

# **PROJECT REPORT**

## **JOB PORTAL**



**Spring 2025**

Submitted by: **Mohsin Sajjad, Muhammad Hassan, Muhammad Afnan Khan**

Registration No: **22pwsce2149,22pwcse2105,22pwcse2155**

Class Section: **A**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Submitted to:

**Engr. Summeya Salahudin**

Month Day, Year (04 07 2025)

Department of Computer Systems Engineering  
University of Engineering and Technology, Peshawar

# Project Title

## JOB SPHERE – Laravel 12 Based Job Portal

### Objective:

To develop a fully functional and user-friendly job portal using Laravel 12 that connects job seekers with employers, allowing job applications, job listings, resume uploads, and admin-level employer verification.

### Tools & Technologies:

- Framework: Laravel 12 (PHP)
- Frontend: Blade Templating, Bootstrap 5, HTML/CSS
- Database: MariaDB/MySQL
- Server: Localhost (XAMPP)
- Version Control: GitHub
- Deployment Platform: Railway (Free Hosting)
- Free hosting (aiven.io)

### 1. Entity Descriptions:

Below is a summary of the core entities, their primary keys, and main attributes:

Entity	Primary Key	Key Attributes	Description
User	UserID	Name, Email, Password, UserType	Stores common account data for both JobSeekers and Employers.
JobSeeker	SeekerID	UserID (FK), DateOfBirth, Address, ProfileSummary	Subtype of User; holds seeker-specific profile details.
Employer	EmployerID	UserID (FK), CompanyName, CompanyDescription, CompanyWebsite, VerifiedStatus	Subtype of User; holds employer-specific company details.
Resume	ResumeID	SeekerID (FK), FilePath, UploadDate, LastUpdated	Stores uploaded resumes for JobSeekers.

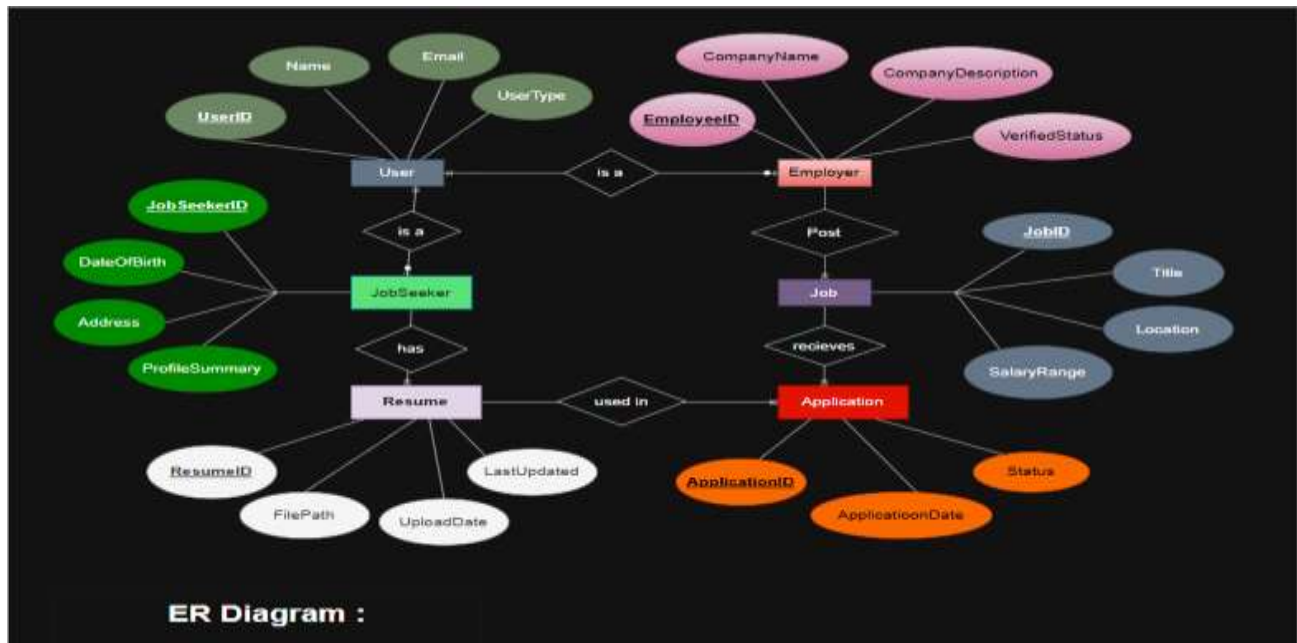
<b>Job</b>	JobID	EmployerID (FK), Title, Description, Requirements, Location, SalaryRange, PostedDate	Represents job listings posted by Employers.
<b>Application</b>	ApplicationID	JobID (FK), SeekerID (FK), ResumeID_(FK), AppliedDate, Status	Captures a JobSeeker's application to a specific Job with one resume.

## 2. Detailed Business Rules:

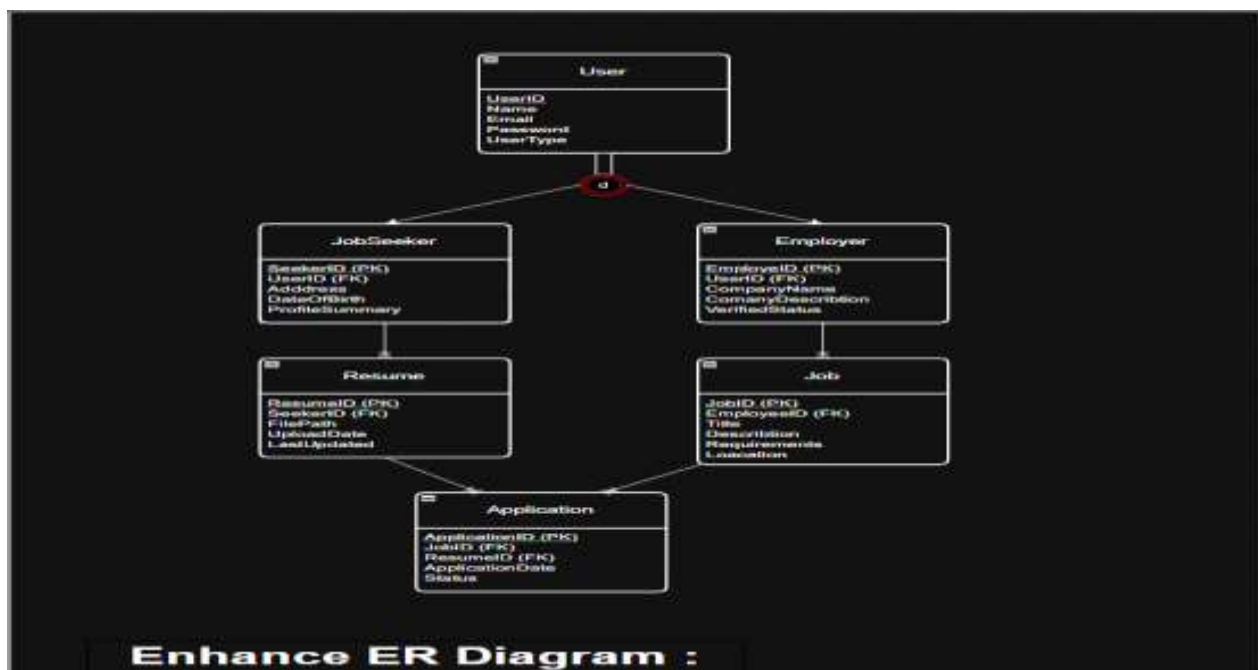
- a. **User registration:** Every account must register as either a JobSeeker or an Employer; the UserType attribute enforces this.
- b. **Unique email:** The Email in **User** must be unique and validated before account activation.
- c. **Email verification:** Only Employers with *VerifiedStatus = TRUE* may post Job listings.
- d. **Job postings:** An Employer may post zero or more Jobs; each Job must reference exactly one Employer.
- e. **Resume upload:** A JobSeeker may upload multiple Resumes, but only one active Resume may be associated per Application.
- f. **Profile completeness:** JobSeekers must complete mandatory fields (*DateOfBirth, Address, Profile Summary*) before applying to any Job.
- g. **Application constraint:** A JobSeeker may apply to the same Job at most once; duplicate applications are disallowed.
- h. **Application status:** Each Application's Status must be one of: Pending, Reviewed, Interview, Offered, Rejected.
- i. **Posting requirements:** Each Job must include at minimum: Title, Description, Requirements, Location, and Posted Date.
- j. **Automatic timestamps:** Applied Date (*Application*) and Posted Date (*Job*) default to the current system date when the record is created.
- k. **Resume updates:** When a JobSeeker updates a Resume, the old version may be retained for audit but only the latest may be linked to new Applications.
- l. **Deletion rules:** Deleting a User cascades to delete associated JobSeeker/Employer sub-records; deleting an Employer prevents deletion if open Job listings exist.

---

## 3. Entity–Relationship Diagram (ERD):



#### 4. Enhanced ER Diagram (EERD):



## Relational Schema:

### USER

<u>User_ID</u>	Name	Email	Password	UserType
----------------	------	-------	----------	----------

### Employer

<u>Employee_ID</u>	<u>User_ID</u>	Company_Name	Company_Discription	Company_website	Verified_status
--------------------	----------------	--------------	---------------------	-----------------	-----------------

### Jobseekers

<u>Seeker_ID</u>	<u>User_ID</u>	Date_Of_birth	Address	Profile_Summary
------------------	----------------	---------------	---------	-----------------

### Job

<u>Job_ID</u>	<u>Employee_ID</u>	Title	Description	Requirement	Location	Salary_range	Posted_Date
---------------	--------------------	-------	-------------	-------------	----------	--------------	-------------

### Resume

<u>Resume_ID</u>	<u>Seeker_ID</u>	File_Path	Upload_Date	Last_Updated
------------------	------------------	-----------	-------------	--------------

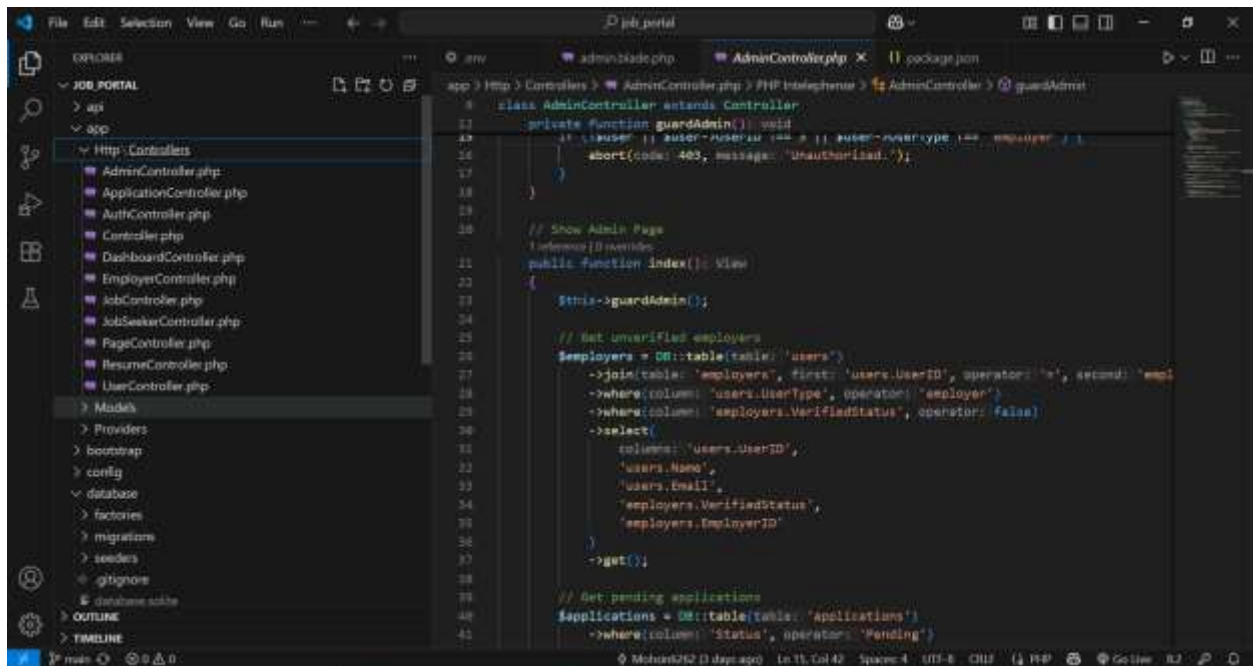
### Application

<u>Application_ID</u>	<u>Job_ID</u>	<u>Seeker_ID</u>	<u>Resume_ID</u>	Applied_Date	Status
-----------------------	---------------	------------------	------------------	--------------	--------

## Implementation on laravel:

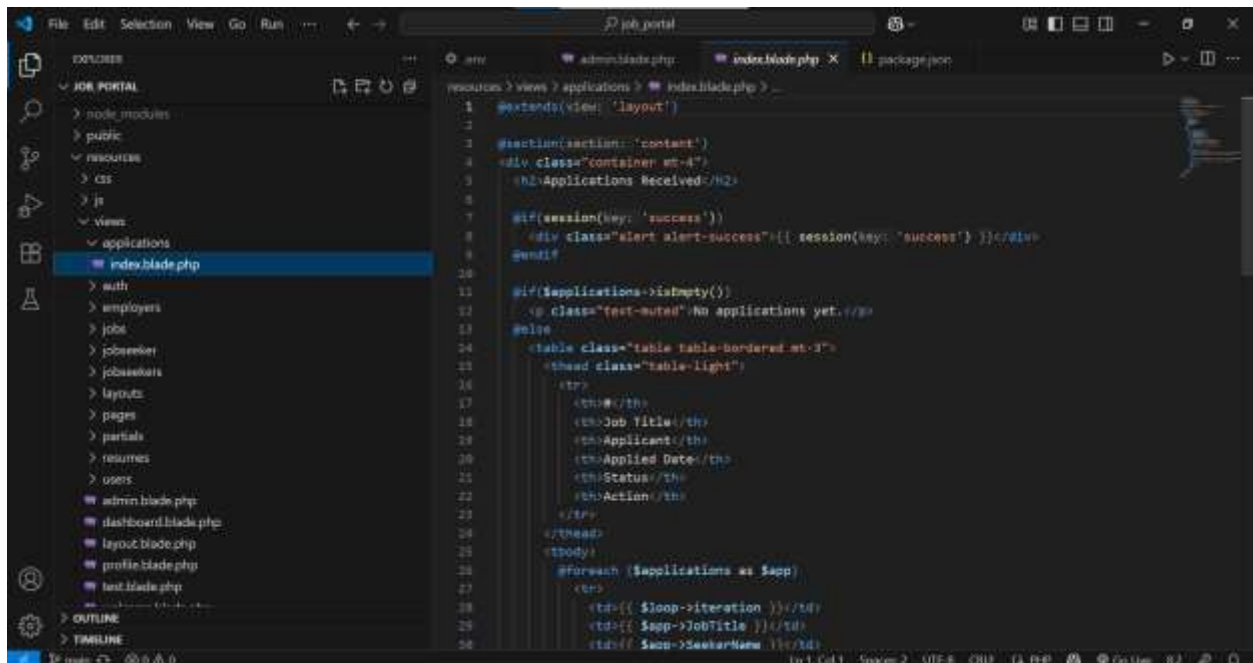
### Controller:

Backend of our project



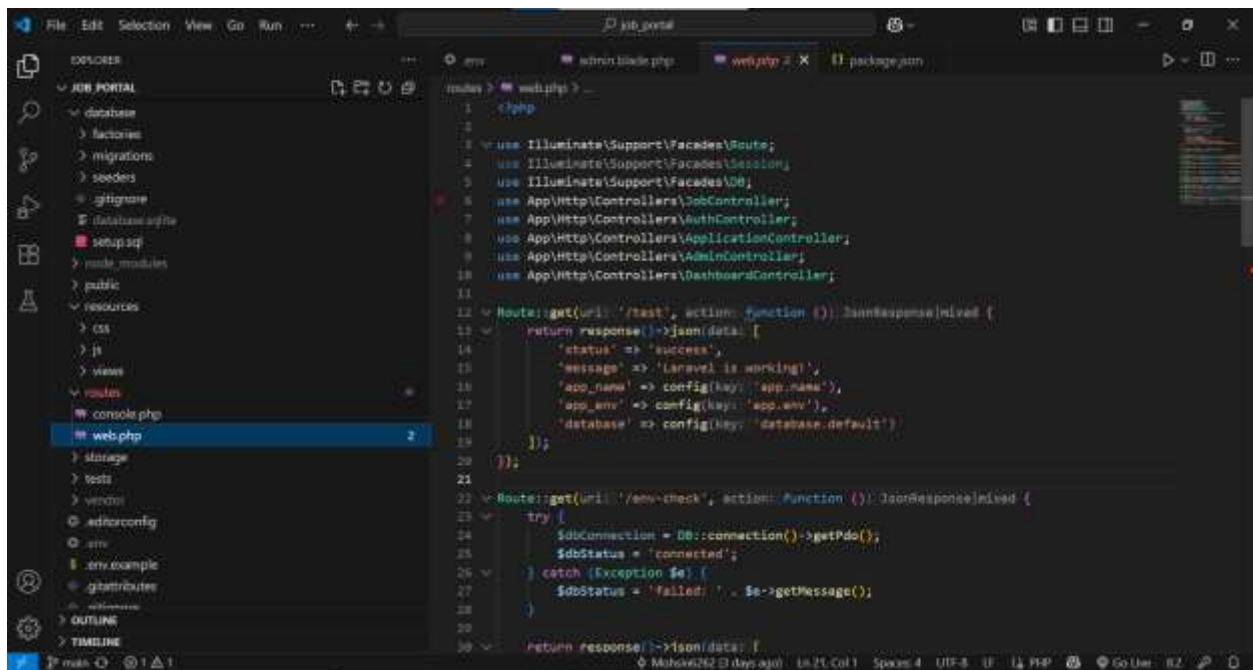
## View:

Frontend HTML



## Routes:

Maps URLs to controller functions



The screenshot shows a code editor with a sidebar on the left displaying the project structure for 'JOB PORTAL'. The main editor area shows the 'web.php' file with the following code:

```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use Illuminate\Support\Facades\Session;
5 use Illuminate\Support\Facades\DB;
6 use App\Http\Controllers\JobController;
7 use App\Http\Controllers\AuthController;
8 use App\Http\Controllers\ApplicationController;
9 use App\Http\Controllers\AdminController;
10 use App\Http\Controllers\DashboardController;
11
12 Route::get(url: '/test', action: function () { $jsonResponse = json([
13     'status' => 'success',
14     'message' => 'Laravel is working!',
15     'app_name' => config(key: 'app.name'),
16     'app_env' => config(key: 'app.env'),
17     'database' => config(key: 'database.default')
18 ]]);
19 });
20
21
22 Route::get(url: '/env-check', action: function () { $jsonResponse = json([
23     try {
24         $dbConnection = DB::connection()->getPdo();
25         $dbStatus = 'connected';
26     } catch (Exception $e) {
27         $dbStatus = 'failed: ' . $e->getMessage();
28     }
29
30     'Return response' => json([data: [
```

Deployment:

## Step 1: Push the project on GitHub:

```
cd C:\xampp\htdocs\job_portal
```

```
git init
```

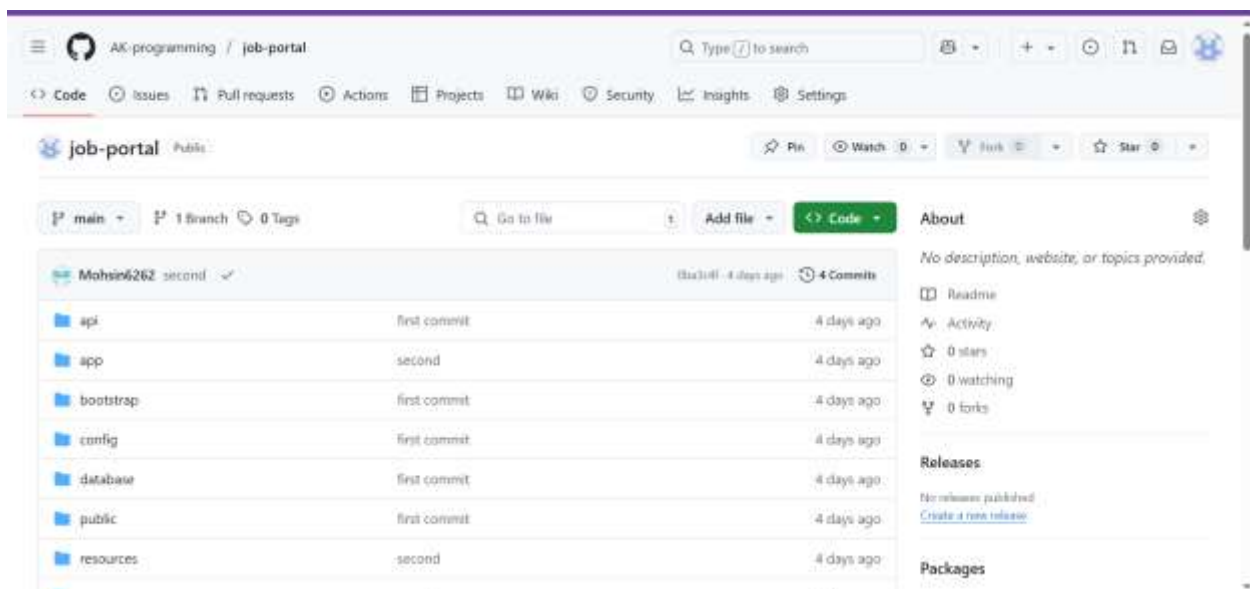
```
git remote add origin https://github.com/Ak-programming/job_portal.git
```

```
git add .
```

```
git commit -m "Pushing project"
```

```
git branch -M main
```

```
git push -u origin main
```

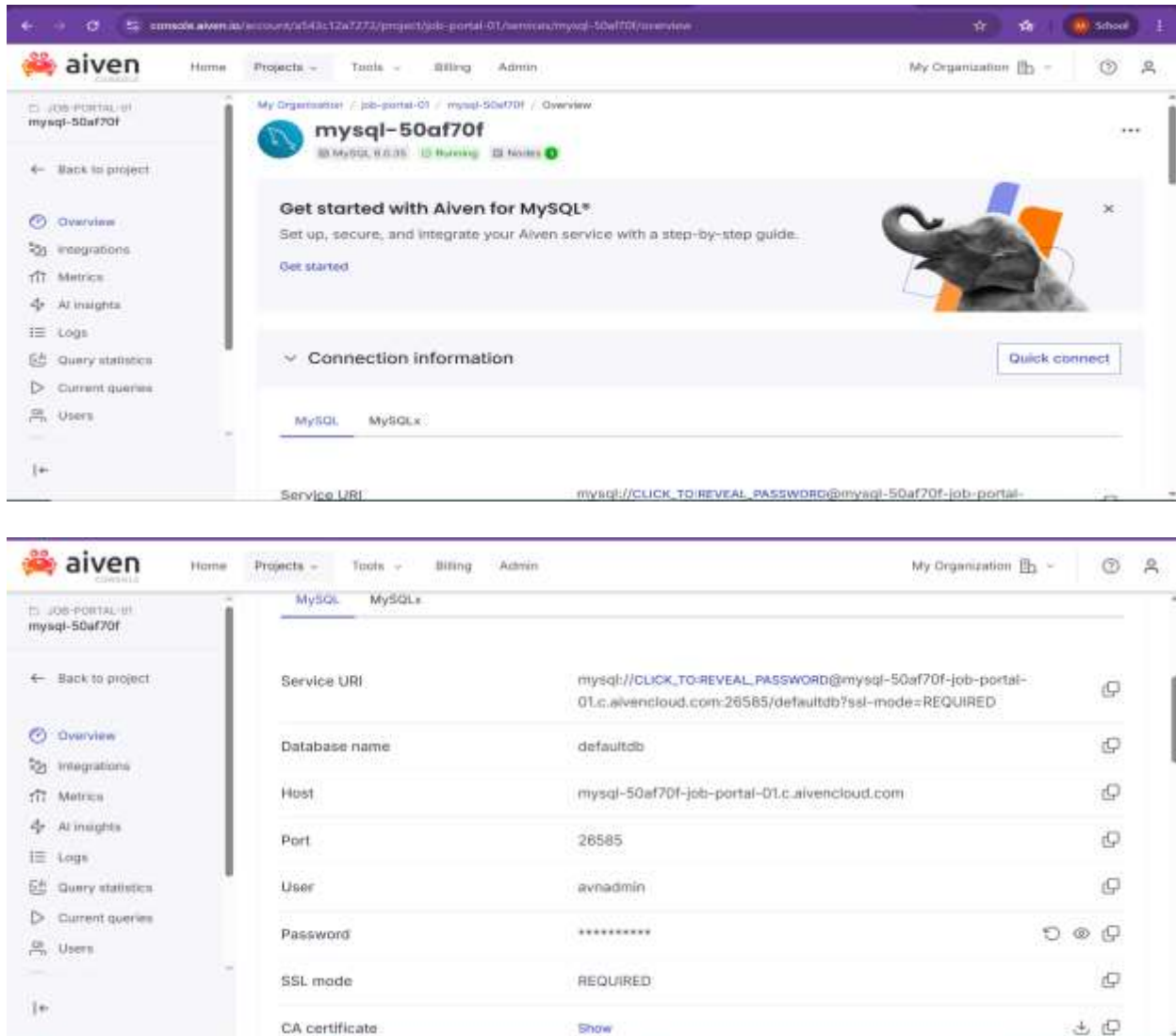


## Step 2: Create MySQL Database on Aiven.io

- Sign in to Aiven.io and create a new MySQL service.
- After creation, note the following credentials:
  - DB\_HOST
  - DB\_PORT
  - DB\_DATABASE

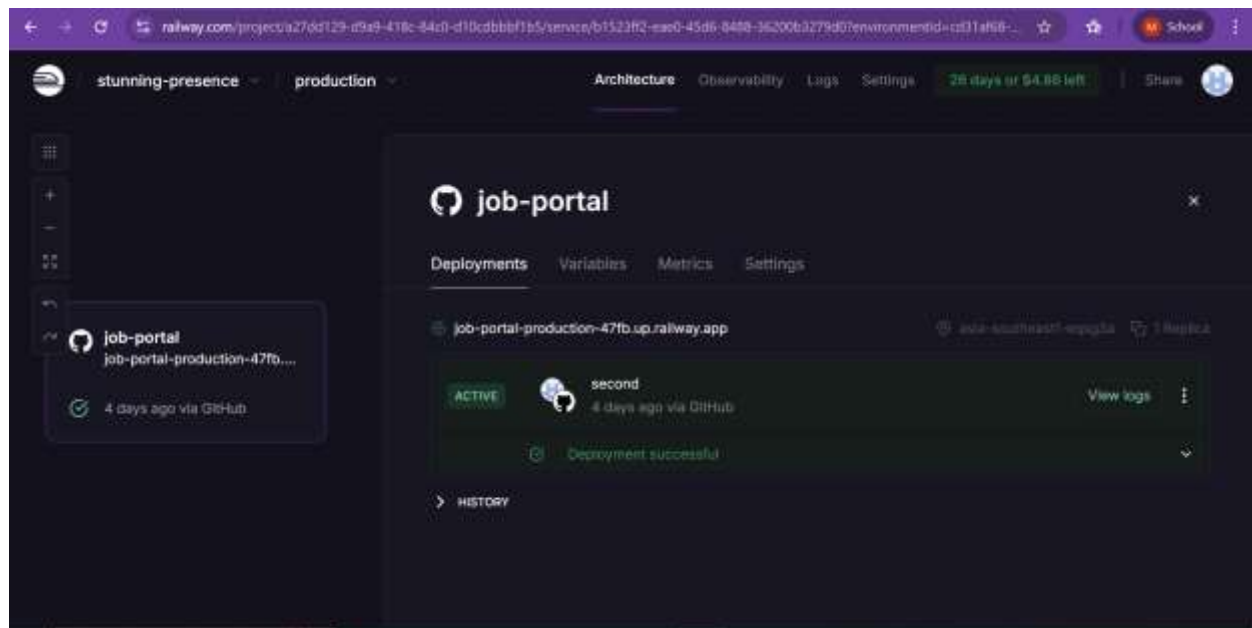


- DB\_USERNAME
- DB\_PASSWORD



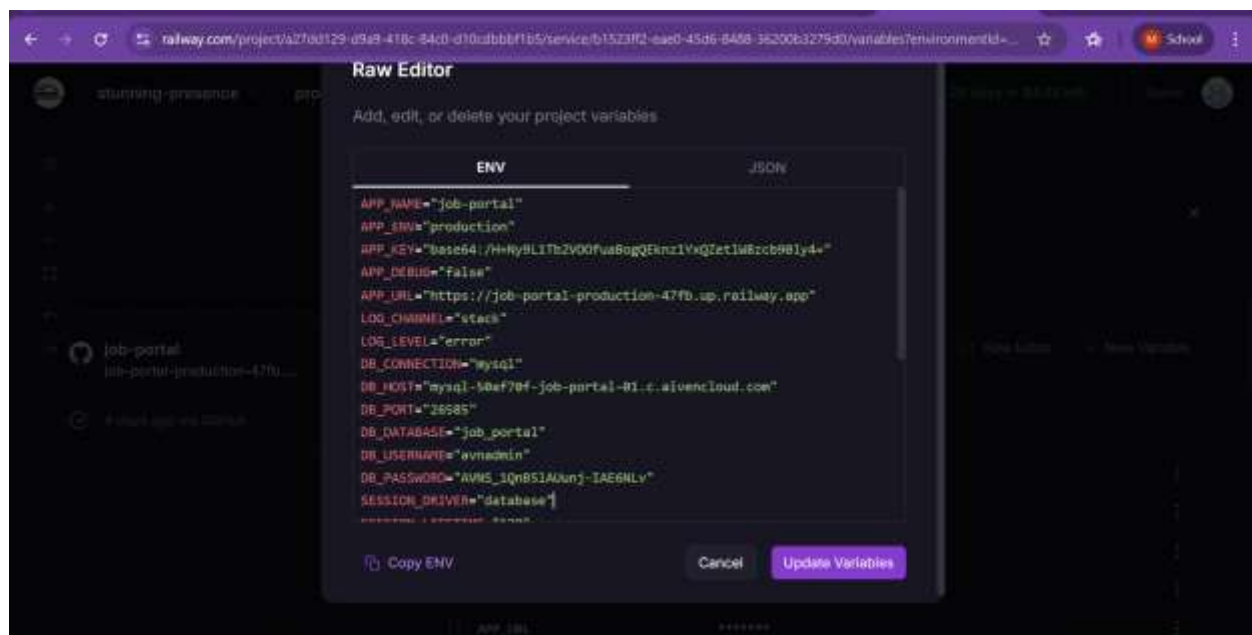
### Step 3: Deploy Backend to Railway

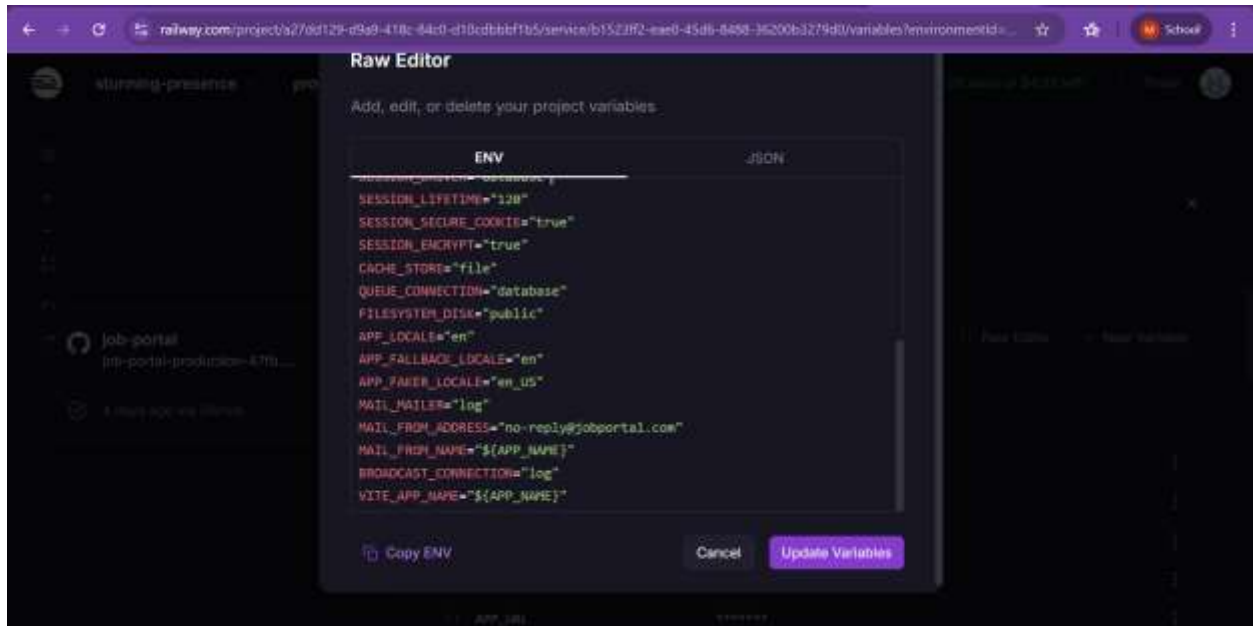
- Go to Railway.app.
- Click on “New Project” → “Deploy from GitHub”.
- Select your Laravel project repository



## STEP 4: Configuring Variable

- After deployment, open the “Variables” tab in Railway.
- Add the following environment variables (from your .env file):





## STEP 5: ACCESS USING MYSQL CLIENT:

```
Command Prompt - mysqlsh --sql mysql://avnadmin:AVNS_1QnBS1AUunJ-IAE6NLv@mysql-50af70f-job-portal-01.c.aivencloud.com:26585/job_portal?ssl-mode=REQUIRED
Microsoft Windows [Version 10.0.19045.5965]
(c) Microsoft Corporation. All rights reserved.

C:\Users\T470s>mysqlsh --sql mysql://avnadmin:AVNS_1QnBS1AUunJ-IAE6NLv@mysql-50af70f-job-portal-01.c.aivencloud.com:26585/job_portal?ssl-mode=REQUIRED
MySQL Shell 8.0.28

Copyright (c) 2016, 2022, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its affiliates.
Other names may be trademarks of their respective owners.

Type '\help' or '\?' for help; '\quit' to exit.
WARNING: Using a password on the command line interface can be insecure.
Creating a Classic session to 'avnadmin@mysql-50af70f-job-portal-01.c.aivencloud.com:26585/job_portal?ssl-mode=REQUIRED'
Fetching schema names for autocompletion... Press ^C to stop.
Fetching table and column names from 'job_portal' for auto-completion... Press ^C to stop.
Your MySQL connection id is 77402
Server version: 8.0.35 Source distribution
Default schema set to 'job_portal'.
MySQL mysql-50af70f-job-portal-01.c.aivencloud.com:26585 ssl job_portal SQL > show tables;

+-----+
| Tables_in_job_portal |
+-----+
| applications          |
| employers             |
| jobs                  |
| jobseekers            |
| migrations            |
| resumes               |
| sessions              |
+-----+
```

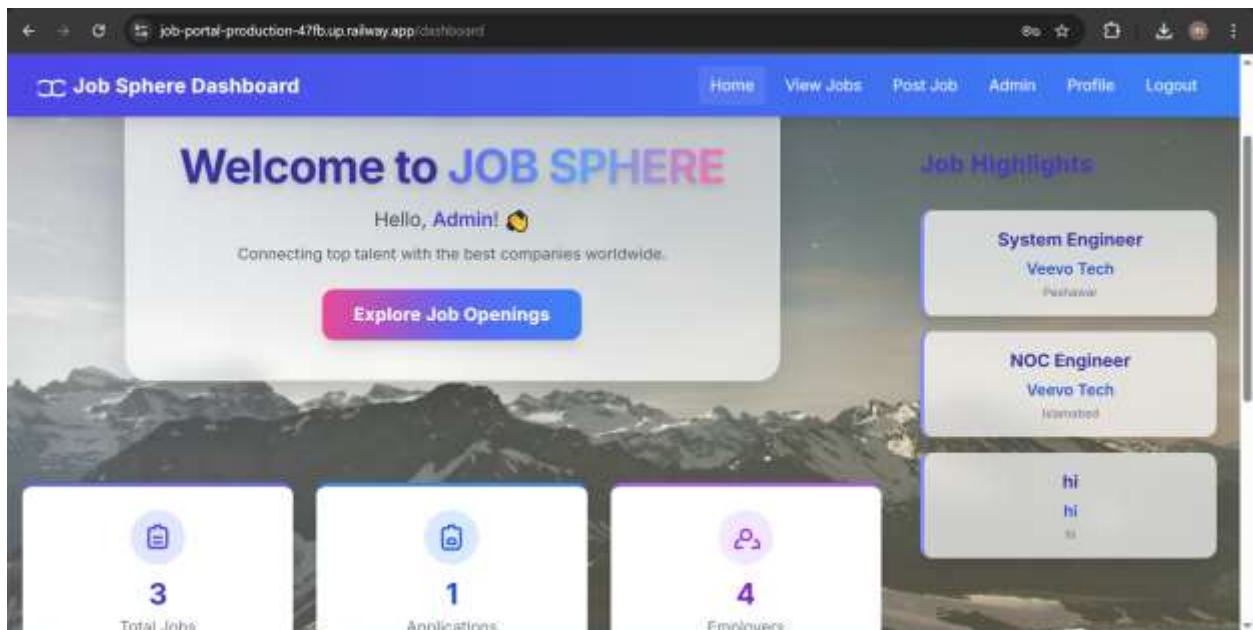
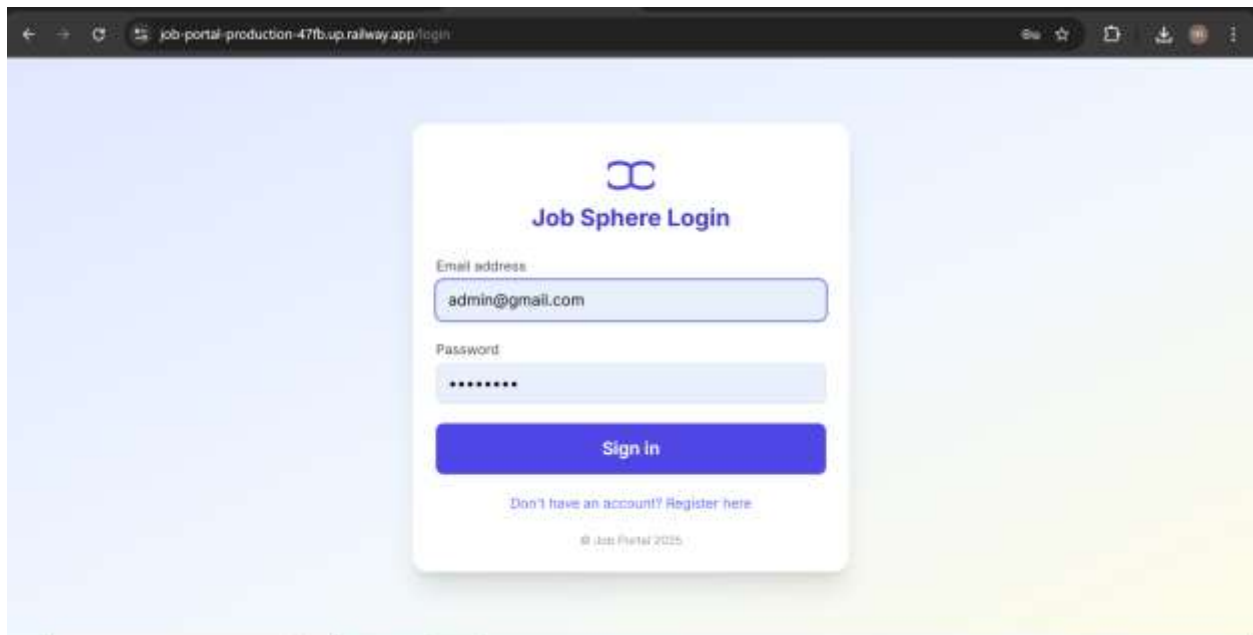
Deployment link:

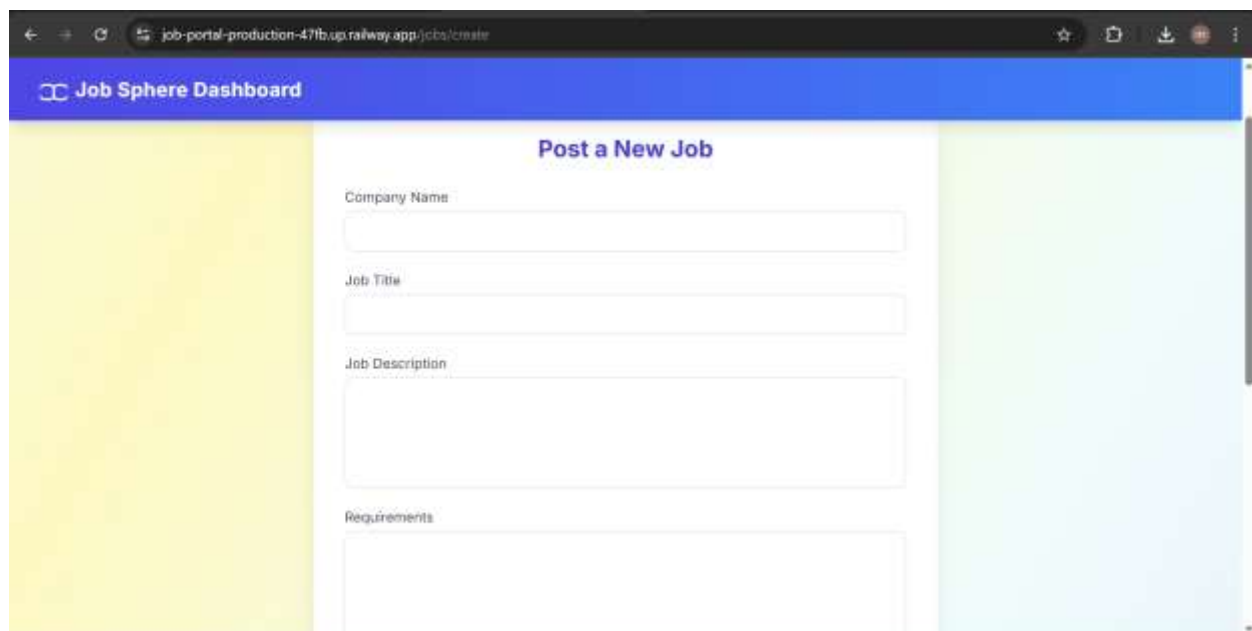
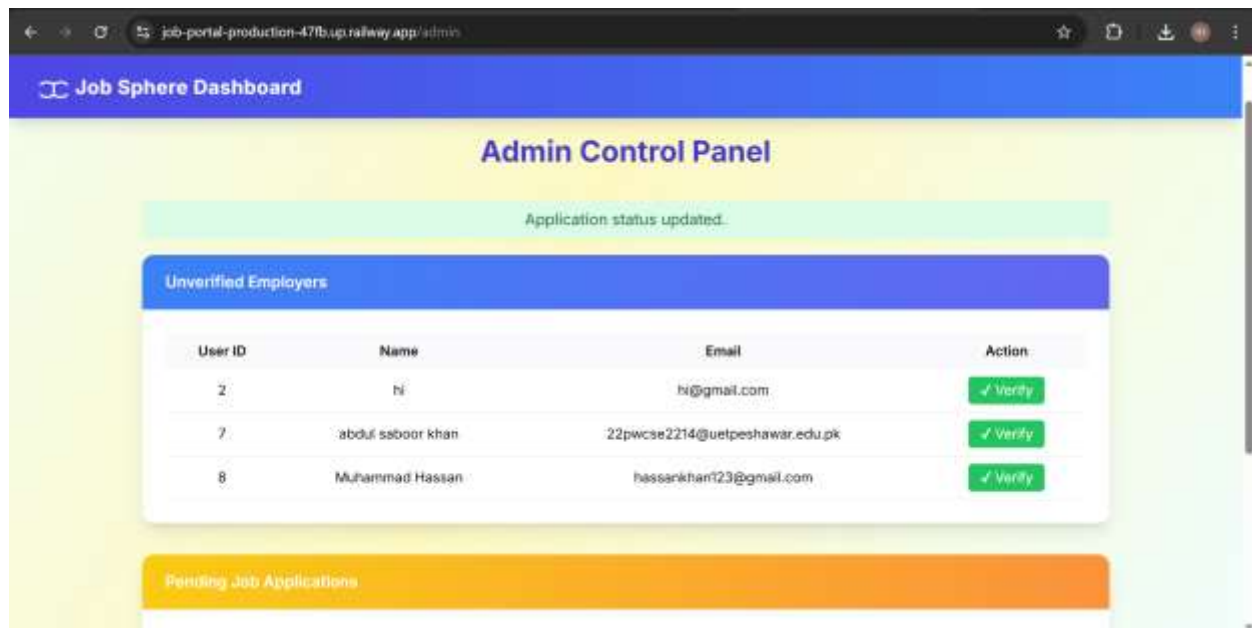
<https://job-portal-production-47fb.up.railway.app>

## Function:

### Admin:

- Admin can review or reject the applications.
- Admin can verify the employer
- Can post new job.





## Employer:

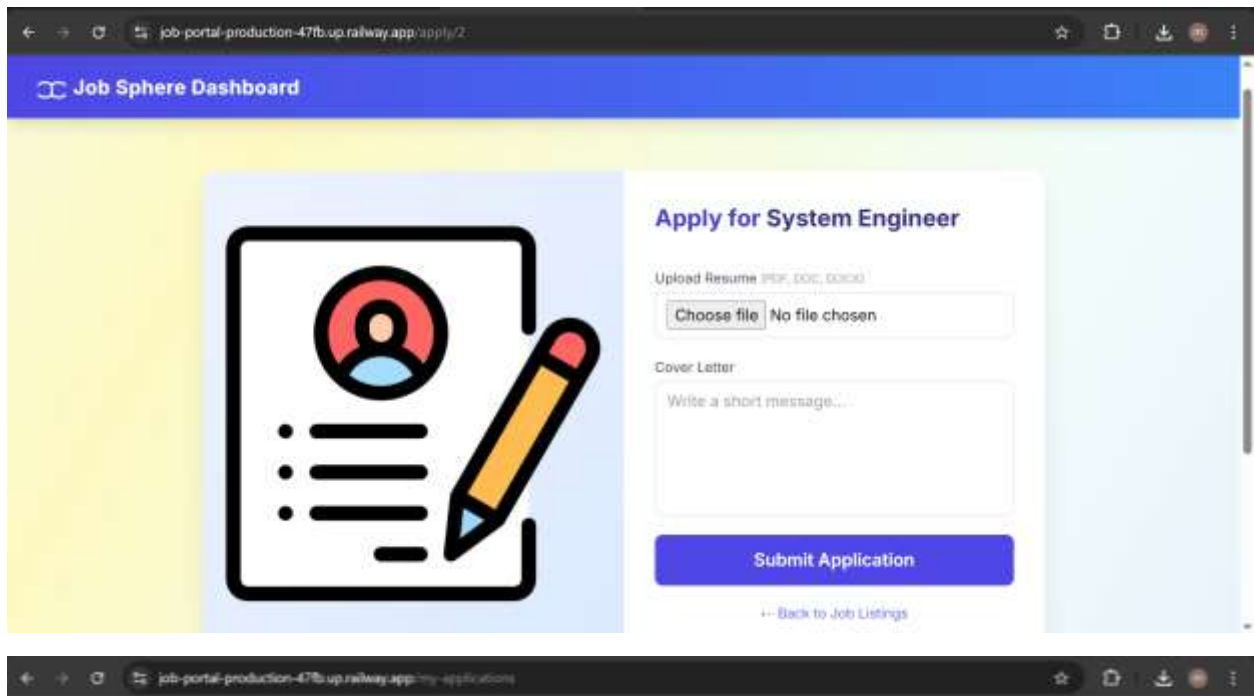
- Employer can post new job.
- Employer itself can apply for new job.
- Can Change the profile info.
- View the job page

The screenshot shows a web browser window with the URL `job-portal-production-47fb.up.railway.app/jobs/create`. The page has a blue header with the 'Job Sphere Dashboard' logo. The main content area is titled 'Post a New Job' and contains a form with four input fields: 'Company Name', 'Job Title', 'Job Description', and 'Requirements'. The form is set against a light blue background with a yellow sidebar on the left.

## JOB SEEKER:

- Job seeker can submit the application for job.
- View the job page.
- Check the application status.
- Change the profile info.

The screenshot shows a web browser window with the URL `job-portal-production-47fb.up.railway.app/jobs`. The page has a blue header with the 'Job Sphere Dashboard' logo. The main content area is titled 'Find Your Next Opportunity' with the subtitle 'Search thousands of jobs from top companies'. Below the title is a search bar with the placeholder text 'Job title, keywords, or company' and a 'Search' button. The search results are displayed in three columns, each showing a job listing for 'System Engineer', 'NOC Engineer', and 'hi'. Each listing includes a company logo, the job title, a brief description, and buttons for 'Show Details' and 'Apply Now'.



### My Applications

Job Title	Applied Date	Status
System Engineer	03 Jul, 2025	Reviewed

## Conclusion

This project provided practical experience with full-stack development using Laravel. The system successfully connects employers and jobseekers with essential features and role-based access. It is scalable, easy to manage, and user-friendly.

## Deployment link:

<https://job-portal-production-47fb.up.railway.app>

## References

- OpenAI. (2024). *ChatGPT conversations for code assistance and debugging*. Retrieved from <https://chat.openai.com>
- diagrams.net. (n.d.). *Free diagram software*. Retrieved from <https://www.diagrams.net>
- Railway. (n.d.). *Railway hosting documentation*. Retrieved from <https://docs.railway.app>

- Aiven. (n.d.). *Database service dashboard for job-portal-01 project*. Retrieved from <https://console.aiven.io/account/a543c12a7272/project/job-portal-01/services>