VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590 018

A Mini - Project Report

On

"DS Resource Hub"

Submitted in partial fulfillment of the requirements for the **MINI PROJECT (BCD586)** course of the 5th semester

Bachelor of Engineering

In

Computer Science & Engineering (DATA SCIENCE)

Submitted by

Ms. Pooja Shankar S
(4AI23CD036)

Ms. Spoorthi H
(4AI23CD053)

Ms. Nishmitha K M
(4AI23CD034)

Ms. Dhanushree D D
(4AI23CD016)

Under the guidance of

Mrs. Shilpa K. V B.E., MTech.

Assistant Professor

Department of CS&E (DATA SCIENCE)

Adichunchanagiri Institute of Technology

CHIKKAMAGALURU - 577102

2025-26

ADHICHUNCHANAGIRI INSTITUTE OF TECHNOLOGY

Jyothinagar, Chikkamagaluru-577102

DEPARTMENT OF CS&E(DATA SCIENCE)

CERTIFICATE

This is to certify that the Mini project work entitled **" DATA SCIENCE RESOURCE HUB "** Is a bonafied work carried out by **Ms. Pooja Shankar S (4AI23CD036) Ms. Spoorthi H (4AI23CD053) , Ms. Nishmitha K M (4AI23CD034), Dhanushree D D (4AI23CD016),** in partial fulfillment for the **Mini Project (BCD586)** course of 5 th semester Bachelor of Engineering in **Computer Science and Engineering (Data Science)** of the Visvesvaraya Technological University, Belagavi during the academic year **2025-2026** It is certified that all corrections and suggestions indicated for Internal Assessment have been Incorporated in the report deposited in the department library. The Mini project report has been approved as it satisfies the academic requirements in respec t of Project Work prescribed for the said Degree.

Signature of the Guide
Mrs. Shilpa K V B.E., M.Tech
Assistant Professor

Signature of Coordinator
Mrs. Shilpa K V B.E., M.Tech
Assistant Professor

Signature of the HOD
Dr. Adarsh M J B.E., M.Tech., Ph.D
Associate Professor and Head

ABSTRACT

Centralized access to academic resources has become an essential requirement in modern educational institutions. With the increasing volume of digital study materials such as notes, question papers, presentations, and project documents, students often face difficulties in locating relevant resources due to scattered storage, lack of organization, and limited accessibility. Traditional methods of sharing academic content through multiple platforms or manual distribution are time-consuming, inefficient, and unreliable. The **Data Science Resource Hub** is developed to overcome these challenges by providing a centralized, structured, and user-friendly platform for managing and accessing departmental academic resources.

This project implements a reliable and interactive web-based resource management system that enhances academic efficiency through features such as secure faculty uploads, categorized resource storage, and easy student access. The platform provides dedicated interfaces for faculty and students, enabling seamless upload, browsing, and downloading of study materials. The Data Science Resource Hub is built using modern web technologies including **Django** for backend development, **HTML, CSS, and JavaScript** for the frontend interface, and **SQLite** for database management. These technologies work together to ensure secure authentication, efficient file handling, and smooth interaction across all modules of the system. Overall, this project delivers an efficient, organized, and scalable solution for academic resource management within the Data Science department. By centralizing learning materials and simplifying access, the Data Science Resource Hub significantly reduces search time, improves resource availability, and supports a digitally enhanced learning environment. The system promotes better coordination between faculty and students and serves as a practical and deployable solution for institutional academic resource management.

ACKNOWLEDGEMENTS

We express our humble pranamas to his holiness **Divine Soul Parama Poojya Jagadguru Padmabushana Sri Sri Sri Dr.Balagangadharanatha Maha Swamiji** and **Parama Poojya Jagadguru Sri Sri Sri Dr. Nirmalanandanatha Maha Swamiji Pontiff,**
Sri Adichunchanagiri Maha Samsthana Matt and Sri Sri Gunanatha Swamiji, Chikkamagaluru branch, Sringeri who have showered their blessings on us.

The completion of any project involves the efforts of many people. We have been lucky enough to have received a lot of help and support from all quarters during the making of this project, so with gratitude, we take this opportunity to acknowledge all those whose guidance and encouragement helped us emerge successful.

We express our gratitude to **Dr. C K Subbaraya**, Director, Adichunchanagiri Institute of Technology.

We express our sincere thanks to our beloved principal**, Dr. C T Jayadeva** for having supported us in our academic endeavors.

We are thankful to the resourceful guidance, timely assistance and graceful gesture of our guide **Mrs. Shilpa K V,** Asst. Professor, Department of CS&E (DATA SCIENCE), who has helped us in every aspect of our project work.

We thank our project coordinator **Mrs. Ramya C K,** Asst. Professor, Department of CS&E (DATA SCIENCE), for her lively correspondence and assistance in carrying on with this project

We are also indebted to **Dr. Adarsh M J,** HOD of CS&E (DATA SCIENCE) Department, for the facilities and support extended towards us.

We would be very pleased to express our heart full thanks to all the teaching and nonteaching staff of CS&E (DATA SCIENCE) Department and our friends who have rendered their help, motivation and support.

Pooja Shankar S

Nishmitha K M

Dhanushree D D

Spoorthi H

TABLE OF CONTENTS

ABSTRACT	**i**
ACKNOWLEDGEMENT	**ii**
TABLE OF CONTENTS	**iii**
LIST OF FIGURES	**iv**
LIST OF TABLES	**v**
LIST OF SNAPSHOTS	**vi**

CHAPTERS	**Page No.**
1. INTRODUCTION	**01**
1.1 Background	01
1.2 Problem Statement	01
1.3 Objective of the System	02
1.4 Significance of the System	03
1.5 Scope of the Project	03
1.6 Methodology	04
1.7 Target Audience	04
1.8 Overview of the Report	04
2. SYSTEM DESIGN	**06**
2.1 System Architecture	06
2.2 Module Design	07
2.3 User Interface Design	07
2.4 Technology Stack	07
3. IMPLEMENTATION	**09**
3.1 Backend Implementation	09
3.2 Frontend Implementation	10
3.3 Database Implementation	10
4. TESTING	**11**
4.1 Testing Objectives	11
4.2 Testing Environment	1

4.3 Types of Testing 12

4.4 Test Cases 13

5. RESULTS AND DISCUSSION 14

5.1 Results 14

5.2 Discussion 17

5.3 Challenges Encountered 17

5.4 Limitations of the Current System 18

6. CONCLUSION 19

6.1 Conclusion 19

6.2 Future Enhancements 19

REFERENCES viii

LIST OF FIGURES

Figure No.	**Description**	**Page No**
2.1	Architecture Diagram of DS Resource Hub	06

LIST OF TABLES

Table No.	Description	Page No.
4.4	Test cases for DS Resource Hub	13

LIST OF SNAPSHOTS

Sl No.	Description	Page No.
5.1	Home Page	06
5.2	Organized notes section	14
5.3	Admin Dashboard	15
5.4	Admin Editing Page	15
5.5	Faculty Details	16
5.6	Resource Adding Page	16

Chapter 1

INTRODUCTION

1.1 Background

- **Context**: Data Science Resource Hub website that provides students with notes, question papers, syllabus details, project information, and faculty profiles for academic support.
- **Problem Statement:** Students in the Data Science department often struggle to access organized academic resources such as notes, previous question papers, syllabus details, project guidelines, and faculty information. This lack of a centralized platform leads to inefficiency and difficulty in preparing for exams, completing projects, and staying updated with academic requirements. To solve this, a unified, easy-to-navigate online resource hub is needed to provide all essential academic materials in one place.
- **Opportunity**: There is an opportunity to create a centralized, user-friendly online platform that organizes and delivers essential Data Science academic resources, making it easier for students to access notes, question papers, syllabus details, project materials, and faculty information quickly and efficiently.

1.2 Problem Statement

- **Overview of the Problem**: Students in the Data Science department face challenges in accessing the academic materials they need because resources like notes, question papers, the syllabus, project information, and faculty details are scattered across different sources. This lack of organization makes it difficult for students to study effectively, prepare for exams, and stay updated with coursework. Without a centralized system, students spend unnecessary time searching for information instead of focusing on learning, which reduces their overall academic efficiency and performance.

- **Specific Issues**:
- Lack of consistency in how resources are shared by faculty or seniors.
- No centralized hub for notes, question papers, syllabus, and project details.
- Time wasted searching for study materials instead of focusing on learning.
- Limited communication about faculty details and academic responsibilities.

1.3 Objective of the System

The objective of the system is to provide a centralized, user-friendly online platform that organizes and delivers essential Data Science academic resources—such as notes, question papers, syllabus details, project information, and faculty profiles—to enhance accessibility, improve learning efficiency, and support students in their academic and project-related activities.

Key Goals:

- **Centralization:** Bringing all academic resources together in one platform.
- **Accessibility**: Ensuring students can easily find and use study materials.
- **Efficiency:** Reducing time spent searching for notes and question papers.
- **Organization:** Structuring content like syllabus, notes, and projects clearly**.**
- **Support:** Helping students prepare for exams and complete projects smoothly.
- **Communication**: Providing clear and updated faculty information.
- **Reliability:** Offering accurate and regularly updated academic resources.
- **Usability:** Creating a simple, clean, and user-friendly interface.

1.4 Significance of the System

- **Efficiency**: Reduces time and effort for students in finding notes, question papers, syllabus, and project materials, enabling more focused learning and better academic performance.

- **Accuracy:** Ensures that all academic resources, including notes, question papers, syllabus, and project details, are correct, up-to-date, and reliable for students' reference.
- **Real-Time Monitoring:** Allows administrators to track updates, resource availability, and user activity instantly, ensuring that students always have access to the latest academic materials.
- **Data Analysis:** Collects and examines usage patterns, resource access frequency, and student interactions to improve the organization, relevance, and effectiveness of the academic resources provided.
- **Cost-Effective:** Minimizes expenses by providing a centralized digital platform for academic resources, reducing the need for printed materials and manual distribution.

1.5 Scope of the Project

In Scope:

- Students can access notes, syllabus, question papers, and project materials from a single platform.
- Provides guidelines, documentation, and reference materials for academic projects.
- Enables access from anywhere at any time, supporting remote learning.
- Displays profiles, contact details, and roles of faculty members.
- Can be expanded to include additional courses, resources, or interactive features in the future.

Out of Scope:

- Does not provide AI-based recommendations or predictive analytics for learning.
- The platform cannot be used without an internet connection.

1.6 Methodology

- **Approach**: The system will be developed using a **web-based platform**, leveraging modern technologies such as **HTML, CSS, JavaScript**, and a backend language like **Python**. The system will interact with a **relational database** (like **MySQL** or **PostgreSQL**) to store and retrieve.
- **Agile Development**: The system will follow an **Agile development** methodology, involving iterative design and feedback cycles to ensure that the system meets the needs of users at each stage of development.
- **Testing**: The system will be tested through a combination of **unit testing**, **integration testing**, and **user acceptance testing** to ensure functionality and user satisfaction.

1.7 Target Audience

- **Faculties:** Faculty members use the system to upload, organize, and manage academic resources, including notes, syllabus, projects, and question papers, while also accessing departmental information efficiently.
- **Students:** Students use the system to easily access and download academic resources such as notes, syllabus, question papers, and project materials to support their learning and exam preparation.

1.8 Overview of the Report

This report is structured into several chapters that detail the development and design of the **DS Resource Hub**. The following chapters include:

- **Chapter 1**: **Introduction** – This chapter describes about the project.
- **Chapter 2: System Design** – Describes the architecture and design of the system.
- **Chapter 3: Implementation** – Discusses the system's development and the technologies used.
- **Chapter 4: Testing and Validation** – Details the testing process and results.

- **Chapter 5: Results and Discussions** – Presents and results obtained and discusses the limitations
- **Chapter 6: Conclusion and Future enhancement -** Summarizes the project and suggests future improvements.

Chapter 2

SYSTEM DESGIN

The system design of the Data Science Resource Hub is based on a client–server architecture that enables centralized management of academic resources. The frontend provides a simple user interface for students to access study materials and for faculty to upload, update, or delete files. The backend handles authentication, file processing, and secure communication with the database. All uploaded materials are categorized and stored in a structured manner for easy retrieval. This design ensures efficient access, improved organization, and a user-friendly experience for the entire department.

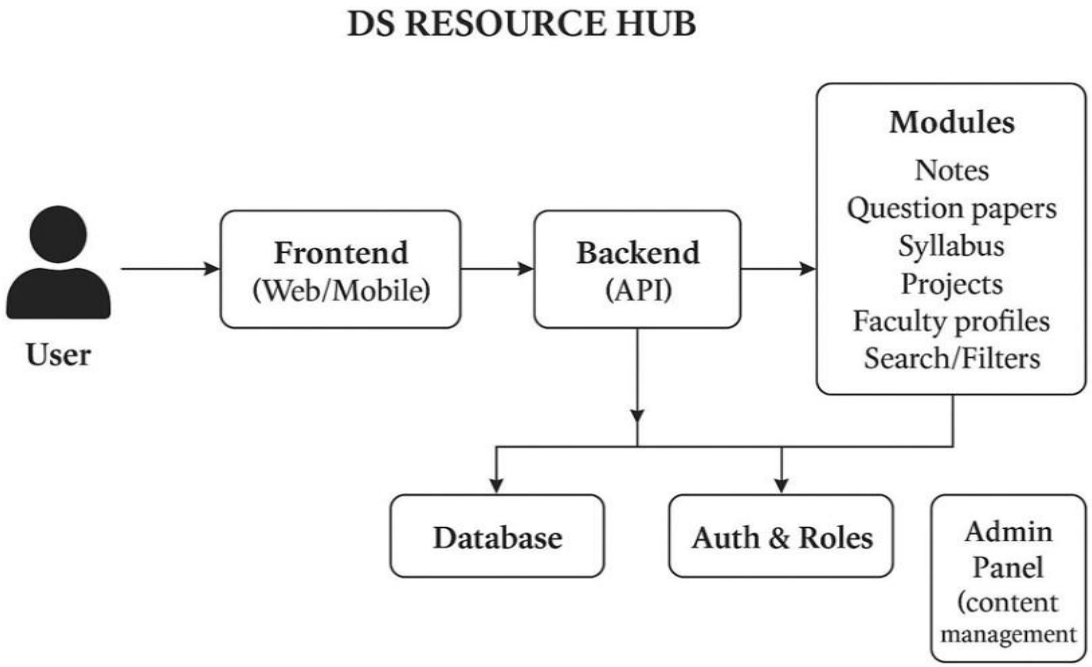

Fig 2.1: High-Level System Architecture of DS Resource Hub

2.1 System Architecture

- **High-Level Overview**: The system follows a client-server model where users interact with the Data Science Resource Hub through a web interface. The backend processes requests, handles authentication, and manages file operations, while the database stores information related to uploaded study materials, categories, and user details for retrieval.

- **Architecture Diagram**: Present a fig 2.1 showing the key components: frontend (UI), backend server, database, and file storage module.
- **Components**:
- **Frontend**: A web interface where faculty upload, update, or delete resources and students browse and download study materials.
- **Backend Server**: Handles user authentication, file processing, category management, and implements business logic for resource access.
- **Database**: Stores metadata such as file names, upload dates, categories, and user profiles for structured and secure resource retrieval.

2.2 Module Design

The system is divided into functional modules, each responsible for a specific operation in managing academic resources.

2.2.1 User Authentication Module

Handles secure login for faculty and students, ensuring that only authorized users can upload or access study materials.

2.2.2 Faculty Resource Management Module

Allows faculty to upload, update, delete, and categorize study materials such as notes, question papers, and project documents.

2.2.3 Student Access Module

Provides students with easy navigation and downloading of available academic resources based on categories or subjects.

2.2.4 File Storage and Retrieval Module

Stores uploaded files in an organized manner and retrieves them efficiently when requested by users.

2.3 Database Design

The database is designed to store structured information about uploaded resources, including file name, type, category, upload date, and uploaded by details. It also maintains user login credentials and access permissions to ensure secure usage. The database supports efficient retrieval, search, and filtering of study materials, making resource management fast and organized.

2.4 User Interface (UI) Design Main

Screens:

- **Login Screen**: Users enter their credentials to securely access the resource hub.
- **Dashboard**: A central interface displaying available options such as upload, download, and resource categories based on user roles.
- **Upload Resource Screen:** Faculty can select files, choose category, and upload study materials to the system.
- **View & Download Screen:** Students can browse categorized resources and download study materials as needed.

2.5 Technology Stack

- **Frontend**: HTML, CSS, and JavaScript used to design a clean and user-friendly interface for browsing and uploading study materials.
- **Backend**: Django framework handles authentication, file management, and server-side processing for secure and organized resource control.
- **Database**: SQLite is used for efficient storage and retrieval of resource information, including file details, categories, and user credentials.

Chapter 3

IMPLEMENTATION

This chapter explains the steps followed to implement the Data Science Resource Hub, covering frontend development, backend functionality, database configuration, and file handling processes. It discusses the technologies used to build the platform, how different components communicate, and the integration techniques applied to ensure smooth resource upload and access. The implementation focuses on secure authentication, categorized data storage, and a user-friendly interface for both faculty and students.

3.1 Backend Implementation

The backend of the Data Science Resource Hub was developed using Django to handle user requests, process file uploads, manage categories, and interact with the database. It ensures secure authentication, authorization, and efficient resource retrieval.

API Endpoints

Authentication:

- POST /login – Authenticates users (Faculty/Student) and grants secure access to the system.
- POST /register – Allows registration of new faculty or student accounts.

Resource Management:

- POST /upload – Allows faculty to upload study materials with file details and category.
- DELETE /delete/:file_id – Enables faculty to delete an uploaded file.
- PUT /update/:file_id – Updates file information or category.

Resource Access:

- GET /resources – Retrieves all study resources categorized by subject or material type.
- GET /download/:file_id – Allows students to download the selected resource.

3.2 Frontend Implementation

The frontend provides the user interface for faculty and students to interact with the Data Science Resource Hub seamlessly and efficiently.

User Interface (UI) Components

- **Login Page:** Allows users to enter credentials and securely log in before accessing the Resource Hub modules.
- **Dashboard:** Displays user-specific options such as upload, manage resources (for faculty), and browse/download materials (for students).
- **Upload Resource Screen:** Enables faculty to select files, choose category, and submit study materials for upload.
- **Resource Library Screen:** Allows students to browse categorized academic resources and download files when needed.

3.3 Database Implementation

Database Setup: SQLite is used as the database system to store resource information and user data in a relational structure, ensuring efficient storage and retrieval of files and associated details.

Database Schema:

- **User Table:** Stores login credentials and role information such as user_id, username, password hash, and user type (Faculty/Student).
- **Resource Table:** Contains details of uploaded materials including resource_id, file_name, category, file_path, upload_date, and uploaded_by.
- **Category Table:** Stores category information such as category_id and category_name for organizing materials.
- **Download Log Table:** Maintains history of resource downloads including download_id, resource_id, user_id, and access_date.

Chapter 4

TESTING

This chapter covers the testing processes and methodologies applied to the Data Science Resource Hub. Testing plays a crucial role in identifying and resolving issues, verifying that the system meets both functional and performance requirements, and ensuring reliable operation for different user roles. Through a combination of unit testing, integration testing, and user interface testing, the platform was evaluated to confirm secure resource upload, smooth navigation, and accurate file retrieval under various conditions.

4.1 Testing Objectives

- Verify that the system performs all intended functions such as upload, download, and resource organization correctly through multiple test cases.
- Ensure that both user roles (Faculty and Students) can access only the features assigned to them without any functional errors.
- Test the accuracy of file uploads, category mapping, and download retrieval operations.
- Confirm that the system maintains security by preventing unauthorized access and handling invalid inputs safely.
- Evaluate the performance and reliability of the platform during multiple simultaneous requests or large file uploads.

4.2 Testing Environment

Hardware: Laptop/PC with minimum 4–8GB RAM and i3/i5 processor for development and testing.

Software:

- Backend and frontend were hosted locally using Django for backend and HTML/CSS/JavaScript for frontend interfaces.
- Database: SQLite used for storing resource information and user data.
- Testing Tools: Postman for API/file upload testing and browser-based manual testing for UI validation.

Operating System: Windows 10 or above.

Browser: Google Chrome and Microsoft Edge used for interface and download compatibility testing.

4.3 Types of Testing

4.3.1 Unit Testing

- **Objective**: To test individual modules or functions in isolation to ensure that each feature behaves correctly during file upload, user authentication, and resource retrieval.
- **Tools**: Postman for backend API validation and browser developer tools for testing UI-level interactions.

Example Test Cases:

- **User Authentication**: Verifies that login functionality correctly validates credentials and grants role-based access.
- **Resource Uploading:** Ensures that the uploaded file is successfully stored and categorized in the database.
- **Resource Downloading:** Confirms that the download operation retrieves the correct file for the selected resource without errors.

4.3.2 Integration Testing

- **Objective**: To test the interaction between various modules of the system, such as the frontend upload interface, backend processing, and database storage, ensuring they function together seamlessly.
- **Example Test Cases**:
- **Resource Upload:** Ensures that the frontend sends the file and its details to the backend, and the backend stores them correctly in the database and file storage.
- **Resource Retrieval:** Verifies that the resource list displayed on the student screen accurately reflects the data stored in the database.

- **User Access Control:** Confirms that only authorized faculty can upload/delete materials, while students are limited to viewing and downloading resources.

4.3.3 Functional Testing

- **Objective**: To evaluate the system based on defined functional requirements and ensure that all features of the Resource Hub operate according to user expectations.
- **Test Scenarios**:
- **Login and Registration**: Tests whether faculty and students can log in successfully and are redirected to their respective dashboards.
- **Resource Management:** Verifies that faculty can upload, update, and delete study materials, and the system correctly reflects those changes.
- **Viewing & Downloading Resources:** Checks that students can browse categorized resources and download files accurately without errors.

4.4 Test Cases

Below are sample test cases for various components:

Table 4.1: Test Cases for DS Resource Hub

Test Case ID	Description	Test Steps	Expected Result	Status
TC-001	Login with valid credentials	Enter valid username and password; click "Login"	User is redirected to the appropriate dashboard	Pass
TC-002	Login with invatid cremtials	Enter invalid username or password; click "Login'	Error message is displa-yed and login fails	Pass
TC-003	View Notes	User selects subject and clicks on notes	Notes are displayed/ downloaded successfully	Pass
TC-004	View Question Papers	User navigates to question papers section	Question papers are displayed correctly	Pass
TC-005	View Syllabus	User selects semester and subject	Correct syllabus details are displayed	Pass
TC-006	View Project Information	User opens projects section	Project details and guidelines are displayed	Pass
TC-007	View Paculty Profiles	User clicks on faculty section	Faculty details are displayed	Pass
TC-008	Search Resources	Admin uploads notes or question papers	Resource is uploaded and visible to users	Pass

Chapter 5

RESULTS AND DISCUSSION

The Data Science Resource Hub successfully achieved its objective of providing a centralized platform for storing and accessing academic resources. The system improved accessibility, reduced search time, and offered a user-friendly interface for both faculty and students. Resource upload and download functions performed reliably and efficiently during testing. A few challenges were encountered in managing file organization and ensuring smooth role-based access, but they were resolved during implementation. Overall, the platform proved effective, useful, and aligned with the project goals.

5.1 Results

Snapshots of the Project with description

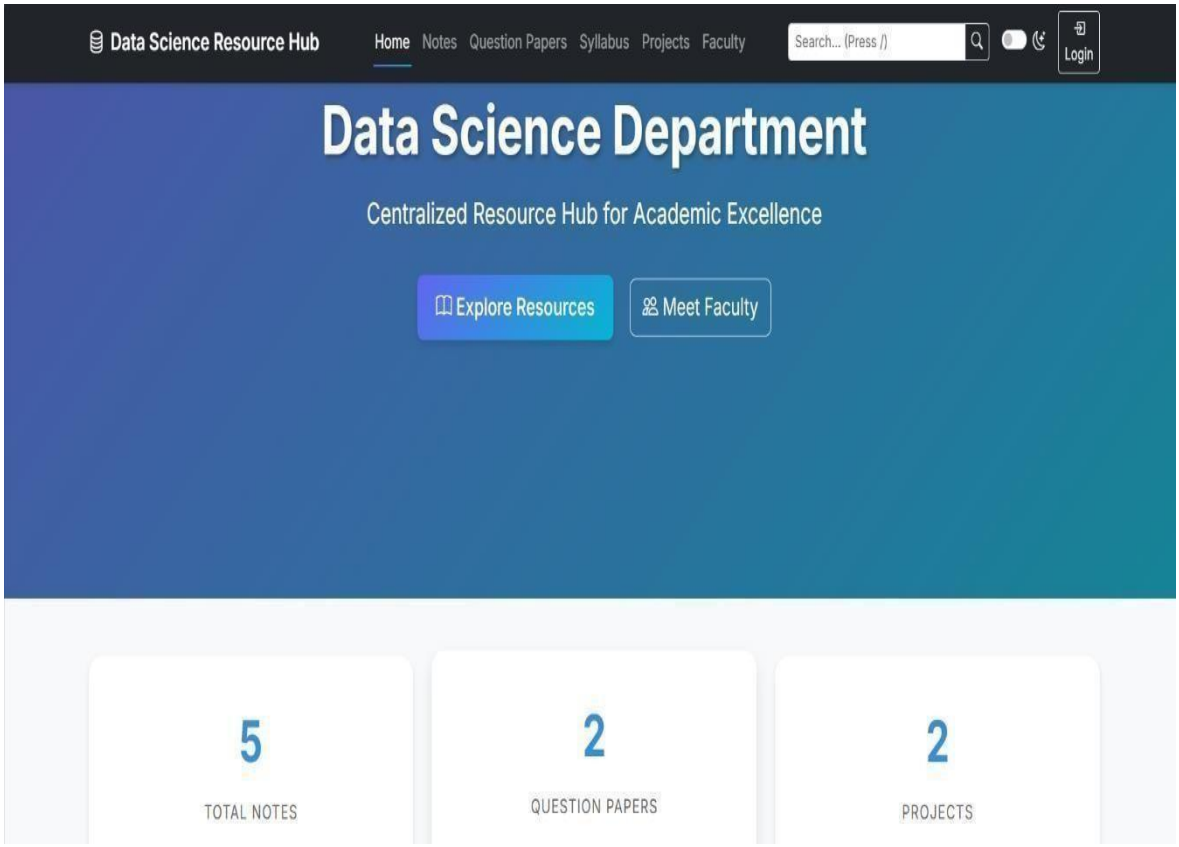

Fig 5.1: Home page

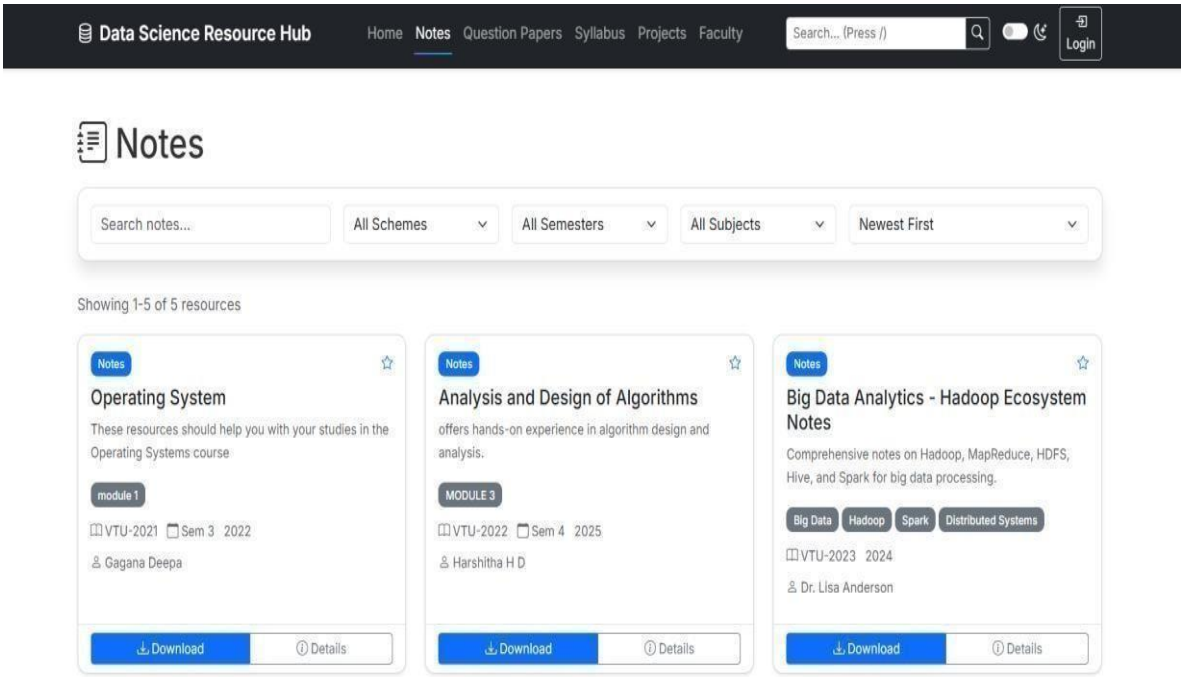

Fig 5.2: Organized notes section

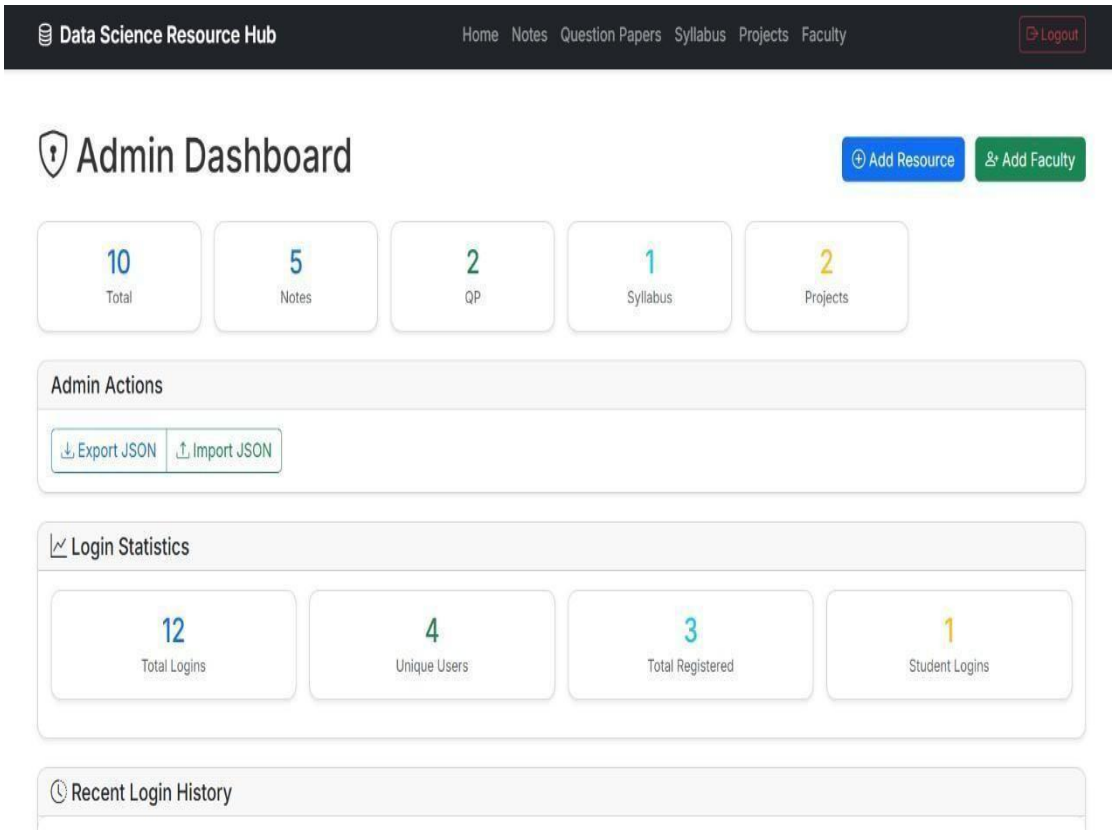

Fig 5.3: Admin Dashboard

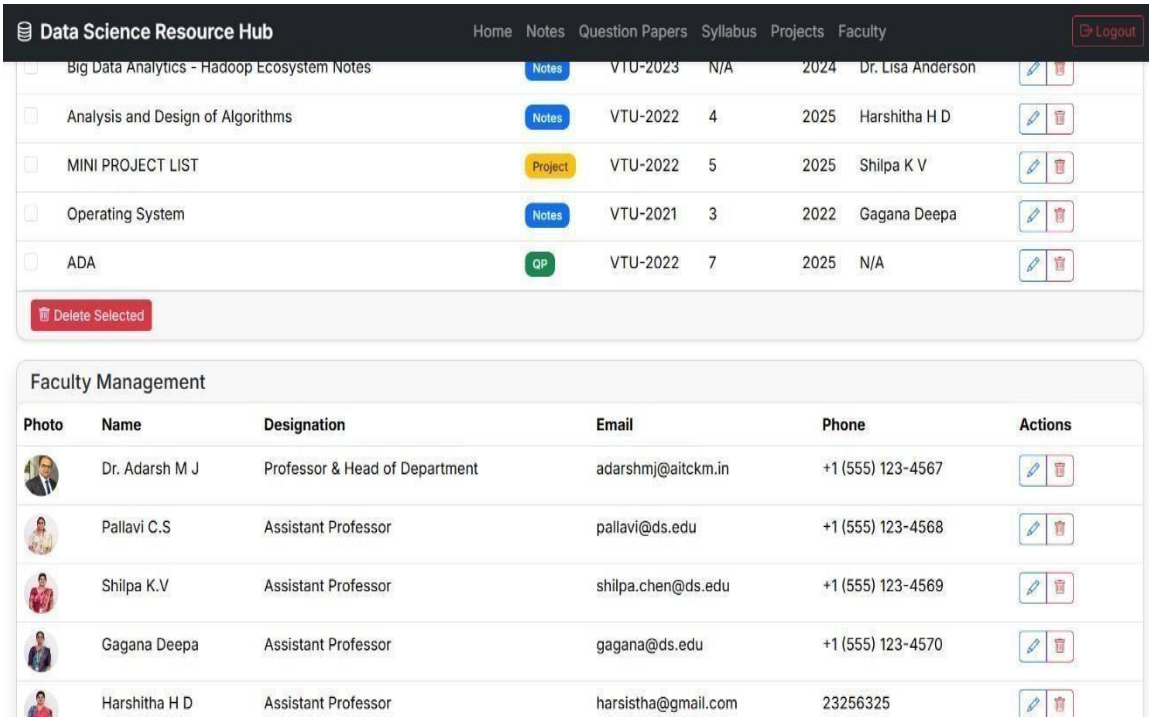

Fig 5.4: Admin Editing page

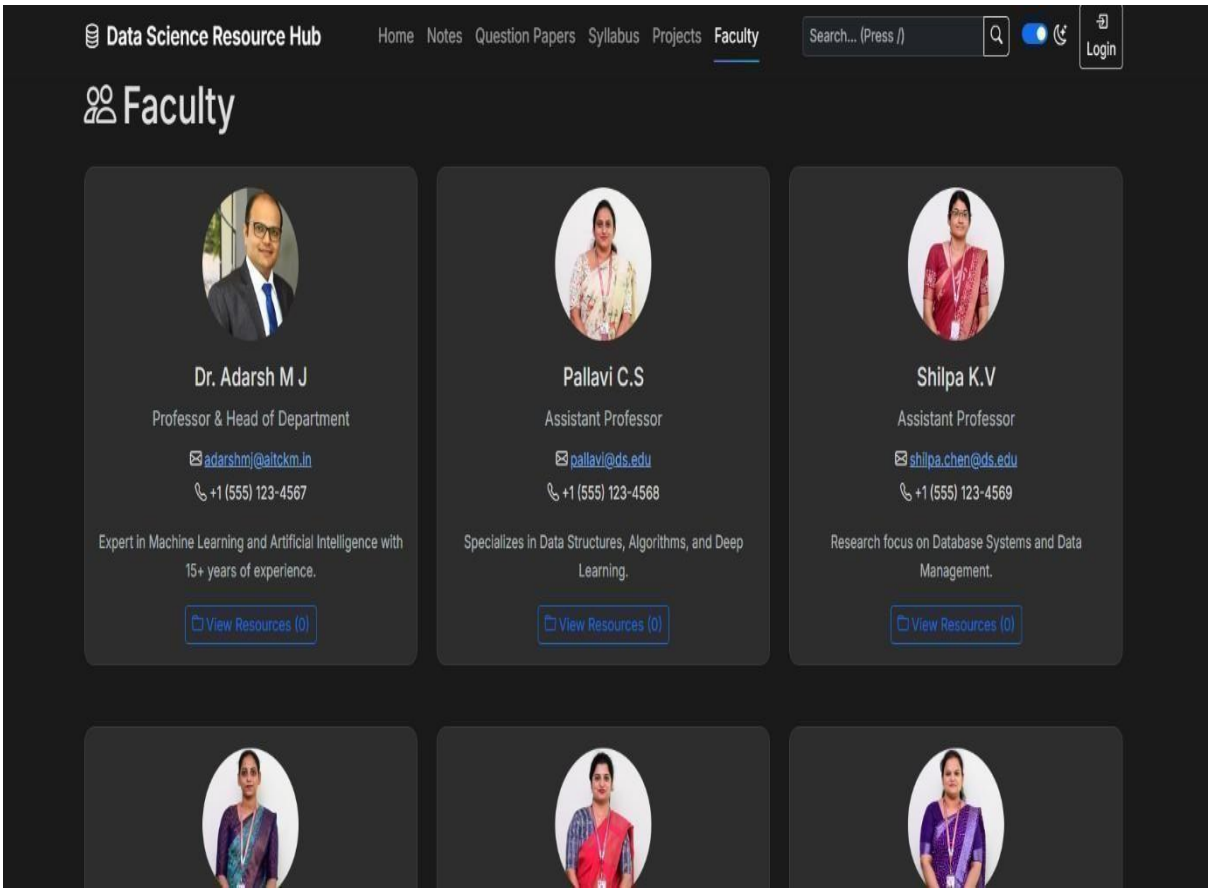

Fig 5.5: Faculty Details

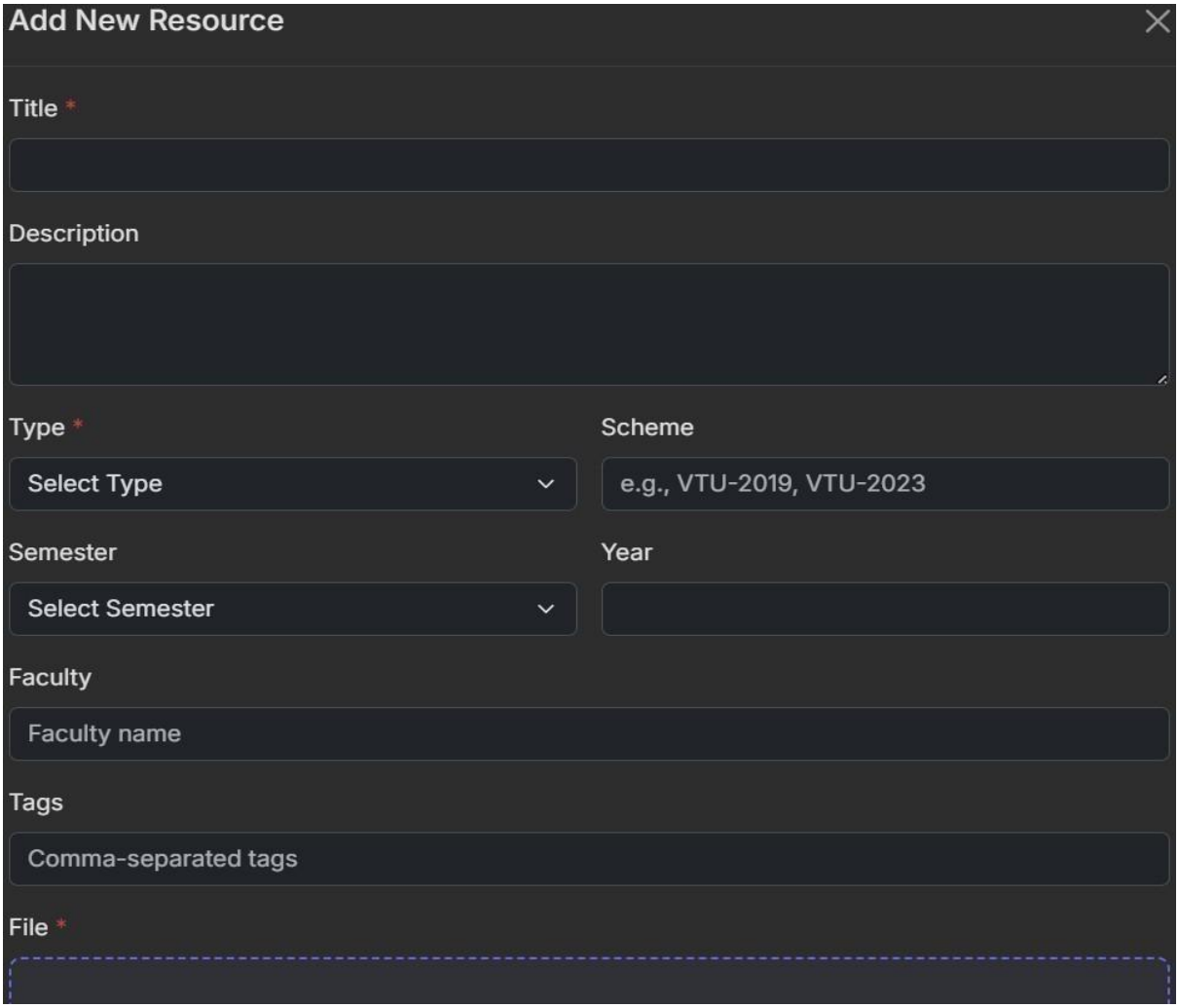

Fig 5.6: Resource adding page

5.2 Discussion

Effectiveness of the System

- The Data Science Resource Hub effectively achieved its purpose of centralizing study materials, improving accessibility, and reducing the time spent searching for academic resources.
- The implementation of role-based access ensured that faculty could manage uploads securely, while students had simplified access to downloading materials, enhancing both security and user experience.

- Overall, the system provided reliable performance, organized resource management, and ease of use, making it a valuable support platform for academic learning within the department.

5.3 Challenges Encountered

During the development of the Data Science Resource Hub, several challenges were encountered. Managing secure file uploads and ensuring only authorized faculty could modify or delete resources required careful implementation of authentication controls. Organizing files in a structured format while maintaining fast retrieval performance also posed a challenge. Additionally, designing a simple and user-friendly interface that accommodated different user roles demanded multiple UI adjustments and testing. Integrating the backend with the database for seamless upload and access operations required debugging and optimization to ensure stability.

5.4 Limitations of the Current System

The current version of the Data Science Resource Hub has certain limitations. The platform is accessible only through a web interface and does not yet include a dedicated mobile application for easier access on smartphones. Advanced search and filtering features are limited, requiring users to browse manually through categories to locate specific resources. The system also lacks automated notifications to alert students when new materials are uploaded. Additionally, the hub does not include analytics or reporting features to track resource usage or download patterns.

Chapter 6

CONCLUSION AND FUTURE ENHANCEMENTS

6.1 Conclusion

- The **Data Science Resource Hub** successfully achieved its main objective of providing a centralized platform for storing and accessing academic resources within the department. By allowing faculty to securely upload study materials and enabling students to easily browse and download them, the system improves resource availability and academic support.
- Overall, the platform demonstrates the benefits of organizing learning materials digitally, enhancing efficiency, saving time, and reducing dependency on scattered storage methods. It serves as a reliable and helpful system for both faculty and students, promoting better learning and streamlined resource management within the institution.

6.2 Future Enhancements

To further improve the effectiveness and usability of the Data Science Resource Hub, the following enhancements are recommended:

- Integration of an advanced search and filter feature to help users quickly locate specific study materials.
- Implementation of a notification system to alert students when new notes, question papers, or resources are uploaded.
- Development of a mobile application to enable easier access to study materials on smartphones.
- Addition of usage analytics to track resource downloads and identify most-used study materials.
- Integration of AI-based suggestions to recommend resources based on user behavior and course requirements.

Push Notifications and Reminders

- Integrating push notifications would enable the system to alert students when new study materials, assignments, or question papers are uploaded. For example, notifications could inform students about recently added notes or upcoming project submission resources.
- Faculty could also receive reminders to upload updated materials or announcements, helping ensure timely sharing of resources and improving communication between faculty and students.

Mobile Application Development

- Developing a dedicated mobile application for Android and iOS could significantly enhance accessibility and user convenience. A mobile app would allow students and faculty to browse, upload, and download study materials directly from their phones, providing a seamless experience with an interface optimized for smaller screens.

REFERENCES

[1] Django Software Foundation. (2024). Django Web Framework. Retrieved from https://www.djangoproject.com/

[2] Python Software Foundation. (2024). Python Programming Language. Retrieved from https://www.python.org/

[3] SQLite. (2024). SQLite Database Engine. Retrieved from https://www.sqlite.org/ [4] W3Schools. (2024). HTML, CSS, and JavaScript Documentation. Retrieved from https://www.w3schools.com/

[5] Bootstrap Documentation. (2024). CSS Framework Guide. Retrieved from https://getbootstrap.com/

Books: Author(s) Last Name, First Initial. (Year). *Title of the Book* (edition). Publisher.

Journal Articles: Author(s) Last Name, First Initial. (Year). "Title of the Article," *Journal Name*, Volume(Issue), Page numbers.

Online Resources: Author/Organization. (Year, Month Day). *Title of the Webpage*. Retrieved from URL.

Software Tools: Organization. (Year). *Title of the Software Tool*. Retrieved from UR