# WEB DEVELOPMENT INTERNSHIP

An Internship Report Submitted to Cotton University for the

Requirements of the UG 4th Semester Examination

In the Department of Computer Science and Information Technology

By

**Samir Jyoti Sarmah**

Enrollment No.: 2311010066


DEPARTMENT OF COMPUTER SCIENCE AND IT

COTTON UNIVERSITY, GUWAHATI

ASSAM-781001, INDIA

राष्ट्रीय इलेक्ट्रॉनिकी एवं सूचना प्रौद्योगिकी संस्थान (रा.इ.सू.प्रौ.सं.)
National Institute of Electronics and Information Technology (NIELIT)
गुवाहाटी केंद्र / Guwahati Centre
इलेक्ट्रॉनिकी एवं सूचना प्रौद्योगिकी मन्त्रालय / Ministry of Electronics and Information Technology (MeitY)
भारत सरकार / Government of India

Ref. No.: 04/37/2017/GHY/TRG/Summer-Internship/**WD-002** Dated: **04/08/2025**

# CERTIFICATE

This is to certify that **Samir Jyoti Sarmah**, a student of **BCA, , Cotton University, Guwahati** bearing the Enrolment number:2311010066, had undertaken a **120-Hours** Summer Internship Program on Web Development at **NIELIT Guwahati**, from 20/06/2025 to **20/07/2025.** The Summer Internship Program was in the domain of **Internship of Web Development**, along with a mini project on **"Web Based Student Record Management System".**

**Mridul Pachani**
**Course Co-ordinator**
**NIELIT Guwahati**

**Sh. Santanu Borgohain**
**Scientist-'E'**
**NIELIT Guwahati**


DEPARTMENT OF COMPUTER SCIENCE AND IT
**COTTON UNIVERSITY, GUWAHATI-781001**
Website: https://csit.cottonuniversity.ac.in
Phone No.: 0361-2608074

Date:……

# CERTIFICATE

The Internship Report entitled "**WEB BASED STUDENT RECORD MANAGEMENT SYSTEM**" submitted by **Samir Jyoti Sarmah (2311010066)** for the requirements of the UG 4th Semester Examination of Cotton University has been examined.

| | |
|---|---|
| Signature of Internal Examiner | Signature of External Examiner |
| Date: | Date: |
| Place: | Place: |

# DECLARATION

I, **Samir Jyoti Sarmah**, bearing **Enrollment No.:2311010066**, hereby declare that the subject matter of the Internship entitled ***"WEB DEVELOPMENT"*** is the record of work done by me. I further declare that the contents of this Internship report did not form the basis for the award of any degree to me or to anybody else to the best of my knowledge. The report has not been submitted to any other University or Institution. This report is being submitted to Cotton University, Guwahati-01 for the UG 4th Semester Examination in the Department of Computer Science and Information Technology

Place:
Date:

**Samir Jyoti Sarmah**
Enrollment No.: 2311010066
Dept. of Computer Science and IT
Cotton University, Guwahati-01

# ACKNOWLEDGEMENT

I am extremely grateful to my supervisor respected **Melur Phangcho**, Department of Computer Science and Information Technology, Cotton University, Guwahati-01 for his valuable advice, encouragement, endless support and guidance with patience during the course of my Internship work and write up this Internship report.

I would like to express my sincere gratitude to the Head of the Department, all the faculty members, students, non-teaching staff and office staff members of the Department of Computer Science and Information Technology, Cotton University, Guwahati-01 for their help and moral support in various forms during the course of my Internship work

I also express my sincere thanks to the entire team at NIELIT Guwahati for providing a conducive learning environment and for their constant assistance and motivation during the course of this project

Samir Jyoti Sarmah

Enrollment No.: 2311010066

# TABLE OF CONTENTS

**DECLARATION**

**ACKNOWLEDGEMENT**

**LIST OF ABBREVIATIONS** **I**

**ABSTRACT** **IV**

**1 Introduction** **1**

1.1 Overview of the Internship . . . . . 1

1.2 Scope of the Internship . . . . . 1

1.3 Organization Profile . . . . . 2

1.4 Duration and Internship . . . . . 3

1.5 Summary . . . . . 3

**2 Technologies and Tools** **4**

2.1 Programming Languages and Web Technologies Used . . . . . 4

2.2 Development Tools . . . . . 4

2.3 Database Design and Structure . . . . . 5

2.3.1 Database Operations . . . . . 7

**3 Internship Task Allocation** **8**

3.1 Overview of Assigned Tasks . . . . . 8

3.2 Nature of Work . . . . . 8

3.3 Skills Developed . . . . . 9

3.4 Summary of Internship Experience . . . . . 10

**4 Introduction to System Design** **11**

4.1 System Architecture . . . . . 11

4.2 Use Case Diagram . . . . . 12

4.3 Entity-Relationship (ER) Diagram . . . . . 14

4.4 Database Design . . . . . 15

4.5 Data Flow Diagram . . . . . 15

4.6 Summary . . . . . 16

**5 Overview of the Development** **17**

5.1 Technology Stack Used . . . . . 17

5.1.1 Frontend Tools: HTML5, CSS3, JavaScript . . . . . 17

5.1.2 Backend Technologies: PHP with PDO . . . . . 17

5.1.3 Database: MySQL . . . . . 18
5.1.4 PDF Generation: FPDF . . . . . 18
5.1.5 Local Server Setup: XAMPP . . . . . 18
5.2 Core Features Implemented . . . . . 19
5.3 Summary . . . . . 21

**6 Result and Learnings** **22**
6.1 Result . . . . . 22
6.2 Discussion . . . . . 22
6.3 Key Takeaways and Learning Experiences . . . . . 23

**7 Conclusion** **24**
7.1 Summary of the Training and Internship Experience . . . . . 24
7.2 Future Enhancements and Career Insights . . . . . 24

**REFERENCES** **26**

# LIST OF ABBREVIATIONS

| | |
|---|---|
| **HTML** | HyperText Markup Language |
| **CSS** | Cascading Style Sheets |
| **JS** | JavaScript |
| **PHP** | Hypertext Preprocessor |
| **SQL** | Structured Query Language |
| **XAMPP** | Apache + MySQL + PHP + Perl |
| **UI** | User Interface |
| **UX** | User Experience |
| **BCA** | Bachelor of Computer Applications |
| **FYUG** | Four-Year Undergraduate |
| **CRUD** | Create, Read, Update, Delete |
| **API** | Application Programming Interface |
| **PDF** | Portable Document Format |
| **DFD** | Data Flow Diagram |
| **DBMS** | Database Management System |

# LIST OF FIGURES

1.1 NIELIT Guwahati Official Logo . . . . . 2

2.1 Structure of the students table viewed in phpMyAdmin . . . . . 6

4.1 System Architecture: Student Record Management System . . . . . 12
4.2 Use Case Diagram of the Student Record Management System . . . . . 14
4.3 ER Diagram of the Student Record Management System . . . . . 14
4.4 Database Design: students table in student_db . . . . . 15
4.5 Data Flow Diagram of the Student Record Management System . . . . . 16

5.1 Technology Stack Diagram of the Student Record Management System . . . . 19
5.2 Interface of Adding Records . . . . . 20
5.3 Interface for Editing and Deleting Records . . . . . 20
5.4 Interface of View Search Records (with Filters) . . . . . 21

# LIST OF TABLES

2.1 Technologies and Tools Used . . . . . 4
2.2 Development Tools Used . . . . . 5
2.3 Structure of the `students` Table . . . . . 6

3.1 Weekly Task Allocation . . . . . 8

4.1 Database Table Design for `students` . . . . . 15
4.2 Summary of System Design Contributions . . . . . 16

# ABSTRACT

This report presents the work completed during my Summer Internship at NIELIT Guwahati, undertaken as part of the 4th Semester Undergraduate curriculum in the Department of Computer Science and Information Technology, Cotton University. The objective of the internship was to design and develop a web-based Student Record Management System that facilitates efficient handling of academic records through a user-friendly interface. The system allows for student data entry, filtering, and PDF report generation, ensuring smooth access and management of records by both administrators and end-users. Built using HTML, CSS, JavaScript, PHP (with PDO), and MySQL, the application leverages server-side scripting for secure data operations and FPDF for document generation. Special attention was given to usability, data validation, and database integrity to ensure reliable and consistent performance. This project provided valuable hands-on experience in full-stack web development and reinforced theoretical knowledge with real-world application. Key outcomes include improved proficiency in front-end and back-end technologies, deeper understanding of CRUD operations and database connectivity, and enhanced problem-solving and system design capabilities. The internship served as a practical platform to bridge academic learning with software development practices, laying a strong foundation for future professional growth.

# CHAPTER 1

## Introduction

This chapter introduces the internship undertaken as a part of the UG 4th Semester academic requirement of the Department of Computer Science and Information Technology, Cotton University. It outlines the goals and relevance of the internship, the nature of the project assigned, and the skills acquired through the development process. The following sections present a detailed account of the internship objectives, project description, tools and technologies used, and the key outcomes gained from this academic experience.

## 1.1 Overview of the Internship

The primary objective of the internship is to provide practical exposure to the concepts and technologies studied during the academic coursework. It was designed to help students gain a better understanding of how real-world software systems are developed and maintained. The internship aimed to bridge the gap between theoretical knowledge and industry practices.

**The specific objectives of the internship were as follows:**

- To understand the end-to-end process of building a dynamic web-based application.
- To apply front-end and back-end skills in a practical project environment.
- To implement secure database connectivity using **PHP** with **PDO**.
- To gain hands-on experience in data storage and retrieval using **MySQL**.
- To design key features such as **search filters, validation, and PDF export**.
- To improve debugging, problem-solving, and code optimization techniques.
- To gain familiarity with essential tools like **XAMPP** and **FPDF**.

## 1.2 Scope of the Internship

The scope of the internship was focused on the **design, development, and implementation** of a real-world web application titled "**Web Based Student Record Management System**." The internship provided an opportunity to work on the entire development lifecycle of a software project from planning and analysis to coding, testing, and documentation.

**During the internship, the scope included:**

- Requirement analysis for a basic student record management system.
- Designing an intuitive and responsive user interface using HTML, CSS, and JavaScript.
- Implementing backend functionalities using PHP and PDO for secure database access.
- Integrating a MySQL database for storing and retrieving student records.
- Enabling filtered search functionality based on Roll Number, Semester, and Result.
- Developing a feature for generating downloadable PDF reports using the FPDF library.
- Testing the system locally using XAMPP to ensure accuracy, usability, and data integrity.

The internship was confined to a single-user level system without user authentication, but it established a strong foundation for building more complex and scalable systems in the future. It also helped strengthen the intern's ability to work independently and meet development goals within a specified time frame.

## 1.3 Organization Profile

The National Institute of Electronics and Information Technology (NIELIT), Guwahati is a premier organization under the Ministry of Electronics and Information Technology (MeitY), Government of India. It is dedicated to providing professional education, training, and skill development in the fields of electronics, information technology, and allied areas. NIELIT Guwahati offers a wide range of programs including formal and non-formal courses, capacity-building initiatives, consultancy, and research-based training projects.


Figure 1.1: NIELIT Guwahati Official Logo

The institute plays a significant role in bridging the skill gap in emerging technologies by collaborating with industries, academic institutions, and government bodies. With well-equipped labs, qualified instructors, and an industry-relevant curriculum, NIELIT Guwahati provides a conducive environment for hands-on learning and practical exposure. During the internship, students gain real-time experience in software development, project execution, and team collaboration, making it a valuable platform for professional growth. The organization's structured

internship program ensures that students align their academic learning with industry demands, especially in domains like web development, programming, data analytics, cybersecurity, and digital technologies.

## 1.4 Duration and Internship

The internship was undertaken as part of the academic curriculum for the 4th Semester of the Undergraduate (FYUG) Bachelor of Computer Applications (BCA) program at Cotton University. It spanned a duration of one month, commencing on 20th June 2025 and concluding on 21st July 2025. During this period, the internship offered practical exposure to core front-end and back-end web development technologies, including HTML, CSS, JavaScript, PHP, and MySQL. The learning process was well-structured and integrated both theoretical knowledge and hands-on development tasks, providing an ideal platform for skill development and real-time problem-solving.

This experience significantly contributed to improving the intern's understanding of industry-relevant tools and practices, while also enhancing soft skills like collaboration, time management, and technical documentation. The internship bridged the gap between academic learning and industry expectations, offering valuable exposure to a professional work environment.

## 1.5 Summary

This chapter summarized the Academic Summer Internship at NIELIT Guwahati, undertaken as part of the 4th Semester FYUG Program at Cotton University. It covered the internship's objectives, scope, and the host organization's profile. A week-wise timeline outlined the training and practical tasks. The internship effectively connected academic learning with real-world experience, enhancing technical skills and professional growth. It also provided exposure to web technologies and development practices in a professional setting. The experience has strengthened foundational skills and encouraged further exploration in the field.

# CHAPTER 2

## Technologies and Tools

The development of the student record management system required a combination of both front-end and back-end technologies. A variety of tools and platforms were utilized to streamline the development process, manage the database, design the user interface, and generate reports. These technologies were selected for their wide industry usage, reliability, and support for rapid application development. By integrating these tools effectively, the project ensured a smooth, secure, and responsive user experience for both administrators and users.

## 2.1 Programming Languages and Web Technologies Used

During the internship, a variety of programming languages and web technologies were employed to design and develop the Student Record Management System. These technologies were selected based on their wide adoption, flexibility, and compatibility with modern web application development practices. The table below summarizes the core languages and web technologies utilized and their respective roles in the system.

Table 2.1: Technologies and Tools Used

| TOOL / TECHNOLOGY | CATEGORY | PURPOSE |
|---|---|---|
| HTML5, CSS3 | Frontend | Designing the structure and styling of the user interface |
| JavaScript | Frontend | Client-side interactivity |
| PHP (with PDO) | Backend | Server-side scripting and secure database interaction |
| MySQL, FPDF (PHP Library) | Database / PDF Generation | Data storage and management; exporting filtered records into PDF reports |
| XAMPP | Local Server | Running the PHP server and MySQL database locally |

## 2.2 Development Tools

The development of the Student Record Management System was greatly facilitated by a comprehensive set of tools and technologies that played a crucial role throughout the development lifecycle. These tools not only streamlined the processes of coding, testing, and deployment but also significantly enhanced developer productivity, ensured cross-platform consistency, and enabled effective debugging and maintenance. From local server management and code editing

to real-time testing and report generation, each tool contributed to building a robust and efficient application environment, ensuring the successful implementation of project goals.

Table 2.2: Development Tools Used

| TOOLS | DESCRIPTION |
|---|---|
| XAMPP | Used to test and run the application, inspect layout responsiveness, and debug JavaScript code using developer tools. |
| phpMyAdmin | A browser-based tool for managing MySQL databases. It was used for creating tables, running SQL queries, and maintaining the student records database. |
| MySQL | A widely used open-source relational database management system (RDBMS) used to store and retrieve structured student data. It provided a robust backend for handling operations such as record insertion, filtering, and secure storage. |

The tools mentioned above were essential for setting up a reliable and efficient development environment. XAMPP provided the complete local server stack necessary to run and test the application in real-time, allowing seamless debugging and responsiveness testing. phpMyAdmin offered an intuitive interface for managing the MySQL database, helping streamline tasks such as creating tables, executing SQL queries, and maintaining student data. MySQL served as the backbone of data storage, offering a structured and scalable approach to handling student records securely. Together, these tools ensured that the development process was both smooth and effective, from database design to deployment and testing.

## 2.3 Database Design and Structure

The table structure of the Student Record Management System is designed to be simple yet efficient, comprising fields such as roll number, student name, semester, and result. These attributes capture essential academic information for each student in a structured and accessible format.

To ensure data consistency and performance, the system employs primary keys and appropriate indexing techniques, which allow for faster query execution and reliable data organization. The integration of PDO (PHP Data Objects) along with prepared statements provides a secure and standardized way to interact with the database, significantly reducing the risk of SQL injection attacks.Data validation plays a critical role in maintaining the integrity of the system. It is enforced at multiple levels — on the client side using JavaScript, which prevents invalid or incomplete submissions, and on the server side using PHP, which verifies

the correctness and format of data before any insertion or update operation.This multi-layered approach to data handling ensures that records are accurate, secure, and free from redundancy or inconsistency. Additionally, the database schema is intentionally minimal, focusing only on essential data points, which makes it lightweight yet highly scalable. It can be easily extended in the future by introducing additional tables or establishing relational links to support more complex academic features such as course management or attendance tracking.

To further enhance data reliability and usability, the system supports regular backups and includes a PDF export option powered by the FPDF library. This feature enables administrators or users to generate downloadable academic reports, making the system useful for official documentation, result publishing, and archiving purposes.

**The structure is as follows:**

```
CREATE DATABASE student_db;
USE student_db;

CREATE TABLE students (
  rollno INT(11) PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  semester INT(11) NOT NULL,
  result VARCHAR(20) NOT NULL
);
```

Table 2.3: Structure of the `students` Table

| Column | Data Type | Constraint | Description |
|---|---|---|---|
| `rollno` | INT(11) | PRIMARY KEY | Unique identifier for each student |
| `name` | VARCHAR(100) | NOT NULL | Full name of the student |
| `semester` | INT(11) | NOT NULL | Current semester of the student |
| `result` | VARCHAR(20) | NOT NULL | Final result/status (e.g., Passed/Failed) |

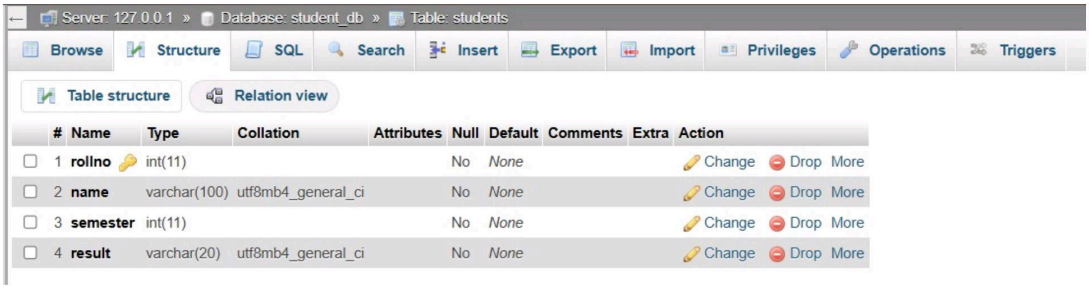



Figure 2.1: Structure of the students table viewed in phpMyAdmin

The above figure shows the structure of the `students` table created within the `student_db` database using phpMyAdmin. It displays the four columns—`rollno`, `name`, `semester`, and `result`—along with their respective data types and constraints. The `rollno` field is set as the primary key to uniquely identify each record. This visual representation in phpMyAdmin helps in quickly verifying the database schema and ensuring that the intended structure has been implemented correctly.

### 2.3.1 Database Operations

Once the `students` table was created, several SQL operations were implemented to manage student data efficiently. These operations included:

- **Insertion of Records:** New student entries were added through a web form that interacted with the backend using PHP and executed SQL `INSERT` queries.
- **Filtering and Searching:** The application supported searching records based on roll number, semester, or result using SQL `SELECT` queries with `WHERE` conditions.
- **PDF Generation:** Filtered student data could be exported into PDF format using the FPDF PHP library for reporting purposes.
- **Data Integrity and Security:** PDO (PHP Data Objects) was used for all database interactions to prevent SQL injection and ensure secure and reliable communication with the MySQL database.

This integration between the database and the web interface allowed for smooth data handling, user-friendly interaction, and accurate reporting during the internship project.

This chapter provided an in-depth overview of the core technologies, tools, and database structure used throughout the internship project. The selection of appropriate web technologies, combined with a structured database and secure implementation practices, ensured a robust and scalable student record management system. These technical foundations played a crucial role in successfully achieving the objectives of the project.

# CHAPTER 3

## Internship Task Allocation

The internship provided a valuable opportunity to apply theoretical knowledge in a real-world project environment. During the internship period, I was entrusted with responsibilities that covered various stages of software development, from planning and design to coding and testing. This hands-on experience helped me to better understand industry practices, enhance technical skills, and develop a structured approach to solving problems. The tasks were carefully divided into logical segments to ensure smooth progress and measurable outcomes throughout the development process.

## 3.1 Overview of Assigned Tasks

During the course of the Academic Summer Internship at NIELIT Guwahati, the assigned tasks were structured to enhance both theoretical understanding and practical skills in web development. The internship was divided into multiple phases, each focusing on different aspects of modern web technologies and development workflows.

Table 3.1: Weekly Task Allocation

| WEEK | ACTIVITIES |
|---|---|
| WEEK 1 | Familiarization with the internship objectives, understanding the project requirements, and setting up the development environment. Refreshed key front-end technologies including HTML, CSS, and JavaScript. |
| WEEK 2 | Introduction to back-end technologies such as PHP, XAMPP, and MySQL. Explored basic server-side scripting, form handling, and database connectivity. Also learned how to use phpMyAdmin to manage databases. |
| WEEK 3 | Developed the final group project: a Web-Based Student Record Management System. Tasks included designing the UI, implementing form validation, database integration using PHP MySQL, and building a query module with PDF export functionality. |

## 3.2 Nature of Work

The nature of work undertaken during the Academic Summer Internship at NIELIT Guwahati primarily involved practical, hands-on development of a functional web application, along with exposure to various aspects of modern software development practices. The internship focused on both front-end and back-end development components, enabling a full-stack understanding of how dynamic web systems operate. Interns were also introduced to version control systems

and deployment environments, which helped simulate real-world development workflows. Collaborative coding, debugging sessions, and regular feedback cycles further enhanced technical proficiency and project-based learning.

The core project is a Web-Based Student Record Management System, developed using technologies such as HTML, CSS, JavaScript, PHP, and MySQL. Initial tasks included understanding the project requirements, designing a responsive user interface, and setting up a local development environment using XAMPP and phpMyAdmin.
This comprehensive experience allowed for the development of both technical skills and soft skills such as problem-solving, debugging, time management, and teamwork within a collaborative project environment.

## 3.3 Skills Developed

The Academic Summer Internship at NIELIT Guwahati offered a valuable platform to strengthen technical competencies and professional aptitude through practical exposure and structured mentoring sessions. By actively developing the Web-Based Student Record Management System, several key skills in web technologies, database handling, and project execution were effectively cultivated. The process also encouraged analytical thinking, efficient code structuring, and the use of best practices in development. Overall, the internship bridged the gap between academic learning and industrial expectations, fostering a deeper understanding of real-world project development.

**Technical Skills**

**1. Front-End Development:** Gained hands-on experience in designing responsive and user-friendly web interfaces using HTML, CSS, and JavaScript. Developed a clear understanding of layout structuring, styling techniques, and client-side scripting for interactivity.
**2. Back-End Development:** Learned server-side scripting using PHP to manage form validation, data processing, and integration with databases. Focused on writing modular and secure code using PDO for database access.
**3. Database Management:** Worked with MySQL to store, manage, and retrieve structured data efficiently. Gained practical experience in writing SQL queries and managing relational tables.
**4. PDF Report Generation:** Used the FPDF PHP library to dynamically generate downloadable PDF reports from database records, enhancing report automation within the application.
**5. Development Tools:** Used XAMPP for local server management and Visual Studio Code (VS Code) for development and debugging, improving productivity and code organization.

**Professional Skills**

**1. Problem-Solving:** Improved analytical skills by identifying and resolving bugs, optimizing code, and addressing design issues throughout the project development cycle.
**2. Team Collaboration:** Collaborated with peers by sharing responsibilities, integrating individual code modules, and maintaining clear communication for effective teamwork.
**3. Project Management:** Followed a disciplined approach by setting goals, managing time efficiently, and ensuring timely completion of assigned tasks and deliverables.
**4. Communication:** Enhanced verbal and written communication through documentation, peer discussions, and task presentations.
**5. Adaptability:** Quickly adapted to new tools and frameworks during the internship, improving flexibility and continuous learning ability.

## 3.4 Summary of Internship Experience

This chapter detailed the practical aspects of the internship, highlighting the technical and professional skills developed through active participation in a real-world project. The hands-on exposure to various phases of software development—from requirement analysis and interface design to server-side scripting and database integration—provided a comprehensive understanding of full-stack web development.Working on the Student Record Management System allowed me to apply theoretical knowledge in a structured and collaborative environment. The experience not only enhanced my proficiency in tools such as HTML, CSS, JavaScript, PHP, and MySQL but also deepened my understanding of best practices in secure coding, data validation, and user experience design.

Additionally, the internship played a vital role in strengthening my problem-solving abilities, communication skills, and time management. Interactions with mentors and peers at NIELIT Guwahati fostered a professional work ethic and a proactive approach to learning. Overall, the internship significantly contributed to my academic growth and has better prepared me for future industry engagements and more complex software projects

# CHAPTER 4

## Introduction to System Design

System design is a vital stage in the software development life cycle where the overall structure and functionality of the system are planned in detail, including the architecture, components, interfaces, and data management strategies to meet the specified requirements efficiently and reliably, In the case of the Student Record Management System, the design phase provided a clear foundation for development by mapping out how different modules interact, how data flows through the system, and how users engage with various functionalities, ultimately ensuring the creation of a robust, user-friendly, scalable, and maintainable solution.

## 4.1 System Architecture

The architecture of the Student Record Management System follows a standard three-tier architecture, which separates the system into three main layers: the presentation layer, the application layer, and the data layer. This structure enhances modularity, scalability, and maintainability, making the system more efficient and easier to debug and upgrade.

**Presentation Layer (Client Side):** This is the user-facing layer of the system, developed using HTML, CSS, and JavaScript. It allows users to interact with the system through web pages like index.php for data entry and query.php for querying student records.

**Application Layer (Server Side):** This layer contains the core logic of the application and is developed using PHP. It handles form submissions, validates user inputs, interacts with the database, and generates dynamic responses. It acts as a bridge between the user interface and the database.

**Data Layer (Database Server):** The data layer consists of a MySQL database that stores all student records. The database is designed to ensure data integrity, enforce constraints, and support secure storage and retrieval of information. SQL queries are executed from the server-side scripts to interact with this layer.

The layers interact via standard HTTP requests and responses.The client sends requests through forms or AJAX, PHP handles processing and database communication, and responses are returned. This structure ensures clear separation and easier maintenance.

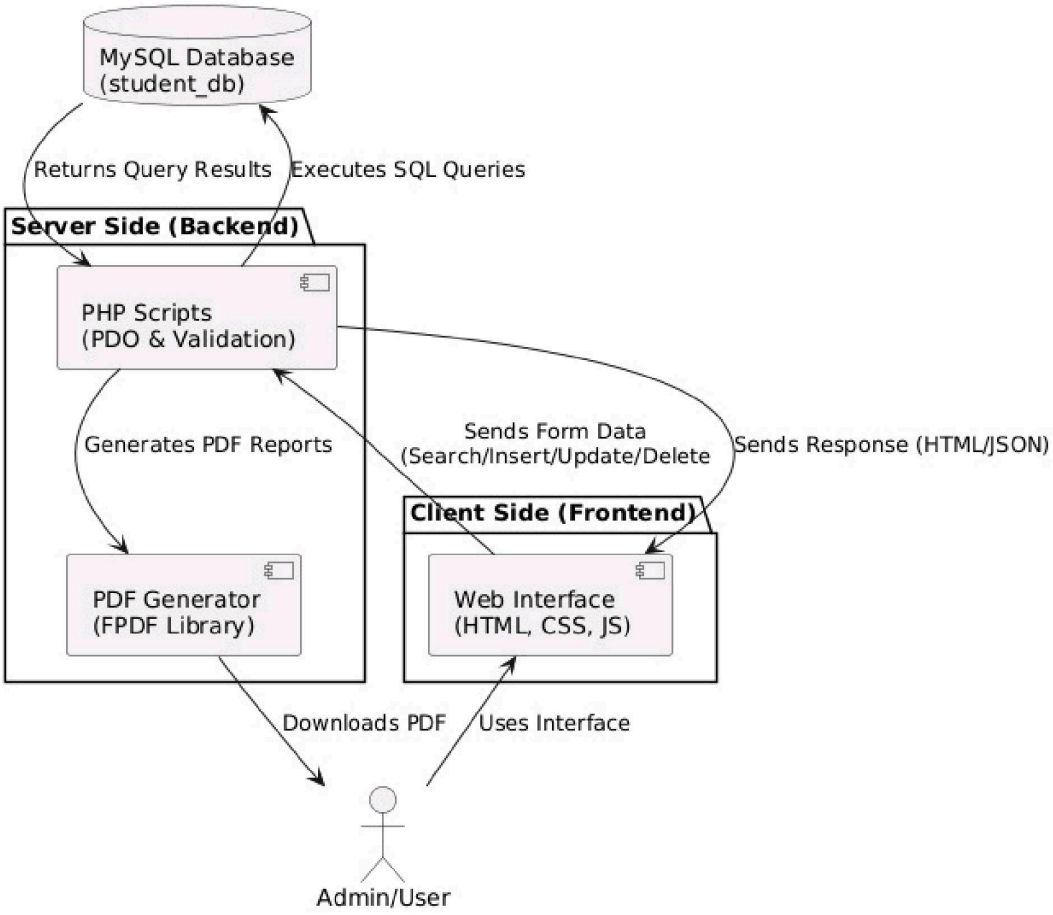
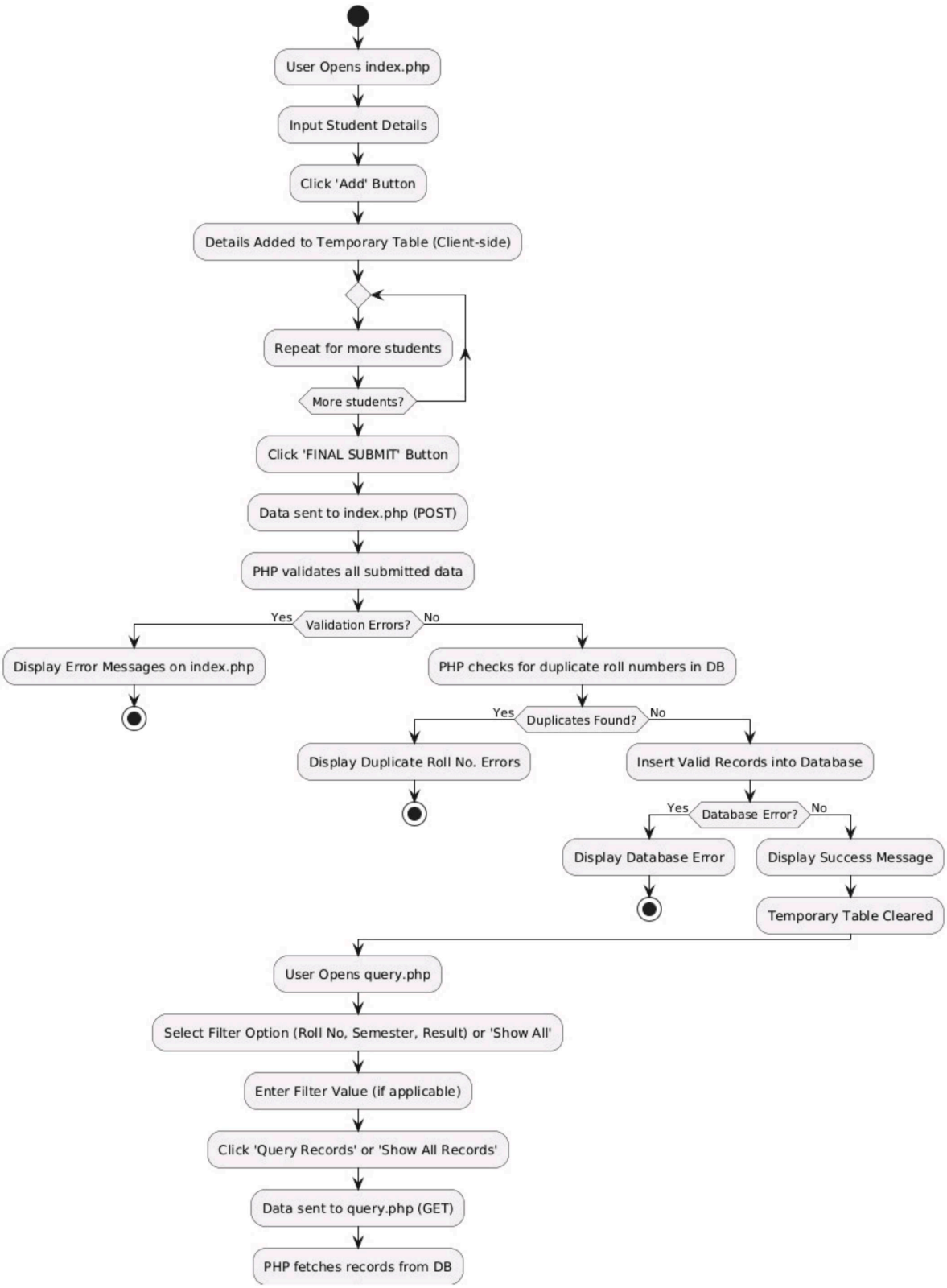



Figure 4.1: System Architecture: Student Record Management System

## 4.2 Use Case Diagram

The Use Case Diagram demonstrates the various interactions between the user and the system. It outlines the main functionalities provided by the Student Record Management System, such as adding student records, querying existing data, updating or deleting entries, and downloading PDF reports. Each use case represents a discrete action that a user can perform, contributing to the overall functionality and usability of the system.

This diagram serves as a high-level visual representation that clarifies the boundaries of the system and the user's role in interacting with different modules. It ensures that the design remains user-centric and task-oriented. Additionally, it facilitates better communication among developers, stakeholders, and users by clearly specifying what the system is supposed to do.By visualizing all interactions in one place, the Use Case Diagram becomes an essential tool in system planning, requirement validation, and further development phases.It ensures that all essential operations are accounted for, and no critical user activity is overlooked, ultimately leading to a more robust and reliable application design.


User Opens index.php
Input Student Details
Click 'Add' Button
Details Added to Temporary Table (Client-side)
Repeat for more students
More students?
Click 'FINAL SUBMIT' Button
Data sent to index.php (POST)
PHP validates all submitted data
Validation Errors?
Yes
No
Display Error Messages on index.php
PHP checks for duplicate roll numbers in DB
Duplicates Found?
Yes
No
Display Duplicate Roll No. Errors
Insert Valid Records into Database
Database Error?
Yes
No
Display Database Error
Display Success Message
Temporary Table Cleared
User Opens query.php
Select Filter Option (Roll No, Semester, Result) or 'Show All'
Enter Filter Value (if applicable)
Click 'Query Records' or 'Show All Records'
Data sent to query.php (GET)
PHP fetches records from DB

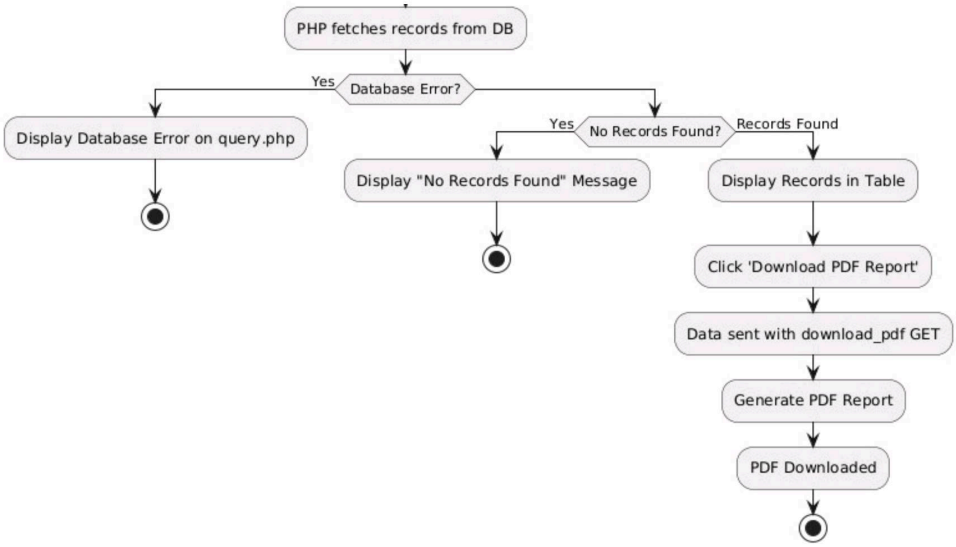



Figure 4.2: Use Case Diagram of the Student Record Management System

## 4.3 Entity-Relationship (ER) Diagram

The Entity-Relationship (ER) Diagram represents the logical structure of the database used in the Student Record Management System. It defines the key entity **Student** along with its attributes: `rollno`, `name`, `semester`, and `result`. The attribute `rollno` is the primary key that uniquely identifies each student.

**To ensure data integrity, the following constraints are applied:**

- `name` accepts only letters and spaces.
- `rollno` must be between 1000 and 9999.
- `semester` must be an integer between 1 and 8.
- `result` can only be "Passed" or "Failed".

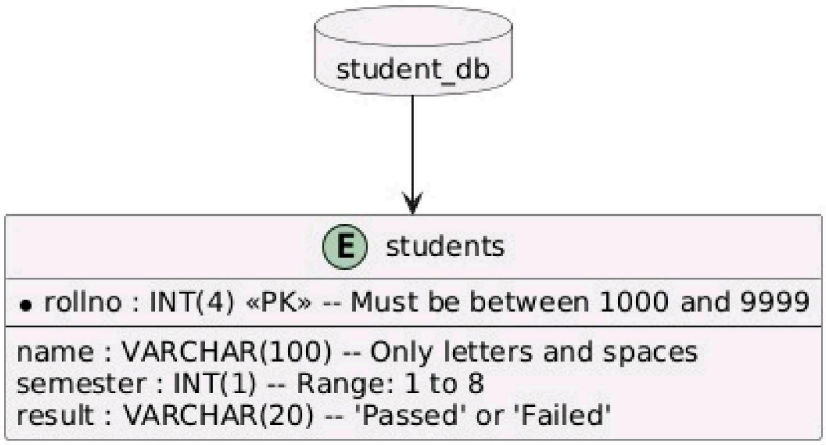



Figure 4.3: ER Diagram of the Student Record Management System

## 4.4 Database Design

The system uses a relational database named `student_db` to store and manage student records. It includes a single table named `students`, which captures essential academic details of each student.

Table 4.1: Database Table Design for `students`

| Field (Data Type) | Constraints and Description |
|---|---|
| `rollno (INT)` | Primary key, must be unique. Accepts only numeric values ranging from 1000 to 9999. Used as the unique identifier for each student. |
| `name (VARCHAR(100))` | Full name of the student. Accepts only alphabetic letters and spaces; no digits or special characters allowed. |
| `semester (INT(11)` | Indicates the student's current semester. Must be an integer between 1 and 8. |
| `result (VARCHAR(20)` | Specifies academic status. Can only be either `'Passed'` or `'Failed'`. |

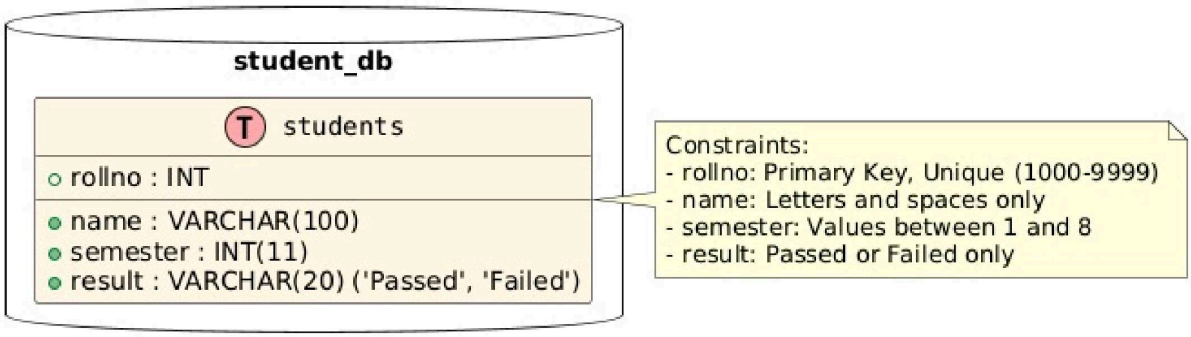



Figure 4.4: Database Design: students table in student_db

This structured database schema not only ensures efficient data storage but also enforces strict validation rules, thereby maintaining the accuracy, consistency, and integrity of student academic records throughout the system.

## 4.5 Data Flow Diagram

The Data Flow Diagram (DFD) provides a visual representation of how data moves through the system. It highlights the processes, external entities, data stores, and data flows involved.

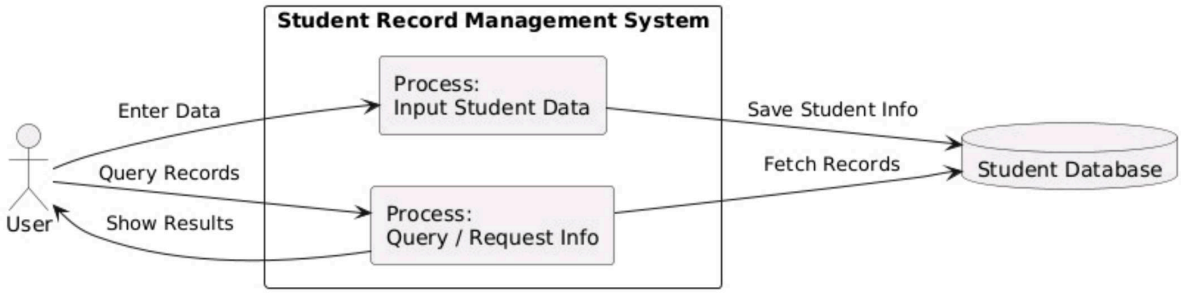



Figure 4.5: Data Flow Diagram of the Student Record Management System

Data Flow Diagram (DFD) illustrates how data flows through the Student Record Management System. This structure enhances the clarity of operations, supports modular system design, and provides a foundation for debugging and future development.

Table 4.2: Summary of System Design Contributions

| **Component** | **Contribution to Reliability and Efficiency** |
|---|---|
| Student Data Entry | Students' academic records are submitted via a front-end form and processed by the server, ensuring structured data input. |
| Validation Module | Ensures the input data (e.g., roll number, semester, result) meets the required format and constraints, minimizing data entry errors. |
| Database Interaction | Validated data is stored or retrieved from the MySQL database using optimized queries, supporting fast access and secure storage. |
| Admin/User Interaction | Admins or users can query, view, or export student records based on filters, enhancing usability and data accessibility. |

## 4.6 Summary

The design of the Student Record Management System has been guided by principles aimed at maximizing reliability and efficiency. The system adopts a client-server architecture that separates concerns, enabling better performance and easier maintenance. Use case modeling ensures that all user interactions are clearly defined, reducing ambiguity and minimizing errors during development. The Entity-Relationship (ER) Diagram and database design enforce data consistency through primary keys, constraints, and well-defined data types. These choices help maintain the integrity of academic records and reduce redundancy. The use of a structured data flow model enhances clarity in how information moves through the system, making the system more predictable and easier to debug or expand in the future.

# CHAPTER 5

## Overview of the Development

This chapter outlines the practical implementation of the Student Record Management System developed during my Summer Internship at NIELIT Guwahati. It covers the tools, technologies, and frameworks utilized in the development process, along with the major features implemented in the system. The objective of this chapter is to provide a detailed overview of the system's design and functionality, highlighting how various components were integrated to achieve a responsive and user-friendly interface. It includes a discussion on the technology stack used, the core functionalities implemented, and a brief summary.

## 5.1 Technology Stack Used

### 5.1.1 Frontend Tools: HTML5, CSS3, JavaScript

The frontend of the Student Record Management System was developed using standard web technologies to ensure a responsive and user-friendly interface.

- **HTML5:** Used to define the structure and content of the web pages. It provides semantic elements and a clean markup to build the layout of the forms, tables, and dashboard components.
- **CSS3:** Employed to style the visual elements of the interface, ensuring consistency in layout, colors, spacing, and responsive behavior across devices. Custom styles were used to enhance usability and visual clarity.
- **JavaScript:** Utilized for client-side scripting to add interactivity, such as form validation, dynamic filtering of data, confirmation prompts before deletion, and improving the overall user experience without frequent page reloads.

### 5.1.2 Backend Technologies: PHP with PDO

The backend logic of the system was implemented using PHP, a server-side scripting language that enabled dynamic data processing and interaction with the database.

- **PHP:** Served as the core backend language to handle user requests, process form submissions, perform CRUD operations (Create, Read, Update, Delete), and control the overall logic of the application.

- **PDO (PHP Data Objects):** Used for secure and efficient interaction with the MySQL database. PDO provides a data-access abstraction layer, allowing prepared statements that protect against SQL injection, making the system more robust and secure.

### 5.1.3 Database: MySQL

MySQL was used as the backend relational database management system to store, retrieve, and manage student records efficiently.

- **Structure:** The database consisted of normalized tables with fields such as Student Name, Roll Number, Registration Number, Semester, and Result, ensuring data consistency and integrity.
- **Functionality:** It enabled secure data storage, retrieval based on filters, and update/delete operations through the PHP backend using PDO.
- **Reliability:** MySQL was chosen for its reliability, scalability, and compatibility with PHP, making it an ideal choice for handling academic records in a structured format.

### 5.1.4 PDF Generation: FPDF

The FPDF library (Free PDF) was used to implement PDF generation functionality within the Student Record Management System.

- **Purpose:** To allow administrators or users to export filtered student academic records into a professional and printable PDF format.
- **Integration:** The library was integrated using PHP, where data fetched from the MySQL database was dynamically formatted and written into a structured PDF report.
- **Features:** The generated PDFs included headers, tabular data, and aligned formatting to improve readability and support for offline access and documentation.
- **Output:** Exported PDF files could be downloaded directly through the user interface, improving data portability and administrative convenience.

### 5.1.5 Local Server Setup: XAMPP

XAMPP was utilized as the local development environment to run the web-based Student Record Management System during the internship.

- **Purpose:** To host the application locally and test all features without deploying it to a live server.

- **Components Used:**
  - **Apache:** To run the PHP files and serve the web application.
  - **MySQL:** To manage and store student records in a relational database.
  - **phpMyAdmin:** To visually interact with the MySQL database.
- **Advantages:** Provided a complete package for development and testing in a controlled, offline environment. It also helped in simulating real server behavior.

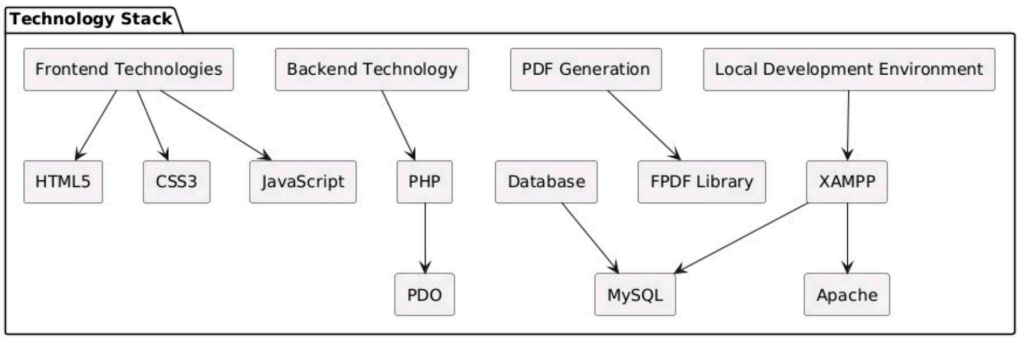



Figure 5.1: Technology Stack Diagram of the Student Record Management System

## 5.2 Core Features Implemented

The Student Record Management System incorporates several essential features to facilitate the efficient handling of academic data. The primary functionalities developed during the internship include:

- **Add Student Record**
  A user-friendly form-based interface is provided for administrators to input new student details, including name, roll number, department, semester, and marks. Server-side validation is implemented using the PHP PDO method to ensure data integrity prior to submission, as illustrated in Fig. 5.2.
- **Update & Delete Operations**
  To maintain the accuracy and integrity of student records, update and delete functionalities are integrated within a client-server architecture. This design minimizes data entry errors by ensuring that all operations are validated both on the client and server sides. Users can update existing student details through a pre-filled form interface, reducing the chances of incorrect entries. Similarly, deletion requires confirmation before removing a record from the database, preventing accidental loss of data. The interfaces for these operations are illustrated in Fig. 5.3.

- **View & Search Records (with Filters)**
  All stored student records are displayed in a structured tabular format. Users can search for specific entries using filter options such as name, roll number, or semester. This functionality enhances accessibility and significantly reduces the time required to locate individual records, as illustrated in Fig. 5.4.

- **Generate PDF Report**
  Using the FPDF library, filtered records can be exported into a downloadable PDF format. This feature is useful for reporting, documentation, or academic evaluation purposes.

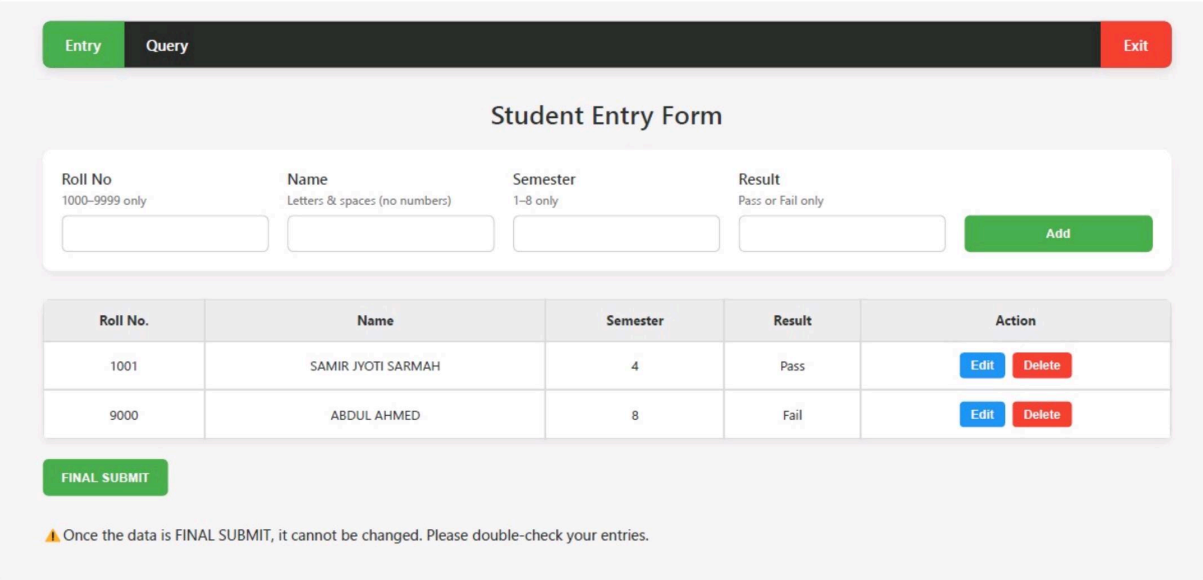
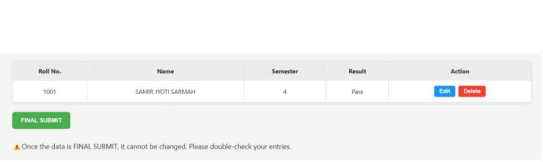
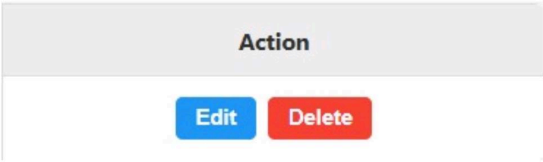



Figure 5.2: Interface of Adding Records



(a) Figure 1: Edit & Delete Interface




(b) Figure 2: Action Panel

Figure 5.3: Interface for Editing and Deleting Records

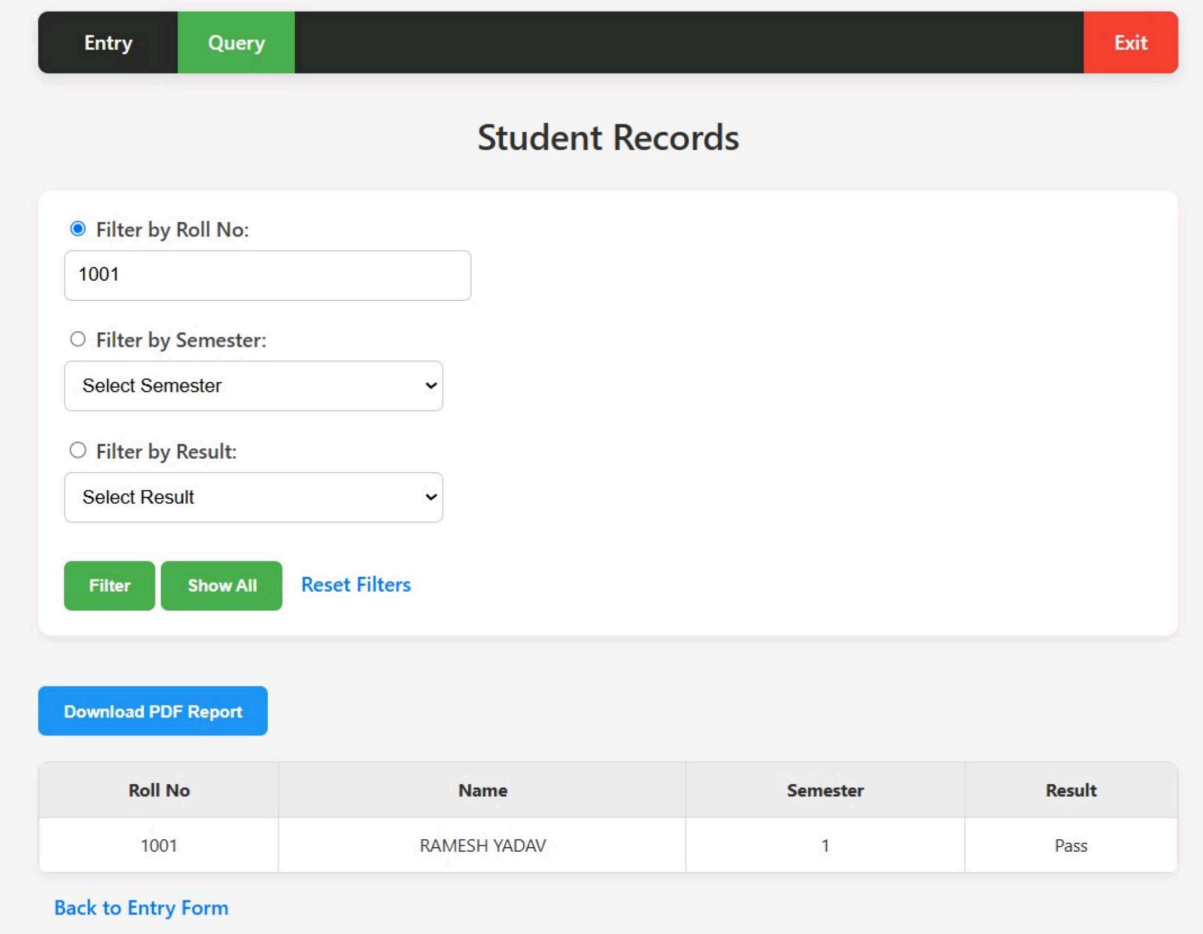



Figure 5.4: Interface of View Search Records (with Filters)

## 5.3 Summary

The development of the Web-Based Student Record Management System followed a structured and phased approach, aligning with the objectives set during the academic summer internship at NIELIT Guwahati. The project was built using a combination of front-end and back-end technologies such as HTML, CSS, JavaScript, PHP, and MySQL, ensuring both usability and performance. The development process began with user interface design and form validation, followed by the integration of server-side logic for secure data processing using PDO. A query module was implemented to allow efficient retrieval and filtering of records, along with a PDF export feature using the FPDF library. Each component was designed to support scalability, interactivity, and a responsive layout to ensure usability across devices. Emphasis was placed on building a user-friendly experience, validating inputs thoroughly, and creating a clean, minimal interface similar to professional applications. Through this project, practical knowledge of database connectivity, form handling, user input validation, and dynamic web content generation was significantly strengthened. Overall, the development phase demonstrated a complete software development cycle—from planning and interface design to backend integration and testing—resulting in a functional and professional-grade student record system.

# CHAPTER 6

## Result and Learnings

### 6.1 Result

The completion of this project led to the successful development of a fully functional **Web-Based Student Record Management System**, tailored to meet the academic record-keeping needs of educational institutions. The system enables efficient management of student data, including roll number, name, semester, and result status (Pass/Fail), with features such as:

- Dynamic form input and manipulation before submission.
- Real-time client-side validation and secure server-side validation using PDO.
- Search and filter functionality for record retrieval based on multiple criteria.
- PDF report generation using the FPDF library for easy documentation.

The interface was designed to be user-friendly and responsive, ensuring accessibility across various devices and screen sizes. The final system provides a reliable, efficient, and professional solution that reduces manual workload and increases data accuracy.
This outcome reflects not only the successful implementation of theoretical concepts learned during the academic curriculum but also the development of a practical solution that could be deployed in real educational environments.

### 6.2 Discussion

The development of the *Web-Based Student Record Management System* during the academic summer internship at **NIELIT Guwahati** served as an enriching learning experience that combined theoretical knowledge with practical implementation. The project provided an opportunity to explore the full stack of web development—from UI design using HTML and CSS to scripting with JavaScript and server-side programming with PHP and MySQL. One of the key challenges faced during development was ensuring seamless interaction between the client-side and server-side, especially while maintaining data integrity and validation. This was addressed through rigorous testing and implementation of secure coding practices such as prepared statements using **PDO** to prevent SQL injection. The integration of the **FPDF** library for generating structured reports in PDF format added a professional touch to the system and simulated a real-world application scenario. Furthermore, designing a responsive layout helped in understanding cross-browser compatibility and UI responsiveness—an essential skill in modern web development.

Through this project, I gained exposure to real development tools like **XAMPP**, and learned to manage folder structures, organize scripts, and maintain clean, modular code. The overall experience bridged the gap between academic learning and software development practices in the industry.

## 6.3 Key Takeaways and Learning Experiences

The Summer Internship at **NIELIT Guwahati** provided a valuable platform to translate academic concepts into a real-world application. The development of the **Web-Based Student Record Management System** enabled the team to acquire a wide range of technical and soft skills essential for future endeavors in software development.

**Key Takeaways:**

- Web Development Experience:
  Gained hands-on experience in both front-end (HTML, CSS, JavaScript) and back-end (PHP, MySQL, PDO) development, enhancing our understanding of complete web application workflows.
- Database Management:
  Understood the importance of relational database design, SQL queries, and secure data insertion techniques using prepared statements.
- PDF Report Generation:
  Learned how to generate professional documents using the FPDF library—useful in real-world applications for generating receipts, certificates, and reports.
- Overall Outcome:
  The internship significantly enhanced our problem-solving abilities, teamwork, and adaptability, equipping us with industry-relevant skills and preparing us for future academic and professional challenges.

# CHAPTER 7

## Conclusion

### 7.1 Summary of the Training and Internship Experience

The academic summer internship at NIELIT Guwahati was a highly enriching and practical learning experience.It was undertaken as a part of the UG 4th Semester curriculum of the Department of Computer Science and Information Technology, Cotton University. The four-week program combined theoretical learning, hands-on development, and exposure to modern software tools and technologies in a structured format, internship supervisor has assigned the task of developing a Web-Based Student Record Management System. This project allowed us to apply our knowledge of web technologies such as HTML, CSS, JavaScript, PHP, and MySQL to build a functional and user-friendly application.Worked on designing a responsive UI and implemented secure back-end operations using PHP with PDO. We also learned to generate PDF reports using the FPDF library, allowing us to go through the entire development lifecycle. In addition to the main project, the internship featured a bootcamp on cloud computing and introductory sessions on backend development. These sessions broadened our understanding of current industry tools and practices.

Guidance from mentors at NIELIT Guwahati played a crucial role in refining our problem-solving skills, improving code efficiency, and instilling the importance of professional software development standards.

Overall, the internship experience effectively bridged the gap between classroom knowledge and real-world application. It gave us a solid foundation for future academic growth and career pursuits in the field of software development.

### 7.2 Future Enhancements and Career Insights

While the current version of the Web-Based Student Record Management System meets its primary objectives of student data entry, validation, querying, and PDF report generation, there is significant scope for future enhancements. Features such as user authentication and role-based access control can be added to ensure secure usage by administrators, teachers, or departmental staff. Additionally, integration with cloud-based databases could improve scalability and accessibility across institutions.

A dashboard with real-time statistics (e.g., number of students per semester, pass/fail

ratio) and graphical data visualization can also enhance usability for academic analysis. Other suggested improvements include bulk upload via Excel/CSV, email notifications, and mobile responsiveness through a dedicated mobile-first design.

From a career perspective, this internship offered valuable insights into real-world software development. It reinforced our understanding of front-end and back-end integration, improved our skills in PHP, MySQL, and client-side scripting, and introduced us to tools such as FPDF, XAMPP, and version control practices. This practical exposure not only boosted our technical proficiency but also clarified our interest in full-stack development, UI/UX design, and project management.

The experience laid a strong foundation for pursuing future opportunities in web development, internships, and research projects in the field of Computer Science and Information Technology.

## References


[1] W3Schools, *CSS Tutorial*, Available at: https://www.w3schools.com/css/

[2] W3Schools, *JavaScript Tutorial*, Available at: https://www.w3schools.com/js/

[3] Code With Harry, *Tutorials* , Available at: https://www.codewithharry.com/

[4] PHP Documentation, *PHP Manual*, Available at: https://www.php.net/manual/en/

[5] MySQL Developers, *MySQL 8.0 Reference Manual*, Available at: https://dev.mysql.com/doc/

[6] FPDF Team, *FPDF – PHP Class to Generate PDF Files*, Available at: http://www.fpdf.org/

[7] Apache Friends, *XAMPP Official Page*, Available at: https://www.apachefriends.org/index.html

[8] phpMyAdmin, *phpMyAdmin Official Documentation*, Available at: https://docs.phpmyadmin.net/

[9] Plantuml, *Diagram and Flow-chart making tool*, Available at: https://plantuml.com/

[10] LaTeX, *Documentation* , Available at: https://www.overleaf.com/learn/latex

[11] OpenAI, Available at: https://openai.com/


[0]Soft Copy: https://github.com/samirjsarmah/NIELIT-Summer-Internship-Report