

# PROJECT REPORT



**HIGHER NATIONAL DIPLOMA IN INFORMATION TECHNOLOGY**

**HARDY ADVANCED TECHNOLOGICAL INSTITUTE(SLIATE)**

**TITLE:-Nanasa Tuition Class Center Management System**

**MODULE: - HNDIT 2304-Project (Group)**

**YEAR OF SUBMISSION: - 2021**

No	Index No	Name
01	AMP/IT/2019/F/0006	O.K.D.Harshana Lakmal
02	AMP/IT/2019/F/0080	H.B.R.Chathuni Ariyasena
03	AMP/IT/2019/F/0083	K.A.K.H.Diwyanjalee
04	AMP/IT/2019/F/0084	W.M. Nisansala Sewwandi
05	AMP/IT/2019/F/0094	A.M. Lakshitha Udayanga

**Name of Supervisors : Ms. T.H. Dinusha Darshani**

**Signature : .....**

**Date : .....**

## DECLARATION

The final project to be carried out during our final semester studying in the Department of Information Technology, Hardy Advanced Technological Institute was the successfully completed. This project was decided by all of our team and we all confirm that this project is not a project that has been done before or owned by others.

- AMP/IT/2019/F/0006 .....
- AMP/IT/2019/F/0080 .....
- AMP/IT/2019/F/0083 .....
- AMP/IT/2019/F/0084 .....
- AMP/IT/2019/F/0094 .....

## **Acknowledgement**

In today's world of competition there is a competition for survival. Project is like a bridge between theoretical and practical work. We volunteered to be involved in this unique project. We have our pilot with us, and we would like to pay our respects to Madam Dinusha Darshani, head of department of information technology, hardy advanced technology institute ,who led us with love and encouragement up to this stage. Finally, we would like to thank all our lectures and brothers, sisters who have no words to say thank you, but our hearts are always filled with the joy that comes from every person.

## **Abstract**

Nanasa tuition center is an institution that has been conducting advanced examination results for a long time but its administrative efficiency has decreased. Furthermore, due to the current epidemic situation, had to do it online so we implemented a successful methodology through the team project.

## TABLE OF CONTENTS

Title Page	i
Declaration	ii
Acknowledgment	iii
Abstract	iv
Table of content	v
List of figures	vii
List of table	viii
Abbreviations	ix
<b>CAPTER 01 – INTRODUCTION</b>	<b>1</b>
1.1 Introduction of Nanasa Tuition Class	1
1.2 Current system	2
1.3 Purpose of the project	2
1.4 The way of doing the project	3
<b>CHAPTER 02-PROBLEM DEFINITION &amp; SYSTEM ANALYST</b>	<b>3</b>
2.1 Problem	3
<b>CHAPTER 03-SYSTEM DESIGN</b>	<b>4</b>
3.1 ER diagram	4
<b>CHAPTER 04-DEVELOPMENT</b>	<b>4</b>
4.1 Client side development	4
4.1.1 Setup the requirement	4
4.1.2 Creating interface	5
4.1.3 Home Page	5
4.1.4 Registration Page	7
4.1.5 Login Page	9
4.1.6 verify email	10
4.1.7 Verification success data base	11

4.1.8 Verification process success	12
4.1.9 Verification not success	13
4.1.10 Payment Page	13
4.1.11 Contact us	15
4.1.12 Recover password	16
<b>CHAPTER 05-EVALUATION &amp; CONCLUSION</b>	<b>17</b>
5.1 Evaluations	17
5.2 Conclusions	17
5.3 Future improvements	17
<b>CHAPTER 06-REFERENCES</b>	<b>18</b>
<b>LIST OF FIGURES</b>	
Figure1.1: Nanasa tuition class center	1
Figure1.2: Queue in class	2
Figure4.1: Creating interface	5
Figure4.2: Home page	5
Figure4.3: Logo	6
Figure4.4: home page php code	6
Figure4.5: home page css code	7
Figure4.6: Registration Page	7
Figure4.7: Registration Page php code	8
Figure4.8: wrong register(error)	8
Figure4.9: wrong register(error) php code	9
Figure4.10: Login page	9
Figure4.11: Login page html code	10
Figure4.12: verify email page	10

Figure4.13: verify email page php code	11
Figure4.14: verification success data base	11
Figure4.15: verification process success	12
Figure4.16: verification process success php code	12
Figure4.17: verification not success	13
Figure4.18: payment page	13
Figure4.19: payment page html code	14
Figure4.20: payment page css code	14
Figure4.21: contact us page	15
Figure4.22: contact us page php code	15
Figure4.23: contact us page javascript code	16
Figure4.24: recover password	16

## **LIST OF TABLE**

Table 3.1 Data flow diagram

4

## **ABBREVIATIONS**

- PC –Personal Computer
- HTML-Hyper Text Markup Language
- PHP-Hyper Pre-Processor
- OS-Operating System
- SQL-Structured Query Language
- URL-Uniform Resource Locator
- WWW-World Wide Web
- DFD-Data Flow Diagram



## **Chapter 01- INTRODUCTION**

### **Introduction**

Nanasa Tuition class center is an educational institution which has been maintaining high examination results for a long time. However, due to the current epidemic situation class process was weakened. This project was carried out to make the Nenasa institute of education more efficient by implementing that process under online technology. The objective of our project is to provide an efficient service to all students through the online system.



Figure 1.1 Nenasa Tuition Class Center

### **1.1 Introduction of Nenasa Tuition Class Center**

Nenasa Educational Institution is the where we did the project. It is located in Wellawaya area and has been providing continuous education to the students for many years. A large number of students from grade 01 to grade 13 come to this place for education. The education center was very comfortable and had about two staff to charge tuition fees, provide tutorials and keep records of students and teachers information. This educational institute produces excellent student.

## 1.2 Current System

Nenasa education currently operate through the file system, so students have to wait in long queues to pay tuition, receive tutorials and attend attendance. It's a very time wasting process every day in this class. moreover, such queuing is unsafe due to the current epidemic. some students have to wait in long queues after school to return to class.

Students record their attendance at the first entry in to the class, pay the tuition fees, and then enter the class with the tutorial set. The biggest reason for queuing is to find the book where the students of institute are registered. Because the data keepers of this institute look at each other's grades and record them one by one. Sometimes finding a name is really hard and a waste of them. so a large queue will be generated.



Figure1.2 Queue in class

## 1.3 Purpose of the project

Many tuition class centers traditionally maintain all the information in their institution by writing in books this method reduces the efficiency here. Also, due to the current epidemic, these activities had to be done from home but today, with the advancement of technology, these processes are internet based. It stores information about all the students who attend the classes and the information about the teachers who teach. Many things like exams, lectures, class fees are done online. The most important thing is to be able to get the information you need quickly.

So in our project we created online. In this system, students and teachers can access this. Another feature of this new system is prepaid system. In the prepaid mode, a certain amount can be deposited. At the end of the month, the system automatically reduces its account balance. When all students use this feature, there will be no more queues for cashiers. It will be really useful for students and staff as well.

## **1.4 The way of doing the project**

First we studied the Nenasa institute. We've identified its weaknesses. It's really useful to develop a new online system. We turn to the internet to find out what kind of system is appropriate to solve those problems. We will also learn more about php and sql. we also learned and used the bootstrap service to respond to it. after referring to those sources, we began building our project. We had some problems but in the end we reached the goal of our project..

## **CHAPTER 02 – PROBLEM DEFINITION & SYSTEM ANALYSIS**

### **2.1 Problems**

There are various errors in the current system. Failure to find the required information about a student or teacher quickly, Difficulty in updating relevant information, Inability to conduct examinations due to the epidemic situation, Difficulty in storing large amounts of information, Large time to mark class attendance, Class fees are difficult to pay. There are many such problems in the current system.

## CHAPTER 03 –SYSTEM DESIGN

### 3.1 Data flow diagram

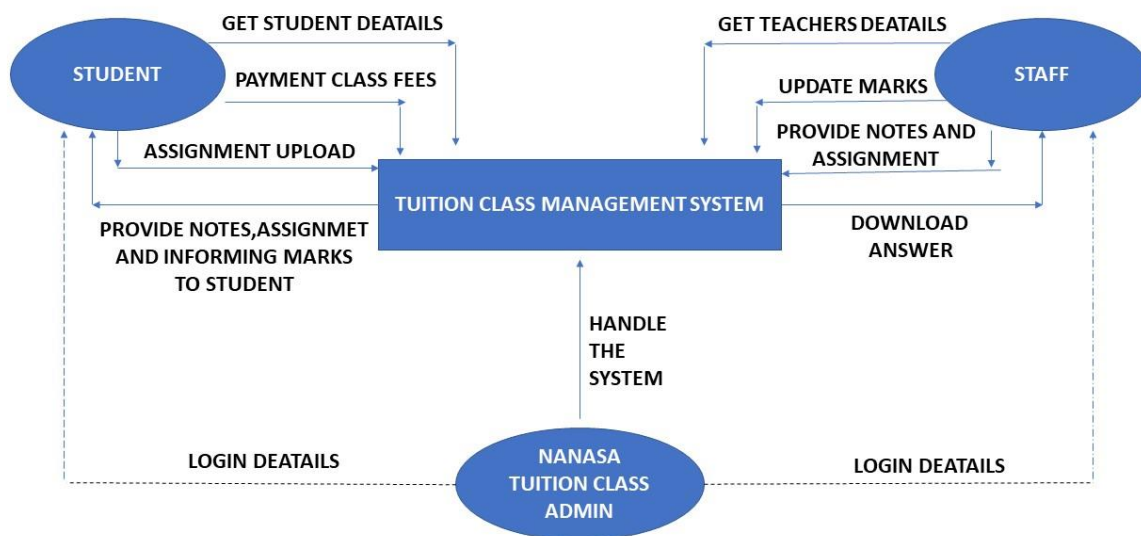


Table 3.1 Data flow diagram

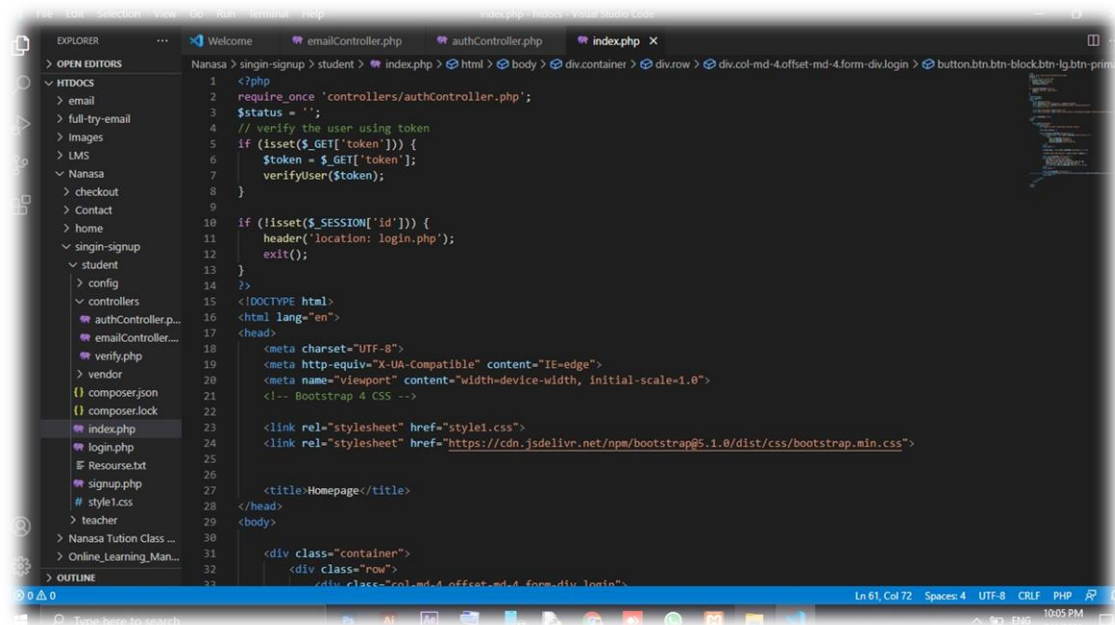
## CHAPTER 04 –DEVELOPMENT & TESTING

### 4.1 Client side development

#### 4.1.1 Setup the requirement

First of all we started the Xampp sever and created our system folder inside the NSdox folder where Xampp server installed. And then opened the system folder using sublime text code editor.

## 4.1.2 Creating interface



```
1 <?php
2 require_once 'controllers/authController.php';
3 $status = '';
4 // verify the user using token
5 if (isset($_GET['token'])) {
6     $token = $_GET['token'];
7     verifyUser($token);
8 }
9
10 if (!isset($_SESSION['id'])) {
11     header('location: login.php');
12     exit();
13 }
14 >?
15 <!DOCTYPE html>
16 <html lang="en">
17 <head>
18     <meta charset="UTF-8">
19     <meta http-equiv="X-UA-Compatible" content="IE=edge">
20     <meta name="viewport" content="width=device-width, initial-scale=1.0">
21     <!-- Bootstrap 4 CSS -->
22     <link rel="stylesheet" href="style1.css">
23     <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.css">
24
25     <title>Homepage</title>
26 </head>
27 <body>
28
29     <div class="container">
30         <div class="row">
31             <div class="col-md-4 offset-md-4 form-div login">
```

Figure: 4.1 Creating interface

We used php language. For the easy maintains we thought to add header sections using php. for that we should start the connection. this is coding part

## 4.1.3 Home Page

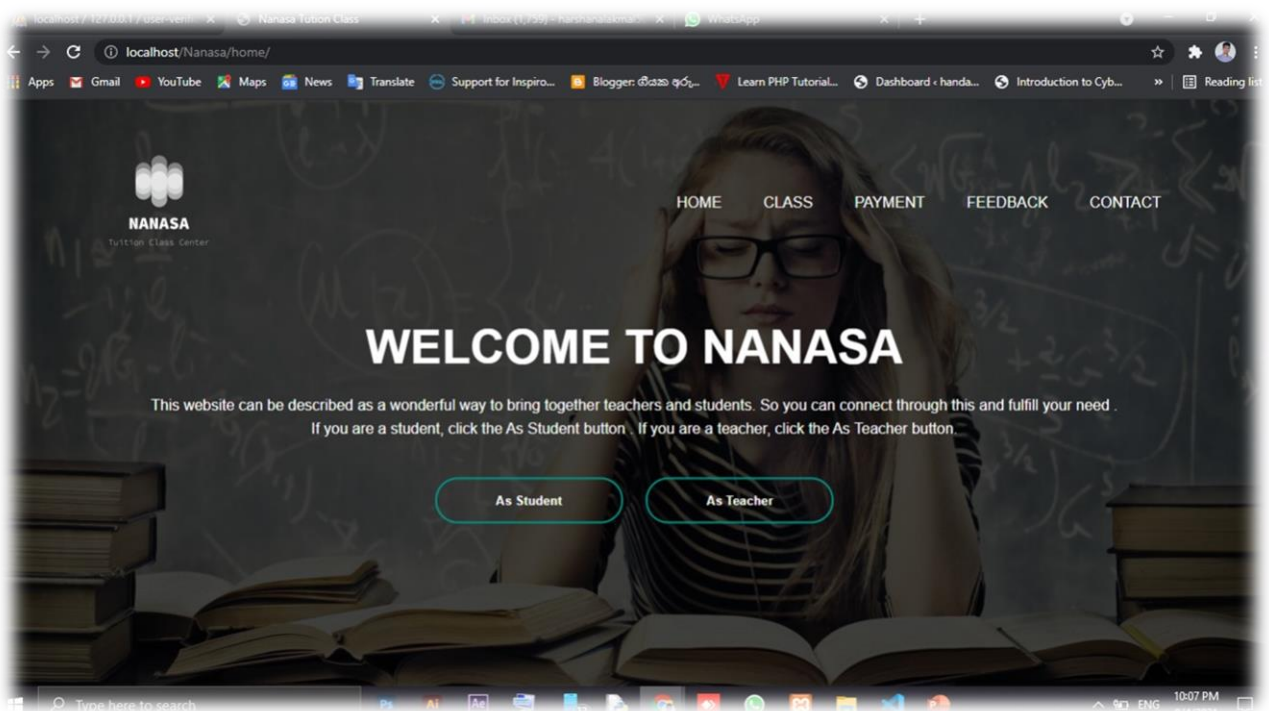


Figure : 4.2 Home page

Our homepage is designed primarily for students and teachers only. It has five options. The company appears on the left.



Figure:4.3 logo

```
index.php X
Nanasa > home > index.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <title>Nanasa Tution Class</title>
5  <link rel="stylesheet" href="style.css">
6  </head>
7  <body>
8
9  <div class="banner">
10   <div class="navbar">
11     
12     <ul>
13       <li><a href="#">Home</a></li>
14       <li><a href="#">Class</a></li>
15       <li><a href="../checkout/index.html">Payment</a></li>
16       <li><a href="#">Feedback</a></li>
17       <li><a href="../Contact/contact-us.php">Contact</a></li>
18     </ul>
19   </div>
20
21   <div class="content">
22     <h1>WELCOME TO NANASA</h1>
23     <p>This website can be described as a wonderful way to bring together teachers and students. So you can connect th
24
25     <div>
26       <a href="../signin-signup/student/signup.php"><button type="button"><span></span>As Student</button></a>
27       <a href="../signin-signup/teacher/signup.php"><button type="button"><span></span>As Teacher</button></a>
28     </div>
29   </div>
30 </div>
31
32 </div>
```

Figure:4.4 home page php code

```
index.php # style.css X
Nanasa > home > # style.css > *
1 *{
2   margin: 0;
3   padding: 0;
4   font-family: arial;
5 }
6 .banner{
7   width:100%;
8   Height:100vh;
9   background-image:linear-gradient(□rgba(0,0,0,0.75),□rgba(0,0,0,0.75)),url(Images/student-study-stress-img.jpg);
10  background-size: cover;
11  background-position: center;
12 }
13 .navbar{
14   width:85%;
15   margin:auto;
16   padding:35px 0;
17   display:flex;
18   align-items:center;
19   justify-content:space-between;
20 }
21 .logo{
22   width:150px;
23   cursor:pointer;
24 }
25 .navbar ul li{
26   list-style:none;
27   display:inline-block;
28   margin:0 20px;
29   position:relative;
30 }
31 .navbar ul li a{
32   text-decoration:none;
```

Figure:4.5 home page css code

#### 4.1.4 Registration Page

### Register

Username

Email

Password

Confirm Password

[Sign Up](#)

Already a member? [Sign In](#)

Figure 4.6 : Registration Page

```

1  <?php
2
3  session_start();
4
5  require 'config/db.php';
6  require_once 'config/constants.php';
7
8
9  $errors = array();
10 $username = "";
11 $email = "";
12
13 // if user clicks on the sign up button
14 if (isset($_POST['signup-btn'])) {
15     $username = $_POST['username'];
16     $email = $_POST['email'];
17     $password = $_POST['password'];
18     $passwordConf = $_POST['passwordConf'];
19
20     // validation
21     if (empty($username)){
22         $errors['username'] = "Username required";
23     }
24
25     if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
26         $errors['email'] = "Email address is invalid";
27     }
28
29     if (empty($password)){
30         $errors['password'] = "Password required";
31     }
32
33     if (empty($passwordConf)){
34         $errors['passwordConf'] = "Password confirmation required";
35     }
36
37     if ($password != $passwordConf){
38         $errors['passwordConf'] = "Passwords do not match";
39     }
40
41     if (count($errors) > 0){
42         // show errors
43     }
44
45     // insert user
46 }

```

Figure 4.7 : Registration Page php code

### Register

- Username required
- Email required
- Password required

Username

Email

Password

Confirm Password

Figure :4.8 wrong register(error)



```

19 // validation
20 if (empty($username)){
21     $errors['username'] = "Username required";
22 }
23
24 if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
25     $errors['email'] = "Email address is invalid";
26 }
27
28
29 if (empty($email)){
30     $errors['email'] = "Email required";
31 }
32
33 if (empty($password)){
34     $errors['password'] = "Password required";
35 }
36
37 if ($password != $passwordConf) {
38     $errors['password'] = "The two password do not match";
39 }
40
41
42 $emailQuery = "SELECT * FROM user WHERE email=? LIMIT 1";
43 $stmt = $conn->prepare($emailQuery);
44 $stmt->bind_param('s', $email);
45 $stmt->execute();
46 $result = $stmt->get_result();
47 $userCount = $result->num_rows;
48 $stmt->close();
49
50 if ($userCount > 0) {
51     $errors['email'] = "Email already exists";
52 }

```

Figure :4.9 wrong register(error) php code

You can register here by entering your user name and email.the special features here are **sign up** and **sign in**. If you are a new student,you will need to go through **sign up**.If you have already registered,you can **sign in**.

#### 4.1.5 Login Page

Login to an account ,customers must have registered before.And password should exactly match with previously added password otherwise he/she couldn't login and they 'll see this error.

Figure: 4.10 Login page

```

index.php style.css authController.php login.php X
Nanasa > singin-signup > student > login.php > html > body > div.container > div.row > div.col-md-4.offset-md-4.form-div.login > form > div > a
17
18 <div class="container">
19 <div class="row">
20 <div class="col-md-4 offset-md-4 form-div login">
21 <form action="login.php" method="post">
22 <h3 class="text-center">Login</h3>
23 <br>
24
25 <?php if(count($errors) > 0): ?>
26 <div class="alert alert-danger">
27 <?php foreach($errors as $error): ?>
28 <li><?php echo $error; ?></li>
29 <?php endforeach; ?>
30 </div>
31 <?php endif; ?>
32
33 <div class="form-group">
34 <label for="username">Username or Email</label>
35 <input type="text" name="username" value="<?php echo $username; ?>" class="form-control form-control-lg">
36 </div>
37
38 <div class="form-group">
39 <label for="password">Password</label>
40 <input type="password" name="password" class="form-control form-control-lg">
41 </div>
42 <br>
43 <center>
44 <div class="form-group">
45 <button type="submit" name="login-btn" class="btn btn-primary btn-block btn-lg">Login</button>
46 </div>
47 </center><br>

```

Figure: 4.11 Login page html code

#### 4.1.6 verify email

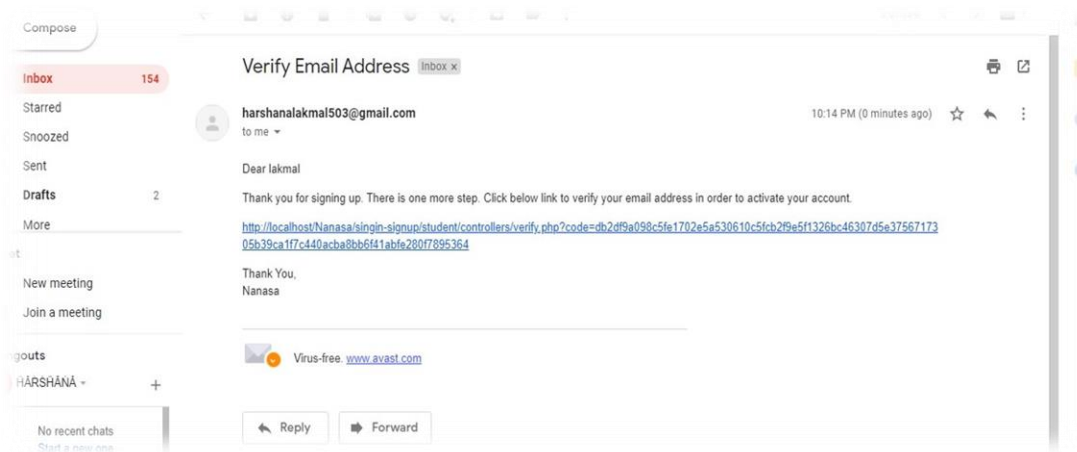


Figure: 4.12 verify email page

This will allow you to view the verified email address. Then you can go to the verification success database.

```

69
70
71
72 $verification_code = sha1($email . time());
73 $verification_URL = 'http://localhost/Nanasa/singin-signup/student/controllers/verify.php?code=' . $token;
74
75 $query = "INSERT INTO user (username, email, verified, token, password) VALUES ('{$username}', '{$email}', fal
76
77 $result = mysqli_query($connection, $query);
78
79 // mail sending code
80 $to = $email; // receiver
81 $sender = 'harshanalakmal503@gmail.com'; // email address of the sender
82 $mail_subject = 'Verify Email Address';
83 $email_body = '<p>Dear ' . $username . '</p>';
84 $email_body .= '<p>Thank you for signing up. There is one more step.
Click below link to verify your email address in order to activate your account.</p>';
85 $email_body .= '<p>' . $verification_URL . '</p>';
86 $email_body .= '<p>Thank You, <br>Nanasa</p>';
87
88
89 $header = "From: {$sender}\r\nContent-Type: text/html;";
90
91 $send_mail_result = mail($to, $mail_subject, $email_body, $header);
92
93 if ( $send_mail_result ) {
94     // mail sent successfully
95     echo 'Please check your email.';
96 } else {
97     // mail could not be sent
98     echo 'Error.';
99 }
100

```

Figure: 4.13 verify email page php code

## 4.1.7 verification success data base

Showing rows 0 - 0 (1 total. Query took 0.0006 seconds.)

SELECT \* FROM 'user'

Options: Show all | Number of rows: 25 | Filter rows: Search this table

	id	username	email	verified	token	password
	47	lakmal	harshanalakmal427@gmail.com	1	\$2y\$10\$7PzwjEzHgUuAS5UzQjTOL301sQz6jF5dHBM04nRJ...	

Query results operations: Print | Copy to clipboard | Export | Display chart | Create view

Figure: 4.14 verification success data base

#### 4.1.8 Verification process success

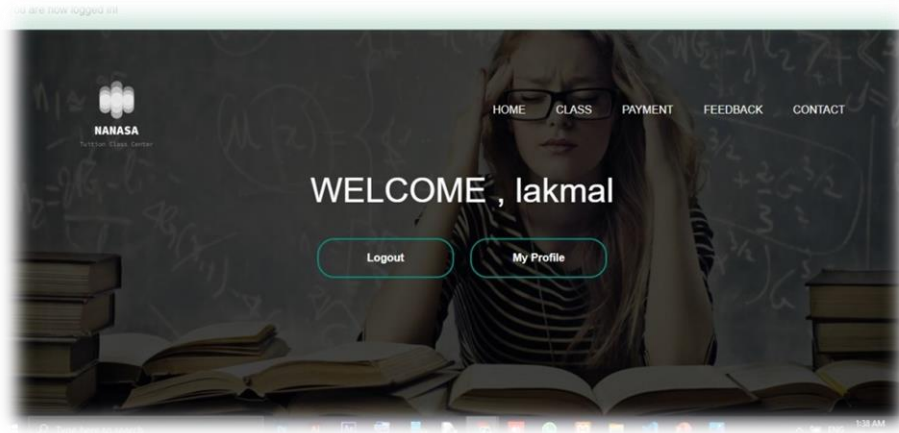


Figure: 4.15 verification process success

When the verification is successful, a welcome page with the name of the student will be displayed. This will let you know that the verification process was successful.

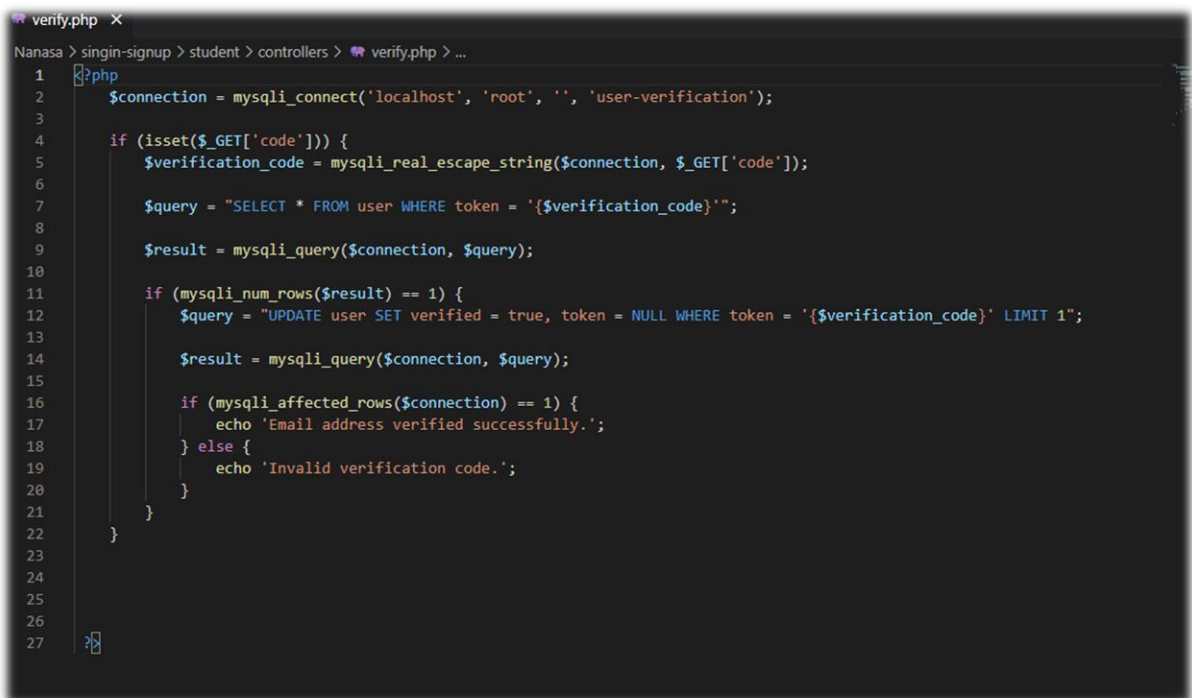


Figure: 4.16 verification process success php code

#### 4.1.9 Verification not success

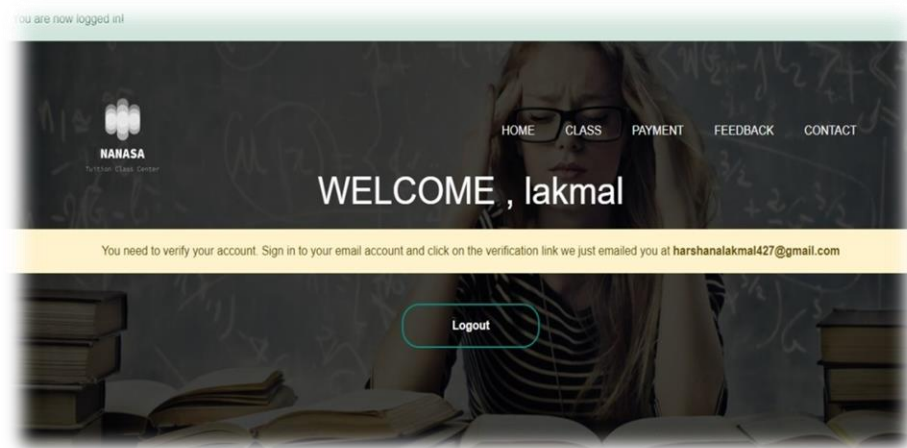


Figure:4.17 verification not success

This error show when verification is not successful.

#### 4.1.10 payment page

Figure:4.18 payment page

Payment for the relevant month will be made through this web page. For this the student's register number is required.

```

verify.php # styles.css index.html X
Nanasa > checkout > index.html > ...
20
21 <h2>Payment Gateway</h2>
22 <div class="form">
23   <div class="card space icon-relative">
24     <label class="label">Card holder:</label>
25     <input type="text" class="input" placeholder="Coding Market">
26     <i class="fas fa-user"></i>
27   </div>
28   <div class="card space icon-relative">
29     <label class="label">Card number:</label>
30     <input type="text" class="input" data-mask="0000 0000 0000 0000" placeholder="Card Number">
31     <i class="far fa-credit-card"></i>
32   </div>
33   <div class="card-grp space">
34     <div class="card-item icon-relative">
35       <label class="label">Expiry date:</label>
36       <input type="text" name="expiry-data" class="input" data-mask="00 / 00" placeholder="00 / 00">
37       <i class="far fa-calendar-alt"></i>
38     </div>
39     <div class="card-item icon-relative">
40       <label class="label">CVC:</label>
41       <input type="text" class="input" data-mask="000" placeholder="000">
42       <i class="fas fa-lock"></i>
43     </div>
44   </div>
45   <div class="btn">
46     Pay
47   </div>
48 </div>
49
50 </div>
51 </div>

```

Figure:4.19 payment page html code

```

verify.php # styles.css X index.html
Nanasa > checkout > # styles.css > ...
1 @import url('https://fonts.googleapis.com/css?family=Baloo+Bhaijaan|Ubuntu');
2
3 *{
4   margin: 0;
5   padding: 0;
6   box-sizing: border-box;
7   font-family: 'Ubuntu', sans-serif;
8 }
9
10 body{
11   background: #009688;
12   margin: 0 10px;
13 }
14
15 .payment{
16   background: #f8f8f8;
17   max-width: 360px;
18   margin: 80px auto;
19   height: auto;
20   padding: 35px;
21   padding-top: 70px;
22   border-radius: 5px;
23   position: relative;
24 }
25
26 .payment h2{
27   text-align: center;
28   letter-spacing: 2px;
29   margin-bottom: 40px;
30   color: #0d3c61;
31 }
32

```

Figure:4.20 payment page css code

#### 4.1.11 Contact us



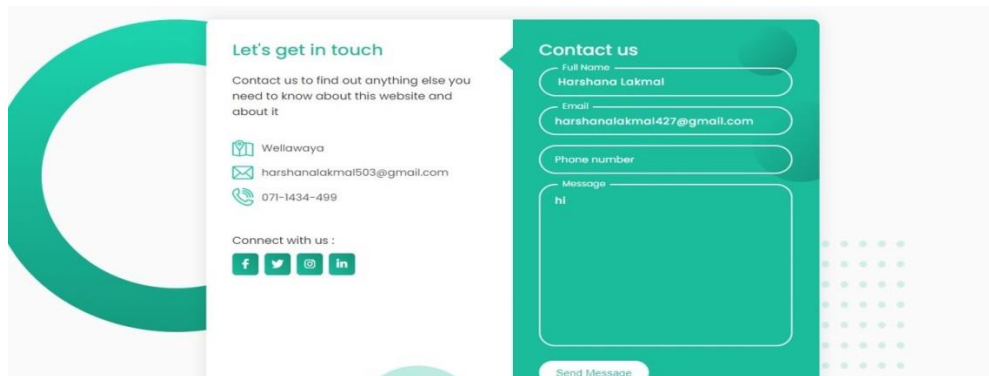


Figure:4.21 contact us page

If you want to know any information ,a separate page has been aside for that as well.

```

Nanasa > Contact > contact-us.php > html > head
1  <?php
2      $status = '';
3      // checking if the form is submit
4      if ( isset($_POST['submit']) ) {
5          $fullname = $_POST['fullname'];
6          $email = $_POST['email'];
7          $subject = $_POST['subject'];
8          $message = $_POST['message'];
9
10         $to = 'harshanalakmal503@gmail.com';
11         $mail_subject = 'Message from Website';
12         $email_body = "Message from Contact Us Page of the Website: <br>";
13         $email_body .= "<b>From:</b> {$fullname} <br>";
14         $email_body .= "<b>Subject:</b> {$subject} <br>";
15         $email_body .= "<b>Message:</b><br>" . nl2br(strip_tags($message));
16
17         $header = "From: {$email}\r\nContent-Type: text/html;";
18
19         $send_mail_result = mail($to, $mail_subject, $email_body, $header);
20
21         if ( $send_mail_result ) {
22             $status = '<p class="success">Message Sent Successfully.</p>';
23         } else {
24             $status = '<p class="error">Error: Message Was Not Sent.</p>';
25         }
26     }
27     ?>

```

Figure:4.22 contact us page php code

```

index.php # style.css authController.php login.php contact-us.php JS app.js
Nanasa > Contact > perfect contact > JS app.js
1  const inputs = document.querySelectorAll(".input");
2
3  function focusFunc() {
4      let parent = this.parentNode;
5      parent.classList.add("focus");
6  }
7
8  function blurFunc() {
9      let parent = this.parentNode;
10     if (this.value == "") {
11         parent.classList.remove("focus");
12     }
13 }
14
15 inputs.forEach((input) => {
16     input.addEventListener("focus", focusFunc);
17     input.addEventListener("blur", blurFunc);
18 });
19

```

Figure:4.23 contact us page javascript code

## 4.1.12 Recover password

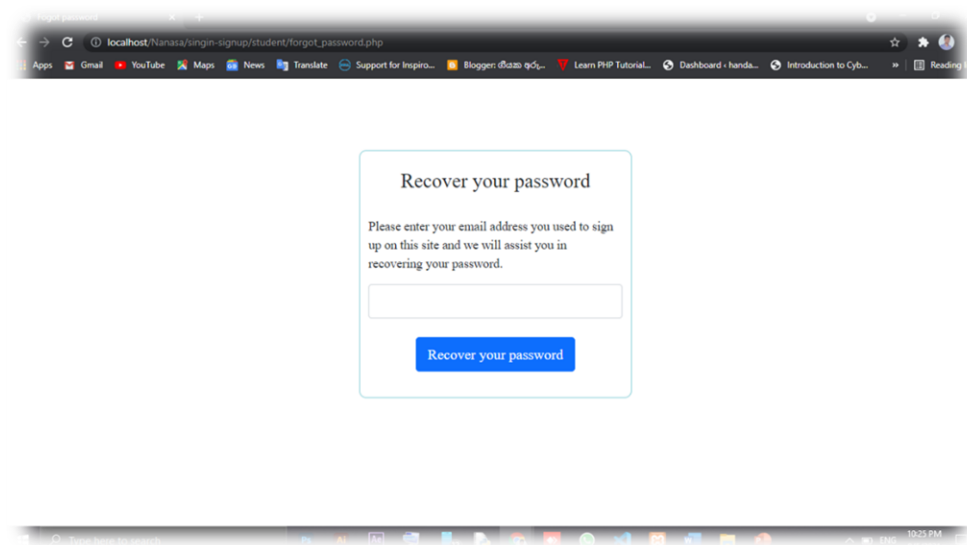


Figure:4.24 recover password

## CHAPTER 05- EVALUATION & CONCLUSION



## **5.1 Evaluation**

As we hoped to do the project, we have reached success in the following explanation. There is a handwritten process for registering students. Under this system, students can now register using their mobile device without having to go to the institute. Other advantages of this system is that there are no queues for order book due to the online system, the staff of the institution can find the student reports with one click. With the prepaid system ,the queue of cashiers also ends with the new online system. The uses of online card payment will also reduce the amount of money used with coins and paper, which will be of global value.

## **5.2 Conclusions**

- To use this online system, internet is a basic service. if there is a connection failure, whole system will be break.\
- Online services do not have a standard.
- If you have lost or forgotten your password, you should contact the administrator to retrieve it.

## **5.3 Future Improvements**

- We hope to improve this online system with fast accessibility on a fully featured hosting service.

.

## **CHAPTER 06- REFERENCE**

### **6.0 References**

## References

Guidelines, P. (2021, 07 03). *sliate*. Retrieved from <https://lms.sliate.ac.lk>

Materials, L. (2021, 7 1). *youtube*. Retrieved from <https://youtube.com>