Overview

Overview allows the user (a **Student**) to see how the semester evolves.
It keeps track of what has already be done in each course the user is registered in.
A **course** consists of many **Lecture**s. The exact number of Lectures is not always
 knowns in advance.
A course can have many **Labor** requirements.
Each labor can be marked as successful or or as a failure.

A course can have **Homework**s too.
For each course there is a corresponding **Exam**.
Each Exam has its **requirement**s which could be base on Homework or on Labor.
A Student have a list of course he takes on a given.
The app should let the user **add** new data every time the user needs to.
The user should be able to **modify** any Data at any time.

Domain Model :

Student Professor Course Labor	Homework Exam Requirement Lecture

Use Cases :
Administration Use Case:

Add a Student :
Basic Course :
The User adds a new Student. He gives the Student a name , he set the faculty. The System then checks whether there is already a Student with given name. The System adds the Student and set a new ID to the student.

Alternate Course:

- There is already a Student with that name. The System tells the User what is wrong and wait for another request.
- The name and / or the faculty is empty. The System tells the user what is wrong and wait for the next request.

Add a Course:

Basic Course:
The User tells the System he wants to add a new course. The System asks then after the course's name. The User enter the name.
The System asks then if the course will have Homework.
The User answer with Yes or No
The System then asks if the course will have Labor work.
The User answer with Yes or No.
The System asks then the type of requirements the course has to be able
to take part to the Exam and displays a list of Requirements
The User choose a Requirement type and validate his choice.
The System creates the new course and wait for the next request.

Alternate Course:

- The User enter an empty name. The System displays an error message and wait for the next requests.

- The User enter a name of a course already present.
  The System displays an error message and wait for the next requests.

Add Homework:

Basic Course:
The User tells the System he wants to adds a homework.
The System shows the User a list of Courses and asks him to choose a course into
which he wants to add the homework with the possibility to cancel the action.
The User chooses a Course. The System then asks the user to set the deadline for this homework.
The System then adds the Homework to the database. When a new Homework is added, it is marked as undone and unsuccessful.

Alternate Course:
- The user cancel the action. The System then stops and wait for the next requests.
- The user choose a date already defined for this course. The System displays an error message and wait for the next requests.

Add Labor:

Basic Course:
The User tells the System he wants to adds a labor
The System shows the User a list of Courses and asks him to choose a course into
which he wants to add the labor with the possibility to cancel the action.
The User chooses a Course. The System then asks the user to set the deadline for this labor.
The System then adds the labor to the database. When a new labor is added, it is marked as undone and unsuccessful.

Alternate Course:
- The user cancel the action. The System then stops and wait for the next requests.
- The user choose a date already defined for this course. The System displays an error message and wait for the next requests.

Add Exam :
Basic Course :
The User tells the System he wants to adds an Exam .The System shows the User a list of Courses and asks him to choose a course into
which he wants to add the Exam with the possibility to cancel the action.
The User chooses a Course. The System then asks the user if he wants to set the deadline for this Exam. The User accepts and set the Date.
The System displays a list of Requirements and asks the user to set the Requirement for this Exam.
The System then adds the Exam to the database. When a new Exam is added, it is marked as undone and unsuccessful.

Alternate Course:

- The user cancel the action. The System then stops and wait for the next requests.
- The user chooses a Course whose the Exam has already been set. The System displays an error message and wait for the next requests.

Add Lecture :

Basic Course :
The User tells the System he wants to adds a Lecture. The System shows the User a list of Courses and asks him to choose a course into
which he wants to add the Lecture with the possibility to cancel the action.
The User then adds the Lecture's Title and the System saves the Lecture into the database.

Alternate Course :

- The user cancel the action . The System wait then for others requests.
- The user enters an empty Title. The System displays an error message and wait for the next requests.

Reset :
Basic Course :
The user chooses to delete everything that was saved and start new. The System then asks for confirmation before reseting.

Alternate Course : The user chooses to cancel . The System does nothing and wait for another request.

Checking Use Case

Check The Labor / Homework / Exam / Course State :
Basic Course :
The User asks the State the labor of a course : The System displays a list of Course to choose from.
The user chooses a course and then the System print an overview of the labor / homework / Exam / Course from that course.

Alternate :
The User cancels the action. The System does nothing and wait for the next requests.

General Use Case :

Get The list of Course :
Basic Course :
The user asks to the System the list of Course . The System then replies with list of saved Courses.

Alternate Course :

- The User cancels the action. The System does nothing and wait for the next requests.

Get The list of Labor / Homework / Exam / Lecture :
Basic Course :
The user asks to the System the list of Labor / Homework / Exam / Lecture. The System then displays a list of saved Courses
and asks the user to choose the Course for which he want the list of Labor / Homework / Exam / Lecture.
The System then replies with the list of Labor / Homework / Exam / Lecture.

Alternate Course :

- The User cancels the action. The System does nothing and wait for the next requests.

Get The list of successful Labor / Homework / Exam :
Basic Course :
The user asks to the System the list of Labor / Homework / Exam . The System then displays a list of saved Courses
and asks the user to choose the Course for which he want the list of successful Labor / Homework / Exam. The System then replies with the list of successful Labor / Homework / Exam .

Alternate Course :

- The User cancels the action. The System does nothing and wait for the next requests.

Get Student Name / Faculty :

Basic Case :
The User asks the Student's Name / Faculty to the System. The System then replies with the Student's Name / Faculty.

Alternate Course :
There is no registered Student : The System displays an error message and wait for the next requests.

Administrative Use Cases:

Add A Course :
Basic Course :

Alternate Course:

Remove a Course :
Basic Course :

Alternate Course:

Add Homework:
Basic Course :

Alternate Course:

Remove Homework:
Basic Course :

Alternate Course:

Add Labor:
Basic Course :

Alternate Course:

Remove Labor:
Basic Course :

Alternate Course:

Add Lecture:
Basic Course :

Alternate Course:

Remove Lecture:
Basic Course :

Alternate Course:

Add Requirement:
Basic Course :

Alternate Course:

Remove Requirement:
Basic Course :

Alternate Course:

Add Student:
Basic Course :

Alternate Course:

Remove Student:
Basic Course :

Alternate Course:

Add Exam
Basic Course :

Alternate Course:

Remove Exam:
Basic Course :

Alternate Course:

Reset :