



İZMİR EKONOMİ ÜNİVERSİTESİ

Software Requirements Specification for “Exam Timetable”

Group 4

Ece Naz Gökalp 20210602028

Sine Öykü Yaşar 20230602074

Kaan Bilgili 20210602012

Karya Tanrıkul 20230602064

Ayşenaz Gel 20230602027

Ata Kaan Karakuş 20220602045

November 13, 2025

1 Introduction

The "Exam Scheduler" is a desktop application designed to help the student affairs authority generate an exam table during the exam period. The application takes as input a fixed number of courses, the students enrolled in them, and the classrooms that will be used. The primary output is a generated exam schedule.

This document defines the software requirements for the Exam Scheduler System. Its purpose is to clearly outline what and how the system must accomplish its expectations given by the school administrator.

The sections that follow describe the scope of the system, detail the required features, and specify performance, quality, and operational criteria.

2 User Requirements

The user requirements can be detailed as follows:

2.1 Functional Requirements:

Functional Requirement 1: The user shall be able to import necessary data in the form of CSV files.

Rationale: This is the basis for the main functionality of the program, which is to generate exam schedules compliant with the rules specified in later functional requirements. The user needs to give the program required data, which are the CSV files containing a student list, course list, attendance list and classroom list and their capacities for the program to be able to configure the schedule.

Functional Requirement 2: The user shall be able to configure exam period and constraints.

Rationale: This requirement is necessary to ensure that the administrator gets an exam schedule fit to the school needs. There may not be a generated table with the imported files, in that case the user will be able to extend the exam period or change the constraints to get a schedule without the need to re-import the CSV files.

Functional Requirement 3: The user shall be able to review a visualized version of the tables generated by the program.

Rationale : Administrators need a clear and organized way to inspect the generated tables to verify the correctness. Detailed information should be visible about each class and the exams that will be held in them as well as the exam schedule for each student. For each course, the exact time and date of the exam will be stated clearly.

Functional Requirement 4: The user shall be able to update input data by changing the files and re-importing them.

Rationale: If there is an issue with generating the schedule with the files currently imported or if the user wishes to change the schedule for any reason, they should be able to re-import the changed csv files.

Functional Requirement 5: The user shall be able to export the chosen files.

Rationale: The user might wish to share the created schedule with other people, causing the need for an export option.

2.2 Non-Functional Requirements

Non-Functional Requirement 1: The program shall be easy-to-use.

Rationale: The administrator should be able to use all the functionalities of the program after a 3-hour training.

3 System Requirements

The system requirements can be detailed as follows:

3.1 Functional Requirements

Functional Requirement 6: The system shall be able to read the imported CSV files and parse the information correctly. X

Rationale : This requirement is essential because the system relies solely on imported CSV data to create an accurate exam schedule.

Functional Requirement 7: The system shall offer the user a help menu.

Rationale : A help menu is required to help users understand how to use the system. GUI

Functional Requirement 8: The system shall ensure the number of exams a student has in one day does not exceed two.

Rationale : Limiting students to maximum two exams per day helps to avoid excessive overload. ALG

Functional Requirement 9: The system shall generate a schedule where a student does not have two consecutive exams in a day. ALG

Rationale: To ensure students have an adequate break between exams for rest, mental preparation, and logistical travel (e.g., moving between different buildings or classrooms). This prevents the undue stress and performance disadvantages associated with continuous back-to-back testing.

Functional Requirement 10: The system shall ensure that classroom capacities are not exceeded. ALG

Rationale: Each classroom has a set amount of seats and desks. In order to prevent any students or instructors from experiencing panic because of there being not enough room in the classroom, the system should appoint an appropriate amount of students to a classroom.

Functional Requirement 11: The system shall detect and prevent conflicts for student exam schedules. ALG

Rationale : It is crucial to identify and prevent scheduling conflicts. This feature ensures that no student is assigned to conflicting exams and prevents operational issues throughout the exam period.

Functional Requirement 12: The system shall show the user the created schedule in the software's user interface. GUI

Rationale : This function is important to provide immediate visual feedback and validation. The user must be able to review, assess, and interact with the generated schedule within the application.

Functional Requirement 13 : The system shall allow exporting the final schedule. X

Rationale : This helps seamless integration with existing management processes, automated reporting, and efficient analysis of the workflow.

3.2 Non-Functional Requirements

Non-Functional Requirement 2: The system shall be able to run on a Windows system.

Rationale : The target platform is Windows since Windows is the required OS within the organization.

Non-Functional Requirement 3: In case of unexpected errors, the system should show a clear error message rather than crashing.

Rationale : This requirement is important to ensure system stability and usability. A sudden crash frustrates the user and can cause data loss, whereas a clear message explains the problem and allows the user to take action.

Non-Functional Requirement 4: The system shall offer a GUI with high usability, ensuring that all functions are easily accessible.

Rationale : Offering a highly usable GUI ensures that all functions are easily accessible, reducing user errors and improving task performance.

Non-Functional Requirement 5: The system shall be in English.

Rationale : The system shall use English as the language since the education at the institution is delivered in English.