|  |  |
| --- | --- |
| **Project title** | **Unis** |
| **Author(s)** | **Mosolea Victor, Olaru Sebastian, Stretea Cosmin** |
| **Group** | **30422** |

# Task Description

**Short Description:** The purpose of this Java application is to make the interaction between students, professors and secretaries easier. It does this by giving them a platform where they can all see and edit relevant information regarding the educational process, therefore getting rid of some of the bureaucracy that we all encounter on a daily basis.

**Students features:**

* create an account
* see their academic situation
* see their exam schedule
* see and pay taxes for the University

**Professors features:**

* schedule exams for the subject they teach
* see the exam schedule for their students so they can take it into account when scheduling their own
* set and modify the grades for their students
* upload and see their wealth declaration (in .png format)

**Secretaries features:**

* see, edit and edit student data
* delete students from the university’s database
* approve exams that have been scheduled by the professors
* add new taxes
* assign taxes to students

**Start Page/Login Page:** Here all of the categories mentioned above (students, professors and secretaries) can log into their accounts. Users can also create a new account, which by default will be a student account.

**Create account Page:** The purpose of this page is to allow students to create an account. The new user must fill all the fiels displayed on the screen. Each user must have an unique username, therefore, if the person creating the account chooses an already existing username, they will be prompted with a warning and they must input a new one. The form for creating a new account also checks for the *password* field and the *confirm password* to match. If they don’t match, again the user will be prompted and they will have to try again. After the new account is created succesfully, the user is redirected to the start page where they can now login with their newly created account.

**Student Pages:**

* **Academic Situation:** Here the students can see all of their grades, and an overview of their academic situation
* **Exam Schedule:** On this page the users can see their exam schedule and for each exam whether the exam is approved or not. Exams are approved by secretaries and after doing that the specific date of the exam becomes official.
* **Taxes:** This page allows the students to see all of the taxes they have to pay and can pay them by using the button placed next to the tax name.

**Professor Pages:**

* **Exam Schedule:** Now this page acts in a different way because the professors can also add exams by clicking the plus button which will open a new window where they will have to choose the date for their exam. The whole exam schedule is displayed so that professors can take into account all other exams when scheduling theirs.
* **Catalog:** On the catalog page the professors can see all of the enrolled students and by clicking on one of them they can further either add their grade or modify the existing one.
* **Wealth Declaration:** Here professors can add their wealth declarations by choosing and existing .png file on their computer and they can also see a preview of their already stored declaration.

**Secretary Pages:**

* **Student Database:** On this page the secretaries can see a list of all students on the left and a form on the right. The form can either be used to add a new student or edit information of an already enrolled student if the student’s name is clicked beforehand. By clicking the X next to the students name, the secretary can permanently delete the student from the database along with all existing data regarding that student.
* **Exam Schedule:** This page is now used in yet another way as the secretaries are only allowed to approve an exam scheduling, therefore making that exam date official.
* **Taxes:** In the taxes menu, the secretaries can do two things: they can add a new tax which has a tax name and an amount, and they can assign existing taxes to existing students.

1. **Class Discovery**

|  |  |
| --- | --- |
| **Student** | |
| Knows their own personal data | User |
| Can see all of their grades | Grade |
| Can see all of their taxes | Tax |
| Can see all of their exams | Exam |
| Make a payment |  |

|  |  |
| --- | --- |
| **Professor** | |
| Knows their own personal data | User |
| Can see all of their students | Student |
| Can add/change grades | Grade |
| Can add/remove/edit an exam | Exam |
| Can see all other exams | Exam |
| Can upload and see their wealth declaration | - |

|  |  |
| --- | --- |
| **Secretary** | |
| Knows their own personal data | User |
| Can see all students | Student |
| Can see all taxes | Tax |
| Can see all exams | Exam |
| Add new tax, assign tax to student | Tax, Student |
| Add/remove/edit students | Student |
| Approve/disapprove exam | Exam |

1. **Class Diagram**

A screenshot of a computer

Description automatically generated with medium confidence

**A picture containing text, monitor, screen

Description automatically generatedGraphical user interface, text, application

Description automatically generatedGraphical user interface, text

Description automatically generatedText

Description automatically generatedGraphical user interface, text

Description automatically generated with medium confidenceGraphical user interface, text

Description automatically generated**

Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedA picture containing text, outdoor

Description automatically generatedGraphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated