

# Book Recommendation System using Flask

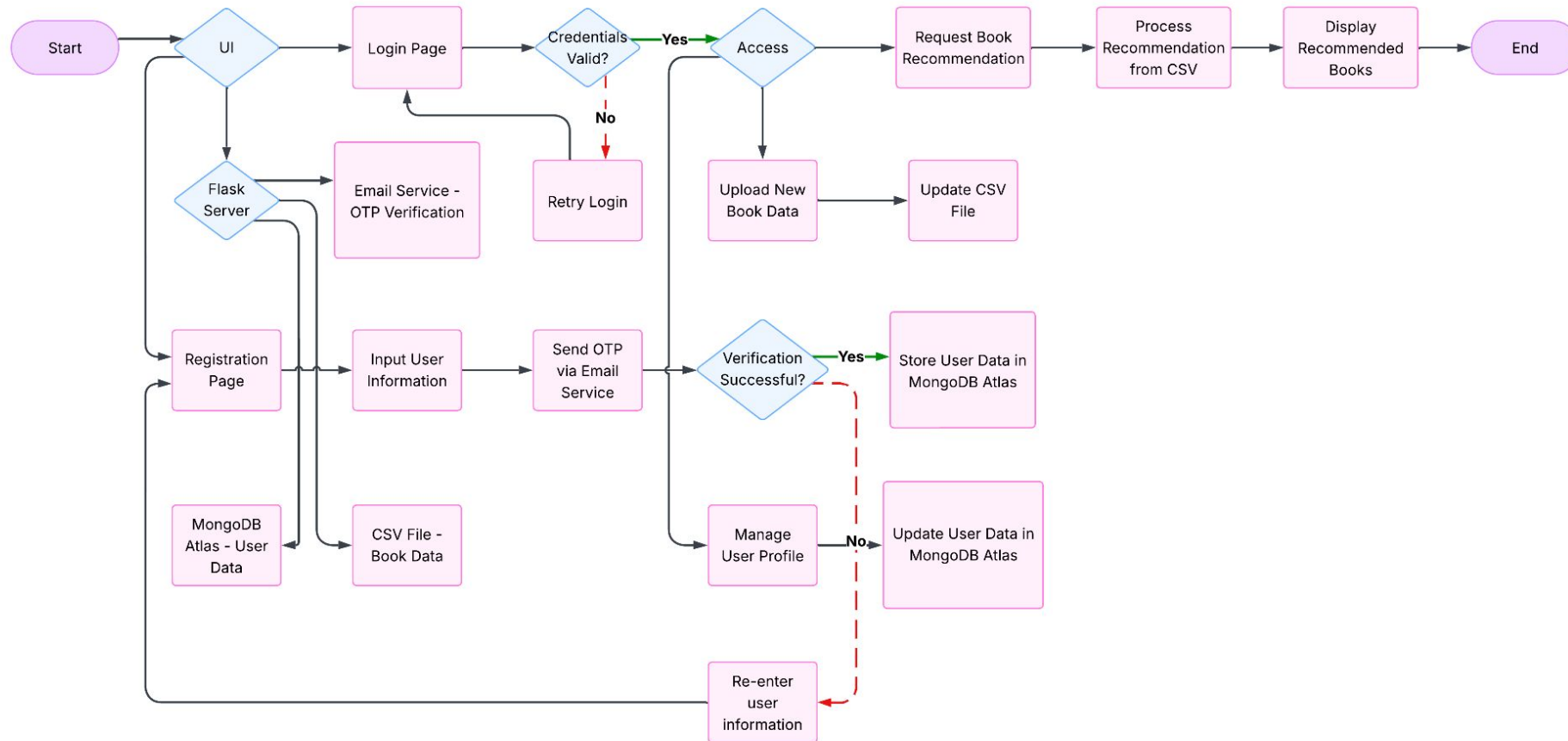
Personalized Book Suggestions • Flask • Machine Learning  
Shreyash Dhekane [D15A - Roll No.16]

# Introduction

- This project presents a Flask-based Book Recommendation System that aims to enhance the user experience by providing relevant book suggestions using a content-based filtering algorithm.
- The system uses metadata from books such as title, author, and publisher to identify similar titles and recommend them to the user.
- Along with book recommendation features, the platform also includes user authentication through MongoDB Atlas, and a secure registration process that incorporates OTP (One-Time Password) verification via email to ensure the authenticity of users.
- The platform is designed with a clean and responsive interface using Flask, and is integrated with machine learning models that utilize natural language processing (NLP) techniques to compute similarities between books.
- A structured database of books along with associated metadata and cover images is used for the recommendation engine. This system not only helps users discover books tailored to their tastes but also provides features like book upload, deletion, and a contact form for feedback.

# System Overview

- User registration and OTP-based login.
- Dashboard with personalized book search and recommendations.
- Collaborative filtering algorithm powered by cosine similarity.



# Implementation Screenshots

Good Old Friend [Home](#) [Login](#) [Register](#)

### Join Today

Username

Email

Password

Confirm Password

[Sign Up](#)

Already Have an Account? [Sign In](#)

Good Old Friend [Home](#) [Login](#) [Register](#)

### Log In

Email

Password

☐ Remember Me


[Login](#)

[Forgot Password?](#)

[Need an Account?](#) [Sign Up Now](#)

# Implementation Screenshots

Good Old Friend   Home   Upload Book   Delete Book   Contact Us   Recommender   Account   Logout



## SHREYASH DHEKANE

d2022.shreyash.dhekane@ves.ac.in

### Account Info

Username

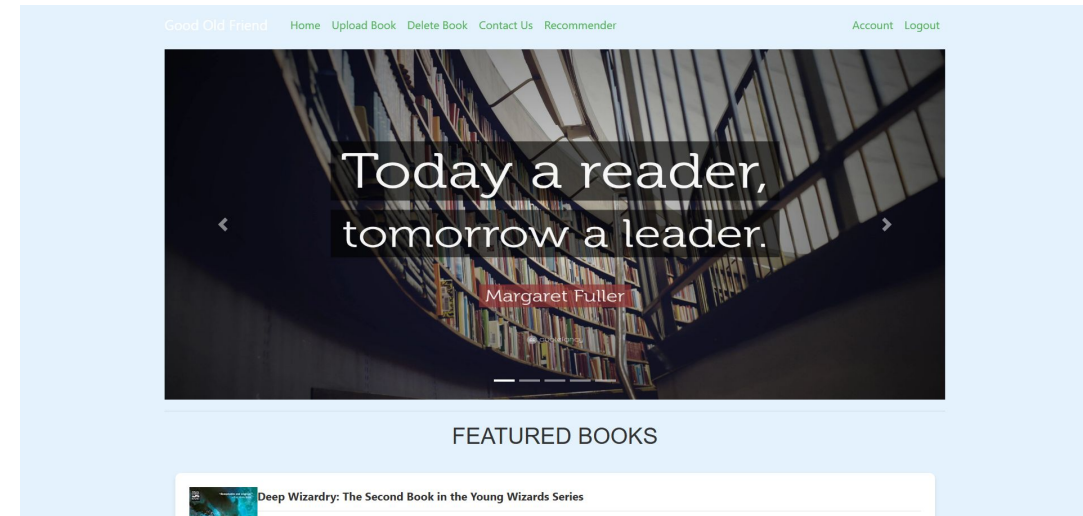
Shreyash Dhekane

Email

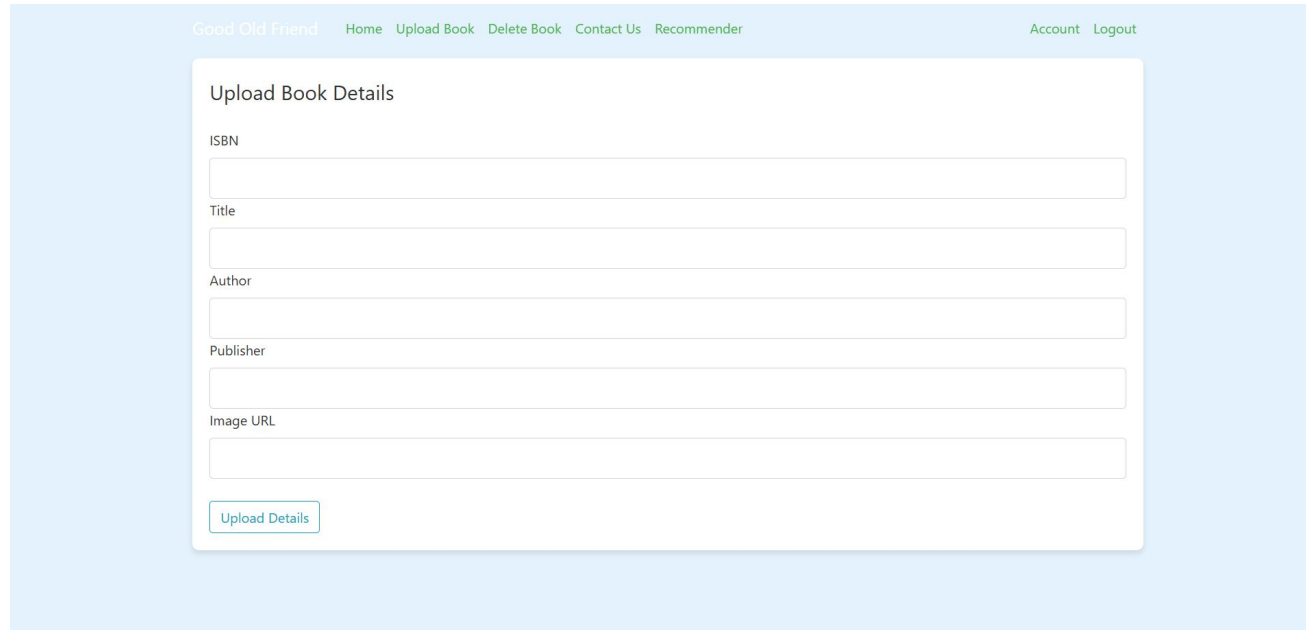
d2022.shreyash.dhekane@ves.ac.in

Update Profile Picture

No file chosen



# Implementation Screenshots



This screenshot shows the 'Upload Book Details' form on a light blue background. The form is a white box with a title 'Upload Book Details' at the top. It contains five input fields: 'ISBN', 'Title', 'Author', 'Publisher', and 'Image URL'. Each field is a simple white rectangle with a thin border. At the bottom of the form is a button labeled 'Upload Details'.

Good Old Friend   Home   Upload Book   Delete Book   Contact Us   Recommender   Account   Logout

### Upload Book Details

ISBN

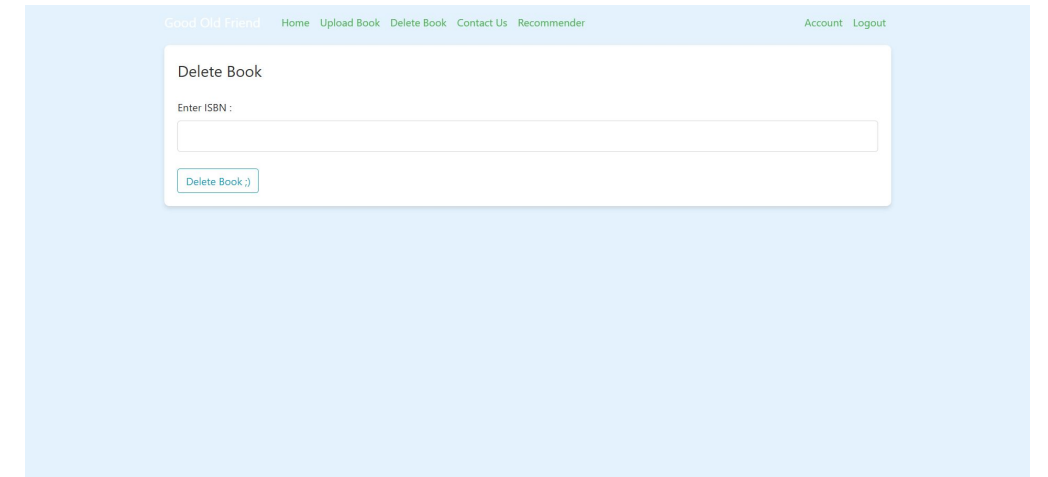
Title

Author

Publisher

Image URL

Upload Details



This screenshot shows the 'Delete Book' form on a light blue background. The form is a white box with a title 'Delete Book' at the top. Below the title is a label 'Enter ISBN :' followed by a single input field. At the bottom of the form is a button labeled 'Delete Book :)'. The background of the entire page is light blue.

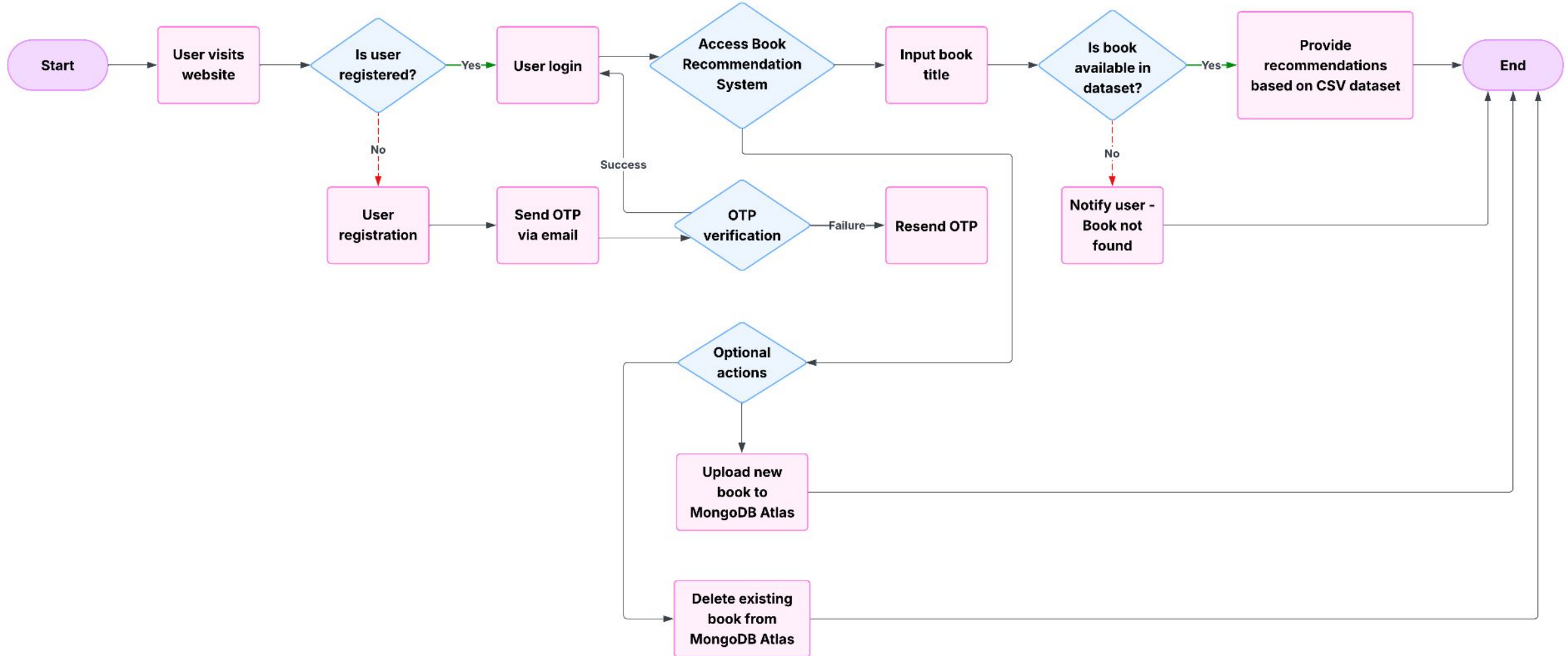
Good Old Friend   Home   Upload Book   Delete Book   Contact Us   Recommender   Account   Logout

### Delete Book

Enter ISBN :

Delete Book :)

# System Flow / UML



## Challenges Faced

1. Email verification and OTP delivery.
2. Sparse and inconsistent rating data.
3. Session management in Flask.
4. Speed of generating recommendations.

## Tech Stack

1. Frontend: HTML, CSS.
2. Backend: Python with Flask.
3. Database: MongoDB Atlas
4. Libraries: Pandas, Numpy, Scikit-learn, Flask-Mail.



# Future Scope

1. **Migrate fully to cloud DB (MongoDB Atlas):** Move the application's data storage from local SQLite to MongoDB Atlas for better scalability, security, and cloud accessibility.
2. **Add genre/author filtering with NLP:** Use Natural Language Processing to analyze book descriptions and enable recommendations based on genre and author preferences.
3. **Deploy on platforms like Heroku or AWS:** Host the web application on cloud platforms like Heroku or AWS to make it accessible online with robust uptime.
4. **Mobile app version using Flutter or React Native:** Develop a cross-platform mobile app using Flutter or React Native to reach users on Android and iOS devices.

# Conclusion

## **1. Provides relevant book recommendations in real-time:**

The system delivers personalized book suggestions instantly by analyzing user preferences and comparing them with others using collaborative filtering.

## **2. Simple user experience powered by smart backend:**

The user interface is intuitive and minimalistic, while the backend intelligently processes data and generates accurate recommendations seamlessly.

## **3. Foundation ready for full-scale deployment:**

With a working prototype in place, the system is well-structured and scalable, making it ready for enhancements and deployment to a production environment.