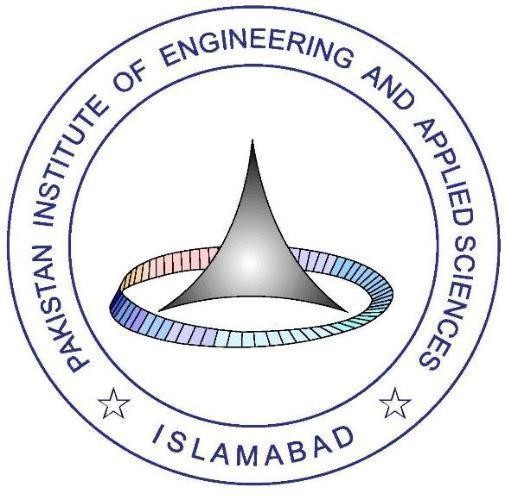
**Development of**

**Job Recruitment Portal**



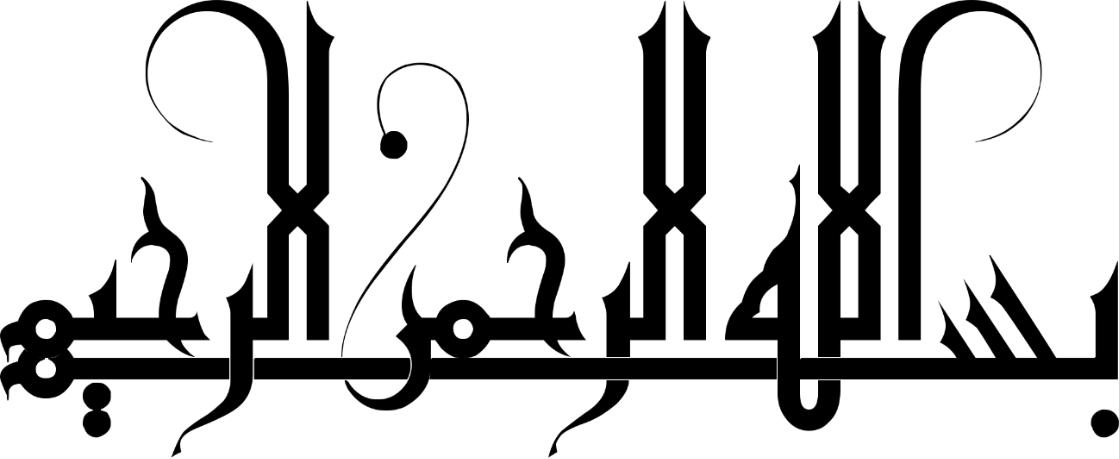
**Mohammad Ahsan Junaid**

**Abdul Rehman**

**Thesis submitted in partial fulfillment of requirements for the Degree of Bachelor of Sciences in Computer and Information Sciences**

**Department of Computer and Information Sciences, Pakistan Institute of Engineering & Applied Sciences, Nilore, Islamabad, Pakistan.**

**May, 2023**



***In the name of Allah, the Entirely Merciful, the Especially Merciful***

**Department of Computer and Information Sciences,**

**Pakistan Institute of Engineering and Applied Sciences, Nilore, Islamabad 45650, Pakistan**

**Declaration of Originality**

I hereby declare that the work contained in this thesis and the intellectual content of this thesis are the product of my own work. This thesis has not been previously published in any form nor does it contain any verbatim of the published resources which could be treated as an infringement of the international copyright law.

I also declare that I do understand the terms ‘copyright’ and ‘plagiarism,’ and that in case of any copyright violation or plagiarism found in this work, I will be held fully responsible for the consequences of any such violation.

**Certificate of Approval**

***This is to certify that the work contained in this thesis entitled***

**“Development of a Web Application, Job Recruitment**

**Portal”**

***was carried out by***

**Mohammad Ahsan Junaid**

**Abdul Rehman**

***Under our supervision and that in our opinion, it is fully adequate, in scope and quality, for the degree of BS Computer and Information Sciences from Pakistan Institute of Engineering and Applied Sciences (PIEAS).***

***Approved by:***

**Signature: Supervisor: *Dr. Anila Usman***

**Signature: Co-Supervisor: *Sir Irfan Hameed***

***Verified by:***

**Signature: Head, DCIS: Dr. Javaid Khurshid**

**Stamp:**

**This project is dedicated to our beloved family, teachers and friends, who offered us unconditional love and support throughout the course of this project.**

**Acknowledgement**

First and foremost, I would like to thank Allah, our sole creator, helper, guider, endower of everything that we have in our lives and forgiver for all the sins that we are guilty of committing. If it wasn’t for His enormous bounty, we would have never achieved anything in our lives and be lost.

I express my sheer appreciation to my supervisor Dr. Anila Usman. She has always been keen on helping and has provided me the guidance on every aspect concerning my project. Even with her grueling schedule she has always taken out time to clear away our doubts or solved any problem which impeded my work.

I also express my sheer appreciation to my co-supervisor, Sir Irfan Hameed. He has always been there for me whenever I needed him, and he has provided me with the necessary guidance on how to proceed with my project. He has helped me immensely in understanding the fundamentals of web development.

Lastly but most importantly I would express my eternal gratitude towards my parents who have always been our backbone throughout the years and have never let me down no matter what.

**Mohammad Ahsan Junaid**

**Abdul Rehman**

**PIEAS, Nilore, Islamabad**

**Table of Contents**

Contents

[Abstract 1](#_Toc135944673)

[Chapter 1: Introduction 2](#_Toc135944674)

[1.1. Problem Definition 2](#_Toc135944675)

[1.2. Proposed System 3](#_Toc135944676)

[1.3 System Functions 4](#_Toc135944677)

[Chapter 2: Literature and Tech Review 5](#_Toc135944678)

[2.1. Management Portal 5](#_Toc135944679)

[2.2. Programming Languages and Frameworks 5](#_Toc135944680)

[2.2.1. HTML 5](#_Toc135944681)

[2.2.2. CSS 6](#_Toc135944682)

[2.2.3. JavaScript 6](#_Toc135944683)

[2.2.4. Bootstrap 7](#_Toc135944684)

[2.2.5. PHP 7](#_Toc135944685)

[2.2.6. MySQL 7](#_Toc135944686)

[2.3. Software’s and Technologies Used 8](#_Toc135944687)

[2.3.1. XAMPP 8](#_Toc135944688)

[2.3.2. Visual Studio 8](#_Toc135944689)

[2.3.3. Apache 9](#_Toc135944690)

[2.3.4. Github 9](#_Toc135944691)

[2.3.5. phpMyAdmin 10](#_Toc135944692)

[2.3.6. 000WebHost 10](#_Toc135944693)

[Chapter 3: Software Design & Architecture 11](#_Toc135944694)

[3.1. System Overview 11](#_Toc135944695)

[3.2 Functional Requirements 11](#_Toc135944696)

[3.2.1. Users / Applicant 11](#_Toc135944697)

[3.2.2. Employers 11](#_Toc135944698)

[3.3 Non Functional Requirements 12](#_Toc135944699)

[1. Security 12](#_Toc135944700)

[2. Availability 12](#_Toc135944701)

[3. Scalability 12](#_Toc135944702)

[4. Performance 12](#_Toc135944703)

[5. Usability 12](#_Toc135944704)

[3.4 System Architecture 13](#_Toc135944705)

[3.4.1. Client-Side Elements 13](#_Toc135944706)

[3.4.2. Components for the server 13](#_Toc135944707)

[3.4.3. Infrastructure Elements 13](#_Toc135944708)

[3.4.4. Application Layer 13](#_Toc135944709)

[3.4.5. Database Layer 13](#_Toc135944710)

[3.5. Database Design 14](#_Toc135944711)

[3.5.1. Data Storage 14](#_Toc135944712)

[3.5.2. Data Security 14](#_Toc135944713)

[3.5.3. Data Organization 14](#_Toc135944714)

[3.5.4. Data Scalability 14](#_Toc135944715)

[3.5.5. Data Persistence 14](#_Toc135944716)

[3.6. Scope 15](#_Toc135944717)

[3.6.1. Features Falling Within Scope 15](#_Toc135944718)

[3.6.2. Scope in Real World Use 16](#_Toc135944719)

[Chapter 4: Implementation 17](#_Toc135944720)

[4.1. Technology & Tools Used 17](#_Toc135944721)

[4.2. Front End Development 17](#_Toc135944722)

[4.2.1. Home Page 17](#_Toc135944723)

[4.2.2. Job Posting Page 19](#_Toc135944724)

[4.2.3. Login Page 20](#_Toc135944725)

[4.2.4. Applicant Login Page 21](#_Toc135944726)

[4.2.5. Employer Login Page 22](#_Toc135944727)

[4.2.6. Employer Login Page 23](#_Toc135944728)

[4.2.7. Sign Up Page 24](#_Toc135944729)

[4.2.8. Applicant Registration Page 25](#_Toc135944730)

[4.2.9. Employer Registration Page 26](#_Toc135944731)

[4.2.10. Applicant Dashboard – Edit Profile 27](#_Toc135944732)

[4.2.11. Applicant Dashboard – Job Applications 28](#_Toc135944733)

[4.2.12. Applicant Dashboard – Mailbox 29](#_Toc135944734)

[4.2.13. Applicant Dashboard – Read / Reply Mail 30](#_Toc135944735)

[4.2.14. Applicant Dashboard – Settings 31](#_Toc135944736)

[4.2.15. Employer Dashboard 32](#_Toc135944737)

[4.2.16. Employer Dashboard – Company Details Page 33](#_Toc135944738)

[4.2.17. Employer Dashboard – Create Job Page 34](#_Toc135944739)

[4.2.18. Employer Dashboard – Posted Jobs 35](#_Toc135944740)

[4.2.19. Employer Dashboard – Job Applications Page 36](#_Toc135944741)

[4.2.20. Employer Dashboard – Mailbox 37](#_Toc135944742)

[4.2.21. Employer Dashboard – Reply / Create Mail Page 38](#_Toc135944743)

[4.2.22. Employer Dashboard - Settings 39](#_Toc135944744)

[4.2.23. Employer Dashboard – Applications Database 40](#_Toc135944745)

[4.2.24. Admin Dashboard 41](#_Toc135944746)

[4.2.25. Admin Dashboard – Active Jobs 42](#_Toc135944747)

[4.2.26. Admin Dashboard – Applicants Database 43](#_Toc135944748)

[4.2.27. Admin Dashboard – Employers Database 44](#_Toc135944749)

[4.3. Backend Development 45](#_Toc135944750)

[4.3.1. How to Calculate Workload? 45](#_Toc135944751)

[4.3.2. How to Handle Workload Weights? 45](#_Toc135944752)

[4.4. Database Structure 46](#_Toc135944753)

[4.4.1. Home Page 46](#_Toc135944754)

[4.5. Web Hosting 46](#_Toc135944755)

[4.4.2. Home Page 46](#_Toc135944756)

[Chapter 5: Structure 47](#_Toc135944757)

[5.1. Front End Structural Diagram 47](#_Toc135944758)

[5.2. Backend and Database Diagrams 47](#_Toc135944759)

[5.2.1. ER Diagram 47](#_Toc135944760)

[5.1.1. Use Case Diagram 47](#_Toc135944761)

[5.1.1. Sequence Diagram 47](#_Toc135944762)

[5.1.1. Relational Schema 47](#_Toc135944763)

[Chapter 6: User Roles 48](#_Toc135944764)

[6.1 . Administrator 48](#_Toc135944765)

[6.2 . Employer 48](#_Toc135944766)

[6.2.1 ER Diagram 48](#_Toc135944767)

[5.1.2. Use Case Diagram 48](#_Toc135944768)

[5.1.2. Sequence Diagram 48](#_Toc135944769)

[6.3 . Applicant / Candidate 48](#_Toc135944770)

[Chapter 7: Conclusion and Future Works 49](#_Toc135944771)

[7.1. Conclusion 49](#_Toc135944772)

[7.2. Future Works 49](#_Toc135944773)

[References 50](#_Toc135944774)

[Appendix B – React JS Installation and Setup 53](#_Toc135944775)

[Appendix C – Libraries Used in Project 55](#_Toc135944776)

[● Bootstrap 55](#_Toc135944777)

[ React 55](#_Toc135944778)

[ Axios 55](#_Toc135944779)

[ React-Router-Dom 55](#_Toc135944780)

**Table of Figures**

[Figure 2.1. HTML Page 5](#_bookmark11)

[Figure 2.2. CSS Code 6](#_bookmark13)

[Figure 2.3. Salient Features of React JS 8](#_bookmark18)

[Figure 2.4. Collections in MongoDB 9](#_bookmark20)

[Figure 3.1. Intended Use of the System 10](#_bookmark25)

[Figure 4.1. Employee Entity 19](#_bookmark60)

[Figure 4.2. Course Entity 20](#_bookmark61)

[Figure 4.3. Workload Entity 20](#_bookmark62)

[Figure 4.4. Payment Entity 21](#_bookmark63)

[Figure 4.5. ER Diagram of the System 22](#_bookmark65)

[Figure 4.6. User Interface Structure 23](#_bookmark67)

[Figure 4.7. Home Page 25](#_bookmark75)

[Figure 4.8. Employee Page 26](#_bookmark77)

[Figure 4.9. Add Employee Page 26](#_bookmark79)

[Figure 4.10. Edit Employee Page 27](#_bookmark81)

[Figure 4.11. Course Page 28](#_bookmark84)

[Figure 4.12. Add Course Page 28](#_bookmark85)

[Figure 4.13. Edit Course Page 29](#_bookmark88)

[Figure 4.14(a) Workload Page 30](#_bookmark89)

[Figure 4.14(b) Workload Page 31](#_bookmark90)

[Figure 4.15. Add Workload Page 32](#_bookmark92)

[Figure 4.16. Edit Workload Page 33](#_bookmark95)

[Figure 4.17. Payment Page 34](#_bookmark97)

[Figure 4.18(a) Report Page 35](#_bookmark98)

[Figure 4. 18(b) Report Page 35](#_bookmark99)

[Figure 4.18(c) Report Page 36](#_bookmark100)

[Figure 4.18(d) Report Page 36](#_bookmark101)

[Figure 4.19(a) Configuration Page 37](#_bookmark103)

[Figure 4.19(b) Configuration Page 38](#_bookmark104)

[Figure 4.19(c) Configuration Page 39](#_bookmark105)

# Abstract

The Job Recruitment Portal is a comprehensive web-based platform designed to help employers and applicants streamline the job recruitment process. The portal aims to connect job seekers and employers through an efficient and user-friendly interface, facilitating seamless job search, application, and selection procedures. The administration, employers, and applicants are the three main actors in the system. As the super user, the admin has the authority to manage the database, monitor system statistics, and perform other administrative tasks. Employers can post job openings, review applicant profiles, select or reject applicants, and download resumes as well as contact applicants through the in-built mailbox. Applicants, on the other hand, can search available job listings, apply for relevant positions, and showcase their qualifications through resume submissions. The Job Recruitment Portal aims to improve the efficiency and effectiveness of the job recruitment process for both employers and applicants by providing a centralized platform for job posting and application management. To ensure a robust and password protected secure system, the project employs a number of technologies and frameworks. We investigate the design, development, and evaluation of the Job Recruitment Portal, as well as its performance and potential future enhancements, in this thesis. The outcomes demonstrate the system's effectiveness in facilitating seamless job recruitment procedures and improving the overall experience for both employers and applicants.

# Chapter 1: Introduction

The Job Recruitment Portal aims to revolutionize the job recruitment process by providing employers and applicants with a user-friendly and efficient platform. Traditional methods of recruitment frequently result in delays, inefficiencies, and missed opportunities. This project addresses these issues by providing a centralized system with three primary actors: administrator, employers, and applicants. The database and system statistics are managed by the administrator, while employers can post jobs, review applicants, and download resumes. Applicants can look for jobs and apply for them. The portal makes use of technologies and frameworks to ensure a stable system. The focus of evaluation is on performance, usability, and user satisfaction. The findings contribute to the field of job recruitment and provide guidance for future improvements. The goal is to improve the overall experience for employers and applicants by streamlining the recruitment process, saving time and effort.

## Problem Definition

Traditional recruitment methods frequently face a number of challenges that impede the process's efficiency and effectiveness.

* Handling a large volume of job applications manually is time-consuming and labor-intensive, resulting in delays and potential oversight of qualified candidates.
* Job seekers struggle to find relevant opportunities among numerous sources, resulting in potential positions being missed.
* The lack of standardized formats for resumes and job postings makes objective candidate evaluation and decision-making difficult.
* Without a centralized system, employers struggle to effectively track and manage applications.
* Limited Reach and Narrow Candidate Pool: Conventional techniques, such as newspaper ads or physical job postings, may only reach a narrow pool of individuals. Due to this, there may be a lack of diversity and skilled applicants who are not actively looking for work through conventional routes may go unnoticed.
* High costs: Conventional hiring practices can be pricey, including participating in job fairs and newspaper advertising. Advertising, travel, and other associated costs may require organizations to set aside large sums. This can be difficult, especially for startups and small businesses with scarce resources.
* The management of applications, paperwork, and communication can be inefficient and prone to mistakes when done manually. Traditional techniques frequently lack automated components.

## Proposed System

The solution for this is a modern “Job Recruitment Portal”, As the world moves on to a digital era, searching for Jobs or finding the right Employee for one, like all other aspects of daily life, are being shifted to be online, saving time, human resources, accessible and providing ease of use. We intended to create a platform that will automate the process where organizations can put up job vacancies, where applicants can go through them and apply in the right job.

Where employers would have the ease of checking through CV’s and selecting the suitable candidates etc. And applicants will find a platform that will allow them to apply in multiple jobs according to their qualifications easily as well as get side-by-side updates.

It is web-based platform, with a good UI/UX having a Landing page, Job Vacancies Page from where one can apply into preferred job after registering with the system as an applicant where email validation shall be done, then provide data and upload their CV and apply for jobs they want, they can also search by the search bar and apply search filters for their needs like locations, experience requirements etc. Employers would have access to a job posting page where they set the requirements and the details of the job like title, salary, details etc. This system has a database for all records of jobs i.e. Job Titles (with details), Applications (CV attached, Data provided) as well as details of users such as employers, and most importantly the applicants like their Names, CNIC, Addresses, Qualifications, and so on.

To secure all this data and avoid mishaps, security measures will be taken starting off with proper Authorization being implemented like access-controls over user types, where an applicant can only view, update his/her own data rather than having access of others, the employer only accessing data of applicants who have applied to his posted job, Admin having access of everything including the database etc.

Applicants/Candidates can apply in one or more jobs with easy without entering same data (name, CNIC, Addresses etc.) again for each job they apply. On the employers end they can check all the applications to a certain job, contact applicants regarding concerns through in-built communication channel and download CV’s and data of applicants for further processing if needed.

## 1.3 System Functions

The following are key system functions:

* **User Registration and Login Authentication:**

Employers and Applicants can register and create accounts on the portal, that are password protected and encrypted in database, hence even admin cannot see them.

User login authentication ensures secure system access and protects user data.

* **Job Posting and Management:**

Employers can create and post job openings, describing the job's details and requirements.

As needed, job listings can be edited, updated, or removed.

* **Filtering and Job Search:**

Applicants can use specific criteria to search for available jobs. Filtering options allow applicants to refine their search results and find relevant job opportunities.

* **Management of Applicant Profiles:**

Employers can review applicant profiles, including resumes, can establish contact if they have applied to any job posting by that employer.

* **Candidate Selection:**

Based on their suitability for the job, employers can mark applicants as selected / rejected.

Selected candidates can be contacted for interviews, assessments, or other stages of the selection process through the in-built mailbox.

* **Download Resume/CV:**

Employers can download applicants' resumes or CVs for offline review or further evaluation.

* **Management and the Admin Dashboard:**

The system administrator has access to a comprehensive dashboard for system management.

Admin functions include approval of employer account, managing users, databases, and generating system statistics. Admin reserves rights to delete any posted job, or employer account as well from here.

* **In Built Mailbox:**

The in-built mailbox offers applicants and employers to communicate any misunderstandings or important notices and details.

* **Change Password / Deactivate Accounts:**

Employers and applicants can both use this option to modify their passwords inside the portal. To ensure account security, users can reset their passwords and access their account settings. Users are also able to deactivate their accounts within the portal.

# Chapter 2: Literature and Tech Review

Chapter 2 contains many basic concepts, technologies and frameworks that are significant in the conduct and development of the project titled “*Job Recruitment Portal”.*

## Management Portal

Information systems are large, linked collections of information, data and processes. They are employed in virtually every element of human life, business, and industry.

The phrase "information system" in technology refers to any tool or information system that facilitates the gathering and utilization of data. Information systems can be utilized to give aid inside a company or for personal benefit.

A good information system enables the user to quickly access, comprehend, and react to information. Users get access to the most up-to-date information whenever they need it to complete a task.

## Programming Languages and Frameworks

In this section, different technologies and tools that can be used to develop this information system are discussed.

### HTML

A common markup language for building and organizing web pages is HTML (Hypertext Markup Language). It offers a collection of tags that specify the organization and display of content on the internet. Text, images, links, forms, and other elements can all be included in HTML and displayed in a web browser. It offers a logical, hierarchical structure that enables web designers to efficiently organize and format the content. Developers can use HTML to build semantically sound web pages that are compatible with assistive technology and search engines. Along with JavaScript and CSS (Cascading Style Sheets), HTML is a core language used in modern web development. All of the main web browsers support it, ensuring cross-platform compatibility.

### CSS

CSS is a style sheet language that is used to describe the presentation and formatting of HTML and XML documents. It provides a set of rules and properties that control how web pages look visually. CSS allows web designers to separate a webpage's content from its presentation, providing greater flexibility and consistency across multiple pages. Designers can define various aspects such as colors, fonts, layout, and animations by applying CSS to HTML elements. CSS encourages efficient web development by allowing the reuse of styles across different pages, making it easier to maintain and update a website's overall design. It works with HTML to create visually appealing and responsive web pages. Web designers have precise control over the aesthetics of their websites with CSS, which improves the user experience and overall presentation of content.

### JavaScript

JS (JavaScript) is an interpreted high-level programming language that is primarily used to add interactivity and dynamic behavior to web pages. All modern web browsers support it, allowing developers to create interactive elements, perform calculations, manipulate web page content, and respond to user actions in real-time. JavaScript can be embedded directly within HTML documents or included as separate files, improving website functionality and user experience. It includes event handling, DOM manipulation, AJAX for asynchronous communication, and the ability to create interactive web applications. It is a versatile language that can be used in server-side environments, mobile app development, and other software projects in addition to web development.

### Bootstrap

Bootstrap is a popular front-end framework for developing responsive and mobile-first websites. Bootstrap, created by Twitter, is a collection of pre-designed templates, CSS styles, and JavaScript components that make it easier to create modern and visually appealing web pages. It uses a grid-based layout system, which allows developers to easily arrange and align page elements. Bootstrap also includes a plethora of reusable user interface components, such as navigation bars, buttons, forms, modals, and carousels, that can be easily customized and integrated into web projects. Because it is built with HTML, CSS, and JavaScript, it is compatible with all modern web browsers. Bootstrap provides developers with a time-saving and efficient solution for creating responsive and professional-looking websites, thanks to its extensive documentation and active community support.

### PHP

PHP (Hypertext Preprocessor) is a popular server-side scripting language that is used in web development. It is primarily used for the development of dynamic and interactive web pages and web applications. PHP code is embedded directly into HTML, allowing developers to seamlessly mix logic and presentation. PHP is a versatile programming language that can interact with databases, handle form data, generate dynamic content, and perform a variety of server-side tasks. It is supported by the majority of web servers and runs on a variety of operating systems, making it extremely accessible. PHP includes a large number of built-in functions and libraries that help to simplify common web development tasks. PHP allows developers to create robust and scalable web applications that power some of the internet's most popular websites.

### MySQL

MySQL is a relational database management system (RDBMS) that is free and open source that allows for the efficient storage, management, and retrieval of structured data. MySQL employs a client-server architecture, in which the server manages and stores data, while clients (such as applications or websites) interact with the server to perform various database operations. MySQL provides a robust and scalable solution for data-driven applications by supporting SQL (Structured Query Language) for querying and manipulating data. It includes data security, transaction support, and support for a variety of data types and indexing mechanisms. MySQL is well-known for its speed, dependability, and simplicity of use, making it a popular choice for both small-scale projects and enterprise-level applications.

## Software’s and Technologies Used

In this section, different technologies and tools that can be used to develop this portal are discussed.

### XAMPP

XAMPP is a free and open-source software package that includes everything needed to set up a local web server environment. "XAMPP" is an abbreviation for cross-platform (X), Apache (A), MariaDB (M), PHP (P), and Perl (P). It contains all of the components required to run dynamic web applications on a local machine, making it an excellent choice for developers and web designers. XAMPP is a pre-configured package that includes the Apache web server, the MariaDB (formerly MySQL) database management system, the PHP scripting language, and the Perl programming language. It also includes tools for database administration, such as phpMyAdmin, and a file transfer server, FileZilla. XAMPP works with a variety of operating systems, including Windows, macOS, and Linux. It makes it easier to set up a local development environment, allowing users to test and debug web projects before deploying them to a live server. Because of its ease of installation, comprehensive feature set, and community support, XAMPP is widely used by both developers and beginners.

### Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) that offers a comprehensive set of tools and features for software development. It supports a wide range of programming languages and frameworks, including C#, C++, Visual Basic, and.NET. Visual Studio is an easy-to-use and highly productive environment for developing, debugging, and deploying applications across multiple platforms and devices. It has code editors, project management tools, integrated debugging capabilities, and a large collection of libraries and extensions. Visual Studio also integrates seamlessly with source control systems, allowing developers to effectively collaborate on projects. Visual Studio has become a go-to choice for professional developers and students alike, thanks to its powerful features, extensive documentation, and supportive community. Visual Studio provides the tools and resources needed to streamline the development process and deliver high-quality software, whether you're creating desktop applications, web applications, mobile apps, or cloud-based solutions.

### Apache

If Apache, also known as Apache HTTP Server, is a popular open-source web server. It is one of the most widely used and long-lasting web server solutions available. Because of its flexibility, stability, and performance, Apache is a popular choice for hosting websites and serving web content. It works with a variety of programming languages and frameworks and supports multiple operating systems, including Windows, Linux, and macOS. URL rewriting, SSL/TLS encryption, virtual hosting, and load balancing are just a few of the features and modules available to enhance Apache's functionality. It has strong security measures and customizable configuration options, allowing administrators to tailor the server to their specific requirements. Because of the extensive community support and documentation provided by Apache, it is accessible to both novice and experienced users. Overall, Apache is a dependable and feature-rich web server software that has remained a mainstay of the web hosting industry.

### Github

GitHub is a web-based platform that provides hosting for version control repositories. It enables developers to effectively collaborate on projects, track changes, and manage source code. Developers can use GitHub to create repositories to store their code and easily share and collaborate with others by inviting contributors to join their projects. The platform includes features like branching and merging that allow developers to work on different versions of the code at the same time and seamlessly merge their changes. GitHub also has an issue tracking system, which allows developers to report and track bugs or feature requests within a project. It also includes a robust code review system that allows peers to review and provide feedback on each other's code changes. GitHub hosts millions of public repositories as an open-source community, making it a valuable resource for discovering and contributing to a wide range of projects. It has evolved into a necessary tool in modern software development, encouraging collaboration, code sharing, and community-driven innovation.

### phpMyAdmin

phpMyAdmin is a graphical user interface (GUI) tool for managing and administering MySQL databases that is accessible via the web. It has an easy-to-use interface that allows users to perform database operations like creating and managing databases, tables, and indexes, running SQL queries, importing and exporting data, and managing user privileges. phpMyAdmin makes it easier to manage MySQL databases, especially for users who are unfamiliar with the command-line interface. Users can easily interact with their databases thanks to its user-friendly interface, making it a valuable tool for web developers, database administrators, and system administrators. phpMyAdmin is written in PHP and supports multiple languages, allowing users all over the world to have a localized experience. It is a widely used tool because of its ease of use, robust features, and active community support. phpMyAdmin is a convenient and powerful solution for managing MySQL databases, whether used for small-scale projects or large-scale enterprise applications.

### 000WebHost

000webhost is a free web hosting platform that allows users to host their websites at no cost. It has an easy-to-use interface and a variety of features that make website building and management easier. Users can enjoy free web hosting services with 000webhost without the requirement to invest in a dedicated hosting solution. Users may create dynamic and interactive websites because to the platform's support for PHP and MySQL. It includes a website builder and is compatible with popular content management systems (CMS) such as WordPress. While 000webhost is free, it does have some restrictions, like restricted storage space, bandwidth, and server resources. phpMyAdmin is an easy and powerful tool for administering MySQL databases, whether used for small-scale projects or large-scale enterprise applications.

# Chapter 3: Software Design & Architecture

## 3.1. System Overview

The purpose of the project is to develop a Job Recruitment system that provides all the necessary information or details of jobs available, allowing applicants to opt those and employers to check resumes and select / reject application.

## Functional Requirements

### 3.2.1. Users / Applicant

1. Registration: Applicants should be able to create an account by providing their personal details and creating login credentials.

2. Login: Applicants should have the ability to log into their accounts using their registered email and password.

3. Forgot Password: Applicants should have an option to recover their forgotten password through a password reset process.

4. Update Profile: Applicants should be able to edit and update their profile information, including personal details, work experience, education, and skills.

5. View Jobs: Applicants should have access to a list of job openings, with the ability to search, filter, and browse through available positions.

6. Apply for Job: Applicants should be able to submit their applications for specific job listings, including attaching their resumes and any additional required documents.

7. Upload Resume: Applicants should have the ability to upload their resumes or CVs to their profiles for employers/recruiters to review during the application process.

### 3.2.2. Employers

1. Registration: Employees should be able to create an account on the job portal by providing necessary personal information and contact details.

2. Login: Employees should have the ability to securely log into their accounts using their registered credentials.

3. Forget password option: Employees should have the option to recover/reset their forgotten passwords through a secure process.

4. Select/reject: Employees should be able to select or reject job applicants based on their qualifications and suitability for the position.

5. Add Job posts: Employees should have the capability to create and post job openings with relevant details such as job title, description, requirements, and application instructions.

6. Delete Job posts: Employees should be able to remove job posts from the portal once the position has been filled or is no longer available.

7. Search, view, or download resume of applicants: Employees should be able to search for specific job applicants, view their resumes, and have the option to download or save them for further evaluation.

## Non Functional Requirements

### Security

To ensure confidentiality and prevent unauthorized access, the system should put appropriate security safeguards in place, such as encryption, secure authentication, and access controls.

### 2. Availability

The job portal should always be available and functional, offering users uninterrupted service with minimal downtime.

### 3. Scalability

Without noticeably degrading in performance, the system should be able to handle an increasing number of users, job postings, and resumes.

### 4. Performance

Even with high traffic and large records, the portal should offer a quick and responsive user experience, with quick loading times for pages and search results.

### 5. Usability

The portal's interface should be simple and easy to use, making it simple for both employers and job seekers to navigate, look for jobs, and manage profiles and applications.

## System Architecture

A Job Recruitment Portal's system architecture is client-server, with various components interacting to provide the desired functionality. A high-level overview of the system architecture is provided below:

### 3.4.1. Client-Side Elements

* UI (User Interface): This component represents the portal's frontend and is in charge of displaying the user interface to employers and applicants. To ensure a responsive and interactive experience, it is typically built with HTML, CSS, and JavaScript frameworks.

### 3.4.2. Components for the server

* The web server handles incoming client requests and serves the appropriate web pages. It handles communication between clients and backend services.
* The application server is where the Job Recruitment Portal's core business logic resides. It manages job listings, applicant data, and performs various operations based on user actions. It communicates with the database server in order to retrieve and store data.
* The database server stores and manages data from the portal, such as user information, job listings, applicant profiles, and application data.

### 3.4.3. Infrastructure Elements

* Networking: This component ensures that clients, web servers, application servers, and database servers are all connected. It is in charge of routing requests and responses across the network.
* Security: Security components are in charge of authentication, authorization, encryption, and secure communication protocols in order to protect user data and maintain system integrity.
* Storage: To store resumes, the system architecture includes storage components such as file servers or cloud storage solutions.

### 3.4.4. Application Layer

* Contains the application's main logic for the job portal.
* It handles request processing, authentication, and business logic and is written in PHP.
* Interacts with the database layer to retrieve and modify data

### 3.4.5. Database Layer

* Manages data retrieval and storage.
* Uses a relational database management system (like MySQL) to keep track of user data, resumes, and other job-related information.

## Database Design

Our Web Application needs a database to store, organize, and manage data efficiently. The data can be of different types, including user information, Job information and so on.

### 3.5.1. Data Storage

A database provides a centralized location to store data for a web app. It allows data to be easily managed and retrieved by the application.

### 3.5.2. Data Security

Databases can provide security measures to protect sensitive information. It can limit access to specific data and protect against data breaches.

### 3.5.3. Data Organization

Databases allow data to be organized in a structured manner, making it easy to retrieve data based on specific criteria.

### 3.5.4. Data Scalability

Databases can handle large amounts of data, making it easy to scale up or down the application as needed.

### 3.5.5. Data Persistence

Databases ensure that data is saved even if the application is shut down or restarted. This makes it possible to continue from where the application left off.

Overall, a database is a crucial component of our web app, and it allows the application to store and manage data efficiently and securely. In order for a database we need tables that store the data for the portal, the tables created are as follows:

* admin
* apply\_job\_post
* cities
* company
* countries
* job\_post
* mailbox
* reply\_mailbox
* states
* users

## Scope

### Features Falling Within Scope

Here are some features that fall within the scope / covered by a job portal:

* + - 1. Job Listings: A job portal's main purpose is to offer a platform for employers to post open positions, and for job seekers to look for and apply to those positions. Employers should be able to specify job details, requirements, and application processes through the portal, which should also provide job seekers with search and filtering options.
      2. Candidate profiles: Typically, job portals have tools that allow applicants to create and manage their profiles. This could entail managing job applications, uploading resumes, providing personal and professional information, emphasizing skills and qualifications, etc.
      3. Employer Profiles: On the job portal, employers should be able to create and manage their profiles. This includes disseminating company information, publishing job listings, managing and reviewing application materials, and corresponding with applicants.
      4. Search and Filtering: Job seekers should have the option to search for positions based on a variety of factors, including location, industry, job type, level of experience, and keywords. The search results can be trimmed down to better match particular preferences using advanced filtering options.
      5. Application Management: The job portal should make it easier for both employers and job seekers to manage job applications. Reviewing, shortlisting, and corresponding with applicants should be possible for employers. Job seekers ought to be informed of the status of their applications and have access to their application history.
      6. Communication and Messaging: The portal might have messaging features that make it easier for job seekers and employers to communicate. Direct messaging, email alerts, or integration with other communication channels may be used for this.
      7. Resume Database: Some job portals give employers access to a database of resumes from candidates who have chosen to make their profiles searchable. Employers can proactively look for candidates thanks to this.

### Scope in Real World Use

They offer a central hub for connecting job seekers and employers, making it simpler for employers to find qualified candidates and for job seekers to research job opportunities, and make job openings more visible to both employers and job seekers. Wider audiences can be reached by employers, improving the likelihood of luring in qualified candidates.

It can also be used as follows:

1. Public job portals: Easily accessible job listings and open government hiring.

2. Private Job Portals: With a centralized recruitment platform, businesses can streamline their hiring processes.

3. Industry-Specific Portals: These websites link job seekers and employers in specialized fields.

4. University portals: Alumni and students can access internships and jobs.

5. International job opportunities are listed on global job portals.

6. Freelance Job Portals: These sites help match independent contractors with clients for project-based work.

# Chapter 4: Implementation

## Technology & Tools Used

The technologies and frameworks that are used to develop system are;

* + 1. Bootstrap
    2. PHP
    3. MySQL
    4. Apache
    5. Xampp
    6. Visual Studio
    7. Brackets

## Front End Development

The process of developing a website's user interface and interactive elements is known as front-end development.

1. The common markup language used to organize the content of web pages is HTML. It outlines the page's structure and elements, including headings, paragraphs, images, forms, and more.

2. A stylesheet language called CSS is used to describe the presentation and styling of an HTML document. It gives you the ability to manage how elements, such as colors, fonts, layouts, and responsive design, appear.

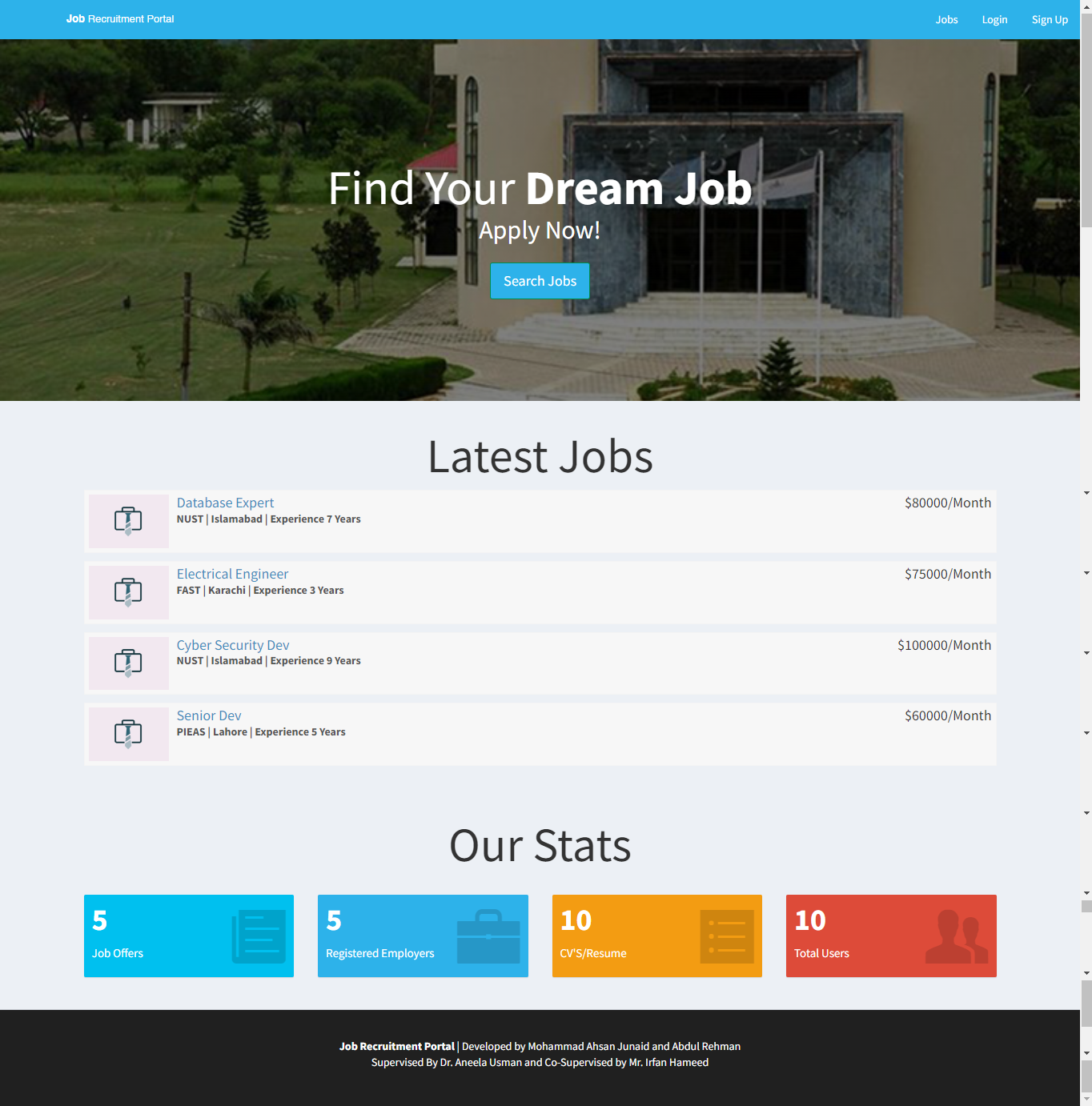
3. JavaScript is a web-based programming language that gives web-pages interactivity and dynamic functionality. You can work with HTML elements, manage events, send AJAX requests, and make animations.

.

### Home Page

The following features would be displayed on the home page's output:

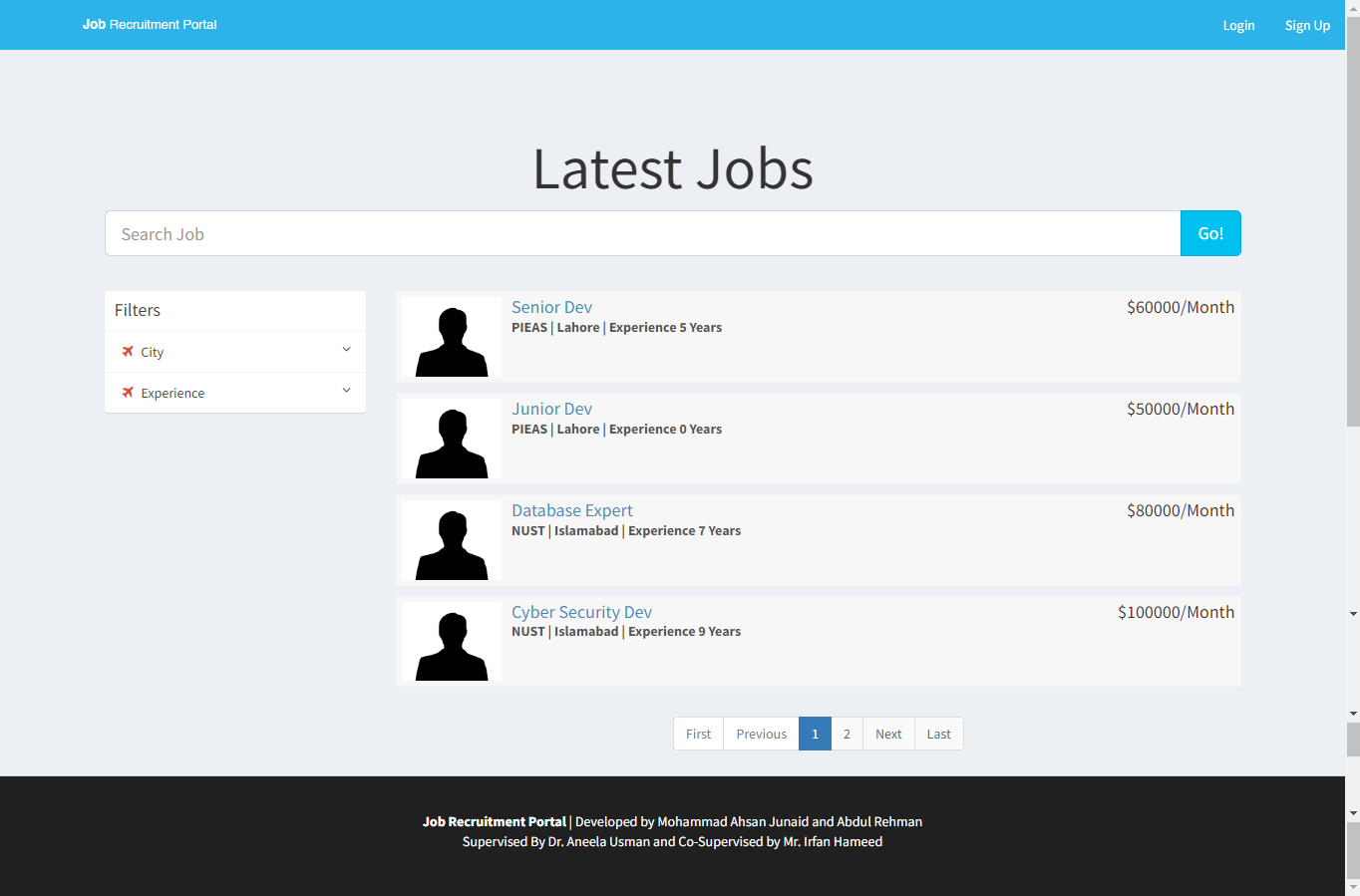
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.
* Latest Jobs section:  
  Shows details about the newest job listings. The job title, employer name, location, level of experience necessary, and maximum salary is all listed. For up to four job listings, the data is dynamically retrieved from the database and displayed in a loop.
* Stats section:  
  statistical data regarding the job portal. It lists the total number of positions available, employers who have registered, CVs and resumes, and users overall. Additionally, dynamically retrieved from the database are these statistics.



### Job Posting Page

The following features would be displayed on the page's output:

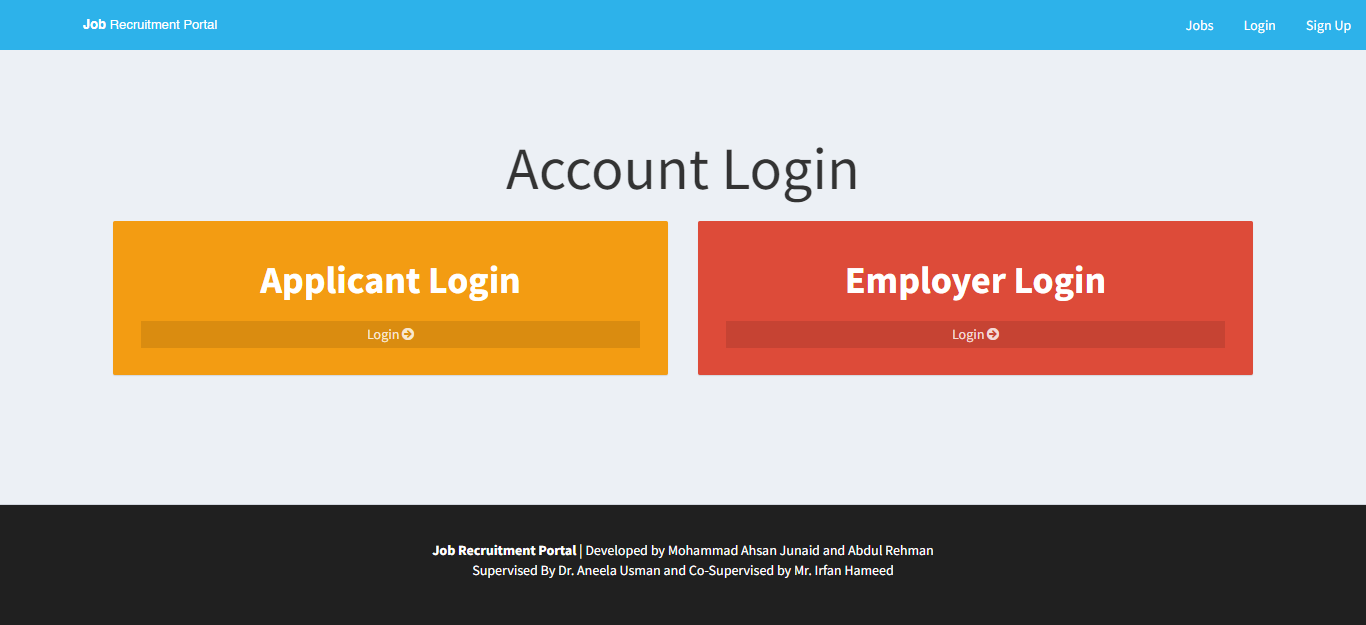
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.
* Search Jobs:  
  To search for jobs, users can enter a search term and press the button.
* Filters Section:  
  The page has a section for filters where users can narrow down job listings by "City" and "Experience." Users may refine their search results using the options provided.
* Job Listings:  
  Based on the entered search terms or selected filters, the page dynamically loads job listings. Users can navigate through multiple pages of the job listings thanks to the pagination feature of the job listings.



### Login Page

The following features would be displayed on the page's output:

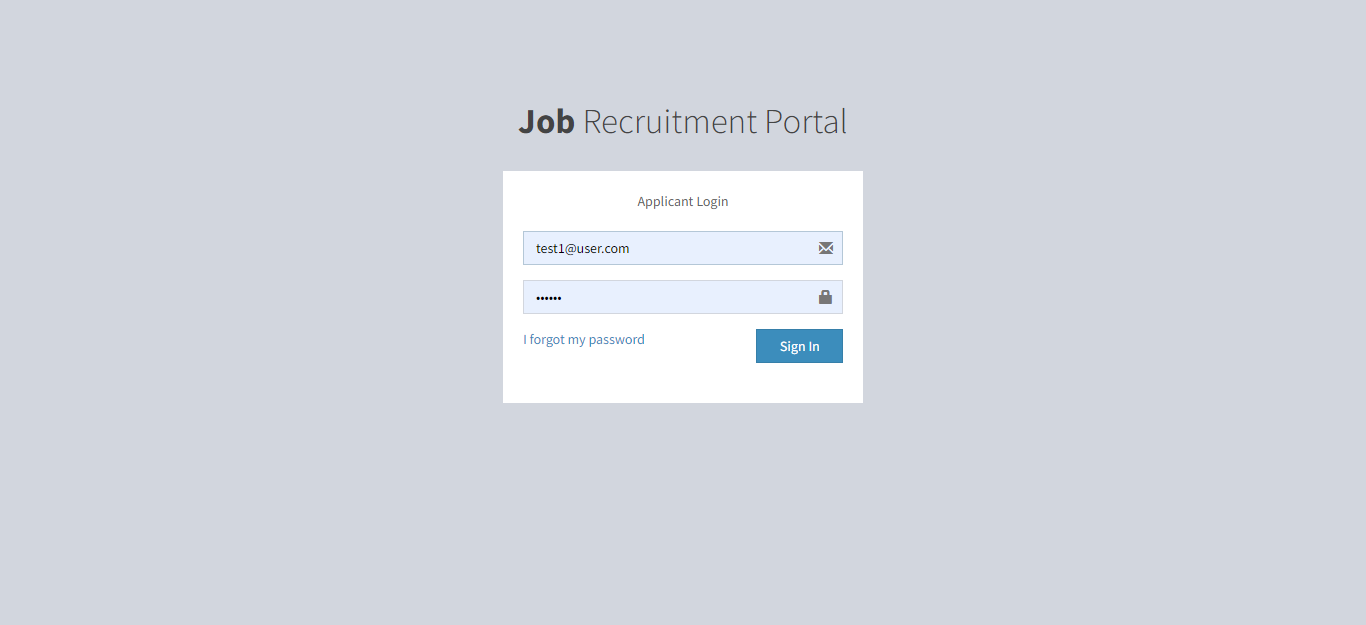
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.
* Search Jobs:  
  To search for jobs, users can enter a search term and press the button.
* Filters Section:  
  The page has a section for filters where users can narrow down job listings by "City" and "Experience." Users may refine their search results using the options provided.
* Job Listings:  
  Based on the entered search terms or selected filters, the page dynamically loads job listings. Users can navigate through multiple pages of the job listings thanks to the pagination feature of the job listings.



### Applicant Login Page

The following features would be displayed on the page's output:

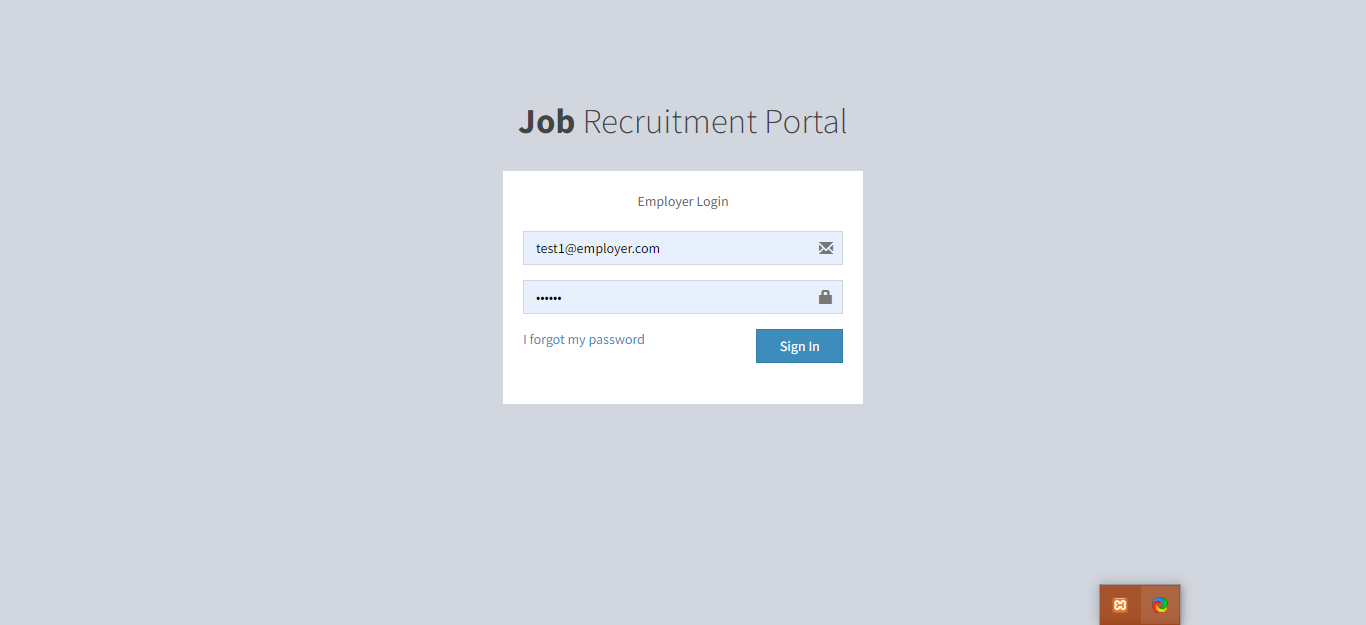
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.
* Search Jobs:  
  To search for jobs, users can enter a search term and press the button.
* Filters Section:  
  The page has a section for filters where users can narrow down job listings by "City" and "Experience." Users may refine their search results using the options provided.
* Job Listings:  
  Based on the entered search terms or selected filters, the page dynamically loads job listings. Users can navigate through multiple pages of the job listings thanks to the pagination feature of the job listings.



### Employer Login Page

The following features would be displayed on the page's output:

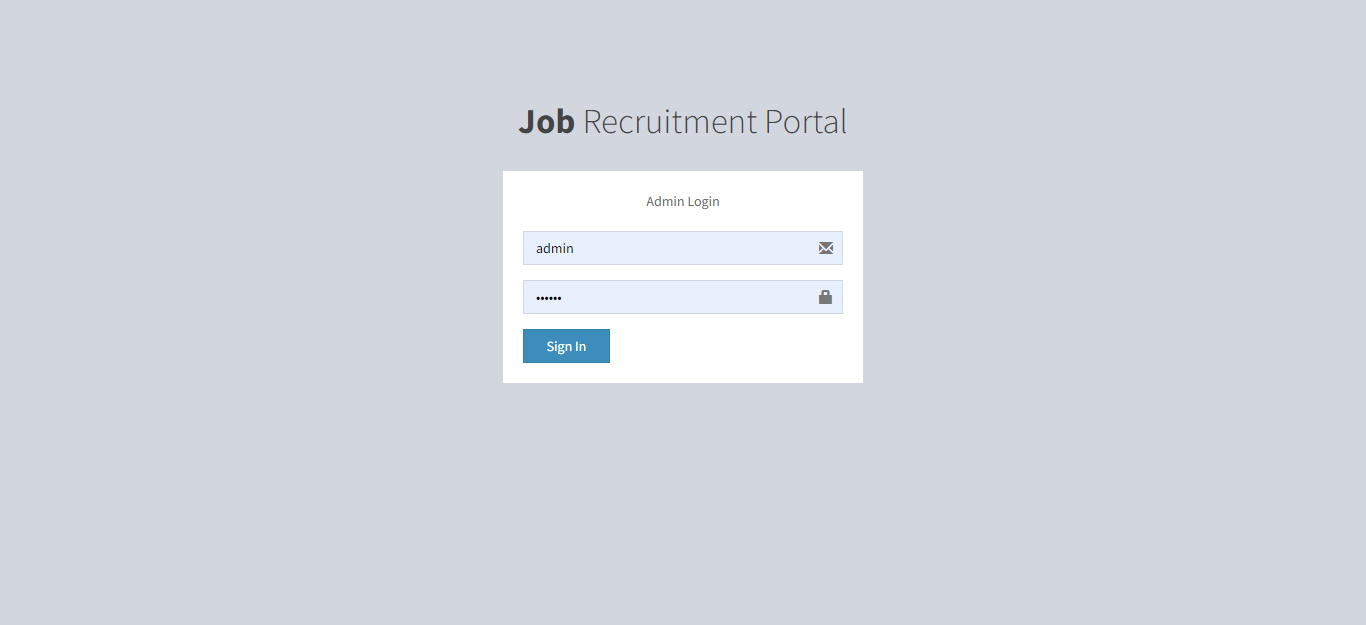
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.
* Search Jobs:  
  To search for jobs, users can enter a search term and press the button.
* Filters Section:  
  The page has a section for filters where users can narrow down job listings by "City" and "Experience." Users may refine their search results using the options provided.
* Job Listings:  
  Based on the entered search terms or selected filters, the page dynamically loads job listings. Users can navigate through multiple pages of the job listings thanks to the pagination feature of the job listings.



### Employer Login Page

The following features would be displayed on the page's output:

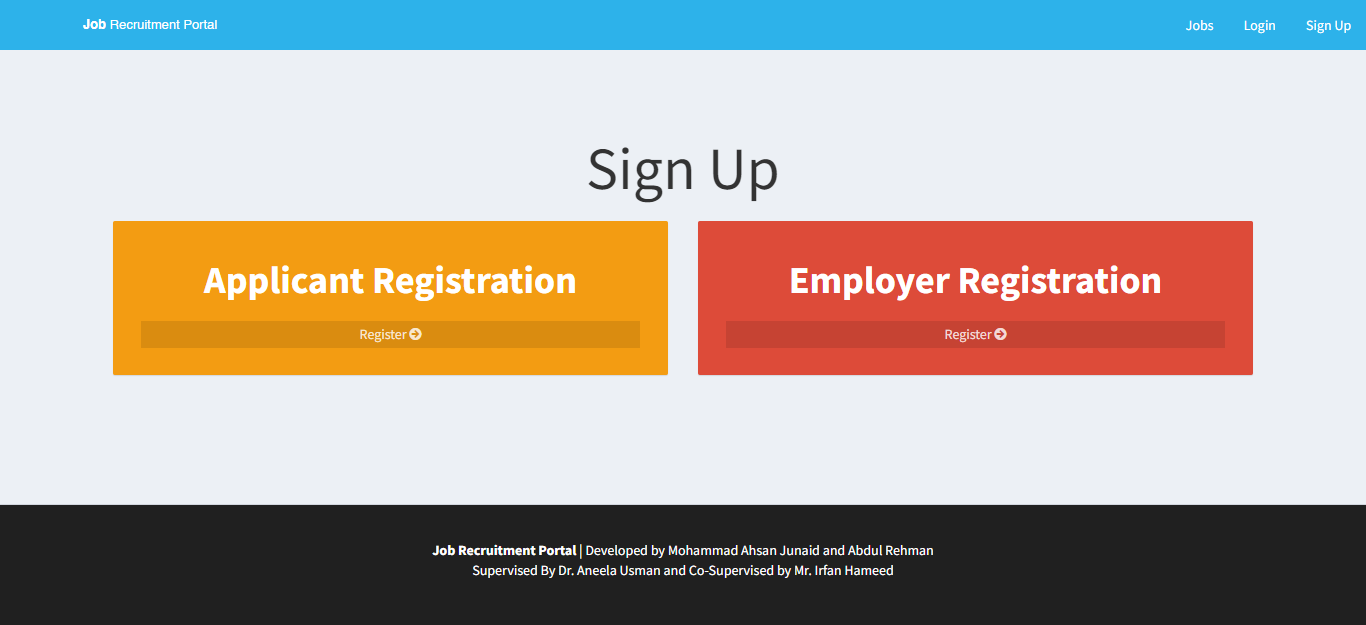
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.
* Search Jobs:  
  To search for jobs, users can enter a search term and press the button.
* Filters Section:  
  The page has a section for filters where users can narrow down job listings by "City" and "Experience." Users may refine their search results using the options provided.
* Job Listings:  
  Based on the entered search terms or selected filters, the page dynamically loads job listings. Users can navigate through multiple pages of the job listings thanks to the pagination feature of the job listings.



### Sign Up Page

The following features would be displayed on the page's output:

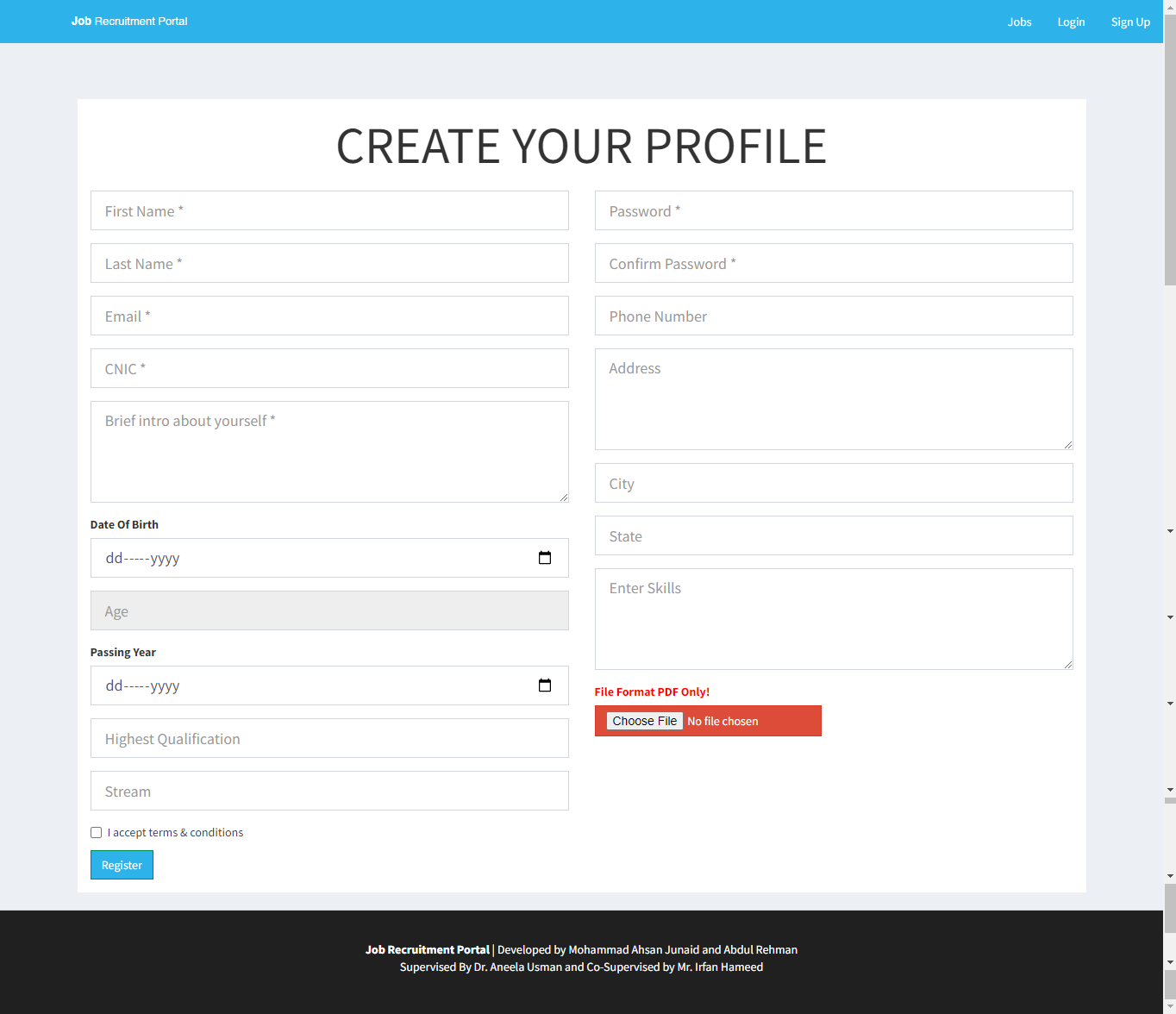
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.
* Search Jobs:  
  To search for jobs, users can enter a search term and press the button.
* Filters Section:  
  The page has a section for filters where users can narrow down job listings by "City" and "Experience." Users may refine their search results using the options provided.
* Job Listings:  
  Based on the entered search terms or selected filters, the page dynamically loads job listings. Users can navigate through multiple pages of the job listings thanks to the pagination feature of the job listings.



### Applicant Registration Page

The following features would be displayed on the page's output:

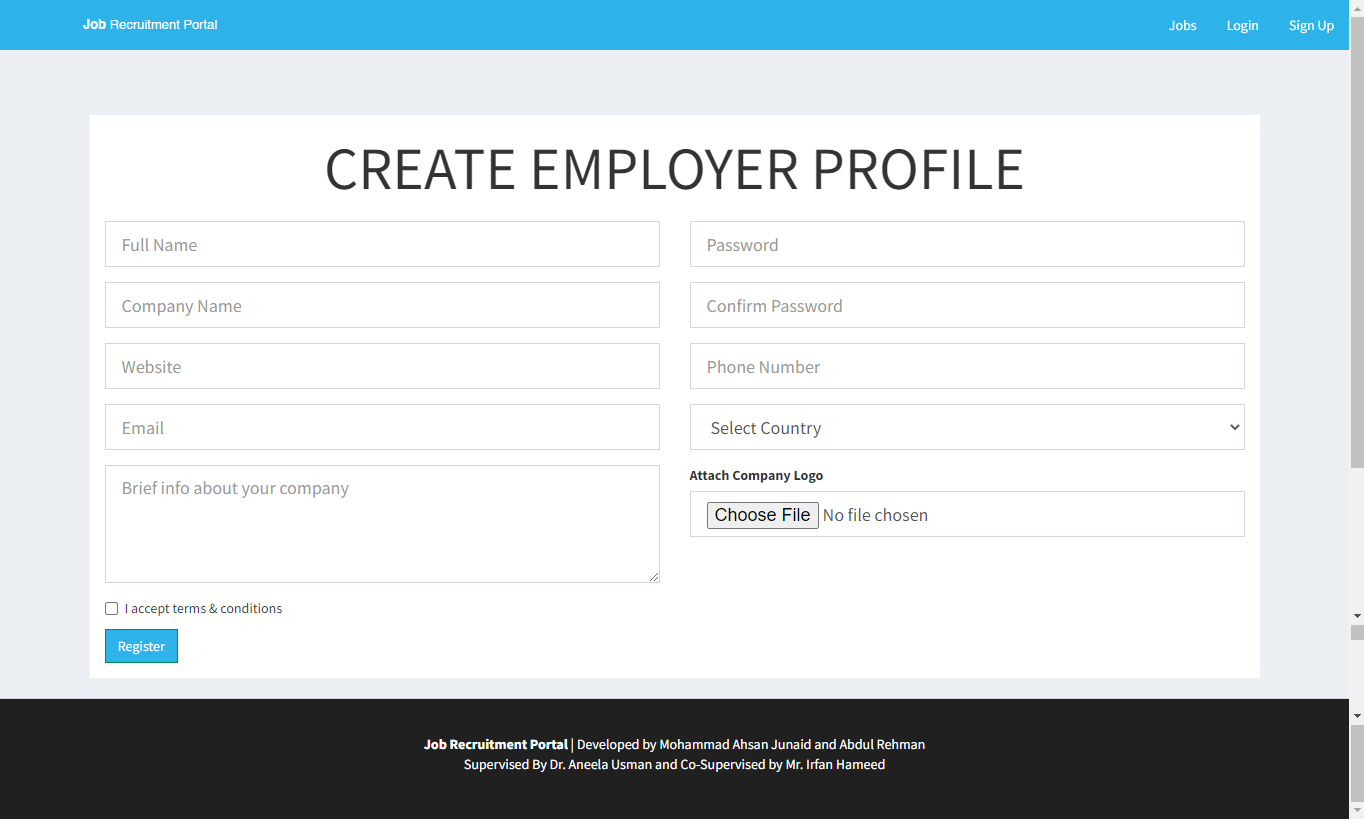
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Registration Page

The following features would be displayed on the page's output:

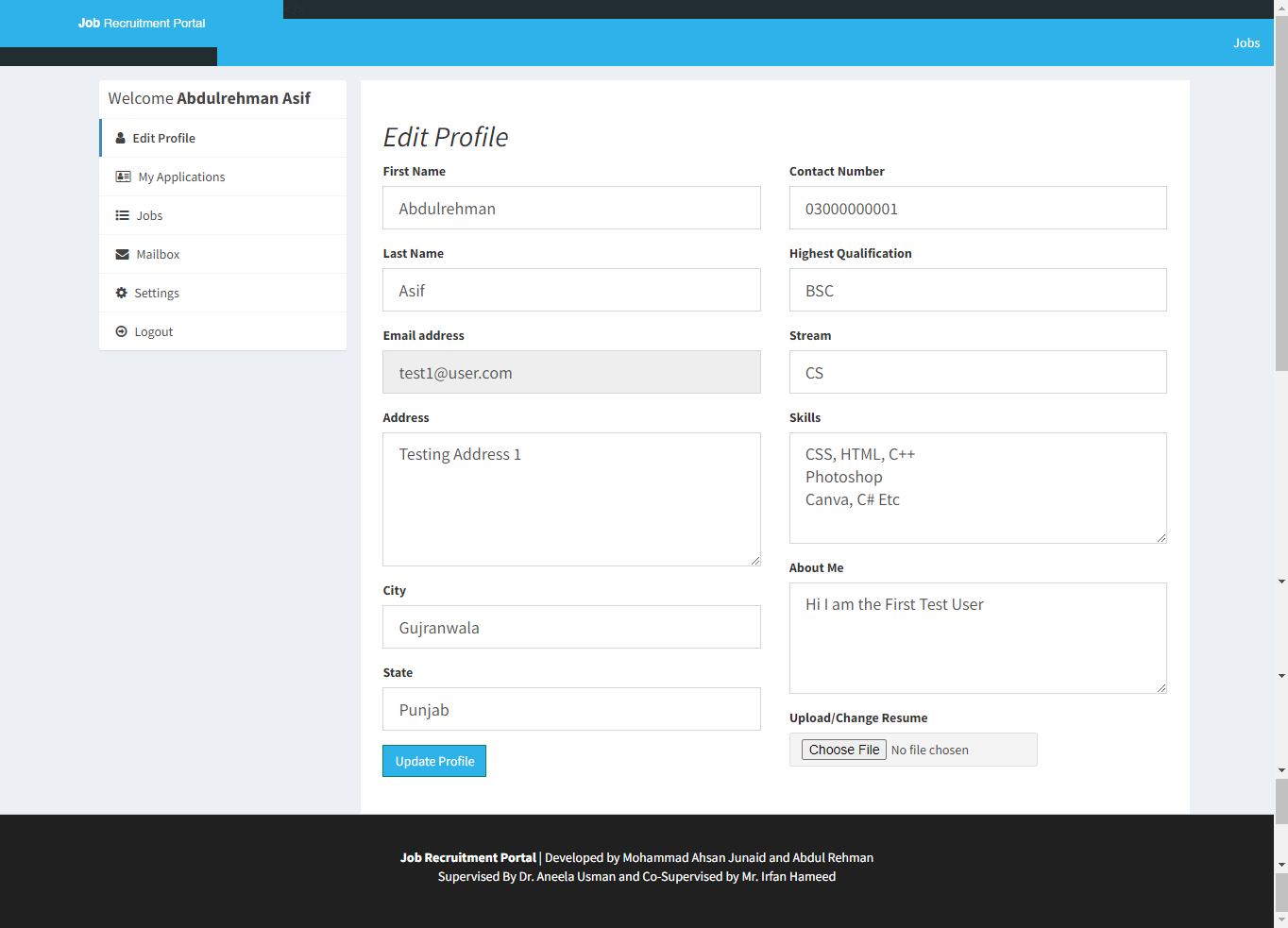
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Applicant Dashboard – Edit Profile

The following features would be displayed on the page's output:

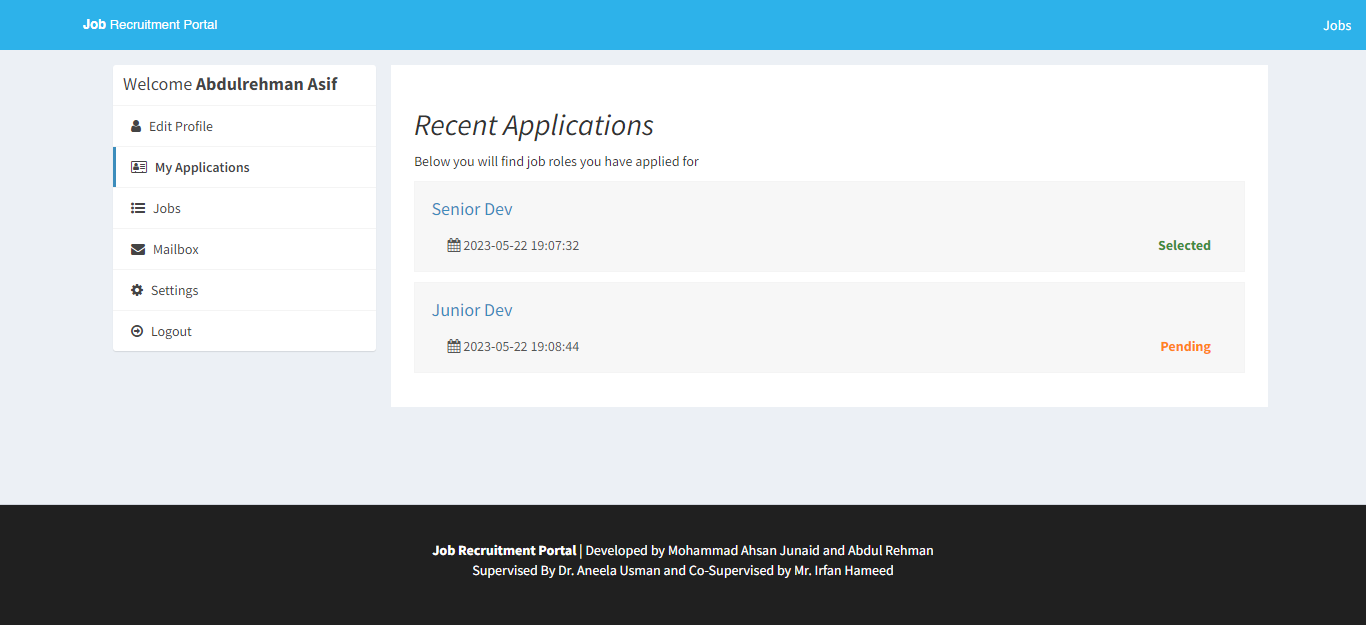
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Applicant Dashboard – Job Applications

The following features would be displayed on the page's output:

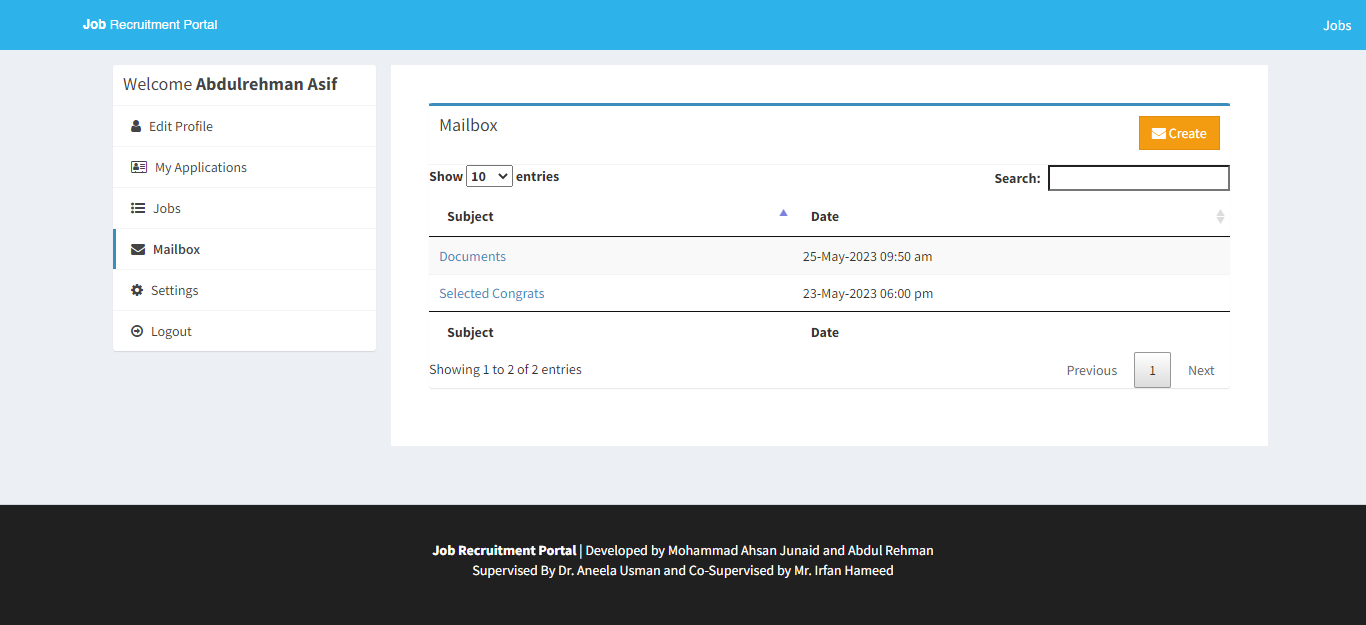
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Applicant Dashboard – Mailbox

The following features would be displayed on the page's output:

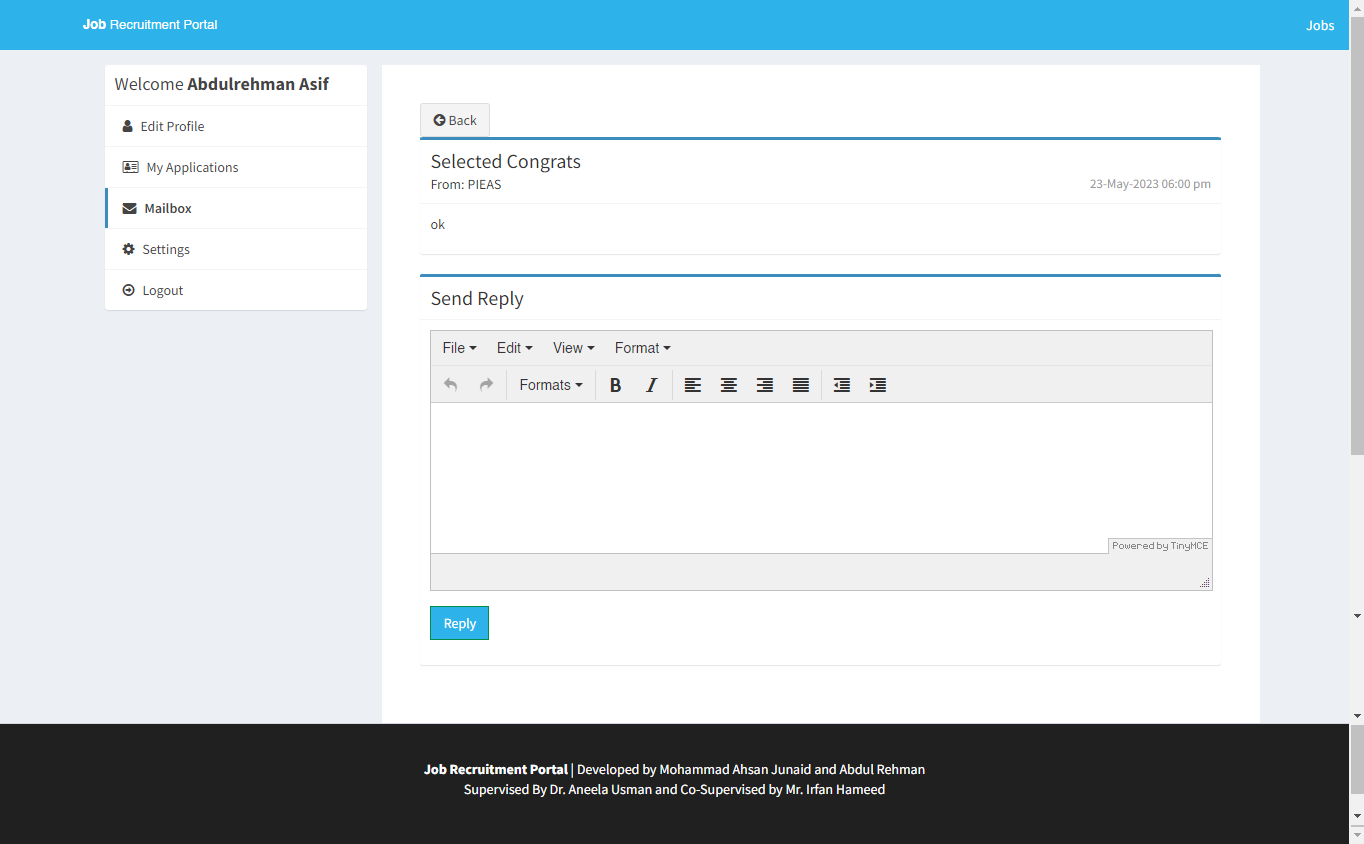
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Applicant Dashboard – Read / Reply Mail

The following features would be displayed on the page's output:

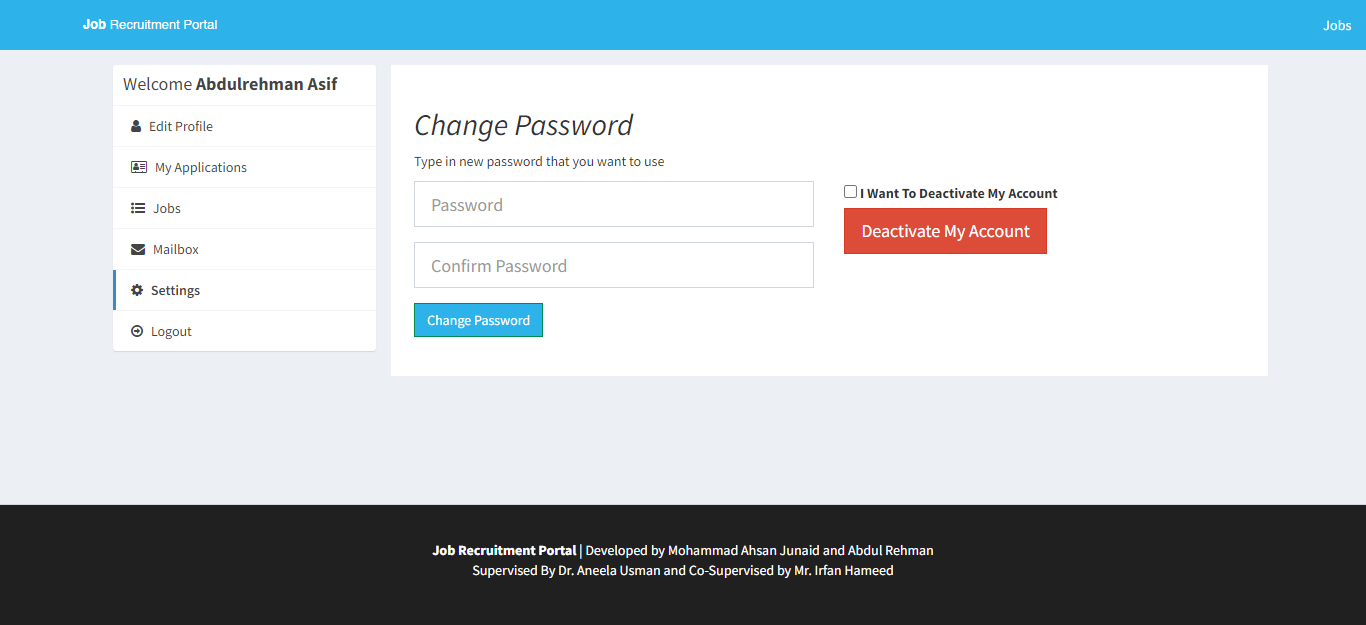
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Applicant Dashboard – Settings

The following features would be displayed on the page's output:

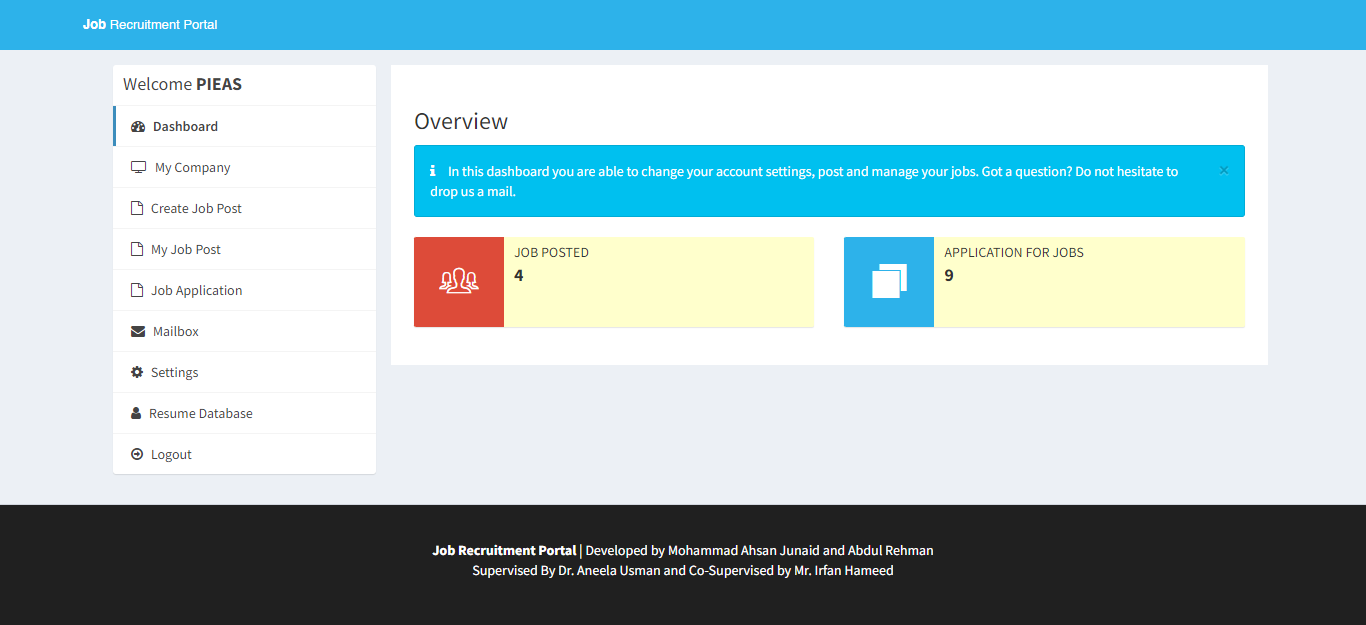
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard

The following features would be displayed on the page's output:

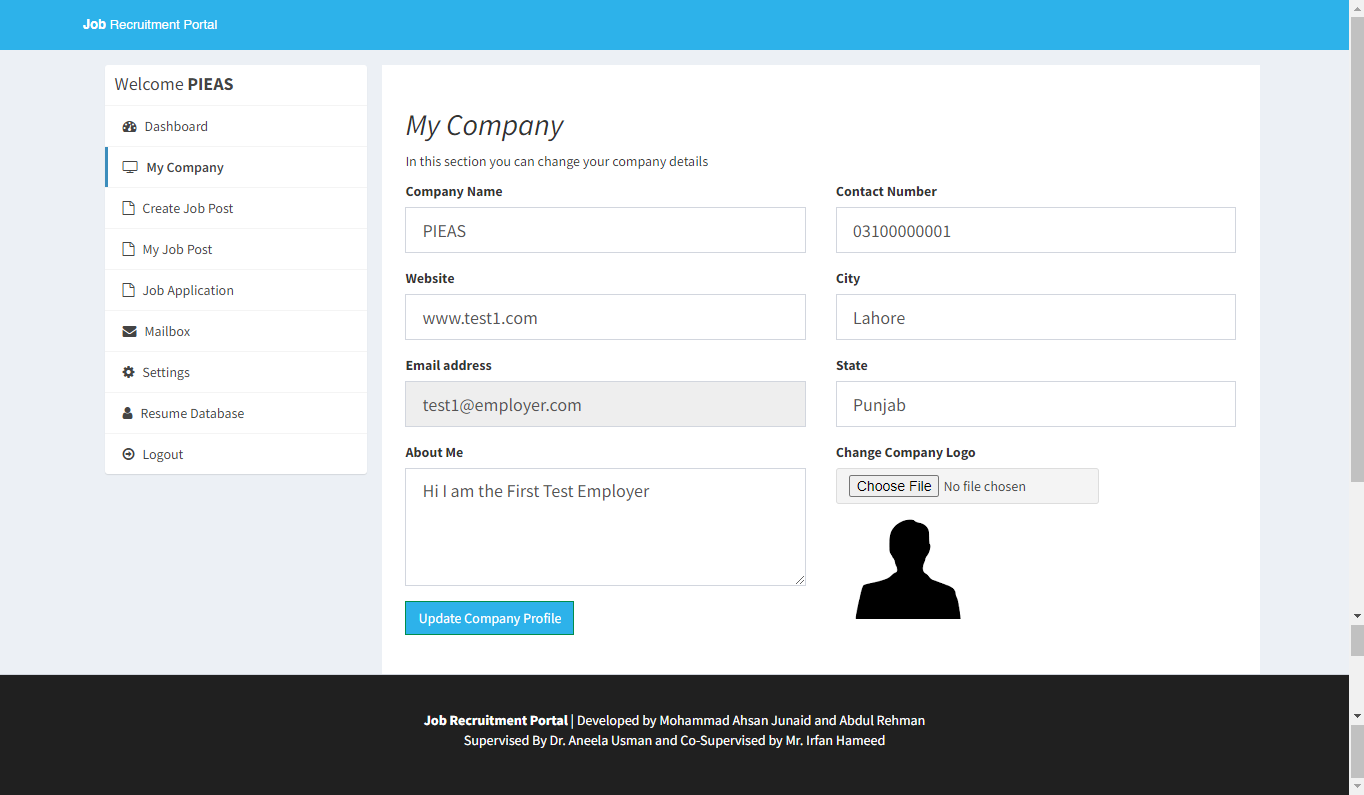
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard – Company Details Page

The following features would be displayed on the page's output:

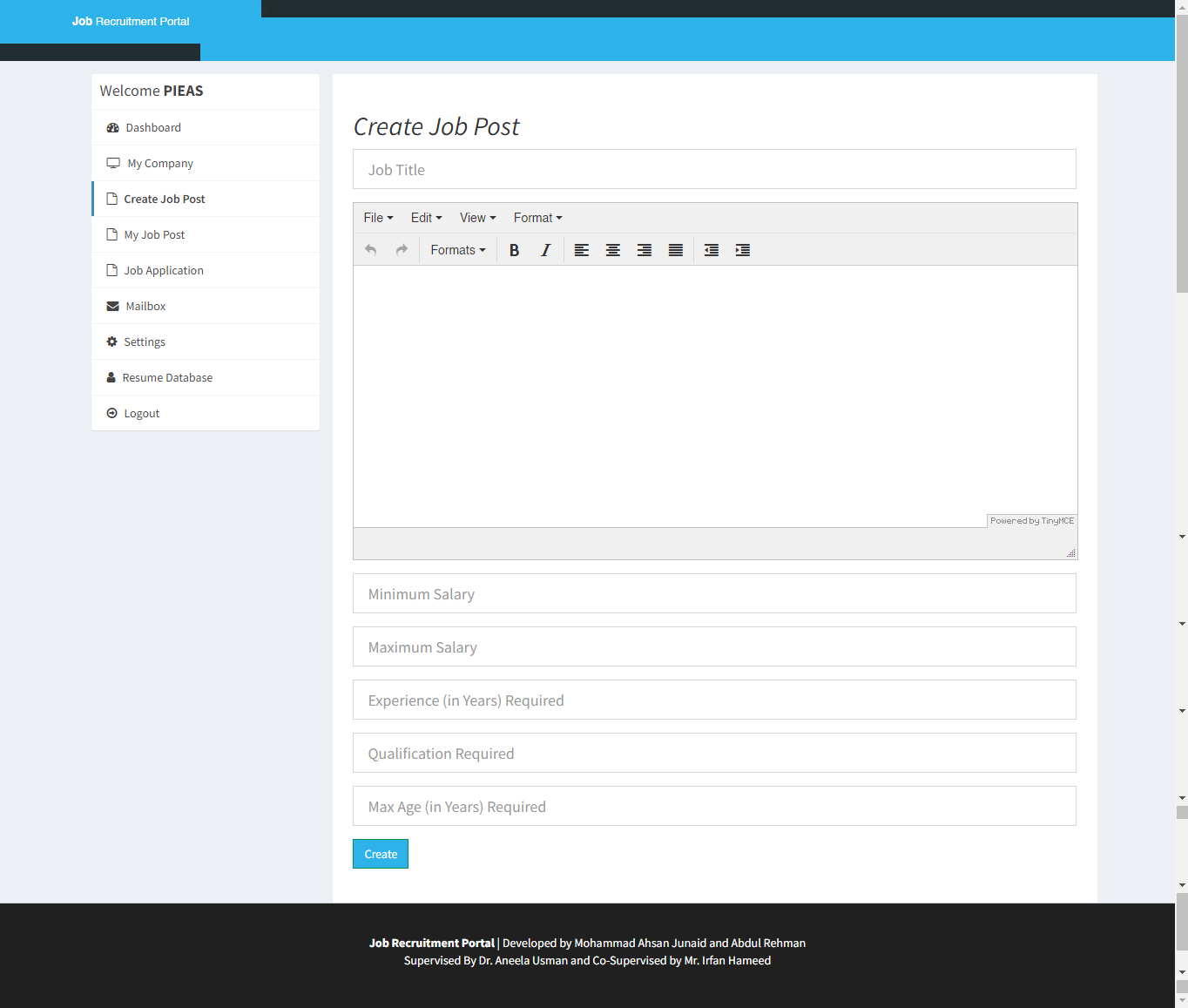
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard – Create Job Page

The following features would be displayed on the page's output:

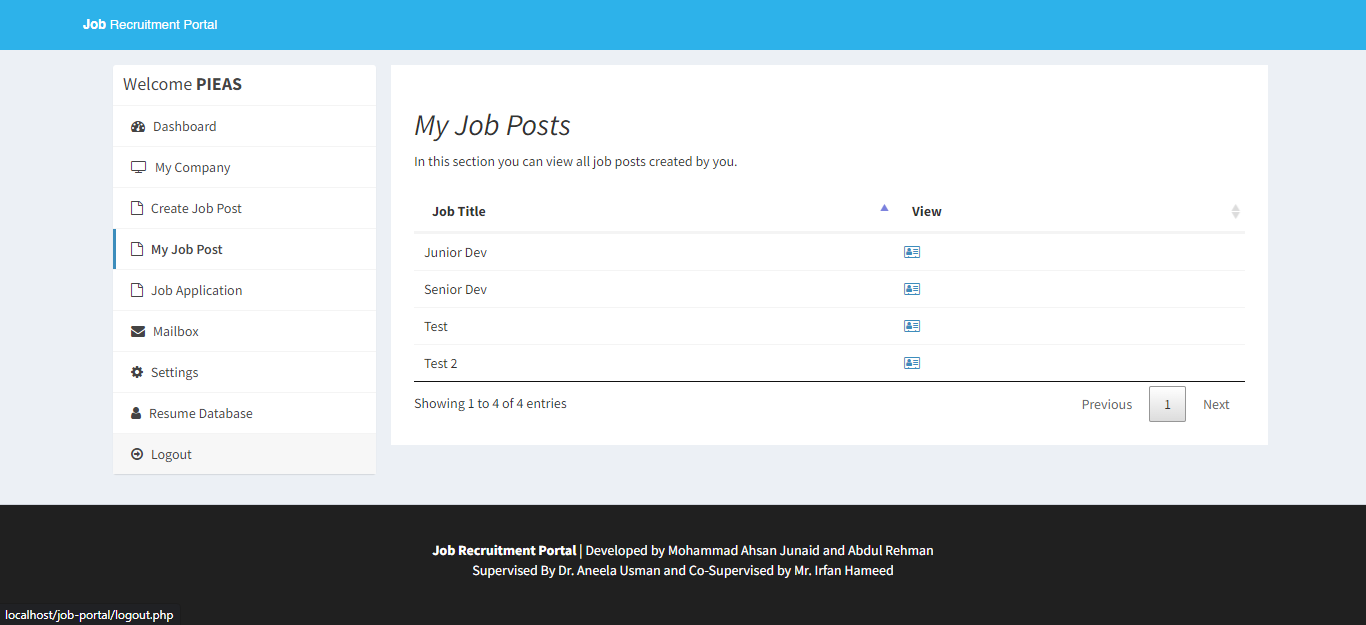
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard – Posted Jobs

The following features would be displayed on the page's output:

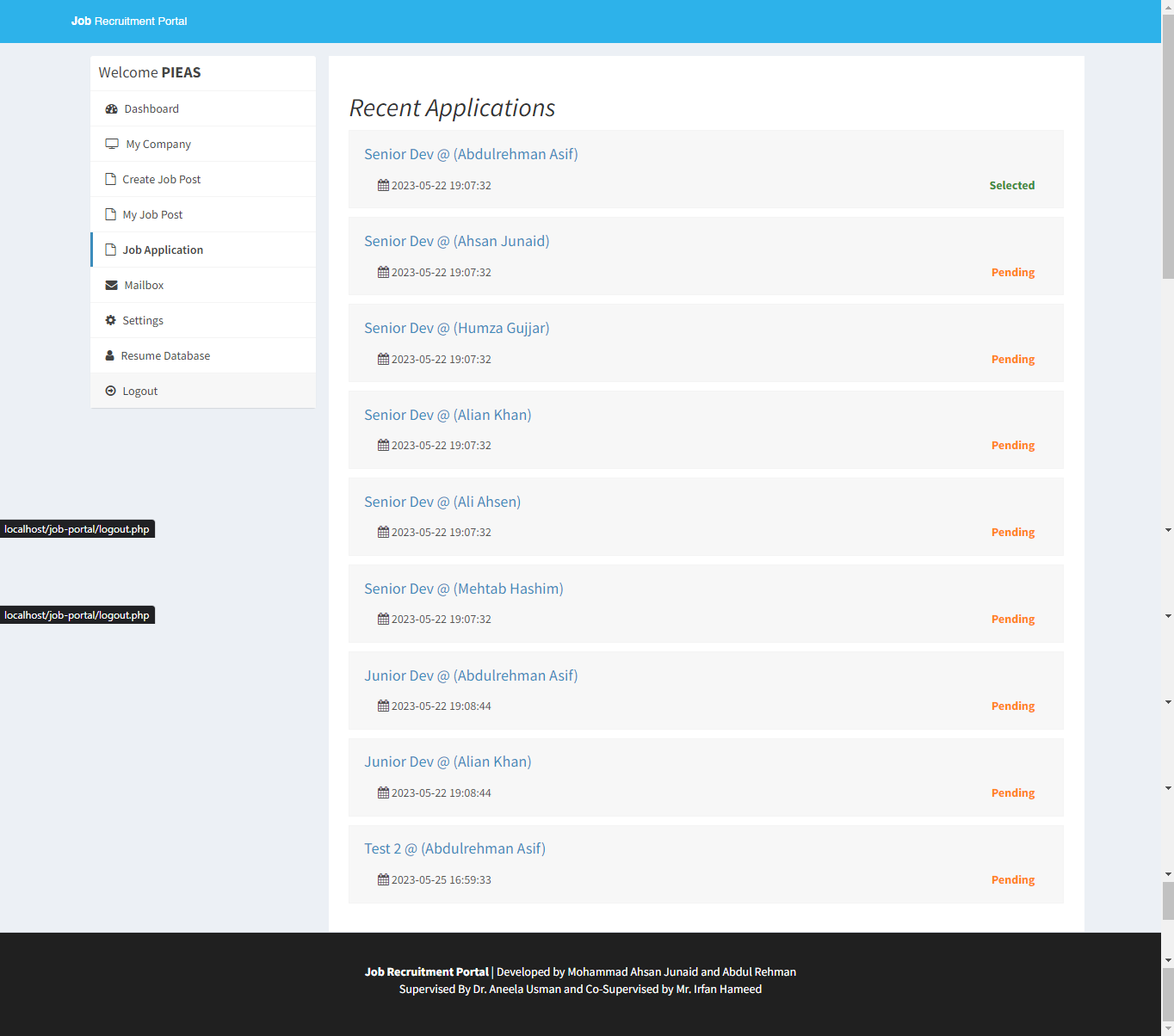
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard – Job Applications Page

The following features would be displayed on the page's output:

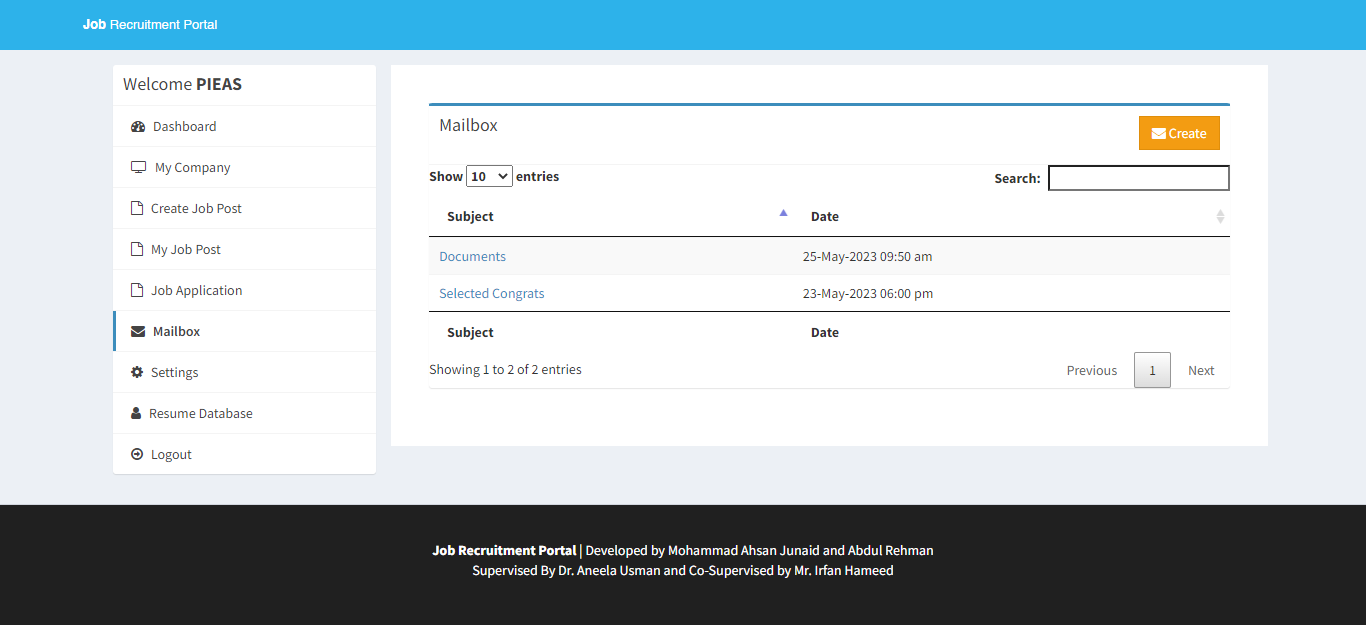
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard – Mailbox

The following features would be displayed on the page's output:

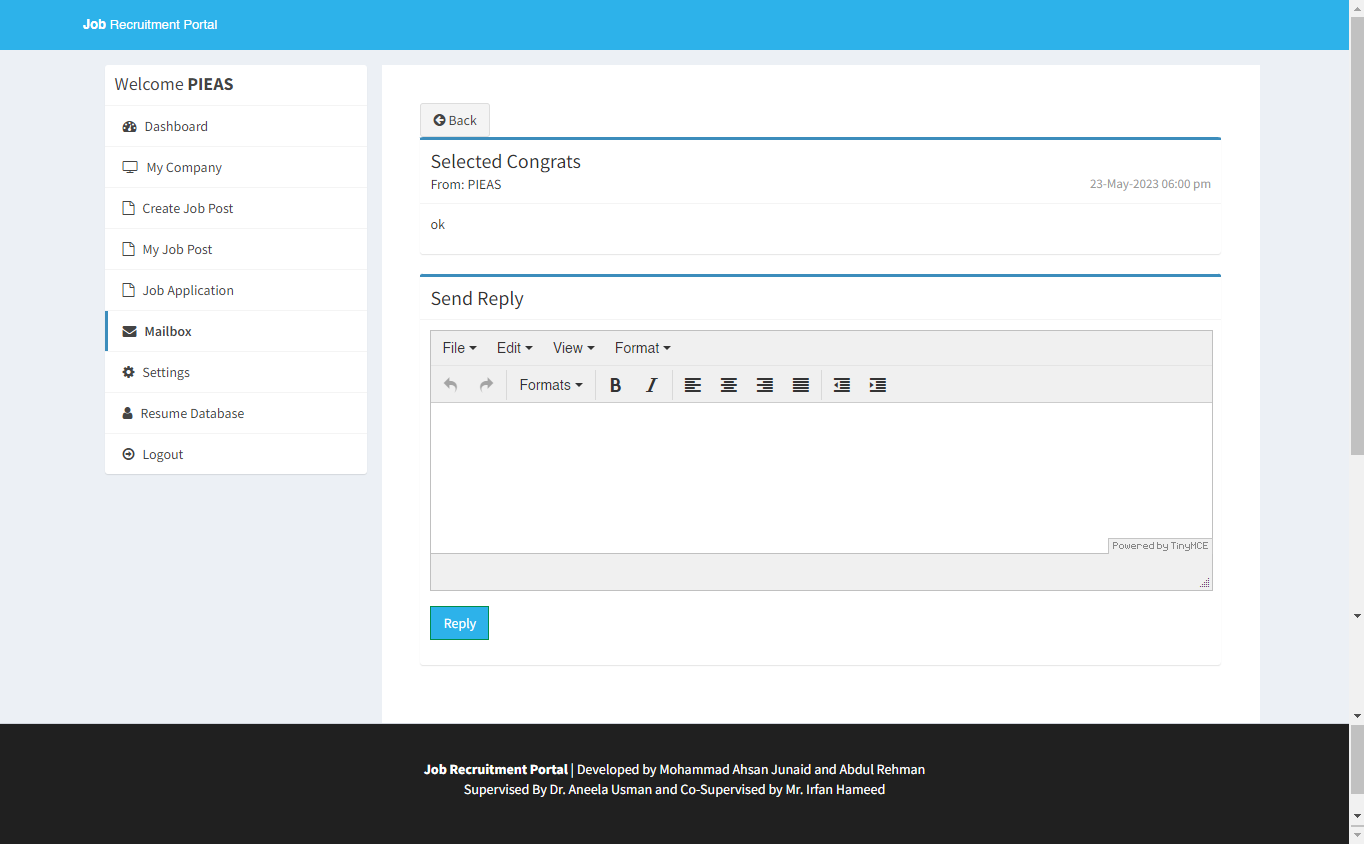
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard – Reply / Create Mail Page

The following features would be displayed on the page's output:

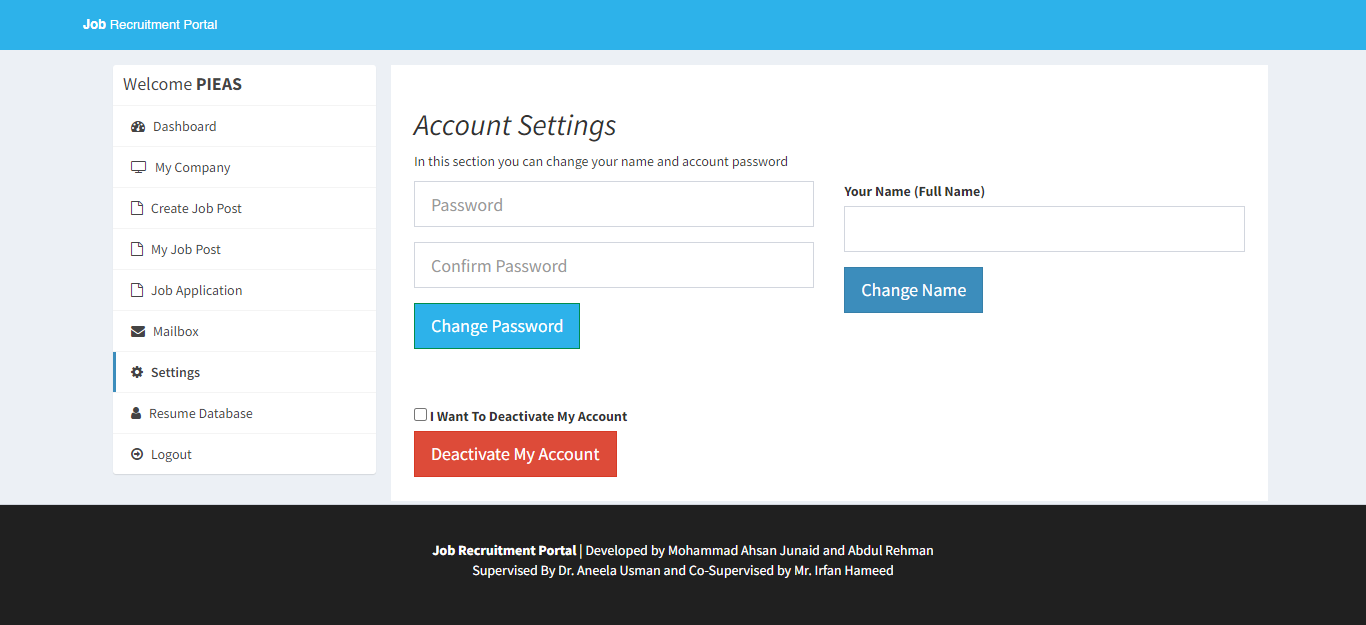
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard - Settings

The following features would be displayed on the page's output:

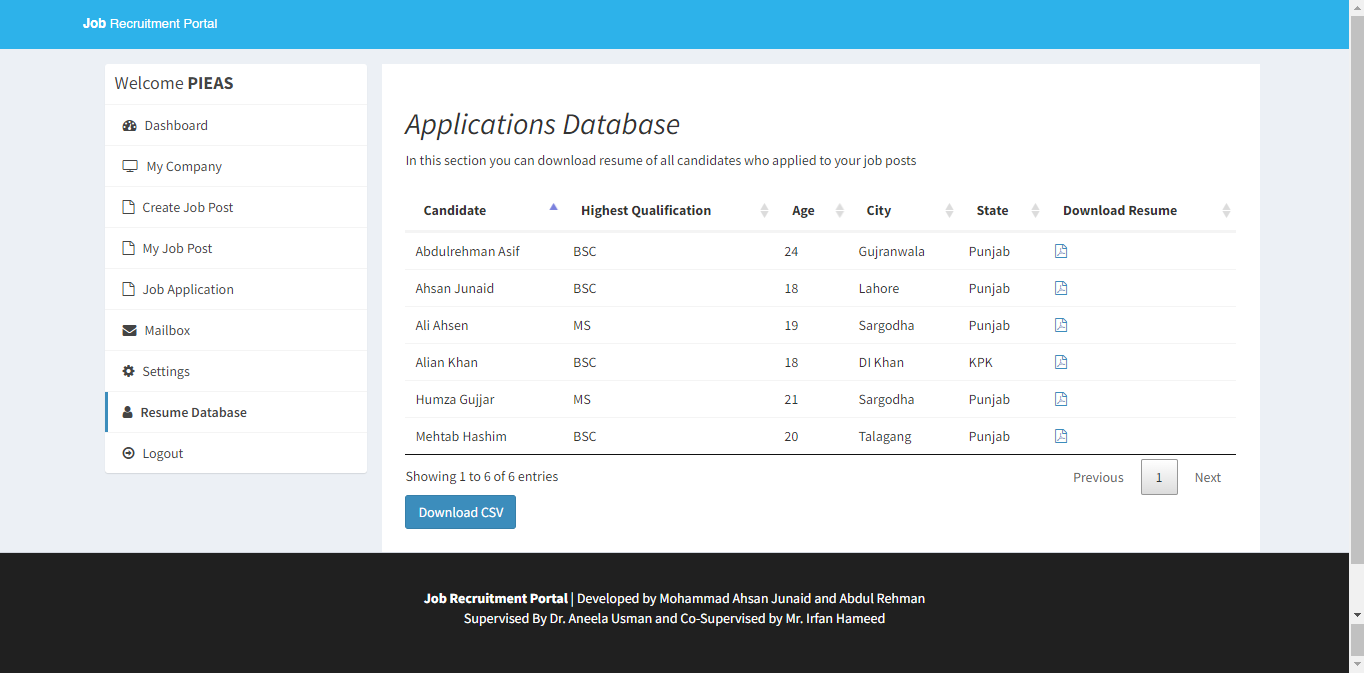
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Employer Dashboard – Applications Database

The following features would be displayed on the page's output:

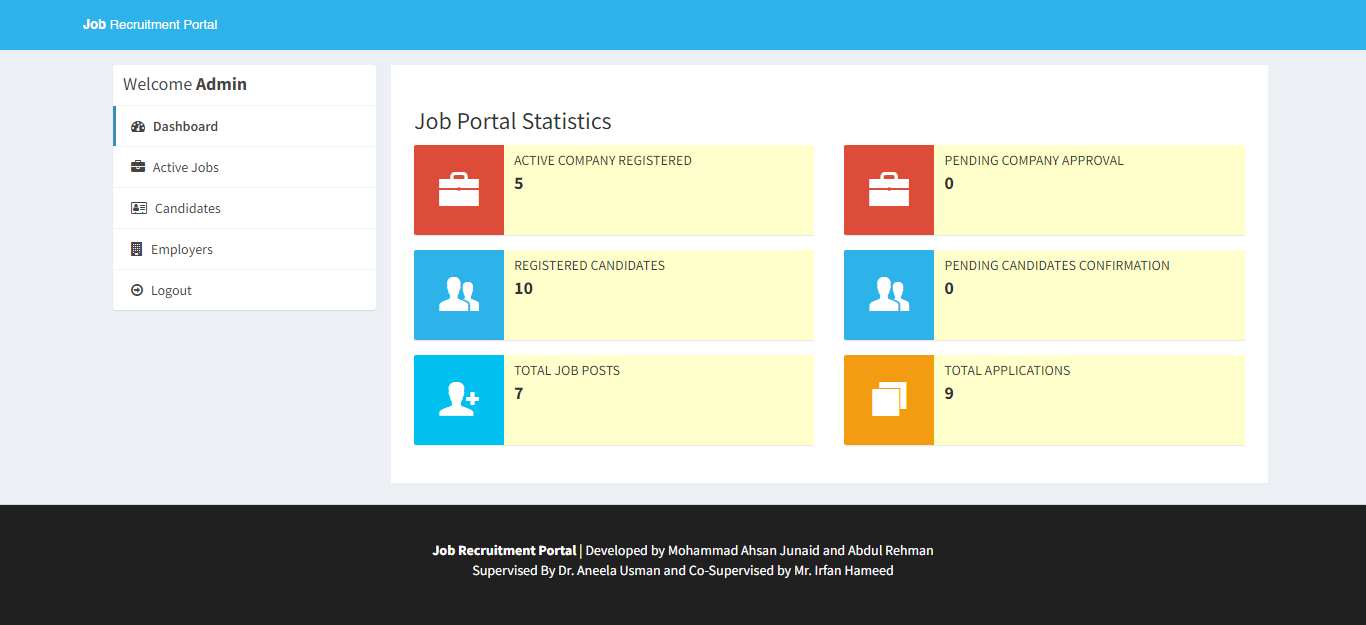
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Admin Dashboard

The following features would be displayed on the page's output:

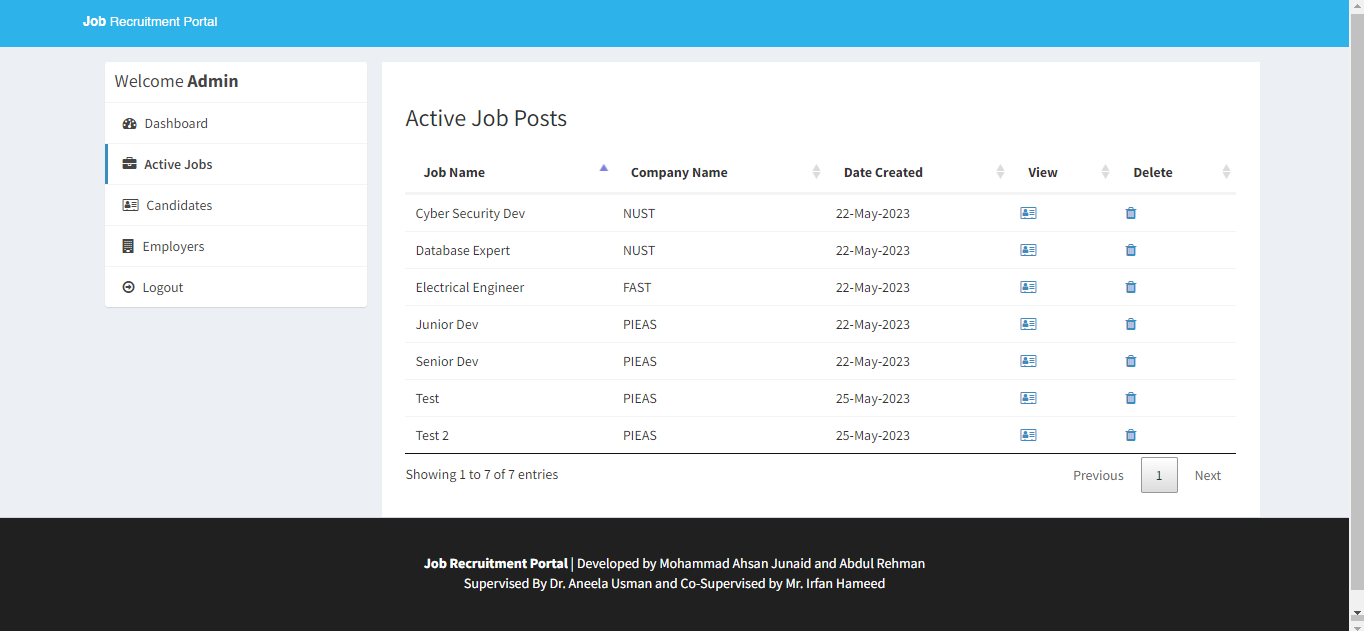
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Admin Dashboard – Active Jobs

The following features would be displayed on the page's output:

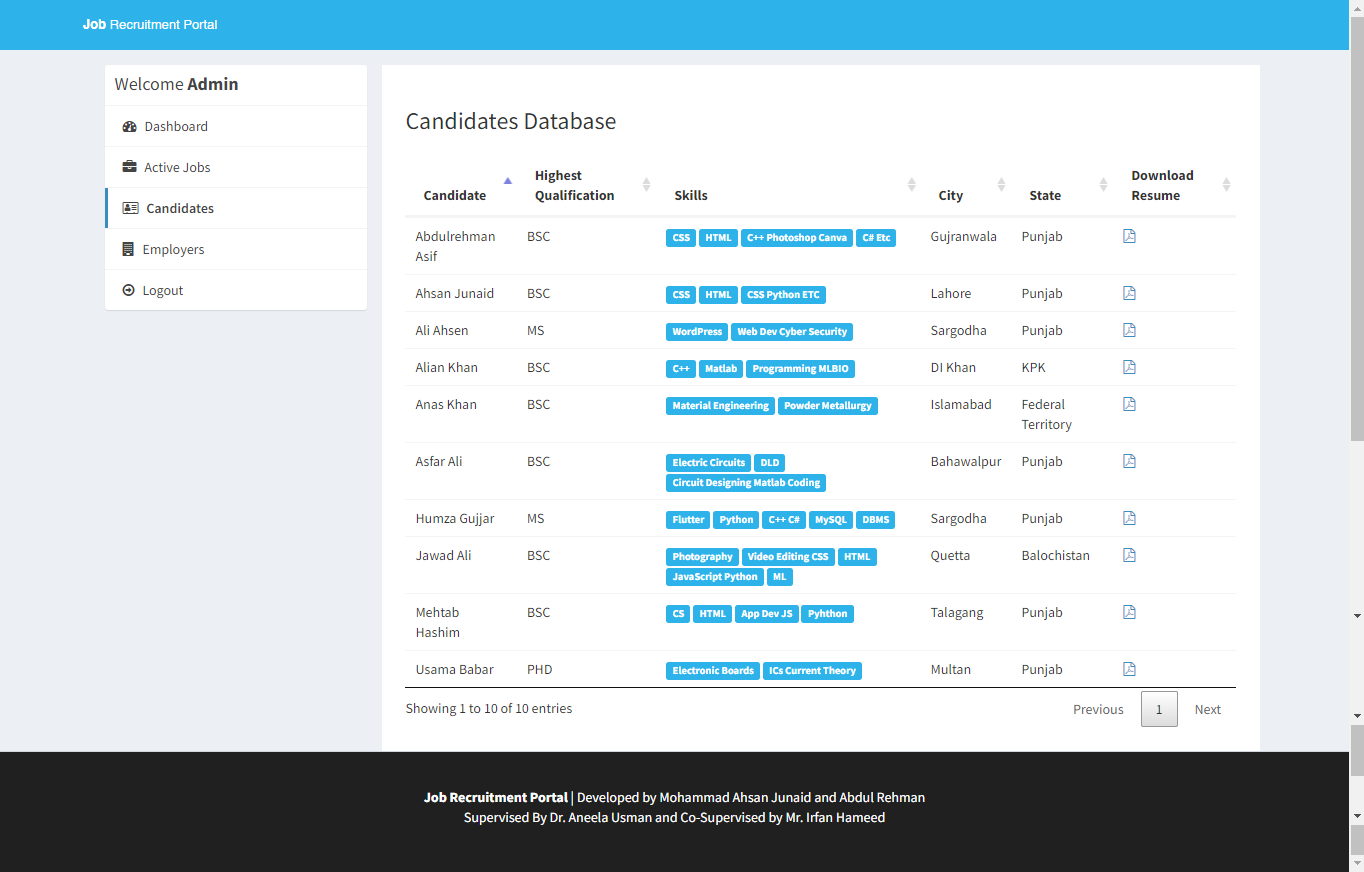
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Admin Dashboard – Applicants Database

The following features would be displayed on the page's output:

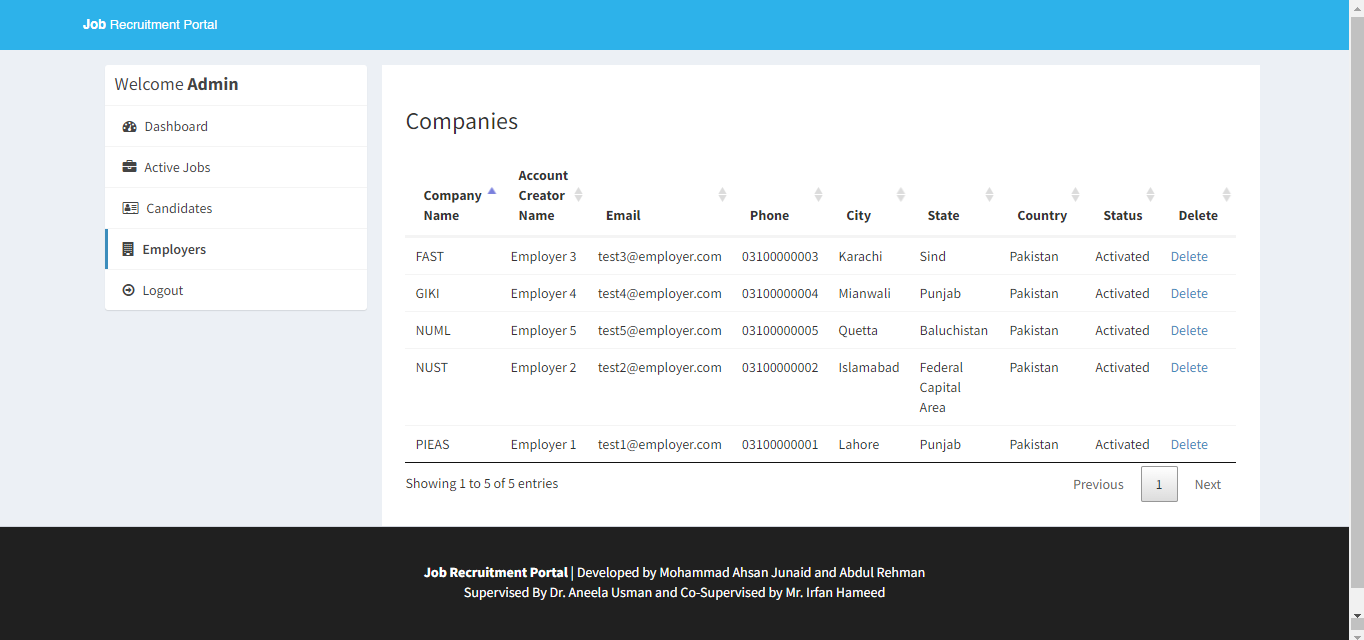
* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



### Admin Dashboard – Employers Database

The following features would be displayed on the page's output:

* Navigation Menu:  
  Depending on their session status, users can access various website sections such as "Jobs," "Login," "Sign Up," "Dashboard," and "Logout" via a navigation menu that would be visible in the header.



## Backend Development

### How to Calculate Workload?

Workload calculation was one of the challenging tasks as there was no previous system in the institute that calculates workloads of its employees. Moreover, there are several factors that can contribute in workload so how to tackle them and bring into an automated system was also a big challenge for me. For this, a formula was derived to calculate workload’s final score.

Final Score = [Factor 1\*wFactor 1] + [Factor 2\*wFactor 2] + [Factor 3\*wFactor 3] & so on... where

* + - 1. Factor is the value that user types in the workload form.
      2. wFactor is the weight of a particular factor.

### How to Handle Workload Weights?

Each factor of the workload contains its weight. So, if the weights of factors are set static, it becomes a hectic task to change the weight of a factor repeatedly because there may come a need in future to alter any of the weights.

In order to tackle this situation, a configuration page is developed where all the weightage parameters can be updated.

## Database Structure

The features of the information system are following:

### Home Page

Home page is the first page that will be shown to the user. The interface of the login page is shown in figure 4.7.

## Web Hosting

The features of the information system are following:

### Home Page

Home page is the first page that will be shown to the user. The interface of the login page is shown in figure 4.7.

# Chapter 5: Structure

## Front End Structural Diagram

The technologies and frameworks that are used to develop system are;

* + 1. Bootstrap
    2. ReactJS
    3. MongoDB

## Backend and Database Diagrams

### ER Diagram

A concept or thing within a system, such as a person/role, object, concept, or event, is referred to as an entity.

### Use Case Diagram

A concept or thing within a system, such as a person/role, object, concept, or event, is referred to as an entity.

### Context Diagram

A concept or thing within a system, such as a person/role, object, concept, or event, is referred to as an entity.

### Relational Schema

A concept or thing within a system, such as a person/role, object, concept, or event, is referred to as an entity.

# Chapter 6: User Roles

## . Administrator

The technologies and frameworks that are used to develop system are;

* + 1. Bootstrap
    2. ReactJS
    3. MongoDB

## . Employer

### ER Diagram

A concept or thing within a system, such as a person/role, object, concept, or event, is referred to as an entity.

### Use Case Diagram

A concept or thing within a system, such as a person/role, object, concept, or event, is referred to as an entity.

### Sequence Diagram

A concept or thing within a system, such as a person/role, object, concept, or event, is referred to as an entity.

## . Applicant / Candidate

The technologies and frameworks that are used to develop system are;

* + 1. Bootstrap
    2. ReactJS

# Chapter 7: Conclusion and Future Works

## 7.1. Conclusion

The technologies and frameworks that are used to develop system are;

* + 1. Bootstrap
    2. ReactJS

## 7.2. Future Works

The technologies and frameworks that are used to develop system are;

* + 1. Bootstrap

# References

1. “HTML | PyCharm,” *PyCharm Help*. [https://www.jetbrains.com/help/pycharm/editing-](https://www.jetbrains.com/help/pycharm/editing-%20%20%20%20%20html-files.html) [html-files.html](https://www.jetbrains.com/help/pycharm/editing-%20%20%20%20%20html-files.html) (accessed Aug. 15, 2022).
2. “Returning To Plain HTML + CSS. Over the past week, I tried my hand at… | by Danny Krug | Medium.” [https://medium.com/@dannykrug/returning-to-plain-html-css-](https://medium.com/%40dannykrug/returning-to-plain-html-css-4386b61f3da) [4386b61f3da](https://medium.com/%40dannykrug/returning-to-plain-html-css-4386b61f3da) (accessed Aug. 18, 2022).
3. “React – A JavaScript library for building user interfaces.” <https://reactjs.org/> (accessed Aug. 28, 2022).

[4].“Databases and Collections — MongoDB Manual.” <https://www.mongodb.com/docs/manual/core/databases-and-collections/> (accessed Aug. 29, 2022).

1. “The Complete 2022 Web Development Bootcamp,” *Udemy*[.](https://www.udemy.com/course/the-complete-web-development-bootcamp/) <https://www.udemy.com/course/the-complete-web-development-bootcamp/> (accessed Aug. 16, 2022).
2. M. O. contributors Jacob Thornton, and Bootstrap, “Bootstrap.” <https://getbootstrap.com/>(accessed Aug. 24, 2022).
3. “MongoDB Atlas: Cloud Document Database | MongoDB.” [https://www.mongodb.com/cloud/atlas/lp/try2?utm\_content=controlhterms&utm\_source](https://www.mongodb.com/cloud/atlas/lp/try2?utm_content=controlhterms&utm_source=google&utm_campaign=gs_emea_pakistan_search_core_brand_atlas_desktop&utm_term=mongodb&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=12212624545&adgroup=115749718983&gclid=Cj0KCQjwr4eYBhDrARIsANPywChc7Rl0dXbU5oNRLKUhiUPNuxtg1suhu9H0H8seDK0506n9W24dCKAaAp4CEALw_wcB)

[=google&utm\_campaign=gs\_emea\_pakistan\_search\_core\_brand\_atlas\_desktop&utm\_ter](https://www.mongodb.com/cloud/atlas/lp/try2?utm_content=controlhterms&utm_source=google&utm_campaign=gs_emea_pakistan_search_core_brand_atlas_desktop&utm_term=mongodb&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=12212624545&adgroup=115749718983&gclid=Cj0KCQjwr4eYBhDrARIsANPywChc7Rl0dXbU5oNRLKUhiUPNuxtg1suhu9H0H8seDK0506n9W24dCKAaAp4CEALw_wcB) [m=mongodb&utm\_medium=cpc\_paid\_search&utm\_ad=e&utm\_ad\_campaign\_id=12212](https://www.mongodb.com/cloud/atlas/lp/try2?utm_content=controlhterms&utm_source=google&utm_campaign=gs_emea_pakistan_search_core_brand_atlas_desktop&utm_term=mongodb&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=12212624545&adgroup=115749718983&gclid=Cj0KCQjwr4eYBhDrARIsANPywChc7Rl0dXbU5oNRLKUhiUPNuxtg1suhu9H0H8seDK0506n9W24dCKAaAp4CEALw_wcB) [624545&adgroup=115749718983&gclid=Cj0KCQjwr4eYBhDrARIsANPywChc7Rl0d](https://www.mongodb.com/cloud/atlas/lp/try2?utm_content=controlhterms&utm_source=google&utm_campaign=gs_emea_pakistan_search_core_brand_atlas_desktop&utm_term=mongodb&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=12212624545&adgroup=115749718983&gclid=Cj0KCQjwr4eYBhDrARIsANPywChc7Rl0dXbU5oNRLKUhiUPNuxtg1suhu9H0H8seDK0506n9W24dCKAaAp4CEALw_wcB) [XbU5oNRLKUhiUPNuxtg1suhu9H0H8seDK0506n9W24dCKAaAp4CEALw\_wcB](https://www.mongodb.com/cloud/atlas/lp/try2?utm_content=controlhterms&utm_source=google&utm_campaign=gs_emea_pakistan_search_core_brand_atlas_desktop&utm_term=mongodb&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=12212624545&adgroup=115749718983&gclid=Cj0KCQjwr4eYBhDrARIsANPywChc7Rl0dXbU5oNRLKUhiUPNuxtg1suhu9H0H8seDK0506n9W24dCKAaAp4CEALw_wcB) (accessed Aug. 24, 2022).

1. “W3Schools Free Online Web Tutorials.” <https://www.w3schools.com/>(accessed Aug. 25, 2022).
2. “MDN Web Docs.” [https://developer.mozilla.org](https://developer.mozilla.org/) (accessed Aug. 25, 2022).
3. “Database Management System”, published by IT Series Publications and the authors fo the book are Imran Saeed, Tasleem Mustafa, Tariq Mahmood and Ahsan Raza Sattar. [https://freebooks.pk/books/database-management-system](https://freebooks.pk/books/database-management-system/)/ (accessed Aug. 26, 2022).
4. “Entity Diagram - diagrams.net.” <https://app.diagrams.net/> (accessed Aug. 29, 2022).

**Appendix A – Instructions to Use System**

Instructions to use Information System:

* To use the system, a user must have an account and must be logged in the system.
* All the configuration related to programs, weights and pay rates will be on the configuration page.
* All the employees and courses should be added and checked before filling the workload page.
* In the course section, while adding courses, if the course has theory credit, lab credit must be filled with zero and vice versa.

# Appendix B – React JS Installation and Setup

1. **Install Nodejs**

Packages for React are managed and shared using NPM, the Node package manager. Along with Nodejs, NPM will also be installed. It is possible to download and set up Node.js from the official Node Js website.

https://nodejs.org

Once the Installation of Node is complete. Open Node Js Command Prompt and we can check the version as well.

1. **Install Create-React-App Tool**

The following action is to use NPM to install a project called create-react-app. Utilizing our technology, this tool makes it simple to construct React applications. This can be set up either permanently at the system level or momentarily at a folder level.

npm install -g create-react-app

1. **Creating a New React Project**

After installing create-react-app, we can start building our first react application. Following is command used to create a new project

create-react-app name-of-project

Don't use an uppercase character while creating the project.

1. **Running the React Application**

Now lets run it locally on our system using npm start command. Launch the browser and visit http://localhost:3000. We can see our first React Application in the browser.

cd name-of-project npm start

Now, we have created a new project using react and executed the project.

1. **Install Visual Studio Code**

Download and install Visual Studio Code from the following URL

<https://code.visualstudio.com/download>

After the VS Code installation, open the project that has been created earlier using the VS Code. The Project has the following 3 folders

* 1. Node\_modules
  2. Public
  3. src

The output we have seen when the project is executed comes from a file called Index.html which resides inside the public folder.

# Appendix C – Libraries Used in Project

Name and purpose of libraries used in this project:

### Bootstrap

This is a CSS framework that is used to create responsive websites.

### React

This is a front-end java script library used to build user interfaces.

### Axios

This library is used to make http requests from frontend.

### React-Router-Dom

This library is used to make routes in frontend.