

**PC649 : Summer Internship**

**Jobify**

**Id : 202312079**

**Name : Hetvi Shah**

**Mentor : Astha Mehta**

**Sparks to Ideas**

**Index:**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Topic** | **Page No** |
| 1 | Introduction | 3 |
| 2 | Functional Requirements | 4 |
| 3 | Non-Functional Requirements | 5 |
| 4 | Methodologies | 6 |
| 5 | Design | 8 |
| 6 | Coding | 14 |
| 7 | Snapshots | 15 |
| 8 | Testing | 25 |
| 9 | Summary | 27 |

**Introduction**

* Jobify is a next-generation job portal that is transforming the way employers and job seekers connect.
* By providing a user-friendly platform for employers to post job openings and for job seekers to search and apply for jobs, Jobify is streamlining the hiring process and making it easier than ever for top talent to find the right opportunity.
* With advanced search functionality Jobify is designed to provide a seamless and efficient experience for both employers and job seekers.
* The platform's matching algorithm ensures that job seekers are presented with the most relevant job openings, while employers are able to quickly and easily find the best candidates for their positions.

**Key features:**

* Visitors:
  + The visitor will be able to browse the website and view all the job posts, but will not be able to apply for the post.
  + Once the visitor has browsed the site they will be able to register based on their roles.
* Employers:
  + The employers will be able to view the job posts posted by other employers and will be able the even post a new job opening.
  + The employers will be able to view the job applications of the applicants and then find the most relevant candidate suitable for the jobs.
* Employees:
  + The employees will be able to browse all the job posts provided and will be able to apply for the suitable position.
  + The employees will get the email regarding the most relevant job post based on the niche of the employee.
  + The employees will be able to search for the particular job based on the location, niche and the relevant keywords.
  + The employees will be able to update their details and upload CV.

**Functional Requirements**

* **User Management**
  + User Registration: Job seekers and employers must be able to register on the platform by providing basic information such as name, e-mail, and password.
  + User Profile Management: Users must be able to create and manage their profiles, including uploading resumes, adding skills, and updating personal information.
  + Login and Authentication: Users must be able to log in to the platform using their registered credentials.
* **Job management**
  + Job Posting: Employers must be able to post job openings on the platform, including job title, description, requirements, and other relevant details.
  + Job Search: Job seekers must be able to search for job openings by keyword, location, category, and other relevant criteria.
  + Job Application: Job seekers must be able to apply for job openings, including uploading their resume and cover letter.
* **Matching and Recommendation**
  + Job Matching Algorithm: The platform must have a built-in algorithm that matches job seekers with relevant job openings based on their skills, experience, and preferences.
* **Reporting and Analytics**
  + Job Posting Analytics: The platform must provide employers with analytics and insights on their job postings, applications, and other relevant metrics.
  + Job Application Analytics: The platform must provide the employees the detailed information about all the job posts for which they have applied.

**Non Functional Requirements**

* **Performance**: The website should handle a large number of concurrent users with minimal latency.
* **Security**: Employers and Job seekers authentication should be protected. Sensitive user information should be protected.
* **Usability**: The website should have an intuitive and user-friendly interface for easy navigation and usage. Response times should be fast to enhance the user experience.
* **Scalability**: The website should be able to handle a growing number of users and properties.
* **Reliability:** The website should be stable and reliable, minimizing the chances of downtime or crashes.

**Methodologies**

The Agile Process Model is a software development approach based on iterative and incremental development. The term "Agile" reflects the model's emphasis on adaptability and flexibility. In the Agile process model, the entire project is divided into smaller iterations or timeframes, typically lasting from one to four weeks. Each iteration involves a cross-functional team working collaboratively to deliver a working product increment.

**Phases of Agile Model:**

* Requirements Gathering: Collaborate with stakeholders to gather and prioritize requirements, capturing them in a dynamic product backlog.
* Design the Requirements: Perform design activities just-in-time during each iteration, planning the implementation of identified requirements.
* Construction/Iteration: Core phase involving development, testing, and review of functional increments of the software in short iterations.
* Testing: Integral part of Agile development, ensuring software quality and meeting defined requirements throughout the process.
* Deployment: At the end of each iteration, a potentially shippable product increment is available for deployment or feedback.
* Feedback: Frequent stakeholder and end-user feedback drives continuous improvement and adaptation throughout the development process.

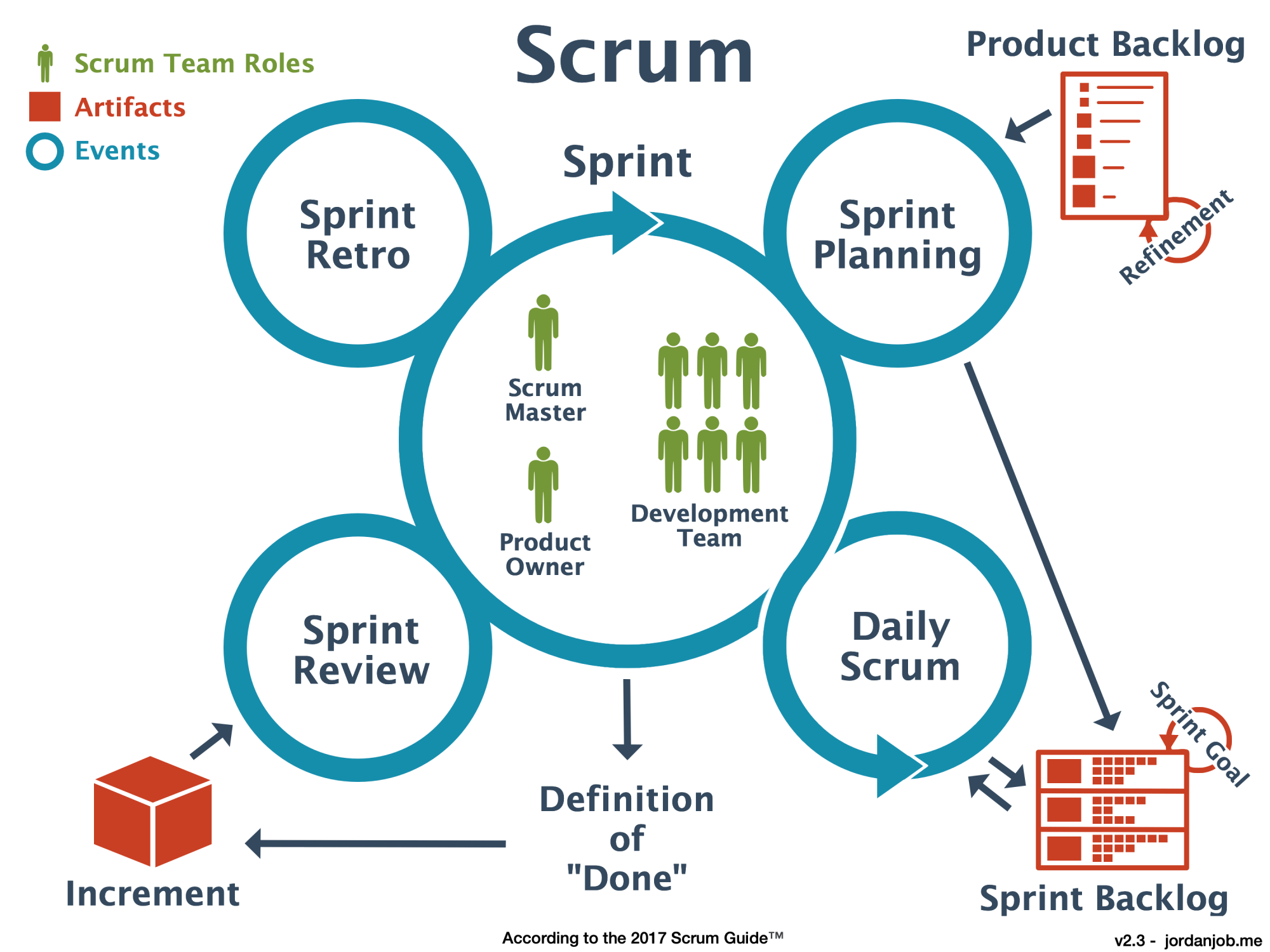
The Agile Process Model enables teams to respond to changing requirements, deliver incremental value to customers, and continuously improve the product. It reduces project risks and accelerates product delivery, making it a popular choice for modern software development projects.

Sample User Stories:

* As a user, I want to sign up with my email and password to access the platform.
* As a user, I want to update my profile information.

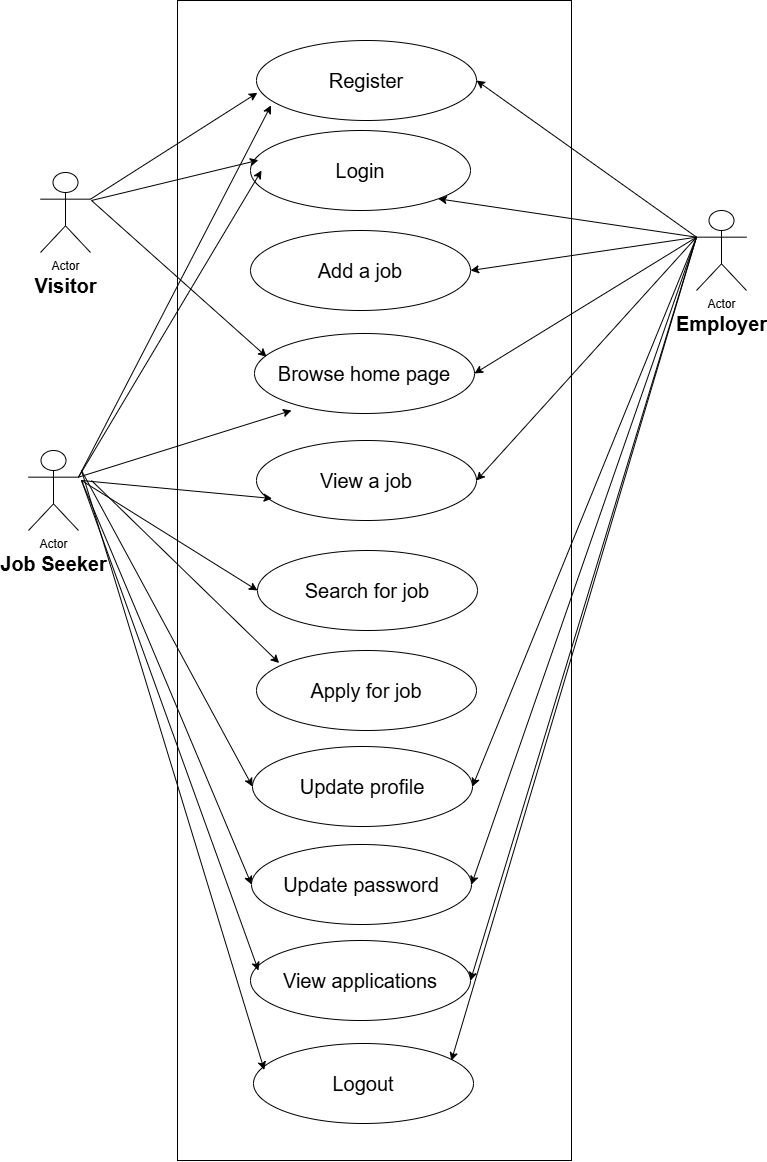
**Due to the following reasons, I chose scrum as Software Development Life Cycle Model:**

* 15-day Sprints: Scrum's fixed-duration iterations enable incremental progress.
* User Stories: Well-suited for expressing requirements in user story format.
* Quick Development: Structured approach for fast-paced development and review.



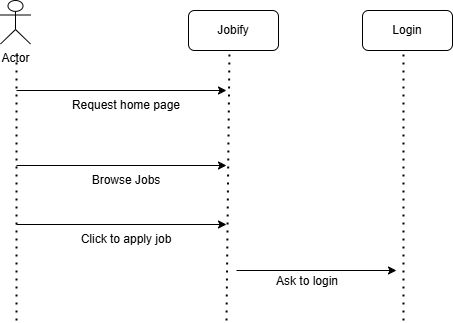
**Diagrams**

**Use Case Diagram:**

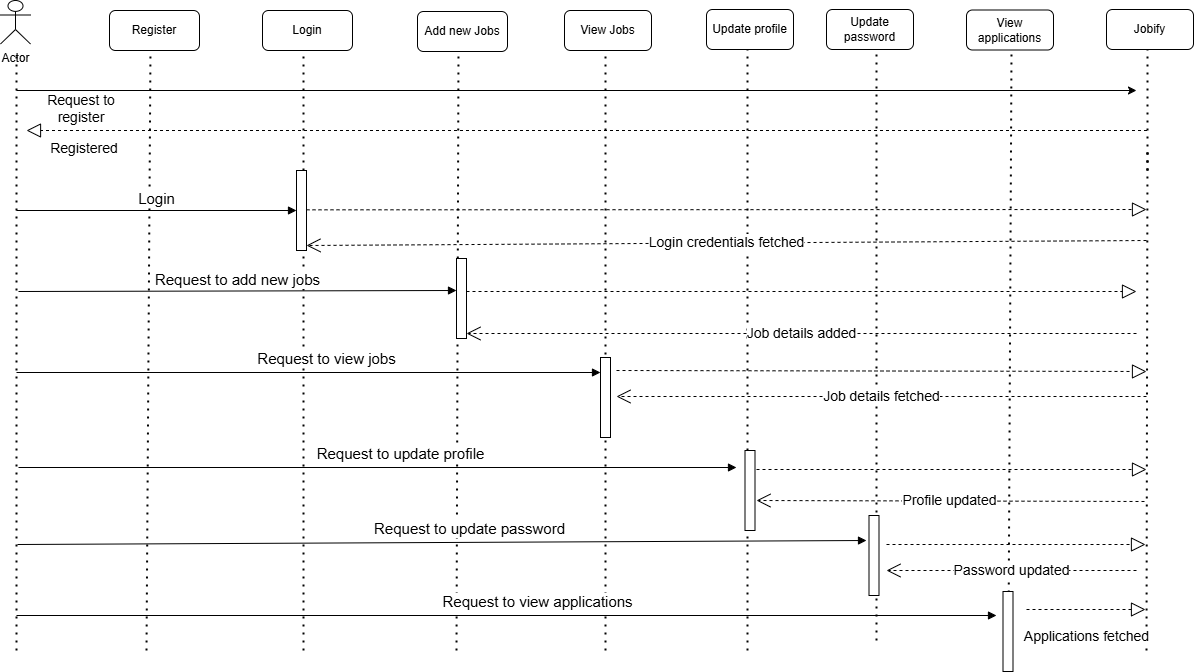
****

**Sequence Diagrams:**

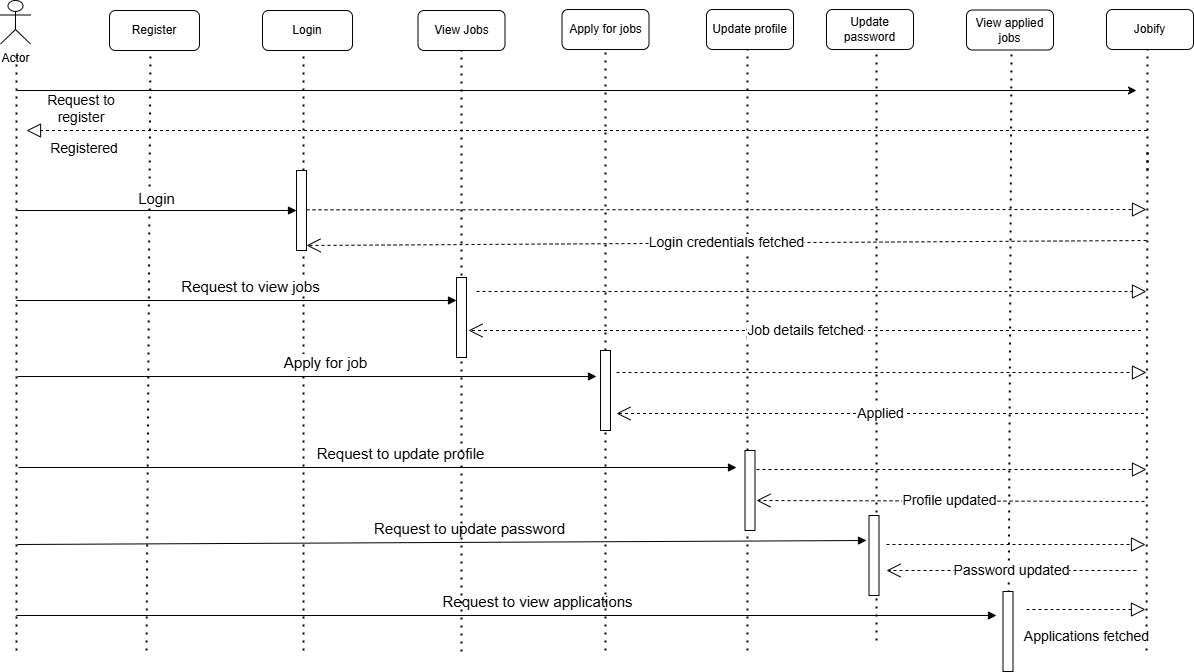
* **Visitor:**



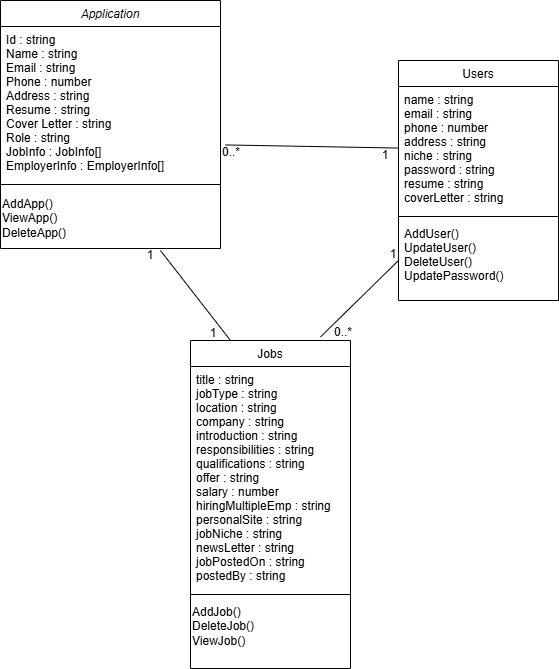
* **Employer:**



* **Job Seeker:**

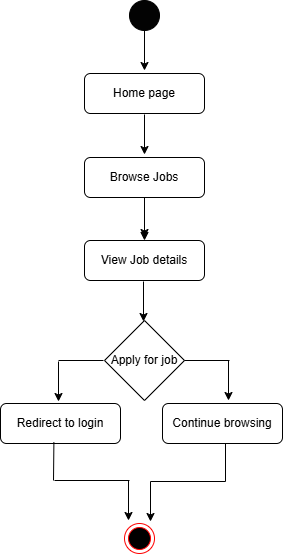
****

**Class Diagram:**

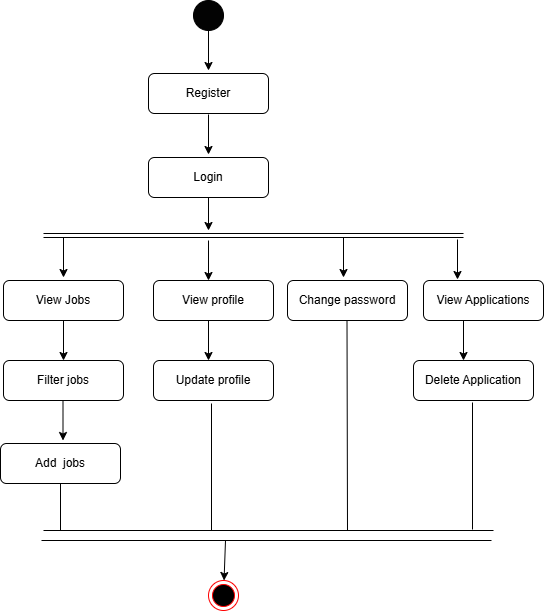
****

**Activity Diagram:**

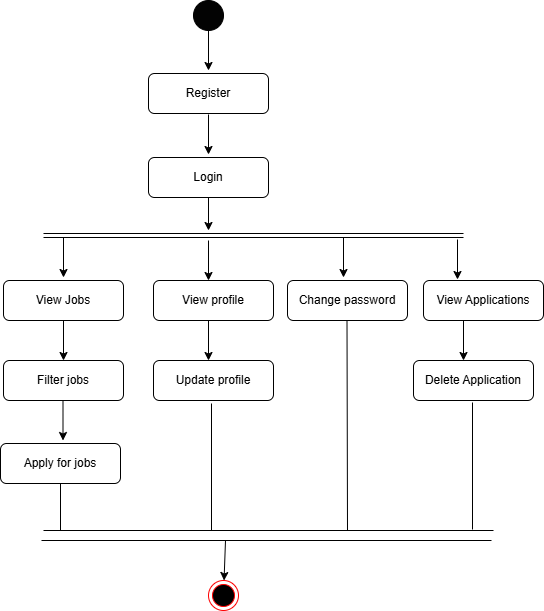
* **Visitor**



* **Employer:**



* **Job Seeker:**

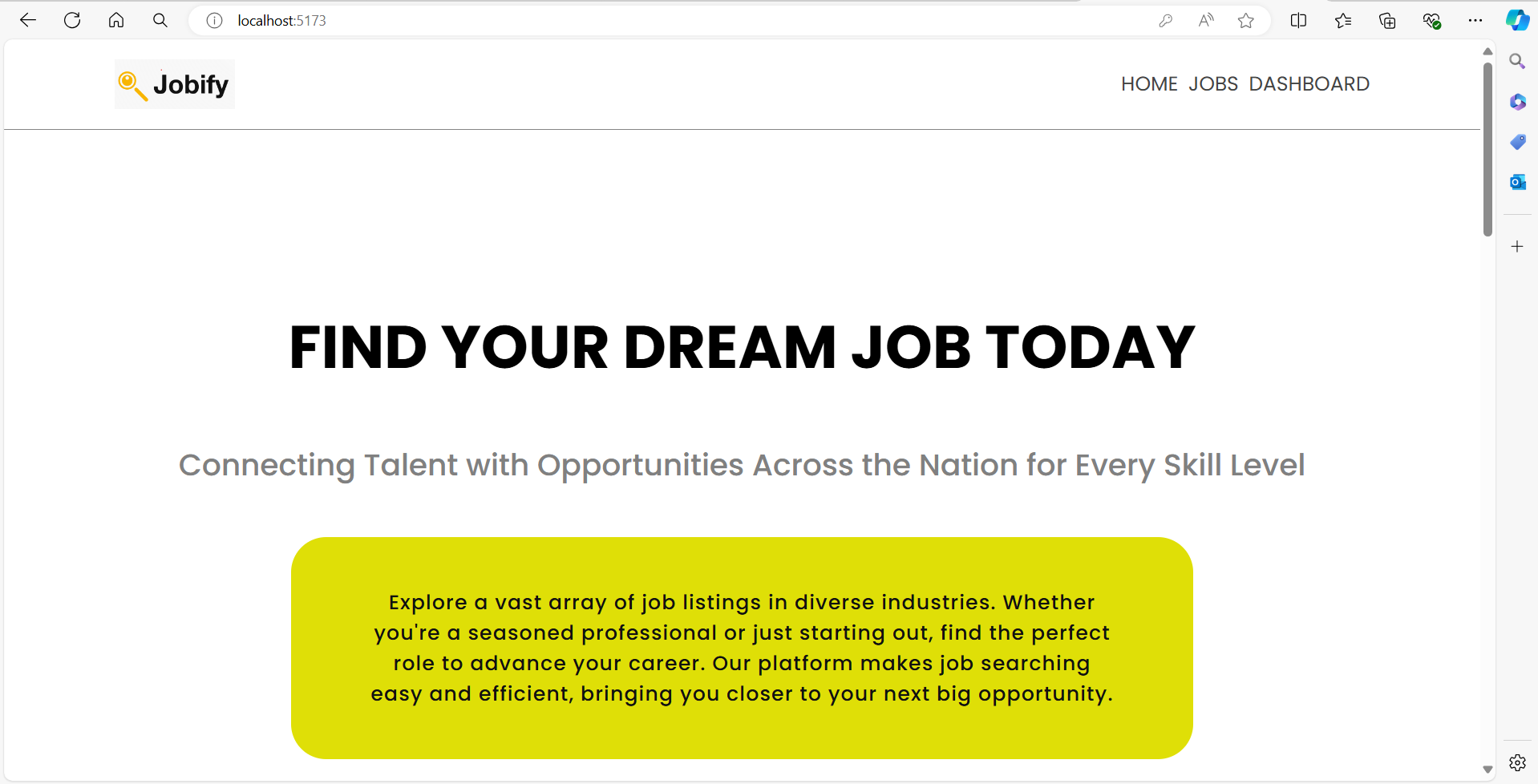


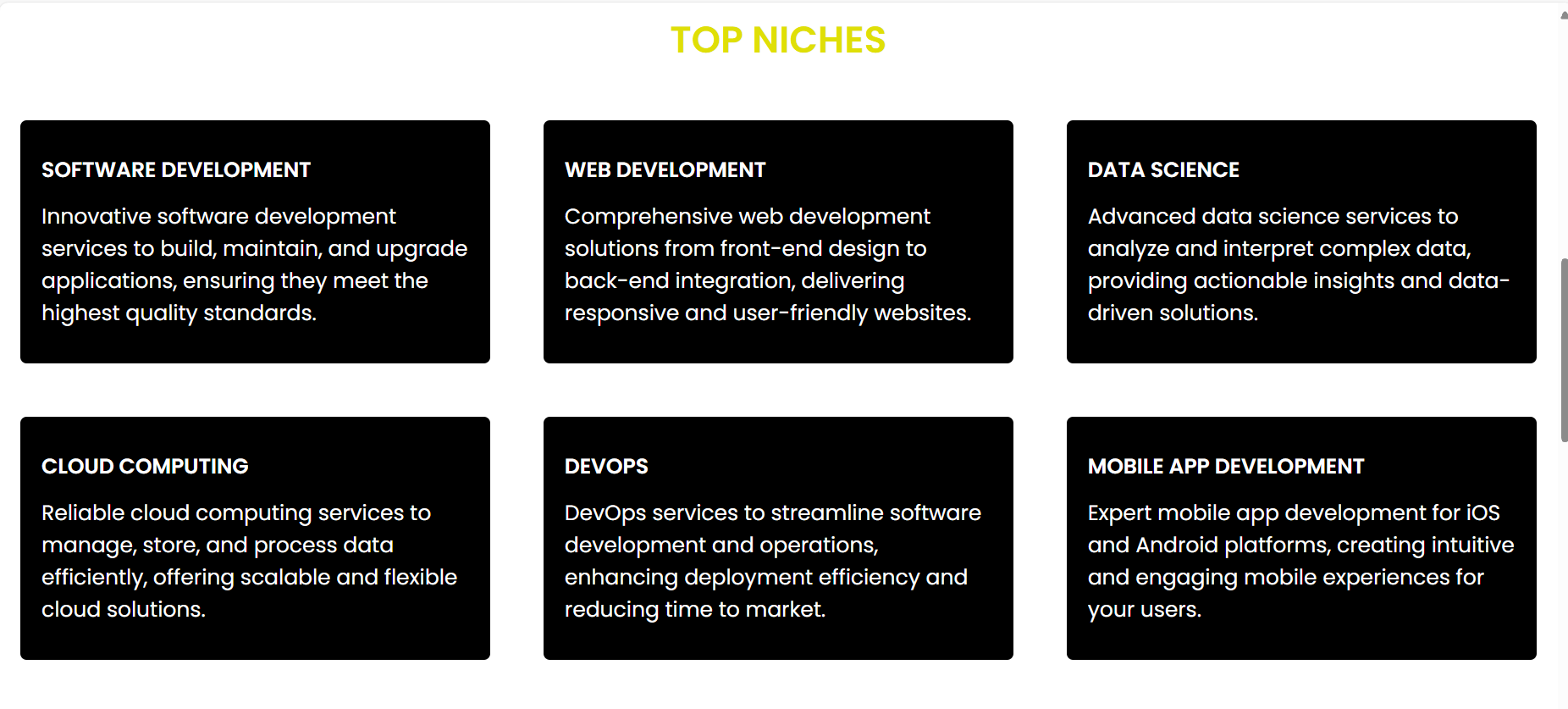
**Coding**

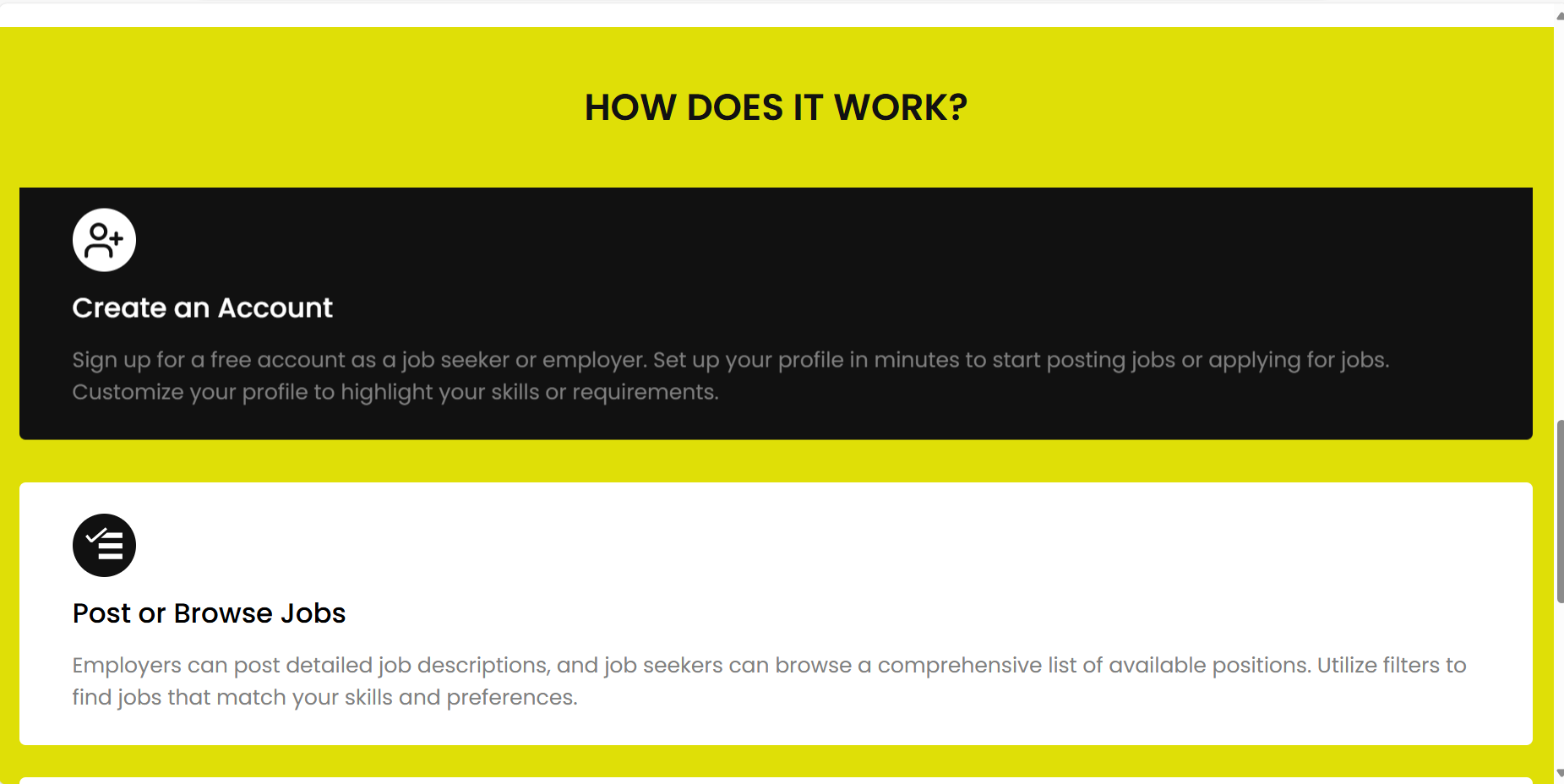
* **FrontEnd:**
* React: Used for building the user interface.
* Vite: A build tool for optimizing and serving the frontend.
* React Router DOM: For handling client-side routing.
* Redux: State management for managing application state.
* React Toastify: For displaying notifications and alerts.
* React Icons: For incorporating icons into the UI.
* Bootstrap: For responsive and styled UI components.
* React Spinner: For loading indicators and spinners.
* **BackEnd:**
* Node.js: JavaScript runtime for server-side development.
* Express.js: Web framework for building the backend API.
* **Database:**
* MongoDB: NoSQL database for storing user, job, and application data.
* **Other tools and services:**
* Bcrypt: For hashing and securing passwords.
* Cloudinary: For storing and managing images.
* Cookie-Parser: For handling cookies in Express.
* Nodemailer: For sending emails (e.g., notifications, password resets).

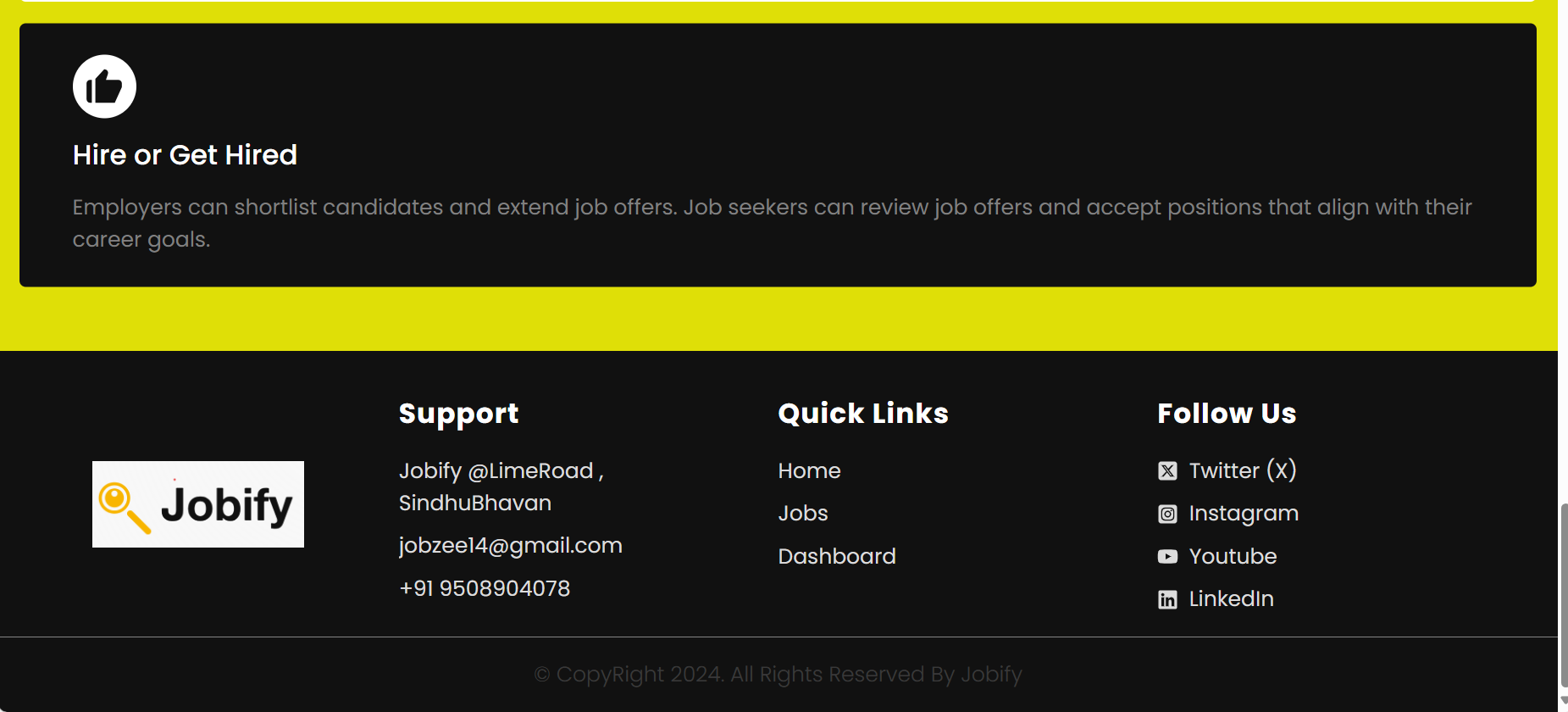
**Snapshots**

Home Page:

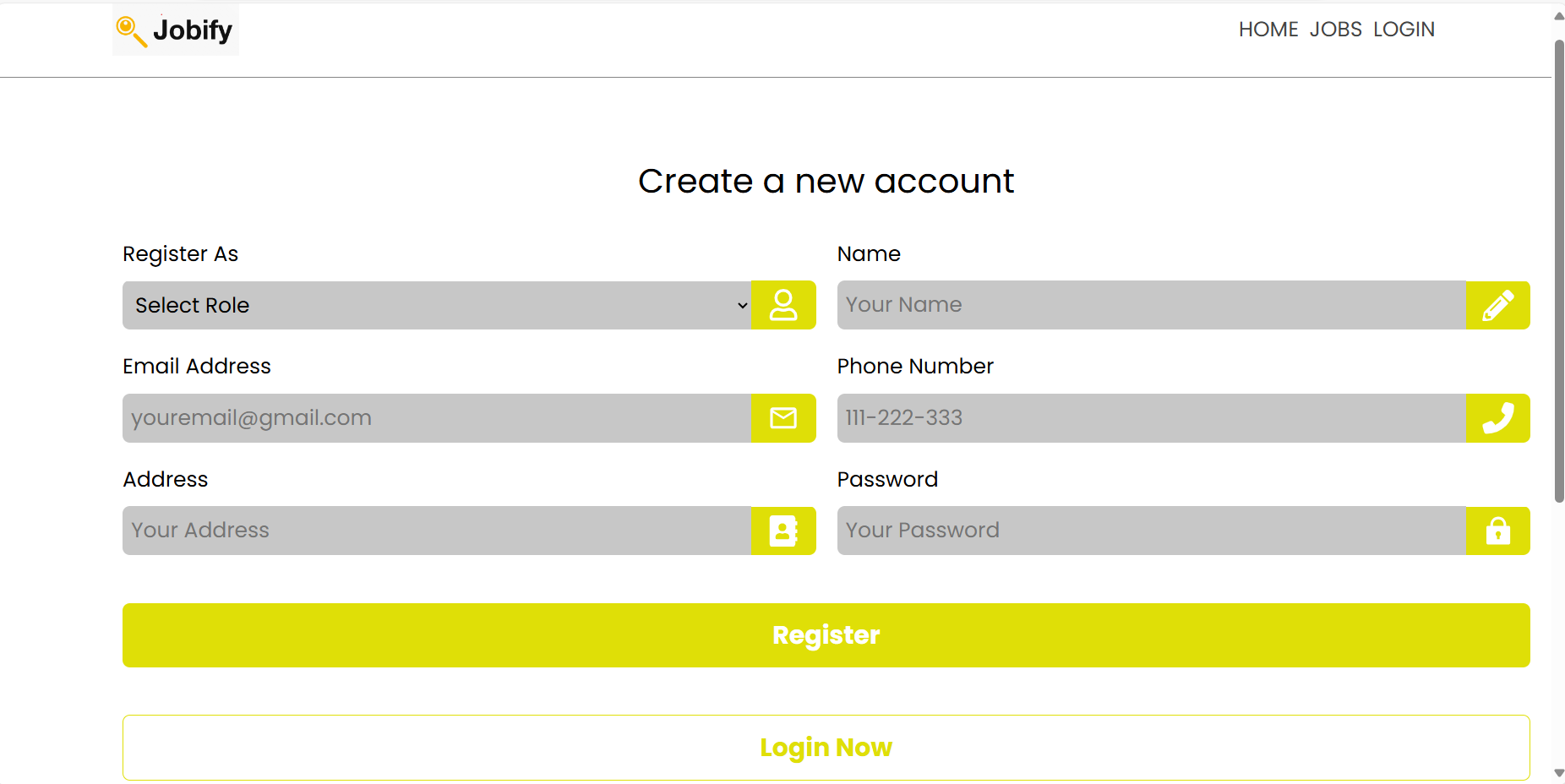




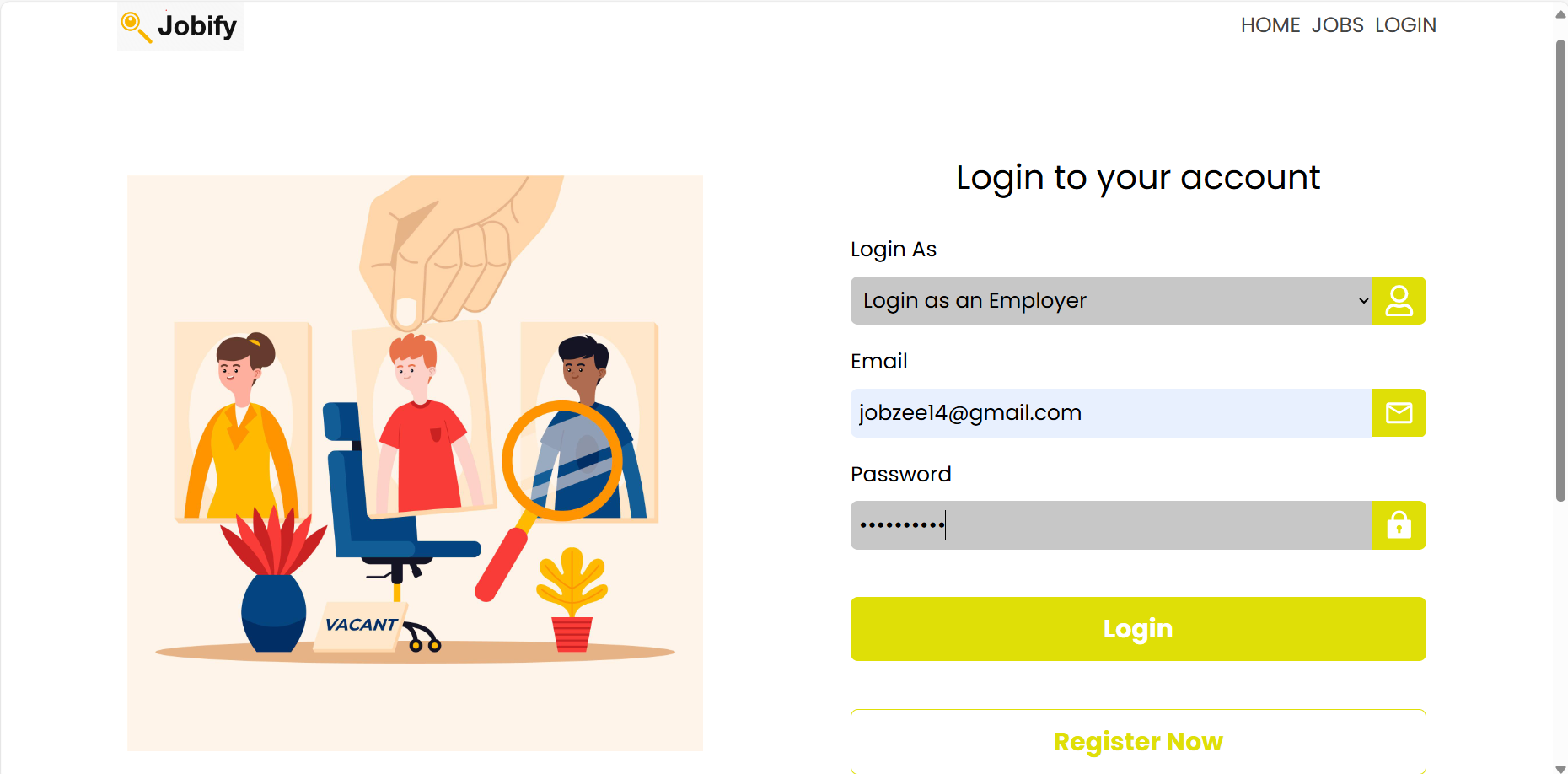




Register as employer:



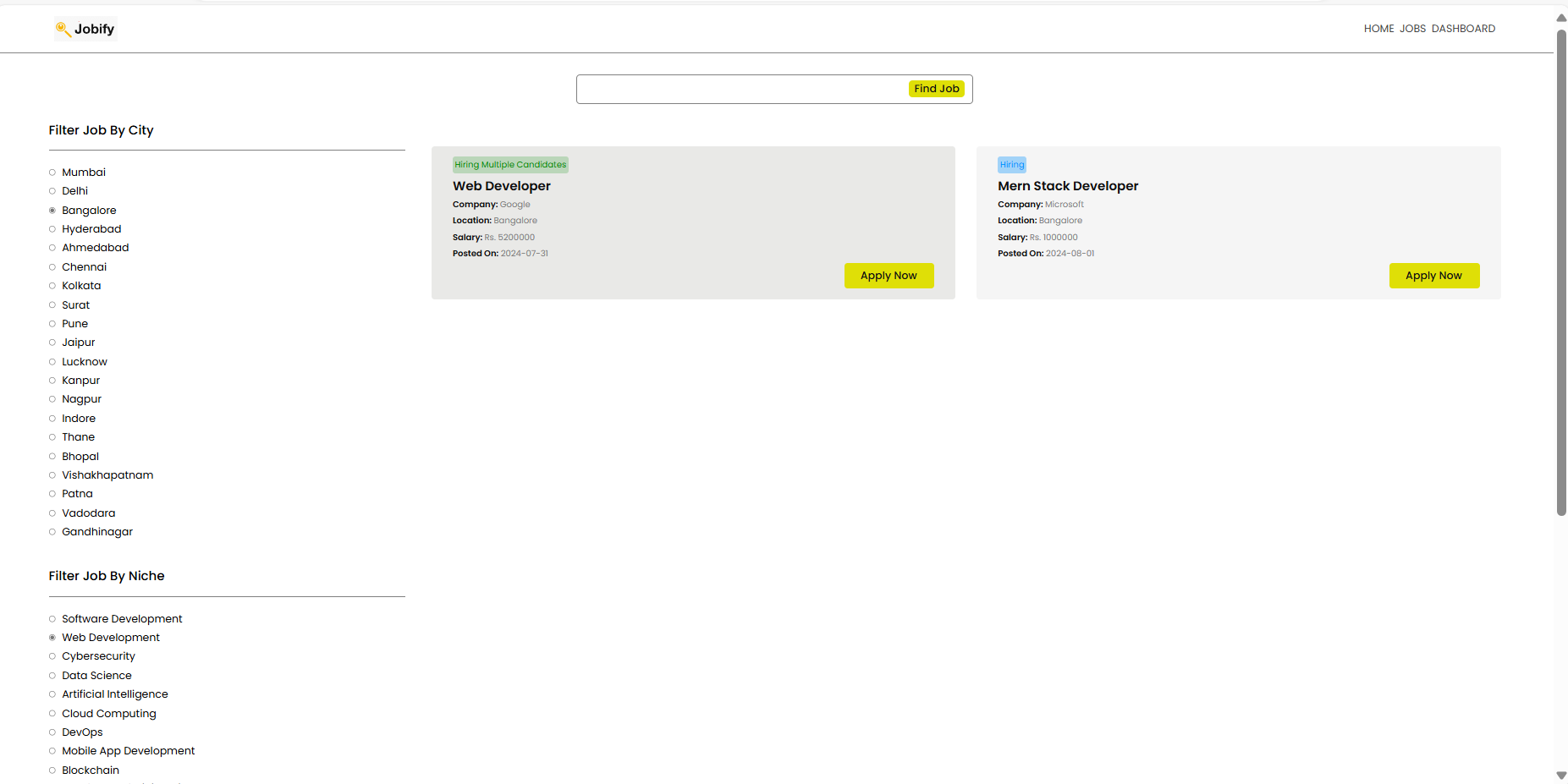
Login as Employer:



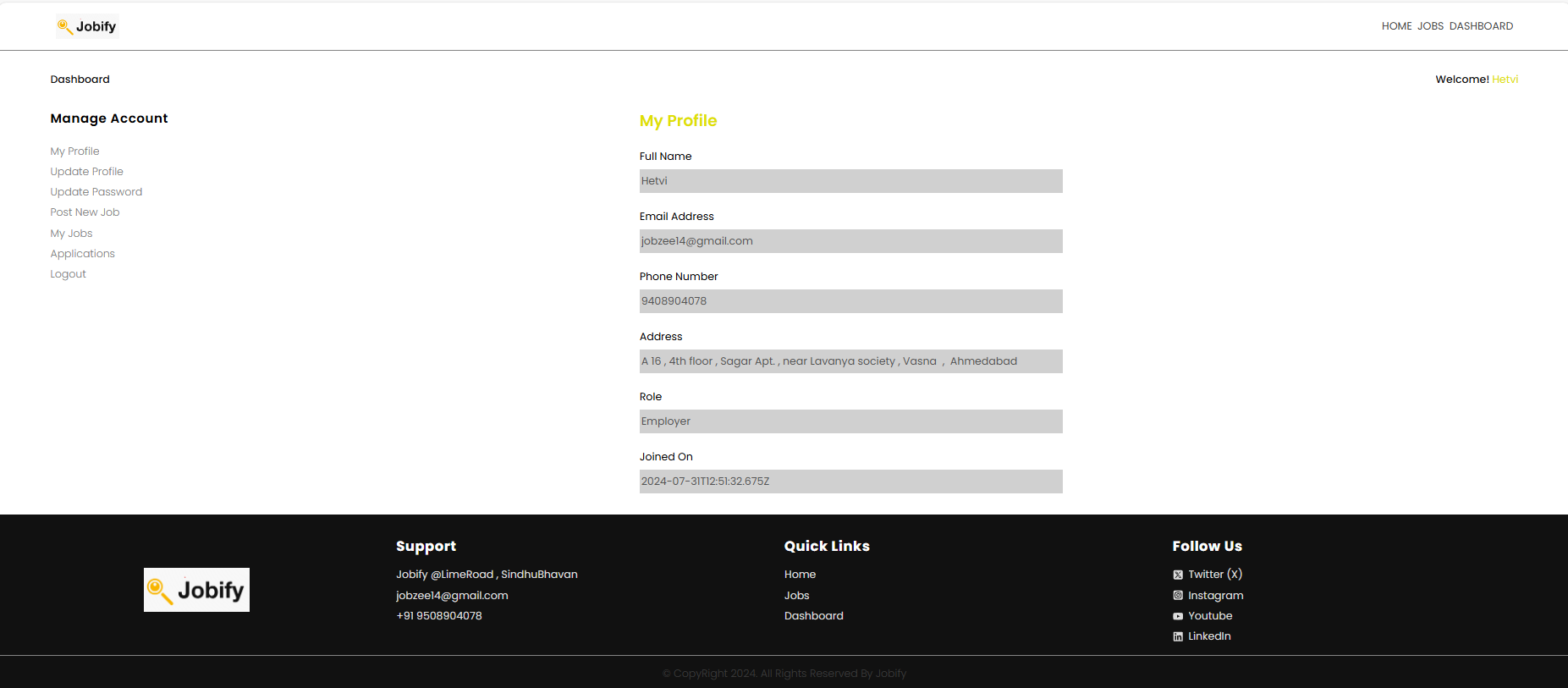
View Jobs Page:



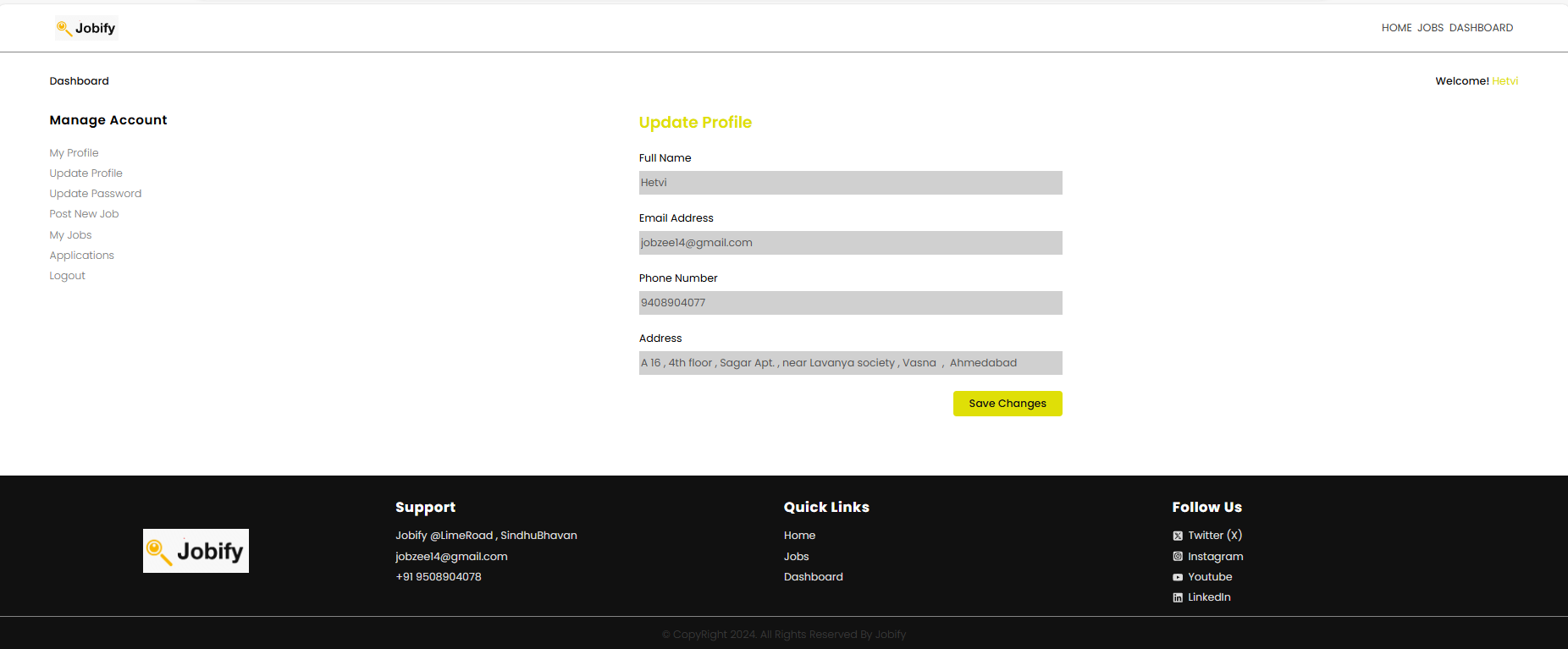
Search based on particular location , niche , keyword:



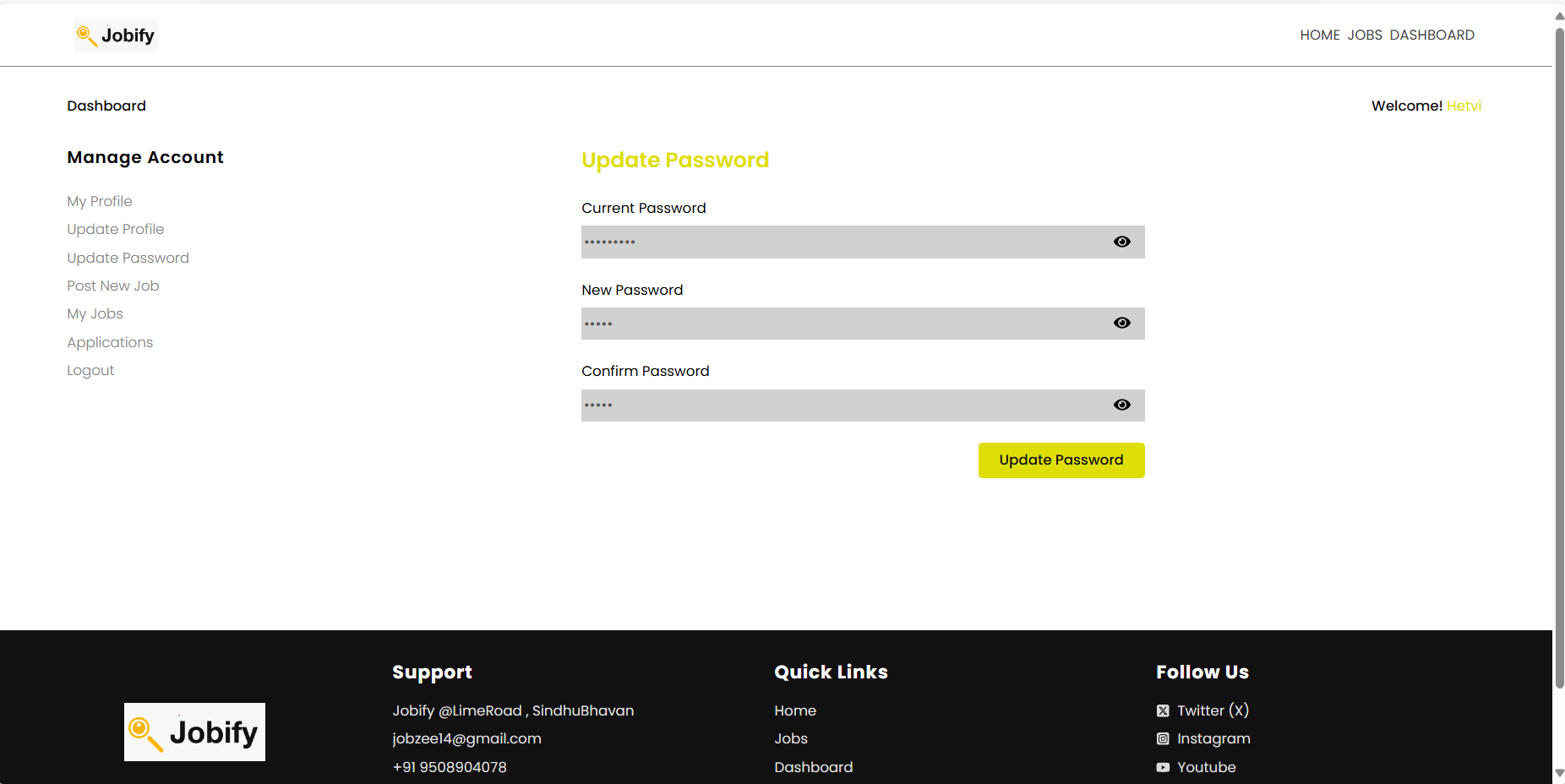
View Profile:



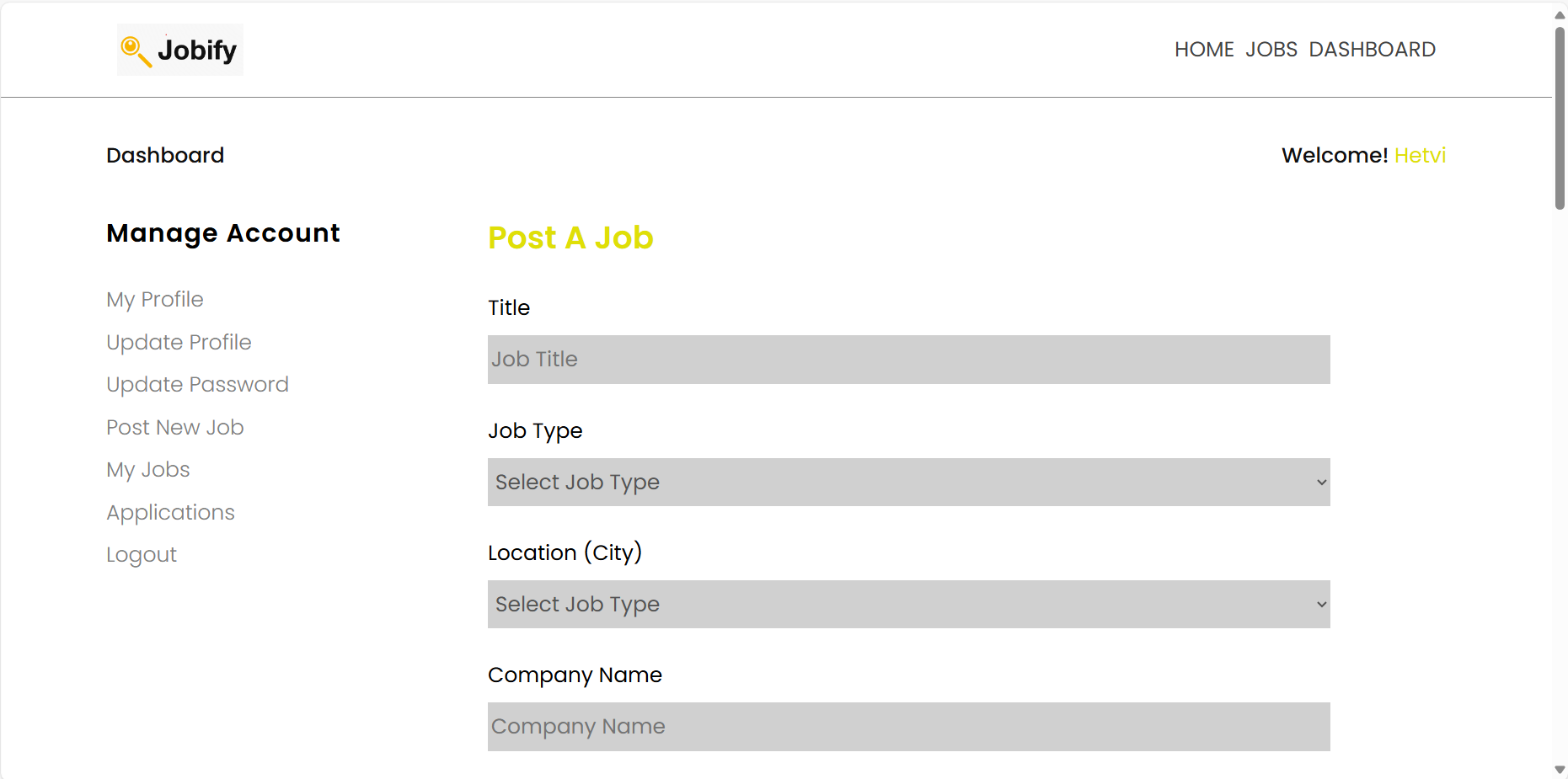
Update Profile:

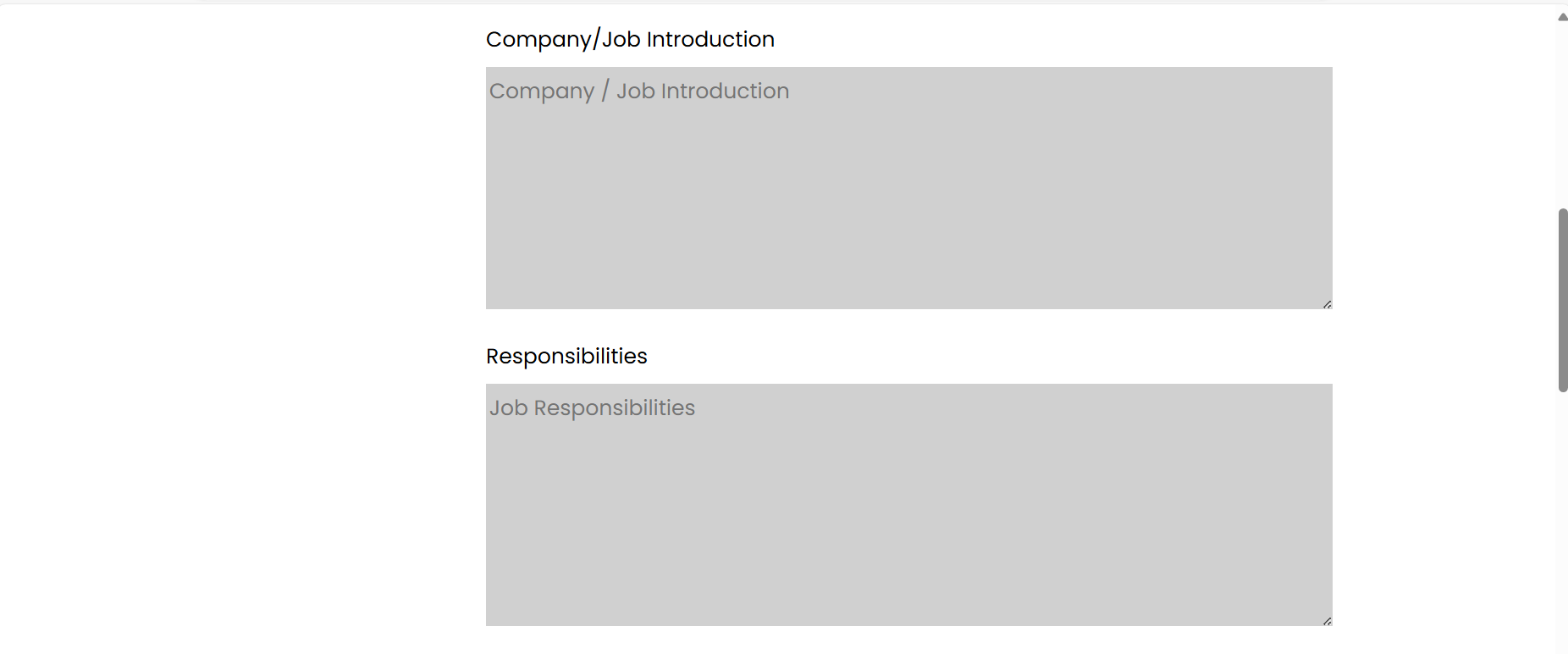


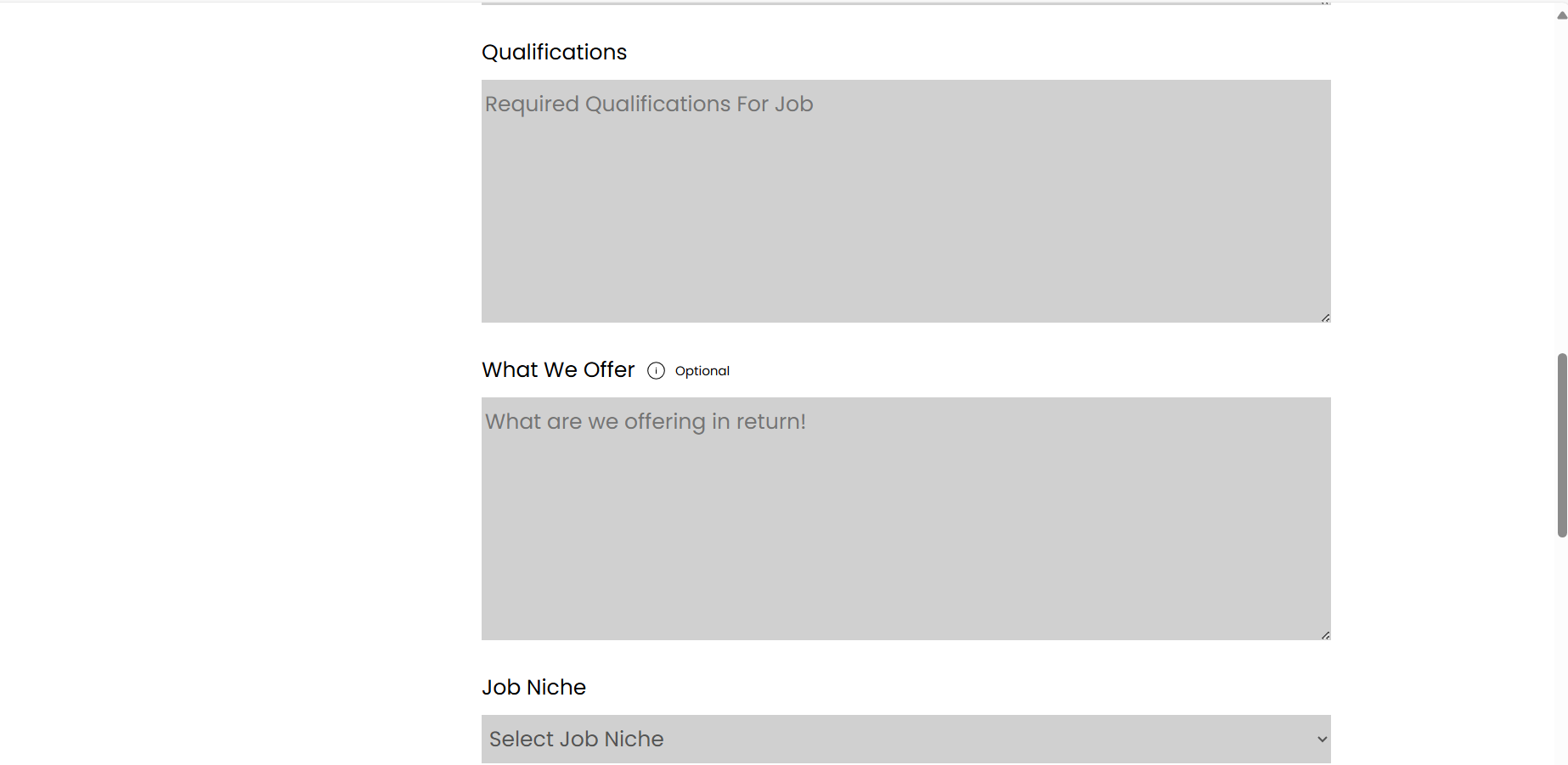
Update Password:

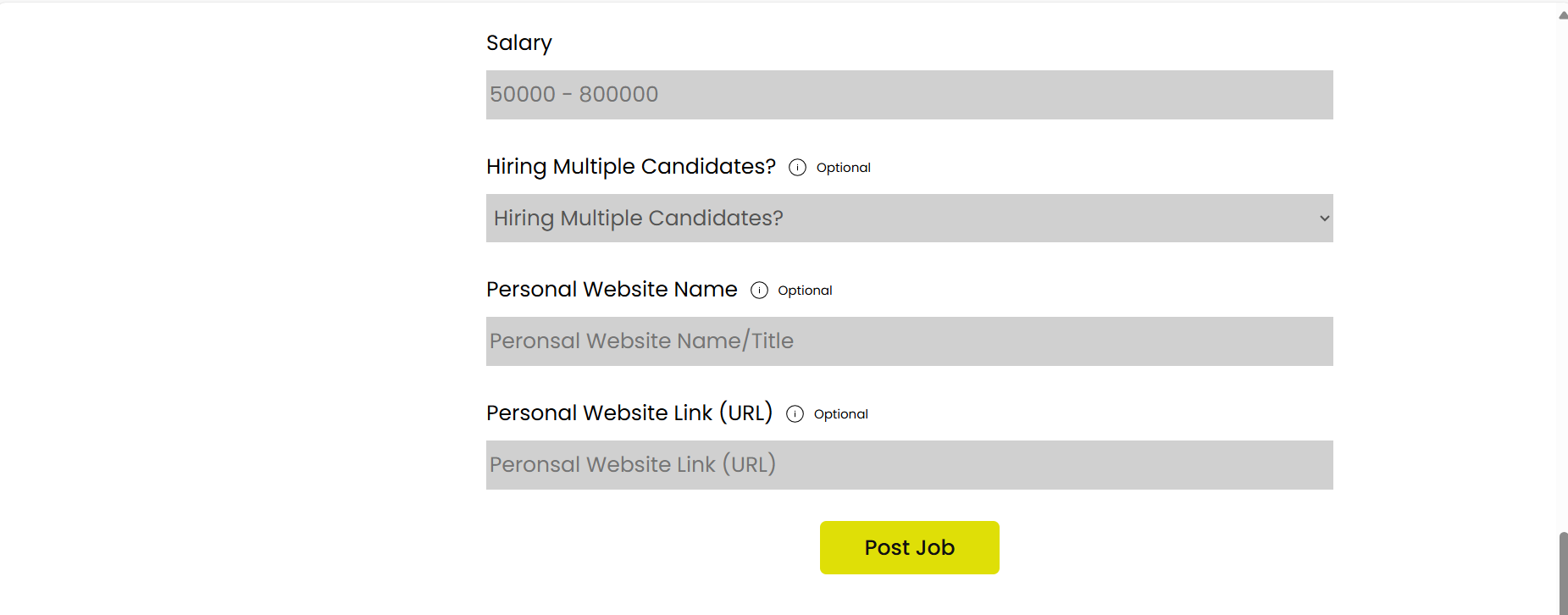


Post a job:





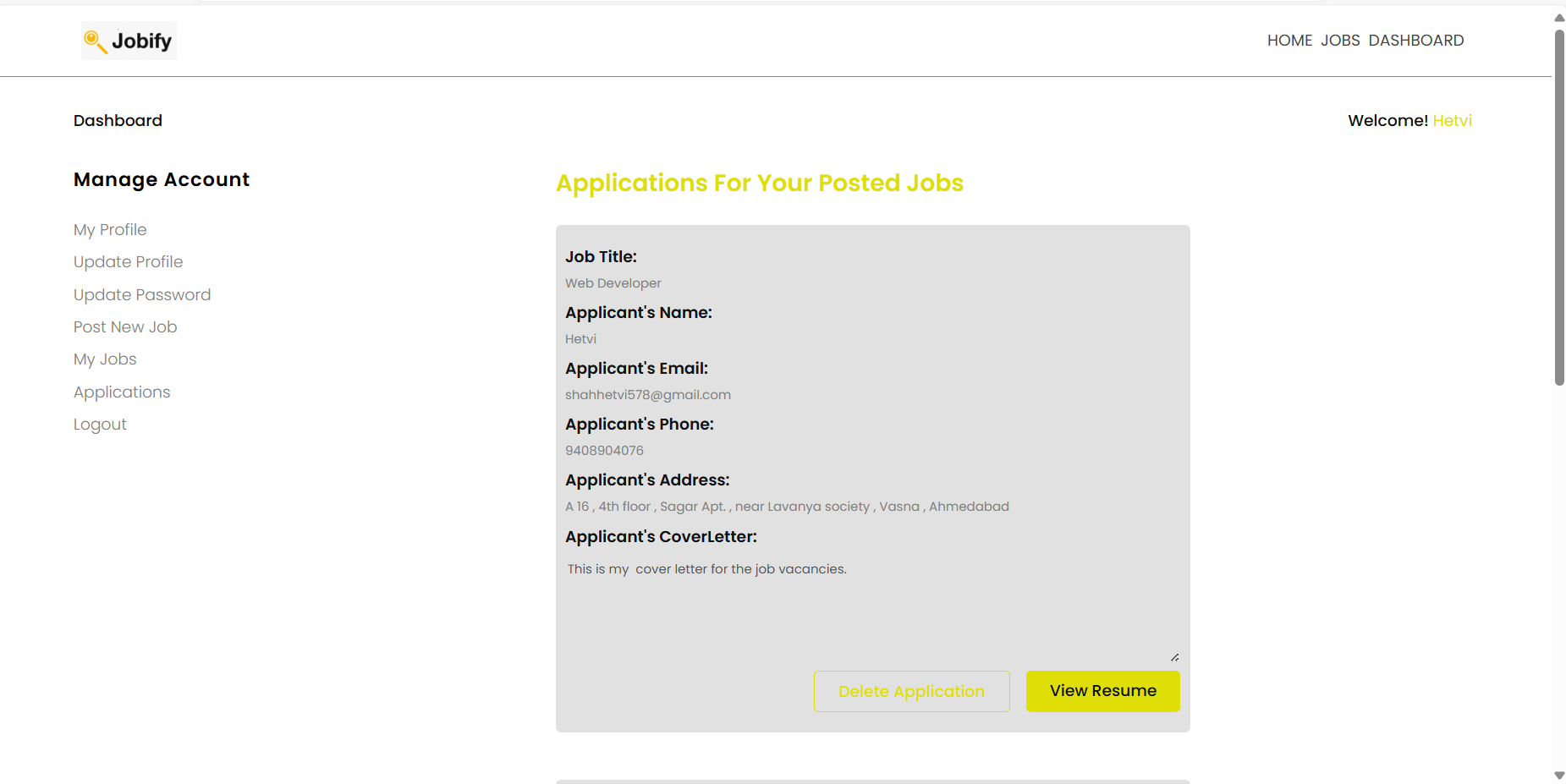




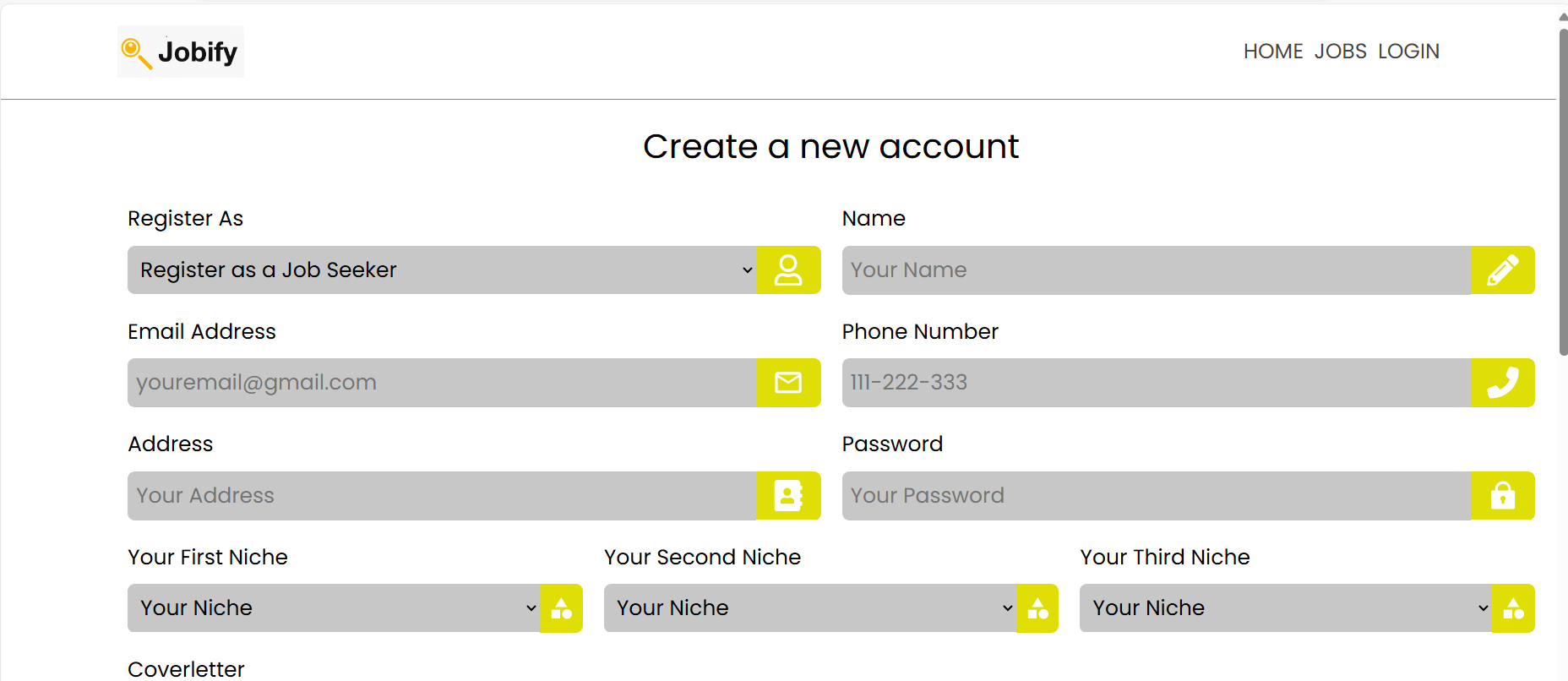
View our posted jobs:



View Applicants:

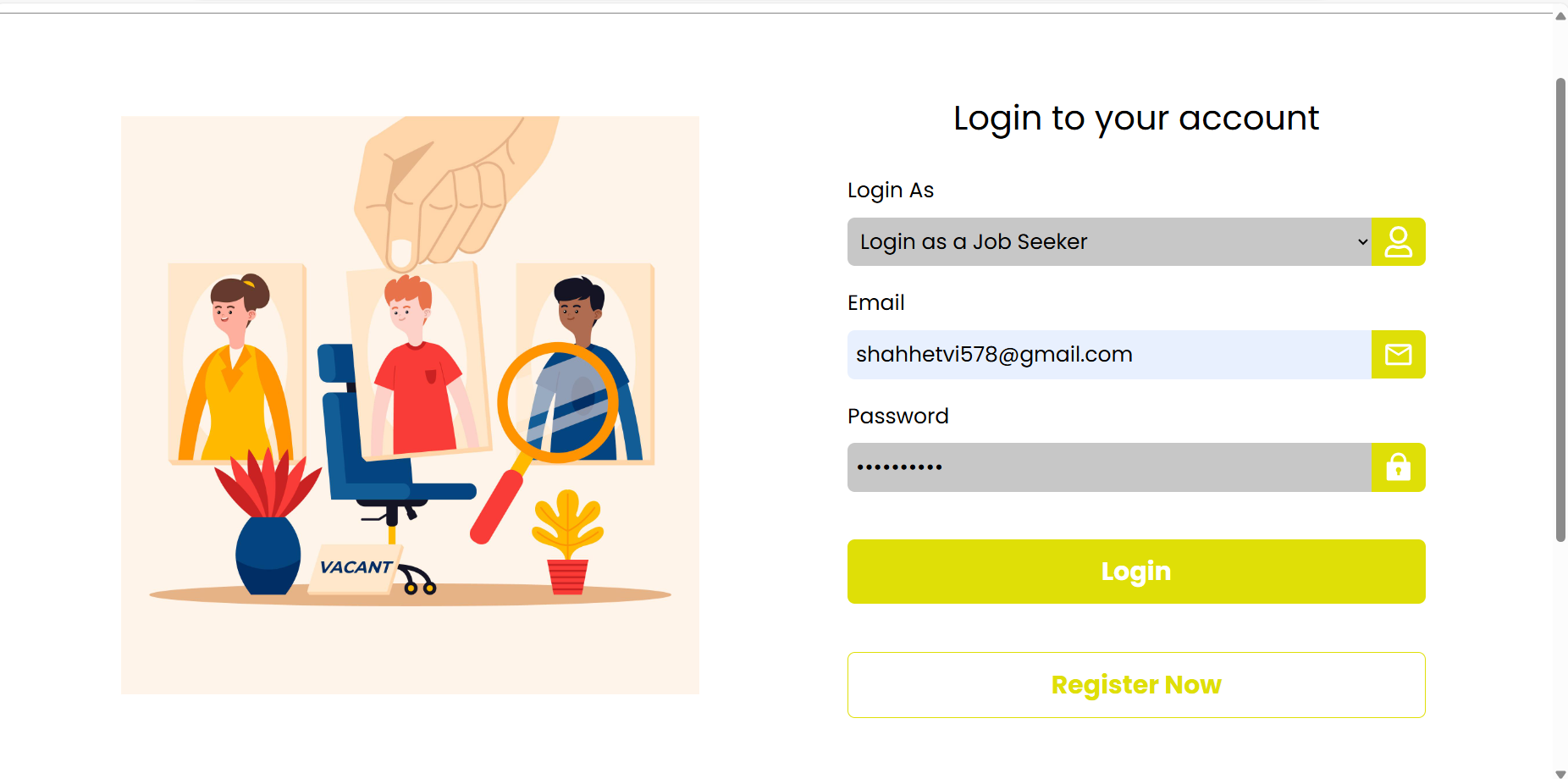


Register as job seeker:

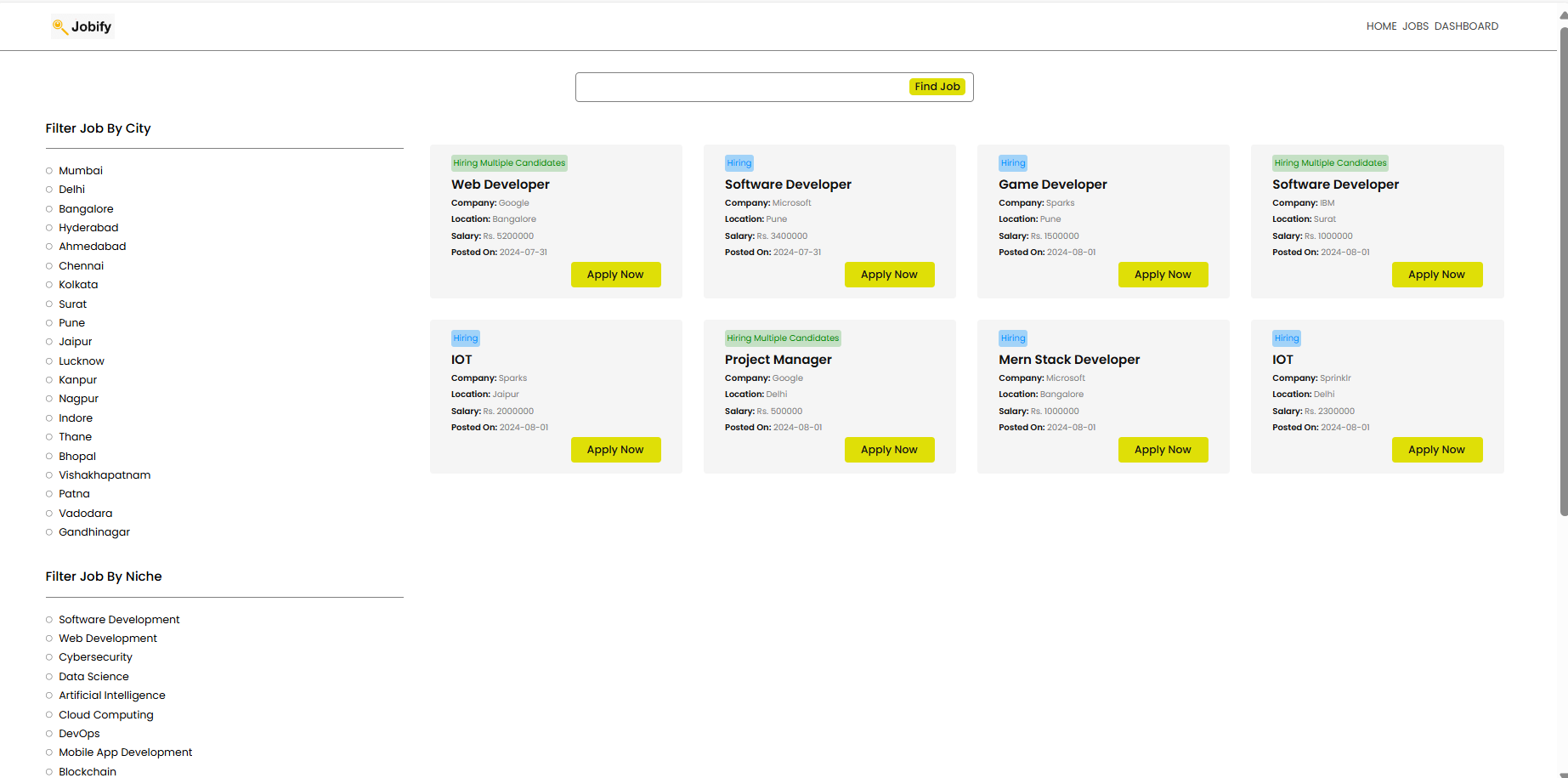




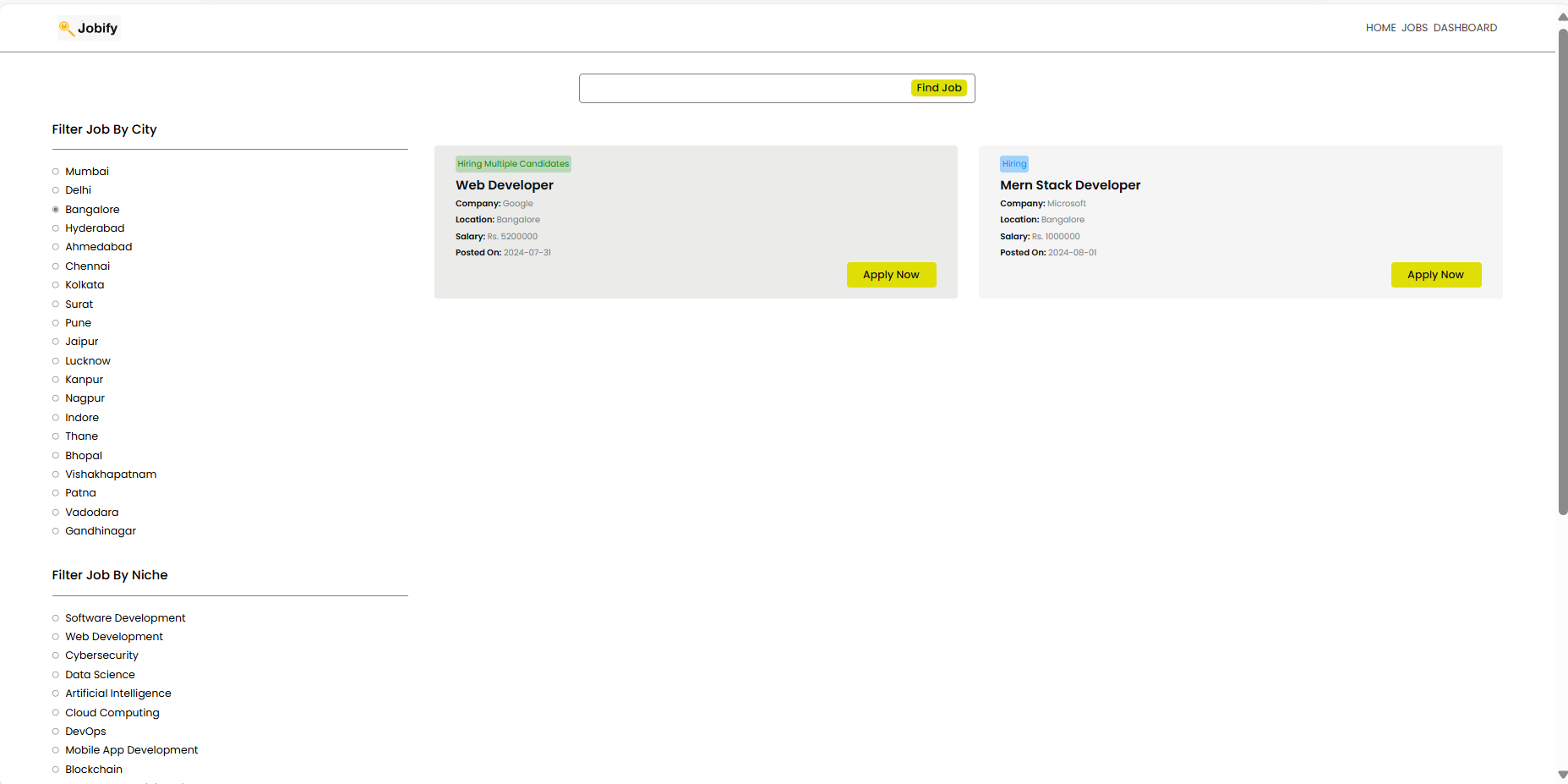
Login as job seeker:



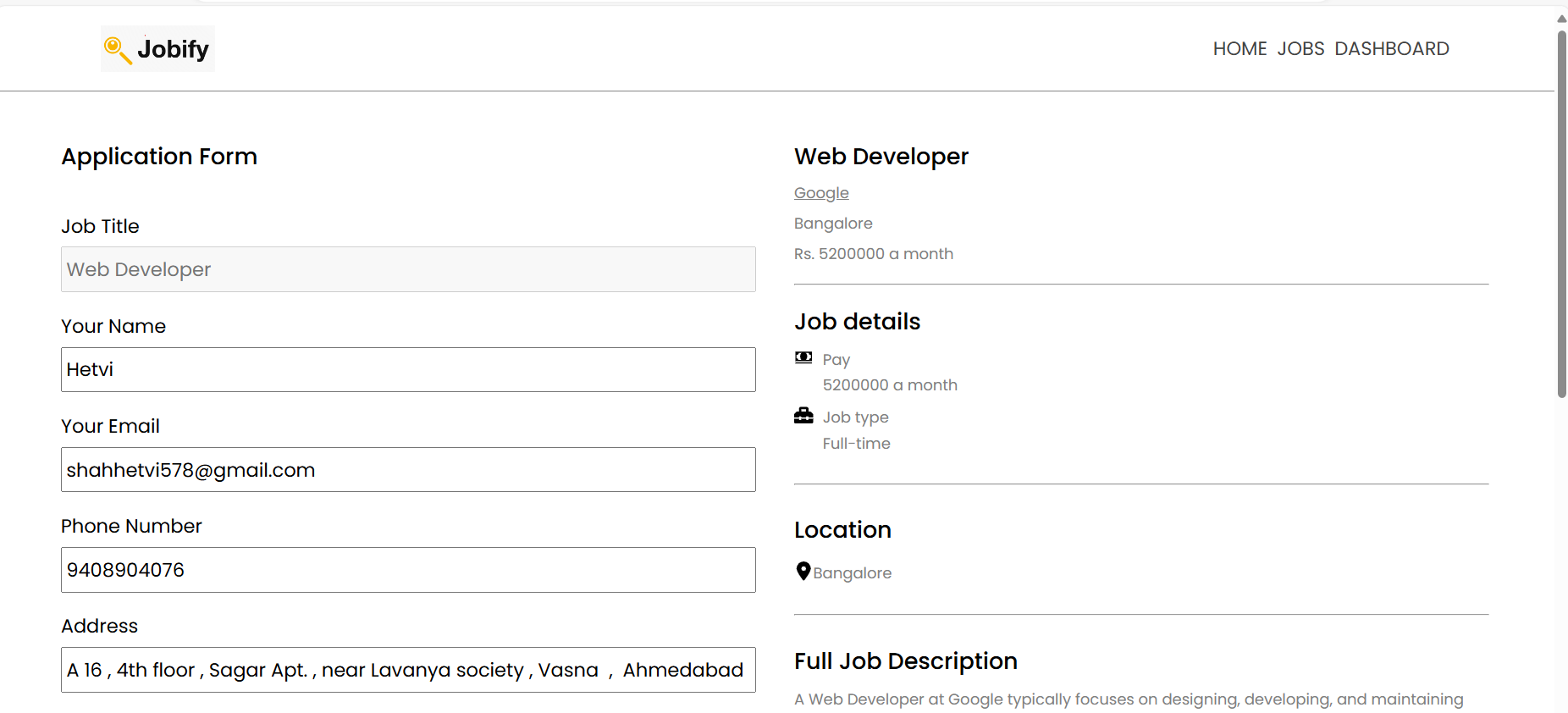
View jobs:



View based on particular niche, location:

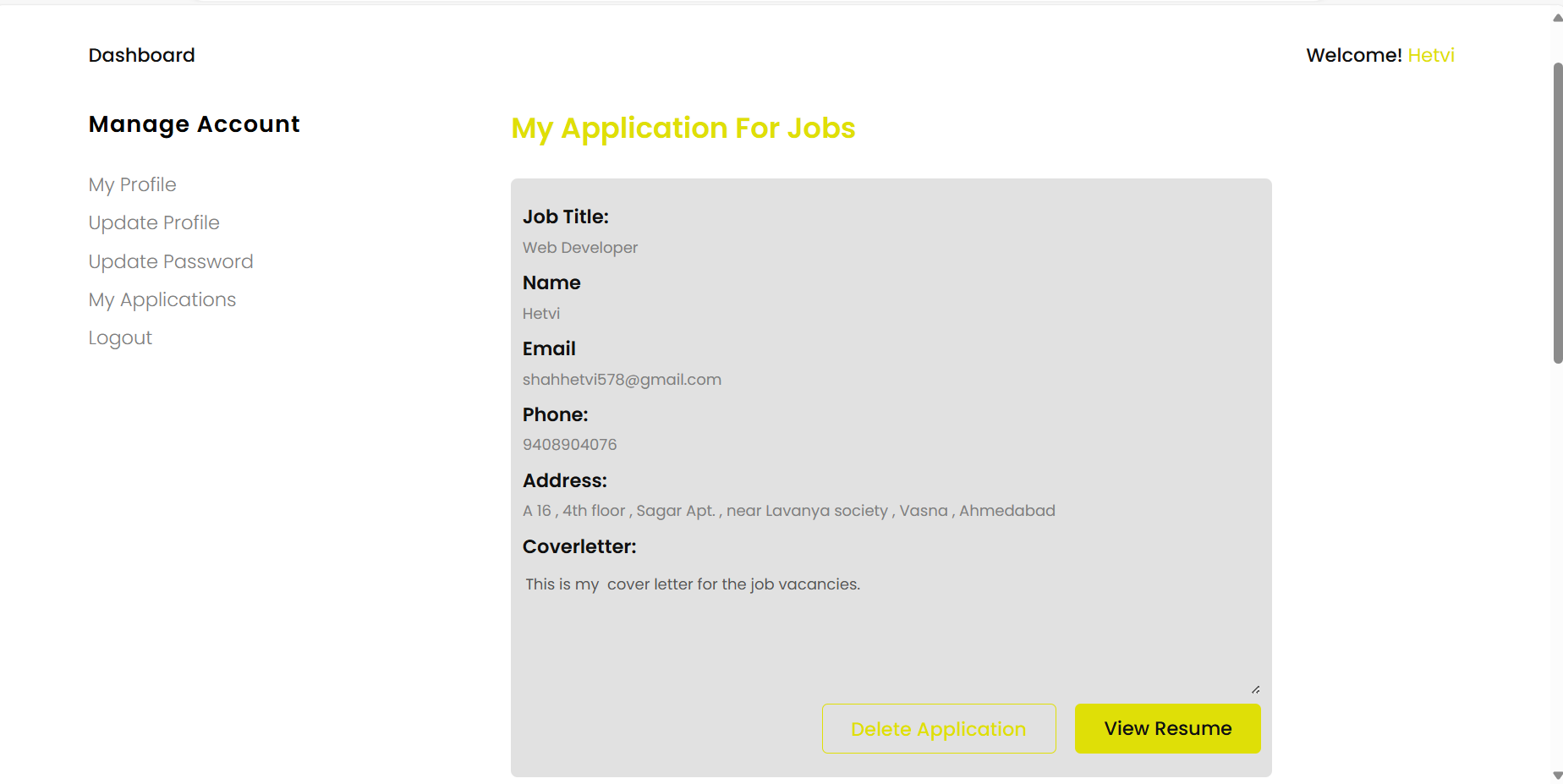


Apply page:





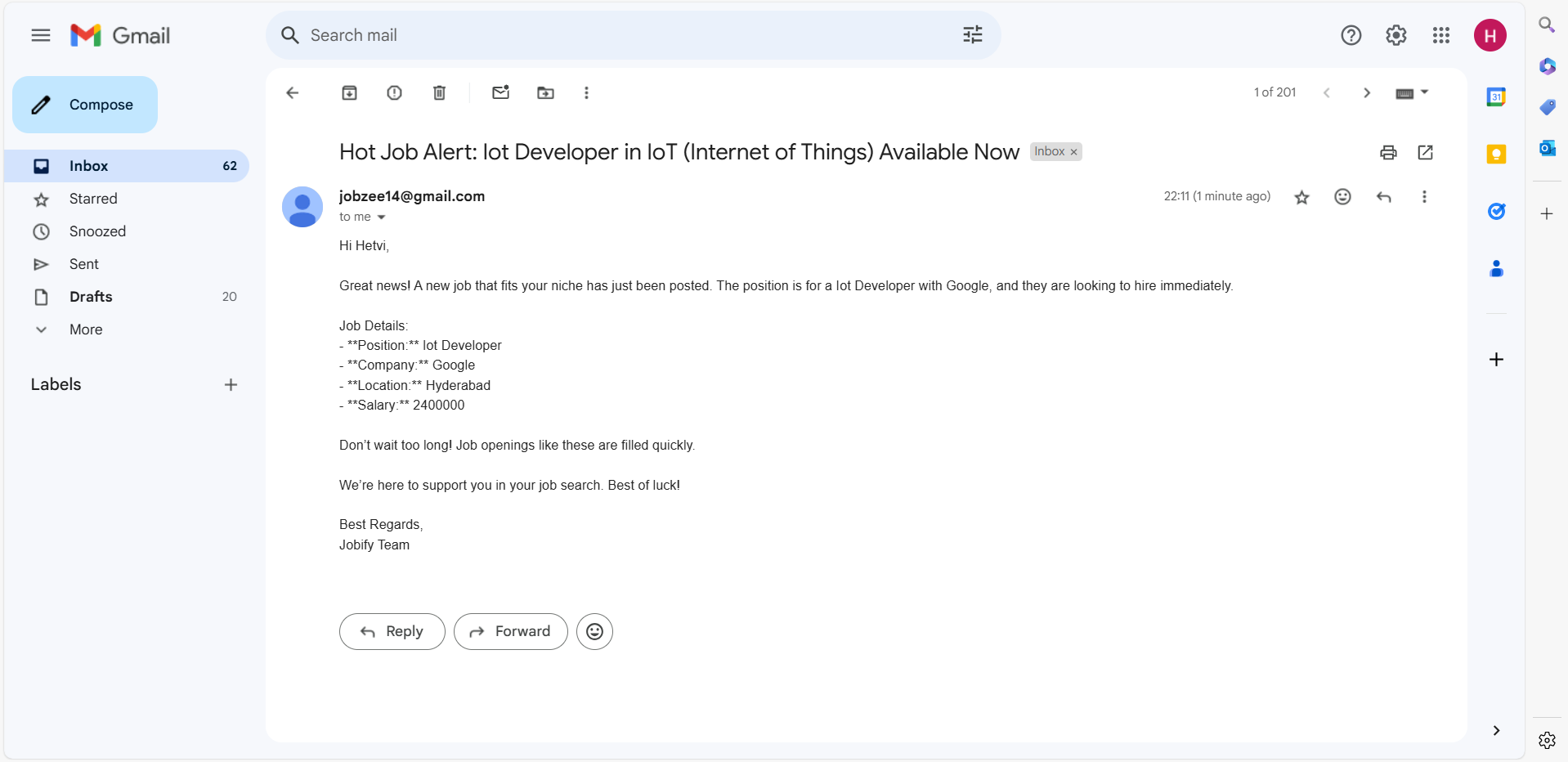
Applications done for jobs:



View Resume:

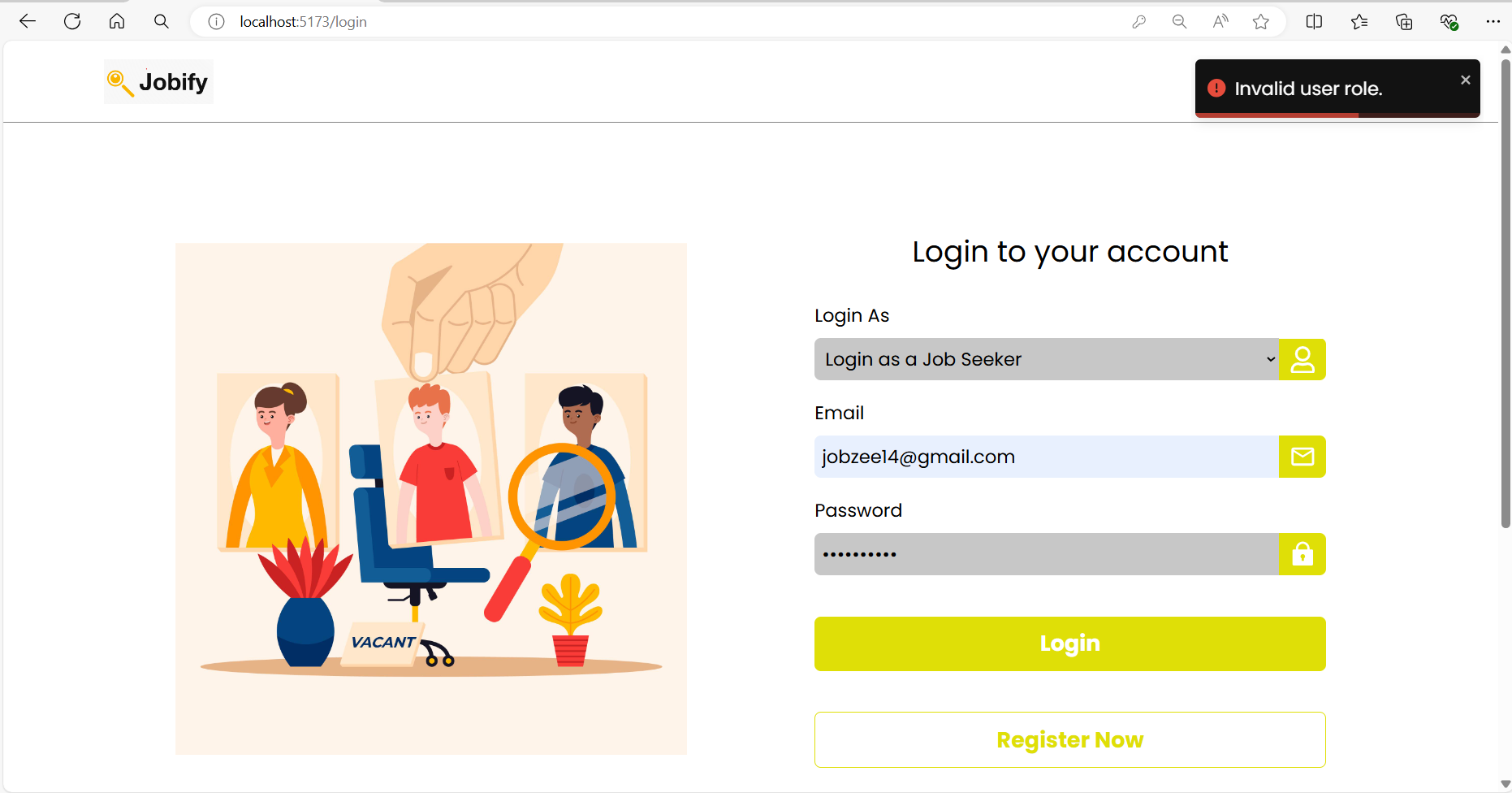


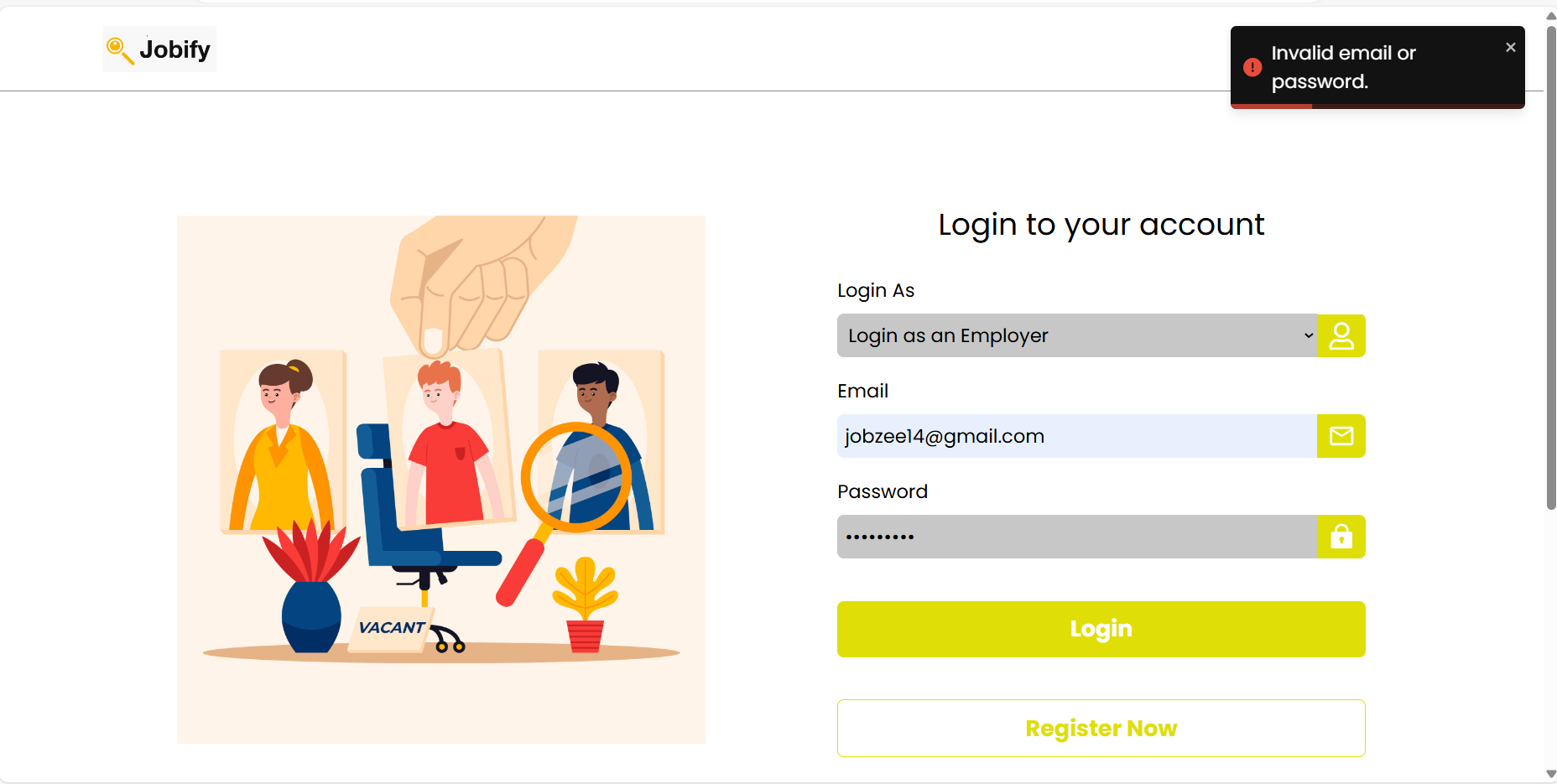
When new job application is posted and it matches with job seekers niche the email is sent:



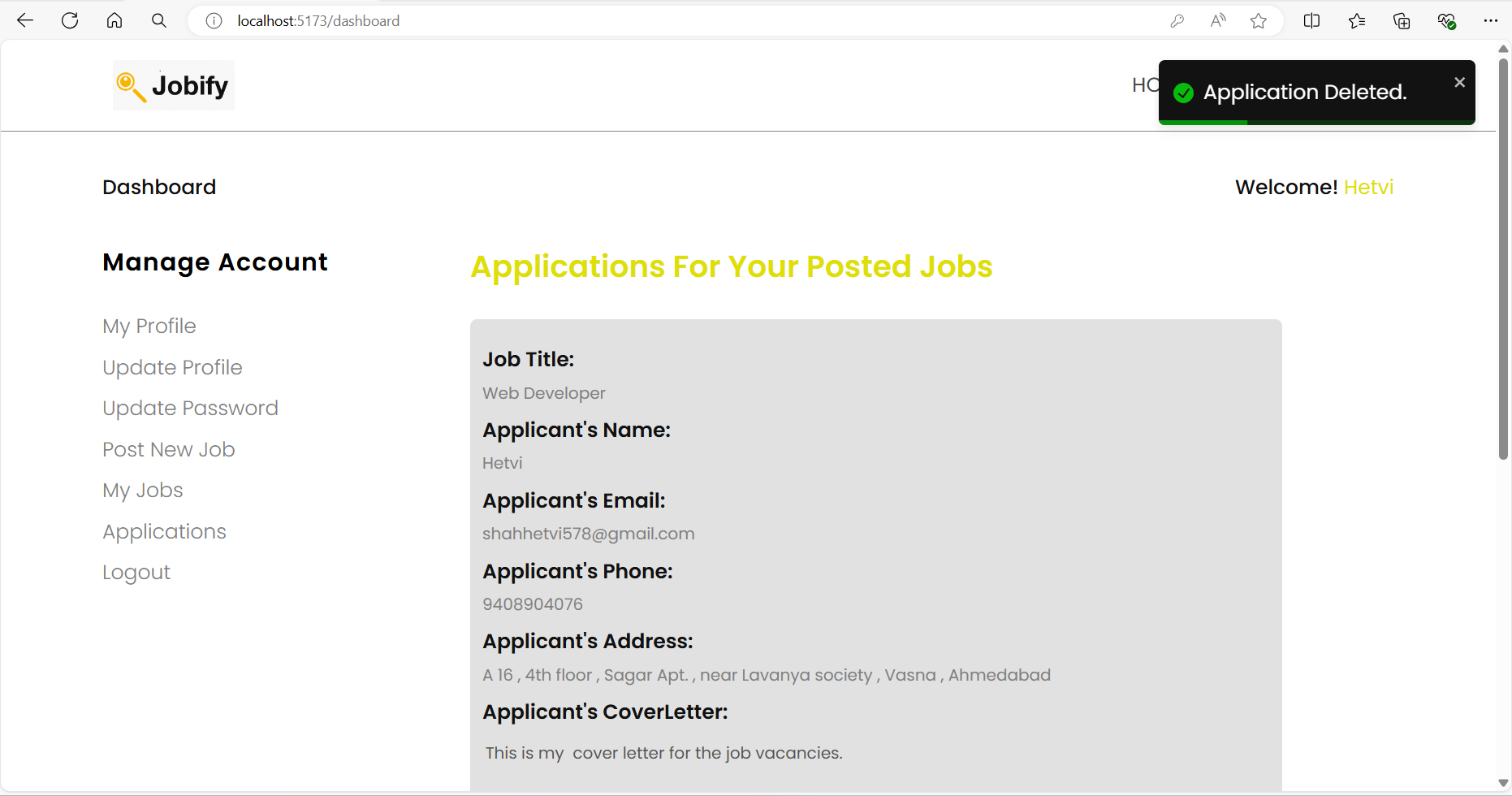
**Testing**

Invalid user role selected:

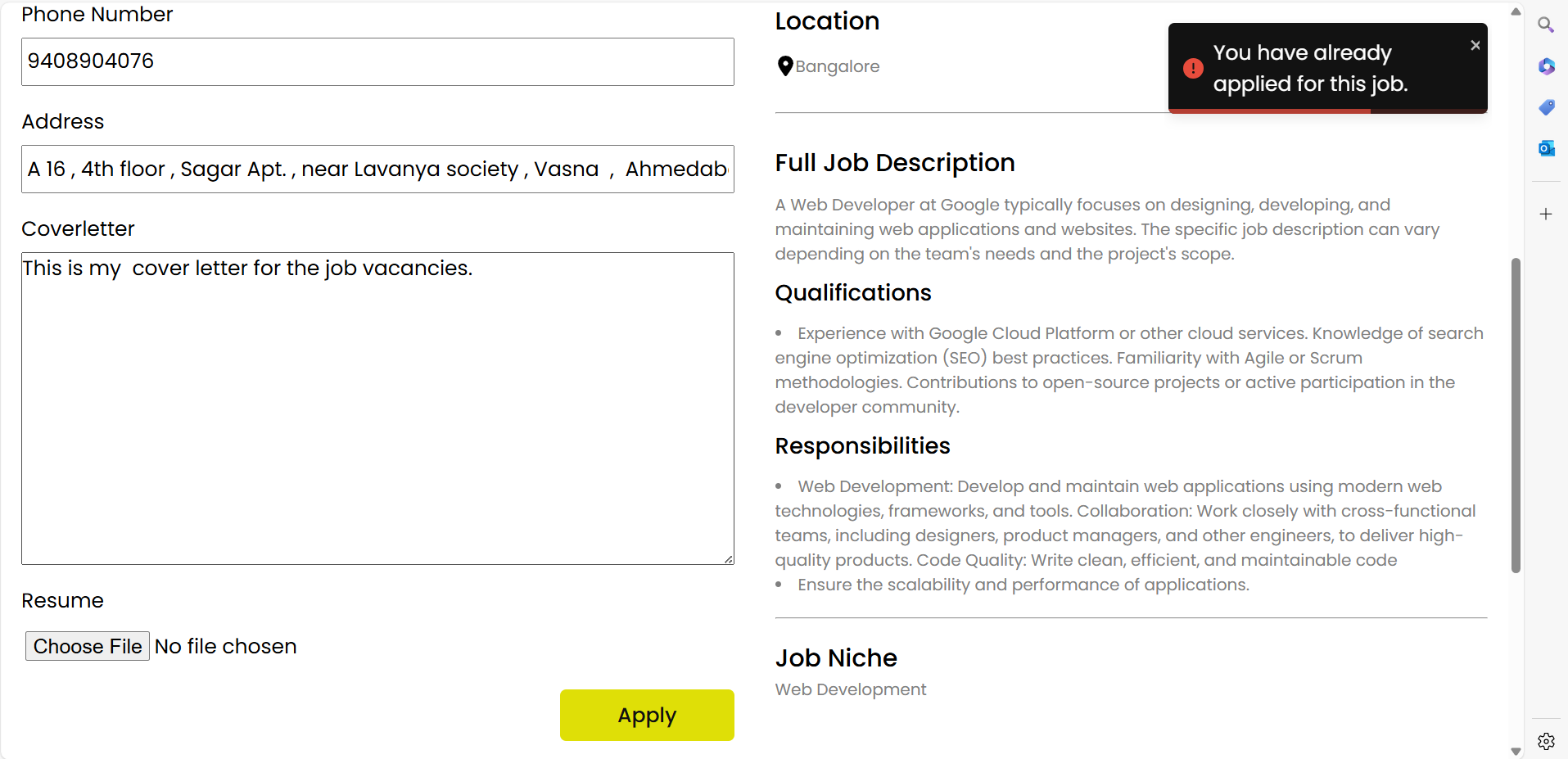


Invalid password or email:  


Deleting the job seekers application:



Cannot apply again for the already applied job:



**Summary and lessons learned**

* The jobify project taught me vital skills in user centric design, agile development, data security, responsive design, troubleshooting, time management, documentation and continuous improvement.
* This lessons have significantly enriched my capabilities as a developer.
* Understanding user requirements and efficient team work were also key takeaways from this project.
* Overall, it has been a transformative and valuable learning experience.

Demo : <https://drive.google.com/uc?id=19Q4KmFmETRqf5Viczbve9sQSgwcRpIb6&export=download>