Activity Scheduling (Automatic TimeTable Generator)

Bhoomil Dayani[1], Raj Busa[2], Divyesh Butani[3]

Smt. Kundanben Dinsha Patel Department of Information Technology,
CSPIT, CHARUSAT, Gujarat, India
[1]20it022@charusat.edu.in
[2]20it011@charusat.edu.in
[3]20it012@charusat.edu.in

***Abstract*— The purpose of the project is to find an optimal solution to the time tabling problem which is one of the highly constrained N-P hard problems. The need of this sort of a time tabling software arised as manually designing a time table takes too much time and effort and if overlap occurs among the timetable, the timetable is redesigned using hit and error methods which has very high time cost. So, in this project we are trying to develop such a software which will automatically and will decrease the effort and time required for the generation of a time table in accordance to the given input. The expected main input is about teachers, classes and subject's data along with the maximum workload of a teacher in a week to generate a valid time-table. The main constraints that this software should satisfy are that a teacher should not have a lecture in more than one class at the same time slot and a class should not have more than one lecture in a given time slot. The solution which we will get from this project should have to satisfy the above-mentioned constraints.**

***Keywords* — Time table scheduling, Genetic algorithm, Constraints, optimal solution.**

I. Introduction

In institutions with a big number of students, we now use a manual way of creating time tables. Here, the manual approach requires the instructor to prepare the timetable, which takes a lot of time. Most colleges offer a variety of courses, each of which contains a number of disciplines. There are now fewer faculties, and each faculty is responsible for teaching many subjects. As a result, the timetable now required to allocate the faculty to the available time slots in a way that prevented scheduling conflicts and made the greatest possible use of all topic requests from the faculty. For this, we employ a special algorithm. We suggested using a time table object in our approach for creating time tables.

Even though most faculty organisation tasks are now automated, creating lecture schedules is still typically done manually, which requires a lot of time and effort. In addition to being utilised extensively in schools, colleges, and other teaching settings.

In these scenarios, precisely designed time tables are reused for the entire generation without any alterations, which is dull. Other examples That difficulty arises when there aren't enough employers or employees, which forces schedule adjustments or the quick filling of vacant seats. They must plan their course so that it fits the requirements of the current time frame and the facilities at their disposal. However, they must adjust their schedule to accommodate both the new course additions and the newly enrolled students in the new batches. This can lead to rescheduling the entire timetable for all of its batches in order to schedule it as soon as feasible before the batch courses begin. Another issue that comes up when establishing exam schedules. When many batches are scheduled to take tests on the same day, scheduling must be done carefully to account for any issues with the facilities that are available to hold these exams concurrently.

II. LITERATURE SURVEY

A literature survey is a report that evaluates the data from the literature in relation to a suggested piece of work. A literature review, which summarises all prior research on a topic and establishes the framework for ongoing research, may be an essential component of a research endeavour. It is the most crucial section of your report since it directs your research in the most appropriate manner. It helps you to set a goal for your analysis.

evolutionary techniques used to solve the problem of scheduling time. Methods such as Genetic Algorithms, Evolutionary Algorithms etc. used with mixed success. In this paper, we have reviewed the problem of scheduling an educational timeline with a genetic algorithm. We also solved the problem with a mimetic hybrid algorithm, a synthetic genetic defense network and compared the result with that found in the genetic algorithm. The results show that GAIN is able to reach a possible solution faster than that of GA [1].

Discover the study schedule that is possible at the university's main department is a recurring problem facing academics. This paper represents an evolutionary algorithm (EA) approach based on solving the university's robust timetable problem. Moving to problematic chromosome representation. Heuristics and contextual-based thinking using timetables may have been obtained at the right computer time. An ingenious genetic modification scheme has been used to improve cohesion. The comprehensive curriculum plan presented in this paper is approved, evaluated and discussed using real-world data from a major university. C [2].

Suggests a common solution to the problem of timing. Most of the proposed previous heuristic programs of difficulty from the perspective of students. This solution, however, works from the point of view of the subject, that is, the availability of the instructor at a given time. Although all potential barriers (e.g., teacher availability, etc.) are solved firmly, the planning solution presented in this paper is flexible, with the primary purpose of resolving academic and academic conflict, teacher-related issues [3].

Introduces an effective timing algorithm that can effectively manage both strong and weak obstacles, which is used in an automated timeline system. So that each teacher and student can look at their timetable after they have completed a particular semester but do not plan. The Timetable Generation System generates a timeline for each class and teachers, in line with the teacher's calendar, availability and power of visual resources and other rules applicable to different classes, semesters, teachers and grade level [4].

Algorithm tools proposed to solve the problem of timing while providing teacher availability admissions. This algorithm uses a heuristic approach to provide a complete solution to the difficulty of scheduling school time. Initially it uses randomly generated title sequences to create a temporary timeline. If a teacher is divided beyond the maximum allowable subjects the subjects are transferred to the Clash data structure [5].

III. PROBLEM STATEMENT

A restricted satisfaction issue with numerous loose parameters can be used to mimic the problem of creating timetables. These issues should be modeled in a format that can be handled by the scheduling algorithm. Planning involves allowing a number of intelligent limits where tasks can be performed simultaneously. For example, in an attempt to organize classes at a tertiary institution, two courses taught by the same faculty may not be scheduled at the same time. Similarly, the two subjects required by the same group of students should not be contradictory.

IV. EXISTING SYSTEM

The construction of a timetable is tedious and time consuming. Since there aren't any active time table generators, this is currently done manually. The main issue while inputting the timetable is slot collisions. Slot collisions are the main issue while entering the timetable. As a result, even previously developed software does not follow the standards. So, the current system is time consuming, a tedious process that requires manual labor and simultaneously, with little flexibility.

The platform we used to develop this software is web application. The programming language used to implement this software is JavaScript. In the development of the UI, React.js is used while backend is implemented using node.js. To make it a multi user app login facility is provided and to store the data corresponding to the user, mongodb database is used locally. Project is currently run on local severs but we have planned it to deploy it using Heroku and we will also change the database from local to mongodb atlas.

This projected is completed in 4 steps.

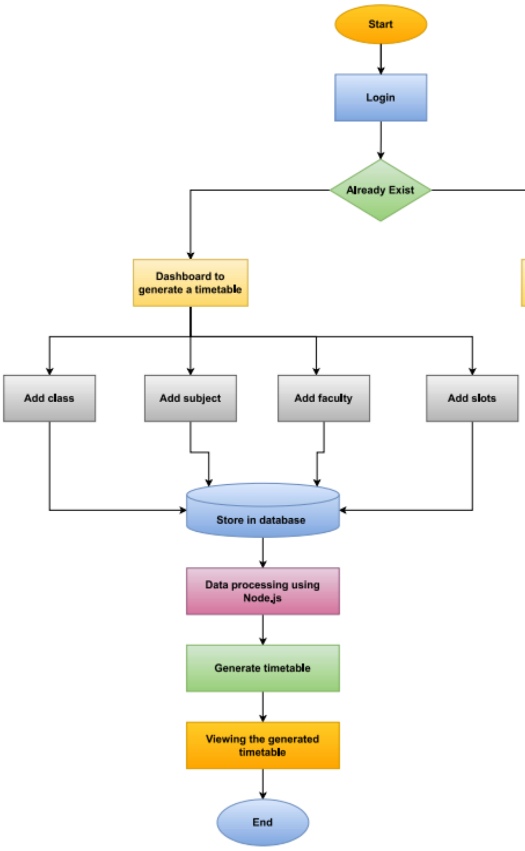
- Step1
  We have designed our algorithm for this problem by analyzing the problem deeply and designed the UI for the project.
- Step2
  In the second step our project we worked to improve the efficiency of our algorithm and calculated the time complexity of it. We also proved the correctness of our algorithm.
- Step3
  In this phase we implemented the algorithm on the backend and get the desired results by providing our dummy inputs.
- Step4
  Backend was implemented and Rest API’s were developed to interact with database and frontend
- Step5
  Frontend was developed and iteration of backend with our UI is done in this step.

The entire process of creating a schedule is done manually while considering all potential limitations, both large and little.

V. PROPOSED SYSTEM

Faculty members now create timetables manually. It requires lots of time. The creation of a semester schedule is one of the top duties at the beginning of each academic year. This may seem simple, but creating a schedule that takes into account everyone's availability over the course of all semesters can be a hard job. The manual process of creating timetables in most cases can be tedious and time-consuming for faculty members.

The final system should be able to construct timetables entirely automatically, which will save an institute administration a tonne of time and work. It will provide a schedule that can be used for all semesters as well. The timetable should be planned in accordance with the university's set schedule for each course and the workload of the faculty members who will teach the corresponding disciplines. This also emphasises the efficient use of resources, such as academic personnel, labs, and spaces. These inputs will be used to build potential time tables for the working weeks days for teaching faculty. This will integrate by using all resources as efficiently as possible while taking into account the limits.

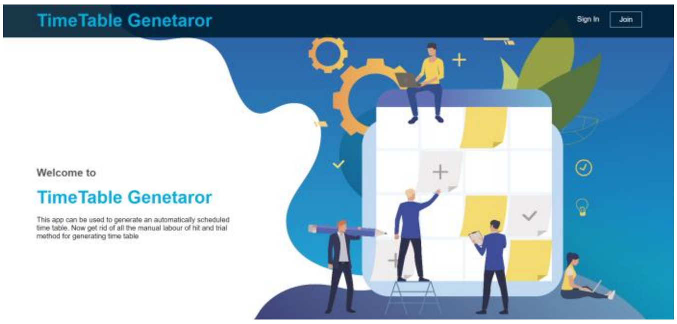


Home Page contains a Login and Registration Page, where the teacher will fill in some basic information about him, her appointment, and the department to create a teacher account in the application. You will be taken to the homepage, where you will find the several Departments and a menu bar with choices for adding classes, subjects, teachers, and spaces to the currently selected page. If you click on the home button, you will be taken to the appropriate page. And the database has all information at all times. Check out all the information on all the professors, classes, times, and subjects. Then Once your task is complete you can exit the app. An application to connect to a website. With this we can save all the fields that will be useful to keep all the details of the timetable or we can save the names of the teachers of the different classes or branches. With this website we can create, add, update or delete teacher information as per your concerns. We can use the website to store teacher information and thus be able to create our own timetable. You will be taken to a website with sections for First, Second, and Third; after selecting one, you will be taken to a page with a faculty name and a button that, when clicked, will display a timetable. There are other details about the professors to be added when you click on "Add Faculty, Classroom and Times," including Teacher Name, Subject, Lecture Timing, Slots, Semester, Credite, and Generate.

VI. RESULT ANALYSIS

1. **Welcome Page**
   The welcome page has two controls which are signup and login. If someone is using this software for the very first time, he needs to register and signup button will open a window where he can register him or her. But if user has already registered himself/herself or used this software before, he simply needs to click on the login button which will lead him/her to a window where he/she need to put the credentials.

- Signup: A user should use this to register himself/herself.
- Login: A user should use this to login in to system.



Welcome to

TimeTable Genetaror

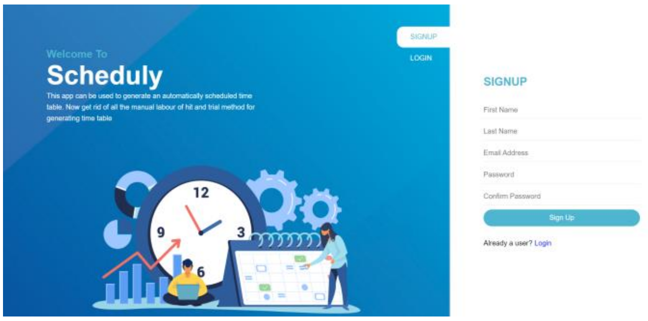
This app can be used to generate an automatically scheduled time table. Now get rid of all the manual labour of hit and trial method for generating time table

(Figure 1: Welcome Page Interface)

2. **Signup**
   Whenever a new user wants to use this system, he must register. Sign up page consists of 5 input boxes and a submit button.

- Login: To log in to the application.
- Signup: To register to use this software.
- First Name: A user should fill this input box by providing his/her first name.
- Last Name: A user should fill this input box by providing his/her last name.
- Email Address: A user should fill this input box by providing his/her email.
- Password: A user should fill this input box by providing a strong password. This password will be re required when user wants to login into the application.

- **Confirm Password**: A user should fill this input box by re writing his/her password.
- **Sign Up Button** (Right below input boxes): This button is used to submit the credentials.



This app can be used to generate an automatically scheduled time table. Now get rid of all the manual labour of hit and trial method for generating time table

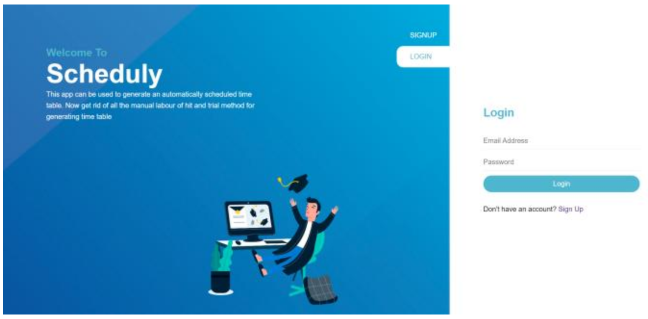
Already a user? Login

(Figure 2: Sign-up page Interface)

3. Log In

This page consists of two input boxes and a login button. If the credentials are correct, you will be led to dashboard.

- Email Box: A user should be required to fill this input box with a valid email which he used to register him/her self.
- Password: User should enter his password in this box.
- Login button: By clicking on this button user will be directed to the dashboard if the credentials are turned out to be true.



This app can be used to generate an automatically scheduled time table. Now get rid of all the manual labour of hit and trial method for generating time table

Don't have an account? Sign Up

(Figure 3: Log-in Page Interface)

4. Dashboard

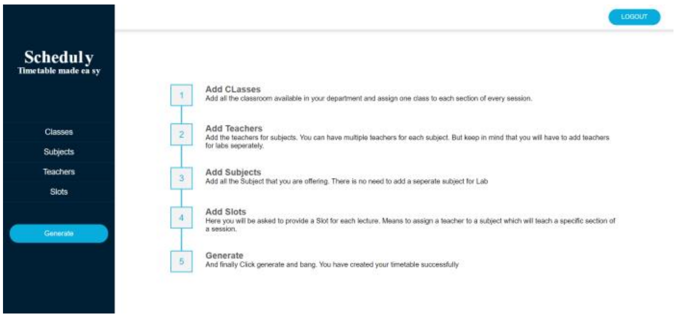
This page will be shown after a user successfully login to the application. It consists of a header and a navigation panel.

In header

- Home Button: This will lead to the dashboard whenever someone presses it.
- Logout Button: This will log out a user from the application and takes him/her to welcome screen.

In navigation panel, there are 5 buttons which are described below.

- Classes: It has a submenu which shows Add and All Classes Button. By clicking the Add button, a new page will be opened where user can add a new class. By clicking on the All Classes button, all the classes will be shown on the UI.
- Subjects: It has a submenu which shows Add and All Subjects Button. By clicking the add button, a new page will be opened where user can add a new s ubject. By clicking on the All Subjects button, all the subjects will be shown on the UI.
- Teachers: It has a submenu which shows Add and All Teachers Button. By clicking the add button, a new page will be opened where you can a new teacher. By clicking on the All Teachers button, all the teachers will be shown on the UI.
- Slots: It has a submenu which shows Add and All Slots Button. By clicking the Add button, a new page will be opened where you can add a new slot. By c licking on the All Slots button, all the slots will be shown on the UI.
- Generate: This button will send a request to the backend to run algorithm and will show the returned output.



1 Add CLasses
Add all the classroom available in your department and assign one class to each section of every session.

2 Add Teachers
Add the teachers for subjects. You can have multiple teachers for each subject. But keep in mind that you will have to add teachers for labs seperately.

3 Add Subjects
Add all the Subject that you are offering. There is no need to add a seperate subject for Lab

4 Add Slots
Here you will be asked to provide a Slot for each lecture. Means to assign a teacher to a subject which will teach a specific section of a session.

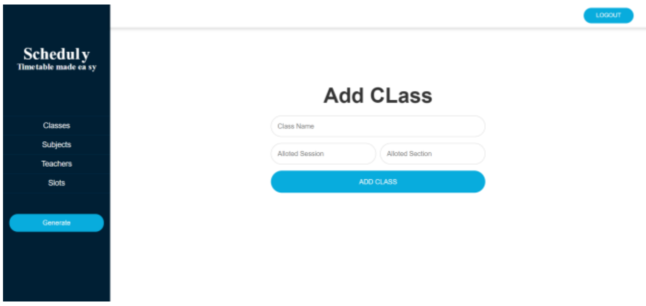
5 Generate
And finally Click generate and bang. You have created your timetable successfully

(Figure 4: Dashboard Page Interface)

5. Add Class

To add a new class, you need to click on classes in navigation panel and select add from submenu. On this page user will be able to add a new class. It has 3 input fields and a submission button.

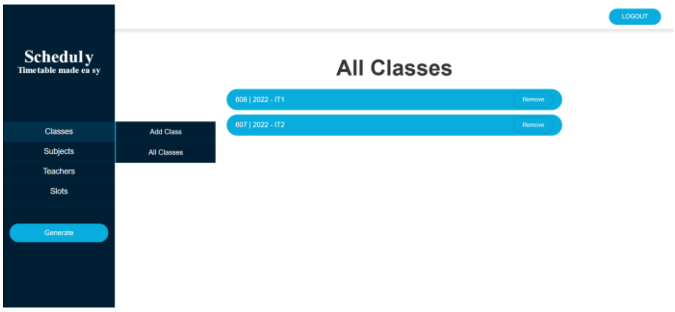
- Class Name: User need to add a class name such as N-7, N-4.
- Session: This input field requires a session name such as 2018, 2020.
- Section: User need to enter the section of the class.
- Add Class Button: This button will save the records.



(Figure 5: Add Class Page Interface).

6. View All Classes

This will show all the classes added by a user. You can go to this page by clicking on classes in navigation panel and select all classes from submenu. It has only one button Remove which will remove a class.



608 | 2022 - IT1

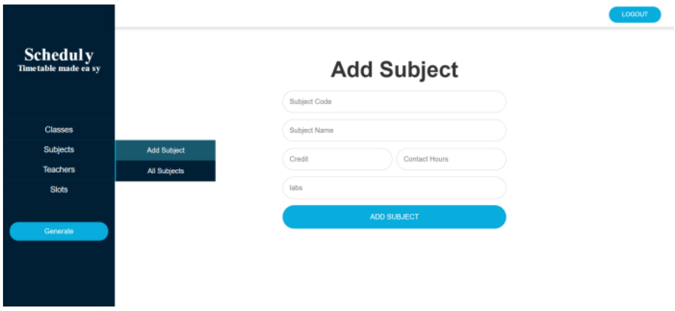
607 | 2022 - IT2

(Figure 6: View Classes Page Interface)

7. Add Subject

To add a new subject, you need to click on click on subjects in navigation panel and select add from submenu. On this page user will be able to add a new Subject. It has 5 input fields and a add button.

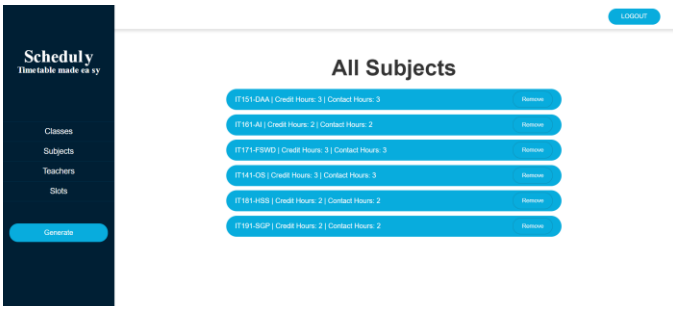
- Subject Name: This will be the name of subject to be added such as Operating System etc.
- Subject Code: This will be the code of subject to be added such as CS311 etc.
- Credit Hours: This will be the credit hours of a subject and it cannot be greater than 3.
- Contact Hours: This will be the contact hours of a subject and it cannot be greater than credit hours.
- Labs: Number of labs assigned to a subject.
- Add Subject Button: This button will save the records.



(Figure 7: Add Subject Page Interface)

8. View All Subjects

This will show all the subjects added by a user. You can go to this page by clicking on subjects in navigation panel on left and select all subjects from submenu. It has only one button Remove which will remove a subject.



IT151-DAA | Credit Hours: 3 | Contact Hours: 3

IT161-AI | Credit Hours: 2 | Contact Hours: 2

IT171-FSWD | Credit Hours: 3 | Contact Hours: 3

IT141-OS | Credit Hours: 3 | Contact Hours: 3

IT181-HSS | Credit Hours: 2 | Contact Hours: 2

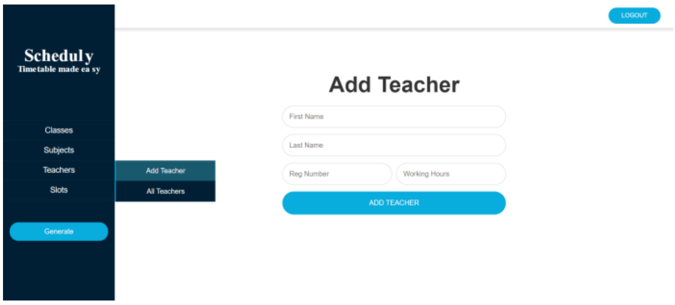
IT191-SGP | Credit Hours: 2 | Contact Hours: 2

(Figure 8: View Subject Page Interface)

9. Add Teacher

To add a new teacher, you need to click on click on teachers in navigation panel and select add from submenu. On this page user will be able to add a new teacher. It has 4 input fields and a add button.

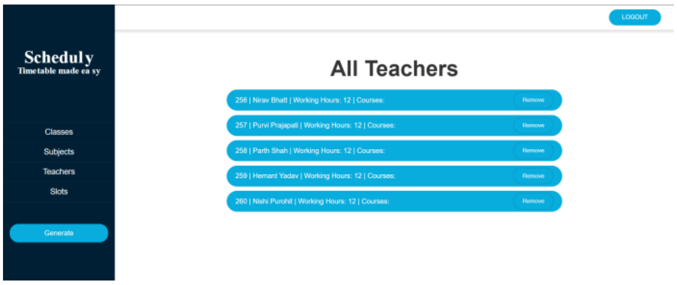
- First Name: This will be the first name of teacher to be added such as Samyan.
- Last Name: This will be the last name of teacher to be added such as Qayyum.
- Subject Code: This will be the code of subject to be added such as CS311 etc.
- Reg Number: This will be the Reg Number of a teacher and it cannot be greater than duplicated.
- Working Hours: This will be the working hours of a teacher in a week.
- Add Teacher Button: This button will save the records.



(Figure 9: Add Teacher Page Interface)

10. View All Teachers

This will show all the teachers added by a user. You can go to this page by clicking on teachers in navigation panel on left and select all teachers from submenu. It has only one button named as Remove which will remove a teacher.



256 | Nirav Bhatt | Working Hours: 12 | Courses:

257 | Purvi Prajapati | Working Hours: 12 | Courses:

258 | Parth Shah | Working Hours: 12 | Courses:

259 | Hemant Yadav | Working Hours: 12 | Courses:

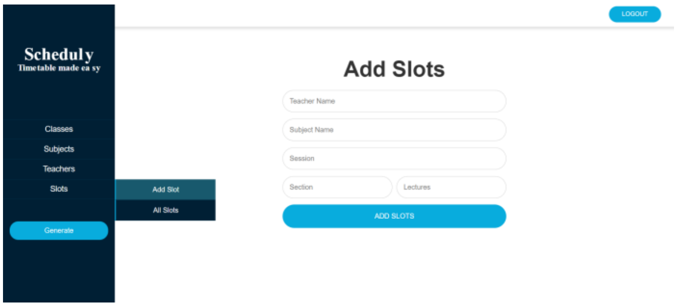
260 | Nishi Purohit | Working Hours: 12 | Courses:

(Figure 10: View Teachers Page Interface)

11. Add Subject

To add a new subject, you need to click on click on subjects in navigation panel and select add from submenu. On this page user will be able to add a new Subject. It has 5 input fields and a add button.

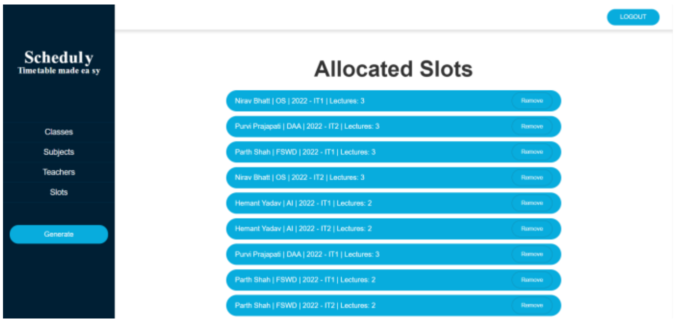
- Teacher Name: This will be the name of teacher to be assigned to a subject.
- Subject Name: This will be the name of subject to be assigned to the above teacher.
- Session: This input field will be the session name of the class such as 2018, 2020.
- Section: This input field will be the section of the class to which the teacher is being assigned.
- Add Slots Button: This button will save the records.



(Figure 11: Add Slots/Lectures Page Interface)

12. View All Slots

This will show all the slots added by a user. You can go to this page by clicking on slots in navigation panel on left and select all slots from submenu. It has only one button named as Remove which will remove an instance of a slot.

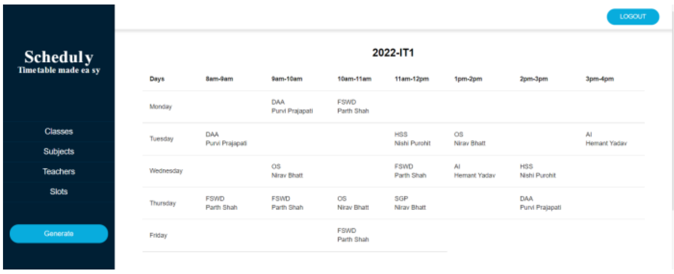


Nirav Bhatt | OS | 2022 - IT1 | Lectures: 3
Purvi Prajapati | DAA | 2022 - IT2 | Lectures: 3
Parth Shah | FSWD | 2022 - IT1 | Lectures: 3
Nirav Bhatt | OS | 2022 - IT2 | Lectures: 3
Hemant Yadav | AI | 2022 - IT1 | Lectures: 2
Hemant Yadav | AI | 2022 - IT2 | Lectures: 2
Purvi Prajapati | DAA | 2022 - IT1 | Lectures: 3
Parth Shah | FSWD | 2022 - IT1 | Lectures: 2
Parth Shah | FSWD | 2022 - IT2 | Lectures: 2

(Figure 12: View Slots/Lectures Page Interface)

13. Timetable

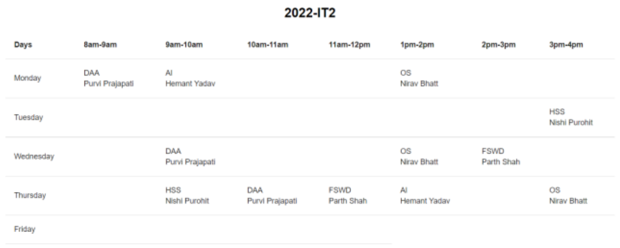
By clicking on the Generate Button from navigation panel on left, you will see the generated time table. Our output will be shown in tables on this page. Each table will represent time table of one class. Each row will represent a working day and each slot will represent a time slot. In each cell we will show the assigned teacher name and assigned subject name.



2022-IT1

Days	8am-9am	9am-10am	10am-11am	11am-12pm	1pm-2pm	2pm-3pm	3pm-4pm
Monday		DAA Purvi Prajapati	FSWD Parth Shah				
Tuesday	DAA Purvi Prajapati			HSS Nishi Purohit	OS Nirav Bhatt		AI Hemant Yadav
Wednesday		OS Nirav Bhatt		FSWD Parth Shah	AI Hemant Yadav	HSS Nishi Purohit	
Thursday	FSWD Parth Shah	FSWD Parth Shah	OS Nirav Bhatt	SGP Nirav Bhatt		DAA Purvi Prajapati	
Friday			FSWD Parth Shah				

(Figure 13: Output page UI (IT1))



2022-IT2

Days	8am-9am	9am-10am	10am-11am	11am-12pm	1pm-2pm	2pm-3pm	3pm-4pm
Monday	DAA Purvi Prajapati	AI Hemant Yadav			OS Nirav Bhatt		
Tuesday							HSS Nishi Purohit
Wednesday		DAA Purvi Prajapati			OS Nirav Bhatt	FSWD Parth Shah	
Thursday		HSS Nishi Purohit	DAA Purvi Prajapati	FSWD Parth Shah	AI Hemant Yadav		OS Nirav Bhatt
Friday							

(Figure 14: Output page UI (IT2))

VII. FUTURE SCOPE

Managing a large faculty and giving out assignments on time is a physically challenging undertaking. Therefore, our suggested system will aid in resolving this contradiction. As a result, we are able to create a schedule for any quantity of courses and semesters. With the help of this programme, you may make flexible pages that can be used with a variety of tools that are more effective and liberated. This system generates different schedules for each class, genre, and lab on its own. If another timeline is required, it can be created using a mix of various slots. You experience the pain of implementing a timeline even when the project minimises time utilisation. The project is being created in a way that ensures there are no time conflicts that prevent you from getting the desired timetable features. The project is a good time plan for departments and the rest of the college because it can be improved in the future. The process and tactics utilised for this project can be kept while making additional changes to accomplish this improvement.

VIII. CONCLUSION

It is a difficult task to manage many Faculty and assign subjects on time physically. Therefore, our proposed system will help overcome this inconsistency. Therefore, we can produce a timeline for any number of courses and semesters. This program will help you to create flexible pages so that in using such a program we can use a variety of tools that are more efficient and freer to use. Different timetables for each class, genres and labs are automatically generated by this system. A combination of different slots can be obtained to make another timeline as needed. The project minimizes time usage and you feel the pain in installing a timeline. The project is being created in a way that ensures there are no time conflicts that prevent you from getting the desired timetable features. The project is a good time plan for departments and the rest of the college because it can be improved in the future. The process and tactics utilised for this project can be kept while making additional changes to accomplish this improvement.

IX. References

[1] Bhaduri a “university timetable scheduling using genetic algorithm”. Advances in Recent Technologies inCommunication and Computing, 2009. ART Com '09. International Conference.A. R. Al-Ali, Imran A. Zualkernan, Assia Lasfer, Alaa Chreide, and Hadel Abu Ouda, “GRPS-Based Distributed Home-Monitoring Using Internet-Based Geographical Information System,” IEEE Trans. Consumer Electron., vol. 57, no.4, pp. 1688-1694, Nov. 2011.

[2] Dipti Srinevasan “automated time table generation using multiple context reasoning for university modules” Published in: evolutionary computation, 2002. ceca '02. proceedings of the 2002 congress on (volume:2).

[3] Practices in timetabling in higher education institutions. R. A. Oude Vrielink ,E. A. Jansen , E. W. Hans , J. van Hillegersberg.The paper was published on 27 October 2017. https://www.researchgate.net/publication/320675938_Practices_in_timetabling_in_higher_education_institutions_a_systematic_review.

[4] V. Abhinaya, K. Sahithi, K. Akaanksha, “Online Application of Automatic Timetable generator”, International Research Journal of Engineering and Technology, February 2019.

[5] AnirudhaNanda “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..

[6] Dipesh Mittal, Hiral Doshi, Mohammed Sunasra, Renuka Nagpure, “Automatic Timetable Generator using Genetic Algorithm”, International Journal of Advanced Research in Computer and Communication Engineering, February 2015.

[7] Carter M. W., Laborite G., “Recent developments in practical course timetabling”, Lecture Notes in Computer Science, Vol. LNCS1408, Springer-Verlag, 1998, pp. 3-19.

[8] Downland K. A., “Simulated annealing solutions for multi-objective scheduling and timetabling”, In Modern Heuristic Search Methods. Wiley. Chichester, England, 1996, pp. 155-166. European Journal of Operations Research, Vol. 54, 1991, pp. 39-47.

[9] A STUDY ON AUTOMATIC TIMETABLE GENERATOR. Parkavi A (M.S. Ramaiah Institute of Technology). The paper was published in May 2018 https://www.researchgate.net/publication/32626533.

[10] Schaef, A.,“A survey of automated timetabling”, Artificial Intelligence Review, No. 13, 1999, pp. 87-127.

[11] Saritha M, Pranav Kiran Vaze, Pradeep, Mahesh NR, “Automatic time table generator”, International Journal of Advanced Research in Computer Science and Software Engineering, May 2017.

[12] Yu, T. L., “Time-table scheduling using neural network algorithms”, IJCNN International Joint Conference on Neural Networks, 1990, Page(s): 279 -284 vol.1. 1.

[13] https://www.includehelp.com/algorithms/n-queens-problem-andsolution-using-backtracking-algorithm.aspx

[14] Anuja Chowdhary “TIME TABLE GENERATION SYSTEM”. Vol.3 Issue.2, February- 2014, pg.