

Take-Home Assignment: Job Listing Page (Frontend + Backend)

Recreate the given **Job Listings Page** with filtering and search functionality using modern full-stack technologies. Only this screen needs to be built.

Frontend Requirements

Build the UI exactly as shown in the provided link , focusing on responsiveness, accessibility, and user experience.

<https://www.figma.com/design/7QhBVha6GNEqRGP9anvTs1/Web-task?node-id=0-1&p=f&t=YOLAAIaRH2FgYjFR-0>

Must Include:

- **Search Bar:** Search by job title and location
- **Filter Sidebar:**
 - Location filters (Remote, Near me, etc.)
 - Salary (Hourly, Monthly, Yearly)
 - Date of posting
 - Work experience
 - Type of employment (multiple selection)
- **Job Cards:**
 - Company logo, job title, job type, location, salary, time posted
 - Pagination at the bottom
- **Side Widgets:**
 - "Email me for jobs" subscription form
 - "Upload your resume" call-to-action

Recommended Tech Stack:

- React + TypeScript
- Tailwind CSS

Backend Requirements

Must Build APIs:

- `GET /jobs`
 - Returns paginated job listings
 - Supports search by title and location (`?title=` , `?location=`)
 - Supports filters like salary, date posted, work experience, and type of employment
- `POST /subscribe`
 - Accepts and stores email subscriptions
- `POST /upload-resume`
 - Accepts and stores a file upload

Recommended Tech Stack:

- NestJS or Node.js + Express
- PostgreSQL
- ORM: Prisma, TypeORM
- File handling: Multer

Evaluation Criteria

- **Frontend UI:** Pixel-perfect, responsive layout
- **Filter Logic:** Filters applied correctly to job list
- **Search Function:** Efficient debounced searching
- **Backend API:** Pagination, search, filtering, clean structure
- **Code Quality:** Clean, reusable components and services
- **Documentation:** Setup steps in README, with `.env.example`

Submission

- GitHub repo link with README and setup instructions
- Postman collection or API documentation (if any)