**Analysis**

**Jan Lishak – 294322**

**Lenka Orincakova - 293085**

**Juan Iglesias Trebolle – 293143**

**João Bernardo Baptista Vieira Dias - 293133**

**Software Technology Engineering**

**1st Semester**

**30th October 2019**

Supervisor: Astrid Hanghøj, Michael Viuff | Number of words :

# Summary

VIA University College needs to plan exam schedule before the examination starts. The schedule plan must meet many specific criteria. Scheduling in general is simple task but when there are so many conditions for the right plan, the task can become very complex. Computers might possibly help to speed up the process of scheduling.

Our system is able to do: 1) the administrator collects the information, 2) the administrator is uploading the information and system checks the requirements, 3) the schedule is created.

The system contains information about students, co-examiners and examiners. The schedule must be flexible and delivered to the students in a way that the information is always updated and easy to access. It would be better if the schedule could be viewed by a student or a teacher in a way that it filters all irrelevant information.

**Table of content**

[Summary 2](#_Toc23339833)

[Functional requirements 4](#_Toc23339834)

[Non - functional requirements 5](#_Toc23339835)

[Use Case Diagram 6](#_Toc23339836)

[Use Case Description 6](#_Toc23339837)

[Link between requirements and use cases 8](#_Toc23339838)

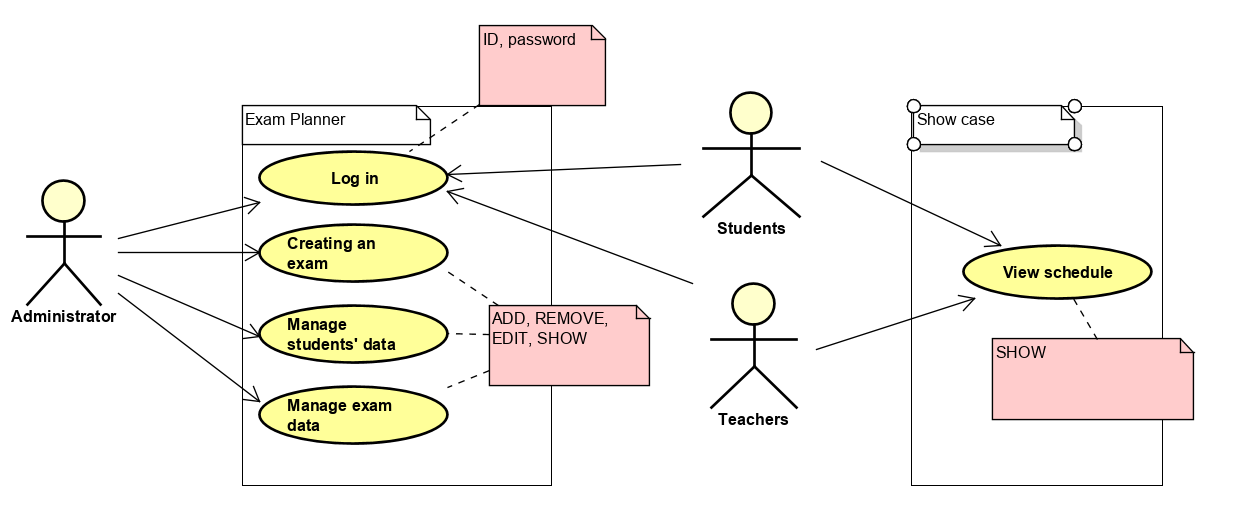
# Functional requirements

1. As an administrator, I want to be registered as only administrator with access to all information.
2. As an administrator, I want to be able to upload information about teachers, students and classrooms.
3. As an administrator, I want the system to check if there are two exams in the same day, with the right amount of time between them.
4. As an administrator, I want the system to check if the rooms are suitable for exams and if the examiners and co-examiners are available.
5. As an administrator, I want the system to check that the examiners’ and co-examiners’ work day is 8 hours max.
6. As an administrator, I want the system to check if the written exam is before oral exam.
7. As an administrator, I want the system to check if the time of the written exam is 3 hours and the time of the oral exam is 20 minutes.
8. As an administrator, I want the system to check if the exam on 7th semester is at least 3 days before graduation.
9. As an administrator, I want the system to check if the class for the exam is the same as class used during the semester.
10. As an administrator, I want the system to check if the service department is informed about the exam at least one day in advance to prepare the room for exam.
11. As an administrator, I want to be able to add, remove, edit and show information about exam.
12. As an administrator, I want to be able to add, remove, edit and show information about student.
13. As a student, I want to be registered as student with access to the schedule.
14. As a student, I want to be able to see the schedule.
15. As a teacher, I want to be registered as a teacher with access to the schedule.
16. As a teacher, I want to be able to see the schedule.

# Non - functional requirements

1. Every update in the system includes writing to a file.

# Use Case Diagram



# Use Case Description

|  |  |
| --- | --- |
| **Use case** | **Log in** |
| **Summary** | Administrator, students and teacher can enter in the system. |
| **Actor** | Administrator, students, teachers. |
| **Precondition** | The administrator, students and teacher have a VIA ID to log in. |
| **Postcondition** | System recognizes ID and view the schedule. |
| **Base sequence** | 1. System asks for VIA ID and password. 2. If the ID equals administrator’s ID, system allows him to manage all the data (add, remove, edit, show). 3. Else, the system only shows the schedule (show). |

|  |  |
| --- | --- |
| **Use case** | **Creating an exam.** |
| **Summary** | Check if all requirements for a specific exam are met. |
| **Actor** | Administrator |
| **Precondition** | Administrator creates an exam. |
| **Postcondition** | Exam is created with all necessary information. |
| **Base sequence** | 1. Enter the date of exam. 2. If the date is not in the exam period (January, June), go to step 1. 3. The system shows every program which is taught at VIA UC. For each program, teachers, students, rooms are set. Choose the program for exam. 4. Each program is divided in several semesters. Select the semester you want to have an exam in. 5. Every course from the semester is shown. Choose the course (SDJ, RWD, MSE, etc.) 6. System filters the classes. Choose the class for exam (X, Y, Z, etc.). 7. Choose the type of exam. The system will show the classroom which is assigned to specific class (E 301). If the exam is of type written and is before oral exam, list all the rooms which minimum size is 40 and list all the features of the classroom (HDMI, etc.). 8. If the specific classroom is not available, go to step 7 and choose another one. 9. Approve the selected exam or go to step 7 to select another. 10. System shows a list of examiners. Choose the examiner. 11. If the examiner has reached maximum hours (8 hours) and is not available on the specific date, go to step 10 to choose another one. 12. Approve the selected examiner or go to step 10 to select another. 13. System shows a list of co-examiners. Choose the co-examiner. 14. If the co-examiner has reached maximum hours (8 hours) and is not available on the specific date, go to step 13 to choose another one. 15. Approve the selected co-examiner or go to step 13 to select another. 16. The exam is uploaded with assigned date, examiner, co-examiner, time, room and list of the students. 17. If the exam is oral, the order of students is uploaded. |

|  |  |
| --- | --- |
| **Use case** | **Manage students’ data** |
| **Summary** | Refreshing information about students. |
| **Actor** | Administrator |
| **Precondition** | System contains all the information. |
| **Postcondition** | Information about students is refreshed. |
| **Base sequence** | System shows current information.  ADD:   1. If adding new student, then set up this information:    1. Name (two strings)    2. ID    3. Program (whole name)    4. Class (just letters)    5. Email (in the format “id@via.dk 2. System validates data and prompts for illegal values. 3. If the input is valid then the system adds a new student with given information.   End the use case for ADD  SHOW:   1. System shows a list of students all information from step 1.   EDIT:  Step 1-4 as above and then   1. Enter or edit values for one of the student fields from step 1. 2. System updates the selected field in the student.   End the use case for EDIT.  REMOVE:  Step 1-6 as above and then   1. Verify deleting the selected student. 2. System approves the deletion and removes the student from the list.   End the use case for REMOVE. |

|  |  |
| --- | --- |
| **Use case** | **Manage exam data** |
| **Summary** | Refresh data when information is changed. |
| **Actor** | Administrator |
| **Precondition** | System contains all the information. |
| **Postcondition** | New exam is uploaded, and new information is added. |
| **Base sequence** | ADD:   1. If adding new exam, then set up this information:    1. Date (exam period)    2. Program (Software Engineering, etc.)    3. Semester    4. Class    5. Course    6. Type of exam    7. If the type of exam is written   then also features   * 1. Room   2. Examiners   3. Co-examiners  1. If the date and time steps over existing date and time, go to step 1 and choose another date and time. 2. If the semester is 7th and the date is not three days before graduation, go to step 1 and choose another date. 3. System validates data and prompts for illegal values. 4. If the input is valid then the system adds a new exam with given information.   End the use case for ADD  SHOW:   1. System shows a list of exams with all information from step 1.   EDIT:  Step 1-4 as above and then   1. Enter or edit values for one of the exam fields from step 1. 2. System updates the selected field in the exam.   End the use case for EDIT.  REMOVE:  Step 1-6 as above and then   1. Verify deleting the selected exam. 2. System approves the deletion and removes the exam from the list.   End the use case for REMOVE. |

|  |  |
| --- | --- |
| **Use case** | **View schedule** |
| **Summary** | Showing the schedule to students and teachers. |
| **Actor** | Students, teachers |
| **Precondition** | ID was recognized as student’s or teacher’s ID. |
| **Postcondition** | Exam schedule is shown. |
| **Base sequence** | 1. System recognizes ID of the actor. 2. Exam schedule is shown. |

# Link between requirements and use cases

**Use case Covered requirements**

Log in 1,13,15,

Creating an exam 2,3,4,5,6,7,8,9,10

Managing students’ data 12

Manage exam data 11

View schedule 14,16