**Project Name: Staff (University Social Media Platform)**

**Summary & Domain:**

* **Domain**: **Education**
* **Summary**: The application is a university-branded platform designed to facilitate interaction and collaboration between **faculty** and **students** in an academic social network environment. By allowing staff to showcase their academic contributions and enabling students to engage with them, the application enhances the university's reputation and creates a connected academic ecosystem. The platform supports features like staff profiles (publications, research, responsibilities), content sharing, online discussions, club and event access, and student queries via a callback feature. The platform is powered by **Spring Boot**, secured by **Spring Security**, and backed by a **MySQL** database.

**User Stories**

1. **Student Registration & Interaction**
   * Student should register, follow my professors, and like, comment, or share their posts.
   * Student able to participate in online discussions related to my subjects.
   * Student able to access information about clubs and upcoming university events.
   * Student able to submit queries through a "request a callback" form.
2. **Staff Profile Management**
   * Staff able to update my profile with my achievements, research, experience, and publications.
   * Staff able to create posts related to my subject or college.
   * Staff able to interact with students and answer their questions in online discussions.
3. **Search & Sorting**
   * As an admin or user, able to sort staff based on their designation and campus.

**Key Stakeholders**

* **University Administration**: Responsible for overall university branding and student engagement. They are interested in the app's impact on student-staff interactions and university reputation.
* **Faculty Members**: Looking to share their expertise, research, and updates with students while engaging in discussions.
* **Students**: Interested in following staff for learning purposes, engaging with subject-related content, and receiving answers to academic queries.

**Problem Statement**

Universities often struggle to create a cohesive platform that allows students to interact with faculty outside of the classroom setting. Moreover, there is a need for real-time discussions and information sharing between students and faculty members in a way that strengthens the university's brand. Existing tools often lack comprehensive engagement features and are not fully integrated with university resources like clubs and events.

**Use Cases**

1. **Staff Posting and Interaction**
   * Staff can post subject-related content, and students following them can engage by liking, commenting, or sharing the post.
   * Staff can answer questions in online discussion threads.
2. **Student Participation in Online Discussions**
   * Students can participate in thread-based discussions, similar to platforms like Quora or StackOverflow, and get answers from staff or other knowledgeable peers.
3. **Event and Club Access**
   * Students can view information about upcoming university events and join clubs.
4. **Callback Request Submission**
   * Students can submit queries via a callback form, which gets directed to the relevant department or staff member.
5. **Sorting of Staff Members**
   * Admins and students can sort staff based on their designation and the campus they belong to.

**Your Role**

As the **backend developer**, you:

* Designed and implemented the backend architecture using **Spring Boot**.
* Handled **authentication and authorization** using **Spring Security** to ensure secure user access.
* Managed **user profiles**, post interactions, and discussion threads.
* Developed the **"request a callback" feature** and integrated the event and club access functionality.
* Ensured efficient data storage and retrieval using **MySQL**, optimizing database queries to support sorting and filtering.

**Challenges Faced:**

* The difficulty you faced when pulling data or posts to the homepage, specifically showing content from users the current user is following and displaying counts of likes, shares, and comments from multiple tables, can be a common challenge in applications with complex data relationships

**Solution:**

* To solve the problem of pulling posts from users that the current user follows, while also aggregating likes, shares, and comments from multiple tables, we used a combination of SQL joins and unions.

**Outcome & Results**

* The application successfully fostered a stronger connection between staff and students by providing a unified platform for academic and extracurricular interaction.
* Faculty members actively shared updates, and students engaged with these posts, creating a dynamic learning environment.
* The "request a callback" feature streamlined student queries, improving communication with administrative departments.
* Positive feedback from students and staff on the app's ease of use and ability to host meaningful academic discussions.

**Learning Experience**

* Gained hands-on experience with **Spring Boot** and **Spring Security** for user management and authentication, enhancing your expertise in securing web applications.
* Worked extensively with **MySQL**, improving your ability to design and optimize database queries for better performance.
* Tackled the challenge of implementing **thread-based discussions** that required efficient data storage and retrieval for smooth real-time interactions.

Top of Form

Bottom of Form