

Mini Project Report

On

PLACEMENT MANAGEMENT SYSTEM

Submitted By:

ADITYA KUMAR SINHA

ANSHU DEO

ASHISH GAURAV

In partial fulfillment for the award of the degree of

B. TECH DEGREE

In

INFORMATION TECHNOLOGY



SCHOOL OF ENGINEERING
COCHIN UNIVERSITY OF SCIENCE & TECHNOLOGY
KOCHI-682022
MAY 2021

**Division of Information
Technology**

**School of Engineering
Cochin University of Science & Technology
Kochi-682022**

CERTIFICATE

Certified that this is a bonafide record of the project work titled

PLACEMENT MANAGEMENT SYSTEM

Done by

*ADITYA KUMAR SINHA
ANSHU DEO
ASHISH GAURAV*

*of IV semester Information Technology in the year 2021 in partial
fulfillment of the requirements for the award of Degree of Bachelor of
Technology in Information Technology of Cochin University of Science &
Technology.*

Ms Sahiya A A
Project Guide

Dr. Binsu C Kovoov
Head of the Division

ACKNOWLEDGEMENT

We take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. We extend our sincere and heartfelt thanks to our esteemed guide, **Ms Sahiya A A** for providing us with the right guidance and advice at the crucial junctures and for showing me the right way. We also take this opportunity to express a deep sense of gratitude to our class coordinator, **Mr Riyas N K** for their cordial support, valuable suggestions and guidance. We extend our sincere thanks to our respected Head of the division **Dr. Binsu C Kovoov**, 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 our friends and family for the support and encouragement they have given us during the course of our work.

TABLE OF CONTENTS

ABSTRACT	5
1. INTRODUCTION	6
1.1 PROJECT AIMS AND OBJECTIVES	6
1.2 BACKGROUND OF PROJECT	7
1.3 OPERATION ENVIRONMENT	8
2. SYSTEM ANALYSIS	9
2.1 SOFTWARE REQUIREMENT SPECIFICATION	9
2.2 EXISTING VS PROPOSED	17
2.3 SOFTWARE TOOL USED	18
3. SYSTEM DESIGN	23
3.1 TABLE DESIGN	23
3.2 DATA FLOW DIAGRAM'S	29
4. SYSTEM IMPLEMENTATION	32
4.1 MODULE DESCRIPTION	33
4.2 SCREEN SHOTS	33
5. SYSTEM TESTING	63
5.1 UNIT TESTING	64
5.2 INTEGRATION TESTING	66
6. CONCLUSION & FUTURE SCOPE	67
7. REFERENCES	68

ABSTRACT

The project is aimed at developing an application for the “WEB BASED PLACEMENT MANAGEMENT SYSTEM” of the college. The system is an application that can be accessed and effectively used throughout the organization with proper login enabled. This system can be used as an application for the Placement Officers in the college to manage the student information with regard to placement.

Student logging should be able to upload their personal and educational information. The key feature of this project is that it is one time registration enabled. Our project provides the facility of maintaining the details of the students. It reduces the manual work and consumes less paper work to reduce the time. Overall, this project of ours is being developed to help the students as well as staff of placement to maintain the process in the best way possible and also reduce the human efforts.

CHAPTER 1

INTRODUCTION

This chapter gives an overview about the aim, objectives, background and operation environment of the system.

1.1PROJECT AIMS AND OBJECTIVES

The project aims and objectives that will be achieved after completion of this project are discussed in this sub-chapter. The aims and objectives are as follows:

- Student Registration
- Company Registration
- Admin login
- Companies login page where they can add requirement details for recruitment
- Student login page where student can find companies arriving for placements and internships
- Admin can change student's status from data given by recruiters
- Previous Admins can add new admins

1.2 BACKGROUND OF PROJECT

The purpose of the project “WEB BASED PLACEMENT MANAGEMENT SYSTEM”, is - the manual work makes the process slow and other problems such as inconsistency and ambiguity on operations. In order to avoid this web based placement managed system is proposed, where the student information in the college with regard to placement is managed efficiently. It intends to help fast in fast access procedures in placement related activities and ensures to maintain the details of the student. Students logging should be able to upload their personal and educational information. The key feature of this project is that it is one time registration enabled.

The placement cell calls the companies to select their students for jobs via the campus interview. The placement cell allows the companies to view the student details in selective manner. They can filter the students profile as per their requirement. The job details of the placed students will be provided by the administrator. The administrator plays an important role in our project. Our project provides the facility of maintaining the details of the students and gets the requested list of candidates for the company who would like to recruit the students based on given query.

OPERATION ENVIRONMENT

PROCESSOR	INTEL CORE PROCESSOR OR BETTER PERFORMANCE
OPERATING SYSTEM	WINDOWS 7, WINDOWS10, UBUNTU
MEMORY	1 GB RAM OR MORE
HARD DISK SPACE	MINIMUM 3 GB FOR DATABASE USAGE FORFUTURE
DATABASE	MY SQL

CHAPTER 2

SYSTEM ANALYSIS

In this chapter, we will discuss and analyze about the developing process of Placement Management System including software requirement specification (SRS) and comparison between existing and proposed system. The functional and non-functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process is carried out. Besides that, existing vs proposed provides a view of how the proposed system will be more efficient than the existing one.

2.1 SOFTWARE REQUIREMENT SPECIFICATION

2.1.1 GENERAL DESCRIPTION

PRODUCT DESCRIPTION:

Placement Management System is a computerized system which helps admin to manage the process smoothly in electronic format. It reduces the risk of paper work such as file lost, file damaged and time consuming. It can help user to manage the transaction or record more effectively and time-saving.

PROBLEM STATEMENT:

The problem occurred before having computerized system includes:

- **File lost**

When computerized system is not implemented file is always lost because of human environment. Sometimes due to some human error there may be a loss of records.

- **File damaged**

When a computerized system is not there file is always due to some accident like spilling of water by some member on file accidentally. Besides some natural disaster like floods or fires may also damage the files.

- **Difficult to search record**

When there is no computerized system there is always a difficulty in searching of records if the records are large in number.

- **Space consuming**

After the number of records become large the space for physical storage of file and records also increases if no computerized system is implemented.

- **Cost consuming**

If there is no computerized system then to add each record paper will be needed which will increase the cost for the management of placement process.

2.1.2 SYSTEM OBJECTIVES

- **Improvement in control and performance**

The system is developed to cope up with the current issues and problems of placement. The system can add user, validate user and is also bug free.

- **Save cost**

After computerized system is implemented less human force will be required to maintain the process thus reducing the overall cost.

- **Save time**

Students will be able to search companies by using few clicks of mouse and register themselves in the companies of their choice thus saving his valuable time.

- **Companies can register remotely**

Companies have register for recruiting and coming to campus. Admins later have to approve whether they allow or not to participate companies in the placements.

- **Security**

Only admins can see the list of students placed and list of Students registered for placements. Also only previous admins can add new admins.

2.1.3 SYSTEM REQUIREMENTS

2.1.3.1 NON FUNCTIONAL REQUIREMENTS

- **Product Requirements**

- EFFICIENCY REQUIREMENT

- When the placement management system will be implemented companies and students will find it easy to interact with each other.

- RELIABILITY REQUIREMENT

- The system should accurately performs member registration, member validation, password generation, data upload and filter results.

- USABILITY REQUIREMENT

- The system is designed for a user friendly environment so that student and companies can perform the various process related tasks easily and in an effective way.

- **Organizational Requirements**

- IMPLEMENTATION REQUIREMENTS

- In implementing whole system it uses HTML in front end with PHP as server side scripting language which will be used for database connectivity and the backend and the database part is developed using MySQL.

DELIVERY REQUIREMENTS

The whole system is expected to be delivered in six months of time with a weekly evaluation by the project guide.

2.1.3.2 FUNCTIONAL REQUIREMENTS

1. NORMAL USER

1.1 USER LOGIN

Description of feature

This feature is used by the user to login into the system. They are required to enter reg. id and password before they are allowed to enter the system. The user id and password will be verified and if an invalid id is there then the user is not allowed to enter the system.

Functional requirements

- user id is provided when they register
- The system must only allow user with valid id and password to enter the system
- The system performs authorization process which decides what user level can access to.
- The user must be able to logout after they finished using system.

1.2 REGISTER NEW USER

Description of feature

This feature can be performed by all users to register new user to create account.

Functional requirements

- System must be able to verify information
- System must be able to delete information if information is wrong

1.3 REG. NEW STUDENT

This feature allows to add new student

Functional requirements

- System must be able to verify information
- System must be able to enter academic details into table.
- System must be able to not allow two students having same registration id.

1.5 PLACE STUDENTS

DESCRIPTION OF FEATURE

This admin will have to change student status to placed based on data given by Companies. He can also see details of students registered for placements and internships.

Functional requirements

- System must be able to select the data from the database based on student status
- System must be able to show the filtered companies in the companies table.

1.5 COMPANIES DETAILS ADDITION

DESCRIPTION OF FEATURE

This feature allows companies to add details of their requirements in students and their Basic details like job positing place and salary.

Functional requirements

- System must be able to enter student information in database.
- System must be able to update company information.

2.1.4 SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system

2.1.4.1 SOFTWARE REQUIREMENTS

- Operating system- Windows 10 is used as the operating system as it is stable and supports more features and is more user friendly
- Database MYSQL-MYSQL is used as database as it is easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.
- Development tools and Programming language- HTML is used to write the whole code and develop webpages with CSS, java script for styling work and php for server side scripting.

2.1.4.2 HARDWARE REQUIREMENTS

- Intel core i5 8th generation is used as a processor because it is faster than other processors and provides reliable and stable and we can run our pc for long time. By using this processor, we can keep on developing our project without any worries.
- Ram 8 GB is used as it will provide fast reading and writing capabilities and will in turn support in processing.

2.2 EXISTING VS PROPOSED SYSTEM

- i. Existing system does not have any facility for internship whereas proposed system will have a facility of internship as well.
- ii. Existing system does not have a facility of more than one admin whereas proposed system has a facility of more than one admin.
- iii. Existing system does not have any facility of adding CV whereas proposed system has a facility of adding CV.
- iv. Existing system does not have any option to deny any particular company whereas proposed system will have this facility.

2.3 SOFTWARE TOOLS USED

The whole Project is divided in two parts the front end and the back end.

2.3.1 FRONT END

The front end is designed using of html, Php, CSS, Java script

- **HTML- HTML or Hyper Text Markup Language** is the main markup language for creating web pages and other information that can be displayed in a web browser. HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like `<html>`), within the web page content. HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although some tags represent *empty elements* and so are unpaired, for example ``. The first tag in a pair is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.
- **CSS- Cascading Style Sheets (CSS)** is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation. CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve

content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied.CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called *cascade*, priorities or *weights* are calculated and assigned to rules, so that the results are predictable.

- **JAVA SCRIPT- JavaScript (JS)** is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications. JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. The application of JavaScript to use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers.

2.3.2 BACK END- The back end is designed using MySQL which is used to design the databases

- **PHP- PHP** is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for *Personal Home Page*, it now stands for *PHP: Hypertext Preprocessor*, a recursive backronym. PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.
 - **MYSQL- MySQL** ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation . MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open-source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open-source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality.
- Applications which use MySQL databases.

Placement Management System

include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

CHAPTER 3

SYSTEM DESIGN

3 . 1 TABLE DESIGN

Placement Management System

➤ STUDENT ADD DETAILS

Field	Data type	Default	Key	Extra
Student ID	INT(11)	NOT NULL	Primary key	Auto increment
Password	VARCHAR(255)	NULL		
Gender	VARCHAR(2)	NULL		
DOB	Date(yyyy/mm/dd)	NULL		
Email	VARCHAR(255)	NULL		
Address	VARCHAR(255)	NULL		
Phone	INT(10)	NULL		
Branch	VARCHAR(255)	NULL		
Tenth Marks	INT(3)	NULL		
Tenth Pass Year	INT(5)	NULL		
TwelfthMarks	INT(3)	NULL		
Twelfth Pass Year	INT(5)	NULL		
CGPA	INT(11)	NULL		
Passing Year	INT(5)	NULL		
Backlogs	INT(11)	NULL		
Apply for	VARCHAR(255)	NULL		
Status	VARCHAR(255)	NULL		
Appply Count	INT(11)	NULL		
Image	Upto 512kb	NULL		

➤ STUDENT PLACEMENT

Field	Data Type	Default	Key	Extra
Student ID	INT(11)	NOT NULL	Foreign key	References Student
Company ID	INT(11)	NOT NULL	Foreign key	References Company
Student Name	VARCHAR(255)	NULL		
Company Name	VARCHAR(255)	NULL		
Package	INT(11)	NULL		

Placement Management System

➤ STUDENT INTERNSHIP

Field	Data Type	Default	Key	Extra
Student ID	INT(11)	NOT NULL	Foreign key	References Student
Company ID	INT(11)	NOT NULL	Foreign key	References Company
Student Name	VARCHAR(255)	NULL		
Company Name	VARCHAR(255)	NULL		
Stipend	INT(11)			

➤ REGISTERED INTERNS

Field	Data type	Default	Key	Extra
Student ID	INT(11)	NOT NULL	Foreign key	References Student
Student Name	VARCHAR(255)	NULL		
Company Name	VARCHAR(255)	NULL		

Placement Management System

➤ ADMIN DETAILS

Field	Data type	Default	Key
Admin ID	INT(11)	NOT NULL	Primary Key
Admin Name	VARCHAR(255)	NULL	
Password	VARCHAR(255)	NULL	
Post	VARCHAR(255)	NULL	
Email	VARCHAR(255)	NULL	
Phone	Int(10)	NULL	
DOB	Date(yyyy/mm/dd)	NULL	
Qualification	VARCHAR(255)	NULL	

➤ REGISTERED PLACEMENTS

Field	Data Type	Default	Key	Extra
Student ID	INT(11)	NOT NULL	Foreignkey	References Student
Student Name	VARCHAR(255)	NULL		
Company	VARCHAR(255)	NULL		

Placement Management System

➤ COMPANY ADD DETAIL

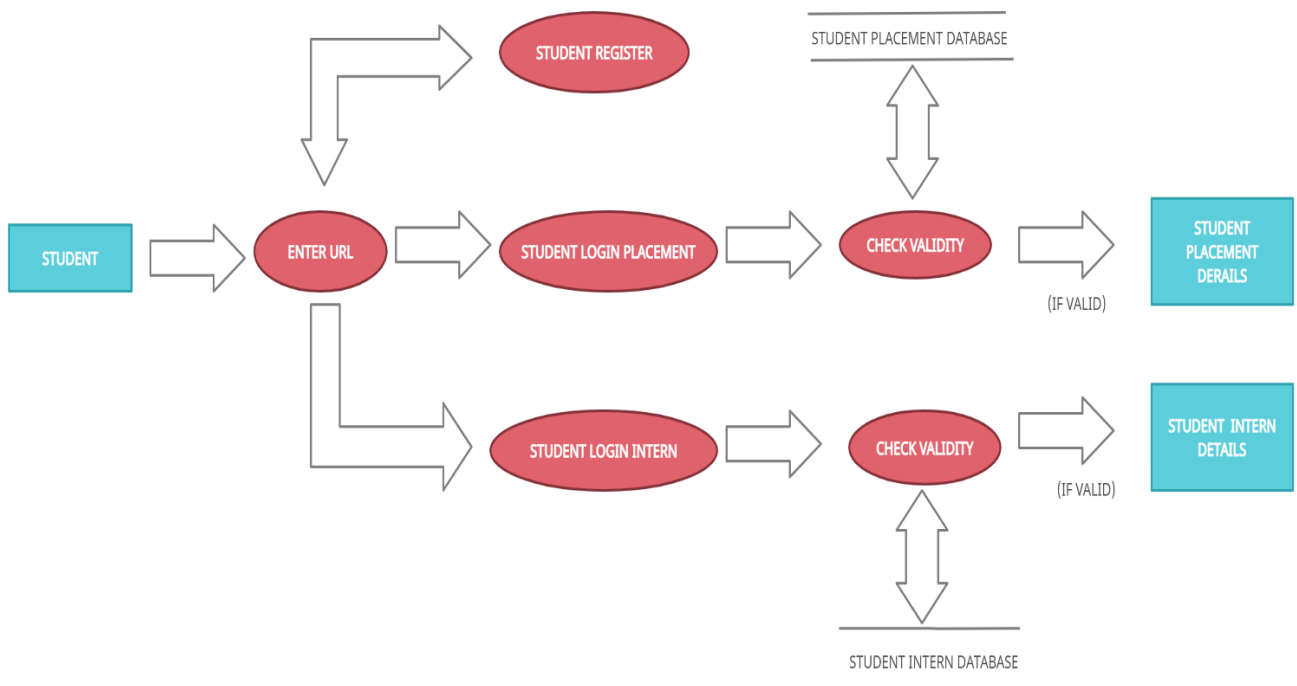
Field	Data Type	Default	Key
Company id	INT(11)	NOT NULL	PRIMARY KEY
Company name	VARCHAR(255)	NULL	
Password	VARCHAR(255)	NULL	
Website	VARCHAR(255)	NULL	
Address	VARCHAR(255)	NULL	
Status	VARCHAR(255)	NULL	
Coming Date	Date(yyyy/mm/dd)	NULL	
Approval	VARCHAR(255)	NULL	

➤ COMPANY BRANCH

Field	Data Type	Default	Key	Extra
Company Name	VARCHAR(255)	NOT NULL	FOREIGN KEY	References Company
Company Type	VARCHAR(255)	NULL		
Branch	VARCHAR(255)	NULL		
Minimum CGPA	INT(11)	NULL		
Maximum Backlogs	INT(11)	NULL		
Max Salary	INT(11)	NULL		
Max Stipend	INT(11))	NULL		
Job Profile	VARCHAR(255)	NULL		
Place of Posting	VARCHAR(255)	NULL		

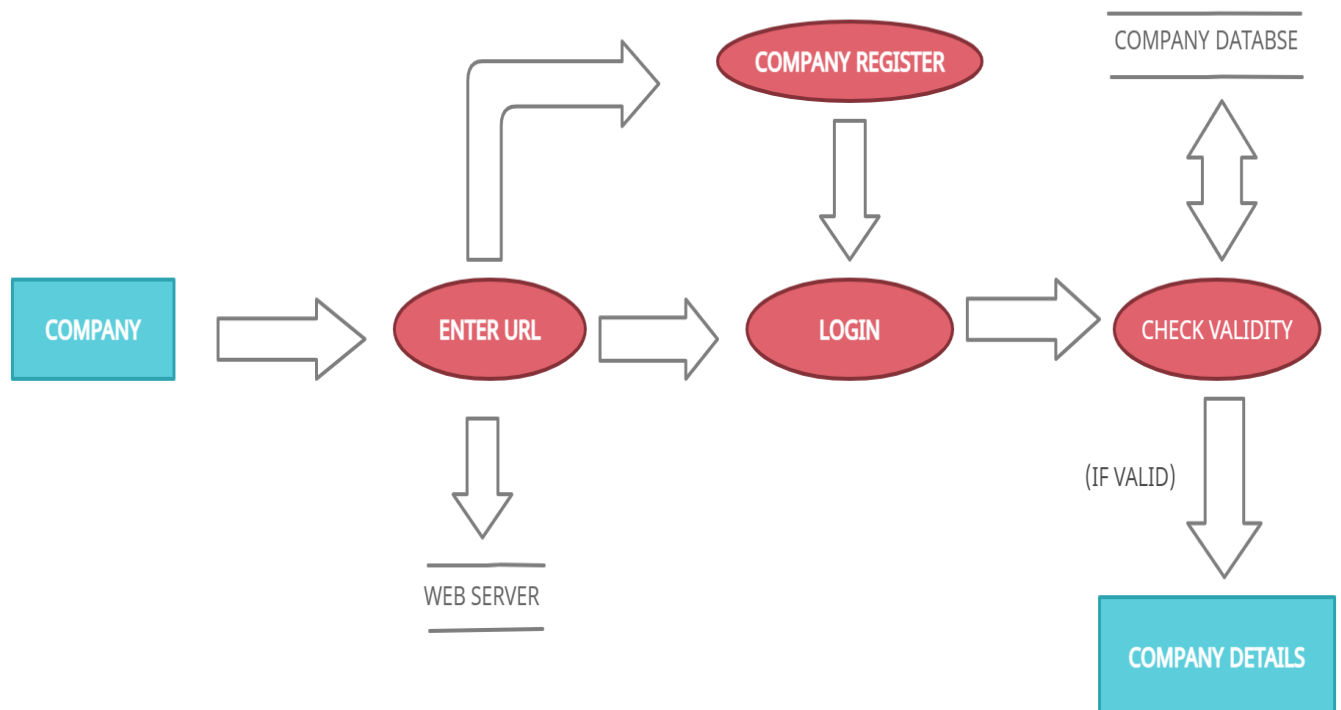
3.2 DATA FLOW DIAGRAMS

DATA FLOW DIAGRAM FOR STUDENT LOGIN



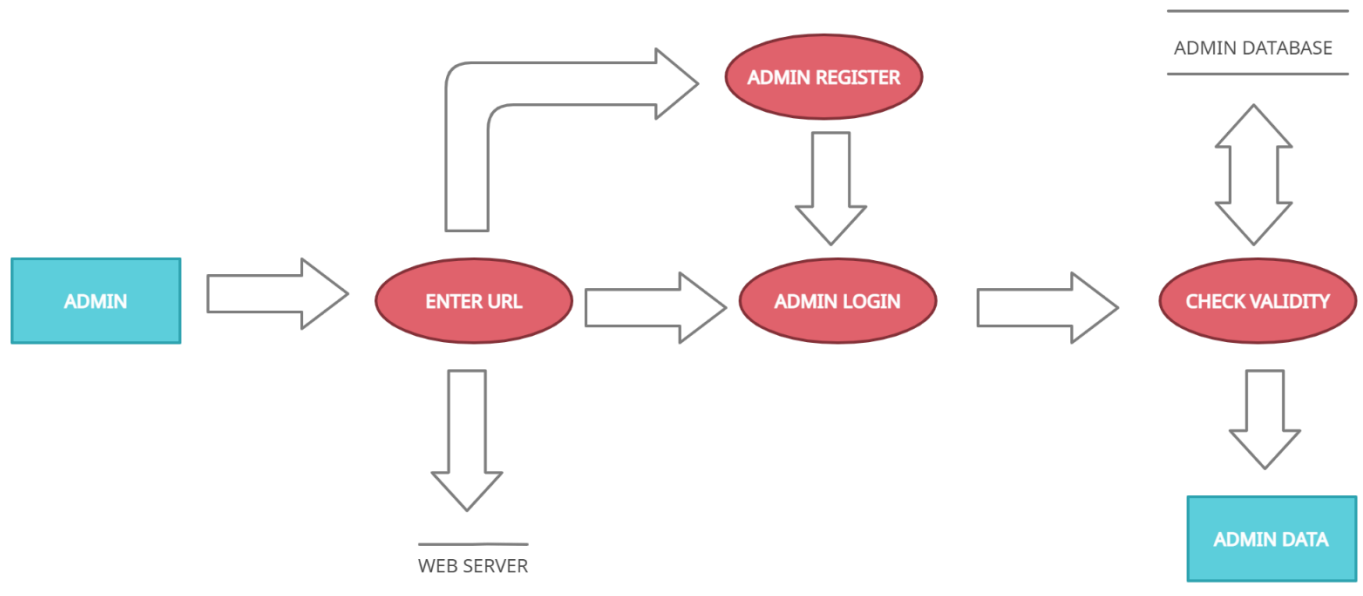
After entering to the home page of the website, STUDENTS can choose the STUDENT LOGIN option where they are asked to enter username & password, and if he/she is a valid user then STUDENT login page will be displayed.

DATA FLOW DIAGRAM FOR COMPANY LOGIN



After entering to the home page of the website, COMPANY can choose the COMPANY LOGIN option where they are asked to enter username & password, and if he/she is a valid user then COMPANY login page will be displayed.

DATA FLOW DIAGRAM FOR ADMIN LOGIN

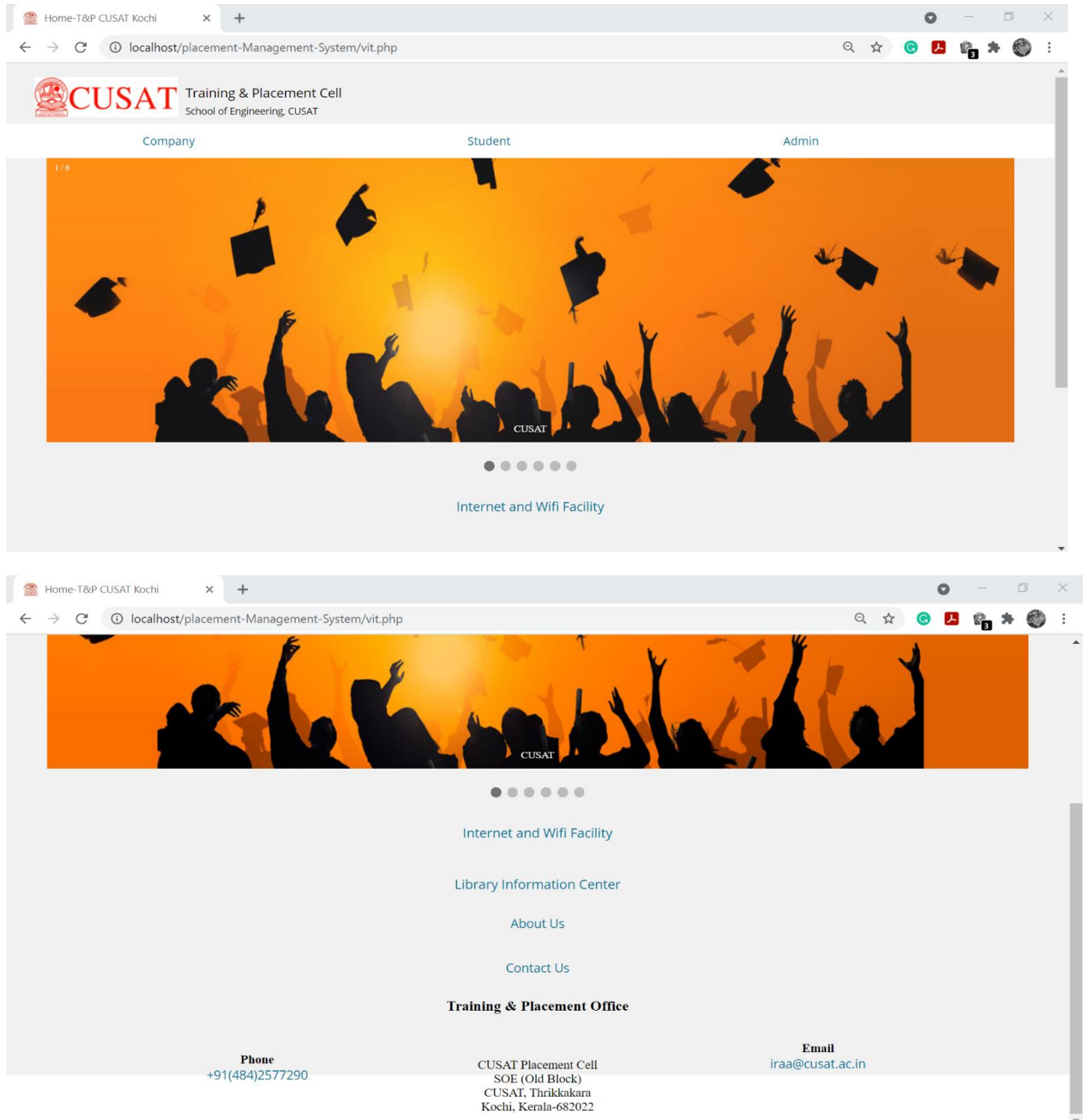


After entering to the home page of the website, ADMIN can choose the ADMIN LOGIN option where they are asked to enter username & password, and if he/she is a valid user then ADMIN login page will be displayed

CHAPTER 4

SYSTEM IMPLEMENTATION

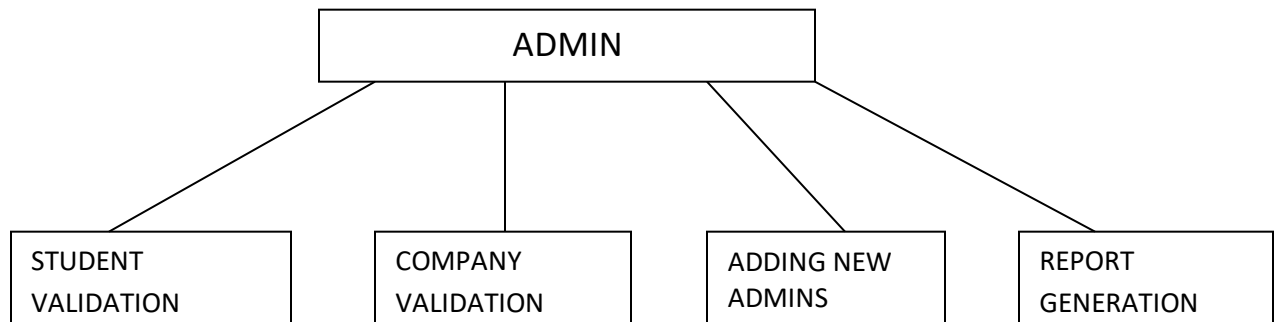
4.1.1 Screenshot for homepage



4.1 MODULE DESCRIPTION

For Placement Management System it is divided into the following Modules:

4.1.1 Admin Module



The following module contains various facilities like student validation, company registration, adding new admins, and report generation.

4.1.1.1 Code for Admin module

4.1.1.1.1 Code for entering admin username and password(login)

```
<?php
//This is for Protection
//The user with active session will be redirected to its specific index page
session_start();
if(isset($_SESSION['user']))
{
    if($_SESSION['user']=='admin')
    {
        header("Location: index_admin.php");
    }
    if($_SESSION['user']=='student_int' )
    {
        header("Location: index_student_intern.php");
    }
    if($_SESSION['user']=='student_place' )
    {
        header("Location: index_student_placement.php");
    }
    if($_SESSION['user']=='company')
    {
        header("Location: index_company.php");
    }
}
```

```
}
}
?>

<?php
$admin_id="";
$admin_name="";
$a_email="";
$post="";
$con_number="";
$dob2="";
$qualification="";
$errors = array();
$db = mysqli_connect('localhost', 'root', '', 'database1');
if (isset($_POST['login_admin']))
{
    //The mysqli_real_escape_string() function escapes special characters in a string for use in an SQL statement.
    // receive all input values from the form
    $admin_name = mysqli_real_escape_string($db, $_POST['admin_name']);
    $admin_id = mysqli_real_escape_string($db, $_POST['admin_id']);
    $admin_password = mysqli_real_escape_string($db, $_POST['admin_password']);

    // form validation: ensure that the form is correctly filled ...
    if (empty($admin_name))
    {
        array_push($errors, "Username is required");
    }
    if (empty($admin_id))
    {
        array_push($errors, "Id is required");
    }
    if (empty($admin_password))
    {
        array_push($errors, "Password is required");
    }

    if (count($errors) == 0)
    {
        //The md5() function calculates the MD5 hash of a string
        $password = md5($admin_password);
        $query = "SELECT * FROM admin WHERE ADMIN_NAME='$admin_name' AND ADMIN_ID='$admin_id' AND A_PASSWORD='$password'";
        //The mysqli_query() function performs a query against the database.
        $results = mysqli_query($db, $query);
    }
}
```

Placement Management System

```
if (mysqli_num_rows($results) == 1) ////Return the number of rows in a result set:
{
    $_SESSION['admin_id'] = $admin_id;
    $_SESSION['admin_name'] = $admin_name;
    $_SESSION['success'] = "Admin logged in";
    $_SESSION['user'] = "admin";//session variable used for protection
    header('location: index_admin.php');
}
else
{
    array_push($errors, "Wrong username/password combination");
}
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Admin login </title>
    <link rel="stylesheet" type="text/css" href="1.css">
    <link rel="shortcut icon" type="image/png" href="head_logo.jpeg">
    <link rel="stylesheet" type="text/css" href="st1.css">
    <style>
body {
    background-image: url('c6.jpg');
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: 100% 100%;
}
a:link, a:visited {
    color: #007399;
    text-decoration: none;
}

</style>
</head>
<body>
    <div class="list-unstyled3" >
        <ul>
            <li><a href="vit.php"></a></li>
            <!-- <li style
                ="float:right"><a href="logout.php">Log Out</a></li> -->
        </ul>
        <div class="para">
```

```
<p> Training & Placement, CUSAT Kochi</p>
</div>
</ul>

</div>
<div class="header" >
  <h2>Admin Login</h2>
</div>
<!--/*The PHP superglobals $_GET and $_POST are used to collect form-data.-->
<form method="post" action="admin_login.php">
  <?php include('wrong.php'); ?>
  <div class="input-group">
    <label>Admin Name</label>    <!--
The <label> element does not render as anything special for the user. However, it provides a usability improvement for mouse users, because if the user clicks on the text within the <label> element, it toggles the control.-->
    <input type="text" name="admin_name" >
  </div>
  <div class="input-group">
    <label>Admin Id</label>
    <input type="text" name="admin_id" >
  </div>
  <div class="input-group">
    <label>Password</label>
    <input type="password" name="admin_password">
  </div>

  <div class="input-group">
    <button type="submit" class="btn" name="login_admin">Sign in</button>
  </div>

  <p>
    Not enrolled yet? <a href="admin_register.php">Sign up</a>
  </p>
</form>
</body>
</html>
```

4.1.1.1 Code for company validation

```
<?php
$company_id="";
$company_name="";
$company_type="";
$website="";
$address2="";
$coming_date="";
$errors = array();
$positives=array();
$db = mysqli_connect('localhost', 'root', '', 'database1');
if (isset($_POST['approve_company']))
{
    // receive all input values from the form
    $company_name = mysqli_real_escape_string($db, $_POST['company_name']);

    if ($company_name=='Company Name')
    {
        array_push($errors, "Select company or No company with Pending Approval");
    }

    // Finally, register user if there are no errors in the form
    if (count($errors) == 0)
    {
        $query = "update company set APPROVAL='approved' where COMPANY_NAME='$company_name'";
        ;
        mysqli_query($db, $query);

        //if we want to empty all the entered details then again initialize all the$variable_n
        ame to "";

        array_push($positives, "Successfully approved the company");
        //header('Location: index_company.php');
    }
}

if (isset($_POST['disapprove_company'])) //????????????????//
{
    // receive all input values from the form
    $company_name = mysqli_real_escape_string($db, $_POST['company_name']);

    if ($company_name=='Company Name')
```

Placement Management System

```
{
    array_push($errors, "Select company or No company to reject");
}

// Finally, register user if there are no errors in the form
if (count($errors) == 0)
{
    $query = "update company set APPROVAL='rejected' where COMPANY_NAME='$company_name'";
;
    mysqli_query($db, $query);

    $query = "delete from companybranch where COMPANY_NAME='$company_name'";
    mysqli_query($db, $query);
    array_push($positives, "Successfully rejected the company");

}
}

?>

<!DOCTYPE html>
<html>
<head>
    <title>Admin Change Status of Company</title>
    <link rel="stylesheet" type="text/css" href="1.css">
    <link rel="shortcut icon" type="image/png" href="vit.jpg">
    <link rel="stylesheet" type="text/css" href="stl.css">
    <style>
body {
    background-image: url('c13.jpg');
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: 100% 100%;
}
a:link, a:visited {
    color: #007399;
    text-decoration: none;
}

</style>
</head>
<body>
    <div class="list-unstyled3" >
```

Placement Management System

```
<ul>
    <li><a href="vit.php"></a></li>
    <li style="float:right"><a href="logout.php" title="<?php echo $_SESSION['admin_name']; ?>">Log Out</a></li>

    <div class="para">
        <p> Training & Placement, CUSAT Kochi</p>
    </div>
</ul>

</div>
<div class="header">
    <h2>Company Approval</h2>
</div>
<!--/*The PHP superglobals $_GET and $_POST are used to collect form-data.-->

<form method="post" action="admin_company_approval.php">
    <?php include('wrong.php'); ?>
    <?php include('positive.php'); ?>
<div class="input-group">
    <label>Companies Visiting</label>
    <!-- <input type="text" name="gender" value="<?php echo $gender; ?>"> -->
    <select name="company_name" >
        <option>Company Name</option>
        <?php
            $db1 = mysqli_connect('localhost', 'root', '', 'database1');
            $user_check_query1 = " SELECT COMPANY_NAME FROM company where STATUS='visiting' and APPROVAL='not approved' and APPROVAL<>'rejected'";
            $result1 = mysqli_query($db1, $user_check_query1);
            //The mysqli_fetch_assoc() function fetches a result row as an associative array

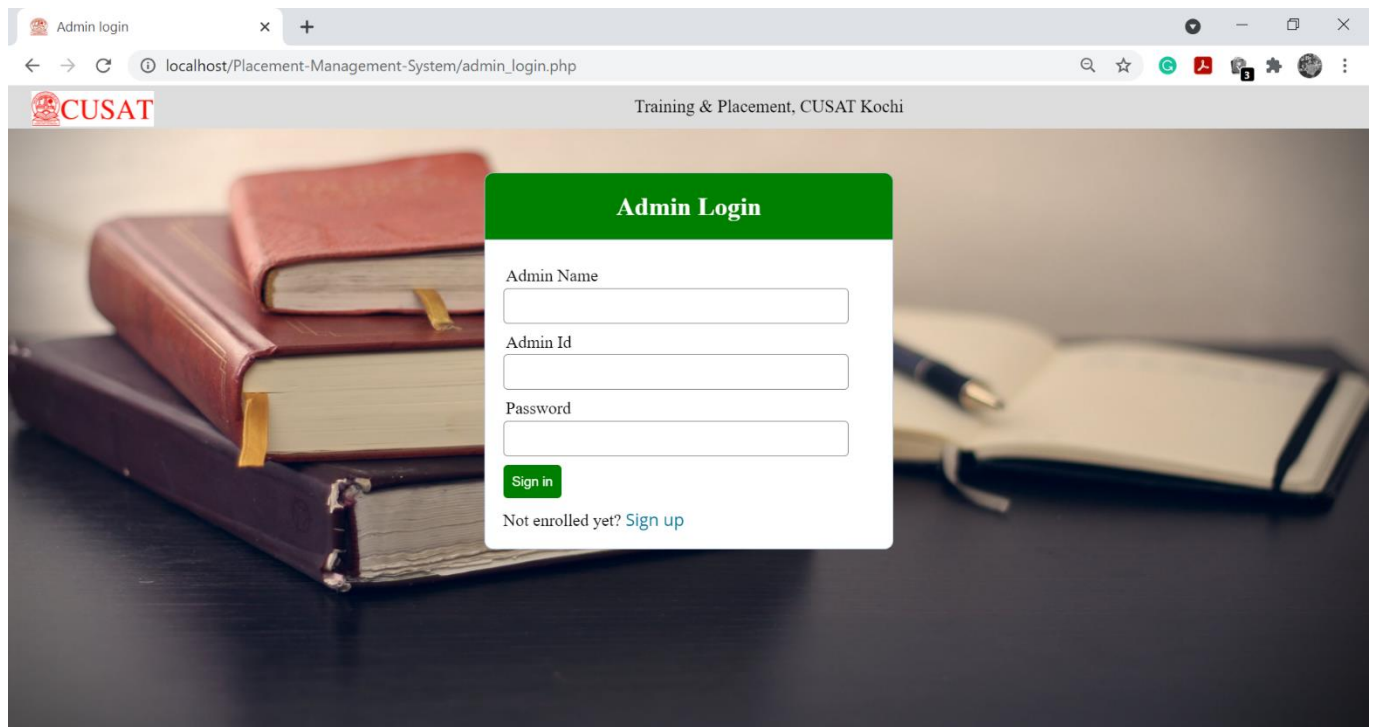
            // $count=mysqli_num_rows($result);

            while($count=mysqli_fetch_array($result1))
            {
                // $user = mysqli_fetch_assoc($result);
                // $ans=$user['COMPANY_NAME'] ;
                // $count--;
                ?>
                <option> <?php echo $count["COMPANY_NAME"]; ?> </option>
            }
        <?php
        ?>
    </div>
</form>
```

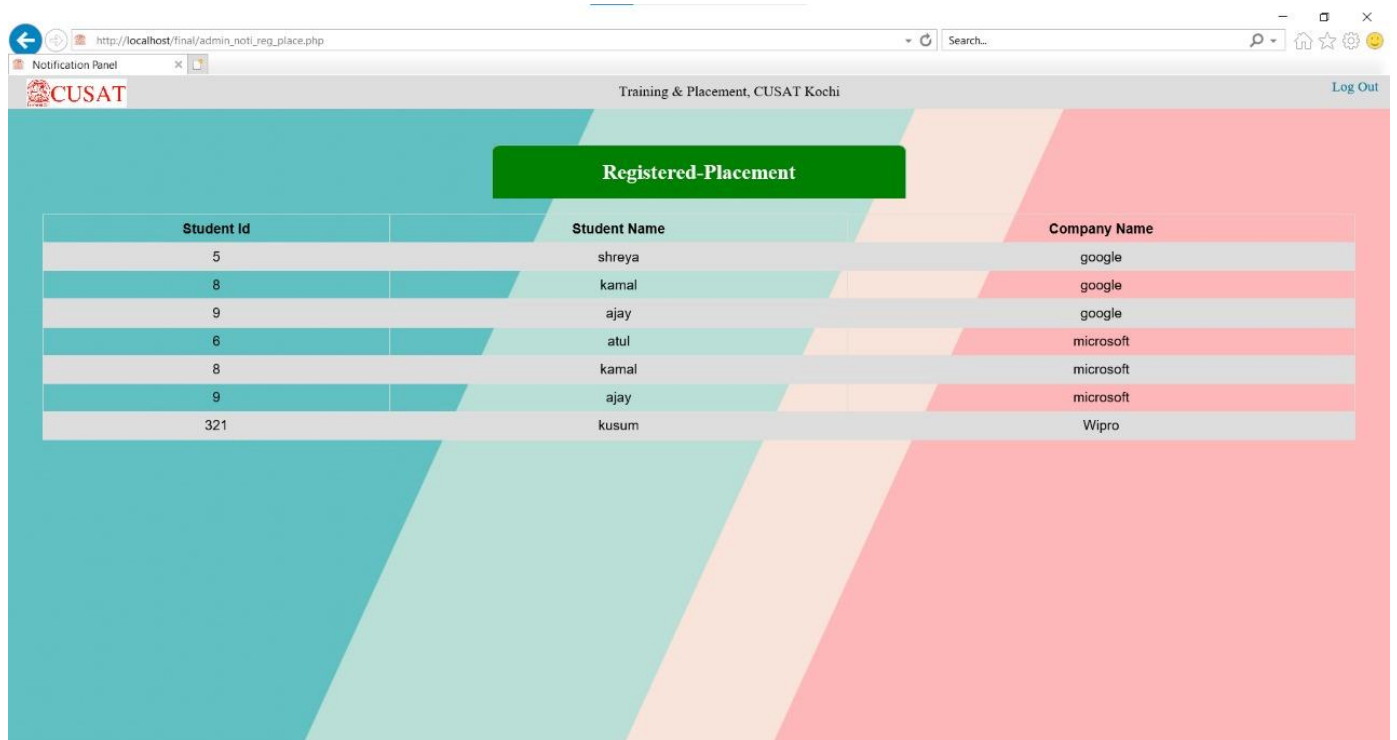
```
</select>
</div>

<div class="input-group">
  <button type="submit" class="btn" name="approve_company">Approve</button>
  <button type="submit" class="btn" name="disapprove_company">Reject</button>
</div>
<p>
  Approved the company? <a href="index_admin.php">Go Back</a>
</p>
</form>
</body>
</html>
```

4.1.1 Screenshot for Admin login



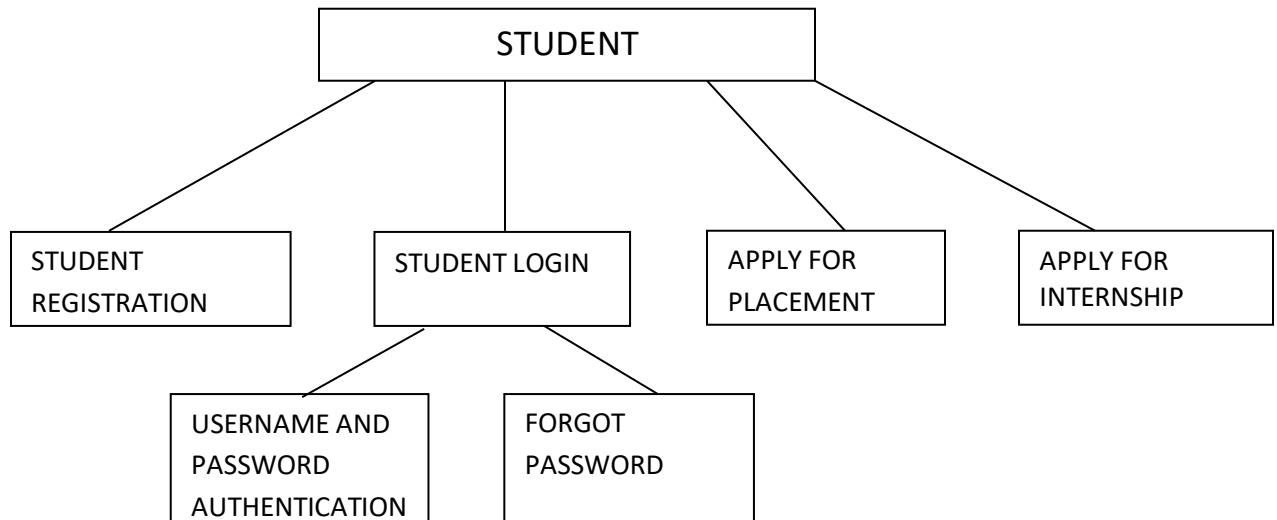
Placement Management System



The screenshot displays a web browser window with the URL `http://localhost/final/admin_noti_reg_place.php`. The page header includes the CUSAT logo and the text "Training & Placement, CUSAT Kochi". A green button labeled "Registered-Placement" is centered above a table. The table has three columns: "Student Id", "Student Name", and "Company Name". The table contains seven rows of data. The background of the page features a colorful geometric pattern with teal, light blue, orange, and pink diagonal stripes.

Student Id	Student Name	Company Name
5	shreya	google
8	kamal	google
9	ajay	google
6	atul	microsoft
8	kamal	microsoft
9	ajay	microsoft
321	kusum	Wipro

4.1.2 Student Module



The following module contains various facilities like student registration, student login. Any student if at any moment forgets his password, he can retrieve it from forgot password option.

4.1.2.1 Code For Student account creation

```
<?php
$student_id="";
$student_name="";
//$father_name="";
//$mother_name="";
$dob="";
$gender="";
$st_email="";
$address1="";
$contact_num="";
$branch="";
$tenth_per="";
$tenth_pass="";
$twelfth_per="";
$twelfth_pass="";
```

```
$cgpa="";
$pass="";
$backlogs="";
$apply="";
$st_password="";

$errors = array();
$db = mysqli_connect('localhost', 'root', '', 'database1');
if (isset($_POST['reg_student']))
{
    $student_id = mysqli_real_escape_string($db, $_POST['student_id']);
    $student_name = mysqli_real_escape_string($db, $_POST['student_name']);
    // $father_name = mysqli_real_escape_string($db, $_POST['father_name']);
    // $mother_name = mysqli_real_escape_string($db, $_POST['mother_name']);
    $dob = mysqli_real_escape_string($db, $_POST['dob']);
    $gender = mysqli_real_escape_string($db, $_POST['gender']);
    $st_email = mysqli_real_escape_string($db, $_POST['st_email']);
    $address1 = mysqli_real_escape_string($db, $_POST['address1']);
    $contact_num = mysqli_real_escape_string($db, $_POST['contact_num']);
    $branch = mysqli_real_escape_string($db, $_POST['branch']);
    $tenth_per = mysqli_real_escape_string($db, $_POST['tenth_per']);
    $tenth_pass = mysqli_real_escape_string($db, $_POST['tenth_pass']);
    $twelfth_per = mysqli_real_escape_string($db, $_POST['twelfth_per']);
    $twelfth_pass = mysqli_real_escape_string($db, $_POST['twelfth_pass']);
    $cgpa = mysqli_real_escape_string($db, $_POST['cgpa']);
    $pass = mysqli_real_escape_string($db, $_POST['pass']);
    $backlogs = mysqli_real_escape_string($db, $_POST['backlogs']);
    $apply = mysqli_real_escape_string($db, $_POST['apply']);
    $st_password1 = mysqli_real_escape_string($db, $_POST['st_password1']);
    $st_password2 = mysqli_real_escape_string($db, $_POST['st_password2']);

    //Backlogs can be empty

    if (empty($student_id))
    {
        array_push($errors, "Student Id is required");
    }
    if (empty($student_name))
    {
        array_push($errors, "Student Name is required");
    }
    /* if (empty($father_name))
    {
        array_push($errors, "Father's Name is required");
    }
    if (empty($mother_name))
```

```
{
    array_push($errors, "Mother's Name is required");
}*/
if (empty($dob))
{
    array_push($errors, "Date of birth is required");
}
if (empty($gender))
{
    array_push($errors, "Gender is required");
}
if (empty($st_email))
{
    array_push($errors, "Email is required");
}
if (empty($branch))
{
    array_push($errors, "Branch is required");
}
if (empty($tenth_per))
{
    array_push($errors, "10th % is required");
}

if (empty($tenth_pass))
{
    array_push($errors, "10th pass year is required");
}
if (empty($twelfth_per))
{
    array_push($errors, "12th % is required");
}
if (empty($twelfth_pass))
{
    array_push($errors, "12th pass year is required");
}

if (empty($cgpa))
{
    array_push($errors, "cgpa is required");
}
if (empty($pass))
{
    array_push($errors, "College passing year is required");
}
```

```
if (empty($address1))
{
    array_push($errors, "Address is required");
}
if (empty($apply))
{
    array_push($errors, "apply for field is required");
}
if (empty($st_password1))
{
    array_push($errors, "Password is required");
}
if ($st_password1 != $st_password2)
{
    array_push($errors, "passwords do not match");
}

if (filter_var($st_email, FILTER_VALIDATE_EMAIL))
{
}
else
{
    array_push($errors, "email is not a valid email address");
}

// if(($apply=='internship' or $apply=='Internship') and $pass!='2020')
// {
//     array_push($errors, "Not eligible for internship i.e not 3rd year student");
// }
// if(($apply=='placement' or $apply=='Placement') and $pass!='2019')
// {
//     array_push($errors, "Not eligible for placement i.e not 4th year student");
// }
if(strlen($contact_num)!=10)
{
    array_push($errors, "Contact Number not correct");
}
if($cgpa<1 or $cgpa>10)
{
    array_push($errors, "This CGPA not possible");
}
if($tenth_per<1 or $tenth_per>100)
{
    array_push($errors, "This 10th % not possible");
}
if($twelfth_per<1 or $twelfth_per>100)
```

```
{
    array_push($errors, "This 12th % not possible");
}
if(((int)$twelfth_pass-(int)$tenth_pass)<2)
{
    array_push($errors, "10th and 12th pass not correct");
}
$user_check_query = "SELECT * FROM student WHERE STUDENT_NAME='$student_name' AND STUDENT_ID='$student_id' LIMIT 1";
$result = mysqli_query($db, $user_check_query);
//The mysqli_fetch_assoc() function fetches a result row as an associative array.
$user = mysqli_fetch_assoc($result);

if ($user)
{ // if user exists
    if ($user['STUDENT_NAME'] === $student_name)
    {
        array_push($errors, "Student already registered");
    }
    if ($user['STUDENT_ID'] === $student_id)
    {
        array_push($errors, "Id already exists");
    }
    if ($user['EMAIL'] === $st_email)
    {
        array_push($errors, "email already exists");
    }
}

if (count($errors) == 0)
{
    $password = md5($st_password1); //encrypt the password before saving in the database
    if(isset($_FILES['image']['tmp_name']))
    {
        $file = addslashes(file_get_contents($_FILES['image']['tmp_name']));
    }
    else
    {
        // $file = null;
        //echo "not set";
    }

    $query = "INSERT INTO student (STUDENT_ID,S_PASSWORD,STUDENT_NAME,GENDER,DOB,EMAIL,ADDRESS,CONTACT_NO,BRANCH,TENTH_PER,TENTH_PASS_YEAR,TWELTH_PER,TWELTH_PASS_YEAR,CGPA,PASSING_YEAR,BACKLOGS,APPLY_FOR,IMAGE)";
```

```
VALUES('$student_id', '$password', '$student_name', '$gender', '$dob', '$st_email', '$address1', '$contact_num', '$branch', '$tenth_per', '$tenth_pass', '$twelfth_per', '$twelfth_pass', '$cgpa', '$pass', '$backlogs', '$apply', '$file');"
mysqli_query($db, $query);

//$_SESSION['student_name'] = $student_name;
$_SESSION['success'] = "Student registered successfully";
/*if(mysqli_query($db, $query))
{
    echo '<script>alert("Image Inserted into Database")</script>';
} */
header('location:student.php');
}
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Registration form for Student</title>
    <link rel="stylesheet" type="text/css" href="1.css">
    <link rel="shortcut icon" type="image/png" href="head_logo.jpeg">
    <link rel="stylesheet" type="text/css" href="st1.css">
    <style>
body {
    background-image: url('c2.jpg');
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: 100% 100%;
}
a:link, a:visited {
    color: #007399;
    text-decoration: none;
}

</style>
</head>
<body >
    <div class="list-unstyled3" >
        <ul>
            <li><a href="vit.php"></a></li>
            <!-- <li style="float:right"><a href="Logout.php">Log Out</a></li> -->

            <div class="para">
                <p> Training & Placement, CUSAT Kochi</p>
            </div>
```

```
        </ul>

    </div>
<div class="header">
    <h2>Student Registration</h2>
</div>

<form method="post" action="student_register.php" enctype="multipart/form-data">
    <?php include('wrong.php'); ?>
    <div class="input-group">
        <label>Student id</label>
        <input type="text" name="student_id" value="<?php echo $student_id; ?>">
    </div>
    <div class="input-group">
        <label>Student name</label>
        <input type="text" name="student_name" value="<?php echo $student_name; ?>">
    </div>
    <!--<div class="input-group">
        <label>Father name</label>
        <input type="text" name="father_name" value="<?php //echo $father_name; ?>">
    </div>
<div class="input-group">
        <label>Mother name</label>
        <input type="text" name="mother_name" value="<?php //echo $mother_name; ?>">
    </div> -->
    <div class="input-group">
        <label>Date of birth</label>
        <input type="date" name="dob" value="<?php echo $dob; ?>" >    <!-- yyyy-mm-dd; --
>
    </div>
<div class="input-group">
    <label>Gender</label>
    <!-- <input type="text" name="gender" value="<?php echo $gender; ?>"> -->
    <select name="gender" >
    <option value="Male">Male</option>
    <option value="Female">Female</option>
    </select>
</div>
<div class="input-group">
    <label>Email</label>
    <input type="email" name="st_email" value="<?php echo $st_email; ?>">
</div>
<div class="input-group">
    <label>Address</label>
    <input type="text" name="address1" value="<?php echo $address1; ?>">
</div>
```



```
<div class="input-group">
  <label>Contact Number</label>
  <input type="number" name="contact_num" value="<?php echo $contact_num; ?>">
</div>
<div class="input-group">
  <label>Branch</label>

  <!-- <input type="text" name="branch" value="<?php echo $branch; ?>" -->

  <select name="branch" >
    <option value="CSE">Computer Science and Engineering</option>
    <option value="IT">Information Technology Engineering</option>
    <option value="ECE">Electronics and Comm Engineering</option>
    <option value="EEE">Electrical Engineering</option>
    <option value="ME">Mechanical Engineering</option>
    <option value="CE">Civil Engineering</option>
    <option value="SFE">Safety & Fire Engineering</option>

  </select>

</div>
<div class="input-group">
  <label>10th percentage</label>
  <input type="text" name="tenth_per" value="<?php echo $tenth_per; ?>">
</div>
<div class="input-group">
  <label>10th pass year</label>
  <!--<input type="text" name="tenth_pass" value="<?php echo $tenth_pass; ?>" -->
  <select name="tenth_pass" >
    <option value="2016">2016</option>
    <option value="2017">2017</option>

  </select>
</div>
<div class="input-group">
  <label>12th percentage</label>
  <input type="text" name="twelfth_per" value="<?php echo $twelfth_per; ?>">
</div>
<div class="input-group">
  <label>12th pass year</label>
  <!--
<input type="text" name="twelfth_pass" value="<?php echo $twelfth_pass; ?>" -->
  <select name="twelfth_pass" >
    <option value="2018">2018</option>
    <option value="2019">2019</option>
  </select>
```

```
</div>
<div class="input-group">
  <label>CGPA</label>
  <input type="text" name="cgpa" value="<?php echo $cgpa; ?>">
</div>
<div class="input-group">
  <label>passing year</label>
  <!-- <input type="text" name="pass" value="<?php echo $pass; ?>" -->
  <select name="pass" >
    <option value="2012">2022</option>
    <option value="2023">2023</option>
  </select>
</div>
<div class="input-group">
  <label>Backlogs</label>
  <input type="text" name="backlogs" value="<?php echo $backlogs; ?>">
</div>
<div class="input-group">
  <label>Apply for</label>
  <!--<input type="text" name="apply" value="<?php echo $apply; ?>" -->
  <select name="apply" >
    <option value="Internship">Internship</option>
    <option value="Placement">Placement</option>
  </select>
</div>
<div class="input-group">
  <label>Password</label>
  <input type="password" name="st_password1">
</div>
<div class="input-group">
  <label>Confirm password</label>
  <input type="password" name="st_password2">
</div>
<div class="input-group">
  <label>Upload Resume(Max Size 512kb)</label>
  <input type="file" name="image" id="image" required="required"/>
</div>

<div class="input-group">
  <button type="submit" class="btn" name="reg_student" id='submit'>Sign up</button>
</div>

<p>
```

```
        Already registered? <a href="student.php">Sign in</a>
    </p>
</form>

<script>
$(document).ready(function(){
    $('#submit').click(function(){
        var image_name = $('#image').val();
        if(image_name == '')
        {
            alert("Please Select Image");
            return false;
        }
        else
        {
            var extension = $('#image').val().split('.').pop().toLowerCase();
            if(jQuery.inArray(extension, ['png','jpg','jpeg']) == -1)
            {
                alert('Invalid Image File');
                $('#image').val('');
                return false;
            }
        }
    });
});
</script>

</body>
</html>
```

4.1.2.2 Code For Student login (placement)

```
<?php

session_start();
if(isset($_SESSION['user']))
{
    if($_SESSION['user']=='admin')
    {
        header("Location: index_admin.php");
    }
    if($_SESSION['user']=='student_int' )
    {
        header("Location: index_student_intern.php");
    }
    if($_SESSION['user']=='student_place' )
    {
        header("Location: index_student_placement.php");
    }
    if($_SESSION['user']=='company')
    {
        header("Location: index_company.php");
    }
}

?>

<?php
$student_id="";
$student_name="";
$father_name="";
$mother_name="";
$dob="";
$gender="";
$st_email="";
$address1="";
$contact_num="";
$branch="";
$tenth_per="";
$tenth_pass="";
$twelfth_per="";
$twelfth_pass="";
$cgpa="";
```

```
$pass="";
$backlogs="";
$apply="";
$st_password="";
$errors = array();
$db = mysqli_connect('localhost', 'root', '', 'database1');

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

    $student_name = mysqli_real_escape_string($db, $_POST['student_name']);
    $student_id = mysqli_real_escape_string($db, $_POST['student_id']);
    $apply = mysqli_real_escape_string($db, $_POST['apply']);
    $st_password = mysqli_real_escape_string($db, $_POST['st_password']);

    if (empty($student_name))
    {
        array_push($errors, "Student Name is required");
    }
    if (empty($student_id))
    {
        array_push($errors, "Student Id is required");
    }
    if (empty($apply))
    {
        array_push($errors, "apply is required");
    }
    if (empty($st_password))
    {
        array_push($errors, "Password is required");
    }
    if(($apply!='placement' and $apply!='Placement' ))
    {
        array_push($errors, "You entered incorrect info for Apply");
    }

    if (count($errors) == 0 )
    {
        /* The md5() function calculates the MD5 hash of a string */
        $password = md5($st_password);
        $query = "SELECT * FROM student WHERE STUDENT_NAME='$student_name' AND S_PASSWORD='$password' AND STUDENT_ID='$student_id' AND APPLY_FOR='$apply'";
        /*The mysqli_query() function performs a query against the database. */
        $results = mysqli_query($db, $query);
    }
}
```

```
if (mysqli_num_rows($results) == 1 ) /////Return the number of rows in a result set:
{
    $_SESSION['student_name'] = $student_name;
    $_SESSION['success'] = "placement";
    $_SESSION['student_id'] = $student_id;
    $_SESSION['user'] = "student_place";
    header('location: index_student_placement.php');
}
else
{
    array_push($errors, "Wrong username/password combination or Not Registered yet");
}
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>student login for placement</title>
    <link rel="stylesheet" type="text/css" href="1.css">
    <link rel="shortcut icon" type="image/png" href="head_logo.jpeg">
    <link rel="stylesheet" type="text/css" href="st1.css">
    <style>
body {
    background-image: url('c2.jpg');
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: 100% 100%;
}
a:link, a:visited {
    color: #007399;
    text-decoration: none;
}

</style>
</head>
<body>
    <div class="list-unstyled3" >
        <ul>
            <li><a href="vit.php"></a></li>
            <!-- <li style="float:right"><a href="Logout.php">Log Out</a></li> -->

            <div class="para">
                <p> Training & Placement, CUSAT Kochi</p>
            </div>
        </ul>
    </div>
</body>
</html>
```

```
        </div>
        </ul>

    </div>
<div class="header">
    <h2>Student Log in for Placement</h2>
</div>
<!--/*The PHP superglobals $_GET and $_POST are used to collect form-data.-->
<form method="post" action="student_login_placement.php">
    <?php include('wrong.php'); ?>
    <div class="input-group">
        <label>Student Name</label>
        <input type="text" name="student_name" >
    </div>
    <div class="input-group">
        <label>Student Id</label>
        <input type="text" name="student_id" >
    </div>
    <div class="input-group">
        <label>Apply for</label>      <!--
The <label> element does not render as anything special for the user. However, it provides a usability improvement for mouse users, because if the user clicks on the text within the <label> element, it toggles the control.-->
        <input type="text" name="apply" value="Placement" readonly >
    </div>
    <div class="input-group">
        <label>Password</label>
        <input type="password" name="st_password">
    </div>

    <div class="input-group">
        <button type="submit" class="btn" name="login_student_place">Sign in</button>
    </div>

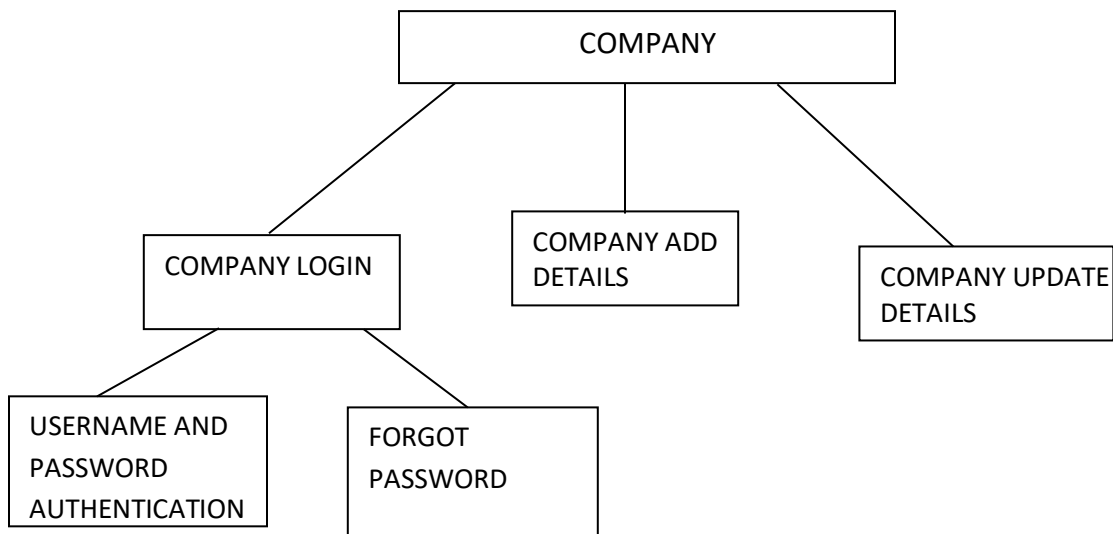
    <p>
        Not enrolled yet? <a href="student_register.php">Sign up</a>
        <br>
        <a href="">Forgot Password</a>
    </p>
</form>
</body>
</html>
```

4.1.2 Screenshot for Student login

The image shows two screenshots of a web application. The top screenshot is the 'Student Log in for Placement' page, which has a green header and a white login form. The form contains fields for Student Name, Student Id, Apply for (with a dropdown menu showing 'Placement'), and Password. Below the fields is a green 'Sign in' button and links for 'Not enrolled yet? Sign up' and 'Forgot Password'. The bottom screenshot is the 'Student Profile' page, which has a green header and a white profile card. The card displays the following information:

Student Name : Aman Zaidi
Student Id : 171
Status : -
D.O.B : 2000-06-16
Gender : Male
Email : aman11@gmail.com
Address : Delhi, India
Conatet Number : 9874785121
Branch: CSE
10th % : 91
10th pass year : 2016
12th % : 88
12th pass year : 2018
CGPA : 9
Final Year : 2012
Backlogs : 0
Apply For : Placement

- **4.1.3 Company Module**



The following module contains various facilities like company login, and adding job details. Further any company if at any moment forgets his/her password he/she can retrieve it from ‘forgot password’ option.

4.1.3.1 Code For Company Login

```
<?php
$company_id="";
$company_name="";
$company_type="";
$website="";
$address2="";
$coming_date="";
$errors = array();
$db = mysqli_connect('localhost', 'root', '', 'database1');
if (isset($_POST['login_company']))
{
```

```
$company_name = mysqli_real_escape_string($db, $_POST['company_name']);
$company_id = mysqli_real_escape_string($db, $_POST['company_id']);
$c_password = mysqli_real_escape_string($db, $_POST['c_password']);
if (empty($company_name))
{
    array_push($errors, "Company name is required");
}
if (empty($company_id))
{
    array_push($errors, "Company Id is required");
}
if (empty($c_password))
{
    array_push($errors, "Password is required");
}
if (count($errors) == 0)
{
    $password = md5($c_password);
    $query = "SELECT * FROM company WHERE COMPANY_NAME='$company_name' AND C_PASSWORD='$password' AND COMPANY_ID='$company_id'";
    //The mysqli_query() function performs a query against the database.
    $results = mysqli_query($db, $query);
    $user = mysqli_fetch_assoc($results);
    if($user['APPROVAL']=='rejected')
    {
        array_push($errors, "Company is not valid");
    }
    else
    {
        if (mysqli_num_rows($results) == 1) /////Return the number of rows in a result set:
        {
            $_SESSION['company_name'] = $company_name;
            $_SESSION['company_id'] = $company_id;
            $_SESSION['user'] = "company";
            $_SESSION['success'] = "Company logged in";
            header('location:index_company.php');
        }
        else
        {
            array_push($errors, "Wrong username/password combination");
        }
    }
}
}
```

```
?>
<!DOCTYPE html>
<html>
<head>
  <title>Company Login</title>
  <link rel="stylesheet" type="text/css" href="1.css">
  <link rel="shortcut icon" type="image/png" href="head_logo.jpeg">
  <link rel="stylesheet" type="text/css" href="st1.css">
  <style>
body {
  background-image: url('c1.jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: 100% 100%;
}
a:link, a:visited {
  color: #007399;
  text-decoration: none;
}
</style>
</head>
<body>
  <div class="list-unstyled3" >
    <ul>
      <li><a href="vit.php"></a></li>
      <!-- <li style="float:right"><a href="Logout.php">Log Out</a></li> -->

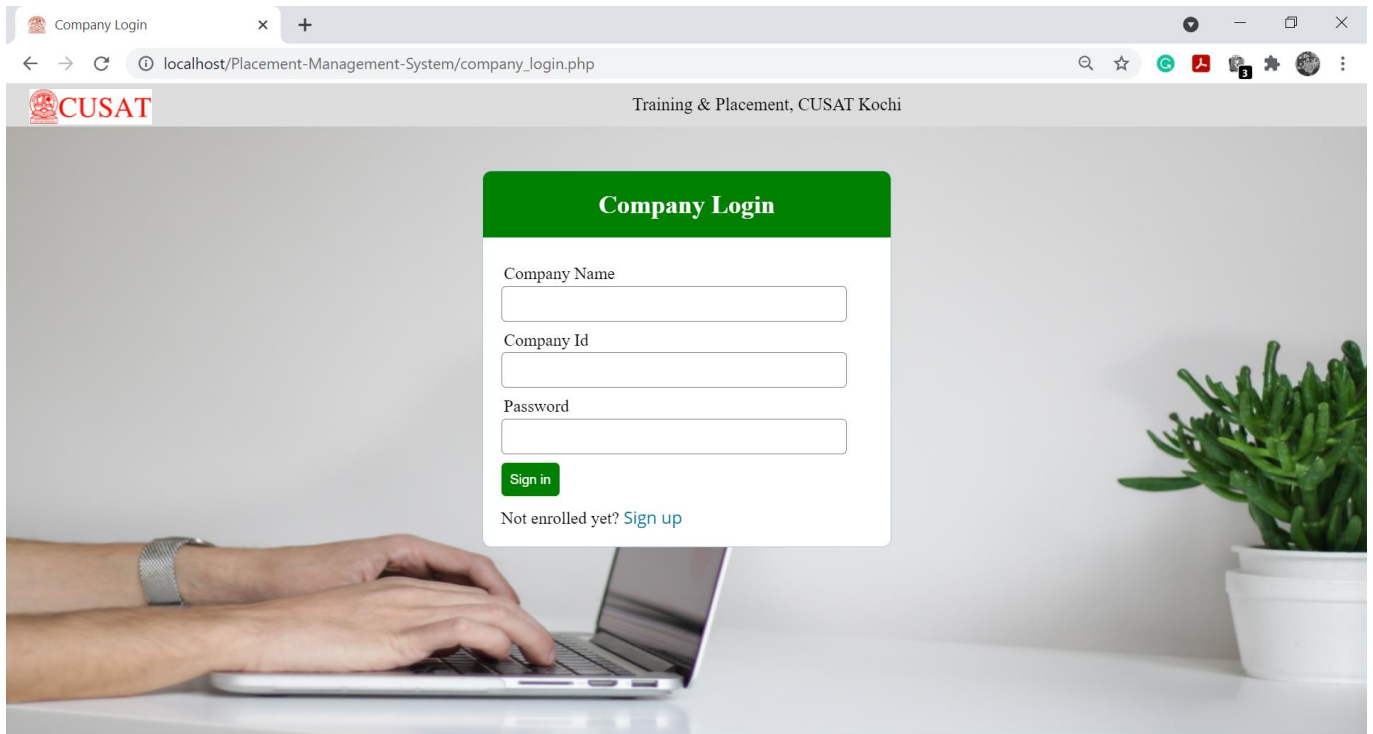
      <div class="para">
        <p> Training & Placement, CUSAT Kochi</p>
      </div>
    </ul>

  </div>
  <div class="header">
    <h2>Company Login</h2>
  </div>
  <!--/*The PHP superglobals $_GET and $_POST are used to collect form-data.-->
  <form method="post" action="company_login.php">
    <?php include('wrong.php'); ?>
    <div class="input-group">
      <label>Company Name</label> <!--
The <label> element does not render as anything special for the user. However, it provides a usability improvement for mouse users, because if the user clicks on the text within the <label> element, it toggles the control.-->
      <input type="text" name="company_name" >
```

```
</div>
<div class="input-group">
  <label>Company Id</label>    <!--
The <label> element does not render as anything special for the user. However, it provides a usability improvement for mouse users, because if the user clicks on the text within the <label> element, it toggles the control.-->
  <input type="text" name="company_id" >
</div>
<div class="input-group">
  <label>Password</label>
  <input type="password" name="c_password">
</div>
<div class="input-group">
  <button type="submit" class="btn" name="login_company">Sign in</button>
</div>

<p>
  Not enrolled yet? <a href="company_register.php">Sign up</a>
</p>
</form>
</body>
</html>
```

4.1.3 Screenshot for company's login & company's details



The screenshot shows a web browser window with the title "Company Login". The address bar displays "localhost/Placement-Management-System/company_login.php". The page header includes the CUSAT logo and the text "Training & Placement, CUSAT Kochi". The main content area features a "Company Login" form with a green header. The form contains three input fields: "Company Name", "Company Id", and "Password". Below these fields is a green "Sign in" button and a link "Not enrolled yet? Sign up". The background of the page shows a person's hands typing on a laptop keyboard next to a potted plant.

Company Login

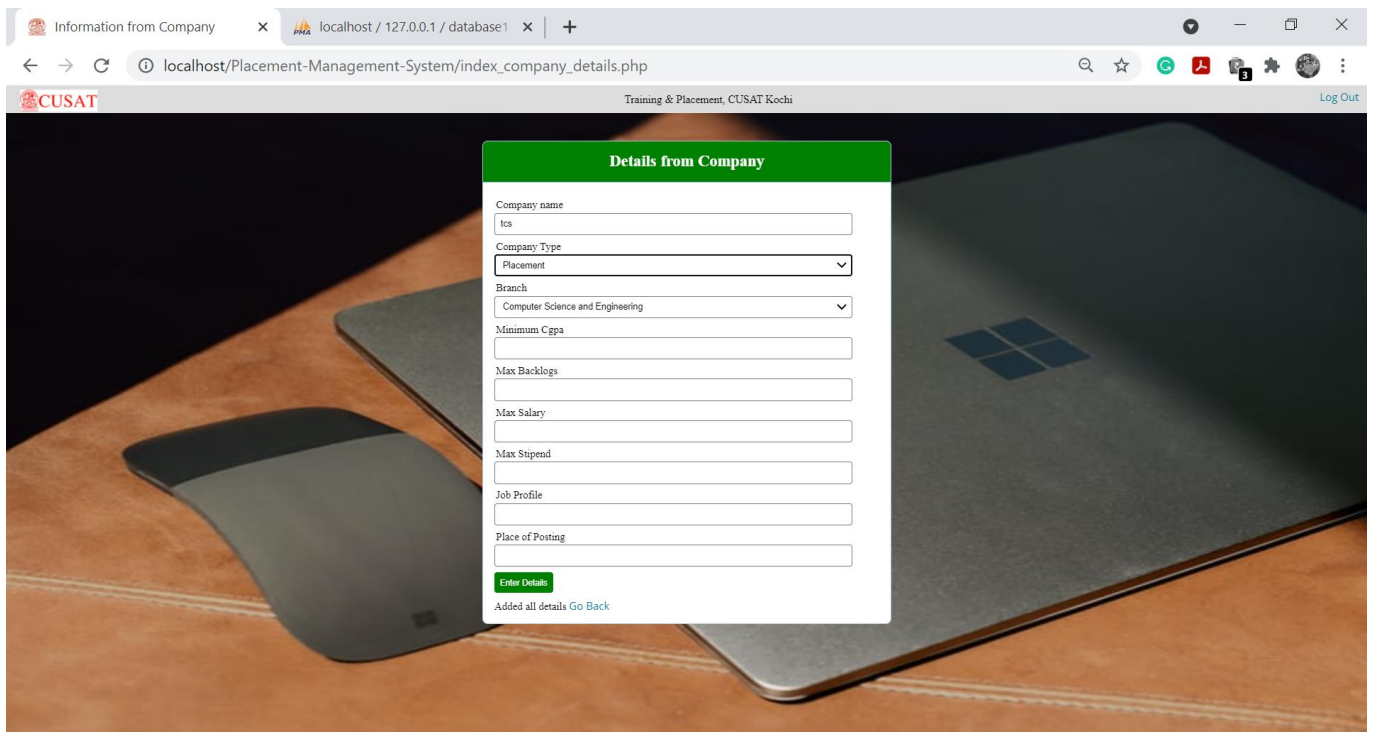
Company Name

Company Id

Password

Sign in

Not enrolled yet? Sign up



The screenshot shows a web browser window with the title "Information from Company". The address bar displays "localhost / 127.0.0.1 / database1" and "localhost/Placement-Management-System/index_company_details.php". The page header includes the CUSAT logo and the text "Training & Placement, CUSAT Kochi". The main content area features a "Details from Company" form with a green header. The form contains several input fields and dropdown menus: "Company name" (with "Its" entered), "Company Type" (with "Placement" selected), "Branch" (with "Computer Science and Engineering" selected), "Minimum Cgpa", "Max Backlogs", "Max Salary", "Max Stipend", "Job Profile", and "Place of Posting". Below these fields is a green "Enter Details" button and a link "Added all details Go Back". The background of the page shows a laptop and a mouse on a wooden desk.

Details from Company

Company name

Its

Company Type

Placement

Branch

Computer Science and Engineering

Minimum Cgpa

Max Backlogs

Max Salary

Max Stipend

Job Profile

Place of Posting

Enter Details

Added all details Go Back

All Project Code

Posted

On

GitHub

<https://github.com/adityakumarsinha03/Placement-Management-System>

CHAPTER 5

SYSTEM TESTING

The aim of the system testing process was to determine all defects in our project. The program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not.

Our Project went through two levels of testing:

1. Unit testing
2. Integration testing

UNIT TESTING

Unit testing is undertaken when a module has been created and successfully reviewed. In order to test a single module we need to provide a complete environment i.e. besides the module we would require

- The procedures belonging to other modules that the module under test calls
- Non local data structures that module accesses
- A procedure to call the functions of the module under test with appropriate parameters

Unit testing was done on each and every module that is described under module description of chapter 4.

1. Test For the admin module

- Testing admin login form- This form is used for log in of administrator of the system. In this we enter the username and password if both are correct administration page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password.
- Student account addition- In this section the admin can verify student details from student academics info and then only add student details to main placement database it contains add and delete buttons if user click add button data will be added to student database and if he clicks delete button the student data will be deleted.

2. Test for Student login module

- Test for Student login Form-This form is used for log in of Student. In this we enter the username and password if all these are correct student login page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password.
- Test for account creation- This form is used for new account creation when student does not fill the form completely it asks again to fill the whole form when he fills the form fully it gets redirected to login page.

3. Test for Company login module-

- Test for Company login form- This form is used for log in of company. In this we enter the username and password if all these are correct company login page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password.

INTEGRATION TESTING

In this type of testing, we test various integration of the project module by providing the input. The primary objective is to test the module interfaces in order to ensure that no errors are occurring when one module invokes the other module.

CHAPTER 6

CONCLUSION & FUTURE SCOPE

This website provides a computerized version of placement management system which will benefit the students as well as the staff of the college. It makes entire process online where student can enter details, staff can generate reports and company can apply online. It also has a facility for student login where student can login and can see list the companies coming as well as companies in which they are applicable to apply. We can store information of all students. Various companies can access their information. It has a security such that only previous admins can add new admin. Admin can see the list of students registered for placements and internships and also the list of students successfully placed.

There is a future scope of this facility that many more features such as placement study materials, coding problem tutorials can be added by teachers as well as students can give mock tests online. A feature of group chat where students can discuss various issues regarding placements can be added to this project thus making it more interactive and more user friendly which fulfills each user need in the best way possible.

CHAPTER 7

REFERENCES

- http://www.w3schools.com/html/html_intro.asp
- http://www.w3schools.com/css/css_background.asp
- http://www.w3schools.com/js/js_datatypes.asp
- http://www.w3schools.com/sql/sql_insert.asp
- http://www.w3schools.com/sql/sql_update.asp
- http://www.w3schools.com/php/php_forms.asp
- **Fundamentals of software engineering** by **Rajib mall**, PHI learning
- **Web development and application development** by **Ivan Byross**, BPB publications