

## Roommate web app

### Open Source Libraries:

**Django:** A high-level Python web framework that provides robust features for user authentication, database management (with Django ORM), and MVC (Model-View-Controller) architecture. It offers excellent support for building user-centric applications.

**React:** A JavaScript library for building user interfaces. React's component-based architecture and virtual DOM make it efficient for creating dynamic and interactive UI components, enhancing the user experience.

**Express.js:** A minimalistic web application framework for Node.js that simplifies the creation of server-side applications. It offers routing, middleware support, and integration with various databases, making it suitable for building RESTful APIs and handling backend logic.

**Redux:** A predictable state management library for JavaScript applications. Redux can help manage the application's state, including user profiles, preferences, and search results. It provides a single source of truth and facilitates state updates across components.

**Axios:** A popular HTTP client library for making API requests from JavaScript. Axios simplifies handling asynchronous requests to fetch or send data to the backend, making it useful for retrieving and updating roommate listings, user profiles, and messages.

**Leaflet:** An open-source JavaScript library for interactive maps. Leaflet can be utilized to integrate map functionality into the web app, allowing users to search for rooms or potential roommates based on location.

**Socket.IO:** A library that enables real-time, bidirectional communication between the client and server. Socket.IO can be used to implement real-time messaging functionality for users to communicate with potential roommates.

**Moment.js:** A JavaScript library for parsing, manipulating, and formatting dates and times. Moment.js can assist in handling and displaying timestamps for activities like listing creation, message exchanges, and review submissions.

**Chart.js:** A flexible and feature-rich JavaScript library for creating interactive charts and graphs. Chart.js can be employed to visualize data analytics and display statistics related to roommate matching, user ratings, or other relevant metrics.

**TensorFlow.js:** A JavaScript library for machine learning. TensorFlow.js can be leveraged to develop advanced roommate matching algorithms based on machine learning models, enabling more accurate compatibility assessments.

## Architecture

### User Registration and Authentication:

- User registration: Set up a registration procedure that allows new users to open an account on the platform and enter necessary data such as their name, email address, and password.
- User authentication: Use secure authentication technologies, such as password hashing, to confirm a user's identity during login.

### User Profiles and Preferences:

- User profiles: Enable users to establish and manage their profiles, which may include personal data, preferences, and suitability standards for roommates.
- Preferences and filters: Implement a system that allows users to indicate their choices for location, spending limit, lifestyle, and other pertinent factors. On the basis of these preferences, provide filtering options to improve roommate search results.

### Roommate Matching Algorithm:

- Compatibility evaluation: Create an algorithm or list of standards to match users according to their tastes, way of life, and other compatibility-related aspects. Think about things like pet friendliness, cleanliness, smoking policies, and noise tolerance.
- Ranking and scoring: provide possible roommate matches ratings or scores to provide consumers a concrete indicator of how compatible they are.

### Roommate Listings and Search:

- Listing creation: Allow users to create and manage roommate or available room listings, complete with information on location, rent, amenities, and roommate preferences.
- Search functionality: Users should be able to search for available rooms or potential roommates based on location, price range, and other given criteria.

### Messaging and Communication:

- In-app messaging: Implement a message feature within the app so users may communicate with potential housemates while maintaining their privacy and security.
- Notifications: Set up a notification system to inform users of new messages, suggested roommate matches, or pertinent developments.

### Reviews and Ratings:

- Review system: Introduce a method for users to rate and review their current or former housemates, fostering trust and offering insightful input for other users.

- Rating calculation: Calculate overall ratings and build user reputation scores based on reviews and other relevant data.

#### Trust and Safety Measures:

- User verification: Implement user verification procedures, such as email, phone, and ID verification, to confirm users' identities and information.
- Reporting and moderation: Establish a means for users to report questionable or inappropriate activity, as well as a moderation procedure to deal with such reports quickly and effectively.

#### Data Security and Privacy:

- Data protection: Implement secure user data transfer and storage while adhering to best practices for encryption, access restrictions, and data privacy laws.
- Privacy settings: Give users the chance to choose how much information is published on their profiles and who can contact them.

#### What to store in databases:

##### User Data:

- User profiles: Information such as name, email, username, password (hashed), contact details, and additional optional details like bio, profile picture, and preferences.
- User preferences: Criteria like location, budget range, lifestyle preferences, cleanliness, smoking habits, pet preferences, and any other relevant factors that contribute to roommate compatibility.
- User ratings and reviews: Data related to user ratings, reviews, and feedback left by other users after sharing a living space.

##### Roommate Listings:

- Listing details: Information about available rooms, including location, rent, lease terms, amenities, property type, room size, utilities, and any specific requirements or restrictions.
- Roommate preferences: Details of the type of roommate being sought, including preferred gender, age range, occupation, and other desired characteristics.
- Listing status: Keep track of whether a listing is active, inactive, or has been rented out.

##### Messaging and Communication:

- Messages: Content of conversations between users, including sender, recipient, timestamp, and message content.

- Notification preferences: User preferences for receiving notifications related to new messages or updates on listings.

#### Search and Matching Data:

- Match results: Data related to roommate matches, including compatibility scores, match suggestions, and any specific attributes that contribute to the matching algorithm.
- Search history: Keep track of user search history, search queries, and filters applied to improve the search experience and provide personalized recommendations.

#### Trust and Safety:

- Verification data: Store information related to user verifications, such as email verification status, phone number verification status, and any other identity verification processes implemented.
- Reported content: Keep track of reported users, listings, or messages, along with any corresponding actions taken.