF1.19 - Subletting policy
- P3.** SF1.20 - Age of building

**Relevant Stories:**

S1, S2, S3, S4

**Priority 1. Registration.** Reference # RGX.X.

**P1.** RG1.0 - University students shall be able to:

- P1.** RG1.1 - Register using their SFSU email
- P1.** RG1.2 - Create an account and student profile
- P1.** RG1.3 - Use roommate compatibility score feature
- P1.** RG1.4 - Access student neighbor network and associated pages
- P2.** RG1.5 - Contact landlords, tenants, and other registered users through the site

**P1.** RG2.0 - Landlords and Star Tenants shall be able to:

- P1.** RG2.1 - Register with their personal or business email
- P1.** RG2.2 - Create an account
- P1.** RG2.3 - Post listings
- P2.** RG2.4 - Access calendar features
- P2.** RG2.5 - Track their listings

**P2.** RG3.0 - Star Tenants, Tenants and Landlords shall be verified by the administrator by:

**P1.** RG3.1 - Secure and unique link that shall need to be clicked within two hours.

**P2.** RG4.0 - Anyone who has an account (verified or unverified) shall be able to:

- P1.** RG4.1 - Delete their account.
- P2.** RG4.2 - Recover said deleted account if they act within 1 week of opting to disable it.

**Relevant Stories:**

S1, S2, S3, S4, L1, L2, L3

**Priority 1. Roommate Compatibility Score.** Reference # RCX.X.

Several applications currently on the market (roomiapp.com, roomiematch.com, spareroom.com, roomzoom.com, badi.com, and more) match roommates according to personality and lifestyle. We propose incorporating roommate compatibility into our service to help students live with less stress off campus. To better understand what is important in a roommate to SFSU students specifically, we will additionally conduct an interview of our peers. Pegasus shall provide the following to registered students, tenants, and star tenants:

- P1.** RC1.0 - Match registered students who are looking for a shared domicile with tenants and star tenants currently residing in a domicile
- P1.** RC2.0 - Match registered students with other registered students for the purpose of moving into a domicile together
- P1.** RC3.0 - Allow registered students to search listings by roommate compatibility
- P1.** RC4.0 - Display roommate compatibility score (compatibility of current tenants and registered student who is searching the site) on a listing
- P1.** RC5.0 - Roommate compatibility for pegasus shall be evaluated using structured data input by users during registration. Data shall be connected with the registered user profile and shall include content such as:
  - P1.** RC5.1 - Lifestyle priorities (family, studying, hobbies, exercise, going out)
  - P1.** RC5.2 - Morning person or night owl
  - P1.** RC5.3 - Personality traits
  - P1.** RC5.4 - Comfort level with roommate guests
  - P1.** RC5.5 - Preferred frequency of gatherings in the home
  - P1.** RC5.6 - Cleanliness
  - P1.** RC5.7 - Special skills (like cooking, languages you speak, home maintenance skills)
  - P1.** RC5.8 - Major
  - P1.** RC5.9 - Interests and hobbies
  - P2.** RC5.10 - Self described introversion / extroversion
  - P2.** RC5.11 - Relationship status
- P1.** RC6.0 - Structured data input by users may be evaluated for simple similarity
- P2.** RC7.0 - A clustering algorithm and ranking algorithm shall be used to evaluate compatibility
- P2.** RC8.0 - Roommate compatibility shall be evaluated using data mining of currently existing social media profiles (if possible)

**Relevant Stories:**

S1, S2, S3, S4

**Priority 1. Student Neighbor Network.** Reference # SNX.X.

- P1.** SN1.0 - Allow registered users to post information about their neighborhoods in neighborhood pages.

- P1.** SN1.1 - Allow registered users to post information about neighborhood events.
  - P1.** SN1.2 - Allow registered users to post information about nearby services.
  - P2.** SN1.3 - Allow registered users to post information about entertainment.
  - P2.** SN1.4 - Allow registered users to post other information about the neighborhood, such as security.
- P1.** SN2.0 - Allow registered users to view available pages for the neighborhood of a domicile.

### **P2.** SN3.0 - Classified Pages

- P1.** SN3.1 - As a way to foster community spirit, pages allow users to share information about moving services, or buying / selling furniture and household items. Very similar to Craigslist, but more communal. This shall be a page within the student neighbor network, and should allow filtering for services, items, time of posting, etc.

#### **Relevant Stories:**

S1, S2, S3, S4

## **Priority 1. Administrative Tasks.** Reference # ATx.x

Administrative tasks consist of:

- P1.** AT1.0 - Reviewing and approving listings.
    - P1.** AT1.1 - Checking for inappropriate content including explicit photos or language
    - P2.** AT1.2 - Checking for discriminatory content
    - P2.** AT1.3 - Verifying authenticity of listings (for example by making sure that the photos associated with the listing aren't photos available elsewhere online)
  - P1.** AT2.0 - Managing registered user accounts and profiles
  - P1.** AT3.0 - Maintaining integrity of listings and removing them if shown to be fraudulent
  - P2.** AT4.0 - Moderating posts in neighborhood pages and classifieds pages.
- Inappropriate content shall be removed as detected.

#### **Relevant Stories:**

A1

## **Priority 1. Messaging**

Registered Students shall be able to message landlords and star tenants.

- P1.** M1.0. Registered students shall be able to contact landlords and star tenants through the application without giving out their email addresses
  - P1.M1.1** users should have profiles pic
  - P1.M1.2** users should have their name displayed by their profile
  - P2.M1.3** chat should shows if the user is online or not
- P2.M2.0.** Messages should show the message is been read or not
  - P1.M2.1** users shall be able to take photo and send it.

#### **Relevant Stories:**

S1, S2, S3, S4, L1, L2, L3

**Priority 2. Watch List.** Reference # WLx.x

Registered users can create a “watch list” for residences that meet their search criteria, i.e. a room of size at least 10’x10’ and of price less than \$1,000 per month. When residences appear that match these pre-set criteria, the users shall get notified through their choice of notification method, e.g.. Telephone, email, through the app, etc.

- P1.** WL1.0 - Allow registered users to save searches that match certain criteria.
- P2.** WL2.0 - Allow registered users to choose whether or not to be notified if any new residences appear on market that match that search.
- P2.** WL3.0 - If any residences do appear, and the registered user opted to be notified, they should be notified.
- P2.** WL3.1 - Allow registered users to select the method of notification (via email, app, etc.).

**Relevant Stories:**

S1, S2, S3, S4

**Priority 2. Geographic overlay.** Reference # GOx.x

Verified and unverified users can:

- P1.** GO1.0 - Visualize the available apartments on a geographic overlay
- P2.** GO2.0 - Users can additionally use the geographic overlay to:
  - P1.** GO2.1 - Map their route from the apartment to campus
  - P2.** GO2.2 - Map from the apartment to another selected location (such as their work address, or child care)
  - P3.** GO2.3 - Place markers for points of interest
  - P3.** GO2.4 - See neighborhood boundaries and crime statistics
- P2.** GO3.0 - Users can also visualize nearby landmarks, like parks, grocery stores, or transit stops.

**Relevant Stories:**

S1, S2, S3, S4

**Priority 2. Calendar.** Reference #CLx.x

A calendar can be embedded in listings and on registered student accounts. The calendar shall:

- P1.** CL1.0 - Allow landlords and star tenants to post their open house hours on their listing.
- P1.** CL2.0 - Allow registered students to sign up for an open house with one click.
- P1.** CL3.0 - Display the calendar on the profiles of landlords, star tenants, and registered users.
  - P1.** CL3.1 - For landlords and star tenants, display on the calendar the dates and times for applicant appointments and open houses.
  - P2.** CL3.2 - For registered students, display a calendar with upcoming open house appointments.

**P1.** CL4.0 - Once a registered student signs up for a listing's open house hours, their profile shall be forwarded to the landlord/star tenant who posted the listing.

**P1.** CL4.1 - Landlords and star tenants shall be able to view a list of links to user profiles that have signed up for an appointment.

**P2.** CL4.2 - A registered user shall be able to cancel their appointment, which shall also remove the user's profile from the list of links to profiles.

**P2.** CL6.0 - Send notifications to landlords, tenants, and registered students to remind them of upcoming appointments.

**Relevant Stories**

*L1, L2, L3, S1, S2, S3, S4*

**Priority 3. Application.** Reference #Ax.x

After registering for an open house, registered users shall be able to fill out an additional form, containing additional information required by the landlord or star tenant who posted the listing:

**P1.** A1.0 First / Last Name

**P1.** A1..0 Contact information. (Whether it be through the app, email, telephone, etc.)

**P2.** A1.0 Any questions or concerns regarding the residence.

**Relevant Stories:**

*L1, L2, L3*

**Priority 3. Props, Likes, and Shares.** Reference #PLSx.x

Registered users shall be able to share listings and review listings and neighborhood pages.

**P1.** PLS1.0 Allow registered users to share listings.

**P1.** PLS1.1 Share with a link.

**P3.** PLS1.2 Share through social media.

**P1.** PLS2.0 Allow registered users to like listings.

**P1.** PLS2.1 Registered users shall be able to like a listing.

**P1.** PLS2.2 Registered and unregistered users shall be able to view the number of likes on a listing.

**P2.** PLS3.0 Allow registered users to post reviews about listings and neighborhoods.

**P1.** PLS3.1 Registered users shall be able to comment on listings and neighborhood pages.

**P1.** PLS3.2 Registered and unregistered users shall be able to view all comments on a listing or neighborhood page.

**P3.** PLS4.0 Registered users can endorse aspects of another registered user's profile via giving props.

**Relevant Stories:**

*S1, S2, S3, S4*

**Priority 3. Variety of photos.** Reference #VPx.x

Landlords and tenants are required to post photos of their listings that accurately reflect the living conditions.

**P1.** VP1.0 They must take photos of main rooms suchs as

- P1.** VP1.1 Living room
  - P1.** VP1.2 Family room
  - P1.** VP1.3 Dining room
  - P1.** VP1.4 Bathroom
  - P2.** VP1.5 Property from the outside
  - P2.** VP1.6 Master bedroom
- P1.** VP2.0 Photos of the listed domicile

**Relevant Stories:**

S1, S2, S3

**Priority 3. Find roommates**

Registered students shall be able to seek out other registered students who are looking to co-lease a domicile.

- P1.FR1.0.** Allow users to search potential roommates based on category such as majors, courses, hobbies, roommate compatibility score, etc.
  - P1.FR1.1** Allow users to see all potential roommates list on the listings.
  - P2.FR1.2** Organize the listings with highest matching rate (best match)
  - P3.FR1.3** Display the matching categories on the list (in common).
- P2.FR2.0.** Allow users to contact roommates after matching the categories.
  - P1.FR2.1** Users shall see the contact information include photos, contact lists.
  - P2.FR2.2** Allow users to message the roommates through the site.
- P3.FR3.0.** Allow users to see potential roommates profile.
- P4.FR4.0.** Allow users to make an appointments based on the schedule availability.

**Relevant Stories**

S1, S2

**Priority 3. Notifications.**

The application shall have the ability to send notifications to registered users when a desired domicile becomes available, and as a reminder for upcoming open houses.

- P1.N1.0.** Allow landlords and tenants to send message to users who are on the watchlist.
- P2.N2.0.** Notify registered students of the available domicile.
- P3.N3.0** Engage users to receive information without knowing the contact detail
  - P1.N3.1** Allow users to have option for notify information through email, phone, social network, etc.
  - P2.N3.2** Provide prompt and accurate content delivery.
- P3.N3.0.** Allow users to unsubscribe when they do not want to receive information.
- P4.N4.0.** Work as appointment/open-house reminder.

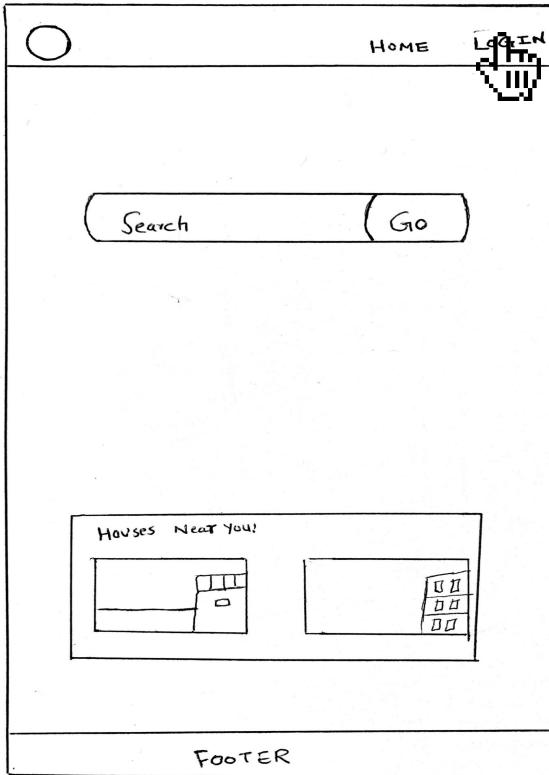
**Relevant Stories**

S1, S2, S3, S4

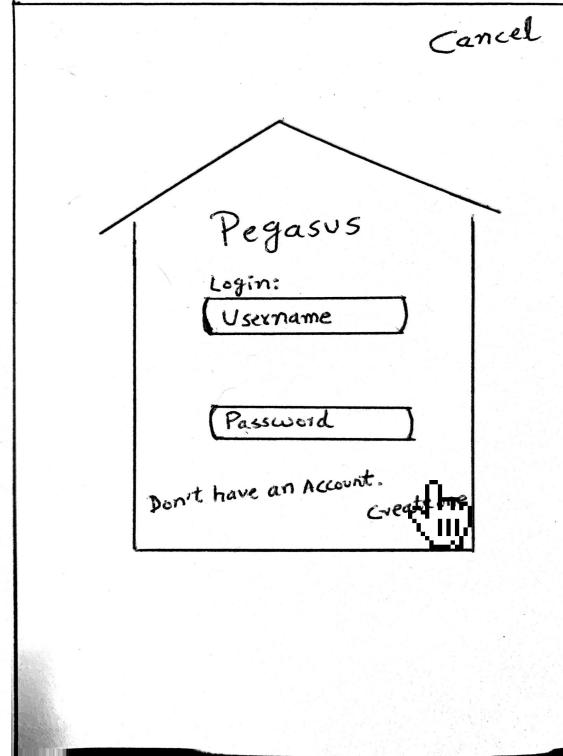
### 3 - UI MOCKUPS AND STORYBOARDS

*Hand drawn storyboards and UI Mockups in Adobe XD follow on the next 23 pages.*

# Home Page, Login page, Signup Page



Screen 1: Home Page



Screen 2: Login Page

A hand-drawn sketch of a sign-up form. At the top, there are 'HOME' and 'ACCOUNT' buttons. Below the buttons is a user profile icon. To the right of the profile icon are tabs for 'THE BASICS', 'THE HOBBIES', 'SPORTS', and 'FOOD', with 'THE BASICS' being the active tab. The form contains fields for 'Name' (with a placeholder 'Names \_\_\_\_\_'), 'Email' (with a placeholder 'Emails \_\_\_\_\_'), 'Password' (with a placeholder 'Password \_\_\_\_\_'), and 'Confirm Password' (with a placeholder 'Confirm Password \_\_\_\_\_'). To the right of these fields are three numbered sections (1, 2, 3) each with a list of four options. At the bottom right are 'Save', 'Next', and 'Cancel' buttons.

Screen 3: Sign-up page

# Alessandro StoryBoard

①

The Basics

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_

Scanned with  
CamScanner

Screen 1: User SignUp page

HOME ACCOUNT

THE BASICS

How clean are you?

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

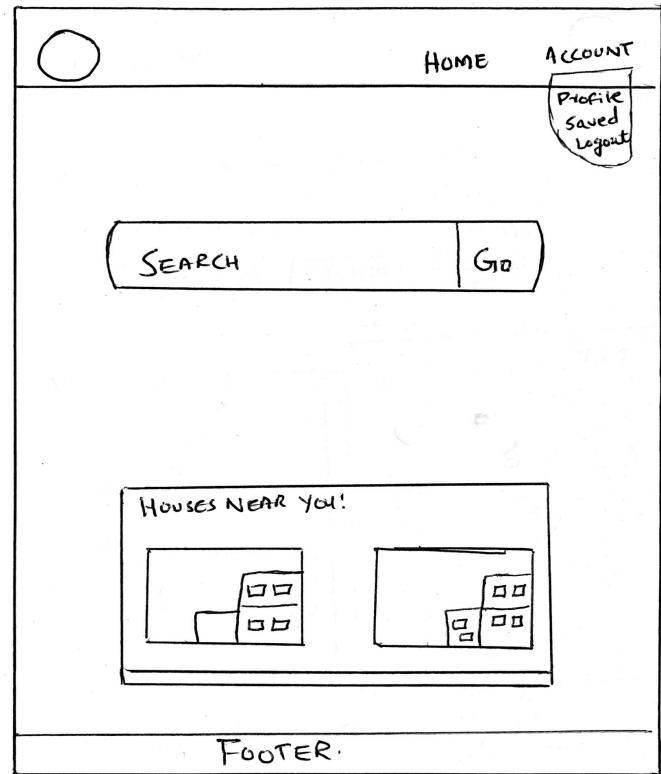
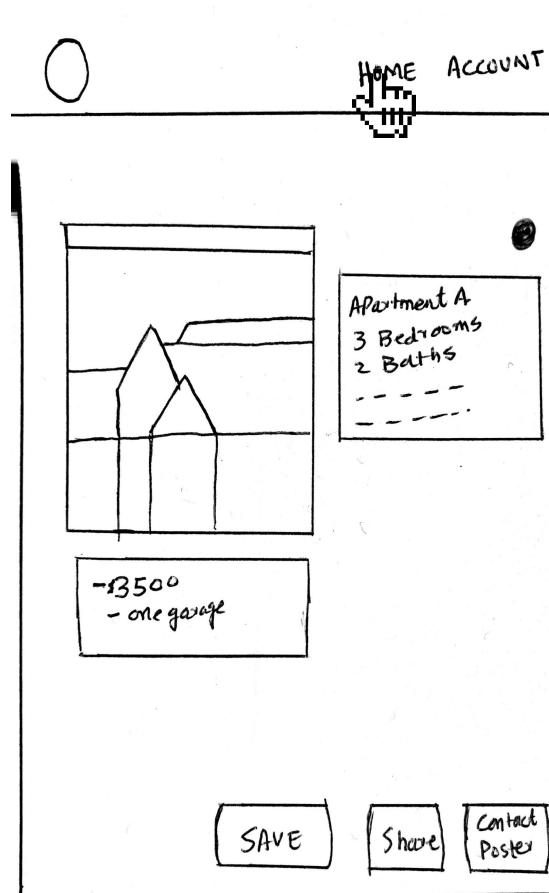
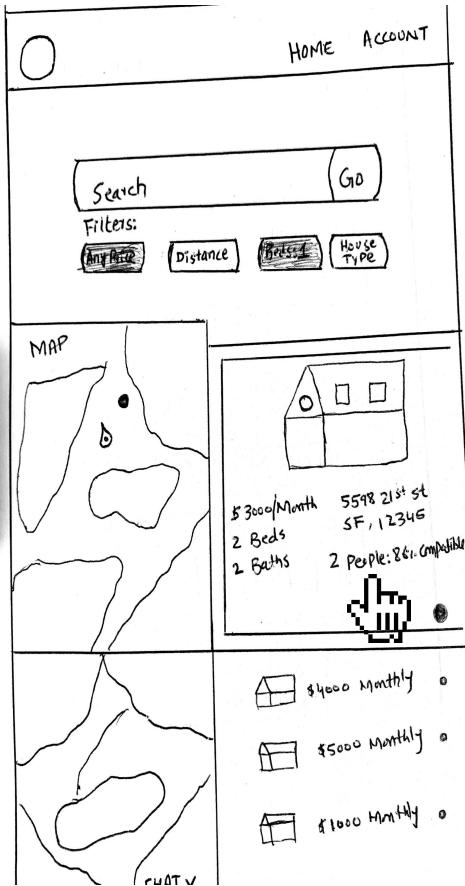
THE HOBBIES SPORTS FOOD

Name: \_\_\_\_\_

Email: \_\_\_\_\_

Save Next Cancel

Screen 2: User profile Survey Page

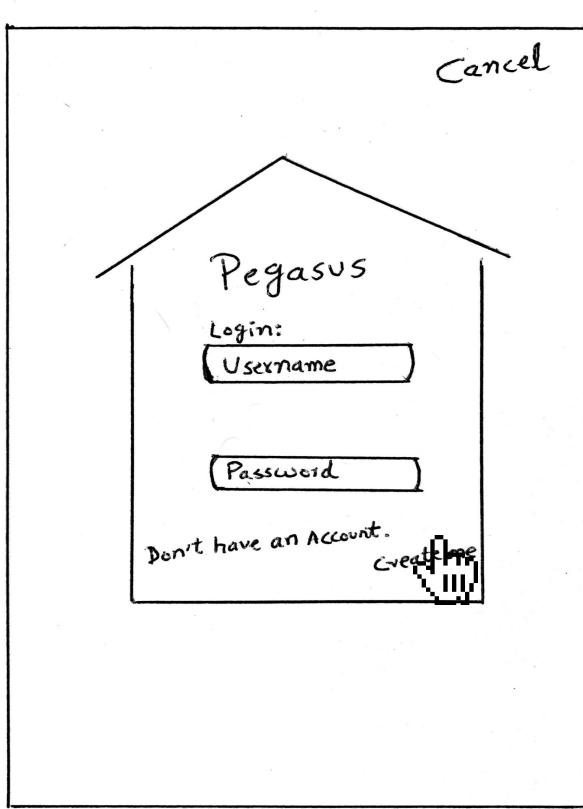


Screen 3: Search Apartment Page

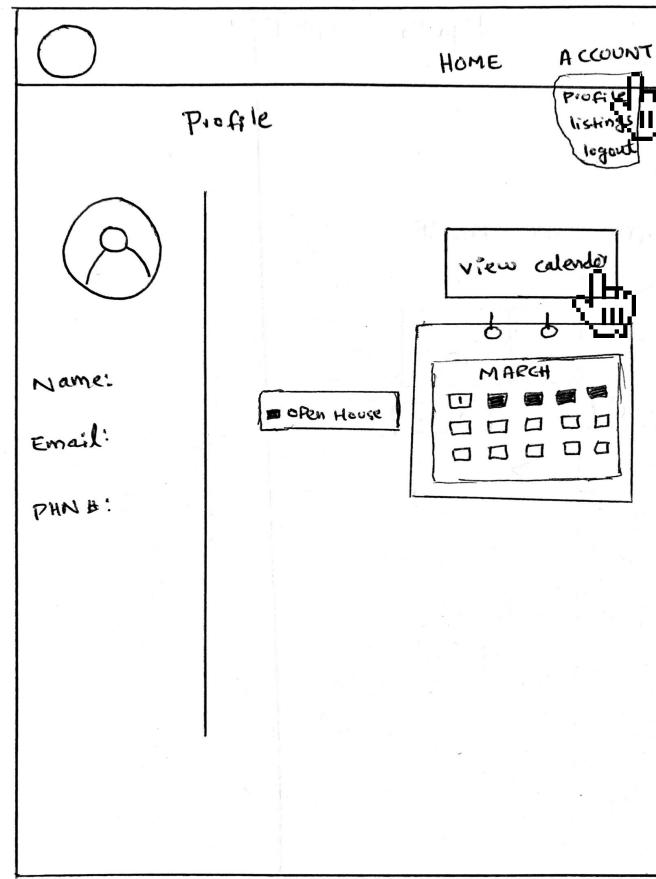
Screen 4: Apartment Description

Screen 5: Home page

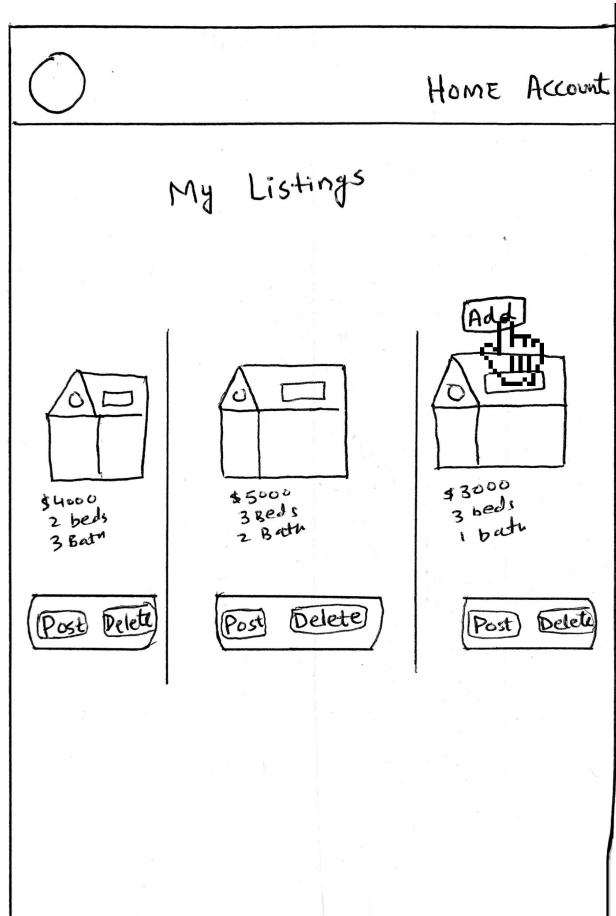
# Marybeth Storyboard



Screen 1: User Sign In page



Screen 2: User profile page



Screen 3: Listing property page

HOME ACCOUNT

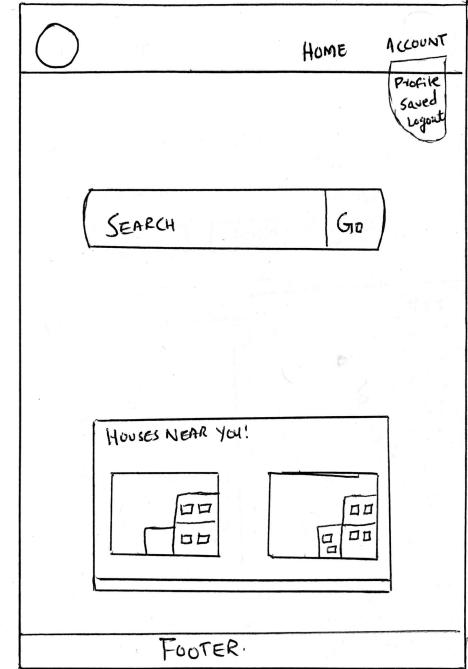
Adding New House

Two large empty rectangular input fields.

Description .....  
# of rooms  
# of Bathrooms  
# of Garage

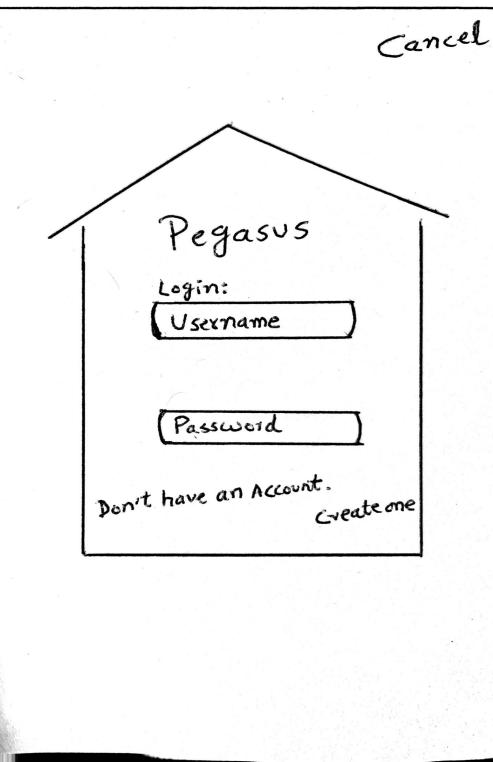
[Submit] [Cancel]

Screen 4: Adding new list



Screen 5: Home page

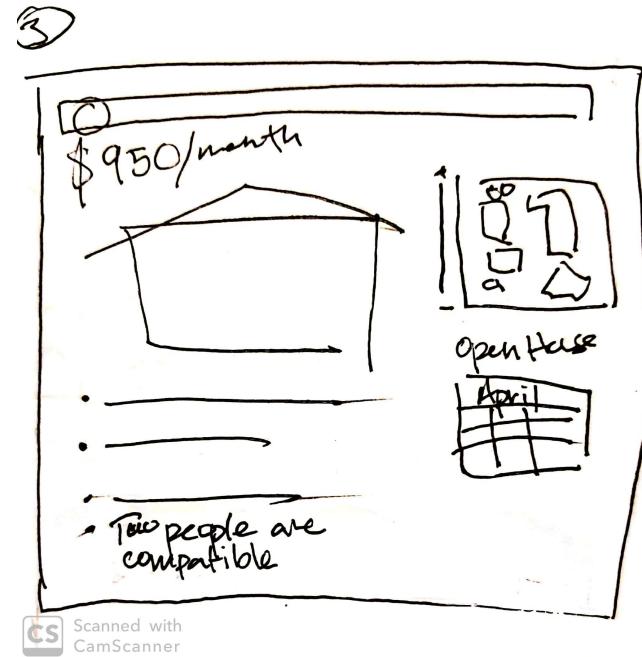
# Andy Clarkson StoryBoard



1. Login screen.



2. After login, the browse listings view displays listings.



3. Clicking on a listing displays the listing Andy is interested in.

## Velvet Crowe - Admin - StoryBoard

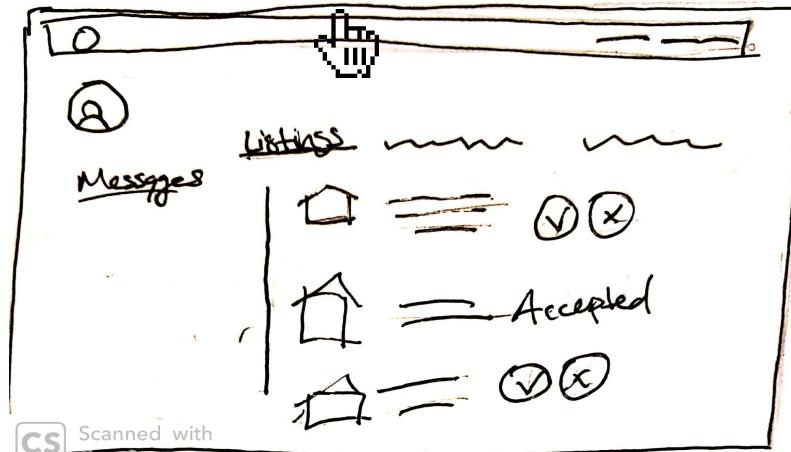
①



Scanned with  
CamScanner

1. The listings that need to be approved or rejected

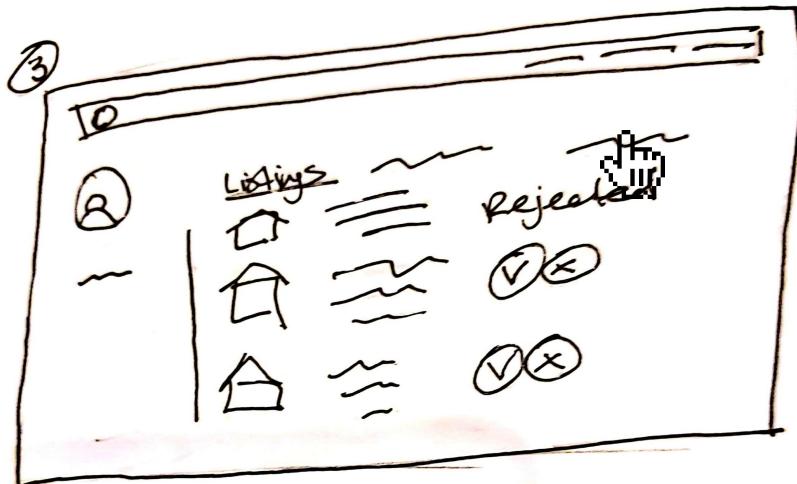
②



Scanned with  
CamScanner

2. Clicking on listings shows the approved listings view.

## Velvet Crowe - Admin - StoryBoard



Scanned with  
CamScanner

Line oddas for Rejection

3. A display of the rejected listings

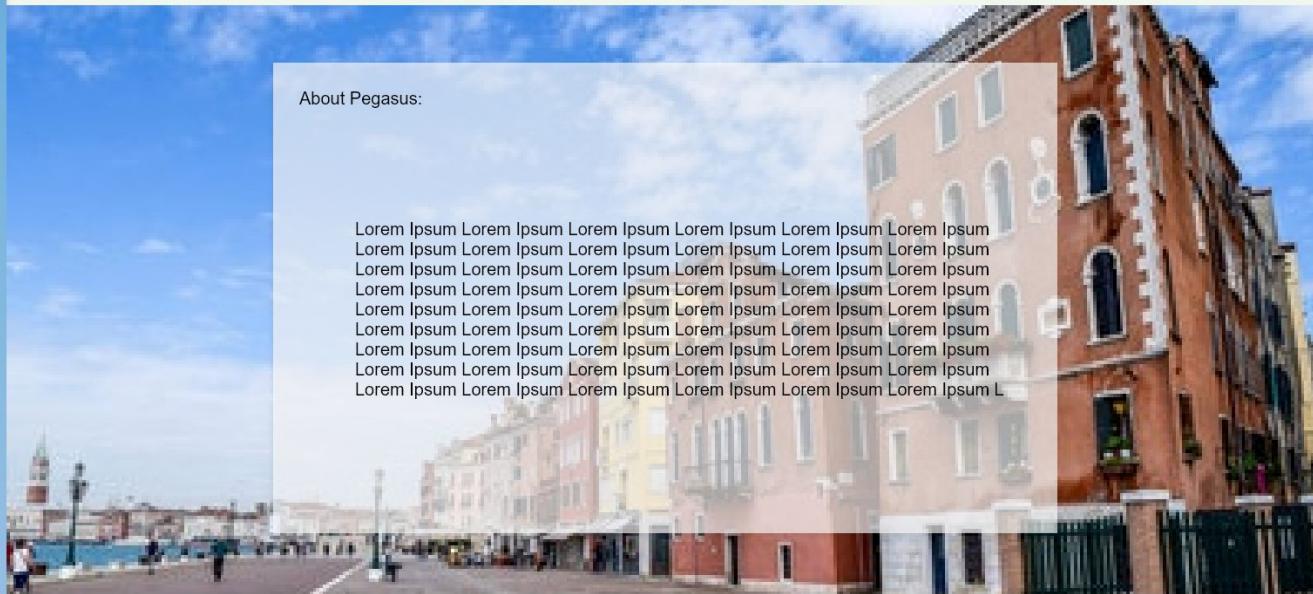


Scanned with  
CamScanner

4. A display of comments removed from a listing or page in the activity log.



Search\_



About Pegasus:

Lore Ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum L

Screen 1: Home Page for someone who does not have an account.



Cancel



Screen 2: Alessandro wants to create an account so he is directed to the login page, and from the he "...clicks don't have an account?"



The Basics

The Hobbies

The Sports

The Food



Email

School E-mail

Password

Confirm Password

**Save**

1. How clean are you?

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

2. Are you a morning person?

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

3. Are you a night owl?

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

4. How clean are you?

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

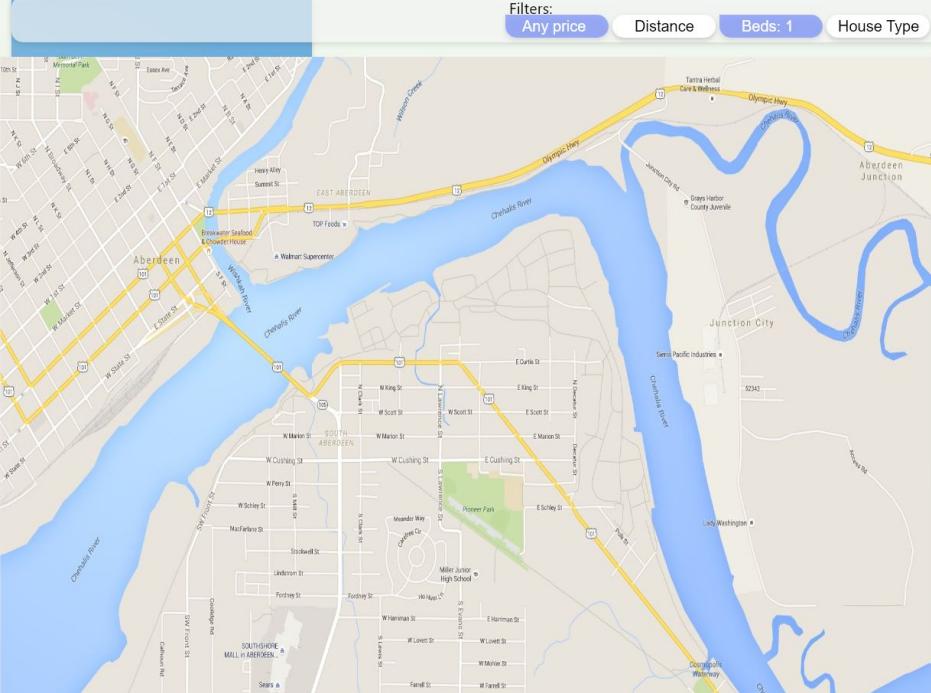
5. Are you a morning person?

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

6. Are you a night owl?

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Save**



This screenshot shows a search results page for property listings. The top navigation bar includes 'Home' and 'Account' links, and a dropdown menu for 'Profile', 'Saved', and 'Logout'. A large circular cursor points to the 'Logout' button. The main content area displays four property cards:

- \$3000/Month**  
2 Beds  
2 Bath Rooms  
5598 21st Street, SF, 12345  
2 People: 86% Compatible
- \$4000/Month**  
4 Beds 3 Baths  
12 Monkey Street, SF, 12345
- \$2000/Month**  
3 Beds 1 Baths  
19 Monkey Street, SF, 12345
- \$1000/Month**  
2 Beds 1 Baths  
14 Monkey Street, SF, 12345

The filters applied are 'Any price', 'Distance', 'Beds: 1', and 'House Type'.

Screen 4: This page is dedicated to browse listings. The circular colors correlates to the chat room screen (chat room screen as not been made yet). The highlighted filters are the filters that are in use.



Search \_

Save

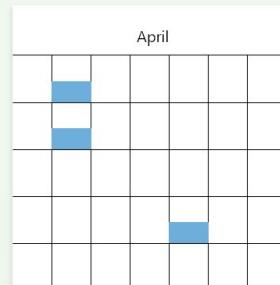
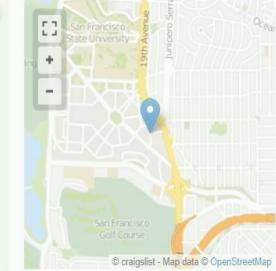
Share

Contact

Cancel



Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum  
Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum Lore ipsum L



Screen 5: This screen shows the listing that Alessandro is interested in. The calendar shows open houses and you can reserve a spot. The color again corresponds to the chat room.

Alessandro

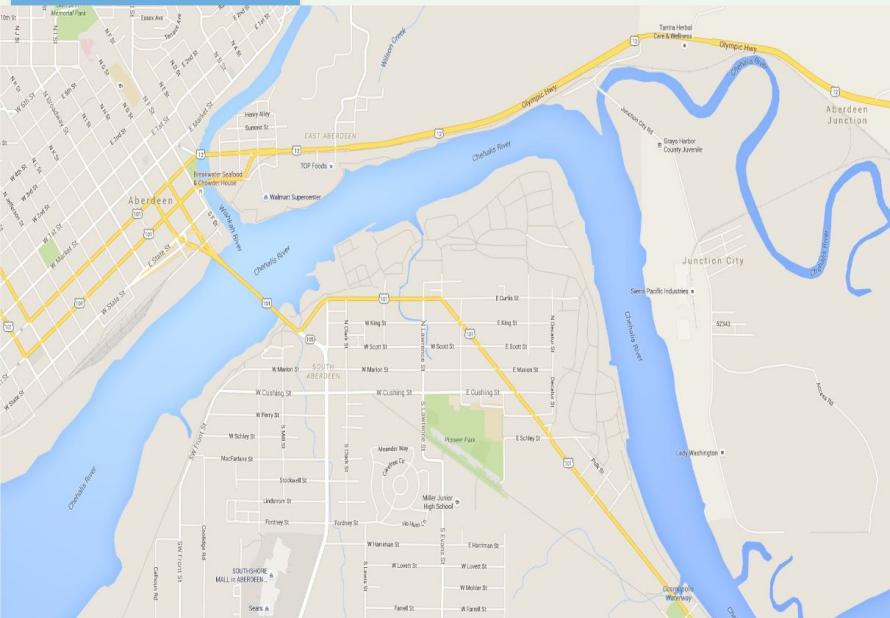


Cancel



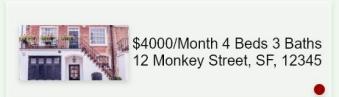
Screen 1: Login Page

Filters:  
[Any price](#) [Distance](#) [Beds: 1](#) [House Type](#)

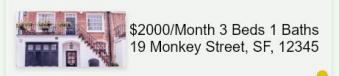




\$3000/Month  
2 Beds  
2 Bath Rooms



\$4000/Month 4 Beds 3 Baths  
12 Monkey Street, SF, 12345



\$2000/Month 3 Beds 1 Baths  
19 Monkey Street, SF, 12345



\$1000/Month 2 Beds 1 Baths  
14 Monkey Street, SF, 12345

## Screen 2: Browsing Listings



Search\_

Save Share Contact Cancel



Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum Lorem Ipsum  
Lorem Ipsum Lorem Ipsum L



April						

## Screen 3: Interested Listings

The screenshot shows a user profile interface. On the left, there's a sidebar with a circular user icon and sections for Name (Marybeth L.), School E-mail (bogus@sfsu.com), Email (bogus@aol.com), Password (\*\*\*\*\*), and Change Password? (with a Save button). The main area has tabs for Profile, My Listings, and Calendar. A hand cursor is pointing at the Calendar tab. Below the tabs is a calendar grid for April, with three horizontal bars highlighted in blue across the first, second, and fourth weeks.

Screen 1: This screen shows the profile view of a poster, and this specific view shows the calendar.



Profile

My Listings    Calendar

Name  
Marybeth L.

School E-mail  
bogus@sfsu.com

Email  
bogus@aol.com

Change E-mail?  
Password  
\*\*\*\*\*  
Change Password?  
Save

Three property listings:

- \$3000/Month  
2 Beds   2 People: 86% Compatible  
2 Bath Rooms •
- \$3000/Month  
2 Beds   2 People: 86% Compatible  
2 Bath Rooms •
- \$3000/Month  
2 Beds   2 People: 86% Compatible  
2 Bath Rooms •

Post   Delete   Post   Delete   Post   Delete

Add Place +

Screen 2: This screen shows all of Marybeth's listings she posted



Name

Marybeth L.

School E-mail

bogus@sfsu.com

Email

bogus@aol.com

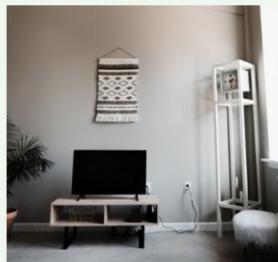
Change E-mail?

Password

\*\*\*\*\*

Change Password?

Save



## Description

Number of Rooms: 3

Number of bath rooms: 2

Address: 555 California st San Fran ca 94123

Price : \$3000

Pets: cats only

Add

Cancel



April				

Screen 3: This screen shows when Marybeth wants to add a new listings.



The screenshot shows a user interface for managing listings. On the left, there's a sidebar with a user profile icon and the word "Messages". The main area has three tabs: "Listings" (which is active), "Community", and "Activity Log". Below the tabs, there are two identical listing cards. Each card features a thumbnail image of a brick building with a balcony, followed by listing details: "\$3000/Month", "2 Beds", "2 Bath Rooms" on the left; and "5598 21st Street, SF, 12345" and "2 People: 86% Compatible" on the right. To the right of the second listing card are two circular buttons: a green "Yes" button and a red "No" button. A hand cursor is hovering over the red "No" button. Below the buttons, the word "Accepted" is displayed in green text.

Screen 1: This screen shows a screen of the admin. This screen specifically belongs to Velvet and here she is approving a listing



The screenshot shows a web-based real estate administration platform. On the left, there's a sidebar with a user icon and a "Messages" button. The main area has a header with tabs: "Listings" (highlighted in blue), "Community", and "Activity Log". A hand cursor is positioned over the "Activity Log" tab. Below the header, there are two identical listing cards. Each card features a thumbnail image of a brick building with a staircase and flower boxes, followed by listing details: "\$3000/Month", "5598 21st Street, SF, 12345", "2 Beds", "2 People: 86% Compatible", and a small red circular icon. To the right of each card are two buttons: a green "Yes" button and a red "No" button. In the bottom right corner of the main area, the word "Rejected" is displayed in red text.

Screen 2: This screen shows a screen of the admin. This screen specifically belongs to Velvet and here she is rejecting a listing



Listings Community Activity Log

Comments Removed

user1:  
Lordk ladjkfk Invi ka Lordk ladjkfk Invi kav  
...

user2:  
Lordk ladjkfk Invi ka Lordk ladjkfk Invi kav  
...

user3:  
Lordk ladjkfk Invi ka Lordk ladjkfk Invi kav  
...



Messages

Chat Listing

Screen 3: This screen shows the comments that Velvet removed from the chat rooms.



The screenshot shows a user interface for a platform called Velvet Crowe. At the top, there is a dark blue header bar with a user profile icon, a "Home" link, and an "Account" link. Below the header, there is a navigation bar with three tabs: "Listings", "Community", and "Activity Log", with "Activity Log" being the active tab. On the left side, there is a sidebar with a circular profile picture placeholder and a "Messages" button. The main content area contains a list of comments from users "user1", "user2", and "user3". Each comment is followed by a series of dots indicating it has been removed. The comments themselves are mostly gibberish. At the bottom of the list, there is a downward arrow and a hand cursor icon pointing towards it.

Comments Removed

user1:  
Lordk ladjkfk Invi ka Lordk ladjkfk Invi kav

user2:  
Lordk ladjkfk Invi ka Lordk ladjkfk Invi kav

user3:  
Reason: ka;fd kjkasd ;fkljas;lkdf jksajf  
Date: 13/34/0000

Screen 4: This screen shows the comments that Velvet removed from the chat rooms. It is the same as screen 3 except if you slide any comment to the left it will show description when Velvet removed the comment.

## 4 - HIGH LEVEL ARCHITECTURE, DATABASE ORGANIZATION

### **Overview**

- For our database, we're using the following setup. We have one database for testing, pegasus-dev, and one for production, pegasus-prd. The schema, triggers, etc. in pegasus-prd should mirror that of pegasus-dev. The idea for the separation is to keep one copy of the database always in working order. We also have scheduled backups of these databases occurring three times a day, in case we need to restore after an accidental change, deletion, etc.

### **Media**

- Media storage will be done on the filesystem itself, while the database will contain the paths to these files. From what we were able to research, BLOBS seem to work better for a large amount of smaller files, but when the file size is increased, having the files stored on the filesystem is more efficient. We have not at this moment considered any media type other than static images.

### **Table Schema**

Table	Table Purpose
active_listings	Hold current listings for residences.
admins	Hold admins to the site.
auth_group	Django built-in.
auth_group_permissions	Django built-in.
auth_user	Django built-in.
auth_user_groups	Django built-in.
auth_user_user_permissions	Django built-in.
disabled_users	Hold disabled users.
django_admin_log	Django built-in.
django_content_type	Django built-in.

django_migrations	Django built-in.
django_session	Django built-in.
landlords	Hold landlords.
residences	Hold domicile information.
star_tenants	Hold star_tenants.
students	Hold students.
unverified_users	Hold users who have not verified their registration

Table	Column Name	Column Type
active_listings	residence_id	integer
active_listings	tenants	character varying (100)
active_listings	owner	character varying (15)
active_listings	price	double precision
admins	username	character varying (15)
admins	password	character varying (50)
admins	date_of_birth	date
admins	physical_address	character varying (100)
admins	email	character varying (100)
admins	profile_picture	character varying (100)
admins	creation_time	date
admins	first_name	character varying (20)
admins	last_name	character varying (20)
students	username	character varying (15)

students	password	character varying (50)
students	date_of_birth	date
students	physical_address	character varying (100)
students	email	character varying (100)
students	profile_picture	character varying (100)
students	creation_time	date
students	first_name	character varying (20)
students	last_name	character varying (20)
star tenants	username	character varying (15)
star tenants	password	character varying (50)
star tenants	date_of_birth	date
star tenants	physical_address	character varying (100)
star tenants	email	character varying (100)
star tenants	profile_picture	character varying (100)
star tenants	creation_time	date
star tenants	first_name	character varying (20)
star tenants	last_name	character varying (20)
landlords	username	character varying (15)
landlords	password	character varying (50)
landlords	date_of_birth	date
landlords	physical_address	character varying (100)
landlords	email	character varying (100)
landlords	profile_picture	character varying (100)

landlords	creation_time	date
landlords	first_name	character varying (20)
landlords	last_name	character varying (20)
landlords	agency	character varying (50)
active_listings	residence_id	integer
active_listings	tenants	character varying (100)
active_listings	owner	character varying (15)
active_listings	price	double precision
active_listings	creation_time	date
expired_listings	residence_id	integer
expired_listings	tenants	character varying (100)
expired_listings	owner	character varying (15)
expired_listings	price	double precision
expired_listings	expiration_time	date
direct_messages	sender	character varying (15)
direct_messages	recipient	character varying (15)
direct_messages	message	character varying(500)
direct_messages	send_date	timestamp with timezone
groups_table	group_name	character varying (50)
groups_table	group_location	character varying(100)
groups_table	creation_time	timestamp with timezone
groups_table	group_picture	text
groups_table	group_owner	character varying (15)

group_members	group_name	character varying (50)
group_members	username	character varying (15)
group_posts	post_id	integer
group_posts	poster	character varying (15)
group_posts	post_date	date
group_comments	poster	character varying (15)
group_comments	comment	character varying (500)
group_comments	comment_date	date

## API

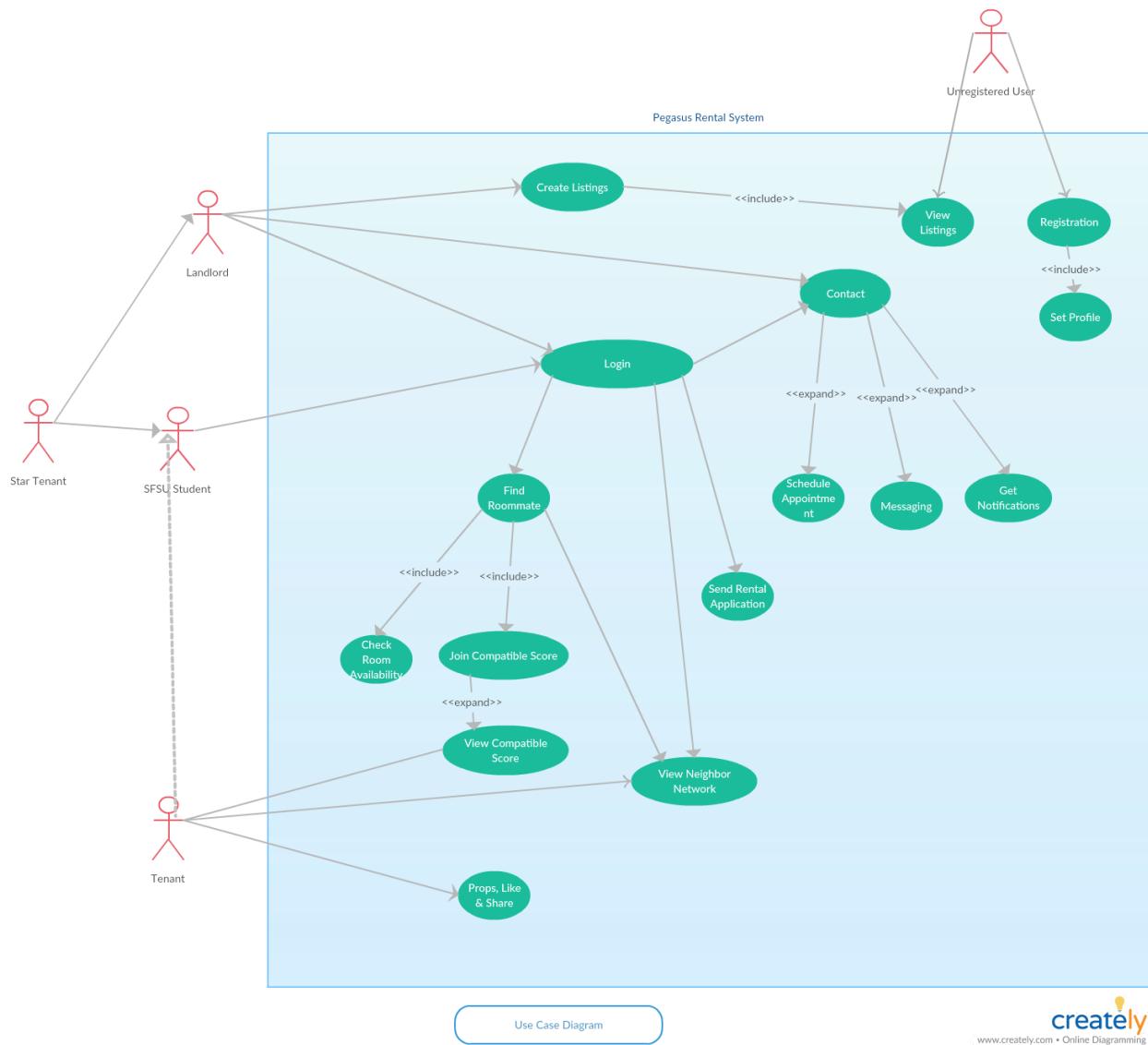
- CRUD operations will be handled by Django's API for models, and not through any direct SQL. As such, we have not planned for any additional algorithms to use, other than what comes packaged with Django. As far as the user query terms are concerned, the idea is that they will enter whatever filters that they want into the form, which will be cleansed by Django, and through Django's API, go to the database.

## Processes

- One process we have been discussing is the process of generating a “roommate compatibility score” with other tenants. This will be done via a mandatory survey at user registration. An example question might be “On the below scale, choose how introverted are you? 0: Very Introverted 1: Somewhat Introverted 2: Neutral (...). The responses to this survey can be stored within the database and calculated on the fly against a particular user when they are trying to determine their compatibility score with someone else. As a starting point, we propose an unsupervised learning algorithm, k-means-clustering, to cluster users into groups based on the similarity of their survey responses. If the groups are large, we could additionally use pairwise ranking to suggest most similar users.

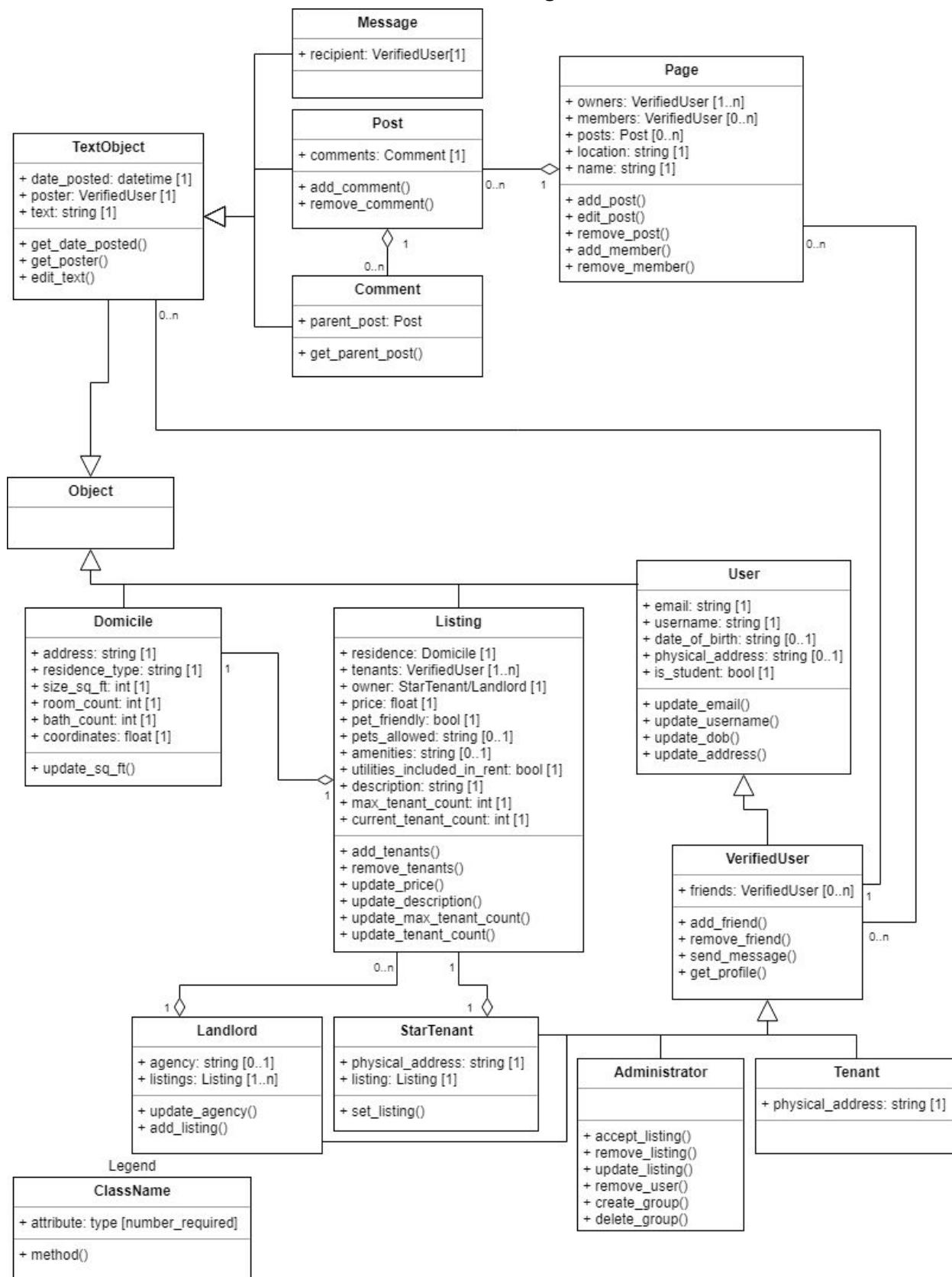
## 5 - HIGH LEVEL UML DIAGRAMS

**Use Case Diagram**

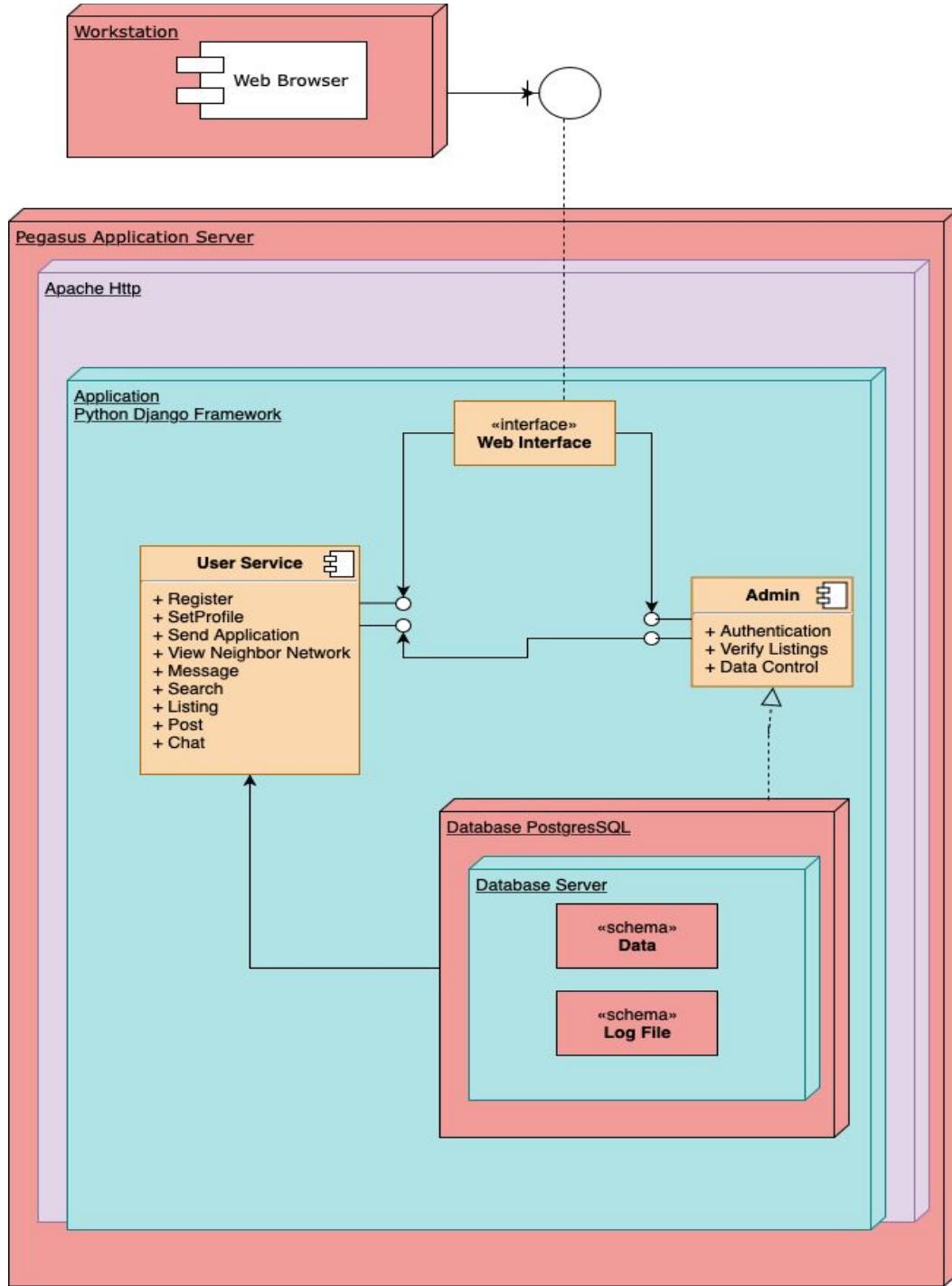


Use Case Diagram

## Class Diagram



## Component and Deployment Diagram



## 6 - ACTUAL KEY RISKS FOR OUR PROJECT AT THIS TIME

### **Skills Risks**

- Working with Django. Two team members have experience working with the Django framework, and the rest do not. There is a learning curve with Django that we will have to overcome. To address this, three of the team members have been assigned to study Django tutorials and help teach the others.
- Understanding the front-end frameworks. Two team members have experience working with React, and the others do not. The team members with experience in React do not have experience working with Django. Similar to our approach for learning Django, three team members have been assigned to learn how to connect React to Django and to teach the others.
- Comfort working with git. We have agreed upon how to use git to manage our project in GitHub. While some team members are already experts using git, those of us who are less comfortable will need to practice. The team leader will compile git documentation into an easily readable flowchart for team members to reference.

### **Scheduling Risks**

- There is not much time left in the semester, so incorporating all the features we want will be a challenge. Using Trello to break the deliverables into manageable tasks with regular due dates will help us stay on track.

### **Technical Risks**

- We are still not sure about all the APIs / libraries / packages we will need to use in order to achieve the functionality we want. The website is still in the concept stage. Going forward, each team member will be assigned tutorials to read through, so that we can be better prepared about the possibilities.
- How to calculate roommate compatibility score. We have discussed and decided upon the kind of data we want to collect and how we would like the score represented. We have even discussed machine learning algorithms we can use. However, the results of these may be difficult to interpret. We will not know if our roommate compatibility score has real world meaning.

### **Teamwork Risks**

- In our team, we have some team members that are experienced, and some team members that are less experienced (a common issue in any team of students). The challenge here is that it can be difficult to divide the work evenly. We will address this problem by assigning very specific tasks to each team member. The less experienced team members can ask the more experienced ones when they need help, but everyone will get a chance to contribute.

### **Legal / Content Risks**

- Licensing for the APIs we want to use. The APIs we would like to use, for example Google Maps or Google Calendar, may not be free under all circumstances. We need to research how to use these APIs for free, or with educational credits, in our application. Issues may arise with any other API we would like to use as the project progresses.

- HTTPS certificates. Currently, under AWS free tier, we are not able to get an Amazon signed identity certificate in order to run on HTTPS. To scale our application, we would like to use HTTPS. We may be able to get a signed certificate for free using educational credits of some kind, but it is not clear how. We will need to get in touch with Professor Song about how to obtain a signed certificate at no cost to us.
- Since there is a networking component to our application, we have to deal carefully with user data. Users will be able to make their profiles private and their data unavailable to view by others. Registered students should be able to opt in or out of the roommate compatibility score. Data for each user will be stored securely as well.

## 7 - PROJECT MANAGEMENT

For Milestone 1, we used Slack and Google Drive to organize our tasks. To manage Milestone 2 better, we began using the free version of Trello, an online project management tool (<https://trello.com/home>). We organized our deliverables into lists in Trello indicating the current state of the deliverable: Not started, Doing, Code review, Done, and Snoozed (to indicate that the deliverable is waiting on something else to complete). Each deliverable card contains a description of the deliverable and additional sub-deliverables that are organized into checklists of tasks, each of which is assigned to a team member. You can also set due dates for each deliverable, so we were able to divide the M2 tasks into chunks, staggering the due dates throughout the 2 weeks. We were able to connect Trello to our team Slack channel as well, so that we receive Slack notifications when there is activity in Trello. This was really convenient to get everyone organized, and it's easy to see who's on track with their deliverables and who could use some help. We plan to continue using Trello for the remainder of the project. We continue to use Google Drive for storing and editing our working documents, and Slack for communication.