Software Engineering and Web Technologies Laboratory

Integrated Project on

Workshop Attended

Bachelor of Engineering

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

Team No: 3

Div: A

Aakash Jain	01FE21BCS228	141
Sparsh Kumar	01FE21BCS363	158
Guruprasad Pattar	01FE21BCS226	140
Mallikarjun R M	01FE21BCS227	155
Santoshi Vajrangi	01FE21BCS298	144

Faculty In charges

Padmashree D.Desai, Sadaf Mujawar

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

HUBLI–580 031 (India)

Academic year 2023-24

Table of Content

Chapter 1
INTRODUCTION

1. Preamble

In the dynamic landscape of academic and professional events, the current methods of managing and disseminating information have proven to be inadequate and fragmented. Conferences, symposia, seminars, and workshops often suffer from outdated information, a lack of real-time updates, and inconsistent data across various platforms. This inefficiency poses challenges for both organizers and attendees, hindering their ability to seamlessly engage with events.

Our project leverages a combination of cutting-edge technologies to provide a seamless, interactive, and efficient experience. The project integrates a powerful Database Management System - WampServer (Windows, Apache, MySQL, PHP/Python/Perl) - to establish a local server environment for handling event data with precision and speed. This backend framework is complemented by React.js, a frontend framework chosen for its ability to deliver a responsive and engaging user interface.

An Integrated Development Environment (IDE) serves as the coding and debugging hub, enabling our team to create a robust, reliable, and user-friendly platform. As we embark on this journey, EventHub is not just a technological solution but a commitment to transforming the way academic and professional events are managed. By addressing the current inefficiencies, we aspire to elevate the experience for both organizers and attendees, setting a new standard for event management in the digital age. EventHub is poised to become the go-to platform for anyone seeking to seamlessly plan, participate in, and stay updated on a diverse range of events.

2. Problem Definition

The current process of managing and displaying information about various academic and professional events such as conferences, symposia, seminars, and workshops is often fragmented and inefficient. Organizers and attendees struggle with outdated information, lack of real-time updates, and inconsistent data across different platforms. There is a need for a centralized system that can handle event details effectively and provide users with a seamless, interactive experience.

.

3. Objectives

1. To develop a centralized platform for the management of event details.
2. Secure user authentication and login.
3. To simplify administrative tasks associated with event management for increased operational efficiency.
4. To Validation for data entry to ensure data integrity.
5. To create Form for entering event details.

Chapter 2
SOFTWARE REQUIREMENT SPECIFICATIONS

2.1.1 Use case diagram

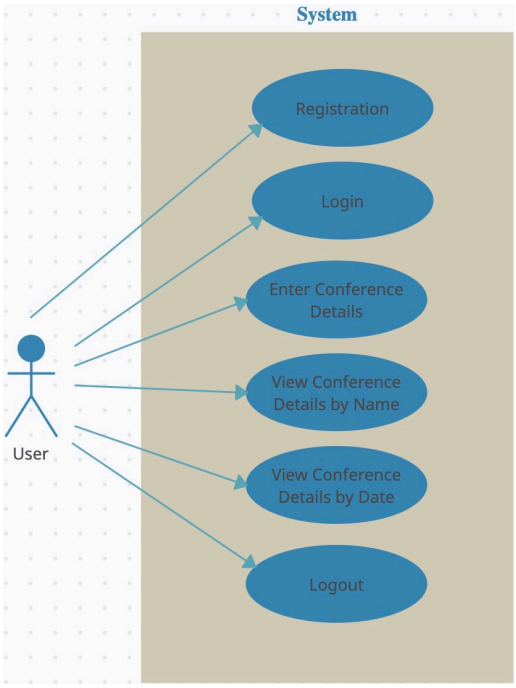

Fig. 2.1. Use case Diagram

The conference management system supports several key use cases. Users can register for a conference, creating an account and enrolling in events. The login functionality allows authentication for accessing user accounts. Additionally, users can input conference details, including name, date, and location. The system facilitates viewing conference details based on either the conference's name or date. A logout feature is available for users to terminate their sessions and log out of their accounts. These use cases collectively provide a comprehensive and efficient conference management experience.

2.1.2 Functional Requirements

1. It should perform Search Functionality:
    - Search events by name and date.
2. It should have User Interaction:
    - Interactive buttons for editing and administration.
3. It should be able to update the database with the credentials of a new user.
4. It should have real-time Updates:
    - Reflect changes immediately in the system.

2.2 Non-Functional Requirements

1. It must be able to respond within 2 seconds for all operations.
2. It must have growing number of users and event entries.
3. It must be able to perform secure login and user authentication.
4. There must be intuitive user interface and navigation.

2.3.1 Hardware requirements

The successful implementation of the centralized event management system necessitates a robust hardware infrastructure. The following are the recommended hardware requirements.

1. Server

- Processor: Multi-core processor with sufficient processing power
- RAM: Minimum of 8 GB or as dictated by system load
- Storage: Adequate storage capacity for database management and system operation

2. Networking:
    - High-speed internet connection for real-time updates
    - Network infrastructure capable of handling concurrent connections during peak usage

3. Security:
    a. Firewalls and intrusion detection/prevention systems
    b. Secure socket layer (SSL) for encrypted data transmission
4. Backup Systems:
    a. Regularly scheduled backup mechanisms for data protection
    b. Redundancy measures to ensure system availability.

2.3.2 Software requirements

To ensure the seamless operation of the centralized event management system, the following software components are essential:

1. Database Management System:
    - WampServer (Windows, Apache, MySQL, PHP/Python/Perl) for providing a local server environmentBackend Framework.
2. Frontend Framework:
    - React.js for developing a responsive and interactive user interface
3. Version Control:
    - Git for source code management and collaboration
4. Security:
    - Implementation of security best practices, including regular software updates and patches
5. Development Tools:
    - Integrated Development Environment (IDE) for coding and debugging purposes

2.4 Test plan and Test cases

TC_ID	Description	Steps	Input	Expected Output	Actual Output	Test Case Result
TC01	Valid Username	1. Create UsernameValidator instance	“Rahul"	True	True	Pass
TC02	Valid Password	1. Create PasswordValidator instance	“Rahul@123”	True	True	Pass
TC03	Invalid Username	1. Create UsernameValidator instance	“Rajesh"	False	False	Pass
TC04	Invalid Password	1. Create PasswordValidator instance	“Abcd@123”	False	False	Pass

Chapter 3
SYSTEM DESIGN AND IMPLEMENTATION

3.1 MVC Architecture

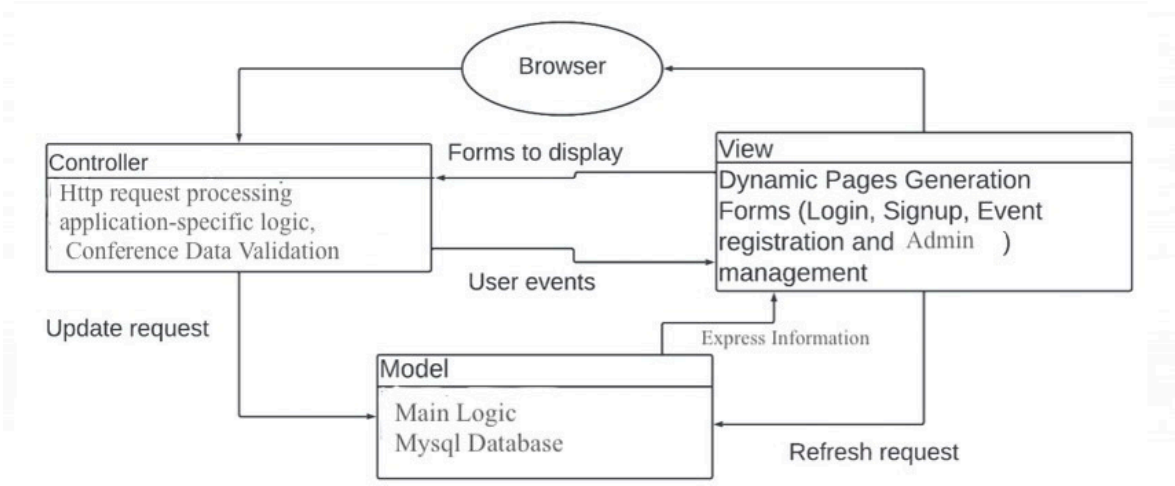

Fig. 3.1 MVC Architecture

3.2 MERN framework

The MERN stack is a popular JavaScript-based framework for building web applications, consisting of four main components: MongoDB, Express.js, React, and Node.js. Each component plays a specific role in the development process, collectively providing a full-stack solution. Let's explore each component in the context of the Result Analysis web application:

MongoDB:

MongoDB is well-suited for storing student information, grades, and other academic data. Its flexibility accommodates changes in the data structure, providing scalability for future enhancements.

Express.js:

Express.js serves as the backend framework, handling HTTP requests from the client, routing those requests to the appropriate endpoints, and managing the communication with the MongoDB database.

React:

React is used on the client side to create a responsive and interactive user interface. It facilitates the rendering of result summaries, graphs, and other visual components, providing a seamless user experience.

Node.js:

Node.js serves as the server runtime, enabling the execution of server-side logic. It works in conjunction with Express.js to handle incoming HTTP requests, process data, and communicate with the MongoDB database.

3.3 Detailed Database Description (Mongo DB/ MySQL)

1] Faculty Schema (FacultyModel)

The database structure is designed to capture student details, course information, and the relationship between them. It allows for efficient retrieval of data related to student performance, course details, and their interconnections. The unique constraints and relationships ensure data integrity and support the objectives of the project. Lets refer database as Models:

a. *Faculty Schema Fields:*
    1. Username: String - Represents the name of the faculty.
    2. password: String - Represents the password of the faculty.
b. *Faculty Model:*

    The facultyModel is created using mysql with the name 'faculty' and the defined facultySchema. This model is used to interact with the 'users' collection in mysql.

2] Conference Schema (ConferenceModel)

The second schema is for representing conference details that faculty might attend. It contains information about the faculty, and details about the event itself.

a. Conference Schema Fields:

1. Faculty Name: String (required) - Represents the name of the faculty who attending the conference.
2. Conference Details: String (required) - Represents the details of the conference..
3. Place: String (required): Represents the location name of the faculty the conference is registered to.
4. Date start: Date (required): Represents the starting date of the event.
5. Date end: Date (required):- Represents the last date of the event.
6. Invited/Deputed: String (required): Represents the if faculty was Invited or Deputed for conference**.**

2.4. Modules description

1. Login page for login..
2. Signup page for signing in.
3. Page for entering conference details.
4. Page to view conference details by faculty name.
5. Page to view conference details by conference date.

1. **Registration Module:**
    - *Description:* The Registration Module manages user accounts and provides functionality for faculty registration. It ensures secure authentication, allowing individuals to create and manage their profiles.

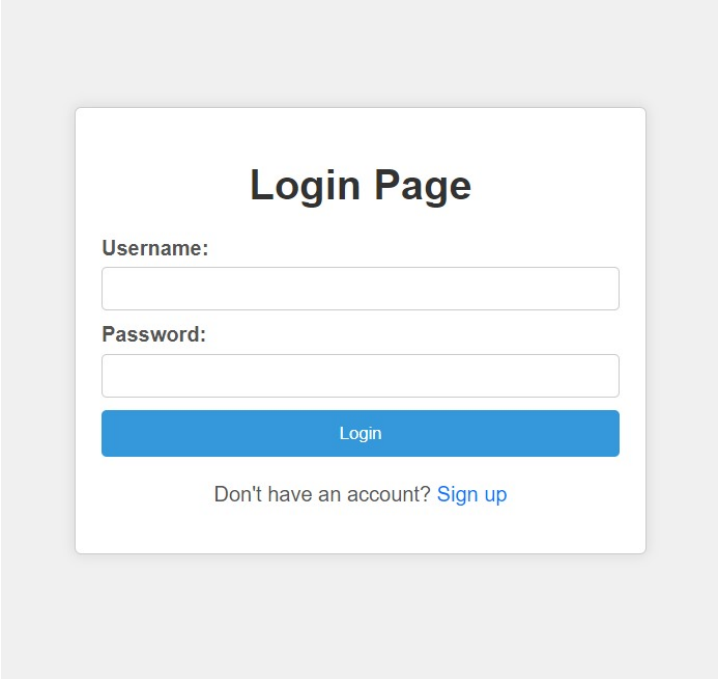

Fig. 3.2. Website Login Page

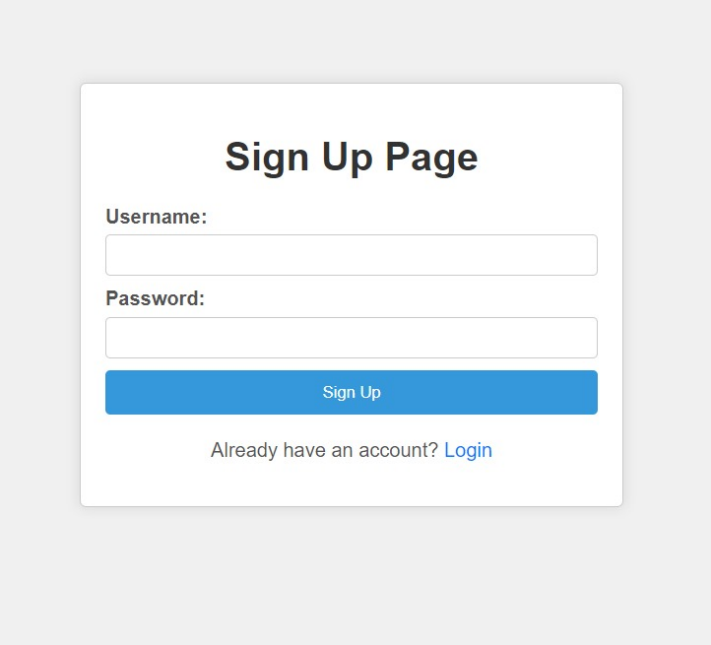

Fig. 3.3. Signup Page

2. **Conference Booking Module:**

   - *Description:* This module enables faculty to book for conference seamlessly. Faculty can select the conference details.

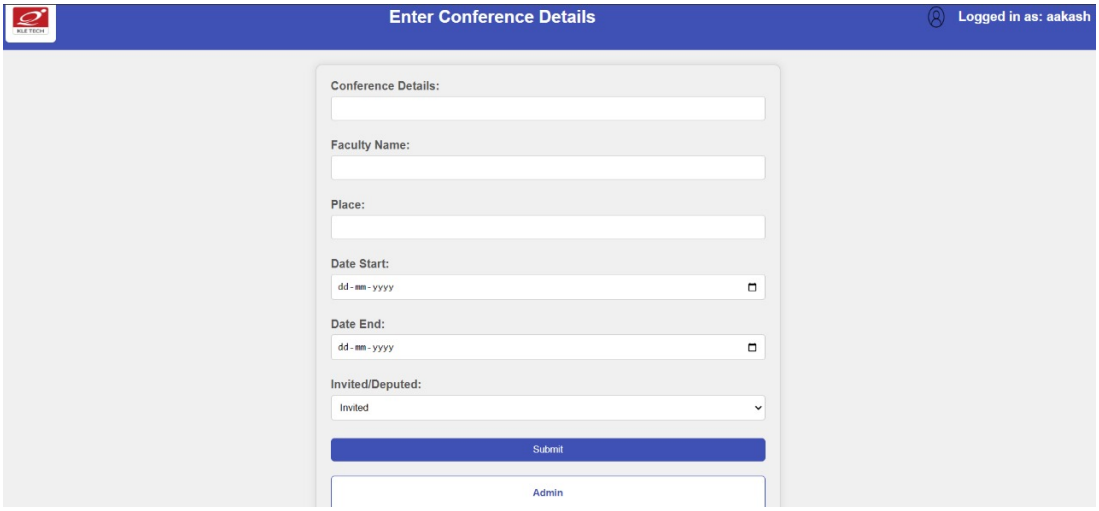

Fig. 3.4. Conference Details Page

3. **Event Display Module:**

   - *Description:* The Event Display Module focuses on presenting information about upcoming conferences. It includes features such as an event calendar, event lists, and detailed event pages.

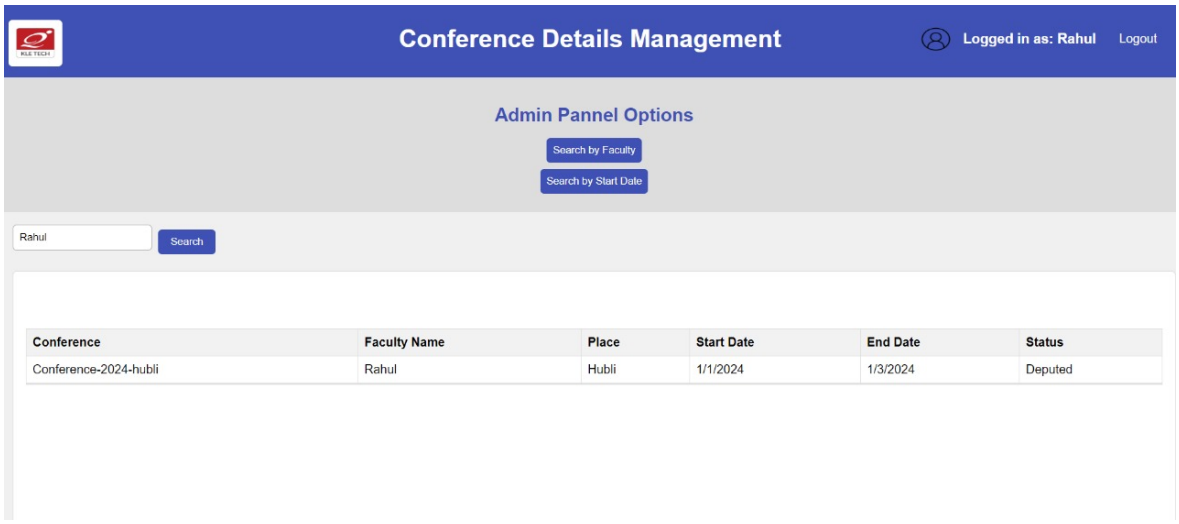

Fig. 3.5. Admin Function Page

4. Calendar Module:

   - Description: The Calendar Module provides a visual representation of scheduled conferences. Faculty can view events on a dynamic calendar, facilitating easy navigation and planning.

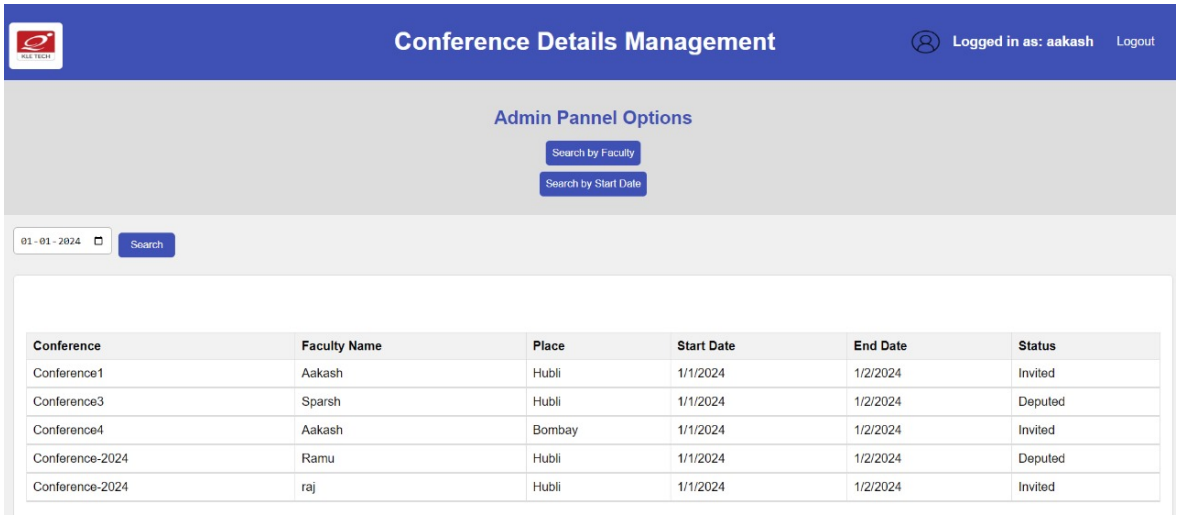

Fig. 3.6. Admin Functionality for Conference date

Chapter 4
RESULTS AND DISCUSSIONS

4.1 Results/Snapshots with description

A] Registration successful: Signifies the completion of the registration process. Users successfully enrolled, validating the system's capability to handle registrations effectively.

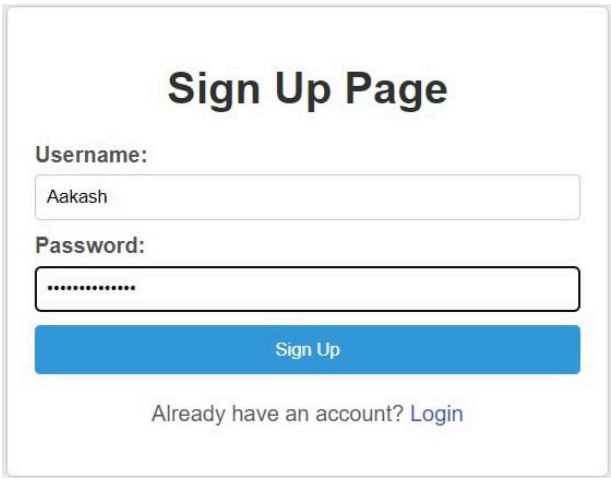

Fig. 4.1. Signed Successfully

B] Login Successful: Indicates that faculty have securely accessed the conference management system.

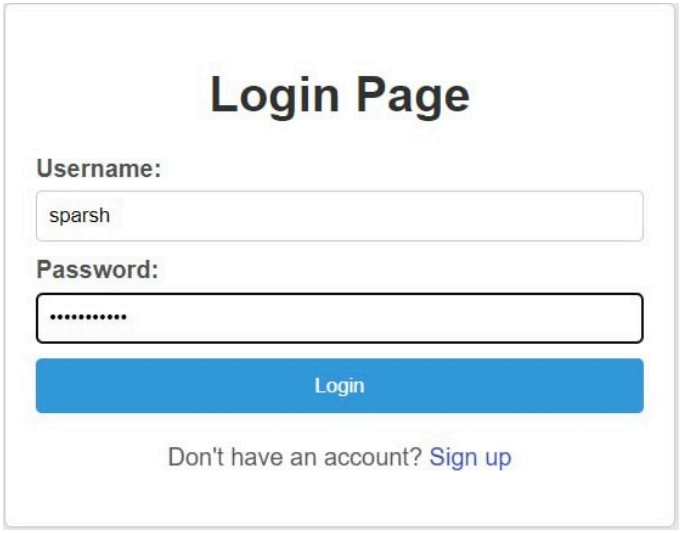

Fig. 4.2. Login Successful

C] Conference details entered: Denotes the successful entering of a conference within the university system. This functionality confirms the seamless registration for conference.

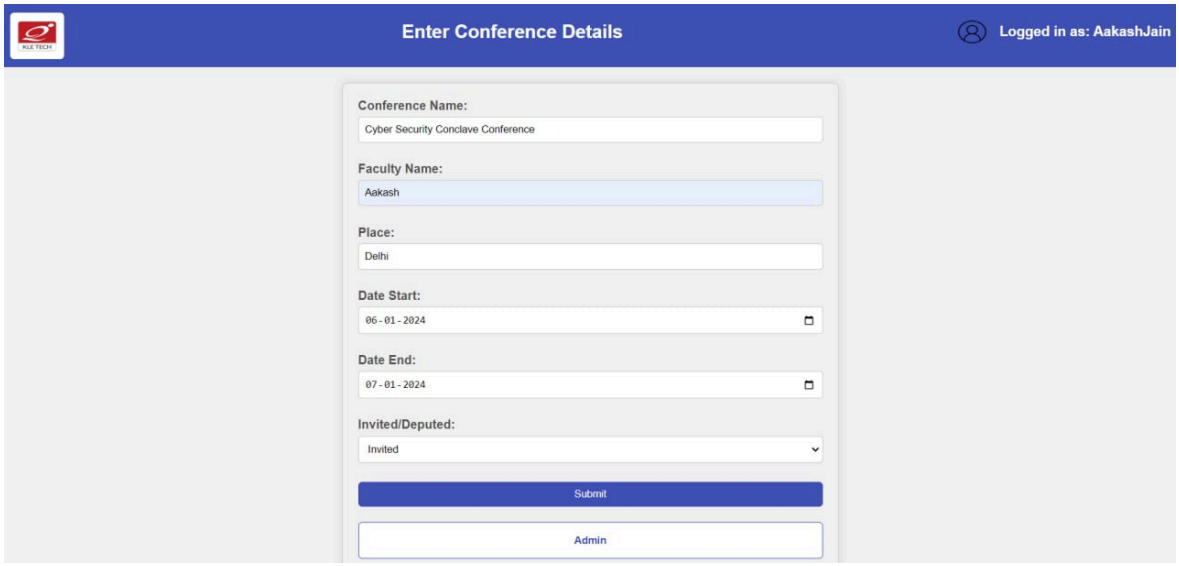

Fig. 4.3. Details Submitted to Server

D] Conference fetched Successfully: Confirms the successful fetching of an conference within the university's event and club management system. This functionality ensures that event details are accurately displayed, contributing to effective communication.

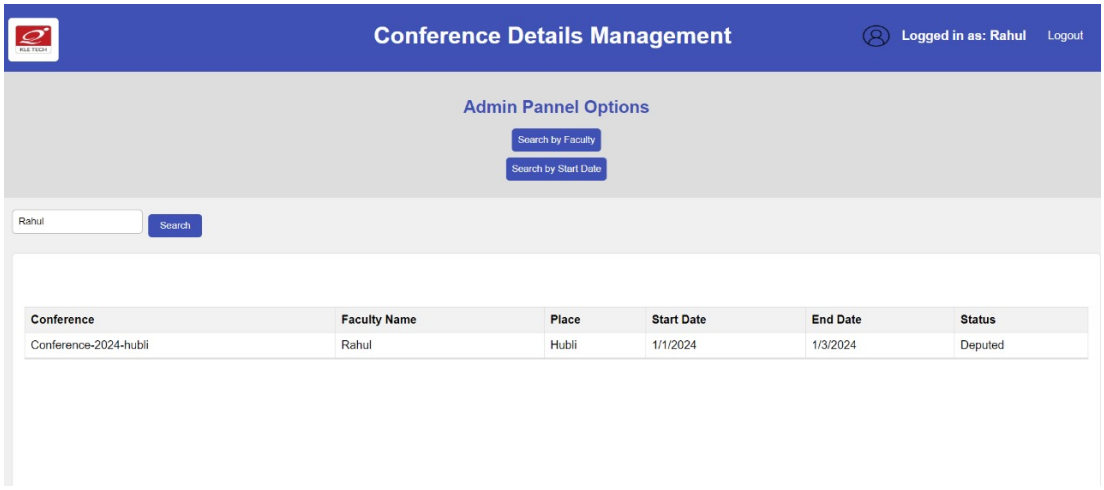

Fig. 4.4. Admin Faculty Details Retrieved

4.2 Testing Report

A] Signin Success: Testers simulate sign attempts, utilizing the tool to evaluate error messages and security measures. This ensures a secure and user-friendly experience.

	Command	Target	Value
1	✓ open	http://localhost:3000/login	
2	✓ set window size	1552x832	
3	✓ click	linkText=Sign up	
4	✓ click	css=input:nth-child(2)	
5	✓ type	css=input:nth-child(2)	Rahul
6	✓ click	css=input:nth-child(4)	
7	✓ type	css=input:nth-child(4)	Rahul123
8	✓ click	css=button	

B] Login Success: Testers simulate incorrect login attempts, utilizing the tool to evaluate error messages and security measures. This ensures a secure and user-friendly experience.

9	✓ click	css=input:nth-child(2)	
10	✓ type	css=input:nth-child(2)	Rahul
11	✓ click	css=input:nth-child(4)	
12	✓ type	css=input:nth-child(4)	Rahul123
13	✓ click	css=button	
14	✓ click	id=conferenceDetails	
15	✓ type	id=conferenceDetails	Conference-2024-hubli

C] Create Event Success: Selenium IDE verifies the success of event creation in the university's system. Testers use the tool to confirm accurate processing and display of event details.

	Command	Target	Value
16	✓ click	id=facultyName	
17	✓ type	id=facultyName	Rahul
18	✓ click	id=place	
19	✓ type	id=place	Hubli
20	✓ click	id=dateStart	
21	✓ type	id=dateStart	2024-01-01
22	✓ click	id=dateEnd	
23	✓ type	id=dateEnd	2024-01-03
24	✓ click	id=status	
25	✓ select	id=status	label=Deputed
26	✓ click	css=.button	
27	✓ click	css=.admin-btn	
28	✓ click	css=.sidebar-button:nth-child(2)	
29	✓ click	css=.input	
30	✓ type	css=.input	Rahul

C] Logout Success: Selenium IDE is instrumental in validating successful logos within the university's Event and Club management system.

31	✓ *click*	css=.button	
32	✓ *mouse over*	css=.button	
33	✓ *mouse out*	css=.button	
34	✓ *click*	css=.sidebar-button:nth-child(3)	
35	✓ *click*	css=.input	
36	✓ *type*	css=.input	2024-01-01
37	✓ *click*	css=.button	
38	✓ *click*	css=.logout-button	

4.3 Testing Tool

Selenium IDE:

Event and club management in a university setting, Selenium IDE serves as an invaluable tool for ensuring the smooth functioning of web applications. This Integrated Development Environment simplifies the testing process by enabling testers and developers to effortlessly record, edit, and replay interactions within the university's event and club management system. With Selenium IDE, users can create test scripts without the necessity of in-depth programming skills, thanks to its user-friendly interface and compatibility as a Firefox and Chrome browser extension. This automation tool plays a pivotal role in validating the various functionalities of the system, such as creating and updating events, managing club memberships, and ensuring a seamless user experience. By utilizing Selenium IDE, testing professionals can efficiently execute and refine test cases, contributing to the reliability and efficiency of event and club management processes within the university

5. Conclusion & Future scope.

In summary, the centralized system designed to manage and display information about academic and professional events represents a significant improvement in addressing the fragmented and inefficient processes prevalent in the current scenario. By providing a unified platform, the project has successfully mitigated issues like outdated information, lack of real-time updates, and data inconsistency across multiple channels. The resulting platform not only streamlines event management but also elevates the user experience, offering a seamless and interactive environment for both organizers and attendees.

Future Scope: Looking ahead, the project holds promising potential for further evolution and expansion. Future development could explore integration with emerging technologies like augmented reality (AR) or virtual reality (VR) to enhance the immersive aspect of the user experience. Additionally, advanced analytics tools could be incorporated to gather valuable insights from user interactions, empowering organizers to make informed, data-driven decisions for future events. Furthermore, the creation of a dedicated mobile application could be considered, providing users with a more convenient and accessible way to engage with event information. Overall, the project's future scope extends to incorporating innovative technologies and refining features to continuously improve the event management ecosystem.