

ABSTRACT

The "Examination Portal" is a web-based application designed to streamline and enhance the examination process for educational institutions. Built using the MERN stack (MongoDB, Express.js, React.js, Node.js), this portal allows for efficient creation, scheduling, and management of exams.

Institutions can manage exams seamlessly, providing secure access to exam materials and real-time monitoring of ongoing exams. Instructors can easily update exam details and course information, while students benefit from an intuitive interface to access their exams and view results. The system generates detailed reports and analytics, offering valuable insights into student performance and aiding in decision-making.

This project aims to modernize the traditional examination process, making it more efficient, secure, and user-friendly for both educators and students.

Technology Used For:

Web Application:

Frontend: React.js

Backend: Express.js, Node.js

Database: MongoDB

IDE: Visual Studio Code

PROJECT DESCRIPTION

MODULE DESCRIPTION

- LOGIN MODULE
- STUDENT MODULE
- FACULTY MODULE
- PARENT MODULE

LOGIN PAGE

- Handles user authentication and session management.
- Provides role-based access for Students, Faculty and Parents.
- Utilizes JSON Web Tokens (JWT) for secure session handling.

STUDENT MODULE

- Dashboard with announcements, upcoming exams, and results info.
- Detailed course information and syllabus download.
- Exam schedule and streamlined exam-taking process.
- Access to exam results and password management.
- Feedback system for course and exam experiences.

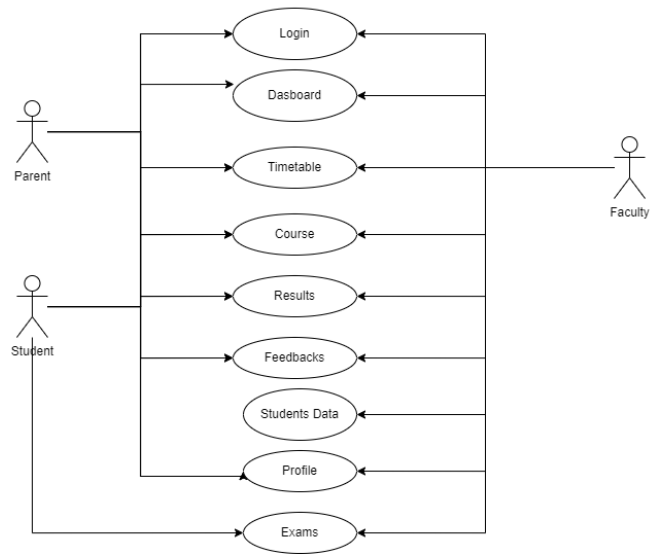
FACULTY MODULE

- Dashboard for announcements, feedback, and course management.
- Course management: add, update, delete courses, and syllabus download.
- Exam scheduling and invigilation tools.
- Results management and access to student performance data.
- Feedback system for course and exam experiences.

PARENT MODULE

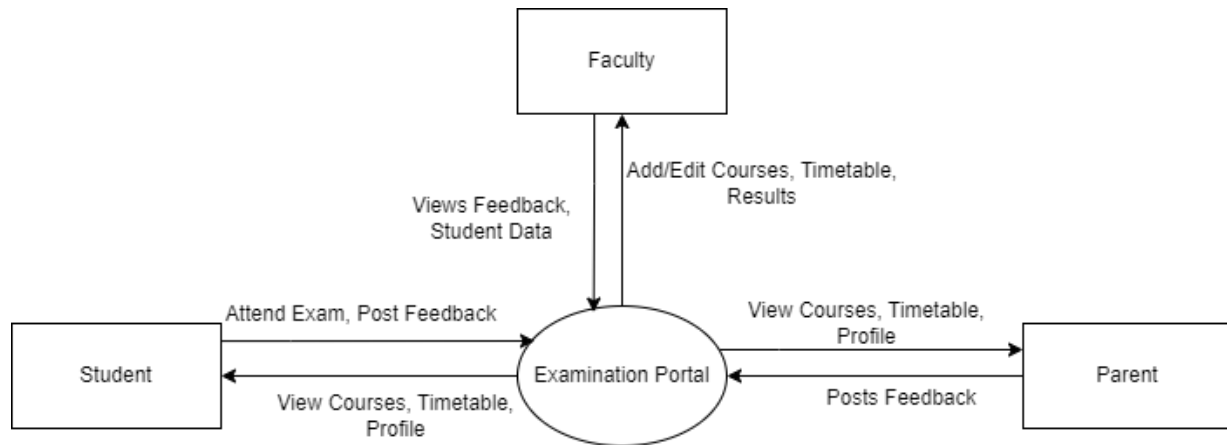
- Dashboard with announcements, upcoming exams, and results info for child.
- Course details and syllabus download.
- Access to child's exam results and profile management.
- Feedback system for course and exam experiences.

USE CASE DIAGRAM FOR EXAMINATION PORTAL:



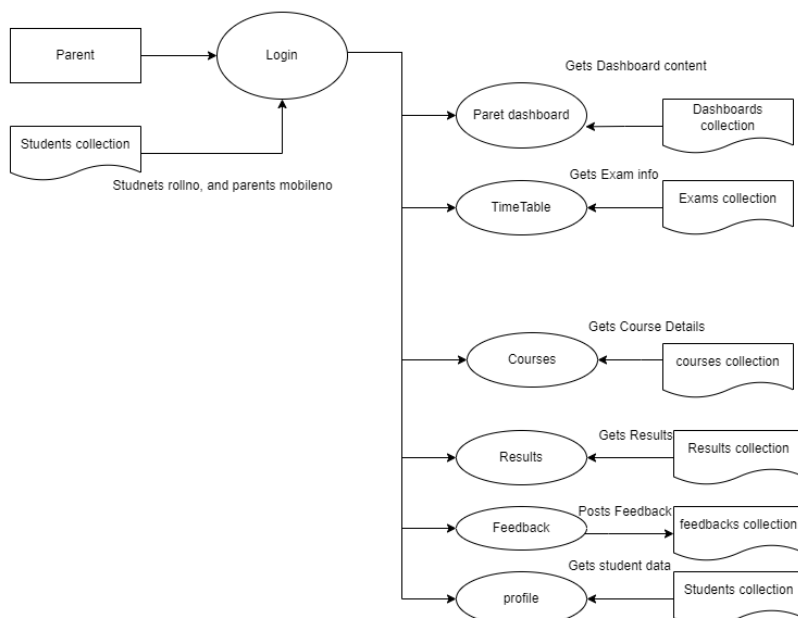
DATA FLOW DIAGRAM:

Level 0 DFD:

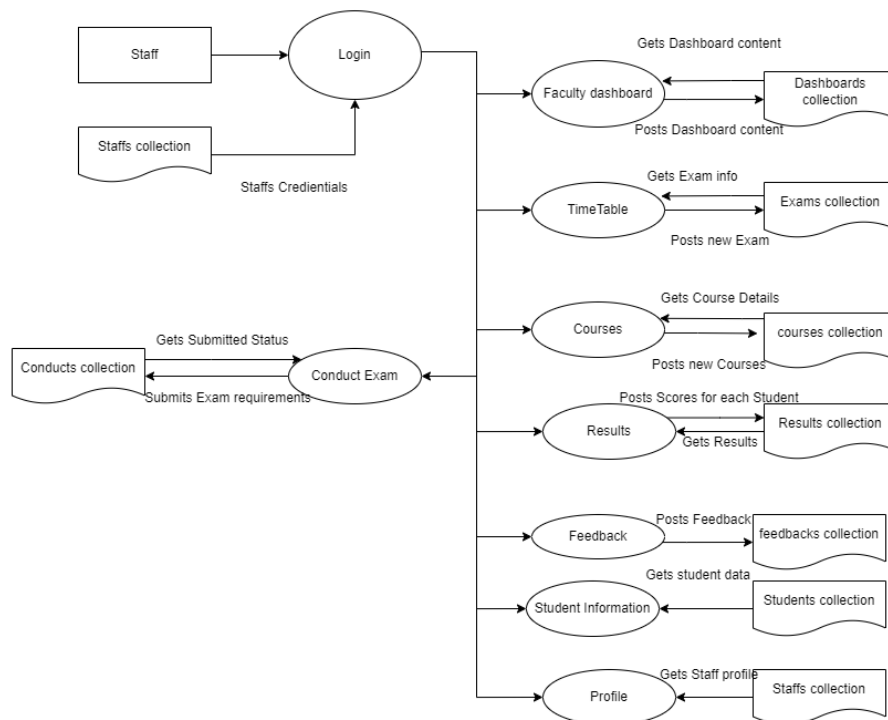


Level 1 DFD:

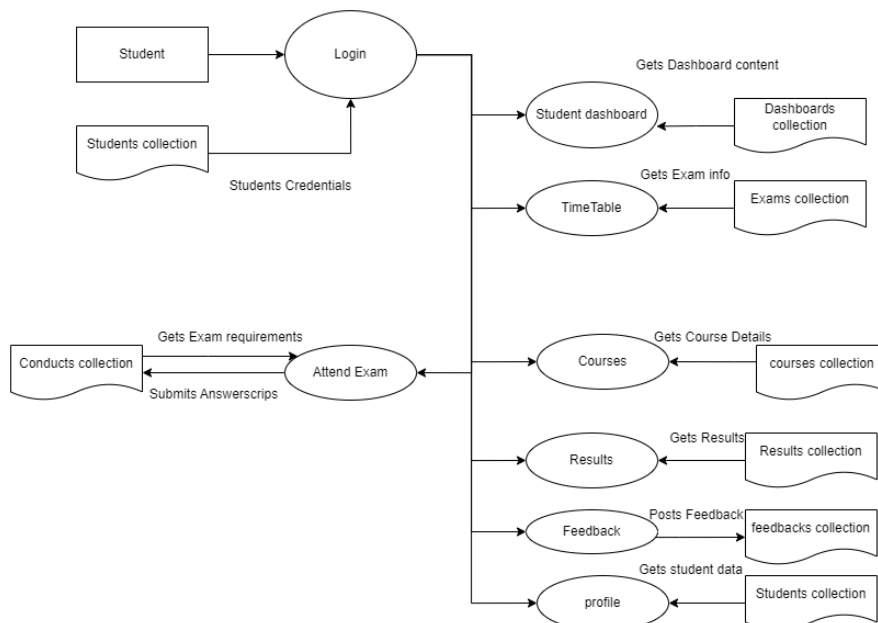
DFD For Parent:



DFD For Faculty:



DFD For Student:



DATABASE DESIGN

1.DOCUM ENT NAME: COURSES

DOCUMENT DESCRIPTION: Details of the courses.

```
{
  courseCode: type: {String, required: true, unique: true,}
  courseName: { type: String, required: true,}
  courseType: {type: String, enum: ['Core', 'Elective'], required: true, },
  createdAt: {type: Date, default: Date.now, }
}
```

2. DOCUMENT NAME: DASHBOARD

DOCUMENT DESCRIPTION: Details of the Dashboard Information.

```
{
  _id: { type: Number, default: 1
  message: String,
  date: { type: Date, default: Date.now }
}
```

3. DOCUMENT NAME: EXAMS

DOCUMENT DESCRIPTION: Details of the Exams information.

```
{
  examName: {type: String, required: true,unique:true },
  subjects: {
    subjectName: {type: String, required: true },
    courseCode: {type: String, required: true, unique: true },
    date: {type: Date, required: true },
    startTime: {type: Date, required: true },
    endTime: {type: Date, required: true },
  }
}
```

4. DOCUMENT NAME: FACULTIES

DOCUMENT DESCRIPTION: Details of the Faculty Information.

```
{
  staffid: { type: String, required: true, unique: true },
  fullName: { type: String, required: true },
  dob: { type: Date },
  fatherName: { type: String },
  address: { type: String },
  mobile: { type: String },
  parentNo: { type: String },
  bloodGroup: { type: String },
  dept: { type: String }
}
```

5. DOCUMENT NAME: FEEDBACKS

DOCUMENT DESCRIPTION: Details of the feedbacks information.

```
{
  username: { type: String, required: true },
  userType: { type: String, required: true },
  feedbackType: { type: String, required: true },
  message: { type: String, required: true },
  date: { type: Date, required: true }
}
```

6. DOCUMENT NAME: RESULTS

DOCUMENT DESCRIPTION: Details of the Results information.

```
{
  examName: { type: String, required: true },
  subjects: {
    subject: { type: String, required: true },
    students: {
      rollNo: { type: String, required: true },
      score: { type: Number, required: true },
      outOf: { type: Number, required: true },
      gradePoint: { type: Number },
      grade: { type: String },
      isAbsent: { type: Boolean, default: false }
    }
  }
}
```

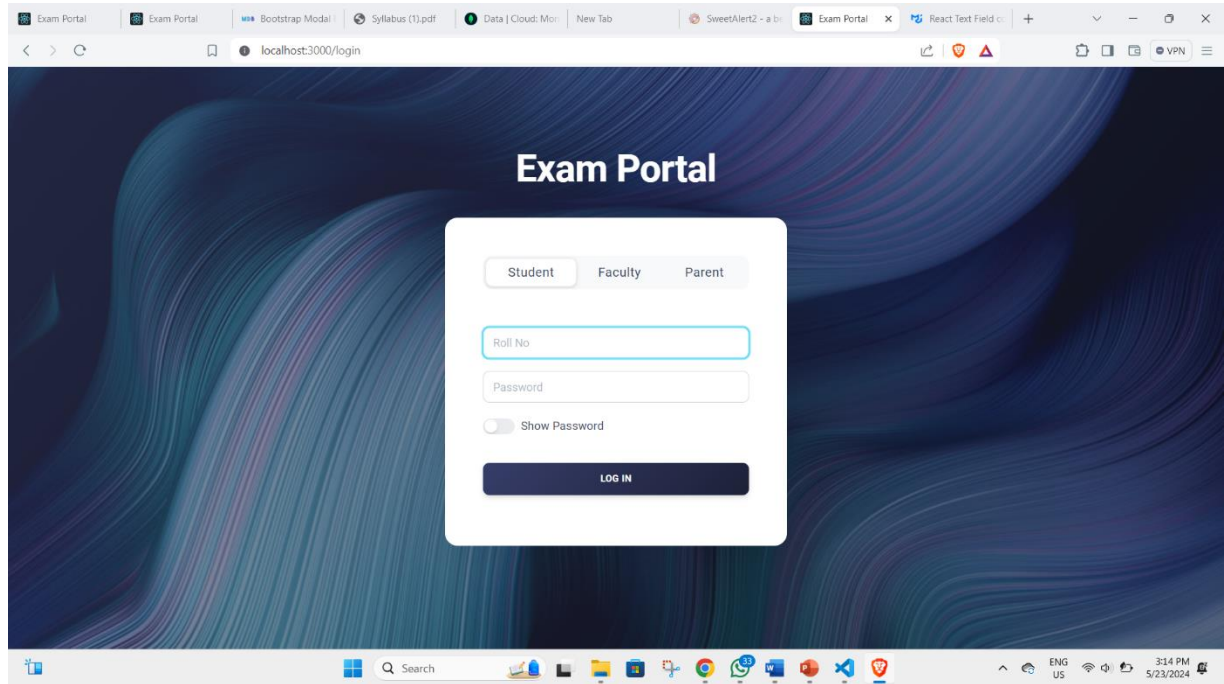

7. DOCUMENT NAME: STUDENTS

DOCUMENT DESCRIPTION: Details of the Students information.

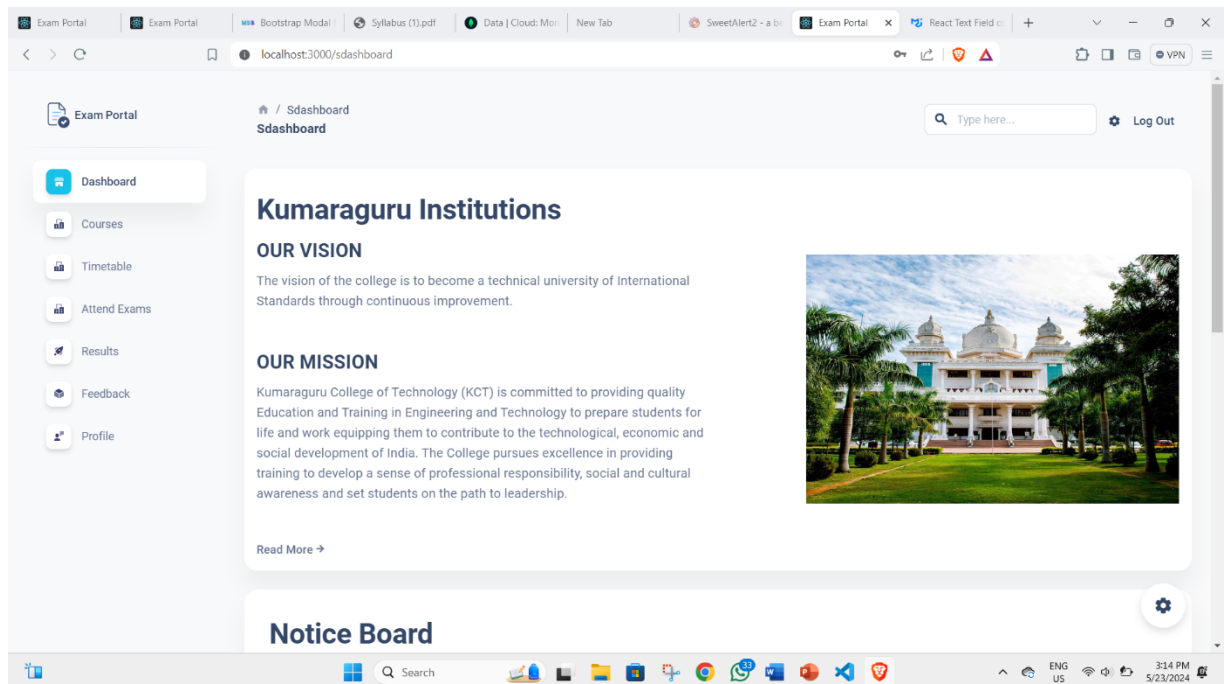
```
{
  rollNo: { type: String, required: true, unique: true },
  fullName: { type: String, required: true },
  dob: { type: Date },
  fatherName: { type: String },
  motherName: { type: String },
  address: { type: String },
  mobile: { type: String },
  parentNo: { type: String },
  bloodGroup: { type: String },
  course: { type: String }
}
```

FORM DESIGN

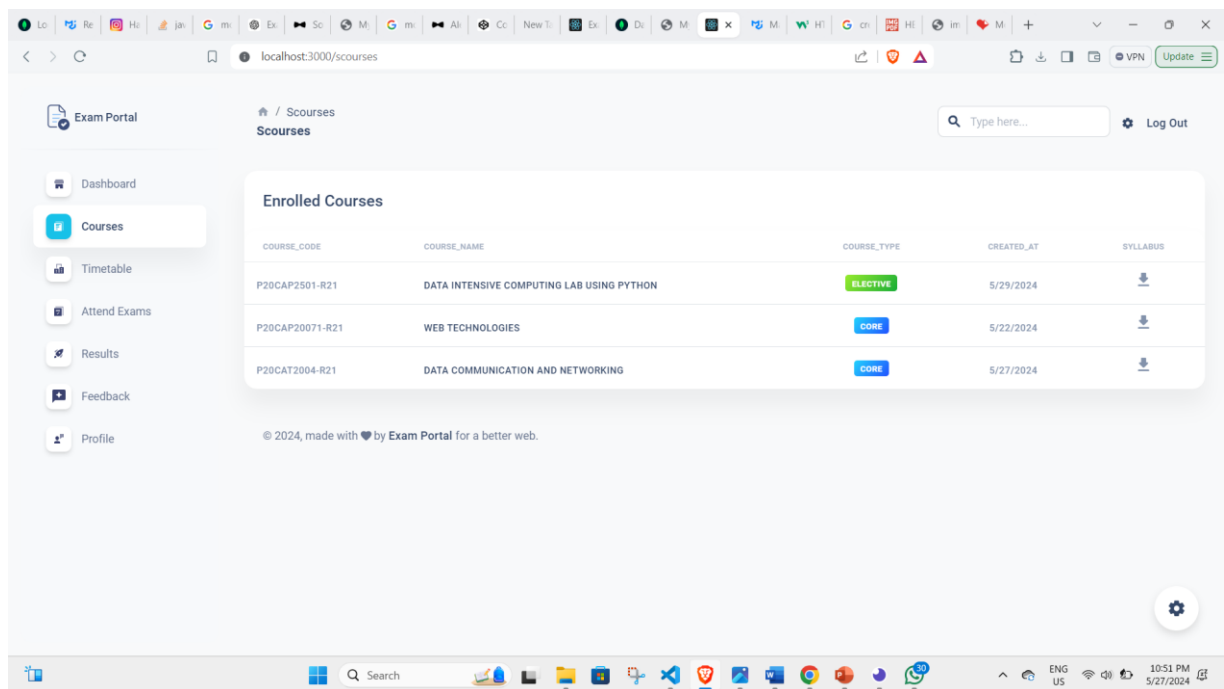
SCREENSHOT



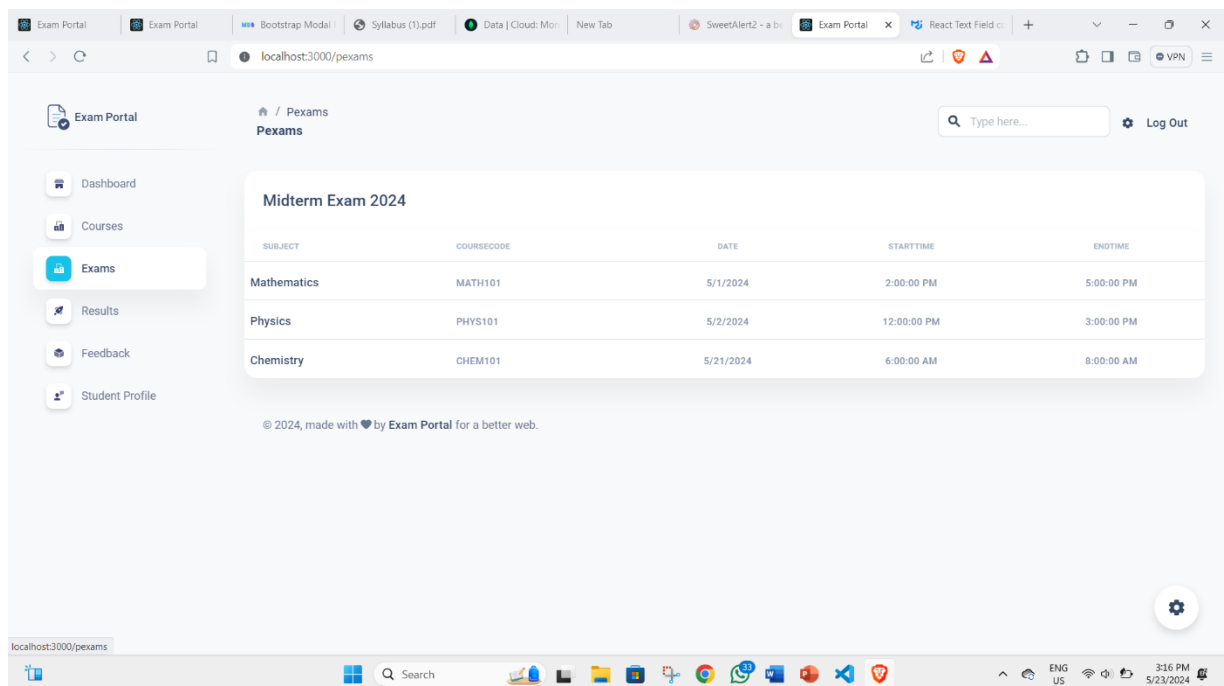
Login Page



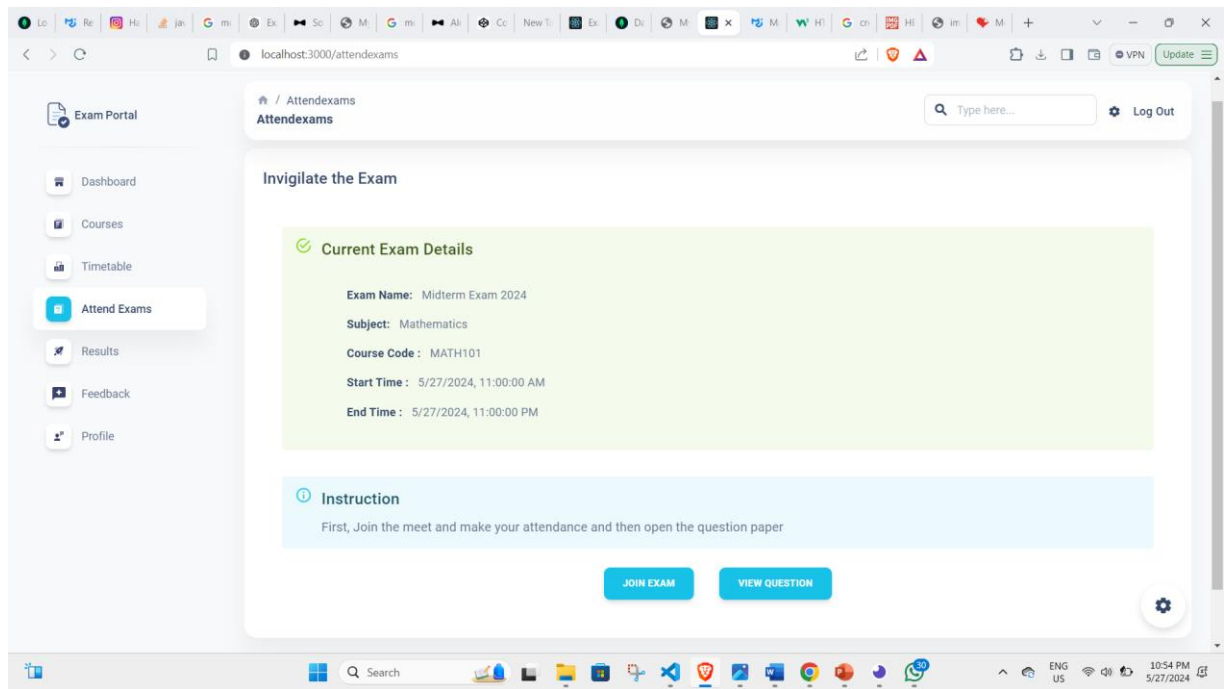
Dashboard Page



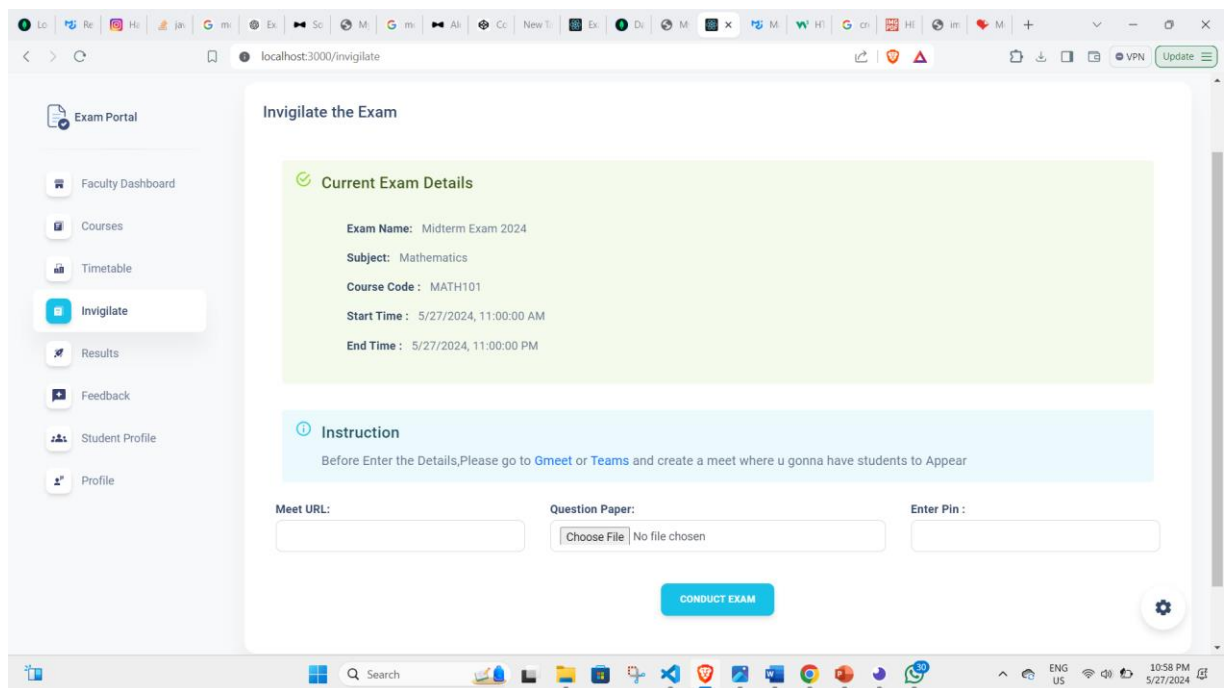
Course Page



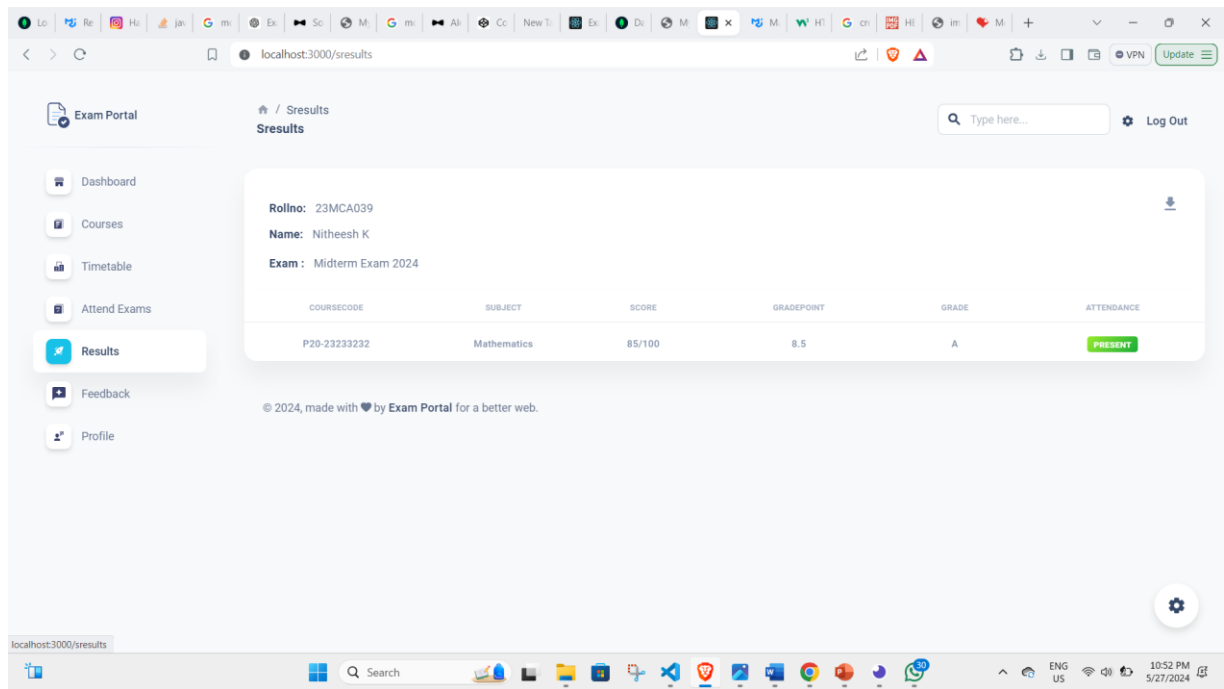
Timetable Page



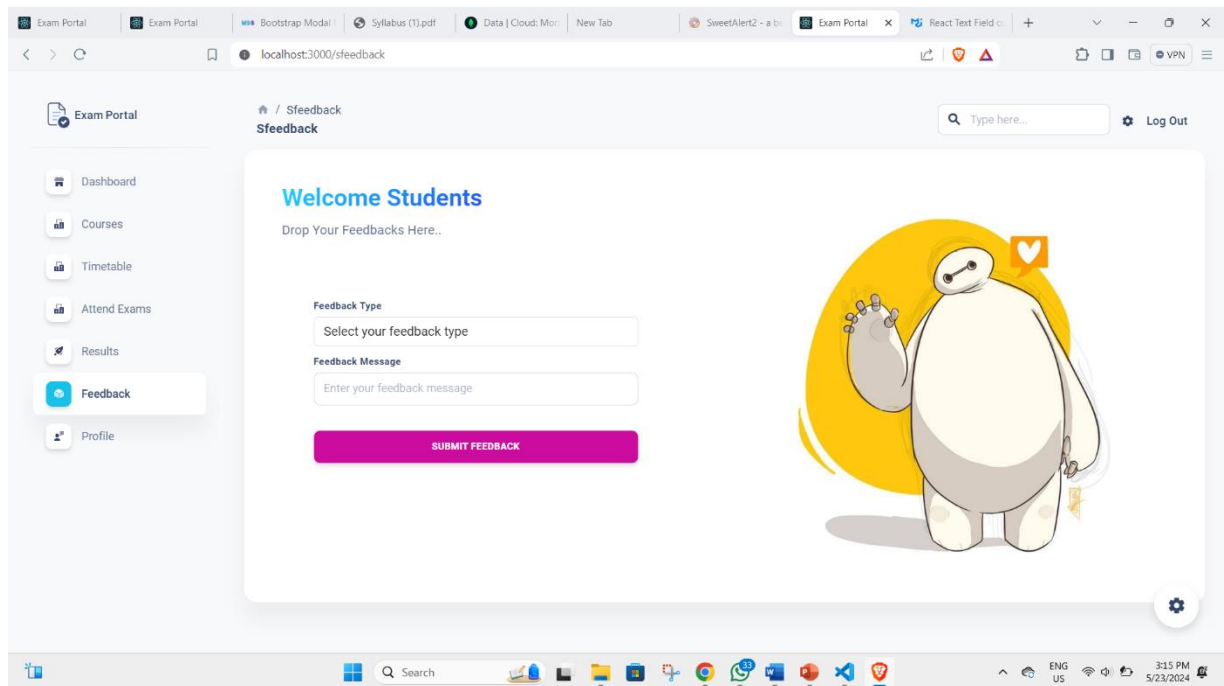
Attend Exam Page



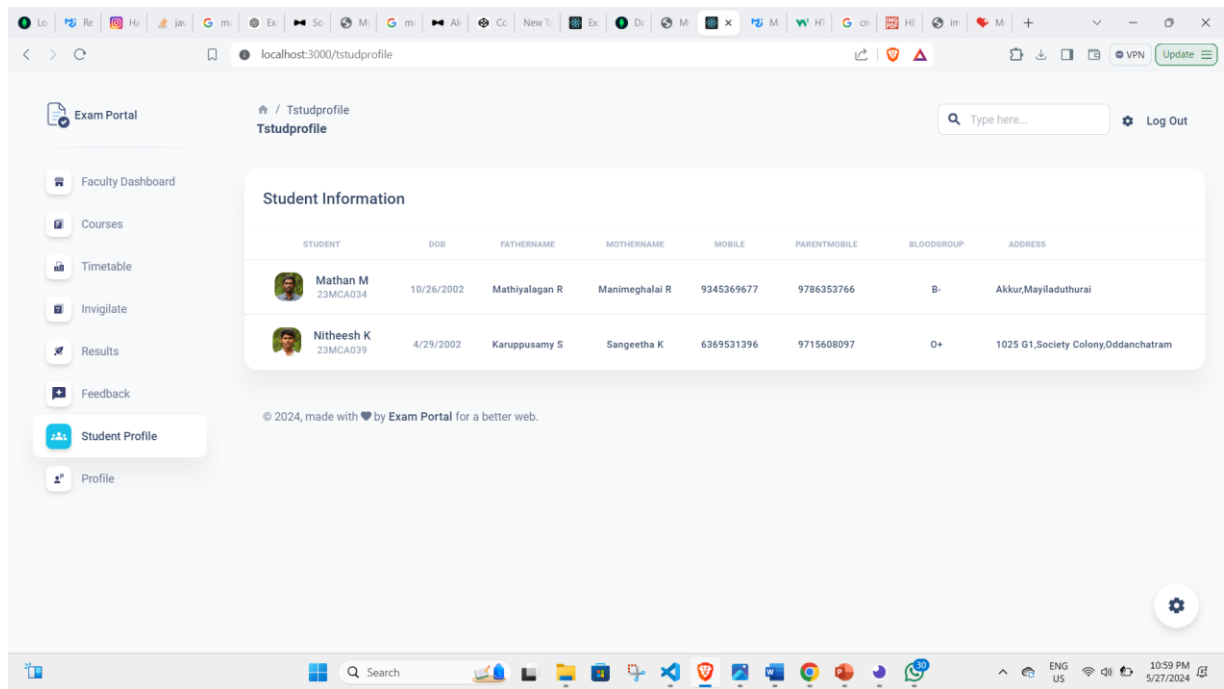
Invigilate Exam Page



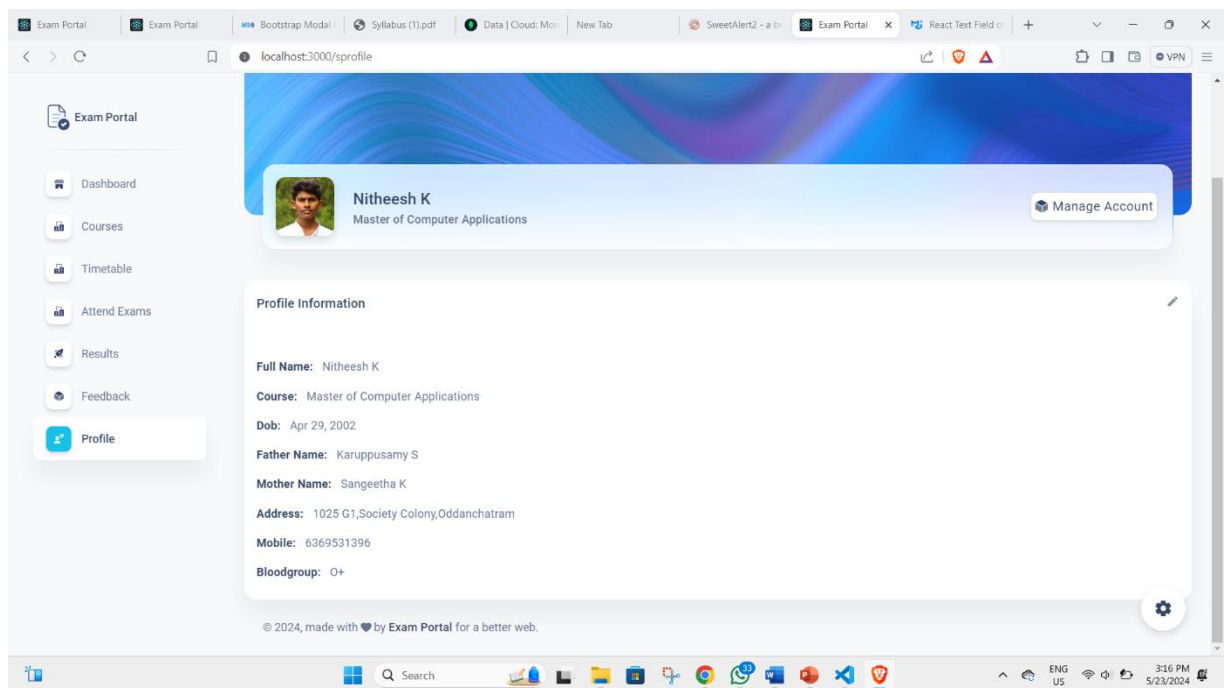
Result Page



Feedback Page



Student Information Page



Student Profile Page

TESTING

PERFORMANCE TESTING

Each and every component of the “**Examination Portal**” web application went through each stage of the approached methodology. It was identified that each unit behaved differently as a single entity and differently when it was integrated with other units in respect of speed. The test environment of the application plays a crucial role on its performance. Moreover the technical specifications of the platform where the application is installed plays a major role.

SYSTEM TESTING

In system testing, integration testing passed components such as report issues, adding project, user registration, custom fields, and tags are taken as input. The system elements are tested. The environment and behaviour of the system and its elements is tested here. The administrator system fulfils the objective in efficient manner. The administrator deals the processes with less expectation of system resources and user friendly with the help of software which takes care of the system resources.

REGRESSION TESTING

Whenever a change in a software application is made, it is quite possible that other areas within the application have been affected by this change. Regression testing is performed to verify that a fixed bug hasn't resulted in another functionality or business rule violation. The intent of regression testing is to ensure that a change, such as a bug fix should not result in another fault being uncovered in the application. Our application “Examination portal” is tested properly then and there whenever the bugs are being fixed, to ensure that it doesn't affect other modules.

UNIT TESTING

Unit testing is a level of software testing where individual units/components of software are tested. It is necessary to test each module by giving them different kind of inputs. This ensures the stability and consistency of the application.

TEST CASE ID	TEST DESCRIPTION	EXPECTED RESULT	ACTUAL RESULT	RESULT
T1	Login	Verify if user credentials are valid	User is successfully logged in	PASS
			Invalid credentials provided	FAIL
T2	Managing Courses	Ensure all required fields are filled correctly	Courses is successfully added to the courses	PASS
			Missing information leads to failed courses CRUD	FAIL
T3	Managing Exams	Verify changes are correctly updated	Exams details are successfully updated	PASS
T4	Managing Results	Process students results without errors	Results are successfully processed	PASS
			Errors encountered during order processing	FAIL
T5	Logout	Ensure User session is successfully terminated	User is logged out	PASS
T6	Attend Exam	Ensure meet link is working	Join the meet	PASS
			Errors encountered during importing link	FAIL
T7	Conduct exam	Ensure I gets and posts data correctly	Exam data is successfully recorded	PASS
			Failed to process the data	FAIL

CONCLUSION AND FUTURE ENHANCEMENT

CONCLUSION

The Examination Portal serves as a comprehensive platform for efficient exam management, student assessment, and academic performance analysis. It has facilitated seamless coordination between faculty, students, and parents, ensuring streamlined exam conduct and result processing. The portal's user-friendly interface, real-time monitoring capabilities, and secure data management have significantly enhanced the examination process, leading to improved academic outcomes and student satisfaction.

FUTURE ENHANCEMENT

In the future the system will have the following enhancements,

- **Advanced Analytics:** Implement advanced data analytics for deeper insights into exam trends and student performance.
- **Enhanced Security:** Integrate additional security measures like multi-factor authentication and data encryption.
- **LMS Integration:** Connect with Learning Management Systems for seamless course and exam management.
- **Mobile App:** Develop a mobile app version for convenient access and real-time notifications.
- **Feedback System:** Incorporate a feedback mechanism for students to provide input on exams and courses.
- **AI Proctoring:** Explore AI-based proctoring solutions for secure remote exams and cheating prevention.