1. **OBJECTIVE**

Our objective is to develop a solution for the tedious task of time table generation with more accuracy and in a less time consuming process. So now the timetable needed to schedule the faculty at provided time slots in such that their things do not overlap and the time table schedule makes best use of all faculties subject demands. Proposed system works on php and uses mysql as a database.

The idea behind the system is to create an instance of data provided and managed by the user under given circumstances..

1. **ABSTRACT**

The manual system of preparing time tables in colleges with large numbers of students is very time consuming and usually ends up with various classes clashing either in the same room or with the same teachers having more than one class at a time.

In the existing system, each task is carried out manually and processing is a very tedious job. The Organization is not able to achieve its need in time and the results too may not be accurate. Due to all the manual maintenance, there are number of difficulties and drawbacks that exist in this system

To overcome all these problems , propose to make an automated system. The system will take various inputs like details of students, subjects and class rooms and teachers available, depending upon these inputs it will generate a possible time table, making optimal utilization of all resources in a way that will best suit any of constraints or college rules. List of subjects may include electives as well as core subjects.

It is a comprehensive timetable management solution for Colleges or organizations which help to overcome the challenges in manually setting the timetable. By using this software it will be very easy for faculty to get a timetable.

1. **INTRODUCTION**

Time table scheduling has been in human requirements since all thought of managing time effectively. It is widely used in schools, and other fields of teaching and working like crash courses, coaching centers, training programs etc. In early days, time table scheduling was done manually with one single person or some group involved in task of scheduling it with their hands, which takes a lot of effort and time.While scheduling even the smallest constraints can take a lot of time and the cases are even more worse when the number of constraints or the amount of data to deal with increases. In such cases perfectly designed time table is reused for the whole generation without any changes, proving to be dull in such situations. Other cases that cause problems are when the number of employers/workers are weak, resulting in the rescheduling of the time table or they need to fill on empty seats urgently. They need to schedule their course to meet the need of current duration and facilities that are available to them. However, their schedule should meet the requirements of the new course additions and newly enrolled students to fresh batches. This may result in rescheduling the entire time table once again for its entire batches and to be scheduled in shortest possible time before the batch courses start. Another problem that occur when scheduling time table for exams. When multiple batches have their exam on same day, they need to be schedules effectively taking into account all problems related to the facilities that are available to conduct these exams simultaneously.

**3.1 Existing System -**

In the existing system, each task is carried out manually and processing is a very tedious job. The Organization is not able to achieve its need in time and the results too may not be accurate. Due to all the manual maintenance, there are a number of difficulties and drawbacks that exist in this system. Drawbacks of the Existing System:

* Increased transaction leads to the increased source document and hence maintenance becomes difficult.
* If any student or staff entry is wrongly made then the maintenance becomes very difficult.

**3.2 Proposed System** -

The proposed system is designed to be more efficient than the actual manual system. It invokes all base tasks that are now carried out manually, such as the forms transactions and reports which is an added advantage.

Description:

* Most colleges have a number of different courses and each course has ‘n’ number of subjects.
* Now there are limited faculties, and each faculty might be teaching more than one subject.
* So now the time table needed to schedule all the faculty at provided time slots in such a way that their timings do not overlap and the time table schedule will make the best use of all faculty subject demands.
* We use a customized algorithm for this purpose.
* In our Timetable Generation algorithm we propose to utilize a timetable object.
* Fitness score relates to the quantity of crashes. This object comprises Classroom objects and the timetable for them likewise a fitness score for the timetable.
* Classroom objects comprise week objects. Week objects consist of Days, Days comprises Timeslots.
* Timeslot has an address in which a subject, student gathering going to that particular address and educator showing to the subject it is related will be shown.
* Also further on discussing, we have utilized composite configuration (design), which makes it well extendable numerous obligations.
* In every obligation class the condition as determined in our inquiry is now checked between both the timetable objects. On the off chance that condition is fulfilled, there is a crash available then score is augmented by one.

**3.3 Purpose:**

Normally timetable generation is 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 change their timetable more than twice a year which puts a double workload on the timetable designer. Considering workload with thisscheduling 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.

**3.4 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 resource utilization and optimization.

**3.5 Challenges :**

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 task 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. **LITERATURE SURVEY**

Trying to develop a software which helps to generate Timetables for an Institution automatically. By looking at the existing system we can understand that timetable generation is done manually. Manually adjust the timetable when any of the faculty is absent, and this is the big challenge for Automatic Timetable Generation that manages the timetable automatically when any of the faculty is absent.

As we know all institutions/organizations have their own timetable,managing and maintaining these will not be difficult. Considering workload with this scheduling will make it more complex. As mentioned , when Timetablegeneration is being done, it should

consider the maximum and minimum workload that is in a college. In those cases timetable

generation will become more complex. Also , it is a time consuming process.

**4.1 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.

**4.2 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, MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

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.

**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.

1. **Implementation**

The Project is working on EasyPHP devserver 17. We used Visual Studio for Design and coding of the project. Created and maintained all databases into Mysql, in that we create tables, write query for store data or record of the project.

**Technologies used :**

| **Category** | **Name** | **Version** |
| --- | --- | --- |
| **1. Frontend** | **Html** | **5** |
| **2. Styling** | **Css** | **3** |
| **3. Backend** | **Php** | **6+** |
| **4. Database** | **Mysql** | **5.5+** |
| **5. Server** | **Apache** | **10** |

**CODE :**

**File : index.php**

<?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>';

}

}

?>

**File : Addteacher.php**

<?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)

)";

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>";

}

?>

**File : Addsubject.php**

<?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>";

}

?>

1. **Testing**

**6.1. White Box Testing**

White box testing is a test technique in which the internal structure or code of an application is visible and accessible to the tester. In this technique, it is easy to find loopholes in the design of an application or fault in business logic. Statement coverage and decision coverage/branch coverage are examples of white box test techniques.

**6.2. Black Box Testing**

Blackbox testing is a software testing technique in which testing is performed without knowing the internal structure, design, or code of a system under test. Testers should focus only on the input and output of test objects.

From the name itself we can probably understand that it implicates interacting with the system that you are testing as a mystery box. It means that you are not knowledgeable enough about the internal working of the system but you know how it should behave.

**6.3. Alpha Testing**

Alpha testing is a type of acceptance testing performed by the team in an organization to find as many defects as possible before releasing software to customers.

For example, the pet insurance website is under UAT. UAT team will run real-time scenarios like buying an insurance policy, buying annual membership, changing the address, ownership transfer of the pet in the same way the user uses the real website. The team can use test credit card information to process payment-related scenarios.

**6.4. Beta Testing**

Beta Testing is a type of software testing which is carried out by the clients/customers. It is performed in the Real Environment before releasing the product to the market for the actual end-users.

Beta Testing is carried out to ensure that there are no major failures in the software or product, and it satisfies the business requirements from an end-user perspective. Beta Testing is successful when the customer accepts the software.

Usually, this testing is typically done by the end-users. This is the final testing done before releasing the application for commercial purposes. Usually, the Beta version of the software or product released is limited to a certain number of users in a specific area.

So, the end-user uses the software and shares the feedback with the company. The company then takes necessary action before releasing the software worldwide.

**Test Cases :**

**User Login/Registration:**

To begin with login, users need to register by filling up basic registration details.

There are multiple fields in the registration page and every field has to fill by the user.

Users cannot use characters in the login id field.

**Admin Login: -** Admin login id and password are kept compulsory fields, and if the admin id or password doesn’t match then it will show an error message.