e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

AUTOMATIC TIME TABLE GENERATOR

Manthu Kavya[*1], Maram Lasya[*2], Pabbisetty Venkata Baby Suraksha[*3], Muthuvairavan Pillai[*4]

[*1,2,3]UG Student, Dept Of CSBS, R.M.D Engineering College, Thiruvallur, India.

[*4]Associate Professor, Dept. Of CSBS, R.M.D Engineering College, Thiruvallur, India.

ABSTRACT

Preparing a schedule is not an easy task and takes considerable time and effort. Timetables have a number of uses including: the coordination of class hours in schools and colleges; the adjustment of the time of railway and bus work; and so on. Classically, drawing up a schedule is a time consuming and labour intensive task. The purpose of this paper is to examine if Genetic Algorithms can cut the length of this process and enhance the performance with reduced human input.

It speeds up and increases the reliability of the timetable generation for sure with the help of a Genetic Algorithm. The scheduling process within this approach involves two phases; in the first phase, all the core classes of the institute will be scheduled and this is done at a centralized level. The next phase on the classes that are offered per department only. This stage of the timetabling process entails drawing up daily and weekly work schedules for the departments' activities which is mainly done by manual effort. Due to the importance of creating a workable timetable, previous year's timetables are usually borrowed and recycled with the aim of making one shortest time schedule possible. Such a manual process is susceptible to human error and awkwardness where RGA comes in beneficial as it solves these problems by producing an accurate and optimum timetable.

Keywords: Genetic Algorithm (GA), Timetable Generation, Optimization Techniques, Scheduling Conflicts, Resource Utilization, Workload Balancing.

I. INTRODUCTION

The Automatic Timetable Generator is a software tool developed in Java for the purpose of generating timetables without any human assistance. Normally, the preparation and management of timetables is done manually and this process is very slow and ineffective. This software makes it easier by defining the periods of class automatically thereby facilitating access to the schedules of the lecturers on their mobile phones. It also controls the timetable in case a certain teacher does not show up, comes in late or leaves early.

To achieve better control of workload among employees, the system stipulates and allocates the maximum and the minimum number of teaching hours for each of the faculty members on a daily, weekly and monthly basis so that even distribution is achieved. Faculty members can also request for leave through the application, indicating the period, the reason and a substitute teacher if any. Such selection of a substitute may be done only in case the requested period fully allows the said substitute, to fit into the schedule of the system. Faculty members' requests for substitution may be accepted or denied by the substitute and all the requests and approvals of leaves are at the disposition of the principal who is able to control and complete them.

II. EXISTING SYSTEM

The preparation of manual timetable in colleges is most of the time redundant and unpleasantly involving, therefore often resulting in class timetable problems such as conflicting classes for the same instructor or for the same classroom occupied by several lessons. With numerous courses which have multiple subjects and relatively few lecturers who are also teachers of many subjects, manual planning creates difficulty in the optimization of the resources available. In line with the above concerns, a software-based, computer integrated timetable generator can be designed. The system would consider several dimensions such as the number of subjects, teachers, hours allowed per subject per teacher, priorities among subjects, and directives on number of weeks to complete the coverage of the subjects. The system would then proceed to create appropriate timetables for every work day each designed to effectively allocate resources and avoid clashes in schedules. From all the optimal solutions the system will rank the available schedules and propose the best one that will distribute the work fairly among all individuals while maintaining order in the timetable.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

III. APPROACH AND PROPOSED METHODOLOGY

The Automatic Timetable is a Java-based software designed to streamline and automate the timetable creation process. This tool allows for efficient period management and provides faculty members with instant access to their schedules through notifications on their phones. The system is adaptable, updating timetables in real-time to accommodate teacher absences, late arrivals, or early departures, saving significant time and effort. With this software, faculty no longer need to manage detailed period schedules or monitor workload limits manually. Teachers can also request leave by specifying dates, reasons, and a substitute faculty member. When selecting a substitute, the system checks the availability of the chosen faculty to ensure they are free during the required period. Substitute faculty can approve or decline these requests, and the principal has full oversight, with the ability to view all leave applications and substitute responses to approve or reject requests as needed. Overall, this comprehensive timetable management solution overcomes the challenges of traditional scheduling methods in colleges, ensuring smooth, automated scheduling and reducing administrative burdens.

FEASIBILITY ANALYSIS:

Feasibility studies are used to evaluate the likelihood of success in pursuing a given course of action. More precisely, it reflects on the practicality and the difficulties in implementing a design task. Generally conducted prior to any design work or project execution, the study also plays a pivotal role in identifying the extent of possible constraints, and if at all, aspiration could be achieved.

ECONOMIC FEASIBILITY:

Building the envisaged system does not need any extra structures or specialized apparatus as all the dependencies are satisfied by utilizing open-source applications. The instruments utilized are all freemium and under open-source provision, while the system employs JSP for the development, hence the overall development is economically advantageous.

TECHNICAL FEASIBILITY:

The planned approach is achievable from a technical point of view, since only ordinary hardware and software tools are required, which are currently within the system. It will only require the installation of JSP and the database MySQL which are both free for download. In addition, the system is also built to be growing, in a way where new modules can be added when deemed necessary in the coming future. The application system will also provide forms and interfaces that are easy to use, such that all the users will find it very easy to use and access all the information.

BEHAVIORAL FEASIBILITY:

Behavioral feasibility is concerned with estimating the resources required for the introduction of the proposed information system, with emphasis on its users and their learning curve. Since the system has a simple and easy to use graphical user interface, there is very little training involved whereby users can easily handle and use the application without any guidance.

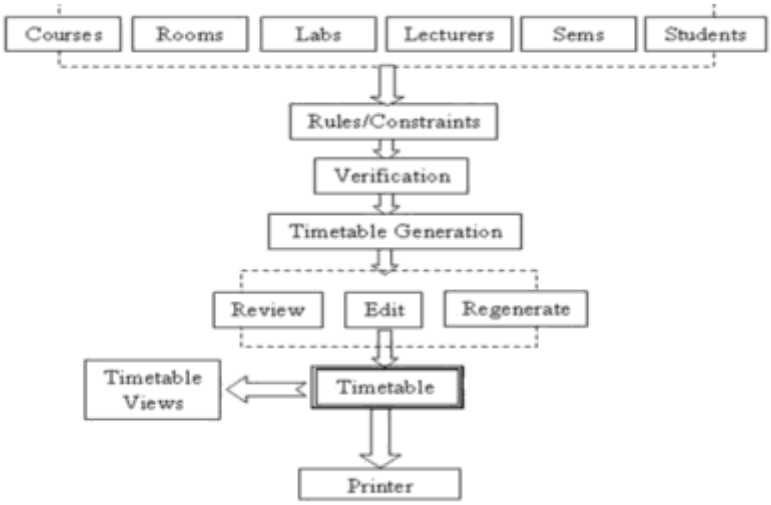

Fig 1: Block diagram of time table generation

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

GENETIC ALGORITHM:

Computational problems, especially those that need decent coverage of large solution spaces, employ metaheuristic techniques such as genetic algorithms. Such algorithms are very flexible and therefore effective for dynamic scenarios. For instance, in the case of timetabling applications, self-adaptive techniques contribute to the improvement of overall solution quality due to changing circumstances. The Fundamental Concepts behind Genetic Algorithms Genetic Algorithms or GAs is a classification of software techniques used in programming. It is considered as a form of evolutionary algorithms which is a subset of artificial intelligence techniques. This way of solving problems was researched by Professor John Holland in the 1960s, however, the publication 'Adaptation in Natural and Artificial Systems' in the 1970s released by the author raised widespread interest towards the research of GA.

This is partially based on the principles of evolution and structure of genetic algorithms development. Evolutionary gas proposes the existence of such organisms which reproduce rapidly. They do not have unlimited food or space, thus there is a struggle for existence and the best adapted types prevail and become more numerous. Find Thin Air Selection The selection scheme that selects mating individuals using a rotative 'roulette wheel' is a popular method of assigning any individual to the mating pool based on a probability. This method even helps improve the chances of selecting strong candidates by selecting the best individuals with higher fitness. This technique would also work, for example, if there were five individuals training for a certain fitness this method would probabilistically select individuals according to how fit they are so that the desirable traits are picked for the next generation.

IV. RESULTS AND DISCUSSION

There are many benefits of integrating a genetic algorithm when developing a timetable generation system. Considering that a genetic algorithm is a suitable optimization technique, it works by producing better and better schedules through evolutions of the candidate solutions. It is also designed to accommodate numerous constraints and objectives such as avoiding conflicts in scheduling, utilization of resources, and balancing the work for teachers. Regardless of these merits, there are certain limitations that can be encountered in the system. The overreliance on quality and sufficient input data is one such limitation. Random errors or missing data usually result in very low quality solutions and recommendations. Moreover, the number of potential solutions and the number of such solutions that need to be examined can become very large and very complex and may require large amounts of computational power, as especially the more complex the scheduling and the more variables it has.

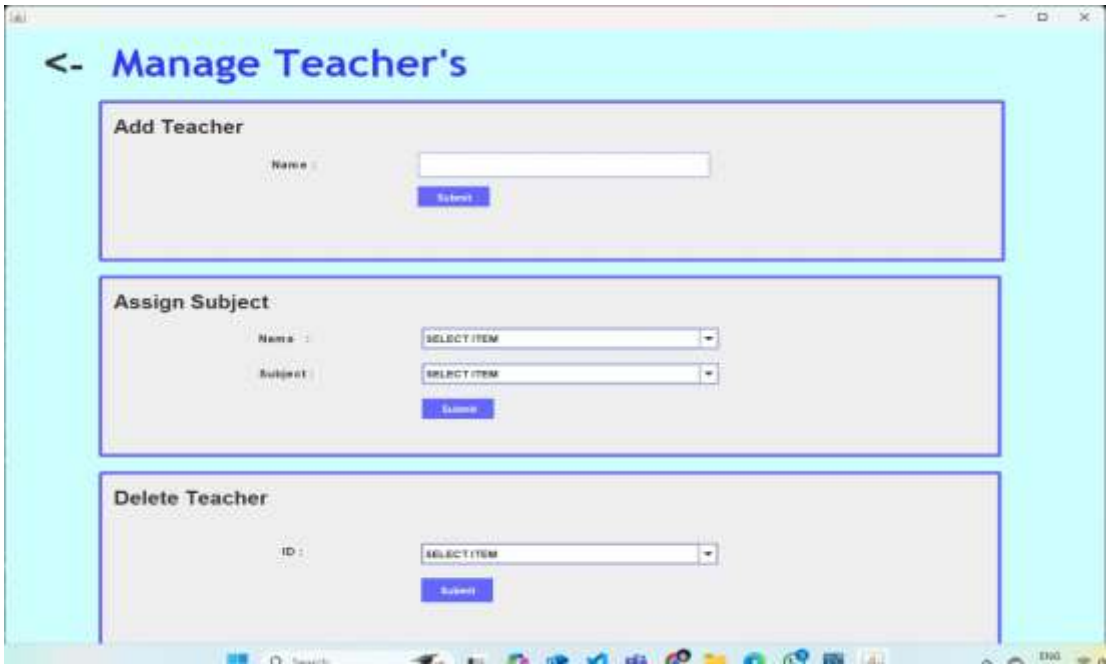

Fig 1: Output

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

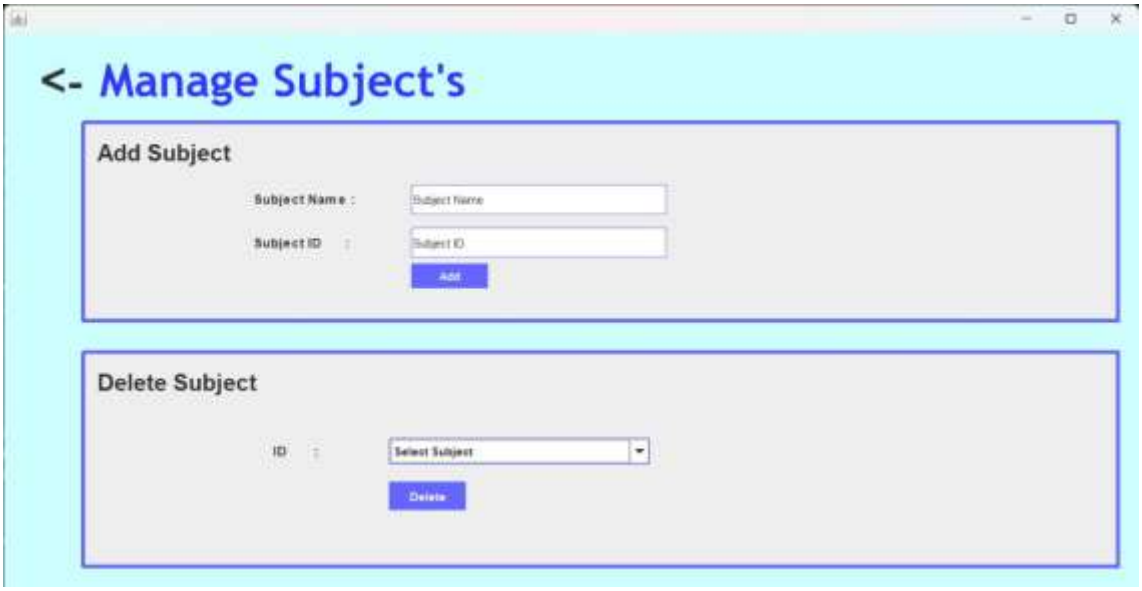

Fig 2: Output

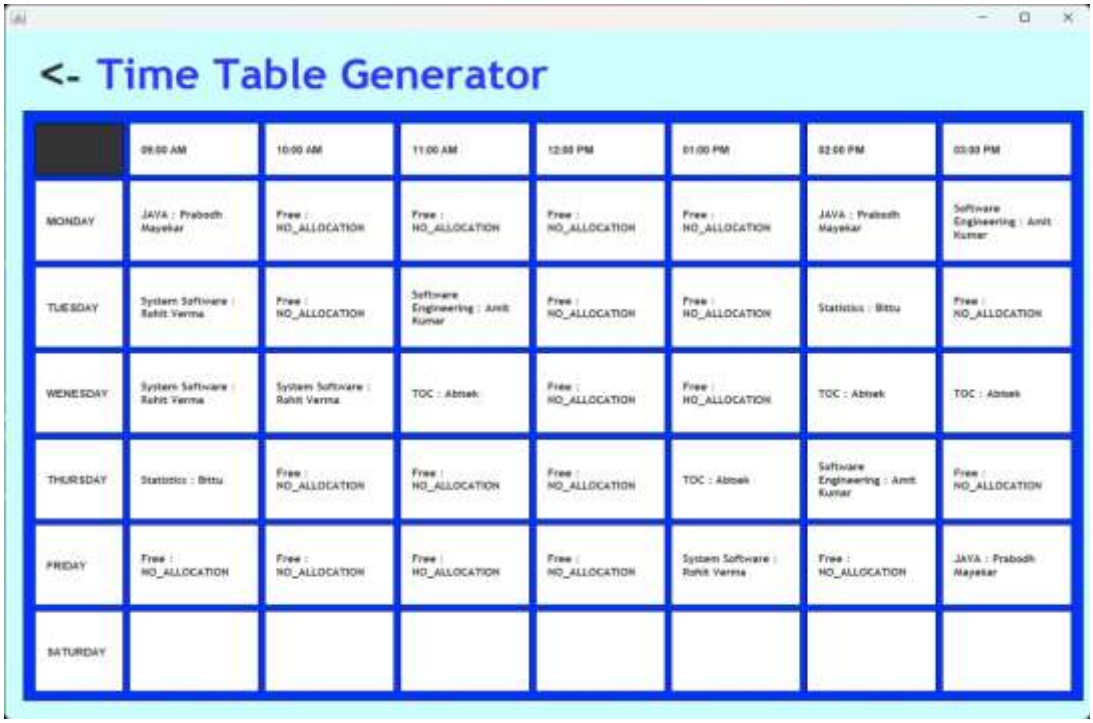

Fig 3: Output

V. CONCLUSION

It has been proven that the use of genetic algorithms (GA) offers a viable and useful solution to the lecture timetabling problem. This aspect of the method demonstrates great promise for the future in helping to construct more equitable timetables for the students. The structure of the problem is robust and can be implemented for many other types of timetabling problems. For example, experimental results show that the positive outcome of the approach can be attributed mainly to the fact that a mutation operator is used. The GA-based timetabling framework has shown successful emulation of several practical applications, including scheduling of universities, and therefore can be expected to work for other problems of comparable complexity and size. There are no reasons to assume that the problem on which it was tested is easier than other real-world scheduling problems. Nonetheless, a lot of effort is still needed, especially to examine the level at which

this approach, if any, goes up in performance when tackling larger or varying timetabling problems. Also the ability of the algorithm to work efficiently in other situations and with more datasets should also be reviewed.

VI. REFERENCES

[1] M. Doulaty, M. R. Feizi Derakhshi, and M. Abdi, "Timetabling: A State-of-the-Art Evolutionary Approach", International Journal of Machine Learning and Computing,Vol3, No3, June 2013.

[2] Anirudha Nanda, Manisha P. Pai, and AbhijeetGole, "An Algorithm to Automatically Generate Schedule for School Lectures Using a Heuristic Approach", International Journal of Machine Learning and Computing, Vol. 2, No. 4, August 2012.

[3] Alberto Colorni, Marco Dorigo, Vittoria Maniezzo, A GENETIC ALGORITHM TO SOLVE THE TIMETABLE PROBLEM, Centre for Emergent Computing , Napier Univer sity , Edinburgh EH10N 5DT, UK 2000.

[4] J.J Grefenstette, editor. Proceedings of the First Interna tional Conference on Genetic Algorithms and their Applications. Practice and Theory of Automated Timetabling VI Pro ceedings of The 6th International Conference on the practice and Theory of Automa.