Automatic Time-Table Generation System using Genetic Algorithm

Amey Salvi[1], Omkar Khanvilkar[2], B.W. Balkhande[3]

Department of Computer Engineering, Bharati Vidyapeeth College of Engineering, Navi Mumbai, India[1, 2, 3]

Abstract: Time-Tables are very important for any university administration. It informs the students and teachers when and where to be, the availability of the rooms and the availability of the teachers. But, the generation of time-tables manually is a very tedious task and is almost always inconvenient. This project aims at generation of course schedules automatically in a manner which ensures that they are optimal and thorough with little or no redundancy. It involves taking basic constraints and the structure of a time-table and incorporating them into the genetic algorithm. This system would serve as an ideal platform for less strenuous work by providing the administrator with an interactive interface for the time-table generation for all courses and departments.

I. INTRODUCTION

The existing system drafts a time-table by taking into consideration that each lecture contained one class of students, one professor, and a certain number of time slots which could be chosen freely. Initially it was mostly applied to schools. The problem in schools being relatively simple because of their less complex class structures, classical methods, such as linear programming approaches could be used conveniently. However, bearing in mind the cases of higher secondary schools and universities, which contain complex class-structures, increases the complexity of the problem. The existing system generates a lot of paperwork and is not flexible as changes can be difficult to make. Classical methods are thus found derisory to handle the problem, predominantly because of the vast number of variables and multiple functions [3].

The aim of this project is the automatic generation of course schedules using the principles of Genetic Algorithm which would prove to be instrumental in reducing the high cost and slow turnaround involved in generation of near-optimal time-table.

This can be implemented using various technologies such as HTML5, JavaScript, CSS3, J2SE SDK, MySQL and Tomcat Apache. HTML5 is a markup language used for structuring and presenting content on the World Wide Web. Cascading Style Sheets (CSS) is a style sheet language which is used to present a document written in a markup language in a manner which is more pleasing on the eye. JavaScript is a high level, dynamic and interpreted programming language. It has been standardized in the ECMA Script language specification. Combined with HTML and CSS, JavaScript is one of the three critical technologies of World Wide Web content production. Tomcat Apache is a web container developed at the Apache Software Foundation. Tomcat is a web server that is used to implement servlets and JSPs. Tomcat comes with the Jasper compiler that compiles JSPs into servlets. J2SE SDK lets you develop and deploy Java applications on desktops and servers. Java offers the opulent user interface, performance, versatility, compactness, and

security that today's applications require. MySQL is the world's most widely used open-source database. It is the most supportive database for PHP [2].

II. PROJECT STATEMENT

The most common issues faced during time-table scheduling can be represented as a constraint satisfying problem with multiple constraints. The goal of portraying these constraints in an organized fashion is attained by replicating the format by the algorithm.
Multiple tasks can be achieved simultaneously by allowing these constraints to be paired in a number of different ways [1]. It is desirable for timetables to satisfy both the hard and soft constraints. However, it is classically not easy for all these constraints to be satisfied. The soft constraints can be abandoned in order to find a convenient solution. The hard constraints though, are a must to be met at all times.

III. REQUIREMENT ANALYSIS

The current methods used have a slow turnaround and the goal of the most optimum time-table to be generated is not mostly attained. The most persistent or obstinate problem is that of scheduling. During the process of scheduling many constraints need to be considered. University timetable scheduling problem presents a set of tasks (classes) and a set of resources (rooms, groups, instructors) to pick from.

The objective is to assign the tasks to their resources while sustaining all of the hard constraints – no resource should be allocated by multiple tasks at the same time. The scheduling is a process of placing those pegs into a timetable, in a manner that no conflicting classes (those which allocate the same resource, either a batch or an instructor) are placed in that exact timeslot.
The proposed system was developed to resolve the timetabling issue which is being faced by almost every college each academic year and limit the high cost and

slow turnaround involved in the generation of near-optimal timetables. The system has capabilities for input of the various courses, rooms and computer labs for lectures and labs respectively, departments, programs, lecturers and the specifications of a few constraints by means of which the timetable is fashioned. The proposed system for this project seeks to create optimal timetables using the principles of genetic algorithm (concepts of selection and crossover).

The major advantages of this system are:
- It utilizes minimal processing or computing power.
- Unlike the manual timetabling system, the proposed system offers flexibility.
- It's instrumental in reducing the time necessary for the generation of near-optimal timetables.
- It increases productivity.
- It almost eliminates paperwork.

IV. SYSTEM ANALYSIS

The main components of this system are registration, login, entering details, looking at and reviewing the resulting time-table.

The proposed system provides the user to have an easier and less monotonous way to generate a time-table. The system provides an interface to enter details regarding course, rooms, programs, lecturers and a few further specifications which are input into the database.
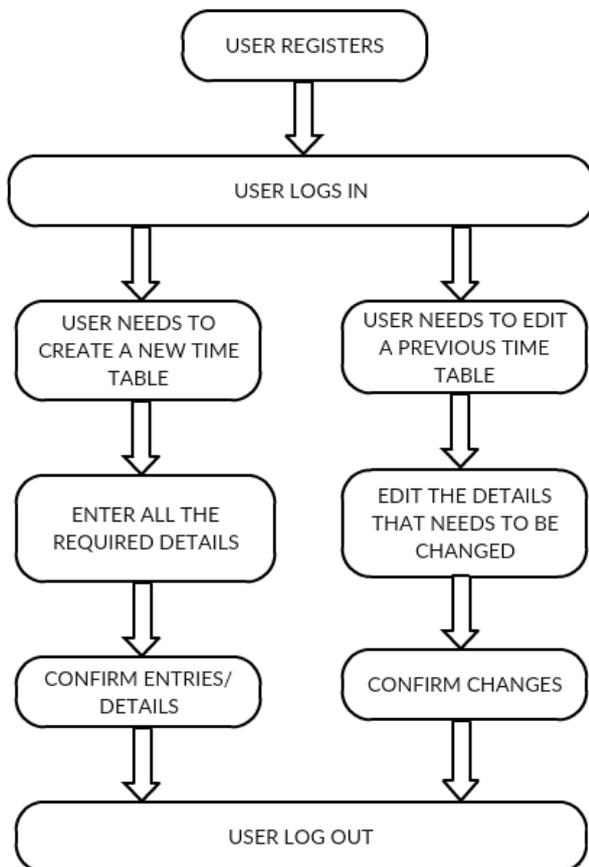

Figure 1: System Flow Chart

The users have to register themselves only then can they be able to proceed into the system. The registered members have to login first in order to access their account. Consequently, the users need to enter details regarding themselves and the course they are in charge of. The administrator will then be entrusted to generate a time-table based on these entries from the entire staff.
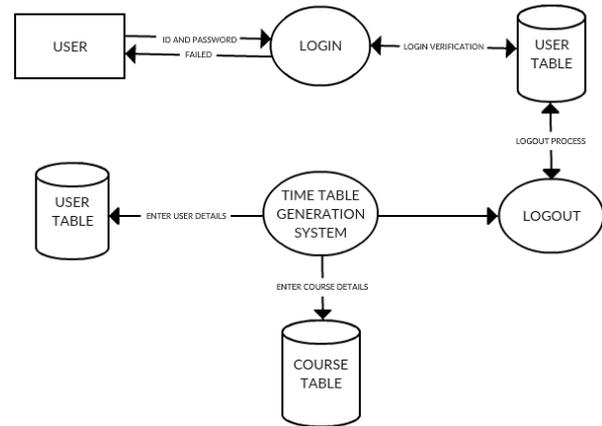
V. SYSTEM ARCHITECTURE

Figure 2: Data Flow Diagram

The system architecture comprises of primarily three parts namely Graphical User Interface (GUI), front end and back end.

The technologies majorly used areJ2SESDK, MySQL and Tomcat Apache. Java Platform Standard Edition (Java SE) lets you develop and deploy Java applications on desktops and servers in today's demanding embedded environments. Java offers the opulent user interface, performance, versatility, compactness, and security that today's applications require. It gives customers and enterprises features that minimize the costs of deployment and maintenance of their Java-based IT environment.

MySQL is well known as world's primarily used open-source database (back-end). It is most supportive database for PHP as PHP-MySQL is the most frequently used open-source scripting database pair. The user-interface which WAMP, LAMP and XAMPP servers provide for MySQL is easier to work with and reduces our workload to a considerable extent [2].

Tomcat Apache is a web container developed at the Apache Software Foundation. It is used for the addition of tools for configuration and management but can also be configured by editing configuration files that are ordinarily XML-formatted. Tomcat includes its own HTTP server internally. The Tomcat servlets engine is often used in collaboration with an Apache web server or other web servers. Tomcat can also function as an independent web server. Since its inception in Java, Tomcat runs on any operating system that has a JVM [2].

VI. ADVANTAGES

- A compatible and precise timetable is guaranteed and the system is therefore well organized and reliable.

- Allocation of periods to classes and professors is completely automated and does not restrict the user any processing.
- It offers an easy means for data entry and revision through an intuitive interface.
- Time-tables generated are between 60% to 80% optimum.
- The system can be maintained without any hindrance by the developer.

The developed application will enable users to input details in a hassle-free manner. It enhances the user experience and provides a comforting approach to generate a time-table.

VII. CONCLUSION

The proposed system will make the procedure of time table generation easier considerably which may otherwise require to be done using spread sheet manually which would more than often lead to problems that are arduous to overcome. The algorithm includes many techniques, aimed at improving the efficiency of the solution generated. It also aims at addressing the hard constraints. Most of the flexible soft constraints are also prolifically taken care of.

A number of hours which are spent on creating a fulfilling timetable can be reduced through the automation of the timetable.

REFERENCES

[1] Mugdha Kishor Patil, Rakhe Shruti Subodh, Prachi Ashok Pawar and Naveena Narendrasingh Turkar, MVPS's KBT COE, Nashik– Web Application for Automatic Time Table Generation (2014).
[2] W3Schools - MySQL, Tomcat Apache, HTML, CSS, JAVA.
[3] Dipti Srinivasan, Tian Hou Seow, Jian Xin Xu, National University of Singapore - Automated Time Table Generation Using Multiple Context Reasoning for University Modules (2002).