

JSS MAHAVIDYAPEETA
JSS SCIENCE AND TECHNOLOGY UNIVERSITY
JSS TI CAMPUS, MYSORE-570 006
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING



***ADVANCED TIME TABLE GENRATOR BASED ON
PERSONAL PREFERENCES***

Report submission for the partial fulfillment of the course

JAVA PROGRAMMING (EC663)
EVENT II & IV

Submitted by

Roll No.	USN	NAME	Sem	Section	Marks Awarded
34	01JST17EC062	PANNAGA S	6	B	
18	01JST17EC034	G N RANJAN	6	B	

Under the guidance of

Supreetha M
Assistant Professor
Department of Electronics & Communication Engineering
JSSSTU, Mysore
2019-2020

Educate Elevate Enlighten

JSS Mahavidyapeetha
JSS Science And Technology University
(Established Under JSS Science and Technology University Act No. 43 of 2013)
(Formerly Known as SJCE)



DEPARTMENT OF ELECTRONICS AND COMMUNICATION

LTE-1 : Report presentation and evaluation (20 marks)

Group No :12	Sem : VI	Section :B	Period of semester: Even -2019-2020	Faculty :MS
--------------	----------	------------	-------------------------------------	-------------

Title of the Project : Advanced time table generator based on personal preferences

Sl. No.	USN	NAME	Problem Identification (Max: 10 Marks)	Presentation (Max: 5 Marks)	Viva (Max: 5 Marks)	Total (Max: 20 marks)
1	01JST17EC062	PANNAGA S				
2	01JST17EC034	G N RANJAN				

Date of Submission :	Signature of Faculty :
Remarks :	

Instructions for submission

1. Prepare 10 -20 pages of report
2. Font size: headings 16, sub headings 14, content 12 – Times new roman (font)
3. Direct download from website not allowed
4. Simulation task includes, introduction, writing codes, flow charts, results, graphs/plots, calculations, verification and conclusion.

CONTENTS

Chapter-1: INTRODUCTION

1.1 Motivation	1
1.2 Problem statement	2
1.3 Objectives	2

Chapter-2: LITERATURE SURVEY

2.1 Previous Research	3
2.2 Summary of Literature Review	3

Chapter-3: METHODOLOGY

3.1 Block Diagram	4
3.2 Requirements	7

Chapter-4: RESULTS AND INTERPRETATION

4.1 Code	8-9
4.2 Results	10-13
4.3 Anticipated bottleneck	14
4.4 Novelty of the project	15
4.5 Advantages and Applications	15
4.6 Conclusion	16
4.7 Future Scope	16

REFERENCES	16
-------------------	----

INTRODUCTION

Generating a timetable is a time consuming task and is done manually in more than 95% of universities. The process of designing, scheduling and executing a time table in universities is a hectic job for the time table committee. Although most of the college administrative work has been computerized, yet the lecture time table scheduling is still mostly done manually due to its technical difficulties. This process, that is the manual scheduling of lecture - timetable requires considerable time and effort to be put by the time table generating committee. Timetabling is known to be a non-polynomial complete problem i.e. there is no known efficient way to locate a solution. Also, the most striking characteristic of non-polynomial - complete problems is that, no best solution to them is known. Hence, in order to find a solution to a timetabling problem, a heuristic approach is chosen. This heuristic approach, therein, leads to a set of good solutions (but not necessarily the best solution).

1.1 MOTIVATION

In a general educational timetabling problem, a set of events (e.g. courses and labs, etc) are assigned into a certain number of time slots (time periods) subject to a set of constraints, which often makes the problem very difficult to solve in real-world circumstances . In fact, large-scale timetables such as university timetables may need many hours of work spent by qualified people or team in order to produce high quality timetables with optimal constraint satisfaction and optimization of timetable's objectives at the same time. Therefore in order to avoid all such mental stress on the time table committee, we propose an algorithm to generate a timetable that is housed inside a GUI , a graphical user interface in order to increase the dynamic and give freedom and ease of understanding by any member of the time table committee. While setting a timetable, importance is given to effective utilization of resources such as the classroom,the teacher, etc. This becomes a very tedious task which needs to be addressed at least once a year by every academic institute. Most institutes deal with this problem manually, i.e. a trial and error method is used to set a timetable.

1.2 PROBLEM STATEMENT

The need of the hour is an algorithm to generate a time table for a specific class, that takes in user defined constraints, here user being the time table setting committee or the faculty members, and based on the seniority of the lecturers, a time table is generated and is conveniently viewed on a graphical user interface (GUI).

1.3 OBJECTIVES

The main objectives of this JAVA program is as follows :

1. To create an algorithm that takes in multiple constraints such as :
 - a) The number of teaching hours per week per subject by each individual lecturer
 - b) The number of labs and the number of batches that the class should have
 - c) The time slot division , I.e., the teaching hours (6hrs) into morning and afternoon sessions.
2. To build a GUI platform that is driven by the algorithm that we design
3. To ease out the workload of the timetable setting committee
4. To provide more flexibility to the lecturers as a whole

CHAPTER 2

LITERATURE SURVEY

Research papers, articles and journals were studied and the literature survey was conducted to draw the conclusions.

2.1 PREVIOUS RESEARCH

[1] Automated College time table generator : Many colleges use manual way of preparing timetables with large number of students is very time consuming. This usually ends up with various courses clashing this may be either at same room or with same teachers having more than one course at a time. These are just due to common human errors which are very difficult to prevent in the processes.

[2] A study on time table generator : 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. These algorithms incorporate a numeral of strategy, aimed to improve the operativeness of the search operation.

2.2 SUMMARY OF LITERATURE REVIEW

Several time table generating hypothesis and heuristic approaches were done that could generate a time table according to certain constraints.

Table 2.2.1 Summary of literature review

Sl.No	Reference	Work done	Drawbacks
1	[1]	A hypothesis is proposed that takes in user defined constraints such as classrooms, teachers available, etc.	Does not talk anything regarding the lab allotment
2	[2]	Proposes to use genetic, heuristic algorithms to approach the problem.	Discusses the approach but not suitable for our university constraints.

CHAPTER 3

METHODOLOGY

The proposed solution basically works as described in the following sections by taking in various constraints from each of the faculty members and the time table setting committee as well.

3.1 BLOCK DIAGRAM

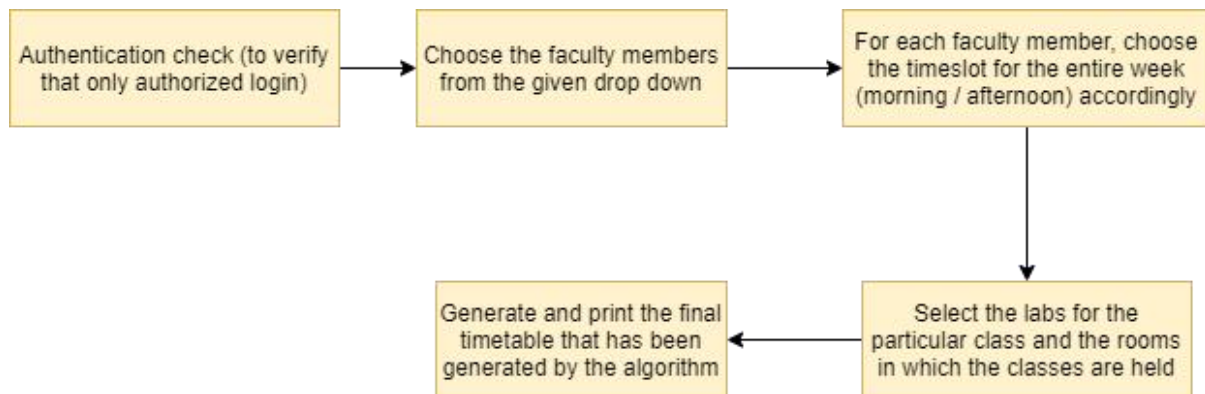


Fig 1.1 : Methodology

3.2 FLOW CHART AND REQUIREMENTS

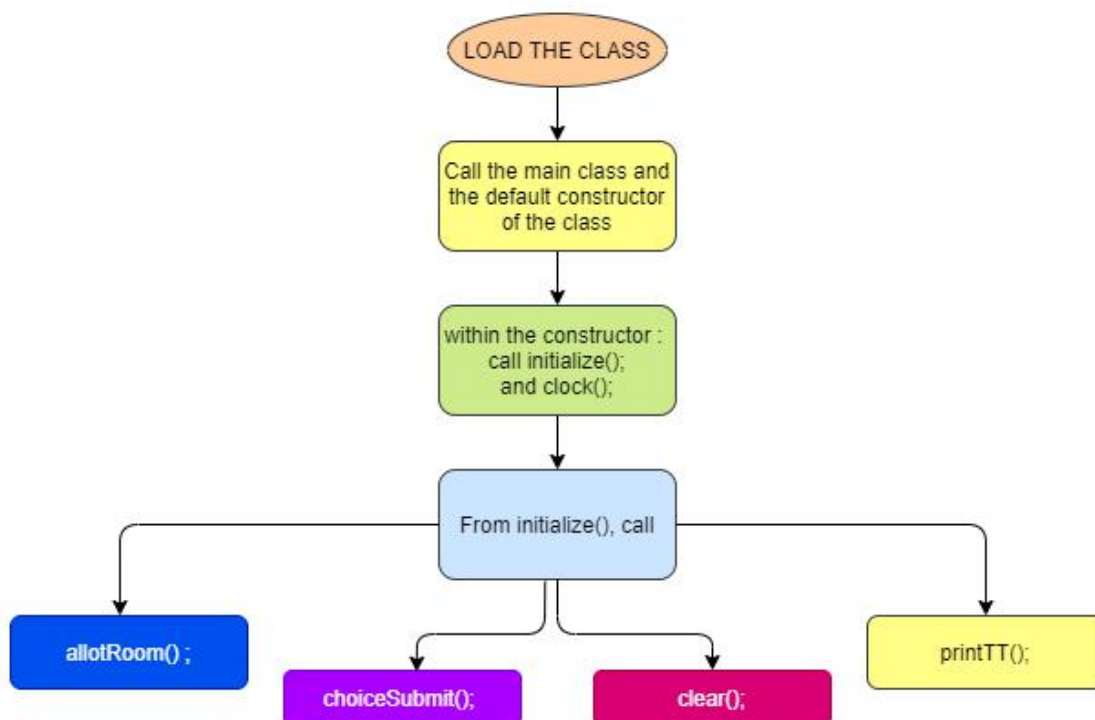


Fig 1.2 : flow chart depicting the overall run

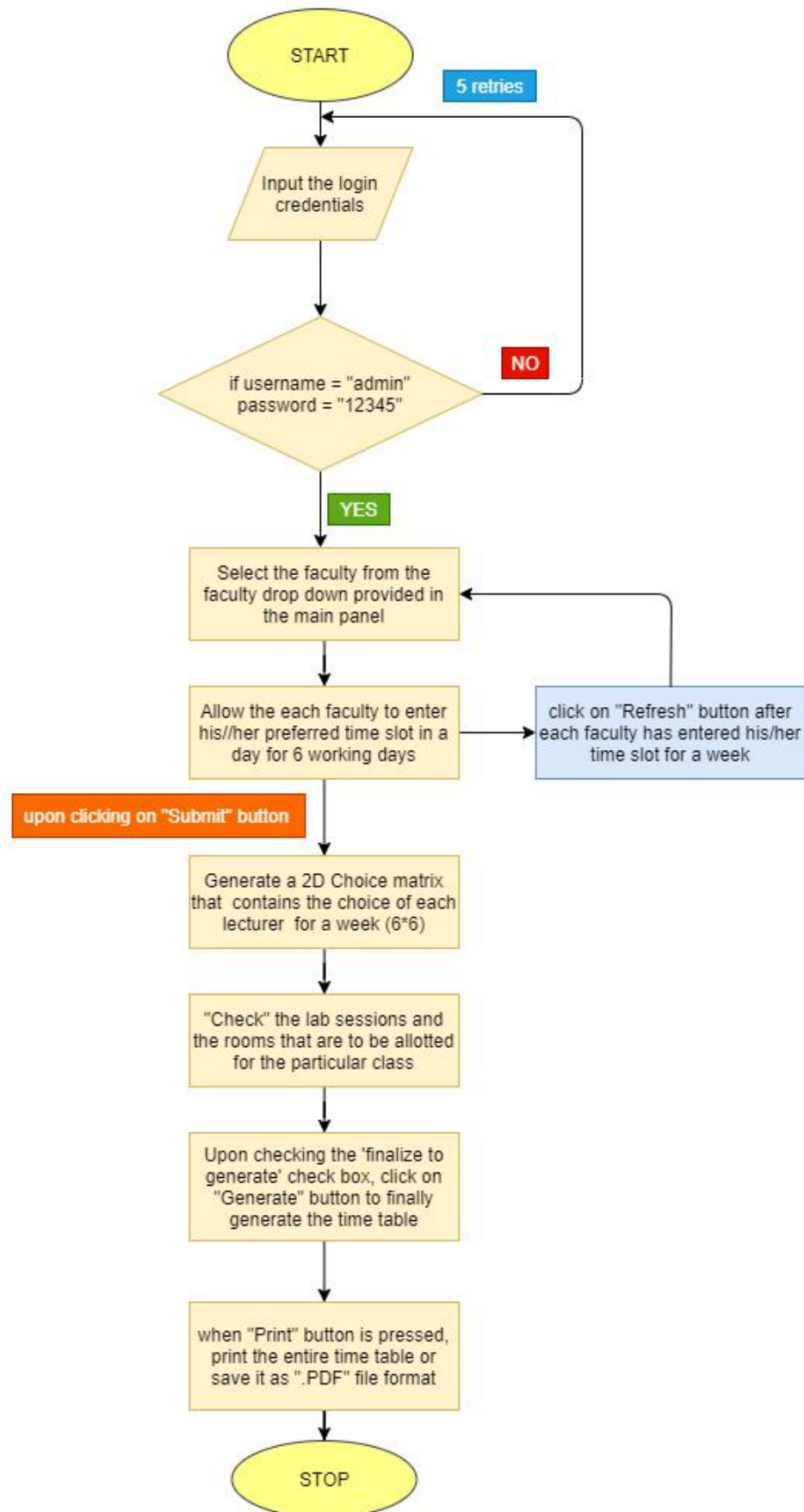


Fig 1.3 : Detailed flow chart of the algorithm

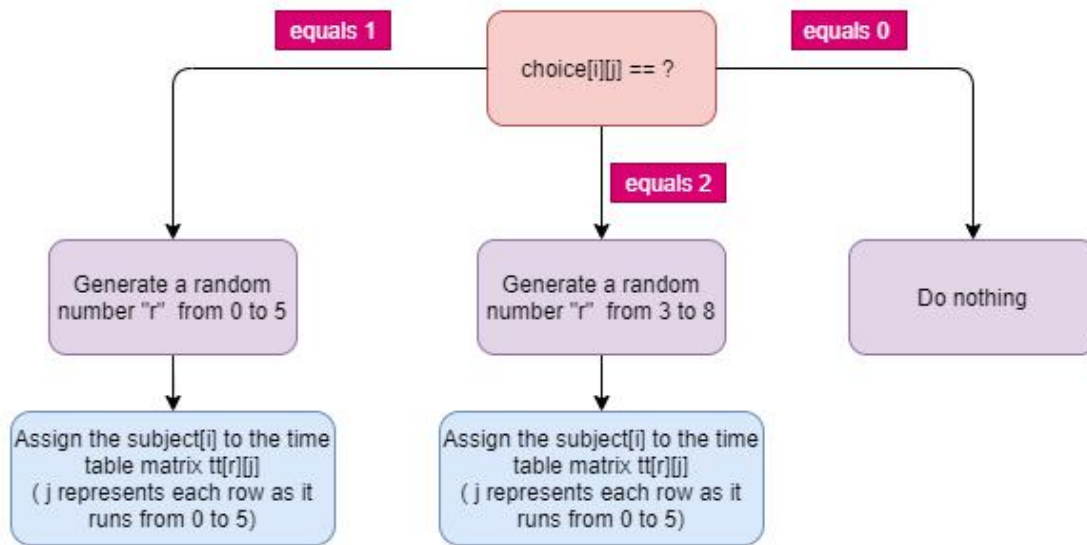
The primary requirement is the “javax.swing” package :

In general Swing is not thread safe. All Swing components and related classes, unless otherwise documented, must be accessed on the event dispatching thread. Typical Swing applications do processing in response to an event generated from a user gesture. For example, clicking on a JButton notifies all ActionListeners added to the JButton. As all events generated from a user gesture are dispatched on the event dispatching thread, most developers are not impacted by the restriction.

As the flow chart depicts, it first takes the user to the login window, where it asks for the user's , that is the authorized personnel the credentials for login, and only lets in to go to the next window if they've entered the correct credentials. Inside the main panel, the faculties are selected from the drop down provided, and 2 radio buttons for each day , for 6 working days have been put for the time slot selection that is either 7:30am - 1:30pm or 11:00am to 5:30pm. Then the user has to select any 4 of the given time slots. After each faculty's time slot selection, the user has to press “Ok” and “Refresh” button in the main panel to ensure that the slots are cleared and are ready to be entered for a new faculty member from the drop down provided. After all the 6 faculty members are entered with their choices, the user need to click on “Submit” button. When the “Ok” button is clicked, a choiceSubmit() function is called, where it inputs a 1D array of string type to store which time slot the faculty has booked. (7:30 am - 1:30 pm as slot 1 , and 11:00 am - 5:30 pm as Slot 2). The 1D array is unique to each faculty member, and finally, when “Submit” button is clicked, Generate() function is called which converts this 1D String array into a 2D string array where in the rows represents the each of the faculty members , and the columns represent the days of the week (6*6 matrix) , and the entries in this matrix would be “1, 2 or 0”, which represents slot 1, slot 2 , or no class for that day. This 2D string array is passed as one of the parameters to printTT() function, where it scans the entire choice matrix (6*6) matrix that is discussed earlier, and then checks if the choice[i][j] = 1 or 2 or 0.

Before we begin to assign the subjects based on the choice matrix, we assign the labs sessions first as it demands 3 consecutive slots. That is either from 7:30am - 10:30am or 11:00am to 1:30pm or 2:30pm -5:30pm, since once we begin assigning the subjects, there might be a possibility of not getting a continuous 3 slots to allot a lab session. Therefore, we assign the labs first. The lab assignment is also done completely randomly. The

following flow charts exactly convey the algorithm explained above.



tt [] [] is a 6 * 9 String array that holds the time table with all the subjects and labs offered, along with the personal preferences of each faculty member

Fig 1.3 : flow that represents choice matrix generation

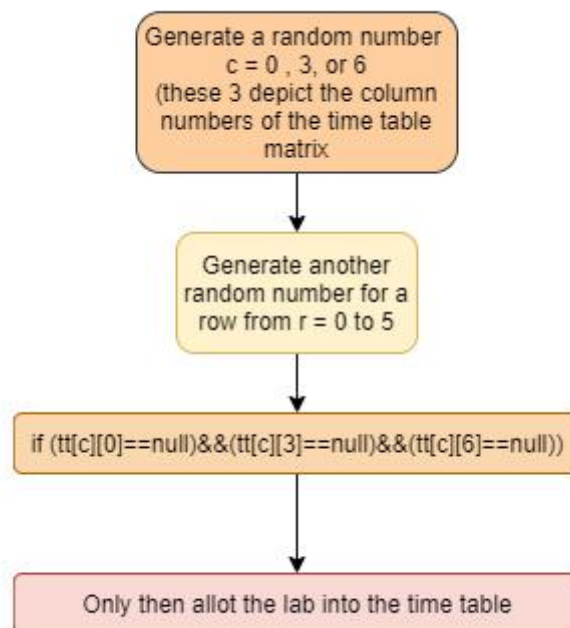


Fig 1.4 : flow that represents the lab session allotment

The algorithm is designed in such a way, such that the allotment of subjects into the timetable is based on the seniority order that is the 1st subject that is going to be allotted into the time table would be the senior most faculty, and would be having the most years of experience amongst the list of faculty members present.

CHAPTER 4

RESULTS AND INTERPRETATION

The algorithm was implemented and the desired output was obtained.

4.1 CODE

```
package timetablegui;

import java.awt.*;

/* Launch the application.
 *
 * public static void main(String[] args)
 */
/*
 * Function to display live clock on main_panel
 */
public void clock()

/**
 * Initialize the contents of the frame.
 */
private void initialize()

/*
 * Function name : allotRoom
 * Operation : It takes in a String variable, each time a room is selected from
               the main_panel of the GUI window, this function is called, and it checks which
               button was pressed based on the string value that is taken and prints it
               accordingly to the room allotted text area in the print_panel.
 */
protected void allotRoom(String string)

/*
 * Function name : choiceSubmit
 * Operation : When we press OK in the GUI window (main_panel), then each time an
               ok is pressed, then this function is called. Then, it takes in the faculty name
               that is selected, class choice, semester choice, section (A/B). So, it goes
               on assigning the choices in String format (mon1 tue2 ...) into individual 1D
               array for each specific teacher.*/
protected void choiceSubmit(String faculty_choice, String class_choice, String
                             sem_choice, String sec_choice)

/*
```

```

    * Function name : Generate

    * Operation : It returns a 2D string array called choice[][] (6*6 matrix), which
                  contains the choice of each faculty members that is assigned by the algorithm.

    */

    protected String[][] Generate()

    /*

    * Function name : clear

    * Operation : It clears all the radio button selection every time 'refresh' button

    */

    private void clear()

    /*

    * Function name : GetTimeSlot1

    * Operation : It returns a String array and in that string array, we will have
                  the days slot (individually), such as mon1, tue2, etc.

    */

    protected String[] GetTimeSlot1()

    /*

    * Function name : printTT

    * Operation : It takes in two parameters, (1) a 2D string array , (2) a 1D array
                  of list of faculties. It prints out the entire timetable based on the preferences
                  that is input from the faculties.

    */

    private void printTT(String x[][], String faculty[])

```

1	2	2	2	0	0
1	1	0	2	2	0
0	2	0	2	1	1
2	2	0	0	1	1
1	0	2	2	0	1
2	0	1	1	1	0

Fig 1.5 : generated choice matrix

The above figure 1.5 depicts a choice matrix that is generated by the Generate () function as explained above . The choice matrix has 6 rows and 6 columns, where each row represents each lecturer (in decreasing order of seniority) , and each column represents a day in a week, from Monday to Saturday.

4.2 RESULTS



The screenshot shows a window titled "Advanced time table generator || Event || Pannaga & Ranjan". The window has a yellow background. At the top center is the JSS Science and Technology University Mysuru logo. Below the logo, there are two input fields: "Username :" and "Password :". Below these fields are three buttons: "Reset", "Login", and "Exit". At the bottom, there is a section for developers' contact information.

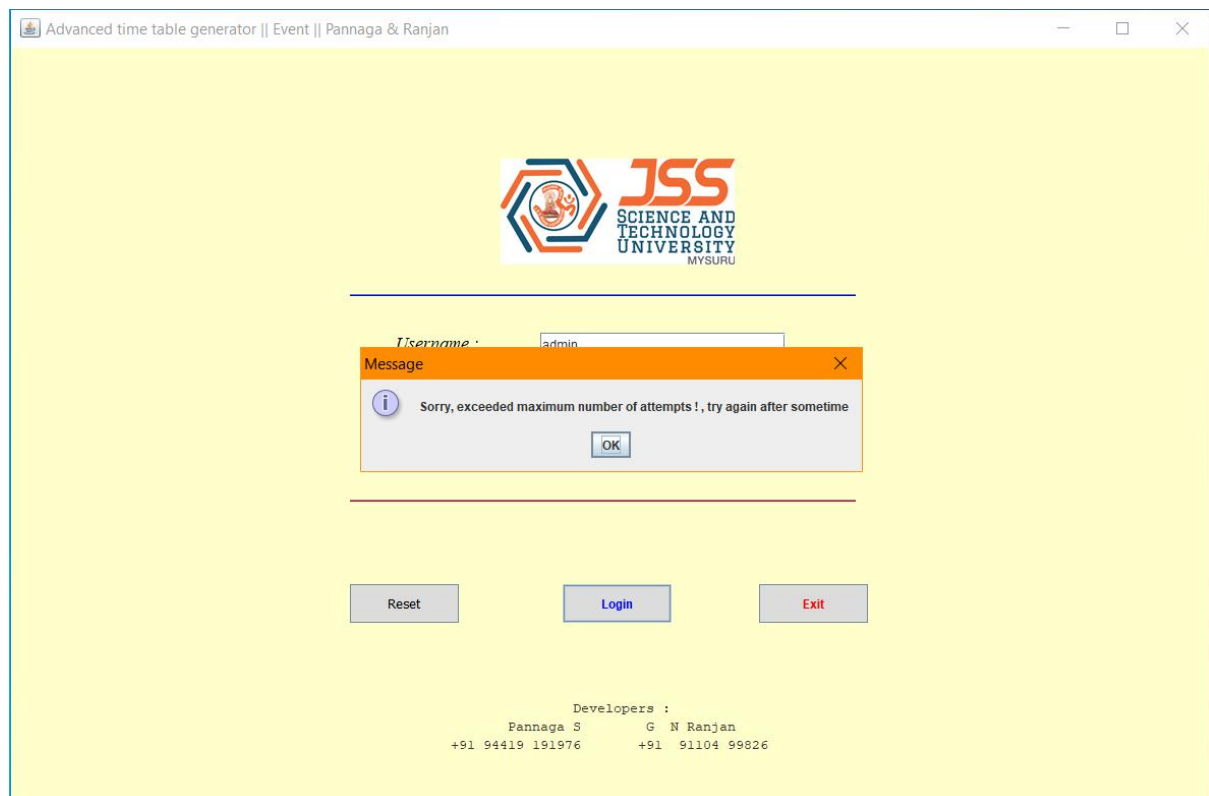
Username :

Password :

Reset Login Exit

Developers :
Pannaga S G N Ranjan
+91 94419 191976 +91 91104 99826

Fig 1.6 : Login window



The screenshot shows the same login window as Fig 1.6, but with an error message displayed. The "Username" field contains the text "admin". A message box is overlaid on the window, stating: "Message: Sorry, exceeded maximum number of attempts ! , try again after sometime". The message box has an "OK" button. The "Username" field is now disabled.

Username :

Message: Sorry, exceeded maximum number of attempts ! , try again after sometime

Reset Login Exit

Developers :
Pannaga S G N Ranjan
+91 94419 191976 +91 91104 99826

Fig 1.7: Output when the admin enters wrong credentials , be it username or password.

Advanced time table generator || Event || Pannaga & Ranjan

JSS Science and Technology University (Formerly SJCE)

Time table generator

Faculty:

No. of Subjects : (Max 7)

Class # : Semester : Section :

Room : ☐ AB104
☐ NPS001
☐ NPS002
☐ EV204
☐ EV205

Days

Days	Time slot selection
MONDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
TUESDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
WEDNESDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
THURSDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
FRIDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
SATURDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM

OK Refresh Submit

Labs :

☐ Design and Implementation Lab
☐ Advanced Communication lab
☐ CMOS VLSI Lab

Finalize to generate

Generate Exit

JSS MAHAVIDYAPEETHA

JSS
SCIENCE AND
TECHNOLOGY
UNIVERSITY
MYSURU

"Be an international leader in engineering education, research and application of knowledge to benefit society globally".

- "To synergistically develop high-quality manpower and continue to stay competitive in tomorrow's world".
- "To foster and maintain mutually beneficial partnerships with our alumni, industry, state and central governments through public services assistance and collaborative research".
- "To create empowered individuals with sense of identity".

21 : 40 : 22

25 / 3 / 28

Fig 1.8 : The main menu

Advanced time table generator || Event || Pannaga & Ranjan

JSS Science and Technology University (Formerly SJCE)

Time table generator

Faculty: MS

No. of Subjects : (Max 7)

Class # : Semester : Section : B

Room : ☐ AB104
☐ NPS001
☐ NPS002
☐ EV204
☐ EV205

Days

Days	Time slot selection
MONDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
TUESDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
WEDNESDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
THURSDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
FRIDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM
SATURDAY	<input type="radio"/> 7:30AM - 1:30PM <input type="radio"/> 11:00AM - 5:30PM

OK Refresh Submit

Labs :

☐ Design and Implementation Lab
☐ Advanced Communication lab
☐ CMOS VLSI Lab

Finalize to generate

Generate Exit

JSS MAHAVIDYAPEETHA

JSS
SCIENCE AND
TECHNOLOGY
UNIVERSITY
MYSURU

"Be an international leader in engineering education, research and application of knowledge to benefit society globally".

- "To synergistically develop high-quality manpower and continue to stay competitive in tomorrow's world".
- "To foster and maintain mutually beneficial partnerships with our alumni, industry, state and central governments through public services assistance and collaborative research".
- "To create empowered individuals with sense of identity".

21 : 41 : 16

25 / 3 / 28

Fig 1.9: Example of a faculty entering their preferable time slots

Advanced time table generator || Event || Pannaga & Ranjan

JSS Science and Technology University (Formerly SJCE)

Time table generator

Faculty: -
No. of Sub: SPK (Max 7)
Class #: 1 Semester: 6 Section: A

Room: AB104
NPS001
NPS002
EV204
EV205

Days
MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
SATURDAY

Time slot selection

7:30AM - 1:30PM

11:00AM - 5:30PM

Labs :
☒ Design and Implementation Lab
☒ Advanced Communication lab
☒ CMOS VLSI Lab
☐ Finalize to generate

JSS
 SCIENCE AND
 TECHNOLOGY
 UNIVERSITY
MYSURU

"Be an international leader in engineering education, research and application of knowledge to benefit society globally".
 • "To synergistically develop high-quality manpower and continue to stay competitive in tomorrow's world".
 • "To foster and maintain mutually beneficial partnerships with our alumni, industry, state and central governments through public services assistance and collaborative research".
 • "To create empowered individuals with sense of identity".

21 : 40 : 54

25 / 3 / 28

Fig 2.0: Depicts clearly the drop down menu for the list of faculty members

Advanced time table generator || Event || Pannaga & Ranjan

Educate Elevate Enlighten

JSS Mahavidyapeetha
JSS Science And Technology University
(Established Under JSS Science and Technology University Act No. 43 of 2013)
(Formerly SJCE)

Teaching Staff :
HOD : Dr. Shankaraiah N
1. Dr. U B Mahadevaswamy
2. Dr. M N Jayaram
3. Dr. Sudarshan Patilkulkarni
4. Dr. S G Gayathri Murthy
5. Dr. M G Veena
6. Dr. S B Rudraswamy
7. Prof. Renuka B S
8. Prof. Sujatha Kumari
9. Prof. Gayitri HM
10. Prof. M Supreetha
11. Prof. Anitha S Prasad
12. Prof. D R Pavithra
13. Prof. Eshwari A M
14. Prof. Kavya M K
15. Prof. Ramya
16. Prof. Vinay B R
17. Prof. Yashwanth S D
18. Prof. Halesh M R
19. Prof. Puneeth
20. Prof. N Shivkumar
21. Prof. M S Praveen

Non-teaching staff :
1. B T Halesh
2. K S Jayashree
3. M J Philip
4. Umashankar
5. Sidappa M S
6. Poornima M
7. Satheesha M
8. Yogesh M

Rooms allotted :
NPS002
NPS001
Labs allotted :
Design and Implementation lab
Advanced Communication lab
CMOS VLSI lab

JSS Science and Technology University (Formerly SJCE)

	7:30am	8:30am	9:30am	11:00am	11:50am	12:40pm	2:30pm	3:30pm	4:30pm
Mon	EC630	EC653	EC663	EC663	EC610	EC620	LabS4	LabS4	LabS4
Tue	FREE	FREE	EC630	EC610	EC640	EC653	LabS2	LabS2	LabS2
Wed	FREE	FREE	FREE	EC663	EC620	EC653	FREE	FREE	FREE
Thu	EC663	FREE	FREE	EC620	EC630	FREE	FREE	EC640	EC653
Fri	LabS1	LabS1	LabS1	EC640	FREE	EC610	FREE	EC630	FREE
Sat	EC620	EC640	EC610	LabS3	LabS3	LabS3	NOCLS	NOCLS	NOCLS

Fig 2.1: The print panel that prints out the entire time table and all the necessary information such as rooms allotted , and the labs allotted including the lab session division

Note that the time table is printing out the **ONLY** starting time of each class. This is done to take care of the alignment. Each class is of 1 hour duration from 7:30am -10:30am slot and 2:30pm - 5:30pm slot, and 50 min duration from 11:00am - 1:30pm time slot. This is done in accordance to our college obligations.

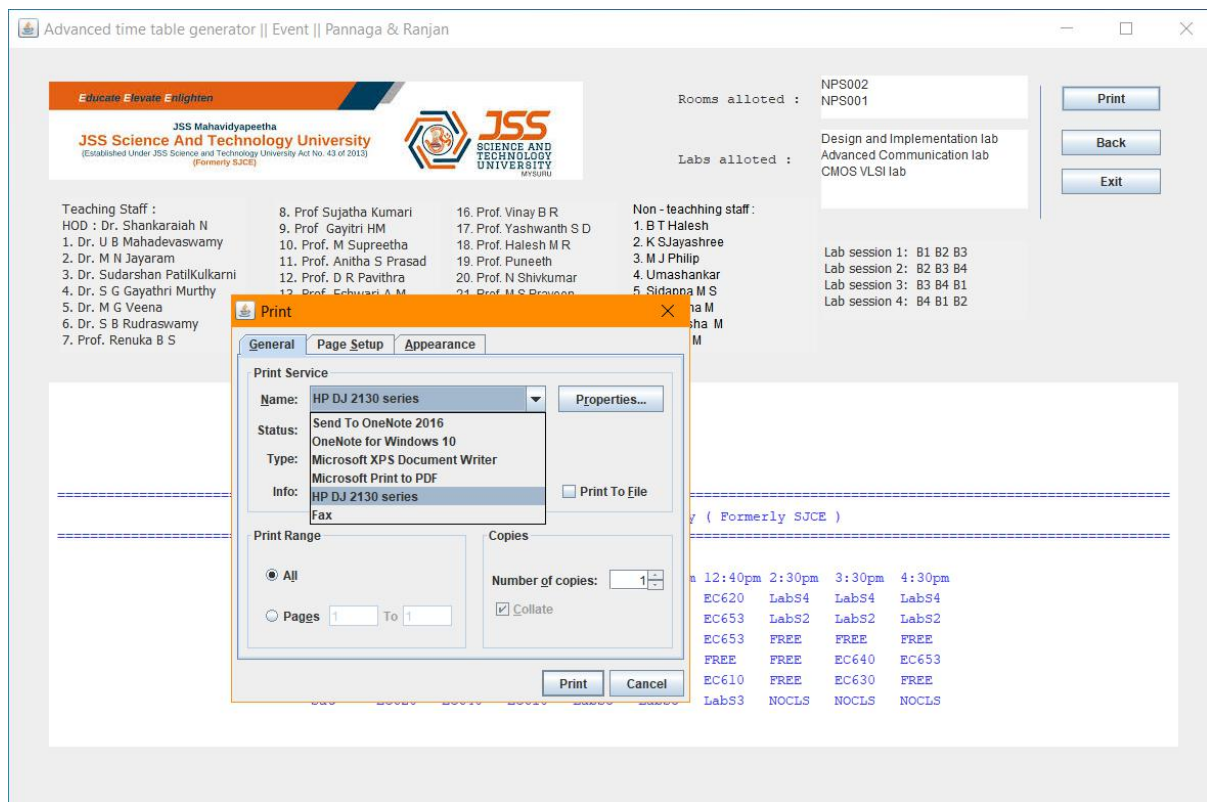


Fig 2.2 : Shows the print options that are available from the drop down menu, upon clicking on the “Print” button

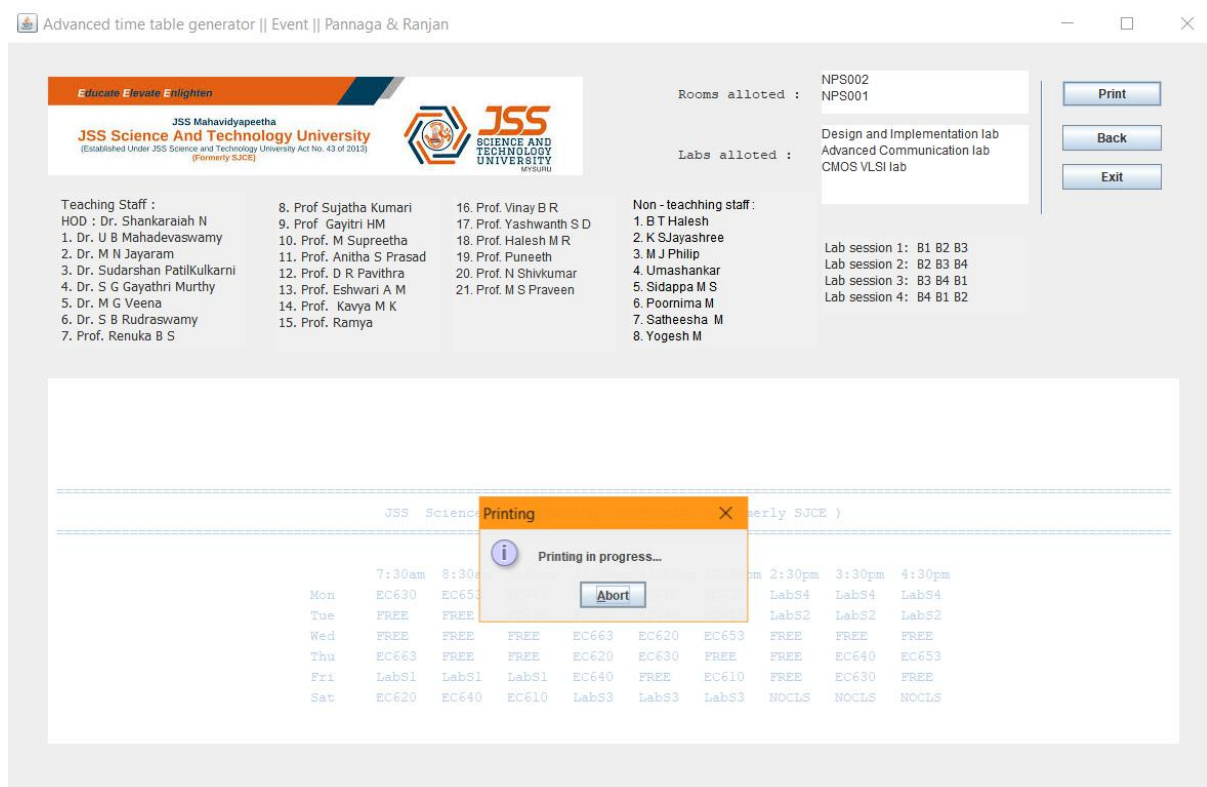


Fig 2.3 : Depicts the printing process

4.3 ANTICIPATED BOTTLENECK

There were a quite a few pit falls that we may face in this algorithm, such as (1) when a lecturer wishes to enter more than 4 hours of teaching slots per week, (2) when the admin fails to enter the correct credentials into the login window, (3) or, in a particular case, for example, say 4 lecturers have been opted for time slot 1, and by co incidence, a lab has been allotted in the time slot 11:00am to 1:30pm time frame. So, the remaining slots that are free to be able to accommodate the necessary lecturers would be only 3, that is from 7:30am to 10:30am, but there are 4 lecturers that are wanting this particular time slot.

We over come each of the problems discussed above as follows :

(1) We initialize a count variable initially assigned to 4, and after each click on radio button (any) the count is decremented by 1, and upon clicking “Refresh” button, the count is again reassigned to 4.

(2) The second problem is also overcome by using a try variable for type int, that is initialized to 5, and then we run a separate thread in order to create a 5 second delay, only after the user has failed to enter even after 5 times. In order to create a delay of 5 seconds, we call the Thread.sleep class with try - catch blocks.

(3) The third problem is overcome by again initializing a separate count variable that is initialized to 10. Then this is decremented after the command enters the if - else block to check whether the choice[i][j] is equal to 1 or 2 or 0. When this count is 0, it enters another while loop where it randomly generates two numbers in the range $r1 = 0$ to 5 and $r2 = 0$ to 8, and checks if `tt[r1][r2]` equal to null, and assigns the subject[i] to that particular time slot. This is done so that all the lecturer’s teaching hours is 4 in our university for a week. Therefore, this ensures that all the faculty members get 4 time slots per week as per they desire.

Also, note that the time table generator algorithm is designed in such a way that there will be no class that will be assigned on Saturdays from 1:30pm to 5:30pm. This condition is put to adhere to our university’s condition.

4.4 NOVELTY OF THE PROPOSED WORK

The graphical user interface developed is unique to our university, and is based on our lecturer's teaching hour flexibility and the constraints they have as per the higher management. Also, the GUI program gives apt results based on the number of teaching hours per faculty per week, and the lab sessions assigned including the number of batches for a particular batch. The generalized way of assigning the batches and the labs is also taken care off , such as if there are 3 labs and 4 batches, the maximum of the 2 numbers are to be considered for knowing the number of lab sessions per week. As per our university, it is 4 batches and hence there are 4 lab sessions per week, so that all batches get equal opportunity to have the lab sessions as per the course designed.

Also, the flexibility of selecting the time slots 7:30am to 1:30pm or, 11:00am to 5:30pm is given to each individual faculty. Since, the algorithm is designed in such a way that the assignment of the subjects into the time slots of the time table is done according to the seniority of the lecturer's, even in an ambiguity situation where a junior faculty member could not be accommodated in his/ her timeslot , they are ensured of getting 4 slots of teaching hours per week, and no 2 consecutive classes per day or in a day is also taken care off which is essential for students to take in.

4.5 ADVANTAGES AND APPLICATIONS

The advantages of this program is to largely decrease the amount of time and effort put by the time table setting committee into making a timetable that doesn't clash , accommodate everyone's schedule, relief from mental stress and strain.

The second advantage is that we are giving the faculty members to choose the time slot according to their convinience.

Applications of this program would be in any university, college, schools, but some of the constraints have to be altered according to the particular university, college, or school, where it is put in use.

4.6 CONCLUSION

The time table generation algorithm was developed that is specific to our university, and our department, Electronics and communications. The time slots that are divided and are accessed as radio buttons as shown in fig 1.8 - fig 2.0, are as per the university's rules and regulations that is put up by the management. Also, the algorithm worked efficiently as it was designed to allot the slots for each lecturer individually according to their personal preferences of time slots and also taking into the consideration of their seniority, that is the seniority of the lecturers. Then, from fig 2.1 - fig 2.3, we can clearly see that the time table that is spat out by the time table generating algorithm. Clearly we can see that the number of teaching hours are 4 per week and we have 4 lab sessions as well.

4.7 FUTURE SCOPE

The future scope for this project would be implementing the algorithm with machine learning algorithms and techniques so that a more efficient way of generating the time table can be obtained if not the classical heuristic approach which we have nowadays in manual time table framing. Also, this project can be expanded for different branches, and different set of constraints based on the university's schedule rules, and regulations and obligations from the lecturer side.

REFERENCES

- [1] International Journal of Scientific & Engineering Research Volume 9, Issue 4, April-2018
ISSN 2229-5518 IJSER © 2018 : Automated college timetable generator
- [2] A, Parkavi. (2018). A STUDY ON AUTOMATIC TIMETABLE GENERATOR.
International Journal of Innovative Research & Growth. May.
- [3] M, Saritha & Vaze, Pranav & Pradeep, & R, Mahesh. (2017). Automatic Time Table Generator. International Journal of Advanced Research in Computer Science and Software Engineering. 7. 204-211. 10.23956/ijarcsse/SV7I5/0234.
- [4] R., Mayuri & C., Sunil & N., Sunita & R., Pushkar & Kumavat, Sanjay. (2015). A Novel Approach for Automatic Timetable Generation. International Journal of Computer Applications. 127. 26-30. 10.5120/ijca2015906512.