A Major Project final report on

College Routine Generation System

Submitted in Partial Fulfillment of the Requirements for

The Degree of **Bachelor of Engineering in Software Engineering**

Under Pokhara University

Submitted by:

Anurag Regmi, 15703

Krrish Ghimire, 15716

Nikhil Pandey, 15719

Date:

22/11/2019

Department of Software Engineering

NEPAL COLLEGE OF

INFORMATION TECHNOLOGY

Balkumari, Lalitpur, Nepal

ACKNOWLEDGEMENT

We would like to express our great appreciation to our project supervisor Mr. Madan Kadariya for his valuable time and for his valuable advices and suggestions throughout this project.

ABSTRACT

The project entitled ‘College Routine Generation System’ is a web-based project. In this project we will be designing and implementing an application which will generate routine for a college with minimum user inputs. First of all, we will model our problem and develop an engine to generate routine for a class. Then we will generate for the whole college. In the next iteration we will develop user interface for the engine. Then we will build interface to save and manage routines. The backend will be developed in Java using Spring Framework. The frontend will be developed in VueJS and Vuetify for styling. Postgres will be used for database. After this project, the product will help management to save time used in developing and managing routine.

Keywords: -

College Routine, Web-based application, Java, VueJS, Postgres

Table of Contents

List of Figures

1. PROBLEM STATEMENT

The main problem when a person manually creates routine for a college is, he/she will not be able to check conflicts efficiently. There are numbers of classes per day for a teacher. Also managing practical classes without conflict is tuff. When a person successfully creates a routine modifying it is another time-consuming task. The problem statement of our project is to deduce an algorithm that generates a conflict free routine. Other challenges are to keep timing constrains of teachers in check, maintaining number of classes in a week and managing practical classes.

2. PROJECT OBJECTIVES

The basic objective of this project is to develop a routine generation system that will:

a. Automate the process of routine generation of class/semester with minimal inputs.
b. Create a conflict free routine.
c. Easy Update the routine after generation.

3. SIGNIFICANCE OF THE STUDY

The system is expected to be user friendly and generate routine with simplicity, removing all the complications involved in manual generation. After this project is complete, it will benefit administration and teachers in following ways -

a. Reduce time and effort given in routine generation by providing them a Simple Interface,
b. Easy maintenance of generated routines as system will track all routines.
c. Reporting section will make it easier to access routi Frontend UI design and develop

4. SCOPE AND LIMITATIONS

The scope of this project is to create a routine generation system for a college. With this system user will be able to -

a. Generate routine for a college.
b. Manage generated routines.
c. View different views of routine (as student, teacher, administration).

Despite our best efforts to make the process automatic, the user still has to insert some data manually, such as courses and subjects, teachers' name and subjects, timings, before he/she could generate routine from the system

5. LITERATURE REVIEW

5.1 REVIEW

Managing classes for the teachers in certain way so that there must not be any conflict with other classes and teachers' timing is a growing and a massive challenge. To overcome that issue, our routine generation system mainly focuses on conflict free routine so that every classes with the numbers of periods in a week can go smoothly. The main purpose of this system is to reduce the workload of admin person as he/she has to make new routine every time in case of any changes in routine of a class. This system manages classes for the teachers in a certain way so that there must not be any conflict with other classes and teachers' timing. To overcome that issue, our routine generation system mainly focuses on conflict free routine so that every classes with the numbers of periods in a week can go smoothly. The main purpose of this system is to reduce the workload of admin person as he/she has to make new routine every time in case of any changes in routine of a class. This system tries to model the problem as N-queen problem so that it will be helpful in generating a conflict free routine automatically. There have been various attempts at creating a routine generator, most of them have failed due to its complexity.

5.2 EXISTING SYSTEMS

While studying the existing systems for the routine generation we found that either they are a component of big software, e.g. ASTI (Any Time Student Information) [1], or standalone software like Automated College Timetable Generator [2] and Free college Schedule Maker [3]. The first type of systems has done a really good job in routine generation, they look premium and feature loaded but the problem with them is you have to buy the whole package with it, which you may not want. Second ones are cheap and are standalone but they are not as effective as previous one. Most of them provide routine generation for only one class, for e.g. Free College Schedule Maker [2]. This is the problem that will be shortened in our system as we will build the system in a way that system will be able to generate weekly routine for whole college with very minimal data fed into the system. System should always be efficient in a way that it shouldn't be heavy in terms of design, developments and implementation.

6. METHODOLOGY

We follow Agile Unified Process for this project.

6.1 AGILE UNIFIED PROCESS

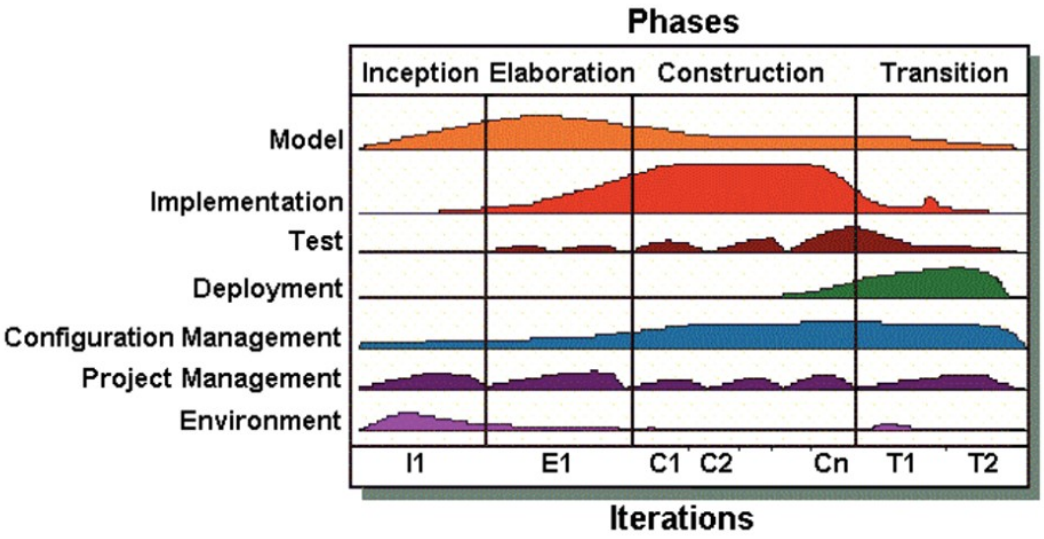

Figure 1: Agile Unified Process

Agile Unified Process[3] has following phases:

6.1.1 MODEL

We discuss the business of the project with the college representative, and analyse the requirements. Then, we discuss the viable solution with the team.

6.1.2 IMPLEMENTATION

We follow TDD approach for implementation. We write unit tests for a requirement. Then add production code to make the unit test pass. We repeat this step until the requirements are met.

6.1.3 TEST

As unit tests are written in implementation phase, we also add some integration tests. These tests validates if all the units work when integrated together.

6.1.4 DEPLOYMENT

We use the CI/CD provided by Gitlab to make sure about the frequent releases.

6.1.5 CONFIGURATION MANAGEMENT

Each release have different versions, and we keep track of each of them.

6.1.6 PROJECT MANAGEMENT

We have regular meeting with team members. This helps to know about the progress and status of the project.

6.1.7 ENVIRONMENT

In our regular meeting with team members, any required resources are discussed and made available as needed.

6.2 TOOLS AND TECHNOLOGIES

Following are the tools and technologies that are used throughout the project.

a) Vue.js (Frontend)

b) Java, Spring Framework (Backend)

c) Git (VCS)

d) Gitlab (Source code repository, CI/CD)

7. APPENDIX

7.1. ER- Diagram

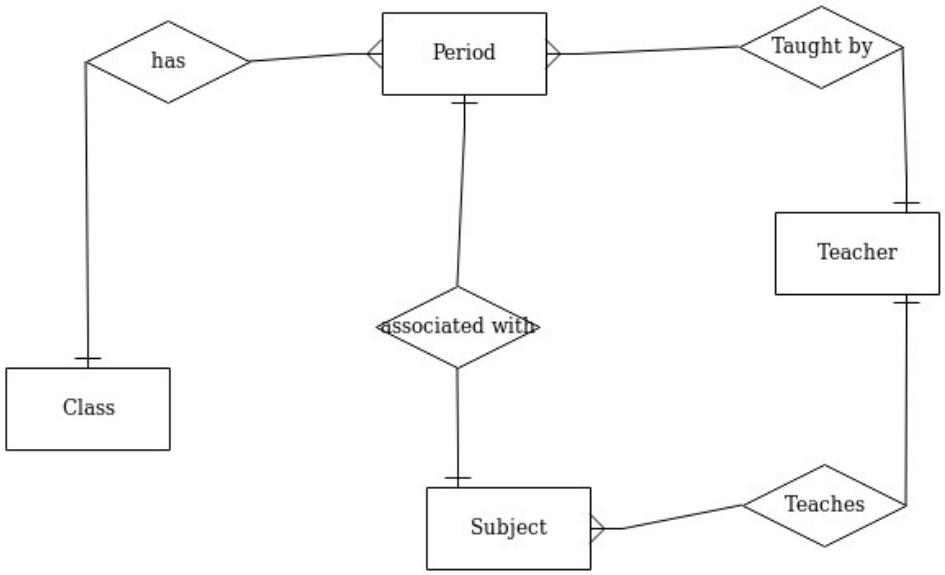

Figure 2: E-R Diagram

8. REFERENCES

[1]: Automated college timetable generator, *https://soft-project.blogspot.com/2009/01/routine-management-system.html*

[2]: ASTI Website, *https://www.atsi.in/*

[3]: Agile Unified Process, *http://www.ambysoft.com/unifiedprocess/agileUP.html*