**Assignment 10: Admin & Employee Portal with Redux**

In this assignment, you’ll enhance your existing app by implementing **admin-specific** and **employee-specific** functionalities using **Redux** for state management and securing access with role-based routing. Up to this point, component communication has primarily been achieved through states and props. In this assignment, you will have the opportunity to manage state using Redux.

**Backend Enhancements**

1. **Changes in POST /user/create Endpoint:**

* Add a new field: type with values only "admin" or "employee" (required).
* Ensure invalid values are rejected.

1. **GET /users Endpoint:**

* Make sure it fetches **all user details** except passwords.

**Admin Portal Functionality**

1. **Routing:**

* Admins should only access **admin-specific** pages.
* Changes in Login API and Routing

The routing will be restricted based on the type of user. For example, when an admin logs in, they will only have access to the admin page. Conversely, if an employee logs in, they will only be able to view the pages developed in the previous assignment.

1. **Employees Page for Admins:**

* Create an **Admin-only** page that displays all users (name, email, type) in a table.

1. **Add Jobs Page:**

* Route: /add-job
* Accessible only by admins.
* Form fields: Company Name, Job Title, Description and Salary.
* On submit, call API: POST /create/job.

**Employee Portal Functionality**

1. **Routing:**

* Employees should only access **employee-specific** pages.

1. **Jobs Page:**

* Route: /jobs
* Accessible only by employees.
* Fetch jobs from GET /jobs API.
* Display job listings in cards or a table format.

**Additional (Optional)**

* Add **loading spinners** during API calls for better UX.
* Add **pagination** for job listings.

**Key Requirements**

* **Login Required** for all routes.
* Use **Redux** for all state management.
* Use **Material UI** or **React Bootstrap** for components (e.g., navbar, cards, forms).

**Useful References**

1. **Routing:**

* [Hygraph Blog](https://hygraph.com/blog/routing-in-react)
* [YouTube Guide](https://www.youtube.com/watch?v=oUZjO00NkhY)
* [StackOverflow Discussion](https://stackoverflow.com/questions/69851678/correct-way-to-redirect-user-to-different-routes-based-on-authentication-in-reac)

**UI Libraries:**

* [Material UI](https://mui.com/material-ui/all-components/)
* [React Bootstrap](https://react-bootstrap.netlify.app/)

**Redux:**

* [Redux Quick Start](https://react-redux.js.org/tutorials/quick-start)

**Submission Guidelines**

**1. GitHub Repository**

* Push the complete Assignment 10 (both frontend and backend) to a **private repository** on your GitHub account.
* Use your **Northeastern email account** to ensure proper access for evaluation.

**2. Canvas Submission**

* Submit a **ZIP file** containing the entire project (frontend and backend).
* Include your **GitHub repository URL** in the Canvas **submission comments**.

**3. Important Notes**

* **Timestamps** on both **Canvas** and **GitHub** will be considered.  
  Late submission on either platform will result in a grade reduction.
* **No additional commits** are allowed to the GitHub repository after the Canvas submission.  
  Any changes made post-submission may lead to disqualification.