1. INTRODUCTION

1.1 OVERVIEW

The project titled as “TEACHER-STUDENT PORTAL “is a web based application. It shows the main details about the college. The clients can connect to the server. Each user first makes their login to server to show their availability. This report discusses the result of the work done in Teacher-Student Portal discusses varies software’s used and various platforms on which the work is done and will be done. It aims for the development of an application for facilitating the students so that without wasting their time they can get all the study material without visiting again and again to teachers.

1.2 OBJECTIVE

The main objective for the development of an application for facilitating the students so that without wasting their time they can get all the study material without visiting again and again to teachers and clear their doubt by asking the query to the subject teacher.

1.3 SCOPE

This is basically designed for facilitating students of our college as here the Student’s information is stored according to their College ID. And College ID includes the college identity. This can also be designed for other institutes by making minor changes to it. It can be used efficiently and also users can be registered by the admin by just filling the form.

2 TOOLS/ ENVIRONMENT USED

2.1 NETBEANS IDE 8.1

NetBeans IDE is an open-source integrated development environment. NetBeans IDE supports development of all Java application types Java ME, web, EJB and mobile applications) out of the box. Among other features are an Ant-based project system, Maven support, refactoring’s, version control (supporting CVS. Subversion, Git and Clear case).

MODULARITY: All the functions of the IDE are provided by modules. Each module provides a well-defined function, such as support for the Java language, editing, or support for the CVS versioning system, and SVN. NetBeans contains all the modules needed for Java development in a single download, allowing the user to start working immediately. Modules also allow NetBeans to be extended. New features, such as support for other programming languages, can be added by installing additional modules. For instance, Sun Studio, Sun Java Studio Enterprise, and Sun Java Studio Creator from Sun Microsystems are all based on the NetBeans IDE.

NetBeans IDE 8.1 was released on November 4, 2015

2.2 MYSQL

MySQL is written in C and C++. Its SQL parser is written in yacc, but it uses a home-brewed lexical analyzer.[18] MySQL works on many system platforms, including AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, macOS, Microsoft Windows, NetBSD, Novell NetWare , OpenBSD , OpenSolaris , OS/2 Warp , QNX , Oracle Solaris, Symbian , SunOS, SCO OpenServer, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists.

The MySQL server software itself and the client libraries use dual-licensing distribution. They are offered under GPL version 2, beginning from 28 June 2000 (which in 2009 has been extended with a FLOSS License Exception) or to use a proprietary license.

Support can be obtained from the official manual. Free support additionally is available in different IRC channels and forums. Oracle offers paid support via its MySQL Enterprise products. They differ in the scope of services and in price. Additionally, a number of third party organisations exist to provide support and services, including MariaDB and Percona.

2.3 HTML

HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages.

- Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus the link available on a webpage are called Hypertext.
- As its name suggests, HTML is a Markup Language which means you use HTML to simply "mark up" a text document with tags that tell a Web browser how to structure it to display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
</body>
</html>
```

2.4 CSS

CSS stands for Cascading Style Sheets.CSS describes how HTML elements are to be displayed on screen, paper, or in other media.CSS saves a lot of work. It can control the layout of multiple web pages all at once. External stylesheets are stored in CSS files.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the mark up languages HTML or XHTML.

2.5 BOOTSTRAP

Bootstrap is a free front-end framework for faster and easier web development. Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins. Bootstrap also gives you the ability to easily create responsive designs.

2.6 SERVLET

Servlet can be described in many ways, depending on the context.

- Servlet is a technology i.e. used to create web application.
- Servlet is an API that provides many interfaces and classes including documentations.
- Servlet is an interface that must be implemented for creating any servlet.
- Servlet is a class that extend the capabilities of the servers and respond to the incoming request. It can respond to any type of requests.

- Servlet is a web component that is deployed on the server to create dynamic web page.

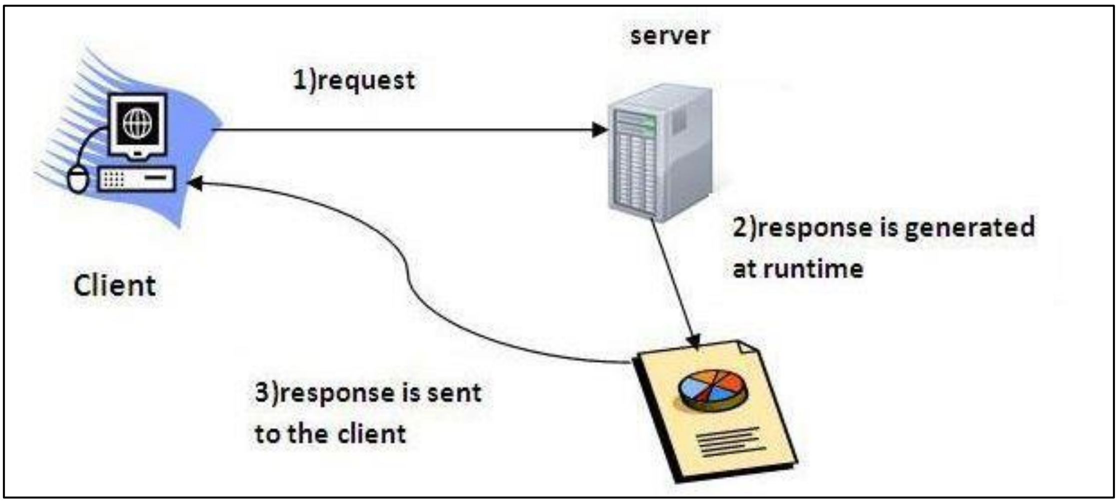

server
1)request
Client
2)response is generated at runtime
3)response is sent to the client

Figure 2.1 Working of Servlet

3. OVERALL DESCRIPTION

This section gives background information about specific requirements of the web based integrated development environment service to be developed in brief. Although we will not describe every requirement in detail, this section will describe the factors that affect the final product.

3.1 PRODUCT PERSPECTIVE

Website will be main user interface where users can operate all the provided functionality. From the student point of view, student will have to functionality to download the files. The Teacher will upload the notes or the files related to the course. The student will download the notes from the website.

3.2. PRODUCT FEATURES

The functions are divided according to the user types such as:- Administrator:-The function of the administrator is to add/edit exams in the test module. Candidate:-The function of the student is to update his/her profile and give various exams.

The function are divided according to the user login such as:

1. Admin:

The function of the admin is to add/delete/update information of the student/teacher/course.

2. Student

The function of the student is it will download the notes, ask the query to the teacher and also update its own profile.

3. Teacher

The function of the teacher is it will upload the notes for the student and also give the answer of the query ask by the student. The teacher also update its own profile.

The website that is made is helpful for the Student and the Teacher. They will be able to communicate to each other by the way of the website. The teacher upload the load to the website & the student’s will download that notes that will be in the upload by the teacher and student’s also ask the query related to the notes that will be uploaded by the teachers.

3.3 USER CLASSES AND CHARACTERISTICS

The various user of this website are classified into the three types:

1. Admin
2. Student
3. Teacher

This website required the user to have the Characteristics such that student should able to communicate to the teacher about their query that will be ask by the student.

3.4 OPERATING ENVIRONMENT

This website is based on the internet so it will run on any operating system with internet access through a web browser.

3.5 DESIGN AND IMPLEMENTATION CONSTRAINTS

The student is allow to download the notes or the file that that will be uploaded by the teacher .while asking for the query the student can also show their profile or also see the event that will be show by the admin.

3.6 ASSUMPTIONS AND DEPENDENCIES

Proper working of this website is dependent on the internet connectivity of the user computers

- It is assumed that the user has basic knowledge of the system.
- It is assumed that the data entered by the user while registering is true.
- It is assumed that the only the student of our college will be use this website

4. EXTERNAL INTERFACE REQUIREMENTS

4.1 USER INTERFACES

REGISTRATION SCREEN:

Various fields available on this screen will be:-

- ID
- Password
- Email id
- Father name
- Name

LOGIN SCREEN

- Login id
- Password

4.2 HARDWARE INTERFACES

User is supposed to interact with the input and output hardware peripherals such as the monitor and the keyboard, mouse. The User interface depends upon the service provider such as Hotmail, Gmail etc.

4.3 SOFTWARE INTERFACES

- Any windows based operating system.
- MySQL Server Database
- Java servlet.

5. UML MODELING

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

So, UML can be described as a general purpose visual modelling language to visualize, specify, construct and document software system. Although UML is generally used to model software systems but it is not limited within this boundary. It is also used to model non-software systems as well like process flow in a manufacturing unit etc.

UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. There are two broad categories of diagrams and then are again divided into sub-categories:

- **Structural Diagrams**

The structural diagrams represent the static aspect of the system. These static aspects represent those parts of a diagram which forms the main structure and therefore stable. These static parts are representing by classes, interfaces, objects, components and nodes. The four structural diagrams are:

- Class diagram
- Object diagram
- Component diagram
- Deployment diagram

- **Behavioural Diagrams**

Any system can have two aspects, static and dynamic. So, a model is considered as complete when both the aspects are covered fully. Behavioral diagrams basically capture the dynamic aspect of a system. Dynamic aspect can be further described as the changing/moving parts of a system. UML has the following five types of behavioral diagrams:

- Use case diagram
- Sequence diagram
- Collaboration diagram
- State Transition diagram
- Activity diagram

USE CASE DIAGRAM

Use case diagrams are considered for high level requirement analysis of a system. So, when the requirements of a system are analyzed the functionalities are captured in use cases. So, we can say that use cases are nothing but the system functionalities written in an organized manner. Now the second things which are relevant to the use cases are the actors. Actors can be defined as something that interacts with the system. The actors can be human user, some internal applications or may be some external applications. So, in a brief when we are planning to draw a use case diagram we should have the following items identified.

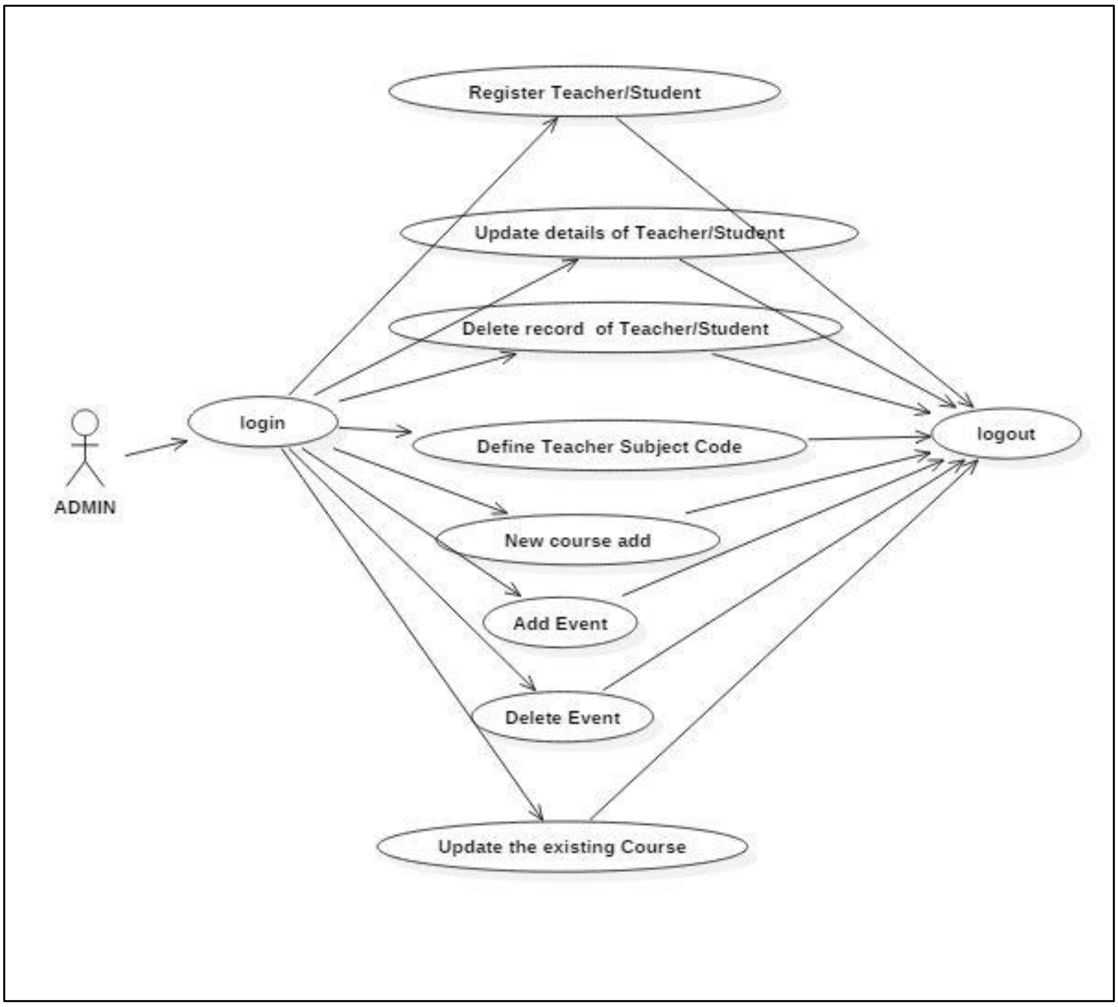

Register Teacher/Student
Update details of Teacher/Student
Delete record of Teacher/Student
ADMIN
login
Define Teacher Subject Code
logout
New course add
Add Event
Delete Event
Update the existing Course

Figure 4.1 Use Case of Admin

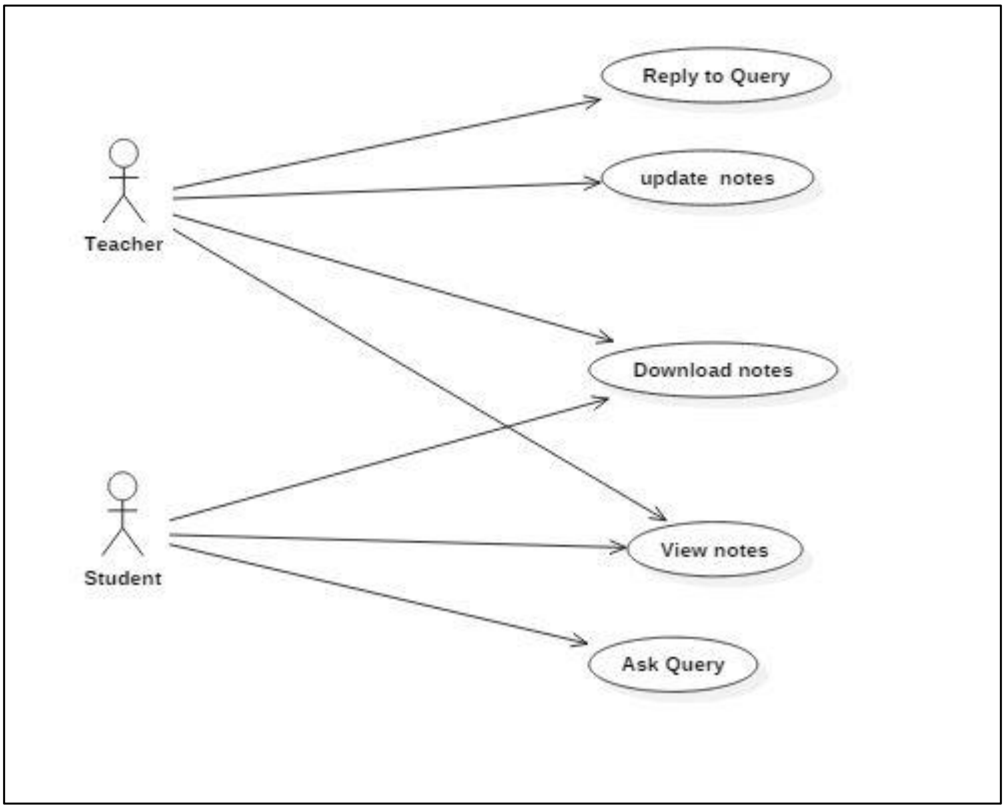


Figure 4.2 Use case of Student & Teacher

DFD

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling 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 process or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

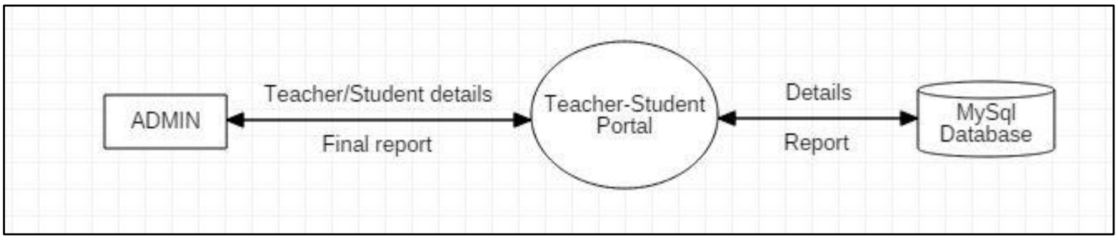


Figure 4.3 DFD 0-Level

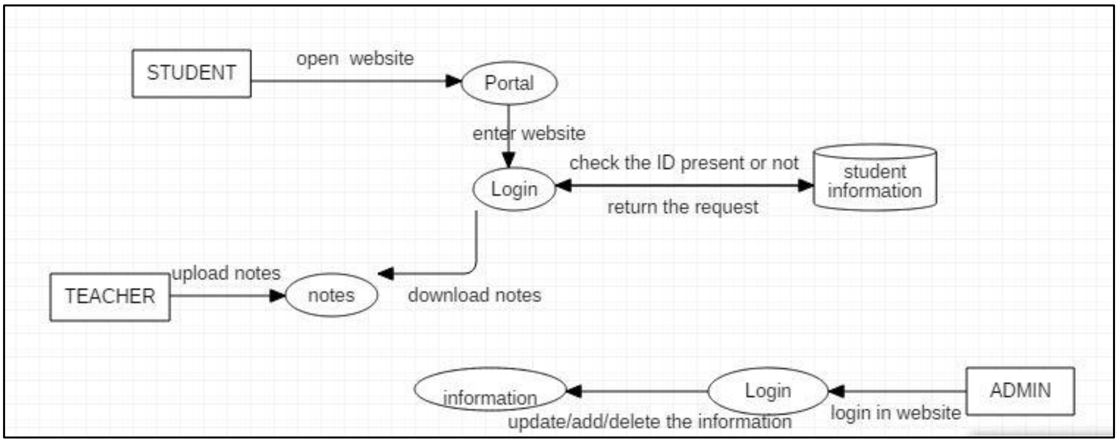


Figure4.4 1 level DFD

E-R DIAGRAM

An entity–relationship model (ER model) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.

In software engineering an ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure that can be implemented in a database, typically a relational database.

Entity–relationship modeling was developed for database design by Peter Chen and published in a 1976 paper. However, variants of the idea existed previously. Some ER modelers show super and subtype entities connected by generalization-specialization relationships, and an ER model can be used also in the specification of domain-specific ontologies. An entity–relationship model is usually the result of systematic analysis to define and describe what is important to processes in an area of a business. It does not define the business processes; it only presents a business data schema in graphical form. It is usually drawn in a graphical form as boxes (entities) that are connected by lines (relationships) which express the associations and dependencies between entities. An ER model can also be expressed in a verbal form, for example: one building may be divided into zero or more apartments, but one apartment can only be located in one building.

Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys". Diagrams created to represent attributes as well as entities and relationships may be called entity-attribute-relationship diagrams, rather than entity-relationship models.

An ER model is typically implemented as a database. In a simple relational database implementation, each row of a table represents one instance of an entity type, and each field in a table represents an attribute type. In a relational database a relationship between entities is implemented by storing the primary key of one entity as a pointer or "foreign key" in the table of another entity

There is a tradition for ER/data models to be built at two or three levels of abstraction. Note that the conceptual-logical-physical hierarchy below is used in other kinds of specification, and is different from the three schema approach to software engineering.

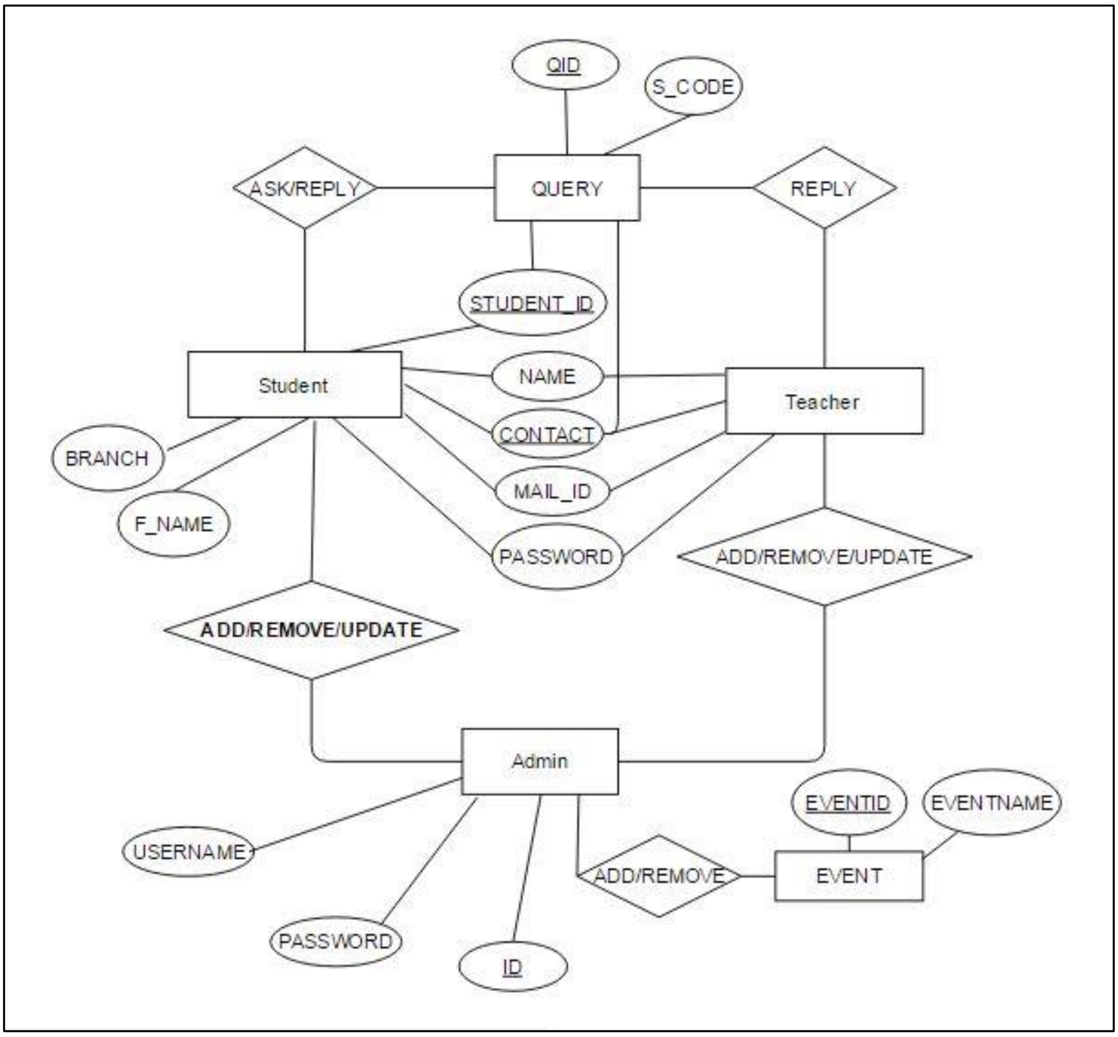


Figure 4.5 ER diagram

DATA DICTIONARY

Field	Type	Null	Key
ID	Varchar(50)	No	Primary
Password	Varchar(30)	Yes	
Username	Varchar(30)	Yes	

Table 1 Admin login table

Field	Type	Null	Key
Name	Varchar(40)	Yes	
Password	Varchar(30)	Yes	
Contact_no	Bigint(20)	No	Primary
Mail_id	Varchar(30)	Yes	

Table 2 Teacher login table

Field	Type	Null	Key
Student_id	Varchar(20)	No	Primary
Password	Varchar(50)	Yes	
S_name	Varchar(50)	Yes	
Phone_no	Varchar(20)	Yes	
Father_name	Varchar(50)	Yes	
Branch	Varchar(50)	Yes	
Email	Varchar(50)	Yes	

Table 3 Student Login table

Field	Type	Null	Key
Code	Varchar(10)	No	Primary
Subject	Varchar(30)	Yes	

Table 4 Subject code table

Field	Type	Null	Key
Sem	Int(10)	No	Primary
Branch	Varchar(30)	No	Primary
Sub1	Varchar(30)	Yes	
Sub2	Varchar(30)	Yes	

Sub3	Varchar(30)	Yes	
Sub4	Varchar(30)	Yes	
Sub5	Varchar(30)	Yes	
Sub6	Varchar(30)	Yes	
Sub7	Varchar(30)	Yes	
Sub8	Varchar(30)	Yes	
Sub9	Varchar(30)	Yes	

Table 5 Course of every branch & table

Field	Type	Null	Key
Eventid	Int(10)	No	Primary
Eventname	Varchar(100)	Yes	

Table 6 Event

Field	**Type**	**Null**	**Key**
Subject	Varchar(50)	Yes	
Topic	Varchar(50)	Yes	
Date	Varchar(50)	Yes	
Time	Varchar(50)	Yes	
Name	Varchar(50)	No	Primary
Path	Varchar(50)	Yes	

Table 7 for uploading the pdf files

Field	**Type**	**Null**	**Key**
Id	Varchar(50)	No	Primary
Code	Varchar(20)	No	Primary

Table 8 for teacher define subject

Field	**Type**	**Null**	**Key**
Qid	Varchar(50)	No	Primary
Sid	Varchar(50)	Yes	
S_code	Varchar(50)	Yes	
Counter	Varchar(50)	No	Primary
Query	Varchar(500)	Yes	
Tid	Varchar(50)	Yes	

Table 9 Student Query

Field	Type	Null	Key
Qid	Varchar(50)	No	Primary
Tid	Varchar(50)	Yes	
Sid	Varchar(50)	Yes	
S_code	Varchar(50)	Yes	
Query	Varchar(500)	Yes	
Counter	Varchar(50)	No	Primary

Table 10 teacher query

6. SYSTEM SPECIFICATION:

6.1 SOFTWARE SPECIFICATION:

- Operating System: Windows 7, 8, 8.1,10
- Software tool : Net Beans 8.1
- Front End : HTML5, CSS
- Back End: Java
- Database: MySQL 5.5
- Local Host Server : XAMPP Server

6.2 HARDWARE SPECIFICATION:

- RAM : 400 MB
- System: HP, Dell etc.
- Processor: Intel(R) Core(TM) i3

7. DESIGN DOCUMENT

There are 3 module in this:

1. Admin

2. Teacher

3. Student

1. Admin

The Admin has the complete control in the sub modules, or the task performed on the modules. The Admin will do the many functionality such that adding the new teacher/student, update the details of the teacher/student, delete the record of the teacher by their phone number and delete the record of the student by their unique college ID. Also the Admin will add the new course according to the selection of the branch and the semester, also perform the updating of the course by choosing the branch & semester. Another function of the admin is that it will add and delete the event that will be shown to all the student & the teacher which are login.

2. TEACHER

The teacher has a control to only on its own profile. Teacher will login by their phone number that will be store in the database. The teacher when login then see his own profile & also that which subject it will taught to the student. The teacher can also show the Query that will be ask by the student. The main work of the Teacher is that it will upload the Notes that will be useful for the student in the study. Also can view or delete the notes that will uploaded by itself.

3. STUDENT

The student modules is that in which the main work is that the Student can download the notes which will be uploaded by the teacher. The student can download or show the notes and also ask the Query related to the notes or any class lecture. The student when login can also see the event that will be added by the admin.

8. PROGRAM CODE

ADMIN.JAVA

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class admin extends HttpServlet {
    static String ID,pass;
    static String username;
    static int flag;
```

```
  protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
          throws ServletException, IOException {
      response.setContentType("text/html;charset=UTF-8");
    String output="";
                ID=request.getParameter("name");
                pass=request.getParameter("pass");
      int i,j;
           Cookie cookie = null;
                  Cookie[] cookies = null;
               cookies = request.getCookies();
                for ( i = 0; i < cookies.length; i++)
                {
                     cookie = cookies[i];
                  if(cookie.getName().equals("name"))
                {    if(ID==null)
                     ID=cookie.getValue();
                     //s1=ID;
                     System.out.println("Id = "+ID);
                }
                  if(cookie.getName().equals("pass"))
                  {   if(pass==null)
                       pass=cookie.getValue();
                }}
               PrintWriter writer = response.getWriter();
               database db=new database();
               Connection con=db.getCon();
              try
              {
                 System.out.println("error 1");
```

```
                    PreparedStatement ps = con.prepareStatement("SELECT
ID,password,username FROM admin_login where ID = ?");
                    ps.setString(1,ID);
                    System.out.println("before the exection ");
                    ResultSet rs = ps.executeQuery();
                    System.out.println("after the exection "+rs);
                     System.out.println("error  2");
                     if(rs.next())
                          {   String password;
                      password=rs.getString("password");
                     System.out.println("error 3");
                      System.out.println("password is"+password);
                      username=rs.getString("username");
                      Cookie cookie2 =new Cookie("username",username);
                      response.addCookie(cookie2);
                      System.out.println("password is ="+password +"matching password is
="+pass);
                      if(password.equals(pass))
                         {
                                flag=1;
                          }              }
               }catch(Exception e)
                {
                System.out.println(e);
                }
                 if(flag==1)
                 {                    savedinfo si = new savedinfo();
                  output+="<!DOCTYPE html>\n" +
"<html>\n";
                  output+=si.headpart()+"<body onload=\"startTime()\" style=\"\">\n"
+si.headerpart() +si.menupart();
```

```
output+="    </body>\n" +
"</html>\n" +
"";writer.println(output);
                  }
}
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>
}
```

LOGIN.JAVA

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class login extends HttpServlet {
 @Override
```

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    String a="1";
    String b="2";
    String d="admin";
 try{
              String name=request.getParameter("name");
              String pass=request.getParameter("pass");
              System.out.println("name="+name);
              System.out.println("pass="+pass);
                   Cookie cookie1 =new Cookie("name",name);
                   Cookie cookie2 =new Cookie("pass",pass);
                   response.addCookie(cookie1);
                   response.addCookie(cookie2);
         String c=name.substring(0,1);
         String z=name.substring(0,5);
         System.out.println(c);
         System.out.println(z);
         System.out.println("in try");
         //request.setAttribute("name",name);
         //request.setAttribute("pass",pass);
      if(b.equals(c)||a.equals(c))
         {
            System.out.println("in student");
            RequestDispatcher rd=request.getRequestDispatcher("student");
            rd.forward(request,response);
         }
      else if(d.equals(z))
              {
                   System.out.println("in admin page");
                   RequestDispatcher rd=request.getRequestDispatcher("admin");
```

```
                    rd.forward(request,response);
               }
        else
         {
                  System.out.println("in teacher page");
                  RequestDispatcher rd=request.getRequestDispatcher("T_login");
                  rd.forward(request,response);
          }
}
         catch(Exception ex)
         {
System.out.println(ex);
         }
   }
   @Override
   protected void doPost(HttpServletRequest request, HttpServletResponse response)
          throws ServletException, IOException {
      doGet(request, response);
   }


}
```

TEACHER_LOGIN.JAVA

```
import java.io.IOException;
import java.io.PrintWriter;
import java.io.Writer;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class T_login extends HttpServlet {
    static int i,j,k,l,m,flag;
    static String ID,pass, password,output,name;
    static  String yr1[],yr2[],yr3[],yr4[];
    static String evn[]= new String[30];
    String yrsub1[],yrsub2[],yrsub3[],yrsub4[];
    boolean year1,year2,year3,year4,error;
    @Override
      protected void doGet(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
           int profile=0;
                ID=request.getParameter("name");
                pass=request.getParameter("pass");
           PrintWriter writer = response.getWriter();
            System.out.println("in teacher servlet");
            int i,j;
                  Cookie cookie = null;
                    Cookie[] cookies = null;
                  cookies = request.getCookies();
                     for ( i = 0; i < cookies.length; i++)
                      {
                         cookie = cookies[i];
                         if(cookie.getName().equals("name"))
                          {   if(ID==null)
                             ID=cookie.getValue();
                             System.out.println("Id = "+ID);
                          }
                         if(cookie.getName().equals("pass"))
                          {   if(pass==null)
                             pass=cookie.getValue();
                          }}
            Cookie fname = new Cookie("ID",ID);
            fname.setMaxAge(60*60*24);
            response.addCookie( fname );
            teacher_record tr = new teacher_record(ID);
            name = tr.name();
            tr.data();
            error=tr.error;
            yr1=tr.yrr1();
            year1=tr.year1status();
            yr2=tr.yrr2();
            year2=tr.year2status();
            yr3=tr.yrr3();
            year3=tr.year3status();
            yr4=tr.yrr4();
            year4=tr.year4status();
            database db=new database();
               Connection con=db.getCon();
              try{
```

```
              System.out.println("before select into t_login");
               PreparedStatement ps = con.prepareStatement("SELECT
password FROM teacher_login where contact_no = ?");
               ps.setString(1,ID);
               System.out.println("before the exection ");
               ResultSet rs = ps.executeQuery();
                      System.out.println("after the exection "+rs);
                 if(rs.next())
                  {
                 password=rs.getString("password");
                 System.out.println("password is"+password);
                 if(password.equals(pass))
                   {
                     flag=1;
                   }
}
 }
          catch(Exception ex)
          {
             System.out.println("exceptionnnnnnnnnnnn");
             System.out.println(ex);
          }
            System.out.println("value of flag is :-" + flag);
                  try
              {
                 PreparedStatement ps1 = con.prepareStatement("SELECT
* from event");
```

```
            ResultSet rs =ps1.executeQuery();
            int event=0;
            while(rs.next())
             {
             evn[event]=rs.getString(2);
             System.out.println("data inside event ="+evn[event]);
             event++;
           }
          }
         catch(Exception e)
          {
            System.out.println("Excetption in event"+e);
          }
         try{con.close(); }catch(Exception e){System.out.println(e);}
       if(flag==0)
        {
writer.println("invalid login");
        }
        else if(!error)
        {


          output="<!DOCTYPE html>\n";
  output+="<html>\n";
    output+="<head>\n";
      output+=" <title>welcome to bkbiet Teacher panel </title>\n";
       output+="<meta charset=\"UTF-8\">\n";
```

```
            output+="<meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0\">\n";
            output+="<link rel=\"stylesheet\" href=\"css/bootstrap.min.css\">\n";
            output+="<script src=\"jquery/jquery-3.1.1.min.js\"></script>\n";
            output+="<script src=\"js/bootstrap.min.js\"></script>\n";
            output+="<script src=\"https://ajax.googleapis.com/ajax/libs/jqueryui/1.11.4/jquery-ui.min.js\"></script>\n";
            output+="<link rel=\"stylesheet\" href=\"https://ajax.googleapis.com/ajax/libs/jqueryui/1.11.4/themes/smoothness/jquery-ui.css\">\n";
            output+="<script src=\"scripts/time.js\"></script>\n";
            output+="<script>\n" +
"                function abc()\n" +
"                {\n" +
"                    \n" +
"                var x = document.getElementById(\"ab\");\n" +
"                var y =x.text;\n" +
"                document.cookie = \"subject=\"+y;\n" +
"               \n" +
"                }\n" +
"            </script>";
            output+="<style type=\"text/css\">\n";
        output+="#navPanel {\n";
        output+="max-width:300px;\n";
        output+="}\n";
```

```
    output+=".panel-title > a {\n";
    output+="    color: inherit;\n";
        output+="display:block;\n";
        output+="text-decoration:none;\n";
    output+="} \n";
    output+=".panel-body {\n";
       output+=" padding: 0px;\n";
   output+=" }\n";
    output+=".list-group {\n";
        output+="padding-left: 0;\n";
        output+="margin-bottom: 0px;\n";
    output+="}\n";
     output+=".glyphicon-plus:before {\n";
       output+="content: \"\\2212\";\n";
    output+="}\n";
    output+=".collapsed .glyphicon-plus:before {\n";
       output+=" content: \"\\2b\";\n";
    output+="}\n";
            output+="</style>\n";
        output+="</head>\n";
        output+="<body onload=\"startTime()\">\n";
            output+="<div class=\"container\">\n";
                output+="<div class=\"row\" style=\" min-height: 150px;background-color: #1b6d85;padding: 0px; border-radius: 15px;\">\n";
                    output+="<div>\n";
                    output+="<div class=\"col-lg-2\" style=\"text-align: right;
float: left\"> \n";
                            output+="<p id=\"time\" style=\"padding-top:30px;
padding-bottom: 0px; color: #fff;font-variant:small-caps; font-
style:normal;  \"></p>\n";
                            output+="<p id=\"txt\" style=\"color: #fff; padding-top:
0px ;font-variant:small-caps; font-style:normal;\" ></p>\n";
                        output+="</div>\n";
                        output+="<div class=\"col-lg-8\" style=\"text-align:
center\"> <h1 style=\"color: #fff ;font-variant:small-caps; font-
style:normal; \">Teacher-Student Portal</h1>\n";
                          output+="<p> B.K. Birla Institute Of Engineering and
Technology</p>\n";
                          output+="</p>\n";
                   output+=" </div>\n";
                    output+=" <div class=\"col-lg-2\"
style=\"background:transparent; text-align: center; float: right \" >\n";
                         output+="<p style=\" color: white\">\n";
                             output+="<br/>\n";
                             output+="Name :"+name+"<br />\n";
                             output+="Login Id :"+ ID+"<br/>\n";
                             output+="<br/><br/>\n";
                             output+="<a href=\"logout\" style=\" color:
white\">Logout</a>\n";
                        output+=" </p>       \n";
                     output+="</div>\n";
                    output+="</div>\n";
                output+="</div>\n";
```

```
            output+="</div>\n";
            // main part
            output+="<div class=\"container\" style=\"padding: 0px;
margin:auto; padding-top: 2px; border-radius: 15px;\">\n";


                 output+="<div class=\"col-md-3\" style=\"background-color:
#1b6d85;min-height: 500px; border: 1px white solid; border-radius: 15px
\">";
                     output+="<div style=\" text-align: center\">
<h4>SUBJECT</h4></div>";
                           output+="<div class=\"panel-group\" id=\"accordion\"
role=\"tablist\" aria-multiselectable=\"true\">";
                              // define menu bar dynamically
                           output+="<div class=\"panel panel-primary\">\n";
              output+="<div class=\"panel-heading\" role=\"tab\"
id=\"headingOne\">\n";
                output+="<h4 class=\"panel-title\">\n";
                 output+="<a role=\"button\" data-toggle=\"collapse\" data-
parent=\"#accordion\" href=\"\" aria-expanded=\"false\" aria-
controls=\"collapseOne\" class=\"collapsed\">\n";
                    output+="<a href=\"T_login\">Home</a>\n";
                 output+="</a>\n";
                output+="</h4>\n";
              output+="</div>\n";
           output+="</div>\n";
                  //define query in menu
           output+="<div class=\"panel panel-primary\">\n";
```

```
          output+="<div class=\"panel-heading\" role=\"tab\" id=\"headingOne\">\n";
            output+="<h4 class=\"panel-title\">\n";
             output+="<a role=\"button\" data-toggle=\"collapse\" data-parent=\"#accordion\" href=\"\" aria-expanded=\"false\" aria-controls=\"collapseOne\" class=\"collapsed\">\n";
               output+="<a href=\"subjectdetails?query=2\">Query</a>\n";
             output+="</a>\n";
            output+="</h4>\n";
          output+="</div>\n";
        output+="</div>\n";
             output+="<div class=\"panel panel-primary\">\n";
          output+="<div class=\"panel-heading\" role=\"tab\" id=\"headingOne\">\n";
            output+="<h4 class=\"panel-title\">\n";
             output+="<a role=\"button\" data-toggle=\"collapse\" data-parent=\"#accordion\" href=\"#collapseO\" aria-expanded=\"false\" aria-controls=\"collapseOne\" class=\"collapsed\">\n";
               output+="Deshbord <span class=\"glyphicon glyphicon-plus pull-right\" aria-hidden=\"true\"></span>\n";
             output+="</a>\n";
            output+="</h4>\n";
          output+="</div>\n";
        output+="<div id=\"collapseO\" class=\"panel-collapse collapse\" role=\"tabpanel\" aria-labelledby=\"headingOne\" aria-expanded=\"false\" style=\"height: 0px;\">\n";
           output+="<div class=\"panel-body\">\n";
             output+="<div class=\"list-group\">\n";
```

```
    output+="<a  href=\"subjectdetails?profile=1\" class=\"list-group-item\" ><span class=\"glyphicon glyphicon-link\" aria-hidden=\"true\"></span>\n";
            output+="profile</a>\n";
         output+="</div>\n";
        output+="</div>\n";
      output+="</div>\n";
     output+="</div>\n";
             if(year1) {
          output+="<div class=\"panel panel-primary\">";
    output+="<div class=\"panel-heading\" role=\"tab\" id=\"headingOne\">";
     output+="<h4 class=\"panel-title\">";
      output+="<a role=\"button\" data-toggle=\"collapse\" data-parent=\"#accordion\" href=\"#collapseOne\" aria-expanded=\"false\" aria-controls=\"collapseOne\" class=\"collapsed\">";
       output+="Year 1 <span class=\"glyphicon glyphicon-plus pull-right\" aria-hidden=\"true\"></span>";
      output+="</a>";
     output+="</h4>";
    output+="</div>";
    output+="<div id=\"collapseOne\" class=\"panel-collapse collapse\" role=\"tabpanel\" aria-labelledby=\"headingOne\" aria-expanded=\"false\" style=\"height: 0px;\">";
     output+="<div class=\"panel-body\">";
      output+="<div class=\"list-group\">";
      i=0;
```

```
            while(yr1[i]!=null)
              {
            output+="<a href=\"subjectdetails?subject="+yr1[i]+"\"
class=\"list-group-item\"> <span class=\"glyphicon glyphicon-link\" aria-
hidden=\"true\"></span>"+ yr1[i]+"</a>";
              i++;
              }
output+="</div>";
          output+="</div>";
        output+="</div>";
      output+="</div>";
                 }
                         if(year2){
      output+="<div class=\"panel panel-primary\">";
        output+="<div class=\"panel-heading\" role=\"tab\"
id=\"headingThree\">";
          output+="<h4 class=\"panel-title\">";
            output+="<a class=\"\" role=\"button\" data-toggle=\"collapse\"
data-parent=\"#accordion\" href=\"#collapseThree\" aria-
expanded=\"true\" aria-controls=\"collapseThree\">";
              output+="Year 2<span class=\"glyphicon glyphicon-plus pull-
right\" aria-hidden=\"true\"></span>";
            output+="</a>";
          output+="</h4>";
        output+="</div>";
```

```
    output+="<div id=\"collapseThree\" class=\"panel-collapse collapse in\" role=\"tabpanel\" aria-labelledby=\"headingThree\" aria-expanded=\"true\">";
     output+="<div class=\"panel-body\">";
      output+="<div class=\"list-group\">";
      j=0;
      while(yr2[j]!=null)
       {
      output+="<a href=\"subjectdetails?subject="+yr2[j]+"\" class=\"list-group-item\"> <span class=\"glyphicon glyphicon-link\" aria-hidden=\"true\"></span>"+ yr2[j]+"</a>";
       j++;
       }
output+="</div>";
     output+="</div>";
    output+="</div>";
   output+="</div>";
             }
        if(year3){
   output+="<div class=\"panel panel-primary\">";
    output+="<div class=\"panel-heading\" role=\"tab\" id=\"headingThree\">";
     output+="<h4 class=\"panel-title\">";
      output+="<a class=\"\" role=\"button\" data-toggle=\"collapse\" data-parent=\"#accordion\" href=\"#collapseThree\" aria-expanded=\"true\" aria-controls=\"collapseThree\">";
```

```
       output+="Year 3 <span class=\"glyphicon glyphicon-plus pull-
right\" aria-hidden=\"true\"></span>";
      output+="</a>";
     output+="</h4>";
    output+="</div>";
    output+="<div id=\"collapseThree\" class=\"panel-collapse collapse
in\" role=\"tabpanel\" aria-labelledby=\"headingThree\" aria-
expanded=\"true\">";
     output+="<div class=\"panel-body\">";
      output+="<div class=\"list-group\">";
        k=0;
      while(yr3[k]!=null)
       {
      output+="<a href=\"subjectdetails?subject="+yr3[k]+"\"
class=\"list-group-item\"> <span class=\"glyphicon glyphicon-link\" aria-
hidden=\"true\"></span>"+ yr3[k]+"</a>";
       k++;
       }
output+="</div>";
     output+="</div>";
    output+="</div>";
   output+="</div>";
             }
             if(year4){
   output+="<div class=\"panel panel-primary\">";
    output+="<div class=\"panel-heading\" role=\"tab\"
id=\"headingThree\">";
```

```
        output+="<h4 class=\"panel-title\">";
          output+="<a class=\"\" role=\"button\" data-toggle=\"collapse\"
data-parent=\"#accordion\" href=\"#collapseThree\" aria-
expanded=\"true\" aria-controls=\"collapseThree\">";
            output+="Year 4 <span class=\"glyphicon glyphicon-plus pull-
right\" aria-hidden=\"true\"></span>";
          output+="</a>";
        output+="</h4>";
      output+="</div>";
      output+="<div id=\"collapseThree\" class=\"panel-collapse collapse
in\" role=\"tabpanel\" aria-labelledby=\"headingThree\" aria-
expanded=\"true\">";
        output+="<div class=\"panel-body\">";
          output+="<div class=\"list-group\">";
l=0;
          while(yr4[l]!=null)
            {
          output+="<a href=\"subjectdetails?subject="+yr4[l]+"\"
class=\"list-group-item\"> <span class=\"glyphicon glyphicon-link\" aria-
hidden=\"true\"></span>"+ yr4[l]+"</a>";
            l++;
            }
output+="</div>";
        output+="</div>";
      output+="</div>";
    output+="</div>";
                      }
```

```
output+="</div>";
            output+="</div>";
//center division
            output+="<div class=\"col-md-6\" style=\"min-height: 500px;
background-color: #1b6d85;  margin: 0px; border: 1px white solid;
border-radius: 15px\">\n";
if(profile==1)
            {
            }
            else if(profile!=2)
            {
output+="<div style=\"text-align: center; display: inline;\">\n";
                    output+="  <h3 style=\"text-align: center; color:white\">
About B K Birla Institute of Engineering & Technology -
BKBIET</h3>\n" +
"\n" +
"<h5 style=\"text-align:left\">B K Birla Institute of Engineering &
Technology (BKBIET) is a well known name in the field of engineering
and tech streams. \n" +
"The institute is located in Pilani, which comes under the Jhunjhunu
district, Rajasthan.\n" +
"BKBIET was set up in 2007 with the vision is “to build a truly different
institution of engineering and technology in Rajasthan, an institution with
a marked difference in quality to serve the local people in global
arena”.\n" +
"The college is affiliated to Rajasthan Technical University (RTU) and
approved by All India Council of Technical Education (AICTE).\n" +
"The mission of BKBIET is “to build an effervescent community of
engineers, where faculty and students are cronies in a mutually
inspirational education process, engrossed in learning. \n" +
"This learning process would lead to the inventions and then to
discoveries”. \n" +
"BKBIET has a large campus spread over 18 acres of land. \n" +
"The campus comprises of pool of facilities with state of the art
infrastructure</h5>\n" +
"\n" +
"\n" +
"\n" +
"<h3 style=\"text-align: center; color:white\">Vision</h3>\n" +
"\n" +
"In the age of globalization, to build a truly different institution of
engineering and technology in Rajasthan, an institution with a marked
difference in quality to serve the local people in global arena, has been
our vision – the vision of Birlas.\n" +
"\n" +
" \n" +
"\n" +
"<h3 style=\"text-align: center; color:white\">Mission</h3>\n" +
"\n" +
"Our aim is to build an effervescent community of engineers, where
faculty and students are cronies in a mutually inspirational education
process, engrossed in learning. This learning process would lead to the
inventions and then to discoveries\n" +
"";
                    output+="</div>\n";
```

```
            }
                output+="</div>\n";
            //center div end
               output+="<div class=\"col-md-3\" style=\" text-align: center;background-color: #1b6d85;min-height: 500px; border: 1px white solid; border-radius: 15px\">\n";
                  output+="<h4 id=\"aa\">  Event And Notice </h4>\n";
                  output+="<marquee direction=\"up\" scrolldelay=\"200\" style=\"text-align:center; width: 100%; height:400px\" onmouseover=\"this.stop();\" onmouseout=\"this.start();\">\n";
                 // show event into scrolling tag
                   int ee=0;
                   while(evn[ee]!=null)
                   {
                  output+="<div id=\"\" style=\" height:auto; width: 100%; border-bottom: 1px solid #000; color: white; padding: 10px\">\n";
                  output+=evn[ee]+"\n";
                  ee++;
                  output+="</div>\n";
                   }
                  output+="</marquee>\n";
               output+="</div>\n";
          output+=" </div>\n";


      output+="</body>\n";
   output+="</html>\n";
writer.println(output);
```

```
}else
                {
writer.println("error into data base contivity please try after sometime ");
                }
    }
    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        doGet(request, response);
    }
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>

}
```

HOME.HTML

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>home page</title>
        <link rel="stylesheet" href="css/bootstrap.min.css">
        <script src="jquery/jquery-3.1.1.min.js"></script>
        <script src="js/bootstrap.min.js"></script>
        <script>
function startTime()
{
    var today = new Date();
    var h = today.getHours();
    var m = today.getMinutes();
    var s = today.getSeconds();
    m = checkTime(m);
    s = checkTime(s);
    document.getElementById('txt').innerHTML =
    h + ":" + m + ":" + s;
    var t = setTimeout(startTime, 500);
}
function checkTime(i)
{
    if (i < 10) {i = "0" + i};  // add zero in front of numbers < 10
    return i;
}
var monthName = new Array('January', 'Febuary', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December');
var hourap = new Array(12, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11);
function showTime()
{
    var dateObj = new Date();
    var day = dateObj.getDate(), month = dateObj.getMonth(), year = dateObj.getFullYear(), hour = dateObj.getHours(), minutes = (dateObj.getMinutes()<=9?'0'+dateObj.getMinutes():dateObj.getMinutes());
    var string  = monthName[month]+
    ' '+day+
    ', '+year;
    var timeDiv = document.getElementById('time');
    if(timeDiv !== null)
    {
        timeDiv.innerHTML = string;
        timeDiv.setAttribute('datetime',year+'-'+(month+1<=9?'0'+(month+1):month+1)+'-'+day+' '+hour+':'+minutes);
    };
};
setInterval(showTime,1000);
</script>
<style>
  .carousel-inner > .item > img,
  .carousel-inner > .item > a > img {
      width: 70%;
      margin: auto;
  }
  </style>
    </head>
    <body onload="startTime()" style="padding-right: 0px">
        <div class="modal fade" id="myModal" role="dialog" style="margin-top:200px;">
            <div class="modal-dialog modal-sm" style=" border-radius: 10px;">
      <div class="modal-content">
        <div class="modal-header bg-info">
          <button type="button" class="close" data-dismiss="modal">&times;</button>
          <h4 class="modal-title text-center">Login with us</h4>
        </div>
        <div class="modal-body">
            <form role="form" action="login" method="post">
                <div class=" form-group">
                    <label>Username</label>
                    <input type="text" class="form-control" placeholder="Username" id="name" name="name"/>
                </div>
                 <div class=" form-group">
            <label>Password</label>
                        <input type="password" class="form-control"
placeholder="password" id="pass" name="pass"/>
         </div>
        <div class=" btn-group">
         <input type="submit" class="btn btn-info" value="Login"/>
         <input type="reset" class="btn btn-primary" value="Reset" style="margin-
left:10px;"/>
         </div>
          </form>
    </div>
    <div class="modal-footer bg-info">
     <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
    </div>
   </div>
  </div>
 </div>
    <nav class="navbar" style="border-radius:0px; border-bottom:1px solid
#3c3c3c; padding:20px;">
  <div class="container">
   <div class="navbar-header">
     <a class="navbar-brand" href="#">
       <img src="images/bkbiet.jpg" alt="" height="70" width="70"></a>
   </div>
  <ul class="nav navbar-nav"style=" margin-top:20px;">
   <li class="active"><a href="#">Home</a></li>
  </ul>
  <button class="btn btn-danger navbar-btn pull-right" data-target="#myModal"
data-toggle="modal">Login</button>
 </div>
</nav>
```

```
   <div id="myCarousel" class="carousel slide" data-ride="carousel">
  <!-- Indicators -->
  <ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
    <li data-target="#myCarousel" data-slide-to="3"></li>
  </ol>
  <!-- Wrapper for slides -->
  <div class="carousel-inner" role="listbox">
    <div class="item active">
        <img src="images/Block-C.jpg" alt=""/>
    </div>
    <div class="item">
        <img src="images/Blocks-New.jpg" alt=""/>
    </div>
    <div class="item">
        <img src="images/clg.jpg" alt=""/>
       </div>
    <div class="item">
        <img src="images/main.jpg" alt=""/>
    </div>
  </div>
  <!-- Left and right controls -->
  <a class="left carousel-control" href="#myCarousel" role="button" data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>
    <span class="sr-only">Previous</span>
  </a>
  <a class="right carousel-control" href="#myCarousel" role="button" data-slide="next">
```

```
    <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>
    <span class="sr-only">Next</span>
  </a>
</div>
  </body>
</html>
```

9. TESTING

This type of testing is performed by developers before the setup is handed over to the testing team to formally execute the test cases. Unit testing is performed by the respective developers on the individual units of source code assigned areas. The developers use test data that is different from the test data of the quality assurance team. The goal of unit testing is to isolate each part of the program and show that individual parts are correct in terms of requirements and functionality.

9.1 ADMIN MODULE:

1. Test case: Login

Input: ID, Password.

Process: Click on the login link. If administrator enters ID and password correct it goes to the admin services otherwise displays the same page with an error message.

Output: Displays the admin services page.

2. Test case: add new register student

Input: ID, Password, name, branch, father name, email, phone no

Process: when click on register then the new student will be register.

Output: change in the database & same page is display again

3. Test case: delete student/teacher

Input: ID

Process: The Student/teacher will be deleted from the database and admin will do that.

Output: changes will take place in database

4. Test case: update student/teacher

Input: ID

Process: The Student/teacher will be update from the database and admin will do that.

Output: changes will take place in database & show the profile.

5. Test case: add/delete course

Input: semester, branch

Process: The admin will be add/delete the course from the database.

Output: changes will take place in database

9.2 TEACHER MODULE

1. Test case: login

Input: ID, Password

Process: Click on the login link. If teacher enters ID and password correct it goes to the teacher services otherwise displays the same page with an error message.

Output: Display teacher page

2. Test case: upload/delete notes

Input: PDF file

Process: the Teacher upload/delete/ view the pdf files

Output: Changes will take place in Database

3. Test case: reply query

Input: code, query

Process: the teacher will give reply to the query that will be ask by the student.

Output: Changes will take place in Database & give the reply to student.

9.3 STUDENT MODULE

1. Test case: login

Input: ID, Password

Process: Click on the login link. If student enters ID and password correct it goes to the student services otherwise displays the same page with an error message.

Output: Display student page

2. Test case: download notes

Input: PDF file

Process: click on download/view the pdf files on the student page

Output: download the file on the system

3. Test case: query

Input: code, query

Process: the student ask the query to the teacher which were taught to them

Output: Changes will take place in Database & show to the teacher

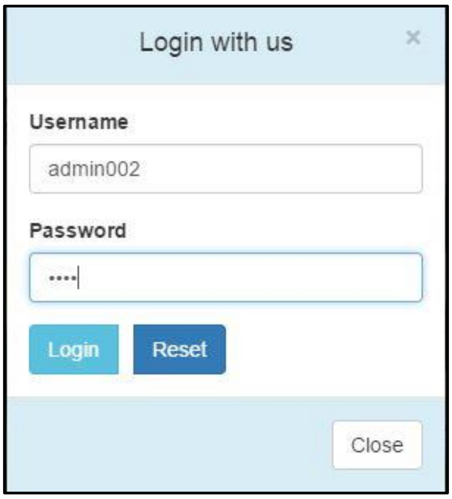
SYSTEM TESTING:

System testing tests the system as a whole. Once all the components are integrated, the application as a whole is tested rigorously to see that it meets the specified Quality Standards. This type of testing is performed by a specialized testing team.

System testing is important because of the following reasons:

- System testing is the first step in the Software Development Life Cycle, where the application is tested as a whole.
- The application is tested thoroughly to verify that it meets the functional and technical specifications.
- The application is tested in an environment that is very close to the production environment where the application will be deployed.
- System testing enables us to test, verify, and validate both the business requirements as well as the application architecture.

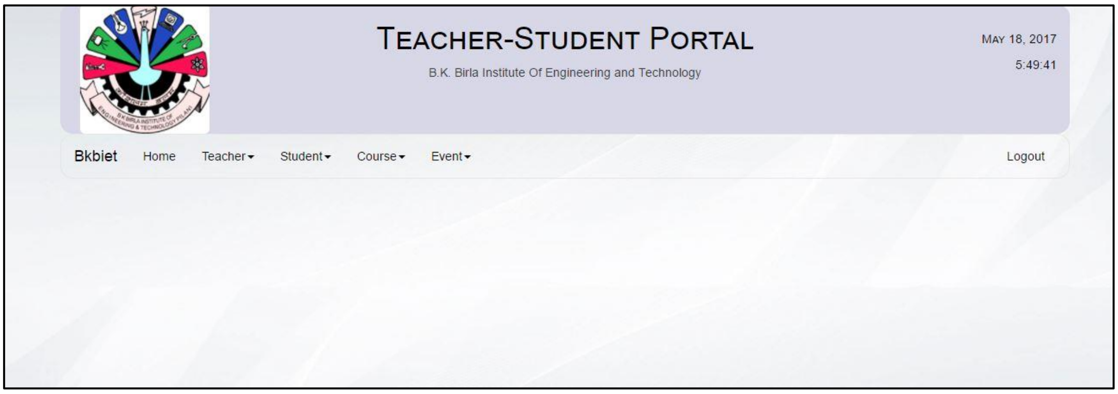
10. INPUT AND OUTPUT SCREEN


Login with us

Username: admin002

Password: ••••

Figure 10.1 Home login screen for enter into website

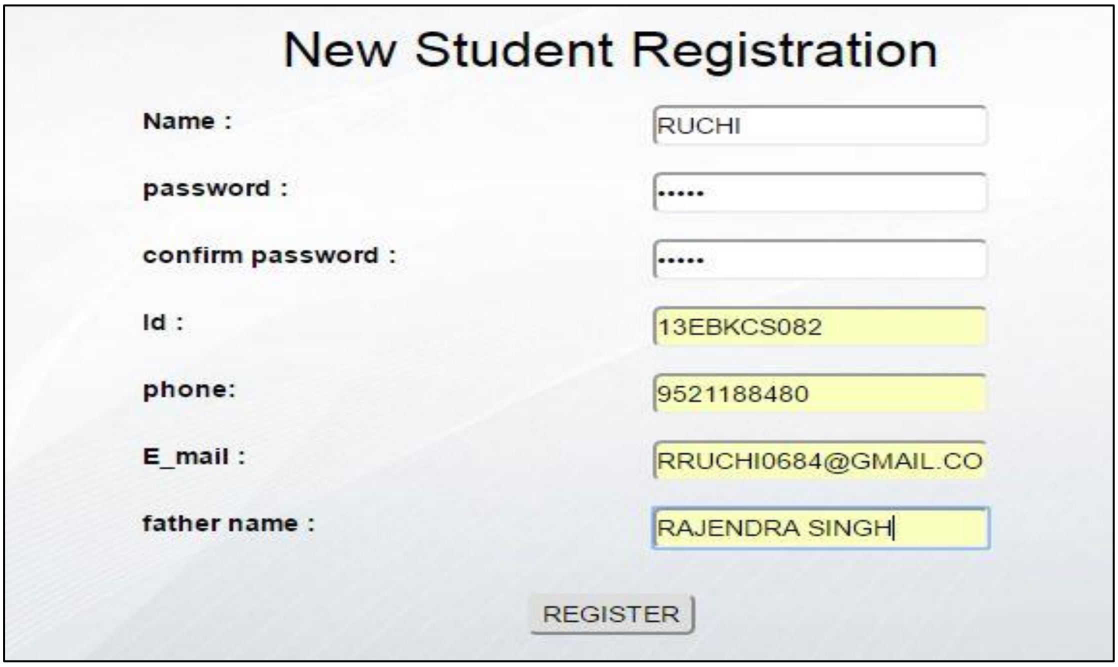


TEACHER-STUDENT PORTAL

B.K. Birla Institute Of Engineering and Technology

MAY 18, 2017
5:49:41

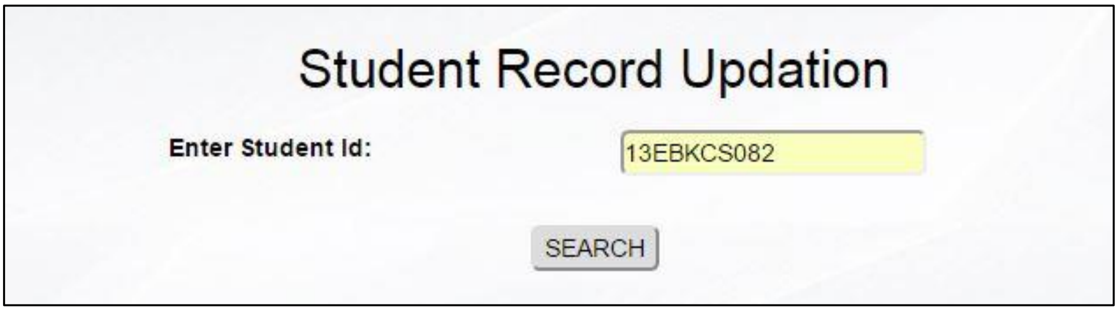
Figure 10.2 After login by the admin get the output



New Student Registration

Name :	RUCHI
password :	•••••
confirm password :	•••••
Id :	13EBKCS082
phone:	9521188480
E_mail :	RRUCHI0684@GMAIL.CO
father name :	RAJENDRA SINGH

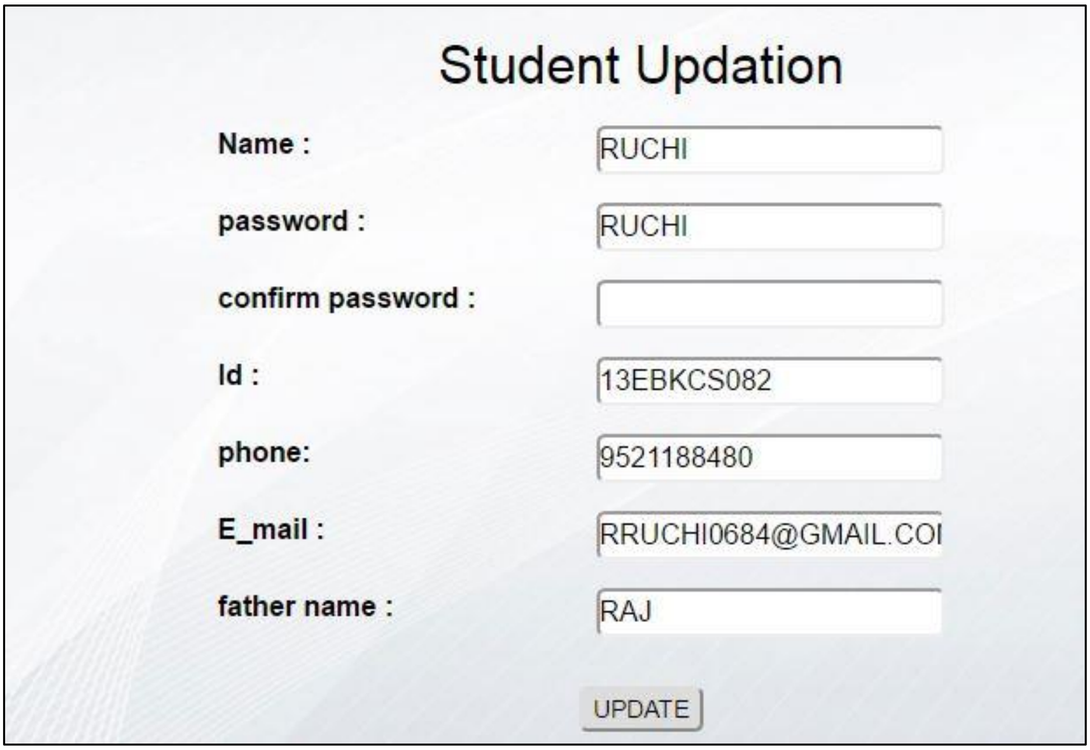
Figure 10.3 New Registration of student by the ADMIN



Student Record Updation

Enter Student Id: 13EBKCS082

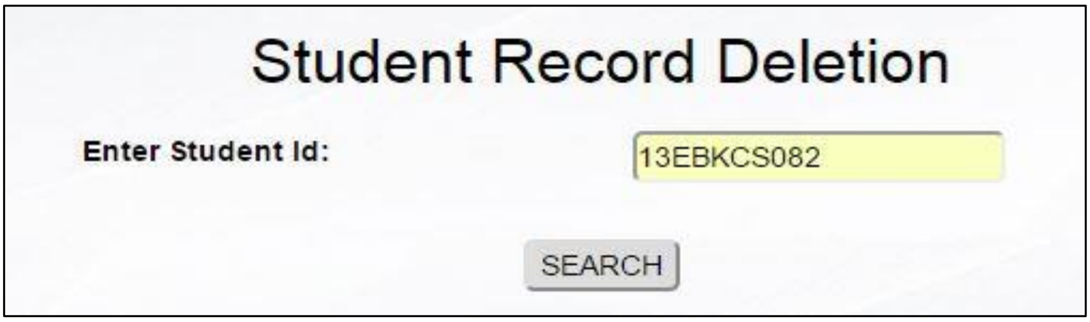
Figure 10.4 Student record is updated by the Admin



Student Updation

Name :	RUCHI
password :	RUCHI
confirm password :	
Id :	13EBKCS082
phone:	9521188480
E_mail :	RRUCHI0684@GMAIL.COI
father name :	RAJ

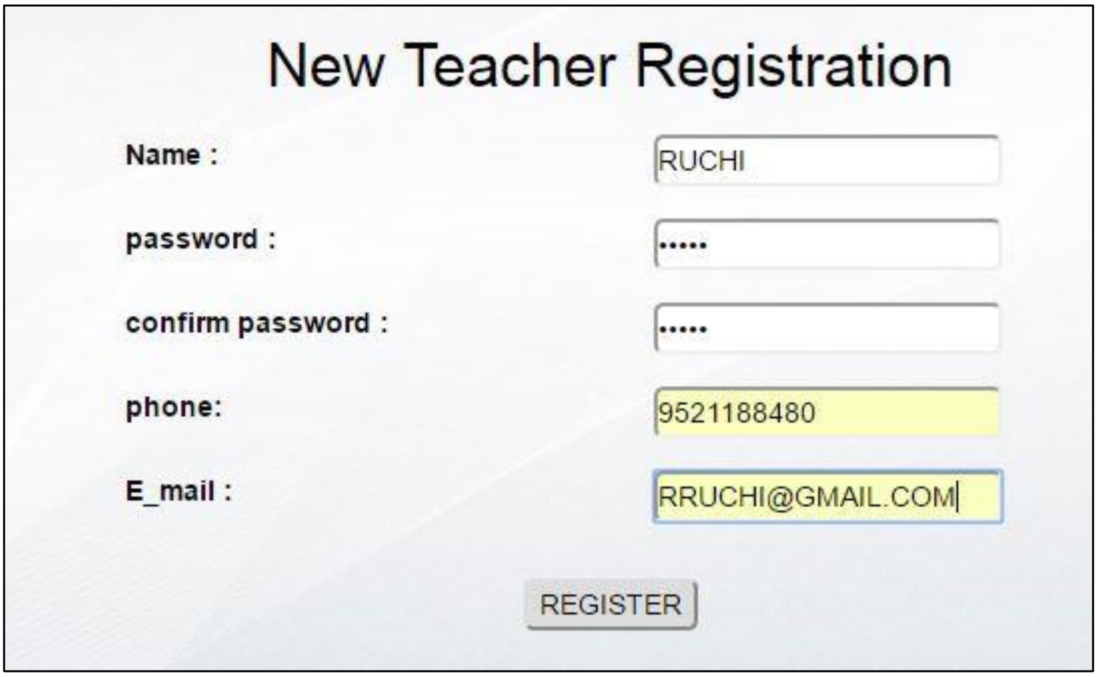
Figure 10.5 After enter the student id, we can update details here.



Student Record Deletion

Enter Student Id: 13EBKCS082

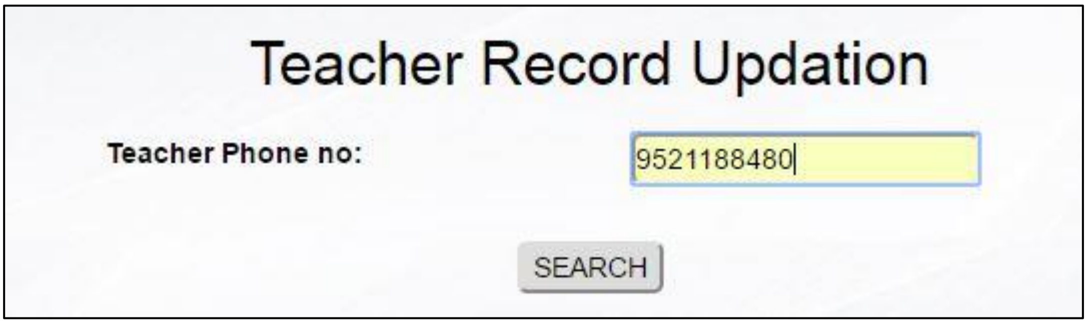
Figure 10.6 Enter student id for the deletion of record



New Teacher Registration

Name :	RUCHI
password :	•••••
confirm password :	•••••
phone:	9521188480
E_mail :	RRUCHI@GMAIL.COM

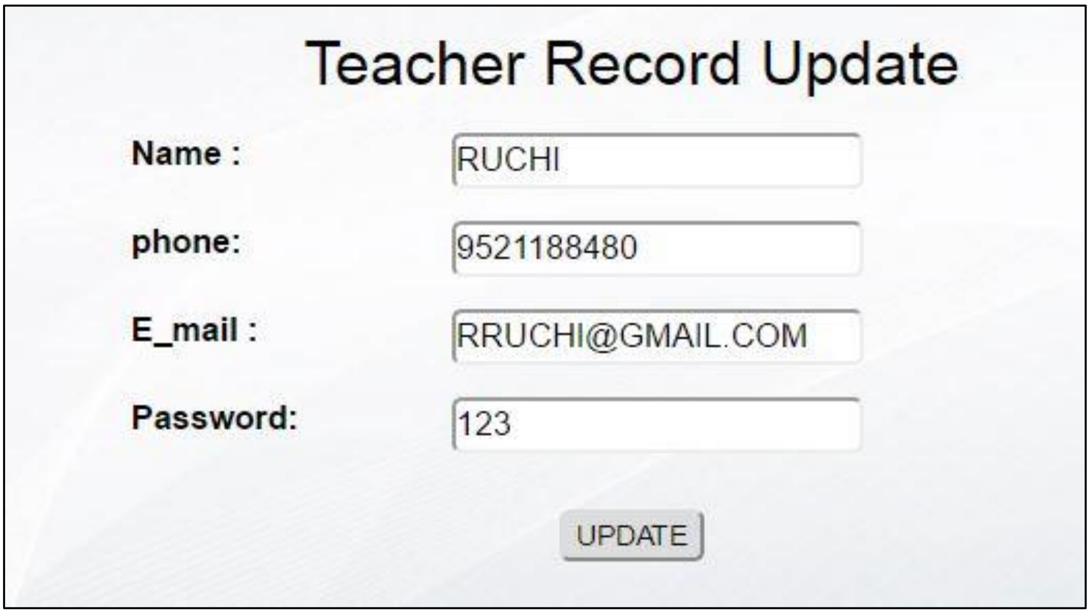
Figure 10.7 New teacher registration by the admin.



Teacher Record Updation

Teacher Phone no: 9521188480

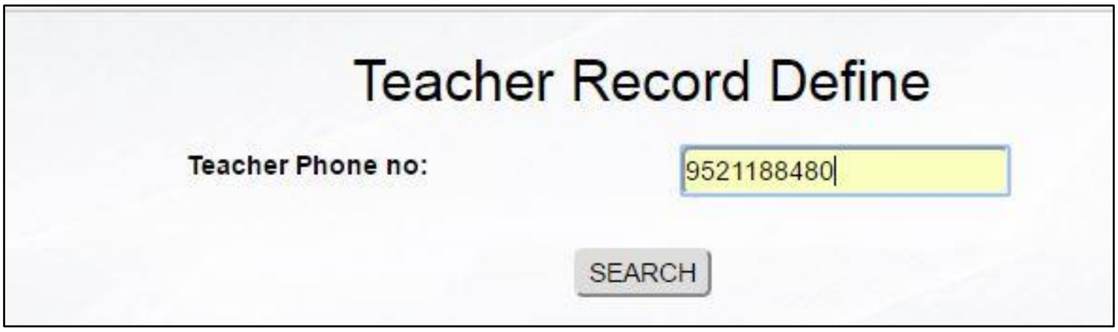
Figure 10.8 Enter teacher phone number for the updation of the record by admin



Teacher Record Update

Name :	RUCHI
phone:	9521188480
E_mail :	RRUCHI@GMAIL.COM
Password:	123

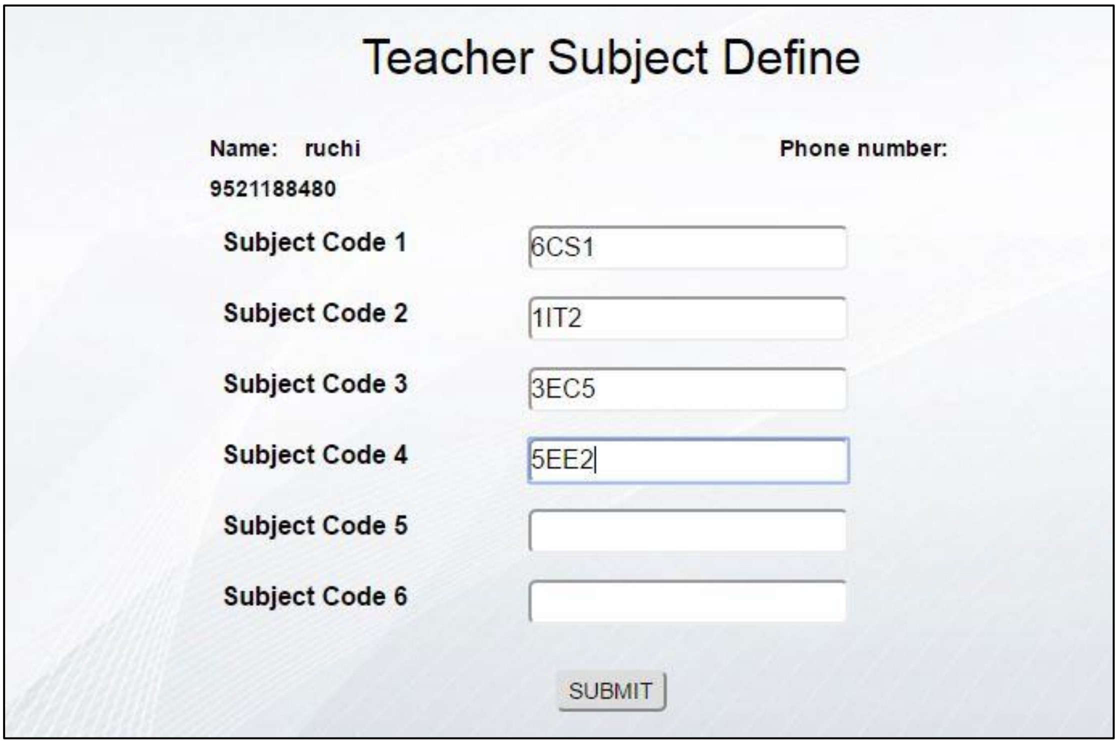
Figure 10.9 Teacher previous record is shown & update by the admin



Teacher Record Define

Teacher Phone no: 9521188480

Figure 10.10 Enter the teacher phone number ,whose subject is to be defibe



Teacher Subject Define

Name: ruchi Phone number: 9521188480

Subject Code 1: 6CS1

Subject Code 2: 1IT2

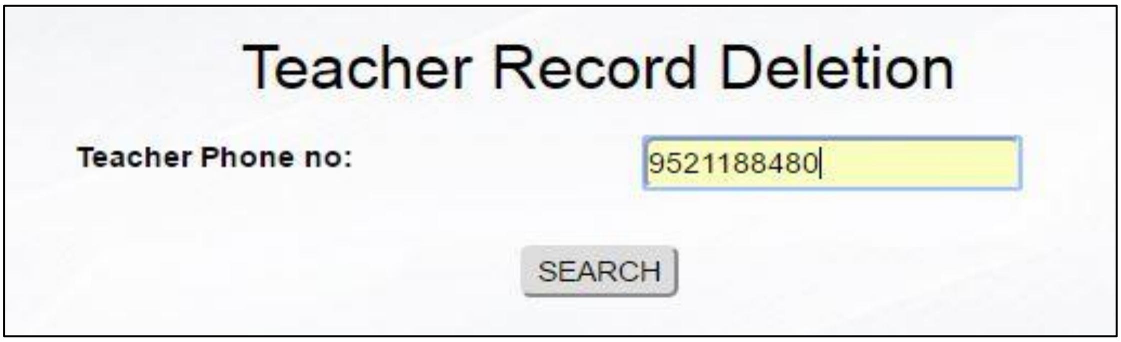
Subject Code 3: 3EC5

Subject Code 4: 5EE2

Subject Code 5:

Subject Code 6:

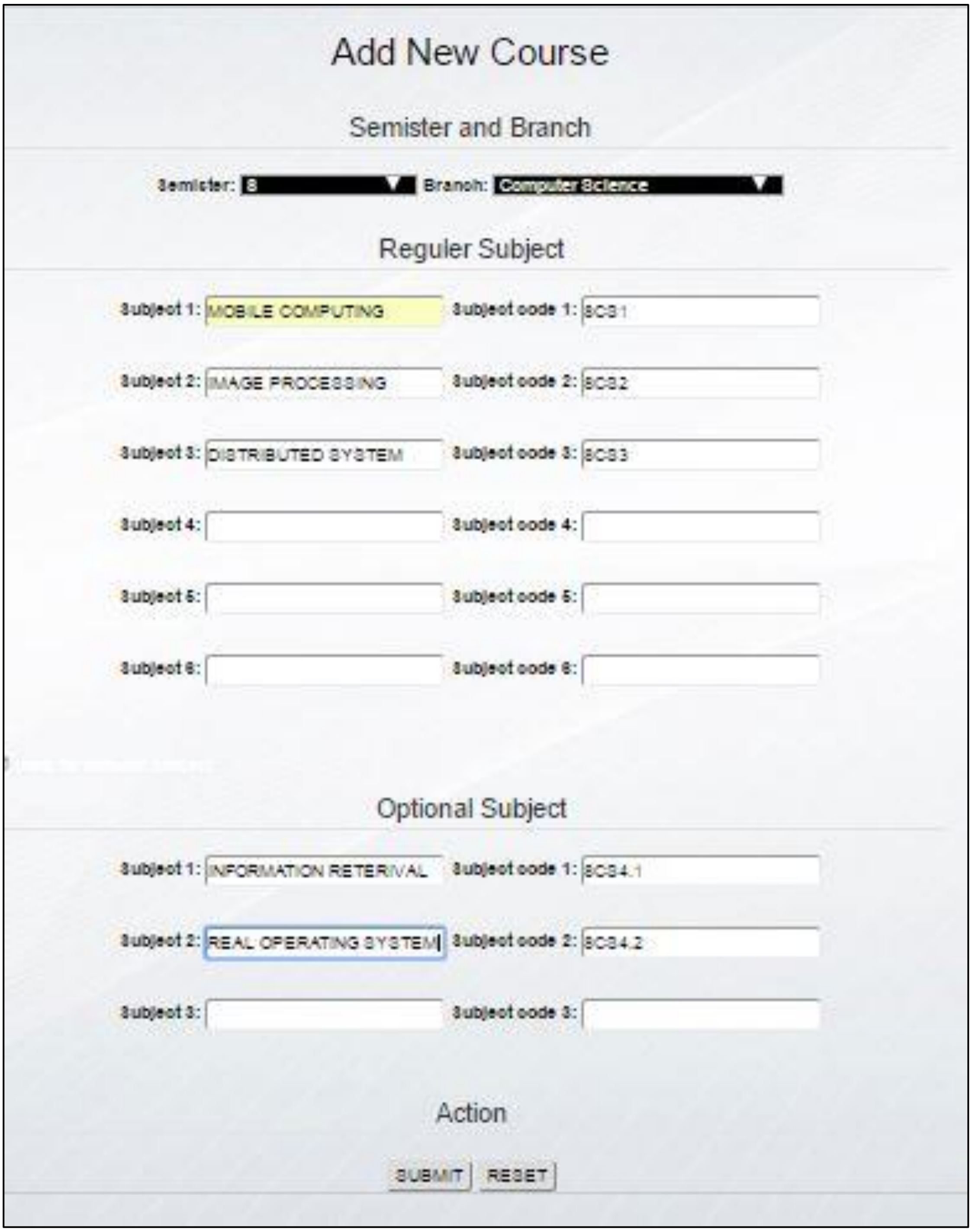
Figure 10.11 Enter the teacher subject which will be taught by the teacher



Teacher Record Deletion

Teacher Phone no: 9521188480

Figure 10.12 Enter the teacher number whose record is delete from the data base.



Add New Course

Semister and Branch

Semister: 8 Branch: Computer Science

Reguler Subject

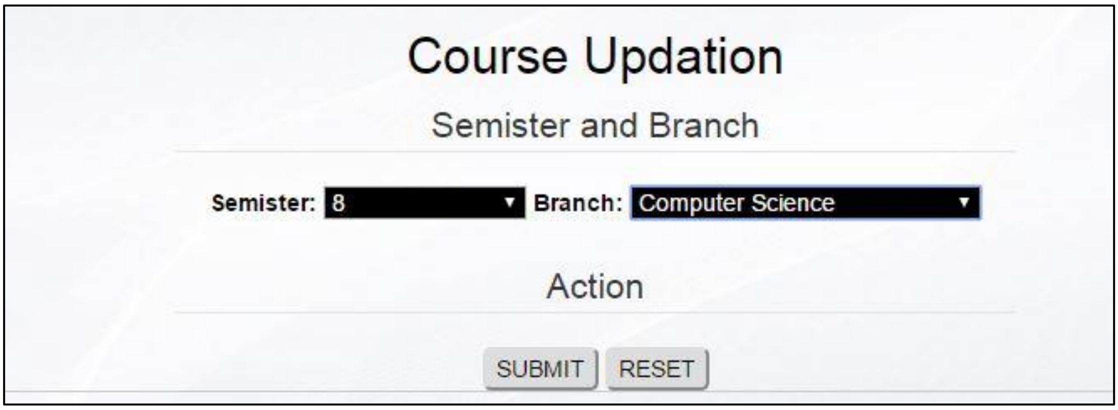
Subject 1:	MOBILE COMPUTING	Subject code 1:	8CS1
Subject 2:	IMAGE PROCESSING	Subject code 2:	8CS2
Subject 3:	DISTRIBUTED SYSTEM	Subject code 3:	8CS3
Subject 4:		Subject code 4:	
Subject 5:		Subject code 5:	
Subject 6:		Subject code 6:	

Optional Subject

Subject 1:	INFORMATION RETERIVAL	Subject code 1:	8CS4.1
Subject 2:	REAL OPERATING SYSTEM	Subject code 2:	8CS4.2
Subject 3:		Subject code 3:	

Action

Figure 10.13 Select the semester & Branch whose course want to enter



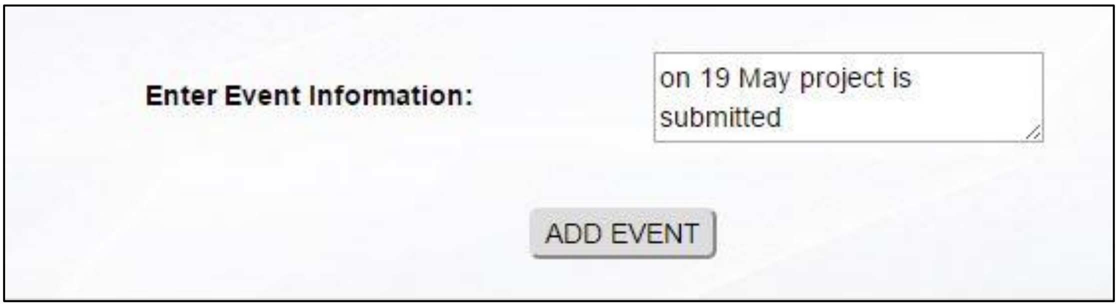
Course Updation

Semister and Branch

Semister: 8 Branch: Computer Science

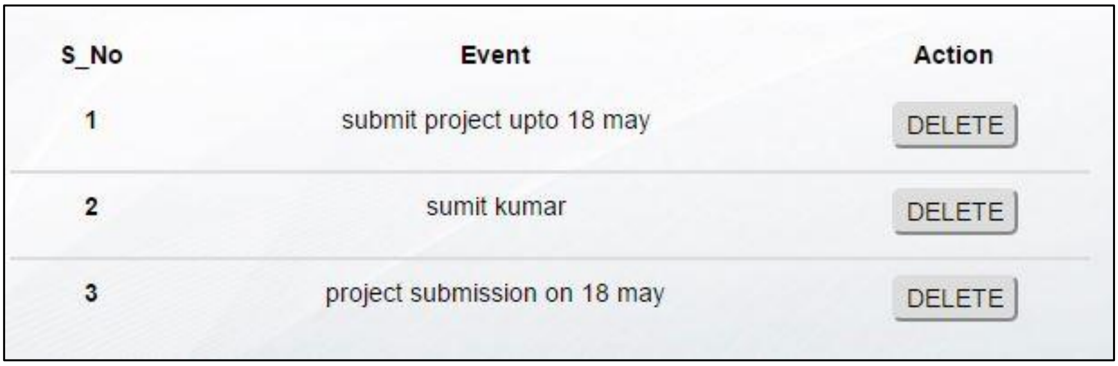
Action

Figure 10.14 Select the semester & Branch whose course want to update



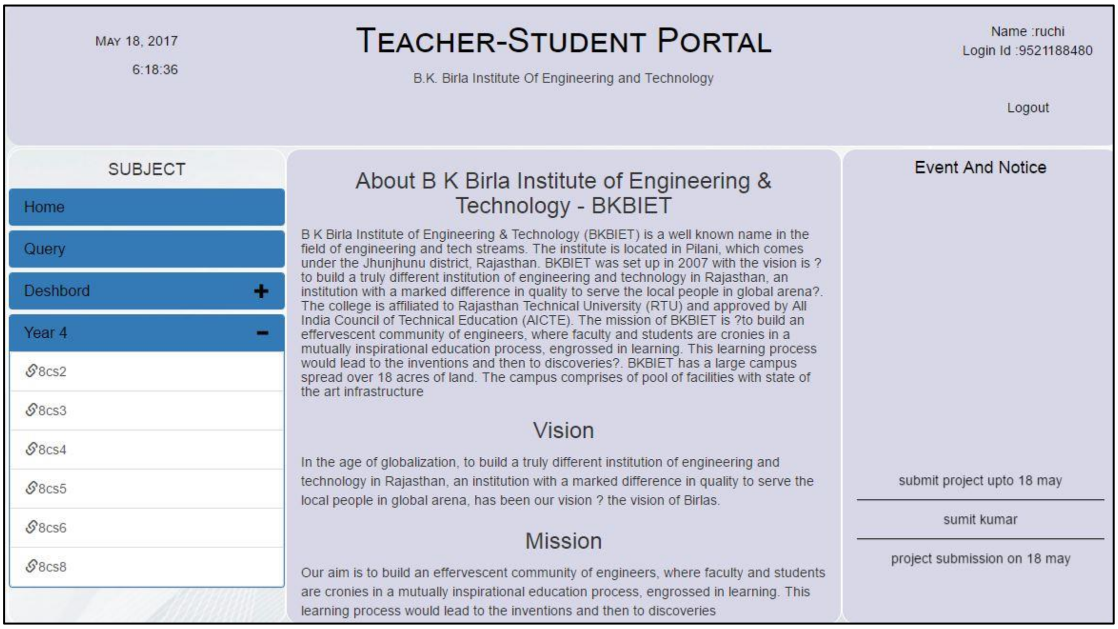
Enter Event Information: on 19 May project is submitted

Figure 10.15 Enter the event information that will be shown to the student & teacher.



S_No	Event	Action
1	submit project upto 18 may	DELETE
2	sumit kumar	DELETE
3	project submission on 18 may	DELETE

Figure 10.16 Show the event that is already present in the database.



MAY 18, 2017
6:18:36

TEACHER-STUDENT PORTAL

B.K. Birla Institute Of Engineering and Technology

Name :ruchi
Login Id :9521188480

About B K Birla Institute of Engineering & Technology - BKBIET

B K Birla Institute of Engineering & Technology (BKBIET) is a well known name in the field of engineering and tech streams. The institute is located in Pilani, which comes under the Jhunjhunu district, Rajasthan. BKBIET was set up in 2007 with the vision is ? to build a truly different institution of engineering and technology in Rajasthan, an institution with a marked difference in quality to serve the local people in global arena?. The college is affiliated to Rajasthan Technical University (RTU) and approved by All India Council of Technical Education (AICTE). The mission of BKBIET is ?to build an effervescent community of engineers, where faculty and students are cronies in a mutually inspirational education process, engrossed in learning. This learning process would lead to the inventions and then to discoveries?. BKBIET has a large campus spread over 18 acres of land. The campus comprises of pool of facilities with state of the art infrastructure

Vision

In the age of globalization, to build a truly different institution of engineering and technology in Rajasthan, an institution with a marked difference in quality to serve the local people in global arena, has been our vision ? the vision of Birlas.

Mission

Our aim is to build an effervescent community of engineers, where faculty and students are cronies in a mutually inspirational education process, engrossed in learning. This learning process would lead to the inventions and then to discoveries

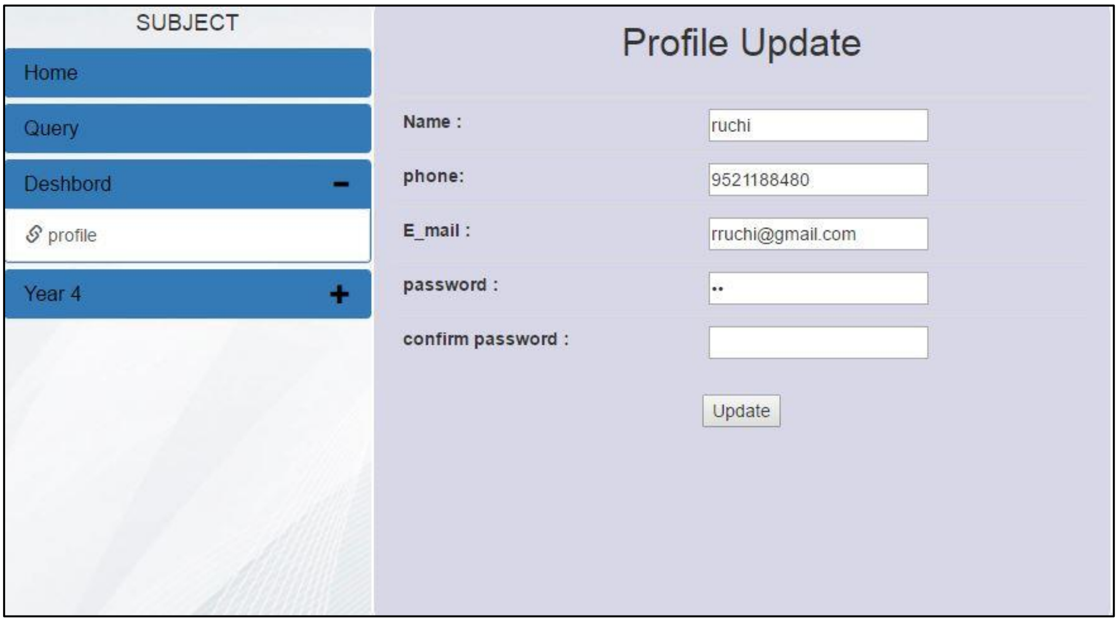
Event And Notice

submit project upto 18 may

sumit kumar

project submission on 18 may

Figure 10.17 When the teacher is login then the home page is this.



Profile Update

Name : ruchi

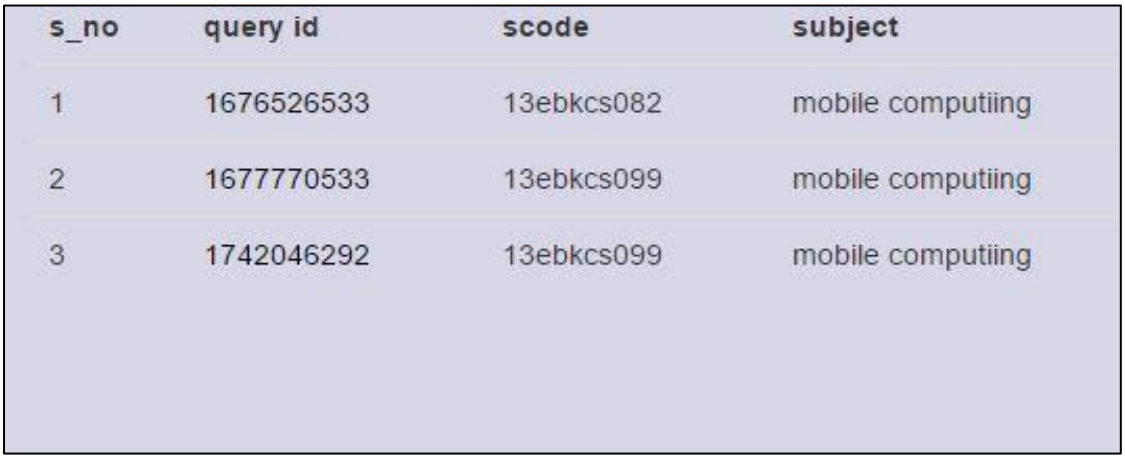
phone: 9521188480

E_mail : rruchi@gmail.com

password : ••

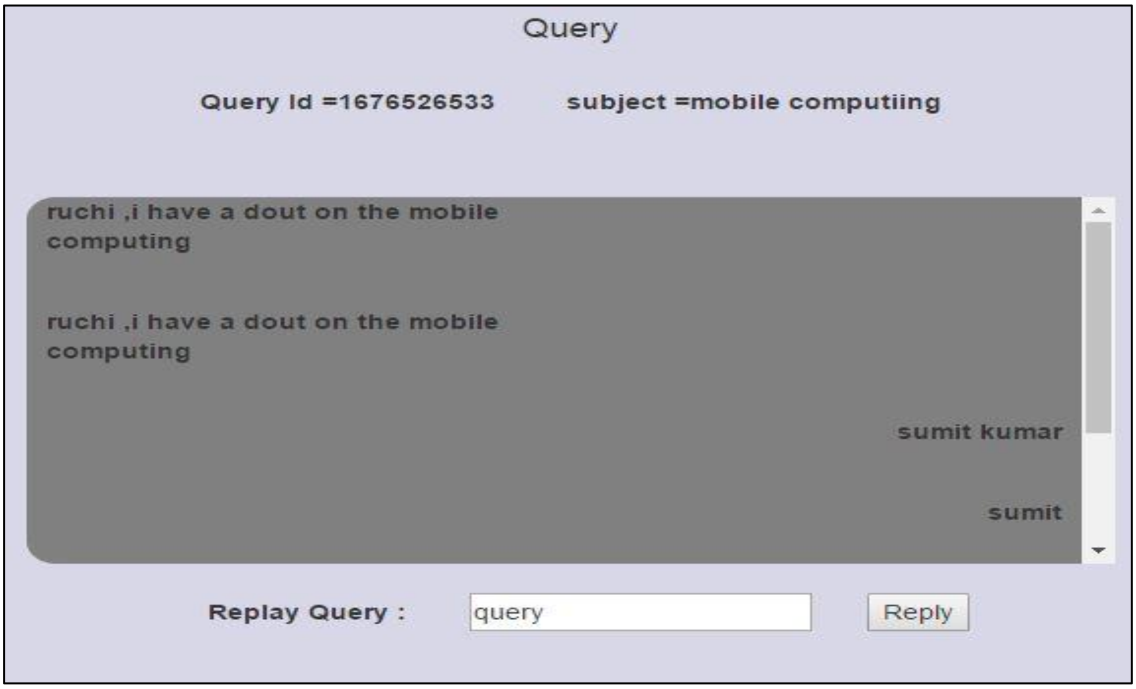
confirm password :

Figure 10.18 The teacher profile shown by the teacher ,only the password is updated here.



s_no	query id	scode	subject
1	1676526533	13ebkcs082	mobile computiing
2	1677770533	13ebkcs099	mobile computiing
3	1742046292	13ebkcs099	mobile computiing

Figure 10.19 Query that is ask by the Student



Query

Query Id =1676526533 subject =mobile computiing

ruchi ,i have a dout on the mobile computing

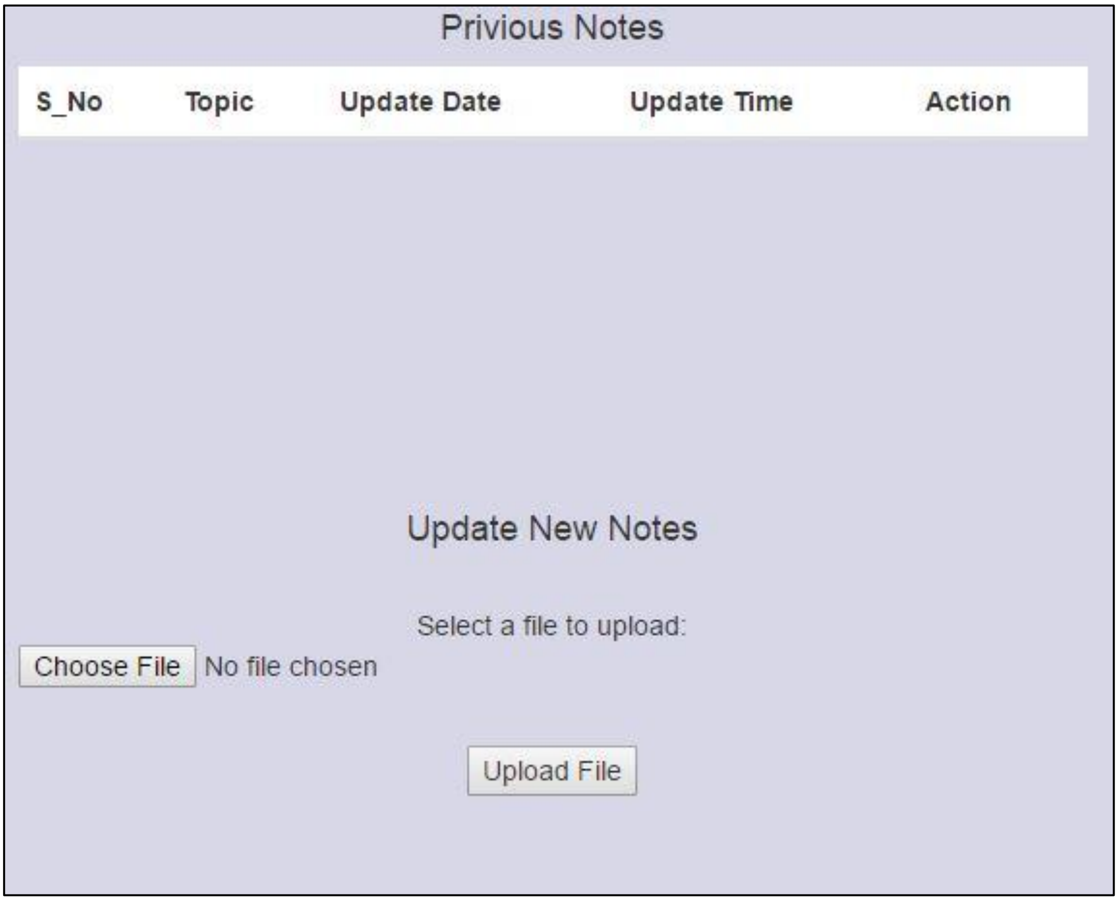
ruchi ,i have a dout on the mobile computing

sumit kumar

sumit

Replay Query : query

Figure 10.20 when click on the Query id then the whole query of student is shown



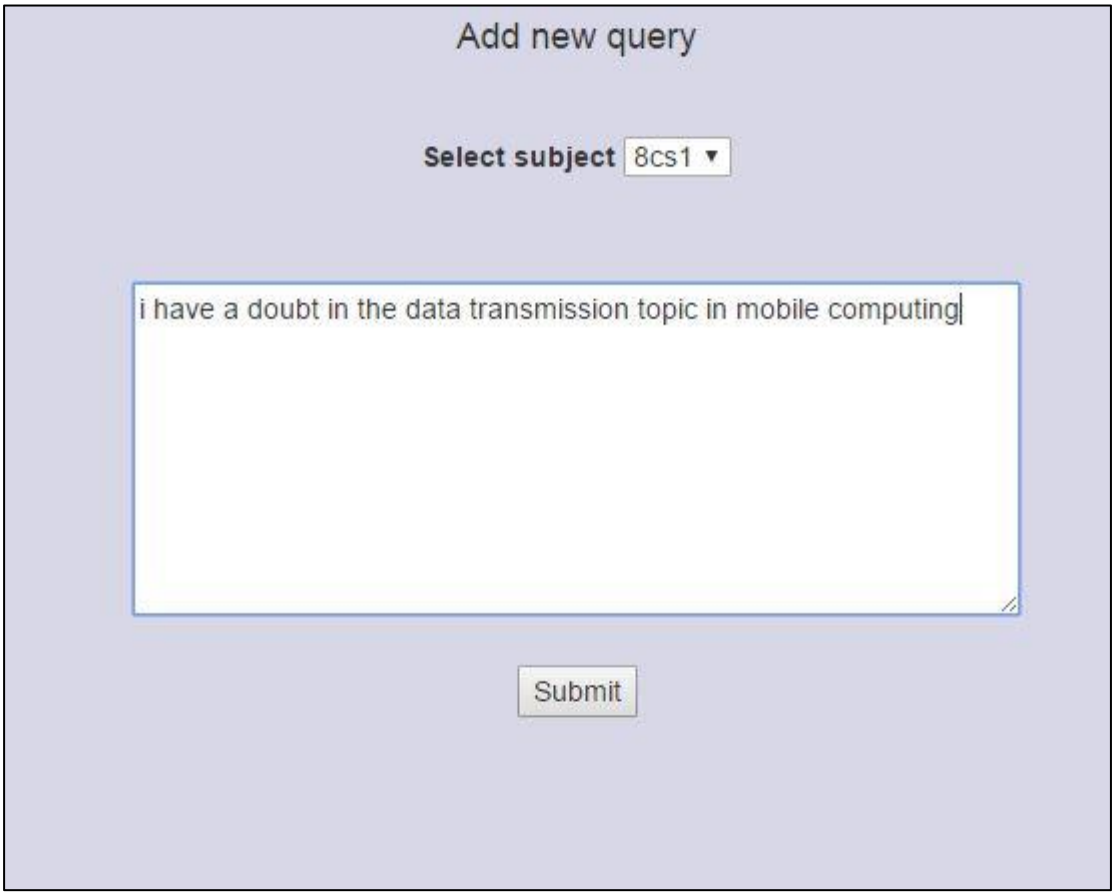
Privious Notes

S_No	Topic	Update Date	Update Time	Action

Update New Notes

Select a file to upload:

Figure 10.21 Select the file that want to upload by the teacher for the students

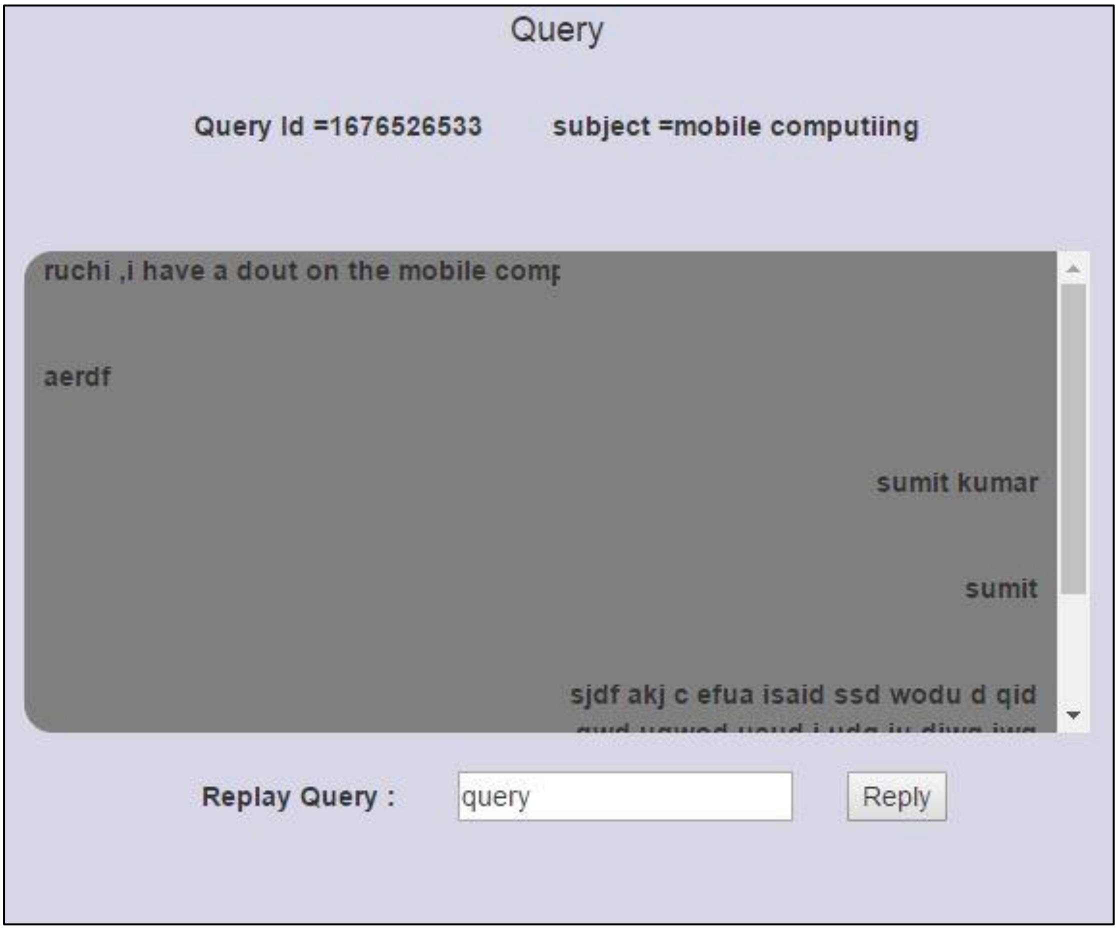


Add new query

Select subject 8cs1 ▾

i have a doubt in the data transmission topic in mobile computing

Figure 10.22 The Student slect the subject & then enter the query



Query

Query Id =1676526533 subject =mobile computiing

ruchi ,i have a dout on the mobile comp

aerdf

sumit kumar

sumit

sjdf akj c efua isaid ssd wodu d qid

Figure 10.23 Student see the reply that will be given by the teacher & again ask another doubt if any.

CONCLUSIONS

A portal storing student's information, registering them, updating their information, deleting student's record who don't need it further and also searching student by using student's College ID. The teacher information can be added, deleted, updated by the admin. Also the admin can add the event and delete the event. Another functionality is that the admin can update the course or add the new course to the website.

A chatting type application which allows users to do conversation with the respective teachers ask them queries/doubts and also the teachers will respond to the queries through this portal itself by replying them through messages. Efficient to use and also easily available.

LIMITATIONS OF THE PROJECT

This project is achieved the goal for which it is created .It has certain number limitation

1. The uploading of notes is work only on the pdf format.

2**.** The student of our college can only login into the website.

3. When the student over his/her college then it will not login into the website.

4. Student ask query only to the teacher which taught them the specific subject.

5. Only one student can ask the query to the single teacher.

REFERENCES

1. Java the Complete Reference 9th Edition by Herbert Schildt

2. https://www.tutorialspoint.com/java

3. https://www.javatpoint.com/java-tutorial

4. https://www.tutorialspoint.com/html

5. http://www.studytonight.com/code/html