**“JOB PORTAL SYSTEM”**

**SUBMITTED BY:**

**Amish Dangal**

**T.U. Registration No. (7-2-1073-02-2021)**

Himalaya Darshan College

*A Project Report Submitted to*

**Faculty of Management, Tribhuvan University**

in partial fulfillment of the requirements for the degree of

**Bachelor of Information System (BIM)**

Biratnagar

May, 2025



# STUDENT DECLARATION

I respectfully affirm that I have completed the Project titled **"JOB PORTAL SYSTEM"** under the esteemed guidance of **"Mr. Jivan Niroula"**, Himalaya Darshan College, Biratnagar-9. This project has been undertaken in partial fulfillment of the requirements for the **Bachelor of Information Management** degree at Faculty of Management, Tribhuvan University. I assure that this work is original and has not been submitted previously elsewhere.

Date: May, 2025 …………………………

Signature

Name: Amish Dangal

# CERTIFICATE FROM THE SUPERVISOR

This is to certify that the summer project titled **"JOB PORTAL SYSTEM"** is an academic work undertaken by “Amish Dangal” in partial fulfillment of the requirements for the degree of Bachelor of Information Management at the Faculty of Management, Tribhuvan University, under my guidance and supervision. To the best of my knowledge, the content presented by him in the summer project report has not been previously submitted elsewhere.

……………………………………

Signature of the Supervisor

Name: Jivan Niroula

Designation: Lecturer, Himalaya Darshan College

Date: 2025/06/03

Biratnagar-9, Morang

# APPROVAL SHEET

This is to certify that the project prepared by **Amish Dangal**, entitled "**JOB PORTAL SYSTEM**" has been thoroughly reviewed and evaluated as part of the partial fulfillment of the requirements for the degree of **Bachelor of Information Management**. The project has been found to meet the academic standards and expectations of the department, and after careful consideration, it has been approved for further development and presentation.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Name** | **Designation** | **Signature** |
| **1** | **Mr. Jivan Niroula** | **Project Supervisor** |  |
| **2** | **Er. Sumit Kumar Shah** | **Program Coordinator** |  |
| **3** | **Mr. Kushal Niroula** | **Internal Examiner** |  |
| **4** | **Mr. Harendra Subedi** | **External Examiner** |  |

Date of Defense: 2025/06/03

Department: Department of IT

Faculty: Faculty of Management

# ABSTRACT

The **Job Portal System** is a simple and user-friendly website that helps job seekers find jobs and lets employers post job openings. Job seekers can make a profile, upload their resume, and apply for jobs online. Employers can add job posts, check applications, and contact suitable candidates directly.

The system also includes an admin panel to manage users, job listings, and the whole platform. This makes the job search and hiring process easier and more organized for everyone.

The system is built using Laravel for the backend, and HTML, CSS, and JavaScript for the frontend. MySQL is used as the database to store all user and job-related data. Currently, the platform is designed mainly for desktop use and does not support mobile devices.

Overall, this project helps people find jobs and helps companies hire the right candidates in a faster and more convenient way.

**Keywords**

*Laravel, Job portal, Job search, Resume upload, Employer login, HTML, CSS, JavaScript, MySQL.*

# ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to all the individuals who played a pivotal role in the successful completion of this summer project report, fulfilling the requirements for the Bachelor of Information Management (BIM) degree under the Faculty of Management, Tribhuvan University. I am extremely thankful to the University for providing us with such a meaningful and impactful opportunity.

I would also like to express my deepest appreciation to **Himalaya Darshan** **College** for fostering an environment conducive to the successful completion of this project. The support I received from the faculty and staff significantly contributed to the development of our work.

My sincere thanks go to our supervisor, **Mr. Jivan Niroula**, whose unwavering guidance, encouragement, and dedication were crucial in shaping the success of my Job Portal System. His insights, expertise, and constructive feedback were invaluable throughout this journey

My deepest appreciation goes to my project partner, **Rabin Kumar Chaudhary** for his outstanding contribution, particularly in handling the backend development of the Job Portal System. He took the lead in tasks such as requirements gathering, system design, backend coding and implementation, deployment, and final documentation. His strong grasp of server-side logic, database integration, and system functionality ensured the smooth operation of our platform. Rabin’s dedication, problem-solving ability, and technical leadership were critical to the success and stability of the entire project.

I also extend my gratitude to the other teachers and faculty members at the college, whose support, encouragement, and academic input helped refine and enhance the quality of our project. Their knowledge and willingness to assist were pivotal in my progress.

In conclusion, I am grateful to everyone who contributed to the success of this project in any capacity. The collaboration, guidance, and encouragement I received have made this experience immensely rewarding. I feel fortunate to have had the opportunity to learn and grow in such a supportive environment.

Sincerely,

Amish Dangal (7-2-1073-02-2021)

BIM 6th Semester

# TABLE OF CONTENTS

[*STUDENT DECLARATION ii*](#_Toc200371252)

[*CERTIFICATE FROM THE SUPERVISOR iii*](#_Toc200371253)

[*APPROVAL SHEET iv*](#_Toc200371254)

[*ABSTRACT v*](#_Toc200371255)

[*ACKNOWLEDGEMENT vi*](#_Toc200371256)

[*TABLE OF CONTENTS vii*](#_Toc200371257)

[*LIST OF FIGURES viii*](#_Toc200371258)

[*LIST OF TABLES ix*](#_Toc200371259)

[*LIST OF ABBREVIATIONS x*](#_Toc200371260)

[CHAPTER 1 - INTRODUCTION 1](#_Toc200371261)

[1.1 Background 1](#_Toc200371262)

[1.2 Problem Statement 1](#_Toc200371263)

[1.3 Objectives 1](#_Toc200371264)

[1.4 Review of related works and Literature 1](#_Toc200371265)

[1.5 Development Methodology 2](#_Toc200371266)

[1.6 Scope and Limitations 3](#_Toc200371267)

[1.6.1 Scopes 3](#_Toc200371268)

[1.6.2 Limitations 3](#_Toc200371269)

[1.7 Report Organization 3](#_Toc200371270)

[CHAPTER 2 – SYSTEM DEVELOPMENT PROCESS 4](#_Toc200371271)

[2.1 Analysis of tasks and activity 4](#_Toc200371272)

[2.1.1 Requirement Analysis 4](#_Toc200371273)

[2.1.2 Feasibility Study 8](#_Toc200371274)

[2.1.3 Object Oriented modeling 9](#_Toc200371275)

[2.2 System design 13](#_Toc200371276)

[2.2.1 User Interface 13](#_Toc200371277)

[2.2.2 Database design/Object oriented design models 14](#_Toc200371278)

[2.3 Implementations 15](#_Toc200371279)

[2.3.1 Tools and Technologies Used 15](#_Toc200371280)

[2.3.2 Module Description 16](#_Toc200371281)

[2.3.3 Testing 17](#_Toc200371282)

[CHAPTER 3 - DICUSSIONS AND CONCLUSIONS 21](#_Toc200371283)

[3.1 Summary 21](#_Toc200371284)

[3.2 Conclusions 21](#_Toc200371285)

[3.2 Recommendations 21](#_Toc200371286)

[REFERENCES a](#_Toc200371287)

[APPENDIX i](#_Toc200371288)

# LIST OF FIGURES

**Figure No. Title Page no.**

[Figure1.1: Iterative Waterfall Model 2](#_Toc200371207)

[Figure 2.1: Use Case Diagram of Job Portal System 5](#_Toc200371208)

[Figure 2.2: Gantt Chart 8](#_Toc200371209)

[Figure 2.3: Class Diagram of Job Portal System 9](#_Toc200371210)

[Figure 2.4: Object Diagram of Job Portal System 10](#_Toc200371211)

[Figure 2.5: Activity Diagram of Job Portal System 11](#_Toc200371212)

[Figure 2.6: Sequence Diagram of Job Portal System 12](#_Toc200371213)

[Figure 2.7: MVC Architecture 14](#_Toc200371214)

# LIST OF TABLES

**Table No.Title Page no.**

[Table 2.1: Use case description (User Registration) 6](#_Toc200371193)

[Table 2.2: Use case description (Login) 6](#_Toc200371194)

[Table 2.3: Use case description (Post Jobs) 6](#_Toc200371195)

[Table 2.4: Use case description (Job Apply) 7](#_Toc200371196)

[Table 2.5: Use case (Edit Profile) 7](#_Toc200371197)

[Table 2.6: Tools Used 16](#_Toc200371198)

[Table 2.7: User Authentication Module 16](#_Toc200371199)

[Table 2.8: Admin Dashboard Module 16](#_Toc200371200)

[Table 2.9: Job Seeker Dashboard Module 16](#_Toc200371201)

[Table 2.10: Employer dashboard module 17](#_Toc200371202)

[Table 2.11: Test case 1 - Registration form 17](#_Toc200371203)

[Table 2.12: Test case 2: Login form 18](#_Toc200371204)

[Table 2.13: Test case 3: Application form 18](#_Toc200371205)

[Table 2.14: Test case 1: Job Application process 19](#_Toc200371206)

# 

# LIST OF ABBREVIATIONS

**Abbreviations Full Form**

CSS Cascading Style Sheet

HTML Hypertext Markup Language

MySQL My Structured Query Language

UML Unified Modeling Language

US Use case

VS Visual Studio

# CHAPTER 1 - ****INTRODUCTION****

## ****1.1 Background****

The Job Portal System is designed as a web-based platform aimed at simplifying the process of job searching and recruitment. It serves both job seekers and employers by providing an accessible space where companies can post job openings and individuals can apply online. This system is particularly beneficial for job seekers who are actively looking for opportunities, as well as for employers who need to find suitable candidates efficiently.

Through the platform, users can search for jobs based on criteria such as skills, experience, and interests. At the same time, employers can review applications and select candidates who closely match their requirements. Features like job search filters, resume uploads, and direct messaging between employers and applicants contribute to streamlining the overall hiring process, making it faster and more effective for all parties involved.

## 1.2 Problem Statement

Finding a good job or the right candidate can be slow and difficult. Employers often spend a lot of time checking resumes one by one, which makes hiring take longer. At the same time, job seekers usually have limited ways to find jobs, like newspapers or asking around, which means they may miss out on better opportunities. This can be frustrating for both sides.

The Job Portal System helps solve these problems by giving both job seekers and employers a single place to connect. It makes it easier to search for jobs, apply online, and find the right match quickly.

## 1.3 Objectives

1. To help job seekers quickly find and apply for jobs
2. To allow employers to easily post jobs, review applications, and connect with right candidates.

## 1.4 Review of related works and Literature

In today’s digital world, job portals play a key role in connecting job seekers and employers. These platforms allow users to easily search for jobs, apply, and manage applications from the comfort of their homes. Our job portal system, like others, aims to make job hunting more efficient and accessible, but it also offers a unique feature: automatic resume generation. This feature allows users who are unsure about creating a resume to generate one automatically by entering key details, making the process easier for those who may not know how to create a professional resume. The following platforms have helped shape the concept of online job portals in Nepal.

**MeroJob**: MeroJob is one of the most widely used job portals in Nepal. It allows job seekers to create profiles, upload resumes, and apply for jobs directly. It also allows employers to post jobs and view applications. The platform has a large user base and is trusted for professional job hunting.

**HamroJobs**: HamroJobs is a simple and easy-to-use platform for job seekers in Nepal. It helps job seekers apply for jobs quickly and easily. Employers can post jobs with minimal effort. It is especially good for freshers looking for their first job or students searching for internships.

**KTMJobs**: KTMJobs focuses on job listings in the Kathmandu Valley. It is a platform that helps small businesses and local companies post job vacancies. The portal is user-friendly, making it easy for job seekers to find and apply for local jobs.

## 1.5 Development Methodology

The development process for our job portal system will follow an iterative methodology, starting with requirement gathering, where we’ll collect the necessary details about the features the system should have, such as job seekers being able to search and apply for jobs easily, and employers being able to post jobs and manage applications. After this, we’ll conduct a feasibility study to make sure the system can be built with the available resources. Then we move to system design, where we plan how the portal will look and function to make it user-friendly and easy to navigate.

Once the design is ready, we’ll proceed to the implementation phase, where we’ll start building the system, adding features like job search, application submission, and user profiles. Following that, we’ll test the system to ensure that it works correctly, fixing any bugs and making sure all features run smoothly.

This process will be repeated in iterations, where each cycle focuses on improving different parts of the system or adding new features. This approach ensures that the system is continuously improving and adapting to user feedback. It also helps identify and fix any issues early, reducing risks and improving the overall quality of the system. By using this method, we can ensure that the job portal will be reliable, functional, and meet the needs of job seekers and employers.

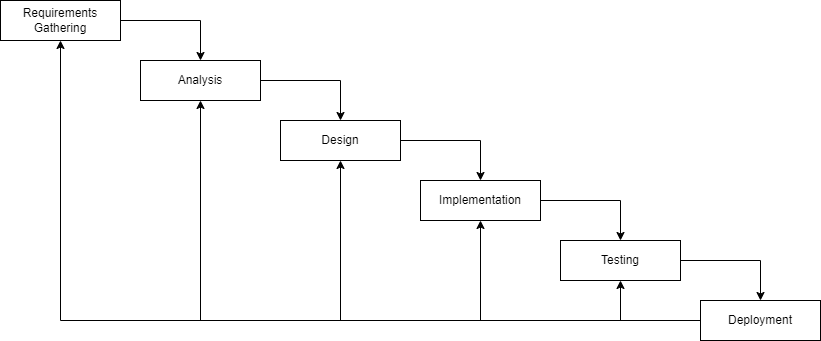


Figure1.1: Iterative Waterfall Model

## 1.6 Scope and Limitations

### 1.6.1 Scopes

The Job Portal System is designed to simplify the hiring process for both job seekers and employers. The platform helps job seekers find jobs based on their skills and location, giving them access to more opportunities that were previously available only through newspapers, word of mouth, or physical job boards. Employers can easily post and update job listings, ensuring job seekers have access to the most current job openings. The system is aimed at making the job search and hiring process quicker, easier, and more accessible.

Job seekers can search for and apply to jobs online, anytime and anywhere, using the website. This platform provides the latest job listings, ensuring job seekers always see the most up-to-date openings.

Employers can quickly post and manage job openings, ensuring their listings are always visible to potential candidates. Employers can review applications and track job seekers, helping them find the right candidates more efficiently.

### 1.6.2 Limitations

* **Internet Access:** Customers need access to the internet to make bookings, which could be a challenge in remote areas or for people with limited internet connectivity.
* **Language Barriers**: System is not available in multiple language. It may exclude non-native speakers or those who are not comfortable with the language used on this platform.
* **Limited Search Filters:** The system only allows job seekers to search using basic filters like skills and location, without more advanced search options.
* **No AI for Job Matching**: The platform does not use AI for automatic job matching or resume screening, so employers must manually review applications.

## 1.7 Report Organization

The report consists of three chapters which will covers all the introductions, analysis, design, testing and development applications.

Chapter-1 covers the background, introduction, problem statements, objectives, literature review, development methodology, scope and limitations.

Similarly, chapter-2 covers Analysis phase (Feasibility study, functional and non-functional requirements), Design, Testing and Implementation phases.

Finally, chapter-3 covers the summary, conclusions and Recommendations.

# CHAPTER 2 – SYSTEM DEVELOPMENT PROCESS

## 2.1 Analysis of tasks and activity

The tasks and activities required to achieve the project objectives have been evaluated. Issues within the project have also been carefully examined to identify potential areas for improvement and address any weaknesses. Once the requirements were gathered, the next step was to break them down into a clear process and design the functional framework.

### 2.1.1 Requirement Analysis

In this project, the tasks and activities needed to meet the objectives of the Job Portal System have been carefully studied. The goal is to make the job search and hiring process easier and more effective for both job seekers and employers. During the requirement analysis phase, the system’s needs were identified, and a basic plan was made to fulfill those needs through clear processes and functional design.

Requirement analysis helps us understand what users expect from the system. For this job portal system, the main needs are simple: job seekers want to search and apply for jobs easily, and employers want to post jobs and review applications quickly. After gathering these requirements, we broke them down into smaller steps and started designing how the system should work. This includes user registration, job posting, job search, applying for jobs, and managing applications. All these features were planned in a way that keeps the system easy to use and helpful for both sides.

**Functional requirements**

Here's a brief description of the functional requirements for JOB PORTAL SYSTEM:

* ⁠**User Registration and Login**: Job seekers and employers can create an account and log in using a valid email and password.
* **Job Search**: Job seekers can search for jobs based on their interest, category, or location.
* **Job Application**: Users can apply directly to job postings through the website.
* **Resume Generation**: For users who do not have a resume, the system provides an automatic resume generation feature. They just need to fill in basic details like education, experience, and skills.
* **Job Posting**: Employers can post job openings with required details like title, description, skills, and deadline.
* **Admin Panel**: Admins can manage users, approve job posts.
* **Payment Feature**: A basic demo payment system is added to show how users would make payments in the future, through eSewa. It is currently for demonstration only and does not handle real money.

Functional Requirements can be briefly described with the help of Use-Case Diagram which is shown below:

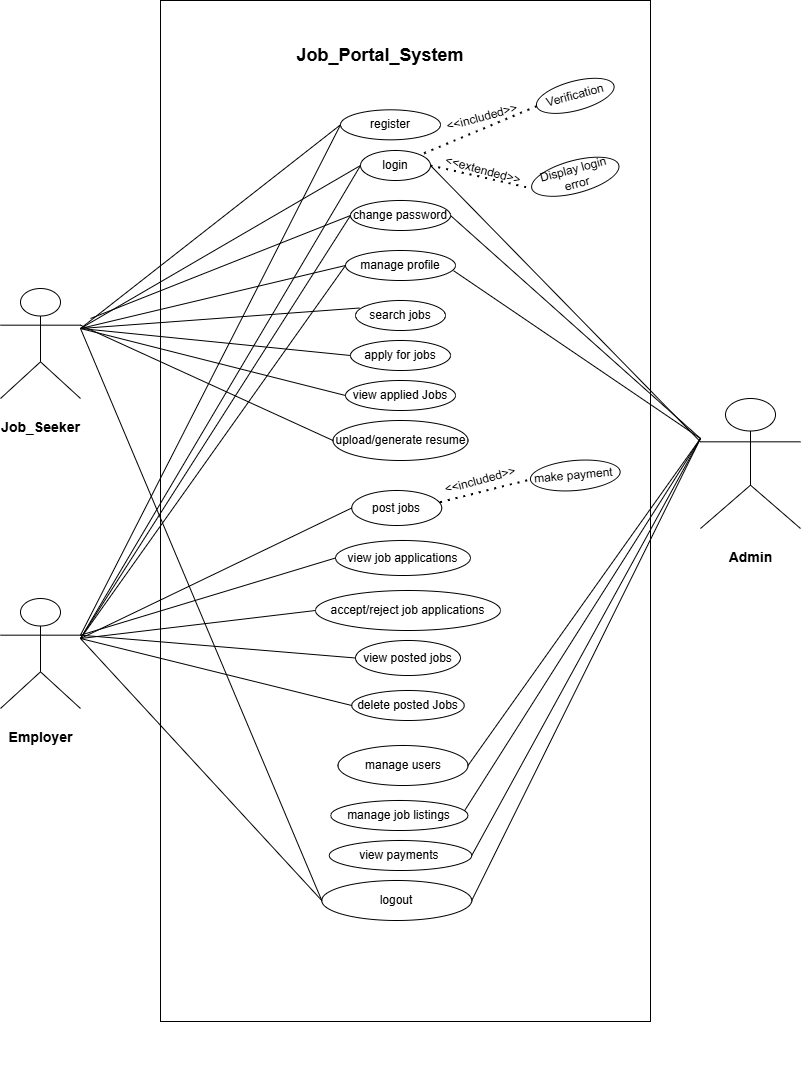


Figure 2.1: Use Case Diagram of Job Portal System

Table 2.1: Use case description (User Registration)

|  |  |
| --- | --- |
| Use-case Identifier | UC1: Registration |
| Primary Actor | Job Seeker |
| Secondary Actor | None |
| Description | The user fills up the required information, and the details are stored in the database. |
| Pre-condition | There should be registration form. |
| Post-condition | User data are saved in database. |
| Success scenario | The user gets redirect to the signup page after “Registration Successful” message appears. |
| Failure scenario | Database is not updated. |

Table 2.2: Use case description (Login)

|  |  |
| --- | --- |
| Use-case Identifier | UC2: Login into the system |
| Primary Actor | Job Seeker, Admin, Employer |
| Secondary Actor | None |
| Description | Admins and users can login from a login form. |
| Pre-condition | Actors needs to know about their username and passwords. |
| Post-condition | Actors get accessed to the system. |
| Success scenario | Actors get redirected to the dashboard. |
| Failure scenario | Displays “Username and password are incorrect” |

Table 2.3: Use case description (Post Jobs)

|  |  |
| --- | --- |
| Use-case Identifier | UC3: manage Jobs |
| Primary Actor | Employer |
| Secondary Actor | None |
| Description | Employer can add their Jobs. |
| Pre-condition | Employer must know their username and password. |
| Post-condition | The database must be updated accordingly |
| Success scenario | Success Message should be displayed. |
| Failure scenario | Database is not configured properly |

Table 2.4: Use case description (Job Apply)

|  |  |
| --- | --- |
| Use-case Identifier | UC4: Job Apply |
| Primary Actor | Job Seeker |
| Secondary Actor | None |
| Description | Job seeker can view and apply jobs and view jobs applied. |
| Pre-condition | Job seeker must know their username and password. |
| Post-condition | Job seeker can apply jobs and view total jobs applied. |
| Success scenario | Jobs can be applied and viewed. |
| Failure scenario | Failure message will be displayed. |

Table 2.5: Use case (Edit Profile)

|  |  |
| --- | --- |
| Use-case Identifier | UC5: Edit job profile |
| Primary Actor | Job seeker, Employer |
| Secondary Actor | None |
| Description | Job seeker & employer can view and edit their information. |
| Pre-condition | Job seeker and employer must know their username and password |
| Post-condition | Database must be updated accordingly. |
| Success scenario | Success message will be displayed. |
| Failure scenario | Failed to edit the profile. |

**Non-functional requirement**

It outlines the system's performance capabilities and limitations that improve its functionality. This could include factors such as speed, security, reliability, usability, and more.

* **User-Friendly Interface**: The system can provide a simple and clean interface so users can easily register, log in, and apply for jobs.
* **Performance**: The system can respond quickly when users search for jobs, upload resumes, or submit applications, without making them wait.
* **Availability**: The system can be available most of the time, especially during working hours, so users can access it whenever needed.
* **Security**: The system can use features like password protection and OTP verification through Gmail to keep user accounts secure.

### 2.1.2 Feasibility Study

A feasibility study for a Job Portal System involves evaluating four key aspects: technical, operational, economic, and schedule feasibility to determine if the project is viable.

**Technical feasibility** assesses whether the necessary technology and infrastructure are available to build the system. The Job Portal System is technically feasible because it uses basic and free technologies like HTML, CSS, JavaScript, and Laravel (PHP). These tools are easy to learn and available for free, which makes development easier for beginners. The system can be built and run using a normal computer and internet connection.

**Operational feasibility** focuses on how the system will integrate with existing operations and workflows. The Job Portal Management System is simple and easy to use, which makes it suitable for both job seekers and employers. It allows users to search, apply, and post jobs without any difficulty. The system is user-friendly and does not require special training, so it can be used in daily operations without causing problems. This makes the system operationally feasible for regular use.

**Economic feasibility** evaluates the financial aspects of the project, including development, operational, and maintenance cost. The Job Portal System is cost-effective because it uses free tools like HTML, CSS, JavaScript, and Laravel. There are no expensive resources needed for development, and some parts of the system are open-source, which helps keep costs low. This makes the project affordable and easy to maintain.

**Schedule feasibility** assesses whether the system can be developed and launched within the desired timeframe. This includes identifying project milestones, evaluating resource availability and addressing potential delays.

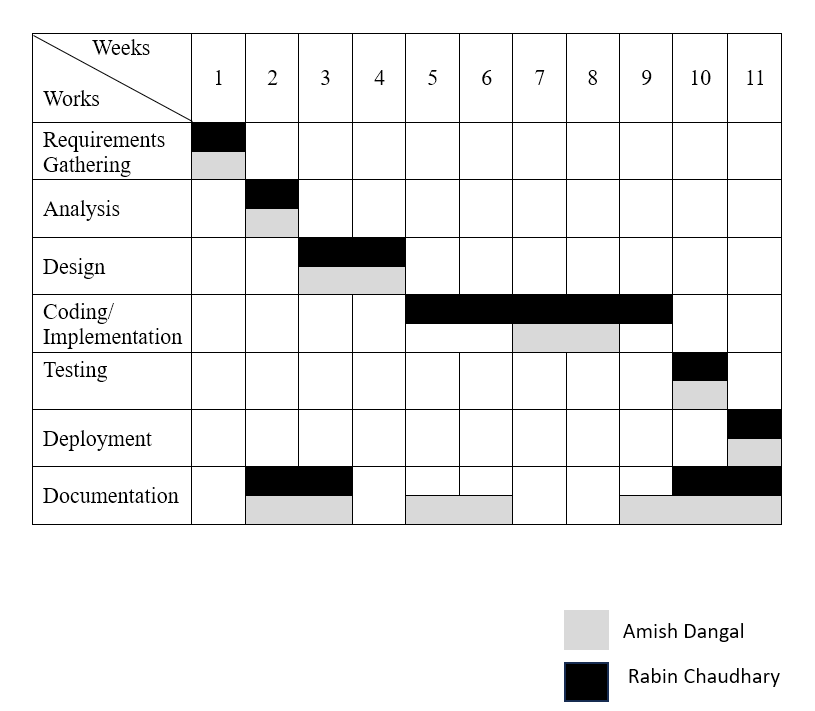


Figure 2.2: Gantt Chart

### 2.1.3 Object Oriented modeling

Object-Oriented Modeling (OOM) is a method of analyzing and designing a system by visualizing it as a group of interacting objects, each defined by their classes, attributes, behaviors (methods), and relationships.

1. **Class Diagram**

Class diagram is UML diagram that shows the structure and relationship between various classes/ entities involved in the system.

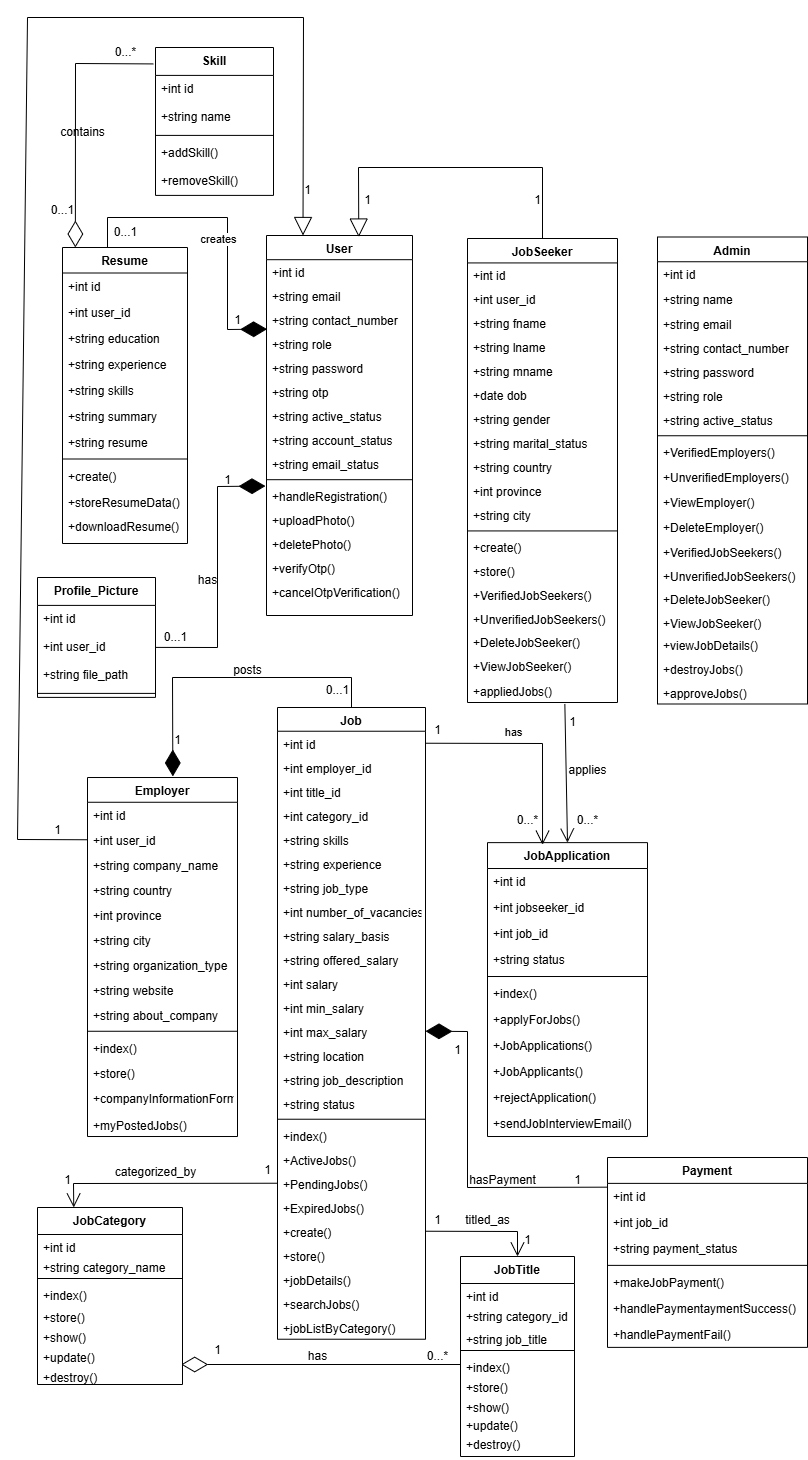


Figure 2.3: Class Diagram of Job Portal System

1. **Object Diagram**

It shows how objects (instances of classes) interact with each other, including their attributes and relationships.

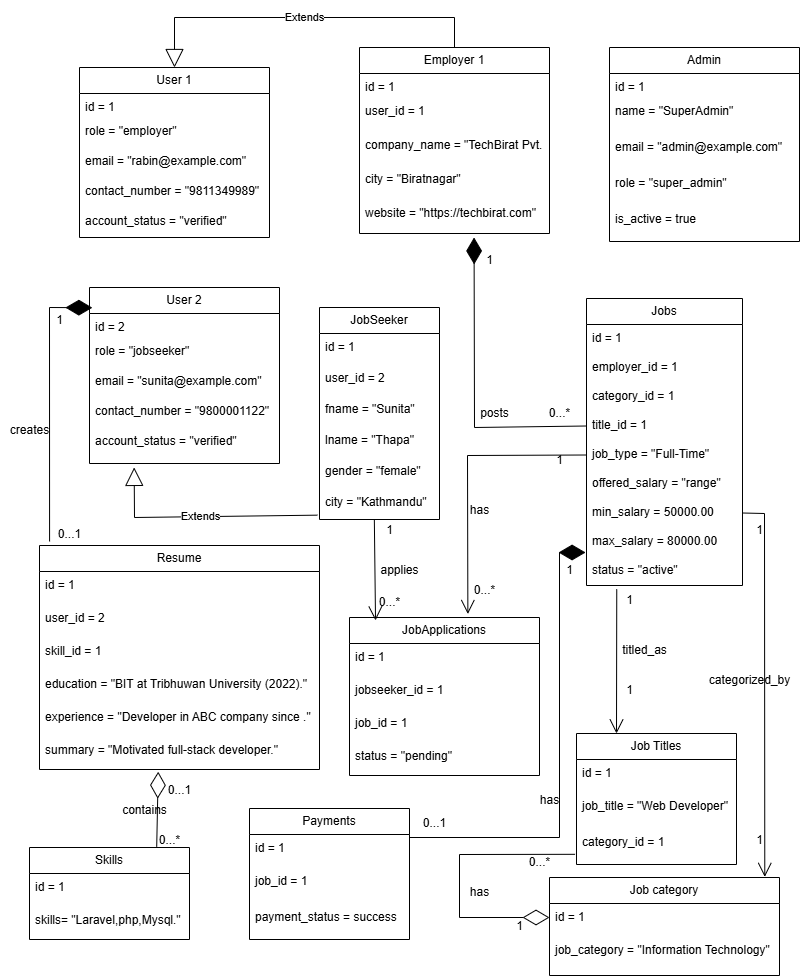


Figure 2.4: Object Diagram of Job Portal System

1. **Activity Diagram**

An activity diagram is a type of UML (Unified Modeling Language) diagram that visually represents the flow of activities or actions within a system or process. It shows the sequence of operations, decisions, and the overall workflow, helping to illustrate how various activities interact and progress.

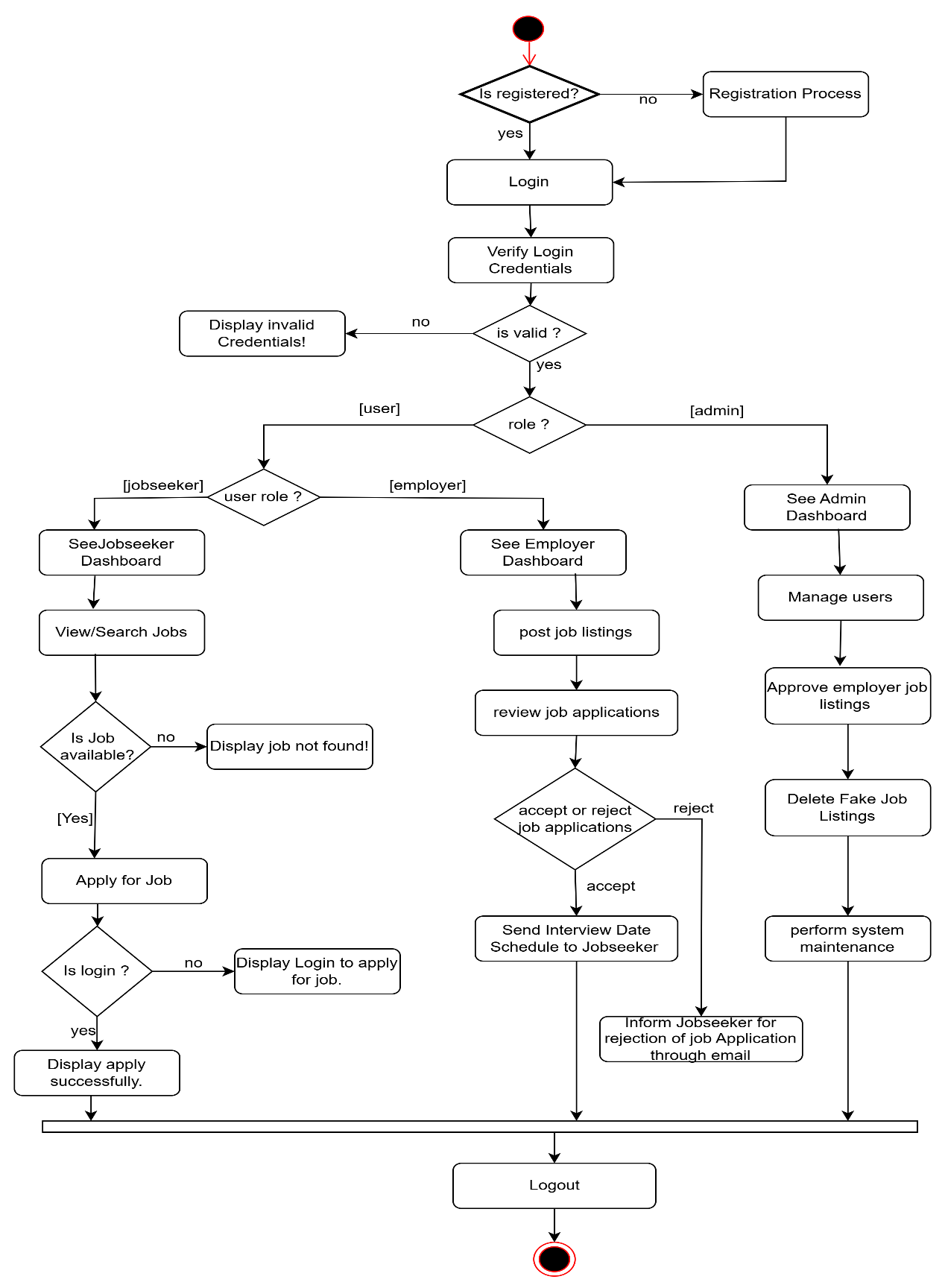


Figure 2.5: Activity Diagram of Job Portal System

1. **Sequence Diagram**

A sequence diagram is a type of UML (Unified Modeling Language) diagram that represents how objects interact with each other in a particular sequence over time. It shows the flow of messages between objects, typically in a linear order, and how these objects collaborate to achieve a specific functionality.

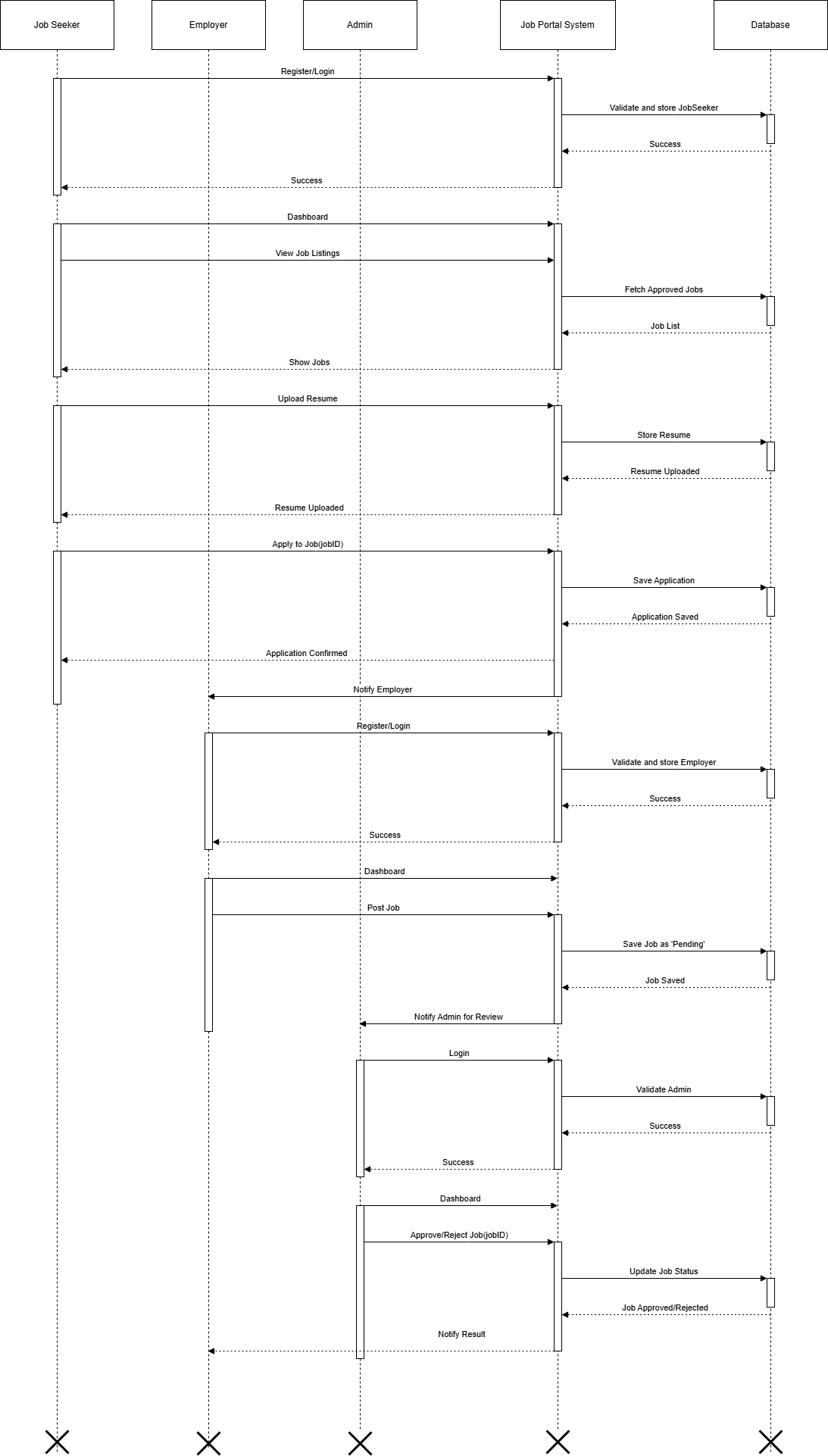


Figure 2.6: Sequence Diagram of Job Portal System

## 2.2 System design

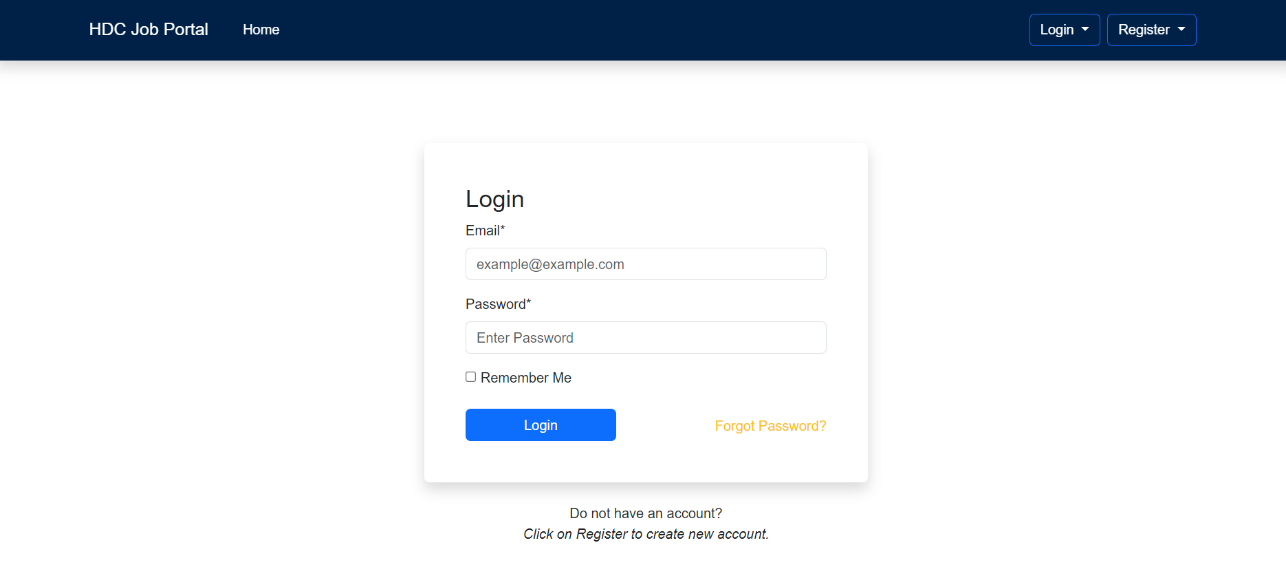
Job Portal System is a website that helps job seekers find jobs and employers find suitable candidates. Users can create accounts, upload resumes, search for jobs, apply for positions, and track their applications. Employers can post jobs, view applications, and contact potential candidates.

System design means planning how the different parts of the website will work together. It includes deciding how users, job posts, resumes, applications, and admin controls will interact with each other. This step helps us understand how data flows through the system and how different features connect, like login, job search, job posting, and application tracking.

### 2.2.1 User Interface

**i. User Login Interface**

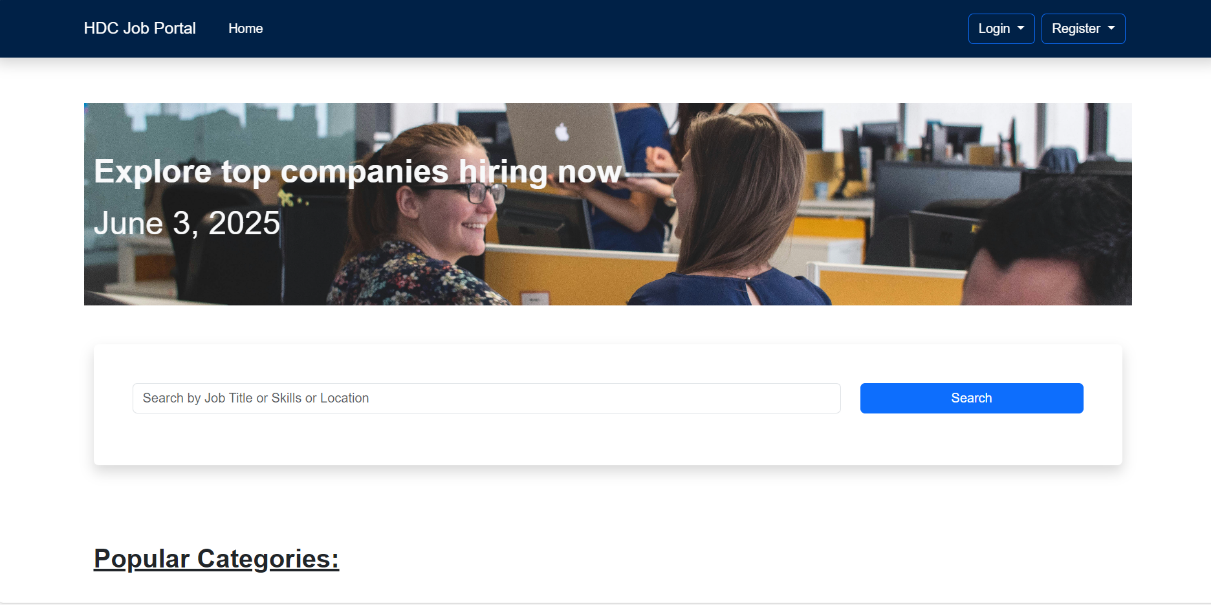
The Login page of Job Portal System allows registered users to access their account by entering their email and password. It verifies the credentials against stored records and grant access if the information is correct. If the login fails, it shows an error message. This page is essential for users to securely access features like, post job listings, apply for jobs, manage job applicants, and managing their profiles.



**Design 1: Login Page Interface**

**ii. Landing Page Interface**

The users landing page of Job Portal System is the main interface users see this page while they visit the website. Users can not access the all features without logging in to the website but if they will successfully logged in then they can access all the features of the site like, post job listings, apply for jobs, manage job applicants, and managing their profiles.

****

**Design 2: Landing Page Interface**

### 2.2.2 Database design/Object oriented design models

System Design describes the design of the overall system. The system is designed on the MVC architecture, Model, View, Controller. We choose this because it will help in the maintenance of the application and facilities other benefits like modification being easy with multiple views and fast development.

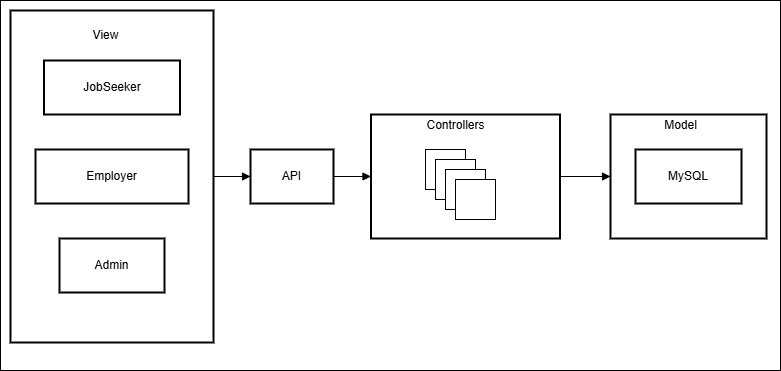


Figure 2.7: MVC Architecture

We have, modules in the view for the Job Seekers, Employers and admin, followed by their controllers for each view, which will act like a bridge between the view and the model. For the model, there is MySQL as a database.

## 2.3 Implementations

The implementation of the Job Portal System involved several phases, from requirement analysis and feasibility study to system design and the integration of core functionalities that enable seamless user registration/login, job posting and application workflows, automatic resume generation, feedback management, and administrative reporting. This section outlines the process followed for implementation, including the tools and technologies used in developing the system.

**Key Features implemented**

1. **User Registration and Login:**

The system allows job seekers and employers to register, log in, and manage their profiles. Secure session-based authentication ensures user data remains protected.

1. **Job Posting and Application:**

Employers can post new job vacancies—including title, description, requirements, location, and salary. Job seekers can browse, filter, and apply for positions directly through the portal.

1. **Admin Panel:**

Administrators can manage user accounts, approve or suspend postings, and oversee overall system activity.

1. **Automatic resume Generation:**

Users enter their education, work history, skills, and other details; the system then compiles this information into a professionally formatted resume.

### 2.3.1 Tools and Technologies Used

1. **Laravel Framework:**

Laravel, a PHP framework, was used for the backend of the job portal. Its MVC structure keeps business logic and presentation separate. Features like Eloquent ORM (for database operations), Blade templates (for page layouts), and Artisan commands (for tasks like setting up and testing) made the app easy to build, maintain, and scale.

1. **MySQL Database:**

**MySQL** was used as the relational database management system for storing user profiles, job postings, applications, CV data, and feedback. MySQL’s relational model made it easy to design and manage these tables and ensured efficient queries for retrieving and updating records. Indexing and transaction support further guaranteed data integrity and fast search performance even as the volume of records grew.

1. **HTML, CSS and JavaScript:**

For the front-end, **HTML** and **CSS** were used to structure and style the web pages. **JavaScript** was used for dynamic content, such as updating available seats in real-time and handling client-side validation for form submissions.

1. **XAMPP:**  
   **XAMPP** was used as a local server environment for development and testing purposes. It provided all the necessary services, including Apache, MySQL, to simulate a live environment before deployment.

|  |  |
| --- | --- |
| **Software** | **Purpose** |
| VS Code | For coding purpose |
| Laravel framework | Simplifies web development |
| Draw.io | For models and diagrams |
| Google chrome | As a web browser for running the system |
| Apache, MySQL | To connect with database |

Table 2.6: Tools Used

### 2.3.2 Module Description

|  |  |
| --- | --- |
| **Element** | **Description** |
| Purpose | To securely manage user logins, registrations, and session handling. |
| Functionality | Allows users to sign up, log in, log out, and manage account credentials. |
| Inputs | Email, password, user role (admin/employer/job seeker). |
| Outputs | Login session, user profile access, error messages (invalid login, etc.). |
| Users | Job seekers, Admins, Employer |

Table 2.7: User Authentication Module

Table 2.8: Admin Dashboard Module

|  |  |
| --- | --- |
| **Element** | **Description** |
| Purpose | Provides tools for admins to manage the job portal and oversee user activities. |
| Functionality | Manage users, jobs, and applications; monitor system activity; generate reports. |
| Inputs | Admin actions such as user management and system monitoring |
| Outputs | User lists, job listings, application stats, system activity reports. |
| Users | Admins only. |

Table 2.9: Job Seeker Dashboard Module

|  |  |
| --- | --- |
| **Element** | **Description** |
| Purpose | Provides job seekers with tools to search jobs, apply, and manage applications. |
| Functionality | Search for jobs, apply to jobs, view application status, update profile |
| Inputs | Job seeker interactions such as search queries, job applications, and profile updates. |
| Outputs | Job listings, application confirmations, application statuses. |
| Users | Job seekers only. |

Table 2.10: Employer dashboard module

|  |  |
| --- | --- |
| **Element** | **Description** |
| Purpose | Provides employers with tools to post jobs, manage listings, and review applications |
| Functionality | Post job listings, view and manage applications, update company profile. |
| Inputs | Employer actions such as job postings, application reviews, and profile updates. |
| Outputs | Job listing confirmations, application lists, applicant details |
| Users | Employers only. |

### 2.3.3 Testing

**UNIT TESTING:**

Unit Testing is a type of software testing where individual parts (or *units*) of a program are tested on their own to make sure they work correctly.

Table 2.11: Test case 1 - Registration form

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test cases** | **Description** | **Expected Result** | **Input** | **Output** | **Remarks** | **Tested by** |
| 1 | Register with valid details | Registration successful, user redirected to dashboard | Valid name, email, password, phone, etc. | Success message, user added to database and transfer to dashboard | Pass | Rabin Kumar Chaudhary |
| 2 | Register with missing required fields | Show validation error message | Missing name or email or password | “Please fill out this field” message | Pass | Rabin Kumar Chaudhary |
| 3 | Register with already registered email | Show duplicate email warning | Email already in use | "The email has already been taken" message | Pass | Rabin Kumar Chaudhary |
| 4 | All fields valid, user already registered | Prevent re-registration | Already account created | “This email has already been taken” message | Pass | Rabin Kumar Chaudhary |

Table 2.12: Test case 2: Login form

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case** | **Description** | **Expected Result** | **Input** | **Output** | **Remarks** | **Tested by** |
| 1 | Login with valid credentials | Login successful, user redirected to dashboard | Valid username and correct password | Successful login, redirect to user dashboard | Pass | Amish Dangal |
| 2 | Login with missing credentials | Show validation error message | Missing username or password field | "Please fill out this field" message | Pass | Amish Dangal |
| 3. | Login with invalid credentials | Show validation error message | Invalid username or password field | “These credentials do not match our records” message | Pass | Amish Dangal |

Table 2.13: Test case 3: Application form

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case** | **Description** | **Expected Result** | **Input** | **Output** | **Remarks** | **Tested by** |
| 1 | Apply for a job with valid details | Application submitted successfully | Logged-in job seeker, selected job, resume uploaded | “Application confirmation” message | Pass | Amish Dangal |
| 2 | Attempt to apply without uploading resume | Show validation error | Logged-in job seeker, selected job, missing resume | “Please upload your resume” message | Pass | Amish Dangal |
| 3 | Attempt to apply without selecting a job | Show validation error | Logged-in job seeker, no job selected, resume uploaded | Please select a job to apply | Pass | Amish Dangal |

**SYSTEM TESTING:**

System Testing is a software testing process where the complete application is tested as one unified system. The goal is to check whether the entire software works according to the given requirements. This is the last stage of testing before the product is released to the end users.

Table 2.14: Test case 1: Job Application process

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Case** | **Test Data** | **Expected Outcome** | **Actual Outcome** | **Remarks** | **Tested by** |
| 1 | Register with valid details | Name: Amish Dangal, Email:  amis@example.com, Phone:9811122234  Password: Amish@123,  Confirm password: Password: Amish@123 | User registered successfully and redirected to login page | User registered successfully and redirected to login page | Pass | Rabin Kumar Chaudhary |
| 2 | Login with valid credentials | Email:  amis@example.com, Password: Amish@123 | Login successful, redirected to dashboard | Login successful, redirected to dashboard | Pass | Amish Dangal |
| 3 | Apply for job with valid data | Job selected: Manager, Resume uploaded | Application submitted; confirmation message shown | Application submitted; confirmation message shown | Pass | Amish Dangal |
| 4 | Register, login, and apply (full flow) | Valid registration > valid login > job selected > resume uploaded | All steps work correctly without any error | All steps work correctly without any error | Pass | Amish Dangal |
| 5 | Withdraw an application | Job: Manager, User confirms withdrawal | Application withdrawn; message displayed | Application withdrawn | Pass | Rabin Kumar Chaudhary |

# CHAPTER 3 - DICUSSIONS AND CONCLUSIONS

## 3.1 Summary

The **Job Portal System** offers a simple and user-friendly way for job seekers to find and apply for jobs. It allows users to register, log in, search for jobs by location, category, or type, apply for jobs, and track the status of their applications. Job seekers can also upload their resume and update their profile easily.

Employers can post job openings, view applications, and manage their hiring process through a separate dashboard. Admins can manage users and verify job listings to make sure the system runs smoothly and safely.

The system reduces manual effort, saves time, and helps both job seekers and employers connect faster. Key modules include user registration and login, job search and application, resume upload, employer job posting, and admin management.

The website is built using HTML, CSS, JavaScript, and PHP. It is designed to be secure, fast, and easy to use on any device. The simple design helps users find what they need quickly and improves their overall experience.

In the future, I plan to create a mobile app, add secure payment for premium features, suggest jobs based on user profiles, and add messaging between job seekers and employers. Feedback from users will help us make the platform even better and more useful.

## 3.2 Conclusions

In conclusion, the “Job Portal System” has made the process of finding jobs and hiring employees easier and more organized. It’s simple and built in features help job seekers search and apply for jobs quickly, while allowing employers to post job openings and manage applications in one place. The system saves time, reduces manual work, and improves communication between job seekers and employers.

With real-time updates, easy navigation, and secure features, the platform gives users a smooth and reliable experience. Job seekers can manage their profiles, upload resumes, and track their applications, while employers can review candidates and handle their job listings effectively.

Built using HTML, CSS, JavaScript, and PHP, the system is designed to be safe, user-friendly, and fast. By bringing everything into one platform, it helps improve job matching, makes hiring more efficient, and supports both users in reaching their goals. Overall, the Job Portal System is a useful tool for connecting the right people with the right jobs.

## 3.2 Recommendations

* Real time tracking
* Mobile app development
* Customer Feedback and Review System
* Multi-language Support
* Secure Payment Gateway
* AI based Recommendation System

# REFERENCES

MeroJobs. (2025, April 27). *MeroJobs – Find jobs and hire top talent in Nepal*. https://merojobs.com/. Accessed on (2025-04-27)

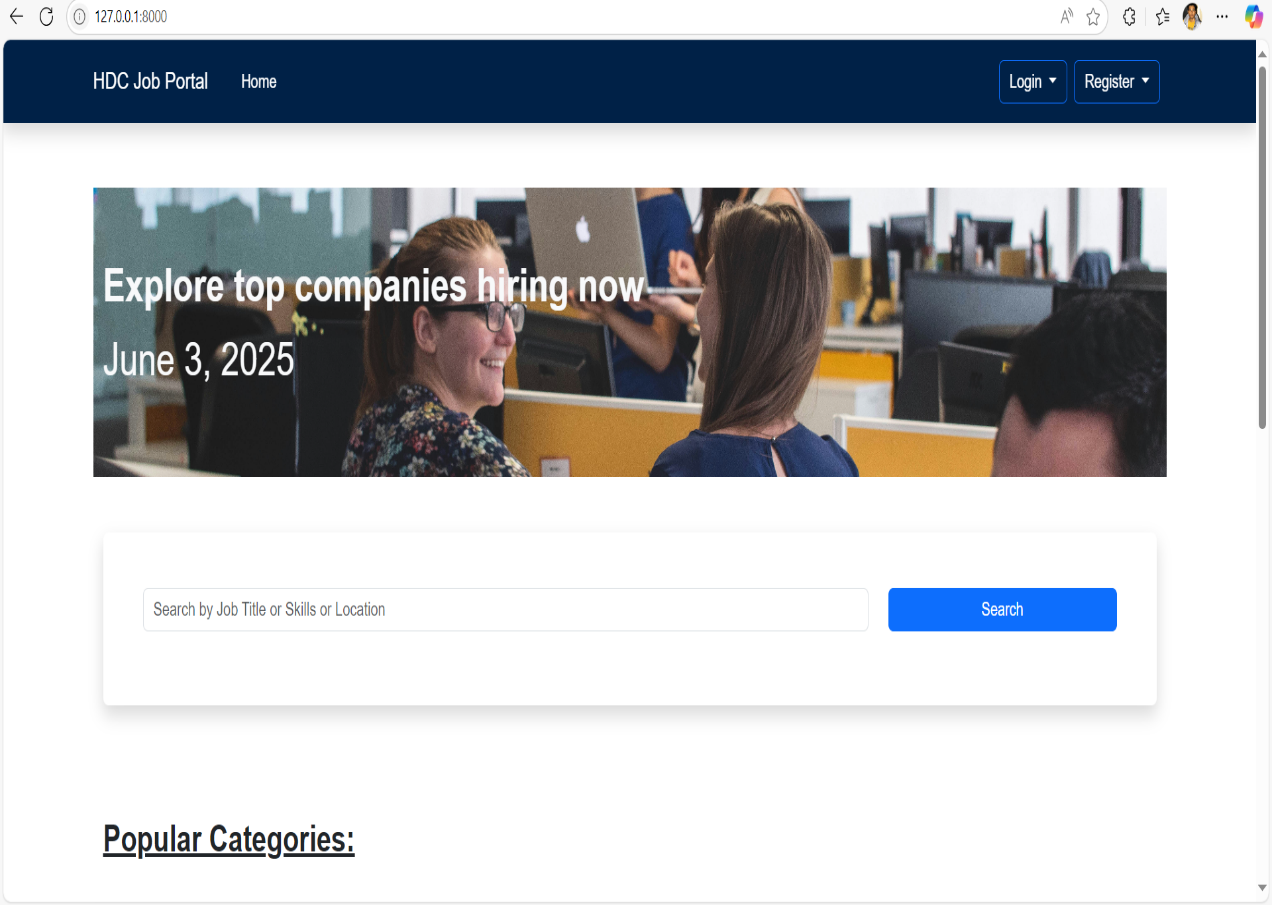
JobsNepal. (2025, April 29). *JobsNepal – Nepal’s leading job portal for employers and job seekers.* https://jobsnepal.com/. Accessed on (2025-04-29)

HamroJobs. (2025, May 10). *HamroJobs – Nepal’s top job portal for job seekers and employers.* https://hamrojobs.com/. Accessed on (2025-05-10)

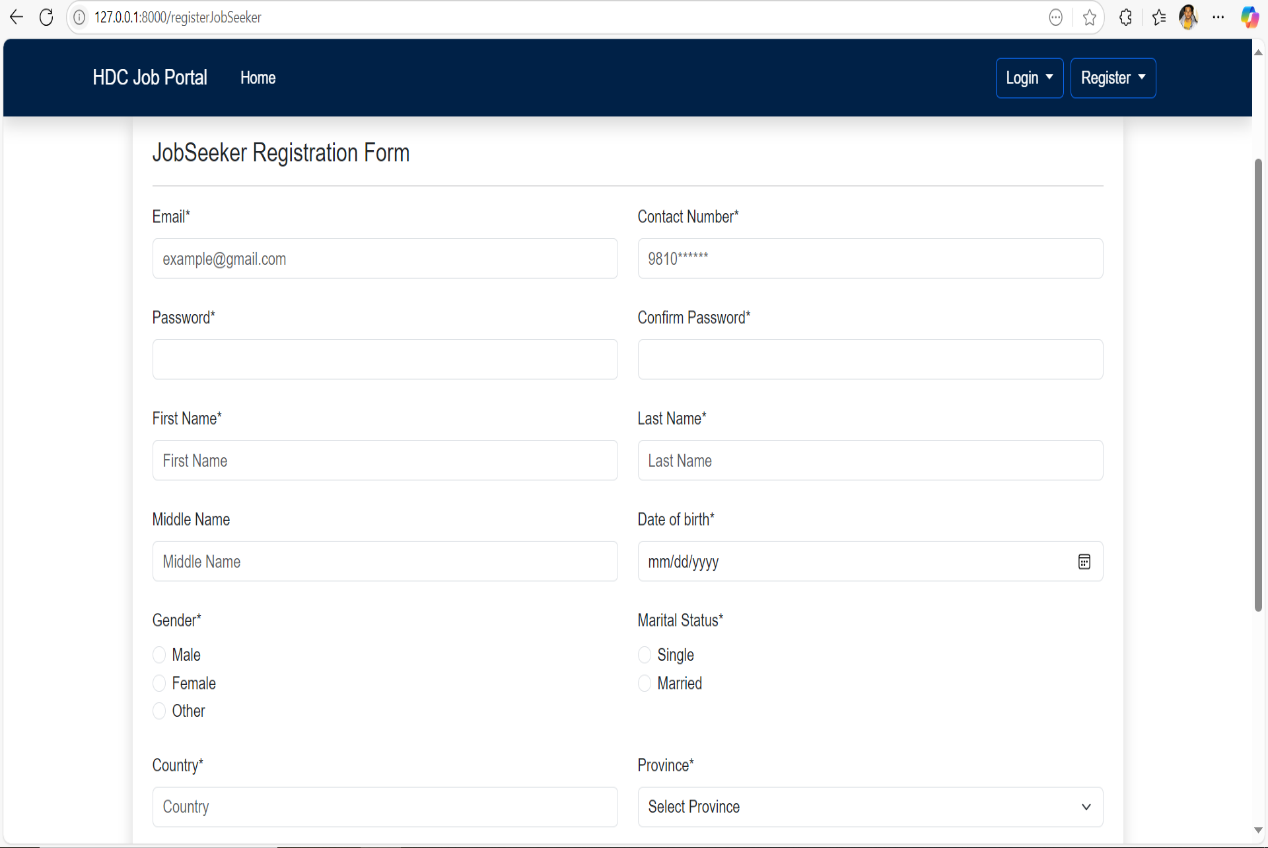
W3Schools. (n.d.). *PHP tutorial*. https://www.w3schools.com/php/. Accessed on (2025-05-20)

Ullman, L. (2011). *PHP and MySQL for dynamic web sites: Visual QuickPro Guide* (4th ed.). Peachpit Press.

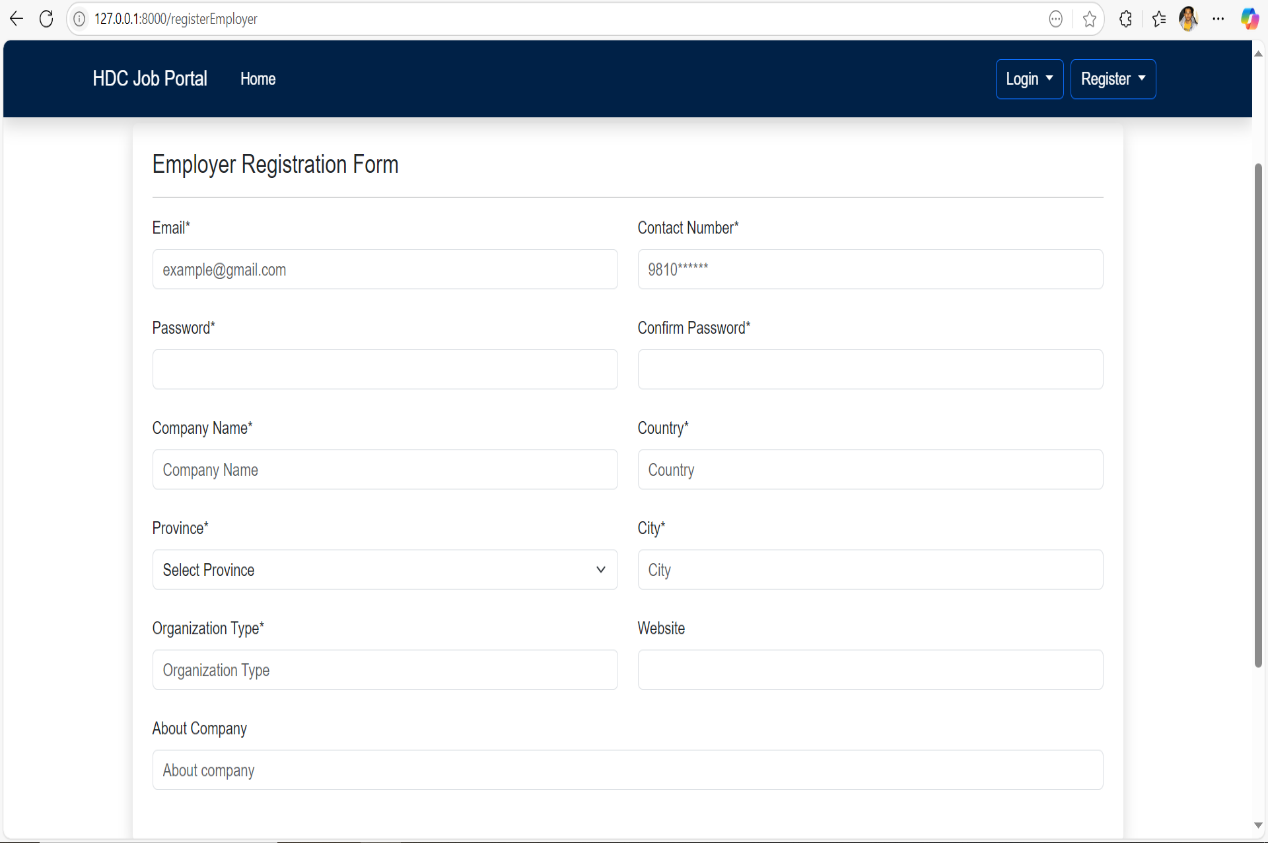
# APPENDIX

****

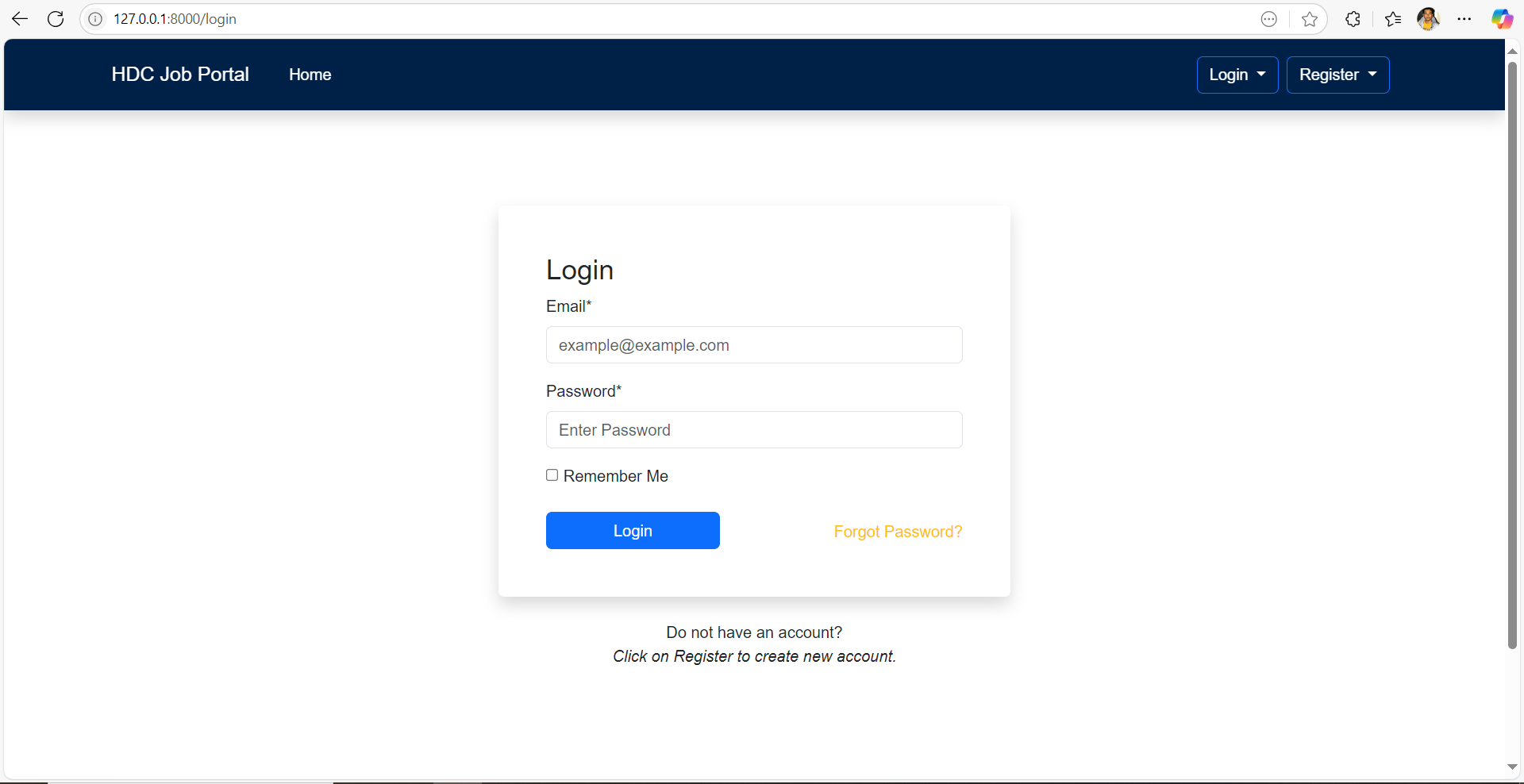
**Appendix 1: Landing Page**

****

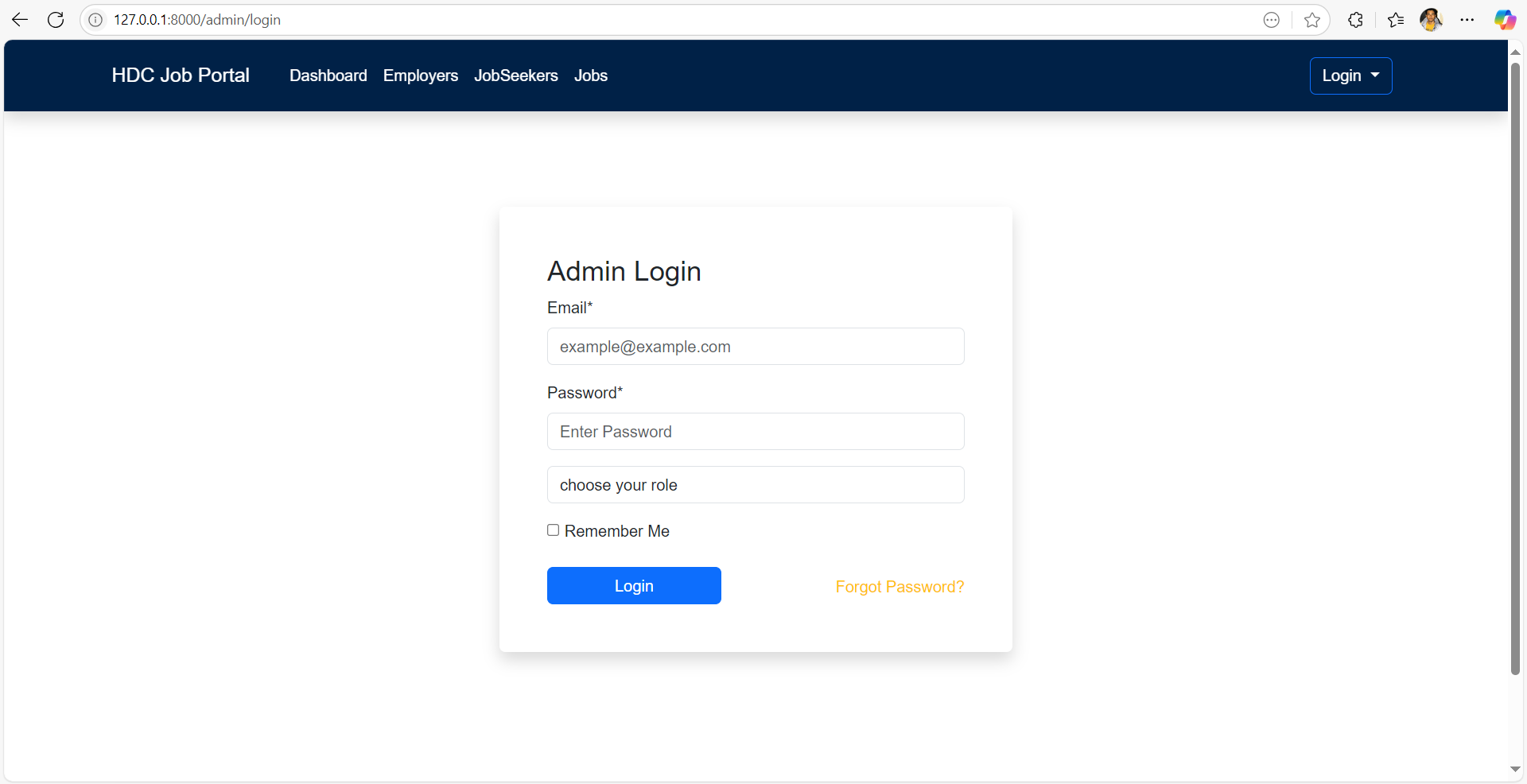
**Appendix 2: Job Seeker Registration Form**

****

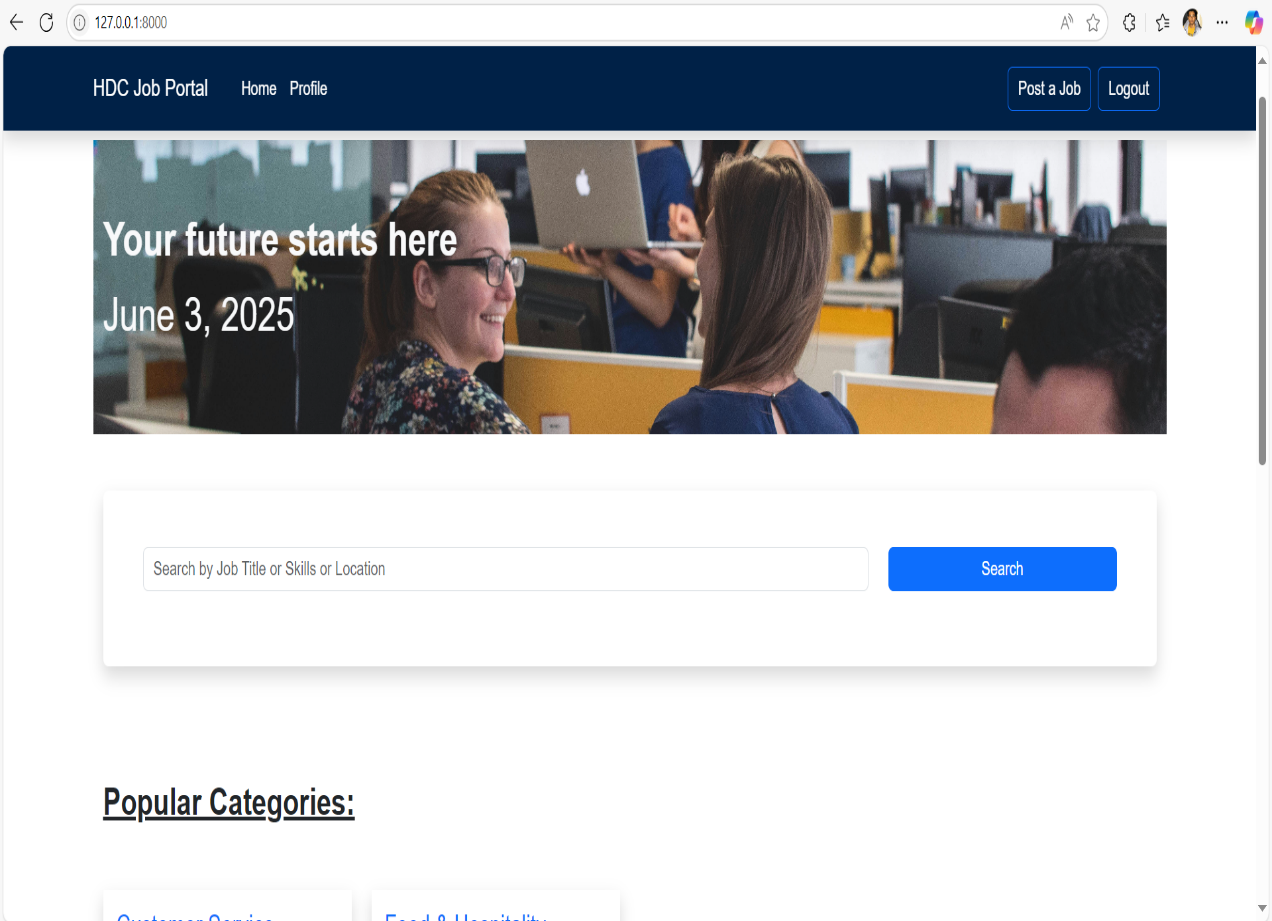
**Appendix 3: Employer Registration Form**

****

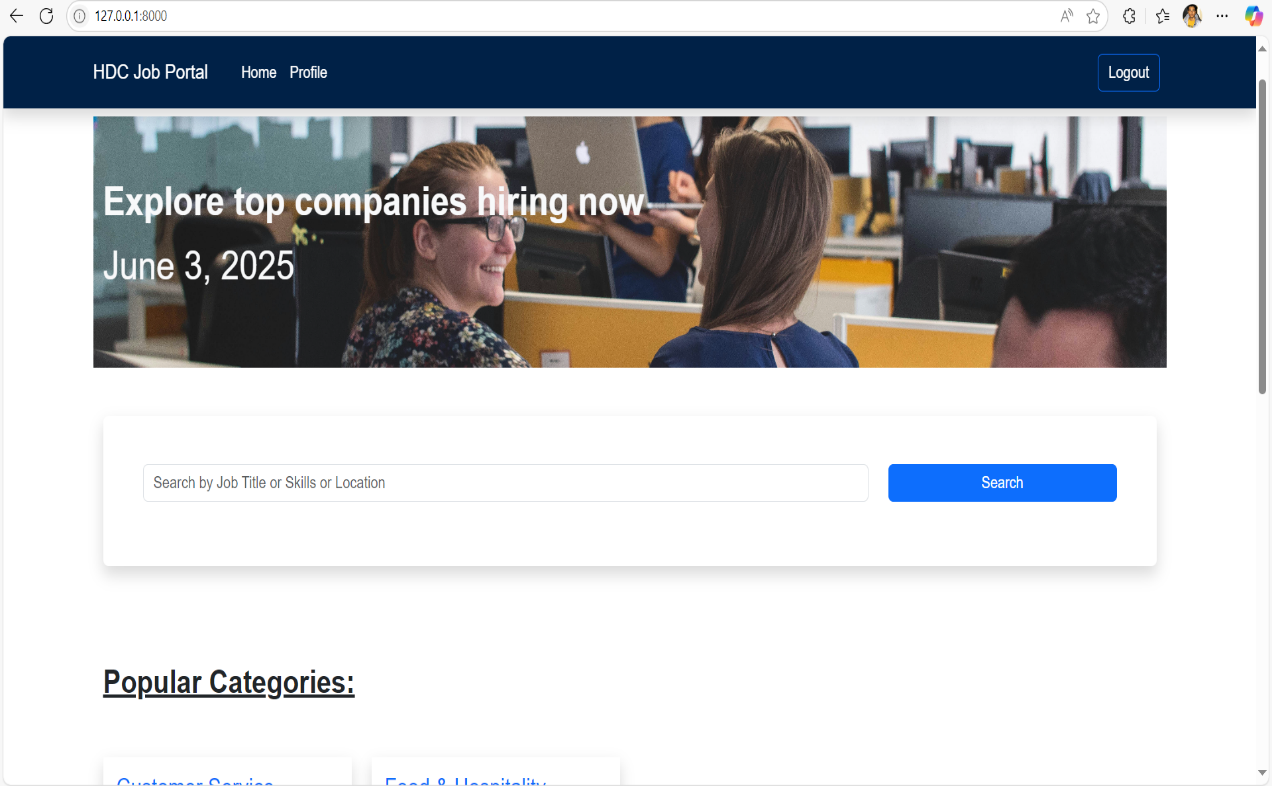
**Appendix 4: User Login Page**

****

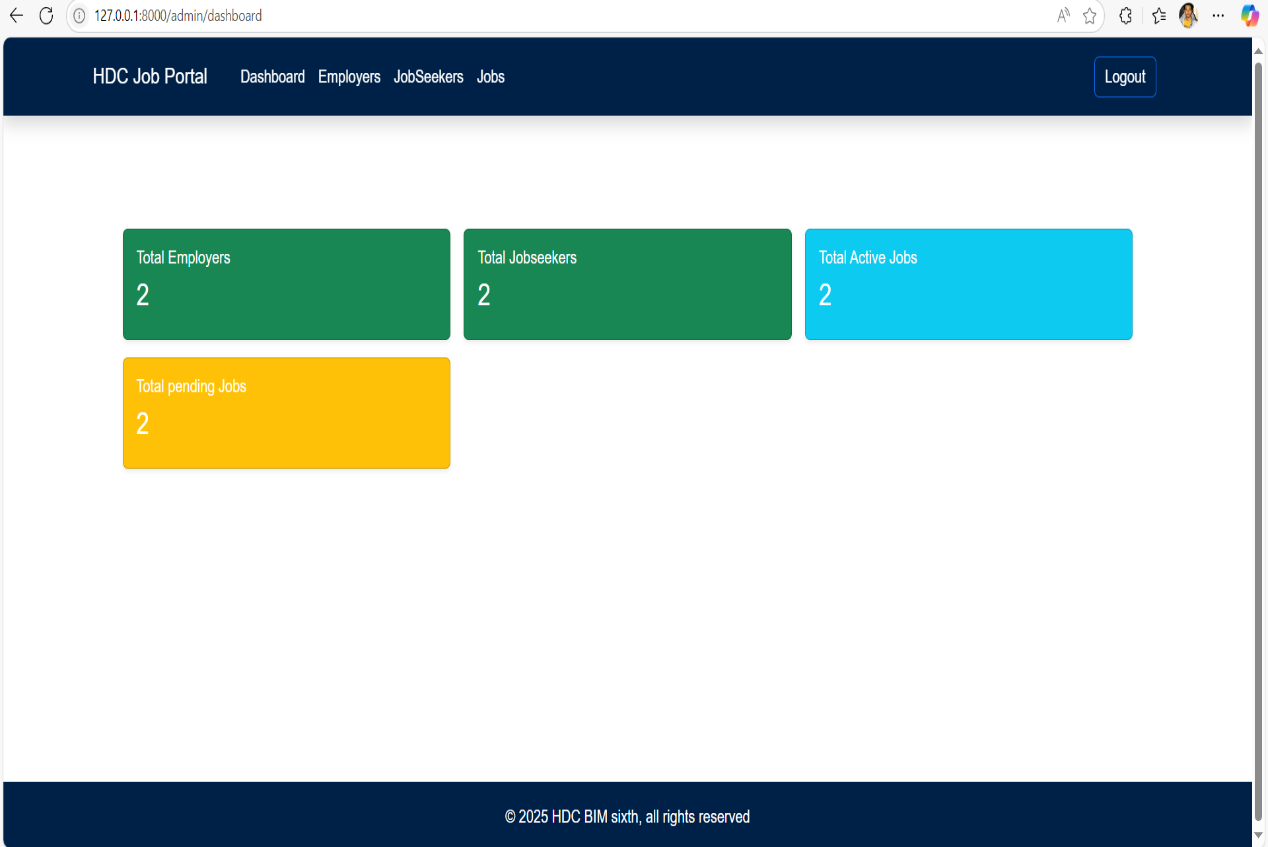
**Appendix 5: Admin Login Page**

****

**Appendix 6: Employer Dashboard after Login**



**Appendix 7: Job Seeker Dashboard After Login**



**Appendix 8: Admin Dashboard After Login**