# Week 7 deliverables

1. **User Model and Authentication API Endpoints**:
   * You’ve set up the backend using **Express.js** and **MongoDB**, and defined the **User model** with fields such as name, email, password, and role (Job Seeker/Job Poster).
   * Implemented **authentication endpoints**:
     + /signup for user registration.
     + /signin for user login.
     + Middleware to **hash passwords** using bcrypt.
2. **Frontend Login and Registration Forms**:
   * Frontend forms for both **sign-up** and **sign-in** are created.
   * Added the option for users to select **Job Seeker** or **Job Poster** during sign-up and sign-in, with corresponding form variations.
   * These forms are connected to the backend via **API requests** using **Axios**.
3. **JWT Implementation for Secure Authentication**:
   * You’ve started **JWT (JSON Web Token)** integration:
     + On successful login, a **JWT is generated** and sent to the user.
     + JWT is stored on the frontend (likely in local storage or a cookie).
     + The backend is set up to verify JWT on protected routes using middleware.
4. **Protected Routes on Frontend and Backend**:
   * You are securing **protected routes**:
     + Backend routes, like /jobs, are restricted to authenticated users.
     + Frontend routes (like viewing job postings or profiles) will only be accessible to authenticated users after login, using JWT for access.

* **HTTPS** is set up for secure communication.
* Completed the **JWT authentication flow** with the frontend and backend fully integrated.