A PROJECT REPORT ON
FEE MANAGEMENT SYSTEM

By

Sonu Kumar Keshari
Student ID: 1802079
Level: 4 Semester: I
B.Sc.(Engineering) in CSE
Session: 2018

Umar Rabiu EL-Yakub
Student ID: 1802081
Level: 4 Semester: I
B.Sc.(Engineering) in CSE
Session: 2018

Course Code: CSE 420

Course Title: System Analysis and Design Sessional

Submitted To
Hasi Saha
Associate Professor
Department of Computer Science and Engineering (CSE)
Faculty of Computer Science and Engineering
Hajee Mohammad Danesh Science & Technology University, Dinajpur.

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF COMPUTER SCIENCE AND ENGINEERING
HAJEE MOHAMMAD DANESH SCIENCE AND TECHNOLOGY
UNIVERSITY, DINAJPUR-5200, BANGLADESH**

ABSTRACT

In institutions and Universities manual payment is mostly used in other for students to pay for their tuition fees and other payments required in that institution or university. Fee Management System is a solution to the manual payment system which is a desktop application that is used to take payments from the students and save their payments in a database with every detail being taken into the records and receipts are printed for the students to keep as an evidence for future purposes.It automates fee collection, tracking, and reporting, enabling administrators to handle fees-related tasks more efficiently and accurately. The system provides a platform for students, parents, and administrators to access relevant information about fees, including fee structures, payment schedules, and overdue fees. Abstract:

This project presents the development of a Fee Management System using Java programming language and MySQL database in NetBeans. The objective of the project was to create an efficient and user-friendly system for managing fees in educational institutions. The system encompasses key features such as student registration, fee payment, fee generation, and report generation. By automating the fee payment process, the system streamlines financial transactions, reduces errors, and minimizes manual efforts. Storing data in a centralized MySQL database ensures data accuracy, security, and easy retrieval. The system generates comprehensive reports on fee collections, outstanding payments, and student records, facilitating effective decision-making and financial planning.

Keywords : Fee Management System, Payments, Tuition Fee, Students, University, Database, Desktop Application.

Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction

Fees Management System is a software solution designed to simplify the process of managing fees-related activities in educational institutions. It automates fee collection, tracking, and reporting, enabling administrators to handle fees-related tasks more efficiently and accurately. The system provides a platform for students, parents, and administrators to access relevant information about fees, including fee structures, payment schedules, and overdue fees.

Additionally, the system allows for the generation of various financial reports, including invoices, receipts, and balance sheets. With the Fees Management System, educational institutions can streamline their fees-related processes, reduce manual workload, and enhance financial transparency. This ultimately helps institutions provide better services to their stakeholders and promote a more effective learning environment.

In today's education system, managing fees and finances has become an essential and challenging task. Manual processes for handling fees-related activities can be time-consuming, prone to errors, and can cause unnecessary delays in the payment process. Moreover, it can be difficult for students and parents to keep track of their payment histories and balance dues, resulting in confusion and disputes.

Fees Management Systems have become an essential tool for educational institutions to manage these issues. These systems help automate fees-related processes, providing a digital platform for students and parents to access information about fees and payments. They also enable administrators to track payments, generate reports, and communicate effectively with stakeholders. By centralizing all financial data, these systems promote transparency and accountability, making it easier to identify and address discrepancies or discrepancies.

In summary, Fees Management Systems are a valuable tool for educational institutions, making the management of fees and finances more efficient, transparent, and accurate. They help institutions save time and resources while enhancing the overall experience for students, parents, and administrators.

1.2 Problem statement

The current fee collection and management process in any organization is manual and time-consuming, resulting in errors and delays. Staff members spend a significant amount of time and resources on administrative tasks, such as calculating fees, generating invoices, sending reminders for overdue payments. This not only increases the workload of staff members but also leads to dissatisfaction among students and other users who may have to wait longer for their fees to be processed. The current system lacks the ability to generate reports and analyze fee- related data, which makes it difficult for the organization to make informed decisions about its finances. There is also a lack of transparency and accountability in the current systems which can lead to errors and inconsistencies in fee calculation and collection. A more efficient, automated and user-friendly fee management system is needed. By implementing a fees management system, the institutions or organization can improve the efficiency and accuracy of the fee collection process, reduce the workload of staff members, provide valuable insights into fee-related data, and improve the overall user experience.

1.3 Objective

The objectives of a fees management system can vary depending on the specific needs of the institutions or organizations, but generally include:

- Providing valuable insights into fee-related data.
- Improving the user experience.
- Streamlining administrative tasks.
- To provide a feature for resetting the image to its original format.
- Compliance, customization and integrations
- Secure and easy to understand
- To help abide by the rules and regulations of institutions and government.

1.4 Motivation

A fee management system has a number of advantages that can help educational institutions, training facilities, and other organisations manage their finances better. First off, automating the charge collection and management procedures can lighten the load on employees, eliminate mistakes, and enhance the customer experience. Second, the system can offer useful data insights that facilitate wise financial planning and decision-making. Thirdly, the system's customization function enables fee schedules and payment plans to be customised to the institution's particular requirements. Fourth, the collection of fees can be streamlined by connecting with other systems, such as student databases and financial management systems, which will lighten the effort of personnel. Fifth, the system can assist businesses in adhering to privacy and financial standards. Finally, a user-friendly design can make it easier for students and other users to pay their fees and make it simple for them to keep track of their progress. Institutions can optimise their financial management procedures, increase user payment experiences, and concentrate on providing high-quality educational and training services by using a fee management system.

Chapter 2

Literature Review

2.1 Introduction

A literature review is an essential component of academic research that provides an overview of the existing literature relevant to the research question or topic of interest. It involves a critical analysis and synthesis of published studies, articles, and other sources related to the research question. A literature review aims to identify the current state of knowledge on the research topic, identify research gaps, and provide a framework for the proposed research.

2.2 Related works

The adoption of Fees Management Systems has become increasingly common in educational institutions worldwide. Many studies have focused on the benefits and challenges of implementing these systems.

The developer in [1] presents methods for avoiding and reducing risks throughout the development of an information system, specifically electronic payment system to handle tuition in the universities in Indonesia. The university policies, regulations and system models are design in such a way to resolve the project key success factors. By implementing the proposed methods, the system has been successfully developed and currently operated. The research is conducted in Parahyangan Catholic University, Bandung, Indonesia.

A study by Alomari and Al-Bataineh (2020)[2] analyzed the use of Fees Management Systems in Jordanian universities. The researchers found that the system had a positive impact on the accuracy and efficiency of financial transactions, reducing errors and improving financial transparency. The study also noted that the system had some challenges, including resistance to change and a lack of technical expertise among some users.

Abdallah and Alshareef (2019) et al. focused on the impact of Fees Management Systems on the satisfaction of students and parents in Saudi Arabia. The researchers found that the system improved the satisfaction of stakeholders, particularly in terms of transparency, convenience, and accessibility of financial information. The study highlighted the need for institutions to provide proper training and support to stakeholders to ensure the effective use of the system.[3]

A study by Gathoni and Kimutai (2020) examined the use of Fees Management Systems in Kenyan universities. The study found that the system helped to reduce the workload of finance staff and enabled timely and accurate fee collection. However, the study also noted that some stakeholders faced challenges in accessing and using the system, particularly those with limited technical skills or internet connectivity.[4]

Overall, the literature highlights the potential benefits of Fees Management Systems, including improved financial accuracy, transparency, and convenience. However, studies also highlight the need for proper training, support, and technical expertise to ensure the effective adoption of these systems. Further research is needed to explore the impact of these systems in different educational contexts and to identify best practices for their implementation and use.

Chapter 3

Methodology

3.1 Introduction

This project proposed a method in which fees are not being paid using it but the payments record are saved in the system for later usage and there's a report used by the administration to analyzed and use for knowing the current state of the university's finances which helps in decision making.

3.2 Proposed System

After a detailed study of existing systems, it is evident that it does not fulfill our objectives. Hence it is necessary to build a new system that will make our objective possible. The main function of this application are as follows

- Students can pay for their courses during enrolment time.
- A student will be printed a receipt when he/she pays for any course.
- The system will provide the university a report to help in decision making situation.
- The Staff will be provided with an account which will be required for logging in
- A new course or subject can be added into the existing list easily using the system.

3.3 Requirement Analysis

The process of determining the needs and expectations of a new product is known as Requirements analysis or requirements engineering. It entails constant communication with the product's stakeholders and end users to set expectations, manage issues, and document all critical needs. Preliminary investigation plays an important role in developing a satisfactory requirement. This task involves information gathering. Data used for designing of the system were gathered through several means. Therefore the method used in the design and collections of information from various sources are as follows:

- Collecting and analysing existing materials on the project topic, written by different expert.
- Studying and understanding the input and output processes of the existing system.

- Primary data: This source has to do with the text book and videos consulted for the development of this project.

3.4 System Design

The process of defining the parts of a system, such as the architecture, modules, and components, the various interfaces of those components, and the data that flows through that system, is known as system design. It is intended to meet unique goals and requirements of a business or organization by creating a cohesive and well-functioning system. System design is the structural implementation of the system analysis.
We have depicted our system using flowchart, data flow diagrams, Schema diagram, Use case diagram and activity diagram.

3.4.1 Flowchart of the System :

A system flowchart is a form of diagram that depicts an algorithm or process by depicting the steps as various types of boxes and their order by linking these with arrows. A step-by-step solution to a given problem can be provided via this diagrammatic representation. These boxes represent process operations, and the arrows connecting them reflect the flow of control.

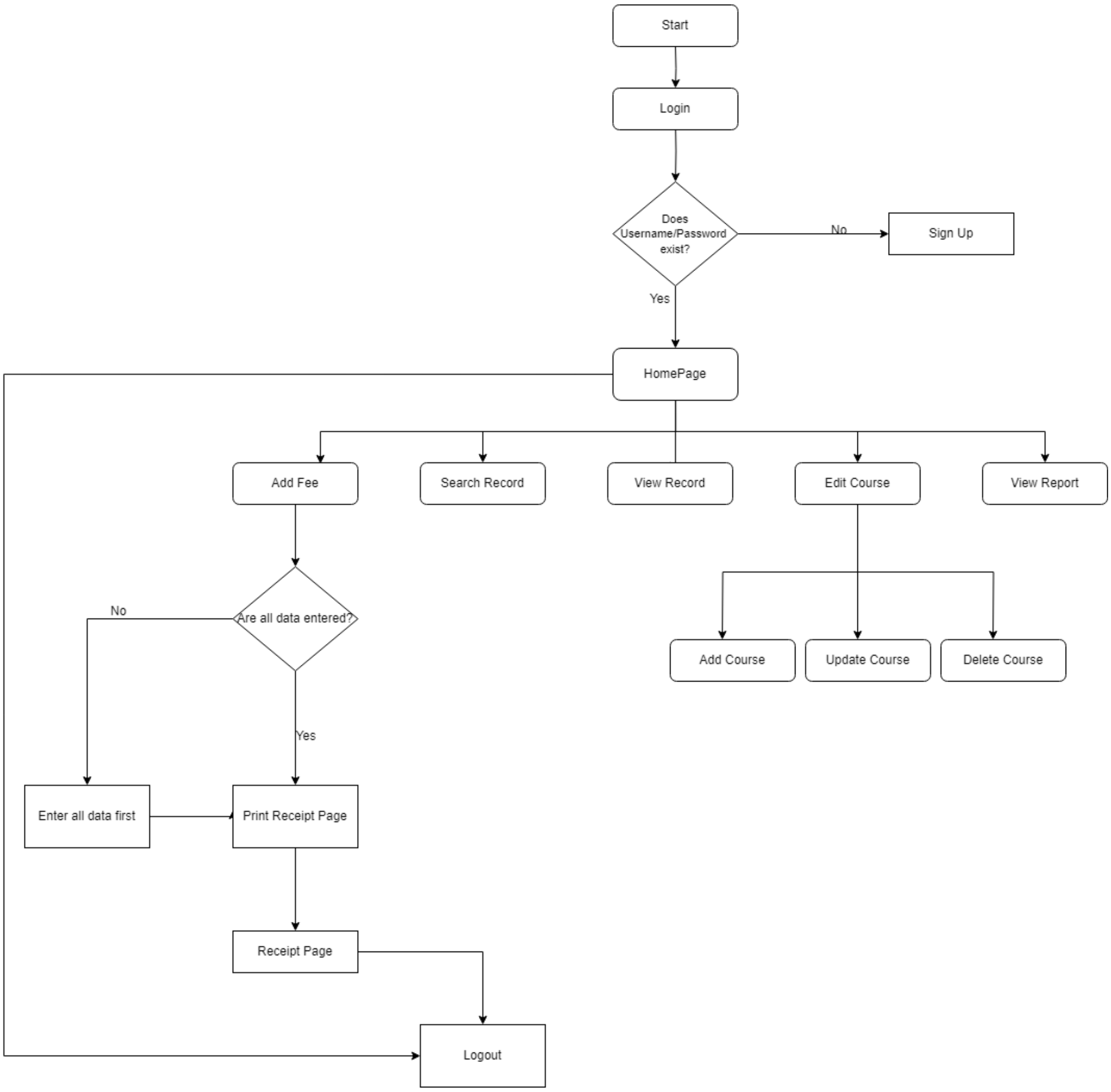


Figure 3.1: Proposed system flowchart

Flowcharts are used in a variety of industries to analyze, create, document, or manage a process or program.In the flowchart, several symbols are used to represent input, output, decisions, connectors, and processes.

3.4.2 Data Flow Diagram :

A Data Flow Diagram depicts the information flow for any process or system. It shows data inputs, outputs, storage sites, and the pathways between each destination using predetermined symbols such as rectangles, circles, and arrows, as well as short text labels. Data flowcharts can range from simple, even hand-drawn, process overviews to detailed, multi-level DFD that delve progressively deeper into how data is handled. They can be used to examine a current system or to create a new one. The four components of Data flow diagram are:

- External entity
- Process
- Data store
- Data flow

DFD can be used to zoom in on a specific piece of information by employing levels and layers. DFD levels are numbered 0, 1, or 2, and can sometimes reach Level 3 or higher. The level of detail required is determined by the scale of what we are attempting to do.

Here, we have shown level 0 and level 1 for this project.

3.4.2.1 DFD Level 0 :

DFD Level 0 is often referred to as a context diagram. It provides a high-level overview of the entire system or process being evaluated or modeled. It is intended to be a high-level overview of the system, displaying the system as a single high-level process with its relationships to external entities.

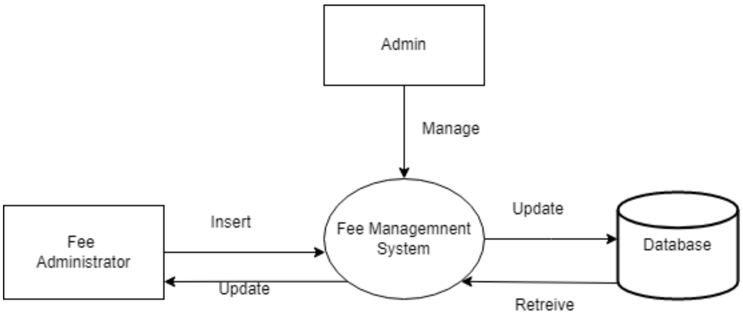


Figure 3.2: Proposed system's DFD Level 0

3.4.2.2 DFD Level 1 :

DFD Level 1 provides a more complete breakdown of context level diagram components. As we break down the high-level process of the context diagram into its subprocesses, we will emphasize the primary functions performed by the system.

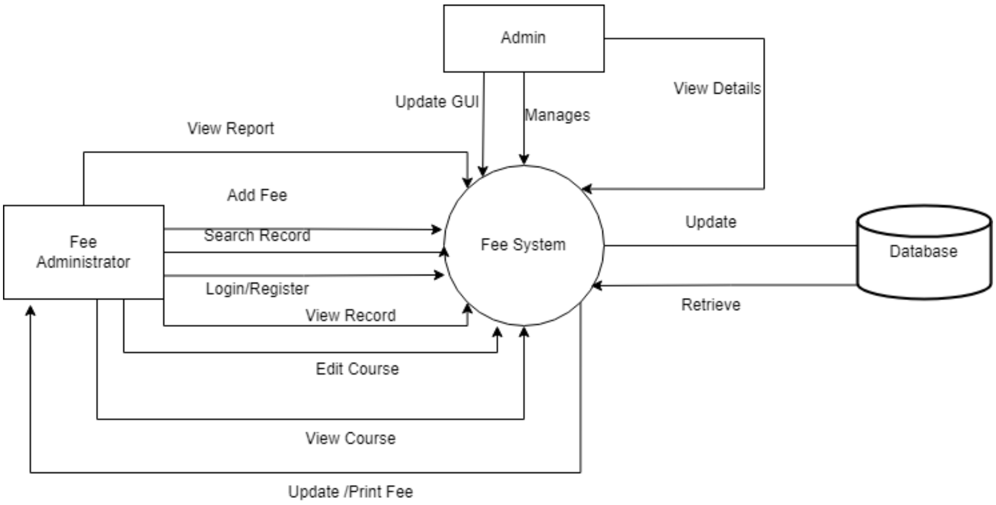


Figure 3.3: Proposed system's DFD Level 1

3.4.3 Activity Diagram

Activity diagrams, which support choice, iteration, and concurrency, are graphical depictions of workflows of sequential activities and actions [5]. The purpose of activity diagrams in the Unified Modelling Language (UML) is to represent both organisational and computational processes (i.e., processes) and data flows that cross across with adjacent operations[5] [6]. Despite the fact that activity diagrams typically depict the overall flow of control, they can also include components that depict the flow of data between activities through one or more data stores.

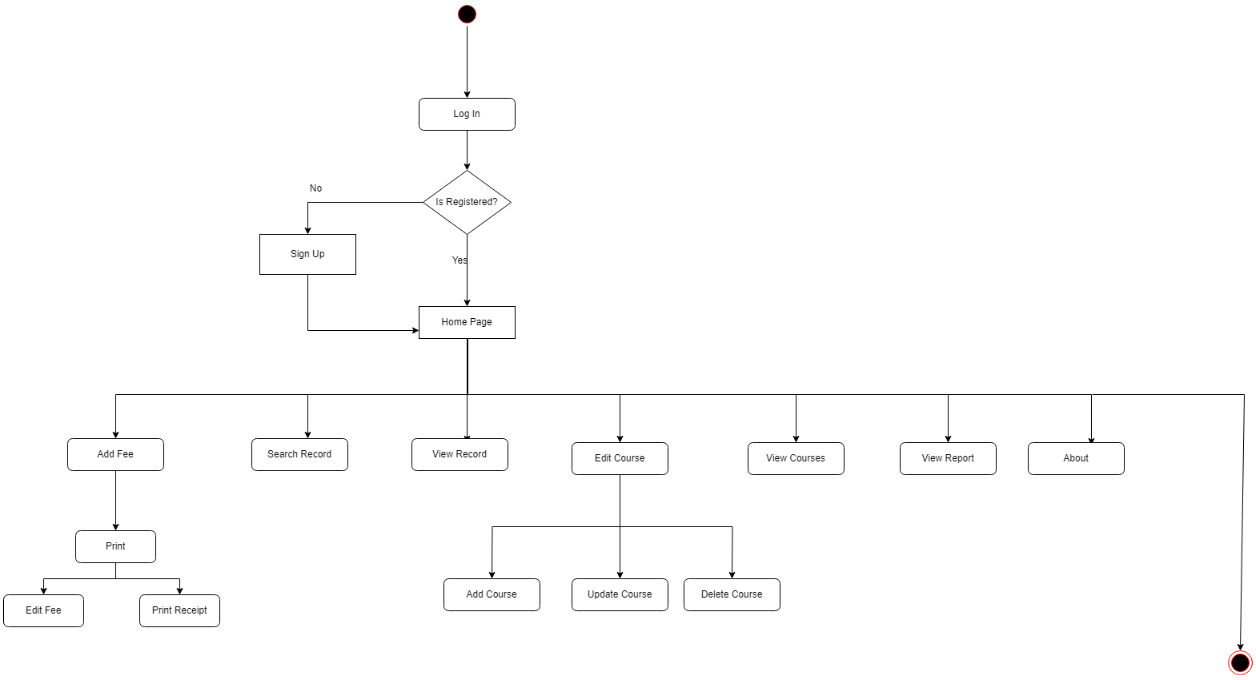


Figure 3.4: Activity Diagram of the proposed System

3.4.4 use case Diagram

It is a set of Scenaries that describe an interation between a user and a system. Use case diagram covers the whole and what functionalities it offers to different types of user .it makes easier to communicate between the client and the system developers.simply we can say that it display the releationship among actor and use case

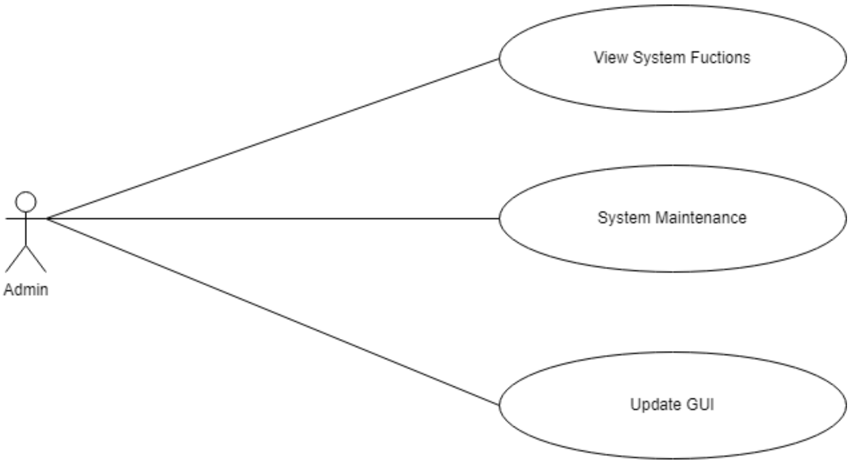


Figure 3.5: Use Case Diagram of the proposed System

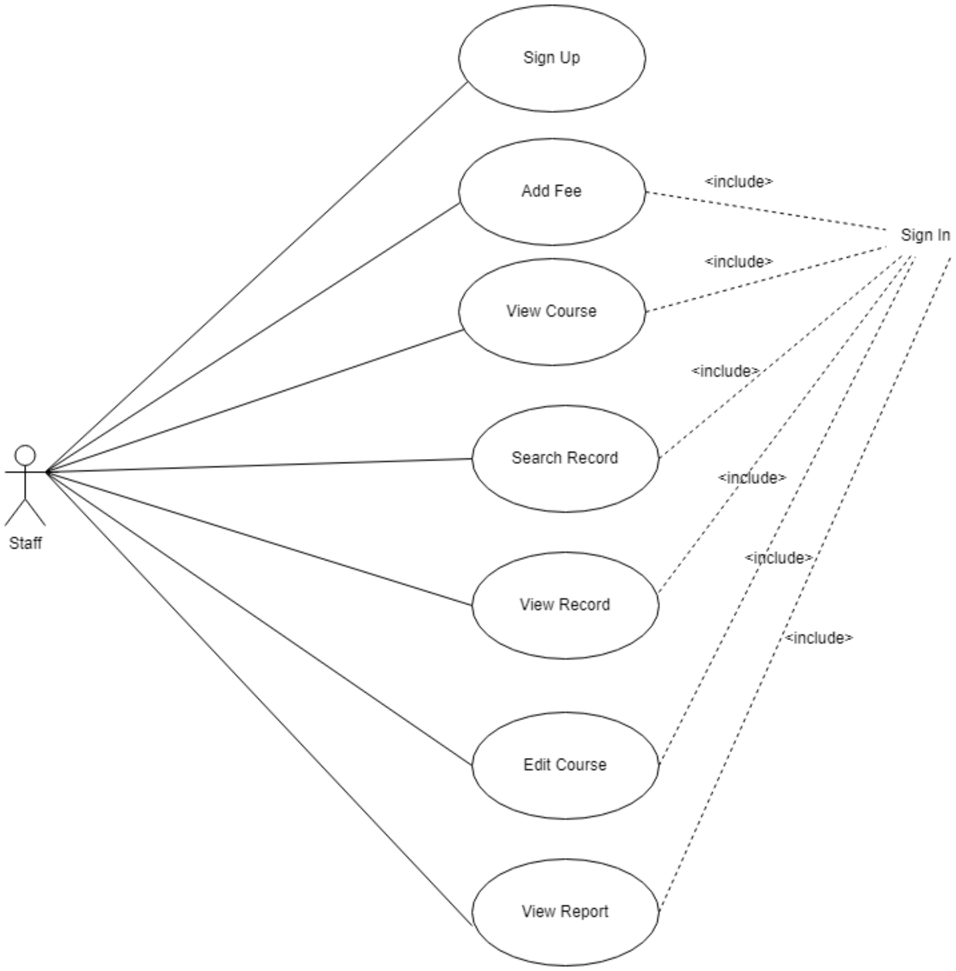


Figure 3.6: Use Case Diagram for staff of the proposed System

3.4.5 Schema Diagram

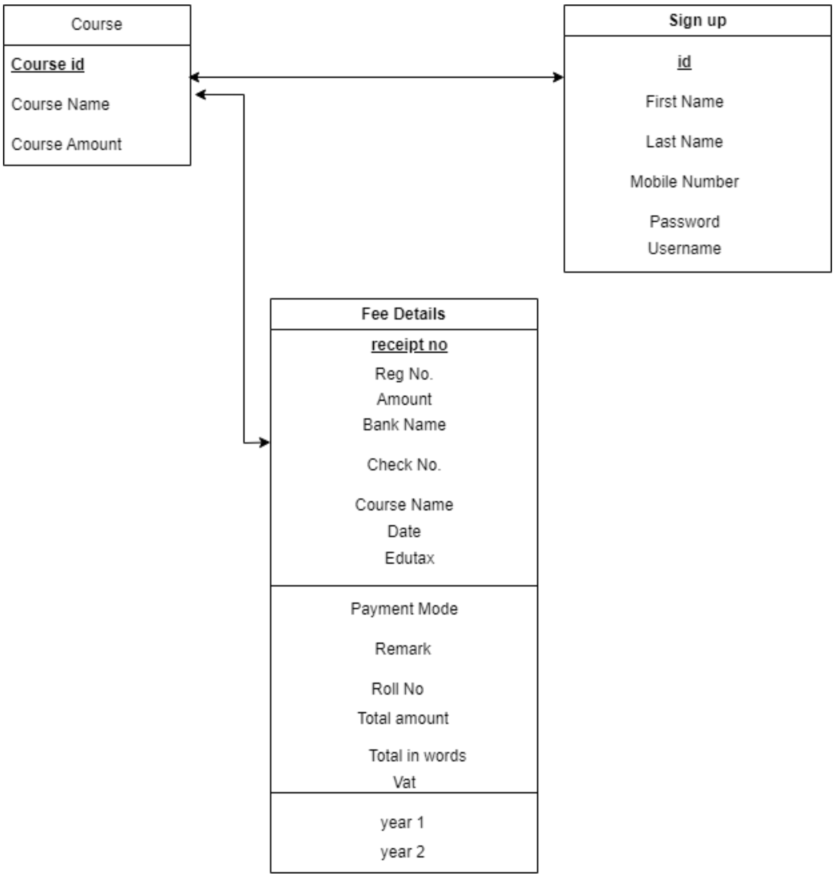


Figure 3.7: Schema Diagram

3.5 DataBase

A database is a systematic collection of data. It makes data management easy and efficient. Databases are designed to have a clear idea of each data and their interactions. Usually, a Database Management System (DBMS) is used to store, retrieve and modify the data. Some examples are, MySQL, Oracle, MongoDB, Firebase Realtime Database etc. In this fee management System,we are using mysql database. The database tables will be created using Data Definition Language (DDL). And other manipulations such as searching, filtering, updating and inserting, will be done by Query language under Data Manipulation Language (DML) [7].

3.5.1 Database table

table is a collection of related data held in a table format within a database. It consists of columns and rows. we are using three database table. they are

- signup
- course
- fee details

Chapter 4

Implementation

4.1 Introduction

The Implementation part defines the system in a manner similar to the Methodology section, but it does so in greater detail—down to the code level. The realisation of the earlier generated thoughts and ideas is the focus of this section. Additionally, it discusses any issues that might have shown up during implementation and how to fix them.

4.2 Tools and Technology

The tools and Technology used to develop fee management system are:

programming language

- **java** is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible.[8]

Framework

- **Java Swing** is a component of the Java Foundation Classes (JFC), which is how window-based applications are made. It is completely Java-written and built on top of the AWT (Abstract Windowing Toolkit) API.Java Swing offers lightweight, platform-independent components. Classes supporting the Java Swing API are available in the javax.swing package, including JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JMenu, JColorChooser, etc.

Database

- **MySQL** Currently,it is the most widely used database management system for relational databases. The Oracle Company supports this open-source database programme. Compared to Oracle Database and Microsoft SQL Server, it is a database management system that is quick, scalable, and simple to use.

Tools

- Netbeans
- Xampp
- Git

4.3 Requriement Analysis

Hardware Requirement

- Processor – Core i3 8th gen (Recommended – core i7 11th gen and above)
- RAM – 4 GB (Recommended – 8 GB)
- Disk space – 2 GB (Recommended – 10 GB)

Software Requirement

- Java – JDK 20 (Recommended – Openjdk version 20)

4.4 Challenges Faced

It's not simple to create an efficient and user-friendly desktop Application. The problems that could occur while creating a system are discussed in this section. For knowledge and experience in the future, it is beneficial to be aware of potential issues that can arise during the development process and how to handle them or avoid them. There are certain difficulties generating minor problems with the Fee Management System, but nothing severe.

Scalability: As the number of users and transactions increases, the fee management system must be able to handle the load without slowing down or crashing. Developing a system that can scale with the growth of the organization can be a significant challenge.

User interface: A user-friendly interface is essential for any system that will be used by a wide variety of users. Designing an interface that is intuitive and easy to navigate can be a significant challenge, particularly when dealing with complex financial data.

Testing and debugging: Testing and debugging are essential to ensure that the system works as intended. Ensuring that the fee management system is thoroughly tested and debugged can be a time-consuming and challenging process.

Maintenance: Once the fee management system is deployed, ongoing maintenance is necessary to ensure that the system remains functional and up-to-date.

Chapter 5

Results and Discussion

5.1 Introduction

In this Section, we discussed the outputs and the results of this project , software testing and debugging ,and maintainence of this system.

5.2 Module description

5.2.1 Login Page

This is the first page that a user gets when the application is opened, it helps the users log in into their account which include two fields namely; UserName and Password and it has three buttons login, Signup and exit button respectively. Below is the page snippet

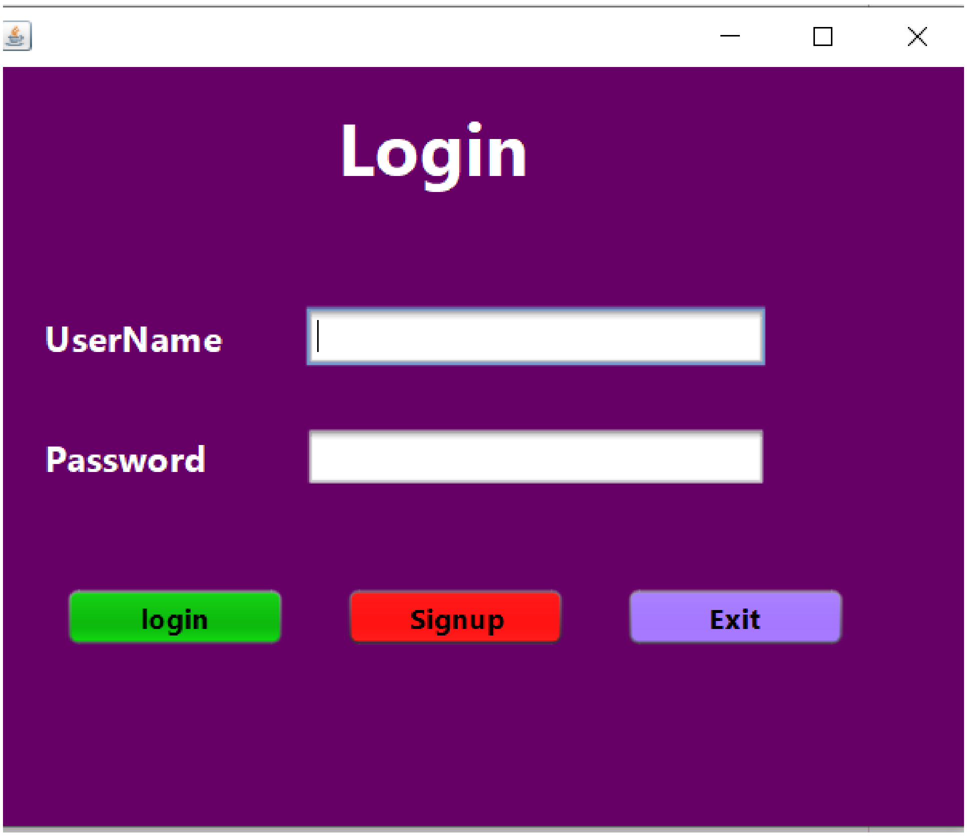


Figure 5.1: login

5.2.2 Signup

The user should required their firstname,lastame, username , password, Confirm password, Mobile no to signup in the application.

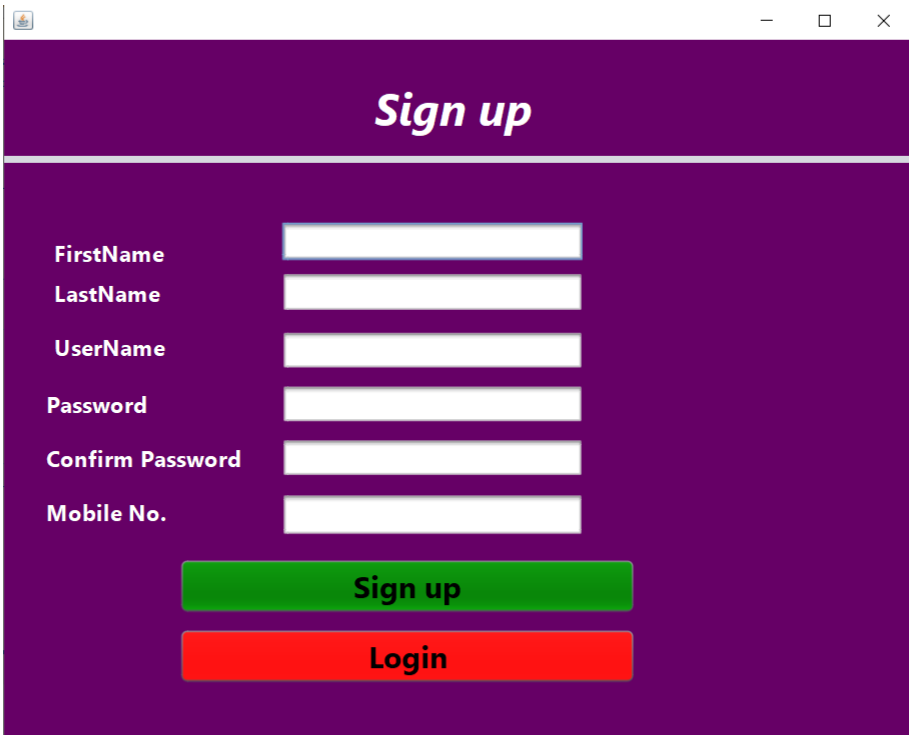


Figure 5.2: Signup

5.2.3 HomePage

A user that logs in with the required credentials is presented with this main and home page which provides a bridge to all the other pages in this system, and it gives the page to logout of the system. Below is the page snippet

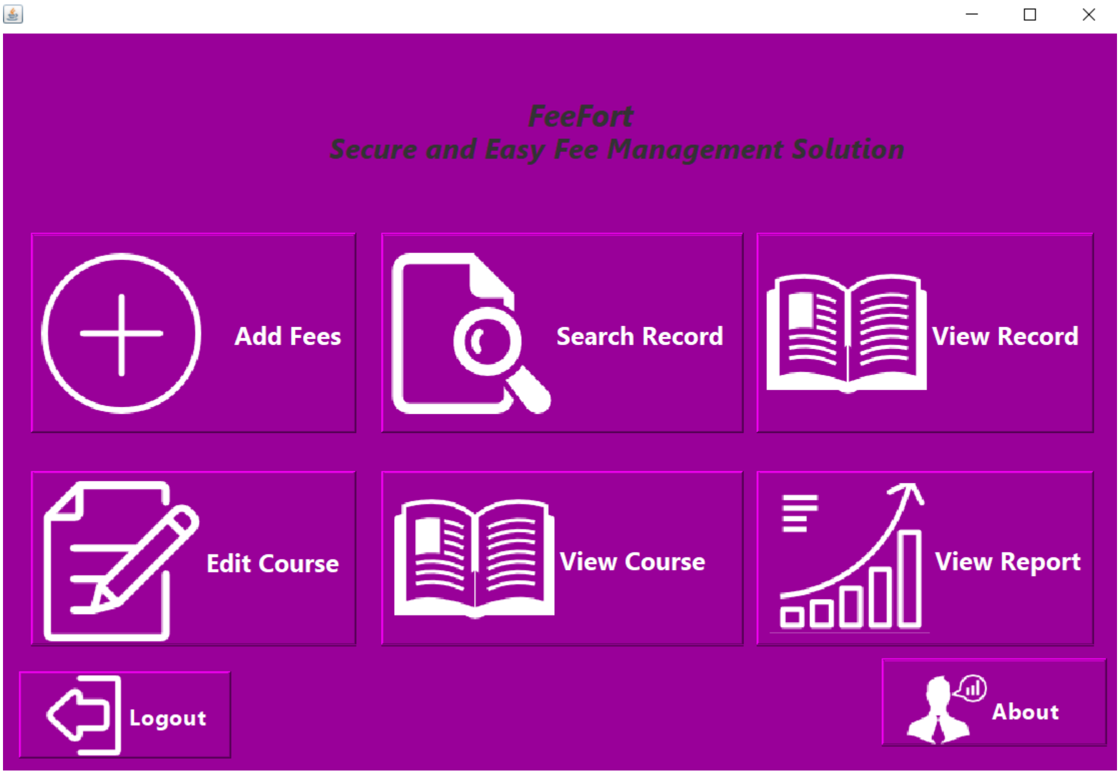
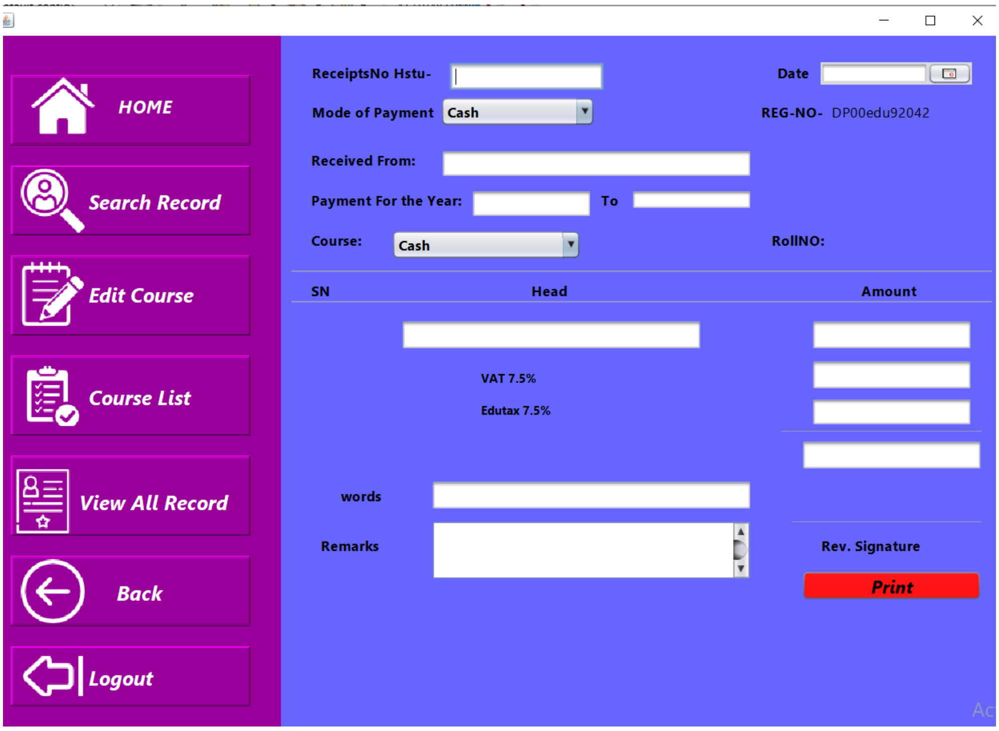


Figure 5.3: Homepage

5.2.4 Add Fees

This is where the payment made by the students is entered with all the required details from the Amount (VAT included), the name of the course being paid for, session, the mode of payment, date to the "Print" receipt button and so on. Below is the page snippet.



Mode of Payment Cash

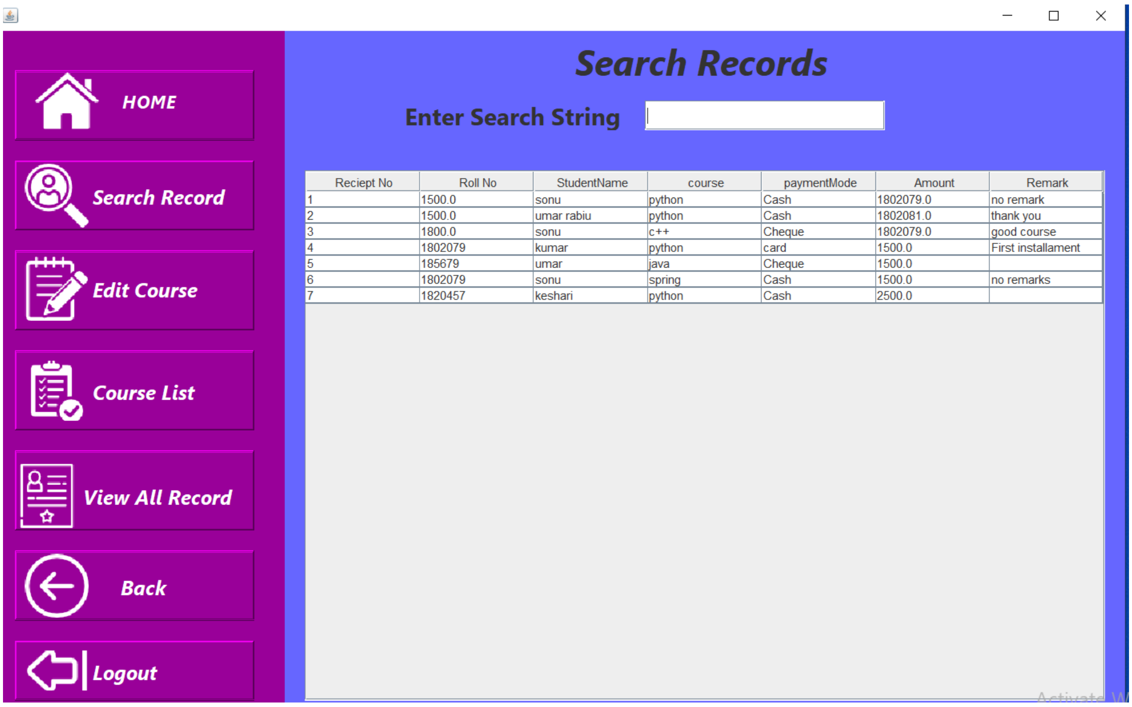
REG-NO- DP00edu92042

Course: Cash

Figure 5.4: Add Fee

5.2.5 Search Record

Search record is the page where all the previous payments made by the students can be checked either by Course name, Session, or by searching name of the student. Below is the page snippet.

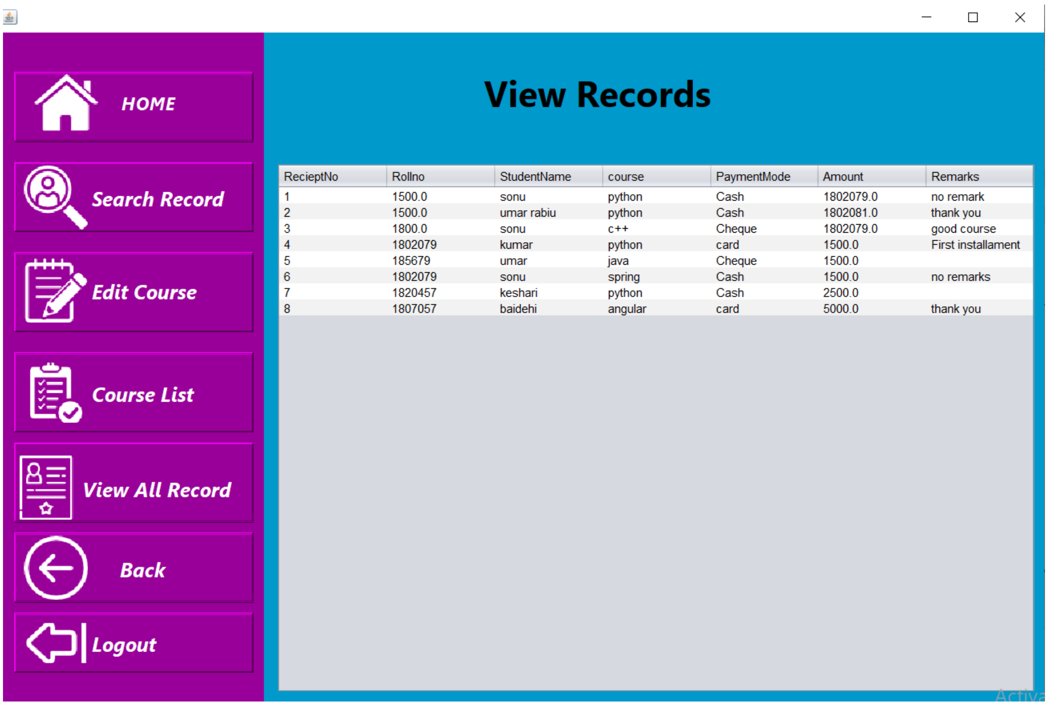


Reciept No	Roll No	StudentName	course	paymentMode	Amount	Remark
1	1500.0	sonu	python	Cash	1802079.0	no remark
2	1500.0	umar rabiu	python	Cash	1802081.0	thank you
3	1800.0	sonu	c++	Cheque	1802079.0	good course
4	1802079	kumar	python	card	1500.0	First installament
5	185679	umar	java	Cheque	1500.0	
6	1802079	sonu	spring	Cash	1500.0	no remarks
7	1820457	keshari	python	Cash	2500.0	

Figure 5.5: Search Record

5.2.6 View Record

This Page is like the search page it helps in accessing the previous records and payments made by the students, but on this page, there is no feature of searching the record can only be viewed according to the session or the course. Below is the page snippet.

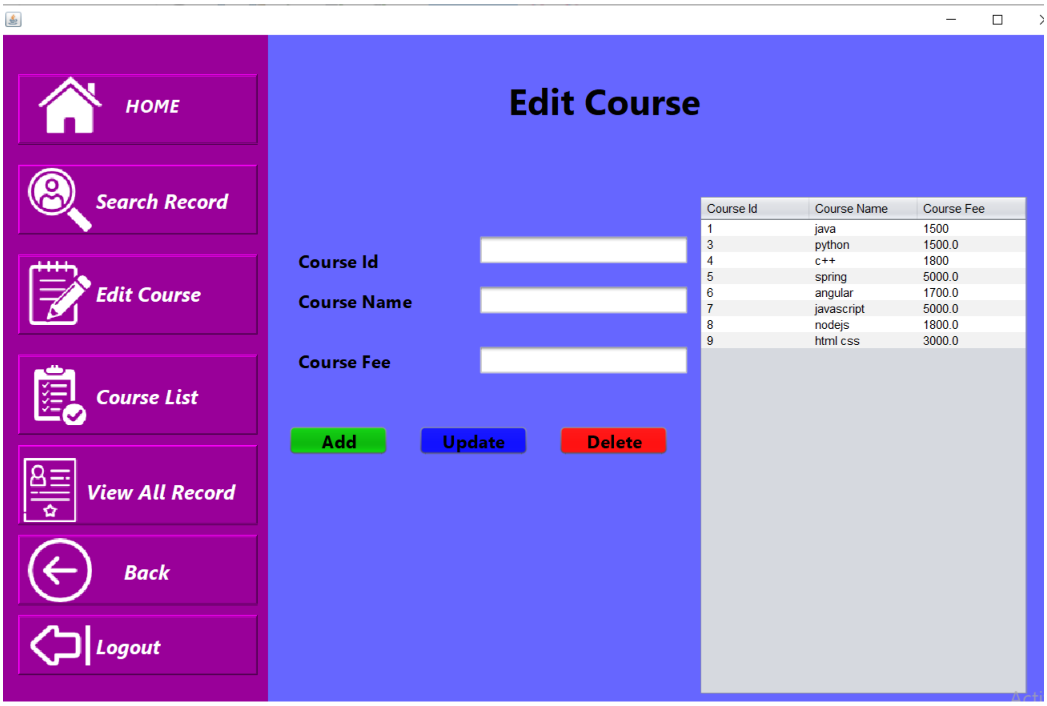


RecieptNo	Rollno	StudentName	course	PaymentMode	Amount	Remarks
1	1500.0	sonu	python	Cash	1802079.0	no remark
2	1500.0	umar rabiu	python	Cash	1802081.0	thank you
3	1800.0	sonu	c++	Cheque	1802079.0	good course
4	1802079	kumar	python	card	1500.0	First installament
5	185679	umar	java	Cheque	1500.0	
6	1802079	sonu	spring	Cash	1500.0	no remarks
7	1820457	keshari	python	Cash	2500.0	
8	1807057	baidehi	angular	card	5000.0	thank you

Figure 5.6: View Record

5.2.7 Edit Course

Courses can be edited by the administration using this page, a course might need to be updated, a new course can be added, and an existing page can be deleted, this can be done using this page. Below is the page snippet.



Course Id

Course Name

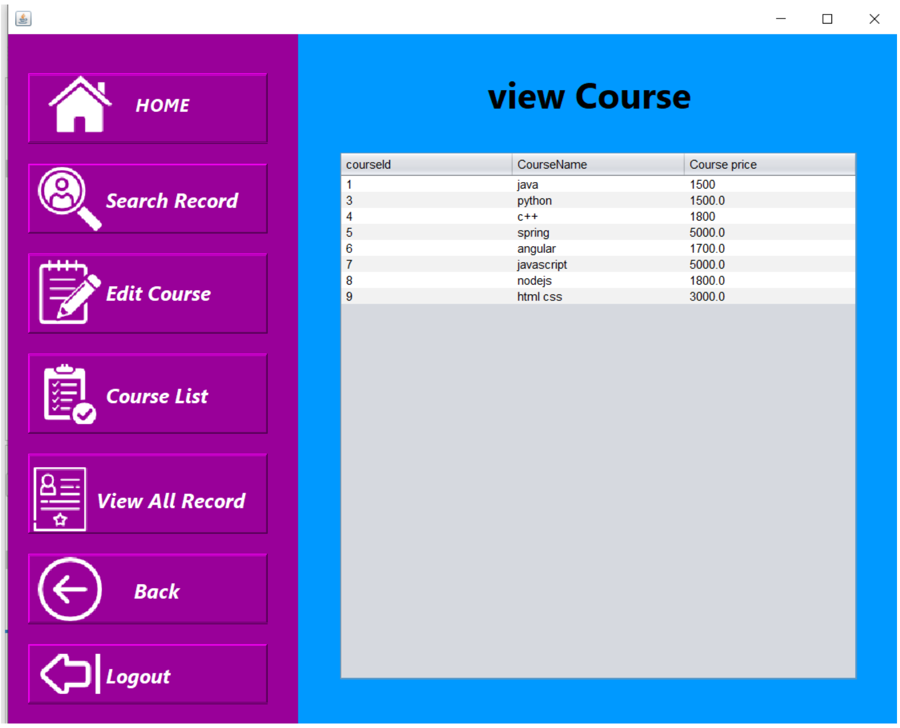
Course Fee

Course Id	Course Name	Course Fee
1	java	1500
3	python	1500.0
4	c++	1800
5	spring	5000.0
6	angular	1700.0
7	javascript	5000.0
8	nodejs	1800.0
9	html css	3000.0

Figure 5.7: Edit course

5.2.8 View Course

Courses offered by the university are listed and provided on this page and they have all the information needed for the students to see. Below is the page snippet.



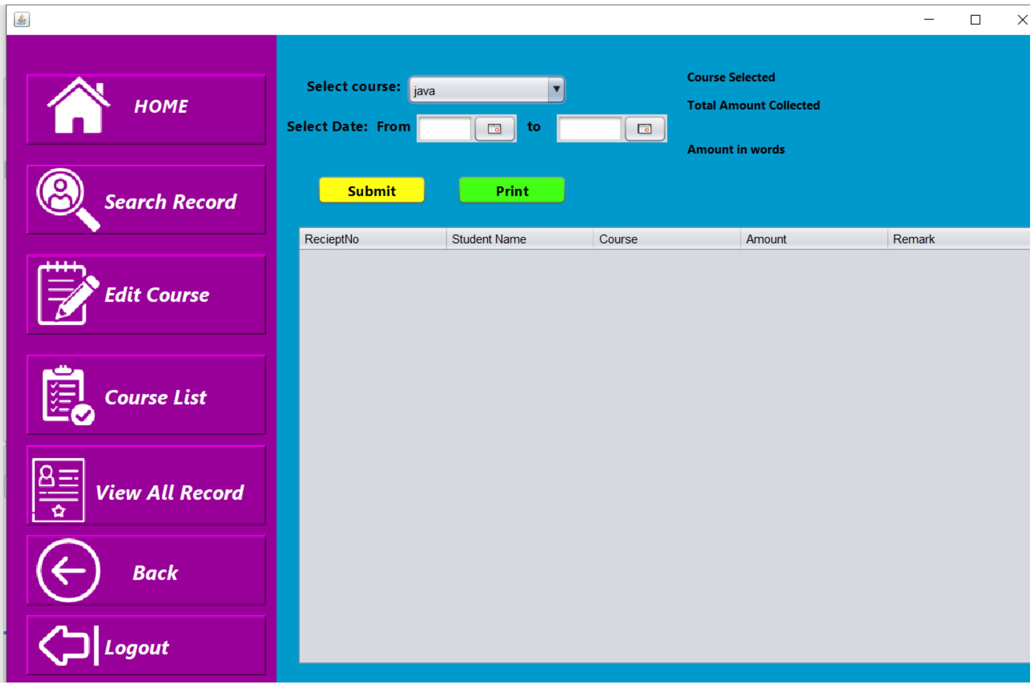
view Course

courseId	CourseName	Course price
1	java	1500
3	python	1500.0
4	c++	1800
5	spring	5000.0
6	angular	1700.0
7	javascript	5000.0
8	nodejs	1800.0
9	html css	3000.0

Figure 5.8: View Course

5.2.9 View Report

Institutions and companies have reports on their finances and are used to knowing how all the money is coming into the company or the institution, the same goes for the Universities on this page that is provided. Below is the page snippet.



Select course: java

Select Date: From to

Course Selected

Total Amount Collected

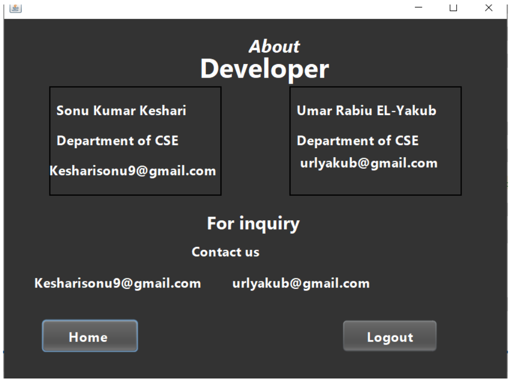
Amount in words

RecieptNo	Student Name	Course	Amount	Remark

Figure 5.9: View Report

5.2.10 About

In this page the information about the developers of this system is provided together with an email address for any other inquiries. Below is the page snippet



About
Developer

Sonu Kumar Keshari
Department of CSE
Kesharisonu9@gmail.com

Umar Rabiu EL-Yakub
Department of CSE
urlyakub@gmail.com

For inquiry
Contact us
Kesharisonu9@gmail.com urlyakub@gmail.com

Figure 5.10: About

5.3 System Testing and Debugging

Testing is an essential component of the software development process.Debugging is the process of correcting problems that occur during program execution.
System testing is concerned with putting the system through its paces in order to determine how far it has progressed in carrying out the desired mission. This was accomplished in two stages. The first is source code testing, which examines the program's logic. Second, there is specification testing, which entails examining the system in terms of what it should do and how it should do it under specified conditions. This comprises entering data, collecting output, comparing it to the output of the previous system, and evaluating whether it can replace the old system.

5.4 System Documentation

System documentation is a crucial aspect of implementation process. It describes the working of components and serves as a method of communication between application developers and users. It also helps future analysis of application either by the same or different system analysts and developers.

5.5 System Maintenance

Maintenance is the ongoing process of making changes and upgrading the application. This normally begins after the application has been put to use. There are three methods for keeping this program up to date. They include:

- **Adaptive maintenance :** Adaptive maintenance involves changing program functions to allow the information system to meet the user's information needs. Because of organizational changes, this form of upkeep may become necessary. Our system may need this type of maintenance according to the change in policies of the Institutes.

- **Corrective maintenance :** This sort of maintenance entails correcting mistakes from a program that may have snuck into the system as a result of defective design or incorrect assumptions. As a result, processing or performance faults are fixed during corrective maintenance.

- **Perfective Maintenance :** Perfective maintenance entails the addition of new programs or the modification of current programs in order to improve the functioning of the information system. This sort of maintenance is performed to respond to extra user needs that may arise as a result of changes within or outside of the business. Outside changes are essentially environmental changes, which may render the information system ineffective and inefficient in the absence of system maintenance. Our system may need to go under this type of maintenance too.

Chapter 6

Conclusion

6.1 Conclusion

In conclusion, this report examined the development of a Fee Management System implemented in Java language with a MySQL database in NetBeans. The project aimed to create an efficient and user-friendly system for managing fees in educational institutions.

Throughout the development process, key features were implemented, including student registration, fee payment, fee generation, and generating reports. The Java programming language provided a robust and versatile framework, while the MySQL database offered reliable data storage and retrieval capabilities.

The developed Fee Management System showcased several advantages. Firstly, it streamlined financial transactions by automating the fee payment process, reducing manual efforts, and minimizing errors. Secondly, it improved data accuracy and security by storing information in a centralized database, ensuring easy retrieval and backup options.

Moreover, the system enhanced administrative tasks by generating comprehensive reports on fee collections, outstanding payments, and student records. This allowed for better decision-making and financial planning within educational institutions.

6.2 Future Works

Although the application performs the intended task it was meant to do. There are some limitations the application has. Some features can be added; these includes:

- Development of equivalent mobile application more and allowing students and parents to access and manage fee-related information on their smartphones, thereby enhancing user convenience and accessibility.

- Provide more security to the system by using the likes of hashing algorithm and more.

- Adding Online Payment Options: Enhance the system by integrating popular online payment gateways, such as Bkash, Nagad and Rocket to provide additional payment options and convenience for users.

- Better UI design.

References

[1] V. S. Moertini, A. A. Athuri, H. M. Kemit, and N. Saputro, "The development of electronic payment system for universities in indonesia: On resolving key success factors," 2011.

[2] M. A. Alomari and A. M. Al-Bataineh, "The impact of electronic financial services on the efficiency and effectiveness of financial transactions: An empirical study on jordanian universities," *Journal of Accounting, Finance and Auditing Studies*, vol. 6, no. 4, pp. 320–335, 2020.

[3] M. I. Abdallah and S. M. Alshareef, "The impact of fees management systems on satisfaction of students and parents in saudi arabia," *International Journal of Engineering & Technology*, vol. 8, pp. 94–97, 2019.

[4] A. W. Gathoni and E. K. Kimutai, "An examination of the use of fees management systems in kenyan universities," *International Journal of Economics, Commerce and Management*, vol. 8, no. 12, pp. 44–61, 2020.

[5] U. R. T. Force, "Omg unified modeling language specification, version 1.4 (final draft)," February 2001.

[6] J. Rumbaugh, I. Jacobson, and G. Booch, *The Unified Modeling Language Reference Manual.* Addison-Wesley, 1999.

[7] A. Silberschatz, H. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. McGraw-Hill, 2010.

[8] B. Eckel, *Thinking in JAVA.* Prentice Hall Professional, 2003.