SCHOLARSHIP MANAGEMENT WEB PORTAL FOR GNDEC

MAJOR PROJECT REPORT

SUBMITTED IN THE PARTIAL FULLFILLMENT OF THE REQUIRMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY

(Computer Science and Engineering)

Submitted By:

Kamalpreet Kaur (2104133)

Parveen Kaur (2104152)

Simrandeep Kaur (2104195)

Submitted To:

Prof. Amandeep Kaur Sohal

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GURU NANAK DEV ENGINEERING COLLEGE

LUDHIANA 141006

(JUNE 2025)

Abstract

The Scholarship Management System (SMS) is a comprehensive web-based platform designed to streamline, digitize, and enhance the entire lifecycle of scholarship management in a college setting. Traditionally, the process of scholarship administration has been plagued by inefficiencies, excessive paperwork, delayed communication, and a high risk of human error. This project addresses those challenges by providing a centralized, secure, and user-friendly system that ensures transparent, timely, and organized management of scholarship-related activities for students, administrative staff, and verification officers. The core objective of this project is to automate and simplify the tasks associated with scholarship distribution from application submission and eligibility verification to evaluation, approval, and disbursement. The system introduces three primary user roles: **Student**, **Clerk**, and **Admin**, each with distinct responsibilities. Students can register and create profiles, browse available scholarships based on eligibility criteria, fill out application forms, and upload required documents. Clerks act as intermediaries who verify the authenticity of documents, ensure completeness of applications, and coordinate communication between students and administrators. The Admin holds overarching control, with the ability to add or update scholarship details, define eligibility rules, manage user roles, oversee application progress, and generate analytics and reports for informed decision-making.The system architecture is built using modern, scalable, and secure technologies. The frontend interface is developed using html, css ,js, , which offers dynamic rendering, The backend is implemented using PHP depending on deployment preferences, both of which provide robust support for, authentication, and integration with the database. MySQL is used as the relational database to manage user information, application data, scholarship records, and system logs. In conclusion, this Scholarship Management System offers a robust solution to modernize the traditional manual process of scholarship management. By leveraging web technologies and database integration, the system enhances accountability, speeds up the decision-making process, and ensures that deserving students can easily access the financial assistance they need to support their education.

Acknowledgement

We are highly grateful to the Dr. Sehijpal Singh, Principal, Guru Nanak Dev Engineering College (GNDEC), Ludhiana, for providing this opportunity to carry out the major project work at SCHOLARSHIP MANAGEMENT WEB PORTAL FOR GNDEC.

The constant guidance and encouragement received from Dr. Kiran jyoti H.O.D. CSE Department, GNDEC Ludhiana has been of great help in carrying out the project work and is acknowledged with reverential thanks.

We would like to express a deep sense of gratitude and thanks profusely to Prof. Amandeep Kaur Sohal, without his/her wise counsel and able guidance, it would have been impossible to complete the project in this manner. We express gratitude to other faculty members of computer science and engineering department of GNDEC for their intellectual support throughout the course of this work. Finally, we are indebted to all whosoever have contributed in this report work.

Simrandeep kaur (2104195)

Parveen kaur (2104152)

Kamalpreet kaur (2104133)

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

1.1Introduction to Project

The Guru Nanak Dev Engineering College (GNDEC) Scholarship Management Web Portal is a complete digital solution designed to improve the accessibility and management of scholarship-related procedures. The portal is intended to provide a streamlined, centralized platform that meets the needs of students, faculty, and administrative staff. It acknowledges the difficulties that come with processing scholarship applications by hand, including delays, inconsistent data, and a lack of transparency.

Smooth application submission, automated eligibility checks, safe document uploads, real-time application status monitoring, and expedited approval workflows are all made possible by this system. It also gives administrators powerful tools for efficiently managing deadlines, creating reports, and keeping an eye on fund disbursement. The portal's digitization and automation of essential procedures not only lessens administrative load but also guarantees increased transparency, equity, and responsibility in scholarship allocation. GNDEC's dedication to using technology to promote academic progress and student empowerment is demonstrated by the creation of this portal.

Students can browse available scholarship opportunities, fill out online applications, upload required documents, and get timely application status updates using this web-based platform's unified interface. The portal facilitates data-driven decision-making, streamlines review and verification procedures, and cuts down on pointless correspondence and paperwork for college administrators and staff.

Constructed using contemporary web technologies, the portal guarantees data security, integrity, and confidentiality and is usable on a variety of devices. It also encourages accountability by keeping thorough records of all actions, which reduces the possibility of bias or human error. By putting this portal into place, GNDEC demonstrates its proactive adoption of digital solutions to improve administrative efficiency and student services.

1.2Project Category

This project is classified as "Web-Based Application Development," with an emphasis on administrative and educational automation. This institute-level project was created to support Guru Nanak Dev Engineering College's (GNDEC) internal operational requirements, particularly in the area of scholarship administration.

This category includes software solutions that are developed to automate and streamline specific institutional processes using web technologies, database systems, and server-side logic.

This project is developed to address the operational challenges in managing student scholarships, which traditionally involve a great deal of paperwork, manual verification, delayed communication, and lack of centralized record-keeping. As such, it is not only a tool for digital transformation but also falls under the category of workflow automation systems.

Grouping:

Domain: Automation of Administration and Education Technology
Platform: Cross-platform and responsive web-based application
Users include scholarship committees, administrative staff, faculty coordinators, and students.
Goal: Automating the institute's data management, communication, and scholarship-related procedures.

Relevance to the Institution:

By providing a solution that substitutes a digital, user-friendly portal for manual, paper-based workflows, the project tackles a crucial administrative task within the college. It supports GNDEC's objective of incorporating technology into academic processes to increase effectiveness, accessibility, and transparency. In addition to expediting scholarship applications and approvals, the project acts as a template for the institute's upcoming digitization projects.

Academic Significance:

The project enables students to work on solutions for practical challenges by utilizing their theoretical knowledge of database management systems, web technologies, and software

engineering. It promotes interdisciplinary collaboration emphasizing data security measures in conjunction with system architecture, UI, and the implementation of modern technologies.

Key Characteristics of the Project Category:

1. **Web-Based System:**

   The application is hosted on a server and accessed through web browsers. It is platform-independent, allowing both students and administrators to access the system from any internet-enabled device (e.g., laptop, desktop, tablet, or smartphone).

2. **Application Software:**

   This system is a standalone application developed to meet specific user needs namely, the efficient handling of scholarship applications and related administrative functions. It consists of user interfaces, database layers, and business logic.

3. **Database-Driven Application:**

   The system manages large volumes of structured data related to student details, scholarships, applications, and disbursement statuses. A relational database system forms the backbone of this application.

4. **Educational Domain Project:**

   Since the core objective is to benefit students and educational institutions, the project falls squarely within the Educational Technology (EdTech) domain. It supports academic equity by ensuring that eligible students can access financial assistance efficiently.

5. **Multi-User Role-Based Access:**

   The system supports different types of users (students, administrators, and possibly reviewers or finance officers), each with customized access levels and functionalities. This makes it a role-based multi-user system, a common feature in enterprise-level web applications.

6. **Real-Time Information System:**

   The portal allows real-time application tracking, status updates, and communication between applicants and administrators, which classifies it under interactive real-time systems designed for transparency and quick action.

1.3Problem Formulation

At institutions such as Guru Nanak Dev Engineering College (GNDEC), scholarships help students focus on academics without any financial disturbances. Even so, the conventional system of processing scholarship applications and granting approvals is often tedious, done by hand, and relies on physical documents.

Some issues we currently face with the system are:

- Objectives Overlapping and Repetition: Students tend to repeatedly submit the same information and relevant documents multiple times, often for different departments or scholarships.
- Deficit Of Information: Students often seek answers to certain questions as well as directions for steps to take next as they have no way of knowing the various stages of progression of their submission during the application review.
- Hand Verification Bottlenecks: Every single application and accompanying document must be cross-checked with the respective administrative personnel, hence adding to the time taken and also increasing the chances of errors being made.
- Data Mismanagement: An update done manually on records that have been kept for several departments will likely have inconsistencies in the data, resulting in a poor-maintained dataset.
- Delay In Communication: These can often be used when talking about messages sent for selection, deadlines, missing paperwork, eligibility, etc. throughout the entire process.

1.4 Identification /Recognition of Need

Scholarships are essential for advancing financial equity, inclusivity, and academic excellence in the context of higher education. As more students apply for different government and private scholarship programs, Guru Nanak Dev Engineering College (GNDEC) is finding it more difficult to efficiently handle these procedures using conventional, paper-based techniques.

Currently, manual data collection, physical document verification, and interdepartmental communication are all part of the scholarship application and approval process. These methods frequently result in inefficiencies like:

- Lack of centralized access to scholarship-related information
- delayed processing and approvals
- increased administrative workload
- poor student visibility into application status
- high likelihood of document misplacement or data inconsistency and students frequently miss opportunities because they are unaware of eligibility requirements, deadlines, or incomplete submissions all of which could be reduced with an automated system providing timely notifications and guidance.

The institution's capacity to:

- Produce real-time reports for monitoring and audits
- Track disbursements and historical records efficiently and Maintain transparency and accountability in scholarship distribution is also limited by the lack of a digital system.

There is an obvious institutional need for a recent, centralized, and user-friendly system to oversee all scholarship-related operations as digital transformation becomes more and more crucial in educational settings. Therefore, the need for this project is recognized as arising from: The current manual process's operational inefficiencies. The increasing number of applications and criteria's complexity. The college's dedication to digital innovation and student welfare.

1.5Existing System

Guru Nanak Dev Engineering College's (GNDEC) scholarship administration procedure is currently mostly manual and paper-based, involving multiple administrative departments and human interventions. Distribution of application forms (physical or downloadable PDFs), hard copy document submission, faculty coordinators' manual verification, and physical form forwarding for approvals are all common steps in the workflow.

Important Aspects of the Current System:

- Manual Form Distribution: Students fill out scholarship forms by hand, either by picking them up from the college office or by downloading them in static formats.
- Physical Document Submission: Students must deliver hard copies of all necessary paperwork to the appropriate offices, including income certificates, identification documents, and grade reports.
- In-Person Verification: Clerks and faculty coordinators manually examine applications to ensure they are eligible, complete, and authentic.

Limitations of the Current System:

Time-consuming Procedures: Every step of the process, from submitting forms to final approval, is delayed by manual operations.

- High Administrative Burden: Organizing records, answering student questions, and managing documents take up a lot of time for faculty and staff.
- Data Inconsistency and Loss Risk: Records stored on paper are susceptible to deterioration, loss, or human error when being entered and verified.
- Lack of Transparency: Students frequently don't know how their applications are progressing or why they were rejected.
- Scalability Problems: The system gets more difficult to administer and prone to mistakes as the number of students and scholarships rises.
- Redundant Submissions: Students who apply for more than one scholarship must submit the same data and supporting documentation.

1.6Objectives

The Objectives of the project:

1. To design a web portal to facilitate students to apply for available scholarships.

2.To facilitate college authorities to manage scholarship beneficiaries and required documentation.

3.To develop a user-friendly Interface for students and authorities to track status of scholarship applications and Report generation required for different purposes.

1. To design a web portal to facilitate students to apply for available scholarships.

The first objective of this project is to design a web portal that enables students to apply for available scholarships in a simple and efficient manner. This digital platform will replace the traditional paper-based application process, making it more accessible and time-saving for students. Through the portal, students will be able to view a list of available scholarships, understand the eligibility criteria, fill out application forms online, and upload the necessary documents. The goal is to create a centralized system that streamlines the application process and reduces the administrative burden on both students and staff.

2.To facilitate college authorities to manage scholarship beneficiaries and required documentation.

The second objective is to facilitate college authorities in managing scholarship applications, beneficiaries, and related documentation**.** The portal will provide a secure and structured environment where administrators can log in to review submitted applications, verify eligibility, and approve or reject applicants based on institutional or government policies. It will also serve as a repository for storing digital copies of documents submitted by students, reducing the need for physical paperwork and minimizing the risk of data loss. This functionality ensures transparency, improves processing efficiency, and allows the institution to maintain accurate records of all scholarship-related activities.

3.To develop a user-friendly Interface for students and authorities to track status of scholarship applications and Report generation required for different purposes.

The third objective focuses on developing a user-friendly interface for both students and administrators to track the progress of scholarship applications and generate various reports as needed**.** Students will be able to log in and check the real-time status of their applications—from submission to approval or rejection—thus improving communication and reducing uncertainty. Administrators will benefit from a dashboard that supports the generation of customized reports, including statistics on applications received, approved, rejected, and disbursed. These reports can be used for internal reviews, audits, or to make data-driven decisions regarding future scholarship planning and resource allocation.

1.7 Proposed System

A web-based Scholarship Management System is suggested for Guru Nanak Dev Engineering College (GNDEC) in order to get around the drawbacks of the current manual procedure. By offering a centralized platform where students can apply for scholarships and administrators can effectively manage, verify, and process applications, this system seeks to automate and digitize the entire scholarship lifecycle.

Key Features of the Proposed System:

- Online Application Portal: Students can view available scholarships, verify eligibility requirements, and submit online applications with the necessary supporting documentation by logging in with their credentials.
- Uploading and Managing Documents: Direct portal uploading of scanned copies of identification documents, academic transcripts, income certificates, etc.
- Management of User Roles: Various user roles with distinct access rights and duties, including Administrator, Faculty Coordinator, Student, and Verifier.
- Automatic Verification of Eligibility: Preliminary evaluations to help students before they apply, based on predetermined criteria.
- Tracking Applications in Real Time: Through email/SMS or portal notifications, students can monitor the status of their application, including whether it has been submitted, is being reviewed, approved, or rejected.
- Reports & Admin Dashboard: Administrators can view analytics, manage applications, create reports, and keep an eye on fund disbursements and deadlines using a centralized dashboard.
- Data Integrity and Security: Putting in place data encryption, login authentication, and frequent backups to guarantee safe access and data security.
- System of Notification: Automated notifications and reminders regarding the status of applications, missing documents, and approval or rejection.

The proposed system's advantages include:

- Increased Efficiency: Reduces staff manual labor and gets rid of delays.
- Increased Transparency: Students have real-time access to their applications.
- Centralized Data Management: All scholarship-related documents are securely stored and can be quickly retrieved and reported on.
- User-friendly Interface: a responsive, easy-to-use interface that works on desktop and mobile devices.
- Scalability: The ability to manage growing user and application populations without experiencing perfect.

1.8 Unique features of the proposed system

The unique features of the proposed Scholarship Management Web Portal are intended to specifically address the difficulties encountered by GNDEC's administrative staff and students. This portal offers a comprehensive, intelligent, and user-centric solution with the following special features, in contrast to generic online forms or simple data entry systems:

1.Dynamic Matching for Scholarships

To reduce out confusion and lost opportunities, the system automatically suggests eligible scholarships to students based on their financial situation, academic standing, and category (SC, ST, OBC, minority, etc.).

2. Multi-Scholarship Application Support:

This feature enables students to submit applications for several scholarships at once without having to enter duplicate data or supporting documentation.

3. Repository and Reusability of Documents:

Students can reuse documents across applications and only upload once. Every user has a secure document repository kept up to date by the system.

4. Access Control Based on Roles:

Roles with clear definitions, including Scholarship Officer, Faculty Verifier, Administrator, and Student, each with customized interfaces and access rights to enhance data security and expedite workflow.

5. Immediately Status Updates and Alerts:

Automated email, SMS, and in-portal notifications for application submission, missing documents, approval phases, and final outcomes.

6. Action Logging and Audit Trail:

Transparency and accountability are ensured by time stamping each action taken on an application, such as approval, rejection, or modification.

7. Admin Dashboard with Analytics:

To aid in decision-making, the central dashboard offers data on the number of applications, scholarship distribution, pending verifications, and user activity.

8. System of Deadlines and Reminders

In order to decrease late or missed submissions, students are given countdowns and reminders when deadlines are approaching.

9. Two-Factor Authentication Secure Login

Keeps sensitive student data safe from unwanted access and guarantees that only authorized users can access their accounts.

10. Responsive Design for Mobile

To guarantee user convenience, the web interface is fully responsive and available On desktops, tablets, and smart phones.

11. Input and Support:

A built-in system that allows users to ask questions, report problems, or request assistance, along with ticket tracking for administrators.

CHAPTER 2 REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATION

The goal of the GNDEC Scholarship Management Portal is to make it easier for qualified students to apply for, review, and receive scholarships. This system seeks to ensure transparency in scholarship distribution, cut down on processing time, and do away with manual paperwork.

2.1 Feasibility Study

The GNDEC Scholarship Management Portal's viability and practicality are assessed by the feasibility study. To guarantee successful development and deployment, it takes into account technical, financial, operational,legal, and schedule-related factors.

2.1.1 Technical Feasibility

This checks if the current GNDEC Scholarship Management Portal’s processes, services, and functions can be developed, executed, and maintained with the existing resources and technology.

Current Infrastructure: GNDEC possesses necessary hardware such as desktop computers, servers, and internet speed. The institution manages student data and several of their services digitally, indicating the presence of at least a foundational IT ecosystem.

Technologies for Development:

Frontend: A responsive interface compatible with desktops and mobile devices will be built using the React.js framework which ensures speed, ease of use, mobile responsiveness, and adaptability. HTML, CSS languages are used for frontend.

Backend: Data would be securely stored and retrieved. These can be built using Node.js or Django.

Database: The application’s data alongside the student’s records MySQL, a widely used relational database.

Technological Compatibility: The open source GNDEC technologies integrate well with the other features making them cost effective. These technologies also support modular design which simplifies adding new features and updating or repairing existing ones.

Maintenance Capability: Routine update processes, server management, data backup, and security checks can be done by GNDEC's IT personnel or other developers. Comprehensive documentation, active communities, and add-ons improve troubleshooting, so selected technologies are accessible.

System Expansion Capability: The system will be designed to integrate additional students and scholarships in the future. Other future functionalities like alumni donations and financial monitoring can also be added.

Security Considerations: The system will implement role-based data access control, data validation, and secure file uploads to protect information.

Possibility of Integration: It is also possible to connect the system with email/SMS alert gateways, college ERP systems, and existing GNDEC student databases. This reduces duplication and ensures proper data integration.

As previously specified, the engineering organizational unit possesses the manpower as well as the supportive technology with the needed framework that ensures enduring robustness which makes the technical viability of the GNDEC Scholarship Management Portal feasible.

2.1.2 Economical Feasibility

The portal will make use of free and open-source technologies such as MySQL, Django/Node.js, React.js.The proposed GNDEC Scholarship Management Portal's financial viability and whether the advantages outweigh the expenses of its creation, implementation, and upkeep are assessed by the Economic Feasibility Study.

This assesses the project's cost-effectiveness. Using free, open-source tools and frameworks lowers development costs. Using GNDEC's servers or affordable cloud services like AWS can help keep hosting costs down.Long-term advantages consist of like Less work for the administration. Quicker processing of scholarships .No need to store documents physically.Reducedreliance on paper, which saves money and the environment.The conclusion is that Due to efficiency improvements, the system provides a high return on investment and is economically feasible.

2.1.3 Operational Feasibility

In terms of daily usability and workflow integration, the GNDEC Scholarship Management Portal is very useful. Scholarship applications are currently handled manually, which frequently results in inefficiencies like misplaced documents, processing delays, and a lack of transparency. By providing a centralized platform where students can quickly submit applications, upload documents, and check the status of their submissions, a digital portal will address these problems. This will expedite the entire process and lower administrative overhead.

Secure access to student records, real-time updates, and automated workflows will all benefit staff members. Clerks and administrators will be able to send notifications for missing information, verify applications more quickly, and keep an orderly digital archive for future use thanks to the system.

Users will need little training to use the portal efficiently because of its intuitive interface. Additionally, the system will be available from any device with an internet connection, giving staff and students flexibility. Along with increasing internal efficiency, the portal will make scholarship distribution more equitable and transparent. While administrators can easily monitor and audit the process, students will have clear visibility into application requirements and deadlines. This operational improvement helps GNDEC achieve its objective of becoming a more technologically advanced andstudent-focused institution.

2.2 Software Requirement Specification

1.Overview

The functional and non-functional requirements for creating an online scholarship management portal are outlined in this document. It seeks to digitize and expedite GNDEC's scholarship procedures while guaranteeing openness, precision, and accessibility for both employees and students.

2. Requirements for Function

- Students will be able to use their college credentials to register and log in.
- Based on their eligibility, students will view and apply for available scholarships.
- The portal will enable document management, editing, and uploading.
- Coordinators and faculty will be able to confirm student applications.
- Administrators are responsible for overseeing scholarship categories, reviewing reports, and accepting or rejecting applications.
- The system will use a portal or email to provide real-time notifications and status updates.
- Users will be able to monitor the current state of their applications in real time.

3. Information Needed

A relational database (MySQL) will house the data. Each student's documents will be stored securely and mapped to their profile. To prevent redundancy, all data must adhere to normalization rules.

- Student Profile: Name, Category, Branch, Roll No., Course, and Contact Details
- Scholarship Information: Name, provider, eligibility, deadline, and description
- Document Uploads: Income certificates, identification documents, grade reports, and other documents in PDF and JPEG formats.
- Application Records: ID, date of submission, status, and comments
- Admin Logs: Time-stamped actions taken by the administrator and verifiers

4. Needs for Performance

- System should support at least 500 concurrent users.
- Under typical load, the application should load in no more than two seconds.
- User requests must be answered in less than three seconds.
- File upload size limit should be 5MB per document with proper validation.

5. Requirements for Dependability

- The system should have a minimum uptime of 99% and be accessible around-the-clock.
- Data and document backups must be performed automatically every day.
- The portal ought to work consistently in all contemporary web browsers.
- The system should respond appropriately to user feedback and gracefully handle unforeseen errors.

6. Sustainability Conditions

- The system must be modular in order to facilitate feature additions and updates.
- Standard documentation and commenting procedures must be followed by the code.
- Scholarship programs should be able to be enabled and disabled dynamically through the admin interface.
- Logs ought to be kept for auditing and debugging purposes.

7. Needs for Security

- To ensure secure data transmission, all communications must be conducted over HTTPS.
- Passwords must be stored using hashing algorithms, such as bcrypt/argon2.
- To prevent unwanted access, role-based access control needs to be implemented.
- To stop SQL injection and XSS attacks, the system must use input validation and sanitization.
- For traceability, faculty and administrative actions should be recorded with timestamps.
- For faculty and administrator accounts, 2-Factor Authentication (2FA) is optional.

8. Needs to See and Feel

- The portal's user interface should be clear, contemporary, and easy to use. It should also be responsive and mobile-friendly.
- Employ a color palette that is consistent with GNDEC's branding.
- User actions should be guided by visual indicators, tooltips, and icons.
- Following user actions, clear error messages and success indicators should be displayed.

9. Dependencies and Assumptions

- Users use contemporary browsers and have access to the internet.
- Institutional email addresses are accessible for enrollment.
- GNDEC will supply or host the server infrastructure that is required.
- The designated IT staff will be in charge of system maintenance.

2.3 SDLC model to be used

Depending on your development timeline and flexibility requirements, the Iterative Waterfall Model or the Incremental Model would be the best Software Development Life Cycle (SDLC) model for the Scholarship Management Web Portal project at Guru Nanak Dev Engineering College (GNDEC).

2.3.1 Iterative Waterfall Model for the SDLC:

This model, which permits feedback and adjustments at every stage, is an improved version of the traditional waterfall model. It works well for academic projects based at institutions where:

- At first, requirements are understood quite clearly.
- Every stage must be formally reviewed and documented.
- Development is approached in a methodical and organized manner.

- Throughout the process, some adjustments or iterations might be required.

Iterative Waterfall Model Requirement Analysis Phases:

1. Requirement Analysis

   Compile and complete the system's requirements, including user roles, functionality, documents, and workflows. The Software Requirement Specification (SRS) document is the output.
2. Design of the System

   Make database schemas, UI mockups, and architectural designs.

   Output: Flowcharts, ER diagrams, and the design document.
3. Execution

   Create the admin panel, document manager, and student module in phases.

   Output: Each module's functional code.
4. Examining

   User Acceptance Testing, Integration Testing, and Unit Testing.

   Prior to deployment, bugs are found and fixed.
5. Implementation

   Putting the web portal up for use or demonstration on a server or local system. Upkeep after-deployment updates or feature additions.

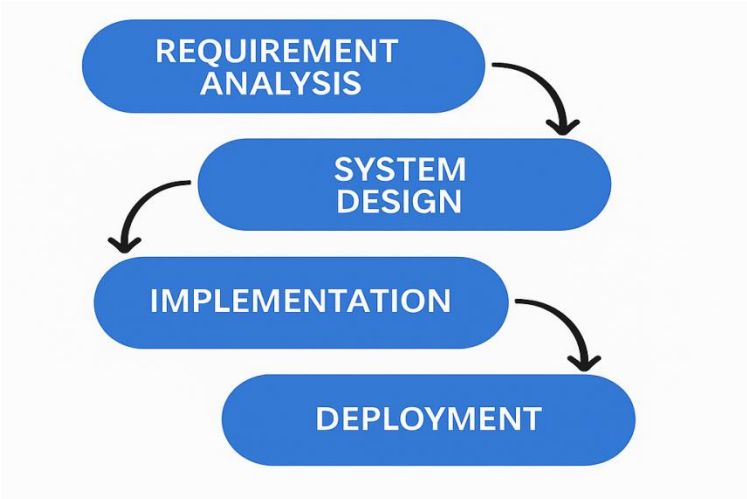


Figure 1: Iterative Waterfall Model

2.3.2 Incremental Model:

With the Incremental Model, software is developed and delivered in manageable chunks known as increments. A functional component is added to the system with each increment, which also goes through the usual stages of analysis, design, implementation, and testing. Over time, the entire system integrates several increments to evolve.

The Incremental Model's Phases:

1. Requirement Analysis

   Core features and optional upgrades make up the requirements.There is a mini-requirement analysis phase for every increment.

2. Design

   Create Future increments are initially supported by the system architecture.

   Before development, every increment is given a thorough design.

3. Execution

   Small iterations are used to carry out development. Every increment is coded separately and combined with the one before it.

4. Examining

   Each increment is examined for integration and functionality.

   Regression testing makes sure that after integration, earlier iterations continue to function.

5. Implementation

   Following testing and approval, each increment can be released separately.

   Even before it is finished, the system is usabl

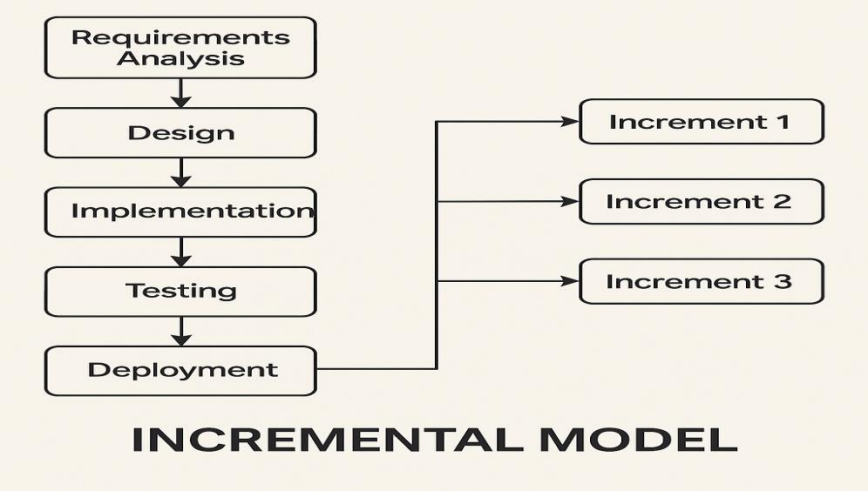


Figure 2 Incremental Model

Chapter 3 System Design

The system design serves as the blueprint for the scholarship portal's construction. In order to guarantee that the application is effective, scalable, and user-friendly, it comprises architectural, data, module, and interface design.

3.1 Design Approach (Function oriented or Object oriented)

3.1.1 Object Oriented

Modeling the system as a group of interdependent object, which are instances of classes, is the main goal of the Object-Oriented Design (OOD) methodology. Every object encapsulates both data (attributes) and behaviors (methods), representing a real-world entity. This design approach encourages greater modularity, code reuse, and organization, which makes it ideal for intricate systems like the Scholarship Management Web Portal.

.

Object-Oriented Design: Why Use It?

The system is complex, modular, and involves multiple user roles, each interacting with shared data in different ways. Object-Oriented Design allows for:

- Encapsulation: Every element can be thought of as an independent entity with characteristics and actions.
- Reusability: Modules can share common features (such as document upload and authentication).
- Scalability: It's simpler to add new functions or features without interfering with already written code.
- Maintainability: The modular class structure makes debugging and extension simpler.
- Real-world Mapping: The system is easy to model and design because it closely corresponds to real-world entities such as students, applications, and scholarships.

3.1.2 Function oriented

The entire system is divided into a number of distinct functions or processes, each carrying out a particular task, according to the function-oriented design approach. With this approach, the system is created from a process perspective, emphasizing the functions that the system must fulfill rather than its data-driven architecture. Inputs are transformed into outputs throughout the portal, which is managed by a number of interrelated functions.

User authentication, form submission, document validation, scholarship filtering, and admin verification are just a few of the functional modules that make up the Scholarship Management Web Portal. For instance, the scholarship application function comprises input validation, document upload, eligibility check, and submission. Each module is further subdivided into sub-functions. The development team can test individual modules independently and implement features in a sequential manner thanks to this organized breakdown. Data Flow Diagrams (DFDs), which show how data moves between functions and external entities like users or databases, are also supported by the function-oriented model.

Even though function-oriented design offers a more straightforward and methodical development process, particularly for small-scale and academic projects, it may not be as scalable or reusable as object-oriented approaches. Nonetheless, it works very well for projects with
clearly defined operations, such as this portal, which allows for the individual design and testing of particular processes (such as "Apply for Scholarship" or "Verify Document").

3.2 Detail Design

System design is the process or art of defining the architecture, components, modules, interfaces and data for a system to satisfied requirements. One could see it as the application of system theory to product development. There is some overlap and synergy with the disciplines of systems analysis, systems architecture and systems engineering.

INPUT DESIGN

The input design is the process of converting the user-oriented inputs in to the computer-based format. The goal of designing input data is to make the automation as easy and free from errors as possible. For providing a good input design for the application easy data input and selection features are adopted. The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right message and help for the user at right time are also considered for the development of the project.

The following points should consider while designing the input:

- What data to input?
- What medium to use?
- How the data should be arranged or coded?

- The dialogue to guide users in providing input
- Data items and transactions needing validation to detect errors
- Methods for performing input validation and steps to follow when errors occur.

OUTPUT DESIGN

When designing output, systems analyst must accomplish the following:

- Determine what information to present
- Decide whether to display, print the information and select the output medium
- Arrange the presentation of information in an acceptable format
- Decide how to distribute the output to intended recipient

Accomplishing the general activities listed above will require specific decisions, such as whether to use preprinted forms when preparing reports and documents, how many line to plan on printed page, or whether to user graphics and color. The output design is specified on layout forms, sheets that describe the location characteristics and format of the column heading and pagination. As we indicated at the beginning of this discussion, these elements are analogous to an architect blueprint that shows the location of each component.

3.2.1 SOFTWARE ARCHITECTURE

The software architecture defines how the system components will interact and ensures the portal is scalable, secure, maintainable, and user-friendly**.**

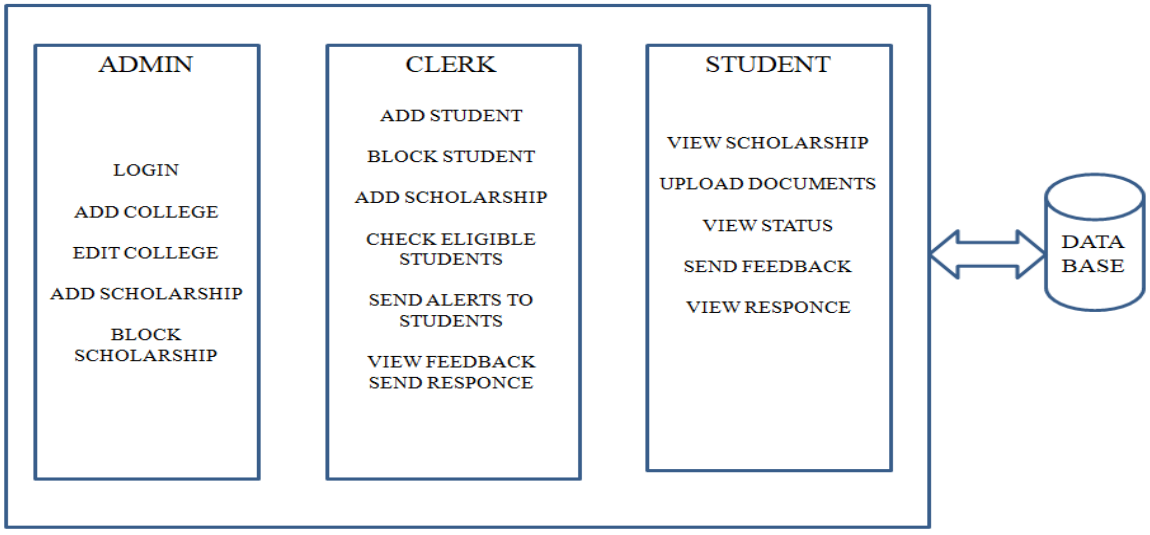


Figure 3 Software Architecture

3.2.2 FLOW CHART OR BLOCK DIAGRAM

The System Flow Chart for the Scholarship Management Web Portal for Guru Nanak Dev Engineering College (GNDEC) illustrates the step-by-step flow of processes within the system. It provides a visual representation of how various components interact with one another and how data moves through the system. This chart is crucial for understanding the system architecture and the sequence of operations, ensuring clarity in both development and operation.

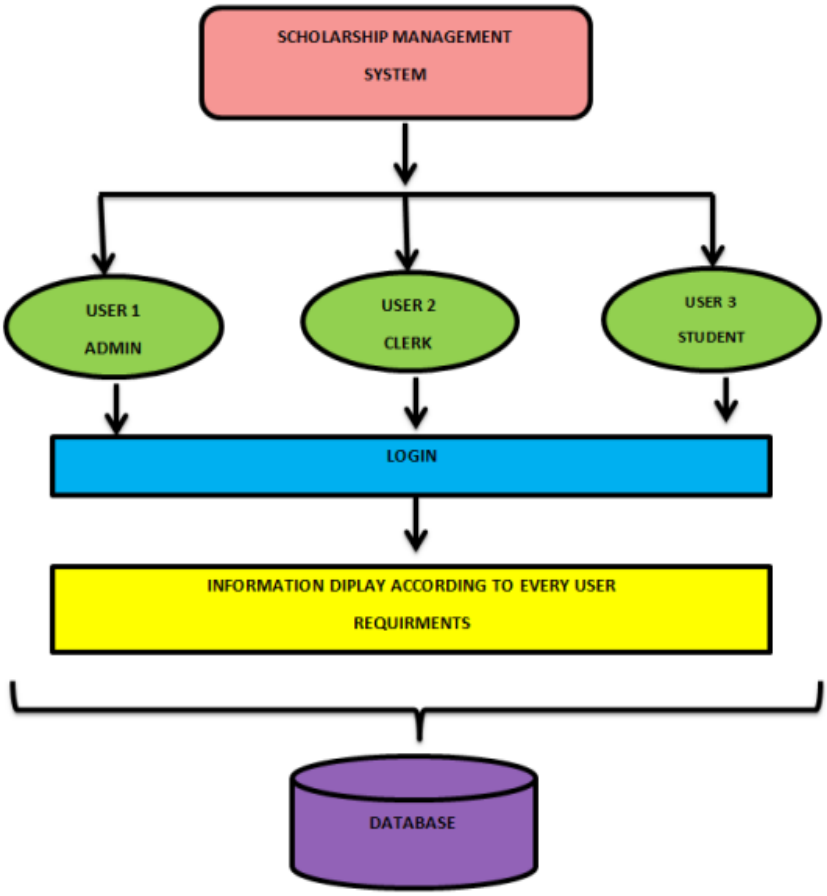


Figure 4 Flow chart of System

It starts when a student enters their login information into the portal. After being verified, the student can look through the various scholarships that are offered, each of which has specific requirements and due dates. The student completes the scholarship application form, attaches the necessary paperwork (such as income certificates and grade reports), and submits it if they are qualified and interested.

The application is sent to the admin panel, where committee members or authorized administrators can examine, accept, or reject it after the system verifies the inputs. The application might, if required, go through an approving committee or a multi-level review process involving several departments. The student is notified at various points during the process.

1) **ADMIN:** The Admin Flowchart is following:

Figure 5 Flow chart of admin

1. LOGIN: Firstly admin can login on this scholarship web portal.

2. VALIDATE: In this section, login information is verified if it is true then admin will go on home page otherwise go to login section again and fill correct login information.

3. ADMIN HOME PAGE: After successfully login on the portal then admin can reach on Home page and check all needed details. Admin can be edit all details, manage notifications and view feedback.

2) **STUDENT:** The Student Flowchart is following:

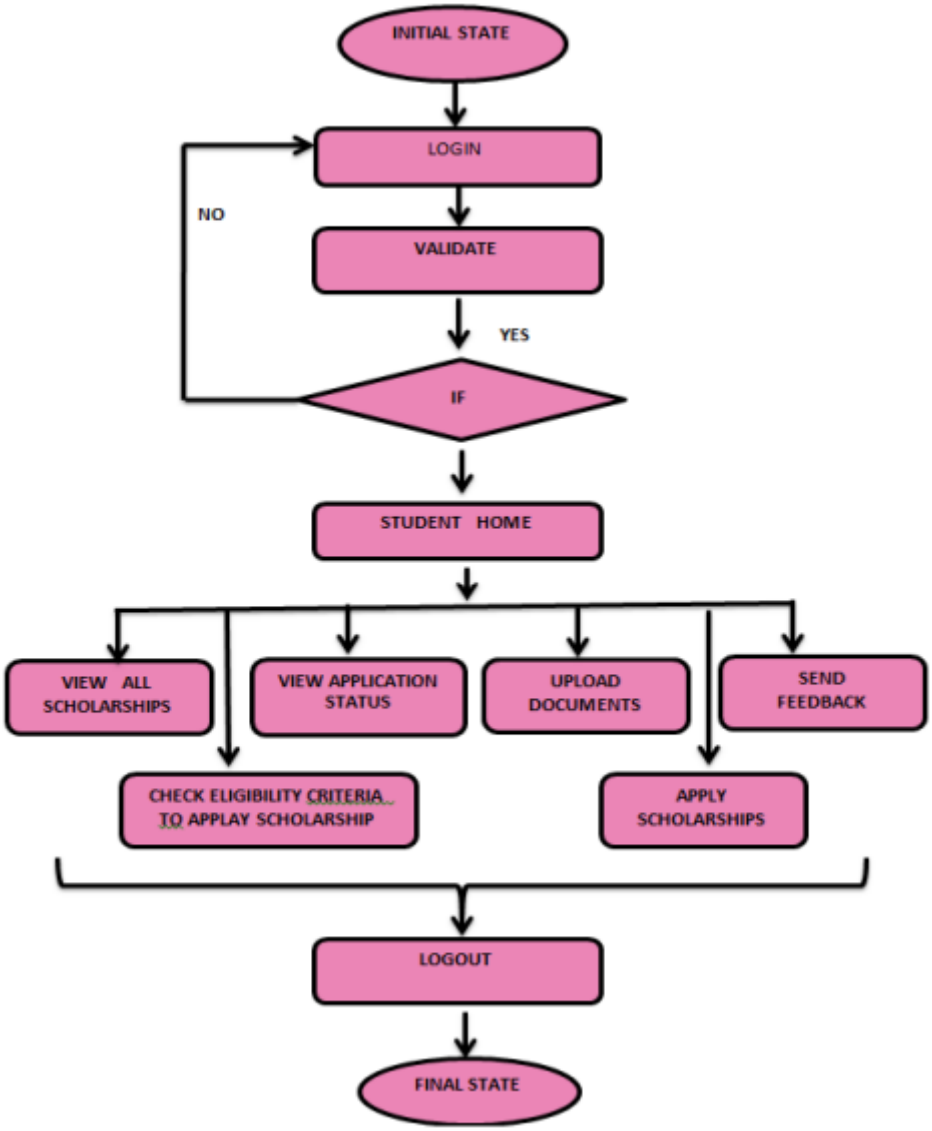


Figure 6 Flow Chart of Student

1. LOGIN AND VALIDATE: Firstly Student login on this scholarship web portal.

2. STUDENT HOME PAGE: After successfully login on the portal then student can reach on Home page and check all needed details. Student can be view all scholarships, view appl. Status, apply scholarship and send feedback etc.

3) CLERK: The Clerk Flowchart is following:

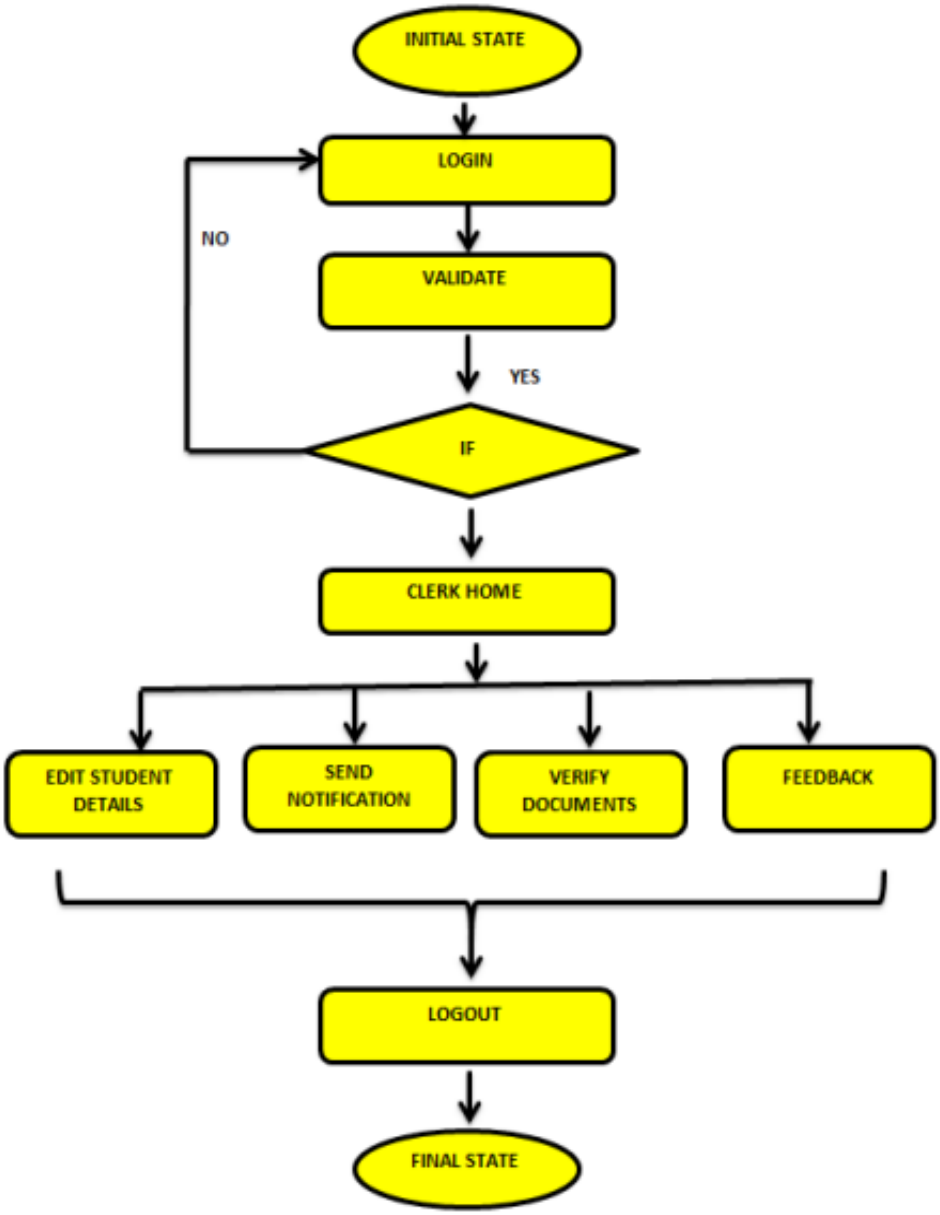


Figure 7 Flow Chart of Clerk

1. LOGIN AND VALIDATE: Firstly Clerk login on this scholarship web portal.

2. CLERK HOME PAGE: After successfully login on the portal then clerk can reach on Home page and check all needed details. Clerk can be edit student details, send notifications, verify documents and feedback etc.

3.2.3 UML DIAGRAMS

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing and documenting the artifacts of software systems, as well as for business modeling and other non- software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs and validate the architectural design of the software.

- Use Case Diagram: The use case diagram is used to identify the primary elements and processes that form the system. The primary elements are termed as "actors" and the processes are called "Use cases". The use case diagram shows which actors interact with each use case.

  ADMIN: Use case diagram for admin is given below:

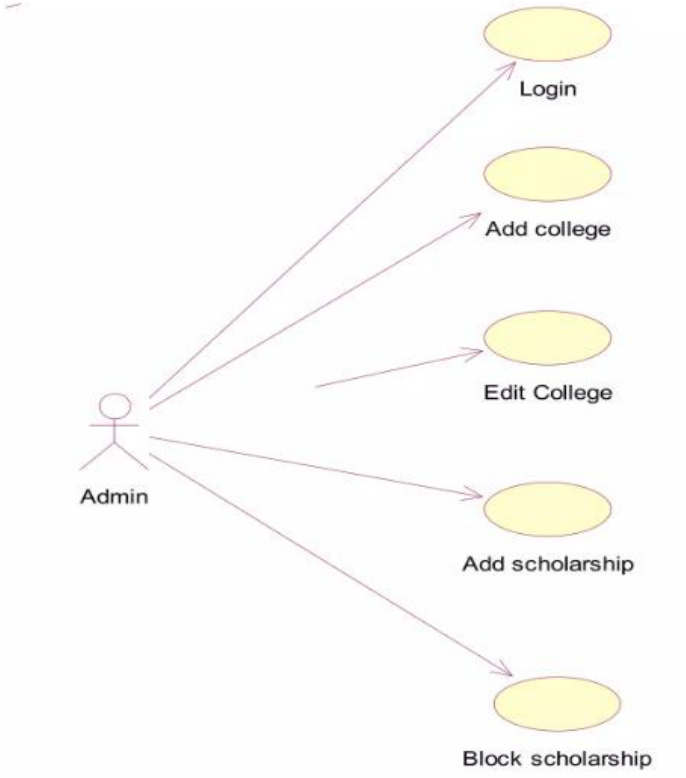


Figure 8 Use case diagram of admin

CLERK: Use case diagram for clerk is given below:

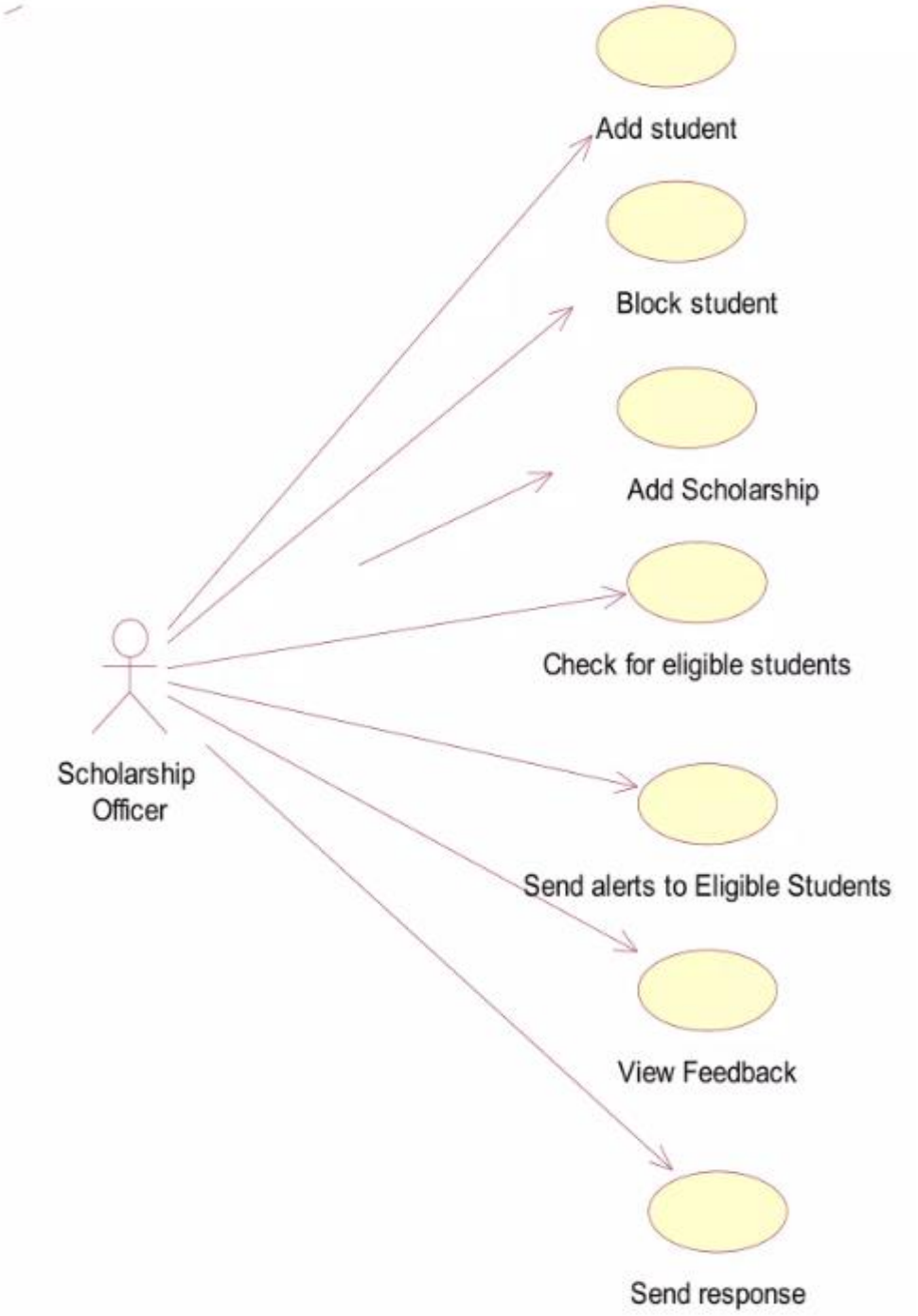


Figure 9 Use case diagram of clerk

STUDENTS: Use case diagram for admin is given below:

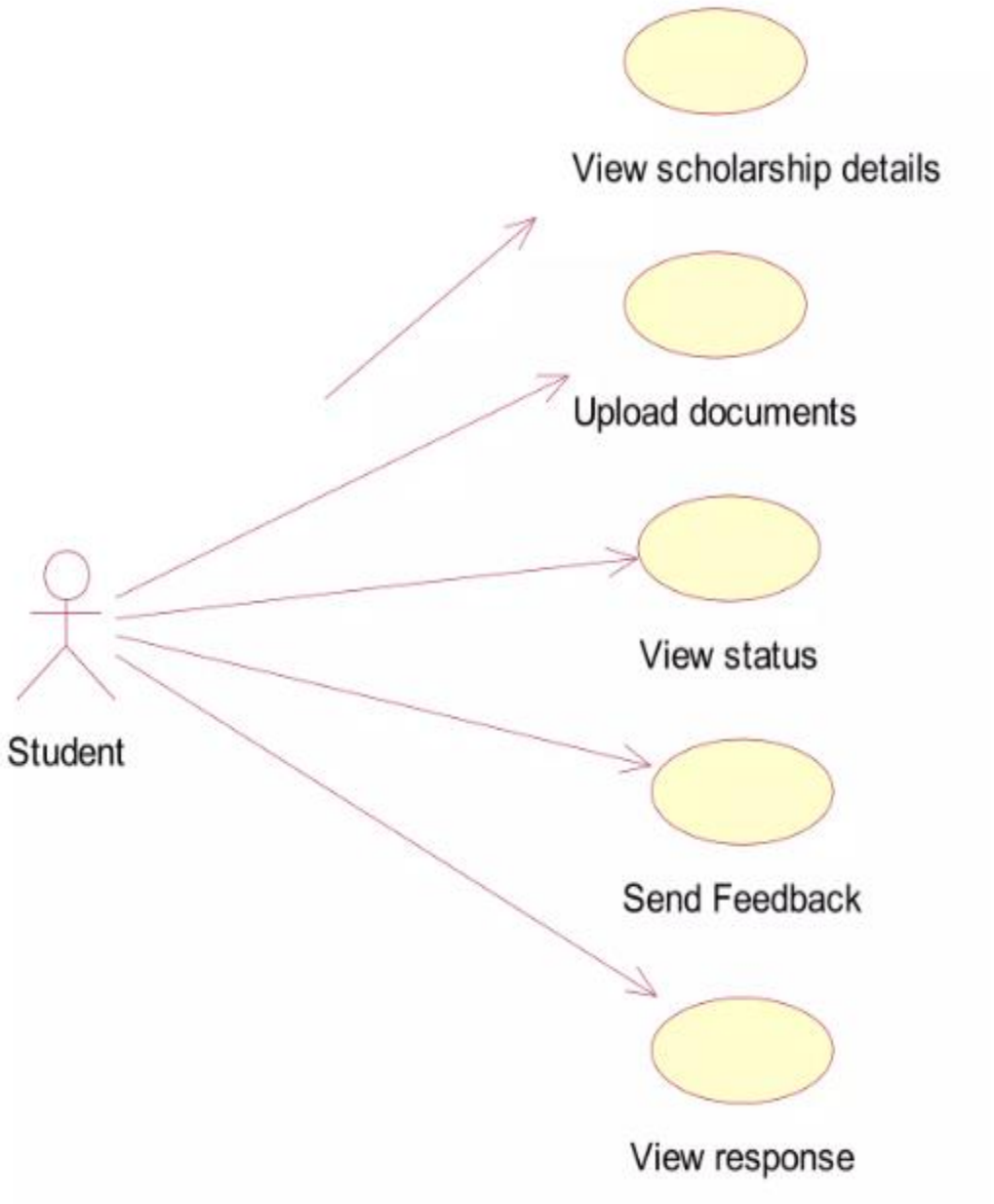


Figure 10 Use case diagram of Student

- State Diagrams: A state diagram, as the name suggests, represents the different states that objects in the system undergo during their life cycle. Objects in the system change states in response to events. In addition to this, a state diagram also captures the transition of the objects state from an initial state to a final state in response to events affecting the system.

ADMIN:

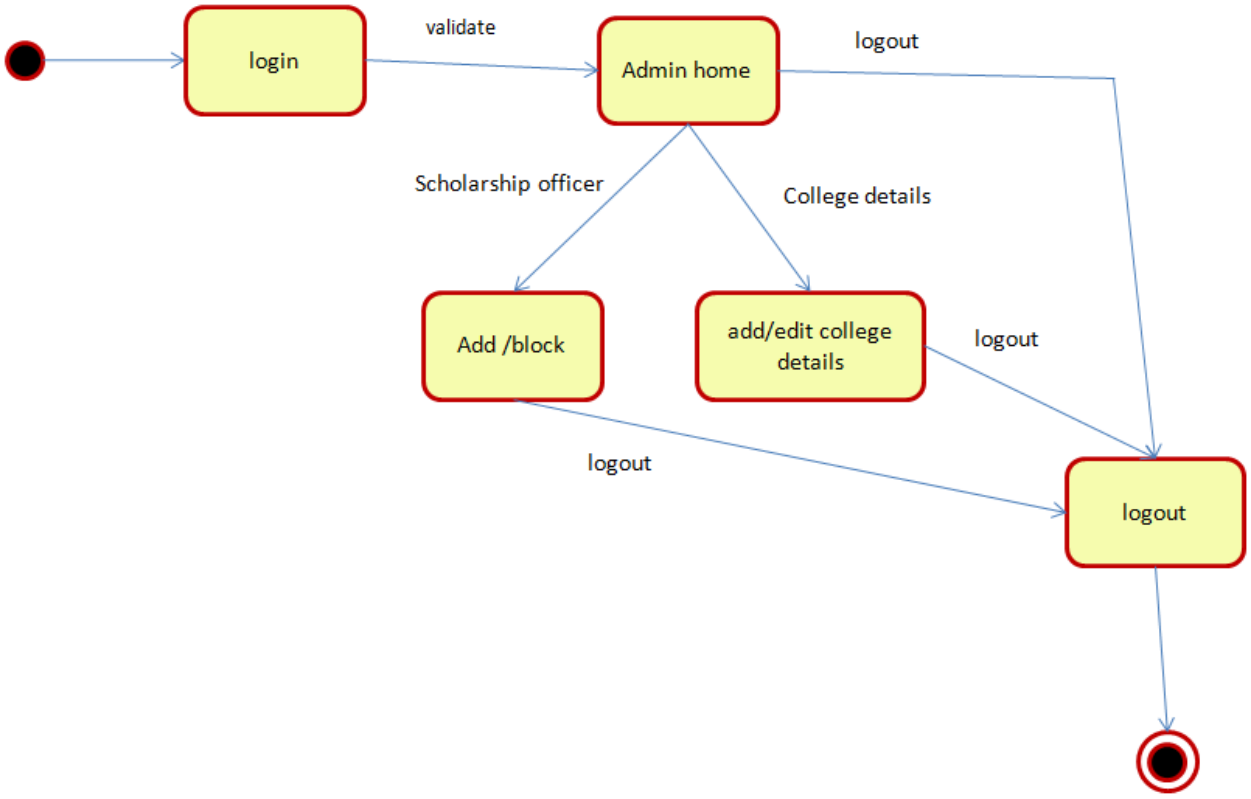


Figure 11 State diagram of admin

CLERK:

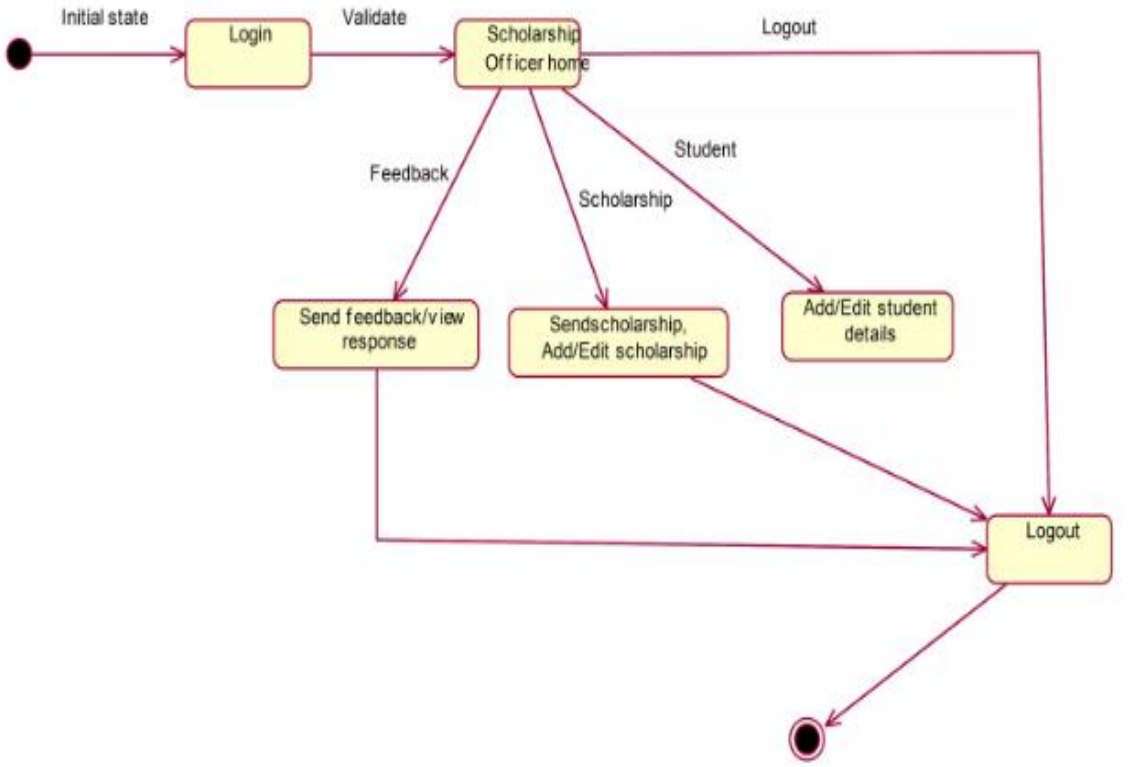


Figure 12 State Diagram of clerk

STUDENTS:

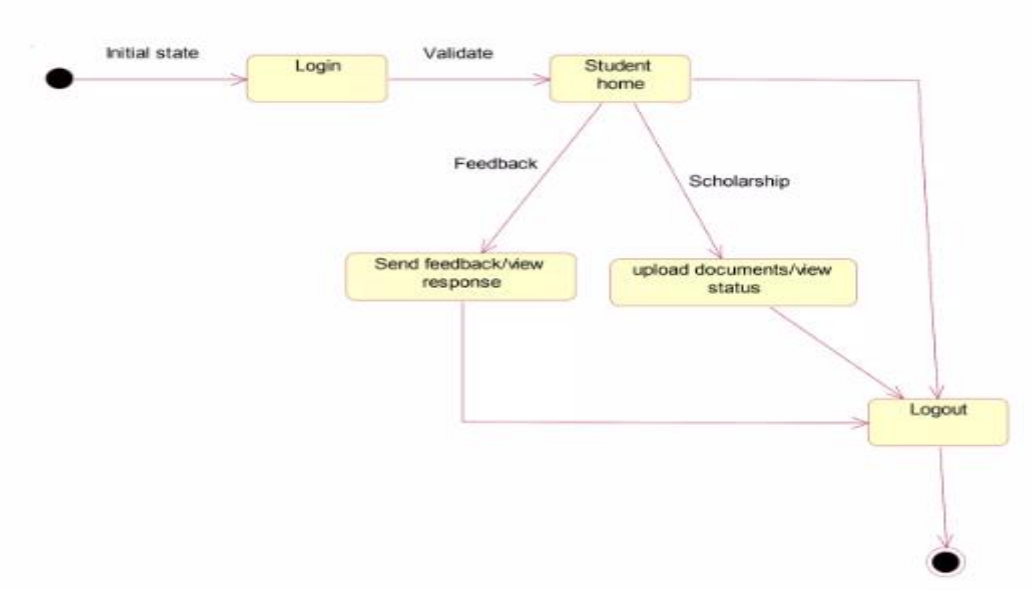


Figure 13 Sate Diagram of students

- Activity Diagram: The process flows in the system are captured in the activity diagram, Similar to a state diagram, an activity diagram also consists of activities, actions, transistions, initial and final states and guard conditions.

ADMIN:

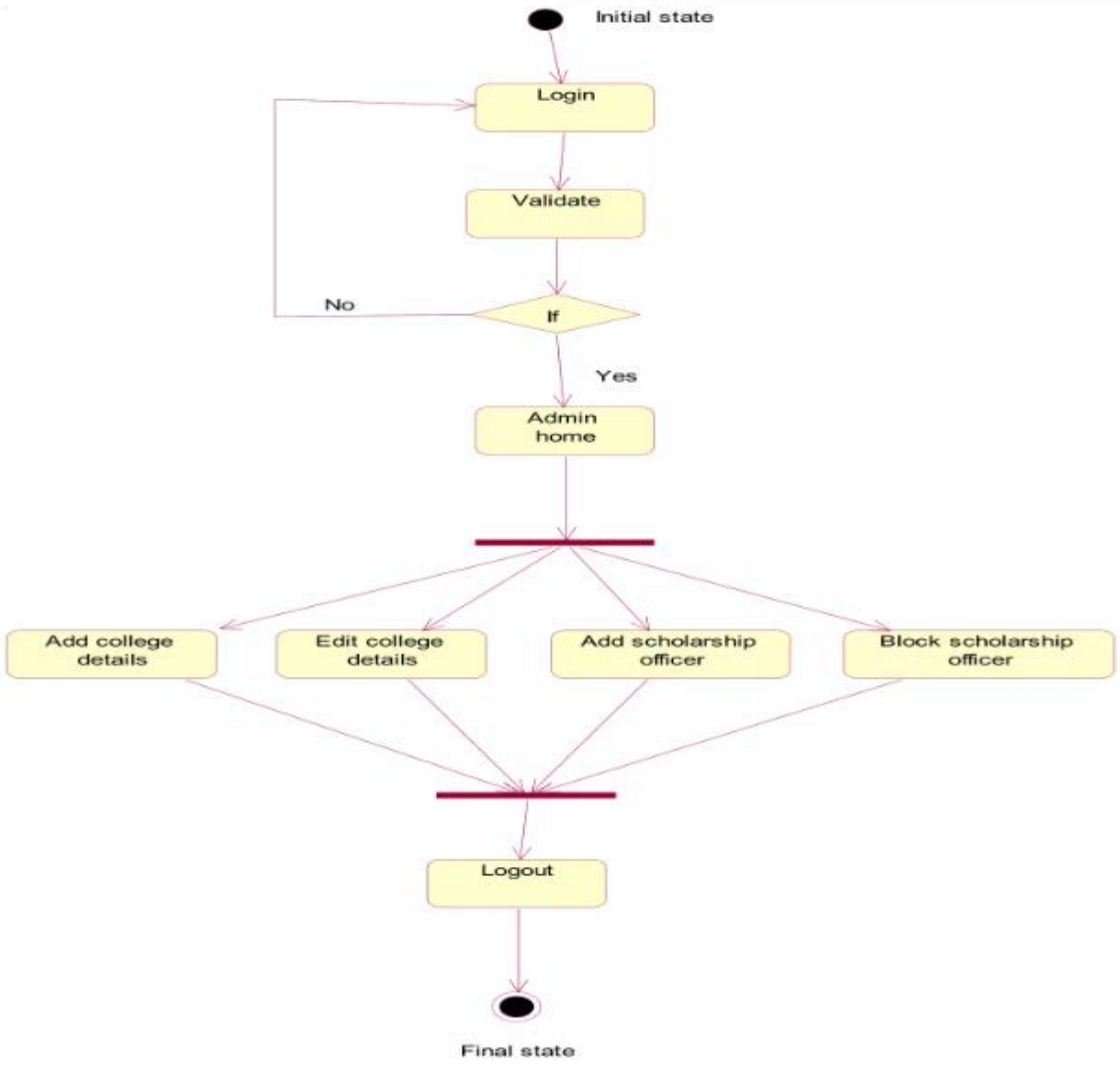


Figure 14 Activity diagram of admin

CLERK:

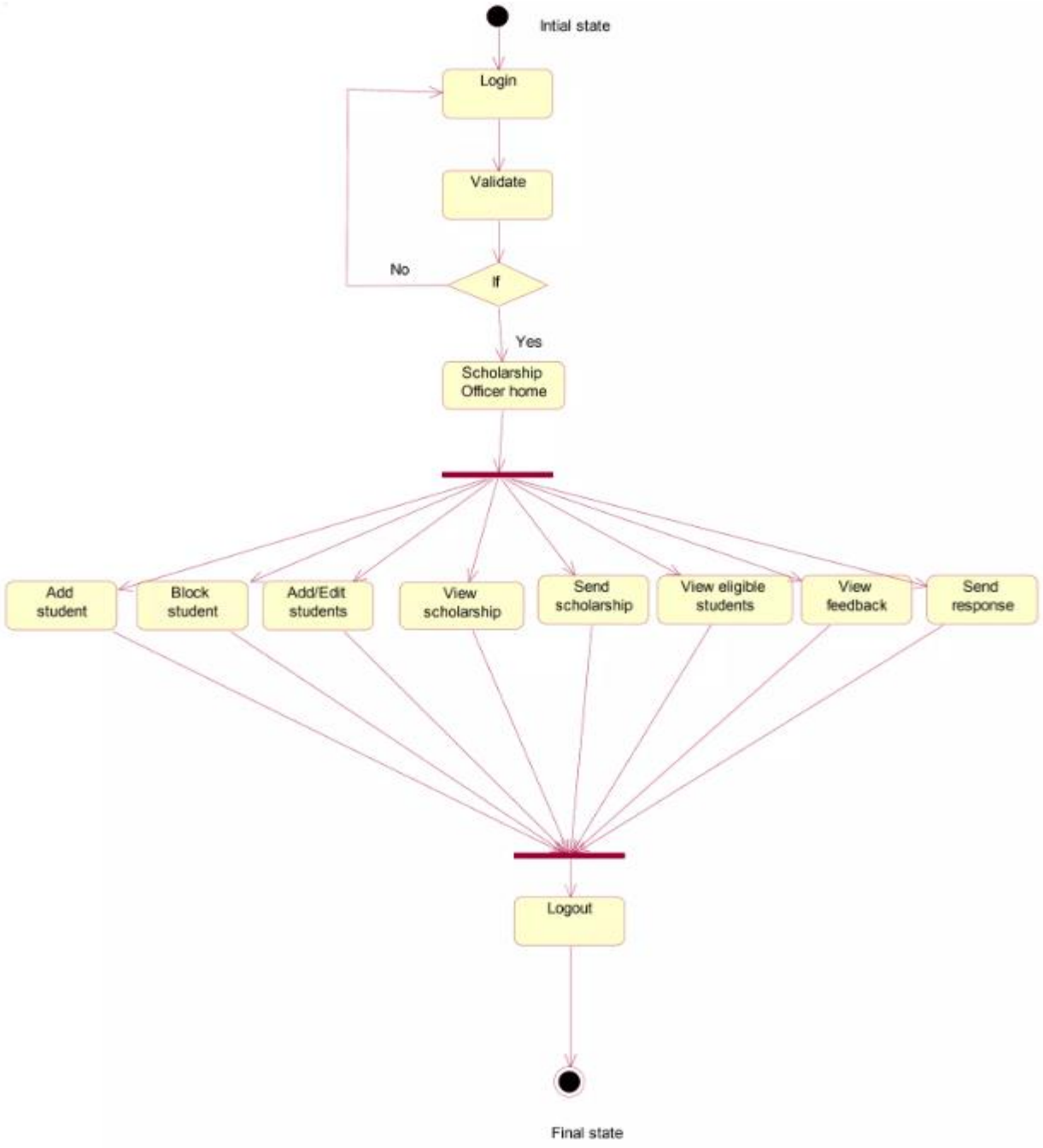


Figure 15 Activity diagram of Clerk

STUDENT:

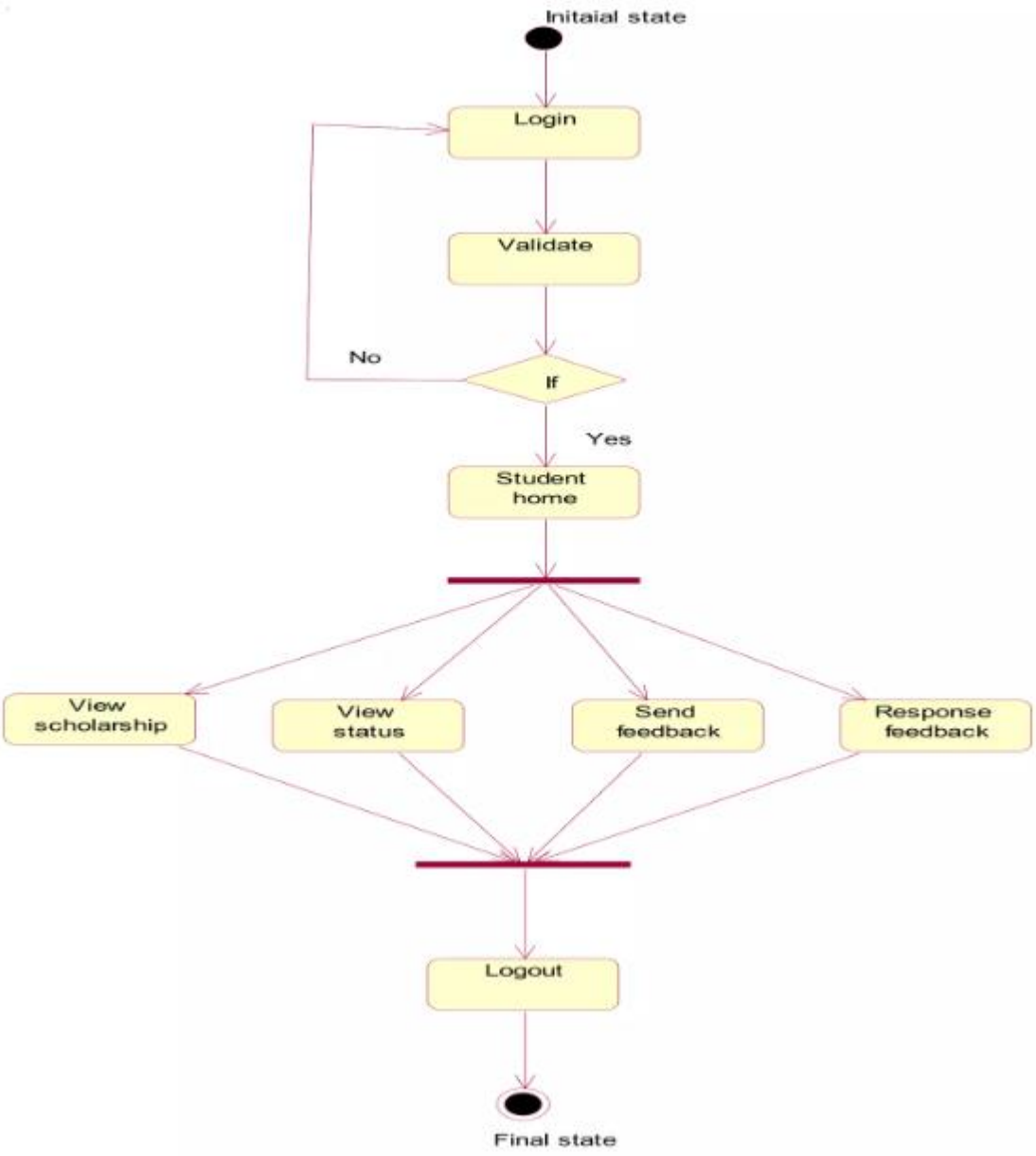


Figure 16 Activity diagram of Student

- Sequence Diagram: A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered. This means that the exact sequence of the interactions between the objects is represented step by step. Different objects in the sequence diagram interact with each other by passing "messages".

ADMIN:

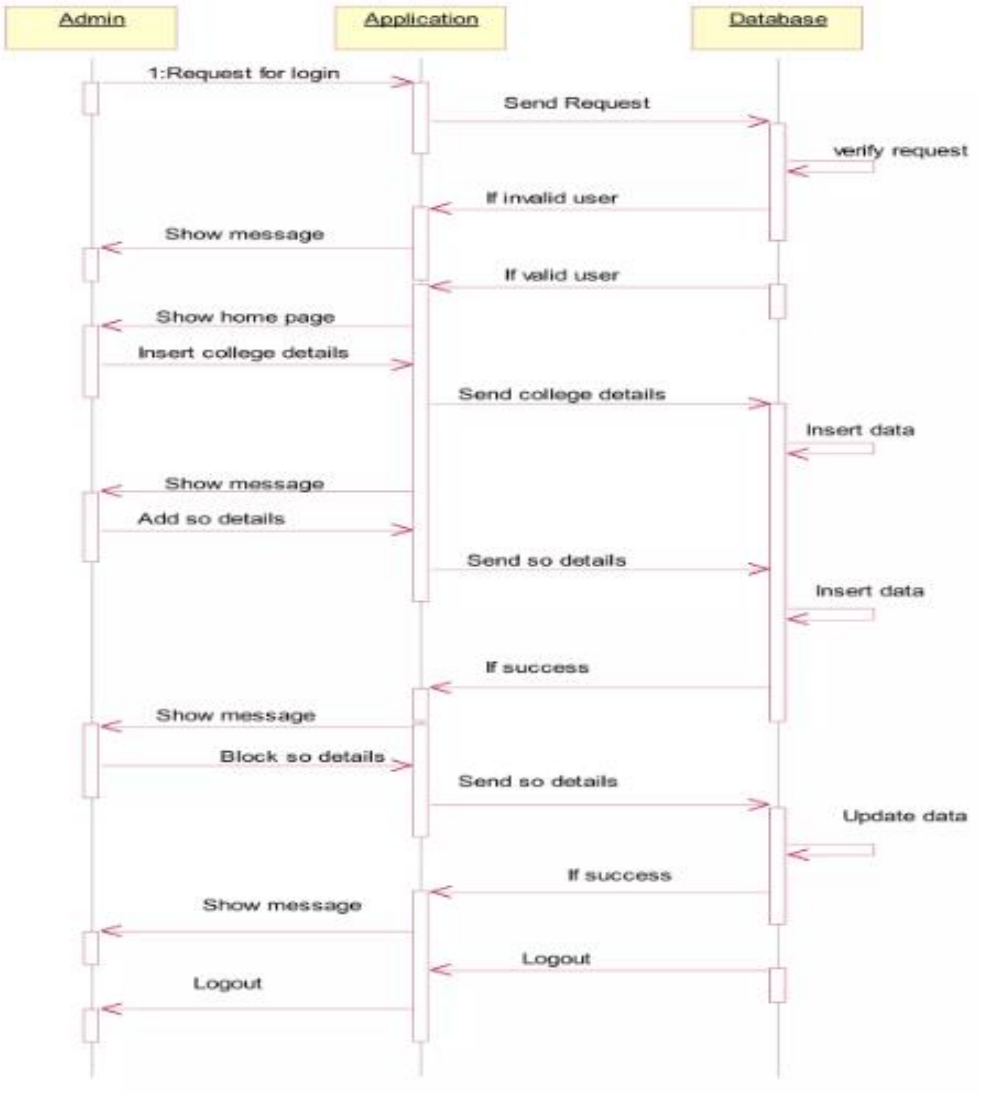


Figure 17 Sequence diagram of Admin

CLERK:

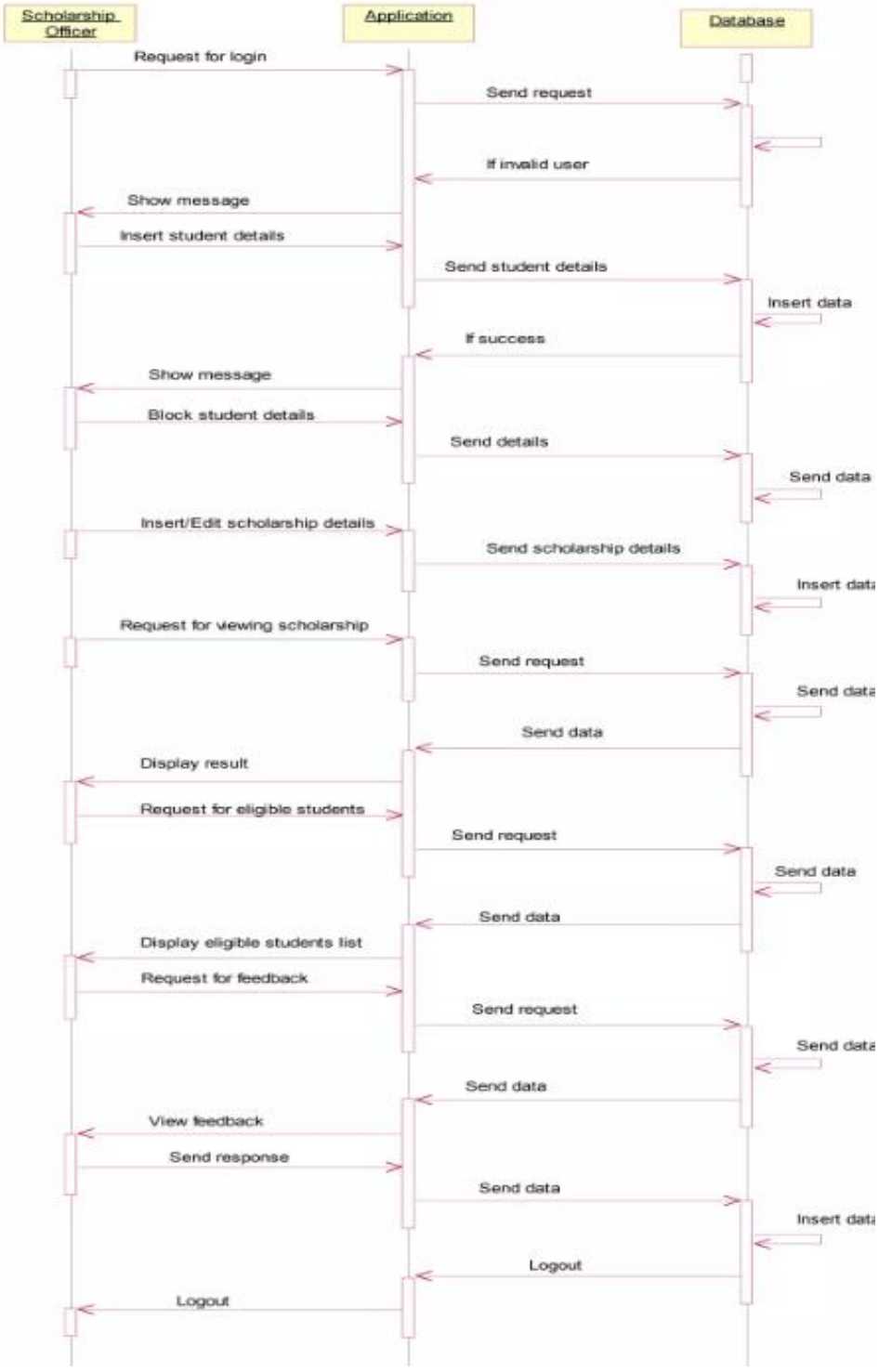


Figure 18 Sequence diagram of Clerk

STUDENT:

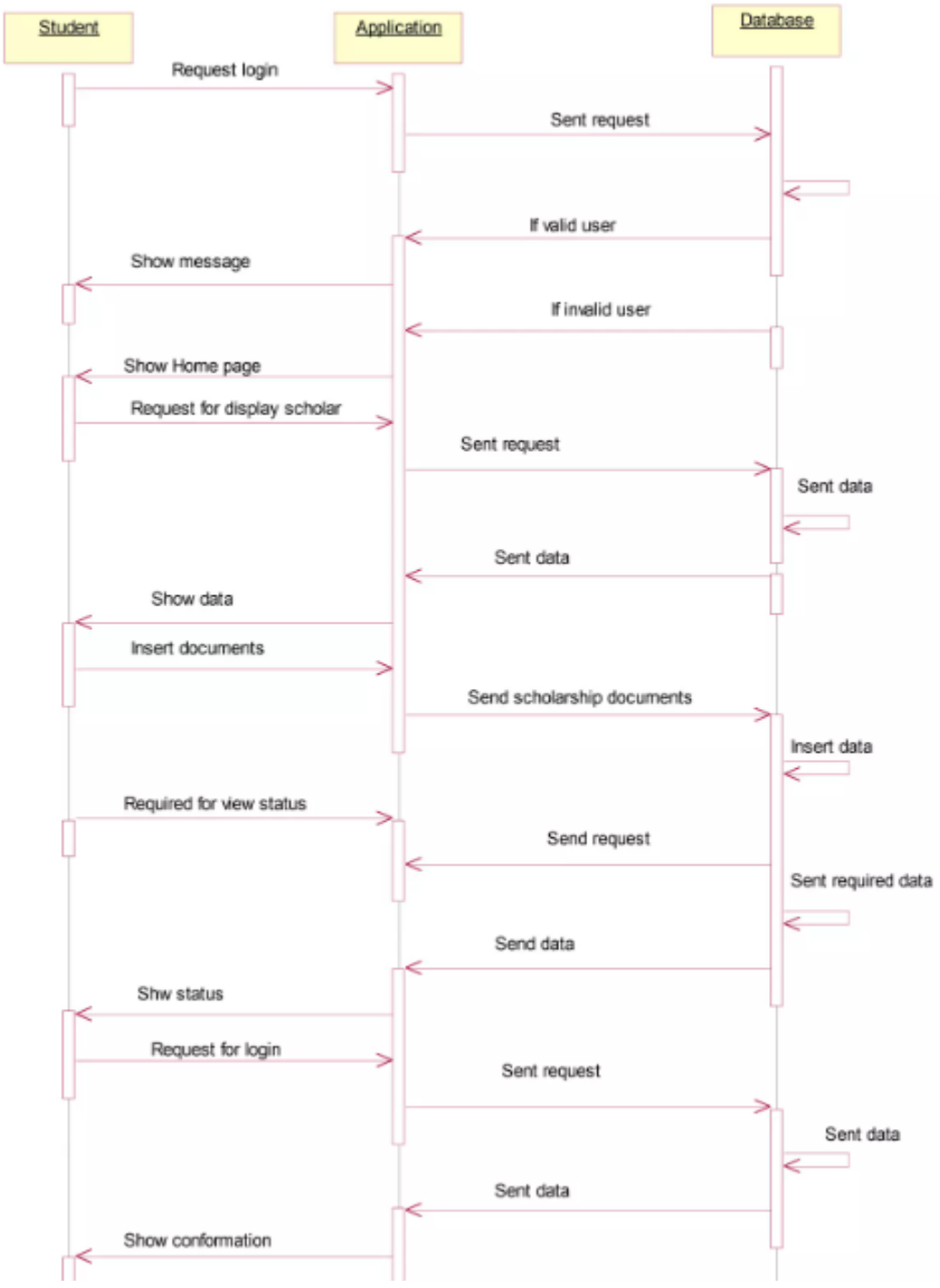


Figure 19 Sequence diagram of Student

- Deployment Diagram: The deployment diagram captures the configuration of the runtime elements of the application. This diagram is by far most useful when a system is built and ready to be deployed.

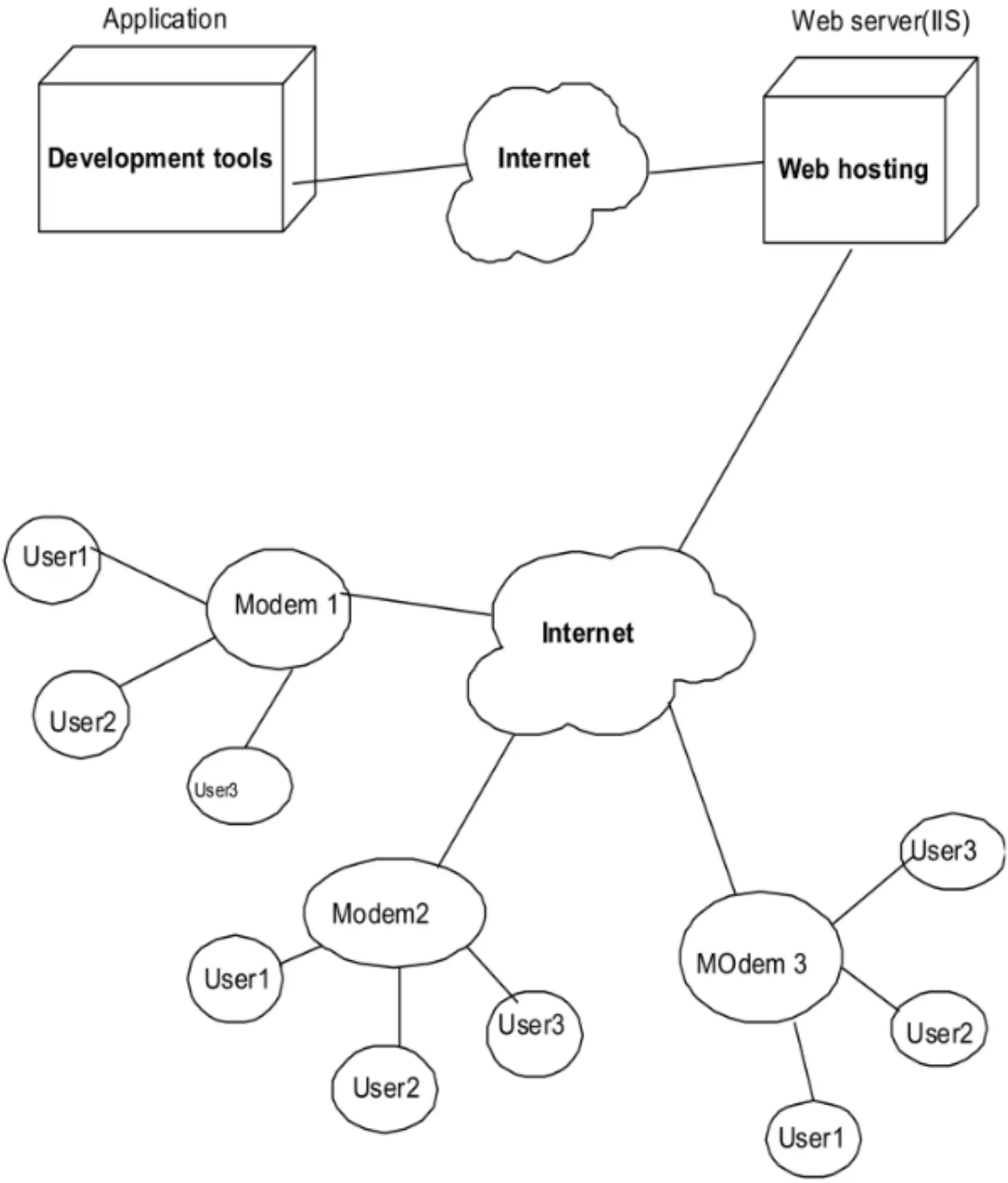


Figure 20 Deployment Diagram

- ER Diagram: An ER model is an abstract way to describe a database. Describing a database usually starts with a relational database, which stores data in tables. Some of the data in these tables point to data in other tables – for instance, your entry in the database could point to several entries for each of the phone numbers that are yours. The ER model would say that you are an entity, and each phone number is an entity and relationship between you and the phone number is 'has a phone number'. Diagram created to design these entities and relationships are called entity-relationship diagrams or ER diagrams.

  Entity Relationships are three kinds:

  1. One-One
  2. One-Many
  3. Mnay-Many

  One-to-one: One instance of an entity (A) is associated with one other instance of another entity (B). For examples, in a database of employees, each employee name (A) is associated with only one social security number (B).

  One-to-Many: One instance of an entity is associated with zero, one or many instances of another entity, but for one instance of entity B there is only one instance of entity A.

  Many-to-Mnay: One instance of an entity is associated with one, zero or many instances of another entity and one instance of entity B is associated with one, zero or many instances of entity A.

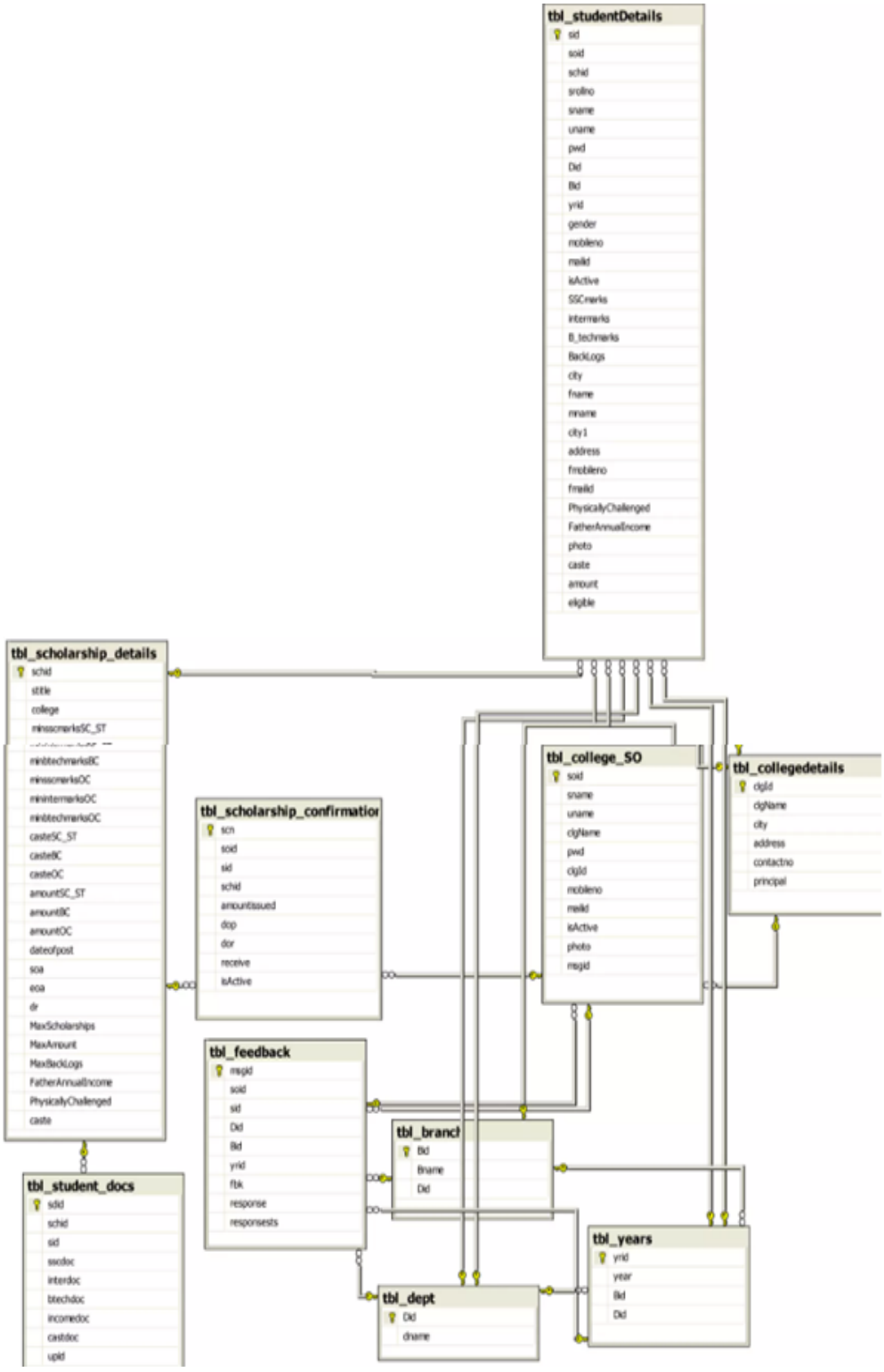


Figure 21 ER Diagram

3.3 USER INTERFACE DESIGN

The GNDEC Scholarship Management System's user interface (UI) is made to be simple, effective, and intuitive. Its main objective is to make the scholarship application and administration process easier for academic staff, administrative personnel, and students. Users can access pertinent features with little effort thanks to a simple layout, responsive design, and simple navigation.

1. Design of the Dashboard

   After logging in, the main dashboard acts as a central location for users. It shows students the status of their applications, scholarships that are available, deadlines, and notifications. It gives administrators the ability to review applications, update records, and create reports. Icons, progress bars, and color codes are examples of visual components that improve usability and speed of comprehension.

2. Accessibility and Navigation

   With menu items like "Apply for Scholarship," "Application Status," "Documents Upload," and "Help" prominently labeled, the interface features a standardized navigation bar. In order to guarantee usability on all devices and for all users, accessibility is given top priority, with support for screen readers, keyboard navigation, and mobile responsiveness.

3. Design and Validation of Forms

   The scholarship application form is divided into sections for financial background, academic records, and personal information. Tooltips and real-time input validation help users fill out forms accurately, reducing mistakes and enhancing the caliber of submissions.

4. Permissions and Roles for Users

   Administrators, reviewers, and students all have different interfaces and permissions. By guaranteeing that each user only engages with pertinent functions, role-based access lowers complexity and boosts security.

5. Comments and Alerts

   The system uses pop-ups, alerts, and confirmation messages to give users real-time feedback. Throughout the process, users are kept informed and involved by receiving email and in-app notifications regarding submission statuses, approval decisions, and impending deadlines.

3.4 METHODOLOGY

1. Analysis of Requirements

A comprehensive requirement-gathering phase preceded the project's start. Through surveys and interviews, input was gathered from important stakeholders, such as academic staff, administrative personnel, and students. This made it easier to pinpoint essential features like listing scholarships, submitting applications, uploading documents, and monitoring status.

2. Development of User Personas

To represent the various user types—students, reviewers, and administrators—user personas were developed based on the requirements that were gathered. To make sure the system satisfies the unique requirements and expectations of every group, these personas served as a guide during the design process.

3. Prototyping and Wireframing

The user interface's structure was delineated using low-fidelity wireframes. Using programs like Adobe XD, these were improved into high-fidelity prototypes. Prior to development, prototypes were tested with a small group of users to gather usability feedback.

4. Principles of Interface Design

The UI was created with accessibility, consistency, and simplicity in mind. To guarantee usability on all devices, a simple layout, responsive design, and easy navigation were prioritized. For visual coherence, UI elements adhered to a standard design system.

5. Creation and Execution

HTML, CSS, and JavaScript frameworks like React or Angular were used to create the front-end interface. Form submissions, status updates, and data retrieval all worked flawlessly thanks to backend integration. Access control based on roles was put in place to ensure.

6.Testing and Evaluation

The system was put through several rounds of testing, including accessibility checks, usability tests, and unit tests. Feedback from real users was used to improve the interface, and bugs and performance issues were fixed to guarantee dependability.

7.Deployment and Maintenance

Following successful testing, the interface was put live on the GNDEC web server. Ongoing monitoring and recurring updates are planned to maintain system performance, correct bugs, and take into account user feedback in the future.

Chapter 4 Implementation and Testing

4.1 Introduction to Languages, IDEs, Tools and Technologies Used for Project Work

The GNDEC Scholarship Management System was developed using a variety of tools and technologies to ensure efficiency, scalability, and user-friendliness. In the development of any software project, the selection of appropriate programming languages, Integrated Development Environments (IDEs), tools, and supporting technologies plays a crucial role in ensuring efficiency, maintainability, and scalability.

1. **Programming Languages**

   - **Frontend:**
     The user interface of the system was developed using HTML**,** CSS**,** and JavaScript, enabling the creation of interactive and user-friendly web pages.
     In some cases, Bootstrap or frameworks like React.js may also be used to enhance the responsiveness and aesthetics of the UI.
   - **Backend:**
     PHP or Python was employed to handle server-side logic, database interactions, and business processes. These languages are widely used due to their simplicity, vast community support, and strong framework ecosystem.
   - **Database Query Language:**
     SQL (Structured Query Language) was used to communicate with the database system, enabling efficient data manipulation and retrieval.

2. **Integrated Development Environments (IDEs)**

   - **Visual Studio Code:**

     VS Code served as the primary IDE due to its lightweight nature, powerful extensions, and support for multiple languages.

   - **Net Beans:**

Depending on the backend language, Net Beans/Eclipse (for Java/PHP) was used for enhanced code management and debugging features.

3. **Tools**
    - XAMPP: These local server environments were used to simulate a web server for testing PHP-based applications and managing the MySQL database.
    - Git: Version control was managed using Git, with GitHub serving as the remote repository. This facilitated team collaboration, code tracking, and change management.
    - Postman: Used for API testing and debugging during backend development, particularly useful when integrating REST ful services.
4. **Technologies**

    - **MySQL:**
      A relational database system was used to store and manage structured data such as student records, scholarship details, and application history.
    - **Apache/Nginx:**
      Web server technologies used to host and serve the web application during deployment or testing.

4.2 Algorithm/ Pseudocode Used

The core logic of the Scholarship Management System revolves around processing student applications, validating eligibility, and managing approvals.

1) **Student Application Submission Algorithm**

Pseudocode:

Start

Prompt student to fill the scholarship application form

Validate input fields (e.g., name, ID, course, GPA, income)

If any field is invalid:

Display error message

Return to form

Else:

Save application data to the database

Set application status to "Pending"

Display confirmation message

End

This algorithm ensures that only valid applications are submitted and stored. Field validation prevents incomplete or incorrect data from entering the system.

2) Eligibility Check Algorithm

Pseudocode:

Start

Retrieve student application from the database

Check GPA ≥ required threshold

Check family income ≤ allowed limit

If both conditions are satisfied:

Mark student as "Eligible"

Else:

Mark student as "Ineligible"

Save eligibility status to database

End

This logic ensures that only students who meet academic and financial criteria are considered for scholarships.

3) Admin Approval Algorithm

Pseudocode:

Start

Admin logs in and views all "Pending" applications

For each application:

Review submitted documents and eligibility status

If documents are valid and student is eligible:

Update application status to "Approved"

Else:

Update status to "Rejected"

Notify student via email or dashboard

End

Admins make final decisions based on eligibility and document verification, and update the status accordingly.

4) Scholarship Disbursement Algorithm

Pseudocode:

Start

For each "Approved" application:

Generate disbursement record

Transfer scholarship amount to student's account (simulated)

Update payment status to "Paid"

Log transaction

Notify student of successful disbursement

End

Handles financial transactions (or simulation thereof) and keeps track of which students have received scholarship funds.

Below is the pseudocode for the Scholarship Application Process:

```
Function ApplyForScholarship(studentID, scholarshipID, documents):
    If not ValidateDocuments(documents):
        Return "Invalid Documents"
    If not CheckEligibility(studentID, scholarshipID):
        Return "Not Eligible"
    SaveApplication(studentID, scholarshipID, documents)
    NotifyAdmin("New Application Submitted")
    Return "Application Submitted Successfully"
```

This algorithm ensures that each student is eligible and submits all required documentation before the application is stored and an admin is notified.

4.3 Testing Techniques: In Context of Project Work

Testing is a crucial phase in the software development life cycle (SDLC) aimed at ensuring the reliability, functionality, performance, and security of the system. For the Scholarship Management System**,** multiple testing techniques were applied to validate that the system performs as expected and meets both user and business requirements.

Here's a breakdown of the key testing modules:

1. **UnitTesting**

   Unit testing involves testing individual modules or components of the system in isolation to verify that each one works as intended.
   Each function or module —such as student registration, login, application submission, and eligibility checks —was tested individually. For instance, the GPA validation function was tested with various edge cases to ensure accurate eligibility results.

   Tools Used:

- PHP Unit (for PHP-based backend)
- Manual function testing (for simpler modules)

2. **Integration Testing**

   Integration testing focuses on verifying that different modules or components of the system work together correctly. Once individual components such as the login system, application form, and database interactions were unit tested, integration testing ensured that they functioned seamlessly when combined. For example, after submitting an application, the eligibility check module correctly fetched data and returned the right status.

   Tools Used:

- API and Frontend
  - o Verify that API calls return the correct data
  - o Handle success and failure cases
- Database and Backend
  - o Test database transactions (commit/rollback)
  - o Check CRUD operations for scholarships and users
- Authentication Flow
  - o User login, logout, and session management
  - o Role-based access control (Admin, Student, Clerk)

3. System Testing

System testing validates the complete and fully integrated system against the defined requirements. The entire Scholarship Management System was tested end-to-end, simulating real-world scenarios. These included a complete cycle from registration to scholarship disbursement. The goal was to ensure that the system performs all tasks without failure under normal conditions.

Activities Involved:

- Scholarship Application Process

o Student submits an application

- Email and Notification System

o Verify email notifications for application status updates

o Ensure correct recipient and content

- Performance Testing

o Load test (handling multiple users at once)

o Respnse time and database query optimization

4. ValidationTesting(Functional)

This type of testing ensures that the software behaves according to the business requirements.

All functional requirements, such as applying for a scholarship, receiving approval, and viewing disbursement status, were tested for correct implementation. For example, the system was checked to ensure that only eligible students could proceed past the application phase.

5. VerificationTesting(Non-Functional)

Non-functional testing ensures aspects like performance, usability, and reliability.

- **Performance Testing:** Checked how the system behaves with multiple concurrent users.
- **Usability Testing:** Verified that the user interface was intuitive and easy to navigate for both students and admins.
- **Security Testing:** Ensured that login mechanisms were secure and data access was restricted properly.

6. User Acceptance Testing (UAT)

UAT is performed by the end users to verify whether the system meets their expectations. Sample students and administrators used the system and provided feedback. Their suggestions were used to refine the interface and workflows, ensuring the system was practical and user-friendly in real-world use.

. • Students

o Can they apply for scholarships easily?

o Is the application form intuitive?

• Clerks

o Can they efficiently review applications?

o Do they receive proper notifications?

• Admin

o Can they manage scholarships and users effectively?

o Is the dashboard providing useful insights?

1.4 Test Cases Designed for the Project Work

To ensure the robustness and correctness of the GNDEC Scholarship Management System, a set of well-defined test cases were designed. These test cases cover various functional scenarios including user authentication, form validation, document handling, and admin operations. Each test case includes the input conditions, the expected outcome, and the result observed during testing to verify the system's behavior.

Test Case ID	Test Description	Input	Expected Output	Result
TC01	Student Login with valid credentials	Email: test@gndec.ac.in, Pass:1234	Dashboard Access	Pass
TC02	Submit scholarship form without document	Form data without file	Error Message: "Document Required"	Pass
TC03	Admin approves valid application	Application ID	Status updated to “Approved”	Pass
TC04	File upload with unsupported format	File: file.txt	Error Message: "Invalid file type"	Pass
TC05	Application by ineligible student	Student below eligibility criteria	Error Message: "Not Eligible"	Pass

Chapter 5 Results and Discussion

This section presents the key findings obtained from the implementation or research conducted in the project. It includes a detailed analysis and interpretation of the results, highlighting how they align with the project's objectives. Comparisons with expected outcomes, previous studies, or standard benchmarks are also discussed to provide meaningful insights. The aim is to explain the significance of the results and their implications in a clear and logical manner.

5.1 User Interface Representation

The User Interface (UI) of the Scholarship Management System plays a vital role in delivering an The proposed GNDEC Scholarship Management Portal's financial viability and whether the advantages outweigh the expenses of its creation, implementation, and upkeep are assessed by the Economic Feasibility Study. The UI of this system is designed with simplicity, clarity, and functionality in mind. The main goal is to ensure that every user, regardless of their technical background, can easily understand and operate the system without confusion. The interface is also developed to be responsive, ensuring it works effectively across different devices such as desktops, laptops, tablets, and smart phones.

5.1.1 Brief Description of Various Modules of the System

1. Student Interface

The student UI focuses on ease of use and straightforward navigation. Key features available to students include:

- Registration and Login: Simple forms for sign in up and login into the system.
- Application Form Access: Students can easily find and complete the scholarship application forms.
- Document Uploads: A user-friendly interface for uploading required documents.
- Application Status Tracking: Students can view the real-time status of the applications.
- Notification Panel: Displays alerts regarding submission deadlines, approvals, document.

2. Clerk Interface

The clerk UI is slightly more complex than the student's interface, as clerks are responsible for reviewing and verifying the submitted applications. Features include:

- Dashboard View: Overview of the number of applications submitted, verified, pending.
- Application Review Tools: Interface for reviewing individual applications, verifying uploaded documents, and providing remarks or feedback.
- Communication Tools: Ability to message students in case of missing print correct information.
- Record Management: Options to view, filter, and update application records easily.

3. Admin Interface

The admin UI offers comprehensive control and system-wide access. It is built to support management and monitoring tasks, including:

- User Management: Creation, modification, and removal of student and clerk accounts.
- Application Monitoring: Real-time analytics and tracking of application numbers, approval rates, and document statuses.
- Report Generation: Tools to export reports in formats like PDF or Excel for internal use or auditing purposes.
- System Configuration: Settings to manage deadlines, eligibility criteria, scholarship categories, and more.

4. Design and Technology

The UI is built using modern web development technologies:

- Frontend: HTML5, CSS3, JavaScript, and React.js to ensure a responsive and dynamic user experience.
- Styling: Clean design with consistent colors, fonts and icons to enhance visual appeal.
- Accessibility: Proper use of labels, alt text, and keyboard navigation ensures that the platform is accessible to users with disabilities.

Conclusion

In summary, the User Interface of the Scholarship Management System is designed with a role- based approach, ensuring that each type of user—Student, Clerk, or Admin—has access to the tools and information they need in a clear, organized, and efficient manner. The UI not only Supports core functionality but also promotes transparency, ease of access, and user engagement throughout the scholarship application and approval process.

5.2Snapshots of System

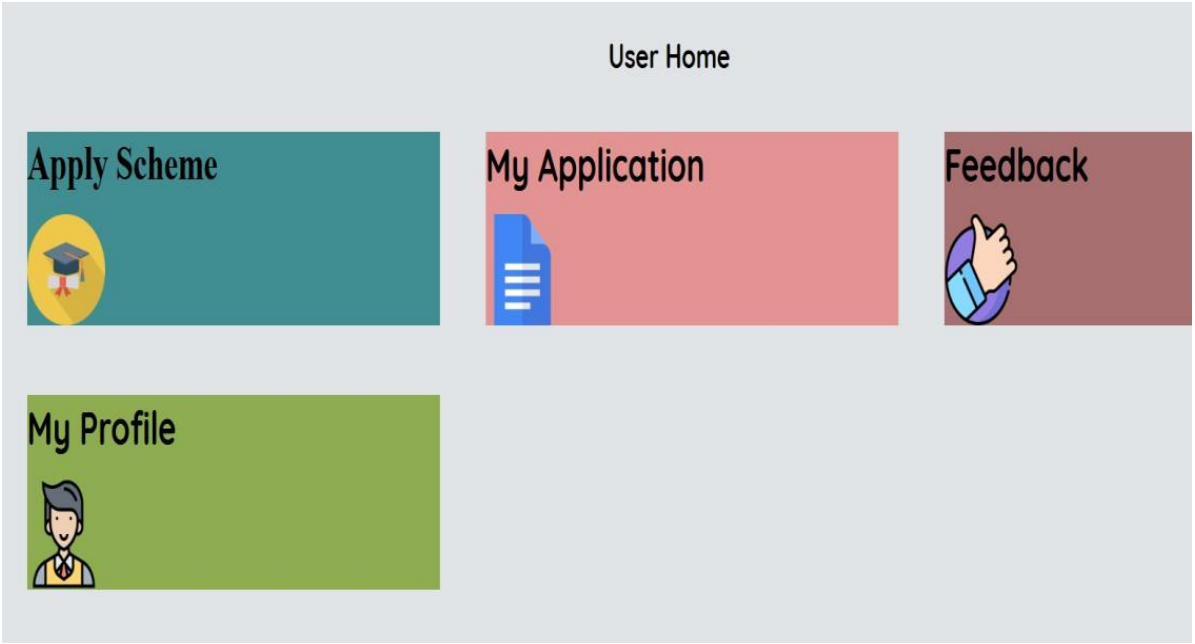
Snapshots of User interface are following:

1. **Snapshot of Login Details**

In login page different users can login.

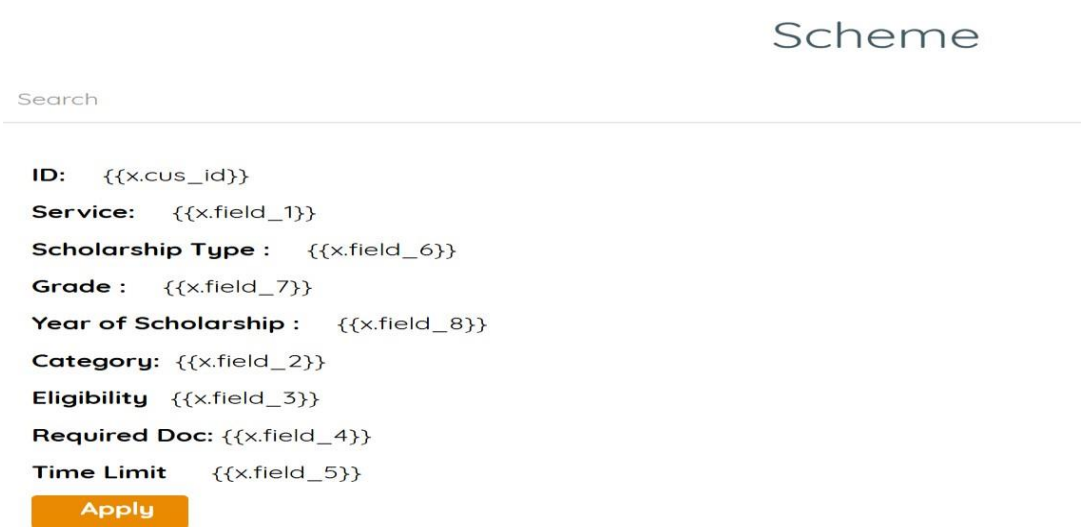
1. User login: It is for Students Login.
2. Staff or Clerk login: It is for Staff Login.
3. Admin Login: It is for admin Login.

2. **Snapshot of User Login Details**



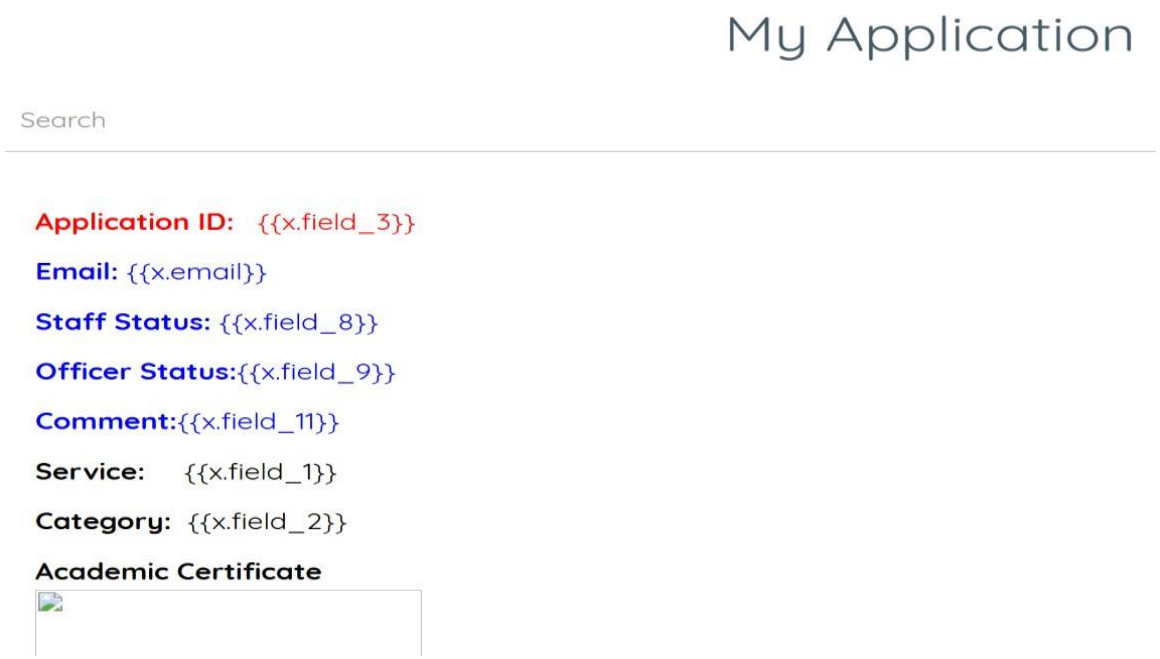
3. **Snapshot of User home page**

It is a user interface for students. In which student can Apply Schemes, See My Application, give feedback, and my Profile.

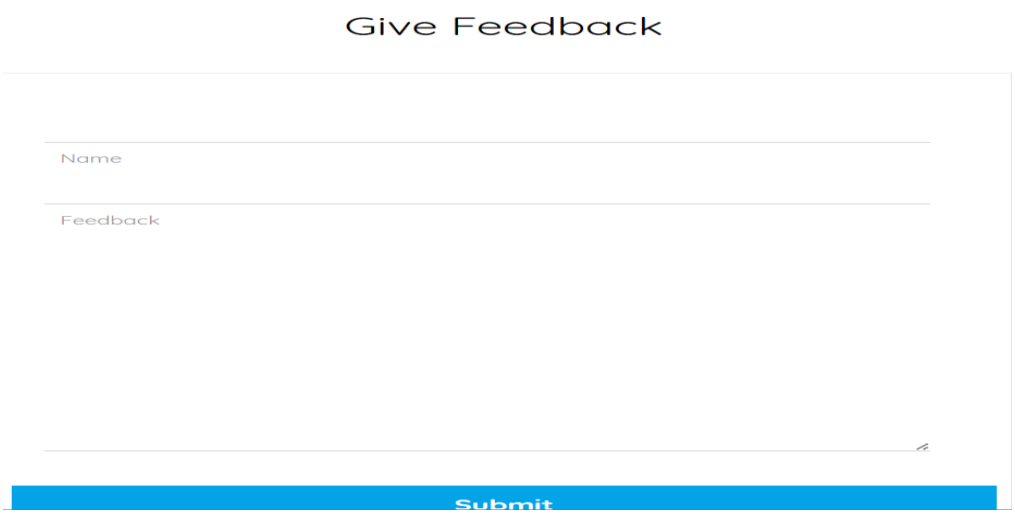


4. Snapshot of Scholarship Scheme

In this students can apply various scholarships.

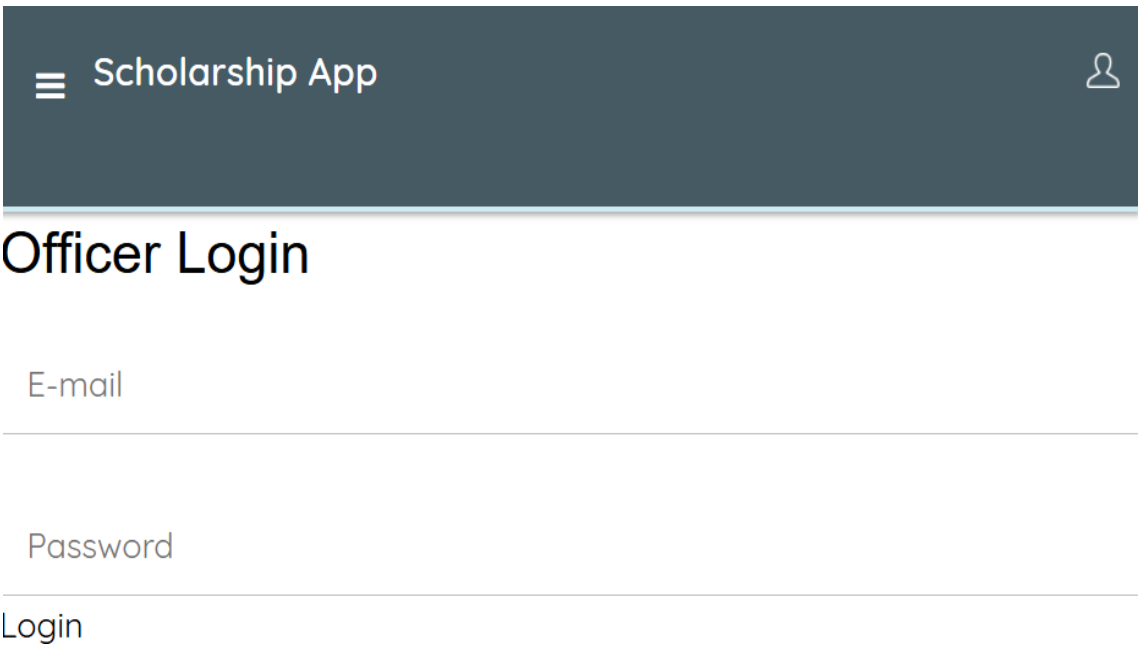


5. Snapshot of Student Application



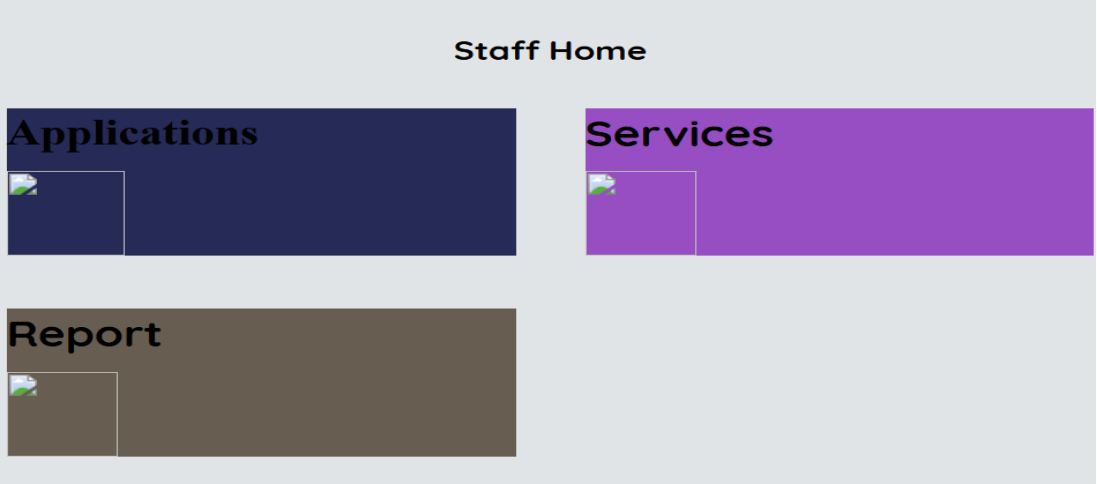
6. Snapshot of student feedback

In this module Students can give feedback.

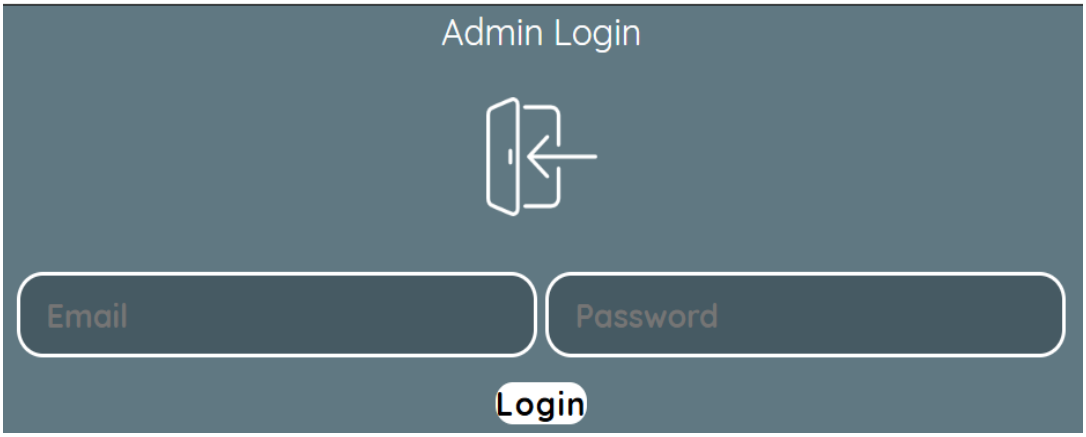


7. Snapshot of clerk Login

Clerk can login By this Interface.

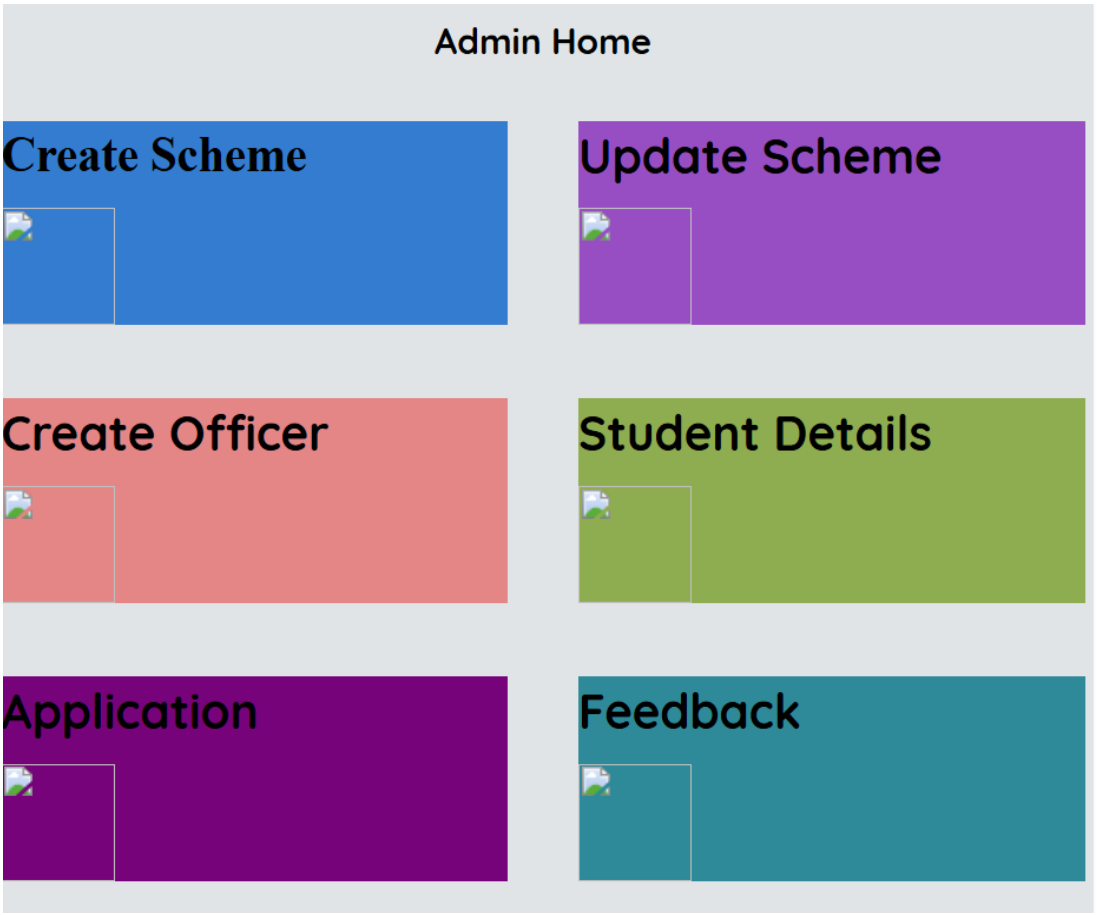


8. Snapshot of Clerk home page



9. Snapshot of Admin Login

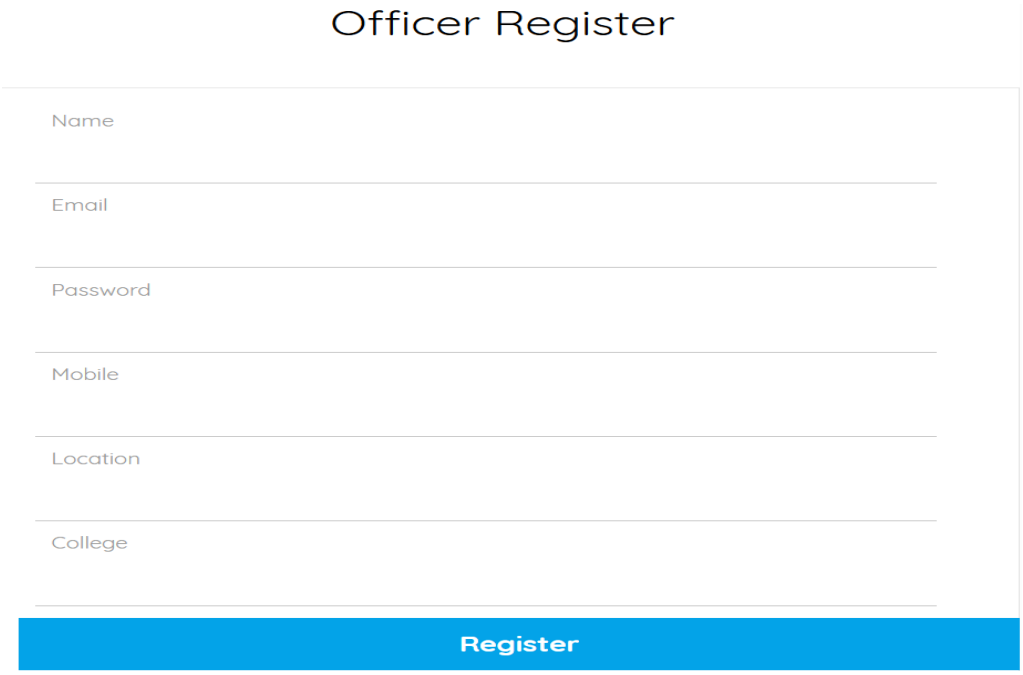
Admin can Login by this interface.



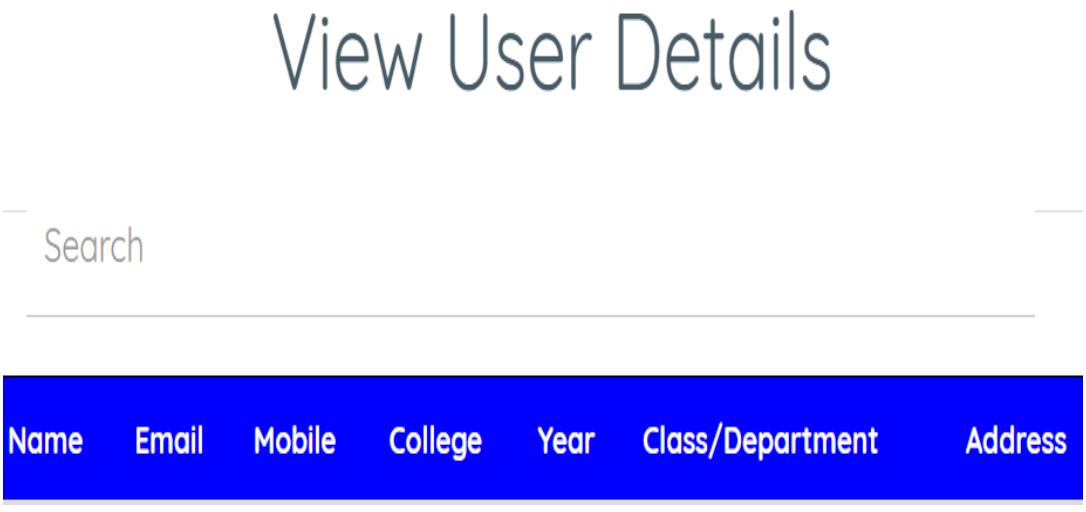
10. Snapshot of Admin Home Page

In home page admin can create Schemes, update scheme, create officer, see student details, applications, and check Feedback.

11. Snapshot of Admin Create Scheme



12. Snapshot of Clerk Register



13. Snapshot of Admin Report

Admin can see all Students details by making reports.

14. Snapshot of Feedback

Admin can see all students Feedback details.

5.3 Backend Representation

The backend is developed using PHP a powerful and lightweight web framework that supports clean architecture and fast development. The backend is structured to be **modular**, **scalable**, and **secure**, ensuring maintainability and smooth integration of new features in the future **(frontend)** and the **database**, handling user requests, and executing business rules.

Key Components of the Backend:

1. **Authentication and Authorization**
    - Secure login and session management.
    - Role-based access control (RBAC) is implemented to ensure that only authorized users can access specific parts of the system:
        - Students can only access their own records.
        - Clerks have limited access to verification sections.
        - Admins have full access to all data and controls.
2. **Business Logic Layer**
    - Validates and processes all application data.
    - Manages work flows like eligibility checking, status updating, and notification triggering.

- Ensures that application deadlines, scholarship criteria, and document types are enforced.

3. **Database Connectivity**
    - The backend communicates with a **My SQL** database using an ORM (like SQL Alchemy for Flask or Django ORM).
    - Handles CRUD operations (Create, Read, Update, Delete) for users, applications, scholarships, and uploaded files.
    - Ensures data integrity through constraints and validations.
4. **Notification System**
    - System notifications are generated by the backend to inform users about application status updates, deadlines, or verification results.
5. **Performance and Scalability**
    - The backend is designed to handle multiple concurrent requests.
    - Code is modular and uses caching techniques where appropriate to improve performance.

Conclusion

The backend of the Scholarship Management System acts as the **bridge between the user interface and the database**, ensuring that all operations are executed securely, efficiently, and accurately. By using a structured, scalable framework and applying best practices in authentication, data handling, and security, the backend provides a solid foundation for the smooth operation of the entire platform.

5.3Snapshots of Database:

The following snapshot presents the database schema of the Scholarship Management System, outlining key entities and their relationships to support efficient management of scholarship applications, approvals, and disbursements.

	Table	Action							Rows	Type	Collation	Size	Overhead
☐	admin	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	dataproject	☆	Browse	Structure	Search	Insert	Empty	Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	feedback	☆	Browse	Structure	Search	Insert	Empty	Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	schemes	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	staff	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	32.0 KiB	-
☐	studentdetails	☆	Browse	Structure	Search	Insert	Empty	Drop	3	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	users	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
	7 tables	Sum							11	InnoDB	latin1_swedish_ci	128.0 KiB	0 B

1. Snapshot of Admin Database

Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Triggers

Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)

```
SELECT * FROM `studentdetails`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

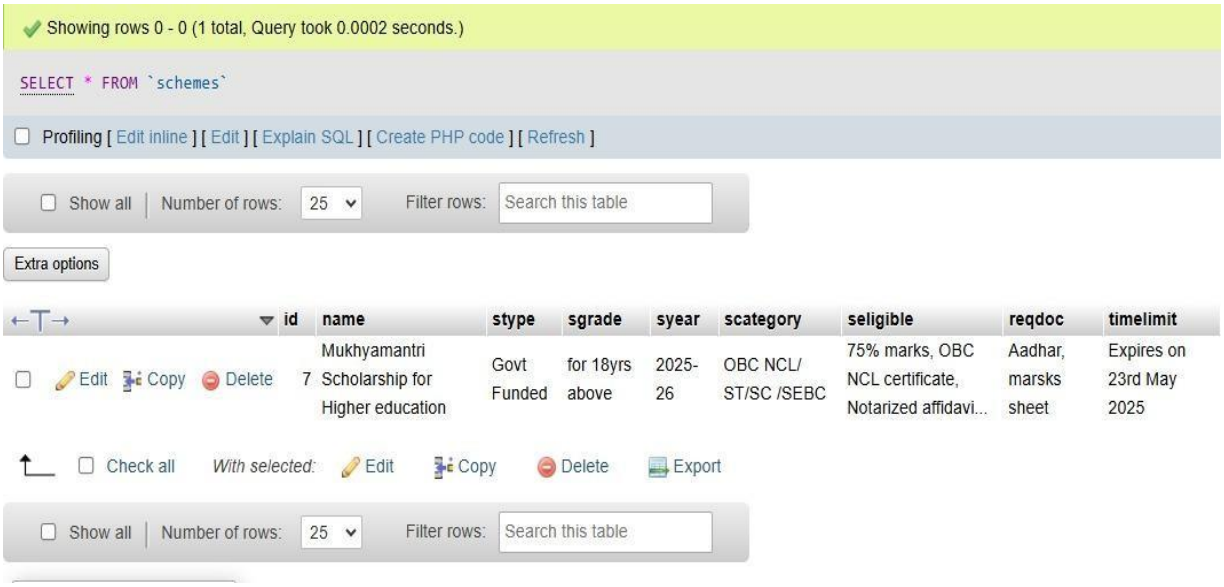
Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	id	fname	photo1	photo2	photo3	email	phone
☐ Edit Copy Delete	1	ysus	6 sem result.PNG	6 sem result.PNG	RES 6.PNG	harman2992003@gmail.com	5466553
☐ Edit Copy Delete	2	Real Test	WhatsApp Image 2025-04-22 at 21.45.34.jpeg	WhatsApp Image 2025-04-21 at 16.18.40.jpeg	WhatsApp Image 2025-04-21 at 16.18.36.jpeg	harman2992003@gmail.com	5466553
☐ Edit Copy Delete	3	Real Test	WhatsApp Image 2025-04-22 at 21.45.34.jpeg	WhatsApp Image 2025-04-21 at 16.18.40.jpeg	WhatsApp Image 2025-04-21 at 16.18.36.jpeg	harman2992003@gmail.com	5466553

Check all | With selected: Edit | Copy | Delete | Export

2. Snapshot of Database



3. Snapshot of Database

4. Snapshot of Backend Files

Chapter 6 Conclusion and Future Scope

6.1 Conclusion

The Scholarship Management Web Portal developed for Guru Nanak Dev Engineering College (GNDEC) offers a comprehensive digital solution that simplifies the cumbersome scholarship management process which traditionally uses a paper-based system. The portal improves the accessibility, transparency, and efficiency of the scholarship process as students can apply, submit documents and check the status of their applications in real-time.

The design of the portal has a simple user interface which reduces the administrative burden, minimizes errors, and provides a clear step-by-step guide for students and administrators alike. The integration of scholarship information such as eligibility, deadlines, and application procedures are readily accessible thus increasing the chances of success.

In essence, the project accomplished its primary objective of developing an easy-to-use, highly accessible, and secure web-based system that automates scholarship management at GNDEC.

6.2 Future Scope

The current system satisfies basic needs, but additional functional requirements could be added to improve the system's effectiveness.

1. Integration with National and Government Portals

Integrating the portal with national scholarship programs (NSP) and state-level scholarship databases would widen access for GNDEC students.

2. Mobile Application Development

Building a dedicated mobile app can enhance accessibility, allowing students to apply, receive notifications, and track applications on the go.

3. Advanced Notification System

Implementing SMS and push notifications (along with email alerts) will keep students better informed about deadlines, status updates, and missing documents.

4. AI-based Eligibility Checker

Adding an intelligent recommendation system using AI can help match students with the most suitable scholarships based on their profile and academic record.

5. Data Analytics and Reporting

Incorporating advanced analytics will allow administrators to generate detailed reports on scholarship trends, student participation, and fund distribution, aiding better policy decisions.

6. Enhanced Security Measures

As the system handles sensitive student data, adding multi-factor authentication (MFA), data encryption, and periodic security audits can further strengthen protection against cyber threats.

7. Feedback and Review System

Introducing a feedback module will allow students to rate their experience, helping developers and administrators continuously improve the system.

REFERENCES

1. Falogme, Carl Henz F. and Motol, Limuel U. 2017. Scholarship Management System for Municipality of Malvar" Retrieved from BatangasStateUniversityMalvar, Campus, June 27, 2018.

2. Gonzales, Laura Angelica O., Pesa, Yeersly D. and Valencia, Ronald R. 2016. Modification of the Scholarship Management Information System ofBatangas States University. Retrieved from Batangas State University Alangilan, Main Campus II, June 26, 2018.

3. Kumar, A., &Yadav, S.(2020). Automation of Scholarship Management System Using Web Technologies. International Journal of Computer Applications, 178(2), 1-6.

4. Rashmi, R., & Sharma, S. (2019). Scholarship Management System: A Web-Based Approach for Education Institutions. International Journal of Innovative Research in Computer and Communication Engineering, 7(9), 15-20.

5. Ahmed, M., & Hossain, M. S. (2020). "Design and Implementation of a Scholarship Management System: A Case Study" International Journal of Advanced Research in Computer Science and Software Engineering, 10(3), 112-118.

6. Gupta, R., & Verma, P. (2019). "E-Scholarship Management System: A Review of Existing Solutions" Journal of Information Technology Research, 8(2), 55-70.