A

Project Report
On

"COLLAGE ADMISSION SYSTEM"

Submitted to

CH. SHIVAJI UNIVERSITY, KOLHAPUR

In partial fulfilment of

BACHELOR OF COMPUTER APPLICATION

Submitted By

Mr. RANDIVE SHUBHAM BALAVANT

Mr. PATIL MANOJ MAHIPATI

Under the guidance of

Asst.Prof. R. D. BUDKE

Through

The Principal

Prof. Dr. U . R . S H I N D E

MAUNI VIDYAPEETH'S

KARMVEER HIRE MAHAVIDYALAY, GARGOTI.

TAL – BHUDARGAD DIST - KOLHAPUR

2024-2025

MAUNI VIDYAPEETH'S

KARMVEER HIRE MAHAVIDYALAY, GARGOTI.

CERTIFICATE

This is to certify that,

1. Mr. Randive Shubham Balavant
2. Mr. Patil Manoj Mahipati

Have successfully completed the project on the topic "Collage Admission System" in satisfactory manner for partial fulfilment of Bachelor Of Computer Application degree for the academic year 2024-2025.

To the best of our knowledge and belief, the matter presented here is original and has not been submitted elsewhere for the award of any degree.

Date: / /2025

Place: Gargoti

Asst.Prof. R. D. Budke		Asst.Prof. R. D. Budke
Project Guide	EXAMINER	H.O.D

MAUNI VIDYAPEETH'S

KARMVEER HIRE MAHAVIDYALAY, GARGOTI.

To

The Registrar,

Shivaji University,

Kolhapur.

Subject: -Recommendation for BCA III Project.

Respected Sir,

I, Prof. Dr. U. R. Shinde the principal of K. H. College Gargoti, recommended that the following students,

1. Mr. Randive Shubham Balavant
2. Mr. Patil Manoj Mahipati

Have completed the project report entitled, "Collage Admission System" As per partial fulfilment according to the syllabus of the Shivaji University, Kolhapur under guidance of Asst. Prof. R. D. Budke.

Place: - Gargoti

Date: - / / 2025

Prof. Dr. U. R. Shinde

Principal

MOUNI VIDHYAPEET'S
KARMVEER HIRE MAHAVIDYALAY, GARGOTI
TAL – BHUDARGAD DIST. KOLHAPUR

GUIDANCE

This is to certify that the project entitled "Collage Admission System" conducted at K.H. Collage, Gargoti by Mr. Randive Shubham Balavant and Mr. Patil Manoj Mahipati. In partial fulfilment of this work for award of Bachelor of Computer Science submitted to Shivaji University, Kolhapur has been completed under my supervision and guidance.

To the best of my knowledge and belief, the presented by them is original in nature and has not been from any source. In addition, this report has not been submitted earlier for any degree or diploma of Shivaji University or any other university.

Place: Gargoti.
Date: / /2025

Asst. Prof. R. D. Budke
(Project Guide)

CERTIFICATE

“ Collage Admission System ”

This is to certify that Mr. Randive Shubham Balavant and Mr. Patil Manoj Mahipati the students of K. H. Collage, Gargoti have developed this project report entitled “Collage Admission System” for our institute.

The data collected and processed is according to our requirement of system. Their work is up to the mark of satisfaction and that was good. As per my knowledge it is their original work and it is carried out very much sincerely.

I wish that all success in their future.

Place: Gargoti

Date: / / 2025

DECLARATION

We undersigned hereby declare that this report entitled “Collage Admission System” for Karmveer Hire Arts, Science, Commerce and Education Collage, Gargoti is our original work prepared under guidance of Asst.Prof. R. D. Budke The Empirical finding in this report is based on data collected by us. The matter presented is this report is not copied from any source.

We understand that such copy is liable for punishment in any way the university Authorities deem to fit. This work has not been submitted to either Shivaji University or any other University.

This work is humbly submitted to Shivaji University, Kolhapur for the award of the degree of Bachelor of Computer Science.

Place: Gargoti

Date: / /2025

Mr. Randive Shubham Balavant

Mr. Patil Manoj Mahipati

ACKNOWLEDGEMENT

It gives great pleasure to remain deeply indebted to our guide Asst.Prof

R. D. Budke for providing us with the required facilities for the academic achievement under whom we had the privilege to work. The faith and Confidence shown by him in us boosted our moral and motivated us to perform better in preferring this project.

We are thankful to those who have contributed either directly or indirectly to this project.

Thanking You.

Mr. Randive Shubham Balavant

Mr. Patil Manoj Mahipati

INDEX

INTRODUCTON TO PROJECT

INTRODUCTOIN

The **University Admission System** is a web-based platform designed to streamline and automate the university admission process. Traditional admission procedures often involve manual paperwork, long queues, and inefficient handling of student applications. This system aims to eliminate these challenges by providing a digital platform where students can apply for university programs, track their application status, and receive admission updates efficiently.

The system is built using HTML, CSS, and JavaScript for the frontend, ensuring a user-friendly interface, while Node.js with Express.js powers the backend, managing API requests and data processing. PostgreSQL serves as the database, storing student records, application details, and admission decisions securely.

This project enhances accessibility, reduces administrative workload, and ensures a smooth and transparent admission process for both students and university staff. By leveraging modern technology, the system improves efficiency, accuracy, and convenience in the university admission process.

Working of System

The **University Admission System** automates and simplifies the admission process by allowing students to apply online, track their applications, and receive admission decisions digitally. The system functions as follows:

1. **Student Registration:**

   New student register by providing there basic details

2. **Application Submission**

   Upload required documents

3. **Tracking and communication**

   Administrators can communicate with applicants if further information is required.

4. **Report generation**

Teacher can also able to generate the report of the student

Need for the System

The traditional university admission process is often time-consuming, inefficient, and prone to errors due to manual handling. Some of the key challenges include:

- **Manual paperwork** leading to delays and loss of documents.
- **Long queues and processing time** causing inconvenience to students.
- **Human errors** in data entry and application processing.
- **Lack of transparency** in tracking application status.
- **Difficulty in managing large volumes of applications** efficiently.

To address these challenges, the **University Admission System** provides a **digital solution** that automates the admission process, improving efficiency, accuracy, and accessibility.

Scope of the System

The **University Admission System** is designed to be a **comprehensive and scalable** solution, covering multiple aspects of the admission process. Its scope includes:

1. **Student Registration and Application Submission**
   - Secure user authentication and profile management.
   - Online application form submission with document uploads.
2. **Application Processing and Review**
   - Automatic validation of student details and eligibility criteria.
   - Centralized database for easy access and management by university staff.
3. **Admission Status Tracking and Notifications**
   - Real-time tracking of application status.
   - Email/SMS notifications for updates and important deadlines.
4. **Administrative Management**
   - Dashboard for university staff to review and process applications.
   - Automated sorting of applications based on eligibility and requirements.
5. **Future Enhancements and Scalability**
   - Integration with online payment gateways for fee submission.
   - AI-based recommendation system for course selection.
   - Expansion to support multiple universities and courses.

Proposed System

Objectives

The **University Admission System** is designed to automate and enhance the traditional university admission process by providing a seamless, digital experience for both students and university administrators. The key objectives of the system are:

1. **To Automate the Admission Process**
    - Enable students to register, fill out applications, and submit required documents online.
    - Reduce manual intervention, paperwork, and processing time.
2. **To Improve Accessibility and Convenience**
    - Allow students to apply from anywhere, at any time.
    - Provide real-time updates on application status via a user-friendly interface.
3. **To Enhance Accuracy and Efficiency**
    - Minimize human errors in application processing.
    - Ensure proper verification of documents and eligibility criteria.
4. **To Provide Secure and Centralized Data Management**
    - Maintain all student records, applications, and admission decisions in a structured database.
    - Ensure data security through authentication and access control mechanisms.
5. **To Offer Transparency in the Admission Process**
    - Provide students with real-time tracking of their application status.
    - Allow university administrators to review applications efficiently through an organized dashboard.
6. **To Enable Future Scalability and Enhancements**
    - Implement AI-based course recommendations and automated document verification.
    - Support integration with payment gateways for fee submission.

Software Requirement Specification (SRS)

The system is built using modern web technologies to ensure efficiency, scalability, and security.

1. Functional Requirements

- **User Registration & Login:** Secure authentication for students and administrators.
- **Application Submission:** Online forms for students to fill in details and upload documents.
- **Application Tracking:** Real-time updates on application status.
- **Admin Dashboard:** Tools for university staff to manage applications.

2. Non-Functional Requirements

- **Scalability:** Ability to handle multiple users and applications simultaneously.
- **Security:** Secure data storage and role-based access control.
- **Usability:** A simple and user-friendly interface for students and administrators.
- **Performance:** Fast processing and retrieval of application data.

3. Technology Stack

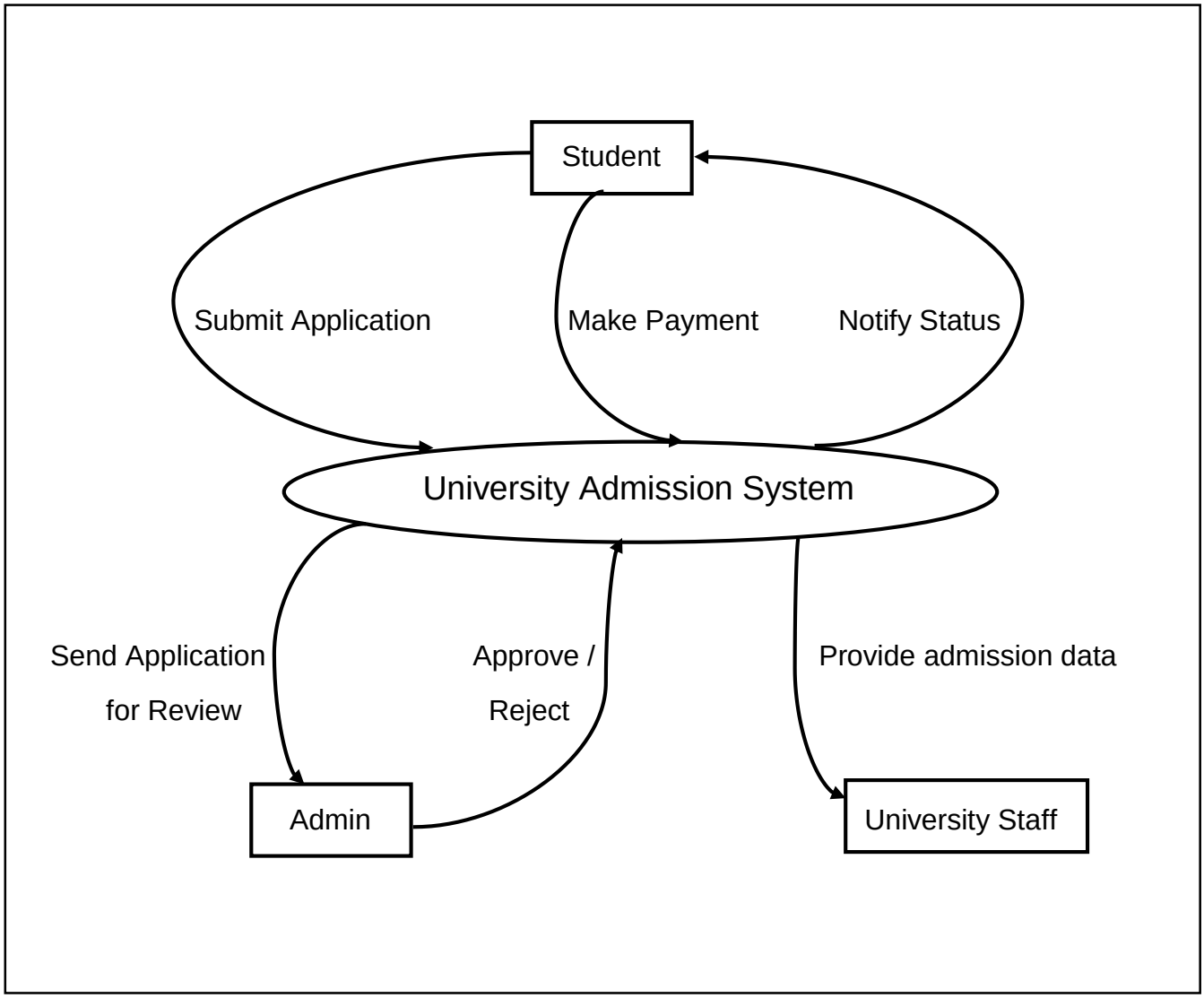
Component	Technology Used
Frontend	HTML5, CSS3, JavaScript
Backend	Node.js with Express.js
Database	PostgreSQL
Version Control	Git & GitHub

System Diagrams

1. **DFD (Data Flow Diagram)**

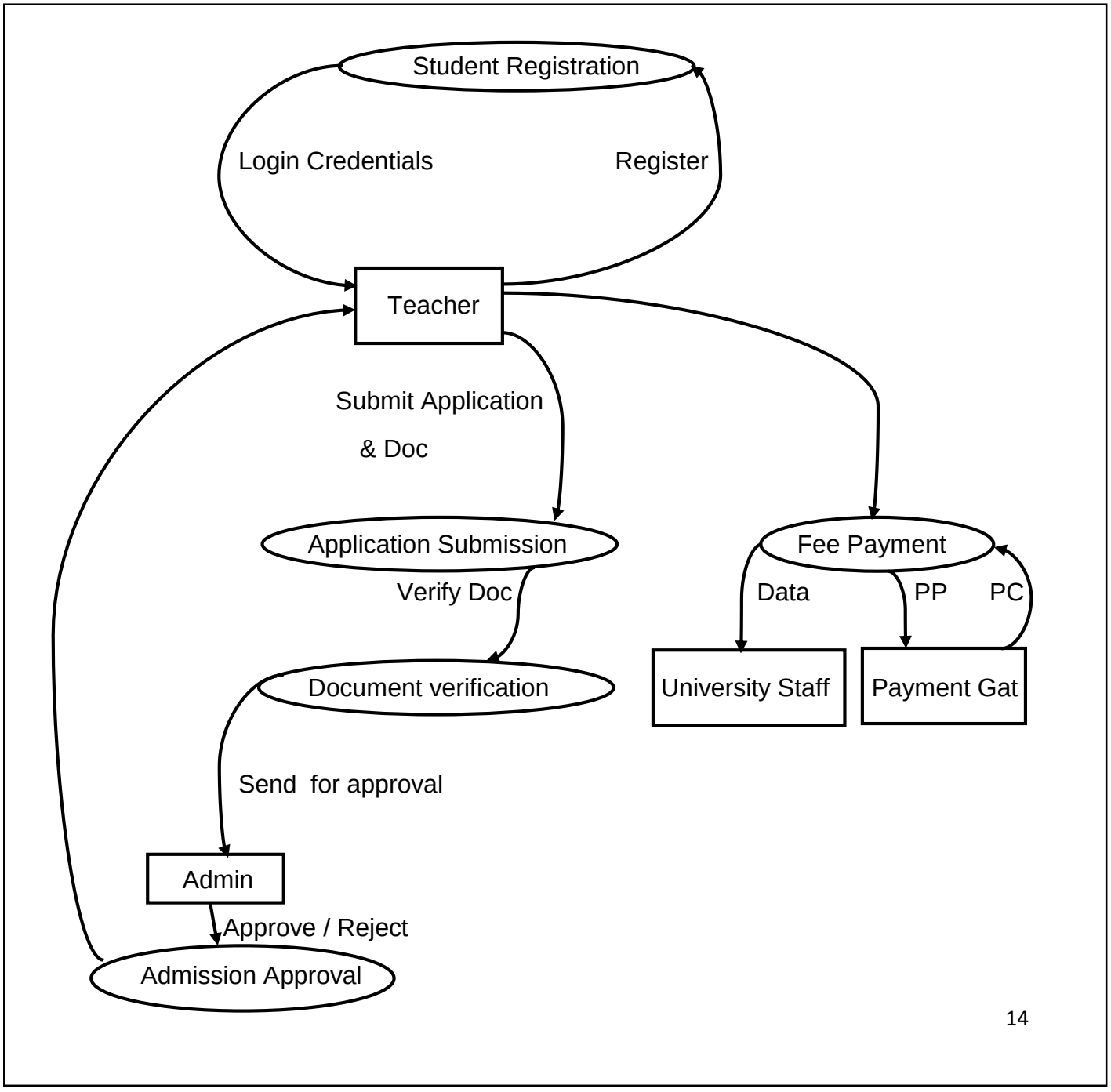
- **DFD Level 0 (Context Diagram) :**
  - Shows the system as a whole with its main entities (users) and data flow.
  - Entities : Teacher Staff, Admin
  - Processes : Student Registration , Application Submission, Admission Approval

Diagram :



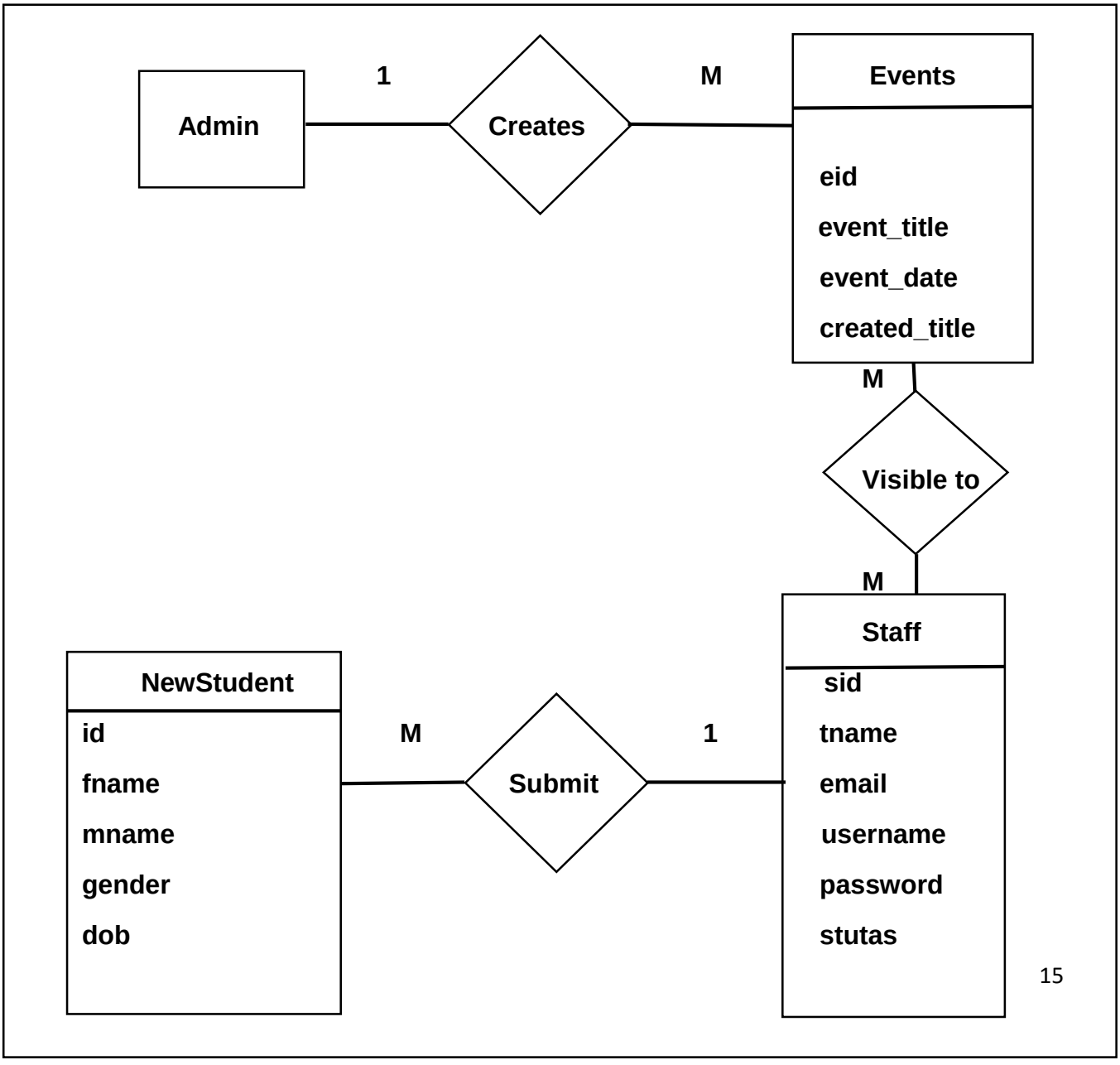
- **DFD Level 1 :**
  - **Breaks down the main process into sub-processes**
  - Student Registration: Teacher enters student details.
  - Application Processing: System verifies documents and eligibility.
  - Application Confirmation: Admin also confirm the application form.

Diagram :



2. **ERD (Entity-Relationship Diagram) :**
   An ERD shows how data is structured in database. Key entities and relationships:
   - Entities and Attributes;
     a. Staff (sid, tname, email, username, password, stutas)
     b. Newstudent (id, fname, mname, lname, gender, dob, twelvem, tenm, address, state, mbno, city, fillfees, addharno, tid, pcode, tname, date)
     c. Events (eid, event_title, event_date, event_time)
   - Relationship:
     a. A teacher can submit multiple application (1 : M.
     b. A Student makes one payment per admission.

Diagram:



System Requirements :

Below is the structured list that you can include in my report.

1. **Hardware Requirements :**

- Minimum Hardware Requirements (For development and Deployment)

Components	Specification (Minimum)	Specification (Recommended)
Processer	Intel Core i3 (10th Gen) AMD Ryzen 3	Inter Core i5/i7 (12th Gen) / AMD Ryzen 5/7
RAM	8 GB	16 GB or More
Storage	256 GB SSD/HDD	512 GB SSD or more
Graphics	Integrated Graphics	Dedicated GPU (Optional)
Network	Broadband Connection	Fiber - optic internet

- Server Requirements (For hosting)

Components	Specification (Minimum)	Specification (Recommended)
Processer	Ouad-core 2.4 GHz	Octa-core 3.0 GHz+
RAM	8 GB	16 GB or More
Storage	100 GB SSD	500 GB SSD or more
OS	Ubuntu 20.04 / Windows Server	Ubuntu 22.04 / Windows Server 2022
Database Server	MySQL 8.0 / PostgreSQL	MySQL 8.0+ / PostgreSQL 14+

2. Software Requirements :

- Development Softwarwe

Software	Purpose
Operating System	Windows 10/11 , MacOS, Linux
Code Editor	VS Code
Frontend Framework	HTML, CSS, JS
Backend Framework	Node.js with Express.js
Database	PostgreSQL
Version Control	Git and GitHub
API Testing Tool	Bruno

- Server-side Software

Software	Purpose
Web Server	Localhost
Database Server	PostgreSQL
Security	JWT

System Design

- Database Design

Sr. No.	Table Name	Description
1	Staff	This table contain the information about the all teachers
2	Newstudent	This table contain the information about the new admission
3	events	This table contain the information about the events

1. Staff Table : This table contain the details about the all teachers who makes the admission in the collage.

Sr. No.	Column Name	Datatype	Size	Key
1	sid	Integer	No size	Primary Key
2	tname	varchar	50	Not null
3	email	varchar	50	Not null
4	username	varchar	50	Not null
5	password	varchar	50	Not null
6	stutas	varchar	50	Not null

2. Newstudent Table: This table contain the information about the student which is make the new admission into the collage

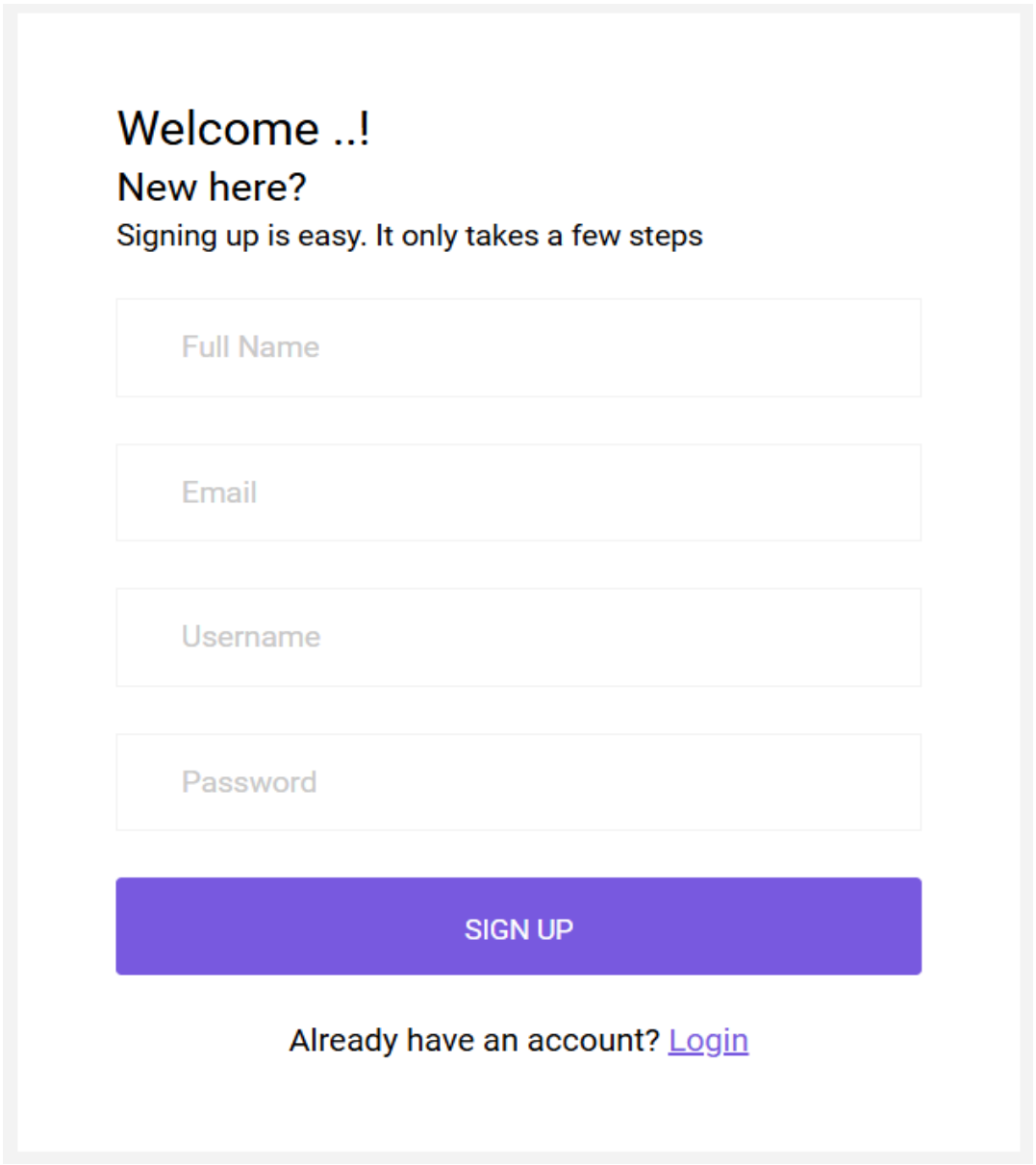
Sr. No.	Column Name	Datatype	Size	Key
1	Id	integer	No size	Primary key
2	fname	varchar	50	Not null
3	mname	varchar	50	Not null
4	lname	varchar	50	Not null
5	gender	varchar	20	Not null
6	dob	date	No size	Not null
7	twelve	varchar	10	Not null
8	tenm	varchar	20	Not null
9	address	varchar	50	Not null
10	state	varchar	15	Not null
11	mbno	varchar	20	Not null
12	city	varchar	20	Not null
13	fillfees	integer	No size	Not null
14	addharno	varchar	20	Not null
15	tid	integer	No size	Not null
16	pcode	varchar	10	Not null
17	tname	varchar	50	Not null
18	date	varchar	100	Not null

3. Events Table: This table contain the details about the events which is organised by the admin

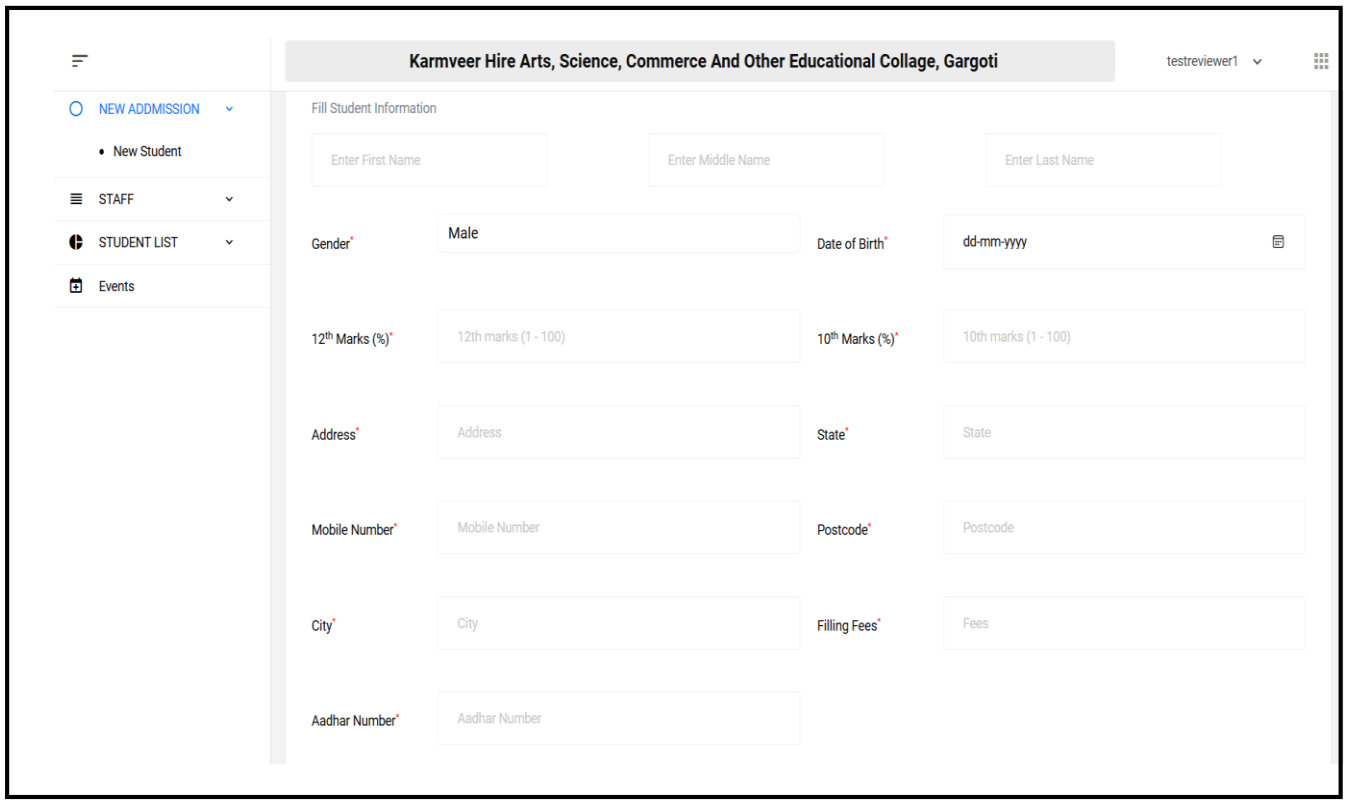
Sr. No.	Column Name	Datatype	Size	Key
1	eid	Integer	No size	Primary Key
2	event_title	varchar	1000	Not null
3	event_date	varchar	30	Not null
4	event_time	varchar	20	Not null

➢ Input Design

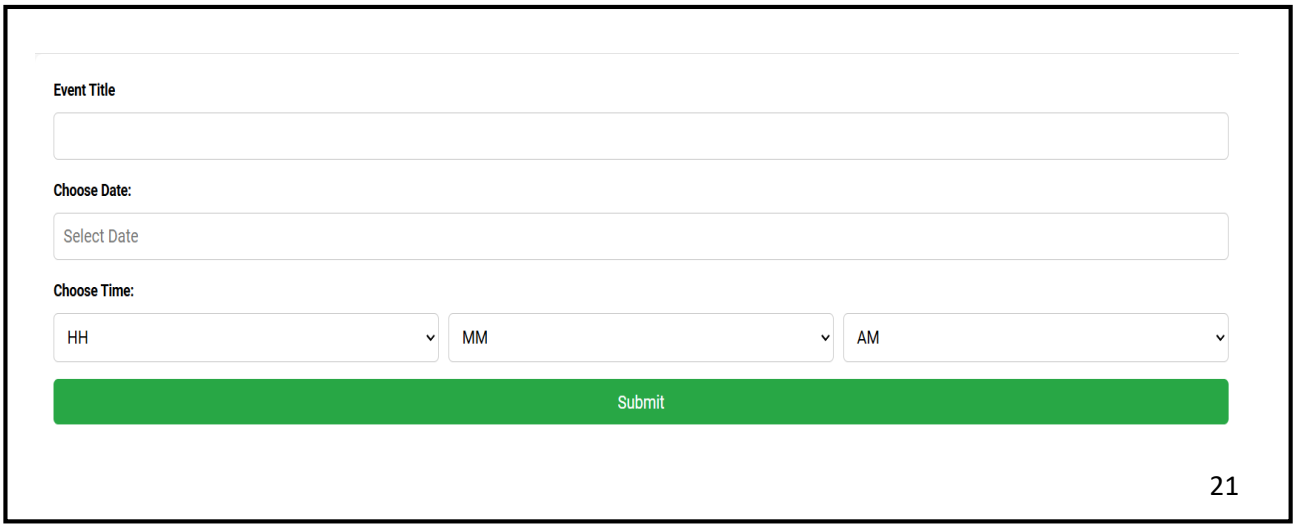
1. Teacher Registration Form
   - Fields : Full name, Email, Username, Password.
   - Validation : Required fields, Email format check



2. Student Registration Form
   - Fields : First name, Middle name, Last Name, Gender, DOB, 12th Marks, 10th Marks, Address, State, Mobile no, Post Code, City, Filling Fees, Aadhar no.
   - Validation : Required fields

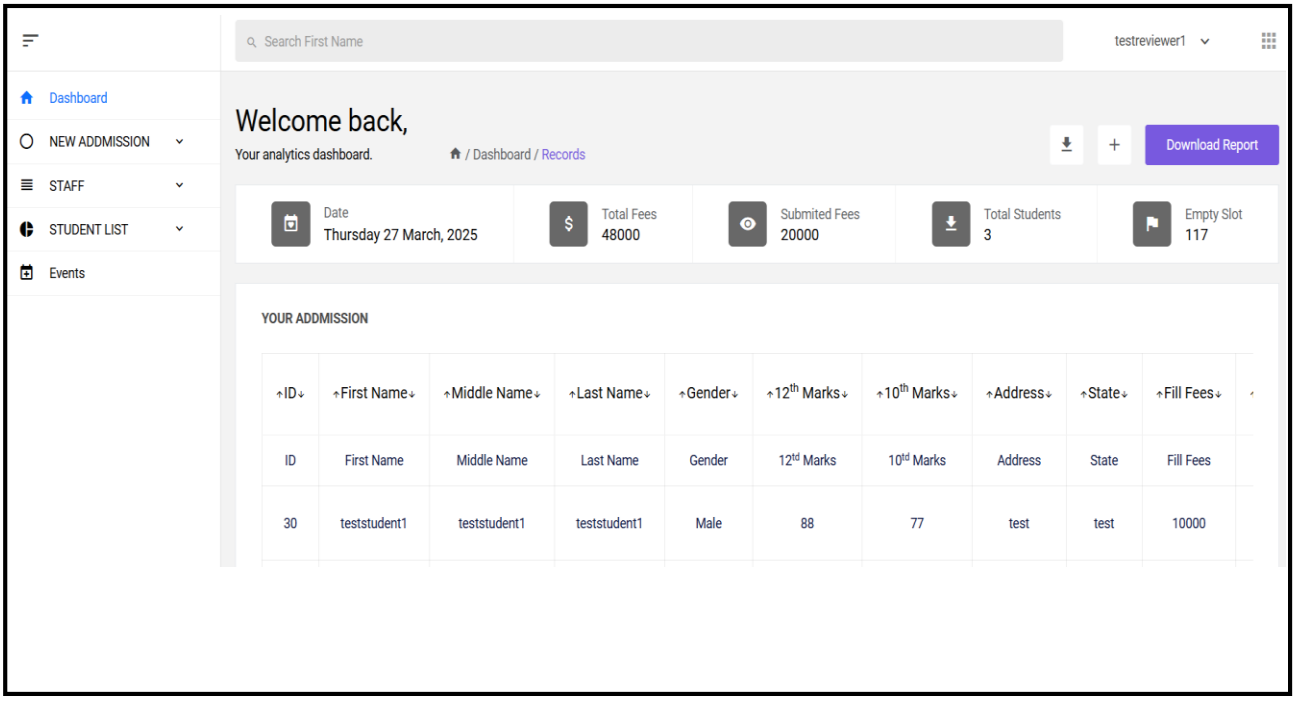


3. Event Registration Form
   - Fields : Event title, Date, Time
   - Validation : Required field

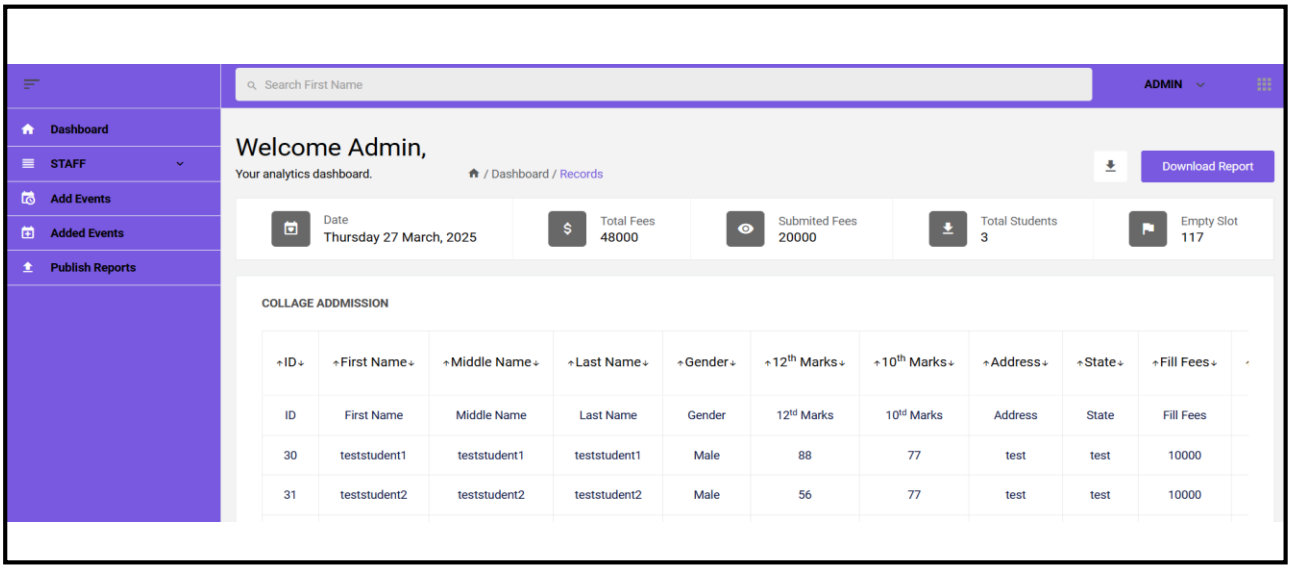


➢ Output Design

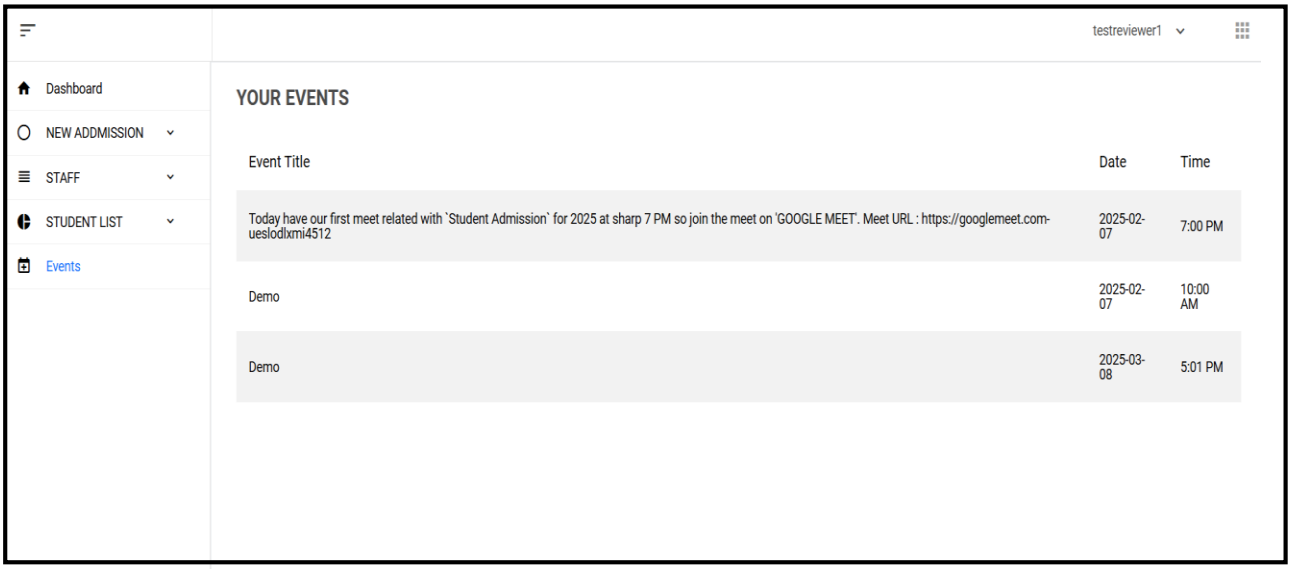
1. Teacher Dashboard :
   - Shows student list which make the admissions by logged in teachers.



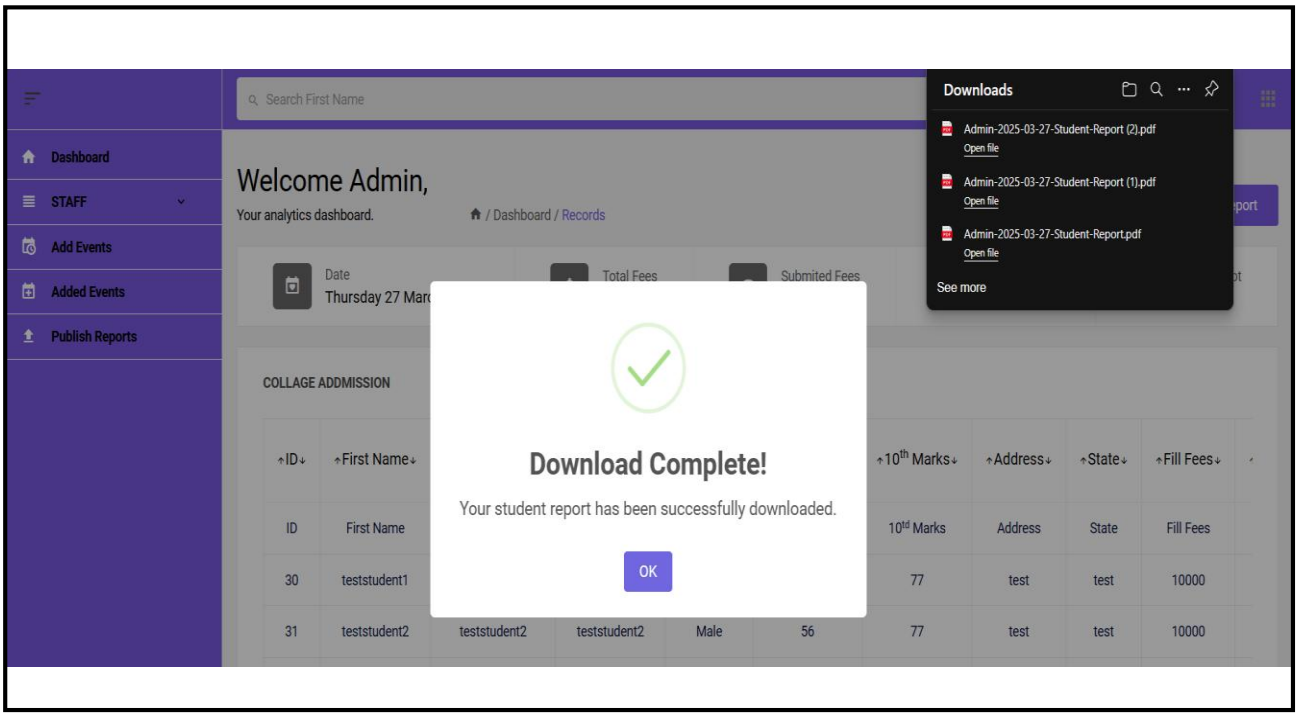
2. Admin Dashboard
   - Shows all student list which make the admission by all teachers.



3. Event Dashboard
   - Shows all events which are organized by the admin / principal



4. Admin Reports
   - Download the all students list as a student report.



User Manual

This **User Manual** provides step-by-step guidance for students, teachers, and admin on using **Collage Admission System**

Table Of Contents

Introduction

The **Collage Admission System** allows students to apply for admissions. Admin can manage admissions and organize collage events. Teacher can also make the admission of the students.

System Requirements

1. **Hardware Requirements**

- PC/Laptop : Minimum 8GB RAM, i3 Processor
- Mobile (Optional) : Android/iOS device
- Internet : Broadband or 4G+

2. **Software Requirements**

- Browser : Chrome, Edge, or Firefox (Latest version)
- Operating System : Windows, macOS, or Linux

User Roles & Access

Role	Permissions
Teacher	Make the student admissions.
Admin	Manage admission, organize events, and approve/reject applications.

Getting Started

1. **Teacher Registration & Login**

- Click on "**Create**" button
- Enter **'Full name, Email, Username, Password'**
- Click on **'Sign Up'** button
- Login using credentials.
- Make the student admission
- Click on **"New admission"** tab
- Fill all details like **'First Name, Middle name, Last name, gender, dob, 10th marks, 12th marks, address'** or so on
- Click on **"Submit"** button

2. **Admin Login**

- Login with credentials
- Manage admissions
- Organize events
- Download reports

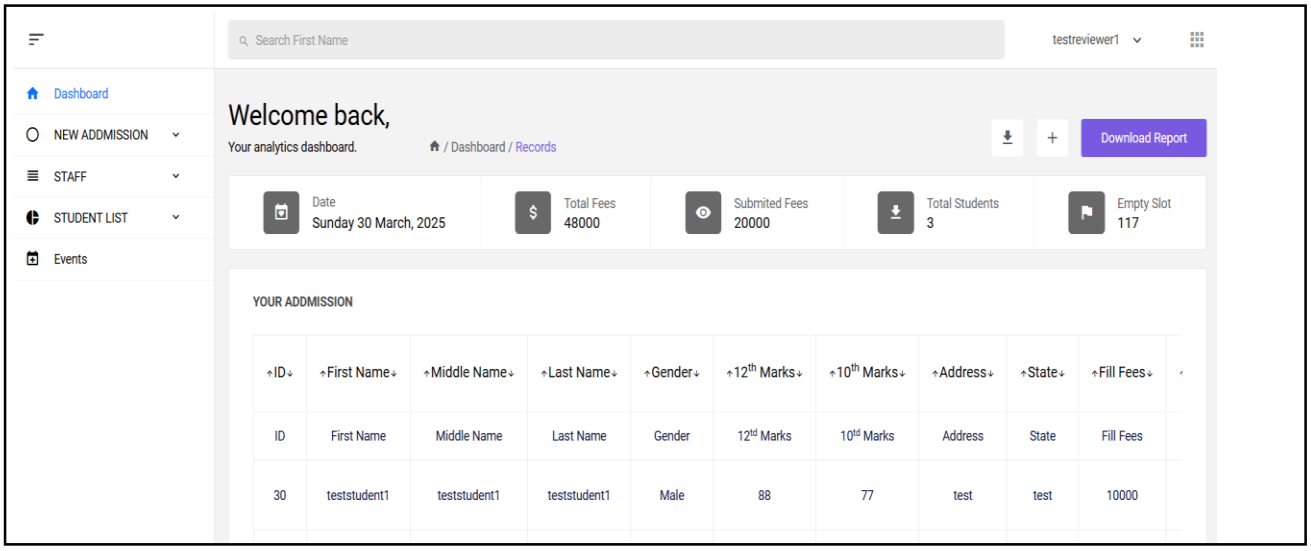
Logout & Security

Always click on **"Logout"** after use. Do not share your password with anyone.

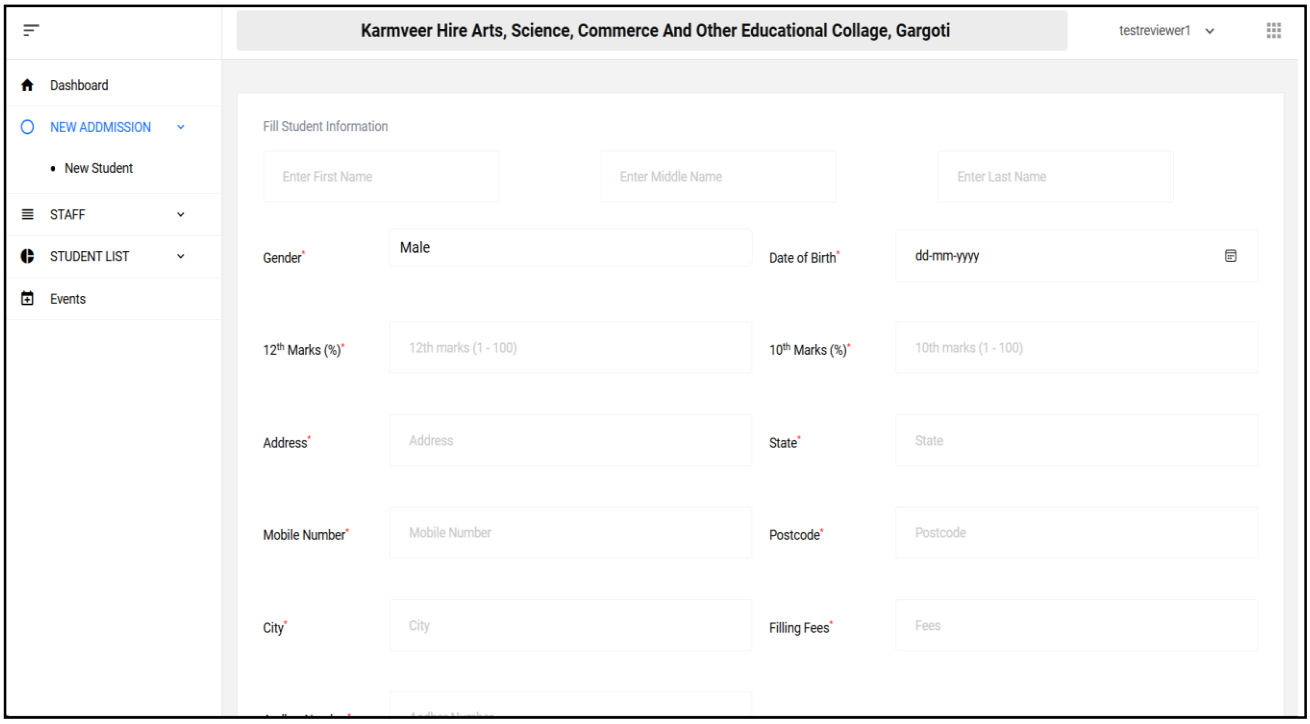
Input and Output Screen

Teacher Page

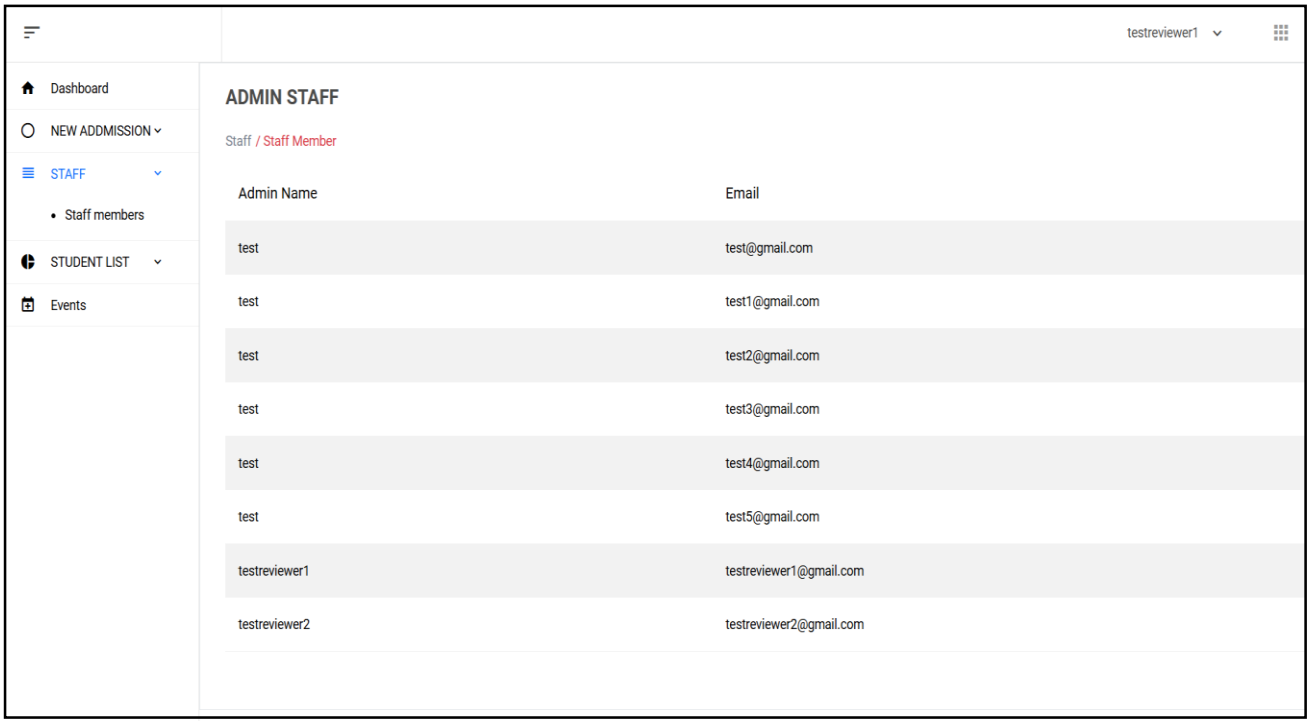
1. Dashboard



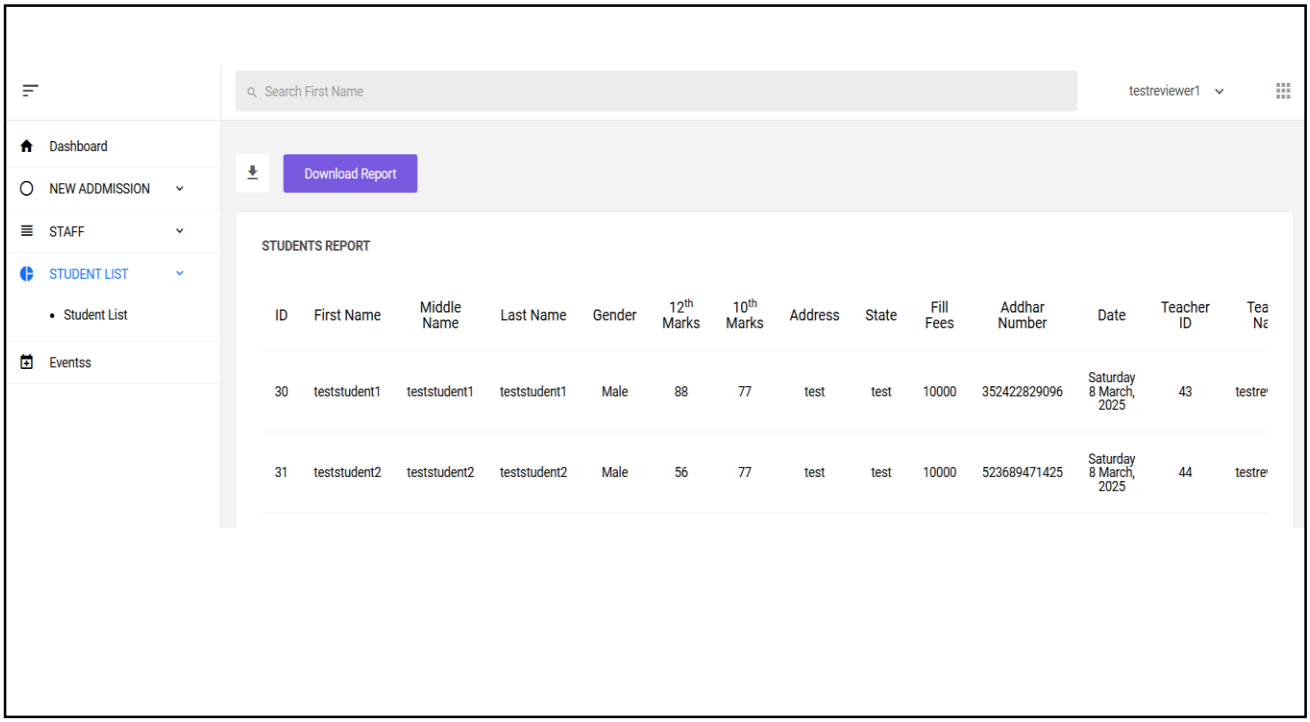
2. New admission Page



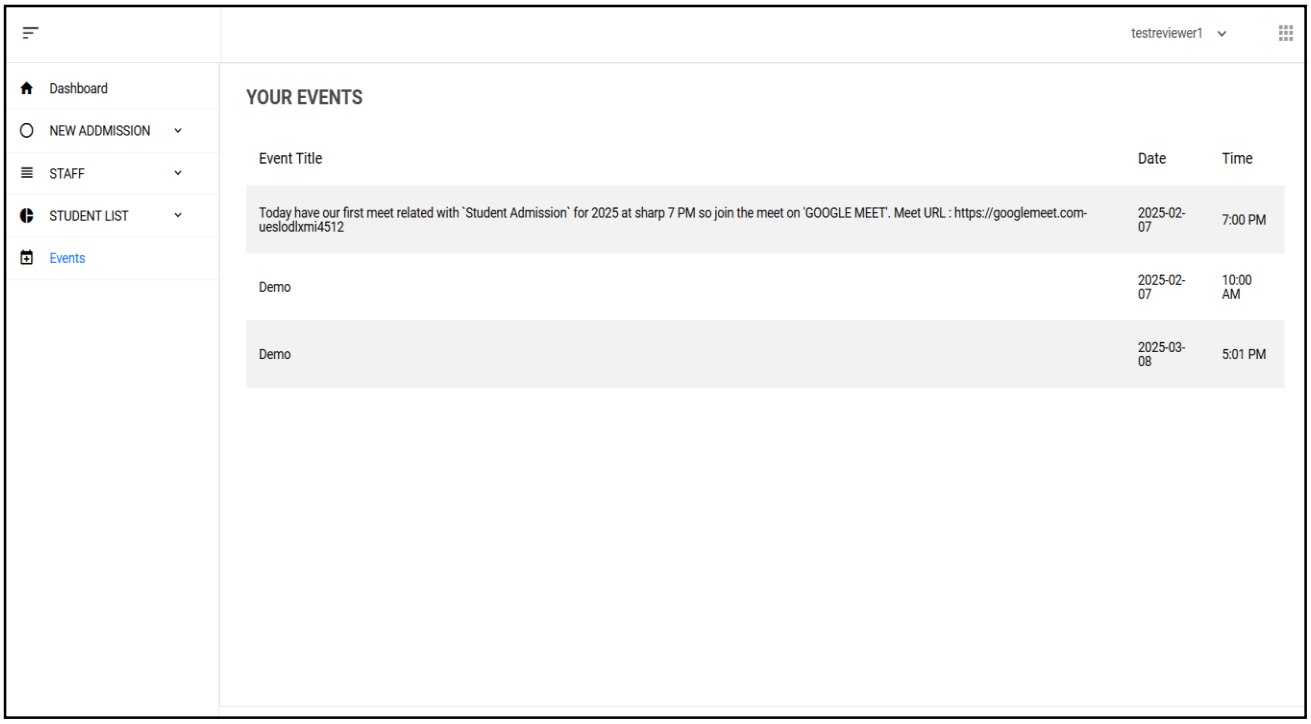
3. Staff Page



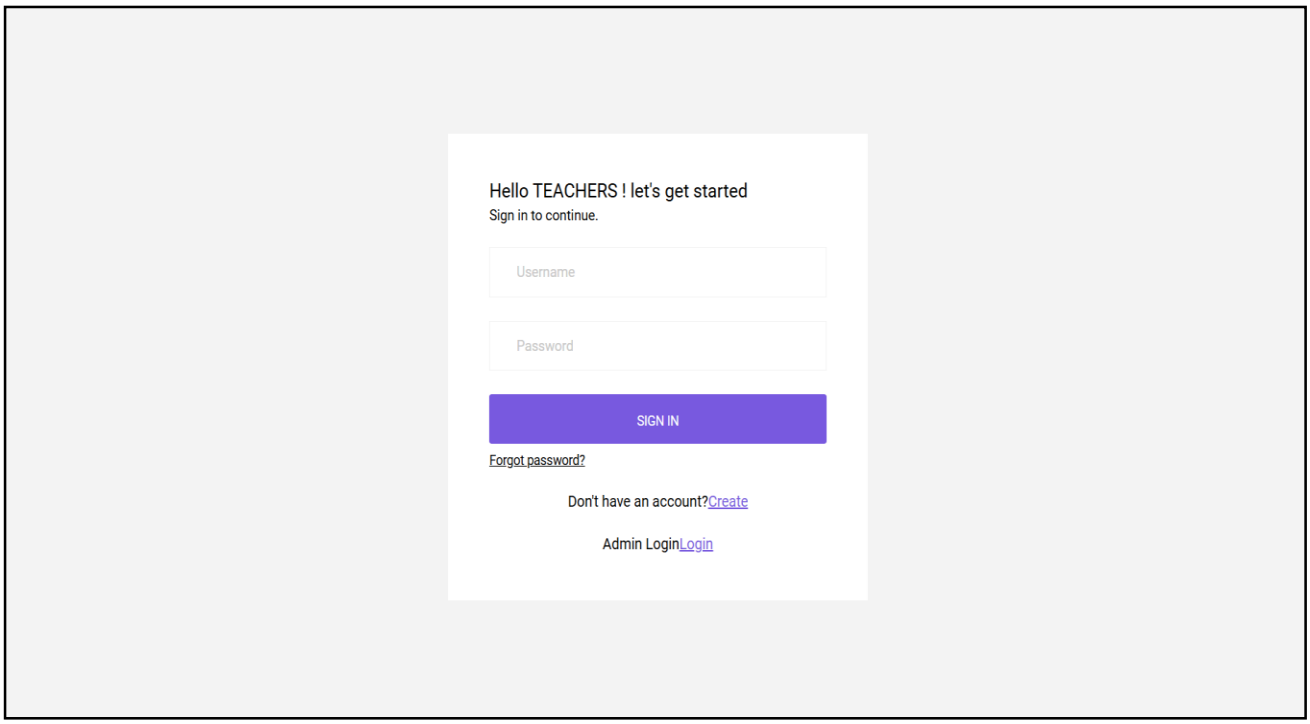
4. Student List Page



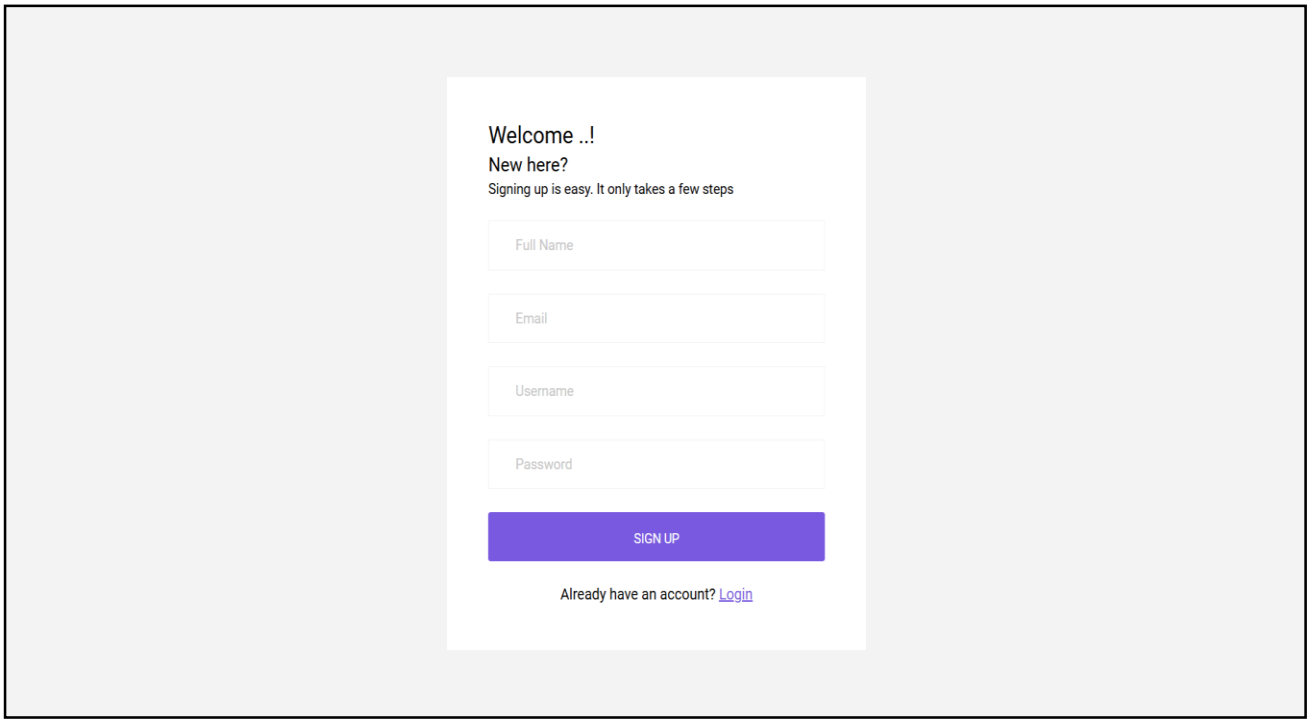
5. Events Page



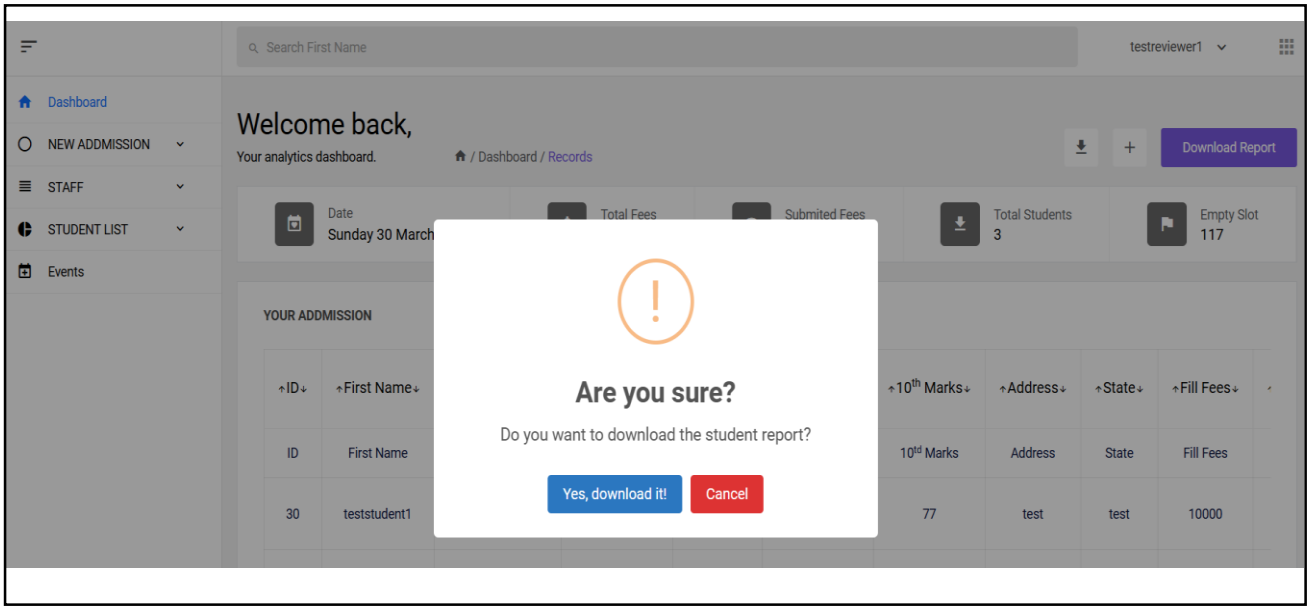
6. Teacher Login



7. Teacher Registration Page



8. Teacher Download Option



Conclusion

The University Admission Management System provides a streamlined solution for handling the complex process of student admissions. By automating tasks such as application submissions, data verification the system significantly reduces administrative workload and minimizes the chances of human error. The user-friendly interface ensures that administrators can easily manage applicant information while students experience a smooth and transparent admission process.

Overall, this project achieves its goal of enhancing the efficiency, accuracy, and accessibility of university admissions. It lays a strong foundation for future upgrades and real-world deployment in educational institutions.

Future Enhancements

While the current system is functional and effective, several future enhancements could further improve its performance, usability, and scalability:

1. Online Payment Integration

Integrate secure payment gateways (like Razorpay, PayPal, or Stripe) to allow students to pay application or admission fees directly through the portal.

This would automate fee tracking and reduce the dependency on manual payment confirmations.

2. Document Upload and Verification

Add a feature where applicants can upload required documents (e.g., certificates, ID proofs) during the application process.

An automatic document verification system using OCR (Optical Character Recognition) technology could speed up validation.

3. Mobile App Version

Develop a mobile application version of the system to allow students and staff to access the platform easily through smartphones.

Push notifications can enhance real-time engagement and updates.

4. Advanced Reporting and Analytics

Include dashboards that display real-time analytics, such as the number of applications received, department-wise enrollment, seat availability, etc.

This would help university management make data-driven decisions quickly.

5. Role-Based Access Control

Introduce multi-level login systems with different privileges (Admin, Reviewer, Applicant).

Ensures better security and organized workflow management within the system.

Bibliography

1. Web Development Resources
    - W3Schools. HTML, CSS, JavaScript, PHP Tutorials. Retrieved from https://www.w3schools.com
    - MDN Web Docs. Web Development Documentation. Retrieved from https://developer.mozilla.org
2. Database Management Systems
    - PostgreSQL Documentation. PostgreSQL Reference Manual. Retrieved from https://www.postgresql.org/docs/
3. Software Development Practices
    - Sommerville, Ian. Software Engineering (10th Edition). Pearson Education, 2015.
4. Admission System References
    - Online articles and samples from educational project repositories like GitHub and SourceForge for understanding the structure of academic management systems.
5. Design Inspiration and UI/UX Guidelines
    - Nielsen Norman Group. UX Research and Guidelines. Retrieved from https://www.nngroup.com
6. Additional Online Tutorials and Forums
    - Stack Overflow. Developer Discussions and Solutions. Retrieved from https://stackoverflow.com
    - GeeksforGeeks. Programming and Computer Science Resources. Retrieved from https://www.geeksforgeeks.org