

# Design Document

## 1. Project Description

**Job-Track** is a two-part platform designed for job seekers to streamline their job search and interview preparation. The **private side** allows users to track every job they apply to, including details such as company, role, date, and status. The **public side** serves as a community-driven hub where users can anonymously share interview questions they've encountered. Together, Job-Track acts as both a **personal application tracker** and a **shared knowledge base** for job seekers.

## 2. User Personas

### Persona 1: The Graduate

**Name:** Sara

**Goal:** Keep track of multiple internship applications.

**Frustrations:** Often loses track of deadlines, statuses, and follow-ups.

**How JobTrack Helps:**

Sara can add each internship she applies to in her personal dashboard. JobTrack automatically organizes all her applications and lets her update progress (e.g., *Applied* → *Interview* → *Offer*) in one place.

### Persona 2: The Job Hopper

**Name:** Alex

**Goal:** Prepare efficiently for technical interviews.

**Frustrations:** Never remembers which companies asked what kind of questions.

**How Job-Track Helps:**

Alex can browse community posts tagged by company and role to prepare for interviews.

### 3. User Stories

As a...	I want to...	So that I can...
Job Seeker	Register and log in securely	Maintain my own job applications privately
Job Seeker	Add, view, edit, or delete job applications	Track my application progress efficiently
Job Seeker	Filter or search my applications by status, role, or company	Quickly locate any job I applied to
Community User	Anonymously post interview questions	Share knowledge with other job seekers
Community User	View and search questions by company or role	Prepare for upcoming interviews
User	Log out	Secure my account on shared devices

### 4. System Architecture

#### Frontend:

- HTML5, CSS3, and Vanilla JavaScript
- Dynamic rendering of both private dashboard and public hub using fetch() calls to RESTful endpoints

#### Backend:

- **Node.js + Express.js** handles routes, authentication, and CRUD operations
- **JWT Authentication** ensures secure session management

#### Database:

- **MongoDB** (Collections: users, applications, questions)
  - users: stores account credentials and profile info
  - applications: stores each user's personal job applications
  - questions: stores public interview questions

## 5. Structure

vbnet

```
src/
├── components/
│   ├── ApplicationTracker.jsx
│   ├── InterviewHub.jsx
│   ├── QuestionList.jsx
│   ├── AddExperienceModal.jsx
│   ├── CompanySearch.jsx
│   ├── Login.jsx
│   ├── Register.jsx
│   ├── Profile.jsx
│   └── Shared/
├── css/
│   ├── InterviewHub.css
│   ├── QuestionList.css
│   ├── ApplicationTracker.css
│   └── Shared.css
├── App.jsx
└── index.jsx
```

```
server/
├── routes/
│   ├── authRoutes.js
│   ├── questionsRoutes.js
│   ├── applicationRoutes.js
│   └── companyRoutes.js
├── controllers/
│   ├── authController.js
│   ├── applicationsController.js
│   └── questionsController.js
├── middleware/
│   └── authMiddleware.js
├── config/
│   └── db.js
└── server.js
```

## 6. Technical Design

### Authentication

- Secure user registration and login using **JWT tokens**
- Tokens stored in **localStorage**
- Middleware validates tokens for protected routes (e.g., /api/applications)

### Private Dashboard (Applications)

- CRUD operations for the applications collection
- Each entry contains: company name, role, application date, and status
- Users can update progress (Applied → Interview → Offer → Rejected)
- Filter applications by company or status

### Public Interview Hub

- CRUD operations for questions collection
- Fields: company, role, questionTitle, questionDescription, difficulty, tips, and optional AI-generated answer
- Search and filter functionality on company, role, or keywords
- AI integration via API to provide contextual mock answers

## 7. CRUD Operations

Operation	Description
Create	Register user / Add new job application / Add interview question
Read	Fetch user profile / Get list of user's applications / Get public questions
Update	Edit application status / Update question details
Delete	Remove job applications / Delete posted questions

## 8. API Routes

Endpoint	Method	Description
/api/auth/register	POST	Create new user
/api/auth/login	POST	Log in & return JWT
/api/auth/profile	GET	Fetch logged-in user info
/api/applications	POST	Add new job application
/api/applications	GET	Retrieve all applications for user
/api/applications/:id	PUT	Update specific application
/api/applications/:id	DELETE	Delete specific application
/api/questions	POST	Create new interview question (public)
/api/questions	GET	Retrieve all public interview questions
/api/questions/:id	PUT	Update an interview question
/api/questions/:id	DELETE	Delete a public question
/api/questions/search	GET	Filter questions by company, role, or keyword

## 9. Team Responsibilities

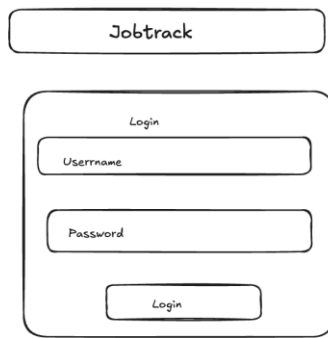
Member	Responsibilities
Zoya	Full CRUD for public questions; design of public interview hub; search and filter features
Arvind	Authentication & profile management; CRUD for private applications; dashboard interface & status tracking

## 10. Design Mock-ups

### Login Page

A minimalistic login form cantered on the screen. Users enter email and password to access their dashboard.

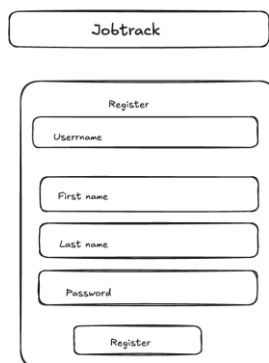
Includes a link to registration.



The image shows a login form for 'Jobtrack'. It consists of a header box with the text 'Jobtrack'. Below it is a rounded rectangle containing the title 'Login'. Inside this rounded rectangle are two input fields: 'Username' and 'Password'. At the bottom of the rounded rectangle is a button labeled 'Login'.

## Registration Page

Allows new users to sign up with name, email, and password. On successful registration, users are redirected to the dashboard.

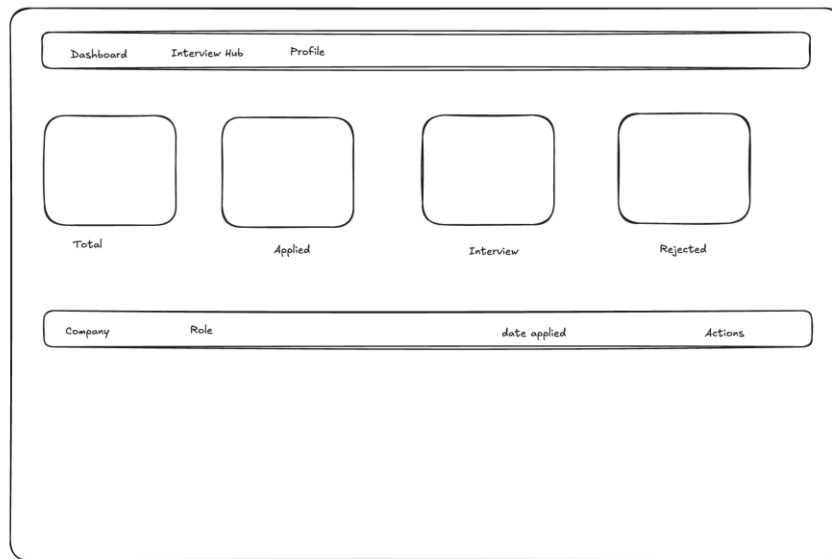


The image shows a registration form for 'Jobtrack'. It consists of a header box with the text 'Jobtrack'. Below it is a rounded rectangle containing the title 'Register'. Inside this rounded rectangle are four input fields: 'Username', 'First name', 'Last name', and 'Password'. At the bottom of the rounded rectangle is a button labeled 'Register'.

## Dashboard (Private)

Displays all job applications in a table or card layout.

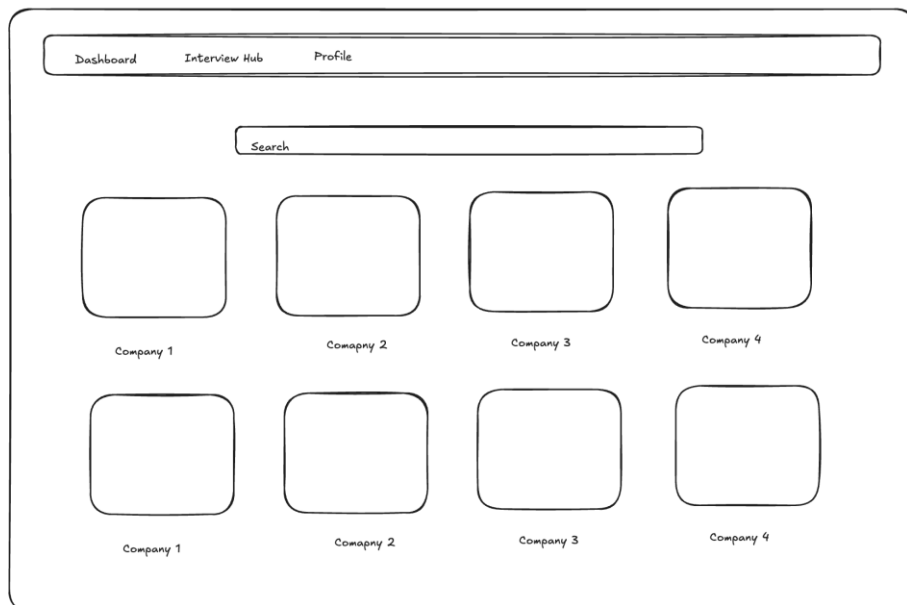
Each row shows **Company | Role | Date | Status**, with options to **edit** or **delete** entries. Filter dropdowns for status and company.



## Public Interview Hub

A searchable list of community interview questions.

Users can post new questions anonymously



## Add Experience Form

<div>Company</div> <div></div> <div>Role</div> <div></div> <div>Difficulty</div> <div></div> <div>Tips</div> <div></div>	<div>Question</div> <div></div>
--	---------------------------------

## 11. Future Enhancements

- Add notifications for upcoming interview dates
- Allow upvoting of public interview questions
- Include resume keyword suggestions based on tracked applications
- Add email integration for status reminders