

Software Requirements Specification

Time Table Management System

Version 1.4

Prepared by

Manan Nahata
Himesh Jain

Table of Contents

1.	Introduction	4
1.1	Purpose	4
1.2	Document Conventions	4
1.3	Intended Audience and Reading Suggestions	4
1.4	Product Scope	4
1.5	References	5
2.	Overall Description	5
2.1	Product Perspective	5
2.2	Product Functions	5
2.3	User Classes and Characteristics	6
2.4	Operating Environment	7
2.5	Design and Implementation Constraints	7
2.6	User Documentation	7
2.7	Assumptions and Dependencies	8
3.	External Interface Requirements	8
3.1	User Interfaces	8
3.2	Hardware Interfaces	8
3.3	Software Interfaces	9
3.4	Communications Interfaces	9
3.5	System Feature 2 (and so on)	11
4.	Other Nonfunctional Requirements	11
4.1	Performance Requirements	11
4.2	Safety Requirements	11
4.3	Security Requirements	12
4.4	Software Quality Attributes	12
4.5	Business Rules	12
5.	Other Requirements	12

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Web-based “timetable management system”. It will explain the purpose and features of the system, the interfaces of the system, what this system will do, the constraints under which it must operate. The document also describes the nonfunctional requirements such as the user interfaces, design and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

1.2 Document Conventions

1.2.1, Priorities for higher-level requirements are assumed to be inherited by detailed requirements.

1.3 Intended Audience and Reading Suggestions

This project is a Module for the timetable management system, that will be using other APIs to get data and it is restricted within the college premises. This has been implemented under the guidance of college professors. This project is useful for the entire college administration, professors and as well as to the students.

1.4 Product Scope

The objective of the project is to create a web-based timetable management system to be used by any institute but it is personalized according to the needs of our institute i.e, The LNM Institute of Information Technology. The main idea is to ease off the database maintenance of subjects, branches, semesters more efficiently.

As it is a web-based system, the maintenance and working is somehow different from the manual mode of approach. Till this date, the timetable making process is completely manual.

Hence, we want to make this system very user-friendly and easy to understand.

Since this is a web-based project, it is easier to fetch the data from the database and remove the unwanted data by just clicking the delete button. This project will help in saving the resources that are currently consumed in timetable management.

The maintenance of this web-based will be much easier and accurate than the existing manual system.

As this web-based system, the security features are somehow higher than the manual system. In this system, there is less chance of mishandling of the data because only the Administrator will login through their login id and password and upload all the related data and generate the timetable. So, there is no misuse of the data.

1.5 References

- 1) Minutes of meeting with Dr. R Gorthi (Software E. Course instructor) .
- 2) Discussion during meeting with Sir.

2. Overall Description

2.1 Product Perspective

Timetable Management System is supposed to be a part of a larger software that is been already in our College (using APIs of other Modules). This module will provide a non-hideous way to design a timetable for college of any size.

2.2 Product Functions

The system should be user-friendly, accurate and fast at the same time.

The system will allow the admin to-

Create a complete timetable with a given set of data (data will be from API of Curriculum & Faculty).

The system must meet the entire needs of the institute.

Interface for input: The system will be having an easy to use and interactive interface to enter all the inputs like the workload for the teachers, how many no. of students of which branches are studying the subject in a semester, the data for the rooms and data for the labs.

Database Capabilities: The system will have a well-designed database to store all the information which will be entered in as the input. Separate database maintaining basic information, subjects, teachers, batches and their associations and other details Database for holding generated timetable and for storing required timetables.

Processing Capabilities: The system will have algorithms to process all the data present in the database and keeping in view the various constraints like that a teacher should not have two consecutive lectures/labs, students have a minimum one hour gaps, proper rooms are allocated for the lectures and tutorials, labs are used optimally so that they are used for the maximum possible time, it will generate the timetable.

Search Panel: The system will have an easy to use to search panel to search according to his need on the timetables stored in the database. The system would give the response to the user queries in the proper format and errors messages will be shown properly to tell the user about his mistakes and to guide him/her for proper use of the system.

Features for assigning priorities for subjects.

Features for editing generated table, saving edited tables and opening saved tables

High portability, works on almost all systems available.

Highly efficient needs only a few minutes to complete the whole procedure

2.3 User Classes and Characteristics

This system is only accessible to Administrator, Dean Academics and Director.

The features that are available to the Administrator are:

1. The administrator has a unique login id and password.
2. Can insert the data related to the subjects, teachers, labs and the college.
3. Can generate the timetable according to the data entered.
4. Can print the timetable according to the branch and semester.
5. Can see the schedule of any branch.

The ideal algorithm-based approach won't be able to bring quick results hence we will proceed with knowledge-based approach and consider some assumptions and analyze the manual way of scheduling timetable and finally write the code. Through this approach, our code will not only be faster but also easy to understand and maintain.

2.4 Operating Environment

Since our chosen platform is web hence we will be having very minimal hardware and software requirements for any machine to run it.

Hardware requirements -

- . Any device with stable internet connectivity
- . Display preferably 16:9 in dimensions
- . Keyboard/Mouse for data input
- . At least 512 MB of RAM
- . At least 128 GB of internal hard disk
- . intel core 2 duo and above version of processors

Software requirements -

- . Any operating system
- . Browsers - Internet Explorer 8 and above, Mozilla Firefox version 35 and above, Any Chromium version.

2.5 Design and Implementation Constraints

HARD CONSTRAINTS:

- 1) Class can only be assigned if the room is spare.
- 2) No teacher or student has more than one class at a time.
- 3) A classroom must have enough seats to accommodate all students.
- 4) Minimum 1hour Break time between different lectures for the teacher.

SOFT CONSTRAINTS:

- 1) Time of class preferred by the teacher.
- 2) Specific Room preferred by any teacher.
- 3) Less walk to attend lectures by the student.
- 4) Lectures are not assigned to time slots which are in instructors' forbidden time slot.
- 5) Lecture hours should be scheduled consecutively.

2.6 User Documentation

- 1) The details of the (i) Analysis, (ii) Design and (iii) Test Cases for package will be delivered along with this software.
- 2) The details of (i)Class diagram (ii)DFD (iii)Sequence Diagram along with Activity Diagram will be provided.

2.7 Assumptions and Dependencies

The requirements will not change once they have been finalized. Sufficient information and documents from the university will be available whenever required. The response time of the system will be accepted whatever it may be. The system would be used only for windows operating system.

Here are the set of assumptions -

- 1) There will be six working days in a week with Sunday being a holiday.
- 2) Schedules of Monday, Wednesday and Friday would be identically same.
- 3) Schedules of Tuesday, Thursday and Friday would be identically same.
- 4) All classes would be of 1 hour each.
- 5) The First version will only comprise of solution for students registered in B-Tech course.
- 6) Repeaters are not considered while designing and implementing the solution.
- 7) Morning Slots (8:00 AM to 12:00 PM) are fixed for CSE and MME students
- 8) Evening Slots (1; 00 PM to 5:00 PM) are fixed for ECE students
- 9) CCE will do both CSE and ECE classes.

3. External Interface Requirements

3.1 User Interfaces

As the system is easy to handle and navigates in the most expected way with no delays. In that case, the system program reacts accordingly and transverses quickly between its states.

3.2 Hardware Interfaces

- . 2 GHz x86 processor
- . Any device with stable internet connectivity
- . Display preferably 16:9 in dimensions
- . Keyboard/Mouse for data input
- . At least 512 MB of RAM
- . At least 128 GB of internal hard disk
- . Intel core 2 duo and above version of processors

3.3 Software Interfaces

Software requirements -

- . Any operating system
- . Browsers - Internet Explorer 8 and above, Mozilla Firefox version 35 and above, Any Chromium version.

Data lifecycle - The administrator selects the type of operation and adds details through UI via a browser in a display and then the added data goes to our database for permanent storage and from there a manipulated data is presented to the user via UI as per request.

3.4 Communications Interfaces

The server in which we will host our application will follow HTTP communication standard. The communications will be done primarily through email hence a mailing module will be needed to install so whenever there is a change in timetable a mail is sent to the students regarding the changes in timetable. Since this version of timetable management system is personalized for LNMIIT hence we will be using student's already existing user login credentials to show them updated timetable.

4 System Features

4.1.1 Description and Priority

Generate the timetable (High Priority)

DESCRIPTION:

The Assistant Registrar is the one responsible for generating the end timetable from the existing rooms, courses, and faculty databases. He must first specify the start time of classes, the duration of a single period or event, the time of lunch break, as well as the total number of classes in a day. Thereafter, he will have to identify the department and semester for which the timetable is being generated, after which, he may begin creating the timetable.

4.1.3 Functional Requirements

REQ-1: View of the timetable

DESCRIPTION:

The application will present a general view of the time table in PDF format.

Output: PDF Timetable Generated

R1.1: PDF timetable generated

Input: "Create" option

Output: A PDF of timetable

REQ-2: Application of the system by the manager /the administrator

DESCRIPTION:

The Assistant Manager should have authority over the complete system. He must have unique username and password as login credentials. Thereafter, he must be able to view, editor delete any database if need be. In addition, he must be able to add new users or remove any existing ones.

R3.1: Login

Input: Username and password

Output Successful authentication, display of the main window

R3.2: Access External APIs

DESCRIPTION

The Assistant Registrar should be able to access data from different APIs like Courses, Faculty List etc.

R3.2: Create, Read, Update and Delete Room(CRUD) Database

The Assistant Registrar should be able to create and manage Rooms database as it does not exist in the current scope of software.

DESCRIPTION

The Assistant Registrar should be able to access data from different APIs like Courses, Faculty List etc.

REQ-4 Generate the timetable

DESCRIPTION:

The Assistant Registrar is the one responsible for generating the end timetable from the existing rooms, courses, and faculty databases. He must first specify the start time of classes, the duration of a single period or event, the time of lunch break, as well as the total number of classes in a day. Thereafter, he will have to identify the department and semester for which the timetable is being generated, after which, he may begin creating the timetable.

R3.4.1: Specify duration and number of periods in a day

Input: Duration of a single period; the number of periods in a day

Output: Prompt for start time of classes

R3.4.2: Give the start time of classes, and the timing of the lunch break

Input: Start time of classes; start time and duration of lunch break

Output: Prompt for department details on successful completion

R3.4.3: Enter department details and semester

Input: Department code and name; semester

Output: Show window for creation of the timetable

3.5 System Feature 2 (and so on)

4. Other Nonfunctional Requirements

4.1 Performance Requirements

The product shall be based on web and must be run from a web server.

The system must be interactive, and the delays involved must be less. So, in every action-response of the system, there are no immediate delays. In case of opening window- forms, of popping error messages and saving the settings or sessions there is delay much below 2 seconds.

In case of opening databases, sorting questions and evaluation there are no delays and the operation are performed in less than 2 seconds for opening, sorting, computing, posting > 95% of the files.

4.2 Safety Requirements

Information transmission should be securely transmitted to server without any changes in information

4.3 Security Requirements

- The main security concern is for the Administrator's account hence proper login mechanism should be used to avoid unauthorized access.
- The product shall be based on the web and must be run from a web server.
- The customer's web browser shall never display a password. It shall always be echoed with special characters representing typed characters.
- The system's back-end servers shall never display a password not even to Database Managers. The password may be reset but never shown.
- The system's back-end servers shall only be accessible to authenticated administrators.
- The system's back-end databases shall be encrypted.

4.4 Software Quality Attributes

Availability

Checking that the system always has something to function and always pop up error messages in case of component failure. In that case, the error messages appear when something goes wrong so to prevail availability problems.

Functionality

Checking that the system provides the right tools for editing databases, creating Time Tables and analyzing the Time Table.

Usability

As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states.

4.5 Business Rules

The administrator has all the rights other users like faculty can report their queries which appears in Administrator's page. All changes will only be made by Administrator.

5. Other Requirements

Appendix A: Glossary

Appendix B: Analysis Models

Appendix C: To Be Determined List