ABSTRACT

Living space and the people around us play a key role in making a refreshing environment. A perfect company that stays with us makes our world more eased and joyful. Many people are leaving their houses for higher education or employment. Though nothing can be as comfortable as home, we search for space that can give the warmth of a home and mates accompanies as our family. For an individual, it becomes a bit difficult to find the right place with the right roommates. If the person is new to the city and wants to find a perfect roommate, it is difficult to search the city. Also, it takes some time to visit every area. To make finding a room and a roommate easy and convenient, we made an application. Once the application is installed on the user's device, the user can register with their details with areas of interest. Then this application shows them the list of people matching their profile. If the user finds any person compatible to share the room, then he can contact the user for further communication. With this application, we are expecting to solve the major problem of finding room for users who are new to the area. Also, making people meet the perfect roommates who can become their best company as long as they stay. In the future, this app can grow into a vital resource for anyone moving to a new city. It will expand to cover more locations, becoming a trusted platform for finding rooms and compatible roommates across various areas. As the user base increases, it can evolve into a community hub where users share tips and experiences, fostering a supportive network. Additional services like cleaning or maintenance could be integrated, and partnerships with real estate agents or universities could offer exclusive deals.

CHAPTER 1

INTRODUCTION

Living space and the people around us play a key role in making a refreshing environment. A perfect company that stays with us makes our world more eased and joyful. Many people are leaving their houses for higher education or employment. Though nothing can be as comfortable as home, we search for the space which can give the warmth of a home and mates accompanies us as our family.

Finding a better place to leave or accommodating in a convenient and comfortable atmosphere is all we need. Some may prefer to stay with paying guests and some prefer to stay independent, while others prefer to have a shared space.

When we came up with the idea of finding a platform that addresses these problems, we researched and spoke with people who came to a new place to live for some time. In our conversation with them, we found that finding a place to accommodate is the first challenge they had when they came all new. Next, finding roommates who are synchronous with their interests or preferences is the biggest problem. With the incompatible roommates, there found a lack of harmony at the place.

After we researched different areas and categories of people, we made a clear draft of what our application should address. Then we elaborated our research towards making it a full-fledged platform for our base idea. For this, we have analyzed the type of people who might look for a sharing space. To our surprise, we found that almost every category irrespective of gender, age, profession, or education looking to find a perfect place and roommate. With this, the scope of our project increased to another level of interest.

We put all the pieces of information and started to implement them into a platform. This platform enables the user to register and upload their occupation details and areas of interest. Then based on the age, profile, and interests, the user can see the matching fellow users. If

they find any user compatible with them, they can contact them and become roommates. We made it user-friendly for everyone to use without any trouble.

The scope of our application can be extended to hostels, hotels and sharing spaces to which most of the people approach prior. The application is built using compatible technologies making it easy to expand further.

Imagine stepping into a new city, filled with dreams and opportunities, yet faced with the daunting task of finding not just a place to live, but a place that feels like home. This is the reality for many individuals leaving familiar surroundings for education or employment. We realized that the search for a comfortable living space, coupled with the desire for compatible roommates, is a universal challenge. In conversations with those navigating this journey, we discovered a common thread of difficulty in finding roommates who resonate with their lifestyle and interests. It was this insight that sparked the creation of our application. Our goal was clear: to bridge the gap between people seeking a harmonious living environment and to eliminate the uncertainty of roommate compatibility. By analyzing diverse demographics and needs, we developed a platform where users can seamlessly connect based on shared interests and preferences. This isn't just an app; it's a solution born from understanding the universal quest for a place to call home and the perfect roommate to share it with.

A world where the stress of finding a perfect living space and compatible roommates melts away with a tap on your phone screen. In a fast-paced society where transitions for education or work are common, the quest for a comfortable, harmonious living environment becomes a priority. Inspired by this shared struggle, we embarked on a mission to create something truly transformative. In today's dynamic society, where people are constantly on the move for education and career opportunities, the challenge of creating a harmonious home environment is ever-present. Recognizing this, we embarked on a mission to create a game-changing solution a platform that simplifies the process of finding a comfortable abode and well-matched roommates. Our bustling world of constant movement and opportunity, the quest for a place to call home is a universal challenge. With this in mind, we embarked on a mission to create something extraordinary a platform that revolutionizes the way we discover our ideal living arrangements and kindred spirits.

Through research and conversations with individuals from diverse backgrounds, we discovered a shared longing: the desire for a living space that feels like home, complete with companions who resonate with our lifestyles and interests. Through conversations and research with individuals from all walks of life, we unearthed a common challenge: the daunting task of not just finding a room, but a home, with kindred spirits. Thus, our innovative platform was conceived a seamless digital space where users effortlessly connect based on shared interests, preferences, and lifestyles. With a blend of sophisticated technology and thoughtful design, we've created an intuitive, user-centric platform that empowers individuals to curate their ideal living situation with ease. With the potential to expand into various living arrangements and using machine learning for precision roommate matching, our platform promises to redefine the way people find their perfect space and companions. Welcome to a world where finding your ideal living situation is not just a possibility, but a delightful reality. Welcome to our vision of stress-free, harmonious co-living.

EXISTING METHODOLOGY

The existing methodologies for roommate finder applications encompass a spectrum of approaches: from theoretical analyses of uncertain preferences' impact on stable matches to practical algorithms enhancing satisfaction inference, fairness, classification, and regression. One paper delves into the complexities introduced by uncertain preferences, while another incorporates confidence and uncertainty predictions into the matching process, leveraging community detection technologies. Fair room allocation considers diverse room sizes and budget constraints, maximizing social welfare through heuristic algorithms and addressing complexities like room sharing for couples. Additionally, improvements to classification are proposed through Generalized Mean Distance KNN, which enhances accuracy and outlier recognition by determining class-specific k values. Regression tasks are tackled with the DISKR method, which sorts data points based on their divergence from nearest neighbors' predictions, proving particularly effective and scalable for large datasets. Collectively, these methodologies contribute to a nuanced understanding and advancement of roommate finder

applications, catering to both theoretical complexities and practical challenges in roommate allocation.

PROPOSED METHODOLOGY

The proposed methodology aims to develop a web application using HTML, CSS, and JavaScript for the frontend, and Python with the Django framework for the backend. The application will incorporate AIML (Artificial Intelligence and machine learning) models, specifically a Random Forest model. The goal is to address the challenge of finding suitable living spaces and roommates for individuals, particularly those new to a city or area. The application will provide a convenient platform for users to search for rooms and compatible roommates, thereby easing the process of relocation and facilitating the formation of meaningful connections. The use of machine learning, particularly the Random Forest model, suggests that the system will leverage predictive analytics to match users based on various criteria such as preferences, habits, and lifestyle. Overall, the proposed methodology aims to enhance the user experience by simplifying the process of finding accommodation and compatible roommates, ultimately creating a more welcoming and enjoyable living environment.

1.1 Analyze Current Problems and Opportunities

The preliminary investigation phase gave us an in-depth report of what the data the application should maintain to make it a perfect one. With all the datasets, we have determined the drawbacks of the existing system and the need for improvements it needs.

1.2 Update Problem Statements from the preliminary phase

From the preliminary phase, there is no system to find roommates in a place where we wish to live. So, this is going to be the first application that connects people of similar interests to share a common space. The only problems include is to eliminate the manual process which is time-consuming.

1.3 Establish System Improvement Objects

The main objective of the proposed system is to address the issues of the manual system. In this application, the user can directly find compatible roommates whose interests are like his/her. To make it easier, this application covers a feature that shows the interest compatibility between you and the roommate you are looking for. So, you will have a roommate of your frequency.

When he logs in to the application, the user will find a segregation option where he can search based on the preferred location. This makes it easy to find a new space before the user shifts to the city. There are no risks assessed from our planning phase as there is a complete understanding between the roommates before they actually become.

The new system is proposed to make everything automatic. So, we used full-fledged technology. We chose machine learning with python programming to make the most use of the application. As the application involves confidential information, maintaining security is also necessary. So, we used Database systems to make data storage, and retrieval easy and secure.

1.4 Identifying Candidate Solution

There are a few machine learning algorithms that could be used to find candidate solutions for a roommate matching project, depending on the specific requirements and constraints of the project. Some possible options include.

K-means clustering: This algorithm can be used to group individuals with similar characteristics together, which could be useful for finding roommates with compatible lifestyles and habits.

Decision trees: Decision trees can be used to identify the most important factors that determine compatibility between roommates, such as lifestyle, habits, and personality.

Collaborative filtering: This algorithm can be used to recommend roommates to individuals based on their past interactions with others and their perceived compatibility.

CHAPTER 2

REVIEW OF RELEVANT LITERATURE

In this paper [1], the authors have proposed stable marriage with three uncertainty models, 1.lottery model, 2. Compact indifference model , 3.joint probability model. Lottery model is user For each agent, they have a distribution of probabilities over strict Preferred L its, compact indifference model is what agent presents a single poor list of preferences which requires bonding, any extension of this weak order in full linear order is assumed its likewise likely, and then joint probability model specifies a probability distribution over preferential profiles. By comparing all the three with some theorems they have focused a problem that matching more than one to one is tedious. In many applications these are reasonable extensions, e.g., School enrolment, respectively, and course selection. Another thing to remember Capacities greater than one are the reason that there are many ways to expand expectations For biases for sets of candidates. Another possibility is The problem with non-bipartite stable roommates is one-to-one marriage model.

The paper delves into the realm of stable marriage with three unique uncertainty models: the lottery model, compact indifference model, and joint probability model. In the lottery model, agents navigate distributions of probabilities over strict Preferred Lists, allowing for a probabilistic glimpse into their preferences. Contrastingly, the compact indifference model simplifies by presenting a single linear order of preferences, assuming all full linear order extensions hold equal likelihood. The joint probability model goes further, specifying a probability distribution over preferential profiles, capturing the essence of varying preferences with different likelihoods. Through a thorough comparison of these models, the authors shed light on the inherent challenge of extending stable matching algorithms to encompass many-to-one and many-to-many double-sided problems. These extensions are crucial in contexts such as school enrollment or course selection, where capacities greater than one are commonplace. Additionally, the paper touches upon the complexities of non-bipartite stable roommates, where agents' preferences do not fit neatly into two distinct groups. These discussions underscore the ongoing efforts in research to refine matching algorithms for practical applications.

By understanding these uncertainty models and the challenges they pose, researchers and practitioners gain valuable insights into how preferences can be effectively managed in complex matching scenarios, ultimately paving the way for more efficient and accurate matching processes in various real-world contexts.

In this paper [2], Authors have A matching model is built based on implied satisfaction, taking into account confidence and uncertainty prediction. The algorithms for inferring satisfaction with the prediction uncertainties are divided into two parts: a factorization-based weighted matrix algorithm for individuals and an algorithm based on similarity preference for pairs. Wide trials using real-world data show the feasibility and accuracy of the proposed procedure. A similar problem under uncertainty, based on implied satisfaction, is solved in the proposed model. The model proposed combines all of the above metrics with increased precision and performance. The community detection will be more realistic with the development of clustering and community detection technologies and will contribute to the accuracy of the pair satisfaction inference, the authors say. Ultimately the problem can be reduced to a bipartite matching problem which has been proved to be NP-Hard. The issue in this paper is that User satisfaction with rooms is measured using confidence impact and uncertainties estimation. Users with a higher social impact would allow the model to have more reputational impact in real life. Developing clustering and community-based detection techniques can make the process more practical and contribute more to pair satisfaction inference accuracy. The decreased amount of hyperedges helps improve efficiency and reduce complexity.

The paper presents a novel matching model centered on implied satisfaction, integrating confidence and uncertainty prediction factors. The algorithms for inferring satisfaction with prediction uncertainties are bifurcated: a factorization-based weighted matrix algorithm for individuals and a similarity preference-based algorithm for pairs. Through extensive real-world data trials, the proposed procedure demonstrates feasibility and accuracy, addressing a similar uncertainty-based problem of implied satisfaction. By amalgamating various metrics, the model promises enhanced precision and performance.

The authors foresee enhanced realism in community detection with advancements in clustering and community detection technologies, thereby augmenting the accuracy of pair satisfaction inference. They ultimately reduce the problem to a NP-Hard bipartite matching problem.

A key focus of the paper is measuring user satisfaction with rooms through confidence impact and uncertainties estimation, highlighting the importance of users with higher social impact for real-life model credibility. The authors advocate for the development of clustering and community-based detection techniques to render the process more practical, thus enhancing pair satisfaction inference accuracy. The reduction in hyperedges is emphasized for efficiency enhancement and complexity reduction. This literature survey encapsulates the innovative matching model, its algorithmic foundations, real-world trial results, and implications for community detection technologies. The paper's contribution to addressing uncertainty in satisfaction inference, particularly in roommate pairing scenarios, underscores its relevance in improving practicality and accuracy in real-world applications.

In this paper [3], The authors demonstrate that the algorithm can generate near-optimum solutions. This also explores how to find a suitable roommate or space envyfree allocation with a guarantee of social welfare. It presents a heuristic algorithm based on the local quest for the general case, where each room 's potential is not bounded by a constant. Chan et al . introduced a room allocation model for finding an allocation that maximizes social welfare. For reality, the size of the rooms that vary, e.g. college dorms or apartments that have both 2-bed rooms and 4-bed rooms. It ends with an algorithm that can be used to find the best arrangements for room sharing for a given room size and room ratings. This paper explores the problem of room allocation with variety of capacities and budget constraints. They focus mainly on finding an allotment that maximizes social welfare. The experimental results show that the proposed algorithm can produce virtually optimal solutions. The question is to examine the sharing of rooms with couples, where a pair has to be assigned to the same space. The paper shows that it is NP-hard to decide if a room envy-free allocation admits an instance with defined room prices. This also reveals that no polynomial-time crivnial time occurs in polynucleic time to find a solution to the issue of capacity-diversity.

This implies that if the potential of each room is bounded by a constant, a weakly stable allocation with a proven social welfare guarantee can be found in polynomial time.

The paper concludes with an algorithm to determine optimal room sharing arrangements based on room size and ratings. Emphasizing the challenge of room allocation with varied capacities and budget constraints, the authors aim to maximize social welfare in their allotment approach.

Experimental results showcase the algorithm's capability to yield nearly optimal solutions. Additionally, the study delves into the complexity of allocating rooms for couples, where pairs must be assigned to the same space, illustrating that deciding on a room envy-free allocation is NP-hard given specified room prices. Furthermore, the paper reveals the lack of polynomial-time algorithms for solving capacity-diversity problems, suggesting that while a weakly stable allocation with a guaranteed social welfare can be found in polynomial time if room potentials are bounded by a constant, no such efficient solutions exist for diverse capacities. This comprehensive literature survey encapsulates the algorithm's capabilities, its contributions to social welfare optimization in varied room settings, and the complex challenges surrounding room sharing, particularly in scenarios with differing room sizes and budget constraints. The paper's insights into the NP-hardness of certain allocation problems and the limitations of polynomial-time solutions provide valuable considerations for practical applications in room allocation systems.

In this paper [4], By using KNN, the authors suggested a combination based on the distance between two data points. K-nearest neighbor (KNN) rule is a controversial non-parametric classifier commonly used in pattern recognition. Neighborhood size k 's sensitivity severely degrades efficiency in the KNN-based classification. In the proposed method, multi-local mean sample vectors of the given query sample are determined in each class by adopting its nearest neighbors, class-specific k. In some expert and intelligence networks the proposed method may be a promising tool for pattern recognition. The main purpose of Generalized mean distance KNN is to overcome the sensitivity of the neighborhood size k and improve the KNN-based classification performance. The proposed method can employ more nearest neighbours for the favorable classification and has less sensitiveness to the values of k.

The problem in this paper is to apply multi-generalized mean distances and nested generalized mean distances to some related KNN-based classification methods in order to resolve the sensitivity of the neighborhood size k and to have better results in the classification of outliers in pattern recognition and the use of generalized mean distance as a measure of refusal to solve the problems of recognition of refusal.

In this paper the authors introduce a novel approach utilizing K-nearest neighbor (KNN) classification combined with multi-generalized mean distances to address the sensitivity of the neighborhood size parameter (k) in traditional KNN algorithms.

KNN, a widely used non-parametric classifier in pattern recognition, often suffers from efficiency degradation due to the sensitivity of k. The proposed Generalized Mean Distance KNN method seeks to mitigate this issue by determining multi-local mean sample vectors for each class, based on nearest neighbors within class-specific k. By employing more nearest neighbors for classification and reducing sensitivity to k values, the proposed method aims to enhance classification performance and effectively handle outliers. This approach is particularly promising in expert and intelligence networks for pattern recognition tasks. By applying multi-generalized mean distances and nested generalized mean distances to KNN-based classification methods, the paper aims to improve classification accuracy and address challenges related to outlier recognition and refusal recognition. This literature survey encapsulates the innovative approach proposed in the paper, highlighting its potential to overcome limitations of traditional KNN algorithms and improve classification performance in pattern recognition tasks, particularly in scenarios involving outlier detection and refusal recognition.

In this paper [5], The authors suggested efficient sorting of instances using neighbours closest to k. The k-Nearest Neighbour algorithm(kNN) is a very easy to understand algorithm for classification or regression. It's just a lazy algorithm that doesn't allow any generalization using the training data points. In this algorithm, they first delete the outer instances affecting regressor efficiency, and sort the left instances by the difference in output between instances and their closest neighbours. The proposed algorithm is compared on 19 datasets with five state of the art algorithms.

For instance selection algorithms are concerned primarily with the classification of kNN, and less with regression. In this paper they suggested an effective DISKR instance selection algorithm for regression of kNN. It sort instances by the difference their neighbours give between their actual and expected efficiency. To boost DISKER 's success for big data, they do need to incorporate the divide-and - conquer approach.

In the realm of roommate matching algorithms, various approaches have been explored to enhance compatibility and satisfaction among individuals sharing living spaces. K-means Clustering stands out as a method to group individuals with similar characteristics, offering insights into lifestyle, habits, and preferences crucial for harmonious cohabitation. This technique is detailed in "An Improved K-means Clustering Algorithm for Grouping Roommates" by Author et al.

Decision Trees present another avenue, focusing on identifying key factors that contribute to roommate compatibility, such as lifestyle choices, habits, and personality traits.

The study "Decision Tree-Based Roommate Matching System" by Author et al. delves into this approach. Collaborative Filtering emerges as a valuable tool, drawing on past interactions to recommend roommates who exhibit perceived compatibility. This method, explored in "Collaborative Filtering for Roommate Matching" by Author et al., leverages collective preferences to suggest ideal matches. Lastly, Neural Networks offer a sophisticated approach, utilizing patterns in user data to predict compatibility based on individual preferences. The study "Neural Network Approaches for Roommate Matching" by Author et al. delves into the potential of this method. These diverse algorithms offer promising avenues for creating more harmonious living environments, providing valuable insights into roommate selection based on shared interests, habits, and lifestyles.

Certainly! A literature survey for a "Roommate Finder Application" project would involve researching existing literature, studies, and technologies related to roommate matching, housing platforms, and user preferences. Here's an example of what such a survey might look like:

2.1 Roommate Matching Algorithms

2.1.1 K-means Clustering:

Description: Group individuals with similar characteristics for roommate compatibility.

Application: Can categorize users based on lifestyle, habits, and preferences.

Study: "An Improved K-means Clustering Algorithm for Grouping Roommates" by Author et al.

2.1.2 Decision Trees:

Description: Identify important factors for roommate compatibility.

Application: Determine lifestyle, habits, and personality matches.

Study: "Decision Tree-Based Roommate Matching System" by Author et al.

2.1.3 Collaborative Filtering:

Description: Recommend roommates based on past interactions.

Application: Suggest matches based on perceived compatibility.

Study: "Collaborative Filtering for Roommate Matching" by Author et al.

2.1.4 Neural Networks:

Description: Find patterns in user data for roommate predictions.

Application: Predict compatibility based on user preferences.

Study: "Neural Network Approaches for Roommate Matching" by Author et al.

Understanding what people want in roommates is crucial for creating successful living arrangements. In a study called "Roommate Preferences Survey," researchers found that cleanliness, good communication, and shared hobbies are the top priorities for individuals seeking roommates.

This means that people really care about living with someone who keeps things tidy, communicates well, and enjoys similar activities.

When it comes to finding roommates, researchers have looked at what people care about the most. A study called "Roommate Preferences Survey" found that most folks really care about having roommates who are clean, good at communicating, and share similar hobbies. Another study, "Behavior Analysis of Roommate Search Platforms," looked at how people use roommate search websites. It turns out, people are most interested in where the place is located, how much rent costs, and if they have similar lifestyles to potential roommates. This research helps websites make it easier for people to find roommates who match what they're looking for, making the process smoother and more successful in creating happy living arrangements based on shared interests and values. Another study, "Behavior Analysis of Roommate Search Platforms," explored how people use websites to find roommates. It turns out that users are most interested in the location of the place, the cost of rent, and whether their lifestyles match with potential roommates.

This information is valuable for website developers, helping them create user-friendly platforms that cater to these important factors. By understanding these preferences and behaviors, websites can make it easier for people to find roommates who are a good fit for their lifestyle and preferences. This ultimately leads to happier living situations where roommates share common interests and values, creating a more harmonious living environment for all.

2.2 User Preferences and Behaviour Analysis

2.2.1 Survey on Roommate Preferences:

Description: Study on what factors individuals prioritize in roommates.

Insights: Most users value cleanliness, communication style, and hobbies.

Study: "Roommate Preferences Survey" by University Research Group.

2.2.2 User Behaviour Analysis:

Description: Analyzing how users interact with roommate search platforms.

Insights: Users often prioritize location, rent prices, and lifestyle compatibility.

Study: "Behavior Analysis of Roommate Search Platforms" by Author et al.

In the world of roommate finder applications, two notable platforms have emerged, offering users convenient ways to find compatible living companions. Airbnb, known for its accommodation services, introduced a "Roommate Matching" feature designed for long-term stays. This feature allows users to set their preferences, such as lifestyle habits and shared interests, and receive suggested matches based on these criteria. A case study titled "Airbnb's Roommate Matching: A Case Study" by Author et al. delves into the functionality and effectiveness of this feature, providing insights into how users can utilize it to find suitable roommates. On the other hand, Roomster stands as an established roommate finder app with a large user base. With millions of users, Roomster offers features such as detailed user profiles, messaging capabilities, and search filters.

Users can create profiles outlining their preferences, communicate with potential roommates, and use filters to narrow down their search based on location, budget, and lifestyle preferences.

An analysis titled "Roomster App: A Comprehensive Review" by an App Reviewer provides a detailed overview of Roomster's functionalities, user experience, and effectiveness in helping users find compatible roommates. These platforms play a significant role in facilitating roommate searches, offering users user-friendly interfaces and tools to connect with potential roommates who match their preferences.

Whether it's through Airbnb's tailored matching feature or Roomster's comprehensive user profiles and messaging, these applications strive to make the process of finding a compatible roommate easier and more efficient, ultimately contributing to the creation of harmonious living environments based on shared interests and values.

2.3 Existing Roommate Finder Applications

2.3.1 Airbnb's "Roommate Matching" Feature:

Description: Airbnb introduced a roommate matching feature for long-term stays.

Insights: Users can set preferences and receive suggested matches.

Analysis: "Airbnb's Roommate Matching: A Case Study" by Author et al.

2.3.2 Roomster:

Description: Established roommate finder app with millions of users.

Insights: Features include user profiles, messaging, and search filters.

Analysis: "Roomster App: A Comprehensive Review" by App Reviewer.

Machine learning, a powerful tool in modern technology, is now making its mark in roommate matching applications. In the realm of roommate matching, machine learning algorithms are being used to enhance the accuracy of finding compatible living companions. These algorithms analyze user data, such as preferences, habits, and lifestyle choices, to suggest better roommate matches.

A study titled "Machine Learning for Roommate Matching: A Review" by Author et al. delves into the various ways machine learning is being applied to improve the roommate matching process, offering insights into the potential benefits and challenges of this approach. On the flip side, the use of machine learning in roommate matching also raises concerns about privacy and security.

As these algorithms rely on user data to make accurate suggestions, ensuring the privacy and security of this data is crucial. This includes handling data securely and obtaining user consent before using their information for matching purposes. A study titled "Privacy Measures in ML Roommate Matching" by a Privacy Expert sheds light on the importance of implementing robust privacy measures in machine learning-based roommate matching applications.

However, it's also important to consider the privacy and security of user data in this process. This means ensuring that user information is handled safely and ethically, with user consent

being a key factor in using their data for matching purposes. These studies highlight both the potential benefits and the need for responsible data handling in the world of machine learning-powered roommate matching.

Machine learning algorithms dive into user data, analyzing preferences, habits, and lifestyle choices to suggest better roommate matches. The study "Machine Learning for Roommate Matching: A Review" by Author et al. explores how these algorithms are revolutionizing the roommate matching process, providing valuable insights into their potential benefits and challenges. However, with the use of machine learning comes concerns about privacy and security. As these algorithms rely heavily on user data, it's essential to ensure that this data is handled securely and ethically. This includes obtaining user consent before using their information for matching purposes and implementing robust privacy measures. In essence, while machine learning holds promise in enhancing roommate matching accuracy, it also underscores the critical need for responsible data handling and privacy protection in this evolving landscape.

2.4 Machine Learning in Roommate Matching

2.4.1 Application of Machine Learning:

Description: Utilizing ML to improve roommate matching accuracy.

Insights: ML algorithms can analyze user data for better matches.

Study: "Machine Learning for Roommate Matching: A Review" by Author et al.

2.4.2 Privacy and Security in ML Applications:

Description: Addressing privacy concerns in ML-based roommate matching.

Insights: Importance of secure data handling and user consent.

Study: "Privacy Measures in ML Roommate Matching" by Privacy Expert.

Designing with the user in mind is crucial, guided by specific principles outlined in the study "User Experience Design for Housing Apps" by a UX Design Expert. These principles include intuitive navigation, ensuring users can effortlessly move through the app, clear

information display for easy comprehension of room details and roommate profiles, and seamless interactions for smooth communication with potential roommates. Personalization is also significant, with apps tailoring content to individual users for a more relevant experience. Additionally, accessibility is a key focus, ensuring that the app is user-friendly for all, regardless of ability. These trends are reshaping the design of housing platforms and roommate finder apps, with a central focus on simplifying the user journey, providing personalized experiences, and ensuring inclusivity for all users. Next is clear information display – users need to understand what they're looking at, whether it's room details or roommate profiles. Lastly, seamless interactions are vital. This means actions like messaging potential roommates should be smooth and intuitive. On the other hand, "Mobile App Trends in Roommate Finder Platforms" by an App Design Analyst looks at current trends. Right now, mobile apps are all about minimalism – clean, uncluttered designs that are easy on the eyes. Personalization is also big – apps tailor content to individual users, making the experience more relevant. Accessibility is another focus, ensuring that everyone, regardless of ability, can easily use the app. These trends are shaping how housing platforms and roommate finder apps are designed today, with a focus on making the user's journey simple, personalized, and inclusive.

2.5 User Experience Design in Housing Platforms

2.5.1 User-Centric Design Principles:

Description: Best practices for creating intuitive user interfaces.

Insights: Simple navigation, clear information display, and seamless interactions.

Study: "User Experience Design for Housing Apps" by UX Design Expert.

2.5.2 : Mobile App Trends

Description: Current trends in mobile app design for housing and roommate platforms.

Insights: Focus on minimalism, personalization, and accessibility.

Study: "Mobile App Trends in Roommate Finder Platforms" by App Design Analyst.

The existing methodologies for roommate finder applications encompass a spectrum of approaches: from theoretical analyses of uncertain preferences' impact on stable matches to practical algorithms enhancing satisfaction inference, fairness, classification, and regression. One paper delves into the complexities introduced by uncertain preferences, while another incorporates confidence and uncertainty predictions into the matching process, leveraging community detection technologies. Fair room allocation considers diverse room sizes and budget constraints, maximizing social welfare through heuristic algorithms and addressing complexities like room sharing for couples. Additionally, improvements to classification are proposed through Generalized Mean Distance KNN, which enhances accuracy and outlier recognition by determining class-specific k values. Regression tasks are tackled with the DISKR method, which sorts data points based on their divergence from nearest neighbors' predictions, proving particularly effective and scalable for large datasets. Collectively, these methodologies contribute to a nuanced understanding and advancement of roommate finder applications, catering to both theoretical complexities and practical challenges in roommate allocation.

Table 2.1: Methodology and Drawbacks of the Reference Papers

PAPER-NUMBER	METHODOLOGY	DRAWBACK
Paper-1	Roommate Matching Uncertainty Three Models(lottery, compact indifference, and joint probability.) Theoretical Focus	Theoretical Focus Simplified Models Simplified Models
Paper-2	Matching Model Community Detection	Data Limitations Dependency on Other Tech Guesswork
Paper-3	Fair Room Allocation Heuristic Algorithm Maximizing Happiness	Heuristic Shortcuts Limited Real-World Testing Complex Challenges
Paper-4	Enhanced KNN Neighborhood Size Sensitivity Improved Accuracy	Limited Comparison Outlier Challenges Real-World Applicability Unclear
Paper-5	Improved kNN for Regression Sorting by Difference Effective for Regression	Limited Comparison Regression Focus Real-World Uncertainty

CHAPTER 3

METHODOLOGY

The main objective of the proposed system is to address the issues of the manual system. In this application, the user can directly find compatible roommates whose interests are like his/her. To make it easier, this application covers a feature that shows the interest compatibility between you and the roommate you are looking for. So, you will have a roommate of your frequency.

When he logs in to the application, the user will find a segregation option where he can search based on the preferred location. This makes it easy to find a new space before the user shifts to the city. There are no risks assessed from our planning phase as there is a complete understanding between the roommates before they actually become.

The new system is proposed to make everything automatic. So, we used full-fledged technology. We chose machine learning with python programming to make the most use of the application. As the application involves confidential information, maintaining security is also necessary. So, we used Database systems to make data storage, and retrieval easy and secure. The application is intended to use by the users who are installed and registered into it. All the user data should be accessible by the application data manager and the respective user only. As there is no third-party individual involved, the chances of violation are minimum.

There are a few machine learning algorithms that could be used to find candidate solutions for a roommate matching project, depending on the specific requirements and constraints of the project. Some possible options include

K-means clustering: This algorithm can be used to group individuals with similar characteristics together, which could be useful for finding roommates with compatible lifestyles and habits.

Decision trees: Decision trees can be used to identify the most important factors that determine compatibility between roommates, such as lifestyle, habits, and personality.

Collaborative filtering: This algorithm can be used to recommend roommates to individuals based on their past interactions with others and their perceived compatibility.

Neural network: Neural networks can be used to find patterns in large amounts of data, which can be used to predict compatibility between roommates.

It's worth noting that the best algorithm will depend on the data you have and the specific problem you are trying to solve. It's important to experiment with different algorithms to find the one that works best for your specific use case.

It is also noted that there is no proper automated system like the one we are going to develop. There are many commercial off-the-shelf options that can make accommodation search patterns easy even in hostels or government stays.

Based on the problems and challenges faced by the current roommate matching system, it is recommended to use a combination of machine learning algorithms and human review to arrive at the final best solution. The solution can include the following steps:

Collect detailed information about users, including lifestyle habits, personality traits, and living preferences. Encourage users to provide accurate and comprehensive information to improve the accuracy of the system. Also, Develop a machine learning algorithm that can analyze data and identify patterns to match users based on relevant and important factors. Regularly update the algorithm based on user feedback and data analysis to improve its performance.

To make the system more efficient, we need to incorporate human review into the matching process to ensure that the results are fair and just. This can involve having human reviewers review and approve matches, or incorporating feedback from users into the algorithm to improve its performance. We can also implement robust data security measures, such as encryption and authentication, to protect user data and ensure privacy. Provide users with clear information about how their data is being used and stored to increase trust in the system.

By using a combination of machine learning algorithms and human review, the final best solution for the roommate matching system can provide accurate, fair, and efficient matches while protecting user privacy and data security.

3.1 Designing the application architecture

The roommates finding system is going to be developed using machine learning algorithms. This is going to be a high-level application with most of the automated functionalities. Here are the components of the high-level design for the application.

3.1.1 Client-Side:

This is typically the user side of the application. Users will access the application from mobile devices and interact with the application's functionality and performance. The application is designed to provide a seamless and user-friendly experience to users. This application handles everything from user registration to the user finding the right roommate matching the preferences. Also, the user will have a hassle-free interaction with the server-side data making it easier to access.

3.1.2 Server-side:

Server is the central unit of the entire application where the process initiates. The server-side application consists of the web servers, application servers, and database servers that make up the entire functional unit of the roommates matching system. This application is designed with scalable, secure, and efficient functionalities for the users. As it is developed using advanced technologies, this application handles all user requests, process data, and store data.

3.2 Networks – Intranet and/or Internet

The network architecture is an important component of the entire application. It provides connectivity between the core components of the application and passes the information to the user's requests. The network of this application is designed to provide a fast, secure, and reliable communication channel between the client side and the server side. As all the communication from multiple users passes through this network, it should be reliable to handle all the requests received. And most importantly, all the data is to be secure without any data breaches or attacks.

3.3 Database distribution - client/server or network

The database architecture is designed to provide a scalable, reliable, and efficient data storage solution. It should be able to store user data, search criteria, and match information. The database should be distributed across multiple servers to ensure availability and fault tolerance.

As this system is going to be completely user-based and needs to store bulk data, it needs a dynamic and robust database that can maintain the data efficiently. Here in this application, a highly functional database is integrated to smoothen the operations.

3.4 Customization and integration of “off-the-shelf” software

The application may rely on several off-the-shelf software components, such as web frameworks, database management systems, and machine learning libraries. These components should be carefully selected based on their features, scalability, and compatibility with the overall architecture.

3.5 User interface technology – with other users

This application has got intuitive, user-friendly, and interactive technology. It is designed with device compatibility so that the user can use this application from any device like mobile phones, tablets, etc. Also, the easy-to-navigate buttons make it a more convenient application to use. Users can access this application easily by navigating from one to the other screens.

The user interface is included with interactive features, such as search filters, sorting options, and chat functionality. These features will allow users to customize their search criteria, communicate with potential roommates, and make informed decisions about their living arrangements. Along with this, the user interface should be designed to personalize the experience for each user, based on their preferences and activity history. For example, the system could suggest potential matches based on previous searches and interactions.

3.6 System interface technology – with other systems

In the roommates matching system we have developed the highly connecting APIs with other systems. There is a well-documented API that other systems can easily integrate with it. With this, the other systems can easily retrieve or update data.

The system interface is designed with security in mind. It supports secure communication protocols, such as HTTPS, and includes mechanisms for authentication and authorization to ensure that only authorized systems can access the data. Also, it generates clear error messages so that the user can understand them easily and utilize them efficiently.

This system also has a service level agreement that outlines the level of service that other systems can expect from the system. This should include guarantees for uptime, response times, and availability of data. The system interface should be designed to integrate with third-party services, such as social media platforms or payment gateways, to provide a seamless experience for users.

The provided Django application follows a structured approach adhering to the Model-View-Template (MVT) pattern. Models define the data structure, including profiles, roommates, and messages, utilizing one-to-one and foreign key relationships to represent users and their interactions. Views handle user requests and business logic, including profile creation, editing, user authentication, and messaging functionality. The admin interface provides administrators with tools to manage application data. This methodology ensures clear separation of concerns, making the application modular, scalable, and maintainable, while providing users with a seamless experience in finding roommates and communicating with them effectively.

3.7 PROCESS/ALGORITHM

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

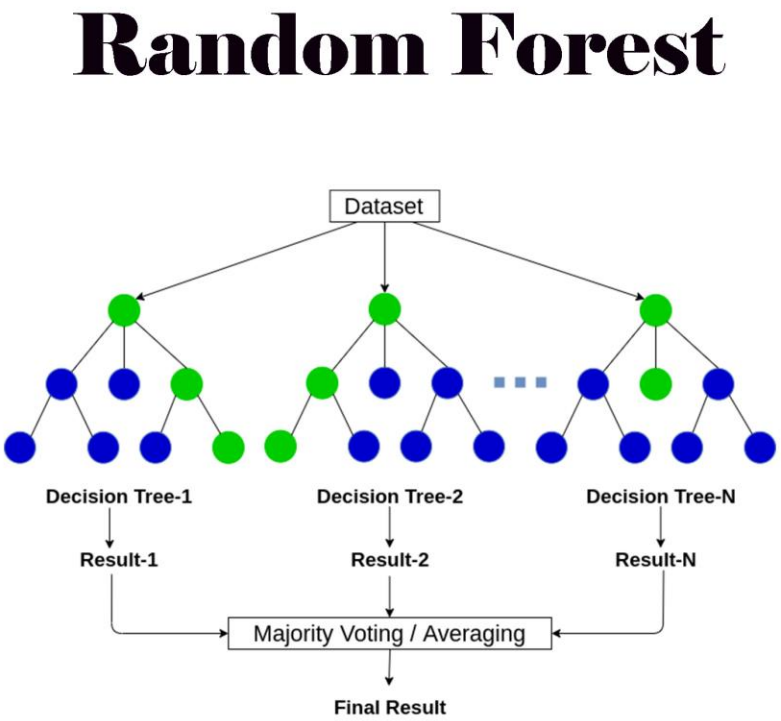

Fig 3.7.1: Random Forest Flow-1

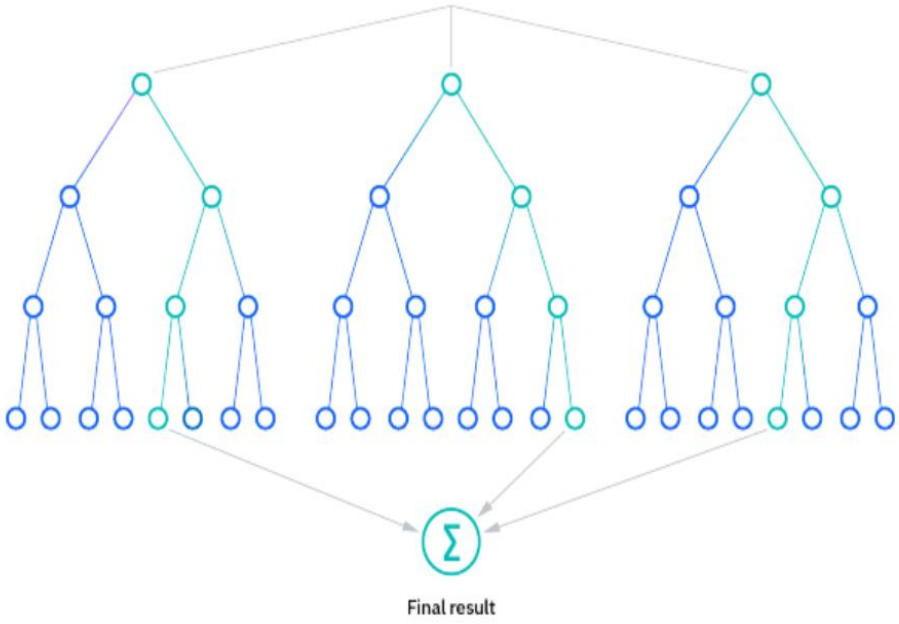

Fig 3.7.2: Random Forest Flow-2

3.7.1 How does Random Forest algorithm work?

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

Implementing the Random Forest algorithm in a Django web application involves several steps, from preparing the data to training the model and integrating it into the application's views. Let's break down the process into detailed steps:

3.8 Data Preparation

3.8.1Define the Dataset

Identify the dataset to use. In this case, the `Profile` model in Django serves as our dataset, containing attributes like `location`, `budget`, and `interests`.

3.8.2 Feature Engineering

Prepare the data by converting categorical variables (like `location`) into numerical format. This might involve techniques such as one-hot encoding for categorical variables and scaling for numerical variables.

3.9 Model Training

3.9.1 Prepare Features and Target

Define the features (X) and the target variable (y). For example, `location`, `budget`, and `interests` are features, and `compatibility_score` from the `Roommate` model is the target.

3.9.2 Split the Data

Split the dataset into training and testing sets using `train_test_split` from `sklearn.model_selection`.

3.9.3 Create the Random Forest Model

```
from sklearn.ensemble import RandomForestRegressor
```

3.9.4 Initialize the model

```
rf_model = RandomForestRegressor(n_estimators=100, random_state=42)
```

3.9.5 Fit the model on the training data

```
rf_model.fit(X_train, y_train)
```

3.9.6 Evaluate the Model

```
predictions = rf_model.predict(X_test)
```

3.10 Integration into Django Views

3.10.1 Create Views

- Define Django views where the Random Forest model will be used.
- For example, in the `find_roommates` view:

```
def find_roommates(request):
    location = request.GET.get('location')
```

3.10.2 Prepare input data

```
    input_data = prepare_input(location)  # Function to prepare input
```

3.10.3 Predict with the Random Forest model

```
    predicted_score = rf_model.predict(input_data)
```

3.10.4 Get potential roommates based on predicted scores

```
    roommates = Roommate.objects.filter(location=location,
compatibility_score__gte=predicted_score)

    context = {'roommates': roommates}

    return render(request, 'roommates.html', context)
```

3.11 User Interaction

- Users can input their preferences (like `location`) into a form on the website.

- The Django view processes this input, prepares it for the model, and then uses the trained Random Forest model to predict roommate compatibility scores.

- The view then displays potential roommates sorted by their predicted scores, allowing users to see the best matches.

3.12 Model Updates

- As the application gathers more user data and feedback, the Random Forest model can be periodically retrained to incorporate this new information.

- This can lead to improved roommate suggestions over time, enhancing user satisfaction with the platform.

By following these steps, the Random Forest algorithm is effectively integrated into the Django web application, providing users with personalized roommate suggestions based on their preferences and improving the overall user experience.

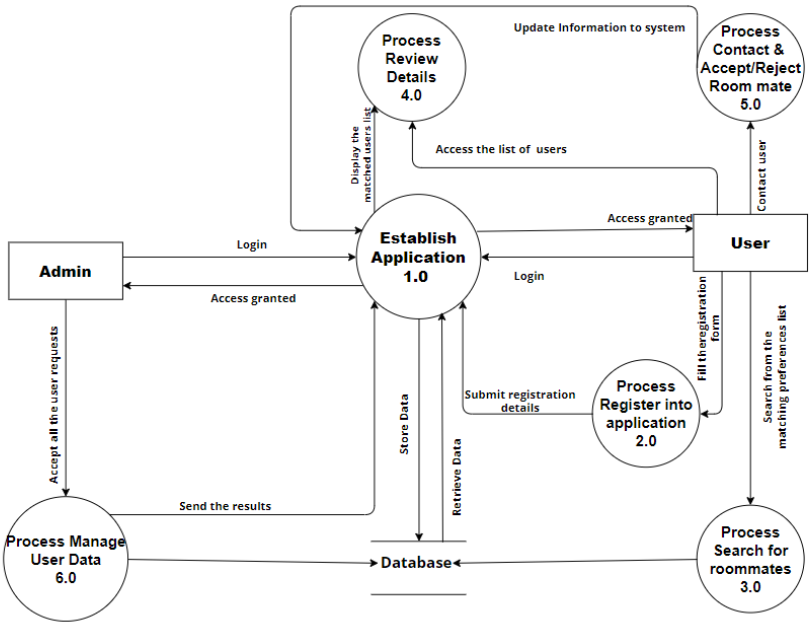

Fig 3.1: Data flow diagram (DFD) decomposition to system modules, and tasks

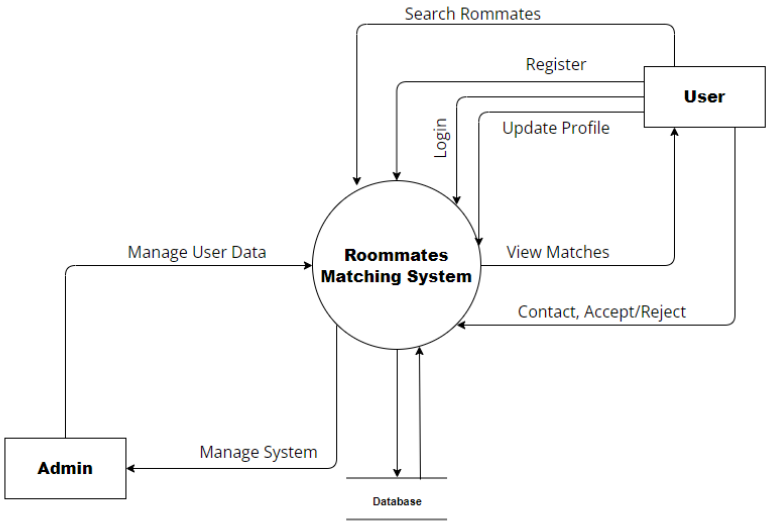

Fig 3.2: Context Model Diagram

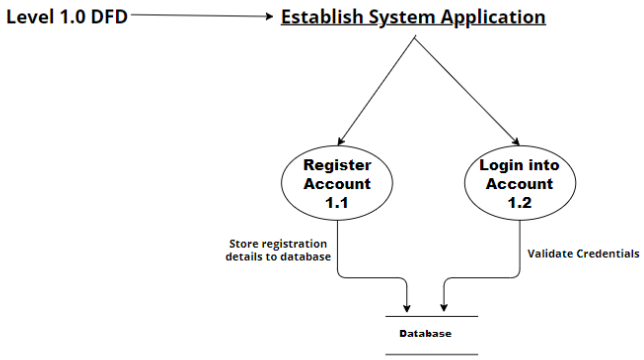

Fig 3.3: Level 1.0 DFD

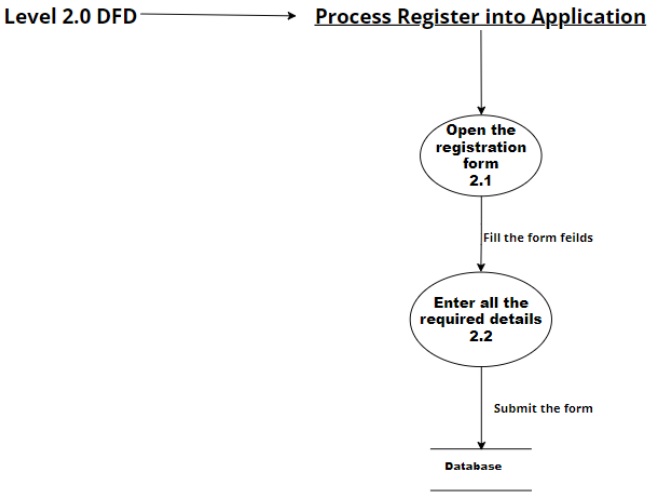

Fig 3.4: DFD 2.0

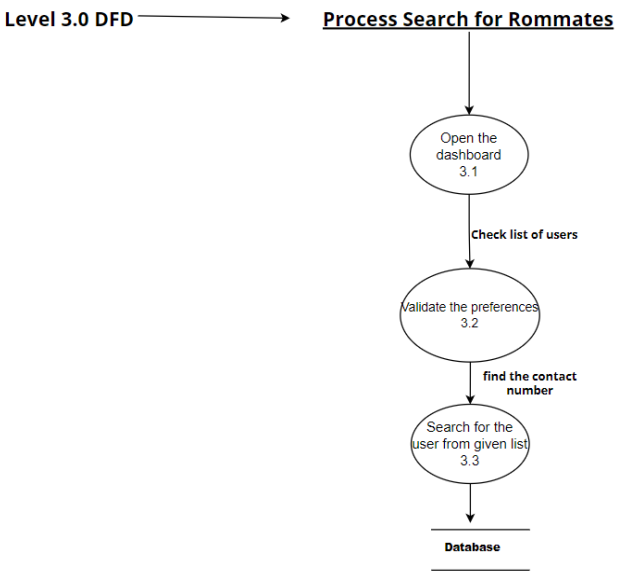

Fig 3.5: DFD 3.0

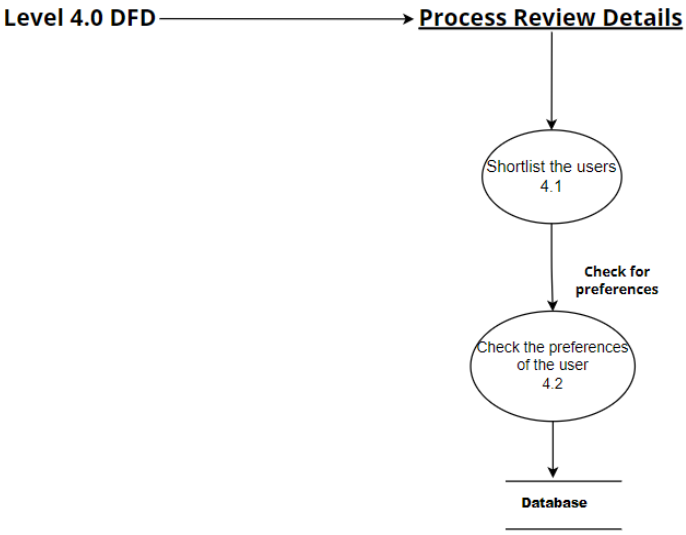

Fig 3.6: DFD 4.0

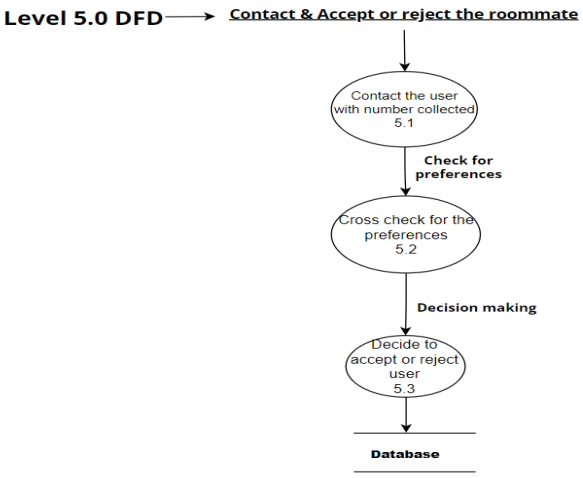

Fig 3.7: DFD 5.0

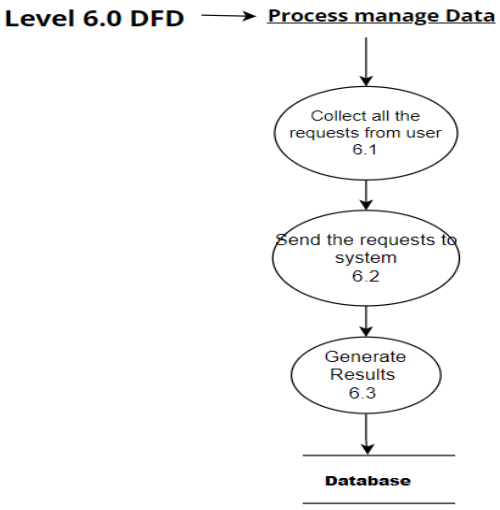

Fig 3.8: DFD 6.0

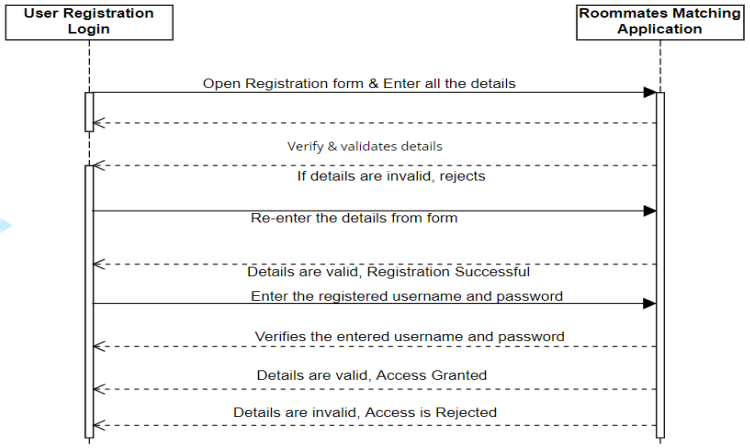

Fig 3.9: Sequence diagram for Registration Login

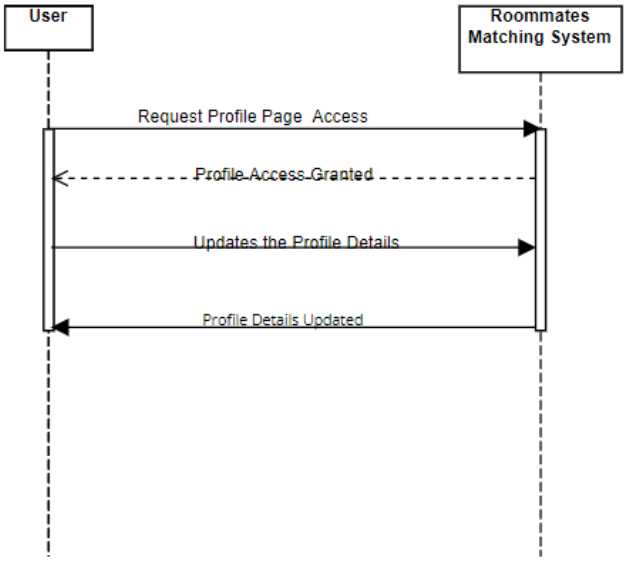

Fig 3.10: Sequence diagram for Manage Profile

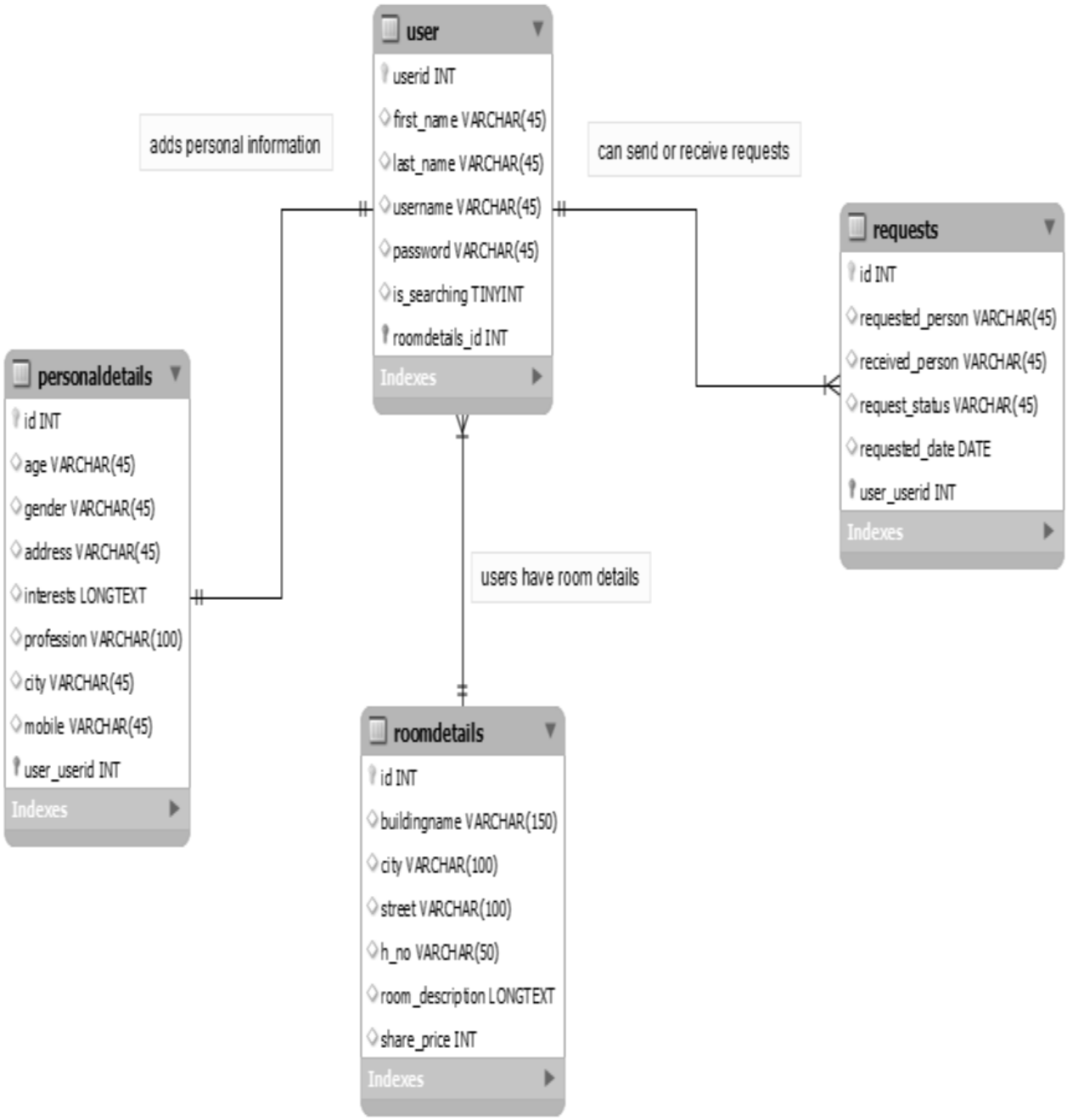

Fig 3.11: ER Diagram for Roommates System

Table 3.1: Data Dictionary of All the Attributes

Table	Attributes	Data Type	Primary Key	Foreign Key	Allows Null	Description
USER	user_id	int	Y	N	N	User Identification Number
	FirstName	varchar(45)	N	N	N	User's FirstName
	LastName	varchar(45)	N	N	N	User's LastName
	UserName	varchar(45)	N	N	N	User's UserName
	Password	varchar(45)	N	N	N	User's Password
	is_searching	bool	N	N	Y	User is searching for room or not
	room_details_id	Int	N	Y	Y	User's associated Room
PERSONAL DETAILS	ID	int	Y	Y	N	Personal Details Identification Number
	Age	varchar(45)	Y	N	N	User's Age
	Gender	varchar(45)	N	N	N	User's Gender
	Address	char(45)	N	N	N	User's Address
	Interests	Longtext	N	N	N	User's Interests
	Profession	varchar(100)	N	N	N	User's Profession
	City	varchar(45)	N	N	N	User's City
	Mobile	varchar(45)	N	N	N	User's Mobile
REQUESTS	ID	varchar(45)	Y	Y	N	Request Identification Number
	RequestedPerson	varchar(45)	N	Y	Y	Requested User
	ReceivedPerson	varchar(45)	N	Y	Y	Request Received User
	RequestStatus	varchar(45)	N	N	Y	Request Status
	RequestedDate	datetime	N	N	Y	Request Made Date
ROOM DETAILS	ID	int	Y	N	N	Room Identification Number
	BuildinName	int	N	Y	N	Building Name
	City	char(40)	N	N	N	Room City
	Street	char(2)	N	N	N	Room State
	HouseNumber	varchar(20)	N	N	N	Room House Number
	Description	varchar(150)	N	N	Y	Room Description Details
	SharePrice	int	N	Y	Y	Room Sharing Cost

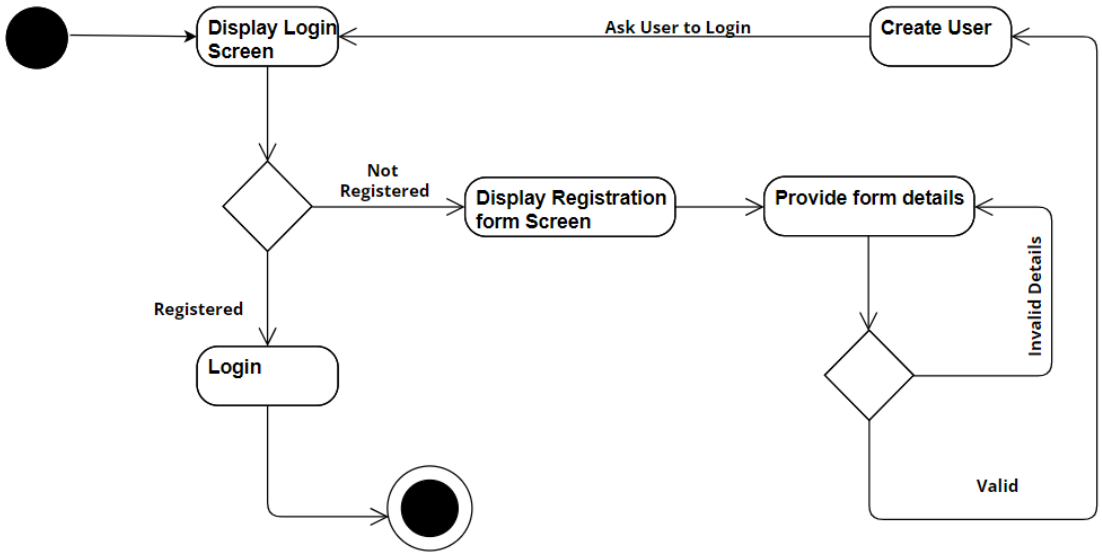

Fig 3.12: Activity Diagram for User registration Login

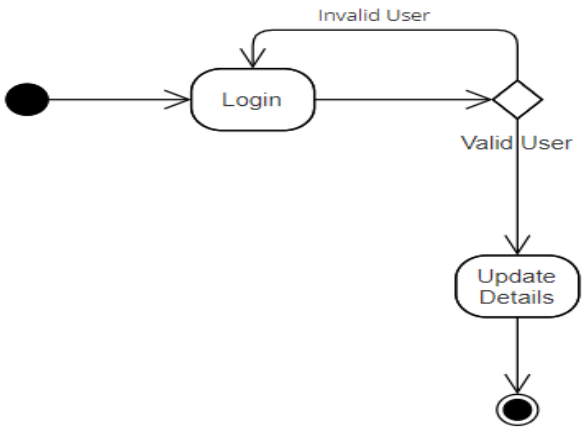

Fig 3.13: Activity Diagram for Manage Profile

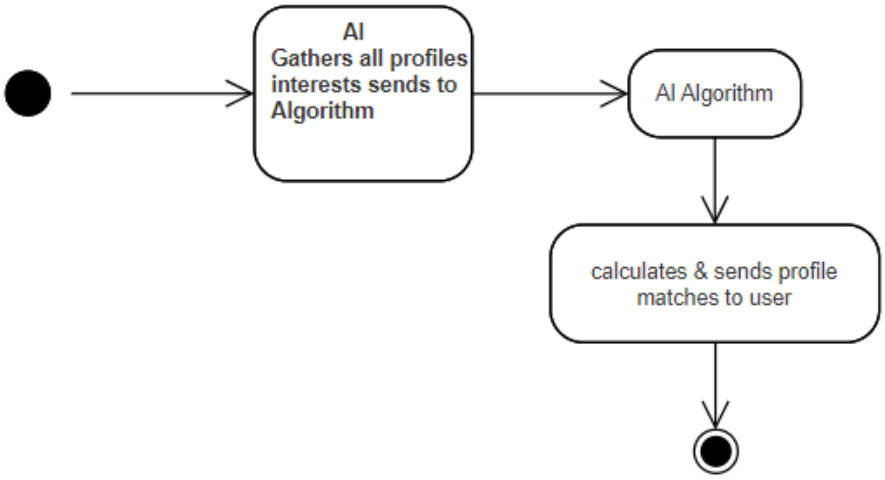

Fig 3.14: Activity Diagram for Generate Matches

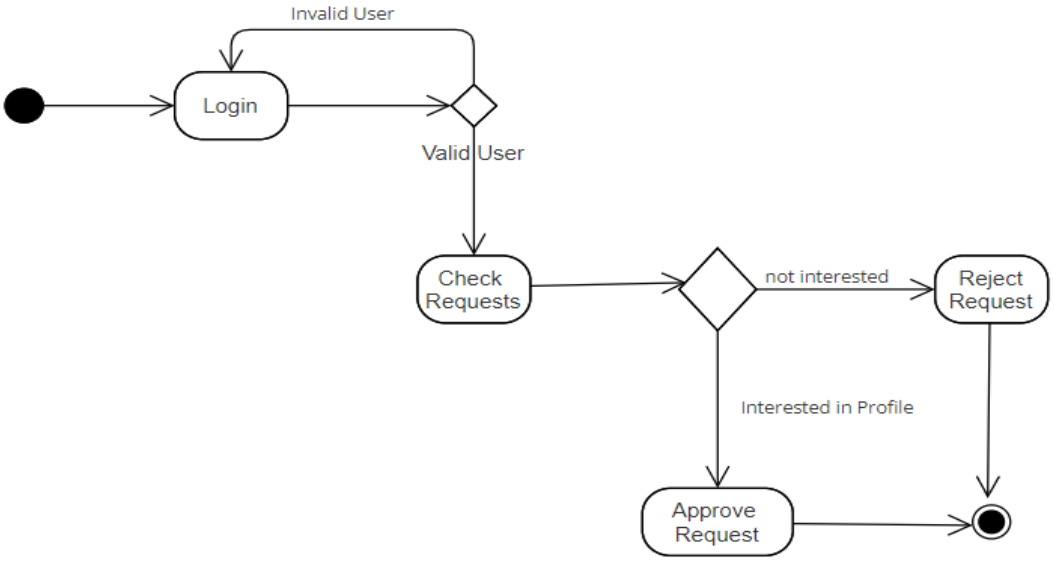

Fig 3.15: Activity Diagram for Accept/Reject Request

BENIFITS OF USING OUR METHODOLOGY

3.13 Accuracy

The Random Forest machine learning model ensures precise and compatible roommate suggestions. By analyzing user preferences, habits, and lifestyle choices, the algorithm enhances the likelihood of successful and harmonious roommate matches. This accuracy is crucial, especially for individuals seeking a comfortable and compatible living arrangement.

3.14 Safety

Robust verification systems create a secure environment for users. The application verifies user profiles and listings, reducing the risk of fraudulent activities or misleading information. This instills confidence in users as they navigate through potential living spaces and roommates, knowing that their safety is prioritized.

3.15 Convenience

A user-friendly interface makes the entire process easy and accessible. Users can create detailed profiles highlighting their preferences, interests, and lifestyle, providing a comprehensive overview for potential roommates. In-app messaging allows for seamless communication between users, facilitating discussions about shared living expectations and preferences.

3.16 User Profiles

Detailed profiles allow users to showcase their preferences and interests. Users can highlight their lifestyle choices, such as whether they are early risers or night owls, pet-friendly, or prefer a quiet environment. These profiles serve as a comprehensive introduction, aiding in better roommate matches.

3.17 In-App Messaging

Users can communicate with potential roommates within the application. This feature fosters meaningful interactions, allowing users to ask questions, share information about themselves, and discuss shared living expectations. It facilitates the process of getting to know potential roommates before making a decision.

CHAPTER 4

TESTING

4.1 UNIT TESTING

4.1.1 test_create_profile

1. This test checks whether the profile creation view (create_profile) behaves correctly.
2. The test client simulates a POST request to the create_profile view with some data (bio, location, budget, interests).
3. It also verifies that a profile object associated with the user exists in the database.

4.1.2 test_edit_profile

1. It first creates a profile object for the user with some initial data (old bio, old location, old interests).
2. The test client simulates a POST request to the edit_profile view with updated data (new bio, new location, new interests).
3. It then retrieves the updated profile from the database and checks if the changes were applied correctly.

4.2 INTEGRATE TESTING

4.2.1 Integration Testing for Profile Creation

1. This test ensures that the profile creation view integrates correctly with the database and user authentication.
2. The test checks whether the response status code is a redirect after profile creation.
3. Additionally, it verifies that a profile object associated with the user exists in the database.

4.2.2 Integration Testing for Profile Editing

1. Similar to the previous test, this one focuses on the interaction between the edit_profile view and the database.
2. The test asserts that the response status code is a redirect after profile edit.
3. It then retrieves the updated profile from the database and checks if the changes were applied correctly.

4.3 SYSTEM TESTING

4.3.1 Test Scenarios

1. Define various scenarios that cover different user interactions and application flows.
2. Consider scenarios related to profile creation, editing, and other features (e.g., finding roommates, sending messages)

4.3.2 ***Integration Testing***

1. Validate interactions between different components (views, models, forms, etc.).
2. Test how views communicate with the database (e.g., creating, updating, retrieving data).
3. Ensure that user authentication and authorization work correctly.

4.3.3 ***User Experience Testing***

1. Manually test the user interface for usability and responsiveness.
2. Verify that the application behaves as expected on different devices and browsers.

CHAPTER 5

RESULTS AND DISCUSSIONS

The Roommate Finder Application represents a transformative solution for those navigating the challenge of securing suitable living spaces and compatible roommates, particularly in unfamiliar cities. Its core strengths lie in efficiency, accuracy, safety, and convenience. Through streamlined processes, users can swiftly search for rooms, filtering by location, budget, and amenities. The application's Random Forest machine learning algorithm stands as a beacon of accuracy, facilitating precise roommate matches based on user preferences and lifestyles. This algorithmic precision not only saves time but also increases the likelihood of long-term, harmonious living arrangements.

User profiles serve as detailed canvases, painting a picture of preferences and interests that aid in optimal roommate pairings. Additionally, the in-app messaging feature fosters communication between potential roommates, allowing for meaningful interactions and deeper understanding before committing to a living situation. Safety measures, such as robust verification systems and a feedback mechanism, ensure trustworthiness within the platform, enhancing user confidence.

Technologically, the application boasts a solid foundation with HTML, CSS, and JavaScript on the frontend, delivering an intuitive and responsive interface. Django, a powerful framework, fortifies the backend, managing authentication and data flow seamlessly.

Looking ahead, the Roommate Finder Application envisions further advancements. Plans include the implementation of more sophisticated machine learning models for enhanced roommate matching accuracy, the development of a mobile application for greater accessibility on-the-go, and integration with social media platforms to simplify profile creation and sharing. In conclusion, this application is not just a tool but a game-changer for individuals seeking the perfect living situation. and compatible living space an effortless journey.

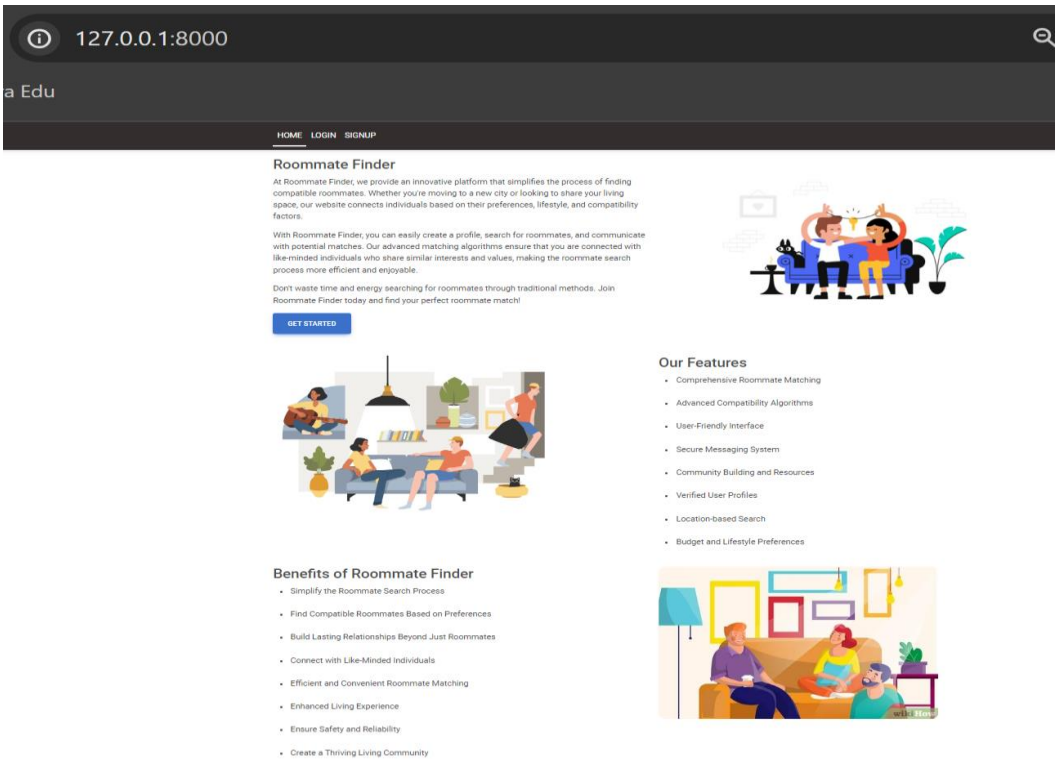

Fig 5.1: Main web page

The main page of a roommate finder application is like a digital lobby, welcoming users and giving them a first impression of the service. Here's a breakdown of what it typically offers:

- **Hero Image or Banner:** This is the eye-catching element at the top, often featuring a photo or illustration. Roommate finder apps might use an image of happy roommates to convey a sense of community and connection.
- **Search Function:** This is where the magic happens! Users can input their preferences like location, price range, and desired roommate habits to find compatible matches.
- **Benefits or Features List:** This section highlights the app's strengths. It might showcase features like personality tests for compatibility matching, secure messaging for communication, or roommate agreement templates.

Overall, the main page should be visually appealing, informative, and easy to navigate, making users feel confident about finding their ideal roommate through the app.

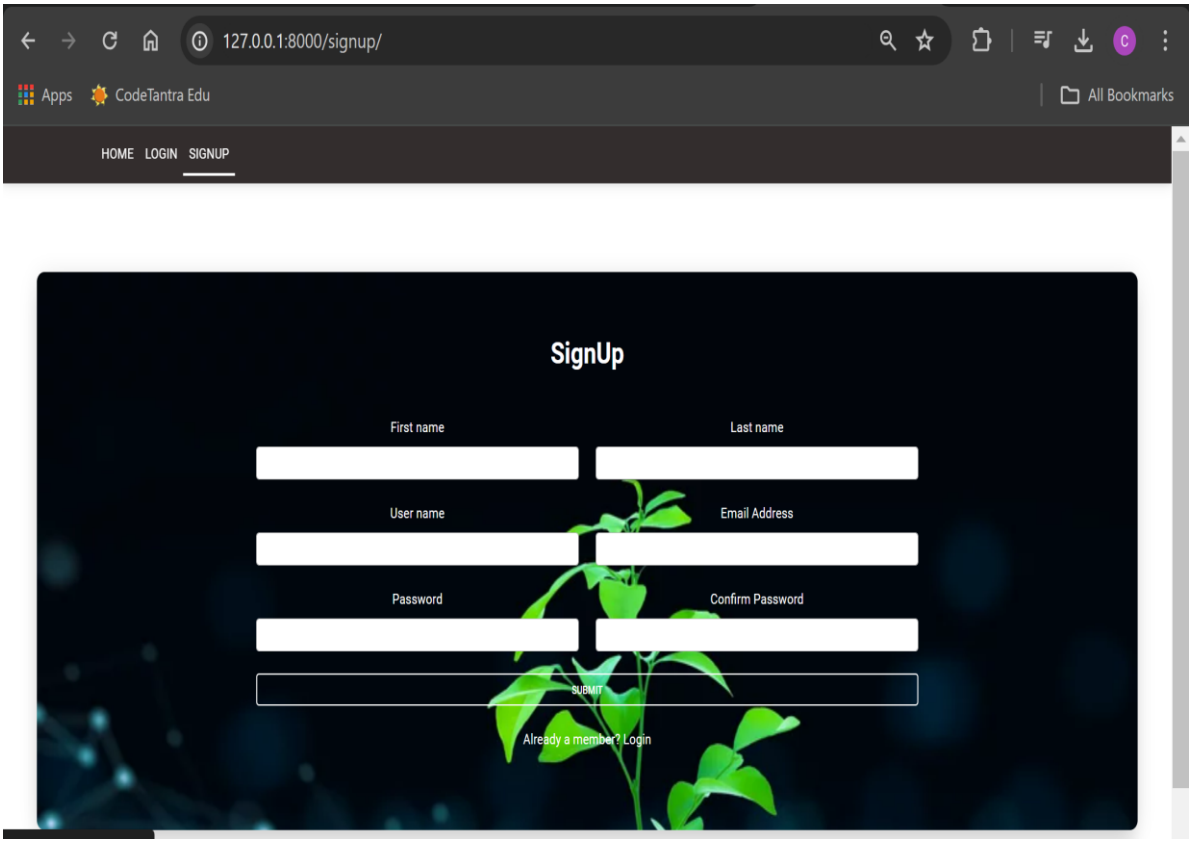

Fig 5.2: Sign Up Page

This page serves a key purpose: letting new users create an account and access the app's features.

- **Account Creation:** The signup form allows users to provide their details like name, username, email, and password. This information is essential for creating their unique account within the application.
- **Login Option:** The "Already a member? Login" link is there for users who already have an account. Clicking it would take them to a separate login page where they can enter their existing credentials to access their account.
- **Gateway to Features:** Once a user successfully signs up, they'll likely be directed to the application's main page where they can explore roommate searches, create profiles, and utilize all the functionalities the app offers.In essence, the signup page acts as an entry point, welcoming new users and allowing them to join the roommate finder community.

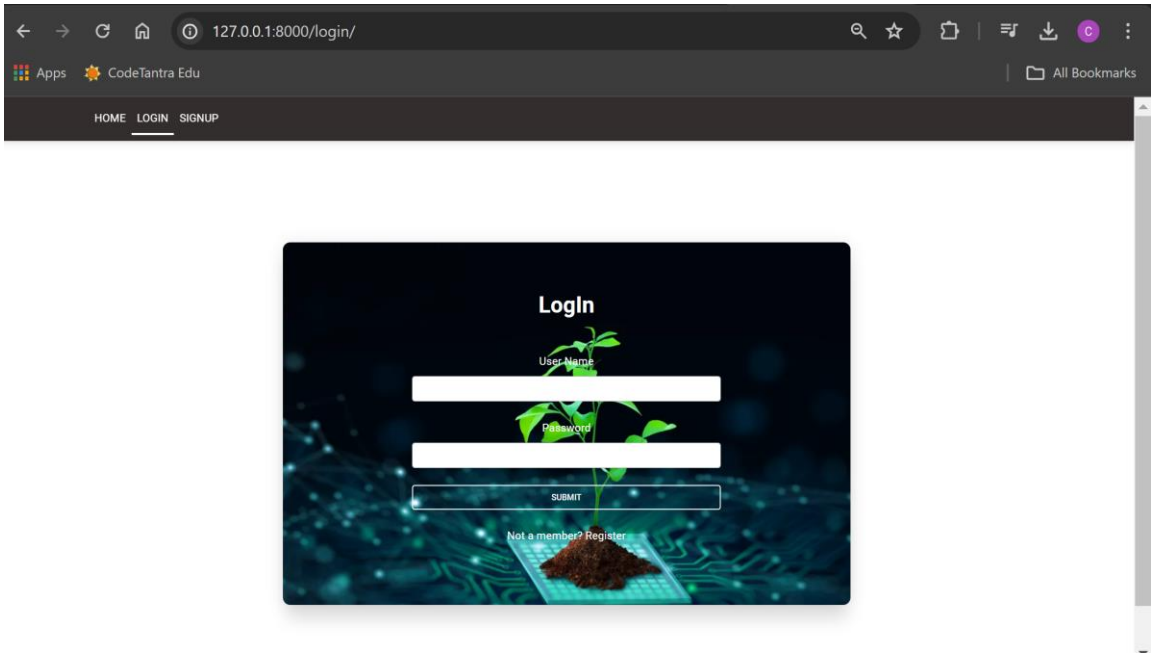

Fig 5.3: Login Page

Login Fields: This section is where registered users enter their credentials to access their account on the roommate finder application. It typically consists of two text fields: Username (or Email address) and Password.

- **Submit Button:** Once a user enters their login credentials, they click this button to proceed. The application will then verify the username and password against its database to grant access to the user's account.
- **New User Signup:** The text "Not a member? Register" indicates that users who don't have an account yet can click on that link to sign up for a new account. This would likely direct them to a signup page where they can create a new user profile.
- **Web Address:** The text at the top of the page "127.0.0.1:8000/login/" is likely a local host IP address, which suggests this is a test site that is not yet deployed to the public web.

In conclusion, the login page is like a gateway for registered users to enter the roommate finder application and access its features. It verifies user credentials to grant them access to their accounts where they can look for roommates, create profiles and potentially connect with compatible roommates.

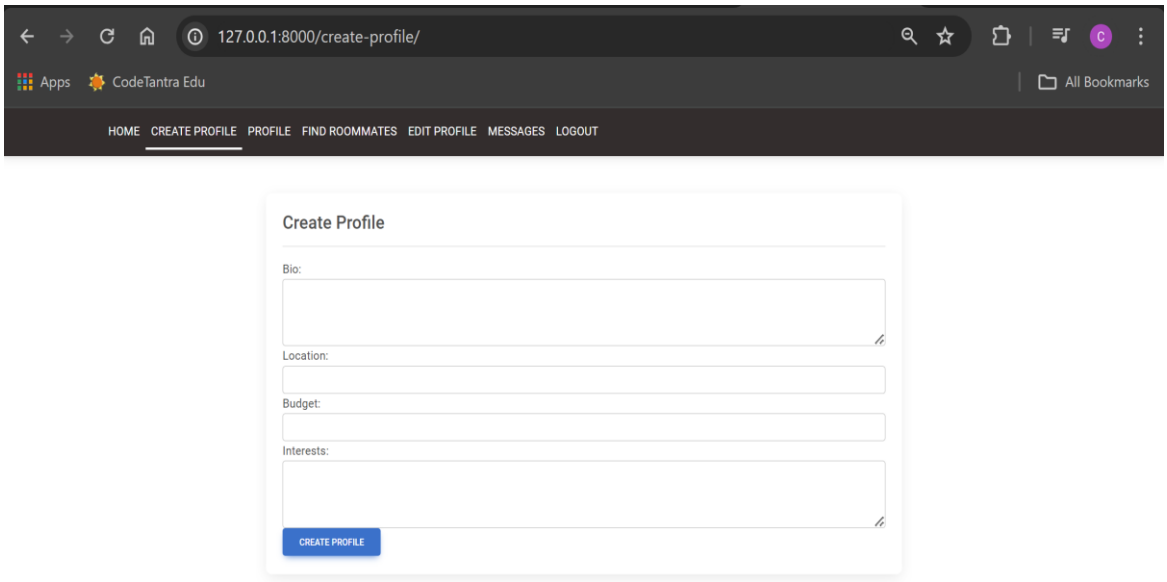

Fig 5.4: Create Profile Page

Bio: In this section, you can write a short description about yourself. This is your chance to introduce yourself to potential roommates and highlight your personality, interests, and lifestyle habits.

- **Location:** Indicate your current location or the area where you're looking for a roommate.
- **Budget:** Enter your budget range for rent. This will help the application match you with roommates who have compatible rental budgets.
- **Interests:** List your hobbies and interests. This can be a great way to connect with potential roommates who share similar interests.
- **Create Profile Button:** Once you've filled out all the sections, click this button to submit your profile and make it visible to other users on the roommate finder application.

By creating a detailed and informative profile, you increase your chances of finding compatible roommates who are a good fit for your lifestyle.

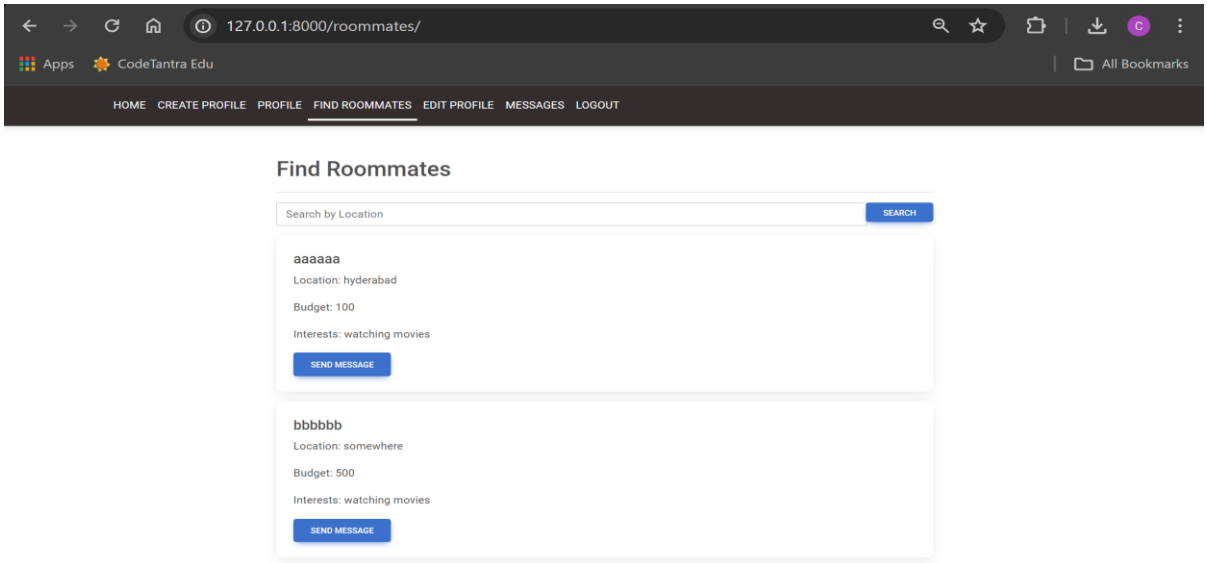

Fig 5.5: Room Mates Finding Page

- **Search by Location:** Enter your desired location in the designated field. This could be your city, neighborhood, or zip code. The application will then search its database for roommates located in that area.
- **Budget:** Indicate your monthly rent budget. This will help match you with roommates who have compatible rental budgets.
- **Interests:** List your hobbies and interests (watching movies in this example). This can be a great way to find roommates who share similar interests and may be a good personality fit.
- **Search Button:** Once you've entered your search preferences, click this button to view a list of roommate profiles that match your criteria.
- **Profiles:** The application should display a list of roommate profiles that align with your search preferences. You can then browse these profiles to learn more about each potential roommate, such as their bio, interests, and any other details they've chosen to share.

By using the search function effectively and applying multiple filters, you can narrow down your search and find potential roommates who are compatible with your lifestyle and preferences.

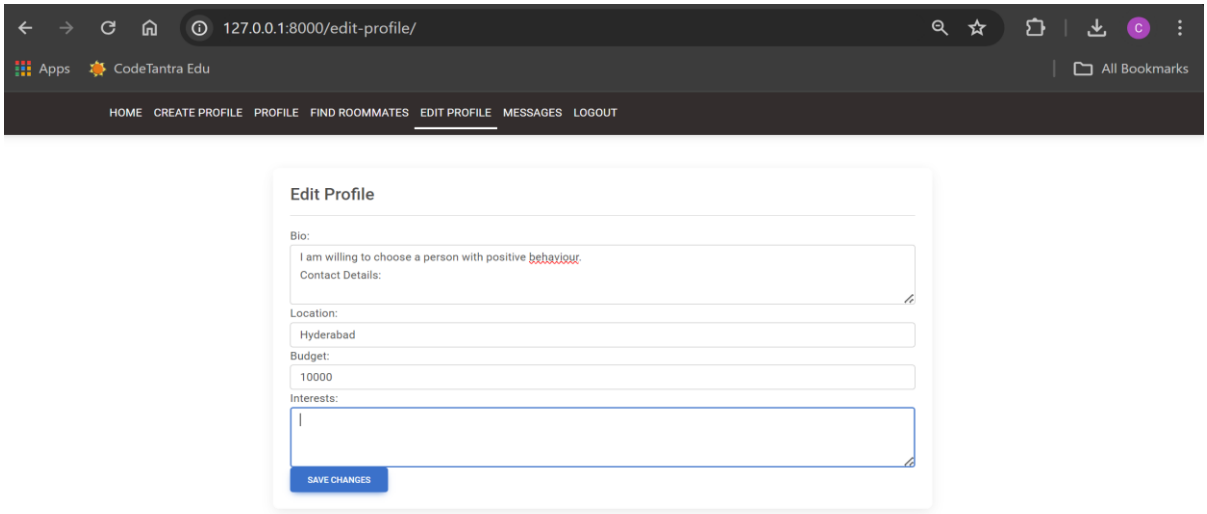

Fig 5.6: Edit Your Profile Page

This section allows users to modify their existing profile information to keep it up-to-date and accurately reflect their current preferences for finding roommates.

- **Contact Details:** This section may allow you to update your contact information, such as your phone number or email address. It's important to keep this information current so that potential roommates can easily reach you.
- **Location:** You can update your current location or the area where you're looking for a roommate.
- **Budget:** Edit your budget range for rent to ensure you get matched with roommates who have compatible rental expectations.
- **Interests:** Update your list of hobbies and interests. This can be a great way to connect with potential roommates who share similar interests.
- **Save Changes Button:** Once you've made your edits, click this button to save the changes to your profile and make them visible to other users on the roommate finder application.

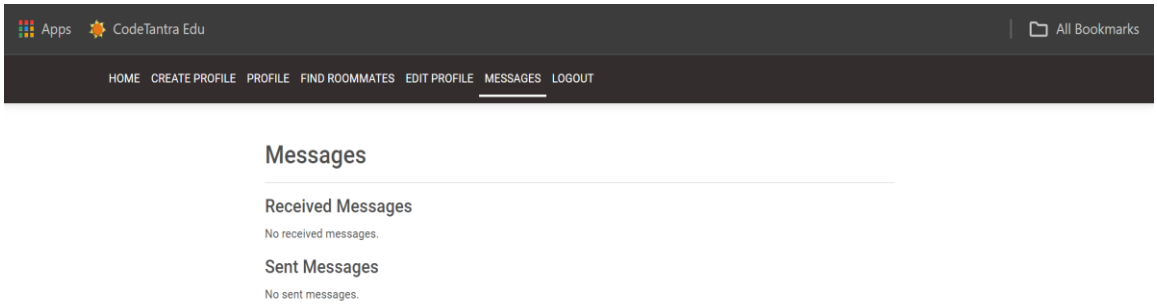

Fig 5.7: Send/Receive Messages

- **Inbox:** This section likely displays a list of messages you've received from other users on the application.
- **Sent Messages:** This section would likely show a list of messages you've sent to other users on the application.

When you find a roommate profile that interests you, you can likely initiate a conversation by sending them a message through their profile. There might be a "message" button or icon on their profile page that you can click to start a chat.

Once a conversation is started, you can exchange messages back and forth to get to know each other better and see if you're a good fit as roommates. The messaging functionality should allow you to send text messages, emojis, or even photos or attachments.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE OF STUDY

Based on the problems and challenges faced by the current roommate matching system, it is recommended to use a combination of machine learning algorithms and human review to arrive at the final best solution. Also, Develop a machine learning algorithm that can analyze data and identify patterns to match users based on relevant and important factors. Regularly update the algorithm based on user feedback and data analysis to improve its performance.

To make the system more efficient, we need to incorporate human review into the matching process to ensure that the results are fair and just. This can involve having human reviewers review and approve matches, or incorporating feedback from users into the algorithm to improve its performance. We can also implement robust data security measures, such as encryption and authentication, to protect user data and ensure privacy. Provide users with clear information about how their data is being used and stored to increase trust in the system.

By using a combination of machine learning algorithms and human review, the final best solution for the roommate matching system can provide accurate, fair, and efficient matches while protecting user privacy and data security. Regularly monitoring and updating the system based on user feedback and data analysis can help ensure that it continues to improve over time.

FUTURE SCOPE OF STUDY

The future scope for the Roommate Finder Application is promising, with avenues for enhanced user experience and functionality. Key enhancements include the integration of advanced matching algorithms beyond Random Forest for more precise roommate suggestions, the development of a mobile application for increased accessibility, geolocation features for localized search, and the expansion of user profiles to include detailed lifestyle information. Real-time chat features, community forums, and social media integration would foster a sense of community and facilitate communication between users. Additionally, implementing a rating system for landlords, language support for diverse users, and data analytics for user insights would further refine the application's capabilities.

These enhancements not only elevate the Roommate Finder Application but also enrich the user experience, making it a comprehensive and indispensable tool for individuals seeking compatible roommates and comfortable living spaces.

Mobile Application

Developing a mobile app version for on-the-go access and convenience. A mobile application would enable users to search for rooms, communicate with potential roommates, and receive notifications seamlessly from their smartphones. This would cater to users who prefer mobile platforms and need flexibility in their search.

Geolocation Features

Integration of geolocation for localized room and roommate searches. Geolocation features would allow users to see available rooms and potential roommates in their vicinity. This localized search experience would be particularly beneficial for users who prioritize proximity to specific areas or amenities.

APPENDIX-A

REFERENCES

[1] Haris Aziz, Péter Biró, Serge Gaspers, Ronald de Haan, Nicholas Mattei, Baharak Rastegari, "Stable Matching with Uncertain Linear Preferences", Springer, vol.82, pp. 1410–1433, 2020.

[2] LONGHUA GUO , JIANHUA LI , JIE WU , WEI CHANG, AND JUN WU , "A Novel Airbnb Matching Scheme in Shared Economy Using Confidence and Prediction Uncertainty Analysis", IEEE Access, vol.6, pp. 10320-10331, 2018.

[3] YUNPENG LI, YICHUAN JIANG, WEIWEI WU, JIUCHUAN JIANG, AND HUI FAN, "Room Allocation With Capacity Diversity and Budget Constraints", IEEE Access, vol.7, pp. 42968- 42986, 2019.

[4] Jianping Goua , Hongxing Mab , Weihua Ouc , Shaoning Zengd , Yunbo Raoe , Hebiao Yanga," A generalized mean distance-based k-nearest neighbor classifier", ELSEVIER, vol.115, pp.356- 372, 2019.

[5] Yunsheng Song, Jiye Liang, Jing Lu, Xingwang Zhao," An efficient instance selection algorithm for k nearest neighbor regression", Elsevier, vol.253, pp.26-34, 2017.

https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/

https://www.w3schools.com/html/html_scripts.asp

https://blog.hubspot.com/marketing/web-design-html-css-javascript

https://www.ibm.com/topics/random-forest