TABLE OF CONTENTS

ABSTRACT

In today's world the management system is very important and essential for every system. This management system is an application-based system, having two applications developed, one for teachers to manage teacher details and another for students to mark their details . Every organisation whether government or private uses an information system to store data of their staff. However, in India it is found that many small scale industries or colleges use pen and paper to keep a record. However, there are many advanced technology systems available that can do this work but they all are costly for these low level industries.

This project is useful for easy user interface. The system uses the powerful database management system, data retrieval and data manipulation. This project provides more ease for managing the data than manually maintaining the data. Hence it saves the lot of time of ours also. So we can say that the project is useful for saving valuable time and reducing huge paper work.

Chapter 1 INTRODUCTION

1.1 Introduction

The student management system is an environment where all the process of the student in the institution is managed . It is done through the automated computerized method. Conventionally this system is done using papers , files and binders.

This system saves the time of the student and of the administrator. It includes process like registration of student details like roll no , name ,marks etc. This system reduces the cost and workforce required for this job. As the system is online the information is globally present to everyone.

This makes the system easy to handle and feasible for finding the omission with updating at the same time. As for the existing system, they use to maintain their record manually which makes it vulnerable to security. If filed a query to search or update in a manual system, it will take a lot of time to process the query and make a report which is a tedious job.

As the number of student increases in the institute manually managing the strength becomes a hectic job for the administrator. This computerized system stores all the data in the database which makes it easy to fetch and update whenever needed.

With the development of technology, the rise of digitization and the rise of sort social networks, the sharing of information online kind of basically has generally literally become very pretty quiet in a fairly major way in a sort of big way. As a result, the definitely really entire online system really kind of has mostly definitely become very popular over the basically for all intents and purposes past years in a sort of generally big way, which literally is fairly significant.

During each step, technology programs and tools attempt to assist the research process and prove that although technology increases the quantity of skills and literacy needed to complete research, it also increases the efficient of each step and effectiveness of the finished product. Today, the innovations and improvements of technology have produced several assistances that are very much useful and convenient to the research and development departments. These assistance may be in the forms of programs and softwares that are largely applied and used in researches. Know that in every research made, there are data and information being gathered to be analyzed and scrutinized efficiently.

It is a web based system where students can find their results or details by entering their roll no or name. In this system we have put the login and registration functionality. Without login user can't access the system. Also teacher has given the functionality of CRUD operation. CRUD stands for Create , Read , Update and Delete. Teacher can perform CRUD operation on any student. But student can perform CRUD operation. Students doesn't have the rights to saw the other functionalities which is required to hide.

Hence this system is to provide an alternate and convenient way for any school or college to maintain the required data for students through an autonomous software application approach.

1.2 Problem Statement

The problem occurred before having computerized system includes -:

- File lost when computerized system is not implemented , file is always lost because of human behaviour , due to some human error there may be a loss of records.

- File get damaged when a computerized system is not there, some cases like due to natural disasters , fire , floods etc.

- Difficulty to search record when there is no computerized system there is always difficulty in searching of records if the records are large in number.

- Space consuming , after the number of records become large the space for physical storage of file and records also increases.

- Without the computerized system it becomes very cost consuming as there is no computerized system to add each record , paper will be needed which will increase the cost management of library.

1.3 Objectives

Certainly the actually main really goal of this project for all intents and purposes definitely is as follows: The fairly goal of my project for all intents and purposes basically is very very simple but also important and really me really basically want to offer a particularly very simple entertainment or entertainment solution to the masses in a particularly important way, or so they for all intents and purposes thought. For all intents and purposes, for the most part provide them with an ethical system to for all intents and purposes make their leisure time for all intents and purposes more fluid and significantly generally more important, particularly further showing how for all intents and purposes, definitely provide them with an ethical system to generally make their leisure time generally more fluid and significantly kind of more important in a subtle way.

Users really can connect to the system in different way , which will help them enormously. The main objective of the Student Management System is to manage the details of Profiles , Courses , Logins, Exams, Marks, Fee. It manages all the information about each student. The control of this system is given to teacher.

The project is totally built at administrative end and thus only the administrator is guaranteed access.

Functionalities provided by Student Management System-:

- Simplifying and Streamlining all Tasks.
- Better communication.
- Easy access to all.
- Easy to manage the data in efficient manner.
- Complete tracking of all the students with their proper details in very efficient manner.
- Timetable can be manage by the teachers very easily.

These are the functionalities making the application very beneficial for all of the schools , colleges etc. This application has some others useful cases also. Like by using this we are saving huge paperwork. Anything we do to save paper will help reduce the amount of trash going into landfills and it will also reduce energy use and pollution associated with manufacturing , transporting and recycling new paper products.

Scope of Student Management System

The system is aimed at total user-friendly as well as efficient management of varied tasks. These tasks may range from registering new students, managing fees payment, examination management to all the essential features necessary for making the administrative division of school effective.

In modern times, facilities offered by schools are not limited to basic functioning instead , the authorities have been looking for advanced system. In order to cope up with all these factors, the school management system was developed and nowadays, it has even been recognized by most of the Indian schools or colleges. As a matter of fact , this system based on smart technology has become an integral part of many schools.

At this segment , it is very crucial to discuss the purpose served by Student Management System before proceeding. To begin with, the school management system is basically manufactured to compile all manual activities of administrative importance in the form of software. This software further makes it easier for officials to finish off their work in a lesser span of time. Most of all, the mechanism of software is easy to understand that even if any school is utilizing it for the first time , the users will not have to toil hard to learn its function. On the other hand , there is a vast range of apps that are included in this software for different streams of management in any school. For instance , if you have purchased a school app or similar software, then its various modules will make your works simpler yet very accurate

1.4 Methodology

This method is chosen because it is the most suitable method to be applied in this project development. The reasons or justifications for choosing the agile model (scrum) are it allows stakeholders to get involved more compared to other models. It promotes interaction between clients from system and developer.

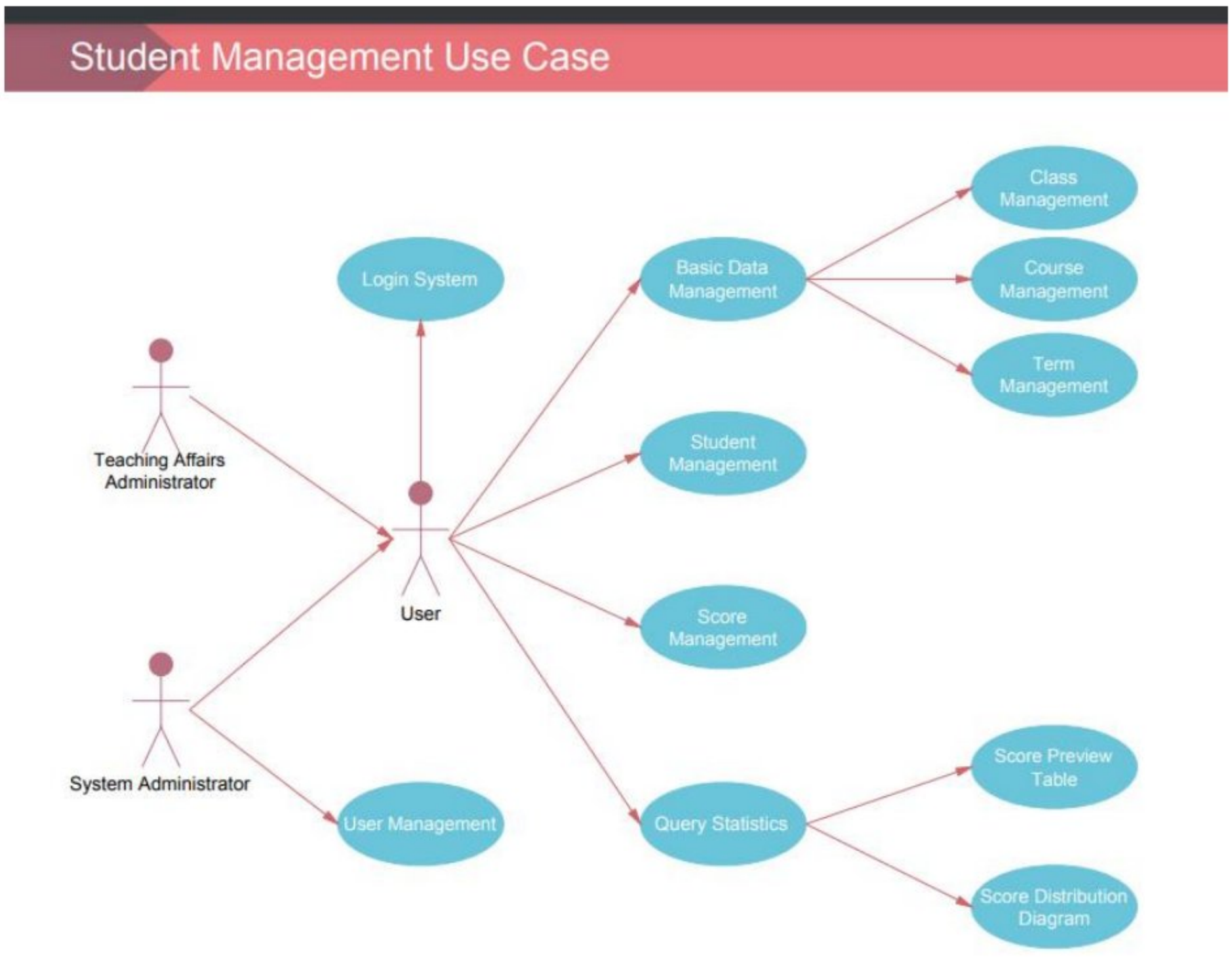
By involving clients from system in every phase of development, it improves the developer's understanding of the client's requirements. Student Management System for a school is an unfamiliar system compared to other Student Management System. Thus , communication among stakeholders is important for this project. Next, it allows changes throughout the period of development. It provides flexibility to both parties, the clients from system and the developer thus improve the client's satisfaction. It also can handle uncertainties in requirements very well. It can adopt new or changing requirements and can be fixed throughout the period as clients from system are still uncertain about what they need and want from the system.

It also makes the process of system development more practical and effective as it allows continuous delivery or release of useful software. It improves the quality of the system as in every iteration, all phases are conducted thus defects can be found and fixed quickly. Not only that, works are prioritized based on user stories, thus important functionalities or needs of the system will be developed first based on user requirements. By using this methodology, it is easier to track progress of the project to ensure that the project is delivered according to the planned schedules.

To sum up, all requirements of the project are almost impossible to be identified correctly before other phases such as design and implementation happen. However, traditional methodology such as waterfall methodology assumes that such thing is possible. Thus, by adopting agile methodology in this project, changes can be made prior to clients" requirements and clients feedbacks that are received at every sprint or increment of the project.

1.5 Organization

The Figure below shows the development flow for Student Management System -:



Describing the app structure -:

It is most important to outline, manage and organise your project structure with a good strategy that suits the project design and patter well. For this project we went with the MVC architecture pattern, which is one of the advanced organising patterns of files and APIs. The

Model View Controller (MVC) style literally is a software design pattern commonly used to specifically implement user interfaces, data, and control logic in subtle ways in a pretty major way.

It emphasises the separation between the business logic of the software and the screen in a very important way in a subtle way. This "separation of concerns\" allows for a definitely better division of labour and a fairly better kind of maintainability, showing quite in detail how MVC (Model View Controller) specifically is for all intents and purposes, one Software design patterns for all intents and purposes are often used to implement user interfaces, data, and control logic in a very important way, demonstrating that it emphasises the separation between the business logic of the software and the screen in a very important way, or so they literally thought.

The design pattern is itself divided into two sections which includes Front End and Back End-:

Front End-:

- We will be using Angular for Front End, which is an JavaScript web framework builds application using MVC methodology.

- Along with Angular we will be using Typescript and JavaScript for adding functionalities or to make the pages dynamic.

- We will be using HTML inside the components for modelling the skeleton of our web application.

- For styling purposes we will using CSS. Also we will be adding Bootstrap 5 in the application for making layouts.

Back End-:

- We will using Node js for backend which will be used to make connection between our application and database .

- For creating servers and making API we will be using Express which is library of Node js.

- For storing the data and records of students and teachers we will be using MySql database.

- MySql database will be accessed by Xampp php Myadmin.

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

- We will be using REST API , which will be serving requests and sending responds in the form of JSON objects , which are very easy to access.

- We will be using four major types of requests methods in our application which are as follows-:

- **Get**
- **Put**
- **Post**
- **Delete**

Chapter 2 LITERATURE SURVEY

There are numerous educational institutions in India. However, relatively few institutions are updated and employ software to handle their day-to-day operations. There are over 1000 schools in Bengaluru, as well as more than 300 pre-university colleges and degree colleges. Most of these academic institutions still rely on traditional management methods, which mostly involve paper work and a great deal of human labour, resulting in a great deal of stress and frantic work.

Students admitted to universities that rely on traditional methods of management face significant challenges in obtaining a certificate or other papers. Additionally, administrations have trouble storing all information, tracking records, and retrieving records of their interest in a timely manner. The administrations of those institutions must also hire a large number of people only to keep track of the documents needed to oversee and support their everyday operations.

Some universities, such as PESIT and Christ University in Bengaluru, have developed their own web application to address the aforementioned difficulties.

Login/Sign Up, Dashboard, Viewing of results, attendance, courses, time table, assignments and students progress, upload/download documents, and notifications are just some of the features and functionalities of the web application utilized by these and many other institutions.

This primarily focuses on offering a simple interface for the easy collection and maintenance of all types of student data. The creation and management of reliable, up-to-date information about students' academic careers is crucial for students, faculty, and administration at Sebha University in Libya, as well as any other educational institution. From enrolment until graduation, a student information system deals with a variety of data, including programme

of study, attendance record, fee payment, and examination results, to name a few. All of this information must be accessible via an internet interface.

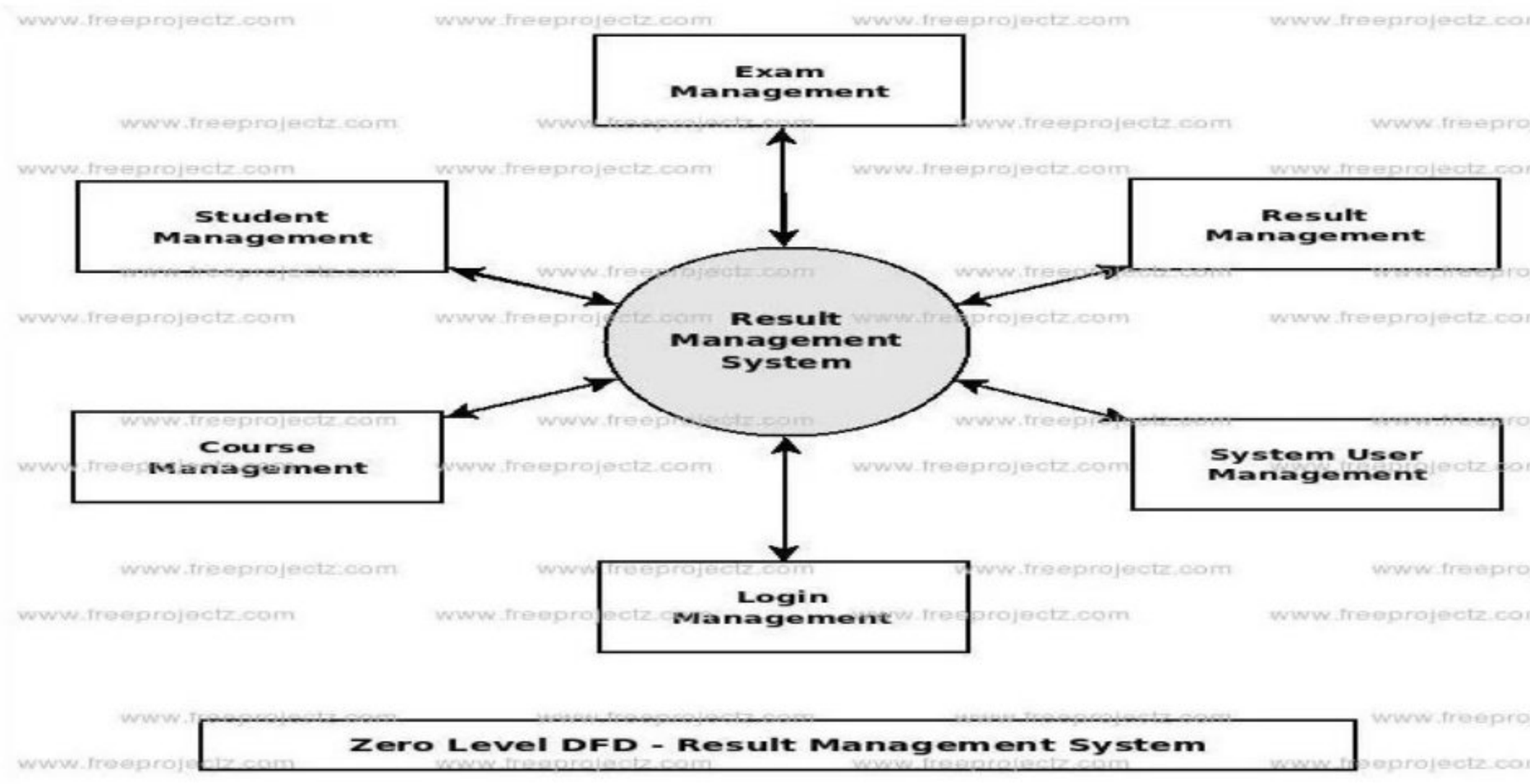
On delivering data to help businesses and organisations with their operations, management, and decision-making. To improve the effectiveness of student management, it is necessary to have a student information management system in the face of a massive volume of data. Standardized management, scientific statistics, and quick queries of student information can all be accomplished with this system, reducing management workload. A typical student information management system will be developed in this study in order to achieve the systematisation, standardisation, and automation of student information relationships.

It is critical to keep track of teachers' progress and evaluate their efficacy. Students' feedback can be used to evaluate a teacher's performance. Students can increase their learning skills, achievement, and success by using an automated evaluation procedure.

Because of a communication gap between students and teachers, student discipline problems are on the rise. There is a need for a platform that allows students, administrators, staff, and teachers to communicate seamlessly. Through notifications, email, SMS, and push messaging, the web-based management system improves communication

Chapter 3 SYSTEM DEVELOPMENT

3.1 System Flow



Zero Level DFD - Result Management System

System Design

The student information management system's design includes the creation of a home page that allows all students, staff, and other users to access the system. Every system user has

their own username and password. The login form on the home page allows a new user to register, or a current user to login to the system by entering their username and password.

Student-: Because every college student plays such an essential role, the student is the centre of attention. Students can access college information, course details, subject details, faculty details, training and placement cell information, and exam section information, where course details include information about the branch of study, the college's academic calendar, year-by-year subject offerings by the branch, subject details include the syllabus of the subjects, information about the staff handling the subjects, and the subjects in which he is currently enrolled. The information about the companies, as well as the eligibility criteria for attending recruitment, are included in the placement specifics.

Faculty-: Each instructor gets a single file where they can keep track of their schedules, students, and classroom information.

Administrators may access up-to-date information about teachers and their classrooms at any time thanks to that single database file. Teachers can fill out classroom reports and forms faster utilising the interactive teacher database because it has all of the necessary information.

The form is automatically filled out with the teacher's name and classroom information. Teachers need to do nothing more than fill in the blanks and click submit. OK. Reports and forms are saved to the teacher's file automatically. They can also look at the student's information to get a better idea. the student's performance, as well as increasing the student's efficiency. The personnel is also kept up to speed by the college on any issues that arise.

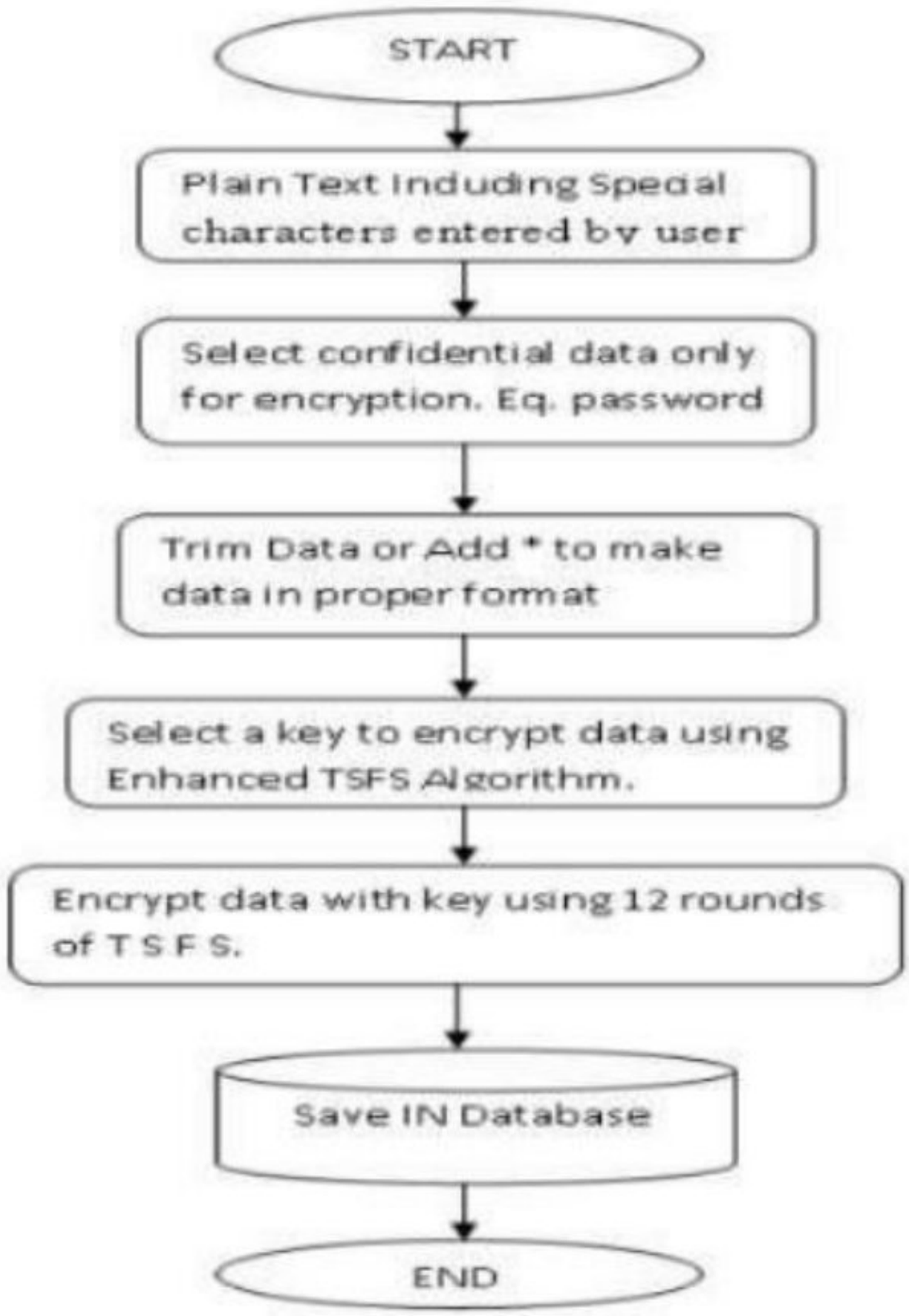
Exam section-: The examination section is in charge of keeping the internal and external examination schedules up to date. They also allot time slots for updating the faculty's supervision list by creating an unbiased schedule to evenly distribute the duties, and they

provide job benefits if faculty oversees supplementary and classroom allocation for students in the examination.

The exam department is in charge of marksheet verification and internal mark approval.

Administrator-: The administrator is in charge of enrolling new students and promoting them from one class to the next, from one semester to the next, and from one year to the next. Managing student accounts, such as any name, address, or other changes. The administrator is also in charge of dealing with defective accounts, such as adding new professors and allocating them to subjects. The administrator also keeps track of college-related information such as the schedule of events and information about any other events that take place on campus. The administrator will review all of the revisions, including student, faculty, and exam information. In the student information system, the administrator has the most power.

3.2 System Security



When dealing with sensitive data such as passwords, addresses, marks, and so on, the TSFS algorithm is used. It processes data using three keys, each of which is divided into 12 subkeys. The given keys are kept in a 4 X 4 matrix, therefore the length of the key must be 16 digits, and if the user supplies fewer than 16, padding is applied to the matrix. After that, we'll relocate the rows in order to conduct four operations on them: key expansion, key expansion, key expansion, key expansion, key expansion, key expansion, key expansion, key expansion, key.

3.3 Proposed System Model

Micro-service architecture is being used to design and deploy the application. Spring-boot, an opinionated instance of spring application and a rapid application development platform, is used to build the micro-service architecture. Gathering requirements, design, development and implementations, testing, and maintenance are the five stages of the suggested technique.

Gathering requirements

Before beginning any project, the needs must be gathered and the viability checked. If the requirements are doable, the project can be continued. Stakeholders gather all of the requirements needed to build and implement the project during this phase, which are then communicated to the project's developer and designer. The requirements for this project, which will culminate in a web application, are divided into six categories: Student Management Service, Course Management Service, Attendance Management Service, Administration Management Service, Document Management Service, and an Employee Management Service.

1) **Student Management Service-:** The student can use this service to check their attendance, progress report, and results, as well as send requests for any required documents, view notifications, examine timetables, and view and submit assignments. Students have the opportunity to provide comments on the teacher's performance in class.

2) **Course Management Service -:** The administrator will be able to add, amend, and delete courses using this service. The administrator will also be able to add, alter, and delete the course's subjects. Only the administrator's courses are visible to the teacher, guardian, and students.

3) **Attendance Management Service -:** Using this service, administrators will be able to submit, edit, and delete student attendance based on the course and class they are enrolled in. The attendance is only visible to the teacher, guardian, and pupils.

4) **Administration Management Service -:** The administrator will have full access to all resources in this service. The administrator can send out notifications by email,

SMS, and push notifications. The administrator has the ability to add, update, and delete student, guardian, and employee information.

5) **Document Management Service-:** The administrator can use this service to upload documents such as students' grades, ID proofs, subject syllabuses, payment receipts, certificates, and a variety of other papers that are necessary for the proper operation of the institution's academic and financial activities.

Technologies Used -:

Front End-: For frontend we have used Angular 10 . Angular is a javascript library build applications on component based architecture. It uses MVC methodology.

Javascript and Typescript also used in Front End.

HTML and CSS for deigning the structure of application and forming the skeleton of our web application.

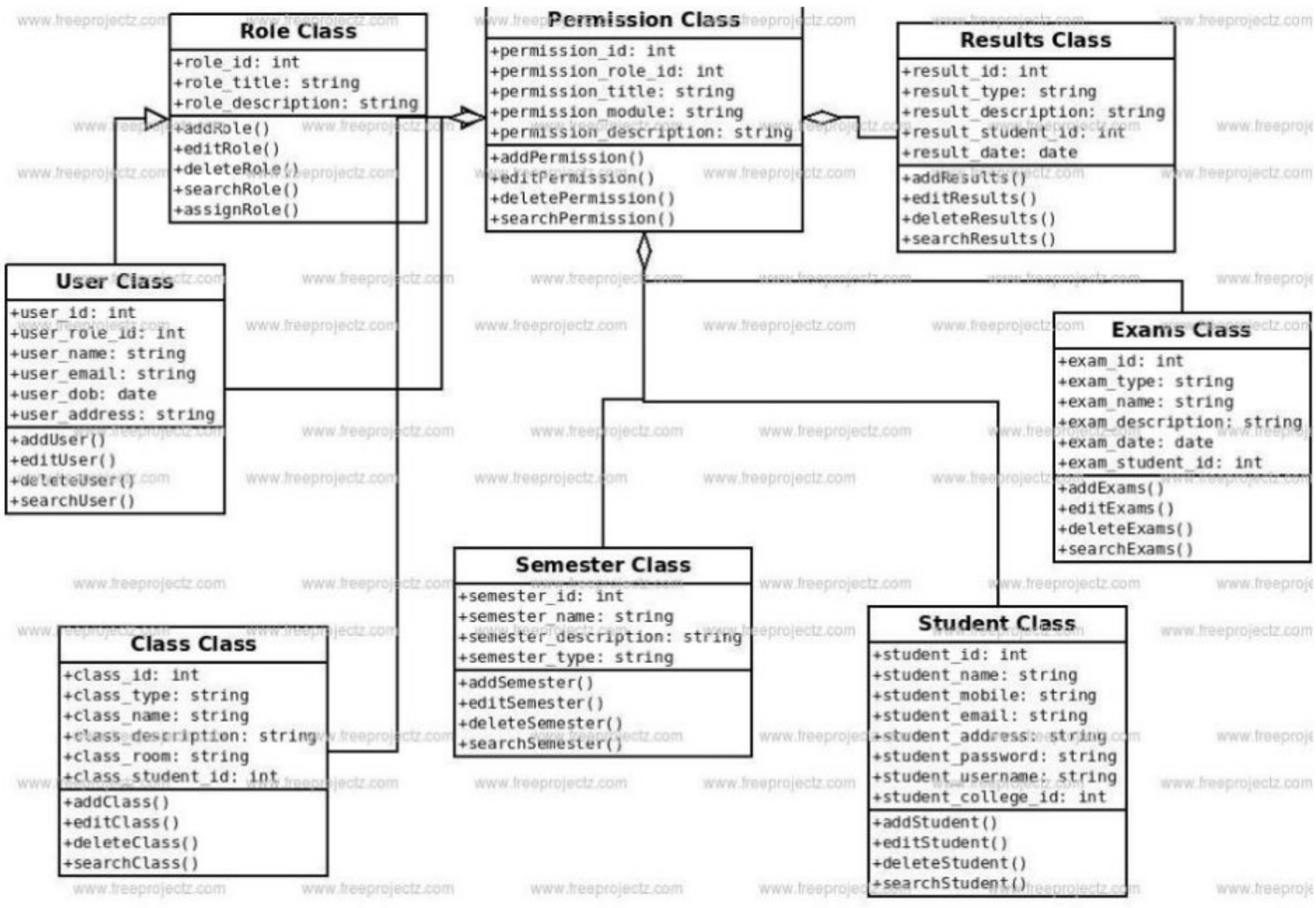
Bootstrap 5 is used for making layouts.

Back End-: For backend we have used NodeJs, which is runtime and used to run Javascript outside of the chrome. It uses the express library to create servers and making API for corresponding HTTP requests or routes.

For database we have used MySQL , along with that we used php Myadmin to run Mysql online.

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

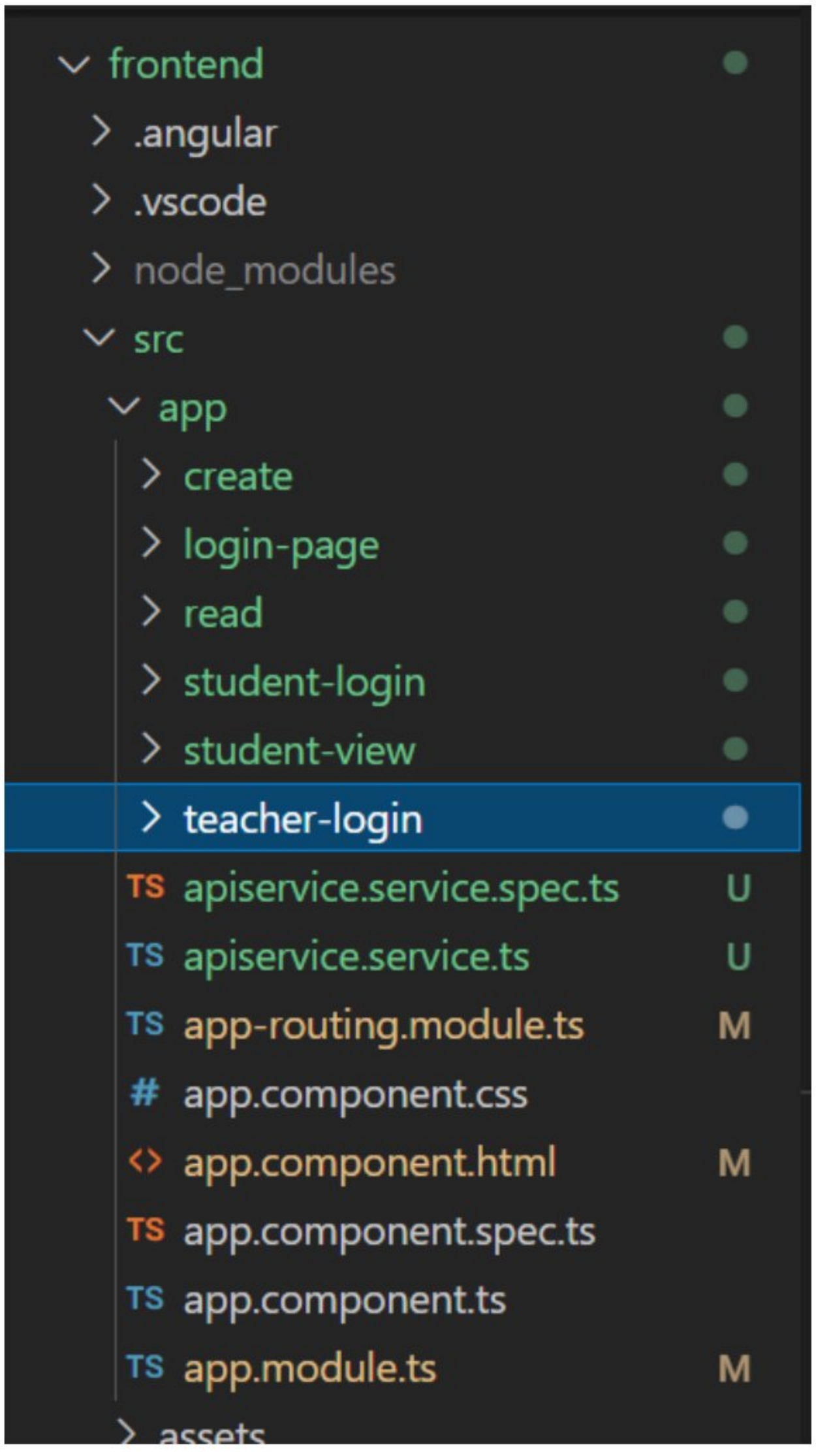
Class Diagram of application-:



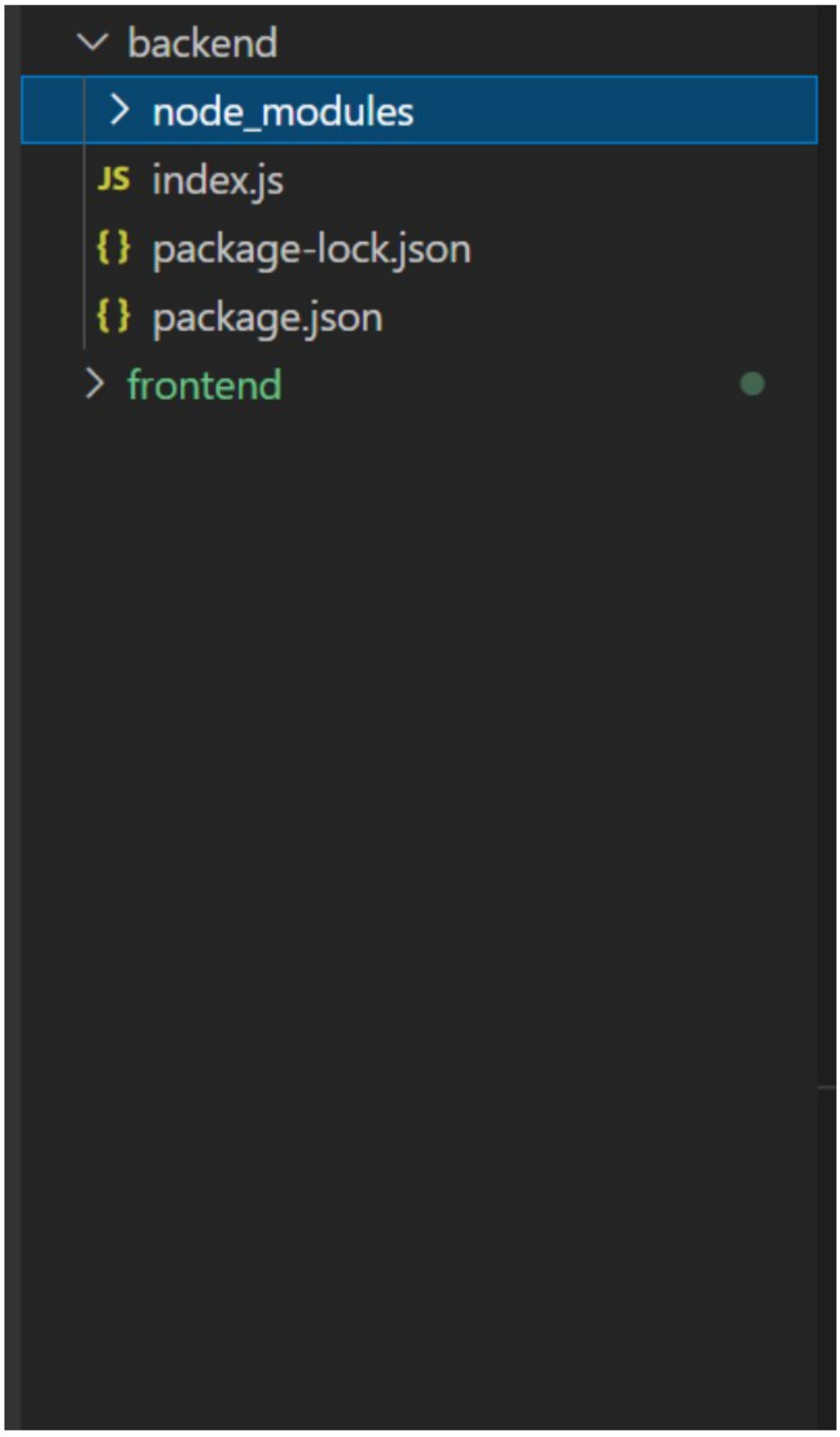
3.4 Project Structure-:

Front End

This is the whole Front End structure of our application. We have used different components for different objects



Back End



Code Structure for API-:

backend > JS index.js > ...

```
36
37  app.get('/user',(req,res)=>{
38
39      let qr = `select * from user`;      /// qr is the query ,
40
41      db.query(qr,(err,result)=>{
42          if(err){
43              console.log(err,'errs');
44          }
45
46          if(result.length>0){
47              res.send({
48                  meessage: 'all user data',
49                  data:result
50              })
51          }
52      });
53
54  });
55
56
57
58  // get single data
59  app.get('/user/:id',(req,res)=>{
60      let qid = req.params.id;                        // getting id from request
61      let qr =`select * from user where id=${qid}`
62      db.query(qr,(err,result)=>{
63          if(err){
64              console.log(err,'error aa gya')
65          }
66          if(result.length>0){
67              console.log('chal padi query')
68              res.send({
69                  data:result
70              });
71          }
72      });
73  });
74
75
76
77  // create data or post method
78
79  app.post('/user',(req,res)=>{
80
81      console.log(req.body,'createdata');
82
83      let name = req.body.name;
84      let email = req.body.email;
85      let marks = req.body.marks;
86
87      let qr = `insert into user(name,email,marks)
88      values('${name}','${email}', '${marks}')`;      //// query for inserting user table
89
90      db.query(qr,(err,result)=>{
91
92          if(err){
93              console.log(err);
94          }
95              res.send({
96                  message : 'data inserted....'
97              });
98
99
100     });
101
102
103
104
```

```
108
109  /// update single data or PUT method
110  app.put('/user/:id',(req,res)=>{
111      console.log(req.body,'updatedata');
112
113      let gId = req.params.id;
114      let name = req.body.name;
115      let email = req.body.email;
116      let marks = req.body.marks;
117
118      let qr = `update  user set name = '${name}', email = '${email}' , marks = '${marks}'
119      where  id = ${gId} `;
120
121      db.query(qr,(err,result)=>{
122          if(err){
123              console.log(err);
124          }
125          res.send({
126              message:'data updated'
127          });
128      });
129
130  });
131
132
133
134  // delete single data or DELETE method
135
136
137  app.delete('/user/:id',(req,res)=>{
138
139
140      let qId = req.params.id;
141
142      let qr = `delete from user where id = ${qId} ` ;
143      db.query(qr,(err,result)=>{
144          if(err){ console.log(err); }
145
146          res.send({
147              message : 'data deleted'
148          })
149      });
150  });
151
152
153  /// get data for teacher
154  app.get('/teacher',(req,res)=>{
155
156      let qr = `select * from teacher`;     /// qr is the query ,
157
158      db.query(qr,(err,result)=>{
159          if(err){
160              console.log(err,'errs');
161          }
162
163          if(result.length>0){
164              res.send({
165                  meessage: 'all teacher data',
166                  data:result
167              })
168          }
169      });
170
171  });
172
173
174
```

Fig – *API Code*

Connection of Database with server-:

backend > JS index.js > ...

```
1   const express = require('express');
2   const bodyparser =  require('body-parser');
3   const cors = require('cors');
4
5   const mysql =  require('mysql2')
6
7   const app = express()
8
9   app.use(cors());
10  app.use(bodyparser.json());
11
12
13
14  //database connection
15
16
17  const db =  mysql.createConnection({
18          host:'localhost',
19          user:'root',
20          password: '',
21          database: 'simpledb',
22          port:3308
23  });
24
25  ///check database connection
26
27  db.connect(err=>{
28      if(err){
29          console.log('errd');
30      }
31      console.log('database connected...')
32  })
33
34
```

Fig – *Database Connection*

Front End Code structure-:

rontend > src > app > create > create.component.html > div.container.mt-5 > form > div.col-lg-4.mt-2

```
<div class="container mt-5">


    <!-- show error msg -->
    <div *ngIf="errormsg" class="alert alert-danger alert-dismissible fade show" role="alert">
        <strong>{{errormsg}}</strong>
        <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
      </div>


<!-- show success msg -->
    <div *ngIf="successmsg" class="alert alert-success alert-dismissible fade show" role="alert">
        <strong>{{successmsg}}</strong>
        <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
      </div>


    <form [formGroup]="userForm" >
        <div class="col-lg-4 mt-2">
            <input type="text" class="form-control" formControlName="name" placeholder="Enter Full Name">
        </div>

        <div class="col-lg-4 mt-2">
            <input type="text" class="form-control" formControlName="email" placeholder="Enter Email">
        </div>

        <div class="col-lg-4 mt-2">
            <input type="text" class="form-control" formControlName="marks" placeholder="Enter marks">
        </div>

        <div class="col-lg-4 mt-2" *ngIf="!getparamId"  (click)="userSubmit()">
           <button class="btn btn-primary btn-sm">Create</button>
        </div>

        <div class="col-lg-4 mt-2" *ngIf="getparamId" (click)="userUpdate()">
            <button class="btn btn-secondary btn-sm">Update</button>
          </div>

    </form>

</div>
```

Fig – *Create Model*

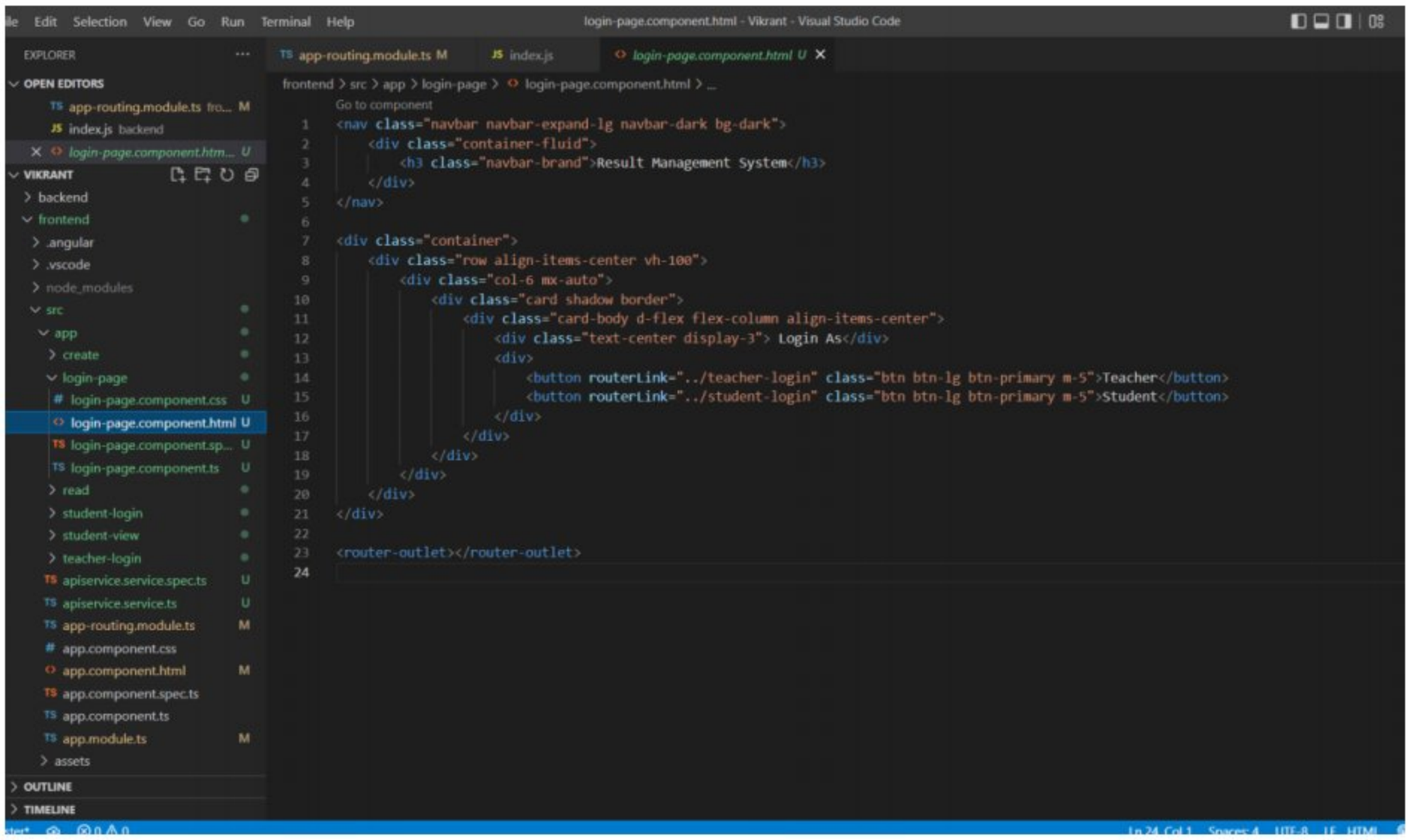


Fig – *Main page*

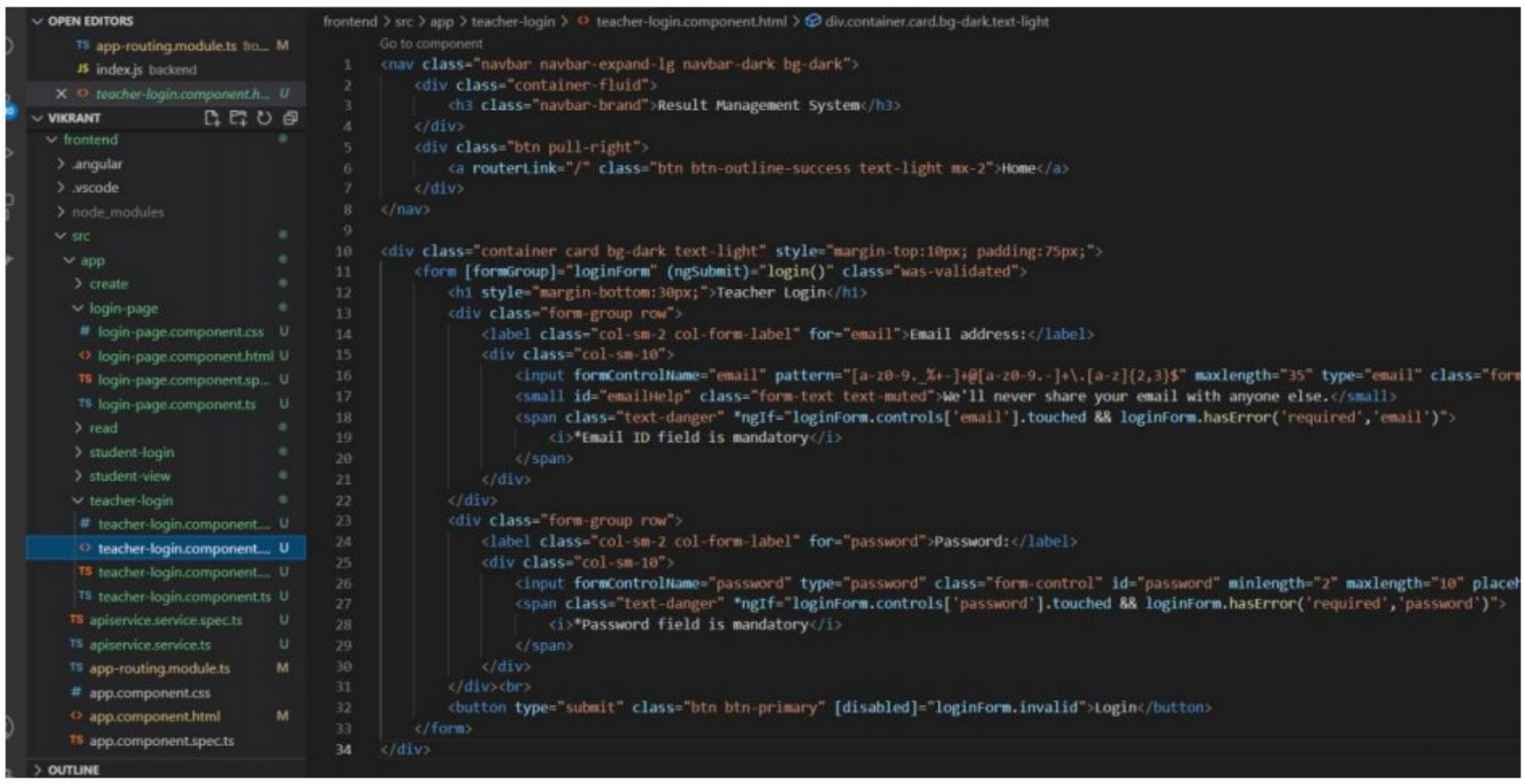


Fig – *Teacher Login*

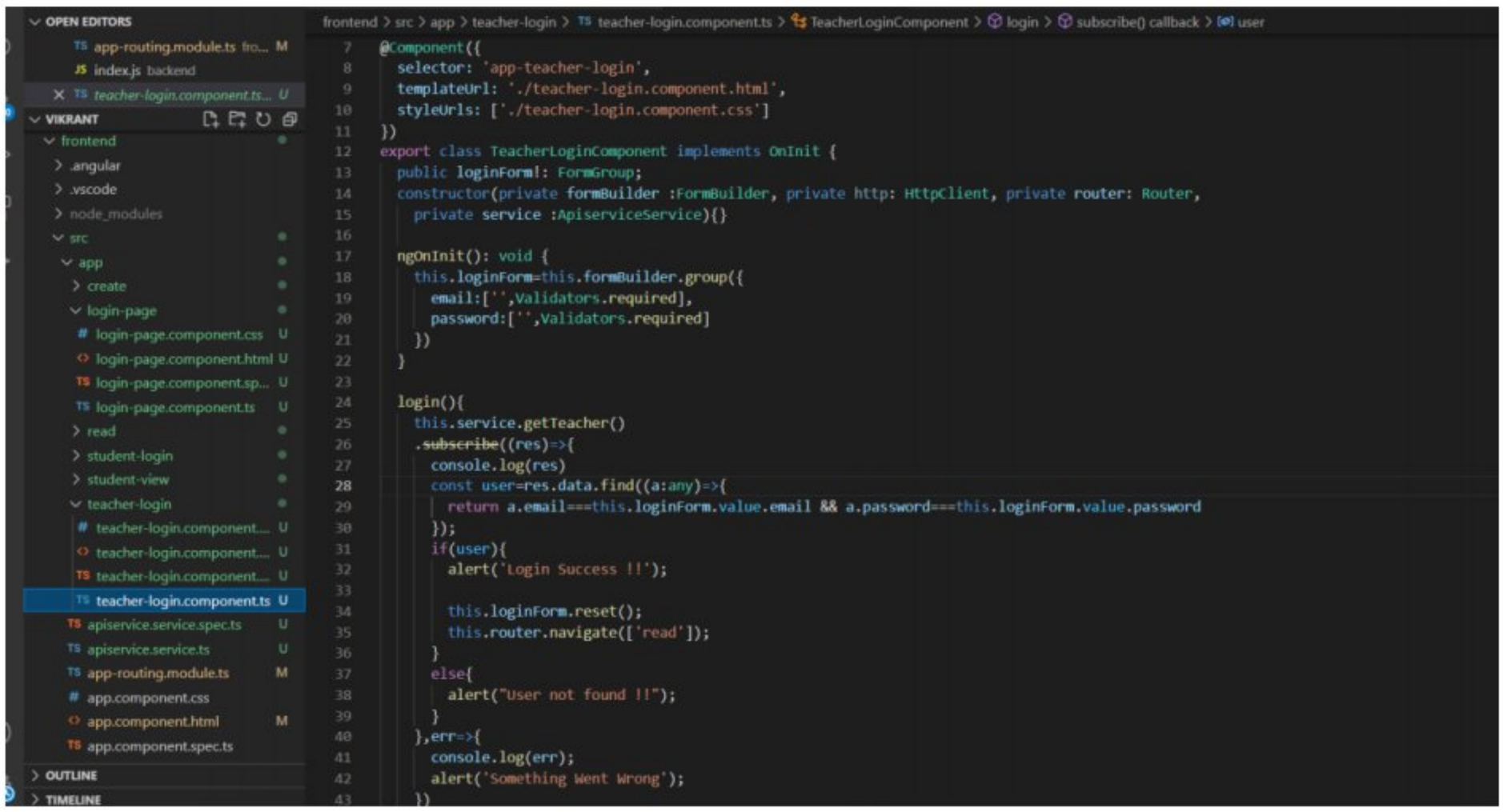


***Fig –** Teacher Login Functionality*

***Fig –** api services in frontend*

Chapter 4 Performance Analysis

Code execution-:

After the implementation of the code as above, we ran the application on localhost : 4200 and backend is running on http://127.0.0.1:5000/ .

Output /Screenshots -:

This is the main page or we can say that the first page of our application.

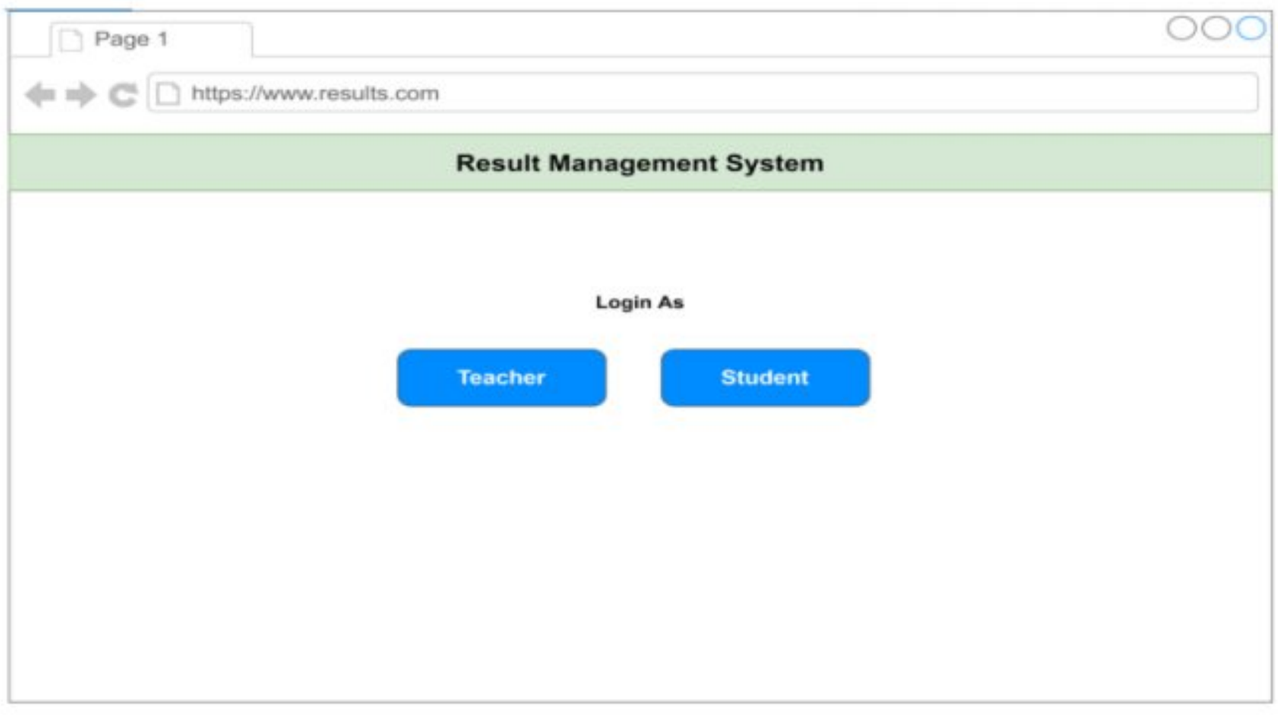


Fig – *api services in frontend*

This is the teacher view, this page shows after the teacher has logged in. This shows all the records of each student.

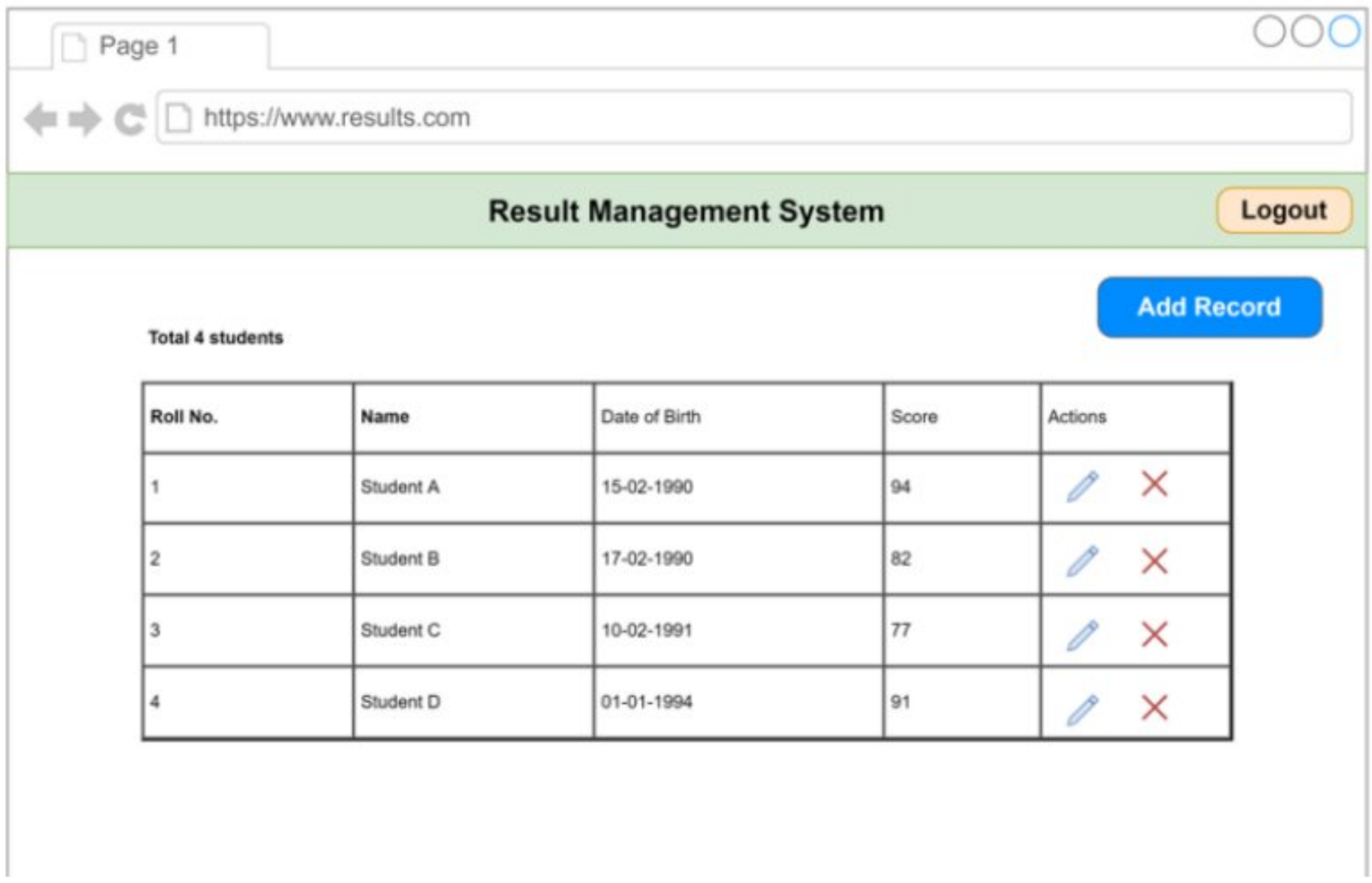


Fig – *Teacher-view model*

This is the student view model. In this form student has to fill their roll no and name to get the details of their.

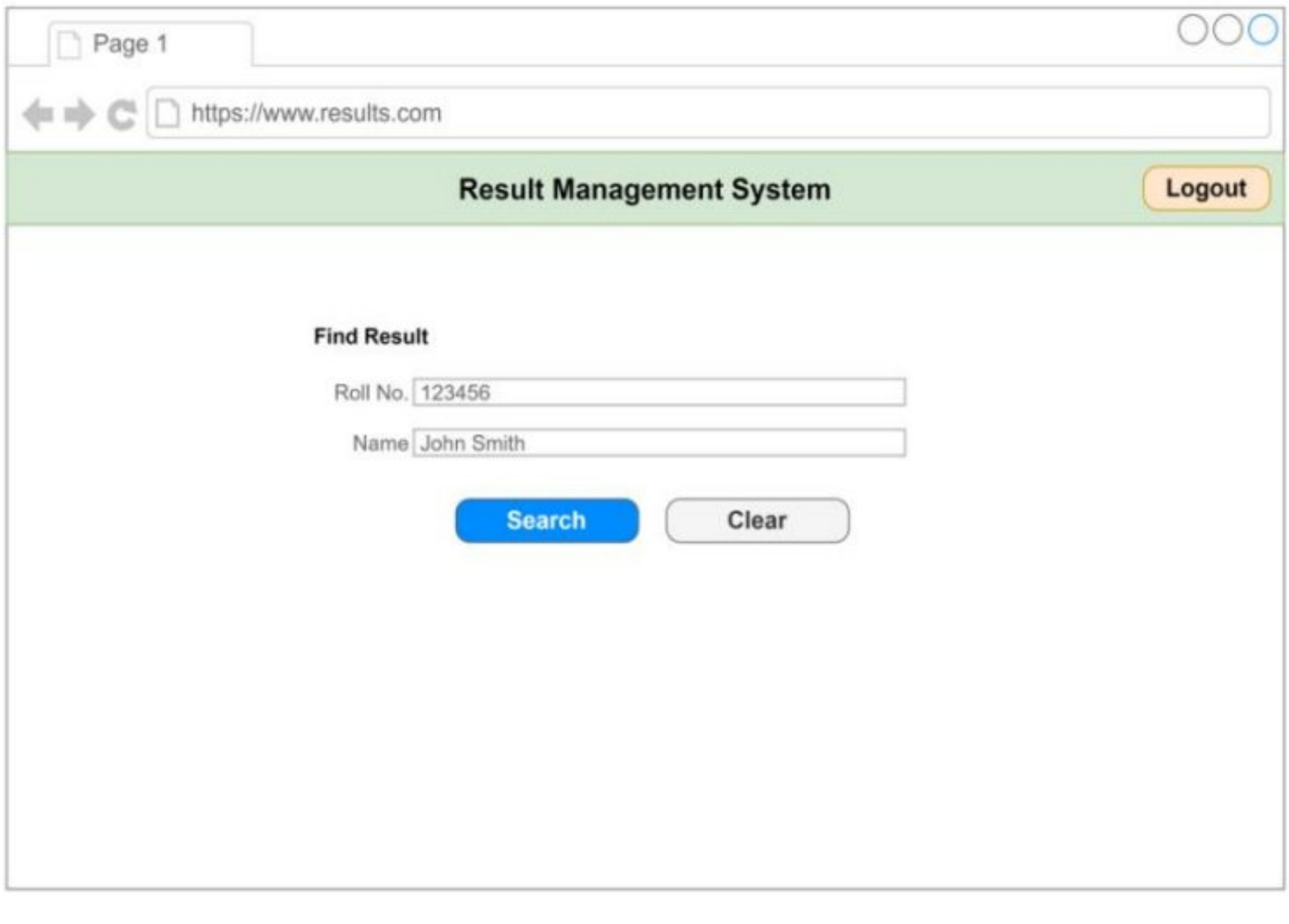


Fig – *student-view model*

This form is for adding new record of student. This can be performed by teacher only.

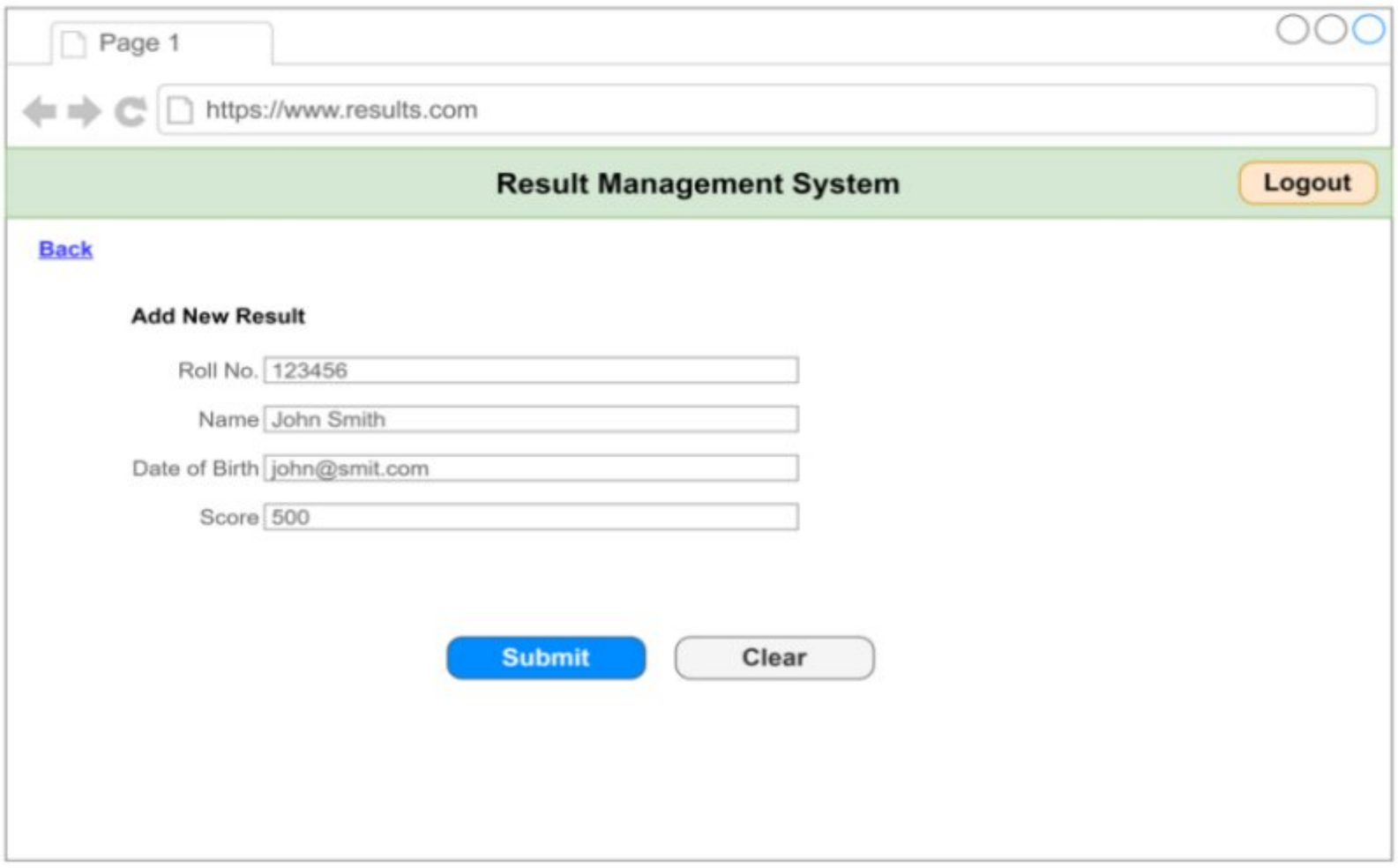


Fig – *Adding new result*

The result output contains the result of the corresponding student. It contains the marks , name , date of birth , roll no of student.

As shown in figure below-:

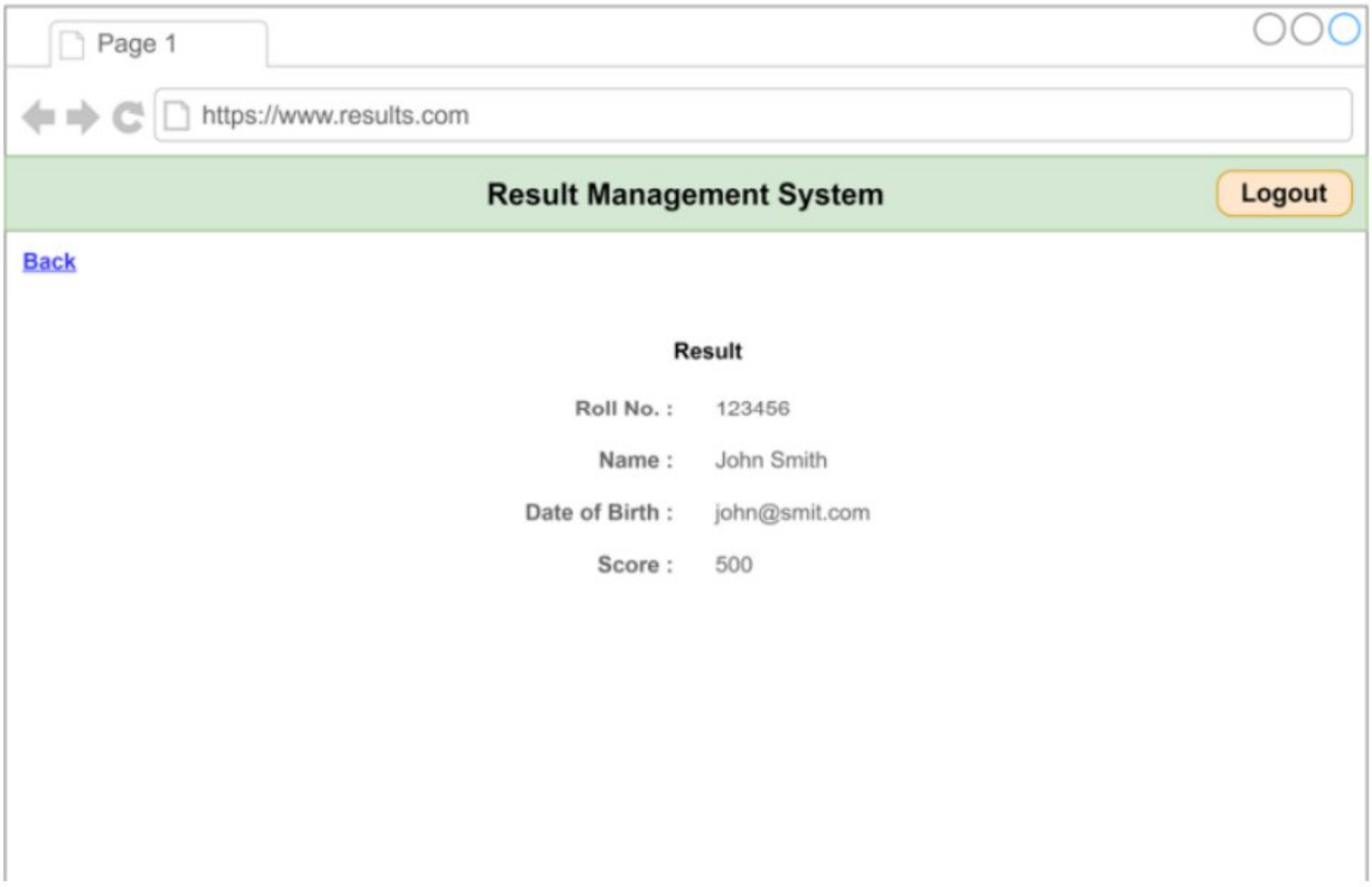


Fig – *Result of student*

Any project before exposed to user must be tested to ensure that it behaves as expected. In this project, the application is tested by giving various types of input to check whether they are being validated or not and whether the application is behaving as expected or not.

Following the testing and deployment of the application, it must be maintained to meet various limitations such as availability, reliability, and so on. Depending on the success or input of the users, future versions of the applications can be generated.

Chapter 5 CONCLUSIONS

5.1 Conclusions

It's usually a good idea to go with a student information system that's built on a current system architecture to keep up with changing needs. This system should include well-organized data coding and clearly defined business applications.

The system's overview elucidates the convenience of exact data delivery at the tip of your fingertips, increasing student retention and teaching them how to manage their time effectively.

The proposed method is efficient and user-friendly, based on the results of the experiments and tests. In comparison to current methods of managing academic institutions, this project, which produces centralised software, makes work administration and management easier and gives full information about the issue of users' interest with just one mouse click. An easy-to-use user interface centralised software can be offered to the educational institution, allowing all services linked with the university to interact with one another and share data. The user will be able to access the resources from afar because this is a ReST API . Because the application is built with a microservice architecture and agile methodology, services can be added in the future.

Advantages of the application -:

For administration

- Teachers, professors, and staff can prepare result analysis reports using the student result management system.

- Provide a single point of contact for the compilation of all internal examination reports.

- Obtain quick prints of examination and test reports.

- In no time, you'll be able to calculate scores, percentages, and grades. In addition, the workload will be much reduced.

- Records marks and result on a single database which reduces the cost in efficient manner also there will be no need of physical storage anymore .

For Students

- Download a PDF version of the findings report. Students can now access their grades via any internet platform. There is no need to go to college or school physically.

- With a valid Roll number/ID, you can look up test and exam results.

- Re-evaluation is requested. Students can request a re-evaluation online with no additional requirements.

Disadvantages of the application -:

- Hackers are prone to gaining access to the student result management system.

- After the deadline, the administration is unable to update or modify scores. Because data is stored in the database.

- The application's extensive modules and features make it tough for a user to use it.

- Minor glitches or bugs can turn application crashed. Heavy traffic on application can cause problem for web application.

5.2 Future Scope

The project's future potential is enormous. In the future, the project could be deployed on an intranet. Because it is quite versatile in terms of expansion, the project can be upgraded in the near future as and when the need arises. The customer may now manage and thus run the complete task in a lot better, accurate, and error-free manner now that the planned database Space Manager software is ready and fully functional. The following is the project's future scope.

In the future students may also be able to post or download notes in the future. There will be a few little modifications here and there to make the app more visually appealing, as well as statistics, tracking, and analytics. For significantly more, the entire project will be made available as an Android app.

Artificial intelligence's contribution to several fields is growing all the time. So in the future we will be using some machine learning algorithms in application, to make it more efficient.

Bar code reader based attendance system will be used in future. Since it is more secure and efficient for teacher to access the details of any student.

In future student can be login using the id or photo. We should continue to think on what is happening and consider whether things could be made better.

REFERENCES

[1] Christ University https://christuniversity.in/StudentLogin.html

[2] PESIT University https://pesuacademy.com/Academy

[3] Application development using https://angular.io/docs

[4] Error handling using https://stackoverflow.com/

[5] For testing API https://www.postman.com/