AUTOMATIC TIME TABLE GENERATOR

[1]**Mitali Algamwar,** [2] **Vaishnavi Gogulwar,**[3]**Jahnavi Hedau,**[4]**Sanket Meshram**

[5] **Prof.Anand Donald**

Students, Department of Computer Science & Engineering[1,2,3,4]

Professor, Department of Computer Science & Engineering[5]

Rajiv Gandhi College of Engineering Research and Technology, Chandrapur, Maharashtra, India

ABSTRACT :- This project is aimed at developing a Time Table Generator for Colleges. Timetable generation is a very well known problem that is faced in every educational institution. Most colleges have a number of different courses and each course has a number of subjects. Now there are limited faculties, each faculty teaching more than one subjects. So now the time table needed to schedule the faculty at provided time. slots in such a way that their timings do not overlap and the time table schedule, makes best use of all faculty subject demands. The Automatic Timetable Generator system will take various inputs like details of faculty, students, subjects and depending upon these inputs it will generate a possible time table, making optimal utilization of all resources in a way that will best suit the specified constraints or college rules. Automatic Timetable Generator system is an automated system which generates time table according to the data given by the user. The main requirement of the application is to provide the details about the branch, semester, subjects, labs and total no. of periods. system is an automated system which generates time table according to the data given by the user. . The main requirement of the application is to provide the details about the branch, semester, subjects, labs and total no. of periods. List of subjects may include electives as well as core subjects .Students need to elect their electives if any. Then the application generates the time table according to your need.

KEYWORDS: Time tabling, Timetable Generator, Constraints, Timeslots, scheduling.

Planning timetables is one of the most complex and error-prone applications. There are still serious problems like generation of high cost time tables are occurring while scheduling and these problems are repeating frequently. There for there is a great requirement for an application distributing the course evenly and without collisions. Our aim here is to develop a simple, easily understandable, efficient and portable application, which could automatically generate good quality timetables with in seconds. Sure, an automatic timetable generator using Python and Tkinter is a tool designed to create schedules for various activities or events. Tkinter is a standard GUI (Graphical User Interface) toolkit for Python, providing a set of tools for building graphical applications. The automatic timetable generator typically takes input parameters such as the list of activities, their durations, and any constraints or preferences, and then generates an optimized timetable that meets these criteria. Using Tkinter, users can interact with the application through buttons, text fields, and other graphical elements to input data and view the generated timetable. The generator algorithm then processes this input to produce an efficient timetable layout, which can be displayed within the Tkinter interface or exported to other formats for printing or further use. Overall, this project combines the power of Python for algorithmic processing with Tkinter for creating a user-friendly interface, resulting in a handy tool for generating timetables quickly and efficiently.

I.INTRODUCTION

II. LITERATURE REVIEW

In This paper is focused on the development of a Time Table Generator for Colleges. Timetable technology is a well-known issue that affects every academic institution. The problem arises as a result of disagreements, with some of the options provided by the staff. Colleges are expected to create timetables for each semester that change as quickly as a fully new semester.. Administrators added the important factors for adding the Student, add Staff, add Subject, Time Table Entry, Time Table Update [1]

In This paper Time table generation is tedious job for educationalist with respect to time and man power. Providing a automatic time table generator will help to generate time table automatically. Proposed system of our project will help to generate it automatically also helps to save time. It avoids the complexity of setting and managing Timetable manually. In our project we are going to use algorithms like genetic, heuristic, resource scheduling to reduce these difficulties of generating timetable. [2]

In This paper Generating timetable for university with many branches, different years, and multiple batches is a tiresome job. It takes a lot of time and is required for every semester that is of six months. Thus, a lot of time and manpower is taken up by this process. In some cases, the manual process of creating timetables is too cumbersome. Creating temporary timetables when a faculty is on leave is impractical. In this paper, we have created an algorithm to generate timetables which can save a lot of time and pressure on the person doing this job manually. [3]

IV. SOFTWARE SYSTEM

**Software Description:** Tkinter , python, vs code
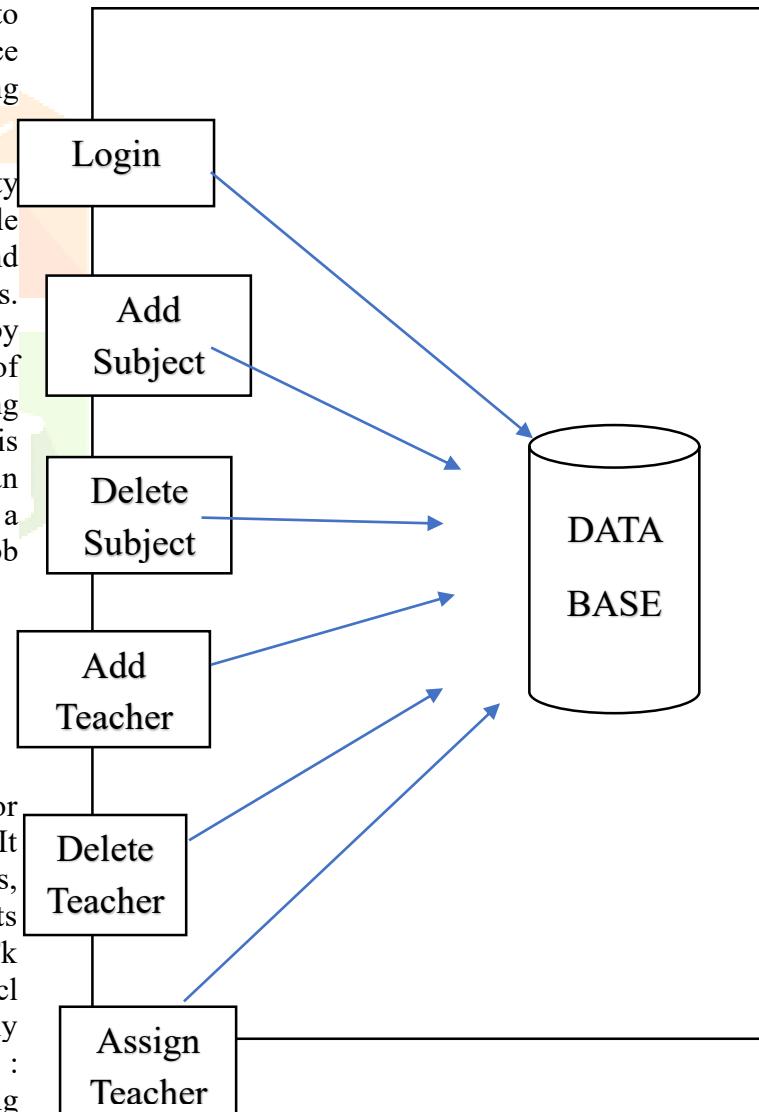
Tkinter : Tkinter is a standard Python library for creating graphical user interfaces (GUIs). It provides a framework for building windows, dialogs, buttons, menus, and other GUI elements in Python applications. Tkinter is based on the Tk GUI toolkit, which originated as part of the Tcl scripting language but has been ported to many other languages, including Python. **Python** : Python is a widely-used, high-level programming language known for its readability and simplicity. It is used for a variety of purposes, including web development, data analysis, artificial intelligence, scientific computing, and more. Tkinter is one of the libraries available in Python for creating GUI applications.

VS Code : VS Code (Visual Studio Code) is a popular open-source code editor developed by Microsoft. It supports various programming languages and features like syntax highlighting, code completion, debugging, and version control integration. VS Code is highly extensible through its vast ecosystem of extensions, which can enhance its functionality for specific programming tasks or languages.

V. HARDWARE SYSTEM

- Laptop or PC
- Windows 7 or higher
- I3 processor system or higher
- 8 GB RAM or higher
- 100 GB ROM or higher

III. ARCHITECTURE

VI.RESULT ANALYS

1.LOGIN PAGE

2.HOME PAGE

2[A1] ADD SUBJECT

2[A2]DELETE SUBJECT

2[B1]ADD TEACHER

2[B2]DELETE TEACHER
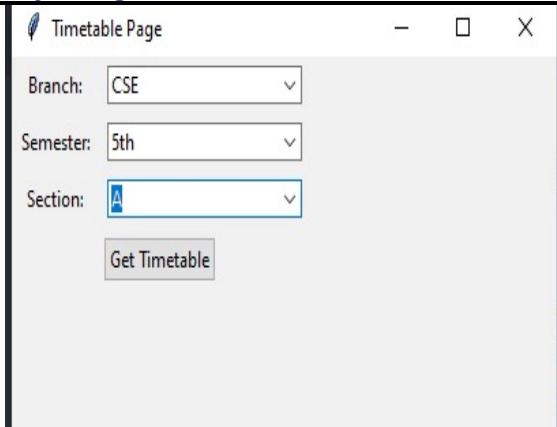
2[C] ASSIGN TEACHER

2[D]TIMETABLE PAGE

IV. FUTURE SCOPE

Enhanced User Interface: Improving the user interface to make it more intuitive, user-friendly, and visually appealing can enhance the user experience. Integration of modern design principles, themes, and interactive elements can make the application more engaging.

Advanced Scheduling Algorithms: Incorporating advanced scheduling algorithms and optimization techniques can improve the efficiency and effectiveness of the timetable generation process. Algorithms such as genetic algorithms, simulated annealing, or constraint programming can be explored to handle complex scheduling constraints and produce better timetables.

Integration with AI and Machine Learning: Integrating AI and machine learning techniques can enable the system to learn from past schedules and user preferences, thereby improving its ability to generate optimal timetables. Predictive analytics can be used to anticipate scheduling conflicts and suggest alternative solutions.

Mobile Applications: Developing mobile applications for Android and iOS platforms can extend the reach of the timetable generator, allowing users to access and manage schedules on their smartphones and tablets. Mobile apps can offer additional features such as push notifications, calendar integration, and offline access.

V. CONCLUSION

In conclusion, an automatic timetable generator created with Tkinter in Python offers a robust solution for efficiently managing schedules across various domains. Its future scope is promising, with opportunities for enhancement and expansion in several areas:

User Experience: Improving the user interface and incorporating modern design principles can enhance usability and engagement.

Advanced Algorithms: Integration of advanced scheduling algorithms and optimization techniques can improve efficiency and handle complex constraints effectively..

VI. BIBLIOGRAPHY

1. Ghrushneshwari Dattatray Mhaise, Chetana Sudhakar Kurhade, Mayuri Yadav Devre, Paya Sonawane, Tambe Balasaheb Parasram, "AUTOMATIC TIME TABLE GENERATOR",2023 International Research Journal of Modernization in Engineering Technology and Science, Yeola , India,2023,pp.1-4

2. Akshay puttaswamy, H M Arshad Ali Khan, Chandan S.V, Parkavi.A, "A STUDY ON AUTOMATIC TIMETABLE GENERATOR",2018 International Journal of Innovative Research and Growth, Bangalore,2018,pp. 1-4

3. Swapnil Gadekar, Pathan Danish Khan, Anil Salve, "AUTOMATIC TIME TABLE GENERATOR SYSTEM",2022 ijariie.com, Aurangabad, India,2022,pp1-5

4. K. R. Sugunamukia , M. Arun Anoopb, "A SMART TIME TABLE GENERATOR",2021 International Journal of Research Publication and Reviews,2021,pp1-4

5. Varsha S, Deekshitha S, Bhargav Gupta L S, Samith V, Vignesh S,Bhavana Gowda, "DESIGN AND IMPLEMENTATION OF TIME TABLE GENERATOR",2022 Journal of Emerging Technologies and Innovative Research (JETIR), Bengaluru, India,2022,pp1-4