

Software Engineering Final Submission

CSC 648/848 Fall 2021

Project: RoomMe

Section 02

Team 06

<http://3.22.208.237/>

Raul-Izaiah Rodriguez - Team lead - rrodriguez2@mail.sfsu.edu

Jay Jaber - Backend engineer/Deployment

Kaung Htun - Backend engineer

Vy Ngo - Frontend engineer

Nael Yun - Frontend engineer

Tanishq Pradhan - Full-stack engineer/Github master

Vandit Malik - Database manager

Milestone 5

Date Submitted	Dec 7th, 2021
-----------------------	----------------------

#1: Product summary:

- Name of product - RoomMe
- Final list of priority 1 functionality: functional requirements committed for delivery (will be our grading criteria)
 1. Users shall be able to register for a personal account.
 2. Users shall be able to log into their account.
 14. Users shall be able to contact each other for inquiries via email.
 3. Users shall be able to list one or more units/rooms to be rented.
 6. Users shall be able edit posted listings.
 7. Users shall be able to delete listings.
 8. Users shall be able to search for rooms to rent.
 9. Users shall be able to rent a room alone or with roommates.
 10. Users shall be able to find new roommates. Users shall be able to post and search roommate listings to find others with similar interests.
- What's unique about our product: What is unique about our product is the ability to filter by amenities required in a room and by interest desired in a roommate.
- Url for our website - <http://3.22.208.237/>

#3: Screenshots of site

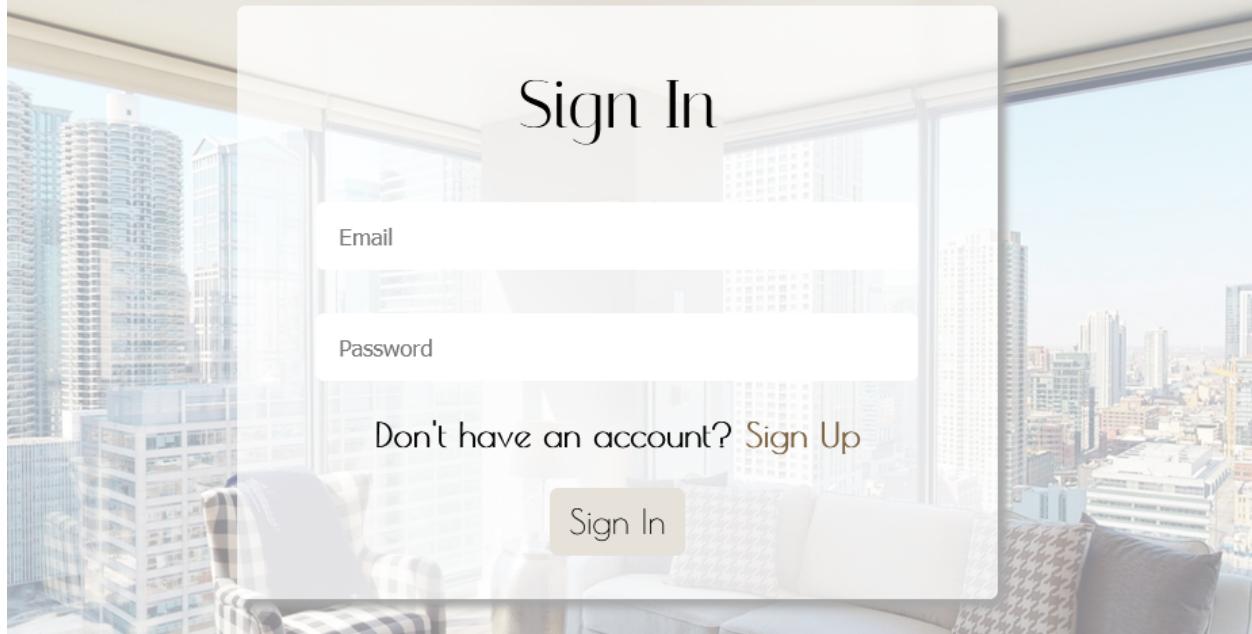
Home page/Search

The screenshot shows the homepage of the RoomMe website. At the top, there is a navigation bar with links for "Home", "About", "Sign Up", and "Sign In". Below the navigation bar is a search bar with a "Filter" button, a dropdown menu set to "All", and a search input field with placeholder text "Search...". A "Go" button is located to the right of the search input. The main content area displays four room listings in cards:

- Room**: A card featuring a large house icon inside a location pin shape. Below it is a blurred image of a room with a checkered sofa. The text "Nice Room" is displayed.
- Room**: A card showing a large window overlooking a city street with trees and buildings. The text "Absolutely nice room" is displayed.
- Room**: A card showing a bright living room with a large window, a sofa, and a lamp. The text "Room" is displayed.
- Room**: A card showing a modern interior with a fireplace, a sofa, and framed pictures on the wall. The text "Room" is displayed.

Login

Home About **RoomMe** Sign Up Sign In



Admin actions



Edit Profile

The screenshot shows a web browser window with the URL 3.22.208.237/admin. The page title is "Edit Profile". The main content area features the "RoomMe" logo and navigation links for Home, About, Upload, and a user icon. A large, semi-transparent overlay covers the center of the page, containing the text "Edit Your Profile" and a placeholder profile picture icon. Below the placeholder are three input fields: "Name" (placeholder: "Your Full Name"), "Password" (placeholder: "Your Password"), and "Email" (placeholder: "Your Email"). To the left of the "Name" field is a "Profile" link, and below it is a "View List" link. The background of the overlay shows a blurred image of a modern living room with a sofa and a cityscape view through large windows.

Profile

View List

Edit Your Profile

Name

Your Full Name

Password

Your Password

Email

Your Email

Edit User's Listings

[Home](#)[About](#)

RoomMe

[Upload](#)[Profile](#)[View List](#)[Edit Your List](#)

Absolutely nice room

[Edit](#)[Delete](#)

2 bedrooms, 1
bathroom

Upload Listing

The screenshot shows a mobile browser displaying the 'RoomMe' website at the URL 3.22.208.237/add-room. The page title is 'Upload Listing'. The main heading is 'List Your Room'. The form fields include:

- Title:** Room Title (placeholder text)
- Address:** Room Address (placeholder text)
- Description:** Room Description (placeholder text, with a validation message: "Please fill out this field.")
- Room Count:** Number of Rooms (dropdown menu)
- Occupancy:** Occupancy (dropdown menu)
- Lease Period:** Time for Rent (dropdown menu)
- Price:** \$0-\$999 (dropdown menu)
- Amenities:** A grid of checkboxes for various amenities:
 - Backyard
 - Furnished
 - Gym
 - Internet
 - Laundry
 - Parking
 - Pool
 - Private Bathroom
 - Utilities

#4: Screenshots of important DB tables

```
+-----+
4 rows in set (0.000 sec)

MariaDB [groupDB]> DESCRIBE User;
+-----+
| Field      | Type
+-----+
| id         | int(11)
| name       | varchar(64)
| phone_number | varchar(10)
| email      | varchar(128)
| password   | varchar(128)
| gender     | enum('male','female','other','prefer not to say')
+-----+
6 rows in set (0.001 sec)

MariaDB [groupDB]>
```

```
+-----+
14 rows in set (0.001 sec)

MariaDB [groupDB]> DESCRIBE RoomListing;
+-----+
| Field      | Type
+-----+
| id         | int(11)
| room_count | int(11)
| occupancy  | int(11)
| vacant     | tinyint(1)
| amenities   | set('furnished','utilities included','gym','pool','backyard','garage/parking','internet','private bathroom','laundry')
| room_photos | mediumblob
| description  | varchar(512)
| lease_period | int(11)
| userID      | int(11)
| address     | text
| latitude    | int(11)
| longitude   | int(11)
| title       | varchar(64)
| price_range  | enum('$0-$999','$1000-$1999','$2000-$2999','$3000-$3999','$4000-$5000')
+-----+
14 rows in set (0.001 sec)

MariaDB [groupDB]>
```

```
+-----+
14 rows in set (0.001 sec)

MariaDB [groupDB]> DESCRIBE RoommateListing;
+-----+
| Field      | Type
+-----+
| id         | int(11)
| description | varchar(512)
| interests   | set('exercise','gaming','partying','quiet','media','reading','food','sports')
| duration    | int(11)
| userID      | int(11)
| photo       | mediumblob
| title       | varchar(64)
+-----+
7 rows in set (0.001 sec)

MariaDB [groupDB]>
```

#5: Screenshots of task management system

The image displays two screenshots of a task management system, likely Trello, showing two separate boards: 'backend' and 'frontend'.

backend Board:

- Doing:**
 - /admin or profile route will 1. handle edit profile. 2. create listing form 3. listing (edit, delete)
 - Create an edit User Profile Route
- Done:**
 - Make 'about the team' page the response for GET /about/
 - Create route for landing page
 - Connect to DB on EC2 instance and create tables

frontend Board:

- Doing:**
 - Crying
- Done:**
 - Move 'about the team' handlebars markup from landing.hbs to a new file about.hbs
 - Write CSS media queries to ensure our website is responsive *IMPORTANT*
 - Add frontend form validation - make sure each input is the correct type (ex. number, positive price) and filled out before a user



+

🔍 Boards

YOUR WORKSPACES

cs648: RoomMe

Boards >



backend



documentation



frontend

#6: Team member contributions

** contribution list was compiled and discussed with the entire group present **

Izaiah: (# of commits: 55)

- CRUD operations on listings
- database design and implementation
- image conversion
- backend routes

Jay: (# of commits: 67)

- Free-text and parameter search algorithms
- backend routes
- listing views on frontend
- deployment

Kaung: (# of commits: 21)

- Authentication
- assisted in frontend implementation
- edit user profile route

Vy: (# of commits: 53)

- Frontend design and implementation for all pages
- CSS stylesheets

Nael: (# of commits: 12)

- Frontend design and implementation for all pages
- CSS stylesheets

Vandit: (# of commits: 7)

- Google Maps

Tanishq: (# of commits: 8)

- Github Repository, Frontend responsiveness

#7: Project Analysis: lessons learned, etc

One area for improvement in our project is ensuring that specifications are well understood as soon in the development process as possible. The non-functional requirements for the project were given clearly at the beginning of the semester; however, this being the first group project I've ever taken the lead on, there were some requirements that originally slipped through the cracks as I attempted to break them down for the team. This is a crucial skill for real-world development, as many requirements are not so clear-cut at the beginning. Another aspect we could improve on is separating responsibility between the front and backend teams. Not long into the project, we realized that having all seven members on one zoom call made it difficult for the sub-teams to focus on their respective tasks. We were bottlenecked into solving a single frontend or backend issue at a time. This was counterproductive to the overall development process.

The necessary improvements regarding our communication patterns began to take place naturally, as we shifted towards a more fragmented workflow in the later part of the semester. During Milestone 3, the most implementation-heavy, the backend and frontend teams split off to implement our respective tasks. I think this workflow allows much more flexibility as well as giving the team enough space to work effectively. In the future, I will likely push for this communication style. As far as managing specifications goes, I can ensure that I understand and implement them in digestible chunks, rather than taking them all at once. This is a more manageable process that is less error prone. In the future, I can do a better job of keeping track of all requirements and checking on them often to ensure that development proceeds along the right track.