MentorConnect : A Mentor Allocation System

The **College Mentor Allocation System** is a comprehensive web-based platform designed to facilitate the assignment of junior students to faculties, based on their department, academic interests, and career goals. The system allows students to register and create profiles by providing key details such as their area of study, interests, and future aspirations. Once registered, students are paired with mentors who align with their academic and professional objectives. The platform ensures that students receive personalized guidance and mentorship from seniors who can offer relevant advice and support. It also features an event registration module where students can view and register for upcoming mentorship events, providing opportunities for offline or online interactions with mentors in a structured environment.

In addition to mentor allocation, the system allows students to update their profiles with new interests or goals as their academic journey progresses. It also includes an anonymous feedback feature where students can review their mentoring experience, helping the administration monitor mentor performance and improve the program continuously. The event registration aspect plays a key role, encouraging students to actively participate in mentorship-driven activities like workshops or networking sessions, enhancing their learning experience beyond regular mentor meetings. This holistic approach ensures a streamlined and effective mentorship process that benefits both juniors and mentors, fostering meaningful academic relationships.

College Mentor Allocation System Project – Detailed Description

Overview:

The College Mentor Allocation System is designed to streamline the process of connecting students with mentors within a college. The system allows students to be allocated mentors based on various parameters such as their department, academic interests, future goals, and career aspirations. This project is aimed at facilitating academic guidance, career advice, and personal development by pairing students with experienced mentors.

The system also includes features such as event registration, mentor feedback, and the ability to update student interests and goals over time. It's built as a web-based application with PHP for backend development and MySQL for the database. The system aims to enhance student-mentor interaction by providing a structured and easily navigable platform.

Key Features:

1. **Student and Mentor Registration:**
    - **Student Registration**: Students can create accounts, log in, and update their profile information, which includes academic details, interests, and future career goals.

- **Mentor Registration**: Mentors have their profiles, including their expertise, department, and areas of interest.

2. **Mentor Allocation:**
   - Each student is allocated a mentor based on predefined criteria such as:
     - **Department**: Mentors and students from the same department are prioritized.
     - **Interests**: Aligning student interests with mentor expertise.
     - **Career Goals**: Matching students with mentors who can guide them toward their future goals.
   - The system ensures that each mentor handles a limited number of students to allow for effective mentorship.
3. **Mentorship Events:**
   - The platform includes a calendar and event registration system where students can view and register for upcoming mentorship events.
   - Events are designed to promote interaction between students and mentors, including seminars, workshops, and one-on-one mentoring sessions.
4. **Student Profile Management:**
   - Students can update their **interests** and **future goals** as they progress through their academic journey.
   - This allows the system to reassess and potentially reassign mentors based on evolving needs.
5. **Feedback System:**
   - Students can submit anonymous feedback about their mentors.
   - This feedback is used for assessing mentor performance and ensuring student satisfaction.
   - Mentors can use feedback to improve the quality of mentorship.
6. **Dynamic Dashboard:**
   - **Student Dashboard**: Displays allocated mentor details, upcoming events, and provides the option to update their profile or submit feedback.
   - **Mentor Dashboard**: Allows mentors to view the list of students under their guidance, track interactions, and receive feedback.
7. **Login and Authentication:**
   - The system has role-based authentication for students and mentors.
   - Secure login ensures that only registered users can access their respective dashboards.

System Modules:

1. **Student Module:**
   - Register and create a profile.
   - Update personal information (interests, future goals).
   - View allocated mentor and contact them.
   - Register for mentorship events.
   - Submit feedback on mentors.

2. **Mentor Module:**
    - Register as a mentor.
    - View allocated students and their profiles.
    - Track and manage mentees.
    - Receive feedback from students.
3. **Admin Module (optional):**
    - Manage student and mentor data.
    - Oversee mentor allocations.
    - Manage event scheduling and registration.
    - Review student feedback and mentor performance.
4. **Event Management Module:**
    - Schedule, manage, and promote mentorship events.
    - Allow students to register for events.
    - Track attendance and participation.

Database Design:

The system uses **MySQL** to store and manage data. The key tables include:

1. **Students Table:**
    - Stores student details (ID, name, department, interests, future goals, mentor_id).
2. **Mentors Table:**
    - Stores mentor details (ID, name, department, interests, email).
3. **Feedback Table:**
    - Stores feedback submitted by students (id, student_id, mentor_id, feedback_text, rating, created_at).
4. **Events Table:**
    - Stores event information **(**id, event_name, event_date, event_time, event_location, event_description, capacity, registered_count).
5. **Registrations Table:**
    - Records mentor-student pairings (**id, student_id, event_id)**.

Workflow:

1. **Student Registration:**
    - A student signs up and fills in their details (department, interests, goals).
    - Upon login, they see whether they have been assigned a mentor.
2. **Mentor Allocation:**
    - The system checks for available mentors who match the student's department and interests.
    - If a match is found, the mentor is assigned to the student.
    - The student can view their mentor’s details, such as name, department, and contact information.

3. **Student Interaction with Mentors:**
    - Students can reach out to mentors via the system (email or message).
    - They can ask questions, seek advice, and participate in scheduled events with their mentors.
4. **Feedback Collection:**
    - After each mentorship interaction or periodically, students can submit feedback on their mentors.
    - This feedback is collected and stored anonymously in the system.
5. **Events and Registration:**
    - The system displays upcoming mentorship events that students can register for.
    - Event details such as the date, time, and description are shown on the dashboard.
    - Once registered, students receive reminders and event details.

Functional Dependencies:

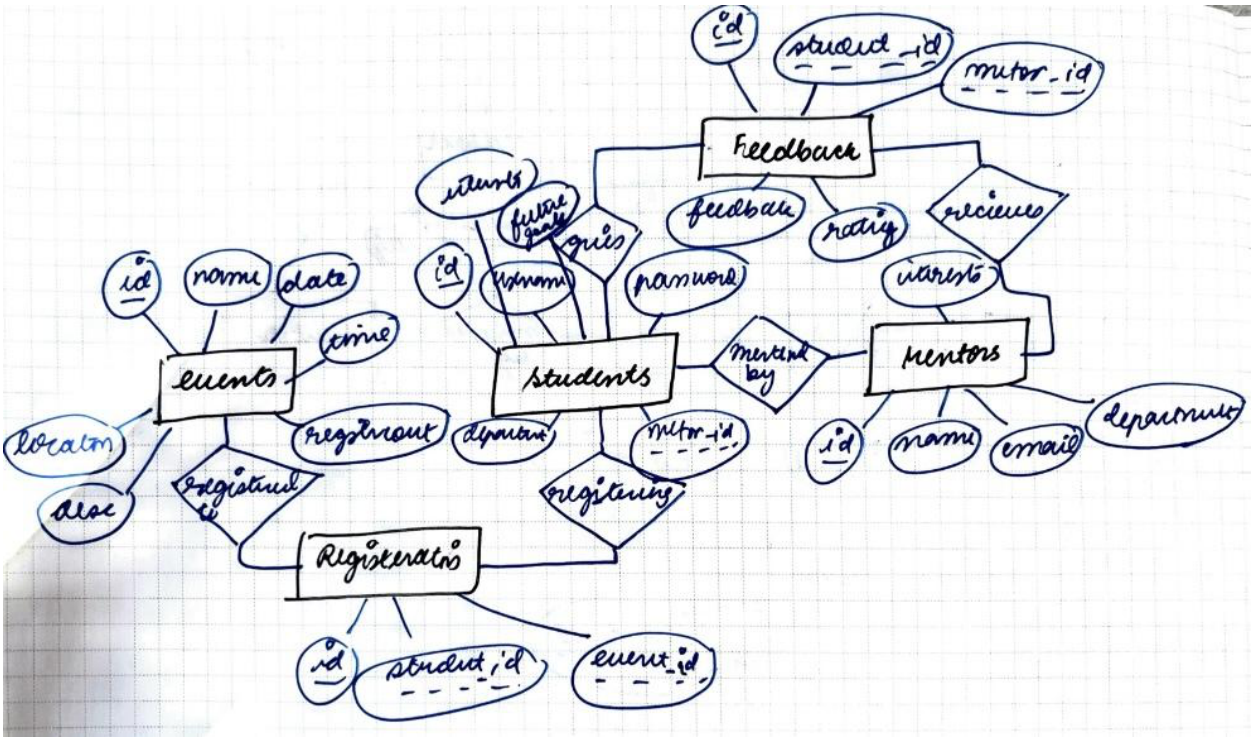
1. **Events:**
    - **id → event_name, event_date, event_time, event_location, event_description, capacity, registered_count**
    - **event_name, event_date → event_time, event_location, event_description, capacity, registered_count**
2. **Feedback:**
    - **id → student_id, mentor_id, feedback_text, rating, created_at**
    - **student_id, mentor_id → feedback_text, rating, created_at**
3. **Mentors:**
    - **id → name, email, department, interests**
    - **email → name, department, interests**
4. **Registrations:**
    - **id → student_id, event_id**
    - **student_id, event_id → id**
5. **Students:**
    - **id → username, name, password, department, mentor_id, interests, future_goals**
    - **username → name, password, department, mentor_id, interests, future_goals**

Final Schema in BCNF:

1. **Events (id, event_name, event_date, event_time, event_location, event_description, capacity, registered_count)**
    - **Functional Dependencies:**
        - **id → event_name, event_date, event_time, event_location, event_description, capacity, registered_count**

        - event_name, event_date → event_time, event_location, event_description, capacity, registered_count
2. **Feedback (id, student_id, mentor_id, feedback_text, rating, created_at)**
    - **Functional Dependencies:**
        - id → student_id, mentor_id, feedback_text, rating, created_at
        - student_id, mentor_id → feedback_text, rating, created_at
3. **Mentors (id, name, email, department, interests)**
    - **Functional Dependencies:**
        - id → name, email, department, interests
        - email → name, department, interests
4. **Registrations (id, student_id, event_id)**
    - **Functional Dependencies:**
        - id → student_id, event_id
        - student_id, event_id → id
5. **Students (id, username, name, password, department, mentor_id, interests, future_goals)**
    - **Functional Dependencies:**
        - id → username, name, password, department, mentor_id, interests, future_goals
        - username → name, password, department, mentor_id, interests, future_goals

ER Diagram:



Data in Events Table:

id	event_name	event_date	event_time	event_location	event_description	capacity
1	Web Development Workshop	2024-11-01	10:00:00	Main Auditorium	A workshop focused on building modern web applications.	50
2	AI and Machine Learning Bootcamp	2024-11-05	14:00:00	Room 101	An in-depth bootcamp on AI and Machine Learning techniques.	40
3	IoT Hands-on Seminar	2024-11-15	16:00:00	Lab 202	A seminar that focuses on building IoT devices and connecting them to the cloud.	60
4	Cybersecurity Basics and Best Practices	2024-11-20	11:00:00	Computer Lab	An introductory seminar on cybersecurity concepts and protecting systems.	50
5	Advanced Web Development Techniques	2024-11-30	15:00:00	Exhibition Hall	A session covering advanced web development techniques, including APIs and DevOps.	40
6	AI for Business Applications	2024-12-05	09:00:00	Conference Room	A workshop on how AI is transforming business operations and decision-making.	50

Data In Mentors Table:

id	name	email	department	interests
9	Dr. Kavita Rao	kavita.rao@university.edu	Computer Science	Artificial Intelligence, Machine Learning
10	Mr. Aman Gupta	aman.gupta@university.edu	Electronics and Communication Engineering	VLSI Design, Signal Processing
11	Prof. Suresh Malhotra	suresh.malhotra@university.edu	Mechanical Engineering	Thermodynamics, Fluid Mechanics
12	Dr. Shalini Reddy	shalini.reddy@university.edu	Biotechnology	Genetic Engineering, Biotechnology Research
13	Ms. Neha Arora	neha.arora@university.edu	Electrical Engineering	Power Systems, Renewable Energy
14	Prof. Vikram Patel	vikram.patel@university.edu	Information Technology	Cybersecurity, Blockchain
15	Dr. Rajesh Desai	rajesh.desai@university.edu	Chemical Engineering	Process Optimization, Nanotechnology
16	Ms. Priya Mehta	priya.mehta@university.edu	Civil Engineering	Structural Engineering, Sustainable Construction
17	Dr. Nikhil Verma	nikhil.verma@university.edu	Instrumentation and Control Engineering	Automation, Robotics
18	Mr. Anil Khanna	anil.khanna@university.edu	Mechanical Engineering	Manufacturing Systems, CAD/CAM
19	Prof. Jyoti Shah	jyoti.shah@university.edu	Biotechnology	Microbiology, Bioprocessing
20	Dr. Ramesh Naik	ramesh.naik@university.edu	Electrical Engineering	Control Systems, Smart Grids
21	Mr. Manish Tiwari	manish.tiwari@university.edu	Computer Science	Data Science, Cloud Computing
22	Prof. Shubham Sen	shubham.sen@university.edu	Civil Engineering	Geotechnical Engineering, Water Resources
23	Ms. Ritu Kumar	ritu.kumar@university.edu	Information Technology	Software Development, Artificial Intelligence

Structure Of Tables:

1 events

Creation: Oct 12, 2024 at 11:00 PM
Last update: Oct 20, 2024 at 12:48 PM

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
id	int(11)		No		auto_increment			
event_name	varchar(255)		No					
event_date	date		No					
event_time	time		No					
event_location	varchar(255)		Yes	NULL				
event_description	text		Yes	NULL				
capacity	int(11)		Yes	NULL				
registered_count	int(11)		Yes	0				

2 feedback

Creation: Oct 13, 2024 at 12:11 AM

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
id	int(11)		No		auto_increment			
student_id	int(11)		No			-> students.id ON UPDATE RESTRICT ON DELETE RESTRICT		
mentor_id	int(11)		No			-> mentors.id ON UPDATE RESTRICT ON DELETE RESTRICT		
feedback_text	text		Yes	NULL				
rating	int(11)		Yes	NULL				
created_at	timestamp		No	current_timestamp()				

3 mentors

Creation: Oct 12, 2024 at 09:46 PM

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
id	int(11)		No		auto_increment			
name	varchar(100)		No					
email	varchar(100)		No					
department	varchar(50)		Yes	NULL				
interests	varchar(255)		Yes	NULL				

4 registrations

Creation: Oct 12, 2024 at 11:00 PM
Last update: Oct 20, 2024 at 12:48 PM

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
id	int(11)		No		auto_increment			
student_id	int(11)		No			-> students.id ON UPDATE RESTRICT ON DELETE RESTRICT		
event_id	int(11)		No			-> events.id ON UPDATE RESTRICT ON DELETE RESTRICT		

5 students

Creation: Oct 12, 2024 at 09:03 PM
Last update: Oct 20, 2024 at 12:21 PM

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
id	int(11)		No		auto_increment			
username	varchar(50)		No					
name	varchar(100)		No					
password	varchar(255)		No					
department	varchar(50)		Yes	NULL				
mentor_id	int(11)		Yes	NULL		-> mentors.id ON UPDATE RESTRICT ON DELETE SET_NULL		
interests	varchar(255)		Yes	NULL				
future_goals	varchar(255)		Yes	NULL				

Technology Stack:

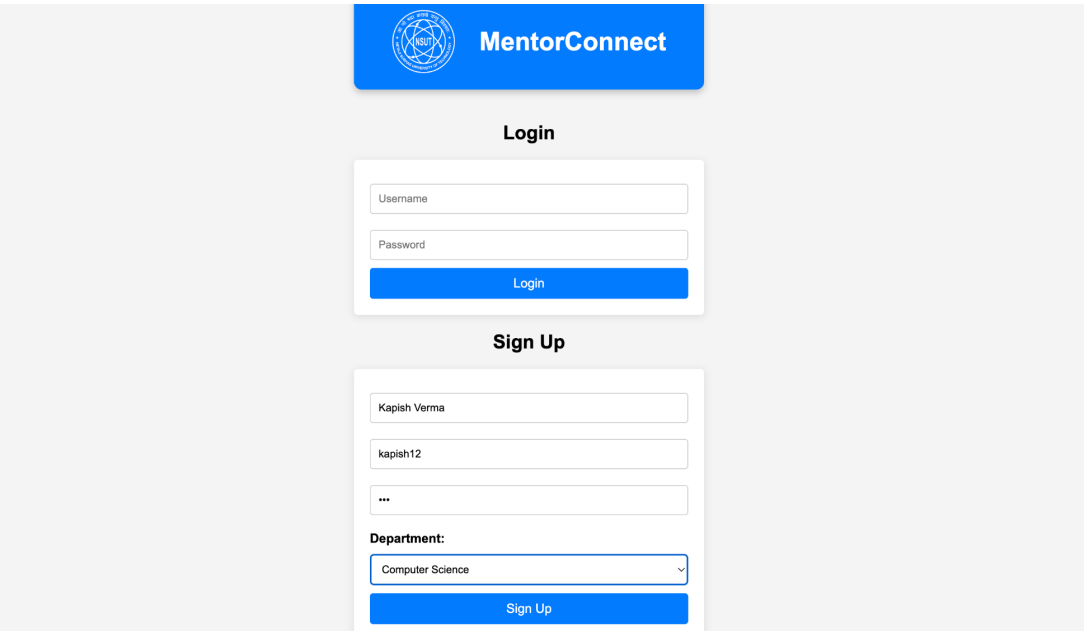
1. **Frontend**:
    - **HTML/CSS**: Used to create the structure and style of the web pages.
2. **Backend**:
    - **PHP**: Handles the logic for user registration, mentor allocation, event management, and feedback submission.
3. **Database**:
    - **MySQL**: Used to store student, mentor, and event data.
4. **Server**:
    - The application can be hosted on any web server like **Apache** or **Nginx**.

Challenges and Considerations:

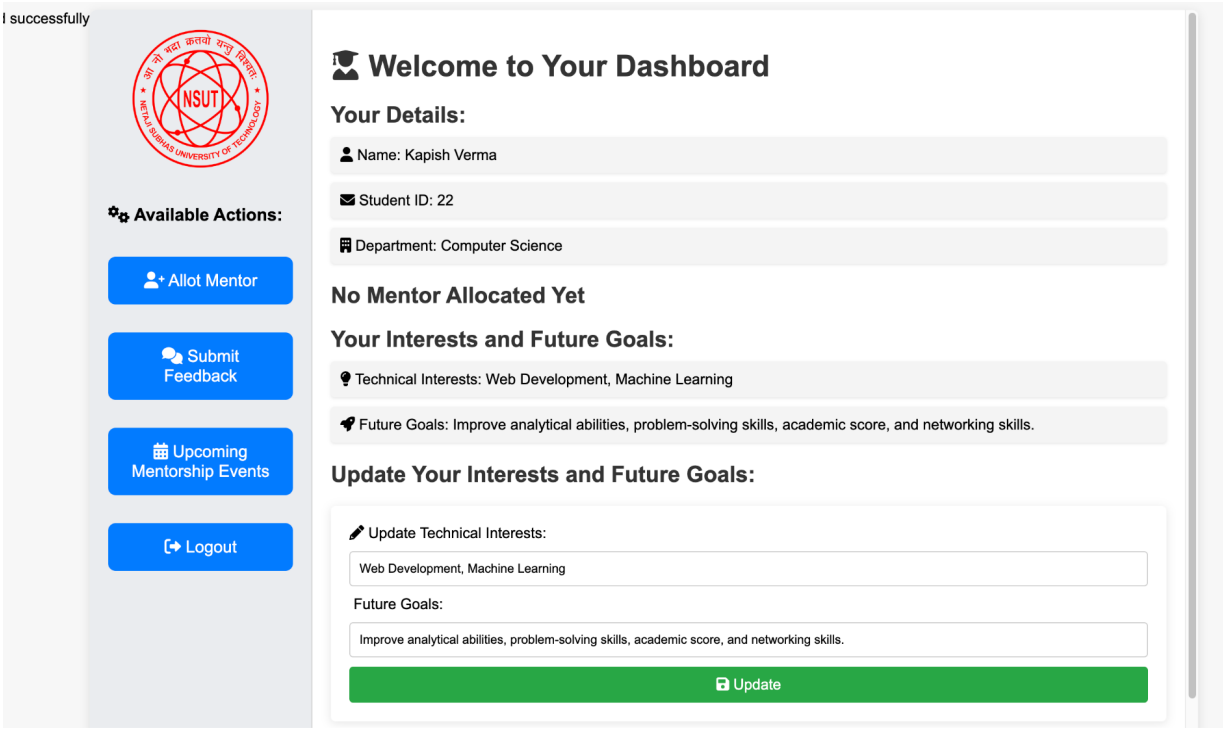
1. **Scalability**: As the number of students and mentors grows, the allocation algorithm should be optimized to ensure a balanced load across mentors.
2. **Security**:
    - Protect sensitive student and mentor information with secure authentication (sessions, hashed passwords).
    - Ensure that feedback remains anonymous to prevent bias or discomfort among students.
3. **User Experience**:
    - Keep the user interface simple and intuitive to encourage usage.
    - Ensure that students can easily update their interests and goals, which could influence mentor reallocation.
4. **Automation**:
    - Automating mentor-student allocation based on predefined rules can save time for administrators.

Proposed User Interface and Website:

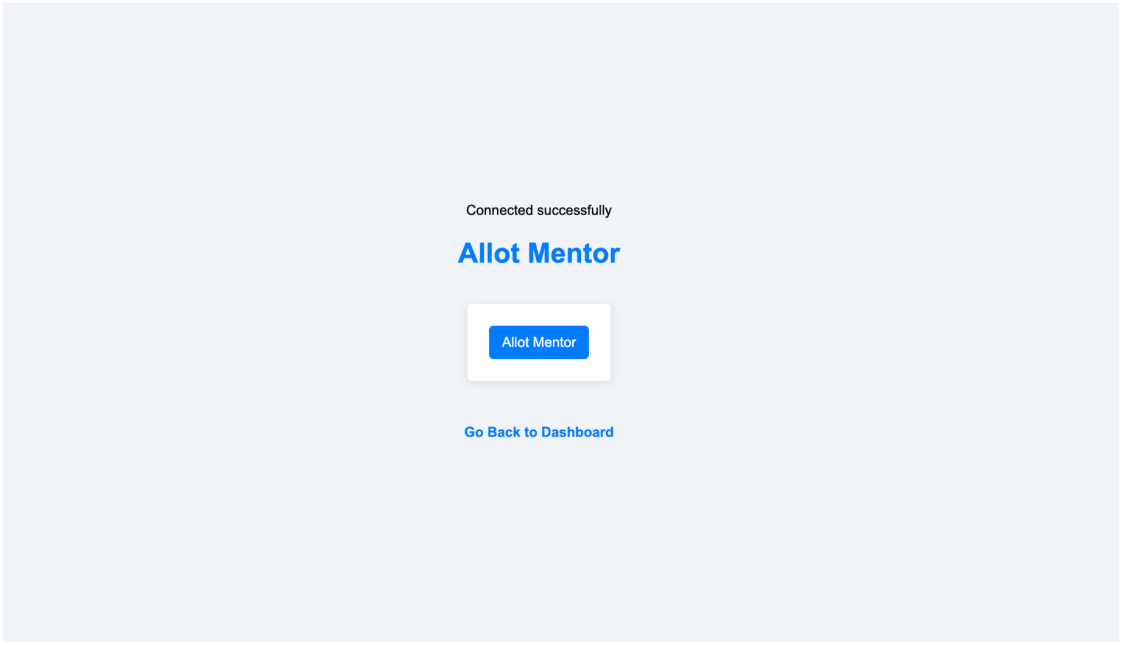
1. **Login/ Sign-up Page**

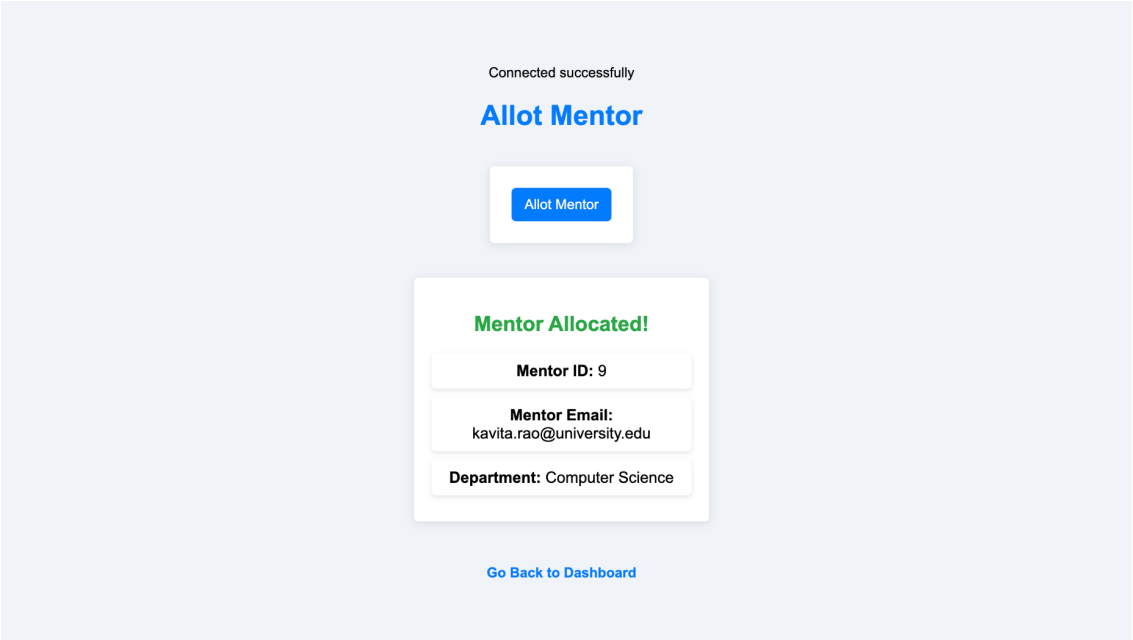


2. **User Dashboard:**

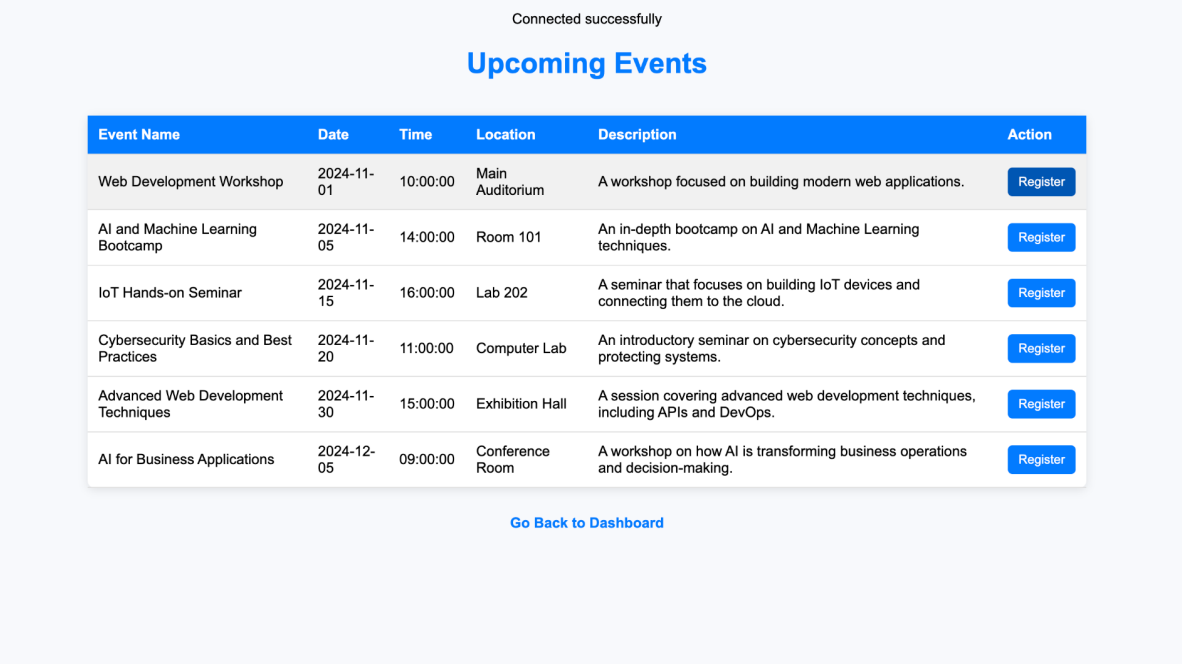


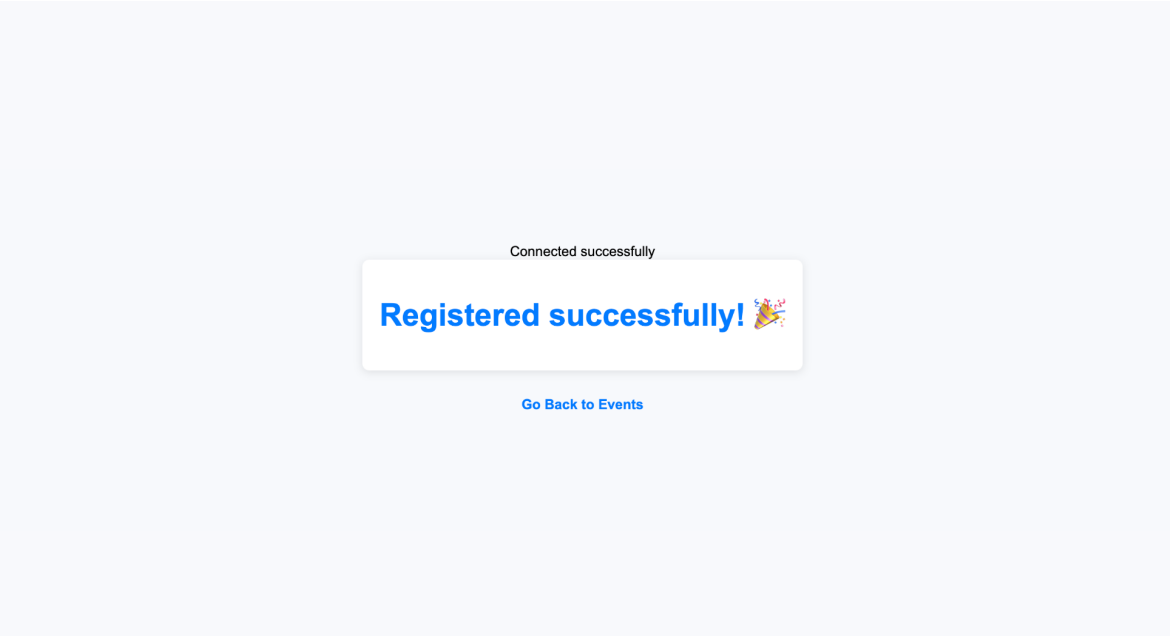
3. Mentor Allocation:



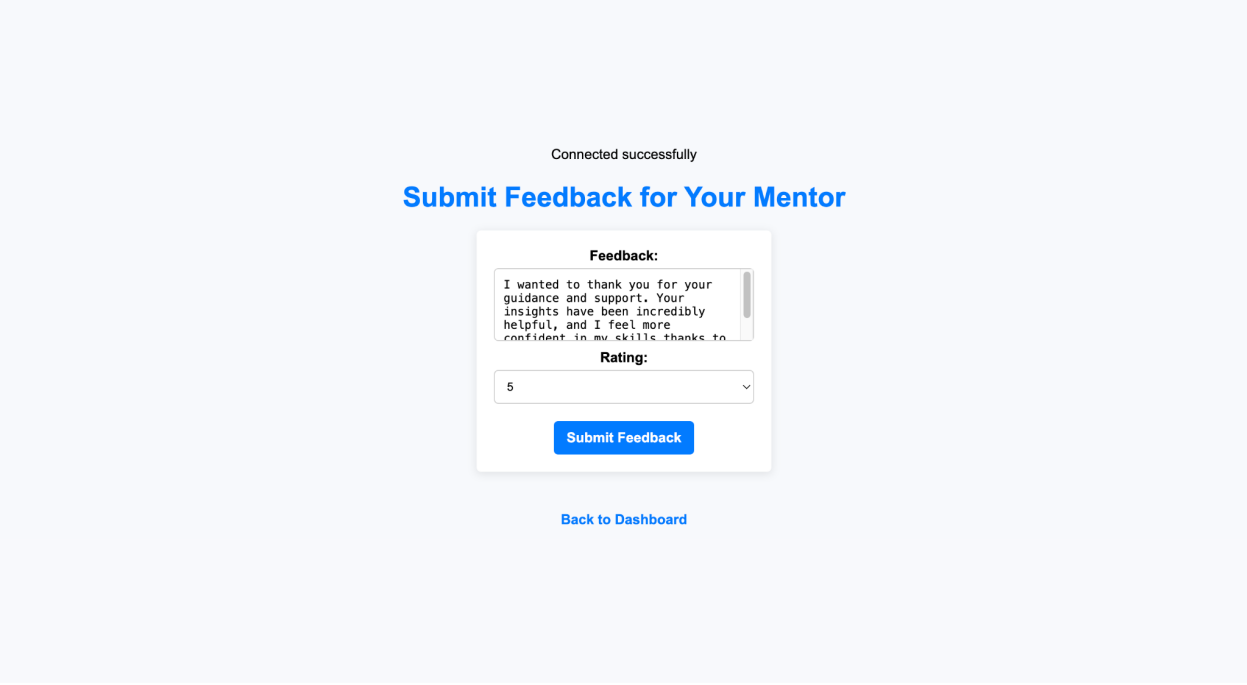


4. Events Registration:

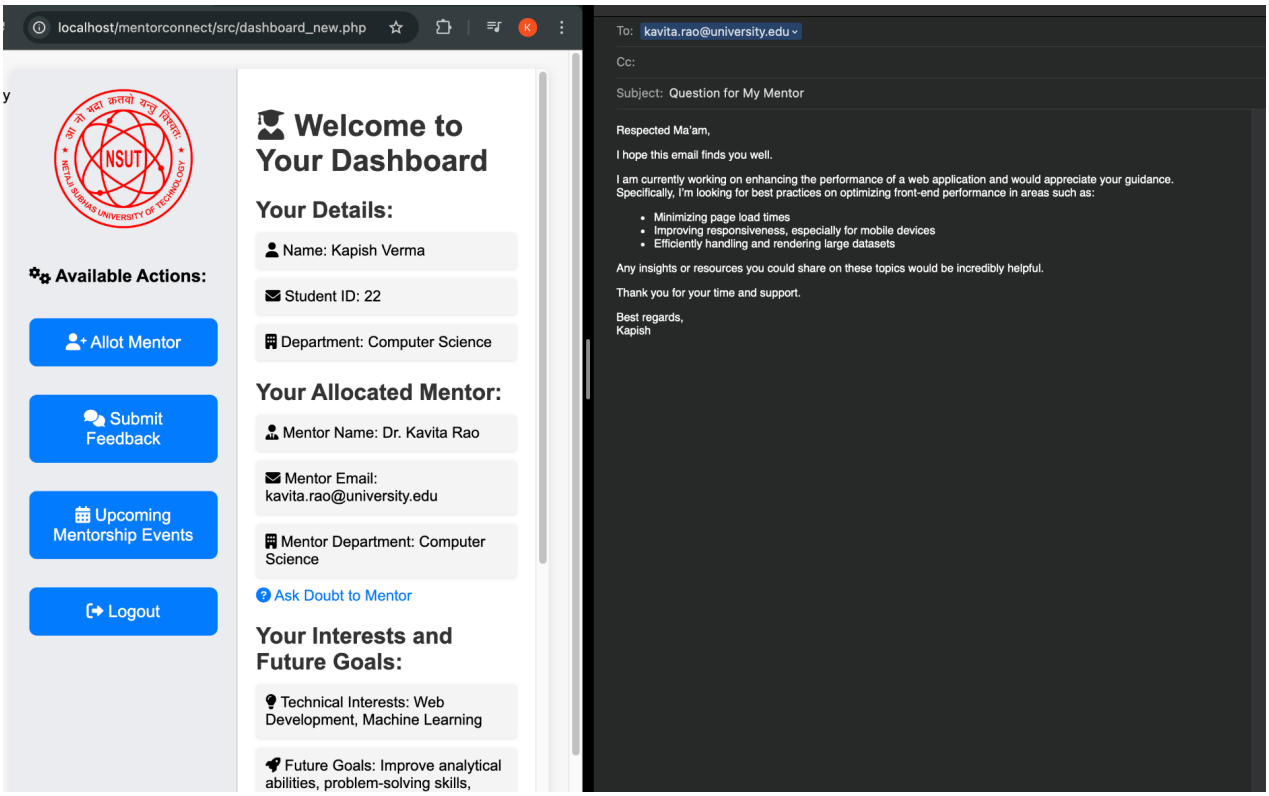




5. Feedback Submission by Student:



6. Doubt- Assistance



Future Enhancements:

1. **AI-Powered Recommendations**:
    - Use machine learning to improve mentor-student matching based on historical success rates, personality traits, and learning styles.
2. **Mobile App Integration**:
    - Develop a mobile version of the system to make it easier for students and mentors to interact on the go.
3. **Notifications**:
    - Integrate email or SMS notifications to remind students and mentors about upcoming events or feedback deadlines.

Link for all the Codes:

https://github.com/kapish19/MentorConnect
