**PROJECT REPORT**

**ON**

**STUDENT MANAGEMENT SYSTEM**

 **Carried Out at**

**CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING**

**Knowledge Park, BANGALORE**

**UNDER THE SUPERVISION OF**

**Mr. N Shanmuganathan**

**C-DAC Bangalore**

**Submitted By**

|  |  |
| --- | --- |
| **Student Name** | **PRN No.** |
| Sneha Bhalchandra Jagtap | 220350120101 |
| Sumedha Keshav Bhalerao | 220350120106 |
| Tejas Sachin Chaudhari | 220350120107 |
| Tejas Vishvajeet Bhosale | 220350120108 |
| Tushar Kumar Patel | 220350120110 |

**Candidate’s Declaration**

We hereby certify that the work being presented in the report entitled **Student Management System**, in partial fulfilment of the requirements for the award of PG Diploma Certificate and submitted in the department of PG-DAC of the C-DAC Bangalore, is an authentic record of our work carried out during the period, 1st August 2022 to 25 September 2022 under the supervision of **Mr. N Shanmuganathan** C-DAC Bangalore.

**(Name of Candidate)**

|  |  |
| --- | --- |
| **Student Name** | **PRN No.** |
| Sneha Bhalchandra Jagtap | 220350120101 |
| Sumedha Keshav Bhalerao | 220350120106 |
| Tejas Sachin Chaudhari | 220350120107 |
| Tejas Vishvajeet Bhosale | 220350120108 |
| Tushar Kumar Patel | 220350120110 |

**Counter Signed by**

--------------------

**ACKNOWLEDGMENT**

This project “**Student Management System**” was truly a great learning experience for us and we are submitting this work to Advanced Computing Training School (CDAC ACTS).

We are very glad to mention **Mr. N Shanmuganathan** for his valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

We are highly grateful to Mr. N Shanmuganathan, Joint Director of ACTS training Centre, C-DAC, for his guidance and support whenever necessary during the course of our journey to acquire Post Graduate Diploma in Advanced Computing (PG-DAC) through C-DAC ACTS, Bangalore. Our heartfelt thanks go to Ms. Uma Prasad our Course Coordinator, PG-DAC who gave all the required support and kind coordination to provide all the necessities.

We acknowledge here out debt to those who contributed significantly to one or more steps. We take full responsibility for any remaining sins of omission and commission.

**CERTIFICATE**

This is to certify that the work titled “**Student Management System”** is carried out by Sneha Bhalchandra Jagtap (220350120101), Sumedha Keshav Bhalerao (220350120106), Tejas Sachin Chaudhari (220350120107), Tejas Vishvajeet Bhosale (220350120108), Tushar Kumar Patel (220350120110) the Bonafide Students of Diploma in Advanced Computing Knowledge Park, Bangalore from 1st August 2022 to 25 September 2022. The Course End Project work is carried out under my direct supervision and completed.

**Mr. N Shanmuganathan**

C-DAC Knowledge Park,

Bangalore - 560100, India

**ABSTRACT**

Student Management System (SMS) provides a simple interface for maintenance of student information. It can be used by educational institutes or colleges to maintain the records of students easily. The creation and management of accurate, up-to-date information regarding a students’ academic career is critically important in the university as well as colleges. Student information system deals with all kind of student details, academic related reports, institute details, course details, curriculum, batch details and other resource related details too. It tracks all the details of a student from the day one to the end of the course which can be used for all reporting purpose, tracking of attendance, progress in the course, exam details, project or any other assignment details, final exam result and all these will be available through a secure, online interface embedded in the college’s website. Different reports and Queries can be generated based on vast options related to students, batch, course, faculty, exams, semesters, certification and even for the entire institute.

In the software we can register as a user and user has two types student and administrator. Administrator has the power to add new user and can edit the students details entered. An admin can add students record ,attendance status with department wise. All students can search his/her basics details and attendance status with their respective roll numbers

**TABLE OF CONTENTS**

[Chapter One. Introduction 10](#_Toc115306503)

[1.1 Introduction 10](#_Toc115306504)

[Chapter Two. Software Requirement Specification 11](#_Toc115306505)

[2.1. Product Perspective 11](#_Toc115306506)

[2.2. Product functions 11](#_Toc115306507)

[2.2.1 Student Related Functions 11](#_Toc115306508)

[2.2.2 Admin Related Functions 13](#_Toc115306509)

[2.2.3 Common Functions 14](#_Toc115306510)

[2.3 Operating Environment 14](#_Toc115306511)

[2.3.1 Client Side*:* 14](#_Toc115306512)

[2.3.2 Server Side: 14](#_Toc115306513)

[2.3.3 Design and Implementation Constraints 15](#_Toc115306514)

[Chapter Three: Tools and Technologies Used 16](#_Toc115306515)

[3.1. React JS 16](#_Toc115306516)

[3.2. Java Spring Boot 17](#_Toc115306517)

[3.3. MySQL 18](#_Toc115306518)

[3.4. JAVASCRIPT 18](#_Toc115306519)

[3.5. JAVA 19](#_Toc115306520)

[3.6. JAVA EE 19](#_Toc115306521)

[3.7. Eclipse IDE 20](#_Toc115306522)

[3.8. Visual Studio Code 21](#_Toc115306523)

[3.9. Postman 21](#_Toc115306524)

[Chapter Four. Application and Description Administrators Detailed Attribute 23](#_Toc115306525)

[4.1. Admin Attributes 23](#_Toc115306526)

[4.1.1. Admin Login : 23](#_Toc115306527)

[4.1.2. Upload Student Data : 24](#_Toc115306528)

[4.1.3. Upload Attendance 24](#_Toc115306529)

[4.1.4. Upload Module Performance : 24](#_Toc115306530)

[4.1.5. Add Notice For Students : 24](#_Toc115306531)

[4.1.6. Review Of Enquiry: 24](#_Toc115306532)

[4.2 Student 24](#_Toc115306533)

[4.2.1. Student Login : 24](#_Toc115306534)

[4.2.2. Student Profile : 25](#_Toc115306535)

[4.2.3. View Attendance : 25](#_Toc115306536)

[4.2.4. View Module Performance : 25](#_Toc115306537)

[4.2.5. View Notice: 25](#_Toc115306538)

[4.2.6 Enquiry: 25](#_Toc115306539)

[Chapter Five. Database Design and GUI 26](#_Toc115306540)

[5.1. Database 26](#_Toc115306541)

[5.2 Use Case Diagram : 28](#_Toc115306542)

[5.2.1 Admin 28](#_Toc115306543)

[5.2.2 Student 28](#_Toc115306544)

[5.2.3 Sequence Diagram 29](#_Toc115306545)

[5.3. LOGIN PAGE 29](#_Toc115306546)

[5.4. Student Profile 30](#_Toc115306547)

[5.5. Attendance 31](#_Toc115306548)

[5.6. MODULE 32](#_Toc115306549)

[5.7. ID CARD 33](#_Toc115306550)

[5.8. NOTICE 34](#_Toc115306551)

[5.9. ENQUIRY 35](#_Toc115306552)

[Chapter Six. Implementation 36](#_Toc115306553)

[6.1. Description 36](#_Toc115306554)

[6.2. Implementation of functions 36](#_Toc115306555)

[6.3. Rest API 36](#_Toc115306556)

[6.4. Basic React Code 37](#_Toc115306557)

[6.5. Login Module 38](#_Toc115306558)

[6.7. Admin Services 39](#_Toc115306559)

[Chapter Seven. Testing 40](#_Toc115306560)

[7.1 TEST PLAN 1: 40](#_Toc115306561)

[7.2 TEST PLAN 2: 41](#_Toc115306562)

[Chapter Eight. Conclusion 43](#_Toc115306563)

[Chapter Nine. References 44](#_Toc115306564)

**LIST OF FIGURES**

Figure 1: Admin Login ....................................................................................................................23

Figure 2: Student Login ...................................................................................................................25

Figure 3: Entity Relationship diagram..............................................................................................26

Figure 4: Class diagram ...................................................................................................................27

Figure 5: Admin Case Diagram .......................................................................................................28

Figure 6: Student Case Diagram ......................................................................................................28

Figure 7: Sequence Diagram ............................................................................................................29

Figure 8: Login Page.........................................................................................................................29

Figure 9: Attendance ........................................................................................................................30

Figure 10: Student Profile ................................................................................................................31

Figure 11: Module ............................................................................................................................32

Figure 12: Student ID........................................................................................................................33

Figure 13: Notice...............................................................................................................................34

Figure 14: Enquiry ............................................................................................................................35

Figure 15: Service.Js..……………………………………………………………………………...36

Figure 16: Rest API…………….......................................................................................................37

Figure 17: Basic Functional Component...........................................................................................38

Figure 18: Admin Notice Code……………….................................................................................39

Figure 19: Admin Login Postman………………………………………………………………….41

Figure 20: Student Login Postman……………………………………………………………..…..42

# Chapter One. Introduction

## 1.1 Introduction

Student Management System (SMS) is a solution tool that is designed to track, maintain, manage and store all the data generated by the Institute. The main objective of this system will be to coordinate scheduling and communications between Admin, Faculty and Students. This project consists of Profile, Attendance, Course Performance Modules, Identity Card, Notices and Enquiry.

This project is mainly divided into two main categories: Admin and Students

* **Admin**
* Admin can login to the portal using his/her Email and password. He / She can view the profile page after the login. Admin can edit the Email Id and Contact Number.
* Admin can add the student’s attendance by firstly selecting the course, then selecting the module, after that he can choose the file and upload it.
* Admin can add the module report by selecting the course, module followed by date, after that admin can choose the file and upload it. Admin can edit and delete the uploaded file.
* Admin can add the notice for students. He / She needs to select the course, date then write a notice and can add it. Admin can edit and delete the added notice.
* Admin can see the enquiry sent by the student which has course name, module and description regarding the enquiry.
* **Student**
* Student can login to the portal by using his/her Email and Password. He / She can view the profile page after the login. Student can edit the Email Id and Contact Number.
* Student can view his/her attendance by selecting the month and year, after submit the monthly attendance can be displayed.
* Student can view his/her module performance which has module name followed by CCEE, internal and total marks.
* Student can view his/her Identity card which consist of Name, PRN, Course, Date of issuing and Valid up to date.
* Student can view the notice section which has Module name and Links for the same.
* Student has been provided with the Enquiry section where he/she can communicate with the admin about what issue he/she has by filling the required details.

# Chapter Two. Software Requirement Specification

## 2.1. Product Perspective

The main aim of the project student management system is to give all the data at one portal.

The project help to handle all the information of the student, attendance of the student, their module wise marks and also the notices which are provided by the administration or any other faculty. This system will also help the student to get their ID card at the time of registration itself. It will help to reduces the barrier between the student and the administration by using the enquiry portal. This will help to replace the process which are maintained and handled by manpower. The main purpose of this project is to handle different section of the Training institutes into consistent manner so the complex function can be handled smoothly and at one platform.

## 2.2. Product functions

### 2.2.1 Student Related Functions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Function ID |  | | | |
| F1 | Function | | Login | |
| Description | | Logging in respective account using Username and Password. | |
| F2 | Function | | Profile Info | |
| Description | | User can review its profile and also can edit some of the fields | |
| F3 | Function | | Attendance record | |
| Description | | Student can review his/her attendance day wise | |
| F4 | Function | | Module Performance record | |
| Description | | Student can keep record of marks obtained in each module in systematic way | |
| F5 | | Function | | ID card issuing | |
| Description | | Student will be able to get the ID card at the time of registration itself | |
| F6 | | Function | | Notice | |
| Description | | Student will be able to receive the notice from same portal | |
| F7 | | Function | | Enquiry | |
| Description | | Students can send their queries to the coordinator to clarify their doubts regarding the admission/registration | |

##### Table 1: Student Related Functions

### 2.2.2 Admin Related Functions

|  |  |  |
| --- | --- | --- |
| Function ID |  | |
| F1 | Function | Upload data |
| Description | Admin can upload the data of the student being enrolled for the specific module |
| F2 | Function | Attendance maintenance |
| Description | Can upload the attendance sheet which will be visible to the student portal |
| F3 | Function | Module marks upload |
| Description | Can upload module wise marks and can update and delete the data as well |
| F4 | Function | Notice |
| Description | Can declare a notice for the student and also can edit and delete the notice |
| F5 | Function | See Enquiries |
| Description | Can see the queries asked by the students |

##### Table 2: Admin Related Functions

### 2.2.3 Common Functions

|  |  |  |
| --- | --- | --- |
| Function ID |  | |
| F1 | Function | Login |
| Description | Logging in respective  account using Email and Password. |

**Table 3: Common Functions**

## Operating Environment

### 2.3.1 Client Side*:*

Desktop Client: - Browser based clients either in Microsoft Windows or Linux Environment.

### 2.3.2 Server Side:

Windows 10 or above Spring boot, MySQL(8.0)

Front End – ReactJS 2.3.3

Design and Implementation Constraints

This project requires an application server, MySQL (8.0), JDK 8.0 or Above, REACT 17.0.2

# Chapter Three: Tools and Technologies Used

These are the necessary tools and materials needed to build the website both the frontend and the backend. These include software and open-source materials.

**Technologies Used:**

## 3.1. React JS

React is a declarative, efficient, and flexible JavaScript library for building user interfaces. It lets you compose complex UIs from small and isolated pieces of code called “components”.

React has a few different kinds of components, but we’ll start with React.component, we use components to tell React what we want to see on the screen. When our data changes, React will efficiently update and re-render our components.

A component takes in parameters, called props (short for “properties”), and returns a hierarchy of views to display via the render method. The render method returns a *description* of what you want to see on the screen. React takes the description and displays the result. In particular, render returns a **React element**, which is a lightweight description of what to render. Most React developers use a special syntax called ***“JSX”*** which makes these structures easier to write.

React is a JavaScript-based UI development library. Facebook and an open-source developer community run it. Although React is a library rather than a language, it is widely used in web development. The library first appeared in *May 2013* and is now one of the most commonly used frontend libraries for web development. React offers various extensions for entire application architectural support, Some of the important feature of the React are*-*

**JSX** –

JSX comes with the full power of JavaScript. You can put *any* JavaScript expressions within braces inside JSX. Each React element is a JavaScript object that you can store in a variable or pass around in your program. JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.

**Components** − React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.

**License** − React is licensed under the Facebook Inc. Documentation is licensed under CC BY 4.0.

The Major React Advantages are-

* Uses virtual DOM which is a JavaScript object. This will improve apps performance, since JavaScript virtual DOM is faster than the regular DOM.
* Can be used on client and server side as well as with other frameworks.
* Component and data patterns improve readability, which helps to maintain larger apps.

## 3.2. Java Spring Boot

Java Spring Framework (Spring Framework) is a popular, open source, enterprise-level framework for creating standalone, production-grade applications that run on the Java Virtual Machine (JVM). Java Spring Boot (Spring Boot) is a tool that makes developing web application and microservices with Spring Framework faster and easier through three core capabilities:

* Autoconfiguration
* An opinionated approach to configuration
* The ability to create standalone applications

Spring Boot is an open-source Java-based framework used to create a Micro Service. It is developed by Pivotal Team. It is easy to create a stand-alone and production ready spring applications using Spring Boot.

Spring Boot is a project that is built on the top of the Spring Framework. It provides an easier and faster way to set up, configure, and run both simple and web-based applications. It is a Spring module that provides the RAD (Rapid Application Development) feature to the Spring Framework.

The Spring Framework (Spring) is an open-source application framework that provides infrastructure support for developing Java applications. One of the most popular Java Enterprise Edition (Java EE) frameworks, Spring helps developers create high performing applications using plain old Java objects (POJOs).

These features work together to provide you with a tool that allows you to set up a Spring-based application with minimal configuration and setup.

## 3.3. MySQL

MySQL is a relational database management system (RDBMS) developed by ***Oracle*** that is based on structured query language (SQL). A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or a place to hold the vast amounts of information in a corporate network.

MySQL is a free source database system, and it enables the cost-effective delivery of reliable and a high-performance and scalable Web-based and embedded database applications. It is a relational database system(RDBMS).

It is a high performing program and scalable to meet the demands of users and data MySQL is written in C and C++, so it is compatible with most of the operating systems available around the world.

Four types of database management systems

A relational database management system is one of four common types of systems you can use to manage your business data.

The other three include:

* hierarchical database systems
* network database systems
* object-oriented database systems

## 3.4. JAVASCRIPT

JavaScript is a high-level language which could be used independently or inculcated into the webpage. It can be used to, handle requests and responses and also add dynamic behaviour and also store information on a website.

JavaScript (often shortened to JS) is a lightweight, interpreted, object-oriented language with first-class functions, and is best known as the scripting language for Web pages, but it's used in many non-browser environments as well**.**

JavaScript is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else. (Okay, not everything, but it is amazing what you can achieve with a few lines of JavaScript code.)

JavaScript is used by programmers across the world to create dynamic and interactive web content like applications and browsers. JavaScript is so popular that it's the most used programming language in the world, used as a client-side programming language by

97.0% of all websites.

## 3.5. JAVA

Java can be used to build applications for a wide range of platforms. Desktops, servers, mobile phones, tablets, Blu-ray players, televisions, and web browsers all use Java, and developers can write Java-based applications for any of these platforms.

Java is platform-independent. Some programs developed in one machine can be executed in another machine. It is designed for building object-oriented applications. It is a multithreaded language with automatic memory management. It is created for the distributed environment of the Internet.

This is the main programming language used to develop the application. The java programming language was formerly developed by the sun Microsystems as proposed by James Gosling it was *first released in 1995*. It runs on the Mac OS, Windows and UNIX. Java is considered to be secure and robust**.** Also it is a multi threading and platform independent (unlike c and c++).

## 3.6. JAVA EE

Java EE is a structured application with a separate client, business, and Enterprise layers. It is mostly used to develop APIs for Desktop Applications like antivirus software, game, etc. It is mainly used for developing web applications. Suitable for beginning Java developers.

A java platform is a distinct software environment for running application written in java. The four java platforms are Java SE, Java EE, Java ME and JavaFx.

The Java EE is a superset of the java SE, and it provides a runtime environment, technologies and API for building and running enterprises web application. All API in java SE are also contained in Java EE.

Java Platform **E**nterprise **E**dition, An application software platform from Oracle based on the Java programming language. Originally developed by Sun, which Oracle acquired in 2010, Java EE services are performed in the middle tier between the user's machine and the enterprise's databases and legacy information systems. Java EE comprises a specification, reference implementation and set of testing suites. Its core component is Enterprise JavaBeans (EJBs), followed by Java Server Pages (JSPs) and Java servlets and a variety of interfaces for linking to the information resources in the enterprise.

The Java EE interfaces include JDBC for databases, JNDI for directories, JTA for transactions, JMS for messaging, JavaMail for email systems and JavaIDL for CORBA connectivity. Java Connectors are interfaces to a variety of legacy applications.

## 3.7. Eclipse IDE

Eclipse is an integrated development environment used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment.

Eclipse is written mostly in Java and its primary use is for developing Java applications. Eclipse software development kit (SDK) is free and open-source software, released under the terms of the Eclipse Public License, although it is incompatible with the GNU General Public License. It was one of the first IDEs to run under GNU Class path.

Eclipse is a free, Java-based development platform known for its plugins that allow developers to develop and test code written in other programming languages. Eclipse is released under the terms of the Eclipse Public License.

The Eclipse IDE is famous for our **Java Integrated Development Environment** (IDE), but we have a number of pretty cool IDEs, including our C/C++ IDE,

JavaScript/TypeScript IDE, PHP IDE, and more.

## 3.8. Visual Studio Code

Visual Studio Code is a source-code editor made by Microsoft for Windows, Linux and macOS.

Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.

Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

In the Stack Overflow 2021 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 70% of 82,000 respondents reporting that they use it.

## 3.9. Postman

Postman is an API (application programming interface) development tool which helps to build, test and modify APIs. Almost any functionality that could be needed by any developer is encapsulated in this tool. It is used by over 5 million developers every month to make their API development easy and simple.

Postman is an application used for API testing. It is an HTTP client that tests HTTP requests, utilizing a graphical user interface, through which we obtain different types of responses that need to be subsequently validated.

An HTTP request method includes Request Method; Request URL, Request Headers,

Request Body, Pre-request Script, and Tests. Let's talk about these Request methods one by one… Request Methods: You will find several types of Request methods in POSTMAN. Depending on your requirements or test you can choose one of them.

# Chapter Four. Application and Description Administrators Detailed Attribute

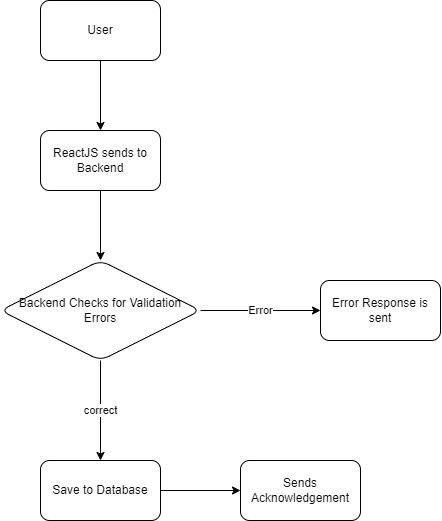
## 4.1. Admin Attributes

The responsibility of admin is to manage all the student related data.

* Admin Detailed Attributes :

### 4.1.1. Admin Login :

This refers to log in of an admin. The admin has accessibility to every section of admin by logging in with user id and password. The admin will use their user id and password to login. After logging in, ReactJs will check to see if all the fields entered by the admin are correct. If the area is not correctly filled, admin remains on the same page but if the requirements are met the data goes to the database and check the information of the admin. Then after authentication admin will get successfully logged in.



**Figure 1: Admin Login**

The administrator needs to login before they can access to student’s data.

### 4.1.2. Upload Student Data :

To manage all student related data first admin has to upload all the student data into system.

4.1.3. Upload Attendance :

After uploading Student’s data Admin can Upload Student attendance data into system.

### 4.1.4. Upload Module Performance :

Admin will also upload Student’s module performance data into system and if necessary he/she can update that data too.

### 4.1.5. Add Notice For Students :

If there is any Notice or Announcement for student then admin can add that notice by filling details in given notice form. Then all the notices will be available to the student.

### 4.1.6. Review Of Enquiry:

If student has some issues or enquiry regarding academics then student will fill the enquiry form that added enquiry data will be shown at admin side.

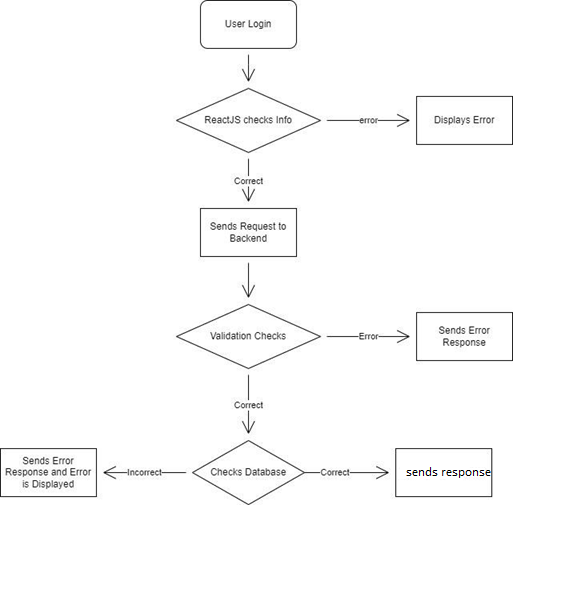
## 4.2 Student

The Student has accessibility to see his/her academic related details.

* Student Detailed Attributes :

### 4.2.1. Student Login :

This refers to log in of student. The student has accessibility to every section of Student side by logging in with user id and password. The student will use his particular PRN and password to login into the portal. After submitting, Reactjs checks if all the fields have been filled correctly. If the condition is not met the student remains on the same page but if the fields are correctly filled the student login information is sent to the database to check to check if the data entered into the areas are same as in the database. If it is correct then student will redirect to its profile page and access all the student side section.



**Figure 2: Student Login**

### 4.2.2. Student Profile :

Student can see own data after successful login and student have access to update his/her own data.

### 4.2.3. View Attendance :

Student’s own attendance data will be shown to student who is logged in.

### 4.2.4. View Module Performance :

Student’s own Module Performance data will be shown to student who is logged in.

### 4.2.5. View Notice:

All the notice uploaded by admin will be view by the student.

### 4.2.6 Enquiry:

If student has some issues or enquiry regarding academics then student will fill the enquiry form that added enquiry data will be shown at admin side.

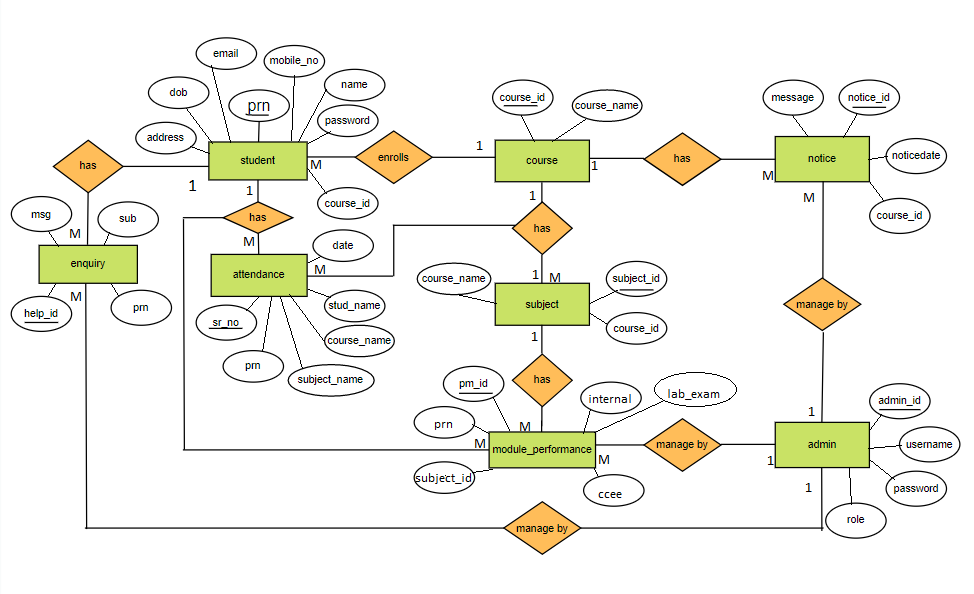
# Chapter Five. Database Design and GUI

Design for database and GUI application

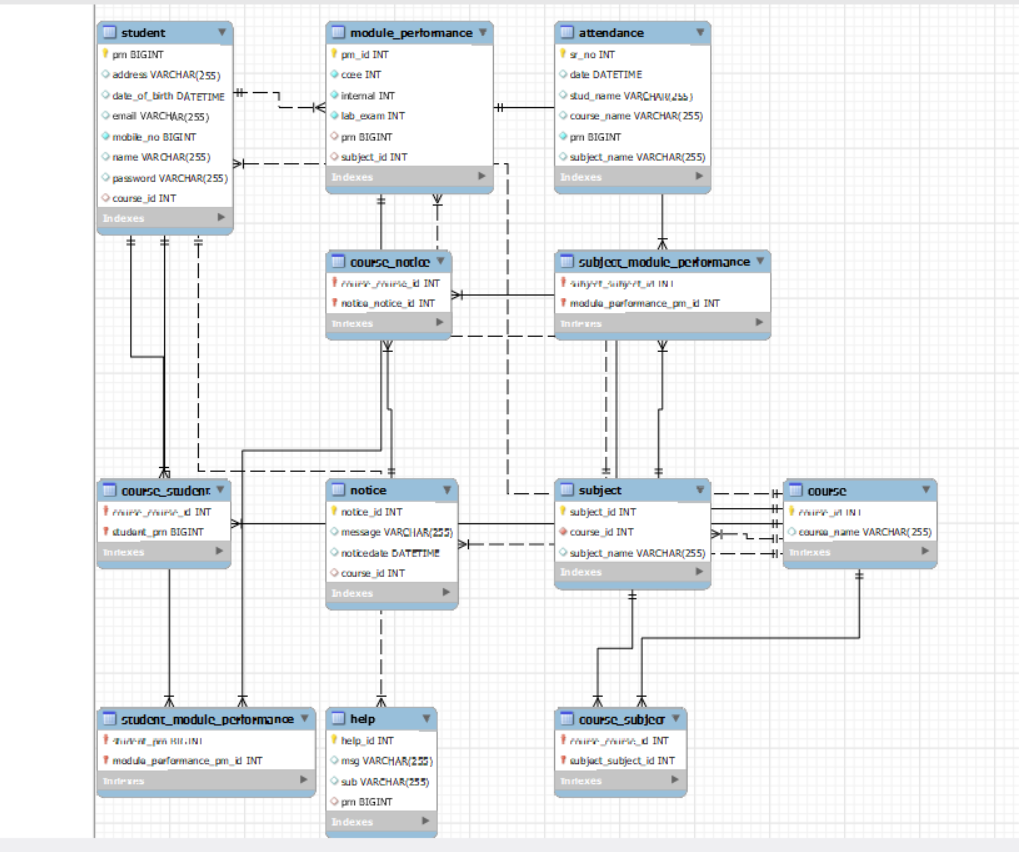
## 5.1. Database

MySQL is a relational database management system based on SQL – Structured Query Language. The application is used for a wide range of purposes, including *data warehousing, e-commerce, and logging applications*. The most common use for MySQL however, is for the purpose of a web database.

This student management system database was made based on Institute requirements. MySQL database is used to save software data for this project. It has eight tables named as Student, Attendance, Enquiry, Course, Subject, Module Performance, Notice and Admin. The system can register student information. Admin can have access to the student data for the important transaction. They can handle the data needed in managing student files as well as the transactions made by the student. This ER Diagram for Student Management System project with a table will give you the idea of how to make a database design for a student management system with ERD and tables

  **Figure 3: Entity Relationship Diagram**

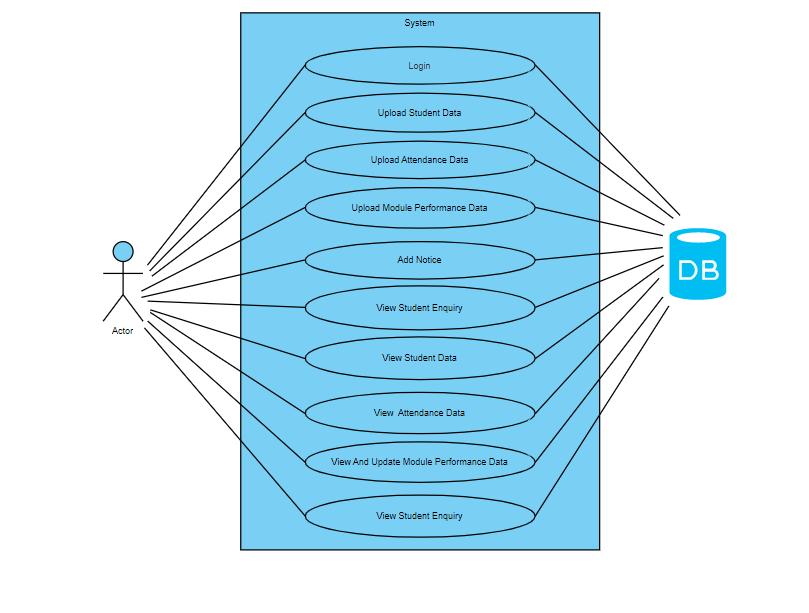
The ER-diagram has a specific name and it consists of columns and rows. For each table, we need to mark at the least one area as a primary key. The primary key is usually specified in a particular table.



**Figure 4: Class Diagram**

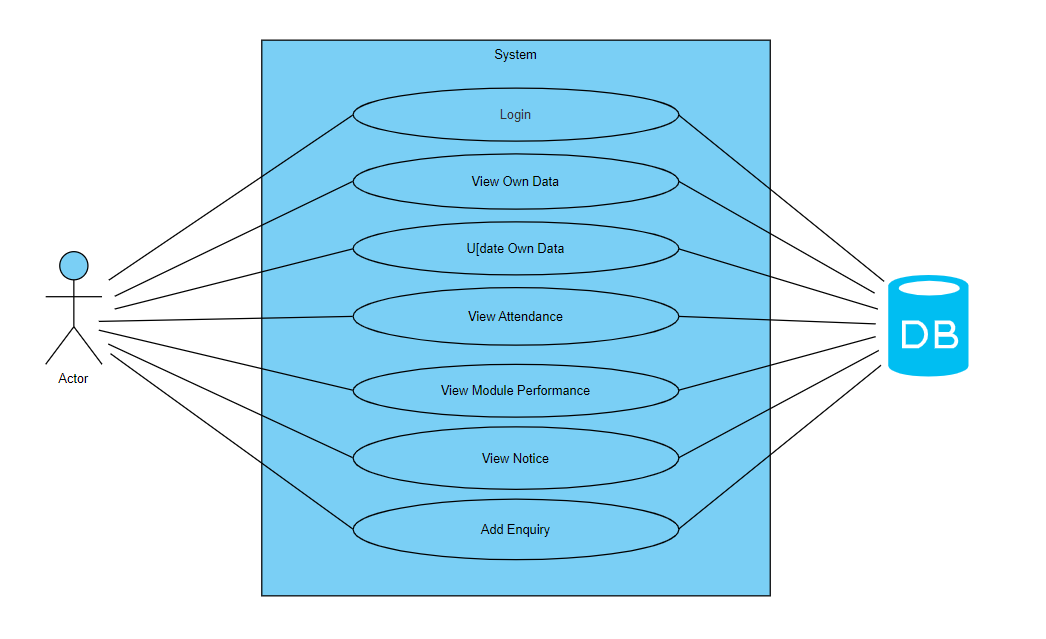
## 5.2 Use Case Diagram :

### 5.2.1 Admin



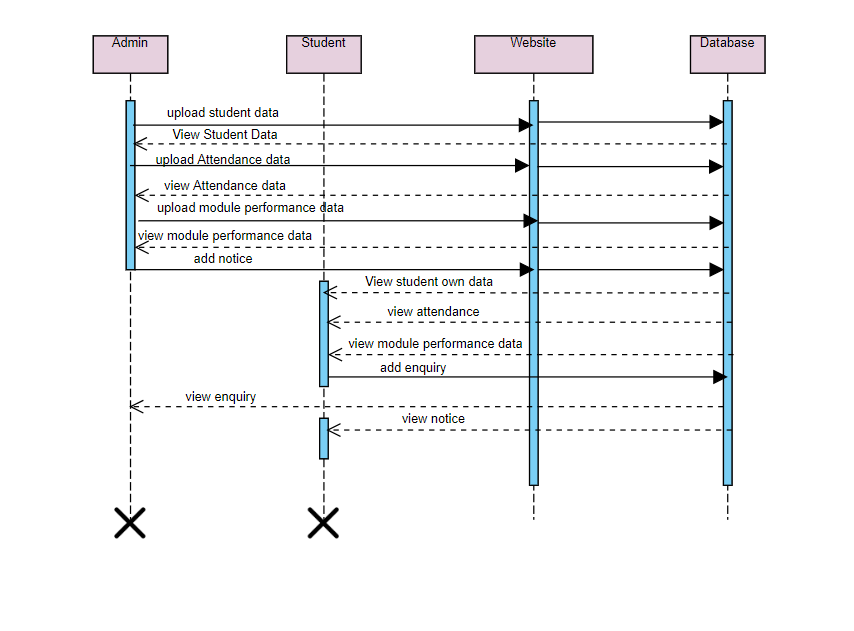
**Figure 5: Admin Case Diagram**

### 5.2.2 Student

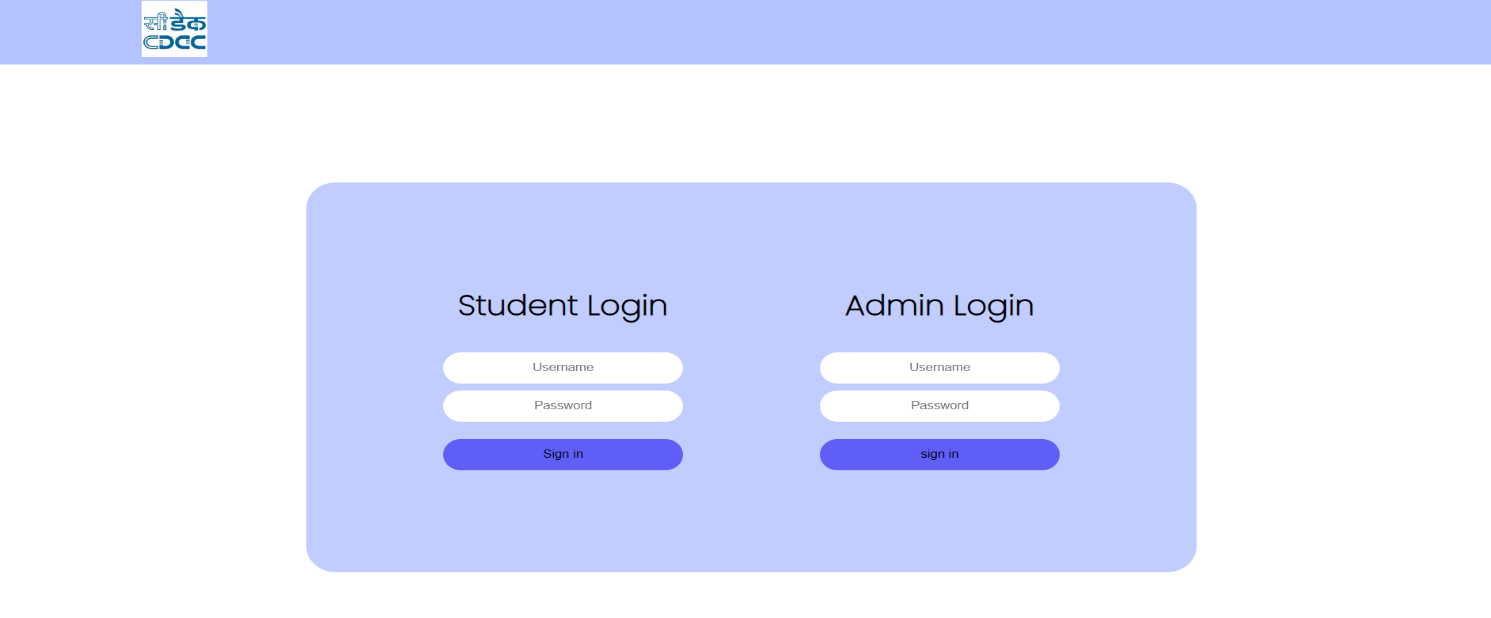


**Figure 6: Student Case Diagram**

### 5.2.3 Sequence Diagram

 **Figure 7: Sequence Diagram**

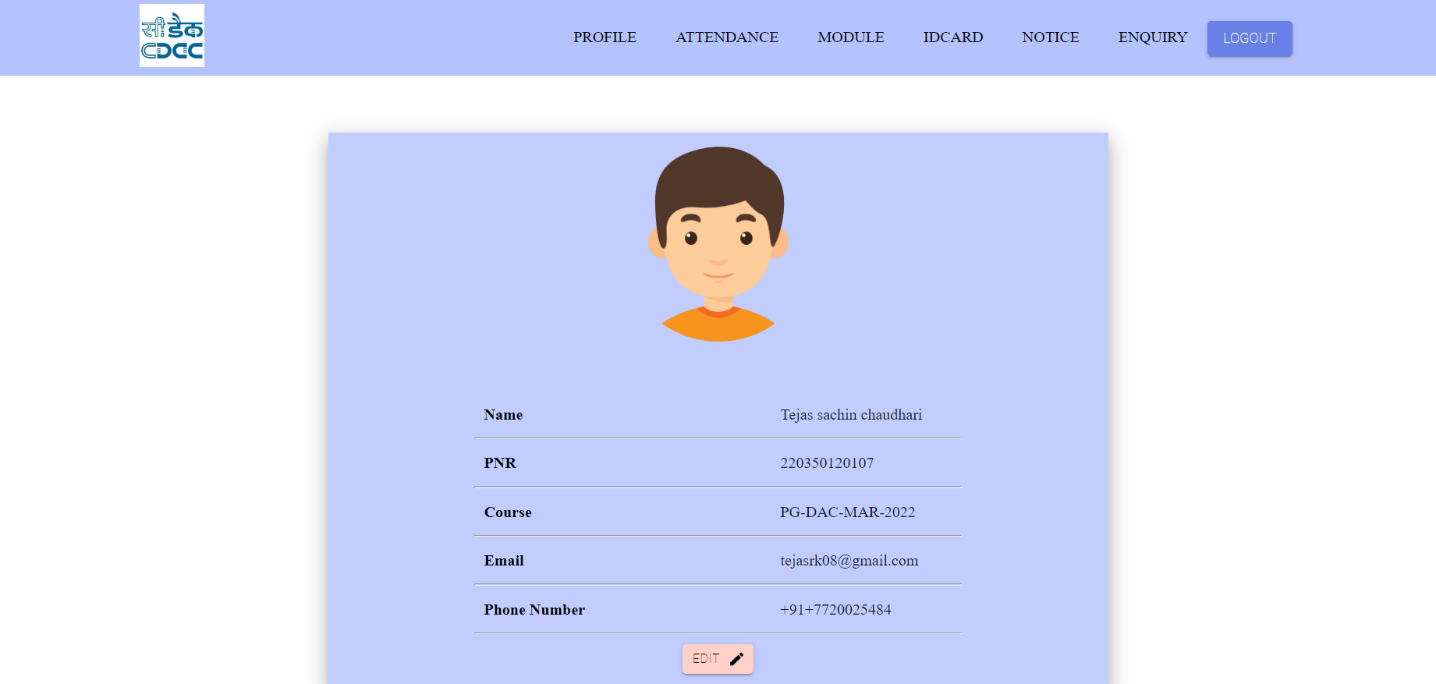
## 5.3. LOGIN PAGE



**Figure 8: Login Page**

The login page allows a user to gain access to an application by entering their username and password. In this project there is two way to access login by selecting status that is either admin or student. Admin has edge over student as he has been given privilege to edit various section whereas when one login as a student they can only view all the section except enquiry where they can post their issues. The home page or the interface is the landing page of the website when the address is typed into the browser. The webpage has separate links for login of admin and student.

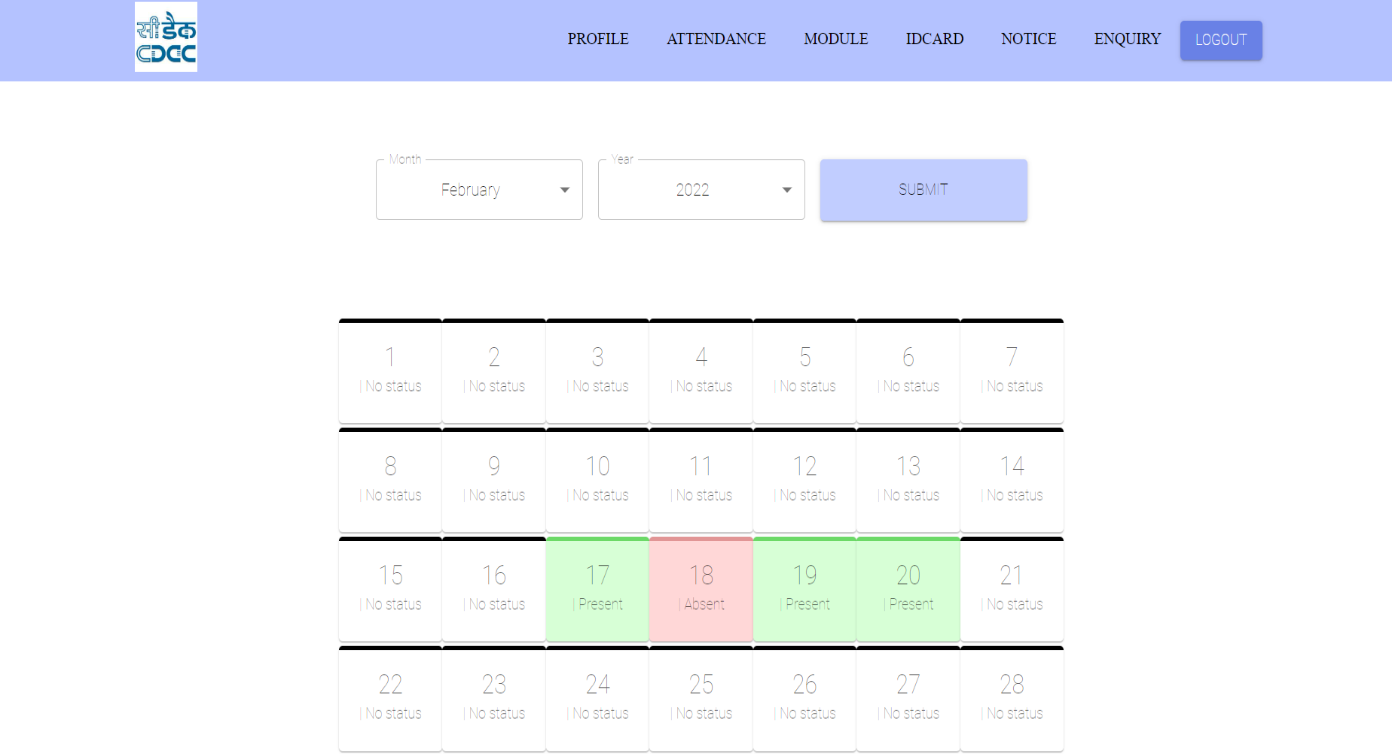
## 5.4. Student Profile



#### Figure 9: Student Profile

The student profile shows the student information your institution allows, such as information their academic load and personal information such as name their PNR no provided by the institution , each of the student's courses are listed under their personal information their contact details such as their email and phone number along with the picture provided by them. These information are stored in our database and can be accessed as per requirement.

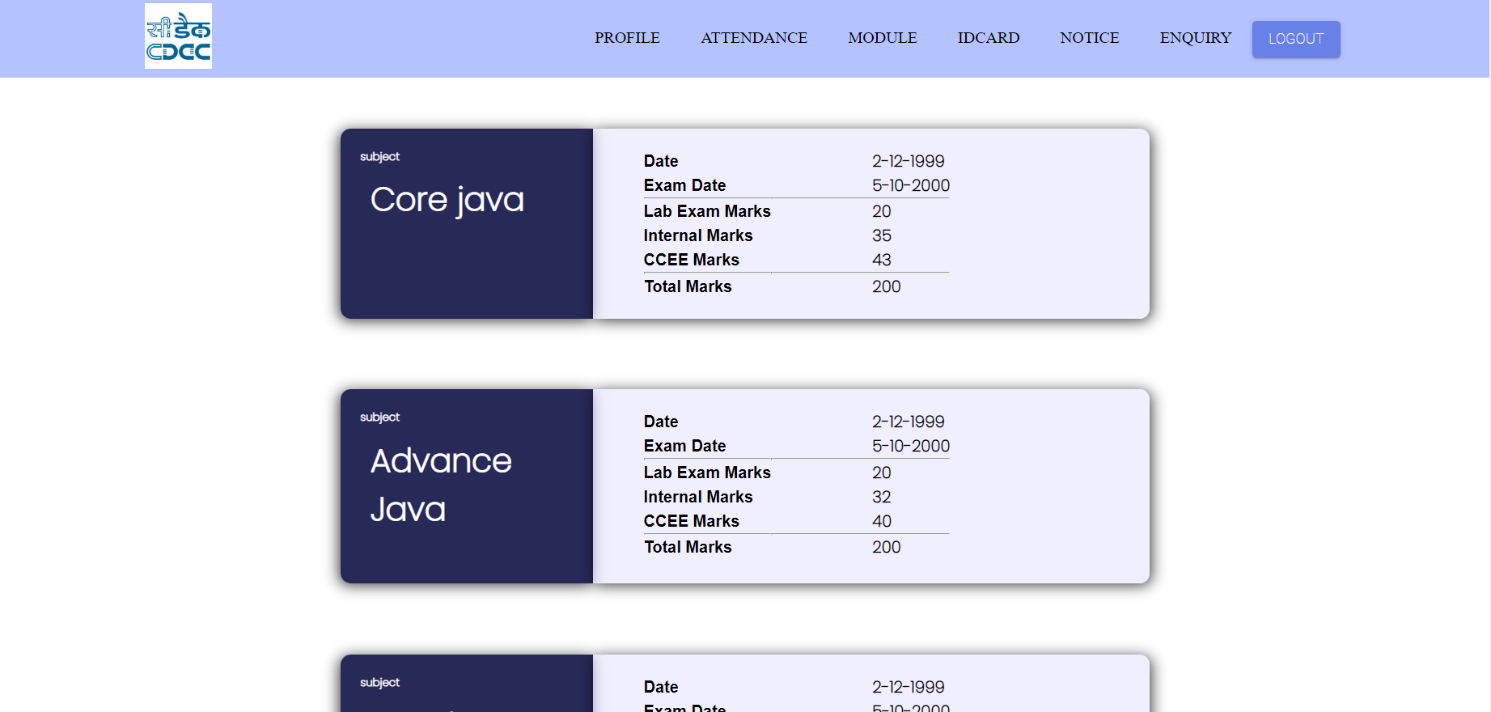
## 5.5. Attendance



**Figure 10 : Attendance**

An attendance sheet is a tool used to keep track of a student's daily attendance. Only admin is allowed to edit and upload the data of student through the data that is taken from excel sheet, once the record is uploaded by admin into the portal it is saved in the database which can be viewed by the students later according to their stream and they can select month wise data red means they are absent and green means they are present.

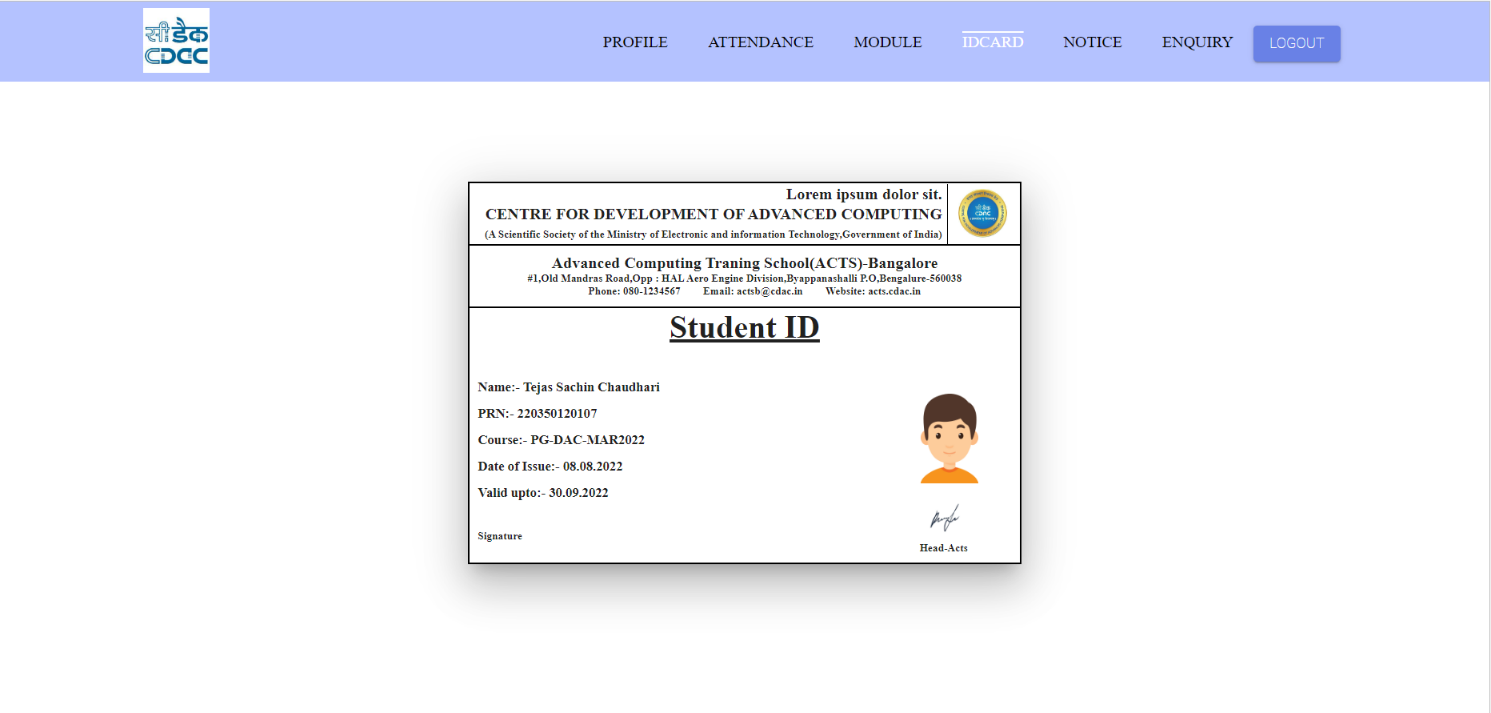
## 5.6. MODULE



**Figure 11:Module**

In this section performance of the student in different module is uploaded according to the marks received by then in different exams such as Lab exam, Internal exam and CCEE, it is done by admin and student can only view it. Student need to select their course and then module of that course will appear such as in DAC we have core java, DSA, Advance java etc.

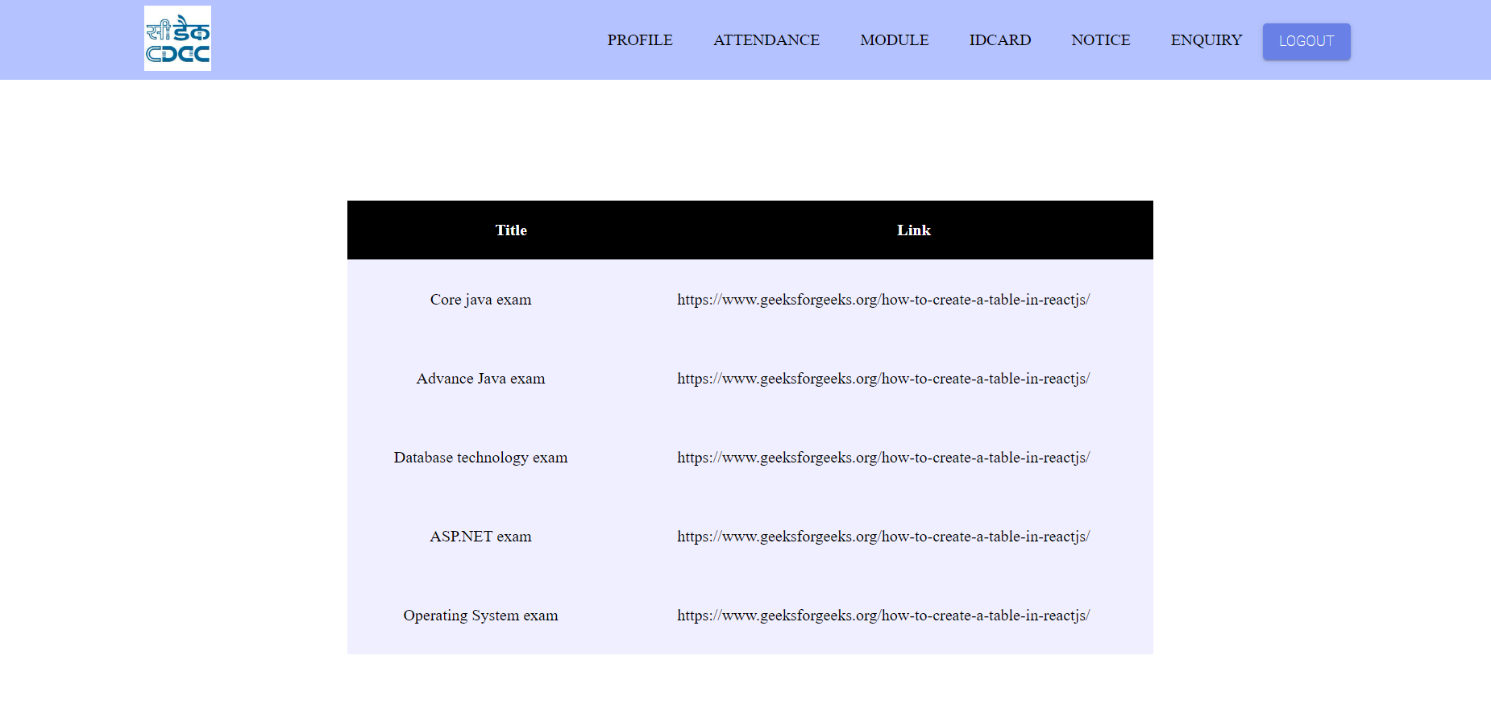
## 5.7. ID CARD



**Figure 12: Student ID**

This section consists of ID Card which shows the student id that is issued by the institution to the student displaying his/her personal information which includes name , PRN , course student has opted for, Date of birth , and till when this id will be valid upto, no changes can be made once the id is issued.

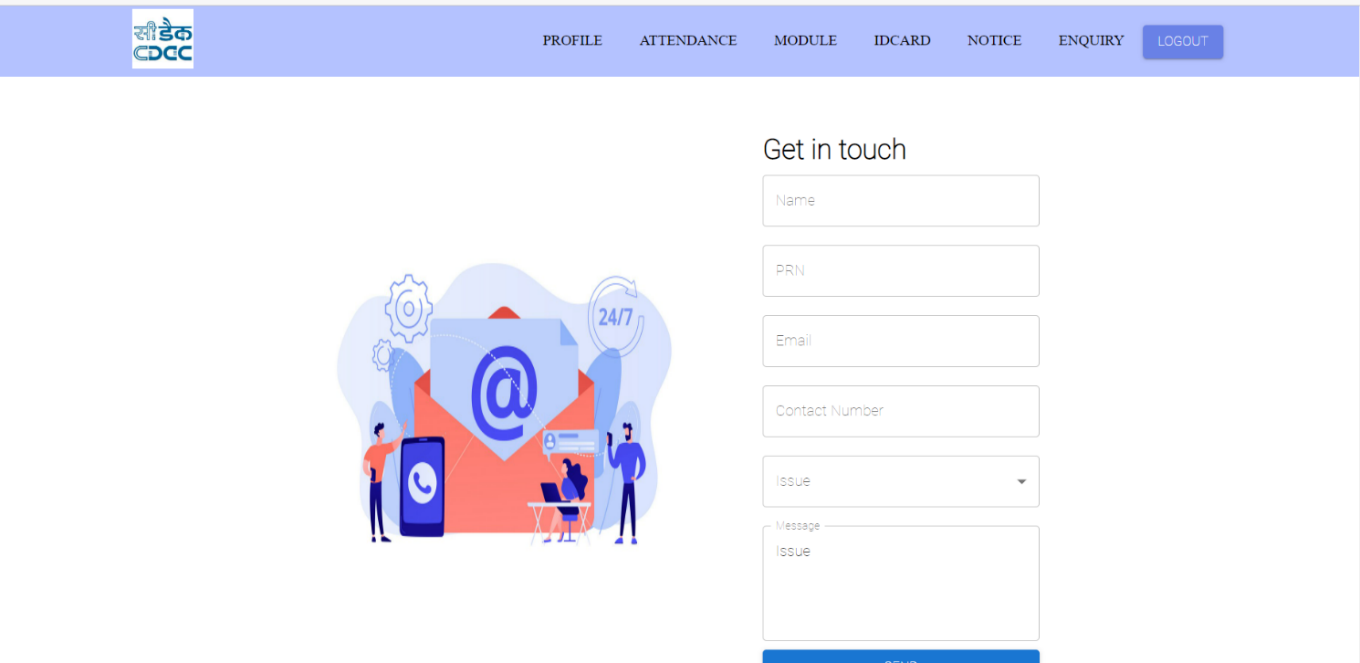
## 5.8. NOTICE



**Figure 13: Notice**

This section is called Notice in which any notice that should be told to students are posted in the link form by the admin. Any kind of file such as pdf png etc is attached with the link and posted under link column in front of the the title row in which module name is listed. Notice can be posted only by admin and students can view them on a regular basis.

## 5.9. ENQUIRY



**Figure 14:Enquiry**

This section is called Enquiry and it is made for the students to raise their question or queries regarding anything that is connected to the institution. First of all student has to write his details so that admin can know from whom the enquiry is coming and then they can choose the kind of issue which is either document, fees or other then they are given a box to post detailing of their query. This section can be edited by only student and admin can view it in their portal.

# 

# 

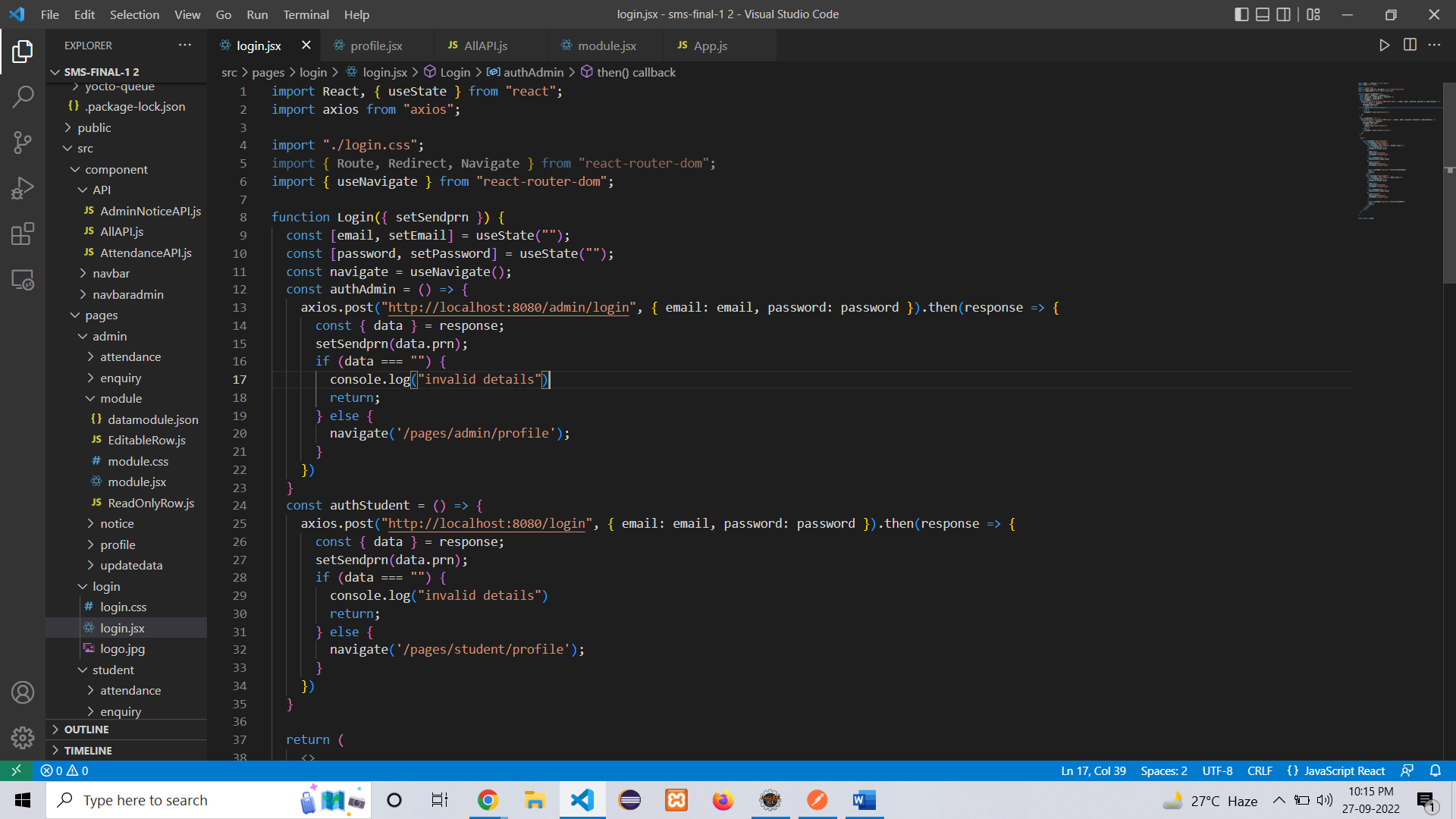
# Chapter Six. Implementation

## 6.1. Description

The created application is based entirely on free source applications and with the intention to overcome the existing problem. All of the functionalities need to be carried out in an orderly manner. The technologies used for the project is ReactJS, Java, HTML, CSS, JavaScript and MySQL..

## 6.2. Implementation of functions

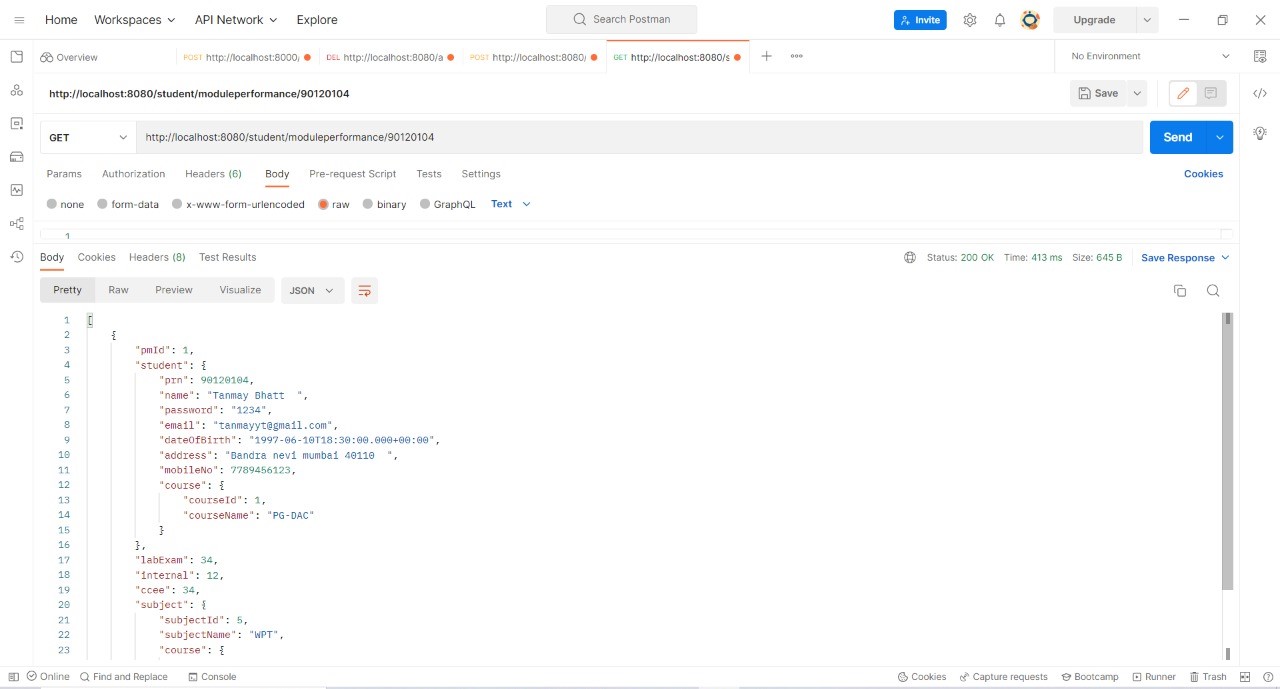
Student Interface and admin interface are the main interfaces created in this project. With the student and admin interface, we used ReactJS. The jsx was used to create the structure of page while the CSS is used in styling of pages. Javascript is dynamic language, so we can make use of asynchronous API calls. These API calls were used for transferring data from front-end to back-end and vice versa.



**Figure 15: Service.js**

## 6.3. Rest API

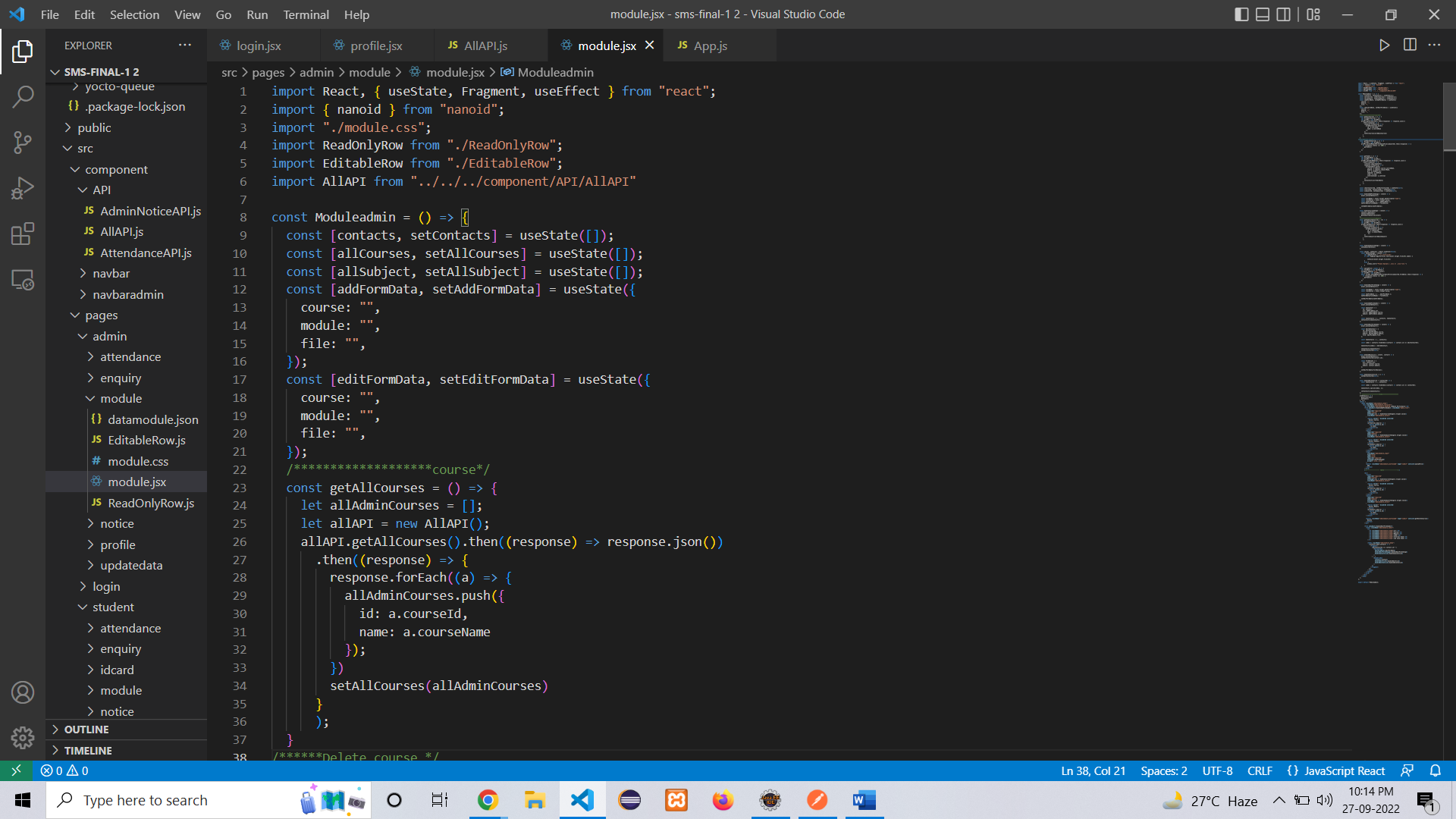
Rest Full API is used in backend. All the data that is passed between front-end and back-end is in JSON format and uses Http Protocol. The REST API is secured with Spring security. The response with appropriate status code is sent wrapped in the Response Entity to the frontend. Example of JSON response is provided below:



**Figure 16: Rest API**

## 6.4. Basic React Code

We are using ReactJS as our view layer as it is one of the most popular programming frameworks of UI. The requests are sent and received using ‘axios’ API of React. We have utilized the useState hook right at the top of our component to save what the user enters in the login page .The credentials entered in the login page are then checked from the database



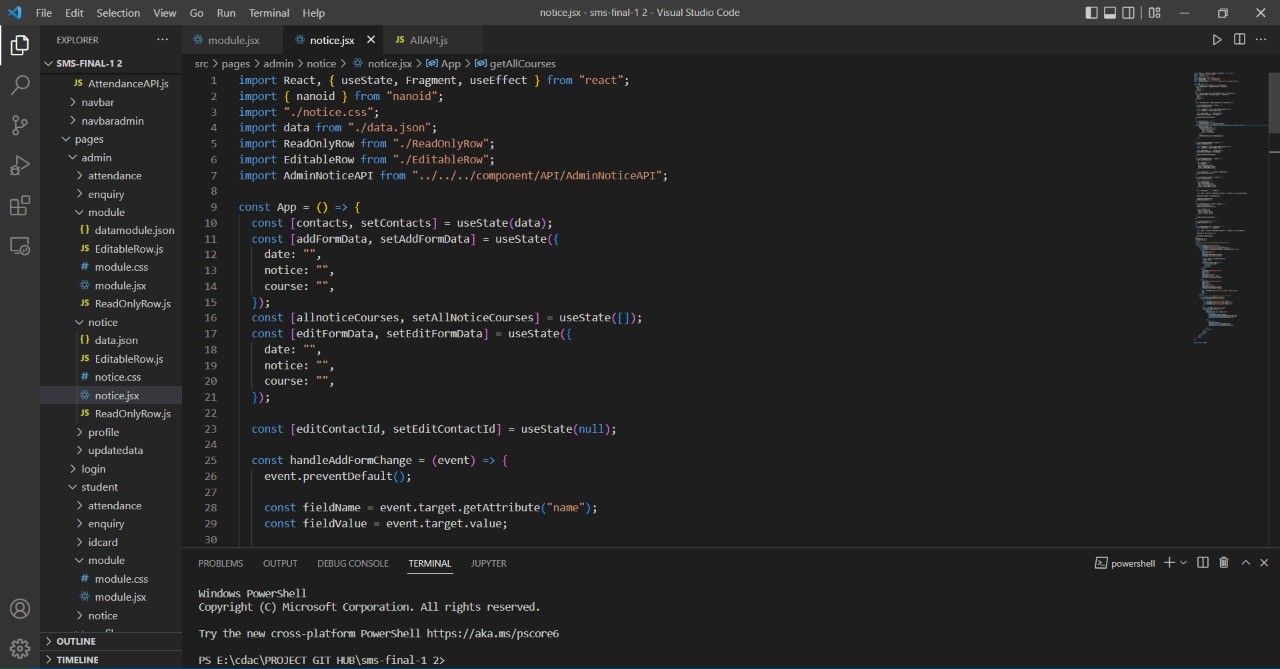
**Figure 17: Basic Functional Component**

## 6.5. Login Module

Admin and Student login are provided in MySQL. Student and admin can login using the values that has been stored using the excel file provided by the institution for logging into the application. Spring security will validate the username and password from the database and after validation the appropriate access.

## 6.7. Admin Services

Admin can access their respected services after successfully logging into the application. He/she can add profile details of students, attendance, module performance and notice for the students. Also, they can look at their generated data from the different sections.



**Figure 18: Admin Notice code**

# Chapter Seven. Testing

Software testing is an investigation conducted to provide stakeholders with Information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs (errors or other defects). Software testing can also be stated as the process of validating and verifying that a software program / application / product: Meets the business and technical requirements that guided its design development;

1. Works as expected; and

2. Can be implemented with the same characteristics.

**TEST REPORT:**

**PROJECT NAME:** STUDENT MANAGEMENT SYSTEM

## 7.1 TEST PLAN 1:

Unit id: Login for Admin

Test case id: Username

Test Type: Unit Testing

Form Name: Login

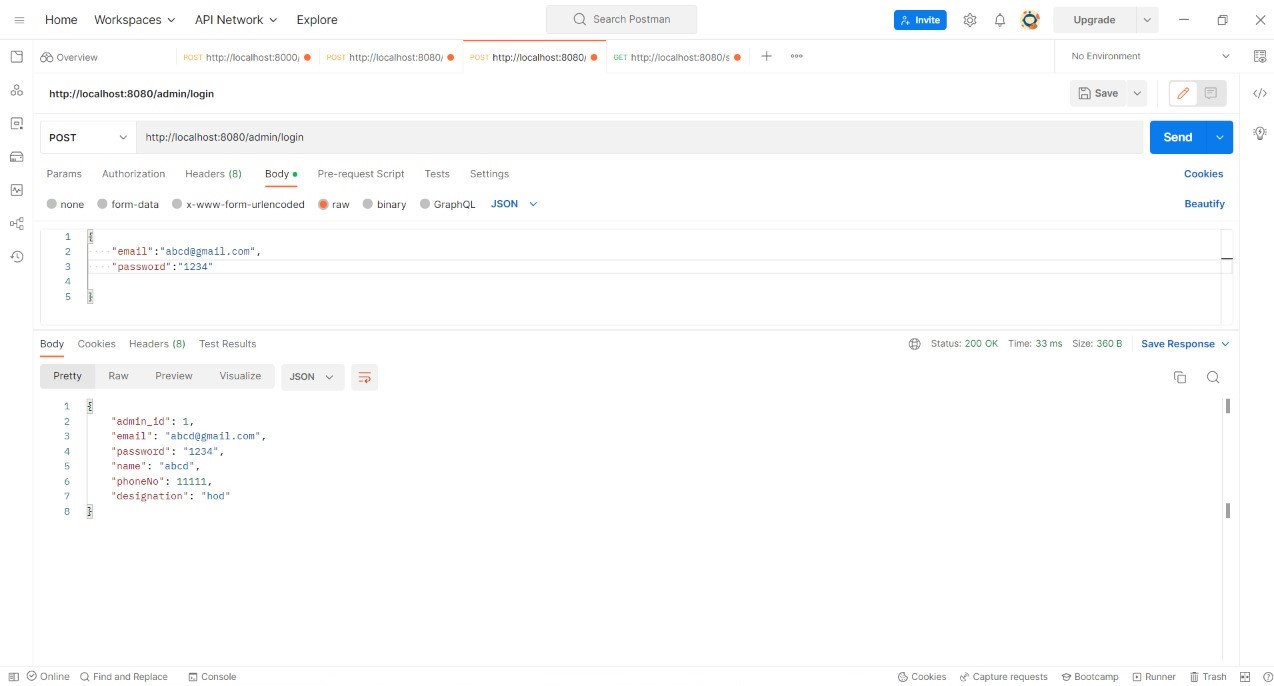
Purpose: To give access to admin after he/she enters valid admin id and password

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No | Input | Expected Output | Results |
| 1. | Column Name: |  |  |
| Username and Password: | Login Successful | Admin Profile Page |
| Invalid Input: | Unable To Login | Login Page |

After Successful Login of Admin, admin Can View his/her own profile.

First Admin have to upload All The student Data into the system. Then only he/she can proceed further to upload Attendance data, Module performance data.

Then admin can also add notices for student.



**Figure 19: Admin login Postman**

## 7.2 TEST PLAN 2:

Unit id: Login for Student

Test case id: Username

Test Type: Unit Testing

Form Name: Login

Purpose: To give access to student after he/she enters valid student id and password

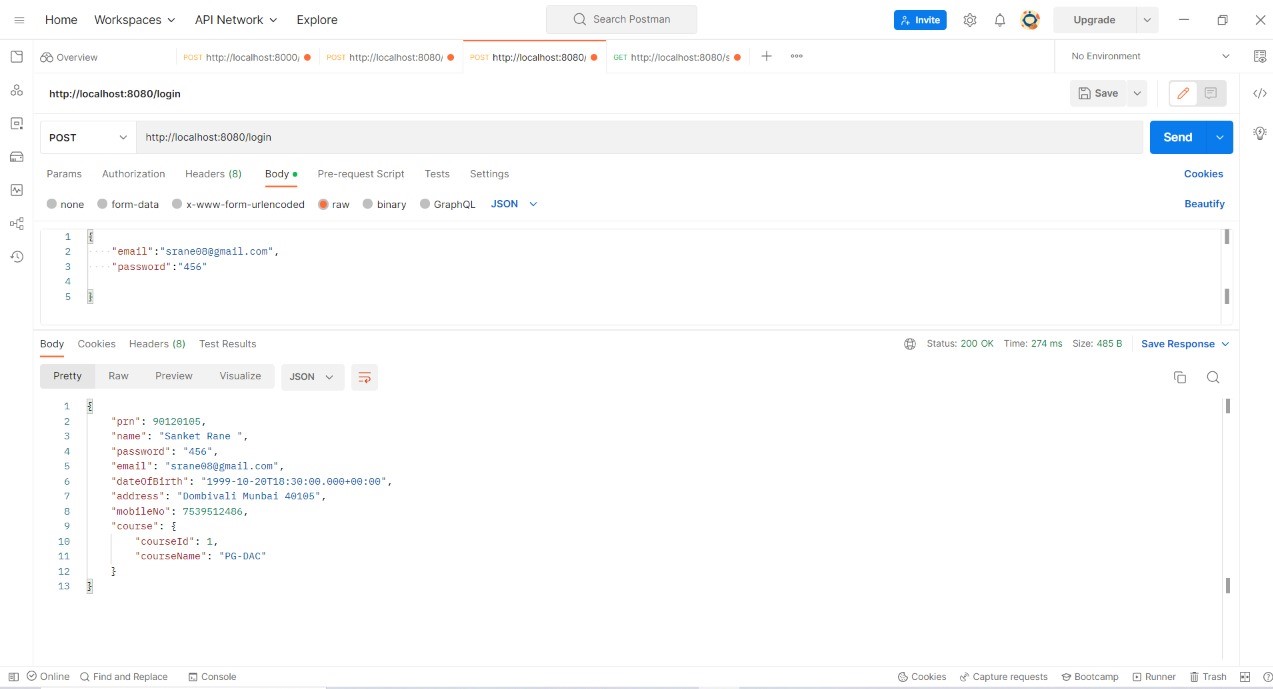
|  |  |  |  |
| --- | --- | --- | --- |
| Sr .No | Input | Expected Output | Results |
| 1. | Column Name: |  |  |
| Username and Password: | Login Successful | Student Profile Page |
| Invalid Input: | Unable To Login | Login Page |

After Successful Login of Student, student Can View his/her own profile and also can update his/her profile data.

Student can view his/her Attendance, Module Performance, Notices uploaded by the admin.

Also if student have some issues or queries, for that student can simply fill a enquiry form and submit it.

Note: Student Login Should be done after admin uploaded the student data into system.



**Figure 20: Student login Postman**

# Chapter Eight. Conclusion

 Student Management System, or SMS, is a web-based platform that helps schools and colleges take student data online for easier management and better clarity. The project also helped to provide knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future.

This project helps in making paperless activities. It reduces the workload from institute employees. It provides more flexibility and ease to students and admin. It developed using ReactJS, MySQL, bootstrap, Spring boot technology. This work has created a little awareness and promotes the idea that the concept of paperless institute is reality.

# Chapter Nine. References

* <https://reactjs.org/docs/getting-started.html>
* <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>
* <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
* <https://docs.oracle.com/en/java/>
* <https://dev.mysql.com/doc/>
* <https://www.eclipse.org/documentation/>
* https://code.visualstudio.com/docs/