JOB PORTAL -User Manual

Introduction

The job portal project is designed to streamline the recruitment process for jobseekers, employers (companies), and administrators. The portal consists of three main modules: Admin, Company (Employer), and Jobseeker, each tailored to meet the specific needs of its users. This documentation provides a comprehensive guide to the functionalities of each module, detailing the workflows, user interactions, and API endpoints involved.

There are three Main Modules in the Job Portal

1.ADMIN MODULE

- **Admin dashboard**
- **Admin Profile**
- **View companies**
- **Company Profile**

  - **Active Jobs**
  - **Job History**

- **View Jobs**

  - **Job Details**
  - **Edit Job Details**

- **View jobseeker**

  - **Jobseeker Profile**

- **Notifications**

2.COMPANY/EMPLOYER MODULE

- **Company Search (Company Landing Page)**
- **View Job**
- **Post Jobs**
- **Profile**
- **Depts_Sector**

- **Add Sector**

  - **My Jobs**

    - **Edit Job**
    - **Applicants**
      - **User Profile**

3.JOB SEEKER MODULE

  - **Job Seeker Search (Job Seeker Landing Page)**

    - **View Job Details**

  - **Job Seeker Profile**
  - **Job Seeker Education**

    - **Add Education**

  - **Job Seeker Experience**

    - **Add Experience**

  - **Applied Jobs**
  - **Saved Jobs**

4.Navbar and Sidebar.

Login and Register Component

1. Landing Page (Login Page)

- **Initial Render**: The first page that users see is the login page.
- **Options**:
  - **"Sign In" Link**: Click here if you are a registered user to proceed with logging in.
  - **Email Input**: New users are prompted to enter their email address.

**2. Registration Workflow**

2.1 User Type Selection

- After entering the email address, the user must select between two options:
  - **Jobseeker**
  - **Employer**
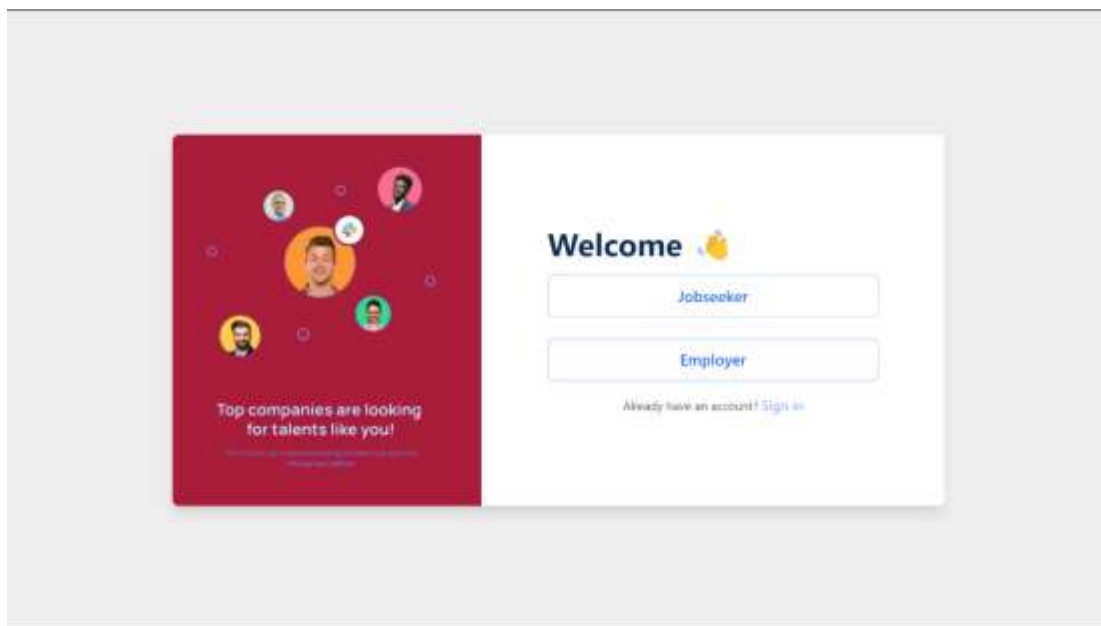
2.2 Jobseeker Registration

1. **Username & Password**:

  - Enter a username and password.
  - **Password Requirements**: Minimum 8 characters, one special character, and one number.
  - **Button**: Register
    - **Action**: Displays a success message and redirects to the personal information page.
    - **API Endpoint**: POST /authentication/register
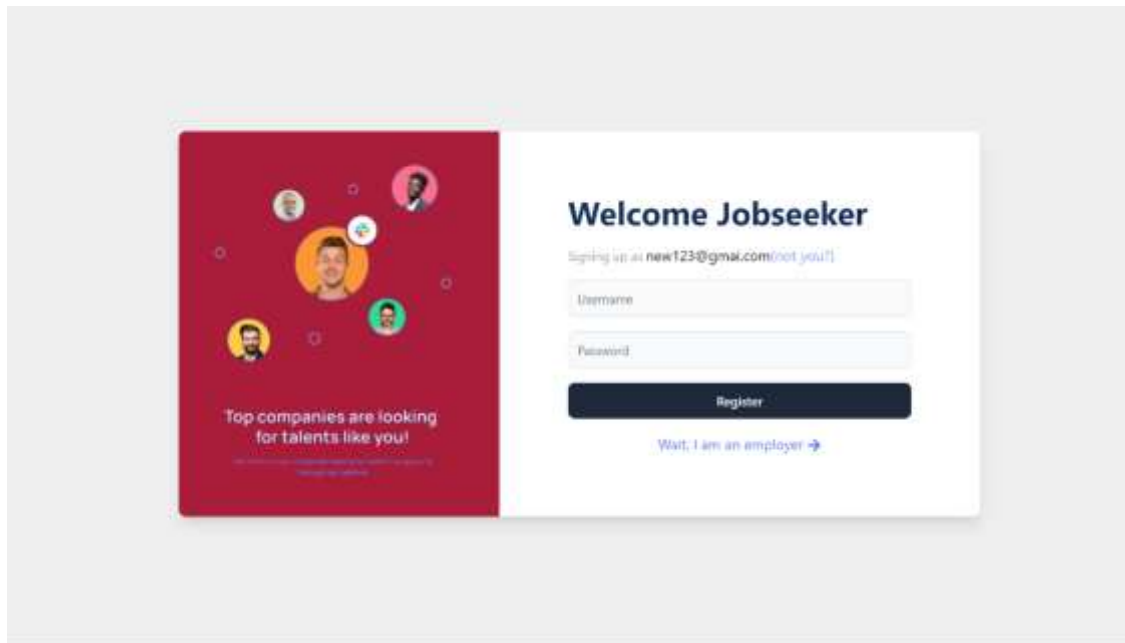
2. **Personal Information**:

   o Fields: Full name, Date of Birth, Mobile, Address 1, Address 2, City, State, Pincode, Profile Image.

   o **All fields are required**.

   o **Button**: Next

     ▪ **Action**: Redirects to the education details page.

     ▪ **API Endpoint**: POST /authentication/register

3. **Education Details**:

   o Fields: Course, Institution Name, From Date, To Date, Education Document.

   o **Button**: Add Education

      ▪ **Action**: Adds the entered education details in real-time and displays them.

      ▪ **API Endpoint**: POST /authentication/register

   o **Button**: Next

      ▪ **Action**: Redirects to the experience details page.

Job Portal

Education

Add Education   Education History

Course

Institution Name

From
dd-mm-yyyy

To
dd-mm-yyyy

Education Document:
Choose File   No file chosen

Add Education +

Next →

4. **Experience Details**:

- o Fields: Job Position, Organization Name, Job Description, From Date, To Date, Experience Document.

- o **Button**: Add Experience

  - **Action**: Adds the entered experience details in real-time and displays them.

  - **API Endpoint**: POST /authentication/register

- o **Button**: Next

  - **Action**: Redirects to the job preferences page.

Job Portal

Experience

Add Experience   **Experience History**

Job Position

Organization Name

Job Description

From
dd-mm-yyyy

To
dd-mm-yyyy

Choose File   No file chosen

Add Experience +

Next →

5. **Job Preferences**:

- Fields: Skills, Job Category, Resume Upload.

- **Button**: Submit

  - **Action**: Displays a "Profile Created Successfully" message and redirects to the login page.

  - **API Endpoint**: POST /authentication/register

2.3 Employer/Company Registration

1. **Username & Password**:

   o Enter a username and password.

   o **Password Requirements**: Minimum 8 characters, one special character, and one number.

   o **Button**: Register

      ▪ **Action**: Displays a success message and redirects to the company information page.

      ▪ **API Endpoint**: POST /authentication/register

2. **Company Information**:

- Fields: Company Name, Mobile, Address 1, Address 2, City, State, Pincode, Company Logo.

- **All fields are required**.

- **Button**: Continue

  - **Action**: Redirects to the sector and department selection page.

  - **API Endpoint**: POST /api/employer/company-info

3. **Sector & Department**:

- o Fields: Sector, Department.

- o **Button**: Add Sector/Department

  - **Action**: Adds the entered sector and department in real-time and displays them.

  - **Button**: Create Sector/Department

  - **Action**: Creates a new sector or department based on user input.

  - **API Endpoint**: POST /authentication/register

- o **Button**: Submit

  - **Action**: Displays a "Company Profile Created Successfully" message and redirects to the login page.

3. Login Workflow

   ▪ **The user uses the registered username/email and password to login**

3.1 Jobseeker Login

   • **Button**: Login

      ○ **Action**: Authenticates the user and redirects to the Jobseeker dashboard.
      ○ **API Endpoint**: POST /authentication/login

3.2 Employer Login

   • **Button**: Login

      ○ **Action**: Authenticates the user.

- **Condition**: If the account is not activated by the admin, an error message "Account Not Activated" is displayed.

- **API Endpoint**: POST /authentication/login

- **Admin Approval**: Employers can only log in after the admin approves and activates their account.

3.3 Admin Login

- **Button**: Login

  - **Action**: Authenticates the user and redirects to the Admin dashboard.

  - **API Endpoint**: POST /authentication/login
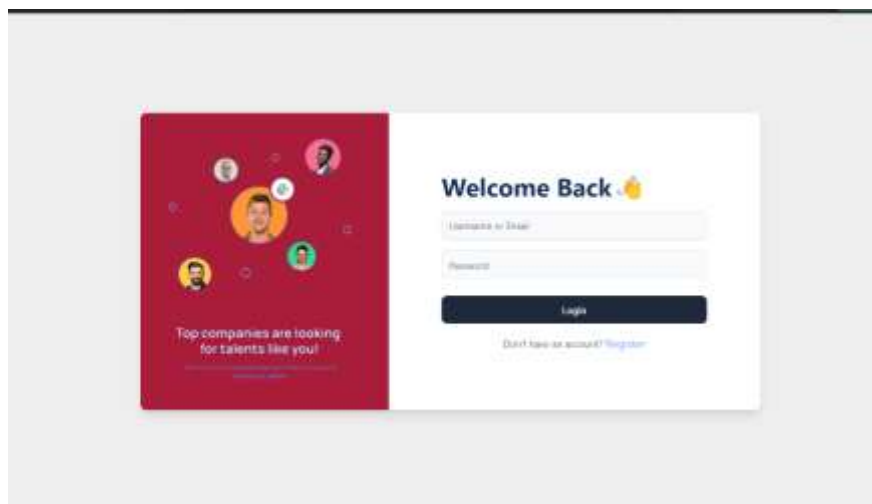
4. Admin Module

The Admin module is responsible for overseeing the entire job portal, managing companies, job categories, sectors, and departments. It ensures that the job postings adhere to the platform's guidelines and facilitates communication between jobseekers and employers.
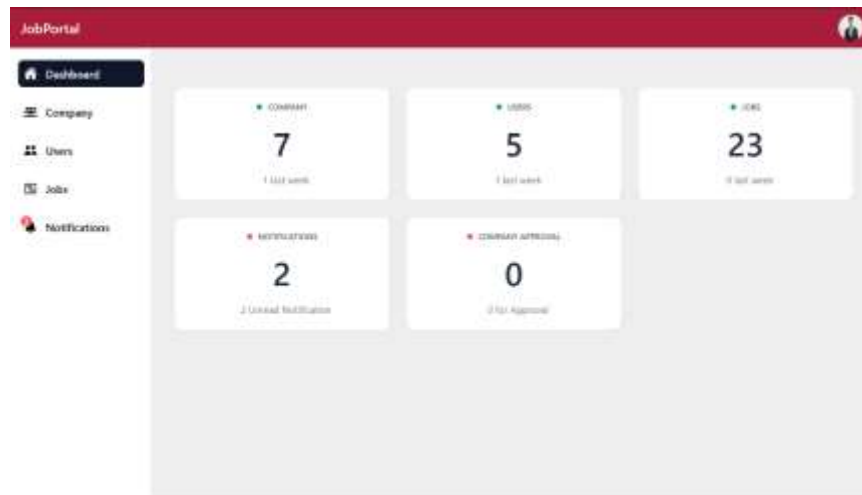
4.1. Admin Dashboard

After logging in, the admin is redirected to the dashboard, which includes two main components:

- **Navbar**:

  - Displays the admin's profile image.
  - **Profile Image Click**:
    - **Dropdown Options**:
      - **Profile**: Redirects to the profile update page.
      - **Sign Out**: Logs out the admin, removes the token, and redirects to the login page.
  - **API Endpoint**: POST /adminn/adminprofileview/

- **Sidebar**:

  - Contains the following tabs: Dashboard, Company, Users, Jobs, Notification.

4.1.1. Dashboard Overview

- **Dashboard**: The default page after login, displaying various statistics:

  - Number of registered companies.
  - Number of companies registered last week.
  - Number of registered jobseekers.
  - Number of jobseekers registered last week.
  - Number of jobs posted.
  - Number of jobs posted last week.
  - Total notifications.
  - Total unread notifications.
  - Total companies waiting for approval.
  - **API Endpoints:**

    - **API Endpoint**: GET /adminn/getalljobs/
    - **API Endpoint**: GET /adminn/getallcompany/

- **API Endpoint**: GET /adminn/getalljobseeker/

- **API Endpoint**: GET /adminn/getallnotification/

4.1.2.Navbar

Overview

The AdminNav component serves as the navigation bar for the admin panel of the JobPortal. It provides quick access to various sections like the dashboard, company views, job seeker views, job listings, and notifications. The component includes user profile management and a logout option. The design is responsive, utilizing a hamburger menu for smaller screens.

Imports

1.React and Hooks:

- Essential for creating the component, managing state, and handling side effects.

2.Flowbite Components:

- Navbar, Dropdown, Sidebar, Drawer are used for UI elements.

3.React Icons:

- Icons from various libraries for visual elements.

4.MakeApiRequest:

- Custom function to make API calls.

5.ProfileContext and NotificationContext:

- Contexts for managing and accessing user profile data and notifications.

State Management

1.isOpen:

- Manages the state of the drawer, determining if it is open or closed.

2.profile:

- Manages the user's profile information, fetched from the API.

Context

1.ProfileContext:

- Provides access to the user's profile information.

2.NotificationContext:

- Provides access to the notifications, including the count of unread notifications.

API Requests

1.Profile Data:

- The component fetches the admin's profile data using MakeApiRequest in a useEffect hook, which runs once when the component is mounted. The API call is authenticated using the Authorization header with the JWT token stored in cookies.

Event Handlers

1.handleLogout:

- Logs out the user by removing the JWT token from cookies and clearing the local storage.

2.handleProfileClick:

- Navigates the user to the admin profile page.

3.handleClose:

- Closes the drawer.

Rendering

1.Navbar:

- The top navigation bar includes the JobPortal logo, hamburger menu for mobile views, and a profile dropdown for accessing profile settings or logging out.

2.Drawer:

- A sidebar menu that is rendered for mobile views when the hamburger menu is clicked. Contains links to different admin panel sections.

3.Sidebar:

- Only visible on larger screens, providing quick access to dashboard, company views, job seeker views, job listings, and notifications.

Styling

Uses Tailwind CSS classes for styling elements with conditional classes based on the active route to highlight the current section.

Workflow

1.Profile Data Fetching:

- Upon mounting, the component fetches the admin's profile data.

2.Navigation Handling:

- The user can navigate to different sections of the admin panel using either the navbar on large screens or the drawer on smaller screens.

3.Profile Dropdown:

- The user can access their profile or logout via the dropdown menu in the navbar.

4.Notifications:

- The notifications section shows a count of unread notifications and allows the admin to access them.

4.1.3SideBar

Overview

The AdminSideBar component provides a static sidebar for the admin panel of the JobPortal. It contains links to the dashboard, company views, job seeker views, job listings, and notifications.

Imports

1.React and Hooks:

- Essential for creating the component, managing state, and handling side effects.

2.React Icons:

- Icons from various libraries for visual elements.

3.NotificationContext:

- Context for managing and accessing notifications.

Context

1.NotificationContext:

- Provides access to notifications and a method to update them.

API Requests

1.Notifications:

- The component uses the updateNotification function from NotificationContext to fetch the latest notifications and update the count of unread notifications.

Rendering

1.Sidebar Menu:

- A vertical menu containing links to different admin panel sections such as the dashboard, company views, job seeker views, job listings, and notifications. Each link includes an icon and a label, with conditional styling based on the active route.

Styling

Uses Tailwind CSS classes for styling the sidebar and links, with conditional classes to highlight the active route.

Workflow

1.Notification Fetching:

- Upon mounting, the component fetches the latest notifications and updates the notification count.

2.Navigation Handling:

- The user can navigate to different sections of the admin panel using the sidebar links.

3.Notifications:

- The notifications section shows a count of unread notifications, and the user can click the link to view them.

4.2. Admin Profile Management

- **Profile Update**:

  - Admin can edit personal details: name, address, phone, city, state, pincode.
  - **Button**: Save Changes

    - **If No Changes Detected**: Displays warning message "No changes detected."
    - **If Changes Are Made**: Displays success message "Profile updated successfully."
    - **API Endpoint**: GET /adminn/adminprofileview/
    - **API Endpoint**: POST /adminn/adminprofileview/

- **Update Credentials**:

  - **Button**: Update Credential

- Opens a modal to edit email and username.

- **Validation**: Both fields are required, and inputs are checked against the database for uniqueness.

- **API Endpoint**: PUT /company/users/
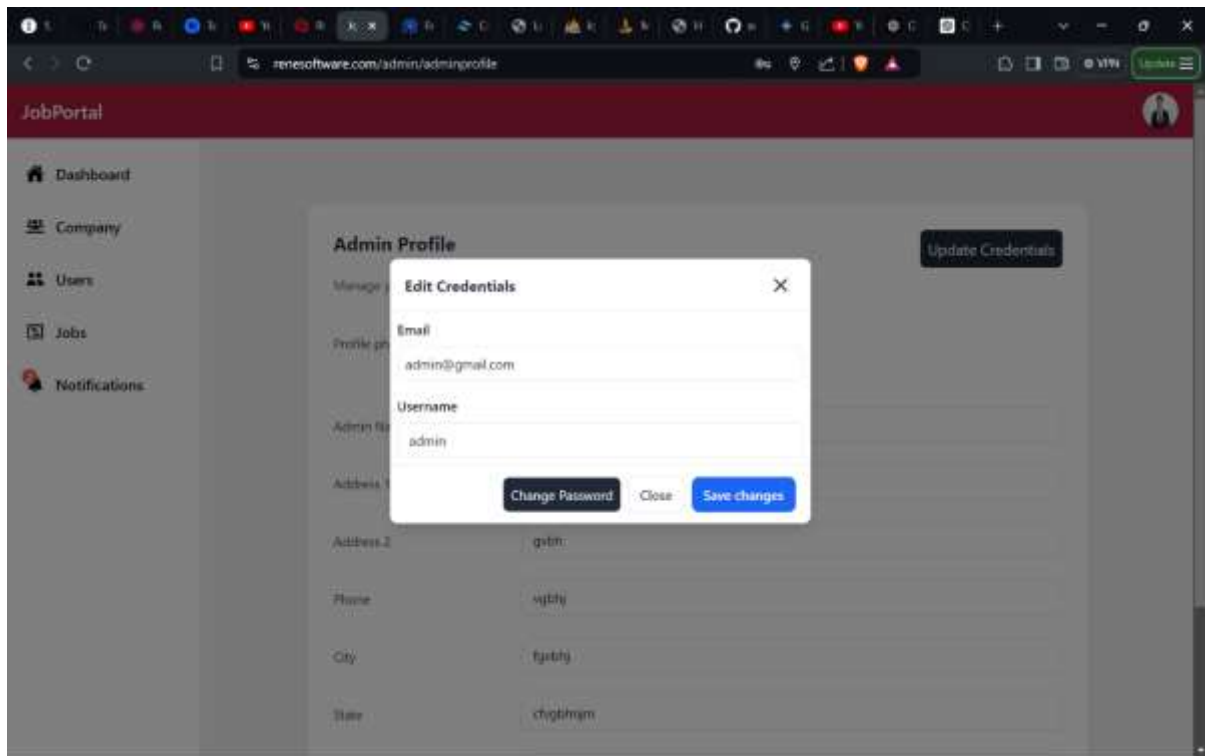
- **Change Password**:

  - **Button**: Change Password

    - Opens a modal to update password with fields for old password, new password, and confirm password.

    - **Validation**: The "Save Changes" button is disabled if new password and confirm password do not match.

    - **Old Password Check**: If incorrect, displays message "Old password incorrect."

    - **If Successful**: Saves the new password and closes the modal automatically.

    - **API Endpoint**: PUT /company/passwordchange/
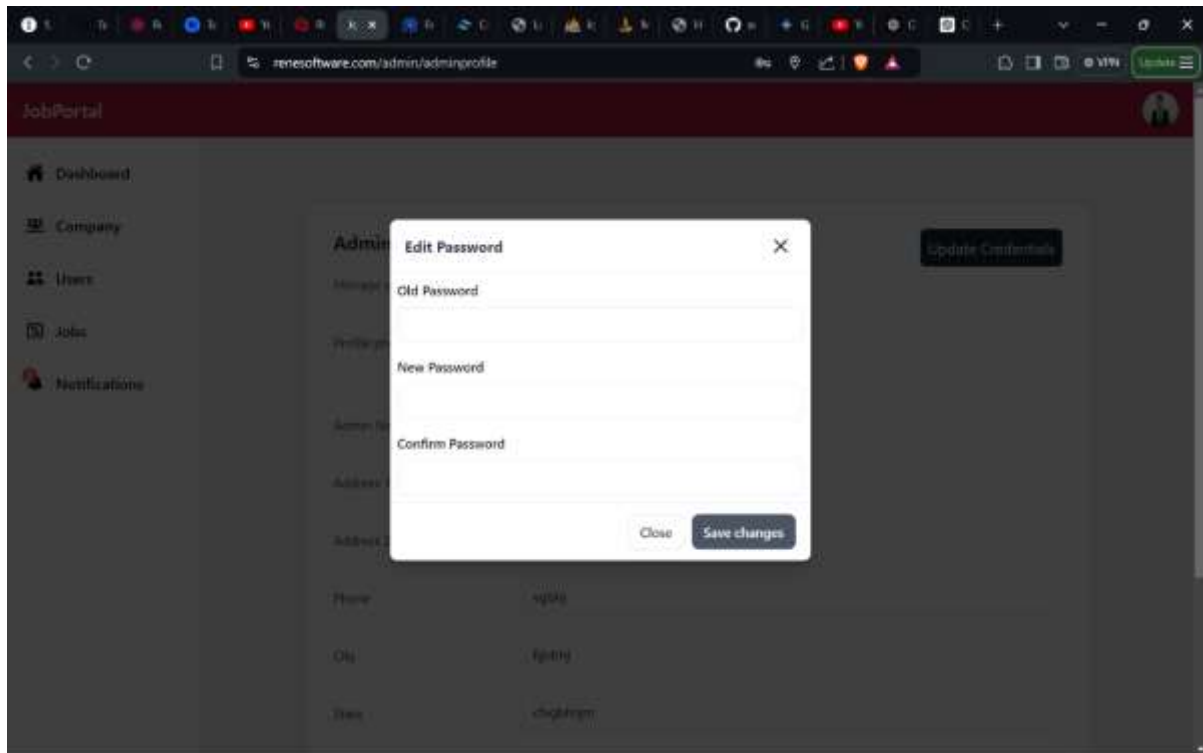
4.3. Company (list all Companies)

- **Company Tab**:

    - Displays a list of registered companies in a table format.

    - Fields: Company Name, State, Mobile, Status, Created On, View, Delete.

    - **Pagination**: Displays 5 companies per page with "Next" and "Prev" buttons.

- **View Company Details**:

    - **Button**: View

        - Opens a detailed page with company information, active jobs, and job history.
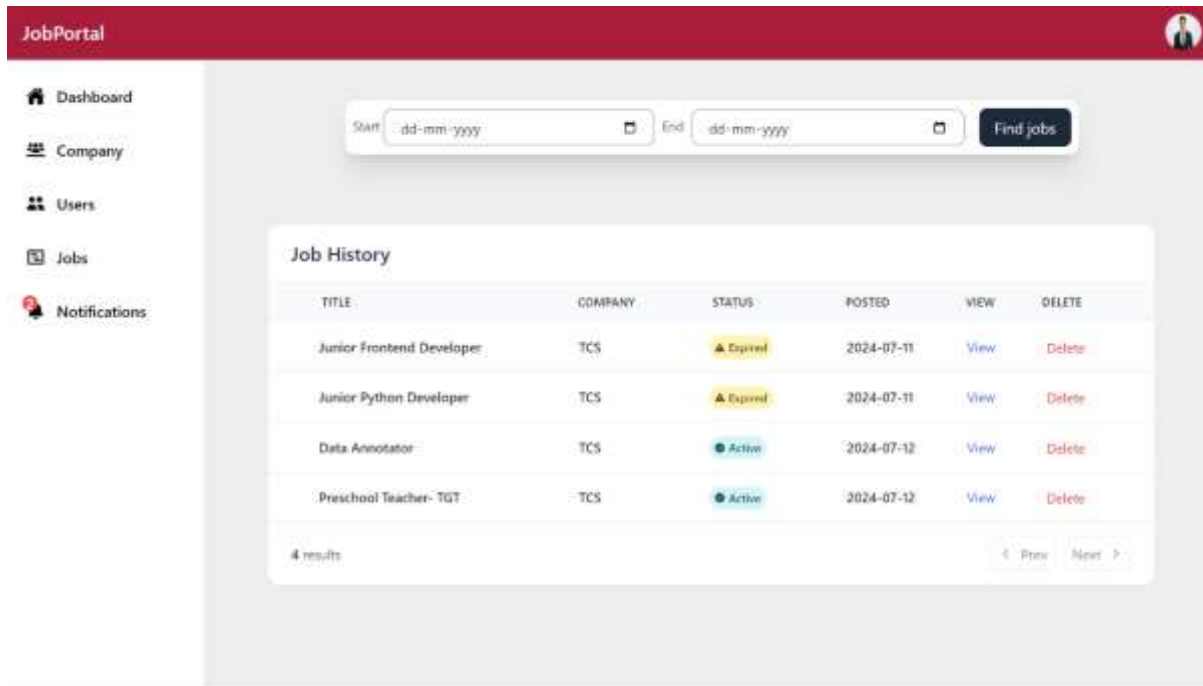
            - **API Endpoint**: GET/company/companyprofileview/

- **Active Jobs**:

  - Shows all active jobs posted by the company.

  - Allows admin to search jobs by start date and end date, view, edit, or delete jobs.

  - **API Endpoint**: GET/ adminn/limitfilteredjobsview
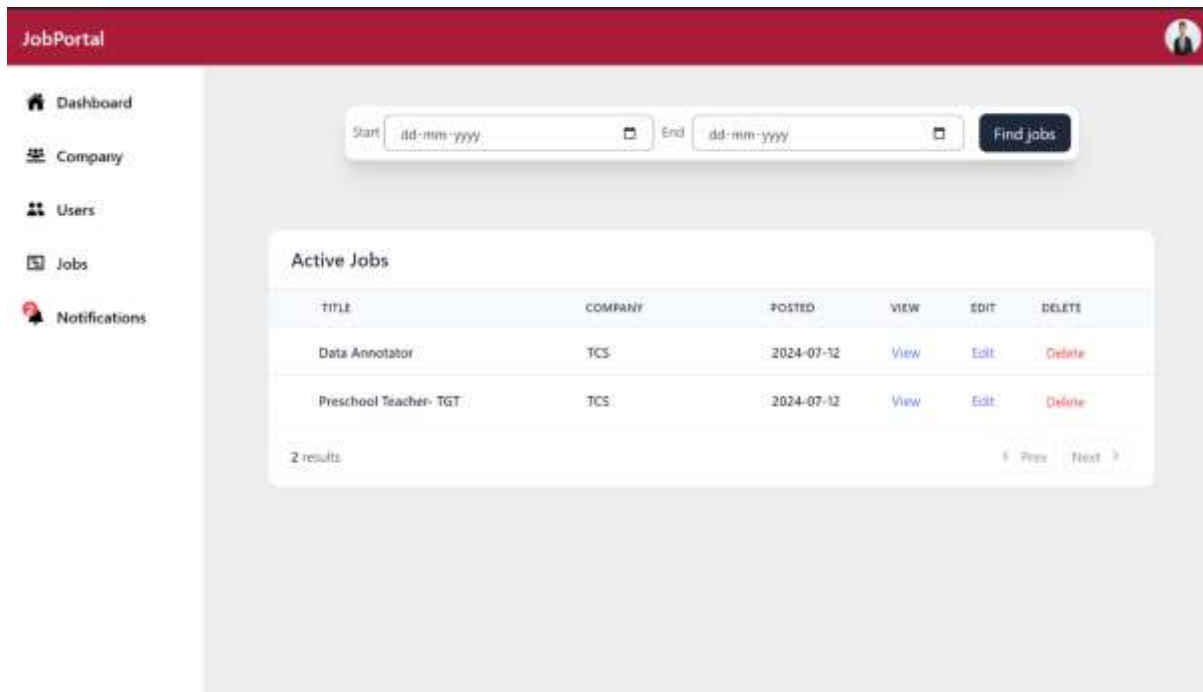
- **Job History**:

  - Shows all jobs posted by the company, including expired and deleted jobs.

  - Same search and management functionality as active jobs.

 - **API Endpoint**: GET/adminn/limitfilteredjobshistory/

- **Edit Company Details**:

    - Admin can edit company details and update credentials.
    - **API Endpoint**: PUT/company/companyprofileview/ (Profile details)
    - **API Endpoint**: PUT/company/users/ (update Credentials)
    - **API Endpoint**: PUT/ company/passwordchange/ (update Password)

- **Add Department & Sector**:

    - **Button**: Add Dept & Sectors

        - Opens a modal to create new sectors and departments or delete existing ones.
        - Notifications are sent when a company creates a new sector or department, which the admin can approve or reject.

- **API Endpoint**: GET /company/getcompanysector/ (get sector)

- **API Endpoint**: GET /company/getdepartments/ (get department)

- **API Endpoint**: DELETE /company/getcompanysector/ (delete sector)

- **API Endpoint**: DELETE/company/getdepartments/ (delete departments)

- **API Endpoint**: PUT/company/addsectordepartments (add departments and sector)
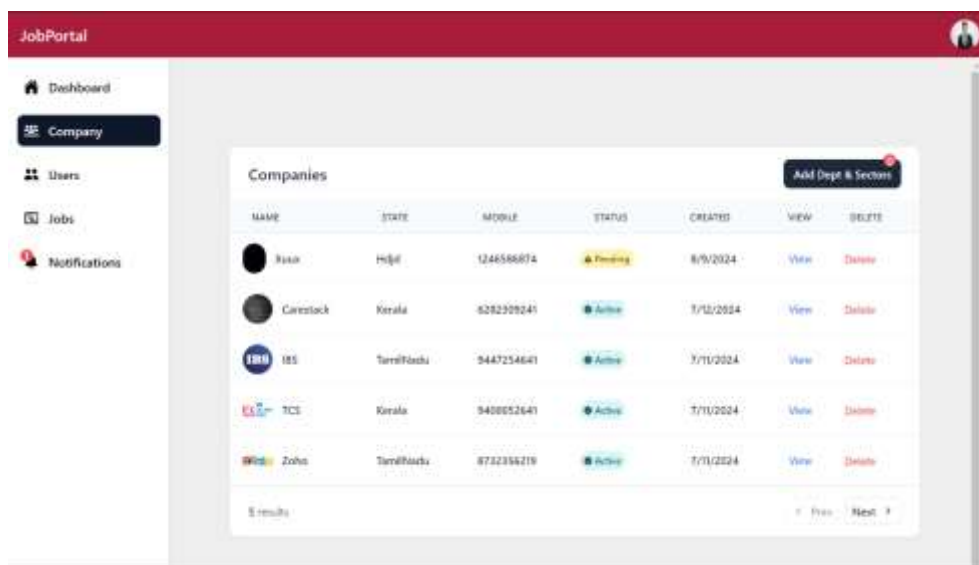
4.4. Users (list all users)

- **Users Tab**:

  - Displays a list of registered jobseekers in a table format.

  - Fields: Name, State, Mobile, Created On, View, Delete.

  - **Pagination**: Displays 5 users per page with "Next" and "Prev" buttons.

  - **Newly Registered Users**: Shown in ascending order, with the most recent registrations first.

  - **API Endpoint**: GET/adminn/limitjobseekerview/

- **View User Details**:

- o **Button**: View

  - Opens a detailed page with user information including Full Name, DOB, Address, Phone, City, State, Pincode, Job Category, Resume, and Skills.

  - Admin can edit these details, similar to company management.

  - **API Endpoint**: GET/company/users/ (get user details)

  - **API Endpoint**: PUT/ jobseeker/jobseekerprofileview/ (update user details)

- **Delete User**:

  - o **Button**: Delete

    - Shows a confirmation modal to delete the user.

    - **If Confirmed**: Deletes the user.

    - **API Endpoint**: DELETE /adminn/getalljobseeker/

4.5. Jobs(list all jobs)

- **Jobs Tab**:

  - Displays a list of jobs posted on the portal, with the most recently created jobs shown first.

  - Fields: Job Title, Company, Posted Date, View, Edit, Delete.

  - **Pagination**: Displays 5 jobs per page with "Next" and "Prev" buttons.

  - **API Endpoint**: GET /adminn/limitjobsview/ (get all jobs)

- **View Job Details**:

  - **Button**: View

    - Opens a detailed page with job information.
    - **API Endpoint**: GET /company/viewjob/

- **Edit Job**:

  - **Button**: Edit

    - Allows admin to edit job details.
    - **API Endpoint**: PUT/company/postjob/

- **Delete Job**:

  - **Button**: Delete

    - Shows a confirmation modal to delete the job.
    - **If Confirmed**: Deletes the job.
    - **API Endpoint**: DELETE /adminn/limitjobsview/

- **Filter Jobs**:

- o Admin can filter jobs by company name, job title, or by date range (start date and end date).

- o **API Endpoint**: GET /adminn/limitjobsview

- **Job Categories**:

  - o Admin can add, view, and delete job categories.

  - o If a company creates a new job category, it triggers a notification for the admin to approve or reject.

  - o **API Endpoint**: GET /company/jobcategory/
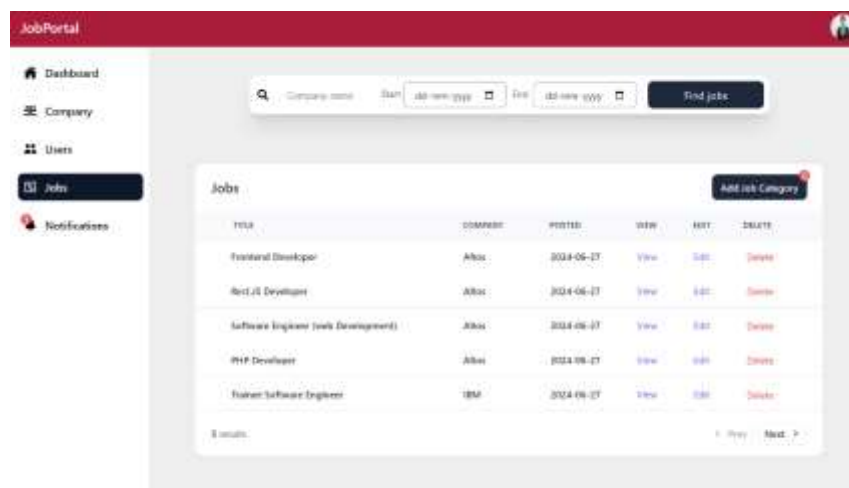
4.6. Notifications

- **Notification Tab**:

  - o Displays all notifications, with unread notifications highlighted in blue and read notifications in white.

  - o **Individual Notification Management**:

- **Mark as Read**: Click to mark a notification as read, which updates the count in the sidebar and dashboard in real-time.

- **Delete Notification**: Click the cross icon to delete individual notifications.

- **Clear All**: Button to clear all notifications.

o **Notification Types**:

- **Newly Registered Company**: Notification for each new company registration.

- **Department Notification**: Triggered when a company creates a new department; admin can approve or reject from the notification itself.

- **Sector Notification**: Similar to department notifications but for sectors.

- **Job Category Notification**: Triggered when a company creates a new job category; admin can approve or reject from the notification itself.

o **API Endpoints**:

- **Mark as Read**: PUT/adminn/notificationn_readed/

- **Delete Notification**: DELETE /adminn/deletenotification/

- **Clear All Notifications**: DELETE /adminn/deletenotification/

- **Get all Notifications**: GET/adminn/getallnotification/
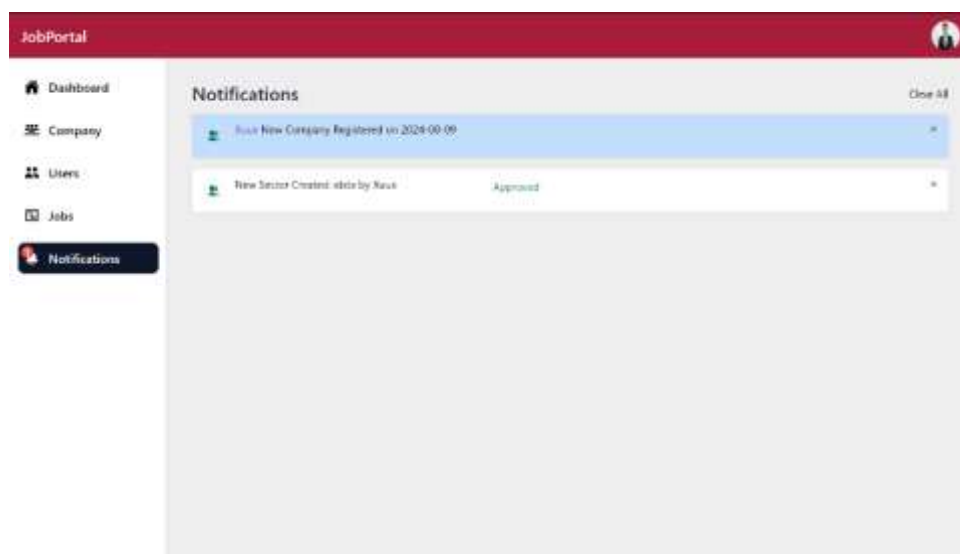
5. Company/Employer Module

The Company module allows employers to post jobs, manage job applications, and maintain their company profiles. It provides a seamless interface for managing job postings, reviewing applicants, and receiving notifications regarding administrative actions on categories, sectors, or departments.

When a company logs in:

- **Redirection: The company is redirected to the company dashboard, displaying jobs posted by the company.**

- **Initial Job Load: The most recent jobs are displayed first, with pagination functionality showing 6 jobs per page. The company can navigate between pages using "Next" and "Prev" buttons.**

5.1.Actions:

- **Filter Jobs: The company can filter jobs by job title, company name, keywords, location, or state.**

- **Find Jobs: After applying filters, clicking the "Find Jobs" button fetches and displays filtered jobs.**

    API Endpoint:

- **Filter Jobs: GET / company/getallpostedjobs/**

5.2. Company Dashboard Navbar

The company dashboard navbar includes three tabs, a bell icon for notifications, and a profile image.

- **Tabs:**

    1. **Home: Redirects to the company's main job listing page.**

    2. **Post Job: Opens the job posting form.**

3. **My Jobs: Displays all jobs posted by the company.**

Links:

- **Home: Link to="/employer/employersearch"**
- **Post Job: Link to="/employer/postjob"**
- **My Jobs: Link to="/employer/myjobs"**
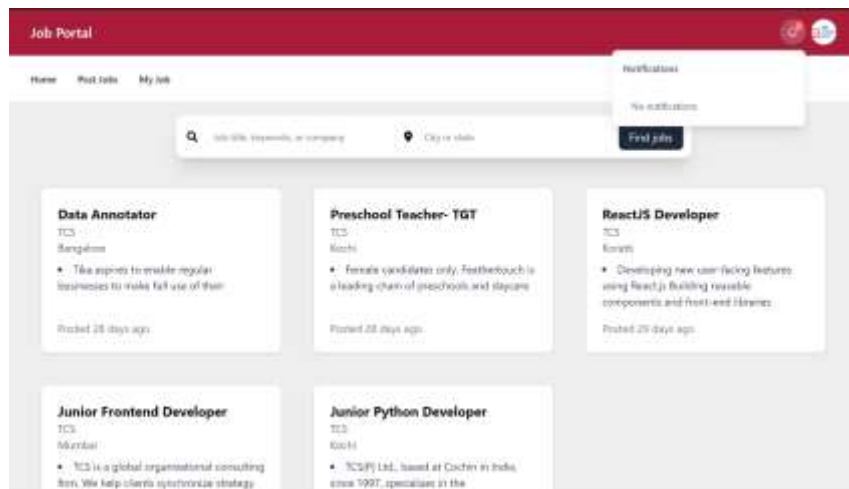
5.3.Notifications

- **Bell Icon: Shows unread notification count.**

  - **Notifications: Unread notifications are highlighted in blue, while read notifications are in white. Notifications can be deleted by clicking the cross icon.**

  - **Notification Types: Include updates on job categories, sectors, and departments created by the company that require admin approval.**

API Endpoint:

- **Fetch Notifications: GET / company/getallcompanynotification/**
- **Update Notification Status: PUT / adminn/notificationn_readed/**
- **Delete Notification: DELETE / adminn/deletenotification/**

5.4. Company Profile Management

Profile Image Dropdown: Options include "Profile," "Depts & Sectors," and "Logout."
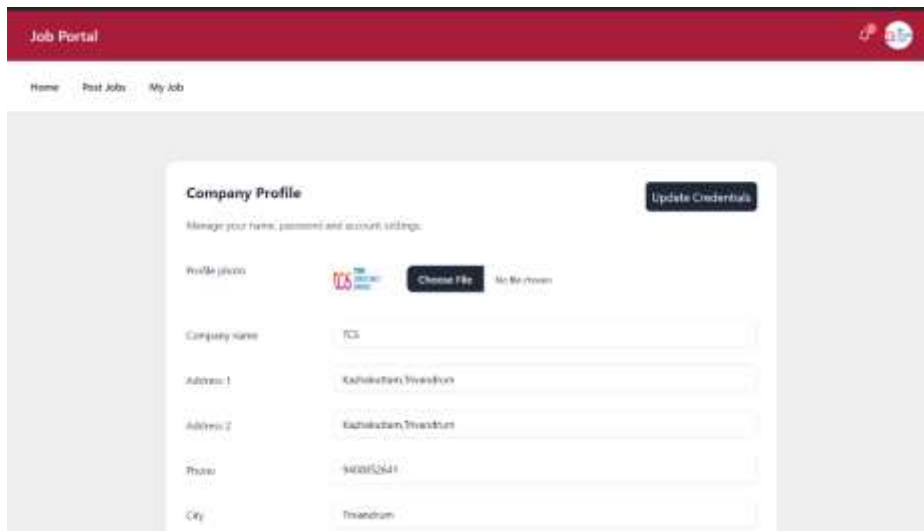
1. **Profile**:

   o **Update Profile**: Clicking on "Profile" opens the company profile page. Here, the company can update its information, including:

      ▪ **Company Name**

      ▪ **Address1 & Address2**

      ▪ **Phone Number**

      ▪ **City**

      ▪ **State**

      ▪ **Pincode**

   o **Save Changes(button)**:

      ▪ **Action**: After making any changes, the user clicks on the "Save Changes" button.

      ▪ **API Endpoint**: PUT / company/companyprofileview/

      ▪ **Behavior**:

         ▪ If no changes are detected, a warning message is displayed: "No changes detected."

         ▪ If changes are made and saved successfully, a success message appears: "Profile updated successfully."

   o **Update Credentials (includes Email, Username, Password)**:

      ▪ **Action**: There is a button labelled "Update Credentials" on the profile page.

- **Functionality**: Clicking on this button opens a modal where the company can edit its email and username.

- **Validation**: Both fields are required, and the system checks if the new email or username already exists in the database.

- **API Endpoint**: PUT / company/users/

- **Change Password**:

  - **Action**: A "Change Password" button is also available on the update Credential modal.

  - **Functionality**: Clicking on this button opens another modal with fields for:

    - **Old Password**

    - **New Password**

    - **Confirm Password**

  - **Validation**: The "Save Changes" button is disabled if the new password and confirm password fields do not match.

  - **API Endpoint**: PUT/ company/passwordchange/

  - **Behavior**:

    - If the old password is incorrect, an error message appears: "Old password incorrect."

    - If correct, the new password is saved, and the modal closes automatically.

2. **Depts & Sectors**:

- **Manage Departments and Sectors**: Clicking on this option takes the user to the "Depts & Sectors" page.

- **Functionality**:

  - The page lists all the sectors and departments associated with the company.

  - The company can add new departments to existing sectors, delete departments from sectors, or delete entire sectors.

  - **Add Sector**: There is a button to add a new sector. Clicking this button redirects the user to a form where they can add multiple sectors at once. After adding, the user is redirected back to the "Depts & Sectors" page.

  - **API Endpoints**:

    - **Add Sector**: POST / company/updatesectoranddepartments/

    - **Add Department**: POST / company/addnewdepartment/

    - **Delete Sector**: DELETE / company/getcompanydepartment/

    - **Delete Department**: DELETE / company/deletedepartment/

3. **Logout**:

- o **Action**: Clicking "Logout" will log the company out of the system.
- o **Behavior**:

  - ▪ The system removes the authentication token.
  - ▪ The user is redirected to the login page.

5.5.Navbar Tabs

The company dashboard features three main tabs: "Home," "Post Job," and "My Jobs."

1. **Home**:

   - o **Default Page**: The "Home" tab is the default landing page when a company logs in.
   - o **Job Listing**: This page displays all jobs posted by the company, sorted by the most recent jobs first.
   - o **Job Details**:

     - ▪ **Action**: Clicking on a job post opens a detailed view of that job.
     - ▪ **Functionality**: The company can view all relevant details of the job.

   - o **Filtering Jobs**:

     - ▪ **Functionality**: The company can filter jobs using various criteria such as:

       - ▪ Job Title
       - ▪ Company Name
       - ▪ Keywords
       - ▪ Job Location (City)
       - ▪ Job Location (State)

     - ▪ **Action**: After setting the desired filters, the user clicks the "Find Jobs" button to view the filtered list.
     - ▪ **API Endpoint**: GET / company/getallpostedjobs/

2. **Post Job**:

- o **Job Posting Form**: This tab opens a page where the company can post a new job.

- o **Required Fields**:

  - Job Title

  - Job Category (With an option to create a new category, which sends a notification to the admin for approval)

  - Job Nature

  - Vacancy

  - Salary (Must be greater than 20,000; otherwise, an error message is shown)

  - Location

  - State

  - Description

  - Responsibility

  - Qualification

  - Experience

  - Keywords (Separated by commas)

  - Company Details (Name, Location, Website)

- o **Live Validation**: The form features live validation for all fields.

- o **Submit Job**:

  - **Action**: After filling in all the required details, the user clicks the "Submit Job" button.

  - **Behavior**: Upon successful submission, a success message is displayed, and the user is redirected to the "My Jobs" page.

  - **API Endpoint**: POST / company/postjob/

3. **My Jobs**:

- ○ **Job Listing**: This tab displays all jobs posted by the company in a table format.

- ○ **Table Columns**:

  - ▪ Job Title

  - ▪ Job Created Date

  - ▪ Applicants (Number of job seekers who applied for the job)

  - ▪ Action (With options to view, edit, delete, or see applicants)

- ○ **Pagination**: The list includes pagination functionality to navigate through multiple jobs.

- ○ **Filter Jobs**:

  - ▪ **Action**: The company can filter jobs by setting a start date and end date to display jobs posted within that range.

  - ▪ **API Endpoint**: GET / company/postjob/

- ○ **Actions in "Action" Column**:

    1.View Job:

- **Action**: Opens a detailed view of the job in a separate page.
- Navigate(`/employer/jobdetails/${jobid}`)

2.Edit Job:

- **Action**: Allows the company to edit the job details. All fields must be filled for the changes to be saved.
- Navigate(`/employer/editjobs/${jobid}`)

3.Delete Job:

- **Action**: Clicking this option opens a confirmation modal before deleting the job.
- **API Endpoint**: DELETE/ company/viewjob/

4.View Applicants:

- **Action**: Opens a page listing all the job seekers who applied for the job.
- Navigate(`/employer/applicants/${jobid}          `)

  Table Columns:

  Name: Name of the job seeker.

  Title: Job title they applied for.

  Applied On: Date the job seeker applied.

  Profile: A "View" link to see the job seeker's profile.

  Resume: A link to view and download the job seeker's resume.

- **API Endpoint**: GET / company/filterapplicants/

- **Profile View**: navigate(`/employer/jobseekerprofile/${userid}`);
- **Download Resume**: DownloadResume()
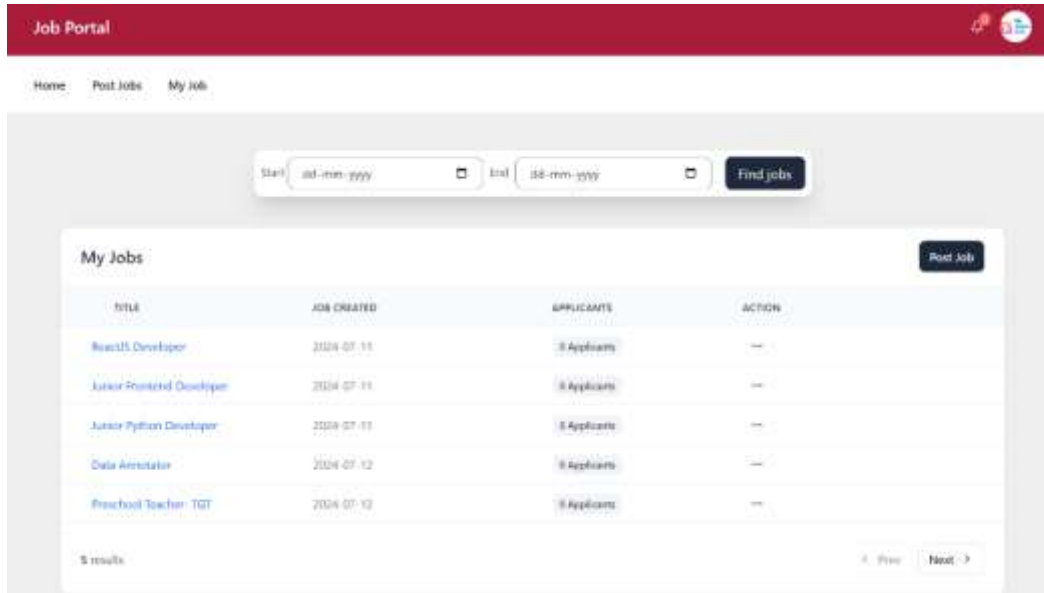
6. Job Seeker Module

The Jobseeker module is designed for individuals seeking employment. It enables users to browse and filter job listings, apply for jobs using an existing or newly uploaded resume, and save jobs for future consideration. Jobseekers can also manage their profiles, including education and experience details

6.1. Login and Initial Load

Functionality:

- **Upon login, if the user is identified as a Jobseeker, they are redirected to the Jobseeker homepage.**
- **The homepage loads the most recent job postings with pagination (6 jobs per page). The jobs are displayed in descending order, with the most recent jobs shown first.**

Pagination:

- **Next Button: Loads the next 6 jobs.**

- **Previous Button: Loads the previous 6 jobs.**

Filtering Jobs:

- **Jobseekers can filter jobs by job title, company name, keywords, job location, or job location state.**

- **Find Jobs Button: After entering the filter criteria, clicking this button triggers the job search and displays the filtered results.**

API Endpoints:

- **Get Job Listings: GET / jobseeker/getalljobs/**
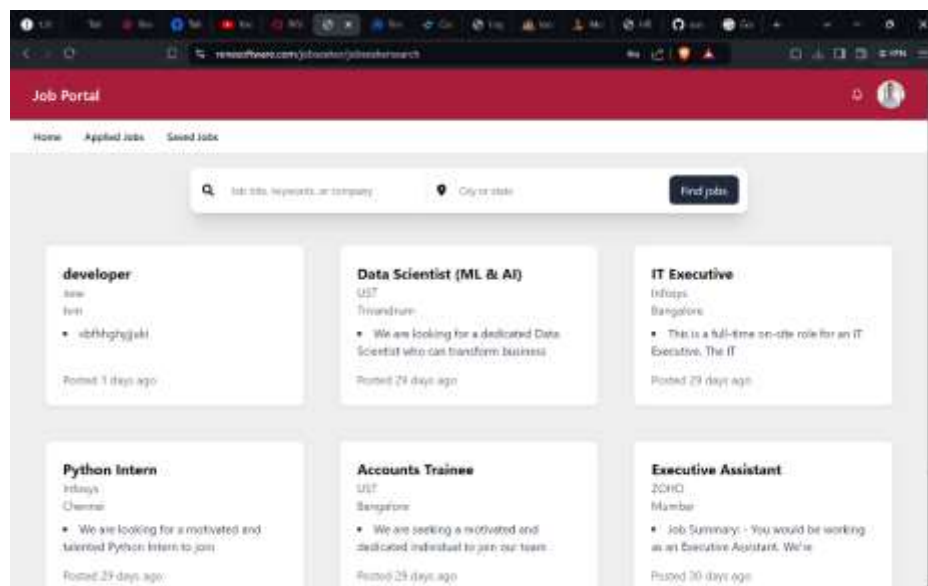
6.2. Viewing Job Details

Navigate(`/jobseeker/jobdetails/${jobid}`)

Functionality:

- **When a job post is clicked, it opens a detailed job description page.**

Buttons:

1. **Apply Now:**

   o **Modal: Presents two radio options:**

      1. **Use Existing Resume: Uses the resume uploaded during registration.**

      2. **Upload New Resume: Allows the jobseeker to upload a new resume.**

   o **Apply Button:**

      ▪ **If "Use Existing Resume" is selected, clicking this button applies for the job.**

      ▪ **If "Upload New Resume" is selected, an input field appears to upload the resume before applying.**

      ▪ **Success Message: Displays a success message and automatically closes the modal after applying.**

2. **Save Now:**

   o **Saves the job to the "Saved Jobs" section.**

   o **Button is disabled after the job is saved.**

API Endpoints:

- **Apply for Job: POST / jobseeker/applyjob/**

- **Save Job: POST / jobseeker/savejob/**
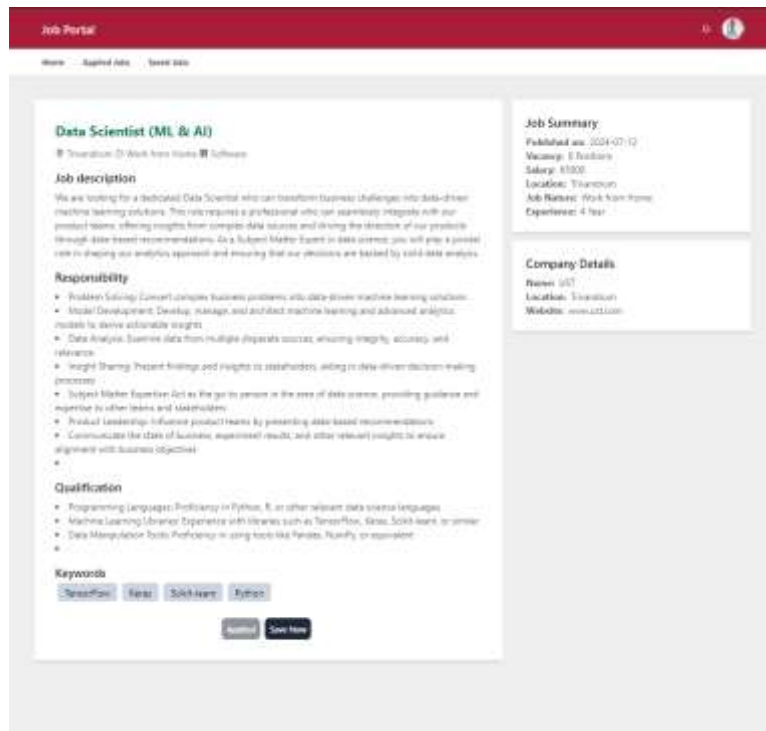
6.3. Navbar Tabs

1.Home

- **Functionality: The default page that loads upon login, showing the most recent jobs. Filters are applied via the "Find Jobs" button.**

2.Applied Jobs

- **Functionality: Displays all jobs the jobseeker has applied for in a table format with fields: title, company, applied on, view, withdraw, and action (pagination included).**

Buttons:

- **View: Opens the full job details and description.**
- **Withdraw:**

- o **Opens a confirmation modal to withdraw the application.**
- o **If confirmed, the application is withdrawn.**

- **Action:**

  - o **View Resume: Displays the resume used for the job application.**
  - o **Download Resume: Downloads the resume.**

API Endpoints:

- **Get Applied Jobs: GET / jobseeker/limitappliedjobview**
- **Withdraw Application: DELETE / jobseeker/applyjob/**
- **View Resume: ViewResume()**
- **Download Resume: DownloadResume()**

3.Saved Jobs
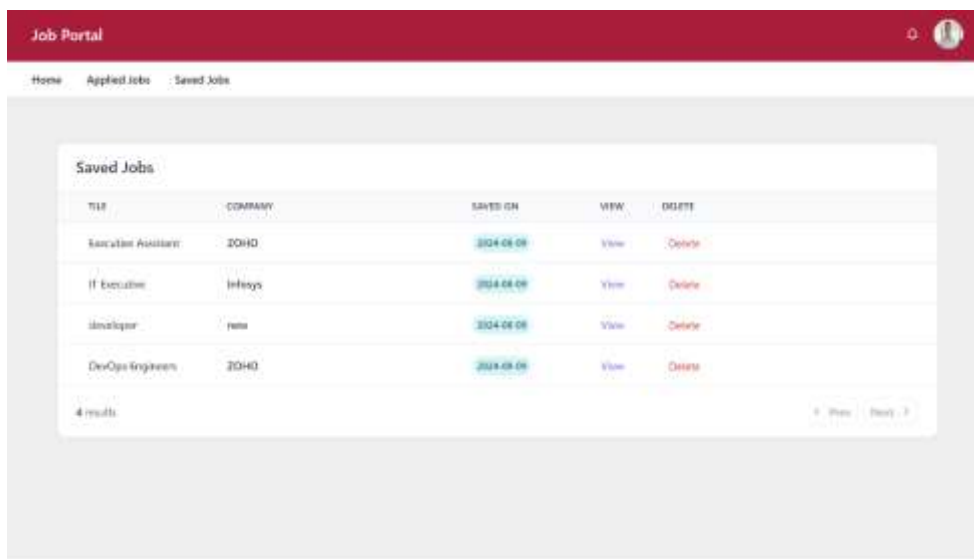
- **Functionality: Displays all jobs saved by the jobseeker in a table format with fields: title, company, saved on, view, delete, and pagination.**

Buttons:

- **View: Opens the full job details and description, where the jobseeker can apply for the job.**
- **Delete: Opens a confirmation modal to delete the saved job.**

API Endpoints:

- **Get Saved Jobs: GET / jobseeker/limitsavejobview/**
- **Delete Saved Job: DELETE / jobseeker/savejob/**

6.4. Profile Image Dropdown

6.4.1.Profile:

- **Update Profile**: Clicking on "Profile" opens the Jobseeker profile page. Here, the user can update its information, including:

  - **Name**

- Address1 & Address2

- Phone Number

- City

- State

- Pincode

- **Save Changes(button)**:

  - **Action**: After making any changes, the user clicks on the "Save Changes" button.

  - **API Endpoint**: PUT / jobseeker/jobseekerprofileview/

  - **Behavior**:

    - If no changes are detected, a warning message is displayed: "No changes detected."

    - If changes are made and saved successfully, a success message appears: "Profile updated successfully."
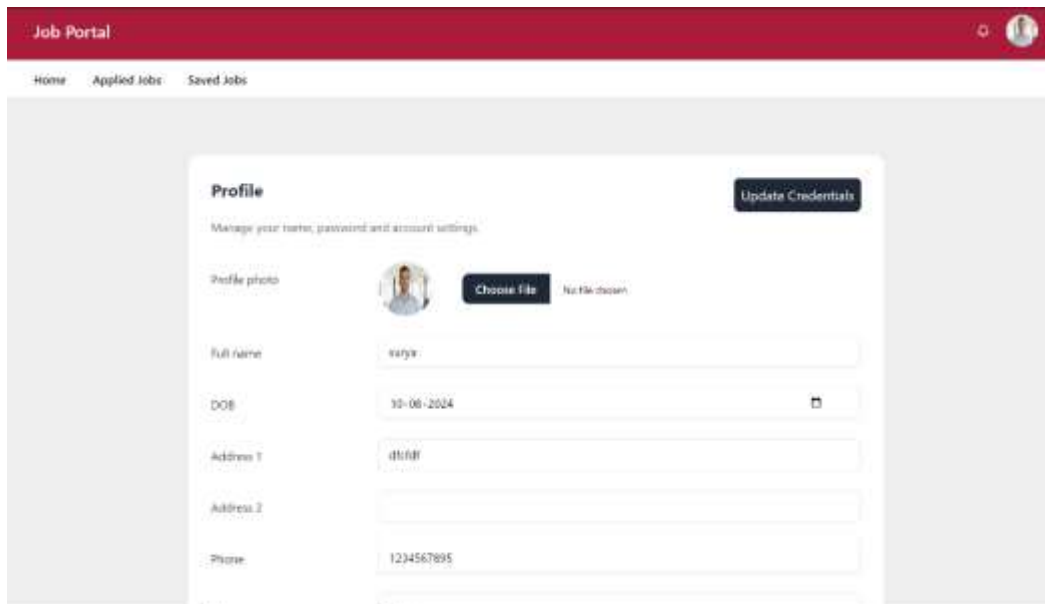
- **Update Credentials (includes Email, Username, Password)**:

  - **Action**: There is a button labelled "Update Credentials" on the profile page.

  - **Functionality**: Clicking on this button opens a modal where the Jobseeker can edit its email and username.

  - **Validation**: Both fields are required, and the system checks if the new email or username already exists in the database.

  - **API Endpoint**: PUT / company/users/

- **Change Password**:

  - **Action**: A "Change Password" button is also available on the update Credential modal.

  - **Functionality**: Clicking on this button opens another modal with fields for:

- **Old Password**

- **New Password**

- **Confirm Password**

- **Validation**: The "Save Changes" button is disabled if the new password and confirm password fields do not match.

- **API Endpoint**: PUT/ company/passwordchange/

- **Behavior**:

    - If the old password is incorrect, an error message appears: "Old password incorrect."

    - If correct, the new password is saved, and the modal closes automatically.

6.4.2.Education

- **Functionality: Displays and allows the management of the jobseeker's education details.**

Buttons:

- **Save Changes: Updates education details.**

- **Delete: Deletes an education entry after confirmation.**

- **Add Education: Opens a page to add new education details.**

API Endpoints:

- **Get Education Details: GET / jobseeker/jobseekereducationview/**

- **Update Education: PUT / jobseeker/jobseekereducationview/**

- **Delete Education: DELETE / jobseeker/jobseekereducationview/**

- **Add Education: POST / jobseeker/jobseekereducation/**

6.4.3.Experience

- **Functionality: Displays and allows the management of the jobseeker's work experience.**

Buttons:

- **Save Changes: Updates experience details.**

- **Delete: Deletes an experience entry after confirmation.**

- **Add Experience: Opens a page to add new experience details.**

API Endpoints:

- **Get Experience Details: GET / jobseeker/jobseekerexperienceview/**

- **Update Experience: PUT / jobseeker/jobseekerexperienceview/**

- **Delete Experience: DELETE / jobseeker/jobseekerexperienceview/**

- **Add Experience: POST / jobseeker/jobseekerexperience/**

6.4.4. Logout:

o **Action**: Clicking "Logout" will log the Jobseeker out of the system.

o **Behavior**:

- The system removes the authentication token.

- The user is redirected to the login page.

- **Navigate('/')**

Conclusion

This job portal provides a robust platform for jobseekers, employers, and administrators, enabling efficient and organized management of job applications, postings, and user profiles. By following the detailed documentation, developers can understand the intricacies of the system, ensuring the successful implementation and maintenance of the platform. The combination of clear workflows, well-defined user interactions, and precise API integrations makes this job portal a powerful tool in the recruitment process