SW Engineering CSC648/848 Fall 2021

RUMI

Section 02 Team 01

—

Team Lead ➜ Alex Shirazi
(Email: aredjaian@mail.sfsu.edu)
Github Lead ➜ Jasmine Kilani
Front-end Leads ➜ Nakulan Karthikeyan
Anmol Burmy
Front-end Team ➜ Jasmine Kilani
Rasul Imanov
Back-end Leads ➜ Joshua Miranda
Cheng-Yu Chuang
Back-end Team ➜ Rasul Imanov

—

Milestone 1

(September 21, 2021)

History Table

Revisions	Date
M1 - v1.0	September 21, 2021
M1 - v2.0	September 28, 2021

Table Of Contents

—

—

1   Executive Summary

It is no secret that the San Francisco Bay Area is home to the nation's top tech talent. Out of 35,000 venture-backed tech companies emerging in the nation's top 25 tech hubs, approximately 42 percent --nearly half are based in the Bay Area. Each year approximately 68,000 people make their way into the city of San Francisco alone where the housing market continues to skyrocket and the competition for talent drives companies to offer employees housing and salary benefits to make up for the high costs of living. According to an ACS Data study, the city of San Francisco is also home to the highest percentage of 18-39 year olds living with roommates at 28%. The number of transplants who enter the Bay Area in need of housing far exceeds that of any other city in the U.S and only continues to grow year after year. The city of San Francisco is also home to San Francisco State University where nearly 30,000 students are enrolled at just one of the 50 colleges and universities based in the Bay Area. Every fall, a combined total of nearly 70,000 students move into the Bay Area to attend some of the highest ranked universities in the nation. One only needs to look so far as the countless decentralized and scattered groups on social media networks and discussion forms to see that the need for housing solutions for transplants is far overdue.

In an effort to meet this demand, our team has built RUMI - the bay area housing solution providing a seamless transition into life in the Bay Area. RUMI serves as a web-based application helping students and young professionals alike find rooms, rent apartments, and connect roommates in the SF Bay Area. RUMI users will have the ability to connect with other RUMI users in search of housing as well as access a directory of rooms available to rent. RUMI would also provide users with the ability to connect with other users in search of housing through in app messaging and searching by major, area of work, gender, hobbies, lifestyles, pets, and etc.. RUMI users also have the ability to list rooms available for rent and to search for apartments for listing directly by the owner of the property, removing the barriers and delays of the middleman or real estate agent in the renting process.

RUMI makes the search for housing in the Bay Area a walk in the park. Previously, searching for an apartment to rent or finding a compatible roommate would take countless hours browsing social media housing groups

and unorganized forms online. With RUMI, roommates no longer need to copy/paste their listing specifications countless times in online groups, discussion forms, and insecure sites online while constantly needing to check and track each platform for communication from potential roommates. Landlords no longer need to spend hours applying for verification or emailing real estate agents in search of tenants. What would previously be a hassle and take hours, can now be a simple, effective and enjoyable process.

The team at RUMI values the end user at every phase of the design, development, testing process. Our team of developers and engineers have first hand experience as students and young professionals in the very region this product has been designed for. With the increasingly unorganized and scattered housing need on social media groups, discussion forums, and the early 2000s hallway bulletin boards, the need for RUMI in the Bay Area is far overdue.

2 Personas and Main Use Cases

PERSONAS

The key personas that our product is geared towards are college freshmen, transfer students, graduate students, international students, and working professionals around the San Francisco bay area. A general characteristic for the personas are people that have **freshly arrived into the San Francisco bay area** with limited knowledge of the area, **passionate about higher education** and **pursuing studies in any degrees offered by San Francisco State University** or any other university in the area, **do not have the financial ability to rent their own individual apartments**, have **good communication skills**, and **able to handle pressure** well.

Let's analyze a persona named **Ana**. Ana has newly arrived in the city of San Francisco. She is a nineteen year old female student that has been accepted into San Francisco State University. She has chosen to pursue a bachelors degree in Social Justice at the university because she has heard so much about how the people of San Francisco are so passionate about social justice, diversity and equality. She is very outgoing and friendly, she can get along with just about everyone. And additionally, she is comfortable co-existing in a shared space because she has shared a house with her siblings and parents. She loves

the city but unfortunately, being so young and passionate, she is not financially stable as a first year student.

Chad is a transfer student from a nearby community college. He is 25 years old and is very outgoing. He has changed his area of study many times in community college because he wasn't satisfied with work opportunities for the majors he chose. Finally, he has been pursuing a transfer degree from his community college in computer science and recently accepted into San Francisco State University. However, because he has been in community college so long earning minimal income from side jobs, he is unable to afford an individual apartment in San Francisco.

Melanie is a graduate student who has been accepted into a masters program at San Francisco State University. Because she is doing a masters degree, she doesn't expect to be a graduate student for long and is also looking for short term housing. Previously, she completed her bachelor's degree in business at University of Texas where cost of living was much cheaper. Since she finds herself in a new location where she doesn't know a lot of people and is much more expensive to live in, she would like to live with roommates that share her interests in business and are also pursuing a masters degree.

Axel is an incoming international student from Germany. He is 21 and would like to pursue his degree in accounting at the San Francisco State University. Since he has just recently arrived into the bay area, he doesn't have many friends. Although he has no friends in the area, he is very sociable and would like to get out of his comfort zone by sharing a living space with other students or professionals around the bay area.

John is a professional software engineer at Google, and he is currently renting a 1 bedroom apartment near San Francisco State University. He is 28 years old and is usually busy with work and social life. Most of the time, he only uses his apartment to sleep. It has recently dawned on John that although he is making a handsome salary working at Google, he would like to cut back on some expenses given that living in the Bay Area is not cheap. So he decides to look for someone to sublease his living room so rent is cut in half.

USE CASES

Student looking for a room

A student is looking for a room to rent rather than an apartment because it is cheaper and the cost of living near San Francisco State University is expensive.

So student:

➔ Registers for an account on RUMI
➔ Logs into their account on RUMI
➔ Input their budget, living preferences, and different filters for room search
➔ Finds a room listing that matches their description
➔ Likes listing to show their interest in the room
➔ Messages host on details regarding the room

Student looking for another student to share a room with

A student is looking for another student to get a room with because she would like to go through the process of finding a room with another student sharing her interests.

So student:

➔ Registers for an account on RUMI
➔ Logs into their account on RUMI
➔ Input their budget, living preferences, and different filters for room search
➔ Finds a room listing that matches their description
➔ Checks for other people that are students who have shown interest in listing
➔ Messages the student also interested in listing
➔ Messages host to work out details on getting a room with another student in the host's residence

Networking

A student newly arrived in the area would like to get to know people, make friends, and room with people that they get along with.

So student:

➔ Registers for an account on RUMI
➔ Logs into their account on RUMI

➔ Input their budget, living preferences, and personal interests to match with people that share similar interests
➔ After looking at the results, they utilize messages to contact multiple people that match their interests
➔ They find a person that share their interests and get along with well
➔ They continue their friendship and move in with said person OR
➔ They keep connecting with others until they find the people that share their interests

Professional looking to share rent

A professional software engineer makes a comfortable yearly salary of $100k renting an apartment near San Francisco State University for $4k/month. Looking at the amount of rent the engineer pays and their salary, they decide to rent out their living room for some discount on the rent they pay every month.

So the professional:

➔ Registers for an account on RUMI
➔ Logs into their account on RUMI
➔ Posts his apartment and room details on RUMI along with roommate preferences
➔ The professional can review any messages they receive from potential future roommates and decide to accept or decline their proposal

3   List of Main Data Items and Entities

➔ **User**

Users who have made an account with RUMI will have complete access to the website; if they have not yet made an account, they shall be automatically redirected upon trying to interact with the website. Unregistered users can only view the website and look at posts, but will not be able to interact with other users or posts.

➔ **Admin User**

Website and System administrators will be able to utilize admin capabilities such that they will be able to (for instance) ban or remove users from the website.

➔ **User Profile**

The user profile will contain all the relevant information provided by the user, including (but not limited to) user ID, profile picture, room/roommate preferences, and other personal information.

➔ **User Preferences**

The user preferences specified by the user will be used to categorize the listed posts on RUMI to suit the user's needs.

➔ **Room Posts**

A page that features various living space posts and links to the post's user profile. This page will also have features that allow for post categorization to better pander to the user, comments to allow registered users to interact with the post, and reviews for each post to allow users to understand the living space's average rating, and more.

➔ **Roommate Posts**

A page that features various posts of users who are looking for a living space; each post will also have a link to their respective user profile along with post categorization to better pander to the user, comments to allow registered users to interact with the post, and reviews for each post to allow users to understand the overall experience other people had in living with poster.

➔ **Reviews**

Registered users can leave reviews regarding living spaces and other RUMI users based on their experiences.

➔ **Post Comments**

Registered users can interact with posts by leaving comments or responding to other comments.

➔ **Messaging Platform**

The messaging system allows registered users to interact with other RUMI users in a private messaging platform.

➔ **Room Preferences**

Registered users can assign living space or roommate preferences such as major, school, year level, smokers, pets, similar interests/hobbies, location, etc..

4   Initial List of Functional Requirements

➔ **Admin User**
1.   An admin shall receive reports about users.
2.   An admin shall ban any users from the website.
3.   An admin shall be able to delete posts.
4.   An admin shall be able to delete comments.

➔ **Guest User**
1.   A guest user shall be able to see the available rooms and roommates on the website.
2.   A guest user shall be able to search any room and roommate available on the website.
3.   A guest user shall be able to view other user's profiles.
4.   A guest user shall be able to sign up and make a new account.

➔ **Account**
1.   A guest user shall be able to make a new account.
2.   An account shall have a unique id for each user.
3.   An account shall require a username, which must pass the conditions of starting with a character (a-z or A-Z) and having the length of at least 3 characters.
4.   An account shall require a valid email.
5.   An account shall require a password, which must pass the conditions of having the length be at least 8 characters long and must contain at least 1 upper case letter and 1 number and 1 of the following special character (/ * - + ! @ # $ ^ &).
6.   An account shall require the age of the user.
7.   An account registration shall require the user to accept the Terms of Service and Privacy Rules of the website.

➔ **Registered User**
1.   A registered user shall be able to login to the website.
2.   A registered user shall be able to update their profile.
3.   A registered user shall be able to see the available rooms and roommates on the website.
4.   A registered user shall be able to search any room and roommate available on the website.

5. A registered user shall be able to create posts.
6. A registered user shall be able to delete their posted post.
7. A registered user shall be able to comment on the posts by other users.
8. A registered user shall be able to see other users' profiles.
9. A registered user shall be able to copy link, share or save room and roommate's profile.
10. A registered user shall be able to set a preferred search for finding a room (location, price, amenities etc) and roommates (majors, school, smoker?, pets?, language, same interest, hobbies etc).
11. A registered user shall be able to get alerts for matching profiles that they are looking for.

➔ **Profile**
1. All registered users shall have a profile.
2. A profile shall contain an about me and information about the user it belongs to.
3. A profile shall contain a profile picture of the user.
4. A profile shall contain all the posts posted by the user.
5. A profile shall contain all the posts liked by the user.

➔ **Posts**
1. A Post shall be posted only by a registered user.
2. A Post shall be posted by a user looking for a roommate or a room.
3. A Post shall be posted by a user who wants to rent a room.
4. A Post shall contain a photo, a caption or a description.
5. A post shall have a post creation date.
6. A post shall be deleted by the user who posted it.

➔ **Comments**
1. Comments shall be posted by registered users under posts.
2. Comments shall contain a comment made on the post and the name of the user who made the comment.
3. Comments shall be deleted by the user who posted it.
4. Comments shall have a creation time.

➔ **Messaging**
   1. Messages shall be sent from a registered user to another registered user.
   2. Messages shall have the message along with the time it was sent on.
   3. Messages shall be replied to.

➔ **Search**
   1. Guest users shall be able to search any post on the website
   2. Registered users shall be able to search any post on the website.
   3. Search shall be filtered to suit the preferability of the user who is searching.
   4. Search shall update the contents of the website as the search terms are entered.

5 List of Non-Functional Requirements

➔ **System requirements**
   1. Server shall have 0 blocker, critical issues in SonarQube
   2. Code coverage for system test shall be more than 70%
   3. Server shall follow MVC architecture
   4. Server shall follow node.js coding style
   5. Server shall deploy on the free AWS EC2 instance
   6. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0
   7. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development

➔ **Usability requirements**
   1. Server down time shall lower than 3 minutes every time updating server
   2. Server shall follow [Web Content Accessibility Guidelines](#)
   3. Each page shall support mobile layout
   4. Server unavailable time shall lower than 5% of the timeUsers shall have more than 95% chance that the experience server critical failure
   5. Server views shall be high contrast to let uses easier to use
   6. Server shall recover within 24 hours if a critical incident happens
   7. Server shall support fuzzy search
   8. Application shall be optimized for standard desktop/laptop browsers

9. Selected application functions must render well on mobile devices
10. The language used shall be English.
11. Application shall be very easy to use and intuitive.
12. No e-mail clients shall be allowed. You shall use webmail.
13. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.

➔ **Performance requirements**
1. Each page shall load visually within 1 second over an LTE connection
2. Server response time shall be lower than 200ms
3. Server shall be able to response 5 users simultaneously
4. Each page shall pass Google PageSpeed Insights
5. Google maps and analytics shall be added
6. No more than 100 concurrent users shall be accessing the application at any time

➔ **Storage, security, environmental requirements**
1. Database shall update date only by root user and server
2. Server shall limit the amount of room list data to query from database every time
3. All password shall be encrypted by bcript.js
4. Server shall allow only admin user to update all posts and users information and normal users to update their own posts and user information
5. Server shall have a rate limit of 20 calls/minute and a burst limit of 3 calls/second
6. Data shall be stored in the team's chosen database MySQL technology on the team's deployment server AWS.
7. Site security: basic best practices shall be applied (as covered in the class)

➔ **Marketing, legal requirements**
1. Each page shall have our application logo in the header
2. Each page shall have license in the footer
3. The website shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Fall 2021. For Demonstration Only" at the top of the WWW page. (Important so not to confuse this with a real application).

→ **Content**
1. Each file shall not exceed 2 Mbytes
2. Each image shall not exceed 10 megapixels
3. The date format shall be as follows: month.date.year

→ **Privacy**
1. Only database admin shall be able to see users' information
2. Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users.

6 Competitive Analysis

Competitors towards our app RUMI include many social media platforms. Facebook has a feature called Facebook groups to bring people together. Facebook has a very large platform and most of the world already uses it, therefore Facebook groups make it convenient for facebook users by allowing them to follow groups to join communities. These groups can be used to create connections between people based on a topic for example friends, car lovers, book clubs, or in our case in terms of competition, finding roommates. There are thousands of groups just for finding roommates. You can have any roommate, or there are some groups to help college students find roommates. The benefit of using our app RUMI over facebook groups is that we will have a UI implemented so college students can see all available options in an easy to understand format and visually appealing to the users.

More competitors towards our app are apps specifically made for finding roommates. There's an App called Roomster and it has over 1 million downloads on the android store. The app is targeted towards people finding rooms, houses and roommates.The app then shows a ui similar to instagram home where the user can scroll through the different roommate offers. They have a messaging system to message others. If the user is looking for roommates to add to their home, it asks for a location and monthly rental price accepted. If the user is looking for a place to live with roommates the app makes the user a post. In each post, users describe who they are, how much they are willing to pay monthly, personality traits, and schedules. Other users who have available rooms can message the posters about living together. However, the app shows everyone in a city that's looking for a roommate. It doesn't have any filters such as for students only, maybe if users don't want to see blank listings for spam or maybe users are only comfortable with a specific

gender as a roommate. Our app will have filters to sort through the listings based on what the user wants to see on the UI. Another problem with the app is that listings don't show whether roommates have been found or not. Our app will only show listings that are open.

Another competition is craigslist. On craigslist, they have very flexible listing options where people make listings whether they are looking for roommates or looking for a place with roommates. These posts can have any content on them to describe what they are looking for. You can post any contact information, how much rent you're willing to pay, and images if necessary. Although craigslist is very flexible on posting, it doesn't seem to be organized or have a UI comfortable for users.

	Facebook Groups	Roomster	Craigslist	RUMI
Mobile App support	x	x	x	x
Supported Messaging System	x	x	x	x
Easy to use	x		x	x
Beautiful UI		x		x
Roommate Filter				x
Source	https://www.facebook.com/groups/discover/	https://www.roomster.com/	https://sfbay.craigslist.org/	Team 01

In terms of an app to look for roommates, there's a lot of competition. Although facebook groups are geared towards bringing communities together, it serves as a viable option to find roommates. It has Mobile app support, you can send messages to other members in your group, and it's easy to use because most of the world uses facebook. Roomster is an app geared specifically towards people finding roommates and houses. It's got mobile support, messaging system, a beautiful UI, but the app is very complicated to use. You have to sign

up, you don't know who's found a room or not, there are many bank accounts and there's no filters to look for specific posts. Craigslist is another app people use to look for roommates, but the problem with craigslist is that there isn't a centralized system for people to easily look for roommates, their UI only consists of a list of html "ads" as posts for roommates. All these apps are viable but they all have something inconvenient about them. RUMI is gonna fix these problems because it will have mobile app support, it will have a strong messaging system, It will be easy to use, it will have a beautiful UI and it will support filters to help users search for roommates, or rooms with ease.

7   High-level System Architecture and Technologies Used

MVC Structure:
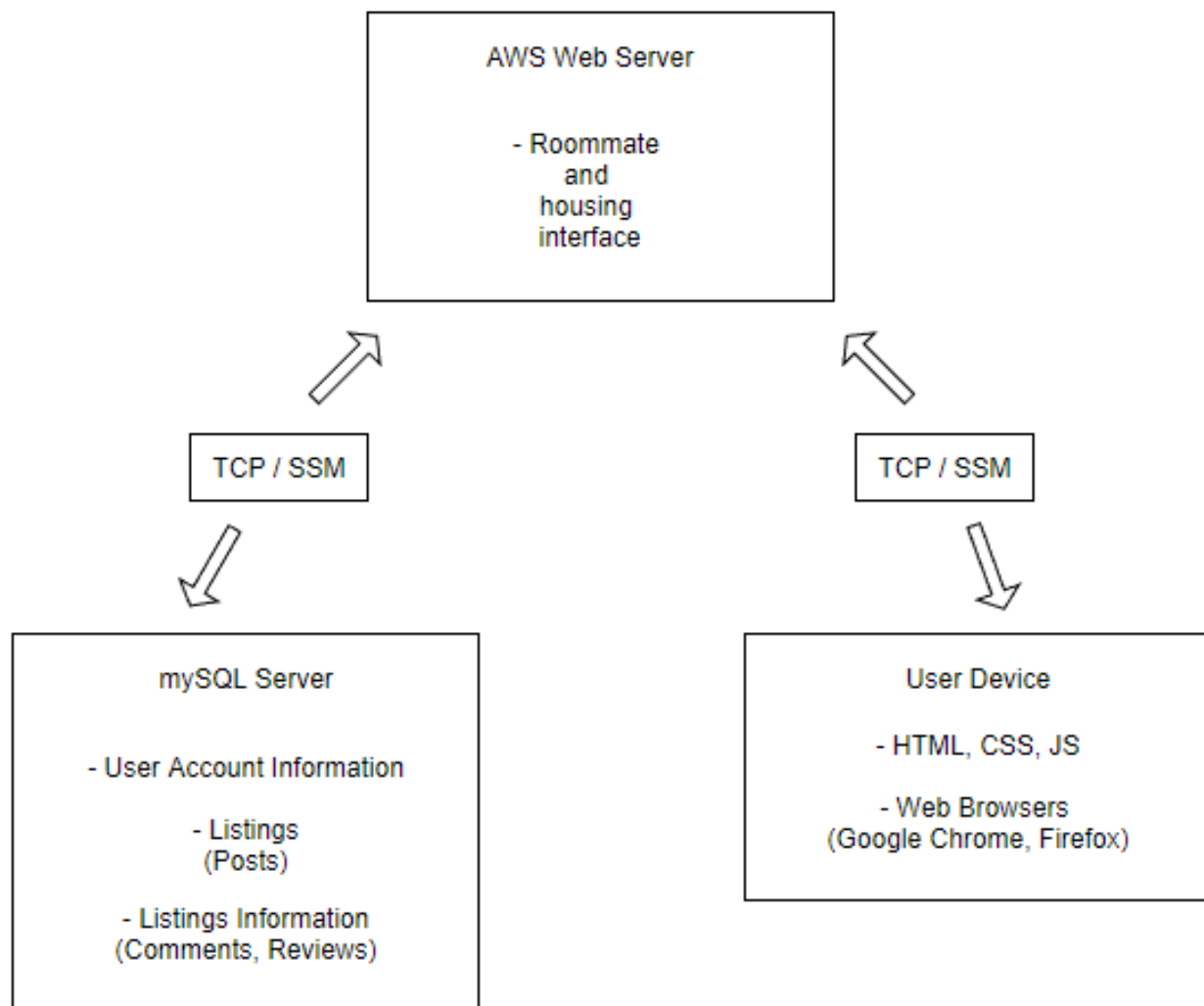
➔ Server Provider - AWS (RAM 1gb, vCPU 1 2.5 GHz)
➔ OS - Linux Ubuntu Server 20.04 LTS
➔ Database - MySQL 8.0
➔ Backend - Javascript ES2015
➔ Frontend - HTML, CSS, Javascript ES2015
➔ IDE - Visual Studio Code
➔ Javascript Library - React.js, Node.js

8   Team and Roles

1. Alex Shirazi : Team Lead
2. Jasmine Kilani : Github Master / Front-End Team Member
3. Nakulan Karthikeyan : Front-End Team Lead
4. Anmol Burmy : Front-End Team Lead
5. Joshua Miranda : Back-End Team Lead
6. Cheng-Yu Chuang : Back-End Team Member
7. Rasul Imanov : Back-End and Front-End Team Member

9   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 **ON TRACK**