Automatic Timetable Generator using Genetic Algorithm

Harsh Bhanushali
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
harsh050498@gmail.com

Greshma Sapra
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
greshmasapra.gs@gmail.com

Surbhi Saroliya
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
saroliyasurbhi@gmail.com

Rujata Chaudhari
Assistant Professor
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
rhchaudhari@apsit.edu.in

Nahid Shaikh
Assistant Professor
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
nashaikh@apsit.edu.in

***Abstract*— Time table generation is tedious job for specialist with reference to time and man power. Providing automatic generator can facilitate to get agenda mechanically. Our generated system can facilitate to generate timetable mechanically by reducing human efforts. Our system prevents the complexness of manually implementing the constrains in timetable generation. The algorithm used include a number of strategies aimed at boosting the search operation's operativity. Our system takes inputs like subjects, subject type, teachers, semester, class room, laboratories and load of subject. By counting on these inputs algorithm will run on certain input variables and finally system will generate potential time tables for operating days of the week for our college, taking into consideration all the basic constraints required while generating the timetable.**

I. INTRODUCTION

A college timetable is a temporal property of a group of lectures and practical's within which all required constraints need to be satisfied. Making such timetables manually is complicated and time overwhelming method. By automating this method with pc power-assisted timetable generator will save lot of precious time of co-ordinators concerned in making and managing course timetables. To generate such system our college sponsored our project which is built to efficiently generate college time mechanically, reducing the need of manual interference by timetable coordinator. Our system is designed in such a way that there are no slot collisions that provides optimality to the timetable, extra options that's enclosed within the system is that there's no headache of giving abundant input, only basic input like subject, semester, faculty, load, classroom number, lab number etc, are required in csv format.

Inputs once received by system will be processed further and the given set of constraints provided in proposed system will be applied on it, further it will be validated and the system will check whether all the requirements are fulfilled or not. Once, validated system will generate a review of the timetable which if approved will be stored in the database and will be available with different views of the timetable like class timetable, lab timetable, faculty timetable and master timetable which can be exported as csv file anytime.

Our system can be used by all the educational institutes as we have built it in a generic method, only concern is the input that the system takes. For this, we have provided option of downloading a sample input file which is editable and the same can be uploaded as input after making changes, to the system. Input file cannot be altered as there are set on parameters on which the algorithm runs and these parameters if manipulated will cause system failure.

II. LITERATURE REVIEW

[1] This paper refers to the numerous computer machinery and software for digitally creating, collecting, storing, manipulating and transferring the knowledge needed for basic tasks of institute. The On-line help desk is a combination of multiple sub modules which are required for college department automation activities. This includes 1.Automatic Timetable generation which is used for generating the time table for the course, which also includes adding, deleting and updating a course, faculty, classroom and lab to generate the timetable. 2.Automatic Internals timetable generation which is used for generating the time table for the internal examination. 3.Student profile management which includes adding, deleting and updating student's personal information, previous education information, academic information and other non-academic information. 4.Library management which is used for barrow and return of books from department library by students and faculty. Using Genetic algorithms all these modules are implemented. Genetic algorithm falls under the category of evolutionary algorithms that use the natural selection principle to determine a set of solutions for the optimal solution.

[2] University Course Timetabling Problem (UCTP) is an academic timetabling problem that deals with the task of assigning educational events (lectures, tutorials, and

laboratories) and lecturers to timeslots and classrooms or lab; to generate a weekly schedule. In this work, they have conducted a survey to research the satisfaction factors of a course timetable for a Practical University Course Timetabling Problem at Fakulti Teknologi dan Sains Maklumat, University Kebangsaan Malaysia. This paper is organized as follows. Section 1 briefly describes the TSMUCTP i.e. Fakulti Teknologi dan Sains Maklumat-university Course Timetabling Problem. Section 2 presents the questionnaires for both lecturers and students, and the result we obtained from this survey. In Section 3, they have discussed the result we gathered from the questionnaires. This survey has clearly demonstrated three types of schedule patterns based on students and lecturer's satisfaction level: the first type is the not-cared patterns which have no influence on the users' satisfaction when these patterns appear in the course timetable. The second type, like in the literature, the penalty pattern, which will decrease the users' satisfaction level when this type of pattern exists in the timetable. The third type of schedule pattern is the reward patterns.

[3] This paper introduces an evolutionary algorithm (EA) approach to solving a severely restricted university timetable problem. The method uses a trouble specific representation of the chromosome. During a rational computational time, heuristics and context-based reasoning are used to achieve realistic timetables. To speed up the convergence, an intelligent adaptive mutation scheme was used. The detailed course timetable structure that was introduced during this paper was validated, checked and addressed using outsized university world data. During this paper, the development of timetable algorithm during a large university department is demonstrated by using the cohort of second year EE (EE2) as an example. They have implemented their paper using the following approach. Chromosome Representation, Initialisation, Initial Population Generation, Crossover, Mutation, Assessment, Selection. The paper addresses the Timetable Problem (TTP), which covers a wide variety of real issues that are constantly faced in educational institutions, and explains how Evolutionary Algorithms (Eas) can be used to resolve arbitrary instances of automated timetable problem effectively.

[4] This project introduces a practical timetable algorithm which can effectively take care of both strong and weak constraints, used in an automatic timetable system. So that once they are finalized for a given term, each teacher and student can access their timetable but they can't edit them. Timetable Generation System generates timetable for each class and teacher, adheres to teacher availability calendar, availability and physical resource capacity (such as classrooms, laboratories, and computer rooms) and rules for different classes, semesters, teachers, and subject levels. The main component in this paper is to produce as output the HTML-based timetable even / odd 4 semester sheet. In which different user inputs such as Teacher List, Course List, Semester List, Room List, Day List and Timeslot are also taken as different rules, facts and constraints using web-based forms that are stored in the knowledge domain based on XML. This knowledge base is input to our server-residing Timetable Generator Algorithm. Knowledge base is in the middle, as it's between our timetable algorithm and the GUI front that's meant in the last. After standardizing the representation of KB, we developed the algorithm for the timetabling.

[5] This paper studies the timetable problem of university courses during a case study associated with a faculty management system that takes care of assigning classes and time slots to students / readers. A two-stage heuristic approach is provided, in which the initial stage groups the courses ready to be performed at the same time. Instead, the second stage assigns the weekly timeslots for each group of courses, followed by the position for each course. Computational tests shall be reported using real data for the proposed solution. This demonstrates that the proposed solution is successful in handling the timetabling of the school course. A two-stage heuristic is proposed to unravel this course timetabling case study. This is often thanks to its simplicity to deal with different hard and soft constraints in two stages. Stage I (Course Grouping) during which all the courses are going to be divided into a couple of groups supported the subsequent constraints:
1.Students can max enrol one course in each group.
2.Lecturers can max enrol one course in each group.
3.Takes under consideration the repeating students to enable them to enrol in certain courses and Stage II (Timeslot Allocation) during this stage takes under consideration time-related constraints:1. One timeslot allocated for only one group. 2.Spread the lecture time for every major/program evenly throughout the weekdays.

III. Problem Statement

Timetable generation manually requires time and it's difficult for the person to handle all the constrains that need to be considered. The hand operated time table planning method in colleges is usually very monotonous and time-consuming, resulting in either the same teachers ending up with more than one class at a time or a number of classes overlapping at the same classroom. Owing to a non-automatic viewpoint, total resource use has proved ineffective.

The solution for the above problems will be handled by the software, as we will be providing a separate module of validation where all the constraints will be checked for providing an optimal solution and all the constrains will be satisfied by our system. The proposed system is used to generate time table automatically. This ensures the following features:

- Easier slot assigning
- Less time consumption
- No clashes of the resources

IV. Existing System

Existing system used by college is not automated instead it's manual process which is done. The process follows following step to generate timetable.

1.Preparing the load sheet, which includes subject, faculty, load of subject, etc.

2. Time table coordinator starts with assigning practical's as they are of 2 hours and require multiple faculties at a time depending on number of batches the division has.

3.Once all the practical's for all the semesters are assigned, next step involves assignment of theory lectures, theory lectures are placed in vacant slots on the time table grid by crosschecking the faculty availability with all semester practical slots.

4.Finally after assignments of all the practical's and theory lectures, final time table is sent for approval to head of department and after approval multiple views are generated from the same.

V. PROPOSED SYSTEM

Proposed system is the system we have created and is successfully generating timetable mechanically by taking only limited input such as subjects, subject type (Practical or Theory), class rooms, laboratories, faculties name, semester and load of each faculty per subject, in csv format. These inputs will be processed further and algorithm will be applied on it, all steps of algorithm (Mentioned in section methodology and working) will run and the system will check whether all the requirements are fulfilled or not. Once, all the requirements are fulfilled system will generate the timetable and will be displayed and asked for confirmation. After confirmation, system will be store timetable in Sql database. It will provide different views of the timetable like class timetable for the students, laboratory time table for respective laboratories, faculty timetable for respective faculties and master timetable for timetable coordinator, all this view can be exported to csv format anytime.

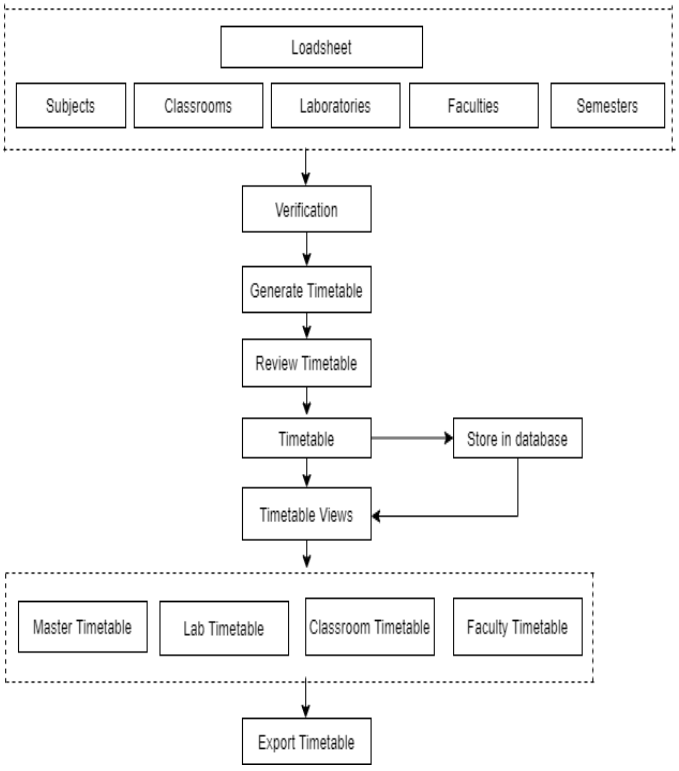


Fig 4.1 Proposed System Architecture

As soon as the user opens the system, he/she will see the home page where he/she will be asked to upload the load sheet, there will be a sample format file which the user can download and modify as per his/her requirement. Once, the user uploads the load-sheet he/she will be shown the data of load sheet and will be asked to verify the data, after verification the data will go the algorithm where faculty name will be considered for generating the population for Genetic algorithm which will be stored in a array and according to the faculty name another array will be generated having load of that faculty, there will be 3 such pairs each one for second year, third year and final year respectively. Once the population will be generated Genetic algorithm will start it's process by selecting any random faculty from array and placing the faculty at random position in timetable grid according to the load assigned, this process will continue until all faculties have been assigned for their respective loads. The algorithm will check for clashes while assigning the faculty by checking the positions current status, if it would be free it will assign the faculty there or else go to another position. In such a way the algorithm will assign all faculties without clashes for all the 3 semesters, once the timetable will be generated it will be displayed to the user and if user approves the time table it will be stored in the database and if user doesn't likes the allotment he/she can re-generate the time table and the algorithm will run again and give a new output, user can get n combinations of output using the regenerate button. The approved and stored time table can be viewed any time. The generated result will fulfil all the required constraints such as:

1. The slots which are assigned for the laboratories must be same for each year and the slots which are assigned for mentoring sessions must be same for all the year students.
2. Slots should not overlap likewise if some faculty is teaching in second year then their slot should not be assigned in third year or final year during that particular time.
3. Slots for compulsory subjects should also get assigned properly taking into consideration there should be no conflict.
4. There are slots which has been allotted for the departmental electives. Elective lecture may be in the classroom or in the laboratory depending upon the requirement of elective.
5. There are slots which is particularly reserved for the projects such as for final year there is 1 day in week which is reserved for project, for third year 4 hours in anyone of weekday is reserved likewise.
6. The generated timetable will also take care that each faculty workload should be equal to the load provided by the institute.

VI. METHODOLOGY AND WORKING OF PROPOSED SYSTEM

Proposed system works on genetic algorithm, which was finalised by after comparing multiple algorithm which follow similar pattern. Genetic algorithm is a search heuristic that is inspired by Charles Darwin's theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction in order to produce offspring of the next generation.

Genetic algorithm follows:

BEGIN /*genetic algorithm*/
Initial population generation;
Fitness check of individuals;
WHILE NOT finished DO LOOP
BEGIN
Mating done by selecting previous generation individuals;
Generation of offspring by crossing over/mutation of the previous generation individuals;
Fitness check of new individuals;
Removing previous generation and using new generation;
IF Population has converged
THEN finishes:= TRUE;
END
END

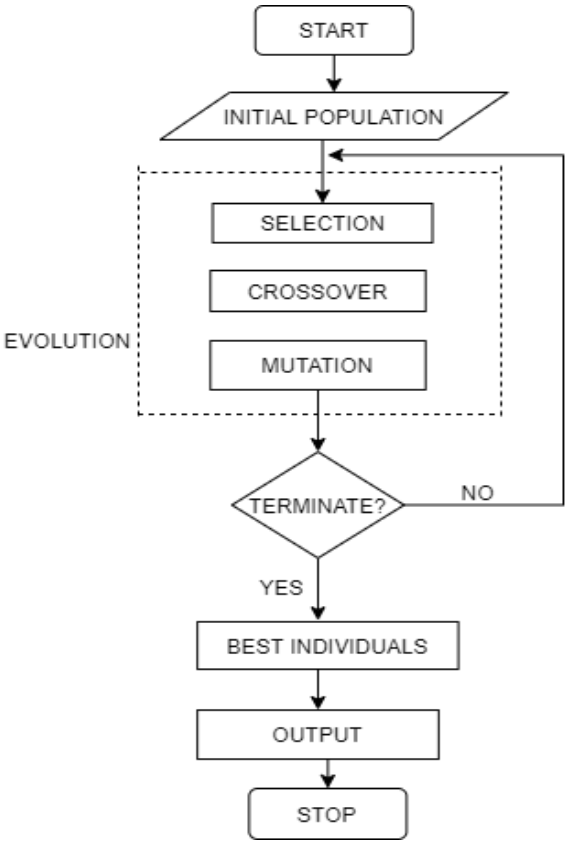


Fig 4.1 Genetic Algorithm Architecture

General Genetic Algorithm steps and their use in our project:

Step 1: Create initial population.
An initial population is created from a random selection of individuals. (which are analogous to chromosomes).

In our project population generation is the step where faculty abbreviation is the component which act as chromosome and the set of faculties abbreviation together forms our population.

Step 2: Evaluate fitness:
A fitness value is assigned to check solution depending on how cloe it actually is to solving the problem.

In our project the fitness value is the load assigned to faculty by the institute. There is a separate list created having data about the load of every faculty in reference with the index value of the faculty abbreviation list.

Step 3: Reproduce (and children mutate) :
The individuals with a greater fitness value are selected to generate new individuals (which can mutate after reproduction). New individual has features of both the parent individuals. (This generation is achieved by 'crossing over' method).

In our project this step comes into picture where the algorithm after assigning slots to a single faculty picks up another faculty from the faculty list and start placing that faculty into slots available, if the new faculty is provided a slot already occupied by the previous one the algorithm roll backs and assigns new slot to the faculty.

Step 4: Final generation: If the latest generation has the fitness value equal to assigned value the solution is considered optimal.

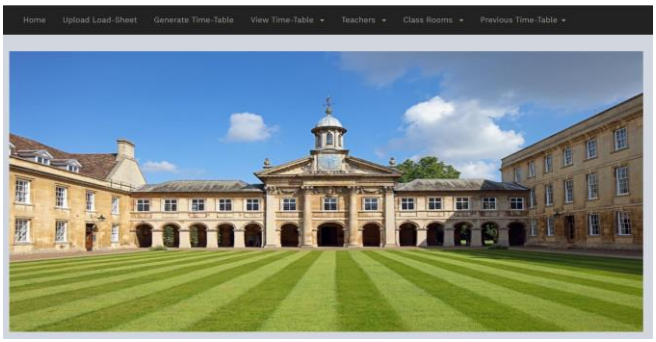
In our project this stage is the stage at which the timetable gets generated trying all the combinations and providing the best solution.

Each iteration consists of following steps:

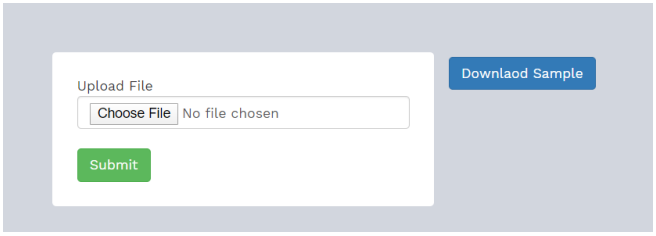
1. Selection: The first step consists of choosing individuals for reproduction. This section is finished indiscriminately with a likelihood betting on the relative fitness of the individuals that best ones are usually chosen for reproduction instead of the poor ones.
2. Reproduction: In the second step, offspring are bred by selected individual in order to generate new chromosomes, the algorithmic program will use each recombination and mutation.
3. Evaluation: Here fitness of new combination is evaluated.
4. Replacement: New combination are updated into algorithm by discarding the previous ones.

The process stop's when the population ends and an optimal solution is achieved.

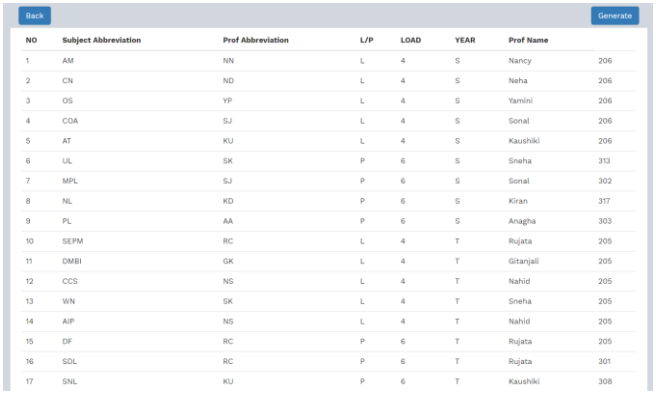
VII. RESULTS



Home page

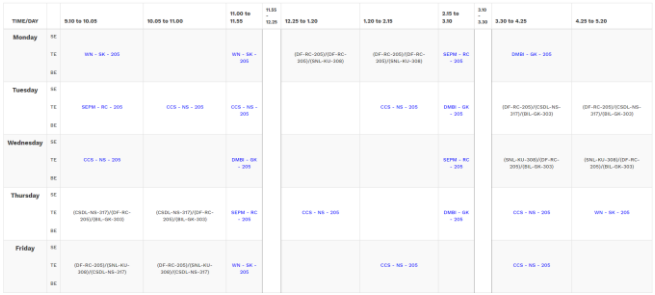


Upload Load Sheet

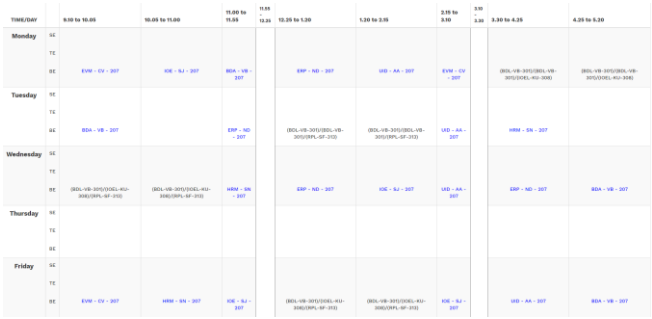


NO	Subject Abbreviation	Prof Abbreviation	L/P	LOAD	YEAR	Prof Name	
1	AM	NN	L	4	S	Nancy	206
2	CN	ND	L	4	S	Neha	206
3	OS	YP	L	4	S	Yamini	206
4	COA	SJ	L	4	S	Sonal	206
5	AT	KU	L	4	S	Kaushiki	206
6	UL	SK	P	6	S	Sneha	313
7	MPL	SJ	P	6	S	Sonal	302
8	NL	KD	P	6	S	Kiran	317
9	PL	AA	P	6	S	Anagha	303
10	SEPM	RC	L	4	T	Rujata	205
11	DMBI	GK	L	4	T	Gitanjali	205
12	CCS	NS	L	4	T	Nahid	205
13	WN	SK	L	4	T	Sneha	205
14	AIP	NS	L	4	T	Nahid	205
15	DF	RC	P	6	T	Rujata	205
16	SDL	RC	P	6	T	Rujata	301
17	SNL	KU	P	6	T	Kaushiki	308

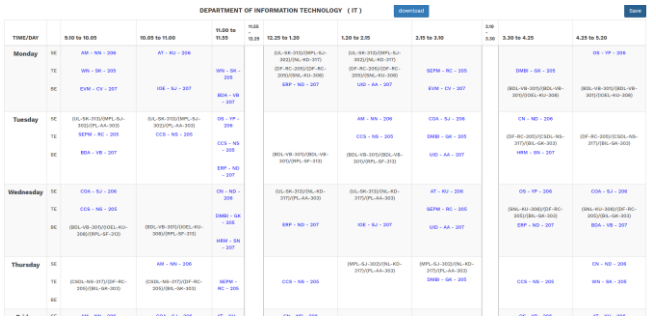
Confirm Laod Sheet Data



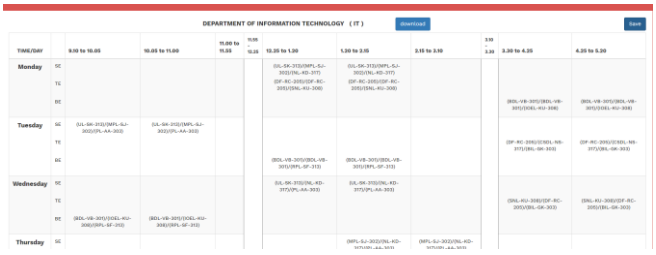
TE



BE with 1 day blank for project slot

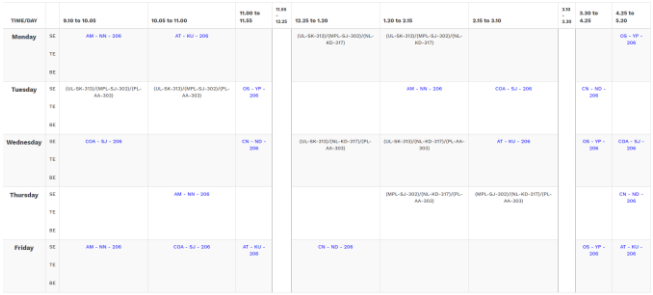


Master View

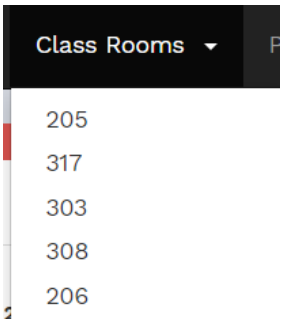


Practicals

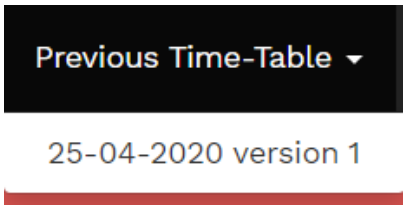
Faculty Time Table



SE



Class And Lab Time Table



Previous Sem Time Table with Master View

VIII. Conclusion And Future Scope

Automatic Timetable Generator using Genetic Algorithm has made it possible to automize the process of generating the timetable taking into consideration all the basics constraints kept in mind while generating the timetable manually, it uses genetic algorithm steps to implement the process. The system is efficiently generating the timetable fulfilling all the requirements and that to in few minutes, which in manual process took hours for same output to be generated.
To use the system there is a basic input format which needs to followed to get the algorithm working, any place where the input format is followed, this system can be used. For example, it can be used by all institutes, schools, etc.

Future Scope for the system:
1. Implementation of various constraints.
2. If required working on security factor.
3. Assignment of slots for various activities.
4. Generation of special timetable i.e. one day timetable or occasional time table.

References

[1] Asha, V. G., Babu, K. N. R. M. On-line help desk for college departmental activities. International Conference on Intelligent Computing and Control Systems (ICICCS) IEEE (2017).

[2] Yang, X. F., Ayob, M., Nazri, M. Z. A. An investigation of timetable satisfaction factors for a practical university course timetabling problem.6th International Conference on Electrical Engineering and Informatics (ICEEI) IEEE (2017).

[3] Ilham, N. I., Saat, E. H. M., Rahman, N. H. A., Rahman, F. Y. A., Kasuan, N. Autogenerate scheduling system based on expert system. 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE) IEEE (2017).

4] Anuja Chowdhary et al, TIMETABLE GENERATION SYSTEM, International Journal of Computer Science and Mobile Computing, Vol.3 Issue.2, February- 2014.

[5] Bong Chia Lih, Sze San Nah, Bolhassan, N. A. A study on heuristic timetabling method for faculty course timetable problem. 9th International Conference on IT in Asia (CITA) IEEE(2015).