Software Requirements Document

CS308: Large Applications Practicum
Indian Institute of Technology Mandi

October-November 2019

Group 9

B17096 Namrata Malkani

B17031 Aashima

B17033 Akhil Rajput

B17064 Suraj Kumar

Time-Table Assist Tool
Requirements Document

Revision History

Version	Date	Author(s)	Description
v1.0	15/10/19	Aashima	Initial version

Table of Contents

1 Introduction

Time-Table Assist Tool is a user interface designed to prevent errors while making time-table. It does not make the time table, it just assists the user by flashing errors. The user interface is connected to a database. The database stores all the information required to evaluate clashes. Each course is to be allotted a slot in the time table, a classroom and instructor(s). The entire decision of allotment lies with the user but the interface blocks him from making clashes. Therefore, data entries violating constraints are not inserted into the time table and user is flashed error so that he/she can resolve it.

1.1 Need and Purpose

A correct time-table is very crucial for better functioning of the institute, faculty and students. Resources like time slots, classrooms and faculty are limited, hence, they should be allocated efficiently. There are additional constraints like transportation etc which can be better understood by humans than the program. With so many requirements and a few ambiguous ones, it is very difficult to design a program which constructs the time-table from scratch. Therefore, time-table is designed manually.

Manual preparation solves the problem of designing the hard algorithm but it increases the chances or errors. The Time-Table Assist Tool solves this problem. It checks constraints each time the user enters new data and blocks the insertion if clash occurs. It displays the reason of clash, so that the user can resolve it.

1.2 Intended Audience

This document is intended for software engineers and the user, who wish to understand how the Time-Table Assist tool works. It gives freedom to the user, at the same time avoids clashing allocations and makes time-table designing process easier.

2 Description

The Time-Table assist tool has a user interface for allocating new courses, slots and classrooms. There is a main database and a run-time database. The main database contains all the information of courses, respective instructor(s) and classrooms available. The time-table is stored in a database which is modified in runtime. At each step, the interface queries the main database to evaluate constraints.

The tool makes sure that a course is added in the database only once, two courses in the same slot do not have same classroom and a faculty is allocated only one course in a particular slot. If the allocation does not cause any clash, it is inserted in the runtime database else the request is blocked and an error is displayed. It shows the reason for clash, so that the user can resolve it.

2.1 Features and Functions

2.1.1 Features

User can use the time-table assist tool on his/her PC. It works offline so it does not need internet connection. The user needs to run a python script which opens up a user interface and all the work is done on the interface.

2.1.2 Functions

The user can

1. make new allocations
2. delete old allocations
3. save the work
4. restore the last saved work

The user can understand the complete database and its working as all the files are available to him/her and the program runs locally.

2.2 Users

Time- Table Assist tool is made for the person who designs the time table by listing all constraints from the information available.

2.3 Operating Environment

2.3.1 Hardware

The Time-Table Assist Tool works on your personal computer. It is a small program.

2.3.2 Software

The Tool can run on any recent version of Linux, such as Ubuntu, Debian, Fedora etc. It requires:

1. Python
2. Sqlite3

3 Specific Requirements

3.1 Design Constraints

The interface creates the runtime database but it needs the main database to evaluate some constraints which are not explicitly visible to the user. Therefore, the main database needs to correct and contain all required information. This is the user's responsibility. The runtime database has been designed observing the institute availability of 8 time slots. If the number of time-slots change, the program needs to be changed slightly.

4 Mini-Project Plan

Version 1:

- Design and implement the main database. Extract data from files, arrange it and import it to the main database.
- Design a basic GUI which takes input from the user, Flashes error if a clash else insert it into the runtime database. Check all the constraints by querying on the runtime and the main database. The runtime database is visible on the terminal each time the user inputs some data.
- Test on few entries.
- The First version fulfils all the needs for adding new allocations.

Version 2:

- Implement Delete, Save and Restore Feature.

- Create a dummy time-table to test all aspects.
- Improve the GUI. Show the current time-table on the GUI.
- Convert all the implementation of constraint checking from the python program to embed it into the database if necessary.
