

Project Name: Right Resource Fit

Description: Right Resource Fit is designed to streamline recruitment processes, enabling the Human Resource Management Team to handle functions related to hiring efficiently.

Purpose: This project will enable the Human Resource Management Team to handle HR functions related to recruitment.

Installation and Setup:

-Clone the repository:

```
git clone  
<https://github.com/springboardmentor106/RightResourceFit_Infosys_Internship_Oct2024_Team_04.git>
```

-Frontend Setup:

```
cd Frontend\RRF
```

```
npm install
```

```
npm start
```

```
npm run dev
```

-Backend Setup:

```
cd backend
```

```
npm install
```

```
npx nodemon
```

Folder Structure:

- Frontend RRF - contains frontend code for UI components and pages.
- backend - contains backend code for user authentication, job posting, email verification, etc.
 - config - holds configuration and connection files.
 - models - defines database structure.
 - routes - contains API endpoints.
 - controllers - logic for handling request

Key Features and Modules:

- User Management: Signup, login, password reset, email link verification.
- Job Posting: Create, update, and delete job posts.
- Application Management: Track and manage job applications.
- Tracking of job status: Follow-up system for HR and job seekers.
- Administration: Admins analyze users and jobs and manages the application

Project Components:

HomePage: The landing page where users are introduced to the application and its main features.

Login: Allows users to log in with their credentials. Handles input validation and authentication requests.

Register: User registration page where new users can create an account. Manages form inputs and user data submission.

User/Admin Dashboard: After logging in, users are directed here to view their matched projects and apply for open positions.

Profile: Shows user information and allows users to update their profiles.

ApplicationForm: Allows users to submit an application for a project, collecting relevant information and documents.

Notification: Email verification link sent to email for password reset in notification page.

API Documentation:

Describe API endpoint:

- Signup:
 - Endpoint: [/api/user/](#)
 - Method: handleSubmit
 - Description: Registers a new user as a job seeker or company.
- Login:
 - Endpoint: [/api/user/](#)
 - Method: handleSubmit
 - Description: Authenticates the user and redirects to the homepage.

- Reset Password:
 - Endpoint: `/api/user/reset-password/`
 - Method: `handleSubmit`
 - Description: Sends Email verification link for password reset verification.

Front-end Documentation

Key routes and components:

- Routes: LandingPage, About, Login, Register, ResetPassword, EmailVerification, User Dashboard, Admin Dashboard, Search for Companies and jobs
- UI Components: Detail how to use and customize the main UI components

Error Handling and Troubleshooting

Common errors and to troubleshoot them:

- Check if less secure app access is enabled for Gmail.
- Port Conflicts: Ensure no other services are running on port.

API information:

Frontend:

1. User Registration API

- Endpoint: `'/api/user/register'`
- Method: `'POST'`
- Request Body:

```
{  
  "name": "John Doe",  
  "email": "john.doe@gmail.com",  
  "password": "password123",  
  "Confirm password": "password123",  
  "phoneNumber": "1234567890"  
}
```

- Response:

```
{  
  "success": true,  
  "message": "Registered successfully!",  
}  
  
  
  

```

2. User Login API

- Endpoint: `/api/user/login`
- Method: `POST`
- Request Body:

```
{  
  "email": "john.doe@gmail.com",  
  "password": "password123"  
}
```

- Response:

```
{  
  "success": true,  
  "message": "Logged in successfully!",  
}
```

3. Password Reset Request API

- Endpoint: `/api/user/reset-password/`
- Method: `POST`
- Request Body:

```
{  
  "email": "john.doe@gmail.com"  
}"
```

- Response:

```
{  
  "success": true,  
  "message": "Password reset link sent to the registered email."  
}
```

4. Email Verification API

- Endpoint: `/api/user/reset-password/`
- Method: `GET`
- Request Body:

```
{  
  "email": "john.doe@gmail.com"  
}
```

- Response:

```
{  
  "success": true,  
  "message": "Password reset link sent!"  
}
```

5. Job Posting API (for Companies)

- Endpoint: `/api/jobs/postings`
- Method: `POST`
- Request Body:

```
{  
  "title": "frontend developer",  
  "description": "We are looking for a skilled frontend developer to join our team.",  
  "location": "Delhi",  
  "salary": "40000-60000",  
  "skills": ["Angular", "React", "Node.js"],
```

```
        "Employment type": "Full-Time",
        "Working schedule": "day shift",
        "Salary": "custom";
    }
}
```

- Response:

```
{
  "success": true,
  "message": "Job posted successfully",
  "
}
```

6. View Job Listings API (for Job Seekers)

- Endpoint: `/api/application?applicantID`
- Method: `GET`

- Response:

```
{
  "success": true,
  "jobs": [
    {
      "title": "frontend developer",
      "location": "Mumbai",
      "salary": "60000-80000",
      "skills": ["JavaScript", "Angular", "Node.js"]
      "Employment type": "Full-time",
      "Working schedule": "Day shift",
      "Salary": "2871 Custom(Monthly)"
    },
    {
      "jobId": "job680",
      "title": "Product Developer",
      "company": "Infosys",
      "location": "Hyderabad",
      "salary": "80000-120000",
      "requirements": ["Product Management", "Agile", "Leadership"]
    }
  ]
}
```

```
    "Employment type": "Full-time",
    "Working schedule": "Day shift",
    "Salary": "2871 Custom(Weekly)"
```

```
}
```

7. Apply for a Job API (for Job Seekers)

- Endpoint: `/api/application?applicantID=...`
- Method: `POST`

- Request Body:

```
{
  "Cover letter": "brief cover letter"
  "resume": "resume.pdf(download)"
}
```

- Response:

```
{
  "success": true,
  "message": "Application submitted successfully."
}
```

8. View Application Status API (for Job Seekers)

- Endpoint: `/api/application?applicantID=...`
- Method: `GET`

- Response:

```
{
  "success": true,
  "applications": [
    {
      "Position": "web developer",
      "Date applied": "November 11,2024",
    }
  ]
}
```

```
        "status": "Reviewed"
    },
    {
        "Position": "web developer",
        "Date applied": "November 11,2024",
        "status": "Applied"
    }
]
```

9. Job Seeker Profile Update API

- Endpoint: `/api/user/profile`
- Method: `PUT`

- Request Body:

```
{
    "name": "John Doe",
    "email": "john.doe@gmail.com",
    "Phone number": "1234567890",
}
```

- Response:

```
{
    "success": true,
    "message": "Profile updated successfully."
}
```

10. Company Dashboard Status API

- Endpoint: `/api/application`
- Method: `GET`

- Response:

```
{
    "success": true,
    "stats": {
        "totalJobPostings": 8,
```

```
        "applicationsReceived": 20,  
        "applicationsPendingReview": 14,  
        "Shortlisted applicants": 5  
    }  
}
```

Backend:

1. User Api:

Route Api : `.../controllers/userController`

Controller: The **user controller** serves as the backbone of the user management system in the application. It facilitates actions like:

- Registering new users.
- Logging users in and authenticating them with JWT.
- Allowing users to update their profiles and reset passwords.
- Providing admin functionality to manage users (e.g., viewing, updating, deleting, restoring users).

Each function is tied to an HTTP endpoint and performs a specific user-related task, ensuring a structured flow of operations related to user management.

Features:

- **JWT Authentication:** Many functions (like `getUserProfile`, `updateUserProfile`, etc.) rely on JWT tokens for authentication. The `req.user.id` comes from the token and ensures that the user is authenticated.
- **Nodemailer:** Used for sending emails, specifically for password reset functionality. SMTP connection details (like host, port, and credentials) are stored in environment variables.

2. Upload Api:

Route Api: `.../controllers/uploadController`

Controller: The **uploadController** is designed to handle file uploads, specifically images and PDFs, and to upload them to Cloudinary. It:

- Validates the file.
- Determines the correct resource type (image or raw for PDFs).

- Uploads the file to Cloudinary.
- Returns the URL of the uploaded file to the client.

This controller allows users to upload files to the cloud with the added benefit of handling transformations and file type validation.

3.Report Api:

Route Api: `../controllers/reportController`

Controller: The `reportController` is responsible for managing user reports and dashboard data. It allows users to:

- Generate new reports and associate them with their account.
- Retrieve dashboard data that might include job postings, applications, and other related information.
- Retrieve a list of all reports associated with a specific user.

This controller provides a way for users to generate and manage their reports, as well as view relevant dashboard data through various API endpoints.

4.Job Api:

Route Api: `../controllers/jobController`

Controller: The `jobController` is responsible for managing job postings in the system, specifically for HR (Admin) users. The controller handles the following functionalities:

1. Creating job postings.
2. Updating job postings.
3. Deleting job postings.
4. Retrieving job listings for applicants to search through based on various filters.

5.Interaction Api:

Route Api: `../controllers/interactionController`

Controller: The `interactionController` handles the management of interactions (such as notifications and application status updates) related to job applications. This includes:

1. Tracking application statuses for a specific applicant.

2. Updating the application status by HR (Admin).
3. Sending notifications related to job applications.
4. Deleting notifications for a specific user or specific notification.

6.Feedback Api:

Route Api: `../controllers/feedbackControllers`

Controller: The `FeedbackController` manages feedback-related operations:

1. Submitting feedback
2. Retrieving all feedback

SMTP Configuration: The controller uses `Nodemailer` for sending emails. It connects to an SMTP server (configured via environment variables) to send the feedback acknowledgment emails. The SMTP connection is verified when the application starts, ensuring that email sending is properly configured.

This controller enables feedback collection and notification via email, helping to keep both users and administrators informed about the feedback submissions.

7.Application Api:

Route Api: `../controllers/applicationController`

Controller: The `applicationController` manages the job application lifecycle.

- Apply for a Job
- Get Application by ID
- Get All Applications
- Update Application Status

8.Error Handling Api:

Route Api: `../constants`

9.AuthMiddleWare Api:

Route Api: `../models/userModel`

Frontend pages

About page:

The "About" page of "Right Resource Fit" (RRF) introduces the platform as a creative and aesthetic-focused job search and recruitment site. The page features HR-themed images, blending visual appeal with elements like management and skill development.

Feedback page:

This feedback page features a clean, user-friendly design with fields for name, email, phone, and feedback. The page provides a simple and elegant interface for users to communicate effectively. On sending feedback in contact page a mail is sent to user id.

Admin:

- Post jobs
- Manage jobs
- Manage users
- Manage user application
- Update their profile

User:

- Apply for jobs
- View their application status of their applied jobs
- Search for jobs
- Notified on their applications
- Update their profile

Screenshots:

Admin Interface

Admin Dashboard



akash

- [Home](#)
- [Post Jobs](#)
- [Manage Jobs](#)
- [User Manage](#)
- [All Applications](#)
- [Manage Profile](#)
- [Logout](#)

Total Jobs
8

Total Applicants
20

Shortlisted Applicants
5

In Review
14

Application Response



Shortlisted: 5
In Review: 14
Rejected: 1

Job Title	Location	Skills	Employment Type
frontend developer	India	Responsive web design, javaScript, Html	Full-time, Part-time
sdjkf	dfkj	askj, sdj	Full-time, Part-time
askjfd	sdf	dsfj	Full-time, Part-time, On demand
java developer	dkjf, dalk	java, react, html	Full-time
forntend developer	haryana	java, angular, .net	Full-time, Part-time, On demand, Negotiable

New job posting



akash

- [Home](#)
- [Post Jobs](#)
- [Manage Jobs](#)
- [User Manage](#)
- [All Applications](#)
- [Manage Profile](#)
- [Logout](#)

Create New Job Posting

Job Title

Job Description

Location

Skills

Employment Type

 Full-time

Manage job postings by updating and deleting jobs



akash

- Home
- Post Jobs
- Manage Jobs
- User Manage
- All Applications
- Manage Profile
- Logout

Manage Job Postings

Create New Job

Search jobs by title, skills or location

frontend developer Description: hey Location: India Skills: Responsive web design, javaScript, Html Employment Type: Full-time, Part-time Working Schedule: Day shift Salary: 2871 Custom (Monthly) Hiring Multiple Candidates: Yes Update Job Delete	sdjkf Description: s;dkf;lsdk Location: dfkj Skills: askj, sdj Employment Type: Full-time, Part-time Working Schedule: Night shift Salary: 4345354 Hourly (Yearly) Hiring Multiple Candidates: Yes Update Job Delete
askjfd	java developer

User management in providing access



akash

- Home
- Post Jobs
- Manage Jobs
- User Manage
- All Applications
- Manage Profile
- Logout

User Management

User Id	Avatar	Name	Email	Phone Number	Admin Access	Status	Actions
670e732e568768a319b68f1b		John Doe	john.doe@example.com	1234567890	No	Active	Grant Access Delete
670fc57a99828a990665017d		Pranav Gupta	pranavgupta.kuna08@gmail.com	5555555555	Yes	Active	Remove Access Delete
6710e0602455591ad4109ca3		John Doe	johndoe@example.com	1234567890	No	Active	Grant Access Delete
6710eab92455591ad4109ca6		Shivam Singh	shivam@google.com	7906215489	No	Active	Grant Access Delete
6710efe72455591ad4109cb5		aa	aa@a.in	1234567890	No	Active	Grant Access Delete

Application and status of applicants



akash

-  Home
-  Post Jobs
-  Manage Jobs
-  User Manage
-  All Applications
-  Manage Profile
-  Logout

All Applications

Search by Job Role

Applicant	Applicant Email	Applied Job	Applied Date	Status	View Applications
 Pranav Gupta	pranavgupta.kuna08@gmail.com	frontend developer	8 November 2024	Hired	<button>See Application</button>
 Pranav Gupta	pranavgupta.kuna08@gmail.com	sdjkf	8 November 2024	Accepted	<button>See Application</button>
 Pranav Gupta	pranavgupta.kuna08@gmail.com	askjfd	8 November 2024	Reviewed	<button>See Application</button>
 Pranav Gupta	pranavgupta.kuna08@gmail.com	java developer	8 November 2024	Reviewed	<button>See Application</button>
 Pranav Gupta	pranavgupta.kuna08@gmail.com	java developer	8 November 2024	Hired	<button>See Application</button>
 Pranav Gupta	pranavgupta.kuna08@gmail.com	forntend developer	8 November 2024	Hired	<button>See Application</button>

Admin profile updation



akash

-  Home
-  Post Jobs
-  Manage Jobs
-  User Manage
-  All Applications
-  Manage Profile
-  Logout

User Profile



Name:
akash ↗

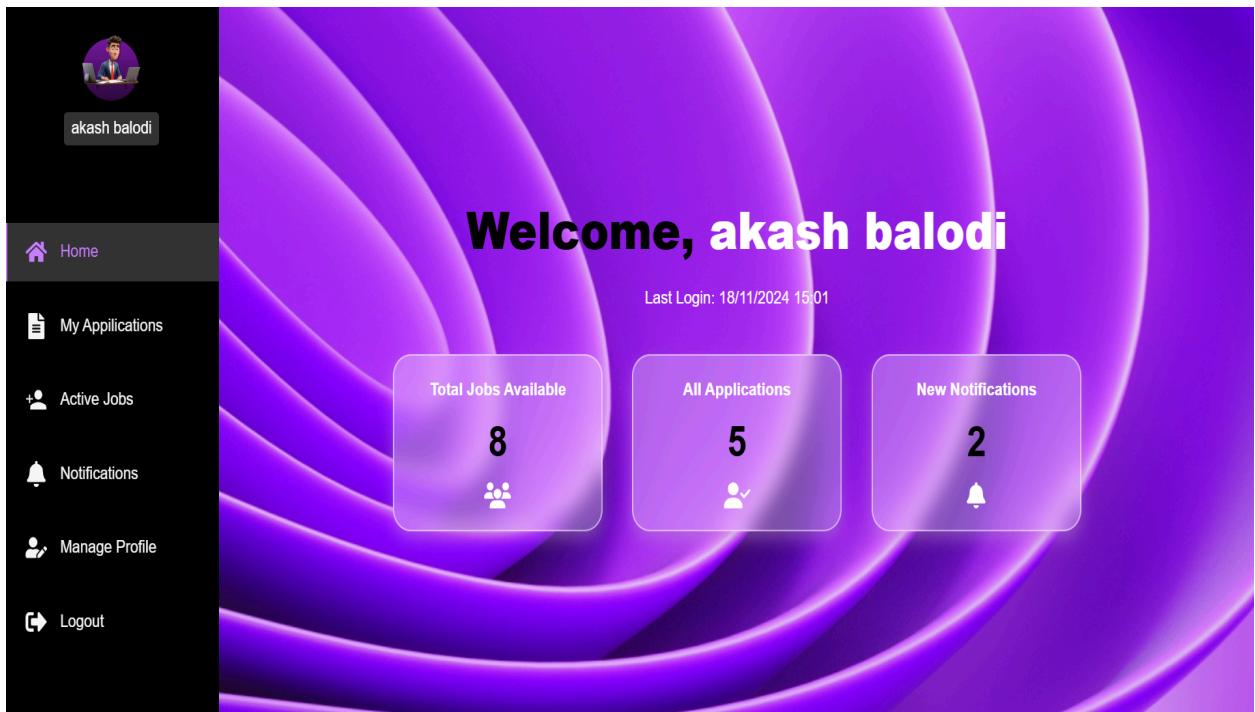
Role
HR Manager

Email:
akashbalodi17@gmail.com

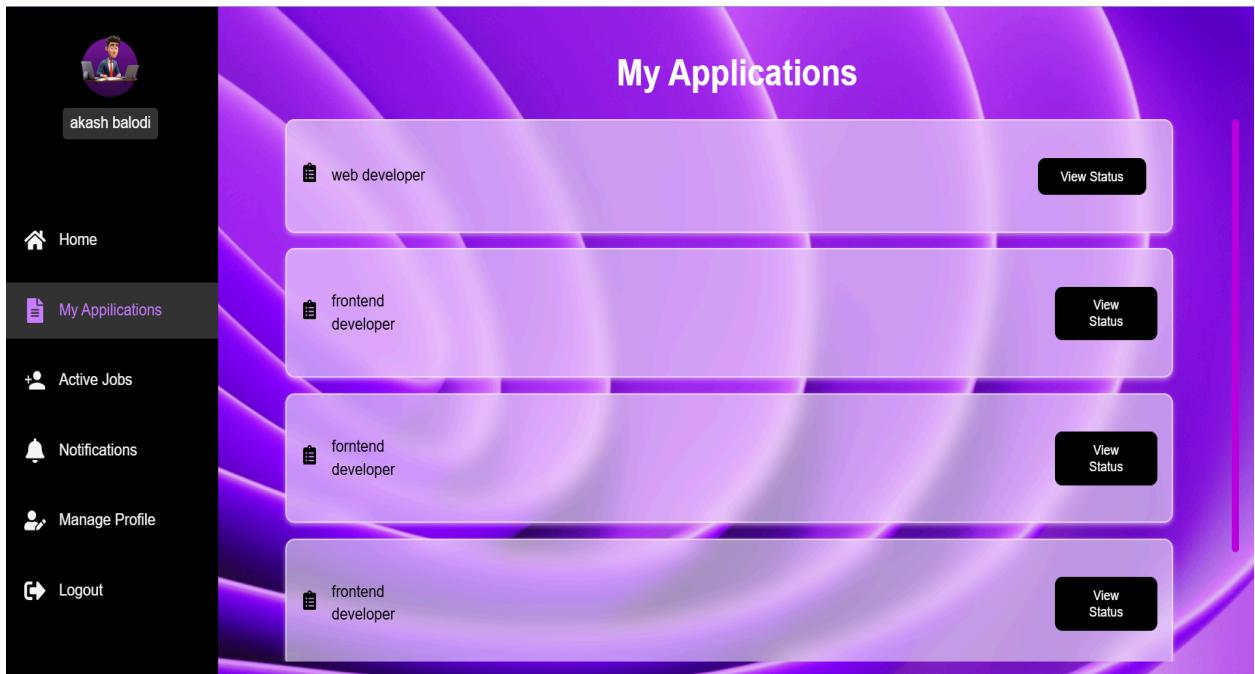
Phone Number:
9058719533 ↗

User interface

User dashboard



User job applications



Job search interface

The screenshot shows a job search interface with a dark sidebar on the left and a purple-themed main content area.

Left Sidebar:

- User profile icon: A person sitting at a desk with a laptop.
- User name: akash balodi
- Navigation links:
 - Home
 - My Applications
 - Active Jobs (highlighted)
 - Notifications
 - Manage Profile
 - Logout

Main Content Area:

Active Jobs

Search jobs by title, skills or location

Job Listing 1: frontend developer

Description: hey
Location: India
Skills: Responsive web design, javaScript, Html
Employment Type: Full-time, Part-time
Working Schedule: Day shift
Salary: 2871 Custom (Monthly)
Hiring Multiple Candidates: Yes

Successfully Applied!

Job Listing 2: sdjkf

Description: s;dkf;lsdk
Location: dfkj
Skills: askj, sdj
Employment Type: Full-time, Part-time
Working Schedule: Night shift
Salary: 4345354 Hourly (Yearly)
Hiring Multiple Candidates: Yes

Successfully Applied!

Bottom Labels:

askjfd java developer

User notifications on jobs applied

The screenshot shows a user notifications page with a dark sidebar on the left and a purple-themed main content area.

Left Sidebar:

- User profile icon: A person sitting at a desk with a laptop.
- User name: akash balodi
- Navigation links:
 - Home
 - My Applications
 - Active Jobs
 - Notifications (highlighted)
 - Manage Profile
 - Logout

Main Content Area:

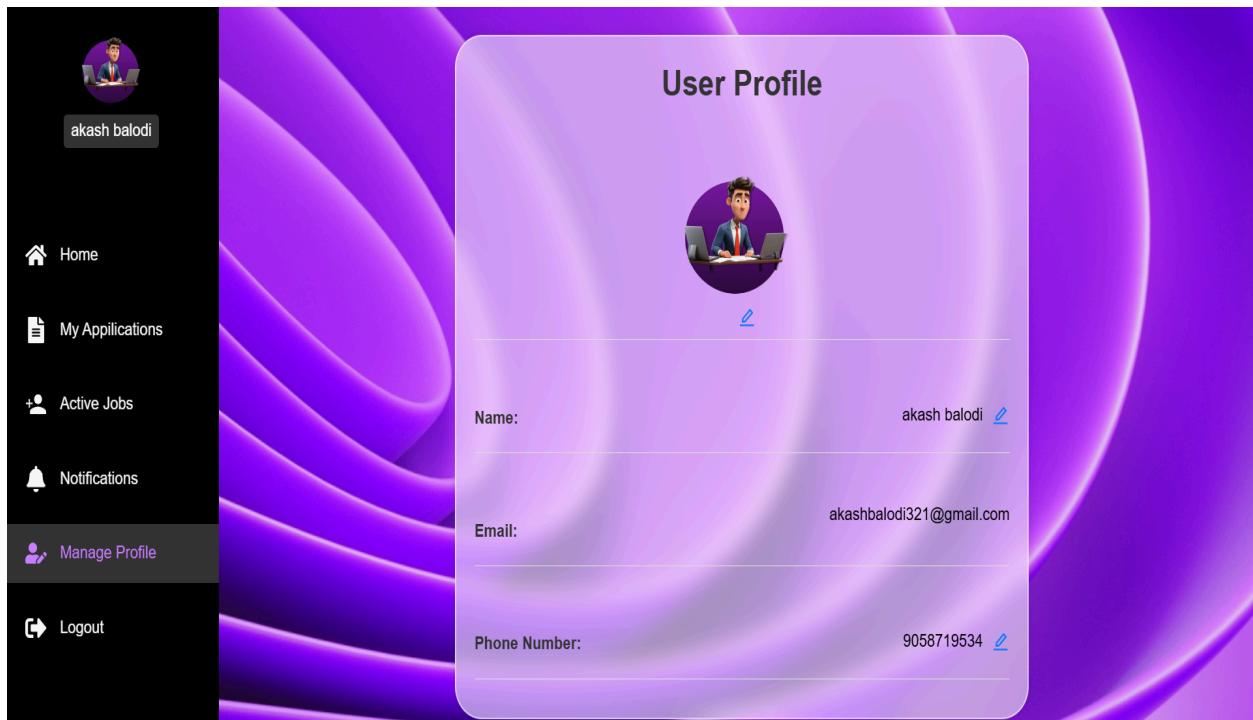
Your Notifications

Clear All

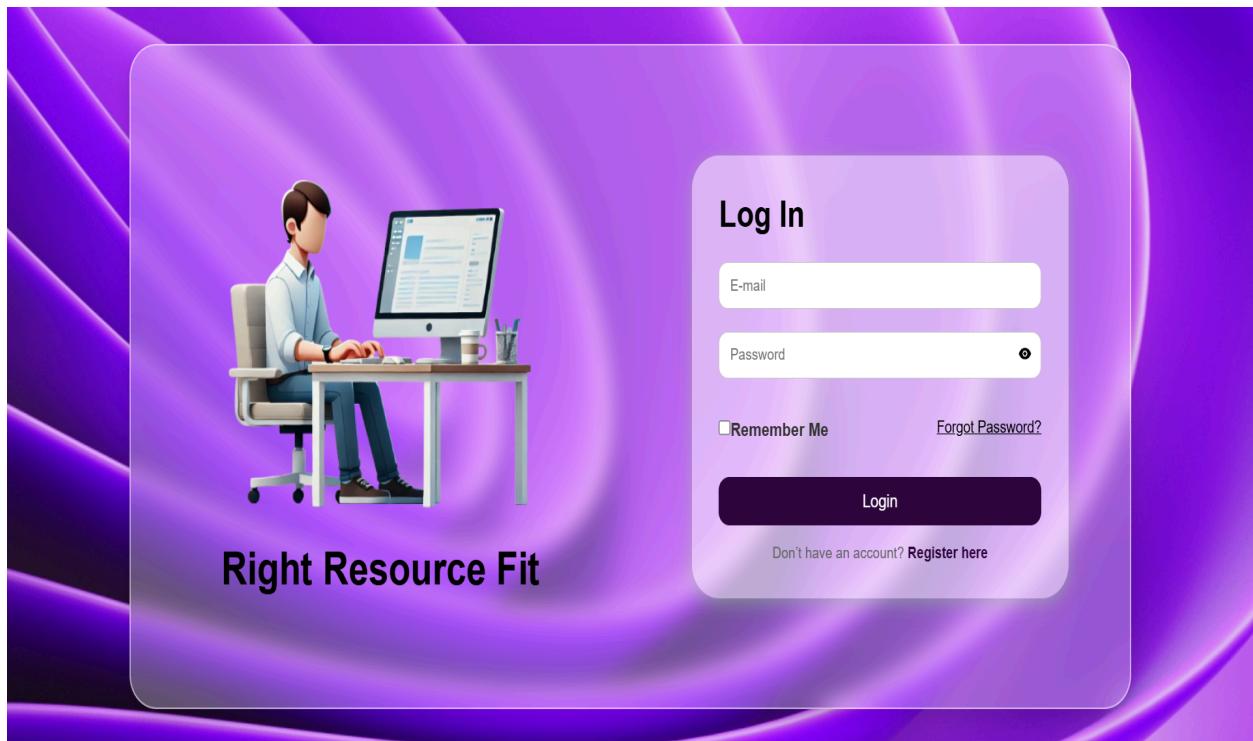
Notification 1: Your application status has been Reviewed
11/11/2024

Notification 2: Your application status has been Accepted
11/12/2024

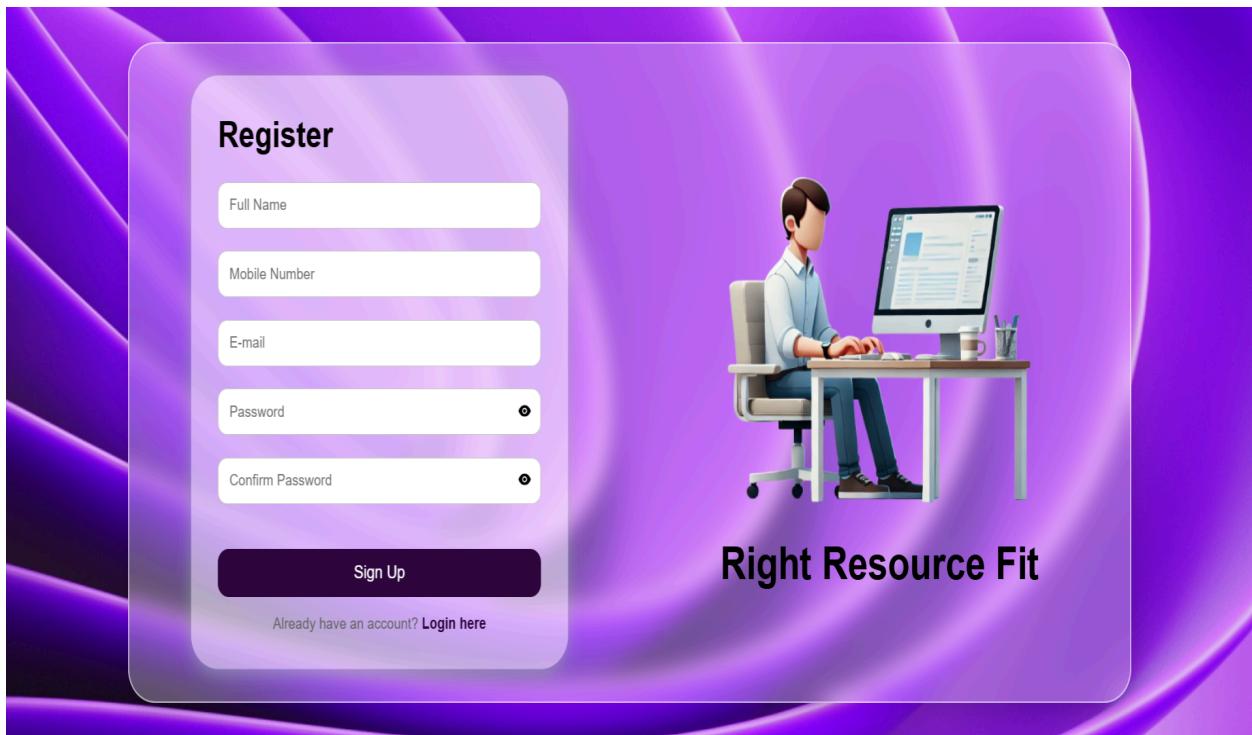
User profile updation:



Login page:



Register page:



Home page:

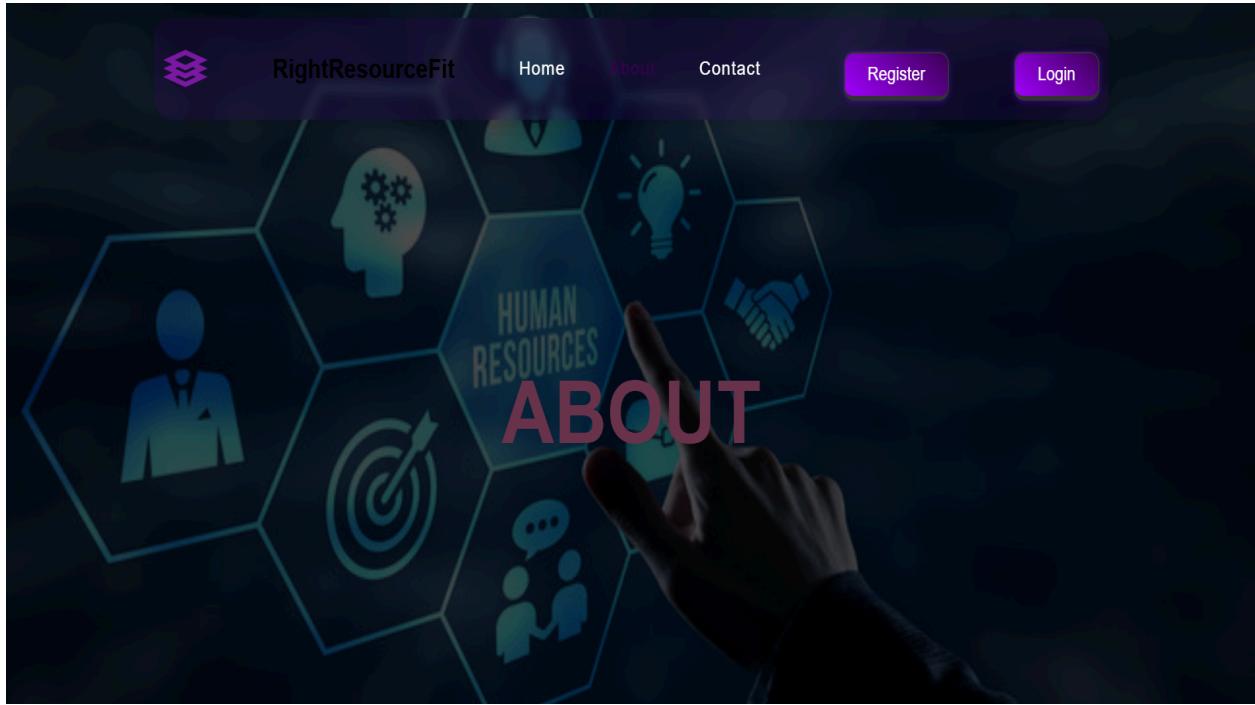
Explore Exciting career opportunities

Explore & find new career opportunities with Right Resource Fit

Right Resource Fit keeps you updated with the latest job opportunities by following industry trends and top professionals on social media.

[Get Started](#) [Watch Demo](#)

About page:



Contact/Feedback page:

A screenshot of the RightResourceFit website's Contact Us form. The form is contained within a light purple rectangular box with rounded corners. At the top center, it says "Contact Us". Below that are four input fields: "Name" (with placeholder "Enter your name"), "Email" (with placeholder "Enter your email"), "Phone" (with placeholder "Enter your phone number"), and "Feedback" (with placeholder "Enter your feedback"). At the bottom of the form is a large, dark purple "Submit" button.