

# Timetable Management System



## CEP Report

By

NAME	RegistrationNumber
Sheryar Ahmed	CIIT/ FA21-BCE-042 /ATD
Hasnain Hafeez	CIIT/ FA21-BCE-036 /ATD
Muhammad Hassan	CIIT/ FA21-BCE-051 /ATD

For the course

Object Oriented Programming

Semester Spring 2023

Supervised by:

Dr. Faisal Mehmood

Department of Electrical & Computer Engineering

COMSATS University Islamabad – Abbottabad Campus

## DECLARATION

We Hasnain Hafeez (CIIT/FA21-BCE-036/ATD), Sheryar Ahmed (CIIT/FA21-BCE-042/ATD), and Muhammad Hassan (CIIT/FA21-BCE-051/ATD) hereby declare that we have produced the work presented in this report, during the scheduled period of study. We also declare that we have not taken any material from any source except referred to wherever due. If a violation of rules has occurred in this report, we shall be liable to punishable action.

Date: \_\_\_\_\_

\_\_\_\_\_  
Hasnain Hafeez  
(CIIT/FA21-BCE-036/ATD)

\_\_\_\_\_  
Sheryar Ahmed  
(CIIT/FA21-BCE-042/ATD)

\_\_\_\_\_  
Muhammad Hassan  
(CIIT/FA21-BCE-051/ATD)

## **ABSTRACT**

In this project, the problem was that we had to generate timetables for any given number of sections, faculty, number of lectures per week without any clash if possible. The methodology we have adopted is that we have created 2D arrays for every section and we fill one faculty member at a time while also checking the validity of the indexes we are randomly generating for them by keeping a record of every valid row/column index. If the indexes are invalid, new ones shall be generated. This shall continue at most 10 million times and then the program will start from scratch. This shall continue at most 10 times. If the clash cannot be resolved, it means that the timetables cannot be created with the given data under the conditions set in the program which determine the validity of the timetables. It has user friendly graphical user interface which is in its most basic form at present. It has three interfaces i.e., Administration, Teachers, Students. Administration has access to both teacher and student timetables as well as a button to generate new timetables. Teachers can view their timetables which have every section they have a lecture in. Students can view their timetables which have every teacher they have a lecture with. The program's output is always one hundred percent accurate according to the test results. Since the program is in its initial stage, many functionalities can be added in it in the future to make it more versatile and flexible.

## TABLE OF CONTENTS

<b>1. Introduction.....</b>	<b>1</b>
<b>2. Literature Survey.....</b>	<b>2</b>
<b>3. Proposed Methodology .....</b>	<b>2</b>
<b>4 Simulation Results .....</b>	<b>5</b>
<b>5. Conclusions.....</b>	<b>8</b>
<b>6. References .....</b>	<b>8</b>

## LIST OF FIGURES

<b>Figure 3.1: Block diagram of the Program.....</b>	<b>3</b>
<b>Figure 3.21 Hierarchy of Timetable Generation.....</b>	<b>3</b>
<b>Figure 3.22: Hierarchy of GUI Class .....</b>	<b>3</b>
<b>Figure 3.3: Combo box Action Listener Flow chart .....</b>	<b>4</b>
<b>Figure 4.1: Showing the login page. ....</b>	<b>5</b>
<b>Figure 4.2: Showing Administration Interface displaying teachers' timetable. ....</b>	<b>5</b>
<b>Figure 4.3: Showing Administration Interface displaying student's timetable. ....</b>	<b>6</b>
<b>Figure 4.4: Showing Teachers Interface .....</b>	<b>6</b>
<b>Figure 4.5: Showing Students Interface.....</b>	<b>7</b>

## LIST OF ABBREVIATIONS

GUI	Graphical User Interface
-----	--------------------------

# **1. Introduction**

Timetable Management System is a program that generates and displays timetables with a GUI that has separate interfaces for Administration, faculty, and students. Administration has access to both teacher and student timetables as well as a button to generate new timetables. Teachers can view their timetables which have every section they have a lecture in. Students can view their timetables which have every teacher they have a lecture with. The program's output is always one hundred percent accurate and that can be verified by testing, and we already have. This project seems to be easy by the description, but it has a lot of complexities at the backend that were handled very carefully. The GUI is very simplistic in this version and has limited functionality but can be improved in future versions.

## **1.1. Objectives**

The main objective of this project is to create a working program that generates and displays timetables for any given number of sections, faculty, number of lectures per week without any clash if possible. There are no constraints for room availability, faculty availability, room capacity etc.

## **1.2. Features**

Features of this program are that you can generate random timetables. If you do not like the ones generated, you can generate new ones in a matter of seconds. Students and teachers can view their respective timetables while administration can view all timetables as well as generate new ones.

## 2. Literature Survey

### 2.1. Greedy Algorithm

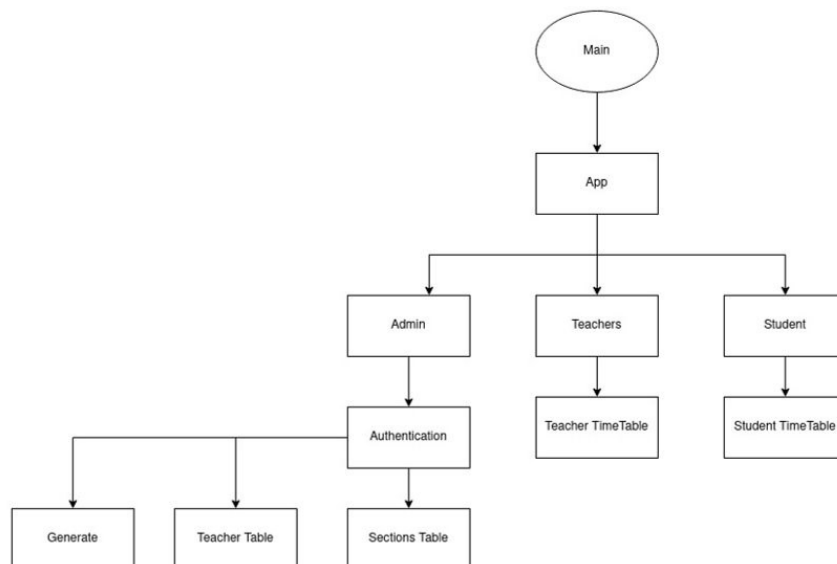
This algorithm was most suitable for this project since we had no constraints given to us by our supervisor. A greedy algorithm is an approach that makes locally optimal choices at each step to find a solution. It doesn't consider the global picture, which might lead to suboptimal outcomes in some cases. However, we developed our own logic because we were unable to fully understand the implementation of this algorithm.

### 2.2. Constraint Satisfaction Problem

This algorithm was studied for a better understanding of the concepts of programming. However, it needed more time than we had for this project and a better level of understanding than we have as undergraduate students. A constraint satisfaction problem involves finding values for variables while satisfying a set of constraints. The goal is to find a solution that meets all constraints simultaneously.

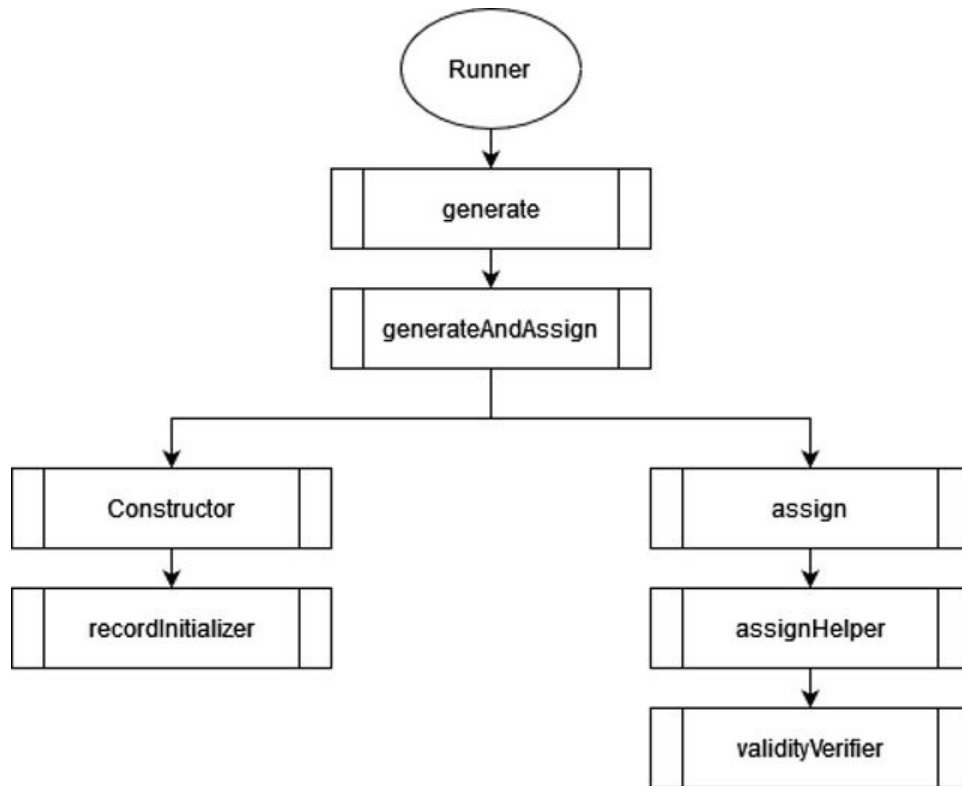
## 3. Proposed Methodology

### 3.1 Block diagram

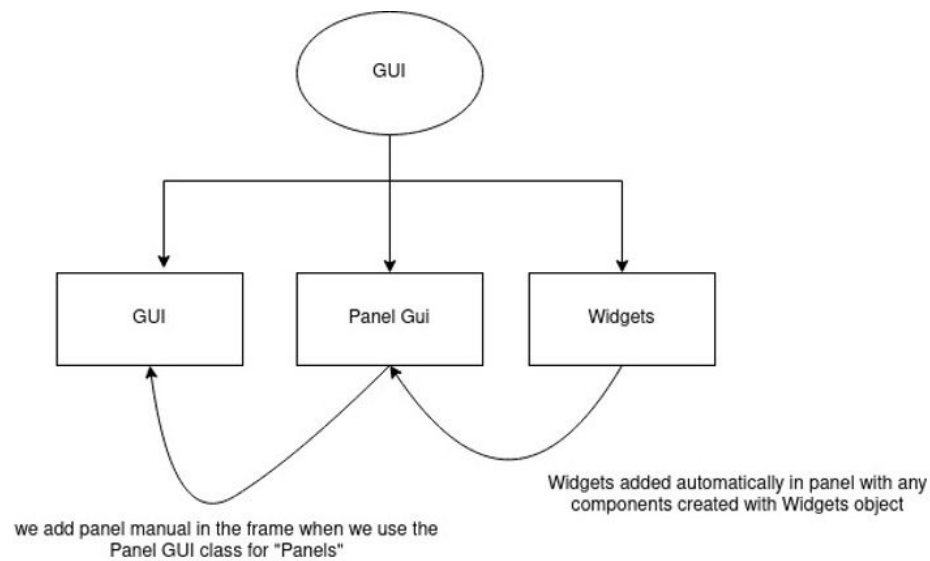


**Figure 3.1: Block diagram of the Program**

### 3.2. Hierarchy Chart



**Figure 3.21 Hierarchy of Timetable Generation**



**Figure 3.22: Hierarchy of GUI Class**

### 3.3 Flowchart

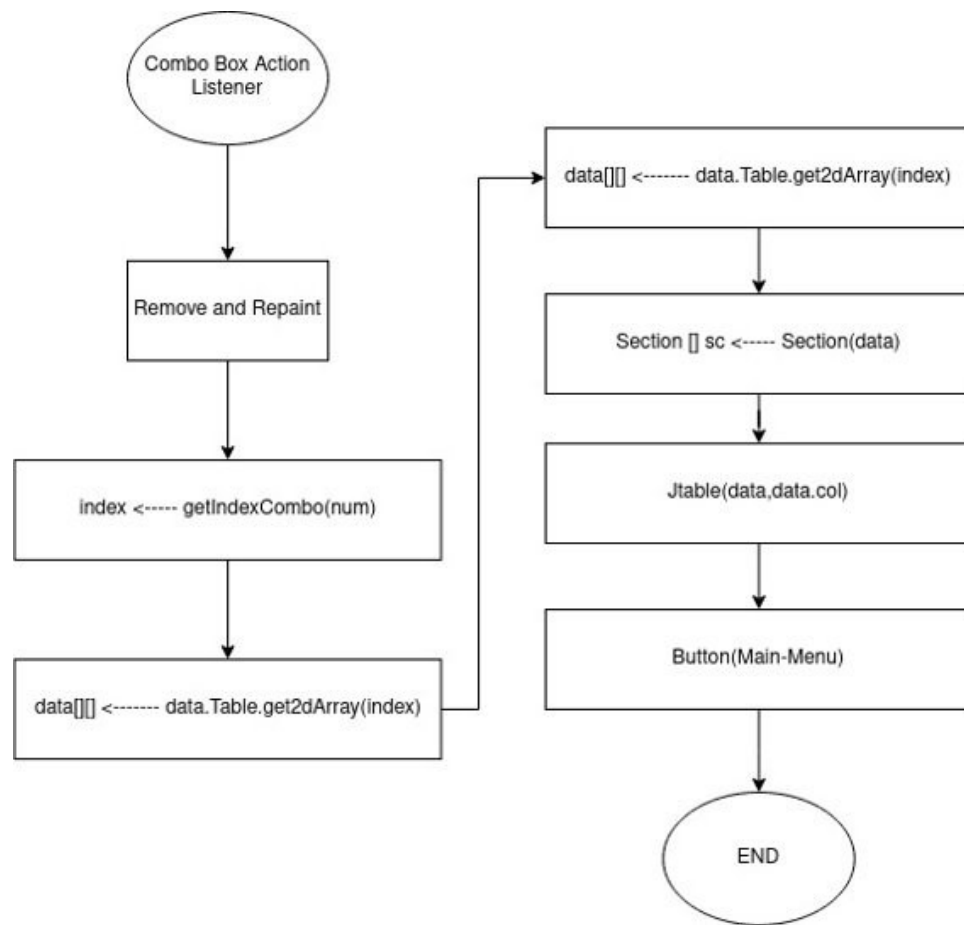


Figure 3.3: Combo box Action Listener Flow chart



## 4 Simulation Results

### 4.1 Software simulation results

Timetable Management System

Comsats University Islamabad

CUONLINE

Student Console

Administration Teachers Student

Enter your password:

.....

Confirm

Figure 4.1: Showing the login page.

Administration

Faculty: Dr. Alam Zaib Sections: BCE-4A Generate

Days	8:00 - 9:30	9:40 - 11:10	11:20 - 12:50	13:20 - 15:10	15:20 - 16:50	17:00 - 18:30
Monday			BCE-4D	BCE-4E		BCE-4C
Tuesday	BCE-4B			BCE-4D		
Wednesday		BCE-4B		BCE-4A		
Thursday	BCE-4E		BCE-4C		BCE-4B	BCE-4A
Friday		BCE-4E	BCE-4A	BCE-4C	BCE-4D	
Saturday						

Main Menu

Figure 4.2: Showing Administration Interface displaying teachers' timetable.

Administration

Faculty	Dr. Alam Zaib		Sections	BCE-4A		Generate
Days	8:00 - 9:30	9:40 - 11:10	11:20 - 12:50	13:20 - 15:10	15:20 - 16:50	17:00 - 18:30
Monday			Dr.Imdad Khan		Zulfiqar Khattak	Ayaz Ali Shah
Tuesday	Zulfiqar Khattak	Dr.Imdad Khan		Ayaz Ali Shah		
Wednesday	Dr.Faisal Mehmood	Ayaz Ali Shah		Dr. Alam Zaib		
Thursday		Zulfiqar Khattak		Dr.Faisal Mehmood		Dr. Alam Zaib
Friday		Dr.Faisal Mehmood	Dr. Alam Zaib		Dr.Imdad Khan	
Saturday						

Main Menu

Figure 4.3: Showing Administration Interface displaying student's timetable.

Teacher

Select the Section

Dr.Imdad Khan

Days	8:00 - 9:30	9:40 - 11:10	11:20 - 12:50	13:20 - 15:10	15:20 - 16:50	17:00 - 18:30
Monday	BCE-4B		BCE-4A		BCE-4D	
Tuesday		BCE-4A	BCE-4D	BCE-4C	BCE-4E	
Wednesday		BCE-4C				BCE-4B
Thursday			BCE-4B	BCE-4E		
Friday	BCE-4E		BCE-4C	BCE-4D	BCE-4A	
Saturday						

Main Menu

Figure 4.4: Showing Teachers Interface

Days	8:00 - 9:30	9:40 - 11:10	11:20 - 12:50	13:20 - 15:10	15:20 - 16:50	17:00 - 18:30
Monday				Dr. Alam Zaib		Dr.Faisal Mehmood
Tuesday	Dr.Faisal Mehmood		Ayaz Ali Shah		Dr.Imdad Khan	
Wednesday	Ayaz Ali Shah				Zulfiqar Khattak	
Thursday	Dr. Alam Zaib			Dr.Imdad Khan	Ayaz Ali Shah	Zulfiqar Khattak
Friday	Dr.Imdad Khan	Dr. Alam Zaib	Zulfiqar Khattak	Dr.Faisal Mehmood		
Saturday						

Main Menu

Figure 4.5: Showing Students Interface

## 4.2. Design / simulation parameters

In this simulation, the parameters are as follows:

- 5 sections
- 3 lectures per teacher for every section
- 5 teachers i.e., Dr. Alam Zaib, Dr. Faisal Mehmood, Dr. Imdad Khan, Ayaz Ali Shah, Zulfiqar Khattak
- Department plus year is BCE-4

Using these parameters, the program has generated timetables for BCE-4A to BCE-4E.

## 4.3. Discussions

There are many different combinations of timetables that can be generated. However, since there are 6 timeslots and Saturday is considered a holiday, leaving 5 days per week, increasing the number of sections, faculty or number of lectures can decrease the chances of timetables being generated.

## 5. Conclusions

This project was time consuming in the sense that we had to develop ideas as well as the logic to implement them. The major task was that of the timetable generation with 100% accuracy and making it resource efficient as possible according to our knowledge. Creating a simple yet interactive Graphical User Interface was also a huge challenge with different interfaces such as Administration, Teachers, and Students. With teamwork and guidance of our supervisor, we have achieved these tasks. This is a basic program, and we can add a lot more functionalities to it in the future.

## 6. References

- [1]. Sedgewick, R. (1998). *Algorithms in C++, Parts 1-4: Fundamentals, Data Structure, Sorting, Searching, Third Edition*. 3rd ed. Addison-Wesley.
- [2]. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, (2022). *Introduction to Algorithms, Fourth Edition*.
- [3]. javatpoint.com. (Accessed June 12, 2023). *Java Swing Tutorial*. JavaTpoint. <https://www.javatpoint.com/java-swing>
- [4]. Stack Overflow. (2012, Dec 8). *How to autofill an array in Java*. In: Stack Overflow. <https://stackoverflow.com/questions/13777108/how-to-autofill-an-array-in-java/13777120#13777120>