CHAPTER ONE
INTRODUCTION

1.1 Introduction

In today's digital era, where online dating has emerged as a popular means for individuals to meet and connect with potential romantic partners, we are thrilled to introduce our project: an engaging and innovative online dating platform specifically designed for the modern single. Our ambition is to offer a user-friendly, efficient, and enjoyable platform that enables individuals to form connections based on shared interests and preferences.
A good feature of our platform is its secure and seamless messaging system. Recognizing the importance of trust in the realm of online dating, we have devoted substantial effort to developing a messaging system that prioritizes user privacy and security. This system is built with advanced encryption technologies, ensuring that all conversations remain private and protected. Users can confidently engage in personal discussions, knowing that their data and conversations are safeguarded against unauthorized access and breaches.

Moreover, our seamless messaging system is designed for ease of use, allowing for smooth and uninterrupted communication. This hassle-free interface enhances the user experience, encouraging more profound and sustained interactions. In addition to text messaging, our system includes real-time messaging with encryption that is achieved using XOR encryption and the Diffie-Hellman Key Exchange algorithm, providing a rich and versatile communication experience. Beyond the technical aspects, we are committed to a safe and respectful online dating environment. Our platform includes robust reporting and blocking features, enabling users to block unwanted contacts.

Our project is not merely about creating an online dating platform; it's about crafting a space where meaningful relationships can flourish. By blending innovation, user-centric design, and a deep commitment to privacy and security, we aim to redefine the online dating experience. We're excited to use cutting-edge technology to help people connect, and we're confident that with our dedication and focus on user needs, we'll create an online dating platform that exceeds expectations in today's digital age.

1.2 Problem Statement

The current challenge lies in the shortcomings of existing online dating platforms, which mostly operate on a price-driven model with limited features for basic subscriptions. The constraints of these platforms result in frustration and wasted time for users seeking meaningful connections. Another challenge in online dating is the ineffective user filtering mechanisms on existing platforms. Often, these filters are too basic or not nuanced enough, leading to users being overwhelmed with matches that don't align with their specific preferences and criteria.

The problem at hand requires the creation of an innovative online dating platform that addresses these limitations, providing a seamless and secure experience. Our goal is to empower individuals to effortlessly discover and connect with potential romantic partners, redefining the online dating experience for the modern single without any cost. Also allowing the user to search the compatible user according to the user interests using different filters.

1.3 Objectives

The objective of developing the System is:

- To develop an easy-to-use platform that connects people looking for partners, highlighting their preferences and interests.
- To enhance the user experience by offering customized and plentiful match-finding options.
- To implement a secure messaging system in the app through real-time communication, utilizing XOR encryption methods for message encryption.

1.4 Scope and Limitation

Scope

The scope of this project includes User Registration and Profile Creation for personalized profiles, prioritizing Secure Messaging to protect user privacy. It also promotes User Interactions through features like liking and swiping for active engagement. Responsive Design ensures a seamless user experience on desktop and mobile devices. These features collectively define the scope, emphasizing user personalization, communication security, and interactive, user-friendly design.

Limitation

One of the major challenges is technical constraints, potentially affecting the scope and complexity of features due to limitations in development resources, and time constraints. Despite prioritizing user data security, the project may not be completely immune to all security threats, and implementing advanced security measures could demand additional resources and expertise.

1.5 Development Methodology

For the development of our online dating platform, we will employ the Waterfall Model, a structured and sequential approach ideal for projects with well-defined goals and stable requirements. This process begins with the Requirements Analysis phase, where we gather comprehensive information through market research to ascertain the features desired in the system. Following this, the System Design phase involves the creation of detailed design documents, outlining the system's architecture, user interface, and technical specifications. Next, in the Implementation phase, we commence coding, focusing on building a robust, user-friendly platform with key features such as secure text messaging and customizable profiles. After implementation, the Integration and Testing phase ensures all software components function seamlessly together, undergoing extensive testing for usability, performance, security, and compatibility. The Deployment phase follows, where the system is made available to users, with the necessary infrastructure established for its support. Lastly, the project transitions into the Maintenance phase, where ongoing monitoring, bug fixing, and updates occur to ensure the system's continuous functionality, security, and relevance to user needs.

1.6 Report Organization

This report document contains five chapters including this chapter. Chapter One stated the introduction, objective, problem statements, and limitations of the system we made. Chapter Two defines and describes the Background Study and Overview of related existing systems. Chapter Three presents the System Analysis and Design including Requirement Analysis and Feasibility Analysis. Chapter Four presents the Implementation, Testing, and debugging are explained. In chapter five, the Conclusion, Limitations, and Future Recommendations are briefly explained.

CHAPTER TWO
BACKGROUND STUDY AND LITERATURE REVIEW

2.1 Background Study

Online dating has witnessed a remarkable transformation in recent years, redefining how individuals seek and cultivate romantic relationships. In the early 1990s, pioneers like Match.com and Kiss.com introduced the concept of online dating, allowing users to create profiles and connect with others. However, it was the advent of smartphones in the early 21st century that brought about a seismic shift. Mobile dating apps like Tinder, Bumble, and OkCupid simplified the process, making online dating more accessible and interactive.

The COVID-19 pandemic in 2020 accelerated the adoption of online dating. With social distancing measures in place, individuals turned to digital platforms to connect safely from their homes. This global surge in online dating usage highlighted the importance of secure messaging systems, detailed user profiles, and privacy controls.

In [1] Modern online dating apps prioritize personalization, using algorithms to match users based on compatibility factors. These apps also emphasize safety, such as user verification and blocking user mechanisms. Inclusivity has become a focal point, recognizing diverse orientations, gender identities, and cultural backgrounds.

In this evolving landscape, our dating system project aims to contribute by creating a user-friendly platform that combines secure communication with an effective matching system. Building upon the history of online dating, we seek to meet the changing expectations of users, providing them with a modern and enjoyable dating experience that caters to the needs of contemporary singles.

2.3 Literature Review

The rise of digital technology has transformed the way people meet and form romantic relationships. Online dating platforms have emerged as popular avenues for individuals to connect with potential partners, providing a virtual space for exploration, interaction, and relationship building. This literature review aims to explore the existing body of research and literature related to online dating platforms, examining various aspects such as user behavior, matching algorithms, communication tools, and social implications.

In [2] The study by Blackhart, Fitzpatrick, and Williamson (2015) examines the dispositional factors that predict the use of online dating sites and behaviors associated with online dating. They found that extraversion, self-esteem, and attachment styles influence individuals' likelihood of using online dating platforms and their subsequent behaviors within these platforms. This research contributes to understanding the psychological factors driving online dating usage and behavior, which can inform the development of tailored services for improved user experiences and outcomes.

In [3] The study by Ellison, Heino, and Gibbs (2006) explores how individuals manage their impressions and present themselves in the online dating environment. It highlights the strategies used to create a positive image and attract potential partners. The research emphasizes the importance of self-presentation and visual cues in shaping perceptions and initial attraction. This study provides valuable insights into the dynamics of online dating profiles and can inform the development of effective strategies for successful online interactions.

In [4], Toma, Hancock, and Ellison (2008) delved into the intricacies of online deception in the context of online dating profiles. Their research examined the prevalence of deceptive practices such as misrepresentation of age, physical appearance, and relationship status. This study offers valuable insights into the challenges of authenticity and trust in the online dating landscape.

In [5], Finkel et al. (2012) conducted a meta-analysis of research on online dating and provided an overview of the strengths and weaknesses of online dating platforms. They discussed factors such as the effectiveness of matching algorithms and the impact of online dating on long-term relationship outcomes. This meta-analysis contributes to a broader understanding of the scientific basis of online dating.

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis

For our online dating platform, system analysis is crucial to identify and enhance system objectives. The first step involves Requirement Analysis, examining both functional (features like user registration, and secure messaging) and non-functional aspects (performance, security). Feasibility is assessed, by considering technical, economic, and operational constraints. The next focus is on designing a data model for user information and a process model illustrating system flow. This systematic analysis ensures the development of an efficient and user-centric online dating platform.

3.1.1 Requirement Analysis

Our system is designed with specific software requirements, both functional and non-functional, that need to be integrated. The following categories outline these essential requirements.

i. Functional Requirements

The functional requirements of this project are to login, view the user dashboard, create a profile, edit a profile, swap and match partners, message, receive notifications, and account settings.

For Users:

- Register and Login with a Google account.
- View User Dashboard
- Create/edit profiles with personal details.
- Search and match based on user preferences..
- Send/receive messages with matches.
- Receive notifications for activities.
- Customize account settings.

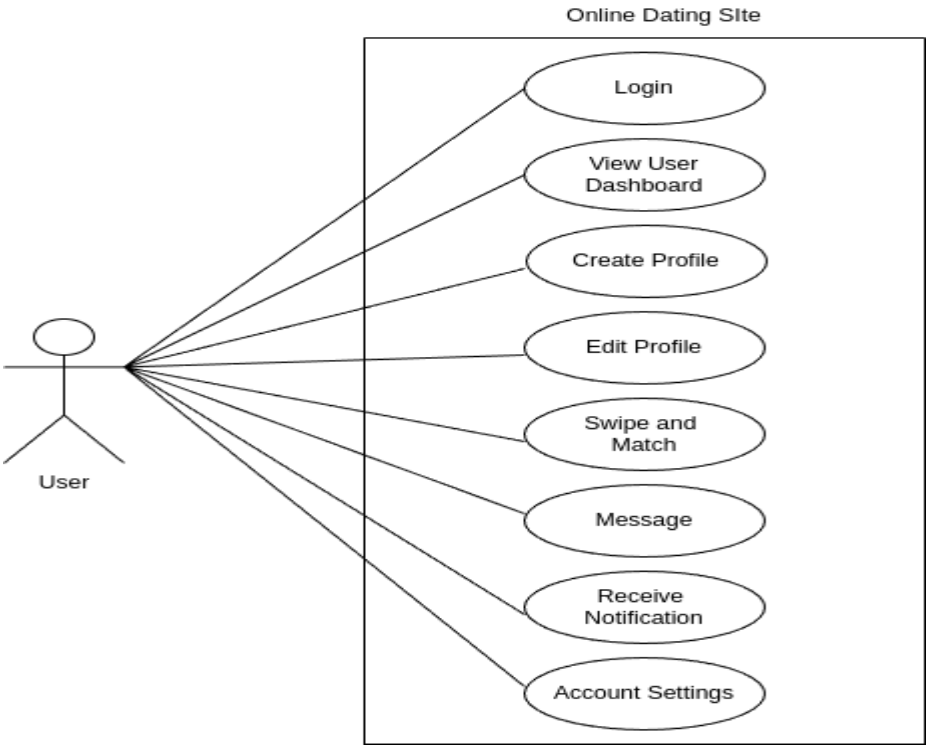


Fig 3.1: Use Case for User

For Admin:

- Admin login.
- View Dashboard.
- Manage user accounts.
- System Management.
- Features Management.

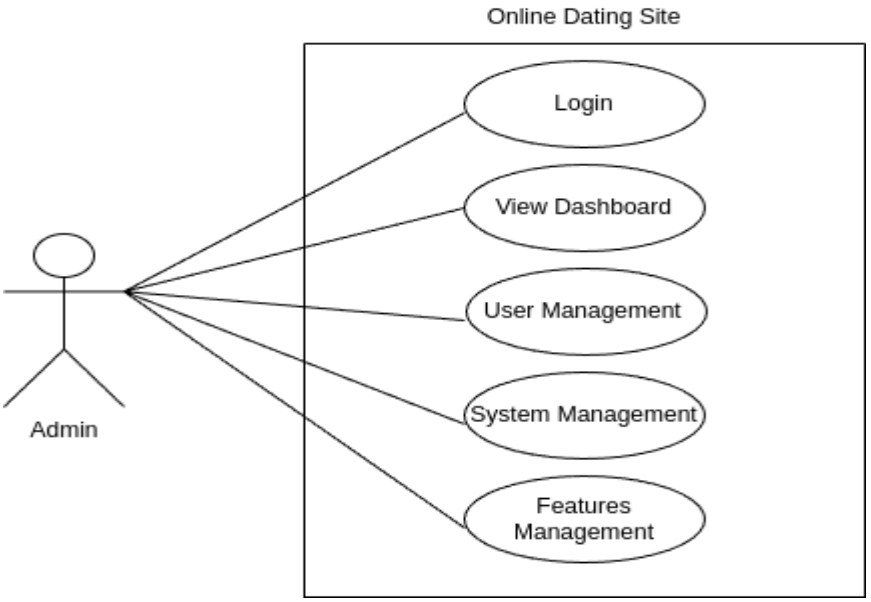


Fig 3.2: Use Case for Admin

ii. Non-Functional Requirement

Non-functional requirements in a dating system focus on the quality attributes and criteria that impact the overall user experience. These encompass factors such as performance, security, scalability, data privacy, and usability. Non-functional requirements ensure that the system not only functions correctly but also provides a reliable, secure, and user-friendly platform, ultimately contributing to its success and user satisfaction.

The non-functional requirements of our system are:

1. Performance: The system must be highly responsive and maintain smooth user interactions.
2. Scalability: The system must adapt to a growing user base without performance degradation.
3. Data Privacy and Security: The system should secure user data, including personal information and messages, that must be securely stored and transmitted.
4. Availability: The system must be accessible 24/7, with minimal downtime for maintenance or unforeseen issues to provide uninterrupted service.
5. Cross-Platform Compatibility: The system should work seamlessly on various devices and platforms, including web browsers, and mobile phones.

3.1.2 Feasibility Analysis

A feasibility study is a process of evaluating a project or idea to determine if it is practical and achievable. It helps assess whether the project is worth pursuing and if it can be successfully completed. In simple terms, a feasibility study examines whether something is doable or not.

i. Technical feasibility

By utilizing JavaScript, HTML, CSS, and Django web framework, we will create a technically feasible online dating platform. Django offers a robust foundation for web application development, providing features such as authentication, database management, and security. Our system will be designed to handle all the necessary functionality, implementing Django's MVT architecture and built-in testing framework for a secure and scalable backend. On the front end, JavaScript, HTML, and CSS will be leveraged to create a visually appealing and responsive user interface. These technologies will enable us to develop a modern and user-friendly online dating platform while adhering to industry best practices.

ii. Operational feasibility

The modular structure of JavaScript, HTML, CSS, and Django allows for easy scalability and adaptability to evolving requirements, making the online dating platform operationally feasible. The user-friendly interface and thorough testing and debugging processes ensure smooth operations for administrators and end-users. JavaScript, HTML, CSS, and Django's stability, along with their developer-friendly environments and extensive available resources, contribute to the feasibility of the system, ensuring successful project execution.

iii. Economic feasibility

The online dating platform, built using open-source technologies such as JavaScript, HTML, CSS, and Django, is economically feasible. The utilization of open-source technologies eliminates the need for expensive software licenses, reducing costs significantly. The use of draw.io for system architecture and user interface design further reduces design costs. Overall, the adoption of open-source technologies and draw.io makes the development of the online dating platform an economically feasible solution for creating an efficient and user-centric online dating platform.

iv. Schedule Feasibility

The schedule was precise and accurate for the timely completion of the project.
This project was completed and submitted within the time limit of the schedule.
To develop this application, we have scheduled the time to develop the application which is shown in the figure below:

Table 3.1 Gantt Chart

ID	Task Name	Duration	2023				
			Apr	May	Jun	July	Aug
1	Study and Analysis	28 days					
2	Data Collection	16 days					
3	Implementation	47 days					
4	Testing and Analysis	7 days					
5	Documentation	105 days					
6	Review	3 days					
7	Presentation and Submission	1 day					

3.1.3 Object Modeling

Object modeling in this online dating system is used to structure and design the software. It defines how objects like Users, Matches, and Messages interact and function within the system. It encapsulates data and behavior, establishes relationships between objects, and guides the system's overall functionality. This approach simplifies maintenance, enhances scalability, and makes the system easier to understand and develop.

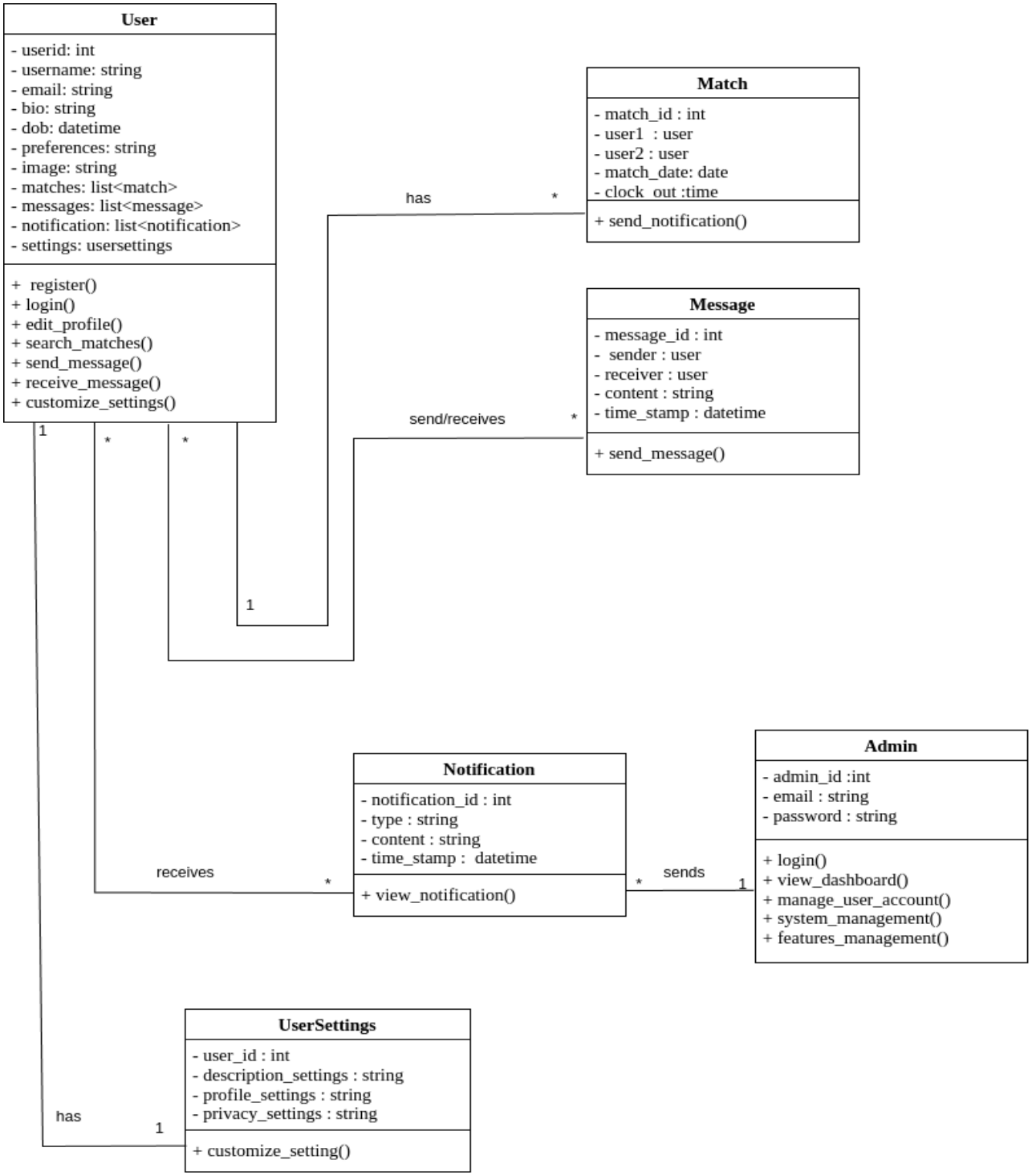


Fig 3.3 Class Diagram

3.1.4 Dynamic Modeling

The dynamic modeling sequence diagram for the online dating system details a user's flow from registration to logout, outlining the sequential use of features like profile editing, matching, messaging, and settings. The component diagram for administrators shows a separate login leading to a dashboard for user, system, and feature management, highlighting backend control essential for system maintenance and operation. Together, they represent the system's dynamic interactions from both user and admin perspectives.

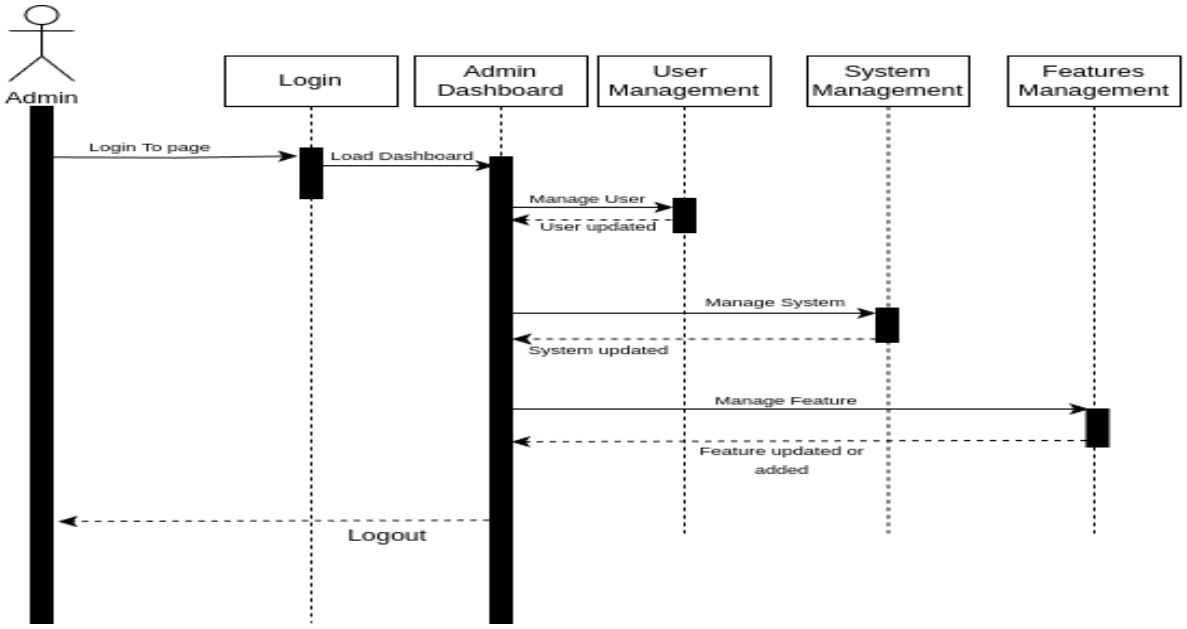


Fig 3.4 Sequence Diagram of Admin

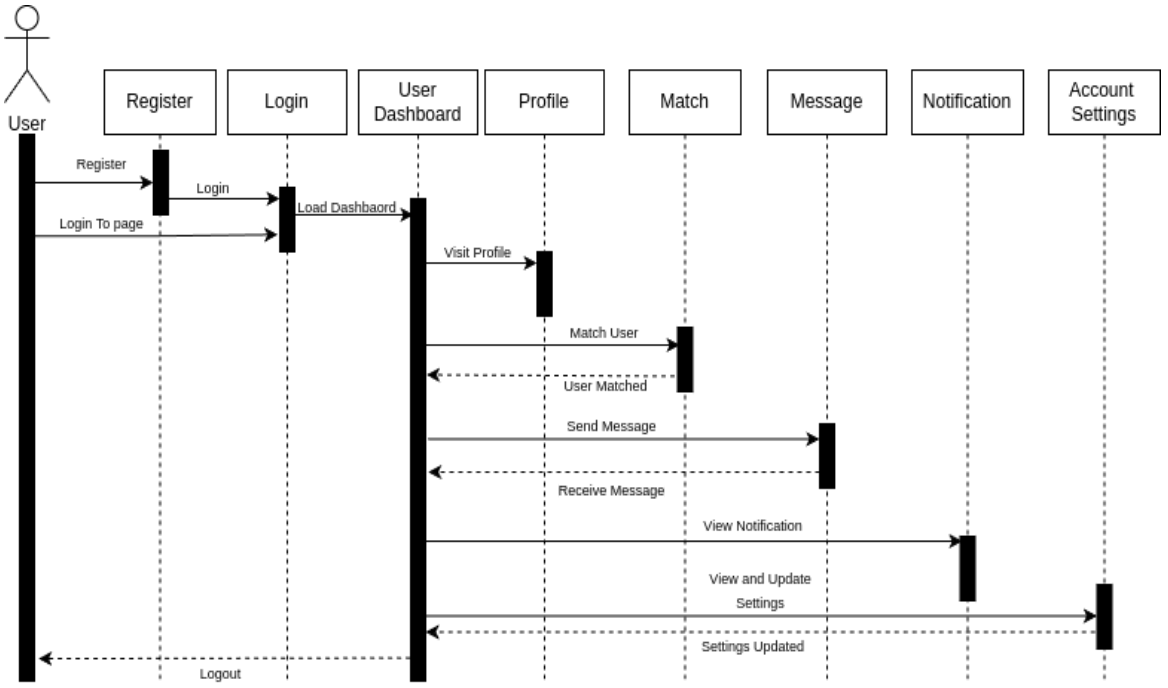


Fig 3.5 Sequence Diagram of User

3.1.5 Process Modeling

Process modeling for the online dating system describes the operational workflows. For users, it includes the steps from registration, login, profile updates, matching, and messaging, along with settings customization. For admins, it outlines login, user and system management, and feature updates. This modeling maps the flow and data transfer through the system, optimizing operations and user experience.

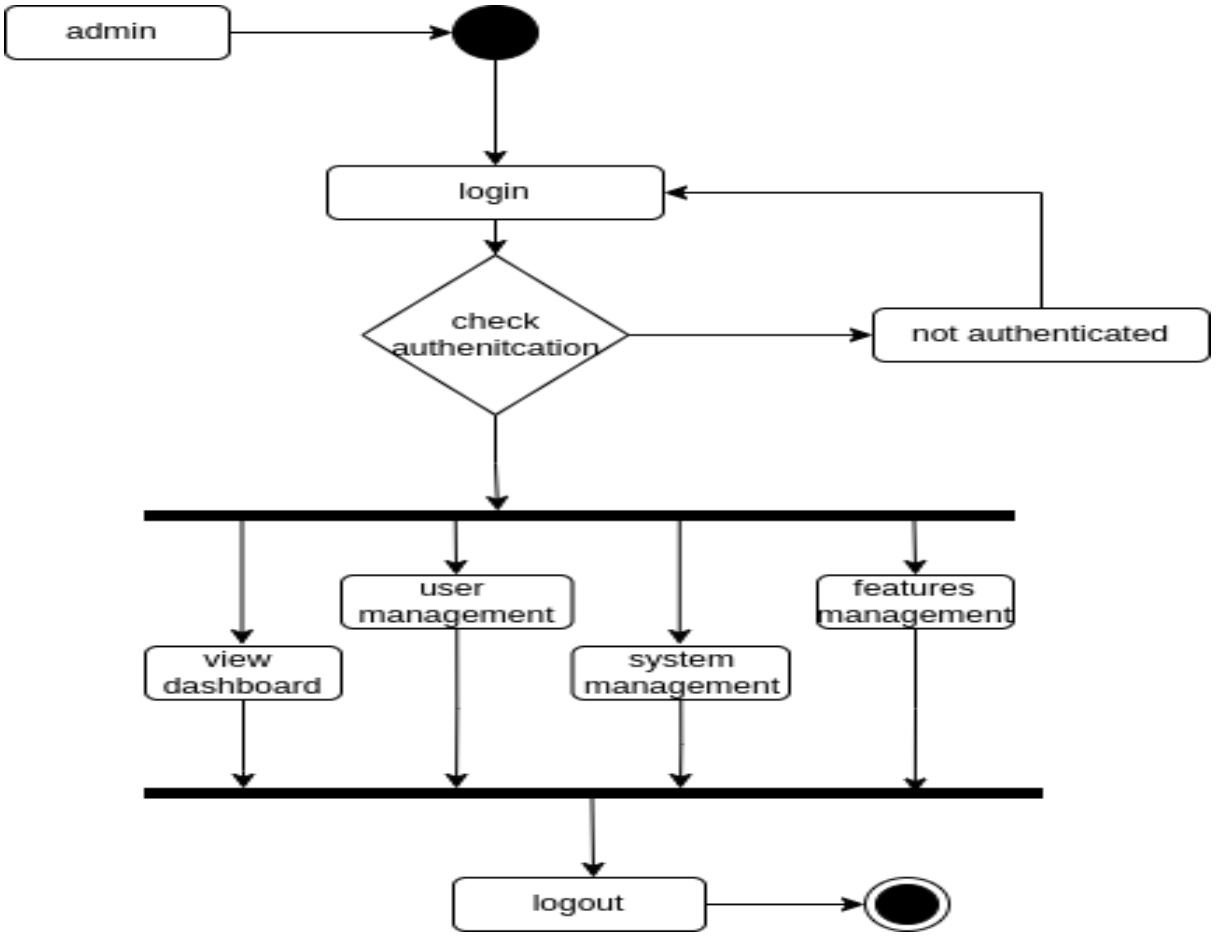


Fig 3.6 Activity Diagram of Admin

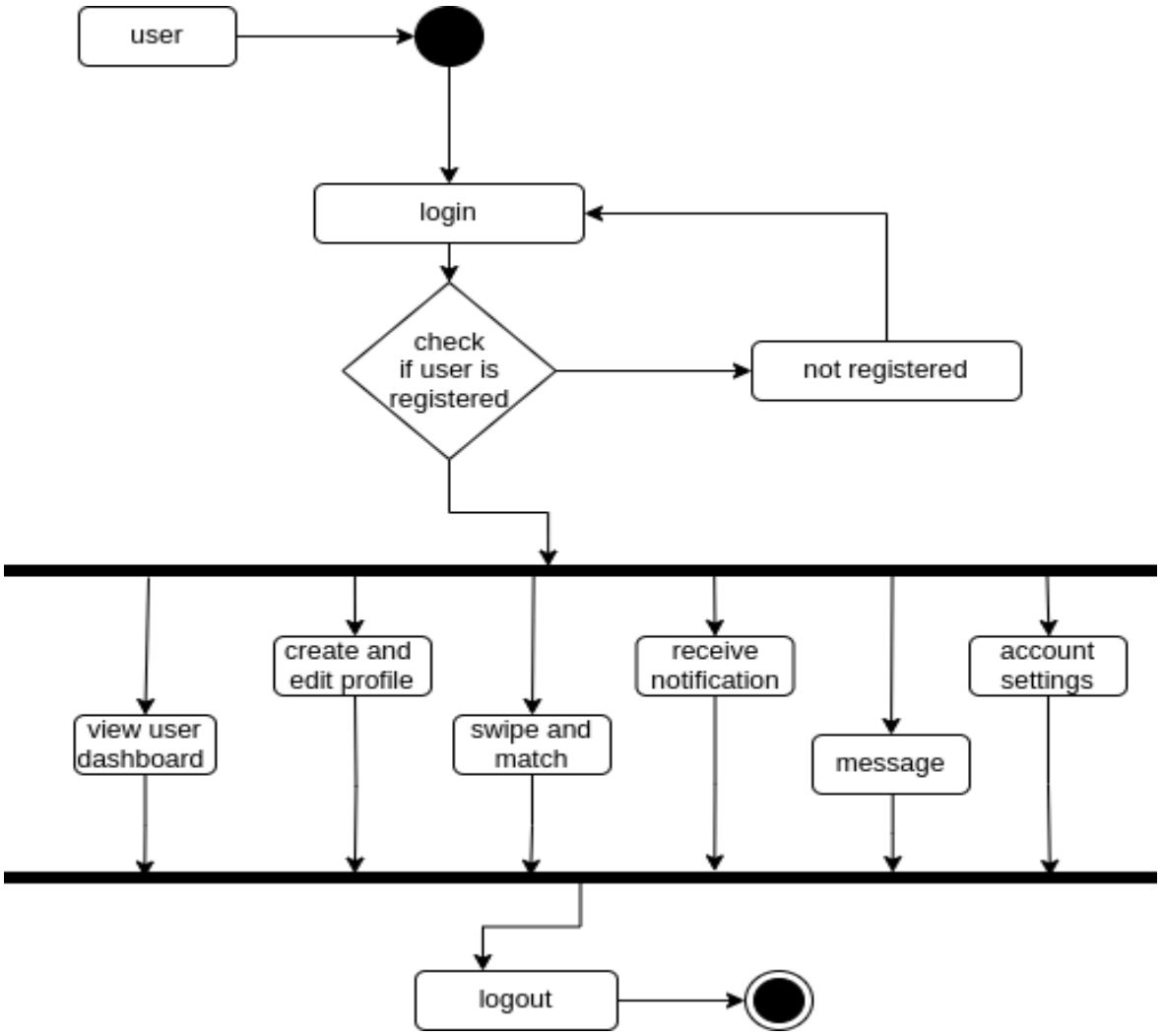


Fig 3.7 Activity Diagram of User

3.2 System Design

System design, also known as system architecture or software design, is a critical phase in the software development life cycle where you transform the requirements gathered during the system analysis phase into a detailed blueprint for building the system. The goal of system design is to create a structured and organized plan for implementing the software system that meets the specified requirements efficiently and effectively.

3.2.1 Component diagram

The component diagram depicts various system components and their relationships. The components include "User," "Match," "Message," "Profile," "System," "Features," and "Admin." There are connectors showing relationships, such as "User ID" connecting "User" to "Match" and "Message," and "Feature Name" linking "Features" to "User." There's also a "System Name" connector from "System" to "User." These connectors indicate data flow or interaction between the components. The diagram is organized with labeled connectors and components, typical of software design schematics for representing system architecture.

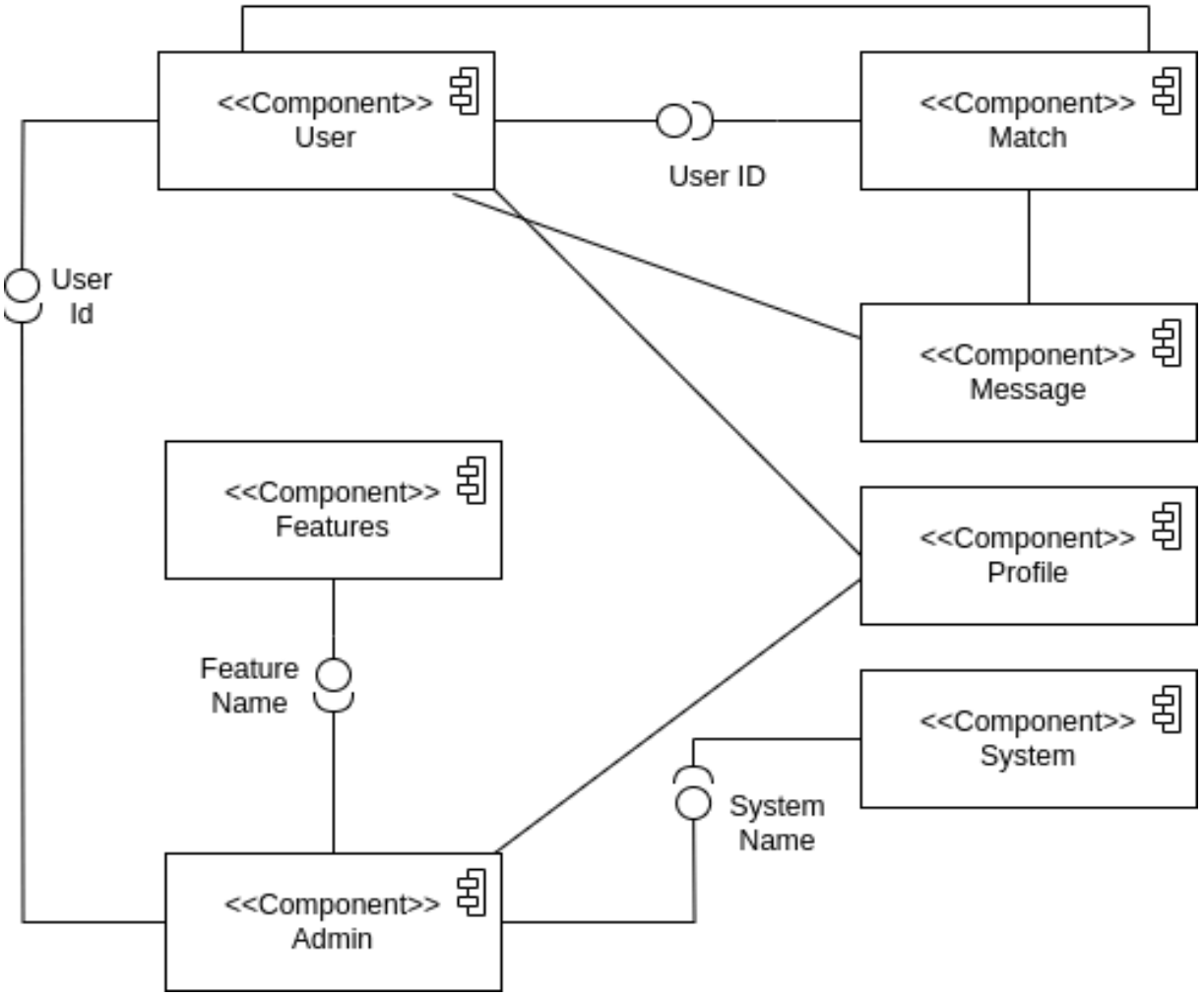


Fig 3.8 Component Diagram

3.2.2 Deployment diagram

The diagram illustrates a deployment diagram for a web application, showing three main nodes: a Client node (web browser), an Application Server node (with Admin, User, Match User, and Message components), and a Database Server node. It demonstrates a three-tier architecture with the Client node linked to the Application Server via HTTP Request and the Server connected to the Database through Database Queries.

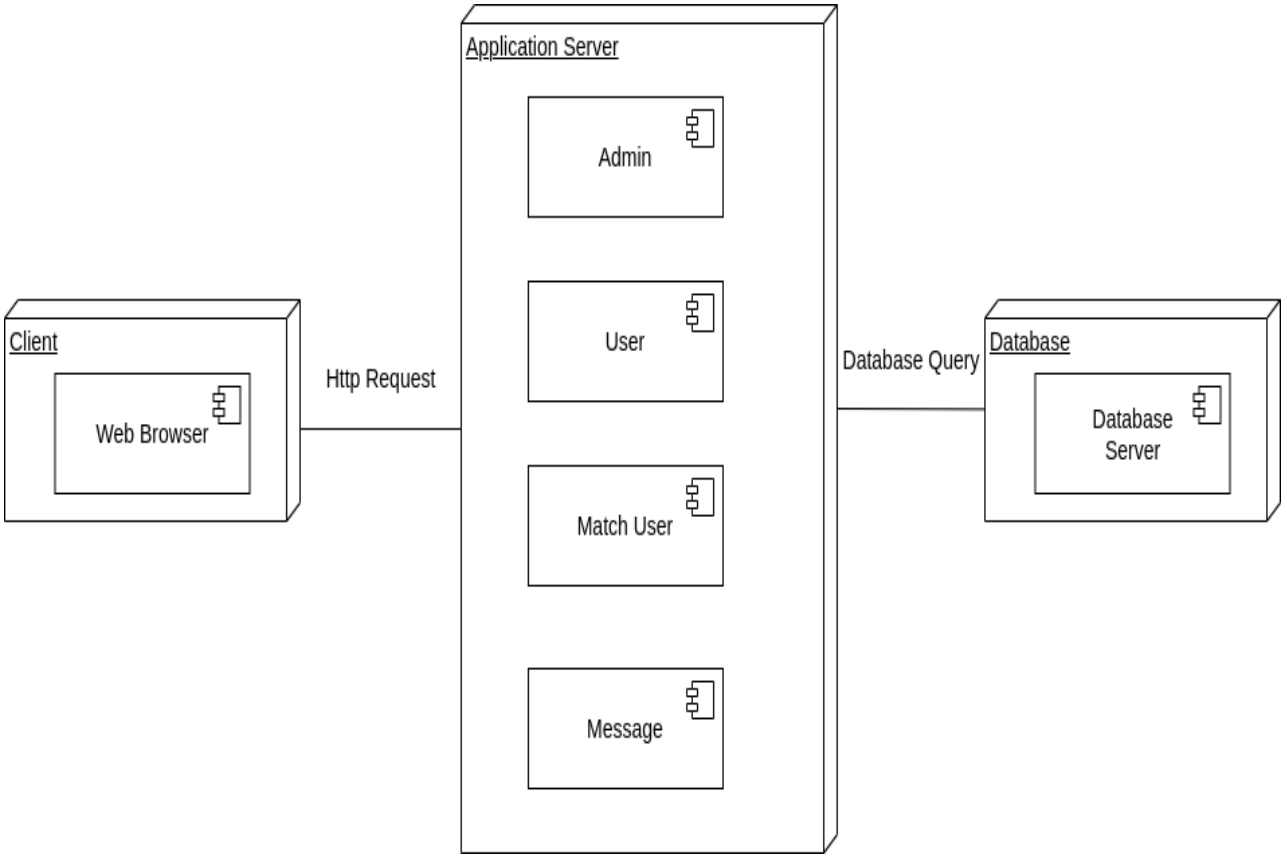


Fig 3.9 Deployment Diagram

3.3 Algorithm Details

XOR-based encryption is used to encrypt and decrypt messages while communicating in the system. As a shared key is needed for this encryption method Diffie-Hellman key exchange is used. The Diffie-Hellman key exchange is a widely used cryptographic protocol that allows two parties to establish a shared secret key over an unsecured communication channel. This shared key can then be used for secure communication, such as encrypting and decrypting messages between users. Here's is of how Diffie-Hellman works:

- Initialization: Both parties, let's call them Alice and Bob, agree on two publicly known values:
    - A large prime number is typically denoted as 'p'.
    - A primitive root modulo 'p', often denoted as 'g'.
- Private Key Generation: Each party generates a private key, which they keep secret. These private keys are usually denoted as 'a' for Alice and 'b' for Bob. These keys can be any random number within a certain range, typically between 1 and 'p-1'.
- Public Key Calculation: Each party calculates their public key using the following formula:
    - Alice's public key (A) = (g^a) % p
    - Bob's public key (B) = (g^b) % p

- Exchange of Public Keys: Alice and Bob exchange their public keys over the insecure channel.
- Shared Secret Calculation: Both Alice and Bob can now calculate the shared secret key using the public key they received and their private key:
    - Alice: Shared Secret (S) = (B^a) % p
    - Bob: Shared Secret (S) = (A^b) % p

The Diffie-Hellman key exchange is a method for securely exchanging cryptographic keys over a public channel. In Python, you can implement Diffie-Hellman like this:

CHAPTER FOUR

IMPLEMENTATION AND TESTING

4.1 Implementation

We'll follow a well-structured implementation process for the dating system project, primarily utilizing the waterfall approach due to its clarity and simplicity. This approach allows us to proceed sequentially through key phases: planning, analysis, design, implementation, and testing. In the planning phase, we'll define project objectives. During analysis, we'll thoroughly understand user needs and current systems. The design phase will address system architecture and functionality. Implementation is the most resource-intensive phase, where we'll write the code for the app. Finally, testing ensures that the system is error-free and meets all requirements before release. This structured approach will help us efficiently deliver a robust and functional dating app.

4.1.1 Tools Used

The following are the tools used for the development of this project.

1. HTML and CSS:

   HTML is utilized for structuring web pages and defining elements like headers, paragraphs, links, and images. CSS will complement HTML to enhance the visual presentation of the system, covering aspects such as fonts, layouts, and colors, ensuring an appealing user interface.

2. Django:

   Django is chosen as the backend framework for this project. It will facilitate the management of dynamic content, databases, and session tracking, ensuring the smooth functioning of the system’s backend processes.

3. JavaScript:

   JavaScript is employed to add interactivity to the web pages, enhancing the overall user experience. It will enable dynamic and responsive features, making the system more engaging and user-friendly.

4. Bootstrap:

   Bootstrap, a popular front-end framework, is used to streamline the development process and ensure mobile responsiveness. It provides pre-designed UI components and layouts, saving time and effort in designing and optimizing the user interface.

4.1.2 Implementation details of modules

Profile Creation Module:

In the Profile Creation Module in an online dating app offers a user-friendly interface for users to set up and customize their profiles. It includes easy options for adding personal details, uploading photos, and selecting preferences for better matchmaking. Focused on user privacy and security, it allows control over personal information visibility. Seamlessly integrated with the app's features, it ensures a smooth and enjoyable profile setup experience

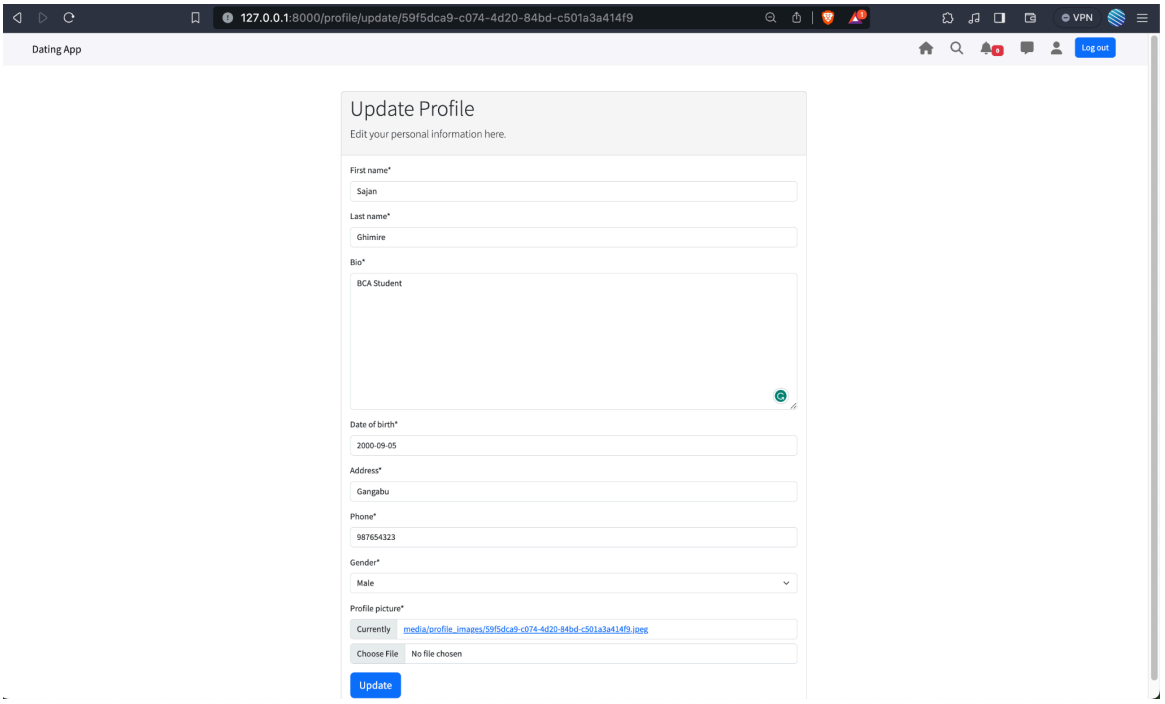


Fig 4.1: Profile Creation Module

Messaging Module:

In messaging module, Django arrange the messaging module, ensuring robust message storage, retrieval, and encryption. Employing XOR encryption and the Diffie-Hellman Key Exchange protocol, it achieve communication security. Additionally, Django tracks user activity, displaying real-time online status for seamless interactions. This combination of Django's capabilities and encryption methodologies guarantees both secure communication and enhanced user experience, fostering effortless and protected exchanges.

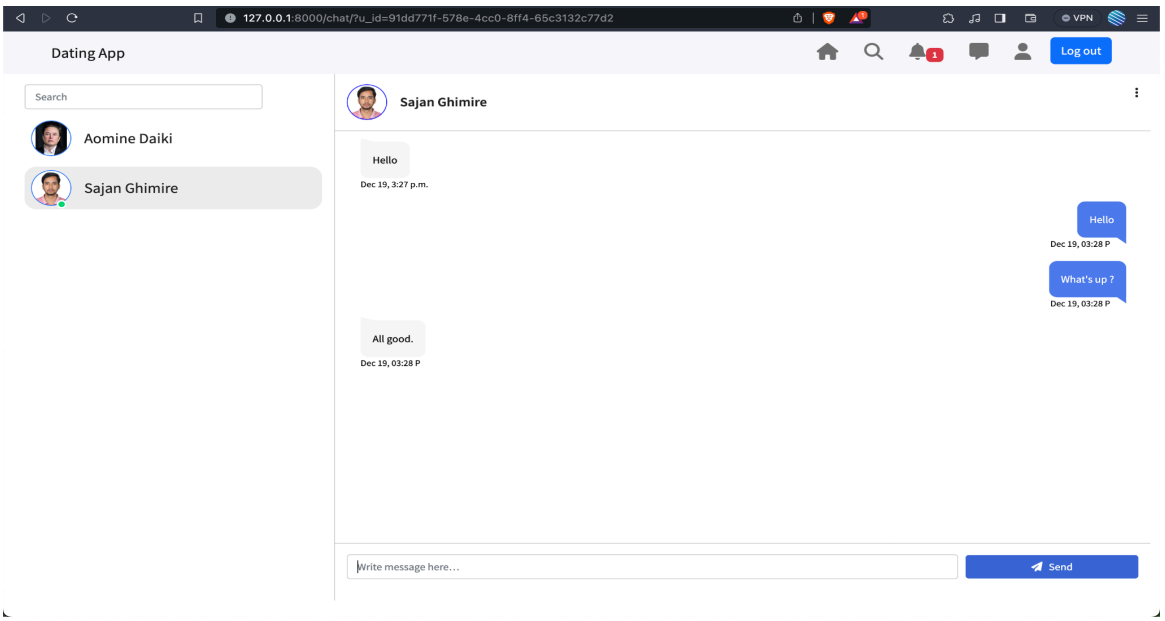
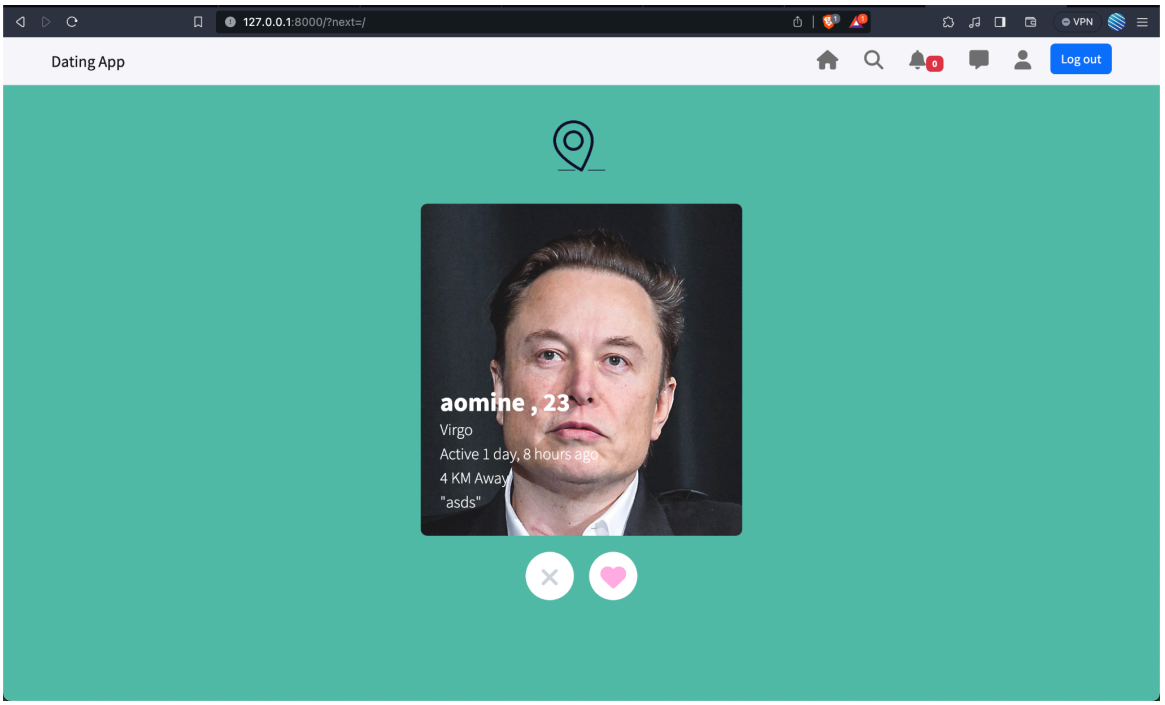


Fig 4.2: Message Module

User Interaction Module:

The User Interaction Module in the dating app presents users with potential matches, showcasing photos and basic information such as age, zodiac sign, and location. Users can express their interest with a simple tap on the heart icon or move on by selecting the cross.



Figu 4.3: User Interaction Module

Notifications Module:

The Notification Module in the app informs users of new interactions like likes. It provides real-time alerts with actionable buttons, like "Send back Heart," to foster quick engagement with potential matches. This module enhances user experience by keeping them updated and connected.

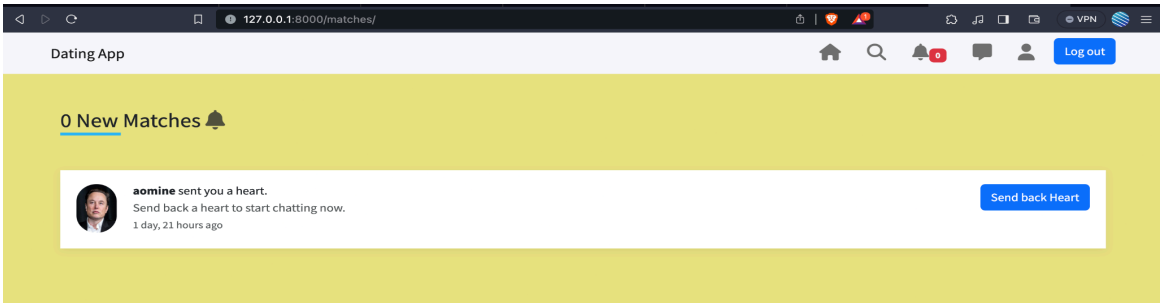


Fig 4.4: Notification Module

Admin Module:

The Admin Module is a centralized dashboard for managing user accounts on a dating app. It provides administrators with an at-a-glance view of user details such as email addresses, and activity status. The module allows for easy editing of user information, account activation or deactivation, and password resets. A sidebar for navigation facilitates quick access to various administrative functions, while an integrated search bar capabilities streamline the management process. Ensuring secure access, the responsive design of the module guarantees efficient administration across various devices.

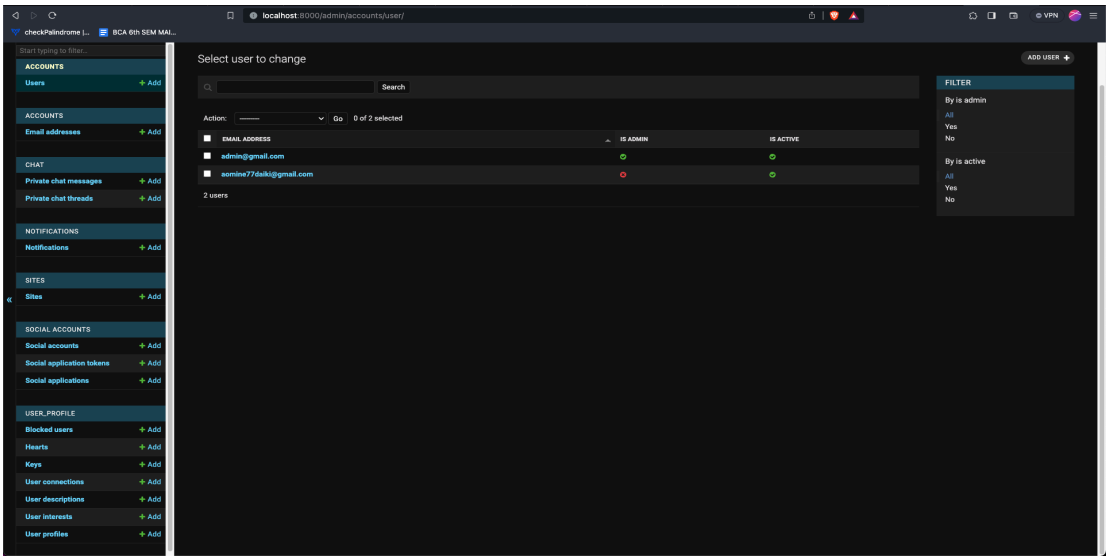


Fig: 4.5: Admin Module

4.2 Testing

Throughout the development process, various rounds of testing were conducted to ensure the reliability and functionality of our dating system. Testing was divided into two main categories: unit testing and system testing.

4.2.1 Test Cases for Unit Testing

Unit testing involves testing individual components or modules of the system to ensure they work correctly. Here are some of the test cases for unit testing:

Table 4.1: Admin Login Unit Testing

S.No	Test Case Description	Test Data	Expected Result	Actual Result	Pass/ Fail
1.	Invalid Credentials	Wrong email and password	Display a message of invalid credentials	Success	Pass
2.	Valid Credentials	Correct Admin credentials	Admin Logs in successfully	Success	Pass

Table 4.2: User Registration and Login unit testing.

S.No	Test Case Description	Test Data	Expected Result	Actual Result	Pass/ Fail
1.	User clicks on “Register” button	Redirected to Google authentication page	The Google authentication page is displayed	Success	Pass
2.	User completes Google authentication process	Google account info is stored in the system’s database.	User’s Google account info is successfully stored	Success	Pass

3.	User Registration	SuccessFull Registration	User registered successfully	Success	Pass
4.	User Clicks on “LogOut” Button	Session terminated user is logged out	User is successfully logged out	Success	Pass

Table 4.3:Validation and Functionality Unit Testing

S.No	**Test Case Description**	**Test Data**	**Expected Result**	**Actual Result**	**Pass/ Fail**
1**.**	The user created a profile and provided incomplete information	Missing Profile details(e.g. name, DOB)	System displays a validation error message for missing information	Success	Pass
2.	Invalid Contact Number format	Invalid contact number format entered	Display a message that the contact number is invalid	Success	Pass
3.	Invalid Date of Birth	Invalid date of birth should	Display a message that the user should be 18 +	Success	Pass
3.	Missing Field Validation	Require filed let blank	Display a message that a field is required.	Success	Pass
4.	User tries to update their profile with valid data	Valid data(e.g. updated interests, bio)	System updates the profile successfully	Success	Pass
5.	User blocks another user	Valid user ID of another user	Other user is blocked successfully	Success	Pass

Table 4.4 Admin Action Unit Testing

S.No	Test Case Description	Test Data	Expected Result	Action Result	Pass/ Fail
1.	Viewing User Profile	Email: Manoj@gmail.com ID: 123	Admin view the user profile of the user with ID 123	Success	Pass
2.	Removing User Profile	Email: santosh@gmail.com ID: 456	Admin removes the user profile of the user with ID 456	Success	Pass
3.	Viewing Blocked Users	Admin checks the blocked User	List of blocked users displayed	Success	Pass
4.	Viewing Connections	Admin checks the connection between users	List of connection between users	Success	Pass

4.2.2 Test Cases for System Testing

System testing focused on the integration of different modules to ensure they worked harmoniously together. Here is a sample test case for system testing:

Table: 4.5: Table Representing System Testing

S.No	Test Case Description	Test Data	Expected Result	Actual Result	Pass/ Fail
1.	Sending a heart.	Sending a heart to another user	Heart sends successfully	Success	Pass
2.	Receiving a heart	Receiving a heart from another user	Heart received successfully	Success	Pass
3.	Rejecting User	Rejecting another registered user	User Rejected Successfully	Success	Pass
4.	Viewing User Profile	Valid user profile data	User Profile displayed with accurate information	Success	Pass
5.	Editing User Profile	Valid profile updated	User profile updated successfully	Success	Pass
6.	Messaging another user	Messaging connected user	Message Successfully sent	Success	Pass
7.	Blocking Users	Blocking a specific user	User is blocked successfully	Success	Pass
8.	Filter Users	Filtering another user on basis of different filters	User filtered by different filters successfully	Success	Pass

CHAPTER FIVE
CONCLUSION AND FUTURE RECOMMENDATIONS

5.1 Conclusion

In conclusion, our dating system project aims to provide an efficient and user-friendly platform for individuals to connect and find potential partners. With features like profile creation, a matching system, secure messaging, and user interactions, the system promises to offer a modern and enjoyable online dating experience. It will be accessible on various devices, ensuring convenience for users looking for meaningful connections.

5.2 Lesson Learnt/Outcome

Implementing an online dating platform is expected to yield positive outcomes. These include an enhanced user experience with a user-friendly interface, improved match accuracy through personalized filters, effective communication tools, and prioritized user safety and privacy. Overall, the online dating platform aims to create a safe and enjoyable environment for users to connect and form meaningful relationships.

5.3. Future Recommendations

For the future of our dating system, we recommend the following: continually enhance features, prioritize security, optimize performance, ensure cross-device compatibility, gather and act on user feedback, explore potential integrations, and consider monetization strategies. These steps will help us maintain competitiveness and adaptability in the evolving online dating landscape.

APPENDICES

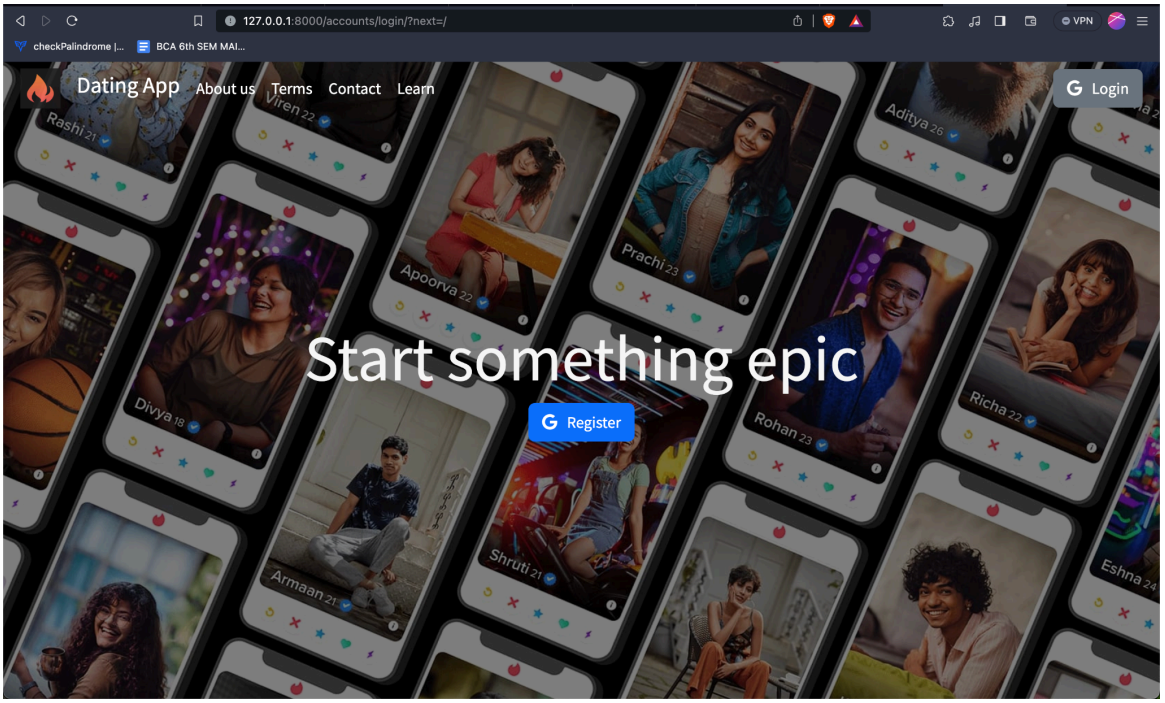


Figure A: User Landing Page

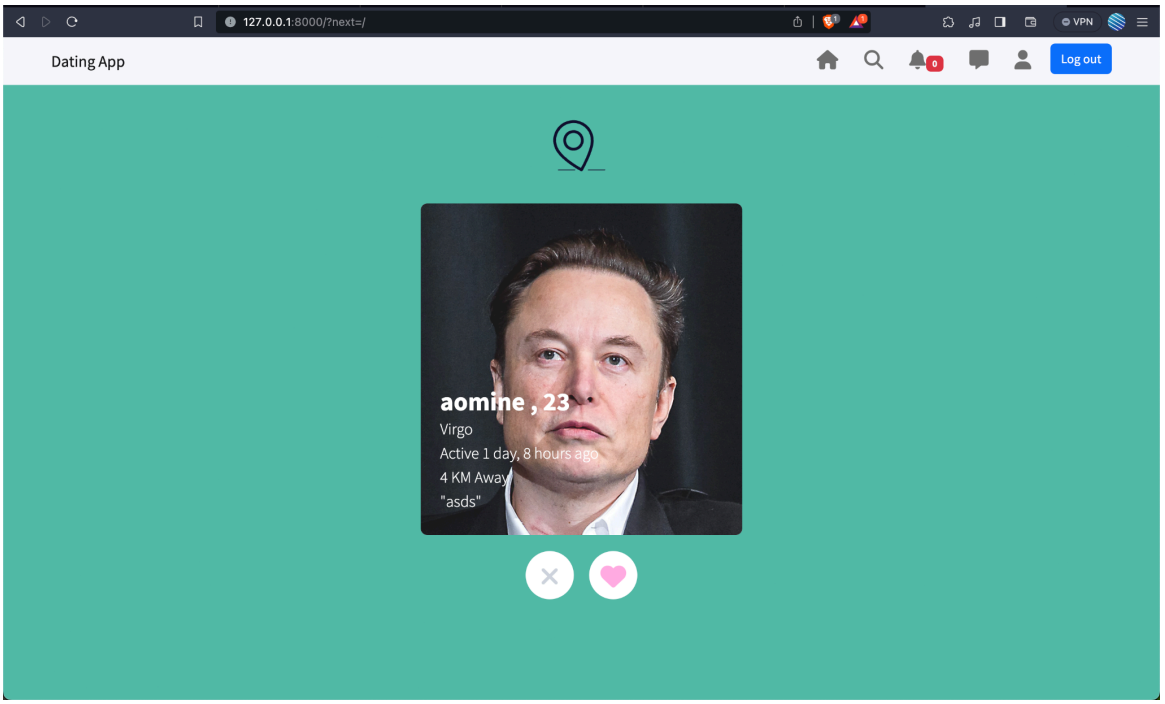


Figure B: User Homepage

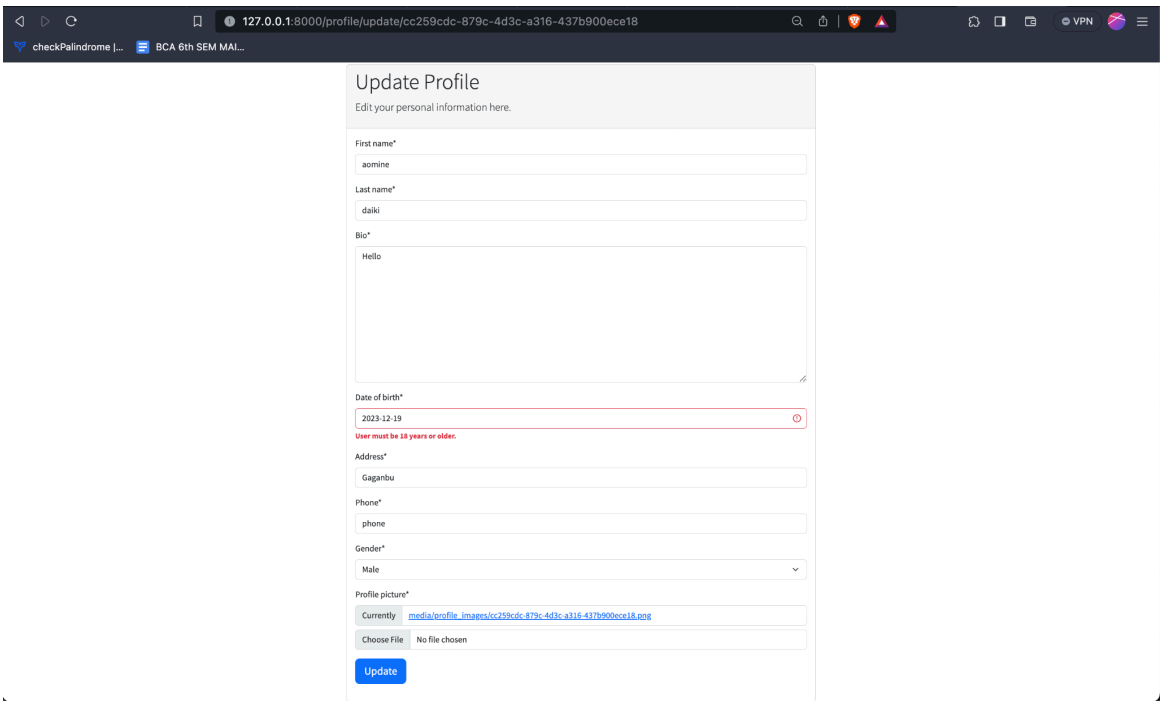


Figure C: Profile Creation with Invalid Date of Birth

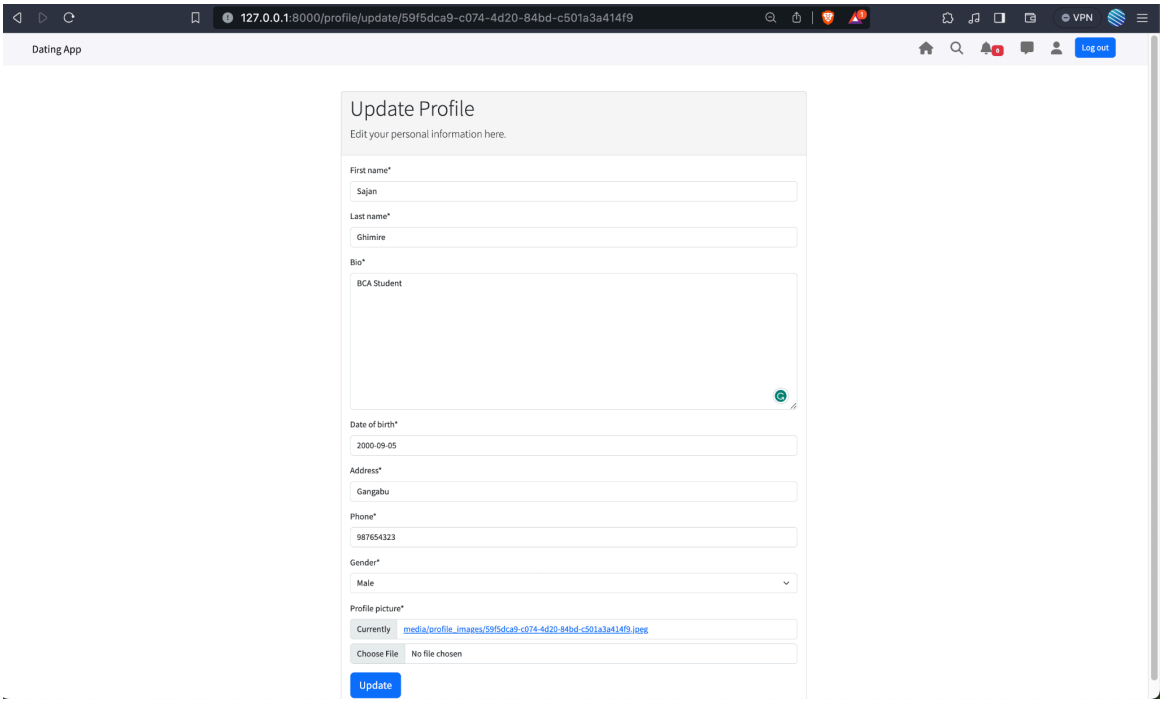


Figure D: Successful Profile Update

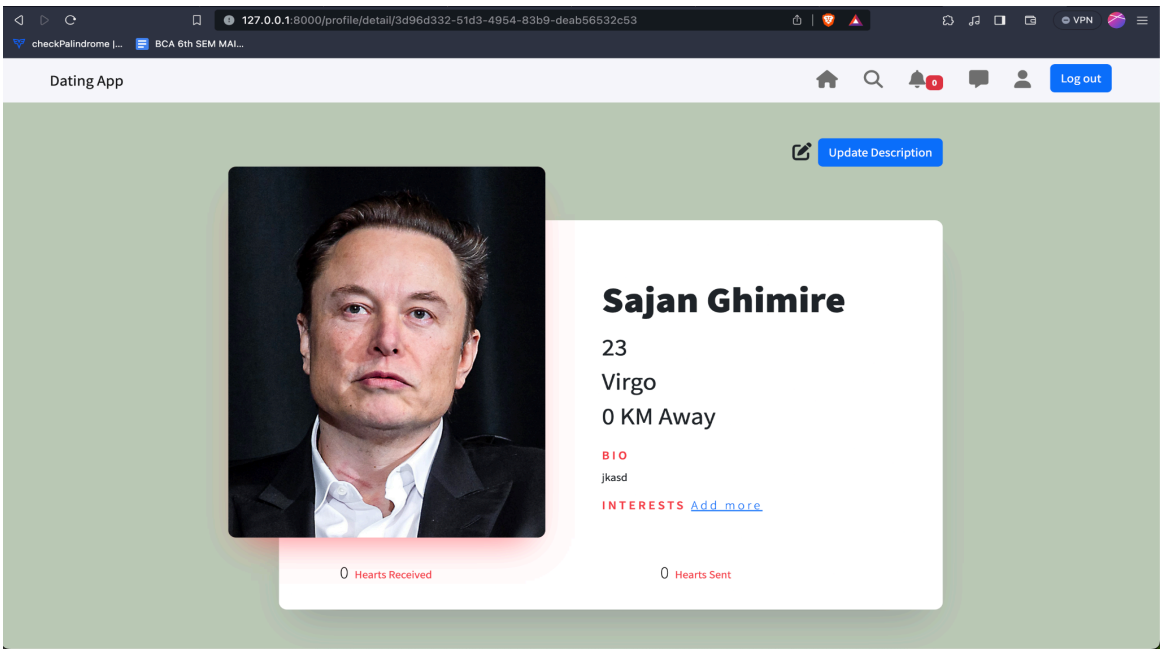


Figure E: Profile Detail Page

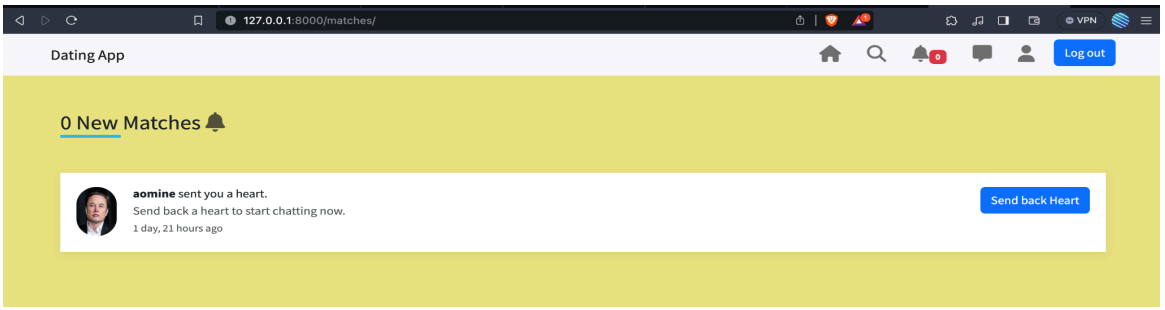


Figure F: Notification Page

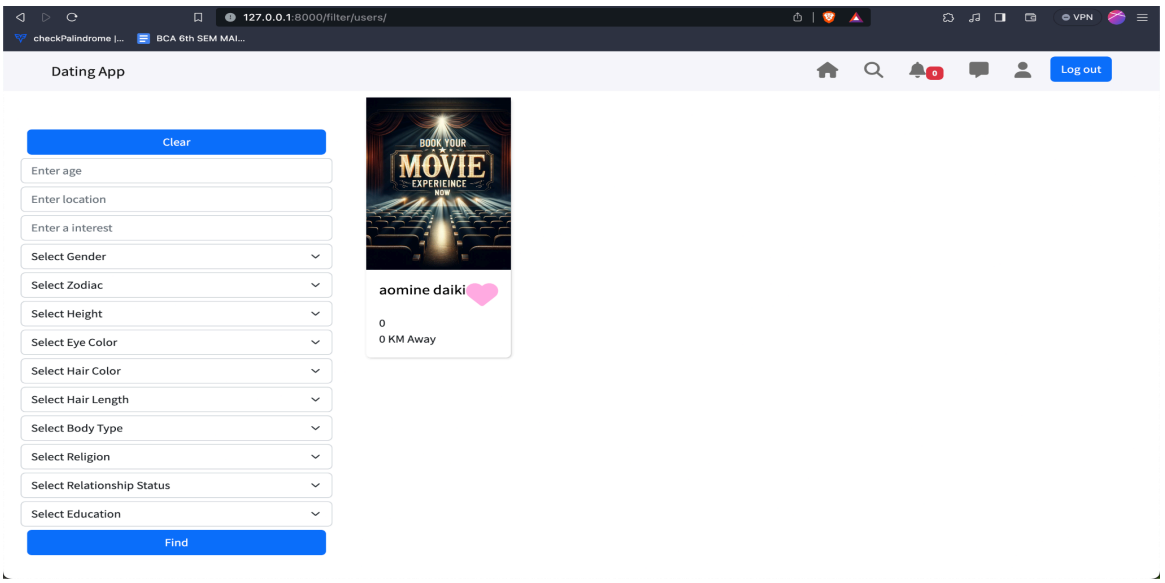


Figure G: User Filter Page

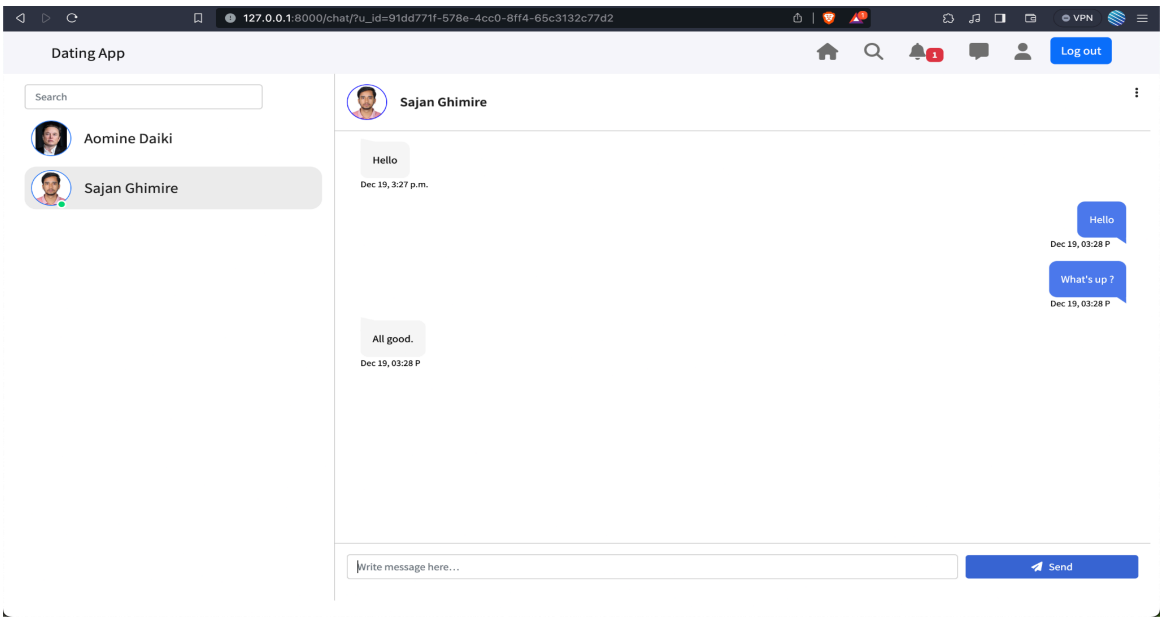


Figure H: Connected User Message Module.

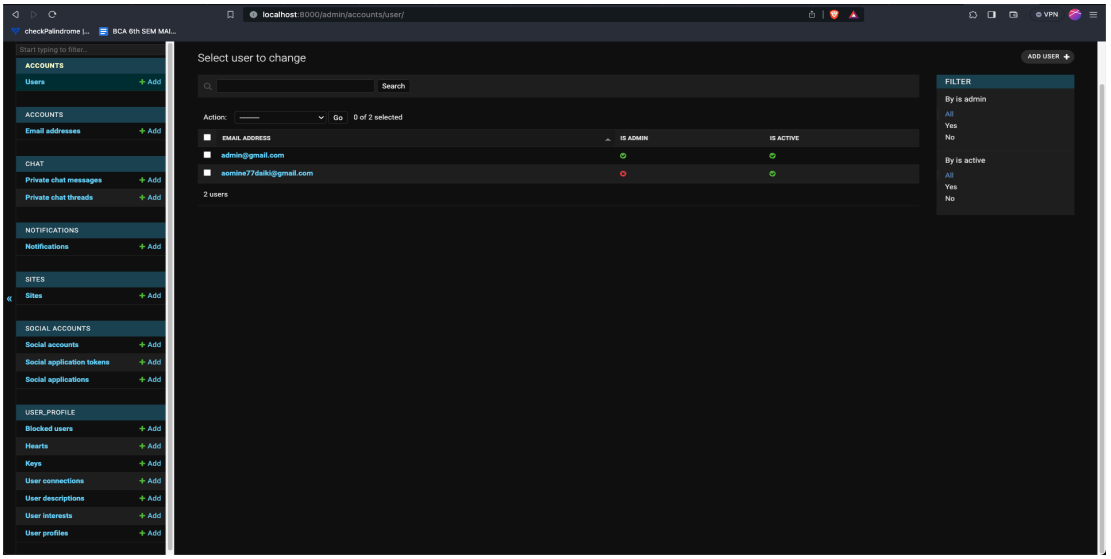


Figure I: User Module in Admin Dashboard.

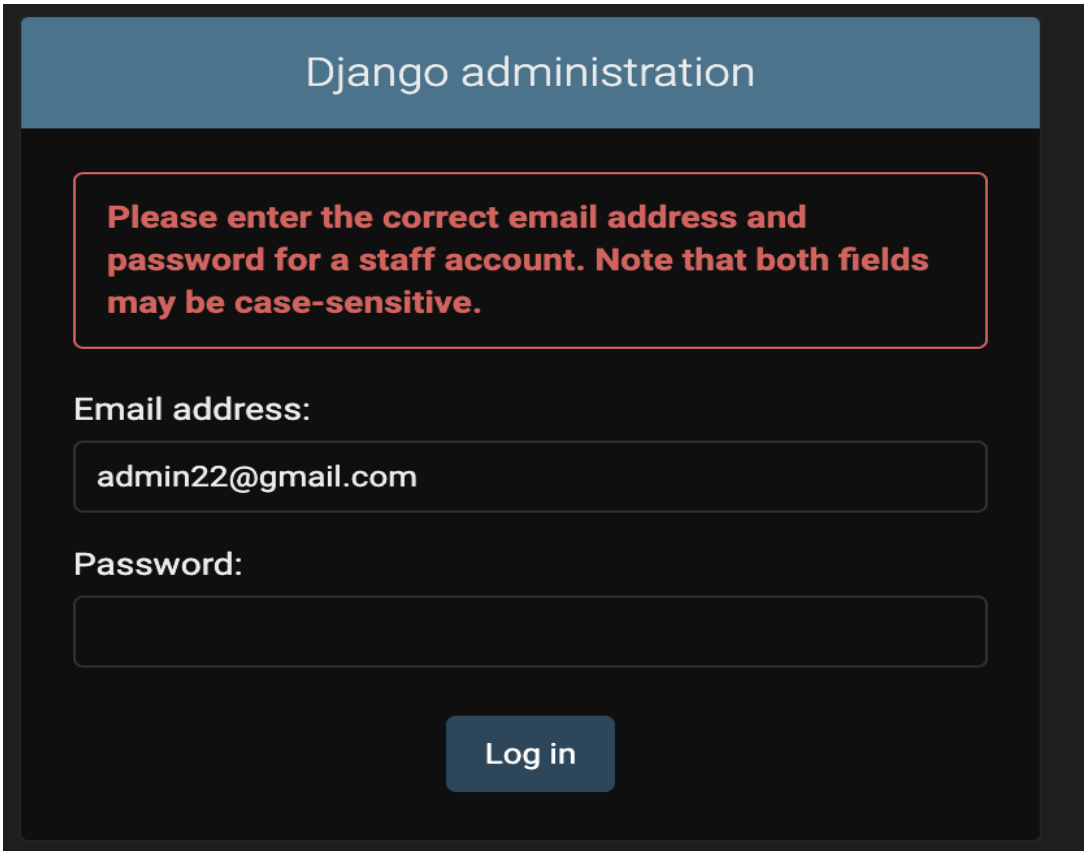


Figure J: Invalid Admin Credentials

REFERENCES

[1] Finkel, E. J., Eastwick, P. W., Karney, B. R., Reis, H. T., & Sprecher, S. (2012). Online dating: A critical analysis from the perspective of psychological science. Psychological Science in the Public Interest, 13(1), 3-66.

[2] Blackhart, G. C., Fitzpatrick, J., & Williamson, J. (2015). Dispositional factors predicting use of online dating sites and behaviors related to online dating. Computers in Human Behavior, 49, 120-129 [Online].

[3] Ellison, N. B., Heino, R. D., & Gibbs, J. L. (2006). Managing impressions online: Self-presentation processes in the online dating environment. Journal of Computer-Mediated Communication, 11(2), 415-441 [Online]**.**

[4] Toma, C. L., Hancock, J. T., & Ellison, N. B. (2008). Separating fact from fiction: An examination of deceptive self-presentation in online dating profiles. Personality and Social Psychology Bulletin, 34(8), 1023-1036.

[5] Finkel, E. J., Eastwick, P. W., Karney, B. R., Reis, H. T., & Sprecher, S. (2012). Online dating: A critical analysis from the perspective of psychological science. Psychological Science in the Public Interest, 13(1), 3-66.