Chapter 4

Approach to Implement TMSFIT

4.1 Introduction

Using the technology mentioned in the previous chapter, we provide an approach to solve the problem as follows. This chapter will discuss the proposed solution (hypothesis), functional requirements (Inputs), nonfunctional requirements (outputs), process, features, the users, and the technology that implements the solution.

4.2 Proposed Solution

Formulate an automated Timetable Management System to the faculty by using Genetic Algorithm as a technique to solve the problem, as it is able to produce a feasible timetable and fulfill as many constraints imposed.

4.3 Requirements Elicitation

Interviews and observation are the main methods of data collection through this research. Requirement elicitation was done with the users of the current system. The main user of the timetabling system is the Timetable administrator.

A proper literature survey of previous timetable systems and their algorithms were required to define problem domain.

Then, tried out Number of trial versions of generic products such as TimeTabler, Mimosa scheduling software, OCTT, EduSwift, eTimetable were installed to a computer to analyze their development and the implementation.

After that, let the admins to check the features of those generic products and according to their feedback prepared a model how it would be in future.

4.3.1 Functional Requirements of the proposed TMSFIT

- Register for TMSFIT
- Admin login
- Admin shall add new subject, Batch, Degree (course), student, resource and lecturer
- Admin shall update the details of subject, Batch, Degree (course), student, resource and lecturer

- Admin shall delete the details of subject, Batch, Degree (course), student, resource and lecturer
- Admin shall do the student enrolment via excel sheet
- Admin shall upload the students via excel sheet
- Admin input year
- Admin input semester
- Admin shall generate timetables
- Admin shall view timetable
- Admin shall change allocated time slots manually
- Admin shall allocate more than two labs as required manually
- Admin shall save the timetable.
- Admin shall view the resources according to the time allocation
- Admin shall view the lecturer preferred time
- Lecturer login
- Lecturer view the lecturer profile
- lecturers can view the timetables
- Lecturer shall submit time preferences
- Student Login
- Student shall view the student profile
- Student shall view timetable
- Change password
- Print Timetable

4.3.2 Non Functional Requirements of the proposed TMSFIT

- Load and accuracy
- The system should be available 24 hours
- The system should at the time
- System must be provide specific information to specific user
- The system should not be failed
- Web accessibility

All these functional and nonfunctional requirements were achieved through the proposed Timetable Management System

4.4 Current Timetable Management System

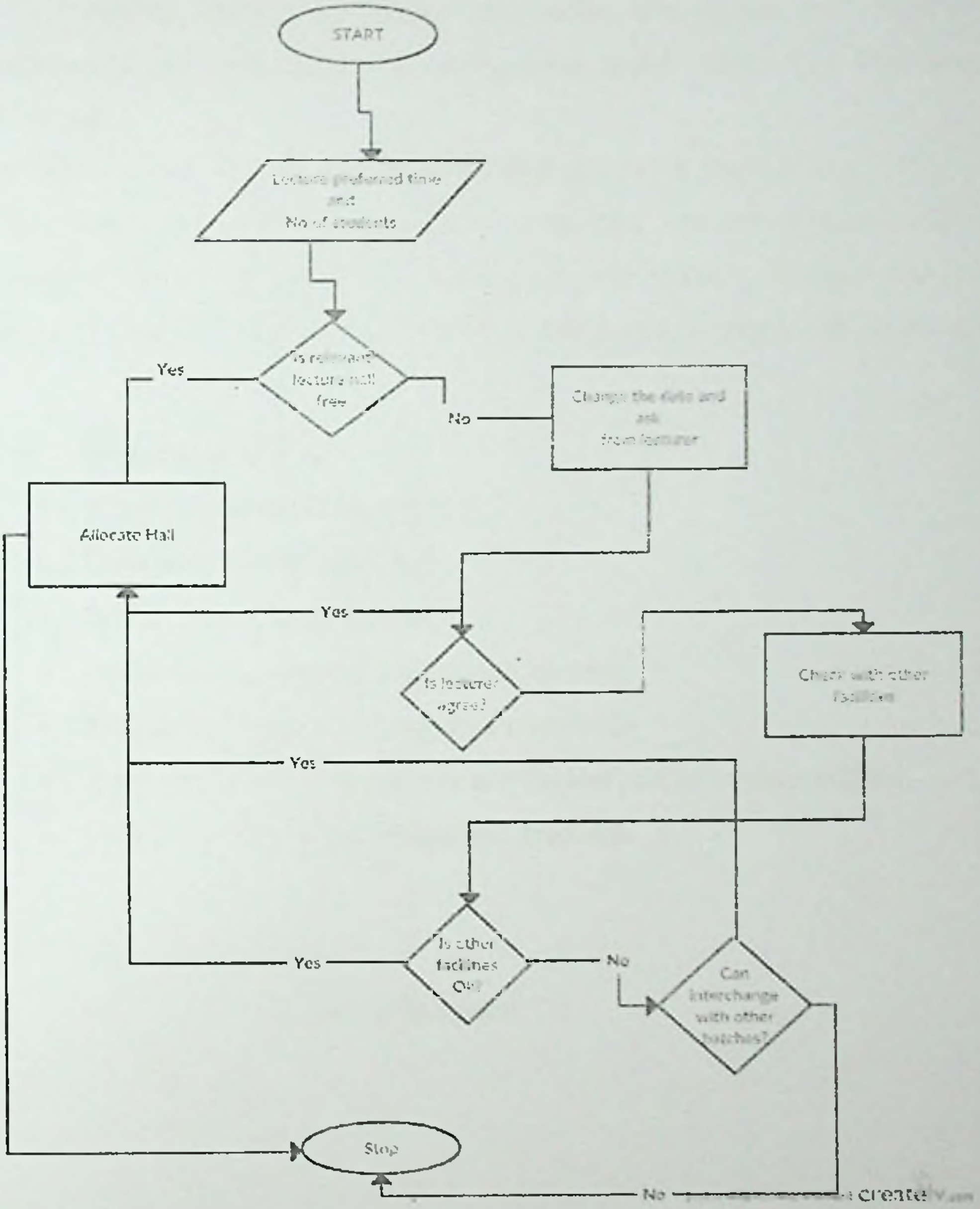
START
Lecture preferred time and No of students
Is relevant lecture hall free
Yes
No
Change the date and ask from lecturer
Allocate Hall
Yes
Is lecturer agree?
Check with other facilities
Yes
Yes
Is other facilities OK?
No
Can interchange with other batches?
Stop
No

Figure 4-1 Flow Chart of the Current Timetabling System

The processes of current timetable management system also used to observe the system well and it is given by a flow chart as above Figure 4 -1

4.5 Process of the TMSFIT

Having received the admin information, the system should verify that information provided. With that, authenticate details admin can log on to the dashboard.

Information must be kept in a database called ttms, with Student data, Lecturer data, Class data, Timeslot data, batch data, semester data, resources data and more. This database must be connected to a web server. Use MYSQL database management system. Using PHP language Yii framework design and develop a web application.

4.6 Features of TMSFIT

- It is a web based online system
- It has user friendly interfaces
- Easy to access to the system
- Accessible through any kind of web browser
- Printable versions of the outputs are available
- Admin profile, Lecturer profile and Student profile can be modified.
- System can be evolved as customer feedback

4.7 Users of the TMSFIT

- Admin of the Timetable System
- Lecturers
- Students

4.8 Technologies used in TMSFIT

4.8.1 Genetic Algorithm

GA is the main technology behind this timetable management system. Using GA, system will generate near optimal timetable solutions not the best timetables.

4.8.2 How GA used in this research?

- Hard constraints (which can't be violated) were used to calculate the fitness value.

- Soft constraints cannot be evaluated
- If breaks one of the hard constraints the schedulė is infeasible

4.8.3 Hard Constraints of Proposed Solution

- Room Overlap – Check if there are two lectures in one room
- Room not enough – No of students of a class is > seats of room
- Required resource not available - Does the lab has required no of Computers?
- Lecturer Overlap – One lecturer can't be in two rooms at the same time
- Student Overlap - One student can't be in two rooms at the same time

4.8.4 System Analysis and Design

Next, system analysis and design sections were done.

According to that, database and the top-level architecture of the system could be figure out.

Because of the fellow facilities, OOAD method was used in the research

- It is better for recognizing objects
- OOAD method organizes the objects by creating object model diagram
- It defines object attributes
- It expressing the behavior of the objects
- It describes how the objects interact

4.8.5 Unified Modeling Language (UML)

Rational Rose and UML diagrams were used in design phase of this TMSFIT to create use case diagrams, class diagrams, sequence diagrams and state chart diagrams

4.8.6 PHP Language

Because of the PHP is a server support, server side language, it was the developing tool used in the TMSFIT. With PHP Object Oriented Programming is accepted.

4.8.7 Eclipse for PHP Plugging

This was used as the code editor of the proposed timetable management system.

4.8.8 WAMP Server

WAMP server with PHP 5.6 and MYSQL installed as the local host. Therefore, TMSFIT could easily install in the www. Root directory.

4.8.9 MYSQL Database Management System

MYSQL was selected to define and store the timetable management (ttms) database.

Database – ttms and the tables were created successfully.

4.8.10 Yii framework

- Yii framework configured to the computer.
- This uses the object oriented concepts.
- Appropriate code construction was done using Yii code generator.
- Yii code generator consist of Controller generator, Curd generator, Form generator, Model generator and the Module generator [23].
- Model generator construct a model class for specified database table.
- ttmsfit PHP project was generated successfully with use of Yii.
- AlgorithmController class is in the controller class and the main algorithm is inside the model class.

4.9 Software and Hardware requirements

The software and hardware mentioned in the previous chapter, were needed to success the proposed TMSFIT

4.10 Interface Design

4.10.1 User Interface Designing with Prototyping

User Interface (UI) prototyping is used as an iterative development techniques in which users are actively involved in the mocking-up of the UI for this system. UI prototypes have several purposes. See the Appendix A

- An item enables to explore the problem space with the users.
- As a design item that enables to explore the solution space of the system.
- It is a way to communicate the possible UI design(s) of the system.
- It is a good foundation to continue developing the system.

4.11 Approach to Timetable Generation Process

Following table will illustrate the input, output, process, features and users of the timetable generation process.

Inputs	**Outputs**	**Process**	**Features**	**Users**
Semester	Generated Timetable	Generate Timetable	Web based system	Admin of the Timetable
Year	Printed Timetable		User friendly interfaces	Lecturers
	View available resources		Accurate	Students

Table 4-1 Approach to Timetable Generation

4.12 Summery

This chapter focuses on the approaches to the design, appropriate development environment and implementation. Design of the approach will be discuss in detail through the next chapter.