WorkPlace Coaction System

A PROJECT REPORT
Submitted By

NIRANJAN DARSHAN
University Roll No 1900290140021

ABHISHEK KUMAR YADAV
University Roll No 1900290140002

Submitted in partial fulfillment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATIONS

Under the Supervision of
DR. Arun Kumar Tripathi
PROFESSOR

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206
(NOVEMBER 2021)

CERTIFICATE

Certified that **Niranjan Darshan (Enrollment No. 1900290140021), Abhishek Kumar Yadav (Enrollment No. 190029014005145)** have carried out the project work having " **WorkPlace Coaction System**" for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Niranjan Darshan
University Roll No. 1900290140021
Abhishek kumar Yadav
University Roll No. 1900290140002

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Dr. Arun Kumar Tripathi
Professor
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Signature of Internal Examiner **Signature of External Examiner**

Dr. Ajay Shrivastava
Head, Department of Computer Applications
KIET Group of Institutions, Ghaziabad

ABSTRACT

WorkPlace Coaction system is a Web Application which help in the smooth working of an organization. It is use to store data about the Employee and the to give the organizational email for secure working in the organization.

It will help to state which team is working under perticular project. Under the supervision of whom the team is performing the task, Each and every person will able to login with there allocated email id and password and know on which project they are working with whom.

Each and every member will able to see the profile of the other member correctly and efficiently.It will provide the security, and transperancy in the organizational work. Every member will know other member also due to which friendly and competitive relation will be seen in the organization which will result in productivity in an organization.

ACKNOWLEDGEMENTS

Success in life is never attained single handedly. My deepest gratitude goes to my thesis supervisor, **Dr. Arun Kumar Tripathi** for his guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Ajay Kumar Shrivastava, Professor and Head, Department of Computer Applications**, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Niranjan Darshan

Abhishek Kumar yadav

Table of contents

Declaration 1

certificate 2

Abstract 3

Acknowledge 4

Table Of Contents 5-7

Chapter-1 Introduction 8

1.1 Project Description 8

Chapter-2 Project category 9

2.1 Technology 9

2.2 Language Used 10

chapter 3 Software Requirement Specification 11

3.1 General description 11

3.1.1 Problem statement 11

3.2 System Objectives 11

3.2.1 Improvement In Control And Performance 11

3.2.2 Requirement Specification 12

3.3Functional Requirements 12

3.4 Non Functional Requirements 12

3.5 Software And Hardware Requirements 13

3.5.1 Software Requirements 13

3.5.2 hardware requirements 14

3.6 Existing Vs Proposed System 14

3.7 Software System Attributes 14

3.8 Feature Of Workplace Coaction System 15

3.9 Preliminary Investigation 16

3.10 Model Used Incremental Model 17

3.1.1 Preliminary Description 17

Chapter 4

4.1 Feasibility Study 18

4.1.1 Economical Feasibility 18

4.1.2 Technical Feasibility 19

4.1.3 Operational Feasibility 19

Chapter 5 Module 20

5.1 Planning And Scheduling 21

5.11 Gain Chart 21

5.2 Data Flow Diagram 22

5.3 Entity Relationship Diagram 24

Chapter 6 Sdlc 28

Chapter 7 Coding 29-93

Chapter 8 Testing 94-97

Chapter 9 Conclusion 98

Chapter 10 Bibliography 99

CHAPTER 1
INTRODUCTION

1.1 PROJECT DESCRIPTION

This project has been developed to override the problems prevailing in the practicing manual system. This application is supported to eliminate and in some cases reduce the hardships faced by the existing system. Moreover, this application is designed for moving the office toward the secure digitalization.It will enchance the transperancy in the work.

No formal knowledge is needed for the user to use this system. Thus, by this all it proves it is user friendly. So, This "WorkPlace Coaction System" can lead to error-free, secure, reliable and fast system.

This application provides the way to manage the team, office and the project for productive outcome. Through this applications we are approaching towards the less use of papers by virtue of which we are indirectly sustaining the environment.

Chapter 2
PROJECT CATEGORY

2.1 TECHNOLOGIES USED

XAMPP Server

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

Notepad++

Notepad++ is a text and source code editor for use with Microsoft Windows. It supports tabbed editing, which allows working with multiple open files in a single window. The product's name comes from the C increment operator. Notepad++ is distributed as free software

TECHNOLOGIES USED

APPLICATION	: Xampp Server, Notepad++
DESIGNING	: HTML, CSS, Javascript
Backend	: PHP

2.2 Language Used (Designing and Developing)

This project has been developed HTML and Java.

- **•HTML:-**
    - HTML stands for Hyper Text Markup Language
    - HTML is the standard markup language for creating Web pages
    - HTML describes the structure of a Web page
    - HTML consists of a series of elements
    - HTML elements tell the browser how to display the content
    - HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

CSS:

- 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

JavaScript:

JavaScript, often abbreviated as JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled and multi-paradigm. It has dynamic typing, prototype-based object-orientation and first-class functions.

PHP:

PHP is an acronym for "PHP: Hypertext Preprocessor"

- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

Chapter 3

SOFTWARE REQUIREMENT SPECIFICATION

3.1 GENERAL DESCRIPTION

This combined aggregation of information and workplace activity constructs a general, specific program or aim which is to be executed or produced within the workplace while working with others as a squad. The history of coaction began many centuries ago, long before the B.C. or A.D. epochs, where at least two persons had to pass on in the attempt of finishing a undertaking, undertaking, or written papers. Therefore, coaction is non a new term, but an enhanced and improved one in the professional workplace.

3.1.1 PROBLEM STATEMENT:

The problem occurred before having computerized system includes:

- Seeking for the help to play this traditional mode.
- Excessive use of Paper for maintaining register and updating data.
- More chance of Unfairness while giving marks due to biasness.

3.2 SYSTEM OBJECTIVES

3.2.1 Improvement in control and performance

The system is developed to cope up with the current issues and problems of forgetting the traditional mechanism. The system identify who is accessing the profile and the data/information will be updated on the portal. To declare the Project and performance of the employee and details.

Save cost

The existing system is based on the pen paper mode and several in the digital mode but is not securedand efficient to work.

Save Time

People at any location will able to perform or know there seniors subordinate team and there uniqueness etc. by registering or Loging in the Portal.

3.2.2 Requirement Specification:

The application requirement specification is produced at the analysis task. The function and performance allocated to application as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints.

3.3 Functional Requirements:

Admin Generated Email and Password

The application will work with Email and password generated by the admin after joining the firm.

Internet Connectivity:

As discussed that Application will work on Online mode so it need regular Internet Connectivity.

Register and Login:

To Work on the web application one should be registered and should have to login with the organizational email and password.

3.4 Non-functional Requirements:

Performance Requirements

- User friendly**:** The system should be user friendly so that it can easily be understand by the user without any difficulty.
- Ease of maintenance :- System should be easy to maintain and use.
- Less time consuming: The system should be less time consuming which could be achieved by good programming.
- Error free: The system should easily handle the user error in any case.
- Static: Application runs on stand alone machine . Support only single user.

3.5 SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system .

3.5.1 SOFTWARE REQUIREMENTS

· **Operating system**- Windows/Linux Operating System This is the web Application which can run on any of the Operating System.

Database: PHPAdmin is used in storing the data in structured manner.

XAMPP is a Software used for server which is use to serve the client what he/she wants from the server.

Browser: Any of the browser can be used to run and test the web application's Appearance and working eg. Internet Explorer, Google Chrome, Mozilla Firefox etc.

· **Development tools and Programming language**-HTML, CSS, Javascript, PHP is used to write the whole designing and operational code. PHP is used for backend maintainance.

3.5.2 HARDWARE REQUIREMENTS

- Desktop/Laptop any configuration.

3.6 EXISTING VS PROPOSED SYSTEM

Existing system does not have a secure facility of Workplace Coaction System application with transperancy in Workplace whereas proposed system is secure and transperancy in the work of the people.

Existing system does not have any facility of generating Email Online whereas proposed system is working on the facility of generating email and password online by the admin with security .

Existing System does not have the facility of registering and generating organizational password Whereas proposed system are more focused on it.

3.7 . Software System Attributes

- **Portability**:- The system should be machine independent.
- **Security**:- The system is designed in such a way that it will store the recorded data in the system of the owner. The system will be secure from unauthorized access of the application.
- **Maintainability**: The system will be designed in a maintainable order. The system can be easily modified and renewed according to the need of the organization.

3.8 . Feature of Workplace Coaction System

- no internet connection required against the computer
- multiple users can login and register on the same portal remotely.
- People can register and login in the system.
- graphics with a classic look and the feel of a royal Web Application
- classic Profile Details to display profile of each employee
- security of data to be stored
- ensures data accuracy (number of alert generated)
- minimize manpower
- minimize time consumption
- greater efficiency
- fast
- better services
- user friendliness and interactive
- minimum time requried
- easy to update
- user friendly
- free for the user

3.9. Preliminary investigation:

Fact Finding:

After obtaining the background knowledge, we began to collect data on the existing system.

The tools that are used in information gathering are as follows:

- Online Apps observation.
- Review of the peoples.

The model we have used is Incremental Model. In this model, first of all the existing system is observed, then customer requirements are taken in consideration then planning, modelling, construction and finally deployment and again adding the new system if asked by the customer to do so.

3.10 . Model used: Incremental Model

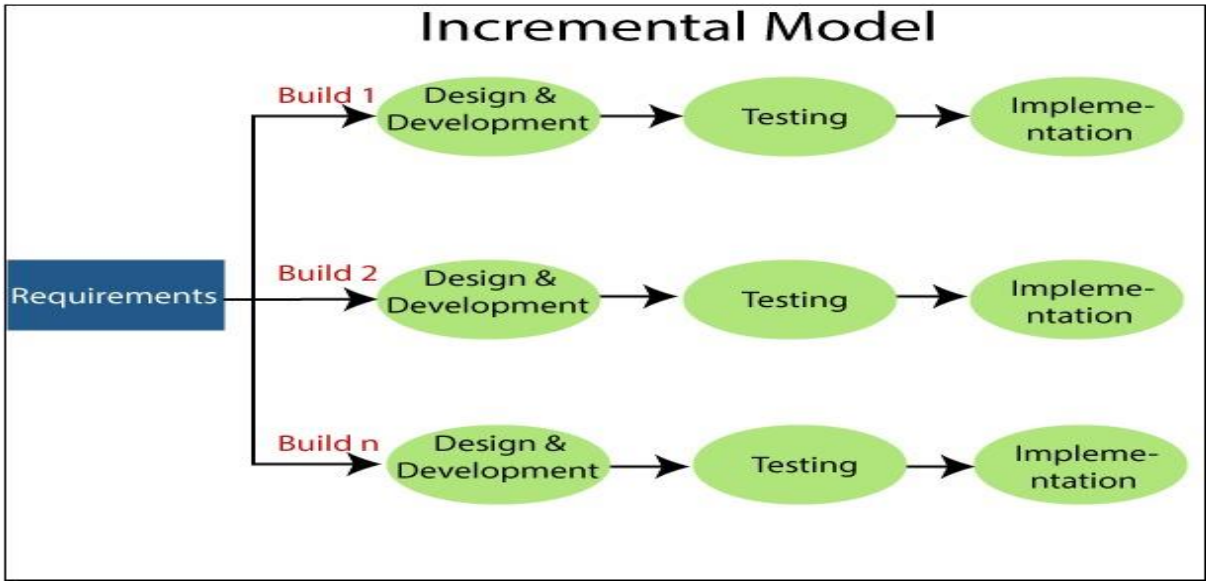


Fig 1.0: Incremental Model

Incremental Model is a software development process where requirements are divided into several stand-alone software development modules. In this project the first increment is often a core product where the basic requirements are addressed, and supplementary features are added in the next increments.

3.10 Preliminary Description:

The first step in the system development life cycle is the preliminary investigation to determine the feasibility of the system. The purpose of preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the system in all respect. Rather, it is the collecting of information that helps committee members to evaluate the merits of project request and make an informed judgement about the feasibility of the proposed project.

Analyst working on the preliminary investigation should accomplish the following objectives:

- Clarify and understand the project request.
- Determine the size of the project.
- Access costs and benefits of alternative approaches.
- Determine the technical and operational feasibility of alternative approaches.
- Report the findings to management with recommendations outlining the acceptance and rejection of the proposal

Chapter 4

4.1 Feasibility study:

After studying and analyzing all the existing and requires functionalities of the system, the next task is to do the feasibility study for the project. Feasibility study includes consideration of all the possible ways to provide a solution to a given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

- ### 4.1.1 Economical Feasibility:

  For the economic feasibility, Economic analysis or cost/benefits analysis is most frequently used technique the effectiveness of a proposed system. it is a procedure to determine the benefits and saving those are expected from the proposes system and compare them with cost .if the benefits outweigh the costs, a decision is taken to design and implement the system. otherwise, further justification or alternative in proposed system will have to be made if it is to have a chance of being approved this is ongoing effort that improves in accuracy at each phase of a system life cycle

- ### 4.1.2 Technical feasibility:

  This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionalities to be provided in the system, as described in the System Requirement Specification (SRS), and checked if everything was possible using different type of front end and back end platform.

4.1.3 Operational Feasibility:

No doubt the technically growing world needs more enhancement in technology, this apps is very user friendly and all inputs to be taken all self-explanatory even to a layman. As far our study is concerned, the clients will be comfortable and happy as the system has cut down their loads and bring the young generation to the same virtual world they are growing drastically.

Operational feasibility cover two aspects.one technical performance

aspects and the other is acceptance within the organization.

Operation feasibility determine how the proposed the system will fit in with the current operation and what needs to implement the system

Chapter 5

5.1 Planning and scheduling

5.1.1 Gantt chart

A Gantt chart can be developed for the entire project or a separate chart can be developed for each function. A tabular form is maintained where rows indicate the task with milestones and columns indicate duration (Days).

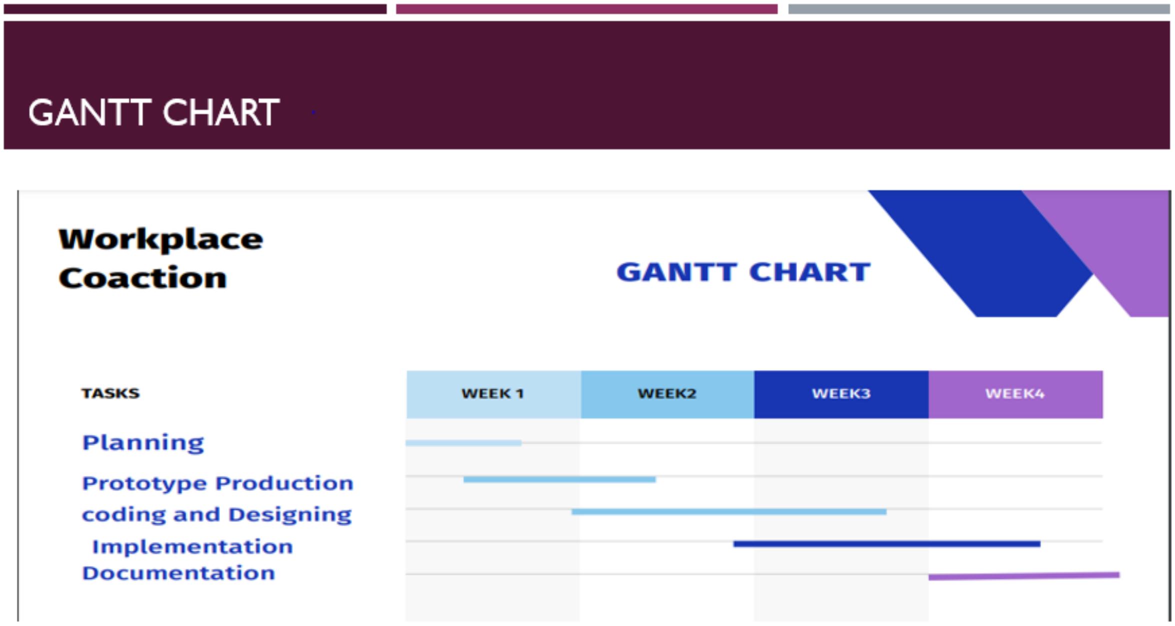


Fig 2.0 Gantt Chart

5.1.2 Software Requirements with specifications:

Name of Components	Specifications
Operating system	Windows
Language	HTML,CSS, Javascript, PHP
Software Development kit	XAMPP, Google Chrome
Markup Language Enable	HTML

5.1.3 Hardware Requirements with specifications:

Name of Components	Specifications
Desktop/Laptop	Any Configuration
Memory Used	6.31 MB

5.2 : DATA FLOW DIAGRAM

Are used to graphically represent the flow of data in a Workplace Coaction System. DFD describes the processes that are involved in a system to transfer data from the admin to the employee, employee to the employee, employee to admin etc.

WorkPlace Coaction System this system shows the flow of data in admin Modules on many Action. It shows the flow of data among the sub module in it Admin data flow on the sub screen.

DFD is the abbreviation for **Data Flow Diagram**. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. Data Flow Diagram can be represented in several ways. The DFD belongs to structured-analysis modeling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes.

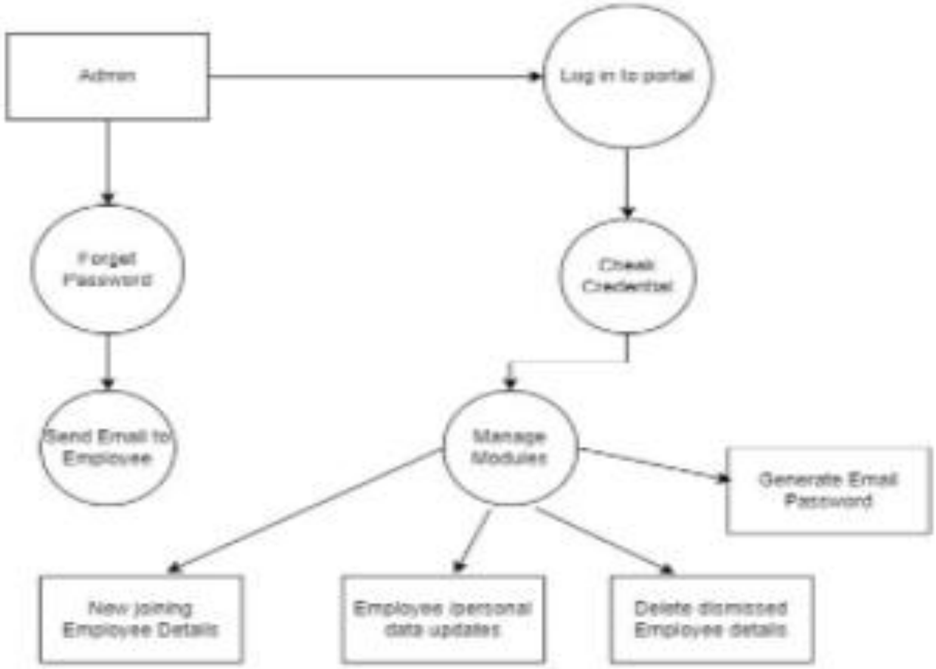


Fig 3.1 Admin Mode DFD

WorkPlace Coaction System this system shows the flow of data in Employee Modules on many Action. It shows the flow of data among the sub module in it Employee data flow on the sub screen.

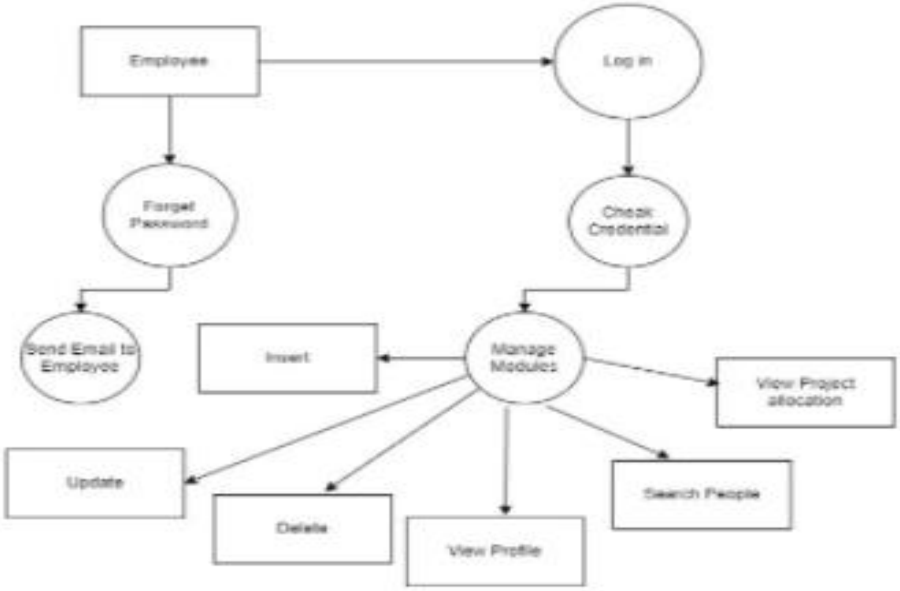


Fig 3.2 Employee DFD

5.3 ENTITY RELATIONSHIP DIAGRAM:

This ER Diagram represents the model of WorkPlace CoAction System Entity. The Entity Relationship Diagram show all visual instrument of Database table and relation between HomePage, Admin Page, Employee Page. All of it have Structured data and every entity may have some attributes.

WorkPlace Coaction System Entity and their Attributes:

1. Admin: Attribute ofAdmin: Email id, Password, Forget Password.
2. Insert New Employee Details: Attributes are: Name, Email, Password, Gender, Qualification, Project, Project Manager, Phone.
3. Delete New Employee: Attributes are: Name, Email, Password, Gender, Qualification, Project, Project Manager, Phone.
4. Update Details of Self: Attributes are: Name, Email, Password, Gender, Qualification, Project, Project Manager, Phone.
5. Employee: Attribute of Employee: Email id, Password, Forget Password.

6. Employee Update Details: Attributes are: Name, Email, Gender, Qualification, Project, Project Manager, Phone.
7. Search Employee: Attributes are: Name, Email, Gender, Qualification, Project, Project Manager, Phone.
8. View Profile of Employee: Attributes are: Name, Email, Gender, Qualification, Project, Project Manager, Phone.

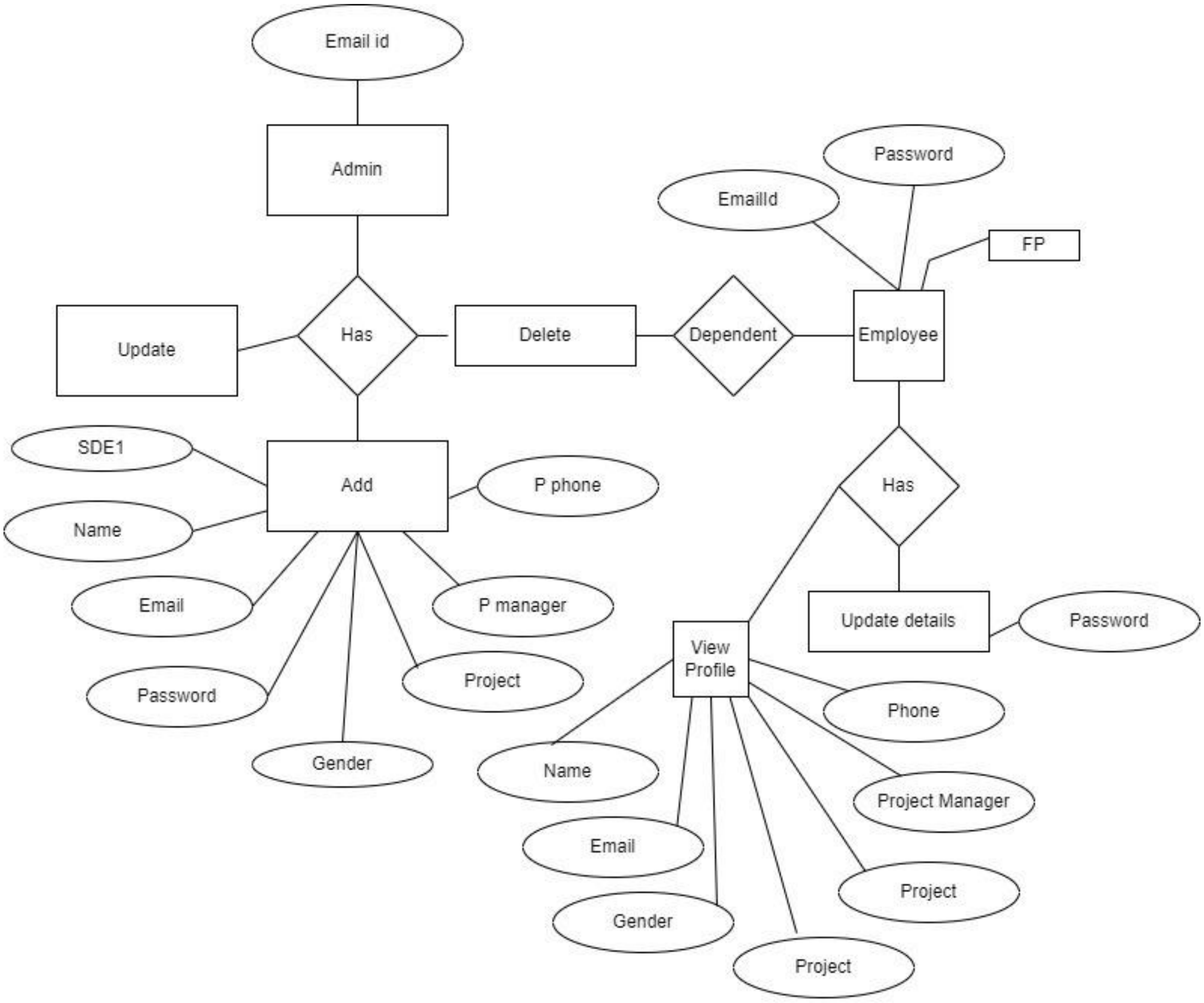


Fig 4.1 Entity relationship Diagram

CHAPTER 5

Modules

5(i) EMPLOYEE MODULE: Under emplpoyee module the employee details will be seen. This module manages activities related to employee' CEO details, appointment of Project and view profile of collegue.

Following points are given below which is related to patients:

a) Search collegue
b) Login Modules
c) See CEO of Organization
d) Update personal details
e) See details of Project Manager
f) See project assigned to you

5(ii) ADMIN MODULE*:* Under this module admin module are given means this module manages activities to registration of employee(new) and updation of details of employee, deletion of resigned employee details .Following points are given below which is related to admin:

(a) Login Modules

(b) Addition of New Employee

(c) Updation of Employee Details

(d) Deletion of Employee Details

CHAPTER 6

SDLC(Incremental Model)

SDLC (Incremental model):

In incremental model the whole requirement is divided into various builds. Multiple development cycles take place here, making the life cycle a **“multi-waterfall” cycle**. Cycles are divided up into smaller, more easily managed modules. Incremental model is a type of software development model like **V-model**, **Agile model** etc.

Then he started building it and in the first iteration the first module of the application or product is totally ready and can be demoed to the customers.

Likewise in the second iteration the other module is ready and integrated with the first module. Similarly, in the third iteration the whole product is ready and integrated. Hence, the product got ready step by step.

Advantages of Incremental Model

- Requirements of the system are clearly understood
- When demand for an early release of a product arises
- When software engineering team are not very well skilled or trained
- When high-risk features and goals are involved
- Such methodology is more in use for web application and product based companies

Disadvantages of Incremental model:

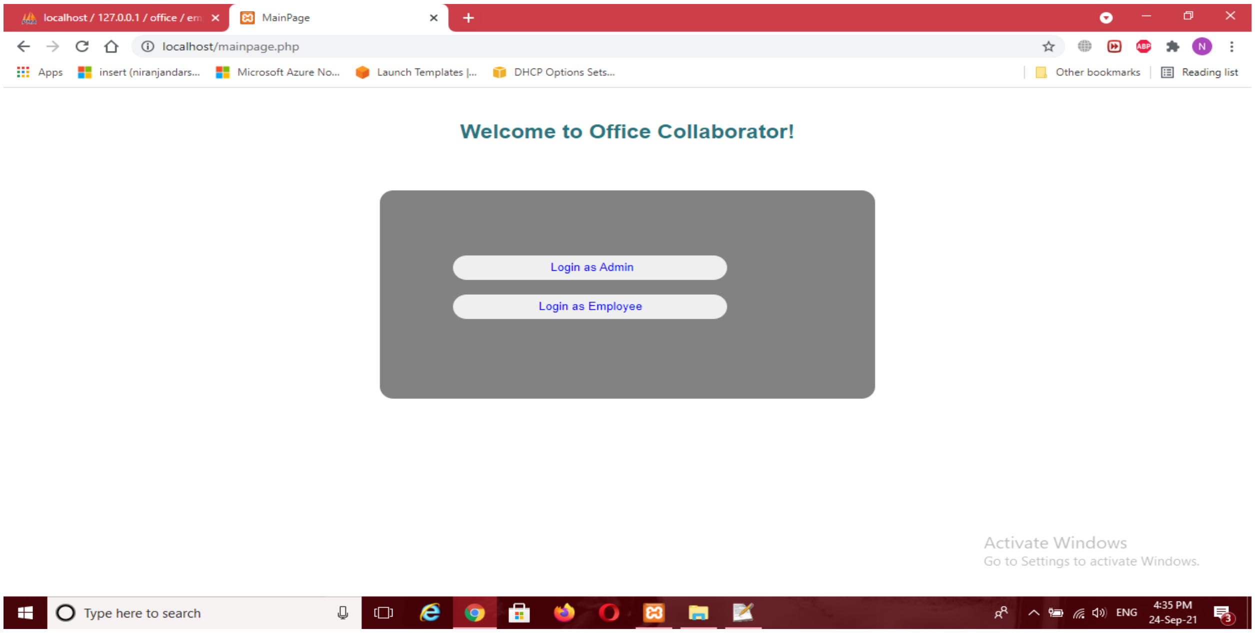
- Needs good planning and design.
- Needs a clear and complete definition of the whole system before it can be broken down and built incrementally.
- Total cost is higher than **waterfall**.

CHAPTER 7

MainPage.php

```
<!DOCTYPE html>
<html>
<head>
  <title>MainPage</title>
  <link rel="stylesheet" type="text/css" href="styleadmin.css">
</head>
<body>
  <h2>WorkPlace Coaction System</h2><br>
  <div class="login">

  <form id="login" method="POST">
      <input type="submit" name="Admin" id="log"  value=" Login as Admin"> <br><br>
                  <input type="submit" name="Employee" id="log" value="Login as Employee">
<br><br>




  </form>
</div>
</body>
</html>
<?php

  $hostname="localhost";
  $username="root";
  $pwd="";
  $dbname="office";
  $conn=new mysqli($hostname,$username,$pwd,$dbname);
  if($conn->connect_error){
  die("Error in connection".$conn->connect_error);

  }
  else{
  if(isset($_POST["Admin"])){
        //include("addemployee.php");
        header("location:admin.php");
  }

  if(isset($_POST["Employee"])){
        //include("updateemployee.php");
        header("location:employeelogin.php");
  }
  }
?>
```

Output:



Welcome to Office Collaborator!

styleprofile.css

```
body
{
   margin: 0;
   padding: 0;
   background: #C38D9E;
   font-family: 'Arial';
}
.login{
      width: 382px;
      overflow: hidden;
      margin: auto;
      margin: 20 0 0 550px;
      padding: 80px;
     border-style:solid;
           background: #FFFFFF;
      border-radius: 15px ;


}
.title{
      width: 200px;
                  height:48px;
      overflow: hidden;
      margin: 20px;
```

```
        margin: 20 0 0 550px;
        text-align:center;
       border-style:solid;
              background: #277582;
        border-radius: 15px ;

}

label{
    color: #228B22;
    font-size: 17px;
}
h1{
            text-align:center;
}

#log{
    width: 80px;
    height: 40px;
    border: none;
    border-radius: 17px;
    padding-left: 7px;
    color: white;
            text-align:center;



}
```

```
a#log:hover, a#log:active {color: yellow;}



a{
   float: right;
   background-color: grey;
}
#readonly{
         background:#DCDCDC;
         border-style:none;
}
#save{
   width: 300px;
   height: 30px;
   border: none;
   border-radius: 17px;
   padding-left: 7px;
   color: blue;

}
input#save:hover, input#save:active {color: red;}
```

Style.css

```
body
{
   margin: 0;
   padding: 0;

   font-family: 'Arial';
}
.login{
      width: 382px;
      overflow: hidden;
      margin: auto;
      margin: 20 0 0 550px;
      padding: 80px;
      background: #828282;
      border-radius: 15px ;

}
.title{



      margin: 20px;


      margin: 20 0 0 550px;
      text-align:center;


                          background: #FFFFFF;
      border-radius: 15px ;
```

```
}
h2{
    text-align: center;
    color: #277582;
    padding: 20px;
}
label{
    color: #000000;
    font-size: 17px;
}
#Uname{
    width: 300px;
    height: 30px;
    border: none;
    border-radius: 3px;
    padding-left: 8px;
}
#Email{
    width: 300px;
    height: 30px;
    border: none;
    border-radius: 3px;
    padding-left: 8px;

}
#Pass{
    width: 300px;
    height: 30px;

    border: none;

    border-radius: 3px;

    padding-left: 8px;


}

#log{

    width: 300px;

    height: 30px;

    border: none;

    border-radius: 17px;

    padding-left: 7px;

    color: blue;




}

input#log:hover, input#log:active {color: red;}


a{

    float: right;

    background-color: transparent;

                              font-color:white;

}

a:hover{

 color: yellow;

}
```

Admin.php

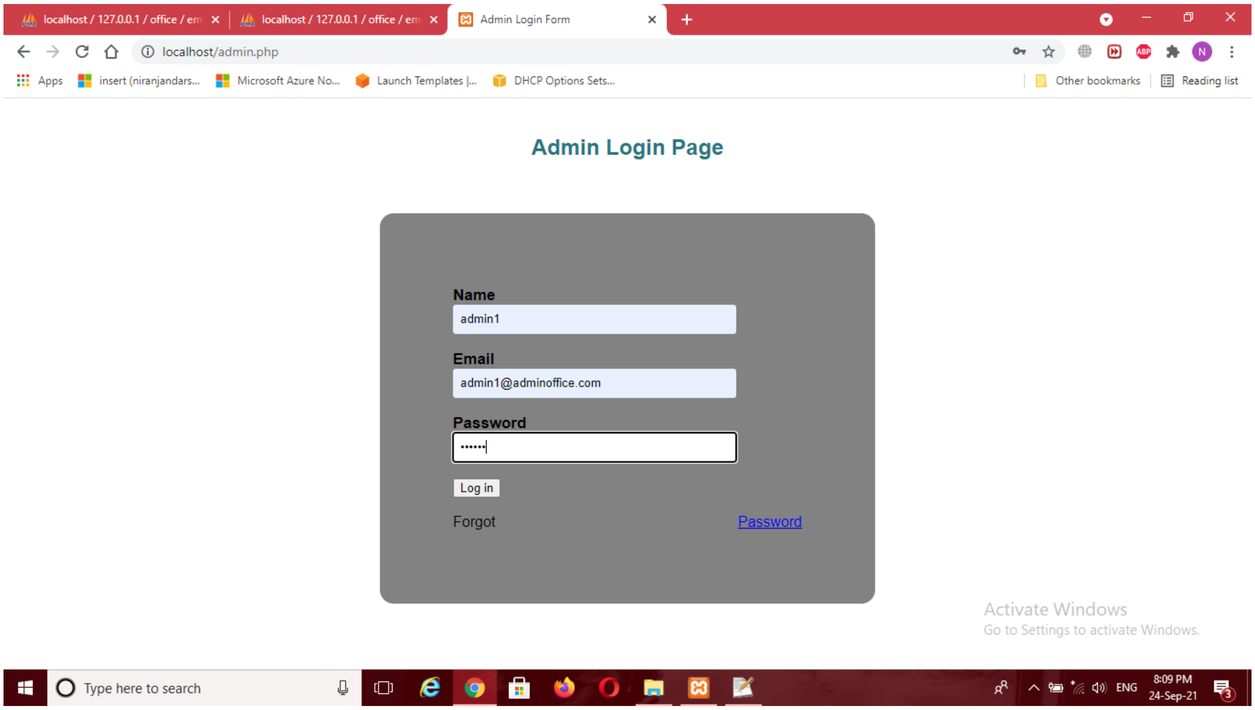
```
<!DOCTYPE html>
<html>
<head>
    <title>Admin Login Form</title>
    <link rel="stylesheet" type="text/css" href="styleadmin.css">
</head>
<body>
    <h2>Admin Login Page</h2><br>
    <div class="login">
    <form id="login" method="POST" >
        <label><b>Name
        </b>
        </label>  <br>
        <input type="text" name="Name" id="Uname" placeholder="Username">
        <br><br>
                                <label><b>Email
        </b>
        </label>    <br>
        <input type="text" name="Email" id="Email" placeholder="email">
        <br><br>
        <label><b>Password
        </b>
        </label>
        <input type="Password" name="Password" id="Pass"
placeholder="Password">
```

```
        <br><br>



        <input type="submit" name="loginadmin" id="loginadmin" value="Log in">


        <br><br>


        Forgot <a href="#">Password</a>
     </form>
</div>
</body>
</html>

<?php
   $hostname="localhost";
   $username="root";
   $pwd="";
   $dbname="office";



   $conn=new mysqli($hostname,$username,$pwd,$dbname);


   if($conn->connect_error){
   die("Error in connection".$conn->connect_error);
   exit();
```

```
  }
  else{



                        if(isset($_REQUEST['loginadmin'])){


                        $name=$_REQUEST['Name'];
                        $email=$_REQUEST['Email'];
                        $password=$_REQUEST['Password'];
                         $iquery="select Name,Email,Password from employee where
Designation='admin'";
                        $result=$conn->query($iquery);
                        if (!empty($name)||!empty($email)||!empty($password)) {
                                if($result->num_rows==0){
                                echo"<script>alert('Invalid Login Credential')</script>";
                                }
                                else{
                                if(($name=='admin1'&&
$email=='admin1@adminoffice.com' &&
$password=='admin1')||($name=='admin2' &&
$email=='admin2@adminoffice.com' && $password=='admin2')){


                      header("location:adminaction.php");


                                        }
                                else{
```

```
                        echo"<script>alert('Invalid Login
Credential')</script>";
                    }

                }




                }
                else{
                    echo'<script>alert("Invalid Login
Credential")</script>';
                }

            }
 }

?>
```

Output:



Admin Login Page

Name: admin1

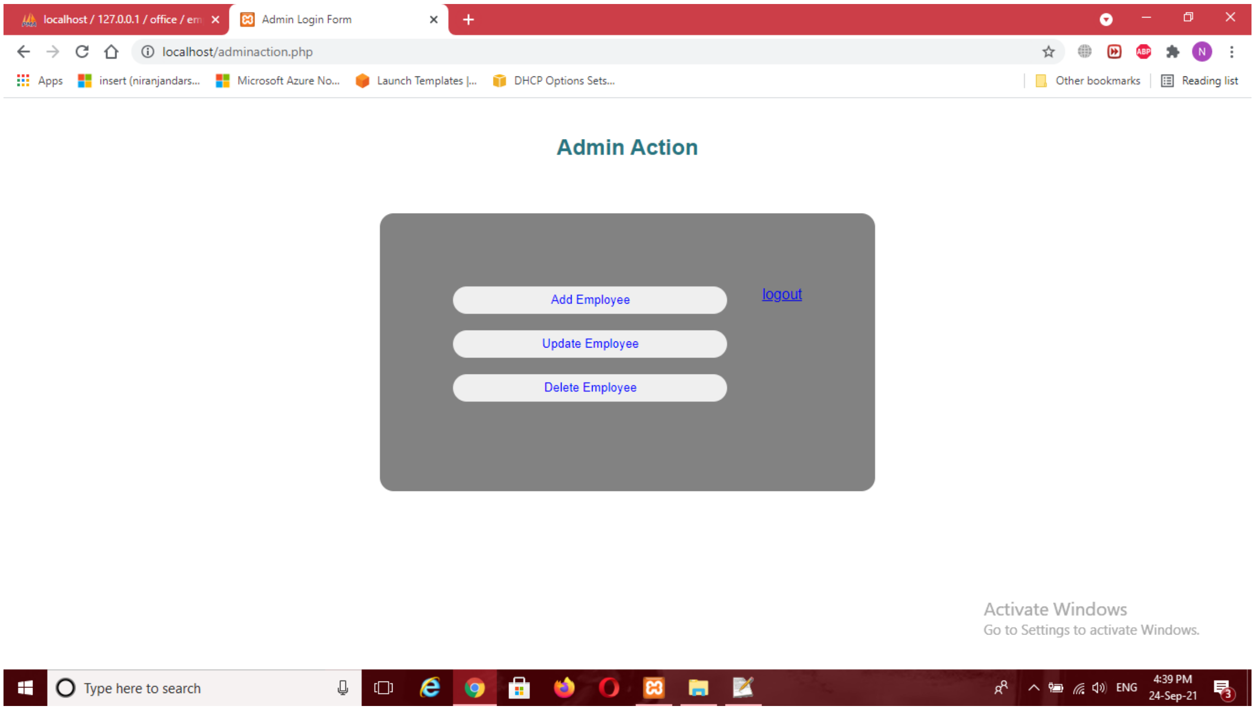
Email: admin1@adminoffice.com

Password: ••••••

Adminaction.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Admin Login Form</title>
    <link rel="stylesheet" type="text/css" href="styleadmin.css">
</head>
<body>
    <h2>Admin Action</h2><br>
    <div class="login">
                                <a href="logoutadmin.php">logout</a>
    <form id="login" method="POST" action="adminaction.php">
         <input type="submit" name="AddEmployee" id="log"  value="Add Employee">
<br><br>
                                      <input type="submit" name="UpdateEmployee" id="log" value="Update
Employee"> <br><br>
                                       <input type="submit" name="DeleteEmployee" id="log" value="Delete
Employee"> <br><br>



    </form>
</div>
</body>
</html>
<?php

   $hostname="localhost";
   $username="root";
   $pwd="";
   $dbname="office";


   $conn=new mysqli($hostname,$username,$pwd,$dbname);
   if($conn->connect_error){
   die("Error in connection".$conn->connect_error);


   }
   else{
   if(isset($_POST["AddEmployee"])){
                              //include("addemployee.php");
                              header("location:addemployee.php");
   }


   if(isset($_POST["UpdateEmployee"])){
                              //include("updateemployee.php");
                              header("location:updateemployee.php");
   }
   if(isset($_POST["DeleteEmployee"])){
                              //include("deleteemployee.php");
                              header("location:deleteemployee.php");
   }



   }



?>
```

Output:



Admin Action

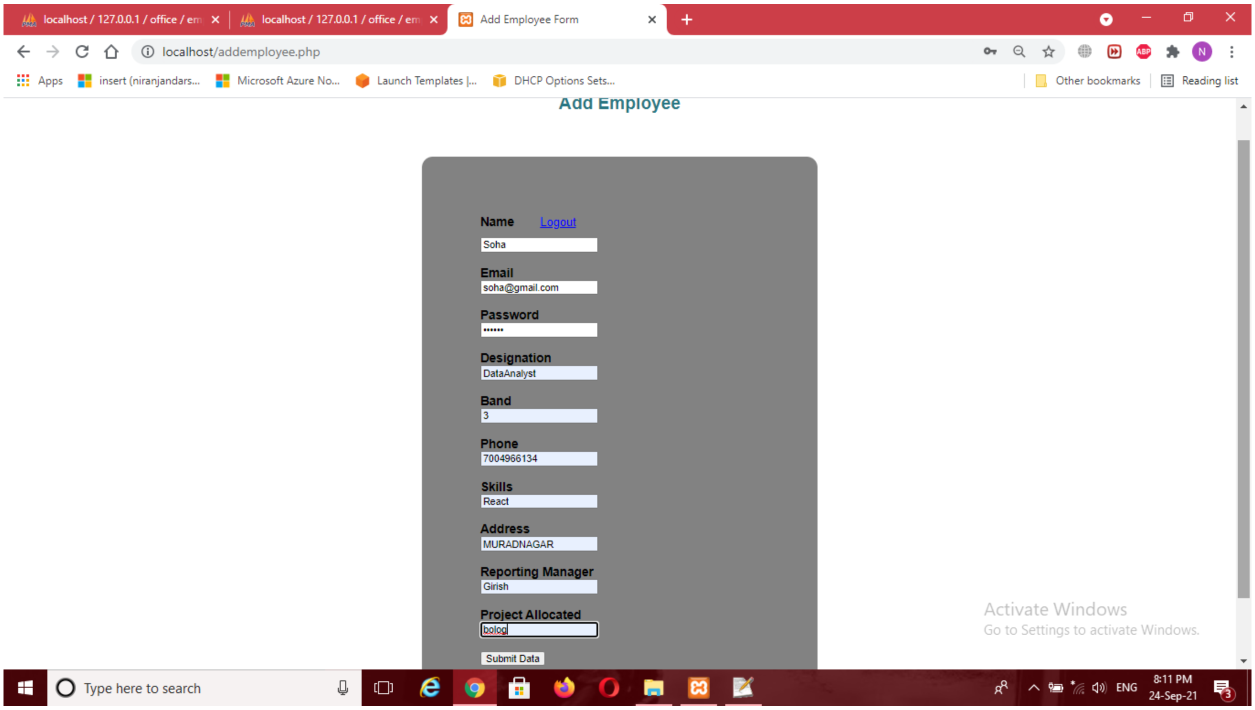
AddEmployee.php

```
<!DOCTYPE html>

<html>

<head>

    <title> Add Employee Form</title>

    <link rel="stylesheet" type="text/css" href="styleadmin.css">

</head>

<body>

    <h2>Add Employee </h2><br>

    <div class="login">

                              <a id="log" href="logoutadmin.php">Logout</a>


    <form id="login" method="POST" action="addemployee.php">

        <label><b>Name

        </b>

        </label>  <br>

        <input type="text" name="Name"  placeholder="Name">

        <br><br>

                              <label><b>Email

        </b>

        </label>    <br>

        <input type="text" name="Email"  placeholder="email">

        <br><br>

        <label><b>Password

        </b>

        </label>   <br>

        <input type="Password" name="Password" placeholder="Password">

        <br><br>
```

```
<label><b>Designation
        </b>
        </label>  <br>
        <input type="text" name="Designation"  placeholder="Designation">
        <br><br>

<label><b>Band
        </b>
        </label>  <br>
        <input type="text" name="Band"  placeholder="Band">
        <br><br>

<label><b>Phone
        </b>
        </label>  <br>
        <input type="text" name="Phone" placeholder="Phone">
        <br><br>

<label><b>Skills
        </b>
        </label>  <br>
        <input type="text" name="Skills"  placeholder="Skills">
        <br><br>

<label><b>Address
        </b>
```

```
    </label>  <br>
    <input type="text" name="Address"  placeholder="Address">
    <br><br>

<label><b>Reporting Manager
    </b>
    </label>  <br>
    <input type="text" name="ReportingManager"  placeholder="Reporting Manager">
    <br><br>

<label><b>Project Allocated
    </b>
    </label>  <br>
    <input type="text" name="ProjectAllocated"  placeholder="Project Allocated">
    <br><br>

                  <input type="submit" name="insertdata"  value="Submit Data">
    <br><br>



  </form>
</div>
</body>
</html>
<?php

  $hostname="localhost";
  $username="root";
  $pwd="";
  $dbname="office";


     $conn=new mysqli($hostname,$username,$pwd,$dbname);
  if($conn->connect_error){
  die("Error in connection".$conn->connect_error);


  }
  else{
  if(isset($_REQUEST['insertdata'])){
   $name=$_REQUEST["Name"];
  $email=$_REQUEST["Email"];
  $password=$_REQUEST["Password"];
  $designation=$_REQUEST["Designation"];
  $band=$_REQUEST["Band"];
  $phone=(int)$_REQUEST["Phone"];
  $skills=$_REQUEST["Skills"];
  $reportingmanager=$_REQUEST["ReportingManager"];
  $projectallocated=$_REQUEST["ProjectAllocated"];
  $address=$_REQUEST["Address"];
   $sqlquery="select Email,Name from employee where Email='$email' and Name='$name'";
  $iquery="insert into employee
values('$name','$email','$password','$designation','$band',$phone,'$skills','$reportingmanager','$p
rojectallocated','$address')";


  $result=$conn->query($sqlquery);
                              if($result->num_rows==0){
                              if
(!empty($name)||!empty($email)||!empty($password)||!empty($designation)||!empty($band)||!em
pty($phone)||!empty($skills)||!empty($reportingmanager)||!empty($projectallocated)||!empty($ad
dress)) {



  if($conn->query($iquery)){
    echo "<script>alert('Data inserted successfully')</script>";


  }


                      }


 else{
  echo "<script>alert('Fill All the Fields')</script>";
  }




  }


   else{
  echo "<script>alert('Data Already Present')</script>";
  }


  }


  }



?>
```

Output:



Add Employee

Name Logout

Soha

Email

soha@gmail.com

Password

••••••

Designation

DataAnalyst

Band

3

Phone

7004966134

Skills

React

Address

MURADNAGAR

Reporting Manager

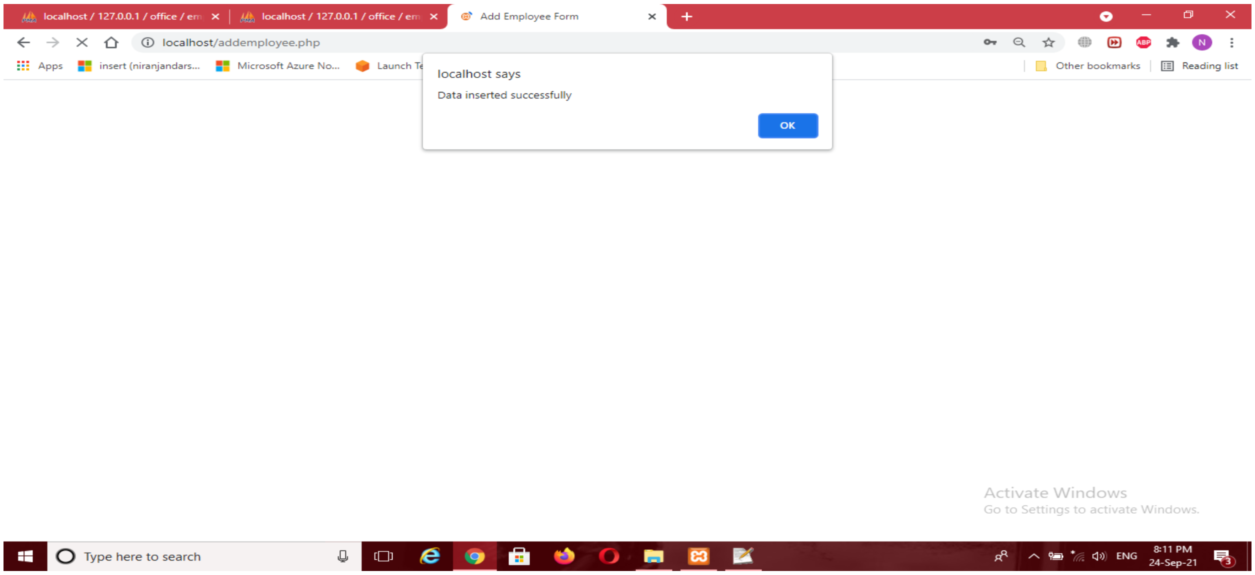
Girish

Project Allocated

bolog

Submit Data

Message After Insertion



localhost says

Data inserted successfully

UpdateEmployee.php

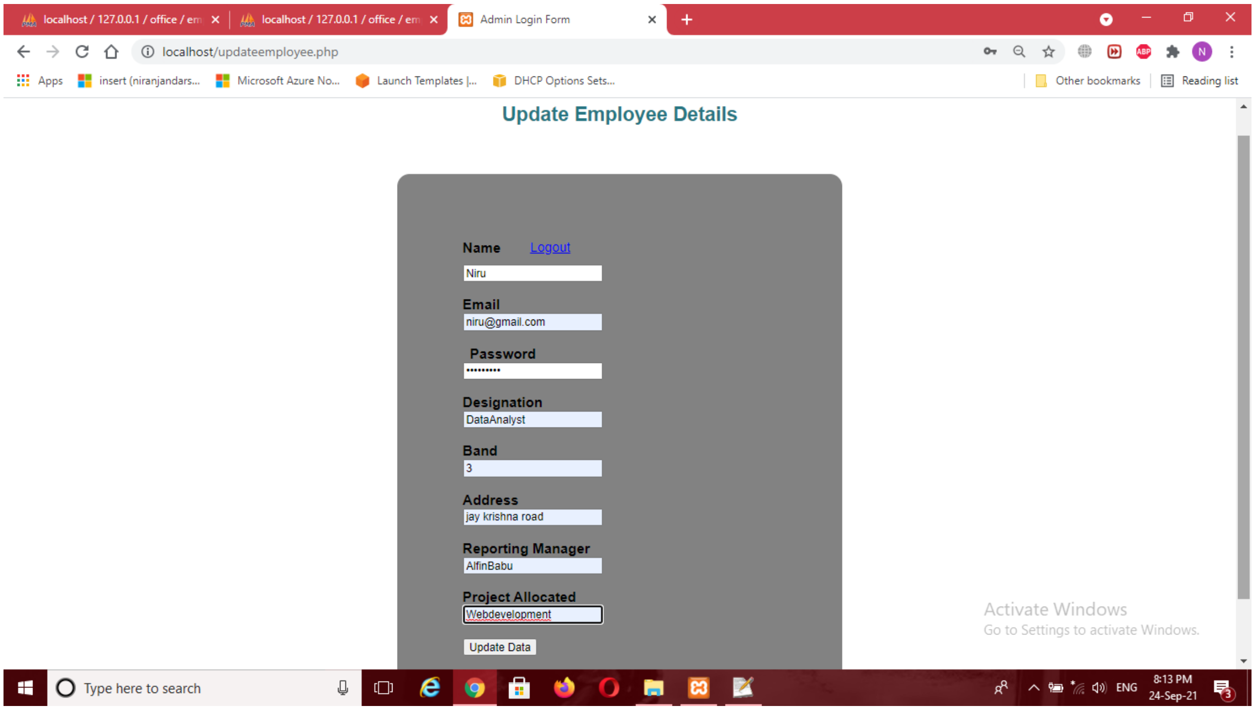
```
<!DOCTYPE html>
<html>
<head>
   <title>Admin Login Form</title>
   <link rel="stylesheet" type="text/css" href="styleadmin.css">
</head>
<body>
   <h2>Update Employee Details</h2><br>

   <div class="login">
                              <a id="log" href="logoutadmin.php">Logout</a>
   <form id="login" method="POST" action="updateemployee.php">
                                 <label><b>Name
       </b>
       </label>  <br>
       <input type="text" name="Name"  placeholder="Name" >
       <br><br>
                                 <label><b>Email
       </b>
       </label>    <br>
       <input type="text" name="Email"  placeholder="email" >
       <br><br>        &nbsp

        <label><b>Password
       </b>
       </label>   <br>
       <input type="Password" name="Password" placeholder="Password">
```

```
        <br><br>



<label><b>Designation
        </b>
        </label>  <br>
        <input type="text" name="Designation"  placeholder="Designation">
        <br><br>

<label><b>Band
        </b>
        </label>  <br>
        <input type="text" name="Band"  placeholder="Band">
        <br><br>



<label><b>Address
        </b>
        </label>  <br>
        <input type="text" name="Address"  placeholder="Address">
        <br><br>

<label><b>Reporting Manager
        </b>
        </label>  <br>
        <input type="text" name="ReportingManager"  placeholder="Reporting Manager">
        <br><br>
```

```
<label><b>Project Allocated
        </b>
        </label>  <br>
        <input type="text" name="ProjectAllocated"  placeholder="Project Allocated">
        <br><br>


 <input type="submit" name="updatedata"  value="Update Data">
        <br><br>


                                    </form>
</div>
</body>
</html>


<?php
  if(isset($_REQUEST['updatedata'])){
   $hostname="localhost";
   $username="root";
   $pwd="";
   $dbname="office";
   $name=$_REQUEST["Name"];
   $email=$_REQUEST["Email"];
  $password=$_REQUEST["Password"];
   $designation=$_REQUEST["Designation"];
   $band=$_REQUEST["Band"];
   $reportingmanager=$_REQUEST["ReportingManager"];
   $projectallocated=$_REQUEST["ProjectAllocated"];
   $address=$_REQUEST["Address"];
    $sqlquery="select Email,Name from employee where Email='$email' and Name='$name'";
                       $iquery="update employee set Password='$password',
Designation='$designation',
                       Band='$band',
ReportingManager='$reportingmanager',ProjectAllocated='$projectallocated',
                       Address='$address' where Email='$email' and Name='$name'";


   $conn=new mysqli($hostname,$username,$pwd,$dbname);
   if($conn->connect_error){
   die("Error in connection".$conn->connect_error);


   }
   else{
                          $result=$conn->query($sqlquery);
                        if($result->num_rows>0){
                          if (!empty($name)&&!empty($email)) {


                                  if($conn->query($iquery)){
                                          echo "<script>alert('Record Updated
Sucessfully')</script>";
                                          header("location:adminaction.php");



                                  }
```

```
                    }
                    else{
                    echo "<script>alert('Name and Email Must Be Filled')</script>";
                }
                }
                else{
                    echo "<script>alert('Record Not Updated Sucessfully')</script>";
                    }


                }
                }
?>
```

Output:



Update Employee Details

Name Logout
Niru

Email
niru@gmail.com

Password
•••••••••

Designation
DataAnalyst

Band
3

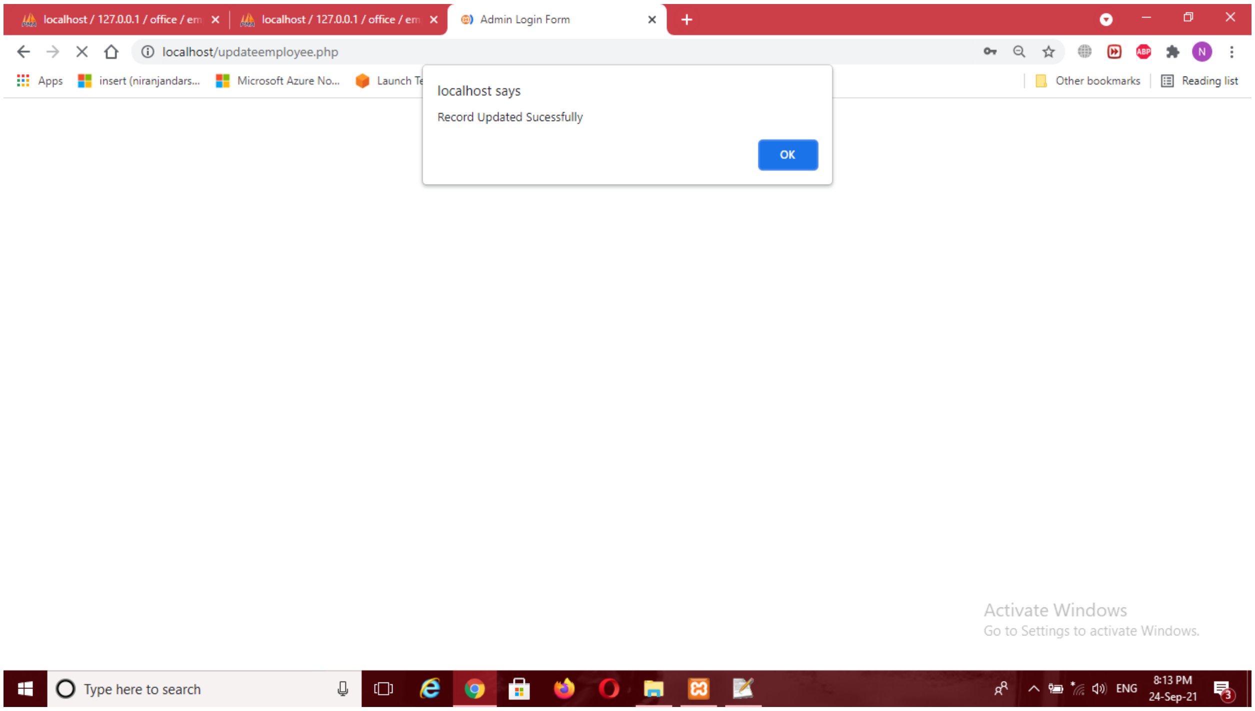
Address
jay krishna road

Reporting Manager
AlfinBabu

Project Allocated
Webdevelopment

Update Data

Message After Updation



localhost says

Record Updated Sucessfully

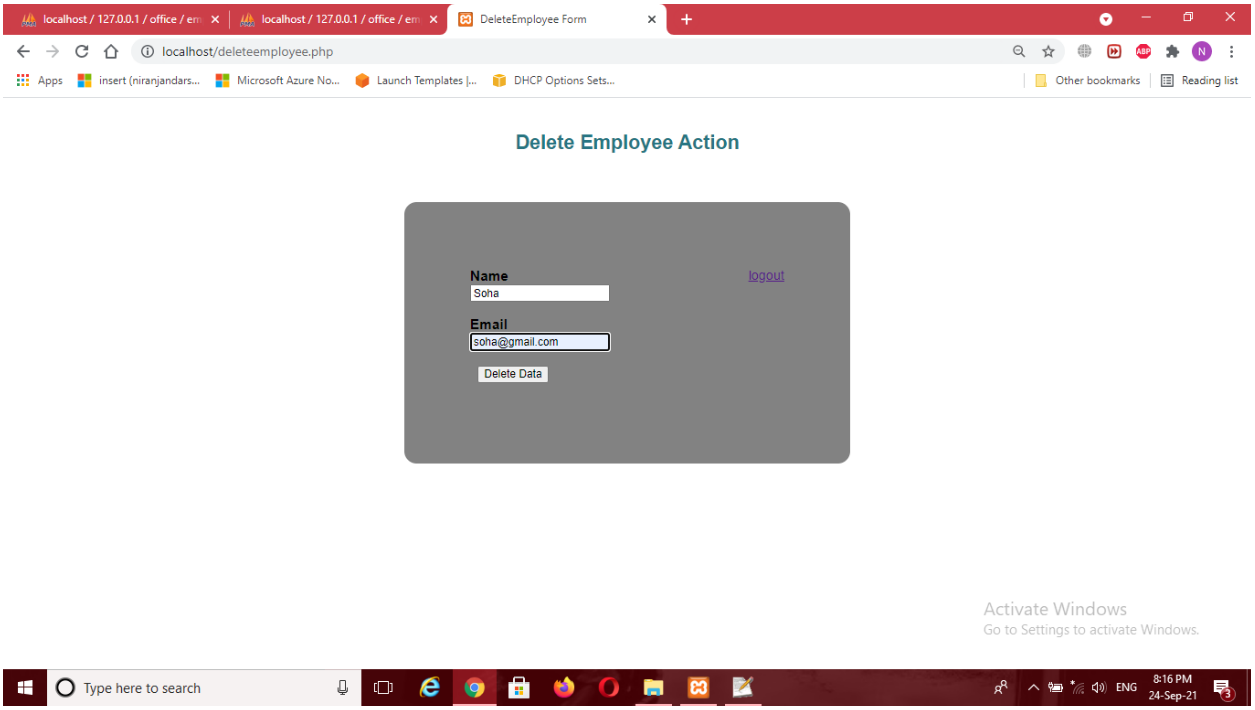
OK

DeleteEmployee.php

```
<!DOCTYPE html>

<html>

<head>

   <title>DeleteEmployee Form</title>

   <link rel="stylesheet" type="text/css" href="styleadmin.css">

</head>

<body>

   <h2>Delete Employee  Action</h2><br>

   <div class="login">

 <a href="logoutadmin.php">logout</a>

   <form id="login" method="POST" action="deleteemployee.php">

                              <label><b>Name

      </b>

      </label>  <br>

      <input type="text" name="Name"  placeholder="Name" >

      <br><br>

                              <label><b>Email

      </b>

      </label>    <br>

      <input type="text" name="Email"  placeholder="email" >

      <br><br>        &nbsp

                              <input type="submit" name="delete"  value="Delete Data">

      <br><br>



                              </form>

</div>
```

```
</body>
</html>



<?php
 if(isset($_REQUEST['delete'])){
  $hostname="localhost";
  $username="root";
  $pwd="";
  $dbname="office";
  $name=$_REQUEST["Name"];
  $email=$_REQUEST["Email"];
   $sqlquery="select Email,Name from employee where Email='$email' and Name='$name'";
                              $iquery="Delete from employee where  Name='$name' and Email='$email'";

  $conn=new mysqli($hostname,$username,$pwd,$dbname);
  if($conn->connect_error){
  die("Error in connection".$conn->connect_error);

  }
  else{



                                  $result=$conn->query($sqlquery);
                               if($result->num_rows>0){
                               if (!empty($name)&&!empty($email)) {

                                  if($conn->query($iquery)){
                                        //include("adminaction.php");
                                                echo "<script>alert('Record Deleted
Sucessfully')</script>";


                          }


                        }
                        else{
                          echo "<script>alert('Name and Email Must Be Filled')</script>";
                        }
                         }


                            else{
                                   echo "<script>alert('No such Employee Exist')</script>";
                          }




 }
 }
?>
```

Output:

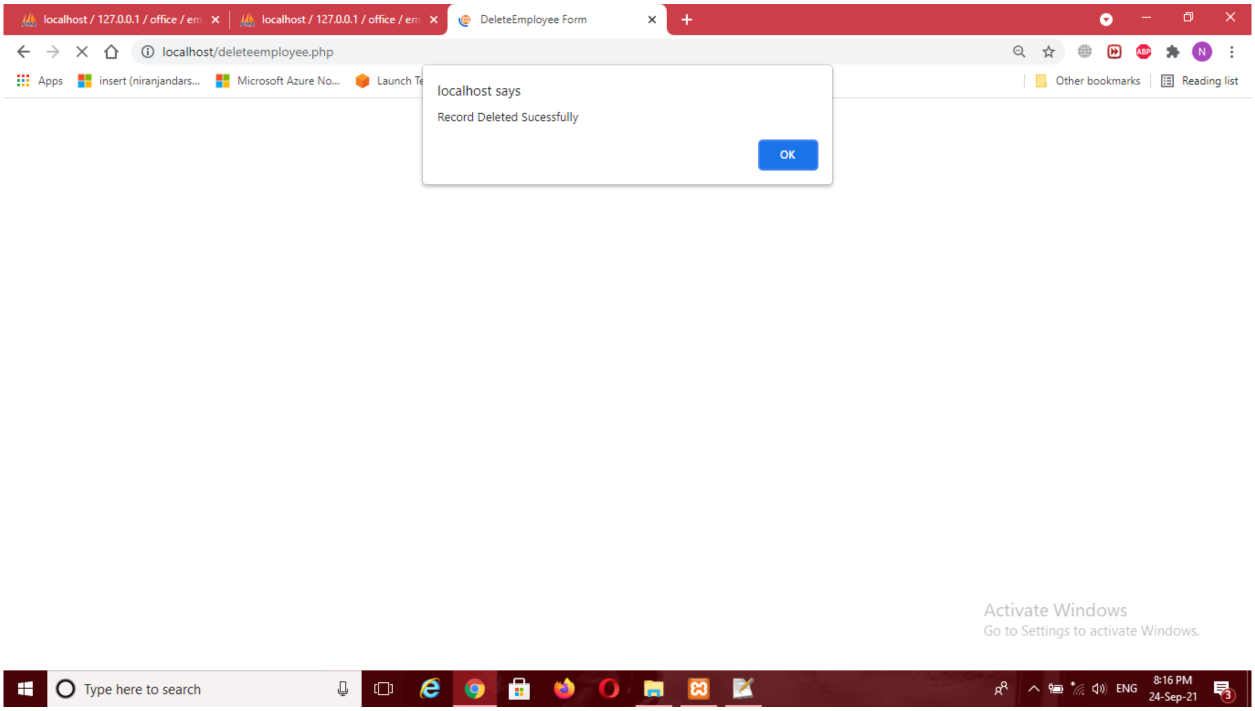


Delete Employee Action

Name
Soha

Email
soha@gmail.com

Message After Deletion



localhost says

Record Deleted Sucessfully

Logoutadmin.php

```
<?php
                session_start();
                unset($_SESSION['Name']);
                header("location:admin.php");
?>
```

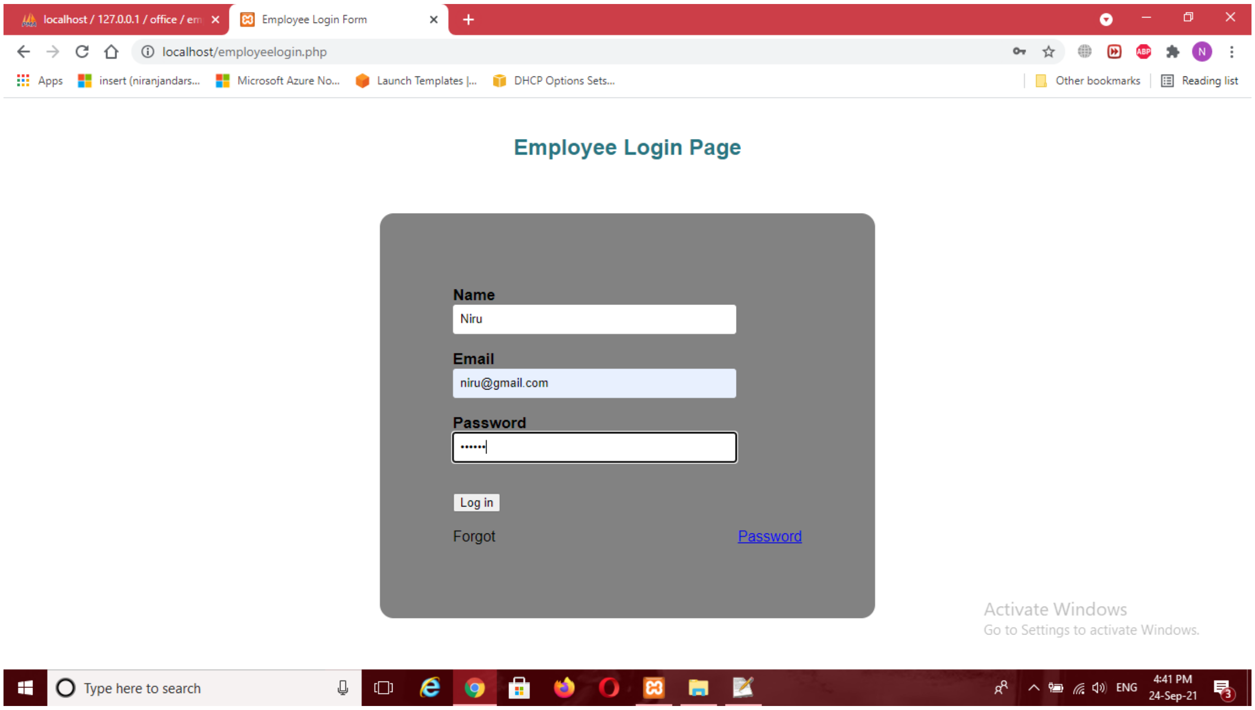
EmployeeLogin.php

```
<!DOCTYPE html>

<html>

<head>

   <title>Employee Login Form</title>

   <link rel="stylesheet" type="text/css" href="styleadmin.css">

</head>

<body>

   <h2>Employee Login Page</h2><br>

   <div class="login">

   <form id="login" method="POST" >

      <label><b>Name

      </b>

      </label>  <br>

      <input type="text" name="Name" id="Uname" placeholder="Username">

      <br><br>

                              <label><b>Email

      </b>

      </label>    <br>

      <input type="text" name="Email" id="Email" placeholder="email">

      <br><br>

      <label><b>Password

      </b>

      </label>

      <input type="Password" name="Password" id="Pass" placeholder="Password">

      <br><br>

                              <div id="message"><p id="welcome"></p></div>
```

```
        <input type="submit" name="loginadmin" id="loginadmin" value="Log in">

        <br><br>

        Forgot <a href="#">Password</a>
    </form>
</div>
</body>
</html>

<?php

   $hostname="localhost";
   $username="root";
   $pwd="";
   $dbname="office";


   $conn=new mysqli($hostname,$username,$pwd,$dbname);
   if($conn->connect_error){
   die("Error in connection".$conn->connect_error);

   }


   else{
```

```
 if(isset($_REQUEST['loginadmin'])){
                        $name=$_REQUEST['Name'];
  $email=$_REQUEST['Email'];
                     $password=$_REQUEST['Password'];


                      $iquery="select Name,Email,Password from employee where
Designation<>'admin' and Name='$name' and
  Email='$email' and Password='$password'";


  $result=$conn->query($iquery);
                     if (!empty($name)||!empty($email)||!empty($password)) {
                     if($result->num_rows==0){
                     echo"<script>alert('Invalid Login Credential')</script>";
                     }
                     else{
                        session_start();
                        $_SESSION['Email']=$email;
                        //echo"HI". $_SESSION['Email'];
                        header("location:employeeaction.php");
                        }
                     }
                     else{
                                echo'<script>alert("Fill all field")</script>';
                        }



                     }
 }
?>
```

Output:



Employee Login Page

Name
Niru

Email
niru@gmail.com

Password
••••••

Log in

Forgot Password

Employeeaction.php

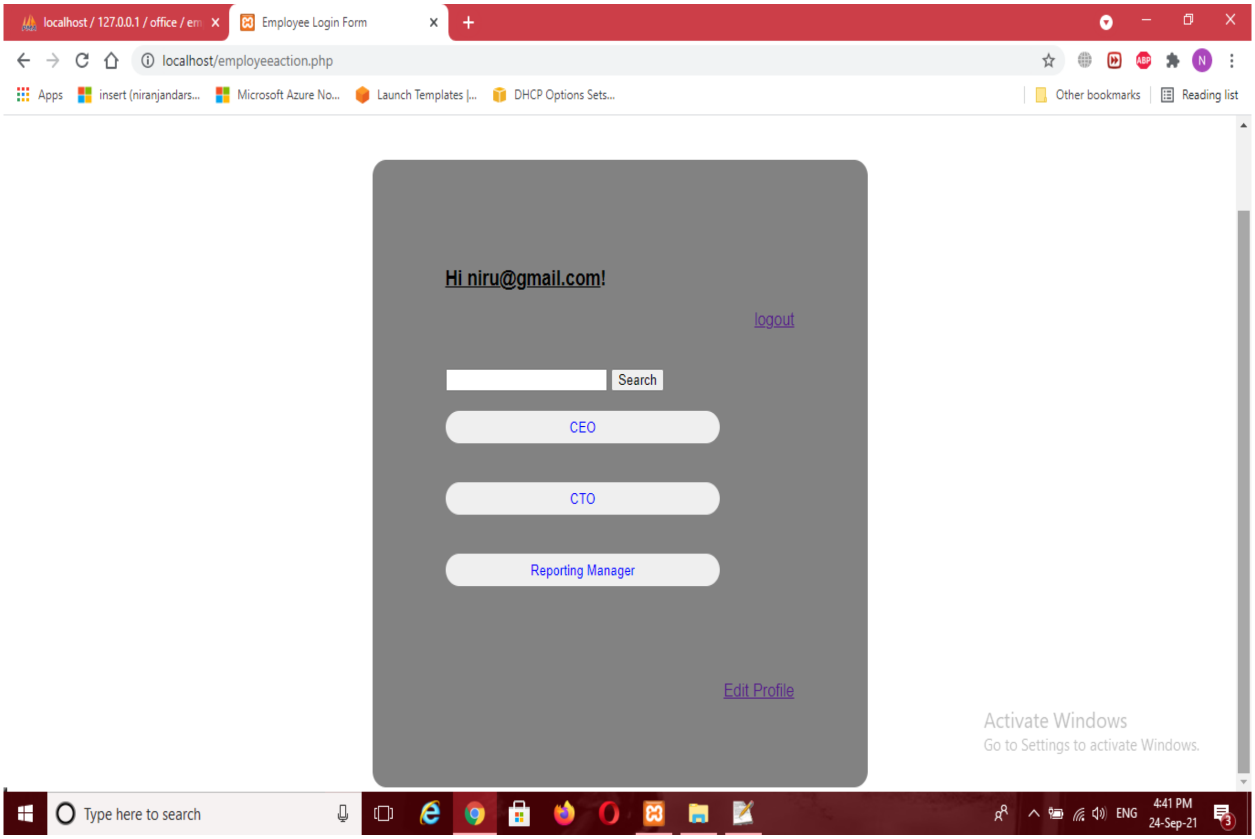
```
<!DOCTYPE html>
<html>
<head>
    <title>Employee Login Form</title>
    <link rel="stylesheet" type="text/css" href="styleadmin.css">
</head>
<body>
    <h2>Employee Action Page</h2><br>
 <div class="login">
      <?php
                                    session_start();
?>


                                    <h3><u><?php echo" Hi ".$_SESSION['Email'];?></u>!</h3>
                                                <a href="logout.php">logout</a>
<br><br>



    <form id="login" method="POST" action="employeeaction.php">
                                     <br>


         <input type="text"  name="datasearch" >


                                    <input type="submit"   name="search"  value="Search">


                                    <br><br>
         <input type="submit" id="log" name="ceo" value="CEO">

        </b>
        <br>

        <br><br>

                              <input type="submit"   id="log" name="cto"  value="CTO">          </b>

         <br>

        <br><br>

        <input type="submit"   id="log" name="reportingmanager" value="Reporting Manager">


        <br><br>
                              <div id="message"><p id="welcome"></p></div>




        <br><br>
                              <br>
          <a href="personaldetail.php">Edit Profile</a>
    </form>
</div>
```

```
<table border="2">

<?php

   $hostname="localhost";
   $username="root";
   $pwd="";
   $dbname="office";
   $conn=new mysqli($hostname,$username,$pwd,$dbname);



   if($conn->connect_error){
   die("Error in connection".$conn->connect_error);

   }


   else{

                              if(isset($_POST['search'])){

                              $searchd=$_REQUEST['datasearch'];

                                 $searchquery= "Select
Name,Designation,Email,Band,Phone,Skills,ProjectAllocated,Address from employee where
Name Like'%$searchd%' or ProjectAllocated Like'%$searchd%'
                                 or Address Like '%$searchd%' or Designation Like '%$searchd%'";
```

```
    $result=$conn->query($searchquery);
if( $result->num_rows>0){
?>
<tr>
<td>Name</td>
<td>'Designation</td>
    <td>Email</td>
<td>Band</td>
<td>Phone</td>
<td>Skills</td>
<td>ProjectAllocated</td>
    <td>Address</td>



</tr>
<?php
    while($rows=$result->fetch_assoc()){
?>


<tr>
<td><?php echo $rows['Name'];?></td>
<td><?php echo $rows['Designation'];?></td>
    <td><?php echo $rows['Email'];?></td>
<td><?php echo $rows['Band'];?></td>
<td><?php echo $rows['Phone'];?></td>
<td><?php echo $rows['Skills'];?></td>
<td><?php echo $rows['ProjectAllocated'];?></td>
    <td><?php echo $rows['Address'];?></td>
    <?php echo "<td><a href='viewprofile.php?email=$rows[Email]'>View Profile</td>";

?>

</tr>



<?php
}
}
}

if(isset($_POST['reportingmanager'])){
    //include('projectmanager.php');
    header("location:projectmanager.php");

}
if(isset($_POST['ceo'])){
    //include('ceo.php');
    header("location:ceo.php");

}
if(isset($_POST['cto'])){
    //include('cto.php');
    header("location:cto.php");

}
```

```
  }
?>
</table>
</body>
</html>
```

Output:


localhost / 127.0.0.1 / office / em
Employee Login Form
localhost/employeeaction.php
Hi niru@gmail.com!
logout
Search
CEO
CTO
Reporting Manager
Edit Profile
Activate Windows
Go to Settings to activate Windows.
Type here to search
4:41 PM
24-Sep-21

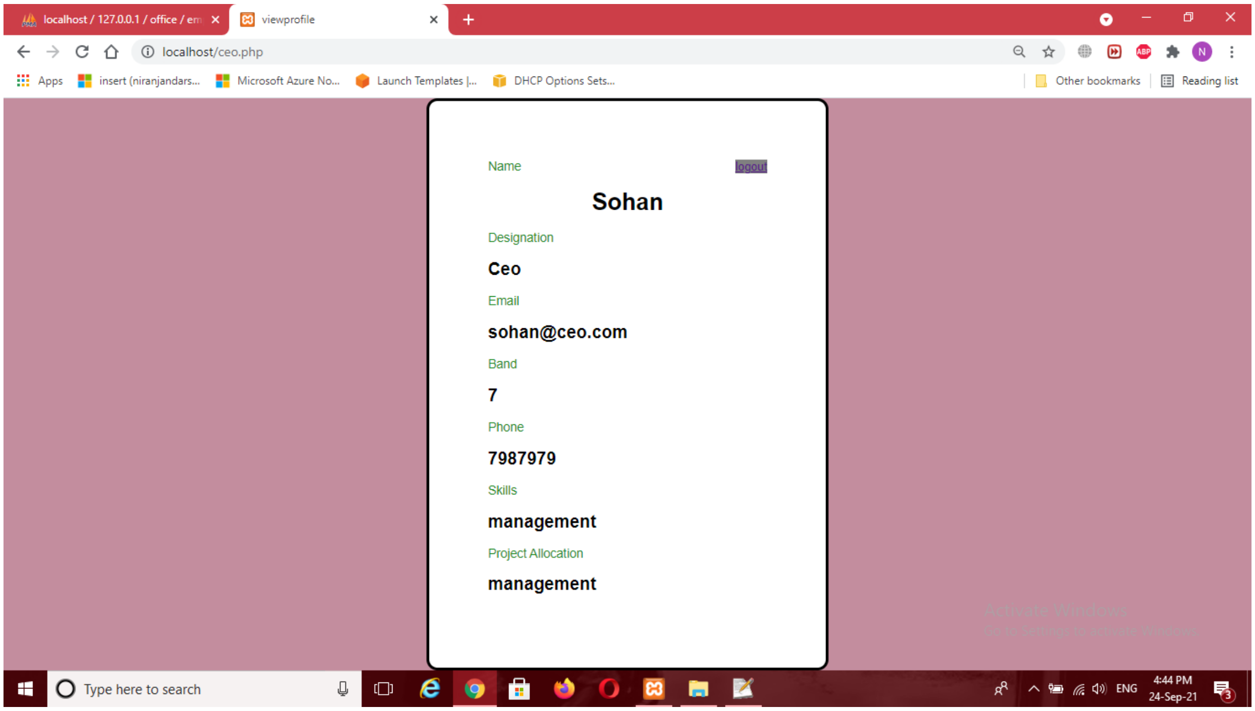
CEO.php

```
<!DOCTYPE html>
<html>
                <head>
                <title>viewprofile</title>
                <link rel="stylesheet" type="text/css" href="styleprofile.css">
                </head>
                <body>
                <div class="login">
                        <a href="logout.php">logout</a>



<?php
                $hostname="localhost";
                $username="root";
                $pwd="";
                $dbname="office";
                $query="select * from employee where Designation='Ceo'";
                $conn=new mysqli($hostname,$username,$pwd,$dbname);

                if($conn->connect_error){
                die("Connection not established".$conn->connect_error);

                }
```

```
else{

$result=$conn->query($query);

if($result->num_rows>0){
while($rows=$result->fetch_assoc()){

?>
<label> Name</label>
<h1><?php echo $rows['Name'];?></h1>
<label> Designation</label>
<h2><?php echo $rows['Designation'];?></h2>
<label> Email</label>
<h2><?php echo $rows['Email'];?></h2>
<label> Band</label>
<h2><?php echo $rows['Band'];?></h2>
<label> Phone</label>
<h2><?php echo $rows['Phone'];?></h2>
<label> Skills</label>
<h2><?php echo $rows['Skills'];?></h2>
<label> Project Allocation</label>
<h2><?php echo $rows['ProjectAllocated'];?></h2>


<?php

}
}
```

```
                    else{
                    echo"Record Not found";
                    }
                    }

?>
                    </div>
                    </body>
                    </html>
```

Output:



Name

Sohan

Designation

Ceo

Email

sohan@ceo.com

Band

7

Phone

7987979

Skills

management

Project Allocation

management

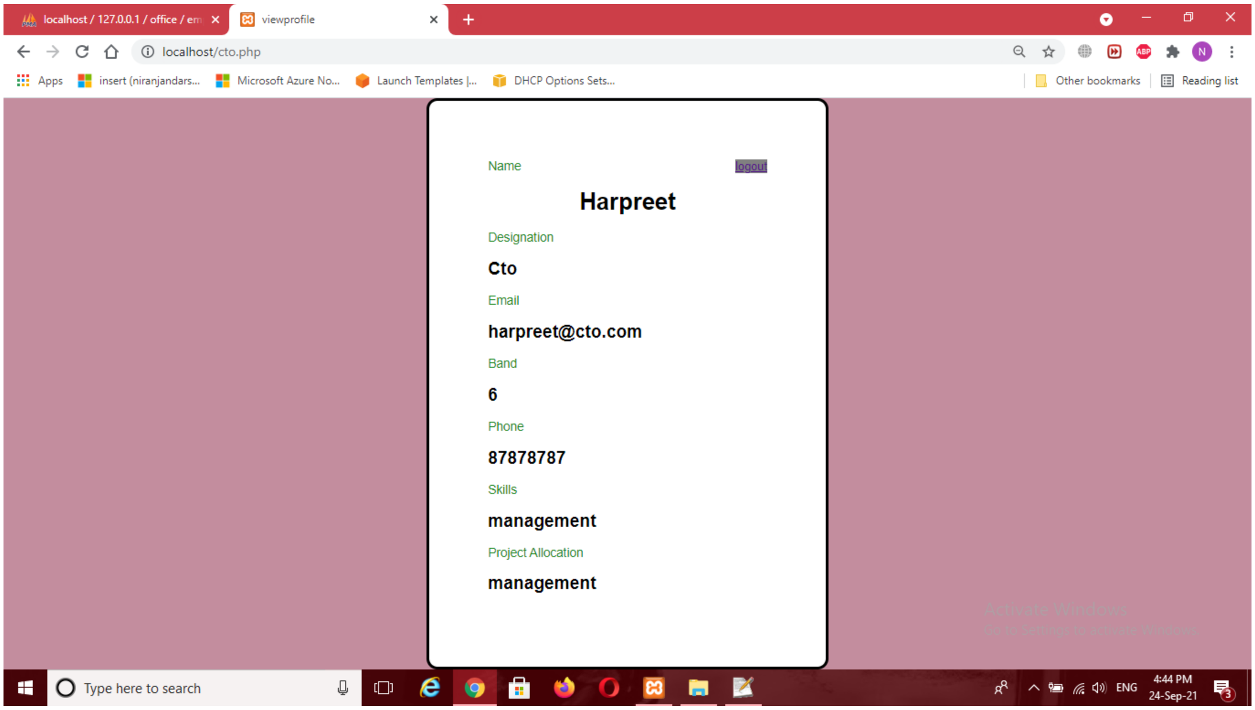
CTO.php

```
<!DOCTYPE html>
<html>
                <head>
                <title>viewprofile</title>
                <link rel="stylesheet" type="text/css" href="styleprofile.css">
                </head>
                <body>
                <div class="login">
                        <a href="logout.php">logout</a>




<?php
                $hostname="localhost";
                $username="root";
                $pwd="";
                $dbname="office";
                $query="select * from employee where Designation='Cto'";
                $conn=new mysqli($hostname,$username,$pwd,$dbname);

                if($conn->connect_error){
                die("Connection not established".$conn->connect_error);

                }
                else{
```

```
$result=$conn->query($query);

if($result->num_rows>0){
while($rows=$result->fetch_assoc()){

?>
<label> Name</label>
<h1><?php echo $rows['Name'];?></h1>
<label> Designation</label>
<h2><?php echo $rows['Designation'];?></h2>
<label> Email</label>
<h2><?php echo $rows['Email'];?></h2>
<label> Band</label>
<h2><?php echo $rows['Band'];?></h2>
<label> Phone</label>
<h2><?php echo $rows['Phone'];?></h2>
<label> Skills</label>
<h2><?php echo $rows['Skills'];?></h2>
<label> Project Allocation</label>
<h2><?php echo $rows['ProjectAllocated'];?></h2>



<?php

}
}

else{
```

```
                echo"Record Not found";
                }
                }

?>
                </div>
                </body>
                </html>
```

Output:



Name

Harpreet

Designation

Cto

Email

harpreet@cto.com

Band

6

Phone

87878787

Skills

management

Project Allocation

management

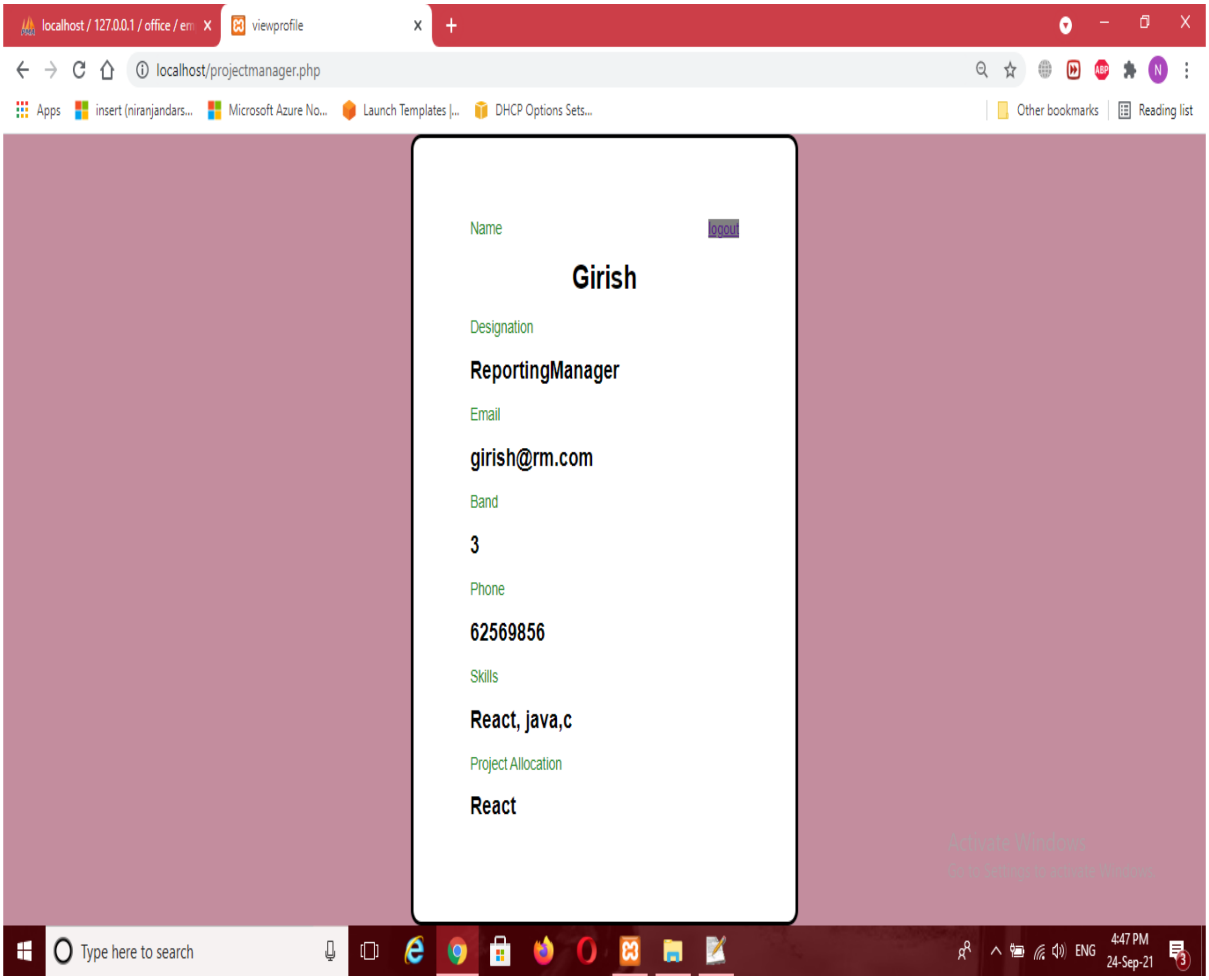
Projectmanager.php

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
                <head>
                <title>viewprofile</title>
                <link rel="stylesheet" type="text/css" href="styleprofile.css">
                </head>
                <body>
                <div class="login">
                        <a href="logout.php">logout</a>



<?php
                $hostname="localhost";
                $username="root";
                $pwd="";
                $dbname="office";
                $email=$_SESSION['Email'];
                $query="select * from employee where Name=(select
ReportingManager from employee where Email='$email') and
                ProjectAllocated=(select ProjectAllocated from employee where
Email='$email')";
```

```
$conn=new mysqli($hostname,$username,$pwd,$dbname);

if($conn->connect_error){
die("Connection not established".$conn->connect_error);

}
else{

$result=$conn->query($query);

if($result->num_rows>0){
while($rows=$result->fetch_assoc()){

?>
<label> Name</label>
<h1><?php echo $rows['Name'];?></h1>
<label> Designation</label>
<h2><?php echo $rows['Designation'];?></h2>
<label> Email</label>
<h2><?php echo $rows['Email'];?></h2>
<label> Band</label>
<h2><?php echo $rows['Band'];?></h2>
<label> Phone</label>
<h2><?php echo $rows['Phone'];?></h2>
<label> Skills</label>
```

```
                    <h2><?php echo $rows['Skills'];?></h2>
                    <label> Project Allocation</label>
                    <h2><?php echo $rows['ProjectAllocated'];?></h2>


                    <?php

                    }
                    }

                    else{
                    echo"Record Not found";
                    }
                    }

?>



                    </div>
                    </body>
                    </html>
```

Output:



Name

Girish

Designation

ReportingManager

Email

girish@rm.com

Band

3

Phone

62569856

Skills

React, java,c

Project Allocation

React

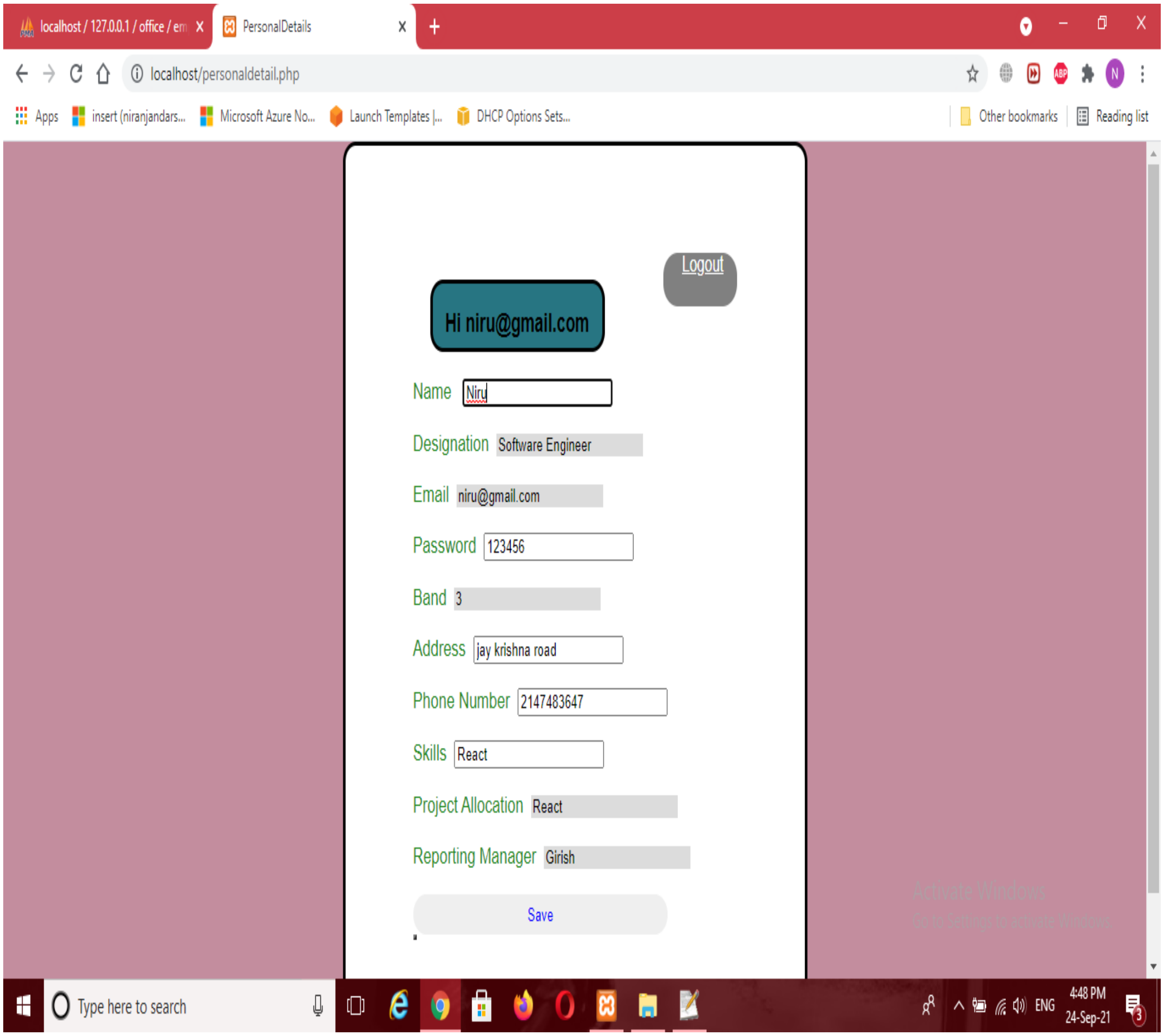
Personaldetails.php

```
<!DOCTYPE html>
<html>
                    <head>
                    <title>PersonalDetails</title>
                    <link rel="stylesheet" type="text/css" href="styleprofile.css">
                    </head>
                    <body>
                    <div class="login">
                             <a id="log" href="logout.php">Logout</a>

                    <table border="2">
                    <form method="GET" action="personaldetail.php">
                    <?php
  session_start();

                    ?><div class="title"><h3><?php echo "Hi  ".
$_SESSION['Email'];?></h3></div>

<?php
                    $hostname="localhost";
                    $username="root";
                    $pwd="";
                    $dbname="office";
                    $email=$_SESSION['Email'];

                    $query="select * from employee where Email='$email'";
                    $conn=new mysqli($hostname,$username,$pwd,$dbname);

                    if($conn->connect_error){
                    die("Connection not established".$conn->connect_error);

                    }
                    else{

                    $result=$conn->query($query);

                    if($result->num_rows>0){
                    while($rows=$result->fetch_assoc()){

                    ?>

                    <label> Name </label>&nbsp;
                    <input type="text" name="Name" value="<?php echo
$rows['Name'];?>"></input>
                    <br><br>
                    <label> Designation</label>&nbsp;
                    <input type="text" name="Designation" id="readonly" value="<?php echo
$rows['Designation'];?>" readonly></input>
                    <br><br>
                    <label> Email</label>&nbsp;
                      <input type="text" name="Email" id="readonly" value="<?php echo
$rows['Email'];?>"readonly></input>

                    <br><br>
                    <label> Password</label>&nbsp;

                    <input type="text" name="Password" value="<?php echo
$rows['Password'];?>"></input>

                    <br><br>

                    <label>Band</label>&nbsp;

                    <input type="text" name="Band" id="readonly" value="<?php echo
$rows['Band'];?>" readonly></input>

<br><br>

                    <label> Address</label>&nbsp;

                    <input type="text" name="Address" value="<?php echo
$rows['Address'];?>"></input>

                    <br><br>

                    <label> Phone Number</label>&nbsp;

                    <input type="text" name="Phone" value="<?php echo
$rows['Phone'];?>"></input>

<br><br>

                    <label> Skills</label>&nbsp;

                    <input type="text" name="Skills" value="<?php echo
$rows['Skills'];?>"></input>

<br><br>

                    <label> Project Allocation</label>&nbsp;

                    <input type="text" name="ProjectAllocated"  id="readonly" value="<?php
echo $rows['ProjectAllocated'];?>" readonly></input>

                    <br><br>

                    <label> Reporting Manager</label>&nbsp;

                    <input type="text" name="ReportingManager"  id="readonly" value="<?php
echo $rows['ReportingManager'];?>" readonly></input>

                    <br><br>


                    <input type="submit" name="save" id="save" value="Save">

                    </form>
```

```
<?php

}

}




if(isset($_REQUEST['save'])){
$name=$_GET['Name'];
$skills=$_GET['Skills'];
$password=$_GET['Password'];
$phone=$_GET['Phone'];
$address=$_GET['Address'];
$email=$_GET['Email'];

$queryedit="update employee SET Name='$name',Skills='$skills',
Password='$password',Phone='$phone',
Address='$address' where Email='$email'";

$data=mysqli_query($conn,$queryedit);
if($data){
   echo"<script> alert('Record updated Successfully');</script>";
}
}

}
```

```
?>
                </div>
                </table>
                </body>
                </html>
```

Output:



Hi niru@gmail.com

Name Niru

Designation Software Engineer

Email niru@gmail.com

Password 123456

Band 3

Address jay krishna road

Phone Number 2147483647

Skills React

Project Allocation React

Reporting Manager Girish

Viewprofile.php

```
<!DOCTYPE html>
<html>
        <head>
        <title>viewprofile</title>
        <link rel="stylesheet" type="text/css" href="styleprofile.css">
        </head>
        <body>

        <div class="login">
        <h1>Profile</h1>
            <a id="log" href="logout.php"><b>Logout</b></a><br><br>

        <table border="2">

<?php
        $hostname="localhost";
        $username="root";
        $pwd="";
        $dbname="office";
        $email=$_GET['email'];
        $query="select * from employee where Email='$email'";
        $conn=new mysqli($hostname,$username,$pwd,$dbname);

        if($conn->connect_error){
        die("Connection not established".$conn->connect_error);

        }
```

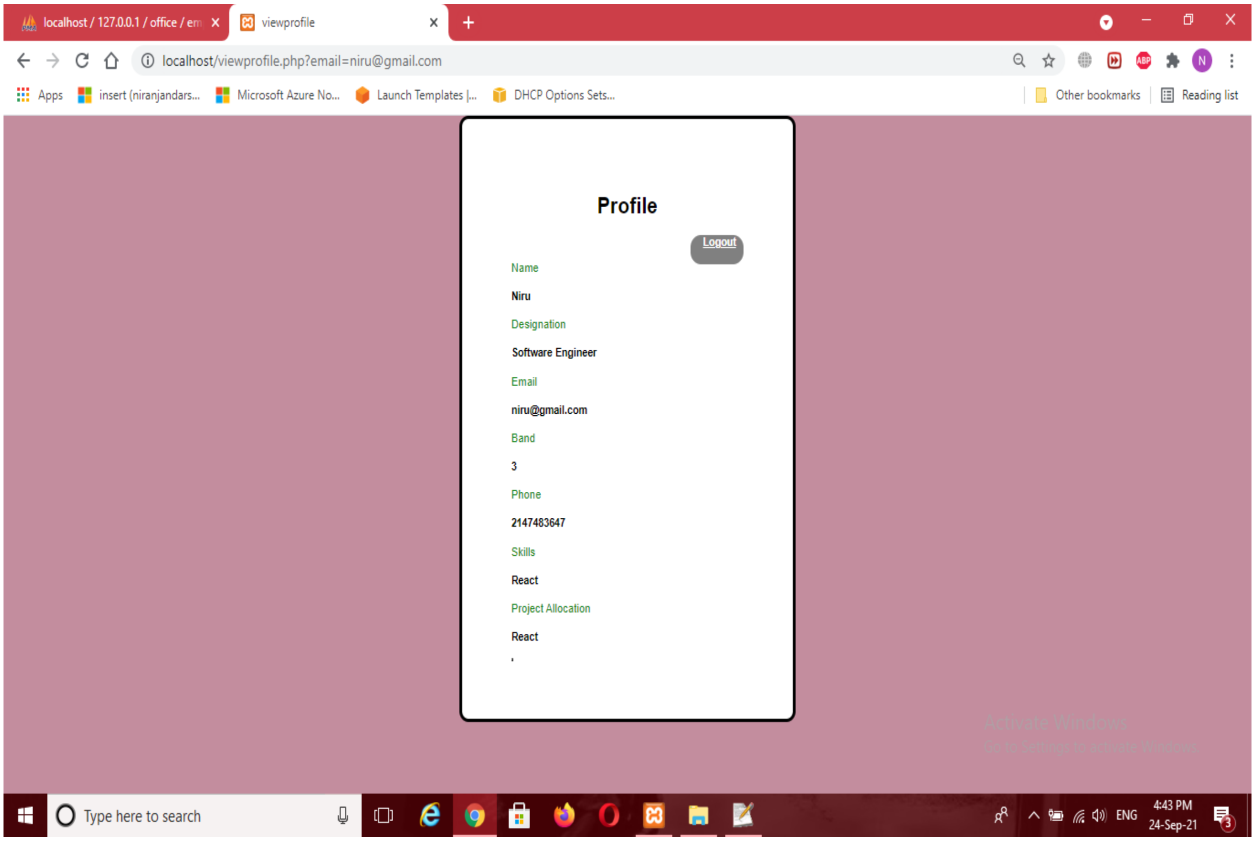
```
else{


$result=$conn->query($query);


if($result->num_rows>0){
while($rows=$result->fetch_assoc()){


?>
<label> Name</label>
<h4><?php echo $rows['Name'];?></h4>
<label> Designation</label>
<h4><?php echo $rows['Designation'];?></h4>
<label> Email</label>
<h4><?php echo $rows['Email'];?></h4>
<label> Band</label>
<h4><?php echo $rows['Band'];?></h4>
<label> Phone</label>
<h4><?php echo $rows['Phone'];?></h4>
<label> Skills</label>
<h4><?php echo $rows['Skills'];?></h4>
<label> Project Allocation</label>
<h4><?php echo $rows['ProjectAllocated'];?></h4>



<?php


}
}
```

```
          else{
          echo"Record Not found";
          }
          }

?>
          </div>
          </table>
          </body>
          </html>
```

Logout.php

```
<?php
                    session_start();
                    unset($_SESSION['Name']);
                    header("location:employeelogin.php");
?>
```

Output:



Profile

Logout

Name

Niru

Designation

Software Engineer

Email

niru@gmail.com

Band

3

Phone

2147483647

Skills

React

Project Allocation

React

Chapter 8

Testing

Testing is an investigation conducted to provide stakeholders with information about the quality of the software product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects), and verifying that the software product is fit for use.

Software testing involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

- meets the requirements that guided its design and development,
- responds correctly to all kinds of inputs,
- performs its functions within an acceptable time,
- it is sufficiently usable,
- can be installed and run in its intended environments, and
- Achieves the general result its stakeholder's desire.

8(i) Static vs. dynamic testing:

There are many approaches available in software testing. Reviews, walkthroughs, or inspections are referred to as static testing, whereas executing programmed code with a given set of test cases is referred to as dynamic testing.

Static testing is often implicit, like proofreading, plus when programming tools/text editors check source code structure or compilers (pre-compilers) check syntax and data flow as static program analysis. Dynamic testing takes place when the program itself is run. Dynamic testing may begin before the program is 100% complete in order to test particular sections of code and are applied to discrete functions or modules. Typical techniques for these are either using stubs/drivers or execution from a debugger environment.

8(ii) White-box testing:

White-box testing (also known as clear box testing, glass box testing, transparent box testing and structural testing) verifies the internal structures or workings of a program, as opposed to the functionality exposed to the end-user. In white-box testing, an internal perspective of the system (the source code), as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g., in-circuit testing (ICT).

While white-box testing can be applied at the unit, integration, and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements.

Techniques used in white-box testing include:

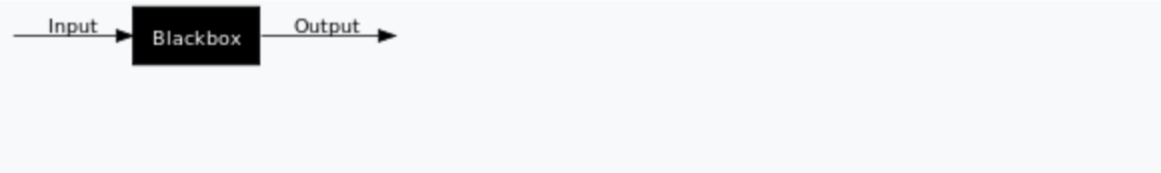
- API testing – testing of the application using public and private APIs (application programming interfaces)
- Code coverage – creating tests to satisfy some criteria of code coverage (e.g., the test designer can create tests to cause all statements in the program to be executed at least once)
- Fault injection methods – intentionally introducing faults to gauge the efficacy of testing strategies
- Mutation testing methods
- Static testing methods

Code coverage tools can evaluate the completeness of a test suite that was created with any method, including black-box testing. This allows the software team to examine parts of a system that are rarely tested and ensures that the most important function points have been tested. Code coverage as a software metric can be reported as a percentage for:

- Function coverage, which reports on functions executed
- Statement coverage, which reports on the number of lines executed to complete the test
- Decision coverage, which reports on whether both the True and the False branch of a given tens.
- it has been executed

100% statement coverage ensures that all code paths or branches (in terms of control flow) are executed at least once. This is helpful in ensuring correct functionality, but not sufficient since the same code may process different inputs correctly or incorrectly. Pseudo-tested functions and methods are those that are covered but not specified (it is possible to remove their body without breaking any test case).

11(iii) Black-box testing



Black-box testing (also known as functional testing) treats the software as a "black box," examining functionality without any knowledge of internal implementation, without seeing the source code. The testers are only aware of what the software is supposed to do, not how it does it. Black-box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, state transition tables, decision table testing, fuzz testing, model-based testing, use case testing, exploratory testing, and specification-based testing.

Specification-based testing aims to test the functionality of software according to the applicable requirements This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behavior), either "is" or "is not" the same as the expected value specified in the test case. Test cases are built around specifications and requirements, i.e., what the application is supposed to do. It uses external descriptions of the software, including, specifications, requirements, and designs to derive test cases. These tests can be functional or non-functional, though usually functional.

Specification-based testing may be necessary to assure correct functionality, but it is insufficient to guard against complex or high-risk situations.

One advantage of the black box technique is that no programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may emphasize different areas of functionality. On the other hand, black-box testing has been said to be "like a walk in a dark labyrinth without a flashlight." Because they do not examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case or leaves some parts of the program untested.

This method of test can be applied to all levels of software testing: unit, integration, system and acceptance. It typically comprises most if not all testing at higher levels, but can also dominate unit testing as well.

CHAPTER 9

Conclusion

This report will allow the users to store the details of admin and employee of the organisaztion. This report package will allow storing the details of all the data related to the firms. The system will be storing enough to withstand regressive yearly operations under conditions where the database is maintained and cleared over a certain time of span. The implementation of the system in the Organisation will considerably reduce data entry, time and also provide past information.

CHAPTER 13

Bibliography

(i)www.tutorialponit.org

(ii)www.wikipedia.com

(iii)www.studytonight.com

(iv)www.w3schools.com

.