

MARTIN LUTHER CHRISTIAN UNIVERSITY



“Development of Training and Placement Management System”

By

Aochiba Jamir

Regd No:B160484

Project work submitted to the
Computer Sciences Department, MLCU
in partial fulfilment of the requirements
for the award of the degree
Master of Computer Applications (MCA)

Year of submission

2018

DEPARTMENT OF COMPUTER SCIENCES

NONGRAH, DONGKTIEH, BLOCK-1, SHILLONG, MEGHALAYA

File No. – VI/Csc/PROJECT/2016/1-

MARTIN LUTHER CHRISTIAN UNIVERSITY

NONGRAH, DONGKTIEH, BLOCK-1, SHILLONG, MEGHALAYA

DEPARTMENT OF COMPUTER SCIENCES



CERTIFICATE

This is to certify that the project entitled
**“Development of Training and Placement Management
System”**

Submitted in partial fulfilment of the requirement for the award of the
degree of

MASTERS OF COMPUTER APPLICATIONS

Of

MARTIN LUTHER CHRISTIAN UNIVERSITY

is a result of benefited work carried out by

Aochiba Jamir
Regd No:B160484

During the academic year (2016-2018)

Examiner(s):
(Name and signature)

Head,
Computer Sciences

Department

- 1.
- 2.



COMMUNITY INSTITUTE OF TECHNOLOGY

Achieving Excellence Together

Govt. Register No. SR/CIT-B/13 of 2013

Hopkinson Road, Lower Lachumiere, Shillong
East Khasi Hills District, Meghalaya-793001

CIT

Date: 19th June 2018

ORGANISATION CERTIFICATE

This is certify that **Mr. Aochiba Jamir, Enrollment no B160484, course MCA,** Year 2016-2018 a student of Martin Luther Christian University, Meghalaya has successful completed the project title “Development of Training and Placement Management System” from 14-02-2018 to 15-06-2018 during the tenure of their project training, their conduct and contribution have been very good.

We wish them all the best and success.

(Signature

)

ACKNOWLEDGEMENT

This project consumed huge amount of work, research and dedication. Still, implementation would not have been possible if i did not have the support of many individuals and organizations. Therefore i would like to extend my sincere gratitude to all of them.

First of all I am thankful to “Community Institute of Technology” for giving me an opportunity to work in their institute for my project and also for their support and for providing necessary guidance concerning my project’s implementation.

I am also grateful to them for provision of expertise, and technical support in the implementation. Without their superior knowledge and experience, the Project would lack in quality of outcomes, and thus their support has been essential.

I would also like to express my sincere thanks to the faculty of Computer Sciences Department of Martin Luther Christian University who devoted their time and knowledge in the implementation of this project.

Nevertheless, i express my gratitude toward my family and colleagues for their kind co-operation and encouragement which helped me in completion of this project.

DECLARATION

I hereby declare that the project work entitled “**Development of Training and Placement Management System**” is a record of an original work done by me under the guidance of Sir Lenin, Head of Computer Sciences Department, Martin Luther Christian University, and this project work is submitted in the partial fulfilment of the requirements for the award of the degree of Masters in Computer Application (MCA). The results embodied in this project have not been submitted to any other University or Institute for the award of any degree or diploma.

Date:

Aochiba Jamir

CONTENTS

TITLE	PAGE NO
1. About the Project.....	2
1.1 Introduction.....	2
1.2 Abstract.....	3
1.3 Title of the Project.....	4
1.4 Aim of the Project.....	5
1.5 Draw Back of Existing System.....	6
2. Feasibility Study.....	7
2.1 Introduction.....	7
2.2 Economic Feasibility.....	7
2.3 Technical Feasibility.....	7
2.4 Behavioural Feasibility.....	7
3. Project Planning and Schedule.....	8
3.1 Software Process Model.....	8
3.2 Gantt Chart.....	9
4. System Requirement Specification (SRS).....	10
4.1 Hardware Requirements.....	10
4.2 Software Requirements.....	10
5. Software Design Description (SDD).....	11
5.1 Data Flow Diagram (DFD).....	11
5.2 ER Diagram (Data Dictionary, Table Structure).....	12-18
5.3 Screen Shots.....	19-22
6. Software Test Documents.....	23
6.1 Test Approach-Unit Testing, System or Integration Testing.....	23
6.2 Test Cases.....	23-27
7. Source Code.....	28-45
8. Conclusion.....	46
9. References.....	47

1. About the Project

1.1 Introduction

The project titled “Training and Placement Management System” has been developed for various training and placement management centres. It is a web-based application and is developed using PHP as the front end and MYSQL as the back end.

The main purpose of this project is to tackle the hardships/workload faced in the manual system. It will be very user-friendly and hence no prior knowledge is needed for the user to use this system. The “Training and Placement Management System”, will lead to error free, secure, reliable and fast management system. Thus, helping various organizations to utilize and manipulate their resources more efficiently.

1.2 Abstract

The “Training and Placement Management System” is a computerised management system with four levels of users i.e. Admin, Centre, Employer and Student. In the admin level, the admin will have control over the entire system. The admin can allocate or de-allocate employers, update or add new courses. Also add or update centres. The centre level will have access only to information related to that particular centre. They can update their centre’s details, add assessors, verify students and list students for placement. In the student and employer level, one can register and register themselves to a particular centre. They can login to the system once registered and can check the list of placements.

DEVELOPMENT OF TRAINING AND PLACEMENT MANAGEMENT SYSTEM

1.4 Aim of the Project

The main aim of the project is to manage the details of Training, Placement, Company, Students, and Registration by automating the existing manual system with the help of computers, so that valuable data can be stored for a longer period with easy accessing and manipulation of the same.

1.5 Drawback of Existing System

The existing system for managing the data is done manually by hand and is maintained in paper. This system adds difficulty to users in verifying records and it is a difficult and long task to update information. Also in case of viewing or retrieving information, a user has to run through large number of records to look in the details of a particular student, which results in waste of time. There is also the possibility of losing/misplacing data if maintained in paper.

2. Feasibility Study

2.1 Introduction

A feasibility study is an analysis of the viability of an idea through a disciplined and documented process of thinking through the idea from its logical beginning to its logical end. Feasibility study is carried out in order to assess the viability of a project.

2.2 Economic Feasibility

- There was no economical cost in developing this project since it was developed for academic purpose.
- It requires no maintenance cost.

2.3 Technical Feasibility

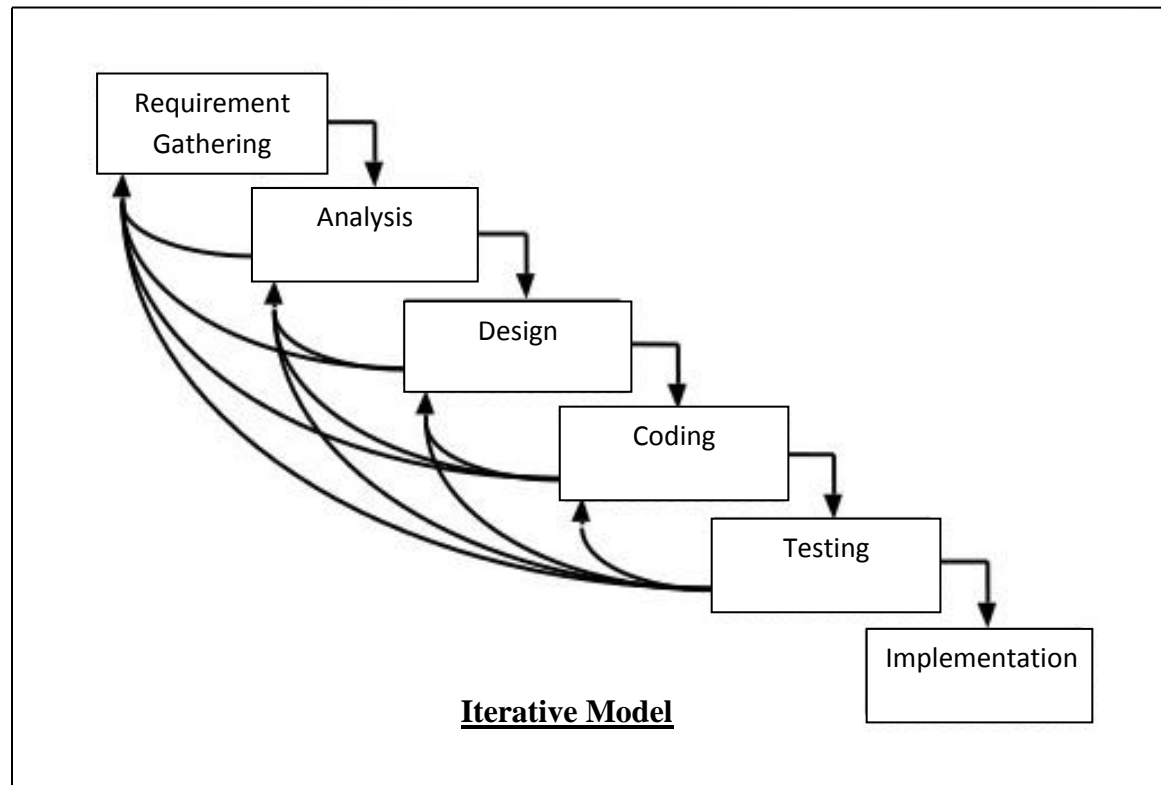
- It will run on any version of windows.
- It is light and does not consume much space.
- It does not require any extra software or application to run.

2.4 Behavioural Feasibility

- It has a user friendly interface. There will be no problem for new users.
- No special skills are required for operating the system.

3. Project Planning and Schedule

3.1 Software Process Model

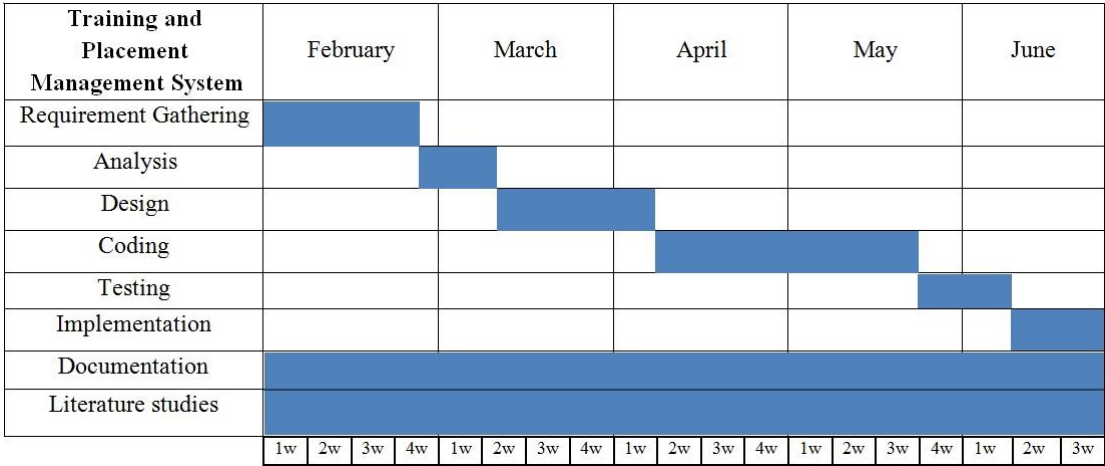


Training and Placement Management System was developed using Iterative Model.

In the Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.

An iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements. This process is then repeated, producing a new version of the software at the end of iteration of the model.

3.2 Gantt Chart



4. System Requirement Specification (SRS)

4.1 Hardware Requirement

- Processor: Pentium processors and higher
- RAM: 512mb minimum
- Hard Disk: 1Gb minimum

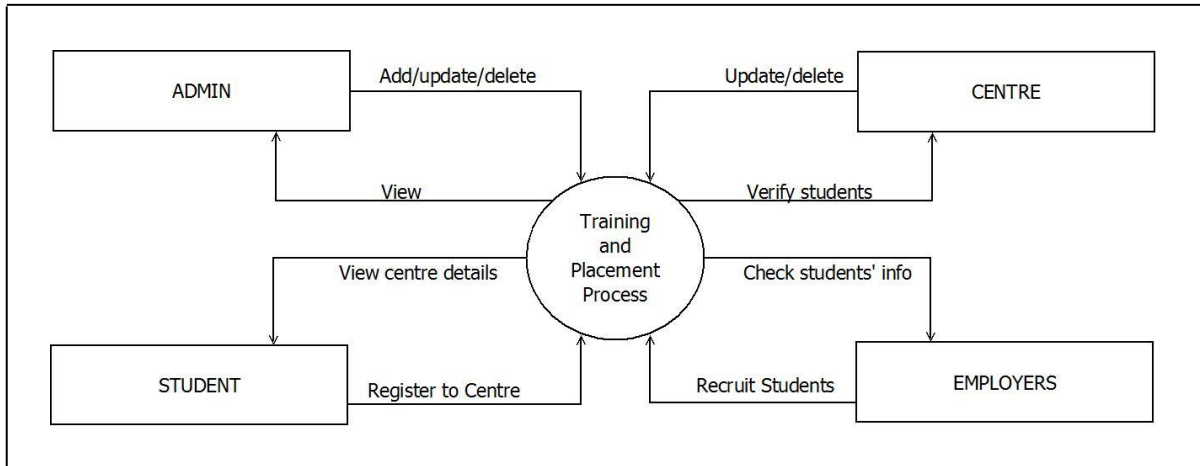
4.2 Software Requirement

- Server: XAMPP
- Languages: Php (Front end)
- Operating System: Windows 7 and above
- Database: MYSQL (Back end)

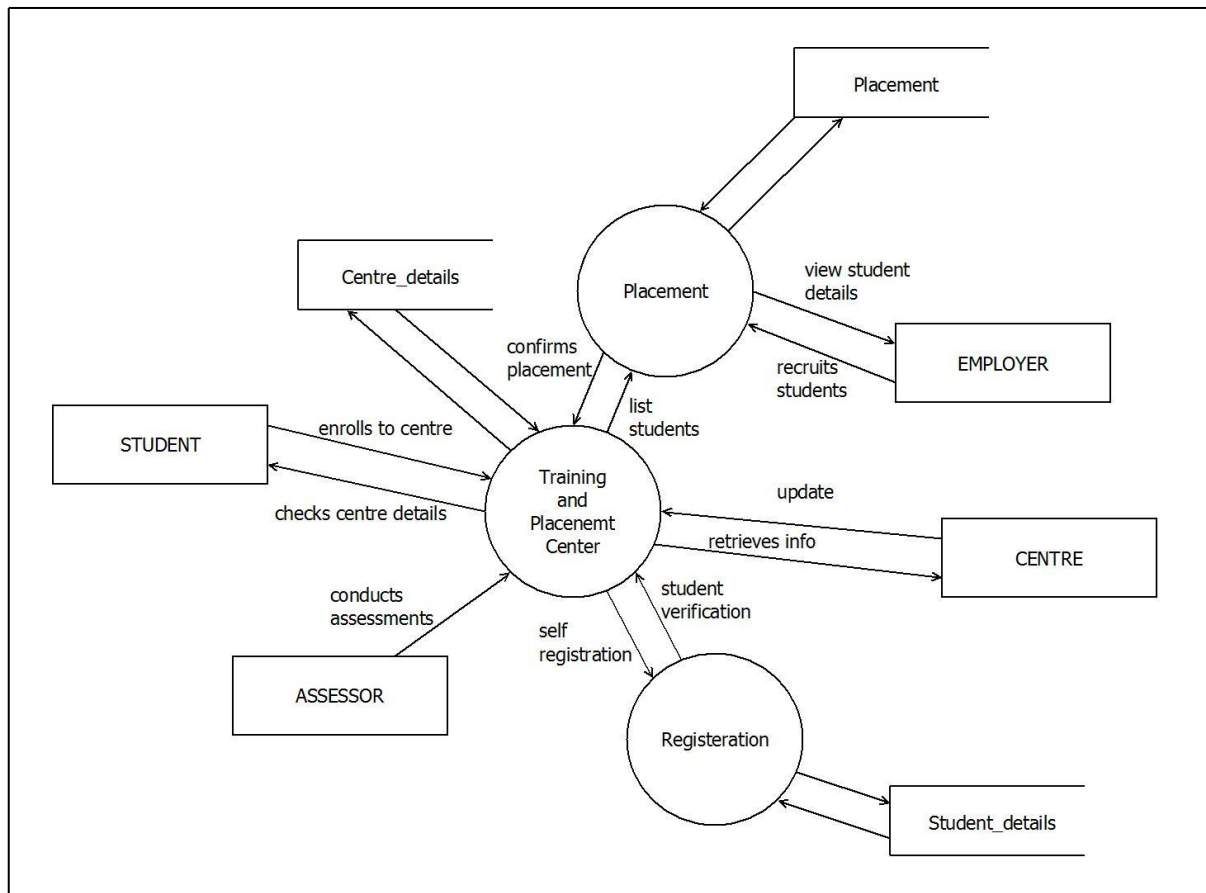
5. Software Design Description (SDD)

5.1 Data Flow Diagram (DFD)

Level 0 DFD



Level 1 DFD



5.2 ER Diagram

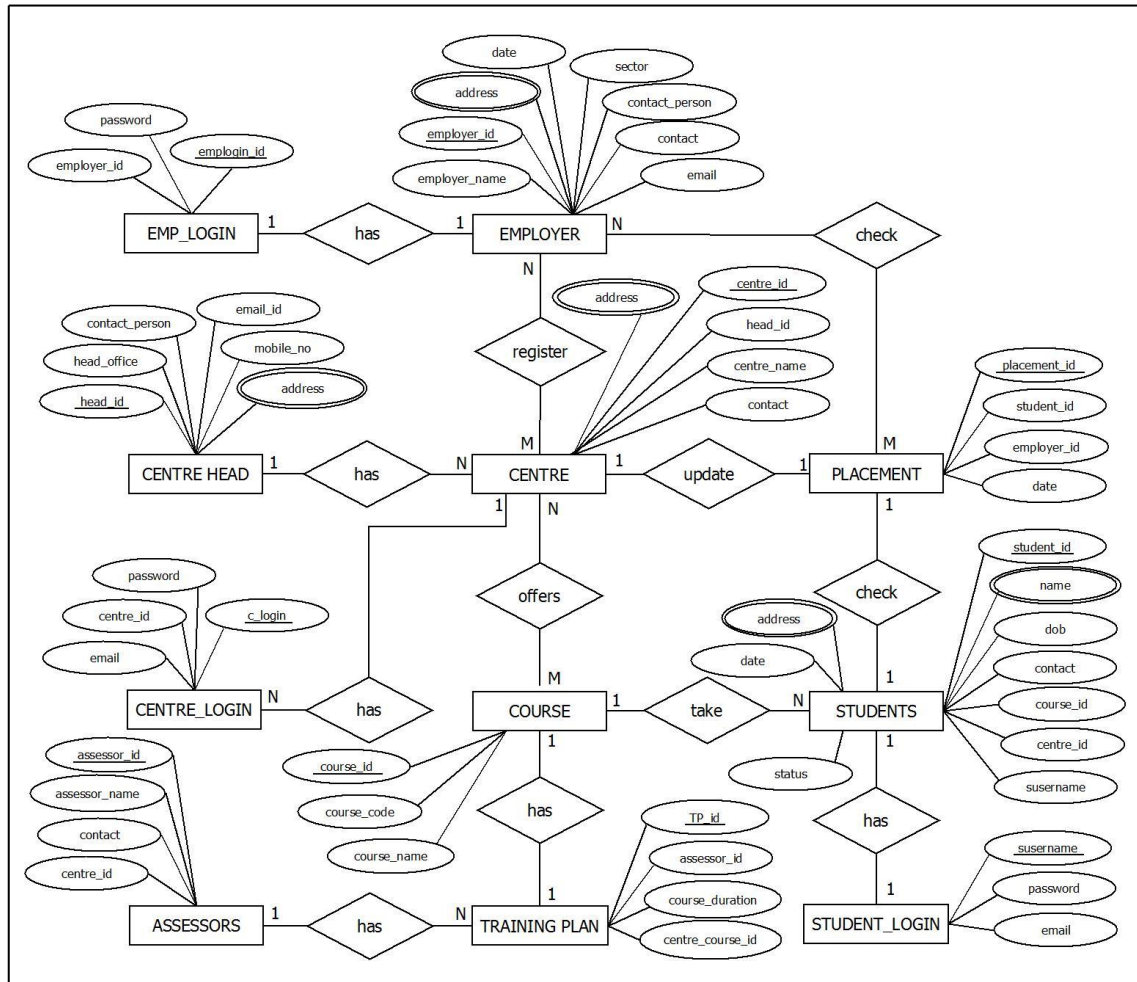
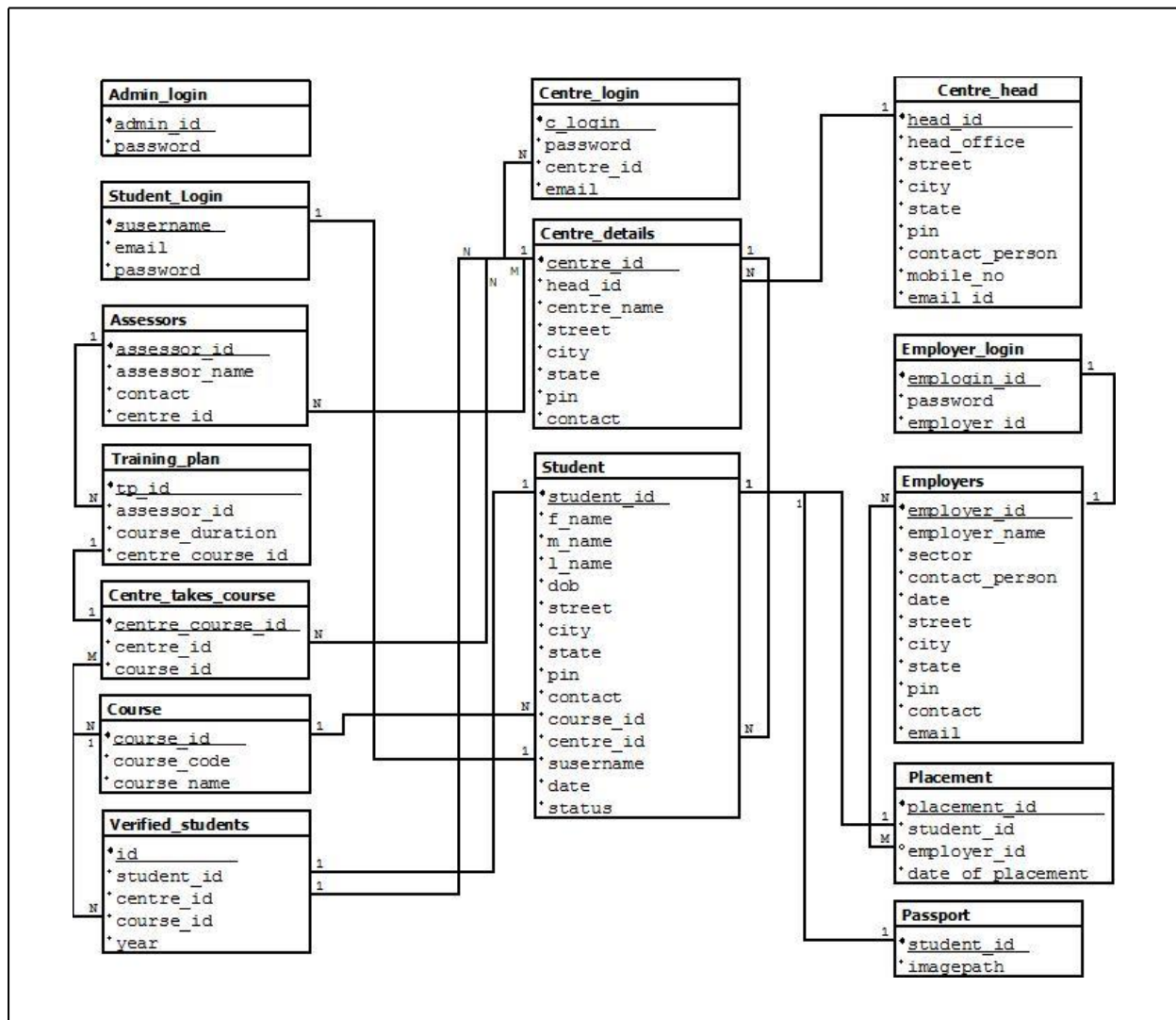


Table Structure



Data Dictionary

Table 1: Admin_Login

Column	Type	Null	Default	Comments
<u>admin_id</u>	varchar(50)	No		Login id for admin
password	varchar(150)	No		Password for admin

Table 2: Assessors

Column	Type	Null	Default	Links to	Comments
<u>assessor_id</u>	int(11)	No			Assessor ID
assessor_name	varchar(50)	No			Name of assessor
contact	varchar(100)	No			Contact number of assessor
centre_id	int(11)	No		centre_details -> centre_id	Centre ID

Table 3: Centre_details

Column	Type	Null	Default	Links to	Comments
<u>centre_id</u>	int(11)	No			Centre ID
head_id	int(11)	No		centre_head -> head_id	Centre Head ID
centre_name	varchar(50)	No			Name of the centre
street	varchar(50)	No			Address
city	varchar(50)	No			Address
state	varchar(50)	No			Address
pin	int(11)	No			Pin code
contact	varchar(100)	No			Contact no. of the centre

Table 4: Centre_head

Column	Type	Null	Default	Comments
<u>head_id</u>	int(11)	No		Head Centre ID
head_office	varchar(50)	No		Name of Head office
street	varchar(50)	No		Address
city	varchar(50)	No		Address
state	varchar(50)	No		Address
pin	int(11)	No		Pin code
contact_person	varchar(50)	No		Name of Head in charge of the office
mobile_no	varchar(100)	No		Contact no. of the office
email_id	varchar(50)	No		Email ID of the office

Table 5: Centre_login

Column	Type	Null	Default	Links to	Comments
<u>c_login</u>	varchar(50)	No			Login ID for centre
password	varchar(150)	No			Password for centre
centre_id	int(11)	No		centre_details -> centre_id	Centre ID
email	varchar(50)	No			Email address of the centre

Table 6: Centre_takes_course

Column	Type	Null	Default	Links to	Comments
<u>centre_course_id</u>	int(11)	No			ID for course offered by centre
centre_id	int(11)	No		centre_details -> centre_id	Centre ID
course_id	int(11)	No		course -> course_id	Course ID

Table 7: Course

Column	Type	Null	Default	Comments
<u>course_id</u>	int(11)	No		ID for course
course_code	varchar(10)	No		Code name for course
course_name	varchar(50)	No		Name of course

Table 8: Employer_login

Column	Type	Null	Default	Links to	Comments
<u>employlogin_id</u>	varchar(50)	No			Login ID for the employer
password	varchar(200)	No			Password for employer login
employer_id	int(11)	No		employers -> employer_id	Employer's ID

Table 9: Employers

Column	Type	Null	Default	Comments
<u>employer_id</u>	int(11)	No		ID for employer/company
employer_name	varchar(100)	No		Name of employer/company
sector	varchar(50)	No		Sector of the company
contact_person	varchar(50)	No		Name of the manager of the company
date	date	No		Date of registration
street	varchar(100)	No		Address
city	varchar(100)	No		Address
state	varchar(100)	No		Address
pin	int(7)	No		Pin code
contact	varchar(100)	No		Contact no. of the company
email	varchar(100)	No		Email ID of the company

Table 10: Passport

Column	Type	Null	Default	Links to	Comments
<u>student_id</u>	int(11)	No		student -> student_id	Student ID
imagepath	varchar(200)	No			Image path for student's passport

Table 11: Placement

Column	Type	Null	Default	Links to	Comments
<u>placement_id</u>	int(11)	No			ID for placement
student_id	int(11)	No		student -> student_id	Student's ID
employer_id	int(11)	No		employers -> employer_id	Employer's ID
date_of_placement	date	No			Date for the placement/interview

Table 12: Student

Column	Type	Null	Default	Links to	Comments
<u>student_id</u>	int(11)	No			ID for student
f_name	varchar(20)	No			First name of the student
m_name	varchar(20)	No			Middle name of the student
l_name	varchar(20)	No			Last name of the student
dob	date	No			Date of birth of the student
street	varchar(50)	No			Address
city	varchar(50)	No			Address
state	varchar(50)	No			Address
pin	int(7)	No			Pin code
contact	varchar(100)	No			Contact no. of the student
course_id	int(11)	No		course -> course_id	Course ID
centre_id	int(11)	No		centre_details -> centre_id	Centre ID
susename	varchar(50)	No		student_login -> susename	Username of the student
date	date	No			Date of registration
status	varchar(25)	No			Status for application form

Table 13: Student_login

Column	Type	Null	Default	Comments
<u>susename</u>	varchar(50)	No		Username for student login
email	varchar(50)	No		Email address of the student
password	varchar(200)	No		Password for student login

Table 14: Training_plan

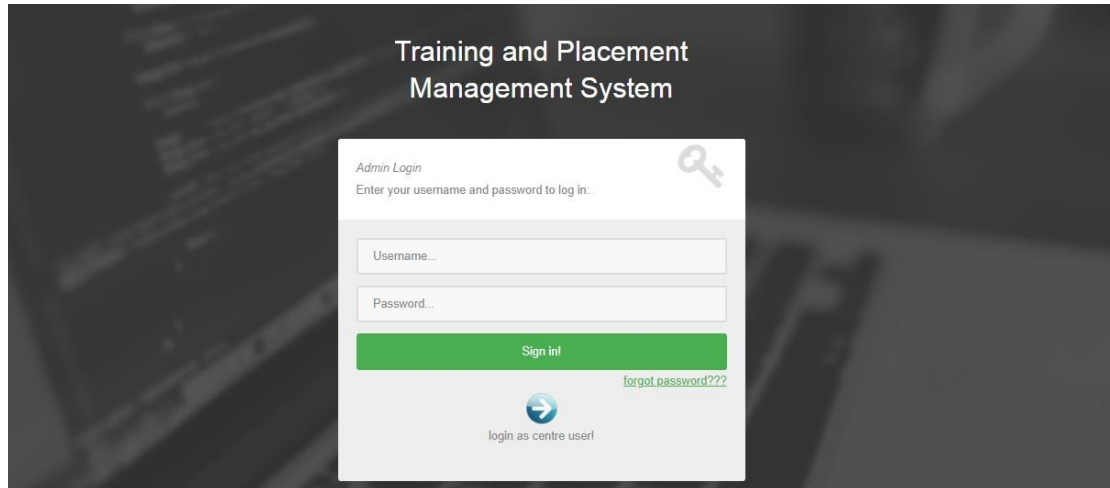
Column	Type	Null	Default	Links to	Comments
<u>TP_id</u>	int(11)	No			ID for training plan
assessor_id	int(11)	No		assessors -> assessor_id	Assessor's ID
course_duration	varchar(25)	No			Duration of the course
centre_course_id	int(11)	No		centre_takes_course -> centre_course_id	ID of course offered by centre

Table 15: Verified_students

Column	Type	Null	Default	Links to	Comments
<u>id</u>	int(11)	No			ID for verified student's application form
student_id	int(11)	No		student -> student_id	Student's ID
centre_id	int(11)	No		centre_details -> centre_id	Centre ID
course_id	int(11)	No		course -> course_id	Course ID
year	year(4)	No			Year of joining centre

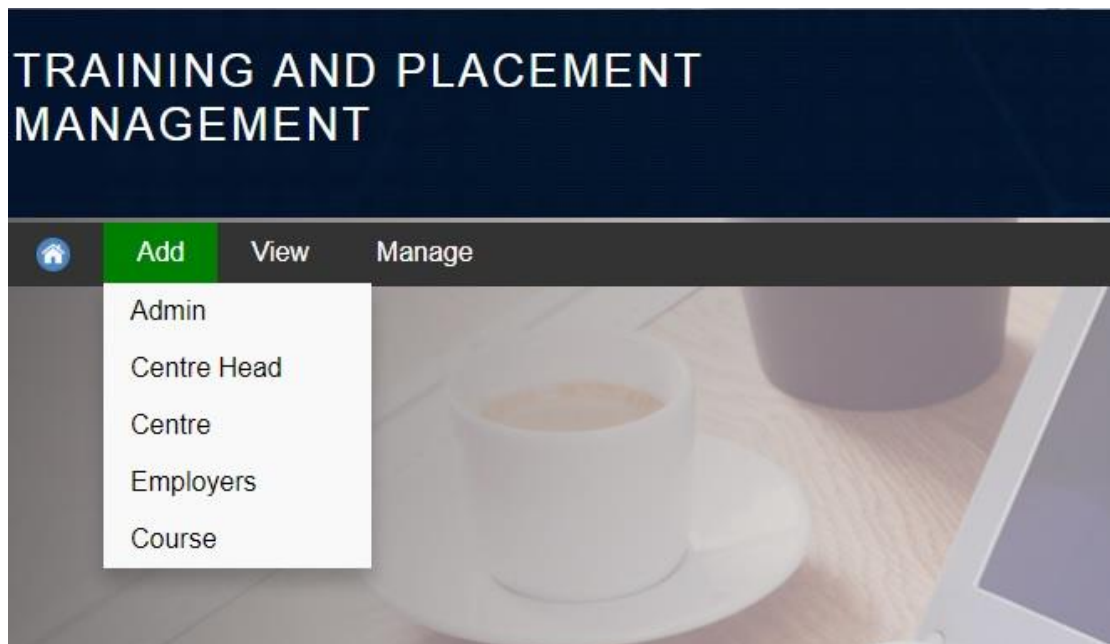
5.3 Screen Shots

Admin Login



The screenshot shows the 'Admin Login' interface for the 'Training and Placement Management System'. The background is a dark, blurred image of a laptop keyboard. The login form is a white box with a light gray border. At the top right of the form is a key icon. The text 'Admin Login' is in a small font, followed by the instruction 'Enter your username and password to log in:'. Below this are two input fields: 'Username...' and 'Password...'. A green 'Sign in!' button is positioned below the password field. To the right of the button is a link that says 'forgot password???'. At the bottom of the form is a circular arrow icon and the text 'login as centre user!'. The entire form is centered on the screen.

Admin Page



Centre User Page

TRAINING AND PLACEMENT
MANAGEMENT

QWE(CENTRE)

Centre Details

Assessors

Students

Placement

logout

Centre name:

MLCU

Centre Head name:

a

Address:

Goralane, Shillong, Meghalaya, 793001

Contact:

9999999999

Course Offered:

Add course

Course id	Course code	Course name	Duration	Assessor	
6	pgdca	asdasd	1year	yellow	
1	MCA	master in computer application	3month	Orange	

Add Course

TRAINING AND PLACEMENT
MANAGEMENT

QWE(CENTRE)

Centre Details

Assessors

Students

Placement

logout

Centre name:

MLCU

Centre Head name:

a

Address:

Goralane, Shillong, Meghalaya, 793001

Add course

Select course

---select course---

Duration

---select duration---

Select assessor

---select assessor---

update

Add/Update and Delete Assessors

TRAINING AND PLACEMENT
MANAGEMENT

QWE(CENTRE)

Centre Details

Assessors

Students

Placement

logout

Assessor ID	Assessor Name	Contact		
3	red	4444442342		
10	green	3453453453		
11	yellow	1231231231		
12	Orange	1231231231		

Add assessors

Student Verification

TRAINING AND PLACEMENT
MANAGEMENT

QWE(CENTRE)

Centre Details
Assessors
Students
Placement
logout

APPLICATION FORM



Selected Centre	MLCU
Selected Course	asdasd
First name	iii
Middle name	iii
Last name	iii
Date of birth	2018-06-15
Street	iii
City	iii
State	iii
Pin	123123
Contact	1231231231

accept

reject

go back

List Students for Placement

TRAINING AND PLACEMENT
MANAGEMENT

QWE(CENTRE)

Centre Details
Assessors
Students
Placement
logout

List students for placement

Name of student	Employer's name	Date	Action
iii iii	ZXCZXCZXC	2018-06-15	delete

[Add students](#)

Home Page

TRAINING AND PLACEMENT
MANAGEMENT

HOME CENTRES ADMISSION

LOGIN | SIGNUP



Let's **Work** Together
and **Explore** Opportunities...



100% job placement guarenteed!!!
Signup today and be a part of it

Centres

Admission/Enrollment

Student Signup

TRAINING AND PLACEMENT
MANAGEMENT

HOME CENTRES ADMISSION

LOGIN | SIGNUP

Username

Enter user name ...

Email ID

Enter email id....

Password

Enter password....

Confirm password

Enter email id....

Sign Up

Already have an account??
[Click here](#) to Login

Student Register

TRAINING AND PLACEMENT
MANAGEMENT

HOME CENTRES ADMISSION

XCV

APPLICATION FORM

Select Centre*

:

---Select a Centre---

▼

Select Course*

:

---Select a Course---

▼

First name*

:

Middle name*

:

Last name*

:

Date of birth*

:

dd-mm-yyyy

Street*

:

City*

:

State*

:

Pin*

:

Contact*

:

upload passport: Choose File No file chosen

submit | cancel

Employer Register

TRAINING AND PLACEMENT
MANAGEMENT

HOME CENTRES ADMISSION

LOGIN | SIGNUP

Name of company/employer

Sector

Contact person's name

Email

Date [dd-mm-yyyy]

Street

City

State

Pincode

Mobile number

Register

6. Software Test Document

6.1 Test Approach-Unit Testing, System or Integration Testing

Unit testing

In unit testing, each module is test separately so that errors can be detected at early phase. Unit testing emphasizes the verification effort on the smallest unit of software design i.e. a software component or module. Unit testing is a dynamic method for verification, where program is actually compiled and executed. Unit testing is performed in parallel with the coding phase. Unit testing tests units or modules not the whole software.

Integration Testing

In integration testing, the basic functional unit of the system were integrated and testing was done to see if there were no erroneous results. During the integrating testing some errors were detected, out of which most of them were fixed.

System Testing

System testing is designed to validate a fully developed system to assure that it meets its requirements. The test cases are therefore designed solely based on SRS document. All the basic units mentioned in the design document are to be tested. Every intermediate results of each module are match with expected result.

6.2 Test Cases

A test case is a document that describes an input, action or event and an expected response, to determine if a feature of an application is working currently. A test case contains particulars such as test case identifiers, test case name, objective, test condition, input data requirement, expected results.

The process of developing test case can help find problems in the requirements or design of an application, since it requires completely thinking through the operation of the application. For this reason, it is useful to prepare test cases early in the cycle if possible.

Test for admin operation

Test case ID	Operation	Inputs	Expected Results	Result
1	Login	Admin login ID and Password	If login ID and password are correct, admin can access the admin panel. Else an error message will be displayed.	Verified
2	Logout		Log out the admin from the admin panel.	Verified
3	Add another admin	Admin login ID, password, and confirm password.	If admin login ID is available, create new admin else error message will be displayed.	Verified
4	Add head office/centre	Name of the centre, name of the head of the centre, street, city, state, pin, mobile number, email id.	If name of centre does not exist, add to database, else an error message will be displayed.	Verified
5	Add centre	Name of centre, select head office/centre, street, city, state, pin and mobile number.	If centre already exist, an error message will be displayed else add centre to database.	Verified
6	Add employer/ company.	Select the row to be added. Employer login ID, password and confirm password.	If employer already exist, an error message will be displayed else employer will be added.	Verified
7	Add new course	Course code and course name.	Check and display error message if course already exist, else add course.	Verified

8	View	Select desired option from the menu to be viewed.	Display a list of data of the selected option from the database.	Verified
9	Delete admin	Select desired admin to be deleted.	Delete admin, else display an error message.	Verified
10	Update/delete Head centre/ office	Select desired Head centre to be deleted or updated.	Delete or update Head centre respectively, else display an error message.	Verified
11	Update/delete centre	Select desired centre to be deleted or updated.	Update or delete centre respectively, else display an error message.	Verified
12	Update/delete employer	Select desired employer to be updated or deleted.	Update or delete employer respectively, else display an error message.	Verified

Test for centre user operation

Test Case ID	Operation	Inputs	Expected Results	Result
1	Login	Centre user login ID and password	If login ID and password are correct, centre user can access the centre user panel. Else an error message will be displayed.	Verified
2	Logout		Log out centre user from the centre user panel	Verified
3	Update centre details	Street, city, state, pin, contact	Update address or contact else display an error message.	Verified

4	Add course	Select course name, select duration, select assessor	If course already exist display an error message else add course.	Verified
5	View		Display a list of assessors.	Verified
6	Add assessors	Assessor name and mobile number	If assessor already exist display an error message else add assessor.	Verified
7	Update or delete assessor	Select a desired row to be updated or deleted	Delete or update assessor respectively else display an error message.	Verified
8	View students		Display number of students registered under a particular centre.	Verified
9	Student verification	Select student to be verified	If verified, change status to verified else change status to rejected.	Verified
10	Placement	Student ID, employer ID, date	If student already exist display error message else add student.	Verified

Test for student operation

Test Case ID	Operation	Inputs	Expected Result	Result
1	Login	Student login ID and password	If login ID and password are correct student can register themselves to a centre, else display an error message.	Verified
2	Logout		Logout student.	Verified

3	Self-registration	Student name, address, contact, email, course and centre	Make student form available for centre user to be verified and display status as not verified.	Verified
4	View placement list		Display list of students listed for placement.	Verified
5	Student signup	Student login ID, password, confirm password, email	If student login already in use display error message else signup student.	Verified

Test for employer operation

Test Case ID	Operation	Inputs	Expected Result	Result
1	Login	Employer login ID, password	If login ID and password is correct, login employer else display an error message.	Verified
2	Logout		Logout employer.	Verified
3	Employer signup	Employer login ID, password, confirm password, email	If employer login ID already in use display an error message else signup employer.	Verified
4	Check placement		Display a list of students listed under placement.	Verified

7. Source Code

Centre Details

```
<?php
include('connect.php');
include('centrehome.php');
function input($data)
{
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
<html>
<head>
<style>
.spacing{
    padding:10px 50px 10px 50px;
    font-size: 18px;
    color:teal;
}
.spacing h3{
    color:black;
}
.space{
    position: absolute;
    right: 70px;
    color:Blue;
    cursor: pointer;
    text-decoration:none;
}

</style>
</head>
<body>
<center>
<form  action="<?php  echo  htmlspecialchars($_SERVER["PHP_SELF"]);?>"
method="POST">
```

```
<?php
```

```
$query3=mysqli_query($con,"select centre_id from centre_login where  
c_login='$user'");
```

```
while($show3=mysqli_fetch_assoc($query3))
```

```
{
```

```
    $centre_id=$show3['centre_id'];
```

```
}
```

```
$streetErr= $street= $cityErr= $city= $stateErr= $state= $pinErr= $pin= $contactErr=
```

```
$contact= $selectcourse= $selectcourseErr= $duration= $durationErr=
```

```
$selectassessor= $selectassessorErr= "";
```

```
if ($_SERVER["REQUEST_METHOD"] == "POST")
```

```
{
```

```
    if(empty($_POST['street']))
```

```
    {
```

```
        $streetErr="this field cannot be empty";
```

```
    }
```

```
    else
```

```
    {
```

```
        $street=input($_POST['street']);
```

```
        if (!preg_match("/^[a-zA-Z0-9. ]*$/", $street))
```

```
        {
```

```
            $streetErr = "Only letters and white space allowed";
```

```
        }
```

```
    }
```

```
    if(empty($_POST['city']))
```

```
    {
```

```
        $cityErr="this field cannot be empty";
```

```
    }
```

```
    else
```

```
    {
```

```
        $city=input($_POST['city']);
```

```
        if (!preg_match("/^[a-zA-Z ]*$/", $city))
```

```
        {
```

```
            $cityErr = "Only letters and white space allowed";
```

```
        }
```

```
    }
```

```
    if(empty($_POST['state']))
```

```
    {
```

```

        $stateErr="this field cannot be empty";
    }
    else
    {
        $state=input($_POST['state']);
        if (!preg_match("/^[a-zA-Z ]*$/",$state))
        {
            $stateErr = "Only letters and white space allowed";
        }
    }
    if(empty($_POST['pin']))
    {
        $pinErr="this field cannot be empty";
    }
    else
    {
        $pin=input($_POST['pin']);
        if (!preg_match("/^[0-9]{6}$/",$pin))
        {
            $pinErr = "Only numbers allowed";
        }
    }
    if(empty($_POST['contact']))
    {
        $contactErr="this field cannot be empty";
    }
    else
    {
        $contact=input($_POST['contact']);
        if (!preg_match("/^[0-9]{10}$/",$contact))
        {
            $contactErr = "Only letters and white space allowed";
        }
    }
    if($_POST['selectcourse']!="")
    {
        $selectcourse=($_POST['selectcourse']);
    }
    else
    {

```

```

        $selectcourseErr="this field cannot be empty";
    }
    if($_POST['selectassessor']!="")
    {
        $selectassessor=($_POST['selectassessor']);
    }
    else
    {
        $selectassessorErr="this field cannot be empty";
    }
    if($_POST['duration']!="")
    {
        $duration=($_POST['duration']);
    }
    else
    {
        $durationErr="this field cannot be empty";
    }
    if(isset($_POST['subaddress']))
    {
        if($streetErr != "" || $cityErr != "" || $stateErr != "" || $pinErr !=
"" )
        {
            echo "<script>alert('Update Unsuccessful!!!');
window.location='cdetails.php'</script>";
        }
        else
        {
            $query="update centre_details set street='$street',
city='$city', state='$state', pin='$pin' where centre_id=$centre_id";
            if(MYSQLI_query($con,$query))
            {
                echo "<script>alert('Successfully updated!!!');
window.location='cdetails.php'</script>";
            }
            else
                echo "<script>alert('Update Unsuccessful!!!');
window.location='cdetails.php'</script>";
        }
    }
}

```

```

        if(isset($_POST['subcontact']))
        {
            if($contactErr != "")
            {
                echo "<script>alert('Update      Unsuccessful!!!');
window.location='cdetails.php'</script>";
            }
            else
            {
                $query="update centre_details set contact='$contact'
where centre_id=$centre_id";
                if(MYSQLI_query($con,$query))
                {
                    echo "<script>alert('Successfully updated!!!');
window.location='cdetails.php'</script>";
                }
                else
                    echo "<script>alert('Update      Unsuccessful!!!');
window.location='cdetails.php'</script>";
            }
        }
        if(isset($_POST['subcourse']))
        {
            if($selectcourseErr != "" || $selectassessorErr != "" ||
$durationErr != "")
            {
                echo "<script>alert('Update      Unsuccessful!!!');
window.location='cdetails.php'</script>";
            }
            else
            {
                $query7=mysqli_query($con,"select * from
centre_takes_course where centre_id='$centre_id' AND course_id='$selectcourse'");
                $numrows=mysqli_num_rows($query7);
                if($numrows=='0')
                {
                    $query="insert into centre_takes_course
values('@','$centre_id','$selectcourse)";
                    if(MYSQLI_query($con,$query))
                    {

```

```

                                $selctc=mysqli_query($con,"select
centre_course_id from centre_takes_course where centre_id='$centre_id' and
course_id='$selectcourse'");

                                while($ctc=mysqli_fetch_assoc($selctc))
                                {
                                    $ctc_id=$ctc['centre_course_id'];
                                    mysqli_query($con,"insert into
training_plan values ('@','$selectassessor','$duration','$ctc_id')");
                                    echo "<script>alert('Successfully
updated!!!'); window.location='cdetails.php'</script>";
                                }
                            }
                            else
                                echo "<script>alert('Update
Unsuccessful!!!'); window.location='cdetails.php'</script>";
                            }
                            else
                                echo "<script>alert('Course already exist!!!');
window.location='cdetails.php'</script>";
                        }
                    }
                }
            }
            $query="select * from centre_details where centre_id='$centre_id'";
            $display=mysqli_query($con,$query);
            while($show=mysqli_fetch_assoc($display))
            {?>
            <!--<tr>
                <td class="z"><?php $hid=$show['head_id'];
                                $query2=mysqli_query($con,"select
head_office from centre_head where head_id='$hid'");

                                while($show1=mysqli_fetch_assoc($query2))
                                {

                                    $head_office=$show1['head_office'];

                                    echo $show1['head_office'];

                                }

                                ?></td>

                <td class="z"><?php echo $show['centre_name'] ?></td>
                <td class="z"><?php echo $show['street'] . ", " . $show['city'] . ", ".
$show['state'] . ", ". $show['pin']; ?></td>

```

```

        <td class="z"><?php echo $show['contact'];?></td>
    </tr>

</table>-->
</center>
<div class="spacing">
<h3>Centre name:</h3>
<?php echo $show['centre_name'];?><hr>
<h3>Centre Head name:</h3>
<?php echo $head_office;?><hr>
<h3>Address:</h3>
<?php echo $show['street'] . ", " . $show['city'] . ", " . $show['state'] . ", " .
$show['pin'];?><a class="space" data-toggle="modal" data-target="#address"></a><hr>
    <div class="modal fade" id="address" role="dialog">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header">
                    <button type="button" class="close" data-dismiss="modal">&times;</button>
                    <h4 class="modal-title">Change address</h4>
                </div>
                <div class="modal-body" style="color:black">
                    Street<input type="text" name="street" value="<?php echo $show['street'];
?>">
                        City<input type="text" name="city" value="<?php echo
$show['city']; ?>">
                        State<input type="text" name="state" value="<?php echo
$show['state']; ?>">
                        Pin<input type="text" name="pin" value="<?php echo $show['pin'];
?>">
                </div>
                <div class="modal-footer">
                    <input type="submit" name="subaddress" value="update">
                </div>
            </div>
        </div>
    </div>
</div>
<h3>Contact:</h3>

```



```
<?php echo $show['contact'];?><a class="space" data-toggle="modal" data-
target="#contact"> </a><hr>
```

```
<div class="modal fade" id="contact" role="dialog">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal">&times;</button>
        <h4 class="modal-title">Change contact</h4>
      </div>
      <div class="modal-body" style="color:black">
        Contact<input type="text" name="contact" value="<?php echo $show['contact'];
?>">
      </div>
      <div class="modal-footer">
        <input type="submit" name="subcontact" value="update">
      </div>
    </div>
  </div>
</div>
```

```
<h3>Course Offered:</h3>
<a class="space" data-toggle="modal" data-target="#selectcourse"><u>Add
course</u></a><br>
<div class="modal fade" id="selectcourse" role="dialog">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal">&times;</button>
        <h4 class="modal-title">Add course</h4>
      </div>
      <div class="modal-body" style="color:black">
        Select course<select name="selectcourse"><option value="">---select course---
</option>
        <?php
          $selquery=mysqli_query($con,"select * from course");
          while($selshow=(mysqli_fetch_assoc($selquery)))
          { ?>
```

```

                                <option                value="<?php                echo
$selshow['course_id'];?>"><?php echo $selshow['course_name'];?></option>
                                <?php }?></select>
                                Duration<select        name="duration"><option        value="">---select
duration---</option>
                                <option value="1month">1 month</option>
                                <option value="2month">2 months</option>
                                <option value="3month">3 months</option>
                                <option value="6month">6 months</option>
                                <option value="1year">1 year</option>
                                <option value="2year">2 years</option>
                                <option value="3year">3 years</option>
                                </select>
                                Select assessor<select name="selectassessor"><option value="">---
select assessor---</option>
                                <?php
                                $selassquery=mysqli_query($con,"select * from assessors");
                                while($selassshow=(mysqli_fetch_assoc($selassquery)))
                                { ?>
                                <option                value="<?php                echo
$selassshow['assessor_id'];?>"><?php                echo
$selassshow['assessor_name'];?></option>
                                <?php }?></select>
                                </div>
                                <div class="modal-footer">
                                <input type="submit" name="subcourse" value="update">
                                </div>
                                </div>

                                </div>
                                </div>
                                <table>
                                <tr>
                                <th>Course id</th>
                                <th>Course code</th>
                                <th>Course name</th>
                                <th>Duration</th>
                                <th>Assessor</th>
                                <th> </th></tr>
                                <?php

```

```

$query=mysqli_query($con,"select
*,course_code,course_name,course_duration,assessor_id from centre_takes_course
inner join training_plan on
centre_takes_course.centre_course_id=training_plan.centre_course_id
inner join course on centre_takes_course.course_id=course.course_id where
centre_id='$centre_id'");
while($cshow=mysqli_fetch_assoc($query))
{
    $assessor_id=$cshow['assessor_id'];
}
?>

```

```

<tr>
<td><?php echo $cshow['course_id'];?></td>
<td><?php echo $cshow['course_code'];?></td>
<td><?php echo $cshow['course_name'];?></td>
<td><?php echo $cshow['course_duration'];?></td>
<td><?php

```

```

    $subquery=mysqli_query($con,"select assessor_name from assessors where
assessor_id='$assessor_id'");
    while($subcshow=mysqli_fetch_assoc($subquery))
    {
        echo $subcshow['assessor_name'];
    }?></td>

```

```

<td style="text-align:right"><a href="delcourse.php?id=<?php echo
$cshow['course_id']; ?>"></a></td>

```

```

</tr>
<?php } ?>

```

```

</table>

```

```

</div>

```

```

<?php
}

```

```

?>

```

```

</form>

```

```

</div>

```

```

</body>

```

```

</html>

```

Employer Register

```
<?php include("home.php");?>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
.container{  
    margin-left:auto;  
    margin-right:auto;  
    width:40%;  
    padding: 80px 10px 5px 10px;  
    background-color:none;  
    font-size:20px;  
    color:midnightblue;  
    font-family: 'Comic Sans MS',Comic Sans MS;  
}
```

```
.container1{  
    margin-left:auto;  
    margin-right:auto;  
    width:40%;  
    padding: 1px 10px 10px 10px;  
    background-color:none;  
    font-size:15px;  
    color:midnightblue;  
}
```

```
input[type=submit] {  
    width: 98%;  
    background-color: #4CAF50;  
    color: white;  
    padding: 10px;  
    margin: 2px 0;  
    border: none;  
    border-radius: 4px;  
    cursor: pointer;  
}
```

```
input[type=text] ,select{  
    width: 98%;  
    padding: 2px 10px;
```

```

margin: 8px 0;
display: inline-block;
border: 0;
    outline: 0;
    background: transparent;
    border-bottom: 1px solid black;
border-color: green;
    font-size: 17px;

}
body {font-family: Verdana, sans-serif;}
</style>
</head>
<body>
<?php
$empErr= $emp= $streetErr= $street= $cityErr= $city= $stateErr= $state= $pinErr=
$pin= $mobileErr= $mobile= $sector= $sectorErr= $cpname= $cpnameErr= $date=
$dateErr= $email= $emailErr="";
if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    if(empty($_POST['emp']))
    {
        $empErr="this field cannot be empty";
    }
    else
    {
        $emp=input($_POST['emp']);
        if (!preg_match("/^[a-zA-Z ]*$/",$emp))
        {
            $empErr = "Only letters and white space allowed";
        }
    }
    if(empty($_POST['sector']))
    {
        $sectorErr="this field cannot be empty";
    }
    else
    {
        $sector=input($_POST['sector']);
        if (!preg_match("/^[a-zA-Z ]*$/",$sector))

```

```

        {
            $sectorErr = "Only letters and white space allowed";
        }
    }
    if(empty($_POST['cpname']))
    {
        $cpnameErr="this field cannot be empty";
    }
    else
    {
        {
            $cpname=input($_POST['cpname']);
            if (!preg_match("/^[a-zA-Z ]*$/",$cpname))
            {
                $cpnameErr = "Only letters and white space allowed";
            }
        }
    }
    if(empty($_POST['date']))
    {
        $dateErr="this field cannot be empty";
    }
    else
    {
        {
            $date=input($_POST['date']);
        }
    }
    if(empty($_POST['street']))
    {
        $streetErr="this field cannot be empty";
    }
    else
    {
        {
            $street=input($_POST['street']);
            if (!preg_match("/^[a-zA-Z0-9 ]*$/",$street))
            {
                $streetErr = "Only letters and white space allowed";
            }
        }
    }
    if(empty($_POST['city']))
    {
        $cityErr="this field cannot be empty";
    }
}

```

```

else
{
    $city=input($_POST['city']);
    if (!preg_match("/^[a-zA-Z ]*$/",$city))
    {
        $cityErr = "Only letters and white space allowed";
    }
}
if(empty($_POST['state']))
{
    $stateErr="this field cannot be empty";
}
else
{
    $state=input($_POST['state']);
    if (!preg_match("/^[a-zA-Z ]*$/",$state))
    {
        $stateErr = "Only letters and white space allowed";
    }
}
if(empty($_POST['pin']))
{
    $pinErr="this field cannot be empty";
}
else
{
    $pin=input($_POST['pin']);
    if (!preg_match("/^[0-9]{6}$/",$pin))
    {
        $pinErr = "Only numbers allowed";
    }
}
if(empty($_POST['mobile']))
{
    $mobileErr="this field cannot be empty";
}
else
{
    $mobile=input($_POST['mobile']);
    if (!preg_match("/^[0-9]{10}$/",$mobile))

```

```

        {
            $mobileErr = "Invalid mobile number";
        }
    }
    if(empty($_POST['email']))
    {
        $emailErr="this field cannot be empty";
    }
    else
    {
        $email=input($_POST['email']);
        if (!filter_var($email, FILTER_VALIDATE_EMAIL))
        {
            $emailErr = "invalid email id";
        }
    }
    if(isset($_POST['sub']))
    {
        if($sempErr != "" || $streetErr != "" || $cityErr != "" || $stateErr != "" ||
$pinErr != "" || $mobileErr != "" || $sectorErr != "" || $cpnameErr != "" || $dateErr !=
"" || $emailErr != "")
        {
            if($sempErr)
            {
                echo " <script>alert('$sempErr!!!');
window.location='employerregister.php'</script>";
            }
            else if($streetErr)
            {
                echo " <script>alert('$streetErr!!!');
window.location='employerregister.php'</script>";
            }
            else if($cityErr)
            {
                echo " <script>alert('$cityErr!!!');
window.location='employerregister.php'</script>";
            }
            else if($stateErr)
            {

```



```

        echo                                "<script>alert('$stateErr!!!');
window.location='employerregister.php'</script>";
    }
    else if($pinErr)
    {
        echo                                "<script>alert('$pinErr!!!');
window.location='employerregister.php'</script>";
    }
    else if($mobileErr)
    {
        echo                                "<script>alert('$mobileErr!!!');
window.location='employerregister.php'</script>";
    }
    else if($sectorErr)
    {
        echo                                "<script>alert('$sectorErr!!!');
window.location='employerregister.php'</script>";
    }
    else if($cpnameErr)
    {
        echo                                "<script>alert('$cpnameErr!!!');
window.location='employerregister.php'</script>";
    }
    else if($dateErr)
    {
        echo                                "<script>alert('$dateErr!!!');
window.location='employerregister.php'</script>";
    }
    else if($emailErr)
    {
        echo                                "<script>alert('$emailErr!!!');
window.location='employerregister.php'</script>";
    }
}
else
{
    $query="insert                into                employers
values('@','$semp','$sector','$cpname','$date','$street','$city','$state','$pin','$mobile','$e
mail)";

    $query1=mysqli_query($con,"select * from employers where
employer_name='$semp' and sector='$sector'");

```

```

$numrows=mysqli_num_rows($query1);
if($numrows=='0')
{
    if(mysqli_query($con,$query))
    {
        echo "<script>alert('Successfully added!!!');
window.location='home1.php'</script>";
    }
    else
        echo "<script>alert('Entry Unsuccessful!!!');
window.location='employerregister.php'</script>";
    }
    else
        echo "<script>alert('Employer already exists!!!');
window.location='employerregister.php'</script>";
    }
}
}
function input($data)
{
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
<form action="" method="POST">
<div class="container">
Name of company/employer <input type="text" name="emp"><br>
Sector <input type="text" name="sector"><br>
Contact person's name <input type="text" name="cpname"><br>
Email<input type="text" name="email"><br>
Date <input type="date" name="date"><br>
Street <input type="text" name="street"><br>
City <input type="text" name="city"><br>
State <input type="text" name="state"><br>
Pincode <input type="text" name="pin"><br>
Mobile number <input type="text" name="mobile"><br>
<input type="submit" name="sub" value="Register"><br>

```


<center>Already have an account??
<u>Click here</u> to login</center>

</div>

</form>

<div class="container1">

<hr><center>

We will verify your request and your username and password will be emailed shortly.

Please use the given username and password to login next time.</center>

</div>

</body>

</html>

<?php include("footer.php");?>

8. Conclusion

The project “Development of Training and Placement Management System” was developed to computerize the management and handling of data thus saving time comparatively as compared to the conventional way of managing records.

In the future, the web application can have more functions where the registered students can check their assessment marks and final result and also the centre users can track the progress or status of a student that has been placed in a certain company.

This project was developed using Php as front end and MYSQL as the back end.

9. References

- i. www.w3schools.com
- ii. www.tutorialspoint.com