**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:*

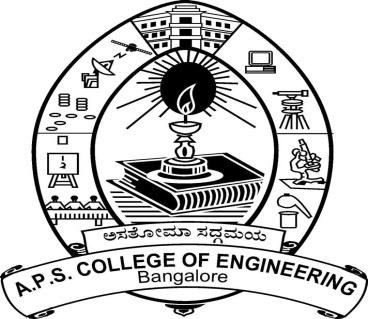
**NAGABHUSHAN V (1AP18IS016)**

*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.

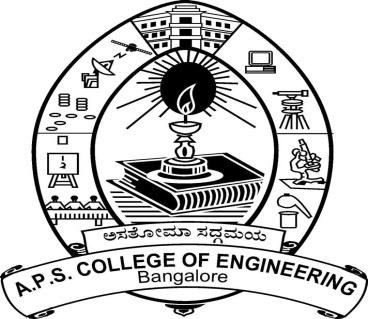
2020-2021

# APS COLLEGE of ENGINEERING

Anantha Gnana Gangothri,

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

**DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING**



## CERTIFICATE

This is to certify that the project work entitled

### “AUTOMATIC TIMETABLE GENERATOR”

Is a bonafide work carried out by:

**NAGABHUSHAN V 1AP18IS016**

In partial fulfillment of the requirement for “DBMS Laboratory with Mini Project” of Fifth semester Bachelor of Engineering in Information Science & Engineering of Visvesvaraya Technological University, Belgaum during the year 2020-2021.

It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the

“DBMS Laboratory with Mini Project” of Fifth semester Bachelor of Engineering in Information Science and Engineering.

---------------------------- ---------------------------

**Mrs. Nita Meshram Dr. Suresh H**

Asst. Professor, Assoc. Professor &

Dept. of IS & E, Head Dept. of IS & E,

APSCE. APSCE.

**Name of the external Signature and Date** 1.

2.

#### 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)

1. NAGABHUSHAN V (1AP18IS016)

III

#### ABSTRACT

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 lecturetimetabling 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.

IV

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS**        Acknowledgement | | III |
| Abstract | | IV |
| Table of Contents | | V |
| List of Figures | | VII |
| **Chapter 1** | **INTRODUCTION** |  |
| ❖ 1.1 | Database Management System | 8 |
| ❖ 1.1.1 | Introduction to DBMS | 8 |
| ❖ 1.1.2 | History of DBMS | 8 |
| ❖ 1.1.3 | Application of DBMS | 9 |
| * 1.1.4 * 1.1.5 * 1.2 * 1.2.1 * 1.2.2 * 1.2.3 * 1.3 * 1.4 | Advantages of DBMS  Disadvantages of DBMS  Project  Overview of Project  Features of Project  Application of Project  MySQL  HTML/PHP/CSS/JavaScript | 10  10  10  10  11  11  11  12 |
| **Chapter 2** | **REQUIREMENTS SPECIFICATION** |  |
| ❖ 2.1 | Overall Description | 13 |
| ❖ 2.2 | Specification Requirements | 13 |
| ❖ 2.2.1 | Software Requirements | 13 |
| ❖ 2.2.2 | Hardware Requirements      V | 13 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Chapter 3** | | **DETAILED DESIGN** |  | |  |
| ❖ 3.1 | | System Design |  | | 14 |
| ❖ 3.2 | | Entity Relationship Diagram |  | | 15 |
| ❖ 3.3 | | Relational Schema |  | | 16 |
| **Chapter 4** | | **IMPLEMENTATION** |  | |  |
| ❖ 4.1 | | Introduction to Software used |  | | 18 |
| ❖ 4.1.1 | | XAMPP |  | | 18 |
| * 4.1.2 * 4.1.3 * 4.2 * 4.3 * 4.4 | | phpMyAdmin  Visual Studio Code  Source Code  Problem Description  Result |  | | 18  18  19  23  23 |
| **Chapter 5** | | **SNAPSHOTS** |  | |  |
| ❖ 5.1 | | Home Page |  | | 24 |
| ❖ 5.2 | | Admin Login Page |  | | 24 |
| * 5.3 * 5.4 | | Add Teachers  Teachers Details |  | | 25  26 |
| * 5.5 * 5.6 | | Add Subjects  Subjects Details |  | | 26  27 |
| * 5.7 * 5.8 | | Add Classrooms  Classrooms Details |  | | 28  28 |
| * 5.9 * 5.10 * 5.11 * 5.12 * 5.13 * 5.14 | | Allot Theory Subjects  Theory Subjects Details Allotted  Allot Practical Subjects  Practical Subjects Details Allotted  Allot Classrooms  Classrooms Details Allotted    VI |  | | 29  30  30  31  32  32 |
| * 5.15 * 5.16 * 5.17 * 5.18 * 5.19 * 5.20 * 5.21 * 5.22 * 5.23 | Generate Timetable  Select Teachers Timetable  View Teachers Timetable  Select Semester Timetable  View Semester Timetable  Faculty Login Page  Faculty Page  Student Login Page  Student Page | | | 33  34  34  35  36  36  37  38  38 | |
|  | Conclusion | | | 40 | |
|  | Future Enhancement | | | 40 | |
|  | Bibliography  **LIST OF FIGURES** | | | 40 | |
| 3.1 | JSP Architecture | | | 14 | |
| 3.2 | ER Diagram | | | 16 | |
| 3.3 | Schema Diagram  VII | | | 17 | |

#### CHAPTER 1 INTRODUCTION

##### 1.1 DATABASE MANAGEMENT SYSTEM

###### 1.1.

A database is simply 1 INTRODUCTION TO DBMS 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.

Page

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 sets 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 "postrelational" 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
* Finance
* Online Shopping
* Human Resource Management

Page

###### 1.1.4 ADVANTAGES OF DBMS

* Reducing Data Redundancy
* Sharing of Data
* Data Integrity
* Data Security
* Privacy
* Backup and Recovery
* Data Consistency

###### 1.1.5 DISADVANTAGES OF DBMS

* Increased costs. One of the disadvantages of DBMS is Database systems require sophisticated hardware and software and highly skilled personnel
* Management complexity
* Maintaining currency
* Frequent upgrade/replacement cycles

##### 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 APPLICATION OF PROJECT**

This system can be used by Schools and Colleges to create Timetable.

##### 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 Pro
* Front end design: HTML, PHP, CSS, JavaScript
* Back end design: MySQL

###### 2.2.2 HARDWARE REQUIREMENTS

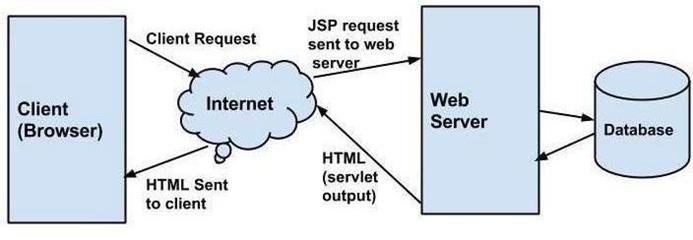
* Processor: Intel CORE i5
* RAM: 8GB
* Hard disk: 1TB

#### 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

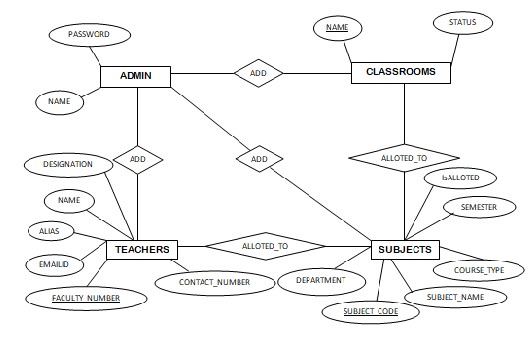
Three-tier Client / Server database architecture is commonly used architecture for web applications. Intermediate layer called Application server or Web Server stores the web connectivity software and the business logic (constraints) part of application used to access the right amount of data from the 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-attributerelationship 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 conceptuallogical-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.

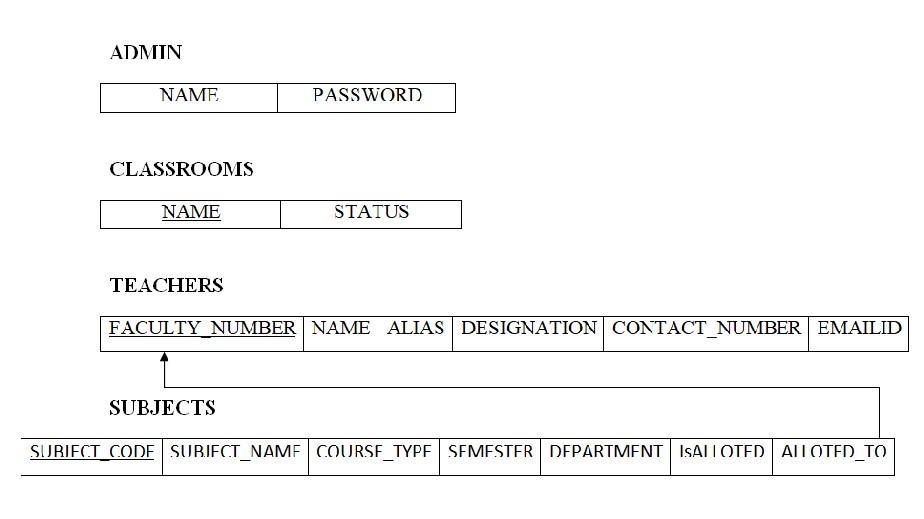
Cardinality notations define the attributes of the relationship between the entities. Cardinalities can denote that an entity is optional.



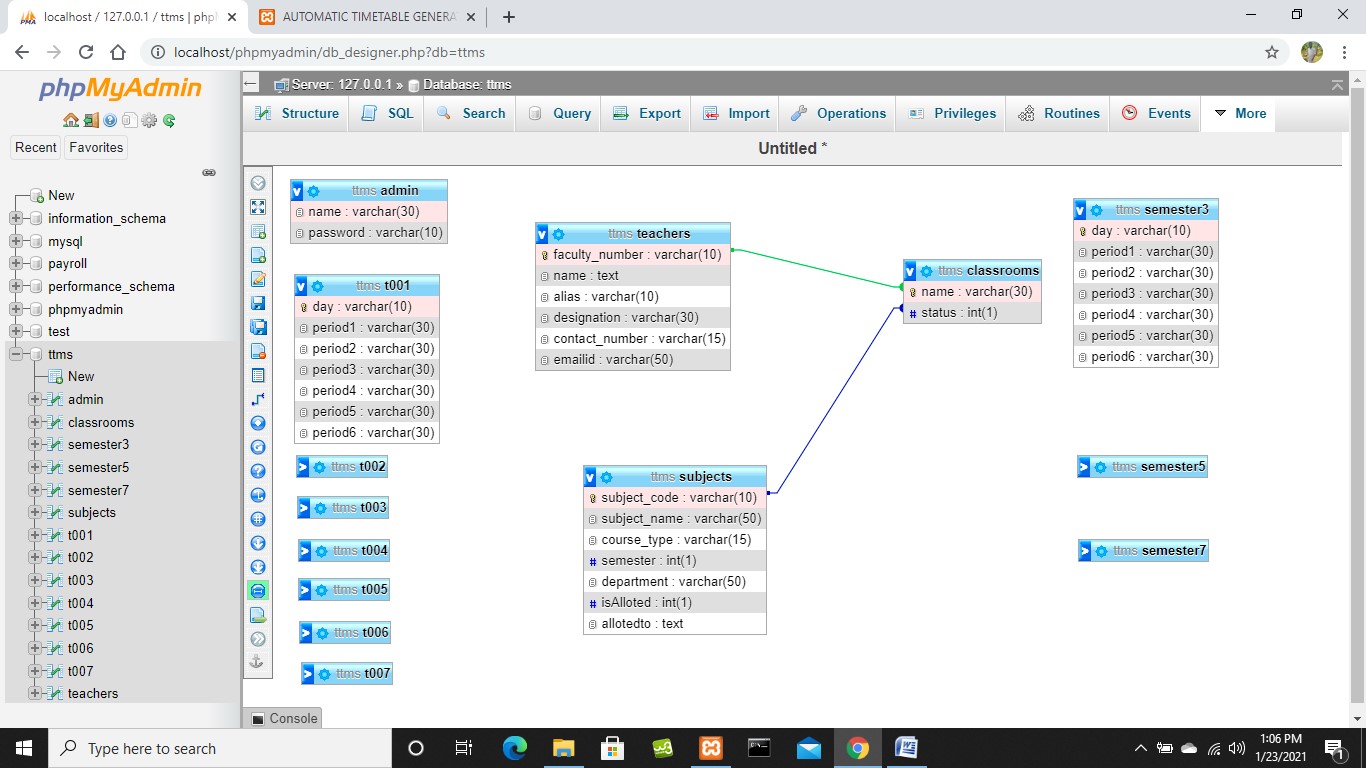
**Fig. 3.2: Enhanced ER diagram of Automatic Timetable Generator**

###### 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.



**Fig. 3.3 (a): Relational Schema**



**Fig. 3.3 (b): Relational Schema**

#### CHAPTER 4 IMPLEMENTATION

##### 4.1 INTRODUCTION TO SOFTWARE USED

The softwares used for project are:

* XAMPP
* phpMyAdmin
* Visual Studio Code

###### 4.1.1 XAMPP

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

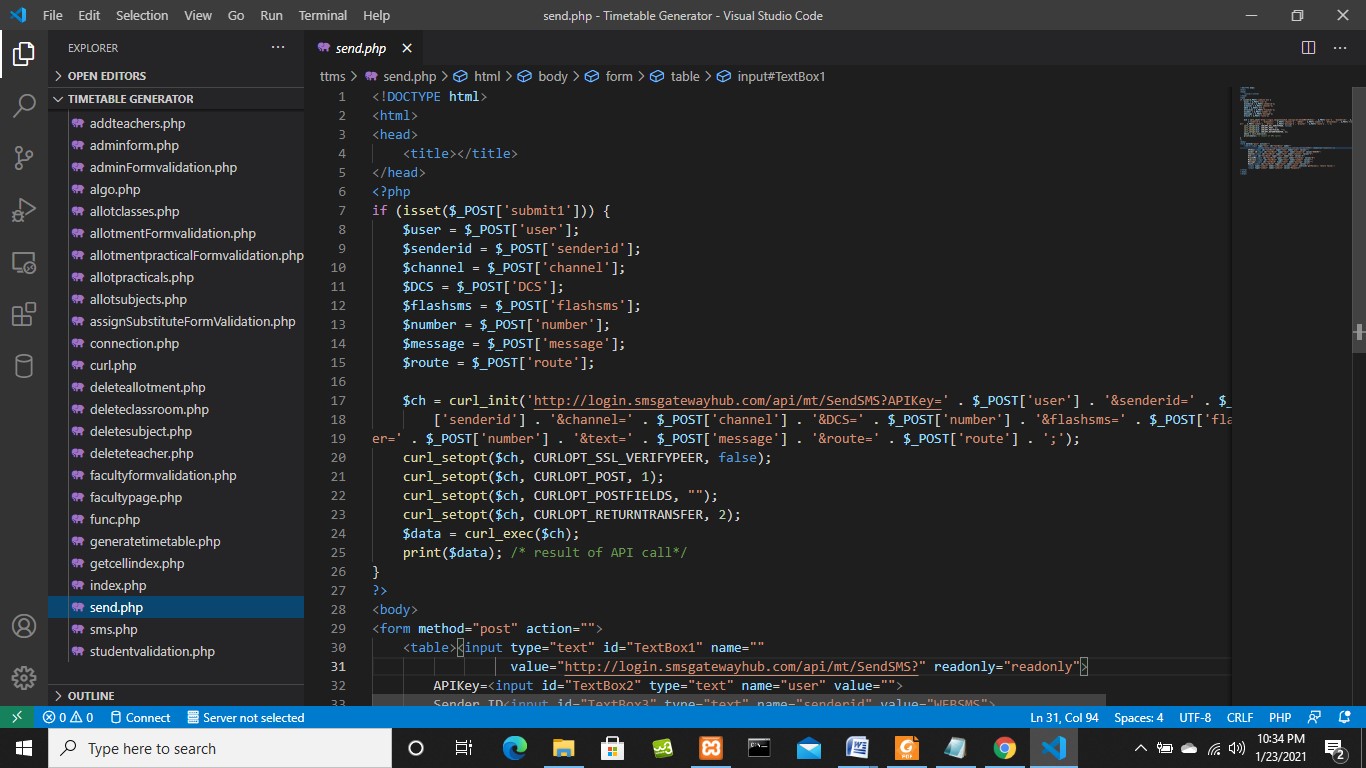
phpMyAdmin is a free and open source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it has become one of the most popular MySQL administration tools, especially for web hosting services.

###### 4.1.3 Visual Studio Code

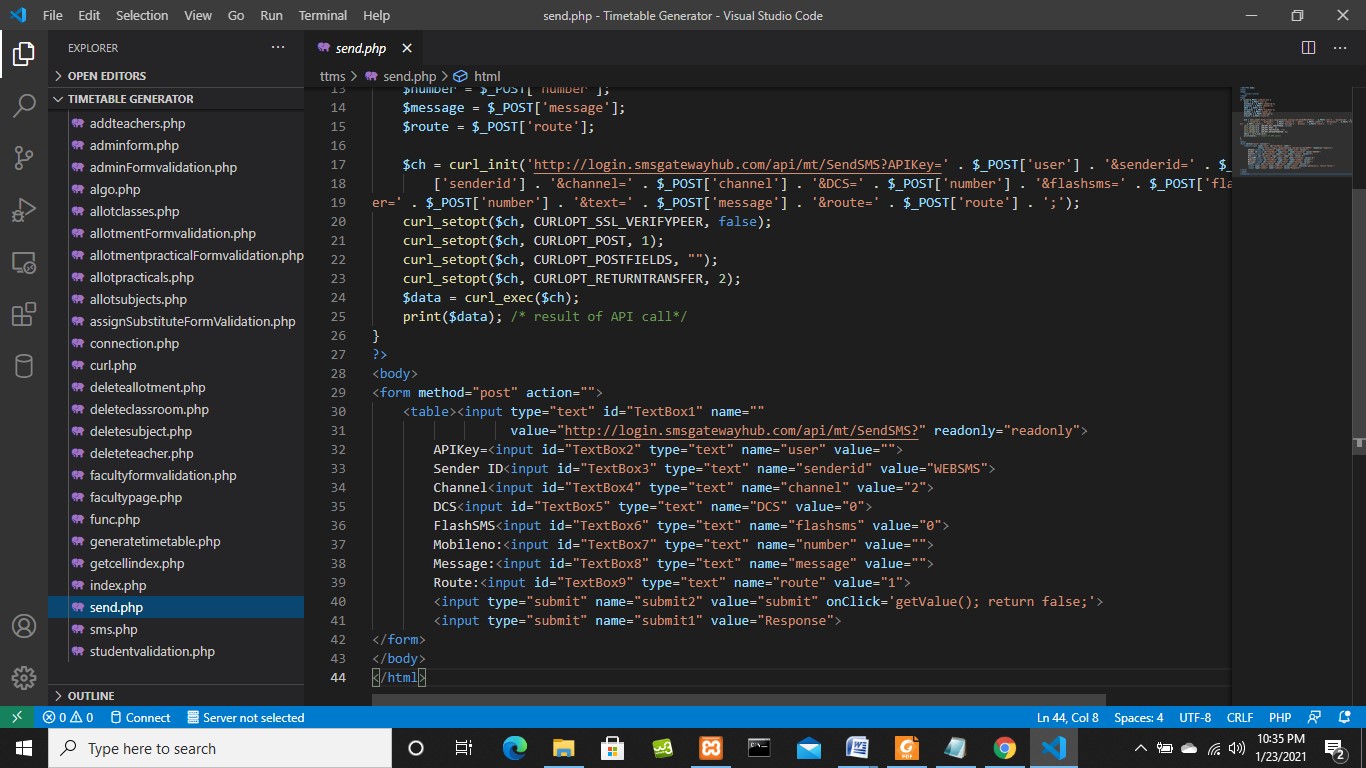
Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install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

**4.2 SOURCE CODE**

#### HTML

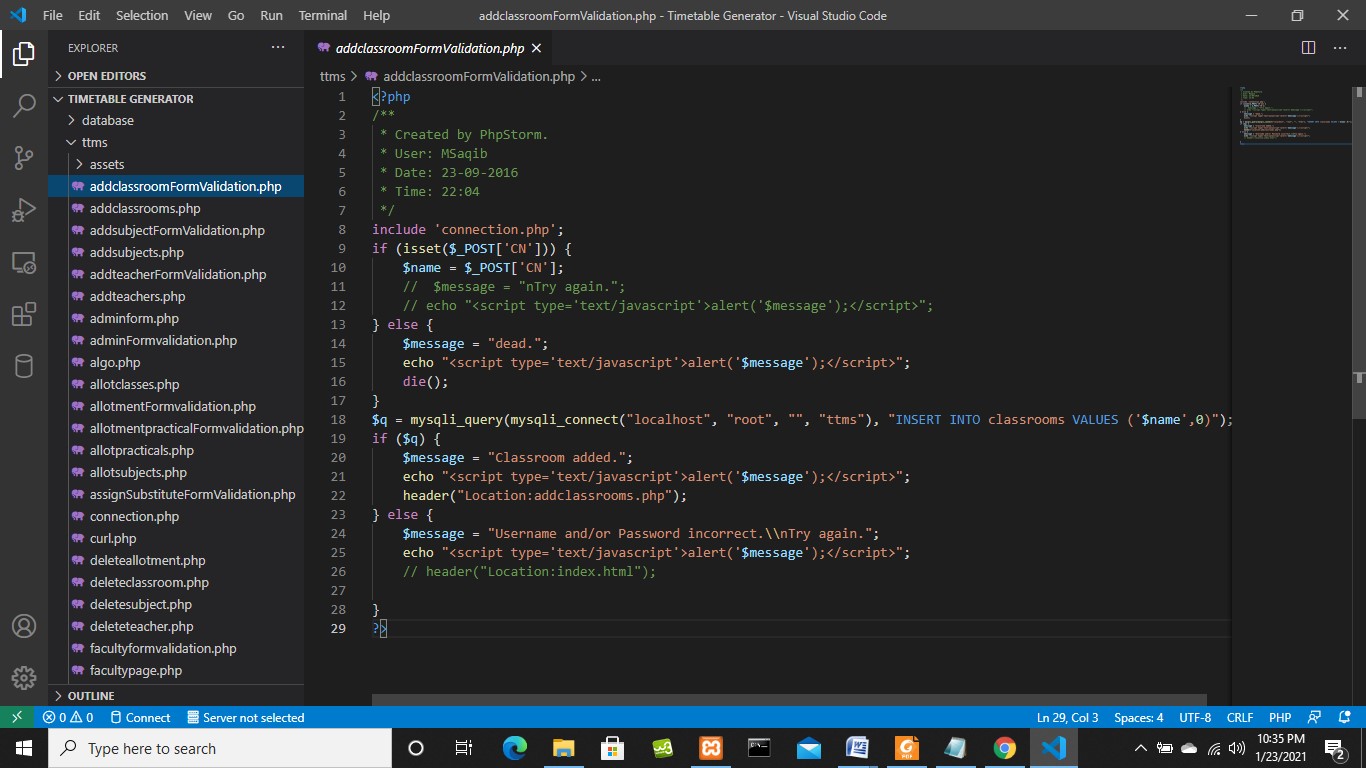


**Fig. 4.2 (a): HTML Code**



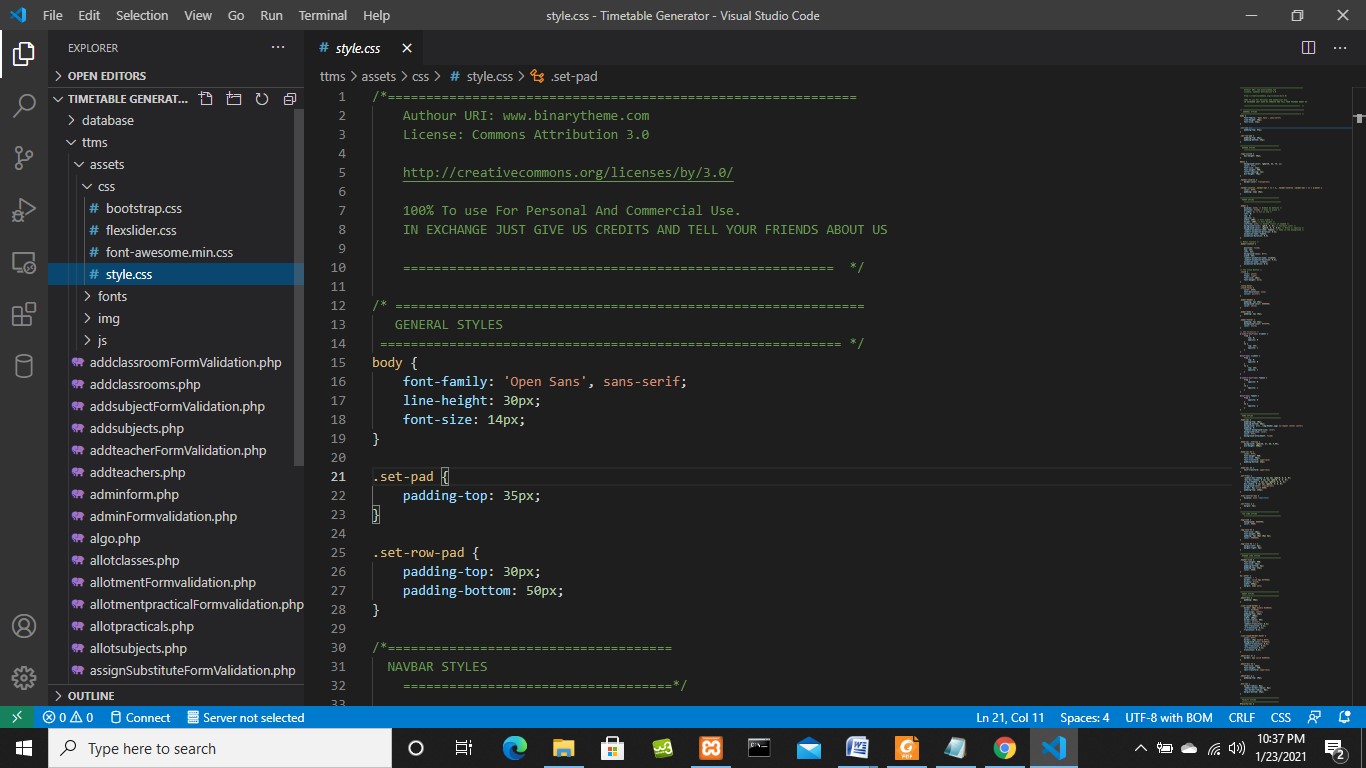
**Fig. 4.2 (b): HTML Code**

#### PHP

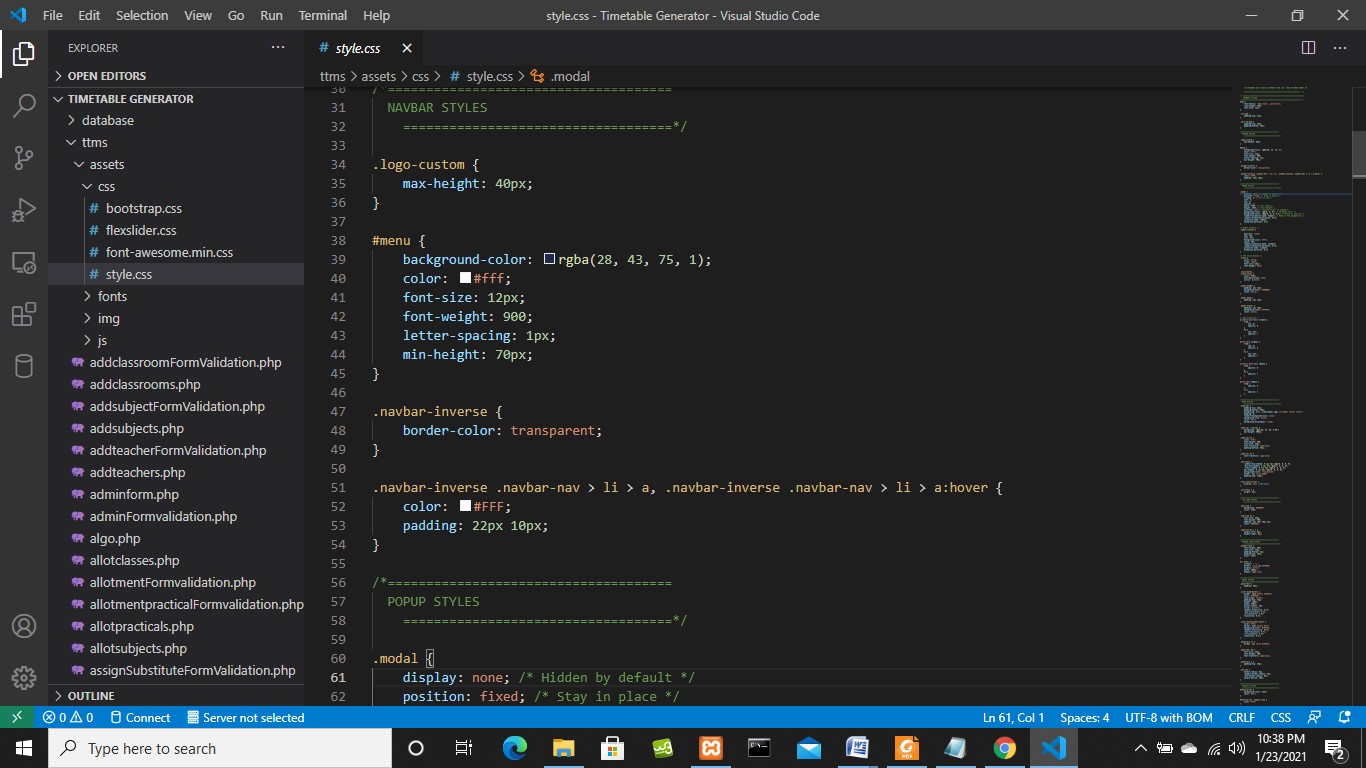


**Fig. 4.2 (c): PHP Code**

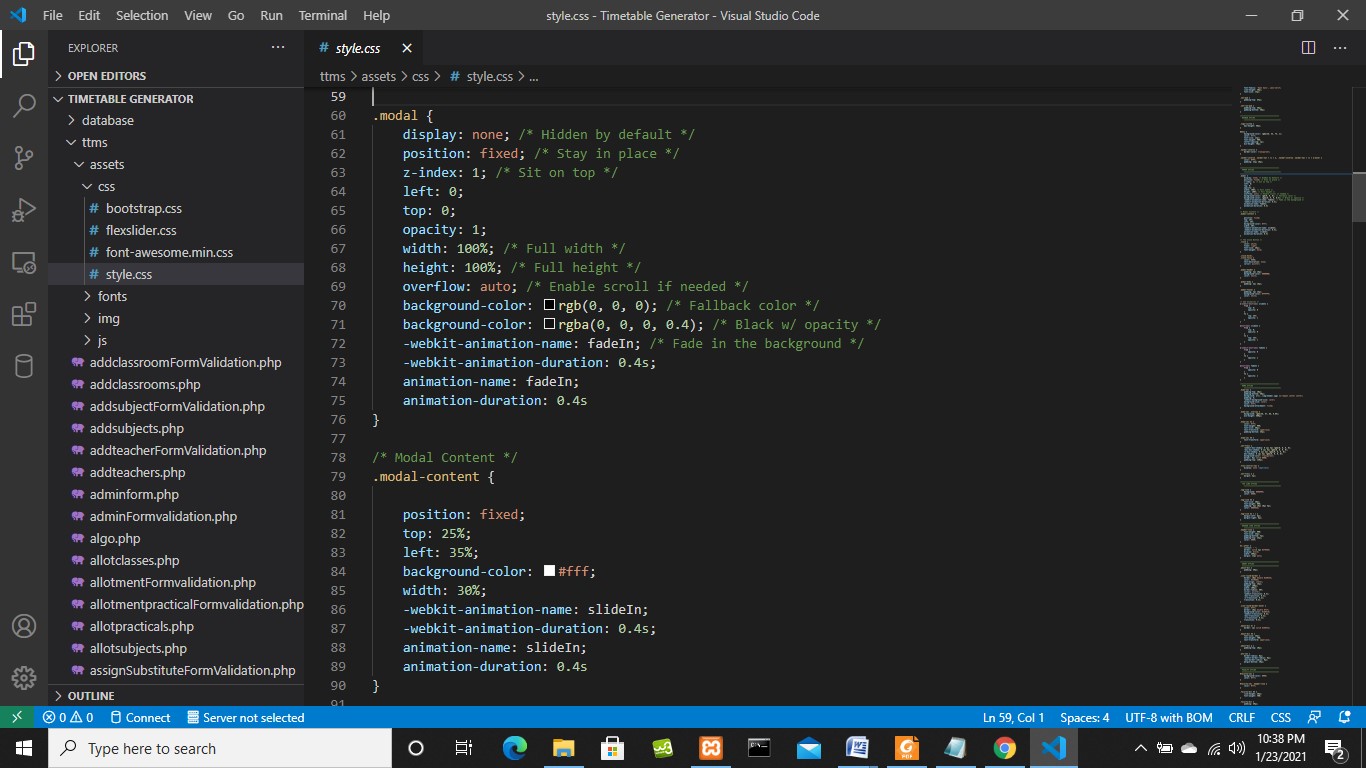
#### CSS



**Fig. 4.2 (d): CSS Code**

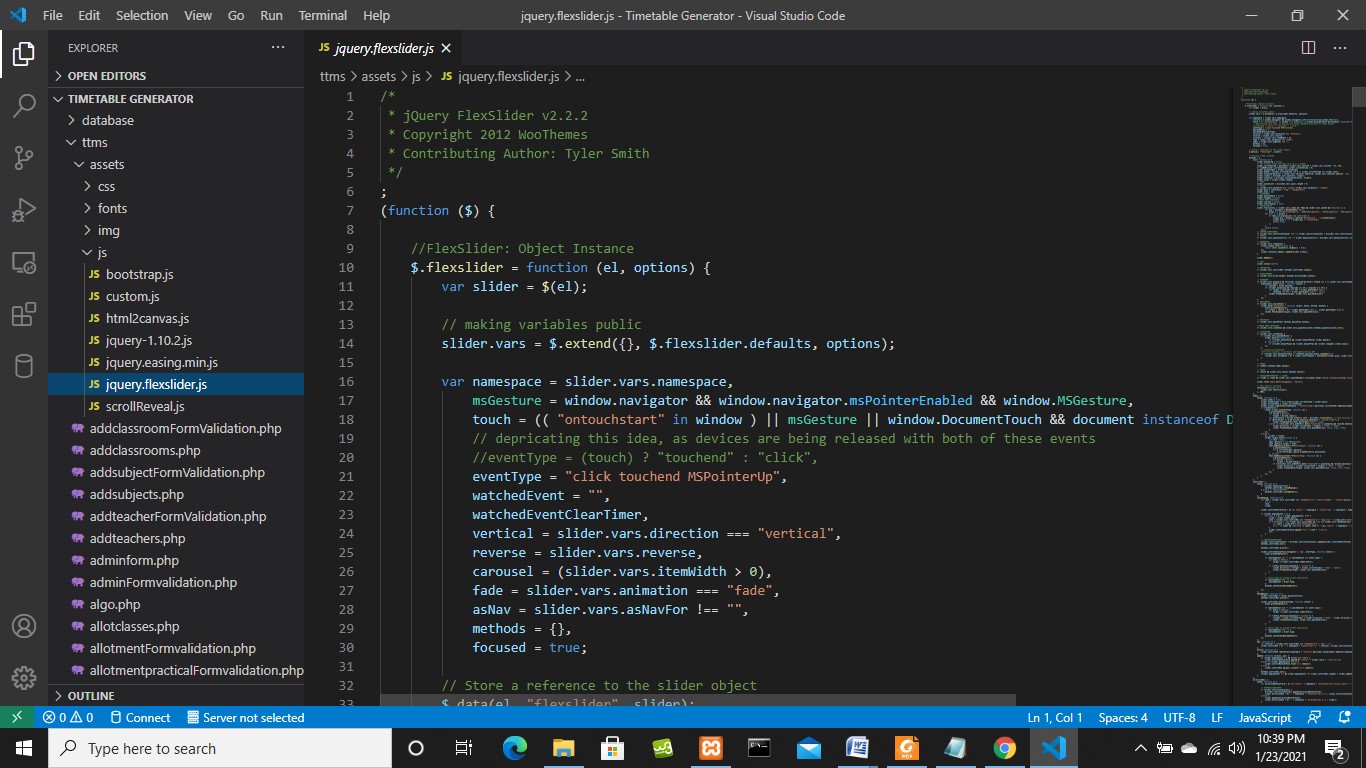


**Fig. 4.2 (e): CSS Code**



**Fig. 4.2 (f): CSS Code**

#### JavaScript



**Fig. 4.2 (f): JavaScript Code**

##### 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:

**For Admin Login**

“name”; “password”;

**For Classrooms**

“name”; “status”;

**For Teachers Login**

“faculty\_number”; “name”; “alias”; designation”; “contact\_number”; “mailid”;

**For Subjects**

“subject\_code”; “subject\_name”; “course\_type”; “semester”; “department”;

“isalloted”; “allotedto”;

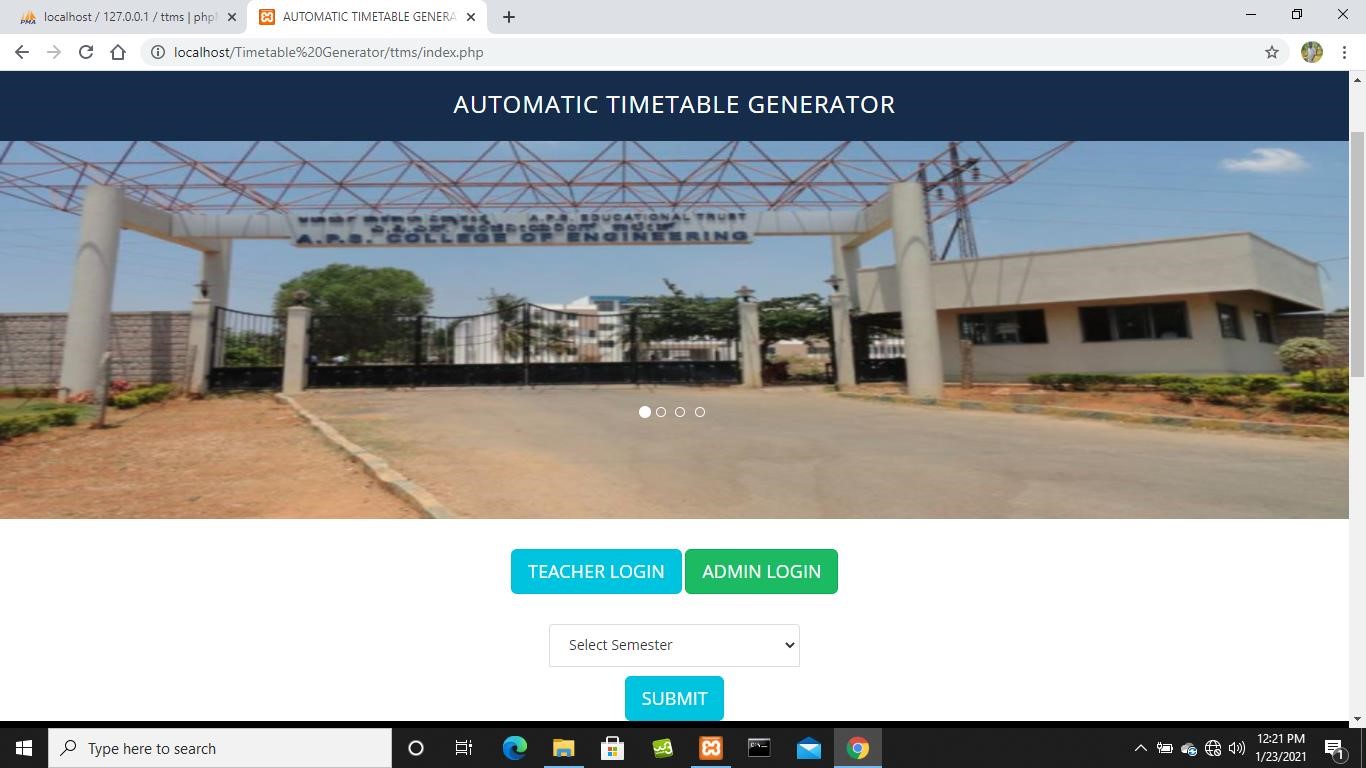
##### 4.4 RESULT

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

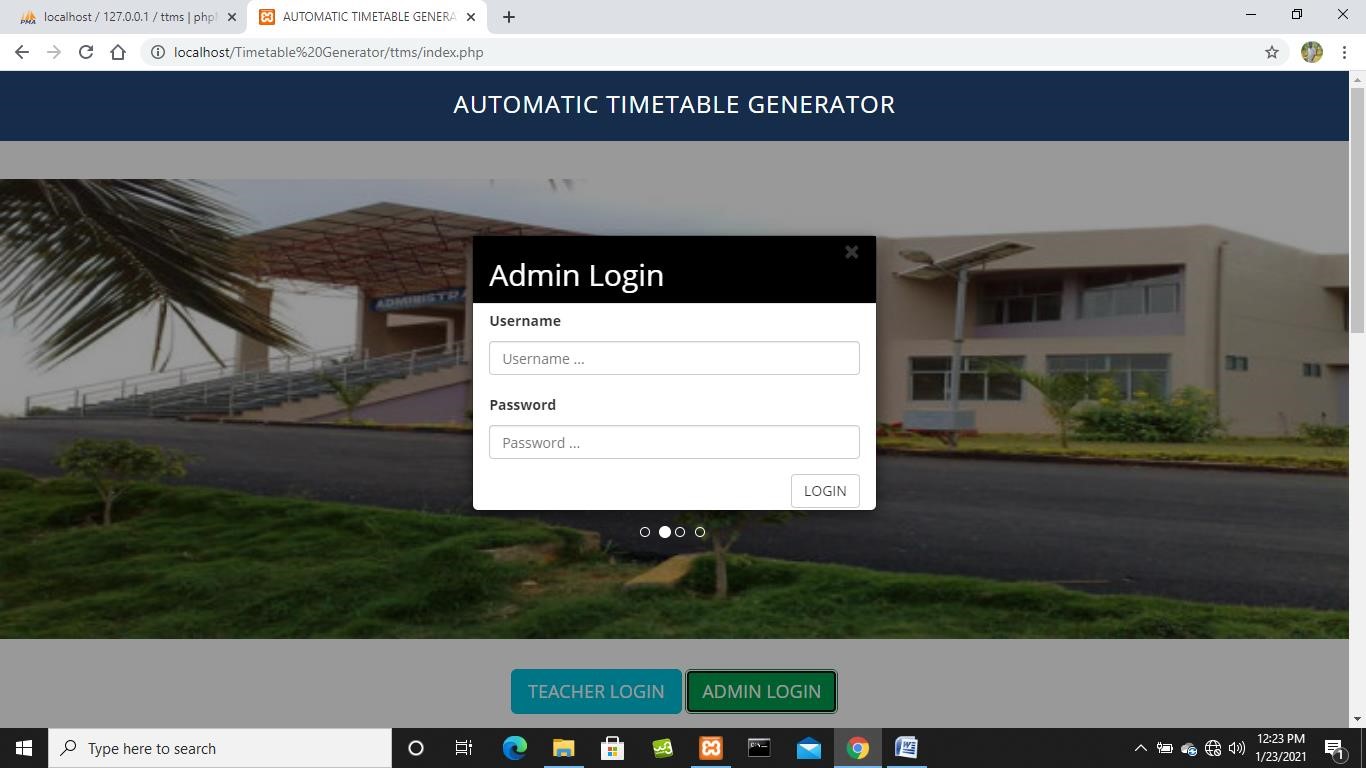
#### CHAPTER 5 SNAPSHOTS

##### 5.1 HOME PAGE



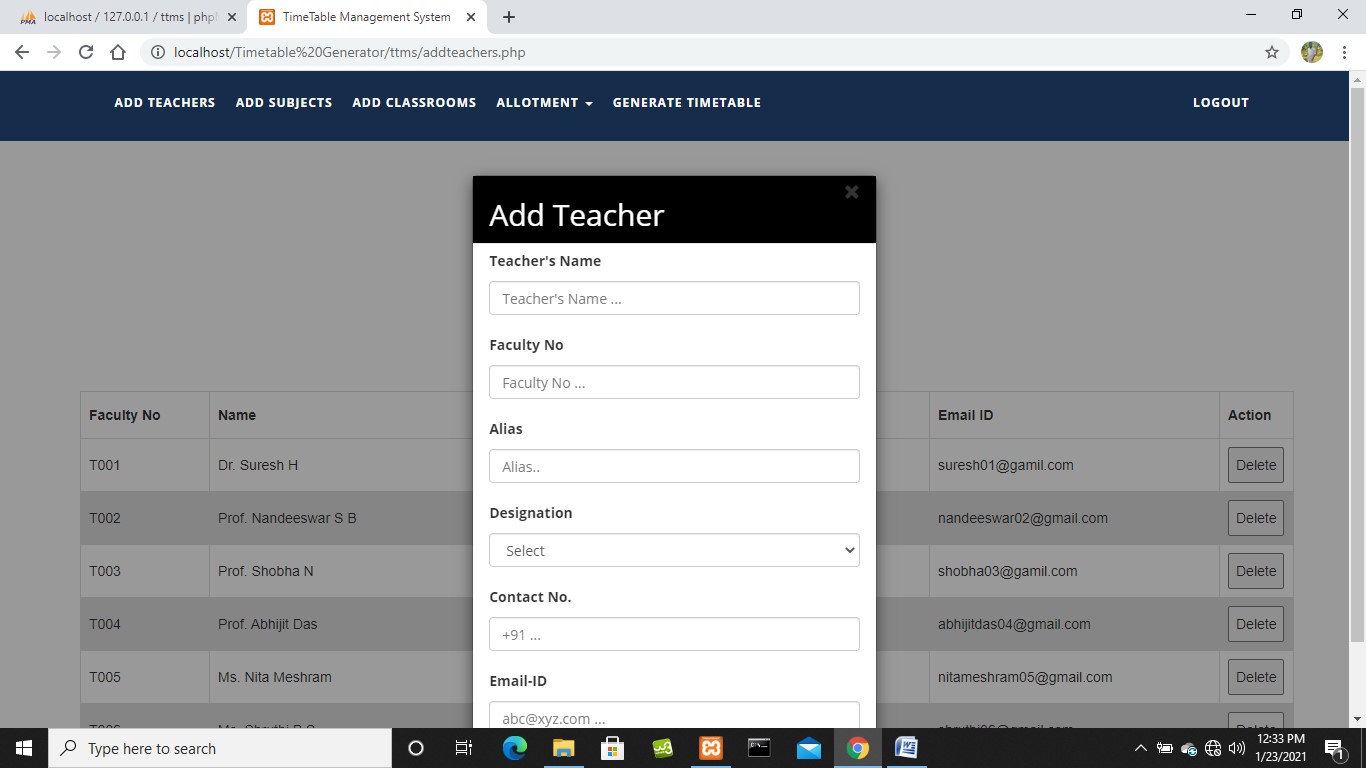
**Fig. 5.1: Home Page**

##### 5.2 ADMIN LOGIN PAGE



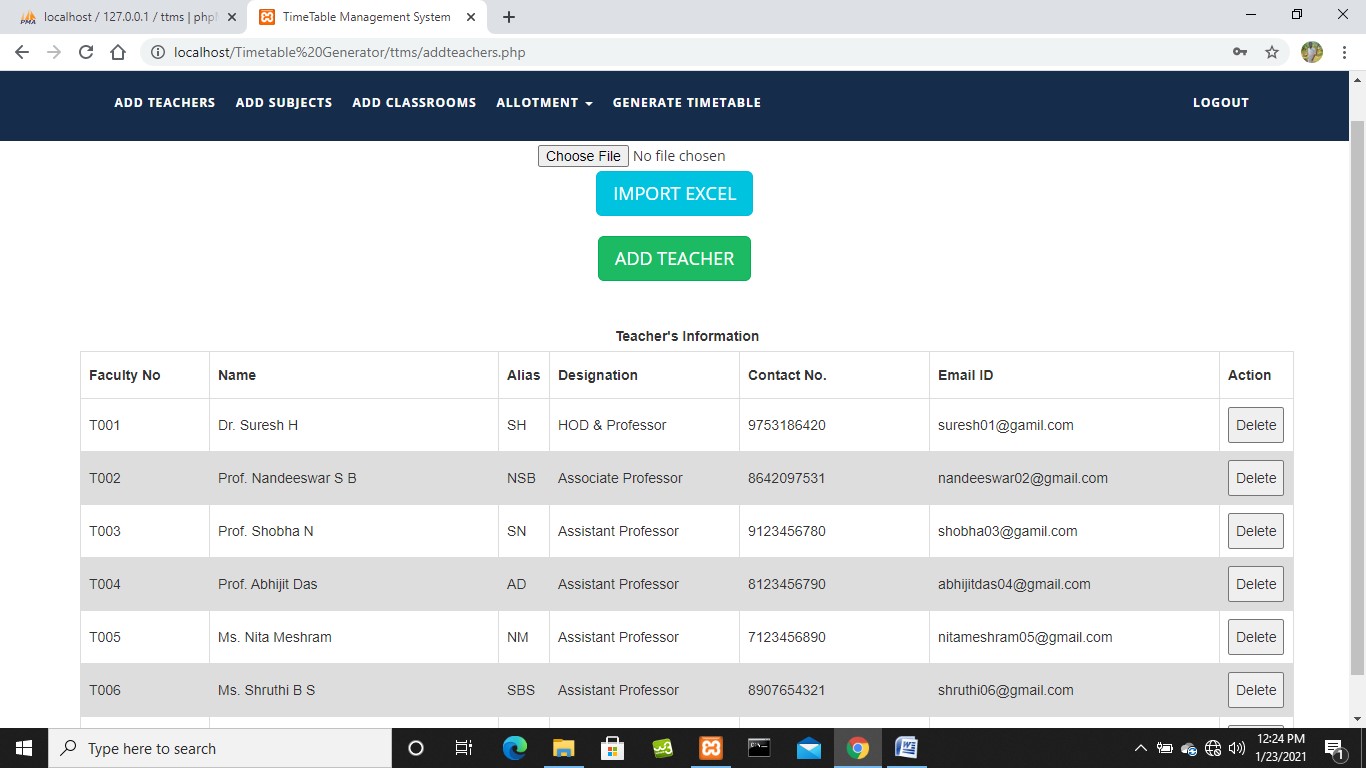
**Fig. 5.2: Admin Login Page**

##### 5.3 ADD TEACHERS



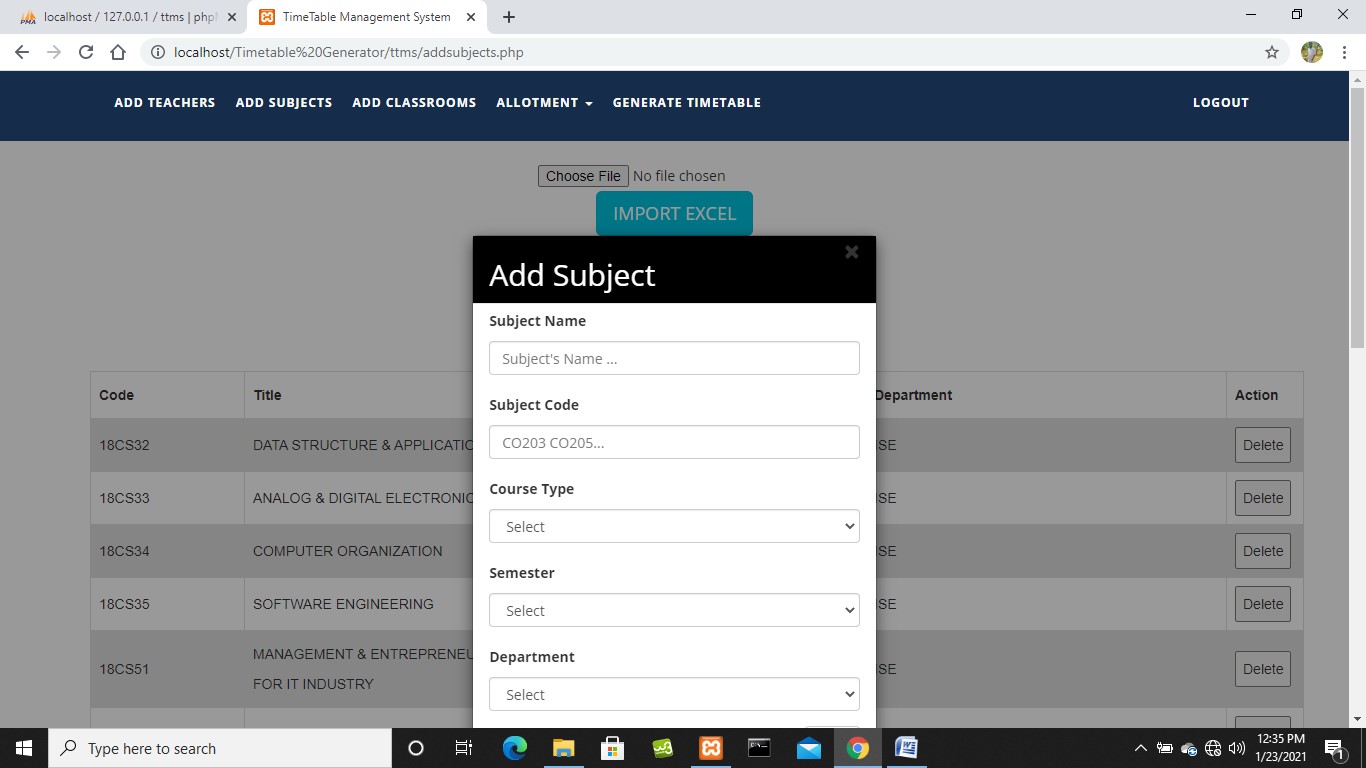
**Fig. 5.3: Add Teachers**

##### 5.4 TEACHERS DETAILS



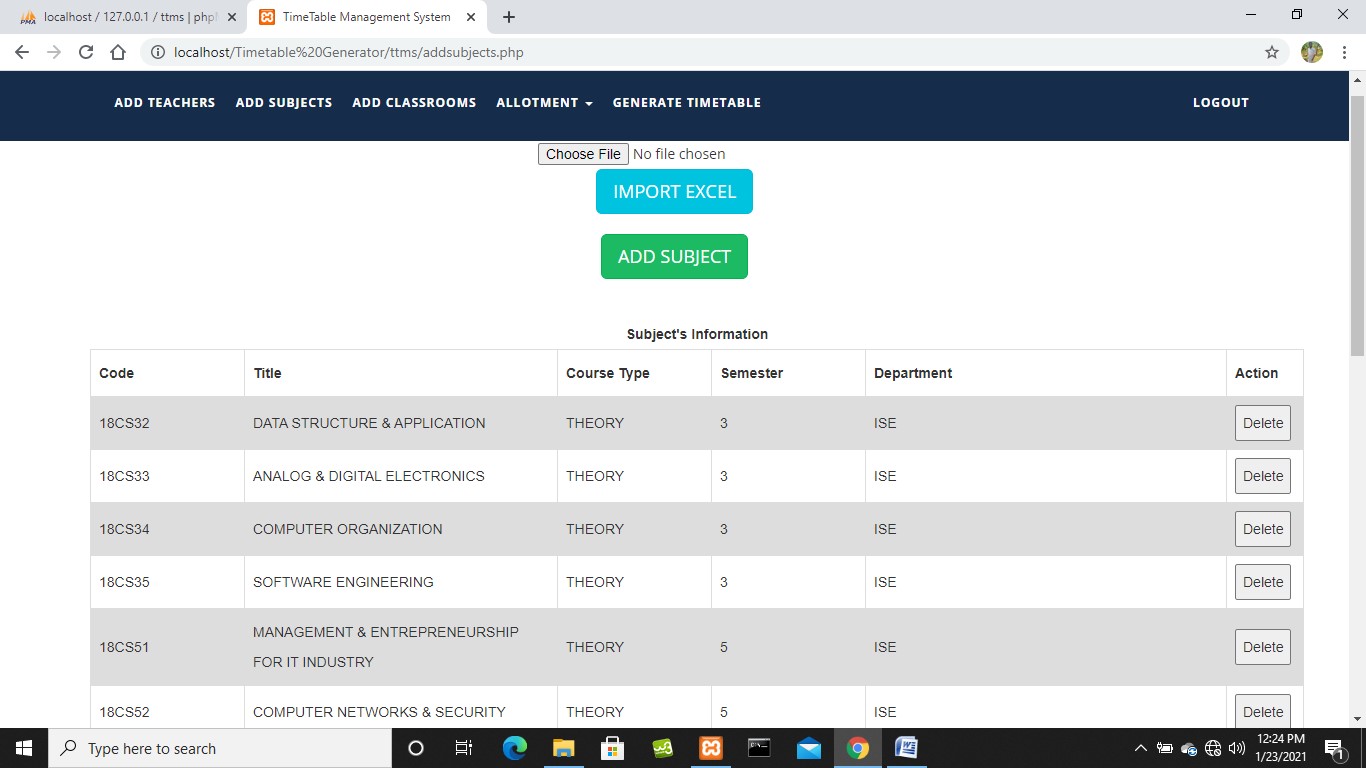
**Fig. 5.4: Teachers Details**

##### 5.5 ADD SUBJECTS



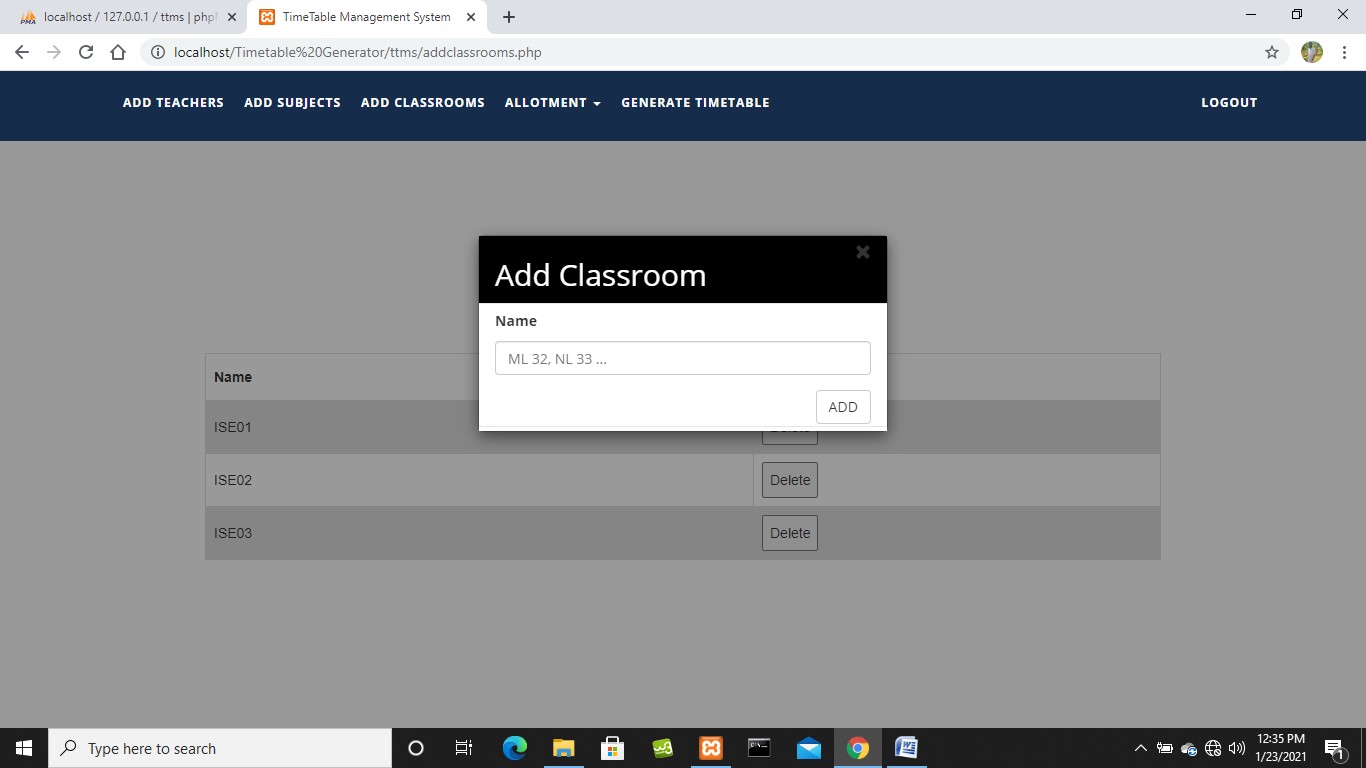
**Fig. 5.5: Add Subjects**

##### 5.6 SUBJECTS DETAILS



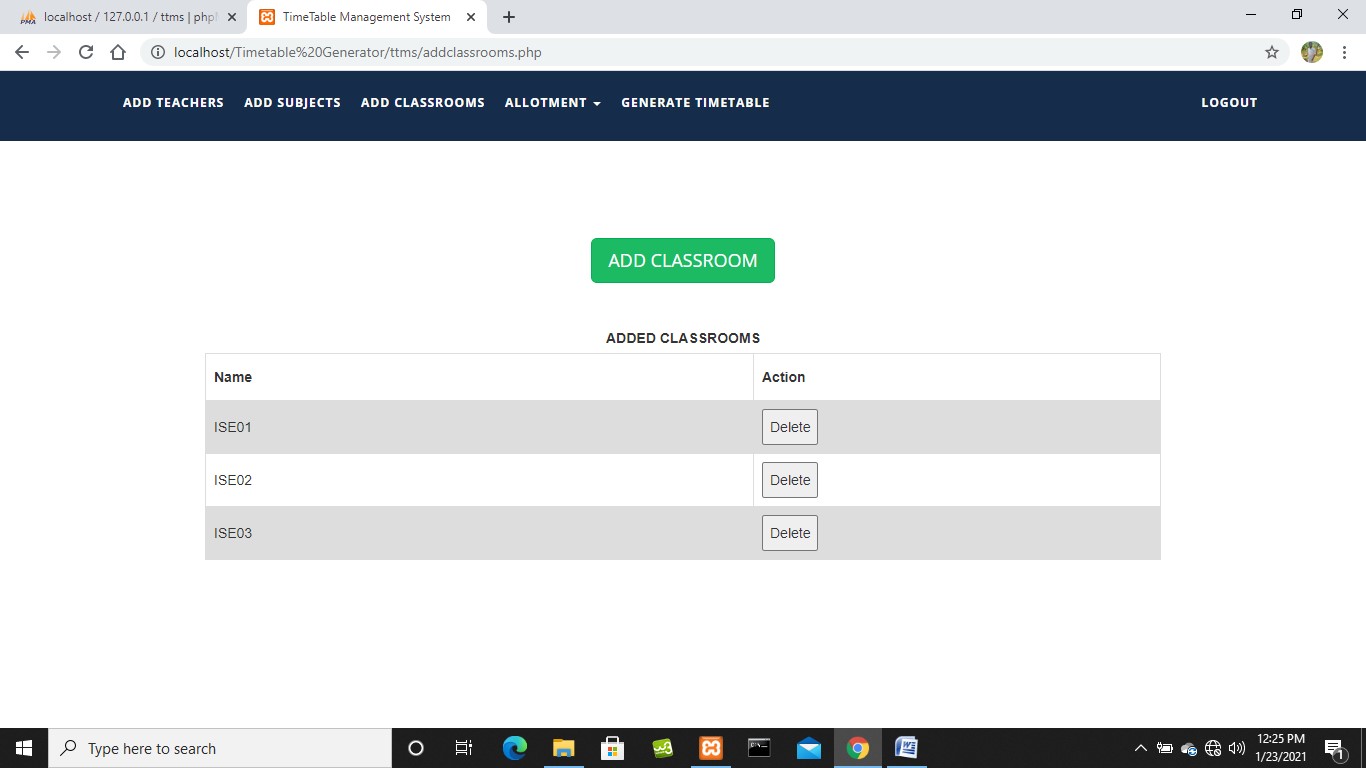
**Fig. 5.6: Subjects Details**

##### 5.7 ADD CLASSROOMS



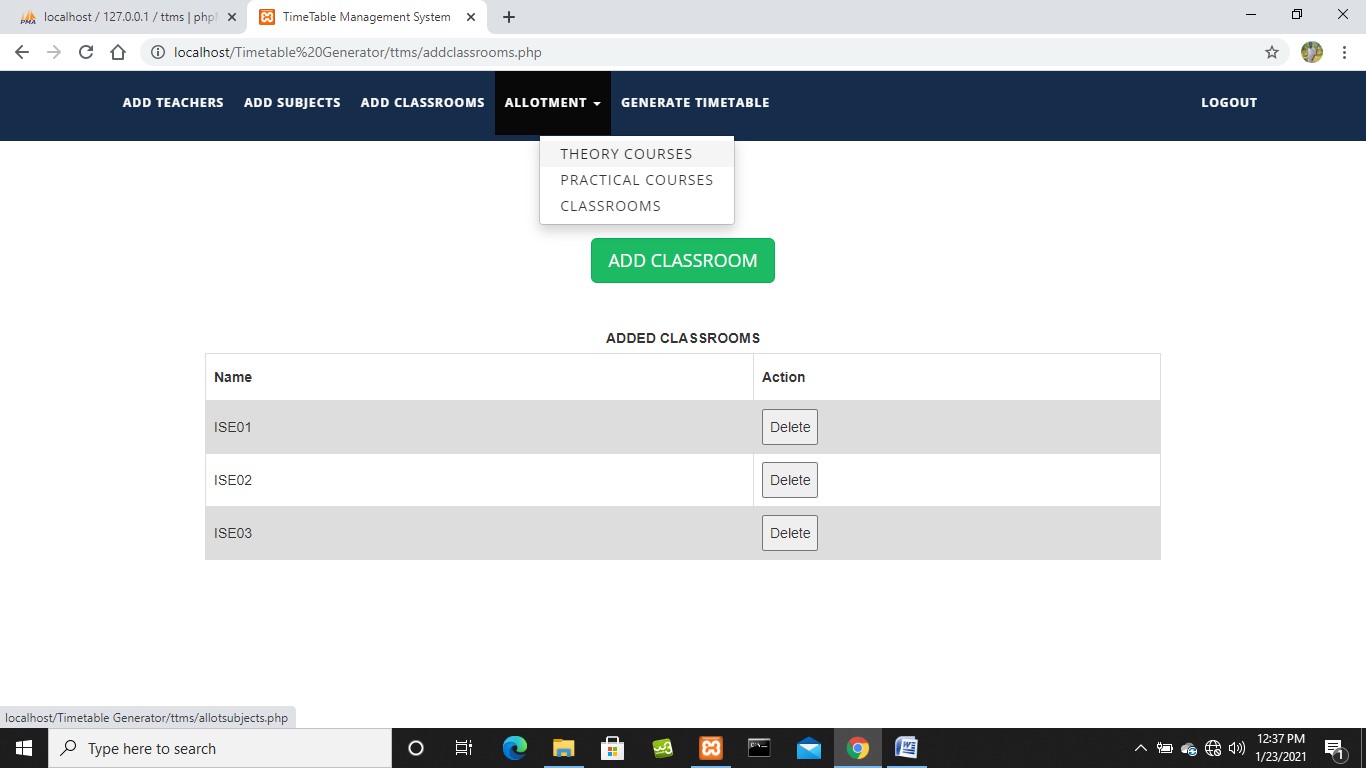
**Fig. 5.7: Add Classrooms**

##### 5.8 CLASSROOMS DETAILS



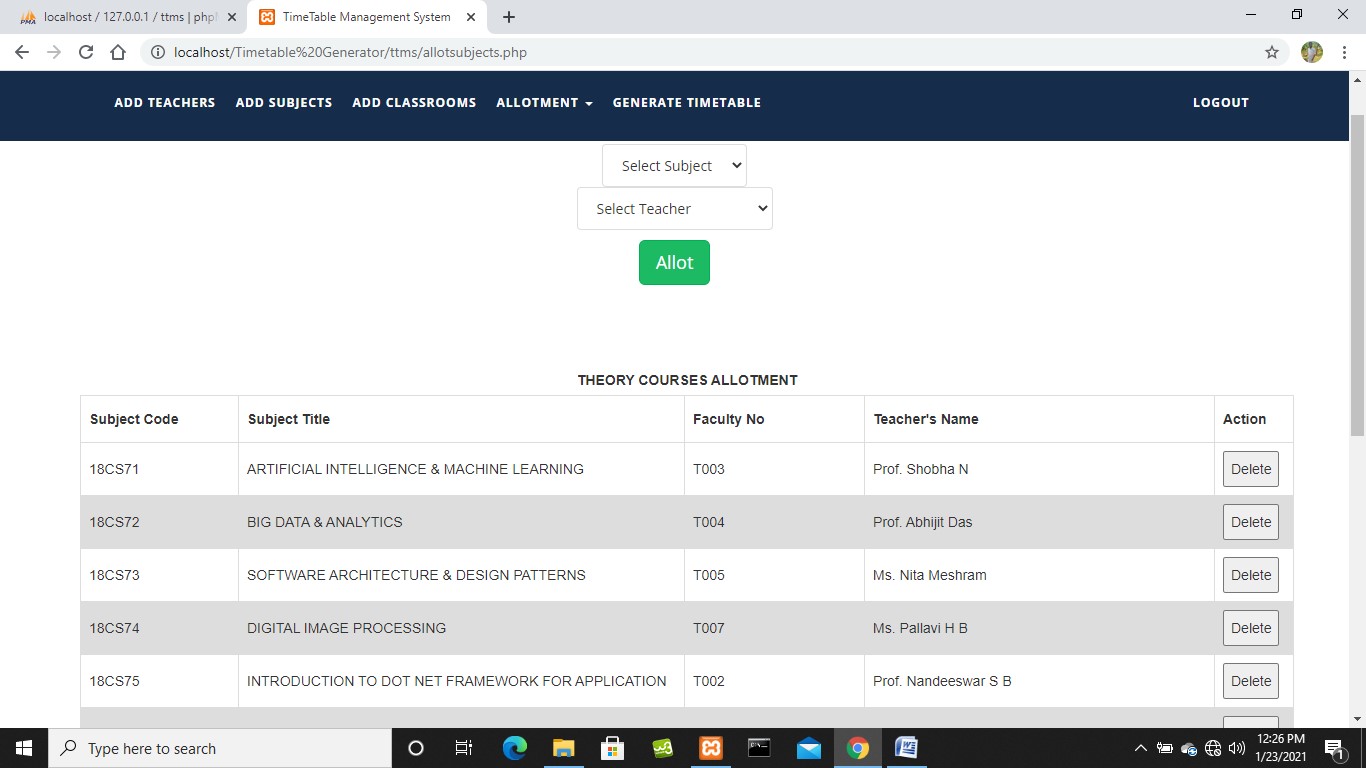
**Fig. 5.8: Classrooms Details**

##### 5.9 ALLOT THEORY SUBJECTS



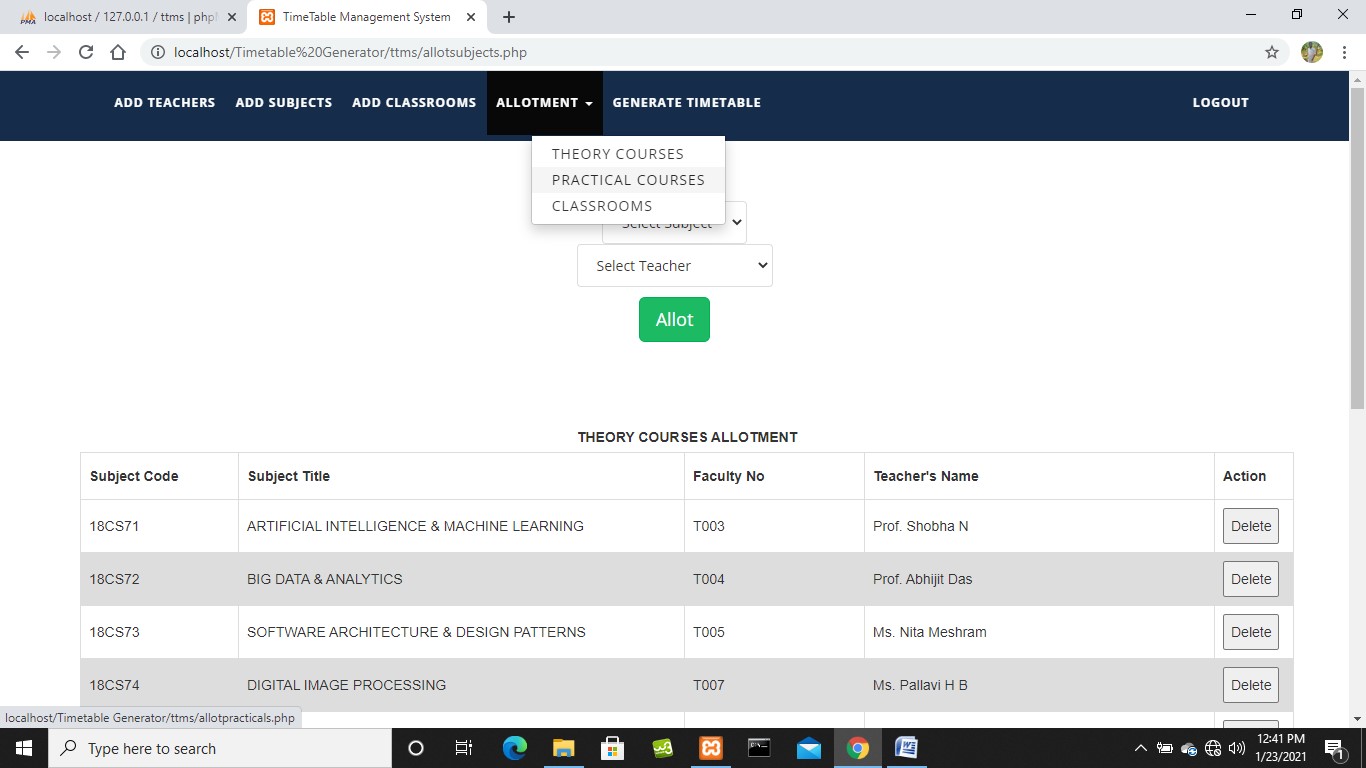
**Fig. 5.9: Allot Theory Subjects**

##### 5.10 THEORY SUBJECTS DETAILS ALLOTTED



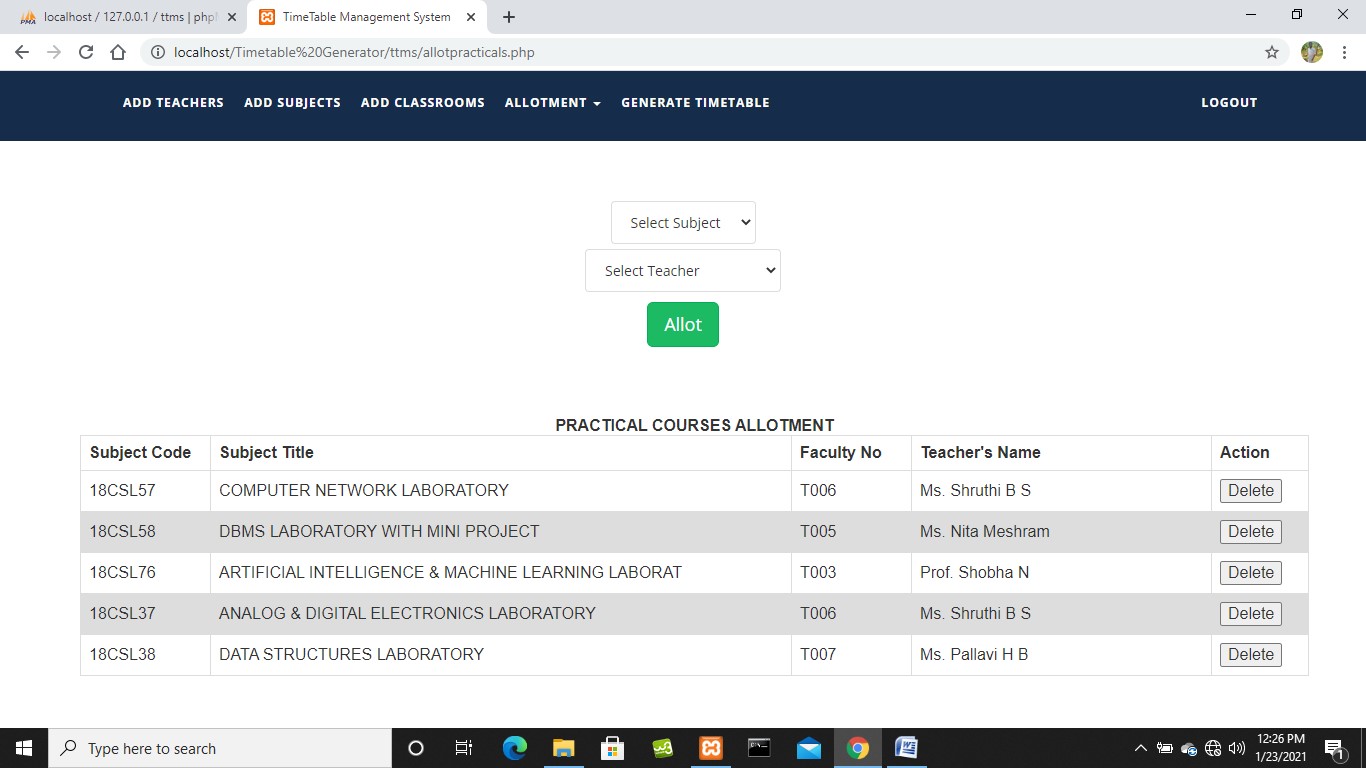
**Fig. 5.10: Theory Subjects Details Allotted**

##### 5.11 ALLOT PRACTICAL SUBJECTS



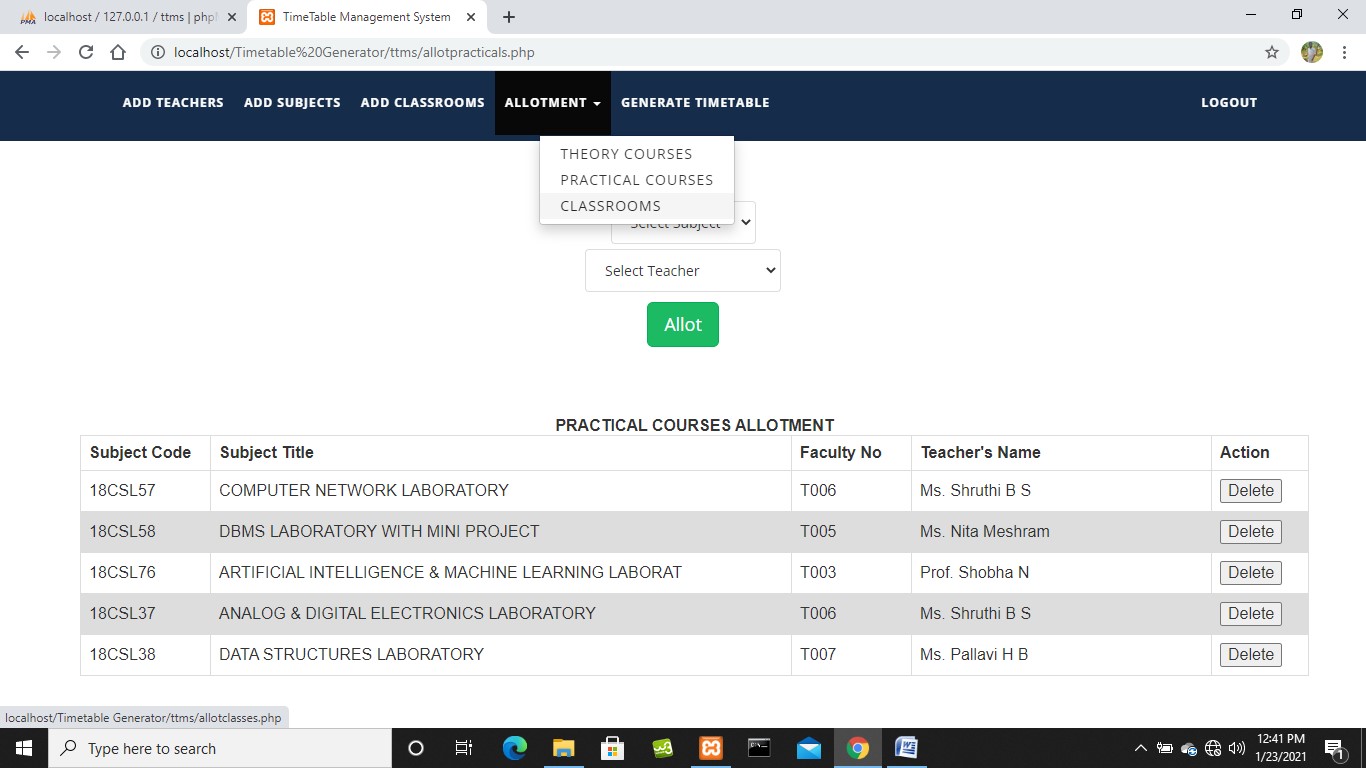
**Fig. 5.11: Allot Practical Subjects**

##### 5.12 PRACTICAL SUBJECTS DETAILS ALLOTTED



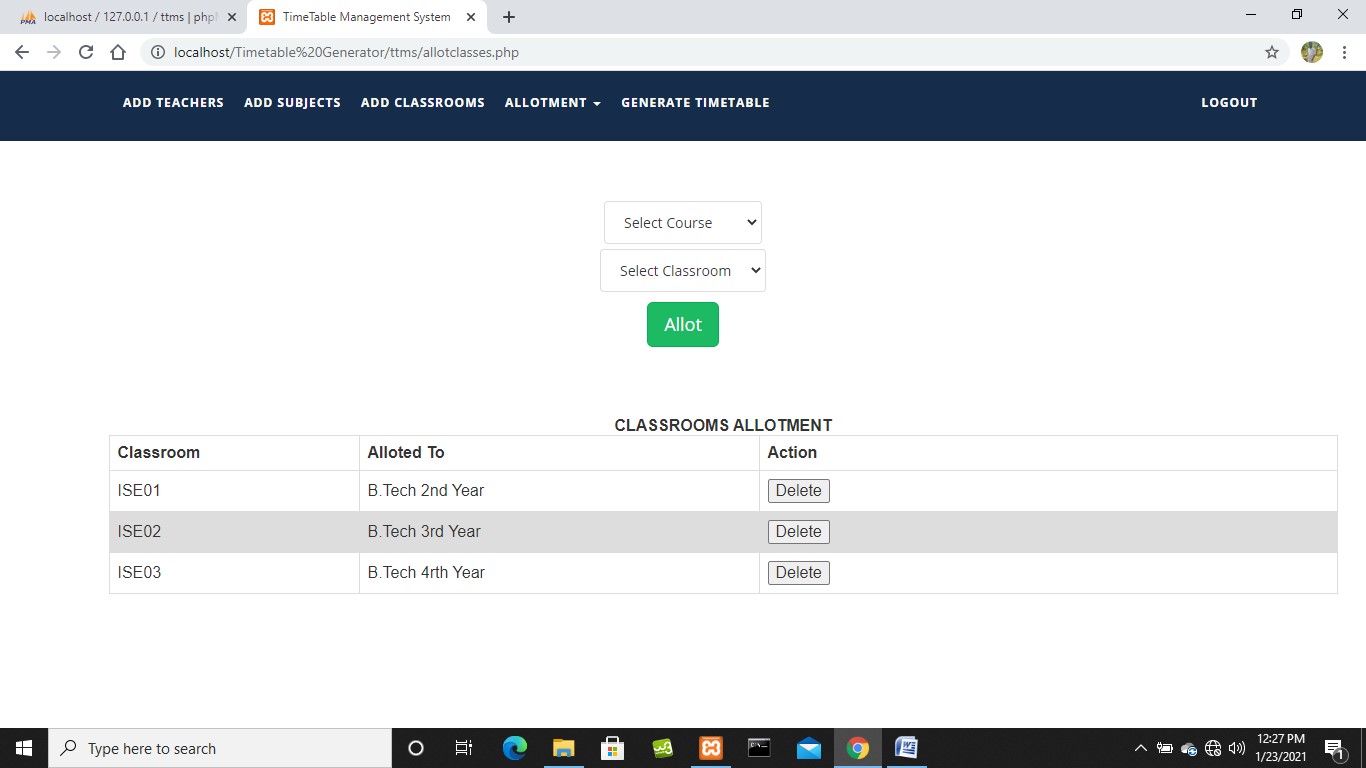
**Fig. 5.12: Practical Subjects Details Allotted**

##### 5.13 ALLOT CLASSROOMS



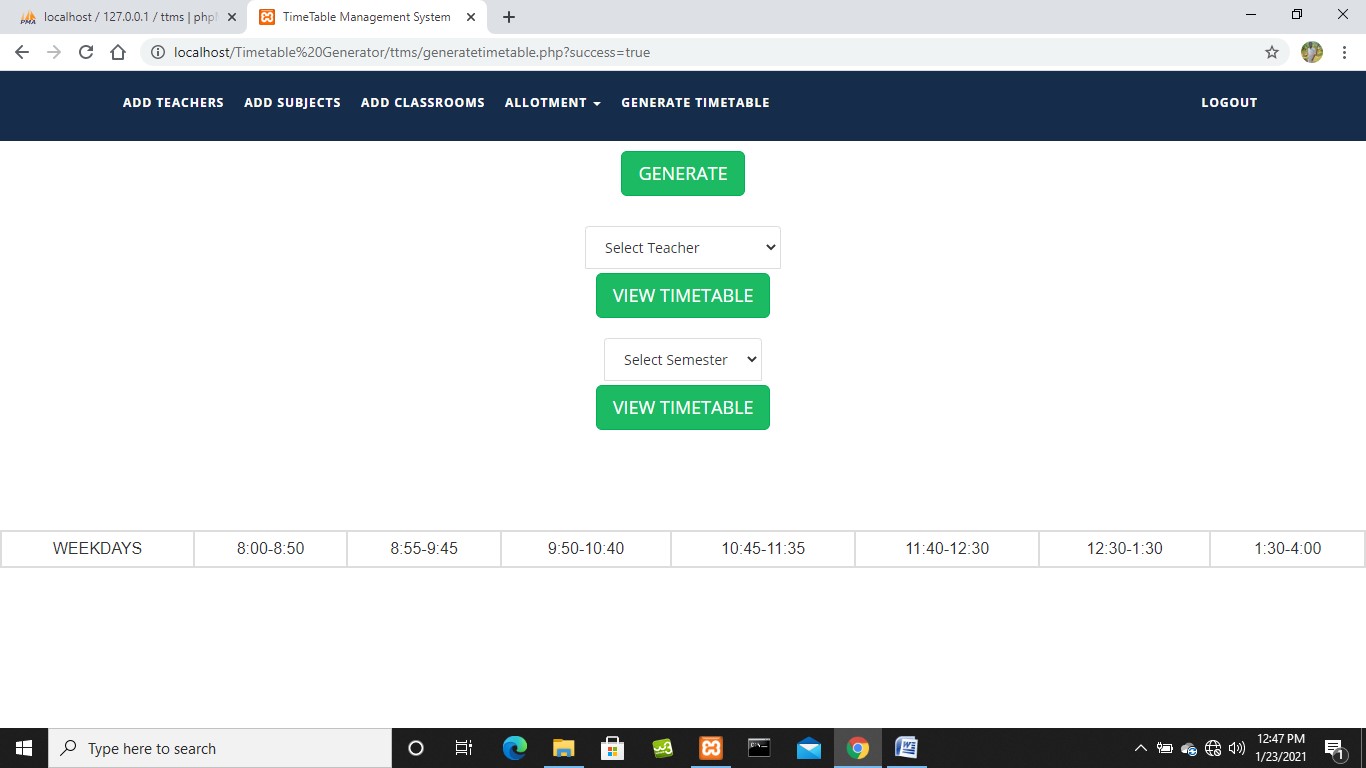
**Fig. 5.13: Allot Classrooms**

##### 5.14 CLASSROOMS DETAILS ALLOTTED



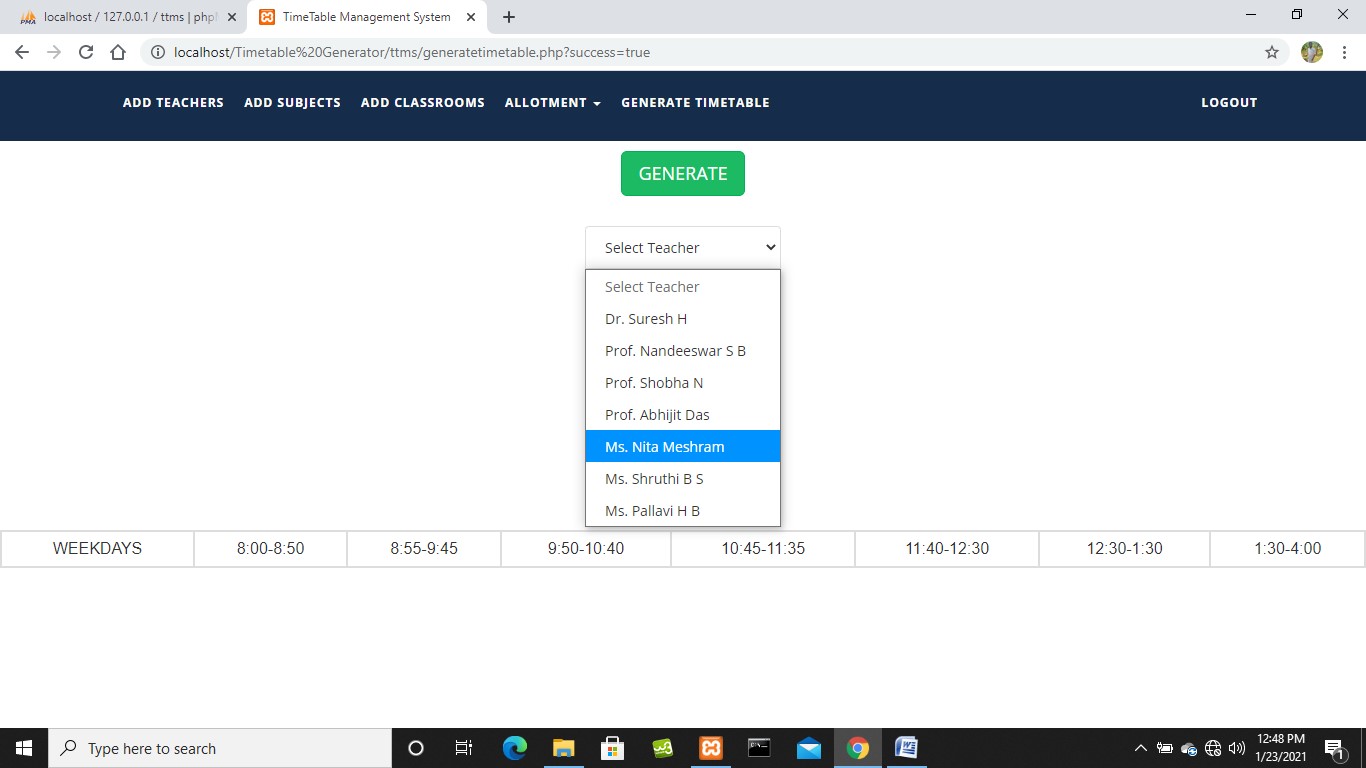
**Fig. 5.14: Classrooms Details Allotted**

##### 5.15 GENERATE TIMETABLE



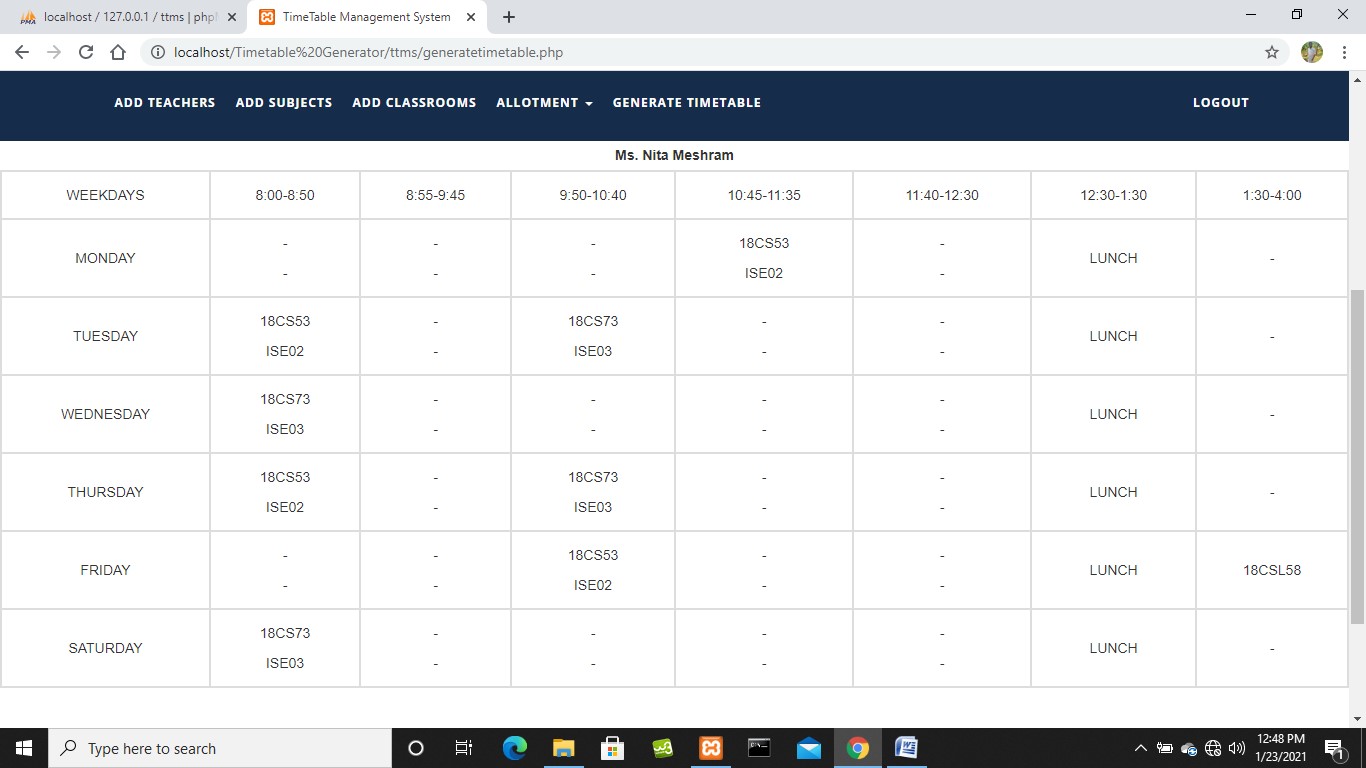
**Fig. 5.15: Generate Timetable**

##### 5.16 SELECT TEACHERS TIMETABLE



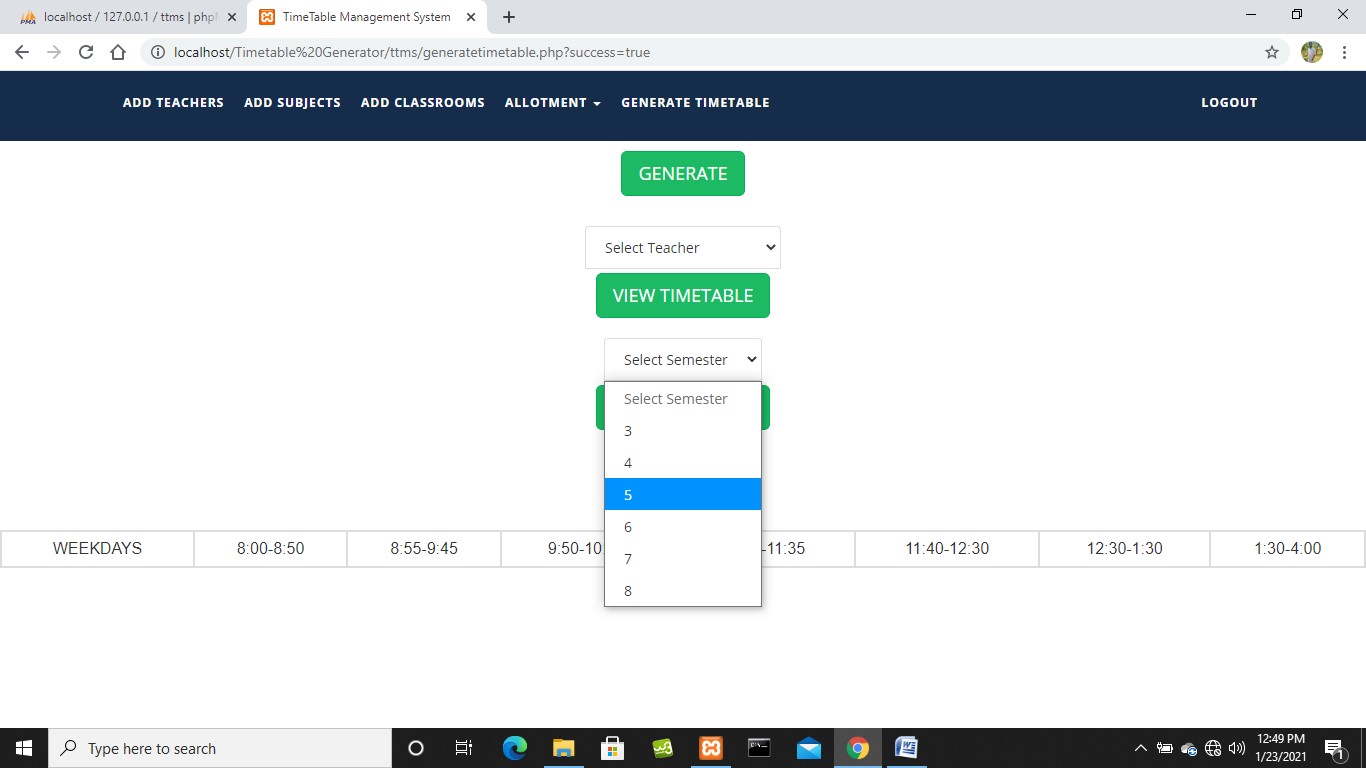
**Fig. 5.16: Select Teachers Timetable**

##### 5.17 VIEW TEACHERS TIMETABLE



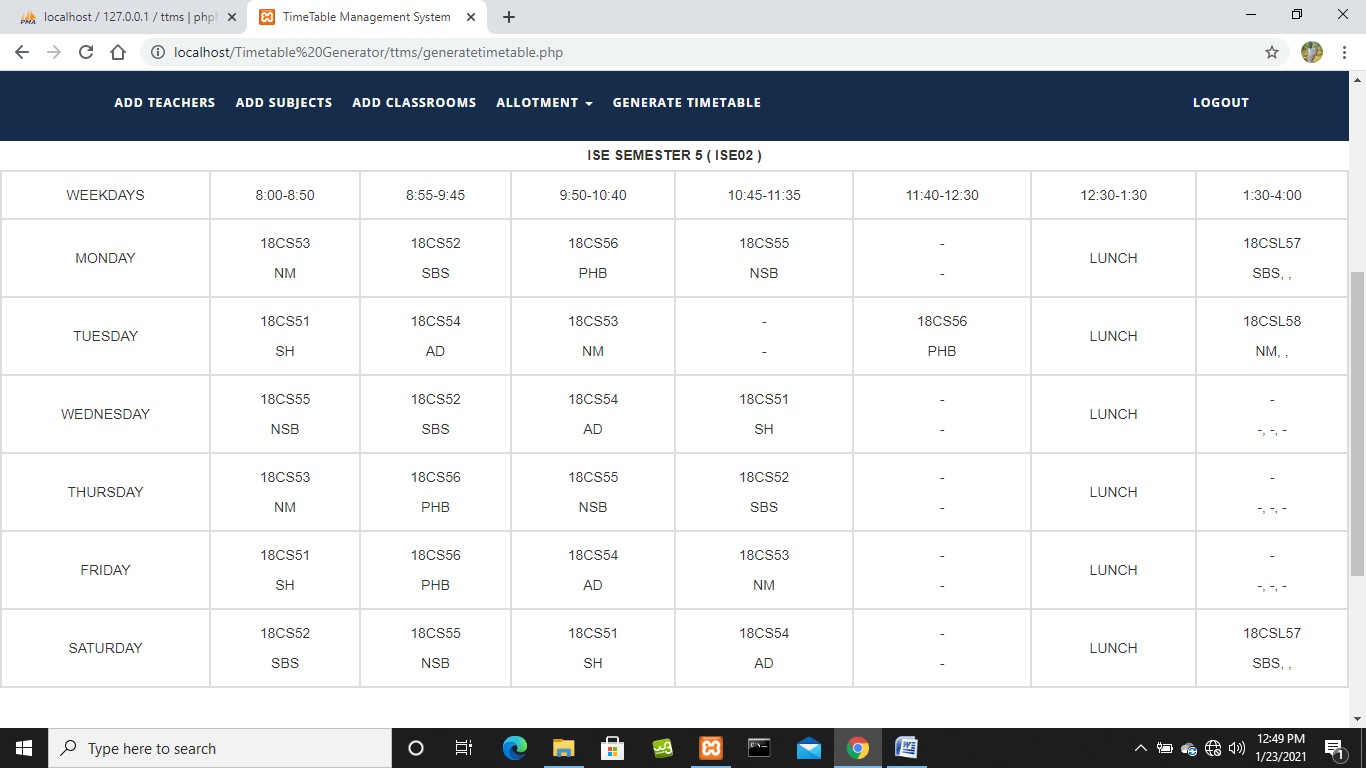
**Fig. 5.17: View Teachers Timetable**

##### 5.18 SELECT SEMESTER TIMETABLE



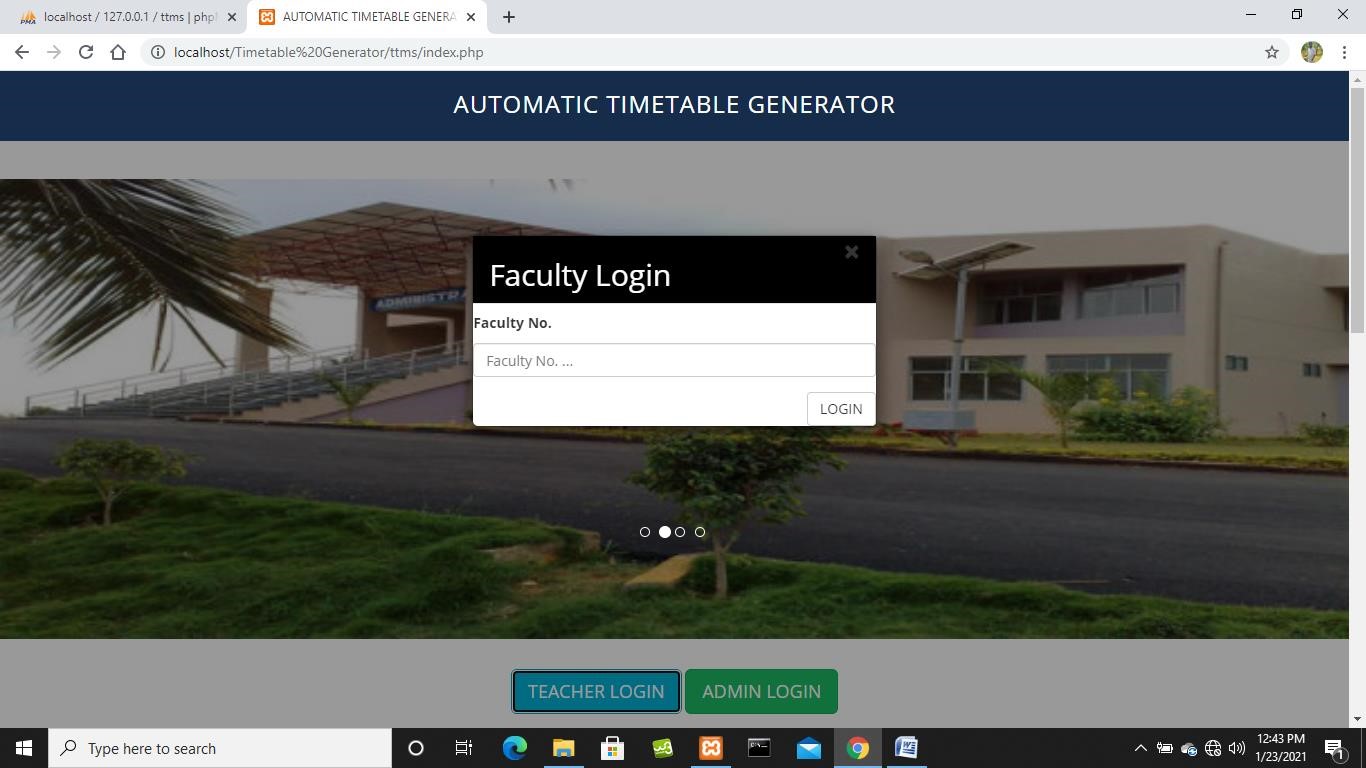
**Fig. 5.18: Select Semester Timetable**

##### 5.19 VIEW SEMESTER TIMETABLE



**Fig. 5.19: View Semester Timetable**

##### 5.20 FACULTY LOGIN PAGE



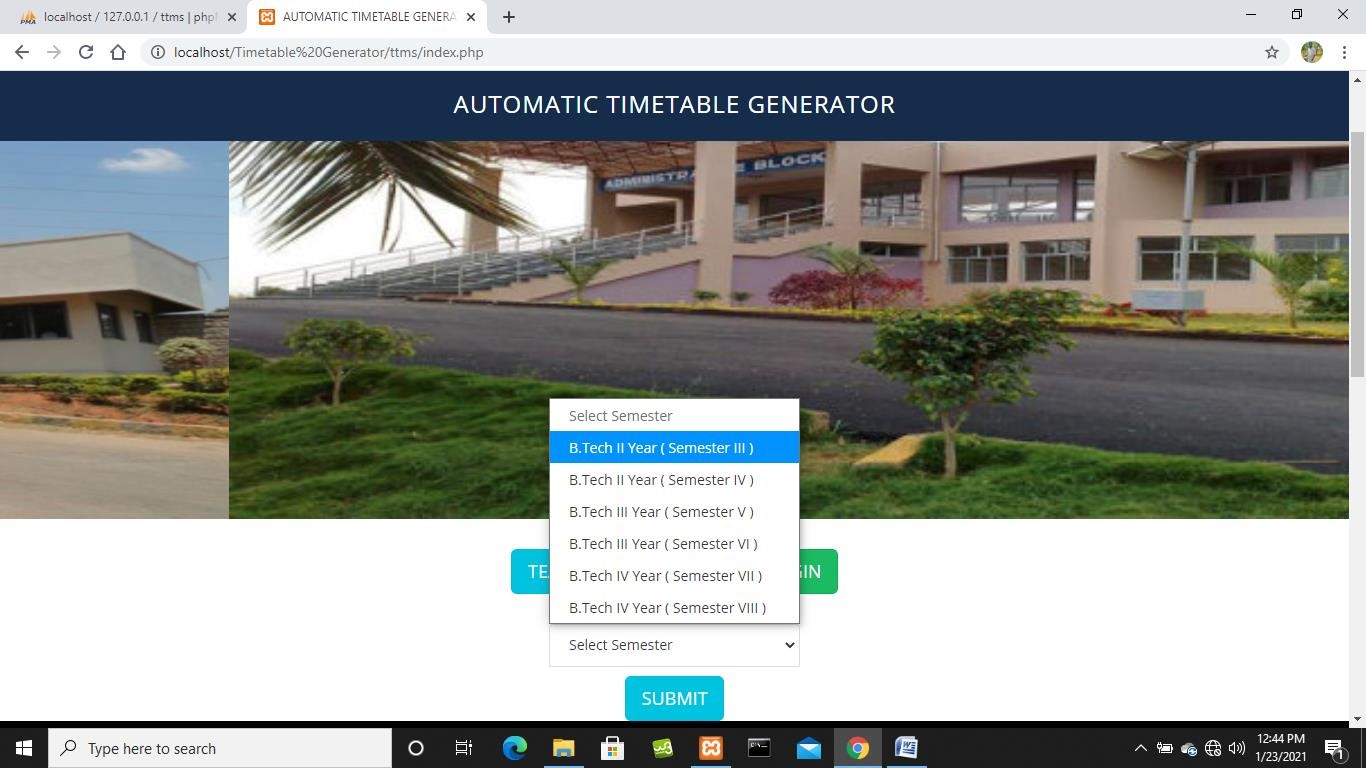
**Fig. 5.20: Faculty Login Page**

##### 5.21 FACULTY PAGE



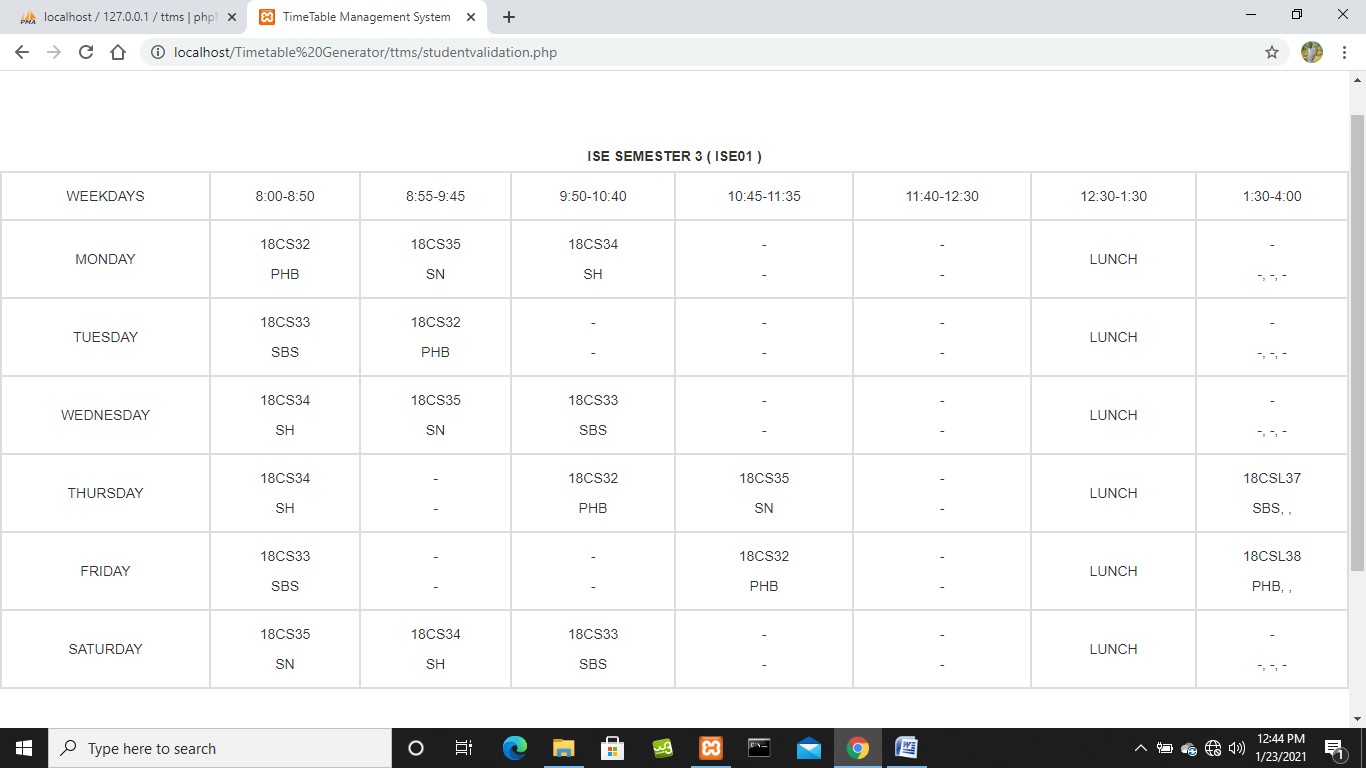
**Fig. 5.21: Faculty Page**

##### 5.22 STUDENT LOGIN PAGE



**Fig. 5.22: Student Login Page**

##### 5.23 STUDENT PAGE



**Fig. 5.23: 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.

#### FUTURE ENHANCEMENT

In our system, there are some problems those are User has to format it a bit after it is prepared. The transaction are 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

* Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson
* XAMPP tutorial on how to create database - <https://www.tutorialspoint.com/index.htm>
* PHP project models and databases
* References for HTML, PHP, CSS and JavaScript in W3Schools and TutorialsPoint -<https://www.tutorialspoint.com/index.htm>and <https://www.w3schools.com/>
* General information in Google and YouTube -<https://www.youtube.com/>