

TIMETABLE GENERATOR SOFTWARE USING PHP

Rahul Singh¹, Dr. Santosh Dwivedi², Mr. Shadab Ali³

¹UG student of Department of Computer Application, Shri Ramswaroop Memorial College of Management Lucknow, Uttar Pradesh, India.

²Professor, Department of Computer Application, Shri Ramswaroop Memorial College of Management Lucknow, Uttar Pradesh, India.

³Assistant professor, Department of Computer Application, Shri Ramswaroop Memorial College of Management Lucknow, Uttar Pradesh, India.

ABSTRACT

This research paper presents the design and implementation of an automatic timetable generator website software using HTML, CSS, JavaScript, and PHP. The creation of a timetable is a difficult undertaking that entails efficiently organizing multiple activities or events within predefined restrictions. By offering a user-friendly interface for producing schedules, the suggested program intends to automate this process. The backbone of the system is HTML, CSS is used for style, JavaScript is used for interactivity, and PHP is used for server-side functionality. The program includes algorithms that optimize the timetable-generating process and guarantee that user-defined limitations are satisfied. The results show that the program is useful and usable in producing accurate and efficient schedules for a variety of areas, including educational institutions, corporate organizations, and event management.

1. INTRODUCTION

Making a timetable is a very time-consuming and hectic task especially when you have to consider numerous activities, constraints, and preferences.

Efficiently doing these tasks required a lot of careful consideration and optimization. To resolve these problems, this research paper presents the design and implementation of an automatic timetable generator using HTML, CSS, JavaScript, and PHP.

The goal of this research paper is to develop a web-based software solution that automates the process of creating timetables and provides users with a user-friendly interface. By using HTML,

CSS, JavaScript for the frontend interface, and PHP for the backend functionality the proposed system provides an optimized solution for creating an efficient and accurate timetable that follows all the constraints given by the developer as well as the user.

The software includes intelligent algorithms to consider the factors such as subject duration, availability of faculty, and user-defined constraints such as the maximum number of lectures/labs in a week to create efficient timetables. These algorithms optimize the process of making the timetable and ensure all the activities are divided into the most appropriate time slots while meeting the specified constraints.

The effectiveness and usability of the timetable generator are taken out through thorough testing and analysis. The features like correctness of the timetable, its efficiency, and customer satisfaction will be taken into consideration to validate the system's capabilities in generating accurate and efficient timetables for educational institutions. This research paper aims to reduce the load of manually creating a timetable for an educational institution, reduce human errors and improve the overall efficiency of the system's timetable schedules. The software has the potential to rationalize the scheduling process, save valuable time, and ensure optimal usage of all the resources. The following sections of this research paper discuss the methodology, implementation, system evaluation, and results obtained from testing the automatic timetable generator. In addition, the implications, limitations, and future improvements of the system are discussed. In short, the timetable generator software, built using HTML, CSS, JavaScript, and PHP, provides a promising solution to the complexity of timetable generation. With the combination of intelligent algorithms and a user-friendly interface, the system aims to improve scheduling efficiency and accuracy, benefiting the educational institutions.

2. PROBLEM STATEMENT

Setting timetables may be regarded as an instance of limited satisfaction with multiple variables and loose parameters. These difficulties should be represented in a way that the scheduling algorithm can manage. Planning entails setting several logical boundaries on where activities may be completed.

simultaneously. For example, in an attempt to organize lessons in a tertiary institution, two courses taught by the same faculty may not be scheduled at the same time. Similarly, the two disciplines required by the same set of pupils should not be conflicting.

3. EXISTING SYSTEM

The making of the timetable is time-consuming and hectic. Currently, it is made with the help of human labor as there are no automated systems available. While generating the timetable the main concern is the collision between the slots. Distributing the time is a very tedious task and that makes the generation of entire timetable does not result well when done manually by a person. Therefore, even the existing software systems do not satisfy the rules associated with the creation of timetables. So, the current system is a time-consuming and hectic process that requires human labor and results in very little flexibility. To solve this issue the new system takes the approach towards an algorithm-based application that provides the time distribution between different activities (theory and labs) in such a way that there is no clash between the activities or collision between the timeslots. In the improved version of the existing system, the subjects will be given importance according to the number of lectures in a week. This has given us the hint of assigning the subject a certain number in a week as well as it will help to assign the subjects appropriately. In addition, the matching labs are also considered and assigned according to the dates so that the curriculum runs smoothly without any collision. This helps to satisfy the limit of the number of courses per day for both theory and labs. The proposed system is based on an analytical algorithm that manages obstacles and schedules individual resources. The software generates different custom versions of a timetable that includes normal timetables, faculty timetables, lab-only timetables, etc.

4. PROPOSED SYSTEM

The home page contains an admin login and a faculty login. The admin can log in by entering their username and password and the faculty can log in by using the assigned faculty number that is unique for every faculty.

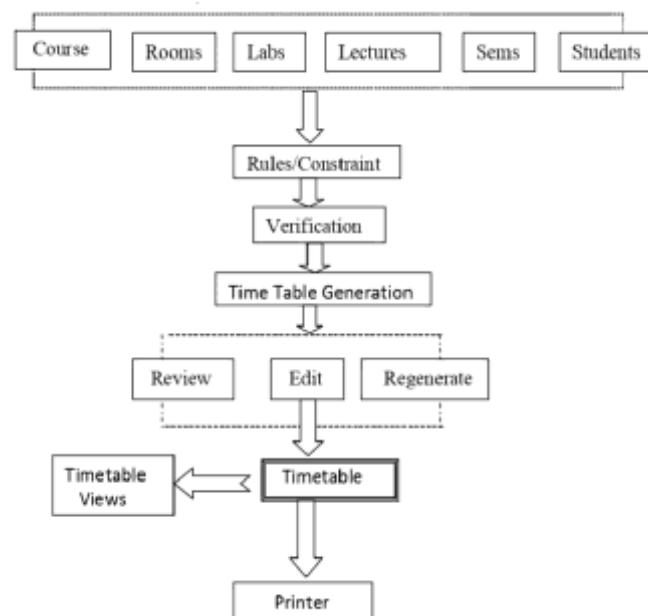
Admin after logging in can do the following tasks:

- Add teachers by providing faculty id, faculty name, alias name, assigned position, department, contact number, and email of the teacher.
- Add subjects by providing subject name, subject id, subject type, semester, and department.
- Add classrooms by providing the room number.
- Allot subject: here the admin can allot rooms to the courses, and allot theory and labs to the teachers.
- Generate timetable: the admin can generate the timetable and save it for students and faculty to see.

On the other hand, the faculty can see the semester timetable and timetable for only a single teacher too and the student doesn't have to log in, they can simply see and download the timetable from the homepage.

5. SYSTEM ARCHITECTURE

To explore how scheduling works, you can view data flow diagrams that show how the system works and what is implemented in the system. Shows the elements of the system. It also shows the interactions between these elements. Models determine the composition and constraints of these models. When faced with a complex problem, the system divides it into easy-to-solve parts with simple solutions. By combining all these small solutions, we can find solutions to complex problems. A flowchart is a diagram representing an algorithm, workflow, or process, showing the steps as boxes of various types and connecting them with arrows to indicate their sequence. This schematic shows the problem-solving model for the system.



6. FUTURE SCOPE

The immediate software is only bound to generate timetables for a college department only but the future scope of this software is endless. There are many places where there is a need for a timetable like a business. Thus, the timetable generator for an institution can be modified in the future as a prototype for generating timetables for other occasions. Also, more features will be planned to be added like sending emails to the faculty reminding them about their current lecture, filing a leave application so that software can generate a substitute, and adding custom constraints only applicable to a particular institution or department. In summary, the potential of this project is very high and can be modified or improved in a lot of ways.

7. CONCLUSION

It may be a troublesome assignment to oversee numerous Workforce and relegate subjects on time physically. Subsequently, our proposed system will help overcome this irregularity. Subsequently, we can produce a timeline for any number of courses and semesters. This program will assist you to form adaptable pages so that in utilizing such a program we can utilize a variety of more effective devices and be more liberate to utilize. Different timetables for each course, sort, and lab are naturally produced by this framework. A combination of distinctive spaces can be gotten to from another timeline as required. The venture minimizes time utilization and you're feeling the torment of introducing a timeline. The extension is created in such a way that no space clashes happen giving the timetable highlights as you would like. Future enhancements that can be made to the extent make it a great time to arrange for offices and the rest of the college. This enhancement can be accomplished by making extra changes by keeping the strategy and procedures utilized for this venture.

8. REFERENCES

- [1] Bhaduri is a “university timetable scheduling using genetic algorithm”. Advances in Recent Technologies in Communication and Computing, 2009. ART Com '09. International Conference.
- [2] Dipti Srinivasan “Automated timetable generation using multiple context reasoning for university modules” Published in Evolutionary computation, 2002. ceca '02. proceedings of the 2002 congress on (volume:2).
- [3] Anirudha Nanda “An Algorithm to Automatically Generate Schedule for School Lectures Using a Heuristic Approach”. International Journal of M achine Learning and Computing, Vol. 2, No. 4, August 2012.
- [4] Carter M. W., Laborite G., “Recent developments in practical course timetabling”, Lecture Notes in Computer Science, Vol. LNCS1408, Springer-Verlag, 1998, pp. 3-19.
- [5] Schaeff, A., “A survey of automated timetabling”, Artificial Intelligence Review, No. 13, 1999, pp. 87-127.
- [6] Yu, T. L., “Time-table scheduling using neural network algorithms”, IJCNN International Joint Conference on Neural Networks, 1990, Page(s)