TEACHER-SUBJECT ALLOCATION SYSTEAM

## MINI PROJECT REPORT

***Submitted by***

**KHAIRNAR AISHWARYA (02)**

**JAGTAP URMILA ANIL (07)**

***In partial fulfilment of the requirements for the Degree of***

# BBA C.A

## In

Bachelor of Computer Application

# Mahatma Gandhi Vidyamandir M.S.G. College

**Loknete Vyankatrao Hire road Malegaon camp**

Tal: Malegaon Dist: Nashik, Pincode:423105

## DEPARTMENT OF BBA C.A.

**M.S.G COLLEGE MALEGAON**

CERTIFICATE

*Certified that this is a bonafide record of the project*

*work entitled*

*“TEACHER SUBJECT ALLOCATION SYSTEM”*

*done by the following students*

*KHAIRNAR AISHWARYA(02)*

*JAGTAP URMILA ANIL(07)*

*Of the IVth semester, Bachelor of Computer Application in the year 2024 in partial fulfilment of the requirements to the award of Degree Bachelor of BBA C.A. of MSG College Malegaon.*

*Mr. D. J. Deore Mrs. Komal Shinde Head of the Department Project Guide*

*Place: Malegaon Date:24/1/24*

## ACKNOWLEDGEMENT

Here we gladly present this project report on **“TEACHER SUBJECT ALLOCATION SYSTEM”** as part of the 4th semester BCA. At this time of submitting this report we use this opportunity to mention those people who are with us along the work. We take this occasion to thank God, almighty for blessing us with his grace and taking our endeavour to a successful culmination. We extend our sincere and heartfelt thanks to our esteemed ***guide, Mrs. Komal Shinde*** for providing us with the right guidance and advice at the crucial junctures and for showing us the right way. We extend our sincere thanks to our respected ***Head Of the Department Mr. D.J. Deore***, for allowing us to use the facilities available. We would like to thank the other faculty members also, at this occasion. Last but not the least; we would like to thank friends for the support and encouragement they have given us during the course of our work.

***Submitted by***

**KHAIRNAR AISHWARYA**

**JAGTAP URMILA ANIL**

# 

## ABSTRACT

## 

Traditional subject allocation is becoming obsolete due to its inconvenience, inefficiency, and low accuracy. Currently, web-based subject allocation systems have been widely implemented due to the development of Web technologies. This paper introduces a new web-based subject allocation system, which combines all useful features in other commercial systems and implements new functions that are practical in subject allocation. Its powerful features and friendly user interfaces allow teachers and admin to handle their subject allocations in a convenient, efficient, and systematical way. In addition, this system also has very good portability and extensibility, and the system security has been strongly enhanced by multiple security strategies.

**TABLE OF CONTENTS**

**1. INTRODUCTION**  05

1.1 PROJECT OVERVIEW 05

1.2 PROJECT OBJECTIVES 05

**2. SYSTEM ANALYSIS** 07

2.1 EXISTING SYSTEM 07

2.2 PROPOSED SYSTEM 08

**3. FEASIBILITY STUDY** 09

3.1 TECHNICAL FEASIBILITY 10

3.2 ECONOMICAL FEASIBILITY 11

3.3 OPERATIONAL FEASIBILITY 12

**4. REQUIREMENT ANALYSIS AND SPECIFICATION** 13

4.1 HARDWARE CONFIGURATION 14

4.2 SOFTWARE CONFIGURATION 15

**5. SYSTEM DESIGN 16**

5.1 PROJECT DESIGNING 16

5.2 DIAGRAM 17

5.3 DATABASE DESIGN (TABLE STRUCTURE) 21

5.4 DATA FLOW DIAGRAM 25

**6. SYSTEM IMPLEMENTATION STRUCTURE** 28

6.1 SYSTEM IMPLEMENTATION 28

6.2 SYSTEM TESTING 29

**CONCLUSION 30**

**REFERENCES 31**

**APPENDIX**

**1.SAMPLE SCREEN SHOT 32**

**2.SOURCE CODE 42**

**CHAPTER 1**

**INTRODUCTION**

**1.1 PROJECT OVERVIEW**

“Teacher Subject Allocation Management System” is web based application system which helps to allocate subject to the teachers. In this system educational administrator allocate different subjects to the teacher and teacher check that allotment by their employee id.

**1.2** **PROJECT OBJECTIVES**

* Provide an interactive platform between administrator and teachers.
* It is very much faster than manual system.
* Easy and fastest record finding technique.
* It is very much flexible to work.
* It is very user oriented.
* Data can be stored for a longer period.

**CHAPTER 2**

**SYSTEM ANALYSIS**

**2.1 EXISTING SYSTEM**

The existing system of teacher subject allocation typically involves school administrators or educational authorities assigning teachers to specific subjects based on their qualifications, expertise, experience, and sometimes preferences. This allocation process aims to ensure that each subject area is covered adequately and that teachers are placed where they can best utilize their skills and knowledge. It may also involve considerations such as class sizes, student needs, and overall curriculum requirements.

Implementing a teacher subject allocation system can offer several advantages:

**ADVANTAGES:**

* Optimal Resource Utilization.
* Specialization and Expertise.
* Teacher Satisfaction.
* Consistency and Continuity.
* Customization and Flexibility.
* Equity and Fairness.
* Professional Development.
* Improved Student Engagement.

**2.2 PROPOSED SYSTEM**

This proposed system is a tool for educational industry to be used for the Subject allotment. This system is a web based system which will help educational organizations such as universities, colleges and schools to improve the process of managing subject allotment. This system can help universities and other educational firms to increase their productivity and quality of services.

**Advantages of proposed System:**

* Easy to generate report.
* It is very much faster than manual system.
* Easy and fastest record finding technique.
* It is very much flexible to work.
* Man power required is very less.
* Data can be stored for a longer period

**CHAPTER 3**

**FEASIBILITY STUDY**

A feasibility analysis is undertaken to determine the possibility or probability of either improving the existing system or developing a completely new system. It helps to obtain an overview of the problem and to get rough assessment of whether feasible solution exists. There are three aspects in feasibility study portion of the preliminary investigation.

**Types of feasibility study:-**  
1) Technical Feasibility  
2) Economical Feasibility

3)Operational Feasibility

**3.1 TECHNICAL FEASIBITY:**

This involves questions such as whether the technology needed for the system exists, how difficult it will be to build, and whether the firm has enough experience using that technology. The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs, and Procedures. This can be quantified in terms of volumes of data, trends, frequency of updating, etc. in order to estimate whether the new  
system will perform adequately or not.

The technical feasibility in the proposed system deals with the technology used in the system. It deals with the hardware and software used in the system whether they are of latest technology or not. It happens that after a system is prepared a new technology arises and the user wants the system based on that technology.

Thus it is important to check the system to be technically feasible. The minimum memory requirement is 32MB of RAM while 64MB is better to have for better performance. As far as software is concerned, MySQL and PHP should be installed on the server.

**3.2 ECONOMICAL FEASIBILITY:**

Economic analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known as cost/benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits out weight costs, then the decision is made to design and implement the system.

Implementation of this system will be a lifetime investment, which will ensure returns to the store of good services and market value throughout the future. So the system is found economically feasibility.

**3.3 OPERATIONAL FEASIBILITY:**

It is a measure of how well a proposed system solves the problems, and takes advantages of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

Operational feasibility covers two aspects. One is the technical performance aspect and other is the acceptance within the organization. Operational feasibility determines how the proposed system will fit the current operations and what, if any job restructuring and retraining may be needed to implement the system.

In the system operational feasibility checks, whether the user who is going to use the system is able to work with the software with which the system id coded and also the mind of the user going to use system. If the user does not understand or is able to work on the system further development is waste.

The system is easy to learn and it will require a very short time to learn the

operation of the system for a person having knowledge in accounting. So that

system was operationally feasible

**CHAPTER 4**

**REQUIREMENT ANALYSIS AND SPECIFICATION**

**PROJECT MODULES:**

In this project, we use PHP and **MySQL** database. It has one modules i.e Admin

#### ADMIN MODULE:

* Admin is the super user of the website who can manage everything on the website. Admin can log in through the login page

* **Dashboard**: In this section, admin can see all detail in brief like the total course, total subjects and total teachers.

*  **Course**: In this section, admin can manage course (add/update/delete). *** Subject**: In this section, admin can mangae subject (add/update/delete). * **Teacher:** In this section, admin can manage teacher (add/update).

* **Subject Allocation:** In this section, admin can allocate subjects to their teachers.

** **Search:** In this section, admin can search uploaded details of subject allotment Admin can also update his profile, change the password and recover the password.

#### Brief of Home Page

It is home page of ***“Teacher Subject Allocation Management System”*** on this teacher can view allotment of subject by the help of their employee id and name.

# SYSTEM REQUIREMENTS

**4.1 HARDWARE CONFIGURATION**

The section of hardware configuration is an important task related to the software development; insufficient random access memory may adversely affect the speed and efficiency of the entire system. The process should be powerful enough to handle the entire operation. The hard disk should have sufficient capacity to store the file and application.

Processor :Core i3 and above

Processor speed :1.4 GHz Onwards

System memory :128 Mb minimum 256 Mb recommended

Cache size :512 KB

RAM :1 GB(Minimum)

Network card :Any card can provide a 100mbps speed

Network connection :UTP or Coaxial cable connection

Printer Dpi :Inkjet/Laser Color printer provides at least 1000

Hard disk :80Gb

Monitor :SVGA Color 15”

Mouse :104 keys US Key Serial, USB or PS/2

Modem :56.6 Kbps

**4.2 SOFTWARE CONFIGURATION**

A major element in building a system is the section of compatible software since the software in the market is experiencing geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system.

This document gives a detailed description of the software requirement

specification. The study of requirement specification is focused especially on the functioning of the system. It allows the developer or analyst to understand the system, function to be carried out, the performance level to be obtained and corresponding interfaces to be established.

Front end tool :Visual Studio Code Editor

Backend :MySQL XAMPP Server

Operating system :Windows 10

Client Side :HTML, CSS ,JS

**CHAPTER 5**

**SYSTEM DESIGN**

**5.1 PROJECT DESIGNING**

**Design Document**

* The entire system is projected with a physical diagram which specifics the actual storage parameters that are physically necessary for any database to be stored on to the disk. The overall systems existential idea is derived from this diagram.
* The relation upon the system is structure through a conceptual ER- Diagram, which not only specifics the existential entities but also the standard relations through which the system exists and the cardinalities that are necessary for the system state to continue.
* The content level DFD is provided to have an idea of the functional inputs and outputs that are achieved through the system. The system depicts the input and output standards at the high level of the systems existence.

* 1. **FLOW CHARTS**

**i) User flow chart**

**TSAS**

**Got Website**

**Check allotment**

**View Allotted Subjects**

**ii) Admin Flow Chart**

**TSAS**

**Login**

**Allocate Subjects**

**Manage Course/Subject/Teacher**

**Use Case Diagrams Admin**

**Sign in**

**Dashboard**

**Course (Add/Delete)**

**Subject (Add/Delete)**

**Teacher (Add/Update)**

**Allot Subjects**

**Search allotted subjects**

**View Profile**

**Change passward**

**ENTITY-RELATIONSHIP Diagrams**

E-R (Entity-Relationship) Diagram is used to represents the relationship between entities in the table.

The symbols used in E-R diagrams are:

SYMBOL PURPOSE

Represents Entity sets.

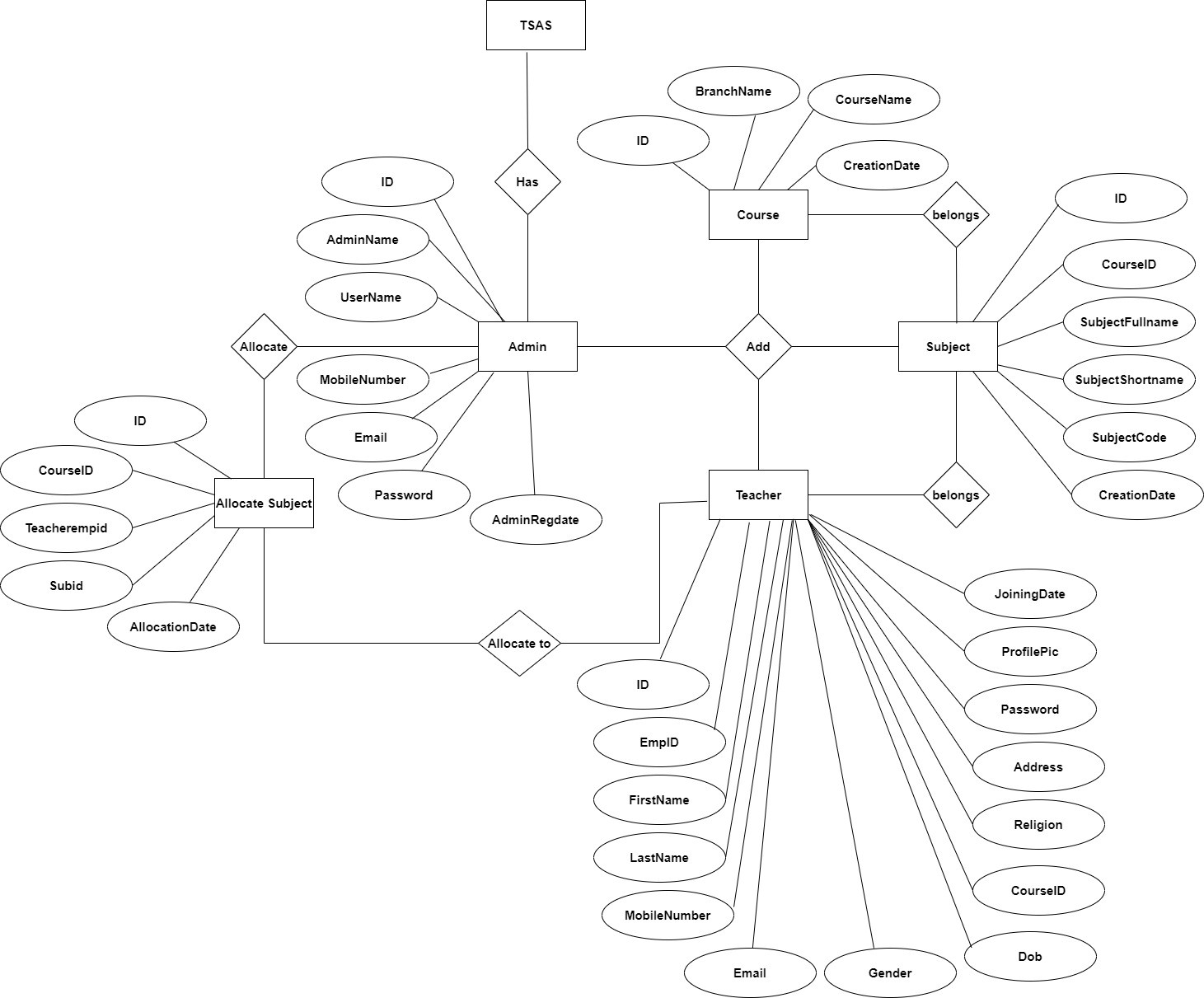
Represent attributes.

Represent Relationship Sets.

Line represents flow

Structured analysis is a set of tools and techniques that the analyst. To develop a new kind of a system:

The traditional approach focuses on the cost benefit and feasibility analysis, Project management, and hardware and software selection a personal considerations.



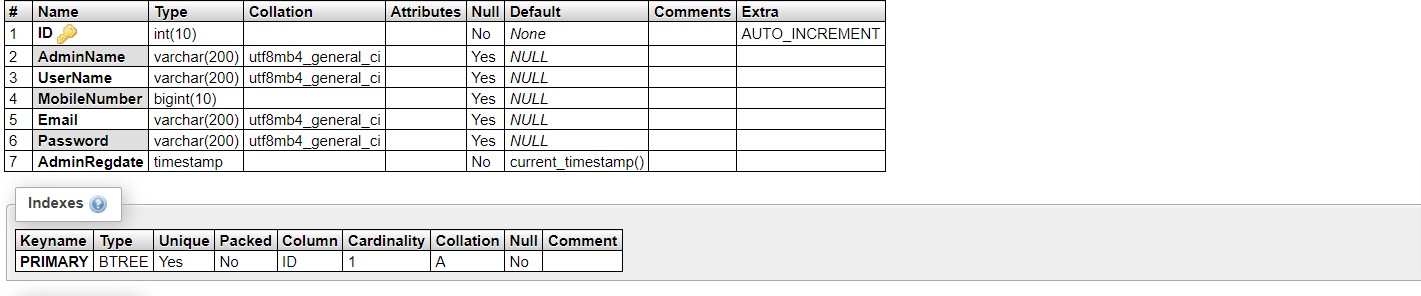
### 5.3 DATABASE DESIGN

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

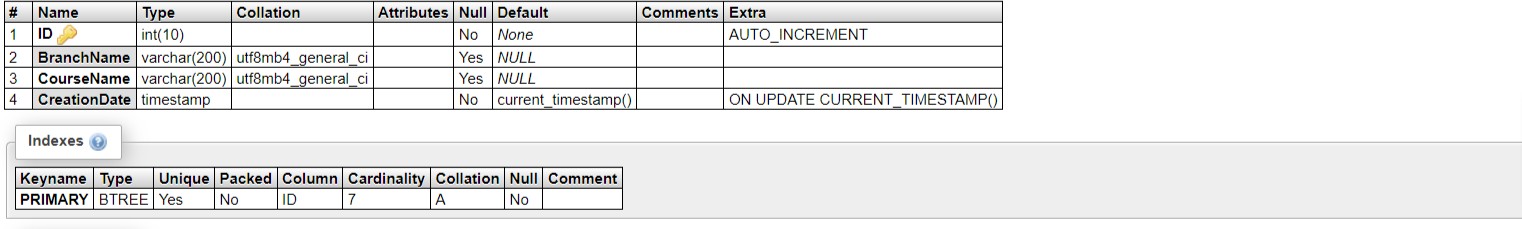
A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MySQL database has been chosen for developing the relevant databases.

**Teacher Subject Allocation System (TSAS) MySQL tables :**

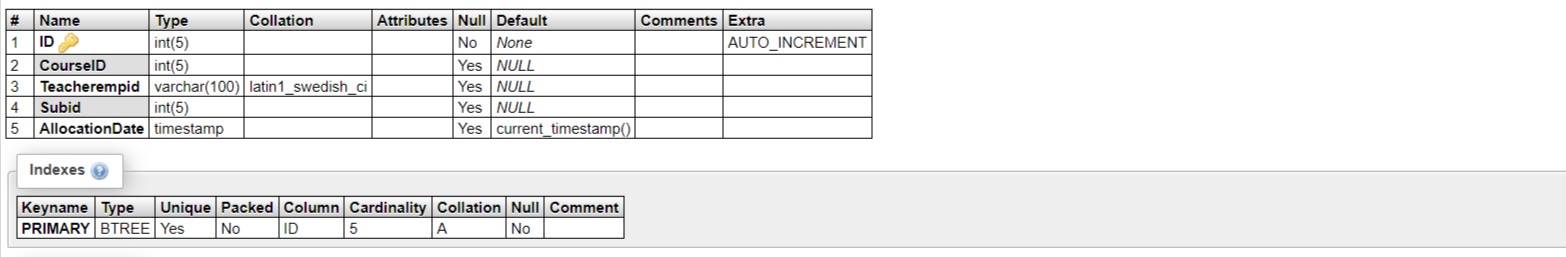
**tbladmin table Structure** : This table store the login details of admin.



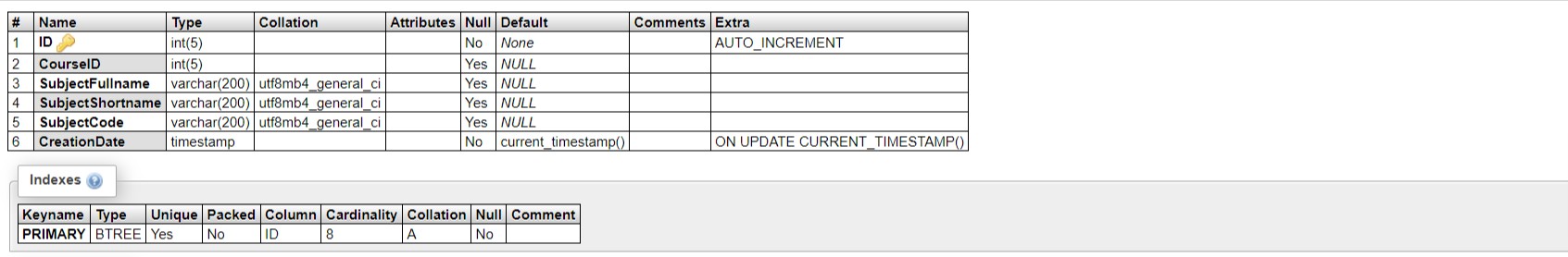
**tblcourse table structure :** This table store the detail of courses.



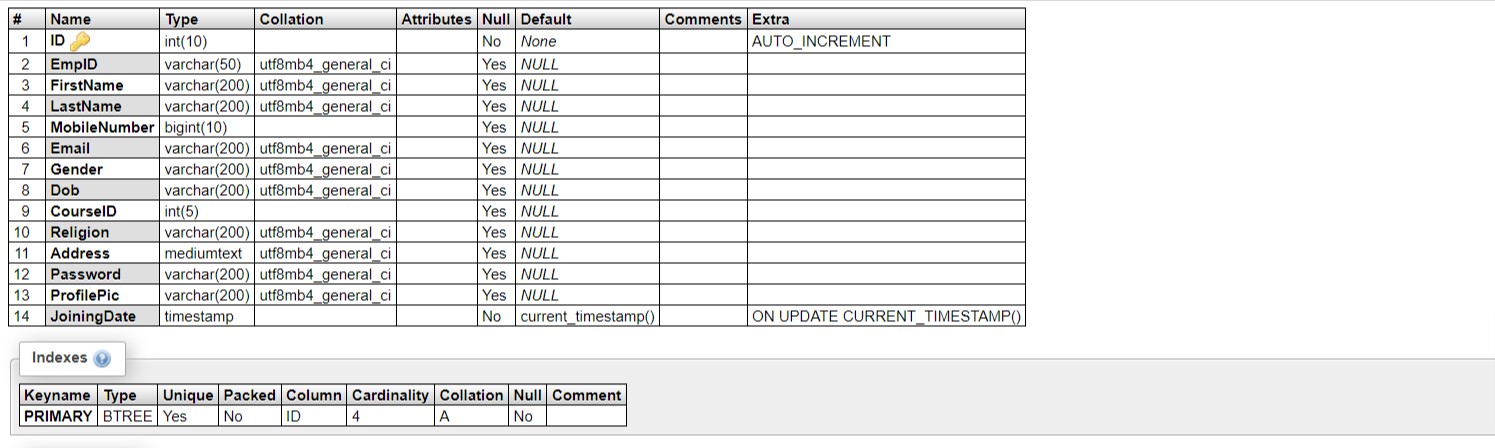
**tblsuballocation table structure :** This table store subject allocation details which is add by administrator.



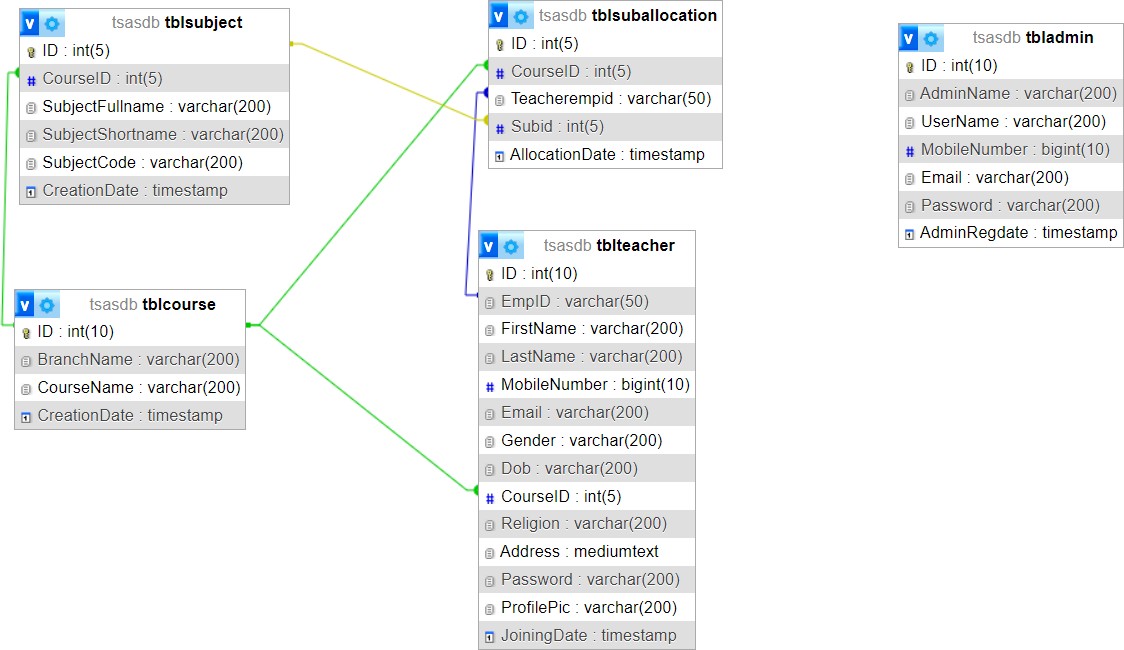
**tblsubject table structure :** This table store subject details.



**tblteacher table structure :** This table store details of teachers which is added by administrator.



**Database Schema**



**Data Flow Diagram**

Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

It shows how data enters and leaves the system, what changes the information, and where data is stored.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

The following observations about DFDs are essential:

1. All names should be unique. This makes it easier to refer to elements in the DFD.
2. Remember that DFD is not a flow chart. Arrows is a flow chart that represents the order of events; arrows in DFD represents flowing data. A DFD does not involve any order of events.
3. Suppress logical decisions. If we ever have the urge to draw a diamond- shaped box in a DFD, suppress that urge! A diamond-shaped box is used in flow charts to represents decision points with multiple exists paths of which the only one is taken. This implies an ordering of events, which makes no sense in a DFD.
4. Do not become bogged down with details. Defer error conditions and error handling until the end of the analysis.

**5.4 DATA FLOW DIAGRAM**

**Zero Level DFD**

**TSAS**

**Subject Management**

**Password Management**

**Changing Password Management**

**Teacher Management**

**Authorization Management**

**Admin Management**

**Course Management**

**Login Management**

**First Level DFD**

**Course**

**Management**

**TSAS**

**Admin**

**Management**

**Subject Management**

**Teacher Management**

**Password Management**

**Changing Password Management**

**Login Management**

**Authorization Management**

**Second Level DFD**

**Sign in**

**Forgot Password**

**Admin**

**Login to system**

**Check Roles of access**

**Manage Course (Add/Del/Update)**

**Check Credential**

**Manage Subject (Add/Del/Update)**

**Manage Teacher (Add/Del/Update)**

**Manage Module**

**Subject Allocation (Add/Del/)**



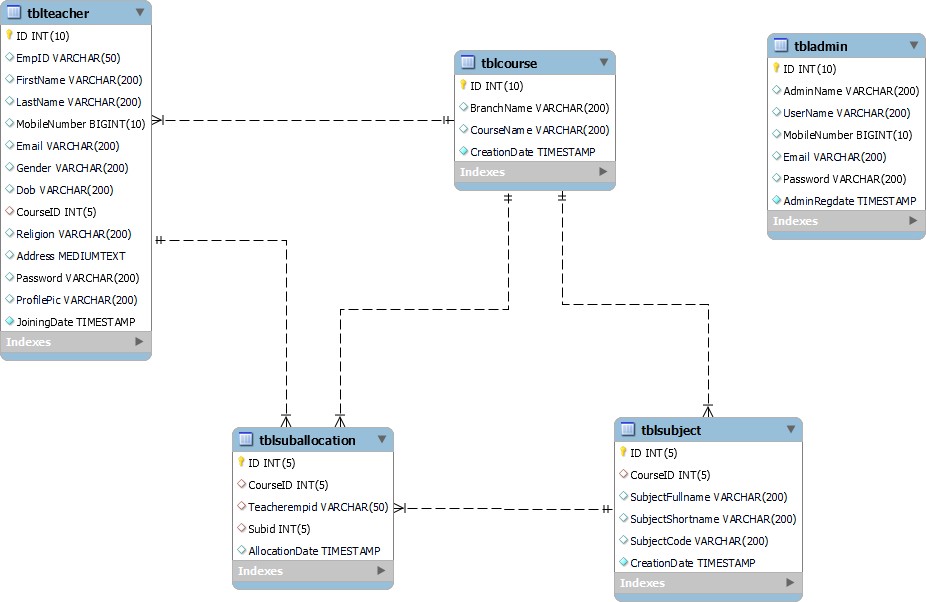
**Search Subject Allocation**

**Update Profile**

**Change Password**

**Class Diagram:**

The class diagram shows a set of classes, interfaces, collaborations and their relationships.



**CHAPTER 6**

**SYSTEM IMPLEMENTATION AND STRUCTURE**

**6.1 SYSTEM IMPLEMENTATION**

Implementing a system for teacher subject allocation involves several steps:

1.Requirement Analysis: Understand the requirements of the teacher subject allocation system, including the number of teachers, subjects, classes, and any constraints or preferences.

2.Database Design: Design a database to store information about teachers, subjects, classes, and allocation preferences.

3.User Interface Design: Design a user interface for teachers/administrators to input preferences and view allocations.

4.Algorithm Design: Develop algorithms to allocate teachers to subjects based on constraints such as teacher expertise, class schedules, and preferences.

5.Backend Development: Implement the backend logic to handle data processing, allocation algorithms, and database interactions.

6.Frontend Development: Develop the frontend application to interact with users, allowing them to input preferences and view allocations.

7. Testing: Test the system thoroughly to ensure it functions correctly and meets the requirements.

8. Deployment: Deploy the system on the desired platform, ensuring it is accessible to users and meets performance requirements.

9. Maintenance and Updates: Provide ongoing maintenance and updates to the system as needed, addressing any issues or adding new features.

### 6.2 SYSTEM TESTING

System testing is a series of different test whose primary purpose is to fully exercise computer based system.

We can say that it will run according to its specifications and in the way users expect. Special test data are input for processing, and the results examined. A limited number of users may be allowed to use the system so that analyst can see whether they try to use it in unforeseen ways. It is desirable to discover any surprises before the organization implements the system and depends on it.

* We follow Black Box testing.
* Black box testing attempts to find errors in following
* Incorrect or missing function
* Interface errors
* Errors in data structure
* Initialization and termination errors

### CONCLUSION

### The currently implemented “Teacher Subject Allocation System” provides educational institute and teacher powerful features to handle subject allocation. The utilization of DBMS produces high system efficiency in data manipulation. Cross platform attributes of PHP and MySQL make it a portable system on most operating systems with slight modifications. In addition, the system security is strengthened by multiple security schemes. The database design is very important during implementation because the database structure can significantly affect system efficiency and flexibility. Currently, the database structure is constructed in a very flexible manner, so that new data attributes or items can be easily added to the system without changing current structure significantly. For further implementation, more features can be added to the system, such as the management of backup data or disaster recovery. More information could be explored according to users’ requirements.

### REFERENCE

For PHP https://[www.w3schools.com/php/default.asp](http://www.w3schools.com/php/default.asp) https://[www.sitepoint.com/php/](http://www.sitepoint.com/php/) https://[www.php.net/](http://www.php.net/) https://[www.killerphp.com/](http://www.killerphp.com/) <http://www.tutorialspoint.com/php/>

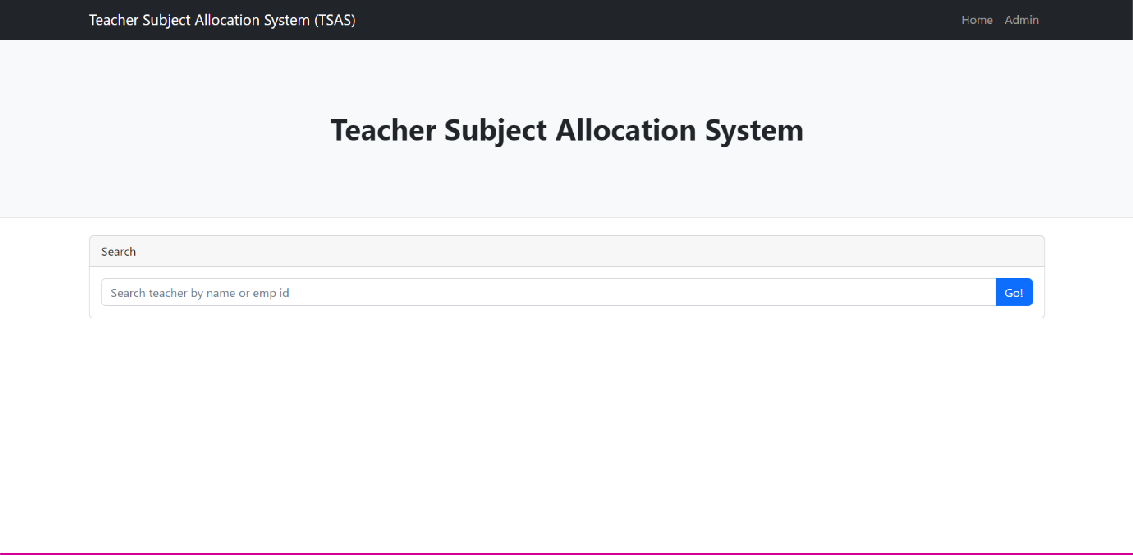
For MySQL https://[www.mysql.com/](http://www.mysql.com/) [http://www.mysqltutorial.org](http://www.mysqltutorial.org/)

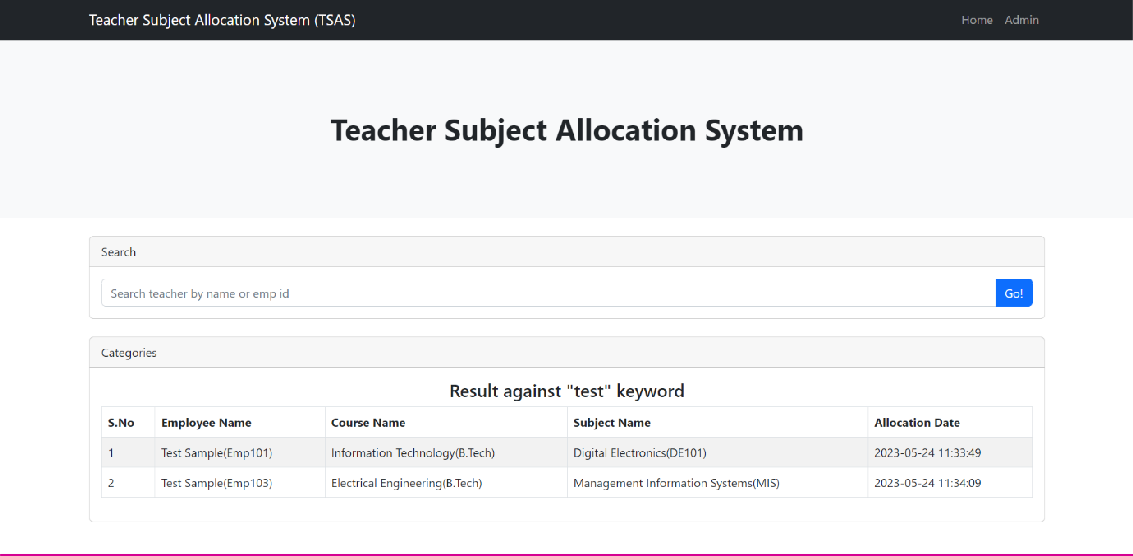
For XAMPP

https://[www.apachefriends.org/download.html](http://www.apachefriends.org/download.html)

### Output Screen of Project

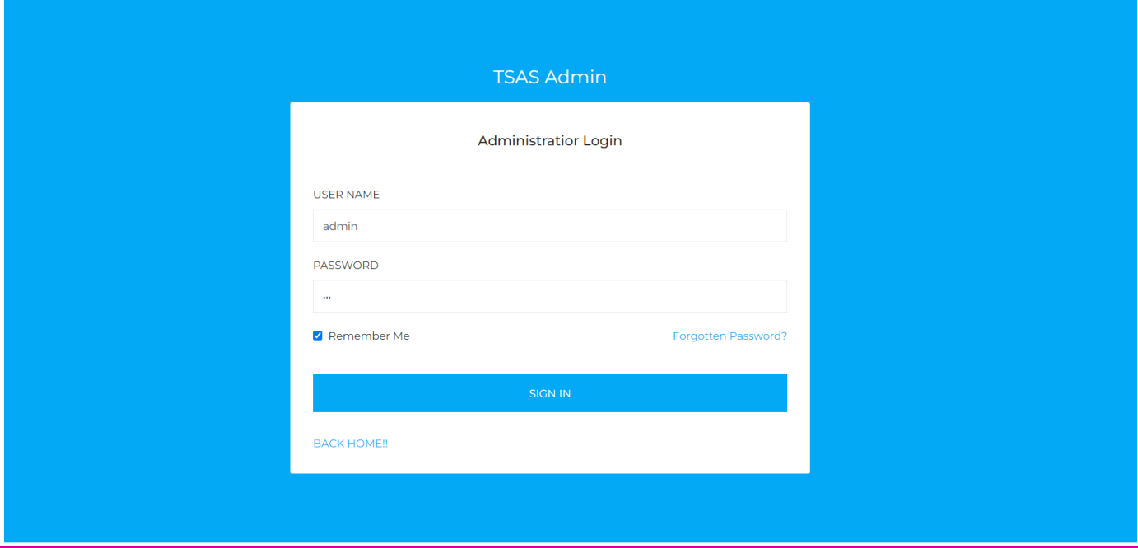
***Home Page***



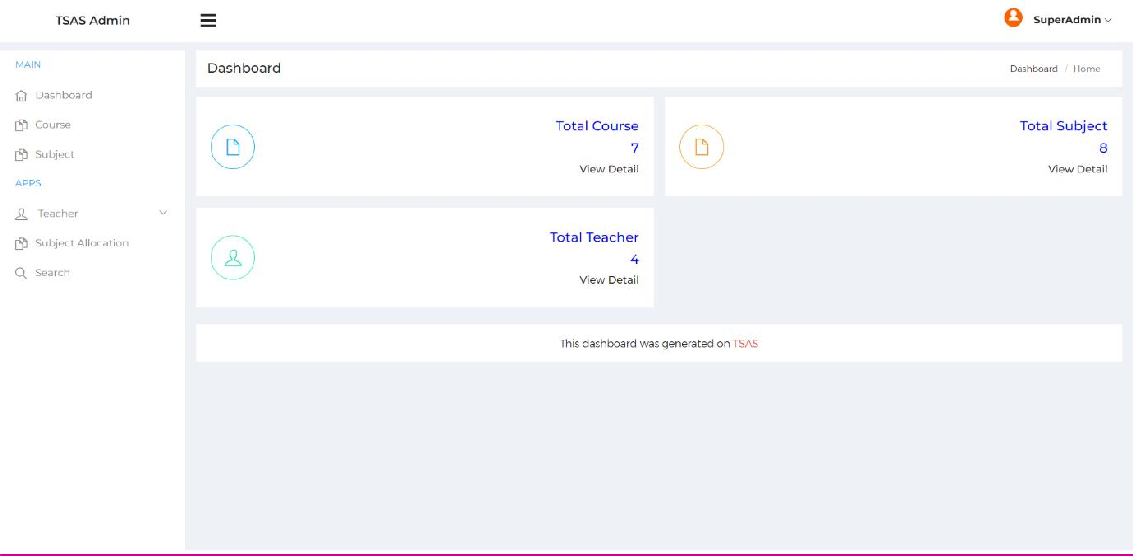
***Search Teacher***

### Admin Module

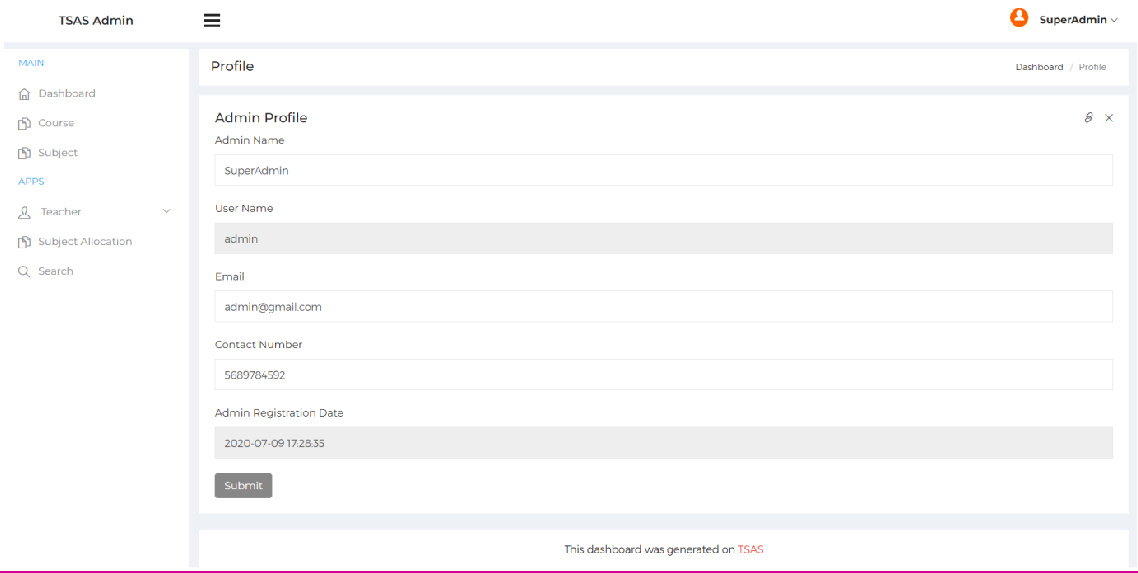
***Admin Login***

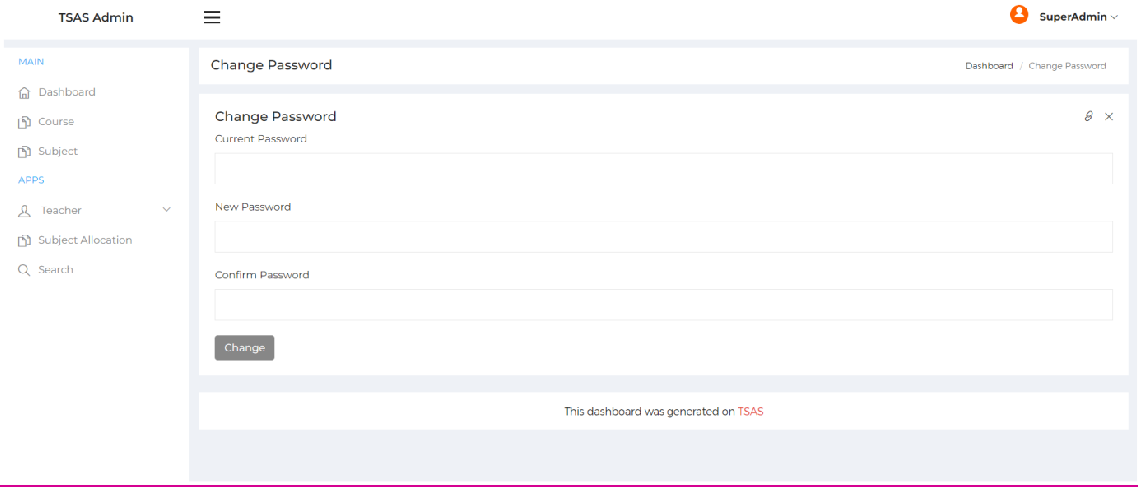


***Dashboard***

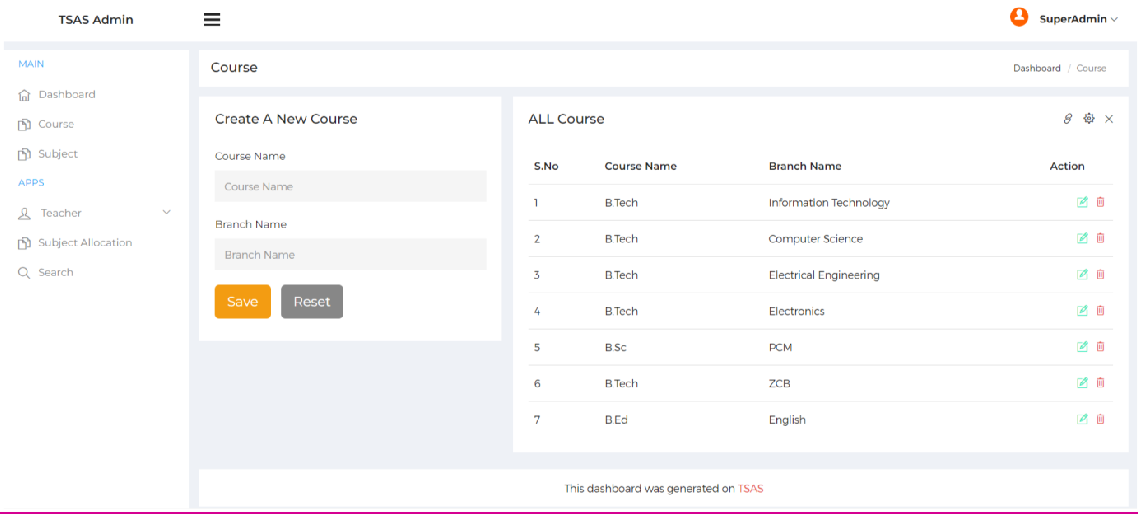


***Admin Profile***

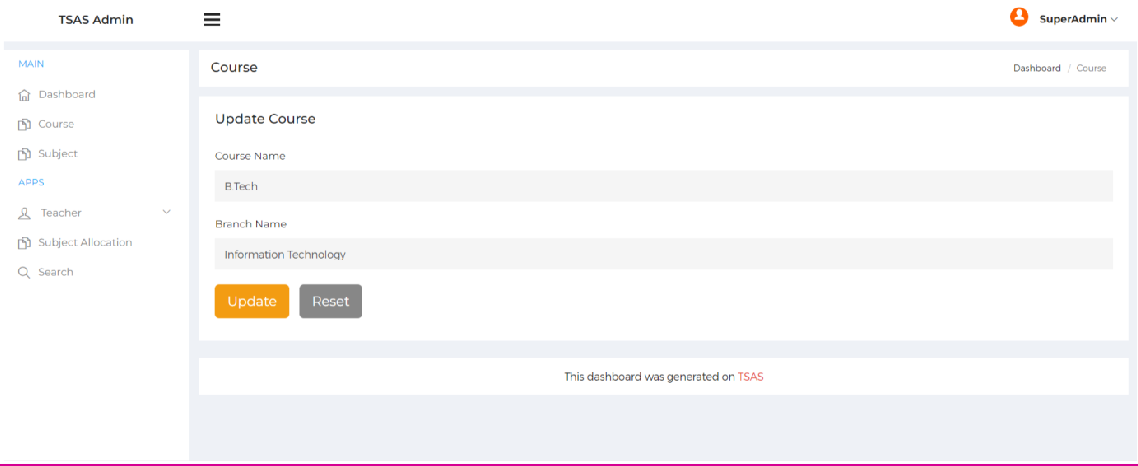


***Change Password***

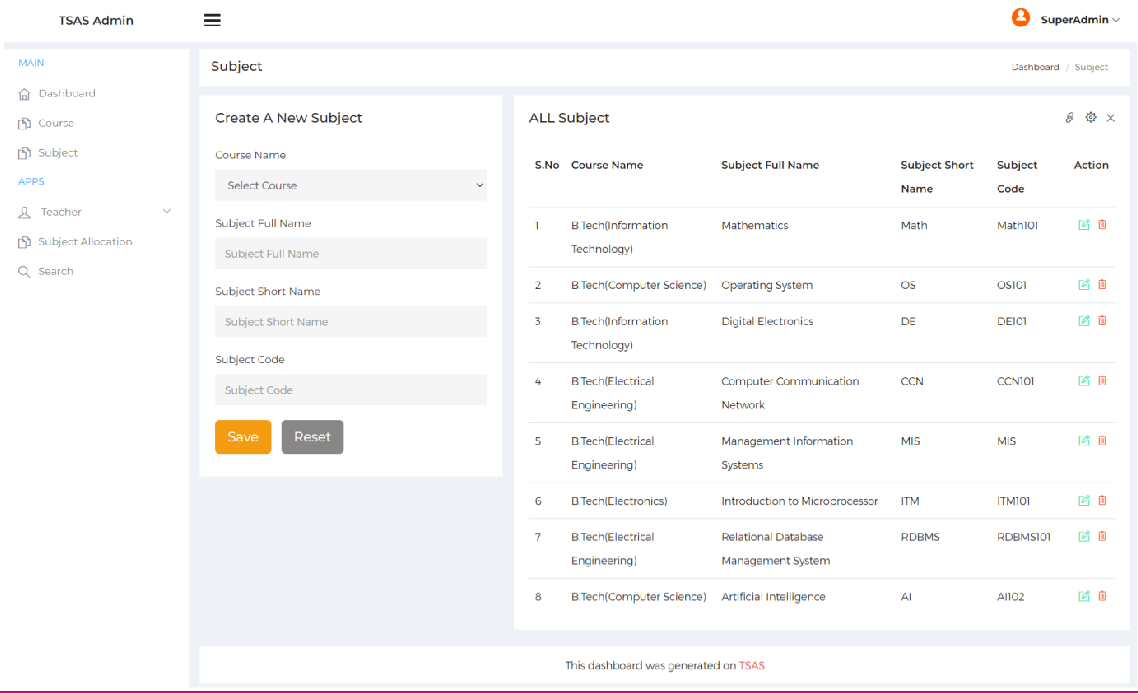
***Course***



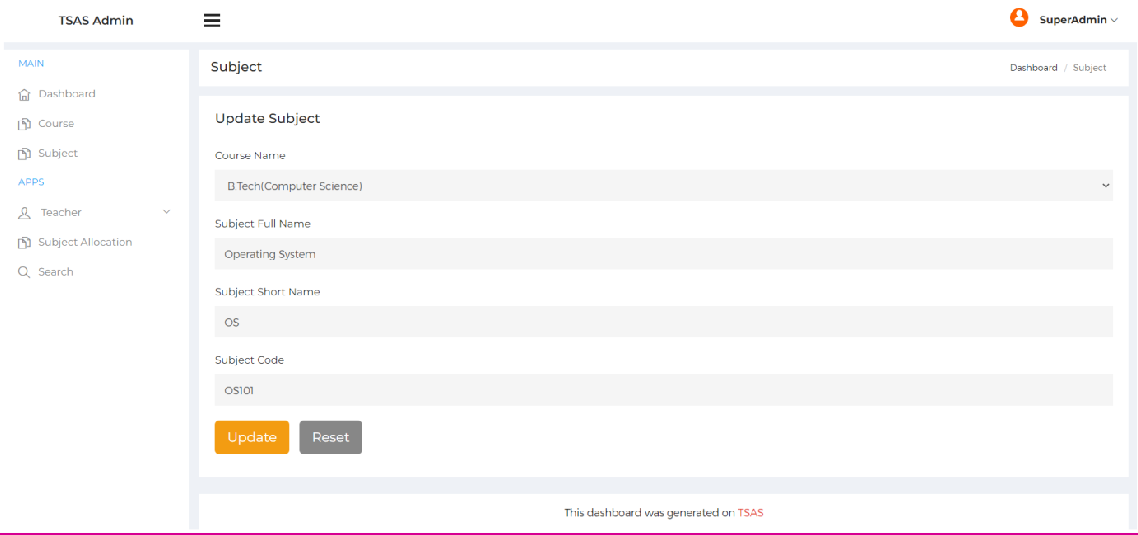
***Update Course***



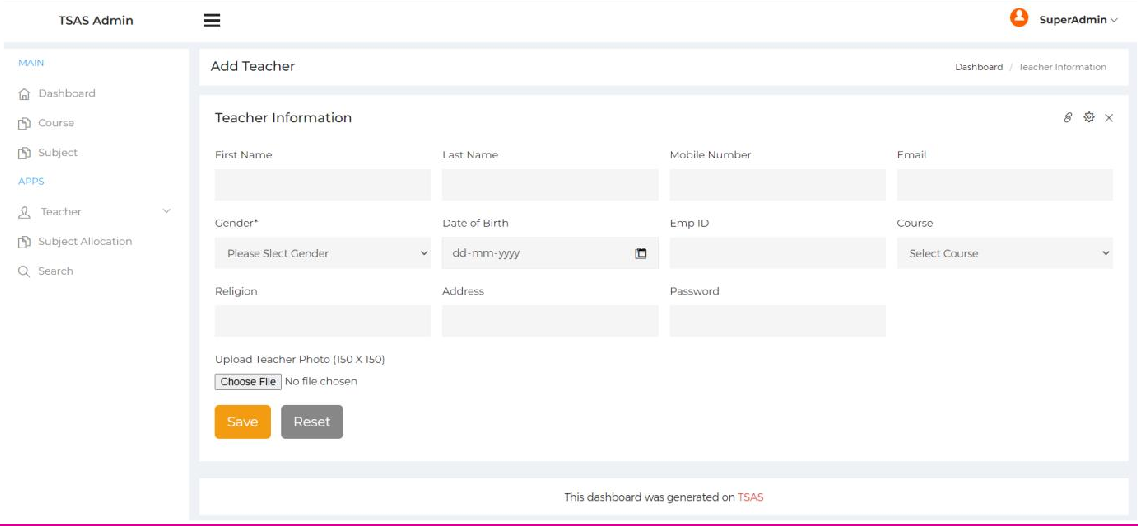
***Subject***



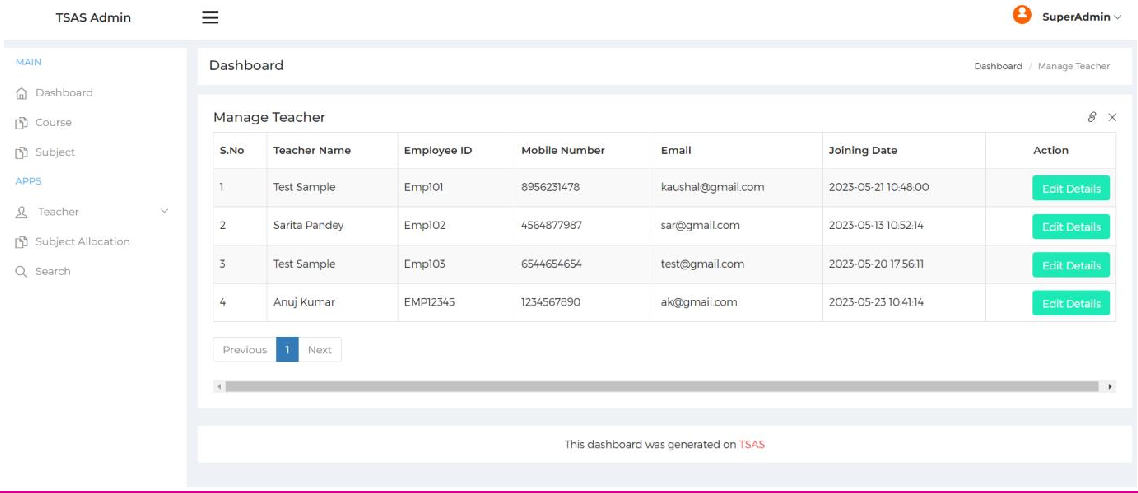
***Update Subject***



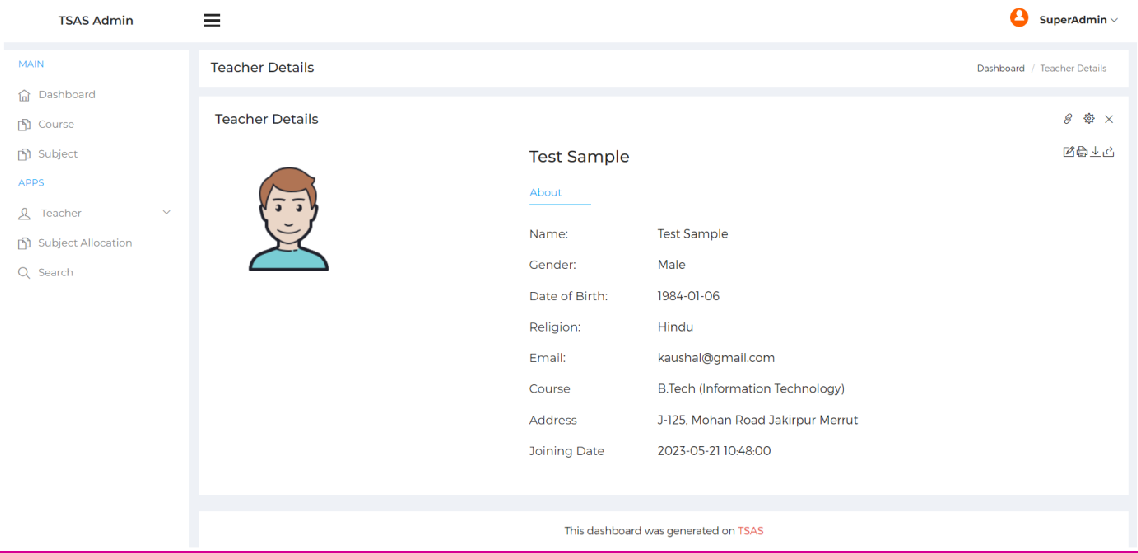
***Add Teacher***

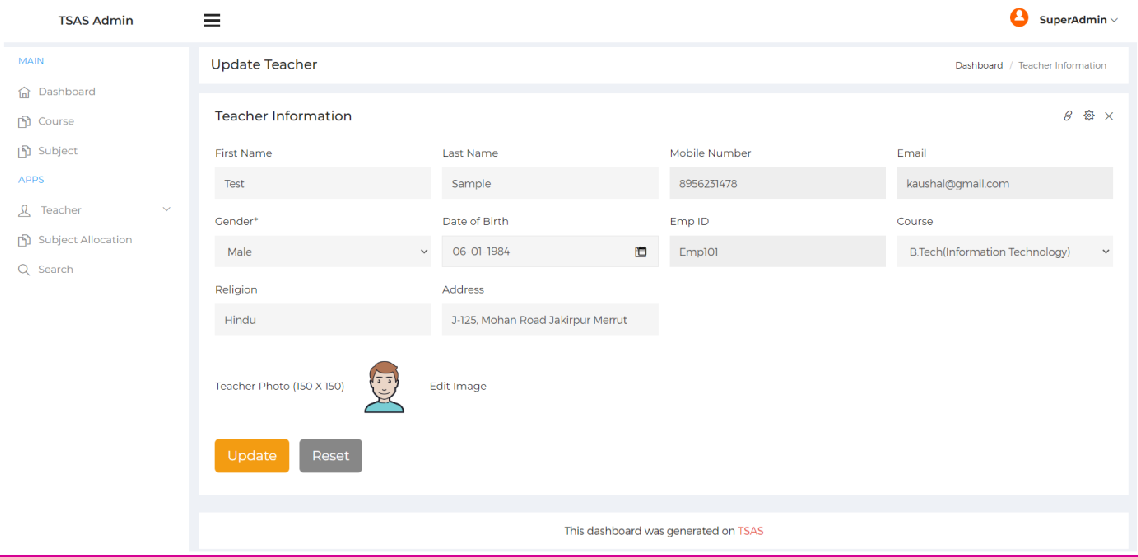


***Manage Teacher***

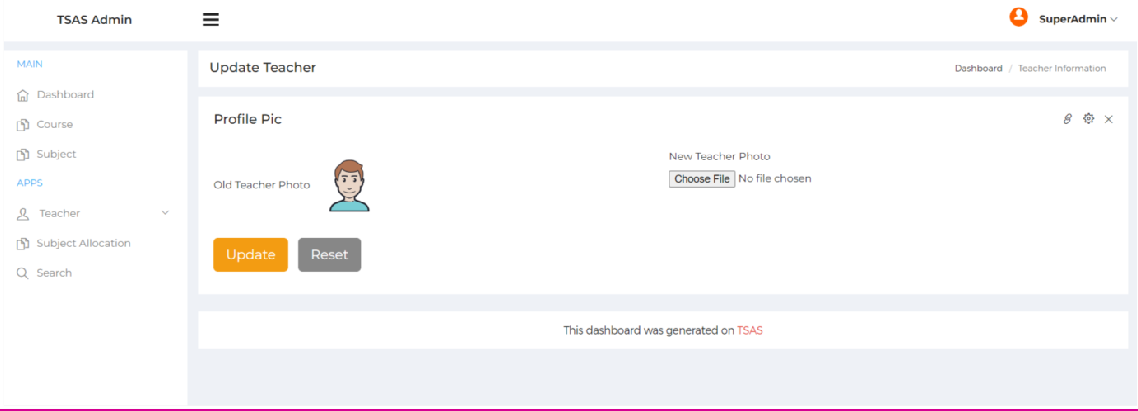


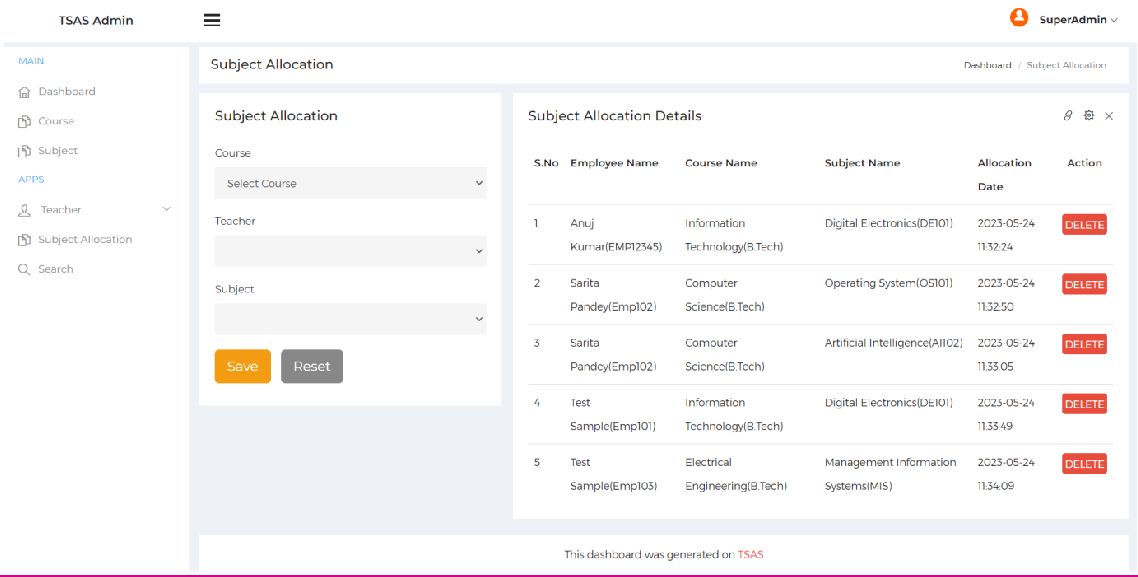
***View Teacher Detail***

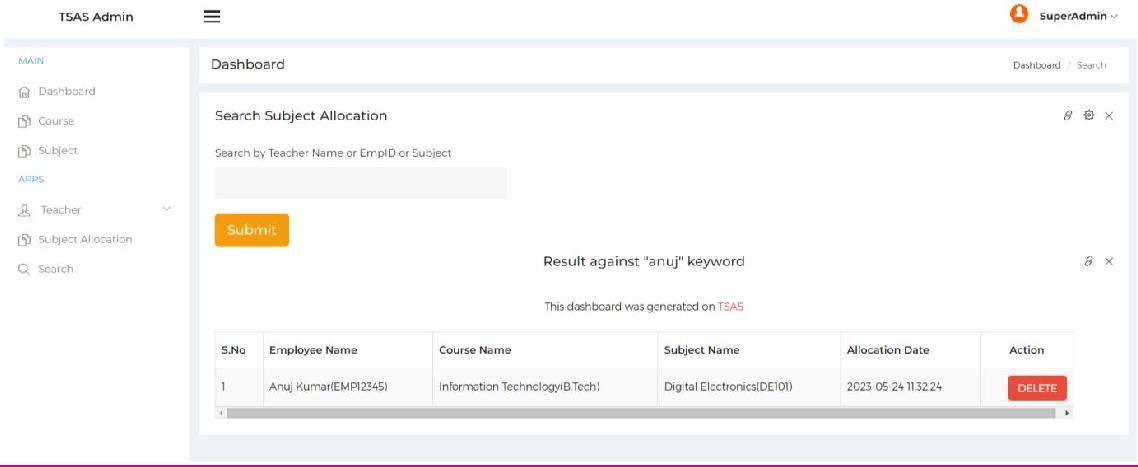


***Update Teacher Detail***

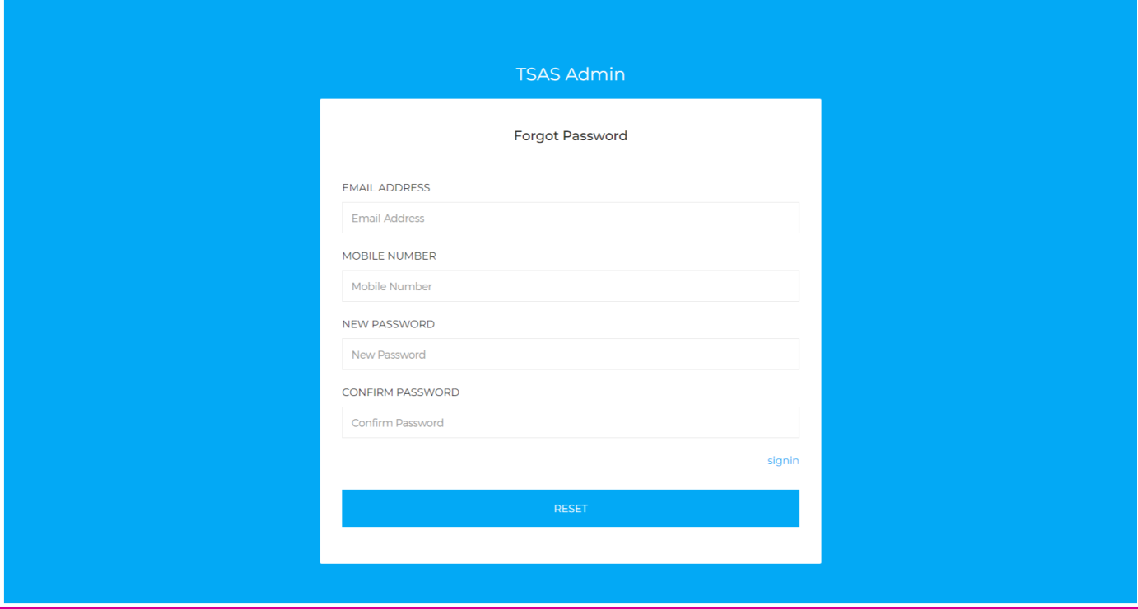
***Update Teacher Image***



***Subject Allocation***

***Search***

***Forgot Password***



**SOURCE CODE:**

**INDEX.PHP File**

<?php include('admin/includes/dbconnection.php'); ?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Teacher Subject Allocation System</title>

<!-- Favicon-->

<link rel="icon" type="image/x-icon" href="assets/favicon.ico" />

<!-- Core theme CSS (includes Bootstrap)-->

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

</head>

<body>

<!-- Responsive navbar-->

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

<div class="container">

<a class="navbar-brand" href="#!">Teacher Subject Allocation System (TSAS)</a>

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

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

<ul class="navbar-nav ms-auto mb-2 mb-lg-0">

<li class="nav-item"><a class="nav-link" href="index.php">Home</a></li>

<li class="nav-item"><a class="nav-link" href="admin/login.php">Admin</a></li>

</ul>

</div>

</div>

</nav>

<!-- Page header with logo and tagline-->

<header class="py-5 bg-light border-bottom mb-4">

<div class="container">

<div class="text-center my-5">

<h1 class="fw-bolder">Teacher Subject Allocation System</h1>

<!-- <p class="lead mb-0">A Bootstrap 5 starter layout for your next blog homepage</p> -->

</div>

</div>

</header>

<!-- Page content-->

<div class="container">

<div class="row">

<!-- Blog entries-->

<!-- Side widgets-->

<div class="col-lg-12">

<!-- Search widget-->

<div class="card mb-4">

<div class="card-header">Search</div>

<form method="post">

<div class="card-body">

<div class="input-group">

<input class="form-control" type="text" placeholder="Search teacher by name or emp id" aria-label="Enter search term..." aria-describedby="button-search" name="searchdata" />

<button class="btn btn-primary" id="button-search" name="search" type="submit" type="button">Go!</button>

</div>

</div></form>

</div>

<!-- Categories widget-->

<?php

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

{

$sdata=$\_POST['searchdata'];

?>

<div class="card mb-4">

<div class="card-header">Categories</div>

<div class="card-body">

<h4 align="center">Result against "<?php echo $sdata;?>" keyword </h4>

<table class="table table-striped table-bordered" border="1">

<thead>

<tr>

<th>S.No</th>

<th>Employee Name</th>

<th>Course Name</th>

<th>Subject Name</th>

<th>Allocation Date</th>

</tr>

</thead>

<tbody>

<?php

$sql="SELECT tblsuballocation.ID as suballid,tblsuballocation.CourseID,tblsuballocation.Teacherempid,tblsuballocation.Subid,tblsuballocation.AllocationDate,tblteacher.EmpID,tblteacher.FirstName,tblteacher.LastName,tblcourse.BranchName,tblcourse.CourseName,tblsubject.ID,tblsubject.CourseID,tblsubject.SubjectFullname,tblsubject.SubjectShortname,tblsubject.SubjectCode from tblsuballocation join tblteacher on tblteacher.EmpID=tblsuballocation.Teacherempid join tblcourse on tblcourse.ID=tblsuballocation.CourseID join tblsubject on tblsubject.ID=tblsuballocation.Subid where tblsuballocation.Teacherempid like '%$sdata%' || tblteacher.FirstName like '%$sdata%'";

$query = $dbh -> prepare($sql);

$query->execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query->rowCount() > 0)

{

foreach($results as $row)

{ ?>

<tr>

<td><?php echo htmlentities($cnt);?></td>

<td>

<?php echo htmlentities($row->FirstName);?> <?php echo htmlentities($row->LastName);?>(<?php echo htmlentities($row->Teacherempid);?>)

</td>

<td>

<?php echo htmlentities($row->BranchName);?>(<?php echo htmlentities($row->CourseName);?>)

</td>

<td>

<?php echo htmlentities($row->SubjectFullname);?>(<?php echo htmlentities($row->SubjectCode);?>)

</td>

<td>

<?php echo htmlentities($row->AllocationDate);?>

</td>

</tr>

<?php

$cnt=$cnt+1;

} } else { ?>

<tr>

<td colspan="9"> No record found against this search</td>

</tr>

</tbody>

</table>

</div>

</div>

<?php } }?>

<!-- Side widget-->

</div>

</div>

</div>

<!-- Footer-->

<!-- Bootstrap core JS-->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"></script>

<!-- Core theme JS-->

<script src="js/scripts.js"></script>

</body>

</html>

**DASHBOARD.PHP File**

<?php

session\_start();

error\_reporting(0);

include('includes/dbconnection.php');

if (strlen($\_SESSION['tsasaid']==0)) {

header('location:logout.php');

} else{

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>TSAS Admin : Dashboard</title>

<link href="../assets/css/lib/chartist/chartist.min.css" rel="stylesheet">

<!-- Styles -->

<link href="../assets/css/lib/owl.carousel.min.css" rel="stylesheet" />

<link href="../assets/css/lib/font-awesome.min.css" rel="stylesheet">

<link href="../assets/css/lib/themify-icons.css" rel="stylesheet">

<link href="../assets/css/lib/menubar/sidebar.css" rel="stylesheet">

<link href="../assets/css/lib/bootstrap.min.css" rel="stylesheet">

<link href="../assets/css/lib/unix.css" rel="stylesheet">

<link href="../assets/css/style.css" rel="stylesheet">

</head>

<body>

<?php include\_once('includes/sidebar.php');?>

<?php include\_once('includes/header.php');?>

<div class="content-wrap">

<div class="main">

<div class="container-fluid">

<div class="row">

<div class="col-lg-8 p-r-0 title-margin-right">

<div class="page-header">

<div class="page-title">

<h1>Dashboard</h1>

</div>

</div>

</div><!-- /# column -->

<div class="col-lg-4 p-l-0 title-margin-left">

<div class="page-header">

<div class="page-title">

<ol class="breadcrumb text-right">

<li><a href="#">Dashboard</a></li>

<li class="active">Home</li>

</ol>

</div>

</div>

</div><!-- /# column -->

</div><!-- /# row -->

<div id="main-content">

<div class="row">

<div class="col-md-6">

<div class="card">

<div class="media">

<div class="media-left meida media-middle">

<span><i class="ti-file f-s-22 color-primary border-primary round-widget"></i></span>

</div>

<div class="media-body media-text-right">

<?php

$sql1 ="SELECT \* from tblcourse";

$query1 = $dbh -> prepare($sql1);

$query1->execute();

$results1=$query1->fetchAll(PDO::FETCH\_OBJ);

$totcourse=$query1->rowCount();

?>

<h4 style="color: blue">Total Course</h4>

<h4 style="color: blue"><?php echo htmlentities($totcourse);?></h4>

<a href="course.php"><h5>View Detail</h5></a>

</div>

</div>

</div>

</div>

<div class="col-md-6">

<div class="card">

<div class="media">

<div class="media-left meida media-middle">

<span><i class="ti-file f-s-22 color-warning border-warning round-widget"></i></span>

</div>

<div class="media-body media-text-right">

<?php

$sql2 ="SELECT \* from tblsubject";

$query2 = $dbh -> prepare($sql2);

$query2->execute();

$results2=$query2->fetchAll(PDO::FETCH\_OBJ);

$totsub=$query2->rowCount();

?>

<h4 style="color: blue">Total Subject</h4>

<h4 style="color: blue"><?php echo htmlentities($totsub);?></h4>

<a href="subject.php"><h5>View Detail</h5></a>

</div>

</div>

</div>

</div>

<div class="col-md-6">

<div class="card">

<div class="media">

<div class="media-left meida media-middle">

<span><i class="ti-user f-s-22 color-success border-success round-widget"></i></span>

</div>

<div class="media-body media-text-right">

<?php

$sql3 ="SELECT \* from tblteacher";

$query3 = $dbh -> prepare($sql3);

$query3->execute();

$results3=$query3->fetchAll(PDO::FETCH\_OBJ);

$totteacher=$query3->rowCount();

?>

<h4 style="color: blue">Total Teacher</h4>

<h4 style="color: blue"><?php echo htmlentities($totteacher);?></h4>

<a href="manage-teacher.php"><h5>View Detail</h5></a>

</div>

</div>

</div>

</div>

</div>

<?php include\_once('includes/footer.php');?>

</div>

</div>

</div>

</div>

<!-- jquery vendor -->

<script src="../assets/js/lib/jquery.min.js"></script>

<!-- nano scroller -->

<script src="../assets/js/lib/jquery.nanoscroller.min.js"></script>

<!-- sidebar -->

<script src="../assets/js/lib/menubar/sidebar.js"></script>

<!-- bootstrap -->

<script src="../assets/js/lib/bootstrap.min.js"></script>

<!-- Circle Progress Bar -->

<script src="../assets/js/lib/circle-progress/circle-progress.min.js"></script>

<script src="../assets/js/lib/circle-progress/circle-progress-init.js"></script>

<script src="../assets/js/lib/chartist/chartist.min.js"></script>

<script src="../assets/js/lib/chartist/chartist-init.js"></script>

<script src="../assets/js/lib/sparklinechart/jquery.sparkline.min.js"></script>

<script src="../assets/js/lib/sparklinechart/sparkline.init.js"></script>

<!-- Bar Chat Js -->

<script src="../assets/js/lib/peitychart/jquery.peity.min.js"></script><!-- scripit init-->

<script src="../assets/js/lib/peitychart/peitychart.init.js"></script><!-- scripit init-->

<script src="../assets/js/lib/datamap/d3.min.js"></script>

<script src="../assets/js/lib/datamap/topojson.js"></script>

<script src="../assets/js/lib/datamap/datamaps.world.min.js"></script>

<script src="../assets/js/lib/datamap/datamap-init.js"></script>

<!-- scripit init-->

<script src="../assets/js/lib/owl-carousel/owl.carousel.min.js"></script>

<script src="../assets/js/lib/owl-carousel/owl.carousel-init.js"></script>

<script src="../assets/js/lib/morris-chart/raphael-min.js"></script>

<script src="../assets/js/lib/morris-chart/morris.js"></script>

<script src="../assets/js/lib/morris-chart/morris-init.js"></script>

<script src="../assets/js/scripts.js"></script>

</body>

</html><?php } ?>

**PROFILE.PHP CODE File**

<?php

session\_start();

error\_reporting(0);

include('includes/dbconnection.php');

if (strlen($\_SESSION['tsasaid']==0)) {

header('location:logout.php');

} else{

if(isset($\_POST['submit']))

{

$adminid=$\_SESSION['tsasaid'];

$AName=$\_POST['adminname'];

$mobno=$\_POST['mobilenumber'];

$email=$\_POST['email'];

$sql="update tbladmin set AdminName=:adminname,MobileNumber=:mobilenumber,Email=:email where ID=:aid";

$query = $dbh->prepare($sql);

$query->bindParam(':adminname',$AName,PDO::PARAM\_STR);

$query->bindParam(':email',$email,PDO::PARAM\_STR);

$query->bindParam(':mobilenumber',$mobno,PDO::PARAM\_STR);

$query->bindParam(':aid',$adminid,PDO::PARAM\_STR);

$query->execute();

echo '<script>alert("Profile has been updated")</script>';

echo "<script>window.location.href ='profile.php'</script>";

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>TSAS Admin : Profile </title>

<!-- Styles -->

<link href="../assets/css/lib/font-awesome.min.css" rel="stylesheet">

<link href="../assets/css/lib/themify-icons.css" rel="stylesheet">

<link href="../assets/css/lib/menubar/sidebar.css" rel="stylesheet">

<link href="../assets/css/lib/bootstrap.min.css" rel="stylesheet">

<link href="../assets/css/lib/unix.css" rel="stylesheet">

<link href="../assets/css/style.css" rel="stylesheet">

</head>

<body>

<?php include\_once('includes/sidebar.php');?>

<?php include\_once('includes/header.php');?>

<div class="content-wrap">

<div class="main">

<div class="container-fluid">

<div class="row">

<div class="col-lg-8 p-r-0 title-margin-right">

<div class="page-header">

<div class="page-title">

<h1>Profile</h1>

</div>

</div>

</div>

<!-- /# column -->

<div class="col-lg-4 p-l-0 title-margin-left">

<div class="page-header">

<div class="page-title">

<ol class="breadcrumb text-right">

<li><a href="dashboard.php">Dashboard</a></li>

<li class="active">Profile</li>

</ol>

</div>

</div>

</div>

<!-- /# column -->

</div>

<!-- /# row -->

<div id="main-content">

<div class="row">

<div class="col-lg-12">

<div class="card alert">

<div class="card-header">

<h4>Admin Profile</h4>

</div>

<div class="card-body">

<div class="basic-form">

<form method="post">

<?php

$sql="SELECT \* from tbladmin";

$query = $dbh -> prepare($sql);

$query->execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query->rowCount() > 0)

{

foreach($results as $row)

{ ?>

<div class="form-group">

<label>Admin Name</label>

<input type="text" class="form-control" name="adminname" value="<?php echo $row->AdminName;?>" required='true'>

</div>

<div class="form-group">

<label>User Name</label>

<input type="text" class="form-control" name="username" value="<?php echo $row->UserName;?>" readonly="true">

</div>

<div class="form-group">

<label>Email</label>

<input type="email" class="form-control" name="email" value="<?php echo $row->Email;?>" required='true'>

</div>

<div class="form-group">

<label>Contact Number</label>

<input type="text" class="form-control" name="mobilenumber" value="<?php echo $row->MobileNumber;?>" required='true' maxlength='10'>

</div>

<div class="form-group">

<label>Admin Registration Date</label>

<input type="text" class="form-control" id="email2" name="" value="<?php echo $row->AdminRegdate;?>" readonly="true">

</div>

<?php $cnt=$cnt+1;}} ?>

<button type="submit" class="btn btn-default" name="submit">Submit</button>

</form>

</div>

</div>

</div>

</div>

</div>

<?php include\_once('includes/footer.php');?>

</div>

</div>

</div>

</div>

<!-- jquery vendor -->

<script src="../assets/js/lib/jquery.min.js"></script>

<script src="../assets/js/lib/jquery.nanoscroller.min.js"></script>

<!-- nano scroller -->

<script src="../assets/js/lib/menubar/sidebar.js"></script>

<script src="../assets/js/lib/preloader/pace.min.js"></script>

<!-- sidebar -->

<script src="../assets/js/lib/bootstrap.min.js"></script>

<script src="../assets/js/scripts.js"></script>

</body>

</html><?php } ?>