A Project Report
Submitted as part of requirement for the course
Mini Project (16MCA5DPMP)
V Semester MCA

By

Sneha V K
(1BM16MCA39)
&
Sarthak SenGupta
(1BM16MCA37)

Submitted to
Prof. Lakshminarayana P
Associate Professor

Department of Computer Applications
B.M.S COLLEGE OF ENGINEERING
Bull Temple Road, Bangalore-560019
2018-2019

B M S COLLEGE OF ENGINEERING

Bull Temple Road, Bangalore-560019.

DEPARTMENT OF MCA

CERTIFICATE

This is to certify that **Ms. SNEHA V K (1BM16MCA39)** and **Mr. SARTHAK SENGUPTA (1BM16MCA37)** has satisfactorily completed the project entitled **"ONLINE COURSE REGISTRATION SYSTEM"** for the course **MINI PROJECT- 16MCA5DPMP**, prescribed by BMS College Of Engineering (Autonomous college under VTU), 5th semester MCA course, in this college during the year 2018- 2019.

Signature of Batch In charge

Student Name: SNEHA V K
USN: 1BM16MCA39

Student Name: SARTHAK SENGUPTA
USN: 1BM16MCA37

Signature of HOD

Examiner1:

Examiner2:

AKNOWLEDGEMENT

We consider it a privilege to place our deepest sense of gratitude and sincere thanks to our beloved HOD, **Dr. Sujatha D N** for their sincere co-operation, guidance and supervision during the course of this project work undertaken by us.

We are extremely thankful and wish to express our sincere gratitude to our respected coordinator and our project guide **Prof. Lakshminarayan P** for their kind co-operation and providing valuable suggestions and constant encouragement for the improvement and successful completion of this project.

We express pure and sincere thanks and gratitude for the guidance, timely advices, support, sincere co-operation, suggestions, ideas and also constructive criticisms provided by all the teaching staff of the Master Of Computer Applications, BMS College of Engineering , Bangalore.

Sneha V K (1BM16MCA39)
Sarthak Sengupta (1BM16MCA37)

.

ABSTRACT

The **"ONLINE COURSE REGISTRATION"** system is a software application that is used to register for courses every semester by the students. This application will reduce manual work and easy to organize time easily. Using this software will help to manage document of the student. Login option for both admin and student will be provided. All information regarding the courses prescribed for particular semester should be available on the registration page making it easy to choose for student. The student should be able to view their registered courses and select them, later submit it. Once they are done submitting it will go to admin so that they can view who has not done the process, and insist them to complete soon. The idea for the project was mostly motivated by the online registration system offered by many other colleges. It seems unacceptable that large colleges would have a system that offers such a low level of flexibility and lacks functionality to make the process easier for the students. Combining our knowledge of other course registration systems, we came up with a list of features that are important to an online course registration system. Saves time taken to fill form and waiting for proctor signature. Project reduces errors will happen during registration. Students need not remember course codes and course names. No space for ambiguity. Students need not rush for filling form and submitting. It contains individual logins to access. Admin can view the registered student's details and delete if there is pending of approval. This system will help both students and college in order to manage document of students.

CONTENTS

List of Tables:

List of Figures:

1. INTRODUCTION

1.1 Problem Statement:

Every student has to face problem of manual registrations using forms at college for different records purpose and must be submitted with the stipulated date and time to college office without any delay. This consumes lots of time of student and prone to error is major issue and again rework needs time. Also college faculties won't be available for various reasons to approve, it also delays the work. Lots of confusions and error takes place while filling forms and submitting. To overcome this issue we have come up with the solution "Online Course Registration System". Further problem description, pros and cons are described below.

1.1 Problem Description:

Recently, many colleges have moved from manual course registration process to web-based course registration process that allows students to register for courses prescribed for that particular semester. As more colleges follow this trend and begin to only offer online registration, it will become increasingly important for these systems to be considered usable by all students. In addition to the system being usable, it must also provide the student with convenient access to information that will ease the course registering process.

The idea for the project was mostly motivated by the online registration system offered by many other colleges. It seems unacceptable that large colleges would have a system that offers such a low level of flexibility and lacks functionality to make the process easier for the students. Combining our knowledge of other course registration systems, we came up with a list of features that are important to an online course registration system.

Between the two of us we had experiences with a number of course registration systems, each of them had their strengths, but none of them encompassed all the necessary functionality in a reasonable way. Some of the common drawbacks that we encounter in our college are:

- College must dedicate every semester's first day to get course registered by students.
- Any student who won't be able to make it on dedicated day will prolong the process.
- It costs paper use and time of every student as well as college staff.
- Student need to remember all courses with name and code respectively.
- Any mistakes on paper will cost extra forms to register, use of correctors and whiteners messes the look.

Looking at the drawbacks from existing systems, we decided to use system that would have the following features:

- Login option for both admin and student will be provided
- All information regarding the courses prescribed for particular semester should be available on the registration page making it easy to choose for student.
- The student should be able to view their registered courses and select them, later submit it.
- Once they are done submitting it will go to admin so that they can view who has not done the process, and insist them to complete soon.

2. PROJECT PLAN

Project plan is made before initiation to work accordingly to meet deadlines and schedules as per the planning made by the team. It is helpful to track work and find results in every phase and can analyze load of work and speed of work to be increased, decreased or maintained.

2.1 Objective:

Our objective is to create online course registration application which overcomes the drawbacks of the current manual process that is in practice. Proposed system helps students and also college faculties to save time and also first day of college would not be dedicated to register for courses and need not wait for student to complete the process repeating it many times with warnings. Students also need not rush to college from far places just to do this registering process. The proposed system will solve all these issues. Here we will discuss regarding the project plan and respective process.

2.2 Representation of project plan using PERT Analysis:

A Program Evaluation Review Technique (PERT) chart is a project management tool used to schedule, organize, and coordinate tasks within a project. It a methodology developed by the U.S. Navy in the 1950s to manage the Polaris submarine missile program.

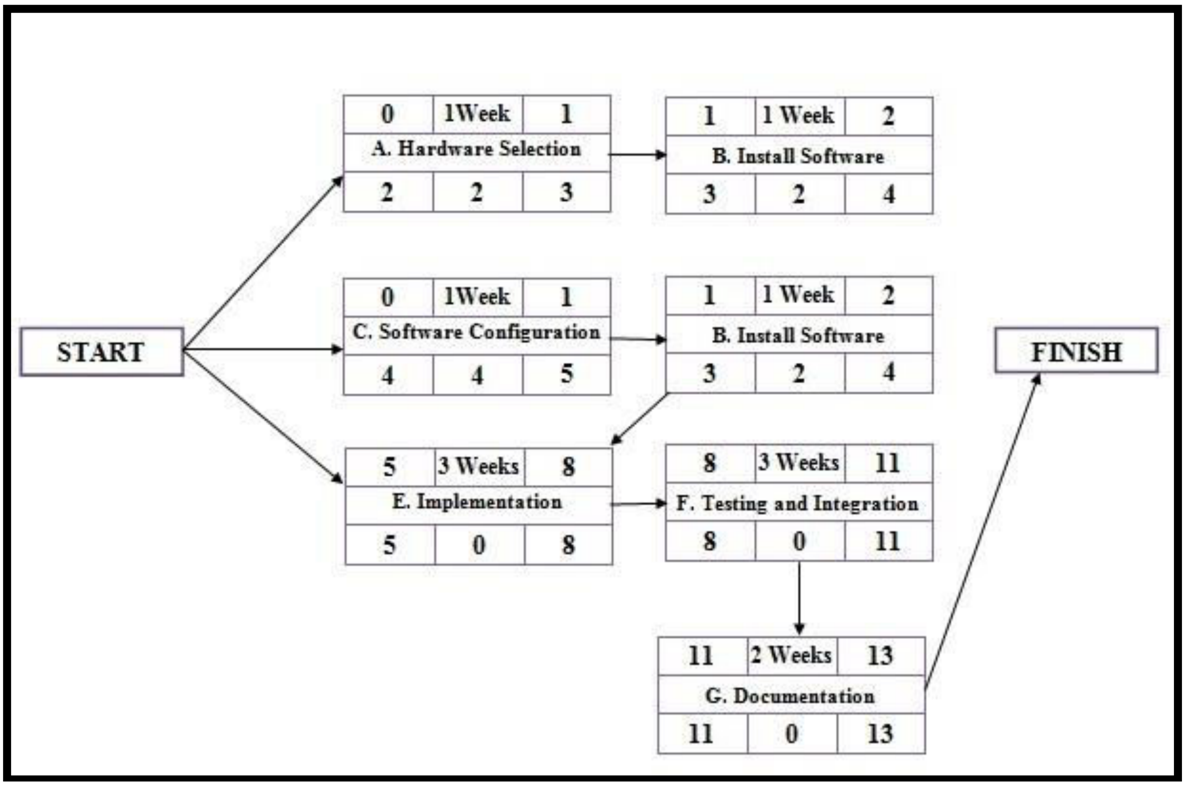


Fig 1 PERT Chart (On Weekly Basis)

2.3 Representation using Gantt chart:

A Gantt chart is a horizontal bar chart developed as a production control tool in 1917 by Henry L. Gantt, an American engineer and social scientist. Frequently used in project management, a Gantt chart provides a graphical illustration of a schedule that helps to plan, coordinate, and track specific tasks in a project.

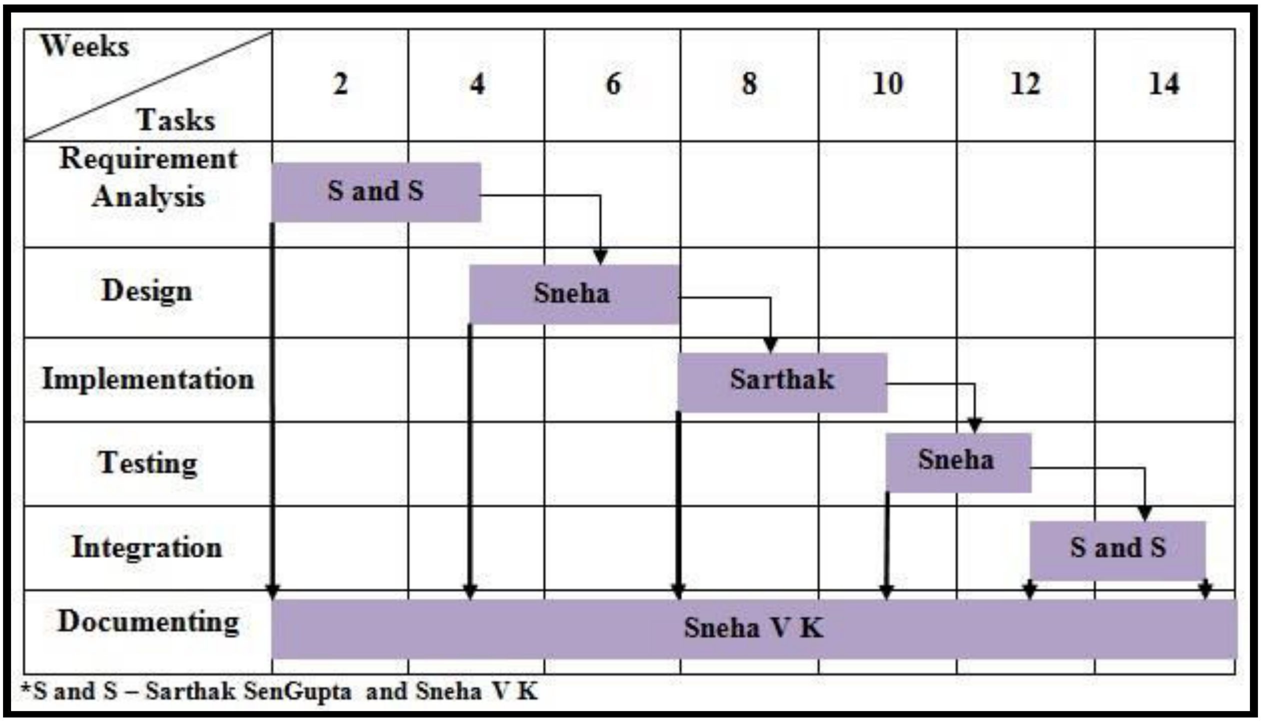


Fig 2 Gantt chart (On Weekly Basis)

2.4 Risk Assessment:

The risk identified is the mobile application implementation there would be version issue as well as platform incompatibility issue. The risk assessed in manual process can be overcome by this application, it tries to reduce and bring a proper co-ordination between all the levels of authentication. Still there are many future enhancements in the proposed system that is to be implemented after those improvements are made it can be converted into mobile application.

2.5 Advantages of proposed system:

- Saves time taken to fill form and waiting for proctor signature.
- No errors will happen during registration.
- Need not remember course codes and course names.
- No space for ambiguity.
- Students need not rush for filling form and submitting.
- It contains individual logins to access.
- Admin can view the registered student’s details and delete if there is pending of approval.

3. SOFTWARE REQUIREMENT SPECIFICATION (SRS)

All the related functional options behavioral understanding will be structured with the help of software requirement specification documentation as we need that before the development starts all the related functionalities that are required to be integrated for the working in real time that should be discussed and outline so that this documentation format can be referred at the time of development and implementation. Functional requirements and non-functional requirements will be discussed and documented making a clear understanding for all the associate of the proposed system.

Software requirement specification consists of many phases they are:

3.1 Feasibility Study

3.2 Users

3.3 Functional Requirements

3.4 Non-Functional Requirements

3.5 System Requirements

3.1 Feasibility Study:

The feasibility study involves analysis of the problem and collection of the relevant information relating to the project. The main aim is to determine whether the undertaking project is economical and technically operable to develop the project.

3.1.1 Technical Feasibility

3.1.2 Economical Feasibility

3.1.3 Operational Feasibility

3.1.1 Technical Feasibility:

The technical feasibility is mainly focused on the current available resource in the college for developing the proposed system. If the current resource doesn’t satisfy the requirement of proposed system, then the proposed system may not be worthy to develop. Therefore, the system developed here is worthy, which satisfy all the minimum software and hardware requirements without expectation of any high end processor and systems to work with.

3.1.2 Economical Feasibility:

Economical feasibility determines whether the software product being developed gives benefits for the college. One should decide whether the proposed system will be completed with cost assigned to it so that it is economically feasible for the college.

This project is found to be economically feasible it does not need additional funding for development. All software and devices used in the development of the project is licensed and user friendly.

3.1.3 Operational Feasibility:

Operational feasibility determines whether the proposed system which is being developed solves the problem statement. It should tell us whether the product is going to completely or partially solve the problem present in the existing system.

This project is found to be operationally feasible since all the requirements are valid and can be fulfilled. It can also replace the existing manual system.

Modular representation of project process:

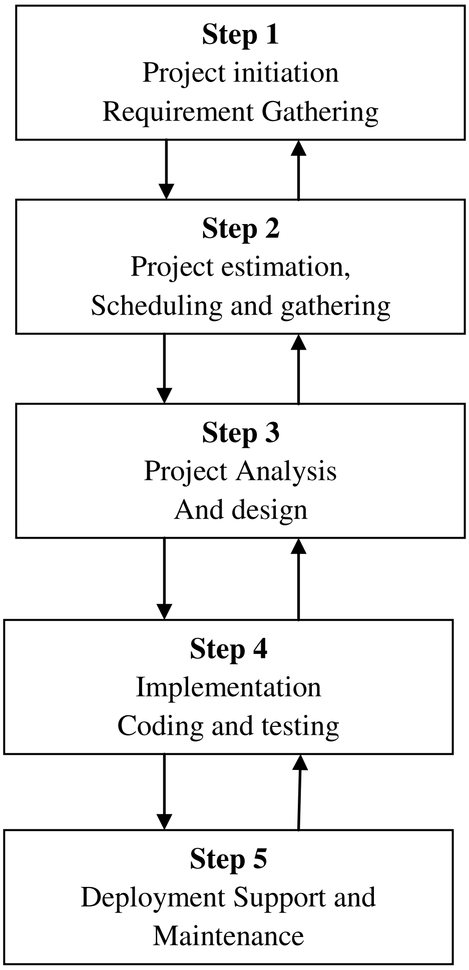


Fig 3 Proposed system process representation

3.2 Users:

This following phase explains about the various stakeholders who are important for the project "Online Course Registration System" and also about their roles in the system.

3.2.1 Admin

3.2.2 Student

3.2.3 Proctor

3.2.4 HOD

3.2.1 Admin:

- Admin has a login.
- Admin can view and delete registration if any issue arises and need to delete duplicate entries.

3.2.2 Student:

- Student has separate logins.
- After login student can register.
- Student can do course registration.

3.2.3 Proctor:

- Proctor has separate login.
- Can view and approve the registered student.

3.2.4 HOD:

- HOD has separate login.
- Student who has registered can be approved.

3.3 Functional Requirements:

Functional requirements describe the functionalities that are available in the proposed system and how they work. The application provides the following functionalities:

- Student login with username and password.
- Student will register with name and other credentials.
- Student fills the course registration from.
- Proctor views and approves the student registration.
- HOD approves after proctor has approved.
- Admin can view and insist other students to register as soon as possible
- If in case of any duplication of registration or need to be disapproved it can be done using delete option provided to admin.

3.3.1 Functional Modules:

3.3.1.1 Admin Module:

Administrator can login and view the approved registration and have a count of how many students has yet to complete registration for course.

3.3.1.2 Student Module:

Student can login and proceed with course registration.

3.3.1.3 Proctor Module:

Proctor can login, view and approve the registered student.

3.3.1.4 HOD Module:

HOD can login, view and approve the registered student.

3.4 Non-Functional Requirements:

3.4.1 Robustness:

The application can easily handle all the related errors that can arise at the time of execution on the online platform with the life revision of working.

3.4.2 Portability:

The application will be portable in nature where it will work as according to the functional requirements in multiple environments and can be access with the authenticated control provided.

3.4.3 Correctness:

All the related information that will be provided to the users with respect to different statistics and metrics will be checked properly for the correctness from multiple sources and model.

3.4.4 Availability:

The application will be available according to the requirements as it is based on service and the related meantime failure is calculated to 10 seconds.

3.4.5 Security:

All the related transfer encryptions and the central data management are incorporated so as to provide detailed security mechanism for the users. All the related central working required security provisions will be provided as online attacks or wrong inputs.

3.4.6 Compatibility:

Application is built using scripting language. This project is compatible to browser and system with minimum hardware requirements.

3.5 System Requirements:

Hardware and Software requirements for this particular project are as follows:

3.5.1 Hardware requirements:

Sl No	Hardware component	Name
1	Processor	Intel Core i3-4030U CPU@ 1.90GHz
2	RAM	4GB
3	Hard Disk	1TB

3.5.2 Software requirements:

Sl No	Software component	Name
1	Operating System	Windows 8 (Minimum)
2	Front end	PHP 7.1
3	Back end	My SQL 5.6.16
4	Server	XAMPP 1.8.3

As of now it works on basic configured desktop and laptop with decent hardware configurations further we need to update it to mobile application in future. Planning to focus on android first.

4. ANALYSIS AND DESIGN

It is the most necessary step of developing software. The purpose of this phase is to deal an easy solution for the complications. This phase is the footstep that moves from complication section to solution section. Design of the structure may be most demanding factor poignant the operating system's quality that has a severe brunt with the latter stages. Design document is the output of this phase.

4.1 Analysis:

4.1.1 General Consideration:

- Our project runs on computer.
- It is web application.
- All request and response has pre-defined format.

4.1.2 Development Method:

The development method of this project is agile methodology. Agile SDLC model is a mixture of iterative and incremental process models. Every step involves cross-functional idea running simultaneously on planning, requirement collection and so on. Finally the working component is given to the customer and stakeholders.

4.1.3 Architectural strategy:

To develop this application we have used PHP for front end with HTML and CSS. MySQL server for the backend support. According to us PHP is well supported scripting language where one can design both server side and client side application request and response in formatted way. Complete project runs on XAMPP server that supports both PHP and MySQL. For effective GUI we have blended with HTML and CSS.

Advantages of using PHP:

- It is a cross platform.
- It has got easy features to connect to backend (database).
- It is easy to use.
- It is open source software.

Advantages of using XAMPP server:

- XAMPP is portable so that it can be carried on a pen drive.
- Security settings are strict by default, no one can access web server.
- PHP error debugging is enabled by default, which helps when debugging scripts.

4.2 Design:

1. Data Flow Diagram (DFD)

2. Entity Relationship Diagram (ERD)

3. Use Case Diagram

4. Class Diagram

5. Sequence Diagram

6. Activity Diagram

4.2.1 Data Flow Diagram:

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

DFD Design (LEVEL 0):

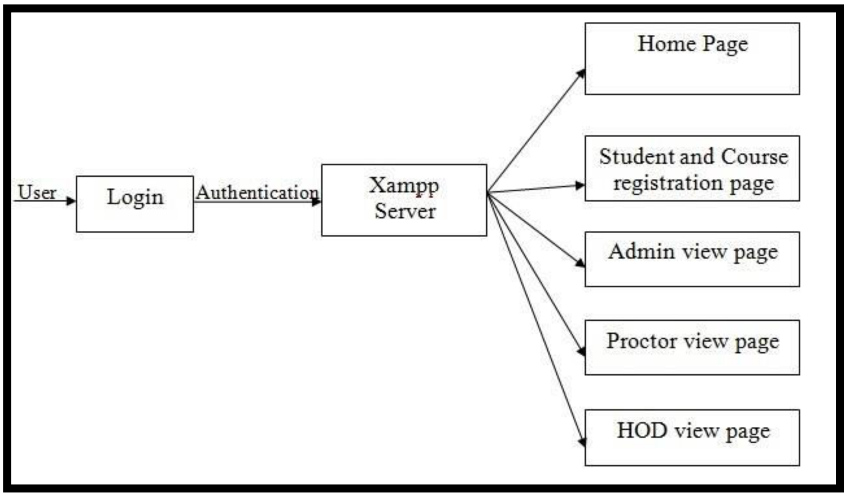


Fig 4 DFD Design (Level 0)

DFD Design (LEVEL 1):

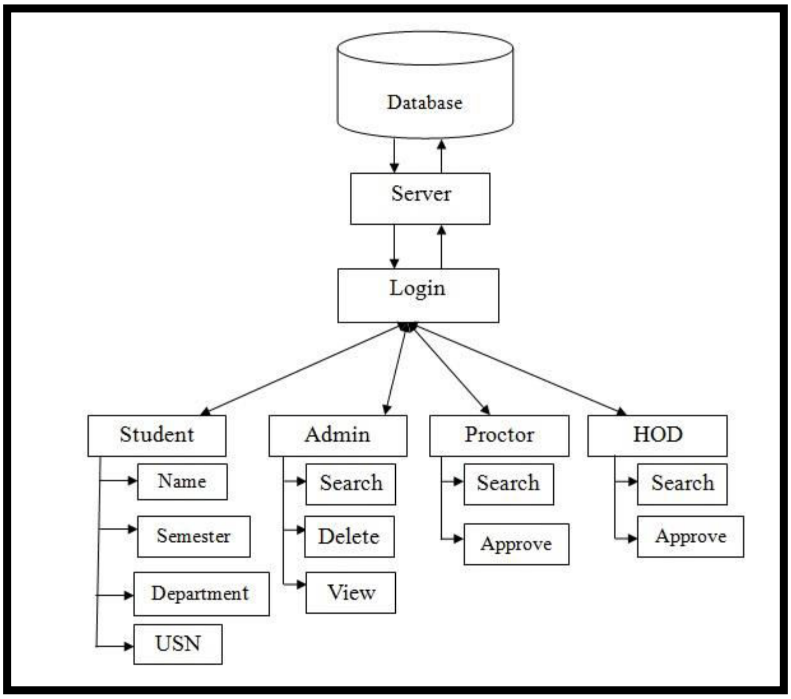


Fig 5 DFD Design (Level 1)

4.2.2 Entity Relationship Diagram:

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases.

ER Diagram:

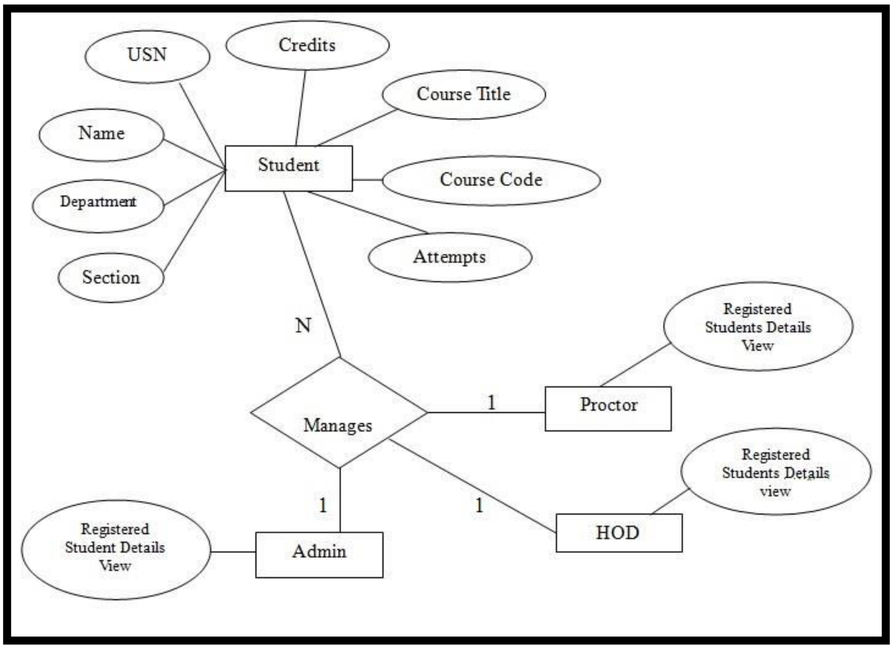

USN
Credits
Course Title
Name
Student
Course Code
Department
Section
Attempts
N
Registered Students Details View
Manages
1
Proctor
Registered Students Details view
1
1
HOD
Registered Student Details View
Admin

Fig 6 ER Diagram

4.2.3 Use Case Diagram:

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors).

Use Case Diagram:

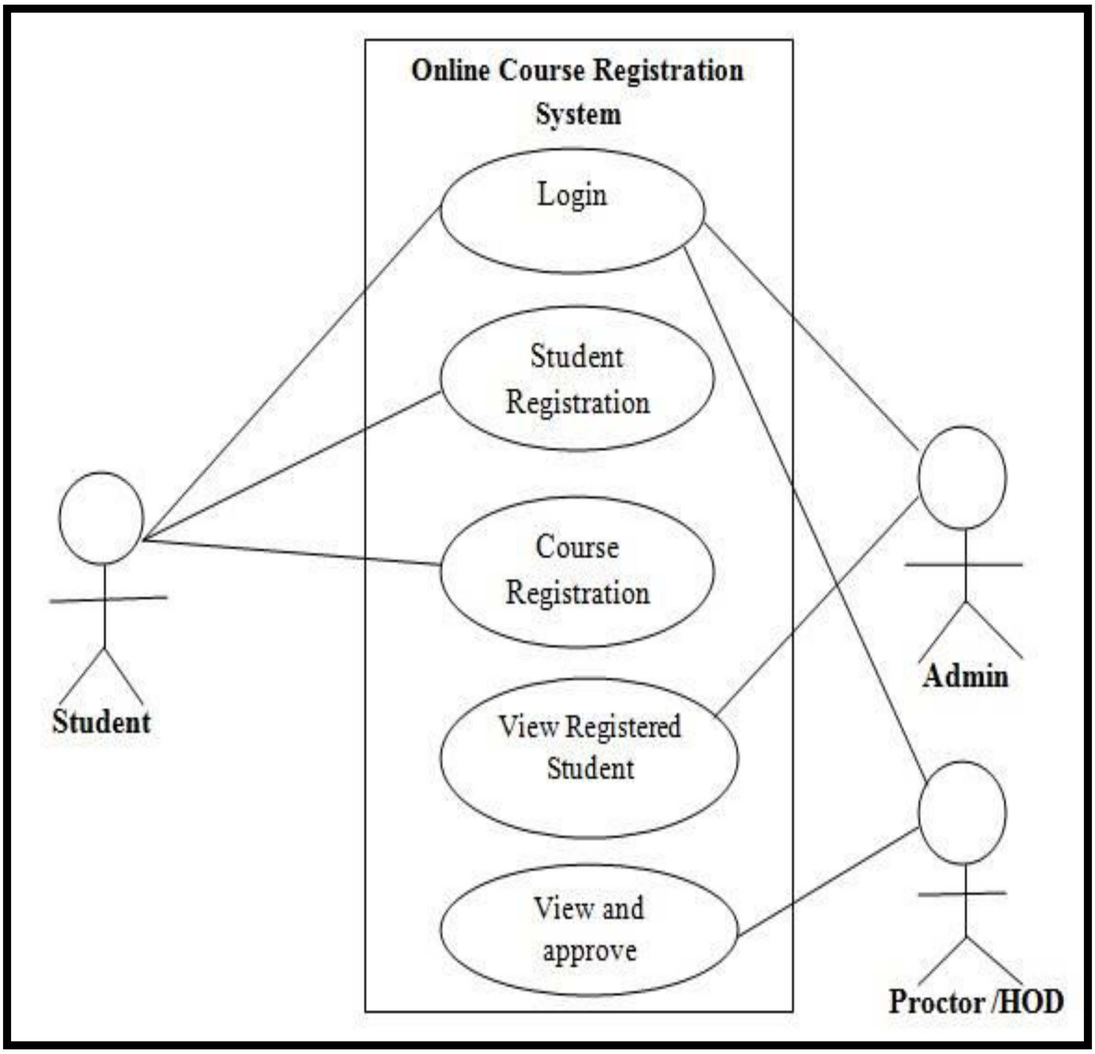


Fig 7 Use case Diagram

4.2.4 Class Diagram:

Class diagram in UML is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

Class Diagram:

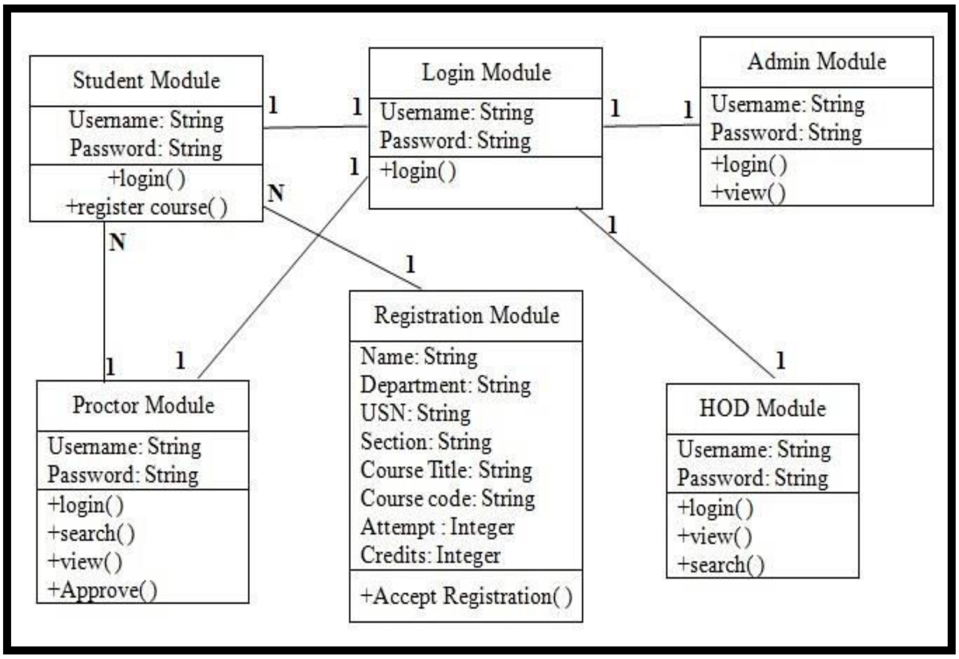


Fig 8 Class Diagram

4.2.5 Sequence Diagram:

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place.

Sequence Diagram:

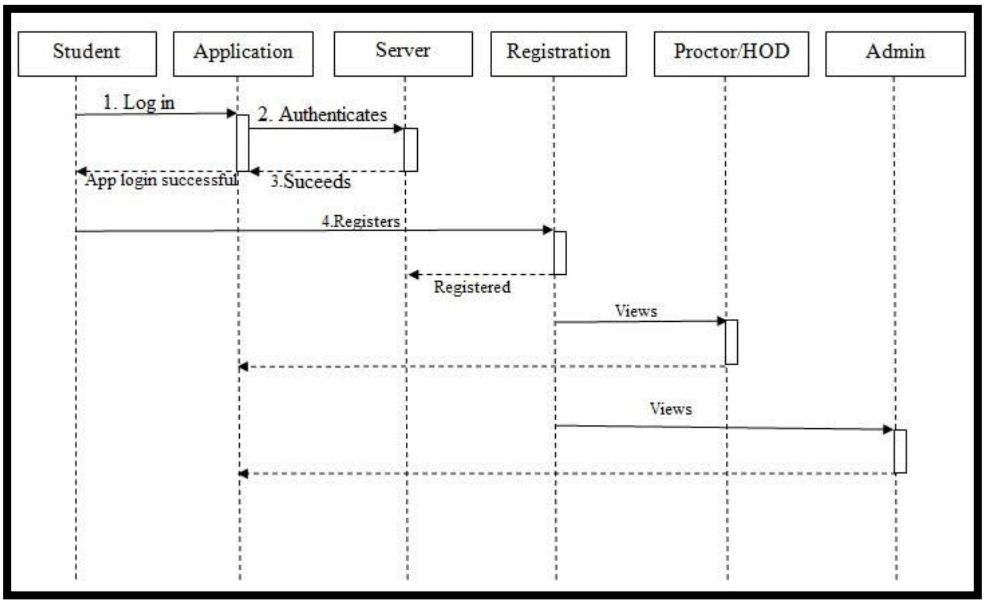


Fig 9 Sequence Diagram

4.2.6 Activity Diagram:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

Activity Diagram:

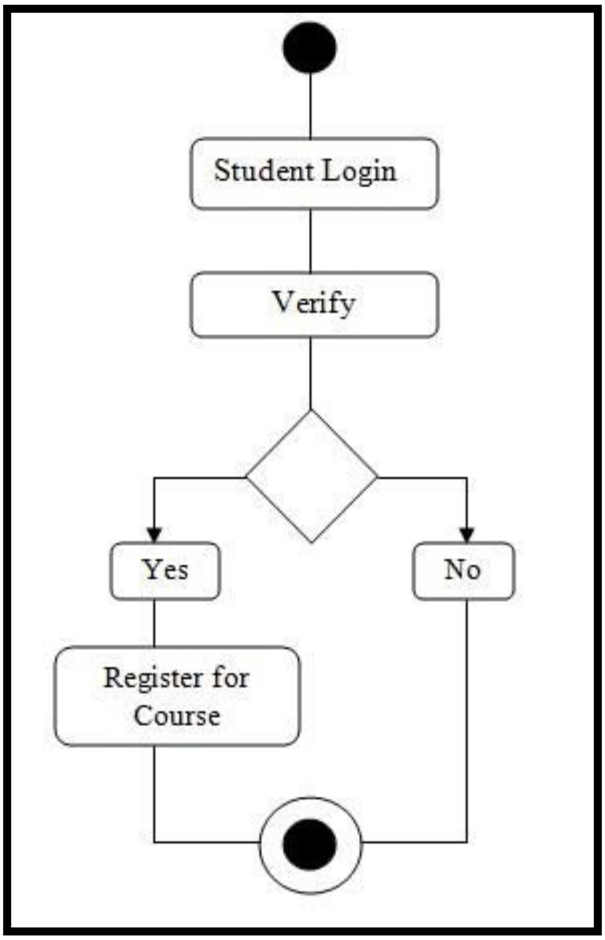


Fig 10 Activity Diagram

5. IMPLEMENTATION

The process of building an entire system is known as system implementation. The building of each component of the system and its integration to form a complete system is known as system implementation and integration. The team specified for the implementation is held responsible for the construction of the GUI, database, middleware, server side programming and integration. Some code may be already available and some code must be freshly written. Both should be welcomed by the implementation team.

An activity to build a web application using some existing code is handled by the project teams who have designed the activity.

- People who are not involved in application designing are Application users and owner.
- The person who designs the application that is to be used by all kind of users is application designer.
- People who build the network and continuously monitor whether it works are known as application builders.

5.1 Modules Implemented:

Project has been divided into four modules they are:

- Admin
- Student
- Proctor
- HOD

5.1.1 Admin:

Admin Module is mainly used for determining students who have completed course registration and student who haven’t. It also has a delete option so that the admin can delete student who has left the course in middle or any duplication.

Pseudo Code:

```
Start
Step 1: Admin has to login.
Step 2: View students who have completed course registration
Step 3: Delete if there is duplication or any discontinuity in the course.
End
```

5.1.2 Student:

Student module is used to login and register with details and complete the respective course registration process.

Pseudo Code:

```
Start
Step 1: Student logs in
Step 2: Register his/her details
Step 3: It redirects to course registration page.
Step 4: Selects the courses respective to semester
End
```

5.1.3 Proctor:

Proctor module is used to login and view the students who are registered for course and approve it for further procedure redirecting to HOD module.

Pseudo Code:

```
Start
Step1: Proctor logs in
Step2: Can view student details who have registered
Step3: Can approve it to send it further
End
```

5.1.4 HOD:

HOD module is used to login and view the student details that have been approved by the proctor it also has option to view and approve.

Pseudo Code:

```
Start
Step1: HOD logs in
Step2: Can view student details approved by proctor
Step3: Can approve
End
```

5.2 Implementation Description:

Each web page in this project contains interactive interface compromising HTML, CSS design and PHP scripting in order to work dynamically. Working on this project we solved many challenges and have implemented, project is yet not completely implemented it needs lots of additional feature which we will be completing in future also conversion of web based application to mobile application is yet another improvement needed. It also holds links and database connectivity in order to make it much more interactive and user friendly.

5.3 Coding Snippets:

5.3.1 Home page:

```
<html>
<head>
<title>Welcome Page</title>
<link href="style.css" rel="stylesheet" type="text/css">
<meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
<header>
<div class="row">
<div class="logo">
<img src="bms.png">
</div>
<ul class="main-nav">
<li ><a href="">HOME</a></li>
<li><a href="">ABOUT US</a></li>
<li><a href="">CONTACT US</a></li>
</ul>
</div>
<div class="hero">
<h1>Welcome Students </h1>
<div class="button">
<a href="http://localhost/Student/Login/studentlog.php" id="stud" class=" btn btn-
one"><b>Student</b></a>
<a href="http://localhost/Admin/logs/adminLog.php" class="btn btn-two"><b>Admin</b></a>
<a href="http://localhost/Proctor/proc_log.php" class="btn btn-three"><b>Proctor</b></a>
<a href="http://localhost/HOD/HodLog.php" class="btn btn-four"><b>HOD</b></a>
</div>
</div>
</header>
</body>
</html>
```

5.3.2 Login page:

```
<?php
$host="localhost";
$user="root";
$password="";
$db="bmsce";
$con = mysqli_connect($host,$user,$password, $db);
mysqli_select_db($con, $db);
if(isset($_POST['username'])){
$uname=$_POST['username'];
$password=$_POST['password'];
$sql= "select * from loginform where Username='".$uname."'AND Password='".$password."' limit 1";
$result=mysqli_query($con, $sql);
if(mysqli_num_rows($result)==1){
//echo "You have successfully logged in.";
//exit();
function redirect($url)
{
header('Location: '.$url);
exit();
}
redirect('http://localhost/Student/registration/index.php');
}
else
{
echo "Incorrect Password!!!";
}
}
?>
<html>
<head>
<title>Login Form</title>
<link rel="stylesheet" type="text/css" href="style1.css">
</head>
<body>
<div class="loginbox">
<img src="red3.png" class="avatar">
<h1>Login Here</h1>
<form method="POST" action="#">
<p>Username</p>
<input type="text" name = "username" placeholder="Enter Username">
<p>Password</p>
<input type="password" name="password" placeholder="Enter Password">
<input type="submit" name="" value="Login">
<label style="visibility:hidden" id="err">Incorrect Password</label> <br>
<a href="#">Forgot Password?</a><br>
<a href="#">Don't have an account?</a>
</form>
</div>
</body>
</html>
```

5.3.3 Student and Course Registration page:

```
<?php
include "config.php";
if(isset($_POST['name']))
{
if(!mysqli_select_db($con, $dbname))
{
echo 'Database not selected';
}
$name = $_POST['name'];
$sem = $_POST['sem'];
$dept = $_POST['dept'];
$usn = $_POST['usn'];
$sec = $_POST['sec'];
$sql = "INSERT INTO register1 (name,sem,dept,usn,sec) VALUES ('$name','$sem','$dept','$usn','$sec')";
if(!mysqli_query($con, $sql))
{
echo 'Not Registered';
}
else
{
echo 'Registered';
}
header("refresh:2; url= index.php");
}
?>
<!DOCTYPE html>
<html>
<head>
<title>Registration Form</title>
<link href="style.css" rel="stylesheet" type="text/css">
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="jquery-1.12.0.min.js" type="text/javascript"></script>
</head>
<body>
<a style="float:right";  href="http://localhost/Student/Login/studentlog.php" id="stud" class=" btn btn-
one"><b>Logout</b></a>
<h3>BMS COLLEGE OF ENGINEERING, BANGALORE - 19</h3>
<h1>Autonomous Institute, affilated to VTU</h1>
<h2>COURSE REGISTRATION FOR THE ACADEMIC YEAR <select name="year1" id="y1" >
<option value="-1">Year:</option>
<option value="2018">2018</option>
<option value="2019">2019</option>
<option value="2020">2020</option>
</select>
TO
<select name="year2" id="y2" >
<option value="-1">Year:</option>
<option value="2019">2019</option>
<option value="2020">2020</option>
<option value="2021">2021</option>
</select></h2>
<div class="simple-form">
<form id ="registration" action="index.php" method="POST">
```

```
<input type="text" name="name" id="btn" placeholder="Enter your Full Name"
maxlength="30"><br><br>
<select name="sem" id="sem_sel" >
<option value="-1">Semester</option>
<option value="1">I</option>
<option value="2">II</option>
<option value="3">III</option>
<option value="4">IV</option>
<option value="5">V</option>
<option value="6">VI</option>
</select><br><br>
<input type="text" name="dept" id="btn" placeholder="Enter your Department"
maxlength="30"><br><br>
<input type="text" name="usn" id="btn" placeholder="Enter your USN" maxlength="30"><br><br>
<input type="text" name="sec" id="btn" placeholder="Enter your Section" maxlength="30"><br><br>
<input type="Submit" value="Submit">
</form>
</div>
<div class="form1">
<form id = "particulars" action="index.php" method="POST">
<table id = "tt1" cellpadding = "10">
<td>
<table>
<tr>
<td align="center"><b>Sl.No.</b></td>
<td align="center"><b>Course Title</b></td>
<td align="center"><b>Course Code</b></td>
<td align="center"><b>Registration Type</b></td>
<td align="center"><b>Attempt number</b></td>
<td align="center"><b>Credits</b></td>
</tr>
<tr>
<td><select id="sel_code3">
<option value="0">- Select Code -</option>
</select></td>
<td><select name="rt1" id = "rrty1" />
<option value="-1">Select Registration Type</option>
<option value="1">Regular</option>
<option value="2">Re-register</option></td>
<td><select name="att_nm1" id = "atmt1" />
<option value="-1">Select Attempt</option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
</td>
<td><select id="sel_cred3">
<option value="0">- Select Title -</option>
</select></td>
</tr>
<tr>
<td align="center">4</td>
<td><select id="sel_title4" onchange="remove_dup(this)">
<option value="0">- Select Title -</option>
<?php
// Fetch Course
```

```
</tr>
<tr>
<td align="center">5</td>

<td><select id="sel_title5" onchange="remove_dup(this)">
<option value="0">- Select Title -</option>
<?php
// Fetch Course
$sql_course = "SELECT * FROM course";
$course_data = mysqli_query($con,$sql_course);
while($row = mysqli_fetch_assoc($course_data) ){
$courseid = $row['id'];
$c_title = $row['c_title'];
// Option
echo "<option value='".$courseid."' class='{$courseid}' >".$c_title."</option>";
}
?>
</select>
</td>
<td><select id="sel_code5">
<option value="0">- Select Code -</option>
</select></td>
<td><select name="rt1" id = "rrty1" />
<option value="-1">Select Registration Type</option>
<option value="1">Regular</option>
<option value="2">Re-register</option></td>
<td><select name="att_nm1" id = "atmt1" />
<option value="-1">Select Attempt</option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
</td>
<td><select id="sel_cred5">
<option value="0">- Select Credits -</option>
</select></td>
</script>
</body>
</html>
```

5.3.4 Admin page:

```
<html>
<head>
<title> Display Data from DB</title>
<style type="text/css">
table{
border: 2px solid red;
background-color: #FFC;
}
th{
border-bottom: 5px solid #000;
}
td{
border-bottom: 2px solid #666;
}
</style>
</head>
<body>
<h1>Display Data from Database</h1>
<a style="float:right"; href="http://localhost/Student/Login/studentlog.php" id="stud" class=" btn btn-one"><b>Logout</b></a>
<?php
include "config.php";
$sqlget = "SELECT * FROM register1";
$sqldata = mysqli_query($con, $sqlget) or die('Error Getting Data!!!');
echo "<table>";
echo "<tr><th>Name</th><th>Semester</th><th>Deaprtment</th><th>USN</th><th>Section</th></tr>";
while ($row = mysqli_fetch_array($sqldata, MYSQLI_ASSOC)){
echo "<tr><td>";
echo $row['name'];
echo "</td><td>";
echo $row['sem'];
echo "</td><td>";
echo $row['dept'];
echo "</td><td>";
echo $row['usn'];
echo "</td><td>";
echo $row['sec'];
echo "</td></tr>";
}
echo "</table>";
?>
<form method = "post" action="Display.php">
Enter: <input type="text" name="usn" id="usn"><br>
<input type="submit" value="Delete">

<?php
include "config.php";

if (isset($_POST['usn']))
{
$usn = $_POST['usn'];

if($con === false){
```

```
die("ERROR: Could not connect. " . mysqli_connect_error());
}

$sql = "DELETE FROM register1 WHERE usn = '".$usn."'";
$result = mysqli_query($con, $sql);
if(!$result)
{
echo "ERROR: Could not able to execute $sql. "
. mysqli_error($con);
}
else{
echo "Record was deleted successfully.";
header("Refresh:0");
}
mysqli_close($con);
}
?>
</form>
</body>
</html>
```

5.3.5 Proctor page:

```
<html>
<head><title>Showing Proctor Students Data</title>
<style type="text/css">
table{
border: 2px solid red;
background-color: #FFC;
}
th{
border-bottom: 5px solid #000;
}
</style>
</head>
<body>
<h1>Proctor Students</h1>
<a style="float:right";  href="http://localhost/Proctor/proc_log.php" id="stud" class=" btn btn-
one"><b>Logout</b></a>
<form method="post" action="proc.php">
</form>
<!-- Displaing all students under a specific proctor -->
<?php
session_start();
$user_name = '';
if (!empty($_SESSION['usrnm'])) {
$user_name = $_SESSION['usrnm'];
}echo "Welcome User: ".$user_name.".";
?>
<?php
include "config.php";
$sqlget = "SELECT * FROM register1 WHERE proc_usrnm = '".$user_name."'";
$sqldata = mysqli_query($con, $sqlget) or die('Error Getting Data!!!');
echo "<table>";
echo "<tr><th>Name</th><th>Semester</th><th>Deaprtment</th><th>USN</th><th>Section</th></tr>";
```

```
while ($row = mysqli_fetch_array($sqldata, MYSQLI_ASSOC)){
echo "<tr><td>";
echo $row['name'];
echo "</td><td>";
echo $row['sem'];
echo "</td><td>";
echo $row['dept'];
echo "</td><td>";
echo $row['usn'];
echo "</td><td>";
echo $row['sec'];
echo "</td></tr>";
}
echo "</table>";
?>
<!-- Proctor Approval -->
<br />
<form method="post" action="proc.php">
<table>
<tr>
<td colspan="2">
<?php
mysqli_select_db($con,'bmsce');
$query = "SELECT name FROM register1 WHERE proc_usrnm = '".$user_name."'";
$result = mysqli_query($con, $query);
?>
<select name='sel1'>
<option value="0">- Select Student -</option>
<?php
while($row = mysqli_fetch_array($result))   {
echo "<option value='".row['name']."'>" .$row['name']."</option>";
} ?>
</select></td>
<td colspan="2">
<input type="checkbox" name="checkval" id = "checkval" value="Approved">Approve<br>
<?php $qry = "INSERT INTO register1(Proc_App) values('checkval') WHERE name = '".$row['name']."'";
$res = mysqli_query($con, $qry);
?>
</td>
</tr>

<tr>
<td colspan="2">
<?php
mysqli_select_db($con,'bmsce');
$query1 = "SELECT name FROM register1 WHERE proc_usrnm = '".$user_name."'";
$result1 = mysqli_query($con, $query1);
?>
<select name='sel2'>
<option value="0">- Select Student -</option>
<?php
while($row = mysqli_fetch_array($result1))   {
echo "<option value='".row['name']."'>" .$row['name']."</option>";
} ?>
</select>
```

```
</td>
<td colspan="2">
<input type="checkbox" name="checkval1" id = "checkval1" value="Approved">Approve<br>
</td>
</tr>
<tr>
<td colspan="2">
<?php
mysqli_select_db($con,'bmsce');
$query2 = "SELECT name FROM register1 WHERE proc_usrnm = '".$user_name."'";
$result2 = mysqli_query($con, $query2);
?>
<select name='sel3'>
<option value="0">- Select Student -</option>
<?php
while($row = mysqli_fetch_array($result2))   {
echo "<option value='".row['name']."'>" .$row['name']."</option>";
} ?>
</select>
</td>
<td colspan="2">
<input type="checkbox" name="checkval2" id = "checkval2" value="Approved">Approve<br>
</td>
</tr>
<tr>
<td colspan="2">
<?php
mysqli_select_db($con,'bmsce');
$query3 = "SELECT name FROM register1 WHERE proc_usrnm = '".$user_name."'";
$result3 = mysqli_query($con, $query3);
?>
<select name='sel4'>
<option value="0">- Select Student -</option>
<?php
while($row = mysqli_fetch_array($result3))   {
echo "<option value='".row['name']."'>" .$row['name']."</option>";
} ?>
">
</td>
</tr>
</table>
</form>
</body>
</html>
```

5.3.6 HOD page:

```
<html>
<head><title>Showing Proctor Students Data</title>
<style type="text/css">
table{
border: 2px solid red;
background-color: #FFC;
}
th{
border-bottom: 5px solid #000;
}
</style>
</head>
<body>
<h1>Students Records</h1>
<form method="post" action="hod.php">
<h2>Search Records by Semester</h2>
<a style="float:right";  href="http://localhost/HOD/HodLog.php" id="stud" class=" btn btn-one"><b>Logout</b></a>
Enter Semester: <input type="text" name="sem" id="sem"><br>
<input type="submit" value="Search">
<br />
<br />
<?php
include "config.php";
if (isset($_POST['sem']))
{
$sem = $_POST['sem'];
if($con === false)
{
}
echo "<table>";
echo "<tr><th>Name</th><th>Semester</th><th>Deaprtment</th><th>USN</th><th>Section</th></tr>";
while ($row = mysqli_fetch_array($result, MYSQLI_ASSOC)){
echo "<tr><td>";
echo $row['name'];
echo "</td><td>";
echo $row['sem'];
echo "</td><td>";
echo $row['dept'];
echo "</td><td>";
echo $row['usn'];
echo "</td><td>";
echo $row['sec'];
echo "</td></tr>";
}
echo "</table>";
mysqli_close($con);
}
?>
</form>
</body>
</html>
```

5.4 Screenshots:

5.4.1 Home Page:

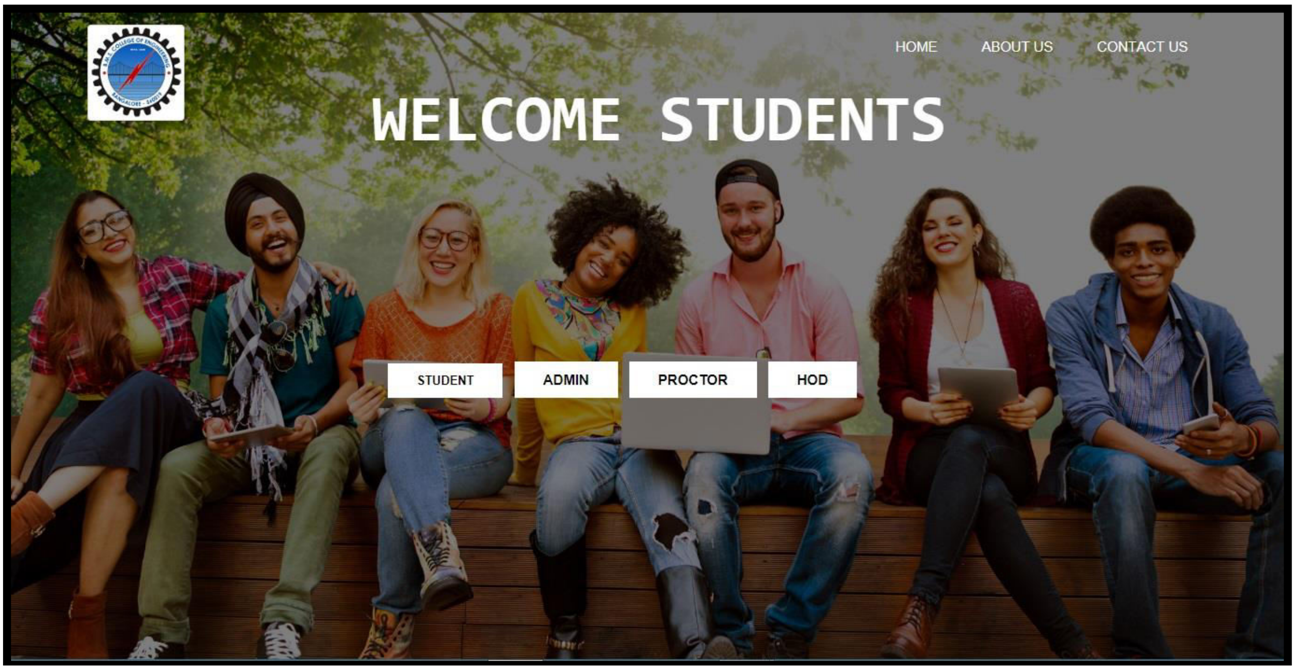
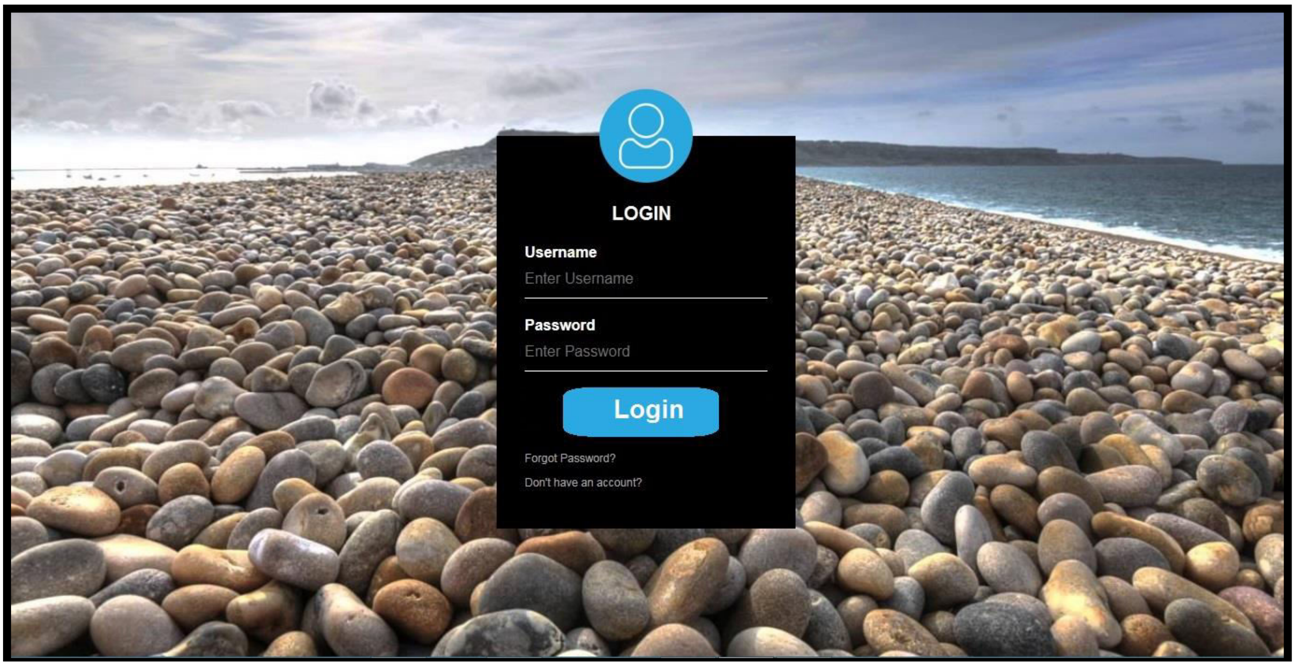
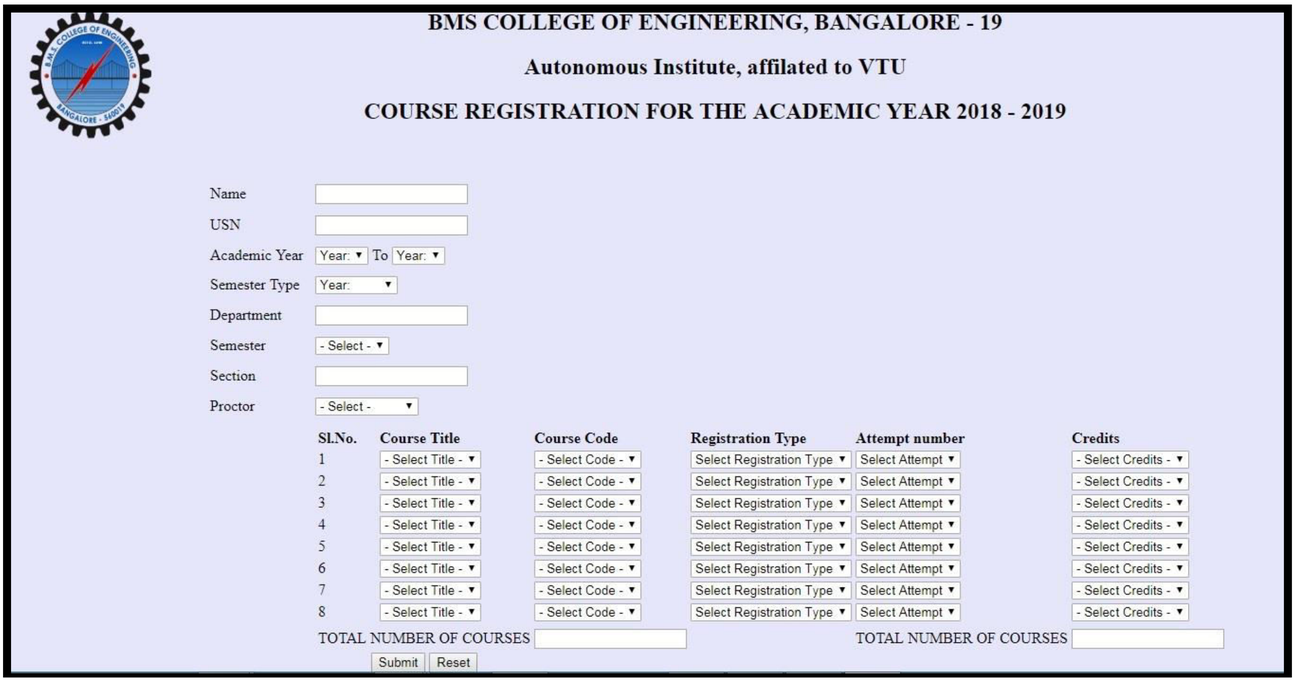


Fig 11 Home Page

5.4.2 Login Page:



5.4.3 Student and Course Registration page:



BMS COLLEGE OF ENGINEERING, BANGALORE - 19

Autonomous Institute, affilated to VTU

COURSE REGISTRATION FOR THE ACADEMIC YEAR 2018 - 2019

Name

USN

Academic Year Year: ▾ To Year: ▾

Semester Type Year: ▾

Department

Semester - Select - ▾

Section

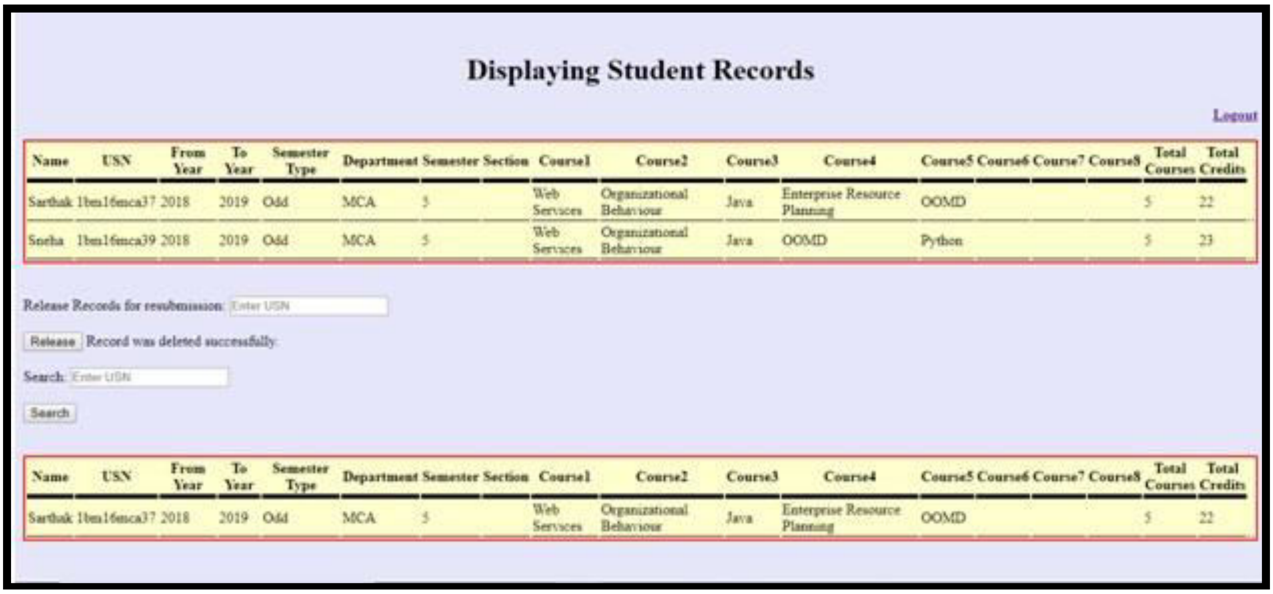
Proctor - Select - ▾

Sl.No.	Course Title	Course Code	Registration Type	Attempt number	Credits
1	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾
2	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾
3	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾
4	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾
5	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾
6	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾
7	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾
8	- Select Title - ▾	- Select Code - ▾	Select Registration Type ▾	Select Attempt ▾	- Select Credits - ▾

TOTAL NUMBER OF COURSES TOTAL NUMBER OF COURSES

Fig 13 Student registration page

5.4.4 Admin page:



Displaying Student Records

Name	USN	From Year	To Year	Semester Type	Department	Semester	Section	Course1	Course2	Course3	Course4	Course5	Course6	Course7	Course8	Total Courses	Total Credits
Sarthak	1bm16mca37	2018	2019	Odd	MCA	5		Web Services	Organizational Behaviour	Java	Enterprise Resource Planning	OOMD				5	22
Sneha	1bm16mca39	2018	2019	Odd	MCA	5		Web Services	Organizational Behaviour	Java	OOMD	Python				5	23

Release Records for resubmission: Enter USN

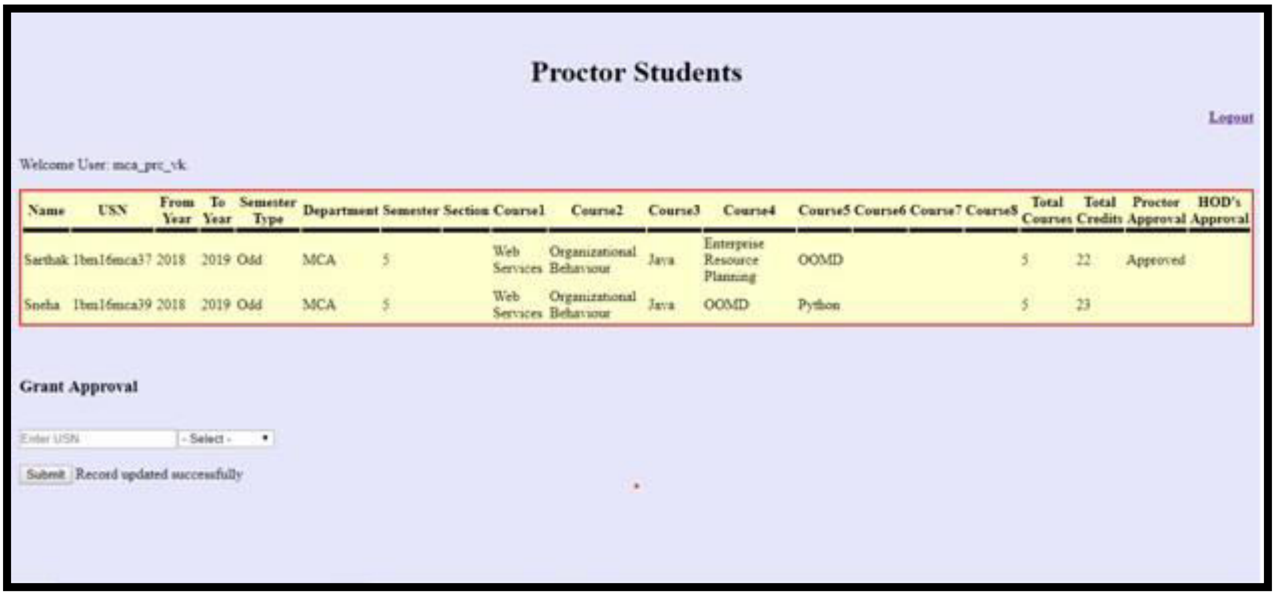
Record was deleted successfully.

Search: Enter USN

Name	USN	From Year	To Year	Semester Type	Department	Semester	Section	Course1	Course2	Course3	Course4	Course5	Course6	Course7	Course8	Total Courses	Total Credits
Sarthak	1bm16mca37	2018	2019	Odd	MCA	5		Web Services	Organizational Behaviour	Java	Enterprise Resource Planning	OOMD				5	22

Fig 14 Admin view page

5.4.5 Proctor page:



Proctor Students

Welcome User: mca_prc_vk

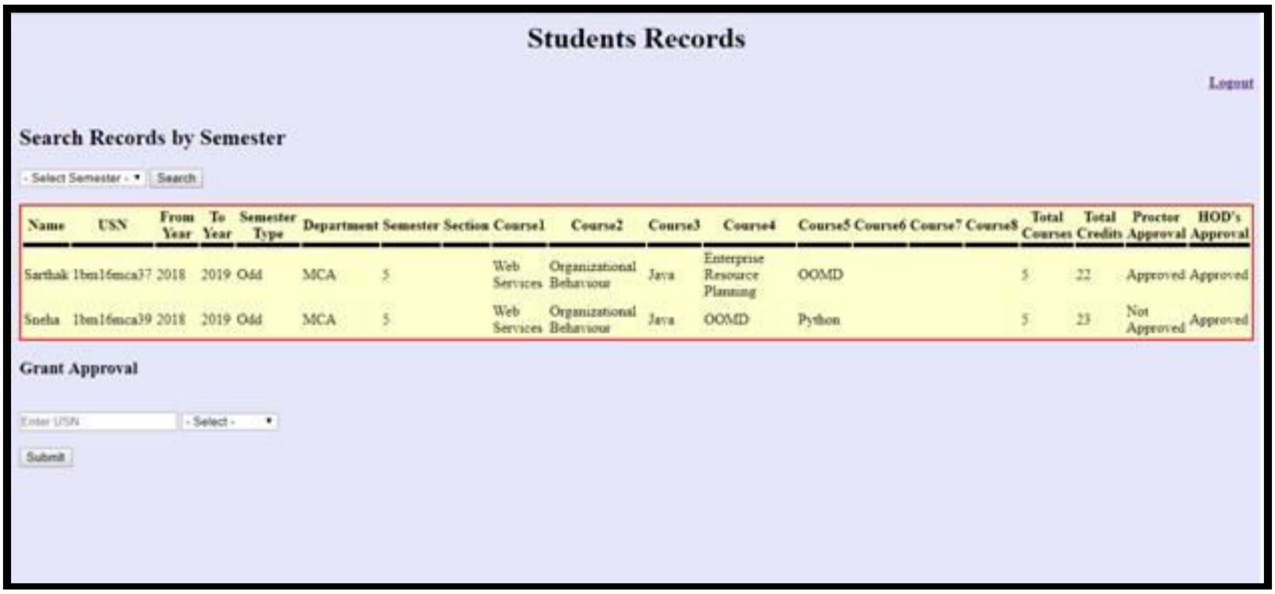
Name	USN	From Year	To Year	Semester Type	Department	Semester	Section	Course1	Course2	Course3	Course4	Course5	Course6	Course7	Course8	Total Courses	Total Credits	Proctor Approval	HOD's Approval
Sarthak	1bm16mca37	2018	2019	Odd	MCA	5		Web Services	Organizational Behaviour	Java	Enterprise Resource Planning	OOMD				5	22	Approved	
Sneha	1bm16mca39	2018	2019	Odd	MCA	5		Web Services	Organizational Behaviour	Java	OOMD	Python				5	23		

Grant Approval

Record updated successfully

Fig 15 Proctor View Page

5.4.6 HOD page:



Students Records

Search Records by Semester

Name	USN	From Year	To Year	Semester Type	Department	Semester	Section	Course1	Course2	Course3	Course4	Course5	Course6	Course7	Course8	Total Courses	Total Credits	Proctor Approval	HOD's Approval
Sarthak	1bm16mca37	2018	2019	Odd	MCA	5		Web Services	Organizational Behaviour	Java	Enterprise Resource Planning	OOMD				5	22	Approved	Approved
Sneha	1bm16mca39	2018	2019	Odd	MCA	5		Web Services	Organizational Behaviour	Java	OOMD	Python				5	23	Not Approved	Approved

Grant Approval

Fig 16 HOD View Page

6 .SOFTWARE TESTING

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs (errors or other defects).

Software testing can be stated as the process of validating and verifying that a software program/application/product:

- Meets the requirements that guided its design and development.
- Works as expected.
- Can be implemented with the same characteristics.
- Satisfies the needs of stakeholders
- Software testing, depending on the testing method employed, can be implemented at any time in the development process.

6.1 Black Box Testing:

Black-box testing tests functional and non-functional characteristics of the software without referring to the internal code of the software. It uses external descriptions of the software like SRS (Software Requirements Specification), Software Design Documents to derive the test cases.

The validation (Project design and play), verification (Accessing application in multiple system throughout the organization), and general usability testing (User interface, Bug free and faster access).

6.2 White Box Testing:

The proposed application contains various different modules and integrated successfully. All independent paths within a module, logical decisions, and loops at their boundaries and within their operational bounds and Database internal data structures and validations are working as per the client requirements.

6.2.1 Unit testing:

The proposed application is tested completely that verify the functionality of a specific section of code and sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine if they are fit for use

6.2.2 Integration Testing:

The application includes many and different constraints of functionalities and these modules are integrated and tested as per the client requirements. The administrators can interact with the Database and dynamically working for add, update, delete, modify, and manipulation purpose.

6.2.3 Validation Testing:

The application is an Intranet based application that can be accessed throughout the organization or a specified group/Dept and contains with many forms. All forms have many fields with different constraints to enter information about users or clients. All fields‟ validations are done using JavaScript and JQuery and php. Example user name and password, email, phone number, date of birth, validations are done.

6.2.4 System Testing:

The software or hardware testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. Application is an Intranet based application that can be accessed throughout the organization or a specified group/Dept. This application will access throughout the organization and accessing in different systems. Application is completely working in all the systems in the organization.

6.3 Test Cases:

6.3.1 Login Test Case:

SI # Test Case	UTC - 01
Name of Test	Unit Testing of **“Login”**
Item being Tested	Login Page
Sample Input	Username Password
Expected Output	Login Successful
Actual Output	Same as expected output
Remarks	Successful

Table 1: Test case of Login

6.3.2 Student registration Test case:

SI # Test Case	UTC - 01
Name of Test	Unit Testing of **“Student Registration”**
Item being Tested	Student Registration Page
Sample Inpu	Name Semester Department USN Section
Expected Output	Successfully Registered
Actual Output	Same as expected output
Remarks	Successful

Table 2: Test case of student registration

6.3.3 Course registration Test case:

SI # Test Case	UTC - 01
Name of Test	Unit Testing of **“Course Registration”**
Item being Tested	Course Registration Page
Sample Input	Course Title Course Code Registration Type Attempts Credits
Expected Output	Successfully Registered
Actual Output	Same as expected output
Remarks	Successful

Table 3: Test case of course registration

6.3.4 System Testing (version compatibility):

SI # Test Case	STC - 01
Name of Test	System Testing of **“PHP version”**
Item being Tested	PHP compatibility
Sample Input	Execute the application on v 5.5 onwards
Expected Output	Best Performance on v 7
Actual Output	Same as expected output
Remarks	Successful

Table 4: Test case for PHP version

7. CONCLUSION

This project is relatively simple to understand and implement. If fulfills all the current requirements of college to complete course registration without much manual work. It is user friendly, student with basic computer skills can easily use the system. New features and modules can be added into the system as per user requirement. The project is flexible in that aspect. It is economical as well in the long run, requiring less manpower and money. So that it would be more comfortable and easy to implement and also to work. It resembles an up gradation with the ongoing technology in computer science as well as in development of small and useful applications as per the customer requirements.

It improves the level of automation in the college by reducing task of filling form and manual works carried out as a part of documentation for every semester in college office. They can keep entries in digital format and reduce use to hard copies and wait for other authentication process. It also saves a day of work that has to be carried out by students, faculties and HOD as well as office staff.

8. FUTURE ENHANCMENT

This proposed system can be enhanced with more features and modules so that it will be up to date and serves the newly added requirements of the end user. In future this web application can be connected with provisional result sheet so that if there is any failures and if its exceeding limit of credits to apply per semester it won’t allow students to register for further semester until unless student has cleared the backlogs and in case student have to withdraw the course he/she may drop a course with permission and add that same dropped course to the semester he/she is willing to study. It can have proctor module improved so that proctor can comment on what is wrong and what must be corrected it goes as a notification once the student logs in to his/her account.

Mobile application can be developed in order to ease the procedure micro service is best compared to macro service. Today’s generation heading towards the advanced technologies every work must be completed in one click is the expectation of end user including us. Waiting time for server to communicate on the request and response part is reduced to seconds, people need that service to be so quick and easy, auto generation and responsive application. Students can have an interactive registration every semester and in one login id the student’s complete course details and results can be stored. Also an integrated SGPA and CGPA calculator can also be implemented so that students can see what must be improved and proctors can also view and talk to students and improvement methodologies.

9. BIBLIOGRAPHY

We would like to specify the names of some books which we have refered that have paid us a lot and have helped us to complete this project successfully they are:

1. Learning PHP, MySQL & JavaScript,5th Edition with jQuery,CSS & HTML5 book by Robin Nixon, published by O'Reilly Media.
2. Learning jQuery book by Jonathan Chaffer and Karl Swedberg.
3. HTML & CSS: Design and Build Web Sites book by Jon Duckett.
4. Learning Web Design book by Jennifer Niederst Robbins.

We would like to specify the names of some web sites which we have refered that have paid us a lot and have helped us to complete this project successfully they are:

1. https://www.tutorialspoint.com/sql/
2. https://www.w3schools.com/html/
3. https://stackoverflow.com/questions/6088027/how-to-retrieve-users-information-from-the-database-for-admin-in-php
4. https://www.w3schools.com/php/
5. https://www.slideshare.net/mobile/HabiburRahman23/online-course-registration-system-development-software-engineering-project-presentation