Software Requirement Specification (SRS)

Routine Generation and Modification System (RGMS)

Muntasir Ahmed
UG02-32-13-004

Table of Contents

1. Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose for this document is described.

1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the "Routine Generation and Modification System" (RGMS) software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

1.2 Scope of Project

The software will help to create a routine for any university, school or college. Primarily the user has to provide information like number of courses, teacher's name and register students to a course etc. Then the user will see a box of seven day with various slots in each day. User will also see the course list beside the screen. She/he can drag and drop any course on the routine. If there is any type of conflict, it will show that. All the information along with the routine will be saved in the database. User can also update the routine if she/he wants.

The software is a desktop application. It doesn't require Internet connection to generate the routine. The information will be saved in its local database.

More specifically, the information of name or id of the courses, id of the students registered with the course and course credit will be taken input from an excel sheet and user will ask to give input about the teacher's id and number of slot needed per course. User can see the type and number of conflict of any course by clicking on the slot in the routine.

1.3 Glossary

Term	Definition
User	Who interacts with the system
Slot	1.5 Hour class

DESC	Description
RAT	Rational
DEP	Dependency
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.

1.4 Overview of the Document

The remainder of this document includes two chapters. The second one provides an overview of the system functionality and system interaction with other systems. This chapter also introduces user and his/her interaction with the system. Further, the chapter also mentions the system constraints and assumptions about the product.

The third chapter provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify the requirements more precisely for different audiences.

2. Overall Description

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of users that will use the system and what functionality is available for each type.

2.1 Product Perspective

The system mainly consists of two parts: one is the software and another one is the database. Initially the software saves all the data on the database and when needed, it extracts data from the database and process to show results.

The software will initially gather the necessary information such as course id, batch id, and id of the students registered with the course, Department etc.

It will also ask for the data like room id, teacher id, slot required per course etc. This data will then be saved in the database. And when any user interacts with the system, this information of database will help the software to show how many conflicts are there and available room etc.

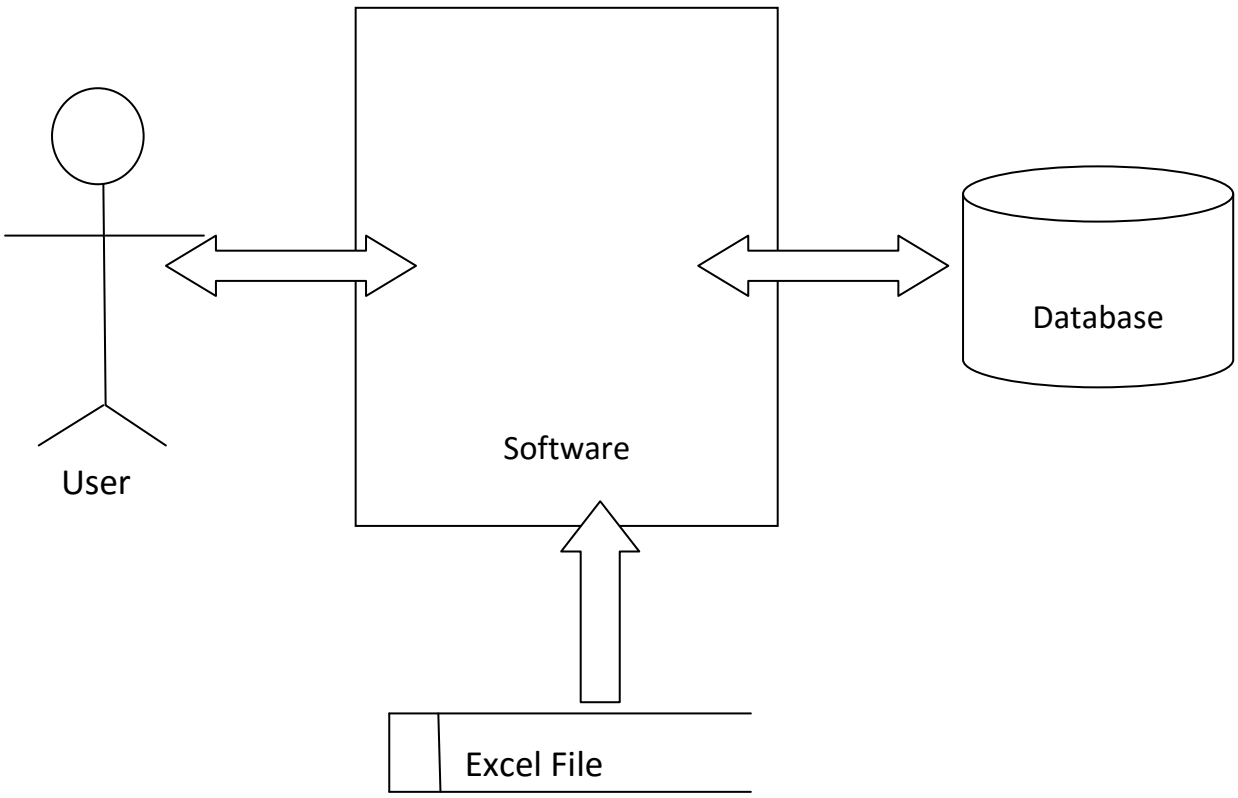

Figure 1: System Environment

2.2 Product Functions

With the software, the user will be able to create a routine. The screen will be divided into various row and column. The row represents days in week and column represents various time slots in a day. When any user click on any course beside the routine and drop it in any time slot, he will be able to see if there is any conflict or not. The conflicts can be one batch has two courses on same time, one teacher has two courses on same time, one room has been allocated to two courses on same time or room is not enough for any course and any student has two courses on same time.

Database will help to show the conflicts. The more the conflicts is, the more deep the color will be. If there is no conflict, it will be white. User can also change any dropped course to another position. By showing suggestion, this software will guide the user to create a complete routine.

2.3 User Characteristics

There is mainly one type of user who uses the system. She/he will only be allowed to generate and modify it.

This software will be installed in one pc and will have access to one user. The user will give input, will make necessary correction.

The user will be the admin of the system and will have the full permission to make any kind of changes in the routine.

Though the routine will be mailed to other teacher after completion, since they have no interaction with the system, they will not be count as user.

2.4 Constraints

The software is constrained with the system interface. It will not generate any routine automatically. Rather it solely depends on user choice. Until user select or drop any course, it will not show any suggestion or conflict.

Moreover, the system depends on the information of database. So the user doesn't provide enough information, the system might show wrong information or might not show any conflict at all.

To let the system work appropriately, the database is needed to be up to date according to students' information.

3. Specific Requirements

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all its features.

3.1 External interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

3.1.1 User Interfaces

When a user opens the software, he will see a home screen. On home screen, there will be several buttons. Clicking each button will create a new window.

If the user clicks on the Room info button, the following screen will appear.

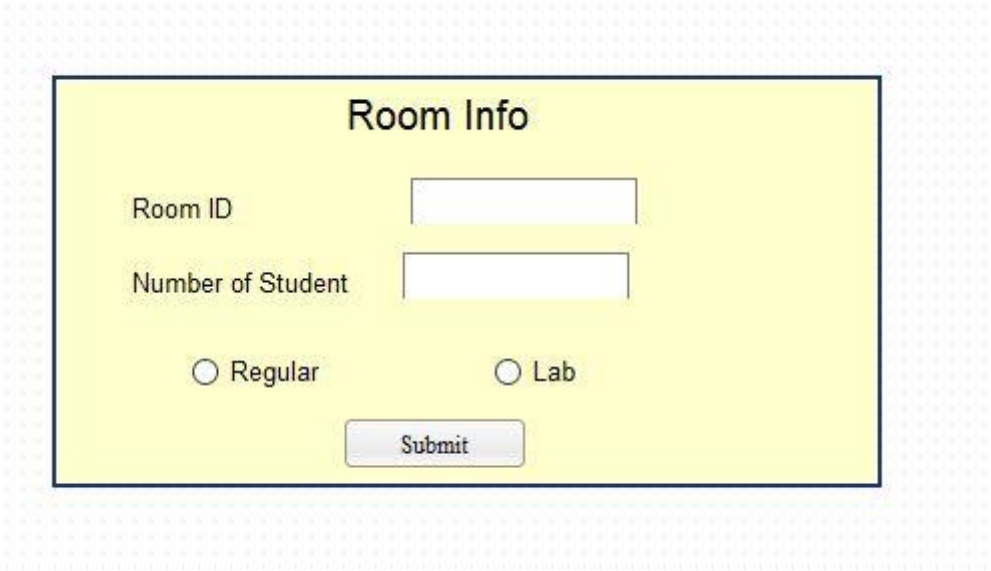

Fig 2. Room Info

If the user clicks on Teacher and Slot button, the following screen will appear.

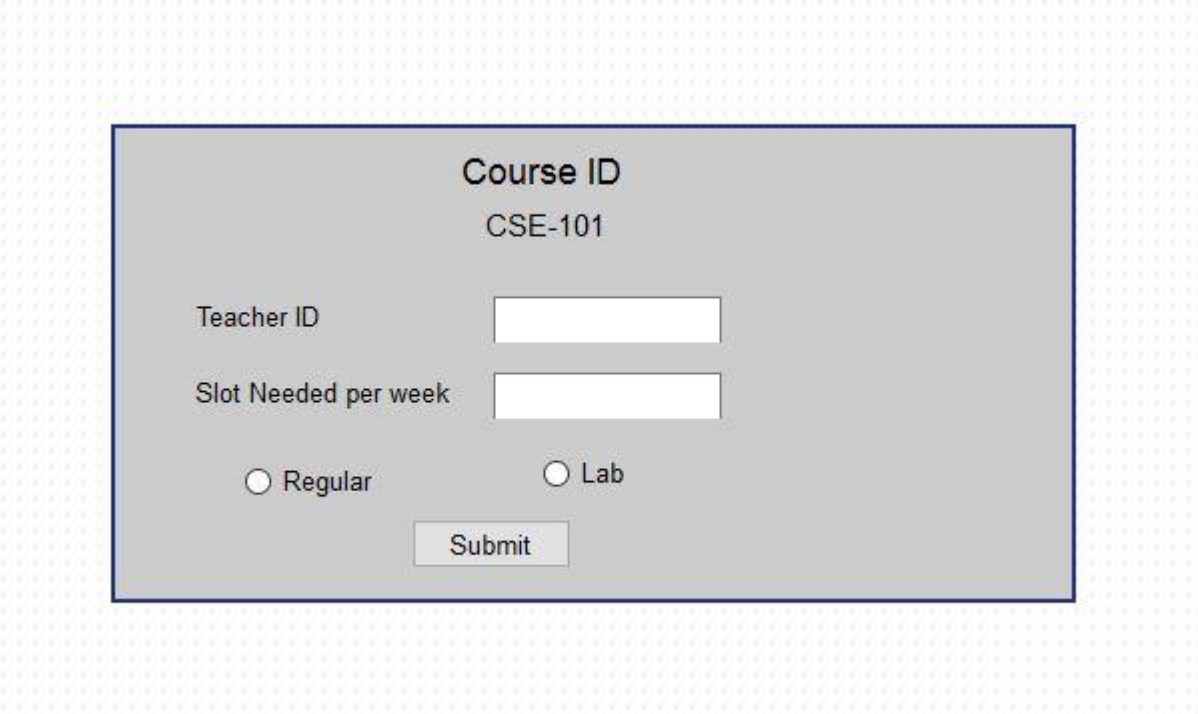

Fig 3: Teacher and Slot

The main interface of this software is the routine interface. A partial snapshot of the routine interface is given below

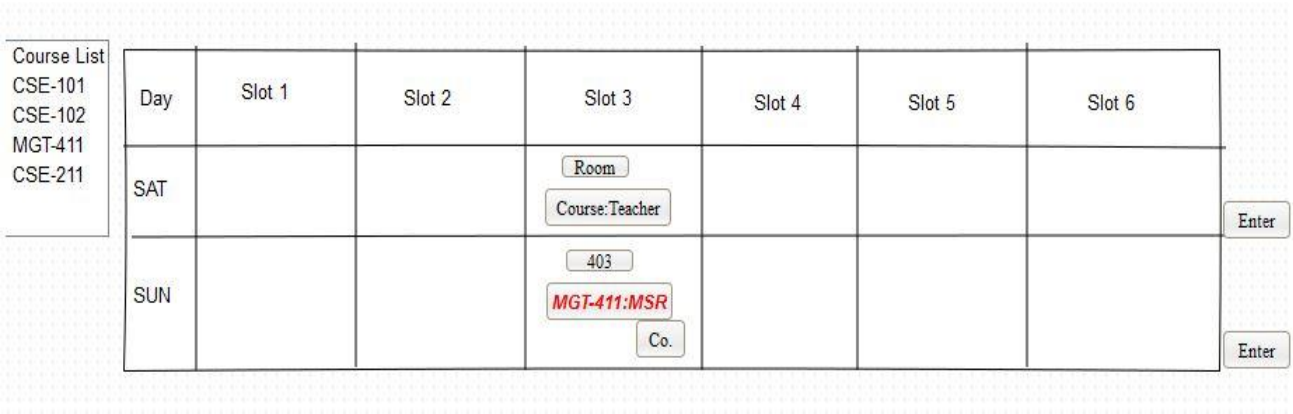

Fig 4: Routine

How the user will interact with the system is given in the functional requirements.

3.2 Functional Requirements

This section includes the requirements that specify all the fundamental actions of the software system.

3.2.1 Functional Requirement 1

ID: FR1

TITLE: Import Data by user

DESC: Click on the import button. In the beginning of a semester, import course id, credit per course, id of the students according to registered courses from an excel file. The excel file format will be expressed later. Clicking on import button will import all data from excel file to database.

RAT: Import primary data from excel file

DEP: None

3.2.2 Functional Requirement 2

ID: FR2

TITLE: Input Information about Teacher and Slot
DESC: After importing necessary information, user will see a button in home screen named “TEACHER AND SLOT”. When He clicks on the button a new window will appear. Here Course id will be auto generated. He has to input teacher’s id who will take the course. And also has to input number of slot needed for any course per week. Ex. A 3 credit course generally has 2 slot of 1.50 hour in one week. After completing information, user presses submit.

RAT: In order for maintaining teacher’s schedule (Fig 3)

DEP: FR1

3.2.3 Functional Requirement 3

ID: FR3

TITLE: Input information of Room

DESC: User will see a button in home screen named “ROOM”. When he click on the button a new window with some dialogue box will appear. The first dialogue box will ask the user about room id. Second box will be the number of students which can be occupied in that room. Third row will have two buttons about “Regular/Lab”. If it is a lab, user has to tick on lab, otherwise has to tick regular class. After completing information, user presses submit. (Fig 2)

RAT: In order for suggesting room in routine.

DEP: None

3.2.4 Functional Requirement 4

ID: FR4

TITLE: Open Routine Window

DESC: User will see a “Create Routine” Button in his home screen. When he clicks on the button, a page will be shown where the course id will be listed on one side. The rest of the screen will be divided into seven day and various slot in one day. Initially the boxes will be blank.

RAT: In order to create a new routine

DEP: FR1, FR2, FR3

3.2.5 Functional Requirement 5

ID: FR5

TITLE: Select and Drop Course

DESC: User can select on any course which is listed beside the screen. When he selects any course, possible drop position will be shown in the routine box. He can drop that course in possible shown position or he can drop in any other box. Possible position will be shown based on no conflict. If he drops the course in any other box and if any conflict occurs, the course will be a different color and a conflict button will appear beside the course id.

RAT: Drag and Drop Course

DEP: FR4

3.2.6 Functional Requirement 6

ID: FR6

TITLE: Change Course

DESC: In order to change any course that is currently on the routine, user will select the course id from the routine box. Again like FR5, possible position will be shown and user can drop the course in any box. If any conflict occurs, the course will be a different color and a conflict button will appear beside the course id.

RAT: Change a course

DEP: FR5

3.2.7 Functional Requirement 7

ID: FR7

TITLE: Select a room

DESC: After completing FR5/FR6, user will be able to see a “room” button appearing beside the course id. When he clicks on the room button, based on the number of student, registered to that course, possible room number will be suggested. User can choose any of that room or can input any other room if it is available.

RAT: In order to give room

DEP: FR5/ FR6

3.2.8 Functional Requirement 8

ID: FR8

TITLE: Create a new row in any slot per Day.

DESC: At the end of each day, other side of the screen, a enter button will be shown. Clicking this button will create a new row in each slot in that day. This option will allow the user to add multiple courses in one slot/day.

RAT: Drop multiple courses in one slot.

DESC: FR4

3.2.9 Functional Requirement 9

ID: FR9

TITLE: Click to show conflict

DESC: After dropping any course, if any conflict occurs, a conflict button will occur beside the course id. When user clicks on the button, a new window will pop up along with the routine in background. The new window will have the type of conflict and details of conflict written.