A

Mini-Project Report on

Time Table Generator using Django Framework

Submitted in partial fulfillment of the requirements
for the degree of
BACHELOR OF ENGINEERING
IN
Computer Science & Engineering
Artificial Intelligence & Machine Learning

By

Samiksha Patil (22106041)

Nikita Patil (22106081)

Rohan Patil (22106044)

Jay Wadnere (22106105)

Under the guidance of

Prof. Viki Patil

Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)
A. P. Shah Institute of Technology
G. B. Road, Kasarvadavali, Thane (W)-400615
University Of Mumbai
2023-2024

A. P. SHAH INSTITUTE OF TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled "**Time Table using Django Framework**" is a bonafide work of Samiksha Patil (22106041), Nikita Patil (22106081), Rohan Patil (22106044), Jay Wadnere (22106083) submitted to the University of Mumbai in partial fulfillment of the requirement for the award **of Bachelor of Engineering in Computer Science & Engineering (Artificial Intelligence & Machine Learning).**

____
Prof. Viki Patil
Mini Project Guide

____
Dr. Jaya Gupta
Head of Department

A. P. SHAH INSTITUTE OF TECHNOLOGY

Project Report Approval

This Mini project report entitled “**Time Table Generator using Django Framework”** by Samiksha Patil (22106041), Nikita Patil (22106081), Rohan Patil (22106044), Jay Wadnere (22106105) is approved for the degree of ***Bachelor of Engineering*** in ***Computer Science &Engineering***, (AIML)

2023-24.

External Examiner: ________________

Internal Examiner: _______________

Place: APSIT, Thane
Date:

Declaration

We declare that this written submission represents my ideas in my own words and whereothers' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission hasnot been taken when needed.

Samiksha Patil	Nikita Patil	Rohan Patil	Jay Wadnere
(22106041)	(22106081)	(22106044)	(22106105)

ABSTRACT

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

These are just due to common human errors which are very difficult to prevent in processes such as these. To overcome these problems people usually taking the previous years timetable and modifying it but still it is atedicios job to incoperate changes.

To overcome all these problems we 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. The case is similar to schools and other educational institutions. So our aim is to develop a general purpose which can efficiently generate optimal solutions.

Keywords: Time Table Generator (TTG), Django.

Index

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

Time table scheduling has been in human requirements since they thought of managing time effectively. It is widely used in schools, colleges and other fields of teaching and working like crash courses, couching centers, training programs etc. In early days, time table scheduling was done manually with a single person or some group involved in task of scheduling it with their hands, which take lot of effort and time. While scheduling even the smallest constraints can take a lot of time and the case is even 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 whole generation without any changes, proving to be dull in such situations. Other cases that can cause problem is when the number of employers/workers are weak, resulting in rescheduling of time table. Institutions/Schools/Collages/Universities are the regular users of such timetables. 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 requirement of new course addition 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 batches course start. Another problem that occur when scheduling time table for exams. When multiple batches have exam on same day, they need to be schedules effectively taking into account all problems related to facilities that are available to conduct these exams simultaneously.

Objectives:

Optimizing resource utilization: The primary objective of a timetable generator is to efficiently allocate resources such as classrooms, teachers, and equipment to various activities or events, maximizing their utilization.

Minimizing conflicts: Time table generators aim to minimize conflicts by scheduling tasks or events in such a way that overlaps, clashes, or double bookings are avoided, ensuring smooth operation and minimal disruptions.

Meeting constraints: Timetable generators adhere to various constraints such as room capacities, teacher availability, student preferences, and institutional policies, ensuring that schedules are compliant with specific requirements and regulations.

Enhancing productivity: By automating the scheduling process, timetable generators help save time and effort, allowing administrators, educators, or managers to focus on more strategic tasks, thereby enhancing overall productivity.

Improving efficiency: Timetable generators optimize the allocation of time slots, sequences, and resources, leading to more efficient use of available time and resources, ultimately improving organizational efficiency.

CHAPTER 2
LITERATURE SURVEY

2. LITERATURE SURVEY

2.1. HISTORY

The history of time table generators dates back to the increasing complexity of scheduling demands in various domains, primarily in educational institutions and workplaces. While the exact origins may not be pinpointed, the development of these generators can be traced through key milestones:
In the late 1990s and early 2000s, the advent of web frameworks such as Django revolutionized the development of web applications. Django, with its robust features and rapid development capabilities, became a popular choice for building complex and scalable web-based systems, including vehicle service management systems.
Literature on vehicle service management systems using Django framework emerged as organizations recognized the benefits of leveraging modern web technologies for streamlining operations and improving efficiency. Research and case studies highlighted the advantages of using Django for developing vehicle service management systems, such as its modular architecture, built-in security features, and extensive community support.

Over the years, the literature on vehicle service management systems has evolved to cover various aspects, including:

1. System Architecture: Studies have explored different architectural patterns and design principles for building scalable and maintainable vehicle service management systems using Django.
2. Data Management: Research has focused on best practices for managing vehicle data, including storing, querying, and analyzing data efficiently using Django's ORM (Object-Relational Mapping) capabilities.
3. User Interface Design: Literature has emphasized the importance of user-centered design principles for developing intuitive and user-friendly interfaces for vehicle service management systems, leading to improved user satisfaction and productivity.

2.2-LITERATURE REVIEW

Algorithmic Approaches:

Explore various algorithms used in timetable generation, such as genetic algorithms, simulated annealing, or heuristic methods.

Assess the effectiveness of different algorithms in producing optimal or near-optimal schedules.

User Requirements and Preferences:

Investigate studies addressing the diverse needs of users, considering factors like room availability, teacher preferences, and student constraints.

Real-World Applications:

Automated timetable generators find applications in various real-world scenarios, including educational institutions, transportation systems, healthcare facilities, and corporate environments. In educational institutions, these tools are used to schedule classes, exams, extracurricular activities, and other academic events, optimizing resource utilization and enhancing productivity. Similarly, in transportation systems, automated timetable generators play a vital role in scheduling routes, optimizing vehicle assignments, and minimizing travel times, thereby improving service reliability and customer satisfaction.

The literature reviewed underscores the importance of automated timetable generators in efficiently scheduling tasks and activities across diverse domains. Algorithmic approaches, constraint satisfaction techniques, hybrid methodologies, and real-world applications demonstrate the versatility and effectiveness of automated timetable generators in optimizing resource allocation, minimizing conflicts, and meeting scheduling constraints. Future research directions may focus on developing advanced algorithms, integrating machine learning techniques, and addressing emerging challenges in automated timetable generation

Literature Survey:

A writing review is an evaluative report of data set up in the proposed writing work. A writing survey may be an essential piece of the plan when leading an investigation because it covers all for starters dove the subject and sets the stage for current investigation. It is the most important part of your report because it guides you in the area of your investigation. It aids you in deciding what to study for your exam. "Time Table Planning Using Inherited Fake Safe Organization" suggests Booking is one of the most significant undertakings encountered in actuality. Beautiful planning issues, such as workforce booking, item booking, schooling plan, and so on, are available. Instructive timetable planning can also be a delicate task due to the various constraints that must be met in order to produce a feasible outcome in a timely manner. Procedures such as inheritable Calculations GAs) have been used with mixed success

(We examined our foundation's booking arrangement and previously attempted to work it with the inheritable IJCRT2304429 International Journal of Creative Research Thoughts (IJCRT) d590www.ijcrt.org 2023 IJCRT | Volume 11, Issue 4 April 2023 | ISSN: 2320-2882calculation.

A. Timetabling

It is large and highly constrained, but above all the problem differs greatly for diverse colleges and learning institutions. It is hard to write a universal agenda, fitting for all possible timetable problems. Even though manual creation of timetable is sustained, it is still universal, because of the lack of suitable computer programs (Gore & Poonam Sonawane, March 2017). Timetable problems

There exist a lot of diverse timetable problems such as:

- University Timetable
- Exam Timetable
- School Timetable
- Sports Timetable
- Worker Timetable

Moreover, there exist a lot of problem solving methods, which typically use the concept of customary optimization algorithms such as genetic algorithms, Backtracking, Constraint Logic Programming.

In recent years two major approaches appear to have been victorious.

- Local Search Procedures
- Constraint Programming (CP) (Gore & Poonam Sonawane, March 2017).

B. The Local Search Procedures The local search measures such as Simulated Annealing, Tabu Search and Genetic Algorithms. These method convey constraints as various cost functions, which are minimized by a Heuristic Search of enhanced solution in a neighborhood of some opening realistic result (Gore & Poonam Sonawane, March 2017).

1. Simulated annealing (SA) Simulated annealing is a probabilistic method used for similar to the global optimum of a given function. Purposely, it is a metaheuristic to fairly accurate global optimization in a huge search space. It is frequently used when the search space is distinct. Simulated annealing is a technique for finding a good result to an optimization dilemma. If there is a condition where we want to maximize or reduce something, our problem can likely be tackle with simulated annealing (Gore & Poonam Sonawane, March 2017).

In the paper {1}, the proposed algorithm can be further adapted to suit as per the requirements of different institutes and universities. It considers a wide range of constraints divided into soft and hard, to generate faculty, classroom/laboratory and student timetables. The complexity of the proposed algorithm is n3. In {2}, the paper presents a categorisation of the methodologies conducted in recent years based on chronology. category and application. {3}We propose two pattern-based formulations and a solution algorithm that simultaneously exploits column generation and a team of metaheuristics to build and improve solutions. {4} Graph Colouring Algorithm (GCA) takes 11 percent of the time, Genetic and Graph Colouring (GCA) takes 25 percent, Heuristic and Iterated Local Search Algorithms (ILSA) take 44 percent of time, and Heuristic Algorithm takes 20 percent. {5}.This means developers can use imperative programming for business logic and declarative programming for user interface (UI) development. {6}. The fitness score relates to the quantity of crashes the timetable has experienced.

CHAPTER 3

Problem Statement

3. PROBLEM STATEMENT

Problem Statement:

Creating a timetable manually can be a time-consuming and error-prone task, especially in educational institutions or workplaces with complex scheduling requirements. The need for an automated Time Table Generator arises to efficiently allocate resources, such as rooms, faculty, and time slots, while adhering to various constraints and preferences.

Solution:

Develop a Time Table Generator application that takes into account the following aspects:

Input Parameters -
Number of days in the week.
Available time slots for classes.
List of subjects and their respective durations.
Faculty availability and preferences.
Room availability and capacities.

Constraints -
Avoid clashes in faculty schedules.
Ensure room capacity matches the class size.
Accommodate special requests or constraints.

CHAPTER 4

Experimental Setup

4. Experimental Setup

4.1 Hardware Setup

1. Networking Equipment: - Network switch or router to connect your server to the local network or the internet. 2. Laptop

4.2 Software Setup

Django Framework: - Install Django framework on your server. You can use pip, the Python package installer, to install Django: pip install Django.

Development Tools: - Install development tools such as IDEs (e.g., PyCharm, Visual Studio Code), text editors (e.g., Sublime Text, Atom), and debugging tools to facilitate development and troubleshooting.

Testing and Debugging: - Set up testing frameworks like Django's built-in test framework, pytest, or unittest for automated testing of your Django application. - Use Django Debug Toolbar or similar tools for debugging and optimizing your application's performance

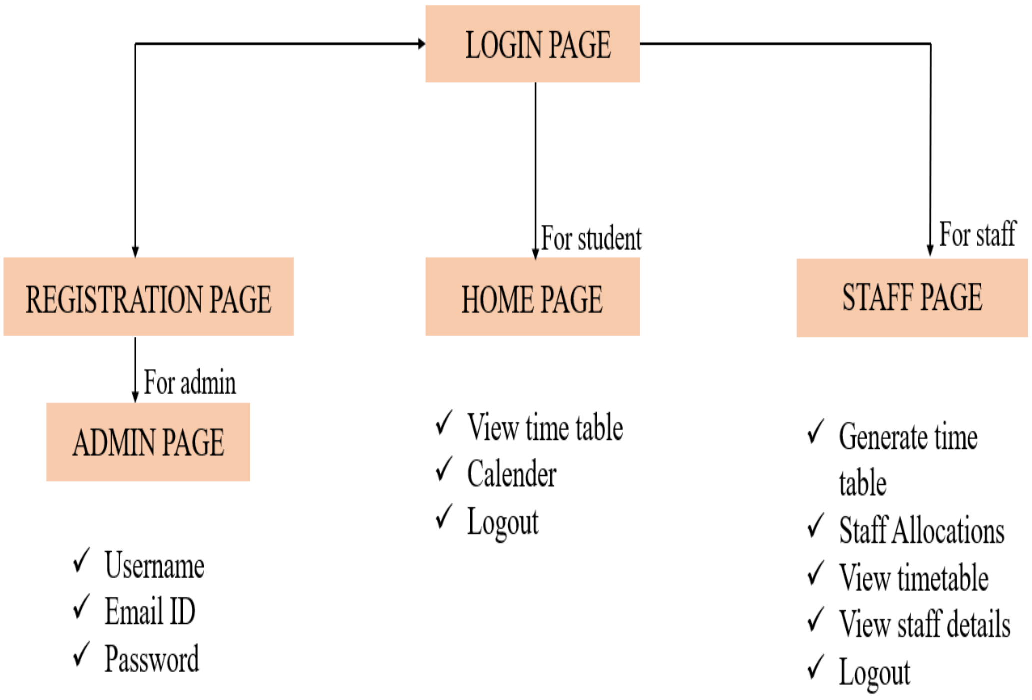
CHAPTER 5

PROPOSED SYSTEM

& IMPLEMENTATION

5.Proposed System & Implementation

5.1 Block diagram of proposed system



Registration Page:

The registration page allows new users, such as administrators or staff members, to create accounts for accessing the system.

Users typically provide basic information such as name, email address, and password.

Additional fields may include contact information, role within the institution (e.g., administrator, teacher), and any other relevant details.

The registration process may also include email verification or captcha to ensure security and prevent spam accounts.

Login Page:

The login page is where registered users can enter their credentials to access the system.Users typically input their email address or username along with their password.

The system verifies the credentials against the stored data in the database and grants access upon successful authentication.

Home Page:

The home page serves as the main dashboard or landing page upon logging into the system.

It provides an overview of important information and navigation options for accessing various features.

Common components of the home page may include:

Quick links to key functions such as adding teachers, rooms, courses, etc.

Summary of upcoming events, schedules, or announcements.

Navigation menu for accessing different sections of the system.

Search functionality to quickly find specific information within the system.

Personalized greetings or messages based on the user's role or preferences.

Notifications or alerts for any important updates or reminders.

Add Teachers:

This feature allows administrators or authorized users to add new teachers or instructors to the system.

Users typically input details such as name, contact information, qualifications, and assigned subjects or courses.

Additional functionalities may include uploading profile pictures, setting availability, and assigning teaching preferences.

The system may also provide options for managing existing teacher profiles, such as editing, deleting, or archiving.

Add Rooms:

This feature enables administrators to add new rooms or facilities available within the institution, such as classrooms, laboratories, or meeting rooms.

Users input information such as room number, capacity, facilities/equipment available, and any specific attributes (e.g., wheelchair accessibility).

Additional functionalities may include categorizing rooms by type, assigning room numbers, and managing room availability.

Add Timings:

This feature allows administrators to define and manage timing schedules for classes, meetings, or other events.

Users input details such as start time, end time, duration, and recurrence patterns (e.g., daily, weekly).

The system may provide options for setting up custom time slots, specifying break times, and managing exceptions (e.g., holidays, special events).

Add Courses:

This feature enables administrators to add new courses or subjects offered by the institution.

Users input details such as course name, code, description, prerequisites, and credit hours.

Additional functionalities may include categorizing courses by department, assigning instructors, and managing course enrollments.

Add Departments:

This feature allows administrators to create and manage departments or academic units within the institution.

Users input details such as department name, description, and contact information.

Additional functionalities may include assigning department heads, managing faculty members, and setting departmental goals or objectives.

Add Sections:

This feature enables administrators to create and manage sections or groups within courses.

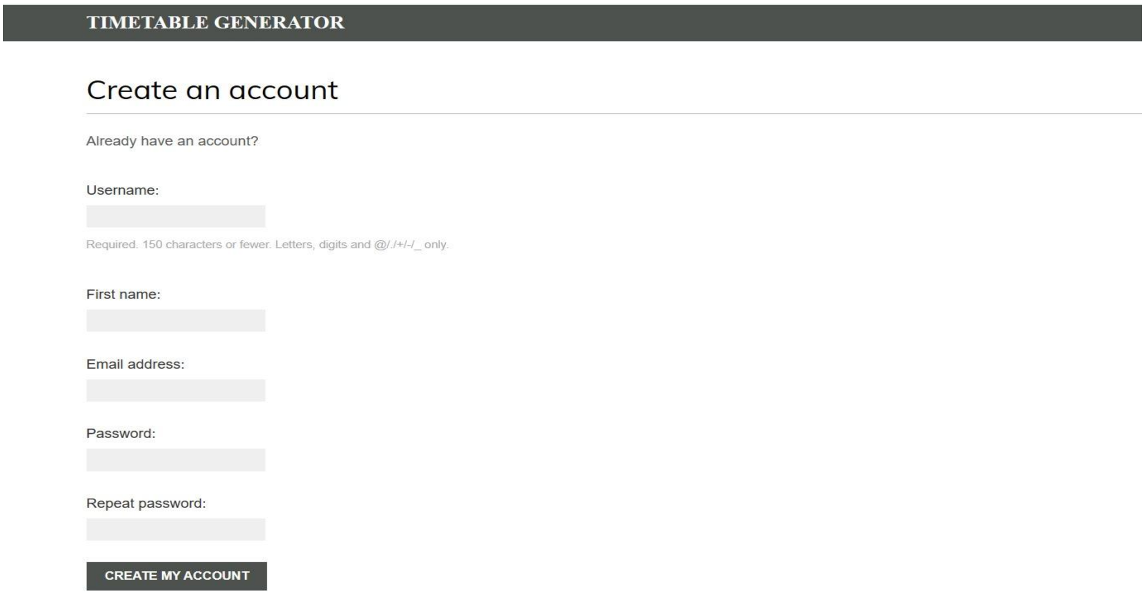
Users input details such as section name, code, capacity, and meeting times.

Additional functionalities may include assigning instructors, managing student enrollments, and setting section-specific requirements or activities.

Each of these components plays a vital role in the overall functionality of the system, facilitating efficient management of resources, schedules, and academic operations within the institution.

5.3 Implementation

1) Registration Page



TIMETABLE GENERATOR

Create an account

Already have an account?

Username:

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

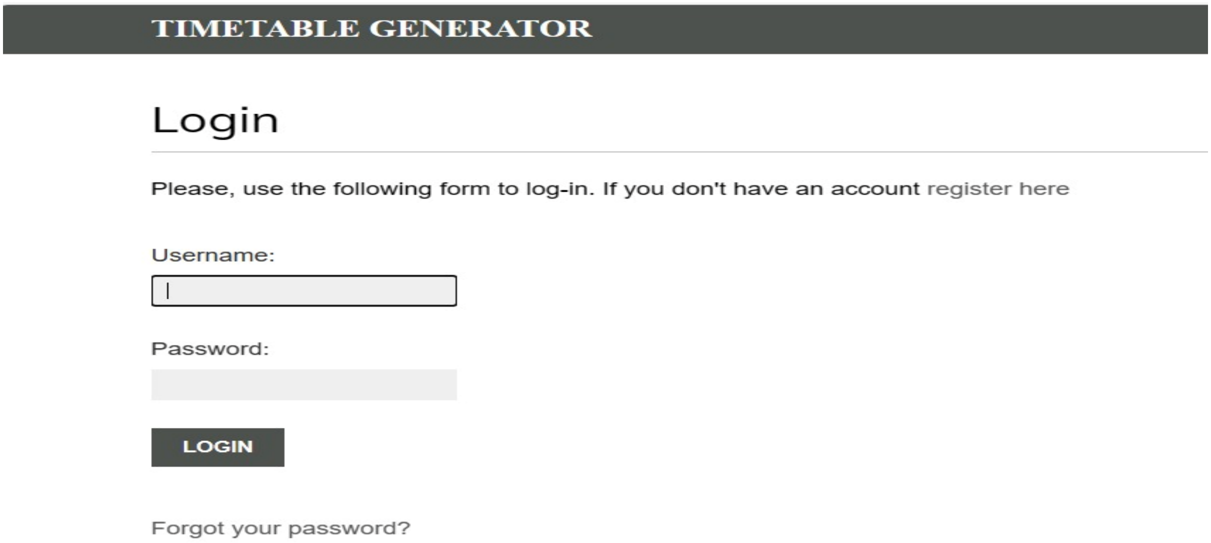
First name:

Email address:

Password:

Repeat password:

2) Login Page



TIMETABLE GENERATOR

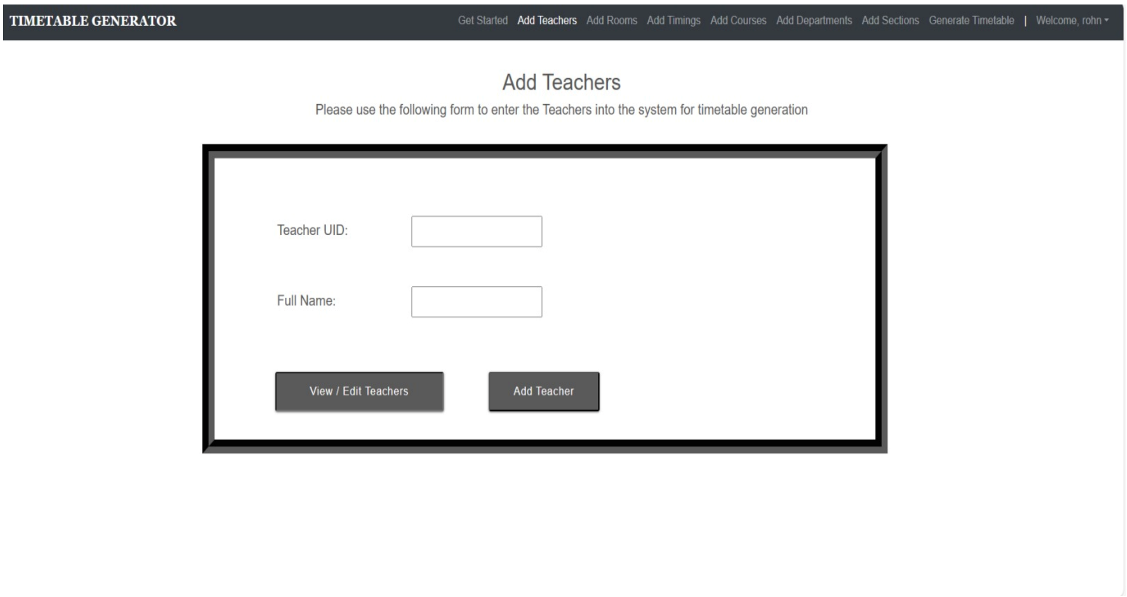
Login

Please, use the following form to log-in. If you don't have an account register here

Username:

Password:

3) Home Page

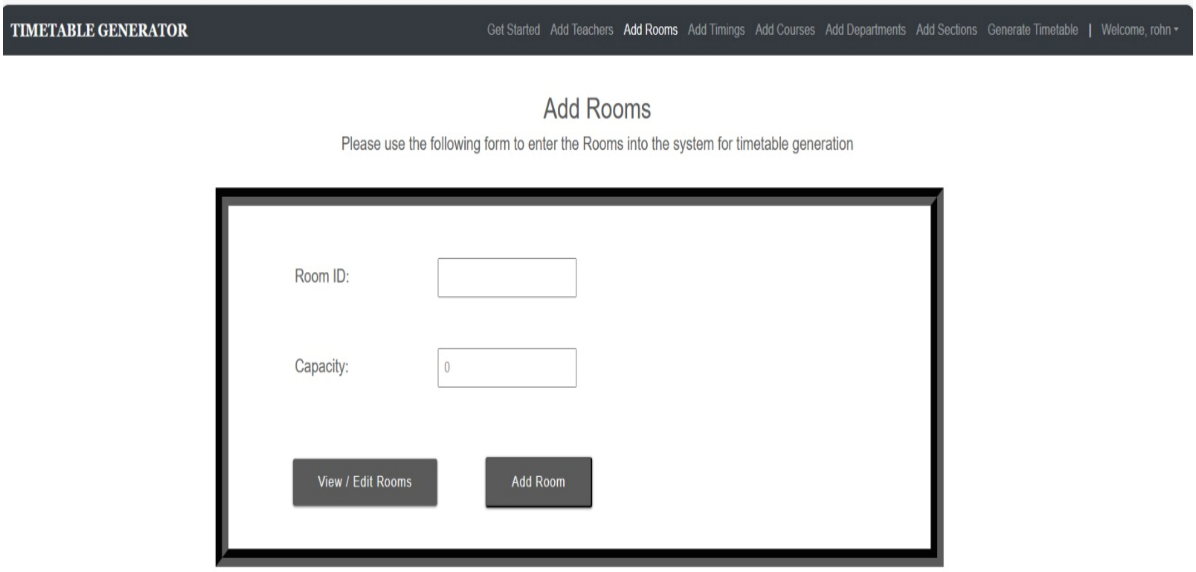


Add Teachers

Please use the following form to enter the Teachers into the system for timetable generation

Teacher UID:

Full Name:

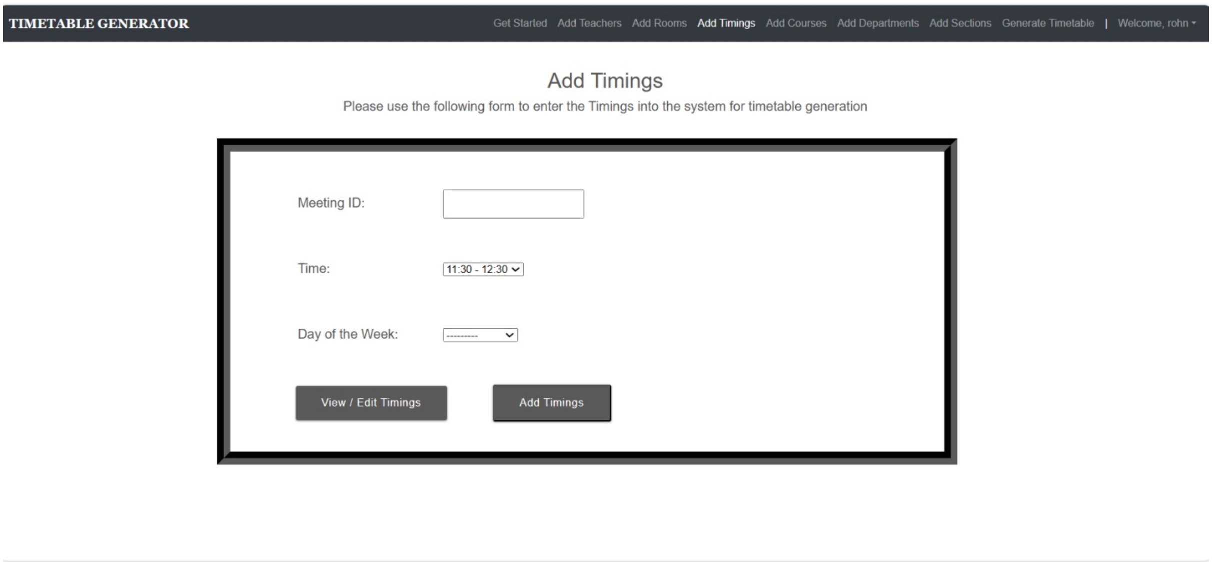


Add Rooms

Please use the following form to enter the Rooms into the system for timetable generation

Room ID:

Capacity: 0



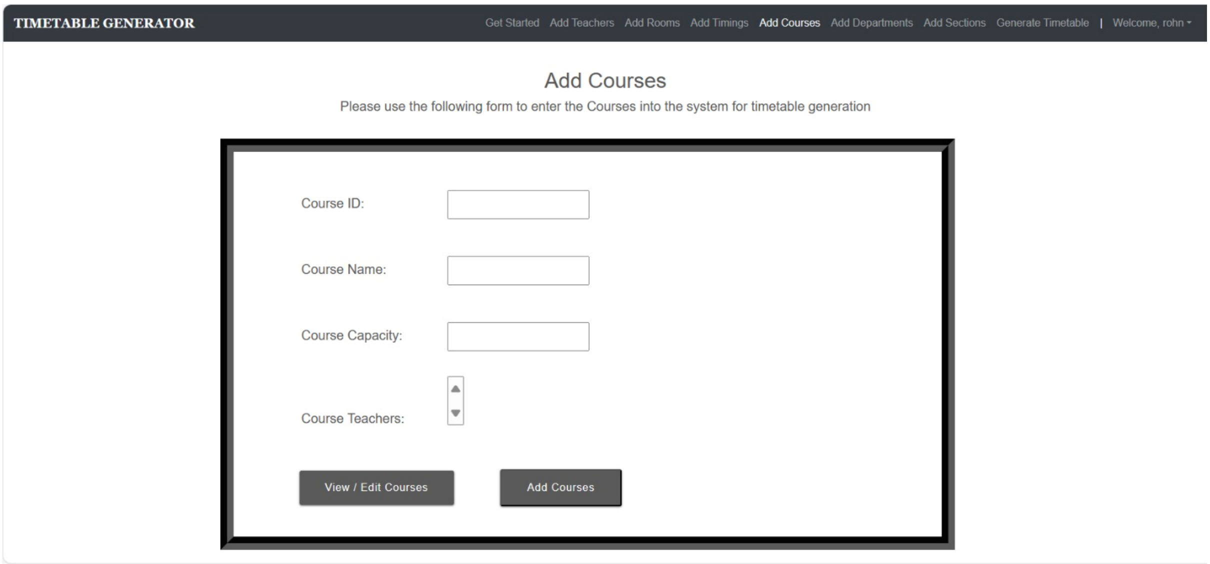
Add Timings

Please use the following form to enter the Timings into the system for timetable generation

Meeting ID:

Time: 11:30 - 12:30

Day of the Week: ---------



Add Courses

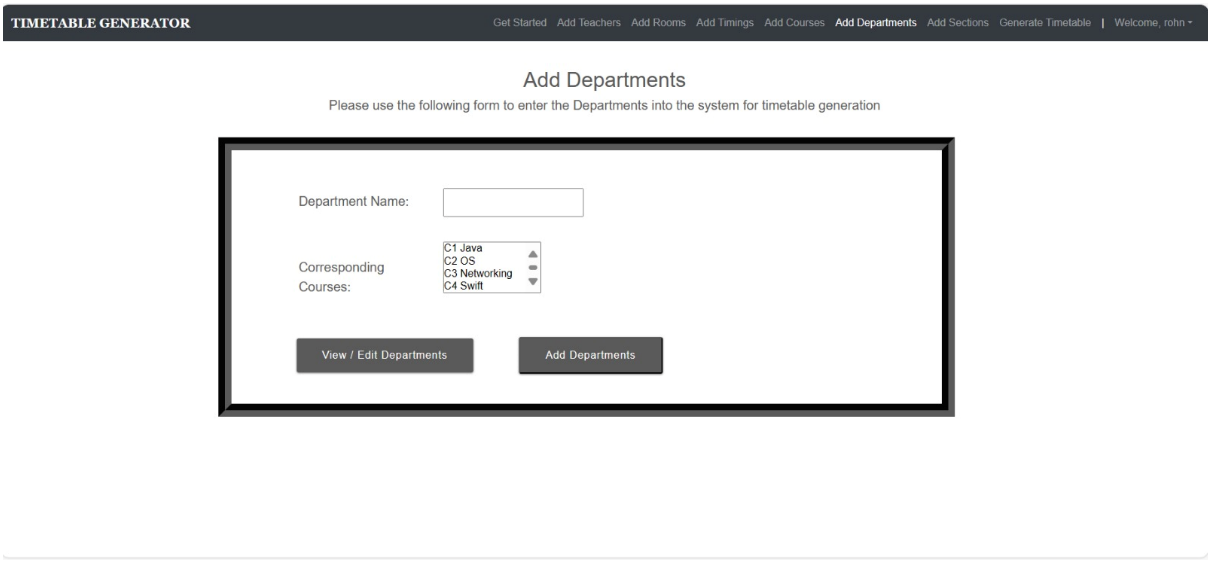
Please use the following form to enter the Courses into the system for timetable generation

Course ID:

Course Name:

Course Capacity:

Course Teachers:

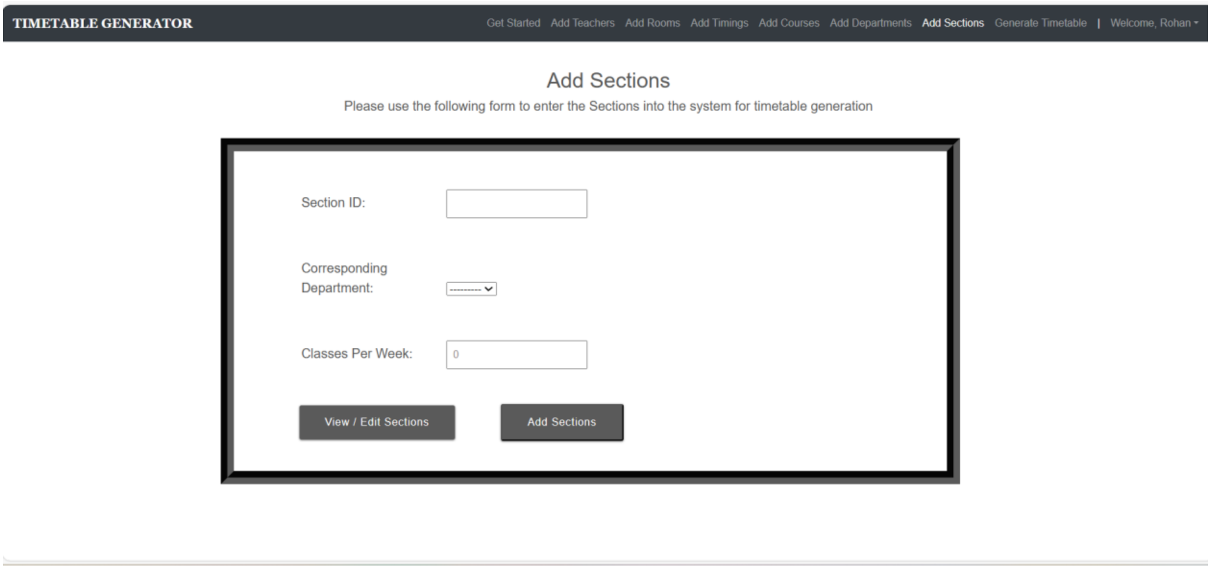


Add Departments

Please use the following form to enter the Departments into the system for timetable generation

Department Name:

Corresponding Courses: C1 Java, C2 OS, C3 Networking, C4 Swift



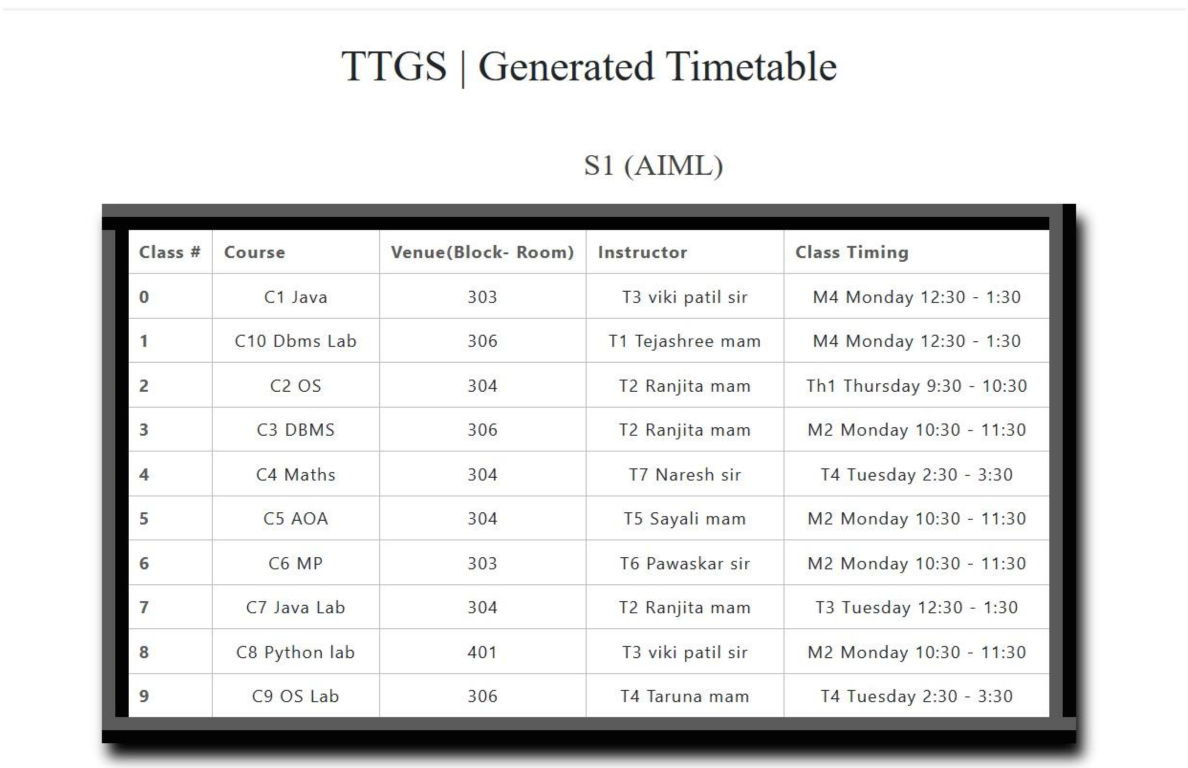
Add Sections

Please use the following form to enter the Sections into the system for timetable generation

Section ID:

Corresponding Department: ---------

Classes Per Week: 0



TTGS | Generated Timetable

S1 (AIML)

Class #	Course	Venue(Block- Room)	Instructor	Class Timing
0	C1 Java	303	T3 viki patil sir	M4 Monday 12:30 - 1:30
1	C10 Dbms Lab	306	T1 Tejashree mam	M4 Monday 12:30 - 1:30
2	C2 OS	304	T2 Ranjita mam	Th1 Thursday 9:30 - 10:30
3	C3 DBMS	306	T2 Ranjita mam	M2 Monday 10:30 - 11:30
4	C4 Maths	304	T7 Naresh sir	T4 Tuesday 2:30 - 3:30
5	C5 AOA	304	T5 Sayali mam	M2 Monday 10:30 - 11:30
6	C6 MP	303	T6 Pawaskar sir	M2 Monday 10:30 - 11:30
7	C7 Java Lab	304	T2 Ranjita mam	T3 Tuesday 12:30 - 1:30
8	C8 Python lab	401	T3 viki patil sir	M2 Monday 10:30 - 11:30
9	C9 OS Lab	306	T4 Taruna mam	T4 Tuesday 2:30 - 3:30

CHAPTER 6

Conclusion

6. Conclusion

Automatic Timetable Generator is a web based application for generating timetable automatically. It is a great difficult task that to manage many Faculty's and allocating subjects for them at a time manually. So proposed system will help to overcome this disadvantage. Thus we can generate timetable for any number of courses and multiple semesters. This system will help to create dynamic pages so that for implementing such an system we can make use of the different tools are widely applicable and free to use also.

Future suggestions -

The Automatic Timetable Generator is driven portal for educational organization and is a web based application which will be helpful for creating Timetabels . This project will be a great helpful for the institiutions because, It is agreat difficult task that to manage many Faculty's and allocating subjects for themat a time manually and this project will help to manage it properly. This manage timetable for faculty with considering maximum and minimum workload , and can be managed easily.

References

Research paper -

[1] Khyrina Airin Fariza Abu Samah, Siti Qamalia Thusree, Ahmad Firdaus Ahmad Fadzil, Lala Septem Riza, Shafaf Ibrahim, Noraini Hasan, “A Greedy-based Algorithm in Optimizing Student’s Recommended Timetable Generator with Semester Planner”, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 13, No. 1, 2022.

[2] Ghrushneshwari Dattatray Mhaise, Chetana Sudhakar Kurhade, Mayuri Yadav Devre, Payal Sonawane, Tambe Balasaheb Parasram, “AUTOMATIC TIME TABLE GENERATOR”, International Research Journal of Modernization in Engineering Technology and Science, 04/April-2023.

[3] S.P.Ramesh, Anuj Gangwar, Ketan Singh, Anookarsh Ayan, “Automatic Timetable Generator”,Annals of R.S.C.B., ISSN:1583-6258, Vol.25, Issue 4, 2021.

[4] Abulaziz Aminu, wahyu Caesarendra, Umar S Haruna, Abubakar Sani, Mansur Sa’id, Daniel SP amungkas, Sumantri R Kurniawan, Endang Kurniawan, “Design and Implementation of An Automatic Examination Timetable Generation and Invigilation Scheduling System Using Genetic Algorithm”, 2019 IEEE .

[5] Mrunmayee V. Rane, Vikram M. Apte, Vishakha N. Nerkar, Mani Roja Edinburgh, K.Y. Rajput, “Automated Timetabling System for University Course”, 2021 IEEE.