Software Requirements Specification

for

Management computer system automatic of college students

Version 1.0 approved

Prepared by
Bondrescu Clara-Ruxandra
Dobra Cristian-Sebastian
Preda Eduard-Gabriel

University of Craiova,
Faculty of Automation, Computers and Electronics

01/03/2022

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Bondrescu Clara-Ruxandra	01.03.2022	Added purpose, document conventions, intended audience and reading suggestions, product scope	1.0.0
Dobra Cristian-Sebastian	01.03.2022	Added references, product perspective, product functions, user classes and characteristics	1.0.1
Preda Eduard-Gabriel	01.03.2022	Added operating environment, design and implementation constraints, user documentation , assumptions and dependencies	1.0.2
Bondrescu Clara-Ruxandra, Dobra Cristian-Sebastian, Preda Eduard-Gabriel	03.03.2022	Added user interfaces	1.0.3
Preda Eduard-Gabriel	04.03.2022	Added software interfaces, hardware interfaces, communications interfaces	1.0.4
Dobra Cristian-Sebastian	04.03.2022	Added performance requirements, safety requirements, security requirements	1.0.5
Bondrescu Clara-Ruxandra	04.03.2022	Added software quality attributes, business rules	1.0.6

Bondrescu Clara-Ruxandra	12.03.2022	Added system features: 4.1, 4.2, 4.11, 4.12, 4.15, 4.16	1.0.7
Dobra Sebastian-Cristian	12.03.2022	Added system features: 4.6, 4.7, 4.8, 4.9, 4.13, 4.14	1.0.8
Preda Eduard-Gabriel	12.03.2022	Added system features: 4.3, 4.4, 4.5, 4.10, 4.17.	1.0.9

1. Introduction

1.1 Purpose

The application is used for the purpose to facilitate the Automatic Management of Faculty's Students (GASF).

1.2 Document Conventions

The format of the report has the following characteristics: default margins (1 inch), 1 line spacing, font Arial with a size of 11. The following characters "<>" will be use to mark the name of a part of code which is used in the implementation of the application. The following abbreviations will be used in this document:

-IDE = Integrated Development Environment
-SQL = Structured Query Language
-DB = Database
-HTML = Hypertext Markup Language
-CSS = Cascading Style Sheets
-HTTP = Hypertext Transport
-SRS = Software Requirement Specification
-GASF = Automatic Management of Faculty's Students

1.3 Intended Audience and Reading Suggestions

The types of audience for this document are the following:

-developers
-documentation writers
-users
-administrators

The rest of the document contains, in this order, the following categories:

-overall description of the product (product perspectives, product functions, user classes, design etc.)
-external interface requirements (user/software/hardware/communications interfaces)
-system features
-other nonfunctional requirements

1.4 Product Scope

The purpose of the project is to develop a software system to automate the process of management of the students of a faculty that would allow the following specific actions:

- collecting data corresponding to curricula, study formations and their students;
- data management;
- generating reports;
- providing information through a web portal;
- system query;
- document flow management within the faculty secretariat.

1.5 References

Support links:
- curricula - http://www.ace.ucv.ro/invatamant/didactica/planuri.php
- institutionalized study - http://www.ace.ucv.ro/invatamant/utile/orar.php
- evaluation methods - http://www.ace.ucv.ro/invatamant/calitate/evaluare.php

Useful links:
- http://caxapa.ru/thumbs/468328/misra-c-2004.pdf
- https://www.w3schools.com/
- https://www.codecademy.com/
- https://www.pluralsight.com/
- http://www.tutorialsteacher.com/
- https://www.freecodecamp.org/
- https://www.tutorialspoint.com/
- https://developer.mozilla.org/
- https://www.pluralsight.com/courses/understanding-aspdotnet-core-2x/
- https://www.toptal.com/angular/angular-5-asp-net-core/

Books:
- Adam Freeman: Pro ASP.NET Core MVC 2, Apress, 2017
- Michael Morrison: Head First JavaScript, O'Reilly Media, 2008
- Rob Larsen: Beginning HTML and CSS, Wrox, 2013

2. Overall Description

2.1 Product Perspective

This project was created with the idea of a self-contained product that helps both the automatic management of the students of a faculty and the structural organization of its curriculum. This application will be able to change the way that information about students, teachers, schedules and curricula are stored in physical form in a compact, computer-stored system.

Student information will be available to any member of the secretariat or to any teacher through a web portal. These members can create schedules, the virtual catalog, the distribution of scholarships to students when they are logged in. Teachers can also create and close reports on student evaluation as system administrators.

For trouble-free use of the product, the important information about students, schedules, and teachers stored in a database can be accessed by sending queries to the database server.

2.2 Product Functions

The final version of the product will have the following features:

-user/administrator registration
-user login
-administrator creation and removal
-adding or removing students
-adding or removing schedules
-search functions for students, scholarships, curricula
-create and close reports on student evaluation
-online catalog with updated grades
-personal data about students and teachers

2.3 User Classes and Characteristics

The application will be accessed by developers, administrators and users.

1. Administrators (secretariat and teachers) will have the following rights:
-right to add / edit students
-addition / completion of schedules
-view details school situation
-grading students and distributing scholarships.
-have access to all site information

2. Users will have the following rights:
-can access the site with the help of the account
-they can view both their personal information and the rest of the site, but they can’t make changes.

2.4 Operating Environment

The user minimum requirements are:

-The application will work on the Windows (versions 10 and 11) operating system.

-Web browser: Google Chrome version 98.0.4758.102, Microsoft Edge version 98.0.1108.62 or Opera GX version 83.0.4254.70.

The characteristics of the computer systems on which the computer system will be installed are: - processor: 1 GHz or faster; - memory: at least 128 Mb; - hard disk space: minimum 10 Mb.

2.5 Design and Implementation Constraints

The integrated development environment, frameworks and the technologies used to create this project are the following:

-Visual Studio IDE 2022 Community Edition version 17.1.0
-HTML version 5
-CSS version 3
-C# programming language version 8.0 or newer
-SQL Server
-HTTP protocol
-ASP.NET Core 3.0 or newer

2.6 User Documentation

This document is the only user documentation available for the web application. For more information, please contact us via e-mail or phone number available at the end of this document and on the website.

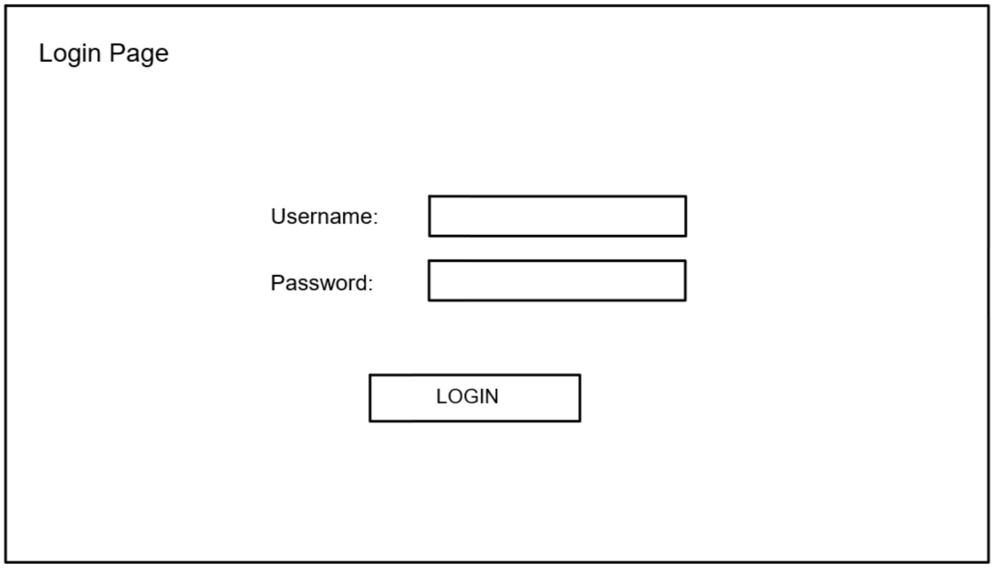
2.7 Assumptions and Dependencies

Not applicable.

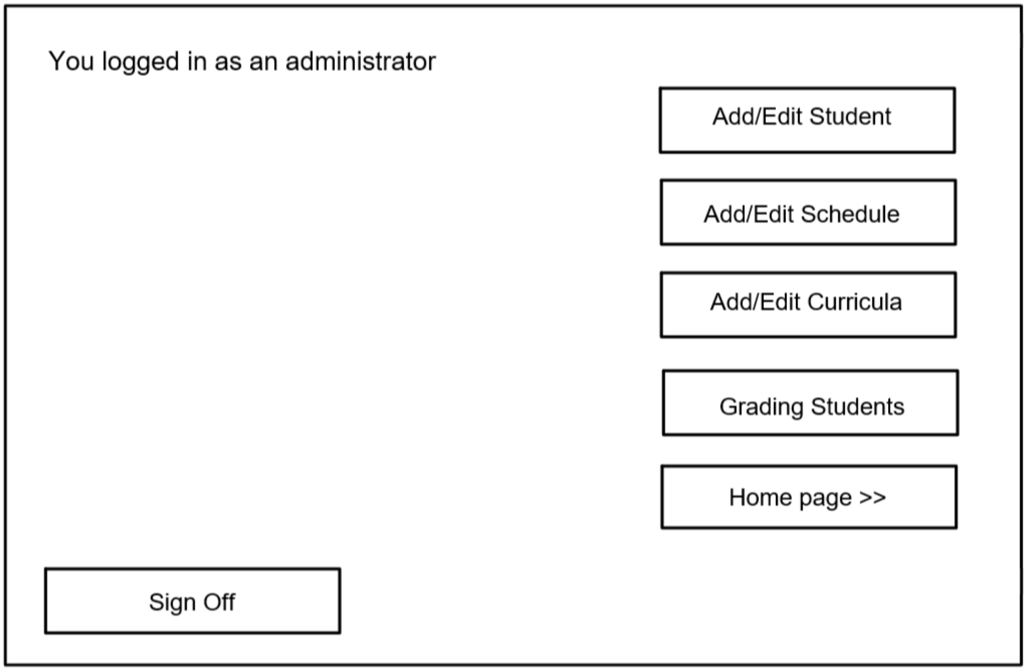
3. External Interface Requirements

3.1 User Interfaces

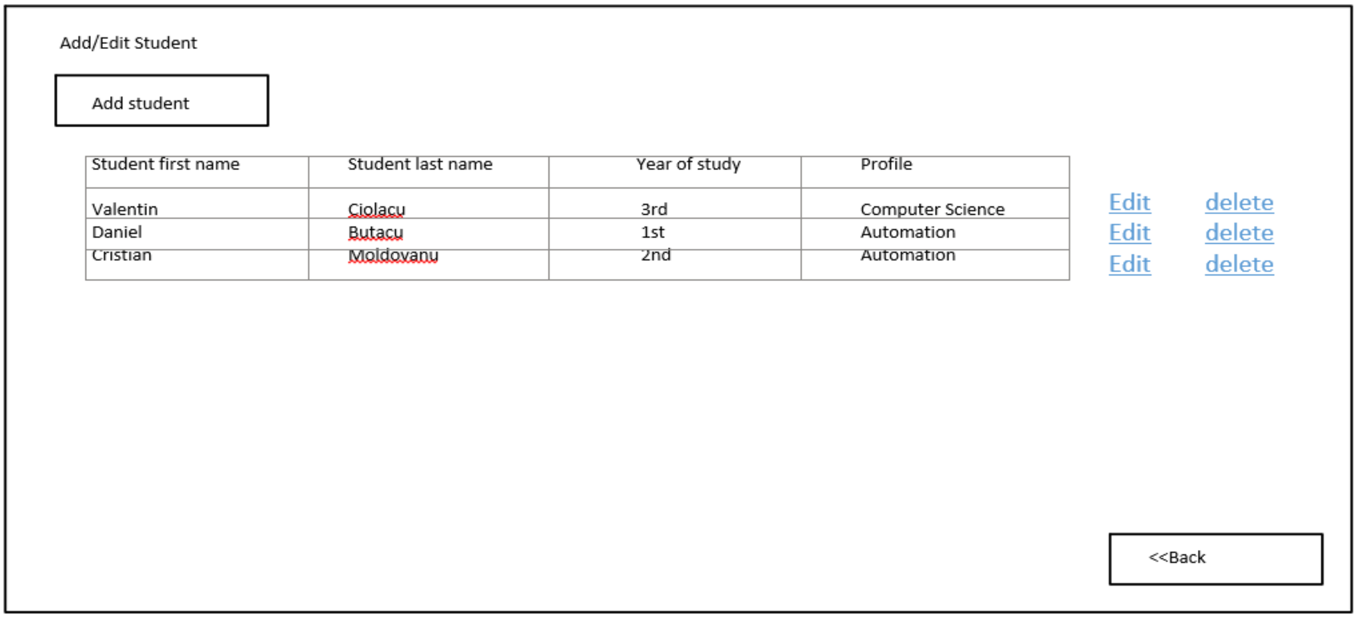
In the first instance, the user will have to enter his username and password that he received from the administrator or, if he is the administrator himself, to authenticate with his data. The next step is done by the application itself, which recognizes whether the username and password correspond to an administrator or a user (student).



If the username and password correspond to an administrator, then a window will open containing the message on the left that you have been logged in as administrator, and on the right 4 buttons that will open other windows if pressed, but also a button bottom left for log out.


You logged in as an administrator

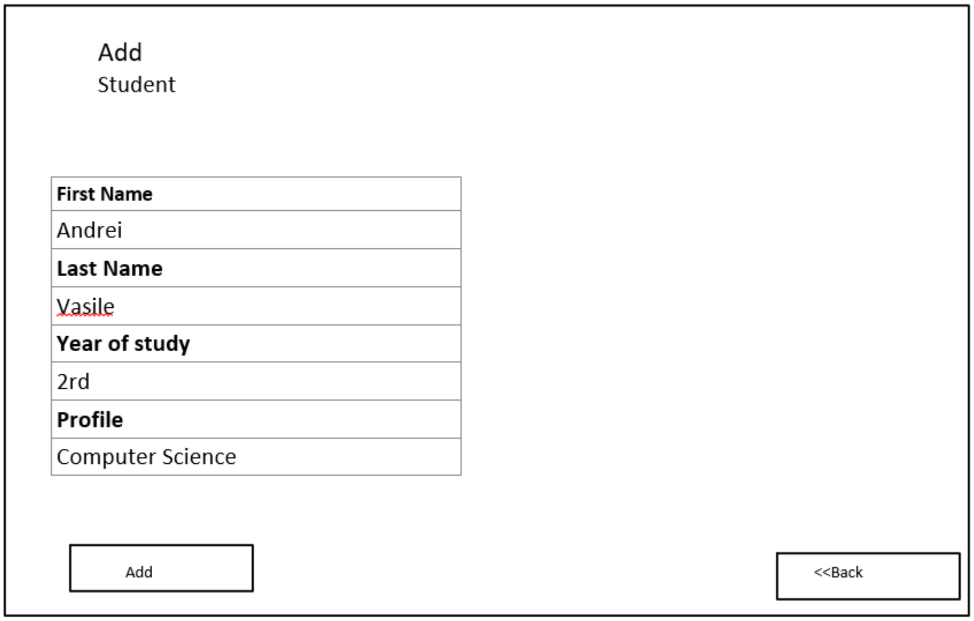
The first button from the administrator is for managing the students from faculty.



Add/Edit Student

Student first name	Student last name	Year of study	Profile		
Valentin	Ciolacu	3rd	Computer Science	Edit	delete
Daniel	Butacu	1st	Automation	Edit	delete
Cristian	Moldovanu	2nd	Automation	Edit	delete

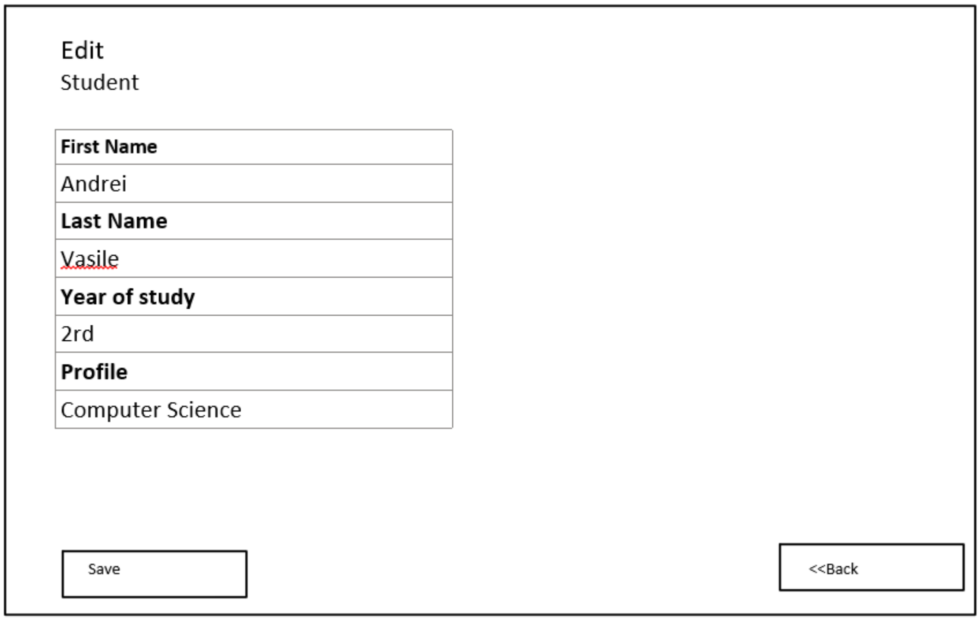
The Add student button opens another page where we can add the students of faculty.



Add
Student

First Name
Andrei
Last Name
Vasile
Year of study
2rd
Profile
Computer Science

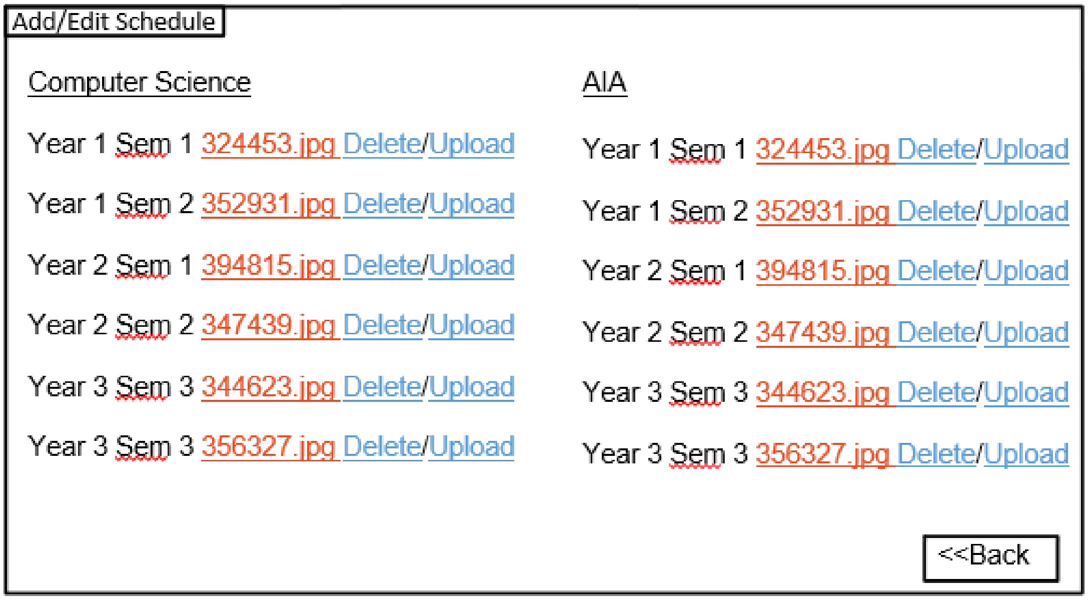
The edit button opens another page where there can be edited the data of students.



Edit
Student

First Name
Andrei
Last Name
Vasile
Year of study
2rd
Profile
Computer Science

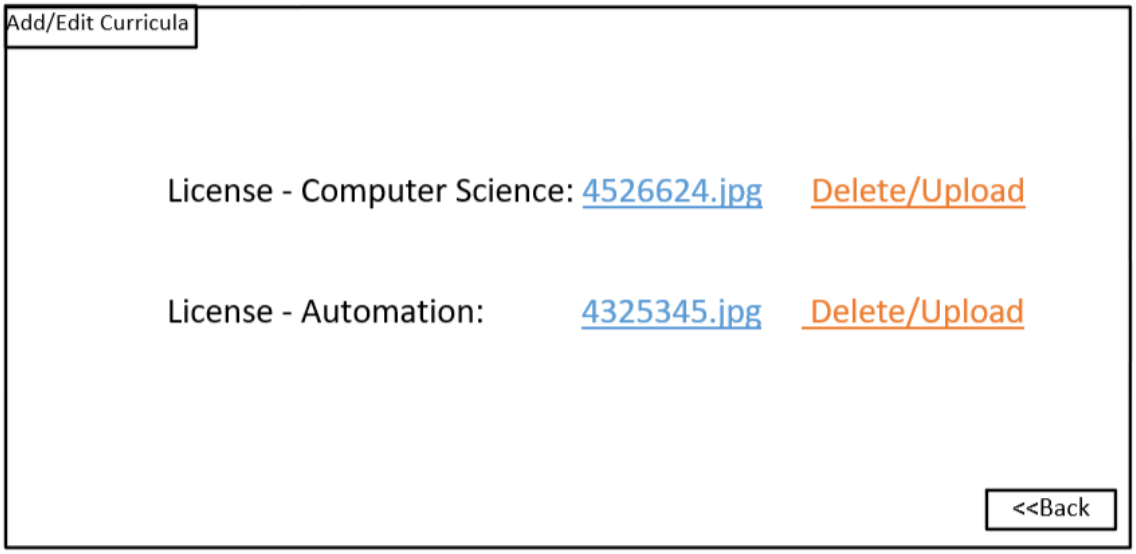
The second button from the administrator opens the page with the organization of schedules. Pressing add/edit Schedule(from Administrator account) pops up the next page. 324453.jpg represents the photo of the schedule which can be deleted(Delete) and replaced(Upload).



Add/Edit Schedule

Computer Science	AIA
Year 1 Sem 1 324453.jpg Delete/Upload	Year 1 Sem 1 324453.jpg Delete/Upload
Year 1 Sem 2 352931.jpg Delete/Upload	Year 1 Sem 2 352931.jpg Delete/Upload
Year 2 Sem 1 394815.jpg Delete/Upload	Year 2 Sem 1 394815.jpg Delete/Upload
Year 2 Sem 2 347439.jpg Delete/Upload	Year 2 Sem 2 347439.jpg Delete/Upload
Year 3 Sem 3 344623.jpg Delete/Upload	Year 3 Sem 3 344623.jpg Delete/Upload
Year 3 Sem 3 356327.jpg Delete/Upload	Year 3 Sem 3 356327.jpg Delete/Upload

The third button on the administrator page opens the Add/Edit Curricula page, where he can upload or delete the curricula for each profile.

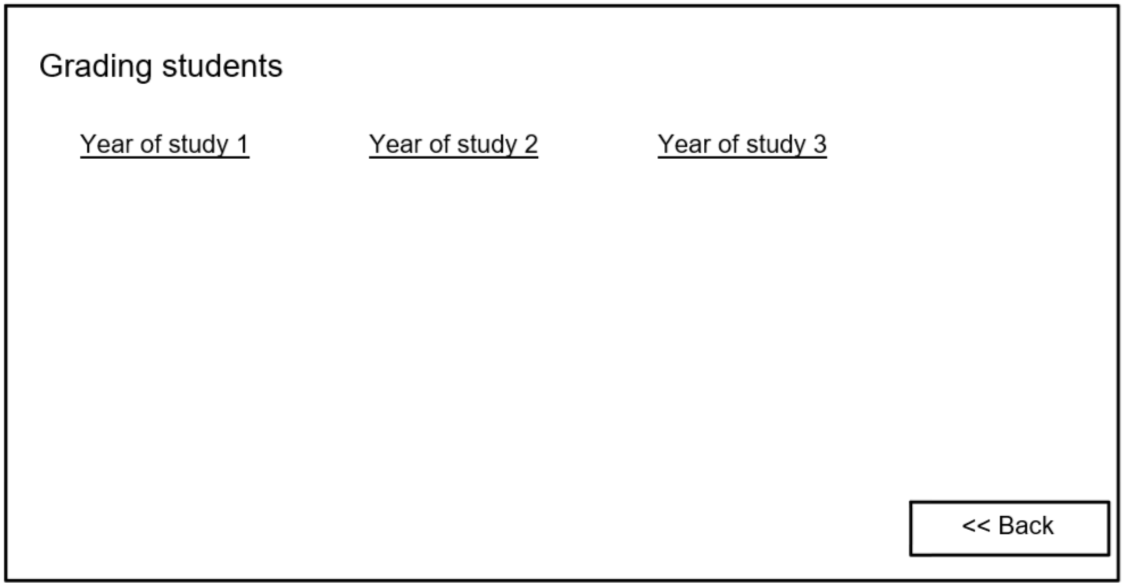


Add/Edit Curricula

License - Computer Science: 4526624.jpg Delete/Upload

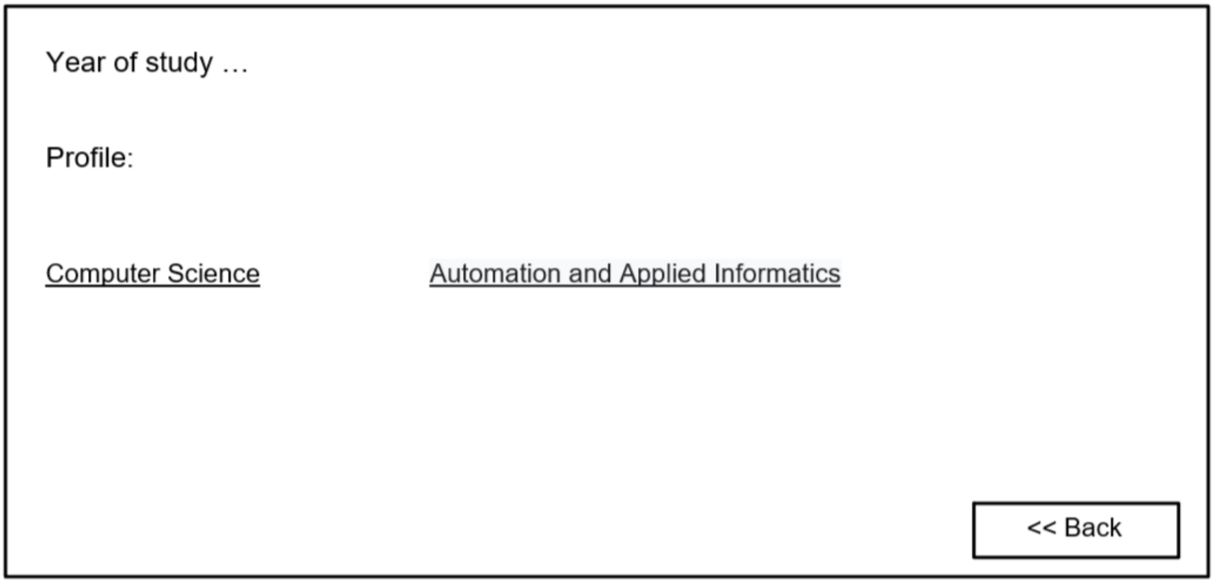
License - Automation: 4325345.jpg Delete/Upload

The 4 button from the administrator opens a page for grading management of students.

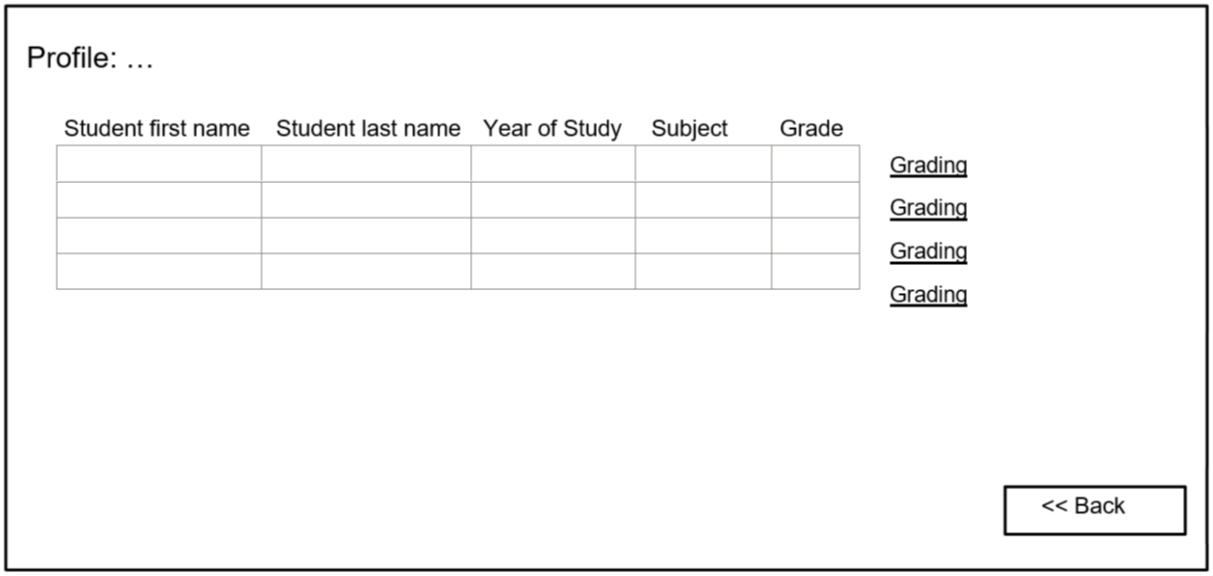


Grading students

Any of the students 'years of study will be selected so that later the students' section can be chosen, where their grades obtained in the subjects will be added.

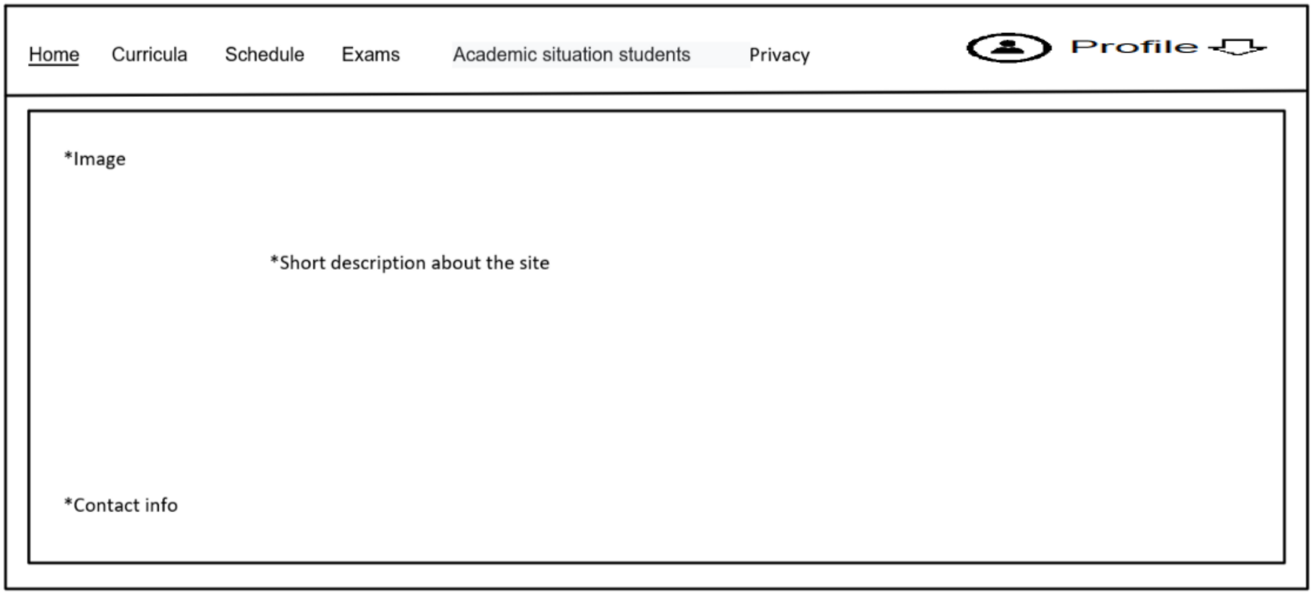


After selecting the students' specialization, a new page will open where the grades obtained by the students will be added.



Student first name	Student last name	Year of Study	Subject	Grade	
					Grading
					Grading
					Grading
					Grading

If the username and password correspond to a student or if the administrator decides to press the home button, the main page of the web application will open.

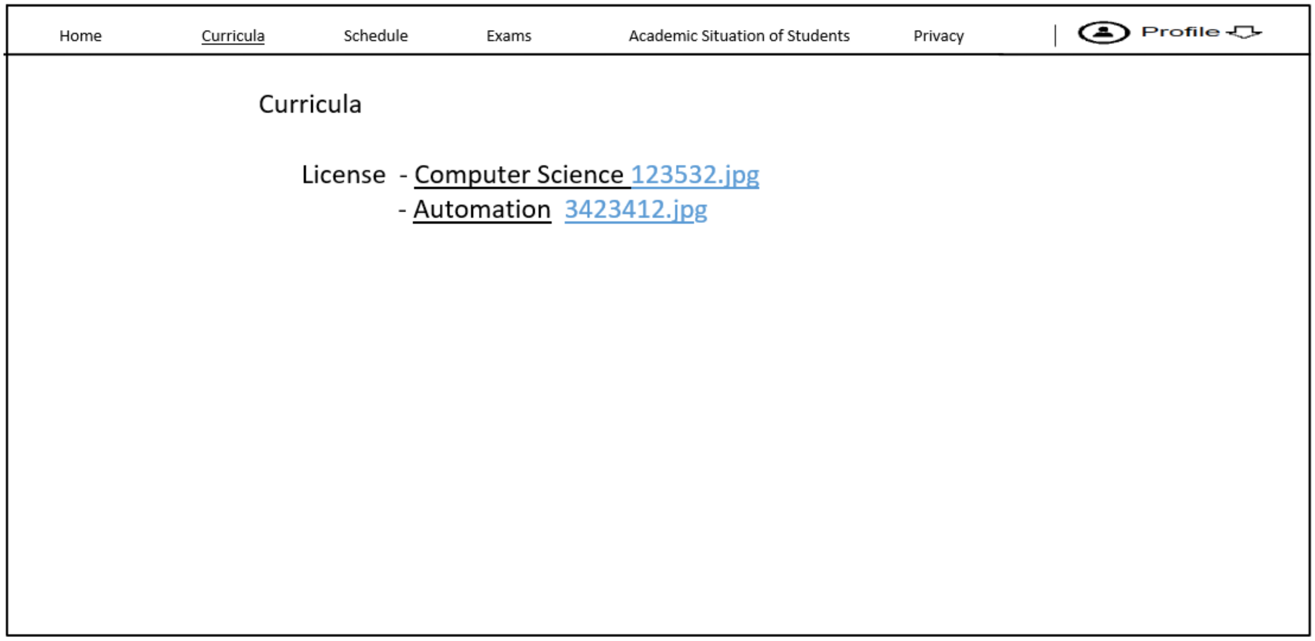


*Image

*Short description about the site

*Contact info

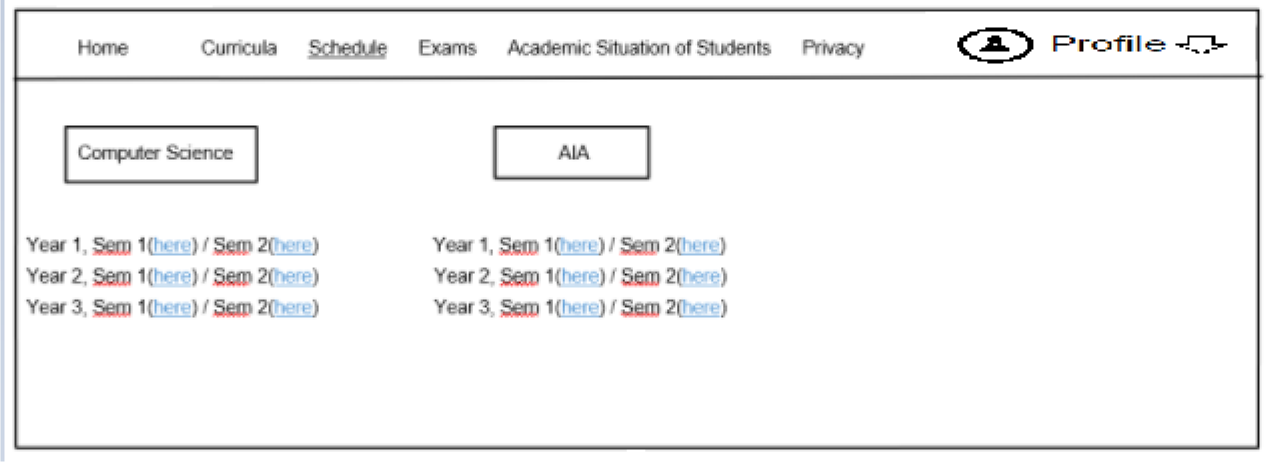
Selecting the curriculum will open the page for it.



Curricula

License - Computer Science 123532.jpg
- Automation 3423412.jpg

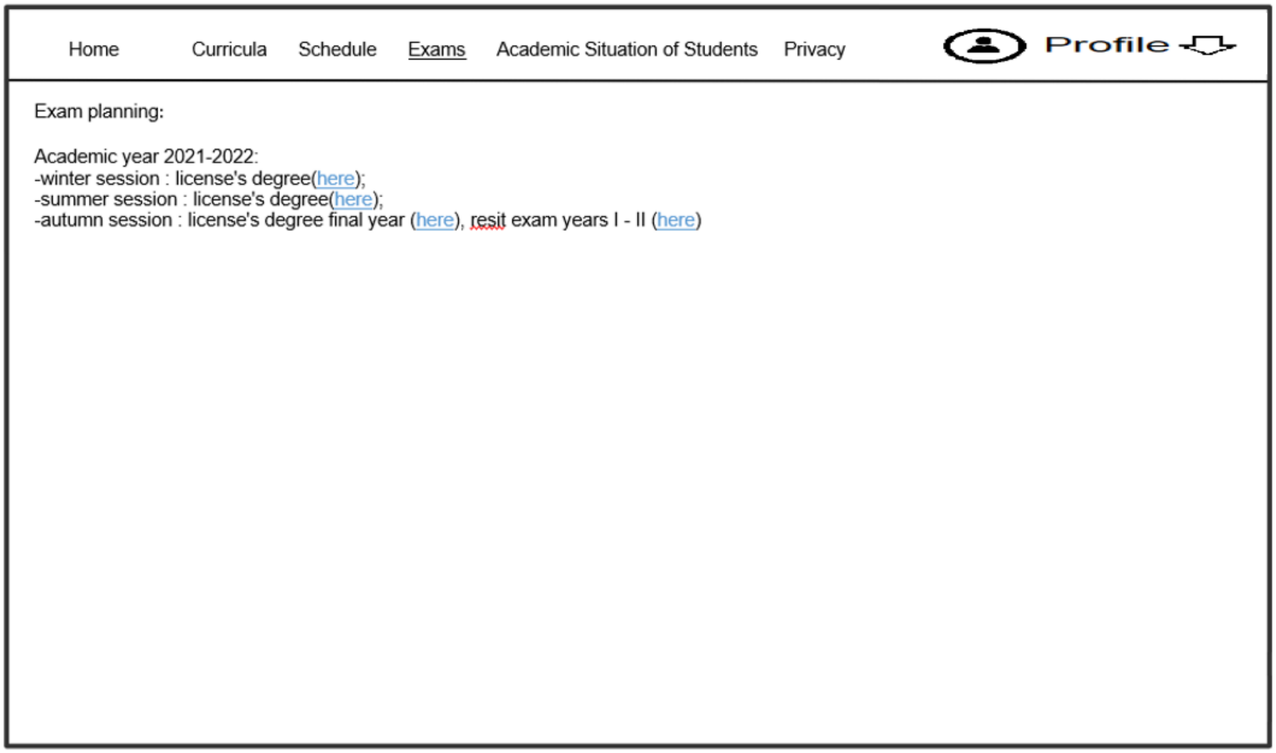
Selecting Schedule will open the page for it.



Year 1, Sem 1(here) / Sem 2(here)
Year 2, Sem 1(here) / Sem 2(here)
Year 3, Sem 1(here) / Sem 2(here)

Year 1, Sem 1(here) / Sem 2(here)
Year 2, Sem 1(here) / Sem 2(here)
Year 3, Sem 1(here) / Sem 2(here)

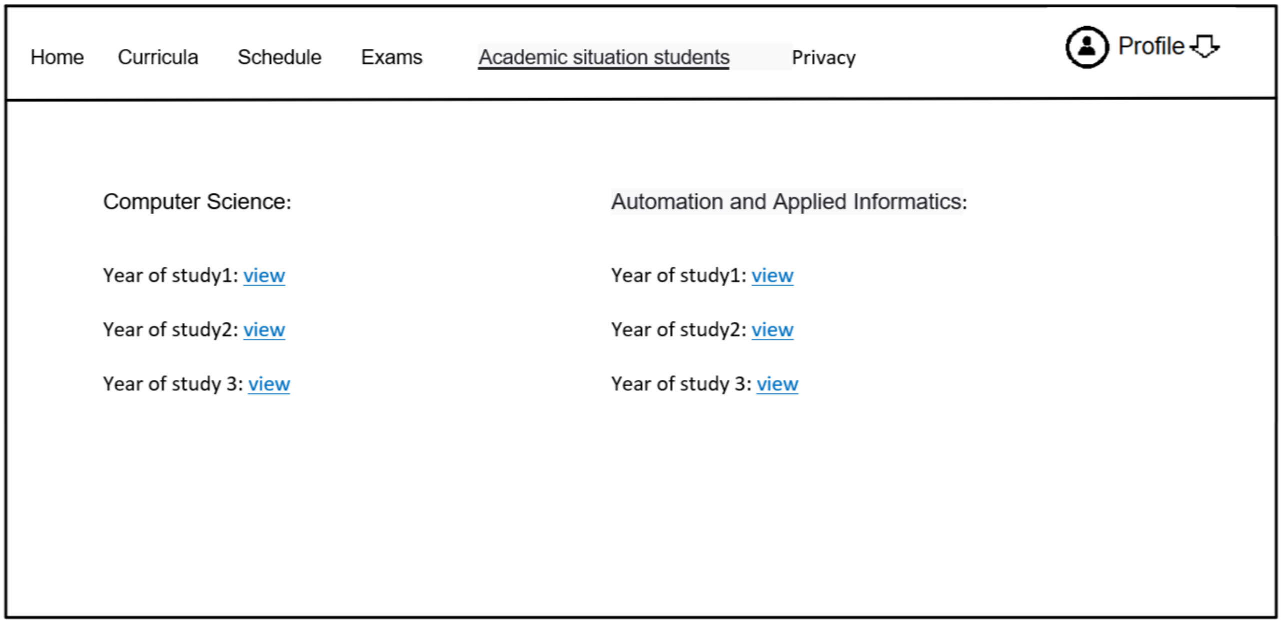
If you select the Exams, a new page will appear, containing the date for the exams in the current year.



Exam planning:

Academic year 2021-2022:
-winter session : license's degree(here);
-summer session : license's degree(here);
-autumn session : license's degree final year (here), resit exam years I - II (here)

If we select Academic Situation Students a new page will appear, in which we will be able to view the grades obtained by students in all subjects.



Computer Science:

Year of study1: view

Year of study2: view

Year of study 3: view

Automation and Applied Informatics:

Year of study1: view

Year of study2: view

Year of study 3: view

3.2 Hardware Interfaces

Not applicable.

3.3 Software Interfaces

There will always be a connection between the server and the database. There will be several functions (to read from the database and to write to the database or to delete from the database). There will also be a function that will view the data in the database. Function prototypes:

private void btn_Insert_Click(object sender, EventArgs e)
private void btn_Update_Click(object sender, EventArgs e)
private void btn_Delete_Click(object sender, EventArgs e)
private void DisplayData()

3.4 Communications Interfaces

The program will have a web page for communication with the users, thus requiring a web browser and network server communications protocols.

As communication standards, the application uses a HTTP protocol.

The web page will be synchronized with the information in the database server.

4. System Features

In this paragraph are noted and described the functional requirements for the application. Those requirements will be organized by the use case in sections.

4.1 Login administrator

4.1.1 Description and Priority

The administrator will be able to log in to the application using his specific account that the application will recognize.

Priority: High

4.1.2 Stimulus/Response Sequences

Input: On the register page the following information will be completed: administrator-specific password and username, then press the Login button.

Output: Login status.

4.1.3 Functional Requirements

-The administrator will enter the username and password.

-The administrator will press the login button.

REQ-1: There must be a textbox in which the administrator enters the username.

REQ-2: There must be a textbox in which the administrator enters the password.

REQ-3: There must be a button that the administrator must press after entering the data.

REQ-4: If the entered user is the wrong one, then the administrator cannot log in successfully and an error message will appear.

REQ-5: If the password entered does not correspond to the user then the administrator will not be able to authenticate successfully and an error message will appear.

REQ-6: If the username and password are entered correctly, the “administrator permissions” page will open.

4.2 Signing off as an administrator

4.2.1 Description and Priority

The administrator can log out by clicking on the sign off button.
Priority: High

4.2.2 Stimulus/Response Sequences

Input: There is a Sign off button on the administrator's permissions page.

Output: After selecting the Sign off button the administrator will exit its page.

4.2.3 Functional Requirements

-The administrator selects the sign off button.
REQ-7: There must be a button for the administrator to press when they want to log out.
REQ-8: After pressing the button, the administrator is redirected to the login page.

4.3 Add students

4.3.1 Description and Priority

If the administrator presses the button Add/Edit Student, it will open a new page which will allow him to use the specified functionalities. In this page, there are the next buttons : Add Student, Edit, Delete and Back, also a tabel which contains all the data about all the students. If he presses the Add Student button, a new page will show where he will be able to add personal data of the student. After he will complete the data, he will press the add button from the current page, and if he presses the <<Back button, he will return to the Add/Edit Students page, where he can verify the added student to the tabel.

Priority: High

4.3.2 Stimulus/Response Sequences

Input1: There is the Add/Edit Student button on the Administrator page.

Output1: A page for Add/Edit Students will appear.

Input2: In this page there is an Add Student button.

Output2: Open a new page for Add Students.

Input3: In the Add Student page, the administrator will complete the personal data about the Student(First Name, Last Name, Year, Profile) and press Add button.

Output3: The data will be sent to the database through a query, and then it will be forwarded in the tabel from Add/Edit Student page.

Input4 : In Add Student page there is the <<Back button.

Output4: The administrator will return to the add student edit page and will be able to view the data about the students added to the table on this page.

4.3.3 Functional Requirements

-The administrator selects the Add/Edit student button and opens the Add/Edit Student page.

REQ-9: On the Add/Edit Student page there is an Add Students button, which will redirect the administrator to the Add Student page.

REQ-10: On the Add/Edit Student page there is a table that will be filled in automatically after the administrator enters the student data on the Add Student page because it is connected to the database.

REQ-11: In the Add Student page there are text boxes in which the administrator will add the name, surname, year, section of the student, who each have a tag that specifies which column in the table on the previous page will be completed and the information will be stored in the database server by pressing the Add button.

REQ-12: On the Add Student page there is the back button which if pressed by the administrator will be sent back to the previous page.

REQ-13: After adding students, the Administrator can see on the Add/Edit Student page the table with their data sorted by Year, Section and Name by querying the database.

4.4 Edit students

4.4.1 Description and Priority

On the Add/Edit Student page, the administrator can select the edit button next to the line in the table that contains the information they want to change for a particular student. After pressing the button, a new page called Edit Student will open where you can make changes in any field you want.

Priority: High

4.4.2 Stimulus/Response Sequences

Input1: There is an edit button on the add student edit page.

Output1: The student edit page will open.

Input2: On the student edit page the administrator can change any date of the student in a certain field (Name, Surname, year, specialization) and press Save button.

Output2: The modified data will be updated in the database by a query, and then it will be modified in the table on the student add / edit page.

Input3 : In Edit Student page there is the <<Back button.

Output3: After pressing the <<Back button, the administrator will return to the add student edit page and will be able to view the data about the students added as well as modified student data to the table on this page.

4.4.3 Functional Requirements

-The administrator selects the edit button next to any row in the table on the add student edit page to open the student edit page.

REQ-14: In the student edit page there are textboxes that contain the information of the student selected for editing, and the administrator can change any of the fields and finally click the Save button, resulting in changing the data both in the database and in the table on the Add / Edit Student page. each field has a tag to each column in the table.

REQ-15:On the Edit Student page there is the back button which if pressed by the administrator will be sent back to the previous page.

REQ-16:After editing students, the Administrator can see on the Add/Edit Student page the table with their data sorted by Year, Section and Name by querying the database.

4.5 Delete Students

4.5.1 Description and Priority

On the Add / Edit Student page, the administrator can select the delete button next to the row in the table that contains the information about the student they want to delete.

Priority:High

4.5.2 Stimulus/Response Sequences

Input: On the Add/Edit Student page there are delete buttons next to each row in the table.

Output: The student data will be deleted and the line in the table containing it will disappear.

4.5.3 Functional Requirements

REQ-17: On the Add/Edit Student page there are delete buttons next to each line in the student information table.

REQ-18: The administrator will select any delete button to erase the student data from the table, and it will be deleted from the database by a query.

4.6 Add Schedule

4.6.1 Description and Priority

On the Add Schedule page, the administrator can add a new schedule for the specified study year by selecting the Add Schedule button.

Priority: medium

4.6.2 Stimulus / Response Sequences

Input: there is an add button on the add time page

Output: the button will add a new schedule to the schedule page

4.6.3 Functional Requirements

REQ-19: On the add time page there is the upload button, with the help of which the administrator can add a picture with the specific schedule of each section, year and semester, and the images will be connected to the database.

REQ-20: In the add time edit page there is a back button that will send the administrator to his specific page.

4.7 Delete Schedule

4.7.1 Description and Priority

On the Add/Edit Schedule page, the administrator can select the delete button next to the row in the table that contains the information about the schedule they want to delete.

Priority: Low

4.7.2 Stimulus/Response Sequences

Input: On the Add/Edit Schedule page there are delete buttons next to each row.

Output: The schedule image will be deleted and a new image will be uploaded and saved as the new schedule.

4.7.3 Functional Requirements

REQ-21: On the Add/Edit Schedule page there are delete buttons next to each line.

REQ-22: The administrator will select any delete button to erase the schedule image from the page, and it will be deleted from the database by a query.

4.8 Add Curricula

4.8.1 Description and Priority

On the Add / Edit Curriculum page, the administrator can add pictures of the curriculum for each section.

Priority: Medium

4.8.2 Stimulus / Response Sequences

Input1: The administrator presses the Upload button.

Output1: A field will open from which the administrator will add the desired picture.

Input2: Select the picture.

Output2: Add picture.

4.8.3 Functional Requirements

REQ-23: In the add curriculum edit page there are upload buttons, with the help of which the administrator can add pictures with the curriculum specific to the school year, and the images will be connected to the database.

REQ-24: In the add curriculum edit page there is a back button that will send the administrator to his specific page.

4.9 Delete Curricula

4.9.1 Description and Priority

On the Add/Edit Curricula page, the administrator can select the delete button next to the line that contains the information about the curricula they want to delete.

Priority:Low

4.9.2 Stimulus/Response Sequences

Input: On the Add/Edit Curricula page there are delete buttons next to each line.

Output: The curricula data will be deleted and the line containing it will be empty.

4.9.3 Functional Requirements

REQ-25: On the Add/Edit Curricula page there are delete buttons next to each line.

REQ-26: The administrator will select any delete button to erase the curricula image from the page, and it will be deleted from the database by a query.

4.10 Grading Students

4.10.1 Description and Priority

On the Grading Students page, the administrator can select the year of study of the students for whom they want to add Notes. After selecting a year of study, a new page will open specific to the selected one where the administrator will select the student section where he will want to add grades. After selecting the section, a new page specific to it will open, in which there will be a table with student data and a column where the administrator will add notes.

Priority: High

4.10.2 Stimulus / Response Sequences

Input1: On the Grading students page there are buttons for selecting the year of study of the students.

Output1: Open the page for the selected study year.

Input2: On the page of the study year there are buttons for selecting the student profile.

Output2: Open the respective section of the selected student section.

Input3: The student section page will contain a table, where the administrator will write in the Subject and Grade column the subject and the student's grade obtained in it, as well as the Grading button.

Output3: After pressing the Grading button, the subject and the student's grade will be added to the table, but also to the database.

4.10.3 Functional Requirements

REQ-27: On the Grading students page the administrator can click on the button with the student's year of study which will redirect him to a new page.

REQ-28: On the page of the selected year of study, the administrator can choose the student section where he / she will want to add grades.

REQ-29: There is also a back button on the study year page that will return the administrator to the previous page if he selected the wrong study year.

REQ-30: On the student section page there is a table with their data and two columns where the administrator will add the subject and the grade obtained by the student to it by pressing the grading button, both in the table and in the database through a query.

REQ-31: On the student section page is a back button that the administrator can press to go to the previous page.

4.11 Home page

4.11.1 Description and Priority

On each page there is a home page button that redirects both the user and the user to the main menu of the application.

Priority: high

4.11.2 Stimulus / Response Sequences

Input: On each page there is a home page return button available for both user and admin

Output: Selecting the respective button will redirect both the user and the admin to the main page of the application

4.11.3 Functional Requirements

REQ-32: In any page of the application both the user and the admin can press the home page button which will redirect them to the main page.

REQ-33: The main page will have a background image, a description of the site and the contact details of the faculty.

4.12 Login user

4.12.1 Description and Priority

The user will be able to log in to the application using his specific account that the application will recognize.

Priority: High

4.12.2 Stimulus/Response Sequences

Input: On the register page the following information will be completed: user-specific password and username, then press the Login button.

Output: Login status.

4.12.3 Functional Requirements

-The user will enter the username and password.

-The user will press the login button.

REQ-34: There must be a textbox in which the user enters the username.

REQ-35: There must be a textbox in which the user enters the password.

REQ-36: There must be a button that the user must press after entering the data.

REQ-37: If the entered user is the wrong one, then the administrator cannot log in successfully and an error message will appear.

REQ-38: If the password entered does not correspond to the user then the user will not be able to authenticate successfully and an error message will appear.

REQ-39: If the username and password are entered correctly, the user will enter in home page.

4.13 View Curricula

4.13.1 Description and Priority

When you click on the Curriculum (Curricula) button, its page will be displayed where several options will appear that will include information related to it.

Priority: medium

4.13.2 Stimulus / Response Sequences

Input: Press the button next to the specialization whose curriculum we want to view.

Output: The curriculum will be displayed.

4.13.3 Functional Requirements

REQ-40: Press the Curriculum button in the ship.

REQ-41: Press the desired button for the profile whose curriculum we want to view, and a picture connected to the database will open.

REQ-42: The selected curriculum will be displayed.

4.14 View Schedule

4.14.1 Description and Priority

When you click on the Schedule button, the page specific to it will open, in which the profiles that include the schedules for the respective year and semester are displayed.

Priority: medium

4.14.2 Stimulus / Response Sequences

Input: Select the button next to any semester.

Output: The page containing the schedule will open.

4.14.3 Functional Requirements

REQ-43: Press the Schedule button on the ship.

REQ-44: Select the Here button for the schedule from the Profile / Year / Semester we want.

REQ-45: The selected time will be displayed.

4.15 View exams

4.15.1 Description and Priority

Click the Exams button in the navebar and the page will open with the summer, autumn and winter sessions.

Priority: Medium

4.15.2 Stimulus / Response Sequences

Input: Click the Here button next to any exam session.

Output: An image will be displayed that includes all the exams for that session.

4.15.3 Functional Requirements

REQ-46: Press the Exams button in the navebar and display its page.

REQ-47: Click the Here button from the session we want to see.

REQ-48: The image with the exam table will be displayed.

4.16 View Academic situation of students

4.16.1 Description and Priority

From the navebar you will select Academic situation of students which will open the page where users and administrators will be able to view the grades obtained by the students of the faculty.

priority: high

4.16.2 Stimulus / Response Sequences

Input: Press the view button next to the section and the year of study of the students

Output: Redirect to a new page where data and grades from student subjects will be displayed in a table.

4.16.3 Functional Requirements

REQ-49: Users or the administrator select one of the view buttons, which will open a new page.

REQ-50: the newly opened page will contain the table with the information of the students from the selected section and year, but also the grades for the subjects obtained by them.

REQ-51: There is a back button on this page that redirects users to the previous page.

4.17 Privacy View

4.17.1 Description and Priority

The Privacy Page contains the Terms and Conditions plus the Privacy Policy.

Priority: Low

4.17.2 Stimulus/Response Sequences

Input: The Privacy page opens.

Output: The Privacy page provides us with personal account information and regulations.

4.17.3 Functional Requirements

REQ-52: Access the Privacy page from the navebar.

REQ-53: Information about both the Account and the regulations will be displayed.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The web application must be able to manage a large amount of information about students and output in the web browser the information retrieved from the database server in less than 10 seconds from the initial request.
The web application should be able to handle a medium number of users, at least 100 users at any given moment of time.
The interval between each request from the user and its response from the database server must be less than 5 seconds.

5.2 Safety Requirements

Not applicable.

5.3 Security Requirements

The personal information of the users will be safely stored in the server database.
This information is secured, requiring a user and a password in order to access it.
The user logged in will be able to see only its personal information.
The administrators are the only users who can access the students’ accounts and have the attribute to modify the information.

5.4 Software Quality Attributes

As software quality attributes, the final product must have the following characteristics:

1. Each class name will be written in upper camel case (e.g.: ClassName)
2. Each function and variable will be written in lower camel case (e.g.: functionName, variableName)
3. All comments will be multi-line / * comment * /, the comments for variables and additional information will be made by using "//"
4. Each accolade will be on a separate line, without existing any other character on that line
5. Each class must not have more than 2500 lines written
6. Only one space (line) is allowed between instructions to write an orderly, organized code in logical sections
7. All variables will be declared before use
8. All statements must have open and close accolades
9. Use parentheses to make clauses in an expression apparent
10. All variables will have a name with a certain meaning so the ambiguity is removed

5.5 Business Rules

The user must not give his personal account to other people.
The user is not allowed to use other people accounts.
The system must have at least one administrator account to access and interact with
the application.

6. Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>