Code Report

First of all, we had to create a project in C# language about an application which manage all the administration of a University. The goal of this project is to use all the competence that we learned during this semester and also competence that we already had. We were distributed in group of 5 to 7 students. Our team is the number 14 and is composed by Yoan Gabison Baptiste Gaidou Clarysse Gielen Paul Froidefond Quentin Navarre and Eliott Tourtoulou.

We’ll recap our work in this report.

1. Imposed Task

We had few mandatory tasks to do in our code but also around the code for example this report or a ULM representation of our strategy.

a. Mandatory implementation

The assignment precise 4 task that we must do:

1. Administrator Login with Super Privileges for creating courses, timetables etc.

We implement it in an inherited class of User with some specific privilege only for Administrator. You can see it on Line 964 in the class Application, the function called HomePage\_Administrator()

1. Faculty Login with Privileges pertaining to their students

Faculty is also an inherited class of users but this one can manage some attribute of his class or his group of students, and also manage exam’s marks etc… Furthermore, this faculty class is less omnipotent than the admin one. All is set up from line 1713 in the Application class, the function called HomePage\_Faculty() you can see how to see detail pertaining to their student in the class Faculty with the function called MenuSeeStudentInformation line 61

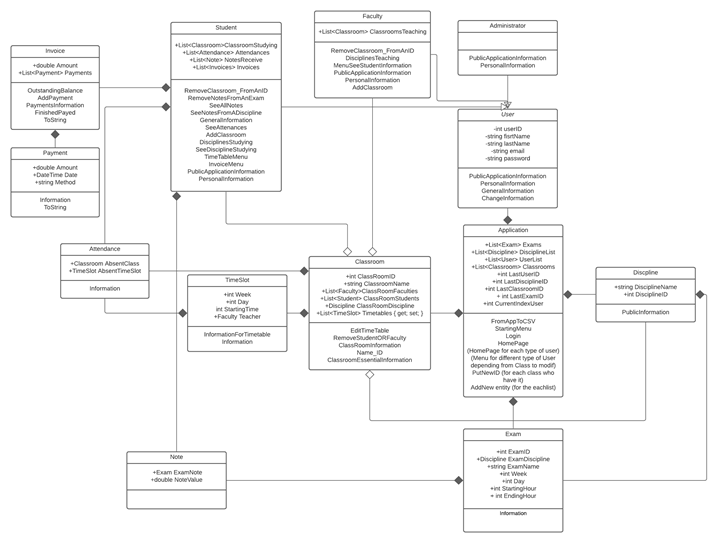
1. Student Login with privileges that only permits the to see details pertaining to them.

Faculty is also an inherited class of users to see what they can do it’s start at the line 910 of the class Application by the function HomePage\_Student, there are some class in the student that is called by this function to see details pertaining to them. there.

1. Manage Student Registration and Enrolment in courses.

All the part of Enrolment can be managed by the Administrator Users, you can see it in the Application class, around line 1316 by the function ClassRoomMenu\_Administrator, when a student is in a classroom, he is now enrolled to the discipline of the classroom

b. ULM Representation



Here, the dependence between different class is highlighted. For example, in the top of the diagram we can see all the structure of Student Faculty and Administrator Cass, all inherited from the User one.

2. Task that we choose

a. Manage Exams, Assignments and Results.​

We implemented this point as a faculty and administrator privilege as we can read it from line

Exams/Assignment : line 1009 of application class by the function ExamMenu\_Administrator, so it’s make by an adminstrator user

Results : line 1732 in the function HomePage\_Faculty in application class the it’s in the case 4 of the fisrt switch, so it’s make by a faculty user

b. Manage Student Attendance.​

We also script the attendance management by writing a dedicated class to it. For manage it’s in class Classroom that change the attribute List<Attendance> by the function EditTimeTable line 15 only an adminastator can acceed to it, the function is called by the ClassRoomMenu\_Administrator function in application class.

c. Manage Time Table.​

Managing timetable is working with 2 class called TimeSlot and TimeTableAffichage. This second class is only here to create a neat and visible Timetable on the console. There are many function created to make the display of the timetable for the student, it is in the Student class and start from the function TimeTableMenu at the line 291 and it’s called inside a function in the application class : HomePage\_Student at the line 926 and only accessible by a student

d. Manage of university’s fees

Furthermore, we implement the management of university’s fees. Actually, as the assignment ask us to complete 3 tasks of our choose, fees was only a bonus for us but we decide to go to the end of the implementation because we found it interesting to do.

There are 2 class for this Payment and Invoice and it’s manage inside application class by an administrator at the line 1220 by the function UserMenu\_Administrator where you can add fees. And a student can manage to pay it by the function AddPayment from the invoice class at line 56 , called by in the student class at line 343 in the function InvoiceMenu.