Laboratory Assignment AND Assessment Requirements Specification

Version 1.0

20 March 2024

Developed by:

Iulia Groza

Robert Harangus

933/1,

Version History

Version	Description of Change	Author	Date
V01	Initial/Modification of document	Iulia Groza	18 March 2024
V02	Completion of document	Robert Harangus	20 March 2024

Contents

1. Introduction

The application is written in Java and is designated for teachers to be able to assign homework & assignments to students and keep track of them.

1.1.Purpose

The application allows the user to easily manage a list of students, assign assignments to them, mark grades for each assignment, extend deadlines and manage delays.

1.2.Scope

The scope of the document is to give information about the system: regarding the users, functionalities, purpose, usability, data management and user scenarios.

1.3. Definitions, Acronyms, and Abbreviations

MAP = Metode Avansate de Programare

FR = Feature Requirement

1.4. Document Overview

The document is organized in chapters and subchapters describing the general purpose and scope of the document, the product description and requirements such as functional and user interface requirements.

2. Product/Service Description

The application allows the user to add students and assignments. Also, he/she can assign a mark for a student on an assignment. The user can modify information about a student at any time or delete the student from the list of students.

2.1.Product Context

The product is independent and self-contained.

2.2.User Characteristics

Users that will be using this product are university teachers teaching the MAP subject.

3. Requirements

When the program starts, the input data is read from the following text files:

- Students.txt file, which contains information about idStudent (student's number), name, group, email, name of the professor.
- Assignment.txt file, which contains the following information: laboratory number (unique identifier), brief description of the requirement, deadline – the week of the semester in which the assignment should be delivered (1-14), the week in which the assignment was received (1-14).

The teacher should be able to see all students and assignments, add a student or an assignment, delete, find or edit any information about a student.

Also, the teacher can assign a grade from 1 to 10 for each assignment. Each week of delay will be penalized by 2.5 points.

An assignment can be delivered at most 2 weeks after its deadline, otherwise it will be marked with 1.

The file Catalog.txt will store information about the grades. For each grade, it will contain the id of the grade, the name of the student, the lab id and the value of the mark given.

3.1.Functional Requirements

List the functional requirements (FR) of the system.

Section/ Requirement ID	Requirement Definition
FR1.0	Implement CRUD operations for the Student entity
FR1.1	Add student
FR1.2	Edit information about a student
FR1.3	Delete student
FR1.4	Show all students
FR1.5	Find student
FR2.0	Manage laboratory assignments and subjects.
FR2.1	Add laboratory assignment
FR2.2	Edit information about a laboratory assignment
FR2.3	Delete laboratory assignment
FR2.4	Show all laboratory assignments
FR2.5	Extend the deadline for an existing subject
FR3.0	Manage grading
FR3.1	Assign a grade to a student on an assignment
FR3.2	Edit a grade to a student on an assignment
FR3.3	Delete a grade to a student on an assignment
FR3.4	Show all grades
FR3.5	Find grade of a student on an assignment
FR3.0	Filter students based on different criteria
FR4.0	Generate reports
FR4.1	Generate reports of laboratory grades for each student
FR4.2	Generate reports for hardest assignments
FR4.3	Generate reports of students who can enter the exam
FR4.4	Generate reports of students who delivered all assignments on time
FR5.0	Notify students by email

Section/ Requirement ID	Requirement Definition
FR5.1	Notify students by email when adding a new laboratory assignment
FR5.2	Notify students by email when modifying the delivery date of an existing assignment

3.2. User Interface Requirements

The user should be presented a menu where each option is describing one of the functional requirements. After choosing an option the program should ask the user to enter the needed information.

3.3. Usability

- The user documentation and help should be complete.
- The help should be context sensitive and explain how to achieve common tasks.
- The system should be easy to learn.

3.4. Data Management

The data should be stored in text files.

4. User Scenarios/Use Cases

The application allows the user to:

- Add student
- Edit information about a student
- Delete student
- Show all students
- Find student
- Add laboratory assignment
- Edit information about a laboratory assignment
- Extend a deadline for an assignment
- Delete a laboratory assignment
- Show all assignments
- Assign a grade to a student on an assignment
- Edit a grade to a student on an assignment
- Delete a grade to a student on an assignment
- Show all grades
- Find grade of a student on an assignment
- Filter students based on different criteria
- Generate reports of laboratory grades for each student
- Generate reports for hardest assignments
- Generate reports of students who can enter the exam
- Generate reports of students who delivered all assignments on time
- Notify students by email when adding a new laboratory assignment

- Notify students by email when modifying the delivery date of an existing assignment

Please refer to the Analysis and Design Document.