

Infosys Springboard Internship - Week 2 Report

Name: Jyothi Sunkara

GitHub Repository: https://github.com/JyothiSunkara/INFOSYS_INTERNSHIP

Week 2 Objectives:

1. Build a basic layout with Tailwind CSS.
2. Display the user dashboard after login.
3. Set up department-wise scoping in the backend.

Tasks Completed:

1. Layout with Tailwind CSS

- Created a responsive dashboard layout using Tailwind CSS.
- Sidebar navigation with tabs: Home, People, Achievements, Shoutouts, Analytics, Settings.
- Top navigation bar with search, profile avatar, and logout button.
- Dark/Light theme toggle implemented using local storage.

2. User Dashboard after Login

- Dashboard accessible only after successful login (protected route).
- Displayed user details: username, email, department, role (Admin/User).
- Editable profile modal to update username, email, and department.
- Users can see their initials on the profile avatar and click to edit.

3. Department-Wise Scoping

- Backend endpoint `/user/dashboard` created to fetch users.
- Admin users can view all users across departments.

- Regular users can view only users from their department.
- Implemented Axios API integration with access and refresh token handling.

Key Features Implemented:

- Sidebar navigation with active tab highlighting.
- Responsive top navbar with search bar.
- Profile editor modal to update user information.
- Axios interceptors handle **token expiration**.
- Protected and public routes handled via `authService.isAuthenticated()`.

What I Learned:

- How to use **Tailwind CSS** to build responsive and visually appealing layouts.
- How to **protect routes** in React using `react-router-dom` and authentication checks.
- How to implement **department-wise data filtering** from backend APIs.
- Handling **state and forms** in React for login, registration, and profile editing.
- Managing **authentication tokens** securely in frontend.
- How to **persist errors** and handle asynchronous API calls in React.

Challenges & Solutions:

- **Admin/User role mismatch:** Solved by converting role dropdown to `is_admin` boolean during registration.
- **Department filtering:** Used SQLAlchemy with `func.lower` and `trim` to handle case-insensitive filtering.

- **Routing after login:** Updated React routes to redirect based on authentication.

Outcomes:

- Functional, responsive dashboard using Tailwind CSS.
- Dashboard shows user info and department-specific users.
- Error messages displayed until successful actions.
- Protected routes ensure only authenticated users can access the dashboard.
- Foundation ready for additional features: Achievements, Shoutouts, Analytics, Settings.

Status: COMPLETED SUCCESSFULLY

Date: October 09, 2025