Application Form For NOC and LOR

project report submitted in the partial fulfilment

of the requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY

Submitted By

MAVURI YAMINI
[211FA07016]
NAMBURU NAGA THANMAIE
[211FA07038]

Under the guidance of

Dr. Srikanth Yadav. M
Associate Professor

Vignan's Foundation for Science, Technology and Research

(Deemed to be University)

Guntur, A.P., India.

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

Department of Information Technology
Vignan's Foundation for Science, Technology and Research
(Deemed to be University)
Vadlamudi, Guntur, Andhra Pradesh - 522213, India
MAY -2025

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

DECLARATION

I hereby declare that the work described in this document, entitled "**APPLICATION FOR LOR&NOC"** which is being submitted by me in the partial fulfillment of the requirements for the completion of Project of in Dept. of Information Technology to Vignan's Foundation for Science Technology and Research (Deemed to be University), Andhra Pradesh, is the result of investigations carried out by me under the Guidance of **Dr. Srikanth Yadav. M, Associate Professor** Information Technology to Vignan's Foundation for Science Technology and Research (Deemed to be University), Andhra Pradesh. The work is original and has not been submitted for any Degree/Diploma of this or any other university.

Place: Vadlamudi

Date:

M. Yamini
(211FA07016)

N. Naga Thanmaie
(211FA07038)

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

CERTIFICATE

This is to certify that the project report entitled **"Application for LOR&NOC"** being submitted by **M. Yamini (211FA07016) and N. Naga Thanmaie(211FA07038)** in partial fulfilment of the requirements for the award of Bachelors degree, Department of Information Technology, Vignan's Foundation for Science, Technology and Research, (Deemed to be University), Vadlamudi, Guntur District, Andhra Pradesh, India, is a Bonafide work carried out by them under my guidance and supervision.

Dr. Srikanth Yadav. M	Mrs.Nazma Sultana	Dr. K. Sujatha
Project Guide	Project Coordinator	HOD, IT

External Examiner

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

AKNOWLEDGEMENTS

I express my sincere thanks to Dr. Srikanth Yadav. M, Associate Professor, Information Technology to Vignan’s Foundation for Science Technology and Research (Deemed to be University), Andhra Pradesh, India, for his/her valuable guidance and encouragement. I sincerely thank Dr. K. Sujatha, Head of the Department, Department of Information Technology for his valuable suggestions.

I am deeply indebted to my family members, relatives and friends for their continuous motivation, encouragement and timely support in completing the research work.

M. Yamini
(211FA07016)

N. Naga Thanmaie
(211FA07038)

Project Experience Information Sheet

1. Title of the Project : Applying for LOR & NOC

2. Nature of the project : Design

3. Project Category : Societal centric

4. Any further classification on the nature of the project : System

5. Constraints considered :

Constraint Standard	Remarks
Environmental	By digitizing the LOR/NOC request process, the project helps reduce paper usage, contributing to an eco-friendly approach.
Ethical	The system ensures ethical handling of student data by implementing login authentication and data privacy measures.

6. Major Courses Covered in the Project : HTML, CSS and JavaScript, DBMS

7. SDGs Mapped :

SDG Name	SDG No	Supporting Remarks
Quality Education	4	The system facilitates timely and efficient access to academic documentation, supporting student mobility and academic progression.
Industry, Innovation and Infrastructure	9	The project promotes digital infrastructure within the university, encouraging innovation in administrative processes.

Signature of the Student **Signature of the Supervisor**

ABSTRACT

The "Applying for LOR & NOC" project is an innovative digital platform developed to address the challenges faced by students and faculty in the manual process of obtaining Letters of Recommendation (LOR) and No-Objection Certificates (NOC). Traditionally, students seeking these documents had to go through a time-consuming and often inefficient offline procedure that involved visiting faculty in person, submitting handwritten or email-based requests, and waiting for approvals and signatures. This not only created bottlenecks in communication but also led to delays, errors in data handling, and lack of a systematic record of applications. Recognizing the need for a more streamlined and transparent system, this web application was developed to automate and manage the entire workflow.The application provides a user-friendly interface where students can choose their respective school or department and fill out a detailed form specifying their request type—whether it is a Letter of Recommendation for higher studies or a No-Objection Certificate for internship or employment. The form collects essential information such as the student's name, SRN, contact details, academic background, purpose of request, organization name, and the preferred faculty recommender. Once submitted, the request is sent to the concerned faculty or administrative staff for review and processing.The system improves efficiency by reducing paperwork, minimizing manual errors, and providing real-time status tracking of applications. It also ensures secure storage and easy retrieval of data, which helps faculty members and administrative staff manage and verify requests systematically. Furthermore, the platform is aligned with the institution's digital transformation goals, contributing to a paperless and more environmentally sustainable campus.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

LOR	Letter Of recommendation
NOC	No Objection Certificate
SRN	Student Registration Number

CHAPTER 01

INTRODUCTION

In academic and professional settings, formal documents such as Letters of Recommendation (LOR) and No Objection Certificates (NOC) play a crucial role in verifying an individual's credentials, intent, and institutional approval. These documents are essential when students apply for higher studies, internships, conferences, workshops, or employment. Traditionally, the process of applying for and issuing LORs and NOCs has been manual, time-consuming, and often prone to delays and inconsistencies.

This project proposes a web-based solution to streamline and automate the application process for LORs and NOCs. The system includes user-friendly interfaces for students and faculty, where students can register, log in, and submit detailed requests, while faculty can log in to review and respond to LOR applications. The application dynamically adapts to the type of request—LOR or NOC—and ensures that all necessary information is collected accurately. With the integration of document templates and automation tools, the platform generates standardized, professional documents efficiently. The system is designed to reduce manual workload, ensure accuracy, and provide a transparent, secure, and organized way of handling formal document requests within educational institutions.

1.1 Purpose

In academic and professional settings, formal documents such as Letters of Recommendation (LOR) and No Objection Certificates (NOC) play a crucial role in verifying an individual's credentials, intent, and institutional approval. These documents are essential when students apply for higher studies, internships, conferences, workshops, or employment. Traditionally, the process of applying for and issuing LORs and NOCs has been manual, time-consuming, and often prone to delays and inconsistencies.

This project proposes a web-based solution to streamline and automate the application process for LORs and NOCs. The system includes user-friendly interfaces for students and faculty, where students can register, log in, and submit detailed requests, while faculty can log in to review and respond to LOR applications. The application dynamically adapts to the type of request—LOR or NOC—and ensures that all necessary information is collected accurately. With the integration of document templates and automation tools, the platform generates standardized, professional documents efficiently. The system is designed to reduce manual workload, ensure accuracy, and provide a transparent, secure, and organized way of handling formal document requests within educational institutions.

1.2 Scope

The scope of this project is to design and develop a user-friendly web application that facilitates the efficient application and generation of Letters of Recommendation (LOR) and No Objection Certificates (NOC). The system allows students to register, log in, and submit detailed requests for LORs or NOCs, depending on their requirement. It includes dynamic form handling to display relevant fields based on the type of application. Faculty members can also register and access a dedicated dashboard to view, accept, or reject student LOR requests. The platform automates the creation of well-structured documents using professional templates and stores them securely for future reference. This project focuses solely on the digital collection, processing, and generation of application requests and documents—it does not include the broader administrative approval process or institutional workflow management. Overall, the system is intended to improve the speed, accuracy, and consistency of document generation, reducing the burden on faculty and administrative staff while enhancing the experience for students.

1.3 Significance of the project

The project holds significant value in enhancing the efficiency, accuracy, and accessibility of applying for Letters of Recommendation (LOR) and No Objection Certificates (NOC) within academic institutions. By transitioning from a manual to a digital process, the system reduces paperwork, minimizes errors, and ensures faster turnaround times for both students and faculty. It promotes transparency and standardization in document generation, helping students present professionally formatted documents to external organizations. For faculty and administrative staff, the project reduces repetitive tasks and allows better management of requests. The system also ensures data security and maintains a digital record for future tracking or verification. Overall, this project supports academic mobility, encourages timely participation in external opportunities, and strengthens institutional credibility through streamlined documentation processes.

1.4 Objectives of the project

- To create a web-based platform for applying and processing LOR and NOC requests efficiently
- To provide separate login and registration modules for students and faculty.
- To design dynamic application forms that adjust based on whether the request is for a LOR or NOC**.**
- To collect and manage key details such as name, SRN number, address, contact info, email, and request-specific information.
- To enable faculty members to view, accept, or reject LOR requests through a secure dashboard.
- To automate the generation of professional LOR and NOC documents using predefined templates**.**
- To reduce the manual workload on faculty and administrative staff by automating repetitive processes.
- To provide a user-friendly interface that supports smooth navigation and quick application submission.
- **For LOR Applications:**
  - **Faculty for LOR**: The name of the faculty member who will provide the Letter of Recommendation will be recorded.
  - **Organization Name**: The name of the organization to which the applicant is applying (for higher education, internships, etc.) will be captured to tailor the LOR.
  - **Mode of Occupation**: This refers to whether the applicant is seeking to participate in a full-time or part-time role within the organization.
  - **Description of the Request**: A brief explanation of why the LOR is being requested and the specific opportunity the applicant is applying for will be recorded.
  - **Year of Pass-out**: The year the applicant is expected to graduate or has graduated from the institution will be noted for context.
- **For NOC Applications:**
  - **Organization Name**: The name of the organization where the applicant plans to participate or work will be captured.
  - **Mode of Occupation**: This will indicate whether the applicant is seeking to engage in part-time or full-time work or participation.

- **Description of the Request**: A detailed description of why the NOC is needed and what activity the applicant intends to pursue will be included.
- **Part-time or Full-time Status**: This will specify whether the applicant's involvement in the activity will be part-time or full-time.
- **Year of Pass-out**: The applicant’s graduation year will be recorded to provide relevant context for the NOC request

1.5 Features of the project

- **Student Registration & Login**: Allows students to create accounts and securely log in to submit requests.
- **Faculty Registration & Login**: Enables faculty members to manage LOR requests through their own accounts.
- **Dynamic Application Form**: Automatically adjusts fields based on whether the user selects LOR or NOC, ensuring relevant data is collected.
- **Detailed Request Input**: Gathers key information such as name, SRN, contact details, faculty name, organization name, mode of occupation, and year of pass-out.
- **Faculty Request Handling**: Faculty can view incoming LOR requests and choose to accept or reject them.

LOR-Specific Details:

- For applicants requesting a Letter of Recommendation (LOR), the system collects details such as the faculty member providing the LOR, the organization to which the applicant is applying, mode of occupation (full-time or part-time), a description of the request, and the year of graduation. This allows the LOR to be customized and relevant to the applicant's intended purpose.

NOC-Specific Details:

- For No Objection Certificate (NOC) applications, the system collects details like the organization name, mode of occupation, description of the request, whether the position is part-time or full-time, and the year of pass-out. These inputs ensure that the NOC is tailored to the specific activity the applicant wants to pursue.

1.6 Organization of report

- **Chapter – 01: Introduction**

  This chapter outlines the foundation of the project, explaining the background and the context in which the system is developed. It includes the purpose and importance of Letters of Recommendation (LOR) and No Objection Certificates (NOC) in academic and professional scenarios. The chapter also discusses the objectives, scope, significance, and features of the proposed system, highlighting how it can resolve current inefficiencies in manual processes.

- **Chapter – 02: Literature Survey**

  This chapter presents a review of existing studies, tools, and systems related to document automation in educational institutions. It explores the technologies commonly used for similar applications, identifies gaps in current systems, and highlights how the proposed system aims to offer improvements in document handling, automation, and user experience.

- **Chapter – 03: Methodology**

  Here, the development methodology followed in the project is explained in detail. It covers the planning, design, and step-by-step development process, including user requirement analysis, technology selection, and system design. The methodology ensures logical progress and efficient error handling through systematic stages.

- **Chapter - 04 : System design and implementation**

  This chapter provides a comprehensive overview of the technical implementation of the system. It includes details of modules such as student registration and login, faculty registration and management, dynamic application forms, request handling, and document generation. Frontend and backend integration is discussed, along with code structure and screenshots.

- **Chapter – 05: Testing and Results**

  This section focuses on validating the system through functional and non-functional testing. It presents test cases, outcomes, and system responses under various scenarios. The effectiveness, usability, and accuracy of the automated document generation process are also evaluated.

- **Chapter – 06: Conclusion**

  The final chapter summarizes the key outcomes and benefits of the system. It reflects on how the objectives were achieved and suggests future enhancements such as integration with institutional databases, notification systems, or mobile app versions to further improve usability and scalability.

Summary

The introduction chapter provides an overview of the purpose and importance of developing a web-based system for applying for Letters of Recommendation (LOR) and No Objection Certificates (NOC). It explains the background context, highlighting the significance of LOR and NOC in academic and professional settings, such as higher education admissions, internships, and job applications. The chapter identifies the current challenges in the manual documentation process, including time delays, inconsistencies, and administrative burden. It outlines the project's objectives, which include automating the application and generation process, improving accuracy, and ensuring standardized document formats. Additionally, the chapter emphasizes the benefits to both students and institutions and introduces the structure of the overall repo

CHAPTER 02

LITERATURE SURVEY

The literature survey for the project "Applying for LOR & NOC" highlights the need for automating the traditional manual process of requesting and generating academic documents such as Letters of Recommendation (LOR) and No Objection Certificates (NOC). Existing systems in many institutions rely on time-consuming, error-prone manual workflows that often lack standardization and efficiency. Research in educational automation suggests that web-based systems using technologies like HTML, CSS, JavaScript, PHP, and databases can streamline the process, reduce administrative workload, and improve the accuracy and consistency of generated documents. Studies also emphasize the importance of secure user authentication, role-based access (for students and faculty), dynamic form handling, and template-driven document creation to ensure a professional output. Drawing from these insights, the proposed system is designed to offer a complete digital solution that simplifies application, approval, and document generation processes for both LOR and NOC, improving user experience and institutional efficiency.

2.1 Existing Research

Existing research in the field of academic process automation has shown a consistent shift towards digital systems that replace outdated manual procedures for issuing formal documents like Letters of Recommendation (LOR) and No Objection Certificates (NOC). In many educational institutions, students are still required to fill out physical forms or send email requests to faculty members, who then manually draft and sign these documents. This traditional method often results in delays, miscommunication, inconsistent formatting, and additional burden on faculty and administrative staff.

Several studies and pilot implementations have explored the use of web-based platforms to simplify this workflow. For instance, research highlights the effectiveness of using structured web forms to collect accurate applicant data, which can then be used to auto-populate predefined document templates. This not only ensures consistency but also speeds up the entire process. Web development technologies such as HTML, CSS, JavaScript for frontend design and PHP, Python, or Node.js for backend logic are commonly used. Databases like MySQL or MongoDB are employed to securely store user data, application records, and document histories.

Moreover, existing literature emphasizes the importance of integrating role-based access systems to differentiate between student, faculty, and admin users. This ensures that only authorized personnel can approve, modify, or issue documents. Some systems even incorporate automated notifications and tracking features to inform students of the status of their requests, enhancing transparency and user experience.

Despite these advancements, gaps still exist. Most current systems are either fragmented or tailored to specific departments, lacking scalability and cross-departmental integration. They also often require manual document uploads instead of generating documents dynamically based on form inputs. Hence, there's a growing need for an all-in-one, scalable, and automated platform that covers the complete lifecycle of applying for and issuing LOR and NOC.

The proposed project builds on these research findings to deliver a web-based solution that addresses these limitations. It aims to provide a unified interface where students can register, submit detailed applications, and track their status; faculty can review and approve requests; and the system can automatically generate and format LOR/NOC documents based on the collected data. This aligns with global digital transformation trends in educational administration and contributes to a more streamlined and transparent academic environment.

2.2 Key findings

- **Manual Processes Are Time-Consuming**: Research consistently indicates that traditional approaches for requesting LOR (Letter of Recommendation) and NOC (No Objection Certificate) are heavily dependent on paperwork, in-person communication, and manual drafting. These practices not only consume significant time for both students and faculty but also increase the chances of human errors, such as spelling mistakes, incorrect formatting, or incomplete information. This can result in delayed application approvals or the need for document reissuance.
- **Web-Based Automation Increases Speed and Accuracy**: Automation tools using web technologies like PHP, JavaScript, and MySQL have proven to enhance operational speed and accuracy in academic institutions. Once a student submits a request through an online portal, the information can be directly mapped to a standard template, allowing quick and consistent generation of the required

document. Automated workflows also reduce faculty workload by providing a structured review and approval interface.

- **Role-Based Access Enhances Security**: A key finding in most studies is the need for clear access control. Implementing role-based logins (student, faculty, admin) ensures that each user has access only to the features they need. For example, students can only view their own application status, while faculty can access the requests assigned to them. This adds a critical layer of data security and prevents unauthorized changes or access.
- **Template-Based Generation Ensures Consistency**: Automated systems that use predesigned templates for generating LOR and NOC documents ensure consistency in language, layout, and structure. This standardization is important not just for institutional professionalism but also for external stakeholders (like universities or companies) who expect a certain level of formality in such documents.
- **Dynamic Forms Enable Intelligent Data Collection:** Several studies highlight the advantage of using smart forms that dynamically change based on user input. For instance, if a user selects "Applying for LOR," the system reveals additional fields like faculty name, organization, and purpose of request. Similarly, for "NOC," fields like part-time/full-time and mode of occupation are enabled. This ensures only relevant information is collected, making document generation more accurate and contextual.
- **Scalability and Flexibility Are Missing in Many Existing Systems:** While some institutions have adopted digital methods, these systems are often department-specific or not designed for long-term scalability. Research stresses the need for centralized, scalable platforms that can be expanded to serve multiple departments or even multiple institutions over time, without major restructuring.
- **User Experience Enhances Adoption and Efficiency:** Intuitive user interfaces, real-time application tracking, and automated notifications contribute to a better experience for both students and faculty. When users find the system easy to use and informative, the chances of successful adoption and continuous usage increase significantly.

These findings provide strong justification for developing a web-based system for LOR and NOC applications. Such a system would not only solve the inefficiencies of manual processing but also offer a scalable, secure, and user-friendly solution for both academic and administrative stakeholders.

Summary

The literature survey provides a comprehensive examination of previous work, tools, and systems developed to streamline the generation of academic documents such as Letters of Recommendation (LOR) and No Objection Certificates (NOC). Traditionally, these documents have been prepared manually, often resulting in delays, inconsistencies, and a significant administrative burden on faculty and staff. The reviewed literature shows a growing trend towards the digitization and automation of such processes through web-based platforms, which aim to eliminate inefficiencies and human errors.A recurring theme in the literature is the implementation of form-driven user interfaces that allow students to input relevant data, which is then validated and processed by faculty or administrative personnel via secure logins. These systems utilize modern web technologies—such as HTML, CSS, JavaScript, and PHP—for front-end and server-side processing, while MySQL or similar relational databases are used to store and manage user data and document records.

Studies have demonstrated that automation improves turnaround times, provides a standardized format for official documents, and offers tracking mechanisms for both students and faculty. Security measures like authentication, role-based access, and time-stamped records are often integrated to ensure document authenticity and prevent misuse. Moreover, some systems are designed to automatically generate PDF outputs or email notifications, enhancing communication and reducing the need for physical follow-ups.The literature also points to challenges such as initial development complexity, resistance to change in institutions used to manual methods, and the need for periodic maintenance of the system. However, these are outweighed by the long-term benefits, including improved transparency, documentation traceability, reduced paperwork, and scalability for future institutional growth.

Overall, the findings from the literature support the development of an integrated, customizable system that can efficiently handle LOR and NOC applications within academic institutions, ultimately contributing to smoother administrative operations and better user experience for students and staff alike.

CHAPTER 03

METHODOLOGY

The methodology for this project follows a structured, step-by-step approach to design, develop, and implement a web-based application for automating the application process of Letters of Recommendation (LOR) and No Objection Certificates (NOC). The process is divided into the following key stages

3.1 Overview of the project

The "Applying for LOR & NOC" project is an advanced web-based application developed to simplify and automate the generation of Letters of Recommendation (LOR) and No Objection Certificates (NOC) for students in academic institutions. Traditionally, the process of requesting and approving these documents has been manual, time-consuming, and prone to errors or delays due to paper-based handling and dependency on faculty availability. This project aims to provide a centralized digital platform that not only streamlines the entire process but also ensures accuracy, consistency, and traceability.

The system introduces a multi-user interface, including modules for students and faculty. Students are required to register or log in using their credentials. Once authenticated, they can choose whether to apply for a LOR or a NOC. The platform presents a dynamic, easy-to-use form based on the type of request. For LOR, students must provide specific details such as the recommending faculty name, target organization or institution, mode of occupation (internship, job, or education), a brief description of the purpose, and their year of graduation. For NOC, the form captures the organization name, mode of occupation (online/offline), nature of work (part-time/full-time), year of pass-out, and justification for the request.

On the faculty side, teachers or authorized staff can sign up or log in to access their dashboard where incoming requests are listed. Faculty members can review request details, verify the legitimacy of the application, and either approve or reject them. Upon approval, the system uses pre-defined professional templates to auto-generate the LOR or NOC document, significantly reducing manual effort. These documents can then be downloaded, printed, or emailed to the student or the target organization.

From a technical standpoint, the project utilizes HTML, CSS, JavaScript for building the front-end interface to ensure responsiveness and user-friendliness. PHP is used for handling server-side logic and form processing, while MySQL serves as the backend database to store user information, application history, and document records securely. The system ensures data validation, session management, and role-based access control to protect sensitive information and maintain institutional standards.

In terms of functionality and impact, the system:

- Enhances transparency and reduces turnaround time for issuing official academic documents.
- Minimizes paperwork and human error through automation.
- Allows faculty to manage and track multiple requests efficiently.
- Establishes a scalable framework that can be extended to other academic certificates and processes.
- Provides students with timely access to institutionally endorsed documents.

3.2 Proposed method

The proposed method for the project "Applying for LOR & NOC" is designed to create an efficient, digitized, and streamlined solution for students and faculty to manage academic document requests such as Letters of Recommendation (LOR) and No Objection Certificates (NOC). This method revolves around building a **dynamic web-based application** that supports real-time interactions, secure access, and automated document generation. The goal is to replace traditional manual processes that are time-consuming, prone to errors, and dependent on physical communication, with a digital system that ensures accuracy, traceability, and ease of use.

3.2.1 User Registration and Login:

The system begins with a registration module where users, categorized as students or faculty, must register by entering essential information such as name, SRN number, email, and contact details. Secure authentication is achieved through password encryption and login credentials. This ensures that only authorized users can access the platform, maintaining the confidentiality and integrity of the data.

3.2.2 Application Form Submission:

Once logged in, students can initiate the process of applying for either an LOR or NOC through a guided application form. The form is tailored depending on the type of document selected:

- For **LOR**, students must provide details such as the name of the recommending faculty, the target organization or university, the mode of occupation (e.g., internship, higher studies), a brief description of the purpose, and the year of pass-out.
- For **NOC**, students are required to enter details like the organization's name, nature of the job (part-time or full-time), mode of occupation, purpose of the request, and pass-out year.

3.2.3 Faculty Dashboard and Review:

Faculty members, upon logging in, access a personalized dashboard where they can view all pending requests from students. Each request is presented with the student’s details and reason for application. Faculty members have the authority to either approve or reject applications based on institutional guidelines or personal judgment. This system reduces delays caused by physical follow-ups and ensures transparency in decision-making.

3.2.4 Template-Based Document Generation:

Once approved, the system uses predefined, professional templates to auto-generate the required document. These templates ensure consistent formatting and include all relevant data dynamically pulled from the form.

Benefits of proposed method

- Minimizes manual effort and paperwork.
- Ensures timely responses and document generation.
- Enhances student satisfaction by offering a transparent, user-friendly platform.
- Enables faculty to manage document requests efficiently

3.3 Pre-processed data

Pre-processing data is an essential component in the development of web-based systems, especially in academic applications such as "Applying for LOR & NOC." This stage ensures that all user inputs—whether provided by students or faculty—are clean, structured, and safe for storage and further use in automated document generation. As the system handles sensitive data such as personal details, academic records, and institutional endorsements, it is critical to ensure data integrity, accuracy, and security through robust pre-processing techniques.

3.3.1. Data Collection and Validation:

The first step involves capturing user input from web forms during registration or while submitting an LOR/NOC request. Validation checks are applied immediately to verify that all mandatory fields are filled correctly. For instance:

- Names should not include numbers or special symbols.
- SRN (Student Registration Number) must match a defined institutional format.
- Email addresses and phone numbers must follow standard formats.
- Drop-downs for document type (LOR or NOC), employment mode, and year of pass-out must be properly selected.

3.3.2. Standardization and Normalization:

To ensure consistency, the data is standardized into predefined formats:

- All names and organizations are formatted with proper capitalization.
- Dates are converted to a uniform format (e.g., YYYY-MM-DD).
- Descriptions are formatted in clean, paragraph-style structures for professional document output.
- Email IDs are converted to lowercase for consistency in database matching.

3.3.3. Duplicate and Conflict Detection:

The system checks for duplicate entries or conflicts, such as:

- Multiple applications from the same student for the same purpose.
- Redundant registration details for faculty or students already present in the database.

3.3.4. Role-Based Data Access:

Once data is pre-processed and stored, access is granted based on the user's role:

- Students can view and edit their own requests.
- Faculty members can view, approve, or reject requests assigned to them.
- Admins can monitor all data flows and maintain system health.

Pre-processing of data is a foundational step in the development of any reliable and efficient web-based application, especially one dealing with official academic documents such as LOR (Letter of Recommendation) and NOC (No Objection Certificate). In this project, the pre-processing phase ensures that all data collected from users—be it students or faculty—is accurate, complete, and consistent before being stored or processed further. This includes validation of fields such as SRN number, email addresses, contact numbers, and selection of the correct document type (LOR or NOC).

Additionally, this phase involves removing unwanted characters, standardizing input formats, checking for duplicate entries, and handling missing or optional data intelligently. Data encryption and secure handling also form part of pre-processing to ensure user privacy and system security. These measures not only prevent errors during document generation but also improve user experience by minimizing rejection due to incorrect or incomplete information.

Overall, effective data pre-processing ensures that subsequent stages of the application, such as document creation, faculty approval, and system automation, function smoothly and efficiently. It enhances the system's robustness and reliability, supporting seamless academic workflows and reducing administrative burden.

3.4 Summary

The literature survey for the project "Applying for LOR & NOC" delves deeply into the current advancements and ongoing challenges in the automation of academic documentation processes. It emphasizes the transition from traditional, manual workflows to modern, digital systems that prioritize speed, accuracy, and user convenience. Manual methods, while still in use in many institutions, are widely recognized as inefficient due to their dependency on physical paperwork, delayed communication, and a high possibility of clerical errors. These challenges often hinder students from meeting application deadlines for internships, higher studies, or academic programs.

The reviewed literature presents a wide range of successful implementations of web-based systems across various universities and educational platforms. These systems are generally built using core web technologies such as HTML for structure, CSS for styling, JavaScript for interactivity, PHP or similar server-side scripting languages for backend logic, and MySQL for data management. These technologies work in tandem to create responsive, secure, and scalable platforms that can handle high volumes of data and multiple user interactions simultaneously.

Additionally, the literature discusses the incorporation of user authentication and role-based access control, which ensures that only authorized users (students, faculty, or administrators) can access or modify data. Many systems have also adopted real-time email notifications, downloadable PDF generation of LORs/NOCs, and integration with institutional databases to fetch and validate student information automatically.

Research also underlines the importance of usability in system design. A user-friendly interface with intuitive navigation improves engagement and reduces the learning curve for new users. Moreover, the survey highlights the significance of maintaining data privacy and integrity, particularly because these documents often contain sensitive personal and academic information. Best practices include the use of SSL encryption, database security measures, and audit logs for tracking changes and approvals.

In conclusion, the literature survey not only identifies the gaps in manual systems but also outlines the technical and procedural components necessary to build an efficient automated solution. These insights have shaped the foundation for the proposed system, ensuring that it aligns with current technological standards while addressing real-world institutional needs. The findings support the project's aim of reducing administrative burden, increasing operational efficiency, and empowering students through faster and more reliable access to essential academic documentation.

CHAPTER – 04
IMPLEMENTATION

4.1 System Architecture

I. User Interface Layer:

Student Portal

Registration/Login

Apply for LOR/NOC

Application Form Submission

Faculty portal

Signup / Login

View Requests

Accept / Reject Applications

Faculty Review Module

Displays student applications

Allows decision making (Accept/Reject)

II. Database Layer

Student Table: Name, SRN, Contact, Email, etc.

Faculty Table: Faculty Name, Department, Login Credentials

Application Table: Type (LOR/NOC), Status, Description, Date of Submission

III.

4.2 Module-wise implementation

I. User Registration and Login Module

- Allow students and faculty to register and log in securely.
- **Components:**

  Registration forms for students and faculty.

  Login authentication with session handling.
- **Technologies**: HTML, CSS, JavaScript (frontend), PHP/Node.js (backend), MySQL (database).

II. Student Dashboard Module

- Provide an interface for students to apply for LOR or NOC.
- **Features:**

  Apply for LOR/NOC.

  Edit profile and contact information.

View application history and status.

III. **Faculty Dashboard Module**

- Allow faculty to review and respond to LOR requests**.**
- **Features:**

  View pending LOR applications.

  Accept or reject requests with comments.

IV. **Application Form Module**

- Collect necessary details based on the type of application (LOR or NOC).
- **Features:**

  Dynamic form fields depending on LOR or NOC.

  Data validation and storage in database.

  Field examples: Name, SRN, Org Name, Year of Passout, etc.

V. **Admin/Management Module (Optional):**

- Monitor system usage, manage users, and view all applications.
- User management (students/faculty).

VI. **Database Management Module:**

- Store and retrieve application data and user details.
- **Tables:**

  Users (students and faculty)

  Applications (LOR/NOC)

4.3 Implementation codes/Source codes:

4.3.1 User Interface Layer:

4.3.1.1 Student portal

Register/Login

```
<body>
  <h2>Student Sign Up</h2>
  <form onsubmit="registerUser(event)">
    <input type="text" id="username" placeholder="Username" required />
    <input type="password" id="password" placeholder="Password" required />
    <button type="submit">Register</button>
  </form>
  <p>Already registered? <a href="login.html">Login here</a></p>
  <script>
    function registerUser(e) {
      e.preventDefault();
      const username = document.getElementById("username").value;
      const password = document.getElementById("password").value;
      // Save user in localStorage
      localStorage.setItem("studentUsername", username);
      localStorage.setItem("studentPassword", password);
      alert("Registration Successful!");
      window.location.href = "login.html";
    }
  </script>
</body>
</html>
```

✅ login.html — Student Login Page

```
<body>
  <h2>Student Login</h2>
  <form onsubmit="loginUser(event)">
    <input type="text" id="username" placeholder="Username" required />
    <input type="password" id="password" placeholder="Password" required />
    <button type="submit">Login</button>
  </form>
  <script>
    function loginUser(e) {
      e.preventDefault();
      const enteredUsername = document.getElementById("username").value;
      const enteredPassword = document.getElementById("password").value;
      const storedUsername = localStorage.getItem("studentUsername");
      const storedPassword = localStorage.getItem("studentPassword");
      if (enteredUsername === storedUsername && enteredPassword ===
storedPassword) {
        alert("Login successful!");
        window.location.href = "dashboard.html"; // or subject_registration.html
      } else {
        alert("Invalid credentials. Please try again.");
      }
    }
  </script>
```

```
</body>
```

Apply for LOR/NOC

```
<body>
  <h2 style="text-align:center;">Apply for LOR or NOC</h2>
  <form id="applicationForm">
    <label for="name">Full Name:</label>
    <input type="text" id="name" required>
    <label for="srn">SRN Number:</label>
    <input type="text" id="srn" required>
    <label for="email">Email ID:</label>
    <input type="email" id="email" required>
    <label for="type">Application Type:</label>
    <select id="type" required onchange="toggleSections()">
      <option value="">--Select--</option>
      <option value="LOR">Letter of Recommendation (LOR)</option>
      <option value="NOC">No Objection Certificate (NOC)</option>
    </select>
    <!-- LOR Section -->
    <div class="section" id="lorSection">
      <label for="faculty">Faculty for LOR:</label>
      <input type="text" id="faculty">
      <label for="organizationLOR">Organization Name:</label>
      <input type="text" id="organizationLOR">
      <label for="modeLOR">Mode of Occupation:</label>
      <select id="modeLOR">
        <option value="internship">Internship</option>
        <option value="higher studies">Higher Studies</option>
      </select>
      <label for="descLOR">Description:</label>
      <textarea id="descLOR"></textarea>
    </div>
    <!-- NOC Section -->
    <div class="section" id="nocSection">
      <label for="organizationNOC">Organization Name:</label>
      <input type="text" id="organizationNOC">
      <label for="modeNOC">Mode of Occupation:</label>
      <select id="modeNOC">
        <option value="part-time">Part-Time</option>
        <option value="full-time">Full-Time</option>
      </select>
      <label for="descNOC">Description:</label>
      <textarea id="descNOC"></textarea>
      <label for="passout">Year of Pass-Out:</label>
      <input type="number" id="passout" min="2020" max="2030">
    </div>
    <button type="submit">Submit Application</button>
  </form>
  <script>
    function toggleSections() {
      const type = document.getElementById("type").value;
      document.getElementById("lorSection").style.display = type === "LOR" ?
"block" : "none";
      document.getElementById("nocSection").style.display = type === "NOC" ?
"block" : "none";
    }
    document.getElementById("applicationForm").addEventListener("submit",
function(e) {
      e.preventDefault();
      alert("Application submitted successfully!");
      // You can add logic here to send data to a backend (e.g., PHP or Firebase)
    });
  </script>
</body>
```

Application Form Submission

```
<body>
<h2 style="text-align:center;">Application Form (LOR/NOC)</h2>
<form action="submit_application.php" method="post">
  <label for="name">Full Name:</label>
  <input type="text" name="name" required>
  <label for="srn">SRN Number:</label>
  <input type="text" name="srn" required>
  <label for="email">Email:</label>
  <input type="email" name="email" required>
  <label for="type">Application Type:</label>
  <select name="type" onchange="toggleFields(this.value)" required>
    <option value="">--Select--</option>
    <option value="LOR">LOR</option>
    <option value="NOC">NOC</option>
  </select>
  <div id="lorFields" style="display:none;">
    <label for="faculty">Faculty for LOR:</label>
    <input type="text" name="faculty">
    <label for="organization_lor">Organization (LOR):</label>
    <input type="text" name="organization_lor">
    <label for="mode_lor">Mode (Internship/Higher Studies):</label>
    <input type="text" name="mode_lor">
    <label for="desc_lor">Description:</label>
    <textarea name="desc_lor"></textarea>
  </div>
  <div id="nocFields" style="display:none;">
    <label for="organization_noc">Organization (NOC):</label>
    <input type="text" name="organization_noc">
    <label for="mode_noc">Mode (Part/Full-time):</label>
    <input type="text" name="mode_noc">
    <label for="desc_noc">Description:</label>
    <textarea name="desc_noc"></textarea>
    <label for="passout_year">Pass-Out Year:</label>
    <input type="number" name="passout_year">
  </div>
  <button type="submit">Submit Application</button>
</form>
<script>
  function toggleFields(type) {
    document.getElementById("lorFields").style.display = type === "LOR" ?
"block" : "none";
    document.getElementById("nocFields").style.display = type === "NOC" ?
"block" : "none";
  }
</script>
</body>
```

4.3.1.2. Faculty portal

Signup

```
 <body>
    <h1><i class="fas fa-user-plus"></i> Sign Up</h1>
    <form id="signupForm" onsubmit="return false;">
        <label for="firstName">First Name:</label>
        <input type="text" id="firstName" required maxlength="20">
        <label for="lastName">Last Name:</label>
        <input type="text" id="lastName" required maxlength="20">
        <label for="newUsername">Username:</label>
        <input type="text" id="newUsername" required pattern="[A-Za-z0-9]+">
        <label for="email">Email ID:</label>
        <input type="email" id="email" required>
        <label for="phone">Phone Number:</label>
        <input type="tel" id="phone" required pattern="[0-9]{10}">
        <label for="gender">Gender:</label>
        <select id="gender" required>
            <option value="">Select</option>
            <option value="male">Male</option>
            <option value="female">Female</option>
        </select>
        <button type="submit">Sign Up</button>
    </form>
    <script>
        document.getElementById("signupForm").addEventListener("submit",
function () {
            // Simulate sign-up process
            alert("Sign-up successful!");
            window.location.href = "login.html"; // Redirect to login
        });
    </script>
</body>
```

Login

```
<body>
    <h1><i class="fas fa-sign-in-alt"></i> Login</h1>
    <form id="loginForm" onsubmit="return false;">
        <label for="username">Username:</label>
        <input type="text" id="username" required>
        <label for="password">Password:</label>
        <input type="password" id="password" required>
        <button type="submit">Login</button>
    </form>
    <p>Don't have an account? <a href="signup.html"><i class="fas fa-user-plus"></i> Sign Up</a></p>
    <p><a href="forgot_password.html"><i class="fas fa-unlock"></i> Forgot Password?</a></p>
    <script>
        document.getElementById("loginForm").addEventListener("submit", function () {
            const username = document.getElementById("username").value;
            localStorage.setItem("username", username);
            window.location.href = "subject_registration.html";
        });
    </script>
</body>
```

- **View Request**

```
<body>
    <h1>Submitted Requests</h1>
    <table id="requestTable">
        <thead>
            <tr>
                <th>Type</th>
                <th>Name</th>
                <th>SRN</th>
                <th>Email</th>
                <th>Request Description</th>
                <th>Status</th>
            </tr>
        </thead>
        <tbody>
            <!-- Request rows will be inserted here dynamically -->
        </tbody>
```

```
</table>
<a href="dashboard.html" class="back-link">← Back to Dashboard</a>
<script>
  // Sample dummy data - Replace this with real data from localStorage, backend, etc.
    const sampleRequests = [
      {
        type: 'LOR',
        name: 'Yamini M',
        srn: '22XYZ123',
        email: 'yamini@example.com',
        description: 'LOR for internship at ABC Ltd.',
        status: 'Pending'
      },
      {
        type: 'NOC',
        name: 'Yamini M',
        srn: '22XYZ123',
        email: 'yamini@example.com',
        description: 'NOC for part-time job at DEF Corp.',
        status: 'Approved'
      }
    ];
    Const tableBody = document.getElementById("requestTable").querySelector("tbody");
    sampleRequests.forEach(request => {
      const row = document.createElement("tr");
      row.innerHTML = `
        <td>${request.type}</td>
        <td>${request.name}</td>
        <td>${request.srn}</td>
        <td>${request.email}</td>
        <td>${request.description}</td>
        <td>${request.status}</td>
      `;
      tableBody.appendChild(row);
    });
    </script>
</body>
```

- **Accept / Reject Applications**

```
<body>
  <h1>Application Requests</h1>
  <table>
    <thead>
      <tr>
        <th>Type</th>
        <th>Name</th>
        <th>SRN</th>
        <th>Description</th>
        <th>Status</th>
        <th>Actions</th>
      </tr>
    </thead>
    <tbody id="requestTable">
      <!-- Rows added dynamically -->
    </tbody>
  </table>
  <script>
    const requests = [
      {
        id: 1,
        type: "LOR",
        name: "Yamini M",
        srn: "22XYZ123",
        description: "LOR for internship at ABC Ltd.",
        status: "Pending"
      },
      {
        id: 2,
        type: "NOC",
        name: "Yamini M",
        srn: "22XYZ123",
        description: "NOC for part-time work at DEF Corp.",
        status: "Pending"
      }
    ];
    const tableBody = document.getElementById("requestTable");
    function renderTable() {
      tableBody.innerHTML = "";
      requests.forEach((req, index) => {
        const row = document.createElement("tr");
        row.innerHTML = `
          <td>${req.type}</td>
          <td>${req.name}</td>
          <td>${req.srn}</td>
          <td>${req.description}</td>
          <td class="status" id="status-${index}">${req.status}</td>
          <td>
            <button class="accept" onclick="updateStatus(${index},
'Approved')">Accept</button>
            <button class="reject" onclick="updateStatus(${index},
'Rejected')">Reject</button>
          </td>
        `;
        tableBody.appendChild(row);
      });
    }
    function updateStatus(index, status) {
      requests[index].status = status;
      renderTable();
    }
    renderTable();
  </script>
</body>
```

Faculty Review Module

Displays student applications

```
<body>
  <h1>Submitted Applications</h1>
  <table>
    <thead>
```

```
      <tr>
        <th>Student Name</th>
        <th>SRN</th>
        <th>Application Type</th>
        <th>Description</th>
        <th>Status</th>
      </tr>
    </thead>
    <tbody id="applicationTable">
      <!-- Data rows will be inserted dynamically -->
    </tbody>
  </table>
  <script>
    // Sample application data (can be fetched from a database in real projects)
    const applications = [
      {
        name: "Yamini M",
        srn: "22XYZ123",
        type: "LOR",
        description: "Requesting LOR for internship at Google.",
        status: "Pending"
      },
      {
        name: "Yamini M",
        srn: "22XYZ123",
        type: "NOC",
        description: "Need NOC for part-time work.",
        status: "Approved"
      }
    ];
    const tableBody = document.getElementById("applicationTable");
    applications.forEach(app => {
      const row = document.createElement("tr");
      row.innerHTML = `
        <td>${app.name}</td>
        <td>${app.srn}</td>
        <td>${app.type}</td>
        <td>${app.description}</td>
        <td>${app.status}</td>
      `;
      tableBody.appendChild(row);
    });
  </script>
</body>
```

Allows decision making (Accept/Reject)

```
<body>
  <h1>Application Review</h1>
  <table>
    <thead>
      <tr>
        <th>Name</th>
        <th>SRN</th>
        <th>Type</th>
        <th>Description</th>
        <th>Decision</th>
      </tr>
    </thead>
    <tbody id="appTable">
      <!-- Dynamic content will be inserted here -->
    </tbody>
  </table>
  <script>
    const applications = [
      {
        name: "Yamini M",
        srn: "22XYZ123",
        type: "LOR",
        description: "Requesting LOR for placement at TCS."
      },
      {
        name: "Akhil P",
        srn: "22XYZ456",
        type: "NOC",
    description: "Requesting NOC for part-time remote internship."
   }
  ];
  const table = document.getElementById("appTable");
  applications.forEach((app, index) => {
   const row = document.createElement("tr");
   row.innerHTML = `
    <td>${app.name}</td>
    <td>${app.srn}</td>
    <td>${app.type}</td>
    <td>${app.description}</td>
    <td id="decision-${index}">
     <button class="accept" onclick="makeDecision(${index},
'Accepted')">Accept</button>
     <button class="reject" onclick="makeDecision(${index},
'Rejected')">Reject</button>
    </td>
   `;
   table.appendChild(row);
  });
  function makeDecision(index, decision) {
   const decisionCell = document.getElementById(`decision-${index}`);
   decisionCell.innerHTML = `<strong style="color: ${decision === 'Accepted'
? 'green' : 'red'}">${decision}</strong>`;
  }
 </script>
</body>
```

4.3.1.3. Database Layer

- **Student Table:**

  MySQL table schema

```
CREATE DATABASE IF NOT EXISTS student_portal;
USE student_portal;
CREATE TABLE students (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  srn VARCHAR(20) UNIQUE NOT NULL,
  contact VARCHAR(15),
  email VARCHAR(100) UNIQUE NOT NULL,
  password VARCHAR(255) NOT NULL,
  gender VARCHAR(10),
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

PHP Backend

```
<?php
$host = "localhost";
$user = "root";
$pass = "";
$dbname = "student_portal";
// Create connection
$conn = new mysqli($host, $user, $pass, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
// Get POST values
$name = $_POST['name'];
$srn = $_POST['srn'];
$contact = $_POST['contact'];
$email = $_POST['email'];
$password = password_hash($_POST['password'], PASSWORD_BCRYPT);
$gender = $_POST['gender'];
// Insert query
$sql = "INSERT INTO students (name, srn, contact, email, password, gender)
        VALUES ('$name', '$srn', '$contact', '$email', '$password', '$gender')";
if ($conn->query($sql) === TRUE) {
    echo "Student registered successfully!";
} else {
    echo "Error: " . $conn->error;
}
$conn->close();
?>
```

4.3.1.4. Faculty Table:

MYSQL table schema

```
CREATE TABLE faculty (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    department VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    password VARCHAR(255) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Faculty Registration PHP

```
<?php
$host = "localhost";
$user = "root";
$pass = "";
$dbname = "student_portal";
// DB Connection
$conn = new mysqli($host, $user, $pass, $dbname);
if ($conn->connect_error) die("Connection failed: " . $conn->connect_error);
// POST Data
$name = $_POST['name'];
$department = $_POST['department'];
$email = $_POST['email'];
$password = password_hash($_POST['password'], PASSWORD_BCRYPT);
// Insert query
$sql = "INSERT INTO faculty (name, department, email, password)
        VALUES ('$name', '$department', '$email', '$password')";

if ($conn->query($sql) === TRUE) {
    echo "Faculty registered successfully!";
} else {
    echo "Error: " . $conn->error;
}
$conn->close();
?>
```

- **Application Table:**

  MySQL Table Schema

```
CREATE TABLE applications (
    id INT AUTO_INCREMENT PRIMARY KEY,
    student_id INT NOT NULL,
    application_type ENUM('LOR', 'NOC') NOT NULL,
    status ENUM('Pending', 'Accepted', 'Rejected') DEFAULT 'Pending',
    description TEXT,
    submission_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (student_id) REFERENCES students(id)
);
```

PHP Backend

```
<?php
session_start();  // Assumes student login session
$student_id = $_SESSION['student_id'];  // Get logged-in student ID
$host = "localhost";
$user = "root";
$pass = "";
$dbname = "student_portal";
// DB Connection
$conn = new mysqli($host, $user, $pass, $dbname);
if ($conn->connect_error) die("Connection failed: " . $conn->connect_error);
// POST data
$type = $_POST['application_type'];
$desc = $conn->real_escape_string($_POST['description']);
// Insert application
$sql = "INSERT INTO applications (student_id, application_type, description)
        VALUES ('$student_id', '$type', '$desc')";
if ($conn->query($sql) === TRUE) {
    echo "Application submitted successfully!";
} else {
    echo "Error: " . $conn->error;
}
$conn->close();
?>
```

Application Submission Form

```
<body>
  <h2>Application Form (LOR/NOC)</h2>
  <form action="submit_application.php" method="post">
    <label for="application_type">Application Type:</label>
    <select name="application_type" required>
      <option value="LOR">LOR</option>
      <option value="NOC">NOC</option>
    </select><br><br>
    <label for="description">Description:</label><br>
    <textarea name="description" rows="5" cols="40" required></textarea><br><br>
    <input type="submit" value="Submit Application">
  </form>
</body>
```

-

4.3.1.5. Student dashboard module

student_dashboard.php

```
<?php
session_start();
include 'db_connection.php';
// Assume session contains student ID
$student_id = $_SESSION['student_id'];
// Fetch student details
$student_sql = "SELECT * FROM students WHERE id = '$student_id'";
$student_result = mysqli_query($conn, $student_sql);
$student = mysqli_fetch_assoc($student_result);
// Fetch application history
$app_sql = "SELECT * FROM applications WHERE student_id = '$student_id' ORDER BY submitted_on DESC";
$app_result = mysqli_query($conn, $app_sql);
?>

<body>
<h2>Welcome, <?= $student['name'] ?></h2>
<!-- Profile Editing Section -->
<h3>Edit Profile</h3>
<form action="update_profile.php" method="POST">
  <label>Email:</label>
  <input type="email" name="email" value="<?= $student['email'] ?>" required>
  <label>Contact:</label>
```

```
    <input type="text" name="contact" value="<?= $student['contact'] ?>" required>
    <button type="submit">Update</button>
</form>
<!-- Application Submission Section -->
<h3>Apply for LOR / NOC</h3>
<form action="submit_application.php" method="POST">
    <label>Application Type:</label>
    <select name="type" required>
        <option value="LOR">LOR</option>
        <option value="NOC">NOC</option>
    </select>
    <label>Description:</label>
    <textarea name="description" required></textarea>
    <button type="submit">Submit Application</button>
</form>
<!-- Application History Section -->
<h3>Application History</h3>
<table>
    <tr>
        <th>Type</th>
        <th>Description</th>
        <th>Status</th>
        <th>Submitted On</th>
        <th>Comment</th>
    </tr>
    <?php while ($row = mysqli_fetch_assoc($app_result)) { ?>
    <tr>
        <td><?= $row['type'] ?></td>
        <td><?= $row['description'] ?></td>
        <td><?= $row['status'] ?></td>
        <td><?= $row['submitted_on'] ?></td>
        <td><?= $row['faculty_comment'] ?? 'N/A' ?></td>
    </tr>
    <?php } ?>
</table>
</body>
```

submit_application.php

```
<?php
include 'db_connection.php';
session_start();
$type = $_POST['type'];
$description = $_POST['description'];
$student_id = $_SESSION['student_id'];
$date = date("Y-m-d");
$sql = "INSERT INTO applications (student_id, type, description, status, submitted_on)
        VALUES ('$student_id', '$type', '$description', 'Pending', '$date')";
if (mysqli_query($conn, $sql)) {
   echo                          "<script>alert('Application                              Submitted');
window.location.href='student_dashboard.php';</script>";
} else {
   echo "Error: " . mysqli_error($conn);
}
?>
```

4.3.1.6. Faculty dashboard module

Faculty_dashboard.php

```
<?php
// Start session and include database connection
session_start();
include 'db_connection.php'; // Assumes you have a db_connection.php file
// Fetch pending LOR applications
$sql = "SELECT * FROM applications WHERE type='LOR' AND status='Pending'";
$result = mysqli_query($conn, $sql);
?>
<body>
   <h2 align="center">Faculty Dashboard - Pending LOR Requests</h2>

   <table>
      <tr>
         <th>Application ID</th>
         <th>Student Name</th>
         <th>Description</th>
         <th>Date</th>
         <th>Action</th>
      </tr>
```

```
    <?php while ($row = mysqli_fetch_assoc($result)) { ?>
    <tr>
      <td><?= $row['id'] ?></td>
      <td><?= $row['student_name'] ?></td>
      <td><?= $row['description'] ?></td>
      <td><?= $row['submitted_on'] ?></td>
      <td>
        <form action="respond_lor.php" method="POST">
          <input type="hidden" name="application_id" value="<?= $row['id'] ?>">
          <textarea name="comment" placeholder="Enter comment..."
required></textarea>
          <button type="submit" name="action" value="Accepted">Accept</button>
          <button type="submit" name="action" value="Rejected">Reject</button>
        </form>
      </td>
    </tr>
    <?php } ?>
  </table>
</body>
```

5. Database Management Module

Database Setup (database_setup.sql)

```
-- Create the database
CREATE DATABASE IF NOT EXISTS lor_noc_system;
USE lor_noc_system;
-- Users Table: Stores both students and faculty
CREATE TABLE users (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  srn VARCHAR(20), -- For students
  department VARCHAR(100), -- For faculty
  role ENUM('student', 'faculty') NOT NULL,
  email VARCHAR(100) NOT NULL UNIQUE,
  contact VARCHAR(15),
  password VARCHAR(255) NOT NULL
);
```

```
-- Applications Table: Stores LOR/NOC requests
CREATE TABLE applications (
   id INT AUTO_INCREMENT PRIMARY KEY,
   student_id INT NOT NULL,
   type ENUM('LOR', 'NOC') NOT NULL,
   description TEXT NOT NULL,
   status ENUM('Pending', 'Accepted', 'Rejected') DEFAULT 'Pending',
   submitted_on DATE NOT NULL,
   faculty_id INT,
   faculty_comment TEXT,
   FOREIGN KEY (student_id) REFERENCES users(id),
   FOREIGN KEY (faculty_id) REFERENCES users(id)
);
```

Sample Data Insertion

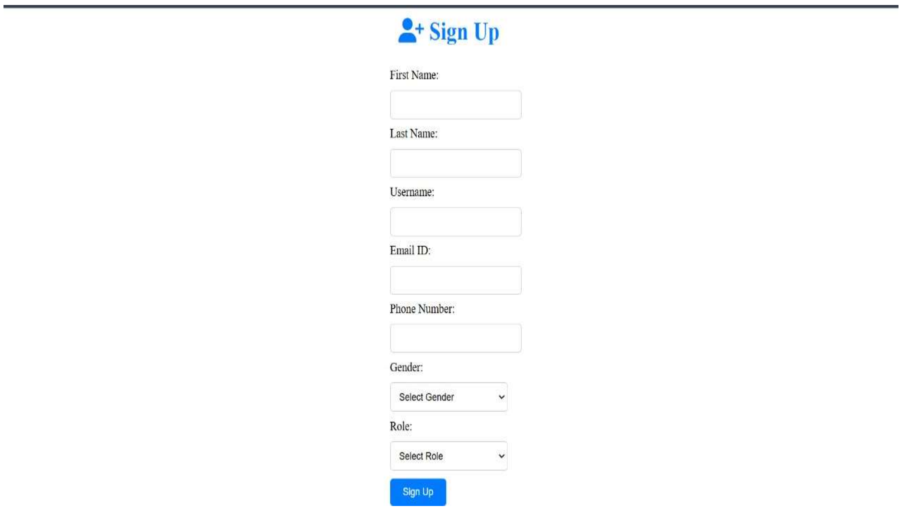
```
-- Insert sample faculty
INSERT INTO users (name, department, role, email, contact, password)
VALUES ('Dr. Smith', 'Computer Science', 'faculty', 'smith@univ.edu', '9876543210',
'hashed_password');
-- Insert sample student
INSERT INTO users (name, srn, role, email, contact, password)
VALUES ('Alice Johnson', '1MS20CS001', 'student', 'alice@univ.edu', '9876501234',
'hashed_password');
```

CHAPTER – 05

RESULTS AND DISCUSSIONS

5.1 SignUP page

The Sign Up page is a user-friendly registration interface designed to collect essential user details for account creation, such as first name, last name, username, email ID, phone number, gender, and role (Student or Faculty). It features a clean and centered layout with labeled input fields and dropdown menus for gender and role selection, enhancing usability and clarity. A prominent "Sign Up" heading with a user-plus icon at the top indicates the page's purpose, and a blue-colored Sign Up button at the bottom allows users to submit the form. This page facilitates new user onboarding and prepares the user for login or dashboard access upon successful registration.



Sign Up

First Name:

Last Name:

Username:

Email ID:

Phone Number:

Gender:

Select Gender

Role:

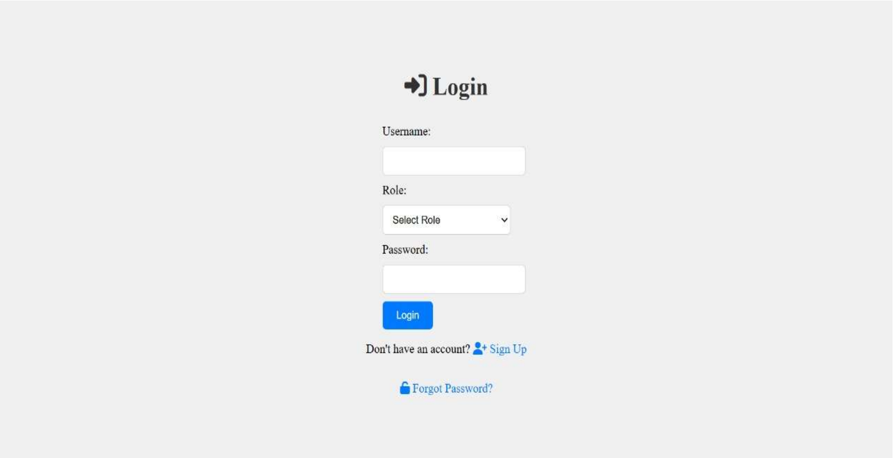
Select Role

Sign Up

Fig – 01 Signup page

5.2 Login page

The Login page provides a simple and secure interface for users to access their accounts by entering their username, selecting their role (such as Student or Faculty), and inputting their password. The design is clean and centered, with clear labels and input fields that guide the user through the process. A bold login heading with a login icon at the top enhances clarity and purpose. Below the login button, there are links for users who don't have an account to sign up and for those who may have forgotten their password, ensuring easy navigation and account recovery options. This page serves as the gateway for authenticated access to the respective user dashboards.



Login

Username:

Role:

Select Role

Password:

Login

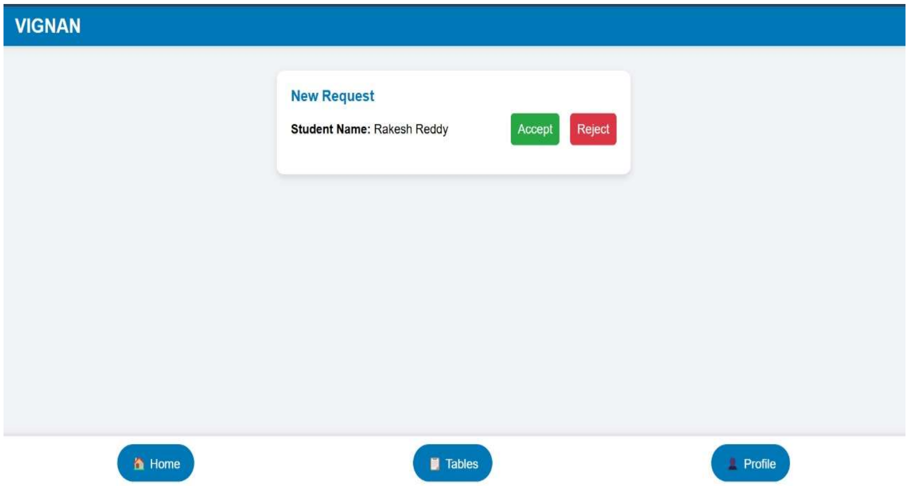
Don't have an account? Sign Up

Forgot Password?

Fig – 02 Login page for faculty&student

5.3 Faculty Home page

The Request Approval page provides a clean and user-friendly interface for managing incoming student requests, displaying a card titled "New Request" with the student's name—in this case, Rakesh Reddy—alongside two prominent buttons: "Accept" in green and "Reject" in red, allowing for quick decision-making by the user. The top banner shows the institutional name "VIGNAN," while the bottom navigation bar includes three clearly labeled buttons—Home, Tables, and Profile—each with distinct icons for intuitive access, supporting seamless navigation within the application.



VIGNAN

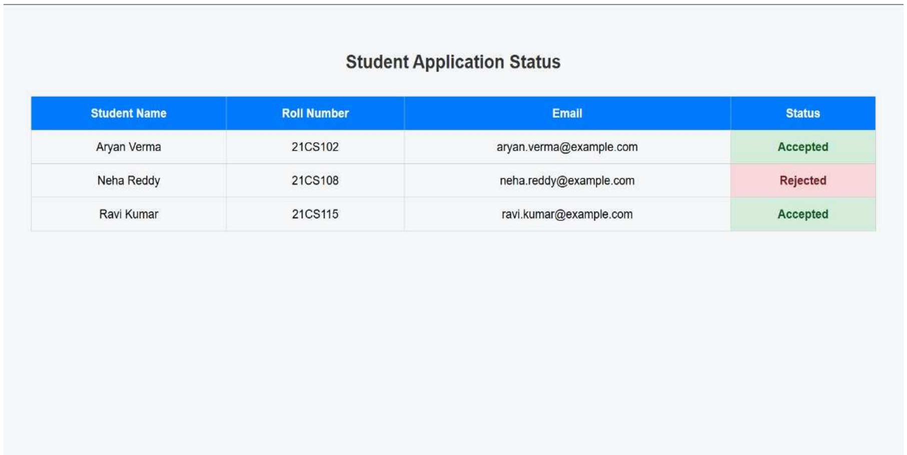
New Request

Student Name: Rakesh Reddy

Fig – 03 Faculty Homepage

5.4 Faculty table

The "Student Application Status" page presents a clear and structured table displaying the approval outcomes of student requests. The table is organized into four columns: **Student Name, Roll Number, Email,** and **Status.** Each row corresponds to an individual student entry, with their name, roll number, and email address clearly displayed. The **Status** column uses color-coded labels—**green for "Accepted"** and **red for "Rejected"—**to visually indicate the result of each application, enhancing readability. The header is bold and centered at the top of the page, ensuring the user immediately understands the purpose of the view. This layout supports efficient review and tracking of application outcomes.



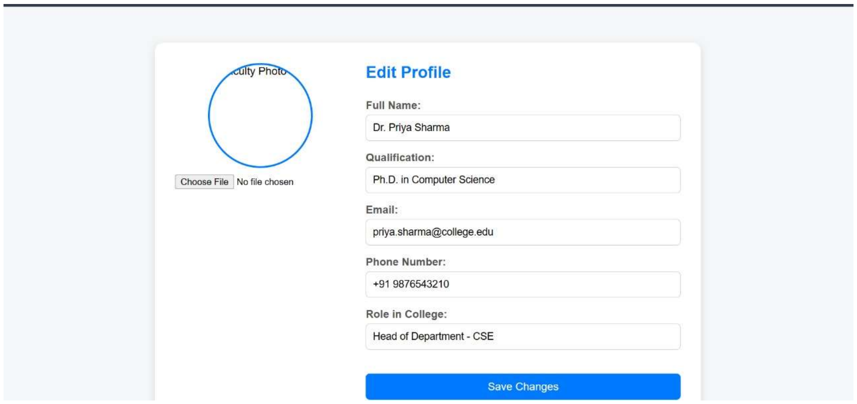
Student Application Status

Student Name	Roll Number	Email	Status
Aryan Verma	21CS102	aryan.verma@example.com	Accepted
Neha Reddy	21CS108	neha.reddy@example.com	Rejected
Ravi Kumar	21CS115	ravi.kumar@example.com	Accepted

Fig – 04 Faculty table

5.5 Faculty profile

The image displays a faculty profile editing interface designed for updating personal and professional details. On the left side, there is a circular placeholder labeled "Faculty Photo" where users can upload their image using the "Choose File" button. The main section on the right features a clearly labeled "Edit Profile" form with a clean and minimalistic design. This form includes pre-filled fields such as Full Name (Dr. Priya Sharma), Qualification (Ph.D. in Computer Science), Email (priya.sharma@college.edu), Phone Number (+91 9876543210), and Role in College (Head of Department - CSE). Each field is editable, allowing the user to update their information as needed. At the bottom of the form, there is a large blue "Save Changes" button for submitting the updated profile..



Edit Profile

Full Name: Dr. Priya Sharma

Qualification: Ph.D. in Computer Science

Email: priya.sharma@college.edu

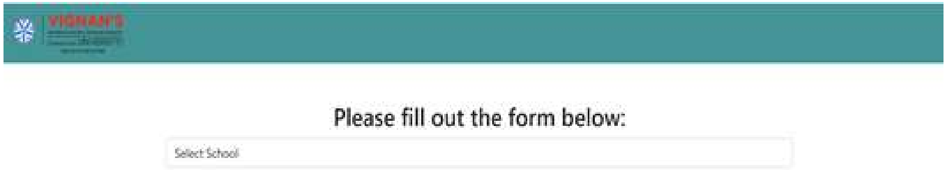
Phone Number: +91 9876543210

Role in College: Head of Department - CSE

Fig – 05 Faculty profile

5.6 Application form

The image displays the initial interface of a web form developed for Vignan's Deemed to be University, where users are prompted to begin the application process by selecting their respective school. The page features a minimalistic and user-friendly design, with the university's logo and name positioned at the top-left corner and a teal header strip giving it a professional appearance. At the center, a clear heading reads “Please fill out the form below:” followed by a dropdown input labeled “Select School,” indicating that this is the first step in a larger data collection process, likely for applications such as Letters of Recommendation (LOR) or No Objection Certificates (NOC). The clean layout ensures ease of use, allowing students or faculty to smoothly begin their submissions.



Please fill out the form below:

Select School

Fig – 06 application form to select the dept.

5.7 Vignan Home page:

The image displays the homepage of Vignan's Foundation for Science, Technology & Research, a deemed-to-be university, showcasing its core mission and inviting message. Dominating the center is a bold tagline, "Building Character for a Brighter Tomorrow," accompanied by a supportive statement emphasizing the university’s commitment to transforming education, nurturing minds, and instilling enduring values. Below the text are two prominent call-to-action buttons labeled “Apply Now” and “International Students,” encouraging prospective students to take the next step. To the right, there is a photo of professionally dressed students standing confidently, symbolizing academic excellence and future leadership. The top navigation bar includes links to key sections like Academics, Admission, Research, People, and University Life, making it easy for visitors to explore the university's offerings. The overall design is clean, modern, and engaging, effectively portraying Vignan’s as a forward-thinking institution focused on holistic development.

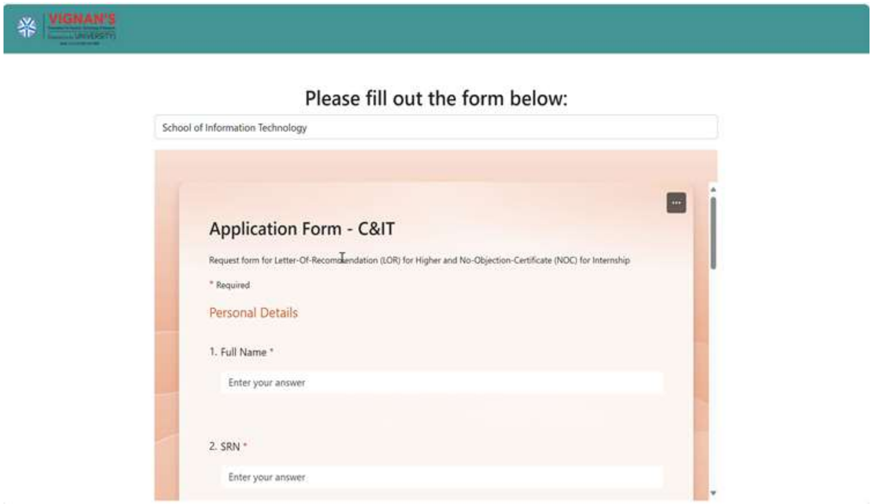
Building Character for a Brighter Tomorrow

We are Transforming Education, Nurturing Minds & Fostering values that transcend challenges

Fig – 07 Vignan's Home page

5.8 Application for specific dept

The image showcases a web-based application form interface designed for students from the School of Information Technology at Vignan's Foundation for Science, Technology & Research. At the top, there is a dropdown menu indicating the selected school, followed by a clearly visible heading, "Please fill out the form below." The main form is titled "Application Form - C&IT," referring to Computer and Information Technology, and is intended for submitting requests for Letters of Recommendation (LOR) for higher studies and No-Objection Certificates (NOC) for internships. The form begins with a "Personal Details" section that includes required fields such as Full Name and SRN (Student Registration Number), aiming to collect essential information for processing the request. The clean, structured design with clear labels ensures ease of use, reflecting an organized and user-friendly digital solution for administrative processes within the institution.



Please fill out the form below:

School of Information Technology

Application Form - C&IT

Request form for Letter-Of-Recomendation (LOR) for Higher and No-Objection-Certificate (NOC) for Internship

* Required

Personal Details

1. Full Name *

Enter your answer

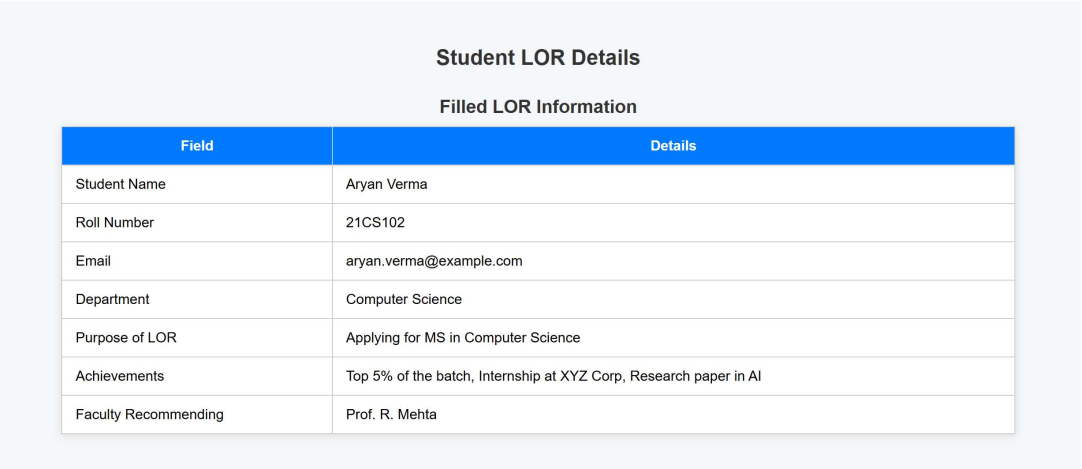
2. SRN *

Enter your answer

Fig – 08 Application form for specific dept

5.9 Student table

The "Student LOR Details" section displays a summarized view of a student's request for a Letter of Recommendation (LOR) in a structured table format. In this case, Aryan Verma, a student from the Computer Science department with the roll number 21CS102, has submitted a request for an LOR to support his application for an MS in Computer Science. His achievements include being in the top 5% of his batch, completing an internship at XYZ Corp, and publishing a research paper in Artificial Intelligence. The recommending faculty member for this request is Prof. R. Mehta. This organized format allows faculty and administrative staff to quickly review and process LOR requests efficiently by presenting all relevant details in one place.



Student LOR Details

Filled LOR Information

Field	Details
Student Name	Aryan Verma
Roll Number	21CS102
Email	aryan.verma@example.com
Department	Computer Science
Purpose of LOR	Applying for MS in Computer Science
Achievements	Top 5% of the batch, Internship at XYZ Corp, Research paper in AI
Faculty Recommending	Prof. R. Mehta

Fig – 09 Student table

5.10 Student portal

The image showcases the homepage of a student portal for Vignan’s Foundation for Science, Technology & Research (Deemed to be University), located at the Guntur Campus, Andhra Pradesh, India. The top section prominently features the university’s name, logo, and its NAAC A+ accreditation badge, reflecting the institution’s high academic standards. The background features a clear view of the university's Aryabhatta Bhavan building, surrounded by greenery and characterized by its distinctive orange and white architectural design. At the bottom of the page, there are three interactive buttons labeled “LOR/NOC,” “Tables,” and “Profile,” indicating user navigation options for managing Letter of Recommendation/No Objection Certificate requests, accessing data tables, and viewing or updating student profile information. This user interface is designed to facilitate easy access to key student services in a visually appealing and institutional-

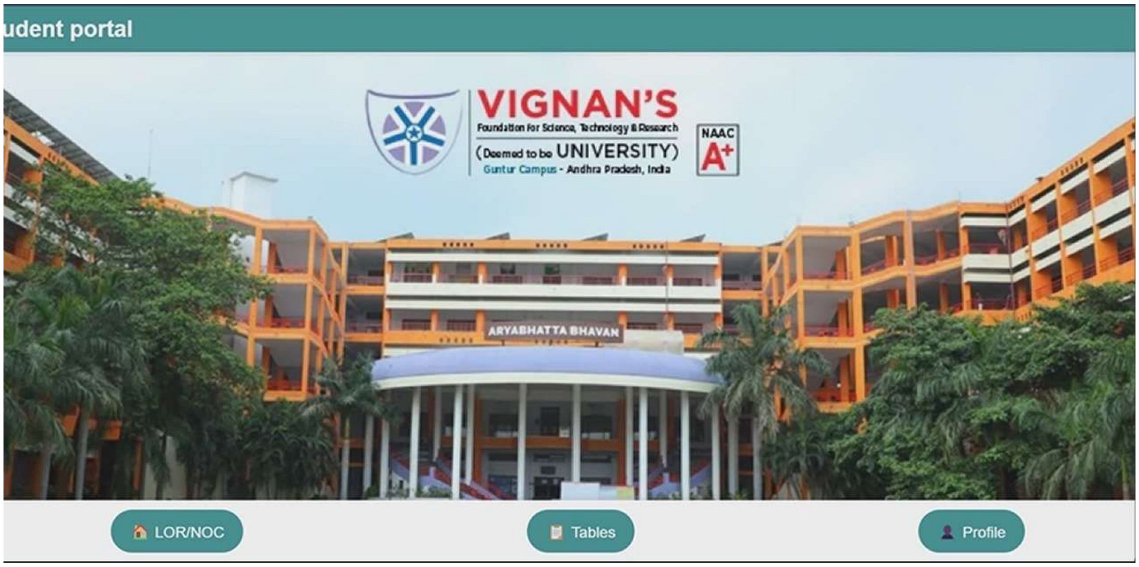
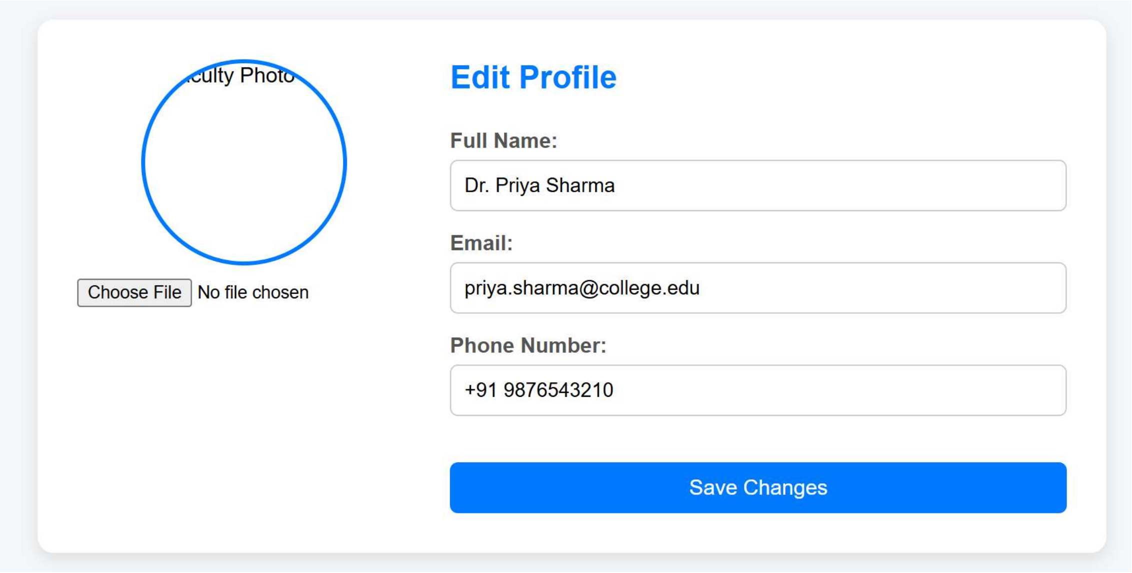

udent portal
VIGNAN'S
Foundation for Science, Technology & Research
(Deemed to be UNIVERSITY)
Guntur Campus - Andhra Pradesh, India
NAAC
A+
ARYABHATTA BHAVAN
LOR/NOC
Tables
Profile

Fig – 10 Student portal

5.11 Student profile

The image displays a Student profile editing interface designed for updating personal and professional details. On the left side, there is a circular placeholder labeled "Faculty Photo" where users can upload their image using the "Choose File" button. The main section on the right features a clearly labeled "Edit Profile" form with a clean and minimalistic design. This form includes pre-filled fields such as Full Name (Dr. Priya Sharma), Qualification (Ph.D. in Computer Science), Email (priya.sharma@college.edu), Phone Number (+91 9876543210), and Role in College (Head of Department - CSE). Each field is editable, allowing the user to update their information as needed. At the bottom of the form, there is a large blue "Save Changes" button for submitting the updated profile. The overall layout is user-friendly and visually organized, making it suitable for faculty members in academic institutions.



Edit Profile

Full Name: Dr. Priya Sharma

Email: priya.sharma@college.edu

Phone Number: +91 9876543210

Fig – 11 Student portal

CHAPTER – 06
CONCLUSION

6.1 Conclusion

The "Applying for LOR & NOC" project was developed to address the challenges faced by students and faculty in manually managing applications for Letters of Recommendation (LOR) and No Objection Certificates (NOC). Traditionally, the process of requesting these documents has been time-consuming, prone to delays, and often lacked a standardized workflow. With this application, a structured and user-friendly system has been introduced to digitize and simplify the entire procedure, benefiting both students and academic staff.

One of the key outcomes of this project is the automation of form submissions and status tracking. Students can now submit their applications by filling in essential details such as their SRN, name, contact information, purpose of the request, faculty details (for LOR), and employment details (for NOC). This eliminates the need for physical paperwork and multiple rounds of follow-up. The system also includes role-based access for faculty and administrators to review applications, accept or reject them, and provide timely feedback—all through a single platform.

In addition to improving efficiency, the system enhances transparency and accountability. Each request is recorded and tracked in real-time, reducing the chances of miscommunication or misplaced documentation. The inclusion of email confirmations and status updates ensures that students stay informed about the progress of their application. Faculty members benefit from the streamlined interface, allowing them to focus on evaluating genuine requests and supporting students more effectively.

Overall, this project has the potential to be implemented across academic institutions to significantly improve administrative processes. It promotes digital transformation in student-faculty communication, encourages timely support for students' career or academic needs, and reduces the operational burden on college administration. As an evolving solution, future enhancements such as document uploads, digital signatures, and integrated approval hierarchies can further boost its functionality. In conclusion, the "Applying for LOR & NOC" system represents a smart, scalable, and impactful step toward modernizing institutional workflows.

6.2 Future Scope

The "Applying for LOR & NOC" system lays a strong foundation for digitizing academic administrative processes, and there are several promising enhancements that can be incorporated in the future to further improve its usability, scalability, and functionality.

- **Integration with Institutional Portals**:

The system can be integrated with the college's official ERP or student portal, allowing auto-fetching of student details such as academic records, course history, and faculty assignments. This would reduce manual input errors and ensure data consistency across platforms.

- **Digital Signatures and Document Uploads:**
  Future versions can support digital signatures by faculty and administration to legally validate the issued LOR and NOC documents. Additionally, applicants could be allowed to upload supporting documents like resumes, offer letters, or academic transcripts to strengthen their applications.
- **Multi-level Approval Workflows:**
  Introducing hierarchical approval chains (e.g., mentor → HOD → Principal) can enhance the authenticity of the documents issued and align the process with institutional policies. Notifications can be sent at each stage to keep all stakeholders informed.
- **Mobile Application Support**:
  Developing a mobile version of the platform would allow students and faculty to interact with the system conveniently on the go. Push notifications could alert users of updates regarding their application status, approvals, or required actions.
- **Analytics and Reporting**:
  Adding a dashboard with analytics can help administrators track the number of applications, approval rates, processing times, and common reasons for rejection. These insights could support data-driven improvements in administrative policies.

6.3 Limitations of the project

The "Applying for LOR & NOC" system, while effective in streamlining the application process for students and faculty, does have certain limitations that restrict its full potential in a real-world institutional environment. One of the primary limitations is the lack of integration with the institution's official databases. This means that student records, faculty availability, or academic performance metrics are not automatically linked, requiring manual entry that can lead to data inconsistencies or errors. Furthermore, the system currently supports only a basic authentication mechanism, which may not be sufficient to ensure data privacy and secure access. Features such as multi-factor authentication, encrypted data transmission, and role-based permissions are not implemented, potentially making the system vulnerable to unauthorized access.

Another significant limitation is the one-level approval workflow. In most academic institutions, applications for Letters of Recommendation (LOR) or No Objection Certificates (NOC) often go through multiple levels of review and approval. However, the current system allows only faculty-level interaction, omitting crucial roles such as department heads or administrative staff. Additionally, the absence of a real-time notification system means students are not promptly informed about the status of their applications, which could lead to unnecessary delays. There is also no provision for students to upload supporting documents—such as resumes, internship offers, or achievement certificates—which are often important for a faculty member to provide an accurate and effective recommendation.

From a technical standpoint, the project is built with a limited scope and may not scale efficiently if adopted by a larger institution with thousands of users. The current system also lacks robust analytics and reporting features, which could help administrators track trends, evaluate faculty response times, or manage peak application periods. Lastly, the user interface, while functional, may not be fully optimized for mobile devices, affecting user experience for students or faculty accessing the portal on smartphones or tablets. Addressing these limitations would require deeper system integration, enhanced security, and user-centered design improvements in future iterations of the project.

- **Lack of Database Integration**: The system does not connect with institutional databases, requiring manual data entry which can lead to errors.
- **Single-Level Approval**: The system allows only faculty to approve/reject requests without involving higher authorities like department heads.
- **No Document Upload Support**: Students cannot attach resumes, certificates, or internship letters, which are often required for LORs.
- **Manual Data Verification**: There's no automated way to verify the accuracy or authenticity of student-provided information.
- **No Offline Access**: Entirely dependent on internet access; there's no support for offline submissions or viewing.

CHAPTER – 07
REFERENCES

1. **W3Schools.** (n.d.). *HTML, CSS, JavaScript Tutorials.* Retrieved from https://www.w3schools.com
2. **MDN Web Docs.** (n.d.). *Web technology documentation.* Retrieved from https://developer.mozilla.org
3. **PHP Manual.** (n.d.). *PHP Documentation.* Retrieved from https://www.php.net/manual/en/
4. **MySQL Documentation.** (n.d.). *MySQL 8.0 Reference Manual.* Retrieved from https://dev.mysql.com/doc/
5. **Bootstrap.** (n.d.). *Front-end framework for responsive design.* Retrieved from https://getbootstrap.com
6. **Font Awesome.** (n.d.). *Icons for web design.* Retrieved from https://fontawesome.com
7. Sharma, R., & Verma, A. (2020). *Online Application Systems: Design and Implementation.* International Journal of Computer Applications, 175(20), 10-15.
8. Singh, P. (2019). *Student Information Management Systems – A Case Study.* Journal of Educational Technology, 8(3), 45-52.
9. Kumar, D., & Yadav, R. (2021). *Automating Academic Processes with Web-Based Portals.* Proceedings of the National Conference on Emerging Technologies in Education.
10. Stack Overflow. (n.d.). *Developer community support.* Retrieved from https://stackoverflow.com