Laboratory Assignment AND Assessment Requirements Specification

Version 1.0

March, 2022

Developed by:

Tirdea Razvan, Tamas Timotei

937

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Description of Change | Author | Date |
| V01 | Initial | Tirdea Razvan  Tamas Timotei | 20.03.2022 |

**Contents**

[Laboratory Assignment AND Assessment Requirements Specification 1](#_heading=h.gjdgxs)

[Version 1.0 1](#_heading=h.30j0zll)

[March, 2020 1](#_heading=h.1fob9te)

[1 Functional Requirements 3](#_heading=h.2et92p0)

[2 Actors 3](#_heading=h.tyjcwt)

[3 Use cases – diagram 3](#_heading=h.3dy6vkm)

[3.1 Use case number 1 (Description of the use case) 3](#_heading=h.1t3h5sf)

[4 Analysis 3](#_heading=h.4d34og8)

[4.1 Entities 3](#_heading=h.2s8eyo1)

[4.2 Relations between entities 3](#_heading=h.17dp8vu)

[4.3 Attributes 3](#_heading=h.3rdcrjn)

[4.4 System behavior 4](#_heading=h.26in1rg)

[4.4.1 Use case 1-2-3 4](#_heading=h.lnxbz9)

[4.5 System events 4](#_heading=h.35nkun2)

[5 Design 4](#_heading=h.1ksv4uv)

**Analysis and design Document**

# Functional Requirements

List the functional requirements (FR) of the system.

|  |  |
| --- | --- |
| Section/ Requirement ID | Requirement Definition |
| F0 | Implement CRUD operations for the student entity |
| F1 | Adding a laboratory theme |
| F2 | Extending the term of delivery for an existing subject |
| F3 | When adding a new laboratory theme, as well as modifying the delivery date of a theme, all students will be notified by email. An unsubscribe option will be provided by the application. |
| F4 | Adding a grade for a particular student to a laboratory topic; any delays due to delays in delivery of a theme will be automatically calculated, showing the student's maximum mark on the topic. Important: A student, on a laboratory theme, has only one grade; |
| F5 | When adding a grade, the following information will be retained in the NameStudent.txt file: a "Theme:" ThemeNumber  • "Delivered in the week:" NumberOfTheDeliveredWeek  • "Deadline:" NumberOfDeadlineWeek  • "Feedback:" feedback, suggestions, and explanations in connection with the reduced made regarding the grade. |
| F6 | The NameStudent.txt file (or its content) will be emailed to the student weekly. |
| F7 | The delays will not be considered if the student has motivation. Also, if the teacher did not enter the notes in time, it will be possible to specify the week in which the subject was delivered. |
| F8 | Filtering entities based on criteria. |
| F9 | Various reports can be generated. |

# Actors

Teacher

# Use cases – diagram



## Use case number 1 (Description of the use case)

Actors: teacher

Description: create a new student

Precondition: - all fields are specified

Postcondition: - a new student was added in the list

|  |  |
| --- | --- |
| Action | System Response |
| Sssss Select add a new student command and fill required data | Check the validity of the requested data |
|  | If the data is valid the system saves the student |
| 3 - | Returns an appropriate message (error / saved ) |

Exceptions: When the fields aren’t filled.

## 3.2 Use case number 2 (Description of the use case)

Actors: teacher

Description: delete student

Precondition: - valid id belonging to an existing student is specified

Postcondition: - the student with the specified id is removed from the list

|  |  |
| --- | --- |
| Action | System response |
| Select the delete student command and pass the student id | Checks if the provided id is not null |
|  | Deletes the student if they exist |
| 3 - | Returns an appropriate message (success/error) |

## 3.3 Use case number 3 (Description of the use case)

Actors: teacher

Description: update student

Precondition: - valid id belonging to an existing student and all other fields for student are specified

Postcondition: - the student with the specified id has the data updated

|  |  |
| --- | --- |
| Action | System response |
| S Select the update student command and input the requested data (id, new name, new group) | Checks if the requested data is valid |
|  | Updates the student if they exist |
| 3 - | Returns an appropriate message (success/error) |

# Analysis

## Entities

Student, Theme/Assignment, Grade

## Relations between entities

One student can have multiple assignments and one assignment can be assigned to many students. It is a many-to-many relationship between the two classes and the grade entity is characterized by being unique per student+theme combination.

## Attributes

Student: id, name, group

Assignment: id, description, deadline, startline

Grade: id(studentId, assignmentId), value, deliver date, feedback

## System behavior

## Use case 1-2-3

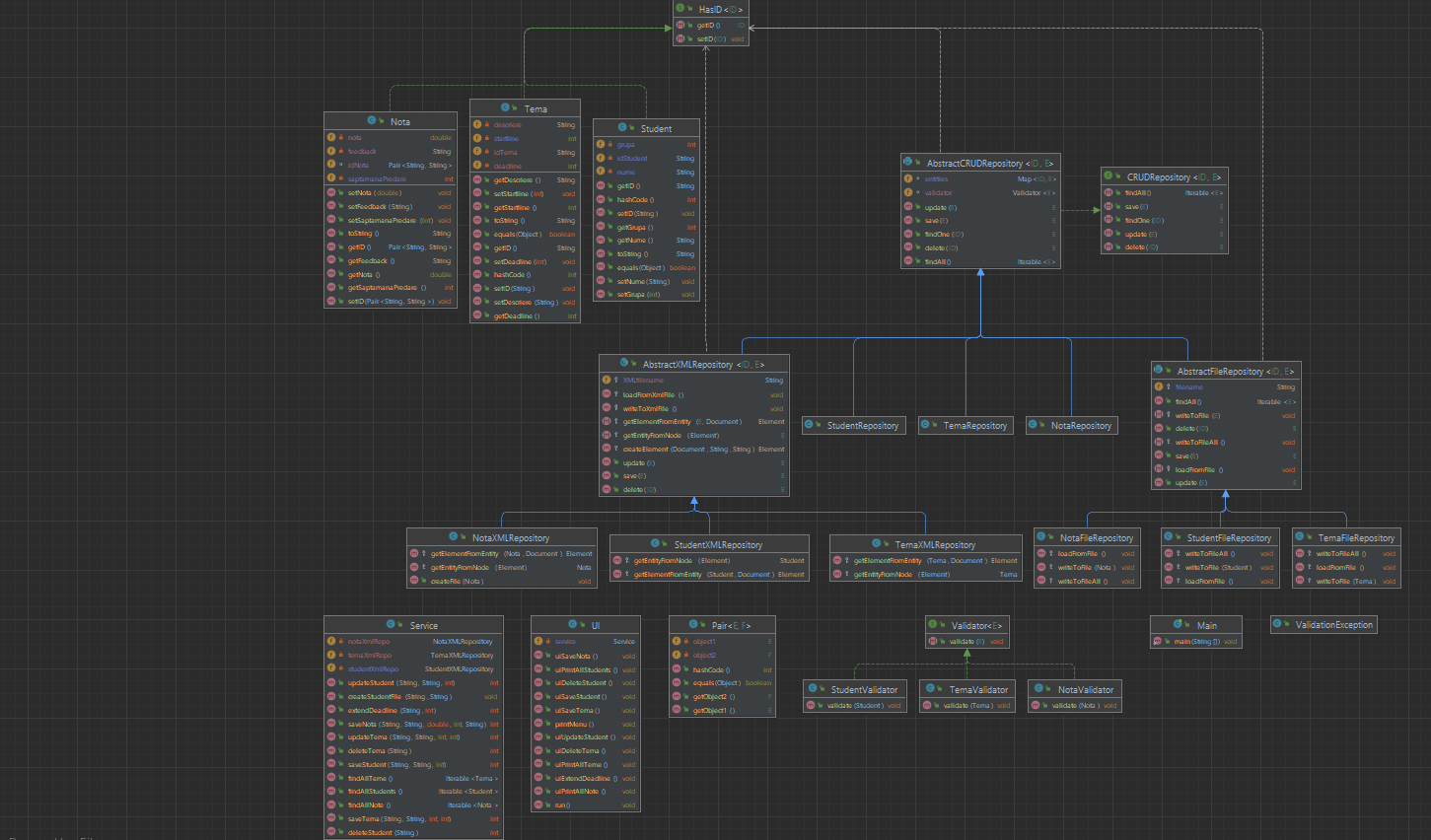
The system will act as a subsystem to a larger environment, in order to speed up a certain process in the company’s workflow.

## System events

After each operation a message is shown to the user either if the command terminated successfully or with an error message.

# Design

* 1. **Class diagram**



* 1. **Sequence diagrams (for each use case)**
* **Add Student Sequence Diagram**

****

* **Delete Student Sequence Diagram**

****

* **Update Student Sequence Diagram**

****

* 1. **GRASP**

GRASP is set of exactly 9 **G**eneral **R**esponsibility **A**ssignment **S**oftware **P**atterns:

1. Information Expert

2. Creator

3. Controller

4. Low Coupling

5. High Cohesion

6. Indirection

7. Polymorphism

8. Pure Fabrication

9. Protected Variations