

INTRODUCTION

1.1 Purpose :

Normally timetable generation done manually. As we know all institutions/organizations have its own timetable, managing and maintaining these will not be difficult. Also many colleges and institutes changes there timetable more than twice a year which puts a double work load to the timetable designer. Considering workload with this scheduling will make it more complex. As mentioned, when Timetable generation is being done, it should consider the maximum and minimum workload that is in a college. In that case, timetable generation will become more complex. In addition, it is a time consuming process.

1.2 Scope :

Automated Timetable Generation system generates timetable for each class and teacher, in keeping with the availability calendar of teachers, availability and capacity of physical resources (such as classrooms, computer laboratories) and rules applicable at different classes, semesters, teachers and subjects level. Best of all, this automated timetable generation system tremendously improves resources utilization and optimization.

1.3 Problem :

The main problem of preparing timetable manually is that the constraint satisfaction is very complex to solve and it takes many attempts to have an avg. satisfaction. This is a very time consuming and uninteresting tasks to set values of a timetable, manually. We have prepared a solution of that time consuming problem is that this system uses Constraint satisfaction, Ant colony optimization & Genetic algorithms for solving of the specified problem.

1.4 Proposed System :

Our system, which is abbreviated as ATGS, is working on AI technology, it only have one user, which is admin, and the admin can register another admin this system works only on local server so no other user can access the database from other system. The response time for generation of timetable by satisfying all the constraints is approx. 2min 50sec; it uses allocation of three faculties for practical classes if one is busy then the other can engage the class which is very helpful for that institutes which have less no. of teachers. In this system, a user can save pdf or print the timetable instantly after generation.

Chapter 2

LITERATURE SURVEY

2.1.1 PHP :

PHP is a script language and interpreter that is freely available and used primarily on Linux Web servers. PHP originally derived from Personal Home Page Tools, now stands for PHP: Hypertext Preprocessor, which the PHP FAQ describes as a "recursive acronym."

PHP executes on the server, while a comparable alternative, JavaScript, executes on the client. PHP is an alternative to Microsoft's Active Server Page (ASP) technology. As with ASP, the PHP script is embedded within a Web page along with its HTML. Before the page is sent to a user that has requested it, the Web server calls PHP to interpret and perform the operations called for in the PHP script.

An HTML page that includes a PHP script is typically given a file name suffix of “. Php” “. Php7” or “.dhtml”. Like ASP, PHP can be thought of as "dynamic HTML pages," since content will vary based on the results of interpreting the script. PHP is free and offered under an open source license.

2.1.2 MY SQL :

MySQL is an Oracle-backed open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing.

MySQL is an important component of an open source enterprise stack called LAMP. LAMP is a web development platform that uses Linux as the operating system, Apache as the web server, and MySQL as the relational database management system and PHP as the server side scripting language.

Originally conceived by the Swedish company MySQL AB, MySQL was acquired by Sun Microsystems in 2008 and then by Oracle when it bought Sun in 2010. Developers can use

MySQL under the GNU General Public License (GPL), but enterprises must obtain a commercial license from Oracle.

2.1.3 Apache :

Apache is a freely available Web server that is distributed under an "open source" license. Version 2.0 runs on most UNIX-based operating systems (such as Linux, Solaris, Digital UNIX, and AIX), on other UNIX/POSIX-derived systems (such as Rhapsody, BeOS, and BS2000/OSD), on Amigos, and on Windows 2000. According to a Net craft (www.netcraft.com) Web server survey 60% of all Web sites on the Internet are using Apache (62% including Apache derivatives), making Apache more widely used than all other Web servers combined. Apache is the most widely used web server software. Developed and maintained by Apache Software Foundation, Apache is an open source software available free.

2.2 (A) Software Development Life Cycle :

SDLC or the Software Development Life Cycle is a process that produces software with the highest quality and lowest cost in the shortest time. SDLC includes a detailed plan for how to develop, alter, maintain, and replace a software system.

SDLC involves several distinct stages, including planning, design, building, testing, and deployment. Popular SDLC models are Incremental, spiral, prototyping, and agile

How SDLC work :

SDLC works by lowering the cost of software development while simultaneously improving quality and shortening production time. SDLC achieves these apparently divergent goals by following a plan that removes the typical pitfalls to software development projects. That plan starts by evaluating existing systems for deficiencies. Next, it defines the requirements of the new system. It then creates the software through the stages of design, development, testing, and deployment. By anticipating costly mistakes like failing to ask the end user for suggestions, SLDC can eliminate redundant rework and after-the-fact fixes.

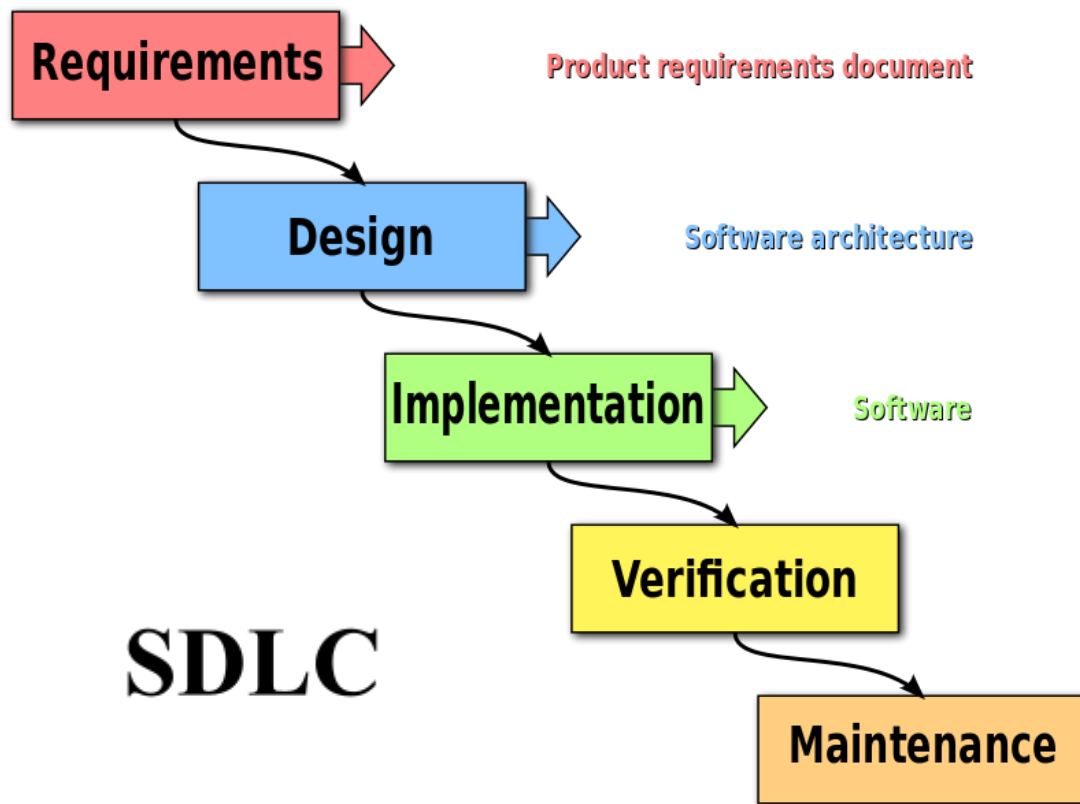


Fig. Software development life cycle

2.2 (B) Prototyping Model :

The Prototyping Model is a systems development method (SDM) in which a prototype (an early approximation of a final system or product) is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed. This model works best in scenarios where not all of the project requirements are known in detail ahead of time. An iterative, trial-and-error process takes place between the developers and the users.

There are several steps in the Prototyping Model :

1. The new system requirements are defined in as much detail as possible. This usually involves interviewing a number of users representing all the departments or aspects of the existing system.
2. A preliminary design is created for the new system.
3. A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
4. The users thoroughly evaluate the first prototype, noting its strengths and weaknesses, what needs to be added, and what should to be removed. The developer collects and analyzes the remarks from the users.

5. The first prototype is modified, based on the comments supplied by the users, and a second prototype of the new system is constructed.
 6. The second prototype is evaluated in the same manner, as was the first prototype.
 7. The preceding steps are iterated as many times as necessary, until the users are satisfied that the prototype represents the final product desired.
8. The final system is constructed, based on the final prototype.

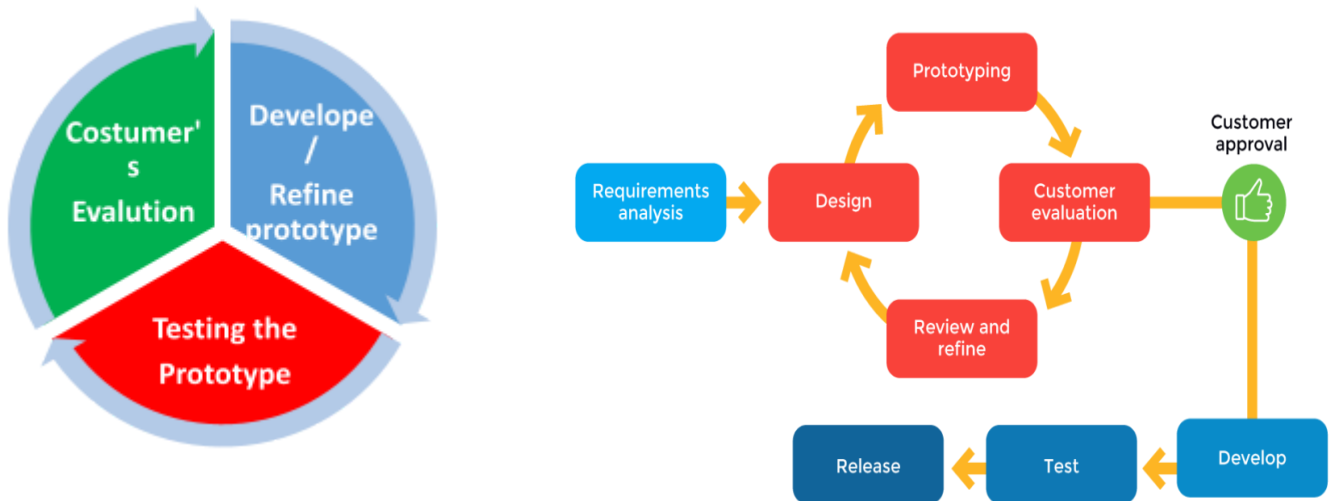


Fig. Prototyping Model

2.3 Forms & Entities :

1. Login Form

- a. User Name
- b. Password

2. Admin Registration

- a. Name
- b. User Name
- c. Password
- d. Conform Password
- e. Mobile No.
- f. Email ID
- g. Designation

3. Add Subject

- a. Subject code
- b. Title
- c. Course type
- d. Semester
- e. Department

4. Add Classroom

- a. Name
- b. Status

5. Add Teacher

- a. Faculty name
- b. Name
- c. Alias
- d. Designation
- e. Contact No.
- f. Email Id
- g. Qualification
- h. Experience

6. Allocation

- a. Theory Course
- b. Classroom Allocation
- c. Practical courses

Chapter 3

ANALYSIS

3.1. SYSTEM REQUIREMENTS

3.1.1 Functional Requirements

The major functional requirements of the system are as follows.

- a) To classify the data on a single site.
- b) To implement an User interface on the system.
- c) User-friendly front-end design using Cascading Style Sheets.
- d) Strong authentication while performing various operations.
- e) Java script validations and alerts wherever needed.

3.1.2 Non Functional Requirements

The major nonfunctional Requirements of the system are as follows

1. Secure access of confidential data (user's details). SSL can be used.
2. Better component design to get better performance at peak time
3. Flexible service based architecture will be highly desirable for future extension.

Constraints :

Constraints are limitations that are outside the control of the project team and need to be managed around. They are not necessarily problems. However, the project manager should be aware of constraints because they represent limitations that the project must execute within. Date constraints, for instance, imply that certain events (perhaps the end of the project) must occur by certain dates.

Resources are usually a constraint, since they are not available in an unlimited supply. 2.4.3.1. Hard Constraints: A timetable, which breaks a hard constraint, is not a feasible solution, and must be repaired or rejected by the timetabling algorithm. Hard constraints include "First Order Conflicts" 1. A lecturer cannot teach more than one class at the same time. 2. To Generate the Timetable based on the no of periods and time schedule.

3.2 Project Requirements :

- **Security of System Software.** - Privacy between the user accounts.
- **Existing & Proposed Systems.** — College management system, School Scheduling System, staff scheduling
- **Problem definition/ Problem Solution of the System:** -The Main Problem that we can identify is that the when Authority generates timetable manually he faces the faculty

class slashes by this he/she performs many permutations and combinations this project is used to minimize such issues.

- **Economic, Technical, Behavioral Feasibility.** -Helpful for organizations, Reduce time and effort, work as a tool.
- **Cost within the time, delivery time specification.** -By estimation,31st March
- **Life Cycle Model** – Prior to Prototyping Model
- **Programming language**-(My SQL, PHP)
- **Technology**-PHP application.
- **web technologies** -Html, CSS, JavaScript, PHP
- **Design tools** -Notepad++
- **Data base specification** – MySQL
- **Web browser** – Any(Google chrome suggested)
- **Timetable Algorithms & logic** – Generic algorithm, Ant colony optimization algorithm, Constraint satisfaction algorithm.

3.3 Software Requirements: (minimum)

- Windows 7
- MYSQL 5.7
- Easy PHP Deserver 14.1VC9

3.4 Hardware Components: (minimum)

- Processor – Dual Core
- Hard Disk – 50 GB
- Memory – 1GB RAM
- Printer(as required)

3.5 Data Requirements:

- Admin Details
- Faculty Details
- Subject Details
- Branch Details
- Timetable Layouts

3.6 Feasibility Study:

Feasibility is defined as the practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software.

Information such as resource availability, cost estimation for software development, benefits of the software to the organization after it is developed and cost to be incurred on its maintenance are considered during the feasibility study.

Types of Feasibility:

1. Technical Feasibility:

This is concerned with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may vary considerably, but might include -:

- The facility to produce outputs in a given time.
- Response time under certain conditions.
- Ability to process a certain volume of transaction at a particular speed.
- Facility to communicate data to distant locations.

In this, system our plan is successfully execute towards the comfort between the system and software requirements.

2. Operational Feasibility:

This is mainly related to human organizational and political aspects. The points to be considered are:

- What changes will be brought with the system?
- What organizational structure are disturbed?
- What new skills will be required? Do the existing staff members have these skills? If not, can they be trained in due course of time?

Our system runs on local host so no new skills will be required. Moreover, the user is perfectly trained to operate the system.

3. Economic Feasibility:

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as Cost / Benefit analysis, the procedure is to determine the benefits and savings that are expected from a proposed system and compare them with costs. If benefits outweigh costs, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have a chance of being approved.

This system is will completely proposed under the expectation of cost expenses as defined in the estimation.

Advantages:

- Admin did not need to worry for faculty time clashes.

- Authority now does not need to perform permutation and combination
- Authority can concentrate on other things rather than wasting their time on preparing Time-Table
- Manual work should be minimized.
- No Internet connection required.
- Admin can generate many combinations in less time.
- Teacher can see all its classes of the week

Disadvantage:

- Some slots can remain empty.
- User has to format it a bit after it is prepared.
- It should be only accessed by single system.
- Time taken to generate the timetable is 4 min 30 sec approx.

Chapter 4

DESIGNING

4.1 Software design:

It is the process of conceptualizing the software requirements into software implementation. This is the initial phase within the SDLC shifting the concentration from the problem to the solution.

When conceptualizing the software, the design process establishes a plan that takes the user requirements as challenges and works to identify optimum solutions. The plan should determine the best possible design for implementing the intended solution.

Major artifacts of the software design process include:

- **Software requirements specification:**

This document describes the expected behavior of the system in the form of functional and non-functional requirements. These requirements should be clear, actionable, measurable, and traceable to business requirements. Requirements should also define how the software should interact with humans, hardware, and other systems.

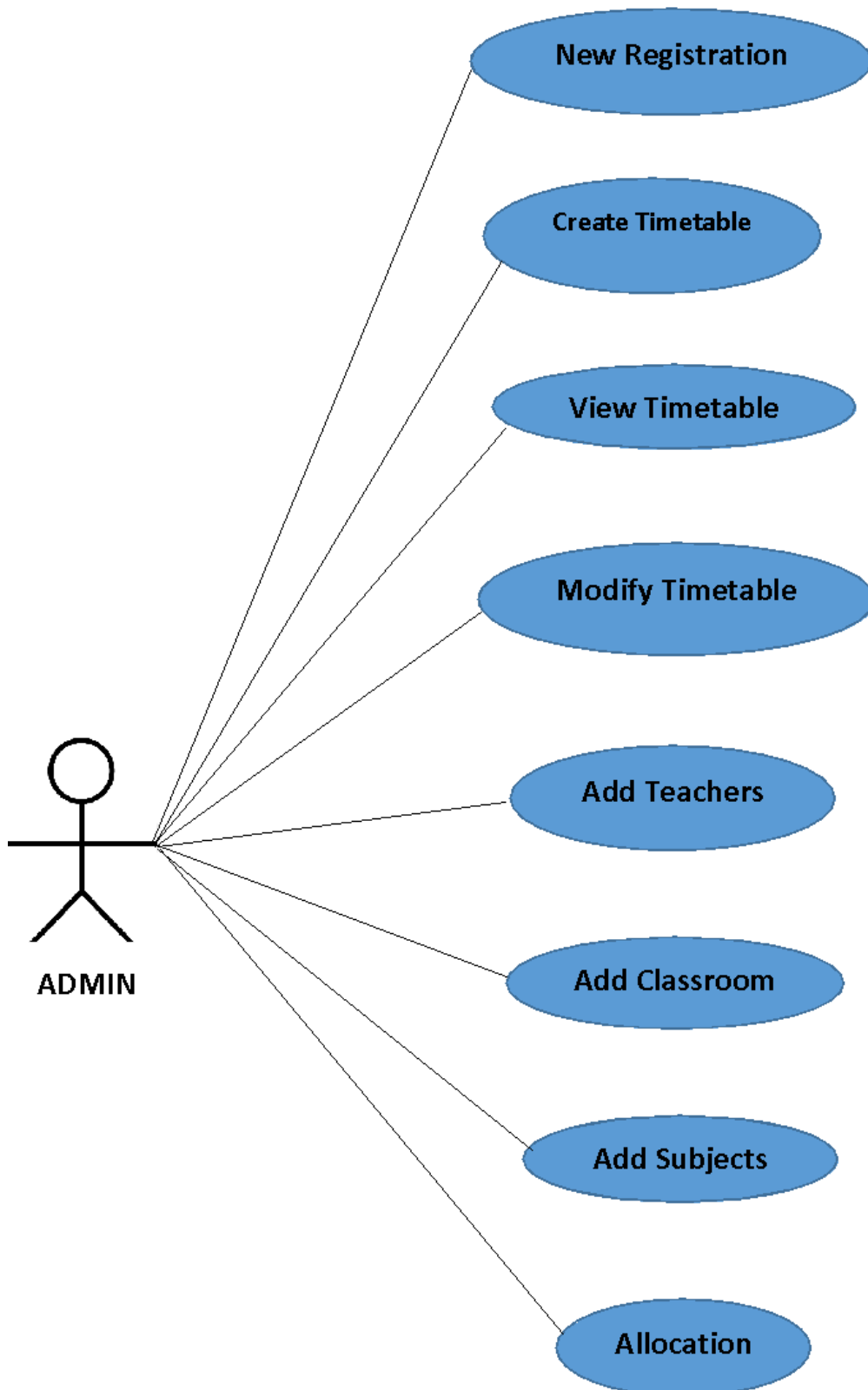
- **High-level design:**

The high-level design breaks the system's architectural design into a less-abstracted view of sub-systems and modules and depicts their interaction with each other. This high-level design perspective focuses on how the system, along with all its components, implements in the form of modules. It recognizes the modular structure of each sub-system and their interaction among one another.

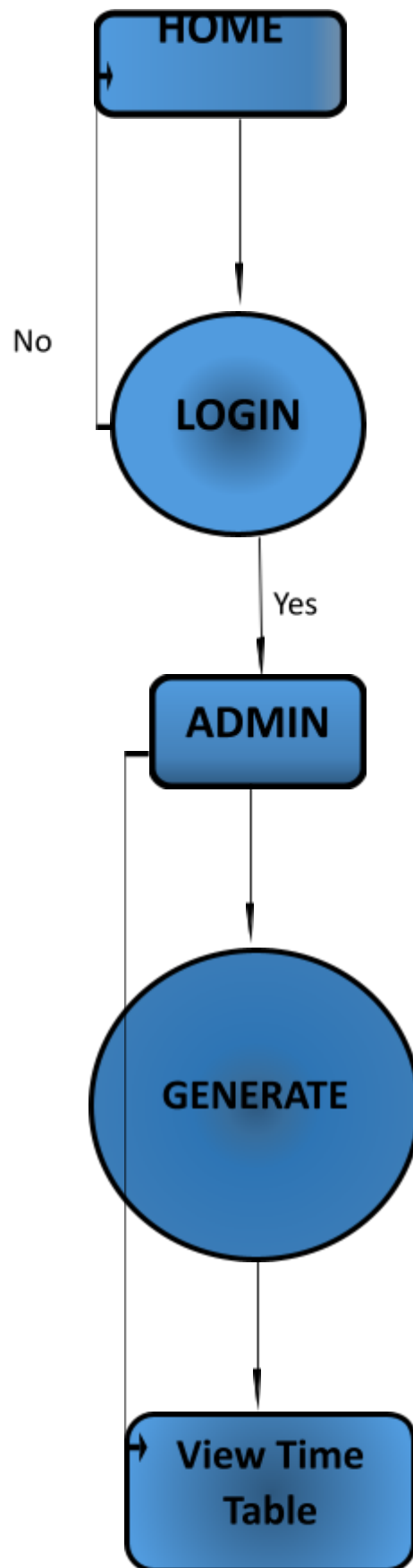
- **Detailed design:**

Detailed design involves the implementation of what is visible as a system and its sub-systems in a high-level design. This activity is more detailed towards modules and their implementations. It defines a logical structure of each module and their interfaces to communicate with other modules.

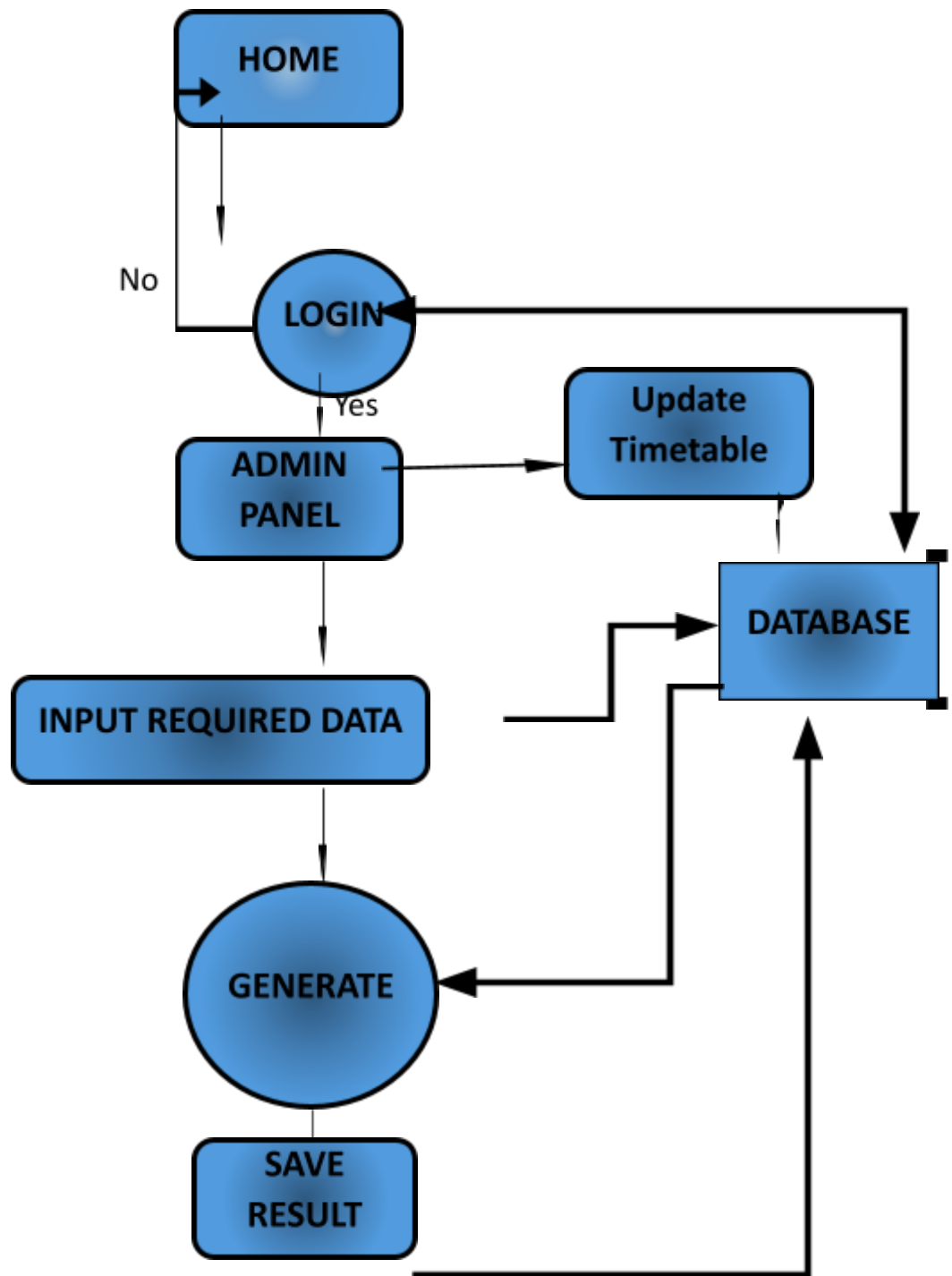
4.2 Use-case diagram



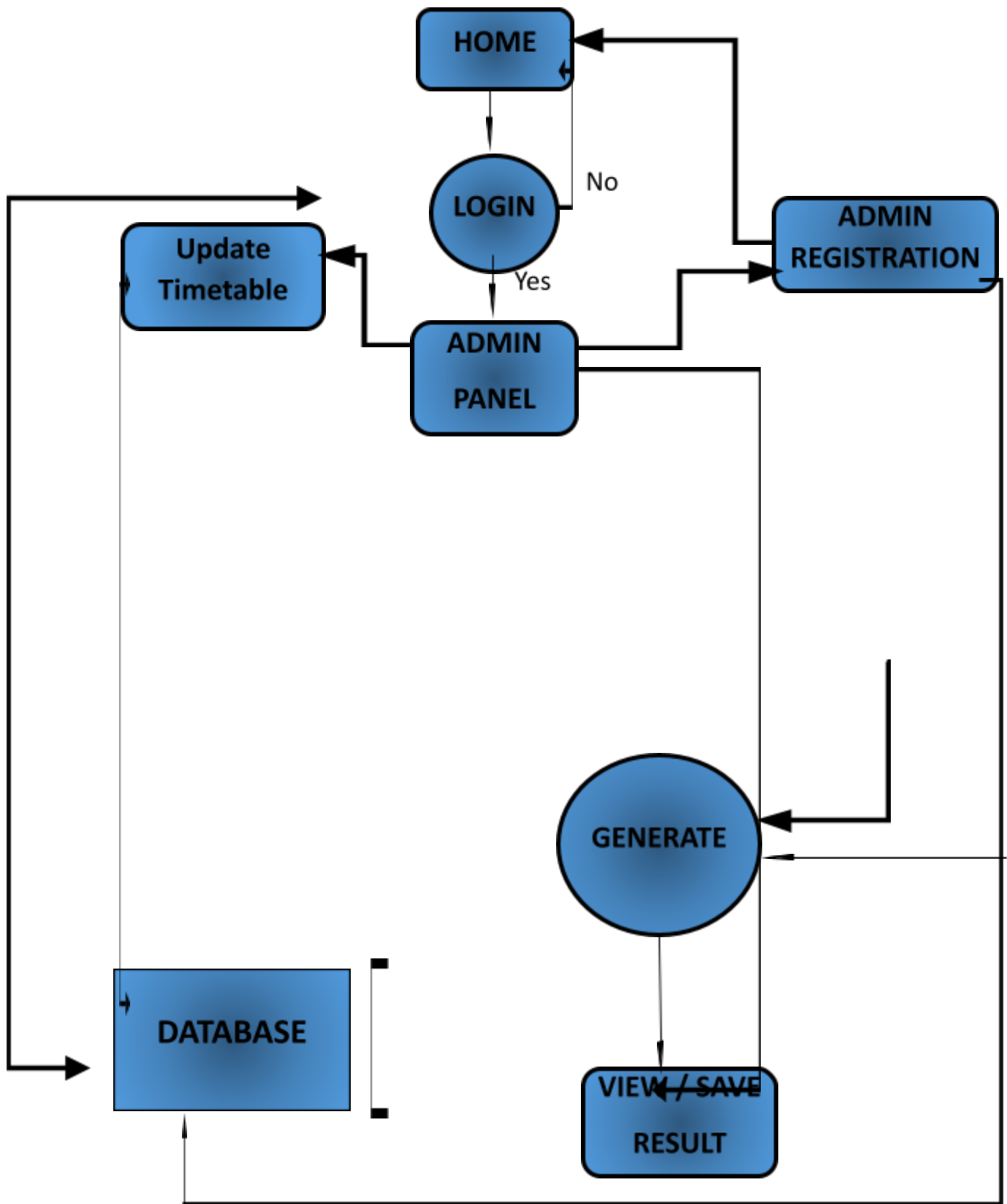
4.3.1 Data flow Diagram (Level 0)



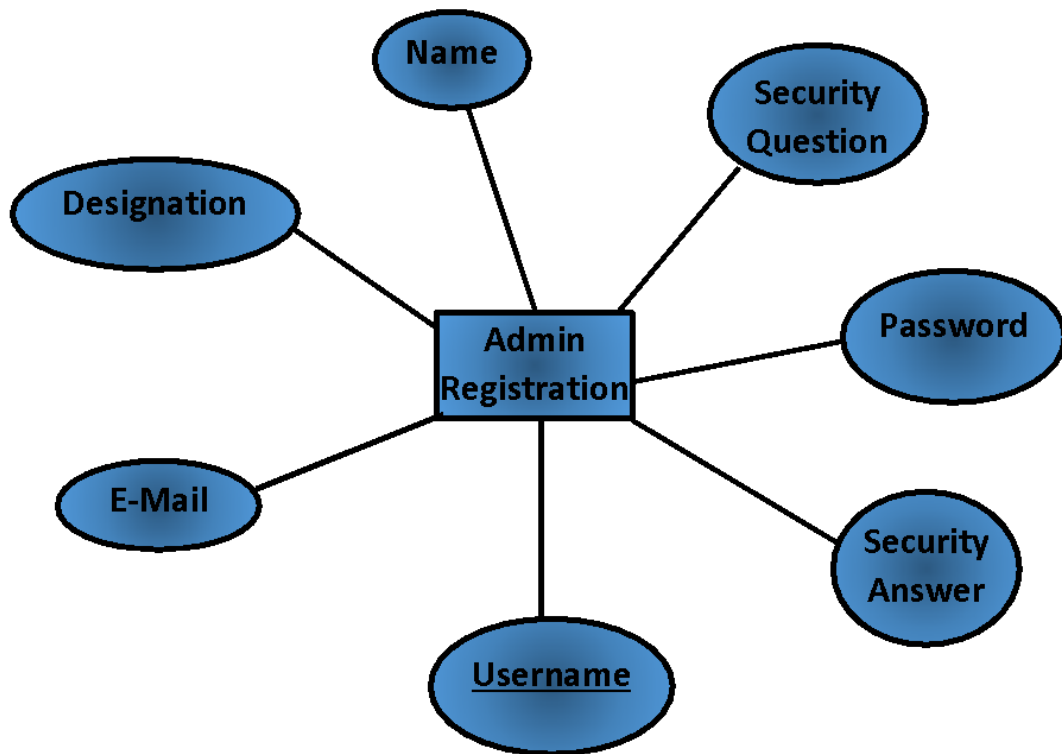
4.3.2 Data flow Diagram (Level 1)



4.3.3 Data flow Diagram (Level 2)

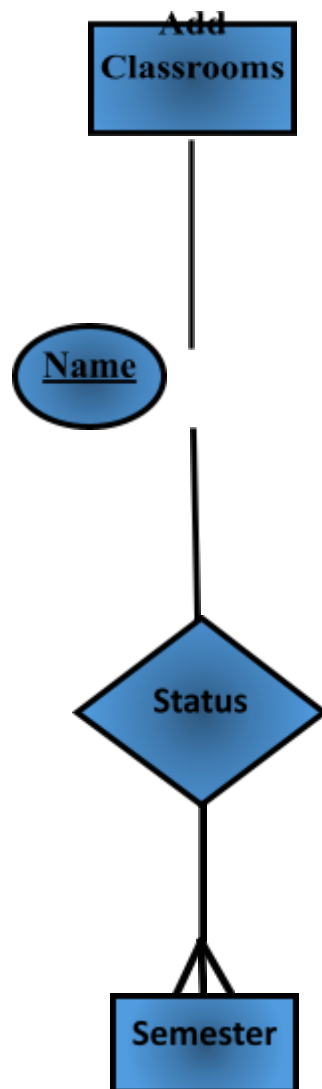


4.4.1 E-R DIAGRAM
(1) Admin Registration



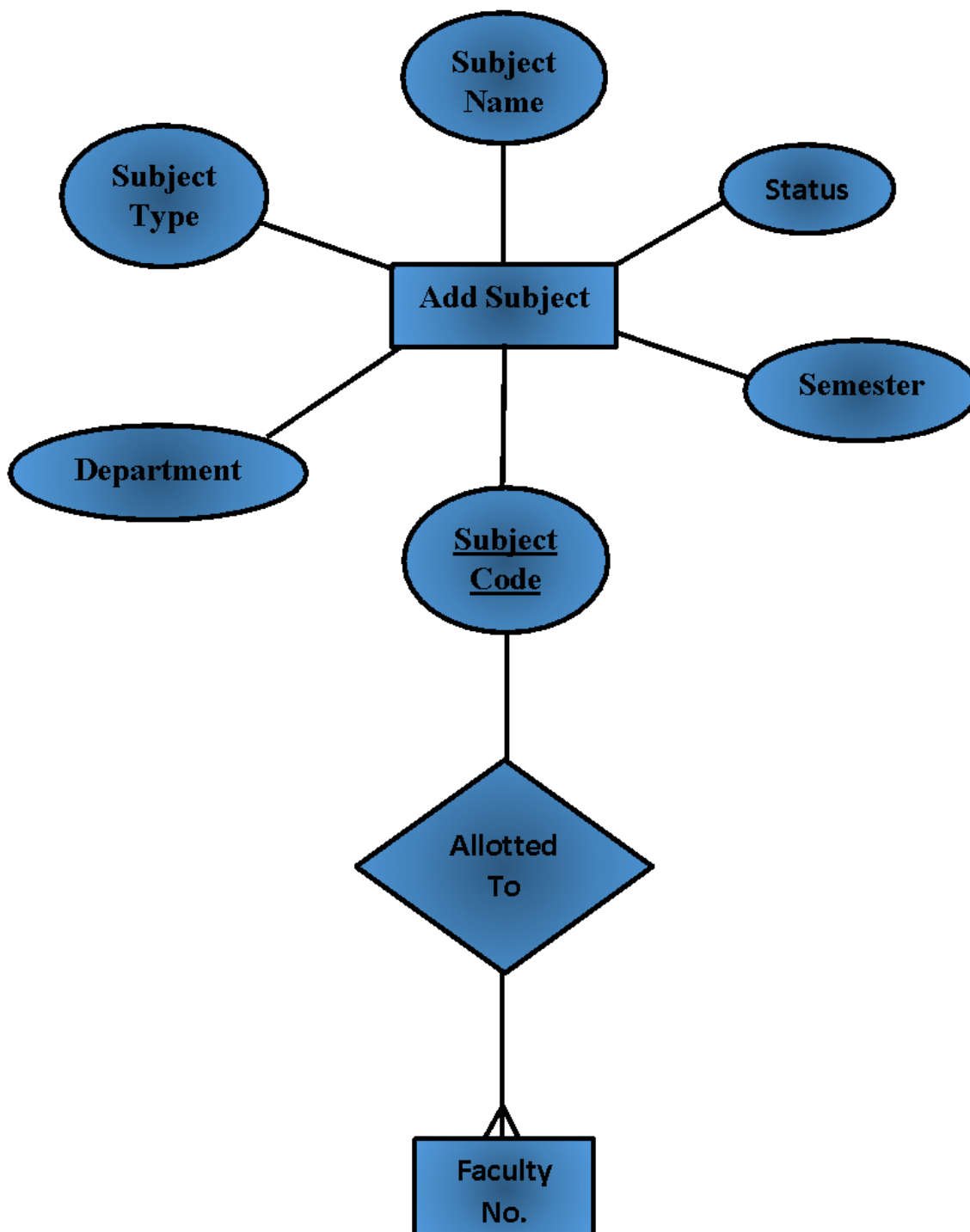
4.4.2 E-R DIAGRAM

(2) Add Classrooms



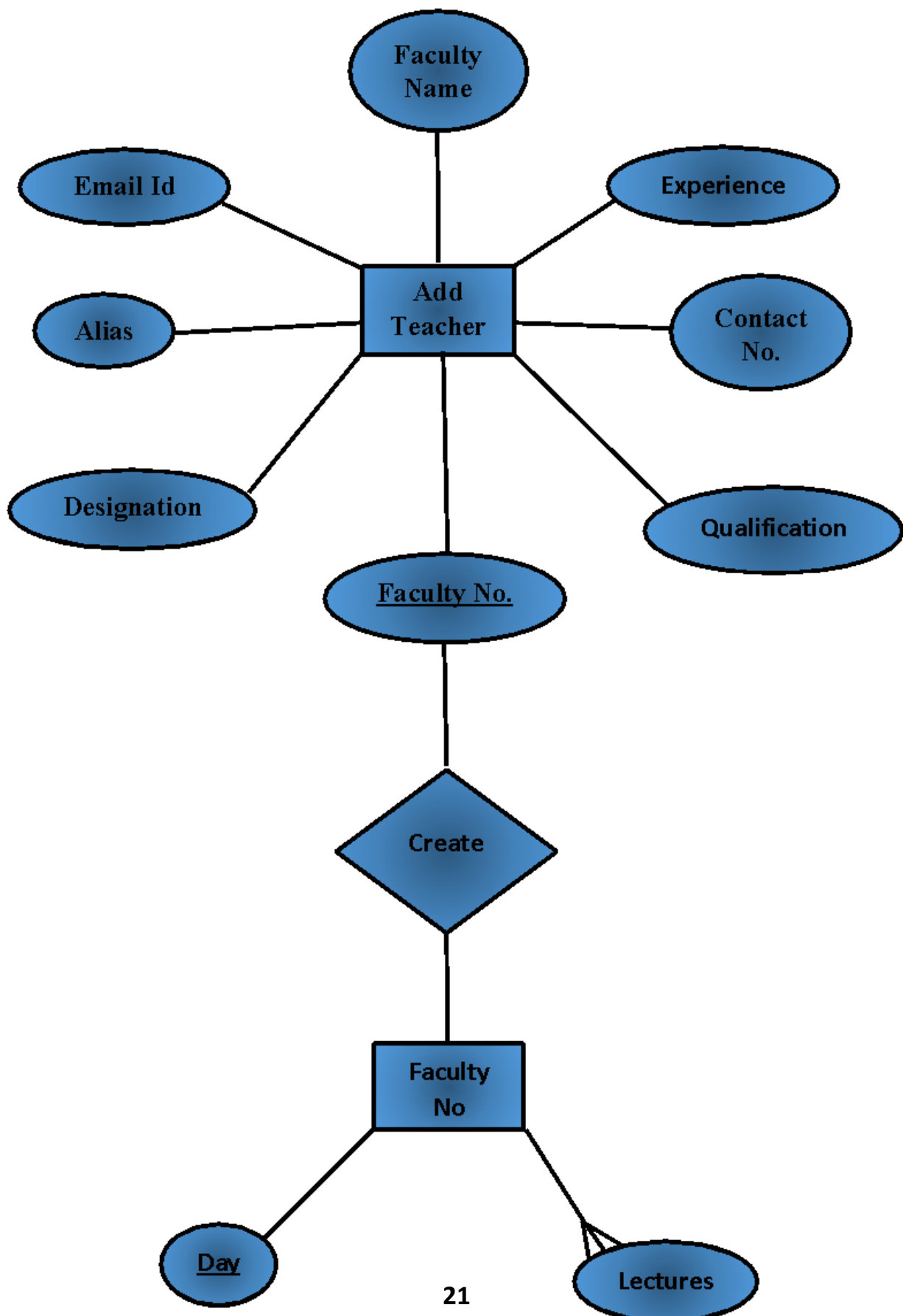
4.4.3 E-R DIAGRAM (3)

Add Subjects



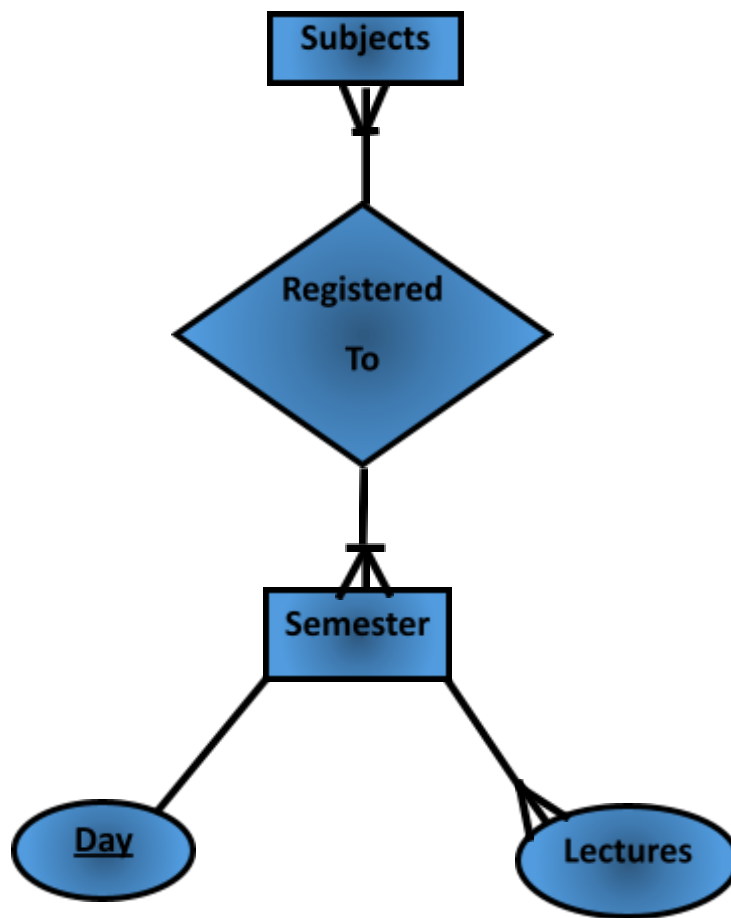
4.4.4 E-R DIAGRAM (4)

Teacher Registration

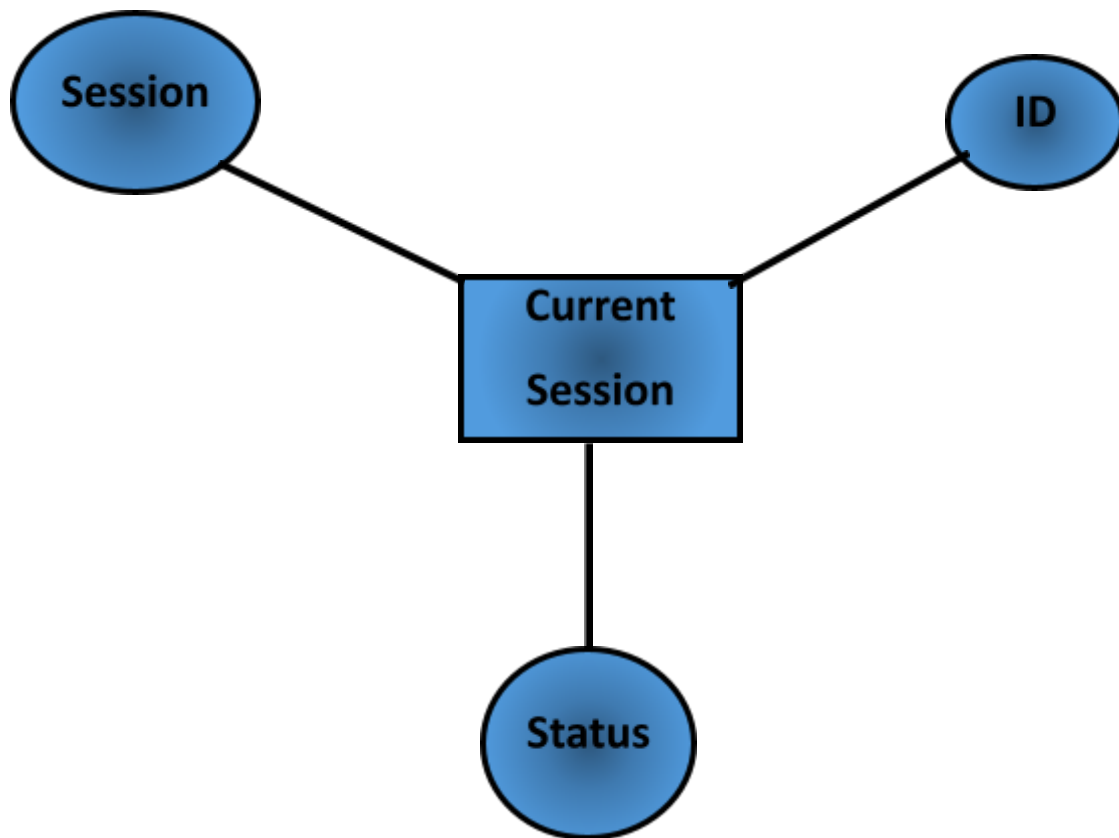


4.4.5 E-R DIAGRAM (5)

Semester Table

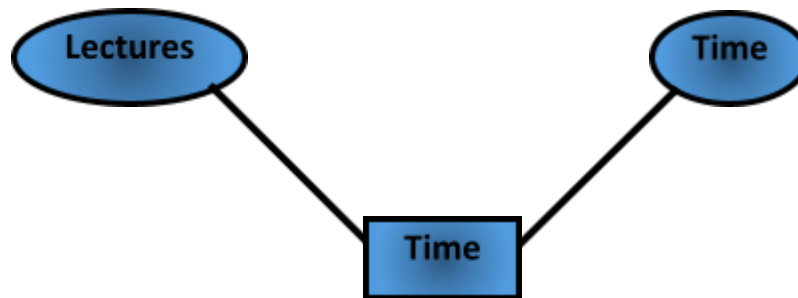


4.4.6 E-R DIAGRAM (6)
Session



4.4.7 E-R DIAGRAM (7)

Time Management



CODING

☐ **Front end.**

- **Html**
- **Css**
- **JavaScript**
- **Php**

☐ **Back end.**

- **MySQL**

Html Coding:

```
<html>  
  
<head>  
  
<title>AUTOMATED TIME TABLE GENERATION SYSTEM - HOME</title>  
  
<link rel='stylesheet' type='text/css' href='css sources/index.css'>  
<link rel='stylesheet' type='text/css' href='css sources/main.css'>  
  
</head>  
  
<body>  
  
<div id="logo">  
  
  
  
</div>  
  
<div class="topnav">  
  
<a href="index.php">Home</a>  
  
<a href="aboutus.html">>About us</a>  
  
</div>  
  
<div class="container-header">  
AUTOMATED TIMETABLE GENERATION SYSTEM  
</div>  
  
<div class="head-container">  
<marquee>ATGS -There is only one Key to Success, Be on Time - Everytime</marquee>  
  
</div>  
  
<div class="body-container">  
  
<div class="title">Welcome</div>  
  
  
  
<div id="para" style="width:390;">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b>This  
is our Project which is basically a tool for generation  
of a format of a random arrangement of a timetable by satisfying all the conditions which  
necessery to create a time table for that institute. This tool helps user to get a stretegy  
to help the user to create an effective time table for all the classes by finding the shortest path.<br>  
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&In our project we used multiple algorithms  
i.e; Ant coloney optimization  
(For searching Shortest Path), Genetic Algorithm (for creating a random arrangement ).<br><br>
```



```
} tr:nth-child(odd) {  
    background-color: #ffffff;  
}
```

PHP CODING: - Login page

```
<?php
include 'config.php';
if (isset($_POST['username']) && isset($_POST['pwd'])) {
    session_start();
    $id = $_POST['username'];
    $password = $_POST['pwd'];
    $query =mysql_query("SELECT * FROM admin_login
    WHERE username = '$id' and password = '$password'");
    $records = mysql_num_rows($query);
    $row = mysql_fetch_array($query);
    if ($records==0)
    {
        echo '<script type="text/javascript">alert("Wrong UserName or Password");
        window.location=\'forget.php\';</script>';
    }
    else
    {
        $_SESSION['Name']=$row['Name'];
        $_SESSION['Username']=$row['username'];
        $_SESSION['Pass']=$row['password'];
        $_SESSION['Mobile']=$row['Mob'];
        $_SESSION['Desig']=$row['designation'];
        $name=$_SESSION['Name'];
        echo '<script type="text/javascript">alert(" '.$name.' Welcome to Admin Panel");
        window.location=\'adminpanel.php\';</script>';
    }
}
?>
```

PHP CODING: - Add Faculty page

```
<?php
'connection.php';

if (isset($_POST['TN']) && isset($_POST['TF']) && isset($_POST['TE']) && isset($_POST['TD']) &&
isset($_POST['AL']) && isset($_POST['quali']) && isset($_POST['exp'])) {

$name = $_POST['TN'];

$facno = $_POST['TF'];

$designation = $_POST['TD'];

$alias = $_POST['AL'];

$contact = $_POST['TP'];

$email = $_POST['TE'];

$quali = $_POST['quali'];

$exp = $_POST['exp'];

} else {

$message = "dead.";

echo "<script type='text/javascript'>alert('$message');</script>";

die();

}

$q = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),

"INSERT INTO teachers VALUES
('$facno','$name','$alias','$designation','$contact','$email','$quali','$exp')");

$sql = "CREATE TABLE " . $facno . " (
day VARCHAR(10) PRIMARY KEY,
period1 VARCHAR(30),
period2 VARCHAR(30),
period3 VARCHAR(30),
period4 VARCHAR(30),
period5 VARCHAR(30),
period6 VARCHAR(30)
)";
```

```

mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), $sql);
$days = array('monday', 'tuesday', 'wednesday', 'thursday', 'friday', 'saturday');
for ($i = 0; $i < 6; $i++) {
    $day = $days[$i];
    $sql = "INSERT into " . $facno . " VALUES('$day',' ',' ',' ',' ')";
    mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), $sql);
}
if ($q) {
    $message = "Faculty Added ";
    echo "<script type='text/javascript'>alert('$message');</script>";
    header("Location:addteachers.php");
} else {
    $message = "No Response\\nTry again.";
    echo "<script type='text/javascript'>alert('$message');</script>";
}
?>

```

PHP CODING: - Add Subject page

```
<?php
include 'connection.php';

if (isset($_POST['SN']) && isset($_POST['SC']) && isset($_POST['SS']) && isset($_POST['SD'])) {

$name = $_POST['SN'];
$code = $_POST['SC'];
$sem = $_POST['SS'];
$course = $_POST['ST'];
$dept = $_POST['SD'];

} else {

$message = "dead.";

echo "<script type='text/javascript'>alert('$message');</script>";

die();

}

$q = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
"INSERT INTO subjects VALUES ('$code','$name','$course','$sem','$dept',0,"",",")");

if ($q) {

$message = "Subject added.";

echo "<script type='text/javascript'>alert('$message');</script>";

header("Location:addsubjects.php");

} else {

$message = "Username and/or Password incorrect.\nTry again.";

echo "<script type='text/javascript'>alert('$message');</script>";

}

?>
```


PHP CODING: - Assign Substitute

```
<?php
include('sms.php');
session_start();
$whose = $_SESSION['shown_id'];
$sub = $_POST['SB'];
$class = $_POST['CN'];
$days = array("monday", "tuesday", "wednesday", "thursday", "friday", "saturday");
$day = $days[( $class - 8 ) / 8];

$periods = array("period1", "period2", "period3", "period4", "period5", "period6");
$period = $periods[( $class - 1 ) % 8];
$query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
"SELECT * FROM teachers WHERE faculty_number = '$whose'");
$row = mysqli_fetch_assoc($query);
$whose_name = $row['name'];
$query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
"SELECT * FROM teachers WHERE faculty_number = '$sub'");
$row = mysqli_fetch_assoc($query);
$sub_name = $row['name'];
$whose = strtolower($whose);
$sub = strtolower($sub);
$query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
"SELECT * FROM $sub WHERE day = '$day'");
```

```

$row = mysqli_fetch_assoc($query);

$available = false;

if ($row[$period] == "-<br>-" || $row[$period] == "-<br>" || $row[$period] == "-") {
    $message = 'Message Sent!';
} else if (!isset($_POST['pwd'])) {
    echo "<script type='text/javascript'>
    alert('Selected substitute teacher is not available!');
    window.location.href = 'generatetimetable.php?display='$whose;</script>";
}

$query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
    "SELECT * FROM $whose WHERE day = '$day'");
$row = mysqli_fetch_assoc($query);
$pieces = explode("<br>", $row[$period]);
$string = "Hello " . $sub_name . ",
    You have to take class " . $pieces[0] . " of " . $whose_name . " in " . $pieces[1] .
    "\n\n-Sent from ATGS GPC BGT";
$_SESSION['s'] = $string;
echo 'Sending SMS...';

if (isset($_POST['pwd'])) {
    echo "<script type='text/javascript'>alert('Message Sent!');
    window.location.href = 'generatetimetable.php?display=" . $whose . "';
    </script>";
?>

```

PHP CODING: - Functions

```
<? php

include('curl.php');

@ini_set('zlib.output_compression', 0);

@ini_set('implicit_flush', 1);

@ob_end_clean();

set_time_limit(0);

ob_implicit_flush(1);

$time = time();

function go($url)

{

header('Location: ' . $url);

exit;

}

function input($text)

{

return trim(htmlspecialchars($text));

}

function output($text, $html = true)

{

if ($html) {

return trim(stripslashes($text));

} else {

return trim(htmlspecialchars(stripslashes($text)));

}

}

function rword($length = 10, $allow = "all")

{

if ($allow == "all")

$characters = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';

elseif ($allow == "num")
```

```

$characters = '0123456789';
elseif ($allow == "en")
$characters = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
elseif ($allow == "low")
$characters = 'abcdefghijklmnopqrstuvwxyz';
elseif ($allow == "up")
$characters = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
else
$characters = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
$randomString = "";
for ($i = 0; $i < $length; $i++) {
$randomString .= $characters[rand(0, strlen($characters) - 1)];
}
return $randomString;
}
function sor($html)
{
$html = htmlspecialchars($html);
return $html;
}
function clean($html)
{
$html = str_replace(array("\r\n", "\r", "\n"), "", $html);
return $html;
}
function show($text, $color = "darkred")
{
echo '<font color="' . $color . '">' . $text . '</font><br/>';
ob_flush();
flush();
}

```

```
function fc($text, $color = "darkred")
{
echo '<font color="' . $color . '"' . $text . '</font>';
ob_flush();
flush();
}

function err($head = "Error", $text = "Error")
{
echo '<div class="head">' . $head . '</div><div class="content">' . $text . '</div>';
ob_flush();
flush();
include "foot.php";
ob_flush();
flush();
die();
}

?>
```

PHP CODING: - Algorithm

```
<?php
ini_set('max_execution_time',300);

/**Class to store subject details**/

class Subject
{
public $code; //Subject code
public $classes = 0; //No. of classes
public $semester; //semester of subject
public $alias; //alias for subject teacher
public $alias2; //alias for subject teacher2 (only for practical subjects)
public $alias3; //alias for subject teacher3 (only for practical subjects)
public $subjectteacher; //faculty number of teacher1
public $subjectteacher2; //faculty number of teacher2
public $subjectteacher3; //faculty number of teacher3
}

/**Class to store teachers details**/

class Teacher
{
public $id; //faculty number
public $days = array(); //schedule
public $classroom_names = array(); //classroom names
}

$subjectslots = array(); //subjects slots for all semesters
$aliasslots = array(); //alias slots corresponding to each subject

$query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), "SELECT * FROM subjects ");
$subjects[] = new Subject(); //to store theory subjects
$practicals[] = new Subject(); //to store practical subjects
```

```

$count = 0;

/** fetching theory subjects and saving in subjects array*/
while ($row = mysqli_fetch_assoc($query)) {
    if ($row['course_type'] == 'LAB')
        continue;
    $temp = new Subject();
    $temp->code = $row['subject_code'];
    $temp->semester = $row['semester'];
    $temp->subjectteacher = $row['allotedto'];
    if (isset($temp->subjectteacher)) {
        $teacheralias_query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
        "SELECT * FROM teachers WHERE faculty_number='$temp->subjectteacher'");
        $row = mysqli_fetch_assoc($teacheralias_query);
        $temp->alias = $row['alias'];
    }
    $subjects[$count++] = $temp;
}
$subjects_count = $count;

/**Fetching teachers and saving into teachers array*/
$query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), "SELECT * FROM teachers ");

$teachers[] = new Teacher();
$count = 0;
while ($row = mysqli_fetch_assoc($query)) {
    $temp = new Teacher();
    $temp->id = $row['faculty_number'];
    $teachers[$count++] = $temp;
}
$teachers_count = $count;

```

```

$r = -1;

/** Genrating timetable for theory courses, with maximum class for each subject equal to 4 */
for ($i = 0; $i < $subjects_count * 6; $i++) {
    $i = $i % $subjects_count;
    $sem = $subjects[$i]->semester;
    $year = ($sem + 1) / 2;
    echo $year;
    $classroom_query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
    "SELECT name FROM classrooms WHERE status='$year'");
    $row = mysqli_fetch_assoc($classroom_query);
    $classroom = $row['name'];
    for ($j = 0; $j < 30; $j++) {
        $subject_teacher;
        for ($z = 0; $z < $count; $z++) {
            if ($teachers[$z]->id == $subjects[$i]->subjectteacher) {
                $tindex = $z;
                break;
            }
        }
        if ($j % 6 == 0)
            $r++;
        if (isset($subjectslots[$sem][$r % 6][$j % 5])) {
            //check if subjectslot is empty
            continue;
        } else if (isset($teachers[$tindex]->days[$sem % 2][$r % 6][$j % 5])) {
            //check if subject teacher is free
            continue;
        } else {
            //check if existing in same day
            $already = false;

```



```

for ($z = 0; $z < 5; $z++) {
if (isset($subjectsslots[$sem][$r % 6][$z])) {
if ($z == ($j % 5)) {
continue;
}
if ($subjectsslots[$sem][$r % 6][$z] == $subjects[$i]->code) {
$already = true;
}
}
}
if ($already) {
continue;
}
// set subject
$subjects[$i]->classes++;
$subjectsslots[$sem][$r % 6][$j % 5] = $subjects[$i]->code;
$aliasslots[$sem][$r % 6][$j % 5][0] = $subjects[$i]->alias;
$teachers[$tindex]->days[$sem % 2][$r % 6][$j % 5] = $subjects[$i]->code;
$teachers[$tindex]->classroom_names[$sem % 2][$r % 6][$j % 5] = $classroom;
break;
}
}
}

/*****check for empty slots in semester's
timetable*****/

for ($i = 1; $i < 7; $i += 2) {
for ($k = 0; $k < 6; $k++) {
for ($j = 0; $j < 5; $j++) {

if (isset($subjectsslots[$i][$k][$j % 5])) {
} else {

```

```

$subjectslots[$i][$k][$j % 5] = "-";
$aliasslots[$i][$k][$j % 5][0] = "-";
}

}

}

}

/*****check for empty slots in teacher's
timetable*****/

for ($i = 0; $i < $count; $i++) {
for ($k = 0; $k < 6; $k++) {
for ($j = 0; $j < 5; $j++) {

if (isset($teachers[$i]->days[1][$k][$j])) {
} else {
$teachers[$i]->days[1][$k][$j] = "-";
$teachers[$i]->classroom_names[1][$k][$j] = "-";
}
}
}
}

/**Fetching info of practical courses **/

$query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), "SELECT * FROM subjects ");
$count = 0;
while ($row = mysqli_fetch_assoc($query)) {
if (!($row['course_type'] == 'LAB'))
continue;
$temp = new Subject();
$temp->code = $row['subject_code'];
$temp->semester = $row['semester'];

```

```

$temp->subjectteacher = $row['allotedto'];
$temp->subjectteacher2 = $row['allotedto2'];
$temp->subjectteacher3 = $row['allotedto3'];
$teacheralias_query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
"SELECT * FROM teachers WHERE faculty_number='$temp->subjectteacher'");
$row = mysqli_fetch_assoc($teacheralias_query);
$temp->alias = $row['alias'];
$teacheralias_query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
"SELECT * FROM teachers WHERE faculty_number='$temp->subjectteacher2'");
$row = mysqli_fetch_assoc($teacheralias_query);
$temp->alias2 = $row['alias'];
$teacheralias_query = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
"SELECT * FROM teachers WHERE faculty_number='$temp->subjectteacher3'");
$row = mysqli_fetch_assoc($teacheralias_query);
$temp->alias3 = $row['alias'];
$practicals[$count++] = $temp;
}
for ($i = 0; $i < 2 * $count; $i++) {
    $i = $i % $count;
    $sem = $practicals[$i]->semester;
    $index = -1;
    $index2 = -1;
    $index3 = -1;
    for ($z = 0; $z < $teachers_count; $z++) {
        if (isset($practicals[$i]->subjectteacher)) {
            if ($teachers[$z]->id == $practicals[$i]->subjectteacher) {
                $index = $z;
            }
        }
    }
    if (isset($practicals[$i]->subjectteacher2)) {
        if ($teachers[$z]->id == $practicals[$i]->subjectteacher2) {

```

```

$index2 = $z;
}
}
if (isset($practicals[$i]->subjectteacher3)) {
if ($teachers[$z]->id == $practicals[$i]->subjectteacher3) {
$index3 = $z;
}
}
}

//checking if all three teachers are free
for ($j = 0; $j < 6; $j++) {
if (isset($subjectslots[$sem][$j][5])) {
continue;
} else if (isset($teachers[$index]->days[$sem % 2][$j][5]) || isset($teachers[$index2]->days[$sem % 2][$j][5]) ||
isset($teachers[$index3]->days[$sem % 2][$j][5])
) {
continue;
} else {
//if free then assign practical
$practicals[$i]->classes++;
$subjectslots[$sem][$j][5] = $practicals[$i]->code;
$aliasslots[$sem][$j][5][0] = $practicals[$i]->alias;
$aliasslots[$sem][$j][5][1] = $practicals[$i]->alias2;
$aliasslots[$sem][$j][5][2] = $practicals[$i]->alias3;
$teachers[$index]->days[$sem % 2][$j][5] = $practicals[$i]->code;
$teachers[$index2]->days[$sem % 2][$j][5] = $practicals[$i]->code;
$teachers[$index3]->days[$sem % 2][$j][5] = $practicals[$i]->code;
break;
}
}

```

```

}
}

/**checks for empty slot**/
for ($i = 1; $i < 7; $i += 2) {
for ($j = 0; $j < 6; $j++) {
if (isset($subjectslots[$i][$j][5])) {}
else {
$subjectslots[$i][$j][5] = '-';
$aliasslots[$i][$j][5][0] = '-';
$aliasslots[$i][$j][5][1] = '-';
$aliasslots[$i][$j][5][2] = '-';
}
}
}

for ($i = 0; $i < $teachers_count; $i++) {
for ($k = 0; $k < 6; $k++) {

if (isset($teachers[$i]->days[1][$k][5])) {}
else {
$teachers[$i]->days[1][$k][5] = "-";
}
}
}

/*****Saving semesters timetable into database*****/
$days = array('monday', 'tuesday', 'wednesday', 'thursday', 'friday', 'saturday');
for ($i = 1; $i < 7; $i += 2) {
$database_name = " semester" . $i . " ";

```

```

for ($j = 0; $j < 6; $j++) {

$query = "UPDATE" . $database_name . " SET period1= " . $subjectslots[$i][$j][0] . "<br>" .
$aliasslots[$i][$j][0][0] . "",

period2=" " . $subjectslots[$i][$j][1] . "<br>" . $aliasslots[$i][$j][1][0] . "",

period3=" " . $subjectslots[$i][$j][2] . "<br>" . $aliasslots[$i][$j][2][0] . "",

period4=" " . $subjectslots[$i][$j][3] . "<br>" . $aliasslots[$i][$j][3][0] . "",

period5=" " . $subjectslots[$i][$j][4] . "<br>" . $aliasslots[$i][$j][4][0] . "",

period6=" " . $subjectslots[$i][$j][5] . "<br>" . $aliasslots[$i][$j][5][0] . " , " . $aliasslots[$i][$j][5][1] . " ,
" . $aliasslots[$i][$j][5][2] . ""

WHERE day=" " . $days[$j] . " " ";

$q = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), $query);

}

}

/*****Saving teachers timetable into database*****/

for ($i = 0; $i < $teachers_count; $i++) {

$database_name = " " . strtolower($teachers[$i]->id) . " ";

for ($j = 0; $j < 6; $j++) {

$query = "UPDATE" . $database_name . " SET period1= " . $teachers[$i]->days[1][$j][0] . "<br>" .
$teachers[$i]->classroom_names[1][$j][0] . "",

period2=" " . $teachers[$i]->days[1][$j][1] . "<br>" . $teachers[$i]->classroom_names[1][$j][1] . "",

period3=" " . $teachers[$i]->days[1][$j][2] . "<br>" . $teachers[$i]->classroom_names[1][$j][2] . "",

period4=" " . $teachers[$i]->days[1][$j][3] . "<br>" . $teachers[$i]->classroom_names[1][$j][3] . "",

period5=" " . $teachers[$i]->days[1][$j][4] . "<br>" . $teachers[$i]->classroom_names[1][$j][4] . "",

period6=" " . $teachers[$i]->days[1][$j][5] . ""

WHERE day=" " . $days[$j] . " " ";

$q = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), $query);

}

}

/*****redirect back to generate timetable **/

header("Location:generatetimetable.php?success=true");

```

?>

PHP CODING: - Time management

```
<?php
include 'connection.php';

if (isset($_POST['NL']) && isset($_POST['MN']) && isset($_POST['ET']) && isset($_POST['STH'])
&& isset($_POST['STM'])) {

    $nol = $_POST['NL'];
    $m = $_POST['MN'];
    $ext = $_POST['ET'];
    $sthr = $_POST['STH'];
    $stmin = $_POST['STM'];
    $am="am";

    // $message = "nTry again.";
    // echo "<script type='text/javascript'>alert('$message');</script>";

}
else
{
    $message = "Try again";
    echo "<script type='text/javascript'>alert('$message');</script>";
    die();
}

$nol=$nol+1;
$scn=$nol/2 + 1;
$cth=$sthr;
$ctm=$stmin;

$q=mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), "INSERT INTO time VALUES
('1','$sthr:$stmin $am');");

for($i=2;$i<$nol;$i++)
{
    if($m>59)
    {
        if($ctm>60 || $ctm==60 || $ctm>00 || $ctm==00)
```



```

{
$cth=$cth+1;
$ctm=$ctm-60;
if($cth>12)
{
$cth=1;
}
if($cth>11)
{
$am="pm";
}
if($i>1)
{
$ctm=$ctm+$m;
}
if($i==$cen)
{
$ctm=$ctm+$ext;
}
if($ctm=0)
{
$ctm=00;
}
$q=mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), "INSERT INTO time VALUES
('$i','$cth:$ctm $am');");
}
}
if($m<60)
{
if($ctm>00 || $ctm==00)
{

```

```

if($cth>12)
{
$cth=1;
}
if($cth>11)
{
$am="pm";
}
if($i>1)
{
$ctm=$ctm+$m;
}
if($ctm>60 || $ctm==60)
{
$cth=$cth+1;
$ctm=$ctm-60;
}
if($i==$cen)
{
$ctm=$ctm+$ext;
}

$q=mysqli_query(mysqli_connect("localhost", "root", "", "atgs"), "INSERT INTO time VALUES
('$i','$cth:$ctm $am');");
}
}
}
if ($q) {
$message = "Time Set Successfully.";
echo "<script type='text/javascript'>alert('$message');</script>";
header("Location:Timemanagement.php");
} else {

```

```

$message = "error occurred\nTry again.";
echo "<script type='text/javascript'>alert('$message');</script>";
// header("Location:index.php");
}
?>

```

PHP CODING: - Add Classroom

```

<?php
include 'connection.php';
if (isset($_POST['CN'])) {
    $name = $_POST['CN'];
} else {
    $message = "dead.";
    echo "<script type='text/javascript'>alert('$message');</script>";
    die();
}
$q = mysqli_query(mysqli_connect("localhost", "root", "", "atgs"),
    "INSERT INTO classrooms VALUES ('$name',0)");
if ($q) {
    $message = "Classroom added.";
    echo "<script type='text/javascript'>alert('$message');</script>";
    header("Location:addclassrooms.php");
} else {
    $message = "Username and/or Password incorrect.\nTry again.";
    echo "<script type='text/javascript'>alert('$message');</script>";
}
?>

```

PHP CODING: - Curl

```
<?php

$agent = $_SERVER['HTTP_USER_AGENT'];

function getcap($url, $ch, $ckfile, $proxy = "", $ref = "")
{
    global $agent;
    if ($ref == "")
        $ref = $url;
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_HEADER, 0);
    curl_setopt($ch, CURLOPT_USERAGENT, $agent);
    curl_setopt($ch, CURLOPT_HTTPHEADER, Array("Content-Type: image/jpeg", "Accept: */*"));
    curl_setopt($ch, CURLOPT_COOKIEJAR, $ckfile);
    curl_setopt($ch, CURLOPT_PROXY, $proxy);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($ch, CURLOPT_NOPROGRESS, false);
    curl_setopt($ch, CURLOPT_VERBOSE, true);
    curl_setopt($ch, CURLOPT_BINARYTRANSFER, 1);
    curl_setopt($ch, CURLOPT_REFERER, $ref);
    return curl_exec($ch);
}
```

```

}

function ngetcap($url, $sch, $ckfile, $proxy = "", $ref = "")
{
    global $agent;
    if ($ref == "")
        $ref = $url;
    curl_setopt($sch, CURLOPT_URL, $url);
    curl_setopt($sch, CURLOPT_HEADER, 0);
    curl_setopt($sch, CURLOPT_USERAGENT, $agent);
    curl_setopt($sch, CURLOPT_HTTPHEADER, Array("Content-Type: image/jpeg", "Accept: */*"));
    curl_setopt($sch, CURLOPT_COOKIEFILE, $ckfile);
    curl_setopt($sch, CURLOPT_PROXY, $proxy);
    curl_setopt($sch, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($sch, CURLOPT_NOPROGRESS, false);
    curl_setopt($sch, CURLOPT_VERBOSE, true);
    curl_setopt($sch, CURLOPT_BINARYTRANSFER, 1);
    curl_setopt($sch, CURLOPT_REFERER, $ref);
    return curl_exec($sch);
}

```

```

function npost($url, $post, $sch, $ckfile, $proxy = "", $ref = "")
{
    global $agent;
    if ($ref == "")
        $ref = $url;
    curl_setopt($sch, CURLOPT_URL, $url);
    curl_setopt($sch, CURLOPT_USERAGENT, $agent);
    curl_setopt($sch, CURLOPT_HTTPHEADER, Array("Content-Type:
application/x-www-form-urlencoded", "Accept: */*"));
    curl_setopt($sch, CURLOPT_COOKIEFILE, "$ckfile");
    curl_setopt($sch, CURLOPT_PROXY, $proxy);

```

```

curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_HEADER, true);
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
curl_setopt($ch, CURLOPT_REFERER, $ref);
return curl_exec($ch);
}

function post($url, $post, $ch, $ckfile, $proxy = "", $ref = "")
{
    global $agent;
    if ($ref == "")
        $ref = $url;
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_USERAGENT, $agent);
    curl_setopt($ch, CURLOPT_HTTPHEADER, Array("Content-Type:
application/x-www-form-urlencoded", "Accept: */*"));
    curl_setopt($ch, CURLOPT_COOKIEJAR, "$ckfile");
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($ch, CURLOPT_HEADER, true);
    curl_setopt($ch, CURLOPT_PROXY, $proxy);
    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_POST, true);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
    curl_setopt($ch, CURLOPT_REFERER, $ref);
    return curl_exec($ch);
}

function get($url, $ch, $ckfile, $proxy = "", $ref = "")
{
    global $agent;

```

```

if ($ref == "")
$ref = $url;

curl_setopt($sch, CURLOPT_URL, $url);

curl_setopt($sch, CURLOPT_USERAGENT, $agent);

curl_setopt($sch, CURLOPT_HTTPHEADER, Array("Content-Type:
application/x-www-form-urlencoded", "Accept: */*"));

curl_setopt($sch, CURLOPT_COOKIEJAR, "$ckfile");

curl_setopt($sch, CURLOPT_HEADER, true);

curl_setopt($sch, CURLOPT_RETURNTRANSFER, true);

curl_setopt($sch, CURLOPT_FOLLOWLOCATION, true);

curl_setopt($sch, CURLOPT_PROXY, $proxy);

curl_setopt($sch, CURLOPT_REFERER, $ref);

return curl_exec($sch);
}

```

```

function nget($url, $sch, $ckfile, $proxy = "", $ref = "")
{
global $agent;

if ($ref == "")
$ref = $url;

curl_setopt($sch, CURLOPT_URL, $url);

curl_setopt($sch, CURLOPT_USERAGENT, $agent);

curl_setopt($sch, CURLOPT_HTTPHEADER, Array("Content-Type:
application/x-www-form-urlencoded", "Accept: */*"));

curl_setopt($sch, CURLOPT_COOKIEFILE, "$ckfile");

curl_setopt($sch, CURLOPT_PROXY, $proxy);

curl_setopt($sch, CURLOPT_FOLLOWLOCATION, true);

curl_setopt($sch, CURLOPT_RETURNTRANSFER, true);

curl_setopt($sch, CURLOPT_HEADER, true);

curl_setopt($sch, CURLOPT_SSL_VERIFYPEER, false);

curl_setopt($sch, CURLOPT_REFERER, $ref);

```

```
return curl_exec($ch);  
}
```

?>

TEST CASE DESIGN

6.1 White Box Testing

White box testing techniques analyze the internal structures the used data structures, internal design, code structure and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing.

Working process of white box testing:

- **Input:** Requirements, Functional specifications, design documents, source code.
- **Processing:** Performing risk analysis for guiding through the entire process.
- **Proper test planning:** Designing test cases so as to cover entire code. Execute rinse-repeat until error-free software is reached. Also, the results are communicated.
- **Output:** Preparing final report of the entire testing process.

Advantages:

- White box testing is very thorough as the entire code and structures are tested.
- It results in the optimization of code removing error and helps in removing extra lines of code.
- Easy to automate.

Disadvantages:

- Main disadvantage is that it is very expensive.
- Missing functionalities cannot be detected as the code that exists is tested.
- Very complex and at times not realistic.

6.2 Black box Testing

Black box testing is the Software testing method which is used to test the software without knowing the internal structure of code or program. Most likely this testing method is what most of tester actually perform and used the majority in the practical life.

Basically software under test is called as “Black-Box”, we are treating this as black box & without checking internal structure of software we test the software. All testing is done as customer’s point of view and tester is only aware of what is software is supposed to do but how these requests are processed by software is not aware. While testing tester knows about the input and expected output’s of the software and they do not aware of how the software or application actually processes the input requests & giving the outputs. Tester only passes valid as well as invalid inputs & determines the correct expected outputs. All the test cases to test using such method are calculated based on requirements & specifications document.

6.3 Unit Testing

Unit Testing is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software.

It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

6.4 Integration Testing

Integration testing is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

- **integration testing:** Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems. See also component integration testing, system integration testing.
- **component integration testing:** Testing performed to expose defects in the interfaces and interaction between integrated components.

- **system integration testing:** Testing the integration of systems and packages; testing interfaces to external organizations (e.g. Electronic Data Interchange, Internet).

6.5 System Testing

System testing is testing conducted on a complete integrated system to evaluate the system's compliance with its specified requirements.

System testing takes, as its input, all of the integrated components that have passed integration testing. The purpose of integration testing is to detect any inconsistencies between the units that are integrated together (called assemblages). System testing seeks to detect defects both within the "inter-assemblages" and also within the system as a whole. The actual result is the behavior produced or observed when a component or system is tested.

System testing is performed on the entire system in the context of either functional requirement specifications (FRS) or system requirement specification (SRS), or both. System testing tests not only the design, but also the behaviour and even the believed expectations of the customer. It is also intended to test up to and beyond the bounds defined in the software or hardware requirements specification(s).

6.6 Acceptance Testing

Acceptance testing, a testing technique performed to determine whether or not the software system has met the requirement specifications. The main purpose of this test is to evaluate the system's compliance with the business requirements and verify if it has met the required criteria for delivery to end users.

There are various forms of acceptance testing:

- Alpha Testing
- Beta Testing

➤ Alpha Testing

This is to assess the Product in the development/testing environment by a specialized testers team usually called alpha testers. Here, the testers feedback, suggestions help to improve the Product usage and also to fix certain bugs.

Here, testing happens in a controlled manner.

➤ Beta Testing

This is to assess the Product by exposing it to the real end-users, usually called beta testers/beta users, in their environment. Continuous feedback from the users

is collected and the issues are fixed. Also, this helps in enhancing/improving the Product to give a rich user experience.

Testing happens in an uncontrolled manner, which means a user has no restrictions on the way in which the Product is being used.

6.6 V Model :

The V-model is a type of SDLC model where process executes in a sequential manner in V-shape. It is also known as Verification and Validation model. It is based on the association of a testing phase for each corresponding development stage. Development of each step directly associated with the testing phase. The next phase starts only after completion of the previous phase i.e. for each development activity, there is a testing activity corresponding to it.

Verification: It involves static analysis technique (review) done without executing code. It is the process of evaluation of the product development phase to find whether specified requirements meet.

Chapter 7

OUTPUT SCREENS

1. Homepage



[Home](#) [About us](#)

Welcome

This is our Project which is basically a tool for generation of a format of a random arrangement of a timetable by satisfying all the conditions which necessary to create a time table for that institute. This tool helps user to get a strategy to help the user to create an effective time table for all the classes by finding the shortest path.

In our project we used multiple algorithms i.e; Ant colony optimization (For searching Shortest Path), Genetic Algorithm (for creating a random arrangement).

NOTE - This project Works only on Local server so no other user can access this software by other system.

Thank You.

Login admin account

Username:

Page 10 of 10

Password:

LOGIN

[Forget Password](#)

2.Admin Panel

Admin Panel - ATGS

localhost/projects/ATGS%20Yash%2012%20-%20Copy/adminpanel.php

AUTOMATED TIMETABLE GENERATION SYSTEM

ATGS - There is only one Key to Success, Be on Time - Everytime

ATGS

Home

Add Teacher

Add Subject


Add Classroom

Allocation

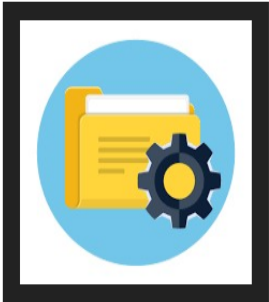
Logout

CSE GPCB


ADMIN PANEL




Registration



Manage



Generate



Modify Timetable

3. Admin Registration

AUTOMATED TIME TABLE GENERATION SYSTEM - HOME

localhost/projects/ATGS%20Yash%2012%20-%20Copy/Registration%20admin.php

AUTOMATED TIMETABLE GENERATION SYSTEM

ATGS - There is only one Key to Success, Be on Time - Everytime

ATGS

HomeAdd TeacherAdd SubjectAdd ClassroomAllocationLogout

ADMIN REGISTRATION FORM

Name:	<input type="text"/>
Username*:	<input type="text"/>
Password:	<input type="password"/>
ConfirmPassword:	<input type="password"/>
Email Id:	<input type="text"/>
Security Question	<input type="text"/>
Designation:	<input type="text"/>
<input type="button" value="Register"/>	<input type="button" value="Reset"/>

4. Add Teacher



Faculty Registration

Add Teacher

ADD TEACHER

Teacher's Information

Faculty No	Teacher's Name ...	Contact No.	Email ID	Qualification	Experience	Action
t02	Faculty No. Ex. t01,t02	5985895454	priyaballey32jdgfyud	M.tech	4	Delete
t03	Alias..	8268219735	kbcسدikbiduf	M.tech	4	Delete
t04	Select	2345678756	kjsdhaisuugdyud	Be/B.tech	5	Delete
t05	+91 ...	9589413144	jhabdsuydtwq	btech	4	Delete
t06	abc@xyz.com ...	8268219735	jhdtsudgfguf	btech	6	Delete
	Qualification					
	Experience					

5. Add Subjects

Subject Registration

localhost/projects/ATGS%20Yash%2012%20-%20Copy/addsubjects.php

AUTOMATED TIMETABLE GENERATION SYSTEM

ATGS - There is only one Key to Success, Be on Time - Everytime

Home Add Teacher Add Subject Add Classroom Allocation Logout

Subject Registration

Add Subject

Subject Name
Subject's Name ...

Subject Code
CO203 CO205...

Course Type
Select

Semester
Select

Department
Select

ADD

ADD SUBJECT

Subject's Information

Code	Course Type	Semester	Department	Action
101	THEORY	1	Computer Engg.	Delete
102	THEORY	1	Computer Engg.	Delete
102p	LAB	1	Computer Engg.	Delete
103	THEORY	1	Computer Engg.	Delete
104	THEORY	1	Computer Engg.	Delete
104p	LAB	1	Computer Engg.	Delete
106	THEORY	1	Computer Engg.	Delete

Type here to search

5:23 PM 19/04/2019

6. Add Classroom

Add Classrooms

localhost/projects/ATGS%20Yash%2012%20-%20Copy/addclassrooms.php

AUTOMATED TIMETABLE GENERATION SYSTEM

IGS - There is only one Key to Success, Be on Time - Everytime

ATGS

HomeAdd TeacherAdd SubjectAdd ClassroomAllocationLogout

Add Classroom

Add Classroom

Name

ML 32, NL 33 ...

ADD

ADD CLASSROOM

ADDED CLASSROOMS

	Action
	Delete
FF5	Delete
FF4	Delete

7. Allocation Panel

ALLOCATION

localhost/projects/ATGS%20Yash%2012%20-%20Copy/alloc.html

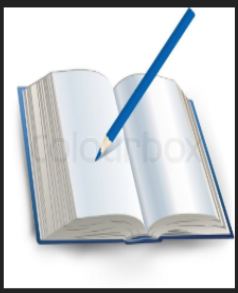
AUTOMATED TIMETABLE GENERATION SYSTEM

ATGS - There is only one Key to Success, Be on Time - Everytime


ATGS

Home Add Teacher Add Subject Add Classroom Allocation Logout


Allocation




THEORY
COURSES



PRACTICAL
COURSES



CLASSROOM
ALLOCATION



SET TIME
SCHEDULE

Type here to search

ENG 5:24 PM 19/04/2019

8. Allot Theory Courses

TimeTable Management System

localhost/projects/ATGS%20Yash%2012%20-%20Copy/allotsubjects.php

AUTOMATED TIMETABLE GENERATION SYSTEM

ATGS - There is only one Key to Success, Be on Time - Everytime

ATGS

HomeAdd TeacherAdd SubjectAdd ClassroomPracticalsClassroomsLogout

Theory Classes Allocation

Select Subject

Select Teacher

Allot

THEORY COURSES ALLOTMENT

Subject Code	Subject Title	Faculty No	Teacher's Name	Action
303	Data structures	t03	tripti adkane	Delete
304	Operating system	t05	simpy awadhiya	Delete

9. Allot Theory Courses

Practicals Allocation

localhost/projects/ATGS%20Yash%2012%20-%20Copy/allotpracticals.php

AUTOMATED TIMETABLE GENERATION SYSTEM

ATGS - There is only one Key to Success, Be on Time - Everytime



[Home](#) [Add Teacher](#) [Add Subject](#) [Add Classroom](#) [Theory](#) [Classrooms](#) [Logout](#)

Practical Classes Allocation

Select Subject ▼

Select Teacher ▼

Select Teacher ▼

Select Teacher ▼

Allot

PRACTICAL COURSES ALLOTMENT

Subject Code	Subject Title	Faculty No	Teacher's Name	Faculty No	Teacher's Name	Faculty No	Teacher's Name	Action
304p	Operating system	t03	tripli adkane	t03	tripli adkane	t03	tripli adkane	Delete
303p	Data structures	t05	simpy awadhiya	t05	simpy awadhiya	t05	simpy awadhiya	Delete
305p	Visual Basic	t04		t04		t04		Delete

10. Account Management

Admin Panel - ATGS

localhost/projects/ATGS%20Yash%2012%20-%20Copy/manage_account.php

AUTOMATED TIMETABLE GENERATION SYSTEM

ATGS - There is only one Key to Success, Be on Time - Everytime

ATGS

Home Add Teacher Add Subject Add Classroom Allocation  Logout

Welcome CSE GPCB

Username	<input type="text" value="CSE"/>
Name	<input type="text" value="CSE GPCB"/>
Password	<input type="text" value="1000"/>
Email Id	<input type="text"/>
Designation	<input type="text" value="Head"/>

[Change Profile](#)

11. View Timetable


COMPUTER ENGINEERING DEPARTMENT SEMESTER 5							
(FF5)							
WEEKDAYS	10:00 am	11:0 am	12:0 pm	1:0 pm	2:0 pm	3:0 pm	4:0 pm
MONDAY	504 SD	511 ta	503 pb	LUNCH	502 SD	508 LS	501P pb, pb, pb
TUESDAY	511 ta	504 SD	503 pb	LUNCH	- -	- -	502P SD, SD, SD
WEDNESDAY	501 pb	511 ta	504 SD	LUNCH	- -	- -	503P pb, pb, pb
THURSDAY	502 SD	508 LS	501 pb	LUNCH	511 ta	- -	501P pb, pb, pb
FRIDAY	503 pb	502 SD	508 LS	LUNCH	501 pb	- -	502P SD, SD, SD
SATURDAY	504 SD	503 pb	502 SD	LUNCH	508 LS	501 pb	503P pb, pb, pb

501: Web Technology, pb: Pragya Balley
 502: Java Programming, :
 503: Hardware Installation & Maintenance, pb: Pragya Balley
 504: Software Enoo.. :

12. Update Timetable

Time Table Generation
localhost/projects/ATGS%20Yash%2012%20-%20Copy/timetableupdate.php

AUTOMATED TIMETABLE GENERATION SYSTEM
ATGS - There is only one Key to Success, Be on Time - Everytime



Home
Modify Timetable
Update Teacher table
Logout

Update Semester Table

Select

search

WEEKDAYS	8:00-8:40	8:40-9:20	9:20-10:00	10:15-10:55	10:55-11:35	11:35-12:15	
monday	303 ta	308 SA	 	 	 	304p ta, ta, ta	update
tuesday	304 SA	301 pb	303 ta	 	 	303p SA, SA, SA	update
wednesday	302 ta	304 SA	301 pb	303 ta	 	305p SD, SD, SD	update
thursday	308 SA	302 ta	304 SA	301 pb	303 ta	306p SD, SD, SD	update

Chapter 8

COST ESTIMATION

8.1 COCOMO Model

Cocomo (Constructive Cost Model) is a regression model based on LOC, i.e **number of Lines of Code**. It is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality. It was proposed by Barry Boehm in 1970 and is based on the study of 63 projects, which make it one of the best-documented models.

The key parameters which define the quality of any software products, which are also an outcome of the Cocomo are primarily Effort & Schedule:

- **Effort:** Amount of labor that will be required to complete a task. It is measured in person-months units.
- **Schedule:** Simply means the amount of time required for the completion of the job, which is, of course, proportional to the effort put. It is measured in the units of time such as weeks, months.

Different models of Cocomo have been proposed to predict the cost estimation at different levels, based on the amount of accuracy and correctness required. All of these models can be applied to a variety of projects, whose characteristics determine the value of constant to be used in subsequent calculations. These characteristics pertaining to different system types are mentioned below.

- **Basic COCOMO:** Computes software development effort and cost as a function of program size expressed in estimated DSIs. There are three modes within Basic COCOMO:
- **Basic Model –**

$$\text{Effort} = a(\text{KLOC})^b$$

- The above formula is used for the cost estimation of for the basic COCOMO model, and also is used in the subsequent models. The constant values a and b for the Basic Model for the different categories of system:

$MM = a * KDSI^b$			
$TDEV = 2.5 * MM^c$			

Basic COCOMO	a	b	c
Organic	2.4	1.05	0.38
Semi-detached	3.0	1.12	0.35
Embedded	3.6	1.20	0.32

- The effort is measured in Person-Months and as evident from the formula is dependent on Kilo-Lines of code. These formulas are used as such in the Basic Model calculations, as not much consideration of different factors such as reliability, expertise is taken into account, henceforth the estimate is rough.

- **Intermediate COCOMO:** An extension of the Basic model that computes software development effort by adding a set of "cost drivers," that will determine the effort and duration of the project, such as assessments of personnel and hardware.

$MM = a * KDSI^b$			
$TDEV = 2.5 * MM^c$			

Intermediate COCOMO	a	b	c
Organic	3.2	1.05	0.38
Semi-detached	3.0	1.12	0.35
Embedded	2.8	1.20	0.32

- **Detailed COCOMO:** an extension of the Intermediate model that adds effort multipliers for each phase of the project to determine the cost drivers impact on each step.

COCOMO was developed by Barry Boehm in his 1981 book, Software Engineering Economics.

Cost Estimation

We Developed this project for college level so, this project comes under the **basic cocomo** category. And we are the beginners in the field of software engineering so we applied the **organic** System

No of LOC = 30,520

KLOC(f) = LOC/1000.

KLOC = 30.52

$$\text{Effort}(f) = a_1 * (\text{KLOC})^{b_2}$$

$$\text{Effort} = 2.4 * (30.52)^{1.05} = 86.9$$

$$T_{\text{dev}}(f) = C_3 * (\text{Effort})^{d_4} \text{ P.M.}$$

$$T_{\text{dev}} = 2.5 * (86.90)^{0.38} = 13.71 \text{ P.M.}$$

$$\text{Total Time Development} = T_{\text{dev}} / \text{staff.} = 2.7 \text{ Months}$$

$$\text{Cost in one Month} = 5000 \text{ Rs}$$

$$\text{Total Cost} = 13,638 \text{ Rs.}$$

Chapter 9

CONCLUSION

So we can conclude that this system which was developed by our team has perfectly meets the maximum requirements and the whole efforts was spend by our team has given the best quality of output. We now that this software has many limitations but, this is our highest limit of optimization of the system. In analysis we studied about the problem and after this we planned a best solution for the core problem. And now, we finally implimented the automated timetable generation system as a timetable generator tool according to the solution we planned.

We used constraints setisfaction as a solution criteria and ant coloney optimization as a solution base. This system is very effective for the generation of timetable using AI. In our system we used localserver and php (hypertext preprocessor) as a server-side scripting language. And now our imagination of a tool developed in php using ai technology got shape.

This system is cost effective, efficient, reliable, optimized and works offline but you have to must open **EASYPHP DEVSERVER** first before using this system.

Chapter 10

USER MANUAL

Chapter 11

BIBLIOGRAPHY

1. Research Gate:

https://www.researchgate.net/publication/326265336_A_STUDY_ON_AUTOMATIC_TIMETABLE_GENERATOR
OR.

2. Php point:

<https://www.phptpoint.com/timetable-generator-free-download/>

3. Slide share:

<https://www.slideshare.net/VasagiriSaiSumanth/timetable-generation-and-management-system>

4. IJSER:

<https://www.ijser.org/researchpaper/Automated-college-timetable-generator.pdf>

5. Academia:

https://www.academia.edu/24552959/Web_Application_for_Automatic_Time_Table_Generation