

SW Engineering CSC648/848 Spring 2019

SFSU House Me

Team 14

Davis Hoang (Davis.hoang2015@gmail.com) (Team Lead / Github Master)

Front-End

Jonathan Fox (Front-End Lead)

Kevin Mitsuda

Jarrett Lee

Back-End

Hashim Jacobs (Back-End Lead)

Eric Chen

Shalaka Aigal

Milestone 1

2/07/2019

Revision Table

03/07/2019	1.0	Initial Document
03/18/2019	2.0	Revised Document

Executive Summary

The lack of access to affordable housing and the increase in housing instability in the bay area has caused many San Francisco State University students to struggling to secure a place to live. According to the president and chief financial officer at San Francisco State, about 3,000 students are waitlisted for room and board every Fall semester. San Francisco State will be unable to accomodate room and board to every student because of the insufficient number of available spaces. This will cause many students to look at other possibilities for housing.

We are looking to develop a housing website for SFSU students to help ease the process of securing a place to live. Students will be able to quickly browse through available housing spaces in the bay area. Students will have the option to search by price or by distance or by housing type. Student will be able to browse through the our website without having to sign in. If a student wishes to contact the landlord then they will be prompted to input their login credentials. Furthermore, our web application will allow homeowners and landlords to list their available housing spaces. They will be able to upload photos, videos, and information of their vacant homes. Homeowners and landlords will have the options to accept or decline inquires about their available spaces from users.

Our site aims to serve a variety people, students who are looking for housing or homeowners who are looking to lease their house. The use of our site will be perfect for students who need an easier process to secure housing. In addition, students will be able to find more affordable housing through homeowners in the bay area.

Personae and main User Cases

Mary: Student looking for room

About Mary:

- Very busy
- Is not local
- Familiar with WWW tools and apps.
- Interested in the city
- Looking to meet new people



Goals and scenario:

- Freshman/ new to the area, looking for a place to rent that makes it convenient to attend SFSU.
- Looking for a web app that clean shows available rooms and makes it easy to find apartments in her parameters.
- If she finds an issue on the site, she is likely to report it to the administration of the site.

Use case:

Non Registered User Browsing Site

1. Mary wants to quickly find a room near SFSU to rent out for the year. So she enters the site and searches by distance, cost, and type and sees a number of listings that fulfill such requirements. She is then able to browse the listings with additional filtering options.

Non Registered User Contacting Poster

2. Mary has found a listing/ room that she would like to get more information from the poster, she will then be prompted to register before she can message the poster.

Carol: Student looking for roommates

About Carol:

- Very busy
- Familiar with the area
- Familiar with WWW tools and apps
- Loves the city
- Outgoing

Goals and scenario:

- Looking for a fellow student that shares similar hobbies to share a room with/ split the bill.
- Is very busy, so will not be monitoring the posting frequently

Use case:

Local Student Creates a Listing to Find Roommates

3. Carol will want to post a listing onto the site, indicating that they are looking for a roommate for a certain apartment. They will be prompted to register or log into the site. From there, there will be a form to input information such as location and price for the location along with an option indicating that the poster is looking for roommates.



Harold: Landowners looking for renters

About Harold:

- Lots of Free time
- Unfamiliar with WWW apps/ tools
- Owns multiple homes



Goals and scenario:

- Looking for someone to rent out unused rooms/ apartments to.
- Harold is unfamiliar with WWW apps, so he will require a web app that makes it simple to post listings without much effort.
- Spend less time having to manage their postings on the site.

Use case:

Landowner Posts Listings to Site

4. Harold will want to post their unused rooms onto the site. There will be an option to post listings onto the site with a simple form. This form will require basic information about the listing and images of the property. Before the form is complete they will be required to register onto the site, i.e. lazy registration.

Administration: Admin for site

About Administration:

- Working on or maintaining the site
- Familiar with WWW tools
- Will only be able to work for a few hours a day

Goals and scenario:

- Looking for a UI that allows easy maintenance and monitoring of the site.
- Listings need to be processed in a way to allow them to be approved and easily taken down by a single user/ admin.

Use case:

Admin's Unique Page to Quickly Maintain Site Needs

5. Administration will want to log into the site and be shown a page that includes a list of pending listings that have yet to go live and reported listings from users. From there, they will be able to take down or approve certain listings quickly.



List of main data items and entities

- Unregistered User : The unregistered user will be able to browse through all the property listings but will not be allowed to post any listing.
- Registered User : The registered user will have a user account on the website will be able to view all the property listings and can also post the listing provided they are logged in on the site. Registered Users can be both Landlord/Seller and Tenant.
- Landlord/Seller: A registered user who has a property to rent and can post the property on the website.
- Tenant: A SFSU student looking for a place to rent.
- Administrator : Users that have special privileges, and have the ability to remove posts from the site, remove items, issue warnings and bans from the site, and generally enforce the Code of Conduct for the site. Administrators also are responsible for helping users when needed.
- Property : The main data item is property listed for sale, or for purchase on the site. The following are the types of property which can be listed on the site.
 - House : Individual house with 1 or more rooms to be rented.
 - Apartment: Apartment with 1 or more rooms to be rented.
 - Room: Single or shared room to rent.
- Posting, registration, and search shall be main function.
- Registration details: User's name, email, contact number will be collected from the users during registration.

Functional Requirements

Unregistered Users:

- 1.1 Unregistered users shall be able to view housing types on the home page.
- 1.2 Unregistered users shall be able to sort by house type, price and distance.
- 1.3 Unregistered users shall be able to see a photos and description of the house.
- 1.4 Unregistered users shall be able to sign up with email and password.
- 1.5 Unregistered users shall be prompted to sign up before contacting homeowners.

Registered Users:

- 2.1 Registered users shall have the same functionality as an unregistered user.
- 2.2 Registered users shall be able to login with their email and password.
- 2.3 Registered users shall be able to contact homeowners about housing inquiry.
- 2.4 Registered users shall be able to post listing of their housing type.
- 2.5 Registered users shall be able to post photos, descriptions and videos of the house

Admin:

- 3.1 Admin users shall have the same functionality as a registered user
- 3.2 Admin users shall review the content and description of all uploaded house listings.
- 3.3 Admin users shall be able to add, delete and edit house listings.
- 3.4 Admin users shall be able to delete or block registered users.

House listings:

- 4.1 House listings shall be able to upload photos and videos.
- 4.2 House listings shall be able to add a description about the house.
- 4.3 House listing shall show the owner of the listing.
- 4.4 House listing shall be viewed by all users.
- 4.5 House listing shall be viewed on the home page.

Non-Functional Requirements

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0 (some may be provided in the class, some may be chosen by the student team but all tools and servers have to be approved by class CTO).
2. Application shall be optimized for standard desktop/laptop browsers (Google Chrome and Mozilla Firefox)
3. Selected application functions must render well on mobile devices
4. Data shall be stored in the team's chosen database technology: Database: PostgreSQL 10.12-10.14
5. No more than 50 concurrent users shall be accessing the application at any time
6. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
7. The language used shall be English.
8. Application shall be very easy to use and intuitive.
9. Google analytics shall be added
10. No e-mail clients shall be allowed
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated.
12. Site security: basic best practices shall be applied (as covered in the class)
13. Before posted live, all content (e.g. apartment listings and images) must be approved by site administrator
14. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
15. The website shall prominently display the following exact text on all pages *"SFSU Software Engineering Project CSC 648-848, Spring 2019. For Demonstration Only"* at the top of the WWW page. (Important so as to not confuse this with a real application).

Competitive Analysis

Feature	Zillow	Facebook Marketplace	Craigslist	Our Product
Text Search	+	+	+	-
Tag Filter	-	-	-	+
Housing List View	+	++	+	++
Housing Map View	+	-	-	-
List Home for Rent/Sale	+	++	+	+
Distance/Time to Campus	-	-	-	+

Zillow, Facebook Marketplace, and Craigslist are several examples of other products that provide a similar service to what we are providing. The advantage of our website is mainly focused on providing housing services for SFSU students. Zillow has a large map view that takes up a large amount of screen space. SFSU students are generally looking in the same area, so that kind of map view is unnecessary for us. The housing view of Facebook Marketplace is very simplistic and straightforward, without leaving out any important information. It is much more streamlined than Zillow's and Craigslist's housing views. We should build ours in a similar way to make the website easy to use. The other websites use a text search, but we will use a tag filter because it accomplishes a similar task with less work involved.

Our website will have features that specifically benefit SFSU students. We can show a listing's distance from campus, so that students can easily see how long it would take them to get to school.

High-level system architecture and technologies used

Deployment Platform

- Server Host: Google Compute Engine
- Operating System: Linux Ubuntu 18.10 x86_64
- Database: PostgreSQL 10.12-10.14
- Web Server: Node.js v11.9.0
- Server-Side Language: Javascript

Frameworks:

- Web Server: Node.js v11.9.0
- Server-Side Language: Javascript
- Web Framework: Reactstrap / Bootstrap 4

Browsers

Our site will be supported on browsers such as Chrome (version 50.0.2661 +) and Safari (version 6 +)

Team 14

- Davis Hoang (Team Lead / Github Master / Document Master)
- Jonathan Fox (Front-End Lead)
- Kevin Mitsuda
- Jarrett Lee
- Hashim Jacobs (Back-End Lead)
- Eric Chen
- Shalaka aigal

Checklist

- Team found a time slot to meet outside of the class : **DONE**
- Github master chosen : **DONE**
- Team decided and agreed together on using the listed SW tools and deployment server : **DONE**
- Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing : **ON TRACK**
- Team lead ensured that all team members read the final M1 and agree/understand it before submission : **DONE**
- Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.) : **DONE**