JOB HIRING PORTAL

EDS6397 - DATABASE MANAGEMENT TOOLS PROJECT

Nagendra Reddy Palugulla - 2250572

Harikesh Govindaiahgari - 2248386

Tharun Pasham - 2256307

Sai Divya Sree Nagapudi Venugopal - 2303233

Abhigna Sowgandhika Vadlamudi - 2268166

Ahmed Hussain Syed - 2273473

Sheema Anush - 2303265

INTRODUCTION (PROPOSAL)

In the ever-thriving employment industry, it is always necessary to keep developing efficient and user-friendly job hiring platforms. Our primary goal is to create a comprehensive web-based Job Hiring Platform, aimed at reducing the gap between job seekers and employers. The platform will enable seamless job searches, streamline the application process, and allow employers to manage their hiring needs effectively.

Our database will cater to three primary user roles: Job Seeker (individuals seeking employment opportunities), Employer (companies and organisations posting job listings), and Administrator (includes those who oversee and manage the entire platform). The platform will utilise a robust technology stack: the front end includes PHP, HTML, CSS, JavaScript, React, or Angular and the database is created using Oracle.

We also put sufficient focus on creating an intuitive and user-friendly UI for all stakeholders by following a comprehensive set of steps including business requirements, logical ER diagrams, relational diagrams, and database creation. The database and the website have been interlinked to achieve smooth data entry and retrieval operations.

This project aims to revolutionise the job hiring process by providing a centralised, efficient, and secure platform for its target consumers. Through careful design and implementation, our Job Hiring Platform will contribute to the creation of a robust ecosystem that facilitates meaningful connections in the job market.

BUSINESS RULES

The proposed database design for the job hiring platform is governed by a set of well-defined business rules to ensure data integrity, security, and consistency. These are listed as follows corresponding to the respective features they contribute to.

1. User Registration and Authentication:
    - All users must provide a valid email address during registration, and email addresses must be unique across the system.
    - Passwords must adhere to a secure password policy, including a minimum length and a combination of alphanumeric characters.
2. Academic Qualifications:
    - Each academic qualification entry must be associated with a valid member ID, ensuring that qualifications are correctly linked to individual members. But each member could have many academic qualifications.
    - The timeframe for academic qualifications should reflect the period during which the qualification was obtained.
3. Professional Experience:
    - Professional experience entries must be linked to a valid member ID, establishing a clear connection between the experience and the member's profile. But each member could have many Professional Experiences.
    - Start dates for professional experience must precede or be equal to end dates, ensuring logical and consistent date ranges.
4. Job Listings:
    - Each job listing must have a unique job ID to distinguish it from other listings.
    - Each job could receive many applications but only one application can be posted to a job by a member
5. Language Proficiency:
    - Language proficiency entries must be associated with a valid member ID, connecting language skills to specific members.
    - Each Person can speak many languages
6. Referees:
    - Referee entries must be unique and linked to a valid member ID, establishing a clear connection between referees and the members they are associated with.
    - Each member could have multiple references and referred entity does not collide with users
7. User Roles:

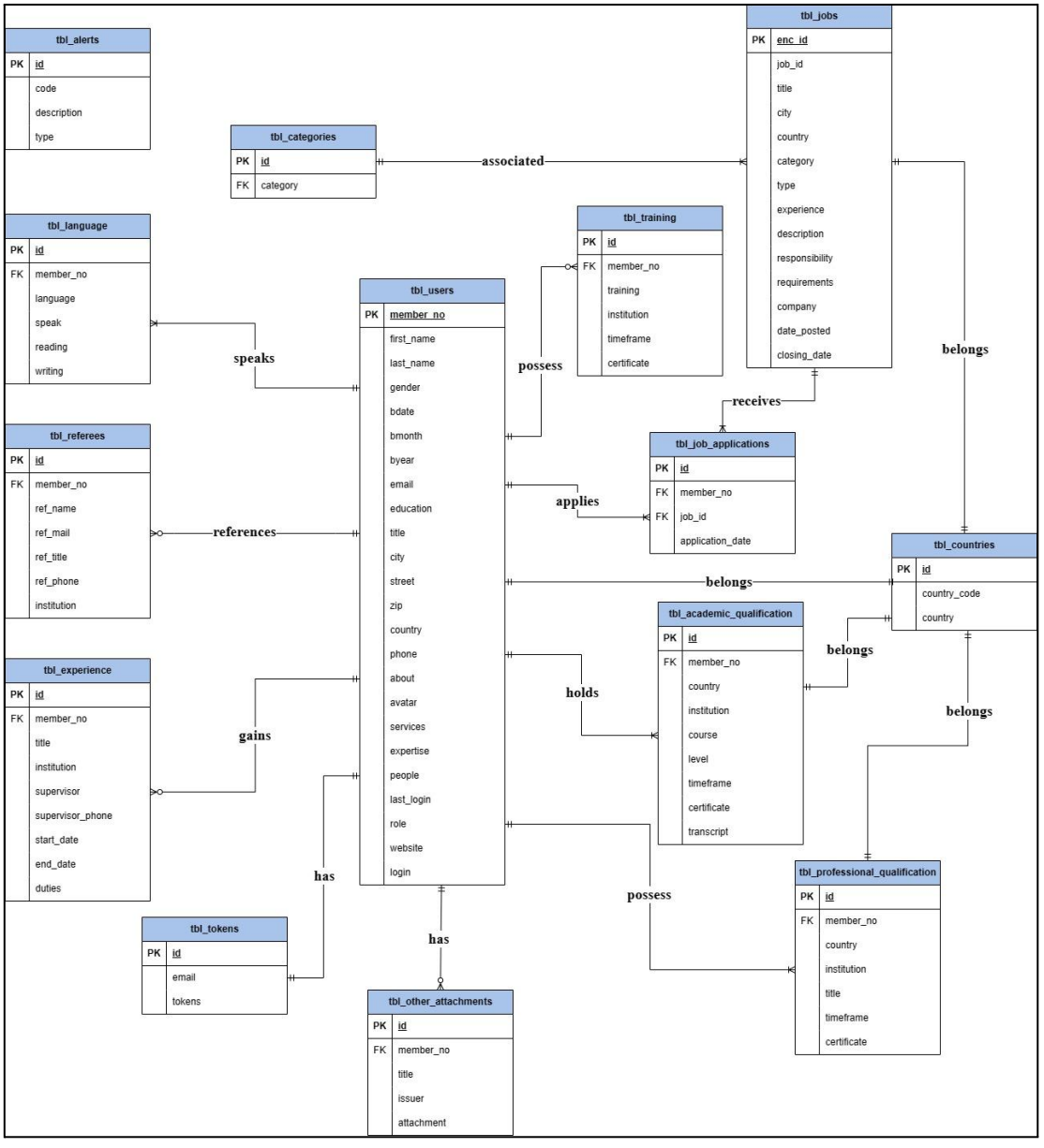
- User roles (e.g., employee, employer) must be assigned based on user registration information and are used to control access to specific features.

8. Alerts:
    - Alert codes and types must follow a predefined set of values, ensuring consistency and facilitating effective communication through alerts.
    - Only purpose of this entity is to have a smooth UI.
9. Tokens:
    - Each token must be associated with a valid email address, linking tokens to specific users for authentication purposes.
    - Tokens should have a limited validity period to enhance security.
10. Training:
    - Training entries must be linked to a valid member ID, connecting training records to specific members.
    - The timeframe for training should accurately reflect the period during which the training occurred.
    - Each member could have undergone many training sessions.
11. Other Attachments:
    - Other attachments must be associated with a valid member ID, ensuring that supplementary materials are correctly linked to individual member profiles.
    - Attachment titles should follow a standardised naming convention for easy identification.
    - Each member could have many attachments .
12. Professional Qualifications:
    - Professional qualification entries must be linked to a valid member ID, establishing a clear connection between qualifications and the members they belong to. But each member could have many Professional qualifications.
    - The timeframe for professional qualifications should accurately reflect the period during which the qualification was obtained.
13. Categories:
    - Categories must have unique category names to avoid ambiguity and ensure a clear classification system.
    - Category IDs must be unique, serving as distinct identifiers for each category.
    - Many jobs could be posted for the same category.
14. Countries:
    - Country codes must adhere to a standard two-letter format, following international conventions.
    - Country names must be unique, preventing duplicate entries for the same country.
    - This table is mostly used for dropdown menus.

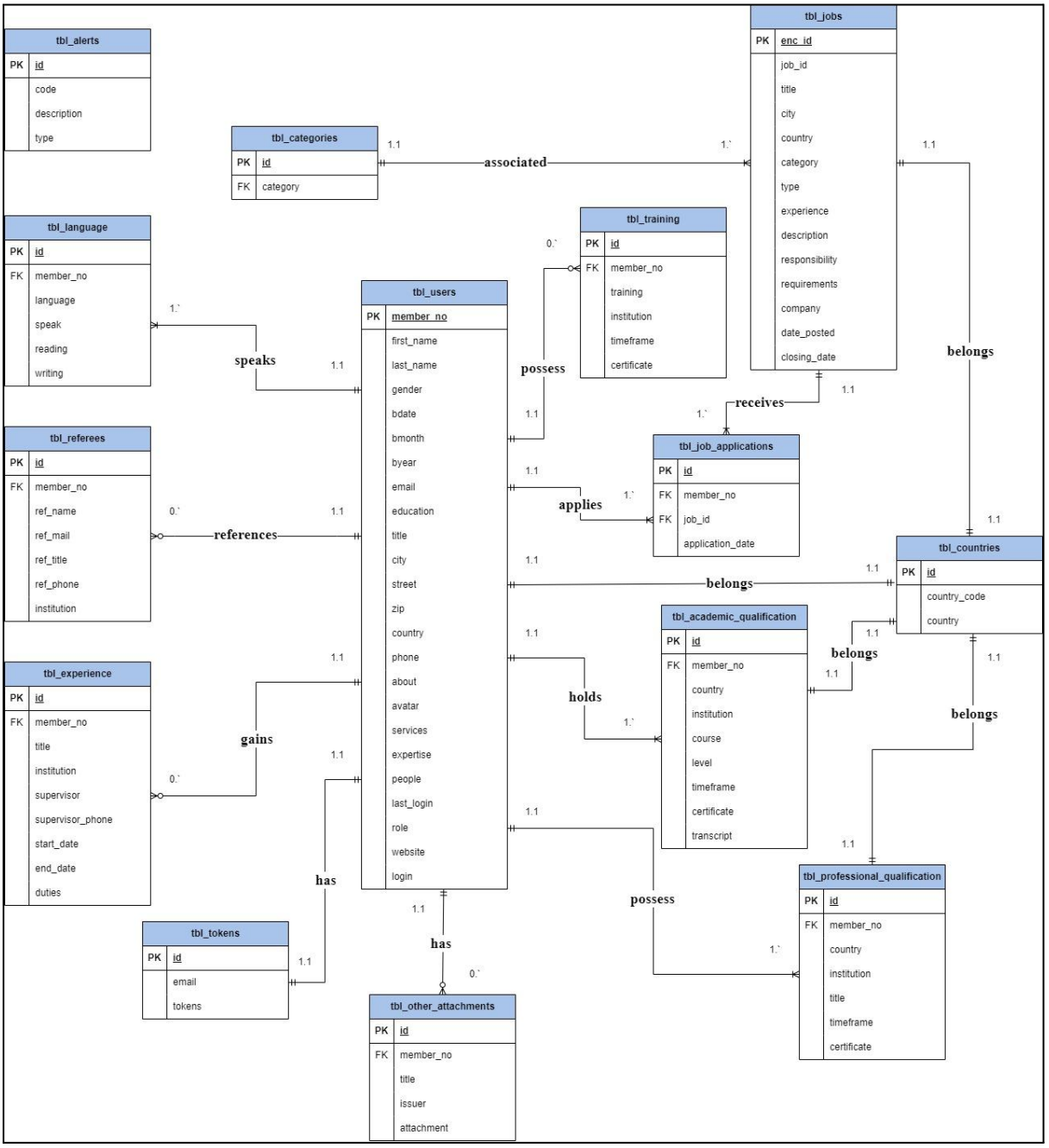
15. Users:
    - User accounts must have a valid role assigned, determining the level of access and functionality available to each user.
    - Usernames or login information must be unique across the system to avoid conflicts.

ENTITY RELATIONSHIP DIAGRAM

Our ERD contains fourteen entities, each serving a primary role in the database. These entities are interlinked using foreign keys and are identified using their primary keys. The ERD given below thus gives the complete idea of the functionality of the database:



RELATIONAL DIAGRAM



ENTITIES

We now look at each entity in detail, identifying its purpose and the respective primary and foreign keys. This enables a deeper understanding of our database and its elements.

1. tbl_academic_qualification: represents academic qualifications of members, storing information such as member ID (PRIMARY KEY), country, institution, course, level, timeframe, and related certificates and transcripts. This table allows the system to maintain a comprehensive record of academic achievements for each member based on their member number (FOREIGN KEY), supporting educational background verification.

2. tbl_alerts: manages system alerts, storing details such as alert ID (PRIMARY KEY), code, description and type. This table is crucial for notifying users about important events or updates within the system, enhancing user engagement.

3. tbl_categories: defines job categories within the system with category ID (PRIMARY KEY) for each category (FOREIGN KEY) and contributes to a structured organisation of information.

4. tbl_countries: stores information about the country the user pursued their education from, including country ID (PRIMARY KEY), country code, and name.

5. tbl_experience: manages professional work experience, storing details such as experience ID (PRIMARY KEY), member no (FOREIGN KEY), job title, duties, start and end dates, etc to facilitate a comprehensive overview of a member's work history for potential employers.

6. tbl_job_applications: tracks job applications, including details such as application ID (PRIMARY KEY), member no (FOREIGN KEY), job ID (FOREIGN KEY), and application date to manage the application process, allowing users to keep track of their submitted applications.

7. tbl_jobs: manages job listings, storing information such as enc ID (PRIMARY KEY), title, location, category, type, experience requirements, description, responsibilities, requirements, company details, posting date, and closing date. This table forms the core of the job posting and search functionality, facilitating a rich database of employment opportunities.

8. tbl_language: records information about members' language proficiency, storing details such as language ID (PRIMARY KEY), member no (FOREIGN

KEY), and proficiency levels to provide insights into a member's language skills for roles requiring specific language abilities.

9. tbl_other_attachments: manages additional attachments, storing information such as attachment ID (PRIMARY KEY), member no (FOREIGN KEY), title, and issuer to support the inclusion of diverse supplementary materials, enhancing complete member profiles.

10. tbl_professional_qualification: records professional qualifications of members with qualification ID (PRIMARY KEY), member no (FOREIGN KEY), etc., and contributes to a comprehensive overview of members' professional credentials.

11. tbl_referees: stores information about the applicant's references, including referee ID (PRIMARY KEY), member no (FOREIGN KEY), and other details to provide contact details for individuals who can vouch for their professional capabilities.

12. tbl_tokens: manages authentication tokens, storing details such as token ID (PRIMARY KEY), email (FOREIGN KEY), and token. This table plays a crucial role in the security of the system by facilitating secure user authentication through token-based mechanisms.

13. tbl_training: records training ID (PRIMARY KEY), member no (FOREIGN KEY), training name, institution, timeframe, and related certificates to get an overview of each member's additional training and skill development.

14. tbl_users: represents user profiles, storing information such as personal details, contact information, about sections, avatar data, expertise, services offered, and role. This table serves as the central hub for user data, supporting the creation and maintenance of diverse member profiles within the system. It contains the attributes member no (PRIMARY KEY) along with many others and serves as the reference table to most of the other entities.

Thus, the entities within the proposed database exhibit a coherent structure designed to improve the job hiring platform's functionality. This well-defined database framework lays the groundwork for a seamless user experience, facilitating meaningful connections between job seekers and employers.

METADATA (Data Dictionary)

	Entity	Datatype	Required (Y/N)	PK/FK
tbl_academic_qualification	id	INT	Y	**PK**
	member_no	VARCHAR(255)	Y	FK
	country	VARCHAR(255)	Y	
	institution	VARCHAR(255)	Y	
	course	VARCHAR(255)	Y	
	level	VARCHAR(255)	Y	
	timeframe	VARCHAR(255)	Y	
	certificate	LONGBLOB	Y	
	transcript	LONGBLOB	N	
tbl_alerts	id	INT(255)	Y	**PK**
	code	VARCHAR(255)	Y	
	description	VARCHAR(255)	Y	FK
	type	VARCHAR(255)	Y	FK
tbl_categories	id	INT(255)	Y	**PK**
	category	VARCHAR(255)	Y	FK
tbl_countries	id	INT(11)	Y	**PK**
	country_code	VARCHAR(2)	Y	
	country_name	VARCHAR(100)	Y	
tbl_experience	id	INT(255)	Y	**PK**
	member_no	VARCHAR(255)	Y	FK
	title	VARCHAR(255)	Y	
	institution	VARCHAR(255)	Y	
	supervisor	VARCHAR(255)	N	
	supervisor_phone	VARCHAR(255)	N	
	start_date	VARCHAR(255)	Y	
	end_date	VARCHAR(255)	Y	
	duties	LONG TEXT	N	

tbl_job_applications	Id member_no job_id application_date	INT(255_ VARCHAR(255) VARCHAR(255) VARCHAR(255)	Y Y Y Y	**PK** FK FK
tbl_jobs	job_id title city country type experience description responsibility requirements company date_posted closing_date enc_id	VARCHAR(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) Long text Long text Log text VARCHAR(255) VARCHAR(255) VARCHAR(255) INT(255)	Y Y Y Y Y N Y Y Y Y Y Y Y	**PK**
tbl_language	id member_No language speak Reading writing	INT(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) VARCHAR(255)	Y Y Y Y Y Y	**PK** FK
tbl_other_attachments	id member_no title issuer attachment	INT(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) LONGBLOB	Y Y Y Y Y	
tbl_professional_qualification	id member_No country institution title timeframe certificate	INT(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) VARCHAR(255) LONGBLOB	Y Y Y Y N Y Y	**PK** FK

tbl_referees	id	INT(255)	Y	**PK**
	member_No	VARCHAR(255)	Y	FK
	ref_name	VARCHAR(255)	Y	
	ref_mail	VARCHAR(255)	Y	
	ref_phone	VARCHAR(255)	N	
	institution	VARCHAR(255)	N	
tbl_tokens	id	INT(255)	Y	**PK**
	email	VARCHAR(255)	Y	
	token	VARCHAR(255)	Y	
tbl_training	id	INT(255)	Y	**PK**
	member_no	VARCHAR(255)	Y	FK
	training	VARCHAR(255)	Y	
	Institution	VARCHAR(255)	N	
	Timeframe	VARCHAR(255)	Y	
	certificate	LONGBLOB	Y	
tbl_users	member_no	VARCHAR(255)	Y	**PK**
	first_name	VARCHAR(255)	Y	
	Last_name	VARCHAR(255)	Y	
	gender	VARCHAR(255)	N	
	bdate	VARCHAR(255)	N	
	month	VARCHAR(255)	N	
	byear	VARCHAR(255)	N	
	email	VARCHAR(255)	Y	
	education	VARCHAR(255)	N	
	title	VARCHAR(255)	N	
	city	VARCHAR(255)	N	
	street	VARCHAR(255)	N	
	zip	VARCHAR(255)	N	
	country	VARCHAR(255)	N	
	phone	VARCHAR(255)	N	
	about	LONGTEXT	N	
	avatar	LONGBLOB	N	
	services	LONGTEXT	N	
	expertise	LONGTEXT	N	
	people	VARCHAR(255)	N	
	Last_login	VARCHAR(255)	N	
	role	VARCHAR(255)	Y	
	website	VARCHAR(255)	N	
	login	VARCHAR(255)	Y	
	member_no	VARCHAR(255)	Y	

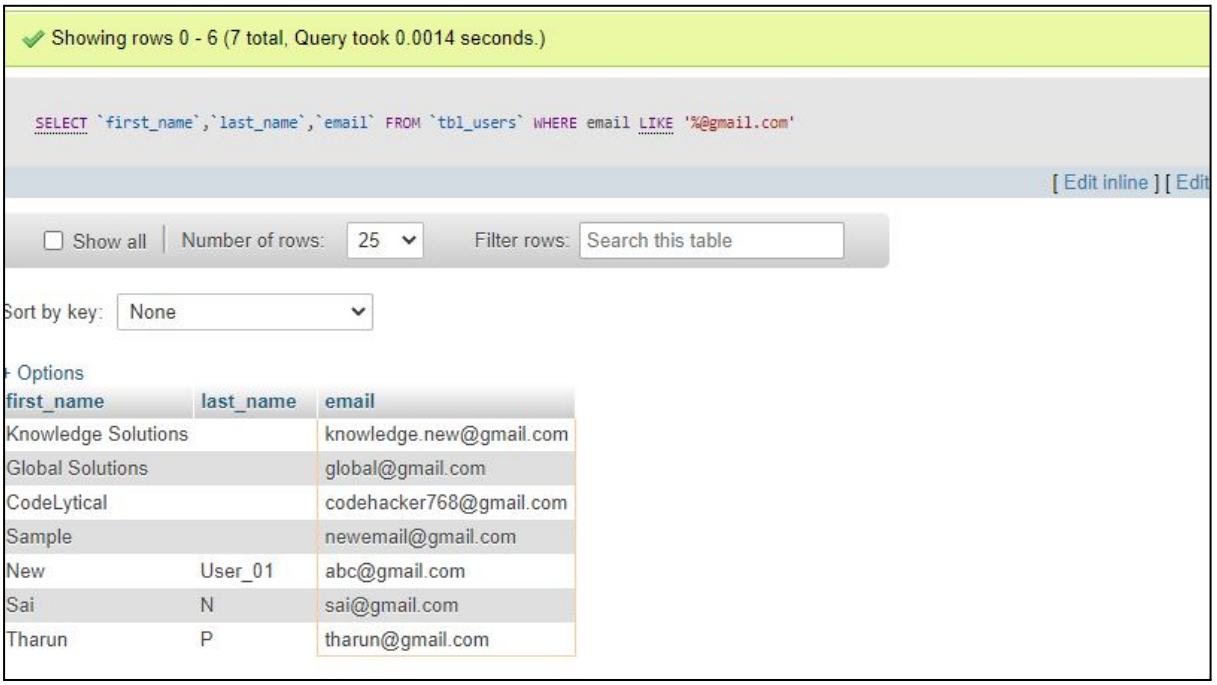
VALIDATION QUERIES WITH CORRESPONDING OUTPUTS:

1) Retrieve the job listings having the highest applications

```
SELECT job_id,title FROM tbl_jobs
WHERE job_id = (
        SELECT job_id FROM tbl_job_applications
        GROUP BY job_id ORDER BY COUNT(job_id) DESC LIMIT  1);
```

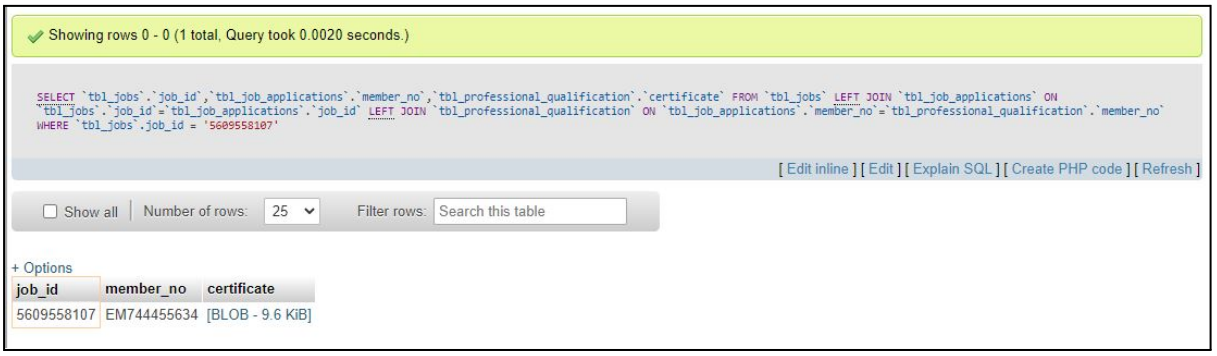
2) Retrieve the name and email of applicants who have a Gmail account.

```
SELECT first_name,last_name,email
FROM tbl_users WHERE email LIKE '%@gmail.com';
```



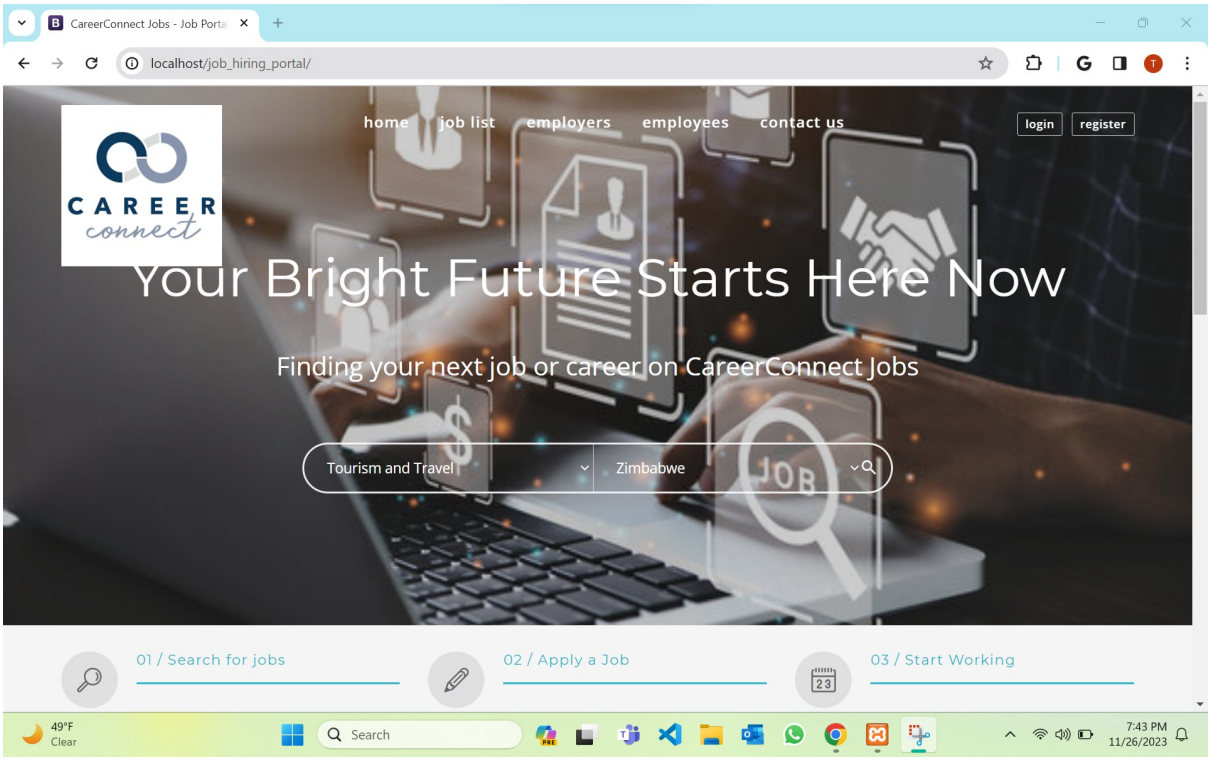
3) Retrieve the qualification of applicants registered for jobs

SELECT tbl_jobs.job_id, tbl_job_applications.member_no,
tbl_professional_qualification.certificate
FROM tbl_jobs
LEFT JOIN tbl_job_applications ON tbl_jobs.job_id=tbl_job_applications.job_id
LEFT JOIN tbl_professional_qualification ON
tbl_job_applications.member_no=tbl_professional_qualification.member_no
WHERE tbl_jobs.job_id = '5609558107';

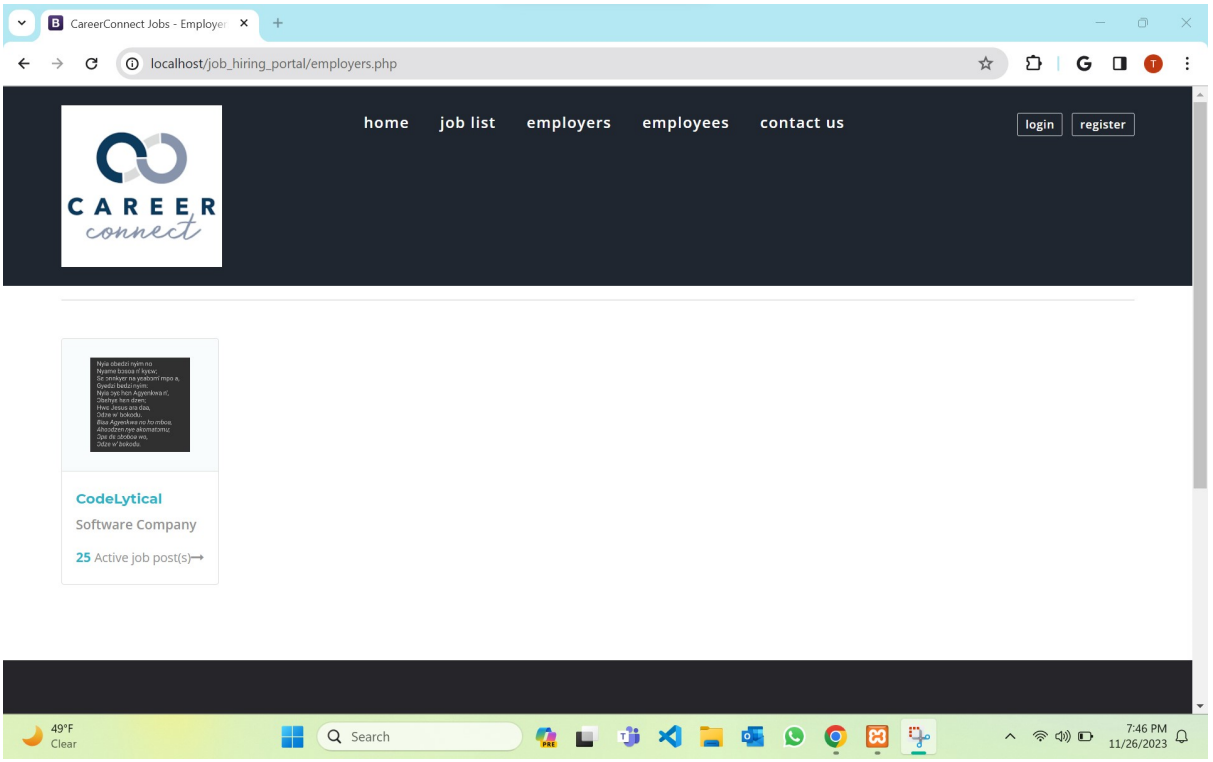


WEBSITE LAYOUT:

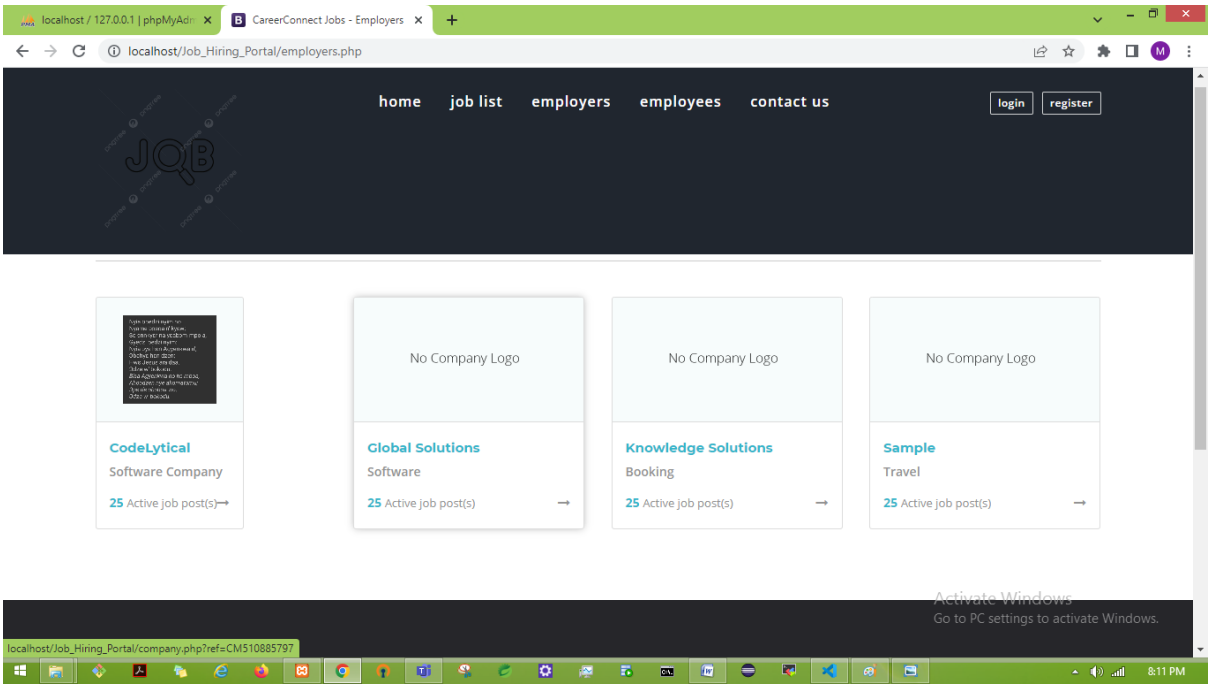
HOME page:



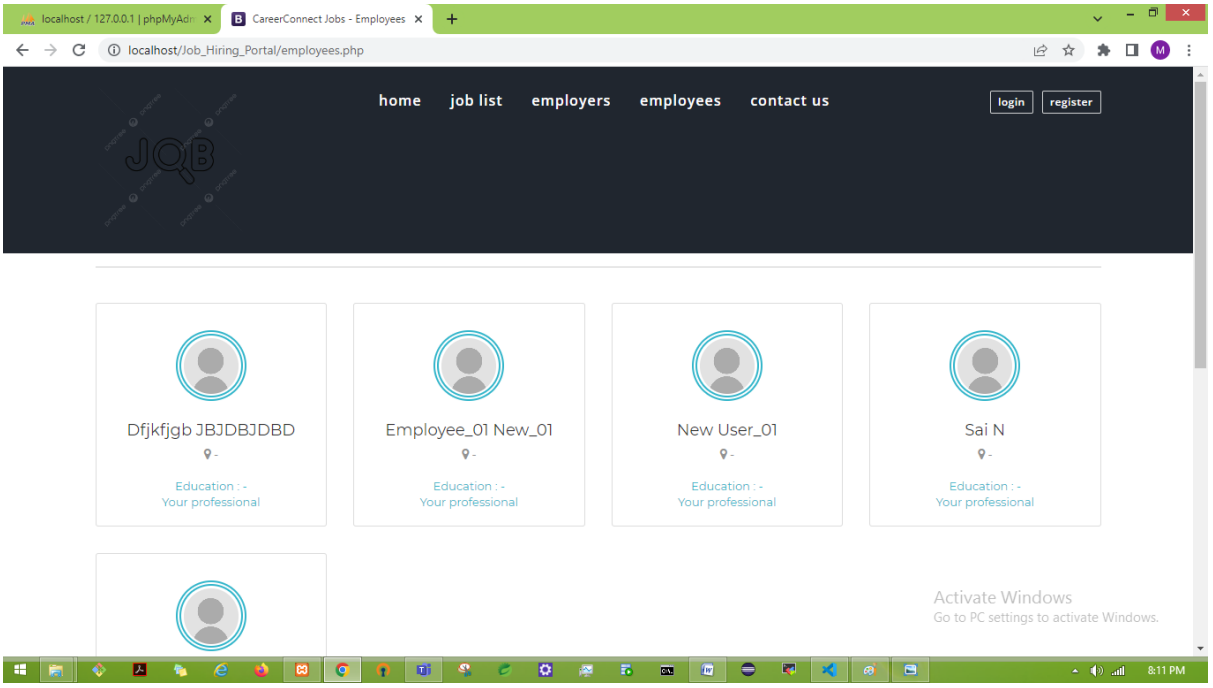
JOB LIST page:



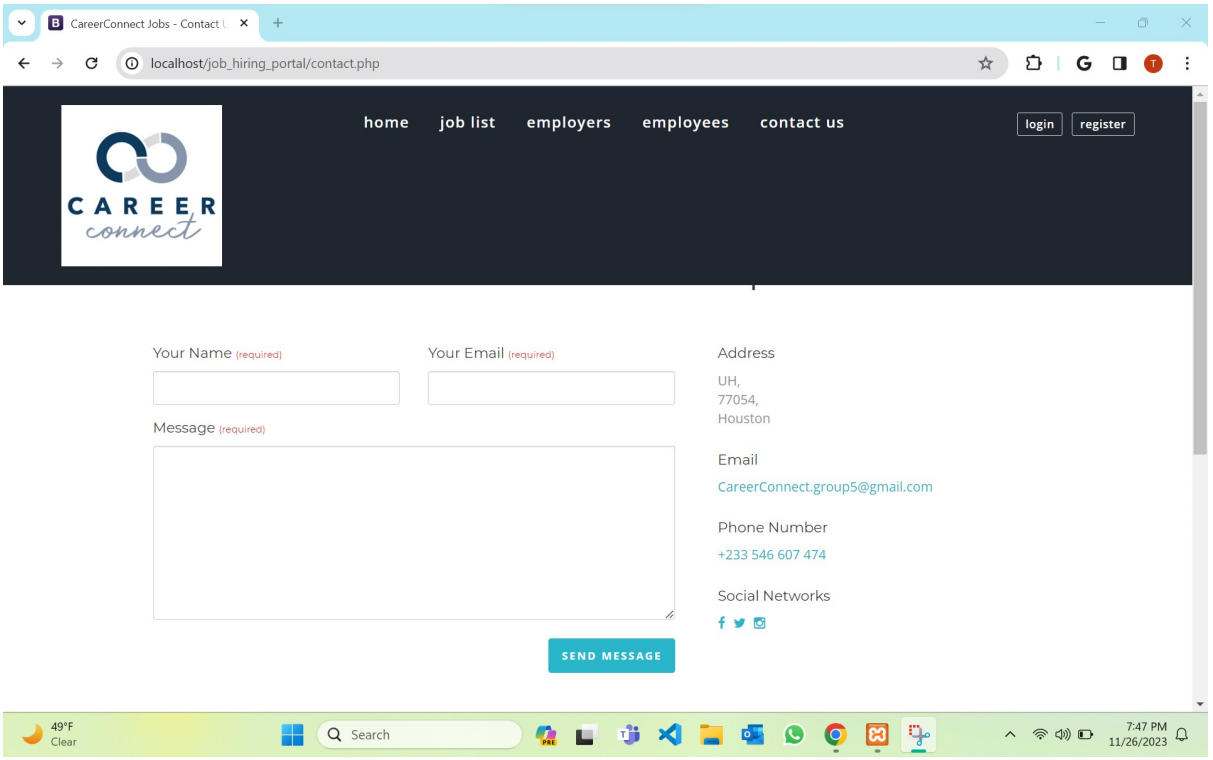
EMPLOYERS page:



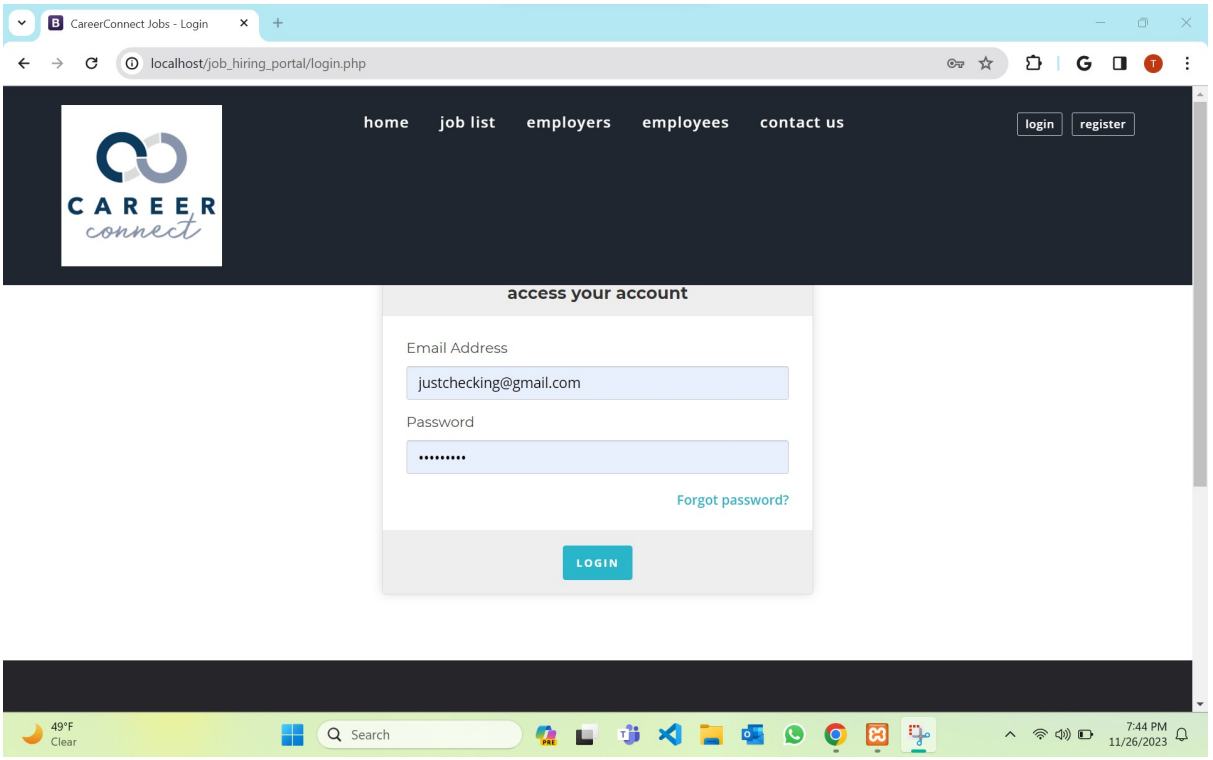
EMPLOYEES page:



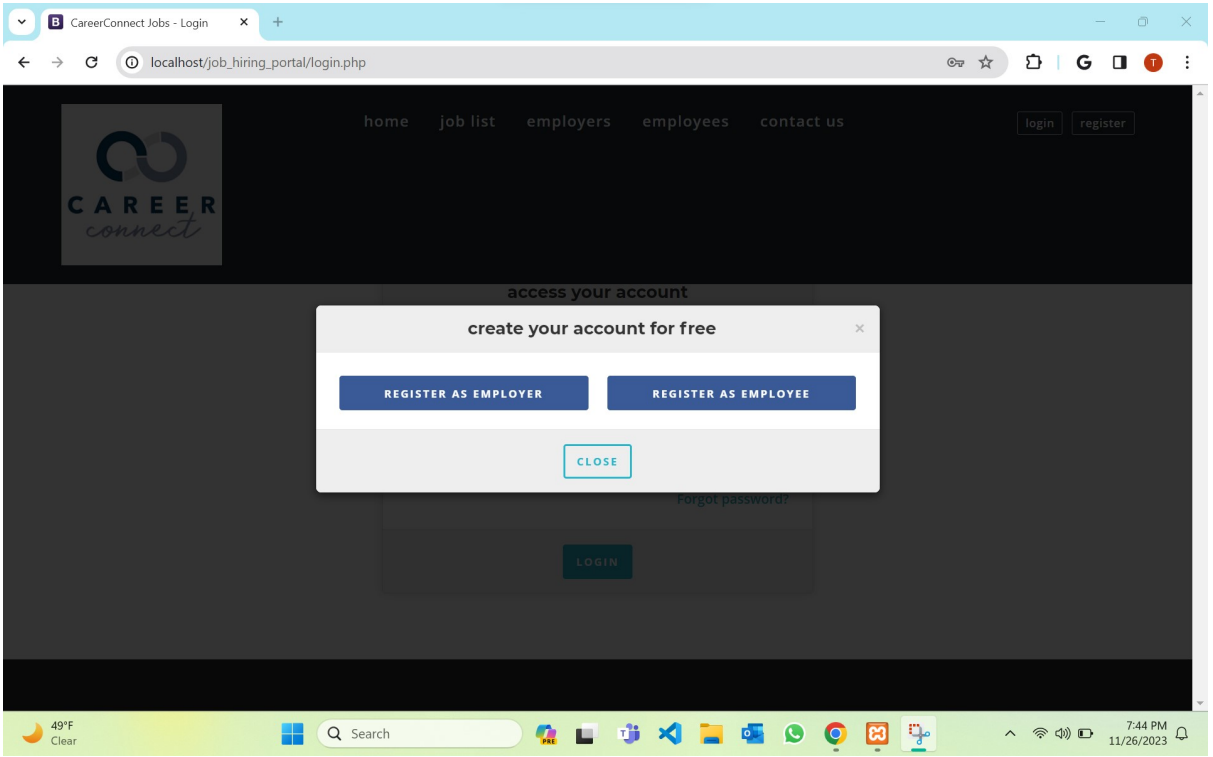
CONTACT US page:



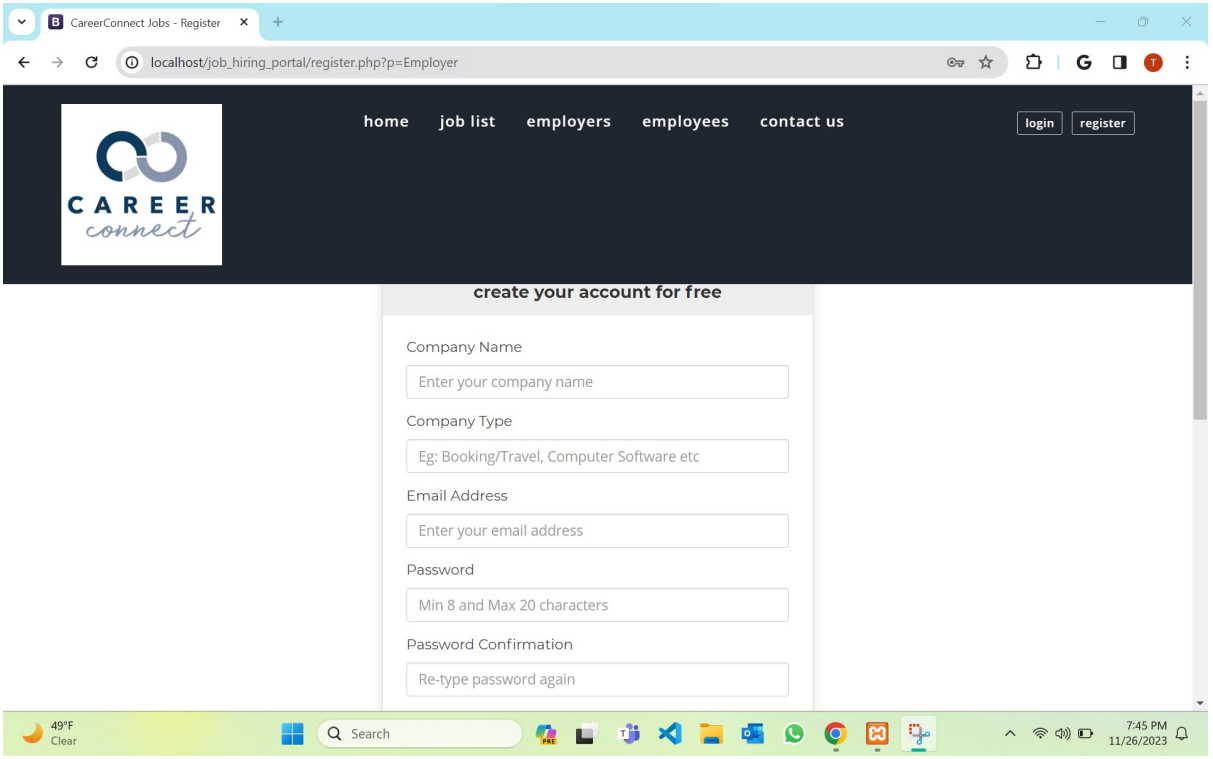
LOGIN page:



REGISTER page:



CREATING AN ACCOUNT:



CONCLUSION

To end briefly, our project to develop a web-based Job Hiring Platform with a database management system has been driven by a commitment to enhancing the efficiency and effectiveness of the job hiring process. By following our business rules, we ensure data integrity and security, crucial aspects in any professional platform.

The ERD we presented reflects a thoughtful design that captures the ins and outs of user profiles, job listings, and associated entities, laying the foundation for a comprehensive and interconnected database. The entities, each serving a specific purpose, are linked through carefully established relationships, guaranteeing a seamless user experience for both job seekers and employers. Moreover, our metadata breakdown provides a clear overview of the data types, requirements, and key relationships, offering transparency in database management.

Our workflow, encompassing the database creation process, validation queries, and data entry screenshots, showcases the practical implementation of our database design. The systematic approach we followed enabled us to make maximum use of our robust technological stack.

The inclusion of diverse validation queries demonstrates our thorough testing and validation process, ensuring the reliability of our platform. By adhering to the project guidelines and not compromising on detail, we have developed a sophisticated job hiring platform that has the potential to improve the way job seekers connect with employers.

MEMBERS AND THEIR CONTRIBUTIONS

Each team member played a vital role in the project's success. We brought our unique skills and perspectives to the table. The collaborative efforts of each team member resulted in a well-rounded and comprehensive Job Hiring Platform that addresses the diverse needs of job seekers and employers alike.

- **Nagendra Reddy Palugulla**: Led the database design, implemented ERD, and contributed to the overall architecture.
- **Harikesh Govindaiahgar**i: Collaborated on the database design, contributed to the frontend development, and ensured a cohesive tech stack.
- **Tharun Pasham**: Focused on frontend development, creating an intuitive and user-friendly UI for the platform.
- **Sai Divya Sree Nagapudi Venugopal**: Collaborated on integrating the front-end with the database along with making changes to the code
- **Abhigna Sowgandhika Vadlamudi**: Ensured the data integrity, implemented validation queries, and contributed to the security aspects of the project.
- **Ahmed Hussain Syed**: Contributed to the security aspects of the platform, ensuring a secure user authentication process.
- **Sheema Anush**: Contributed to the front-end development and led project management, ensuring coordination among team members and overseeing the overall progress of the project.

REFERENCES

GIT REPOSITORY WITH WEBSITE SOURCE CODE

1. Author/Entity: Laravel
   Year: (2023)
   Title: Laravel Framework
   URL: https://github.com/laravel/laravel

2. Author/Uploader:Tech Tips Unlimited. (Sep 5, 2021).
   Title of Video: Online Job Portal using Asp.Net C# and Sql Server.
   Site Name: YouTube.
   URL: [URL of the video]
3. Author/Instructor: Web Coding.
   Title of Course: PHP Laravel 10 for 2023: Build Complete Job Portal.
   Platform: Udemy.
   URL: [URL of the course]