University of
Sadat City

Faculty of Computers
and Artificial
Intelligence

TRAINING REPORT

ON Back-End Using php,laravel

Submitted to

Faculty of Computers and Artificial Intelligence, University of Sadat City

In partial fulfillment of the requirements for the award of the degree

BACHELOR OF Computer Sciences

(Semester - VI)

Place : ITI (information technology institute) Menofia

Date : 15/7/2024

Student name : Abanoub bekhiet Abdallah Saheon

ID :2231001

Student signature :

University of
Sadat City

Faculty of Computers
and Artificial
Intelligence

Introduction

Back-End development is crucial part of any application Now days there are a variety of tools to make and maintain the back-end part of your application or website ,one of these tools is the framework created using php programming language is one of the best frameworks that can deal with databases especially the relational ones like mysql,postgresql

CONTENT

1 – HTML

- HTML is the standard markup language for creating Web pages.
- All HTML documents must start with a document type declaration: <!DOCTYPE html>.
- The HTML document itself begins with <html> and ends with </html>.
- The visible part of the HTML document is between <body> and </body>.
- Content in html : - **1-element** elements are the building blocks of HTML pages. elements are represented by tags.

An HTML element usually consists of a start tag and end tag, with the content inserted in between: <tagname>Content goes here...</tagname> for example:

<h1> My First Heading </h1>
My first paragraph. **notes:**

HTML elements with no content are called empty elements. Empty elements do not have an end tag, such as the
 element (which indicates a line break).

2-Attribute

Attributes provide additional information about HTML elements.

All HTML elements can have attributes

Attributes provide additional information about an element

Attributes are always specified in the start tag.

3-heading

Headings are defined with the <h1> to <h6> tags.

<h1> defines the most important heading. <h6>

4-paragraph

The HTML <p> element defines a paragraph:

5-formating

HTML uses elements like <b> and <i> for formatting output, like bold or italic text.

Formatting elements were designed to display special types of text.

6-image

Images can improve the design and the appearance of a web page. in HTML, images are defined with the <img> tag.

7-tables

An HTML table is defined with the <table> tag.

Each table row is defined with the <tr> tag. A table header is defined with the <th> tag. By default, table headings are bold and centered. A table data/cell is defined with the <td> tag. **8-links**

Links are found in nearly all web pages. Links allow users to click their way from page to page.

HTML links are hyperlinks.

You can click on a link and jump to another document.

9-lists

1-unordered list

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.

The list items will be marked with bullets (small black circles) by default: 2-orderd list

An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.

The list items will be marked with numbers by default:

Use the HTML <ul> element to define an unordered list

Use the CSS list-style-type property to define the list item marker **10-forms**

The HTML <form> element defines a form that is used to collect user input:

<form>form elements</form>

An HTML form contains form elements.

Form elements are different types of input elements, like text fields, checkboxes, radio buttons, submit buttons, and more. **11-Html5**

The most interesting new HTML5 elements are:

New semantic elements like <header>, <footer>, <article>, and <section>.

New attributes of form elements like number, date, time, calendar, and range.

New graphic elements: <svg> and <canvas>.

New multimedia elements: <audio> and <video>.

Notes :

We learn in training more content in html but I write simple content .

2- CSS & CSS3

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

- CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes. Content in CSS : **1-syntax** selector{ property:value; property:value;

}

The selector points to the HTML element you want to style.

Example

```
p {     color: red;
text-align: center;
}
```

2-color

Colors are specified using predefined color names, or RGB, HEX, HSL, RGBA, HSLA values.

In HTML, a color can be specified by using a color name : red , green , black

Example

```
<h1 style="color:Tomato;">Hello World</h1>
```

3-Backgrounds he CSS background properties are used to define the background effects for elements.

4-margin

The CSS margin properties are used to create space around elements, outside of any defined borders.

With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (top, right, bottom, and left).

5-padding

The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

With CSS, you have full control over the padding. There are properties for setting the padding for each side of an element (top, right, bottom, and left).

6-height / width

The height and width properties are used to set the height and width of an element.

The height and width can be set to auto (this is default. Means that the browser calculates the height and width), or be specified in length values, like px, cm, etc., or in percent (%) of the containing block.

7-Box Model

All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content.

8-Fonts / icons

The CSS font properties define the font family, boldness, size, and the style of a text.

The font family of a text is set with the font-family property.

9-position

The position property specifies the type of positioning method used for an element (static, relative, fixed, absolute or sticky).

10-float

The CSS float property specifies how an element should float.

The CSS clear property specifies what elements can float beside the cleared element and on which side.

11-Opacity

The opacity property specifies the opacity/transparency of an element.

The opacity property can take a value from 0.0 - 1.0. The lower value, the more transparent

12-Border

The CSS border properties allow you to specify the style, width, and color of an element's border.

The border-style property specifies what kind of border to display.

13-Transation

CSS transitions allows you to change property values smoothly (from one value to another), over a given duration.

14-animation

CSS animations allows animation of most HTML elements without using JavaScript or Flash!

15-FlexBox

display Specifies the type of box used for an HTML element.

16-Grid System

To make an HTML element behave as a grid container, you have to set the display property to grid or inline-grid.

Grid containers consist of grid items, placed inside columns and rows. **Notes :**

We learn in training more content in CSS but I write simple content .

3- JavaScript

JavaScript is the programming language of HTML and the Web.

JavaScript is easy to learn.

This tutorial will teach you JavaScript from basic to advanced.

Why Study JavaScript?

JavaScript is one of the 3 languages all web developers must learn:

1. HTML to define the content of web pages

2. CSS to specify the layout of web pages

3. JavaScript to program the behavior of web pages

Web pages are not the only place where JavaScript is used. Many desktop and server programs use JavaScript. Node.js is the best known. Some databases, like MongoDB and CouchDB, also use JavaScript as their programming language.

JavaScript and Java are completely different languages, both in concept and design.

JavaScript was invented by Brendan Eich in 1995, and became an ECMA standard in 1997.

ECMA-262 is the official name of the standard. ECMAScript is the official name of the language.

You can read more about the different JavaScript versions in the chapter JS Versions.

Content in JavaScript: 1-

Syntax

- JavaScript syntax is the set of rules, how JavaScript programs are constructed.
- How to declare variables.
- How to assign values.
- How to compute values .

2- Comment

- JavaScript comments can be used to explain JavaScript code, and to make it more readable.
- JavaScript comments can also be used to prevent execution, when testing alternative code.

3- Variables

- JavaScript variables are containers for storing data values..
- Using var , let , const . **4- Data type**
- JavaScript variables can hold many data types: numbers, strings, objects , array , null , Boolean. **5- Functions**
- function is a block of code designed to perform a particular task.
- A JavaScript function is executed when "something" invokes it (calls it).

6- Objects

- You have already learned that JavaScript variables are containers for data values.

7- Strings

- strings are used for storing and manipulating text.
- string is zero or more characters written inside quotes

8- Numbers

- JavaScript has only one type of number. Numbers can be written with or without decimals

9- Arrays

- arrays are used to store multiple values in a single variable.

10 – Conditions

- are used to perform different actions based on different conditions.

11 – Loops

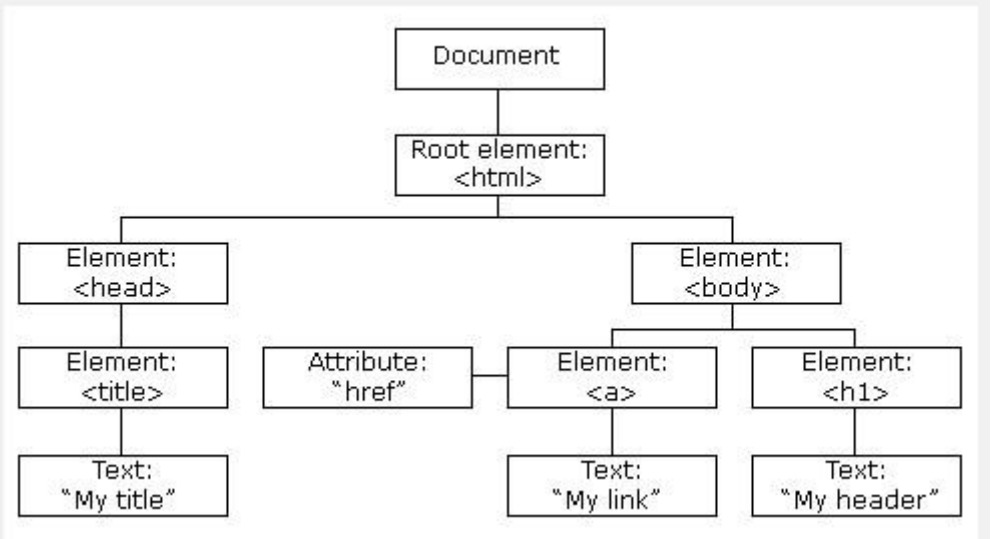
- Loops can execute a block of code a number of times.
- if you want to run the same code over and over again, each time with a different value.

12 – Hosting

- Hoisting is JavaScript's default behavior of moving declarations to the top.
- a variable can be declared after it has been used.
- In other words; a variable can be used before it has been declared

13 – Dom (Document Object Model)

- With the HTML DOM, JavaScript can access and change all the elements of an HTML document.
- When a web page is loaded, the browser creates a Document Object Model of the page.
- The HTML DOM model is constructed as a tree of Objects:



14 – Bom (Browser Object Model)

- There are no official standards for the Browser Object Model (BOM).
- Since modern browsers have implemented (almost) the same methods and properties for JavaScript interactivity, it is often referred to, as methods and properties of the BOM.

15 – Json (JavaScript Object Notation) 16 – Events

- HTML events are "things" that happen to HTML elements.
- When JavaScript is used in HTML pages, JavaScript can "react" on these events

4-SQL

1- Basic SQL Queries:

- SELECT: Retrieve data from tables.
- WHERE: Filter data based on conditions.

2-SQL Clauses:

- ORDER BY: Sort data in ascending or descending order.
- GROUP BY: Group data with the same values.
- HAVING: Filter groups created by GROUP BY.

3- Joins:

- INNER JOIN: Returns matching records from both tables.
- LEFT JOIN: Returns all records from the left table and matching records from the right table.
- RIGHT JOIN: Returns all records from the right table and matching records from the left table.
- FULL JOIN: Returns all records when there is a match in either table.

4-Aggregate Functions:

- COUNT: Count the number of rows.
- SUM: Calculate the total of values.
- AVG: Calculate the average of values.
- MAX/MIN: Find the maximum or minimum value.

5-Data Manipulation Language (DML):

- INSERT: Add new records.
- UPDATE: Modify existing records.
- DELETE: Remove records.

6-Data Definition Language (DDL):

- CREATE: Create new databases, tables, or objects.
- ALTER: Modify existing database objects.
- DROP: Delete tables, databases, or objects.

7-Constraints:

- PRIMARY KEY: Uniquely identify each record.
- FOREIGN KEY: Link records between tables.
- NOT NULL: Ensure column values cannot be null.
- UNIQUE: Ensure all values in a column are unique.

5-php

1-Basic Syntax:

- **PHP scripts start with <?php and end with ?>.**
- **Case-sensitive variables prefixed with $.**

2- Variables and Data Types:

- **PHP supports data types like integers, floats, strings, arrays, and objects.**
- **Type juggling (implicit type conversion).**

3-Constants:

- **Define constants using the define() function.**
- **Constants cannot be changed once set.**

4- Operators:

- **Arithmetic, assignment, comparison, logical, and string operators.**
- **Increment/Decrement operators.**

5- Control Structures:

- **Conditional Statements: if, else, elseif, switch.**

- **Loops: for, while, do-while, foreach.**

6- Functions:

- **User-defined and built-in functions.**
- **Pass by value vs pass by reference.**
- **Variable scope (local, global, and static).**

7- Arrays:

- **Indexed arrays, associative arrays, and multidimensional arrays.**
- **Array functions for manipulation like sorting, merging, and searching.**

8- Superglobals:

- **Predefined variables: $_GET, $_POST, $_SESSION, $_COOKIE, $_SERVER, $_FILES, $_ENV.**
- **Used for accessing global data such as form inputs or environment settings.**

9- Forms and Handling User Input:

- **Form submission using GET or POST methods.**
- **Data sanitization and validation.**

10- Sessions and Cookies:

- **Session handling using $_SESSION.**
- **Cookie management with setcookie() and $_COOKIE.**

11-File Handling:

- **Open, read, write, append, and close files.**
- **Checking if files exist and handling file uploads.**

12-Error Handling:

- **Using try-catch blocks for exceptions.**
- **Triggering custom errors using trigger_error().**

13-Object-Oriented Programming (OOP):

- **Classes and Objects: Define classes and instantiate objects.**
- **Properties and Methods: Accessing class attributes and methods.**
- **Inheritance: Parent-child relationships between classes.**
- **Encapsulation: Use of access modifiers (public, private, protected).**
- **Polymorphism: Method overriding and interfaces.**
- **Abstraction: Abstract classes and methods.**

- **Magic Methods: Special methods like __construct(), __destruct(), and __toString().**

14-Namespaces:

- **Organizing code into different namespaces to avoid naming conflicts.**

15-Date and Time:

- **Manipulating date and time using functions like date(), time(), and strtotime().**

16- Database Interaction (MySQL):

- **Connecting to databases using PDO or mysqli.**
- **CRUD operations (Create, Read, Update, Delete).**
- **Prepared statements for SQL injection protection.**

17-Regular Expressions:

- **Pattern matching using preg_match(), preg_replace(), and similar functions.**

18-File Inclusion:

- **Including PHP files with include() and require() functions.**
- **Conditional includes with include_once() and require_once().**

19-JSON Handling:

- **Encoding data into JSON format using json_encode().**
- **Decoding JSON data using json_decode().**

20-Error Reporting and Debugging:

- **Configuring error reporting levels.**
- **Using var_dump(), print_r(), and debug_backtrace() for debugging.**

21-Web Services and APIs:

- **Sending HTTP requests using cURL.**
- **Handling API responses and interacting with RESTful services.**

6-laravel

1. Introduction to Laravel:

- **PHP framework following the MVC (Model-View-Controller) architecture.**
- **Focuses on simplicity, elegance, and a robust toolkit for web development.**

2. Installation:

- **Requires Composer for installation.**
- **Set up using Laravel Installer or composer create-project.**

3. Directory Structure:

- **App: Contains application logic like Models, Controllers, Middleware.**
- **Config: Configuration files.**
- **Routes: Defines web routes (web.php) and API routes (api.php).**
- **Database: Migrations, seeders, and factories.**
- **Resources: Views, assets, and language files.**

- **Public: Publicly accessible files (e.g., CSS, JS, images).**

4. Routing:

- **Defined in routes/web.php and routes/api.php.**
- **Supports HTTP verbs: GET, POST, PUT, DELETE.**
- **Route parameters and named routes.**

5. Controllers:

- **Handle the application logic.**
- **Created to manage routes and actions.**
- **Resource controllers for CRUD operations.**

6. Views:

- **Uses Blade templating engine.**
- **Blade syntax includes template inheritance, control structures (@if, @foreach), and components.**
- **Views are stored in the resources/views directory.**

7. Models:

- **Represents database tables.**
- **Uses Eloquent ORM for database interaction.**

- **Define relationships like one-to-many, many-to-many, and hasMany.**

8. Migrations:

- **Version control for database structure.**
- **Create, modify, or delete database tables.**
- **Schema building methods for defining table structure.**

9. Eloquent ORM:

- **Object-Relational Mapping (ORM) to interact with the database.**
- **Queries using methods such as all(), find(), where(), and first().**
- **Eloquent relationships (e.g., belongsTo, hasMany, belongsToMany).**

10. Middleware:

- **Filter HTTP requests.**
- **Common middleware includes auth, throttle, and csrf.**
- **Custom middleware for additional request filtering.**

11. Requests and Validation:

- **Capture user input via form requests.**

- **Built-in validation rules using validate().**
- **Custom validation rules can be created.**

12. Authentication:

- **Built-in authentication system.**
- **Auth scaffolding using packages like Breeze, Jetstream.**
- **Supports user login, registration, and password reset.**

13. Authorization:

- **Gate and Policy classes for user permissions.**
- **Define who can perform specific actions in the application.**

14. Database Seeding:

- **Seeders allow populating tables with dummy data.**
- **Factories generate test data for models.**

16. Error Handling:

- **Customizable error handling and reporting.**
- **Automatically handles exceptions and logs errors.**

19. API Development:

- **Support for building RESTful APIs.**
- **API routes are defined in routes/api.php.**
- **Use of resources for transforming models into JSON responses.**

21. Artisan Console:

- **Command-line interface for running tasks.**
- **Common commands: php artisan migrate, php artisan make:model.**
- **Custom commands can be created.**

24. Configuration:

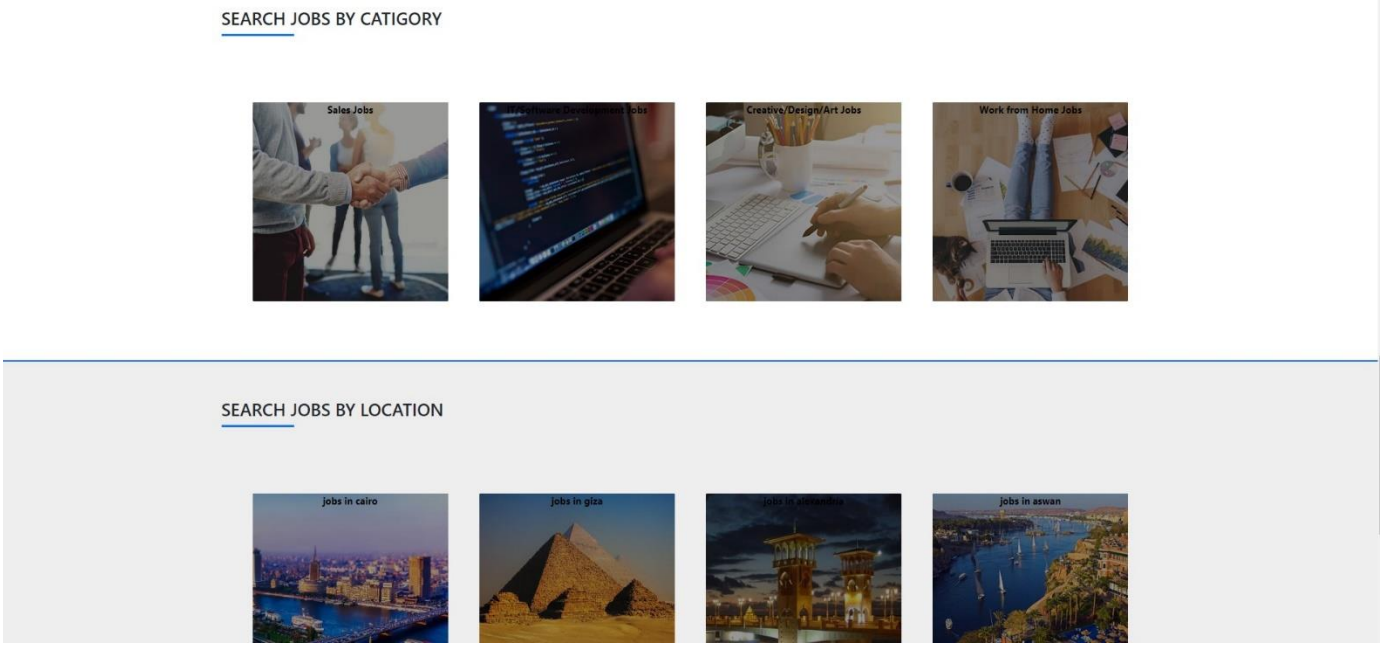
- **Config files located in the config/ directory.**
- **Environment variables stored in the .env file.**

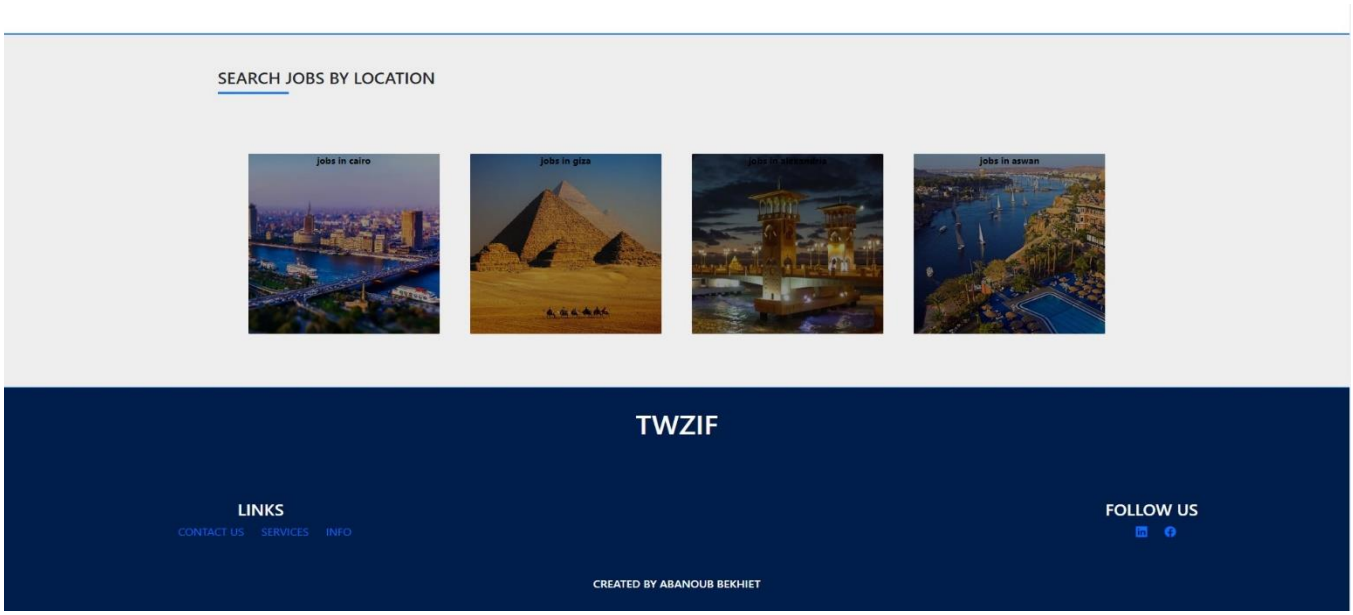
25. Session Management:

- **Tracks user data across multiple requests.**
- **Session drivers include file, cookie, database, and Redis.**

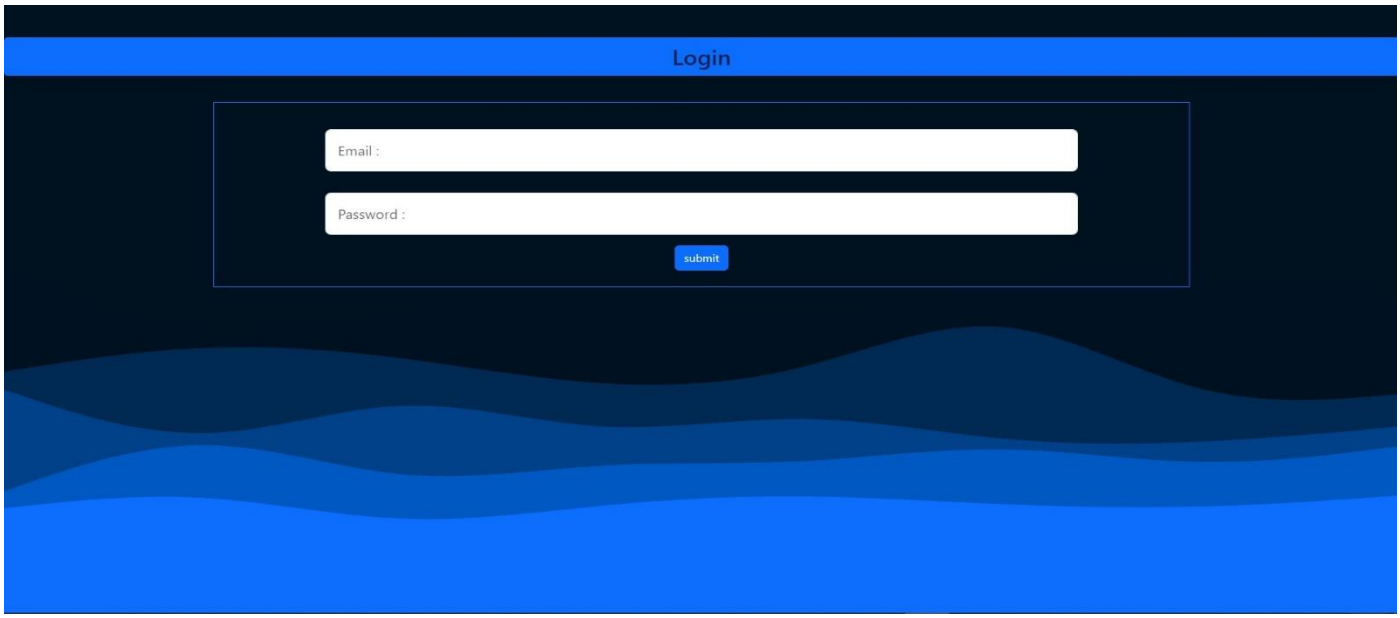
The project content

1-the home page

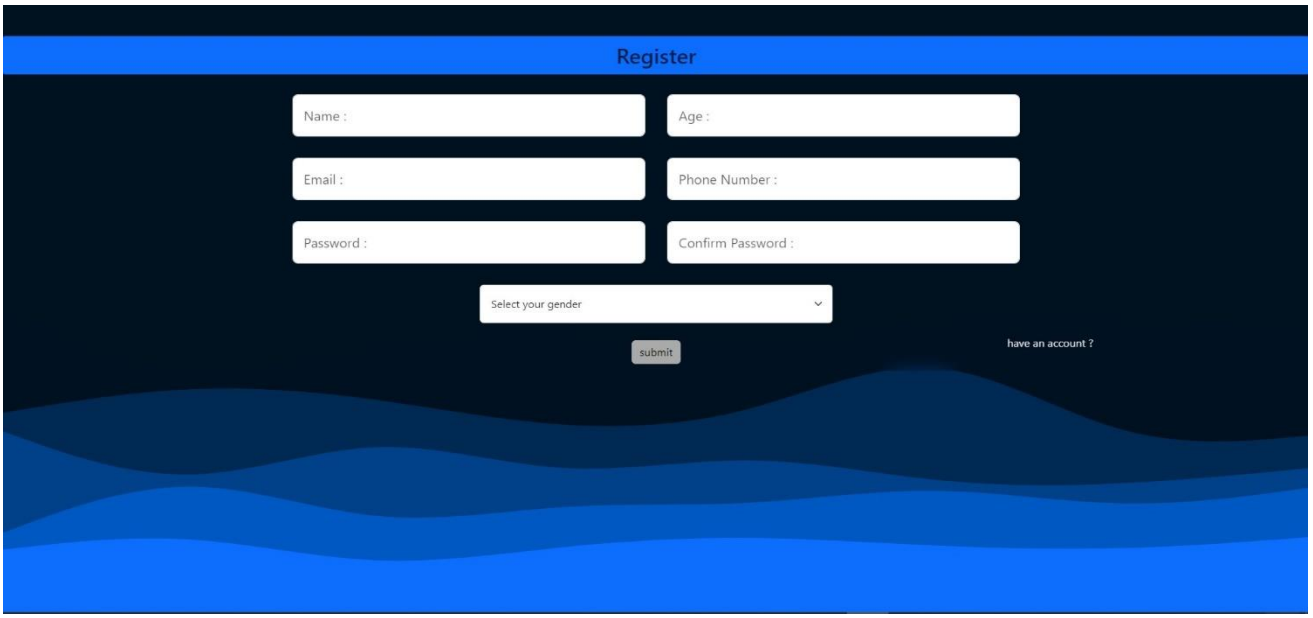



SEARCH JOBS BY LOCATION
jobs in cairo
jobs in giza
jobs in aswan
TWZIF
LINKS
CONTACT US
SERVICES
INFO
FOLLOW US
CREATED BY ABANOUB BEKHIET

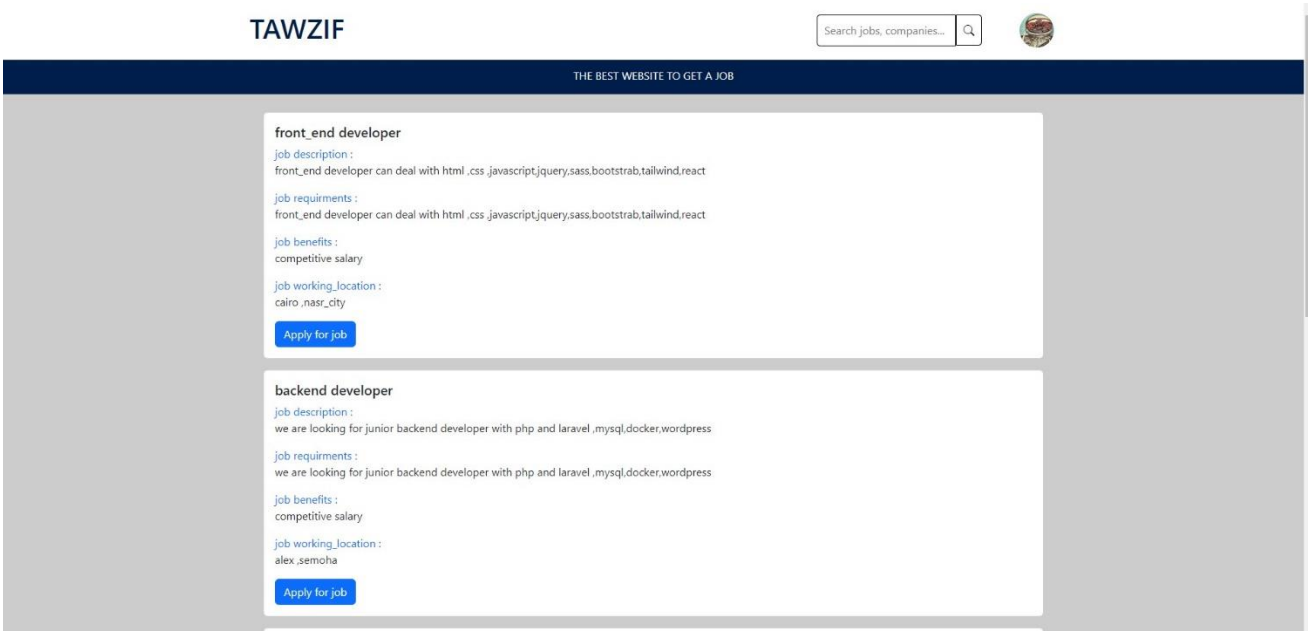
2-login


Login
Email :
Password :
submit

3-Register



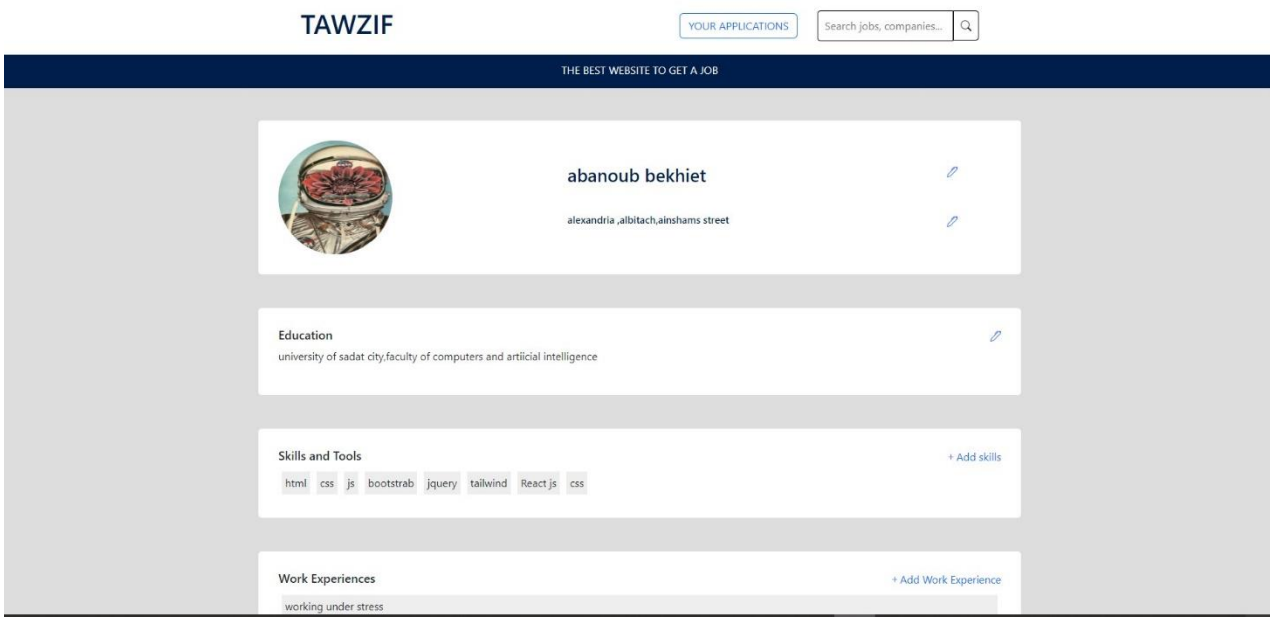
4-candidate after login (person who search for a job)

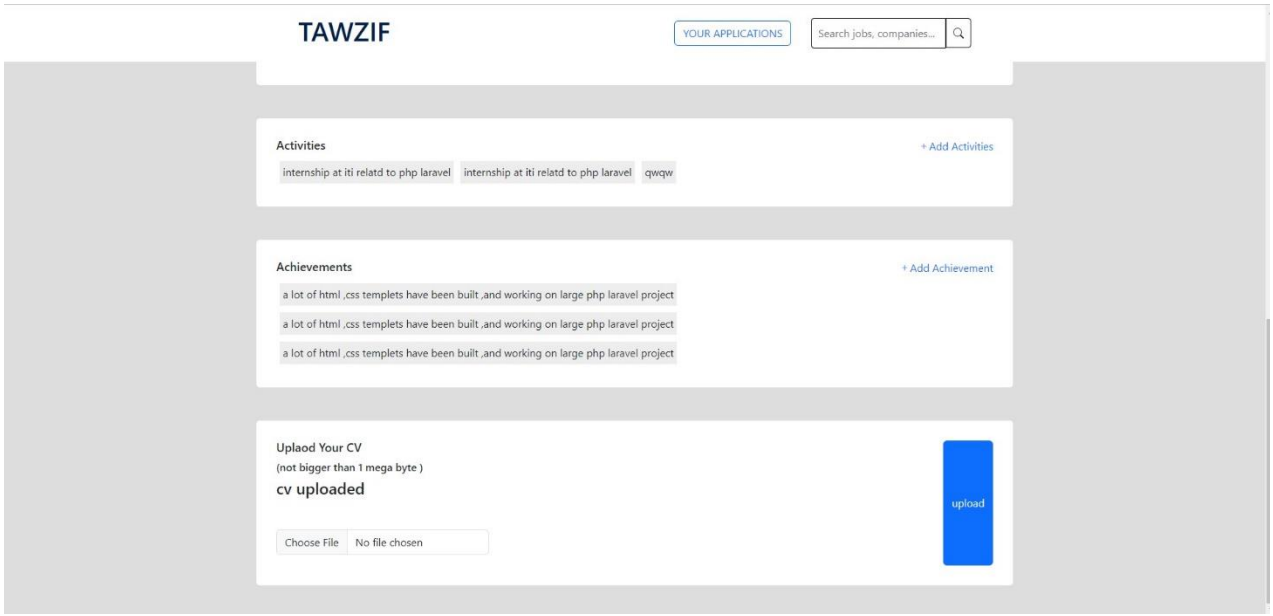


After candidate login been redirected to the page of all jobs in the website from all fields but in the top

of the page there is a search to find jobs in any field he want

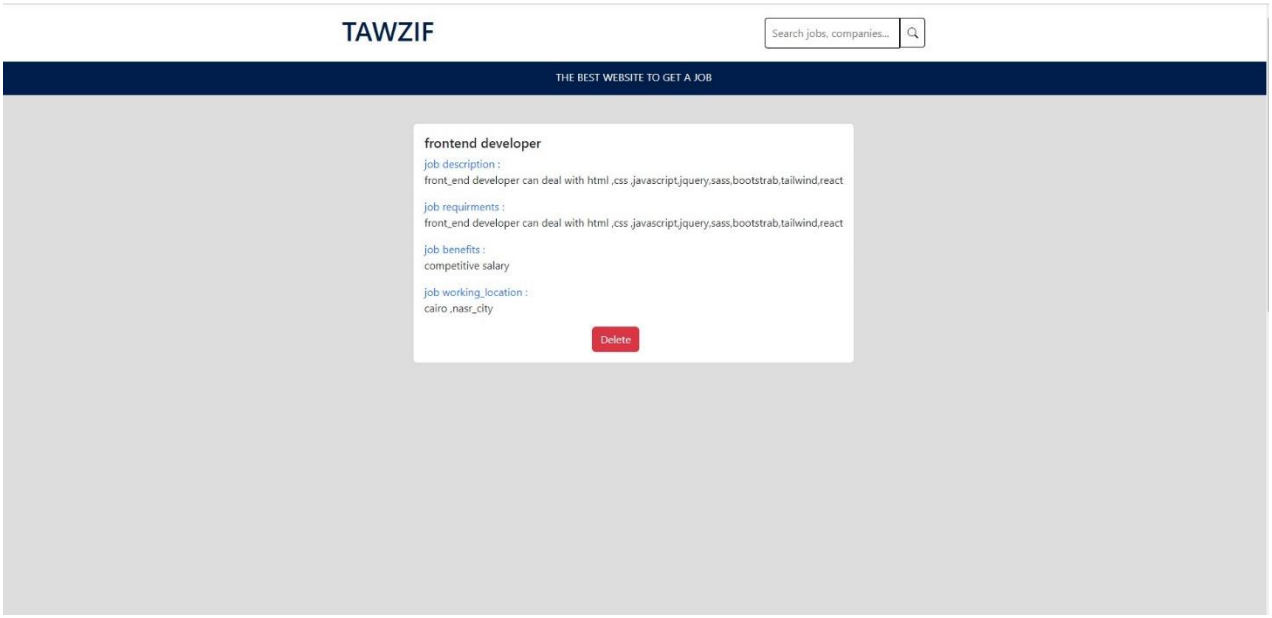
5- candidate profile





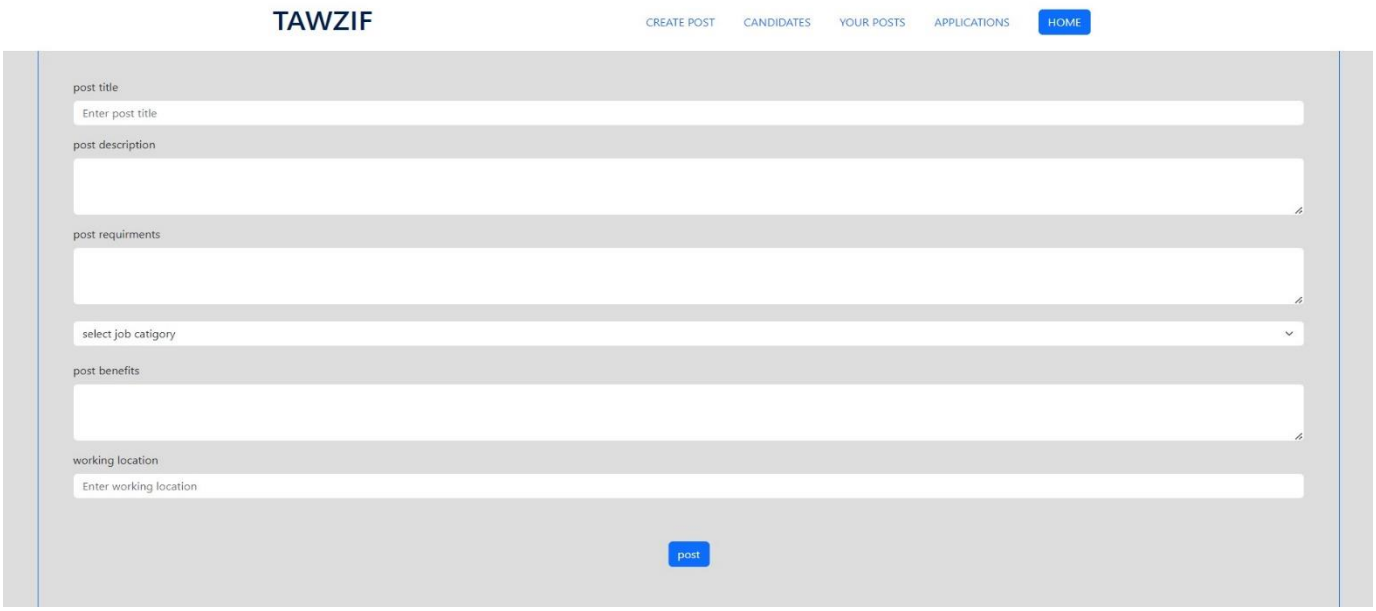
And the candidate has also a profile he can upload his CV ,add education, skills, experience ,achievements ,his last projects ,his photo ,email ,and his name

6- candidate applications



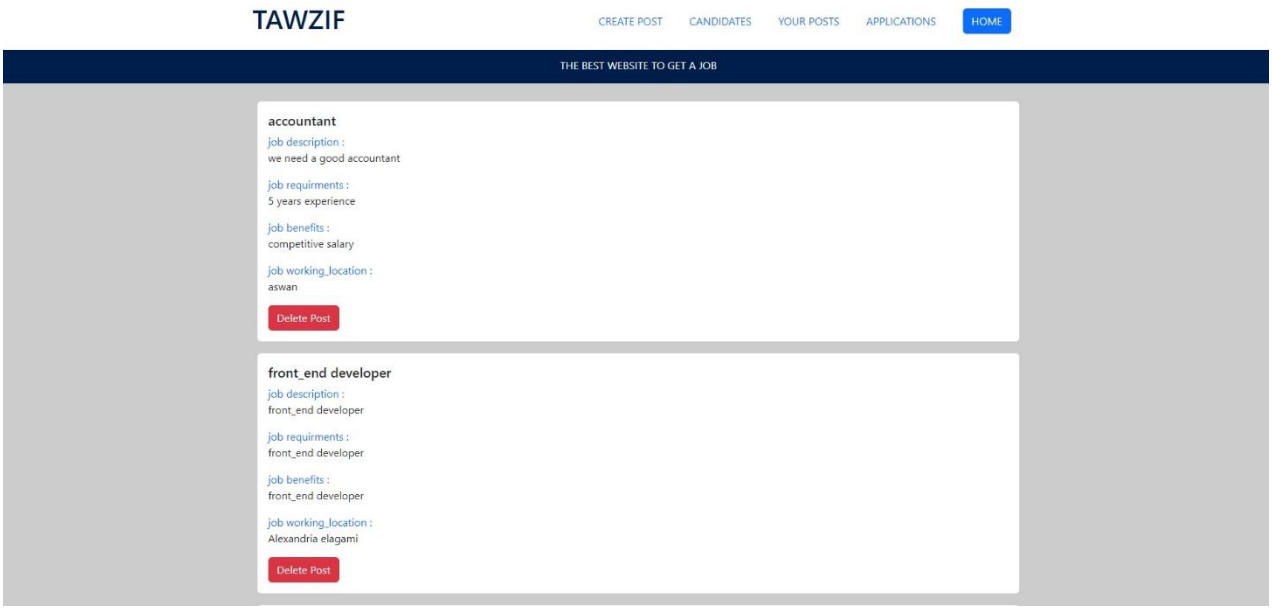
There is also a page that display the candidate applications for the jobs and can delete this applications

7- employee (a person who post jobs) after login



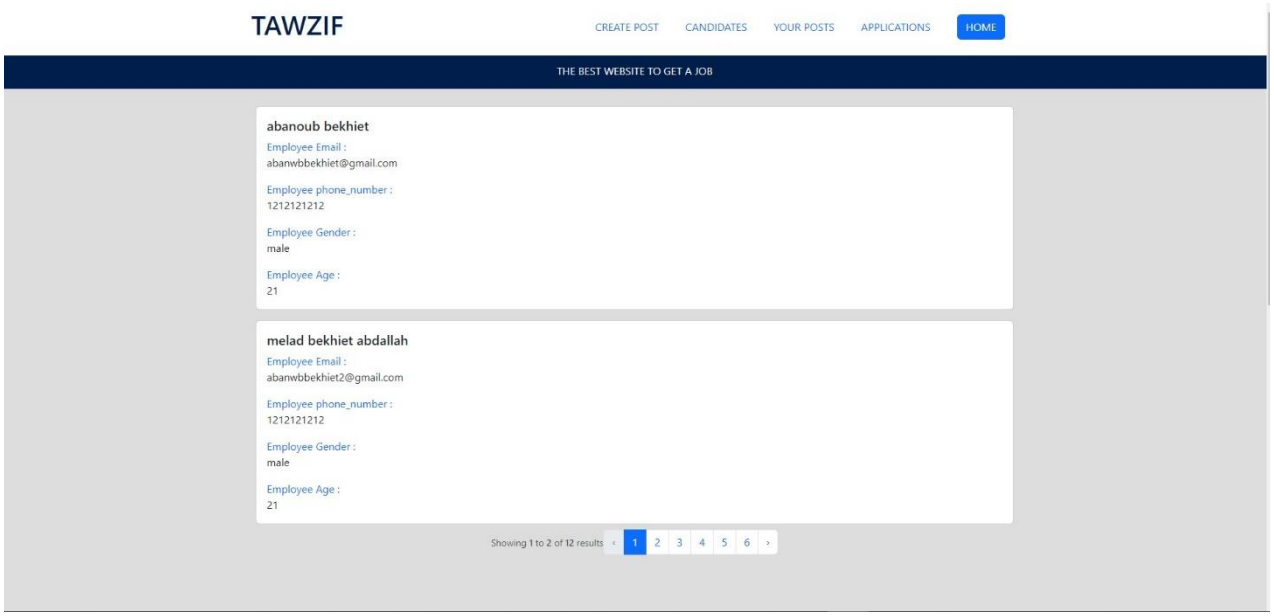
After login he been redirected he a page to create post of a job and the post after that appears in the admin page waiting the admin to accept or reject it

8- employee posts



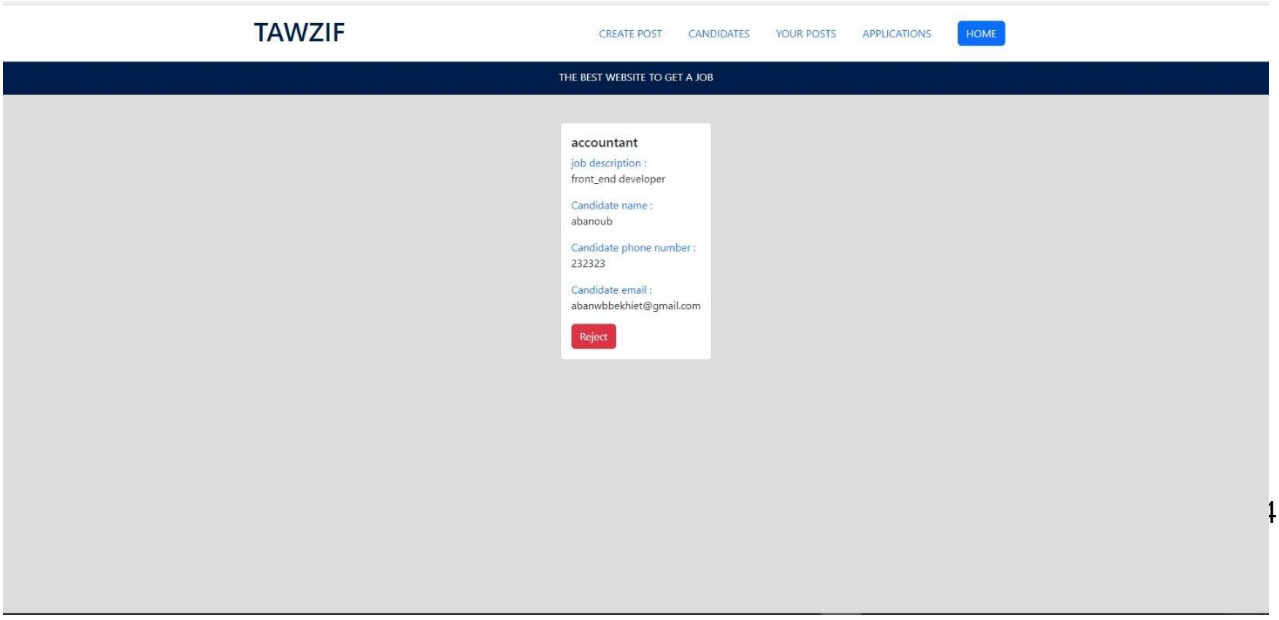
Employee also can see his last posts and can delete them

9-list of candidates profiles in the employee page



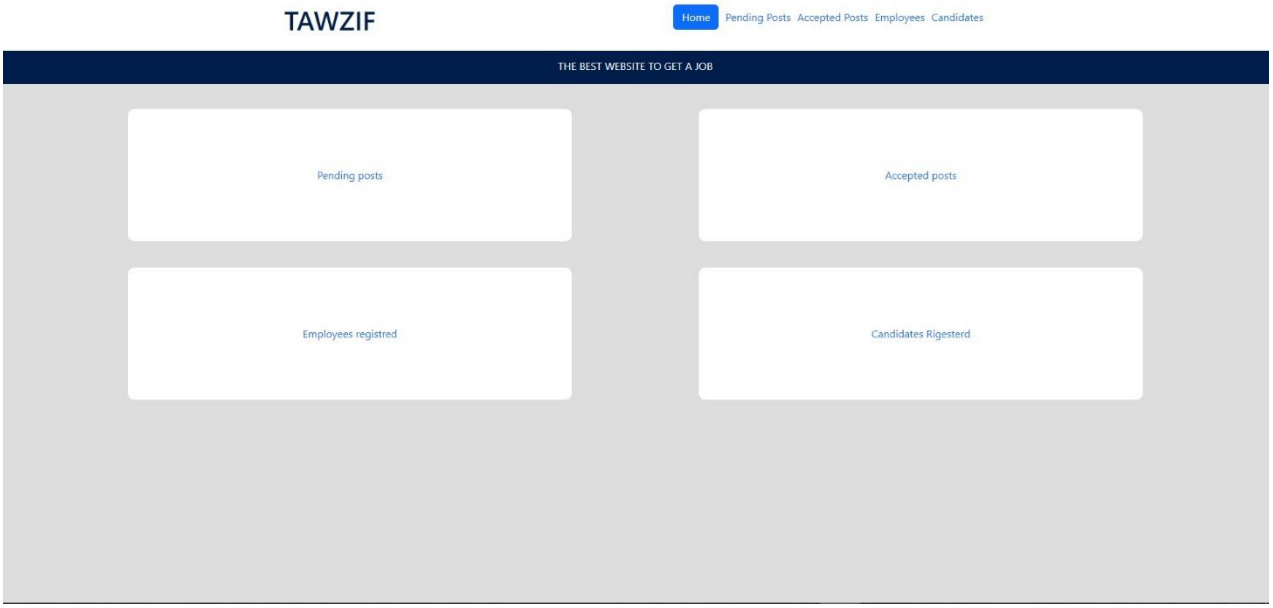
Employee can see the candidates profiels and can contact with them

10- candidates applications



Employee can see the applications of the candidates they post them

11-admin dashboard



After the login of the admin he is been redirected to the dashboard

That make he able to :

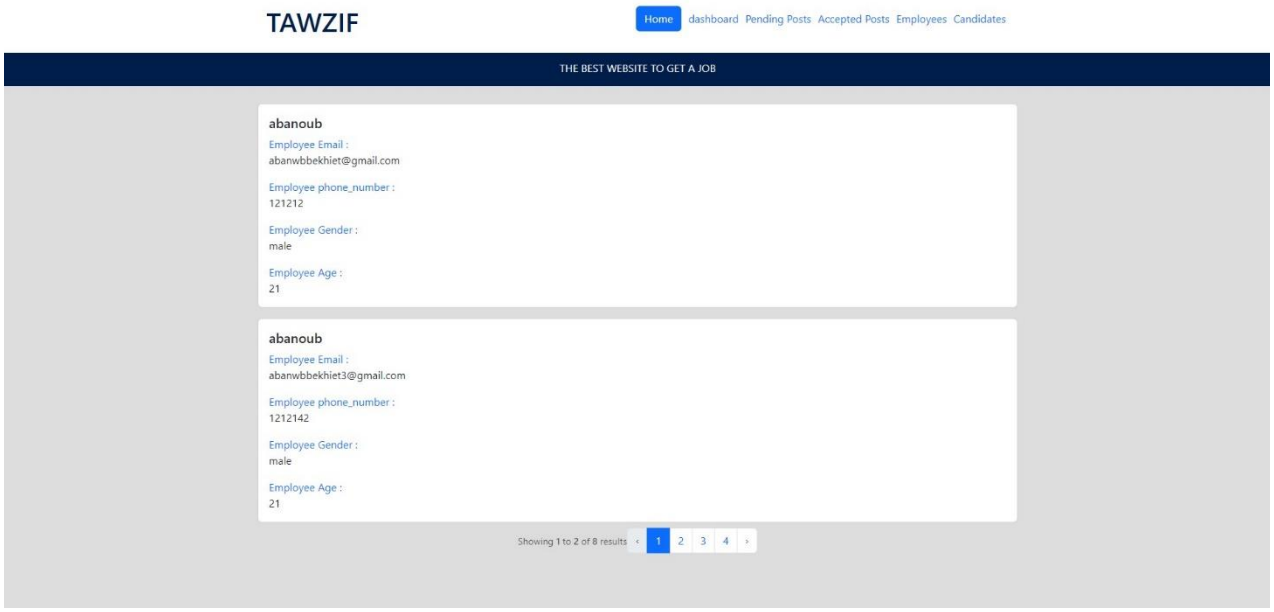
1-see the lists of pending posts and can accept or reject posts of employees

2- see list of the employees profiles

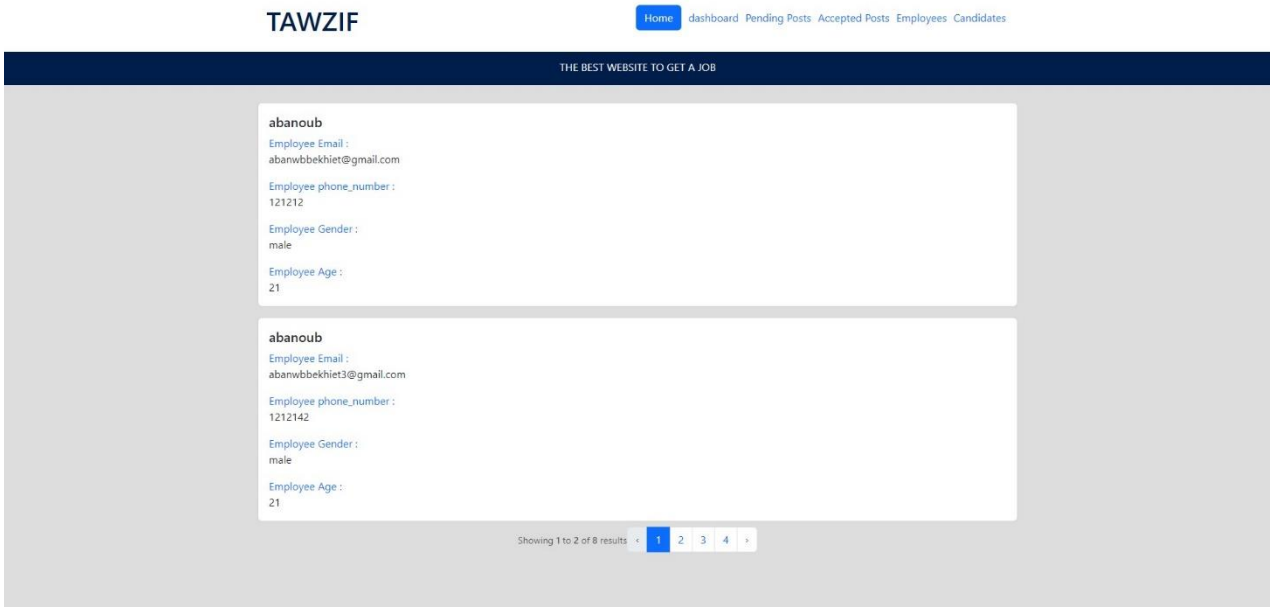
3-see list of the candidates profiles

4-see the list of accepted posts

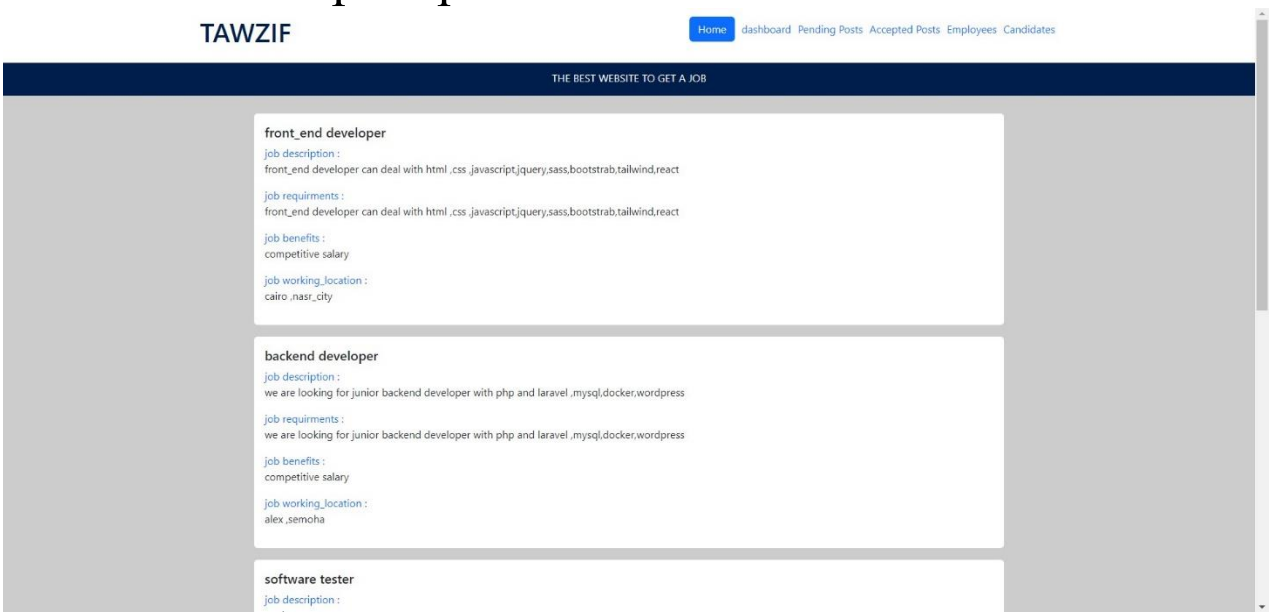
12-list of candidates admin can see



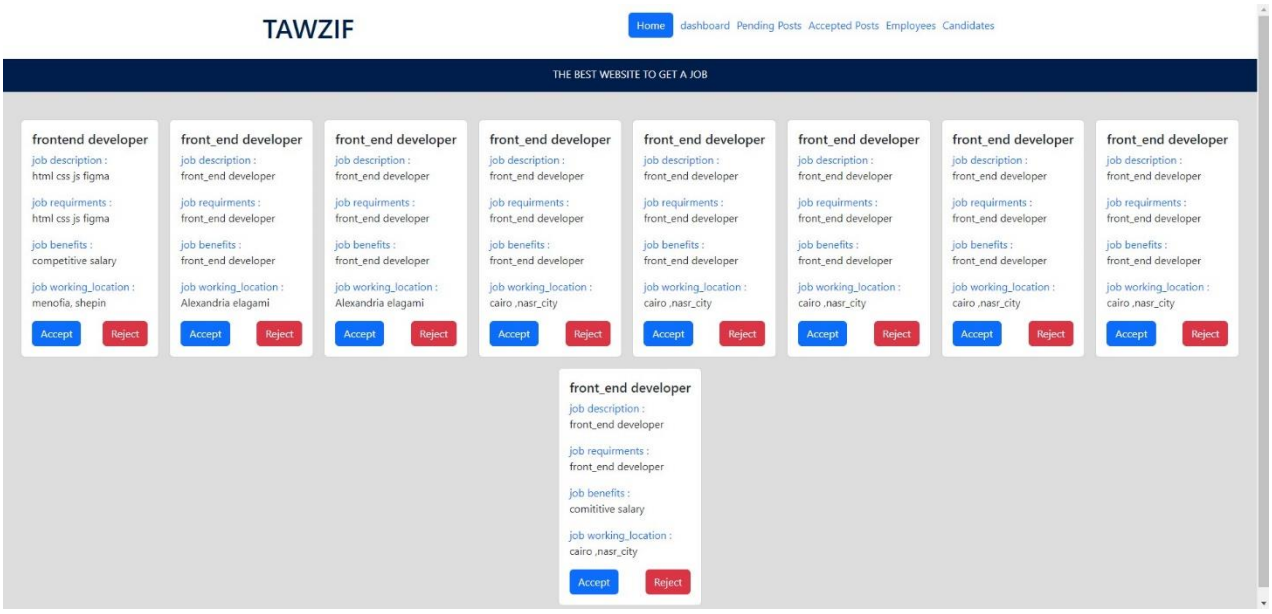
13-list of employees admin can see



14- list of accepted posts admin can see it



15-list of pending posts that admin can reject or accept



16- form for applying for a job using the candidate

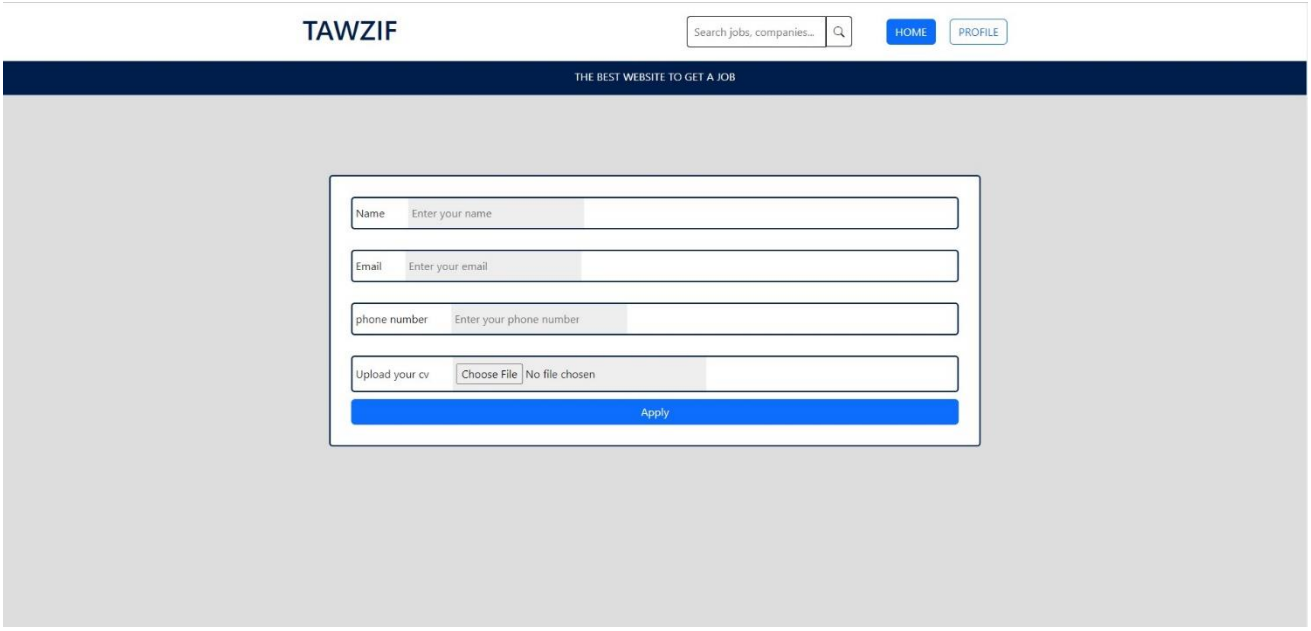
That make him

1-upload a cv

2- write his name

3-write his email

4- write his phone number



This is all of the project and the internship content

I hope this project gain your interest

Thank you for your time