STUDENTS ID CARD GENERATOR SYSTEM

A PROJECT REPORT
Submitted by

A.SEBASTIN NISHANTH

(0121128046)

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

of

BACHELOR OF SCIENCE
IN
COMPUTER SCIENCE

Under the Guidance of

DR. S. MURUGAN MCA., M.Phil., Ph.D

DEPARTMENT OF COMPUTER SCIENCE

ALAGAPPA GOVERNMENT ARTS COLLEGE

(Grade-I college and Re-accredited with “B” Grade by NAAC)

KARAIKUDI - 630 003.

(APRIL 2024)

DEPARTMENT OF COMPUTER SCIENCE

ALAGAPPA GOVERNMENT ARTS COLLEGE

(Grade-I college and Re-accredited with "B" Grade by NAAC)

KARAIKUDI - 630 003.

BONAFIDE CERTIFICATE

Certified that this project report titled **"STUDENTS ID CARD GENERATOR SYSTEM"** is the bonafide work of **A.SEBASTIN NISHANTH (Reg.No: 0121128046)** who carried out the project under my supervision and submitted during the academic year 2023-2024.

The Viva-voce held on ……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

HEAD OF THE DEPARTMENT

ACKNOWLEDGEMENT

I thank God, our almighty for his mercy to his towards me in concluding this project in a very successful way.

I wish to extend my thanks to our principal **Dr. A. PETHALAKSHMI M.Sc., MPhil., Ph.D.,** Alagappa Government Arts College, Karaikudi who has provided all facilities for the successful completion.

I wish to thank my head of the department **Dr. K.C. CHANDRASEKARAN M.Sc., M.Phil., PGDCA., Ph.D ,** Associate Professor, Alagappa Government Arts College, Karaikudi for approving and allowing me to carry out this project work.

I express my sincere thanks to my respected guide **Dr. S. MURUGAN MCA., M.Phil., Ph.D.,** Associate Professor,Department of computer Science, Alagappa Government Arts College, Karaikudi for valuable suggestion and encouragement given throughout the project.

I would also like to thank all the staff members and lab assistant of the department of the Computer Science department for their help rendered during this project.

I wish to express my sincere thanks to my parents for their constant encouragement shown to me in path of success.

Finally, I wish to thank my friends for the constant encouragement and valuable suggestions throughout the course of study.

A.SEBASTIN NISHANTH

(0121128046)

ABSTRACT

The aim of this project is to develop a Student ID Card Generation System to streamline the process of creating and managing data. This system will not only enhance efficiency but also provide a seamless solution for the generation and management of student ID cards, simplifying administrative tasks for educational establishments. By implementing this technology, the project seeks to optimize the overall identification process and ensure accurate record-keeping.

On the management side, it takes a considerable amount of time to issue ID cards to students. By the time they distribute the ID cards, the first semester has passed. Additionally, some students face difficulties in receiving their ID cards due to the overwhelming workload of the staff. In some cases, incorrect details are entered and printed. As a result, students are required to reapply for their ID cards.

In my project, assigning the management as administrators and staff and students as users proves to be beneficial. The management can create, view, and delete users, as well as access student data. Next, staff members should collect email addresses from students and store them in the database. Afterward, they can generate ID cards and save them for printing. Subsequently, students must log in to the website using their email addresses with OTP verification. Then, they need to fill out the visible form with their details and submit it. The system should maintain the data for future reference.

This system enables students to receive their ID cards promptly, avoiding issues such as missing or incorrect details. Additionally, the data of every student is securely maintained.

TABLE OF CONTENTS

INTRODUCTION

1. INTRODUCTION

1.1 PROBLEM STATEMENT

Our students are facing problems in obtaining ID cards from the management. On the management side, it takes a considerable amount of time to issue ID cards to students. By the time they distribute the ID cards, the first semester has passed. Additionally, some students face difficulties in receiving their ID cards due to the overwhelming workload of the staff. In some cases, incorrect details are entered and printed. As a result, students are required to reapply for their ID cards.

1.2 EXISTING SYSTEM

The staff from the studio arrived to take photos and collect data from students to create ID cards. However, on the scheduled day for this task, some students were absent, necessitating another day to gather data from those who were absent. Furthermore, there were instances where student details were wrongly entered and printed, resulting in additional time required to print and distribute the ID cards. As a consequence, by the time the ID cards were provided, half of the academic year for the students had already passed.

1.2.1 DRAWBACKS OF EXISTING SYSTEM

- Incorrect student details being entered.
- Some students unable to receive ID cards.
- Delay in providing student ID cards.
- Absenteeism causing data collection delays.
- Lack of safety for student details.

1.3 PROPOSED SYSTEM

The Proposed system is a complete assigning the management as administrators and staff and students as users proves to be beneficial. The management can create, view, and delete users, as well as access student data. Next, staff members should collect email addresses from students and store them in the database. Afterward, they can generate ID cards and save them for printing. Subsequently, students must log in to the website using their email addresses with OTP verification. Then, they need to fill out the visible form with their details and submit it. The system should maintain the data for future reference.

This system enables students to receive their ID cards promptly, avoiding issues such as missing or incorrect details. Additionally, the data of every student is securely maintained.

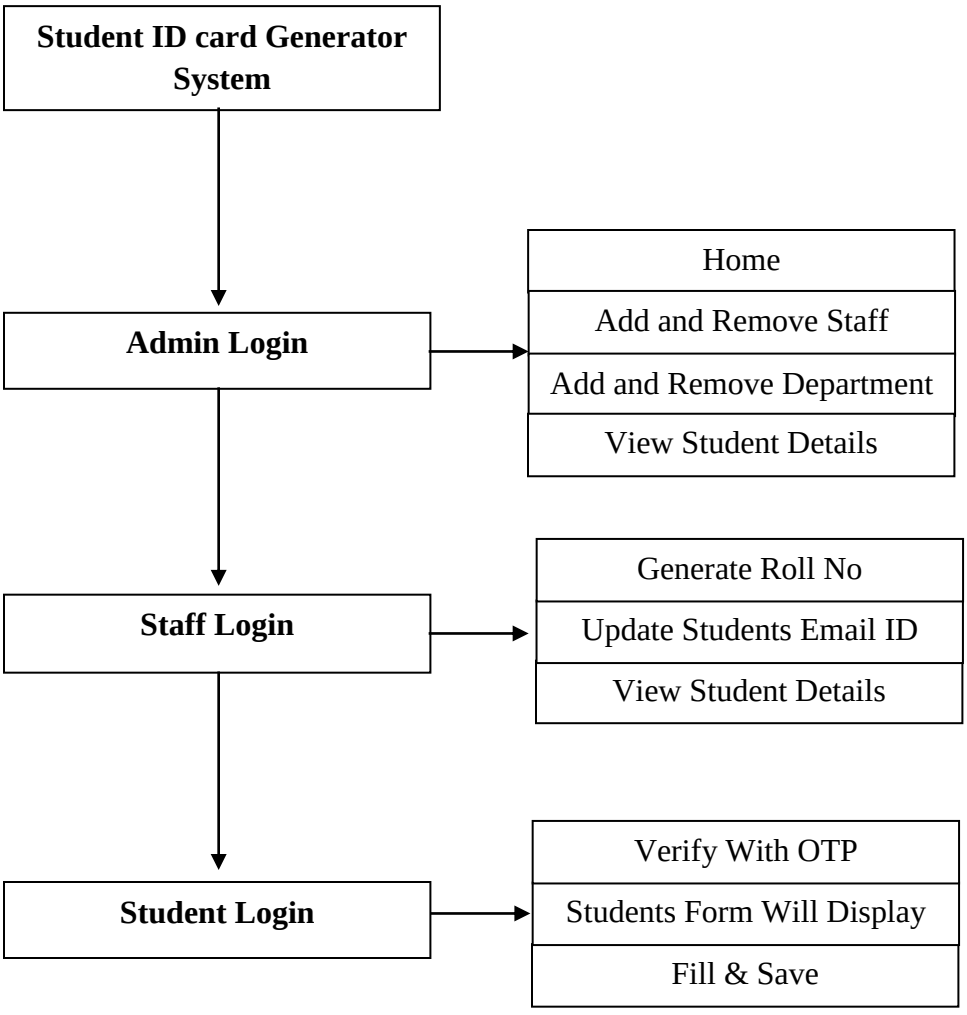
1.3.1 ADVANTAGES OF PROPOSED SYSTEM

- Efficient role assignment for management, staff, and students
- Secure authentication using email and OTP
- User-friendly form submission for students.
- Data security measures implemented
- Error prevention in student records.

SYSTEM DESIGN

2. SYSTEM DESIGN

2.1 ARCHITECTURAL DESIGN



2.2 DATABASE DESIGN

Database is an integrated collection of data and provides a centralized access to the data for the program send it makes possible to treat data as a separate resource. Usually centralized data managing software is called a relational database management system (RDBMS). The most significant different between RDBMS and other type of data management is the separation of data as seen by the program and data as store of on the direct access storage. This is the difference between logical and physical data.

2.2.1 Database Tables

The efficiency of an application developing using a RDBMS a mainly depend upon the database tables, the fields in each table and the way the tables are opened using the contents in them to retrieve the necessary information. A careful selection of tables and their fields are imperative. This project includes the following tables and the fields.

Table Name: adminuser

The adminuser database is for collect and store the username and password of the admins.

S.NO	FIELD NAME	DATATYPE	SIZE	DESCRIPTION
1	Username	Varchar	25	Username of the admin
2	Password	Varchar	25	Password of the admin

Table Name: staffuser

The staffuser database is for collect and store the username, password, category and department of the staffs.

S.NO	FIELD NAME	DATATYPE	SIZE	DESCRIPTION
1	Username	Varchar	25	Username of the staff
2	Password	Varchar	25	Password of the staff
3	Category	Varchar	2	Category of staff users (UG or PG)

4	Department	Varchar	50	Department of the staff users

Table Name: studata

The studata database is for collect and store the details of the every students according to their department.

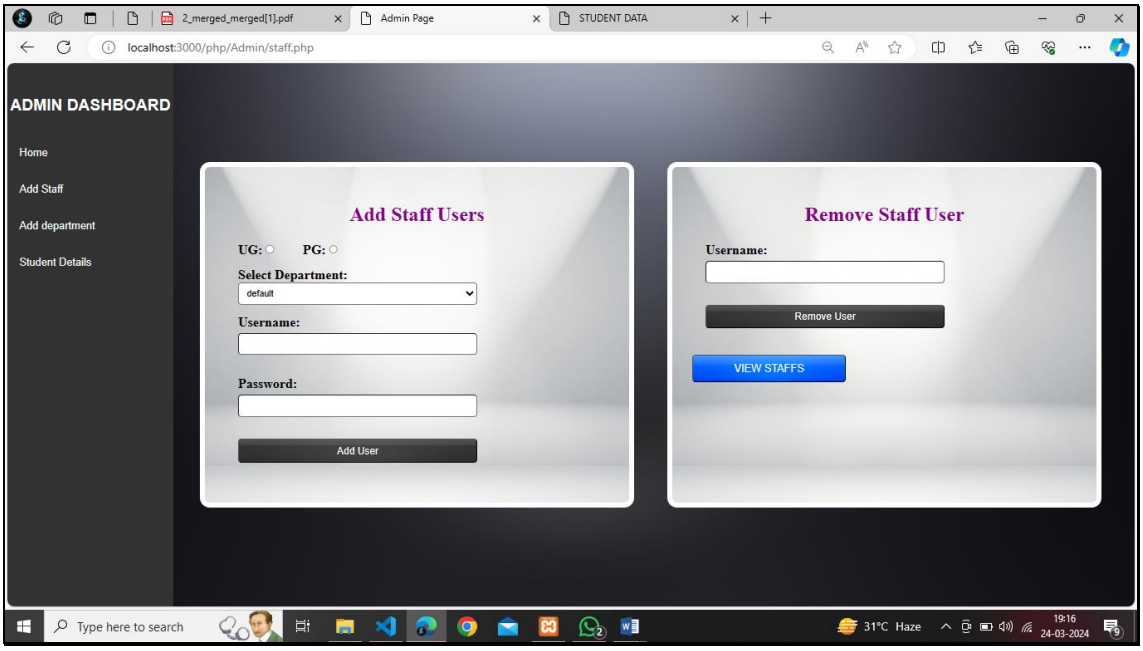
S.NO	FIELD NAME	DATATYPE	SIZE	DESCRIPTION
1	Image	Mediumblob	16mp	Image of the Students
2	Name	Varchar	50	Name of the Students
3	Email_id	Varchar	50	Email_id of the Students
4	Roll_no	Varchar	10	Roll_no of the Students
5	Course	Varchar	30	Course of the Students
6	Address	Varchar	250	Address of the Students
7	Mobile_no	Varchar	12	Mobile number of the Students
8	Dob	Date	-	Date of birth of the Students
9	Blood_group	Varchar	10	Blood group of the Students
10	Batch_year	Varchar	10	Academic year of the Students

2.3 FORM DESIGN

Input design is the process of converting user-oriented inputs to a computer based format. Input design is one of the most expensive phases of the operation of computerized system and is often the major problem of a system. In this project, the input design to make in various forms with various methods.

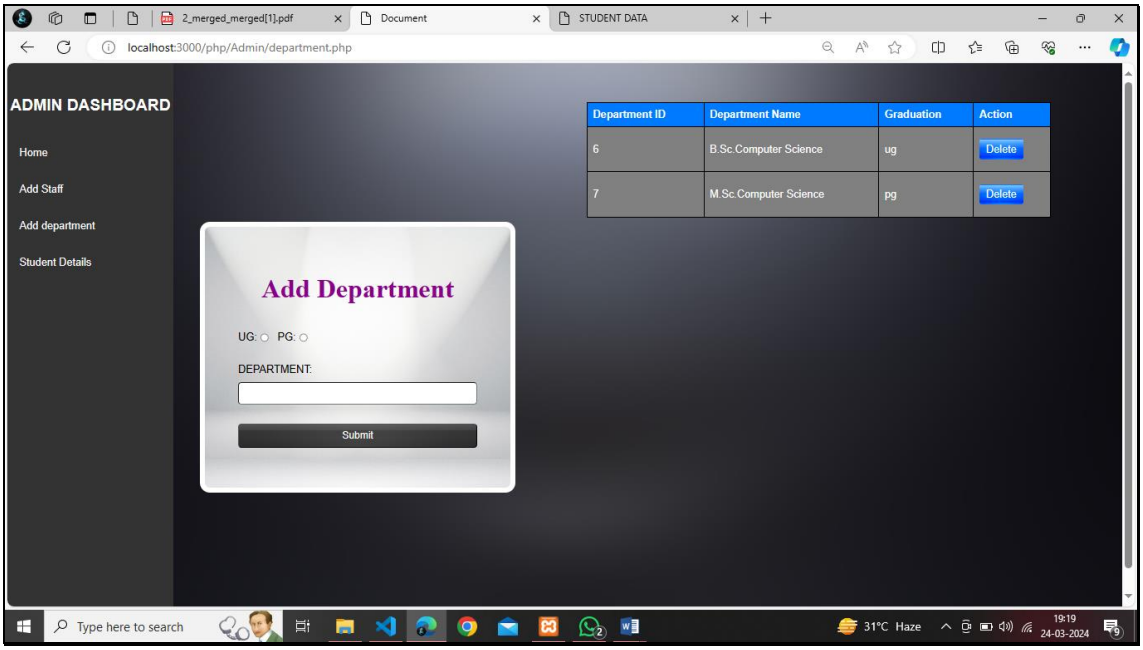
2.3.1 Add & Remove Staff page

The staff were added to facilitate student enrollment and maintain their details. This page includes input fields for graduation type, department selection, username, and password. Additionally, there is a remove staff user input field for username, along with a button to view students



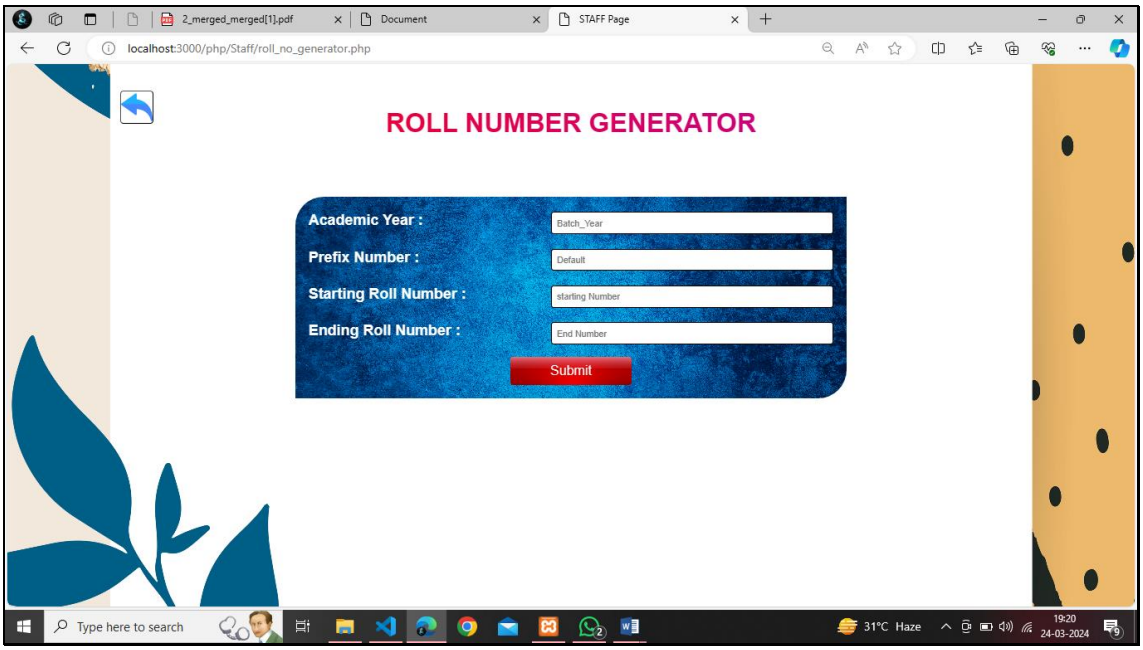
2.3.2 Add Department Page

The Add Department form is designed to input the graduation type and department. This form allows for the addition of departments to store user staff according to department preferences. Additionally, there is a delete button to remove departments from the database.



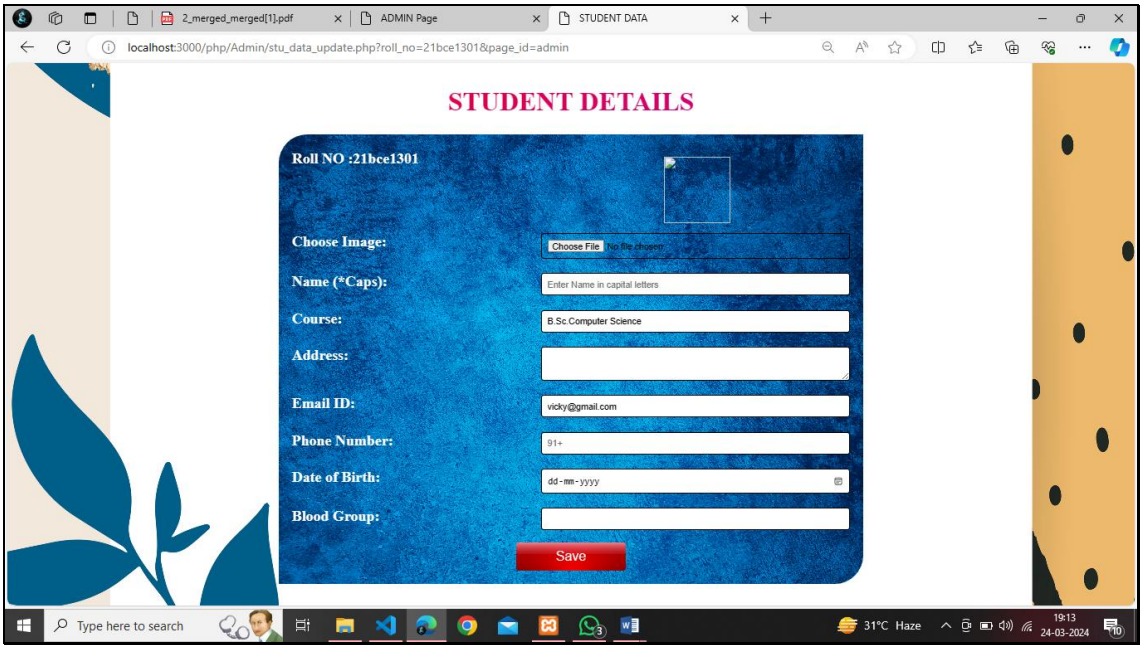
2.3.3 Generate Roll Numbers For Students Page

The Roll number generator form is designed for input the academic year, prefix number, starting roll number, and ending roll number. It creates a form for students according to their roll numbers. Additionally, there is a submit button to store the students' roll numbers.



2.3.4 Student Form Page

The Student form page is designed for inputting the chosen image, name, course, address, email ID, phone number, date of birth, and blood group. Students can fill out this form and save it for their ID cards.



2.4 MODULES EXPLANATION

The proposed model is implemented with the below modules:

1. **Admin**
2. **Staff**
3. **Students Form**

2.4.1 Admin Module

The Admin Module contains the adding staff module, staff were added to facilitate student enrollment and maintain their details. This page includes input fields for graduation type, department selection, username, and password. Additionally, there is a remove staff user input field for username, along with a button to view students , adding department module is to add department with the graduation type and department. This form allows for the addition of departments to store user staff according to department preferences. Additionally, there is a delete button to remove departments from the database and student details module for view the student details in database.

2.4.2 Staff Module

The Staff module contains the roll number generator module is designed for input the academic year, prefix number, starting roll number, and ending roll number. It creates a form for students according to their roll numbers, the update student email module is designed for add and store the emails of the students at database and view students details module is designed with the buttons of update, delete, and generate id for students.

2.4.3 Students Form Module

The Student form module contains the student details form for every students to fill up their details individually and save it for generate id cards for each students.

SYSTEM IMPLEMENTATION

3. SYSTEM IMPLEMENTATION

The Student ID Card Generation System aims to simplify administrative tasks within educational institutions by automating the process of creating and managing student ID cards. Through user authentication, administrators, staff, and students are granted specific access levels, allowing for efficient management of user data. Staff members collect student email addresses, generate ID cards, and save them for printing, while students input their details via a secure web interface. This system ensures prompt ID card issuance, minimizing errors and ensuring data security for all users.

This website implemented by the following procedures. The main link is listed as follows:

3.1 User Sign In
3.2 Admin Page
3.3 Staff Page
3.4 Students Page

3.1 User Sign in

3.1.1 Users input credentials
3.1.2 press "Login"
3.1.3 data is verified with the database.
3.1.4 If correct redirect to the respective page.
3.1.5 else, an error prompts correction.

3.2 Admin Page

3.2.1 To add a department, validate the input fields and add department details to the database.
3.2.2 To remove a department, validate the input field and remove the department from the database.
3.2.3 To add staffs, Validate the input fields and Add staff details to the database.
3.2.4 To remove staffs, Validate the input field and Remove staff from the database.
3.2.5 Implement functionality to view student details in the database.

3.3 Staff Page

3.3.1 Functionality to generate roll numbers.

3.3.2 Validate input fields, Generate roll numbers for students based on the provided parameters.

3.3.3 Functionality to update student emails.

3.3.4 Validate email input, Store student emails in the database.

3.3.5 Update student details, Allow modification of student information.

3.3.6 Delete student details, Remove student from the database.

3.3.7 Generate ID for students, Automatically assign a unique ID cards to each student.

3.4 Students Page

3.4.1 Define a function to create the student details form.

3.4.2 Implement functionality to save filled details for generating ID cards.

3.4.3 Implement form validation to ensure all required fields are filled.

3.4.4 If all required fields are filled and valid:

3.4.5 Store the student's details in the database.

3.4.6 If any required fields are missing or invalid:

3.4.7 Display error messages prompting the student to fill in the missing information or correct any errors.

SYSTEM TESTING

4. SYSTEM TESTING

4.1 TESTING

A “Program unit” stands for a routine or a collection of routines implemented by an individual programmer. It might even be a stand-alone program or a functional unit a larger program.

4.2 UNIT TESTING

All modules of this project are tested with sample data and all outputs are verified. Unit testing is performed prior to integration of the unit into a larger system. It is like coding and debugging -> unit testing -> integration testing A program unit must be tested for Functional tests, Performance tests, Stress tests and Structure tests.

Functional tests refer to executing the code with standard inputs, for which the results will occur within the expected boundaries. Performance test determines the execution time spent in various parts of the unit, response time, device utilization and throughput. Performance testing will help the tuning of the system.

4.3 SYSTEM TESTING

All modules of this project are tested with sample data and all outputs are verified. System testing involves two activities: Integration testing and Acceptance testing. Integration strategy stresses on the order in which modules are written, debugged and unit tested. Acceptance test involves functional tests, performance tests and stress tests to verify requirements fulfillment. System checking checks the interfaces, decision logic, control flow, recovery procedures, and throughput, capacity and timing characteristics of the entire system.

4.4 VALIDATION TESTING

Software testing and validation is achieved through a series of black box tests that demonstrate conformity with requirements. A test procedure defines specific test cases that will be used to demonstrate conformity with requirements. Both, the plan and the procedure are designed to ensure that all functional requirements are achieved,

documentation is correct and other requirements are met. After each validation test case has been conducted, one of the two possible conditions exists. They are,

- The function or performance characteristics conform to specification and are accepted.
- A deviation from specification is uncovered and a deficiency list is created.

The deviation or error discovered at this stage in project can rarely be corrected prior to scheduled completion. It is necessary to negotiate with the customer to establish a method for resolving deficiencies. The various level by level testing processes done in every page will be described as below:

4.4.1 USER LOGIN FORM

- Input verification testing done for registered User Account
- User input validation-testing process done for login form.
- User availability verification testing done for entered data.
- Unit testing process done for login form.

4.4.2 ADMIN PAGE

- Verify that all input fields (graduation type, department selection, username, password) are present and functional.
- Verify the presence and functionality of the input field for entering the username of the staff to be removed.
- Verify the presence of input fields for department name, graduation type, and department selection.
- Test the functionality of viewing student details, ensuring all student information is accurately displayed from the database.
- Data bind process and result verification test is initiated and done.

4.4.3 STAFF PAGE

- Validate the presence and functionality of input fields for academic year, prefix number, starting roll number, and ending roll number.
- Verify the presence and functionality of the form for adding and storing student emails in the database.
- Validate the presence and functionality of buttons for updating, deleting, and generating IDs for students.

4.4.4 STUDENT PAGE

- Verify that all necessary input fields are present and functional, including fields for name, date of birth, address, contact information, academic information, and any other relevant details.
- Test the functionality of saving filled details for generating ID cards.
- Test the integration with the ID card generation process, ensuring that student details are accurately used to generate unique ID cards for each student.

FUTURE ENHANCEMENT

5. FUTURE ENHANCEMENT

The website can be implemented with the following features

- Customizable ID Card Templates
- Integration with Student Information Systems (SIS)
- QR Code Integration
- Student Self-Service Portal

CONCLUSION

6. CONCLUSION

The Student ID Card Generation System presents a robust solution for streamlining administrative processes within educational institutions. By facilitating efficient creation and management of student ID cards, the system aims to enhance productivity and accuracy while reducing administrative burden. Through future enhancements such as biometric authentication, mobile application integration, and advanced reporting capabilities, the system can further optimize user experience and security measures. Overall, the system promises to provide a seamless and secure solution for student ID card management, ensuring timely issuance and accurate record-keeping for educational establishments

REFERENCE

REFERENCES

BOOK REFERENCES

1. "Web Development and Design Foundations with HTML5" by Terry Felke-Morris.
2. "PHP and MySQL Web Development" by Luke Welling and Laura Thomson.
3. "JavaScript: The Definitive Guide" by David Flanagan.
4. "Database Systems: The Complete Book" by Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom.

WEB REFERENCES

1. https://www.w3schools.com/
2. https://stackoverflow.com/
3. https://www.php.net/
4. https://www.codecademy.com/

APPENDIX

APPENDIX

A) Hardware Requirements

Processor	: Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, 1800 MHz, 4 Core(s), 8 Logical Processor(s)
Hard Disk Capacity	: 217GB
Ram	: 8 GB
Memory	: 1.60 GHz
Color Monitor	: 15.6 inches, LENOVO
Mouse	: Logitech
Keyboard	: 120 Keys

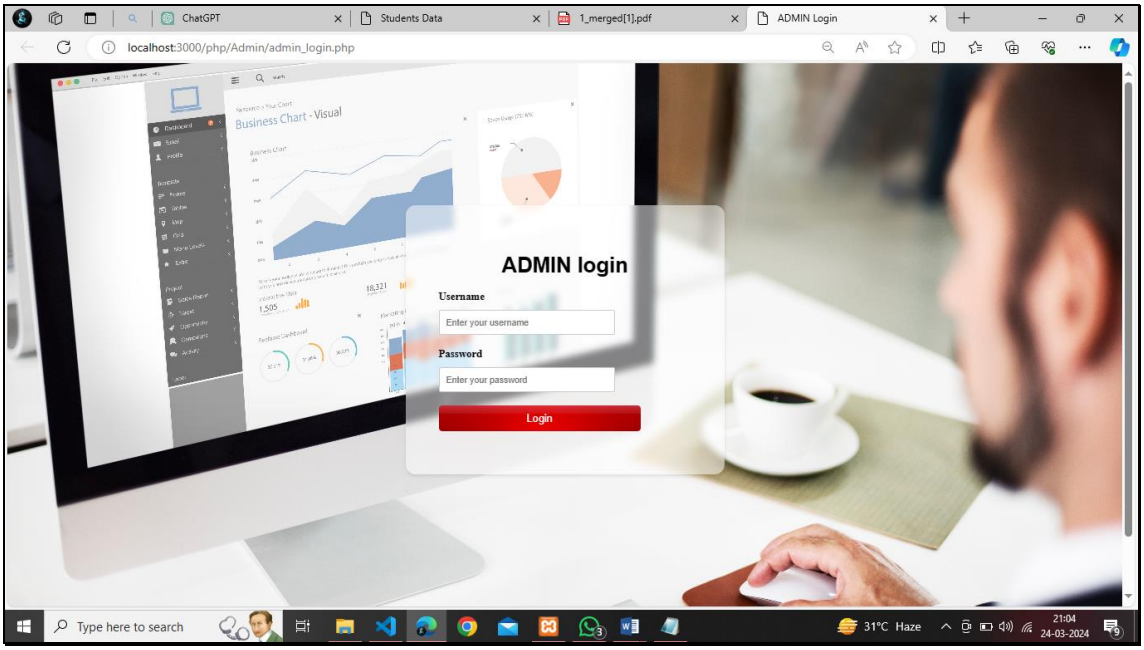
B) Software Requirement

Front end	: PHP
Back end	: MYSQL XAMPP Server
Operating System	: Microsoft Windows 10 pro

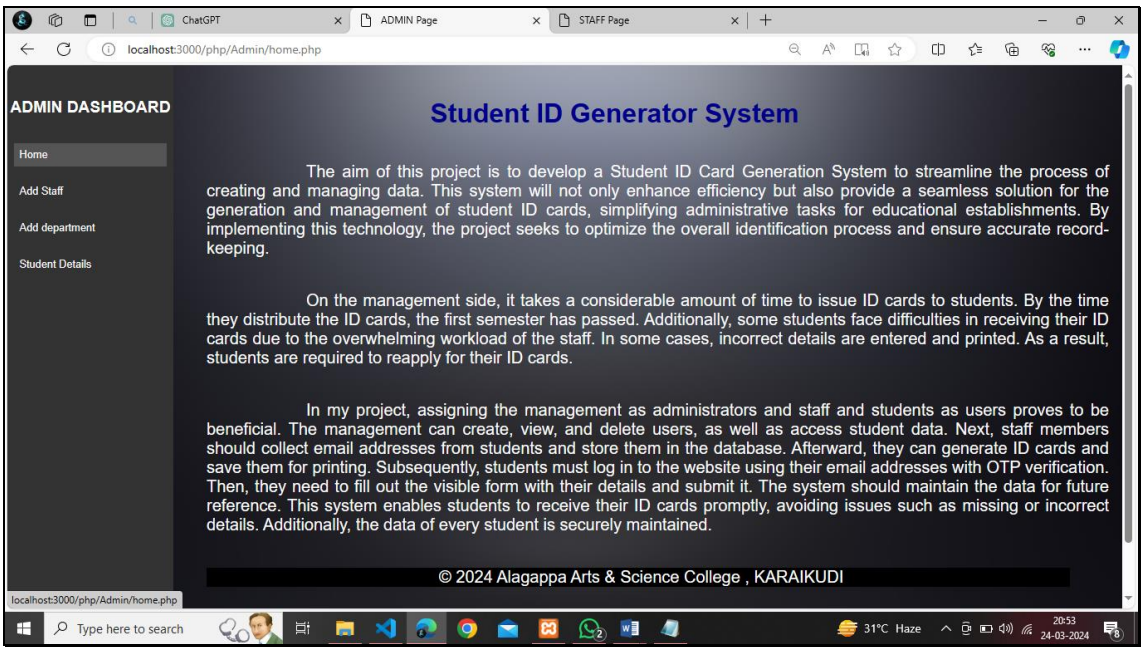
C) Sample Screen Shots

ADMIN PAGE

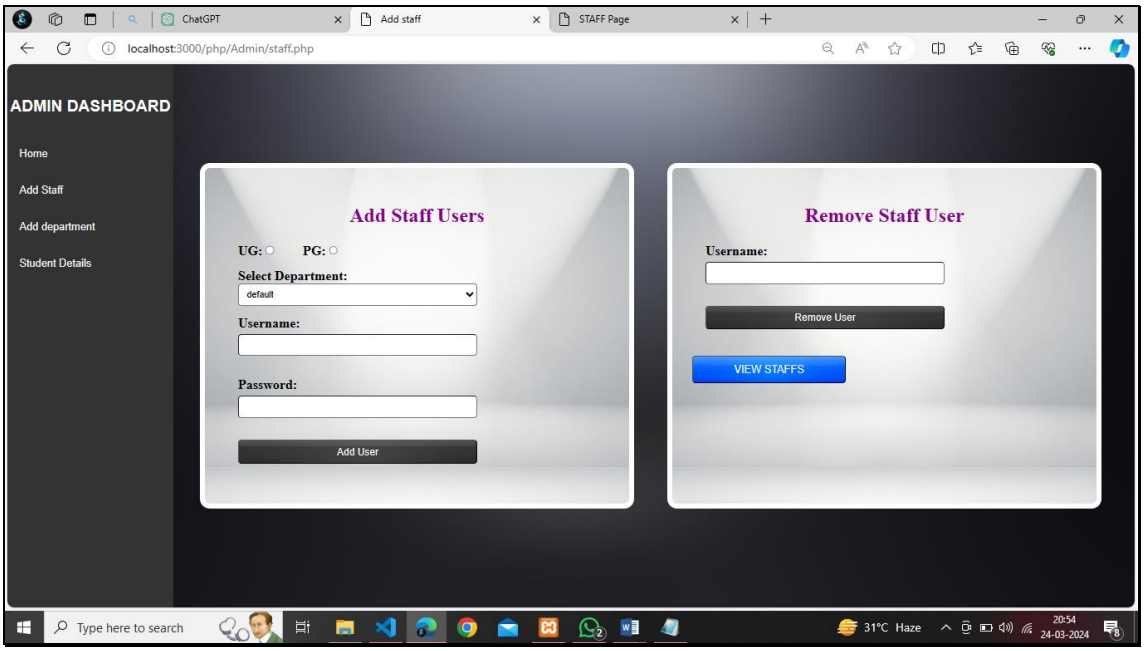
The following screenshot defines that it is the login page of admins.



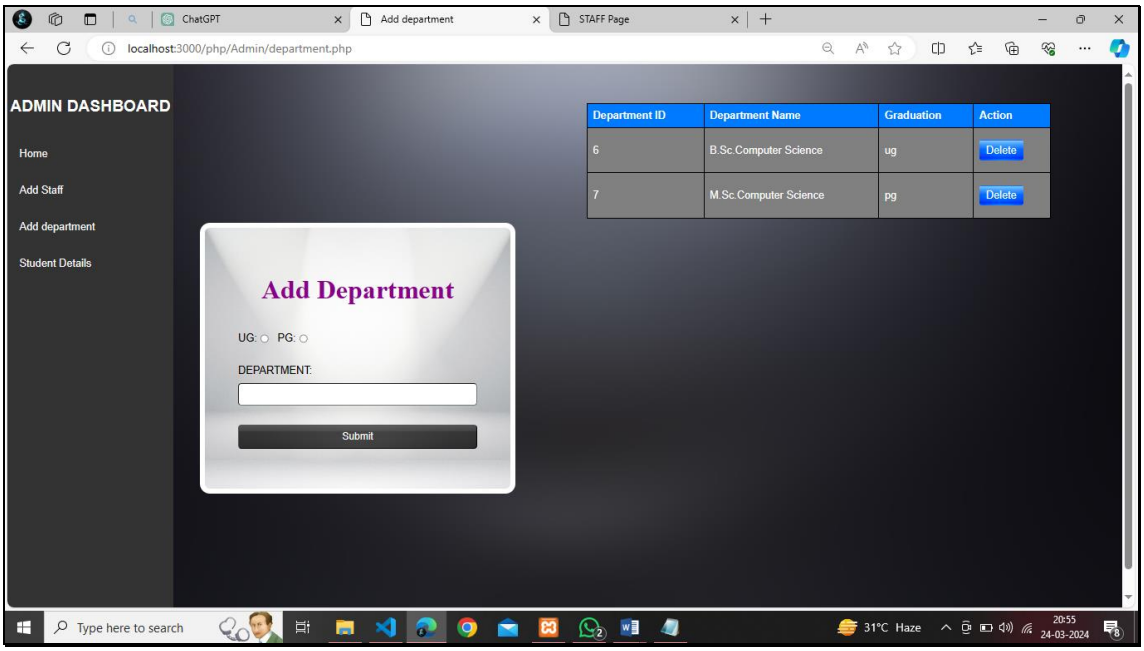
The following screenshot defines that it is the description of the project in admin's page.



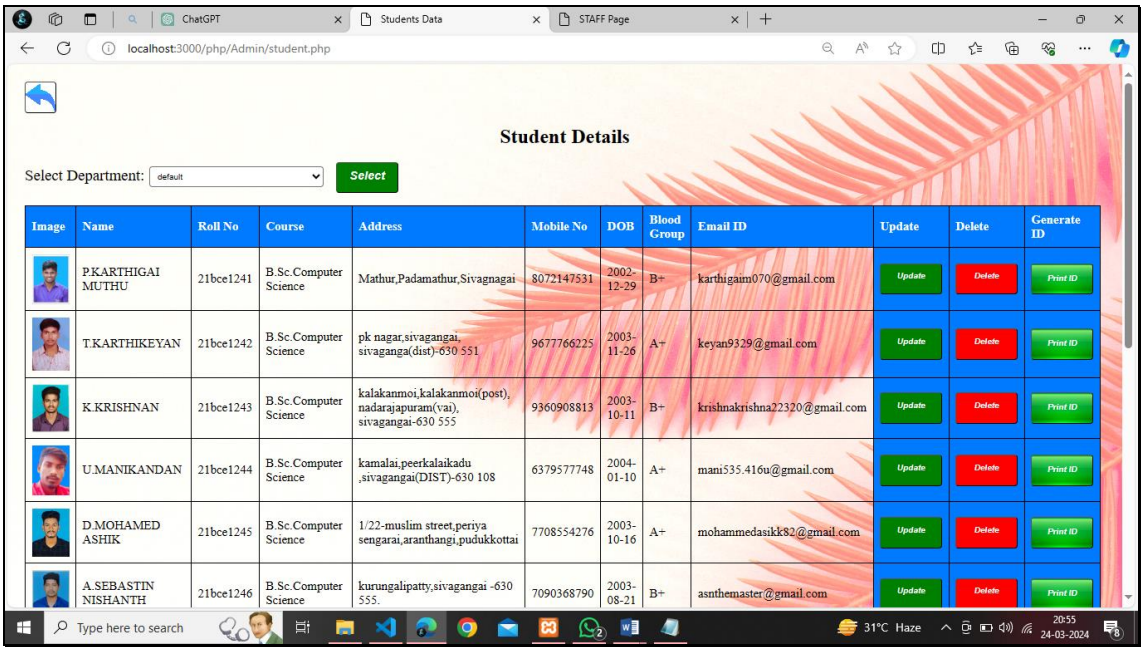
The following screenshot defines that it is for add and remove the staff users. The entered data's where stored in staffuser database table.



The following screenshot defines that it is for add and remove the departments by admin. The entered data's where stored in staffuser database table.

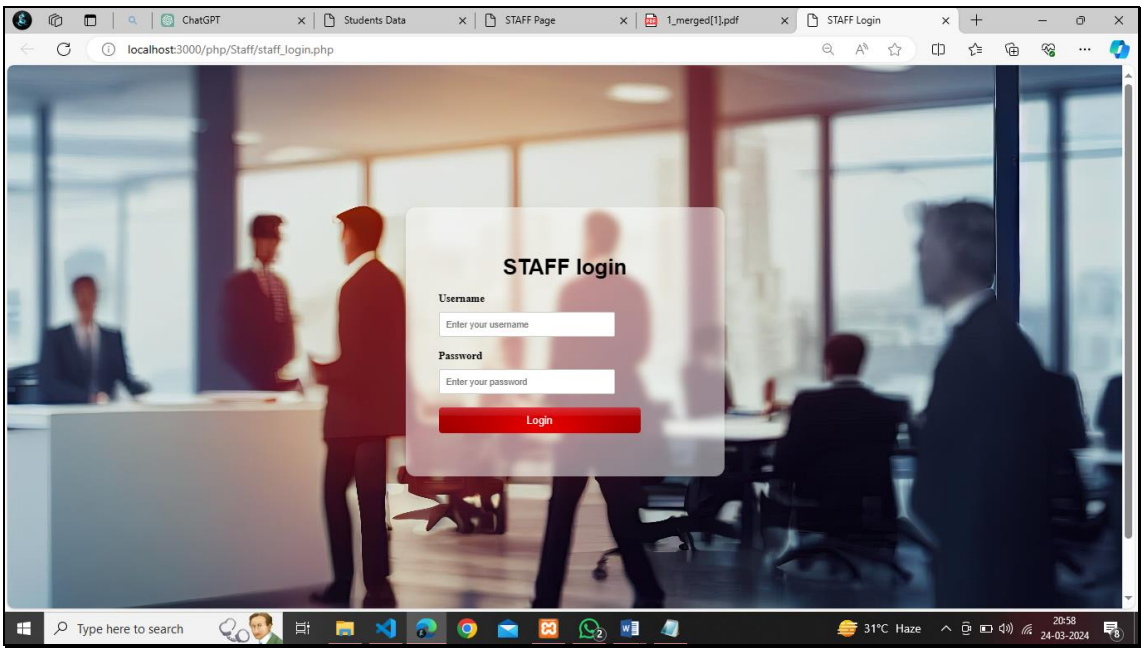


The following screenshot defines that it is for update, delete and pint id cards for students by admin.

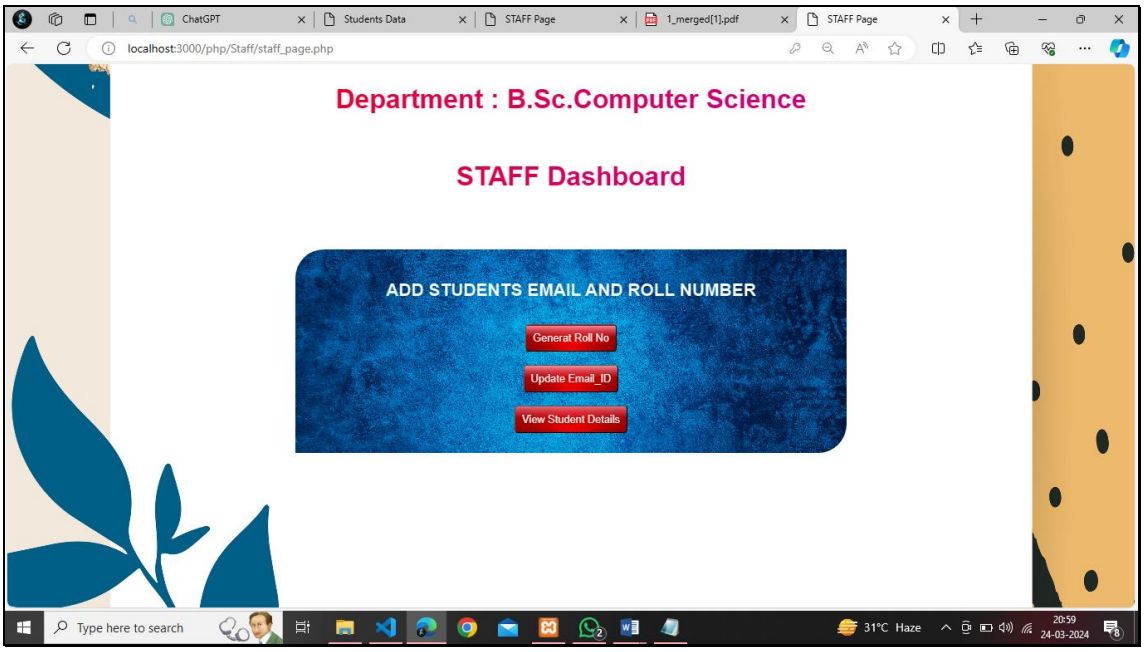


STAFF PAGE

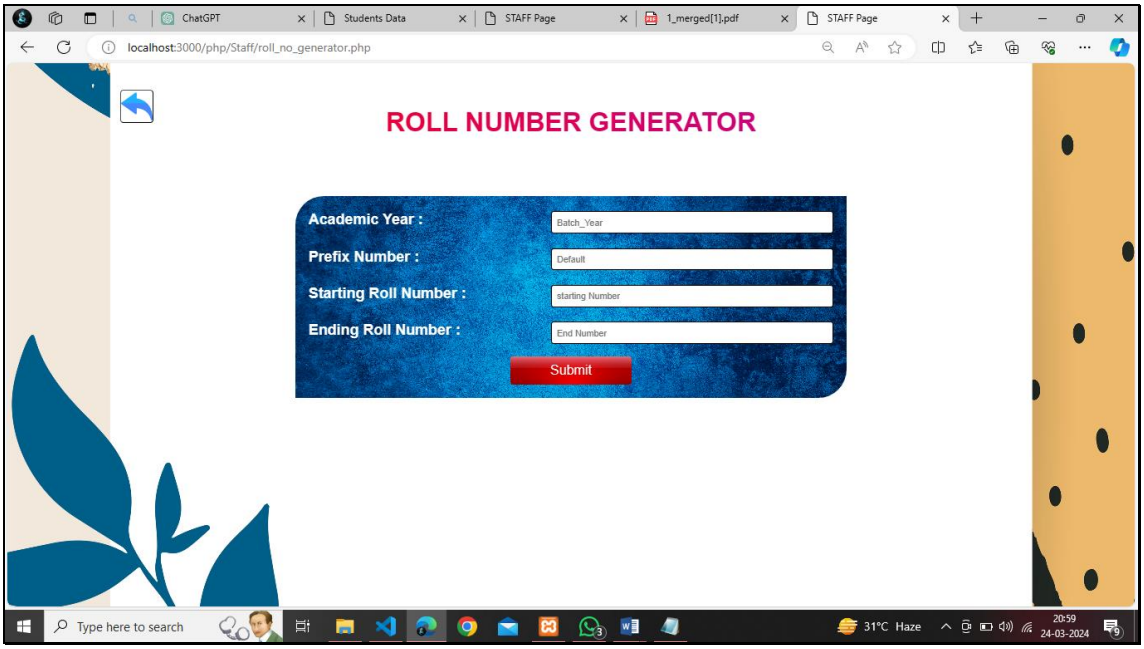
The following screenshot defines that it is the login page of staff users.



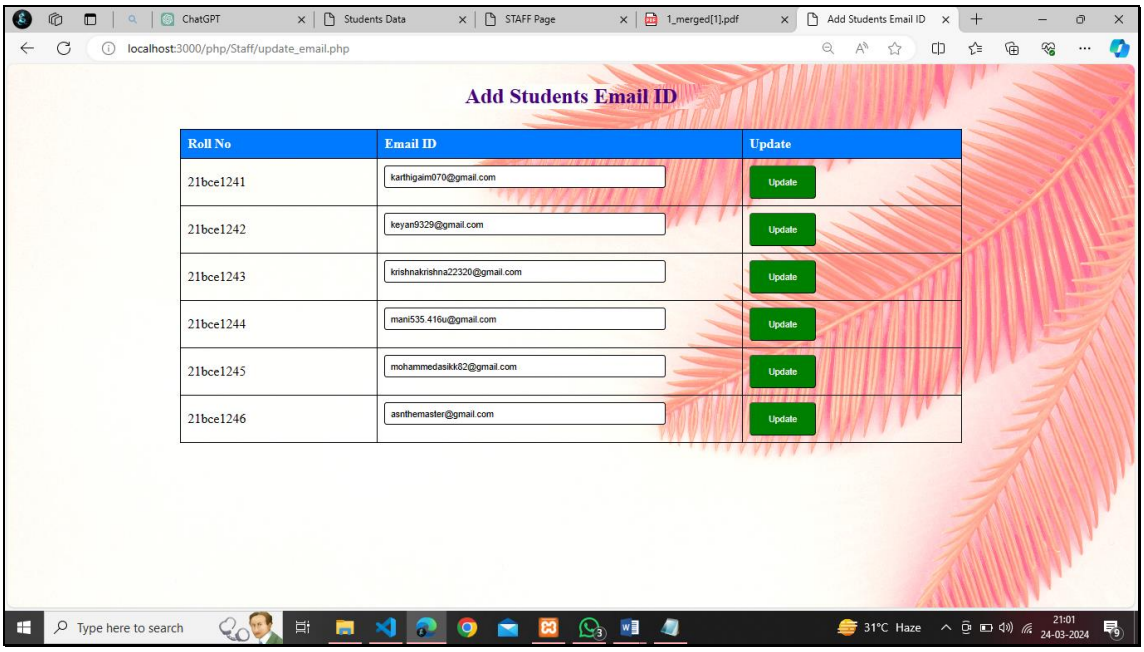
The following screenshot defines that staff can perform process to generate roll numbers, update email id for the students and also can view the details of students.



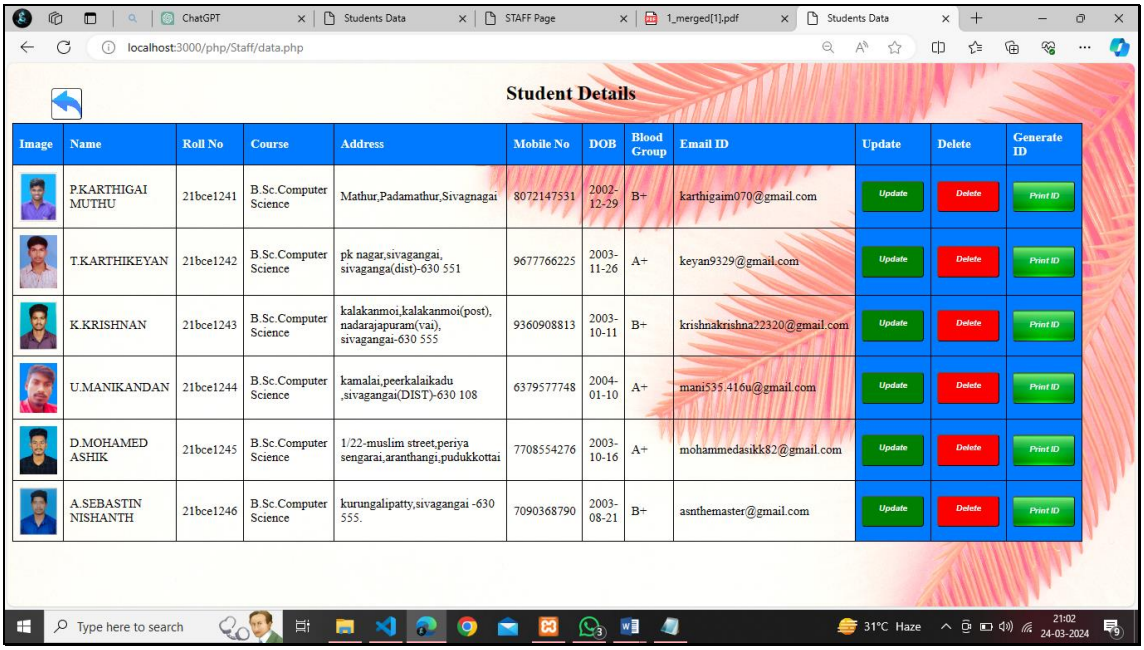
The following screenshot defines that the staff can generate roll number and add academic year for the students. The entered data's where stored in studata database table.



The following screenshot defines that the staff can add and update the email_id for the students. The entered data's where stored in studata database table.

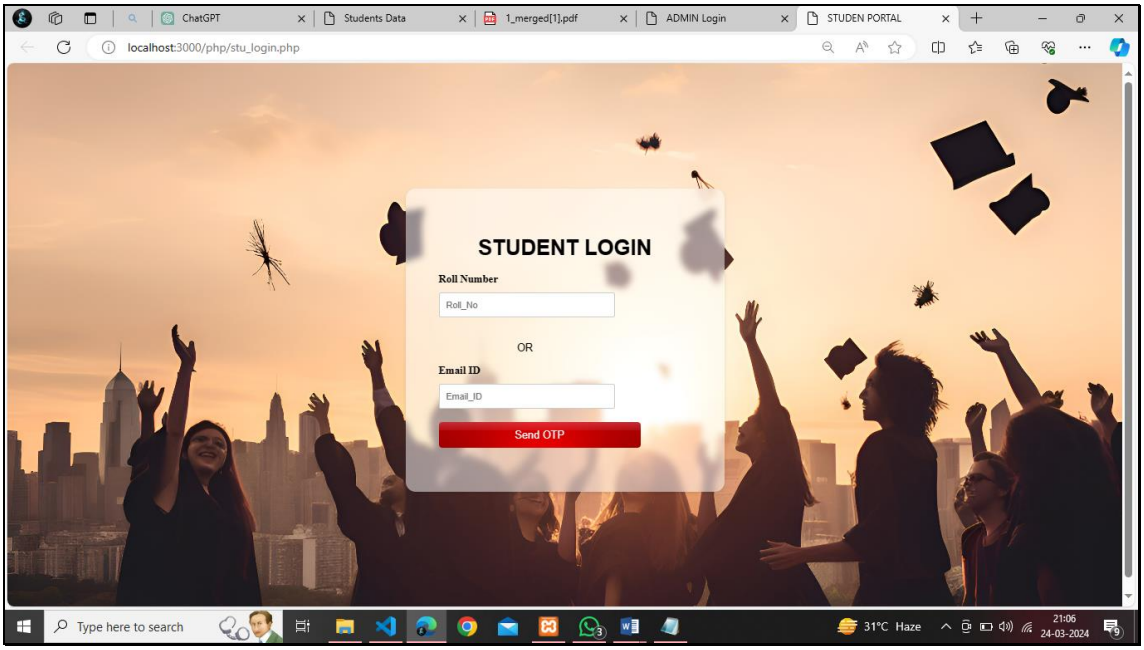


The following screenshot defines that it is for update, delete and pint id cards for students by staffs.

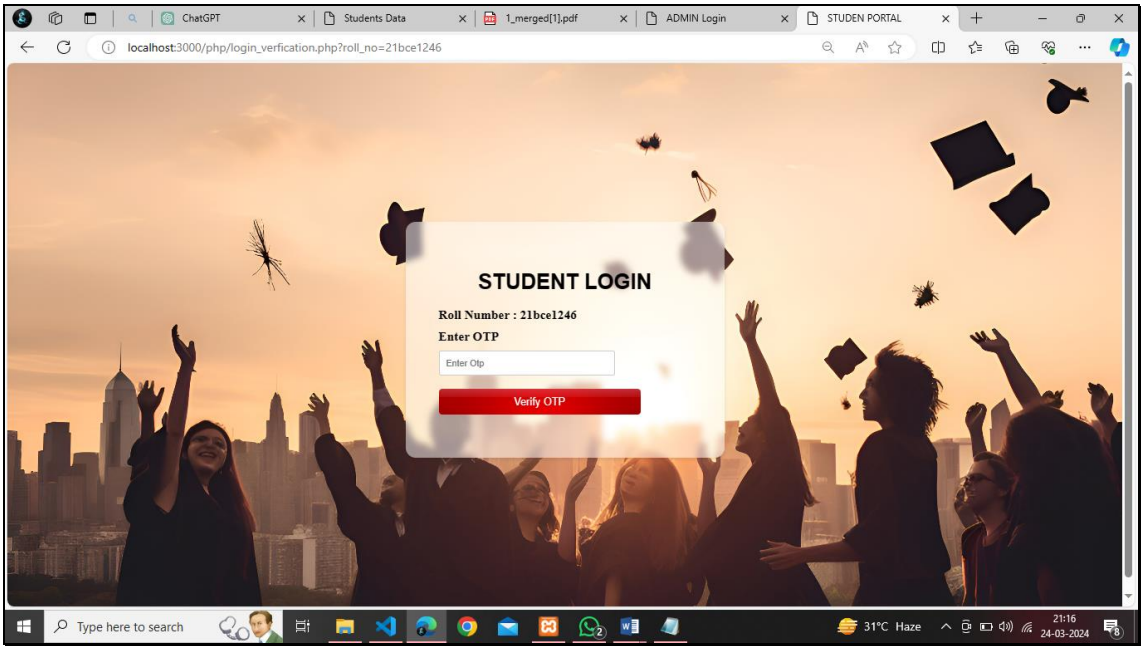


STUDEENT PAGE

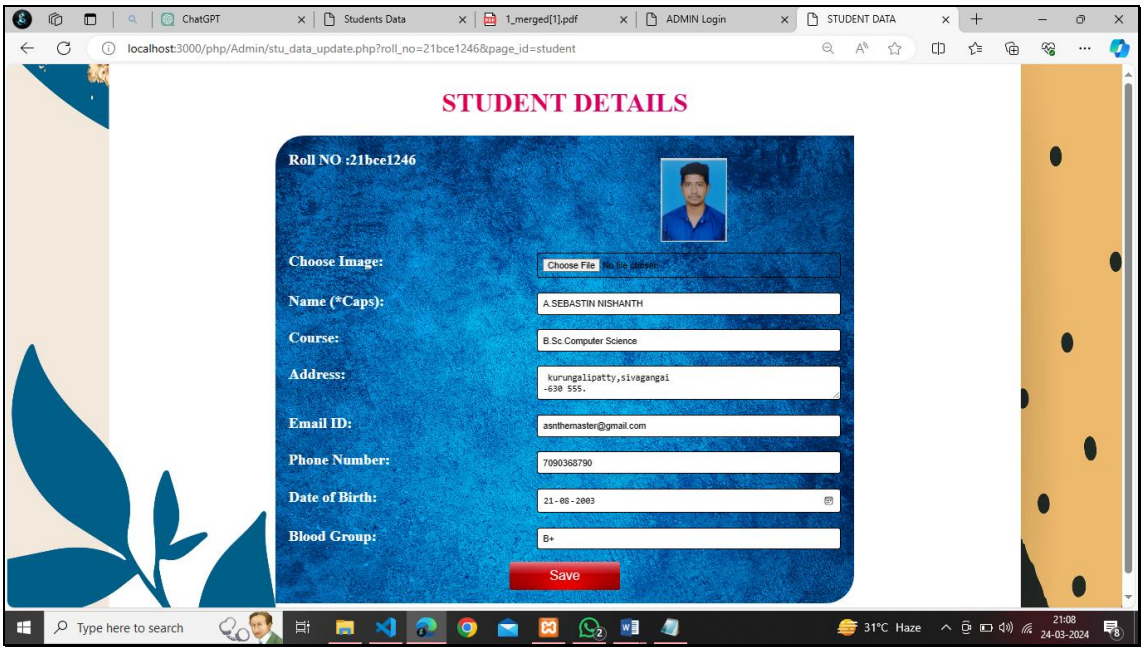
The following screenshot defines that students can login with their roll number or email_id with otp verification.



The following screenshot defines that students are verified with otp which send has already sent to them via email.



The following screenshot defines that students can fill the form with their details and save it. The entered data's where stored in studata database table.



ID CARD GENERATOR PAGE

The following screenshot defines that staff can generate ID cards for students individually and save it as PDF .

D) SAMPLE SOURCE CODE

Admin Login Page

```
<!DOCTYPE html>

<html>

<head>

<title>ADMIN Login</title>

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

</head>

<body>

<div class="all">

<div class="container">

<h1>ADMIN login</h1>

<form action="/php/admin/admin_login.php" method="post">

<label for="username">Username</label>

<input type="text" id="username" name="username" placeholder="Enter your username" required/>

<label for="password">Password</label>

<input type="password" id="password" name="password" placeholder="Enter your password" required />

<button type="submit">Login</button>

</form>

</div>

</div>

</body>

</html>

<?php

$db_host = 'localhost';
```

```
$db_user = 'root';
$db_password = '';
$db_name = 'project';
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);
if($conn)
{
if(!isset($_POST["submit"])){
$user=$_POST['username'];
$password=$_POST['password'];
if($user==null || $password==null){
die();
}
$query="select * from adminuser where username='$user' and password='$password';";
$result=mysqli_query($conn,$query);
if($row=mysqli_fetch_array($result))
{
$user1= $row["username"];
$password1= $row["password"];
if($user1==$user && $password1==$password)
{
header("Location:/php/Admin/home.php");
}
}
else
{
echo "<script>alert('!! UserName And Password  Invalid !!')</script>";}
```

```
}

}

else{

die('Database Connection Error');

}

?>
```

Admin Home Page

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>ADMIN Page</title>

</head>

<body>

<?php

include 'navigation.php';

?>

<h1 >Student ID Generator System</h1>

<h2></h2>

<p >
```

The aim of this project is to develop a Student ID Card Generation System to streamline the process of creatingand managing data. This system will not only enhance efficiency but also provide a seamless solution for thegeneration and management of student ID cards, simplifying administrative tasks for educational establishments.By implementing this technology, the project seeks to optimize the overall identification process and ensureaccurate record-keeping.

On the management side, it takes a considerable amount of time to issue ID cards to students.By the time they distribute the ID cards, the first semester has passed. Additionally, somestudents face difficulties in receiving their ID cards due to the overwhelming workload of thestaff. In some cases, incorrect details are entered and printed. As a result, students arerequired to reapply for their ID cards.

In my project, assigning the management as administrators and staff and students as users proves to be beneficial.The management can create, view, and delete users, as well as access student data. Next, staff members shouldcollect email addresses from students and store them in the database. Afterward, they can generate ID cards and save them for printing. Subsequently, students must log in to the website using their email addresses with OTPverification. Then, they need to fill out the visible form with their details and submit it. The system shouldmaintain the data for future reference. This system enables students to receive their ID cards promptly, avoidingissues such as missing or incorrect details. Additionally, the data of every student is securely maintained.

<footer>

© 2024 Alagappa Arts & Science College , KARAIKUDI

</footer>

</body>

</html>

Add Department page

```
<?php

$db_host = 'localhost';

$db_user = 'root';

$db_password = '';

$db_name = 'project';

$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);

if(isset($_GET['action'])=="delete"){
```

```
$id=$_GET['id'];
global $conn;
$query1= "delete from department where id='$id';";
$result1=mysqli_query($conn,$query1);
if($result1)
{
header("Location:/php/Admin/department.php");
}
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Add department</title>
</head>
<body>
<!-- navigation bar --><
<?php
include 'navigation.php';
?>
<div class="content">
<!-- navigation bar -->
<div class="total">
<div class="add">
<h1>Add Department</h1>
<form action="/php/Admin/department.php" method="post" >
<div class='radio'>
```

```
<div >
<label for="ug">UG:</label>
<input type="radio" id="ug" name="graduation" value="ug">
</div>
<div>
<label for="pg">PG:</label>
<input type="radio" id="pg" name="graduation" value="pg">
</div>
</div>
<div>
<label for="department">DEPARTMENT:</label>
<input type="text" id="department" name="department" required><br><br>
<input type="submit" name="adddepartment" >
</div>
</form>
</div>
<div class="view">
<table>
<tr>
<th>Department ID</th>
<th>Department Name</th>
<th>Graduation</th>
<th>Action</th>
</tr>
<?php
global $conn ;
if ($conn->connect_error) {
die("Connection failed: ". $conn->connect_error);
}
```

```
$sql = "SELECT * FROM department";

$result = mysqli_query($conn, $sql);

if ($result->num_rows > 0) {

// output data of each row

while($row = mysqli_fetch_assoc($result)) {

echo "<tr>";

echo "<td>". $row["id"]. "</td>";

echo "<td>". $row["department"]. "</td>";

echo "<td>". $row["graduation"]. "</td>";

echo "<td> <a href='/php/Admin/department.php?action=delete&id=". $row["id"]. "' >
<button class='delete'>Delete</button> </a></td>";

echo "</tr>";

}

}

?>

<!-- More rows can be added here -->

</table>

</div>

</div>

</div>

</body>

</html>

<?php

global     $conn ;

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

$graduation = $_POST['graduation'];

$department = $_POST['department'];

$check_query = "SELECT * FROM department WHERE department = '$department'";

$check_result = mysqli_query($conn, $check_query);
```

```
if (mysqli_num_rows($check_result) > 0) {

echo "<script>alert('Department Already Exists')</script>";

} else {

$query = "INSERT INTO department ( graduation, department) VALUES ('$graduation', '$department')";

$result = mysqli_query($conn, $query);

if ($result) {

header("Location:/php/Admin/department.php");

} else {

echo "<script>alert('Error: " . mysqli_error($conn) . "')</script>";

}

}

}

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

$query = "DELETE FROM department  ";

$result = mysqli_query($conn, $query);

}

?>
```

Add Staff page:

```
<?php

include 'navigation.php';

?>

<div class="content">

<!-- navigation bar -->

<div class="box1">

<div class="box11">

<div class="add">

<h2>Add Staff Users</h2>
```

```
<form action="/php/Admin/staff.php" method="post" >

<div class='radio'>

<div >

<label for="ug">UG:</label>

<?php

if(isset($_GET['action'])=="setdepartment")

{

$graduation=$_GET['gra'];

if($graduation=="ug")

{

echo"    <input type='radio' name='category' value='ug' onclick=getDepartments('ug') checked>

</div>

<div>

<label for='pg'>PG:</label>

<input type='radio' name='category' value='pg' onclick=getDepartments('pg')>";

}

else{

echo"   <input type='radio' name='category' value='ug' onclick=getDepartments('ug') >

</div>

<div>

<label for='pg'>PG:</label>

<input type='radio' name='category' value='pg' onclick=getDepartments('pg') checked>";

}

}

else{

echo"   <input type='radio' name='category' value='ug' onclick=getDepartments('ug') >

</div>

<div>
```

```
<label for='pg'>PG:</label>
<input type='radio' name='category' value='pg' onclick=getDepartments('pg') >";
}
?>
</div>
</div>
<label for="department">Select Department:
<select id="department" name="department" required>
<option value="default">default</option>
<?php
if(isset($_GET['action'])=="setdepartment")
{
$graduation=$_GET['gra'];
$sql = "SELECT * FROM department WHERE graduation = '$graduation';";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result)>=0)
{
while($row = mysqli_fetch_assoc($result))
{
$department=$row['department'];
echo "<option value='$department'>$department</option>";
}
}
}
?>
</select>
</label>
<label for="username">Username:</label>
<input type="text" id="username" name="username" required><br><br>
```

```
<label for="password">Password:</label>
<input type="password" id="password" name="password" required><br><br>
<input type="submit" name="adduser" value="Add User">
</form>
</div>
<div class="remove">
<h2>Remove Staff User</h2>
<form id=removeuser action="/php/Admin/staff.php" method="post">
<label for="removeuser">Username:</label>
<input type="text" id="username" name="username" required><br><br>
<input type="submit" name="removeuser" value="Remove User">
</form>
<div>
<a href="/php/Admin/view_user.php"> <button type="submit"> VIEW STAFFS</button></a>
</div>
</div>
</div>
</div>
</body>
</html>
<?php
global $conn;
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}
if(isset($_POST['adduser'])) {
$password = $_POST['password'];
$username = $_POST['username'];
```

```
$category = $_POST['category'];
$department = $_POST['department'];
$check_query = "SELECT * FROM staffuser WHERE username = '$username'";
$check_result = mysqli_query($conn, $check_query);
if (mysqli_num_rows($check_result) > 0) {
echo "<script>alert('User Already Exists')</script>";
} else {
$query = "INSERT INTO staffuser (username, password, category, department) VALUES ('$username', '$password', '$category', '$department')";
$result = mysqli_query($conn, $query);
if ($result) {
echo "<script>alert('New USER Created Successfully')</script>";
} else {
echo "<script>alert('Error: " . mysqli_error($conn) . "')</script>";
}
}
}
if(isset($_POST['removeuser'])) {
$username = $_POST['username'];
$query = "DELETE FROM staffuser WHERE username = '$username'";
$result = mysqli_query($conn, $query);
if ($result) {
if (mysqli_affected_rows($conn) > 0) {
echo "<script>alert('USER Removed Successfully')</script>";
} else {
echo "<script>alert('USER Not Available')</script>";
}
} else {
echo "<script>alert('Error executing query: " . mysqli_error($conn) . "')</script>";
```

```
}
}// Close connection
$conn->close();
?>
<script>
function getDepartments(category) {
window.location.href="/php/Admin/staff.php?gra="+category+"&action=setdepartment
";
<?php
//header('location:');
?>
}
</script>
```

Staff Login Page:

```
<?php
$db_host = 'localhost';
$db_user = 'root';
$db_password = '';
$db_name = 'project';
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);
if($conn)
{
if(!isset($_POST["submit"])){
$user=$_POST['username'];
$password=$_POST['password'];
if($user==null || $password==null){
die();
}
```

```
$query="select * from staffuser where username='$user' and password='$password';";
$result=mysqli_query($conn,$query);
if($row=mysqli_fetch_array($result))
{
session_start();
$department=$row['department'];
$gra=$row['category'];
$_SESSION['dep']=$department;
$user1= $row["username"];
$password1= $row["password"];
if($user1==$user && $password1==$password)
{
header("Location:/php/Staff/staff_page.php");
echo 'sucessful';
}
}
else
{
echo "<script>alert(' Invalid UserName And Password ')</script>";
}
}
}
else{
die('not connected');
}?>
```

Staff Page:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" type="text/css" href="/css/style.css" />
<title>STAFF Page</title>
</head>
<body>
<header>
<h1>STAFF Dashboard</h1>
</header>
<div class="stu_data ">
<div class="center">
<div class="heading">
<h3>ADD STUDENTS EMAIL AND ROLL NUMBER</h3>
</div>
<div >
<a href="/php/Staff/roll_no_generator.php"> <button class="action-button" type="submit"> Generat Roll No </button></a> </div>
<div >
<a href="/php/Staff/update_email.php"> <button class="action-button" type="submit">Update Email_ID</button></a>
</div>
<div >
<a href="/php/Staff/data.php"><button class="action-button" type="submit" > View Student Details</button></a>
</div>
</div>
```

```
</div>
</body>
</html>
<?php
$servername = 'localhost';
$username = 'root';
$password = '';
$dbname = 'project';
$conn = mysqli_connect($servername, $username, $password, $dbname);
if($conn) {
if(isset($_POST['submit'])) {
$roll_no = $_POST['roll_no'];
$email_id = $_POST['email_id'];
$check_query = "SELECT * FROM studata WHERE roll_no = '$roll_no' and email_id = '$email_id'";
$check_result = mysqli_query($conn, $check_query);
if(mysqli_num_rows($check_result) > 0) {
echo "<script>alert('Email_ID and Roll number already exists')</script>";
} else {
$sql = "INSERT INTO studata (email_id,roll_no) VALUES ('$email_id','$roll_no')";
$result = mysqli_query($conn, $sql);
if($result) {
echo "<script>alert('Successfully submitted')</script>";
} else {
echo "<script>alert('Failed to submit')</script>";
die( "Error: " . $sql . "<br>" . $conn->error);
}
}
}
```

```
}
$conn->close();
?>
```

Generate Roll_no page

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" type="text/css" href="/css/style.css" />
<title>STAFF Page</title>
</head>
<body>
<header>
<a href="/php/Staff/staff_page.php" ><button class="back"></button></a>
<h1> ROLL NUMBER GENERATOR</h1>
</header>
<div class="stu_data">
<form action="/php/Staff/roll_no_generator.php" method="post" >
<div class="input1">
<!-- Roll Number -->
<label for="roll_number">Academic Year :</label>
<input type="text" name="year" id="roll_number" required placeholder="Batch_Year"/>
</div>
<div class="input1">
<!-- Roll Number -->
<label for="roll_number">Prefix Number :</label>
```

```
<input type="text" name="defult"  id="roll_number" required placeholder="Default"/>

</div>

<div class="input1">

<!-- Roll Number -->

<label for="roll_number">Starting Roll Number :</label>

<input type="text" name="start"  id="roll_number" required placeholder="starting Number"/>

</div>

<div class="input1">

<!-- Roll Number -->

<label for="roll_number">Ending Roll Number :</label>

<input type="text" name="end"  id="roll_number" required placeholder="End Number"/>

</div>

<!-- Submit Button -->

<input type="submit" name="submit" value="Submit" />

</form>

</div>

</body>

</html>

<?php

$servername = 'localhost';

$username = 'root';

$password = '';

$dbname = 'project';

// Create connection

$conn = mysqli_connect($servername, $username, $password, $dbname);

// Check connection

if (!$conn) {

die("Connection failed: " . mysqli_connect_error());
```

```
}
if (isset($_POST['submit'])) {
$defult = $_POST['defult'];
$start = $_POST['start'];
$end = $_POST['end'];
$year=$_POST['year'];
session_start();
$course=$_SESSION['dep'];
while ($start <= $end) {
//finding $start number lenthgh
if($end>=100)
{
if($start<10)
{
$start_number="00".$start;
}
else if($start<100){
$start_number="0".$start;
}
else{
$start_number=$start;
}
}
else{
if($start<10)
{
$start_number="0".$start;
}
else{
```

```
$start_number=$start;
}
}
$roll_number = $defult . $start_number;
$query = "INSERT INTO studata(roll_no,batch_year,course) VALUES ('$roll_number','$year','$course');";
$result = mysqli_query($conn, $query);
$start++;
}
if ($result) {
echo "<script>alert('Successfully Added')</script>";
} else {
echo "<script>alert('Error: Unable to insert data')</script>";
}
}
mysqli_close($conn);
?>
```

Update Student Email_id page

```
<?php
$db_host = 'localhost';
$db_user = 'root';
$db_password = '';
$db_name = 'project';
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}
if(isset($_POST['updateemail']))
```

```
{
echo "<script>alert('Update Sucessfull')</script>";
$roll_no=$_POST['roll_no'];
$email=$_POST['email'];
$query="UPDATE studata SET email_id='$email' WHERE roll_no='$roll_no';";
$result=mysqli_query($conn,$query);
if($result){
//set Alreat
echo "<script>alert('Update Sucessfull')</script>";
}
else{
echo "<script>alert('Update Failed')</script>";
}
}?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Add Students Email ID</title>
</head>
<body>
<h1>Add Students Email ID</h1>
<table>
<tr>
<th>Roll No</th>
<th>Email ID</th>
<th>Update</th>
</tr>
```

```
<?php
session_start();
$course=$_SESSION['dep'];
$query1 = "SELECT * FROM studata where course='$course';";
$result1 = mysqli_query($conn, $query1);
while ($row = mysqli_fetch_assoc($result1)) {
$rollno = $row['roll_no'];
$email = $row['email_id'];
echo "<tr>";
echo "<form action='#' method='post' autocomplete='off'>
<input type='hidden' name='roll_no' value='$rollno' required>";
echo "<td>".$row['roll_no']."</td>";
echo "<td><input type='email' name='email' autocomplete='off' value='$email' required/></td>";
echo "<td><button name='updateemail' class='update'>Update</button></td> </form>";
echo "</tr> ";
}
?>
</table>
</body>
</html>
```

Student details processing page:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Students Data</title>
</head>
<body>
<a href="/php/Admin/home.php" ><button class="back"></button></a>
<h1>Student Details</h1>
<label for="department">Select Department:</label>
<form action="#" method="post">
<select id="department" name="department" required>
<option value="default">default</option>
<?php
$qurey="select * from department;";
$result=mysqli_query($conn,$qurey);
while($row=mysqli_fetch_assoc($result))
{
$department=$row['department'];
echo "<option value='$department'>$department</option>";
}
?>
</select>
<button class="but" name="selete">Select</button>
</form>
<table>
<th>Image</th>
<th>Name</th>
<th>Roll No</th>
<th>Course</th>
<th>Address</th>
<th>Mobile No</th>
<th>DOB</th>
```

```
<th>Blood Group</th>
<th>Email ID</th>
<th>Update</th>
<th>Delete</th>
<th>Generate ID</th>
<tr>
<?php
if(isset($_POST['selete']))
{
$department=$_POST['department'];
// echo $department;
$query1="select * from studata where course='$department';";
$result1=mysqli_query($conn,$query1);
while ($row = mysqli_fetch_assoc($result1)) {
echo "<tr>";
echo "<td>";
retriveImage($row['roll_no']);
echo"</td>";
echo "<td>".$row['name']."</td>";
echo "<td>".$row['roll_no']."</td>";
echo "<td>".$row['course']."</td>";
$rollno=$row['roll_no'];
echo "<td>".$row['address']."</td>";
echo "<td>".$row['mobile_no']."</td>";
echo "<td>".$row['dob']."</td>";
echo "<td>".$row['blood_group']."</td>";
echo "<td>".$row['email_id']."</td>";
echo "<th><a href='/php/Admin/stu_data_update.php?roll_no=$rollno&page_id=admin' ><button class=
```

```
'update' >Update</button></a></th>";

echo" <th><a href='/php/Admin/data_delete.php?deleteroll_no=$rollno' ><button class=

'delete' >Delete</button></a></th>";

echo" <th><a href='/php/Admin/idcard/idcard2.php?roll_no=$rollno' ><button  class=

'id' >Print ID</button></a></th>";

"</tr>";

}

}

else{

$query1='select * from studata';

$result1=mysqli_query($conn,$query1);

while ($row = mysqli_fetch_assoc($result1)) {

echo "<tr>";

echo "<td>";

retriveImage($row['roll_no']);

echo"</td>";

//echo "<td><img src='{$row['image']}' alt='Student Image' width='50' height='50'></td>";

echo "<td>".$row['name']."</td>";

echo "<td>".$row['roll_no']."</td>";

echo "<td>".$row['course']."</td>";

$rollno=$row['roll_no'];

echo "<td>".$row['address']."</td>";

echo "<td>".$row['mobile_no']."</td>";

echo "<td>".$row['dob']."</td>";

echo "<td>".$row['blood_group']."</td>";

echo "<td>".$row['email_id']."</td>";
```

```
echo"                                                    <th><a
href='/php/Admin/stu_data_update.php?roll_no=$rollno&page_id=admin'        ><button
class=

'update' >Update</button></a></th>";

echo"   <th><a    href='/php/Admin/data_delete.php?deleteroll_no=$rollno'    ><button
class=

'delete' >Delete</button></a></th>";

echo" <th><a href='/php/Admin/idcard/idcard2.php?roll_no=$rollno' ><button  class=

'id' >Print ID</button></a></th>";

"</tr>";

}

}

?>

</tr>

</table>

</body></html>
```

//delete student details

```
<?php

$db_host = 'localhost';

$db_user = 'root';

$db_password = '';

$db_name = 'project';

$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);

if ($conn->connect_error) {

die("Connection failed: " . $conn->connect_error);

}

if(isset($_GET['deleteroll_no']))

{

$roll=$_GET['deleteroll_no'];
```

```
global $conn;
$query1= "delete from studata where roll_no='$roll';";
$result1=mysqli_query($conn,$query1);
if($result1)
{
header("Location:/php/Admin/student.php");
}
}
?>
```

Students Login Page:

```
<?php
$db_host = 'localhost';
$db_user = 'root';
$db_password = '';
$db_name = 'project';
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);
if($conn)
{
if(isset($_POST["submit"])){
$roll_no=$_POST['rollnumber'];
$email_id=$_POST['email_id'];
if ($roll_no == "") {
$query = "SELECT * FROM studata WHERE email_id='$email_id'";
$result = mysqli_query($conn, $query);
if ($result) {
$row = mysqli_fetch_array($result);
if ($row) {
$roll_no = $row['roll_no'];
```

```
sendotp($email_id);
header("Location: /php/login_verfication.php?roll_no=$roll_no");
exit; // Ensure script execution stops after redirection
} else {
echo "<script>alert('Invalid Email')</script>";
}
} else {
// Log the error or provide a more informative message
echo "<script>alert('Database error: " . mysqli_error($conn) . "')</script>";
}
}
else{

$query="select * from studata where roll_no='$roll_no' ;";
$result=mysqli_query($conn,$query);
if(mysqli_num_rows($result))
{
if($row=mysqli_fetch_array($result))
{
$roll_no=$row['roll_no'];
$email_id=$row['email_id'];
sendotp($email_id);
header("Location:/php/login_verfication.php?roll_no=$roll_no");
}
}
else
{
echo "<script>alert('!! Invalid roll NO')</script>";
}
```

```
}
}
}
else{
die('Database Connection Error');
}
?>
```

Students Login Verification page:

```
<?php
include ("../phpmailer/otpcode.php");
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="/css/login.css" />
<title>STUDEN PORTAL</title>
</head>
<body>
<div class="all">
<div class="container">
<h1>STUDENT LOGIN</h1>
<form method="post" action="#">
<?php
```

```
if(isset($_GET["roll_no"]))
{
$roll_no=$_GET["roll_no"];
echo "     <label for='rollnumber'>Roll Number : $roll_no</label>
<input type='hidden' value='$roll_no' name='roll_no'/>";

}
?>
<?php
session_start();
$sendOtp=$_SESSION['otp'];
?>
<label for="email">Enter OTP</label>
<input type="text" id="email" name="otp" placeholder="Enter Otp" />
<button type="submit" name="verfication">Verify OTP</button>
</form>
</div>
</div>
</body>
</html>
<?php
if(isset($_POST['verfication'])){
$userotp=$_POST['otp'];
$roll_no=$_POST['roll_no'];
echo $userotp;
if( otpverfiction($userotp))
{
session_start();
$_SESSION['studentlogin']=true;
```

```
$_SESSION['studentroll_no']=$roll_no;


header("location:../php/Admin/stu_data_update.php?roll_no=$roll_no&page_id=studen
t");
}
else{
echo "<script>alert('Invalid OTP');</script>";
}
}
?>
```

Students form page:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>STUDENT DATA</title>
<link rel="stylesheet" type="text/css" href="/css/style.css" />
</head>
<body>
<div class="stu_data">
<div class="heading">
<h1>STUDENT DETAILS</h1>
</div>
<?php
$db_host = 'localhost';
$db_user = 'root';
$db_password = '';
```

```
$db_name = 'project';
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}
if(isset($_GET['roll_no']))
{
$roll=$_GET['roll_no'];
$page_id=$_GET['page_id'];
global $conn;
$query1= "select * from studata where roll_no='$roll';";
$result1=mysqli_query($conn,$query1);
if($result1)
{
while($row=mysqli_fetch_assoc($result1))
{
//$row array to get all value is
$name=$row['name'];
$course=$row['course'];
$rollno=$row['roll_no'];
$address=$row['address'];
$phone=$row['mobile_no'];
$dob=$row['dob'];
$blood_group=$row['blood_group'];
$email=$row['email_id'];
$image=$row['image'];
}
}
}
```

```
?>

<form action="/php/Admin/update_module.php?page_id=".$page_id method="post" enctype="multipart/form-data">

<div class="input1">

<input type="hidden" value="<?php echo $rollno?>" name="roll_no">

<label type="text" name="u" id="roll_number"><?php echo "Roll NO :" .$roll;?></label>

<img style="margin: 10px;" width="100px" src="

<?php echo base64_encode($image); ?>" />

</div>

<div class="input1">

<label for="image">Choose Image:</label>

<input type="file"  name="image" id="image" accept="image/*">

</div>

<div class="input1">

<!-- Name -->

<label for="name">Name (*Caps):</label>

<input type="text" value="<?php echo $name;?>" name="name" id="name"required placeholder="Enter Name in capital letters" />

</div>

<div class="input1">

<!-- Course -->

<label for="course">Course:</label>

<input value="<?php  echo $course?>" type="text" name="course" id="course" readonly required placeholder="Enter Course"/>

</div>

<div class="input1">

<!-- Address -->

<label for="address">Address:</label>

<textarea name="address" id="address"  required placeholder="Address" >
```

```
<?php echo $address;?>

</textarea>

</div>

<div class="input1">

<label for="email">Email ID:</label>

<input value="<?php echo $email;?>"  type="email" name="email_id" id="email" readonly required placeholder="abcd098@gmail.com"/>

</div>

<div class="input1">

<label for="mobile_no">Phone Number:</label>

<input value="<?php echo $phone;?>" type="phone" name="mobile_no" id="mobile_no" required placeholder="91+"/>

</div>

<div class="input1">

<!-- Date of Birth -->

<label for="dob">Date of Birth:</label>

<input value="<?php echo $dob;?>" type="date" name="dob" id="dob" required />

</div>

<div class="input1">

<!-- Blood Group -->

<label for="blood_group">Blood Group:</label>

<input value="<?php echo $blood_group;?>" type="text" name="blood_group" id="blood_group" required />

</div>

<!-- Submit Button -->

<input type="submit" name="submit" value="Save" />

</form>

</div>

</body>

</html>
```

ID Card Generator Page

```
<?php
$db_host = 'localhost';
$db_user = 'root';
$db_password = '';
$db_name = 'project';
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}
if(isset($_GET['roll_no']))
{
$roll_no=$_GET['roll_no'];
$query1="select * from studata where roll_no='$roll_no';";
$result1=mysqli_query($conn,$query1);
if($result1){
echo"</td>";
$row=mysqli_fetch_assoc($result1);
//echo "<td><img src='{$row['image']}' alt='Student Image' width='50' height='50'></td>";
$name=$row['name'];
$roll_no=$row['roll_no'];
$course=$row['course'];
$rollno=$row['roll_no'];
$address=$row['address'];
$mobile=$row['mobile_no'];
$dob=$row['dob'];
$year=$row['batch_year'];
```

```
$blood_group=$row['blood_group'];
$email=$row['email_id'];
}
}
function  retriveImage($roll_no)
{

// Get image data from database
global $conn;
$db=$conn;
$query= "SELECT image FROM studata  where roll_no='$roll_no'";
$result = mysqli_query($db,$query);
?>
<!-- Display images with BLOB data from database -->
<?php if($result->num_rows > 0){ ?>
<div class="gallery">
<?php while($row = $result->fetch_assoc()){ ?>
<image x="0.25" y="0.25" width="200" height="249" src="
<?php echo base64_encode($row['image']); ?></image>" />
<?php } ?>
</div>
<?php
}
else{
?>
<p class="status error">Image(s) not found...</p>
<?php }
}
```

```
?>

<html>

<div class="id">

<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" width="804" height="509" viewBox="0 0 804 509">

<defs>

<linearGradient id="linear-gradient" x1="0.5" y1="1" x2="0.5" gradientUnits="objectBoundingBox">

<stop offset="0" stop-color="#fff"/>

<stop offset="0.228" stop-color="#bbffb6"/>

<stop offset="1" stop-color="#088100"/>

</linearGradient>

<pattern id="pattern" preserveAspectRatio="none" width="100%" height="100%" viewBox="0 0 1053 951">

<rect id="alagappar_logo" width="120" height="109" transform="translate(684 10)" fill="url(#pattern)"/>

<text id="ALAGAPPA_GOVT_ARTS_COLLEGE" data-name="ALAGAPPA GOVT ARTS COLLEGE" transform="translate(113 60)" fill="#0018b9" font-size="37" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0">ALAGAPPA GOVT ARTS COLLEGE</tspan></text>

<text id="_GRADE_-1_COLLEGE_" data-name="(GRADE -1 COLLEGE)" transform="translate(309 91)" fill="red" font-size="19" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0">(GRADE -1 COLLEGE)</tspan></text>

<text id="KARAIKUDI-630_003_TEL:04565-224521" data-name="KARAIKUDI-630 003, TEL:04565-224521" transform="translate(234 119)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0">KARAIKUDI-630 003, TEL:04565-224521</tspan></text>

<text id="NAME_:" data-name="NAME :" transform="translate(241 209)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0" xml:space="preserve">NAME :</tspan></text>

<text id="P.Karthigaimuthu" transform="translate(364 209)" fill="#11007d" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0"><?php echo $name; ?></tspan></text>
```

<text id="IDENTITY_CARD_2021_-2024" data-name="IDENTITY CARD" transform="translate(247 177)" fill="#11007d" font-size="25" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0">IDENTITY CARD <?php echo $year; ?></tspan></text>

<text id="P.Karthigaimuthu-2" data-name="P.Karthigaimuthu" transform="translate(362 247)" fill="#11007d" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0"><?php echo $roll_no; ?></tspan></text>

<text id="P.Karthigaimuthu-3" data-name="P.Karthigaimuthu" transform="translate(365 285)" fill="#11007d" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0"><?php echo $course; ?></tspan></text>

<text width="100" id="P.Karthigaimuthu-4" data-name="P.Karthigaimuthu" transform="translate(366 323)" fill="#11007d" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0"><?php echo $address; ?></tspan></text>

<text id="P.Karthigaimuthu-5" data-name="P.Karthigaimuthu" transform="translate(366 395)" fill="#11007d" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0"><?php echo $mobile; ?></tspan></text>

<text id="P.Karthigaimuthu-6" data-name="P.Karthigaimuthu" transform="translate(369 433)" fill="#11007d" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0"><?php echo $dob; ?></tspan></text>

<text id="Principal" transform="translate(670 479)" fill="#11007d" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0">Principal</tspan></text>

<text id="ROLL_NO_:" data-name="ROLL NO :" transform="translate(241 247)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0" xml:space="preserve">ROLL NO :</tspan></text>

<text id="COURSE_:" data-name="COURSE :" transform="translate(241 285)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0" xml:space="preserve">COURSE :</tspan></text>

<text id="ADDRESS_:" data-name="ADDRESS :" transform="translate(241 323)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0" xml:space="preserve">ADDRESS :</tspan></text>

<text id="MOBILE_NO_:" data-name="MOBILE NO :" transform="translate(241 395)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-

```
weight="700"><tspan x="0" y="0" xml:space="preserve">MOBILE NO :</tspan></text>

<text id="DOB_:" data-name="DOB :" transform="translate(241 433)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0" xml:space="preserve">DOB :</tspan></text>

<text id="B_ve" data-name="B +ve" transform="translate(101 487)" fill="red" font-size="18" font-family="SegoeUI-Bold, Segoe UI" font-weight="700"><tspan x="0" y="0"><?php echo $blood_group; ?></tspan></text>

<g id="Rectangle_2" data-name="Rectangle 2" transform="translate(31 190)" fill="#fff" stroke="#707070" stroke-width="0.5">

<!-- <rect width="200" height="249" stroke="none"/> -->

<!-- <rect x="0.25" y="0.25" width="199.5" height="248.5" fill="none"/> -->

//add image in database

</g>

<rect id="Image_1" data-name="Image 1" width="41" height="40" transform="translate(60 460)" fill="url(#pattern-2)"/>

<line id="Line_1" data-name="Line 1" x2="672" transform="translate(69.5 145.5)" fill="none" stroke="#160663" stroke-width="4"/>

<rect id="ps" width="166" height="71" transform="translate(624 421)" fill="url(#pattern-3)"/>

</g>

</svg>

</div>

<div class="photo">

<img  src="

<?php echo base64_encode($row['image']); ?>" />

<div>

</html>

<script>

window.print();

</script>
```

Css for Login page

```
body {
font-family: Arial, sans-serif;
}
.all {
display: flex;
justify-content: center;
flex-direction: column;
justify-items: center;
padding: auto;
height: 100vh;
}
.container {
width: 380px;
margin: 10px auto;
padding: 50px;
backdrop-filter: blur(5px);
background: rgba(255, 255, 255, 0.5);
border-radius: 15px;
box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
}
h1 {
text-align: center;
margin-bottom: 20px;
```

```
}
ol {
padding: o;
}
label {
display: block;
margin-bottom: 10px;
font-weight: bold;
font-family: 'Times New Roman', Times, serif;
}

input[type="text"],
input[type="email"],
input[type="password"] {
width: 70%;
padding: 10px;
border: 1px solid #ccc;
border-radius: 3px;
margin-bottom: 20px;
}
.index {
width: 380px;
margin: 150px auto;
padding: 50px;
backdrop-filter: blur(5px);
background: rgba(255, 255, 255, 0.5);
border-radius: 15px;
box-shadow: 0 2px 5px rgba(0, 0, 0, 0.5);
text-align: center;
```

```
list-style: disc;
}
button {
width: 80%;
padding: 10px;
align-self: center;
font-size: 16px;
border-radius: 5px;
background: #ffffff url(/im/red_button.jpg) center center/cover no-repeat;
color: #fff;
border: #000000;
cursor: pointer;
transition: background-color 0.3s ease;
}
button:hover {
background: #ffffff url(/im/green_button.jpg) center center/cover no-repeat;
}
button a {
text-decoration: none;
color: inherit;
}
.stu {
width: 280px;
margin: 150px auto;
padding: 50px;
backdrop-filter: blur(5px);
background: rgba(255, 255, 255, 0.5);
border-radius: 15px;
box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
```

```
}
.view {
text-align: center;
margin: 20px;
}
```

Css for Form pages

```
form {
background: #ffffff url(/im/stu_form_color.jpg) center center/cover no-repeat;
padding: 20px;
align-self: center;
border-radius: 43px 0px 43px 0px;
font-size: 140%;
width: 50%;
align-items: center;
display: flex;
flex-direction: column;
font-weight: bold;
backdrop-filter: blur(30px);
}
.center {
background: #ffffff url(/im/stu_form_color.jpg) center center/cover no-repeat;
padding: 20px;
align-self: center;
border-radius: 43px 0px 43px 0px;
font-size: 140%;
width: 50%;
align-items: center;
display: flex;
flex-direction: column;
```

```
font-weight: bold;
backdrop-filter: blur(30px);
}
label {
display: block;
margin-bottom: 8px;
color: #ffffff;
text-align: left;
width: 550px;
}
input[type="text"],
input[type="tel"],
input[type="email"],
input[type="date"],
input[type="file"],
input[type="phone"],
textarea {
width: 80%;
padding: 8px ;
margin-bottom: 16px;
border: 1px solid #ccc;
border-radius: 4px;
border-color: #000000;
}
input[type="submit"] {
background: #ffffff url(/im/red_button.jpg) center center/cover no-repeat;
color: #ffffff;
padding: 10px 60px;
border: none;
```

```
border-radius: 4px;
cursor: pointer;
font-size:20px;
}
input[type="submit"]:hover {
background: #ffffff url(/im/green_button.jpg) center center/cover no-repeat;
}
.stu_data {
display: flex;
flex-direction: column;
}
.input1 {
display: flex;
width: 100%;
margin: 3px;
}
h1 {
text-align: center;
background: linear-gradient(to left, rgb(182, 0, 182), rgb(255, 0, 0));
background-clip: text;
color: transparent;
font-size: 40px;
padding: auto;
}
textarea{
text-align: left;
}
```