

INTRODUCTION

1. Introduction

E-ORDINATIO is a web based application designed for the colleges. In this application seating arrangement of the examinations can be generated easily along with the invigilation duties. The attendance information of the students is available through this. All students can access their timetable and seating arrangement for the current examination. The Invigilation duty reports are generated in the order of date, name or duties attended within a given period through this system, so that the total number of invigilation duties allocated for a teacher can be calculated easily. The invigilator take attendance of the current examination and it can be viewed department wise, room wise, subject wise and date wise by both the faculty and admin thus the number exams attended by a student can be accessed effectively.

The project keeps track of various details in modules such as Students Details, Examination Timing Details, and Hall Details, Invigilation Details with the proper descriptions.

SYSTEM STUDY AND ANALYSIS

2. SYSTEM ANALYSIS

The first phase of software development is system study analysis. The importance of system analysis phase is the establishment of the requirements for the system to acquire developed and installed. Analyzing the project to understand the complexity forms the vital part of the system study. Problematic areas are identified and information is collected. Fast finding or gathering is essential to any analysis of requirement. It is also highly essential that analyst familiarized himself with the objectives, activities and function of organizations in which the system is to be implemented.

2.1 STUDY OF EXISTING SYSTEM

The existing is the manual system. All the procedures are done by the teachers and non teaching staffs in the college by using the written records. To maintain all kinds of reports the college needs to keep track of these files. To get particular seating arrangement details, it is difficult to access as it works manually.

Disadvantages of Existing System:-

- The system is time consuming.
- Human errors can occur
- Less efficient
- Less accurate
- Not user friendly

2.2 STUDY OF PROPOSED SYSTEM

The proposed system promises and enhances the development of organization by improving their day to day operation style and maintenance of data as an instant mode of operation. From writing and entering each details that take a lot of time and probably makes error and staff can easily make the above listed process through the proposed system simply with a little concentration that we are expecting from them.

The proposed system can easily make seating arrangement reports, Invigilation Duty reports and as a sum that reduces the man power, number of staffs will not be a matter. While comparing to the existing to the proposed system the security for data which were stored is more and unauthorized people cannot access the data. So the data have more safety and more secure than the existing system. The data which where already stored cannot be altered without the permission of the authorized person.

The development of this new system contains the following activities, which try to automate the entire process keeping in the view of database integration approach.

ADVANTAGES OF THE PROPOSED SYSTEM

- Students can easily access their informations related to examination such as timetable, seat arrangement.
- Automatic seat arrangement and invigilation duty allocation
- Teachers can view the students who have attended the examination department vice and subject vice
- More user friendly and increased accuracy.
- Higher processing speed.
- Reduce paper work.
- Access data easily.
- Data updating is easy.

2.3 FEASIBILITY STUDY

Feasibility study is a test of the system proposal according to its workability, impact on the organization, ability to meet user and effective use of resources. The objective of feasibility study is not to solve the problem. But to acquire a sense of its scope. During this study, the problem definition is crystalized and benefits are estimated with greater detail at this stage. The result of feasibility study is a system formal proposal. The best system meets the performance requirements at least cost. Since E-ordinatio is a web based application it has no additional expenses to implement.

Types of feasibility:

- Economic feasibility
- Technical feasibility
- Operational feasibility

Technical Feasibility

This study is performed to check whether the proposed system is technically feasible or not. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. Technical feasibility centers on the existing computer system (hardware, software, etc.) and to what extent it can support the proposed addition. This involves financial consideration to accommodate technical enhancement. This system E ordinatio is technically feasible, all requirements including the details of teachers and students are stored in files.

Economic Feasibility

It determines whether the required software is capable of generating financial gain for an organization. It involves the cost incurred on the software development team, estimate cost of hardware and the software, cost of performing feasibility study and so on. The development cost for this software is low. This software, E ordinatio meets the requirements of the users with minimum cost.

Operational Feasibility

E ordinatio is beneficial since it can be turned out into information system that will meet the organization's operating requirements. Operational feasibility accepts of the project are to be taken as an important part of the project implementations. Using different command buttons throughout the application program enhances the operational feasibility. Thus the maintenance and modification is made so easier. E ordinatio fits to the college environment and objectives with regard to the development schedule.

REQUIREMENT ANALYSIS

3. REQUIREMENT ANALYSIS

System configuration means the essential facilities for working of our new application software. There are mainly two classifications for the system configuration. They are:

3.1 HARDWARE SPECIFICATION:

The selection of hardware is very important and proper functioning of any software, the size and capacity requirements are also important.

- Processor : Intel Pentium
- Hard Disk : 40 GB.
- Monitor : Color monitor
- Mouse : Any mouse
- Ram : 2 GB.
- Keyboard : Standard keyboard

3.2 SOFTWARE SPECIFICATION:

One of the most difficult task is selecting software. Once the system is found then we have to determine a particular software package.

- Operating system : Windows 7 and Above
- Front- end : php
- Tool : microsoft expression web
- Back-end : SQL Yog

3.3 ABOUT PHP

PHP is a server side language designed for web development but also used as general purpose programming language. PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or a common interface (CGI) executable. After the PHP code is interpreter and executed, the web server sends

resulting output to its client, usually in form of a part of the generated web page; for example' PHP code can generate a web page's HTML code, an image, or some other data, PHP has also evolved to include a command-line interface (CLI) capability and can be used in standalone graphical applications.

PHP is a general-purpose scripting language that is especially suited to server-side web development – purpose scripting language that is especially suited to serve-side web development, in which case PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere. It can also be used for command-line scripting and client-side graphical user interface (GUI) applications. PHP can be deployed on relational database management systems (RDBMS). Most web hosting providers support PHP for use by their clients. It is available free of charge, and the PHP group provides the complete source code for users to build, customize and extend for their own use.

PHP acts primarily as a filter, taking input from a file or stream containing text and /or PHP instructions and outputting another stream of data. Most commonly the output will be HTML, although it could be JSON, XML or binary data such as image or audio formats. Since PHP 4, the PHP parser compiles over its interpreter predecessor. Originally designed to create dynamic web pages, PHP now focuses mainly on server –side scripting and it is similar to other server –side scripting languages that provides dynamic content from a web server to a client, such as a Microsoft's ASP. NET, Sun Microsystems Java server pages, and mod perl. PHP has also attracted the development of many software frameworks that provides building blocks and design structure to promote Rapid Applications Development (RAD). Some of these includes PRADO, Cake PHP, Symfony, DodeIgniter, Laravel, Yii Framework, Phalcon and Send Framework, offering features similar to other web application framework. PHP is mainly focused on server-side scripting, so you can do anything any other CGI program can do, such as collect from data, generate dynamic page content, or send and receive cookies. But PHP can do much more.

Server-side scripting: This is the most traditional and main target field for PHP. You need thre things to make this work. The PHP parser (CGI or Server module), a web server and a web browser. You need to run the web server, with a connected PHP installation. You can access the PHP program output with a web browser, viewing the PHP pages through the server. All theses can run on your home machine if you are just experimenting with PHP programming. See the installation instruction section for more information.

ABOUT SQLYOG

SQLyog is the most powerful manager, admin and GUI tool for MySQL, combining the features of MySQL Query Browser, Administrator, phpMyAdmin and other MySQL Front Ends and MySQL GUI tools in a single intuitive interface. SQLyog is a fast, easy to use and compact graphical tool for managing your MySQL databases. SQLyog was developed for all who use MySQL as their preferred RDBMS. Whether you enjoy the control of handwritten SQL or prefer to work in a visual Environment, SQLyog makes it easy for you to get started and provides you with tools to enhance your MySQL experience. MySQL manager and admin tool SQLyog provides you with powerful means to manage your MySQL databases. It features the simplicity of MySQL Front, with the power of EMS MySQL Admin. SQLyog provides detailed profile information for every SQL statement executed.

SQLyog has a MySQL manager that restores your previous session the way you left it. Be it a system crash or accidentally closing your MySQL client. SQLyog v0.9 was first released to the public in 2001 as after eight months of development. SQLyog was available free of charge, but with closed source code, until v3.0 when it was made a fully commercial software. Nowadays SQLyog is distributed both as free software as well as several paid, proprietary, versions. The free software version is known as Community Edition at GitHub.

It allows easy access to frequently used SQL scripts. The script files can be stored as files from the SQLyog interface or link to an existing file anywhere where windows can access it - on a local drive or a shared network drive. The SQL scripts and the file links can be organized in folder and subfolders. SQLyog gives you a choice between four different connectivity options. No matter whatever your firewall/proxy/ISP settings are, you should be able to smartly manage your MySQL Server(s).

SYSTEM DESIGN

4.1 ENTITY RELATIONSHIP DIAGRAM

A diagram that depicts set of real world entities and the local relationships among them. An ER Diagram comprises the data objects, entities, attributes, relationships.

Data Object : It is representation of composite information used by the software.

Entity : It is the data that stores information about the system in a database.

Attributes : It describes the property of a data object

Relationships : Connection of objects or entities with each other.

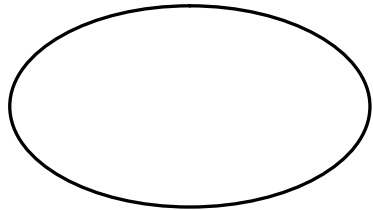
4 TYPES OF RELATIONSHIPS

- i. One to one relationship : One instance of an entity is related to another instance of another entity
- ii. One to many relationship : One instance of an entity is related to several instance of another entity
- iii. Many to many relationship : Many instance of an entity is related to several instance of another entity
- iv. Many to one relationship : Many instance of an entity is related to another instance of another entity

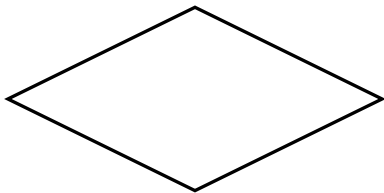
ER DIAGRAM NOTATIONS



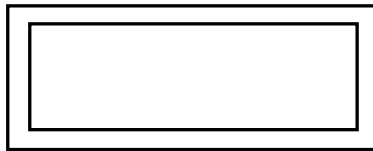
A rectangle box represents an entity



A eclipse represents an attribute



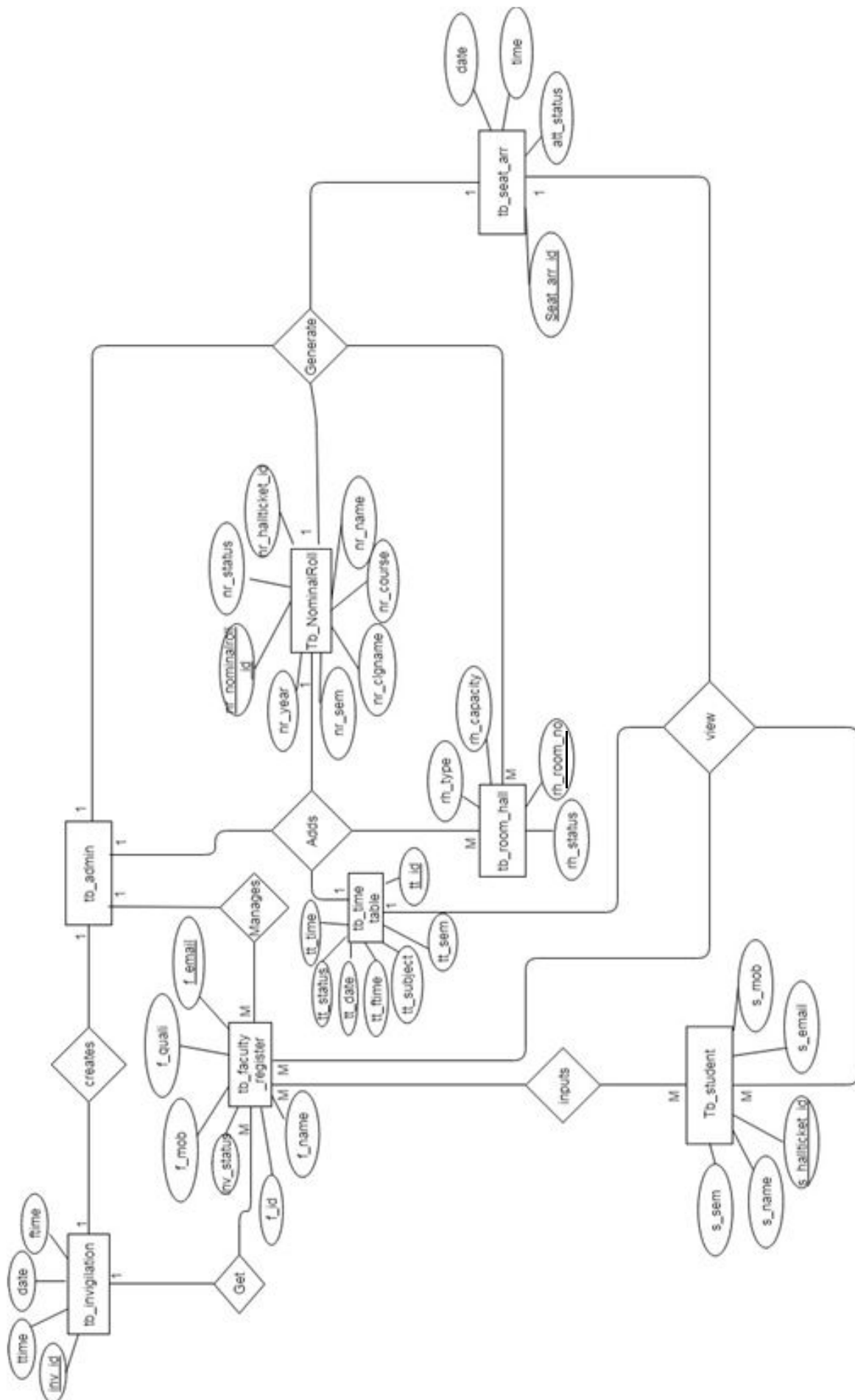
Represents a relationship



Represents a weak entity



Represents a Database



4.2 DATABASE DESIGN

A database design is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called information level design and is taken independent of any individual DBMS.

In the second step, this information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called information Physical level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs in parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives

- Data Integrity
- Data independency

The database design is a logical development in methods used by the computer to access and manipulate data stored in the various parts of the computer system. Database is defined as an integrated collection of data. The overall objective in the development of the database technology has been treat data as organization resources and as an integrated while. The main objectives of database are data integration, data independence.

The database is implemented by using SQL. SQL is the most popular open source database server in existence. It is very commonly used in conjunction with php to create powerful and dynamic server side applications

Table Name: tb_department

Name	Data Type	Constraint	Description
dept_id	int	primary key	department id
dept_name	varchar(50)	not null	name of the department

Table Name: tb_faculty_register

Name	Data Type	Constraint	Description
dept_id	int	foreign key	department id
f_email	varchar(50)	primary key	faculty mail id
f_name	varchar(50)	not null	name of the faculty
f_quali	varchar(50)	not null	Qualification
invstat	varchar(50)	not null	name of the department
f_mob	int	not null	mobile number

Table Name: tb_invigilation

Name	Data Type	Constraint	Description
f_id	Varchar(50)	foreign key	name of the department
Inv_id	int	Primary key	invigilation id
rh_room_no	Varchar(50)	foreign key	room number
date	date	Not null	Date of invigilation
ftime	date	Not null	From time
ttime	date	Not null	To time

Table Name: tb_nominalroll

Name	Data Type	Constraint	Description
nr_nominalroll_id	int	primary key	nominal roll id
nr_hallticket_id	varchar(50)	not null	register no
nr_clgname	varchar(50)	not null	college name
nr_name	varchar(50)	not null	student name
nr_year	varchar(50)	not null	year
nr_sem	varchar(50)	not null	semester
nr_course	varchar(50)	not null	course
nr_status	int	Not null	Nominal roll status

Table Name: tb_room_hall

Name	Data Type	Constraint	Description
rh_room_no	Varchar(50)	Primary key	room no
rh_capacity	int	not null	capacity of the room
rh_type	Varchar(50)	not null	room/hall
rh_status	int	Not null	Status of the room

Table Name: tb_seat_arr

Name	Data Type	Constraint	Description
seat_arr_id	int	primary key	seat arrangement id
nr_nominalroll_id	int	foreign key	nominal roll id
rh_room_no	varchar(50)	foreign key	room no
att_status	varchar(50)	not null	attendance status
date	date	Not null	Date of examination
time	date	Not null	Time of examination

Table Name: tb_student

Name	Data Type	Constraint	Description
s_name	varchar(50)	not null	student name
s_sem	varchar(50)	not null	semester
s_hallticket_id	varchar(50)	primary key	register no
s_mob	int	not null	mobile number
s_email	varchar(50)	not null	mail id
dept_id	int	foreign key	department id

Table Name: tb_timetable

Name	Data Type	Constraint	Description
tt_date	date	not null	date of the exam
tt_sem	varchar(50)	not null	semester
tt_id	int	primary key	Time table id
dept_id	int	foreign key	department id
tt_subject	varchar(50)	not null	subject
tt_status	int	not null	Time

Table Name: tb_login

Name	Data Type	Constraint	Description
username	Varchar(50)	Primary key	username
password	Varchar(50)	not null	password
type	Varchar(50)	not null	admin/faculty/student

4.3 NORMALIZATION

Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependence. Redundant data wastes disk space and creates maintenance problem, if the data exist is more then one place must be changed, the data must be changed in exactly the same way in all occasions. There are few rules for database normalization. Each rule is called a normal form. If the first rule is observed, the database is to be in “First normal form”. If the first two rules are observed, the database is to be in “Second normal form”. If the first three rules are observed, the database is to be in “Third normal form”. Although other levels of normalization are possible third normal form is considered the highest level necessary for most applications; Normalization is carried out in the system for the following reasons:

- To structure the data so that there is no repetition of data, this helps in saving much space of the system.
- To permit simple retrieval of data in response to query and report request
- To simplify the maintenance of data through updates, insertion, deletions
- To reduce the need to restructure or recognize data which new applications requirement arise.

FIRST NORMAL FORM

The first normal form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be single value from the domain of that attribute in other words 1NF disallows.”Relations with Relations” and “Relations as Attribute values within tuples”,the only attribute values permitted by 1NF are single atomic or visible values.

The first step is to put the data into first normal form.Moving data into separate tables where the data is of similar type in each table can do this,each table is given a primary key of foreign key as per requirement of project.In this we form new relations for each mono atomic attribute or nested relation.This eliminates repeating group of data.A relation is said to be in first form if and only if it satisfies the constrains that contains primary key only.

SECOND NORMAL FORM

According to second normal form, for relations where primary key contains multiple attributes, no non key attribute should be functionally dependent on a part of primary key. A relation is said to be in first form if and only if it satisfies all the first normal form conditions for the primary key and every non primary key attributes of the relation is fully dependent on its primary alone.

THIRD NORMAL FORM

According to third normal form relation should not have a non key attribute functionally determine the another non key attribute or by a set of non key attributes. That is, there should be no transitive dependency on the primary key

A relation is said to be in third normal form if and only if it is in second normal form and moreover the non key attributes of the relation should not be depend on the other non key attribute.

E-ORDINATIO is in third normalized form. We removed the chances for keeping data more than once. Then we set primary key to all tables. then it satisfies the first normal form. To make tables in second normal form we implemented the concept of foreign key. We have deleted the field which does not depend on the primary key to satisfy the third normal form. So our tables are become normalized. It helps to reduce the complexity in our project.

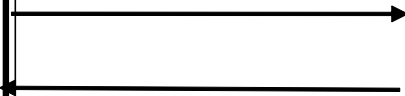
4.4 DATA FLOW DIAGRAM

A diagram that depicts data sources, data sinks, data storage and process performed on data as nodes and logical flow of data as links between the nodes

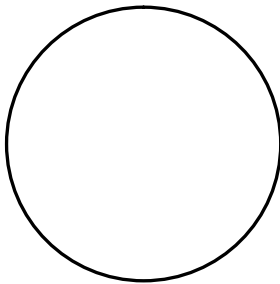
The four symbols used in DFDs are:-



Represents the source and destination of data within the system



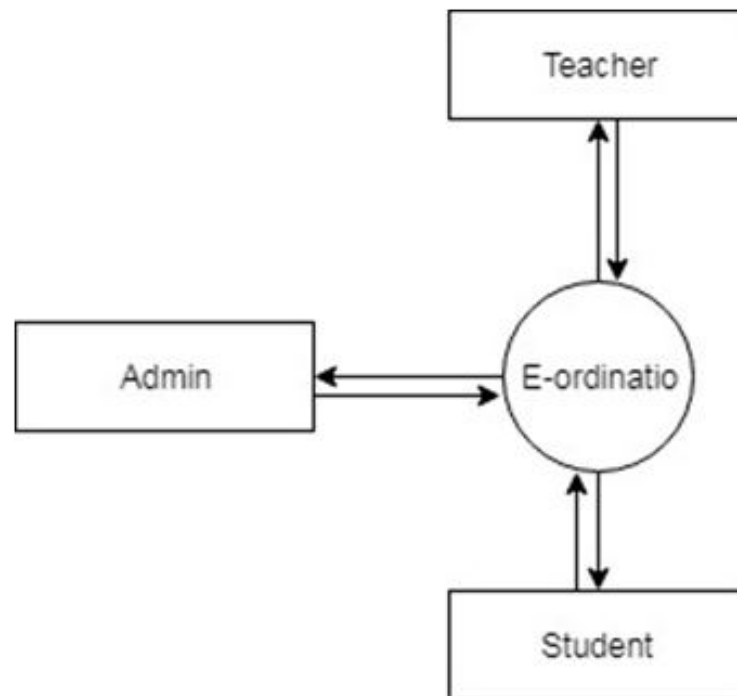
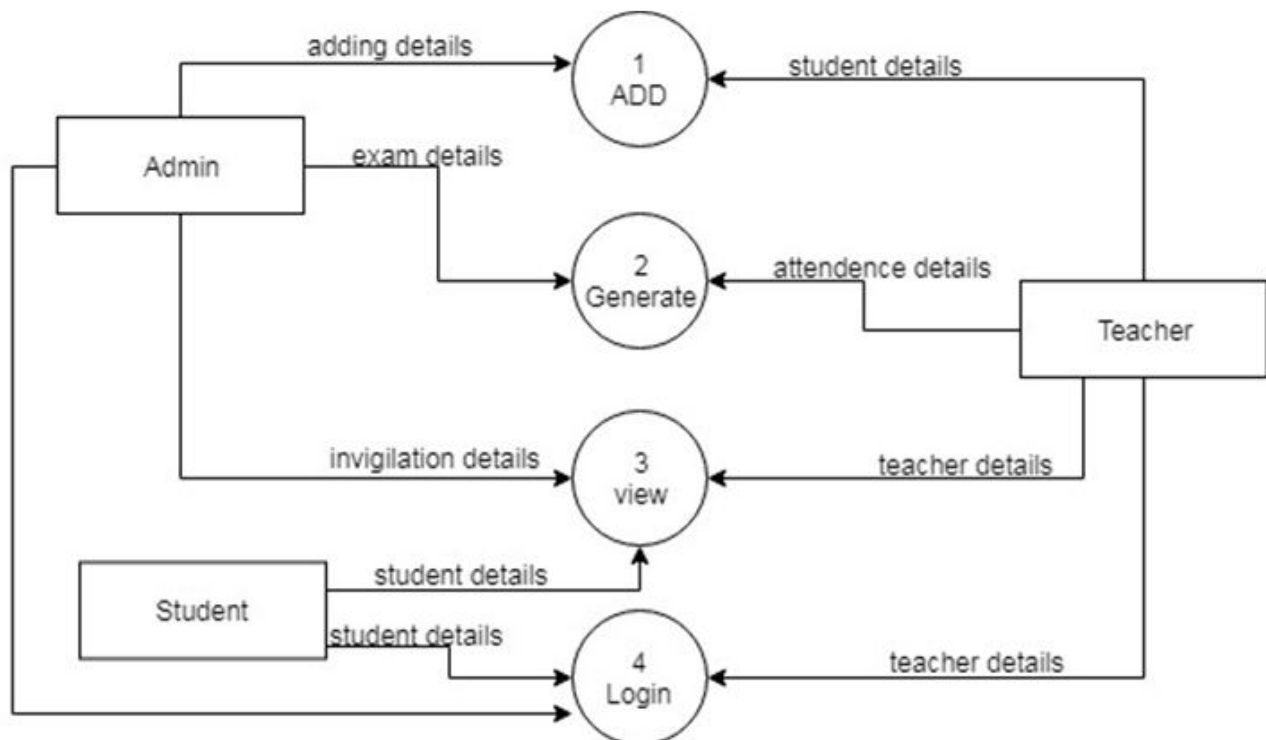
Represents the movement of data from its source to destination within the system

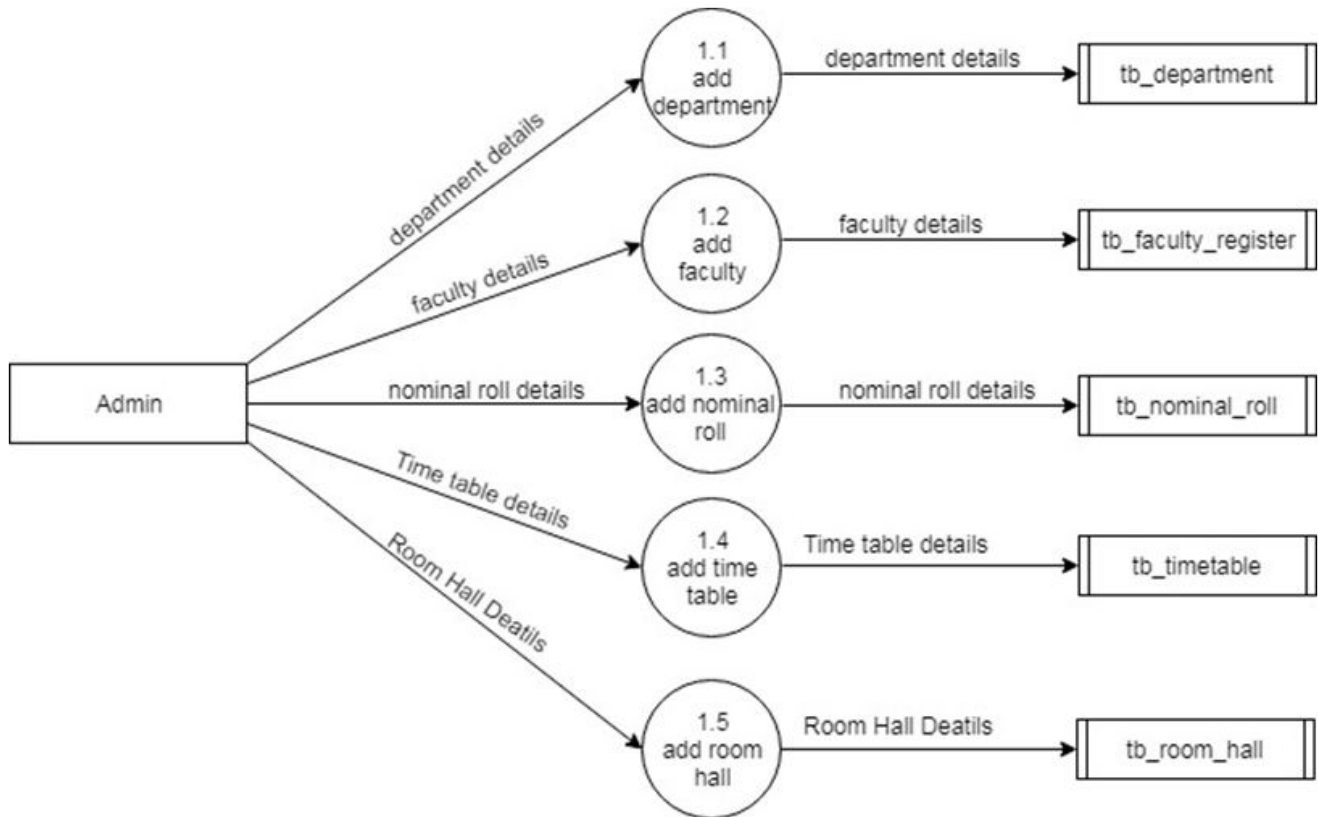
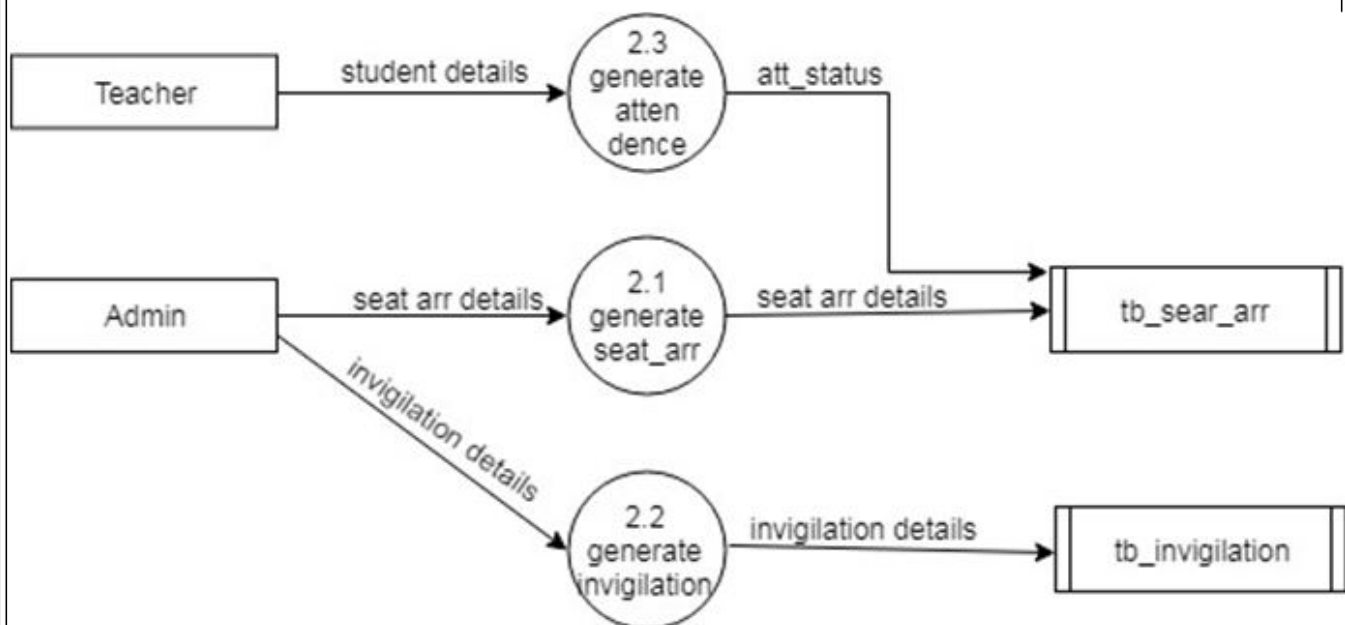


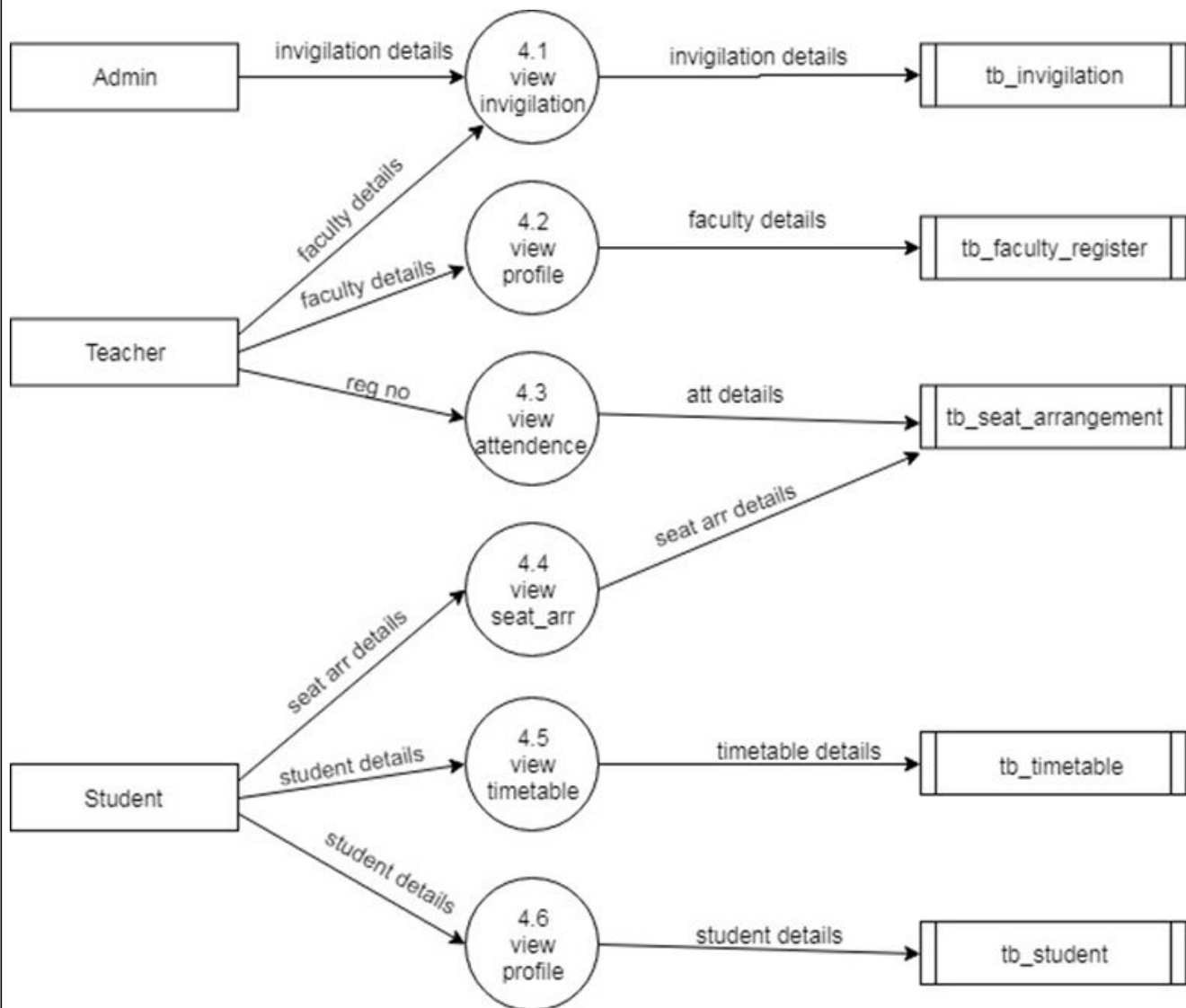
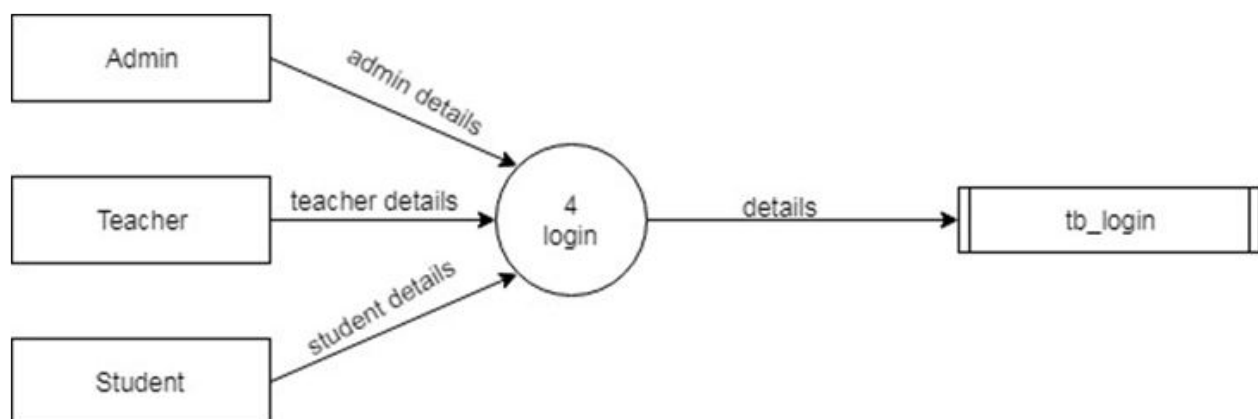
Represents a process that transforms a data



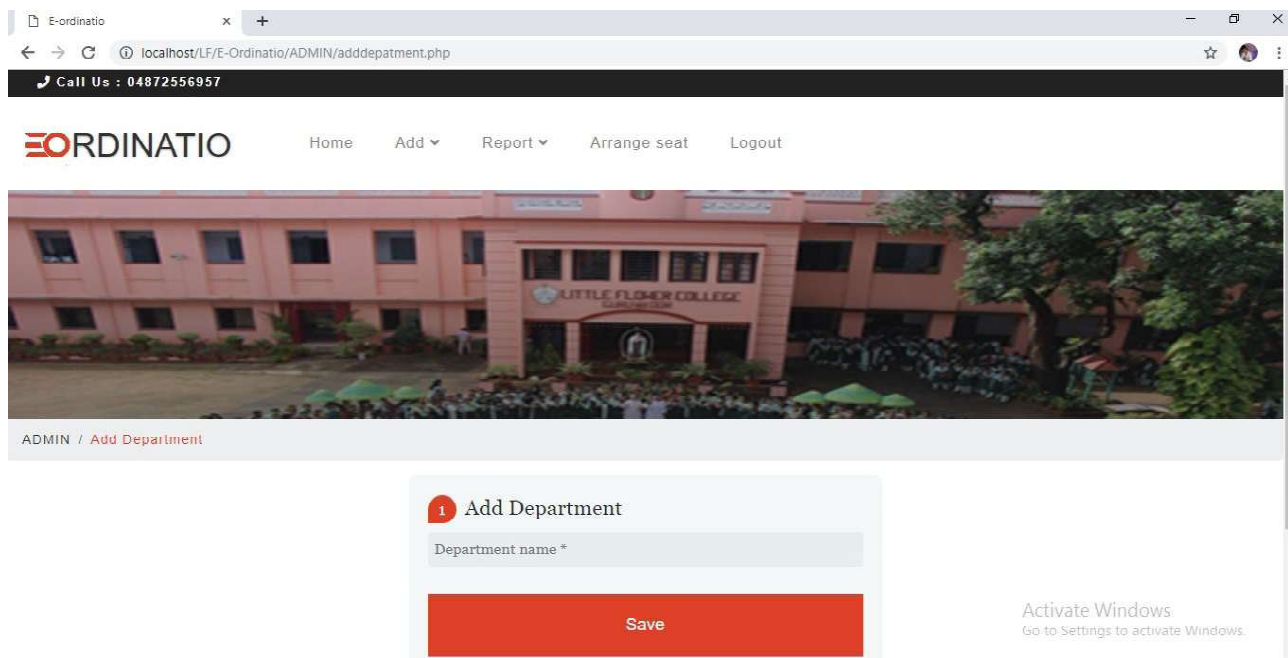
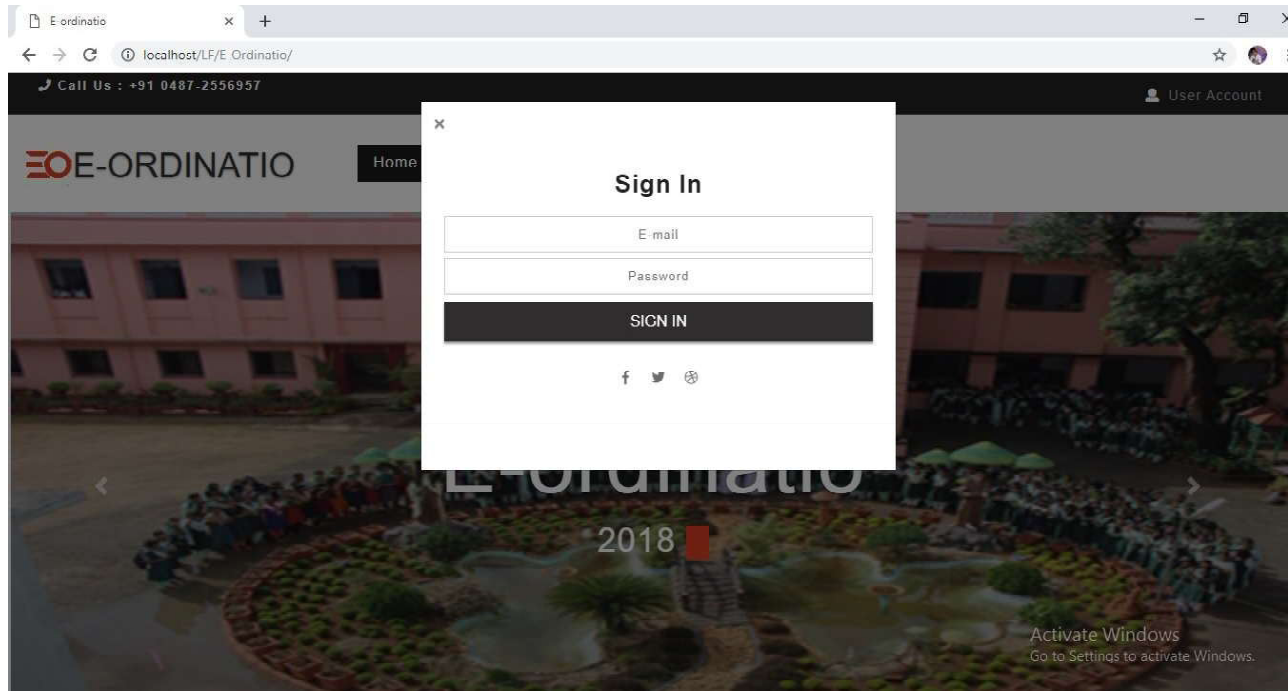
Represents the data storage

LEVEL 0LEVEL 1

LEVEL 2ADDGENERATE

VIEWLOGIN

FORM DESIGN



E-ordinatio x +

localhost/LF/E-Ordinatio/ADMIN/addfaculty.php

1 Faculty Info

Your Name *

Your Mobile *

Your Qualification *

Your Email *

Department:

Select Department

2 Login Info

Your Email *

Password

Save

Activate Windows
Go to Settings to activate Windows.

E-ordinatio x +

localhost/LF/E-Ordinatio/ADMIN/nominalroll.php

ADMIN / Add Nominal roll

1 Nominal Roll Info

Halltricker number *

College name *

Student name *

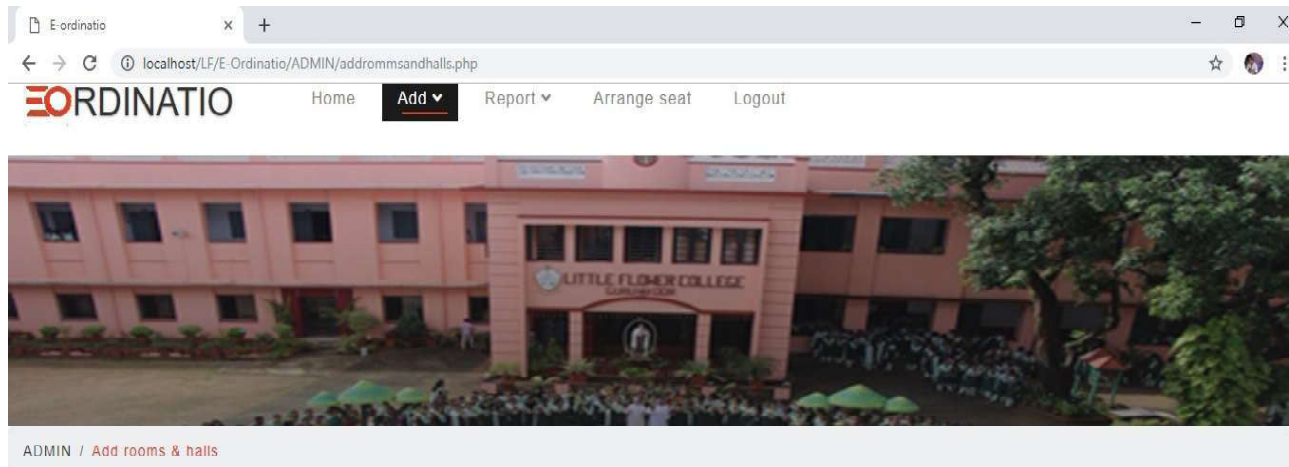
Year *

course *

--Select Semester--

Save

Activate Windows
Go to Settings to activate Windows.



1 Add Rooms and halls

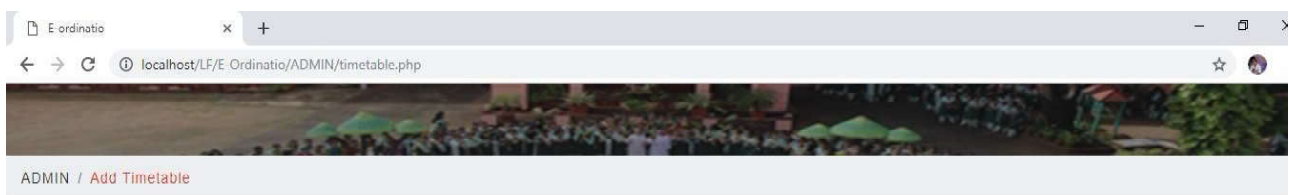
Room or hall number *

Room capacity *

☐ Room ☐ Halll

Save

Activate Windows
Go to Settings to activate Windows.



1 Add Timetable

dd-mm-yyyy

--:--

Subject

Semester:

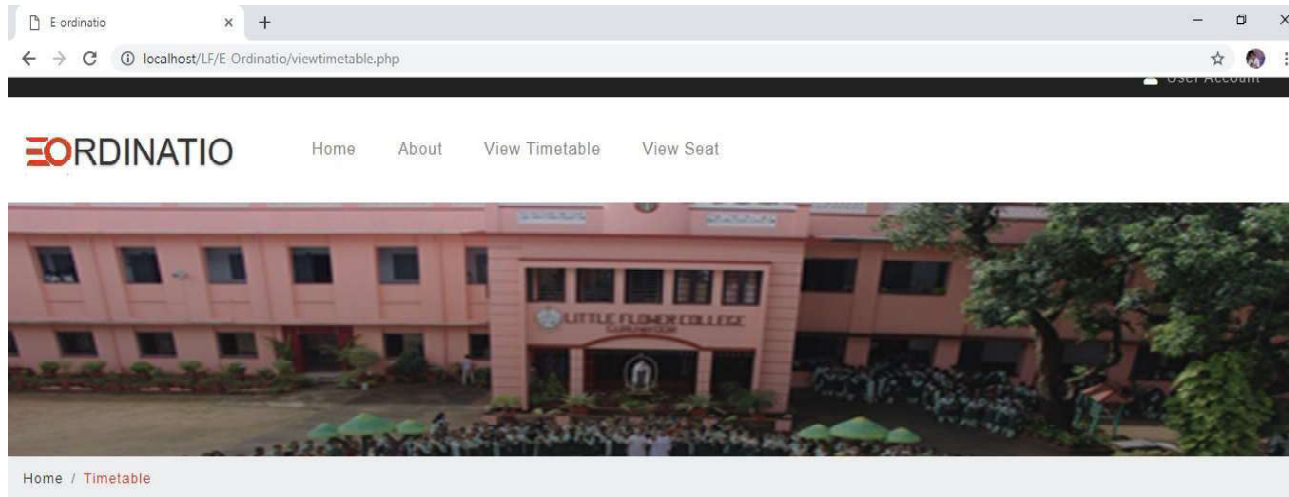
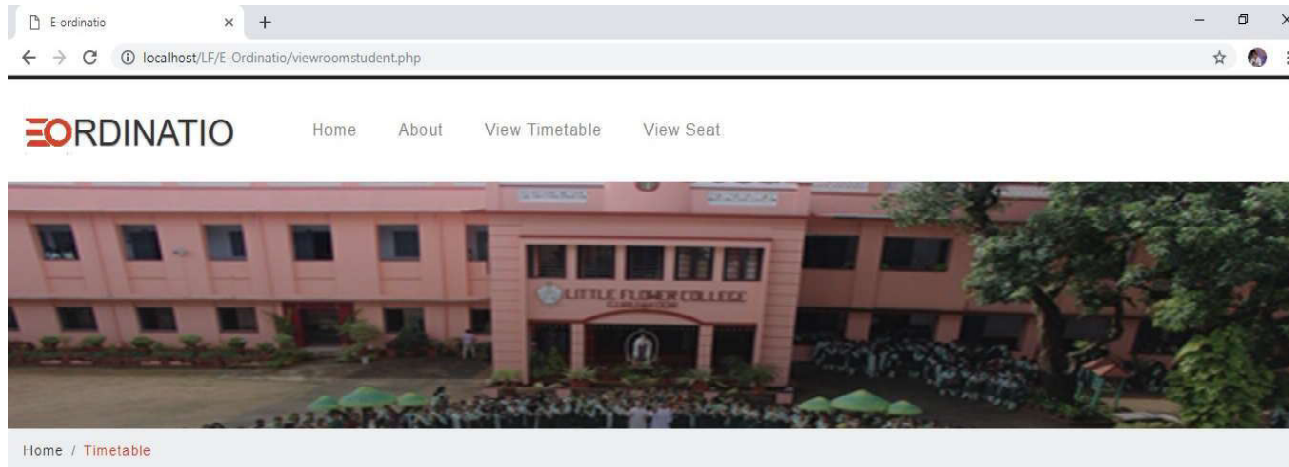
--Select Semester--

Department:

--Select Department--

Apply

Activate Windows
Go to Settings to activate Windows.




E-ordinatio x +

localhost/LF/E-Ordinatio/ADMIN/attendance_per_dept.php

Call Us : 04872556957

EORDINATIO Home Add ▾ Report ▾ Arrange seat Logout




Attendance

Department ▾ Semester ▾ Date

Activate Windows
Go to Settings to activate Windows.

E-ordinatio x +

localhost/LF/E-Ordinatio/TEACHER/Semincrement.php



Teacher / Sem Increment

Sem Increment

Department ▾ Semester ▾

Activate Windows
Go to Settings to activate Windows.

CODING

Seatarrangement.php

<?php

include("subhead.php");

?>

<p style="font-size:x-large;color:#3366CC;text-align:center;padding-bottom:10px;"></p>

<?php

// \$sem=\$_GET['examsem'];

// \$count=\$_GET['count'];

include './connect.php';

\$wwd=\$_GET['date1'];

// \$wwd="2018-11-02";

\$t=\$_GET['t'];

\$select="select * from tb_room_hall where rh_status='0'";

\$sel=mysql_query(\$select);

\$sql="SELECT tt_sem,'dept_name' FROM 'tb_timetable' INNER JOIN 'tb_department' ON 'tb_timetable'. 'dept_id'='tb_department'. 'dept_id' WHERE 'tt_date'='\$wwd'";

\$sq=mysql_query(\$sql);

\$count=0;

\$checkdetails="SELECT * FROM 'tb_nominalroll' WHERE 'nr_sem' IN (SELECT 'tt_sem' FROM 'tb_timetable' WHERE 'tt_date'='\$wwd') AND 'nr_course' IN (SELECT 'dept_name' FROM 'tb_department' WHERE 'dept_id' IN (SELECT 'dept_id' FROM 'tb_timetable' WHERE 'tt_date'='\$wwd'))";

\$schedeta=mysql_query(\$checkdetails);

\$numcheck=mysql_num_rows(\$schedeta);

\$fdep="";


```

$fssem="";

while($s=mysql_fetch_array($sq))
{
    $se=$s[1].":".$s[0]."|";
    $fdep1=$s[1].", ";
    $fssem1=$s[0].", ";
    $count++;
}

$sem=trim($se,"|");
$fdep="" . rtrim($fdep1, ", "). "";
$fssem="" . rtrim($fssem1, ", "). "";

//echo "SEM :". $sem;

//echo "<br>COUNT :". $count;

$nominal=mysql_query("select * from tb_nominalroll where nr_course in($fdep) and nr_sem
in($fssem)");

$nominalnum=mysql_num_rows($nominal);

if($nominalnum>0)
{
    if($t=="m")
    {
        $tim="10:00";
        $ftime="9:30:00";
        $ttime="12:30:00";
    }
}

```

```

else {

    $tim="2:00";

    $ftime="1:30:00";

    // echo $ftime;

    $ttime="4:30:00";

}

/*—————EXAMPLE CODE—————*/

$f=0;

$var=array();

$varsub=array();

    while($fet=mysql_fetch_row($sel))

    {

        $i=0;

        $c=0;

        $rom=$fet[0];

        $roomno=$fet[1];

        // echo "<br>$rom<br>";

        $center= round($roomno/3);

        $side= round($roomno-$center);

        $spdepsem=explode("|",$sem);

        echo "<br>side :::::::::::$side<br>";

        // echo "<br>center :::::::::::$center<br>";

        // echo "<br>—————

```

```

<br>";

foreach ($spdepsem as $sds)
{

    $semdep= explode(":", $sds);

    $department=$semdep[0];

    $semester=$semdep[1];

    //echo "<br>Enter foreach first::::$semdep[0]<br>";

    for ($k=0;$i<$side;$k++)
    {

        // echo "<br>Enter k :$k<br>";

        // echo "<br>i value : $i";

        $a=mysql_query("select * from tb_nominalroll where nr_course='$department' and
nr_sem='$semester' and nr_nominalroll_id not in (select nr_nominalroll_id from tb_seat_arr where
date='$wwd') limit 1 ");

        $fe=mysql_fetch_row($a);

        $fenu=mysql_num_rows($a);

        if($i==$side)
        {

            // echo "<br>i:$i==side:$side ::::Break<br>";

            break;

        }

        if($fenu>0)

```

```
{ $sqlsub="SELECT 'tt_subject' FROM 'tb_timetable' WHERE 'tt_sem'='$semester'
AND 'dept_id'=(SELECT 'dept_id' FROM 'tb_department' WHERE 'dept_name'='$department')
and tt_date='$wwd'";
```

```
    $sqlsub=mysql_query($sqlsub);
```

```
    $ssub=mysql_fetch_row($sqlsub);
```

```
    if(empty($varsub))
```

```
    {
```

```
        $varsub[]=$ssub[0];
```

```
    }
```

```
    else
```

```
    {
```

```
        if(!in_array($ssub[0], $varsub))
```

```
        {
```

```
            $varsub[]=$ssub[0];
```

```
        }
```

```
    }
```

```
    if(empty($var))
```

```
    {
```

```
        $var[]=$department;
```

```
    }
```

```
    else
```

```
    {
```

```
        if(!in_array($department, $var))
```

```
        {
```

```

        $var[]=$department;

    }

}

$regno=$fe[0];

$i++;

$ins=mysql_query("insert into
tb_seat_arr(`nr_nominalroll_id`,`rh_room_no`,date,`time`)values('$regno','$rom','$wwd','$tim')");

}

else

{

    //echo "<br>fenu::$fenu>0else::::::Break<br>";

    break;

}

$f=1;

}

}

// echo "<br>i value : $i<br>";

if($f==1)

{

    // echo "<br>_____
_____
_____
_____<br>";

    //echo "<br>Enter $f<br>";

    if(empty($var))

```

```

{
    $var[]= array();
}

// print_r($var);

foreach ($spdepsem as $sds)
{
    $semdep= explode(":", $sds);

    $department=$semdep[0];

    $semester=$semdep[1];

    // echo "<br>Enter foreach $f loop::::$semdep[0]<br>";

    if(!in_array($department, $var))
    {
        /*echo "<br>Enter:$department<br>";

        echo "<br>Center:$center<br>";*/

        $sqlsub="SELECT 'tt_subject' FROM 'tb_timetable' WHERE 'tt_sem'='$semester'
AND 'dept_id'=(SELECT 'dept_id' FROM 'tb_department' WHERE 'dept_name'='$department')
and tt_date='$wwd'";

        $sqlsub=mysql_query($sqlsub);

        $ssub=mysql_fetch_row($sqlsub);

        if(!in_array($ssub[0], $varsub))
        {
            for ($l=0;$l<$center;$l++)
            {

                // echo "<br>Enter L:$l<br>";

```

```

        $a1=mysql_query("select * from tb_nominalroll where
nr_course='$department' and nr_sem='$semester' and nr_nominalroll_id not in (select
nr_nominalroll_id from tb_seat_arr where date='$wwd') limit 1 ");

```

```

        $fe1=mysql_fetch_row($a1);

```

```

        $fenu1=mysql_num_rows($a1);

```

```

        if($c==$center)

```

```

        {

```

```

            // echo "<br>i:$i==Center:$side ::::Break<br>";

```

```

            break;

```

```

        }

```

```

        if($fenu1>0)

```

```

        {

```

```

            $regno1=$fe1[0];

```

```

            $c++;

```

```

                                $ins1=mysql_query("insert into
tb_seat_arr(nr_nominalroll_id`,`rh_room_no`,date`,`time`)values('$regno1','$rom','$wwd','$tim')");

```

```

                                // echo "insert into
tb_seat_arr(nr_nominalroll_id`,`rh_room_no`,date`,`time`)values('$regno1','$rom','$wwd','$tim')";

```

```

        }

```

```

        else

```

```

        {

```

```

            //echo "<br>fenu::$fenu>0else::::Break<br>";

```

```

            break;

```

```

        }

```

```

        $f=0;

```

```

    }

    }

    }

}

unset($var);

unset($varsub);

}

// echo "<br>c value : $c<br>";

$tot=$i+$c;

// echo "<br>TOTAL:$tot<br>";

if($tot==$roomno)

{

    // echo "<br>TOT:::$tot==ROOMNO:::$roomno:::continue<br>";

    continue;

}

else

{

    //  echo "<br>_____
_____
_____
_____<br>";

    // echo "<br>TOT:::$tot==ROOMNO:::$roomno:::Else<br>";

    foreach ($spdepsem as $sds)

    {

```



```

$semdep= explode(".", $sds);

$department=$semdep[0];

$semester=$semdep[1];

/*echo "<br>Enter foreach $f loop::::$semdep[0]<br>";

echo "<br>Enter:$department<br>";

echo "<br>Center:$center<br>";*/

for ($l=0;$l<$center;$l++)

{

    // echo "<br>Enter L::::$l<br>";

    $a1=mysql_query("select * from tb_nominalroll where nr_course='$department' and
nr_sem='$semester' and nr_nominalroll_id not in (select nr_nominalroll_id from tb_seat_arr where
date='$wwd') limit 1 ");

    $fe1=mysql_fetch_row($a1);

    $fenu1=mysql_num_rows($a1);

    if($c==$center)

    {

        // echo "<br>Break<br>";

        break;

    }

    if($fenu1>0)

    {

        $regno1=$fe1[0];

        $c++;

```

```

        $ins1=mysql_query("insert into
tb_seat_arr(`nr_nominalroll_id`,`rh_room_no`,date,`time`)values('$regno1','$rom','$wwd','$tim')");

        // echo "insert into
tb_seat_arr(`nr_nominalroll_id`,`rh_room_no`,date,`time`)values('$regno1','$rom','$wwd','$tim')";

    }

    else

    {

        // echo "<br>fenu::$fenu>0else:::Break<br>";

        break;

    }

    $f=0;

}

}

}

}

```

```
/*_____END EXAMPLE CODE_____*/
```

/*————— Start Invigilation duties —————
—————*/

```
// echo “daf”;
```

```
$roomno="SELECT 'rh_room_no',COUNT(*) FROM 'tb_seat_arr' where date='$wwd' GROUP
BY 'rh room no'";
```

```
// echo "$roomno";
```

```
$roomn=mysql query($roomno);
```

```
while($rom=mysql_fetch_array($roomn))
```

```

{

// echo "dsf";

$numseat=round($rom[1]/30);

for($k=0;$k<$numseat;$k++)

{

    $facultyno="SELECT * FROM 'tb_faculty_register' WHERE 'f_email' NOT IN (SELECT
'f_id' FROM 'tb_invigilation')";

    $facuno=mysql_query($facultyno);

    $fnumber=mysql_num_rows($facuno);

    // echo $facultyno;

    // echo $fnumber;

    if($fnumber>0)

    {

        $facu="SELECT * FROM 'tb_faculty_register' where invstat=0 limit 1";

        $fac=mysql_query($facu);

        $fa=mysql_fetch_row($fac);

                                                $insinv="INSERT INTO
'tb_invigilation'('f_id','rh_room_no','date','ftime','ttime')VALUES('$fa[4]','$rom[0]','$swwd','$ftime','$ttime)";

        // echo "$insinv";

        $iniv=mysql_query($insinv);

        if($iniv>0)

        {

            $updatef="UPDATE 'tb_faculty_register' SET 'invstat'=1 WHERE 'f_email'='$fa[4]'";

            $upda=mysql_query($updatef);

```

```

    }

}

else

{

    for ($l=1;$l<100;$l++)

    {

        //$as=$l-1;

        $facu1="SELECT * FROM 'tb_faculty_register' where invstat=$l";

        //echo $facu1;

        $fac1=mysql_query($facu1);

        $fn1=mysql_num_rows($fac1);

        if($fn1>0)

        {

            if($tim=="2:00" && date('N', strtotime($wwd)) == 5)

            {

                // echo 'Enter afternoon';

                while ($fid=mysql_fetch_array($fac1))

                {

                    $fidsql=mysql_query("SELECT * FROM 'tb_invigilation' INNER JOIN

'tb_faculty_register' ON 'tb_invigilation`.`f_id`='tb_faculty_register`.`f_email' WHERE

'tb_invigilation`.`f_id`='$fid[4]' AND 'tb_faculty_register`.`invstat`='$l'");

                    $fidsq=mysql_fetch_row($fidsql);

                    $fdate=$fidsq['date'];

                    if(date('N', strtotime($fdate)) == 5)

```

```

        {
            continue;
        }
        else
        {
            $fidf=$fidsq[1];
        }
    }

    $insinv1="INSERT INTO
`tb_invigilation`(`f_id`,`rh_room_no`,`date`,`ftime`,`ttime`)VALUES('$fidf','$rom[0]','$wwd','$ftime','$ttime')";

    $iniv1=mysql_query($insinv1);

    if($iniv1>0)
    {
        $s=$l+1;

        $updatef="UPDATE `tb_faculty_register` SET `invstat`=$s WHERE
`f_email`='$fidf'";

        $upda=mysql_query($updatef);

        break;
    }
}
else
{
    $fidsql1=mysql_query("SELECT * FROM `tb_invigilation` INNER JOIN
`tb_faculty_register` ON `tb_invigilation`.`f_id`=`tb_faculty_register`.`f_email` WHERE
`tb_faculty_register`.`invstat`=$l");

```

```

        $fidsq1=mysql_fetch_row($fidsq1);

        $fidf1=$fidsq1[1];

        $insinv1="INSERT INTO
'tb_invigilation'('f_id','rh_room_no','date','ftime','ttime')VALUES('$fidf1','$rom[0]','$wwd','$ftime','$ttime)';

        $iniv1=mysql_query($insinv1);

        if($iniv1>0)

        {

            $s=$l+1;

            $updatef="UPDATE 'tb_faculty_register' SET 'invstat'=$s WHERE
'f_email'='$fidf1'";

            $upda=mysql_query($updatef);

            break;

        }

    }

}

else

{

    continue;

}

}

}

}

}

}

```

```

/*————— End Invigilation duties —————
—————*/

$timetable="UPDATE 'tb_timetable' SET 'tt_status'='complete' WHERE 'tt_date'='$wwd'";

$timetab=mysql_query($timetable);

if($timetab>0)

{

    echo "<script>alert('Allocate successfully');window.location.href='seatarrangement.php'</
script>";

}

else

{

    echo "<script>alert('Error');window.location.href='seatarrangement.php'</script>";

}

}

else

{

    echo "<script>alert('No nominal roll added');window.location.href='seatarrangement.php'</
script>";

}

?>

<?php

include("footer.php");

?>

```

SYSTEM TESTING

6. SYSYTEM TESTING

System testing is a critical element of software quality assurance and represents the ultimate reviews of specification, design, and coding. Testing present an interesting anomaly for the software. Testing is vital to the success of the system. Error can be injected at any stage during development. The system testing makes logical assumptions that if all the parts of the system are correct, the goal will be successfully achieved. During testing the program to be tested is executed with set of data and output of the program for test is evaluated to determine if the program is performing as expected. A series of testing are performed for the proposed system before the system is ready for user acceptance testing. The testing steps are:

- Unit testing
- Integration testing
- Acceptance testing
- System testing

UNIT TESTING

Unit testing is a software development process that involves sychronized application of a board spectrum of defect prevention. Unit testing focuses verification effort on the smallest unit of the software design, this module is known as Module Testing. Since the proposed system has 3 modules the testing is individually performed on the each module. This type of tests are usually done by developers as they work on code to ensure that the specific function is working as expected. One function may have multiple tests, to catch corner cases or other branches in the code. Unit testing along cannot verify the functionality of a piece of software, but rather is used to ensure that the building blocks of the software work independently from each other.

Using the details design description as a guide important control paths are tested to uncover errors within the boundary of the module. In this testing step the admin,teacher,student modules are found to be working satisfactory as regards to the expected output from the module.

INTEGRATION TESTING

Integration testing is a type of software testing that seeks to verify the interface between components against a software design. Software components may be integrated in an iterative

way or all together. Normally the former is considered as a better practice since it allows interface issues to be located more quickly and fixed. Data can be tested across an interface; one module can have adverse effect on another sub function when combined may not produced the desired function. Integration testing is a systematic technique for constructing the program structure while at the same time conducting test to uncover errors associated within interface. Integration testing works to expose defects in the interfaces between the integrated components. Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system. When more than one set of codes are ready they are integrated and tested for the desired output by entering input data.

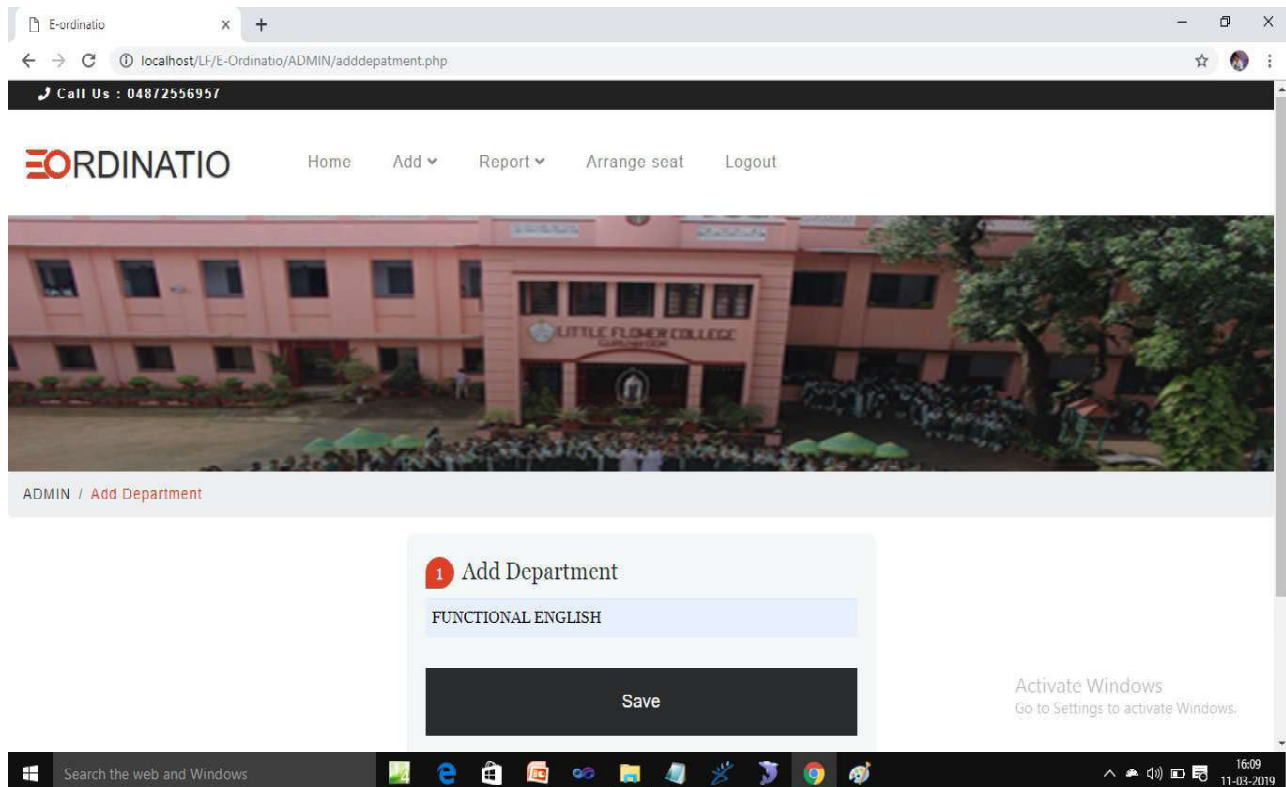
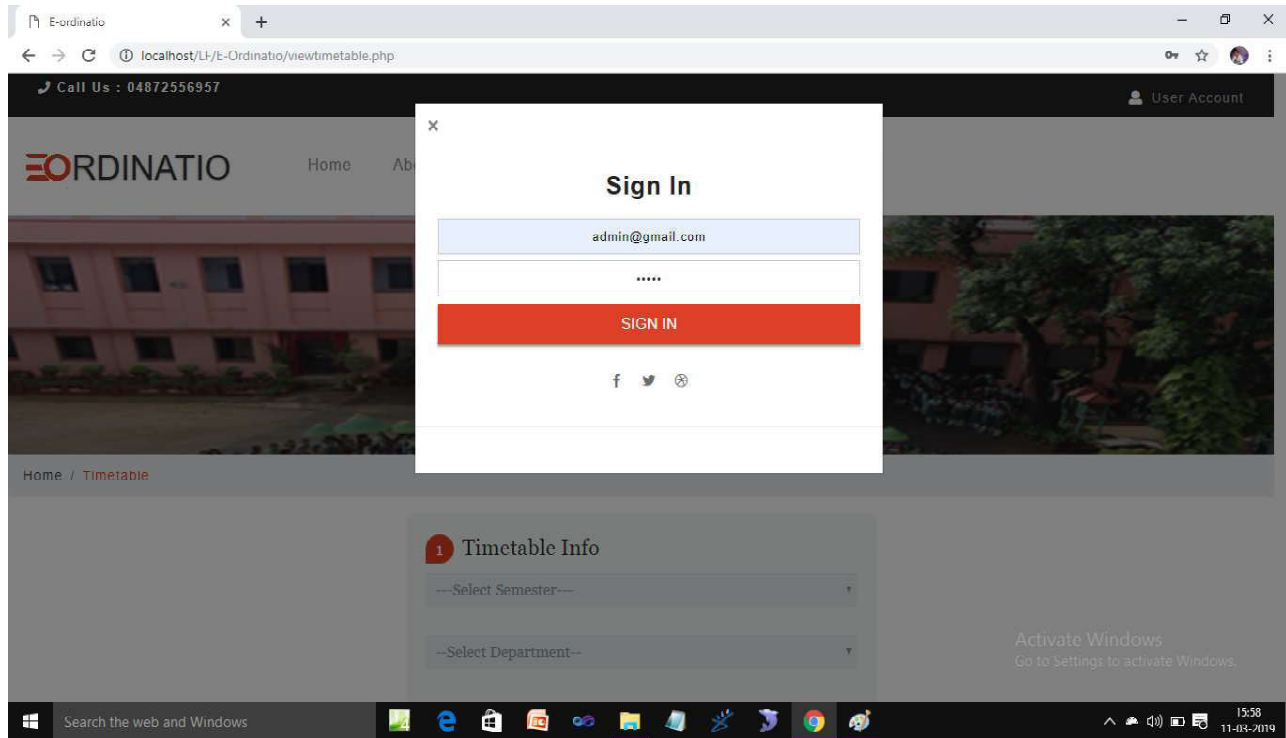
ACCEPTANCE TESTING

Acceptance is used to conduct(pre-release) of a product, service or system as part of a quality management system. This is a common type of non-functional software testing, used mainly in software development and software maintenance project. This focuses on the pre-release of the system to be supported, and/or to become part of the product environment. Acceptance testing is formal testing based on user requirements. It is conducted to determine whether or not the system satisfies the acceptance criteria and enable the user, to determine whether or not to accept the system. While acceptance testing, a group of colleagues will test the system in our developing site. Then we ported the project to other site that is client site to ensure the portability of the system.

SYSTEM TESTING

System Testing or end to end testing, tests a completely integrated system to verify that it meets its requirements. For examole a system test might be involve testing a logon interface, then creating and editing an entry, plus sending or printing results, followed by summery processing or deletion (or archiving) of entries, then logoff. After the entire modules are completely coded they are integrated and tested for the final result.

SCREENSHOTS



E-ordinatio x +

localhost/LF/E-Ordinatio/ADMIN/nominalroll.php

ADMIN / Add Nominal roll

1 Nominal Roll Info

LFAQSCS003

LF

Christin Mariya

2019

Computer Science

VI Sem

Save

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

16:13
11-03-2019

E-ordinatio x +

localhost/LF/E-Ordinatio/ADMIN/addfaculty.php

1 Faculty Info

Hitha Paulson

9846387652

MSc

hitha.paulson@gmail.com

Department:

Computer Science

2 Login Info

hitha.paulson@gmail.com

Save

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

16:11
11-03-2019

E-ordinatio x +

localhost/LF/E-Ordinatio/TEACHER/addstudentdetails.php

1 Student Info

Christin Mariya

VI Sem

9497412201

LFAQSCS003

mariyachristin@gmail.com

Department:

Computer Science

2 Login Info

mariyachristin@gmail.com

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

16:22
11-03-2019

E-ordinatio x +

localhost/LF/E-Ordinatio/ADMIN/timetable.php

ADMIN / Add Timetable

1 Add Timetable

25-03-2019

01:30

BCS6B12 Android Programming

Semester:

VI Sem

Department:

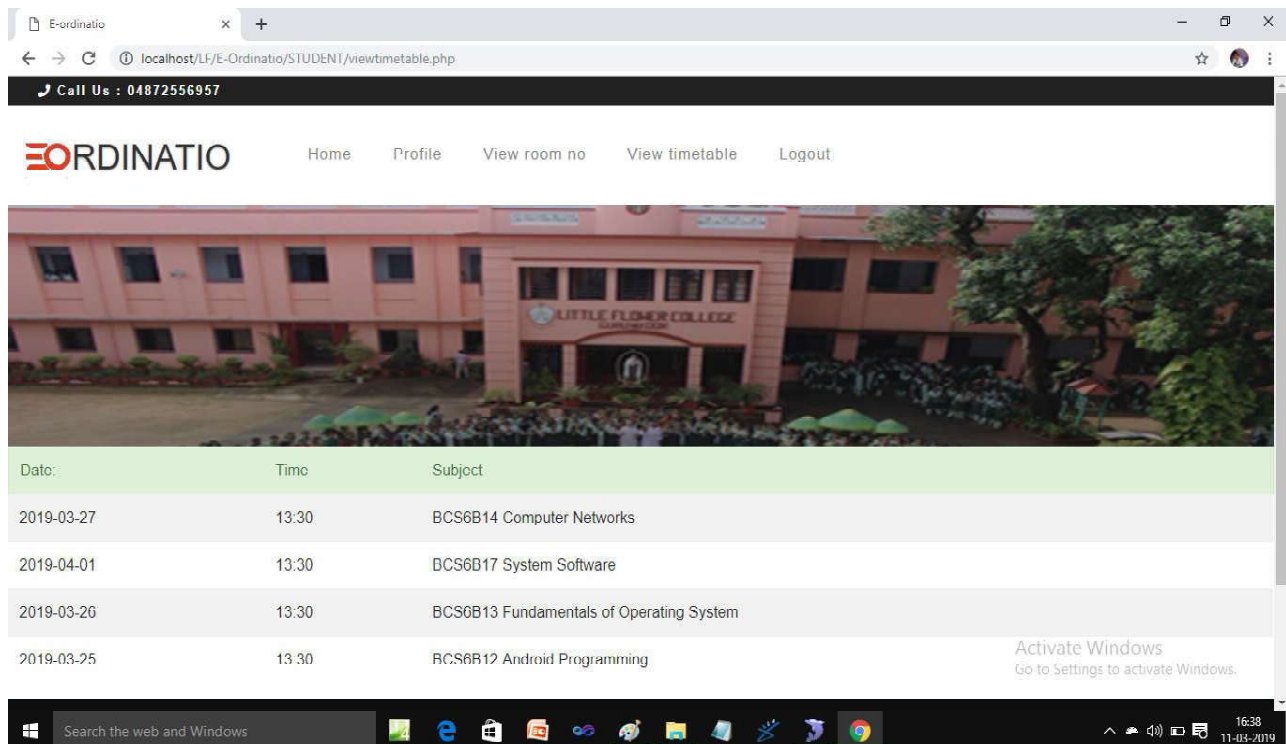
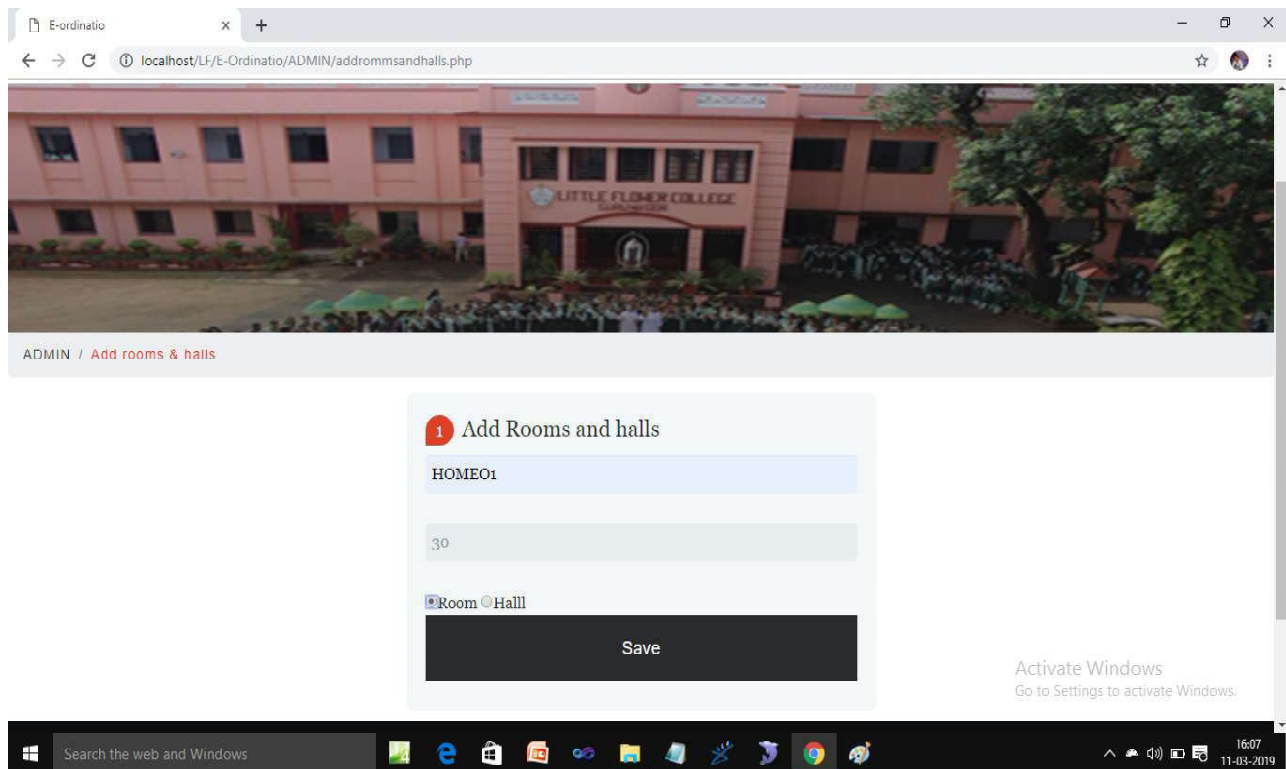
Computer Science

Apply

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

16:15
11-03-2019



Teacher / Exam Attendance

Attendance

Subject: BCS6B14 Computer Networks Year: 2019

Attendance List

Hall Ticket	Student Name	Attendance Status
LFAQSCS001	Arathy Gopakumar	0
LFAQSCS002	Badhira P B	0
LFAQSCS003	Christin Mariya	0
LFAQSCS004	Dhanya M	0
LFAQSS005	Greeshma A R	0
LFAQSCS06	Greeshma Gopinathan	0
LFAQSCS007	Haritha V	0

Activate Windows
Go to Settings to activate Windows.

Call Us : 04872556957

EORDINATIO Home Add Report Arrange seat Logout

Date Name Current Day

Don C Josson

INVIGILATION DUTY REPORT

Day	Hall	Staff	AN/FN	Department
2019-03-25	126	Don C Josson	FN	BCA
2019-03-28	126	Don C Josson	FN	BCA
2019-03-29	126	Don C Josson	FN	BCA


Activate Windows
Go to Settings to activate Windows.

E-ordinatio x +

localhost/LF/E-Ordinatio/STUDENT/viewroomstudent.php

Call Us : 04872556957

EORDINATIO Home Profile View room no View timetable Logout



Date:	Class Room	From Time
2019-03-25	121	2:00
2019-03-26	121	2:00
2019-03-27	121	2:00
2019-04-01	121	2:00
2019-03-05	02	2:00

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

16:42 11-03-2019

E-ordinatio x +

localhost/LF/E-Ordinatio/ADMIN/report_invigilation.php

Date Name Current Day

From: 18-03-2019 To: 02-04-2019

INVIGILATION DUTY REPORT

Name	Department	Count
Annam Sini A P	Malayalam	1
Bindhu V K	Malayalam	1
Sr Dani Maria	English	1
Don C Josson	BCA	3
Neenu Elizabeth	BCA	1
Jeena George	BCA	1
Jestin James	Computer Science	1
Joly Raphy	Malayalam	1

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

16:16 11-03-2019

E-ordinatio

localhost/LF/E-Ordinatio/TEACHER/Semincrement.php

LFAQSCS024	Lubna P H	2019	VI Sem	Computer Science
LFAQSCS025	Mifitha Ummer	2019	VI Sem	Computer Science
LFAQSCS026	Nargees Najeeb	2019	VI Sem	Computer Science
LFAQSCS027	Nithyapriya	2019	VI Sem	Computer Science
LFAQSCS029	Niya	2019	VI Sem	Computer Science
LFAQSCS030	Shameema	2019	VI Sem	Computer Science
LFAQSCS031	Shemy	2019	VI Sem	Computer Science
LFAQSCS033	Sreelakshmi P S	2019	VI Sem	Computer Science

Change Sem

Activate Windows
Go to Settings to activate Windows.

localhost/LF/E-Ordinatio/TEACHER/changeSemester.php?sem=VI Sem&dep=Computer Science

Search the web and Windows

16:28 11-03-2019

E-ordinatio

localhost/LF/E-Ordinatio/TEACHER/viewstudent.php

Teacher / Student List

Students List

Computer Science VI Sem

Hall ticket no	Name	Email	Department	Semester	Action
LFAQSCS001	Arathy Gopakumar	aswathygopakumarc@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS002	Badhira P B	badhurapb487@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS003	Christin Mariya	mariyachristin@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS004	Dhanya M	dhanyadhanus123@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS005	Greeshma A R	greeshmaar585@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS006	Greeshma Gopinathan	greeshmagopinath@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS007	Haritha V	harithav007@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS008	Nikhila K S	nikhilasreedhar26@gmail.com	Computer Science	VI Sem	Remove
LFAQSCS009	Sadhiya	sadhiya127@gmail.com	Computer Science	VI Sem	Remove

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

16:24 11-03-2019

SYSTEM IMPLEMENTATION

SYSTEM IMPLEMENTATION

Implementation is the stage of project when the theoretical design is turned into working system. If the implementation stage is not properly planned and controlled, it can cause chaos. Thus it can be considered to be the most crucial stage in achieving a successful new system and in giving the users the confidence that the new system will work and be effective.

Normally this stage involves setting up a steering committee, which will act as a sounding board for ideas, complaints and programs. The first task is implementation planning; ie, deciding of the methods and timescale to be adopted. Apart from planning, the two major tasks of preparing for implementation are education and training of users and testing the system. Education of users should really have taken place much earlier in the project; at the implementation stage the emphasis must be on training in new skills to give staff confidence they can use system. Once staff has been trained, the system can be tested.

After the implementation phase is completed and the user staff is adjusted to the changes created by the new system, evaluation and maintenance begin. The importance of maintenance is to continue to bring the new system to standard.

The activities of the implementation phase can be summarized as:-

- Implementation planning
- Education and training
- System training

We have implemented the website for our college. We conducted training to the teaching and nonteaching staffs of the college. The software was easy to understand by them. Some of the selected students also attended the training programme. The class was beneficial to both users and staffs.

CONCLUSION AND FUTURE ENHANCEMENT

9. CONCLUSION AND FUTURE ENHANCEMENT

The software developed by us has enabled in reducing various drawbacks faced by the existing system. The newly developed system is easy to understand, use and more accessible to the users. It should work smoothly and very fast and is also saving time and manpower. If anyone having computer knowledge can handle it easily. It also offers the following benefits:

- * Paper work reduction
- * Improved data quality
- * Error prevention
- * Easily generates reports

The software is implemented and tested with real data were found to be error free. Also the system is protected from any unauthorized access. All the necessary validations are carried out on this project, so that any kind of users make use of this software.

In future, we can analyze the results of all the students who have written exams in our college. The total percentage of pass and failed can be calculated through online. The system can be enhanced with the capabilities of department automation too. This can be included with other modules like internal marks, attendance, office automation.

BIBLIOGRAPHY AND REFERENCES

10. BIBILOGRAPHY

BOOKS

- System Analysis and design- Elias M Awet
- Software Engineering principles and practices- Remit Khoharanas
- Database Management System- Micheal H Crawford, John P Dimarcav

WEBSITES

- www.w3school.com
- www.wordplay.com