NEXT-GEN CLG EVENT MANAGER

SYNOPSIS

The **NEXT-GEN CLG EVENT MANAGER** is an advanced web-based application developed using **HTML, CSS, JavaScript, PHP, and Microsoft SQL Server**, with a responsive design ensuring compatibility across all devices. This system is designed to streamline the **management of college events**, eliminating the inefficiencies of manual processes and ensuring secure, automated event tracking and monitoring.

The application provides a user-friendly interface that allows event organizers to manage and update event details without requiring coding knowledge. It supports **real-time event tracking**, automated reminders, and an efficient data management system that ensures secure storage and retrieval of participant records.

Built with **robust security features**, the system ensures data integrity and privacy while providing an efficient and structured approach to **inter-college event organization**. By reducing paperwork, improving event coordination, and ensuring secure access control, **NEXT-GEN CLG EVENT MANAGER** enhances productivity, making college event management more organized, efficient, and accessible for all users.

.

1. INTRODUCTION

1.1 ORGANIZATION PROFILE:

The **NEXT-GEN CLG EVENT MANAGER** is a modern web-based application developed using **HTML, CSS, JavaScript, PHP, and Microsoft SQL Server**, designed to provide a **secure, automated, and efficient event tracking system** for colleges and institutions. In an era where digital transformation is crucial for streamlining operations, this project aims to **enhance event management by eliminating manual inefficiencies and ensuring real-time monitoring** of events.

The primary objective of this project is to establish an **intuitive and structured platform** where **event organizers** can create and manage events, **administrators** can oversee and approve them, and **participants** can register, track schedules, and receive real-time notifications. By automating event tracking and monitoring, the system reduces paperwork, enhances security, and provides **seamless access to event-related data** for all users, Additionally, the **NEXT-GEN CLG EVENT MANAGER** supports **real-time event tracking**, **secure data storage**, allowing event records to be **easily updated and retrieved**.

With a **user-friendly interface, robust security measures, and efficient data management capabilities**, this system simplifies event organization and tracking. It ensures **structured approval workflows**, **minimizes manual errors**, and improves the overall efficiency of managing college events. By **digitizing event processes**, the project enhances **accessibility, security, and operational effectiveness**, making it an indispensable tool for modern educational institutions.

Project Workflow:

1. **Organizer Login** → Creates and manages event requests.
2. **Administrator Approval** → Grants or denies event requests.
3. **Attendee Registration** → Registers for an event.
4. **Check-in Process** → Attendee checks in at the event.
5. **Event Monitoring** → Administrator tracks attendance and event progress.
6. **Check-out Process** → Attendee checks out after the event.
7. **Administrator Oversight** → Monitors event participation and generates reports.

1.2 SYSTEM SPECIFICATION:

A system requirements specification(SRS)also known as a software requirements specification is a document or set of documentation that describes the features and behaviour of a system or software application.

1.2.1 HARDWARE CONFIGURATION:

Processor : Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.70 GHz

Ram : 16 GB

Hard Disk : 500 GB

Monitor : LED 14” Color Monitor

Mouse : Optical Mouse

Keyboard : 104 Keys

1.2.2 SOFTWARE SPECIFICATION:

Operating System : Windows 11 Pro

Front End : HTML, CSS, JS, Bootstrap, PHP

Back End : MySQL 8.0

1.2.2.2FEATURES FOR PROGRAMMING TOOLS:

PHP 8.4.2:

In the ever-evolving realm of web development, PHP (Hypertext Preprocessor) continues to be a powerful and efficient server-side scripting language, providing seamless integration with HTML to create dynamic and interactive web pages. PHP 8.4.2 introduces enhanced performance and modern features, making it an ideal choice for building robust web applications. Its compatibility with modern web standards ensures developers can deliver responsive, adaptive, and secure user experiences.

One of the key strengths of PHP lies in its ability to process real-time data and interact with databases efficiently. The scripting capabilities in PHP 8.4.2 empower developers to dynamically update and display web content by connecting directly to MySQL databases. This

feature ensures that the user interface remains fluid and up-to-date, offering a seamless experience.

Form Handling and Data Security:

PHP 8.4.2 offers advanced features for handling form submissions and validating user input. These capabilities ensure the security and integrity of user data, safeguarding sensitive information against threats. This version enhances secure encryption mechanisms, making it reliable for projects that involve data handling and privacy compliance.

Versatility and User-Centric Design:

PHP 8.4.2 adapts effortlessly to varying user demands, enabling the creation of user-friendly and intuitive interfaces. Its ability to respond dynamically to user actions enhances the engagement level, providing a versatile platform for web application development. This version continues to streamline the process of developing responsive and interactive designs for diverse devices.

Efficient Database Interactions:

PHP 8.4.2 integrates seamlessly with MySQL 8.0, providing robust tools to manage database interactions efficiently. Whether handling complex queries or ensuring smooth data transactions, this synergy ensures that developers can create dynamic and user-centric web applications with minimal effort.

MySQL 8.0

MySQL 8.0 is a powerful and widely used relational database management system (RDBMS), offering advanced features that align perfectly with modern web development needs. MySQL serves as the backbone for managing, storing, and retrieving data efficiently for any web application.

SQL Server Tables:

MySQL 8.0 organizes data into structured tables, allowing efficient grouping and management of related information. It enables developers to create, update, and extract information effortlessly, ensuring that the data flow remains streamlined and reliable.

Primary and Foreign Keys:

MySQL 8.0 supports advanced primary and foreign key constraints to maintain data consistency across multiple tables. Primary keys uniquely identify records in a table, while foreign keys establish relationships between tables, ensuring referential integrity.

Relational Database and Data Consistency:

As a relational database, MySQL 8.0 enables the linking of multiple tables, providing a cohesive way to manage complex datasets. It ensures that relationships between tables remain intact, making it a robust choice for applications requiring structured data management.

Advanced Features of MYSQL 8.0:

1. Data Abstraction **:** Offers physical, conceptual, and view-level abstraction to simplify data handling.
2. Enhanced Security **:** Implements advanced encryption and authentication mechanisms for secure database access.
3. Performance Optimization **:** Features tools like query caching and indexing to improve the speed of data retrieval.
4. Portability **:** Supports multiple operating systems, allowing seamless migration across platforms.

Advantages of MYSQL 8.0:

- Reduces data redundancy and inconsistency.
- Allows efficient data sharing and collaboration.
- Ensures high levels of data security and integrity.
- Provides tools for scaling database operations.
- Offers support for advanced transactions and fault tolerance.

Disadvantages of DBMS:

While MySQL 8.0 offers significant benefits, its reliance on centralization requires robust backup mechanisms to prevent data loss during failures. Additionally, its extensive features may demand more advanced hardware configurations, increasing operational costs.

Features of MYSQL 8.0 In Your Project:

For your project, MySQL 8.0 serves as the foundation for creating a highly scalable and secure database environment. Its compatibility with PHP 8.4.2 ensures smooth integration, enabling dynamic content management and seamless data processing. The advanced features like query optimization, encryption, and support for relational data structures make it a critical tool for developing efficient and secure web applications.

2. SYSTEM STUDY

2.1 EXISTING SYSTEM:

The current event tracking system used in many educational institutions primarily relies on manual processes or standalone tools like paper records and spreadsheets. While these methods may appear manageable for small-scale events, they lack scalability**,** efficiency, and accuracy when handling multiple events across different departments.

Managing event registrations, approvals, and participation manually often results in miscommunication, scheduling conflicts, and delays in event organization. Event details must be updated and shared across different teams individually, leading to **time-consuming and error-prone** processes. Additionally, the **absence of a centralized digital platform** makes it difficult for administrators, staff, and participants to **track event details, manage schedules, and retrieve historical data** effectively.

Security is another major concern in the existing system. **Sensitive data such as participant details, event records, and approvals** remain vulnerable to unauthorized access due to the lack of proper authentication and access control mechanisms.

The **NEXT-GEN CLG EVENT MANAGER** aims to address these challenges by providing a **secure, automated, and user-friendly web-based platform** that streamlines event tracking, enhances communication, and ensures efficient event management.

2.1.1 DRAWBACKS OF THE EXISTING SYSTEM:

1. **Manual Process:**

    - The current event tracking system relies on physical registers or paper-based approvals, increasing administrative workload and errors.

2. **Lack of Integration:**

    - There is no centralized system connecting students, staff, and administrators for seamless tracking of event approvals and participation.

3. **Inefficient Communication:**

    - Staff and students rely on verbal communication or written logs, leading to miscommunication and missing records.

4. **Manual Reporting:**

    - No digital records for event participation, making it difficult to track historical data.
    - Administrators have to manually check logs to verify student participation in past events, leading to delays and inaccuracies.

5. **Security Risks:**

    - Unauthorized participation is possible due to a lack of verification mechanisms.
    - Manual records are vulnerable to tampering or loss, leading to security concerns in event management.

6. **Inefficient Communication:**

    - Staff and students rely on verbal communication or written logs, leading to miscommunication and missing records.
    - No automated tracking of event schedules, participation, or results, causing confusion and inefficiencies.

7. **Manual Reporting:**

    - No digital records for event participation, making it difficult to track historical data.
    - Administrators have to manually check logs to verify student participation in past events, leading to delays and inaccuracies.

8. **Security Risks:**

    - Unauthorized participation is possible due to a lack of verification mechanisms.
    - Manual records are vulnerable to tampering or loss, leading to security concerns in event management.

2.2 PROPOSED SYSTEM:

The **NEXT-GEN CLG EVENT MANAGER** is a **centralized, secure, and automated** web-based solution designed to streamline **event tracking and management** within an institution. This system ensures **efficient coordination between event organizers, security personnel, and participants**, reducing manual efforts and enhancing overall event security and organization.

The proposed system introduces **role-based access**, allowing different users to log in with unique credentials and access tailored features:

- **Admin/Event Manager**:
  - Manages events and participant registrations.
  - Approves or rejects event requests in real time.
  - Monitors participant attendance and generates reports.
- **Security Personnel**:
  - Verifies approved event passes.
  - Records participant check-in and check-out times.
  - Ensures only authorized attendees access events.
- **Participant**:
  - Registers for events online.
  - Receives real-time **status notifications** on approvals.
  - Views recorded check-in and check-out times.

2.2.1 FEATURES OF THE PROPOSED SYSTEM:

- **Real-Time Participant Tracking** – Security personnel log check-ins and check-outs, allowing admins to monitor movement efficiently.
- **Seamless Digital Registration & Approval** – Participants submit event requests digitally, which are instantly approved or rejected by event managers. Approved passes are automatically available for security verification.
- **Comprehensive Admin & Security Controls** – Admins oversee event approvals, manage security personnel, and track participant data in real time.
- **User-Friendly Interface** – A responsive and intuitive dashboard allows users to navigate seamlessly.

- **Status Updates & Notifications** – Participants receive instant updates on their registration status, ensuring clear communication.
- **Robust Security Measures** – Role-based authentication ensures that only authorized users can access specific features, safeguarding event data.

With these features, the **NEXT-GEN CLG EVENT MANAGER** eliminates inefficiencies in event coordination, enhances security, and improves the overall experience for organizers, security personnel, and attendees.

3. SYSTEM DESIGN AND DEVELOPMENT

3.1 FILE DESIGN:

File design plays a crucial role in structuring data storage, access, and management in the system. In the Event Tracking System, the file design ensures efficient record-keeping of event details, student participation, approvals, and event results while maintaining security and ease of access for different users (students, staff, and administrators).

User-Friendly Interfaces:

- **Students:** Can easily register for events and track event schedules and results.
- **Staff:** Access a dashboard to approve student participation, add results, and manage event schedules.
- **Administrator:** Manages staff, tracks event performance, and generates reports.
- **Icons and Results:** Clear navigation icons and detailed reports enhance usability.

3.2 INPUT DESIGN:

Input design determines how administrators, staff, and students interact with the system to input and manage data. The system ensures accurate and secure data entry while maintaining a user-friendly interface.

Input Devices:

- **Students & Staff:** Use keyboards/mouse/touchscreens to register for events and manage approvals**.**
- **Administrators:** Enter event schedules, approvals, and results using a simple mobile/desktop form**.**

Objectives of Input Design:

- Ensure accuracy and completeness in event registrations and approvals.
- Validate data using input integrity controls to prevent errors.
- Provide an intuitive form layout for quick and efficient data entry.
- Tracks all event approvals, student registrations, and result updates.

- Records timestamps of every activity for security and transparency.

3.3 OUTPUT DESIGN:

Output design focuses on presenting processed data clearly to students, staff, and administrators for decision-making and tracking.

Types of Output:

- On-screen dashboards (for real-time event tracking).
- Printable reports (admin and staff access to event participation history).
- Students are notified about event approvals, schedule changes, and results. To check the results, they need to log in.

Objectives of Output Design:

- Pro6vide real-time status updates to students, staff, and administrators.
- Ensure clarity in reports and event status displays.

External and Internal Outputs:

- **Internal:** Administrator dashboards, staff logs, and approval records.
- **External:** Student event registration status and notifications.

3.4 DATABASE DESIGN:

Database design is crucial for securely storing event-related records, including student registrations, staff approvals, event details, and results. The system utilizes a relational database to manage event participation and records efficiently.

Relational Model:

The database includes tables and relationships between:

- **Students:**(User_ID, Name, Department, Contact, Event, College, Registration_Status).
- **Staff:**(Staff_ID, Name, Gender, Event).

Logical Design:

- The system organizes event data into structured tables for efficient access and management.
- SQL queries are used for data retrieval, validation, and reporting.

Goal of DBMS:

- Centralized and secure storage for all event-related data.
- Optimized retrieval and search features for administrators and staff.

3.5 SYSTEM DEVELOPMENT:

The Event Tracking System is a web-based platform that automates event approval and tracking. It provides seamless interaction between students, staff, and administrators while ensuring real-time updates and security.

Key Development Features:

- **Automated Event Registration** – Students can register for events digitally without manual paperwork.
- **Approval Dashboard** – Staff members have an approval dashboard with real-time notifications for student participation requests.
- **Event Scheduling & Management** – Administrators and staff can efficiently manage event schedules, participation records, and results.
- **Secure Data Management** – Ensures all event records, student participation logs, and results are stored securely for auditing and future reference

3.5.1 DESCRIPTION OF MODULES:

Administrative Module:

- Manage staff and event-related data.
- View student details and event reports.
- Oversee event results and approvals.

Staff Module:

- Approve student participation in events.
- Manage event schedules and timetables.
- Track event progress and update results

Student Module:

- Register for events and track participation.
- View event schedules and results.
- Access overall event performance records.

Security & Authorization Module:

- Role-based access for different users.
- Secure student registration and approvals.
- Maintain audit logs for tracking activities.

Use Case Diagram for Event Tracking System:

Actors:

- **Student (User):** Registers for events and views schedules/results.
- **Staff (Organizer):** Manages student participation, event schedules, and results.
- **Administrator:** Oversees event approvals, tracks participation, and manages reports.

Use Case Flow:

1. Student logs in and registers for an event.
2. Staff reviews and approves or denies participation.
3. If approved, the student is added to the event list.
4. The system updates event schedules and sends notifications.
5. Staff tracks attendance and records results.
6. Administrators access participation history and generate reports.
7. Students view final event results and performance records.

4. TESTING AND IMPLEMENTATION

Testing ensures the system operates as intended and meets the requirements. It is the process of executing the program to identify and correct errors systematically. A good testing process ensures high accuracy, reliability, and performance in minimal time. Below is the structured approach for testing:

4.1 TESTING APPROACH:

Two key components are tested:

1. Software Configuration Testing**:**
    - Includes testing against the requirements specification, design specification, and source code.
2. Test Plan Execution**:**
    - Utilizes a test plan, testing tools, and test cases with expected results.

4.2 TESTING TECHNIQUES:

Testing ensures that the Event Tracking System functions correctly and meets all specified user requirements. It plays a crucial role in improving the quality, reliability, and security of the system. The main goal is to identify and fix errors, ensuring the system operates as intended.

Types of Testing Conducted:

- **Unit Testing:** Testing individual modules.
- **Integration Testing:** Verifying multiple modules work together.
- **Validation Testing:** Ensuring business logic functions correctly.
- **Acceptance Testing:** Final approval by users before deployment.

Unit Testing (Program Testing):

Objective: Test individual modules of the system independently to verify their correctness.

Tested Modules:

- **Event Registration:** Ensured students could successfully register for an event with valid inputs.
- **Administrator Approval System:** Verified that administrators could approve or reject events without issues.
- **Event Schedule Management:** Ensured staff could assign event schedules and update event details.
- **Notification System:** Tested whether students received real-time updates on event status and results.

Integration Testing:

Objective: Verify the smooth interaction between different system modules.

Tested Scenarios:

- A student registers for an event → The registration appears in the administrator's approval dashboard.
- The administrator approves the event → It is reflected in the student and staff panels.
- Staff schedule the event → The schedule updates in the system for all users.
- The event is conducted, and results are updated → The results are displayed for students.

Validation Testing:

Objective: Ensure the system meets business logic and user requirements.

Key Validations Performed:

- Students cannot register for an event without filling mandatory details (e.g., event name, category, date, time).
- Staff cannot approve a student's event registration if required details are missing.
- Event results cannot be entered unless all required participant details are recorded.
- Error handling messages appear when incorrect data is entered.

Acceptance Testing:

Objective: Verify if the system meets user expectations and performs well in a real environment.

Approach:

- Administrators, staff, and students tested the system using live event data.
- The system was tested under real-world scenarios to ensure reliability.
- Performance was measured under high user load conditions to ensure smooth operation.

4.3 IMPLEMENTATION:

Database Implementation:

Designed the database schema based on the system's functional requirements and created tables to store:

- **Student event registrations:** (ID, student details, event ID, status).
- **Staff approvals:** (Staff ID, event ID, approval status).
- **Event logs:** (Event ID, student participation, timestamps).

Implemented **MySQL database connectivity** in PHP using **MySQL/PDO** to ensure secure interactions.

Testing and Quality Assurance:

- **Unit Testing**: Verified each module (student event registrations, staff approvals, event logs, and notifications) independently.
- **Integration Testing**: Ensured smooth communication between backend and frontend.
- **User Acceptance Testing (UAT)**: Conducted real-world testing with students, staff, and administrators to validate usability and functionality.

TEST CASE:

CASE NO	TEST CASE DESCRIPTION	EXPECTED RESULT	STATUS
1	Admin can access portal through valid Id and Password.	Portal successfully accessed by valid Id and password	Pass
2	Admin can access portal through invalid Id and Password.	System prompts error message and prevents accessing the portal	Pass
3	Faculty can access portal through valid Id and Password.	Portal successfully accessed by valid Id and password	Pass
4	Faculty can access portal through invalid Id and Password.	System prompts error message and prevents accessing the portal	Pass
5	Admin adds an Faculty with valid data.	Faculty is successfully added	Pass
6	Register student with valid details.	Student is successfully registered.	Pass
7	Register student with missing required fields.	System prompts error message and prevents Student registration.	Pass
8	Faculty schedules an event with valid dates.	Event is successfully Scheduled.	Pass
9	Faculty schedules an event with past dates.	System prompts error message and prevents scheduling.	Pass
10	Faculty adds result an event with valid data.	Result is successfully added.	Pass
11	Faculty adds result an event with invalid data.	System prompts error message and prevents adding Result.	Pass
12	Student views registered events.	Student can successfully see registered events.	Pass
13	Register student with valid details.	Student is successfully registered.	Pass
14	Student views registered events.	Student can successfully see registered events.	Pass
15	Stress test: Multiple users register for an event simultaneously.	System remains functional with minimal delay.	Pass

5. CONCLUSION AND FUTURE ENHANCEMENT

5.1 CONCLUSION:

The **NEXT-GEN CLG EVENT MANAGER** enhances efficiency, security, and convenience for event management. By digitizing registrations, approvals, and participant tracking, it streamlines operations and reduces administrative burdens. Its scalable architecture ensures adaptability for various event sizes, while future enhancements like biometric verification promise even greater security and reliability

5.2 FUTURE ENHANCEMENT:

The implementation of the **NEXT-GEN CLG EVENT MANAGER** marks a significant improvement in event management within educational institutions. By **digitizing the event approval and tracking process**, the system enhances security, efficiency, and transparency.

The structured **modular approach** ensures seamless interaction across different user roles, making the system robust and scalable:

- **Administrative Module**: Empowers administrators with efficient event approval and tracking capabilities.
- **Staff Module**: Allows staff to manage event schedules, approve student participation, and track event progress.
- **Student Module**: Provides a simplified and intuitive experience for students to register for events and view results.

By eliminating **manual paperwork and delays**, this system enhances accountability and streamlines event management. Moving forward, **continuous evaluation and feature upgrades** will ensure that the system adapts to evolving institutional needs, reinforcing secure and efficient event tracking.

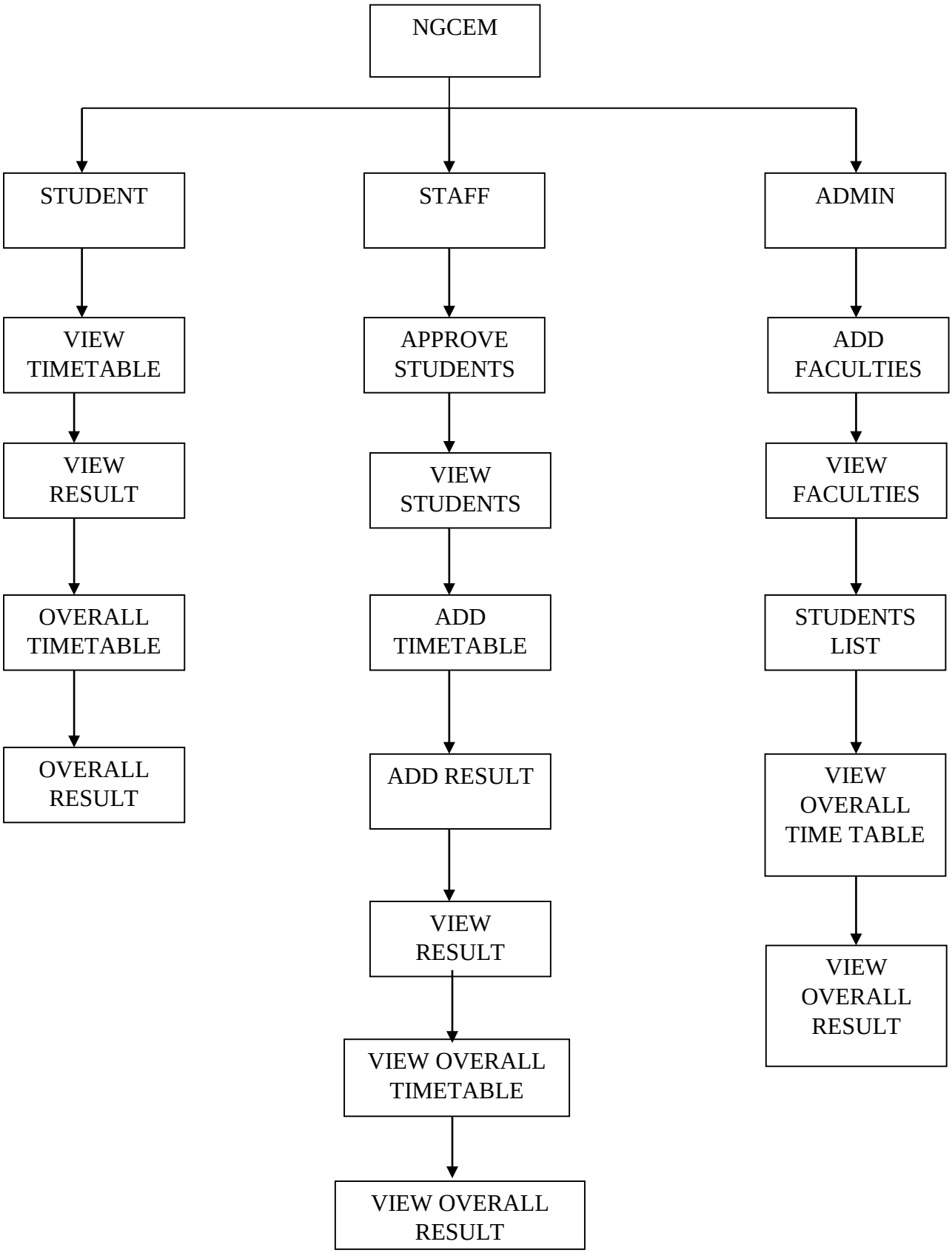
6. BIBLIOGRAPHY

WEBSITE URL REFERENCES

- https://www.w3schools.com/
- https://www.phptutorial.net/

APPENDICES

A. DATA FLOW DIAGRAM



B. TABLE STRUCTURE

1. Table Name: faculty_info

FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS	DESCRIPTION
Id	Int	3	Primary Key	Faculty Id
empid	Varchar	10	Not Null	Employee Id
name	Varchar	30	Not Null	Faculty Name
gender	Varchar	10	Not Null	Gender
mobile	BigInt	10	Not Null	Mobile number
email	Varchar	30	Not Null	Email Address
event	Varchar	20	Not Null	Assigned Event
username	Varchar	20	Not Null	Username
password	Varchar	20	Not Null	Password

2. Table Name: students_info

FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS	DESCRIPTION
Id	Int	3	Primary Key	Student Id
name	Varchar	30	Not Null	Student Name
rolno	Varchar	15	Not Null	Student Roll no
dob	Date	---	Not Null	Date of Birth
gender	Varchar	10	Not Null	Gender
mobile	BigInt	10	Not Null	Mobile number
email	Varchar	30	Not Null	Email Address
department	Varchar	20	Not Null	Department name
event	Varchar	20	Not Null	Registered Event
username	Varchar	20	Not Null	Username
password	Varchar	15	Not Null	Password
status	Varchar	8	Not Null	Approval Status
college	Varchar	20	Not Null	College Name

2. Table Name: timetable

FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS	DESCRIPTION
Id	Int	3	Primary Key	Student Id
event	Varchar	20	Not Null	Event Name
date	Date	---	Not Null	Event Date
time	Varchar	7	Not Null	Event Time
clsno	Varchar	10	Not Null	Venue/Room number
status	Varchar	8	Not Null	Event Status
first	Varchar	20	Not Null	First place winner
second	Varchar	20	Not Null	Second place winner
third	Varchar	20	Not Null	Third place winner

C. SAMPLE CODING

NGCEM Login:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>College Event Management</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/bootstrap/5.3.0/css/bootstrap.min.css">
    <style>
        body {
            font-family: 'Arial', sans-serif;
        }
        .navbar {
            background: linear-gradient(to right, black,black);
        }
        .carousel-item img {
            width: 100%;
            height: 500px;
            object-fit: cover;
        }
        .footer {
```

```
            background: #333;
            color: white;
            display: flex;
            flex-direction: row;
            justify-content: space-evenly;
            padding: 20px;
        }
        @media (min-width:521px) and (max-width:1200px) {
            .carousel-item img {
            width: 100%;
            height: 300px;
            object-fit: cover;
            }
        }
        @media (max-width:520px) {
            .carousel-item img {
            width: 100%;
            height: 200px;
            object-fit: cover;
            }
        }
    </style>
</head>
```

```
<body>
  <nav class="navbar navbar-expand-lg navbar-dark">
    <div class="container">
      <a class="navbar-brand" href="#">Kaamadhenu Arts and Science College</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarNav">
        <ul class="navbar-nav ms-auto">
          <li class="nav-item"><a class="nav-link" href="#">Home</a></li>
          <li class="nav-item dropdown">
            <a class="nav-link dropdown-toggle" href="#" id="eventsDropdown" role="button" data-bs-toggle="dropdown">
              LOGIN
            </a>
            <ul class="dropdown-menu">
              <li><a class="dropdown-item" href="./ad_login.php">ADMIN</a></li>
              <li><a class="dropdown-item" href="./faculty_login.php">FACULTY</a></li>
              <li><a class="dropdown-item" href="./student_login.php">STUDENT</a></li>
            </ul>
          </li>
          <li class="nav-item"><a class="nav-link" href="./student_reg.php">REGISTER</a></li>
```

```
                <li class="nav-item"><a class="nav-link" href="./contactus.html">About
US</a></li>
            </ul>
        </div>
    </div>
</nav>

<div id="eventCarousel" class="carousel slide" data-bs-ride="carousel"data-bs-
interval="3000">
    <div class="carousel-indicators">
        <button type="button" data-bs-target="#eventCarousel" data-bs-slide-to="0"
class="active"></button>
        <button type="button" data-bs-target="#eventCarousel" data-bs-slide-
to="1"></button>
        <button type="button" data-bs-target="#eventCarousel" data-bs-slide-
to="2"></button>
    </div>
    <div class="carousel-inner">
        <div class="carousel-item active">
            <img src="https://media.licdn.com/dms/image/C561BAQEcwx7nO-
7DEg/company-
background_10000/0/1589529957987/kascsathy_cover?e=2147483647&v=beta&t=xIQs0k_
T6PwoqS0-MHAQm_cgEjya2QIS0pk3ViMLNfY" alt="Campus">
            <hgroup>
        <h2></h2>
    </hgroup>
</div>
```

```
        <div class="carousel-item">
            <img src="https://kascsathy.ac.in/images/background/bg1.jpg" alt="College Building">
        </div>
        <div class="carousel-item">
            <img src="https://www.kascsathy.ac.in/images/facilities/green/02.jpg" alt="Green Campus">
        </div>
    </div>
    <button class="carousel-control-prev" type="button" data-bs-target="#eventCarousel" data-bs-slide="prev">
        <span class="carousel-control-prev-icon"></span>
    </button>
    <button class="carousel-control-next" type="button" data-bs-target="#eventCarousel" data-bs-slide="next">
        <span class="carousel-control-next-icon"></span>
    </button>
  </div>
  <div class="container my-5">
    <h2 class="text-center mb-4">Upcoming Events</h2>
    <div class="row">
      <div class="col-md-4">
        <div class="card event-card">
          <img src="./quiz.jpg" class="card-img-top"
            alt="Quiz Event" style="height: 300px;">
```

```
            <div class="card-body">
                <h5 class="card-title">Quiz Competition</h5>
                <p class="card-text">Test your knowledge and compete with the best minds.</p>
                <a href="./student_login.php" class="btn btn-primary"><i class="fa-solid fa-arrow-right"></i> Learn More</a>
            </div>
        </div>
    </div>
    <div class="col-md-4">
        <div class="card event-card">
            <img src="./Paper-Presentation.jpg" class="card-img-top"
                alt="Paper Presentation" style="height: 300px;">
            <div class="card-body">
                <h5 class="card-title">Paper Presentation</h5>
                <p class="card-text">Showcase your research and ideas in front of experts.</p>
                <a href="./student_login.php" class="btn btn-primary"><i class="fa-solid fa-arrow-right"></i> Learn More</a>
            </div>
        </div>
    </div>
    <div class="col-md-4">
        <div class="card event-card">
            <img src="./work.jpg" class="card-img-top"
```

```
                alt="Workshop" style="height: 300px;">
                <div class="card-body">
                    <h5 class="card-title">Technical Workshop</h5>
                    <p class="card-text">Enhance your skills with hands-on workshops conducted by experts.</p>
                    <a href="./student_login.php" class="btn btn-primary"><i class="fa-solid fa-arrow-right"></i> Learn More</a>
                </div>
            </div>
        </div>
    </div>
    <div class="footer">
        <p>&copy; 2025 NGCEM </p>
        <p>Developed By - VISHNU VIGHAS, BSC CS, 2022 - 2025</p>
    </div>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap/5.3.0/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

ADMIN PAGE:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css" rel="stylesheet">
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"></script>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css">
    <style>
        body {
            font-family: "Arial", sans-serif;
            margin: 0;
            background-color: #f8f9fa;
        }
        .sidenav {
            height: 100%;
            width: 260px;
            position: fixed;
            top: 0;
```

```
    left: 0;
    background: linear-gradient(135deg, #6a11cb, #2575fc);
    padding-top: 20px;
    color: white;
    box-shadow: 2px 0 8px rgba(0, 0, 0, 0.2);
}
.sidenav h2 {
    text-align: center;
    color: #fff;
    margin-bottom: 30px;
    font-size: 22px;
    font-weight: bold;
}
.sidenav a {
    padding: 15px 25px;
    text-decoration: none;
    font-size: 18px;
    color: white;
    display: block;
    transition: 0.3s;
    border-radius: 6px;
    margin: 8px 15px;
}
```

```
.sidenav a:hover {
    background-color: rgba(255, 255, 255, 0.2);
    box-shadow: 0px 4px 8px rgba(0, 0, 0, 0.1);
}
.sidenav a.active {
    background-color: #4e54c8;
    color: white;
}
.sidenav .fa {
    margin-right: 12px;
}
.main {
    margin-left: 260px;
    padding: 30px;
    min-height: 100vh;
}
.btn-logout {
    margin-top: 50px;
    background: linear-gradient(135deg, #ff512f, #dd2476);
    color: white;
    padding: 15px 20px;
    border: none;
    border-radius: 6px;
    font-size: 18px;
```

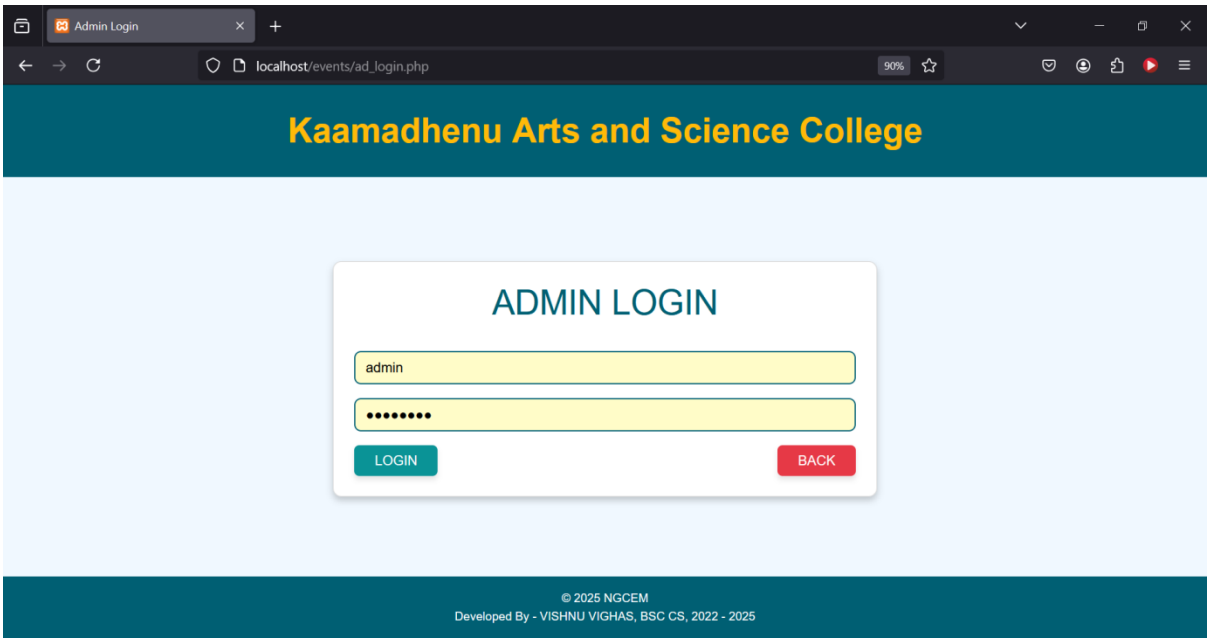
```
    cursor: pointer;
    display: block;
    text-align: center;
    transition: all 0.3s;
    width: calc(100% - 30px);
    margin-left: 15px;
}
.btn-logout a {
    color: white;
    text-decoration: none;
}
.btn-logout:hover {
    transform: scale(1.05);
    box-shadow: 0px 4px 15px rgba(0, 0, 0, 0.3);
}
@media screen and (max-width: 768px) {
    .sidenav {
        width: 200px;
    }
    .main {
        margin-left: 200px;
    }
    footer {
        margin-left: 200px;
```

```
        }
    }
    @media screen and (max-width: 576px) {
        .sidenav {
            width: 100%;
            height: auto;
            position: relative;
        }
        .main {
            margin-left: 0;
        }
        footer {
            margin-left: 0;
        }
    }
  </style>
</head>
<body>
  <div class="sidenav">
    <h2>Admin Dashboard</h2>
    <a href="./ad_addfac.php">
      <i class="fa-solid fa-user-plus"></i> Add Faculty
    </a>
    <a href="./ad_facultydet.php">
```

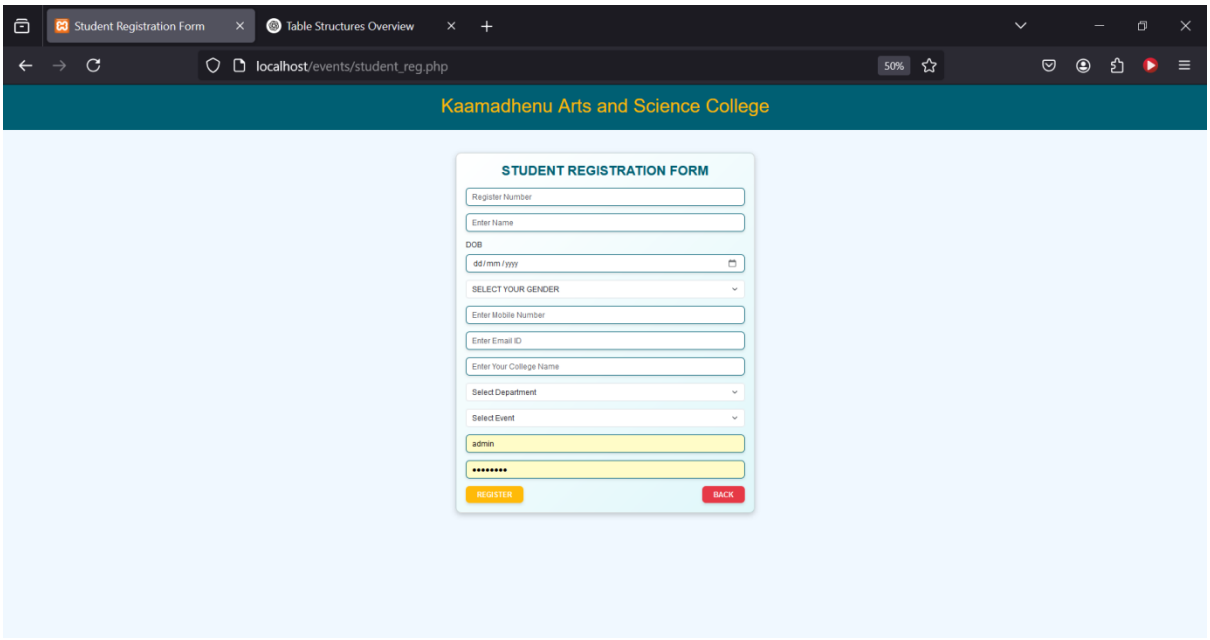
```
        <i class="fa-solid fa-users"></i> View Faculty
      </a>
      <a href="./ad_students.php">
        <i class="fa-solid fa-graduation-cap"></i> Students
      </a>
      <a href="./ad_viewtimetable.php">
        <i class="fa-solid fa-calendar-days"></i> View Time Table
      </a>
      <a href="./ad_viewresult.php">
        <i class="fa-solid fa-chart-bar"></i> View Result
      </a>
      <button class="btn-logout">
        <a href="index.php">
          <i class="fa-solid fa-right-from-bracket"></i> Logout
        </a>
      </button>
    </div>
</body>
</html>
```

D. SAMPLE INPUT:

ADMIN LOGIN:

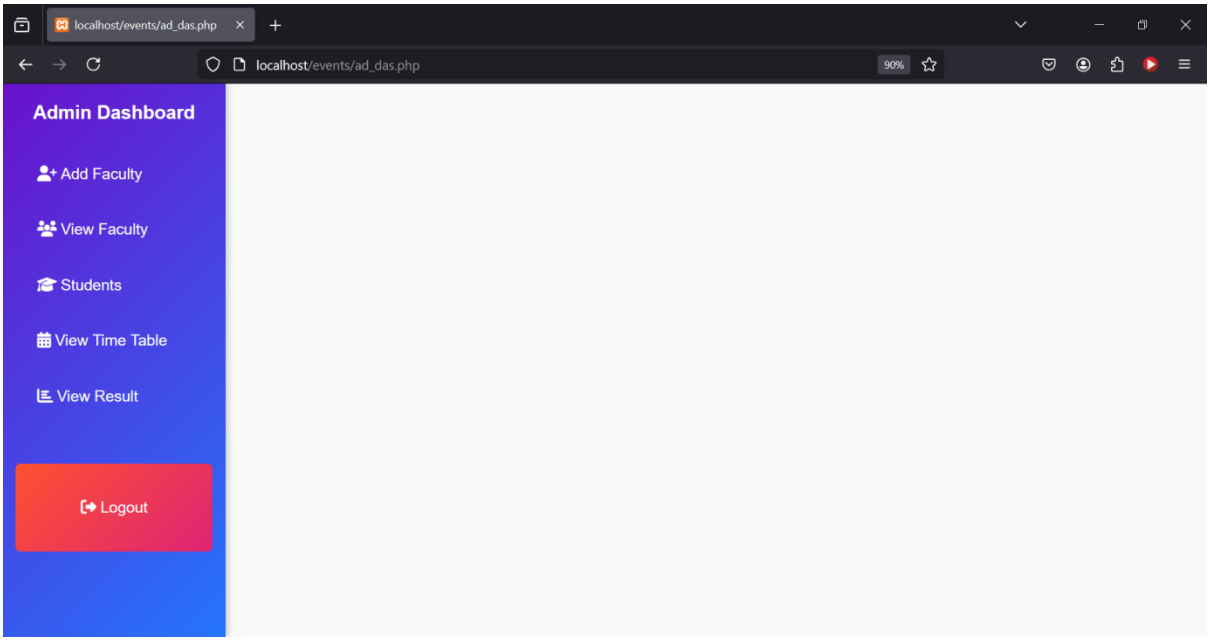


STUDENT REGISTRATION FORM:

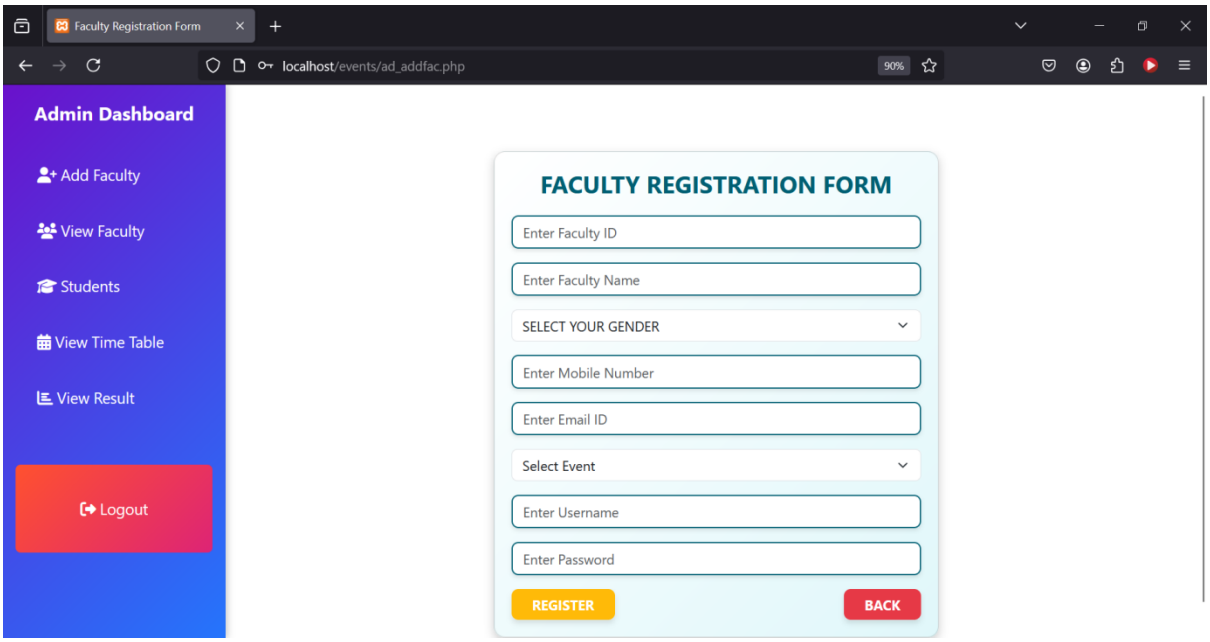


E. SAMPLE OUTPUT:

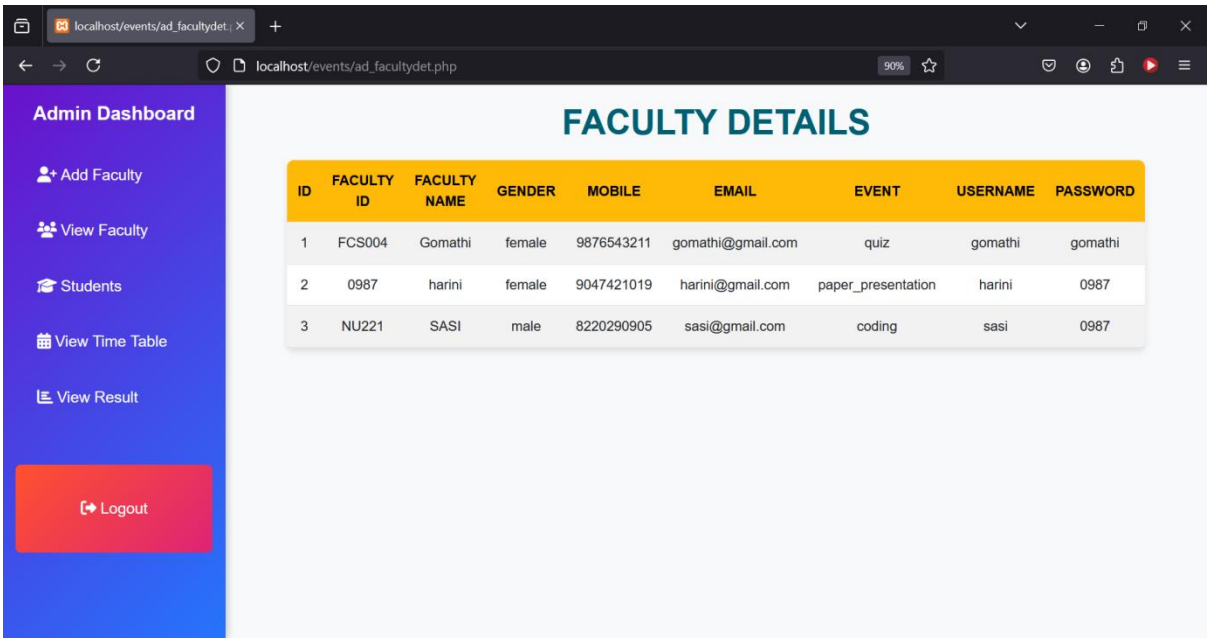
HOME PAGE:



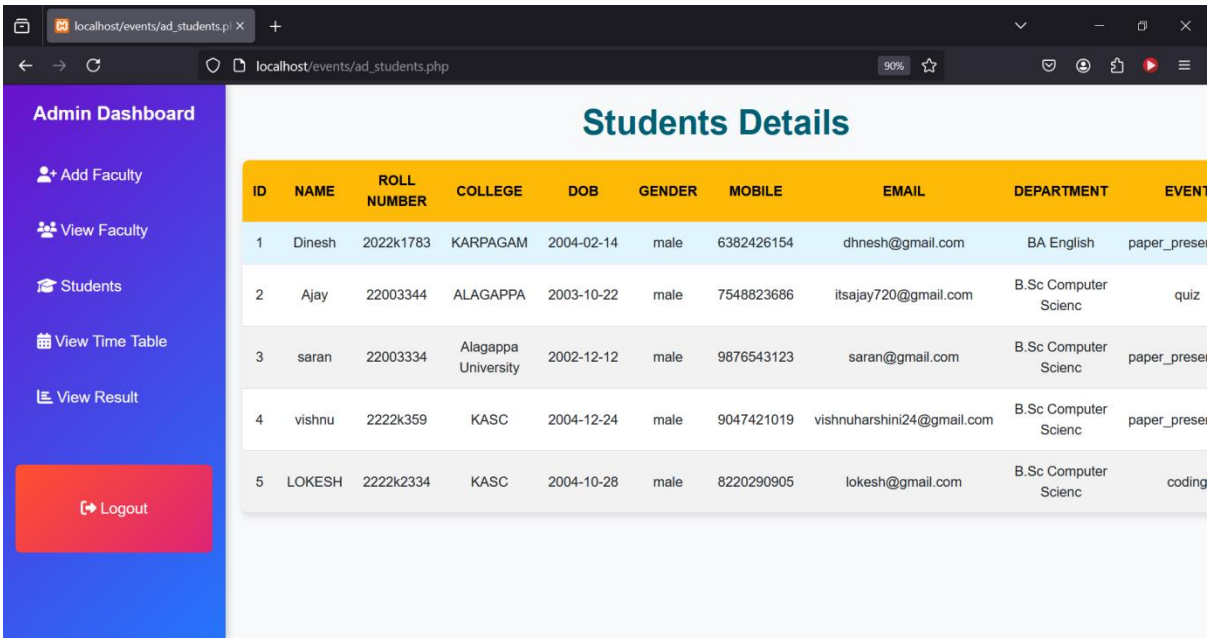
Add Faculty Page:



View Faculty Page:



Students Details Page:



View Result Page: