VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belgaum – 590018, Karnataka.

A Mini Project Report

On

"AUTOMATIC TIMETABLE GENERATOR"

Submitted in partial fulfillment of the requirement for the V semester course of

BACHELOR OF ENGINEERING

In

INFORMATION SCIENCE AND ENGINEERING

Submitted by:

HARSHITH S (1AP18IS008)

Under the guidance of:

Mrs. Nita Meshram

Asst. Professor

Dept. of IS & E

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

APS COLLEGE of ENGINEERING

Anantha Gnana Gangothri,

26th k.m, NH-209, Kanakapura Road, Bangalore-560082.

2020-2021

APS COLLEGE of ENGINEERING

Anantha Gnana Gangothri,
26th k.m, NH-209, Kanakapura Road, Bangalore-560082.

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

Certified that the Project Work entitled ***"AUTOMATIC TIMETABLE GENERATOR"*** has been punctually carried out at ***APS College of Engineering, Bangalore*** by ***HARSHITH S (1AP18IS008),*** bonafide student/s of ***Fifth Semester, B.E.*** in partial fulfillment for the award of degree in ***Bachelor of Engineering*** in ***Information Science & Engineering*** *affiliated to* ***Visvesvaraya Technological University, Belgaum*** during academic year ***2020-2021.*** It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work for the said degree.

Mrs. Nita Meshram
Asst. Professor,
Dept. of IS & E,
APSCE.

Dr. Suresh H
Assoc. Professor &
Head Dept. of IS & E,
APSCE.

Name of the external **Signature and Date**

1.

2.

DECLARATION

We hereby declare that the project work entitled **"Automatic Timetable Generator"** done at **APS COLLEGE OF ENGINEERING, Bangalore,** submitted to **Visvesvaraya Technological University,** in partial fulfilment of requirements for the degree of **Bachelor of Engineering** in **Information Science & Engineering** is a record of original work done by us and no part of it has been submitted for any degree or diploma of any institution previously.

Place:
Date:

HARSHITH S (1AP18IS008)

ACKNOWLEDGEMENT

I take immense pleasure in thanking **Dr. B M Sathish**, Principle, APSCE for having permitted us to carry out our Mini project ‘‘**Automatic Timetable Generator**”. I wish to express our deep sense of gratitude to **Dr. Suresh H**, Assoc. Professor, Head of Department of Information Science & Engineering, for his able guidance and useful suggestions, which helped us in completing this project.

I also express our deep sense of gratitude to our guides **Mrs. Nita Meshram**, Asst. Professor, Dept of ISE for their timely assistance which helped us to complete our project.

Words are inadequate in offering our thanks to all the teaching and non-teaching staff of Dept of ISE for their encouragement and co-operation in carrying out our project.

Finally, yet importantly, we would like to express our heartfelt thanks to our beloved parents for their blessing, our friends for their help and wishes for successful completion of this project.

STUDENT NAME (USN)

HARSHITH S (1AP18IS008)

NAGABUSHAN V (1AP18IS016)

ABSTRACT

A database is the single most useful environment in which to store data and an ideal tool to manage and manipulate that data. The benefits of a well-structured database are infinite, with increased efficiency and time-saving benefits. Our team's interest is centered on this area. At the very start, we create a database Wedlock System. We use MySQL Software for this purpose. We determine attributes and entities and figure out relationship among entities. Then we draw the entity-relationship diagram, covert it to a relational model (relational tables) and trigger the tables. We implement the design, create tables and insert values inside the tables using the tables using MYSQL software. We execute sample queries on the system and verify that our system contains all required information making retrieval of the information fast and efficient.

Time Table Generator is a web-based application which guides you about time table management System.

Even though most college administrative work has been computerized, the lecture timetable scheduling is still mostly done manually due to its inherent difficulties.

The manual lecture-timetable scheduling demands considerable time and efforts. The lecture-timetable scheduling is a Constraint satisfaction problem in which we find a solution that satisfies the given set of constraints. A college timetable is a temporal arrangement of a set of lectures and classrooms in which all given constraints are satisfied.

Creating such timetables manually is complex and time-consuming process. By automating this process with computer assisted timetable generator can save a lot of precious time of administrators who are involved in creating and managing course timetables. Since every college has its own timetabling problem, the commercially available software packages may not suit the need of every college.

Hence, we have developed practical approach for building lecture course timetabling system, which can be customized to fit to any colleges timetabling problem. The college lecture-timetabling problem asks us to find some time slots and classrooms which satisfy the constraints imposed on offered. The proposed system is a website, which allows the student a good user interface also it provides a good user interface to admin & faculties, and they can easily get the required information. The web site provides a variety of facilities to students, admin and faculties.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 DATABASE MANAGEMENT SYSTEM

1.1.1 INTRODUCTION TO DBMS

A database is simply an organized collection of related data, typically stored on disk, and accessible by possibly many concurrent users. Databases are generally separated into application areas. For example, one database may contain Human Resource (employee and payroll) data; another may contain sales data; another may contain accounting data; and so on. Databases are managed by a DBMS. Many Database Systems are being used which are in turn managed by many other Database Management Systems. A Database Management System (DBMS) is a set of programs that manages any number of databases.

Basically, DBMS is a software tool to organize (create, retrieve, update and manage) data in a database. The main aim of a DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have embedded meaning. Database systems are meant to handle large collection of information. Management of data involves both defining structures for storage of information and providing mechanisms that can do the manipulation of those stored information. Moreover, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.

1.1.2 HISTORY OF DBMS

The sizes, capabilities, and performance of databases and their respective DBMSs have grown in orders of magnitude. These performance increases were enabled by the technology progress in the areas of processors, computer memory, computer storage, and computer networks. The concept of a database was made possible by the emergence of direct access storage media such as magnetic disks, which became widely available in the mid-1960s; earlier systems relied on sequential storage of data on magnetic tape. The

subsequent development of database technology can be divided into three eras based on data model or structure: navigational, SQL/relational, and post-relational.

The two main early navigational data models were the hierarchical model and the CODASYL model (network model). These were characterized by the use of pointers (often physical disk addresses) to follow relationships from one record to another. The relational model, first proposed in 1970 by Edgar F. Codd, departed from this tradition by insisting that applications should search for data by content, rather than by following links. The relational model employs set of ledger-style tables, each used for a different type of entity. Only in the mid-1980s did computing hardware become powerful enough to allow the wide deployment of relational systems (DBMSs plus applications). By the early 1990s, however, relational systems dominated in all large-scale data processing applications, and as of 2018 they remain dominant: IBM DB2, Oracle, MySQL, and Microsoft SQL Server are the most searched DBMS. The dominant database language, standardized SQL for the relational model, has influenced database languages for other data models.

Object databases were developed in the 1980s to overcome the inconvenience of object-relational impedance mismatch, which led to the coining of the term "post-relational" and also the development of hybrid object-relational databases. The next generation of post-relational databases in the late 2000s became known as NoSQL databases, introducing fast key-value stores and document-oriented databases. A competing "next generation" known as NewSQL databases attempted new implementations that retained the relational/SQL model while aiming to match the high performance of NoSQL compared to commercially available relational DBMSs.

1.1.3 APPLICATION OF DBMS

There are various fields where a database management system is used. Following are some applications which make use of the database management system:

- Railway Reservation System
- Library Management System
- Banking
- Education Sector
- Credit card transactions
- Social Media Sites
- Telecommunications
- Online Shopping
- Human Resource Management

1.1.4 ADVANTAGES OF DBMS

1. Improved data sharing

An advantage of the database management approach is, the DBMS helps to create an environment in which end users have better access to more and better-managed data. Such access makes it possible for end users to respond quickly to changes in their environment.

2. Improved data security

The more users access the data, the greater the risks of data security breaches. Corporations invest considerable amounts of time, effort, and money to ensure that corporate data are used properly. A DBMS provides a framework for better enforcement of data privacy and security policies.

3. Better data integration

Wider access to well-managed data promotes an integrated view of the organization's operations and a clearer view of the big picture. It becomes much easier to see how actions in one segment of the company affect other segments.

4. Minimized data inconsistency

Data inconsistency exists when different versions of the same data appear in different places. For example, data inconsistency exists when a company's sales department stores a sales representative's name as "Bill Brown" and the company's personnel department stores that same person's name as "William G. Brown," or when the company's regional sales office shows the price of a product as $45.95 and its national sales office shows the same product's price as $43.95. The probability of data inconsistency is greatly reduced in a properly designed database.

5. Improved decision making

Better-managed data and improved data access make it possible to generate better-quality information, on which better decisions are based. The quality of the information generated depends on the quality of the underlying data. Data quality is a comprehensive approach to promoting the accuracy, validity, and timeliness of the data. While the DBMS does not guarantee data quality, it provides a framework to facilitate data quality initiatives.

1.1.5 DISADVANTAGES OF DBMS

1. Increased costs

One of the disadvantages of DBMS is Database systems require sophisticated hardware and software and highly skilled personnel. The cost of maintaining the hardware, software, and personnel required to operate and manage a database system can be substantial. Training, licensing, and regulation compliance costs are often overlooked when database systems are implemented.

2. Management complexity

Database systems interface with many different technologies and have a significant impact on a company's resources and culture. The changes introduced by the adoption of a database system must be properly managed to ensure that they help advance the company's objectives. Given the fact that database systems hold crucial company data that are accessed from multiple sources, security issues must be assessed constantly.

3. Maintaining currency

To maximize the efficiency of the database system, you must keep your system current. Therefore, you must perform frequent updates and apply the latest patches and security measures to all components. Because database technology advances rapidly, personnel training costs tend to be significant. Vendor dependence. Given the heavy investment in technology and personnel training, companies might be reluctant to change database vendors. As a consequence, vendors are less likely to offer pricing point advantages to existing customers, and those customers might be limited in their choice of database system components.

4. Frequent upgrade/replacement cycles

DBMS vendors frequently upgrade their products by adding new functionality. Such new features often come bundled in new upgrade versions of the software. Some of these versions require hardware upgrades. Not only do the upgrades themselves cost money, but it also costs money to train database users and administrators to properly use and manage the new features.

1.2 PROJECT

1.2.1 OVERVIEW OF PROJECT

Automatic Timetable Generator is developed using PHP, CSS, Bootstrap, and JavaScript. Talking about the project, it contains an admin side from where a user can manage all the timetables and records easily. The admin plays an important role in the management of this system. In this project, the user has to perform all the main functions from the admin side. Even it contains teacher side and student side from where they can login and easily access the timetable.

1.2.2 FEATURES OF PROJECT

- Admin Panel
- User management system
- Manage teachers, course/year/section
- Manage subjects, rooms and school year
- View reports
- Maintain class schedule
- View archive

1.2.3 OBJECTIVE AND SCOPE OF PROJECT

OBJECTIVE

- To reduce a time required for generating time table than existing system.
- To generate performance graph for each faculty according to feedback given by student.
- To increase efficiency and accuracy of proposed system.
- To generate defaulter list of students and send the e-mail to defaulter students.
- To help teachers to maintain attendance record of students.
- To reduce paper and labor work.

SCOPE

- Automatic timetable generation by considering hard and soft constraints.
- Feedback generation by using Mashape API.
- Generate defaulter list from attendance sheet.

1.3 MySQL

SQL is a standard language for storing, manipulating and retrieving data in databases. Originally based upon relational algebra and tuple relational calculus, SQL consists of a data definition language, data manipulation language, and data control language. The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data access control. SQL became a standard of the American National Standards Institute (ANSI) in 1986 and of the International Organization for Standardization (ISO) in 1987. Since then, the standard has been revised to include a larger set of features. Despite the existence of such standards, most SQL code is not completely portable among different database systems without adjustments.

1.4 HTML/PHP/CSS/JavaScript

HTML

HTML is a markup language used for structuring and presenting content on the web and the fifth current major version of HTML standard. HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications.

PHP

PHP is a general-purpose scripting language especially suited to web development. The PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive initialize PHP: Hypertext Preprocessor. Fast, flexible and pragmatic, PHP powers everything from your blog to the most popular websites in the world.

CSS

Cascading Style Sheets (CSS) is a simple mechanism for adding style (e.g., fonts, colors and spacing) to Web documents. These pages contain information on how to learn and use CSS and on available software. They also contain news from the CSS working group.

JavaScript

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it.

CHAPTER 2

REQUIREMENTS SPECIFICATION

2.1 OVERALL DESCRIPTION

A reliable and scalable database driven web application with security features that is easy to use and maintain is the requisite.

2.2 SPECIFICATION REQUIREMENTS

The Specific Requirements for **"Automatic Timetable Generator"** is stated as follows:

2.2.1 SOFTWARE REQUIREMENTS

- Operating system: Windows 10
- Front end design: HTML, CSS, JavaScript, PHP
- Back-end design: MySQL
- XAMPP Server
- Visual Studio Code
- Google Chrome

2.2.2 HARDWARE REQUIREMENTS

- Processor: AMD Ryzen 5 3500U
- RAM: 8GB
- Hard disk: 1TB

CHAPTER 3

DETAILED DESIGN

3.1 SYSTEM DESIGN

The web server needs a JSP engine, i.e., a container to process JSP pages. The JSP container is responsible for intercepting requests for JSP pages. A JSP container works with the Web server to provide the runtime environment and other services a JSP needs. It knows how to understand the special elements that are part of JSPs. This server will act as a mediator between the client browser and a database.

The following diagram shows the JSP architecture.

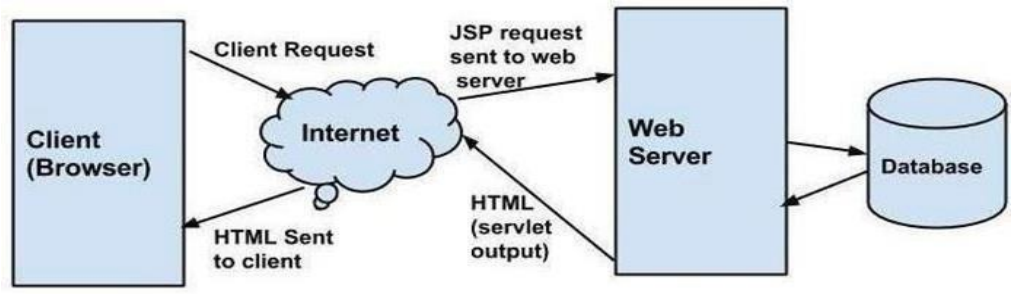


Fig. 3.1: JSP Architecture

Three-tier Client/Server database architecture is commonly used architecture for web applications. Intermediate layer called Application server or Web Server stores the web connectivity software and the business logic (constraints) part of application used to access the right amount of data from the database server. This layer acts like medium for sending partially processed data between the database server and the client. Database architecture focuses on the design, development, implementation and maintenance of computer programs that store and organize information for businesses, agencies and institutions. A database architect develops and implements software to meet the needs of users. Several types of databases, including relational or multimedia, may be created. Additionally, database architects may use one of several languages to create databases, such as structured query language.

3.2 ENTITY RELATIONSHIP DIAGRAM

An entity–relationship model is usually the result of systematic analysis to define and describe what is important to processes in an area of a business. An E-R model does not define the business processes; it only presents a business data schema in graphical form. It is usually drawn in a graphical form as boxes (entities) that are connected by lines (relationships) which express the associations and dependencies between entities. Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys". Diagrams created to represent attributes as well as entities and relationships may be called entity-attribute-relationship diagrams, rather than entity-relationship models.

An ER model is typically implemented as a database. In a simple relational database implementation, each row of a table represents one instance of an entity type, and each field in a table represents an attribute type. In a relational database a relationship between entities is implemented by storing the primary key of one entity as a pointer or "foreign key" in the table of another entity. There is a tradition for ER/data models to be built at two or three levels of abstraction. Note that the conceptual-logical-physical hierarchy below is used in other kinds of specification, and is different from the three-schema approach to software engineering. While useful for organizing data that can be represented by a relational structure, an entity-relationship diagram can't sufficiently represent semi-structured or unstructured data, and an ER Diagram is unlikely to be helpful on its own in integrating data into pre-existing information system. Cardinality notations define the attributes of the relationship between the entities. Cardinalities can denote that an entity is optional.

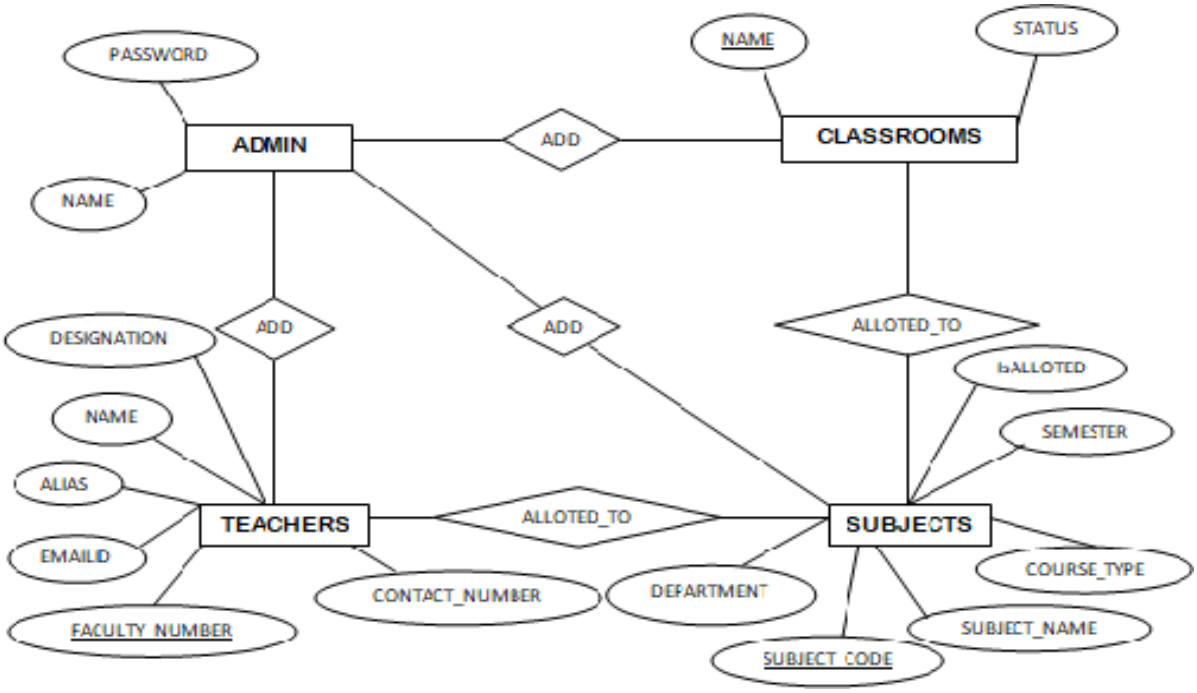


Fig. 3.2: Enhanced ER diagram of Automatic Timetable Generator

3.3 RELATIONAL SCHEMA

The term "schema" refers to the organization of data as a blueprint of how the database is constructed. The formal definition of a database schema is a set of formulas called integrity constraints imposed on a database. A relational schema shows references among fields in the database. When a primary key is referenced in another table in the database, it is called a foreign key. This is denoted by an arrow with the head pointing at the referenced key attribute. A schema diagram helps organize values in the database. The following diagram shows the schema diagram for the database.

ADMIN

NAME	PASSWORD

CLASSROOMS

NAME	STATUS

TEACHERS

FACULTY_NUMBER	NAME ALIAS	DESIGNATION	CONTACT_NUMBER	EMAILID

SUBJECTS

SUBJECT_CODE	SUBJECT_NAME	COURSE_TYPE	SEMESTER	DEPARTMENT	ISALLOTED	ALLOTED_TO

Fig. 3.3 (a): Relational Schema

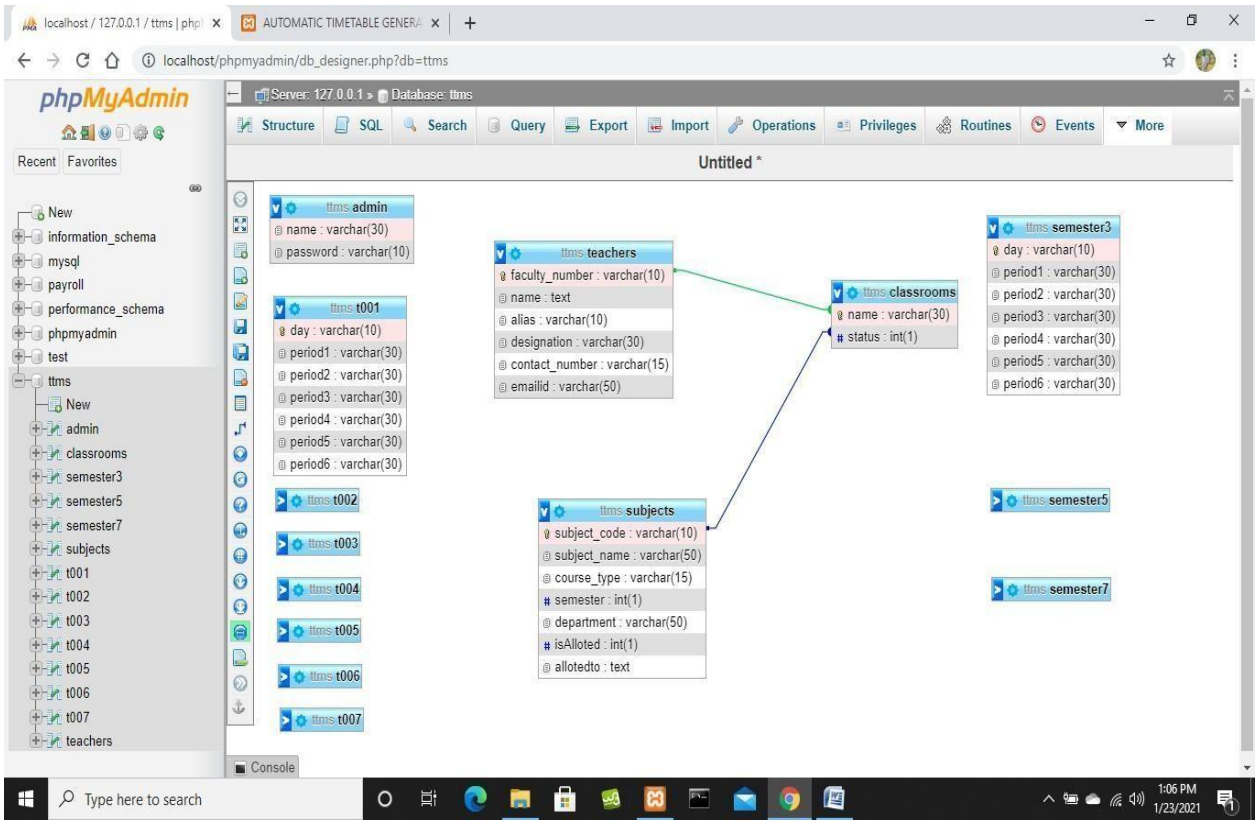


Fig. 3.3 (b): Relational Schema

3.4 SYSTEM FLOW DIAGRAM

Visual representations sometimes help people understand concepts better. A system flow diagram is a way to show relationships between a business and its components, such as customers (according to IT Toolbox.) System flow diagrams, also known as process flow diagrams or data flow diagrams are cousins to common flow charts. In a system flow diagram, the goal is to present a visual representation of some component of the business model, such as a standard customer/clerk transaction at a sandwich shop window.

Draw a circle to represent the external entity dealing with the business. Draw a rectangle directly across from the oval. This represents the entity in the business that interacts with the external entity dealing with the business. Connect the oval and the rectangle with arrows. Label the arrows with the actions or interactions taking place between the two components. Describe what the business component does when a customer interacts with the business. Write these descriptions within the box containing the business component label. Draw a second rectangle directly across from the business component box. This box represents the result of the interaction between the external entity and the business component. Add more boxes and connect the boxes with arrows to flesh out the business component you are describing. System flow diagrams can be simple or complex, depending on the business component being described.

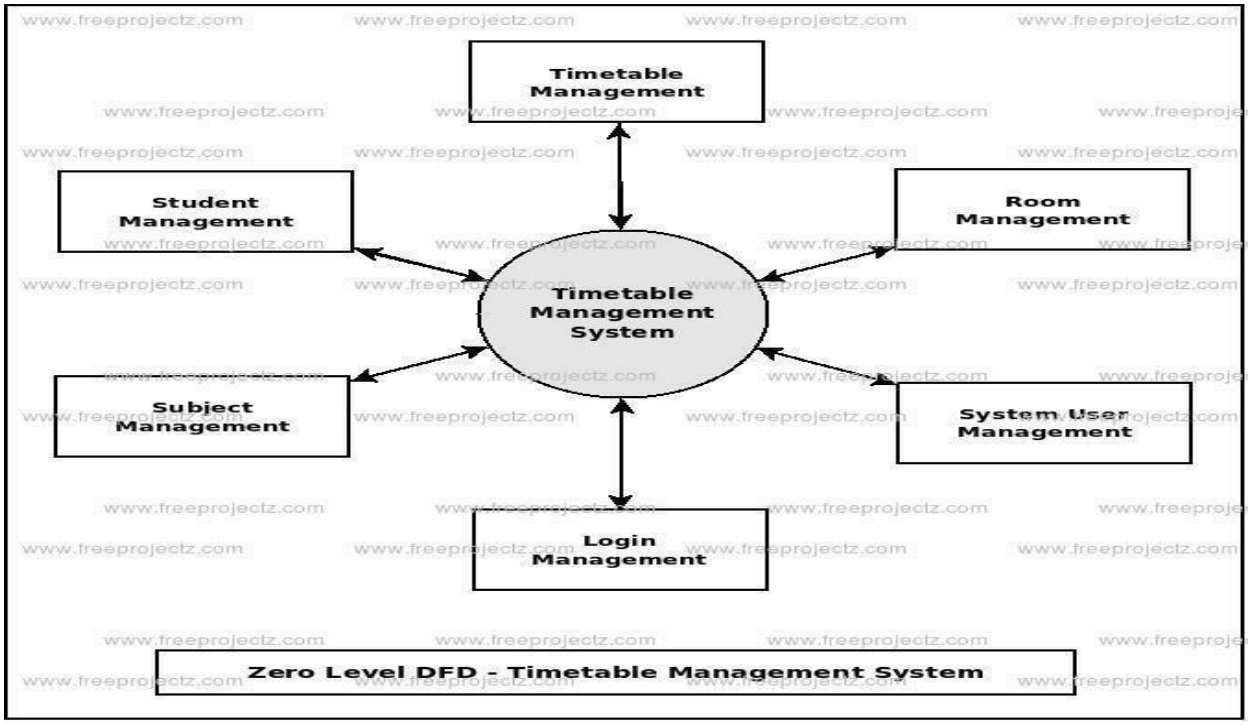


Fig. 3.4 (a): Zero Level DFD

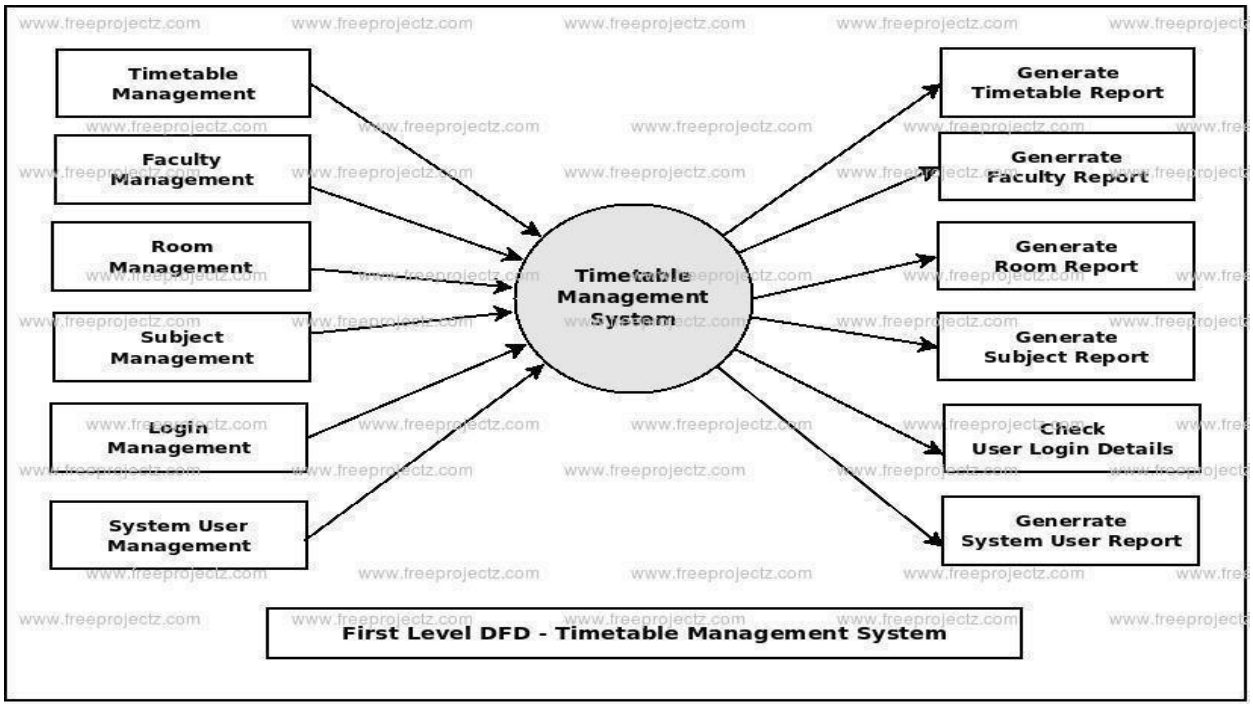


Fig. 3.4 (b): First Level DFD

CHAPTER 4

IMPLEMENTATION

4.1 INTRODUCTION TO SOFTWARE USED

The software used for project are:

- XAMPP Server
- phpMyAdmin
- Visual Studio Code

4.1.1 XAMPP Server

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

4.1.2 phpMyAdmin

phpMyAdmin is a free and open-source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it has become one of the most popular MySQL administration tools, especially for web hosting services.

4.1.3 Visual Studio Code

Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

4.2 SOURCE CODE

HTML

```
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8"/>

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1"/>

<meta name="description" content=""/>

<meta name="author" content=""/>

<title>AUTOMATIC TIMETABLE GENERATOR</title>

<!-- BOOTSTRAP CORE STYLE CSS -->

<link href="assets/css/bootstrap.css" rel="stylesheet"/>

<!-- FONT AWESOME CSS -->

<link href="assets/css/font-awesome.min.css" rel="stylesheet"/>

<!-- FLEXSLIDER CSS -->

<link href="assets/css/flexslider.css" rel="stylesheet"/>

<!-- CUSTOM STYLE CSS -->

<link href="assets/css/style.css" rel="stylesheet"/>

<!-- Google Fonts -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans:400,700,300'

rel='stylesheet' type='text/css'/>


</head>

<body>

<div class="navbar navbar-inverse navbar-fixed-top " id="menu">

<div class="container">
```

```
<div align="center">
<h3 align="center">AUTOMATIC TIMETABLE GENERATOR</h3>
</div>
</div>
</div>

<div id="myCarousel" class="carousel slide" data-ride="carousel">
<!-- Indicators -->
<ol class="carousel-indicators" style="margin-bottom: 160px">
<li data-target="#myCarousel" data-slide-to="0" class="active"></li>
<li data-target="#myCarousel" data-slide-to="1"></li>
<li data-target="#myCarousel" data-slide-to="2"></li>
<li data-target="#myCarousel" data-slide-to="3"></li>
</ol>

<!-- Wrapper for slides -->
<div class="carousel-inner" role="listbox">
<div class="item active">
<img src="assets/img/gate.jpeg" alt="Chania">
</div>

<div class="item">
<img src="assets/img/admin.jpg" alt="Chania">
</div>
```

PHP

```
<?php
if (isset($_POST['submit1'])) {
$user = $_POST['user'];
$senderid = $_POST['senderid'];
$channel = $_POST['channel'];
$DCS = $_POST['DCS'];
$flashsms = $_POST['flashsms'];
$number = $_POST['number'];
$message = $_POST['message'];
$route = $_POST['route'];

$ch = curl_init('http://login.smsgatewayhub.com/api/mt/SendSMS?APIKey=' .
$_POST['user']
. '&senderid=' . $_POST
['senderid'] . '&channel=' . $_POST['channel'] . '&DCS=' . $_POST['number'] .
'&flashsms=' .
$_POST['flashsms'] . '&numb
er=' . $_POST['number'] . '&text=' . $_POST['message'] . '&route=' . $_POST['route'] .
';'); curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($ch, CURLOPT_POST, 1); curl_setopt($ch, CURLOPT_POSTFIELDS, "");
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 2);
$data = curl_exec($ch); print($data); /* result of API call*/
}

?>
```

CSS

```
body {
font-family: 'Open Sans', sans-serif; line-height: 30px; font-size: 14px;
}

.set-pad {
padding-top: 35px;
}

.set-row-pad { padding-top: 30px; padding-bottom: 50px;
}

.logo-custom { max-height: 40px;
}

#menu {
background-color: rgba(28, 43, 75, 1); color: #fff;
font-size: 12px; font-weight: 900; letter-spacing: 1px; min-height: 70px;
}

.navbar-inverse {
border-color: transparent;
}

.navbar-inverse .navbar-nav > li > a, .navbar-inverse .navbar-nav > li > a:hover { color:
#FFF; padding: 22px 10px;
}
```

JavaScript

```
(function ($) {

//FlexSlider: Object Instance
$.flexslider = function (el, options) { var slider = $(el);

// making variables public
slider.vars = $.extend({}, $.flexslider.defaults, options);

var namespace = slider.vars.namespace,
msGesture = window.navigator && window.navigator.msPointerEnabled &&
window.MSGesture, touch = (( "ontouchstart" in window ) || msGesture ||
window.DocumentTouch && document
instanceof DocumentTouch) && slider.vars.touch,
// depricating this idea, as devices are being released with both of these events
//eventType = (touch) ? "touchend" : "click", eventType = "click touchend
MSPointerUp", watchedEvent = "", watchedEventClearTimer,
vertical = slider.vars.direction === "vertical", reverse = slider.vars.reverse,
carousel = (slider.vars.itemWidth > 0), fade = slider.vars.animation === "fade", asNav =
slider.vars.asNavFor !== "", methods = {},
focused = true;

// Store a reference to the slider object
$.data(el, "flexslider", slider);
```

4.3 PROBLEM DESCRIPTION

Automatic Timetable Generator, this system contains an admin side from where a user can manage all the timetables and records easily. The admin plays an important role in the management of this system. In this project, the user has to perform all the main functions from the admin side. Even it contains teacher side and student side from where they can login and easily access the timetable.

The tables used are:

- **For Admin Login**
- **For Classrooms**
- **For Teachers Login**
- **For Subjects**

The table details are as follows:

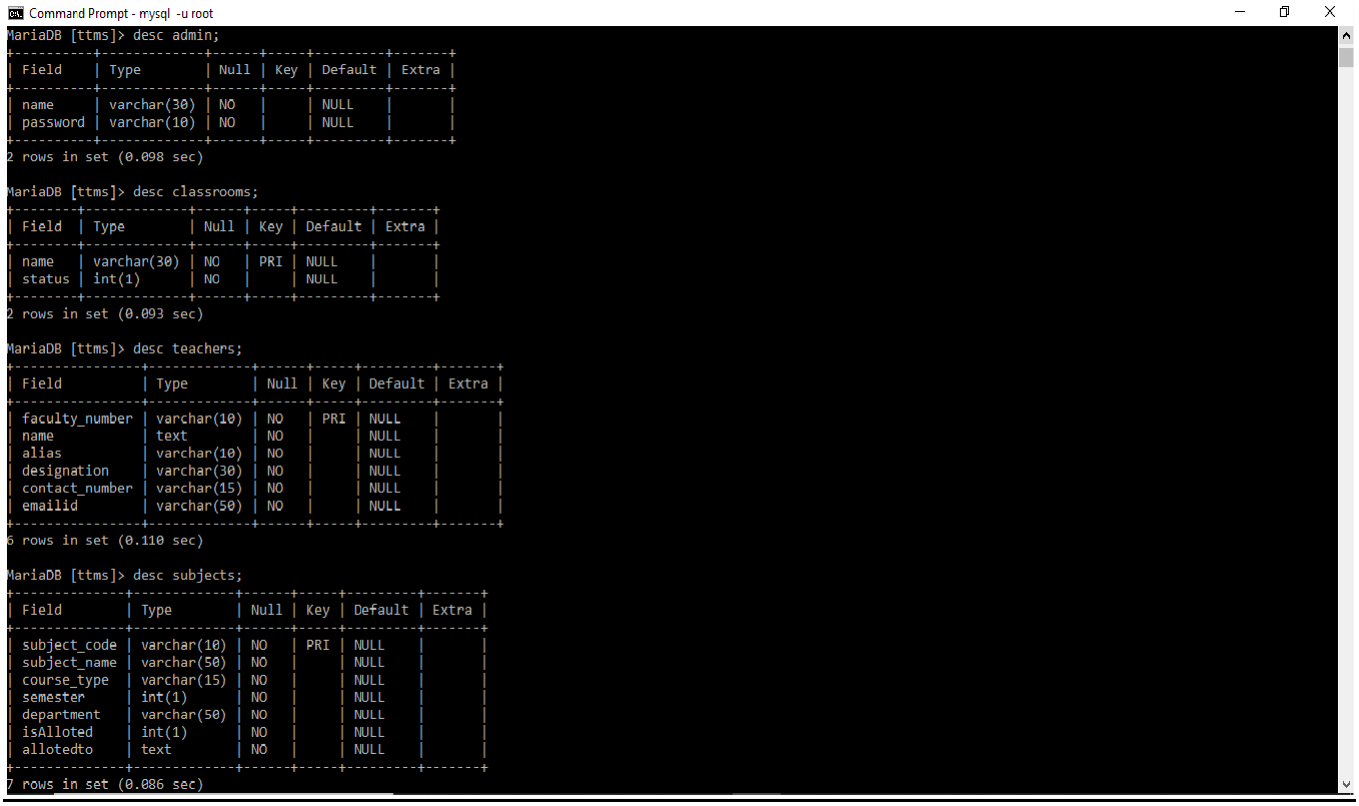


Table. 4.3: Problem Description

4.4 RESULT

The resulting system is able to:

- To add teachers, subjects
- To allot subjects and teachers with respect to classrooms
- To generate the timetable
- To archive the data

CHAPTER 5

TESTING

5.1 SOFTWARE TESTING

Testing is the process used to help identify correctness, completeness, security and quality of developed software. This includes executing a program with the intent of finding errors. It is important to distinguish between faults and failures. Software testing can provide objective, independent information about the quality of software and risk of its failure to users or sponsors. It can be conducted as soon as executable software (even if partially complete) exists. Most testing occurs after system requirements have been defined and then implemented in testable programs.

5.2 MODULE TESTING AND INTEGRATION

Module testing is a process of testing the individual subprograms, subroutines, classes, or procedures in a program. Instead of testing whole software program at once, module testing recommends testing the smaller building blocks of the program. It is largely white box oriented. The objective of doing Module testing is not to demonstrate proper functioning of the module but to demonstrate the presence of an error in the module.

Module testing allows implementing of parallelism into the testing process by giving the opportunity to test multiple modules simultaneously. The final integrated system too has been tested for various test cases such as duplicate entries and type mismatch.

CHAPTER 6

SNAPSHOTS

6.1 HOME PAGE

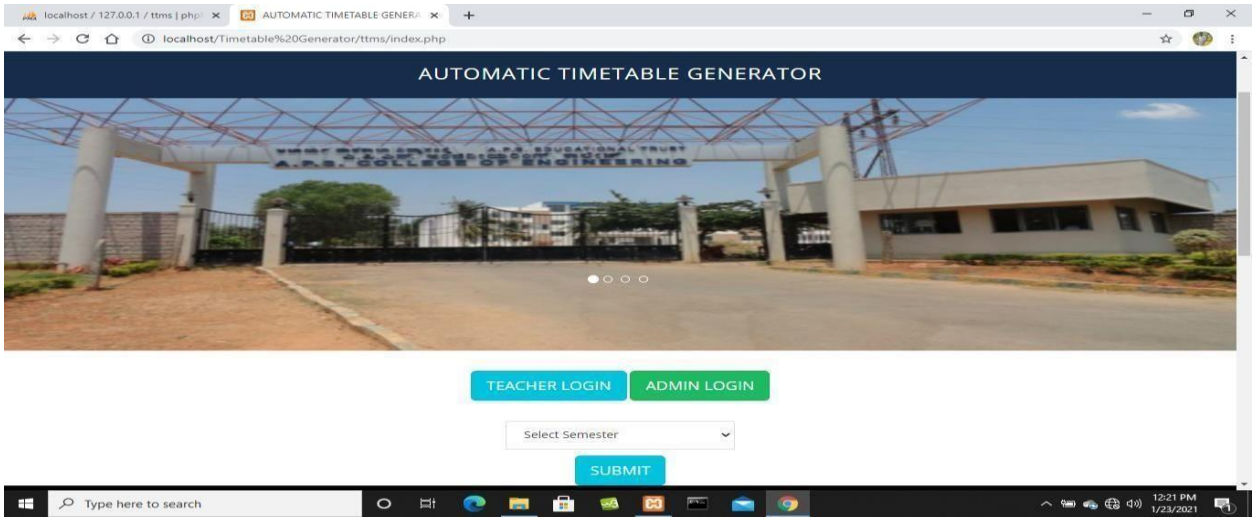


Fig. 6.1: Home Page

6.2 ADMIN LOGIN PAGE

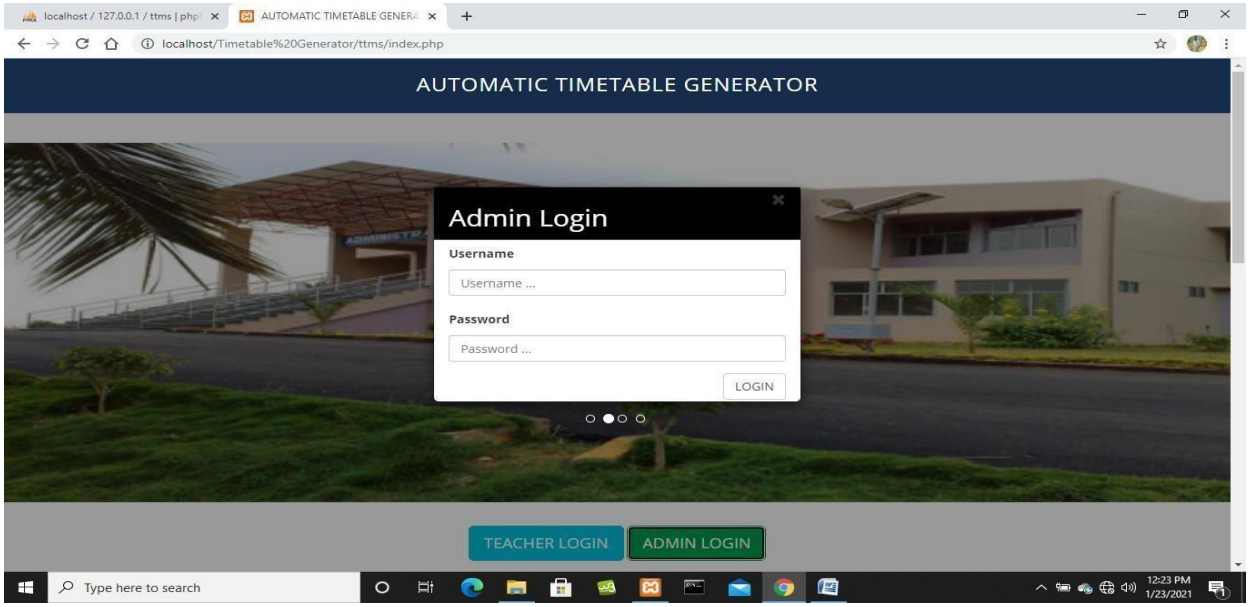


Fig. 6.2: Admin Login Page

6.3 ADD TEACHERS

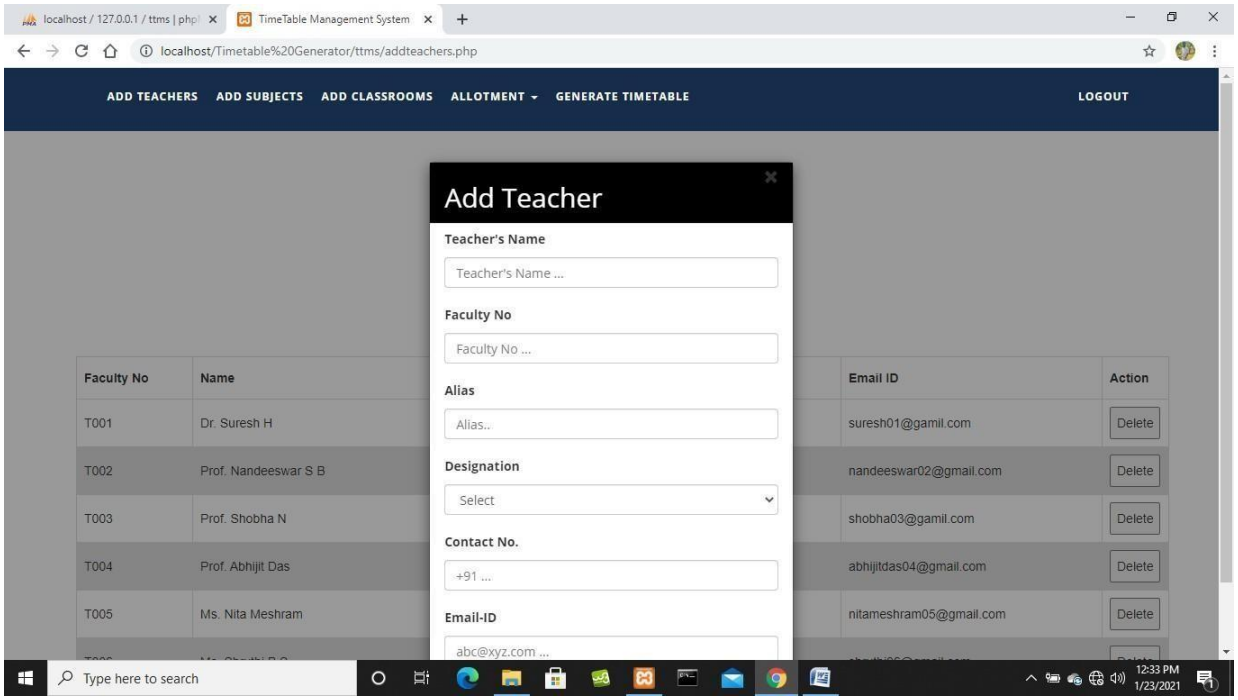


Fig. 6.3: Add Teachers

6.4 TEACHERS DETAILS

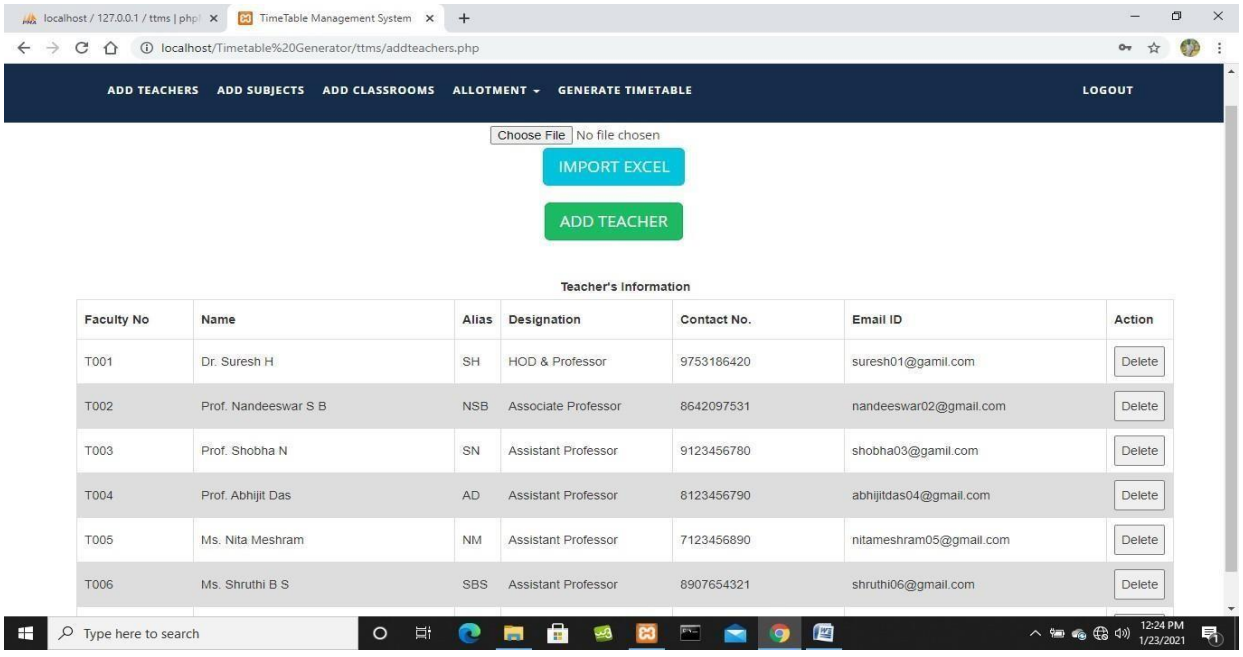


Fig. 6.4: Teachers Details

6.5 ADD SUBJECTS

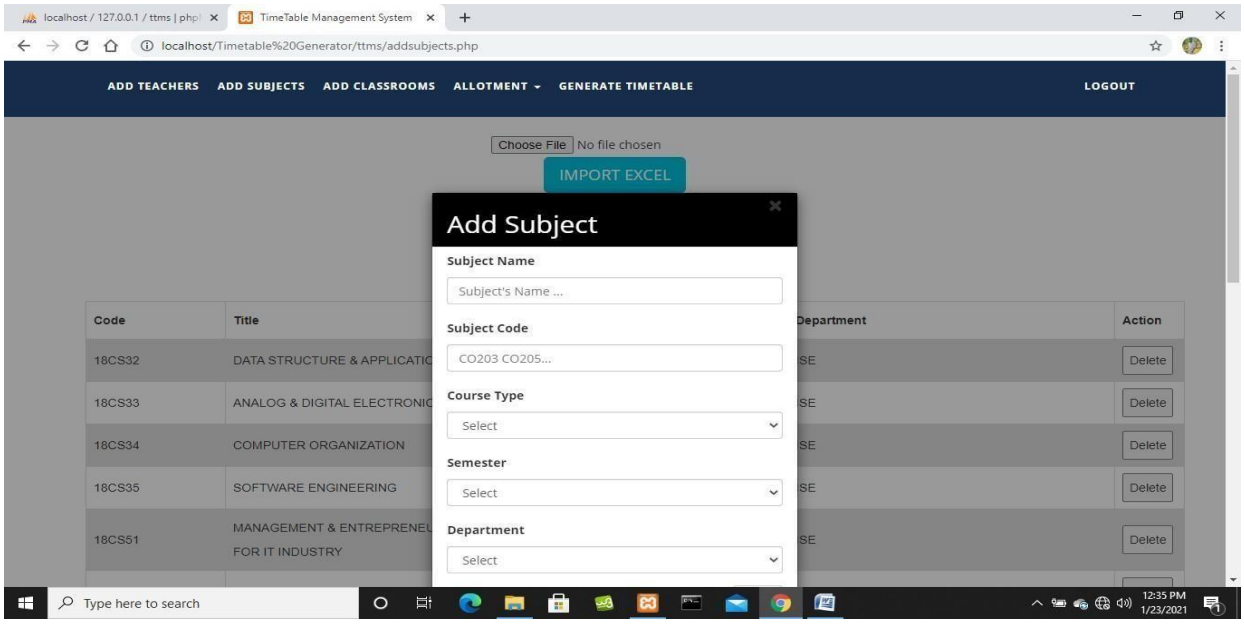


Fig. 6.5: Add Subjects

6.6 SUBJECTS DETAILS

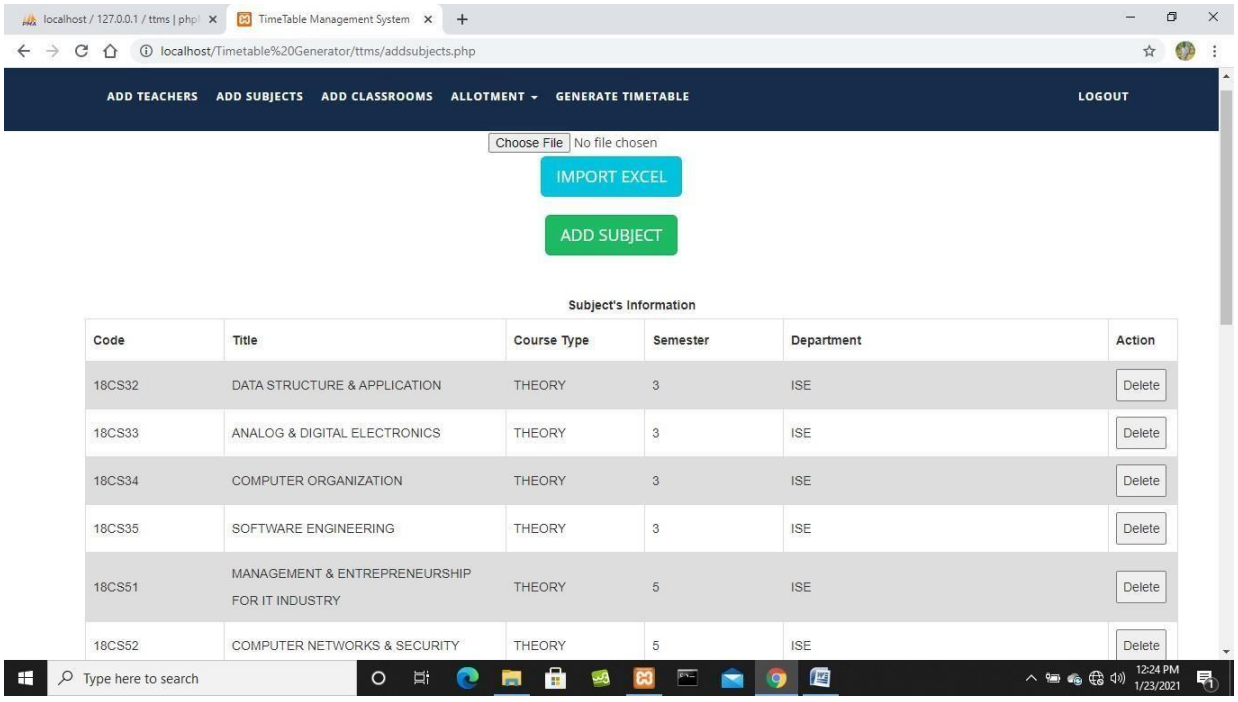


Fig. 6.6: Subjects Details

6.7 ADD CLASSROOMS

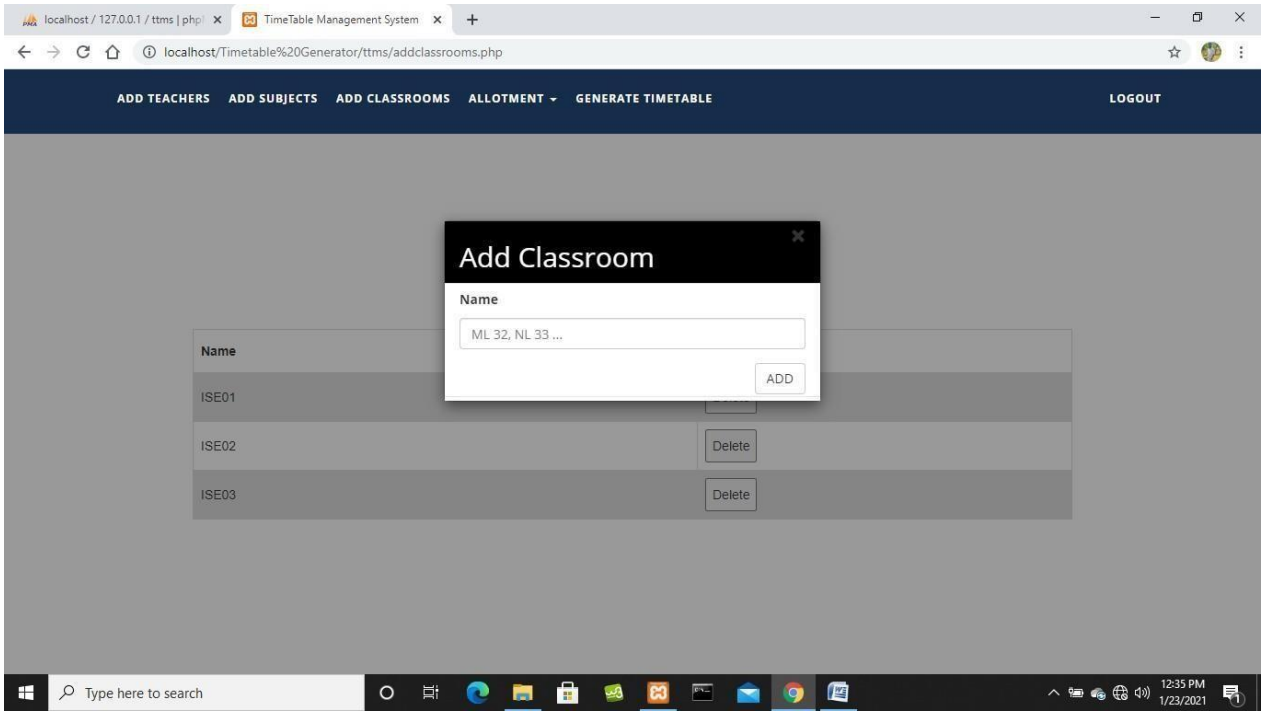


Fig. 6.7: Add Classrooms

6.8 CLASSROOMS DETAILS

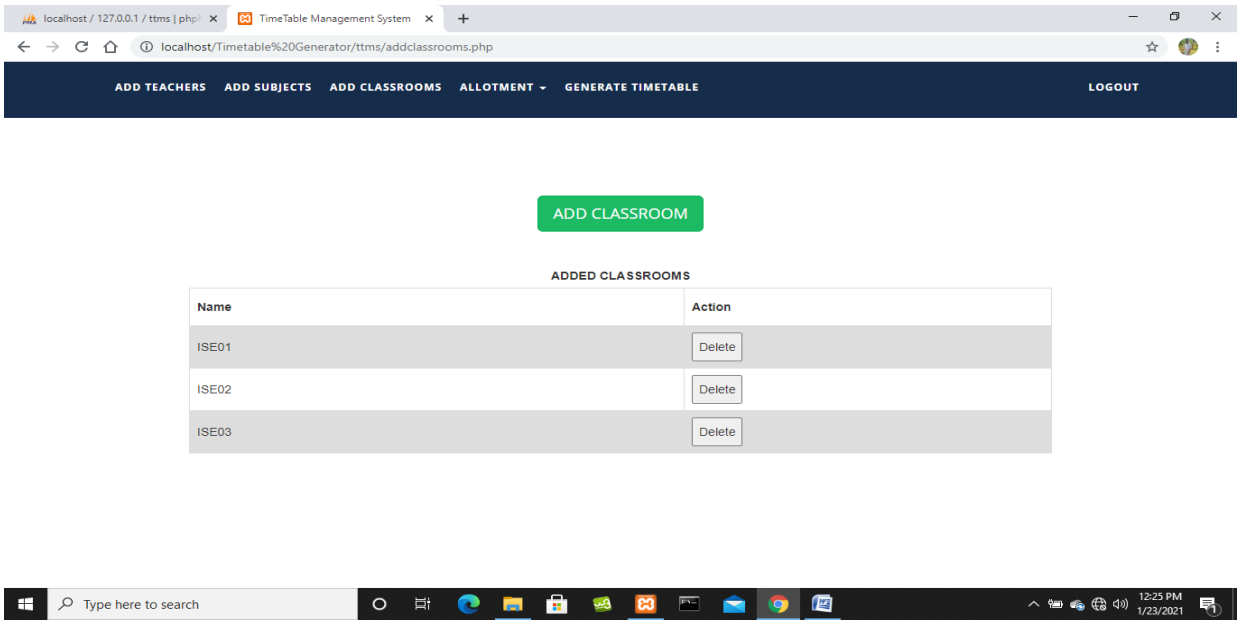


Fig. 6.8: Classrooms Details

6.9 ALLOT THEORY SUBJECTS

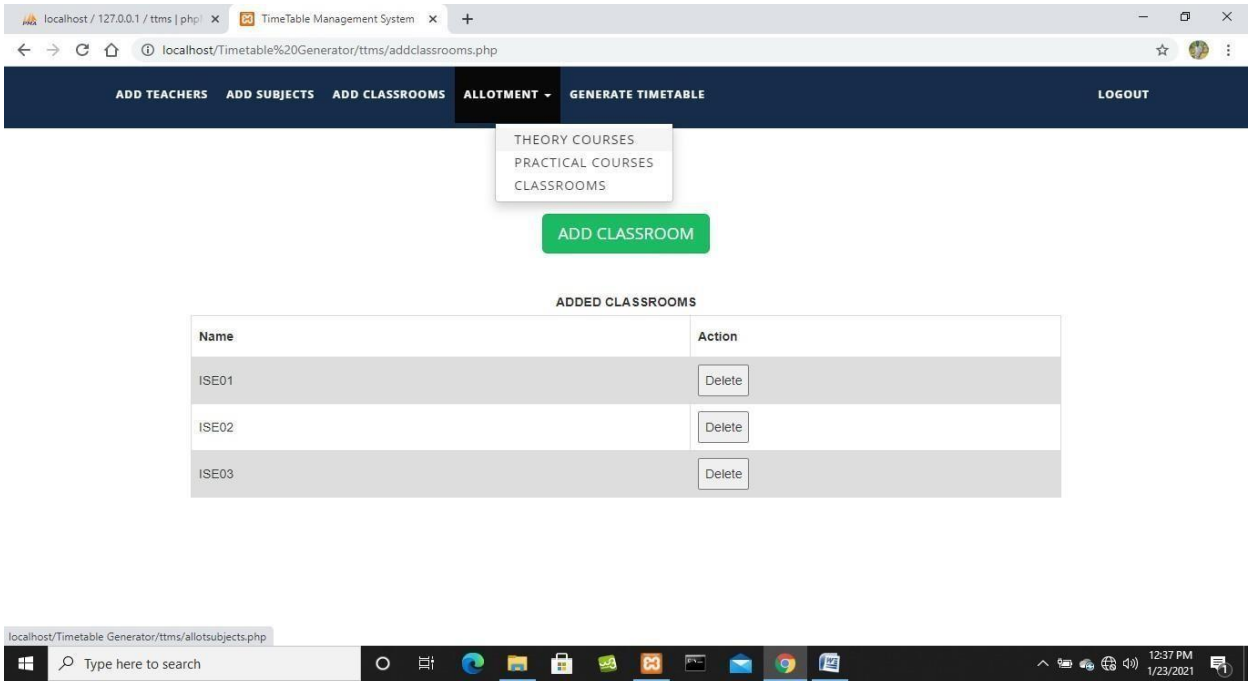


Fig. 6.9: Allot Theory Subjects

6.10 THEORY SUBJECTS DETAILS ALLOTTED

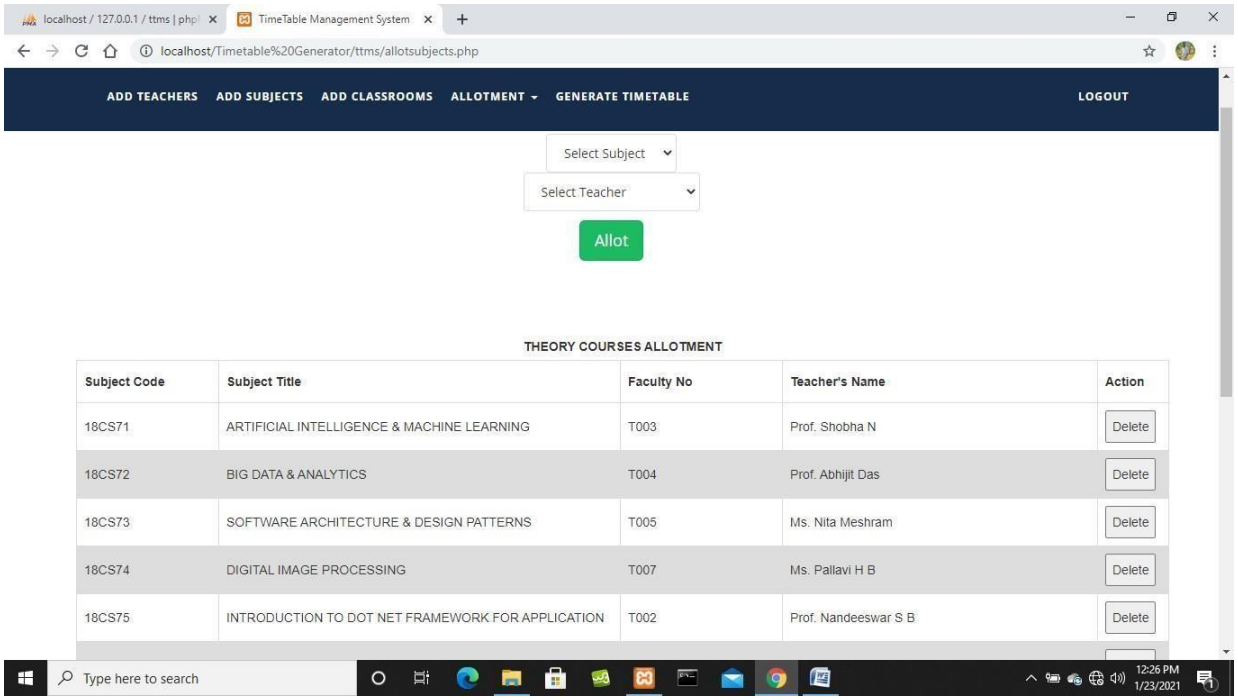


Fig. 6.10: Theory Subjects Details Allotted

6.11 ALLOT PRACTICAL SUBJECTS

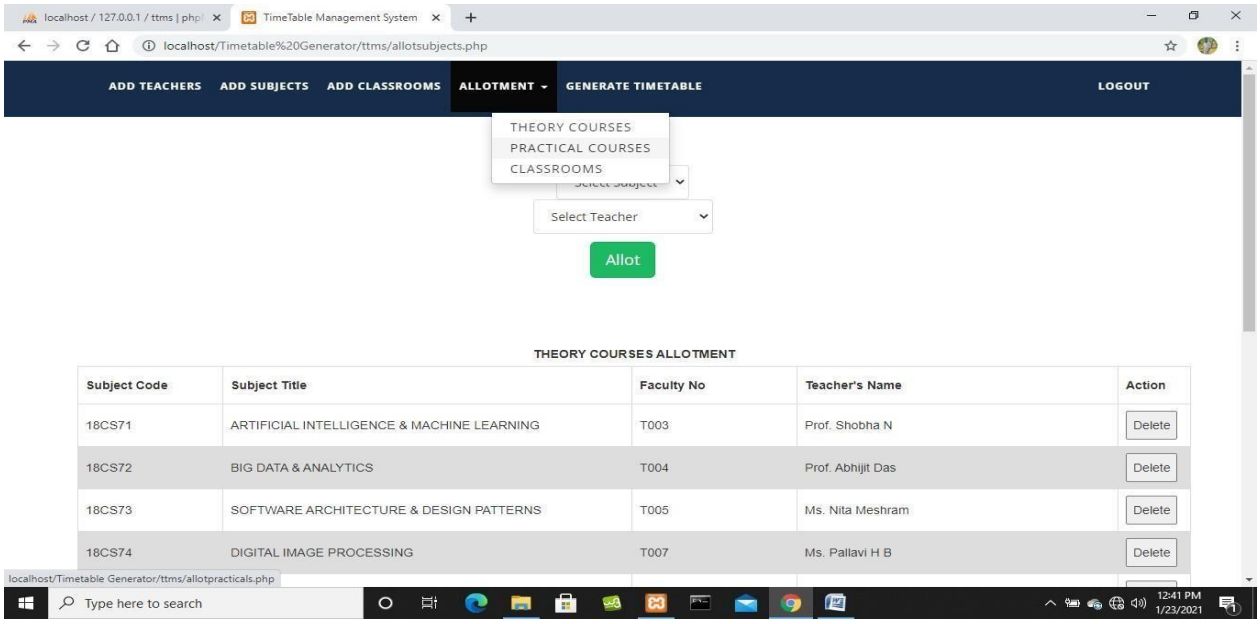


Fig. 6.11: Allot Practical Subjects

6.12 PRACTICAL SUBJECTS DETAILS ALLOTTED

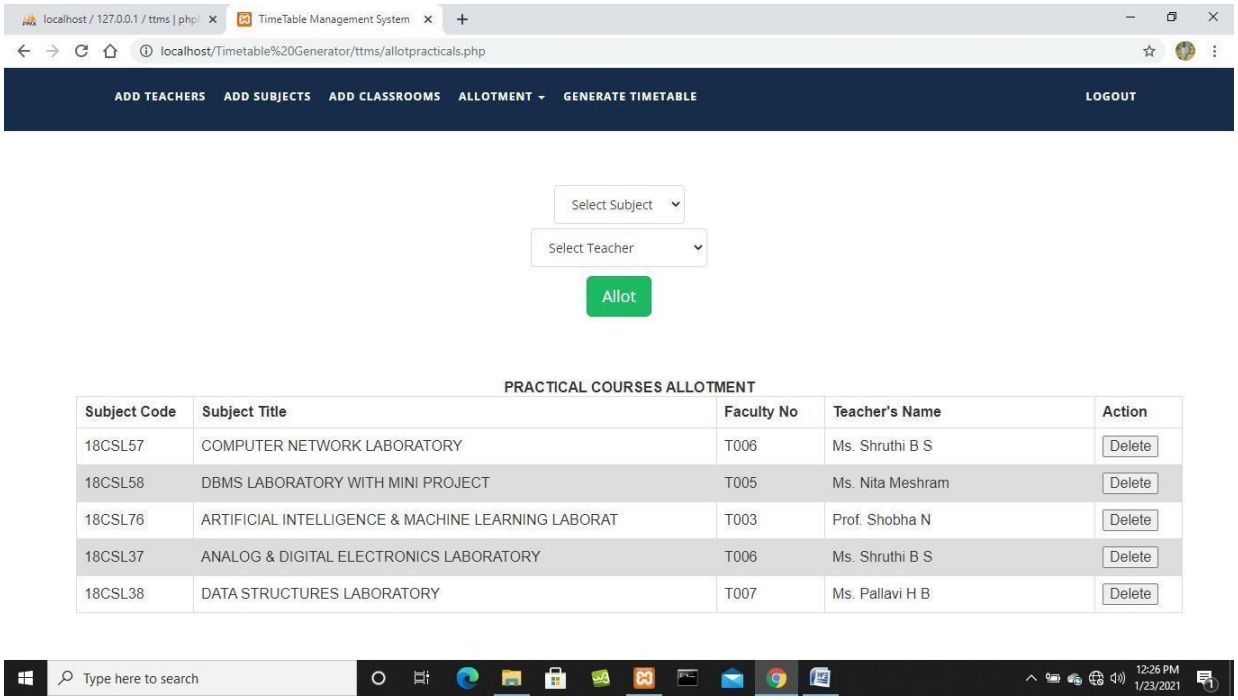


Fig. 6.12: Practical Subjects Details Allotted

6.13 ALLOT CLASSROOMS

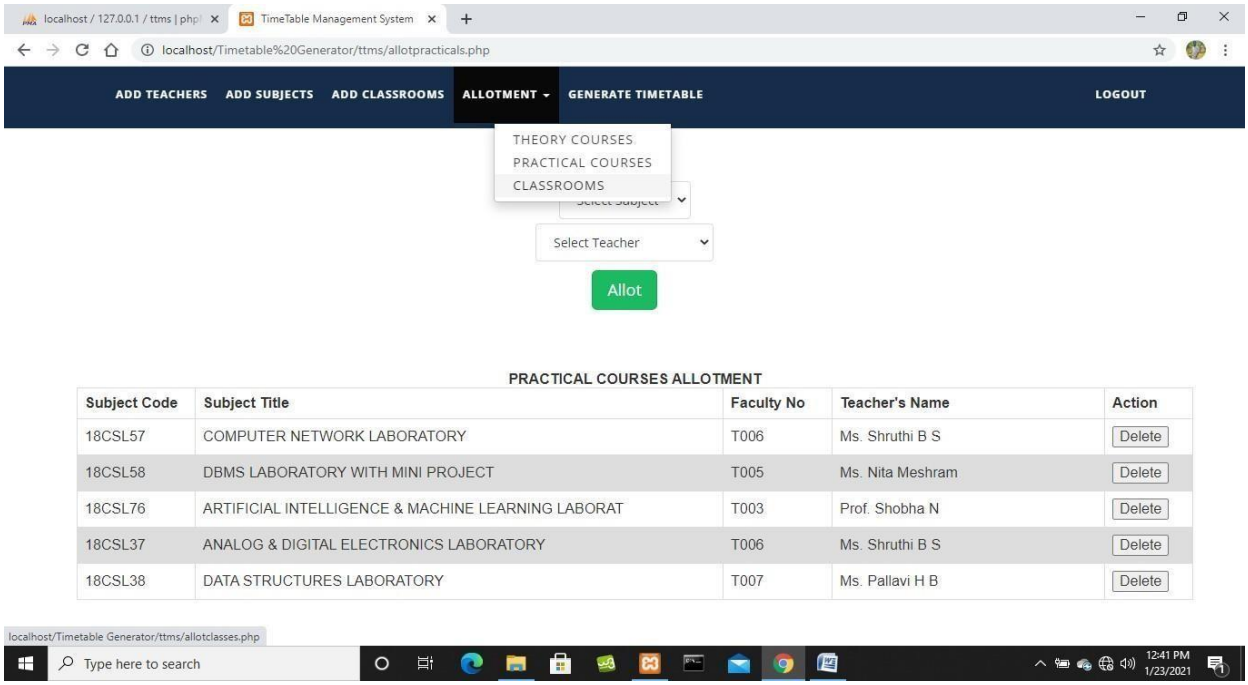


Fig. 6.13: Allot Classrooms

6.14 CLASSROOMS DETAILS ALLOTTED

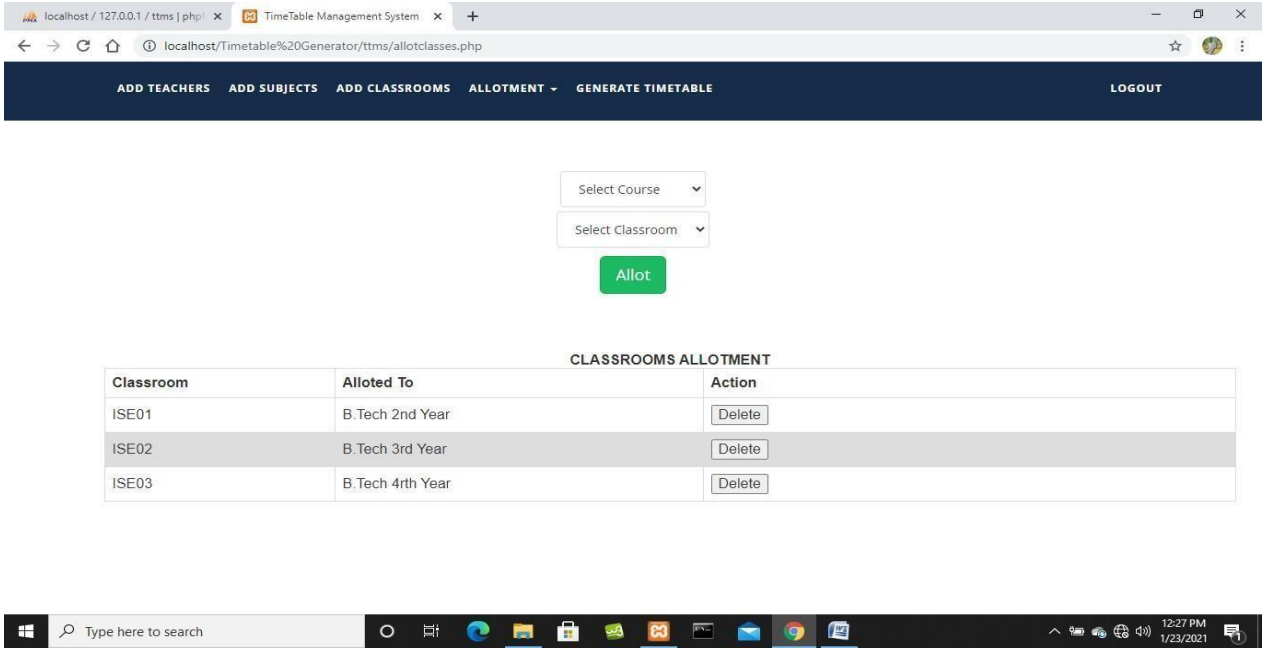


Fig. 6.14: Classrooms Details Allotted

6.15 GENERATE TIMETABLE

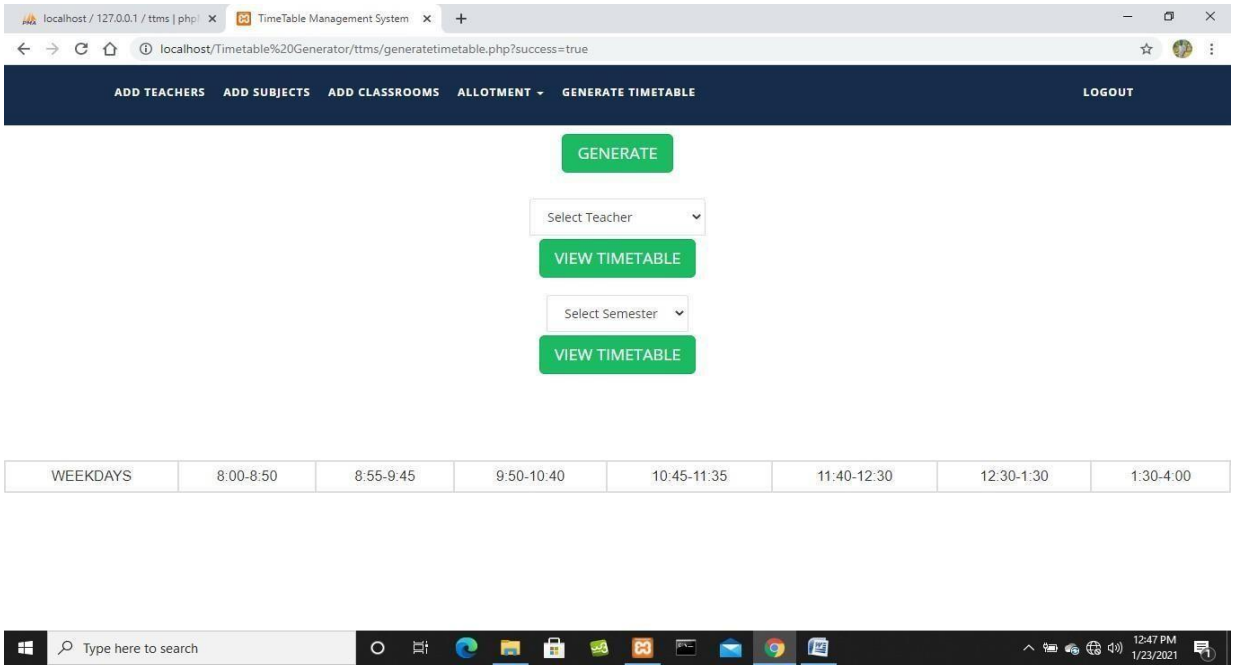


Fig. 6.15: Generate Timetable

6.16 SELECT TEACHERS TIMETABLE

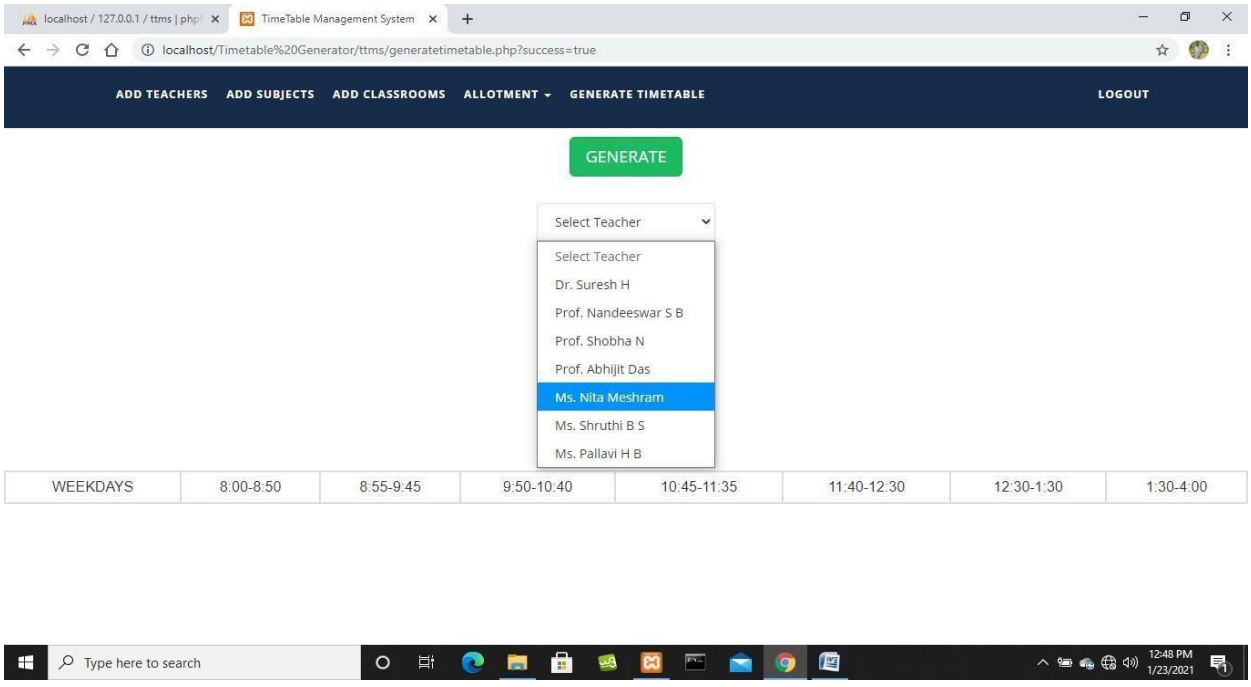


Fig. 6.16: Select Teachers Timetable

6.17 VIEW TEACHERS TIMETABLE

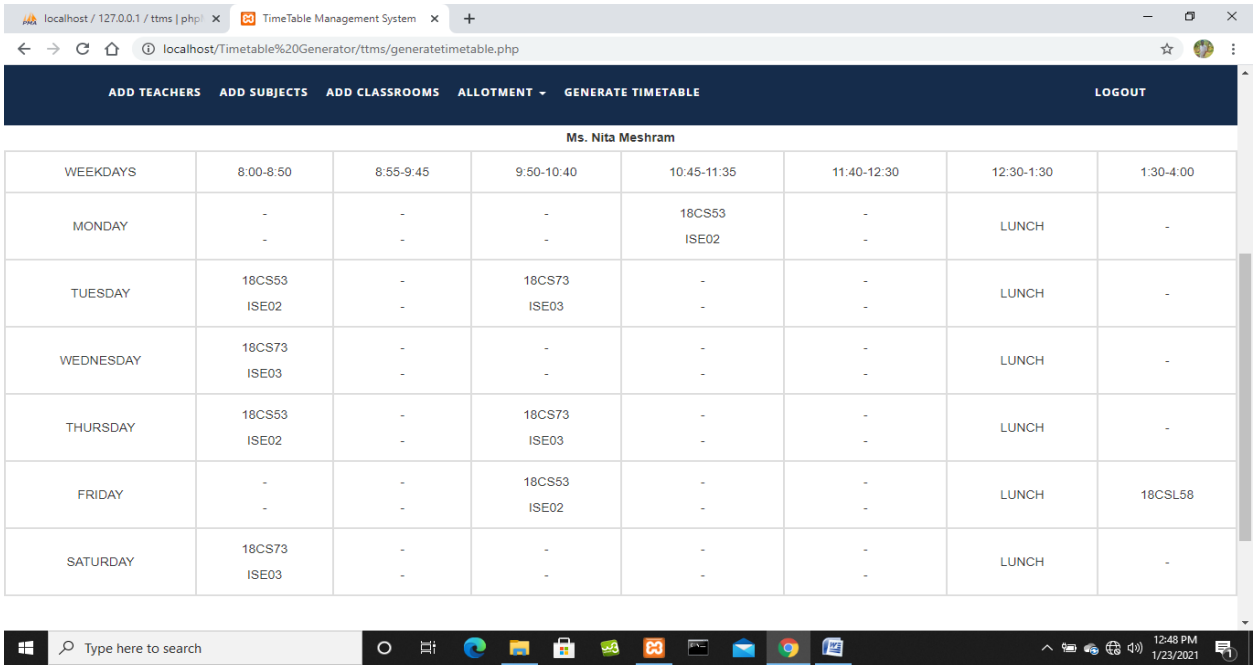


Fig. 6.17: View Teachers Timetable

6.18 SELECT SEMESTER TIMETABLE

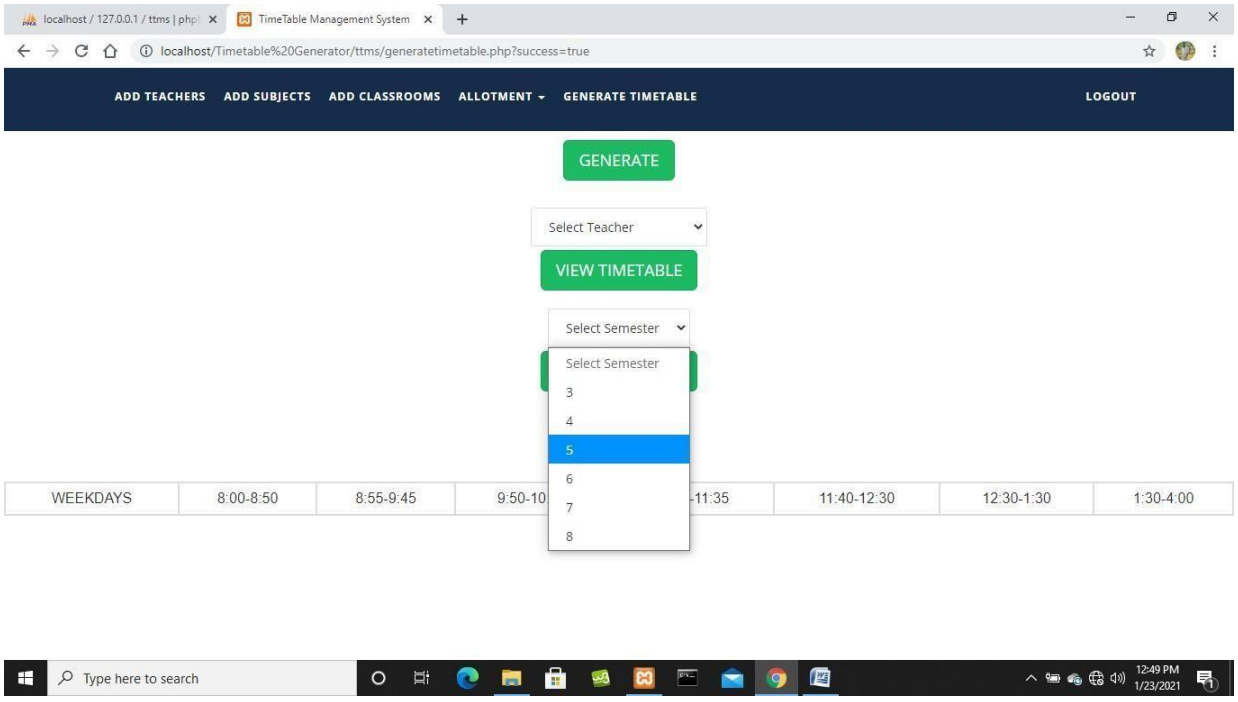
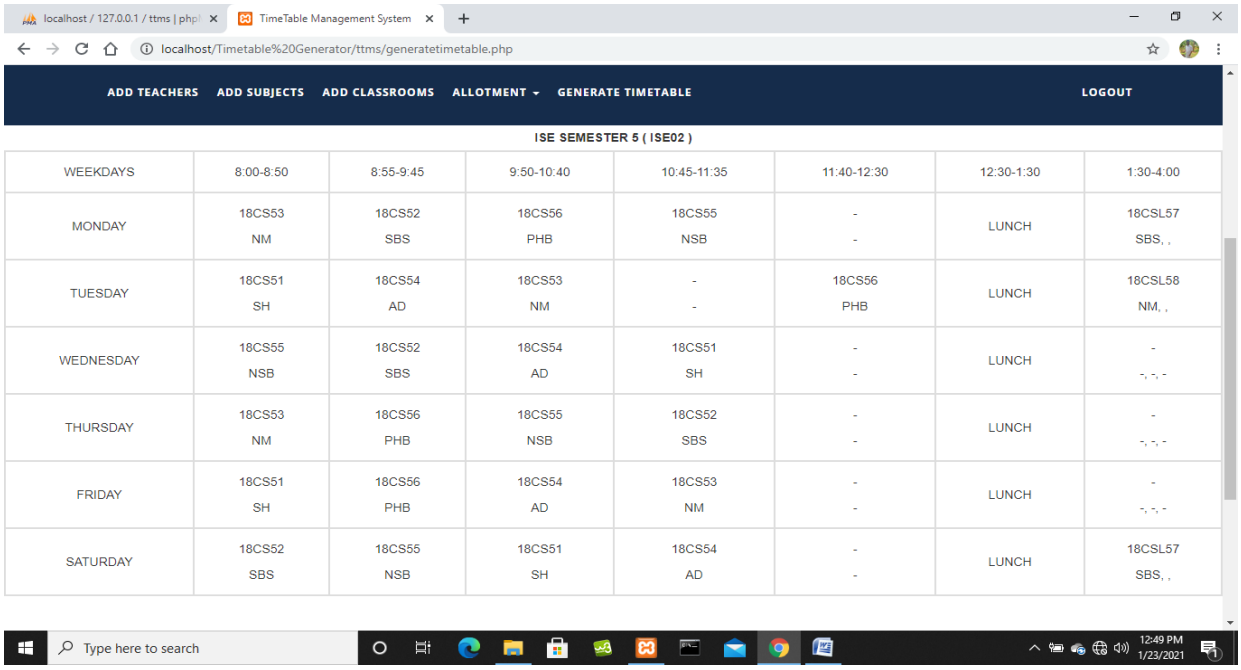


Fig. 6.18: Select Semester Timetable

6.19 VIEW SEMESTER TIMETABLE



WEEKDAYS	8:00-8:50	8:55-9:45	9:50-10:40	10:45-11:35	11:40-12:30	12:30-1:30	1:30-4:00
MONDAY	18CS53 NM	18CS52 SBS	18CS56 PHB	18CS55 NSB	- -	LUNCH	18CSL57 SBS, ,
TUESDAY	18CS51 SH	18CS54 AD	18CS53 NM	- -	18CS56 PHB	LUNCH	18CSL58 NM, ,
WEDNESDAY	18CS55 NSB	18CS52 SBS	18CS54 AD	18CS51 SH	- -	LUNCH	- -, -, -
THURSDAY	18CS53 NM	18CS56 PHB	18CS55 NSB	18CS52 SBS	- -	LUNCH	- -, -, -
FRIDAY	18CS51 SH	18CS56 PHB	18CS54 AD	18CS53 NM	- -	LUNCH	- -, -, -
SATURDAY	18CS52 SBS	18CS55 NSB	18CS51 SH	18CS54 AD	- -	LUNCH	18CSL57 SBS, ,

Fig. 6.19: View Semester Timetable

6.20 FACULTY LOGIN PAGE

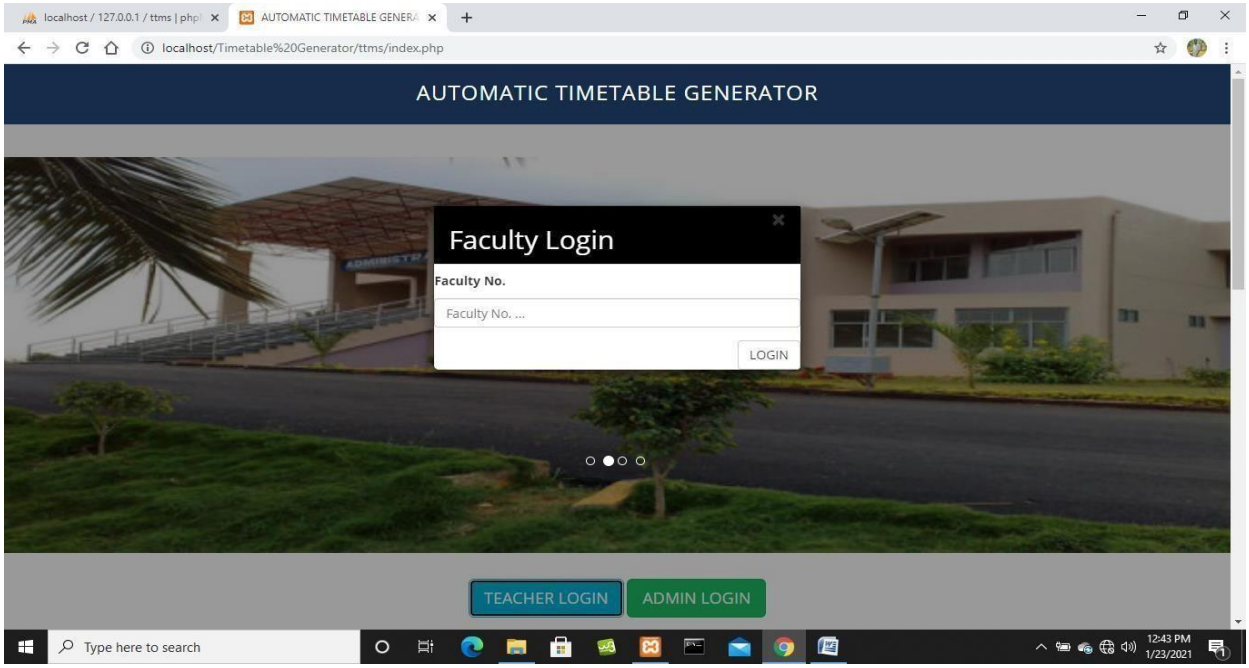


Fig. 6.20: Faculty Login Page

6.21 FACULTY PAGE

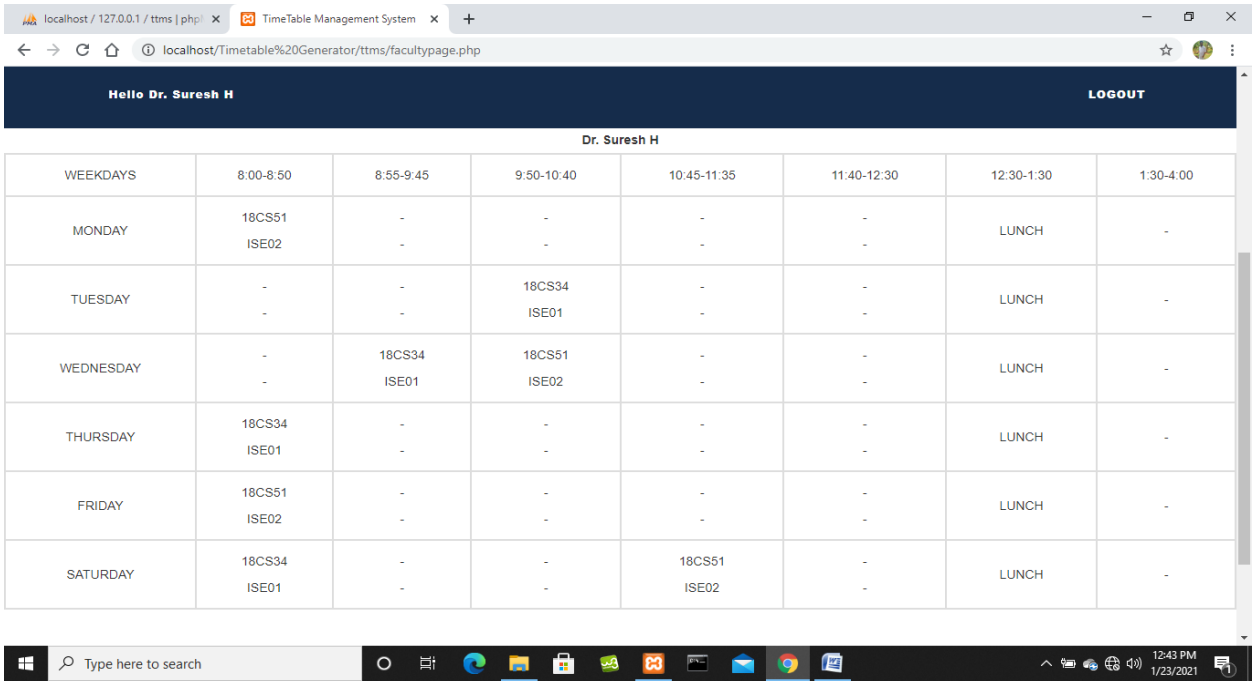


Fig. 6.21: Faculty Page

6.22 STUDENT LOGIN PAGE

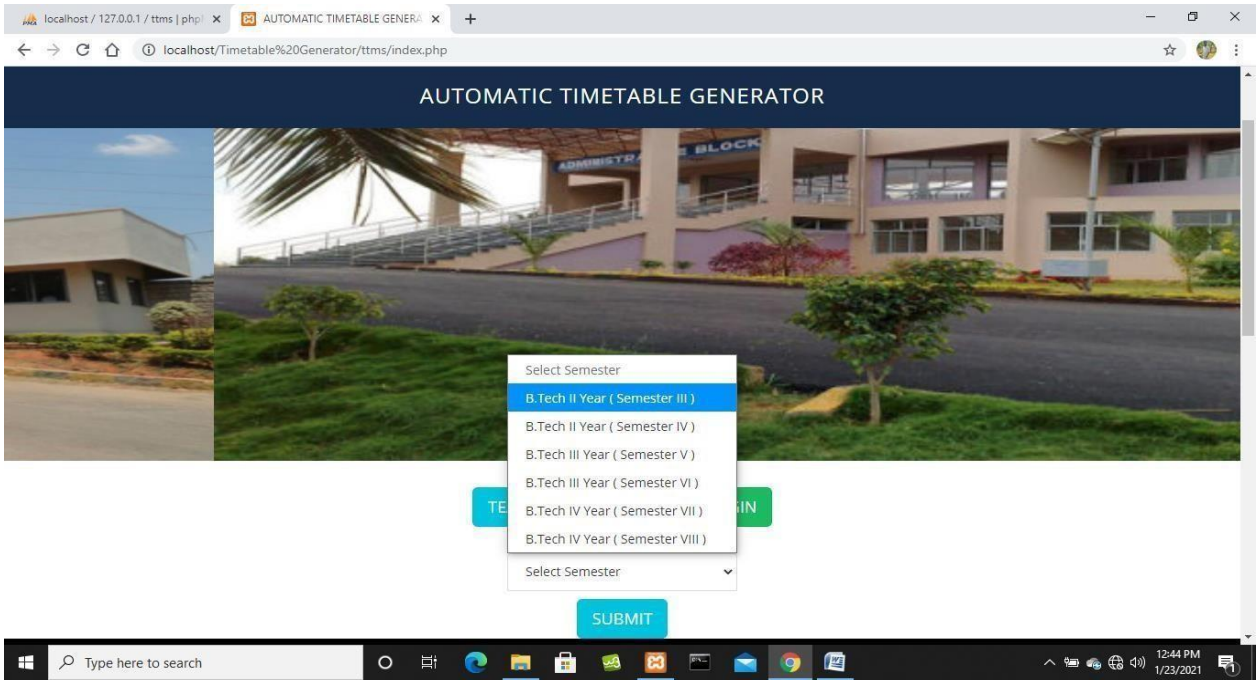


Fig. 6.22: Student Login Page

6.23 STUDENT PAGE

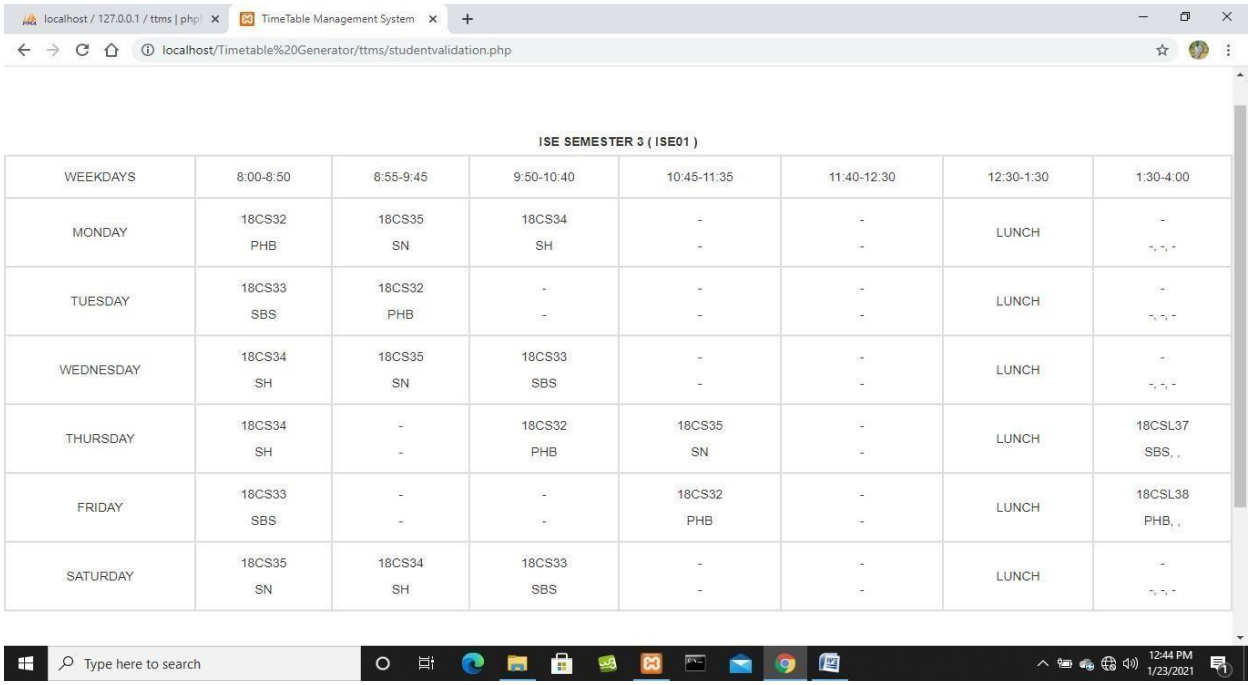


Fig. 6.23: Student Page

CONCLUSION

Generally, this system can be considered a useful system since it helps the lecturer to improve their process of preparing the timetable. By providing support through the Timetable Management system, the usage can be increased to any faculties. If the system is successfully upload to host, to assist administrator, lecturer and student on how to use the system.

Our approach of developing automated timetable system is successful in solving colleges lecture-course timetabling problem. We have also shown that how we can fit our timetabling system as Rich Desktop Application. The graphical user interface (Windows Form Application) used in this application provides an easy way in understanding how application works and also makes ease in providing the input. This application is provided with necessary details of faculty and subjects which are stored in database (SQL SERVER) and then by making use of the available data it generates the lecture-course timetable with minimum time when compared to manual generation of timetable and involves in satisfying all the constraints –

- No overlapping of time slots for any subject.
- There should be a minimum gap of one hour for respective faculty per subject
- No repetition of time slots per faculty.

FUTURE ENHANCEMENT

In our system, there are some problems those are User has to format it a bit after it is prepared. The transaction is executed in off-line mode, hence on-line data for room Student capture and modification is not possible. In future work, we will overcome these disadvantages by using fuzzy logic approaches.

BIBLIOGRAPHY

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. XAMPP tutorial on how to create database - https://www.tutorialspoint.com/index.htm
3. PHP project models and databases
4. References for CSS and HTML - https://www.w3schools.com/html/html_css.asp
5. References for CSS - https://codepen.io
6. PHP. The Complete Reference by Steven Holzer
7. StackOverflow.com referred for PHP
8. YouTube Channels referred - https://www.youtube.com/
    - Edurekha Awa Melvine
    - learnWebCoading
    - Freelancer Rooban
9. References for HTML, CSS, PHP and JavaScript in TutorialsPoint - https://www.tutorialspoint.com/index.htm
10. References for PHP and JavaScript in W3Schools - https://www.w3schools.com/