Employee Portal Website – Final

Jacob Colburn

University of Arizona Global Campus

CST310-Software Development

Instructor: Dr. Joseph Issa

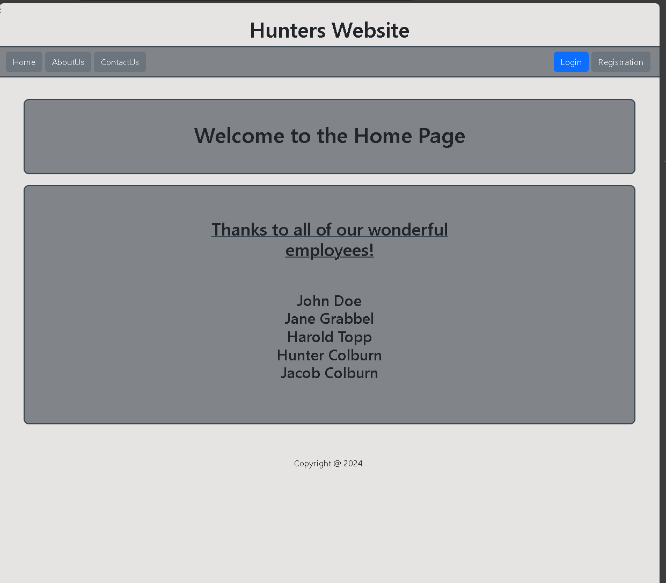
8APR24

**Video and folio links**

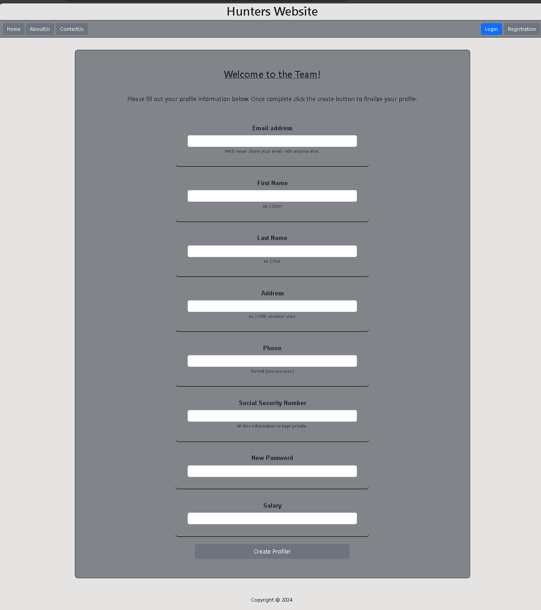
**Folio:** [**https://portfolium.com/jacobcolburn/portfolio**](https://portfolium.com/jacobcolburn/portfolio)

**Video Presentation: https://go.screenpal.com/watch/cZfQqFVswzL**

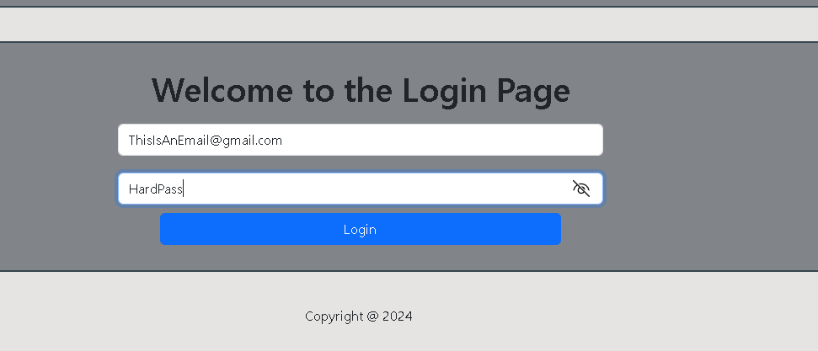
**Screenshots**

Home page (not logged in)

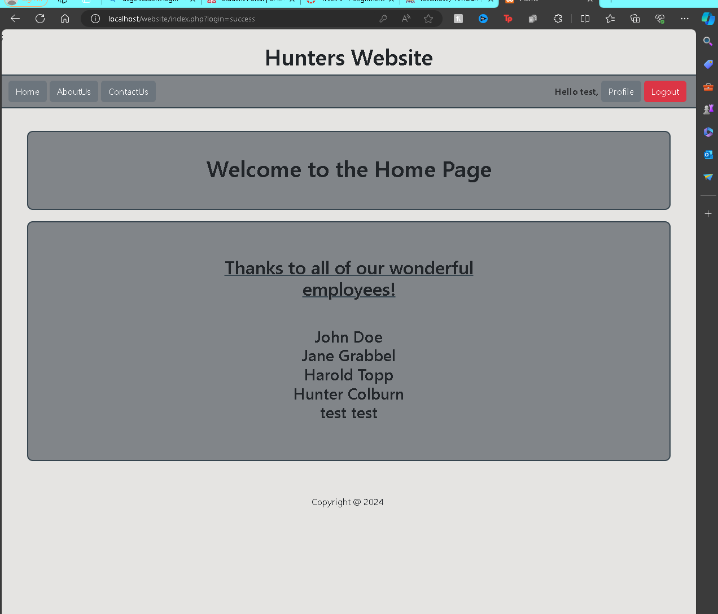
The home page allows users to view a welcome message and see all of the current employees enrolled in the system. There is a navbar implemented with buttons for containing links to the home, about us, and contact pages. As well as two buttons for login and registration for the site.

Registration Page

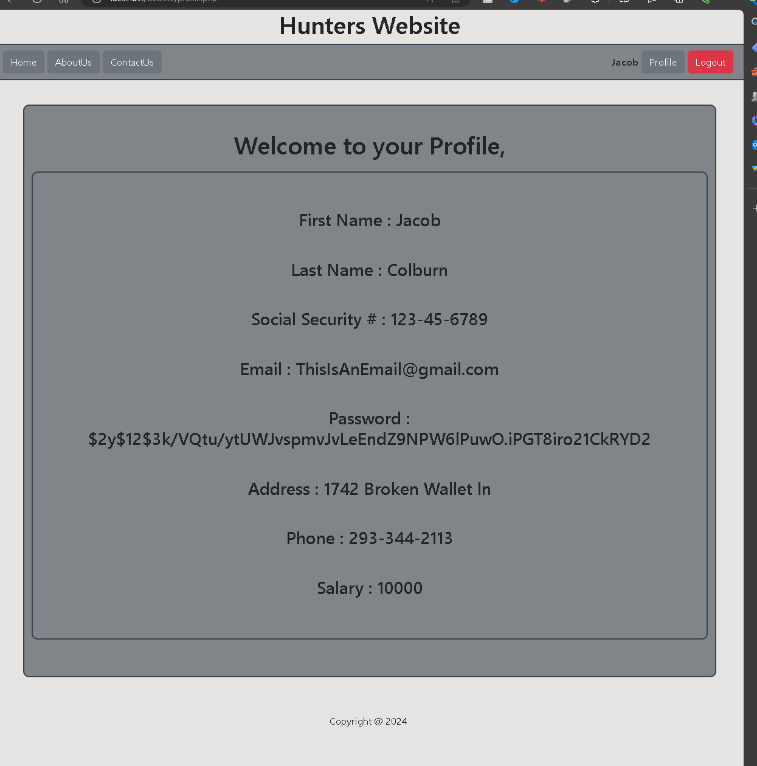
The registration page takes inputs from a user including their name, email address, phone, password, address, and salary. After user inputs all of their information, they will press the submit button and a query will be ran against the database injecting the new user into the database.

Login Page

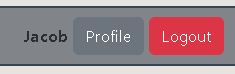
The login page tests the input user credentials against the database. Once verified it enables the user to login and redirects them to the logged in homepage.

Home Page(logged in)

This home page is much like the original home page except the login and registration buttons have been changed to profile and logout buttons. They allow you to of course, logout and view your user profile.

Profile Page

This page runs a database query to select the information where the user id matched the id of the user in the database. The query then pulls all of the necessary information and displays it onscreen for the user to see. I will eventually add an update information button.

Logout button(homepage (not logged in))

The logout button redirects the user to the home page(not logged in) and clears any userid session cookies to ensure that all permissions and credentials are wiped.

Explanation of Implementation

Starting out I created a basic master, index, footer, registration, and login pages containing only the basic html DOCTYPE information.

Next I worked on creating my database and implemented one employees table that contains all of the employees information including the email and password to login to the system later. Once the database was created I created a db connection file that I could require inside my files that needed a connection to the db. Once that was created I added a query to the db on the homepage that would output the current users first name and last name on the home page for the users to see.

Next the registration page was created. I used a form to implement this. I added all of the input fields, and set the forms method to post which sends the database information instead of get which pulls information. It wasn’t requested, but I also implemented an auto-login function into the registration page as well so when you register you will be automatically redirected to the logged in homepage.

Next came the login page. For the login page I realized I would need a bit more security so I added password hashing to hash the password in the database to keep the account more secure. I am new to hashing so I could not figure out how to display it unhashed, so the profile shows the hashed password. Once the passwords were hashed, I created a verify login php file that would run the input email and password against the db it check if the credentials are valid, if not I have errors setups for messaging such as invalid information. If the credentials are valid, a new session and session id is created with the users id number attached, this lets us identify what user is logged in, and then the user is redirected to the logged in homepage.

Next the profile page was created by running a query to the database selecting \* from employee where id =?. This allows us to see information based on the user id attached to the session id to keep other users from viewing private information. Once the user is logged in and can view their profile there is also a logout button that clears and destroys all the session information and redirects back to the not logged in homepage.

Purpose of the Employee Portal

The purpose of the employee portal is to give employees a secure website to login to where they can view their hire information, pay information, put in leave, and see what’s going on with the company and users at the company.It needs to be secure so data will be safe and it must be user friendly. Not all employees will be tech savvy. It can also facilitate communication and collaboration between the employees by giving them an easy way to access all of the functions available to them on the site.

Scripts

Master.php

<?php

//error reports

error\_reporting(E\_ALL ^ E\_NOTICE);

session\_start();

require\_once 'includes/login\_view.php';

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="reset.css">

    <link rel="stylesheet" href="styles.css">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js" integrity="sha384-I7E8VVD/ismYTF4hNIPjVp/Zjvgyol6VFvRkX/vR+Vc4jQkC+hVqc2pM8ODewa9r" crossorigin="anonymous"></script>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.min.js" integrity="sha384-0pUGZvbkm6XF6gxjEnlmuGrJXVbNuzT9qBBavbLwCsOGabYfZo0T0to5eqruptLy" crossorigin="anonymous"></script>

    <title>Employee Portal</title>

</head>

<body>

  <div class="jumbotron">

      <div class="containerMain">

          <h1>Hunters Website<h1>

      </div>

  </div>

  <nav class="navbar navbar-expand-lg">

    <div class="container-fluid">

        <a class="btn btn-secondary btn-padding" href="index.php">Home</a>

        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="navbar-collapse" id="navbarNav">

            <ul class="navbar-nav">

                <li class="nav-item">

                    <a class="btn btn-secondary btn-padding" aria-current="page" href="#">AboutUs</a>

                </li>

                <li class="nav-item">

                    <a class="btn btn-secondary btn-padding" href="#">ContactUs</a>

                </li>

            </ul>

        </div>

        <div class="navbar-text text-center"> <!-- Added text-center class here -->

        <?php echo isset($\_SESSION["user\_firstName"]) ? $\_SESSION["user\_firstName"] : ''; ?>

        </div>

        <ul class="navbar-nav ml-auto">

            <?php

            session\_start();

            if (isset($\_SESSION['userId'])) {

                echo '<li class="nav-item"><a class="btn btn-secondary btn-padding" href="profile.php">Profile</a></li>';

                echo '<li class="nav-item"><a class="btn btn-danger btn-padding" href="logout.php">Logout</a></li>';

            } else {

                echo '<li class="nav-item"><a class="btn btn-primary btn-padding" href="login.php">Login</a></li>';

                echo '<li class="nav-item"><a class="btn btn-secondary btn-padding" href="reg.php">Registration</a></li>';

            }

            ?>

        </ul>

    </div>

  </nav>

  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>

</body>

</html>

Index.php

<?php

    error\_reporting(E\_ALL ^ E\_NOTICE);

class dbConnect{

    private $servername;

    private $user;

    private $pass;

    private $dbname;

    public function connect(){

        $this->servername = "localhost";

        $this->user = "root";

        $this->pass = "";

        $this->dbname = "employee\_db";

        try {

            $dsn = "mysql:host=".$this->servername. ";dbname=".$this->dbname;

            $pdo = new PDO($dsn, $this->user, $this->pass);

            $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_WARNING);

            return $pdo;

        } catch (PDOException $e) {

            echo "Connection error: ".$e->getMessage();

        }

    }

    public function viewEmployees(){

        $sql = "SELECT \* FROM employees";

        $stmt = $this->connect()->prepare($sql);

        $stmt->execute();

        $employees = $stmt->fetchAll();

        foreach ($employees as $employee){

            echo $employee["firstName"]." ".$employee["lastName"].'<br>';

        }

    }

    //public function registerEmployee($email, $password, $firstName, $lastName, $SSN, $address, $phone, $salary){

        //$sql = "INSERT INTO employees(email, password, firstName, lastName, SSN, address, phone, salary) VALUES (?,?,?,?,?,?,?,?)";

        //$stmt = $this->connect()->prepare($sql);

        //$stmt->execute([$email, $password, $firstName, $lastName, $SSN, $address, $phone, $salary]);

    //}

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

    <title>Home</title>

</head>

<body>

    ;

    <?php

    require 'master.php';

    ?>

    <div class="container">

        <h1>Welcome to the Home Page</h1>

    </div>

    <div class="container" type="db">

    <h2>Thanks to all of our wonderful employees!</h2>

    <h3>

    <?php

        $object = new dbConnect;

        //$object -> registerEmployee("tommyboy1@gmail.com","password5","Tommy", "Boothe", "005", "157 lazy dr", "667-543-8441", "49000.00" );

        $object -> viewEmployees();

    ?>

    </h3>

    </div>

    <?php require\_once 'footer.php'; ?>

</body>

</html>

Profile.php

<?php

    error\_reporting(E\_ALL ^ E\_NOTICE);

    include\_once 'includes/viewProfile.php';

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" type="css"href="style.css">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

    <title>Profile</title>

</head>

<body>

    <?php require 'master.php'; ?>

        <div class="container">

                                        <!-- welcome user to their profile and uppercase thier name -->

            <h1>Welcome to your Profile, <?php echo $firstName; ?>

            <div class="container">

               <h3>First Name : <?php echo $firstName; ?> </h3>

               <h3>Last Name : <?php echo $lastName; ?></h3>

               <h3>Social Security # : <?php echo $ssn; ?></h3>

               <h3>Email : <?php echo $email; ?></h3>

               <h3>Password : <?php echo $password; ?></h3>

               <h3>Address : <?php echo $address; ?></h3>

               <h3>Phone : <?php echo $phone; ?></h3>

               <h3>Salary : <?php echo $salary; ?></h3>

            </div>

        </div>

    <?php require\_once 'footer.php'; ?>

</body>

</html>

Login.php

<?php

    error\_reporting(E\_ALL ^ E\_NOTICE);

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

    <title>Employee Login</title>

</head>

<body>

<?php

 require 'master.php';

 check\_login\_errors();

 ?>

<form action="includes/verifyLogin.php" method="post">

<div class="smcontainer">

    <h1 style="text-align: center;">Welcome to the Login Page</h1>

    <div class="login">

        <input

            type="text"

            name="email"

            class="form-control"

            placeholder="Enter your email address"

            required

        />

    </div>

    <div class="login">

        <input

            type="password"

            name="password"

            class="form-control"

            placeholder="Enter Your Password"

            required

        />

    </div>

    <button type="submit" class="btn btn-primary">Login</button>

</div>

</form>

<?php require\_once 'footer.php'; ?>

</body>

</html>

Logout.php

<?php

session\_start();

session\_unset();

session\_destroy();

header("Location: index.php?logout=successfull");

    die();

config\_session.php

<?php

ini\_set('session.use\_only\_cookies', 1);

ini\_set('session.use\_strict\_mode', 1);

session\_set\_cookie\_params([

    'lifetime'=> 1800,

    'domain'=> 'localhost',

    'path'=>'/',

    'secure'=> true,

    'httpOnly'=> true

]);

session\_start();

if(isset($\_SESSION["userId"])){// update session id everu 30 minutes but keeps users id and logged in

    if(!isset($\_SESSION['last\_regeneration'])) {

        regenerate\_session\_id\_loggedin();

    }else{

        $interval = 60 \* 30;

        if(time() - $\_SESSION['last\_regeneration'] >= $interval){

           regenerate\_session\_id\_loggedin();

        }

    }

} else{//updates session id every 30 minutes if user is not logged in

    if(!isset($\_SESSION['last\_regeneration'])) {

        regenerate\_session\_id();

    }else{

        $interval = 60 \* 30;

        if(time() - $\_SESSION['last\_regeneration'] >= $interval){

           regenerate\_session\_id();

        }

    }

}

function regenerate\_session\_id() {

    session\_regenerate\_id();

    $\_SESSION['last\_regeneration'] = time();

}

function regenerate\_session\_id\_loggedin() {

    session\_regenerate\_id(true);

    $userId = $\_SESSION['userId'];

    $newSessionId = session\_create\_id();

    $sessionId = $newSessionId. "\_" . $userId;

    session\_id($sessionId);

    $\_SESSION['last\_regeneration'] = time();

}

Db initialization

<?php

$dsn = "mysql:host=localhost;dbname=employee\_db";

$dbusername="root";

$dbpassword="";

try {

    $pdo = new PDO($dsn, $dbusername, $dbpassword);

    $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_WARNING);

} catch (PDOException $e) {

    echo "Connection error: ".$e->getMessage();

}

Formhandler.php

<?php

session\_start();

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $email= $\_POST["email"];

    $firstName= $\_POST["firstName"];

    $lastName= $\_POST["lastName"];

    $address= $\_POST["address"];

    $phone= $\_POST["phone"];

    $ssn= $\_POST["ssn"];

    $password= $\_POST["password"];

    $salary= $\_POST["salary"];

    try {

        require 'db.ini.php';

        $query = "INSERT INTO employees (email, firstName, lastName, address, phone, ssn, password, salary) VALUES(?,?,?,?,?,?,?,?);";

        $stmt = $pdo->prepare($query);

        $options = [

            'cost' => 12

        ];

        $hashedpassword = password\_hash($password, PASSWORD\_BCRYPT, $options);

        $stmt->execute([$email, $firstName, $lastName, $address, $phone, $ssn, $hashedpassword, $salary]);

        $userId = $pdo->lastInsertId();

        $\_SESSION['userId'] = $userId;

        $\_SESSION['email'] = $email;

        $pdo= null;

        $stmt= null;

        header("Location: ../index.php");

        die();

    } catch (PDOException $e) {

        die("Query Failed: " . $e->getMessage());

    }

}

else{

    header("Location: ../index.php?RegistrationFailed!");

}

Hashpassword.php

<?php

//hashing password for security purposes

$passwordSignup ="";

$options = [

    'cost' => 12

];

$hashedpassword = password\_hash($passwordSignup, PASSWORD\_BCRYPT, $options);

$password = "";

if (password\_verify($password, $hashedpassword)){

    echo "Password is correct";

}

else{

    echo "Password is not correct";

}

Login\_controller.php

<?php

declare(strict\_types= 1);

function is\_input\_empty(string $email, string $password){

    if (empty($email) || empty($password)){

         return true;

     }else{

         return false;

     }

 }

function is\_email\_wrong(array|bool $result){

    if(!$result){

        return true;

    }else{

        return false;

    }

}

function is\_password\_wrong(string $password, string $hashedPassword){

    if(!password\_verify($password, $hashedPassword)){

        return true;

    }else{

        return false;

    }

}

Login\_view.php

<?php

declare(strict\_types=1);

function output\_username(){

    if(isset($\_SESSION["userId"])){

        echo "Hello " . $\_SESSION["user\_firstName"] . ",  ";

    }else{

        echo "Please Login, ";

    }

}

function check\_login\_errors(){

    if (isset($\_SESSION["errors\_login"])){

        $errors = $\_SESSION["errors\_login"];

        echo '<br/>';

        foreach ($errors as $error ) {

           echo 'error ', $error;

        }

        unset($\_SESSION["errors\_login"]);

    }

    else if (isset($\_GET['login']) && $\_GET['login'] === "success"){

        echo '<br/>';

        echo 'login success';

    }

}

Login\_model.php

<?php

declare(strict\_types= 1);

function get\_user(object $pdo, string $email){

    $query ="SELECT \* FROM employees WHERE email = :email;";

    $stmt = $pdo->prepare($query);

    $stmt->bindParam(":email", $email);

    $stmt->execute();

    $result = $stmt->fetch(PDO::FETCH\_ASSOC);

    return $result;

}

function get\_pass(object $pdo, string $hashedpassword){

    $query ="SELECT \* FROM employees WHERE password = :password;";

    $stmt = $pdo->prepare($query);

    $stmt->bindParam(":password", $hashedpassword);

    $stmt->execute();

    $result = $stmt->fetch(PDO::FETCH\_ASSOC);

    return $hashedpassword;

}

Verifylogin.php

<?php

session\_start();

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $email= $\_POST["email"];

    $password= $\_POST["password"];

    try {

        require\_once 'db.ini.php';

        require\_once 'login\_model.php';

        require\_once 'login\_controller.php';

        $errors = [];

        if (is\_input\_empty($email, $password)) {

            $errors["empty\_input"] = 'Cannot Sumbit Empty Fields!';

        }

        $result = get\_user($pdo, $email);

        if (is\_email\_wrong($result)) {

            $errors["login\_incorrect"] = 'incorrect login information!';

        }

        if (!is\_email\_wrong($result) && is\_password\_wrong($password, $result["password"])) {

            $errors['password\_incorrect'] = 'incorrect login information!';

        }

        require\_once 'config\_session.php';

        if($errors){

            $\_SESSION["errors\_login"] = $errors;

            header("Location: ../login.php");

            die();

        }

        $newSerssionId = session\_create\_id();

        $sessionId = $newSerssionId. '\_' . $result["id"];

        session\_id($sessionId);

        $\_SESSION["userId"] = $result['id'];

        $\_SESSION["user\_firstName"] =htmlspecialchars($result['firstName']);

        $\_SESSION['last\_regeneration'] = time();

        header("Location: ../index.php?login=success");

    } catch (PDOException $e) {

        die("Query Failed: " . $e->getMessage());

    }

} else {

    header("Location: ../index.php");

    die();

}

viewPrfile.php

<?php

error\_reporting(E\_ALL);

// databse connection file

require\_once 'db.ini.php';

//start a session

session\_start();

// Check if the user is logged in

if(isset($\_SESSION["userId"])) {

    // Fetch user information from the database

    $userId = $\_SESSION["userId"];

    // fetch the user information from the database using PDO statement to query the database.

    $stmt = $pdo->prepare("SELECT \* FROM employees WHERE id = ?");

    $stmt->execute([$userId]);

    $user = $stmt->fetch(PDO::FETCH\_ASSOC);

    //set the user variables from the database

    if($user) {

        $email = $user["email"];

        $password = $user["password"];

        $firstName = $user['firstName'];

        $lastName = $user["lastName"];

        $address = $user["address"];

        $phone = $user["phone"];

        $salary = $user["salary"];

        $ssn = $user["ssn"];

    }

} else {

    // Redirect to login page if the user is not logged in

    header("Location: login.php");

    exit;

}

References

Connolly, R., & Hoar, R. (2018). Fundamentals of web development (2nd. ed.). Pearson. Retrieved from <https://ashford.redshelf.com/app/ecom/shelf/course-section/7022331>

Krossing, D., (2023), <https://www.youtube.com/@Dani_Krossing>

McGregor, M., 2020 “4 Design Patterns You Should Know for Web Development: Observer, Singleton, Strategy, and Decorator”