Contents

Contents 1

List of Figures 3

Preface 1

1 Project Overview and context 2
- 1.1 Introduction . . . 3
- 1.2 The Faculty of Exact sciences . . . 3
- 1.3 Presentation of the problem . . . 4
- 1.4 Presentation of the solution . . . 5
- Conclusion . . . 5

2 Application Design 6
- 2.1 Introduction . . . 7
- 2.2 Definition . . . 7
- 2.3 UML Diagrams . . . 7
  - 2.3.1 Use Case Diagram . . . 7
  - 2.3.2 Detailed use-case diagrams . . . 8
  - 2.3.3 Class Diagram . . . 14
  - 2.3.4 Sequence Diagram . . . 16
- 2.4 The Relational Model . . . 20
- 2.5 Site Map . . . 22
  - 2.5.1 Student Pages Site Map: . . . 23
  - 2.5.2 Teacher Pages Site Map: . . . 24
  - 2.5.3 Admin Pages Site Map: . . . 25
- Conclusion . . . 25

3 Implementation 26
- 3.1 Introduction . . . 27
- 3.2 Devlopment Tools . . . 27
  - 3.2.1 database development . . . 27
  - 3.2.2 UI development . . . 28
  - 3.2.3 API creation . . . 28
  - 3.2.4 secure the API endpoint . . . 29
- 3.3 Data Base Schema . . . 30
- 3.4 Application Pages . . . 33
  - 3.4.1 Home page . . . 33

3.4.2 Login page 33
3.4.3 First Login 34
3.4.4 Profile page 35
3.4.5 Ranking page 35
3.4.6 Invite page 35
3.4.7 Team section page 36
3.4.8 Create Appointment 37
3.4.9 Suggest a dissertation 38
3.4.10 Ranking management 39
3.4.11 Suggested Themes Management 40
3.4.12 Supervisors Management 40
3.4.13 Team management 41
3.4.14 Calendar page 43
3.4.15 Admin Page 44
Conclusion 46

General conclusion **47**

Bibliography **48**

Notations **49**

List of Figures

1.1 organisation of exact science faculty . . . 3

2.1 Use Case Diagram for Authentication System . . . 8
2.2 Use-case Diagram for student actions . . . 9
2.3 use case diagram for Teacher Actions . . . 10
2.4 Specialty Manager Actions Use-case diagram . . . 12
2.5 Admin use-case diagram . . . 14
2.6 Class Diagram . . . 15
2.7 Sequence Diagram « User Authentication » . . . 16
2.8 Sequence Diagram «Change Information» . . . 17
2.9 Sequence diagram for user Add Teacher . . . 18
2.10 Sequence Diagram «Fixing Student Choices (Master)» . . . 19
2.11 Sequence diagram «Fixing Student Choices (Licence)» . . . 20
2.12 Student Pages Site Map . . . 23
2.13 Teacher Pages Site Map . . . 24
2.14 Admin Pages Site Map . . . 25

3.1 PhpMyAdmin Logo . . . 27
3.2 MySql Logo . . . 28
3.3 React JS library Logo . . . 28
3.4 Laravel Framework Logo . . . 29
3.5 JWT Logo . . . 29
3.6 List of Tables . . . 30
3.7 Table of Students . . . 30
3.8 Table of Teachers . . . 31
3.9 Table of Teams . . . 31
3.10 Table of Jury members . . . 32
3.11 Table of Ranks . . . 32
3.12 Home Page . . . 33
3.13 Login Page . . . 33
3.14 email containing their account details and password . . . 34
3.15 Change Password in first log in . . . 34
3.16 Profile Page . . . 35
3.17 Ranking Page . . . 35
3.18 Invite Page . . . 36
3.19 Supervisors list Page . . . 36
3.20 dissertations list Page . . . 37

3.21 Chat room Page 37
3.22 Create Appointment 38
3.23 Example Of Suggest a Licence Dissertation 38
3.24 Example Of Suggest a Master Dissertation 39
3.25 Add students Rank 39
3.26 Suggested themes list 40
3.27 Supervisors list (speciality manager view) 40
3.28 Teams list 41
3.29 Effortless Scheduling: Seamlessly Manage Appointments by Specialty Managers 41
3.30 View of state of progress of a team 42
3.31 Create jury group for master 42
3.32 Create jury group for licence 43
3.33 Calendar Page 43
3.34 Designation of the jury members 44
3.35 Students Accounts 44
3.36 Organizing Students by Specialty and Years of Study 45
3.37 Add Student account 45
3.38 Teachers Accounts and their roles 46

General Introduction

The use of computers has led to a significant technological breakthrough in all industries, allowing for more efficient management of information. Before computers, information was recorded on paper, resulting in lost time and degradation of information. To be effective, educational institutions must view information as a resource and strategically invest in the development of information systems to gather, store, organize, and share information.

Today, the faculties within the university we are studying are an essential component of the educational system. With the expansion of the student population, there is a growing need for rational and efficient management practices to ensure smooth operations. However, traditional manual management methods are still dominant, which can lead to inefficiencies and delays. It is essential to introduce computerized systems to these training centers to streamline administrative processes and enable faster decision-making. With the advancements in data processing technology, such systems can enhance the quality of education and overall performance of the university.

Managing graduation projects with traditional manual management methods can be particularly challenging. One of the significant issues is that the manual management of these projects can be time-consuming and prone to errors, which can lead to delays and inefficiencies. The process of manually managing graduation projects involves a significant amount of paperwork, which can be overwhelming and challenging to keep organized. This can result in misplaced documents or lost information, leading to further delays and difficulties in tracking the progress of each student's project. Additionally, manually managing the scheduling of defense dates, project proposals, and documentation can be a time-consuming and complex process, which can result in delays and confusion.

Our main objective is to design and develop a web application for project management at the completion of Licence's and master's degrees

Having a web-app dedicated to managing graduation projects would be incredibly useful for educational institutions. It would streamline the process of managing student projects, providing a centralized platform for students, supervisors, and administration to communicate and share project-related documents

our project is organized like so:

- The first chapter provides an overview of the Exact Science faculty and its various departments, as well as the organizational tasks involved in managing final projects.
- The second chapter delves into Unified Modeling Language (UML) and provides a detailed study of visual models that represent the functionality of our web application.
- The third chapter outlines the toolset utilized in implementing our web application, showcasing its features through screenshots and discussing our database tables.

Project Overview and context

1

Contents

1.1 Introduction 3
1.2 The Faculty of Exact sciences 3
1.3 Presentation of the problem 4
1.4 Presentation of the solution 5
Conclusion 5

1.1 Introduction

Mustapha Stambouli University stands out from the others due to its cultural and scientific influence and its contribution to the development of the city. It was established in 1986 as an agronomy-focused national institute of higher education (INES). Later, it was developed into a university with seven faculties that are dispersed among three poles.

1.2 The Faculty of Exact sciences

The Faculty of Exact sciences was created in may 2015 by Executive Decree No. 15 – 110 of 03/05/2015 which entrusts it with the tasks of scientific research related to Higher Education. The Faculty of Exact Sciences of Mustapha Stambouli University includes 04 departments: (mascara university, n.d.a)

- Computer Sciences
- Mathematics
- Chemistry
- Physics

gathering about 1225 students in graduation and 95 in post-graduation, 63 civil servants ATS and 101 teachers-researchers. Several training programs are offered in Licence's and Master's degrees in the fields of Material Sciences (MS) and Mathematics-Informatics (MI) are open, 05 Licence's courses and 09 Master's courses.(mascara university, n.d.b)

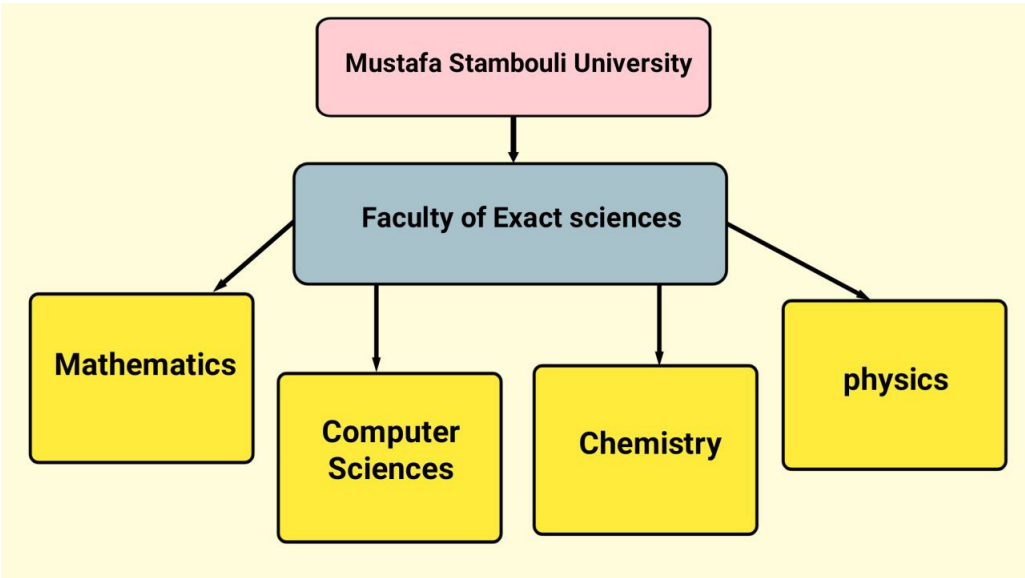


Figure 1.1 – *organisation of exact science faculty*

All departments have adopted the LMD (Licence, Master and Doctorate) study course. It offers students in the field of Computer Sciences with three specialties in master:

- Master in Networks and Distributed Systems (RSD)
- Master in Computer Systems Engineering (ISI)
- Master Information Systems and Web Technology (SITW) As for Licence students who are scheduled to study:
- Licence in Computer Systems (SI)

The Department of physics offers a Licence's degree and two Master's degrees:

- Licence in Physics
- Master in Physics of Materials
- Master in Physics and Renewable Energy

The Department of mathematics provide a Licence's degree and two Master's degrees:

- Licence in Mathematics.
- Master in Differential Geometry
- Master in Mathematical Analysis

The Department of chemistry currently offers a Licence's degree and a Master's degrees:

- Licence in Chemistry
- Master in Macromolecular Materials Chemistry

At the end of each academic career, whether that of 3 years (Licence) or of the 2 years (master) except the Mathematics department, students are required to complete a final project. The administration of the faculty concerned by all the tasks and stages of the viva (soutenance).

1.3 Presentation of the problem

The administration of the Faculty of Exact Sciences carries out all tasks of the defense in a regular or straightforward manner using Word, Excel, or by hand. However, this method has numerous errors and bugs. Our project is being done as part of a computer science Licence dissertation, and its major objective is to design and develop a web application for the Faculty of Exact Sciences to administer and monitor PFE (Projet Fin d'Etude) Master and Licence dissertations. The organization of these final projects involves the following tasks:

- The student has the option to choose their own partner for the project, or they may work alone on the project if they are unable to find a suitable partner.
- The choice of topics (theme): Final projects for students can be either self-selected or proposed by teachers.

- For self-selected projects, the proposed topics need to be validated by the specialty manager, while teachers consult with the specialty manager to designate candidates for proposed themes.
- The specialty manager proceeds to assign supervisors and confirm the final choice of theme with a detailed project description after collecting all proposed themes.
- The specialty manager organizes the defense schedule by selecting jury members, determining the date, and venue of the defense for each project.

1.4 Presentation of the solution

We propose the development of a web application well adapted to the needs of the specialty manager, which will allow students to:

- Invite other students to join their project team.
- Choose a topic to work on from a pre-defined list of options. This feature is designed to help students identify projects that align with their interests and skills.
- Choose a supervisor to guide them through their end of study project. (This feature is designed to help students find the right supervisor who can provide valuable feedback and support throughout the project.)

It will additionally permits speciality manager to:

- Manage student teams and oversee the end of study project progress .
- Manage the list of suggested project themes and topics from other teachers.(This feature is designed to help specialty managers ensure that each project theme is relevant, feasible, and meets the end of study requirements.)
- Set the date of the project presentation
- Assign appropriate jury members to evaluate each team's work

It is worth to state that this web application give all of them (students, teachers) an ability to access a calendar view that displays all of the upcoming end of study project presentation dates

Conclusion

In conclusion, the Faculty of Exact Sciences at Mustapha Stambouli University offers a diverse range of academic programs and research opportunities in the fields of computer science, mathematics, chemistry, and physics. Students are required to complete a final project at the end of each academic career, which is overseen by specialty managers. The web site we propose will allow students to form project teams, choose a topic to work on, and select a supervisor to guide them through the project. The specialty managers will manage the teams, project themes, and presentation dates, and assign appropriate jury members for evaluation. In the following chapter we will analyze this project using the Unified Modeling Language (UML) method.

Application Design

2

Contents

2.1 Introduction 7
2.2 Definition 7
2.3 UML Diagrams 7
2.3.1 Use Case Diagram 7
2.3.2 Detailed use-case diagrams 8
2.3.3 Class Diagram 14
2.3.4 Sequence Diagram 16
2.4 The Relational Model 20
2.5 Site Map 22
2.5.1 Student Pages Site Map: 23
2.5.2 Teacher Pages Site Map: 24
2.5.3 Admin Pages Site Map: 25
Conclusion 25

2.1 Introduction

The success of any software development project depends on how well it is designed and documented. This is where Unified Modeling Language (UML) comes into play. UML is a graphical modeling language that is widely used to specify, visualize, and document software systems. In this chapter, we will introduce the UML diagrams that we have used to model our software system. We will explain the role of each diagram and provide examples of how they are used. The diagrams we will be presenting include the use case diagram, class diagram, sequence diagram, state machine diagram, as well as the component and deployment diagram.

2.2 Definition

Unified Modeling Language (UML) is a standardized graphical modeling language used to specify, visualize, and document software systems. It was created by the Object Management Group (OMG) in 1997 and has since become the de facto standard for software modeling. UML provides a common language for developers, designers, and stakeholders to communicate and understand the architecture and design of a software system. Compared to other modeling languages, such as ER diagrams or flowcharts, UML is more comprehensive and allows for a greater level of detail in system design. While it may have a steeper learning curve, UML is widely used in industry and academia, and there are many resources available for learning UML, including the official OMG UML Specification and various textbooks and online courses. (Object Management Group (OMG), 1997)

2.3 UML Diagrams

2.3.1 Use Case Diagram

the use case diagram is a diagram that shows the relationships among actors and use cases within a system. Use case diagrams are used to capture the functional requirements of a system. Actors represent roles played by users or any other systems that interact with the system being modeled. Use cases represent the functionality of the system being modeled. (Object Management Group (OMG), 2017)

2.3.2 Detailed use-case diagrams

Authentication

Actor	Use Case	Description
User	Login	The user logs in to the system using email and password.
	Change Credentials	The user updates their login credentials by entering new information, including their current password, and is prompted to change their credentials during their initial login.
Admin		Inherits Login and Change Credentials use cases from User.
Teacher		Inherits Login and Change Credentials use cases from User.
Student		Inherits Login and Change Credentials use cases from User.
Authentication System	Verify Credentials	The system verifies the user's email and password to authenticate the user.

Table 2.1 – *Identification Table for Authentication System Use Cases*

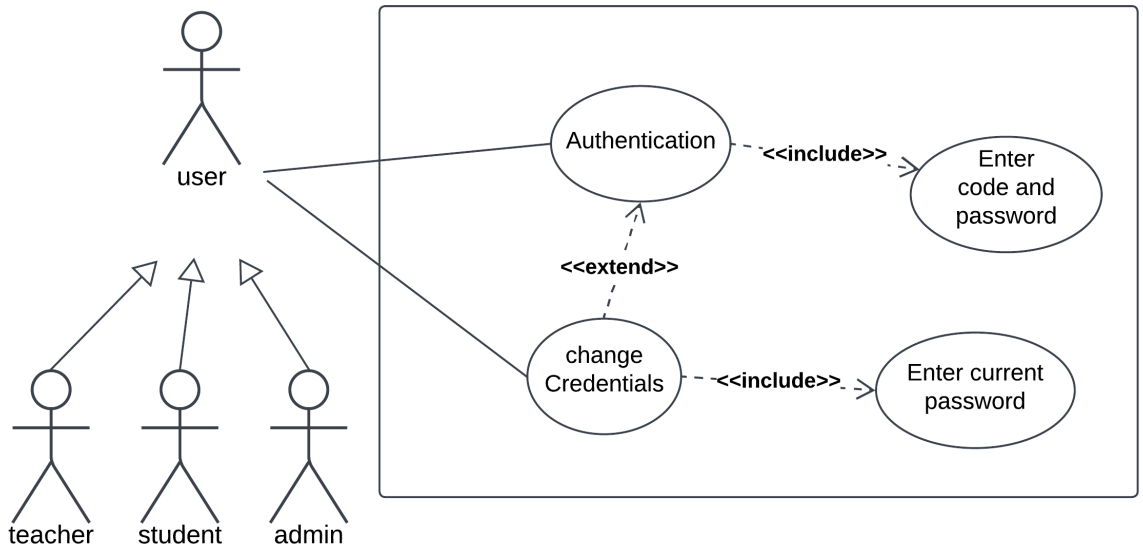


Figure 2.1 – *Use Case Diagram for Authentication System*

Student Actions

Actor	Use Case	Description
Student	Authenticated	Requires the Student actor to be authenticated to perform any of the above actions
	Change Choice List	Allows a student to change their choice list.
	Join Team	Allows a student to join a team
	Consult Result of Affectation	Allows a student to view the results of an affectation process.
	Send Messages	Allows a student to send messages to partner and Supervisor
	Consult Chat Rooms	Allows a student to view chat rooms and received messages
	Consult Appointment Date	Allows a student to view appointment dates
	Accept Invitation	student accept an invitation coming from partner to work in team
	Invite Partner	student invite a partner to join Team
	Consult Presentation Date	Allows a student to view presentation dates
	Consult Ranking	Allows a student to view rankings

Table 2.2 – *Identification table for student actions use cases*

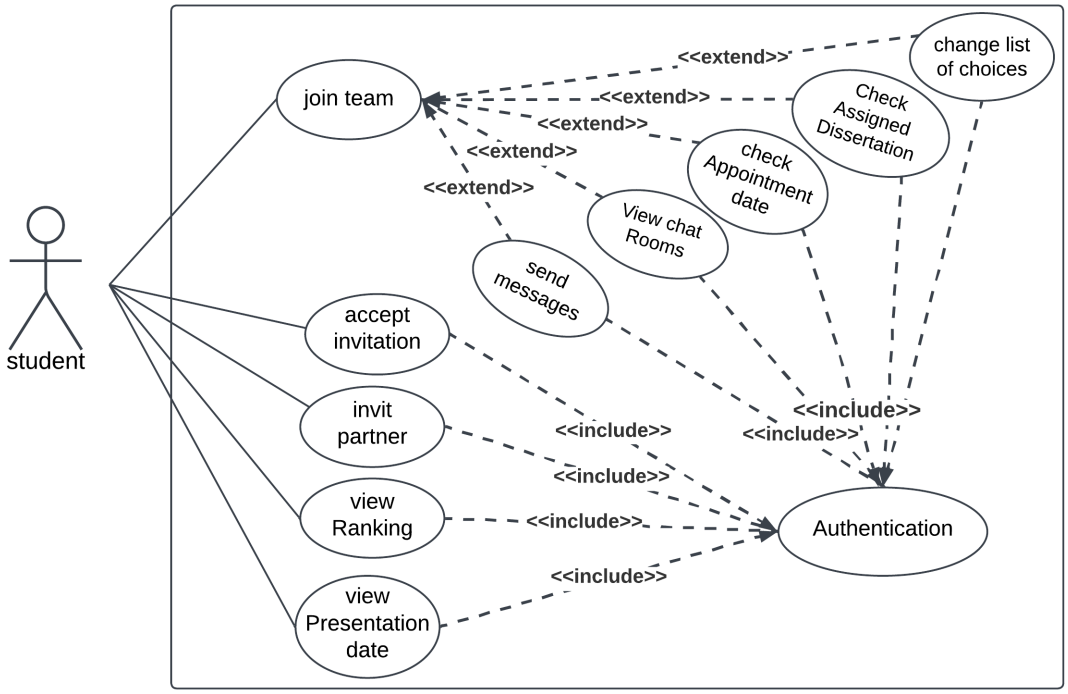


Figure 2.2 – *Use-case Diagram for student actions*

Teacher Actions

Actor	Use Case	Description
Teacher	Consult Presentation Date	Allows a teacher to view presentation dates
	Suggest the Dissertation(Master Degree)	Allows a teacher to suggest a dissertation topic for a master degree
	Supervisor	The Supervisor actor is required for the Suggest Dissertation (Master Degree) use case to be completed
	Create Appointment	Allows a teacher to create an appointment
	Authentication	Requires the Teacher actor to be authenticated to perform any of the above actions
	Fill in the Dissertation Form (for Licence Degree)	Allows a teacher to fill in a dissertation form for a Licence degree
	Consult Chat Rooms	Allows a teacher to view chat rooms
	Send Messages	Allows a teacher to send messages

Table 2.3 – *Identification Table for teacher actions use cases*

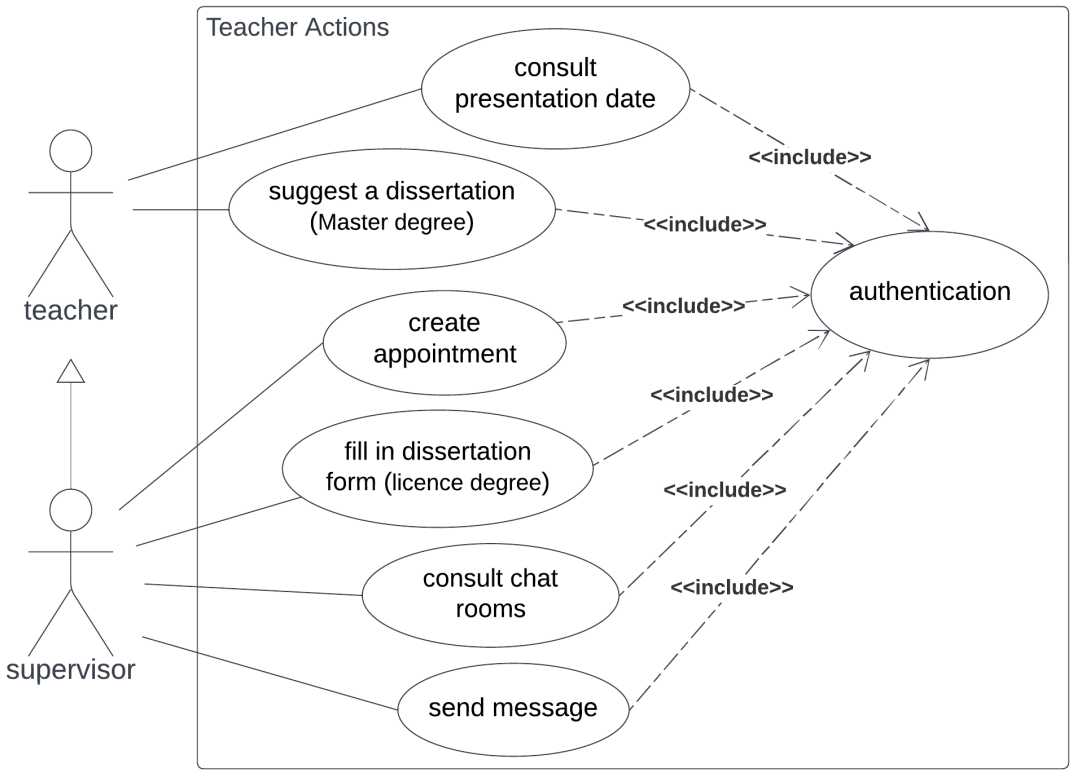


Figure 2.3 – *use case diagram for Teacher Actions*

Specialty Manager

Actor	Use Case	Description
Specialty Manager	Authentication	Allows the specialty manager to authenticate their identity.
	Manage Teams	Extends Create Team use case. Allows the specialty manager to manage the created teams.
	Create Team	Allows the specialty manager to create a new team of one student.
	Create Appointment Period	Allows the specialty manager to create an appointment period for the teams.
	Ranking Management	Allows the specialty manager to add a new ranking for each year scholar.
	Delete Ranking	Allows the specialty manager to delete a ranking.
	Add Ranking	Allows the specialty manager to add a new ranking.
	Create List of Choices	Allows the specialty manager to create a new list of choices, such as a list of dissertation for master degree or list of supervisors for licence degree.
	List of Supervisors	Allows the specialty manager to view the list of Supervisors.
	List of Dissertation	Allows the specialty manager to view the list of dissertation.
	Fixing Student Choices	Extends List of Dissertation use case. Allows the specialty manager to fix the student choices.
	Affect Jury Members to Teams	Allows the specialty manager to affect the jury members to the teams.
	Confirm Theme Suggestion	Allows the specialty manager to confirm the theme suggestion.
	Fixing Presentation Date	Allows the specialty manager to fix the presentation date.

Table 2.4 – *Identification Table for specialty manager actions use cases*

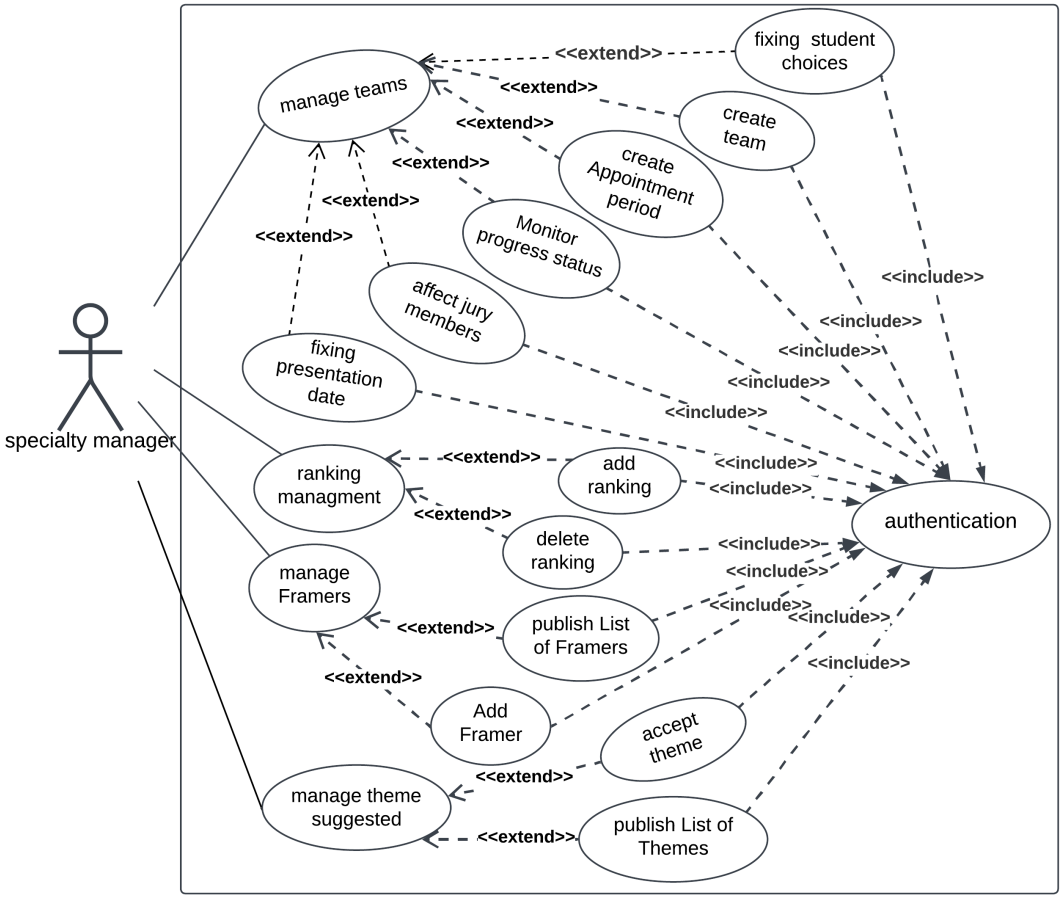


Figure 2.4 – *Specialty Manager Actions Use-case diagram*

Admin action

Actor	Use case	Description
Admin	authentication	involves the authentication and verification of admin credentials, ensuring that only authorized admins can access the system.
	manage student	involves the administration of student-related tasks, such as adding, updating, or deleting student accounts.
	add inscription	allows the admin to add an inscription, which typically involves registering a student for a specialty.
	add	enables the admin to add new accounts, which could be related to students and teachers.
	manage accounts	involves managing user accounts, including tasks such as creating, updating, or deleting user accounts within the system.
	update	allows the admin to update information or data related to specific entities or aspects of the system, such as user profiles or system settings.
	reset	enables the admin to reset or restore certain aspects of the system, resetting passwords and restoring default system configurations.
	manage teacher	focuses on managing teacher-related tasks, such as adding, updating, or deleting teacher accounts within the system.
	delete	allows the admin to delete specific accounts, typically involving the removal of records or data.
	grant role of specialty manager	enables the admin to assign the role of a specialty manager to a teacher, granting them specific privileges and responsibilities related to managing specialties.
	add research areas	involves assign research areas to teacher.

Table 2.5 – *Identification Table for Admin actions use cases*

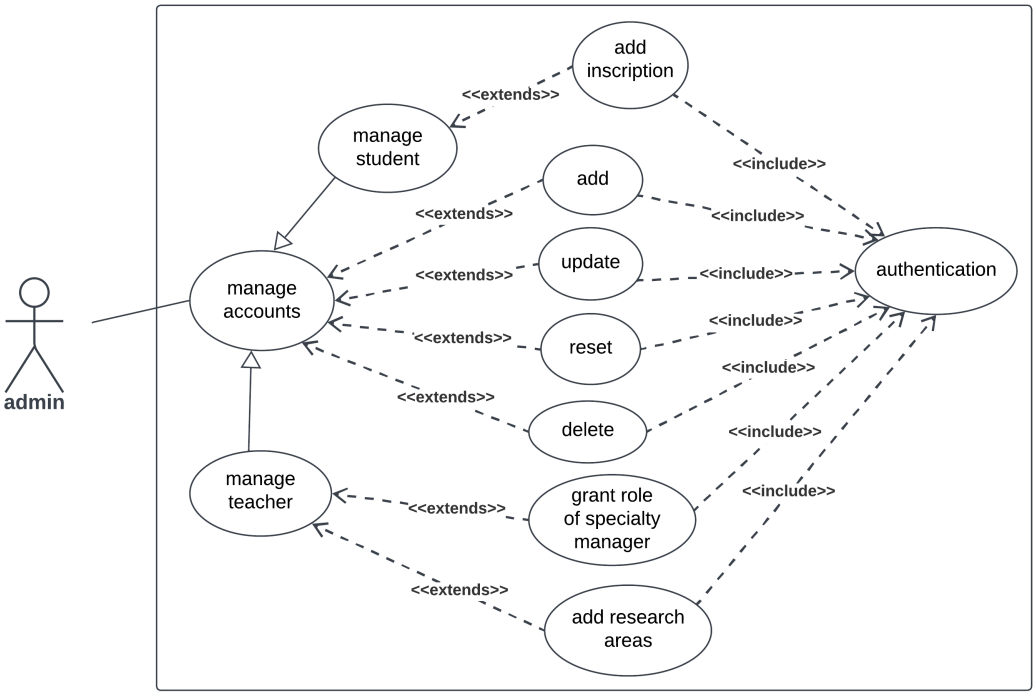


Figure 2.5 – *Admin use-case diagram*

2.3.3 Class Diagram

A class diagram is a type of UML diagram that represents the structure and relationships of classes in a system. It shows the attributes, operations, and associations of each class and how they are related to each other. Class diagrams are used for designing and modeling object-oriented software systems.(Paradigm, n.d.a)

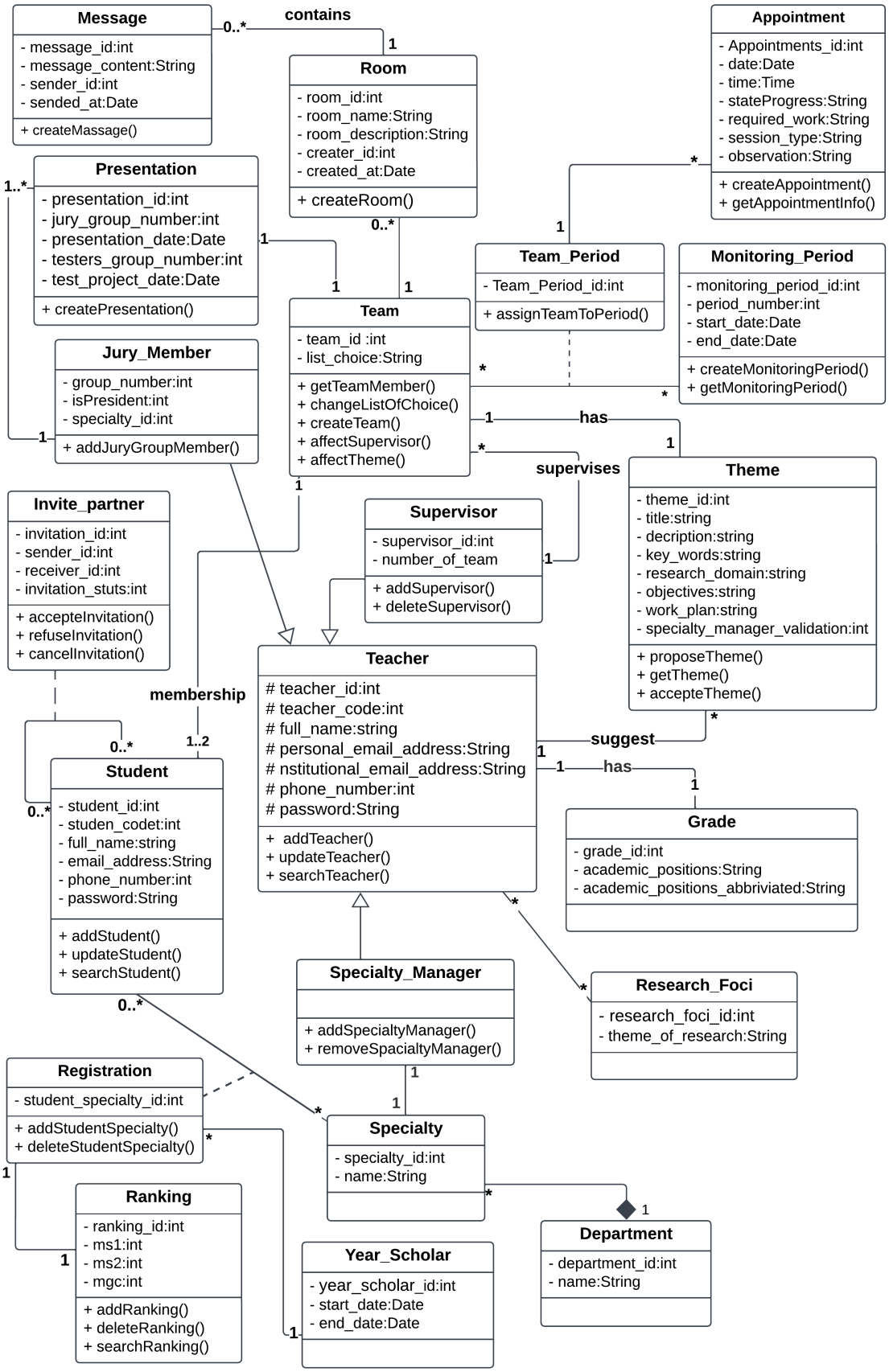


Figure 2.6 – *Class Diagram*

2.3.4 Sequence Diagram

A sequence diagram is a type of UML diagram that shows the interactions between objects in a system over time. It visualizes the flow of messages between objects, as well as the lifeline of each object and the order in which events occur. Sequence diagrams are used to model and document the behavior of a system, and are often used in combination with use case diagrams and class diagrams. (Paradigm, n.d.b)

Sequence diagram For Authentication

Scenario

1. Enter the code or email and password.
2. Make a verification.
3. If the user exist then the system displays the home page
4. If the user visit first time then the system displays change password form

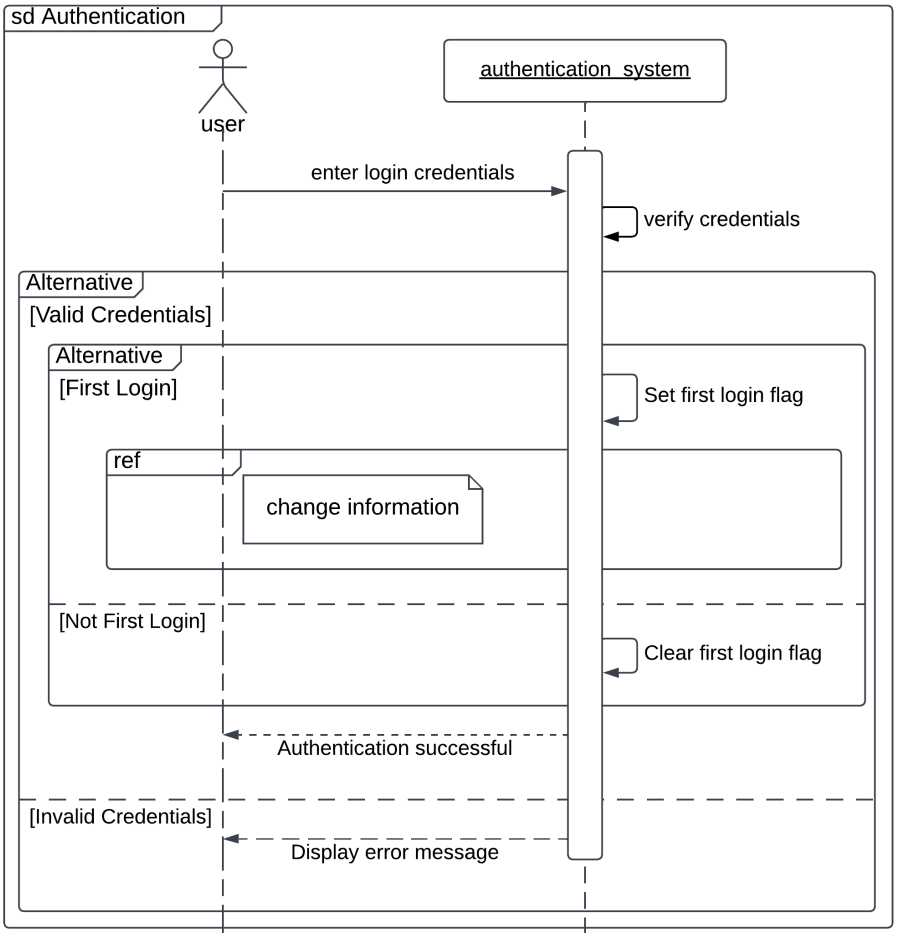


Figure 2.7 – *Sequence Diagram « User Authentication »*

Sequence diagram For Change information

Scenario

1. Enter credentials (email, current password, phone number, new password).
2. Make a verification of password.
3. If the current password is correct then the system will save the new information

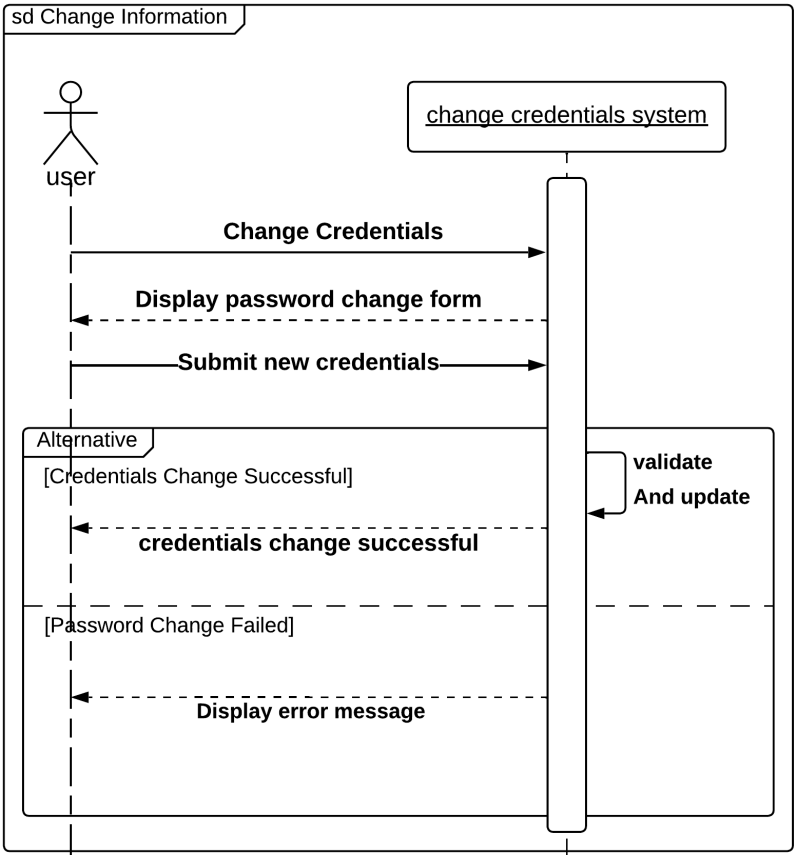


Figure 2.8 – *Sequence Diagram «Change Information»*

Sequence Diagram «Add Teacher»

Scenario

1. Enter teacher information.
2. Make a verification of information.
3. Save teacher information.
4. Send the login credentials to the teacher's inbox email.

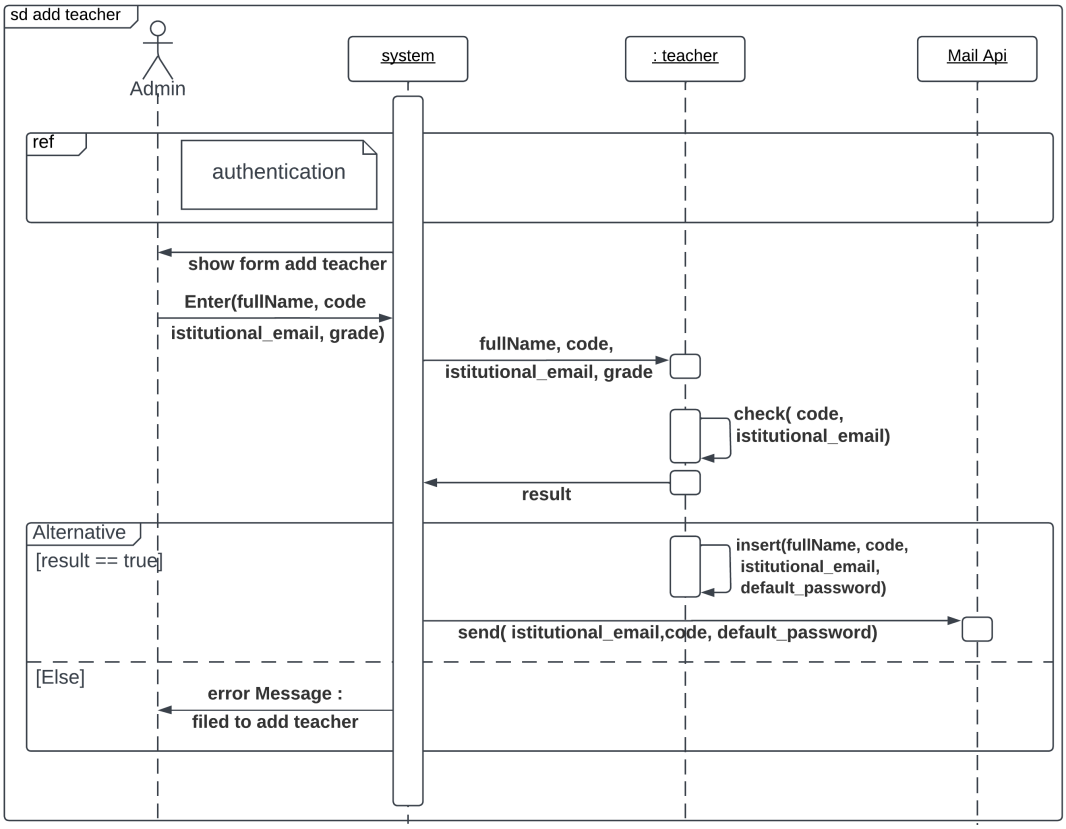


Figure 2.9 – *Sequence diagram for user Add Teacher*

Sequence Diagram «Fixing Student Choices (Master)»

Scenario

1. specialty manager accept dissertations.
2. specialty manager publishes a list of dissertations accepted to teams.
3. Members of teams classify the list of choices.
4. specialty manager fix the student choices.

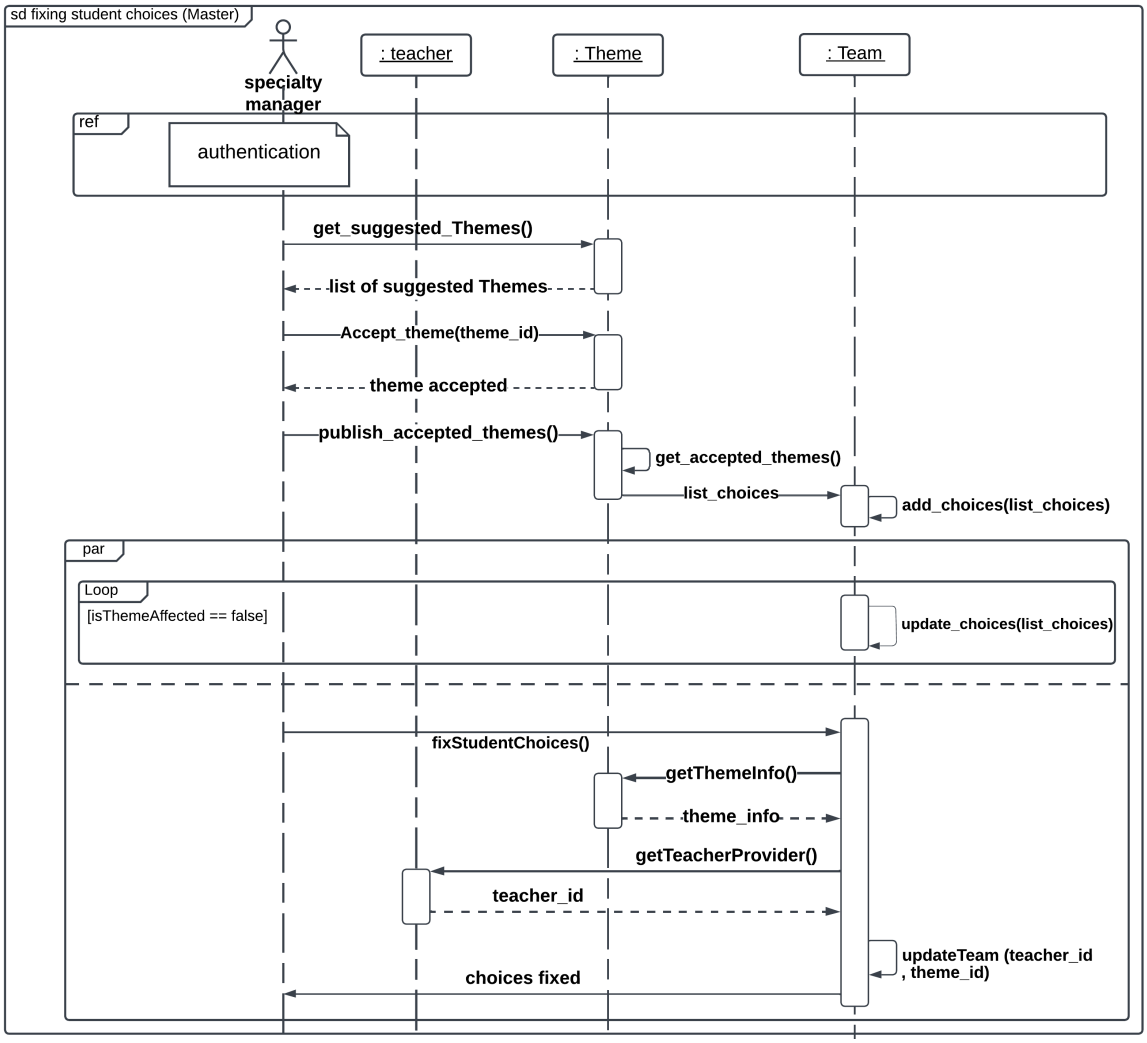


Figure 2.10 – *Sequence Diagram «Fixing Student Choices (Master)»*

Sequence diagram «Fixing Student Choices (licence)»

Scenario

1. specialty manager add supervisors.
2. specialty manager publish list of supervisors.
3. student classify list of choices.
4. specialty manager fix the student choices depend the ranking list.

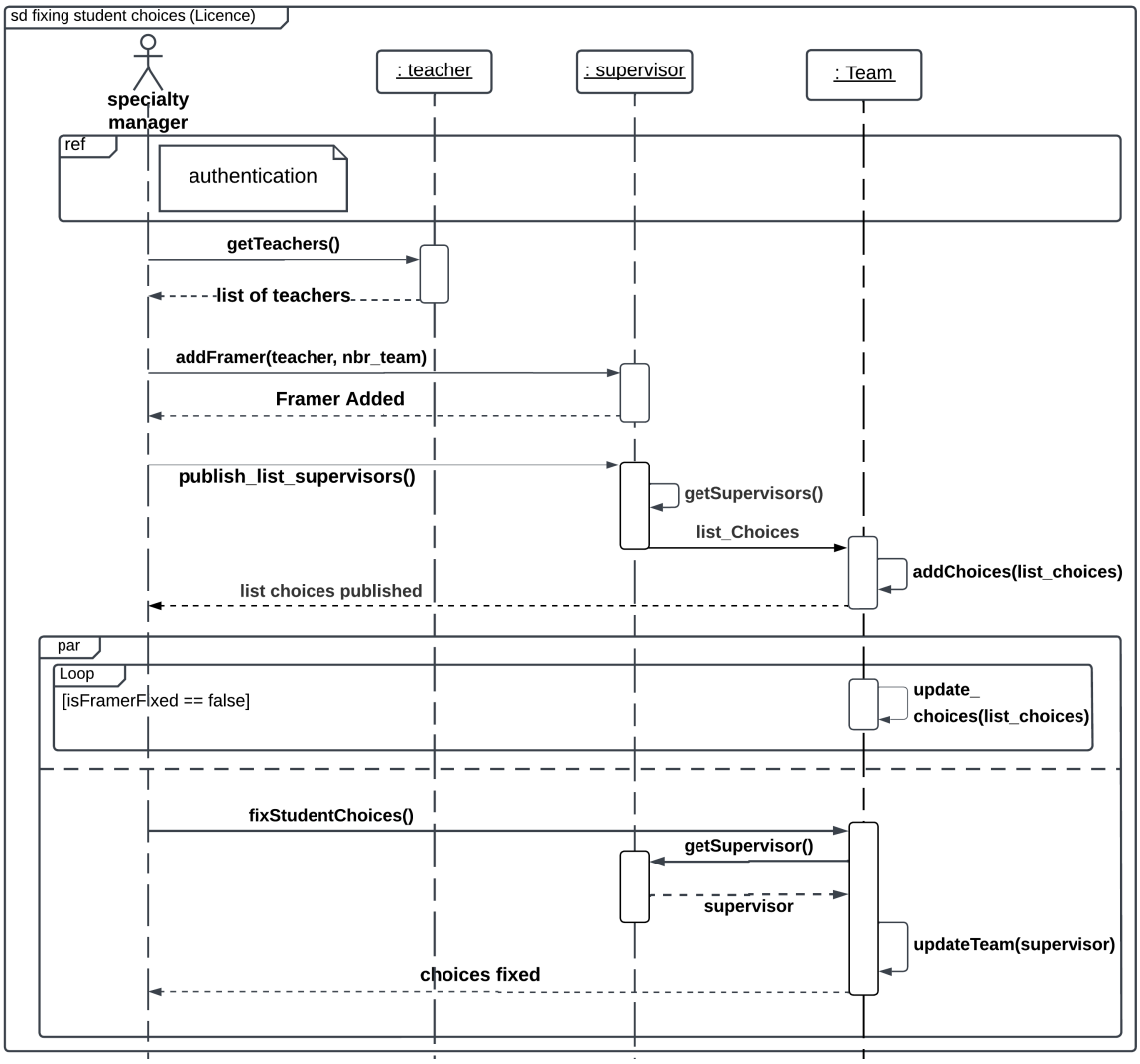


Figure 2.11 – *Sequence diagram «Fixing Student Choices (Licence)»*

2.4 The Relational Model

The relational model is a database model that represents data in the form of relations or tables. In the relational model, data is organized into tables consisting of rows and columns, with each row representing a unique record and each column representing a specific attribute or field.(Oracle, 2008)

The following are the general rules for switching from a class diagram to the relational model

- Identify classes: Identify the classes and their attributes from the class diagram.
- Determine keys: Identify the primary keys for each class.
- Create tables: Create a table for each class, with the class attributes as table columns.

- Define relationships: Determine the relationships between the classes and create foreign keys to represent these relationships.
- Refine data types: Refine the data types for each attribute to ensure they are appropriate for the relational database.

Relational Model

By following the passage rules as described earlier, the resulting relational database schema is as follows:

- Department (dep_id, dep_name)
- Specialty(specialty_id, specialty_fullname, specialty_abbreviated_name, #dep_id)
- Admin(admin_id, email, password)
- Students(#std_id, std_code, std_name, std_phone_number, std_email, password)
- Student_account(std_id, std_name, std_code, default_password, isLogged, logged_At, account_status)
- Student_registration(reg_id, #student_id, #specialty_id, #year_scholar_id)
- Ranks(rank_id, #student_specialty_id, ms1, ms2, mgc, observavtion)
- Teacher(#teacher_id, teacher_code, teacher_name, personal_email, institutional_email, teacher_phone_number, #grade_id, teacher_password)
- Teacher_account(#teacher_id,teacher_name, teacher_code, institutional_email, grade_id, isLogged, logged_at, account_status)
- Grades(grad_id, grade_name, grade_abbreviated_name)
- Supervisors(supervisor_id, nbr_team, #teacher_id,#specialty_id)
- Themes(theme_id, theme_title, theme_description, research_domain, key_word, theme_objectives, work_plans, #specialty_id)
- Invitations(init_id, #sender_id, #reciever_id, status)
- Team(team_id, member_1, member_2, #farmer_id, choice_list, #theme_id, #year_scholar_id)

- Team_appointment(team_appointment_id,#follow_team_id, state_of_progress, required_work, type_of_session, observation)
- Teacher_research_foci(TRF_id, teacher_id, #research_foci_id)
- Research_foci(research_foci_id, theme_of_research)
- Teacher_specialty_manager(TSM_id, #teacher_id, #specialty_id)
- Team_rooms(team_rooms_id, #team_id, room_name, description, creater_id)
- Messages(msg_id, #room_id, sender_id, content, sended_at)
- Team_monitoring_Period(follw_id, #team_id, #period_id)
- Monitoring_Period(period_id, period_number, start_date, end_date)
- Jury_members(jury_id, #teacher_id, group_number, isPresedent, specialty_id)
- Presentation(presentation_id, team_id,jury_group_number, presentation_date, testers_group_number, test_project_date)
- Year_shcolar(year_shcolar_id,start_date,end_date)

note

The student and teacher classes have been separated into two SQL tables for improved security. The "student" and "teacher" tables store personal details like name, code, email, and password. However, certain fields such as name and code are restricted from modification by students or teachers. Any attempts to modify these fields will trigger a process that resets them to their original values from the corresponding "student_account" or "teacher_account" table.

The "student_account" and "teacher_account" tables hold administrative parameters like the default password and account status. Only authorized personnel can access or modify these parameters to ensure the integrity and security of the respective accounts. This table division enables students and teachers to securely access and modify their personal information while preventing unauthorized access or manipulation of critical account parameters.

2.5 Site Map

A site map diagram is a visual representation of the structure of a website, which includes all the pages and the hierarchy of the pages. It allows the users to quickly understand a website's structure and navigation (Techopedia, November 22, 2017). Here are some key features of a site map diagram:

- A site map diagram lists all the pages of a website, including the main pages and subpages.
- It shows the hierarchy of the pages, which means it shows how the pages are related to each other and how they are organized.
- It is a helpful tool for web developers, project managers, writers, and other teammates to collaborate and communicate their ideas effectively.
- Site map diagrams can be used for planning a new website, documenting an existing website, or redesigning a website.

2.5.1 Student Pages Site Map:

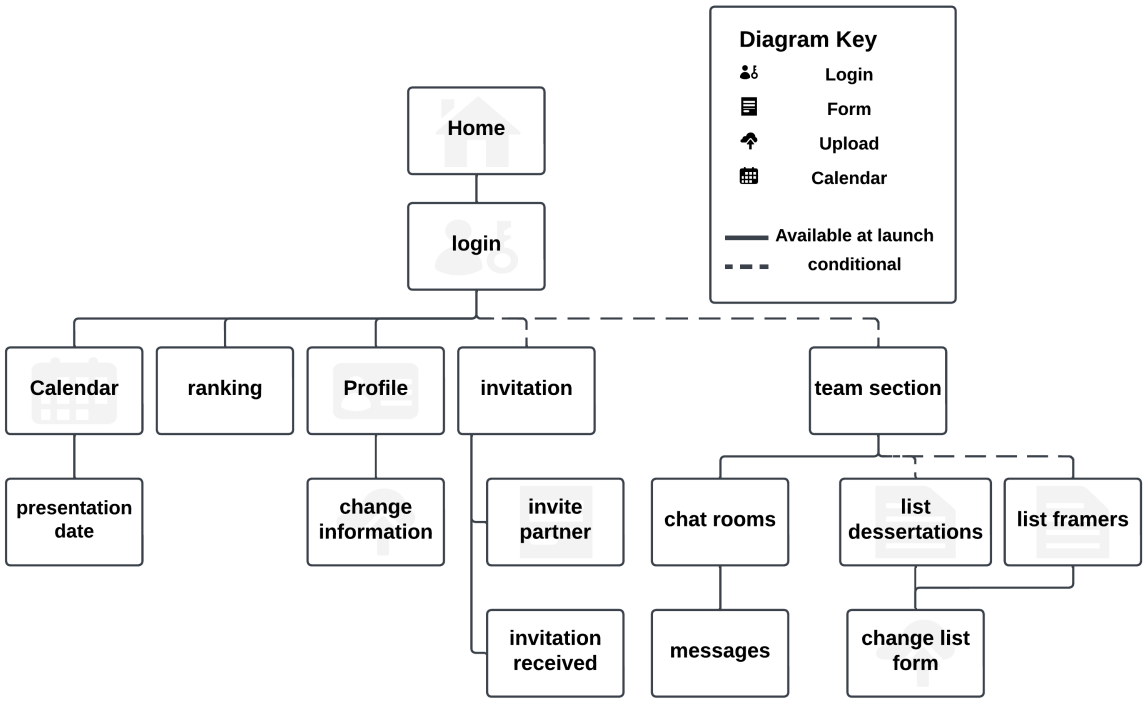


Figure 2.12 – *Student Pages Site Map*

2.5.2 Teacher Pages Site Map:

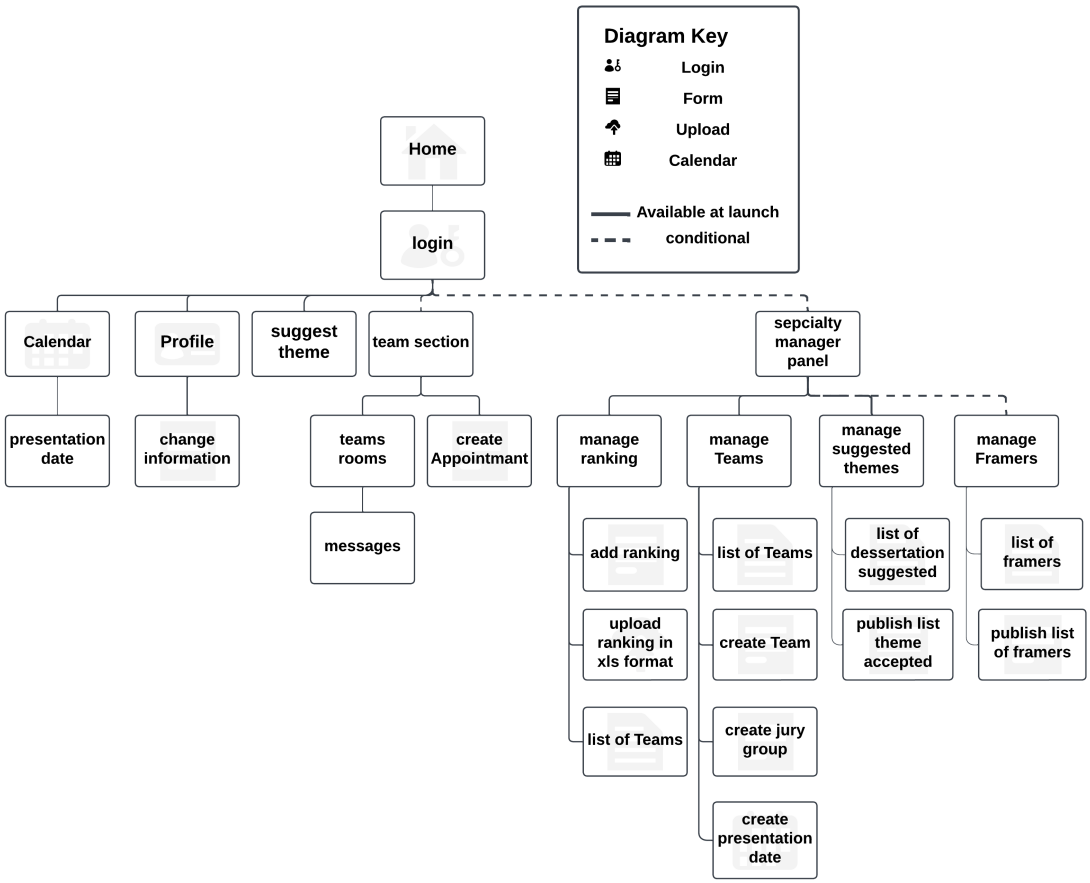


Figure 2.13 – *Teacher Pages Site Map*

2.5.3 Admin Pages Site Map:

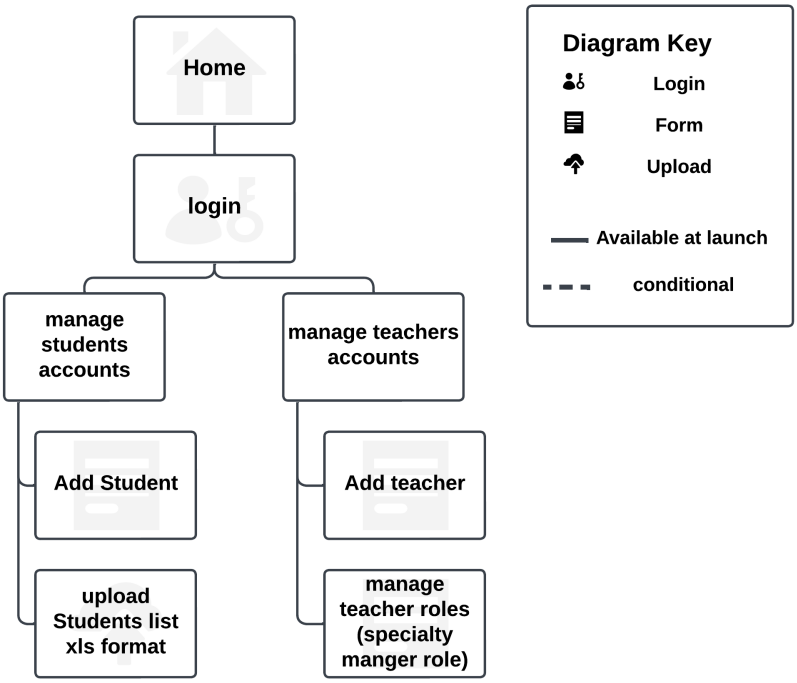


Figure 2.14 – *Admin Pages Site Map*

Conclusion

This chapter highlights the critical role played by the UML specification, Relational model, and Site map in developing robust and efficient information systems. The combination of UML as a modeling language, along with the Relational Model and Site Map, has proven to be highly effective in addressing complex scientific problems. By employing these methodologies, this chapter showcases a comprehensive modeling approach for students, teachers, administrator, and team management through the creation of UML diagrams. The process of translating the design into implementation is a crucial phase in software development, and integrating all modules into a web application establishes an optimal platform for streamlined student data management and secure storage for future reference. Overall, the adoption of UML, the Relational Model, and the Site map contributes to the completeness and effectiveness of the information system, ensuring its success in meeting the intended objectives.

Implementation 3

Contents

3.1 Introduction . . . 27
3.2 Devlopment Tools . . . 27
3.2.1 database development . . . 27
3.2.2 UI development . . . 28
3.2.3 API creation . . . 28
3.2.4 secure the API endpoint . . . 29
3.3 Data Base Schema . . . 30
3.4 Application Pages . . . 33
3.4.1 Home page . . . 33
3.4.2 Login page . . . 33
3.4.3 First Login . . . 34
3.4.4 Profile page . . . 35
3.4.5 Ranking page . . . 35
3.4.6 Invite page . . . 35
3.4.7 Team section page . . . 36
3.4.8 Create Appointment . . . 37
3.4.9 Suggest a dissertation . . . 38
3.4.10 Ranking management . . . 39
3.4.11 Suggested Themes Management . . . 40
3.4.12 Supervisors Management . . . 40
3.4.13 Team management . . . 41
3.4.14 Calendar page . . . 43
3.4.15 Admin Page . . . 44
Conclusion . . . 46

3.1 Introduction

In the last phase of information system design, the system is implemented by writing programs that fulfill the university's requirements. This phase involves using a DBMS and application software to manage the database and physically represent the data. To achieve this, we opted for MySQL to store the database, while React js is used for the user interface. Laravel API is chosen to handle requests, and JWT is implemented for user authorization management.

3.2 Devlopment Tools

Our web application was developed using various tools, which can be categorized into three sections: database development, API creation, and UI design, Additionally, we implemented JWT to secure the API endpoint.

3.2.1 database development

PhpMyAdmin

PhpMyAdmin is a free and open source administration tool for MySQL and MariaDB. It is a web-based application written primarily in PHP, which makes it a popular MySQL administration tool, especially for web hosting services (Wikipedia, 31 March 2023).

PhpMyAdmin allows users to perform a wide range of operations on MySQL and MariaDB databases, such as managing databases, tables, columns, relations, indexes, users, permissions, stored procedures, and triggers. These operations can be performed via the user interface, while users still have the ability to directly execute any SQL statement (phpMyadmin, n.d.).

Figure 3.1 – *PhpMyAdmin Logo*

MySQL

MySQL is an open-source relational database management system that is widely used in the industry. It is a digital store that collects data and organizes it according to the relational model. MySQL is integral to many of the most popular software stacks for building and maintaining everything from customer-facing web applications to powerful, data-driven B2B services. Its open-source nature, stability, and rich feature set, paired with ongoing development and support from Oracle, have meant that internet-critical organizations such as Facebook, Flickr, Twitter, Wikipedia, and YouTube all employ MySQL backends. (Oracle, n.d.) (talend, n.d.)

Figure 3.2 – *MySql Logo*

What makes MySQL a great database option

There are several reasons why one might choose MySQL as their relational database management system:

- Open-source: MySQL is an open-source database system, which means it is free to download, use and modify. This makes it a popular choice for small and large businesses alike.
- High-performance: MySQL is known for its fast performance and ability to handle large volumes of data. It is designed to be scalable, allowing for easy expansion as your data grows.
- Cross-platform compatibility: MySQL can be used on a wide range of platforms, including Windows, Linux, and macOS. This makes it a flexible choice for developers who may be working across different operating systems.

3.2.2 UI development

ReactJS, also known as React or React.js, is a free and open-source JavaScript library for building user interfaces based on components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies React was created by Jordan Walke, a software engineer at Facebook, in 2011, and its first application was for Facebook's newsfeed (Chinmayee Deshpande, Feb 7, 2023) React is primarily used to build User Interfaces (UI) that enhance the speed of applications.(**?**).

Figure 3.3 – *React JS library Logo*

3.2.3 API creation

Laravel is a PHP-based open-source web application framework used for building web applications and APIs. Laravel provides a robust set of tools and features for building APIs that are secure, scalable, and easy to maintain.

Figure 3.4 – *Laravel Framework Logo*

The reason for using Laravel API are:

- Rapid development: Laravel provides a rich set of tools and features that make it easy to build APIs quickly. It has a large community of developers who contribute packages and plugins that can be easily integrated into your application.
- Security: Laravel provides built-in security features such as CSRF protection, encryption, and password hashing. It also supports OAuth2, JWT, and other authentication mechanisms.
- Laravel middleware is a powerful tool for filtering HTTP requests and responses. It allows you to modify requests and responses as they pass through your application, providing a convenient way to implement authentication, rate limiting, and other security measures.

3.2.4 secure the API endpoint

Figure 3.5 – *JWT Logo*

JWT (JSON Web Tokens) is a standard method for securely transmitting information between parties as a JSON object. JWT is used to authenticate users and transmit data securely between applications. The token is a string that includes a header, a payload, and a signature. The header describes the algorithm used to sign the token, the payload contains the data, and the signature is used to verify the authenticity of the token. Working with JWT typically involves generating and validating tokens. The process typically involves:

- JWT is signed using a secret key or public/private key pair, which makes it more difficult for attackers to tamper with or forge a token.
- when the client sends a request to the server, the server checks the token's signature to ensure that it has not been tampered with. If the token is valid, the server extracts the user's ID and any additional data required to perform the requested action.
- JWT can also be configured with an expiration time, after which the token is no longer valid. In this case, the user must log in again to obtain a new token.

3.3 Data Base Schema

Our database comprises multiple tables, each dedicated to storing specific information about students, teachers, specialties, departments academic years, and more.

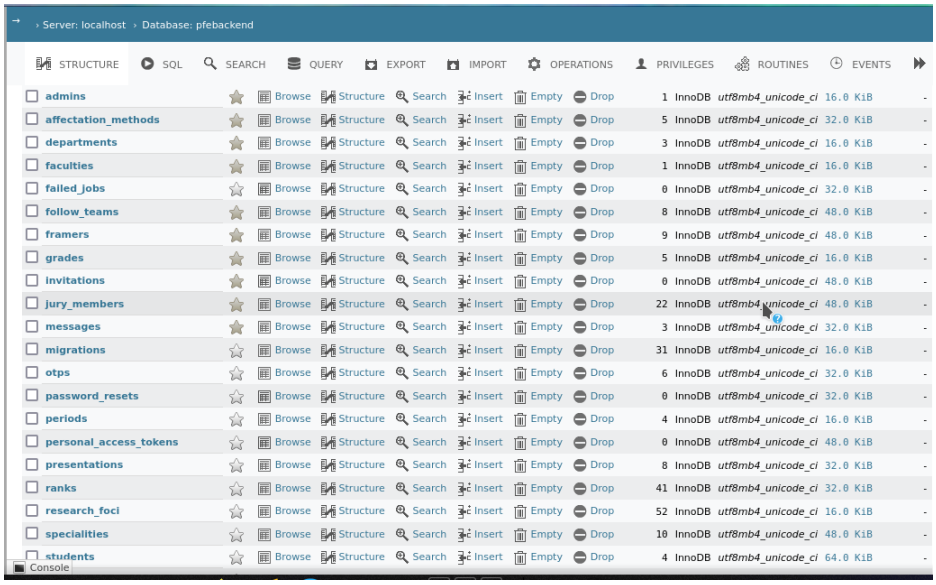


Figure 3.6 – *List of Tables*

The Students table are one of this tables serves as a repository for student information, capturing essential details such as student's code, name, and scholar email phone number and password.

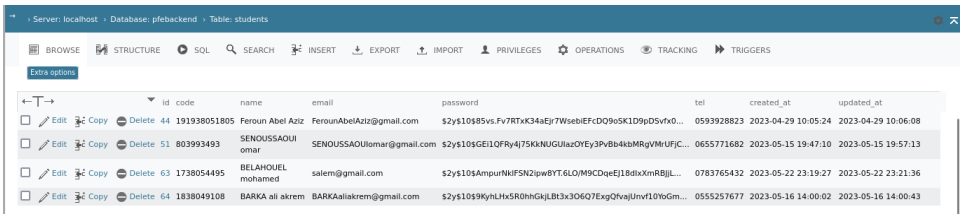


Figure 3.7 – *Table of Students*

The Teachers table contains essential information about our teaching staff, including unique teacher codes, names, personal email addresses, institutional email addresses, telephone numbers, assigned grade IDs, and and encrypted passwords. We prioritize security by encrypting the passwords, ensuring the protection of sensitive information.

› Server: localhost › Database: pfebackend › Table: teachers

BROWSE STRUCTURE SQL SEARCH INSERT EXPORT IMPORT PRIVILEGES OPERATIONS TRACKING TRIGGERS

	id	code	name	personal_email	institutional_email	tel	grade_id	password
Edit Copy Delete	40	987654321	mahi faiza	mahi.faiza@gmail.com	soulimanali2323@gmail.com	0692372823	2	$2y$10$60SjiztoyA9WuB1mbkl9lusqDMMShwjhaKbYQ1/Wlnn...
Edit Copy Delete	45	65148759	boufera fatma	rchamplin@example.org	fboufera@univ-mascara.dz	0555253643	2	$2y$10$rIijAy55lHLSNaaCJ6Npb.BVdZA1vrn..Le0PVpAUWq...
Edit Copy Delete	46	36636768	MEDDEBER Hayat	NULL	hayet.meddeber@univ-mascara.dz	NULL	4	$2y$10$l/HLay.kVU1ZNtbE6Xpm5.6dD.wKNd5MfAz0DhVPUYJ...
Edit Copy Delete	47	48911735	Mokhtari Chakir	mokhtariCkakir@gmail.com	mokhtari.c@univ-mascara.dz	0555304343	2	$2y$10$a9nDIO5QvCLeqcGZIYeBUuJiRMi5voONorSbn8z6PSE...
Edit Copy Delete	48	65292574	YAHIAOUI Ahmed	NULL	a.yahiaoui@univ-mascara.dz	NULL	1	$2y$10$Oms8RN0YmGksx/Yhd8Hjzuayz00S5nFLak//HjMzyOq...
Edit Copy Delete	49	36887029	ABID Sabrina	abidSabrina@gmail.com	sabrina.abid@univ-mascara.dz	0555383422	3	$2y$10$JVJoR5MNC.bjv0EXutOsv.bj5AHvylfiE6DXBhQuLvB...
Edit Copy Delete	50	55243415	azzaz amina	azzazamina@gmail.com	amina.azzaz@univ-mascara.dz	0774772382	2	$2y$10$q0wmcfSueRJlavcpbCFWJO2ICuioRht7aYQLgyfG3ve...
Edit Copy Delete	51	61391000	BAKHTI Benaoumeur	NULL	benaoumeur.bakhti@univ-mascara.dz	NULL	2	$2y$10$92IXUNpkjO0rOQ5byMi.Ye4oKoEa3Ro9llC/.og/at2...

Figure 3.8 – *Table of Teachers*

The Teams table is a central repository that stores information about different teams within our institution, including the names of team members, supervisor IDs, choice lists, scholar years, and additional relevant details

› Server: localhost › Database: pfebackend › Table: teams

BROWSE STRUCTURE SQL SEARCH INSERT EXPORT IMPORT PRIVILEGES OPERATIONS TRACKING TRIGGERS

Extra options

	id	member_1	member_2	supervisor_id	choice_list	theme_id	year_scholar_id	created_at	updated_at
Edit Copy Delete	88	36	NULL	66	[402,407,408,409,410,411]	407	2	2023-05-04 19:36:17	2023-05-16 00:47:37
Edit Copy Delete	89	38	NULL	59	[402,407,408,409,410,411]	408	2	2023-05-04 19:36:25	2023-05-16 00:47:37
Edit Copy Delete	99	44	51	49	[402,407,408,409,410,411]	402	2	2023-05-15 21:03:07	2023-05-16 00:47:37
Edit Copy Delete	100	64	63	NULL	[40,45,46,47,49,50,51,53,54,55,56,57]	NULL	2	2023-05-18 15:36:42	NULL
Edit Copy Delete	125	59	60	66	[402,407,408,409,410,411]	407	2	NULL	NULL
Edit Copy Delete	126	76	73	65	[402,407,408,409,410,411]	409	2	NULL	NULL
Edit Copy Delete	127	46	74	56	[402,407,408,409,410,411]	403	2	NULL	NULL
Edit Copy Delete	128	37	39	59	[402,407,408,409,410,411]	411	2	NULL	NULL
Edit Copy Delete	129	75	77	61	[402,407,408,409,410,411]	NULL	2	NULL	NULL
Edit Copy Delete	130	57	50	55	[402,407,408,409,410,411]	NULL	2	NULL	NULL
Edit Copy Delete	131	62	45	63	[40,45,46,47,49,50,51,53,54,55,56,57]	409	2	NULL	NULL
Edit Copy Delete	132	43	NULL	66	[40,45,46,47,49,50,51,53,54,55,56,57]	NULL	2	NULL	NULL
Edit Copy Delete	133	79	NULL	66	[40,45,46,47,49,50,51,53,54,55,56,57]	407	2	2023-05-16 21:17:41	2023-05-16 21:17:41
Edit Copy Delete	134	78	NULL	67	[40,45,46,47,49,50,51,53,54,55,56,57]	409	2	2023-05-16 21:17:47	2023-05-16 21:17:47
Edit Copy Delete	135	80	NULL	67	[40,45,46,47,49,50,51,53,54,55,56,57]	403	2	2023-05-16 21:17:58	2023-05-16 21:17:58
Edit Copy Delete	137	81	NULL	65	[40,45,46,47,49,50,51,53,54,55,56,57]	NULL	2	2023-05-16 21:18:04	2023-05-16 21:18:04
Edit Copy Delete	138	82	NULL	67	[40,45,46,47,49,50,51,53,54,55,56,57]	411	2	2023-05-16 21:18:11	2023-05-16 21:18:11
Edit Copy Delete	139	83	NULL	68	[40,45,46,47,49,50,51,53,54,55,56,57]	403	2	2023-05-16 21:18:12	2023-05-16 21:18:12
Edit Copy Delete	140	84	NULL	50	[40,45,46,47,49,50,51,53,54,55,56,57]	410	2	2023-05-16 21:18:17	2023-05-16 21:18:17

Figure 3.9 – *Table of Teams*

Within the 'Jury Members' table, we maintain vital information about the esteemed individuals comprising our jury panel. This includes their unique teacher IDs, corresponding group numbers, specialized areas of expertise, and any additional pertinent details

	id	teacher_id	group_number	isPresident	specialty_id	created_at	updated_at
Edit Copy Delete	456	40	1	1	2	2023-05-16 22:22:51	2023-05-16 22:22:51
Edit Copy Delete	457	45	1	0	2	2023-05-16 22:22:51	2023-05-16 22:22:51
Edit Copy Delete	458	46	1	0	2	2023-05-16 22:22:51	2023-05-16 22:22:51
Edit Copy Delete	459	47	1	0	2	2023-05-16 22:22:51	2023-05-16 22:22:51
Edit Copy Delete	460	53	2	0	2	2023-05-16 22:22:51	2023-05-16 22:22:51
Edit Copy Delete	461	59	2	0	2	2023-05-16 22:22:51	2023-05-16 22:22:51
Edit Copy Delete	462	40	1	1	1	2023-05-16 22:24:15	2023-05-16 22:24:15
Edit Copy Delete	463	45	1	0	1	2023-05-16 22:24:15	2023-05-16 22:24:15
Edit Copy Delete	464	46	1	0	1	2023-05-16 22:24:15	2023-05-16 22:24:15
Edit Copy Delete	465	49	1	0	1	2023-05-16 22:24:15	2023-05-16 22:24:15
Edit Copy Delete	466	46	2	0	1	2023-05-16 22:24:15	2023-05-16 22:24:15
Edit Copy Delete	467	45	2	0	1	2023-05-16 22:24:15	2023-05-16 22:24:15
Edit Copy Delete	468	51	3	1	1	2023-05-16 22:37:08	2023-05-16 22:37:08
Edit Copy Delete	469	50	3	0	1	2023-05-16 22:37:08	2023-05-16 22:37:08
Edit Copy Delete	470	53	3	0	1	2023-05-16 22:37:08	2023-05-16 22:37:08
Edit Copy Delete	471	47	3	0	1	2023-05-16 22:37:08	2023-05-16 22:37:08
Edit Copy Delete	472	51	4	0	1	2023-05-16 22:37:08	2023-05-16 22:37:08
Edit Copy Delete	473	47	4	0	1	2023-05-16 22:37:08	2023-05-16 22:37:08

Figure 3.10 – *Table of Jury members*

The Ranks table stores important information about student rankings, including student speciality IDs, grade averages, observations, and additional relevant details

› Server: localhost › Database: pfebackend › Table: ranks

BROWSE STRUCTURE SQL SEARCH INSERT EXPORT IMPORT PRIVILEGES

	id	student_specialite_id	ms1	ms2	mgc	observation	created_at	updated_at
Edit Copy Delete	244	187	11.00	12.00	11.00	1	2023-04-28 14:02:56	2023-04-28 14:02:56
Edit Copy Delete	245	193	12.00	9.00	11.00	1	2023-04-28 14:02:56	2023-04-28 14:02:56
Edit Copy Delete	249	207	9.00	13.00	10.00	1	2023-04-28 14:02:56	2023-04-28 14:02:56
Edit Copy Delete	250	204	10.00	11.00	14.00	1	2023-04-28 14:02:56	2023-04-28 14:02:56
Edit Copy Delete	251	202	10.00	14.00	12.00	4	2023-04-28 14:02:56	2023-04-28 14:02:56
Edit Copy Delete	252	192	9.00	11.00	12.00	4	2023-04-28 14:02:56	2023-04-28 14:02:56
Edit Copy Delete	253	162	12.00	10.00	12.00	3	2023-04-28 14:02:56	2023-04-28 14:02:56
Edit Copy Delete	262	194	14.00	13.00	12.00	1	2023-04-28 14:03:38	2023-04-28 14:03:38
Edit Copy Delete	265	205	14.00	11.00	10.00	2	2023-04-28 14:03:39	2023-04-28 14:03:39
Edit Copy Delete	266	206	11.00	13.00	11.00	1	2023-04-28 14:03:39	2023-04-28 14:03:39
Edit Copy Delete	267	211	13.00	14.00	10.00	4	2023-04-28 14:03:39	2023-04-28 14:03:39
Edit Copy Delete	268	199	12.00	9.00	11.00	1	2023-04-28 14:03:39	2023-04-28 14:03:39
Edit Copy Delete	270	188	0.00	0.00	0.00	1	2023-04-28 15:02:34	2023-04-28 15:02:34
Edit Copy Delete	273	188	0.00	0.00	0.00	1	2023-04-28 15:05:02	2023-04-28 15:05:02
Edit Copy Delete	274	188	0.00	0.00	0.00	1	2023-04-28 15:05:21	2023-04-28 15:05:21
Edit Copy Delete	276	189	12.00	13.00	14.00	3	2023-04-28 15:56:47	2023-04-28 15:56:47
Edit Copy Delete	279	195	18.00	19.00	20.00	3	2023-04-28 15:56:47	2023-04-28 15:56:47
Edit Copy Delete	280	196	19.00	20.00	20.00	1	2023-04-28 15:56:47	2023-04-28 15:56:47
Edit Copy Delete	281	197	20.00	12.00	20.00	2	2023-04-28 15:56:47	2023-04-28 15:56:47
Edit Copy Delete	283	200	12.00	13.00	14.00	2	2023-04-28 15:56:47	2023-04-28 15:56:47

Console

Figure 3.11 – *Table of Ranks*

3.4 Application Pages

3.4.1 Home page

- This is the first page that you will find when you enter the web site

Figure 3.12 – *Home Page*

3.4.2 Login page

- Student, Teacher and Admin can both login.
- They can login using code or email, and password

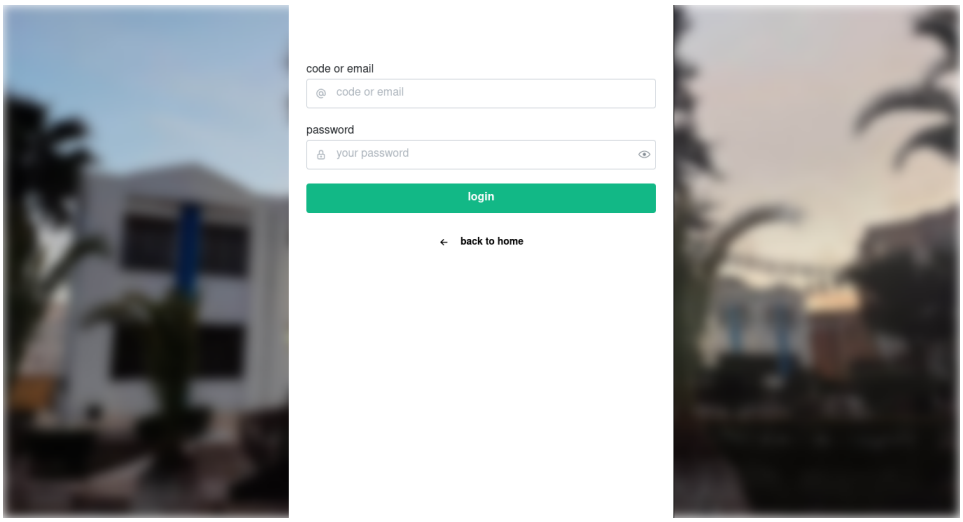


Figure 3.13 – *Login Page*

- Teachers will receive an email containing teacher's account details and password, granting them access to the online platform.

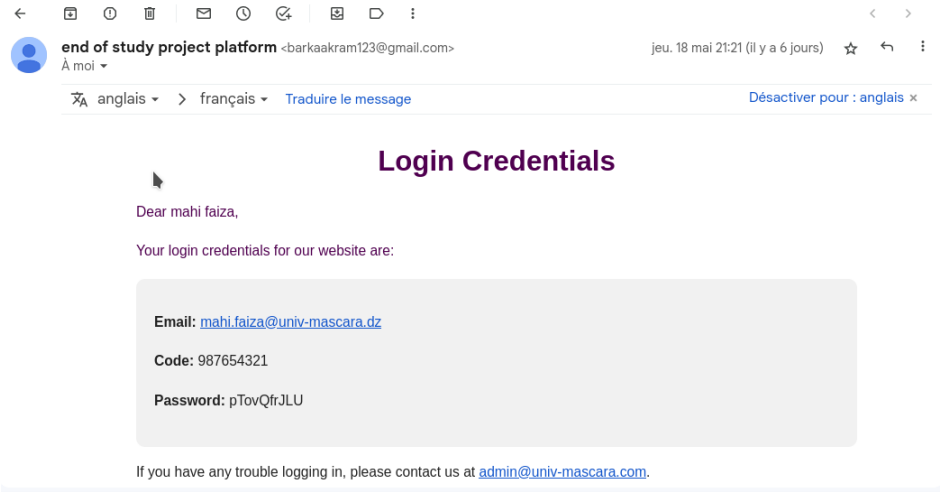


Figure 3.14 – *email containing their account details and password*

3.4.3 First Login

- If the user is logging in for the first time, they will be prompted to change their password

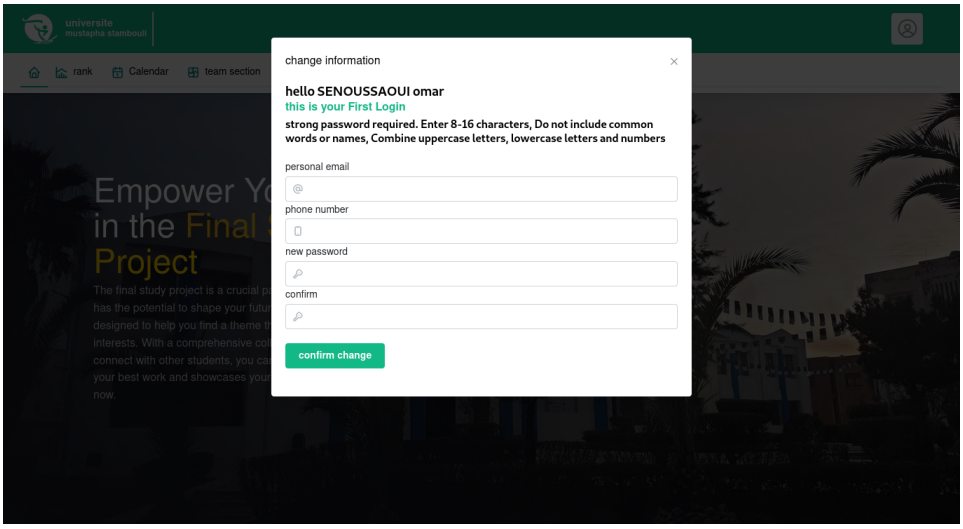


Figure 3.15 – *Change Password in first log in*

3.4.4 Profile page

- User can change his phone number, personal email and password.

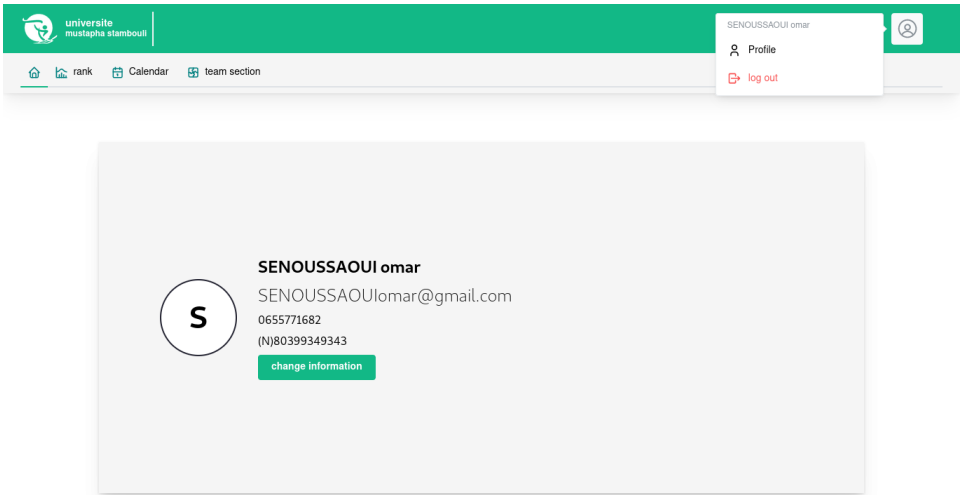
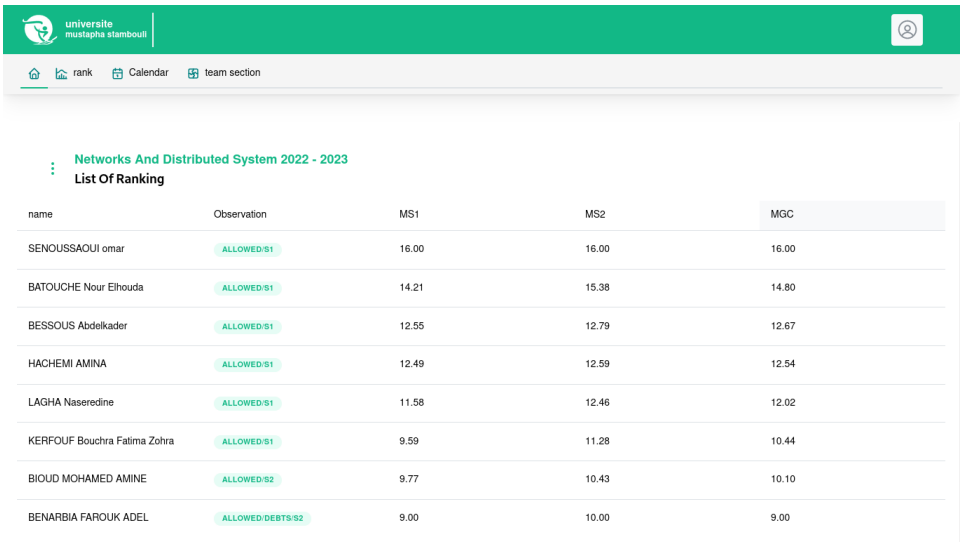


Figure 3.16 – *Profile Page*

3.4.5 Ranking page

- Student can view the ranking of their fellow classmates.



name	Observation	MS1	MS2	MGC
SENOUSSAOUI omar	ALLOWED/S1	16.00	16.00	16.00
BATOUCHE Nour Elhouda	ALLOWED/S1	14.21	15.38	14.80
BESSOUS Abdelkader	ALLOWED/S1	12.55	12.79	12.67
HACHEMI AMINA	ALLOWED/S1	12.49	12.59	12.54
LAGHA Naseredine	ALLOWED/S1	11.58	12.46	12.02
KERFOUF Bouchra Fatima Zohra	ALLOWED/S1	9.59	11.28	10.44
BIOUD MOHAMED AMINE	ALLOWED/S2	9.77	10.43	10.10
BENARBIA FAROUK ADEL	ALLOWED/DEBTS/S2	9.00	10.00	9.00

Figure 3.17 – *Ranking Page*

3.4.6 Invite page

- Student has the option to invite their study partner using his code to collaborate on the project.

- He also has the option to receive an invitation from other student to participate in the project.

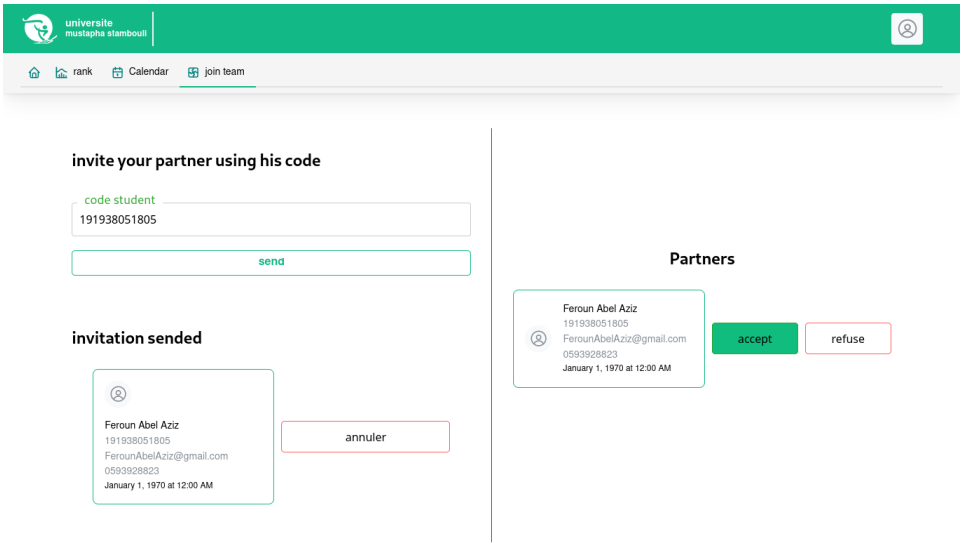


Figure 3.18 – *Invite Page*

3.4.7 Team section page

List Supervisors

- Each team can select their preferred supervisors for the project.
- They can change the Supervisors list anytime (As long as the deadline is not missed)

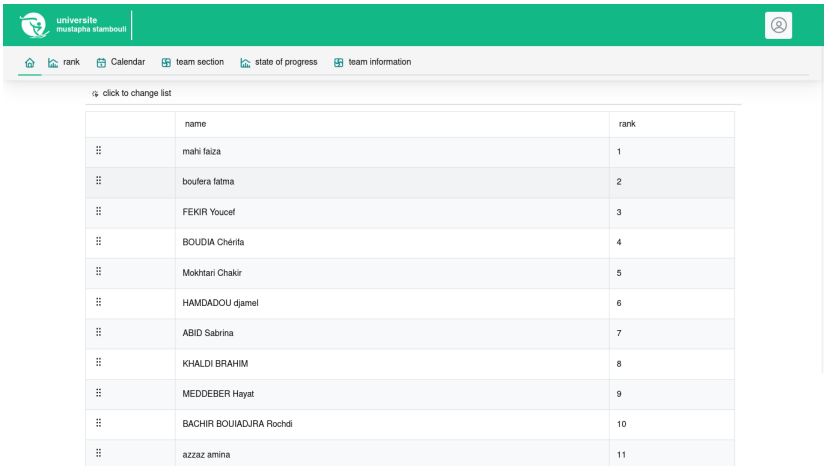


Figure 3.19 – *Supervisors list Page*

List dissertations

- Each team has the option to choose their preferred dissertation topic for the project.
- They can change the list anytime (As long as the deadline is not missed)

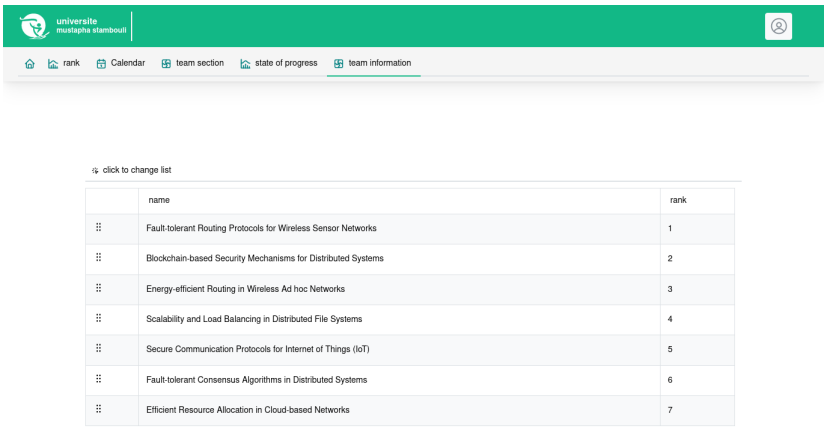


Figure 3.20 – *dissertations list Page*

Chat rooms

- Each team has the option to communicate with the Supervisor and discuss about project

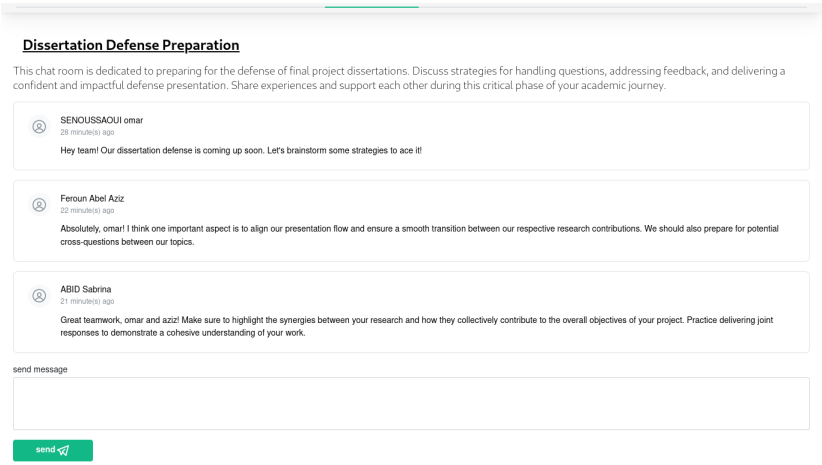


Figure 3.21 – *Chat room Page*

3.4.8 Create Appointment

- Supervisor has the ability to create appointments for their students to meet with them

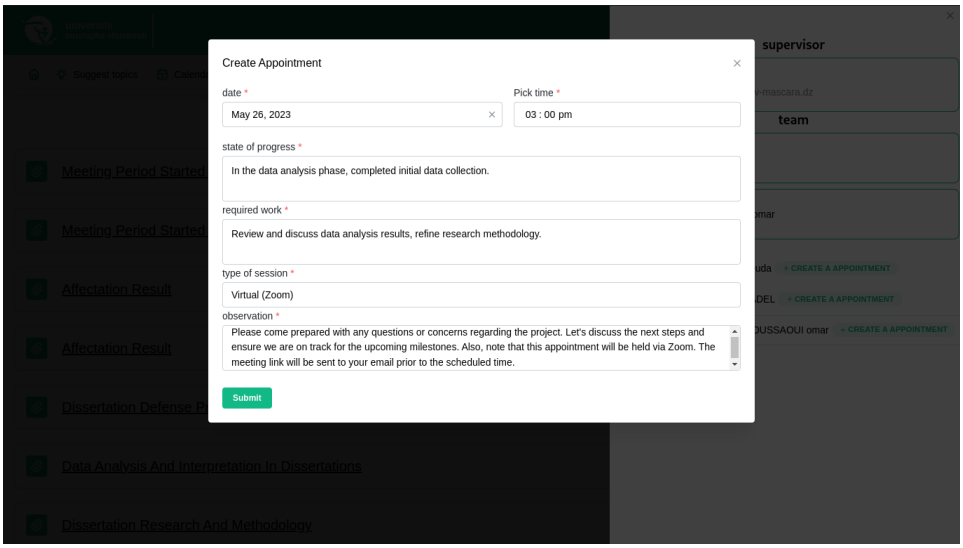


Figure 3.22 – *Create Appointment*

3.4.9 Suggest a dissertation

- Teacher has the ability to suggest a dissertation for Licence's or Master's degree

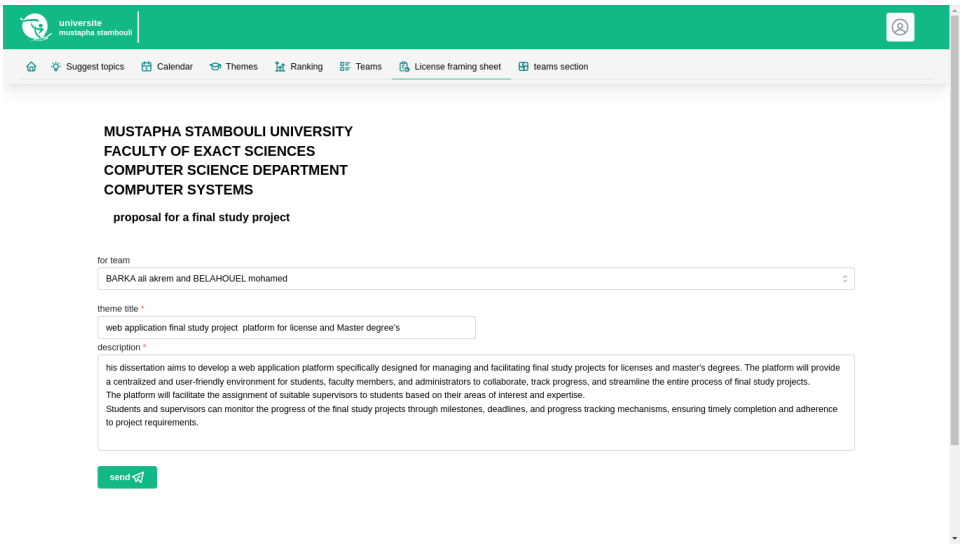


Figure 3.23 – *Example Of Suggest a Licence Dissertation*

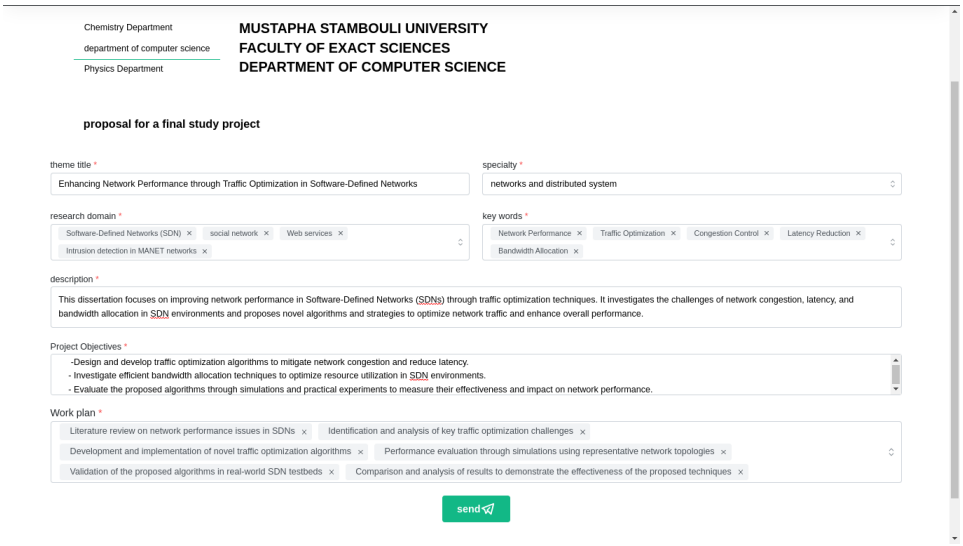


Figure 3.24 – *Example Of Suggest a Master Dissertation*

3.4.10 Ranking management

- Speciality manager has the option to view, delete and add students in ranking list, he can even upload a pre-made ranking list in 'xls' format

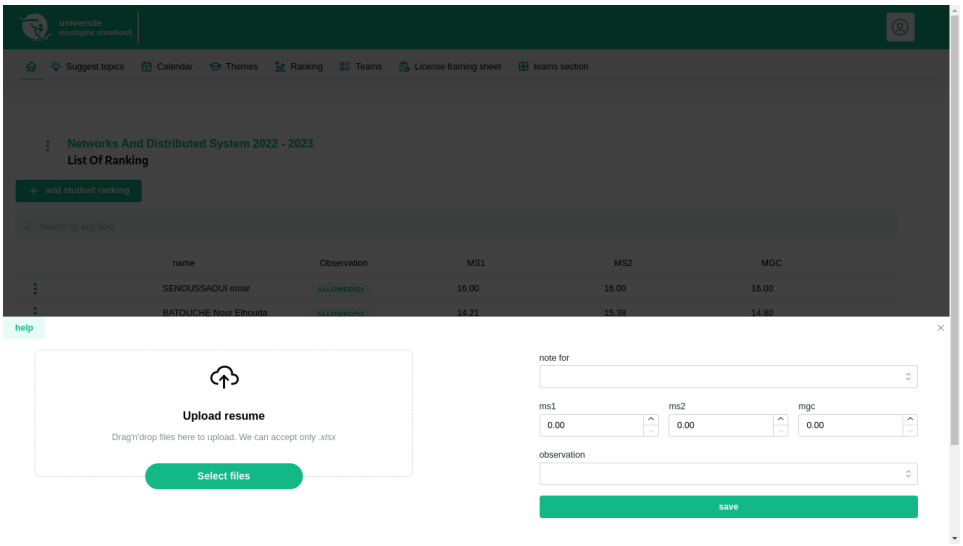


Figure 3.25 – *Add students Rank*

3.4.11 Suggested Themes Management

- The Speciality Manager has access to the complete list of suggested themes.
- He can publish the list of accepted themes.

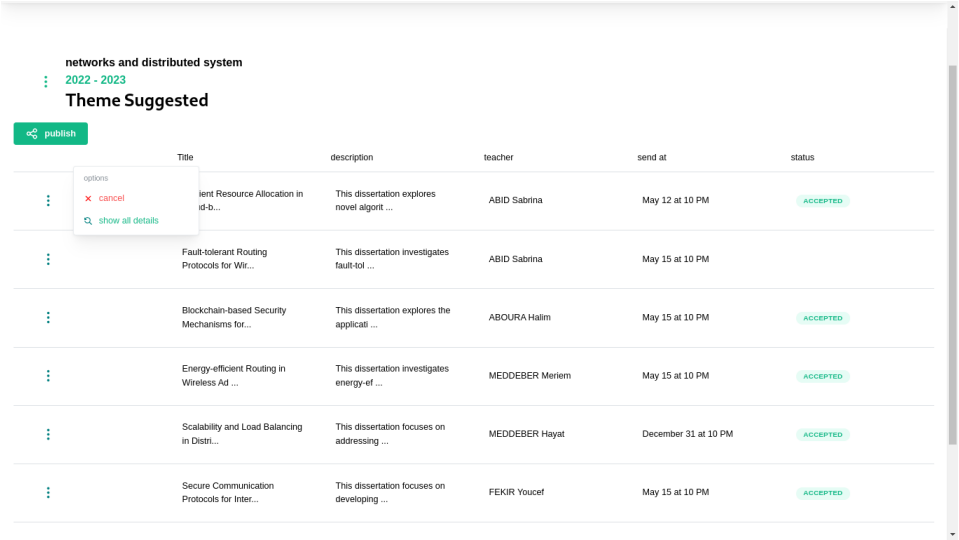


Figure 3.26 – *Suggested themes list*

3.4.12 Supervisors Management

- The Speciality Manager has access to the complete list of Supervisors.
- He has the ability to publish the list of Supervisors.

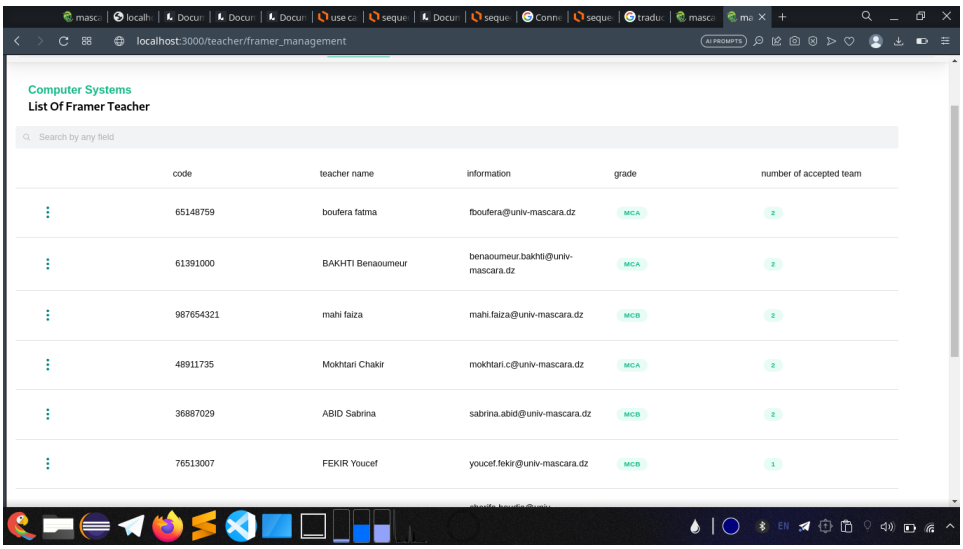


Figure 3.27 – *Supervisors list (speciality manager view)*

3.4.13 Team management

- The Speciality Manager can create team lists.

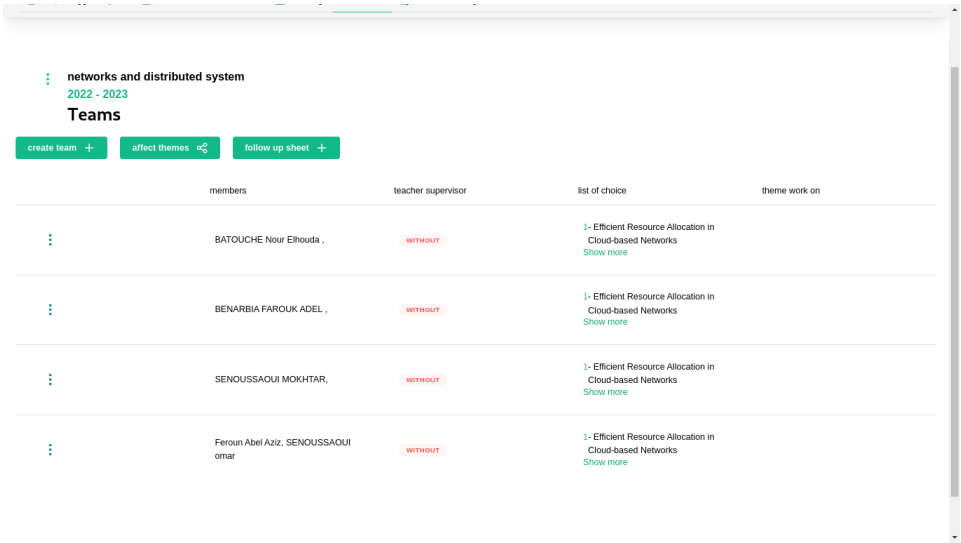


Figure 3.28 – *Teams list*

- The Speciality Manager can view state of progress of each team.

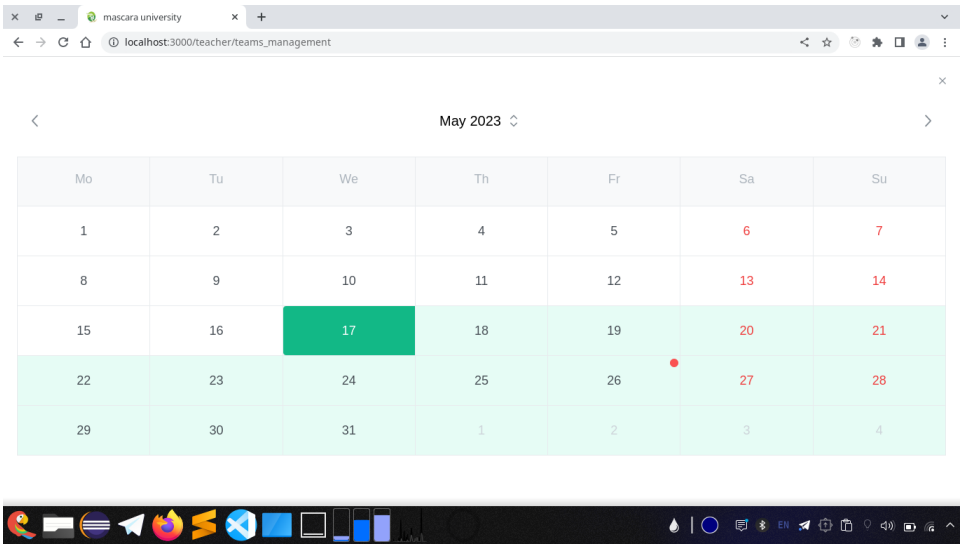


Figure 3.29 – *Effortless Scheduling: Seamlessly Manage Appointments by Specialty Managers*

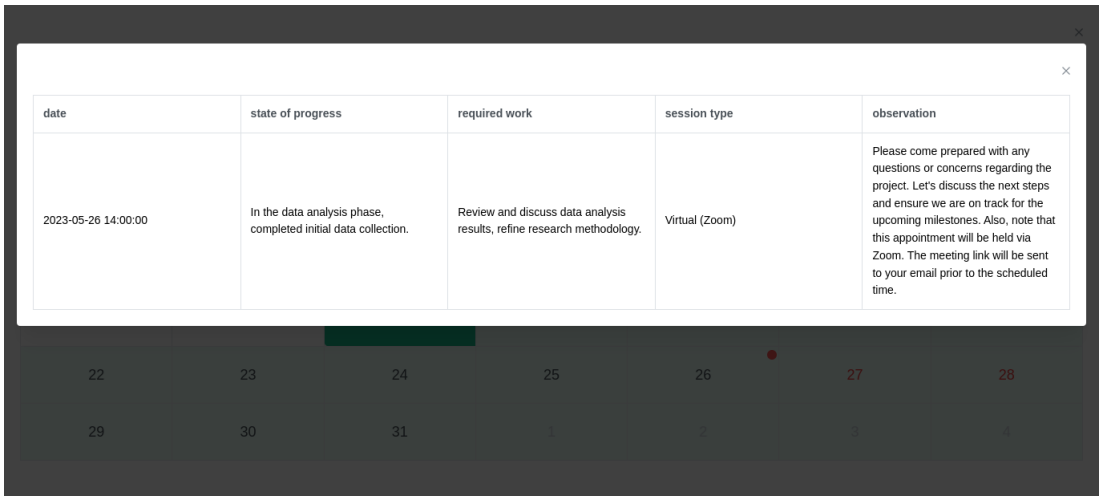


Figure 3.30 – *View of state of progress of a team*

- Speciality manager can create a jury group and schedule a presentation date

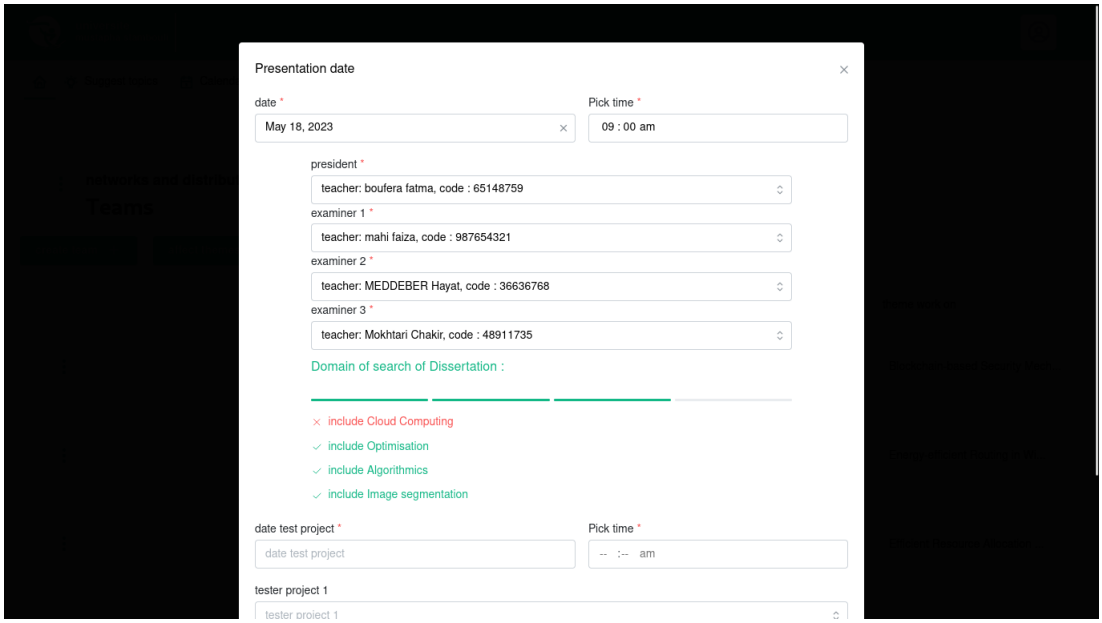


Figure 3.31 – *Create jury group for master*

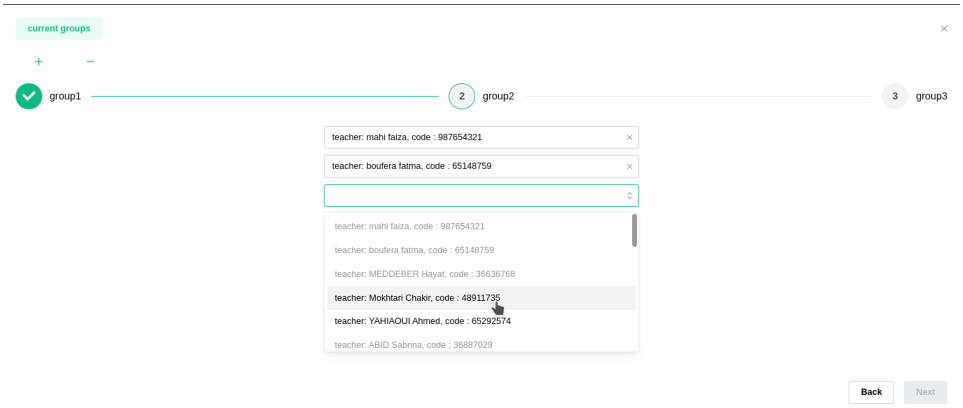


Figure 3.32 – *Create jury group for licence*

3.4.14 Calendar page

- The students and teachers can view the presentation date

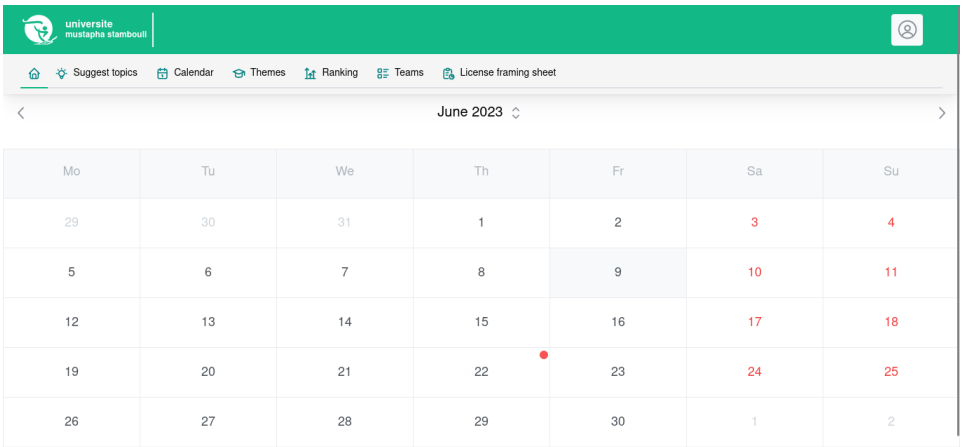
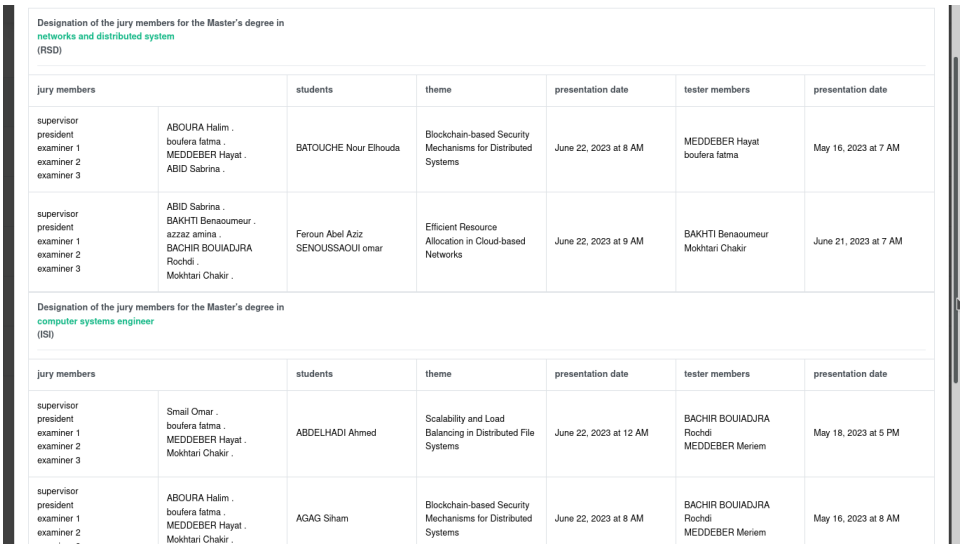


Figure 3.33 – *Calendar Page*



Designation of the jury members for the Master's degree in
networks and distributed system
(RSD)

jury members		students	theme	presentation date	tester members	presentation date
supervisor president examiner 1 examiner 2 examiner 3	ABOURA Halim . boufera fatma . MEDDEBER Hayat . ABID Sabrina .	BATOUCHE Nour Elhouda	Blockchain-based Security Mechanisms for Distributed Systems	June 22, 2023 at 8 AM	MEDDEBER Hayat boufera fatma	May 16, 2023 at 7 AM
supervisor president examiner 1 examiner 2 examiner 3	ABID Sabrina . BAKHTI Benaoumeur . azzaz amina . BACHIR BOUIADJRA Rochdi . Mokhtari Chakir .	Feroun Abel Aziz SENOUSSAOUI omar	Efficient Resource Allocation in Cloud-based Networks	June 22, 2023 at 9 AM	BAKHTI Benaoumeur Mokhtari Chakir	June 21, 2023 at 7 AM

Designation of the jury members for the Master's degree in
computer systems engineer
(ISI)

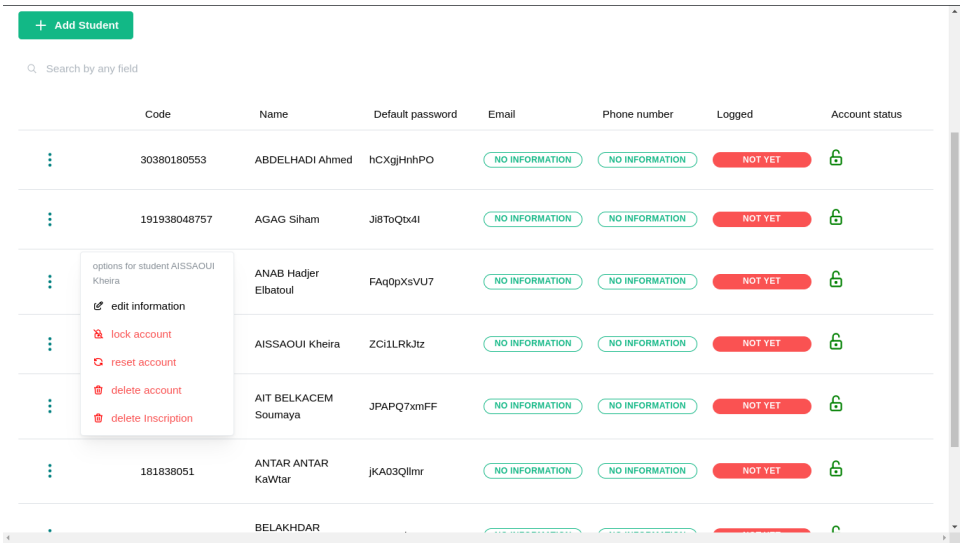
jury members		students	theme	presentation date	tester members	presentation date
supervisor president examiner 1 examiner 2 examiner 3	Smail Omar . boufera fatma . MEDDEBER Hayat . Mokhtari Chakir .	ABDELHADI Ahmed	Scalability and Load Balancing in Distributed File Systems	June 22, 2023 at 12 AM	BACHIR BOUIADJRA Rochdi MEDDEBER Meriem	May 18, 2023 at 5 PM
supervisor president examiner 1 examiner 2	ABOURA Halim . boufera fatma . MEDDEBER Hayat . Mokhtari Chakir .	AGAG Siham	Blockchain-based Security Mechanisms for Distributed Systems	June 22, 2023 at 8 AM	BACHIR BOUIADJRA Rochdi MEDDEBER Meriem	May 16, 2023 at 8 AM

Figure 3.34 – *Designation of the jury members*

3.4.15 Admin Page

Student Account management

- Admin has option to add, edit and delete students
- he can upload a pre-made students list in 'xls' format



+ Add Student

Search by any field

	Code	Name	Default password	Email	Phone number	Logged	Account status
	30380180553	ABDELHADI Ahmed	hCXgjHnhPO	NO INFORMATION	NO INFORMATION	NOT YET	
	191938048757	AGAG Siham	Ji8ToQtx4I	NO INFORMATION	NO INFORMATION	NOT YET	
		ANAB Hadjer Elbatoul	FAq0pXsVU7	NO INFORMATION	NO INFORMATION	NOT YET	
		AISSAOUI Kheira	ZCi1LRkJtz	NO INFORMATION	NO INFORMATION	NOT YET	
		AIT BELKACEM Soumaya	JPAPQ7xmFF	NO INFORMATION	NO INFORMATION	NOT YET	
	181838051	ANTAR ANTAR KaWtar	jKA03Qllmr	NO INFORMATION	NO INFORMATION	NOT YET	
		BELAKHDAR					

options for student AISSAOUI Kheira
- edit information
- lock account
- reset account
- delete account
- delete Inscription

Figure 3.35 – *Students Accounts*

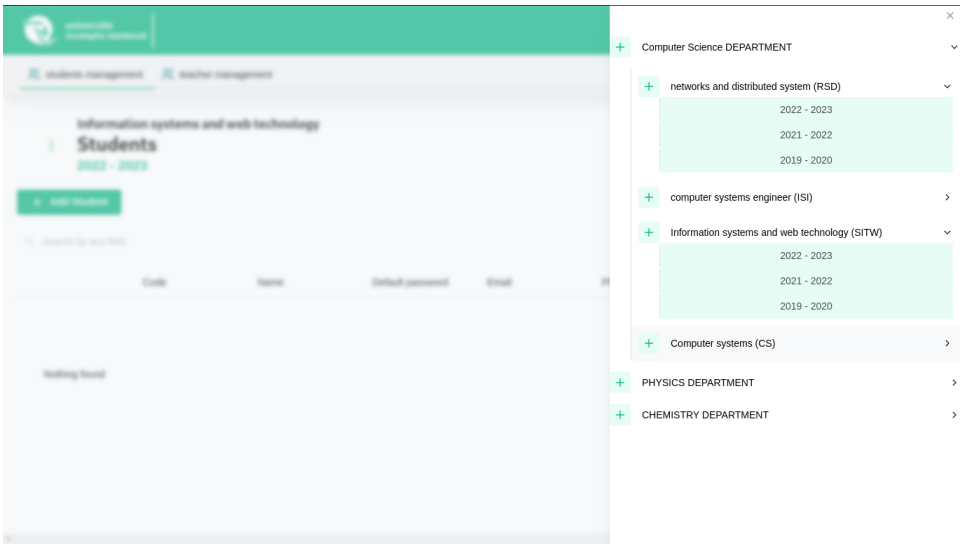


Figure 3.36 – *Organizing Students by Specialty and Years of Study*

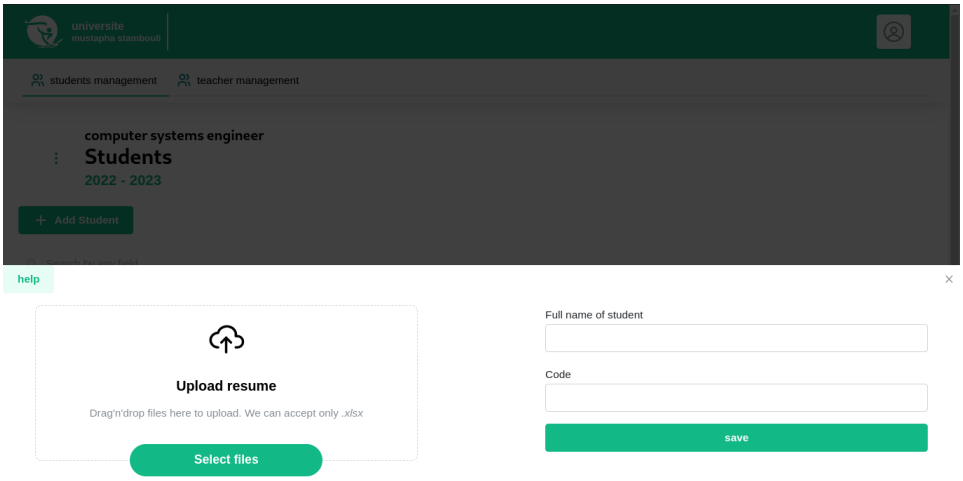


Figure 3.37 – *Add Student account*

Teacher Account management

- Admin has the option to add, edit and delete teachers
- Admin has ability to assign a Specialty Manager role to the teacher if needed.

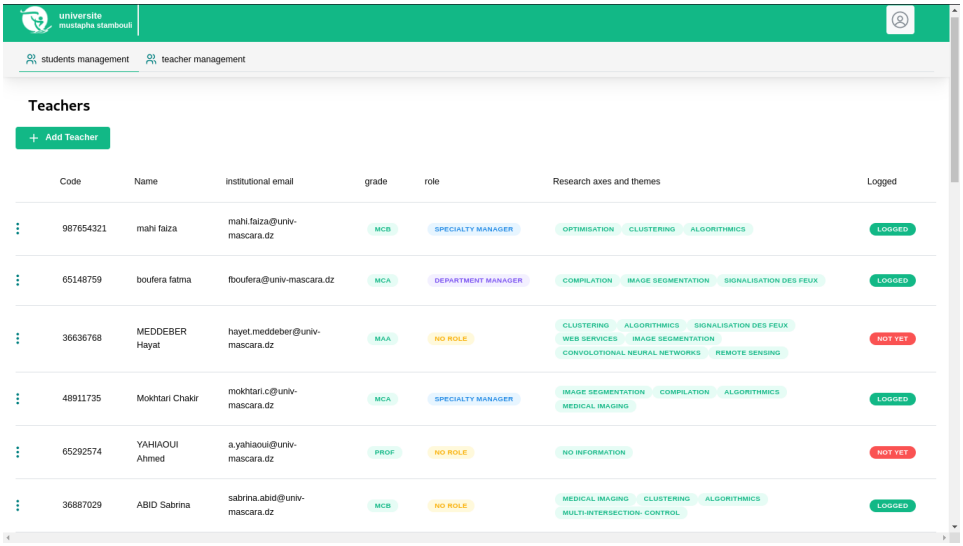


Figure 3.38 – *Teachers Accounts and their roles*

Conclusion

This chapter focused on the development of a specialized website that caters to the needs of specialty managers, teachers, and students. The purpose of the website is to facilitate effective communication, seamless team collaboration, and streamlined project management for the successful completion of final study projects. The implementation of this website involved transitioning from UML (Unified Modeling Language) to the actual implementation of the website. Modern tools such as React JS were used to create a user-friendly interface, while Laravel API handled requests efficiently. The website's administrator manages all user accounts, ensuring smooth operation and efficient user management. By encompassing these key actors and utilizing the transition from UML to implementation, the website provides a comprehensive platform that allows students to connect and collaborate in teams for their final study projects, while providing a means for supervisors to oversee and guide their progress.

General conclusion

This final study project allowed us to present the computer solution designed for the assignment of themes and planning viva presentation within the Faculty of Exact Sciences at Mustapha Stambouli University in Mascara.

Our work consisted of designing and implementing a web application to extend the support management. In this context, we aimed to develop a flexible and scalable application that could be further improved to anticipate the evolving needs of the users.

Firstly, this application improved communication, information exchange, and documentation between students, teachers and the university. It enables control, and monitoring of the training paths through the involvement of multiple stakeholders.

To design our application, we employed the UML methodology. This approach enabled us to thoroughly understand the problem and develop an interactive system to achieve the objectives set during the analysis of the existing system. As a result, we were able to create a stable and scalable system.

Furthermore, this project provided an opportunity to acquire expertise in web development.

Looking ahead, this work remains open to potential improvements, such as creating a customized web service for all the functions of our application so that they can be utilized by other applications.

Bibliography

Chinmayee Deshpande. The Best Guide to Know What Is React, Feb 7, 2023. URL https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs. Accessed on May 21, 2023.

mascara university. Presentation of faculty exact science mascara, n.d.a. URL https://www.univ-mascara.dz/fse/. Accessed on Feb 12, 2023.

mascara university. Presentation of university Mustapha Stambouli, n.d.b. URL https://www.univ-mascara.dz/?page_id=2233. Accessed on May 12, 2023.

Object Management Group (OMG). Unified Modeling Language (UML), 1997. URL https://www.omg.org/spec/UML/About-UML/. Accessed on May 6, 2023.

Object Management Group (OMG). Unified Modeling Language (UML) 2.5.1 Specification, 2017. URL https://www.omg.org/spec/UML/2.5.1/PDF. Accessed on May 6, 2023.

Oracle. *Oracle Database 11g Release 1 (11.1) Documentation*. Oracle Corporation, 2008. URL https://docs.oracle.com/cd/B28359_01/server.111/b28318/intro.htm#CNCPT998. Accessed on May 6, 2023.

Oracle. What is mysql?, n.d. URL https://www.oracle.com/mysql/what-is-mysql/. Accessed on May 20, 2023.

Visual Paradigm. What is a class diagram?, n.d.a. URL https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/. Accessed on May 6, 2023.

Visual Paradigm. What is a sequence diagram?, n.d.b. URL https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/. Accessed on May 6, 2023.

phpMyadmin. Bringing mysql to the web, n.d. URL https://www.phpmyadmin.net/. Accessed on May 20, 2023.

talend. What is mysql? everything you need to know, n.d. URL https://www.talend.com/resources/what-is-mysql/. Accessed on May 20, 2023.

Techopedia. How to create a site map: Your guide to visual site maps, November 22, 2017. URL https://www.gliffy.com/blog/how-to-create-a-sitemap. Accessed on May 6, 2023.

Wikipedia. phpmyadmin, 31 March 2023. URL https://en.wikipedia.org/wiki/PhpMyAdmin. Accessed on May 20, 2023.

Notations

IDE	Integrated Development Environment.
API	Application Programming Interface.
HTML	Hypertext Markup Language.
CSS	Cascading Style Sheets.
SQL	Structured Query Language.
MVC	Model-View-Controller.
CLI	Command Line Interface.
GUI	Graphical User Interface.
JSON	JavaScript Object Notation.
PHP	Hypertext Preprocessor.
HTTP	Hypertext Transfer Protocol

الملخص

تركز هذه الدراسة على إنشاء منصة متخصصة مصممة لتسهيل اختيار مواضيع مشاريع الدراسة النهائية وإدارة الفرق والمشرفين، وتتبع التقدم للطلاب والمعلمين. تهدف هذه المنصة إلى تبسيط عمليات اتخاذ القرار وتعزيز التعاون بين أصحاب المصلحة في المؤسسات التعليمية. من خلال استخدام هذه المنصة، يحصل الطلاب على وصول إلى نظام مركزي يساعدهم على اتخاذ قرارات مستنيرة بشأن مواضيع بحوثهم. تمكن المنصة أيضًا من التنسيق والتواصل الفعال بين الطلاب والمشرفين ومديري التخصصات، مما يسهم في إدارة مشروع فعالة. علاوة على ذلك، يسهل دمج نظام التقويم جدولة عروض المشروع، مما يعزز الشفافية والوصولية. يوفر تنفيذ هذه المنصة بيئة سلسة وفعالة للطلاب للتفوق في دراستهم.

Résumé

Cette étude porte sur la création d'une plateforme spécialisée conçue pour faciliter la sélection des thèmes de projet d'étude final, la gestion des équipes et des superviseurs, ainsi que le suivi des progrès des étudiants et des enseignants. Cette plateforme vise à rationaliser les processus de prise de décision et à améliorer la collaboration entre les parties prenantes des établissements d'enseignement. En utilisant cette plateforme, les étudiants ont accès à un système centralisé qui les aide à prendre des décisions éclairées concernant leurs sujets de recherche. La plateforme permet également une coordination et une communication efficaces entre les étudiants, les superviseurs et les responsables spécialisés, ce qui contribue à une gestion de projet efficace. De plus, l'intégration d'un système de calendrier simplifie la planification des présentations de projet, favorisant ainsi la transparence et l'accessibilité. La mise en œuvre de cette plateforme offre un environnement fluide et efficace aux étudiants pour exceller dans leurs études.

Abstract

This study focuses on the creation of a specialized platform designed to facilitate the selection of final study project themes, team and supervisor management, and progress tracking for students and teachers. The platform aims to streamline decision-making processes and enhance collaboration among stakeholders in educational institutions. By utilizing this platform, students gain access to a centralized system that assists them in making informed decisions regarding their research topics. The platform also enables effective coordination and communication between students, supervisors, and specialty managers, contributing to efficient project management. Furthermore, the integration of a calendar system simplifies the scheduling of project presentations, promoting transparency and accessibility. The implementation of this platform provides a seamless and efficient environment for students to excel in their academic pursuits.