

A Software Engineering Project On

Job Portal

as the partial fulfillment of

Semester -VI study

for the degree of

Bachelor of Science (Computer Science)
(Affiliated to Veer Narmad South Gujarat University)

during the academic year 2022-2023.

Developed By:

Student Name
Bhakhar Dhruvan Prakashbhai

Guided By:

Guide Name
Shweta Bhatiya



Sarvajanik Education Society
Shree Ramkrishna Institute of Computer Education and Applied Sciences
Behind P.T Science College, M.T.B. College Campus, Athwalines, Surat-395 001.



Sarvajanik Education Society

Shree Ramkrishna Institute of Computer Education and Applied Sciences
Behind P.T. Science College, M.T.B College Campus, Athwalines, Surat-395 001.



CERTIFICATE

DEPARTMENT OF COMPUTER SCIENCE

This is to certify that

<i>Student name</i>	<i>Exam no</i>
1) Bhakhar Dhruvan Prakashbhai	20186

has / have successfully completed his / her / their software engineering project work entitled

Job Portal

as the partial fulfillment of

Semester -VI study
for the degree of

Bachelor of Science (Computer Science)

*(Affiliated to Veer Narmad South Gujarat University)
during the academic year 2022-2023.*

Internal Project Guide

HOD
Dept. of Computer Science

Date: _____

Place: Surat

University Examination:

Examination No.: _____

Signature of External Examiners: _____

INDEX for Project Documentation

SR.NO	CONTENTS	Page. No
1.	Introduction	
	• Project Profile	04
2.	System Introduction	
	• System Definition, Objective and scope	05
	• Hardware & Software Requirements	06
3.	Requirement Analysis & Modeling	
	• Expected working of system	07
	• Data Flow Diagrams (DFD)	08
	• Process Specification	11
	• Data Dictionary	12
4.	Design	
	• ER Diagram & Database Design	15
	• System Architecture	16
	• System Flowchart	19
	• Form & Report Designs (Screenshots) Layouts	21
5.	Coding	
	• Coding Approach/style	30
	• Code Snippets	30
6.	Testing	
	• Testing Data Inputs (Data Validations)	34
	• Testing Operations (Actions on each screen)	
7.	New Tools/ Technologies learned/used	
8.	System Limitations/Restrictions and Dependencies/Constraints	
9.	Future Enhancement & Opportunities	
10.	Bibliography & References	

1) INTRODUCTION

❖ Project Profile:

Project Title	Job Portal
Organization	Shree Ramakrishna Institute Of Education and Applied Sciences.
Front -End tool	HTML, CSS
Back -End tool	PHP & My SQL
Operating System	Windows 11
Project Duration	3 Month
Project Guided By	Shweta Bhatiya
Developed By	Bhakhar Dhruvan Prakashbhai

2) System introduction

❖ System Definition:

A job portal is an online platform or website that allows employers or recruiters to post job vacancies and job seekers to search and apply for jobs. These portals typically feature job listings from a variety of industries and locations, and offer tools for job seekers to create profiles, upload resumes, and receive job alerts. Additionally, some job portals may include features such as career advice, salary data, and employer reviews. The primary goal of a job portal is to connect employers and job seekers in an efficient and effective manner.

❖ Objective:

The online job Portal System that is to be developed provides the members with jobs information, online applying for jobs and many other facilities. This system provides service to the job applicants to search for working opportunities.

Job Portal will allow job provider to establish one to one relationships with candidates. This Portal will primarily focus on the posting and management of job vacancies. This system is designed such that ultimately all vacancies will be posted online and would offer employers the facilities to post their vacancies online. It helps to review and manage the resulting applications efficiently through the web. Employer can also find the resume according to key skill in very less amount of time.

❖ Scope:

As of Indian market, there is ample opportunities for the job portal sites, as more and more number of educated and skilled young people are coming out each and every year. Also, as the growth rate of India is zooming to be at a healthy rate over 7%, so it is boom time for corporate also. So, more and more number of lucrative careers will be available for the job seekers. So, it is now the right period for the job portal sites to think out of the box, and to make most of the opportunities available.

❖ Hardware & Software Requirements:

Hardware:-

- **Processor:** Intel(R) Core(TM) i7
- **RAM:** 16GB
- **Hard - Disk:** 100GB or Above

Software:-

- **Backend:** PHP & My Sql (PHP MY Admin)
- **Frontend:** HTML & CSS
- **Operating System:** Windows 11

3) Requirement Analysis & Modeling

❖ Expected Working of System:

The System is Proposed to have following modules:

- Admin Module
- User Module

➤ Admin Module:

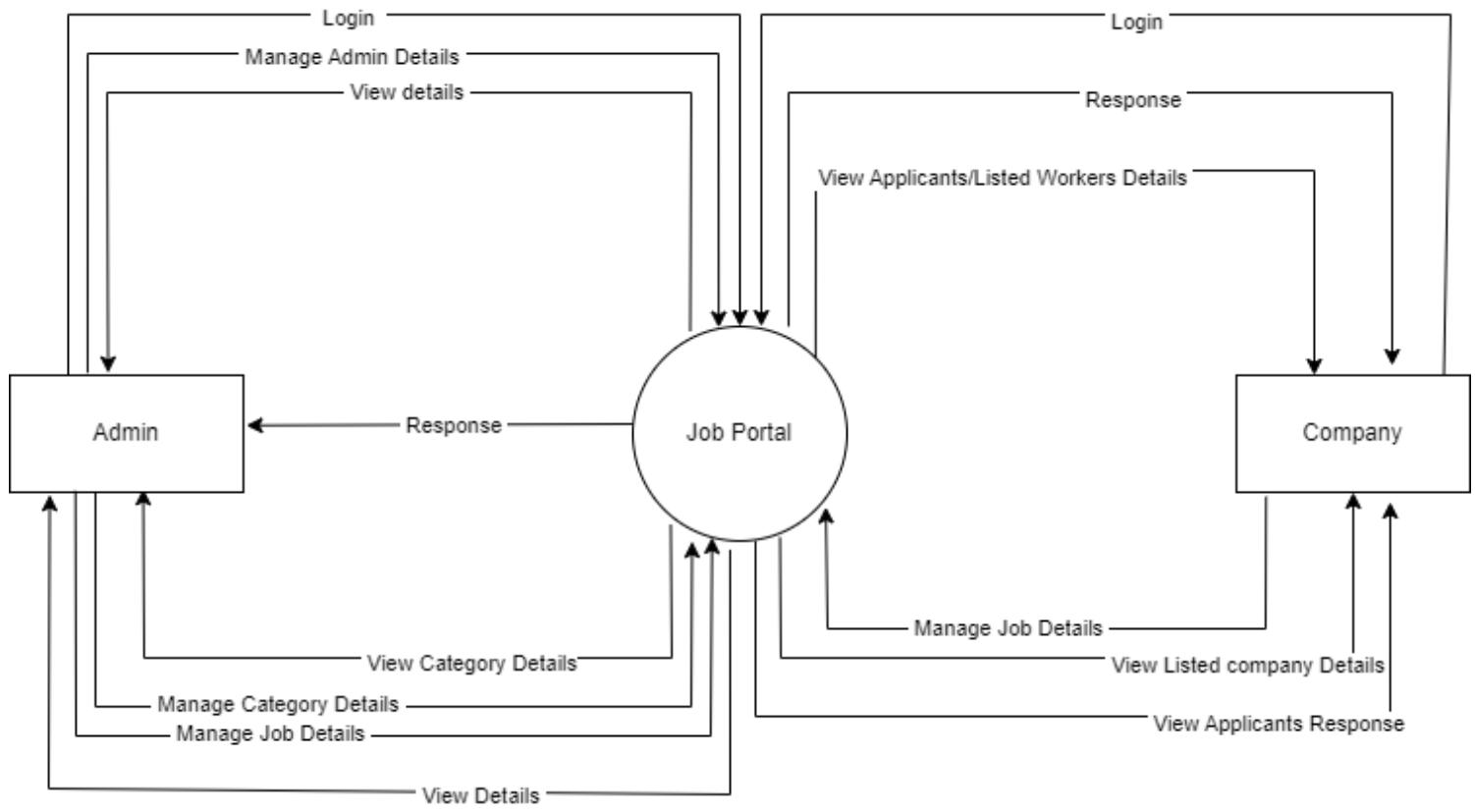
- Admin Login
- Admin Registration
- Admin Dashboard
- Add Admin
- Manage Categories
- Verify Jobs
- Delete Jobs

➤ User Module:

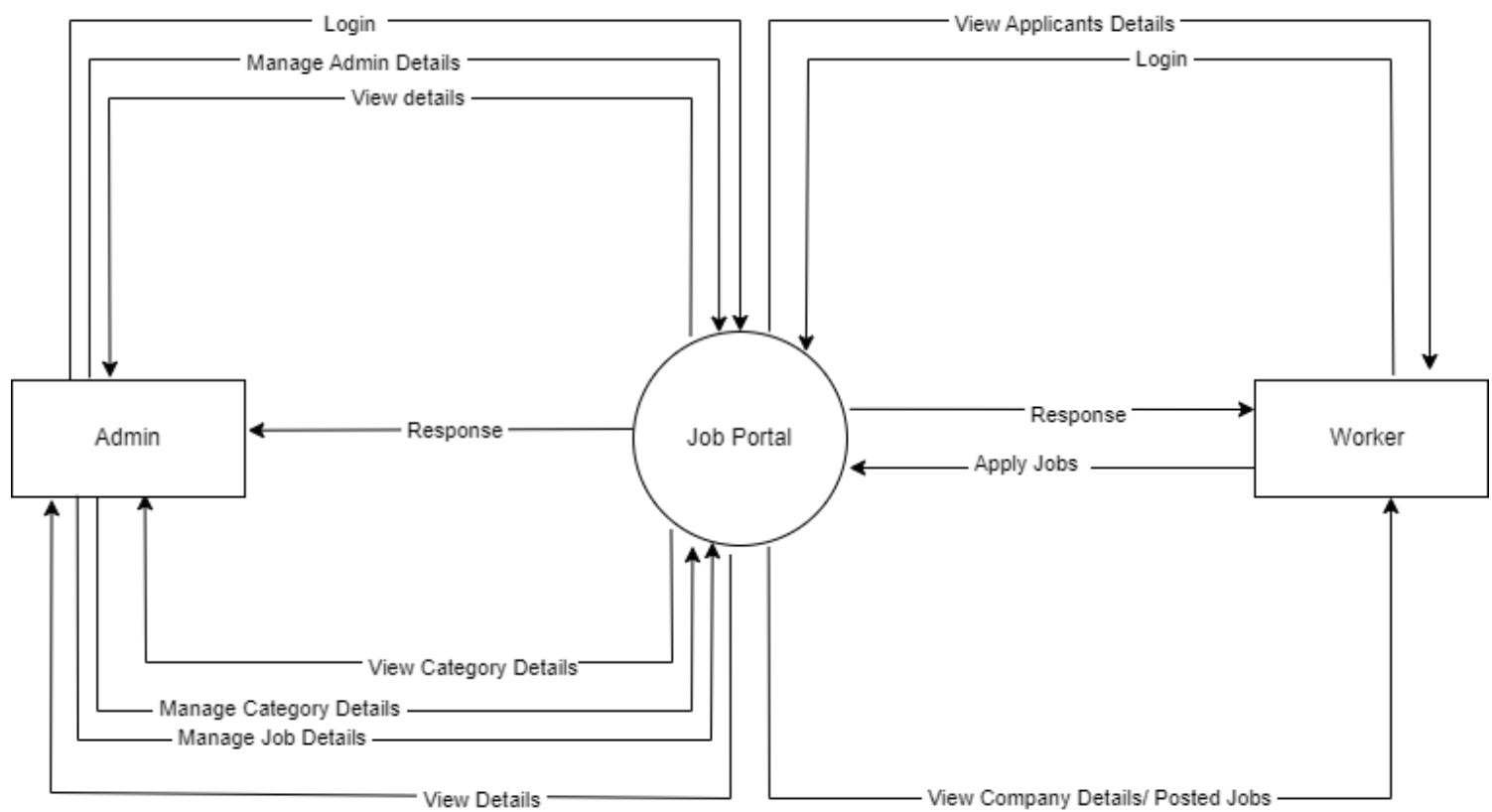
- Index
- User Login
- New User Registration
- User Dashboard
- Company Post Jobs
- Workers Apply for Jobs
- List of Jobs
- List of Workers
- List of Company
- Search for Jobs

❖ Data Flow Diagram:

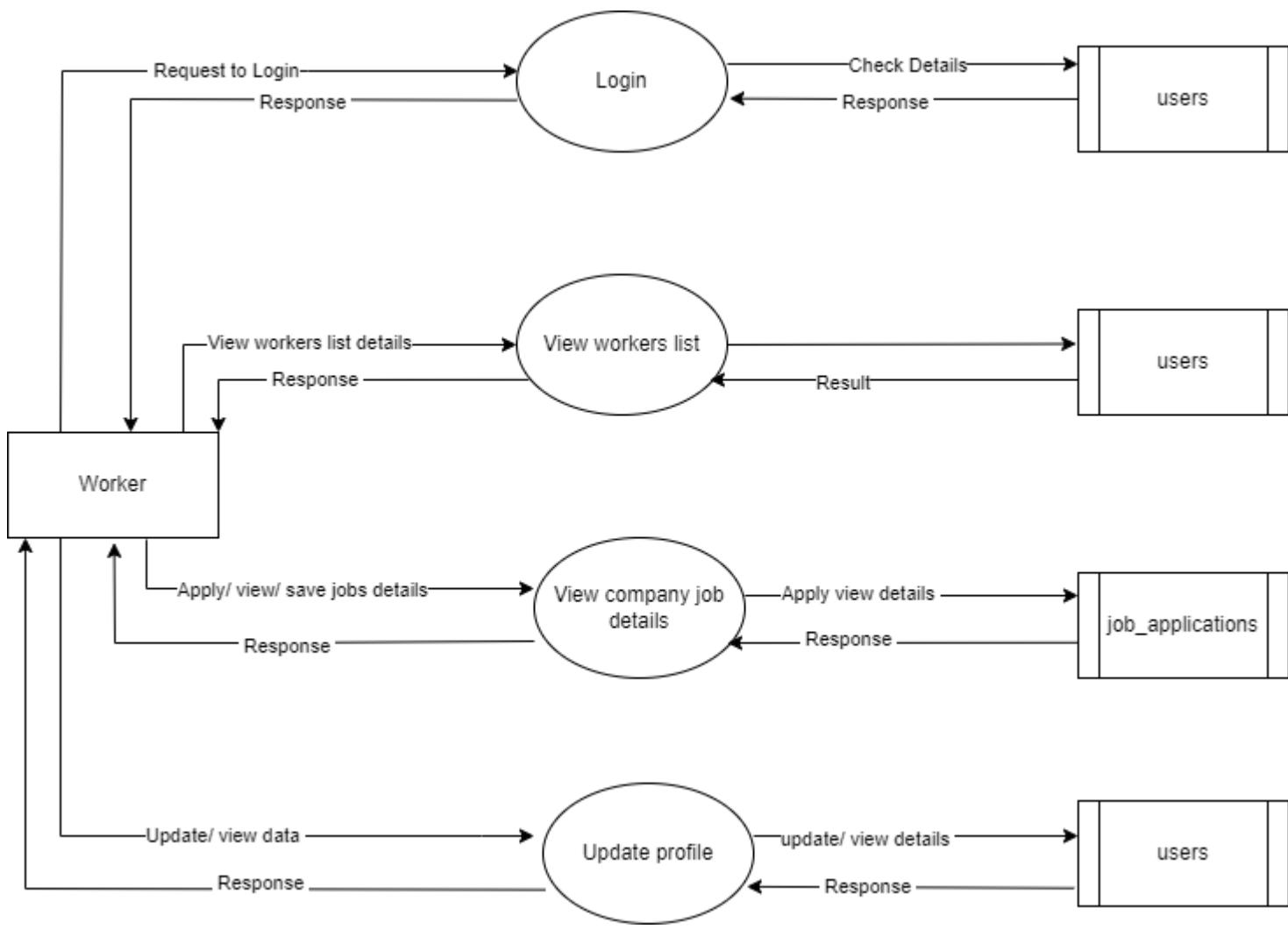
- **0 Level DFD:**



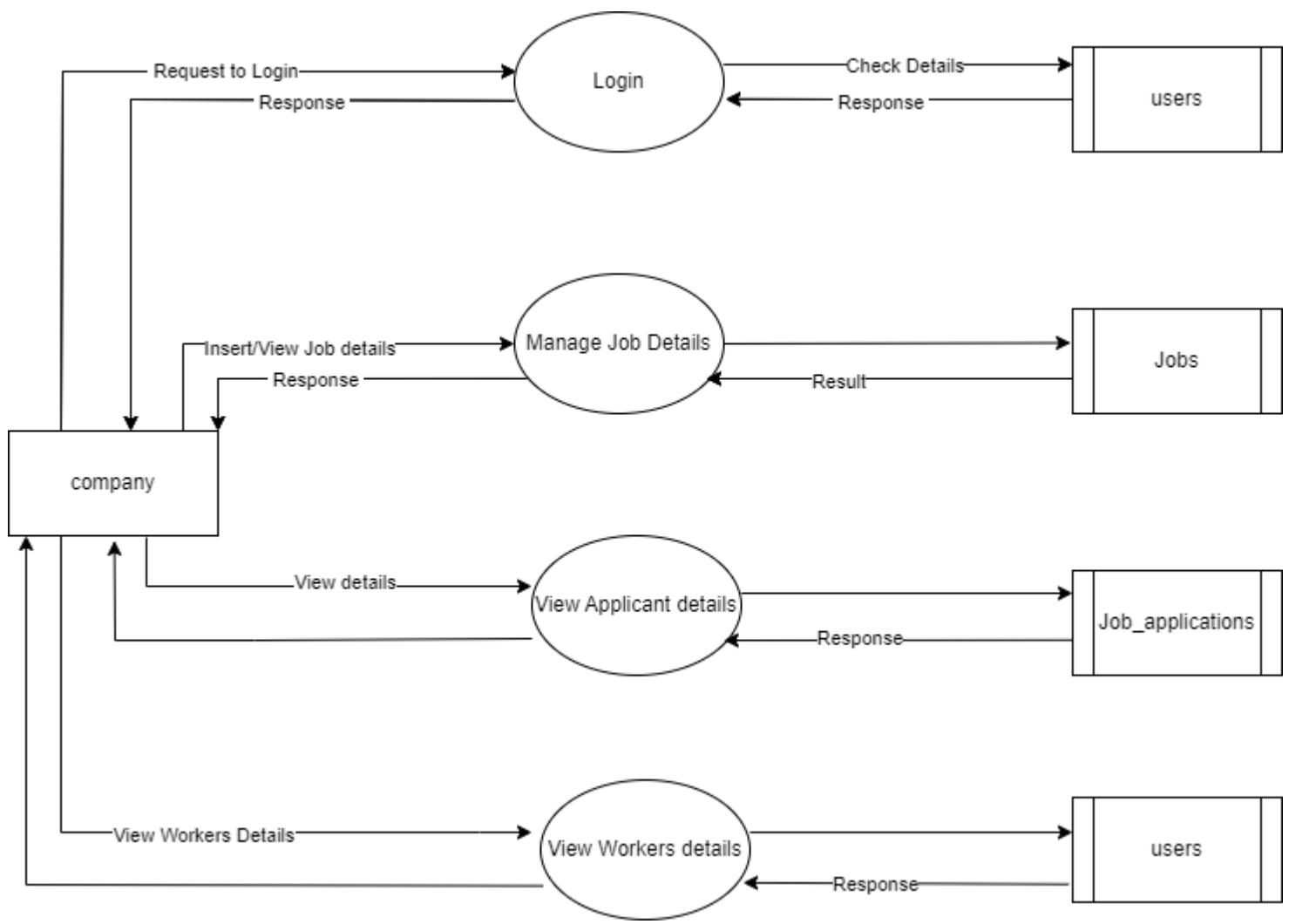
- 0 Level DFD:



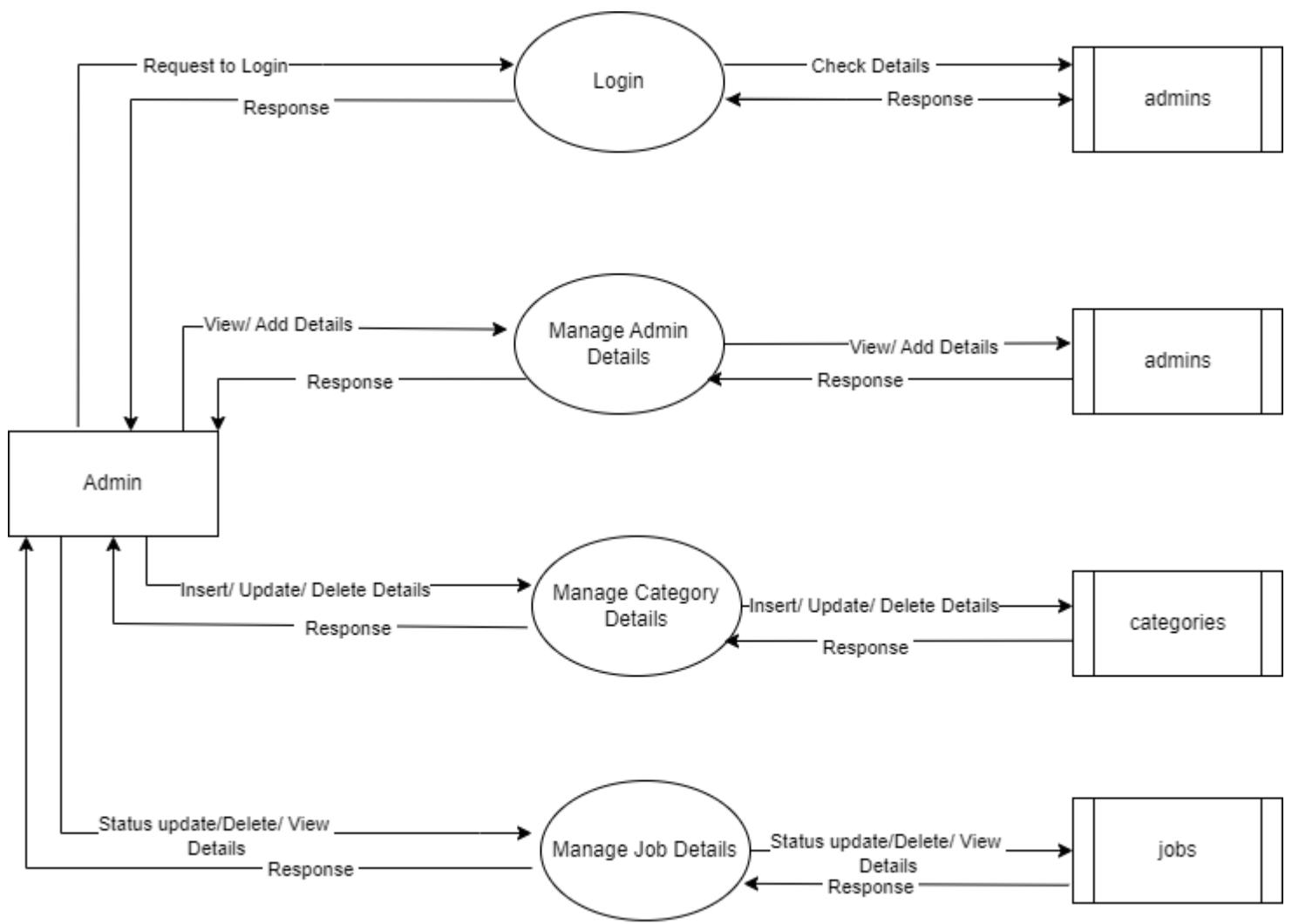
- **1 Level DFD Worker:**



- **1 Level DFD Company:**



- **1 Level DFD Admin:**



❖ Process Specification:

- **Process: 1**

Name	Login
Description:	In this Process User and Admin Login With use of Their Email And Password.

- **Process: 2**

Name	Registration
Description:	In this Process Insert a new record in database. In short save your information of admin/user like ID , Username , Email id, Type,Password.

- **Process: 3**

Name	Admin Dashboard
Description:	Total users, Total Categories, Total Jobs, include in this Process.

- **Process: 4**

Name	Management Categories
Description:	In this Process Admin can update, delete and insert categories.

- **Process: 5**

Name	Admin Information
Description:	Admin can view Admin's information in this process.

- **Process: 6**

Name	Jobs
Description:	Admin can view all jobs and change its status and delete the jobs.

- **Process: 7**

Name	Company Post jobs
Description:	In this process company can post a jobs.

- **Process: 8**

Name	Apply jobs
Description:	Worker can apply for the jobs and he can save it also.

❖ **Data Dictionary**

- **Admin Table**

Name:	admin details
Alias:	admin

Where used/How used:	Admin login.
Content Description:	id+ adminname+ email+ password+ created_at+

- **Categories details:**

Name:	categories details
Alias:	categories
Where used/How used:	categories show
Content Description:	id+ name+ created_at+

- **Jobs details:**

Name:	jobs details
Alias:	jobs
Where used/How used:	jobs
Content Description:	id+ job_title+ job_region+ job_type+ vacancy+ job_category+ experience+

	salary+ gender+ application_deadline+ job_description+ responsibilities+ eduction_experience+ other_benifits+ company_email+ company_name+ application_deadline+ job_description+ responsibilities+ eduction_experience+ other_benifits+ compny_email+ company_name+ company_id+ company_image+ status+ created_at+
--	--

- **Job application table:**

Name:	Job application detail
Alias:	Job application
Where used/How used:	Job application
Content Description:	id+ username+ email+ cv+ worker_id+ job_id+ job_title+ company_id+

	create_at+ status+
--	-----------------------

- **Saved job Table**

Name:	Saved job details
Alias:	Saved_jobs
Where used/How used:	Users saved job
Content Description:	id+ job_id+ worker_id+ created_at+

- **Searches Table**

Name:	Searches details
Alias:	searches
Where used/How used:	Job search
Content Description:	id+ keyword+ created_at+

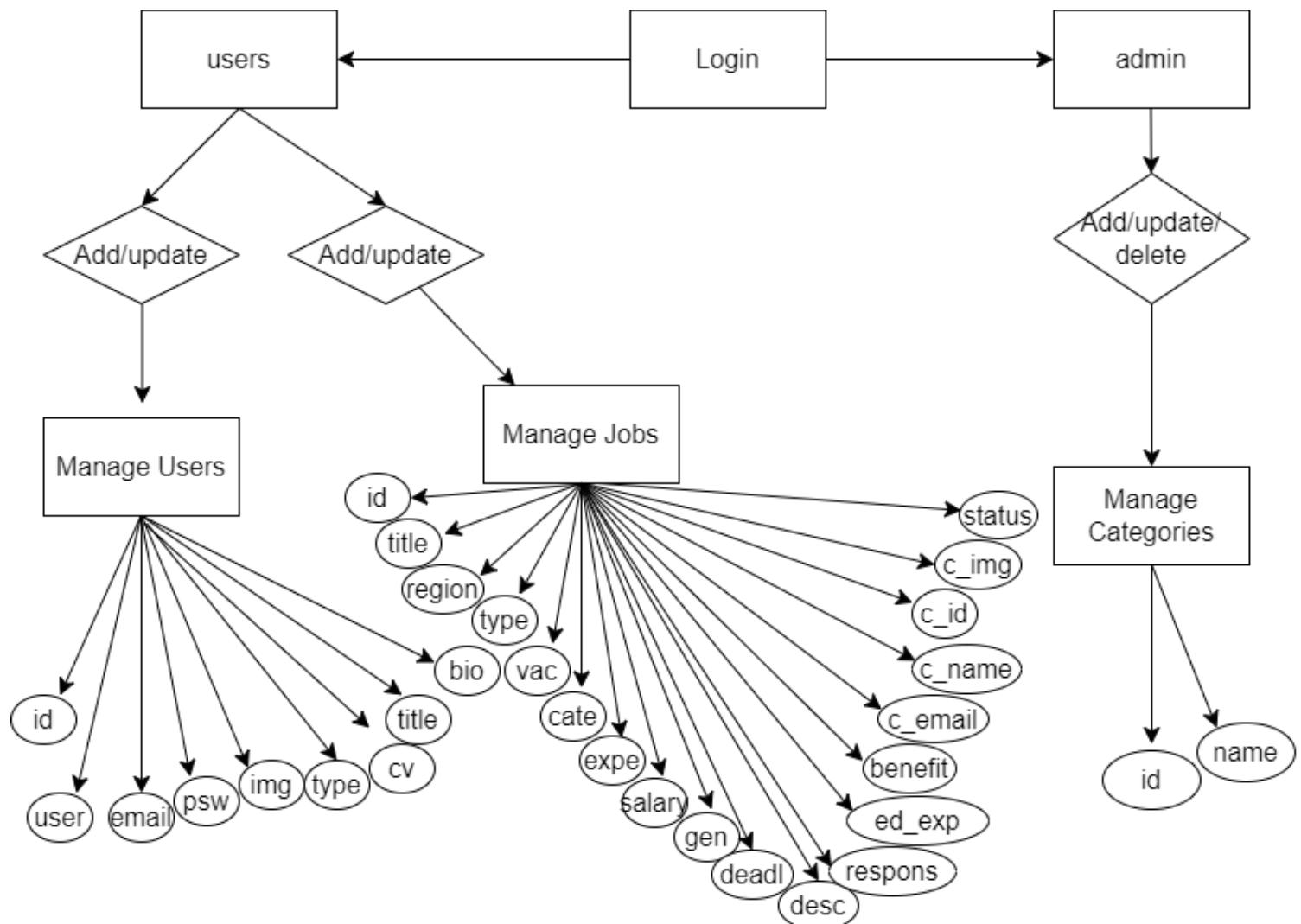
- **User Table**

Name:	User details
--------------	--------------

Alias:	users
Where used/How used:	User login
Content Description:	id+ username+ email+ password+ img+ type+ cv+ title+ bio+ facebook+ twitter+ linkedin+ created_at+

4) Design

❖ E-R Diagram:



❖ Database design:

- Table Name: **admins**

Field Name	Data Type	Field Size	Constraints	Description
Id	Int	3	Primary Key	Contains ID of admin
adminname	Varchar	200		Contains adminname of admin
email	Varchar	200		Contains email of admin
password	Varchar	200		Contains password of admin
Created_at	Timestamp			Contains current time.

- Table Name: **Categories**

Field Name	Data Type	Field Size	Constraints	Description
id	Int	3	Primary Key	Contains id of categories

name	Varchar	200		Contains name of categories
Created_at	Timestamp			Contains current timestamp of categories

- Table Name: Jobs

Field Name	Data Type	Field Size	Constraints	Description
id	Int	3	Primary Key	Contains id of jobs
Job_title	varchar	200		Contains Job title of jobs
Job_region	varchar	200		Contains region of jobs
Job_type	varchar	200		Contains type of jobs
Vacancy	varchar	200		Contains vacancy of jobs
Job_category	varchar	200		Contains category of jobs
experience	varchar	200		Contains experience of jobs
salary	varchar	200		Contains salary of jobs
Gender	varchar	200		Contains gender of jobs
Application_deadline	varchar	200		Contains application_deadline of jobs

Job_description	varchar	200		Contains job_description of jobs
responsibilities	Text	200		Contains responsibilities of jobs
Education_experience	Text			Contains experience of jobs
Other_benifits	Text			Contains Other_benifits of jobs
Company_email	varchar	200		Contains Company_email of jobs
Company_name	varchar	200		Contains Company_name of jobs
Company_email	varchar	200		Contains Company_email of jobs
Company_id	Int	3		Contains Company_id of jobs
Company_image	varchar	200		Contains Company_image of jobs
Status	int	3		Contains status of jobs

- Table Name: **Job_applications**

Field Name	Data Type	Field Size	Constraints	Description
id	Int	3	Primary Key	Contains Application number of job applications
username	varchar	200		Contains username of applications
Email	Varchar	200		Contains email of applications
cv	Varchar	200		Cv of applicant
Worker_id	int	3		Contains id of worker
Job_id	int	3		Contains id of jobs
Company_id	int	3		Contains id of company
status	Int	3		Contains status of job_applications

- Table Name: **Saved_jobs**

Field Name	Data Type	Field Size	Constraints	Description

id	Int	3	Primary Key	Contains id of saved_jobs
Job_id	Int	3		Contains job_id
Worker_id	int	3		Contains worker id
Created_at	Timestamp			Contains current timestamp of saved_jobs

- Table Name: **Searches**

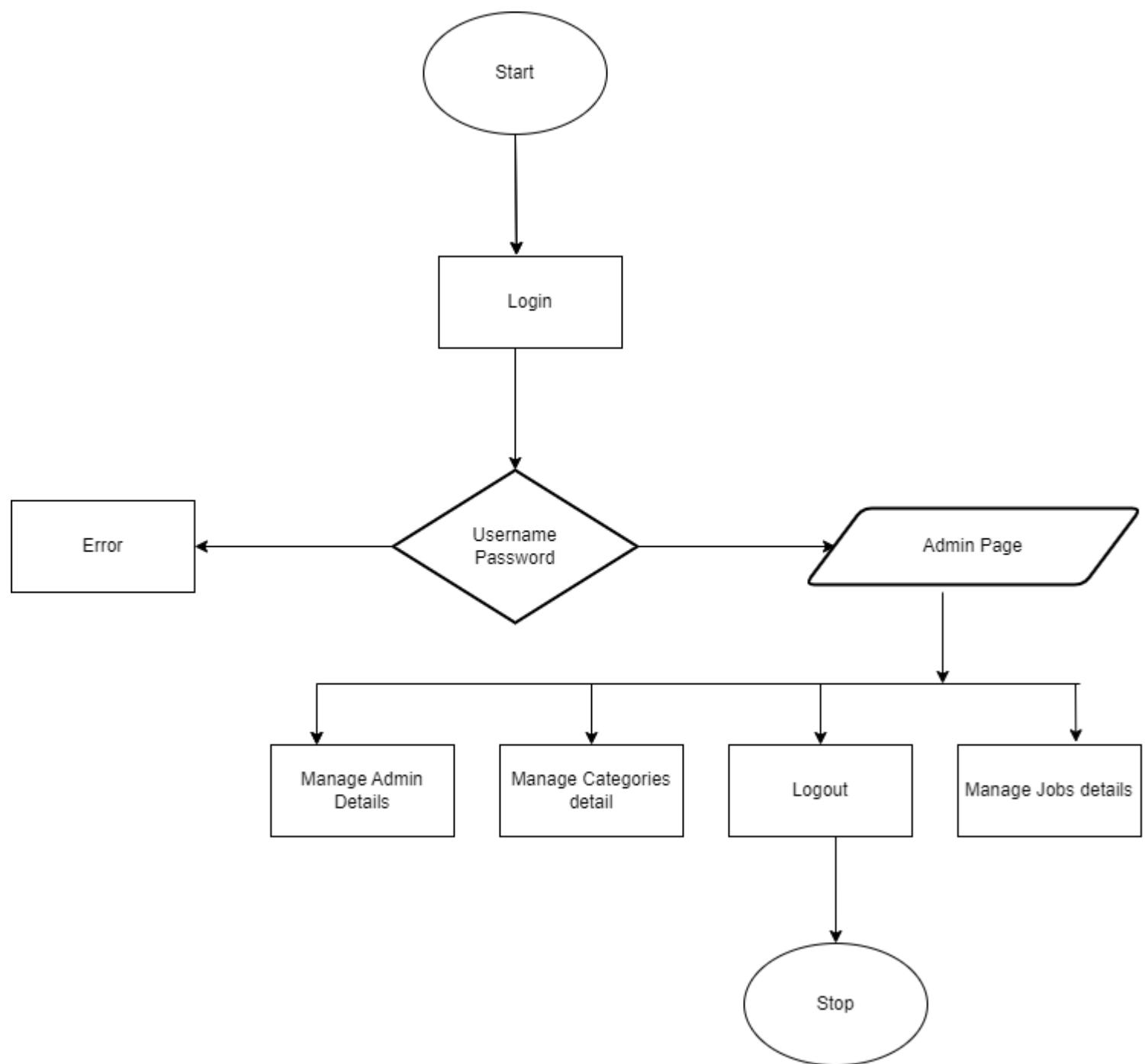
Field Name	Data Type	Field Size	Constraints	Description
id	Int	3	Primary Key	Contains id of searches
keyword	Varchar	200		Contains keyword of searches
Created_at	Timestamp			Contains current timestamp of searches

- Table Name: **users**

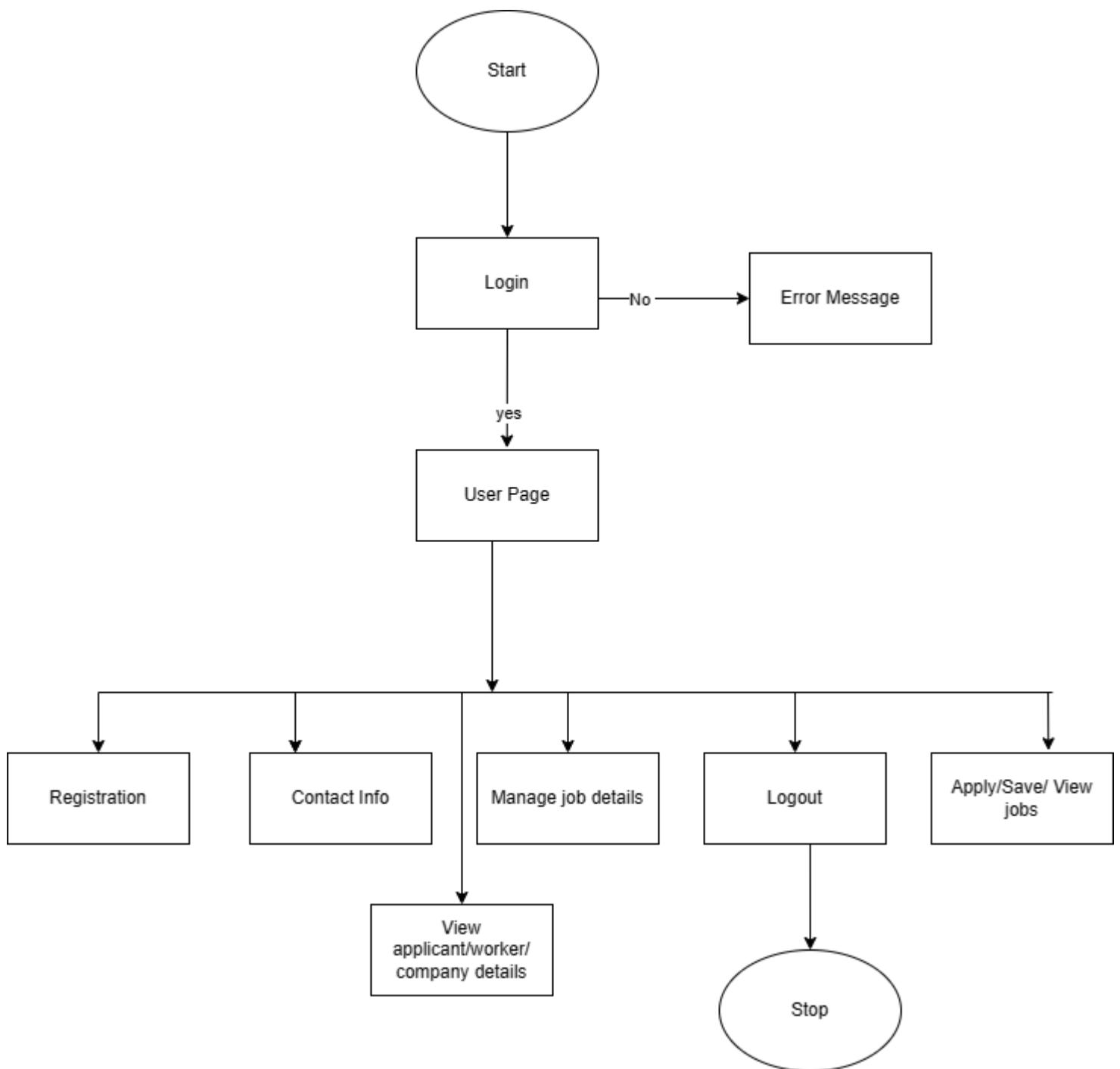
Field Name	Data Type	Field Size	Constraints	Description
id	Int	3	Primary Key	Contains id of users

username	Varchar	200		Contains username of users
Password	varchar	200		Contains Password of users
Img	Varchar	100		Contains images of users
Type	Varchar	200		Contains type of users
Cv	Varchar	200		Contains cv of users
Title	Varchar	200		Contains title of users
Bio	Varchar	200		Contains bio of users
Facebook	Varchar	200		Contains facebook of users
Twitter	Varchar	200		Contains twitter of users
Linkedin	Varchar	200		Contains linkedin of users
Created_at	Timestamp			Contains current timestamp of users

❖ Flow Chart Admin:

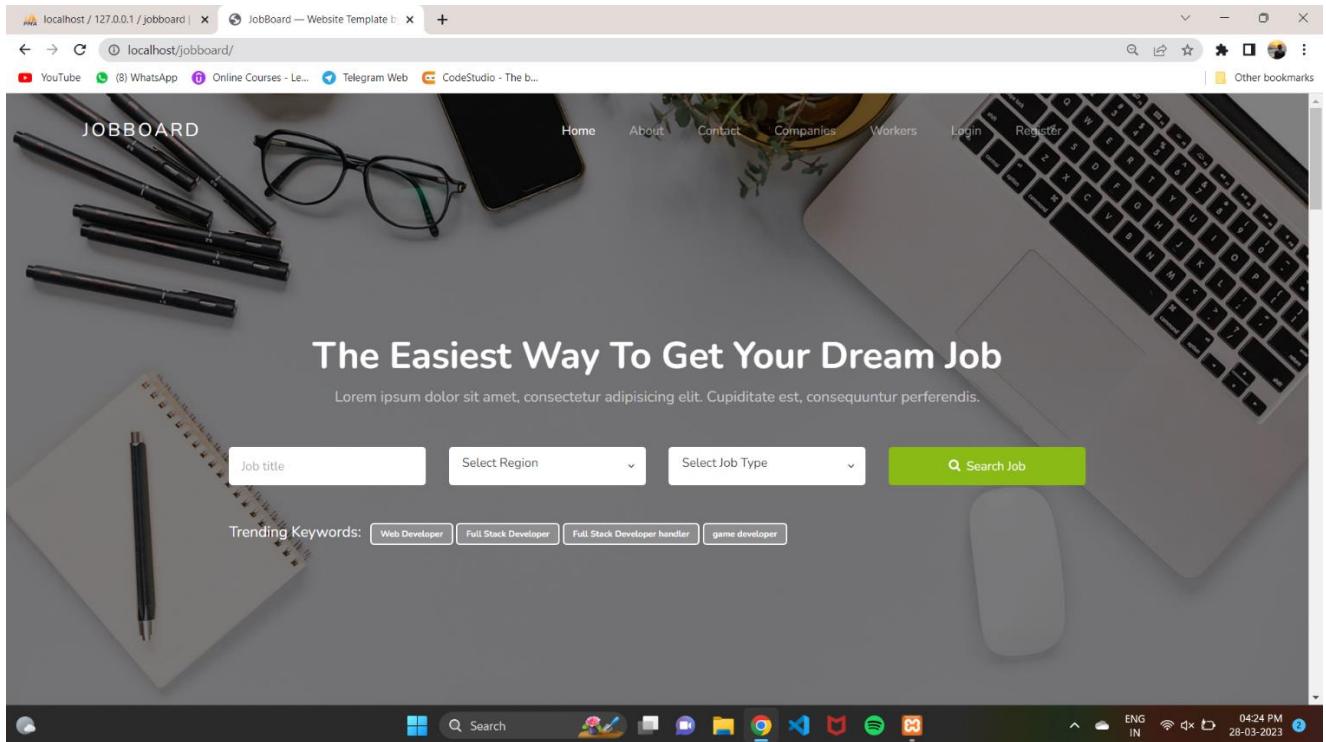


❖ Flow Chart Users:

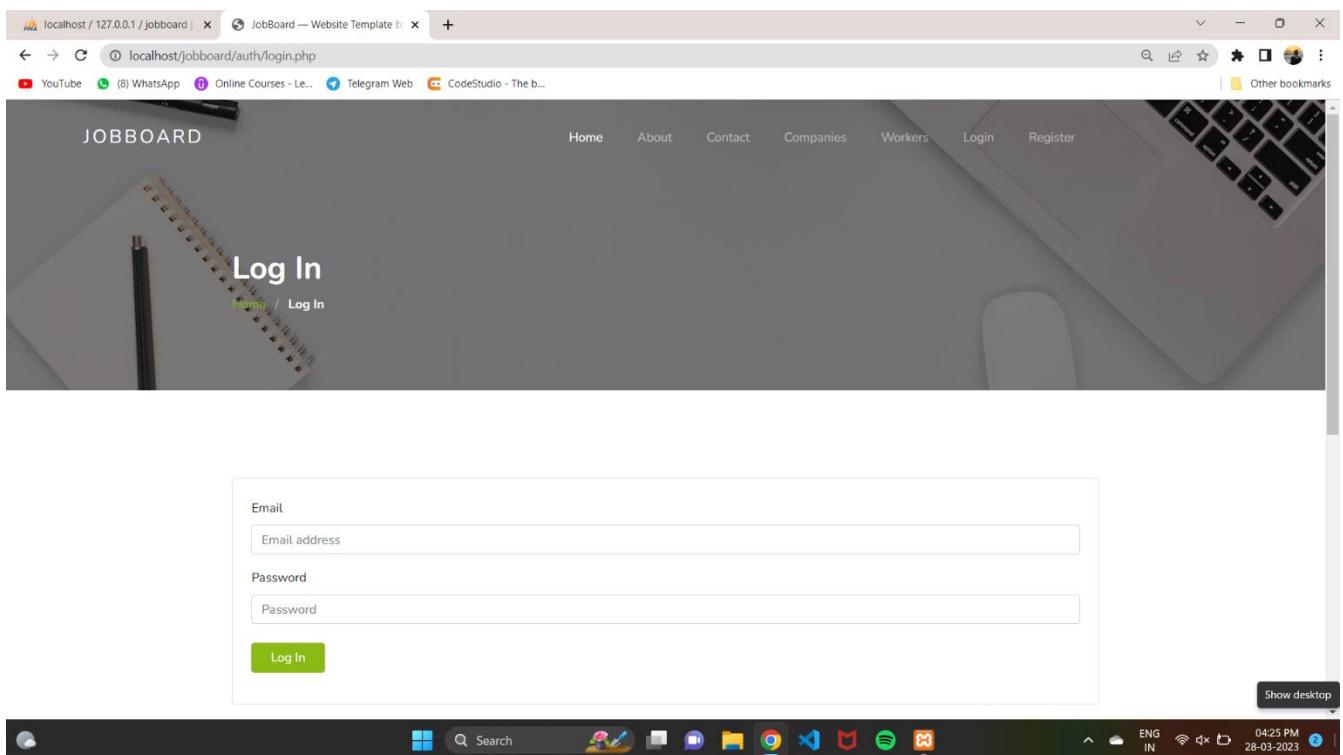


❖ Form & Report Designs:

- Index page



- Login page for user



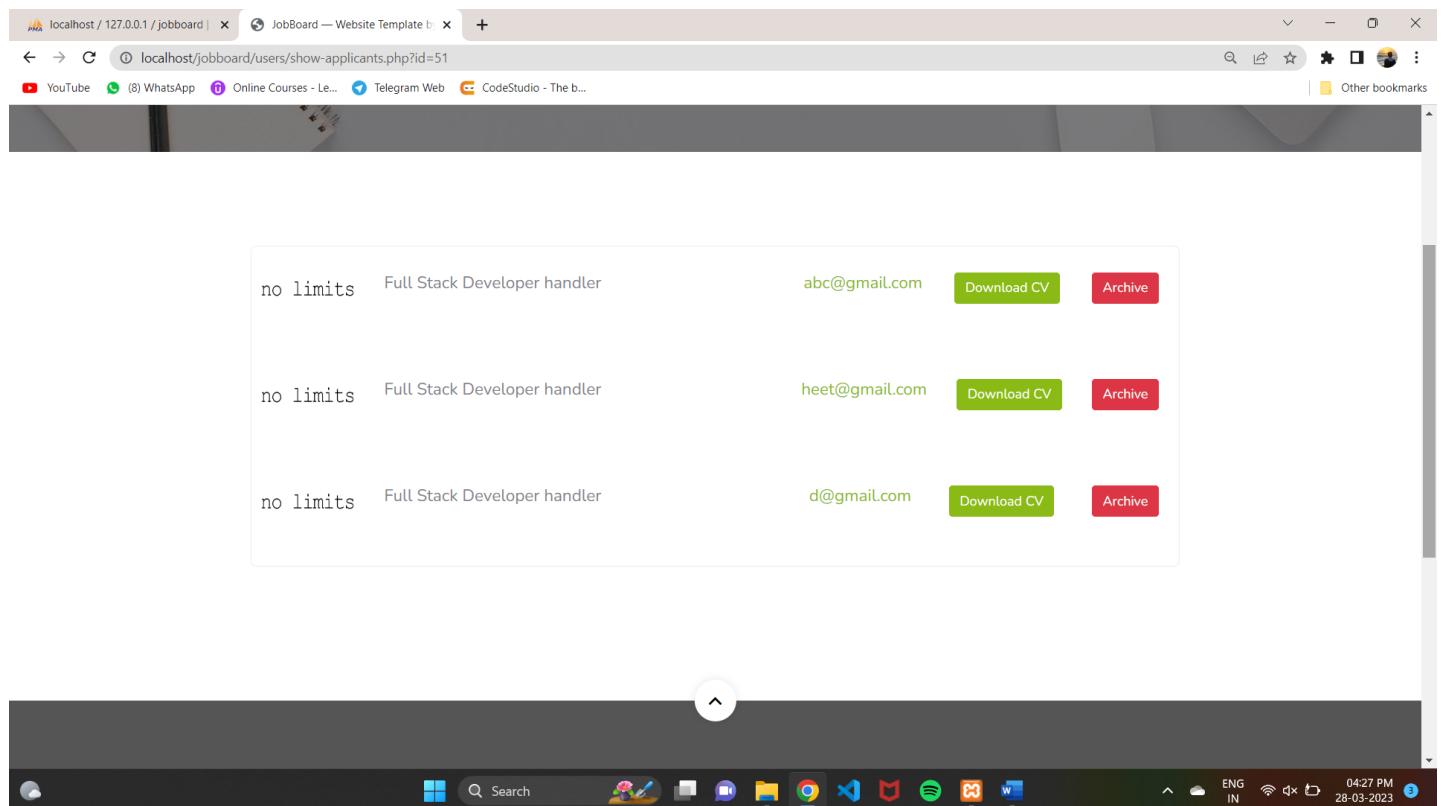
• Company Profile Update

The screenshot shows a web browser window with the URL `localhost / 127.0.0.1 / jobboard` in the address bar. The page title is "JobBoard — Website Template". The main content is titled "Update Profile" with a breadcrumb navigation: "Home" / "Update Profile". Below the title, there are input fields for "Username" (containing "c") and "Email" (containing "c@gmail.com"). There is also a "Bio" text area containing "Software Company". Below these, there are fields for "Facebook" (containing "SC//fb") and "Twitter". The browser's toolbar at the top includes links for YouTube, WhatsApp, Online Courses, Telegram Web, and CodeStudio. The taskbar at the bottom shows various application icons and the system clock at 04:26 PM on 28-03-2023.

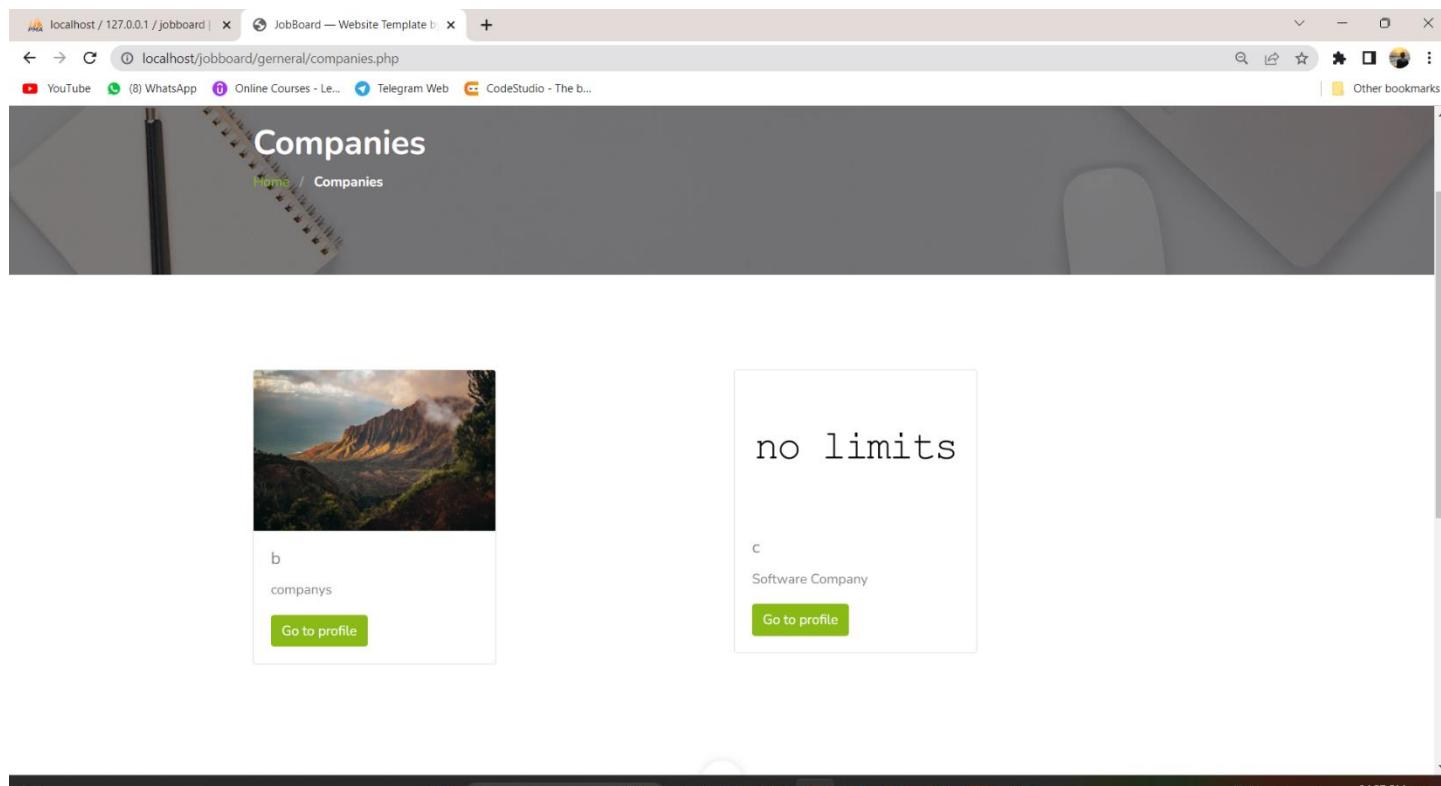
• Company Post a Job

The screenshot shows a web browser window with the URL `localhost / 127.0.0.1 / jobboard` in the address bar. The page title is "JobBoard — Website Template". The main content is titled "Post A Job". The form fields include: "Job Title" (containing "Product Designer"), "Job Region" (a dropdown menu with "Select Region"), "Job Type" (a dropdown menu with "Select Job Type"), "Vacancy" (a text input field with "e.g. 3"), "Job Category" (a dropdown menu with "Select Job Category"), "Experience" (a dropdown menu with "Select Years of Experience"), and "Salary" (a dropdown menu with "Select Salary"). The browser's toolbar at the top includes links for YouTube, WhatsApp, Online Courses, Telegram Web, and CodeStudio. The taskbar at the bottom shows various application icons and the system clock at 04:26 PM on 28-03-2023.

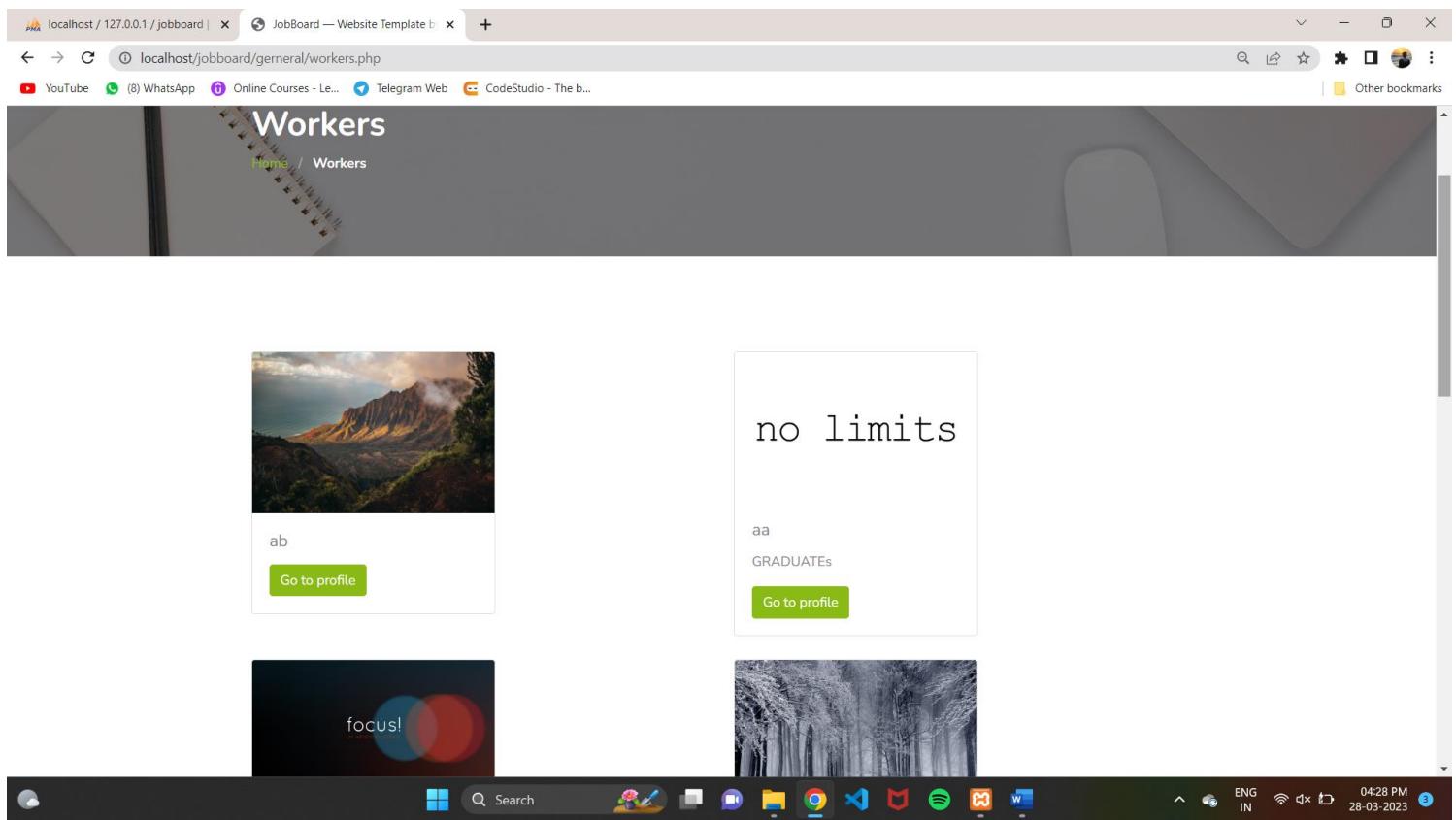
- View Job Applications



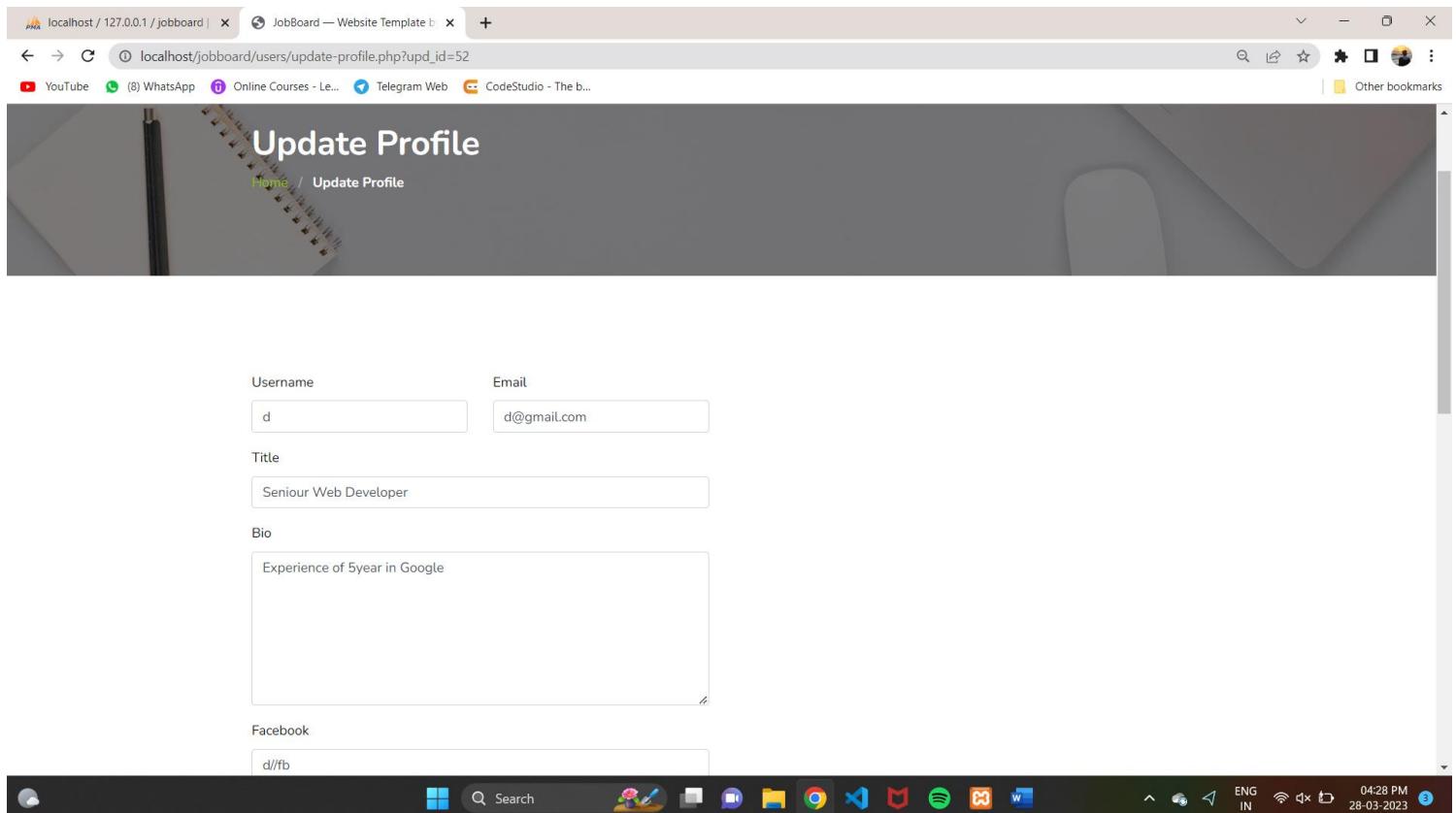
- List of companies



- Worker List



- Update workers Profile



- Saved Jobs

Saved Jobs

no limits	Full Stack Developer handler c	San Francisco	Part Time
 Free Website Template by Free-Template.co	Full Stack Developer b	New York	Part Time
no limits	Full Stack Developer c	San Francisco	Part Time

- Apply/Save Jobs

Full Stack Developer

Job Summary

- Published on: Feb 10 2023
- Vacancy: 5
- Employment Status: Part Time
- Experience: 3-6 years
- Job Location: New York
- Salary: \$70k - \$100k
- Gender: Any
- Application Deadline: Feb 20 2022
- Job Category: Design

Job Description

REACT AND NORD DEVELOPER

Responsibilities

- FULL STACK

Education + Experience

- GRADUATE

Other Benefits

- HOLIDAY ON SATURDAY

Share

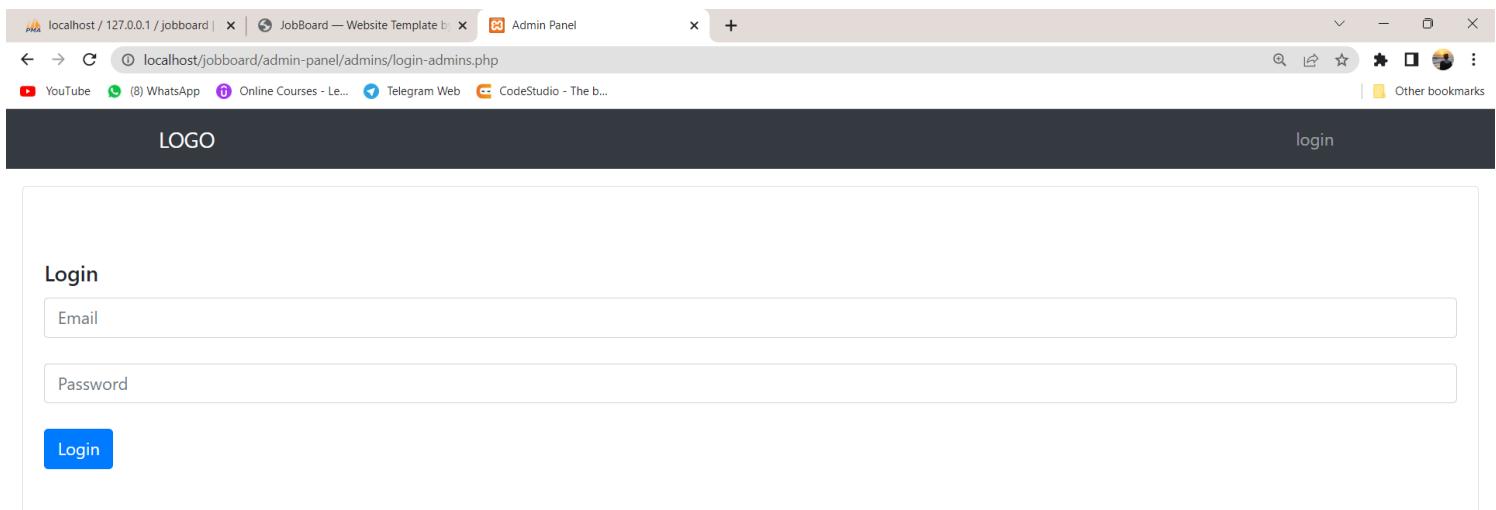
Categories

Designs
Development

Saved Job

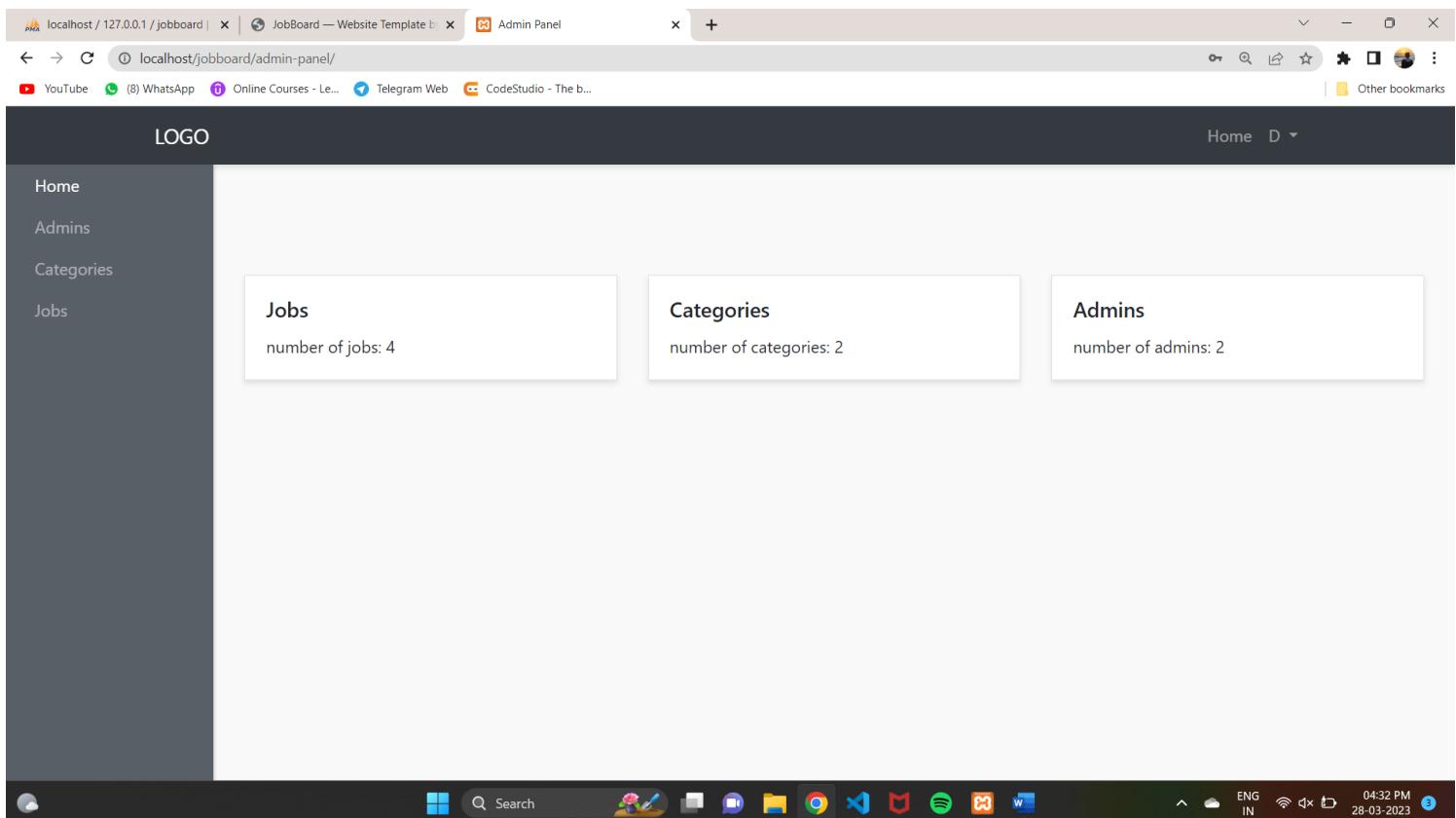
Apply Now

- Admin Login



A screenshot of a web browser showing the 'Admin Panel' login page. The URL in the address bar is `localhost/127.0.0.1/jobboard/admin-panel/admins/login-admins.php`. The page has a dark header with 'LOGO' on the left and 'login' on the right. Below the header is a form titled 'Login' with fields for 'Email' and 'Password', and a blue 'Login' button.

- Admin Dashboard



A screenshot of a web browser showing the 'Admin Panel' dashboard. The URL in the address bar is `localhost/127.0.0.1/jobboard/admin-panel/`. The page has a dark header with 'LOGO' on the left and 'Home D' on the right. On the left is a sidebar with links: Home, Admins, Categories, and Jobs. The main area contains three cards: 'Jobs' (number of jobs: 4), 'Categories' (number of categories: 2), and 'Admins' (number of admins: 2). The taskbar at the bottom shows various application icons and the date/time: 04:31 PM 28-03-2023.

- Admins details

#	Adminname	email
1	D	d@gmail.com
2	A1	a1@gmail.com

- Categories

#	name	update	delete
1	Designs	Update	Delete
3	Development	Update	Delete

- Jobs verification/delete

#	job title	category	company	status	delete
2	Full Stack Developer	design	b	unverified	<button>delete</button>
7	Full Stack Developer	design	c	unverified	<button>delete</button>
9	Full Stack Developer handler	design	c	unverified	<button>delete</button>
10	Game developer	designs	c	unverified	<button>delete</button>

- Database

Database: jobboard

Table	Action	Rows	Type	Collation	Size	Overhead
admins	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 Kib	-
categories	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 Kib	-
jobs	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_general_ci	32.0 Kib	-
job_applications	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_general_ci	16.0 Kib	-
saved_jobs	Browse Structure Search Insert Empty Drop	9	InnoDB	utf8mb4_general_ci	16.0 Kib	-
searches	Browse Structure Search Insert Empty Drop	21	InnoDB	utf8mb4_general_ci	16.0 Kib	-
users	Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	32.0 Kib	-
7 tables	Sum	50	InnoDB	utf8mb4_general_ci	144.0 Kib	0 B

5) Coding

Login Page

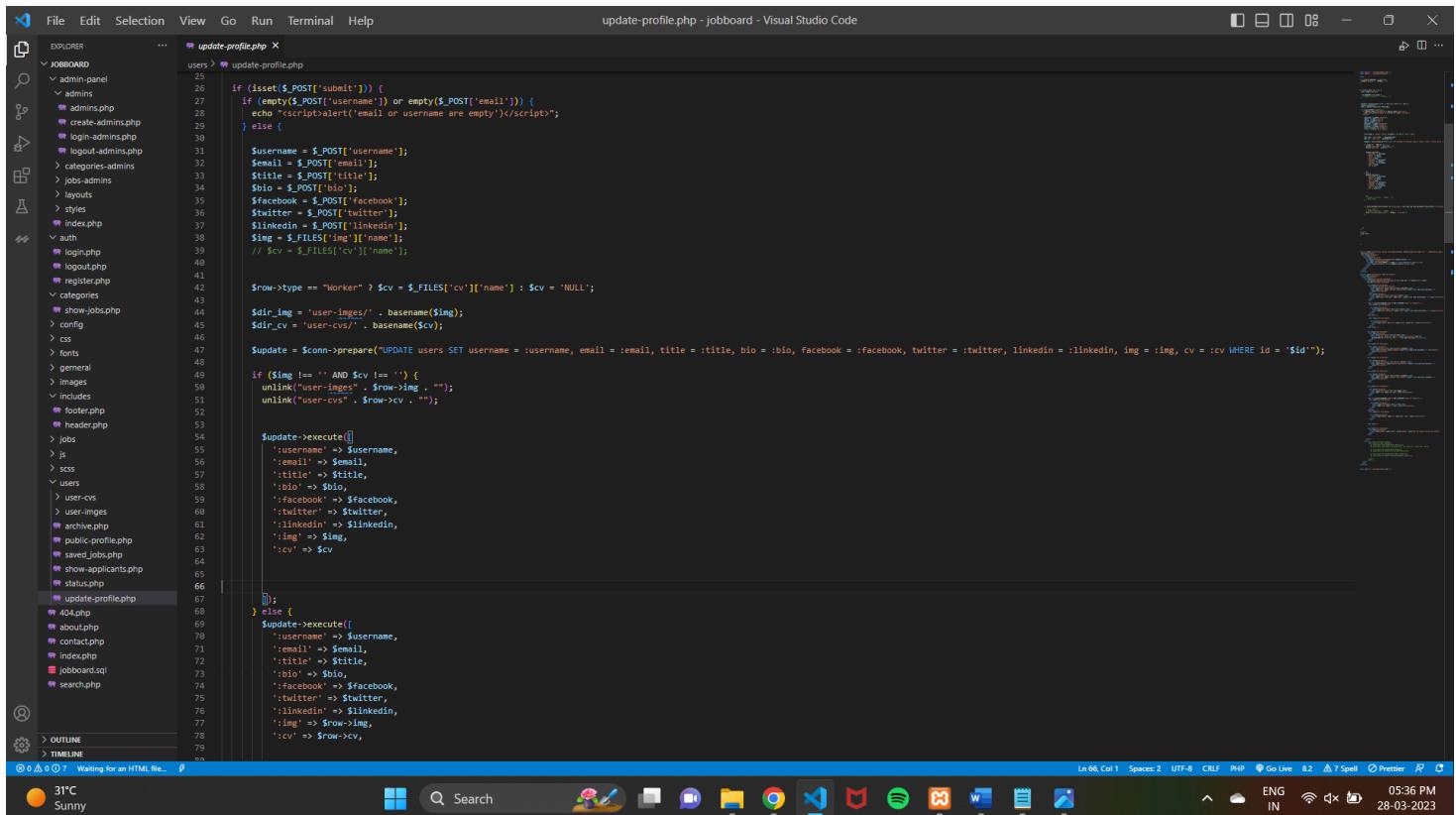
Search

A screenshot of the Visual Studio Code interface. The left sidebar shows a file tree with a folder named 'JOBBOARD' containing various PHP files like 'register.php', 'show-jobs.php', and 'search.php'. The main editor area displays the 'search.php' file, which contains PHP code for handling job search requests. The code includes database queries using PDO and header redirection logic. The status bar at the bottom shows the file is 'Waiting for an HTML file...' and provides standard VS Code status indicators.

```
search.php x
search.php

3
4 <?php
5
6 if (isset($_POST['submit'])) {
7 // echo "submitted";
8
9 if(empty($_POST['job-title']) OR empty($_POST['job-region']) OR empty($_POST['job-type'])) {
10 header("location:" . APPURL . "");
11
12 } else {
13 $job_title = $_POST['job-title'];
14 $job_region = $_POST['job-region'];
15 $job_type = $_POST['job-type'];
16
17 //doing trendy keywords
18 $insert = $conn->prepare("INSERT INTO searches (keyword) VALUES (:keyword)");
19 $insert->execute([
20 'keyword' => $job_title
21 ]);
22
23 $search = $conn->query("SELECT * FROM jobs WHERE job_title LIKE '%$job_title%' AND job_region LIKE '%$job_region%' AND job_type LIKE '%");
24 $search->execute();
25 $searchRes = $search->fetchAll(PDO::FETCH_OBJ);
26
27 // echo "<p>";
28 // print_r($searchRes);
29 // echo "</p>";
30
31 }
32 } else {
33 header("location:" . APPURL . "");
34 }
35
36 ?>
37
38
39 <section class="section-hero overlay inner-page bg-image"
```

Update-Profile

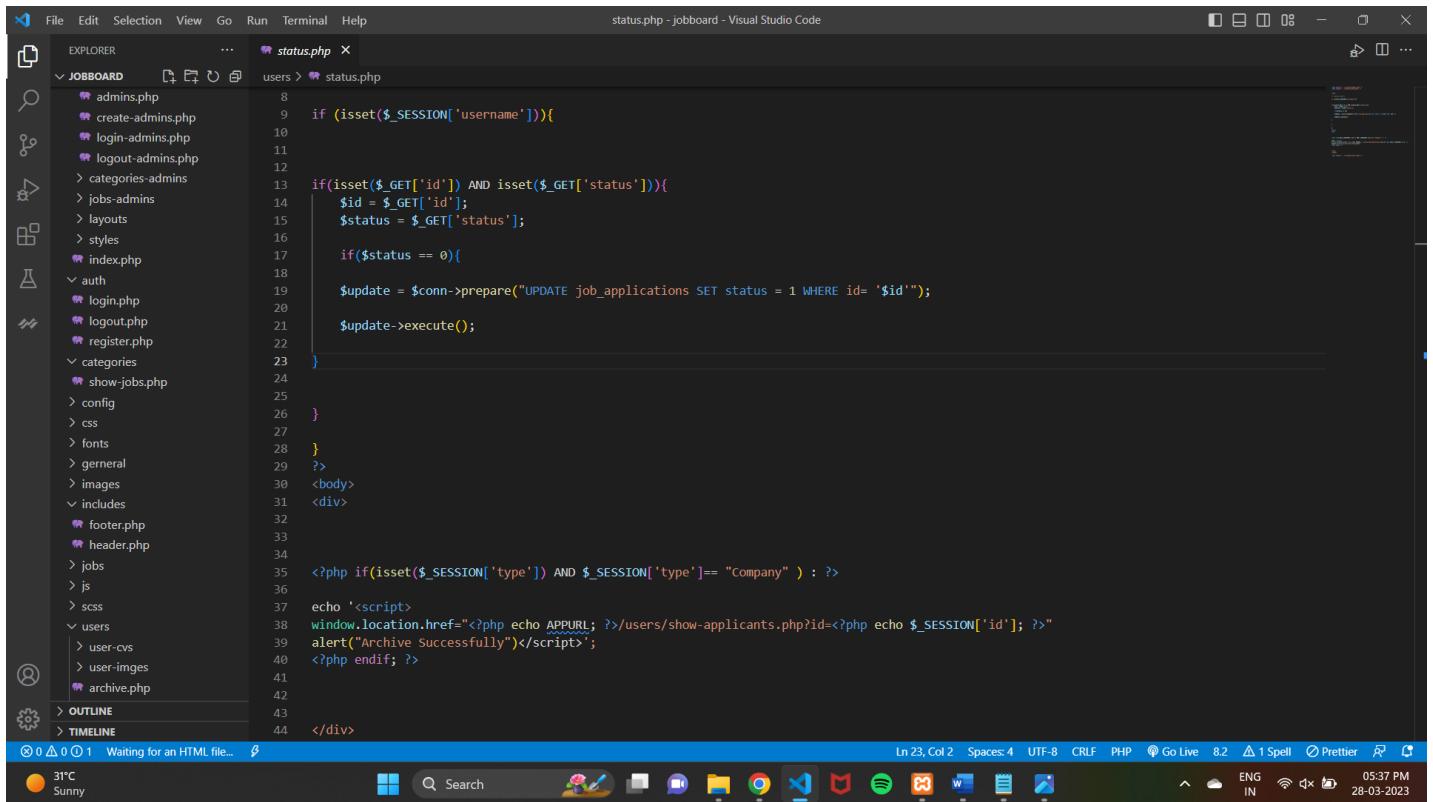


```
File Edit Selection View Go Run Terminal Help
update-profile.php - jobboard - Visual Studio Code
EXPLORER
JOBOARD
users > update-profile.php
25
26 if (isset($_POST['submit'])) {
27     if (empty($_POST['username']) or empty($_POST['email'])) {
28         echo "<script>alert('email or username are empty')</script>";
29     } else {
30
31         $username = $_POST['username'];
32         $email = $_POST['email'];
33         $title = $_POST['title'];
34         $bio = $_POST['bio'];
35         $facebook = $_POST['facebook'];
36         $twitter = $_POST['twitter'];
37         $linkedin = $_POST['linkedin'];
38         $img = $_FILES['cv']['name'];
39         // $cv = $_FILES['cv']['name'];
40
41         $row->type == "Worker" ? $cv = $_FILES['cv']['name'] : $cv = 'NULL';
42
43         $dir_img = 'user-images/' . basename($img);
44         $dir_cv = 'user-cv/' . basename($cv);
45
46         $update = $conn->prepare("UPDATE users SET username = :username, email = :email, title = :title, bio = :bio, facebook = :facebook, twitter = :twitter, linkedin = :linkedin, img = :img, cv = :cv WHERE id = '$id'");
47
48         if ($img !== '' AND $cv !== '') {
49             unlink("user-images/" . $row->img);
50             unlink("user-cv/" . $row->cv);
51
52             $update->execute([
53                 'username' => $username,
54                 'email' => $email,
55                 'title' => $title,
56                 'bio' => $bio,
57                 'facebook' => $facebook,
58                 'twitter' => $twitter,
59                 'linkedin' => $linkedin,
60                 'img' => $img,
61                 'cv' => $cv
62             ]);
63
64         } else {
65             $update->execute([
66                 'username' => $username,
67                 'email' => $email,
68                 'title' => $title,
69                 'bio' => $bio,
70                 'facebook' => $facebook,
71                 'twitter' => $twitter,
72                 'linkedin' => $linkedin,
73                 'img' => $row->img,
74                 'cv' => $row->cv,
75             ]);
76
77     }
78
79 } else {
80     $update->execute([
81         'username' => $username,
82         'email' => $email,
83         'title' => $title,
84         'bio' => $bio,
85         'facebook' => $facebook,
86         'twitter' => $twitter,
87         'linkedin' => $linkedin,
88         'img' => $row->img,
89         'cv' => $row->cv,
90     ]);
91
92 }
```

Ln 66, Col 1 Spaces: 2 UTF-8 CRLF PHP Go Live 8.2 ⚡ 7 Spell ⚡ Prettier

31°C Sunny 05:36 PM 28-03-2023

Status



```
File Edit Selection View Go Run Terminal Help
status.php - jobboard - Visual Studio Code
EXPLORER
JOBOARD
users > status.php
8
9 if (isset($_SESSION['username'])){
10
11
12     if(isset($_GET['id']) AND isset($_GET['status'])){
13         $id = $_GET['id'];
14         $status = $_GET['status'];
15
16         if($status == 0){
17
18             $update = $conn->prepare("UPDATE job_applications SET status = 1 WHERE id= '$id'");
19
20             $update->execute();
21
22     }
23
24
25     }
26
27     ?>
28     <body>
29     <div>
30
31         <?php if(isset($_SESSION['type']) AND $_SESSION['type']=='Company' ) : ?>
32
33         echo '<script>';
34         window.location.href=<?php echo APPURL; ?>/users/show-applicants.php?id=<?php echo $_SESSION['id']; ?>;
35         alert("Archive Successfully")</script>';
36         <?php endif; ?>
37
38
39
40
41
42
43     </div>
44
45 }
```

Ln 23, Col 2 Spaces: 4 UTF-8 CRLF PHP Go Live 8.2 ⚡ 1 Spell ⚡ Prettier

31°C Sunny 05:37 PM 28-03-2023

Show-applications

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** show-applicants.php - jobboard - Visual Studio Code
- Explorer:** Shows the 'JOBBOARD' folder containing files like jobs, js, scss, users, user-cvs, user-imgs, and various image files (e.g., IMG_3590.jpg, IMG_3637.MOV). It also lists PHP files such as archive.php, public-profile.php, saved.jobs.php, show-applicants.php, status.php, update-profile.php, 404.php, about.php, contact.php, and index.php.
- Editor:** The 'show-applicants.php' file is open, displaying PHP code. The code checks if the user is a company and retrieves job applications for a specific user ID if provided via GET. It also handles the 404 error case.
- Bottom Status Bar:** Shows the file is 'Waiting for an HTML file...' at line 80, column 41. Other status indicators include 'Spaces: 2', 'UTF-8', 'CRLF', 'PHP', 'Go Live', '8.2', '10 Spell', 'Prettier', and system information like '31°C Sunny' and the date '28-03-2023'.

Save-jobs

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** saved_jobs.php - jobboard - Visual Studio Code
- Explorer:** Shows the 'JOBBOARD' folder containing files like jobs, js, scss, users, user-cvs, user-imgs, and various image files (e.g., IMG_3590.jpg, IMG_3637.MOV). It also lists PHP files such as archive.php, public-profile.php, saved.jobs.php, show-applicants.php, status.php, update-profile.php, 404.php, about.php, contact.php, and index.php.
- Editor:** The 'saved_jobs.php' file is open, displaying PHP code. This script checks if the user is a worker and retrieves saved jobs for a specific user ID if provided via GET. It also handles the 404 error case.
- Bottom Status Bar:** Shows the file is 'Waiting for an HTML file...' at line 1, column 1. Other status indicators include 'Spaces: 2', 'UTF-8', 'CRLF', 'PHP', 'Go Live', '8.2', '7 Spell', 'Prettier', and system information like '31°C Sunny' and the date '28-03-2023'.

Archive

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** archive.php - jobboard - Visual Studio Code
- Explorer:** Shows the project structure under "JOBBOARD". The "archive.php" file is selected.
- Code Editor:** Displays the PHP code for "archive.php". The code handles session checks, header redirection, and database queries to fetch user and job application data.
- Bottom Status Bar:** Shows "Ln 82, Col 11" and other system information like weather (31°C, Sunny) and date (28-03-2023).

```
<?php require "../includes/header.php"; ?>
<?php require "../config/config.php"; ?>

if (!isset($_SESSION['type']) and ($_SESSION['type'] != "Company")) {
    header("location: " . APPURL . "");
}

if (isset($_GET['id'])) {
    $id = $_GET['id'];

    // if($_SESSION['id']==$id) {
    //     echo("<script>location.href = '".APPURL."'</script>");
    // }

    $select = $conn->query("SELECT * FROM users WHERE id = '$id'");
    $select->execute();
    $profile = $select->fetch(PDO::FETCH_OBJ);

    $getApplicants = $conn->query("SELECT * FROM job_applications WHERE company_id = '$id' AND status = 1 ORDER BY id desc ");
    $getApplicants->execute();

    $getApplicants = $getApplicants->fetchAll(PDO::FETCH_OBJ);

} else{
    echo "404";
}

?>
```

Login-admins

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** login-admins.php - jobboard - Visual Studio Code
- Explorer:** Shows the project structure under "admin-panel". The "login-admins.php" file is selected.
- Code Editor:** Displays the PHP code for "login-admins.php". It handles session checks, POST requests, and database queries to log in an admin.
- Bottom Status Bar:** Shows "Ln 18, Col 4" and other system information like weather (31°C, Sunny) and date (28-03-2023).

```
<?php

if (isset($_SESSION['adminname'])) {
    header("location: " . ADMINURL . "");
}

if (isset($_POST['submit'])) {

    if (empty($_POST['email']) or empty($_POST['password'])) {
        echo "<div class='alert alert-danger text-white'> some input here </div>";
    } else {

        $email = $_POST['email'];
        $password = $_POST['password'];

        $login = $conn->query("SELECT * FROM admins WHERE email = '$email'");
        $login->execute();

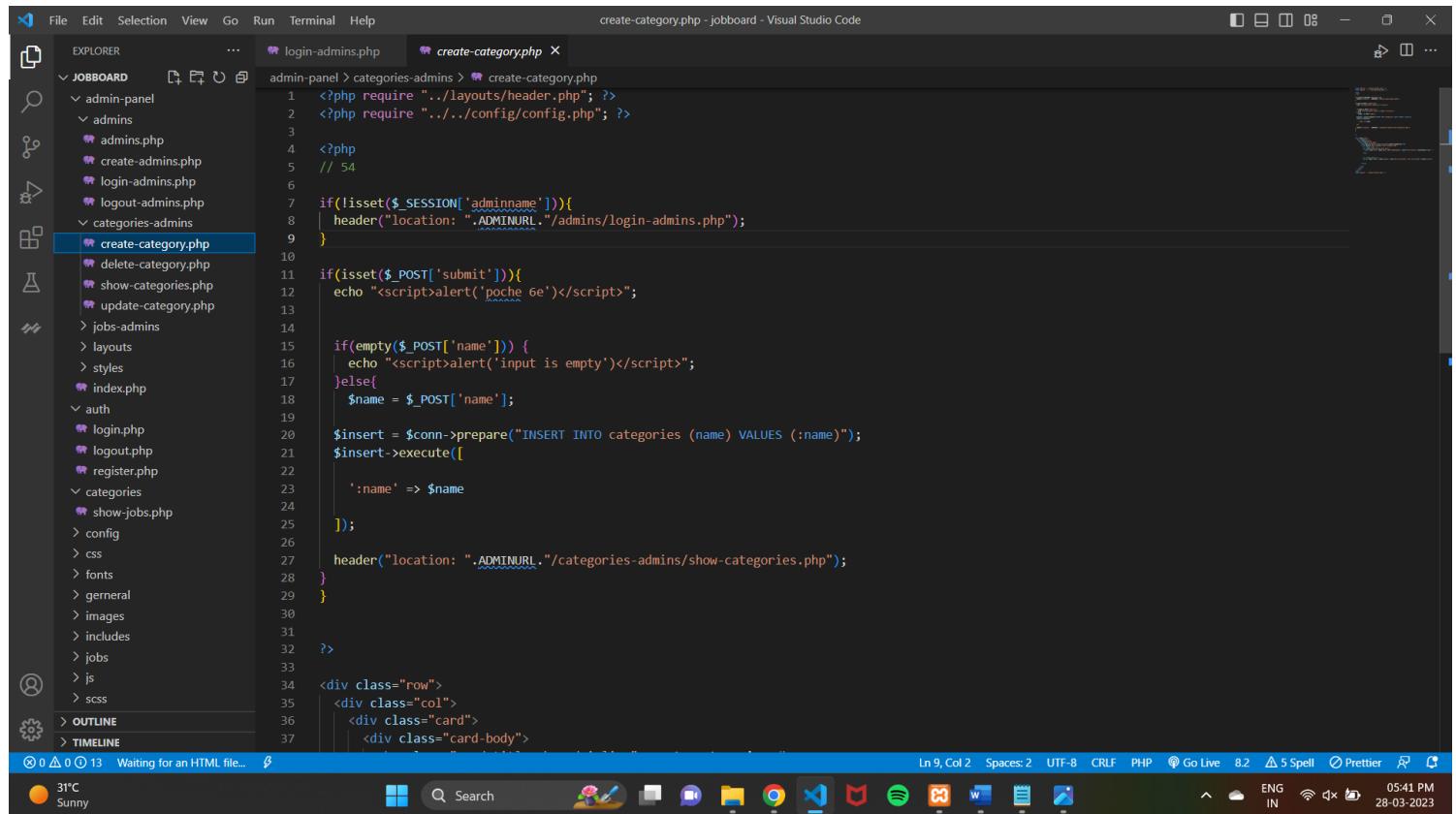
        $select = $login->fetch(PDO::FETCH_ASSOC);

        if ($login->rowCount() > 0) {
            if (password_verify($password, $select['password'])) {

                $_SESSION['adminname'] = $select['adminname'];
                $_SESSION['email'] = $select['email'];

                header("location: " . ADMINURL . "");
                // echo "<script>alert('login')</script>";
            }
        }
    }
}
```

Create-category



```
<?php require "../layouts/header.php"; ?>
<?php require "../../config/config.php"; ?>

<?php
// 54

if(!isset($_SESSION['adminname'])){
    header("location: ".ADMINURL."/admins/login-admins.php");
}

if(isset($_POST['submit'])){
    echo "<script>alert('poche 6e')</script>";

    if(empty($_POST['name'])){
        echo "<script>alert('input is empty')</script>";
    }else{
        $name = $_POST['name'];

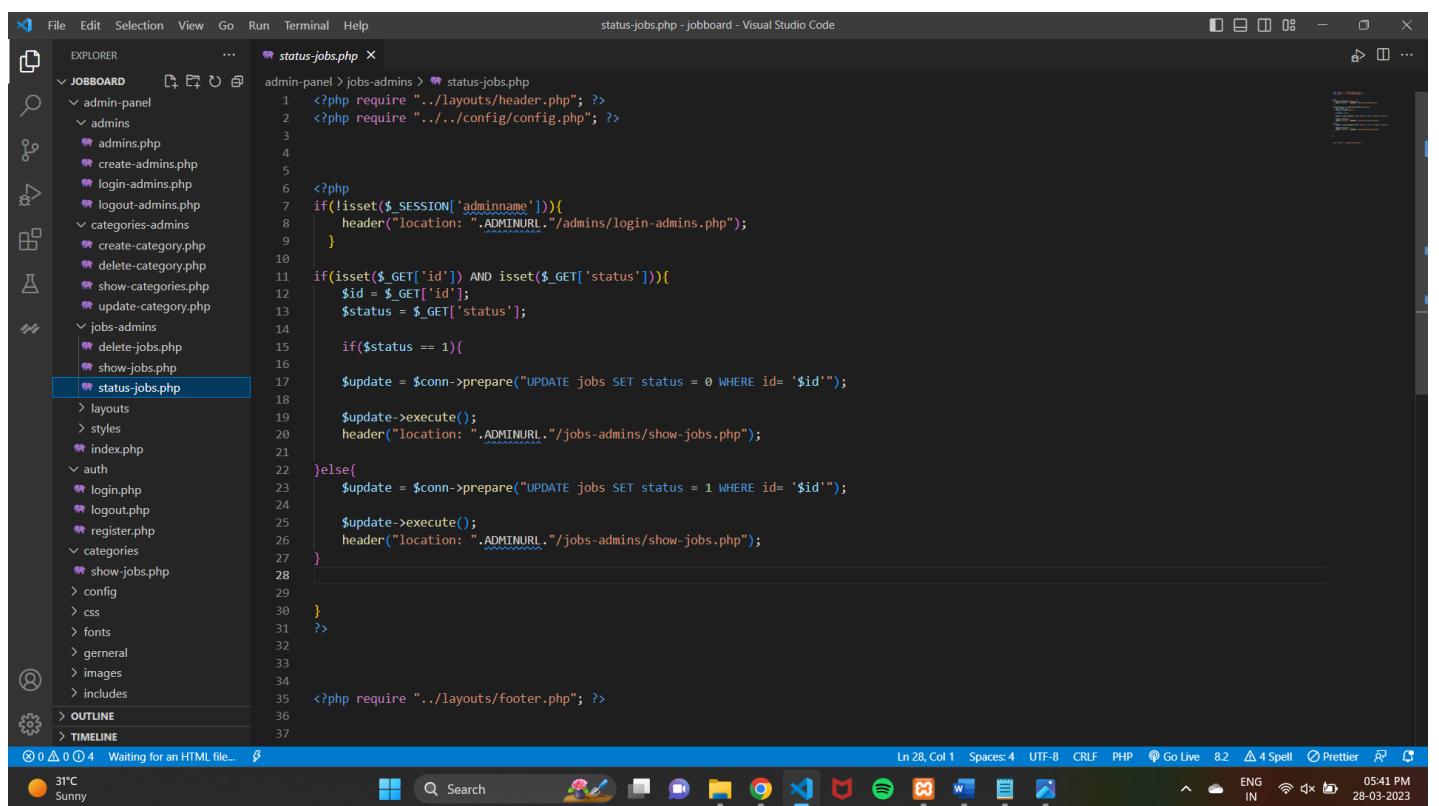
        $insert = $conn->prepare("INSERT INTO categories (name) VALUES (:name)");
        $insert->execute([
            ':name' => $name
        ]);

        header("location: ".ADMINURL."/categories-admins/show-categories.php");
    }
}

header("location: ".ADMINURL."/categories-admins/show-categories.php");
?>

<div class="row">
    <div class="col">
        <div class="card">
            <div class="card-body">
```

Status-jobs



```
<?php require "../layouts/header.php"; ?>
<?php require "../../config/config.php"; ?>

<?php
if(!isset($_SESSION['adminname'])){
    header("location: ".ADMINURL."/admins/login-admins.php");
}

if(isset($_GET['id']) AND isset($_GET['status'])){
    $id = $_GET['id'];
    $status = $_GET['status'];

    if($status == 1){
        $update = $conn->prepare("UPDATE jobs SET status = 0 WHERE id= '$id'");

        $update->execute();
        header("location: ".ADMINURL."/jobs-admins/show-jobs.php");
    }else{
        $update = $conn->prepare("UPDATE jobs SET status = 1 WHERE id= '$id'");

        $update->execute();
        header("location: ".ADMINURL."/jobs-admins/show-jobs.php");
    }
}

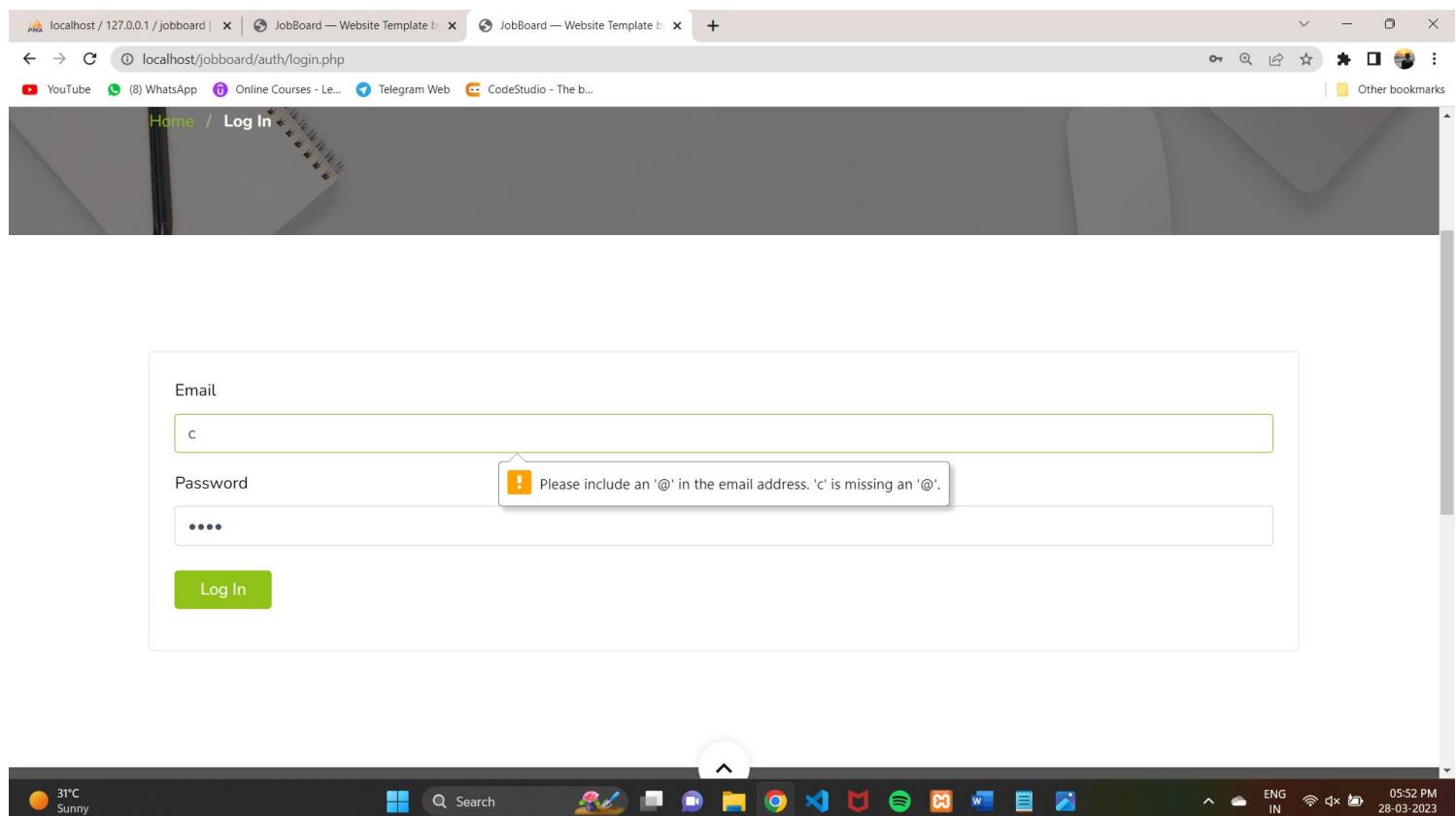
header("location: ".ADMINURL."/jobs-admins/show-jobs.php");
?>

<?php require "../layouts/footer.php"; ?>
```

6) Testing

❖ Login Page:

No.	Input	Expected Result	Actual Result	Result
1	Empty user name	error	error	Login fail
2	Empty password	Error	error	Login Fail



❖ Registration :

No.	Input	Expected Result	Actual Result	Result
1	Empty full name	Error	error	Registration fail
2	Empty email	Error	Error	Registration fail
3	Empty type	Error	error	Registration fail
4	Empty password	Error	Error	Registration fail

The screenshot shows a registration form on a website. The fields are as follows:

- Username:** A text input field containing "Username".
- Email:** A text input field that is empty and has a red border, indicating it is required. An error message box above the field says "Please fill out this field."
- User Type:** A dropdown menu labeled "Select User Type".
- Password:** A text input field containing "Password".
- Re-Type Password:** A text input field containing "Re-enter Password".

At the bottom of the form is a green "Signup" button.



7) New Tools/ Technologies learned/used

- My SQL Database (PHP My Admin)
- HTML
- CSS

8) System Limitations/Restrictions and Dependencies/Constraints

❖ System Limitation / Constraints:

- Admin can't remove another admins
- Company can't send direct mail to the job applicants from the website
- Security Issues.
- Applicant cant get any notification from company side via portal.

9) Future Enhancement & Opportunities

- More efficient user experience.
- Add mail function for company that he directly contact with job applicant.
- Add calendar for scheduling the job interview by company.
- Add forget password module.

10) Bibliography & References

➤ Guidelines Of Shweta Bhatia Madam.

❖ Websites:

➤ www.geeksforgeeks.com

➤ <https://stackoverflow.com/>

➤ <https://chat.openai.com/>