**Project Report**

**On**

**JOB PORTAL PLATFORM**

**In partial fulfillment for the award of the degree**

**Of**

**BACHELOR OF COMPUTER APPLICATION**

**[B.C.A]**

**Year 2023-2024**

**SUBMITTED BY: GUIDED BY:**

**Vimal Bhesaniya - 06 Asst. Prof. Pratiksha Patel**

**Vishala Gajera - 30**

**Yash Kalambe - 53**

**BCA-6th SEMESTER**

**Submitted to:**



**SHRI SHAMBHUBHAI V. PATEL COLLEGE OF**

**COMPUTER SCIENCE & BUSINESS**

**MANAGEMENT**

**Affiliated to**

**Veer Narmad South Gujarat University**

**ACKNOWELDGEMENT**

Acknowledgement for a job portal website documentation typically recognizes the contributions of individuals who have played a significant role in its development. In this case, we extend our gratitude to the team of three students and our mentor, Professor Pratiksha Patel whose collective efforts have made this project possible.

First and foremost, we express our appreciation to the students, Vimal, Vishala, and Yash, whose dedication, creativity, and hard work have been instrumental in every stage of the project. Each member brought unique skills and perspectives to the table, contributing to the success of the job portal website. Their collaboration, problem-solving abilities, and commitment to excellence have been commendable throughout the development process.

Furthermore, we are deeply thankful for the guidance and support provided by Professor Pratiksha Patel, whose expertise and mentorship have been invaluable. Professor Pratiksha Patel not only offered valuable insights and advice but also encouraged innovation and critical thinking, fostering an environment conducive to learning and growth. Their unwavering support and encouragement have been essential in steering the project in the right direction and ensuring its quality and integrity.

Together, the combined efforts of the students and Professor Pratiksha Patel have culminated in the successful completion of the job portal website documentation. We acknowledge and appreciate their contributions, without which this project would not have been possible. We look forward to utilizing the knowledge and skills gained from this experience in our future endeavors.

From,

Our Team

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SR.NO** | **PARTICULAR** | **PAGE**  **NO.** |
| **1** | **Introduction** |  |
|  | 1.1 Project Profile |  |
| **2** | **System Environment** |  |
|  | 2.1 Hardware Used |  |
|  | 2.2 Software Used |  |
| **3** | **Problem Specification** |  |
|  | 3.1 Introduction, objectives & Purpose |  |
|  | 3.2 System Model |  |
|  | 3.3 Feasibility Study |  |
| **4** | **Risk Identification and Management** |  |
|  | 4.1 Risk Monitoring |  |
| **5** | **Technology Used for Development** |  |
| **6** | **Planning** |  |
|  | 6.1 System Planning |  |
|  | 6.2 Time Line Chart |  |
|  | 6.3 System Flowcharts |  |
| **7** | **System Analysis** |  |
|  | 7.1 ER Diagram |  |
|  | 7.2 Data Flow Diagram |  |
|  | 7.3 Data Dictionary |  |
| **8** | **Designing** |  |
|  | 8.1 User Interface |  |
| **9** | **Testing** |  |
|  | 9.1 Software Testing |  |
|  | 9.2 Unit Testing |  |
|  | 9.3 System Testing |  |
| **10** | **Future Scope & Conclusion** |  |
| **11** | **Reference** |  |

1. **Introduction**

* **Project Profile**
* **Project Description**
* **Advantages & Disadvantages**

* **Project Profile :-**

|  |  |
| --- | --- |
| PROJECT TITLE :- JOB DUNIYA | |
| FRONT – END | REACT - JS |
|  |  |
| BACK - END | NODE - JS |
|  |  |
| BROWSER | ALL BROWSER SUPPORTED |
|  |  |
| TECHNOLOGY | HTML , CSS , JAVASCRIPT , BOOTSTRAP , MongoDB Compass ,EXPRESS – JS, Firebase-Storage |
|  |  |
| GUIDED BY | ASST. PROF.PRATIKSHA PATEL |
|  |  |
| PLATFORM | 18.2.0 |
|  |  |
| TOOLS USED FOR | VISUAL STUDIO , MongoDB Compass , POSTMAN,GitHub |
|  |  |
| SUBMITTED TO | SHREE SHAMBHUBHAI V. PATEL COLLEGE |
|  |  |
| DEVELOPED BY | VIMAL BHESANIYA - 06  VISHALA GAJERA - 30  YASH KALAMBE - 53 |

* **Project Description :-**

**The Job Portal Platform** is an online web-based system designed to connect job seekers with employers, providing a seamless and efficient platform for job searching and recruitment. This small project description outlines the key features and functionalities of the Job Portal Platform:

* **Features :-**
* User Registration and Profiles.
* Job Search and Posting By Companies.
* Application Management By Companies.
* Employers can use filters and keywords to narrow down the list of applicants, making the recruitment process more efficient.
* Employers can access data and reports on the performance of job listings and candidate applications. System admin handles all the policy, users, and advisers who are using the system.
* An admin dashboard for platform administrators to manage users, monitor activity, and ensure the platform's smooth operation

1. **User Panel Features: -**
   * + - User Registration and Profile Creation
       - User Login / Logout.
       - Job Application Management
       - Job Alerts and Notifications
       - Feedback and Support
2. **Company Panel Features: -**

* Company Registration and profile creation
* Company Login / logout
* Company Dashboard
* Job Application Management
* Job posting
* **Advantages & Disadvantages :-**
* **Advantages:-**

**1. Wider Reach:** Job portals offer a platform where employers can reach a larger pool of potential candidates. This wider reach increases the chances of finding suitable candidates for job vacancies.

**2. Convenience:** Both job seekers and employers benefit from the convenience offered by job portals. Job seekers can search and apply for jobs from the comfort of their homes, while employers can post job openings and review applications online, saving time and resources.

**3. Cost-Effective:** Job portals often offer cost-effective solutions for both job seekers and employers. Job seekers can access a wide range of job postings for free or at minimal cost, while employers can reach a large audience without the need for expensive traditional advertising methods.

**4. Filtering Options:** Job portals usually provide advanced search and filtering options, allowing job seekers to narrow down their search based on criteria such as location, industry, salary, etc. This helps job seekers find relevant job opportunities more efficiently.

**5. Time-Saving:** Job portals streamline the recruitment process for both parties, saving time that would otherwise be spent on manual job searches or sorting through numerous resumes. This efficiency benefits both job seekers and employers.

* **Disadvantages:-**

**1. Competition:** The popularity of job portals means that job postings often receive a large number of applications, leading to increased competition for job seekers. This can make it challenging for individuals to stand out among the crowd and secure interviews.

**2. Impersonal:** The online nature of job portals can sometimes result in a lack of personal interaction between job seekers and employers. This impersonal aspect may make it difficult for job seekers to gauge company culture or for employers to assess candidate suitability beyond the resume.

**3. Job Scams:** Job portals are susceptible to fraudulent job postings and scams, where individuals or organizations may attempt to deceive job seekers for malicious purposes. Job seekers need to exercise caution and thoroughly research job opportunities to avoid falling victim to scams.

**4. Limited Information:** Job postings on portals often provide limited information about the job role, company, or application process. This lack of detail can make it challenging for job seekers to fully understand the requirements and expectations of a position before applying.

**5. Dependency on Technology:** Job portals rely heavily on technology and internet connectivity. Technical glitches, website downtime, or connectivity issues can hinder the job search process for both job seekers and employers, causing frustration and delays.

1. **System Environment**

* **Hardware Configuration**
* **Software Configuration**
* **Hardware Configuration:-**

|  |  |
| --- | --- |
| Hardware Configuration | |
| Processor | 11th Gen Intel(R) Core(TM) i5-1155G7 @ 2.50GHz 2.50 GHz |
| Hard Disk | 512 MB Or More |
| RAM | 8.00 GB |

* Software Configuration :-

|  |  |
| --- | --- |
| Software Configuration | |
| OS | Windows 11 Pro |
| Browser | Microsoft Internet Explorer, Mozilla and Google Chrome |
| Front-End | React – JS |
| Back-End | Node - JS |

1. **Problem Specification**

* **Introduction , Objectives**

**& Purpose**

* **System Model**
* **Feasibility Study**
* **Introduction , Objectives**

**& Purpose :-**

* **Introduction :-**
* Welcome to our cutting-edge job portal website, where the future of career exploration and talent acquisition converges. Developed with React.js for an immersive frontend experience and powered by Node.js backend with MongoDB, our platform redefines the way individuals discover employment opportunities and organizations identify top-tier talent.
* Through seamless integration of advanced technologies, our job portal transcends traditional job search paradigms, offering a dynamic and intuitive interface that empowers users to navigate the complexities of the modern workforce with ease. Whether you're a seasoned professional seeking new horizons or an organization poised for growth, our platform serves as a catalyst for realizing your aspirations and ambitions.
* With an unwavering commitment to innovation and user-centric design, we invite you to embark on a transformative journey where possibilities abound and opportunities await. Join us as we revolutionize the world of recruitment and pave the way for a future where talent knows no bounds.
* **Objectives :-**

**1. Efficient Job Search:** Our primary objective is to provide job seekers with a seamless and efficient job search experience. Through advanced search filters and personalized recommendations, users can easily find relevant job listings tailored to their skills and preferences.

**2. Streamlined Recruitment Process:** For employers, our platform simplifies the recruitment process by offering tools for posting job vacancies, managing applications, and conducting candidate evaluations. We aim to streamline the hiring process, saving time and resources for businesses.

**3. Enhanced User Experience:** We prioritize user experience in every aspect of our platform. From intuitive navigation to responsive design, our goal is to create a user-friendly interface that ensures a positive experience for both job seekers and employers.

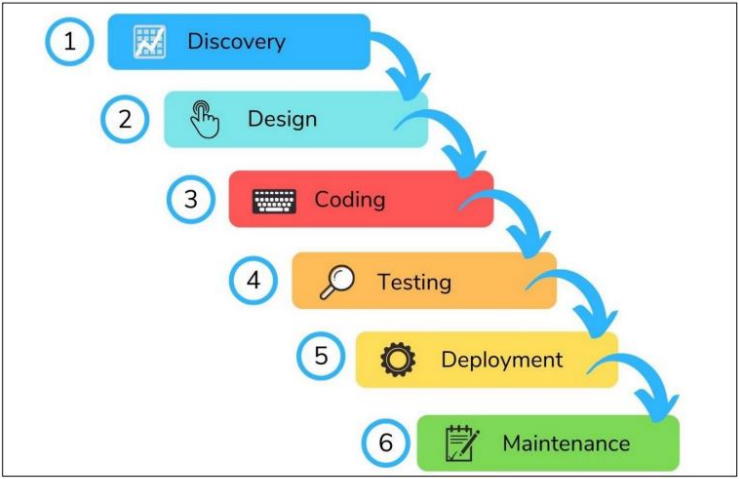
**4. Comprehensive Job Listings:** We strive to provide a comprehensive database of job listings across various industries and locations. By partnering with a wide range of companies, we aim to offer diverse opportunities to our users, catering to different skill sets and career aspirations.

**5. Data Security and Privacy:** Ensuring the security and privacy of user data is a top priority. We implement robust security measures to protect sensitive information and adhere to strict privacy policies to safeguard user confidentiality.

* **Purpose :-**

Our job portal website serves multiple purposes, catering to the needs of both job seekers and employers:

* **For Job Seekers:-**
* Facilitate the job search process by providing easy access to a wide range of job opportunities.
* Offer personalized recommendations based on user preferences and past interactions.
* Provide resources and guidance to help users advance their careers, such as resume tips and interview preparation materials.
* Foster a supportive community where users can connect with peers, seek advice, and share insights.
* **For Employers:-**
* Simplify the hiring process by offering a platform for posting job vacancies and managing applications.
* Provide tools for screening and evaluating candidates, including customizable assessment features.
* Help employers build their employer brand and attract top talent through targeted marketing initiatives.
* Offer analytics and insights to help employers optimize their recruitment strategies and improve hiring outcomes.
* **System Model :-**
* **Waterfall Model :-**
* For any project to be completed, it has to go through stages called Development Life Cycles. System Development Life Cycle (SDLC) is the process of understanding how an Information System (IS) can support business needs, designing the system, building it and delivering it to users. The SDLC composes of four phases: Planning, Analysis, Design and Implementation.
* In order for this project to be developed, the methodology that will be used is the System Structured Analysis and Design Methodology. The SSADM is classified as a Waterfall Development. With Waterfall Development, analyst and users proceed sequentially from one phase to the next and each phase can be mapped out and evaluated (Hevner, 2004). Below, in figure is a diagram on the waterfall methodology.



* **Why We Use Waterfall Model ?**

Waterfall relies on teams following a sequence of steps and never moving forward until the previous phase has been completed. This structure is suited to smaller projects with deliverables that are easy to define from the start.

"Waterfall is generally regarded with some disdain as an inefficient and passé traditional project management approach. But Waterfall can be a useful and predictable approach if requirements are fixed, well documented, and clear, if the technology is understood and mature, if the project is short, and if there’s no additional value gained from 'going agile.' A Waterfall approach can actually provide more predictable end result for budget, timeline, and scope."

* Requirements are very well documented, clear and fixed.
* Product definition is stable.
* Technology is understood and is not dynamic.
* There are no ambiguous requirements.
* The project is short.
* Simple and easy to understand and use.
* Easy to manage due to the rigidity of the model.
* Each phase has specific deliverables and a review process.
* Phases are processed and completed one at a time.
* Works well for smaller projects where requirements are very well understood.
* Clearly defined stages.
* Well understood milestones.
* Easy to arrange tasks.
* Process and results are well documented.
* **Feasibility Study :-**

The feasibility of our project has been judged on the time, technology, resources available, and project length.

* **Time:-** This project takes at least 2 months to be completed if we take the help of reused components; otherwise it may take more than 5 months to be completed.
* **Technology:-** This project is built on React.js & Node.js platform which is worldwide already standard. Therefore the technical risk is not so high hence feasible.
* **Resources:-** Good technology professionals are a basic requirement. Internet and favorable environment, etc. the resources were fully available.

**1. Economic Feasibility:**

The prime requirement of this project is an internet connection as the project been developed is a Website. Moreover, it also demands a good server for data redirection and high speed and efficient CPU ram and processing and managing the online transaction done by the online Jobseeker & Recruiter.

**2. Technical Feasibility:**

This website requires high and up-to-date technology. It needs database feedback and also needs to be accessed through the web and the internet. This can be done easily, so online payment should be facilitated. We provide our clients with the right worker for their job. So that clients can do their work easily and with proper use of their time. We provide our workers with the right clients for their job. So that worker can do their work easily and with proper use of their time.

**3. Management Feasibility:**

Management feasibility ratio and aspect of management news. Here all level of management rise of related basic feasibility and gives their decision for all those feasibilities to use or not is comes in basic feasibility of the system, bus how the requirement of that all, after that all manager.

**4. Operational Feasibility:**

The web application can be beneficial only if it satisfies the organization’s requirement; in such a way that resource utilization & the optimum outcome is justified. A web application should not only be robust but should also be able to work simultaneously with other systems. Operational feasibility means that web applications should not affect any existing system during the development phase or even in the implementation phase.

**5. behavioural feasibility:**

The new system can be beneficial only if it satisfies the organization’s requirements. In such a way that resource utilization and the optimum outcome are justified. A new system should not only be robust but should also be able to work simultaneously with other systems. Operational feasibility means that a new system should not affect any existing system during the development phase or even in the implementation phase. Following are some points underlying the operational feasibility of the system. As the development proceeded many doubts got clear doubt.

* Our project guide- Asst. Prof. Mansi Shah guided us to take proper care and check for the operational feasibility of the system.
* Efforts were made to optimize the human efforts in data collection, storage, retrieval, security, and presentation.
* The proposed system made its best efforts in achieving the necessary function and performance. As required by the user and keeping in mind some infrastructure constraints. Some of the following issues raised are to test the operational feasibility of a project including the following.
* Is there sufficient support for the management from the user?
* Will the system be used and work properly if it is being developed and implemented?
* Will there be any resistance from the user that will undermine the possible application benefits?
* This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration so there is no question of resistance from the user that can undermine the possible application benefits.

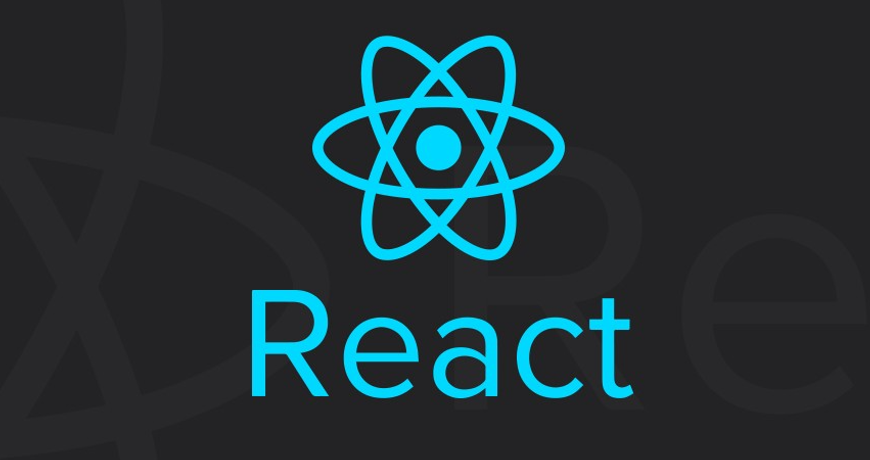
1. **Risk Identification &**

**Management**

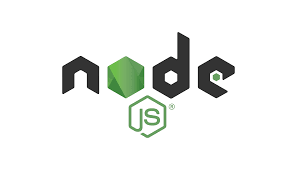
* **Risk Monitoring**

1. **Technology Used For Development**

* **Recat.js:-**



* **What is React.js ?**
* React.js is a popular JavaScript library used for building user interfaces (UIs) in web applications. Developed by Facebook, React.js has gained widespread adoption due to its efficiency, flexibility, and component-based architecture. It allows developers to create interactive and dynamic UIs by breaking down the application into reusable components.
* At its core, React.js utilizes a virtual DOM (Document Object Model) which improves performance by minimizing the need for direct DOM manipulation. Instead of updating the entire DOM when changes occur, React compares the virtual DOM with the real DOM and only applies the necessary updates. This approach results in faster rendering and a smoother user experience.
* One of the key features of React.js is its component-based architecture. Components are encapsulated and reusable pieces of UI that can be composed together to build complex interfaces. This modular approach promotes code reusability, maintainability, and scalability, making it easier to manage large codebases.
* React.js also embraces a declarative programming style, allowing developers to describe what the UI should look like at any given point in time. This simplifies the process of UI development and makes the code more predictable and easier to debug.
* **Advantages of React.js :-**
* **Virtual DOM:** React's virtual DOM enables efficient updates to the actual DOM, resulting in improved performance and faster rendering of UI components.
* **Component-Based Architecture:** React encourages the creation of reusable components, facilitating modular development, code reusability, and easier maintenance.
* **Declarative Syntax:** With a declarative syntax, React simplifies the process of building UIs by allowing developers to describe the desired UI state, rather than focusing on the imperative steps to achieve it.
* **Unidirectional Data Flow:** React follows a unidirectional data flow, ensuring that data flows in a single direction, which makes it easier to understand and debug code, especially in large applications.
* **Rich Ecosystem:** React.js has a vast ecosystem of tools, libraries, and community support, providing developers with access to a wide range of resources to enhance their development workflow and solve common challenges.
* **Performance Optimization:** React's performance optimization features, such as memoization, lazy loading, and code splitting, help improve the overall performance of web applications.
* **Node.js :-**



* **What is Node.js ?**
* Node.js is an open-source, cross-platform JavaScript runtime environment that allows developers to run JavaScript code outside of a web browser. Created by Ryan Dahl in 2009, Node.js has gained significant popularity in the development community due to its efficiency and scalability in building server-side and networking applications.
* At its core, Node.js is built on Chrome's V8 JavaScript engine, which compiles JavaScript code into machine code, resulting in fast execution. It utilizes an event-driven, non-blocking I/O model, which makes it lightweight and efficient for handling multiple connections simultaneously. This architecture enables Node.js to handle high levels of concurrent connections with low overhead, making it suitable for real-time applications such as chat applications, streaming services, and online gaming platforms.
* **Advantages of Node.js :-**
* **Single Language:** With Node.js, developers can write both client-side and server-side code using JavaScript, simplifying the development process and reducing the need to switch between different programming languages.
* **Asynchronous I/O:** Node.js utilizes non-blocking, event-driven architecture, allowing it to handle multiple requests concurrently without getting blocked. This asynchronous I/O model enhances the performance and scalability of applications, particularly those requiring high levels of concurrency.
* **NPM (Node Package Manager):** Node.js comes with NPM, the largest ecosystem of open-source libraries and tools, which streamlines the development process by providing easy access to thousands of reusable modules. This extensive repository of packages covers a wide range of functionalities, enabling developers to rapidly build and deploy applications.
* **Scalability:** Node.js is highly scalable, both vertically and horizontally. Its lightweight architecture and non-blocking I/O make it well-suited for building scalable applications capable of handling increasing workloads and traffic demands.
* **Community Support:** Node.js has a vibrant and active community of developers, contributing to its continuous growth and improvement. The community provides extensive documentation, tutorials, and support forums, making it easier for developers to learn and troubleshoot issues.
* **Cross-Platform Compatibility:** Node.js is cross-platform, meaning it can run on various operating systems, including Windows, macOS, and Linux, providing flexibility in deployment environments.
* **Performance:** Node.js offers excellent performance due to its use of the V8 engine and non-blocking I/O model, making it ideal for building high-performance, real-time applications.
* **JavaScript :-**



* **What is JavaScript ?**
* JavaScript is a high-level, interpreted programming language primarily used for creating dynamic and interactive content on websites. Developed by Brendan Eich in 1995, JavaScript has become one of the most widely-used languages for web development due to its versatility and compatibility with all major web browsers. It is a cornerstone of modern web development alongside HTML and CSS.
* JavaScript enables developers to manipulate webpage content in real-time, allowing for interactive features like form validation, animations, dynamic updates, and responsive user interfaces. It can be used both on the client-side (in the web browser) and server-side (with the help of platforms like Node.js), offering flexibility in application development.
* **Advantages of JavaScript :-**
* **Client-Side Interactivity:** JavaScript allows for the creation of interactive user experiences directly within the browser, reducing the need for frequent server requests and enhancing the responsiveness of web applications.
* **Cross-platform Compatibility:** JavaScript is supported by all major web browsers, including Chrome, Firefox, Safari, and Edge, ensuring consistent behavior across different platforms and devices.
* **Rich Ecosystem:** JavaScript has a vast ecosystem of libraries and frameworks such as React.js, Angular, and Vue.js, which streamline development and provide solutions for common tasks, such as state management and UI components.
* **Asynchronous Programming:** JavaScript's asynchronous programming model, based on callbacks, promises, and async/await, enables efficient handling of tasks like fetching data from servers without blocking the main execution thread, enhancing performance and user experience.
* **Dynamic Typing:** JavaScript's loosely-typed nature allows for flexible and rapid development, as variables can dynamically change types during runtime, accommodating diverse programming styles and requirements.
* **Community Support:** JavaScript boasts a large and active community of developers who contribute tutorials, documentation, and open-source projects, making it easy for newcomers to learn and experienced developers to find resources and solutions.
* **Integration with HTML and CSS:** JavaScript seamlessly integrates with HTML and CSS, enabling developers to manipulate DOM elements, apply styles dynamically, and respond to user interactions effectively, creating rich, visually appealing web applications.
* **CSS :-**



* **What is CSS ?**
* Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. It controls the layout, appearance, and formatting of web pages, enabling developers to separate the content from its presentation, making it easier to manage and maintain websites. CSS works by selecting HTML elements and applying various styles to them.
* CSS consists of a set of rules that define how the content of a web page should be displayed. These rules include properties such as color, font, size, spacing, and positioning. CSS can be applied to HTML elements directly within the HTML document, or it can be linked to an external CSS file, which is the preferred method for larger projects as it promotes consistency and easier maintenance.
* **Advantages of CSS :-**
* **Separation of Concerns:** CSS enables the separation of content (HTML) from presentation (styling), and behavior (JavaScript), facilitating cleaner and more maintainable code. This modular approach allows developers to focus on one aspect of the website without interfering with others, leading to better organization and scalability.
* **Consistency:** With CSS, developers can define styles once and apply them consistently across multiple pages or the entire website. This ensures a uniform appearance and user experience, enhancing brand identity and professionalism.
* **Accessibility:** CSS provides tools for creating accessible web content by allowing developers to control the presentation of elements based on factors such as screen size, device type, or user preferences. This ensures that content is readable and usable by a wider audience, including those with disabilities.
* **Faster Loading Times:** By separating style information into external CSS files, web browsers can cache these files, reducing the need to re-download stylesheets for each page. This results in faster loading times and improved performance, especially on subsequent visits to the same website.
* **Flexibility and Control:** CSS offers a wide range of styling options, including layout techniques like flexbox and grid, as well as animations and transitions. This flexibility allows developers to create visually appealing designs and responsive layouts tailored to different devices and screen sizes.
* **MongoDB :-**



* **What is MongoDB ?**
* MongoDB is a popular, open-source NoSQL database management system designed to handle large volumes of unstructured or semi-structured data. It belongs to the class of document-oriented databases, meaning it stores data in a JSON-like format called BSON (Binary JSON). MongoDB is renowned for its flexibility, scalability, and high performance, making it a preferred choice for modern application development.
* One of MongoDB's key features is its dynamic schema design. Unlike traditional relational databases, which require a predefined schema, MongoDB allows for dynamic schema evolution. This flexibility enables developers to store heterogeneous data types within the same collection and adapt the schema as application requirements change over time, facilitating agile development and faster time-to-market.
* Another notable aspect of MongoDB is its horizontal scalability. By distributing data across multiple nodes in a cluster, MongoDB can handle massive datasets and high-throughput workloads with ease. Its built-in sharding capabilities enable automatic partitioning of data across shards, ensuring efficient data distribution and fault tolerance.
* MongoDB offers a rich query language and powerful indexing capabilities, allowing developers to perform complex queries and efficiently access data. Additionally, it provides support for geospatial queries, text search, and aggregation framework, empowering developers to derive insights from diverse datasets.
* In terms of deployment options, MongoDB can be deployed on-premises, in the cloud, or as a managed service through MongoDB Atlas. MongoDB Atlas offers automated provisioning, scaling, and monitoring of MongoDB clusters, simplifying the management of database infrastructure and reducing operational overhead.
* Security is a top priority for MongoDB, and it offers a comprehensive set of features to ensure data confidentiality, integrity, and availability. These include authentication mechanisms, role-based access control, encryption at rest and in transit, auditing, and compliance certifications.
* Overall, MongoDB's combination of flexibility, scalability, performance, and ease of use makes it a compelling choice for a wide range of applications, from content management systems and e-commerce platforms to real-time analytics and Internet of Things (IoT) solutions. Its vibrant community, extensive documentation, and active development ecosystem further contribute to its popularity and widespread adoption in the software industry.

1. **Planning**

* **System Planning**
* **Time Line Chart**
* **System Flowcharts**
* **System Planning :-**
* System planning in website development refers to the initial phase of the project where the objectives, scope, requirements, and architecture of the website are defined. It involves thorough analysis and decision-making to ensure that the website meets the needs of its target audience and aligns with the overall goals of the organization or business.
* **During the system planning phase, several key activities take place:**
* **Requirement Analysis:** This involves gathering and documenting the functional and non-functional requirements of the website. Functional requirements specify what the website should do, such as user authentication, content management, and e-commerce functionality. Non-functional requirements pertain to qualities like performance, security, scalability, and usability.
* **Stakeholder Engagement:** It's crucial to involve stakeholders, including clients, end-users, marketing teams, and developers, to understand their perspectives, preferences, and expectations for the website. Stakeholder feedback helps in shaping the project's direction and ensures that the final product meets the needs of all involved parties.
* **Goal Setting:** Clear goals and objectives for the website are established during the planning phase. These goals could include increasing online sales, improving brand visibility, enhancing user engagement, or providing better customer support. Setting measurable goals helps in evaluating the success of the website once it's launched.
* **Scope Definition:** The scope of the website project is defined, outlining what features, functionalities, and content will be included in the final product. It's essential to balance the scope with available resources, budget, and timeline constraints to ensure a realistic and achievable project plan.
* **Technology Selection:** Based on the project requirements and objectives, appropriate technologies and platforms are selected for website development. This includes decisions on programming languages, content management systems (CMS), frameworks, hosting solutions, and third-party integrations.
* **Information Architecture (IA):** IA involves organizing and structuring the content and navigation of the website to ensure intuitive user experiences. This includes creating sitemaps, wireframes, and prototypes to visualize the website's structure and layout.
* **Risk Assessment:** Potential risks and challenges that could impact the success of the project are identified and assessed. This allows for proactive risk mitigation strategies to be developed and integrated into the project plan.
* **Existing System :**
* Linkedin.com
* Indeed.com
* Upworks.com
* Noukri.com
* jobDekho.com
* **Feasibility Study :** The feasibility of our project has been judged on the time, technology, resources available, and project length.
* **Time**:- This project takes at least 2 months to be completed if we take the help of reused components; otherwise it may take more than 5 months to be completed.
* **Technology**:- This project is built on React.js & Node.js platform which is worldwide already standard. Therefore the technical risk is not so high hence feasible
* **Resources**:- Good technology professionals are a basic requirement. Internet and favorable environment, etc. the resources were fully available.

**1. Economic Feasibility :** The prime requirement of this project is an internet connection as the project been developed is a Website. Moreover, it also demands a good server for data redirection and high speed and efficient CPU ram and processing and managing the online transaction done by the online Jobseeker & Recruiter

**2. Technical Feasibility :** This website requires high and up-to-date technology. It needs database feedback and also needs to be accessed through the web and the internet. This can be done easily, so online payment should be facilitated. We provide our clients with the right worker for their job. So that clients can do their work easily and with proper use of their time. We provide our workers with the right clients for their job. So that worker can do their work easily and with proper use of their time.

**3. Management Feasibility :** Management feasibility ratio and aspect of management news. Here all level of management rise of related basic feasibility and gives their decision for all those feasibilities to use or not is comes in basic feasibility of the system, bus how the requirement of that all, after that all manager.

**4. Operational Feasibility :** The web application can be beneficial only if it satisfies the organization’s requirement; in such a way that resource utilization & the optimum outcome is justified. A web application should not only be robust but should also be able to work simultaneously with other systems. Operational feasibility means that web applications should not affect any existing system during the development phase or even in the implementation phase.

**5. behavioural feasibility:**

The new system can be beneficial only if it satisfies the organization’s requirements. In such a way that resource utilization and the optimum outcome are justified. A new system should not only be robust but should also be able to work simultaneously with other systems. Operational feasibility means that a new system should not affect any existing system during the development phase or even in the implementation phase. Following are some points underlying the operational feasibility of the system. As the development proceeded many doubts got clear doubt.

• Our project guide- Asst. Prof. Pratiksha Patel guided us to take proper care and check for the operational feasibility of the system.

• Efforts were made to optimize the human efforts in data collection, storage, retrieval, security, and presentation.

• The proposed system made its best efforts in achieving the necessary function and performance. As required by the user and keeping in mind some infrastructure constraints. Some of the following issues raised are to test the operational feasibility of a project including the following.

• Is there sufficient support for the management from the user?

• Will the system be used and work properly if it is being developed and implemented?

• Will there be any resistance from the user that will undermine the possible application benefits?

• This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration so there is no question of resistance from the user that can undermine the possible application benefits.

**Requirement Gathering and Analysis:-**

* **Requirement Gathering :**
* Before developing an elegant application, the best way to identify basic needs and functionality is to take an interview with the director and management who are working in that specific area addressed by the proposed application.
* For gathering basic needs and functionalities, we had a meeting with people personally to understand required functionalities and also noted down some useful features to be included.
* Every successful system passes through requirement gathering because without any requirement the development process is like adding bugs to an empty system.
* Requirement gathering and analysis make the whole picture of the system and because of that, we can identify the features, the modules, and the functionality of the system.
* **Requirement Analysis :**
* System should be able to provide a registration facility.
* System should provide a safe login facility.
* System should provide forgot password facility.
* System should manage email Contact, skills.
* We analyzed our gathered information and we have decided our system should have the following functionality:-
  + **Common models**
    - Registration
    - Login
    - Forget Password (User )
    - Logout
* Module 1 ( USER )
  + User Registration.
  + User Login / Logout.
  + User Change Password.

### Module 2 ( Company/Admin )

* + Company/Admin Login.
  + Company/Admin Logout.
  + Company/Admin Send Email To Applicient.
  + Company/Admin add policy
  + Company/Admin Delete policy.
  + Company/Admin View their connected seekers.
  + Company/Admin Edit policy.
  + Company/Admin Delete policy.
  + Company/Admin Job Posting policy.

* **Project Scope:-**

In the current project of WEB ANALYSIS, we have undertaken, the corresponding scope of our intranet website is as follows: For

This is a user-friendly layout & easy to understand for every user or service provider.

* It will allow the user to recover the password.
* It Also allows Service Providers to recover password
* It is able to Give information about registered Users and also edit that information.
* It allows sending mails.
* This system provides facility to user to send job applications with resume uploading facility for all companies.
* It provides the searchin facility so the seekers can easily find the location of the company and contact it and Applies in to the companies.
* It also provide the job posting facility for companies with dashboard facility – in this dashboard company can handle all the application from job seeker which Is applied by seekers and can hire them with email sending facility
* .pdf file download and view facility of resume
* Project Modules:-

1. User
2. Admin/Company

* Modules Wise Objectives/functionalities Constraints:-

The main objective of the job portal platform is to Provide a Service to find job opportunity to the customer.

1. Admin/Company

* Admin can upload/delete/edit jobs for the job seekers with dashboard facility
* Admin/company can manage Applications
* Admin/company can manage the insurance policy

2.User

* Users can register Itself
* Users can log in
* Users can view the jobs
* Users can find the jobs with filtering facility
* Users get a emails from company
* Users can change their password
* Expected Advantages:-
* **High Security :** There is a high-security mechanism provided by the proposed system. No unauthorized person can make a change in the data that are stored in the application. Only an authorized person can change & also can make a login. Also, systems provide two-step authentication for security reasons.
* **Easy and Fast Response :** All user’s data are stored in an online centralized database. So, users access data from anywhere and on any device. No need for any other device to run the application on the device. Separate login for separate users like faculty, student, and admin.
* **High Amount of Accuracy :** In our application there are less amount of human activities are involved in that case which will have less effect on our accounts. Before the Insurance Company registers, we can verify the insurance company details for security.
* **Easy to use:** jobDuniya is giving an easy and responsive interface for mobile and tablet to online insurance services through the web.
* **System Flow Chart :-**

**START**

**SIGN IN**

**SIGN UP**

**COMAPANY**

**APPLY NOW**

**PROFILE**

**ABOUT US**

**HOME PAGE**

**SEARCH PAGE**

**APPLY JOB**

**NOTIFICATION**

**SIGN IN**

**JOB SEEKER**

**REJECT**

**DETAILS**

**APPROVE**

**DECLINE**

**COMAPANY**

**PROFILE**

**APPLICATION**

**SIGN UP**

**VIEW APPLIED JOB**

**DASHBORAD**

**SEARCH**

**CANDIDATE PROFILE**

**NOTIFACTION**

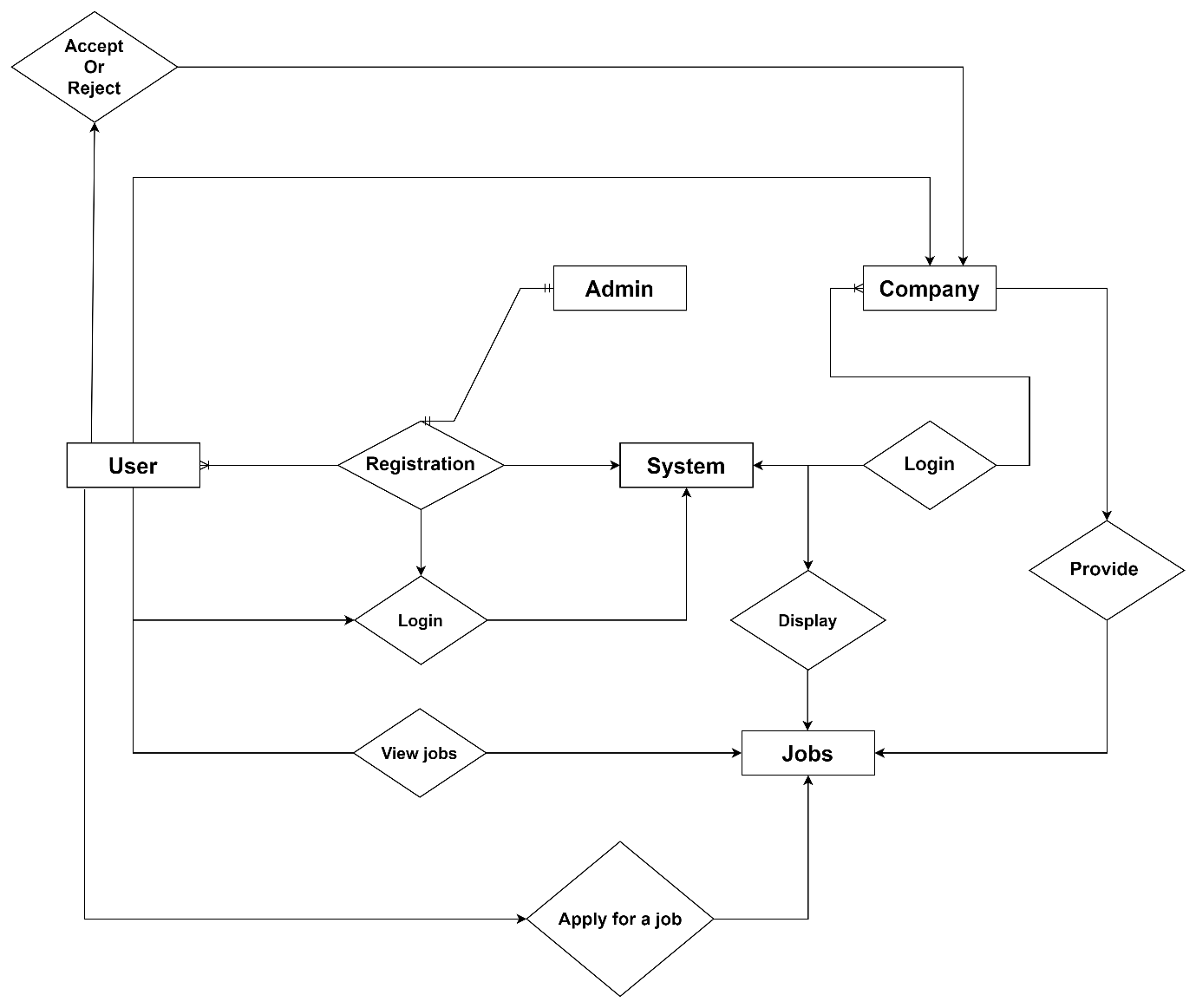
**ABOUT US**

**COMAPANY**

**PROFILE**

1. **System Analysis**

* **ER - Diagram**
* **Data Flow Diagram**
* **Data Dictionary**
* **Entity Relationship Diagram:-**



* **Data Flow Diagram :-**
* A DFD (Data Flow Diagram) is a graphical representation of the flow of data within a system. It illustrates how data moves between processes, data stores, and external entities in a system. DFDs are commonly used in system analysis and design to model the data flow and transformations that occur within a system or between systems.
* **DFDs are beneficial in the following ways:**

Communication: DFDs provide a common language for communication between stakeholders, including business users, analysts, designers, and developers, helping ensure a shared understanding of the system's requirements and design.

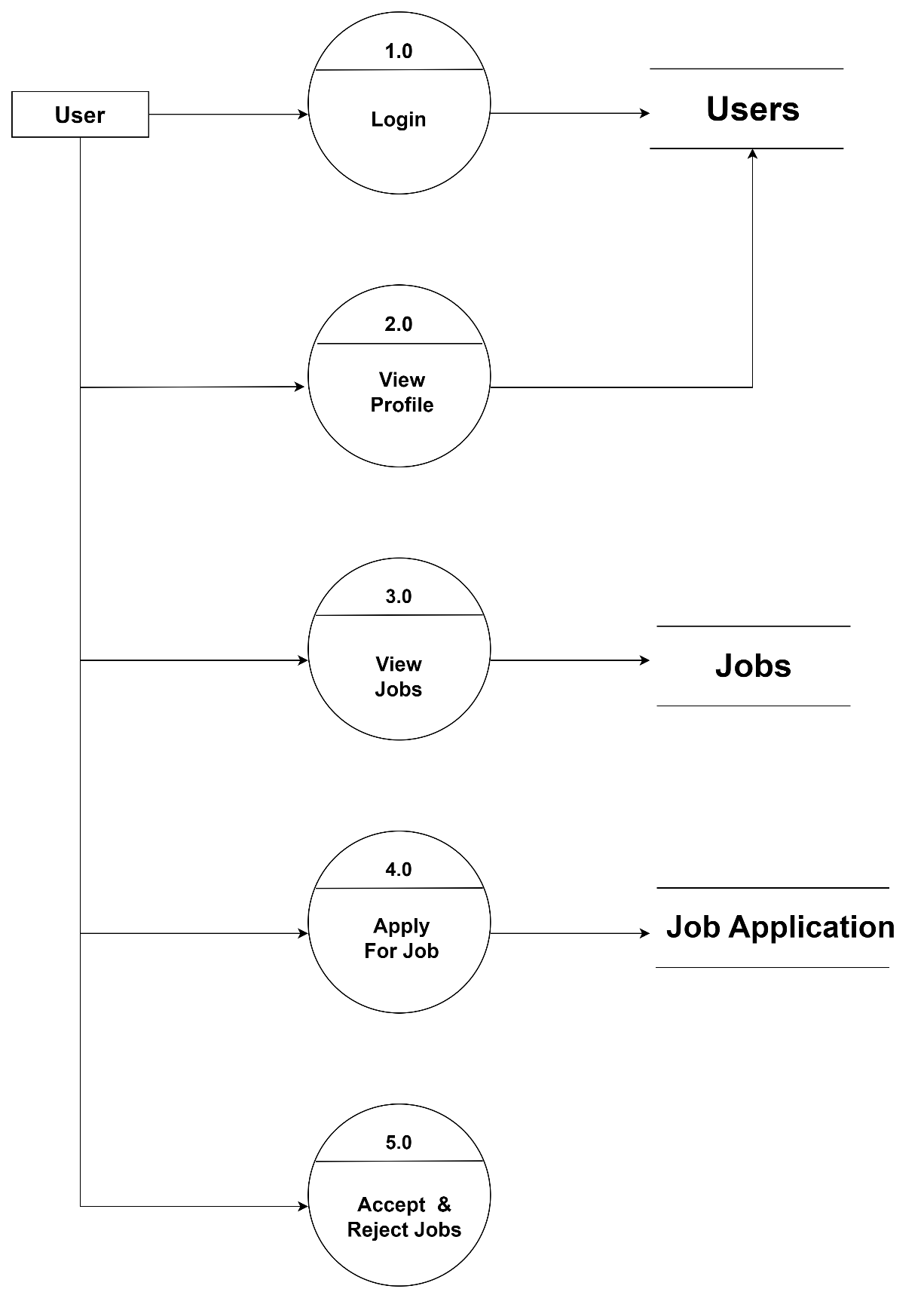
Analysis and Design: DFDs facilitate the analysis and design of systems by identifying data flows, processes, and data stores, which serve as a basis for designing system architectures, interfaces, and data structures.

Problem Identification: DFDs help in identifying potential problems such as data redundancies, bottlenecks, or inconsistencies in data flow, enabling analysts to propose solutions to improve system efficiency and reliability.

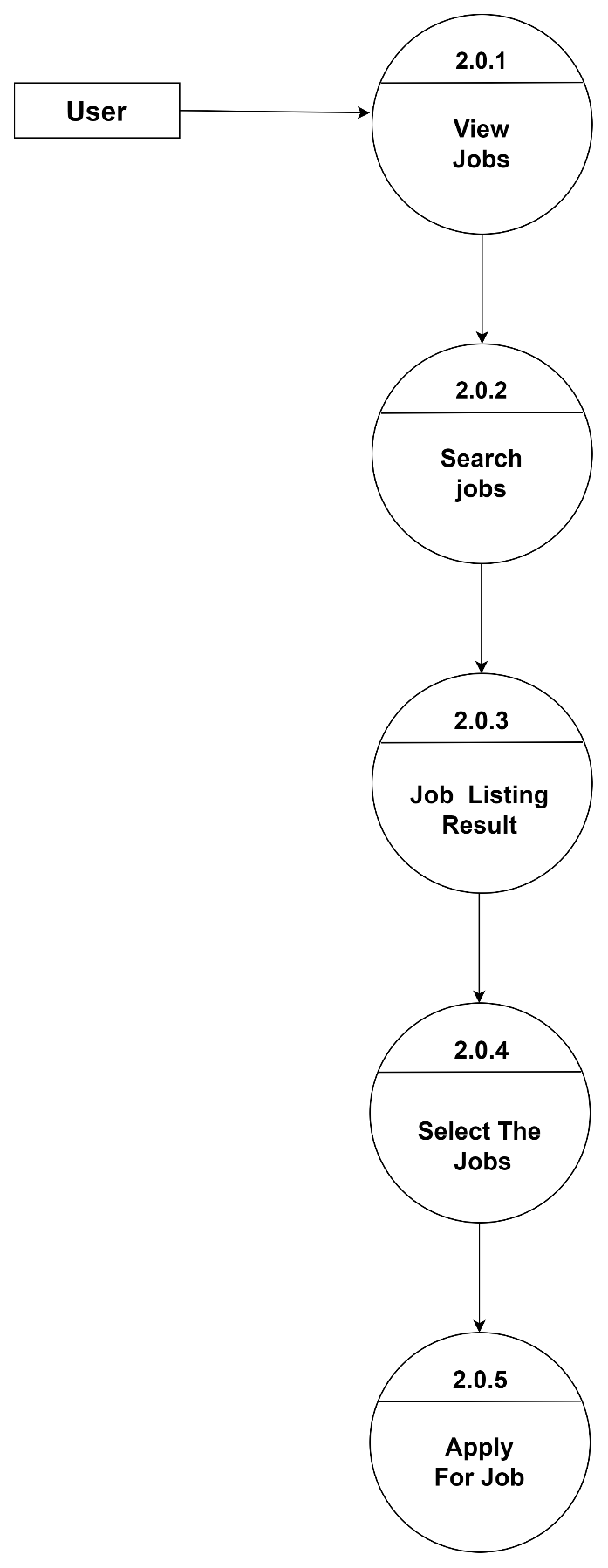
**ZERO LEVEL DFD – JOB DUNIYA**



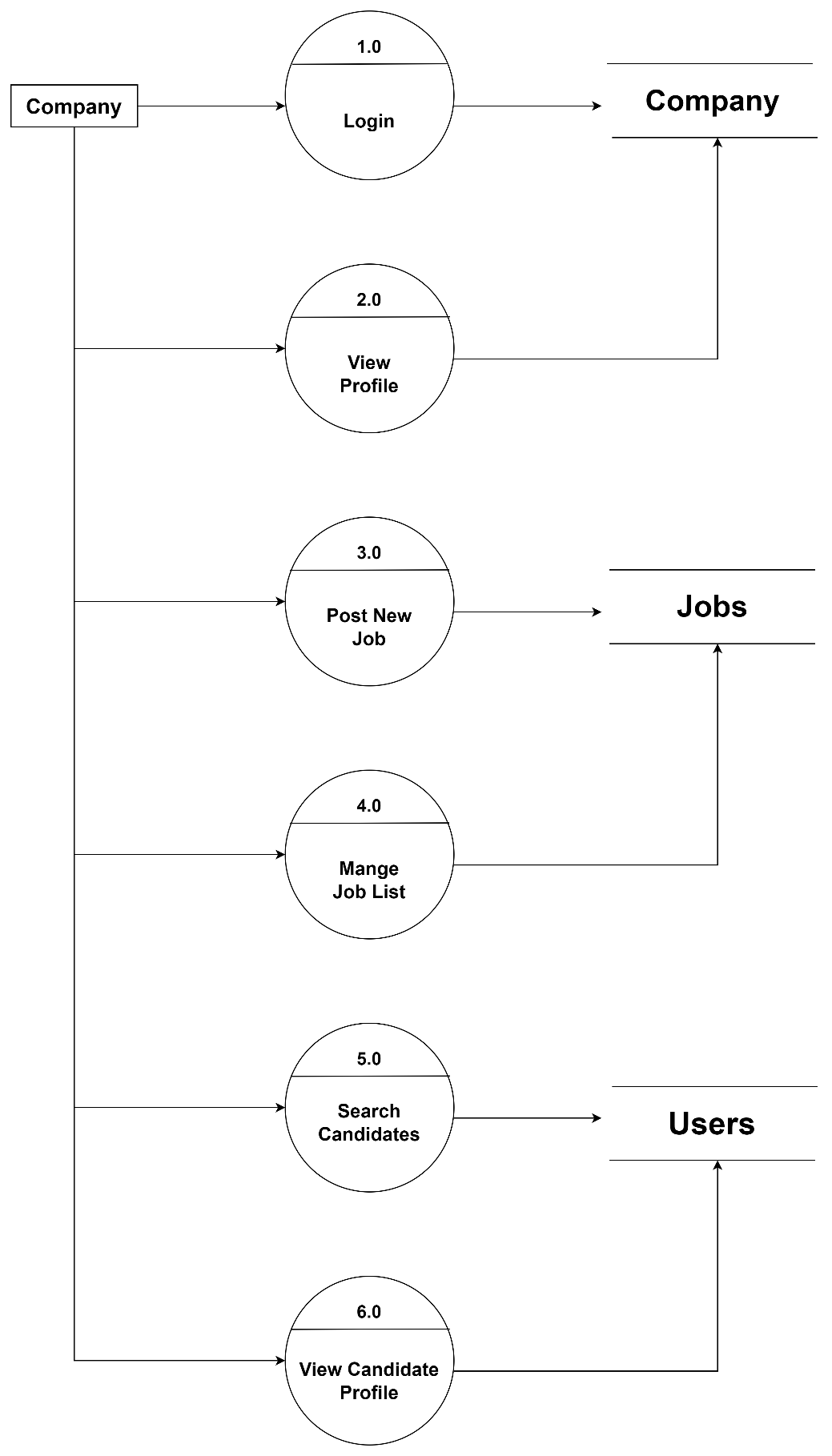
**FIRST LEVEL DFD FOR USER**



**SECOND LEVEL DFD FOR USER**



**FIRST LEVEL DFD FOR COMPANY**



* **Data Dictionary :-**

* **Data Dictionary:-**
* **User:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Characteristics | Description |
| \_id | Autogenerate | Only number allow | \_id is the unique id of user data. |
| Email | Email form | Only email allow | It displays the user’s email. |
| Password | Password (encrypted) | password | It displays the user’s password. |
| firstName | String | Only string allow | It displays the user’s first name. |
| lastName | String | Only string allow | It displays the user’s last name. |
| profileImage | String | Only String (Url) allow | It displays the user’s profile Image. |
| registrationDate | String  (Autogenerate) | Date | It displays the user’s Registration Date. |
| languages | Array[String] | Only string allow | It displays the user’s Languages. |
| cv | String | Only String (Url) allow | It displays the user’s CV. |
| Skills | Array[String] | Only string allow | It displays the user’s Skills. |
| profession | String | Only string allow | It displays the user’s Profession. |
| Description | String | Only string allow | It displays the user’s Description. |
| Location | Array[String] | Only string allow | It displays the user’s  Location. |
| Education | Array[String] | Only string allow | It displays the user’s Education. |
| experience | Array[String] | Only string allow | It displays the user’s Experience. |

* **Company:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Characteristics | Description |
| \_id | Autogenerate (ObjectId) | Only number allow | \_id is the unique id of Company data. |
| Name | String | Only string allow | It displays the Company’s Name. |
| Address | Array  [String] | Only string allow | It displays the Company’s Address. |
| Industry | Password (encrypted) | Only string allow | It displays the Company’s Industry |
| Email | String | Only email allow | It displays the Company’s Email |
| Password | String | password | It displays the Company’s Password |
| Logo | String | Only String (Url) allow | It displays the Company’s Logo |
| TagLine | String | Only string allow | It displays the Company’s Tag Line |
| Website | Array[String] | Only string allow | It displays the Company’s Website |
| Established  Year | String | Date | It displays the Company’s Establish Year. |
| Description | Array[String] | Only string allow | It displays the Company’s Description. |
| secretKey | String | - | It displays the Company’s Secret Key. |
| Owner  Details | Array[String] | Only string allow | It displays the Company’s Owner Details. |
| HRDetails | Array[String] | Only string allow | It displays the Company’s HR Details. |

* **Jobs:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Characteristics | Description |
| \_id | Autogenerate  (ObjectId) | Only number allow | \_id is the unique id of Job’s data. |
| Title | String | Only string allow | It displays the Job’s Title. |
| Position | String | Only string allow | It displays the Job’s Position. |
| JobPostedTime | Date  (Autogenerate) | Date | It displays the Job’s Posted Time. |
| Description | String[Array] | Only string allow | It displays a Job’s Description. |
| Experience | String[Array] | Only string allow | It displays Job’s Experience. |
| JobType | String | Only string allow | It displays Job’s Type. |
| Salary | String | Only number allow | It displays Job’s Salary. |
| Responsibility | String[Array] | Only string allow | It displays Job’s Responsibility. |

* **JobApplication:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Characteristics | Description |
| \_id | Autogenerate  (ObjectId) | Only number allow | \_id is the unique id of Job Application data. |
| userId | ObjectId(Ref : ‘users’) | Only number allow | It displays the Unique Users\_Id. |
| jobId | ObjectId(Ref : ‘jobs) | Only number allow | It displays the Unique Jobs\_Id. |
| Cv | String | Only String (Url) allow | It displays the job Application cv. |
| cId | ObjectId(Ref : ‘company) | Only number allow | It displays the Unique Company\_Id. |
| phoneNum | String | Only number allow | It displays the User’s Phone Number. |
| Email | Email Form | Only email allow | It displays the user’s Email. |
| applyDate | Date | Date | It displays the applied Job Date. |

* **SavedJobs:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Characteristics | Description |
| \_id | Autogenerate  (ObjectId) | Only number allow | \_id is the unique id of Saved Job’s data. |
| Status | String | Only String  allow | It displays the Saved jobs Status. |
| userId | ObjectId(Ref : ‘users’) | Only number allow | It displays the Unique Users\_Id. |
| jobId | ObjectId(Ref : ‘jobs’) | Only number allow | It displays the Unique Jobs\_Id. |

* **UserFollow:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Characteristics | Description |
| \_id | Autogenerate  (ObjectId) | Only number allow | \_id is the unique id of userFollow data. |
| userId | ObjectId(Ref : ‘users’) | Only number allow | It displays the unique User\_id. |
| targetId | ObjectId(Ref : ‘users) | Only number allow | It displays the Unique User id. |

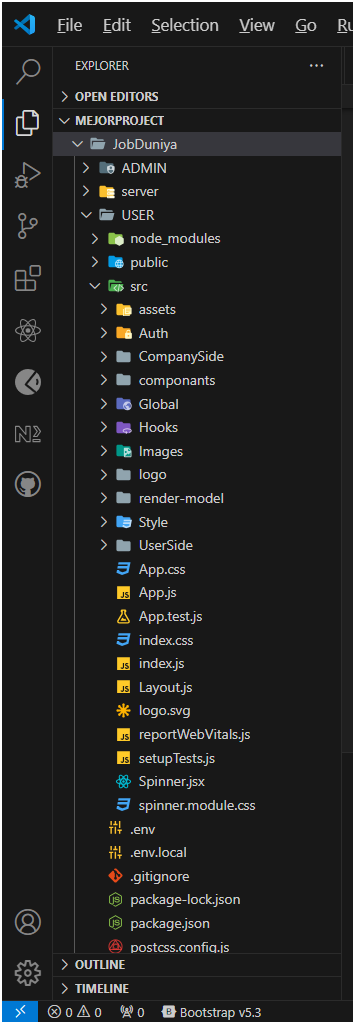
* **CompanyConnections:-**

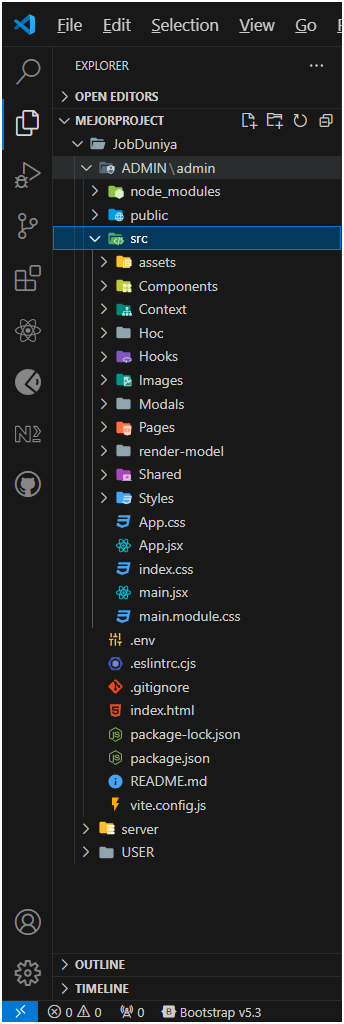
|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Characteristics | Description |
| \_id | Autogenerate  (ObjectId) | Only number allow | \_id is the unique id of company connection data. |
| userId | ObjectId(Ref : ‘users’) | Only number allow | It displays the unique User\_id. |
| targetId | ObjectId(Ref : ‘users) | Only number allow | It displays the Unique Company id. |

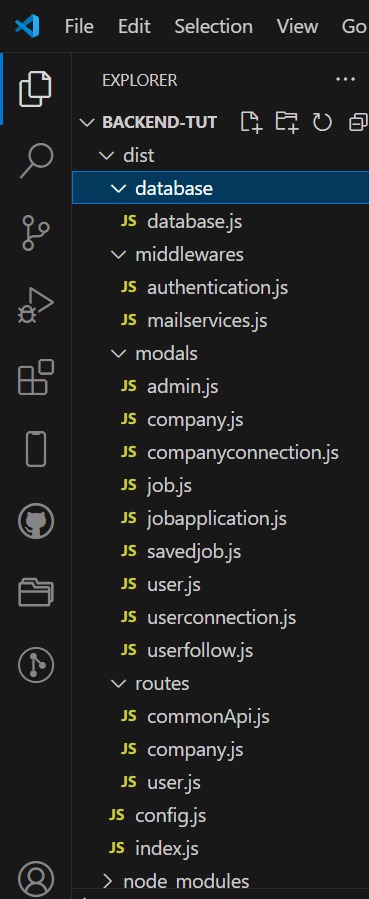
1. **Designing**

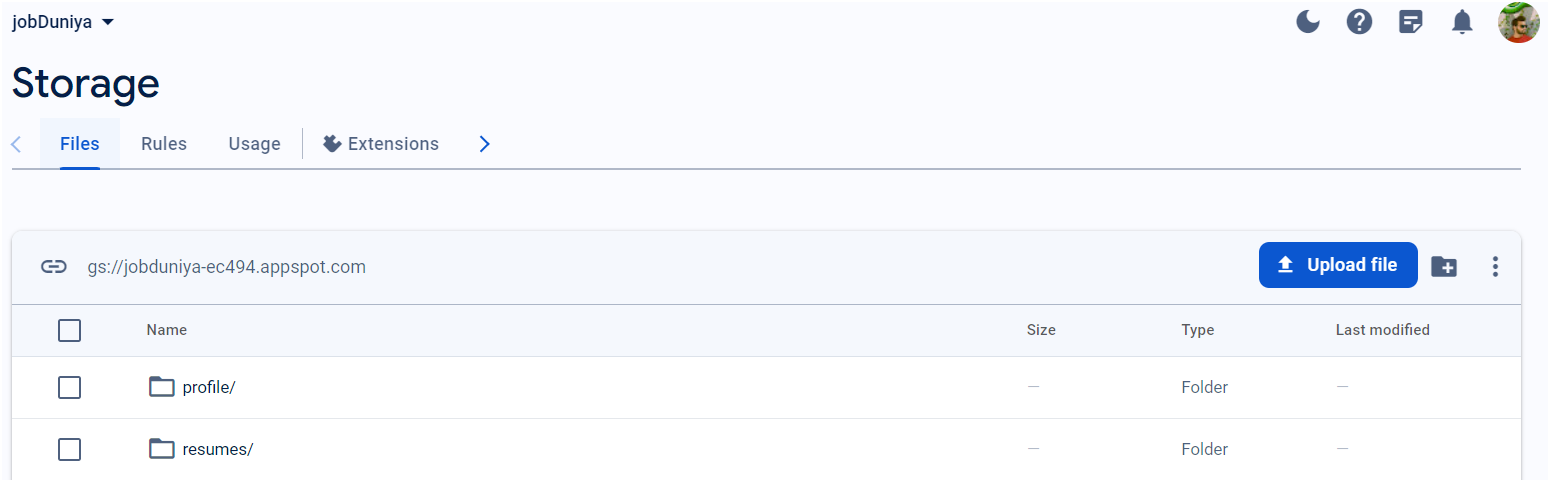
* **Directory Structure**
* **User interface**

Directory Structure



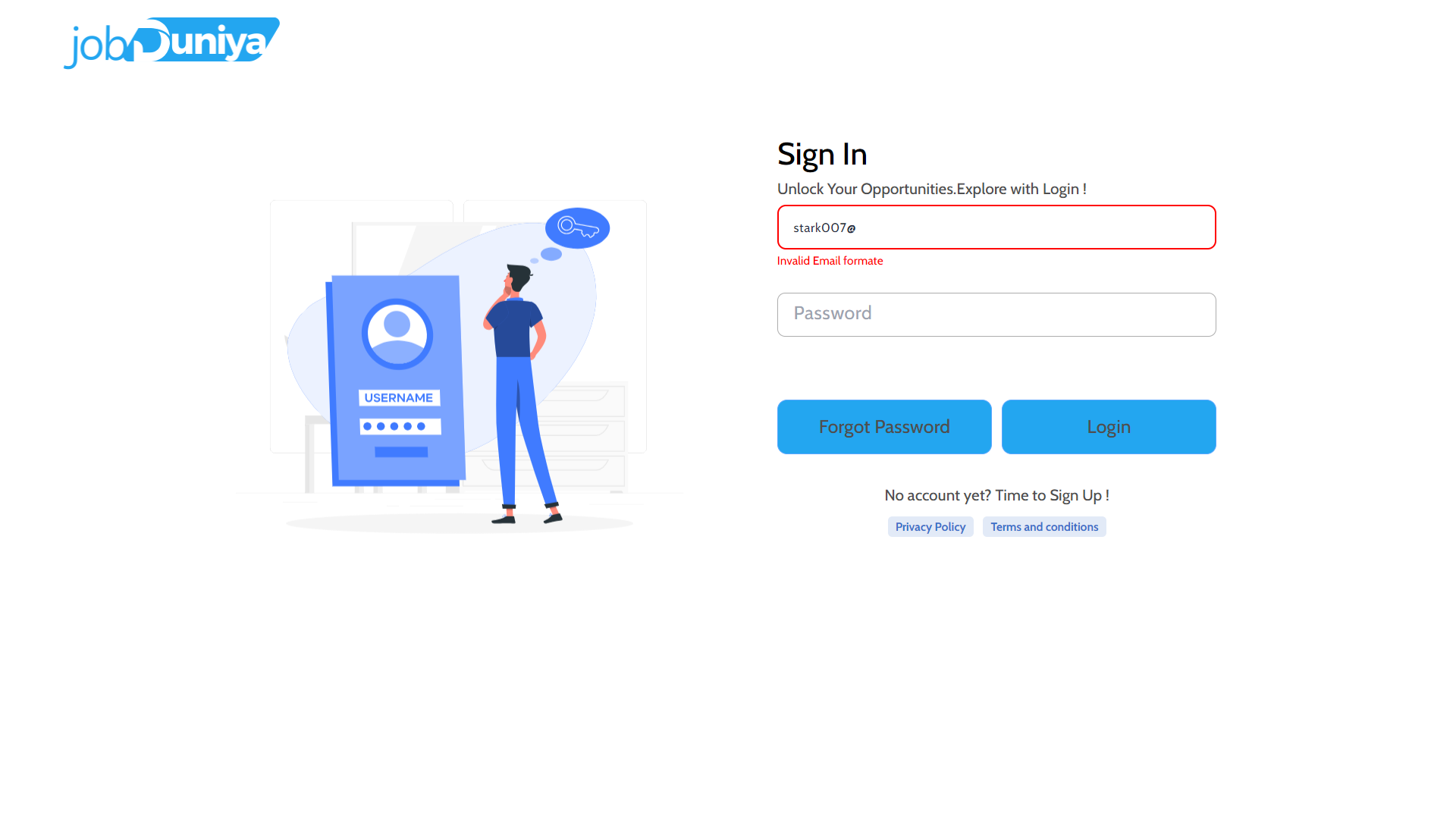
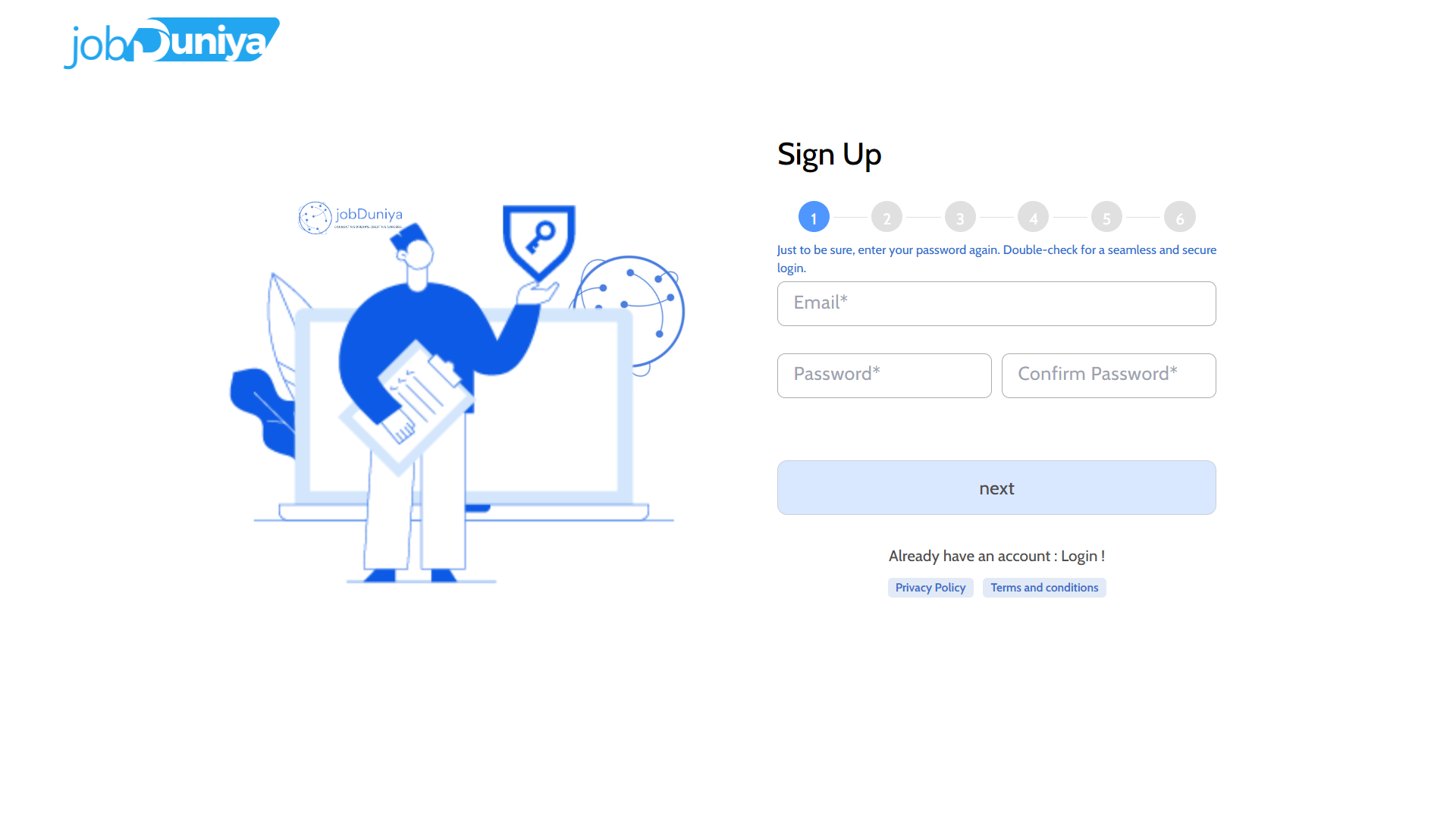
****

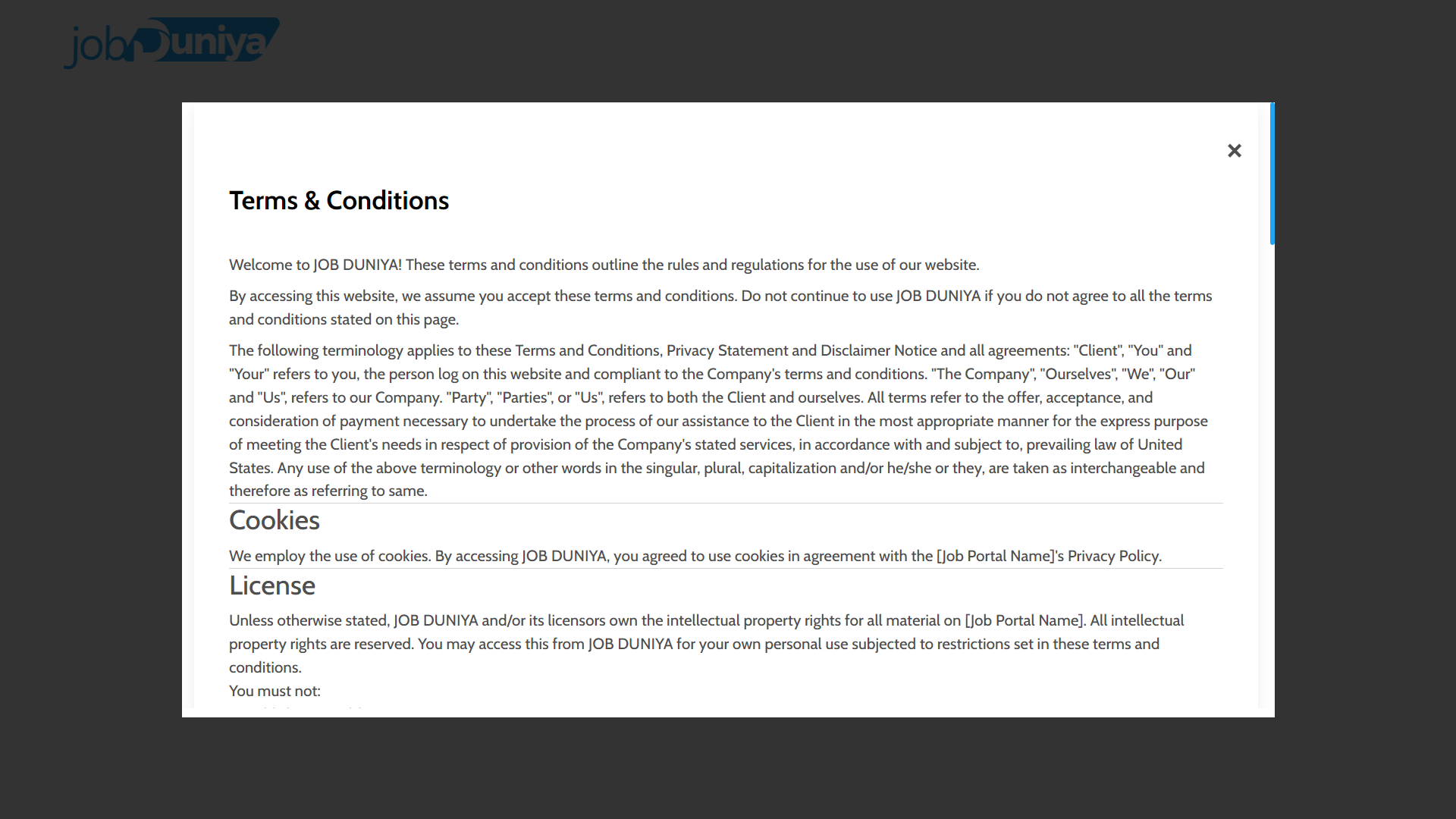


****

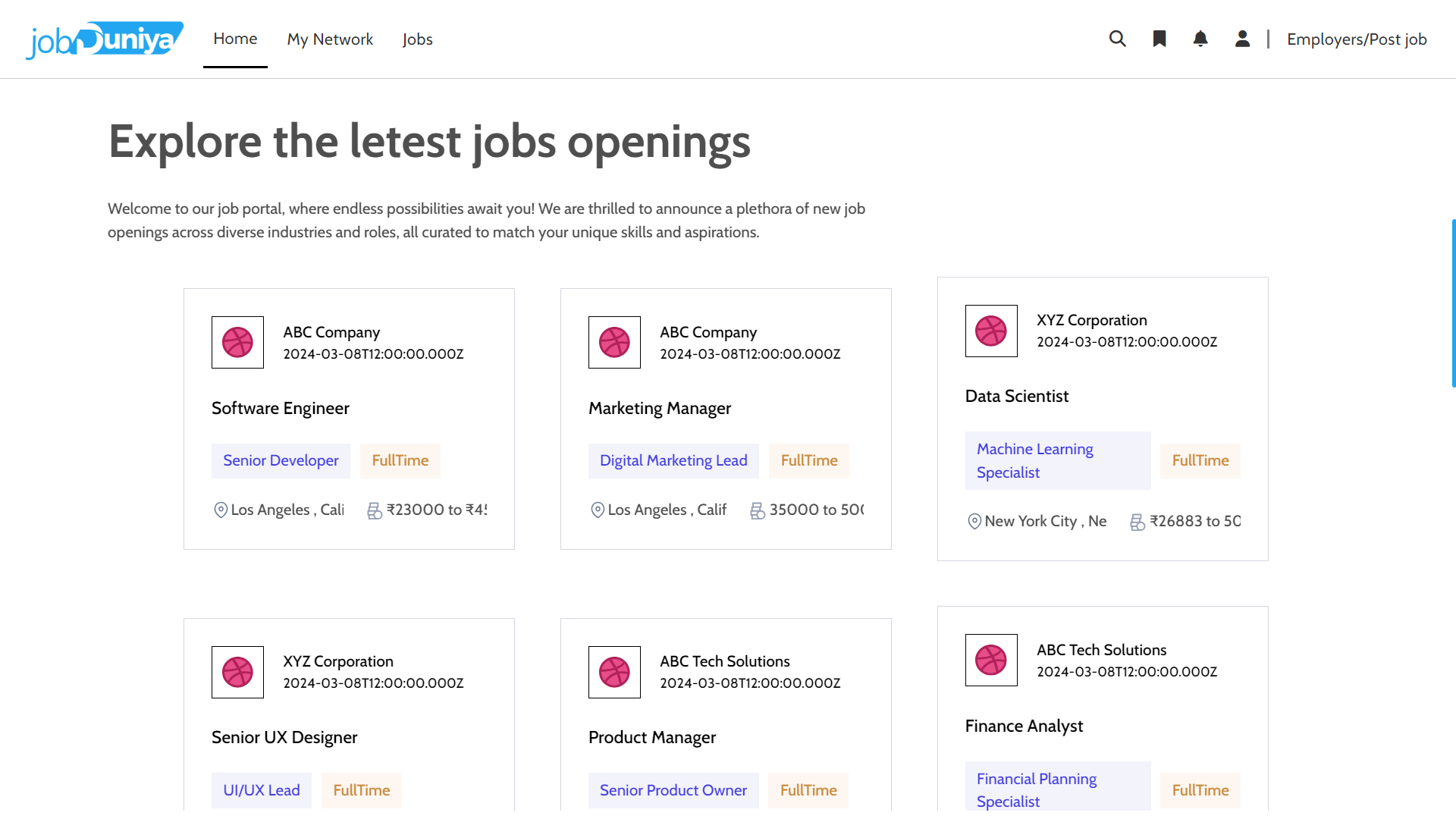
User Interface

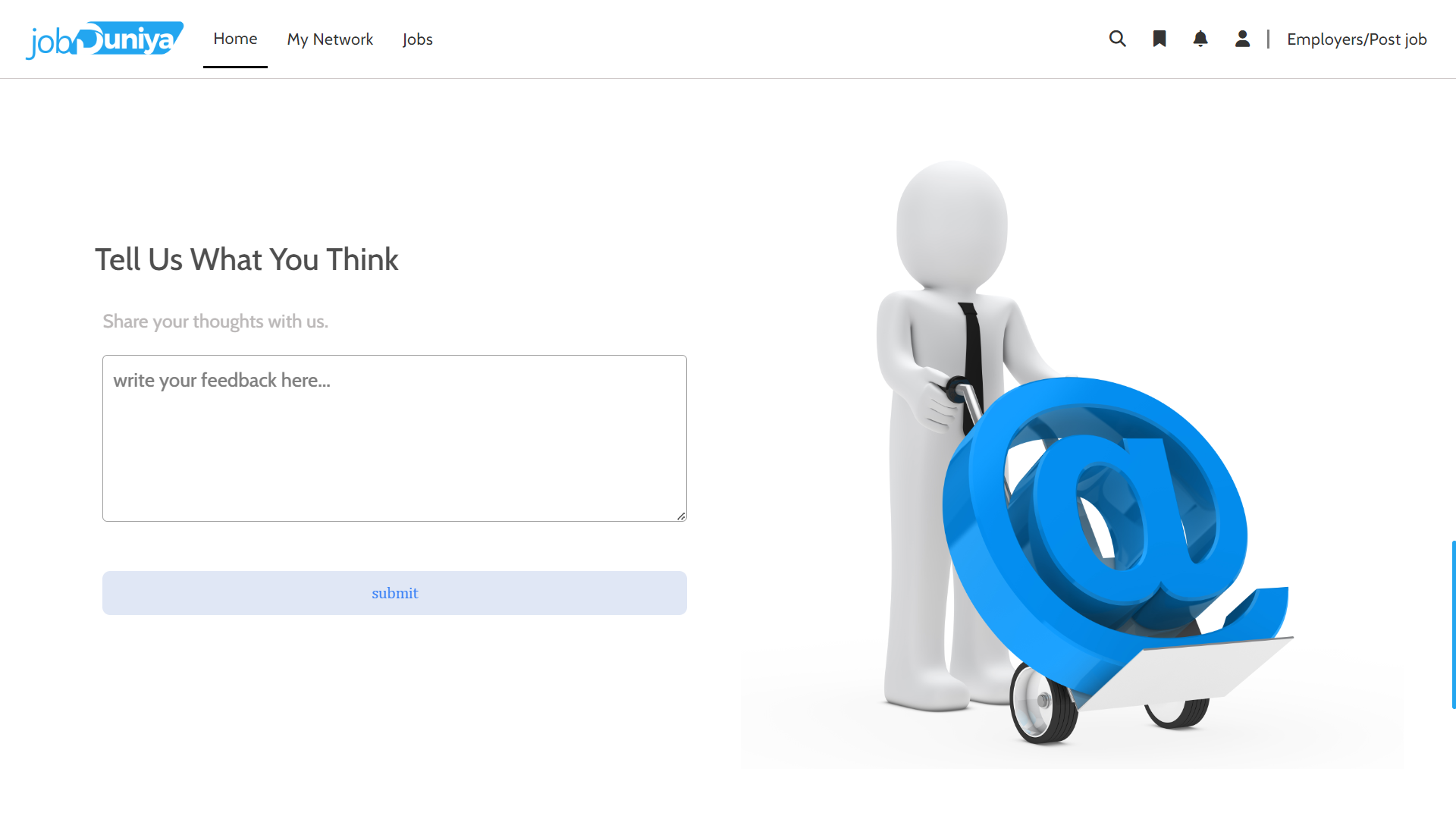
* User-side :-

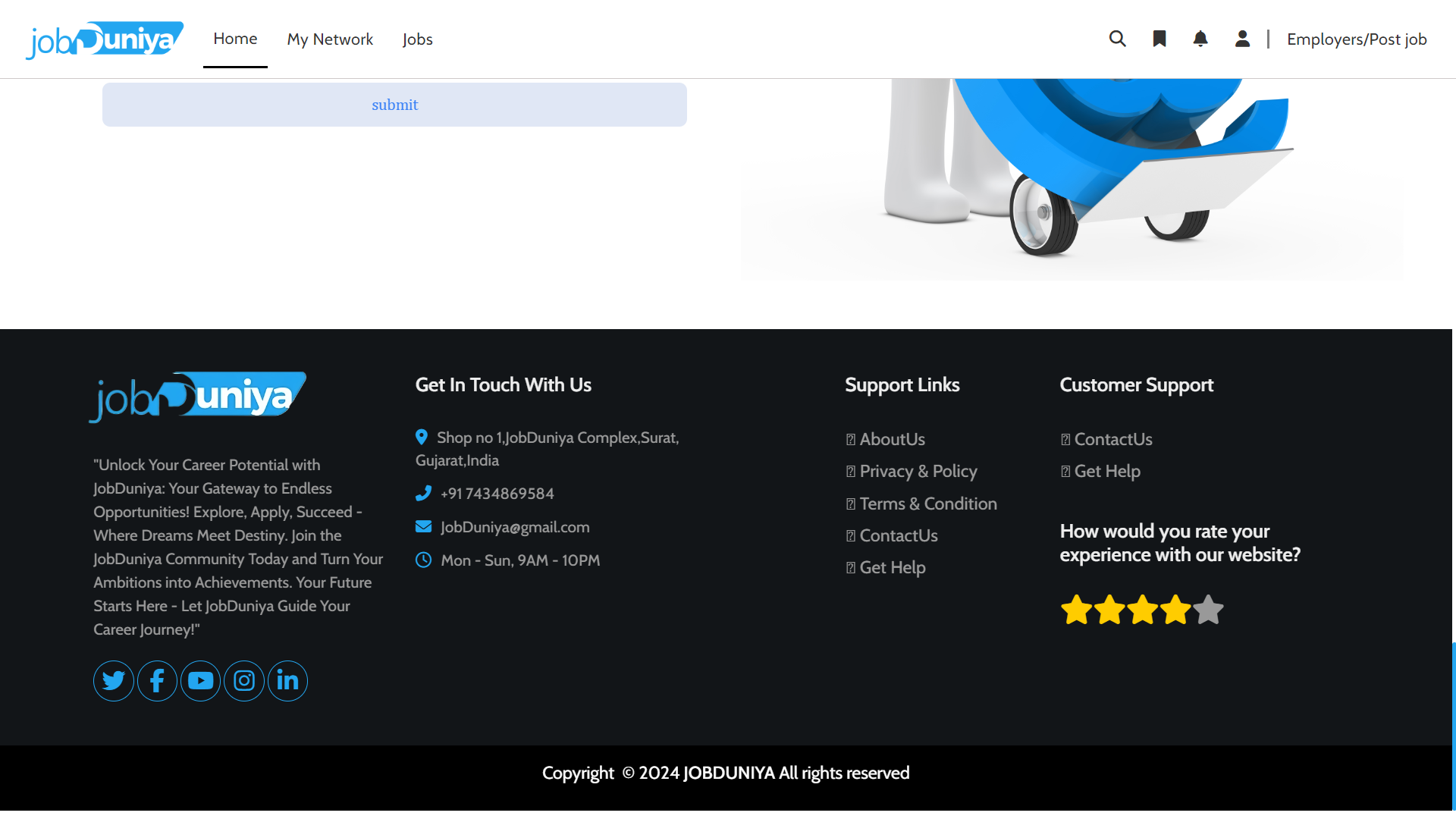
 

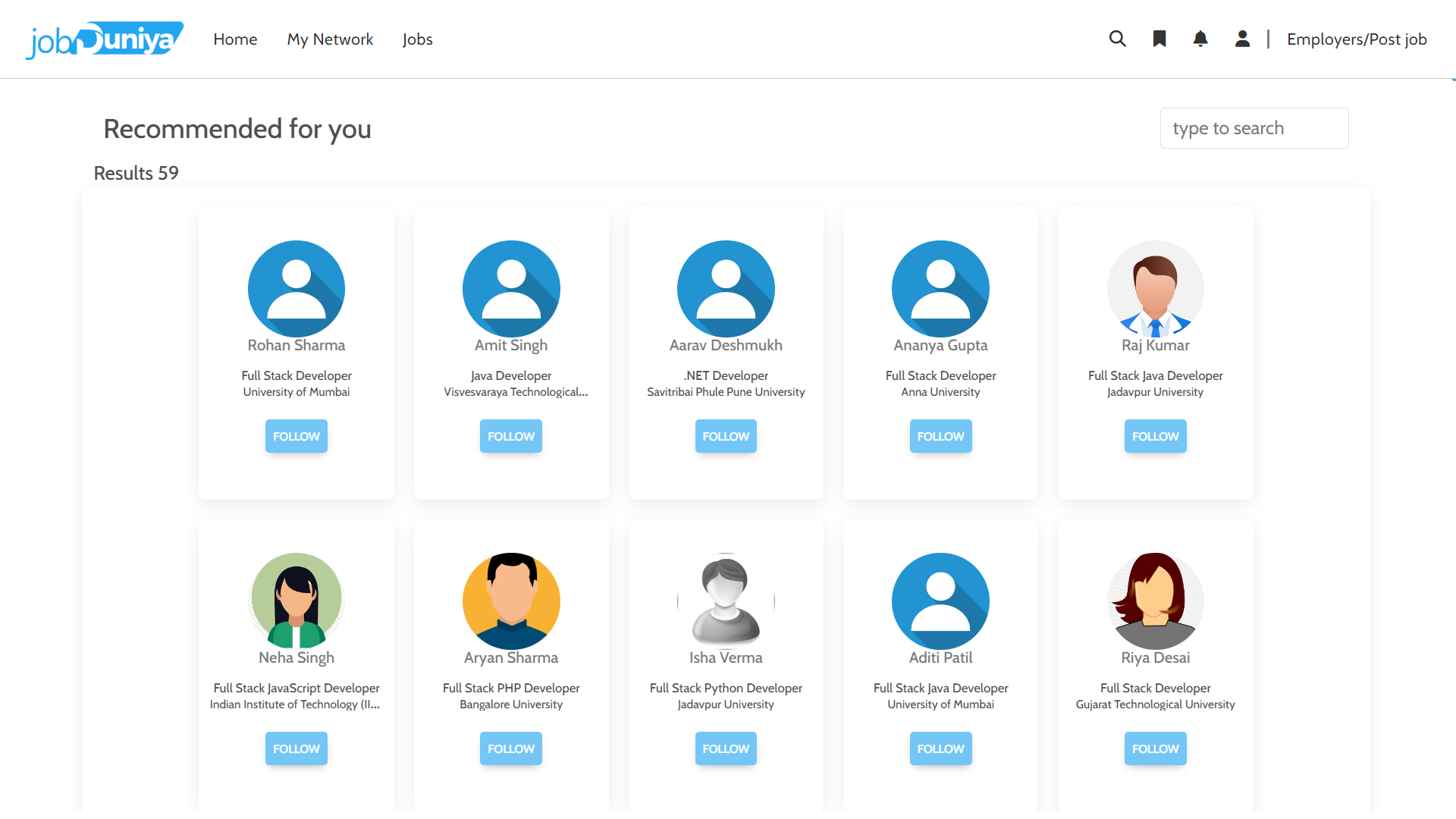


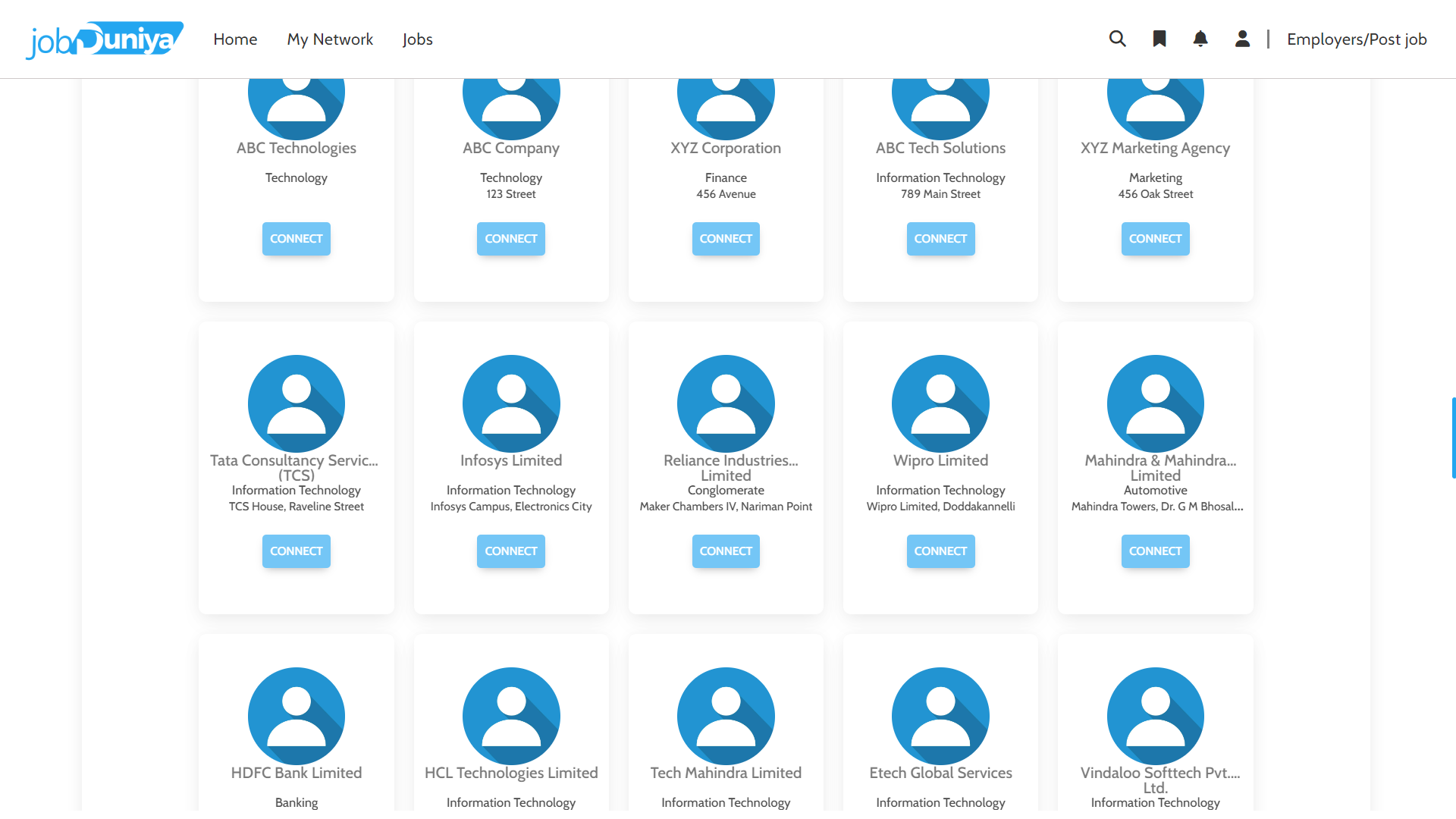


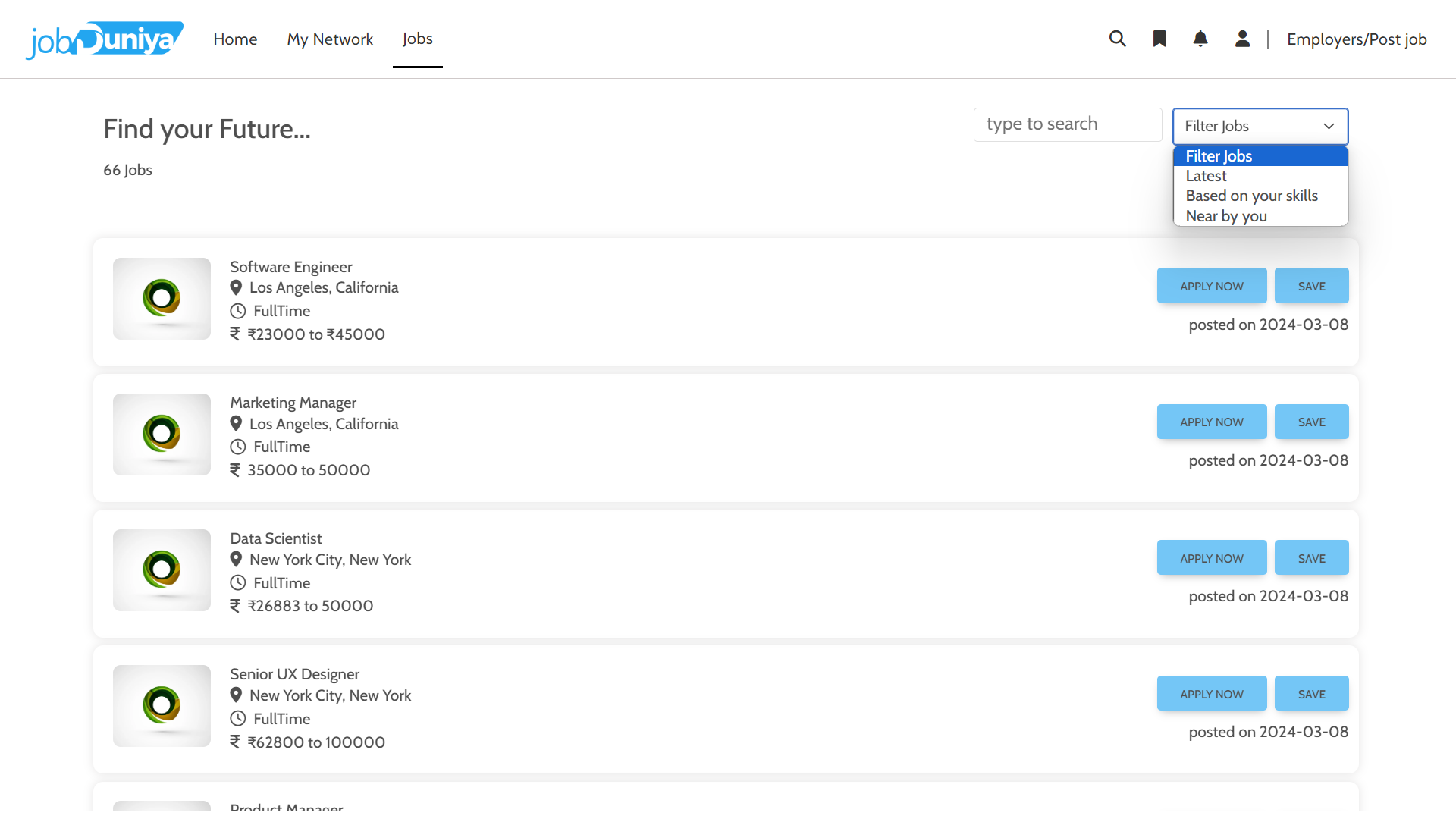


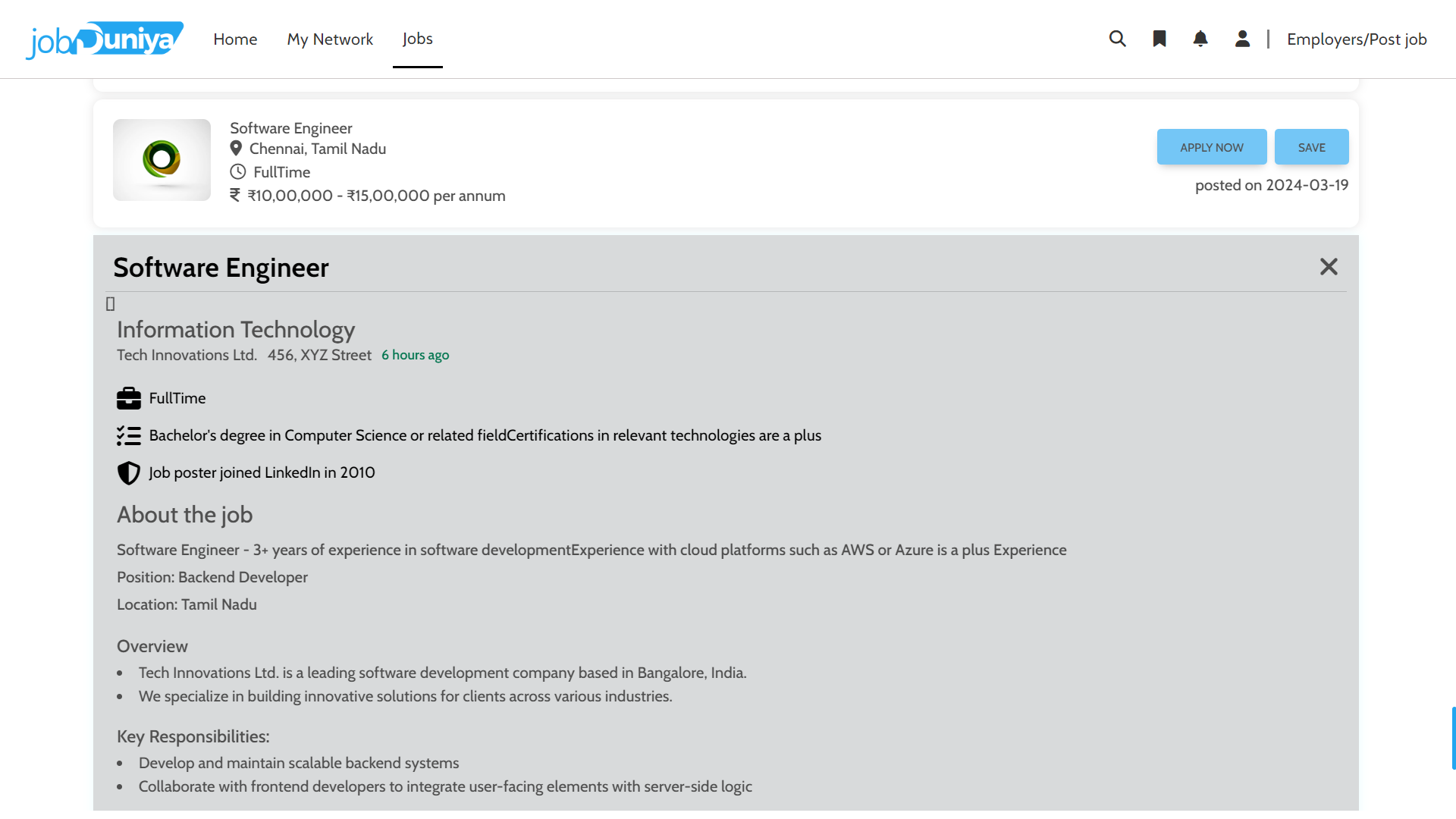


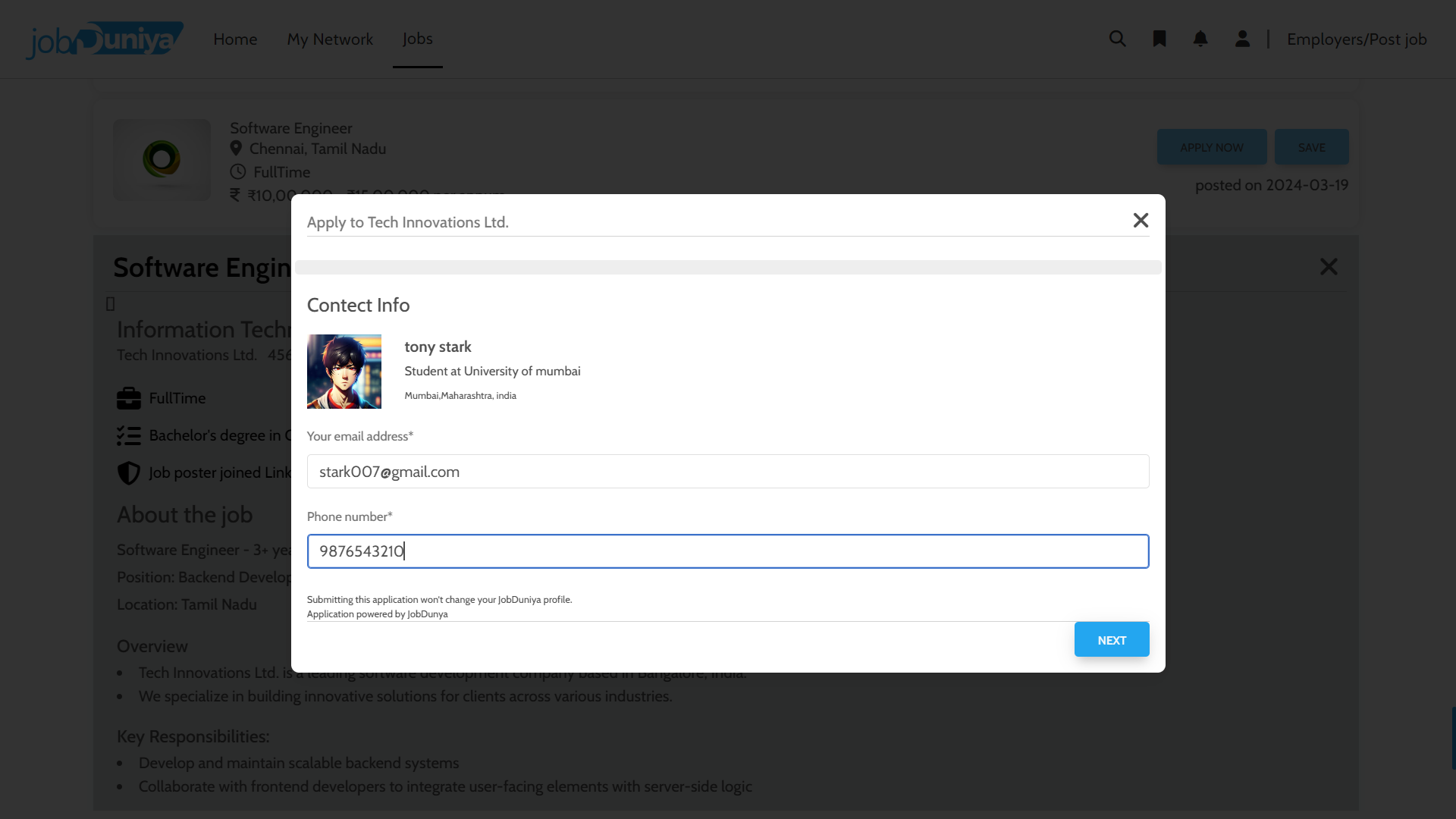


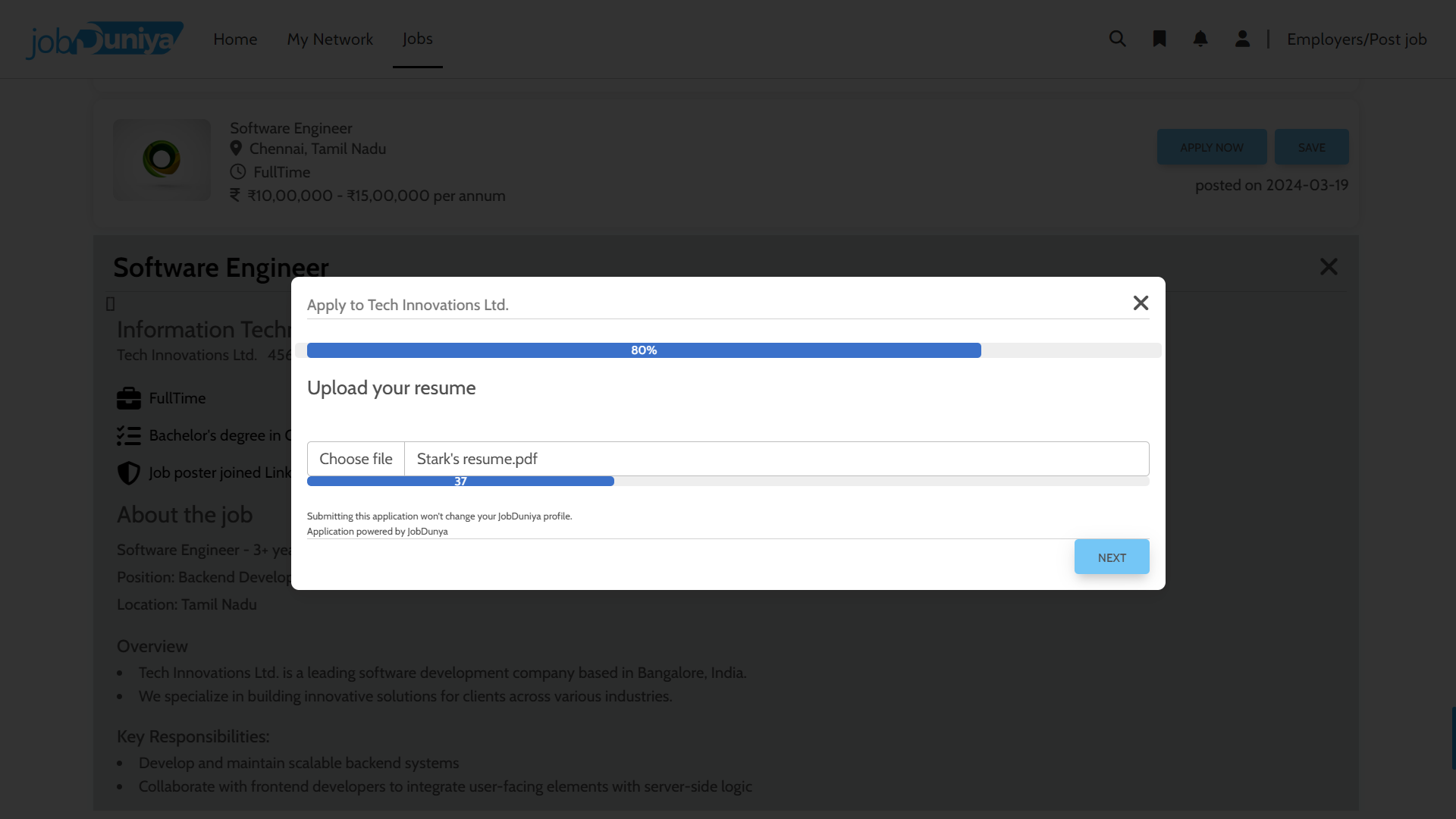


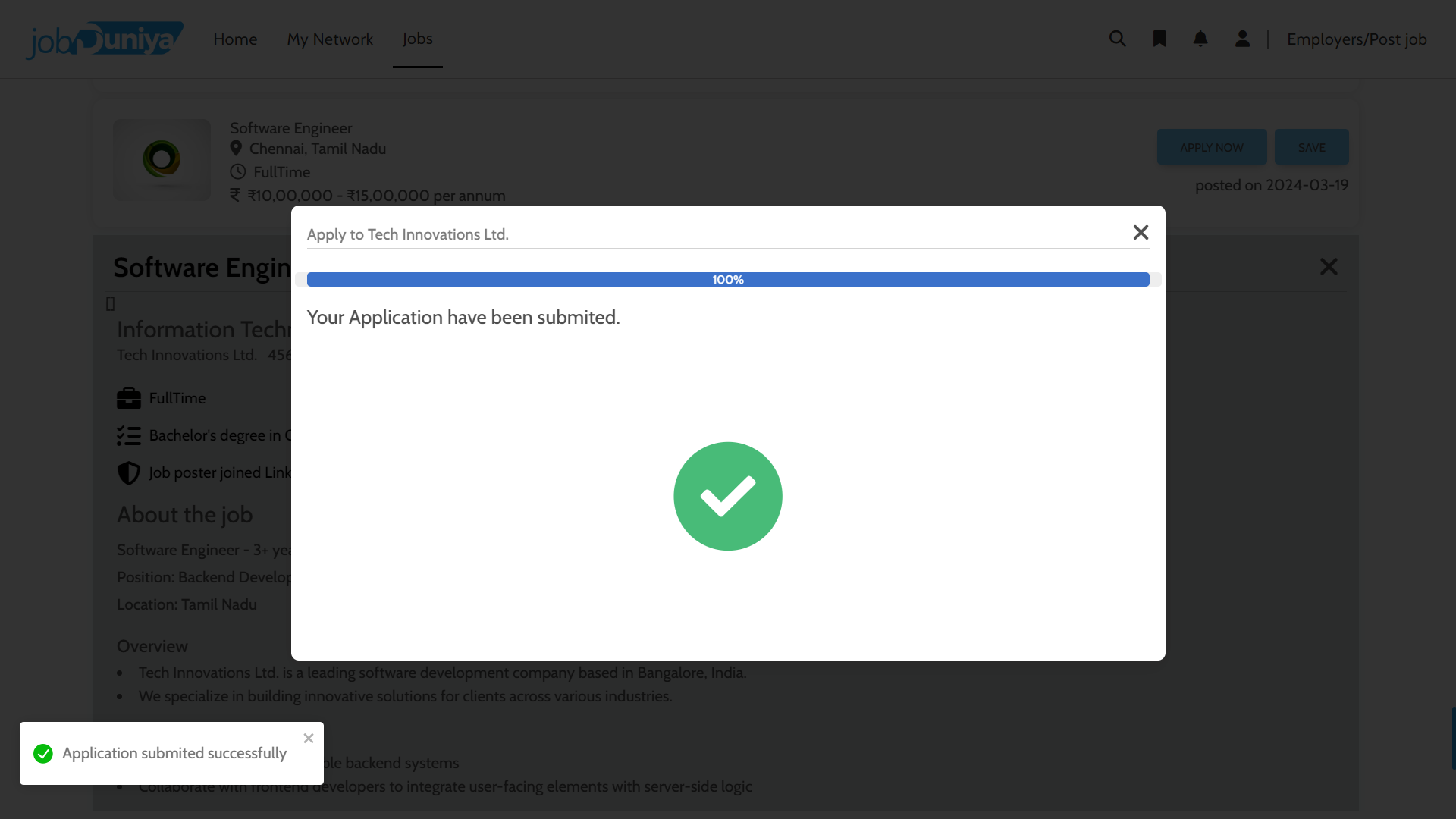


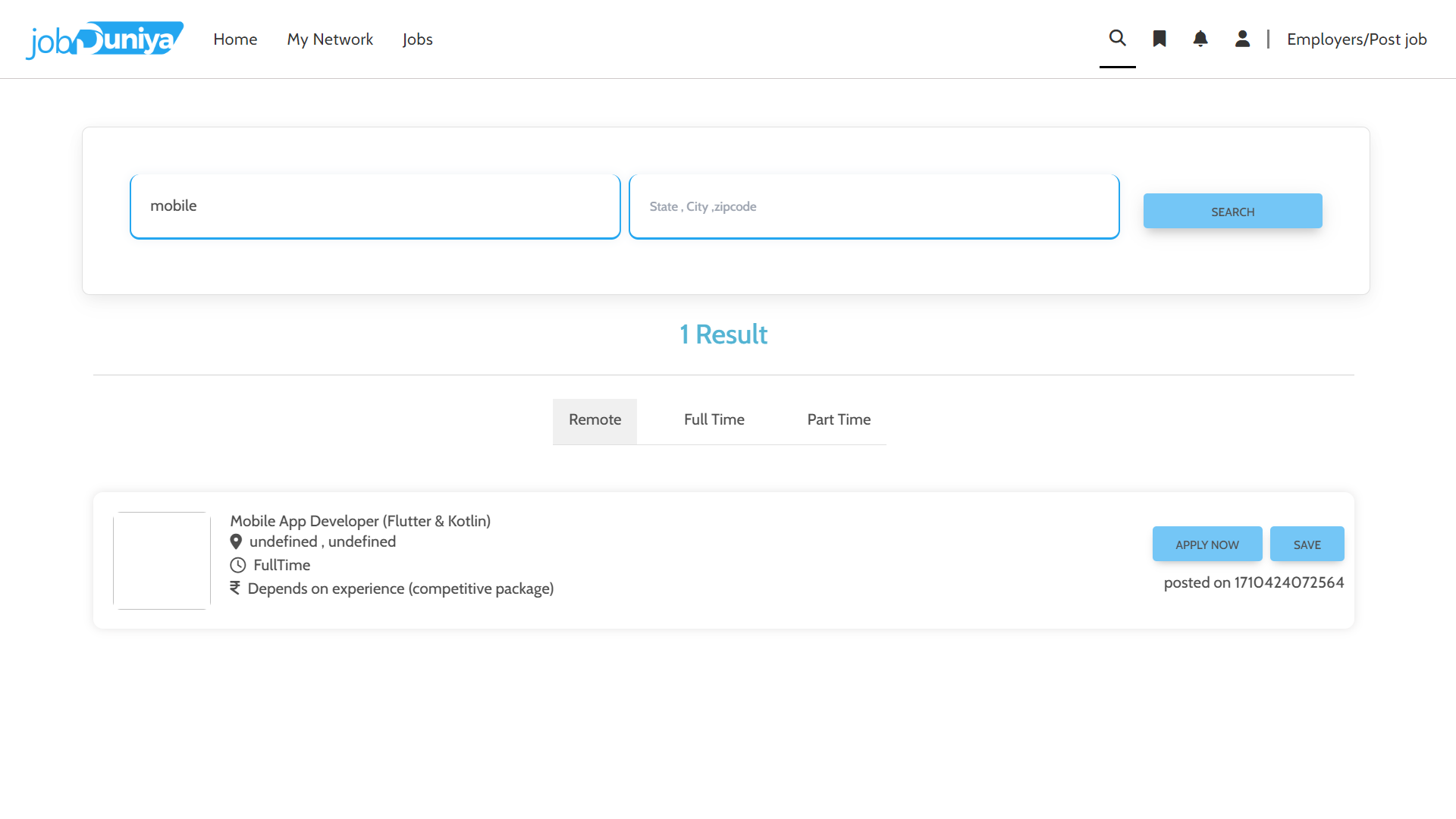


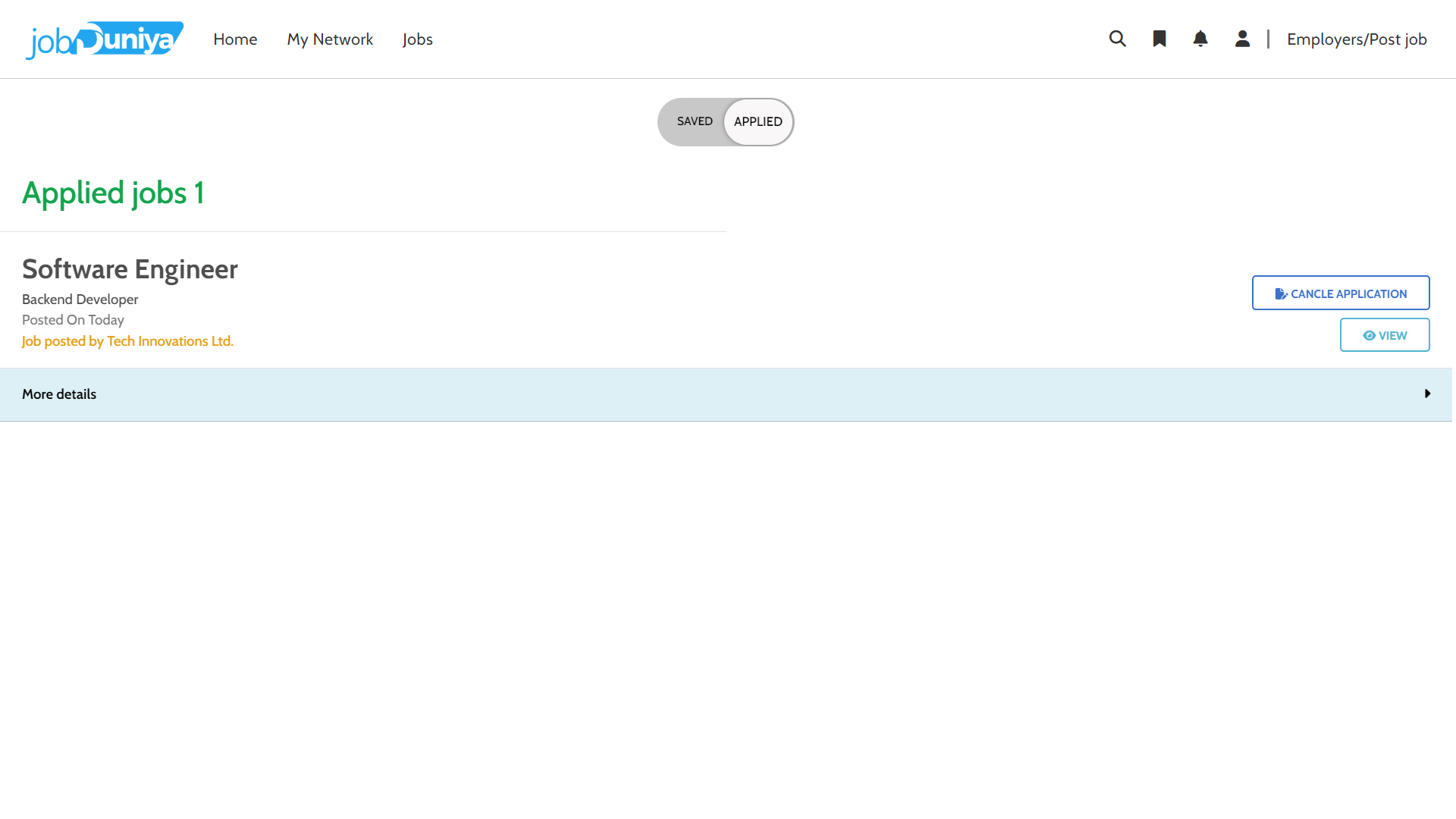




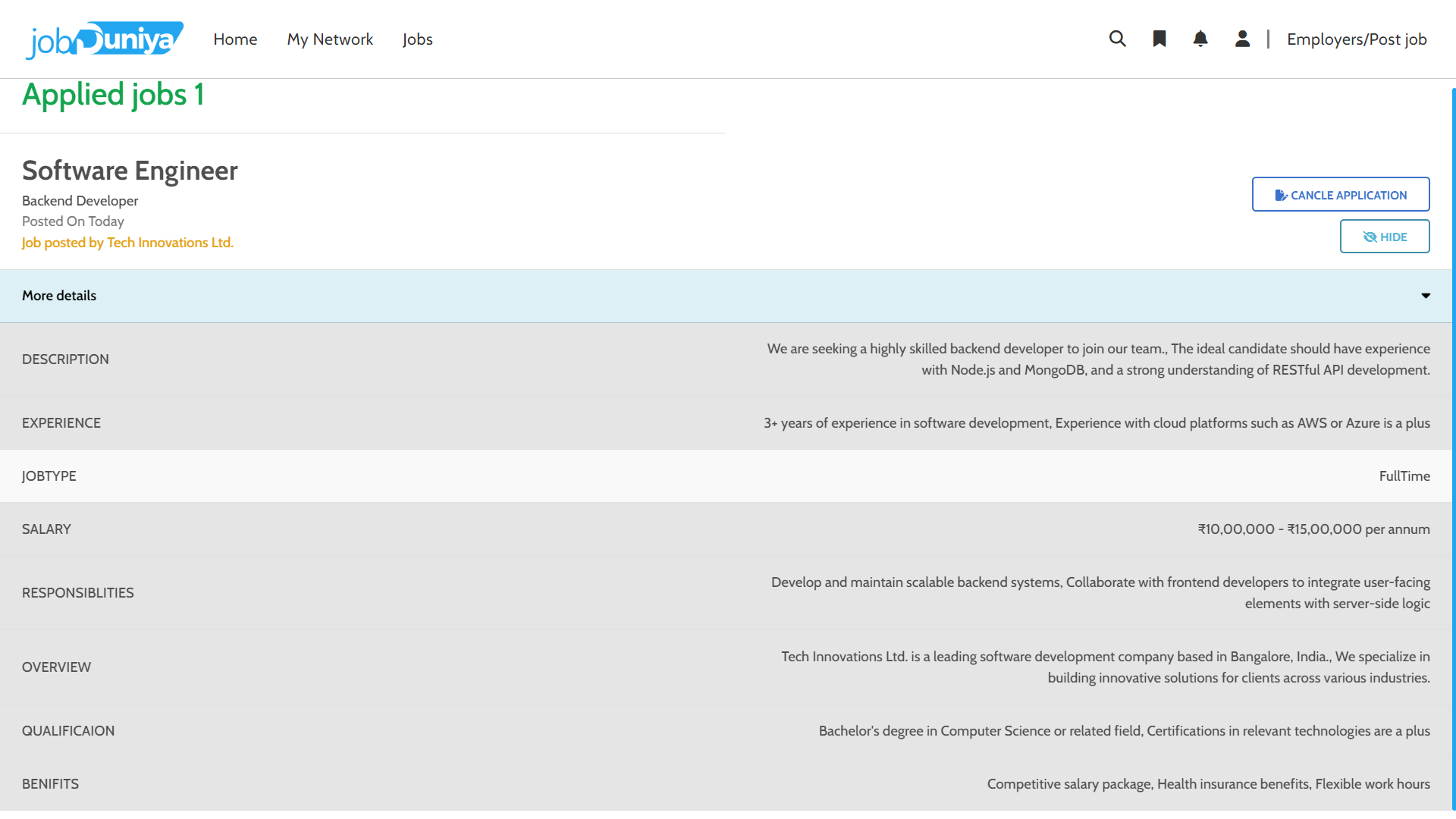


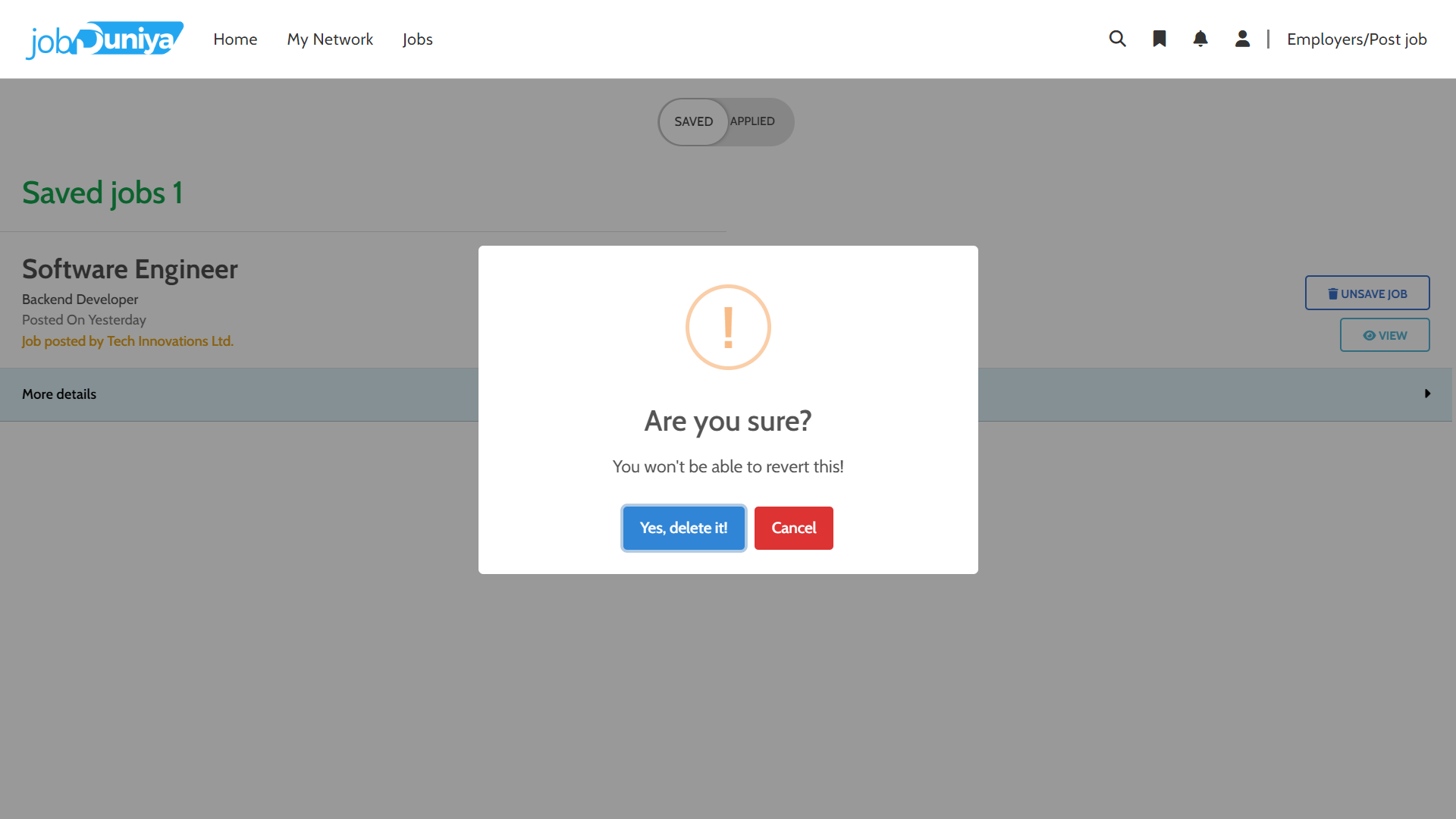


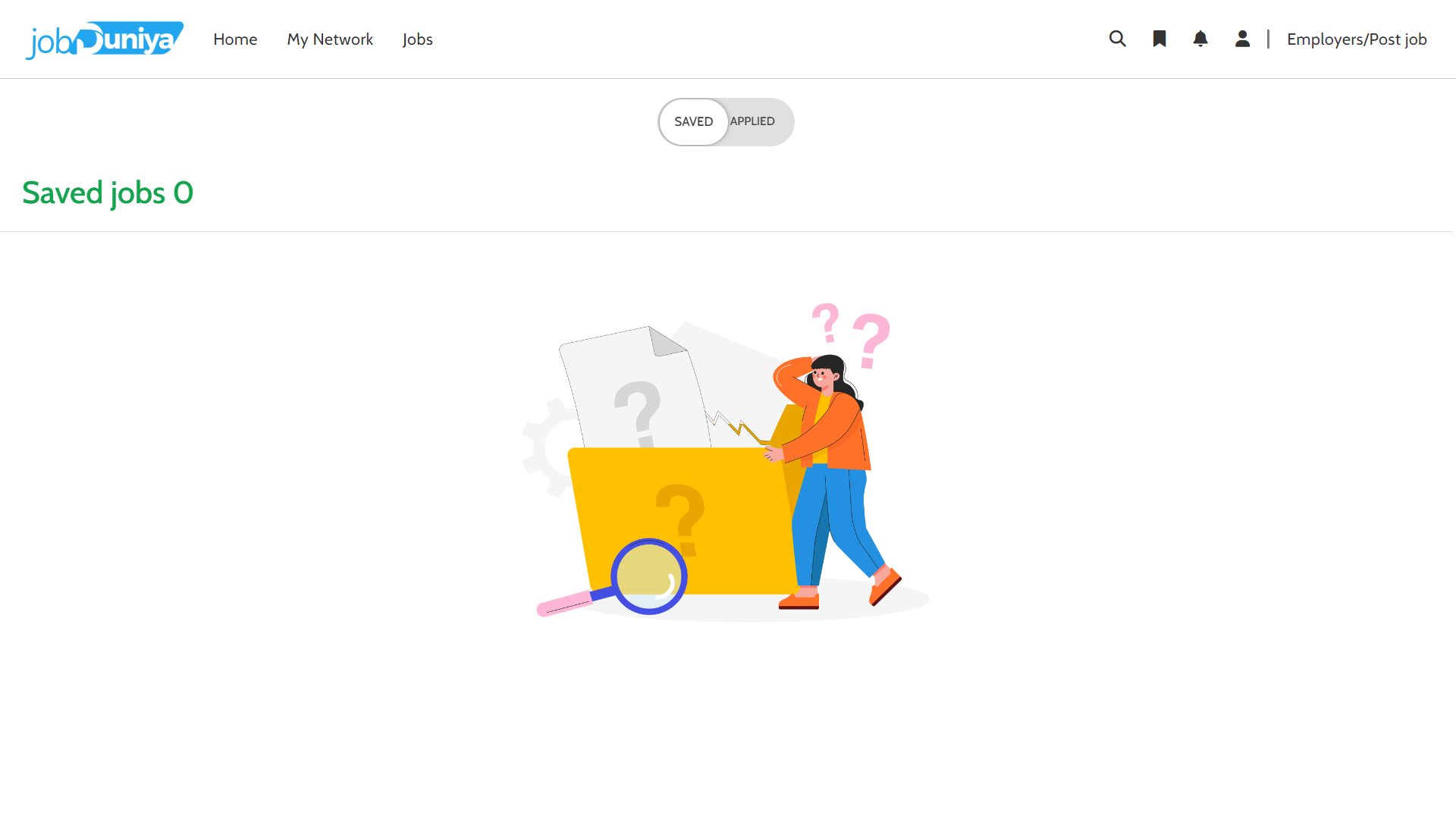
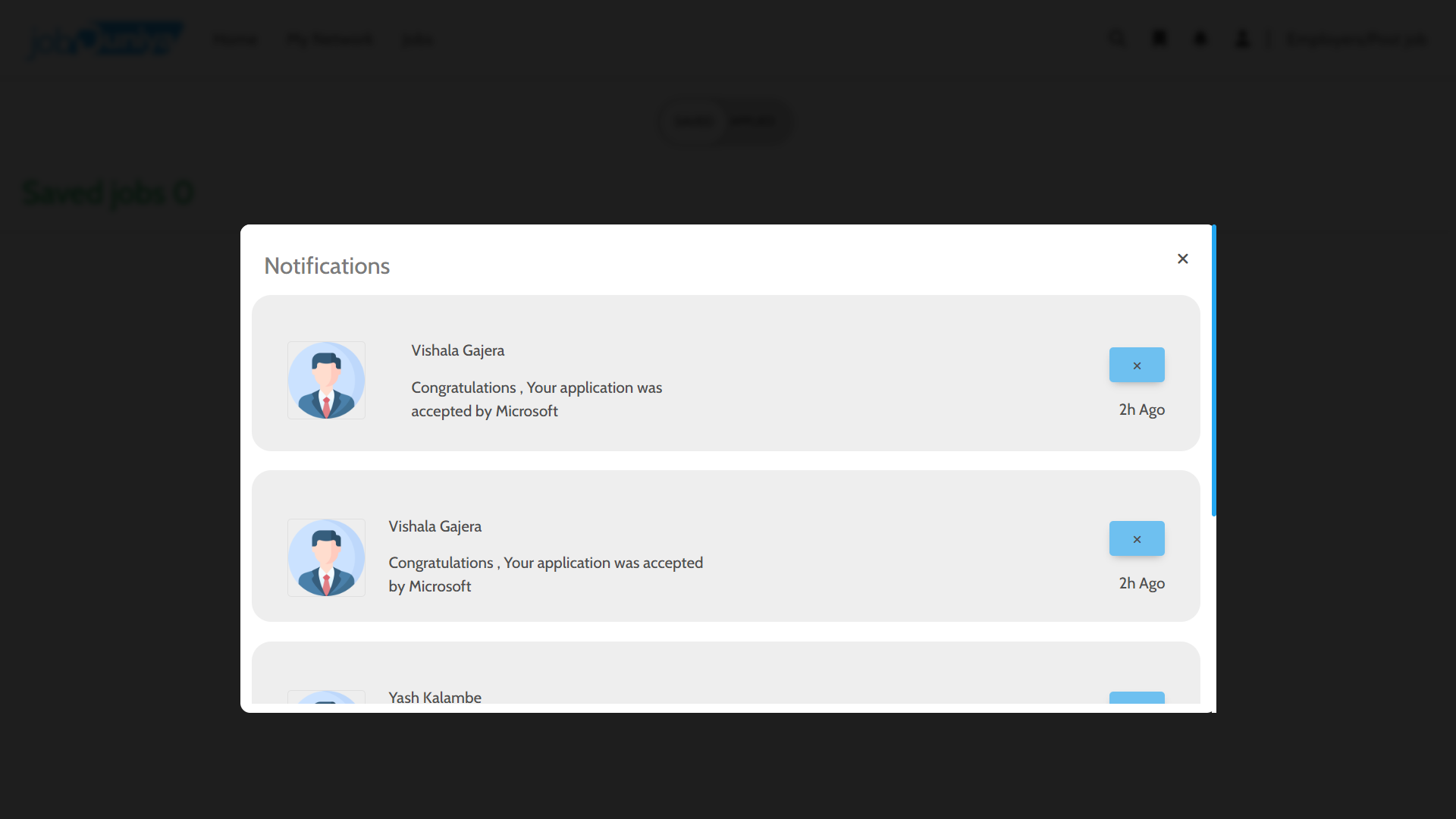


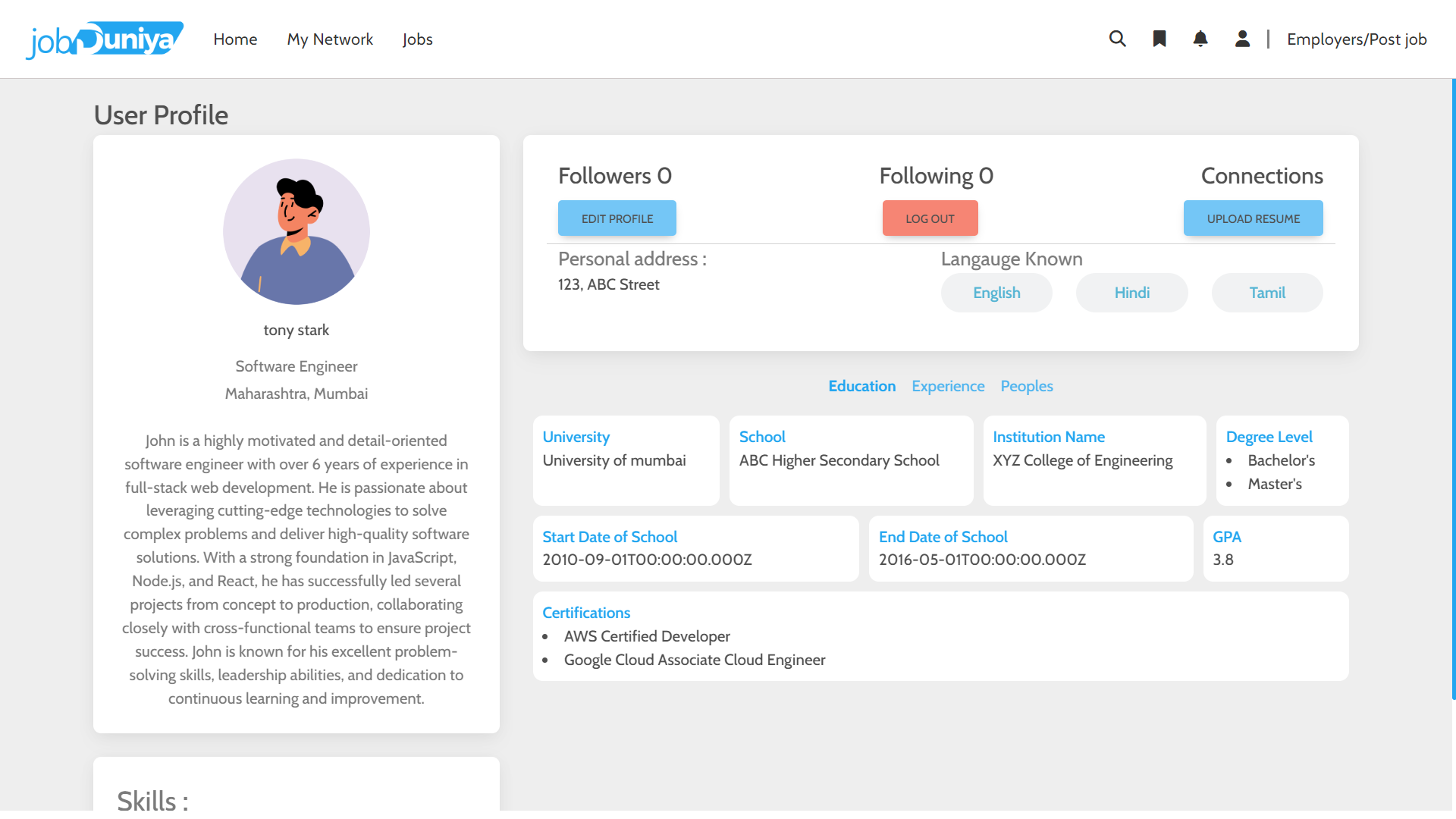


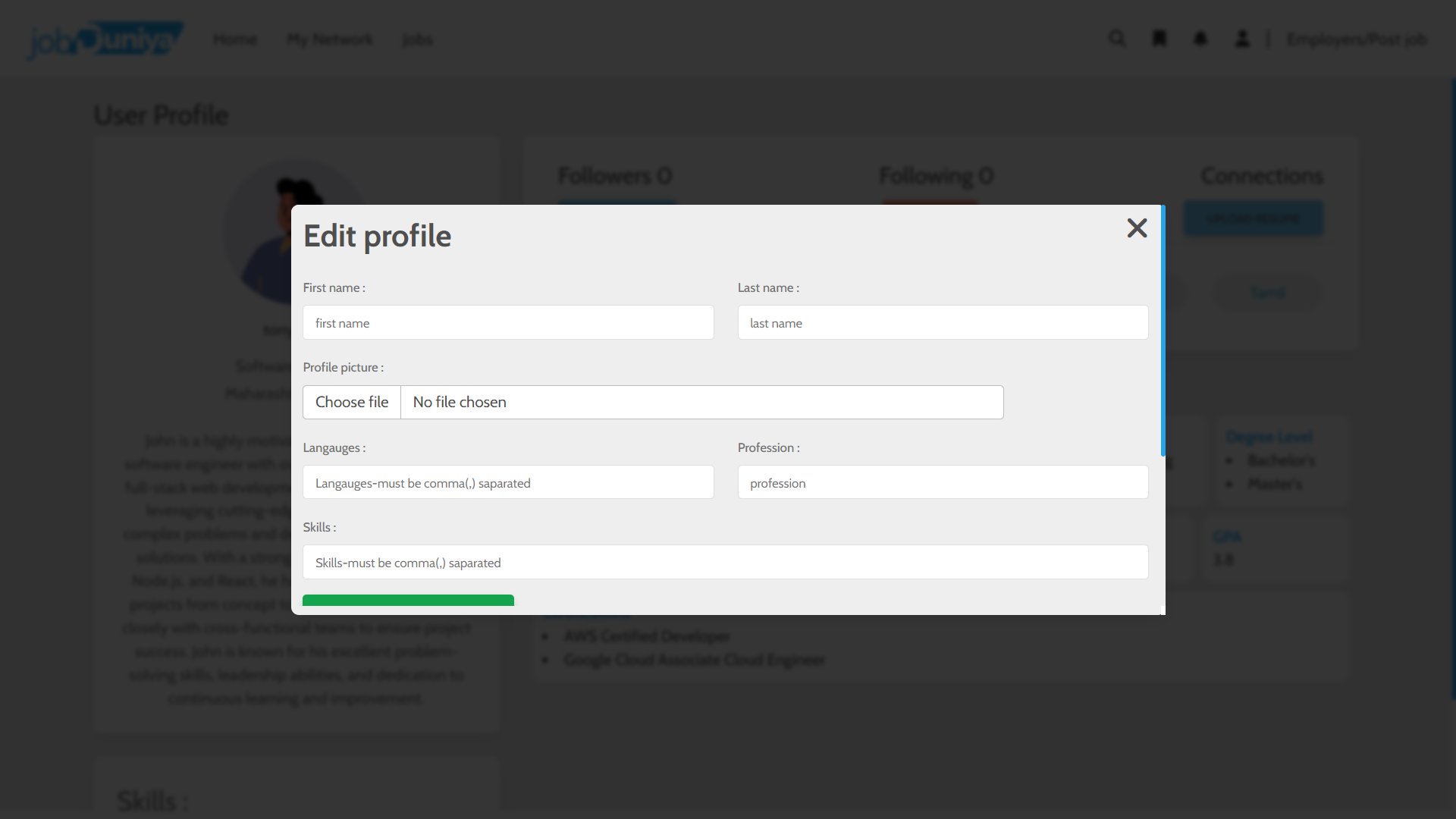


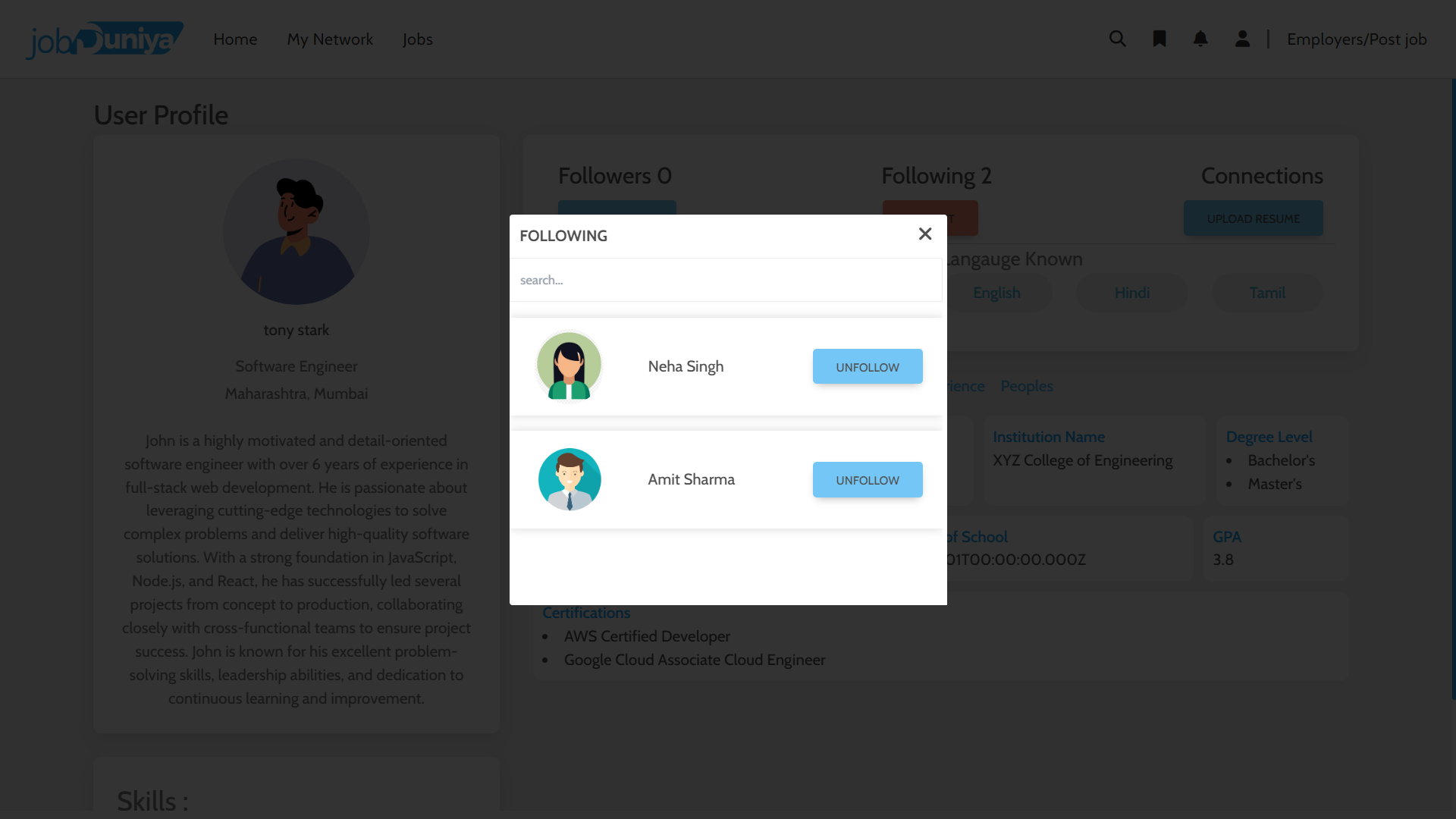




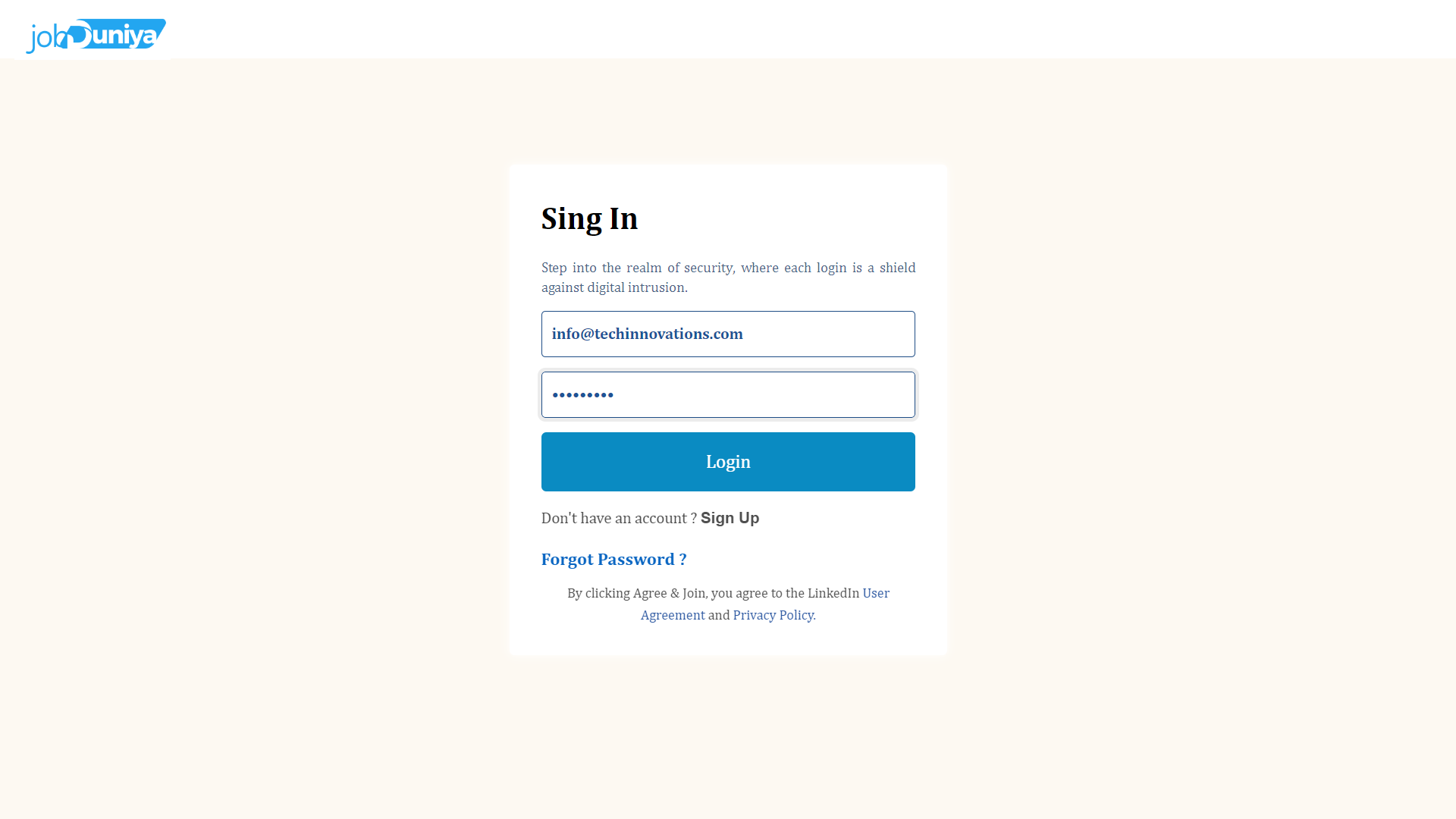
 

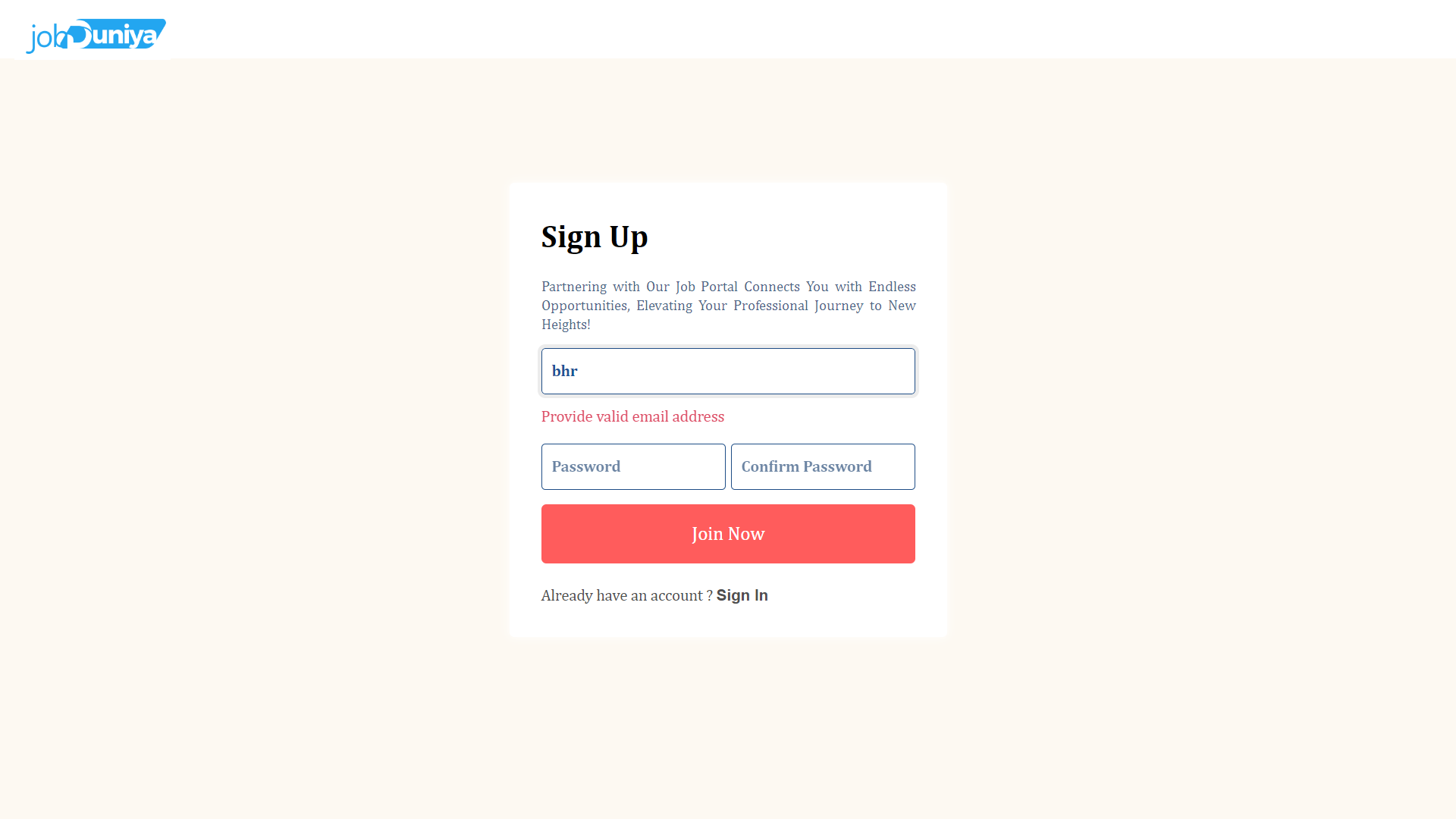


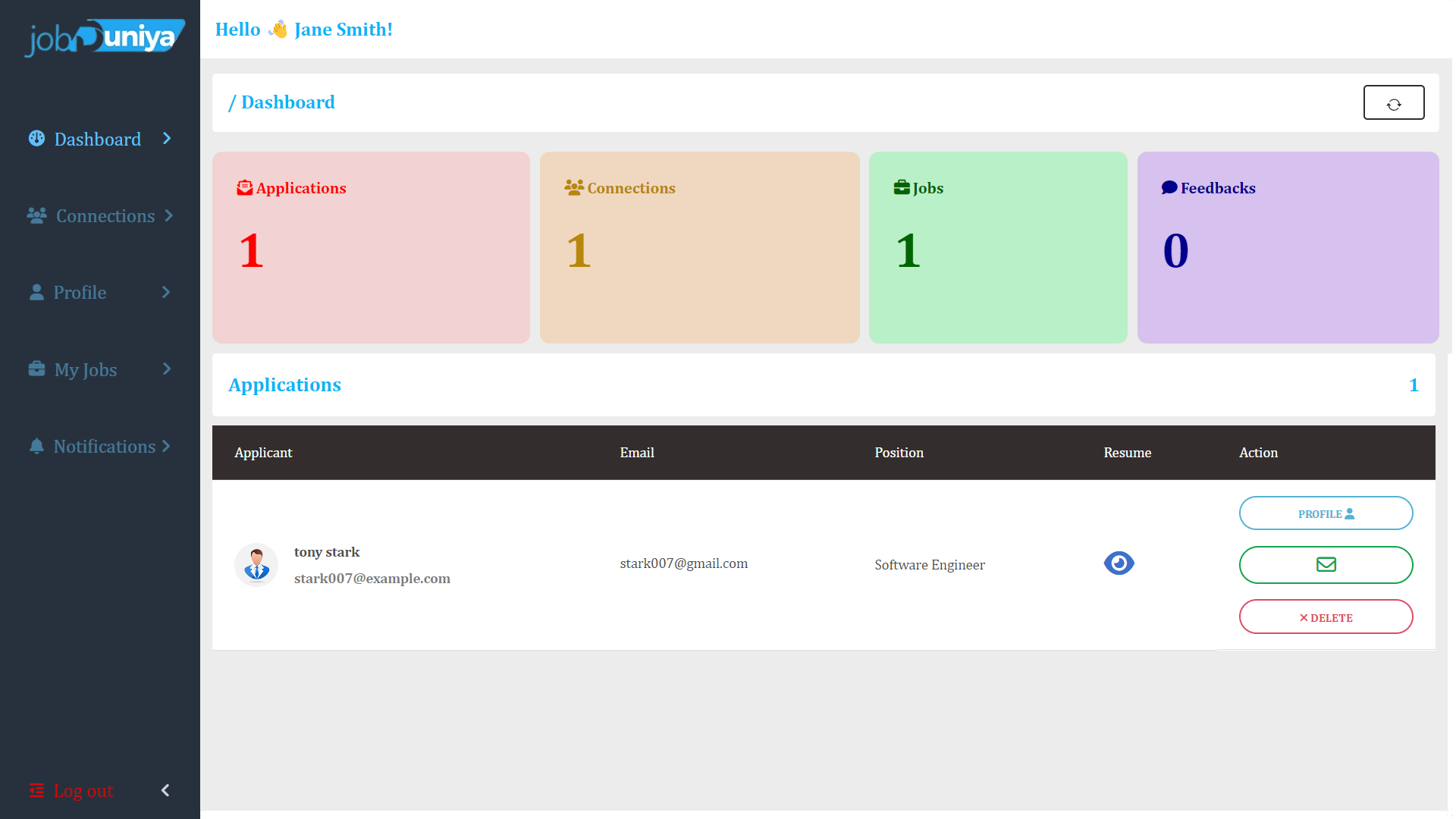


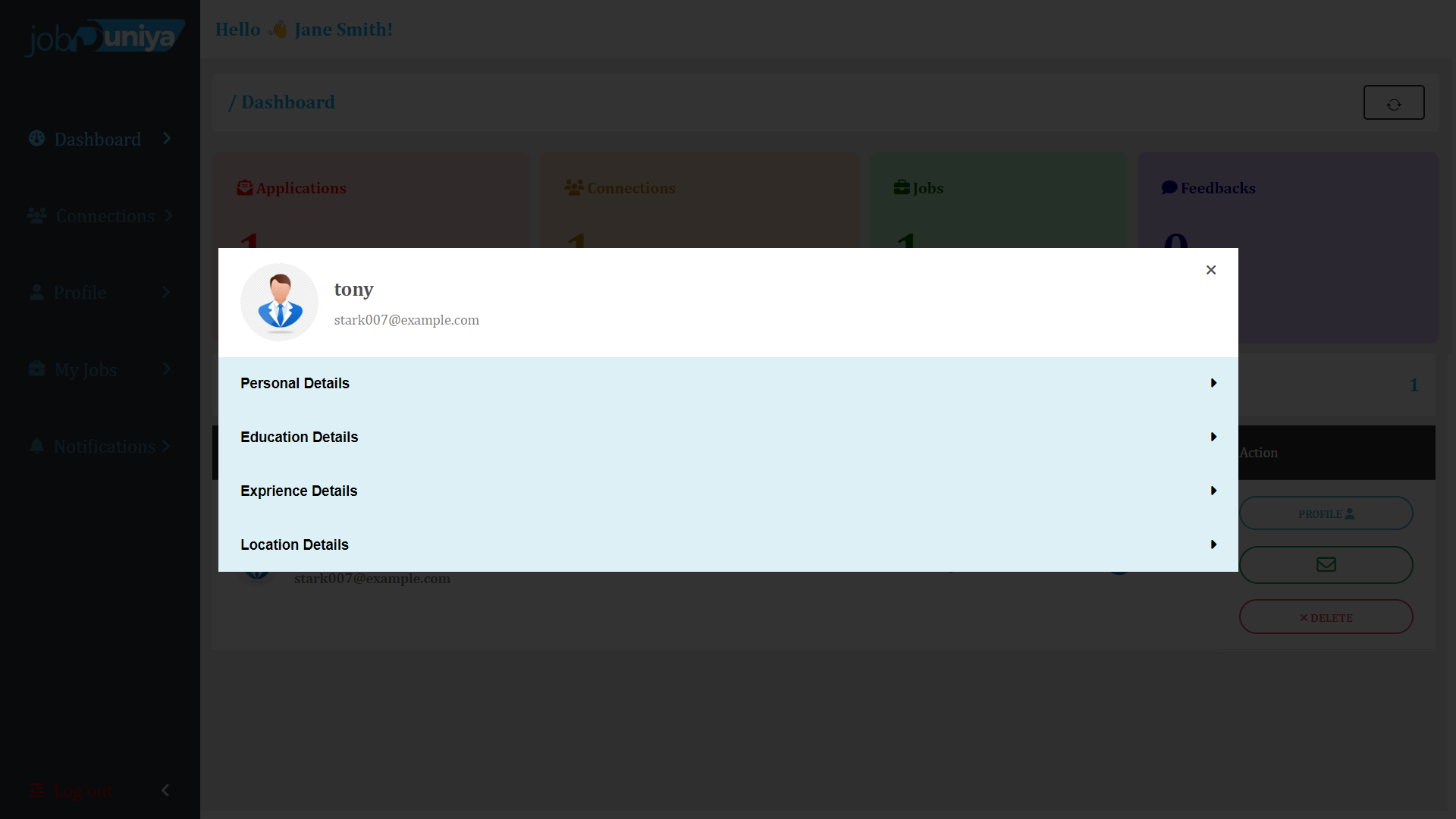


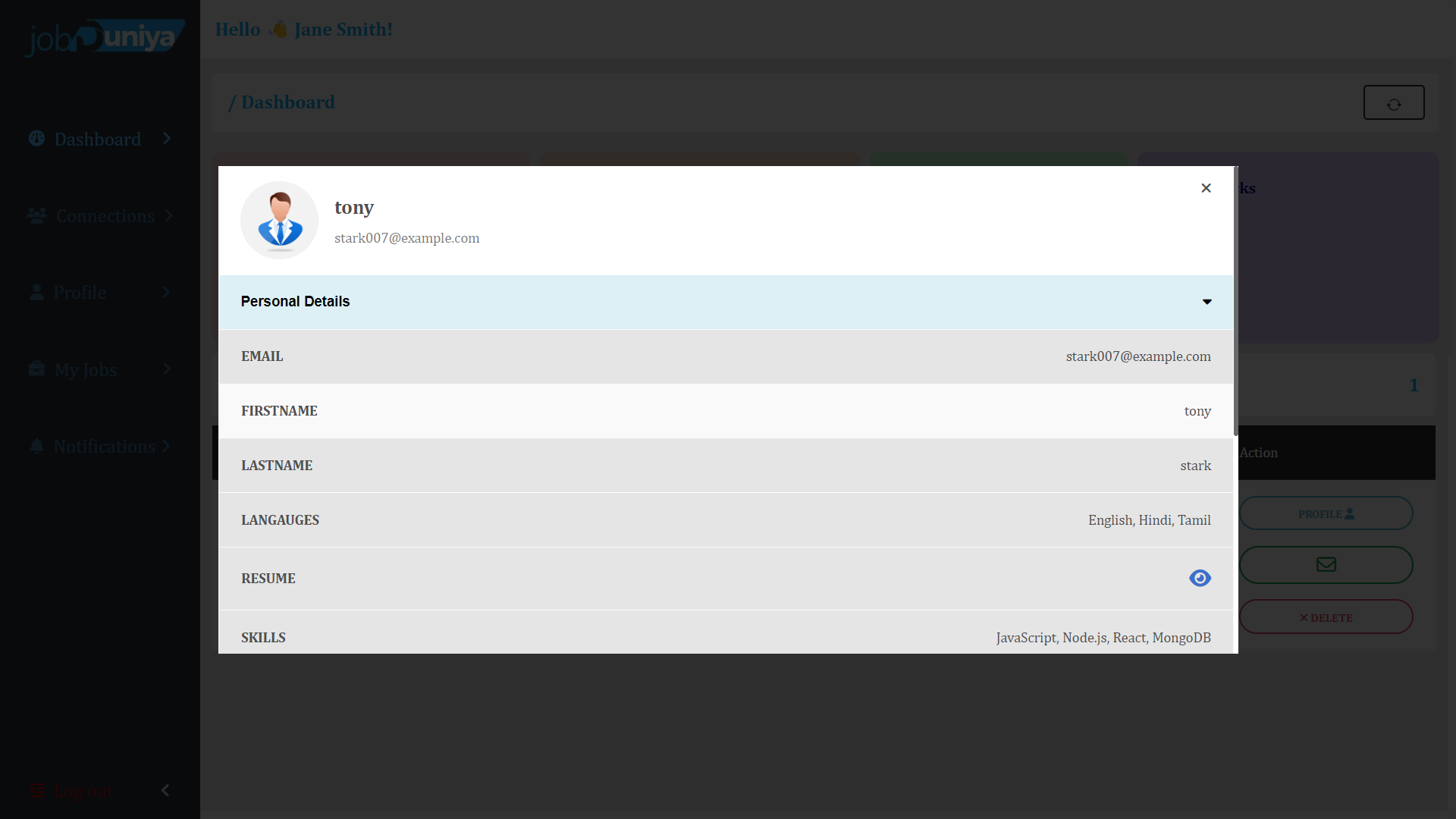
* Company/Admin side :-

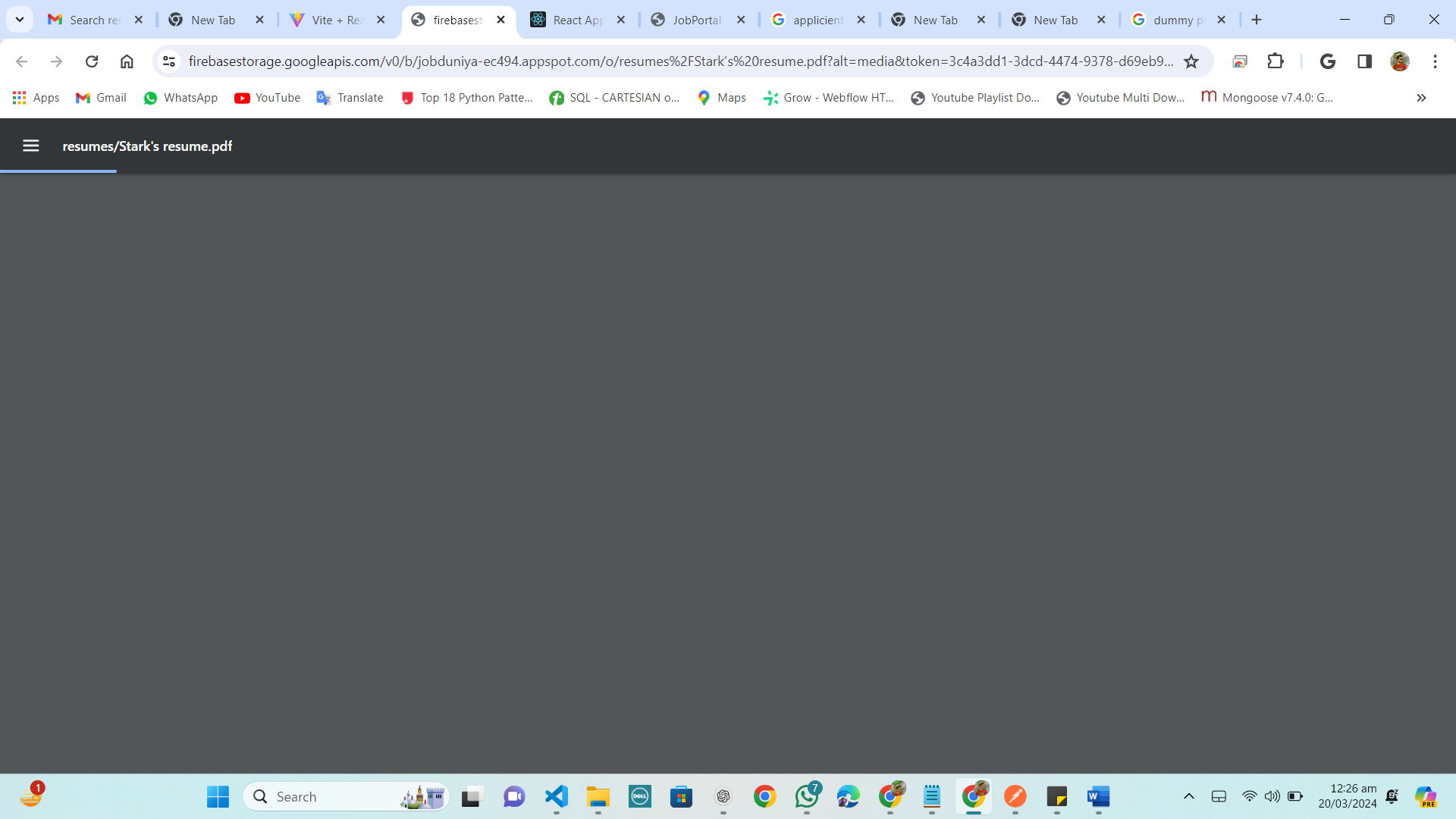


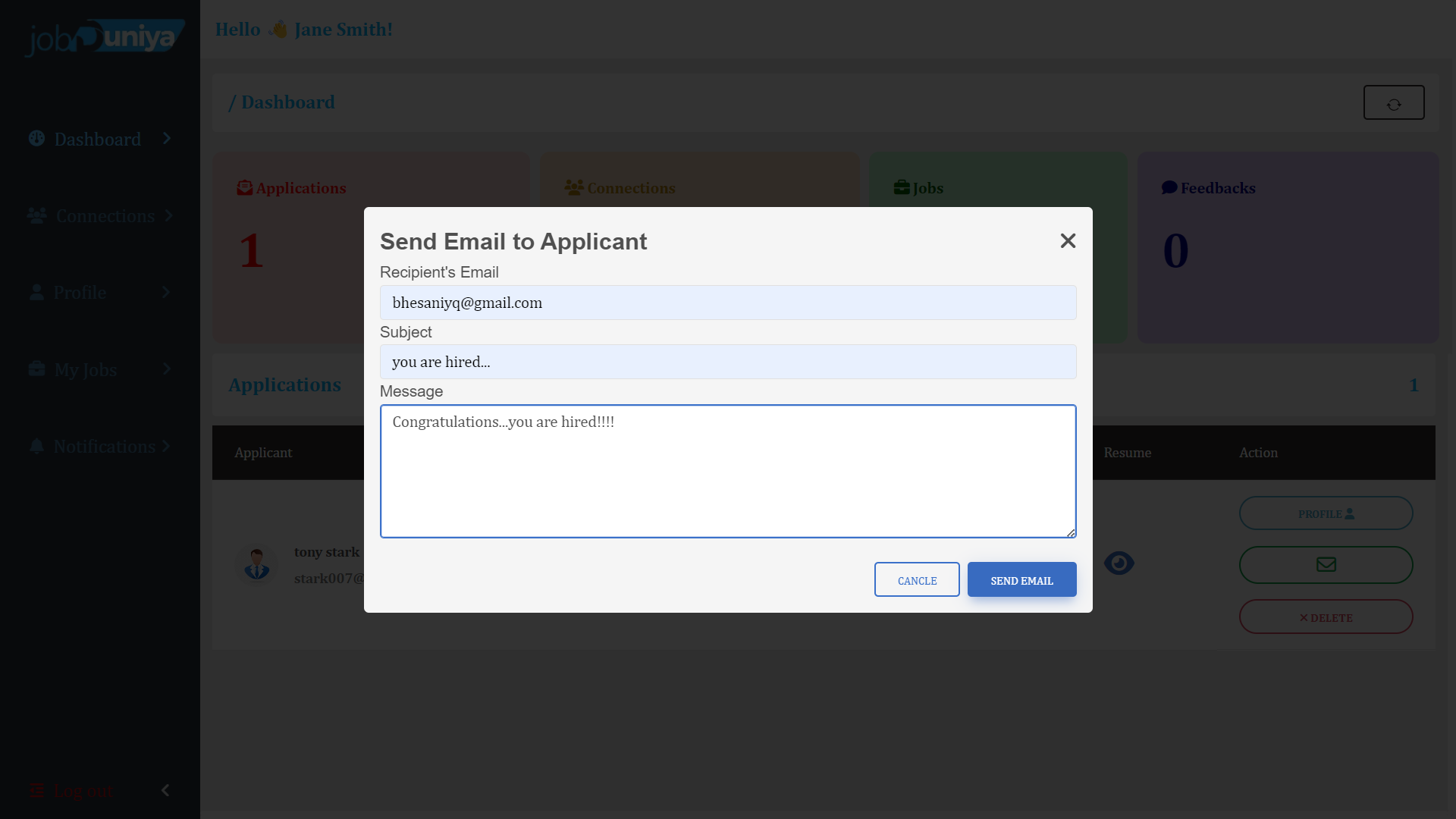


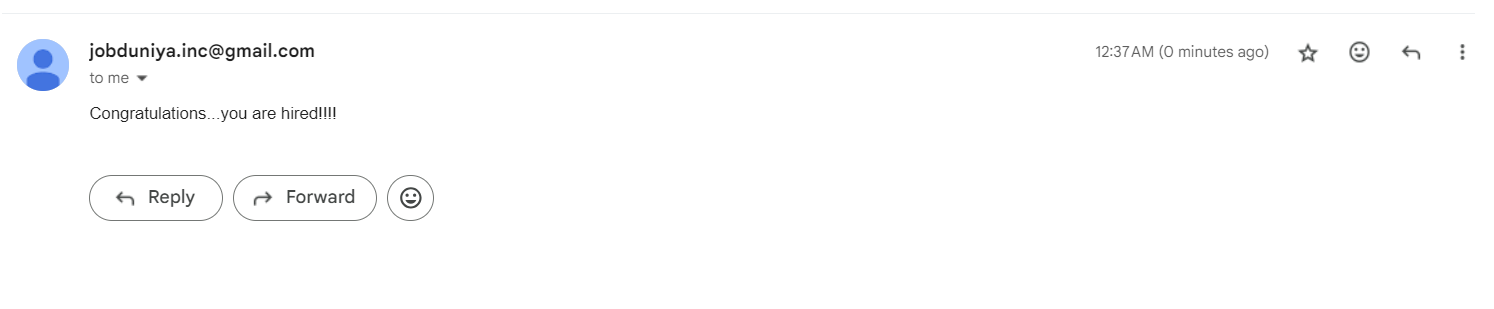




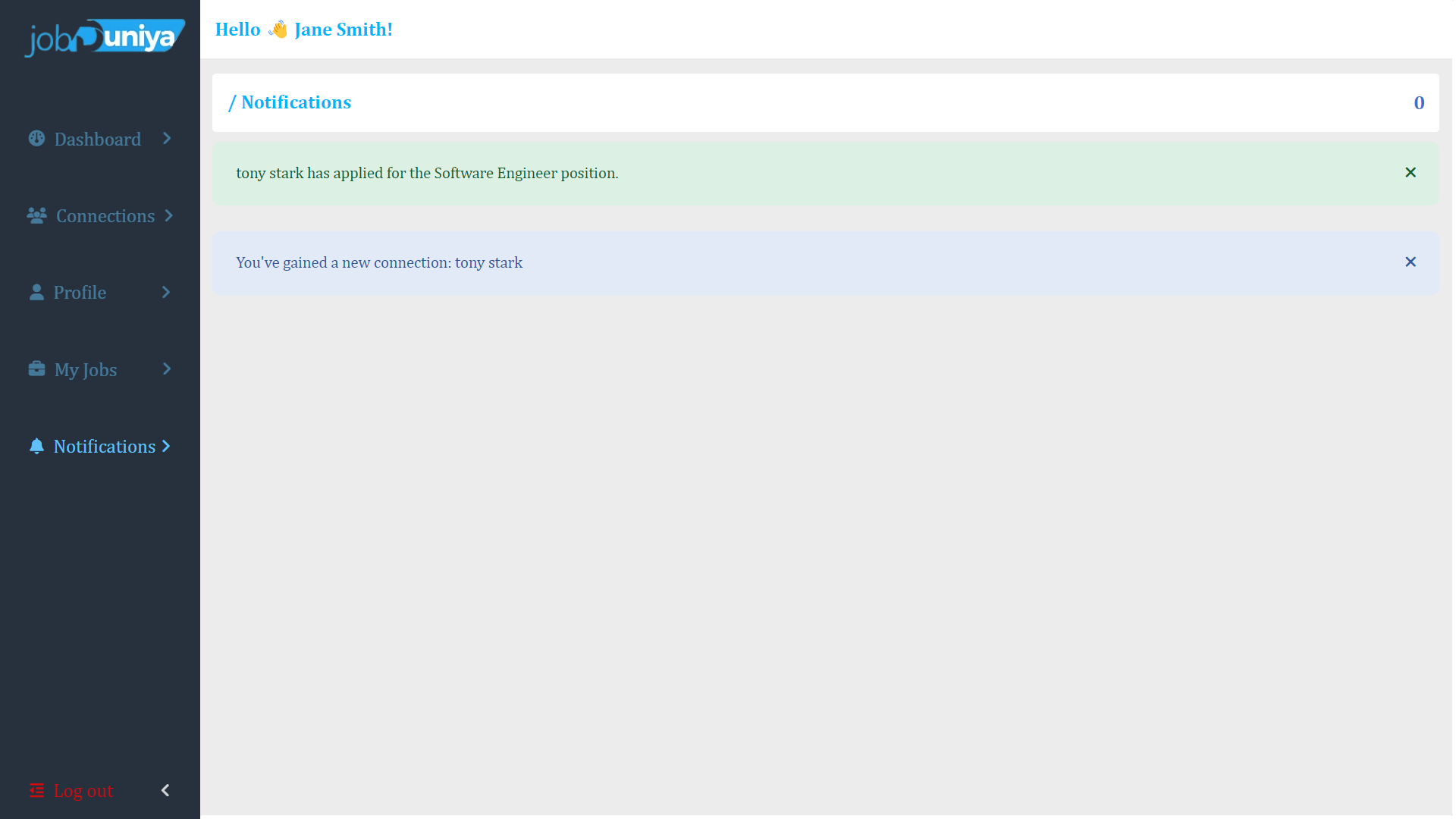


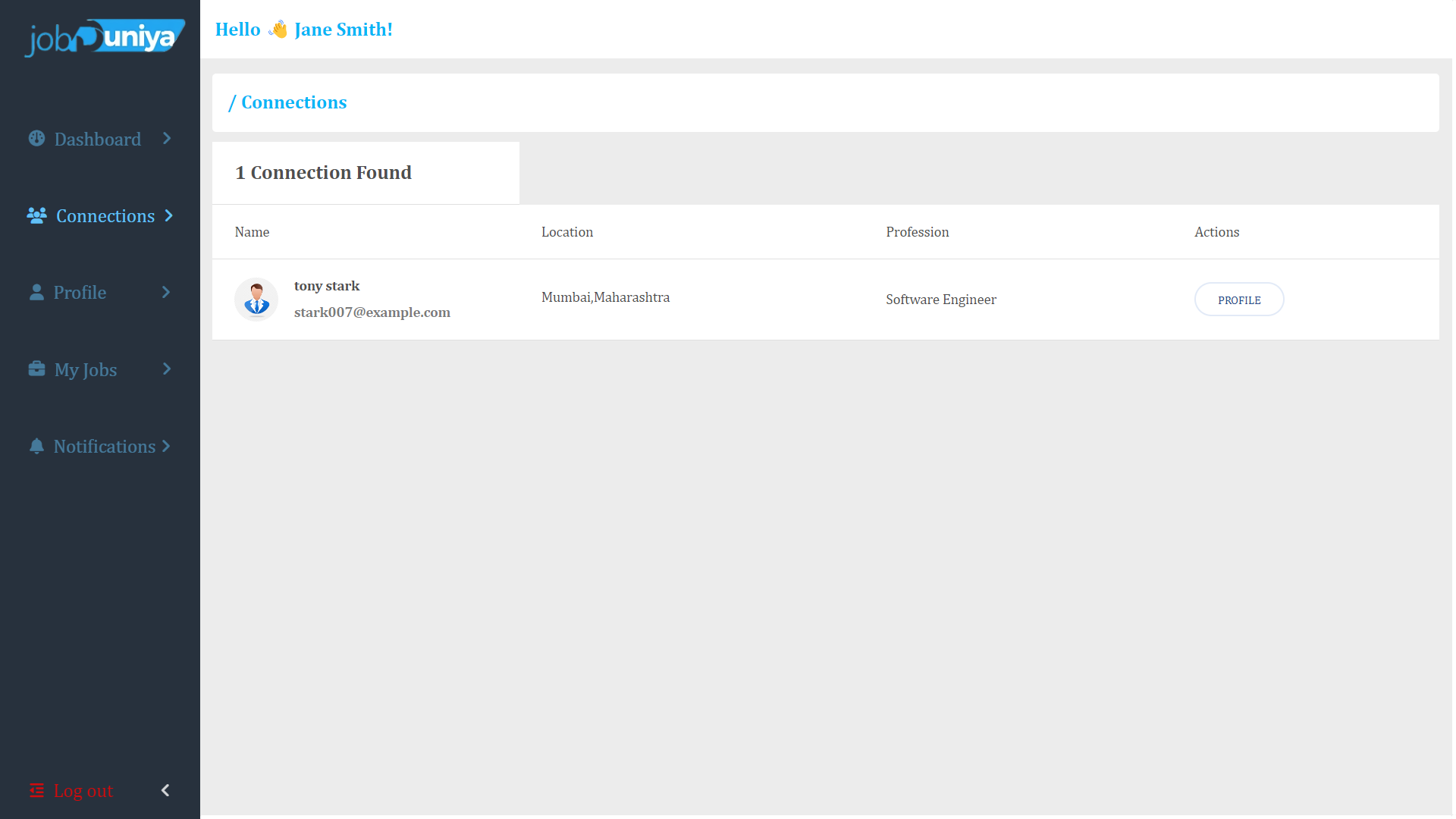


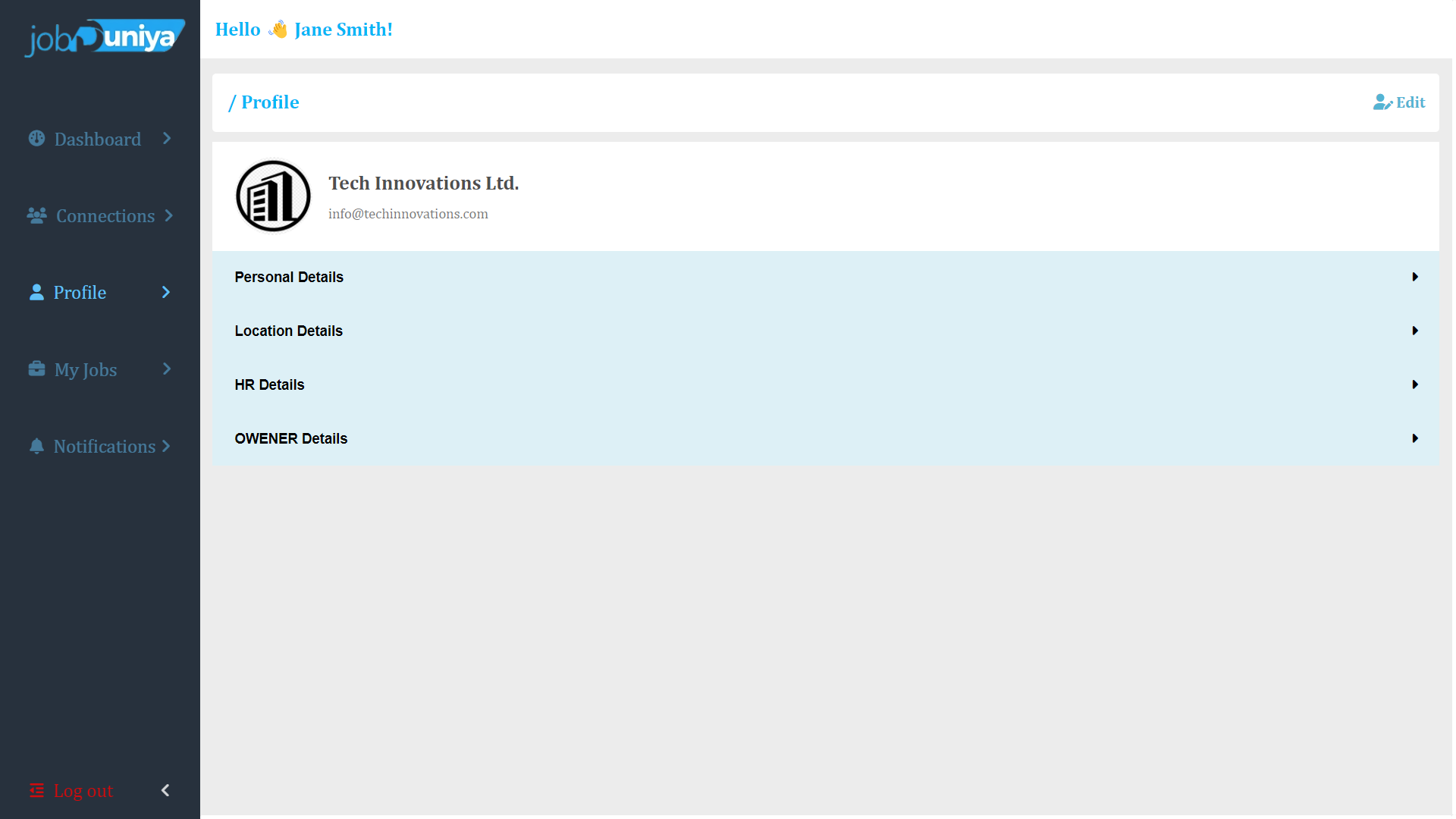


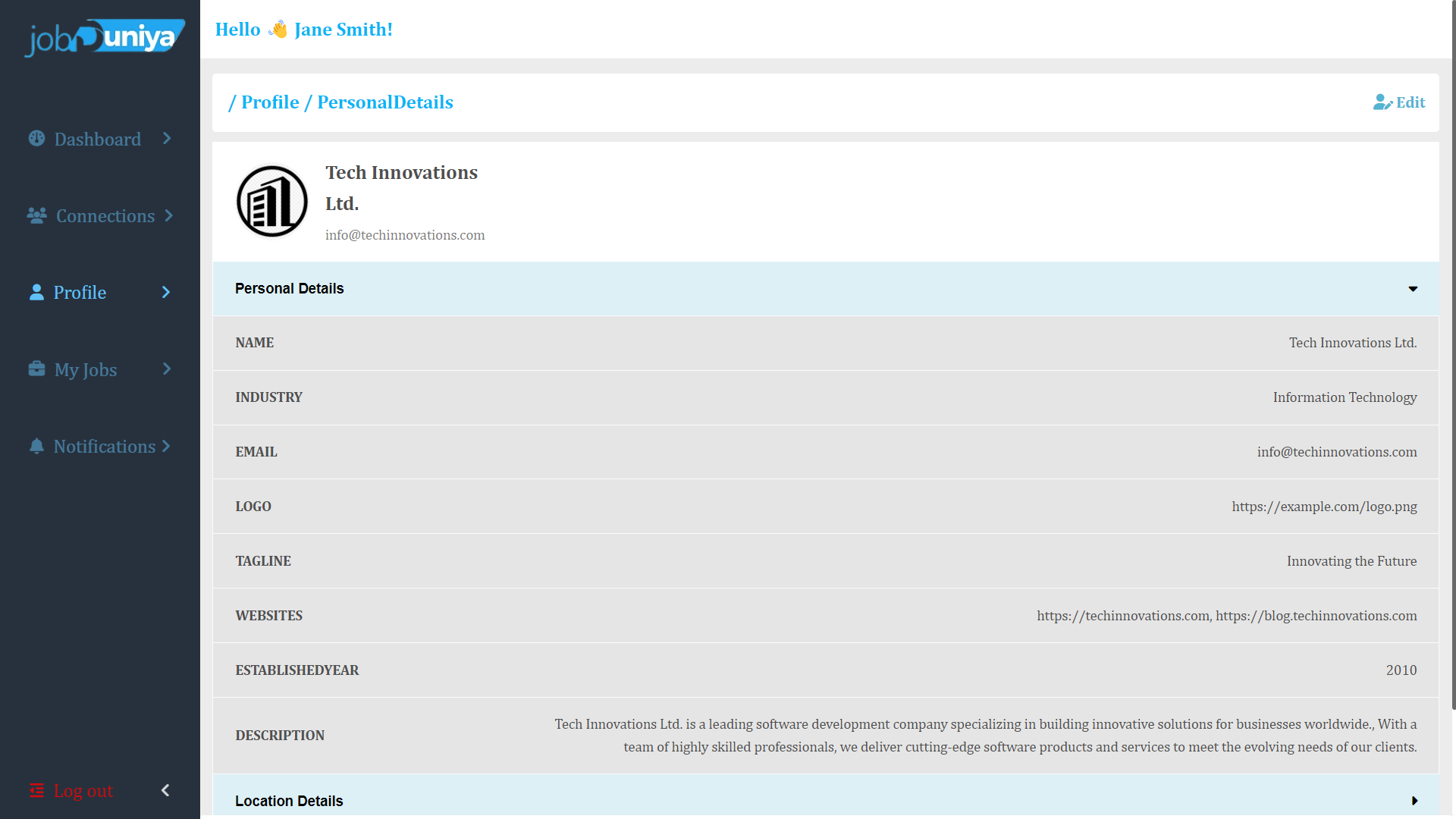
****

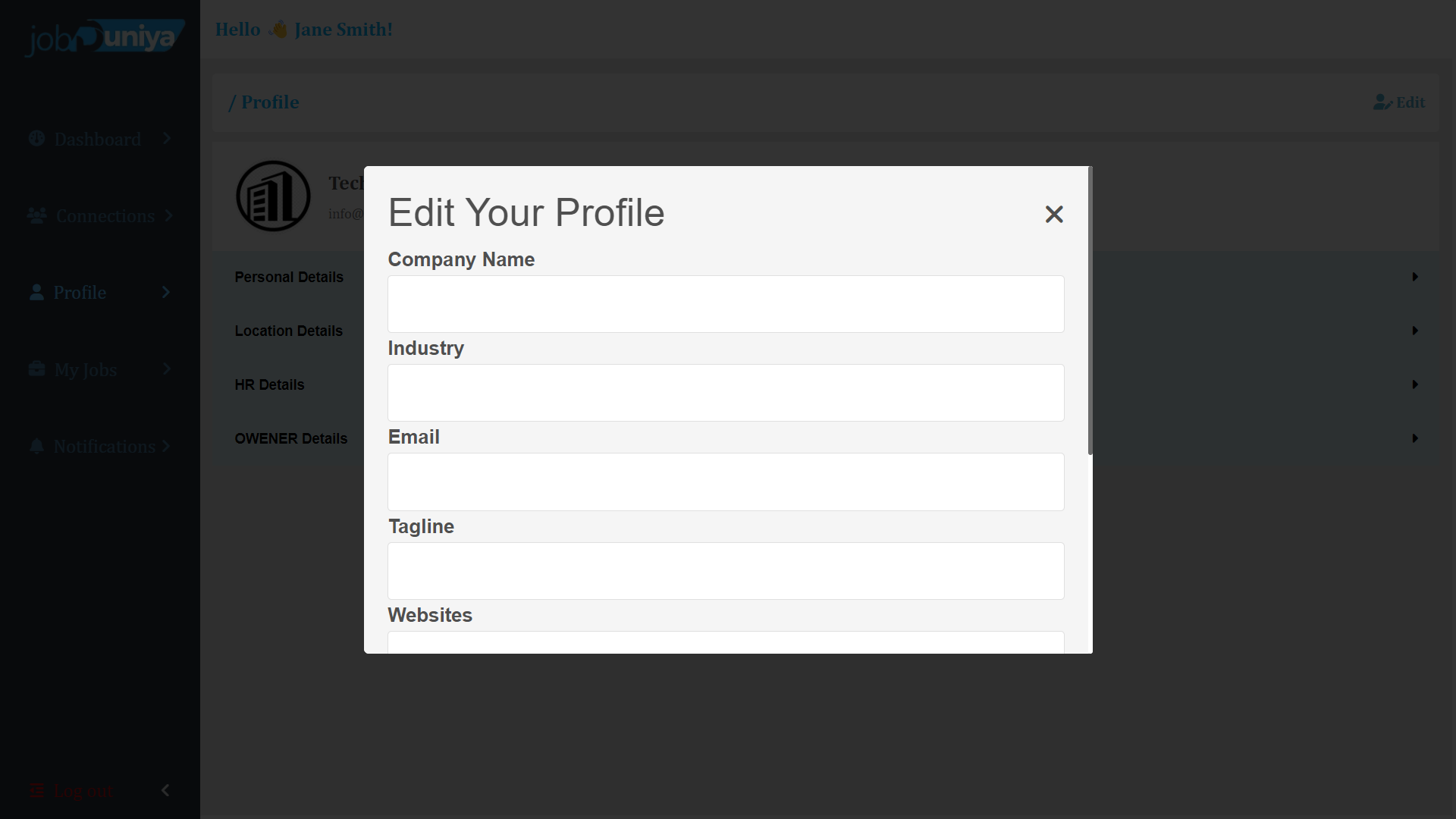


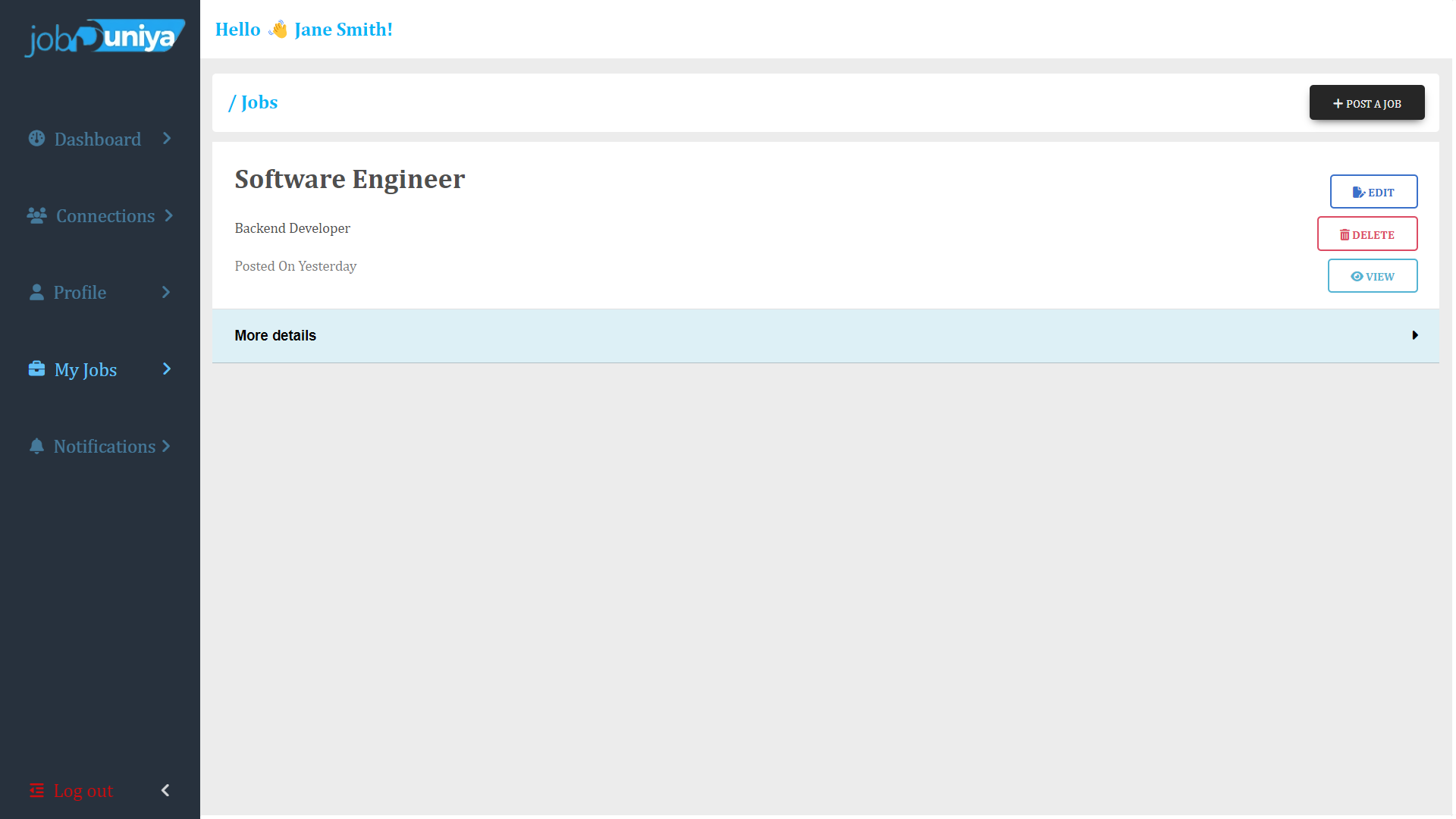


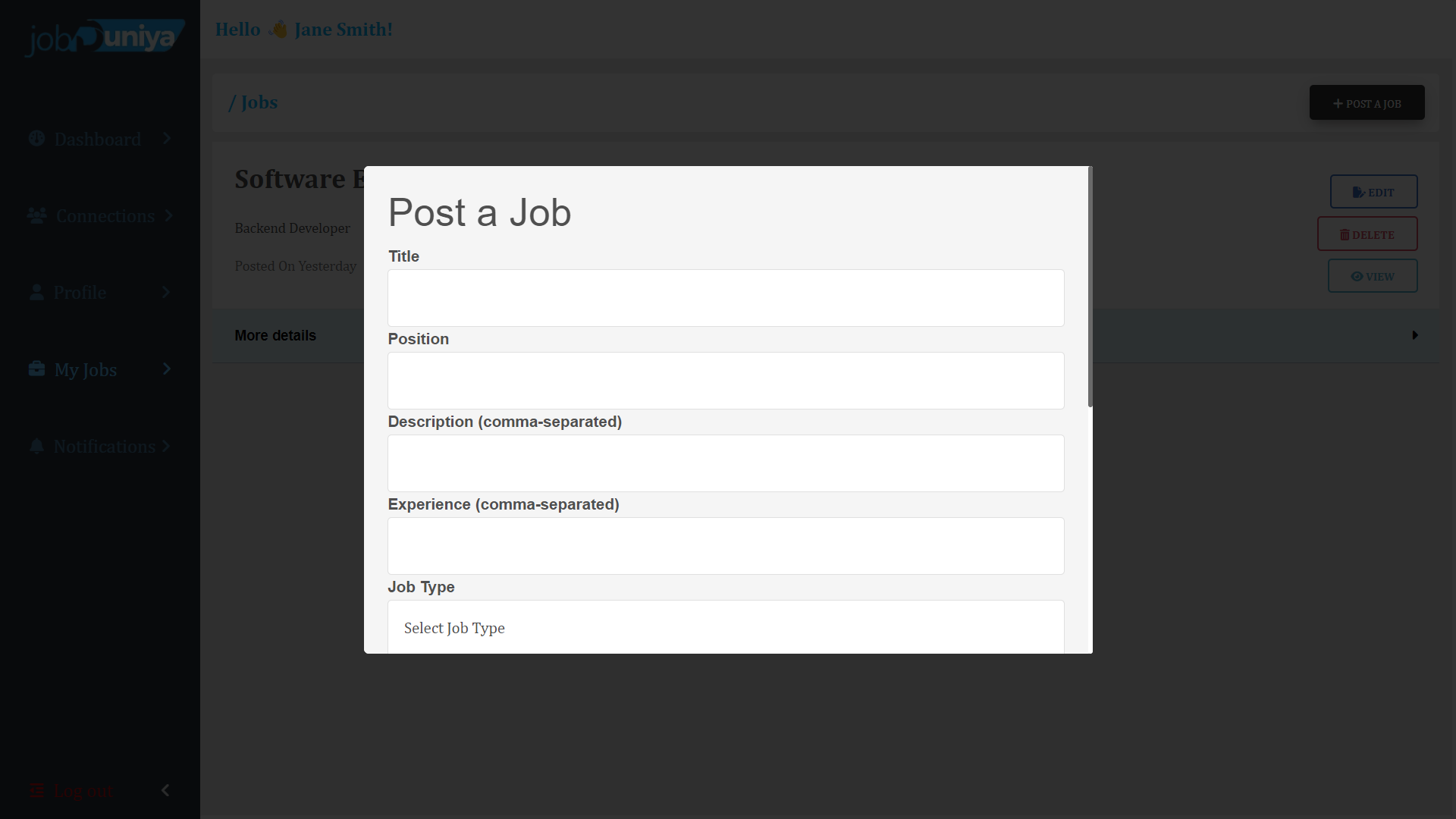
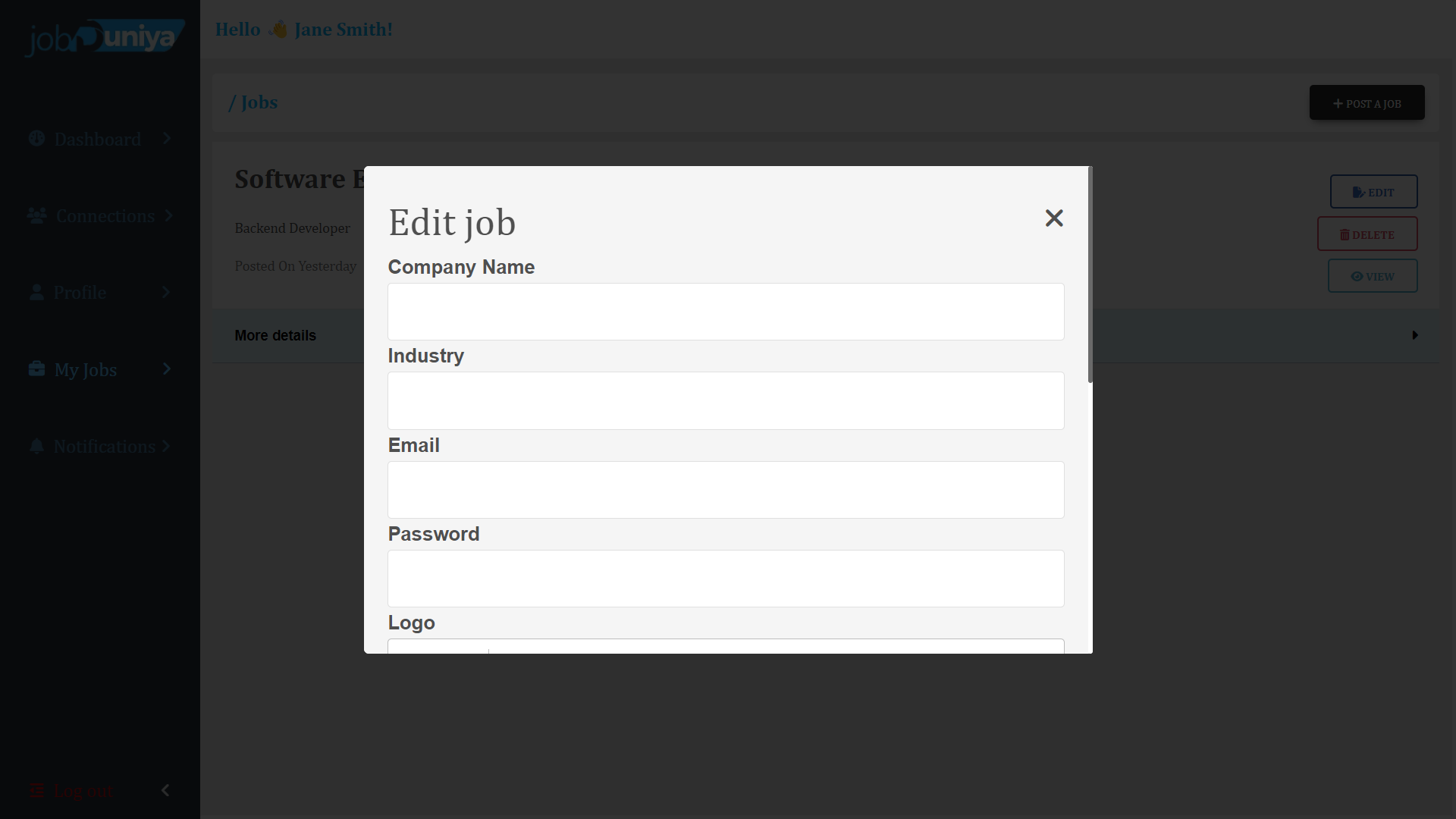


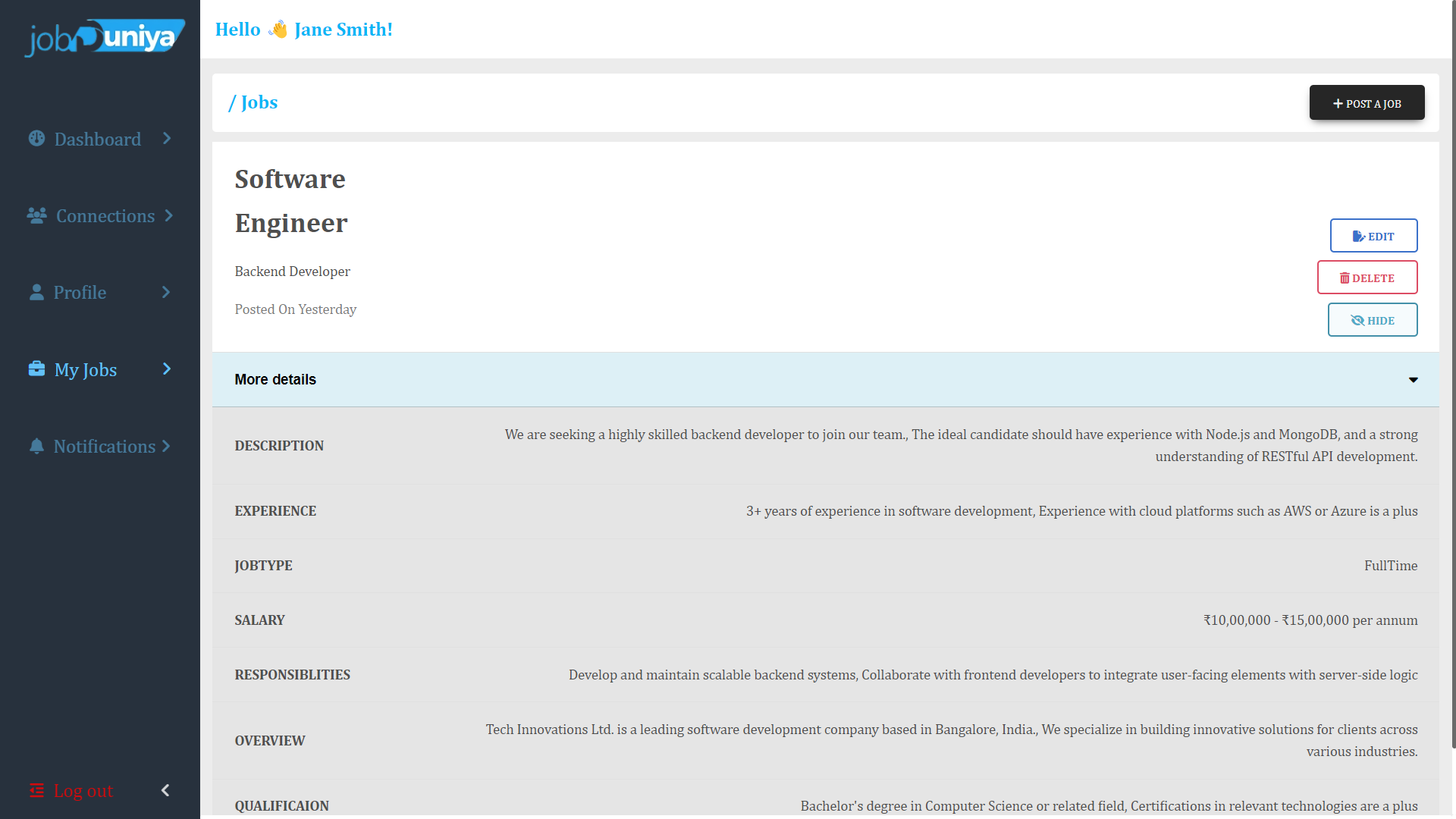






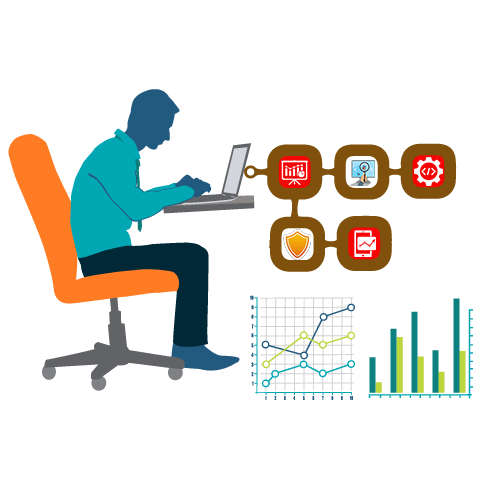




1. **Testing**

* **Software Testing**
* **Unit Testing**
* **System Testing**
* **Software Testing :-**
* **Software Testing:-**
* Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing is an exposure of a system to trial input to see whether the software meets the correct output. Testing cannot be determined whether the software meets the user’s needs, only whether it appears to conform to requirements. Testing can show that a system is free of errors, only that it contains errors. Testing finds errors, it does not correct errors. Software success is a quality product, on time and within cost. Testing can reveal critical mistakes. Testing should, therefore,
* Validate Performance.
* Detects errors.
* Identify.
* Inconsistencies.



* **Test Objective:**
* There Is Strong Evidence That Effective Requirement Management Leads to Overall Project Cost Savings. The Three Primary Reasons For This Are:
* Errors in requirement typically cost over 10 times more to repair than other errors.
* Requirement errors typically comprise over 40% of all errors in a software project.
* The Testing Procedure Should Care For All Of These, As Well, In Order To Attain A Flawless, Error-Free, And Efficient Functioning System; Too, Software Testing Is An Important Phase Of Any Software Development Life Cycle. Various Reports and Data Used For The Same Are The Core Of The System. The Testing, Therefore, Becomes Important In Order To Maintain The Cost As Well As Improve Performance And Consistency. The Testing Procedure For The System Has Been Divided Into Various Parts Ranging For Single Unit Testing To Entire System Testing.
* **Testing Principles:**
* All tests should be traceable to customer requirements.
* Tests should be planned long before testing begins.
* The Pareto principle applies to software testing.
* Testing should begin “in the small” and progress toward testing “in the large”.
* Exhaustive testing is not possible.
* **Unit Testing:**
* Unit Testing is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of the software. It usually has one or a few inputs and usually a single output.
* **Integration Testing:**
* **Integration Testing** is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.
* **Top-down integration:**
* Top-down integration testing is an integration testing technique used in order to simulate the behavior of the lower-level modules that are not yet integrated. Stubs are the modules that act as a temporary replacement for a called module and give the same output as that of the actual product.
* **Bottom-down integration:**
* Bottom-up testing is a specific type of integration testing that tests the lowest components of a codebase first. More generally, it refers to a middle phase in software testing that involves taking integrated code units and testing them together, before testing an entire system or code base.
* **Validation Testing :**
* The process of evaluating software during the development process or at the end of the development process to determine whether it satisfies specified business requirements.
* Validation Testing ensures that the product actually meets the client's needs. It can also be defined as demonstrating that the product fulfills its intended use when deployed in an appropriate environment.
* **System Testing:**
* System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic.
* **Recovery Testing :**
* Recovery testing is a type of non-functional testing technique performed in order to determine how quickly the system can recover after it has gone through a system crash or hardware failure. Recovery testing is the forced failure of the software to verify if the recovery is successful.
* **Security Testing :**
* Security testing is a process intended to reveal flaws in the security mechanisms of an information system that protect data and maintain functionality as intended.
* **Stress Testing :**
* Stress testing is a Non-Functional testing technique that is performed as part of performance testing. During stress testing, the system is monitored after subjecting system to overload to ensure that the system can sustain the stress.
* **Sanity Testing :**
* Sanity testing is the subset of regression testing and it is performed when we do not have enough time for doing testing. Sanity testing is surface-level testing where the QA engineer verifies that all the menus, functions, and commands available in the product and project are working fine.
* **White Box Testing :**
* White-box testing (also known as clear box testing, glass box testing, transparent box testing, and structural testing) is a method of testing [software](https://en.wikipedia.org/wiki/Software) that tests the internal structures or workings of an application, as opposed to its functionality (i.e. [black-box testing](https://en.wikipedia.org/wiki/Black-box_testing)).
* In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases.
* The tester chooses inputs to exercise paths through the code and determines the expected outputs.
* This is analogous to testing nodes in a circuit, e.g. [in-circuit testing](https://en.wikipedia.org/wiki/In-circuit_test) (ICT). White-box testing can be applied at the [unit](https://en.wikipedia.org/wiki/Unit_testing), [integration](https://en.wikipedia.org/wiki/Integration_testing), and [system](https://en.wikipedia.org/wiki/System_testing) levels of the software testing process.
* Although traditional testers tended to think of white-box testing as being done at the unit level, it is used for integration and system testing more frequently today.
* It can test paths within a unit, paths between units during integration, and between subsystems during a system-level test.
* Though this method of test design can uncover many errors or problems, it has the potential to miss unimplemented parts of the specification or missing requirements.

1. **Limitations and future scope of Enahancements, References**

**Limitations of the system:-**

* Since every system has some limitations so our proposed system is also not untouchable in this regard.
* The main limitations of our system are:
* Admin/company cannot update logo image once it is uploaded.
* Now, In our System User can not view the profile of other user.
* User and Admin/Company can not delete it’s profile itself.
* User can not build resume with our system

**Future scope of enhancements:-**

* The Job portal platform site has a wide range to change as per time requirements. That is the reason that not a particular project is ever considered complete because the demand of the user and thinking always change day by day.
* Always do more than what have.
* Some of the enhance which we have thought of are:
* We can give more advanced features for online job portal platform systems including more facilities.
* We will host the platform on an online server to make it accessible worldwide.
* Integrate multiple load balancers to distribute the load of the server.
* Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.
* Create a master and slave database structure to reduce the overload of the database queries.
* User can create his own resume building on our platform without using any other third party Application.
* We can provide facility to user and company/admin to manage profile with advanced security like 2 factor authentication like google.
* Development of mobile application which can run on multiple OS and devices.
* We can give facility of Master Admin who can handle both side company and user.
* We can give smart Notification facility of job application on mobile and browser so user can not be deprived from his/her opportunity for the job

* **References:-**

* https://linkedin.com/
* https://noukari.com/
* <https://upworks.in/>
* <https://apna.com>
* http:s//indeed.com