

Proposal

for

TimeTable Generator

Version 1.0

CS-08

Indian Institute of Information Technology Vadodara

Team Members

Aman Yadav (201651007)

DakshKumar Gondaliya (201651014)

Kirtika Singhal (201651024)

Mayank Pathela (201651029)

Nikhil Sachan (201651034)

Parmeshwar Kumawat (201651035)

Revision History

Name	Date	Reason For Changes	Version
Kirtika Singhal	30/08/2018	Initiate	1.0

Table of Contents

Table of Contents	3
Abstract	4
1. Introduction	4
1.1 User End	4
1.2 Registration	4
1.3 Selection page	4
1.3 Data Entry	5
1.4 Mappings	5
1.5 Generated Timetable	5
2. Proposed Implementation Plans and Timeline	5
2.1 Project Timeline	5
2.2 Tools and Technology	6
3. Requirements and Assumptions	6
3.1 Requirements	6
3.2 Assumptions	6
3.3 Project Risks	7
3.4 Challenges	7
3.5 Future Extensions	7

Abstract

The manual system of preparing timetable in schools with a large number of sections of classes is very time-consuming. So we are trying to develop a software which will reduce the difficulties and time required for generating a timetable by using constraint-based programming. The system will take various inputs like details of teachers, subjects and class sections, depending upon these inputs it will generate a possible timetable.

1. Introduction

Management of almost every school has to handle three types of resources: teachers, students, and rooms. The manual lecture timetable scheduling demands considerable time and efforts. It is a Constraint satisfaction problem where we find a solution that satisfies the given set of constraints like for this problem some of the constraints are: a teacher can't have more than one lecture at a time slot and a class cannot have two lectures at the same slot. So we are designing a software that will take various inputs like details of subjects, teachers, and sections of a class along with the maximum workload for a faculty for a week to generate a valid time-table satisfying maximum number of constraints. The software we will develop will be a web application having an interface for the user.

1.1 User End

It is an interface for the user where he can generate the timetable and modify it. The algorithm will take a maximum of 3-5 minutes to execute and give a valid time-table.

1.2 Registration

When the user visits the web application for the first time and wants to generate the timetable, he will be asked to register an account. This page will ask him to enter his username and password.

1.3 Selection page

This page will contain two options:

- Data entry page for generating a timetable
- View or Modify the generated timetable

1.3 Data Entry

This page will require the user to input the list of teachers, subjects, class sections, and the maximum number of lectures for that subject in a week. After filling in these details he needs to submit it.

1.4 Mappings

This page will provide admin with a separate dropdown list of teachers, subjects, and sections from which he has to select and fill in a row that will be stored as a tuple in the database. The drop-down list will reduce the efforts of typing the input fields again and again. Also there will be a button at the bottom of the page to generate the timetable.

1.5 Generated Timetable

This page will display the generated timetable.

2. Proposed Implementation Plans and Timeline

2.1 Project Timeline

The following is an abstract high-level Proposed Project Plan and Timeline:

	Description of work	Time
Requirement Gathering & Analysis Phase	We would gather requirements related to the project and analyze it.	3Weeks (25 Aug - 5 Sep) (5 Sep - 13 Sep)
Design Phase	Design of the algorithm, web page and its functionalities will be prepared in this phase	3Weeks (15 Sep - 30 Sep)

Coding and Unit Testing	Test-driven development will be used to develop the project. Test cases will be defined at the end of the design phase and coding will be done according to those test cases.	3Weeks (1 Oct- 24Oct)
Testing Phase	During this phase, we will be carrying out intense testing of the whole project modules. We will make sure that the application runs error free across all the devices. We will ensure that everything is working as designed and desired.	1Weeks (28 Oct - 4 Nov)
Deployment Phase	The software will be deployed on the server in this phase.	1Week (5 Nov - 11 Nov)
	Total	2 months 3 weeks

2.2 Tools and Technology

Following is the list of tools and technologies we are planning to use(they may change if extra requirements arrive):

1. React Framework
2. Node.js
3. MongoDB
4. AWS cloud storage

3. Requirements and Assumptions

3.1 Requirements

We will require Amazon Web Service(AWS) for the database.

3.2 Assumptions

We have made certain assumptions regarding the timetable. Some of them are:

- The schools will have access to Computers and internet connections.
- Teachers are available for all the working days in a week, i.e, no preferences are taken into consideration.

3.3 Project Risks

There may occur a chance where the algorithm may not show any solution. In that case, we will start relaxing the constraints and generate a new timetable.

3.4 Challenges

The major challenge in our project is to minimize the number of clashes, ie, a class should not have two lectures at the same time and a teacher can't have more than one lecture at the same slot. We have to minimize the time of execution also.

3.5 Future Extensions

In the future, we can expand it for colleges for which we have to add extra constraints for the backlogs, and preferences for visiting faculties.