
Software Requirements Specification

for Exam Schedule Generator

Version 1.0

Prepared by

Group Name:

**Danish Ahmad
Muhammad Zain**

**19f-0170
19f-0228**

**f190170@cfed.nu.edu.pk
f190228@cfed.nu.edu.pk**

Instructor: Dr. Uzair Iqbal

Course: Software Engineering

Lab Section: CS-6A

Teaching Assistant: Zainab Ijaz

Date: 31/05/2022

Table of Contents

1: Introduction	
1.1 Document Purpose	2
1.2 Intended Audience and Document Scope	2
1.3 Acronyms and Abbreviations	3
1.4 Documenting Conventions	3
1.5 References	3
2 Overall Description	
2.1 Project Overview	3
2.2 Design and implementation	3
2.3 Assumptions and Dependencies	3
2.4 Products Functionality	3
3 Specific Requirements	
3.1 External Interface Requirement	4
3.2 Functional Requirement	5
3.3 Use case diagram	5
3.4 Use cases	6
4 Other Non-Functional Requirement	
4.1 Performance Requirement	7
4.2 Safety and security Requirement	7
4.3 Software Quality attributes	8
5 Other Requirements	
Appendix A [Data Dictionary]	8
Appendix B [Group Log]	9

1 INTRODUCTION

INTRODUCTION:

Most Universities these days have several Faculties and Departments, and each Department offers different Degree/ Study Programs. Each Study Program has multiple batches enrolled and different Courses are taught in a semester in different Study Programs. Since there are limited Faculty members (Course Instructors), each Instructor teaches more than one Course and Lecture rooms are also limited. Therefore, an efficient exam timetable is needed to schedule the exams of students at provided time slots in such a way that their timings do not overlap, and the exam timetable schedule makes the best use of all the instructors and lecture rooms.

1.1 DOCUMENT PURPOSE:

In modern educational Institutions, there is a great need to have an efficient exam timetable generation mechanism, as such timetable scheduling is a very hectic job and can have a lot of clashes if done manually.

1.2 INTENDED AUDIENCE AND DOCUMENT SCOPE:

➤ **Audience:**

The roles and (if possible) names of people who are affected by the exam schedule generator, or whose input is needed to build the software.

1. Course instructors (Teachers).
2. Students.
3. Academics Officer.

➤ **Document scope:**

The academic office will generate the exam schedule using the automated exam scheduler by giving all the information about students, teachers, courses, rooms, and available days, using the excel sheet.

1.3 ACRONYMS AND ABBREVIATIONS:

SRS: Software Requirements Specification
CS: Computer Science
MG: Marketing General
DB: Database

1.4 DOCUMENT CONVENTIONS:

Headings: Arial 18 bold.
Sub-headings: Arial 14 italic + underline.
Paragraph Text: Arial 12

1.5 REFERENCES:

Author: Ian Somerville.
Book name: Software Engineering.
Edition: Tenth Edition.

2 OVERALL DESCRIPTION

2.1 PRODUCT OVERVIEW:

With this efficient exam timetable generator, the organization will be able to make an effective, clash-free exam schedule.

2.2 DESIGN AND IMPLEMENT CONSTRAINTS:

➤ **Memory:**

- 1) Course instructor details.
- 2) Student details.
- 3) Room details.
- 4) Courses available.

➤ **Language:**

- 1) It uses the English language.
- 2) Understandable for almost all the users.

➤ **System:**

- 1) Does not need a specific system to operate.
- 2) Just open the software and click the execute button.

➤ **Time:**

- 1) It will give the service 24/7.

2.3 ASSUMPTIONS AND DEPENDENCIES:

➤ **Assumptions:**

- 1) Project must be completed in a measurable period.
- 2) Records of every course instructor, student, room, and date will be given to the system.
- 3) Data will be updated from time to time as a new instructor or student joins the university.

➤ **Dependencies:**

- 1) PyCharm is required to run the software.
- 2) This software will be only accessible by the academic department.

2.4 PRODUCTS FUNCTIONALITY:

- 1) Read data from files. (Teachers, Students, Courses, Rooms)

- 2) Run the software.
- 3) Print the timetable.

3 SPECIFIC REQUIREMENTS

3.1 EXTERNAL INTERFACE REQUIREMENTS:

➤ **User Interfaces:**

There is a quite simple and easy GUI that provides a user-friendly experience as users just have to press the 'generate' button to generate the exam schedule automatically. The software will load the data files to the system from excel sheets. Once this is done, the output of the schedule will display, and the exam schedule will store in the database.

➤ **Hardware Interfaces:**

- Windows 7 or newer.
- Processor: Core 2 due at least.
- Memory: 3 gigabytes at least.

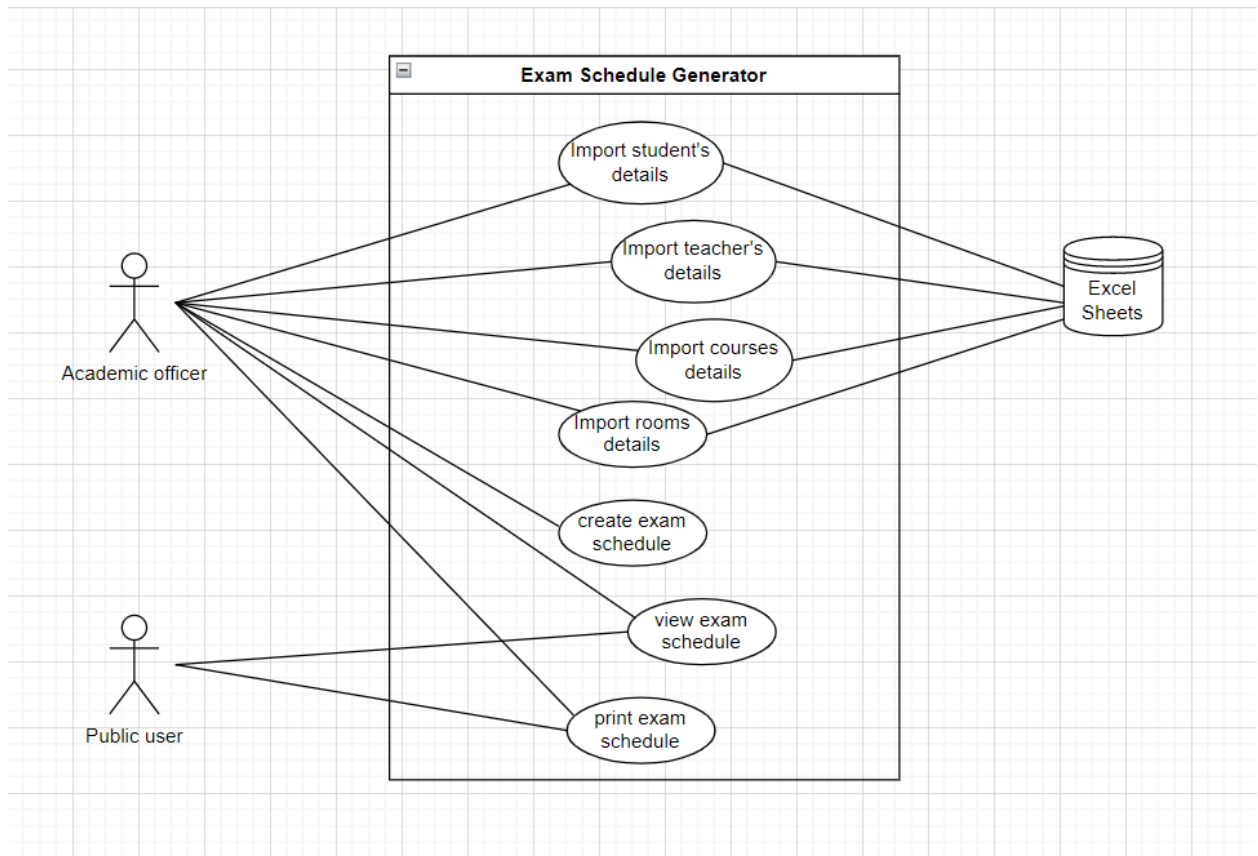
➤ **Software Interfaces:**

- Quite simple and easy interface that shows a "load files" button and "generates schedule" button to the user to load data files from excel sheets and generate an automated exam schedule.

3.2 FUNCTIONAL REQUIREMENTS:

- 1) An exam will be scheduled for each course.
- 2) Exam will not be held on weekends.
- 3) Each exam must be held between 9 am and 5 pm
- 4) A teacher cannot invigilate two exams in a row.
- 5) A student is enrolled in at least 3 courses.
- 6) A student cannot give more than 1 exam at a time.

3.3 USE CASE DIAGRAM:



3.4 USE CASES:

1) U1: (Read data)

Author – Danish Ahmad, Zain Ahsan.

Purpose – To load data from files.

Requirements Traceability – All the required files must be present in an excel sheet.

Priority – High priority.

Preconditions – Data files must be in CSV format.

Postconditions – Data files successfully load in the software.

Actors – Academic officer

Flow of Events

1. **Basic Flow** – Data files are loaded in the software from excel sheets.

2. **Alternative Flow** – Some files are missing while reading from excel sheets.

3. **Exceptions** – Data files got corrupt while loading.

2) U2: (Generate exam schedule)

Author – Danish Ahmad, Zain Ahsan.

Purpose – To generate an exam schedule.

Requirements Traceability – All the exams have been scheduled and follow all the hard constraints that are mentioned above.

Priority – High priority.

Preconditions – Data for courses, teachers, students, and rooms must be available.

Postconditions – A clash-free exam schedule is generated.

Actors – Academic officer

Flow of Events

1. **Basic Flow** – Generate the exam schedule automatically after the files are loaded into a system.

2. **Alternative Flow** – Generate the exam schedule with some clashes.

3. **Exceptions** – Data files got corrupt while loading.

3) U3: (Print exam schedule)

Author – Danish Ahmad, Zain Ahsan.

Purpose – To print an exam schedule.

Requirements Traceability – A clash-free exam schedule has been generated.

Priority – Moderate priority.

Preconditions – A clash-free exam schedule must be generated.

Postconditions – An exam schedule was successfully printed.

Actors – Academic officer, Teacher, Student

Flow of Events

1. **Basic Flow** – Print an exam schedule.

2. **Alternative Flow** – Wait while the printer is busy.

3. **Exceptions** – The exam schedule is not generated successfully.

4 OTHER NON-FUNCTIONAL REQUIREMENTS

4.1 PERFORMANCE REQUIREMENTS:

- Software tries to generate a generic schedule of exams that satisfies almost all the hard constraints.
- Software is responsive to almost every kind of data file to read and write.

4.2 SAFETY AND SECURITY REQUIREMENTS:

- Data files are only accessible to the academic office.
- Academic office can only generate the schedule of exams and only they have access to the software.

4.3 SOFTWARE QUALITY ATTRIBUTES:

- Exam schedule generator is a portable software that can be run on any Windows 7 system or higher.
- Exam schedule generator fulfills all the requirements adequately.
- System is clear and simple as a user manual is provided with it for the best user experience.

5 OTHER REQUIREMENTS

The requirements covered in this portion are soft constraints that the system might adopt:

- All the teachers and students should have a break on Friday from 1 pm to 2 pm.
- If a student is enrolled in an MG course and a CS course, it is preferred that their MG course exam be held before their CS course exam.
- A student shall not have more than 1 exam consecutively.

APPENDIX A – DATA DICTIONARY

Field Name	Data Type	Field Size	Description	Example
Teacher_name	string	50	Contains teacher name	Dr. Ahmad
Student_name	string	50	Contains student name	Habib Khan
Room_no	int	5	Contains room number	10
Course_id	string	10	Contains unique course id	CS120

APPENDIX B – GROUP LOG

Meeting Time	Date	Team Members
10:00 PM - 11:00PM	12 May, 2022	Danish (Manager) and Zain (S.E)
09:30 PM - 11:00PM	20 May, 2022	Danish (Manager) and Zain (S.E)
09:00 PM - 10:30PM	26 May, 2022	Danish (Manager) and Zain (S.E)