



Daffodil
International
University

Department of CIS

TITLE OF THE PROJECT

Smart EDU

Submitted By

Deen Islam

ID: 183-16-367

Supervised By

Mr. Md Sarwar Hossain Mollah

Associate Professor & Head

Department of Computing & Information System (CIS)

Daffodil International University

Course Code: CIS (499)

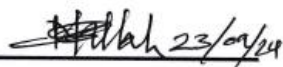
Spring-2024

Submission Date: 23 September 2024

APPROVAL

This Project titled “**Education Management System- (SmartEdu)**”, submitted by **Deen Islam**, ID No: **183-16-367** to the Department of Computing & Information Systems, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on- 23-09-2024.

BOARD OF EXAMINERS



Mr. Md. Sarwar Hossain Mollah
Associate Professor and Head
 Department of Computing & Information System
 Faculty of Science & Information Technology
 Daffodil International University

Chairman



Md. Nasimul Kader
Assistant Professor
 Department of Computing & Information System
 Faculty of Science & Information Technology
 Daffodil International University

Internal Examiner



Mr. Mehedi Hasan
Lecturer
 Department of Computing & Information System
 Faculty of Science & Information Technology
 Daffodil International University

Internal Examiner



Dr. Saifuddin Md. Tareeq
Professor
 Department of Computer Science and Engineering
 University of Dhaka, Dhaka

External Examiner

DECLARATION

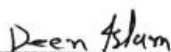
I, **Deen Islam** hereby declare that; the work in this dissertation titled "**Education Management System- (SmartEdu)**" has been done by me under supervision of **Mr. Md. Sarwar Hossain Mollah** Associate Professor and Head, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By



Md. Sarwar Hossain Mollah
Associate Professor and Head
Department of CIS
Daffodil International University

Submitted By



Deen Islam
ID: 183-16-367
Department of CIS
Daffodil International University

Acknowledgement

First and foremost, I want to express my heartfelt gratitude to Almighty Allah for providing me with this invaluable opportunity. Without His guidance, completing this project would not have been possible. I also extend my sincere thanks to DCL for selecting me as an intern trainee. This internship program has been incredibly enlightening and has contributed immensely to my personal and professional growth.

It is often said that a teacher is like a second parent, offering the guidance needed to help a student move forward in life. During my internship, I had the privilege of being mentored by two incredible individuals. **Md. Mazharul Islam**, my intern trainer, and **Md. Sarwar Hossain Mollah**, who supervised my academic work, consistently offered me invaluable advice and support. They helped me make informed decisions and never left me to face challenges alone. Their constant encouragement and guidance were instrumental in reducing my stress and giving me the mental strength to tackle complex tasks.

I am deeply grateful to both of them for their generosity, willingness to assist, and their unwavering belief in my abilities. Their mentorship has been crucial in helping me navigate challenges, and I thank them for teaching me how to manage significant projects in the future.

Dedication

This project is dedicated to Almighty God, my beloved parents, and my elder sister, all of whom have played an essential role in helping me complete my BSC program. I offer my deepest gratitude and praise to Almighty God for His constant support, protection, and guidance throughout this journey. I am also thankful for His blessings, which have been with me throughout my entire academic program.

Abstract

"Smart Edu" is a web-based application developed using the Odoo platform, leveraging Python and a PostgreSQL database. The system is designed to manage the daily operations of educational institutions efficiently. The development process follows the Agile Model, an iterative and incremental approach that allows for continuous improvement and adaptability throughout the project lifecycle.

The primary objective of Smart Edu is to provide a comprehensive solution that benefits both school authorities and students by streamlining administrative tasks, saving time, and reducing costs. This system is equipped to handle various school-related processes, including academic year management, division setup, media creation (e.g., language mediums like Bangla or English), teacher and student management, class and subject organization, fee structuring, and exam grading.

Smart Edu's design focuses on reducing staff workload, improving data security, and enhancing the overall productivity and cost-effectiveness of educational institutions. The centralized database and application are accessible from any location with an internet connection, ensuring that schools can manage their operations without the need for expensive hardware or software.

The system's platform independence allows it to be easily accessed from any internet-connected device, making it a versatile solution for schools of varying sizes and needs. Smart Edu is also customizable to meet the specific requirements of each institution, ensuring that it provides tailored support for unique operational demands.

In summary, Smart Edu is a powerful tool that enhances school management by providing a centralized, secure, and efficient platform for handling all aspects of educational administration. Its deployment on the Odoo platform ensures a robust and scalable solution that can adapt to the evolving needs of schools.

Table of Contents

Chapter 1 Introduction	1
1.1 Introduction	1
Chapter 2 Initial Study	2
2.1 Project Proposal	2
2.2 Background of the DCL/School Management System	3
2.3 Problem area	3
2.4 Possible solution	4
2.5 Objective	5
Chapter-3 Literature Review	5
3.1 Discussion on problem domain based on published articles	6
3.2 Discussion on problem solutions based on published articles	7
3.3 Comparison of three/four leading solutions	9
Chapter 4-METHODOLOGY	11
4.1 What to use	11
4.2 Why to use	13
4.3 Sections of methodology	14
4.4 Implementation plans	15
Chapter 5 Planning	17
5.1 Project Plan	17
5.1.1 Management Plan / Work Breakdown Structure:	17
5.1.2 Resource Allocation	19
5.1.3 Time Boxing:	19
5.1.4 Gantt chart:	20
Chapter 6 – Feasibility	21
6.1 All Possible Types of Feasibility	21
6.1.1 Operational feasibility	21
6.2 Cost Benefit Analysis	22
6.3 DSDM	22
Chapter 7 Foundation	23
7.1 The Problem Area Identification	23

7.1.1 Interview.....	23
7.1.2 Specific Problem Area Identification and Description	24
7.2 Rich Picture	24
7.3 Specific problem area identification and description	25
7.4 Possible Solution:	25
7.5 Overall Requirement List	25
7.5.1 Functional Requirements.....	25
7.5.2 Non-Functional Requirements	26
7.6 Which Technology to be implemented.....	26
Chapter 8 Exploration	27
8.1 Old System Use Case:	28
8.2 Activity Diagram:	29
8.3 Full System Use Case Diagram:	29
8.3.1 Breakdown of Use Case diagram according to the functions of DARUSSUNNAH SCHOOL MANAGEMENT system:.....	30
8.4 Full System Activity Diagram:.....	33
8.5 Requirements Catalogue.....	35
8.6 Requirement Prioritized:.....	36
8.7 Prototype of DCL school management system:	37
Chapter 9 Engineering	39
9.1 Modules of the new system	39
9.2 Use Case	41
9.3 ERD Diagram of School Management System	41
9.4 Sequence Diagram of School Management System.....	42
9.5 Component Diagram of School Management System	42
9.6 Deployment Diagram of School Management System	43
9.7 System Interface Design / Prototype of DCL/School Management System	43
Chapter 10 Deployment/Development	44
10.1 Core Module Coding Samples.....	44
10.2 Possible problem breakdown.....	47
10.3 Prioritization while developing	47

Chapter 11 – Testing	49
11.1 Test Plan Acceptance	50
11.2 Test Case	50
11.3 Unit Test	51
11.4 Module Testing	53
11.5 Integration Testing	54
Chapter 12 – Implementation	63
12.1 Training	63
12.2 Big Bang	64
12.3 Scaling	64
12.4 Load balancing	64
Chapter 13 Critical Appraisal and Evaluation	65
13.1 Objective that could be met	65
13.1.1 Success rate against each objective	65
13.1.2 How much better could have been done	65
13.1.3 Why it could not be done	65
13.1.4 Which objectives have been missed	66
13.1.5 Why these objectives have missed	66
13.2 Objectives totally not met / touched	66
13.2.1 Why it could not be touched	66
13.2.2 What could have been done	66
Chapter 14 – Lesson Learned	66
14.1 Pre Project – Review – Closing:	66
14.2 I have learned	67
14.3 The Problems I Have Faced	67
14.4 What solutions occurred	67
Chapter 15 – Conclusion	68
15.1 Summary of the project	68
15.2 Goal of the project	68
15.3 Success of the project	69
15.4 What I have done in the documentation (stages / activities / plans)	69

15.5 Value of the project	69
15.6 My experience	70
Appendix	70
Plagiarism Report.....	77

List of Figure

Figure 1: DCL/ School Management System Interface	2
Figure 2:stem features option on Smart Edu	9
Figure 3: Gantt Chart	20
Figure 4 : Rich picture	24
Figure 5:Simple data flow SIS	25
Figure 6:Old System Use Case	28
Figure 7:Activity Diagram	29
Figure 8:Full System Use Case	29
Figure 9: Use Case for Academic	30
Figure 10:Use Case for Library	31
Figure 11:Use Case for Payment	32
Figure 12:Activity diagram for Academic	33
Figure 13:Activity diagram for Library	33
Figure 14:Activity diagram for Payment	34
Figure 15:Login form Prototype	37
Figure 16:Dashboard Interface Prototype	37
Figure 17:Student add form Prototype	38

Figure 18:Student Profile Prototype	38
Figure 19:payment structure Prototype	39
Figure 20:Use case diagram	41
Figure 21:ERD diagram	41
Figure 22: Sequence diagram	42
Figure 23:Component diagram	42
Figure 24:Deployment diagram.....	43
Figure 25:System Interface Design.....	43
Figure 26:Database Connection code sample.....	44
Figure 27:Dashboard Code sample	44
Figure 28:Fees Head Code sample	45
Figure 29:Fees Structure Code sample	45
Figure 30:Fees Register code sample	45
Figure 31:Notice code sample	46
Figure 32:Exam Create code sample	46
Figure 33:Result Card code sample	46
Figure 34:Unit Test-1.....	51
Figure 35:Unit Test-2.....	52
Figure 36:Unit Test-2.1.....	53
Figure 37:Module Test-1	54
Figure 38:Integration Test-1.....	55
Figure 39: Integration Test-2.....	56
Figure 40:Integration Test-3.....	56
Figure 41:Integration Test-4.....	57

List of Table

Table 1: Work Breakdown Structure of DSMS.....	19
Table 2:Resource Allocation List	19
Table 3:List of the Time Boxes	20
Table 4: Cost Benefit.....	22
Table 5:Requirement Prioritized.....	36
Table 6:prioritization requirement	49
Table 7:School Management details	49

Table 8:Test case	51
Table 9:Unit Test Case- 1	51
Table 10:Unit test Case – 2.....	52
Table 11:Module test Case	53
Table 12:Integration test Case-1.....	54
Table 13:Integration test Case-2.....	55
Table 14:Test Details -2	60
Table 15:Domain based test	61
Table 16:Final test report.....	62
Table 17:training based	63

Chapter 1 Introduction

1.1 Introduction

In today's rapidly evolving educational landscape, the role of an efficient and streamlined educational system is crucial for the growth and development of any country. A school management system is a key tool designed to facilitate the smooth operation of educational institutions by digitizing and automating a wide range of academic and administrative processes. Many educational institutions are focused on enhancing the quality of instruction, and one of the critical aspects of this advancement is the effective management of educational resources. Modern technological tools, like school management systems, play an essential role in accelerating and optimizing educational processes, fostering better collaboration among students, teachers, and school staff.

Smart Edu is a robust school management system that supports the daily operations of educational institutions, serving as a comprehensive platform for managing various school management tasks. This system is composed of multiple modules, each designed to address specific operational challenges, such as:

- **Academic Year Management**
- **Division and Media Setup**
- **Teacher and Student Profile Management**
- **Class and Subject Management**
- **Fee Management**
- **Exam and Grading System**
- **Result and Report Card Generation**
- **Notice and Communication Management**

The **Smart Edu** system stands out due to its user-friendly interface, which significantly contributes to its growing adoption by educational institutions. This project aims to manage and optimize all school activities, offering features that are essential for modern schools to digitalize their day-to-day operations efficiently.

Developed using **Odoo**—a versatile platform powered by **Python** and **PostgreSQL**—Smart Edu ensures a scalable and secure solution that can be accessed from any device with an internet connection. By streamlining and automating procedures, Smart Edu addresses the needs of all stakeholders, helping them perform their roles more effectively. The system's flexibility allows it to be customized to meet the specific requirements of each institution, ensuring that it provides tailored support for their unique operational demands.

In essence, Smart Edu transforms school management by providing a centralized, efficient, and user-friendly platform that enhances the productivity and effectiveness of educational institutions.

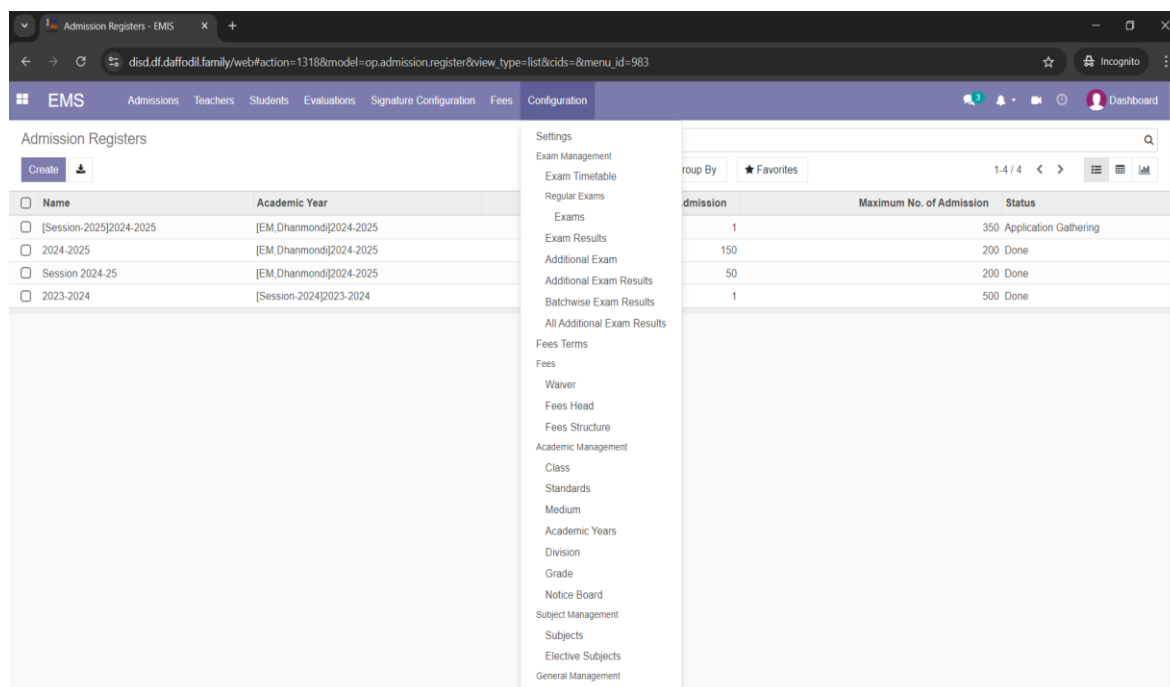


Figure 1: DCL/ School Management System Interface

Chapter 2 Initial Study

2.1 Project Proposal

The primary goal of this project is to develop a comprehensive "**school management system**" called **Smart Edu**. This system will be a web-based application designed to offer solutions for various challenges related to the daily operations of educational institutions. Smart Edu aims to automate the school's most critical and frequently performed operations, thereby improving efficiency and speeding up workflows.

The Smart Edu system is intended to be adopted by schools looking to enhance their operational processes through digital transformation. By automating key activities, Smart Edu will support various essential functions, including:

- **Result Card Generation:** Streamlining the process of creating and distributing student report cards based on exam results.
- **Student and Staff Information Management:** Providing a centralized database for storing and managing detailed information on students, teachers, and administrative staff.
- **Public Notices:** Facilitating the creation and distribution of public notices to all stakeholders within the school community.
- **Attendance System:** Automating the tracking of student and staff attendance to ensure accurate and up-to-date records.

The Smart Edu project is designed to address the diverse needs of modern educational institutions by providing a platform that is both flexible and scalable. By adopting this system, schools can streamline their operations, reduce manual effort, and ensure that all processes are conducted efficiently and accurately.

2.2 Background of the DCL/School Management System

One of the key components of efficient school management is the ability to facilitate constant communication between teachers, students, and administrative staff. The **Smart Edu** system is designed to meet this need by providing a comprehensive web application that addresses the various demands of its users simultaneously. The system is structured around three primary modules:

- **Admin**
- **Student**
- **Teacher**

The Smart Edu system includes a wide range of functional and non-functional features to support the daily operations of a school. Non-functional features, such as notice and event management, enhance the overall user experience, while functional features are critical to the core operations of the school. These functional features include adding students and teachers, creating classes and sections, organizing exams and routines, managing payments, handling promotions, and generating results.

The **Admin** module grants the administrator full access to the system. Administrators are responsible for managing all users, including teachers, students, and staff, along with overseeing various processes. The admin also controls all library-related operations, such as adding books, categorizing them, and approving or denying student requests.

The **Student** module allows students to view their profiles, subject grades, final results, attendance records, notices, and routines. It also provides students with a platform to communicate directly with their course teachers regarding any concerns, syllabus details, or exam-related information.

The **Teacher** module enables teachers to input grades for their assigned courses and create exams. Teachers have direct interaction with students, allowing them to address questions and provide guidance. Additionally, teachers manage the payment features within the system, including setting up fee heads, fee structures, and issuing fee receipts.

The **Smart Edu** system is designed to streamline and automate the essential operations of a school, ensuring that communication and administrative processes are handled efficiently. By providing a centralized platform for managing these tasks, Smart Edu helps educational institutions improve their productivity and maintain a smooth workflow.

2.3 Problem area

In many educational institutions, including those that Smart Edu aims to serve, student and faculty records are often maintained manually on paper. This traditional method is not only insecure but also prone to damage, loss, or unauthorized alterations. The manual process of managing these records is time-consuming, requiring significant effort to retrieve, update, and store information. This inefficiency leads to delays and increases the potential for human error.

Similarly, the task of compiling and computing academic results is cumbersome and error-prone when done manually. This process is not only inefficient but also consumes a considerable amount of time and resources, diverting personnel from more productive tasks. The lack of automation in these critical areas highlights the need for modern technological solutions.

Furthermore, the academic performance of students can be adversely affected by a lack of parental involvement. Many parents are unable to regularly attend school meetings due to time constraints, which can lead to a lack of awareness about their child's progress. This gap in communication can result in students not receiving the necessary support and encouragement from their families, ultimately impacting their academic outcomes.

Teachers and instructors also face significant challenges at the beginning of each academic year, particularly with the allocation of courses and classrooms. The complexity of assigning students to the correct classes and managing their schedules manually often leads to administrative bottlenecks.

Another recurring issue is the recording and tracking of students' grades. The manual process is not only labor-intensive but also increases the likelihood of errors, which can lead to discrepancies in students' academic records. This problem becomes particularly pronounced during the end-of-year assessments when accuracy is paramount.

Lastly, the educational process is further hindered by the lack of effective communication between instructors and students outside of school hours. Without a reliable system in place, important information may not be conveyed in a timely manner, affecting both teaching and learning outcomes.

The **Smart Edu** system is designed to address these challenges by providing a secure, automated platform that streamlines the management of records, grades, and communication, thereby enhancing the overall efficiency and effectiveness of the school management process.

2.4 Possible solution

In response to the challenges identified, I have explored potential solutions to address the inefficiencies in school management. To better understand the needs of educational institutions, I visited several schools that are still using traditional, manual methods for managing their operations. By gathering their requirements and conducting additional research online, I have identified the key features that must be developed to ensure these issues are effectively resolved.

The **Smart Edu** system will incorporate all the necessary features to automate and streamline the management processes. This includes secure digital record-keeping, automated grade calculation, efficient communication channels, and simplified administrative tasks like course and classroom allocation.

A major focus of the solution is on user interface design. I have prioritized creating a simple, intuitive interface to ensure that the system is highly user-friendly. The goal is to make the transition from manual to digital management as seamless as possible for all users, including those with limited technical experience.

To facilitate this transition, I will provide quick training sessions or a video presentation of the system to school staff. This will help them become familiar with the system's features and functionality, allowing them to adapt quickly and confidently. By taking these steps, potential issues can be mitigated, and the adoption of Smart Edu can be smooth and successful.

In summary, the **Smart Edu** system is designed to overcome the limitations of traditional school management methods by offering a comprehensive, easy-to-use solution that meets the specific needs of educational institutions. With proper training and a focus on user-friendly design, the system will enable schools to improve their operational efficiency and overall educational outcomes.

2.5 Objective

The **Smart Edu** project is developed using advanced RDBMS technology with the primary aim of automating traditional manual processes and reducing operational costs. The key objectives of the project are as follows:

- **Computerize All Systems:** Transition all school management processes from manual, paper-based methods to a fully computerized system. This ensures more efficient handling of academic and administrative tasks.
- **Reduce Time and Minimize Errors:** Significantly cut down the time required for various administrative tasks and reduce the potential for human error through automation and streamlined processes.
- **Automate System Management:** Implement automation for every aspect of the system, including record-keeping, grade calculations, attendance tracking, and communication, to ensure smooth and efficient management.
- **Centralized Database Administration:** Provide a centralized database that consolidates all school data in one secure location. This approach simplifies data management and ensures that information is easily accessible and up-to-date.
- **Simplified Controls for Operators:** Design a user-friendly interface with straightforward controls to make the system easy to use for administrators, teachers, and students. This will enhance user experience and facilitate quick adaptation to the system.
- **Eliminate Paperwork:** Move away from paper-based documentation by digitalizing all records and processes. This not only reduces physical storage needs but also contributes to environmental sustainability.

By achieving these objectives, the **Smart Edu** system aims to enhance the efficiency of school operations, reduce costs, and provide a more effective and reliable management solution.

Chapter-3 Literature Review

In Bangladesh, numerous educational institutions exist, yet a significant number still rely on outdated, paper-based systems to manage their daily operations. The reliance on manual processes presents several challenges, particularly for higher education institutions. These organizations often find themselves burdened with extensive paperwork, which requires substantial manual effort to manage and process.

Students enrolled in schools that utilize these traditional methods face difficulties in accessing information and obtaining necessary documentation. The administrative staff in these institutions struggle to track and retrieve documents efficiently, leading to increased workload and the necessity for a larger workforce dedicated solely to document management.

Recent advancements in technology have introduced various online applications designed to enhance the efficiency of school management. These applications offer features such as login systems, dashboards, result displays, attendance tracking, course management, assignment handling, and student activity alerts. The integration of such features aims to streamline operations and reduce the manual workload associated with traditional systems.

This chapter reviews relevant literature and practices in educational management systems, focusing on:

- **Information Systems:** Overview of current information systems used in educational institutions, their functionalities, and limitations.
- **Software Development Life Cycle (SDLC):** Discussion of different phases of SDLC, including planning, analysis, design, development, testing, and maintenance. This includes a comparison of various methodologies used in software development.
- **Comparison Between Methodologies:** Analysis of different software development methodologies (e.g., Agile, Waterfall) and their applicability to educational management systems. Justification for the chosen methodology in the context of Smart Edu.
- **Fact-Finding Techniques:** Examination of techniques used for gathering requirements and data collection in system development. This includes methods such as interviews, surveys, and observation, and selecting the most effective techniques for this research.
- **Development Tools:** Overview of development tools, focusing on database systems. Definitions, types, and comparisons of databases to select the most suitable one for Smart Edu.
- **Programming Languages:** Review of various programming languages, their definitions, types, and comparisons. This includes selecting the best programming language for developing the Smart Edu system.
- **Research on Related Systems:** Examination of existing school management systems, their features, and how they address similar challenges. This includes evaluating their strengths and weaknesses to inform the development of Smart Edu.

By exploring these aspects, this chapter aims to provide a comprehensive understanding of the current state of educational management systems, the technologies available, and the best practices for developing an efficient and effective system like Smart Edu.

3.1 Discussion on problem domain based on published articles

The implementation of "School Management Systems" (SMS) in our country faces several challenges, many of which are highlighted in various published articles. These challenges often stem from the gap between the expectations from such systems and their actual performance. It is crucial for educational institutions to select a system that aligns with their specific needs to ensure effective and efficient operation.

Here are some of the main problems associated with school management systems, as discussed in the literature:

- **Affordability:** Many schools face budget constraints that make it difficult to invest in advanced management systems. The cost of acquiring, implementing, and maintaining these systems can be prohibitive for institutions with limited financial resources. This often leads to the adoption of less comprehensive solutions or reliance on outdated systems.

- **Resources:** The availability of adequate resources, including hardware and software, is a significant concern. Schools may struggle to provide the necessary infrastructure, such as computers and reliable internet connections, which are essential for the effective functioning of a digital management system. Limited resources can impede the implementation and utilization of these systems.
- **Internet Services:** Reliable internet access is crucial for the effective operation of web-based school management systems. In areas with unstable or slow internet connections, schools may experience difficulties in accessing and using online systems. This can lead to disruptions in the management process and hinder the overall efficiency of the system.
- **Maintenance:** Regular maintenance and updates are necessary to keep school management systems functioning smoothly. However, many institutions lack the technical expertise or resources to perform ongoing maintenance, leading to potential system failures or security vulnerabilities. Effective support and maintenance plans are essential to ensure the system remains functional and secure over time.
- **Training:** Proper training for users is critical for the successful implementation of a school management system. Many schools encounter issues when staff members and administrators are not adequately trained to use the system. Without proper training, users may struggle to navigate the system's features and functionalities, leading to inefficiencies and reduced effectiveness.

Addressing these problems is essential for the successful deployment and operation of school management systems. By understanding and mitigating these challenges, educational institutions can better leverage technology to improve their administrative processes and enhance overall efficiency.

3.2 Discussion on problem solutions based on published articles

Addressing the challenges faced by school management systems requires targeted solutions that align with the insights provided in various published articles. Here's a discussion on potential solutions for key problems:

- **Affordability:** To tackle the issue of affordability, vendors should conduct thorough market research to understand the financial constraints of their target audience. The pricing strategy should balance the vendor's profit margins with the budgetary limits of educational institutions. Offering tiered pricing plans or subscription models can make the system more accessible to schools with varying budgets. Additionally, providing options for scaling the system according to the institution's size and needs can also help manage costs effectively.
- **Internet Connection:** For web-based systems, ensuring reliable internet connectivity is crucial. Vendors should clearly communicate the necessity of a stable internet connection for optimal system performance. This includes providing detailed information on the minimum internet requirements and offering guidance on improving connectivity if needed. With the increasing availability of affordable and high-speed data connections, vendors should emphasize the benefits of investing in good internet infrastructure.

Additionally, schools should be encouraged to consider backup internet solutions or offline capabilities to mitigate connectivity issues.

- **Training Users:** Effective training is essential for the successful implementation of a school management system. Vendors should offer comprehensive training programs for all users, including administrators, teachers, and support staff. This training can be provided through workshops, webinars, or video tutorials that cover system functionalities and best practices. Regular refresher courses and ongoing support can help users stay proficient and address any issues that arise.
- **System Monitoring and Maintenance:** Continuous monitoring and maintenance of the system are crucial for its long-term success. Vendors should provide robust support services, including regular system updates, bug fixes, and security patches. Implementing a user-friendly support system, such as a helpdesk or online support portal, can facilitate prompt resolution of issues. Additionally, establishing clear protocols for system monitoring and performance evaluation can help identify and address potential problems before they impact users.

By addressing these challenges with targeted solutions, vendors and educational institutions can enhance the effectiveness and accessibility of school management systems, ensuring they meet the needs of their users and contribute to improved administrative efficiency.

3.3 Comparison of three/four leading solutions

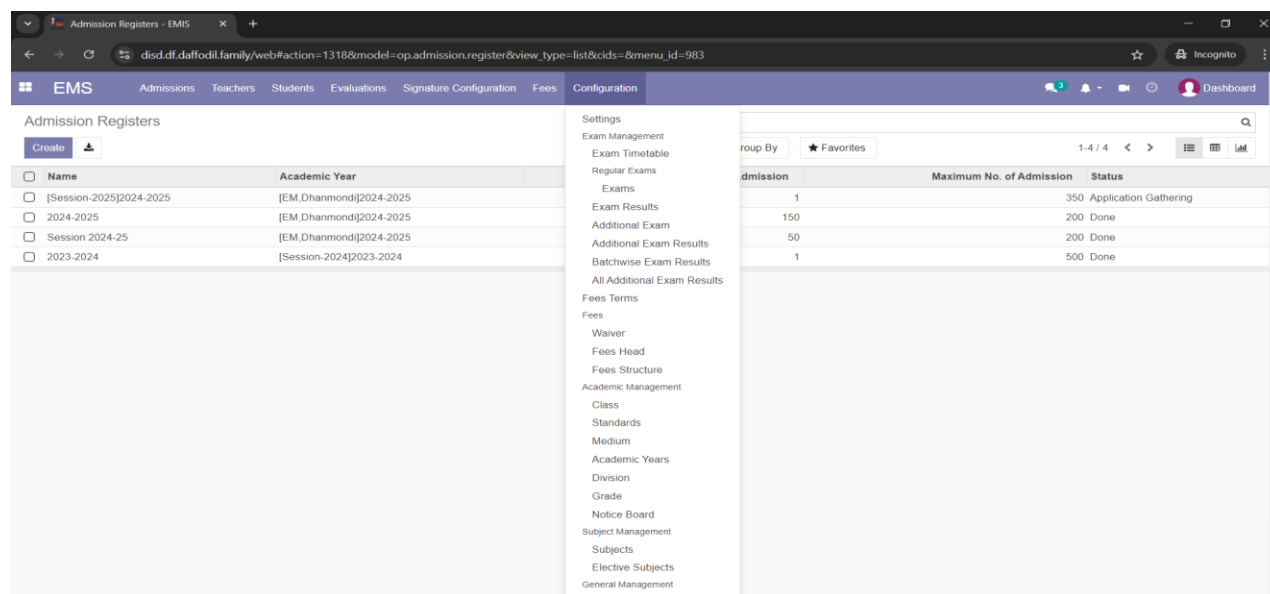


Figure 2:stem features option on Smart Edu

Comparison of three/four leading solutions

Features:

- **Fees Payment Management:** Integrated system for tracking and managing student fees.
- **E-Library System:** Digital library management for easy access to educational resources.
- **Result Card:** Automated generation of student report cards.
- **Assignment:** Tools for creating and managing assignments.
- **Attendance Tracking:** System for recording and displaying student attendance.
- **Grading System:** Comprehensive grading management for exams and assignments.

Limitations:

- **Data Graphical Visualization:** Lacks advanced graphical data representation features for analytics.
- **Non-Banking Payment Integration:** Does not support direct banking transactions for fee payments.
- **Message System:** Limited messaging capabilities for communication between users.

Recommended approach

Administration Panel

The Administration Panel provides comprehensive control over the school's operational processes. Key features include:

- **User Group Setup:** Manage different user groups such as students and teachers. Create and assign roles and permissions to ensure appropriate access levels.
- **Create Class:** Define and set up new classes, including class names, levels, and descriptions.
- **Create Section:** Organize classes into sections to accommodate different student groups or specializations within a class.
- **Create Course:** Set up courses or subjects offered by the school, including course descriptions and prerequisites.
- **Assign Teacher:** Assign teachers to specific classes and courses, ensuring that each course has the appropriate instructor.
- **Create Exam:** Design and schedule exams for various courses, including setting exam dates, formats, and durations.
- **Routine Setup:** Establish and manage class schedules, including daily routines and special timetables.

Teacher Panel

The Teacher Panel is designed to support educators in managing their responsibilities efficiently. Key features include:

- **Create Exam:** Develop and schedule exams for the courses they are responsible for, including creating exam papers and setting deadlines.
- **Marks Upload:** Enter and upload student marks for exams and assignments.
- **Fees Payment:** Manage and monitor fees payment related to their courses or department.
- **Syllabus Upload:** Upload and manage syllabi for their courses, providing students with access to course outlines and requirements.
- **Attendance:** Record and manage student attendance for their classes.
- **Holiday Setup:** Define and manage holidays and breaks within their courses or classes.
- **Student Attendance:** Track and view attendance records for students in their classes.

Student Panel

The Student Panel focuses on providing students with access to essential information and services. Key features include:

- **View Exam Schedule:** Access and view the schedule of upcoming exams.
- **View Result:** Check and download their exam results and academic performance.
- **Payment Receipt:** View and download receipts for payments made, including tuition fees and other charges.
- **View Attendance:** Check attendance records and monitor their participation in classes.

- **View Syllabus:** Access syllabi for their courses to understand course requirements and objectives.
- **Apply Library:** Apply for library access and manage their library account.
- **Book Search:** Search for books and other resources available in the library.
- **View Notices:** Receive and view important notices and announcements from the school.
- **View Events:** Stay updated on school events and activities.

Chapter 4-METHODOLOGY

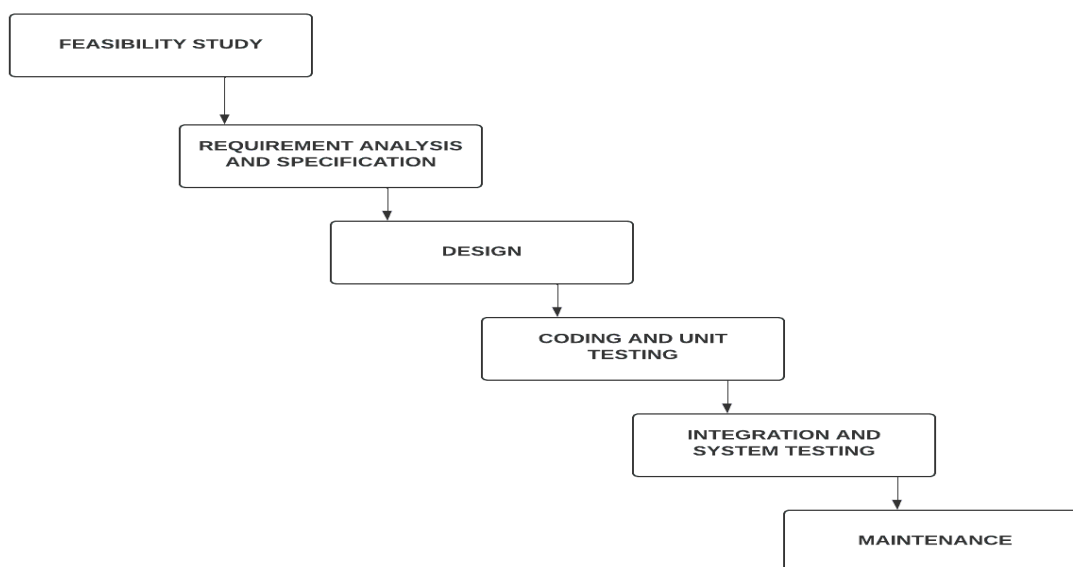
This chapter outlines the methodology used to develop the **Smart Edu** system, focusing on the Software Development Life Cycle (SDLC) and justifying the choice of methodology for the project.

4.1 What to use

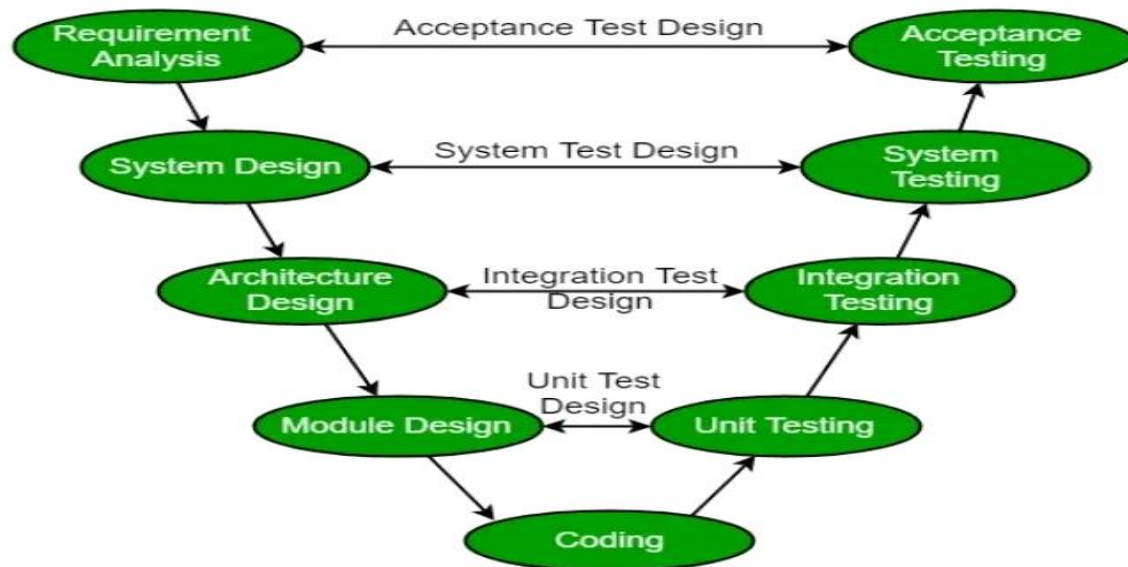
The Software Development Life Cycle (SDLC) is a framework for designing, developing, and maintaining software. It outlines the processes, phases, milestones, and progress of software development. Different models can be applied in the SDLC, each with its own strengths and weaknesses. For the development of **Smart Edu**, we will explore several prominent models:

- **Waterfall Model**
- **V Model**
- **Dynamic Systems Development Model (DSDM)**

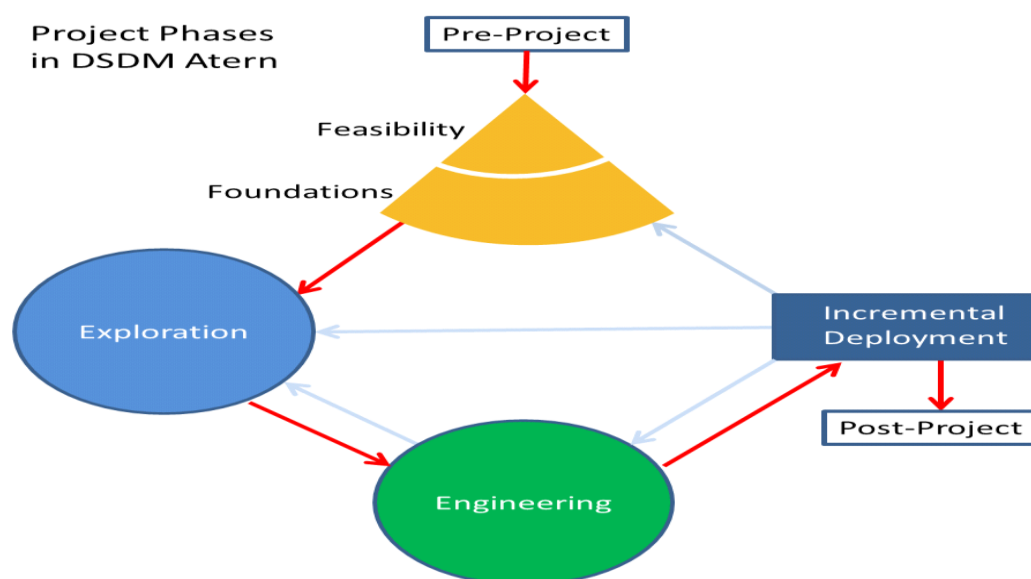
Waterfall Model: Follows a sequential, linear approach where each phase must be completed before moving on to the next. Suitable for projects with clearly defined requirements and limited changes.



V Model: Focuses on verification and validation, with each development phase having a corresponding testing phase. Useful for projects with clear requirements but can be challenging for complex or long-term projects.



Dynamic Systems Development Model (DSDM): An iterative Agile model that emphasizes flexibility and adaptability. It is well-suited for projects that require frequent changes and continuous user feedback.



4.2 Why to use

Dynamic Systems Development Model (DSDM) is selected for the **Smart Edu** project for the following reasons:

- **Adaptability:** DSDM allows for frequent changes and adjustments based on user feedback and evolving requirements, which is crucial for a school management system where needs can change over time.
- **Flexibility:** The model provides flexibility in handling changes, making it ideal for projects with dynamic requirements.
- **Business Alignment:** DSDM ensures that the development aligns with the strategic goals of the organization, delivering real commercial value.
- **Iterative Development:** Enables iterative development and rapid delivery of functional components, ensuring continuous improvement and user satisfaction.

Waterfall Model and **V Model** are also considered, but they have limitations that make them less suitable for the **Smart Edu** project:

- **Waterfall Model:**
 - **Advantages:** Provides a structured and linear approach with clear documentation and well-defined stages. Ideal for projects with fixed requirements and smaller scope.
 - **Disadvantages:** Less flexible and not suitable for projects with frequent changes. Testing occurs late in the process, which can lead to issues being discovered late.
- **V Model:**
 - **Advantages:** Emphasizes early testing and validation, providing a disciplined approach to development. Suitable for smaller projects with clear requirements.
 - **Disadvantages:** High risk and complexity for larger projects. Difficult to accommodate changes once development has started.

Dynamic Systems Development Model (DSDM) offers a balanced approach, allowing for flexibility and iterative development, which aligns with the needs of the **Smart Edu** system.

Applying Methodology for the School Management System

Smart Edu will adopt the DSDM methodology for its development due to its ability to handle evolving requirements and its focus on delivering value through iterative development.

- **Iterative Development:** Allows for the continuous refinement of features based on user feedback.
- **Flexibility:** Enables adjustments to be made as the project progresses, ensuring the final product meets the needs of the educational institution.
- **Stakeholder Involvement:** Encourages active involvement of stakeholders throughout the development process, leading to better alignment with their needs.

By utilizing DSDM, the **Smart Edu** system can be developed efficiently, with the ability to adapt to changing requirements and deliver valuable features incrementally.

4.3 Sections of methodology

The **Smart Edu** project follows the Dynamic Systems Development Model (DSDM) methodology, which is structured into distinct phases to ensure effective project management and successful system delivery. Below is a detailed description of each phase and its sub-phases:

Pre-Project Phase

- **Proposal Evaluation:** During this initial stage, various project proposals are reviewed and evaluated. The most feasible proposal is selected based on criteria such as cost, technical requirements, and alignment with the project's objectives.
- **Finance and Viability Assessment:** Estimations of the project's financial requirements are made, and the project's viability is assessed. This includes budgeting for resources, technology, and other essential components.

Project Lifecycle Phase

The Project Lifecycle Phase is subdivided into several key sub-phases:

- **Feasibility Study**
 - **Objective:** Develop a concept for project management rather than working on the system itself. This involves exploring the project's feasibility and assessing the costs.
 - **Activities:**
 - Address project management queries.
 - Assess and document feasibility.
 - Create a feasibility report that will guide the system development process.
- **Study of the Business**
 - **Objective:** Analyze the business and its needs, focusing on flexibility and requirements.
 - **Activities:**
 - Conduct a detailed analysis of business requirements and constraints.
 - Identify and prioritize project criteria based on user needs.
 - Create and prioritize a set of criteria for the project development.
- **Iteration of the Functional Model**
 - **Objective:** Develop functional prototypes based on collected requirements and validate them with end users.
 - **Activities:**

- Convert requirements into functional models.
- Develop and demonstrate functional prototypes to users.
- Gather feedback and validate the effectiveness of prototypes.
- **Iteration of Design and Development**
 - **Objective:** Transform functional models into final products that meet user needs.
 - **Activities:**
 - Create design prototypes based on earlier functional models.
 - Test prototypes with users to ensure satisfaction.
 - Refine and finalize the design based on user feedback.
- **Implementation**
 - **Objective:** Deploy the system and train users for effective utilization.
 - **Activities:**
 - Implement the final system.
 - Provide training and support for prospective users.
 - Ensure users are equipped to use the system effectively.

Post-Project Phase

- **Testing and Validation:** Conduct thorough testing to ensure that all system components work as expected and meet the defined requirements.
- **Maintenance and Updates:** Perform regular maintenance and updates to address any issues, enhance functionality, and keep the system up to date with evolving needs.

4.4 Implementation plans

The implementation phase is critical for ensuring that the **Smart Edu** system is effectively deployed and operational for all users. This stage involves transitioning the completed application from development to actual use, addressing any issues that arise, and ensuring a smooth rollout. Below is a detailed outline of the implementation plans:

1. Release Planning

- **Release Criteria:** Define the criteria that must be met before the system is released. This includes successful completion of testing phases, resolution of any critical issues, and user acceptance of the system.
- **Deployment Strategy:** Choose a deployment strategy that suits the needs of the institution. Options may include a phased rollout, where the system is introduced in

stages, or a full-scale deployment, where the system is made available to all users simultaneously.

- **Environment Setup:** Prepare the production environment where the system will be deployed. This involves configuring servers, databases, and any other infrastructure required for the system to function properly.

2. Testing

- **System Testing:** Conduct comprehensive testing to ensure that the system operates as expected in the production environment. This includes functional testing, integration testing, and performance testing.
- **User Acceptance Testing (UAT):** Engage end users in testing the system to validate that it meets their needs and expectations. Gather feedback from users to identify any issues or areas for improvement.
- **Issue Resolution:** Address any issues or defects identified during testing. Ensure that all critical bugs are resolved before the system is fully deployed.

3. Deployment

- **Data Migration:** If applicable, migrate existing data from previous systems to the new **Smart Edu** system. This may involve data cleansing, mapping, and transformation to ensure compatibility with the new system.
- **System Installation:** Install the system on the production servers and ensure that all components are properly configured. This includes setting up user accounts, permissions, and system settings.
- **Go-Live:** Officially launch the system for use by the institution. Ensure that all users have access to the system and that they are informed about the go-live date and any relevant instructions.

4. Training and Support

- **User Training:** Provide training to end users on how to use the new system effectively. This may include hands-on training sessions, workshops, and providing user manuals or guides.
- **Support Resources:** Establish support channels to assist users with any issues or questions they may have. This could include help desks, online support forums, and dedicated support staff.

5. Post-Implementation Review

- **Monitoring and Evaluation:** Continuously monitor the system to ensure it is functioning as expected. Evaluate user feedback and system performance to identify any areas for improvement.
- **System Maintenance:** Plan for ongoing maintenance and updates to address any issues that arise and to keep the system up to date with evolving needs.
- **Feedback Collection:** Regularly collect feedback from users to understand their experiences and identify any additional requirements or enhancements.

Chapter 5 Planning

5.1 Project Plan

The **Project Plan** is a comprehensive document that outlines the strategy for executing and monitoring the project. It details the project's scope, cost, and timeline while also addressing risk management, resource allocation, and communication. Effective planning is crucial for reducing risks, overcoming challenges, and ensuring the project proceeds smoothly. This section includes key elements such as the Work Breakdown Structure (WBS), resource allocation, time boxing, and the Gantt chart.

5.1.1 Management Plan / Work Breakdown Structure:

The **Work Breakdown Structure (WBS)** is a hierarchical framework used to organize and manage the tasks of a project. It decomposes the project into smaller, more manageable components, allowing for detailed planning and assignment. Each level of the WBS breaks down the project into increasingly specific tasks and subtasks, facilitating clearer assignment and tracking.

Key Points:

- **Hierarchical Decomposition:** Breaks the project into major deliverables and then into smaller tasks.
- **Task Management:** Helps in defining project scope and assigning responsibilities.
- **Milestones:** Identifies significant milestones and deliverables.

Example WBS for Smart Edu Project:

- **Project Initiation**
 - Define project scope
 - Identify stakeholders
 - Develop project charter
- **Planning**
 - Develop project plan
 - Create detailed schedule
 - Allocate resources
- **Design**
 - System architecture design

- User interface design
- Database schema design
- **Development**
 - Coding
 - System integration
 - Testing
- **Implementation**
 - Deployment
 - Training
 - User support
- **Monitoring and Control**
 - Track progress
 - Manage risks
 - Quality assurance
- **Closure**
 - Finalize documentation
 - Project review
 - Closeout

	Task Name	Duration(Days)	Start	End
1	Introduction	10	28-Jul-23	8/7/2023
2	Initial Study	5	8/8/2023	8/12/2023
3	Literature Review	4	8/13/2023	8/18/2023
4	Methodology	6	8/19/2023	8/24/2023
5	Planning	10	8/25/2023	9/4/2023
6	Feasibility	7	9/5/2023	9/12/2023
7	Foundation	13	9/13/2023	9/25/2023
8	Exploration	15	9/26/2023	10/5/2023
9	Engineering	35	10/6/2023	11/10/2023
10	Deployment	15	11/11/2023	11/26/2023
11	Testing	5	11/27/2023	12/1/2023
12	Implementation	5	12/2/2023	12/6/2023
13	Critical Appraisal and Evaluation	5	12/7/2023	12/11/2023
14	Lessons Learning	3	12/12/2023	12/14/2023
15	Conclusion	2	12/15/2023	12/16/2023
	Total	140 Days		

Table 1: Work Breakdown Structure of DSMS

5.1.2 Resource Allocation

Resource Allocation involves assigning the best available resources to project activities and tasks. It aims to ensure that resources are used efficiently, preventing both underuse and overuse. Proper resource allocation maximizes the potential of organizational assets and enhances overall project performance.

Key Aspects:

- **Resource Assignment:** Allocate resources based on skills and availability.
- **Workload Management:** Monitor and adjust workloads to maintain balance.
- **Resource Optimization:** Ensure optimal use of resources to achieve project goals.

Example Resource Allocation for Smart Edu Project:

- **Personnel:** Developers, designers, testers, project managers, trainers.
- **Tools:** Development tools, project management software, testing tools.
- **Budget:** Financial resources allocated for development, testing, and deployment.

project is required to meet the pre-scheduled project delivery schedule:

	Task Name	Duration(Days)	Resource
1	Introduction	10	Analyst,User
2	Initial Study	5	Analyst
3	Literature Review	4	Analyst,Team Leader
4	Methodology	6	Analyst,Developer
5	Planning	10	Analyst,Developer,Team Leader,Designer
6	Feasibility	7	Analyst,Team Leader
7	Foundation	13	Analyst,Developer,Designer
8	Exploration	15	Analyst,Developer,Designer
9	Engineering	35	Developer,Designer
10	Deployment	15	Analyst,Developer
11	Testing	5	Analyst,Developer,Tester,User
12	Implementation	5	Analyst,Developer,Tester,User
13	Critical Appraisal and Evaluation	5	Analyst,Developer,Tester
14	Lessons Learning	3	Analyst,Developer,User
15	Conclusion	2	Analyst

Table 2:Resource Allocation List

5.1.3 Time Boxing:

Time Boxing is a project management technique where a fixed amount of time is allocated to complete a specific deliverable. The scope and quality of the deliverable must be managed within this time frame, ensuring that deadlines are met while maintaining quality standards.

Key Points:

- **Fixed Time Frame:** Set deadlines for each deliverable.
- **Scope Management:** Adjust scope or quality if necessary to meet deadlines.
- **Quality Assurance:** Maintain a balance between scope, quality, and time.

Time Box	Task Name	Duration(Days)	Resource
TB1	Introduction	10	Analyst,User
	Initial Study	5	Analyst
	Literature Review	4	Analyst,Team Leader
TB2	Methodology	6	Analyst,Developer
	Planning	10	Analyst,Developer,Team Leader,Designer
	Feasibility	7	Analyst,Team Leader
TB3	Foundation	13	Analyst,Developer,Designer
TB4	Exploration	15	Analyst,Developer,Designer
	Engineering	35	Developer,Designer
TB5	Deployment	15	Analyst,Developer
	Testing	5	Analyst,Developer,Tester,User
TB6	Implementation	5	Analyst,Developer,Tester,User
TB7	Critical Appraisal and Evaluation	5	Analyst,Developer,Tester
	Lessons Learning	3	Analyst,Developer,User
TB8	Conclusion	2	Analyst

Table 3:List of the Time Boxes

5.1.4 Gantt chart:

A **Gantt Chart** is a visual project management tool that displays tasks, durations, and dependencies. It helps in scheduling and tracking project progress by showing a horizontal bar for each task, indicating start and end dates, as well as milestones.

Key Features:

- **Task Scheduling:** Display tasks and their duration.
- **Dependencies:** Show task dependencies and sequencing.
- **Milestones:** Highlight significant project milestones.

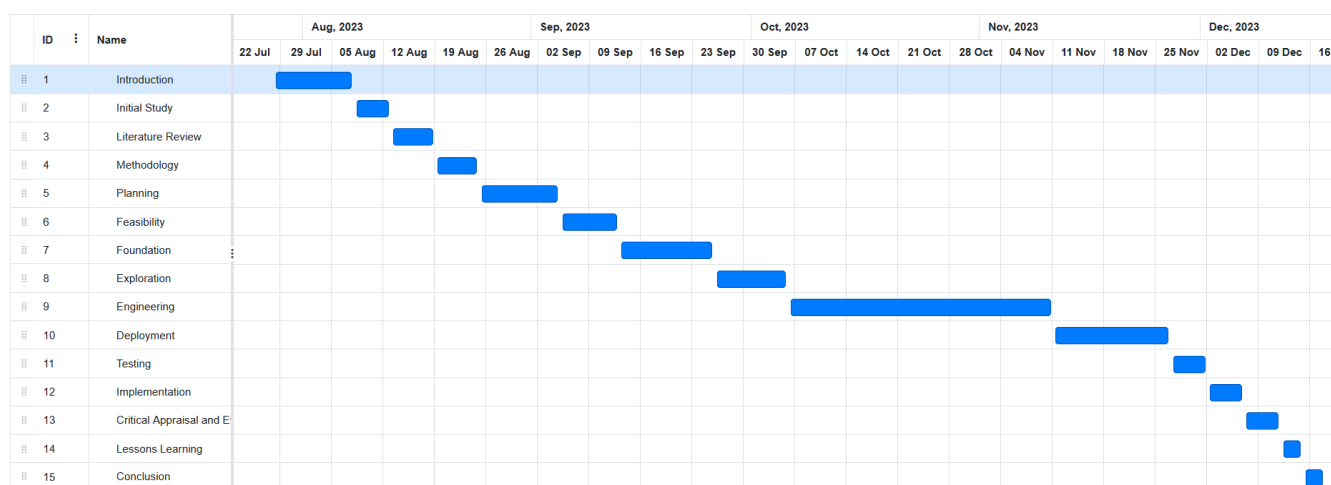


Figure 3: Gantt Chart

Chapter 6 – Feasibility

In assessing the feasibility of a project, it is crucial to evaluate various factors such as technical requirements, scheduling, legal constraints, and economic considerations. This analysis helps determine the likelihood of the project's successful completion and its potential to meet objectives effectively. The following sections discuss different types of feasibility, including operational feasibility, cost-benefit analysis, and the application of the Dynamic Systems Development Method (DSDM).

6.1 All Possible Types of Feasibility

6.1.1 Operational feasibility

Operational feasibility assesses how well a project aligns with the operational needs and requirements of its users. For the Smart Edu system, this includes:

- **User Accessibility:** The web-based application can be accessed from any location with an internet connection, making it highly flexible and convenient for users including students, teachers, and administrators.
- **Ease of Use:** The system is designed to be user-friendly, minimizing the learning curve and ensuring ease of adoption by all stakeholders.
- **Integration:** The system connects various departments and functions, facilitating seamless communication and interaction within the educational institution.
- **Cost Efficiency:** The system's operational costs are minimized as it requires only a computer and an internet connection, eliminating the need for extensive hardware.
- **Reduction of Repetitive Tasks:** By automating many repetitive tasks, the system enhances efficiency and reduces manual workload for users.

Overall Operational Benefits:

- **Accessibility:** Easy access from anywhere.
- **Affordability:** Low-cost implementation and maintenance.
- **Efficiency:** Streamlined processes and reduced manual effort.

6.2 Cost Benefit Analysis

A **Cost-Benefit Analysis (CBA)** evaluates the financial aspects of the project, weighing the costs against the anticipated benefits. The goal is to determine the most cost-effective approach while achieving the desired outcomes.

Key Components:

- **Cost Identification:** Includes costs related to development, implementation, maintenance, and training.
- **Benefit Identification:** Includes operational efficiencies, time savings, improved user satisfaction, and enhanced data management.
- **Analysis:** Compares the total costs to the expected benefits to determine the project's overall financial viability.

Example Cost-Benefit Analysis for Smart Edu Project:

- **Initial Costs:** Development, deployment, and training expenses.
- **Ongoing Costs:** Maintenance and support.
- **Benefits:** Increased operational efficiency, reduced manual tasks, improved data accuracy, and user satisfaction.
- **Return on Investment (ROI):** Calculated based on the net benefits compared to the total costs.

Decision Criteria:

- If the benefits outweigh the costs, the project is considered financially viable.
- If not, alternative solutions or adjustments may be required.

Project Name: SMART EDU

Total cost

	Year 1	Year 2	Year 3	Total
Web-Based Application	100000/=			100000/=
Domain & Hosting	15000/=	15000/=	15000/=	45000/=
Software	1000/=	1000/=	1000/=	3000/=
Internet	3000/=	3000/=	3000/=	9000/=
Training	5000/=	00	00/=	5000/=
Development	2000/=	3000/=	3000/=	8000/=
Maintenance	10000/=	10000/=	10000/=	30000/=
Total	136000/=	32000/=	32000/=	200000/=

Table 4: Cost Benefit

6.3 DSDM

The **Dynamic Systems Development Method (DSDM)** is a robust Agile methodology that covers the entire project lifecycle. It is suitable for managing projects of varying sizes and complexities.

Key Features of DSDM:

- **Vendor-Independence:** DSDM is not tied to any specific technology or vendor, making it adaptable to various environments.
- **Full Lifecycle Coverage:** It addresses all phases of the project, from initial planning to final delivery and maintenance.
- **Scalability:** Proven to handle projects ranging from small to large and across different business sectors.
- **Agile Foundation:** Emphasizes flexibility, iterative development, and continuous stakeholder engagement.

Advantages for Smart Edu Project:

- **Timely Delivery:** Ensures that the project is completed on schedule.
- **Budget Management:** Helps keep the project within budget.
- **Adaptability:** Allows for adjustments based on evolving requirements and stakeholder feedback.

Conclusion: DSDM is an ideal methodology for the Smart Edu project due to its flexibility, scalability, and focus on delivering value within the agreed timeframe and budget.

Chapter 7 Foundation

7.1 The Problem Area Identification

To create effective software, it is essential to identify the specific problem areas accurately. For the Smart Edu project, the issues are identified through:

- **Interviews:** Conducting discussions with stakeholders like students, teachers, and administrators to understand their needs.
- **Observation:** Observing the current manual processes to spot inefficiencies and challenges.
- **Questionnaires:** Collecting detailed requirements and feedback from users through structured questionnaires

7.1.1 Interview

The interview is a procedure where two people or more converse directly about a specific issue or potential solutions. It's one of the greatest methods for identifying requirements. It can happen face to face with student, teacher, staff etc.

7.1.2 Specific Problem Area Identification and Description

Challenges Identified:

- **Manual Record-Keeping:** Handling records manually is prone to errors and inefficiencies.
- **Time-Consuming Processes:** The manual compilation of results and managing of records takes considerable time.
- **Lack of Integration:** Difficulty in managing various aspects of school operations like attendance, grades, and schedules

7.2 Rich Picture

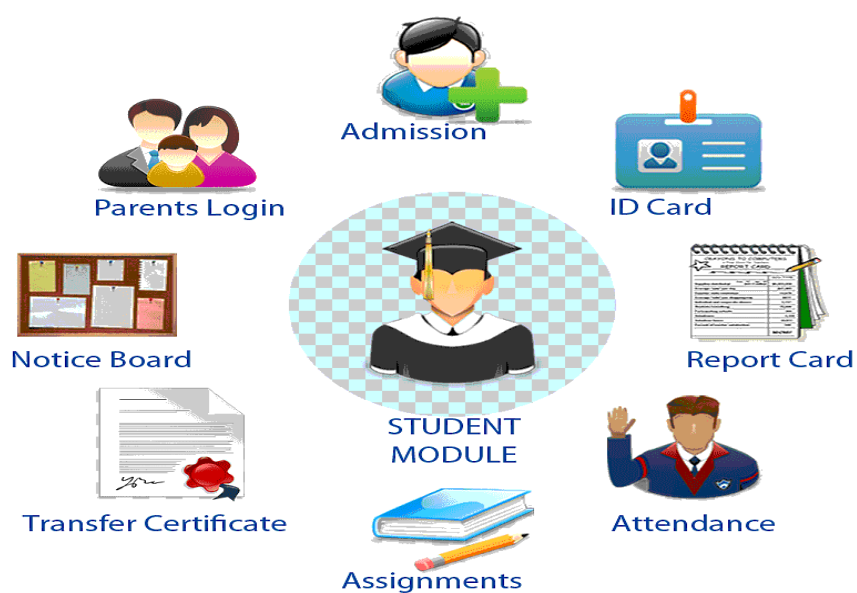


Figure 4 : Rich picture

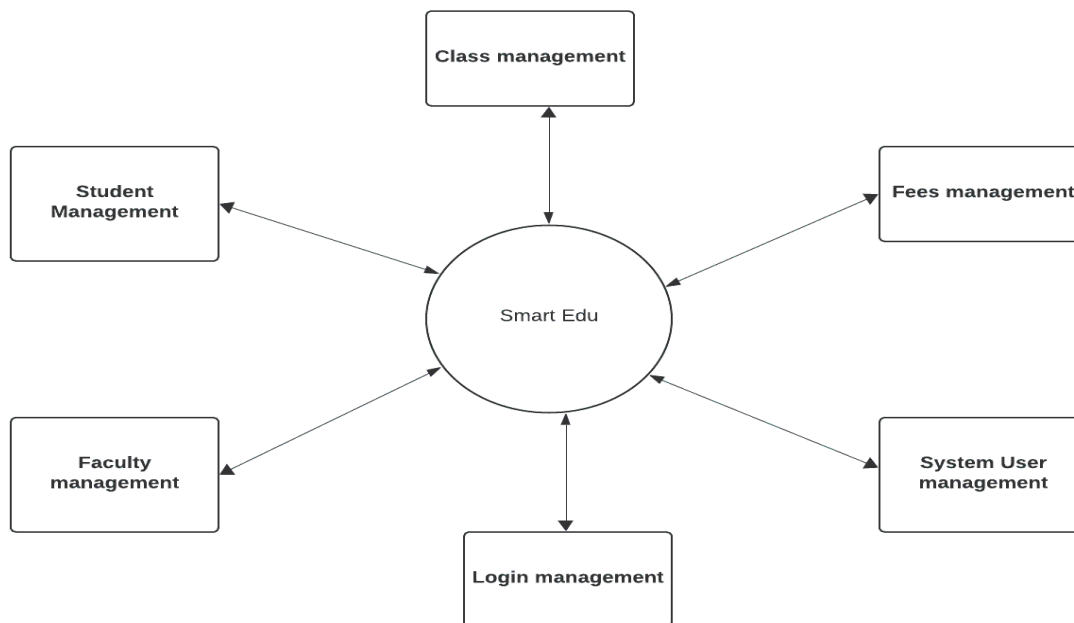


Figure 5: Simple data flow SIS

7.3 Specific problem area identification and description

Challenges Identified:

- **Manual Record-Keeping:** Handling records manually is prone to errors and inefficiencies.
- **Time-Consuming Processes:** The manual compilation of results and managing of records takes considerable time.

Lack of Integration: Difficulty in managing various aspects of school operations like attendance, grades, and schedules.

7.4 Possible Solution:

Proposed Solutions for Smart Edu:

- **Automated Record Management:** Implement Odoo's automated modules to manage student, faculty, and administrative records.
- **Integration of Modules:** Use Odoo's integrated modules for attendance, grading, and scheduling to streamline processes.
- **User-Friendly Interface:** Design a simple and intuitive interface to enhance user experience and reduce training time.

7.5 Overall Requirement List

7.5.1 Functional Requirements

Functional Requirements:

- **Student Management:** Adding, updating, and managing student records.
- **Faculty Management:** Managing teacher profiles and assignments.
- **Attendance Tracking:** Automated tracking of student and faculty attendance.
- **Grading System:** Management of exams, grades, and results.
- **Library Management:** Tracking library books and student checkouts.
- **Fee Management:** Processing and recording fee payments.

7.5.2 Non-Functional Requirements

Non-Functional Requirements:

- **Security:** Ensuring data protection and access control.
- **Usability:** Creating a user-friendly interface.
- **Performance:** Ensuring efficient system performance with minimal downtime.
- **Scalability:** Ability to handle growing numbers of users and records.

7.6 Which Technology to be implemented

Odoo Platform:

- **Odoo (Python, PostgreSQL):** Utilized for its modular approach to manage different school operations.
- **Benefits:**
 - **Modularity:** Allows customization and addition of new features as needed.
 - **Integration:** Seamlessly integrates various aspects of school management.
 - **Accessibility:** Web-based system accessible from any location.

Client-Server Application:

- **Architecture:** Utilizes a client-server model where the Odoo server handles data management and processing.
- **Benefits:**
 - **Centralized Data Management:** All data is stored on the server, ensuring consistency and security.
 - **Scalability:** Easy to scale by adding more servers or resources.

Web-Based Application:

- **Web Technology:** Accessible through web browsers, eliminating the need for extensive software installation.
- **Benefits:**
 - **Cross-Platform Compatibility:** Works across different devices and operating systems.
 - **Real-Time Updates:** Users always have access to the latest version of the system.

Chapter 8 Exploration

8.1 Old System Use Case:

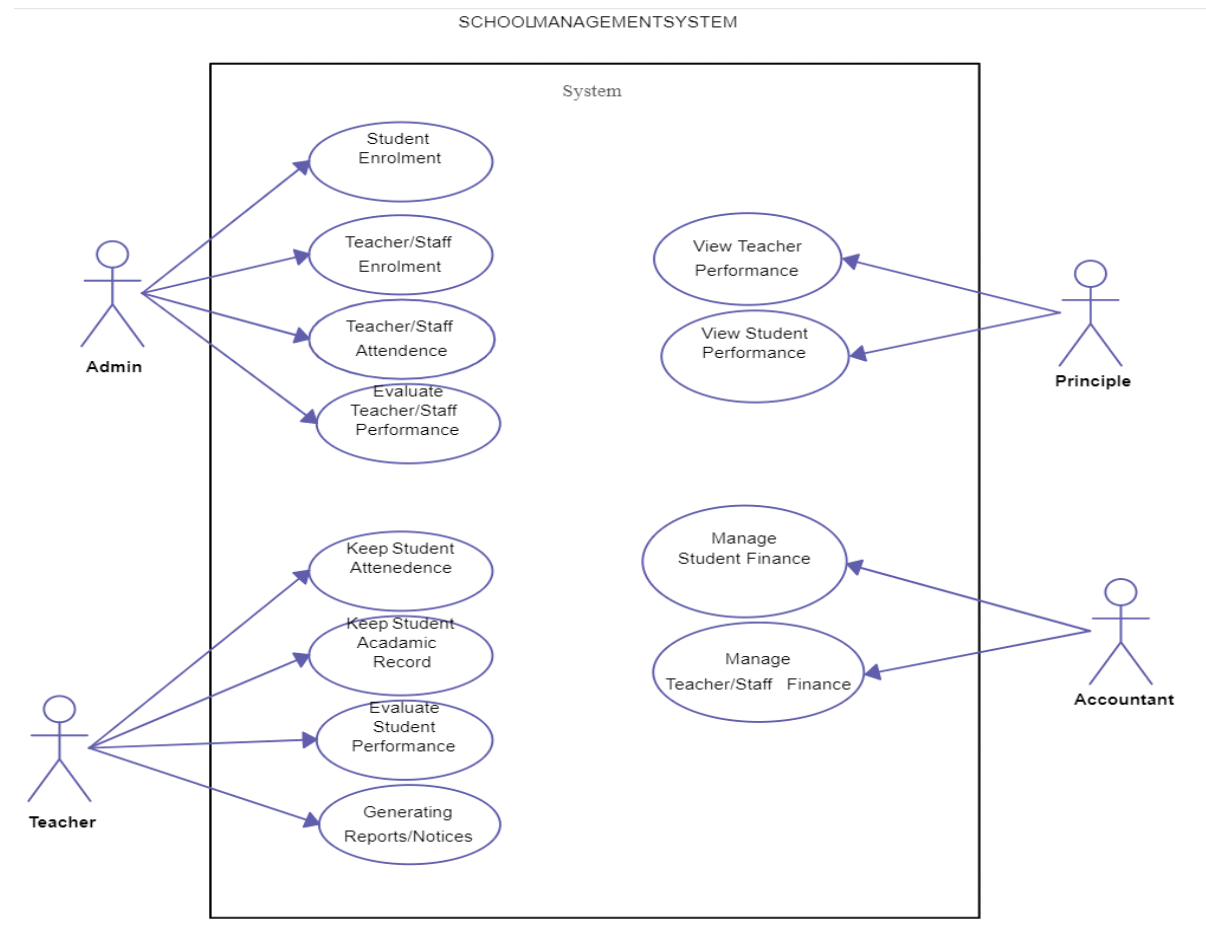


Figure 6:Old System Use Case

In this old school management system have missing many modules. such as: Student access, exam, notice, staff information store etc.

8.2 Activity Diagram:

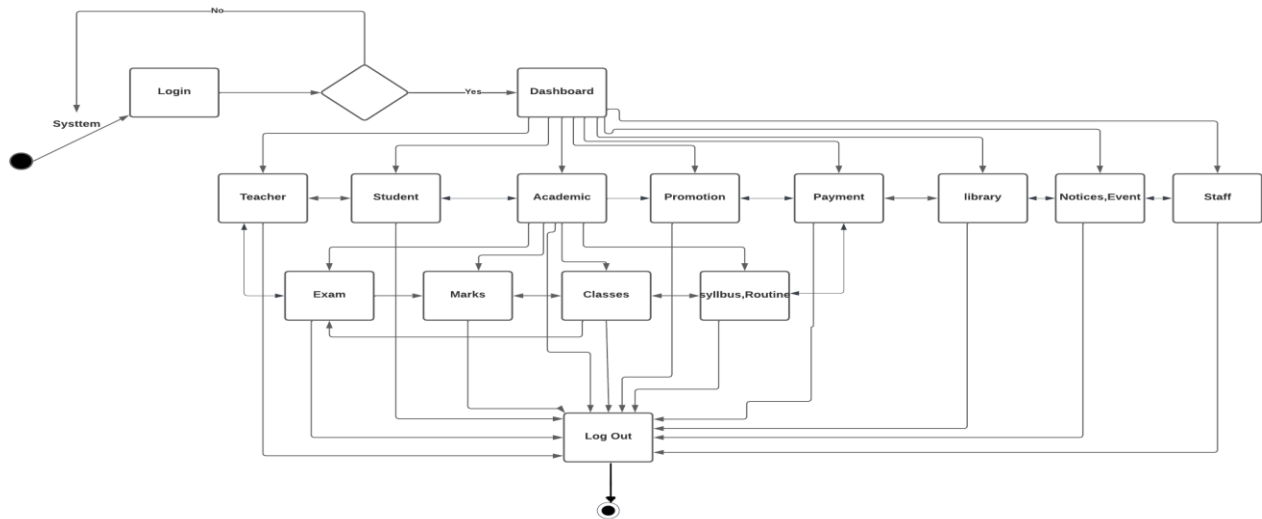


Figure 7:Activity Diagram

8.3 Full System Use Case Diagram:

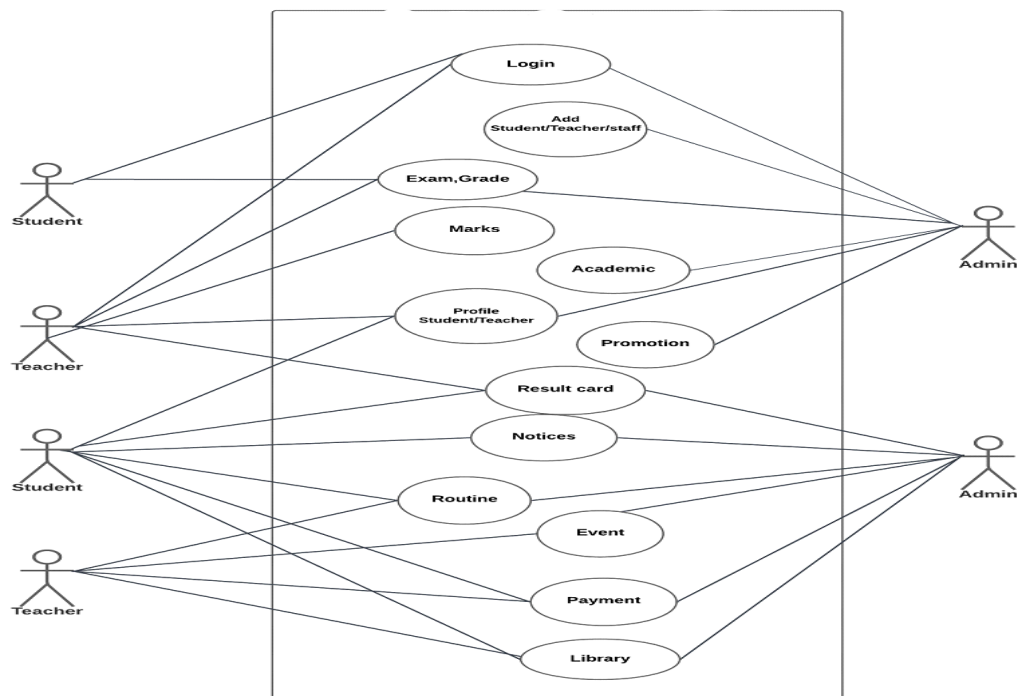


Figure 8:Full System Use Case

8.3.1 Breakdown of Use Case diagram according to the functions of DARUSSUNNAH SCHOOL MANAGEMENT system:

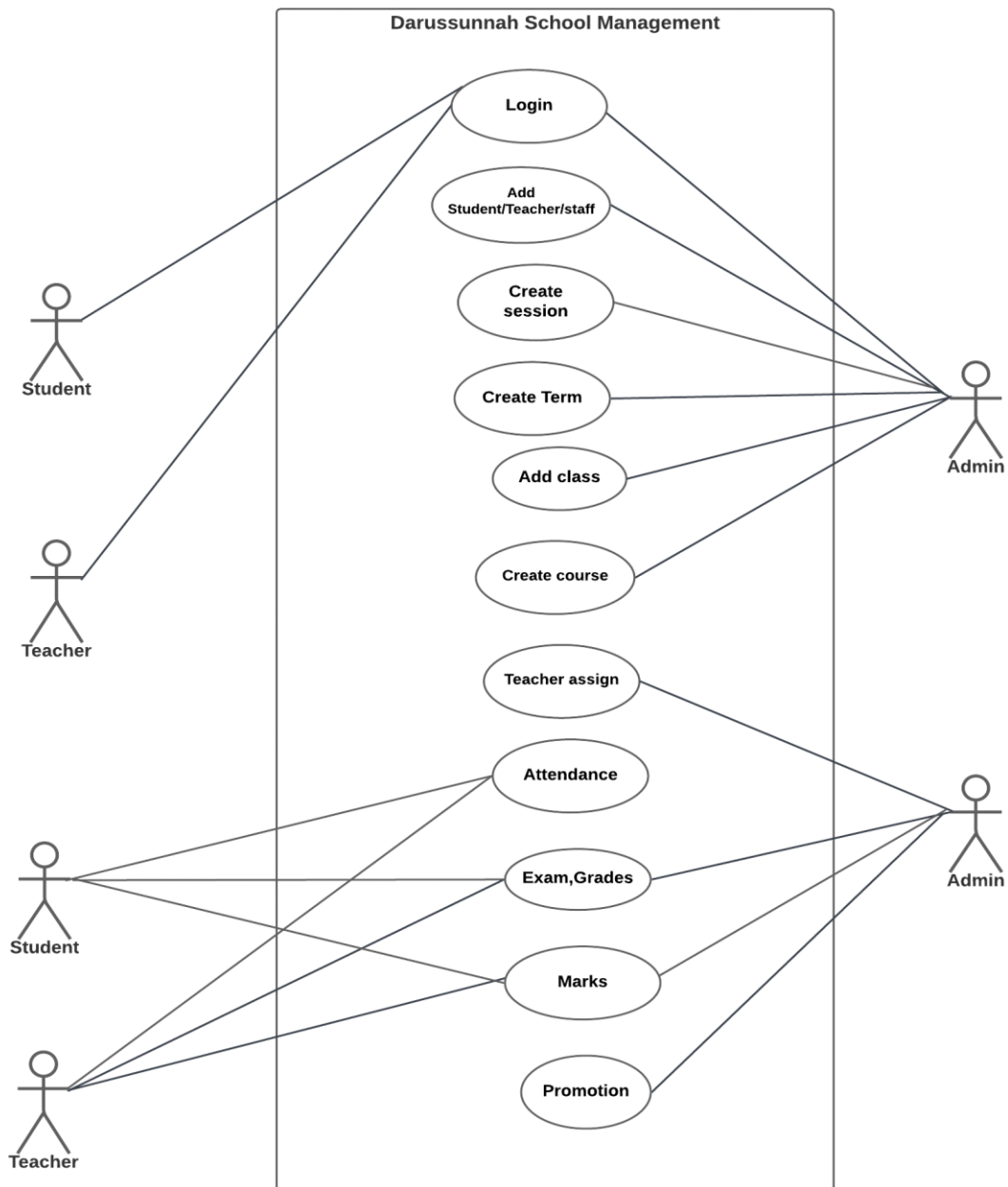


Figure 9: Use Case for Academic

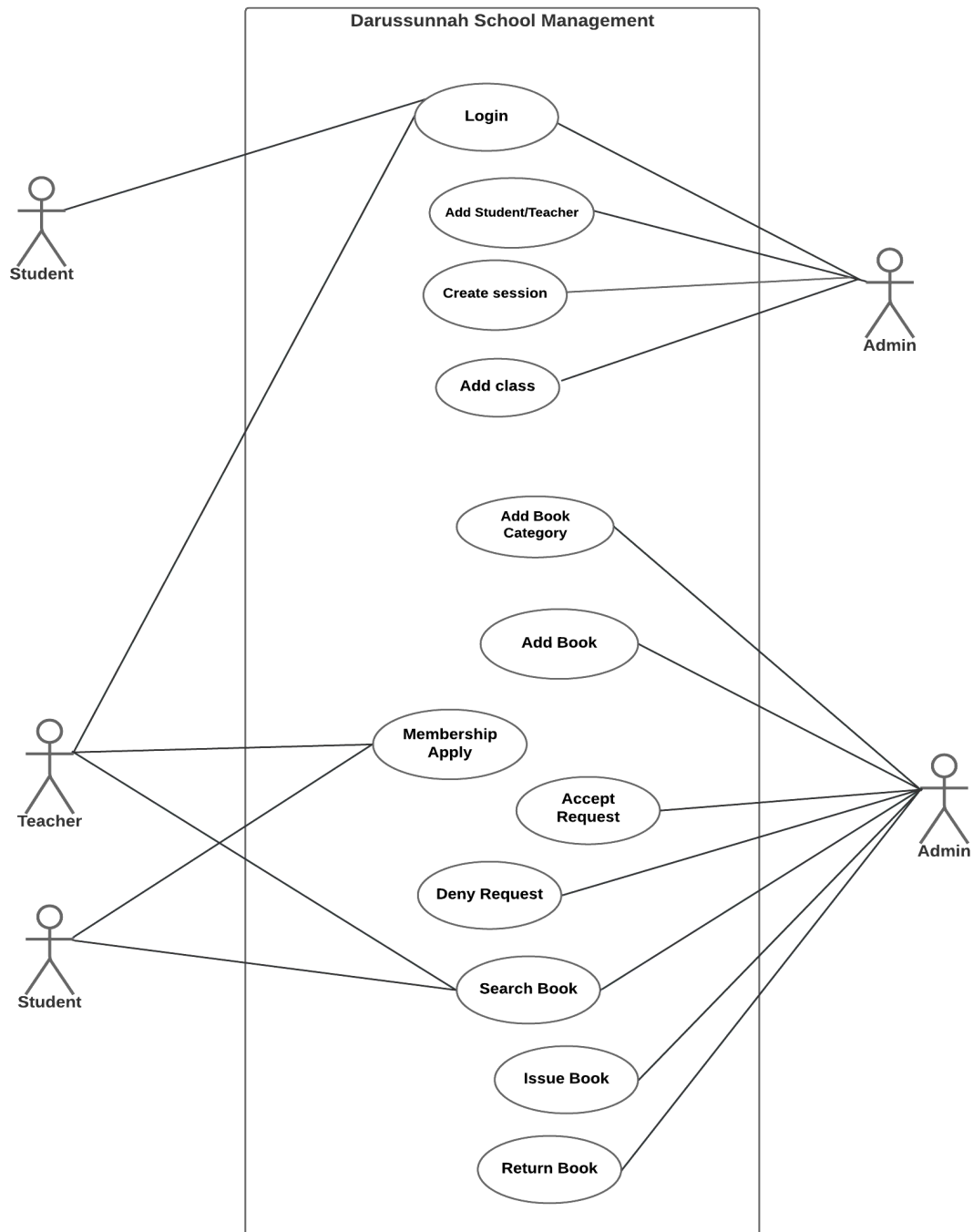


Figure 10: Use Case for Library

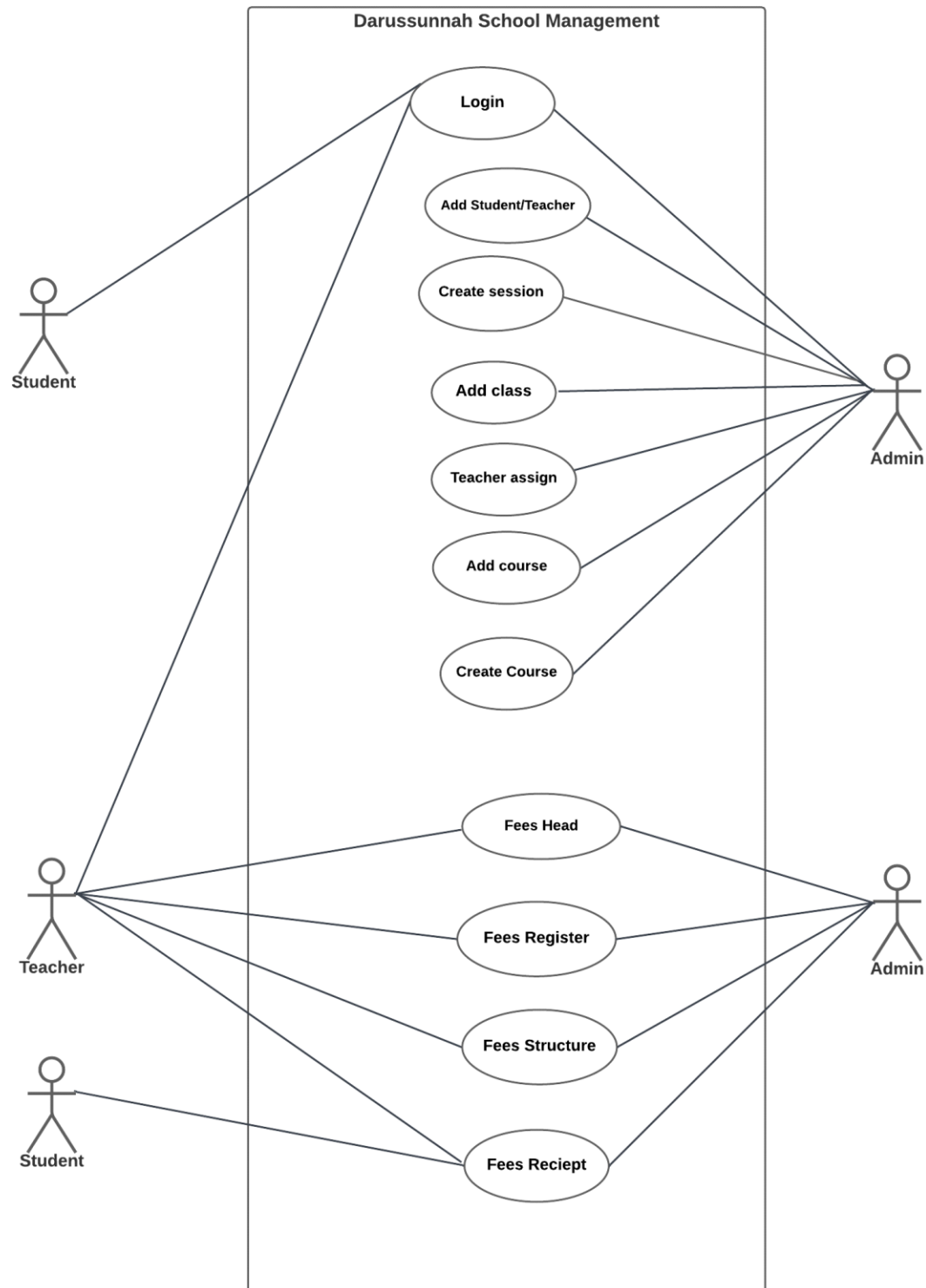


Figure 11: Use Case for Payment

8.4 Full System Activity Diagram:

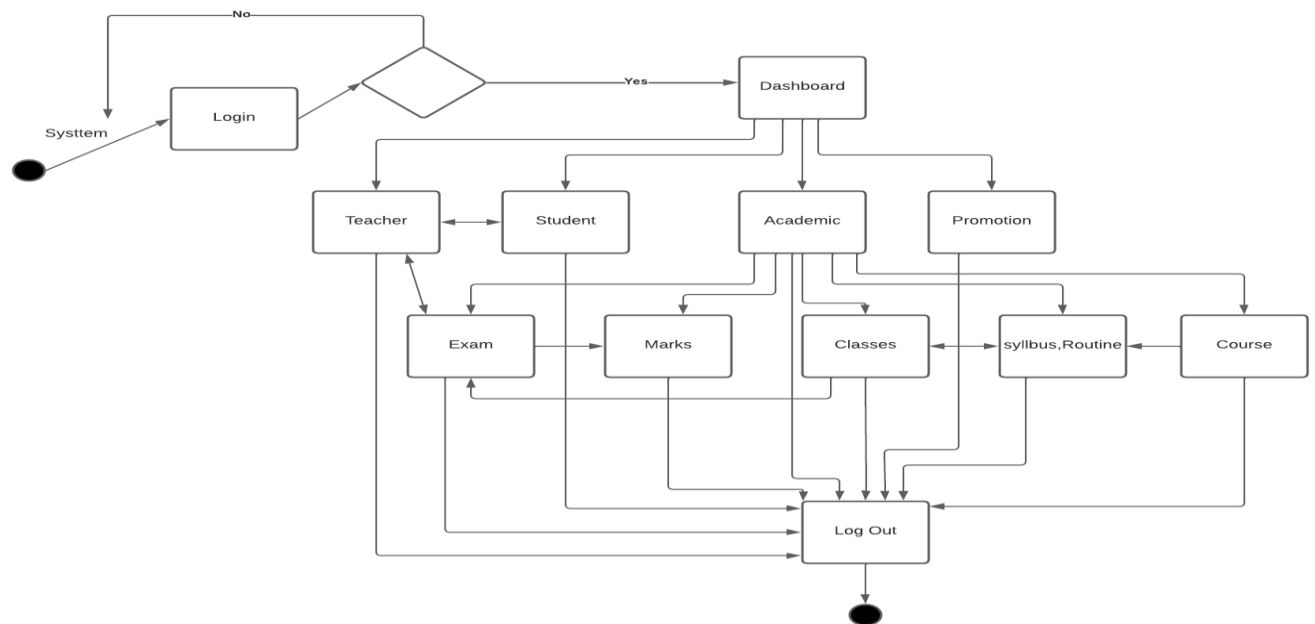


Figure 12:Activity diagram for Academic

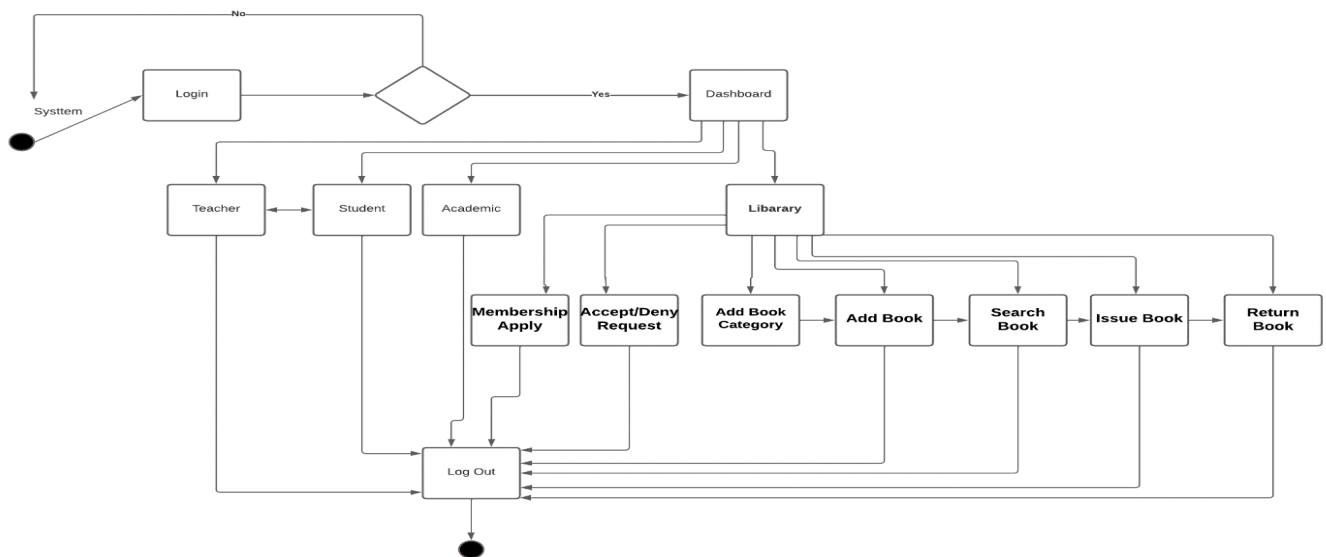


Figure 13:Activity diagram for Library

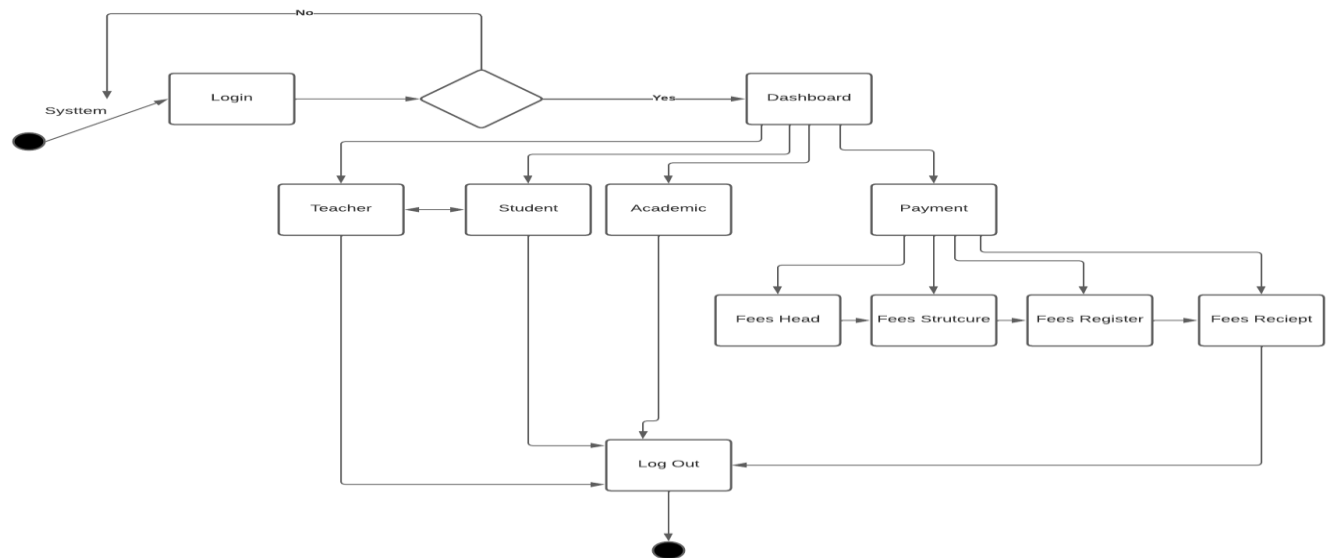


Figure 14:Activity diagram for Payment

8.5 Requirements Catalogue

Requirement catalog for the login system:

Requirement Type	Functional Requirement		
Requirement Name	Login and registration system for the admin		
Requirement Description	The person who will be the admin of this system should be registered at the time of the software installation. When the admin will be registered no one will be able to access the system except the admin with proper authentication.		
Requirement ID	Date Source	Priority	Sign off
M-01	Admin	Must	Admin
Requirement Type	Non-Functional Requirement		
Requirement Name	The attempt of login per day		
Description	Target Value	Acceptance Value	Comment
	1500	1000	Only the admin can log in.

8.6 Requirement Prioritized:

The MoSCow priority methodology is the most effective method for establishing a project's priority list. I'm going to apply this method to create a priority list of all the DCL School Management system requirements that have been discovered. Following is a list of each:

Requirement category	Requirement Name +Explanation
Must have	There must be an authentication method for the school management system. System will able to provide admin, teacher, and student portals. The system requires admin login and registration upon installation. This system will be able to create class, exam, result.
Should have	Project management functions should be simplified.
Could have	Organization's details could be added on the system.
Won't have	Since the school management system is a web-based PHP project, there won't be any installation steps required for teacher or student panel.

Table 5:Requirement Prioritized

8.7 Prototype of DCL school management system:

edu.darussunnah-bd.com/login

DCL-School Login

Login

E-Mail Address admin@ut.com

Password

☐ Remember Me

Login [Forgot Your Password?](#)

Figure 15:Login form Prototype

Admission Registers - EMS

disd.df.daffodil.family/web#action=1318&model=op.admission.register&view_type=list&cids=&menu_id=983

EMS Admissions Teachers Students Evaluations Signature Configuration Fees Configuration Dashboard

Admission Registers

Create

Name	Academic Year
☐ [Session-2025 2024-2025]	[EM,Dhanmondi 2024-2025]
☐ 2024-2025	[EM,Dhanmondi 2024-2025]
☐ Session 2024-25	[EM,Dhanmondi 2024-2025]
☐ 2023-2024	[Session-2024 2023-2024]

Settings

- Exam Management
- Exam Timetable
- Regular Exams
- Exams
- Exam Results
- Additional Exam
- Additional Exam Results
- Batchwise Exam Results
- All Additional Exam Results
- Fees Terms
- Fees
- Waiver
- Fees Head
- Fees Structure
- Academic Management
- Class
- Standards
- Medium
- Academic Years
- Division
- Grade
- Notice Board
- Subject Management
- Subjects
- Elective Subjects
- General Management

Group By ★ Favorites 1-4 / 4 < > [List] [Grid] [Map]

admission	Maximum No. of Admission	Status
1	350	Application Gathering
150	200	Done
50	200	Done
1	500	Done

Figure 16:Dashboard Interface Prototype

The screenshot shows a web browser window with the URL `disd.df.daffodil.family/web#id=3&action=1290&model=op.student&view_type=form&cids=&menu_id=983`. The page is titled "Students / New" and features a navigation bar with tabs: "Create Student User", "Dropout", and "Student Student". The main form is titled "Full Name" and includes the following sections:

- Student Information:** Fields for Student ID, Academic Year, Birth Date, Visa Info, Blood Group, Roll No., Gender, Nationality, Language, and Waiver EMS.
- Category:** Fields for Category, Mobile, and Email.
- Guardian's Mobile:** A field for the guardian's mobile number.
- Family Business:** A field for the family business.
- Family Income:** A field for the family income, currently set to 0.00.
- Address Detail:** A dropdown menu for Address Type.
- Permanent Address:** Fields for Street, Street2, City, State, Zip, and Country.
- Present Address:** Fields for Street, Street2, City, State, Zip, and Country.

Figure 17: Student add form Prototype

The screenshot shows a web browser window with the same URL as Figure 17. The page is titled "Students / Ilhaan Nazik Chowdhury" and features a navigation bar with tabs: "Dropout" and "Student Student". The main form is titled "Ilhaan Nazik Chowdhury" and includes the following sections:

- Student Information:** Fields for Student ID (11002135), Academic Year (Session-2024|2023-2024), Birth Date (02/03/2020), Visa Info, Blood Group, Roll No. (1), Gender (Male), Nationality (Bangladesh), Language (English (US)), and Waiver EMS.
- Category:** Fields for Category, Mobile (01707181819), and Email (vp.amdnr@dis.edu.bd).
- Guardian's Mobile:** A field for the guardian's mobile number (01707181819).
- Family Business:** A field for the family business (ilhaan.11002135@dis.edu.bd).
- Family Income:** A field for the family income, currently set to 0.00.
- Emergency Contact:** A field for the emergency contact (Mohsina shamin Nishat).
- Address Detail:** A dropdown menu for Address Type.

Figure 18: Student Profile Prototype

Figure 19:payment structure Prototype

Chapter 9 Engineering

9.1 Modules of the new system

The Smart Edu system is designed to streamline and manage the various facets of educational institutions. It encompasses several core modules, each serving a specific function to ensure smooth and efficient school management. Below are the key modules:

- **Administration Module**
 - **Overview:** Provides comprehensive control over all aspects of the system.
 - **Features:**
 - **User Management:** Add, update, and manage staff members, teachers, and students.
 - **Course Management:** Create and assign courses to teachers and students.
 - **Library Management:** Manage library books and categories, approve or deny student book requests.
 - **System Configuration:** Configure system settings, manage permissions, and oversee system operations.
- **Student Module**
 - **Overview:** Allows students to access and manage their personal information and academic records.

- **Features:**
 - **Profile Access:** View and update personal profile information.
 - **Grades and Results:** Access subject grades, overall results, and academic performance.
 - **Attendance Tracking:** View attendance records for each subject.
 - **Notices and Announcements:** Stay updated with school notices, events, and important news.
 - **Contact Information:** Access contact details of course teachers for inquiries about the syllabus or exams.
- **Teacher Module**
 - **Overview:** Facilitates teachers in managing their interactions with students and handling academic records.
 - **Features:**
 - **Grade Management:** Add or edit student grades for their own subjects.
 - **Exam Management:** Design and administer exams, set up exam schedules.
 - **Attendance Recording:** Track and record student attendance.
 - **Payment Management:** Oversee student fee payments, including creating and managing fee structures and receipts.
- **Payment Module**
 - **Overview:** Manages all financial transactions related to student fees and payments.
 - **Features:**
 - **Fee Structure:** Define and manage various fee categories and structures.
 - **Fee Receipts:** Generate and manage receipts for fee payments.
 - **Transaction Tracking:** Monitor and track all financial transactions related to student fees.
- **Library Module**
 - **Overview:** Manages library operations and resources.
 - **Features:**
 - **Book Management:** Add, update, and categorize books in the library.
 - **Request Handling:** Approve or deny student requests for borrowing books.
 - **Inventory Management:** Track book availability and manage inventory.
- **Examination Module**
 - **Overview:** Handles all aspects related to exams and assessments.
 - **Features:**

- **Exam Creation:** Design and schedule exams for various subjects.
- **Result Processing:** Process and record exam results.
- **Performance Analysis:** Analyze student performance and generate reports.

Each module is integrated to ensure seamless communication and data flow between different parts of the system, enhancing the overall efficiency and effectiveness of school management. The Smart Edu system leverages the flexibility and modularity of Odoo to address specific needs and provide a cohesive solution for educational institutions.

9.2 Use Case

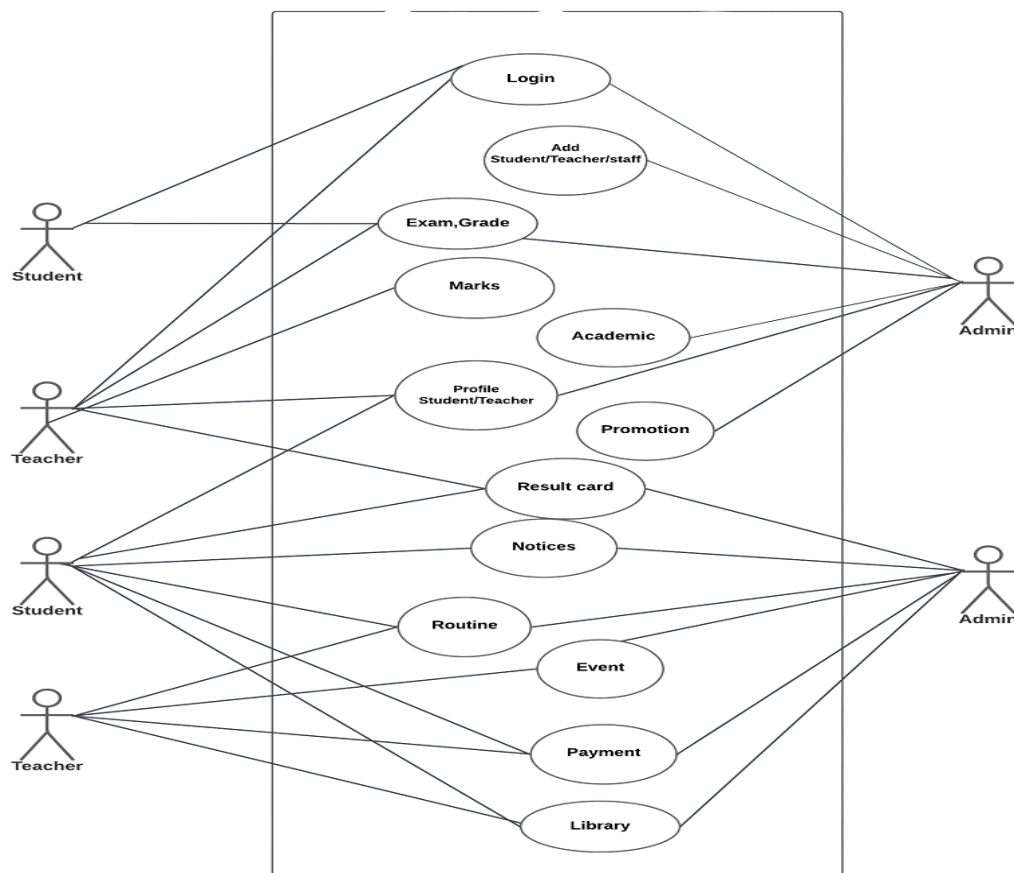


Figure 20: Use case diagram

9.3 ERD Diagram of School Management System

Figure 21: ERD diagram

9.4 Sequence Diagram of School Management System

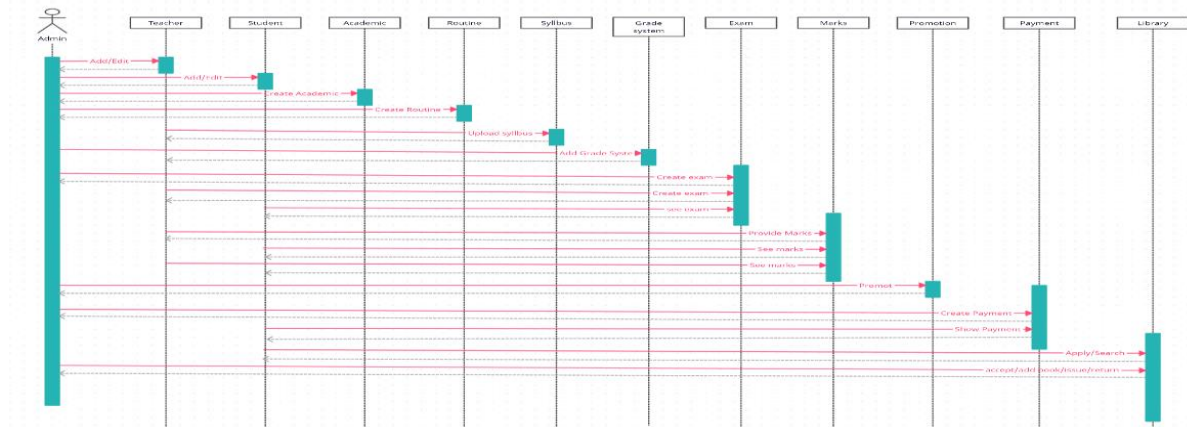


Figure 22: Sequence diagram

9.5 Component Diagram of School Management System

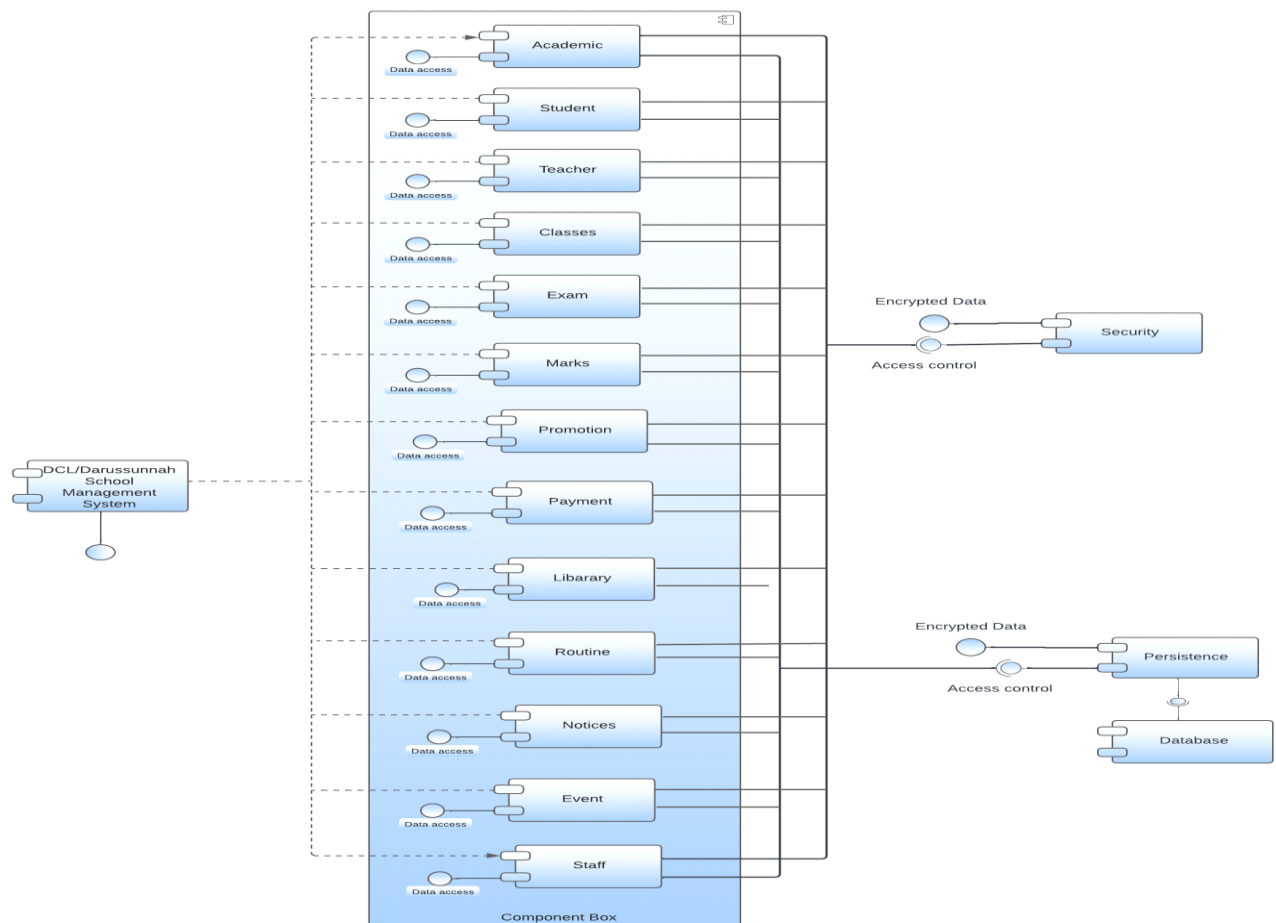


Figure 23: Component diagram

9.6 Deployment Diagram of School Management System

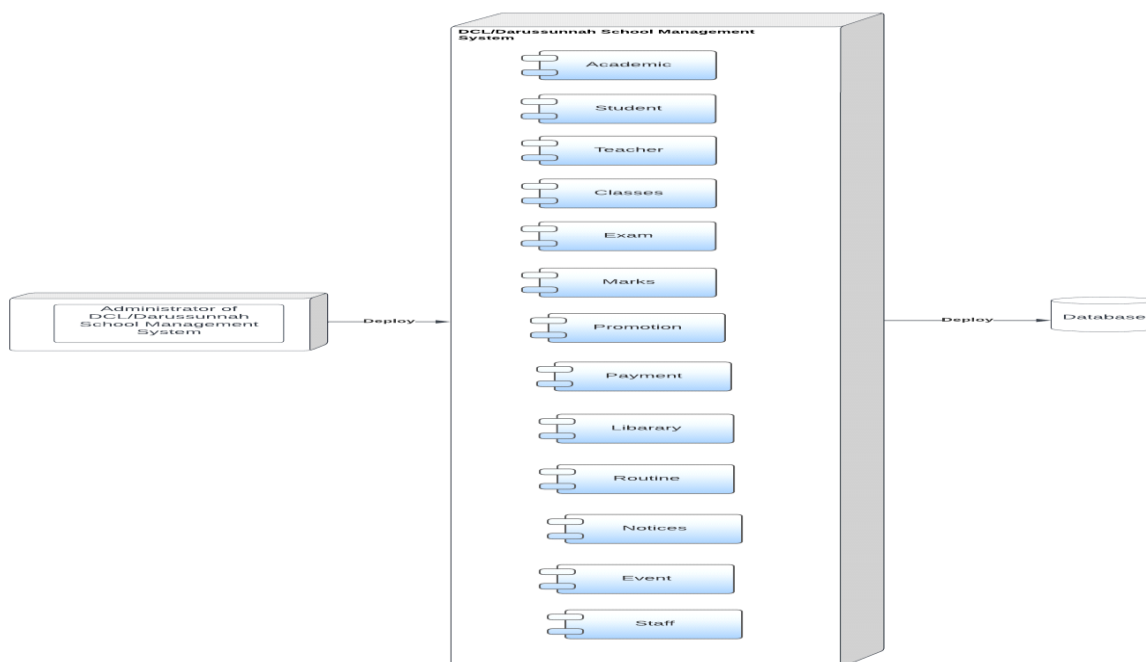


Figure 24:Deployment diagram

9.7 System Interface Design / Prototype of DCL/School Management System

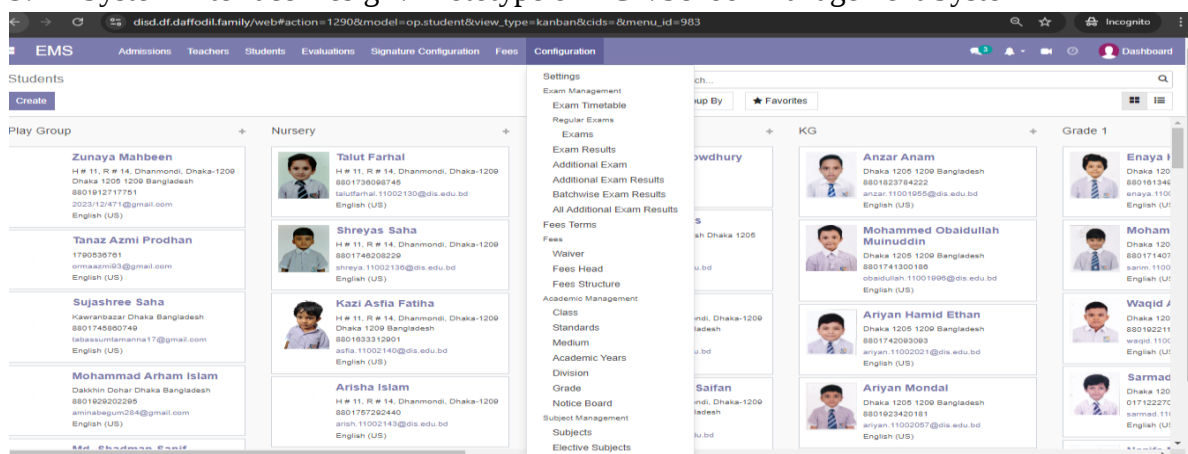


Figure 25:System Interface Design

Chapter 10 Deployment/Development

10.1 Core Module Coding Samples

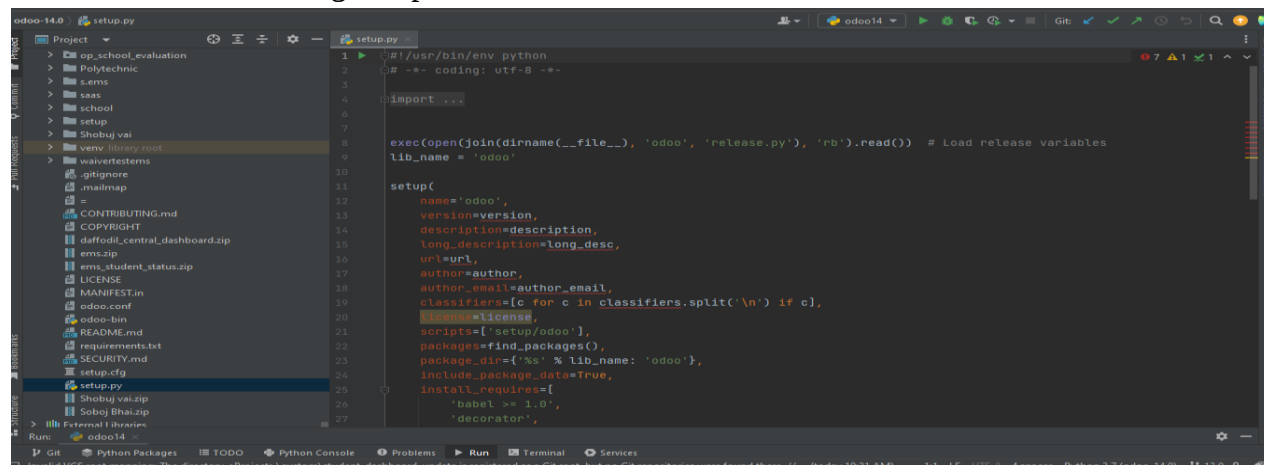


Figure 26: Database Connection code sample

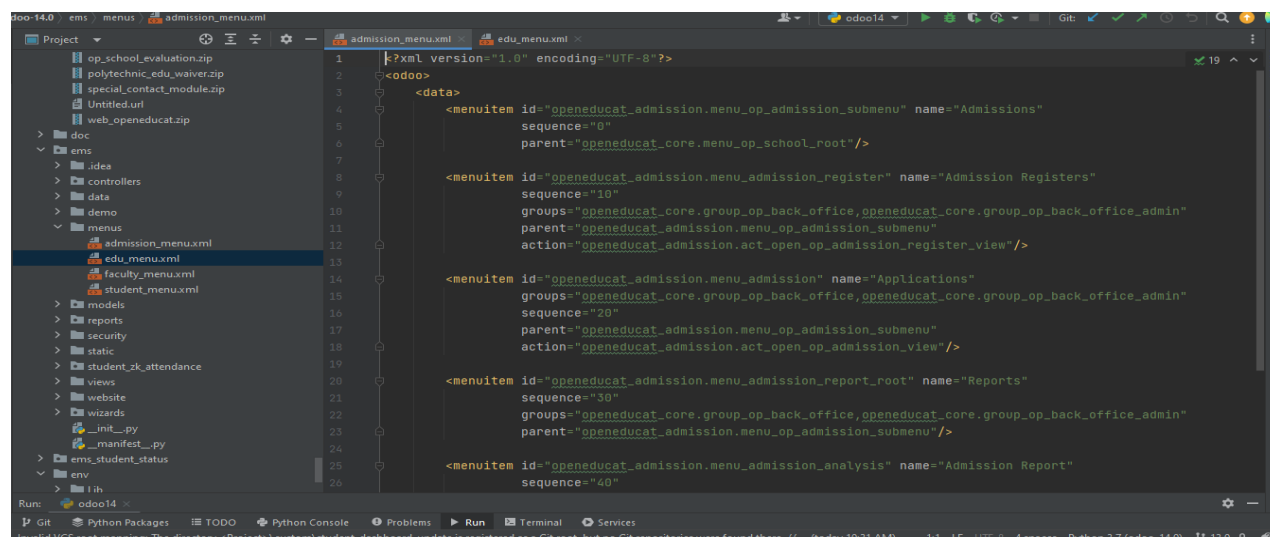


Figure 27: Dashboard Code sample


```

class NoticeBoard(models.Model):
    """Defining student news."""
    _name = 'op.notice_board'
    _description = 'Student Notice Board'

    subject = fields.Char('Notice Title', required=True)
    description_id = fields.Char('Description', help="Description")
    date = fields.Datetime('Date', help="Date of the notice")
    class_id = fields.Many2one('op.class', 'Class', track_visibility='onchange')

    general = fields.Boolean('General', help="Notice For All The Student")
    attachment = fields.Binary('File Attachment')
    title = fields.Char('Pre-header', help='Title of the email.')

    user_ids = fields.Many2many('res.users', 'user_news_rel', 'id', 'user_ids',
                                help='Name to whom this news is related.')
    color = fields.Integer('Color Index', default=0, help='Color index')

    def get_attachment_link(self):
        """Construct and return the attachment link."""
        base_url = '/web/content/op.notice/{}/attachment'.format(self.id)
        return base_url

    @api.onchange('class_id', 'general')

```

Figure 31: Notice code sample

```

class Exam(models.Model):
    """Defining model for Exam."""
    _name = "exam.exam"
    _description = "Exam Information"

    @api.constrains("start_date", "end_date")
    def check_date_exam(self):
        """Method to check constraint of exam start date and end date."""
        for rec in self:
            if rec.end_date < rec.start_date:
                raise ValidationError(
                    _("Exam end date should be greater than start date!")
                )

            for line in rec.exam_schedule_ids:
                if line.timeable_id:
                    for tt in line.timeable_id.exam_timetable_line_ids:
                        if not rec.start_date <= tt.exam_date <= rec.end_date:
                            raise ValidationError(
                                _("Invalid Exam Schedule!")
                            )
                    \n\nExam Dates must be in between Start date and End date !

```

Figure 32: Exam Create code sample

```

<?xml version="1.0" encoding="utf-8"?>
<template id="class_ix_x_report_card">
    <t-call="web.html_container">
        <t-foreach="docs" t-as="o">
            <t-call="web.basic_layout">
                <t-set="current_group_gpa" t-value=""/>
                <t-set="current_group_lg" t-value=""/>
                <t-set="current_group" t-value=""/>
                <div class="page">
                    <div class="row" style="padding: 0; margin: 0;">
                        <div class="col-12 d-flex justify-content-center">
                            <div class="row mb-2">
                                <div class="col-2">
                                    
                                <div class="col-8 text-center">
                                    <h3 style="margin-left: 70px;">
                                        <span t-field="res_company.name"/>
                                    </h3>
                                </div>
                            </div>
                        </div>
                        <div class="row mb-1" style="padding: 0; margin-top: -20px;">
                            <div class="col-12 text-center">

```

Figure 33: Result Card code sample

10.2 Possible problem breakdown

Managing a complex project effectively requires addressing potential issues that might arise during development. Here are some common problems and their respective solutions:

- **Redundant Data**
 - **Problem:** Redundant data can lead to inconsistencies and inefficiencies in the system.
 - **Solution:** Implement a well-structured data model with normalization techniques to avoid data duplication. Use unique constraints and data validation rules to maintain data integrity.
- **Normal Error Handling**
 - **Problem:** Failure to handle errors appropriately can lead to system crashes or unhandled exceptions.
 - **Solution:** Incorporate comprehensive error handling mechanisms, including try-catch blocks and logging to capture and manage errors gracefully.
- **Inadequate Requirement Analysis**
 - **Problem:** Insufficient understanding of requirements can result in a system that does not meet user needs.
 - **Solution:** Conduct thorough requirement analysis sessions with stakeholders, use interviews, questionnaires, and observations, and validate requirements with actual users to ensure they are well understood and documented.
- **Insufficient Testing**
 - **Problem:** Limited testing may result in undetected bugs and issues.
 - **Solution:** Implement rigorous testing protocols, including unit tests, integration tests, and user acceptance tests (UAT). Ensure that testing covers all functional and non-functional aspects of the system.
- **Insufficient Support from Users**
 - **Problem:** Lack of user support can hinder the adoption and effective use of the system.
 - **Solution:** Engage users early in the development process, provide training and support, and gather feedback through surveys to address their concerns and improve the system.

10.3 Prioritization while developing

To effectively manage and prioritize requirements, the MoSCoW method is employed. This method categorizes requirements into four priority levels:

- **Must Have (M)**
 - **Description:** Essential requirements that are critical for the system to function. Without these, the system cannot operate.
 - **Examples:**

- User authentication and authorization
- Core functionalities such as adding and managing students, teachers, and courses
- Basic data integrity and error handling
- **Should Have (S)**
 - **Description:** Important requirements that add significant value but are not critical for the initial launch. These can be implemented after the must-have requirements.
 - **Examples:**
 - Advanced reporting features
 - Detailed analytics and performance metrics
 - Enhanced user interfaces and user experience improvements
- **Could Have (C)**
 - **Description:** Desirable requirements that are not essential but could enhance the system. These can be included if time and resources permit.
 - **Examples:**
 - Integration with other school management tools
 - Additional customization options for users
 - Optional features like automated notifications and reminders
- **Won't Have (W)**
 - **Description:** Requirements that are not considered for the current development phase but may be revisited in future phases.
 - **Examples:**
 - Features not aligned with the current project goals
 - High-cost features that exceed budget constraints
 - Requirements that do not fit within the scope of the initial release

By using the MoSCoW method, the project team can ensure that the most critical and valuable features are developed and implemented first, while managing resources effectively and meeting project deadlines.

Prioritization	Requirement Name +Explanation
1	There must be an authentication method for the school management system. The system requires admin login and registration upon installation.

2	System will able to provide admin, teacher, and student portals.
3	This system will be able to create class, exam, result, payment, library.

Table 6: prioritization requirement

Chapter 11 – Testing

This is web based school management system. It's follow agile model in SDLC. For development i use this tools & programming languages (python, JavaScript, MySQL, html, css5). Mainly used odoo framework. It can store, process student, teachers, staff and library related all information. It have many features. Such as: payment, library, staff, student, admission, notice, exam and result etc.

Project Name	School Management System			
Name of Product	https://disd.df.daffodil.family			
Product Description	DCL School Management System			
Project Description	Web based Odoo framework and MySQL used school management			
	Project Type	Testing /Verification		
Project Duration	Start date	1/4/23	End date	31/12/23

Table 7: School Management details

11.1 Test Plan Acceptance

Testing is critical for ensuring the quality and functionality of the Smart Edu system. As an agile project, testing occurs iteratively with each development cycle.

Objectives of Testing:

- **Validation:** Ensure the system meets its requirements.
- **Efficiency:** Test the system with minimal resources and time.
- **Quality:** Produce high-quality test cases and problem reports.

Testing Approach:

- **Iterative Testing:** Each iteration involves planning, requirements analysis, design, coding, unit testing, and acceptance testing.
- **Documentation:** The test plan details the scope, strategy, and resources needed for testing.

11.2 Test Case

Functional Testing: Functional testing verifies that the system performs as expected based on specified requirements. It focuses on user interactions and system behavior without considering the underlying code.

- **Unit Testing:**
 - **Validation of Input Fields:** Test forms and input fields for correct data validation.
 - **Functionality Checks:** Ensure individual features work as intended.
- **Module Testing:**
 - **Form Submissions:** Test scenarios such as submitting forms with valid and invalid data.
 - **User Actions:** Test core functionalities like login, data entry, and report generation.
- **Integration Testing:**
 - **Data Flow:** Verify that different modules interact correctly (e.g., student data integration with grading and attendance).

Test Case Name	Unit/Module/ Integration Test		
Test Class			
Test Description			
Source of Data	Test Steps	Expected Result	Actual Result

Table 8:Test case

11.3 Unit Test

Unit Test Case- 1:

Test Case Name	Unit Test		
Test Class	Payment=>Fees Head create controller		
Test Description	Validation of a taka while creating a Payment		
Source of Data	Test Steps	Expected Result	Actual Result
Admin	1. Complete all fields on the form except the required 2. Fill out the form and submit it.	This field is required error message will appear while trying to create Fees Head.	A message appears that the field is required for creating the Fees Head.

Table 9:Unit Test Case- 1

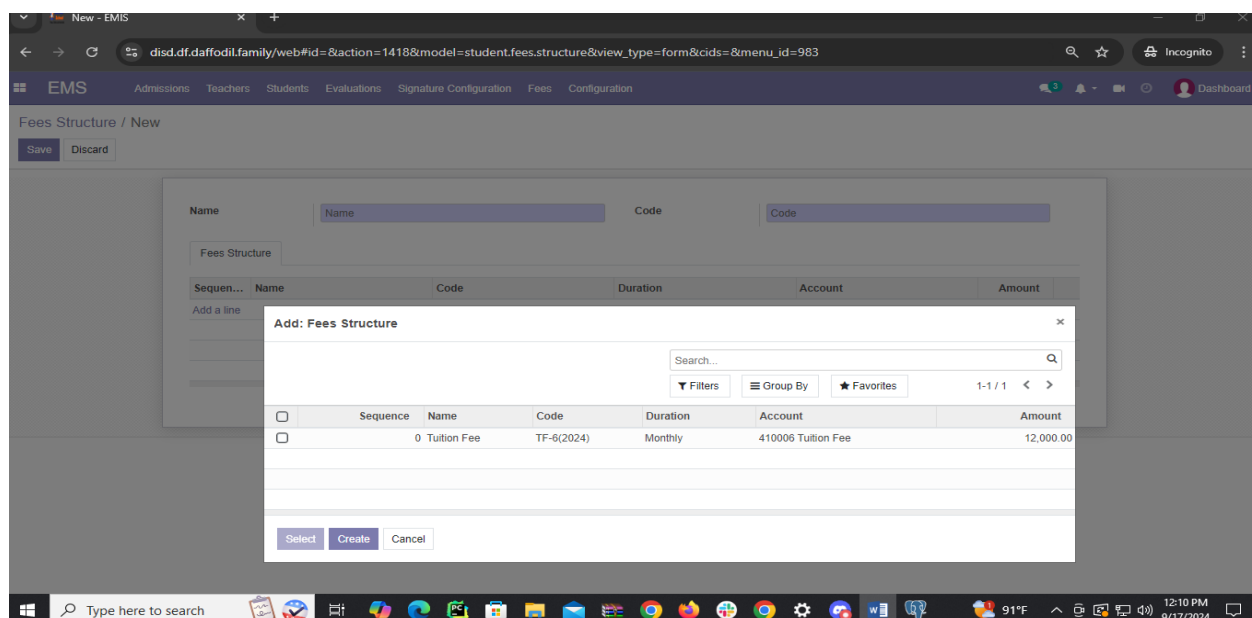


Figure 34:Unit Test-1

Unit Test Case - 2:

Test Case Name	Unit Test		
Test Class	Payment		
Test Description	Fees head use in Fees structure fees register and fees receipt while creating a Payment		
Source of Data	Test Steps	Expected Result	Actual Result
Admin	1. after added fees head then click fees structure option 2. Fees structure can use all type of fees head with all information.	After adding all information on fees head ,then all fees head must be usable on Fees structure fees register and fees receipt while creating a Payment	Successful accessible and usable all fees head on Fees structure fees register and fees receipt while creating a Payment and show all data on page.

Table 10:Unit test Case – 2

Figure 35:Unit Test-2

Fees head data usable vai Fees structure on Fees Receipt page::

Figure 36:Unit Test-2.1

11.4 Module Testing

Module Test Case-1:

Test Case Name	Module Test		
Test Class	The controller of Payment creation.		
Test Description	Make a payment attempt without entering any information.		
Source of Data	Test Steps	Expected Result	Actual Result
Admin	1. Log in to the system(admin) 2. Go to the payment => fees head or if fees head inserted then go to Head Structure and click add record. 3. click on the save option without data.	This field is required alert will be appeared for the required field.	Alert appears that the field is required for creating on the Fees Head or others option.

Table 11:Module test Case

The screenshot shows the 'Fees Head / New' form in the EMS system. The form has a header with navigation links: Admissions, Teachers, Students, Evaluations, Signature Configuration, Fees, and Configuration. Below the header, there are 'Save' and 'Discard' buttons. The form fields include:

- Name**: A text input field.
- Code**: A text input field.
- Eligible for Waiver**: A checkbox.
- Calculations**: A section containing:
 - Duration**: A dropdown menu.
 - Account**: A dropdown menu.
 - Currency**: A dropdown menu with 'BDT' selected.
- Amount**: A text input field with '0.00'.
- Company**: A dropdown menu with 'Daffodil International School' selected.
- Symbol**: A text input field.
- Function**: A button.

Figure 37:Module Test-1

11.5 Integration Testing

Integration Test Case - 1

Test Case Name	Integration Test		
Test Class	The controller of Authentication.		
Test Description	Login with valid data and dashboard redirect.		
Source of Data	Test Steps	Expected Result	Actual Result
Admin	1. Just go to the URL https://edu.darussunnah-bd.com/ 2. Enter correct usernames and password. 3. Click on the Login button.	If the credential is correct it will redirect to the dashboard.	If the credential is correct it will redirect to the dashboard.

Table 12:Integration test Case-1

Figure 38: Integration Test-1

Integration Test Case -2

Test Case Name	Integration Test		
Test Class	Student ,Teacher or Staff information controller		
Test Description	Student ,Teacher or Staff information adds and edits the information.		
Source of Data	Test Steps	Expected Result	Actual Result
Admin	1. Navigate to the page called My Preferences. 2. Click on the edit option. 3. Change the required information	All information will be update successfully.	All information was updated successfully.

Table 13: Integration test Case-2

The screenshot shows the 'Students' page in the EMS system. The interface includes a search bar, filters, and a grid of student profiles. The profiles are organized by grade group: Play Group, Nursery, Pre- KG, KG, and Grade 1. Each profile displays the student's name, photo, and contact information.

Play Group	Nursery	Pre- KG	KG	Grade 1
Zunaya Mahbeen H # 11, R # 14, Dhanmondi, Dhaka-1209 Dhaka 1205 1209 Bangladesh 880191217751 2023121471@gmail.com English (US)	Talut Farhal H # 11, R # 14, Dhanmondi, Dhaka-1209 8801736006745 talutfarhal.11002130@dis.edu.bd English (US)	Ilhaan Nazik Chowdhury 01707181019 vp.emdhn@dis.edu.bd English (US)	Anzar Anam Dhaka 1205 1209 Bangladesh 8801823784222 anzar.110016955@dis.edu.bd English (US)	Enaya Ha Dhaka 1205 12 880161349310 enaya.1100181 English (US)
Tanaz Azmi Proshan 1700536761 omtaazmi693@gmail.com English (US)	Shreyas Saha H # 11, R # 14, Dhanmondi, Dhaka-1209 8801746208229 shreyas.11002136@dis.edu.bd English (US)	Mohammed Anas Dhaka 14 (80) Bangladesh Dhaka 1205 1209 Bangladesh 8801714306392 anas.11002002@dis.edu.bd English (US)	Mohammed Obaidullah Muinuddin Dhaka 1205 1209 Bangladesh 8801741300186 obaidullah.11001995@dis.edu.bd English (US)	Mohamm Dhaka 1205 12 880171407234 sarim.1100181 English (US)
Sujashree Saha Kawranbazar Dhaka Bangladesh 8801745800749 tabassumtamanna17@gmail.com English (US)	Kazi Asfia Fatiha H # 11, R # 14, Dhanmondi, Dhaka-1209 Dhaka 1209 Bangladesh 8801633312001 asfia.11002140@dis.edu.bd English (US)	Ayaz Abrar Dhaka 14 (80) Bangladesh, Dhaka-1209 Dhaka 1205 1209 Bangladesh 01708364165 ayaz.11002019@dis.edu.bd English (US)	Ariyan Hamid Ethan Dhaka 1205 1209 Bangladesh 8801742093093 ariyan.11002021@dis.edu.bd English (US)	Waqid Azf Dhaka 1205 12 880192211712 waqid.1100186 English (US)
Mohammad Arham Islam Dakshin Dohar Dhaka Bangladesh 8801926202295 aminabegum264@gmail.com English (US)	Arisha Islam H # 11, R # 14, Dhanmondi, Dhaka-1209 8801757292440 arish.11002143@dis.edu.bd English (US)	Wassi Bin Jafar Saifan H # 11, R # 14, Dhanmondi, Dhaka-1209 Dhaka 1205 1209 Bangladesh 8801625036166 wass.11002115@dis.edu.bd English (US)	Ariyan Mondal Dhaka 1205 1209 Bangladesh 8801923420181 ariyan.11002057@dis.edu.bd English (US)	Sarmad R Dhaka 1205 12 01712227029 sarmad.110011 English (US)

Figure 39: Integration Test-2

The screenshot shows the 'Students / Talut Farhal' form in the EMS system. The form includes fields for student information, class information, guardians information, library, fees collection details, previous educational info, and remarks. The student's name is 'Talut Farhal'.

Student Information	Class Information	Guardians Information	Library	Fees Collection Details	Previous Educational Info	Remarks						
Student ID 0011002130 Academic Year [Session-2024 2023-2024] Birth Date 07/16/2019 Visa Info Blood Group A+ve	Roll No. 1 Gender Male Nationality Bangladesh Language English (US)	Waiver EMS <table border="1"> <thead> <tr> <th>Name</th> <th>Percentage</th> <th>Amount</th> </tr> </thead> <tbody> <tr> <td colspan="3">Add a line</td> </tr> </tbody> </table>	Name	Percentage	Amount	Add a line						
Name	Percentage	Amount										
Add a line												

Figure 40: Integration Test-3

Figure 41: Integration Test-4

School management system is web based project. Already I have complete testing phase.in below I provide this project testing report table with detailed:

Test Scenario ID	Test Scenario Description	Test case ID	Test Case Description	Test steps	Actual result == Expected result Test data with result	status	Executed by
TS_SM_001	Verify Login system	TS_SM_Login_001	Enter valid email & password	1.Enter valid email 2. Enter valid password	1.valid data input (successfully login & show	pass	Tester_DEEN)

				3.click on login button	dashbo ard) 1.invali d data input (incorr ect email or passwo rd)		
TS_S M_001	Auth system	TS_SM_Login_00 2	Click forget password	1.update password	Succes sfully update	pas s	Tester_ DEEN)
TS_S M_002	Add Stude nt/ teache r/ staff	TS_SM_Profile_c reate_001	New Student/Tea cher/Staff profile created	1.Add Student, teacher, staff	Succes sfully create profile	pas s	Tester_ DEEN)
TS_S M_002	Edit Profile	TS_SM_Profile_ UPDATE_001	Edit profile	1.Admin can be edit profile	Succes sfully edit profile	pas s	Tester_ DEEN)
TS_S M_003	Payme nt system	TS_SM_Payment 001	Payment with (message + due tk)	1.Create fees head 2.create fees structure 3.create fees register 4.create receipt 5.show in student portal	Succes sfully Payme nt with (messa ge + due tk+ show in student portal)	pas s	Tester_ DEEN)
	exam					pas s	Tester_ DEEN)

TS_S M_004		TS_SM_exam_001	Exam Create	1.Teacher /admin create exam	Succes sfully create exam		
TS_S M_005	routin e	TS_SM_routine_001	Create Routine	1. Admin can be include	Succes sfully Routin e added	pas s	Tester_ DEEN)
TS_S M_006	Mark sheet	TS_SM_ Mark_sheet_001	Mark sheet create automatic	1.subject wise marks provide 2.marksh eet create automatic ally	Succes sfully Marksh eet Create (for individ ual student s)	pas s	Tester_ DEEN)
TS_S M_007	Notice	TS_SM_Notice_001	Create notice	1.admin create notice	Succes sfully create notice for every dashbo ard	pas s	Tester_ DEEN)
....							

Table 14: Test Details -1

Test Scenario ID	Functions	Test Case Description	Test case ID	% TCs Executed	% TCs Passed	TCs pending	Priority	Remarks
TS_SM_001	Login	Secure Login+ Registration	TS_SM_Login_001	100 %	100 %	0	High	
TS_SM_001	Auth system	Verify / forget password	TS_SM_Login_002					
TS_SM_002	Add Student/teacher/staff	New Student/Teacher/Staff is created successfully	TS_SM_Profile_create_001	100 %	100 %	0	High	
TS_SM_002	Edit Profile	Admin can be edit profile	TS_SM_Profile_create_002	100 %	100 %	0	High	
TS_SM_003	Payment system	Payment with (message + due tk)	TS_SM_Payment001	100 %	100 %	0	High	
TS_SM_004	exam	Admin & teacher can created exam successfully	TS_SM_exam_001	100 %	100 %	0	High	
TS_SM_005	routine	Admin can be include	TS_SM_routine_001	100 %	100 %	0	High	
....								

Table 14: Test Details -2

Non-functional testing (Goal: Testing product characteristics) –

TEST**PERFORMANCE**

- Performance Testing: Quick response from this project
- Load Testing: maximum operating and browser capabilities on this web
- Failover Testing: system is capable of allocating more resources and backing up all data.
- **Compatibility Testing** capable of working on a variety of hardware, operating systems, applications, networks, or mobile devices.

Test Case #	Test Case	Domain
1	For applications with 1000 concurrent users, the load time should not exceed 5 seconds.	Performance Testing
2	On both Windows and Mac operating systems, as well as in all browsers, this system should be accessible.	Compatibility Testing
3	Alt tags must to be present on each and every online picture.	Accessibility testing.

Table 15:Domain based test

The overall summary of testing activities is included in this section. The data presented here includes

Test cases executed: 50

Test cases pass: 50

Test cases fail :0

Pass: 100%

Fail: 0%

Note: developed by user requirements and testing phase pass.

Defect

Defect is one of the test report's most crucial details. The following information should be in the report.

Total number of bugs: 10

Status of bugs (open, closed, responding):10

Number of bugs :

Open:10

Resolved:10

Closed:10

Breakdown by severity and priority: solve

Test Report Of School Management System

Test		
Test cycle	System Test	
EXECUTED	PASSED	50
	FAILED	0
	(Total)TESTE EXECUTED (PASSED+ FAILED)	
PENDING		0
IN PROGRESS		0
BLOCKED		0
(SUB-TOTAL) TEST PASSED		0
PENDING+ IN PROGRESS+ BLOCKED+ TESTE EXECUTED		50

Table 16:Final test report

Chapter 12 – Implementation

12.1 Training

To ensure a smooth transition to the new Odoo-based Smart Edu system, comprehensive training is essential. Despite Odoo's user-friendly interface, school staff may need support due to varying.

- **Training Sessions:** Conduct hands-on training sessions for school representatives. These sessions will cover system functionalities, including student and staff management, course setup, and exam handling.
- **Training Materials:** Provide user manuals, quick reference guides, and video tutorials to facilitate self-learning.
- **Support:** Offer post-training support to address any issues or questions that arise as users start working with the system.
- **Train-the-Trainer:** Train a group of key users within the school who will, in turn, train other staff members.

User	Training	Time	Comment
Admin	• User group setup(student/teacher) • Create class • Create section • Create course • Assign teacher • Create exam • Routine setup	5 hours	Admin is familiar with all of the functions and operators.
Teacher	• Create exam • Marks upload • Fees Payment • Syllabus upload • Attendance • Holiday setup	4 hours	
	• Student		

Table 17:training based

12.2 Big Bang

The Big Bang approach will be used for migrating data from the old system to Odoo Smart Edu. This involves:

- **Data Extraction:** Extract data from the existing systems, including student records, staff information, and academic data.
- **Data Transformation:** Transform the data into a format compatible with Odoo. This includes mapping data fields and ensuring data integrity.
- **Data Import:** Import the transformed data into Odoo. This is performed in one go to minimize disruptions.

Validation: After import, validate the data to ensure accuracy and completeness. Address any discrepancies or issues that arise.

12.3 Scaling

Scaling in Odoo is handled by configuring the system to handle increasing loads. This involves:

- **Resource Allocation:** Adjust server resources based on user load and data volume. This might include adding more CPU, memory, or storage.
- **Performance Tuning:** Optimize Odoo's performance through database indexing, caching strategies, and code optimizations.
- **Testing:** Perform load testing to ensure the system can handle expected user volumes and data loads effectively.

12.4 Load balancing

Load balancing ensures that the Odoo system remains responsive under varying loads by distributing traffic across multiple servers:

- **Configuration:** Set up a load balancer to distribute user requests across multiple Odoo servers.
- **Monitoring:** Continuously monitor server performance and adjust load balancing configurations as needed to maintain optimal performance.
- **Failover:** Implement failover mechanisms to ensure system availability in case of server failures.

Chapter 13 Critical Appraisal and Evaluation

13.1 Objective that could be met

Every system has its limitations, and while the Smart Edu system has been designed to address key requirements, certain goals have been achieved, including:

- **Global Perspective for User Interface:** The Smart Edu system provides a comprehensive and intuitive interface accessible from various devices, ensuring a seamless experience for users across different platforms.
- **Registration System for Users (Admin):** Admins can efficiently manage user registrations, including students, teachers, and staff, with robust functionality for onboarding and profile management.
- **Fees Payment Management:** The system integrates a comprehensive fees payment module, allowing for efficient management of tuition fees, including generation of receipts and tracking of payment history.

13.1.1 Success rate against each objective

The Smart Edu system has demonstrated a high success rate in achieving its objectives:

- **User Interface:** The interface has been designed for ease of use and accessibility, providing a user-friendly experience for all stakeholders.
- **Registration System:** The registration and management of users have been streamlined, making administrative tasks more efficient.

Fees Payment: The fees payment module operates effectively, covering all necessary functionalities and ensuring smooth transactions.

13.1.2 How much better could have been done

While the Smart Edu system is functional and efficient, improvements could be made in the following areas:

Enhanced Library Management Features: Integrating a more advanced library management system could further benefit the school by streamlining book tracking, lending processes, and inventory management.

13.1.3 Why it could not be done

Scope and Budget Constraints: The initial scope of the project and budget limitations restricted the addition of more advanced features. Library management, while valuable, was not prioritized within the project's budget and timeline.

13.1.4 Which objectives have been missed

Advanced Statistical Data Views and Reporting: The system does not currently support advanced statistical views or detailed reporting functionalities that could provide deeper insights into school performance and analytics.

13.1.5 Why these objectives have missed

Budget and Time Constraints: Due to client budget constraints and the project's timeline, the development of advanced statistical and reporting features was not feasible. These features were not part of the core requirements specified by the client

13.2 Objectives totally not met / touched

Online Payment Integration: The system does not currently support online payment transactions through banks or other methods, which could have enhanced the convenience of fee payments.

13.2.1 Why it could not be touched

API Integration Challenges: Implementing online payment functionality requires integrating with payment gateways and APIs, which was not achievable within the project's budget and timeframe.

13.2.2 What could have been done

Mobile Banking Integration: To address the limitation of online payment, integrating mobile banking systems could be considered. This would allow parents to pay school fees through their mobile banking applications, enhancing payment convenience and functionality

Chapter 14 – Lesson Learned

14.1 Pre Project – Review – Closing:

The **Smart Edu** project, developed using Odoo, aimed to streamline and enhance school management processes. The project followed a structured framework, which included gathering client requirements, formulating a development strategy, selecting the appropriate architecture, and defining project objectives. The use of Odoo's integrated modules facilitated efficient management of student, teacher, and staff data, as well as administrative functions such as fees management and exam tracking.

Throughout the project, we focused on delivering a user-friendly and scalable system. The development process involved customizing Odoo's features to meet the specific needs of the school, ensuring that the system was tailored to the client's requirements.

14.2 I have learned

Working on the **Smart Edu** project has been an invaluable learning experience. Here are some key lessons learned:

- **Customization of Odoo Modules:** I gained significant experience in customizing Odoo modules to fit specific needs, which involved adjusting workflows and interfaces to align with the school's requirements.
- **Integration and Implementation:** I learned how to integrate various functionalities within Odoo, such as user management, fee tracking, and report generation, ensuring a seamless user experience.
- **User Training and Support:** Providing training and support to end-users was a critical part of the project. I developed skills in creating effective training materials and conducting user training sessions.
- **Database Management:** Working with Odoo's database management system provided insights into handling complex data structures and optimizing database performance.
- **Deployment and Scaling:** Deploying the system on a production server and ensuring it could handle the school's operational needs was a valuable experience in system deployment and scaling.

14.3 The Problems I Have Faced

During the **Smart Edu** project, several challenges emerged:

- **Customizing Odoo Modules:** Adapting Odoo's standard modules to meet specific requirements posed challenges, particularly in ensuring that customizations did not conflict with existing functionalities.
- **User Training:** Some users struggled to adapt to the new system due to varying levels of IT proficiency. Providing comprehensive training was essential to ensure smooth adoption.
- **Data Migration:** Migrating existing data into Odoo was complex, requiring careful mapping and validation to ensure data accuracy and integrity.
- **Integration Issues:** Integrating with external systems or modules sometimes led to compatibility issues, requiring additional troubleshooting and adjustments.

14.4 What solutions occurred

To address the challenges faced, several solutions were implemented:

- **Custom Module Development:** To overcome the issues with customizing Odoo modules, I focused on developing custom modules and configurations that aligned with the school's specific needs, while ensuring compatibility with existing functionalities.
- **Enhanced Training Programs:** To address the training challenges, I developed detailed training materials and conducted hands-on training sessions to ensure users were comfortable with the new system.

- **Data Migration Strategy:** I implemented a robust data migration strategy, including thorough data mapping, validation, and testing, to ensure a smooth transition from the old system to Odoo.
- **Integration Testing:** For integration issues, I conducted extensive testing and troubleshooting to resolve compatibility problems and ensure seamless integration with external systems.

By addressing these challenges and implementing effective solutions, the **Smart Edu** project successfully delivered a robust and functional school management system, providing valuable insights and experience for future projects.

Chapter 15 – Conclusion

15.1 Summary of the project

The **Smart Edu** project was developed to modernize school management by transitioning from traditional, manual methods to a comprehensive digital system using Odoo framework. Recognizing that many schools still rely on outdated, paper-based processes, I aimed to create a solution that addresses these challenges by providing an integrated, user-friendly platform. Through extensive research and analysis, both online and through field studies, the project incorporated various features necessary for efficient school management. The system includes modules for student and teacher management, fee processing, exam tracking, and more, all designed to enhance the operational efficiency of schools and improve the overall user experience.

15.2 Goal of the project

The primary goal of the **Smart Edu** project was to automate and streamline school management processes, thereby reducing administrative overhead and improving accuracy. The specific objectives included:

- **Digitizing School Management:** Implementing a digital system to replace manual record-keeping and paper-based processes.
- **Reducing Time and Errors:** Minimizing the time spent on administrative tasks and reducing the potential for human error.
- **Automating Processes:** Ensuring that various administrative functions, such as fee management and exam scheduling, are automated.
- **Centralizing Data:** Providing a centralized database for managing all school-related information.
- **User-Friendly Interface:** Designing an intuitive interface to simplify system operation for users.

- **Eliminating Paperwork:** Reducing the reliance on physical paperwork and documentation.

15.3 Success of the project

The **Smart Edu** system successfully meets all the essential requirements for modern school management. It provides a robust solution for managing various aspects of school operations, including student records, teacher information, and fee transactions. The system has been well-received by users and is capable of significantly improving the efficiency of school administration. The project's success lies in delivering a complete and functional management system that addresses the needs of educational institutions.

15.4 What I have done in the documentation (stages / activities / plans)

In documenting the **Smart Edu** project, I followed a structured approach to capture all stages and activities involved. The documentation includes:

- **Introduction:** An overview of the system and its objectives.
- **Preliminary Research:** Details of initial research and analysis conducted to define project requirements.
- **Development Process:** A comprehensive description of the development process, including design, customization, and implementation.
- **Testing and Training:** Documentation of the testing phase, including test plans and results, as well as training activities for end-users.
- **Deployment and Scaling:** Information on the deployment process and scaling considerations.
- **Tools and Technologies:** An overview of the technologies used, including Odoo and other related tools.

The documentation was created using Microsoft Word for text and tables, and third-party tools for figures and diagrams.

15.5 Value of the project

The **Smart Edu** system offers significant value by transitioning from manual to digital school management. It saves time, reduces administrative burden, and minimizes errors. The automation of various processes not only enhances operational efficiency but also improves the accuracy and accessibility of school data. The system's implementation represents a considerable advancement in how schools manage their daily operations and interact with students and staff.

15.6 My experience

Working on the **Smart Edu** project has been an enriching experience. Despite initial inexperience with some technologies, I acquired valuable skills and knowledge throughout the project:

- **Understanding Odoo:** I gained proficiency in customizing and implementing Odoo modules to meet specific needs.
- **Design and User Experience:** I learned about effective design practices and how to create user-friendly interfaces.
- **Database Management:** I developed skills in designing and managing complex databases.
- **Deployment and Training:** I gained experience in deploying systems and providing user training.

Overall, the project has been a significant learning opportunity, providing insights into software development, project management, and the practical application of technology in education.

Appendix

- [User guide](#)

For Admin:

The screenshot shows a web browser window with the address bar displaying 'disd.df.daffodil.family/web/login'. The page features the Daffodil International School logo and a navigation menu with links: Home, Courses, DIS Branches, Support, Forum, Contact us, Wiki, Shop, Events, Blog, About, and Application. There is a shopping cart icon with '0' items and a 'Contact Us' button. The main login area contains two input fields labeled 'Email' and 'Password', a 'Log in' button, and links for 'Don't have an account?' and 'Reset Password'.

Note: This login system is same for students, teachers. But admin need to create student and teacher profile. If anyone forget own login password, he/she can use password recovery method by email.

The screenshot shows the website disd.df.daffodil.family/web/reset_password. The header includes the Daffodil International University logo and navigation links: Home, Courses, DIS Branches, Support, Forum, Contact us, Wiki, Shop, Events, Blog, About, Application. There are links for 'Sign in' and 'Contact Us'. The main content area has a form titled 'Your Email' with an input field and a 'Confirm' button. Below the button is a link 'Back to Login'.

Note: At first admin need to create session, semester, class, course etc by academic option. After add student and teacher, admin can assign teacher in course and student include in class.

The screenshot shows the EMS interface for 'Admission Registers / [Session-2025]2024-2025'. It includes a table of applications with the following data:

Application Number	Student ID	Name	Admission Date	Application Date	Academic Year	State
AD1095	ID00536	Tausif Islam Rakin	08/31/2024	08/31/2024 15:54:11	[EM,Dhanmondi]2024-2025	Done
AD1092	ID00535	Ayman Rowshnai Khan	08/31/2024	08/31/2024 15:42:31	[EM,Dhanmondi]2024-2025	Done
AD1090	ID00534	Rubaiyat Mahmud	07/31/2024	08/31/2024 15:28:37	[EM,Dhanmondi]2024-2025	Admission Confirm
AD1088	ID00533	Md. Safatfat Mahmud	08/31/2024	08/31/2024 15:20:04	[EM,Dhanmondi]2024-2025	Done
AD1086	ID00532	Mohammad Mainul Hoq	08/31/2024	08/31/2024 14:50:06	[EM,Dhanmondi]2024-2025	Done
AD1083	ID00530	MD Aariz Uddin Saman	08/29/2024	08/31/2024 14:30:17	[EM,Dhanmondi]2024-2025	Admission Confirm
AD1081	ID00529	Aliza Uddin Sanan	08/31/2024	08/31/2024 14:22:57	[EM,Dhanmondi]2024-2025	Done
AD1079	ID00528	Anahita Atseen	08/28/2024	08/31/2024 14:13:21	[EM,Dhanmondi]2024-2025	Admission Confirm
AD1077	ID00527	Arish Anzar	08/31/2024	08/31/2024 13:26:03	[EM,Dhanmondi]2024-2025	Done
AD1075	ID00526	Sareem Hossain Qarar	08/26/2024	08/31/2024 13:05:34	[EM,Dhanmondi]2024-2025	Admission Confirm
AD1073	ID00525	Sauban Hossain Sooadh	08/31/2024	08/31/2024 12:55:22	[EM,Dhanmondi]2024-2025	Done

Note: This is teacher added form. Admin must be added teacher before assign teacher in course.as like this form, student and staff form same. Admin can add their in same way.

The screenshot shows the 'Teachers / New' form in the EMS. It includes fields for:

- Personal Information:** Title, First Name, Middle Name, Last Name, Gender, Email, Phone, Academic Department, Mobile.
- Subjects, Sessions, Library:** (Tabs for additional information)
- Birth Date, Nationality, Language, HR Employee:** (Dropdown menus)
- Blood Group, Visa Info, Emergency Contact:** (Dropdown menus)
- Address Detail:** Address Type, Address, Contact.

Note: Now I am describing and showing how to create exam, routine, event, syllabus, notice etc. Only admin can create this service with out exam. Exam & syllabus can create admin and teachers. Students can show their course related exam.

disd.df.daffodil.family/web#id=58&action=1456&model=time.table&view_type=form&cids=1&menu_id=983

EMS Admissions Teachers Students Evaluations Signature Configuration Fees Configuration Dashboard

Exam Timetable / Final Term Examination 2024

Edit Create Action

3 / 6

Description	Final Term Examination 2024	Academic Class	Grade 7
Time Table Type	Exam	Year	[Session-2024]2023-2024
Time Table Type	Exam		

TimeTable

Exam Date	Week Day	Subject Name	Start Time	End Time	Supervisor	Classroom
05/21/2024	Tuesday	English Language	10.00	12.00	Jenat Jahan	Grade-7
05/22/2024	Wednesday	English Literature	10.00	12.00	Jenat Jahan	Grade-7
05/23/2024	Thursday	Bangla	10.00	12.00	Rayan Rahman	Grade-7
05/26/2024	Sunday	Mathematics	10.00	12.00	Md. Shariful Islam	Grade-7
05/27/2024	Monday	Physics	10.00	12.00	Fahad Sulaiman	Grade-7
05/28/2024	Tuesday	Chemistry	10.00	12.00	Md. Golam Mohiuddin Masum	Grade-7
05/29/2024	Wednesday	Biology	10.00	12.00	Kamrun Nahar	Grade-7
05/30/2024	Thursday	Computer Science	10.00	12.00	Sweet Ronald Rodrick	Grade-7
06/02/2024	Sunday	Geography	10.00	12.00	Sweet Ronald Rodrick	Grade-7
06/03/2024	Monday	BDGS & Global History	10.00	12.00	Rayan Rahman	Grade-7
06/04/2024	Tuesday	Islamiat/Moral Science	10.00	12.00	Mehedi Hasan	Grade-7
06/05/2024	Wednesday	Commerce	10.00	12.00	Rayan Rahman	Grade-7

EMS Admissions Teachers Students Evaluations Signature Configuration Fees Configuration Dashboard

Regular Exam / Final Term Examination 2024

Edit Create Action

2 / 2

Generate result

Exam Detail

Exam Code	2024/06/070	Exam Name	Final Term Examination 2024
Grade System	Grading System	Academic Year	[Session-2024]2023-2024
Exam Start Date	06/10/2024	Exam End date	06/19/2024
Class	Grade 3		

Class Exam Schedule

Class	Exam Schedule
Grade 3	Final Term Examination 2024

Exam Created Date 06/10/2024 11:00:50 Exam Update Date 06/10/2024 11:01:41

Note: Grading rules can add by admin.but everyone can show it.

Grade / Grading System

Save Discard

Grading System	
Grade	
A+	
A	
B	
C	
D	
NI	
Add a line	

Note: Mainly admin can perform on fees payment and library management.

Fees Head / New

Save Discard

Name Code

Eligible for Waiver ☐

Calculations

Duration Amount

Account Company

Currency ☒ Symbol

Function

Note: This notice feature mainly admin control based.

The screenshot shows a web browser window with the URL `disd.df.daffodil.family/web#id=&action=1426&model=op.notice.board&view_type=form&cids=1&menu_id=983`. The browser's address bar shows "Incognito". The page has a purple header with the "EMS" logo and navigation links: Admissions, Teachers, Students, Evaluations, Signature Configuration, Fees, and Configuration. On the right of the header are icons for notifications (3), a bell, a mail icon, and a "Dashboard" link with a user profile icon.

Below the header, the page title is "Notice Board / New". There are two buttons: "Save" and "Discard". The main form area is divided into several sections:

- Date:** A dropdown menu labeled "Date of the notice".
- Class:** A dropdown menu labeled "Select Class".
- File Attachment:** A button labeled "Upload your file".
- Description:** A large text area with a rich text editor toolbar. The toolbar includes icons for bold (B), italic (I), underline (U), text color (A), background color (A), bulleted list, numbered list, link, unlink, undo, redo, and a trash icon.
- Notice Title:** A text input field with the placeholder "Notice Title".
- General:** A checkbox.

At the bottom of the form, there is a "Users" section with a dropdown menu.

Teacher Panel:

Note: Teacher can perform on this option. Such as take/view attendance, create assignment, give marks etc.

EMS Admissions Teachers Students Evaluations Signature Configuration Fees Configuration

Subjects / New

Save Discard

Name		Code	
Maximum marks	0	Minimum marks	0
Term Maximum marks	0	Term Minimum marks	0
Is Grouped	☐		
Is CT	☒	CT Maximum marks	0
		CT Minimum marks	0
Is MCQ	☐		
Is CQ	☐		
Is Practical	☐		
Elective Group			

Standards Teachers

disd.df.daffodil.family/web#id=&action=1304&model=op.assignment&view_type=form&cids=1&menu_id=1010

Assignments Assignments Configuration

Assignments / New

Save Discard

Publish Draft Put

Assignment Name

Class		Subject	
Teacher		Assignment Type	
Reviewer			

Basic Information Allocation Information Submissions

Issued Date		Submission Date	
Marks	0.00		

EMS

Admissions
Teachers
Students
Evaluations
Signature Configuration
Fees
Configuration

3

Exam Results / 1

Save

Discard

Confirm

Draft

Exam Details

Examination

Final Term Examination 2024

Student Details

Student Name

Shekh Abrar Zarif

Roll No

1

Class

Board Standard

Board Wise

☐

Without CT Number

☐

Without SBA Number

☐

Grade System

Grading System

Subject Name	Maximum Mar...	S.B.A	WE 20	Term Exam	Obtain Marks...	Percentage	Highest Mar...	GPA	Letter Grade	Marks After R...
English Language	100.00	19.00	12.00	51.00	82.00	82.00	82.00	4.00	A	
English Literature	100.00	20.00	20.00	50.00	90.00	90.00	90.00	5.00	A+	
Bangla	100.00	17.00	12.00	55.00	84.00	84.00	84.00	4.00	A	
Mathematics	100.00	13.00	16.00	40.00	69.00	69.00	69.00	2.00	C	
Computer Scie...	100.00	17.00	20.00	55.00	92.00	92.00	92.00	5.00	A+	
BDGS & Globa...	100.00	10.00	10.00	40.00	60.00	60.00	60.00	2.00	C	
Geography	100.00	19.00	19.00	60.00	98.00	98.00	98.00	5.00	A+	
Islamiat/Moral ...	50.00	10.00	8.00	30.00	48.00	96.00	48.00	5.00	A+	

Plagiarism Report

183-16-367

ORIGINALITY REPORT			
5%	4%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	3%	
2	Pawan Singh Mehra, Dharendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security - Volume 2", CRC Press, 2023 Publication	<1%	
3	Submitted to University of Lincoln Student Paper	<1%	
4	Tejaswini, Priyanka Rastogi, Swayam Dua, Manikanta, Vikas Dagar. "Early Disease Detection in Plants using CNN", Procedia Computer Science, 2024 Publication	<1%	
5	hdl.handle.net Internet Source	<1%	
6	Submitted to Ashesi University Student Paper	<1%	
7	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dharendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security", CRC Press, 2023 Publication	<1%	
8	Submitted to University of Greenwich Student Paper	<1%	