



AUTOMATED TIMETABLE GENERATOR

EPICS PROJECT REPORT submitted in partial fulfillment of the requirements

Submitted by

MORAMPUDI DEVI SOWMYA 188W1A1232

TADAPANENI PRATHYUSHA 188W1A1251

ABDUL KHALID HUSSAIN 188W1A1202

INJAM SRI LATHA 188W1A1223

Under the Guidance of

Mrs. N. Praveena

Assistant Professor

For the award of the degree

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY



DEPARTMENT OF INFORMATION TECHNOLOGY

V R SIDDHARTHA ENGINEERING COLLEGE

(AUTONOMOUS - AFFILIATED TO JNTU-K, KAKINADA)

Approved by AICTE & Accredited by NBA

KANURU, VIJAYAWADA-7

ACADEMIC YEAR

(2020-21)

V.R.SIDDHARTHA ENGINEERING COLLEGE

(Affiliated to JNTUK: Kakinada, Approved by AICTE, Autonomous)

(An ISO certified and NBA accredited institution)

Kanuru, Vijayawada – 520007



CERTIFICATE

This is to certify that this project report titled “**AUTOMATED TIMETABLE GENERATOR**” is a bonafide record of work done by **M. DEVI SOWMYA (188W1A1232), T. PRATHYUSHA (188W1A1251), A. KHALID HUSSAIN (188W1A1202) and I. SRI LATHA (188W1A1223)** under my guidance and supervision is submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Information Technology, **V.R. Siddhartha Engineering College** (Autonomous under JNTUK) during the year **2020-2021**.

(Mrs. N. Praveena)

Assistant Professor

Dept. of Information Technology

(Dr. M. Suneetha)

Professor & Head

Dept. of Information Technology

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

First and foremost, I sincerely salute our esteemed institution **V.R SIDDHARTHA ENGINEERING COLLEGE** for giving me this opportunity for fulfilling my EPICS project. I am grateful to our principal **Dr. A.V.RATNA PRASAD**, for his encouragement and support all through the way of my project.

On the submission of this Project report, I would like to extend my honour to **Dr. M. Suneetha**, Head of the Department, IT for her constant motivation and support during the course of my work.

I feel glad to express my deep sense of gratefulness to my project guide **N.Praveena, Assistant Professor** for **her** guidance and assistance in completing this project successfully.

I would also like to convey my sincere indebtedness to all faculty members, including supporting staff of the Department, friends and family members who bestowed their great effort and guidance at appropriate times without which it would have been very difficult on my part to finish the project work.

TABLE OF CONTENTS

ABSTRACT	1
CHAPTER-1 Introduction	
1.1Origin of the Problem	2
1.2 Basic definitions and Background	2
1.3Problem Statement	3
1.4 Real time Applications of Proposed work	3
CHAPTER-2 Review of Literature	
2.1Description of Existing Systems	4
2.2Summary of Literature Study	5
CHAPTER-3 Proposed Method	
3.1Design Methodology	6
3.2System Architecture Diagram	7
3.3Description of Algorithms	9
3.4 Description of datasets and Tools	11
CHAPTER-4 Results and Observations	
4.1Stepwise description of Results and Observations	12
4.2Test case results	15
CHAPTER-5 Conclusion and Future study	
5.1 Conclusion	17
5.2 Future study	17
References	18

LIST OF FIGURES

Figure	Page
Figure 3.1: UML representation	6
Figure 3.2: system of architecture	7
Figure 3.3: description of algorithm	9
Figure 4.1.1: Home pane	12
Figure 4.1.2: View Pane	12
Figure 4.1.3: Create Batch	13
Figure 4.1.4: Create Course	13
Figure 4.1.5: Create Professor	14
Figure 4.1.6: Delete pane	14
Figure 4.1.7: Insert data pane	15

LIST OF TABLES

Table Name	Page
Figure 4.2.1: Output	15
Figure 4.2.2: Batch Timetable	16
Figure 4.2.3: Faculty 1 Timetable	16
Figure 4.2.4: Faculty 2 Timetable	16

ABSTRACT

Timetable Generator is for college applications, it is used to generate timetable automatically unlike traditional approach where a person assigns timetable to batches and faculty. Our product is for educational purposes that help staff to solve this tedious task. The Objective of the project is to generate a time table for professors and batches. 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. The system will take various inputs like number of subjects, teachers, batches, credits per subject, subjects assigned to a batch. By relying on these inputs, it will generate possible time tables for working days of the week for teaching faculty. This will integrate by making optimal use of all resources in a way that will best suit the constraints.

The technicians, trained teachers and any person from the administration should be able to operate it for that were providing user interface.

Keywords: Batch timetable, Professor timetable, User interface

CHAPTER-1

Introduction

1.1 Origin of the Problem

In any educational organization preparing a timetable plays a major role. To generate timetables a person should know large amount of data which he gets in various excel sheets, documents, hand written copies. A person must analyse all these aspects and generate timetable to many batches as well as faculty. In a small organization this can be manageable but as the size of the organization grows it will become more and more complex. Moreover, the current online class trend has been tough lately due to various reasons and setting a timetable for these dynamic changes has been frustrating and it is also has been tough for students and faculty to follow it. For this reason, automating this timetable will reduce human errors, efforts and also saves time.

1.2 Basic definitions and Background

Java swing: Swing is a GUI widget toolkit for Java. It is an API for providing a graphical user interface for Java programs.

- Inorder to lessen the developer burden to create or to develop a java based front end or graphical user interface we can use SWING API(application programming interface)
- Swing features are as follows
 1. Components of the swings are of light weighted.
 2. Components of swing has more functionality compared to AWT.
 3. Execution time of swing is less compared to AWT.
 4. Swing components of java are platform Independent.
 5. Swing provides more powerful components compared to AWT.
- Inorder to carryout the following benchmarks the swing components has to precede the MVC(model-view-controller) architecture they are:
 1. Single API to brace multiple impementations.

2. High-level API is not required to have the data if the API is model driven.

3. If the API is to use java beans for the the model representation it provides better services to the programmer(since java beans implements serializable interface and have parameterless constructor, getters and setters etc..).

- Swing Controls:

Every UI contains Three main aspects namely:

1. UI Elements
2. Layouts
3. Behaviour

- Swing controls inherits the properties from the component class hierarchy.
- Component: Component represents an object with Component Class Hierarchy.
- Container: It is a component that contain other swing components.
- JComponent: A JComponent is a base class for all SWING UI components.

Batch Timetable: Defines the CBCS batches and normal batches and also used to distinguish between the lab batches within the same class.

Faculty Timetable: Faculty timetable views the faculty time table along with batch name and subject dealt.

1.3 Problem Statement

Doing college work in a pandemic time is really a tedious task to do. Also working on daily changing timetable is another hectic schedule. Whereas this application is used to generate timetable automatically unlike traditional approach where a person assigns timetable to batches and faculty. Our product is for educational purposes that help staff and also for students to know their present class.

1.4 Real time Applications of Proposed work

This application will be mainly useful for the teaching department in colleges or schools as well as to students.

- Colleges/universities
- Schools
- Other Organizations which can define the start and end times of a task precisely.

CHAPTER-2

Review of Literature

2.1 Description of Existing Systems

S. No.	Title	Authors	Source
1	Timetable using Genetic algorithm	Pranav Khurana	GitHub
Description: The Project uses advanced Machine Learning Techniques which makes it flexible to all kind of time tables. The main theme of genetic algorithm is generating as many solutions as possible and fix the one that has minimum conflicts. But the trial and error process of genetic Algorithm makes it take longer time than usual. The Process Requires minor editing of the output at the end after all.			
2	Timetable using Backtracking	Nihal	GitHub
Description: The backtracking Algorithm is an exhaustive search which is exponential complexity of time. The project misses special cases like labs, and CBCS classes which makes it not viable for College and Universities but only to Schools. College Time tables Varies widely by students and classes they take. Hence this cannot be used for colleges.			
3	Timetable using PHP and Web application		Nevon projects
Description: The Project uses Genetic Algorithm and developed it as a website. Although website makes it handy anywhere. Any Kind of Server issues may lead to not letting you retrieve the data. Also retrieval from server makes the process slower.			

Table-2.1: Literature study

2.2 Summary of Literature Study:

Project – I:

The genetic algorithm (GA), developed by John Holland and his collaborators in the 1960s and 1970s (Holland, 1975; De Jong, 1975), is a model or abstraction of biological evolution based on Charles Darwin's theory of natural selection. Holland was probably the first to use the crossover and recombination, mutation, and selection in the study of adaptive and artificial systems. These genetic operators form the essential part of the genetic algorithm as a problem-solving strategy. Since then, many variants of genetic algorithms have been developed and applied to a wide range of optimization problems, from graph coloring to pattern recognition, from discrete systems (such as the travelling salesman problem) to continuous systems (e.g., the efficient design of airfoil in aerospace engineering), and from financial markets to multi-objective engineering optimization.

Project – II:

Backtracking is an algorithmic-technique for solving problems recursively by trying to build a solution incrementally, one piece at a time, removing those solutions that fail to satisfy the constraints of the problem at any point of time (by time, here, is referred to the time elapsed till reaching any level of the search tree).

Project – III:

The PHP software works with the web server, which is the software that delivers web pages to the world. When you type a URL into your web browser's address bar, you're sending a message to the web server at that URL, asking it to send you an HTML file. The web server responds by sending the requested file. Your browser reads the HTML file and displays the web page.

CHAPTER-3

Proposed Method

3.1 Design Methodology (UML representation)

This is the UML representation of timetable where Course is assigned to batch and Professor is allotted to a course which he/she teaches. All these form a timetable and can be viewed in a View Pane.

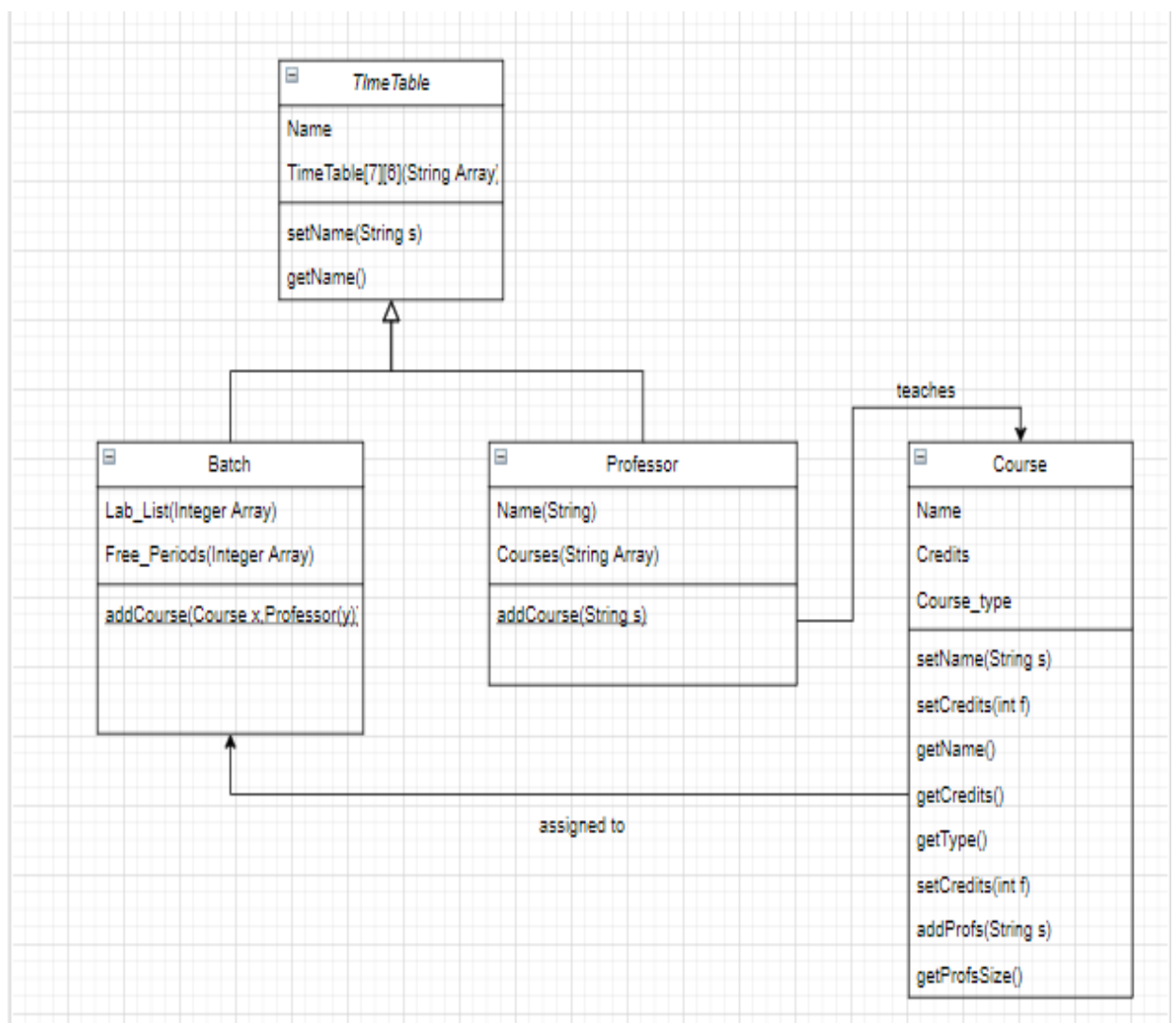


Figure 3.1 UML REPRESENTATION

3.2 System Architecture Diagram

An IT administrator gives all the inputs of Batches, courses and Professors according to feasible hours of the faculty and also students. After assigning everything output is displayed in form a Timetable.

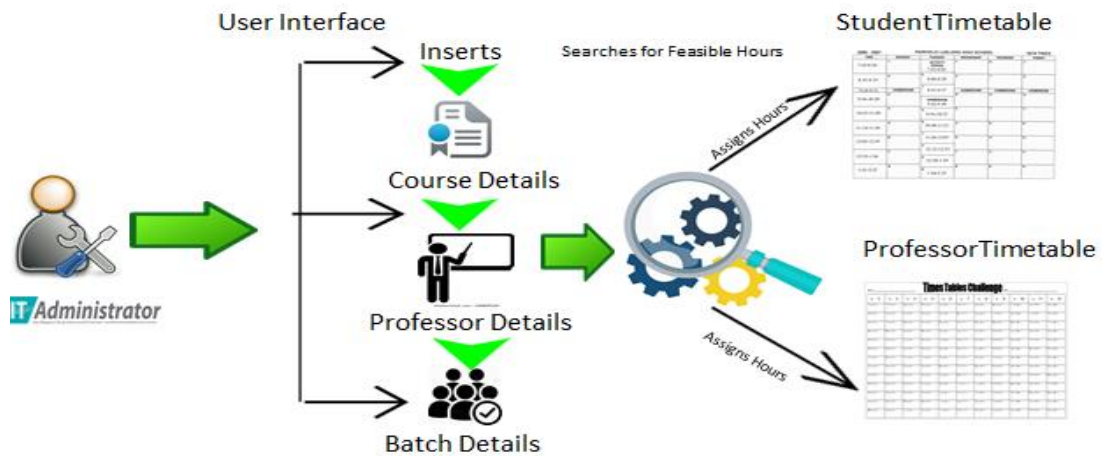


Figure 3.2 SYSTEM ARCHITECTURE DIAGRAM

Description of Modules:

3.2.1 Home Pane

The home pane has no other particular function than to display a welcome Message along with Project Title

3.2.2 View Pane

The View Pane displays the Timetable of Batches and Students. The two combo boxes on the top of the view pane displays the available batches and professors in Drop down.

3.2.3 Create Batch

The Create Batch Pane helps the user to create a batch and assign courses to it. A CBCS course can also be added in the create batch pane. The create Batch Pane Inputs Batch Name and Courses Assigned. If an attempt of assigning courses to the

batch without enough space in timetable has been performed, then a warning popup appears that it is not possible to assign more courses to the batch.

3.2.4 Create Course

The course Pane is the first pane a user must fill in case if a user wants to enter data one by one. The courses will then be appeared on Create Batch Pane Courses list and Create Professor Pane Courses list. In order to assign a course to professor or to a batch you need to register that course first.

3.2.5 Create Professor

The Create Professor Pane must be filled following the Create Course Pane. It inputs Professor Name and Course that the Professor handles. A professor can handle more than one course if the professor's time table permits. If an attempt of assigning courses to the professor without enough space in timetable has been performed, then a warning popup appears that it is not possible to assign more courses to the professor.

3.2.6 Delete Pane

The Delete Pane is used to Delete Courses, Delete Professors and Delete Batch in case of any unnecessary or not used data has been entered.

3.2.7 Insert Data Pane

The Insert Data Pane is the most suggested way of entering data. It allows you to save time and reuse the data. The specifications of how to enter data in Insert Data Pane are mentioned further in the document in Datasets and tools section.

3.3 Description of Algorithms:

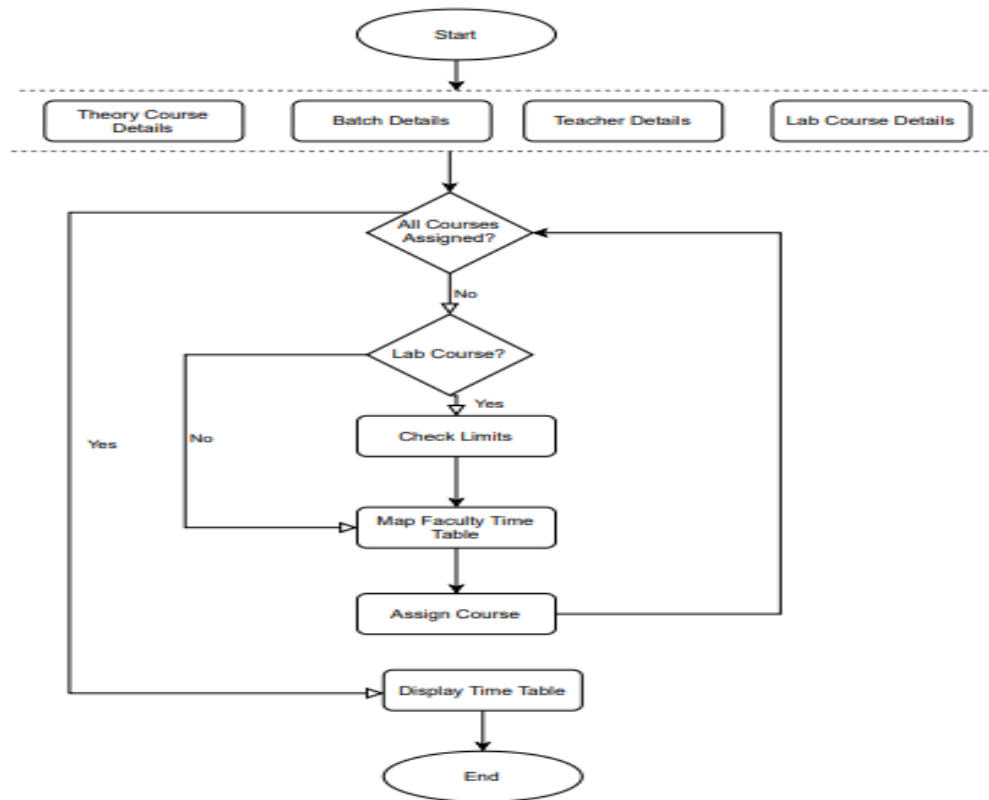


Figure 3.3 Flowchart Of Algorithm

Algorithm:

- Create Classes Professor and Batch separately. Create Time Tables for each of them
- Take Two Time Tables of Faculty and Students. Filter the empty hours.
- `if(Professor.a[i][j]==null && Batch.a[i][j]==null){`

`list.add(i*10 + j);`

`}`

- Add the free hours to the list. Let Course Object be x, Professor Object be y

```
if(list.size()>=x.getCredits()){
    for(i=0;i<x.getCredits();i++){
        k=rand.nextInt(list.size());
        n=list.get(k)%10;
        m=list.get(k)/10;
        this.a[m][n]=x.getName()+"("+y.getName()+")";
        y.a[m][n]=this.getName()+"("+x.getName()+")";
        list.remove(k);
    }
    list.clear();
    return true;
}
```

- Select Random Classes in the given Free hours and assign them to both Faculty and Batch
- The class Professor and Batch inherits from the class TimeTable.
- The Course Class will not inherit from any of the classes but the Professor class and Batch Class holds the references of the Course class in them. All the Courses assigned to a professor are hold in a list.
- Whenever the user presses add course button a method add(Course x,Professor y) will be invoked.
- The add() function is where the logic resides. The add courses calculates the empty slots of both professor and batches.
- Based on the empty cells the add courses calls a random function and assigns Course to that slot of Batch and Professor.

- To get Professor Object based on the Data Filled in the combo Box a Hash Map is used that maps professor name with the actual Professor Object that contains the courses he deals
- Similarly Course Name is mapped with Course Object, Batch Name is mapped with Batch Object.
- Credits play a vital role in assigning classes. Number of credits describe no of period for regular classes whereas for labs no of credits multiplied by 2 determines no of classes.
- The Lab classes must be assigned continuously. Hence The Lab sessions are allotted before the Normal Sessions, to avoid the occupancy of classes.
- The CBCS Classes are generally conducted at the weekend. So, The Weekend Hours are not included in the original list. When the batch is allotted a CBCS class. The hours are selected from excluded hours. Hence No clashes will be there with the original Hours.
- Let's say the list indicates the available hours and lab_list indicates available hours for lab sessions, then
 1. To check if an added Theory course can be allotted to a batch or not:
`if(list.size()>=x.getCredits())`
 2. To check if an added lab course can be allotted to a particular batch or not:
`if(lab_list.size()>2*x.getCredits())`

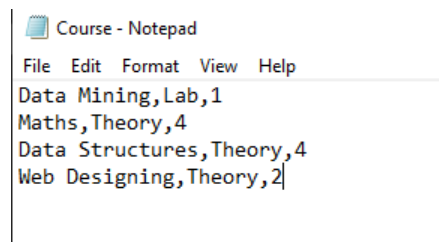
3.4 Description of Datasets and Tools:

The datasets required for generating Timetable are information regarding Batches, courses and Professor Hours.

The Interface allows you to provide them one by one. But at times it becomes essential where you have to insert bulk data at a time to save time.

For that you can insert data via csv files into the application.

Course:



The course.csv file contains three main fields

- Course
- Course Type
- Credits

All written in the same order.

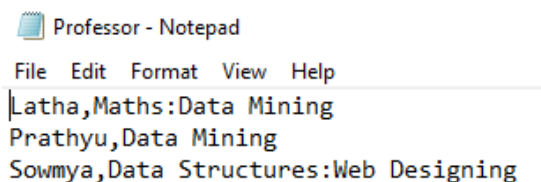
Professor:

The Professor Dataset contains two columns separated by “comma”. A professor can deal more than one subject. So it can be separated by “colon”. Columns of Professor dataset include:

- Professor name
- Courses Dealt

Format:

[Professor_name],[course1]: [course2]:.....[courseN]



CHAPTER-4

Results and Observations

4.1 Stepwise description of Results and Observations

4.1.1 Home Pane

Home pane displays a “welcome” message



Figure 4.1.1 Home Pane

4.1.2 View Pane

View Pane is used to display available batches and professors in Drop down.

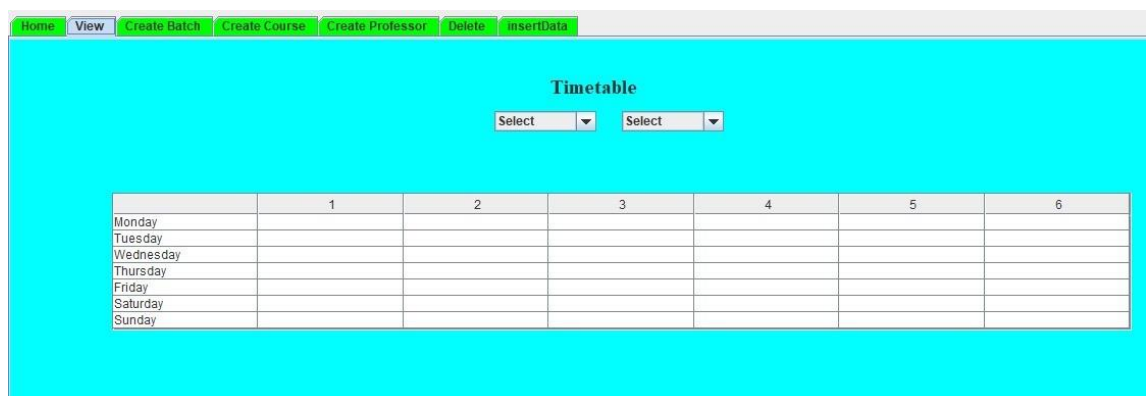


Figure 4.1.2 View Pane

4.1.3 Create Batch

Create Batch is used to assign courses and also CBCS courses for a batch

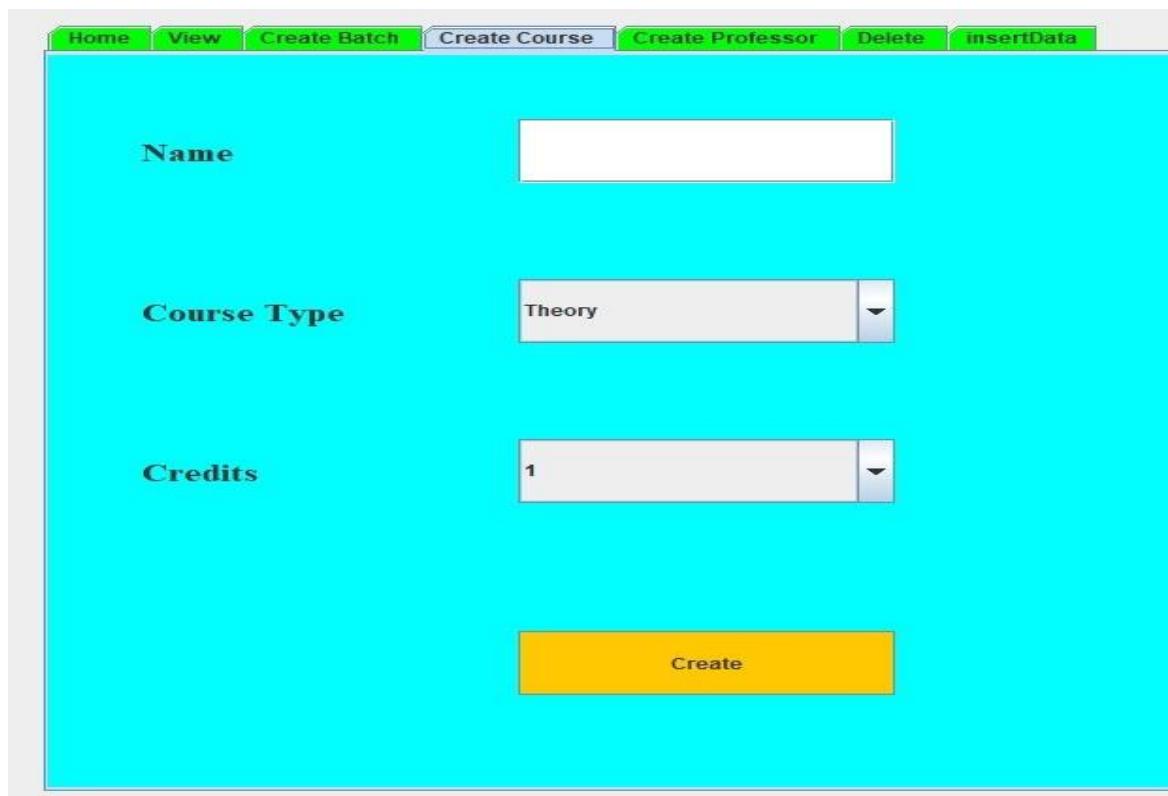


The screenshot shows a web application interface with a navigation bar at the top containing buttons: Home, View, Create Batch (highlighted), Create Course, Create Professor, Delete, and insertData. The main content area has a light blue background and contains three input sections. The first section is labeled 'Name' and has a text input field followed by a yellow 'Create' button. The second section is labeled 'Courses' and has two 'Select' dropdown menus followed by a yellow 'ADD' button. The third section is labeled 'Add CBCS Course' and has a text input field, a 'Credits' dropdown menu, a 'Faculty' text input field, and a yellow 'Create CBCS Course' button.

Figure 4.1.3 Create Batch

4.1.4 Create Course

In Create course a course is created by also giving credits of the course.



The screenshot shows a web application interface with a navigation bar at the top containing buttons: Home, View, Create Batch, Create Course (highlighted), Create Professor, Delete, and insertData. The main content area has a light blue background and contains three input sections. The first section is labeled 'Name' and has a text input field. The second section is labeled 'Course Type' and has a dropdown menu with 'Theory' selected. The third section is labeled 'Credits' and has a dropdown menu with '1' selected. A yellow 'Create' button is located at the bottom of the form.

Figure 4.1.4 Create Course

4.1.5 Create Professor

Create Professor is to assign a particular course to a particular professor.

The screenshot shows a web application interface with a navigation bar at the top containing buttons: Home, View, Create Batch, Create Course, Create Professor (highlighted), Delete, and insertData. The main content area has a light blue background. It features two input sections. The first section is labeled 'Name' and contains a white text input field followed by a yellow 'Create' button. The second section is labeled 'Course' and contains a dropdown menu with 'Select' as the placeholder text, followed by a yellow 'Add' button.

Figure 4.1.5 Create Professor

4.1.6 Delete Pane

Delete Pane is to delete to a course or Professor or Batch.

The screenshot shows a web application interface with a navigation bar at the top containing buttons: Home, View, Create Batch, Create Course, Create Professor, Delete (highlighted), and insertData. The main content area has a light blue background. It contains three distinct sections. The first section is titled 'Delete Course' and includes a 'Course' label, a dropdown menu with 'Select' as the placeholder, and a yellow 'Delete' button. The second section is titled 'Delete Professor' and includes a 'Professor' label, a dropdown menu with 'Select' as the placeholder, and a yellow 'Delete' button. The third section is titled 'Delete Batch' and includes a 'Batch' label, a dropdown menu with 'Select' as the placeholder, and a yellow 'Delete' button.

Figure 4.1.6 Delete Pane

4.1.7 InsertData Pane

InsertData Pane allows you to insert saved data.



Figure 4.1.7 Insert Data Pane

4.2 Test case results

4.2.1 Output

Timetable						
IT-1 Select						
	1	2	3	4	5	6
Monday		Maths(Sowmya)		Maths(Sowmya)		Maths(Sowmya)
Tuesday				Science(Prathyu)		Science(Prathyu)
Wednesday						
Thursday		Physics(Latha)				
Friday			Physics(Latha)			
Saturday			Physics(Latha)		Maths(Sowmya)	

Figure 4.2.1 Output

4.2.2 Batch Timetable will be represented as:

Timetable						
IT-1		Select				
	1	2	3	4	5	6
Monday	Data Structures(Sowmya)	Maths(Latha)				Web Designing(Sowmya)
Tuesday					Data Structures(Sowmya)	
Wednesday						Maths(Latha)
Thursday				Data Mining(Prathyu)		Data Mining(Prathyu)
Friday	Data Structures(Sowmya)	Maths(Latha)		Maths(Latha)		Data Structures(Sowmya)
Saturday			Web Designing(Sowmya)		CBCS-1(KHALID)	
Sunday	CBCS-1(KHALID)					

Figure 4.2.2 Batch Timetable

4.2.3 Faculty1 Timetable will be represented as:

Timetable						
IT-1		Latha				
	1	2	3	4	5	6
Monday						
Tuesday	IT-1(Maths)				IT-2(Maths)	
Wednesday		IT-2(Maths)	IT-1(Maths)		IT-2(Maths)	
Thursday	IT-1(Maths)					
Friday						IT-1(Maths)
Saturday		IT-2(Maths)				
Sunday						

Figure 4.2.3 Faculty Timetable

4.2.4 Faculty2 Timetable if faculty deals more than one course is represented as:

Timetable						
IT-1		Sowmya				
	1	2	3	4	5	6
Monday	IT-2(Web Designing)		IT-1(Web Designing)		IT-1(Data Structures)	
Tuesday			IT-1(Data Structures)			
Wednesday	IT-2(Web Designing)					IT-1(Web Designing)
Thursday			IT-1(Data Structures)			
Friday				IT-1(Data Structures)		
Saturday						
Sunday						

Figure 4.2.4 Faculty Timetable 2

CHAPTER-5

Conclusion and Future study

5.1 Conclusion

Therefore, we conclude it by saying that the project can be successfully used for the current scenario of the college. It covers all the anomaly cases like cbcs, labs and other. The Time Table is editable at the end. Rerunning the program may also give a better solution. For Changing Scenarios the time table is compatible. It can handle small to big changes in the curriculum very easily. The main advantage of the project is it's ready to use nature. Any normal PC with minimal RAM and space can execute the program without using in server connections.

5.2 Future study

The Future Scope of the project is developing offline classes. The present project works well for offline classes also unless there are no class room or lab room clashes. To avoid clashes in classrooms we need to take information about number of rooms and which labs are best suitable for the given lab course. Developing the code for such scenarios will be a great advantage in Future.

REFERENCES

1. Swing Documentation:
<https://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html>
2. Javatpoint:
<https://www.javatpoint.com/java-tutorial>
3. Wikipedia:
[https://en.wikipedia.org/wiki/Swing_\(Java\)](https://en.wikipedia.org/wiki/Swing_(Java))
4. Java Complete Reference:
<https://www.programming-book.com/java-javascript123uo00es0306/>
5. Nevon Projects:
<https://nevonprojects.com/automated-college-timetable-generator/>
6. Youtube Link:
<https://youtu.be/XWdFYHksDQQ>
7. Design Patterns:
https://sourcemaking.com/design_patterns
8. Design Patterns in Java:
<https://www.javatpoint.com/design-patterns-in-java>