

JobSensei – Full Project Documentation


MODULE 1: Authentication & Role-Based Access


 Objective: Let users sign in via Google, GitHub, LinkedIn, assign them a role (user, mentor, admin), and generate a secure JWT.

 Key Steps:

- Set up Express backend server.
- Connect to MongoDB for storing user profiles.
- Configure Passport.js strategies (Google, GitHub, LinkedIn).
- Store and retrieve user sessions with roles.
- Generate a JWT token after successful OAuth.
- Frontend: Use Vite + React to show login buttons.
- Save token in localStorage and redirect user to dashboard.
- Create route guards based on role using JWT.


MODULE 2: Profile Management System

 Objective: Allow users to create and update a detailed profile, including personal info, resume upload, skills, education, experience.

 Key Steps:

- Build profile form UI for each role.
- On backend, create/update profile endpoints.
- Allow resume upload (PDF) using Multer.
- Save resume URL + structured info (skills, education).
- Show profile completion percentage on dashboard.
- Add option to delete account or resume.

MODULE 3: Personalized Dashboards


 Objective: Role-based dashboards showing relevant data: User: job matches, profile tips. Mentor: mentee requests. Admin: stats, user management.


 Key Steps:

- Detect user role via JWT on frontend.
- Render dynamic dashboard components.
- Create backend APIs for analytics (user count, activity).

- • Add profile completion meter.
- • Integrate career progress timeline UI.

MODULE 4: Smart Job Recommendations (AI-Powered)


 Objective: Use AI to recommend jobs based on user profile, skills, and preferences.

 Key Steps:

- • Build job data model (title, skills, company, etc).
- • Scrape or seed jobs (internally or via APIs).
- • Extract user skills and interests from profile.
- • Use basic AI model to match jobs.
- • Show top 5–10 personalized jobs on dashboard.


MODULE 5: Mentorship Matching System


 Objective: Users can request mentorship from experienced mentors.

 Key Steps:

- • Build mentor profiles with areas of expertise.
- • Allow users to search mentors based on field/skills.
- • Request mentorship via form (with goal/interest).
- • Mentors can accept/reject requests.
- • Store mentorship history, and status (pending/active).


MODULE 6: 1-on-1 Session Booking

 Objective: Allow mentees to book calls or chats with mentors.

 Key Steps:

- • Mentors can set availability slots.
- • Mentees can view availability and book session.
- • Confirmation via calendar UI (React calendar).
- • Optional integration with Google Calendar API.
- • Email notification on booking (via NodeMailer).


MODULE 7: AI Career Assistant (Chatbot)

 Objective: Integrate an AI-powered chatbot to answer user career questions.

 Key Steps:

- • Build simple chatbot UI on frontend.
- • Connect to backend route handling prompt.
- • Use OpenAI API for responses.
- • Add context (e.g., user's profile data) in prompt.
- • Log chat history per user in database.

MODULE 8: Resume & Skill Analyzer (AI)


 Objective: Allow users to upload their resume and get AI feedback.

 Key Steps:

- • Upload PDF → Extract text (pdf-parse or similar).
- • Use OpenAI to analyze structure, relevance, tone.
- • Display strengths, improvements, missing keywords.
- • Save analysis results in user profile.


MODULE 9: Learning Hub & Resources


 Objective: Curate and recommend learning materials based on user goals.

 Key Steps:

- • Create a resource model (topic, link, level).
- • Allow admins/mentors to upload content.
- • Match resources based on skills & career goals.
- • Track what users view and save for later.
- • Gamify learning with badges/progress.

MODULE 10: Admin Panel & Analytics

 Objective: Enable admins to manage users, mentors, sessions, and view platform analytics.

 Key Steps:

- • Dashboard: user count, active sessions, role-wise stats.
- • CRUD for all users (view, delete, change roles).
- • Track engagement and conversion metrics.
- • Export reports (CSV, PDF).
- • Manage flagged content or abuse reports.


MODULE 11: Notifications & Messaging

 Objective: Keep users informed via in-app alerts and optional email notifications.

 Key Steps:

- • Design a notification model (type, user, status).
- • Create API to send and retrieve notifications.
- • Trigger alerts on bookings, messages, etc.
- • Show bell icon + unread badge in navbar.
- • Optional: push notifications or email integration.






MODULE 12: Community & Q&A Forum (Optional)

 Objective: Foster a collaborative environment through questions, answers, and discussions.

 Key Steps:

- • Question model: tags, author, upvotes.
- • Answer model linked to question + author.
- • Sorting by recency or popularity.
- • Add voting & comment system.
- • Moderate content with admin tools.

Final Wrap-Up Features

- •  Testing: Unit + integration tests (Jest, Postman)
- •  Security: JWT, HTTPS, role guards, rate limits
- •  Deployment: Vercel (frontend), Render or Railway (backend)
- •  Responsive UI for mobile/tablet
- •  Documentation (README + API docs)