React Crash Course for Backend Developers

1. React Basics - Components & Props

React components are functions that return UI using JSX (JavaScript XML).

Example Component:

```
function Welcome(props) {
  return <h1>Hello, {props.name}</h1>;
}
```

JSX looks like HTML but is written in JavaScript.

Props are values passed to components:

```
<Welcome name="Admin" />
```

2. useState & useEffect

useState allows local state management inside a function component.

Example:

```
const [count, setCount] = useState(0);
```

useEffect is used to run side effects like API calls:

```
useEffect(() => {
  console.log("Component loaded");
}, []);
```

3. Making API Calls

API calls can be made using fetch or axios.

fetch Example:

```
fetch("/api/users").then(res => res.json()).then(data => console.log(data));
```

axios Example:

```
axios.get("/api/complaints").then(res => console.log(res.data));
```

React Crash Course for Backend Developers

Backend should return JSON and handle CORS.

4. Routing with react-router-dom

Install react-router-dom:

npm install react-router-dom

Define routes using <BrowserRouter> and <Routes>:

<Route path="/admin" element={<AdminDashboard />} />

5. Authentication Flow

JWT is stored in localStorage/sessionStorage on the frontend.

Login Example:

```
localStorage.setItem("token", res.data.token);
navigate("/admin");
```

Protect routes using token checks before rendering pages.

6. Example API Calls from React to Backend

POST Example (Axios):

```
axios.post("http://localhost:3000/api/complaints", {
  title: "Pothole",
  description: "Huge pothole in sector 9"
}, {
  headers: { Authorization: `Bearer ${token}` }
});
```

GET Example:

```
axios.get("http://localhost:3000/api/users", {
  headers: { Authorization: `Bearer ${token}` }
});
```

React Crash Course for Backend Developers

Your backend must validate JWT, parse req.body, and return a JSON response.

7. Role-Based Dashboard Structure (Folder & Component Design)

Suggested Frontend Folder Layout:
```
/src
  /components
    /AdminDashboard
    /CandidateDashboard
    /UserDashboard
  /pages
    LoginPage.jsx
    PublicHighlights.jsx
  /routes
    PrivateRoute.jsx
```

Example Route Logic:
```
if (user.role === 'admin') navigate('/admin');
else if (user.role === 'candidate') navigate('/candidate');
else navigate('/user');
```

Each dashboard can fetch data using:
```
useEffect(() => {
  axios.get(`/api/complaints?region=${user.region}`);
}, []);
```