SOLILOQUY

Live your life

A PROJECT REPORT SUBMITTED IN PARTIAL FULFILMENT OF REQUIREMENT
FOR THE AWARD OF THE DEGREE

MASTER OF COMPUTER APPLICATIONS (MCA)
OF
MAHATMA GANDHI UNIVERSITY, KOTTAYAM
BY

NANDHU JAYAN
Reg No: 22PMC139

MARIAN COLLEGE
KUTTIKKANAM
(AUTONOMOUS)

MAKING COMPLETE

Marian College Kuttikanam Autonomous
Peermade, Kerala – 685 531

2023

SOLILOQUY

Live your life

A PROJECT REPORT SUBMITTED IN PARTIAL FULFILMENT OF REQUIREMENT
FOR THE AWARD OF THE DEGREE

MASTER OF COMPUTER APPLICATIONS (MCA)
OF
MAHATMA GANDHI UNIVERSITY, KOTTAYAM
BY

NANDHU JAYAN
Reg No: 22PMC139

MARIAN COLLEGE
KUTTIKKANAM
(AUTONOMOUS)

MAKING COMPLETE

Marian College Kuttikanam Autonomous
Peermade, Kerala – 685 531

2023

A Project Report on

SOLILOQUY

Live your life

SUBMITTED IN PARTIAL FULFILMENT OF REQUIREMENT
FOR THE AWARD OF THE DEGREE

MASTER OF COMPUTER APPLICATIONS (MCA)
OF
MAHATMA GANDHI UNIVERSITY, KOTTAYAM

By
NANDHU JAYAN
Reg No: 22PMC139

Under the guidance of

Mr. Satheesh kumar S
Assistant Professor
PG Department of Computer Applications
Marian College Kuttikkanam Autonomous

MARIAN COLLEGE
KUTTIKKANAM
(AUTONOMOUS)

MAKING COMPLETE

Marian College Kuttikanam Autonomous
Peermade, Kerala – 685 531
2023

PG DEPARTMENT OF COMPUTER APPLICATIONS
Marian College Kuttikkanam Autonomous
MAHATMA GANDHI UNIVERSITY, KOTTAYAM
KUTTIKKANAM – 685 531, KERALA.

CERTIFICATE

This is to certify that the project work entitled

soliloquy

is a bonafide record of work done by

NANDHU JAYAN

Reg No: 22PMC139

In partial fulfilment of the requirements for the award of Degree of
MASTER OF COMPUTER APPLICATIONS [MCA]
During the academic year 2022-2023

Mr. Satheesh Kumar S
Assistant Professor
PG Department of Computer Applications
Marian College Kuttikkanam Autonomous

Mr. Win Mathew John
Head of the Department
PG Department of Computer Applications
Marian College Kuttikkanam Autonomous

External Examiner

External Examiner

Acknowledgements

First of all, I thank the “God Almighty” for his immense grace and blessings in my life and at each stage of my project work

I express my sincere gratitude to Prof Dr Ajimon George, Principal, Marian College Kuttikkanam (Autonomous), Dr. Mendus Jacob, Director, PG Department of Computer Applications for the support given throughout the project work.

I extend my gratitude to Mr. Win Mathew John, HoD, PG Department of Computer Applications, who is a constant source of inspiration and whose advice helped me to complete this project work successfully.

I express my deep sense of gratitude to my project mentor, MR. ROBINS A KATTOOR, Associate Professor, PG Department of Computer Applications, for his profound guidance for the successful completion of this project work.

With great enthusiasm, I express my gratitude to all the faculty members of the PG Department of Computer Applications for their timely help and support.

Finally, I express my deep appreciation to all my friends and family members for the moral support and encouragement they have given to complete this project work successfully.

NANDHU JAYAN

ABSTRACT

Soliloquy is a Django-based social media website project called Soliloquy, designed specifically to provide support and a safe space for individuals experiencing sadness and depression in their lives. The platform aims to create a supportive online community where users can find solace, connect with like-minded individuals, share their thoughts and feelings, and access valuable resources to aid in their journey towards emotional well-being.

Soliloquy focuses on providing a compassionate and empathetic environment, where users can openly express their emotions and find support from others who can relate to their experiences. Through a combination of interactive features and curated content, the platform aims to foster a sense of belonging and encourage positive interactions.

Key features of Soliloquy include user registration and authentication, personalized user profiles, a variety of communication tools, such as private messaging and public forums, and a resource section containing articles, blogs, and professional guidance related to mental health and well-being.

By leveraging the Django framework, Soliloquy ensures efficient data management and user authentication, contributing to a secure and reliable user experience. Django's robust ORM capabilities facilitate seamless integration with the database, allowing for efficient storage and retrieval of user-generated content. Additionally, the project utilizes Django's templating system and URL routing to create a user-friendly and responsive web interface.

Soliloquy aims to empower individuals who are sad or depressed by providing them with a platform where they can find understanding, support, and encouragement from a community of individuals who have experienced similar challenges. The project's ultimate goal is to reduce the sense of isolation often associated with these emotions and promote mental well-being through connections and resources.

In conclusion, Soliloquy is a Django-based social media website project designed to provide a safe and supportive space for individuals facing sadness and depression. By combining interactive features, curated content, and a compassionate community, Soliloquy aims to empower users to express themselves, find support, and access valuable resources on their path to emotional well-being.

TABLE OF CONTENTS

Chapter			Page No
1	Introduction		1
	1.1	Problem Statements	2
	1.2	Proposed System	2
	1.3	Features of the Proposed System	3
2	Functional Requirements		4
3	Non-Functional Requirements		7
4	Features and Highlights		9
5	Technical Aspects		13
6	Challenges		18
7	Future Enhancement		19
8	Conclusion		21
9	References		23
Annexure A	Screen Shots		25

1.INTRODUCTION

1. INTRODUCTION

Soliloquy is a social media website project developed to provide a supportive online community for individuals who are feeling sad and depressed in life. It aims to create a safe space where users can express their emotions, connect with like-minded individuals, and find solace and understanding in their struggles.

In today's fast-paced and interconnected world, many people face feelings of sadness and depression. However, it can be challenging to find the right platform or community that offers understanding, empathy, and constructive support. Soliloquy aims to bridge that gap by providing a dedicated space where individuals can share their thoughts, experiences, and emotions without fear of judgment.

The project leverages the power of technology, specifically the Django web framework, to create an intuitive and user-friendly social media platform. It offers a range of features to facilitate communication, engagement, and personal growth within the community. From creating user profiles and sharing posts to connecting with friends and engaging in private conversations, Soliloquy provides a comprehensive set of tools to foster meaningful interactions.

1.1 PROBLEM STATEMENTS

Lack of Supportive Online Communities:

Many individuals experiencing sadness and depression struggle to find online communities that provide a supportive and understanding environment. Existing social media platforms may not adequately address the unique needs of these individuals, leaving them feeling isolated and unheard.

Limited Opportunities for Expression:

People facing sadness and depression often encounter difficulties expressing their emotions openly. Mainstream social media platforms may not provide a safe space for individuals to share their thoughts and experiences without fear of judgment or criticism.

Inadequate Mental Health Resources:

Access to comprehensive mental health resources and support can be limited. Many individuals may not know where to turn for guidance, assistance, or information related to

their emotional well-being. Existing online platforms may not offer specific resources tailored to their needs.

1.2 PROPOSED SYSTEM

The proposed system for Soliloquy is a social media website built on the Django web framework, designed specifically to address the needs of individuals experiencing sadness and depression. It offers a supportive and empathetic online community where users can connect, share their thoughts and emotions, and access valuable mental health resources. Here are the key components of the proposed system:

User Registration and Profiles:

Users can create accounts and set up personal profiles that reflect their interests and preferences.

Profile information may include a bio, profile picture, and relevant details related to their emotional well-being journey.

Post Creation and Sharing:

Users can create posts to express their thoughts, emotions, or experiences.

They can share these posts publicly with the community or restrict them to a specific group of friends.

Interactions and Engagement:

Users can interact with posts by liking, commenting, and sharing them.

These interactions promote engagement and foster supportive conversations within the community.

Friend Connections:

Users can connect with others by sending and accepting friend requests.

Friend connections enable private messaging and provide a closer support network within the platform.

Private Messaging:

Users can have private conversations with their friends, allowing for more personal and direct support.

Private messaging provides a confidential space for users to share their feelings and seek advice.

1.3 CLIENT REQUIREMENTS

1. User can Register and login
2. User can view the home page
3. User can view the post view page
4. User can add post
5. User can like and comment on the post.
6. User can save others post.
7. User can sent friend reuqest.
8. User can follow others.
9. User can approve friend request
10. User can sent messages to friends.
11. User can watch who all are the friends of the other user.
12. User can recieve messages
13. User can logout
14. Users can view the liked post.
15.AI Chatbot

1.4FEATURES OF THE PROPOSED SYSTEM

User Registration and Authentication:

- Users should be able to create an account and log in securely.
- Implement authentication mechanisms to ensure user privacy and security.

User Profiles:

- Each user should have a profile page that displays their basic information and activity.
- Users can update their profile information and add a profile picture.

Post Creation and Sharing:

- Users can create and publish posts containing text.
- Implement a user-friendly interface for composing and formatting posts.
- Enable users to share their posts with the community or limit the visibility to specific audiences.

Interactions and Engagement:

- Users can like, comment on posts created by others.
- Implement real-time notifications to inform users about interactions on their posts.
- You can save post.

Friend Connections:

- Users can send friend requests and accept/reject requests from others.
- Implement a system to manage friend connections and display a user's list of friends.

Messaging System:

- Users can send private messages to their friends.
- Implement a real-time messaging system to facilitate private conversations.

Responsive Design:

- Develop a responsive and mobile-friendly design to ensure accessibility across different devices.

Chatbot feature:

- You can chat with chat bot.

2.FUNCTIONAL REQUIREMENTS

2. FUNCTIONAL REQUIREMENTS

User Registration and Authentication:

Users should be able to create an account and provide necessary registration information.
The system should authenticate users securely to ensure the privacy and security of their accounts.

User Profile Management:

Users should be able to create and manage their profiles, including adding a profile picture, bio, and relevant details related to their emotional well-being journey.
The system should allow users to edit and update their profile information.

Post Creation and Sharing:

Users should be able to create posts to express their thoughts, emotions, or experiences.
The system should provide options for users to control the visibility of their posts (public or restricted to friends).

Interactions and Engagement:

Users should be able to like, comment on, and share posts created by other users.
The system should display the number of likes, comments, and shares for each post and allow users to interact with them.

Friend Connections:

Users should be able to send friend requests to other users and accept or decline incoming friend requests.
The system should manage and display the user's friend list and allow for easy navigation to friends' profiles.

Private Messaging:

Users should be able to send private messages to their friends within the platform.
The system should facilitate secure and real-time messaging between users, including features like message notifications.

Notifications:

The system should send real-time notifications to users for activities such as new friend requests, post interactions, or important updates.
Notifications should be configurable and allow users to manage their notification preferences.

Search Functionality:

The system should provide a search feature that allows users to search for specific posts, users, or topics of interest.
The search functionality should be efficient and provide relevant results based on user queries.

Responsive Design:

The system should be designed with a responsive layout to ensure optimal user experience across different devices and screen sizes.
The interface should adapt and provide a consistent user experience on desktops, tablets, and smartphones.

Admin Panel:

The system should have an admin panel to manage user accounts, moderate content, and handle system configurations.
The admin panel should provide necessary functionalities for efficient moderation and administration of the platform.

3.NON-FUNCTIONAL REQUIREMENTS

3.NON-FUNCTIONAL REQUIREMENTS

1. AVAILABILITY

The Intercollege System shall be available to users 24 hours a day, 7 days a week, with the exception of being down for maintenance no more than one hour a week. If the system crashes, it should be back up within one hour.

2. RELIABILITY

The average time to failure shall be 30 days. In the event that a server does crash, a backup server will be up and running within the hour.

3.RESPONSE TIME

Each page will load within 2 seconds.

4. SECURITY

Users will be able to access only their own personal information and not that of other users.

5. MAINTAINABILITY

Any updates or defect fixes shall be able to be made on server-side co computers only without any patches required by the user.

6. EFFICIENCY REQUIREMENT:

Improves user accessibility and time flexibility by engaging learners in a learning process.

7. RELIABILITY REQUIREMENT

The system will provide a reliable environment to both teacher and student.

8. USABILITY REQUIREMENT

The website is designed for user friendly environment and ease of use.

3.1 Third-Party Libraries:
Install requirements

```
Package            Version
------------------ ---------
agora-token-builder 1.0.0
asgiref            3.4.1
attrs              21.2.0
autobahn           21.3.1
Automat            22.10.0
certifi            2021.5.30
cffi               1.14.6
channels           3.0.4
charset-normalizer  2.0.4
constantly         15.1.0
cryptography       3.4.7
daphne             3.0.2
defusedxml         0.7.1
Django             4.0.7
django-allauth     0.45.0
django-ckeditor    6.1.0
django-cleanup     5.2.0
django-crispy-forms 1.12.0
django-jazzmin     2.6.0
django-js-asset    1.2.2
hyperlink          21.0.0
idna               3.2
incremental        21.3.0
oauthlib           3.1.1
Pillow             9.0.1
pip                21.3.1
psycopg2           2.9.1
pyasn1             0.4.8
pyasn1-modules     0.2.8
pycparser          2.20
PyJWT              2.4.0
pyOpenSSL          20.0.1
python-dotenv      0.19.0
python3-openid     3.2.0
pytz               2021.1
requests           2.26.0
requests-oauthlib  1.3.0
```

```
service-identity    21.1.0
setuptools          60.2.0
six                 1.16.0
sqlparse            0.4.2
Twisted             22.4.0
twisted-iocpsupport 1.0.1
txaio               21.2.1
typing-extensions   3.10.0.0
tzdata              2022.1
urllib3             1.26.6
wheel               0.37.1
zope.interface      5.4.0
```

3. Architecture of your project

Model:

The Model represents the data and the business logic of the application.
In Soliloquy, the models define the structure of the database tables and handle interactions with the database.
Models may include entities such as User, Post, Comment, Friendship, Message, etc.
Django's ORM (Object-Relational Mapping) is utilized to map the models to database tables and handle database operations.

View:

The View is responsible for rendering the user interface and presenting the data to the users.
In Soliloquy, views handle the logic for processing user requests and generating appropriate responses.
Views interact with the models to retrieve data and pass it to the templates for rendering.
Django's views are typically implemented as Python functions or classes.
Controller (or URL Routing):
The Controller handles the routing of incoming requests to the appropriate views.
In Soliloquy, Django's URL routing system maps URLs to specific views or viewsets.
URL patterns are defined in the project's URL configuration, specifying which view should handle a particular URL.

Templates:

Templates are responsible for generating dynamic HTML pages based on the data received from the views.
In Soliloquy, templates define the structure and layout of the web pages.
Django's template engine is used to process templates and fill them with data before sending them as responses to the users.

Static Files:

Static files such as CSS, JavaScript, and images are used to enhance the user interface and add interactivity.Soliloquy stores these static files separately and serves them directly without processing by the views or templates.

Middleware:

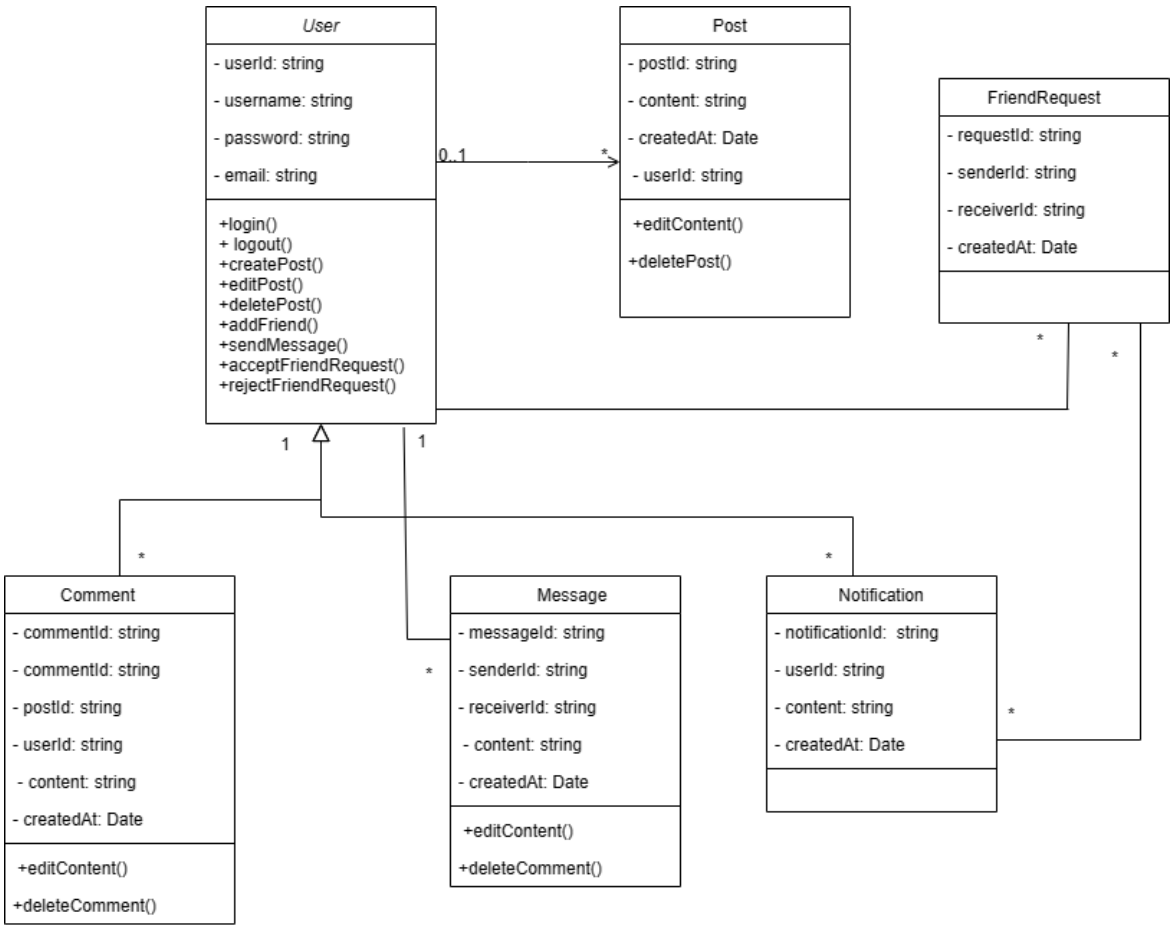
Middleware components intercept and process requests and responses globally.
In Soliloquy, middleware can be used for tasks such as authentication, session management, logging, and error handling.
Django provides built-in middleware and allows for custom middleware implementation.

Additional Components:

Soliloquy may incorporate additional components based on specific requirements.
For example, real-time messaging functionality can be implemented using Django Channels or other WebSocket frameworks.

4.UML DIAGRAM

4.1 CLASS DIAGRAM



5.Challenges faced during the development

Challenges faced during the development

During the development of the Soliloquy project, several challenges may be encountered. Here are some potential challenges that developers may face:

User Privacy and Data Security:

Implementing robust security measures to protect user data and ensure privacy can be challenging.

Addressing potential vulnerabilities, preventing data breaches, and complying with data protection regulations require careful planning and implementation.

Emotional Well-being Considerations:

Developing a platform for individuals experiencing sadness and depression requires sensitivity and empathy.

Striking a balance between providing support and guidance while avoiding potential triggers or harmful content is crucial.

Community Management and Content Moderation:

Building and maintaining a safe and supportive community necessitates effective content moderation.

Developing mechanisms to identify and handle offensive or inappropriate content, and dealing with user conflicts, can be challenging.

Scalability and Performance:

As the user base grows, ensuring the platform can handle increasing traffic and maintain acceptable performance becomes crucial.

Optimizing database queries, caching strategies, and employing scalability practices are essential for a smooth user experience.

Real-time Interactions and Notifications:

Implementing real-time features like messaging, notifications, and updates can introduce complexities.

Ensuring reliable real-time communication, handling concurrency issues, and managing server resources effectively are key challenges.

Designing Intuitive User Interfaces:

Creating user-friendly interfaces that are intuitive and accessible is a constant challenge.

Balancing simplicity and functionality while providing a seamless user experience can require careful design and iteration.

Testing and Quality Assurance:
Ensuring the reliability and stability of the application through thorough testing is a significant challenge.
Developing comprehensive test suites, conducting unit tests, integration tests, and user acceptance testing are time-consuming but crucial steps.

Performance Optimization:
Optimizing the application's performance, including reducing page load times and minimizing database queries, can be challenging.
Profiling and identifying performance bottlenecks, employing caching strategies, and optimizing code and database queries are ongoing tasks.

Cross-browser Compatibility:
Ensuring the application functions correctly and displays consistently across different web browsers and devices can be challenging.
Addressing browser-specific issues, implementing responsive design, and conducting thorough compatibility testing are necessary steps.

Documentation and Maintenance:
Creating comprehensive and up-to-date documentation is essential for project sustainability and future maintenance.
Ensuring that documentation remains in sync with code changes and providing clear guidelines for future development and troubleshooting can be challenging.

6.FUTURE ENHANCEMENT

Machine Learning-Based Sentiment Analysis:
Enhance the sentiment analysis feature by incorporating machine learning algorithms.
Train models to better understand and analyze user sentiments, providing more accurate support and intervention suggestions.

Advanced Community Moderation Tools:
Implement advanced content moderation tools to efficiently detect and handle inappropriate or harmful content.
Explore automated moderation techniques using machine learning to augment human moderation efforts.

Integration with Mental Health Professionals:
Collaborate with mental health professionals to provide expert guidance and resources within the platform.
Integrate features like virtual therapy sessions or access to mental health resources directly from Soliloquy.

Support Groups and Forums:
Create support groups or forums where users can connect and engage in discussions on specific topics related to sadness, depression, and emotional well-being.
Enable users to create or join groups based on their interests or specific challenges they are facing.

Personalized Recommendations:
Develop personalized recommendation systems that suggest relevant content, resources, or connections based on user preferences, interests, and browsing history.
Provide users with tailored content that aligns with their emotional well-being journey.

Mobile Application Development:
Build mobile applications for Soliloquy to offer a more convenient and accessible user experience on smartphones and tablets.
Ensure seamless synchronization and data consistency between the web platform and mobile applications.

Support for Multiple Languages:
Expand the platform's language support to cater to a wider audience.
Provide localization options to make Soliloquy accessible to users from different regions and cultures.

Gamification and Progress Tracking:
Introduce gamification elements to engage and motivate users in their emotional well-being journey.
Implement features that allow users to track their progress, set goals, and earn rewards for achieving milestones.

Collaborative Storytelling and Creative Expression:
Enable users to collaborate on creative projects, such as storytelling, poetry, or artwork, to foster creativity and self-expression.
Provide shared spaces where users can contribute to collective works or collaborate on creative endeavors.

Advanced Analytics and Insights:
Implement comprehensive analytics tools to gather insights into user behavior, engagement patterns, and platform usage.
Utilize data-driven insights to improve the platform, enhance user experience, and make informed decisions for future enhancements.

7.CONCLUSION

In conclusion, Soliloquy is a social media website project developed using Django, aimed at providing a safe and supportive community for individuals experiencing sadness and depression. The project focuses on creating a platform where users can connect, share their thoughts and emotions, and receive support from like-minded individuals.

The project incorporates essential features such as user profiles, post creation and sharing, interactions and engagement, real-time notifications, friend connections, private messaging, a resource section, and sentiment analysis. These features aim to foster a sense of community, provide a supportive environment, and offer valuable resources for users' emotional well-being.

Throughout the development process, various challenges may arise, including ensuring user privacy and data security, managing community moderation effectively, optimizing performance, and designing intuitive user interfaces. Overcoming these challenges requires careful planning, diligent implementation, continuous testing, and maintenance efforts.

Furthermore, the Soliloquy project has exciting future enhancement possibilities, such as integrating machine learning-based sentiment analysis, advanced moderation tools, collaboration with mental health professionals, personalized recommendations, mobile application development, and gamification elements. These enhancements can further enhance the platform's capabilities and improve the user experience.

Overall, Soliloquy aims to create a compassionate and empowering space for individuals facing sadness and depression, providing them with a supportive community and valuable resources to aid their emotional well-being journey.

8.REFERENCES

https://www.kommunicate.io/
https://chat.openai.com

https://testdriven.io/blog/django-channels/#:~:text=Django%20Channels%20%28or%20just%20Channels%29%20extends%20the%20built-in,MQTT%20%28IoT%29%2C%20chatbots%2C%20radios%2C%20and%20other%20real-time%20applications.

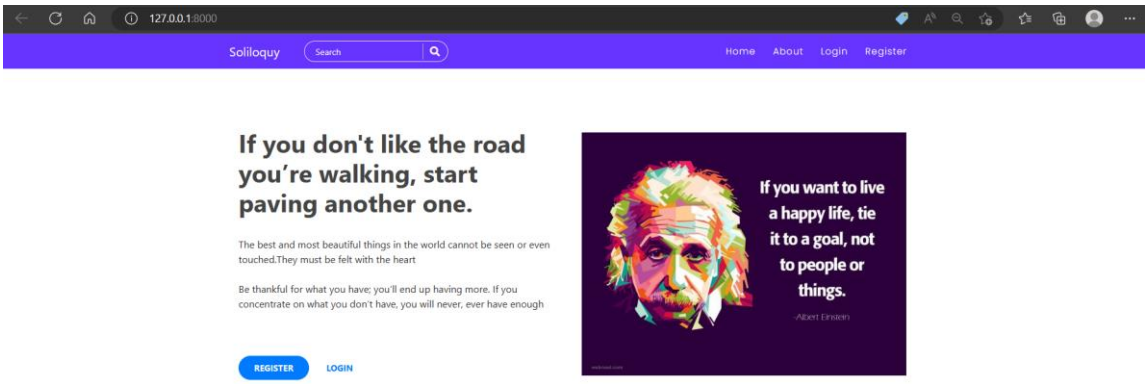
https://codewithstein.com/django-realtime-chat-app-tutorial-simple-django-tutorial-with-channels-and-redis/

(685) Build a Social Media App with Django – Python Web Framework Tutorial - YouTube

9.ANNEXURE

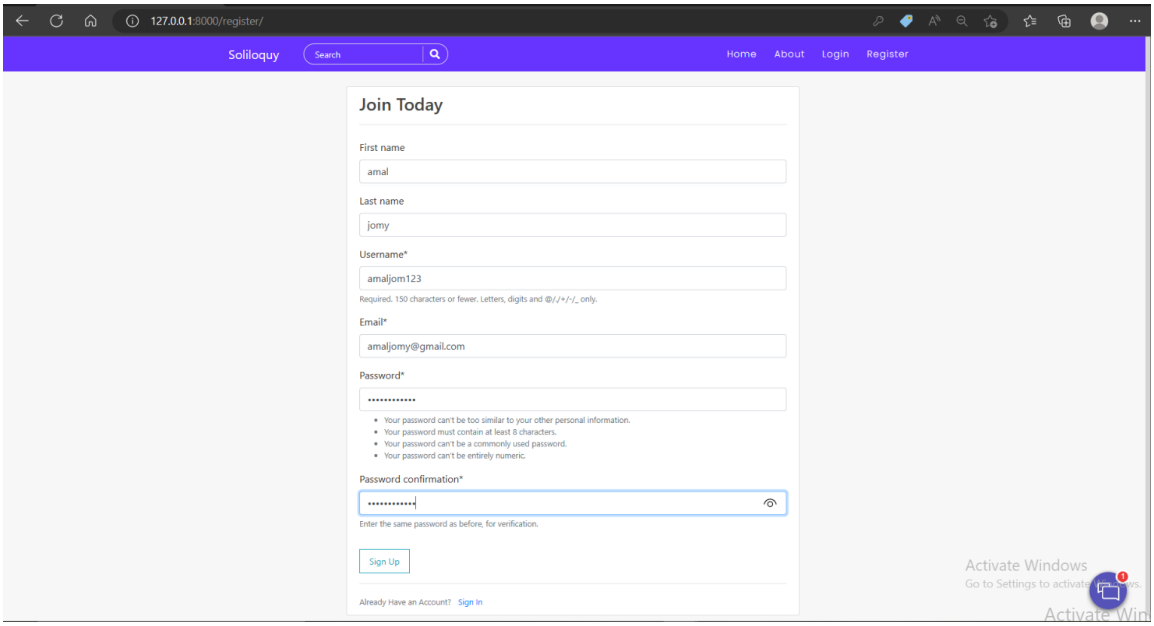
SCREENSHOTS

1.HOME PAGE



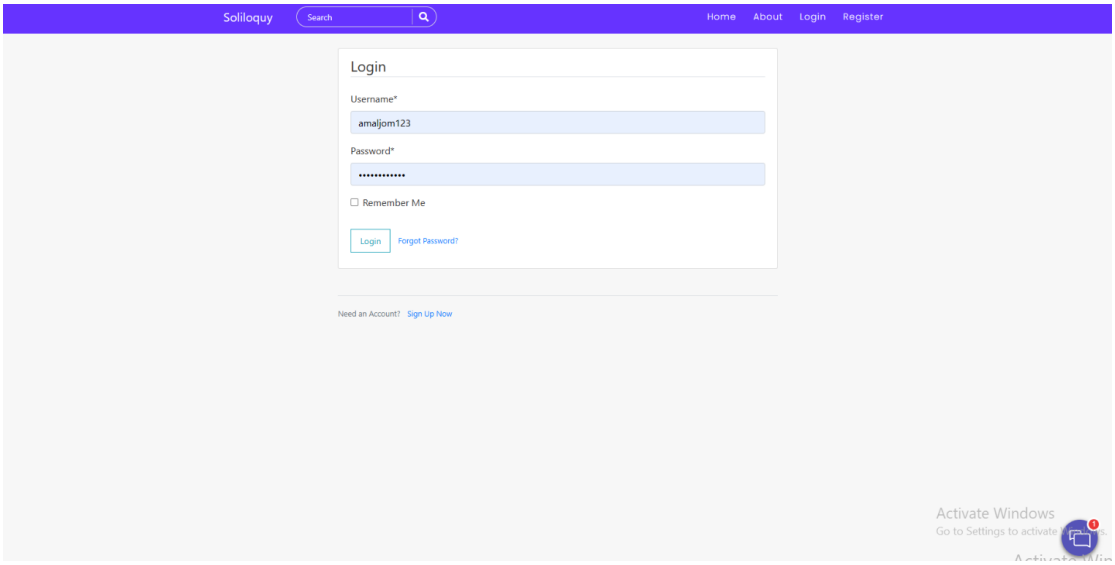
Activate Windows
Go to Settings to activate

2. Registration

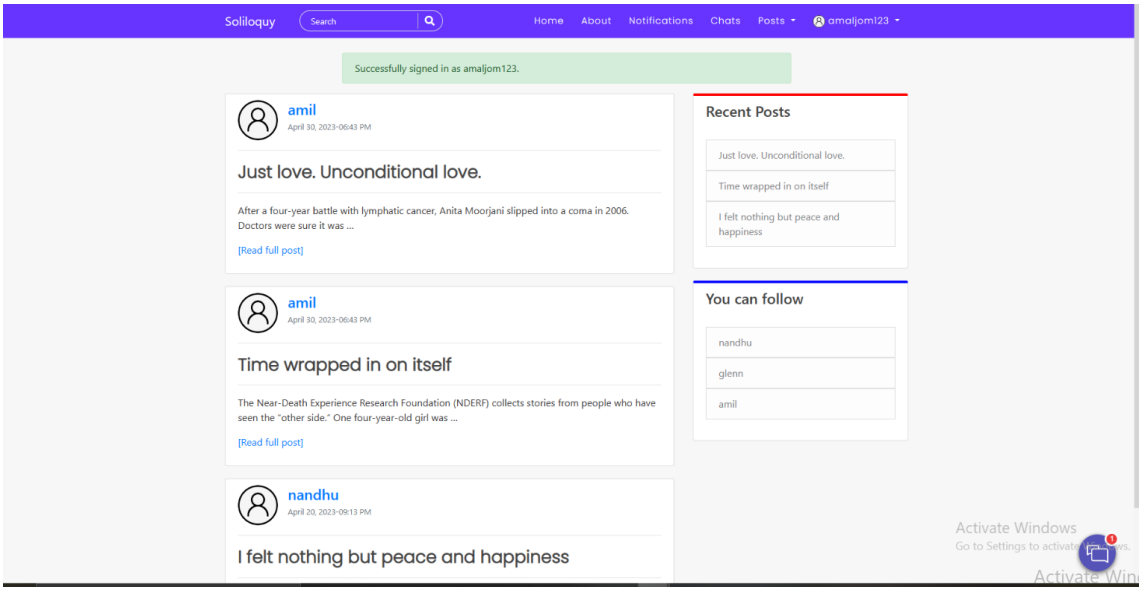


Activate Windows
Go to Settings to activate

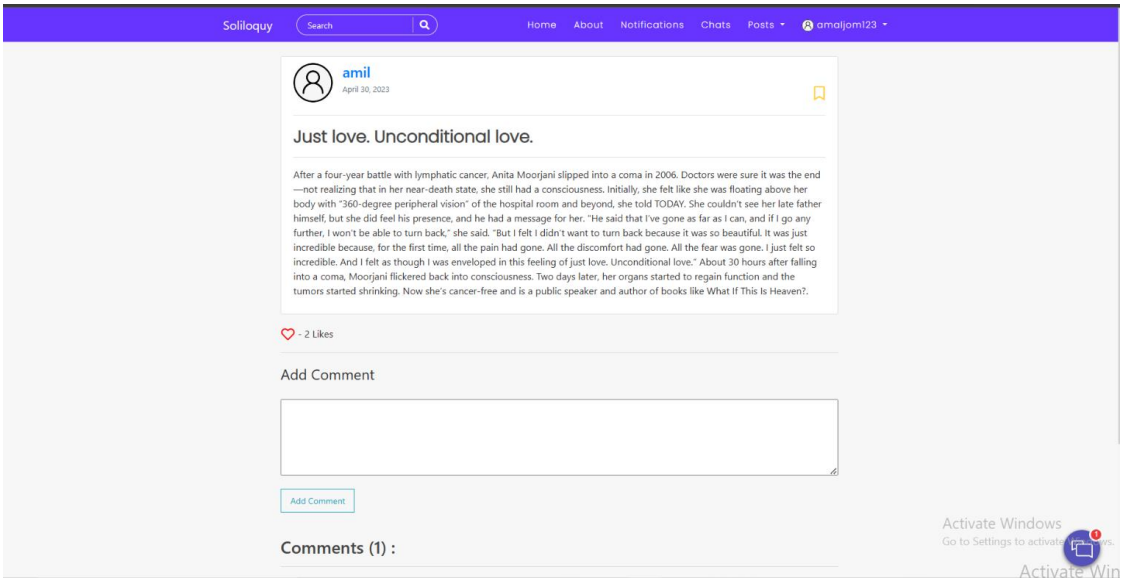
3.login page



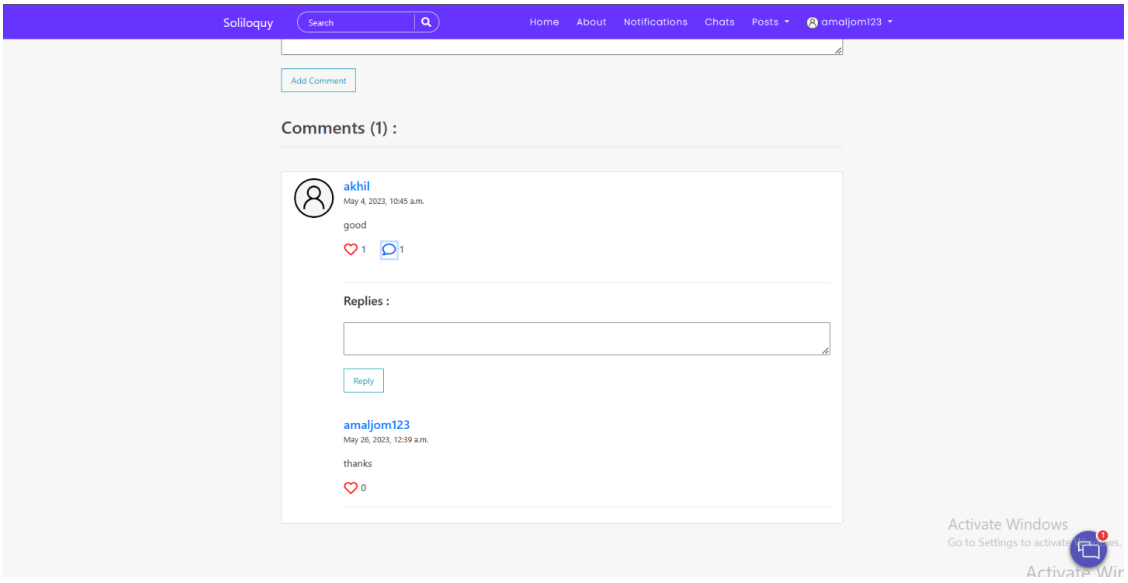
4.Feed page



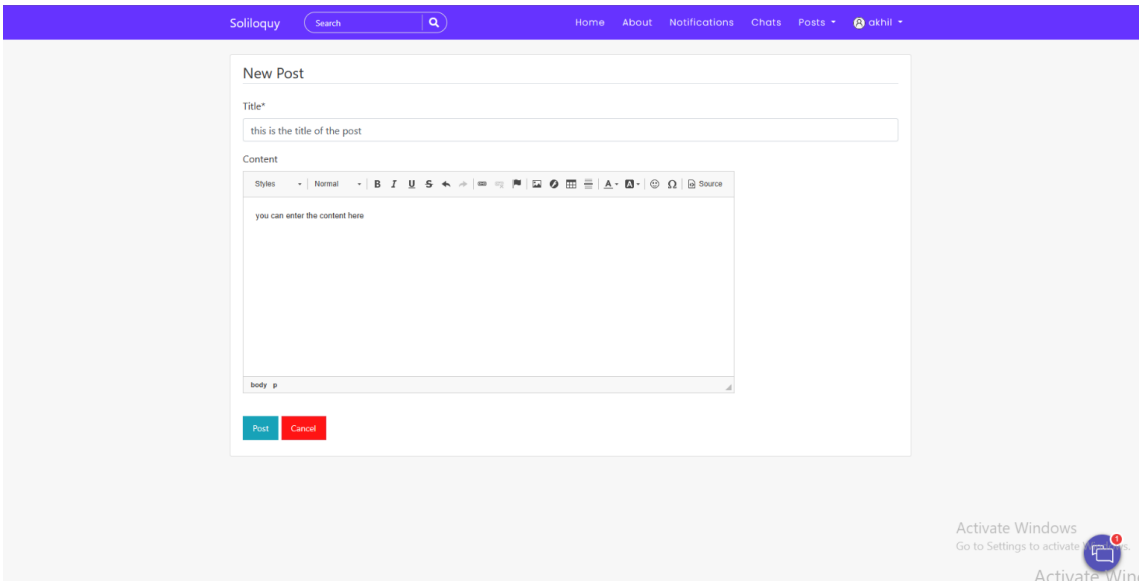
5.view post



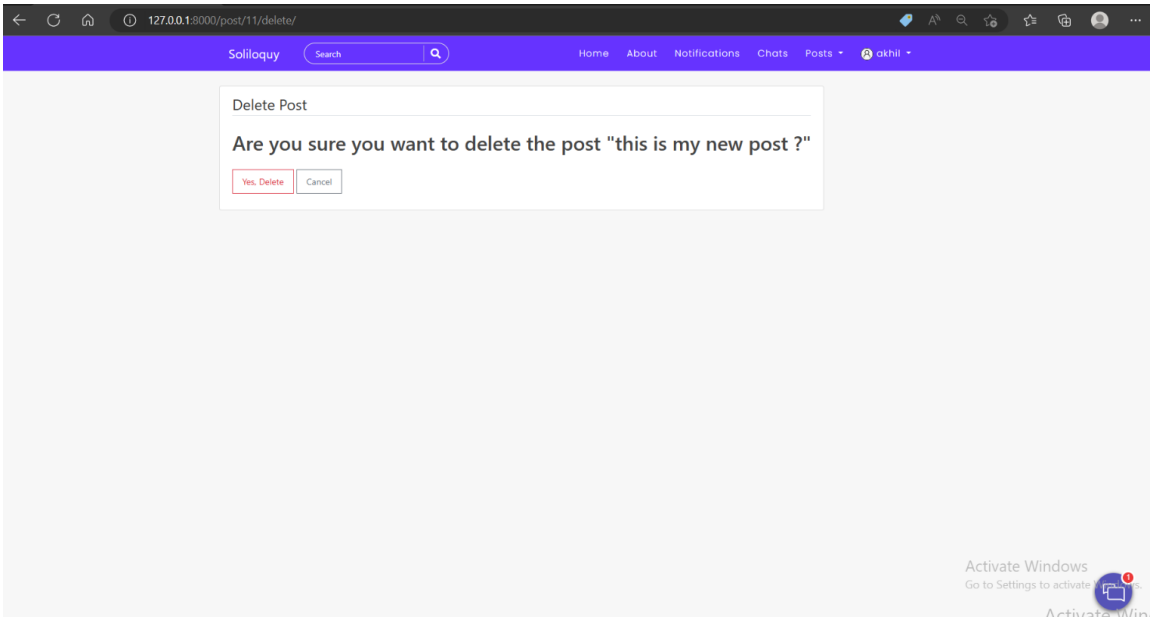
6. comment and like post



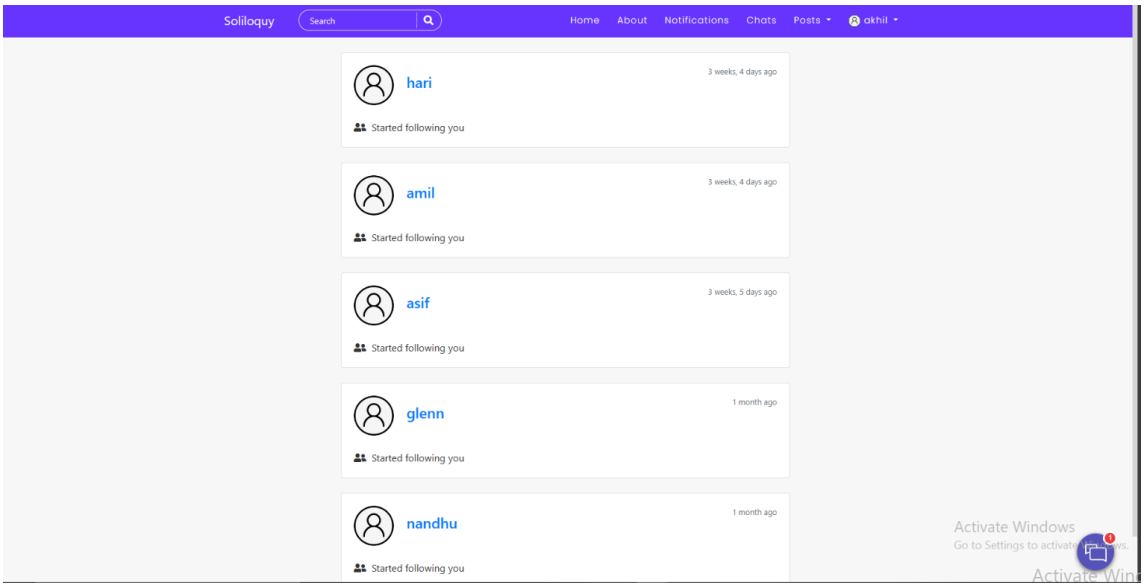
7.Edit post



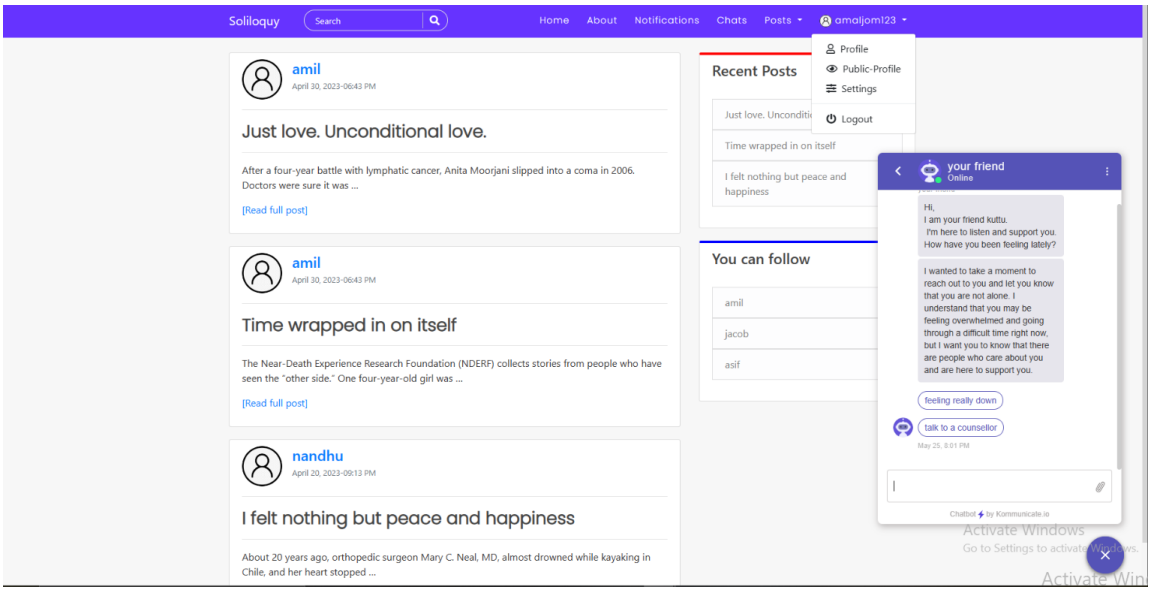
8.delete post



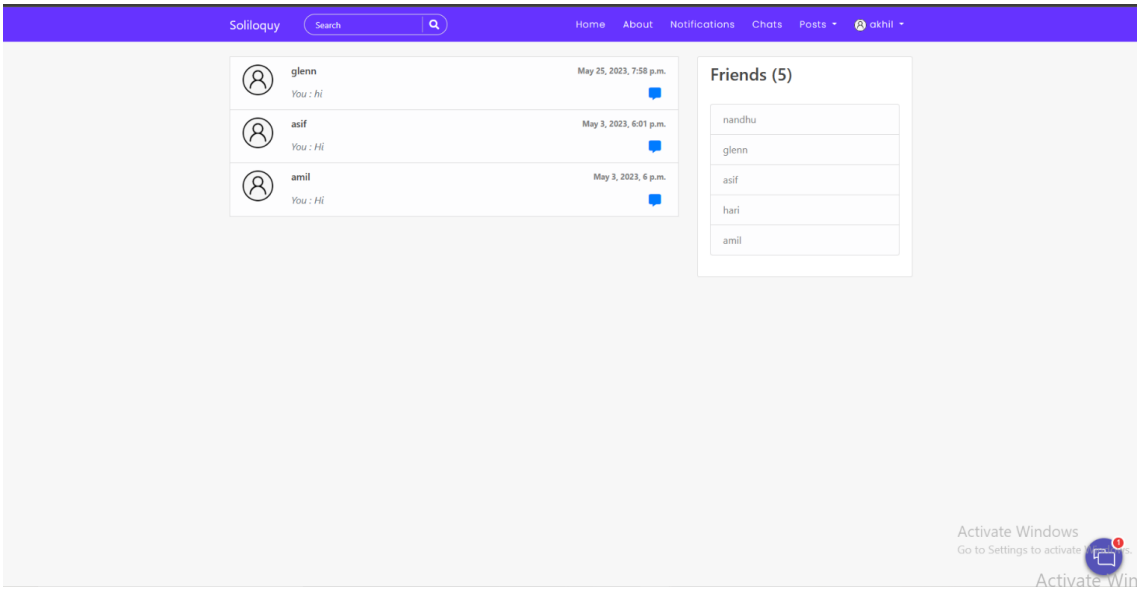
9.Notification page



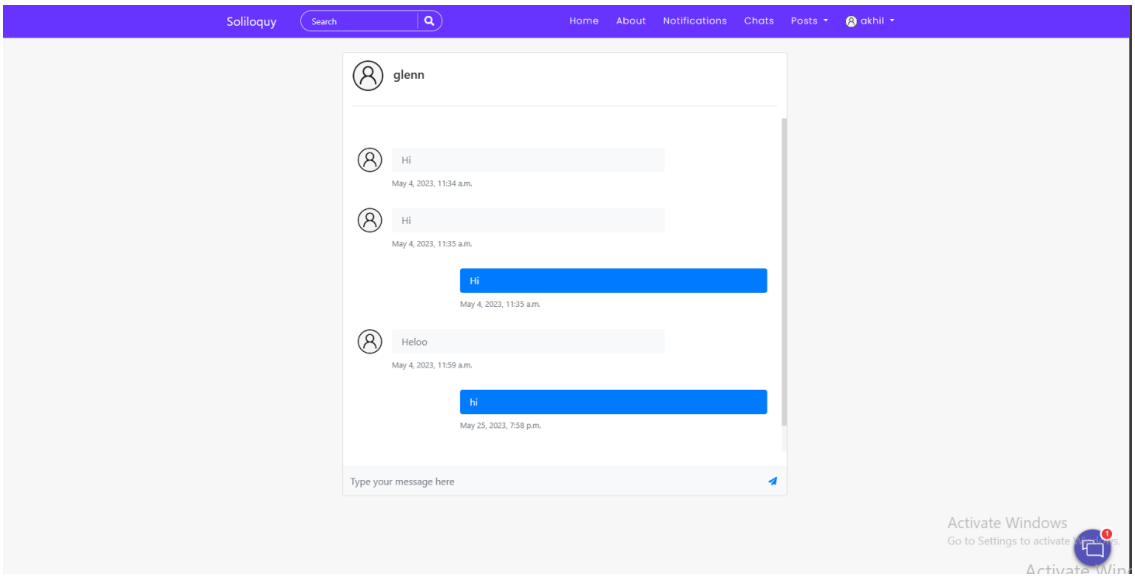
10. AI chatbot



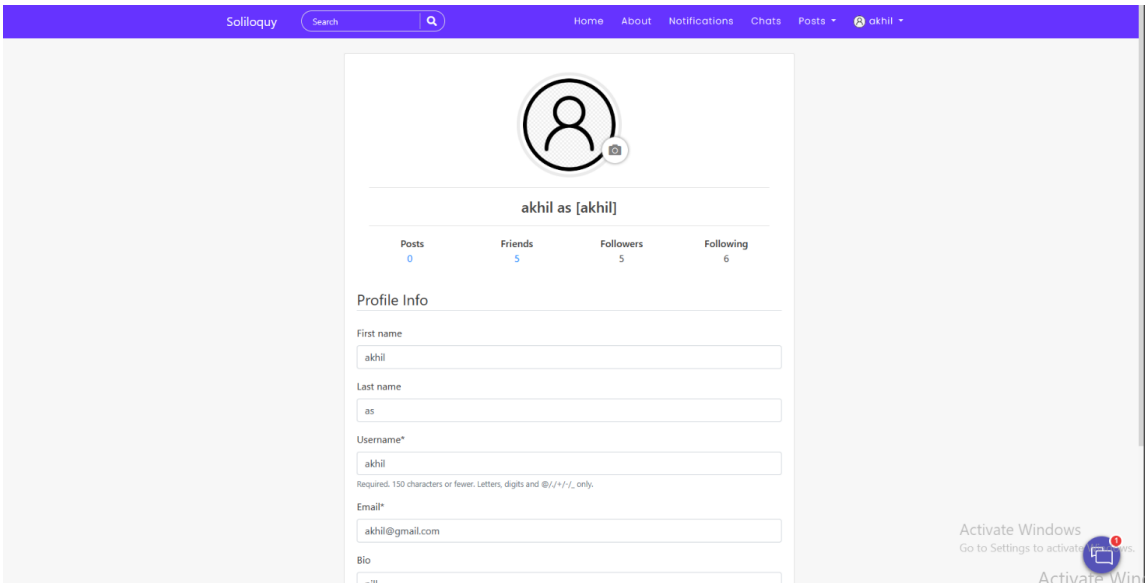
11. view message



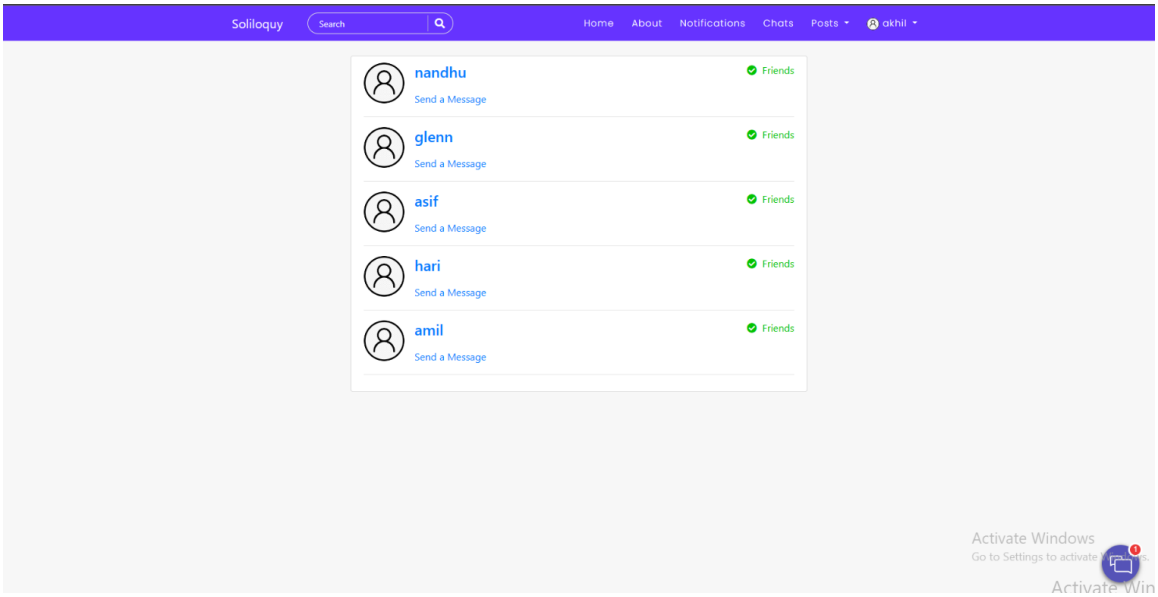
12. chat box



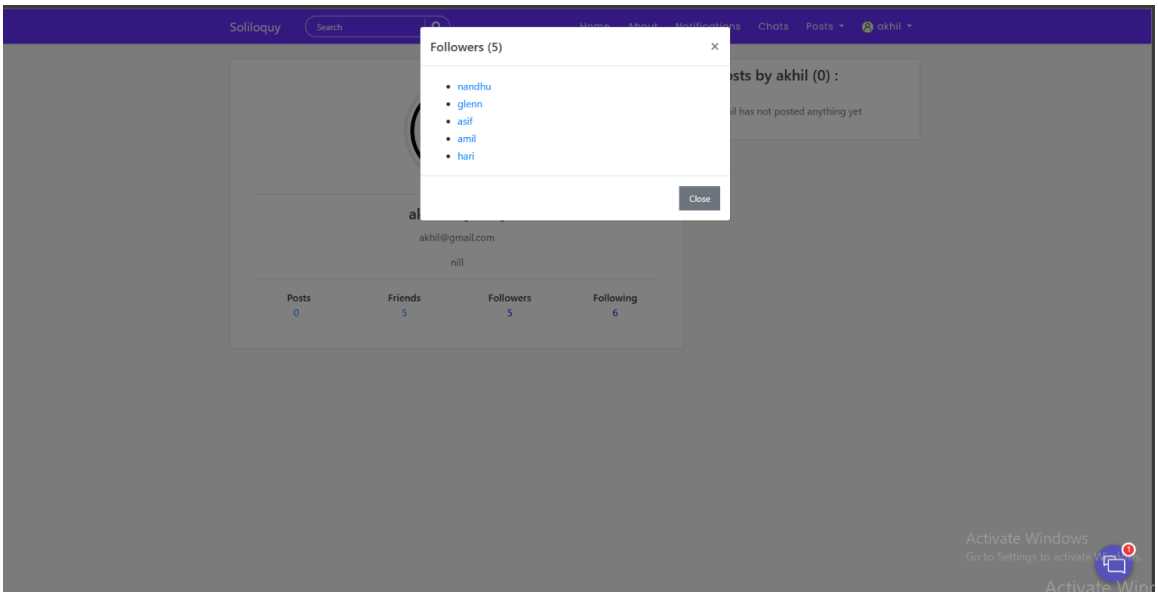
13.profile page



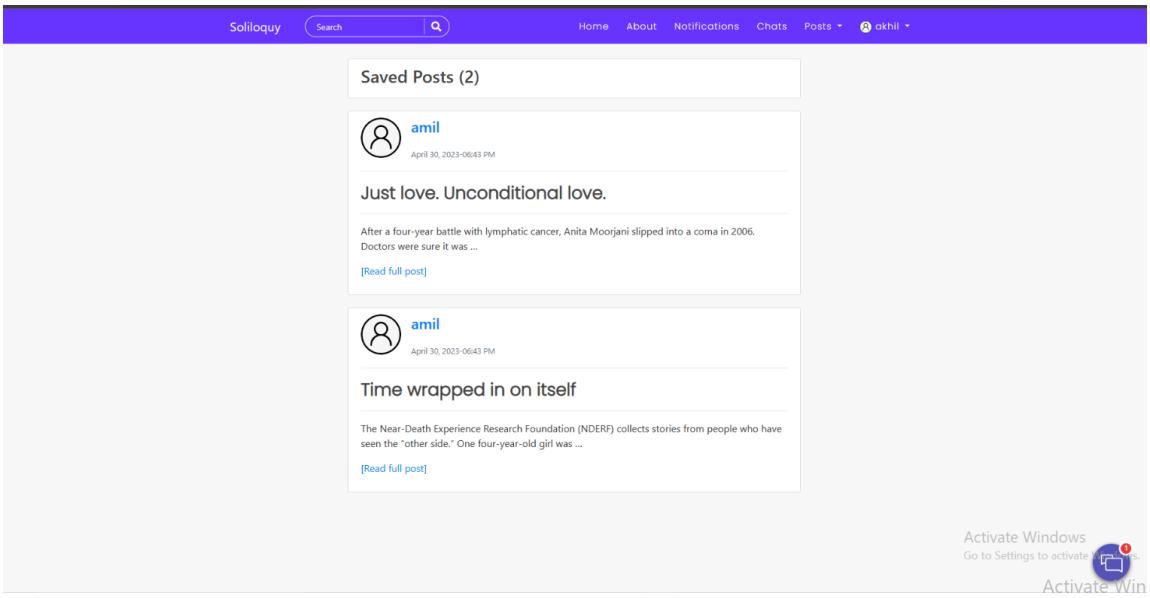
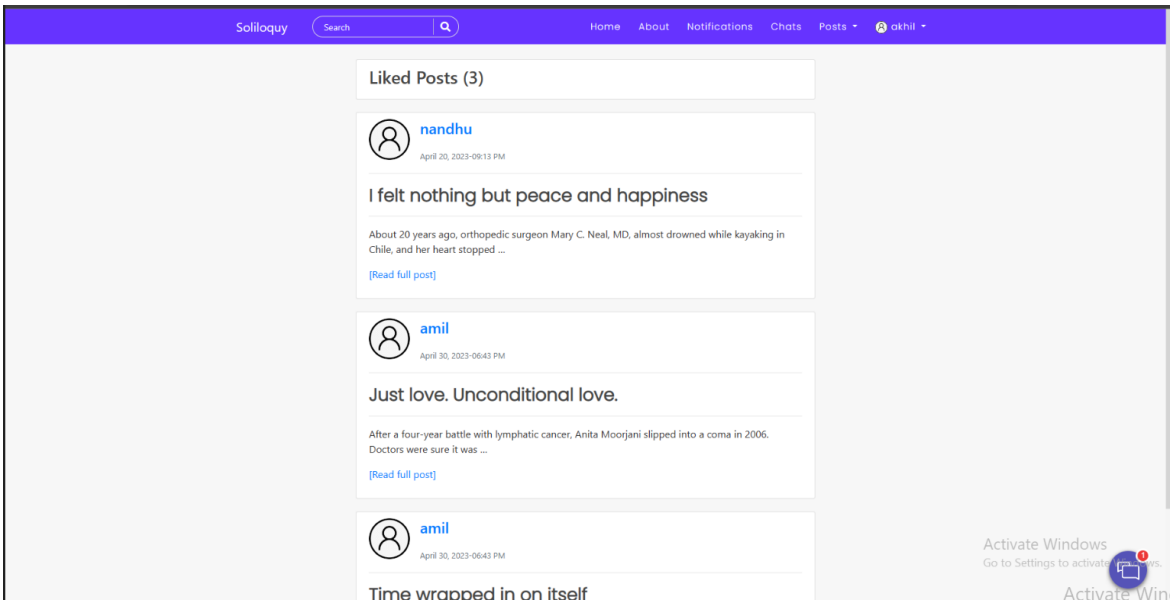
14. view friends



15.view followers



16.saved post



17. liked post