Automatic Faculty Time Table Generation

A. Udaya Sri[1], A. Devisri[2], M. Nageswara Rao[3]
[1,2] *Student, CSE Dept, Sri Vasavi Engineering College, Tadepalligudem, A.P., India*
[3]*Sr. Asst Prof., CSE Dept, Sri Vasavi Engineering College, Tadepalligudem. , A.P., India*
Correspondent Mail Id: ankadiudayasri@gmail.com

ABSTRACT
In contemporary educational institutions, the creation of timetables has become an increasingly complex and time-consuming process. Our project simplifies this by considering key inputs, such as subjects, faculty workloads, and semester details, to create conflict-free weekly schedules. The system adheres to institutional constraints, such as faculty assignments and subjects, while managing multiple sections to ensure balanced schedules without overlap. Designed to improve operational efficiency, the project enables administrators to generate and access timetables easily, promote transparency, and simplify schedule management.

Keywords: Python, Soft and hard Constraints, Automated timetable, Generic Algorithm.

I. INTRODUCTION
Efficient management of academic schedules is a critical component of any educational institution's operations, as it ensures that classes are well organized, faculty resources are used optimally, and students can pursue their education with minimal disruptions. However, the process of creating faculty timetables is traditionally a complex and labour- intensive task. It involves coordinating numerous factors such as course requirements, teacher availability, classroom space, and subject frequency. Academic administrators must often balance these variables manually, and any oversight can result in significant scheduling challenges.

The complexity of timetable creation stems from the need to align multiple, and sometimes conflicting, factors to produce a workable schedule that satisfies both institutional goals and individual preferences. For example, ensuring that a subject is taught by a specific faculty member at a time when both the classroom and students are available requires careful planning. Additionally, the frequency of classes for each subject must adhere to the curriculum, and the teacher's workload must be evenly distributed to prevent overburdening. In larger institutions, this process becomes even more complicated as the number of teachers, courses, and students increases, making it difficult to manually develop accurate and conflict-free timetables.

As educational institutions expand in size, both in terms of faculty members and course offerings, the manual scheduling of timetables is not only time-consuming, but also prone to errors.

With the increasing number of variables to consider, such as faculty assignments, classroom availability, and curriculum requirements, the scheduling teachers for multiple classes at the same time, or overlapping sessions for students. There errors disrupt the flow of academic activities, often requiring additional administrative interventions to resolve them, which further consumes valuable resources and time. Moreover, the manual approach to timetable creation is inherently labour intensive. Academic staff

responsible for scheduling often spend significant amounts of time reviewing course requirements, verifying faculty availability, and coordinating room assignments, which could otherwise be spent on strategic academic and administrative functions. As a result, the manual method can strain institutional resources, particularly in larger schools or universities, where the complexity and volume of scheduling tasks are greater.

To address these challenges, this project sought to automate the process of generating faculty timetables. By leveraging advanced algorithms and optimization techniques, the system automates the coordination of subject frequency, faculty availability, classroom allocation, and teacher workload. These algorithms consider all the necessary factors to create accurate and efficient timetables with minimal human intervention. Automation can significantly reduce scheduling conflicts, such as overlapping classes or double bookings, while ensuring that faculty workloads are evenly distributed and that course requirements are met.

Automation of the timetabling process not only minimizes the potential for errors, but also enhances the overall efficiency of academic administration. By streamlining this time-consuming task, institutions can allocate their administrative resources more effectively, freeing time for staff to focus on other critical aspects of academic management. This solution is also scalable, allowing it to adapt to the needs of institutions as they grow in size and complexity, ensuring that the timetabling process remains efficient, even in larger or rapidly expanding academic environments.

II. LITERATURE SURVEY

[1] The paper "Automatic Timetable Generator" by M.P. Navale et al. discusses a system designed to automate timetable creation, minimizing manual errors and time consumption. The proposed solution utilizes algorithms to allocate classes, resources, and schedules efficiently based on predefined constraints, Ti aims to streamline academic planning for institutions while ensuring conflict-free scheduling.

[2] The paper "Automatic College Timetable Generation" by S. Wankhede et al. presents a system for generating college timetables automatically, addressing the challenges of manual scheduling. The system uses algorithms to manage time slots, faculty availability, and course requirements efficiently.
It aims to reduce errors and optimize resource utilization in academic scheduling.

[3] The paper "Automatic Timetable Generation using Genetic Algorithm" by D. Mittal et al. explores a system that employs genetic algorithms to automate the creation of academic timetables. It focuses on optimizing resource allocation while adhering to constraints like faculty availability and room capacity. The approach aims to produce efficient, conflict-free schedules.

[4] The paper "A Novel Approach for Automatic Timetable Generation" by M. R. Bagul et al. proposes a system to automate timetable creation using advanced algorithms. The approach ensures optimal utilization of resources while meeting constraints like faculty availability and course requirements. It aims to simplify scheduling and reduce manual effort.

[5] The paper "Automatic Timetable Generator" by S. Ambhore et al. introduces a system for generating academic timetables automatically to reduce manual workload and errors. It leverages

algorithms to allocate time slots, faculty, and resources efficiently while adhering to predefined constraints. The system aims to streamline the scheduling process for educational institutions.

[6] The paper "Automated Timetable Generation" by P. Tanksali et al. presents a system designed to automate the creation of academic timetables. It uses advanced algorithms to ensure efficient allocation of resources while satisfying constraints such as faculty availability and classroom capacity. The system aims to simplify scheduling and improve accuracy in educational planning.

[7] The paper "Automated Timetable Generator" by S.S. Ansari et al. discusses a system to automate timetable generation, addressing the challenges of manual scheduling. It uses algorithms to allocate time slots, faculty, and resources effectively while adhering to predefined constraints. The system aims to reduce errors and enhance scheduling efficiency for educational institutions.

METHODOLOGY

It's a systematic a approach or framework that guides how research, analysis, or a specific process is carried out. It encompasses the methods, rules, and procedures that are used to gather, analyze, and interpret data. Methodology is essential for ensuring the reliability, validity, and consistency of results, especially in fields such as science, social science, business, and engineering.

Inputs:

1. **Institutional Constraints**: Rules and policies related to scheduling.
2. **Subjects Data**: Information about courses, including requirements and schedules.
3. **Faculty Workloads**: Availability and workload data of faculty members.
4. **Semester Details**: Information about the semester, such as dates and holidays.

Steps:

1. **Data Input**: Gather all relevant data from the inputs.
2. **Data Validation**: Check that the input data is accurate and complete.
3. **Constraint Checking**: Ensure the data follows institutional rules and policies.

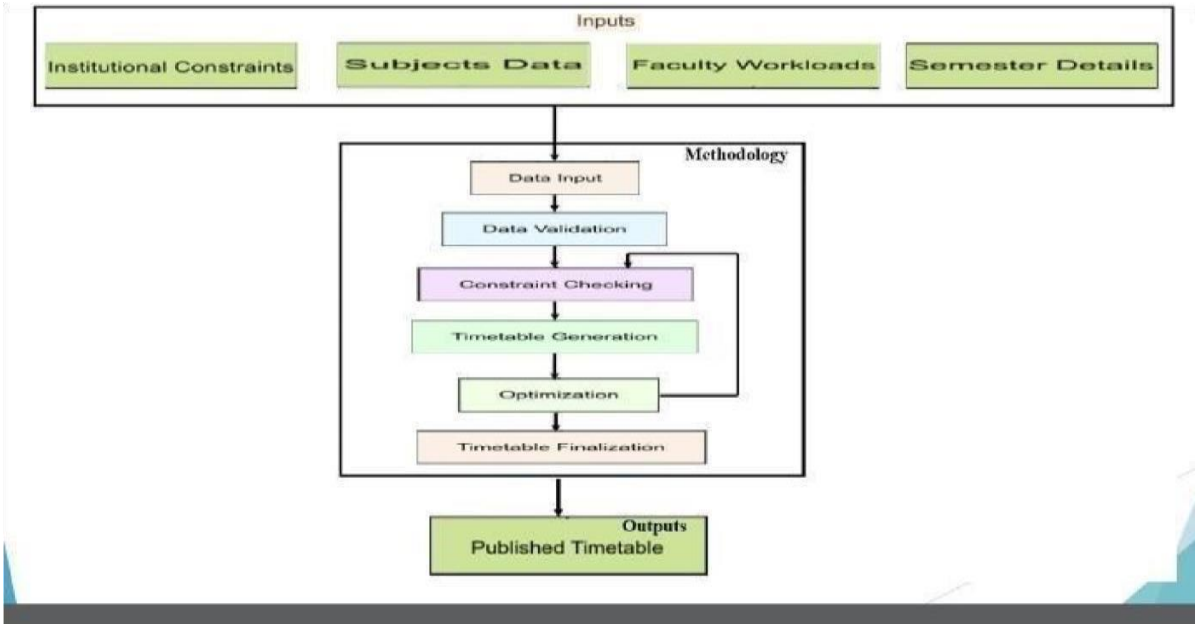


Fig1: METHODOLOGY

4. **Timetable Generation**: Create initial drafts of the timetable based on the data.

5. **Optimization**: Improve the timetable for efficiency and conflict resolution.

6. **Timetable Finalization**: Prepare the final timetable for publication.

Output:

- **Published Timetable**: The final schedule ready for distribution to faculty and students. .This simple

format outlines the methodology for automating the faculty timetabling process efficiently.

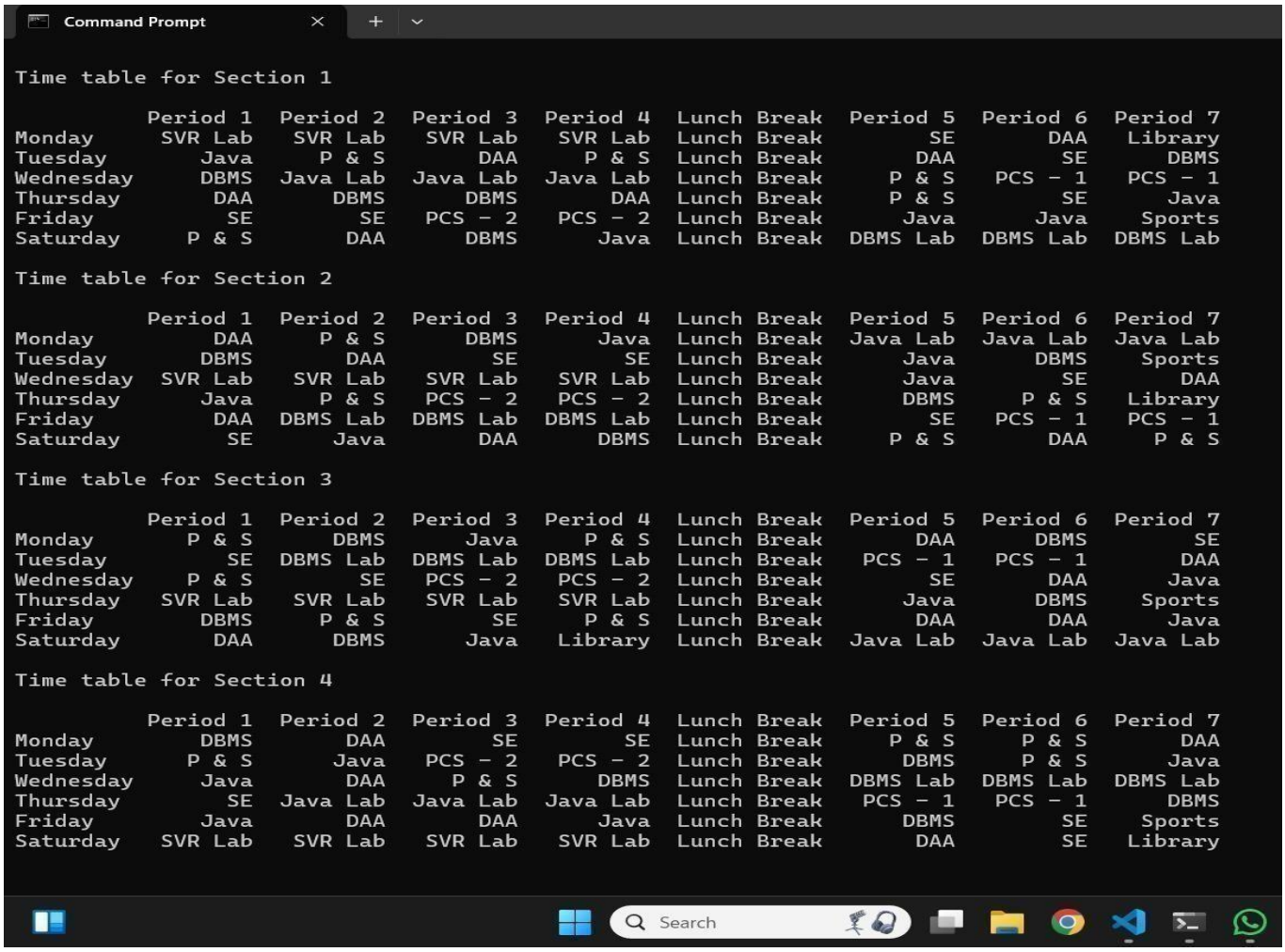
The timetable generation process begins with gathering critical input data (such as institutional constraints, subject data, faculty workloads, and semester details). This data is then validated, checked against constraints, and used to generate an initial timetable. The system optimizes this timetable and finalizes it for publication, ensuring an effective schedule for both students and faculty.

III. EXPERIMENTAL RESULTS

In this result chapter we can see how the system has developed the timetable for four sessions.

There are many constrains which are involved in this.

- One faculty member shouldn't take classes for two sessions at a time.



Command Prompt

Time table for Section 1

	Period 1	Period 2	Period 3	Period 4	Lunch Break	Period 5	Period 6	Period 7
Monday	SVR Lab	SVR Lab	SVR Lab	SVR Lab	Lunch Break	SE	DAA	Library
Tuesday	Java	P & S	DAA	P & S	Lunch Break	DAA	SE	DBMS
Wednesday	DBMS	Java Lab	Java Lab	Java Lab	Lunch Break	P & S	PCS - 1	PCS - 1
Thursday	DAA	DBMS	DBMS	DAA	Lunch Break	P & S	SE	Java
Friday	SE	SE	PCS - 2	PCS - 2	Lunch Break	Java	Java	Sports
Saturday	P & S	DAA	DBMS	Java	Lunch Break	DBMS Lab	DBMS Lab	DBMS Lab

Time table for Section 2

	Period 1	Period 2	Period 3	Period 4	Lunch Break	Period 5	Period 6	Period 7
Monday	DAA	P & S	DBMS	Java	Lunch Break	Java Lab	Java Lab	Java Lab
Tuesday	DBMS	DAA	SE	SE	Lunch Break	Java	DBMS	Sports
Wednesday	SVR Lab	SVR Lab	SVR Lab	SVR Lab	Lunch Break	Java	SE	DAA
Thursday	Java	P & S	PCS - 2	PCS - 2	Lunch Break	DBMS	P & S	Library
Friday	DAA	DBMS Lab	DBMS Lab	DBMS Lab	Lunch Break	SE	PCS - 1	PCS - 1
Saturday	SE	Java	DAA	DBMS	Lunch Break	P & S	DAA	P & S

Time table for Section 3

	Period 1	Period 2	Period 3	Period 4	Lunch Break	Period 5	Period 6	Period 7
Monday	P & S	DBMS	Java	P & S	Lunch Break	DAA	DBMS	SE
Tuesday	SE	DBMS Lab	DBMS Lab	DBMS Lab	Lunch Break	PCS - 1	PCS - 1	DAA
Wednesday	P & S	SE	PCS - 2	PCS - 2	Lunch Break	SE	DAA	Java
Thursday	SVR Lab	SVR Lab	SVR Lab	SVR Lab	Lunch Break	Java	DBMS	Sports
Friday	DBMS	P & S	SE	P & S	Lunch Break	DAA	DAA	Java
Saturday	DAA	DBMS	Java	Library	Lunch Break	Java Lab	Java Lab	Java Lab

Time table for Section 4

	Period 1	Period 2	Period 3	Period 4	Lunch Break	Period 5	Period 6	Period 7
Monday	DBMS	DAA	SE	SE	Lunch Break	P & S	P & S	DAA
Tuesday	P & S	Java	PCS - 2	PCS - 2	Lunch Break	DBMS	P & S	Java
Wednesday	Java	DAA	P & S	DBMS	Lunch Break	DBMS Lab	DBMS Lab	DBMS Lab
Thursday	SE	Java Lab	Java Lab	Java Lab	Lunch Break	PCS - 1	PCS - 1	DBMS
Friday	Java	DAA	DAA	Java	Lunch Break	DBMS	SE	Sports
Saturday	SVR Lab	SVR Lab	SVR Lab	SVR Lab	Lunch Break	DAA	SE	Library

Search

- One subject shouldn't be conducted more than two times per day.
- In a week, we can conduct only one subject two times simultaneously.
- If there is a lab of that particular subject for one session, there shouldn't conduct class for another session at the same time.
- Sports has to be in only seventh period and may library period be any one of 4^{th} period and may library period be any one of 4^{th} period or 7^{th} period.
- The PCS subject class has to be conducted in the week day of lab. The lab subject can be anyone of choosen subject for that particular subject.

Fig2: Experiment Results

Detailed Timetable for Section 1

Section 1 students have a balanced mix of theory classes and lab sessions throughout the week. The SVR Lab dominates one of the days with four consecutive periods, providing significant hands-on experience in system verification and validation.

Throughout the week, there are several classes dedicated to Software Engineering (SE) and Design and Analysis of Algorithm (DAA), focusing on crore theoretical knowledge. The DBMS Lab is spread across

multiple days, allowing students to work on their database management projects. Java Lab sessions are also frequently scheduled, offering practical coding experience.

In addition to academics, students have scheduled periods for Sports, allowing them time for physical activity, and regular time in the Library, which likely serves as a self-study or research period.

Overall, the schedule offers a good mix of theoretical learning in subjects like P&S (Probability & Statistics), PCS (Professional Communication Skills), and DBMS, combined with practical applications in the labs.

Detailed Timetable for Section 2

Section 2 follows a similar pattern, with a strong emphasis on both theoretical and practical learning. DAA and P&S are two of the main theoretical subjects that students engage with multiple times during the week.

The Java Lab takes up a significant part of one day, offering continuous practical sessions that are designed to improve students' programming skills. Additionally, the SVR Lab is scheduled for an entire morning, which is a hands-on session aimed at system verification and validation.

The week includes frequent theoretical sessions on SE, DBMS, and PCS, ensuring that students cover a board range of topics. After practical work in labs, students also have some library time for selfdirected study or project research.

The schedule is rounded out with time for physical activities, with a dedicated Sports session to promote health and well-being.

Detailed Timetable for Section 3

Section 3's timetable offers a blend of practical and theoretical education. One of the highlights of the week is the continuous session in the SVR Lab, allowing for in-depth practical work on system verification and testing projects. Similarly, multiple sessions in the Java Lab ensure students get ample time to practice and refine their programming skills.

There are several theoretical subjects taught throughout the week, including P&S, DBMS, SE, and DAA, all of which play a crucial role in building a strong academic foundation. PCS-1 and PCS2 are additional subjects that likely focus on computing skills and problem-solving approaches.

The week also includes scheduled time in the Library, offering students a chance to catch up on reading, conduct research, or work on group projects. Practical subjects like DBMS Lab ensure that students engage with databases in a hands-on manner, giving them real-world experience.

The inclusion of Sports ensures that students maintain a balance between their academic workload and physical fitness, promoting overall well-being.

Detailed Timetable for Section 4

The schedule for Section 4 provides a good balanced of theoretical and practical sessions. The week includes comprehensive lessons in subjects like DBMS, DAA,SE and P&S, ensuring that students are well-versed in both analytical and computing skills.

Practical experience is provided through multiple lab sessions, including significant time in the SVR Lab and Java Lab. Theses labs are essential for developing hands-on skills in system verification and coding, respectively.

The timetable also includes time in the DBMS Lab, allowing students to work on databases and their managements, which is an important aspect of their studies. Theoretical sessions are frequently mixed with practical ones, giving students a balanced learning experience.

The schedule is rounded out with a dedicated session for Sports, promoting physical fitness and mental refreshment. The Library sessions are also integrated into the week, likely serving as selfstudy periods for students to focus on research or group work.

IV.CONCLUSION

The automatic faculty timetable project represents a significant advancement in the field of academic scheduling, addressing the complexities and challenges associated with traditional and manual methods. By leveraging automated processes, this project not only enhances efficiency but also minimizes the likelihood of scheduling conflicts, ensuring that both faculty and students have a clear and organized timetable.

Throughout the development of this project, we have implemented a systematic methodology that incorporates essential inputs, such as institutional constraints, subject data, faculty workloads, and semester details. The architecture of the system effectively accommodates the needs of various stakeholders, including student, faculty, and administrators, while enabling seamless data management and schedule generation

REFERENCES

[1] M.P. Navale, A. Marne, O. Pasalkhar, A. Pawar, and S. Yadav, “Automatic Timetable Generator,” International Research Journal of Moderation in Engineering Technology and Science, vol. 5, no.
2, pp. 275-278, Feb.2023.

S. Wankhede, A. Sahare, M. Korde, N. Raut, and U. Ladke, “Automatic College Timetable Generation,” International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vl.5, no. 2, pp. 521-527, Mar-Apr. 2019.

[2] D. Mittal, H. Doshi, M. Sunasra, and R. Nagpure, “Automatic Timetable Generation using Genetic Algorithm,” International Journal of Advanced Research in Computer and Communication Engineering, vol. 4, no. 2, pp. 245-248, Feb. 2015.

[3] M.R. Bagul, S.C. Chaudhari, S. N. Nagare, P. R. Patil, and K. S. Kumavat, “A Novel Approach for Automatic Timetable Generation,” International Journal of Computer Applications, vol. 127, no.
10, pp. 26-30, Oct. 2015.

[4] S. Ambhore, P. Walke, R. Ghundgrudkar, A. Alone, and A. Khedkar, "Automatic Timetable Generator," International Journal of Research in Engineering, Science and Management, vol. #, no. 3, pp. 381-385, Mar.2020.

[5] P. Tanksali, I. Dhond, S. Pednekar, V. Singbal and S. Sivaraman, "Automated Timetable Generation," Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., vol. 6, no. 3, pp. 389-396, May- Jun. 2020.

[6] S. S. Ansari, A. Ahmed, A. Aamir, and A. Zaeem, "Automated Timetable Generator," International

Journal for Innovative Research in Science & Technology, vol. 1, no. 11, pp.118- 121, Apr. 2015.