Chapter 1

INTRODUCTION

The data is scattered across multiple transactional systems, so we have to extract it from those systems, transform it into a standardized format and finally load it into a central repository called a data warehouse. The data has to be reorganized so that it is presented to the users in an understandable way.

“The database Management System” is a system that manages databases and organizes the data, so that it can be easily retrieved by the users. This system can be used to manage transactional databases, such as HR systems, banking systems, hospital systems and so on. This project is typically optimized for performing transactions and it provides discrete pieces of information for the users.

The database management system organizes the files to give user more control over their data. The system makes it possible for users to create, edit and update data in database files. Once created, the system makes it possible to store and retrieve data from those database files. It provides functions such as concurrency, security, backup, integrity and data description. It also provides reliability. Database administrators are responsible for creating backups of databases, controlling access and, in general, making sure it works the way it was intended.

The system provides automated methods to create, store and retrieve data and also can make tedious manual tasks a thing of the past. A data base system reduces data redundancy and inconsistency. It allows for concurrent access by multiple users, each with their own specific role. Some users only need to view the data, some contribute to adding new data, while others design and manages the database- all at the same time.

Database management system is a software which is used to manage the database. For example: MySQL, Oracle, and many are a very popular commercial database which is used in different applications. DBMS provides an interface to perform various operations like database creation, storing data in it, updating data.

1.1 Database Management System Architecture

The design of a database management system depends on its architecture. It can be centralized or decentralized or hierarchical. The architecture of a database management system can be seen as either single tier or multi-tier. An n-tier architecture divides the whole system into related but independent n modules, which can be independently modified, altered, changed, or replaced.

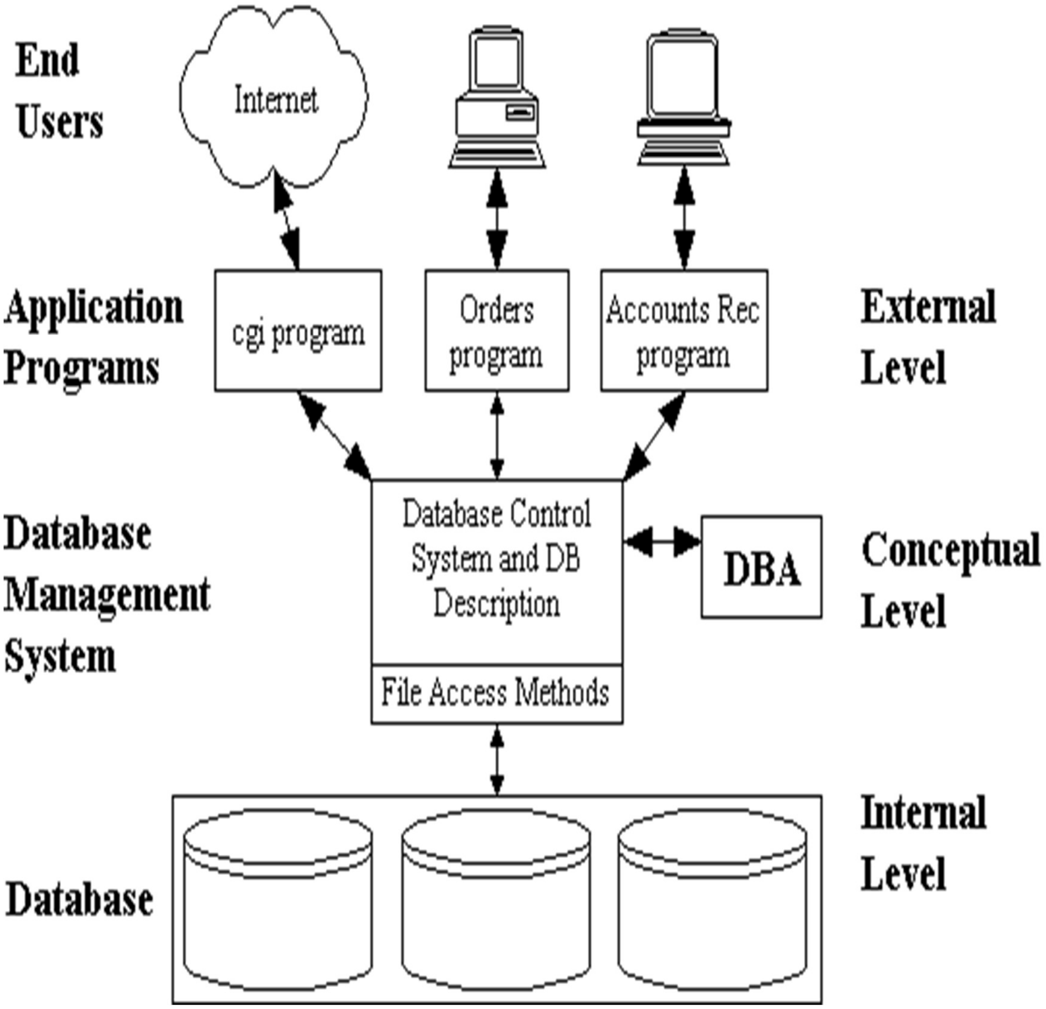


Figure 1.1: The Database Management System Architecture

1.2 Characteristics of Database Approach

A number of characteristics distinguish the database approach from the much older approach of programming with files. In traditional file processing, each user defines and implements the files needed for a specific software application as part of programming the application. For example, one user, the grade reporting office, may keep files on students and their grades. Programs to print a student's transcript and to enter new grades are implemented as part of the application A second user, the accounting office, may keep track of students' fees and their payments, although both users are interested in data about students, each user maintains separate files— and programs to manipulate these files—because each requires some data not avail able from the other user's files. This redundancy in defining and storing data results in wasted storage space and in redundant efforts to maintain common up-to-date data. In the database approach, a single repository maintains data that is defined once and then accessed by various users. In file systems, each application is free to name data elements independently. In contrast, in a database, the names or labels of data are defined once, and used repeatedly by queries, transactions, and applications. The main characteristics of the database approach versus the file-processing approach are the following:

- Self-describing nature of a database system.
- Insulation between programs and data, and data abstraction.
- Support of multiple views of the data.
- Sharing of data and multiuser transaction processing.

1.3 Applications of DBMS

Applications where we use Database Management Systems are:

- Telecom: There is a database to keeps track of the information regarding calls made, network usage, customer details etc. Without the database systems it is hard to maintain that huge amount of data that keeps updating every millisecond.
- Industry: Where it is a manufacturing unit, warehouse or distribution center, each one needs a database to keep the records of ins and outs. For example, distribution center should keep a track of the product units that supplied into the center as well as the products that got delivered out from the distribution center on each day; this is where DBMS comes into picture.

- Banking System: For storing customer info, tracking day to day credit and debit transactions, generating bank statements etc. All this work has been done with the help of Database management systems.
- Online shopping: You must be aware of the online shopping websites such as Amazon, Flipkart etc. These sites store the product information, your addresses and preferences, credit details and provide you the relevant list of products based on your query. All this involves a Database management system.
- Manufacturing: Manufacturing organizations make various kinds of items and deal them consistently. To keep the data about their items like bills, acquisition of the item, amount, inventory network the executives, information base administration framework (DBMS) is utilized.
- Social Media Sites: We all utilization of online media sites to associate with companions and to impart our perspectives to the world. Every day, many people group pursue these online media accounts like Pinterest, Facebook, Twitter, and Google in addition to. By the utilization of the data set administration framework, all the data of clients are put away in the information base and, we become ready to interface with others.
- Education Sector: Presently, assessments are led online by numerous schools and colleges. They deal with all assessment information through the data set administration framework (DBMS). In spite of that understudy's enlistments subtleties, grades, courses, expense, participation, results, and so forth all the data is put away in the information base.
- Human Resource Management: Big firms or organizations have numerous specialists or representatives working under them. They store data about worker's compensation, assessment, and work with the assistance of an information base administration framework (DBMS).

Chapter 2

SYSTEM REQUIREMENTS

System Requirement Specifications (SRS) is an important part of software development process. SRS includes overall description, functional and non-functional requirements, supportability, performance requirement, design constraints, etc. for any applications. These contents are very much useful in fulfilling the goals while implementing software projects.

2.1 Overall Description

This project is to help the organizations to organize the huge amount of data and manipulate it as required. The requirement Specification is a document that describes the external and internal requirements for any system. The Requirements analyst has to identify the requirements by talking to the clients and understanding their needs. The inputs are to be gathered from different resources to build the system based on the different requirements. The Requirements phase translates the ideas of the clients into a formal document. This software helps in tracking details in various sectors such as banks, library, universities, hospitals, Zoo, shops etc. The main goal of the software is to manage the data efficiently

2.2 Functional Requirements

Functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behaviour and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all the cases where the system uses the functional requirements are captured in use cases

Presently, assessments are led online by numerous schools and colleges. They deal with all assessment information through the data set administration framework (DBMS). In spite of that understudy’s enlistments subtleties, grades, courses, expense, participation, results, and so forth all the data is put away in the information base.

2.3 Non-Functional Requirements

- **Performance Requirements:**

  Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system.

- **Reliability:**

  In this system reliability means the mail which is send by the source must reach the target user with any modification and accurate.

- **Security:**

  The web server and database server should be protected from hacking, virus etc

- **Portability:**

  The application will be developed using standard open source software like PHP, Apache web server, MySQL database, Internet Explorer Browser etc these software will work both on Windows and Linux o/s. Hence portability problems will not arise.

- **Availability:**

  This software will be available always, in which a property of software that is there and ready to carry out this task when you need it to be. This is broad prospective and encompasses which is normally called relilability.

- **Maintainability:**

  In this system the presentation layer is clearly separated from the service layer. So any modification in future will be done with less effort. The database will be running at the server. Users access these forms by using the user-ids and the passwords.

2.4 Specific Requirements

Many requirements represent stakeholder-defined characteristics the absence of which will result in a major or even fatal deficiency. Others represent features that may be implemented if time and budget permits. The requirements must specify a level of importance. The specification may include a set of use cases that describe interactions the users will have with the software. It should also include the specifications of hardware and software that are used for building the software.

2.4.1 Software Requirements

As previously mentioned, there are many software that will be part of this project and all of them are required for development purpose.

MySQL is an open-source Relational Database Management System (RDBMS) that is under the GNU GPL and is one of the most widely used. It also allows to scale the project without much overhead. It also has many features such as high availability, query caching, cross platform support and security make it a good candidate for deployment. We will be using MySQL version 5.7.14 hich is the latest version.

o	Database	:	MySQL
O	Server	:	Apache
o	Frontend	:	HTML
o	Scripting language	:	Java Script
o	IDE	:	Sublime
o	Technology	:	PHP

2.4.2 Hardware Requirements

Most of the current computer have enough specification to implement a database. But we need to mention some minimal requirement. These would be the minimum specifications to run the DBMS project. Use cases that describe interactions the users will have with the software.

- **Processor Requirement**

  A basic fast processor is essential for efficient handling of load during server time and development time. A processor with minimum 2 cores can handle the workload. Minimum requirement processor is a Pentium 4(P4). The prerequisites are known as (computer) system requirement and are often used as a guideline as opposed to an absolute rule. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements

- **Memory Requirements**

  The RAM memory will be needed to efficiently run the server and the front end, hence at least 2GB RAM would be necessary. And the hard disk must be at least fourty to eighty GB.

○ Processor	:	Intel P-IV based system
○ Processor Speed	:	2.0. GHz
○ RAM	:	2GB
○ Hard Disk	:	40GB to 80GB

Chapter 3

DATABASE DESIGN

3.1 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.

ER diagram should have mainly 3 components namely, entity, attribute, relationship. The following notations can be used for drawing an ER diagram.

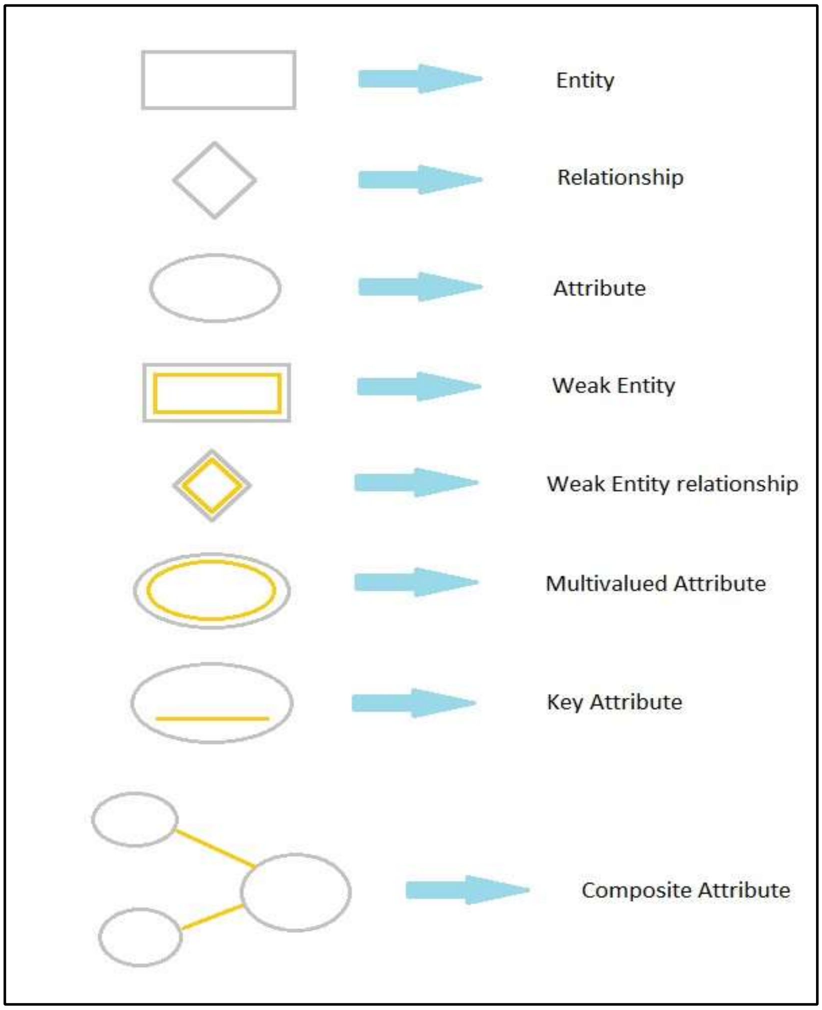


Figure 3.1: E-R Diagram Notations.

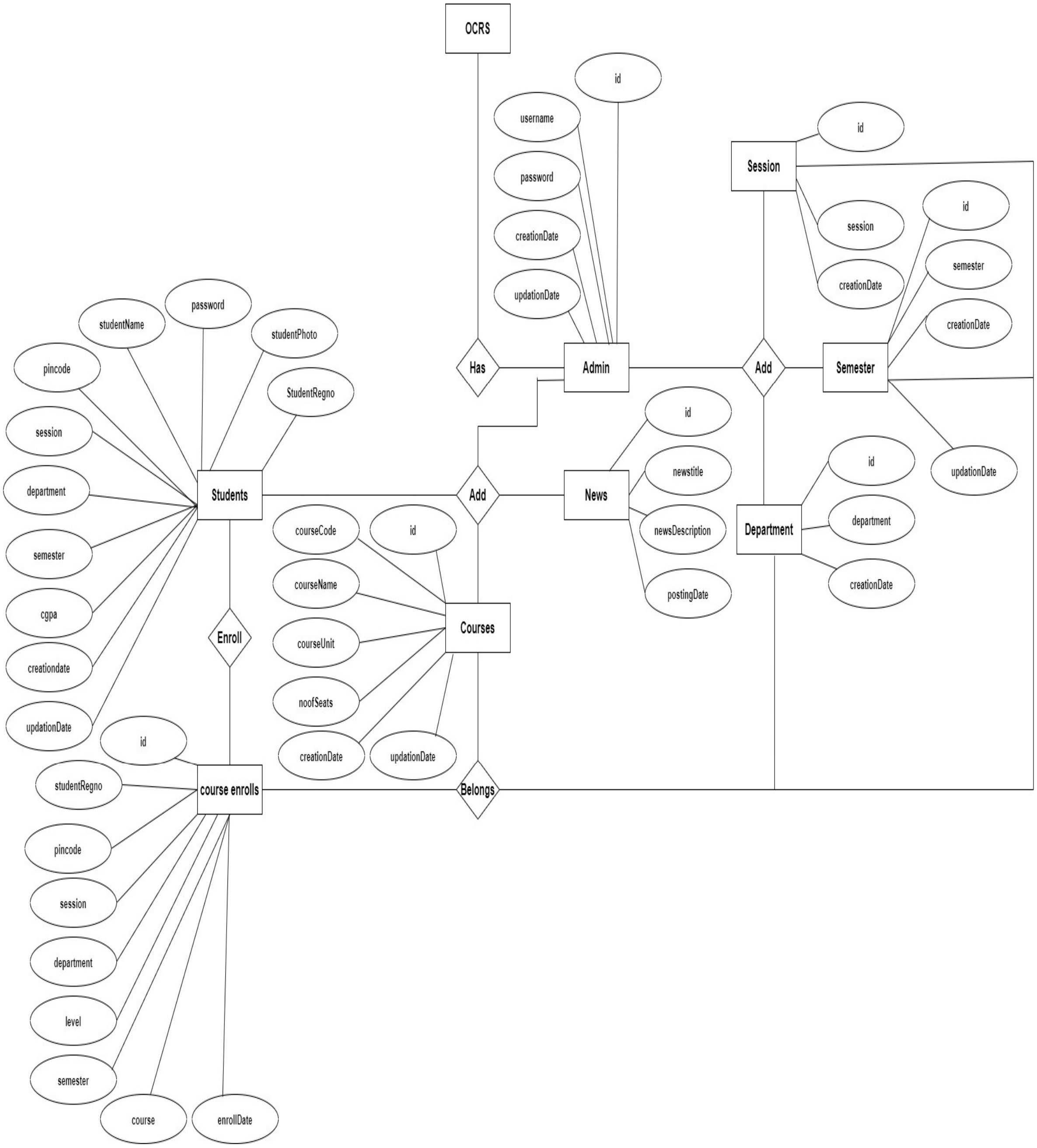


Figure 3.2: Sample E-R diagram.

In figure 3.2 we have seven entities Admin, Booking, Contact, Event Type, Page, Service and User Data. In this Online Banquet Booking System, an admin can add services and event types, and also manage the banquet booking and enquires from the user.

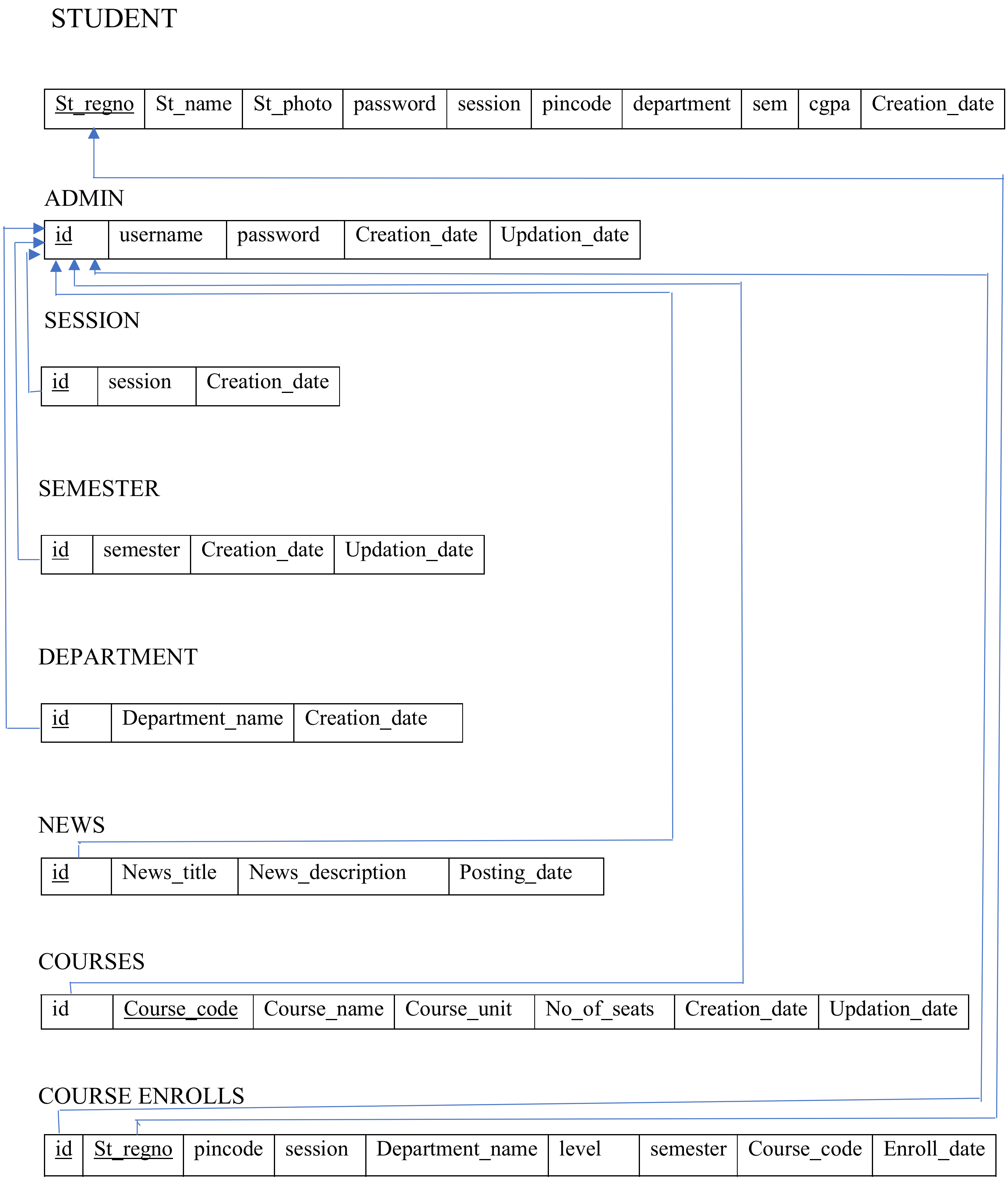
The user after logging in, can book banquets based on his requirements for the event he is organizing and also enquire anything from the admin.

3.2 Relational Schema

The data schema or relational schema considered as the blue print of a database which describes how the data may relate to other tables or other data models. However, the schema does not actually contain data, but it is the structure of a database described in a formal language supported by the Database Management System.

The relational schema diagram gives the relation of one entity with another as well as the information about the key constraints. The below figure is a sample relational schema diagram in which the attributes that are underlined are the primary key and the arrow line is used to represent the mapping.

The relational model of data permits the database designer to create a consistent, logical representation of information. Consistency is achieved by including declared constraints in the database design, which is usually referred to as the logical schema



STUDENT

St_regno	St_name	St_photo	password	session	pincode	department	sem	cgpa	Creation_date

ADMIN

id	username	password	Creation_date	Updation_date

SESSION

id	session	Creation_date

SEMESTER

id	semester	Creation_date	Updation_date

DEPARTMENT

id	Department_name	Creation_date

NEWS

id	News_title	News_description	Posting_date

COURSES

id	Course_code	Course_name	Course_unit	No_of_seats	Creation_date	Updation_date

COURSE ENROLLS

id	St_regno	pincode	session	Department_name	level	semester	Course_code	Enroll_date

Figure 3.3: Sample Relational Schema.

Chapter 4

IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

4.1 Technologies Used

4.1.1 PHP

- PHP stands for PHP: Hypertext Pre-processor
- PHP is a server-side scripting language, like ASP
- PHP scripts are executed on the server
- PHP supports many databases (MYSQL, Informix, Oracle, Sybase, Solid, Generic ODBC, etc.)
- PHP is an open source software
- PHP is free to download and use

4.1.2 MYSQL

- MYSQL is a database server
- MYSQL is ideal for both small and large applications
- MYSQL supports standard SQL
- MYSQL compiles on a number of platforms
- MYSQL is free to download and use

4.1.3 CSS

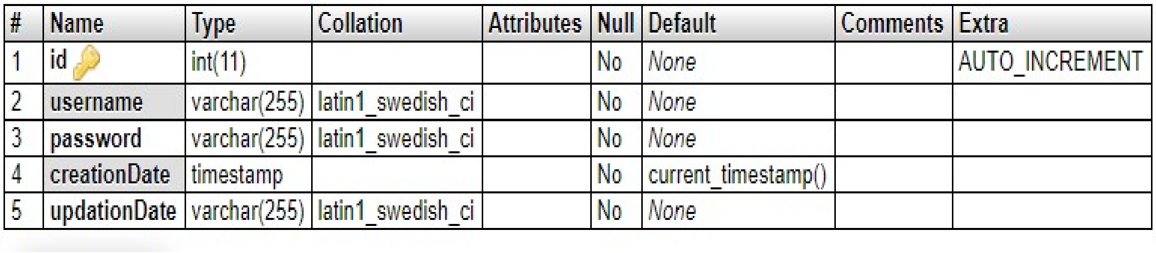
- Cascading Style Sheets (CSS)
- Simple mechanism
- Easy for adding style (e.g., fonts, colors, spacing) to Web documents.

4.2 RDBMS Tables and their Description

Online course registration sysytem contains 10 MySQL tables:

- **Admin:**

This table stores the admin’s login details.

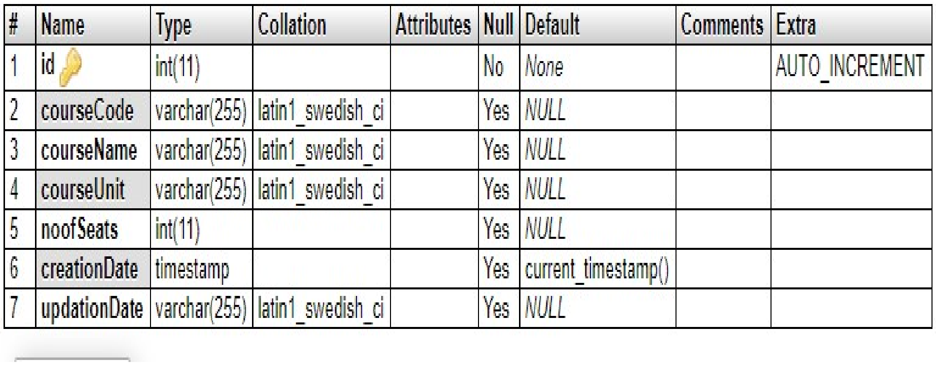


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	username	varchar(255)	latin1_swedish_ci		No	*None*		
3	password	varchar(255)	latin1_swedish_ci		No	*None*		
4	creationDate	timestamp			No	current_timestamp()		
5	updationDate	varchar(255)	latin1_swedish_ci		No	*None*		

Table 1 : Admin Details

- **Course :**

This store the course details

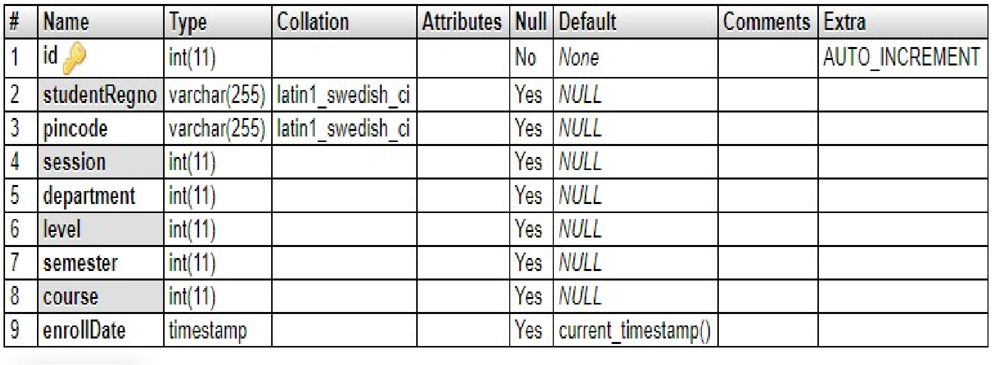


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	courseCode	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	courseName	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
4	courseUnit	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
5	noofSeats	int(11)			Yes	*NULL*		
6	creationDate	timestamp			Yes	current_timestamp()		
7	updationDate	varchar(255)	latin1_swedish_ci		Yes	*NULL*		

Table 2 : Course Details

- **Enroll:**

This table stores the course enroll details

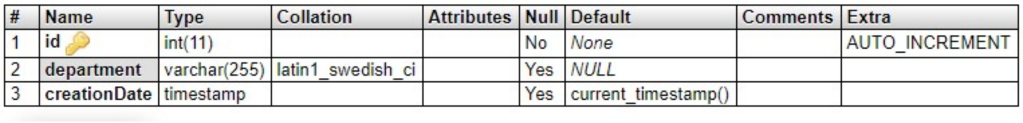


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	studentRegno	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	pincode	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
4	session	int(11)			Yes	*NULL*		
5	department	int(11)			Yes	*NULL*		
6	level	int(11)			Yes	*NULL*		
7	semester	int(11)			Yes	*NULL*		
8	course	int(11)			Yes	*NULL*		
9	enrollDate	timestamp			Yes	current_timestamp()		

Table 3 : Enroll Details

- **Department:**

This table stores the department details.

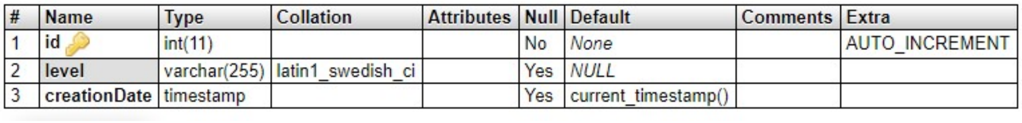


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	department	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	creationDate	timestamp			Yes	current_timestamp()		

Table 4 : Department Details

- **Level:**

This table stores detail of course level.

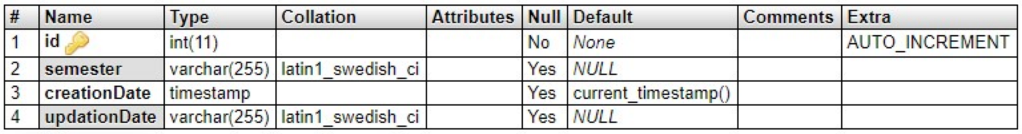


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	level	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	creationDate	timestamp			Yes	current_timestamp()		

Table 5 :Course level Details

- **Semester:**

This table stores the details of semester.

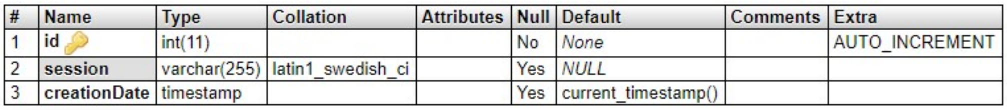


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id** 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	**semester**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	**creationDate**	timestamp			Yes	current_timestamp()		
4	**updationDate**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		

Table 6 : Semester Details

- **Session:**

This table stores the details of session.

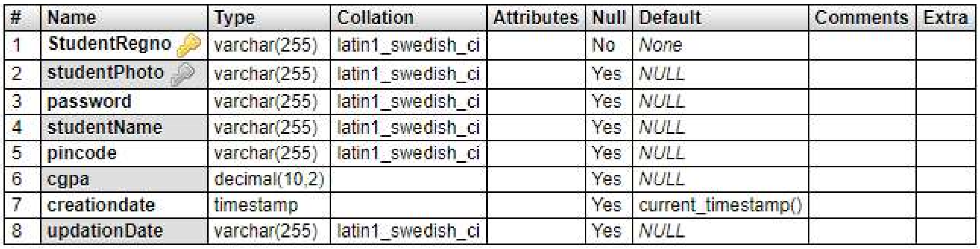


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id** 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	**session**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	**creationDate**	timestamp			Yes	current_timestamp()		

Table 7: Session Details

- **Student:**

This table stores details of student.

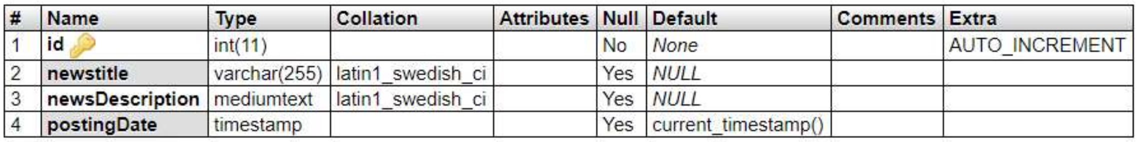


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**StudentRegno** 🔑	varchar(255)	latin1_swedish_ci		No	*None*		
2	**studentPhoto** 🔑	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	**password**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
4	**studentName**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
5	**pincode**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
6	**cgpa**	decimal(10,2)			Yes	*NULL*		
7	**creationdate**	timestamp			Yes	current_timestamp()		
8	**updationDate**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		

Table 8: Student Details

- **News:**

This stores latest news or updates

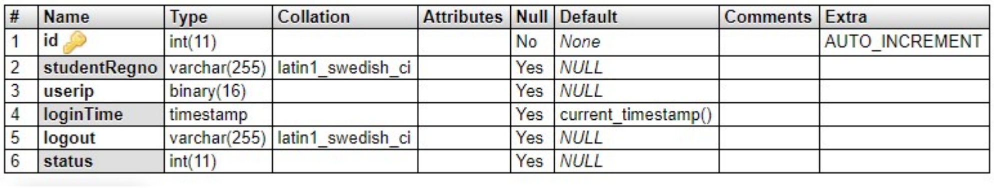


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id** 🔑	int(11)			No	*None*		AUTO_INCREMENT
2	**newstitle**	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	**newsDescription**	mediumtext	latin1_swedish_ci		Yes	*NULL*		
4	**postingDate**	timestamp			Yes	current_timestamp()		

Table 9: News Table

- **User logs:**

This stores details of user login time.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	*None*		AUTO_INCREMENT
2	studentRegno	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
3	userip	binary(16)			Yes	*NULL*		
4	loginTime	timestamp			Yes	current_timestamp()		
5	logout	varchar(255)	latin1_swedish_ci		Yes	*NULL*		
6	status	int(11)			Yes	*NULL*		

Table 10:User Logs

4.3 PHP code to connect MySQL database

```
<?php
// DB credentials.
define('DB_HOST','localhost');
define('DB_USER','root'); define('DB_PASS',");
define('DB_NAME','obbs');

// Establish database connection.
try
{
$dbh     =        new     PDO("mysql:host=".DB_HOST.";dbname=".DB_NAME,DB_USER,
DB_PASS,array(PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES 'utf8'"));
}   catch   (PDOException
$e)
{      exit("Error:      "      .      $e-
>getMessage());
}
?>
```

4.4 PHP code to create index page:

```
<?php
session_start();
error_reporting(0);
include("includes/config.php");
if(isset($_POST['submit']))
{
   $regno=$_POST['regno'];
   $password=md5($_POST['password']);
$query=mysqli_query($con,"SELECT*
FROM students
 WHERE StudentRegno='$regno' and password='$password'");
$num=mysqli_fetch_array($query);
if($num>0)
{
$_SESSION['login']=$_POST['regno'];
$_SESSION['id']=$num['studentRegno'];
$_SESSION['sname']=$num['studentName'];
$uip=$_SERVER['REMOTE_ADDR'];
$status=1;
$log=mysqli_query($con,"insertinto
userlog(studentRegno,userip,status) values('".$_SESSION['login']."','$uip','$status')");
header("location:http:change-password.php");
}else{
$_SESSION['errmsg']="Invalid Reg no or Password";
header("location:http:index.php");
}
}
?>
```

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />
  <meta name="description" content="" />
  <meta name="author" content="" />

  <title>Student Login</title>
  <link href="assets/css/bootstrap.css" rel="stylesheet" />
  <link href="assets/css/font-awesome.css" rel="stylesheet" />
  <link href="assets/css/style.css" rel="stylesheet" />
</head>
<body>
  <?php include('includes/header.php');?>

<section class="menu-section">
      <div class="container">
          <div class="row">
              <div class="col-md-12">
                  <div class="navbar-collapse collapse ">
                      <ul id="menu-top" class="nav navbar-nav navbar-right">
                          <li><a href="index.php">Home </a></li>
                          <li><a href="admin/">Admin Login </a></li>
                           <li><a href="index.php">Student Login</a></li>



                      </ul>
                  </div>
              </div>
```

```
    </div>
  </div>
</section>



<div class="content-wrapper">
  <div class="container">
    <div class="row">
      <div class="col-md-12">
        <h4 class="page-head-line">Please Login To Enter </h4>
      </div>
    </div>
     <span style="color:red;" ><?php echo htmlentities($_SESSION['errmsg']);
  ?><?php echo htmlentities($_SESSION['errmsg']="");?></span>
    <form name="admin" method="post">
    <div class="row">
      <div class="col-md-6">
         <label>Enter Reg no : </label>
          <input type="text" name="regno" class="form-control"  />
          <label>Enter Password :  </label>
          <input type="password" name="password" class="form-control"  />
          <hr />
          <button type="submit" name="submit" class="btn btn-info">
         <span class="glyphicon glyphicon-user"></span> &nbsp;Log Me In
         </button>&nbsp;
      </div>
      </form>
      <div class="col-md-6">
        <div class="alert alert-info">
       <strong> Latest News / Updates</strong>
           <marquee direction='up'scrollamount="2"
```

```
                 onmouseover="this.stop();" onmouseout="this.start();">
               <ul>
                  <?php
$sql=mysqli_query($con,"select * from news");
$cnt=1;
while($row=mysqli_fetch_array($sql))
{
?>
            <li>
          <a href="news-details.php?nid=<?php echo htmlentities($row['id']);?>">
<?php echo htmlentities($row['newstitle']);?>-<?php echo htmlentities($row['postingDate']);
</a>
                  </li>
                 <?php } ?>
               </ul>
             </marquee>
              </div>
                       </div>
       </div>
     </div>
   </div>
   <!-- CONTENT-WRAPPER SECTION END-->
   <?php include('includes/footer.php');?>
   <!-- FOOTER SECTION END-->
   <!-- JAVASCRIPT AT THE BOTTOM TO REDUCE THE LOADING TIME  -->
   <!-- CORE JQUERY SCRIPTS -->
   <script src="assets/js/jquery-1.11.1.js"></script>
   <!-- BOOTSTRAP SCRIPTS  -->
   <script src="assets/js/bootstrap.js"></script>
</body>
</html>
```

Chapter 5

RESULTS

5.1 Admin Login Page:

Here the Admin can login by entering username and password

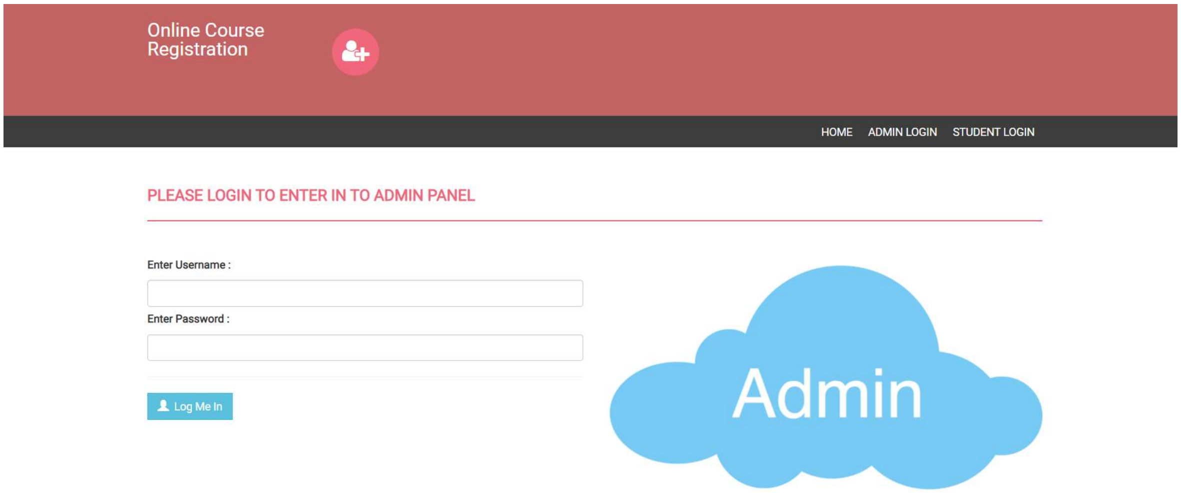


Figure 5.1: Snapshot of Admin login page

5.2 Change Password:

Here the admin can change password

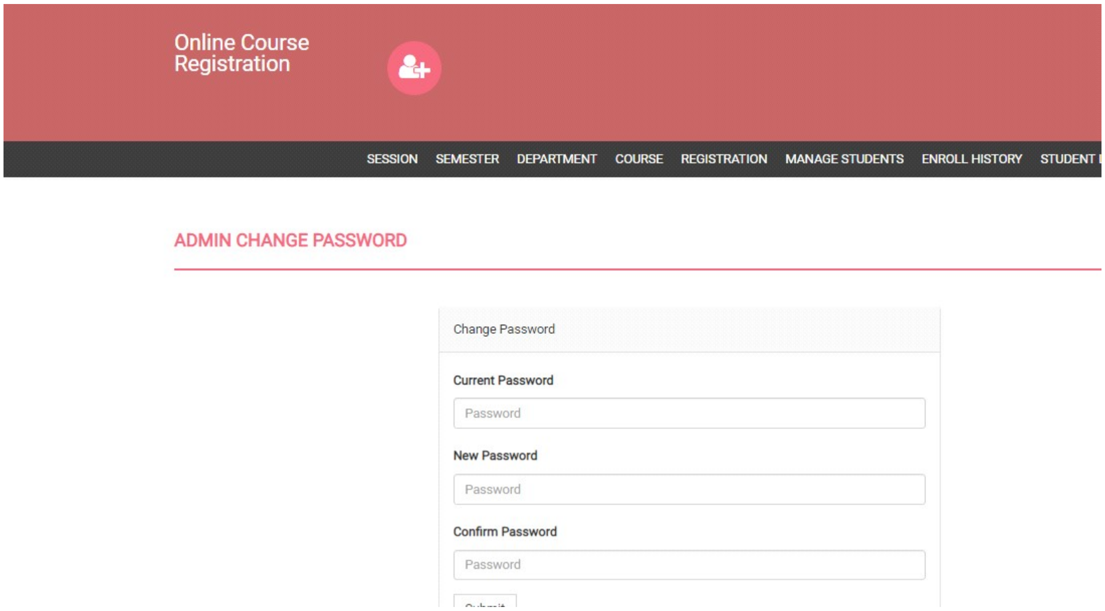
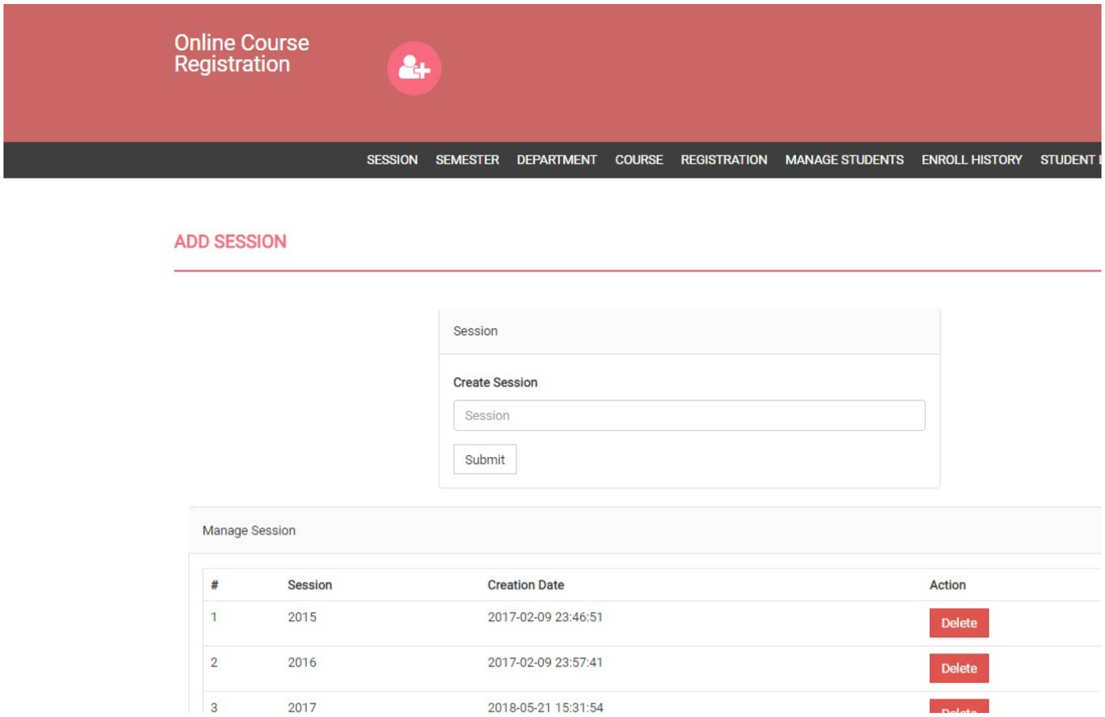


Figure 5.2: Snapshot of change password

5.3 Add/Manage Session

Here admin can add or manage sessions

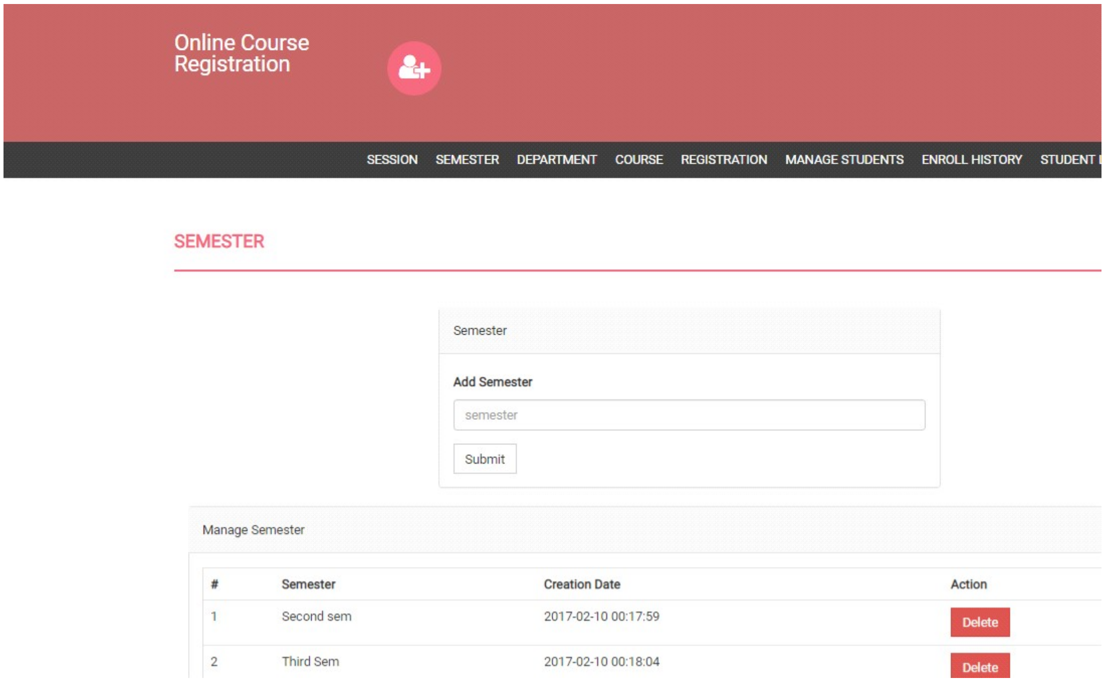


#	Session	Creation Date	Action
1	2015	2017-02-09 23:46:51	Delete
2	2016	2017-02-09 23:57:41	Delete
3	2017	2018-05-21 15:31:54	Delete

Figure 5.3: Snapshot of Add session

5.4 Add/Manage Semester:

Here admin can add or mange semester.

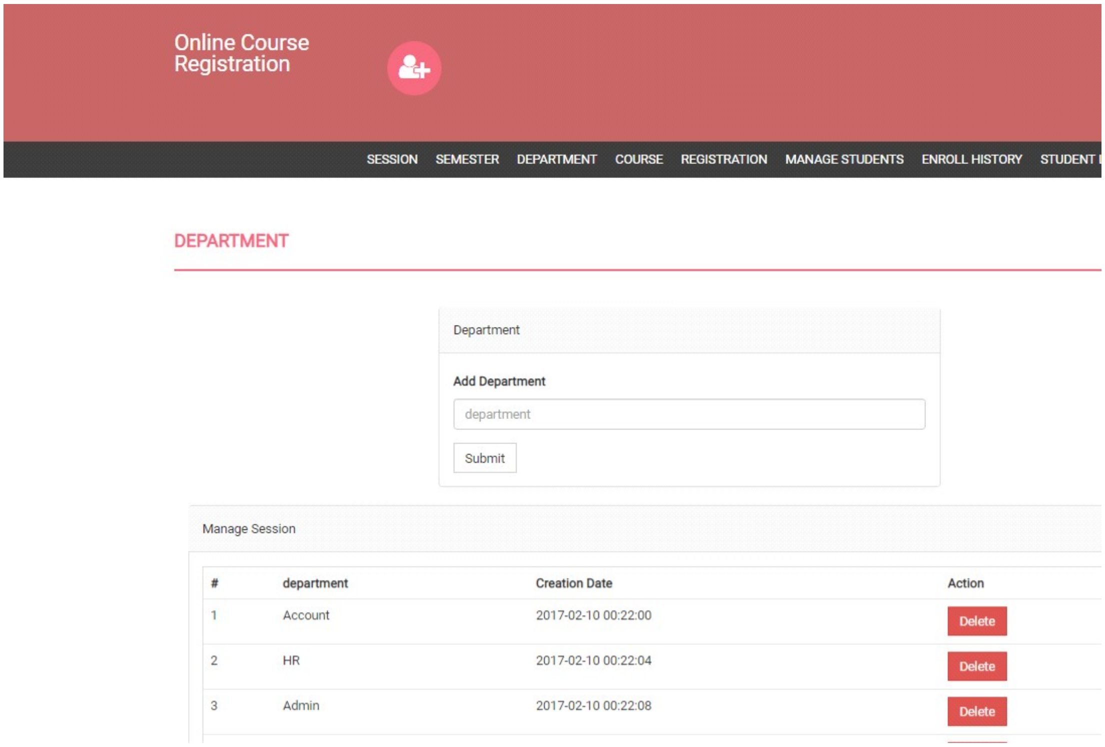


#	Semester	Creation Date	Action
1	Second sem	2017-02-10 00:17:59	Delete
2	Third Sem	2017-02-10 00:18:04	Delete

Figure 5.4: Snapshot of add semester page

5.5 Add/Manage Department:

Here the admin can add or manage department



Department

Add Department

department

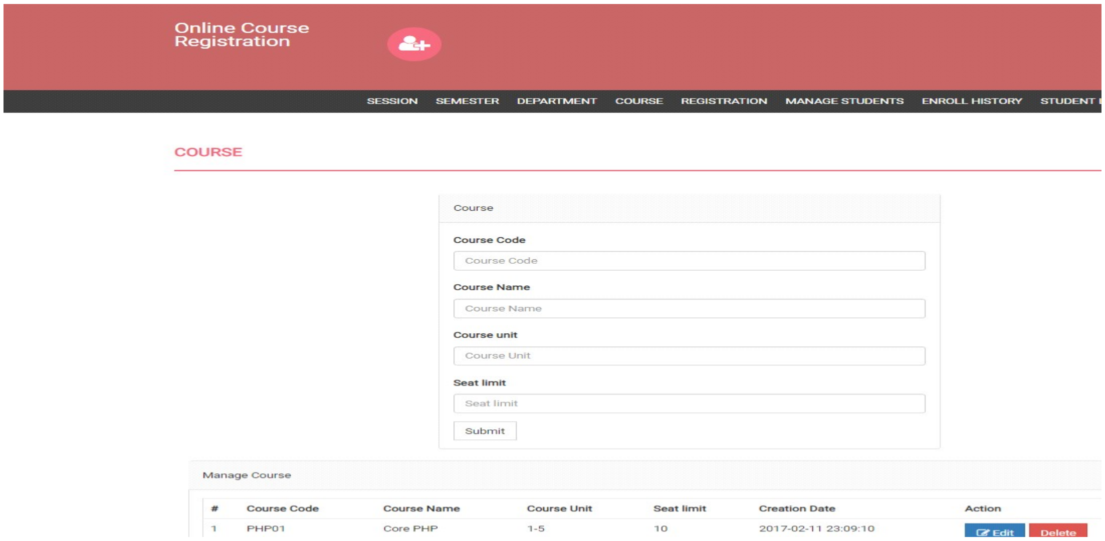
Manage Session

#	department	Creation Date	Action
1	Account	2017-02-10 00:22:00	Delete
2	HR	2017-02-10 00:22:04	Delete
3	Admin	2017-02-10 00:22:08	Delete

Figure 5.5: Snapshot of Department page

5.6 Add/Manage Course

Here admin can add or mange course



Course

Course Code

Course Code

Course Name

Course Name

Course unit

Course Unit

Seat limit

Seat limit

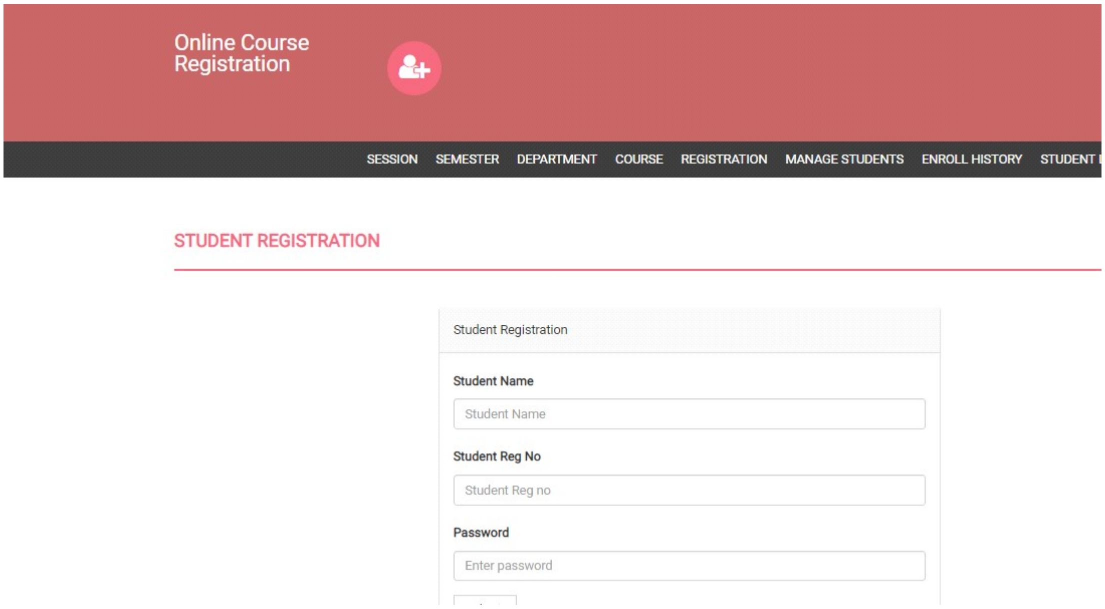
Manage Course

#	Course Code	Course Name	Course Unit	Seat limit	Creation Date	Action
1	PHP01	Core PHP	1-5	10	2017-02-11 23:09:10	Edit Delete

Figure 5.6: Snapshot of Course page

5.7 Student Registration:

Here student can register by entering details



STUDENT REGISTRATION

Student Registration

Student Name

Student Name

Student Reg No

Student Reg no

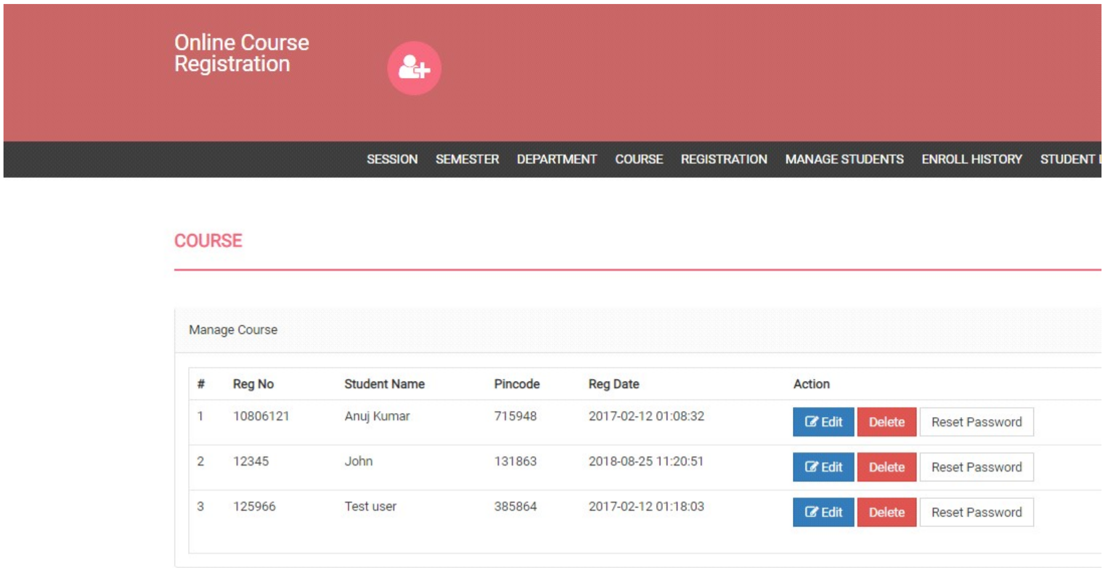
Password

Enter password

Figure 5.7: Snapshot of Student registration page

5.8 Manage Student:

Here student can manage their course



COURSE

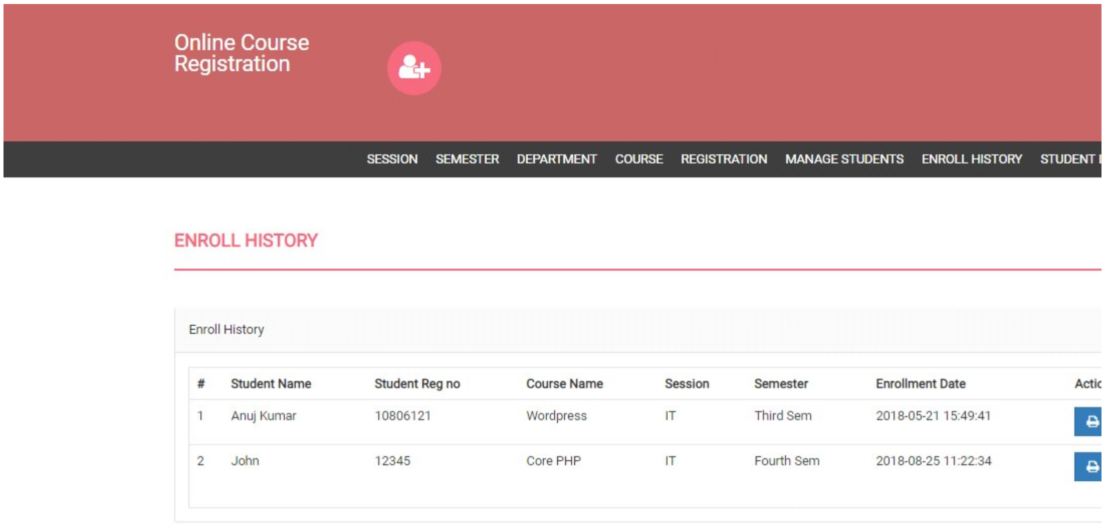
Manage Course

#	Reg No	Student Name	Pincode	Reg Date	Action
1	10806121	Anuj Kumar	715948	2017-02-12 01:08:32	Edit Delete Reset Password
2	12345	John	131863	2018-08-25 11:20:51	Edit Delete Reset Password
3	125966	Test user	385864	2017-02-12 01:18:03	Edit Delete Reset Password

Figure 5.8: Snapshot of Student management page

5.9 Student Enroll History:

Here we can view the student enroll history



ENROLL HISTORY

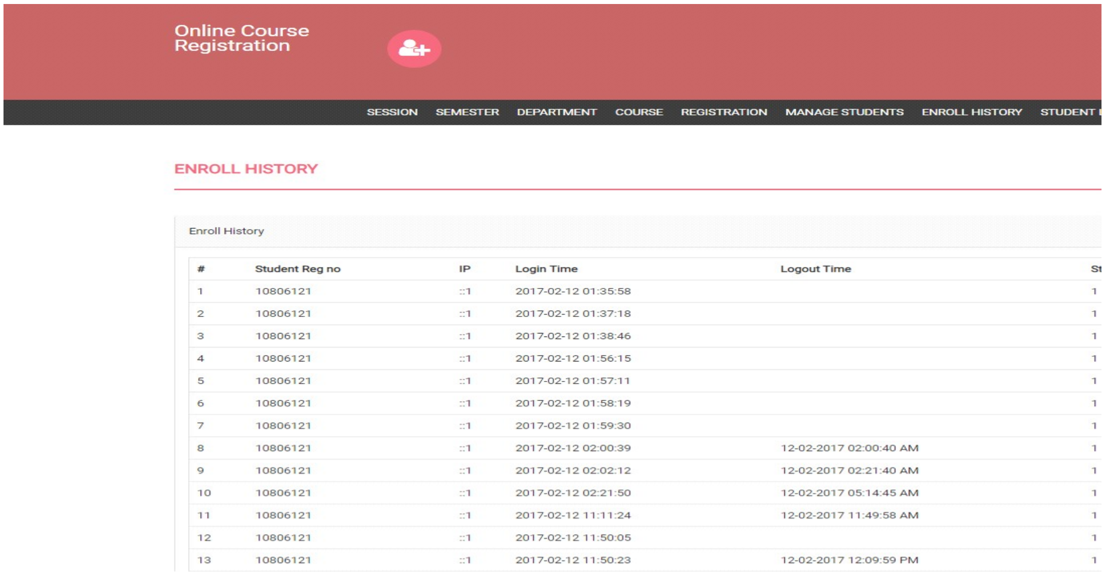
Enroll History

#	Student Name	Student Reg no	Course Name	Session	Semester	Enrollment Date	Actic
1	Anuj Kumar	10806121	Wordpress	IT	Third Sem	2018-05-21 15:49:41	🖶
2	John	12345	Core PHP	IT	Fourth Sem	2018-08-25 11:22:34	🖶

Figure 5.9: snapshot of Student enroll history

5.10 Student Logs:

Here we can view the students login time and details



ENROLL HISTORY

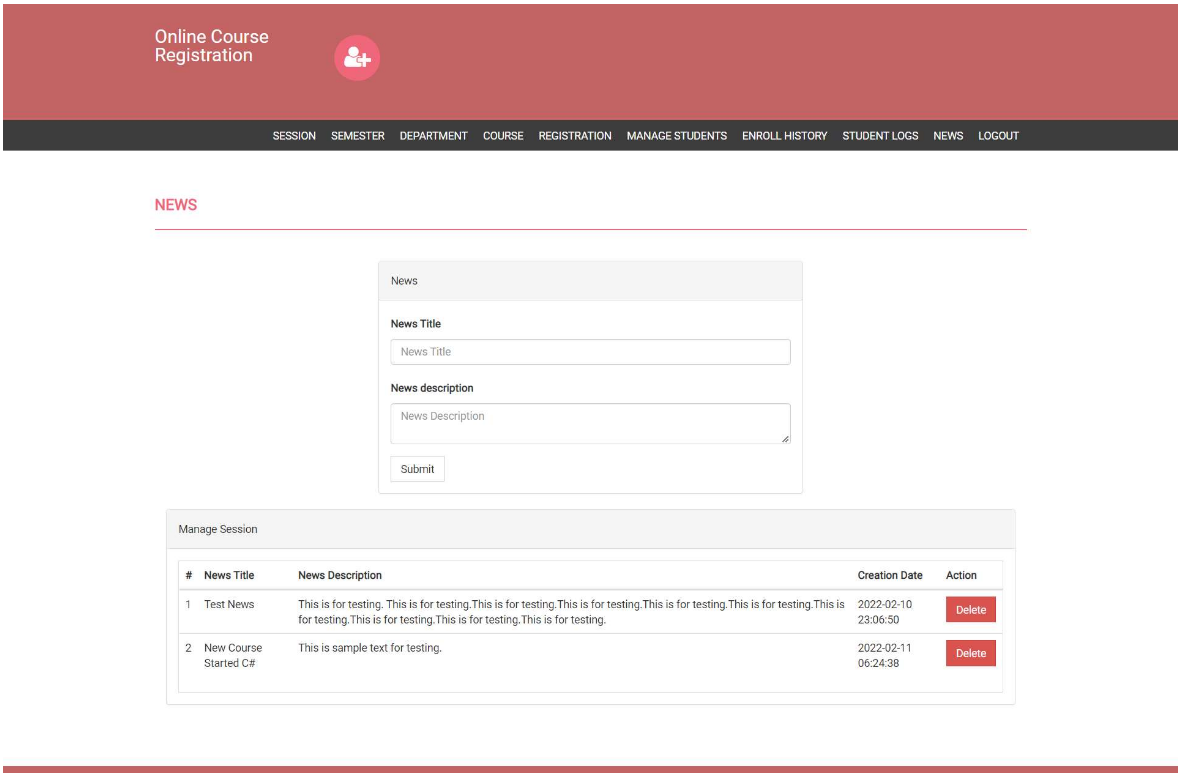
Enroll History

#	Student Reg no	IP	Login Time	Logout Time	St
1	10806121	::1	2017-02-12 01:35:58		1
2	10806121	::1	2017-02-12 01:37:18		1
3	10806121	::1	2017-02-12 01:38:46		1
4	10806121	::1	2017-02-12 01:56:15		1
5	10806121	::1	2017-02-12 01:57:11		1
6	10806121	::1	2017-02-12 01:58:19		1
7	10806121	::1	2017-02-12 01:59:30		1
8	10806121	::1	2017-02-12 02:00:39	12-02-2017 02:00:40 AM	1
9	10806121	::1	2017-02-12 02:02:12	12-02-2017 02:21:40 AM	1
10	10806121	::1	2017-02-12 02:21:50	12-02-2017 05:14:45 AM	1
11	10806121	::1	2017-02-12 11:11:24	12-02-2017 11:49:58 AM	1
12	10806121	::1	2017-02-12 11:50:05		1
13	10806121	::1	2017-02-12 11:50:23	12-02-2017 12:09:59 PM	1

Figure 5.10: Snapshot of Student login

5.11 News:

Here news description and title can be viewed


Online Course Registration
SESSION SEMESTER DEPARTMENT COURSE REGISTRATION MANAGE STUDENTS ENROLL HISTORY STUDENT LOGS NEWS LOGOUT
Submit
Delete

NEWS

News

News Title

News Title

News description

News Description

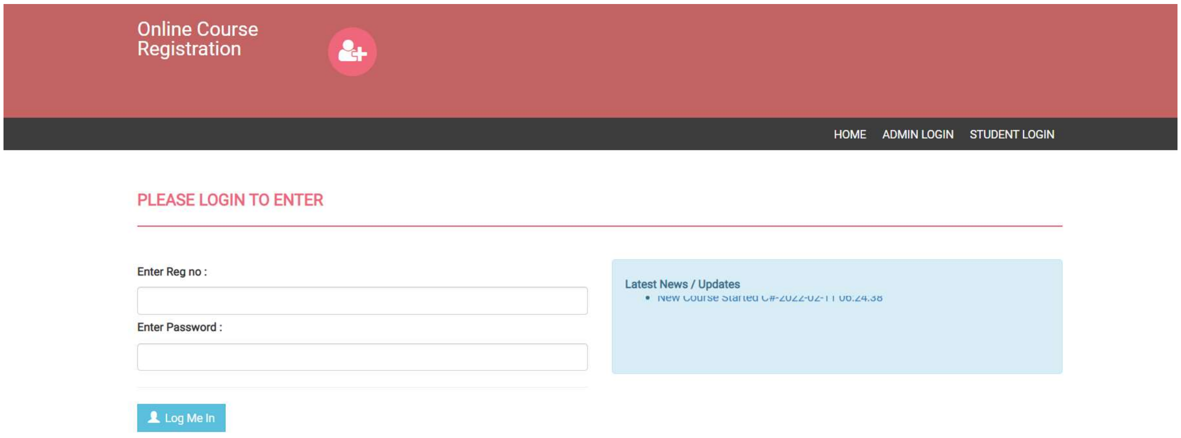
Manage Session

#	News Title	News Description	Creation Date	Action
1	Test News	This is for testing. This is for testing.This is for testing.This is for testing.This is for testing.This is for testing.This is for testing.This is for testing.This is for testing.This is for testing.This is for testing.	2022-02-10 23:06:50	Delete
2	New Course Started C#	This is sample text for testing.	2022-02-11 06:24:38	Delete

Figure 5.11: Snapshot of News page

5.12 Student Login Page:

Here student can login using reg_no and password



PLEASE LOGIN TO ENTER

Enter Reg no :

Enter Password :

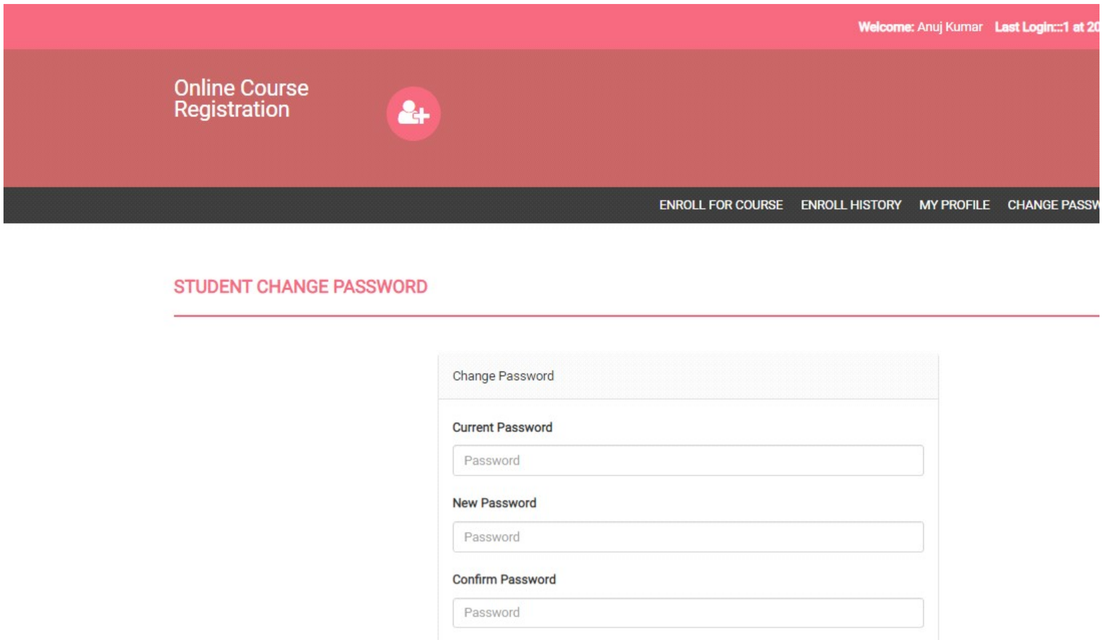
Latest News / Updates

- New Course Started C#-2022-02-11 06:24:38

Figure 5.12: Snapshot of Student login page

5.13 Change Password:

Here student can change the password


Welcome: Anuj Kumar Last Login::1 at 20
Online Course Registration
ENROLL FOR COURSE
ENROLL HISTORY
MY PROFILE
CHANGE PASSW

STUDENT CHANGE PASSWORD

Change Password

Current Password

Password

New Password

Password

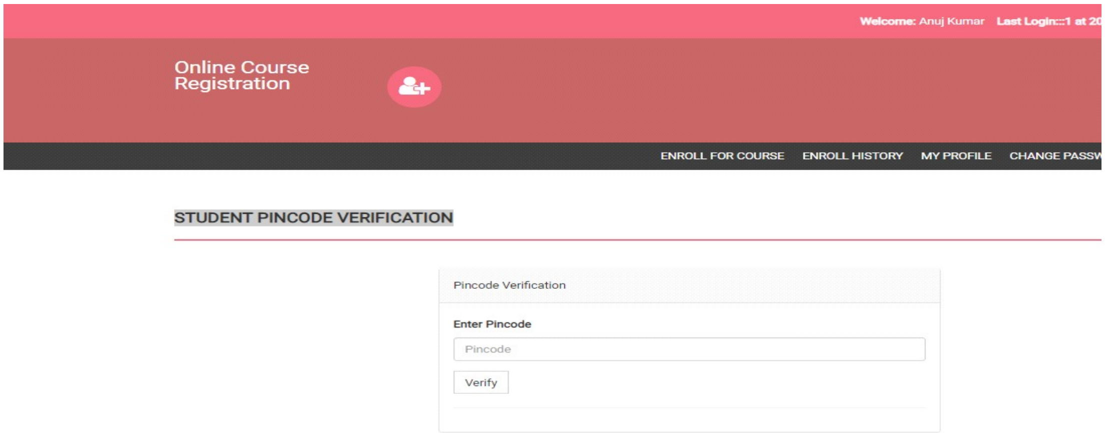
Confirm Password

Password

Figure 5.13: Snapshot of Student change password

5.14 Student Pincode Verification:

Here student pincode verification is done.


Welcome: Anuj Kumar Last Login::1 at 20
Online Course Registration
ENROLL FOR COURSE
ENROLL HISTORY
MY PROFILE
CHANGE PASSW

STUDENT PINCODE VERIFICATION

Pincode Verification

Enter Pincode

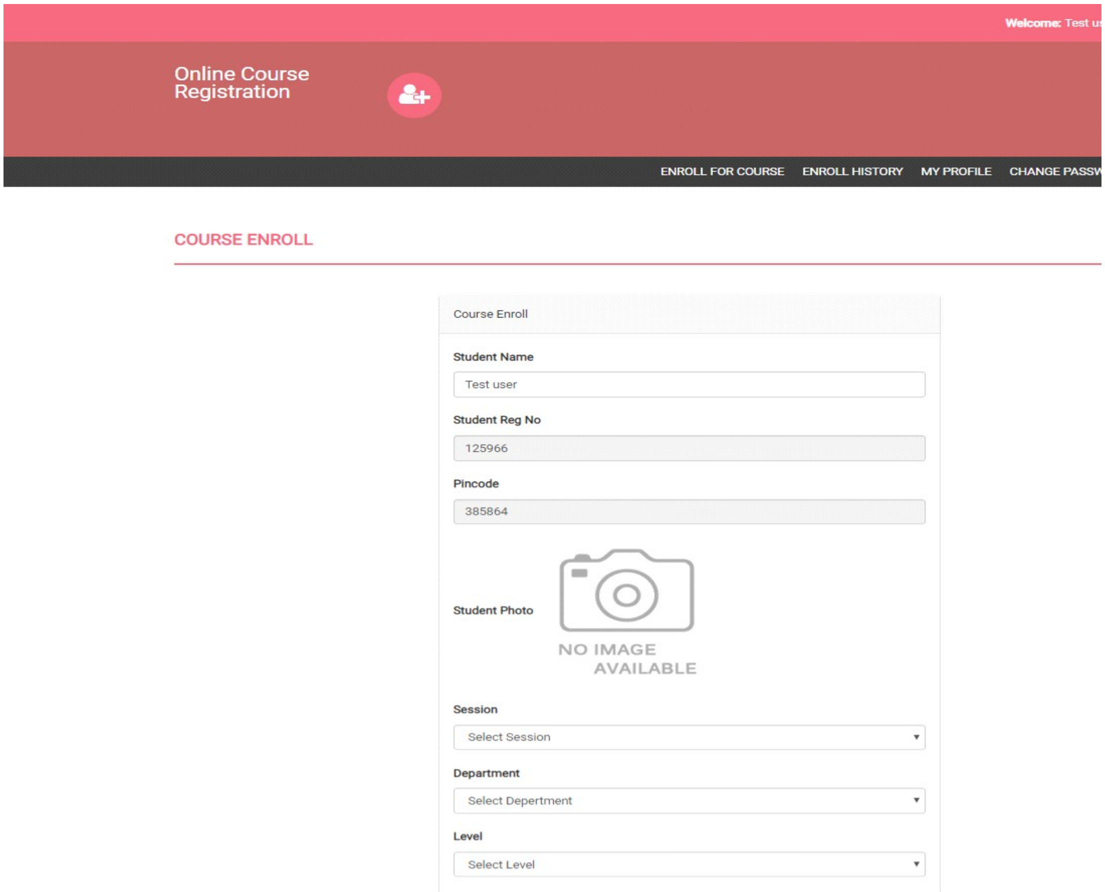
Pincode

Verify

Figure 5.14: snapshot of Student pincode verification

5.15 Course Enroll:

Here student can enroll for a course.



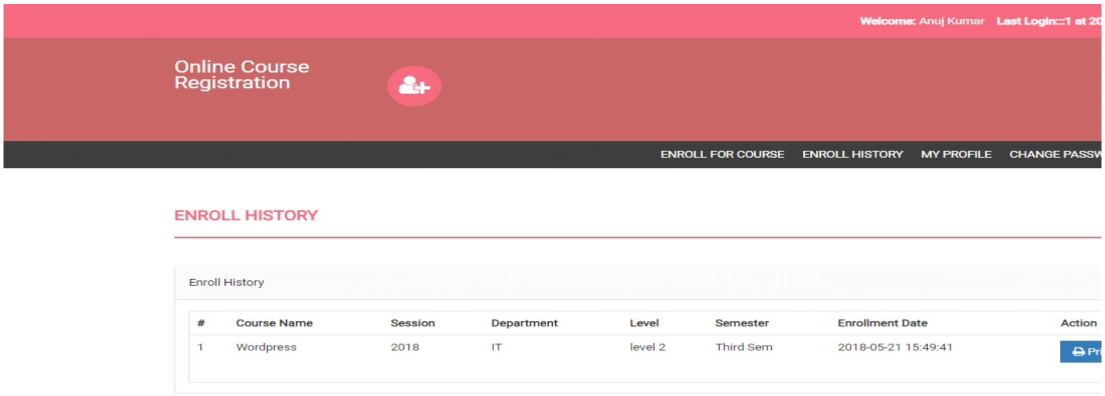
Course Enroll

Student Name	Test user
Student Reg No	125966
Pincode	385864
Student Photo	
Session	Select Session
Department	Select Depertment
Level	Select Level

Figure 5.15: Snapshot of Course enrollment page

5.16 Enroll History:

Here student can view the enroll history



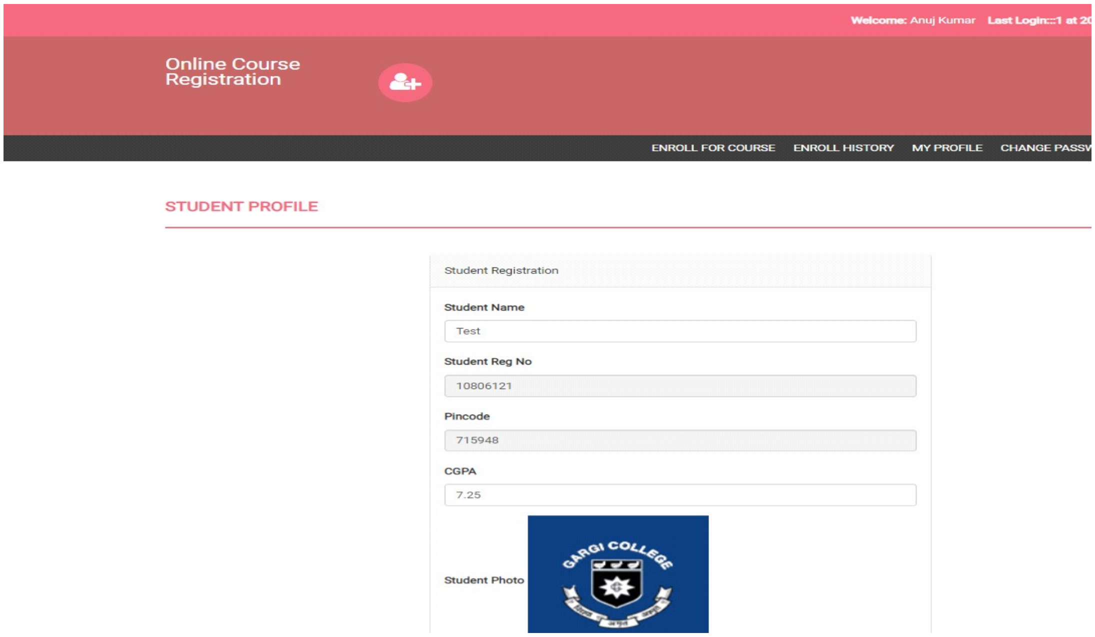
Enroll History

#	Course Name	Session	Department	Level	Semester	Enrollment Date	Action
1	Wordpress	2018	IT	level 2	Third Sem	2018-05-21 15:49:41	Pr

Figure 5.16: Snapshot of Enroll history

5.17 My Profile:

Here student can view thir profile


Welcome: Anuj Kumar Last Login:::1 at 20
Online Course Registration
ENROLL FOR COURSE
ENROLL HISTORY
MY PROFILE
CHANGE PASSW
GARGI COLLEGE

STUDENT PROFILE

Student Registration

Student Name: Test

Student Reg No: 10806121

Pincode: 715948

CGPA: 7.25

Student Photo

Figure 5.17: Snapshot of Student profile page

Chapter 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1. Conclusion

While developing the system a conscious effort has been made to create and develop a software package, making use of available tools, techniques and resources – that would generate a proper System While making the system, an eye has been kept on making it as user-friendly, as cost-effective and as flexible as possible. As such one may hope that the system will be acceptable to any user and will adequately meet his/her needs. As in case of any system development processes where there are a number of shortcomings, there have been some shortcomings in the development of this system also. The project is still under modification. In conclusion, a database is a far more efficient mechanism to store and organize data than spreadsheets; it allows for a centralized facility that can easily be modified and quickly shared among multiple users.

6.2 Future Enhancement

The scope of the project includes that what all future enhancements can be done in this system to make it more feasible to us:-

- Databases for different products range and storage can be provided.
- Multilingual support can be provided so that it can be understandable by the person of any language.
- More graphics can be added to make it more user-friendly and understandable.
- Manage & backup versions of documents online.

REFERENCES

[1] Elmasri, Nawathe, Fundamentals of Database Systems, Pearson Education, 5th Edition, 2006, ISBN- 978-81-317-1625-0.

[2] Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, Tata McGrawHill, 3rd Edition, 2003, ISBN- 0-07-123151-X.

[3] Silberschatz, Korth and Sudarshan, Database System Concepts, Tata McGraw-Hill, 5th Edition, 2002, ISBN- 007-124476-X.

[4] H.P. Mooney, J.W. Evans, ‘A complete relational DBMS for an EMS product’, IEEE Transactions on Power Systems, Volume: 3, Issue: 1, Feb 1988.

[5] Andreas Lubcke , Martin Schaler , Veit Koppen , Gunter Saake, ‘Relational on demand data management for IT-services’ ,2014 IEEE Eighth International Conference on research Challenges in Information Science (RCIS), 28-30 May 2014.

[6]ReferenceLink: https://www.w3schools.com/php/default.asp

https://www.sitepoint.com/php/

https://www.php.net/

[7]ReferenceLink: https://www.mysql.com/

http://www.mysqltutorial.org

[8]ReferenceLink: https://www.apachefriends.org/download.html