

Course Management Support Application-iMaster.Process

Mohamed Wissem Guiza - a42706

Project work presented to the Escola Superior de Tecnologia e Gestão de Bragança to
obtain the Master's Degree in Information Systems under the dual diploma program
with the Université Libre de Tunis

Supervisors:

Professor José Eduardo Moreira Fernandes

Professor Paulo Alexandre Vara Alves

Professor Marwa MASSAABI

Bragança

November 2020

Course Management Support Application-iMaster.Process

Mohamed Wissem Guiza - a42706

Project work presented to the Escola Superior de Tecnologia e Gestão de Bragança to
obtain the Master's Degree in Information Systems under the dual diploma program
with the Université Libre de Tunis

Supervisors:

Professor José Eduardo Moreira Fernandes

Professor Paulo Alexandre Vara Alves

Professor Marwa MASSAABI

Bragança

November 2020

Dedication

The successful realization of this graduation project was made possible through the contribution of many people.

To express my gratitude, to say thank you to all of them is not even enough.

I want especially to thank my parents who always pushed me to the edge of success and believed in me during critical times.

To my sister who always was my source of inspiration and taught me that hard work always pays off.

To my friends who were my second family while living abroad, who always encouraged me throughout this experience.

Thank you from the bottom of my heart.

Acknowledgment

I wish to show my gratitude to everyone who helped me and supported me to achieve this stage and accomplish my final project.

I am deeply grateful to my supervisors, Prof. José Eduardo Fernandes and Prof. Paulo Alves for their professional guidance and generous help during the work project.

I want to thank the IPB family as well as ULT family for creating a comfortable work environment during the past years which helped me to enlarge knowledge and prepared me to start my professional career.

Finally, my deepest appreciation goes to my family who supported me everyday and encouraged me when I felt weak. This achievement would not be possible without their help. Thank you!

Abstract

This project is a part of my graduation to obtain a computer engineering degree and a Master degree in Information Systems.

It was realized within IPB and during this period I tried to develop a solution to facilitate the management of the masters courses in IPB.

The master process is complex and time consuming and thats how was the born of the iMasters Process idea.

iMasters Process aims to handle the submission, approval and assignment process in the masters section at IPB. It aims also to help monitoring the development of the work until the final submission and to handle the jury assignment process for the final defence.

In order to achieve that, i had to understand some new technologies and methodology that helped me to conceive and to develop an optimized solution.

Based on the literature review, ASP.NET Core and MySQL were chosen for the back-end development and React Js for the frontend development of the application.

Keywords: ASP.net Core, ReactJs, EntityFramework, SQL Server 2019 Developer, IPB Master management.

Resumo

Este projeto faz parte da minha graduação para a obtenção do curso de Engenharia da Computação e do Mestrado em Sistemas de Informação.

Foi realizado dentro do laboratório do IPB e durante este período procurei desenvolver uma solução que facilitasse a gestão de mestrado no IPB.

Para tal, tive que compreender algumas novas tecnologias e metodologias que me ajudaram a conceber e desenvolver uma solução otimizada

Contents

- 1 Introduction** **1**

- 2 State of the Art** **3**
 - 2.1 The master process in IPB 3
 - 2.2 The problem statement 4
 - 2.3 Microsoft SharePoint 5
 - 2.4 Workzone 6
 - 2.5 Proposed Solution 8

- 3 Requirement analysis and specification** **9**
 - 3.1 Functional specifications 9
 - 3.1.1 Supervisor 9
 - 3.1.2 Scientific committee member 10
 - 3.1.3 Student 10
 - 3.1.4 Master director 10
 - 3.2 Non-functional specifications 10
 - 3.2.1 Extensibility 10
 - 3.2.2 Scalability 11
 - 3.2.3 Ergonomic 11
 - 3.3 Use case diagrams 11
 - 3.3.1 Actors identification 11
 - 3.3.1.1 Supervisor 12

3.3.1.2	Scientific committee member	12
3.3.1.3	Master director	12
3.3.1.4	Student	12
3.3.2	General use case diagram	12
3.3.3	For Student	14
3.3.4	For Supervisor	14
3.3.5	For Scientific committee member	17
3.3.6	For Master Director	18
3.4	Activity diagrams	21
3.4.1	Proposal submission activity diagram	21
3.4.2	Work choice activity diagram	22
3.5	State machine diagrams	23
3.5.1	State machine diagram for work choice process	24
3.5.2	State machine diagram for proposal process	24
3.6	Site maps	25
3.6.1	Site map for Supervisor	25
3.6.2	Site map for Scientific committee member	26
3.6.3	Site map for student	27
3.6.4	Site map for master director	27
3.7	Database schema	28
3.8	Class diagram	30
4	Technologies and development tools	31
4.1	Front-end frameworks	31
4.1.1	AngularJS	32
4.1.2	Vue	32
4.1.3	React	33
4.2	Back-end frameworks	34
4.2.1	ASP.NET and ASP.NET MVC	36

4.2.2	Ruby on Rails	36
4.2.3	Django	36
4.2.4	Laravel	37
4.3	Development tools and technologies definition	37
4.3.1	Front-end Framework	37
4.3.2	Back-end Framework	38
4.3.3	Other technologies	38
4.3.3.1	Microservices	38
4.3.3.2	Entity framework	39
4.3.3.3	Astah	40
4.3.3.4	SQL Server Management Studio	40
4.3.3.5	ASP.NET Core Identity	41
4.3.4	MVC architecture	42
5	Development of iMaster Process platform	43
5.1	Results	43
5.1.1	Login window	43
5.1.2	Pages for Student	44
5.1.3	Pages for Scientific committee member	46
5.1.4	Pages for Master Director	49
5.1.5	Pages for Supervisor	52
6	Conclusion	57
	Appendix A Detailed use cases	61

List of Figures

2.1	Screenshot from SharePoint project management	5
2.2	Screenshot from Workzone dashboard	7
3.1	General use case	13
3.2	Use case for student	14
3.3	Use case for supervisor	15
3.4	Use case for scientific committee member	17
3.5	Use case for master director	19
3.6	Proposal submission activity diagram	22
3.7	Work choice activity diagram	23
3.8	State machine diagram for work choice process	24
3.9	State machine diagram for proposal process	25
3.10	Site map for supervisor	26
3.11	Site map for scientific committee member	26
3.12	Site map for student	27
3.13	Site map for master director	28
3.14	Database schema	29
3.15	Class diagram	30
4.1	Frameworks downloads in the past two years	31
4.2	Two-way binding architecture	32
4.3	Vue.js data-driven concept	33
4.4	One-way data binding architecture	34

4.5	Graph of the web framework popularity based on GitHub and Stack Overflow scores	35
4.6	Microservices architecture	39
4.7	Entity Framework architecture	39
4.8	ASP.NET Core Identity architecture	41
4.9	MVC architecture	42
5.1	Screenshot for the login window	44
5.2	Screenshot for consulting the available thesis list window	44
5.3	Screenshot for consulting the current thesis window	45
5.4	Screenshot for consulting the current task list window	45
5.5	Screenshot for the student list window	46
5.6	Screenshot for the jury list window	46
5.7	Screenshot for the proposals list window	47
5.8	Screenshot for the analysing proposal window	47
5.9	Screenshot for asking for reformulation window	48
5.10	Screenshot for the received mail window	48
5.11	Screenshot for the accepted thesis list window	49
5.12	Screenshot for consulting the thesis list window	49
5.13	Screenshot for fixing the presentation for a thesis window	50
5.14	Screenshot for consulting the final presentation schedule list window	50
5.15	Screenshot for assigning jury members for the final examination	51
5.16	Screenshot for the confirmed final examinations list	51
5.17	Screenshot for the invited jury members for a final examination	52
5.18	Screenshot for the proposal submission window	52
5.19	Screenshot for the proposals list per supervisor window	53
5.20	Screenshot for the proposals reformulation window	53
5.21	Screenshot for the proposals list of all the supervisors window	54
5.22	Screenshot for the available thesis list window	54

5.23 Screenshot for the thesis list in progress window	55
5.24 Screenshot for the adding task for a thesis window	55
5.25 Screenshot for monitoring work by supervisor window	56

Chapter 1

Introduction

Nowadays, technology, being the solution to several modern problems, is spreading in a rapid pace. It has impacted nearly every sector in the world and education is no exception. With online courses, and several other online tools, education has indeed stayed in line with the technology boom.

Educational management is an important subdivision of the education sector. It refers to the administrative processes of the educational system including human resources (students, professors and other employees) and material resources (finances, buildings, etc.). Educational management is growing more complex as the number of students increases. And this is where technology comes in handy. That is how this project came to life.

iMaster Process is a solution to support the master course's management of the Instituto Politécnico de Bragança. It digitizes and facilitates processes for the IPB master management. It includes most of the frequent administrative services.

iMaster Process also draws the interactions between all the necessary actors for the masters processes. This includes students, supervisors, administration staff and master director. The web-based tool further generates statistics reports depending on the university's needs. This can serve to control the process flow or as feedback to improve the process later on.

This project is hosted by the Escola Superior de Tecnologia e Gestão de Bragança situated in Bragança, Portugal.

The overall project is divided into three parts: iMaster Process, iMaster Statistics and iMaster Schedule. This work focuses on the iMaster Process which have the following objectives:

- Handle the submission, approval and assignment process.
- Monitor the development of a work until the final submission.
- Deal with the jury exam process

The remainder of the document will be divided as follows :

Chapter 2 is about the state of the art, a discussion about the master process in IPB which takes us to stating the problem, and a summary of the similar softwares to the application. The chapter ends by proposing the solution to solve the main problem.

Chapter 3 identifies the different actors in our application and it specifies the functional and non-functional requirements. The chapter presents also the use case diagrams for the different actors, as well as, the state machine diagrams and activity diagrams. Chapter 3 ends by detailing the database schema and the class diagram of the application.

Chapter 4 a discussion about the different available technologies that represent viable options for the project, and a summary of the chosen tools and technologies as back-end and front-end frameworks.

chapter 5 presents the implementation and it contains also the results which are presented as screenshots from our application.

The document ends with a conclusion chapter.

Chapter 2

State of the Art

This chapter contains the principal master course process in Instituto Politécnico de Bragança, it states as well the main problem and the study of some available options. The chapter ends by presenting the proposed solution.

2.1 The master process in IPB

Every professor submits a new project proposal which will be analysed by the scientific committee. The objectives of the proposed work must be clearly defined by the professor. Once analysed and validated, the scientific committee will publish the proposal.

Otherwise the professor will be asked to do some reformulation on the proposal.

When the proposal is published, students may request the project for the dissertation. Once the work is assigned to student, the supervisor have to monitor the development of the work until the final submission.

The student must submit the work in a specific period of time, once it's submitted, the academic services must inform the master director, who have to collect the submitted files of the student and starts to define the jury members and proposes the list to the scientific committee members who will decide whether the jury list is approved or not. Once the jury list is approved, the president of the scientific committee must inform the jury members, supervisors and student and prepare for a preliminary meeting with

the jury members in order to decide whether the submitted work can be proceeded to examination or not.

If the submitted work is not accepted, the student has to choose between reformulating the work or not. In both cases, the student must inform the scientific committee director. Once the work is validated to proceeded to examination, the president of the scientific committee have to define and inform about the final defence date and place and prepare the formal documents.

After that, the final defence will have place and the jury members will assign the final mark and decide whether the report needs adjustments or not.

If the report is clear, the president of the scientific committee have to submit the work to IPB system.

2.2 The problem statement

Master students have to go through the same tedious administrative process every year. This process consists of several interactions between several involved parties including: students, professors, jury member and administration staff.

And after taking a closer look at the master process, the only thing that one can say is that the process needs to be virtualized into one common web application to make life easier for everyone.

The master process is tedious, complex and time consuming especially with the high number of interactions between the multiple actors.

iMaster Process will try to digitize most of the main process in order to make life easier in the masters section.

After a deep research, some similar softwares concerning project management were found. The next sections present the similar softwares analysis.

2.3 Microsoft SharePoint

Microsoft SharePoint [14] is a web-based platform to share informations and files. It is dedicated especially for companies regarding the collaboration features that it provides like project management, sharing documents and messaging.

Microsoft SharePoint is commonly used to create a private platforms for companies which can be accessed only by the authorized persons which make the platform secured.

In term of education [10] , Microsoft SharePoint is used in universities like University of Washington and schools like Lake Washington School District to share informations and documents between teachers, students and administration members. It complements both teaching and learning

The screenshot shows the 'Projects' dashboard in SharePoint. At the top, there is a search bar with the text 'Choose...' and 'Type your search here'. Below the search bar, there are navigation tabs: 'CURRENT PROJECTS', 'ALL PROJECTS', and 'PROJECT DASHBOARD'. The 'PROJECT DASHBOARD' tab is selected. Below the tabs, there is a table of projects. The table has the following columns: Title, Assigned To, Status, Account, Costs Status, Hour Status, and Overdue Tasks. The data in the table is as follows:

Title	Assigned To	Status	Account	Costs Status	Hour Status	Overdue Tas
BPA CRM implementation	Discovery 2013	Not Started	BPA Solutions SA	✓	✓	⚠
BPA Quality implementation	Trial 2013	In Progress	Aero Space Ltd	✓	✓	✖
Harmonic Project	Trial 2013	In Progress	Dream Company	✓	✓	⚠
Plant Expansion	Discovery 2013	Completed	Toledo Work Group	✓	⚠	⚠
New web site	Thorsten Kühn	In Progress	BPA Solutions SA	✓	✓	✖
Proposal Response	Discovery 2013	Not Started	EFECON AG	✓	✓	✓

Figure 2.1: Screenshot from SharePoint project management

[13]

Advantages of Microsoft SharePoint : [3]

- Integration with Microsoft Office.
- Highly customizable.
- Central storage.

Disatvantages of Microsoft SharePoint : [3]

- Complicated maintenance and setup.
- Expensive license.
- Requires employee training.

2.4 Workzone

Workzone [1] is a project-management solution based on cloud and focuses mainly on collaboration and document management. It offers users the ability to manage projects with multiple tasks, share files and collaborate with other users.

Workzone provides some interesting features for document management such as including permission for sharing documents, email notification alerting for users and automation of approval workflow by including emailing approval requests.

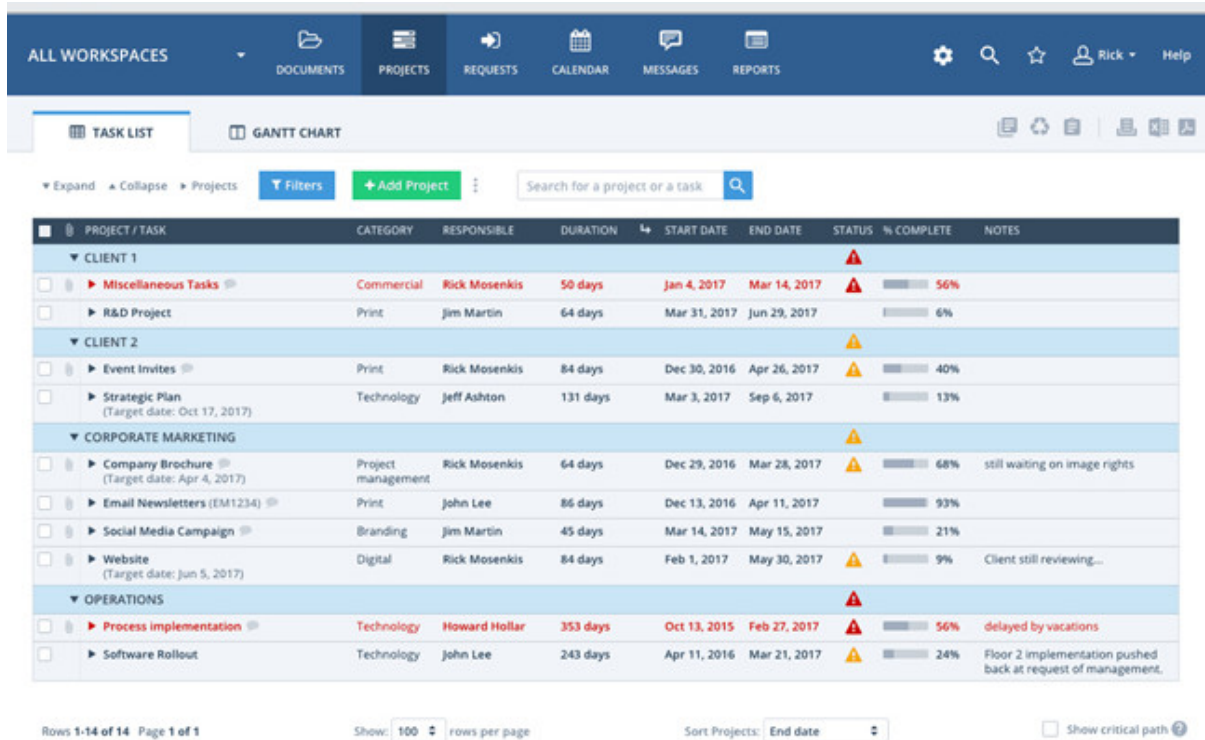


Figure 2.2: Screenshot from Workzone dashboard

[11]

Advantages of Workzone : [12]

- Solid project management.
- Secure platform.
- Easy to use.

Disadvantages of Workzone : [12]

- Archiving issues.
- Workload tracking issues.

2.5 Proposed Solution

The previous solutions are a project management solutions which can be used in universities since it helps to enhance distance learning, sharing informations and documents between different users based on authorization system.

Meanwhile, including all these great features, the previous solutions are limited regarding the masters course process in universities.

The proposed solution is iMaster Process which consists in conceiving and developing a platform to digitize and facilitate the master process in IPB. It consists of a Web application that encloses many services for the previously mentioned actors.

Chapter 3

Requirement analysis and specification

This chapter presents all the functional and non-functional requirements needed to develop the application. It presents as well the use case diagrams, activity diagrams and state machine diagrams. The chapter ends by detailing the site maps for the different users, class diagram and the database schema.

3.1 Functional specifications

Since the application is including several users with different requirements, this chapter specifies the requirements specification per user.

3.1.1 Supervisor

The application must allow the Supervisor to submit a new proposal to be analysed later by the Scientific committee members. The supervisor may check all the previous proposals submitted by other supervisors, as well as, his own proposals (reformulate, delete). The application must provide to the supervisor the possibility to consult his available thesis list which was not assigned to students yet and his thesis list which was already assigned

to students. In the last case, the application must allow the supervisor to set the goals for each assigned thesis to evaluate the work with the student.

3.1.2 Scientific committee member

The application must allow the scientific committee member to consult all the proposals list in order to analyse them. Once analysed, the scientific committee member may accept the proposal or ask for reformulations. The application must provide to the scientific committee member the possibility to consult the students list in order to assign the work.

3.1.3 Student

The application must allow the student to consult all the thesis list and to consult his current thesis and tasks set by his supervisors if he was assigned the work already.

3.1.4 Master director

The application must allow the master director to manage the final presentation for each thesis by fixing the date and the place as well as the jury members. The application must provide to the master director the possibility to add new jury members.

3.2 Non-functional specifications

Despite the fact that the non-functional specifications does not present the functionalities offered by the system, they have a direct impact on the application performance.

3.2.1 Extensibility

Extensibility is the ability to extend in a web application and it can be in form of adding new functionalities or in form of updating existing functionalities.

It is possible to add more functionalities in order to extend our application without having to adapt a lot of code thanks to the chosen design method for the proposed software.

3.2.2 Scalability

Scalability in term of web applications is all about growth. A successful web application is an application who can handle the increasing of number of user without interrupting the available users.

One of the most commonly used tools to assure the scalability in web applications is Microservices which is used in our application.

Our application is designed with scalability in mind. As a result will be able to handle an increase in users without disrupting the end users.

3.2.3 Ergonomic

The graphic interface must be clear and easy to use. We particularly paid attention to details such as the simple access to the different available functionalities , the visibility of images captured, the simplicity to navigate through the different application's windows.

3.3 Use case diagrams

Use cases are really important part while designing a web application. It helps to clarify the different actions for the different actors of the application.

In the following parts, we will present our general use case, as well as, the detailed use case diagrams of the application.

The main use cases will be explained in tables with scenarios and exceptions. The other use cases will be explained with more details in the appendix A.

3.3.1 Actors identification

This part presents the principle actors who will be interacting with each other in order to produce the master process in a good way.

3.3.1.1 Supervisor

Every undergraduate or postgraduate student working on a dissertation needs at least one supervisor from a correspondent department. Generally, the supervisor can be a professor or master director. The supervisor is responsible of guiding the student throughout the work development of the dissertation.

3.3.1.2 Scientific committee member

A Scientific Committee is composed of three members with doctorates or specialists in the field, one of these members will be assigned as president of the committee.

3.3.1.3 Master director

According to the IPB regulations, the master director is a professor of the master course.

3.3.1.4 Student

According to IPB's regulation, to be enrolled in dissertation/project/internship unit, the student must be holder of a bachelor degree or other equivalent degree.

3.3.2 General use case diagram

The figure 3.1 presents the general use case diagram. All the different actors of the system are present in this diagram (supervisor, master director, student, scientific committee member).

The following parts will explain in details every use case per actor.

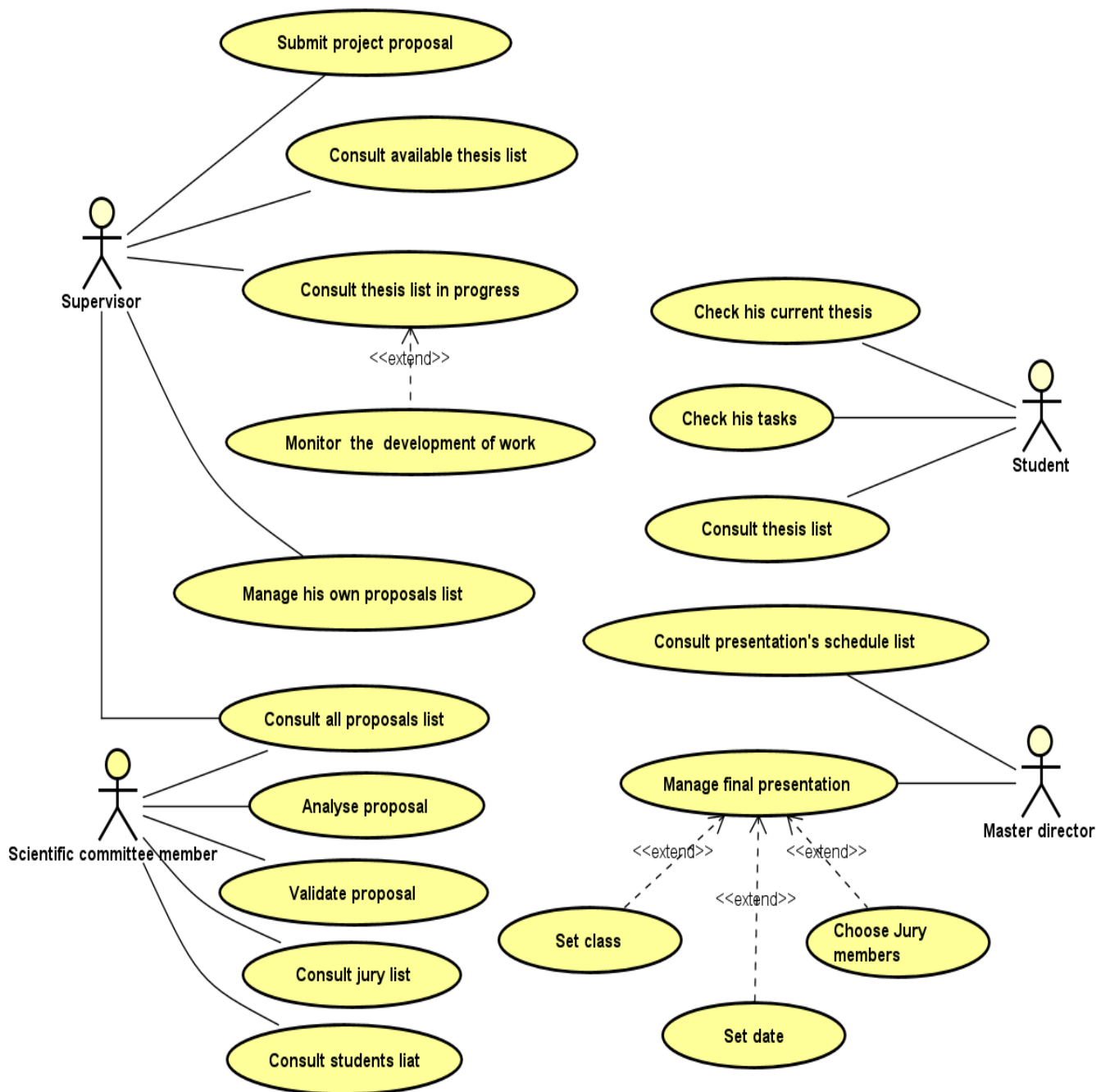


Figure 3.1: General use case

3.3.3 For Student

Figure 3.2 presents the use case for the students. Every student have the right to check his current thesis as well as to check all thesis list. Next part explains the main use case for the student.

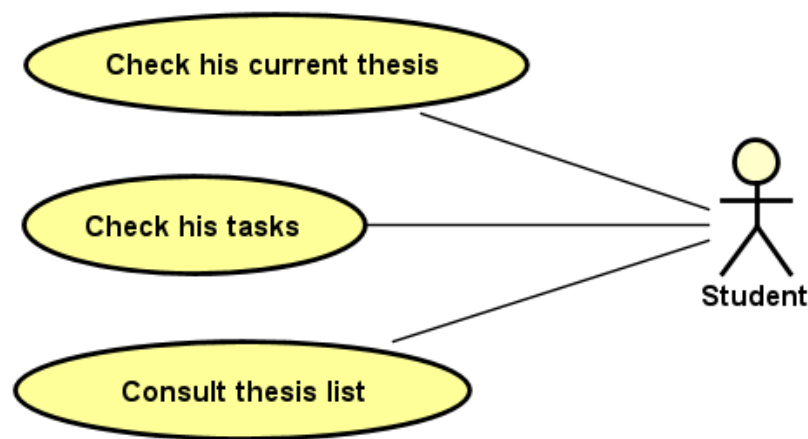


Figure 3.2: Use case for student

3.3.4 For Supervisor

Figure 3.3 presents the use case for the supervisor. Every supervisor have the right to :

- Submit a new project proposal
- Consult all proposals list
- Consult his own proposal list
- Monitor the work development with the students
- Consult the work development
- Consult the thesis lists in progress
- Consult the available thesis list (Not assigned yet)

Table 3.1: Main use case explanation for student

Use Case	Check the current thesis
Actors	Student
Goal	This use case allows the student to check his current thesis
Scenario	<ol style="list-style-type: none"> 1. The student clicks on "My current thesis" in the side menu bar. 2. After consulting his current thesis, the student can check his to-do tasks for the correspondent thesis which was added by his supervisor.
Exception:	<ol style="list-style-type: none"> 1. If the student was not assigned yet any work, the list will be empty 2. If the supervisor did not submit any new tasks, the tasks list will be empty.

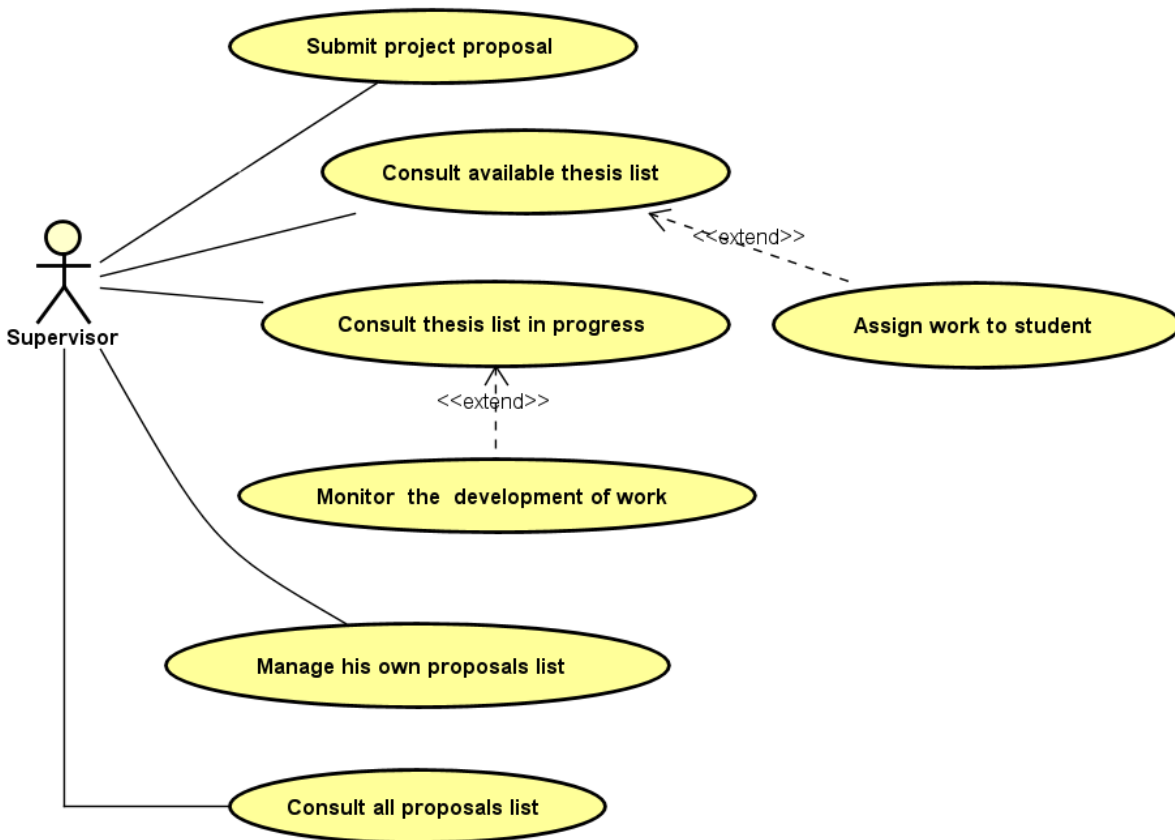


Figure 3.3: Use case for supervisor

Table 3.2: Consult available thesis use case

Use Case	Consult available thesis list
Actors	Supervisor
Goal	This use case allows the supervisor to consult his available thesis list
Scenario	<ol style="list-style-type: none"> 1. The Supervisor clicks on "My thesis" in the side menu bar. 2. After consulting his current thesis, the supervisor can assign the available thesis to student by clicking on "Assign work" he will be redirected to another page to choose one of the students.
Exception:	<ol style="list-style-type: none"> 1. The available thesis list is empty until the scientific committee approve one of the supervisors proposals. 2. If the thesis is already assigned to one of the students, it will not be on the available list anymore.

Table 3.3: Consult thesis list in progress use case

Use Case	Consult thesis list in progress
Actors	Supervisor
Goal	This use case allows the supervisor to consult his in progress thesis list and monitor the development of the work
Scenario	<ol style="list-style-type: none"> 1. The Supervisor clicks on "My thesis list" in the side menu bar to consult his in progress thesis list. 2. The supervisor can add new task for every thesis by clicking on "Add Task" and he can also check the previous task for every thesis by clicking on "Check Tasks" where he will have the option to edit the task by clicking on "Edit Task".

3.3.5 For Scientific committee member

Figure 3.4 presents the use case for the scientific committee member. Every scientific committee member have the right to :

- Validate the proposal
- Consult all proposals list
- Analyse proposals submitted by supervisors

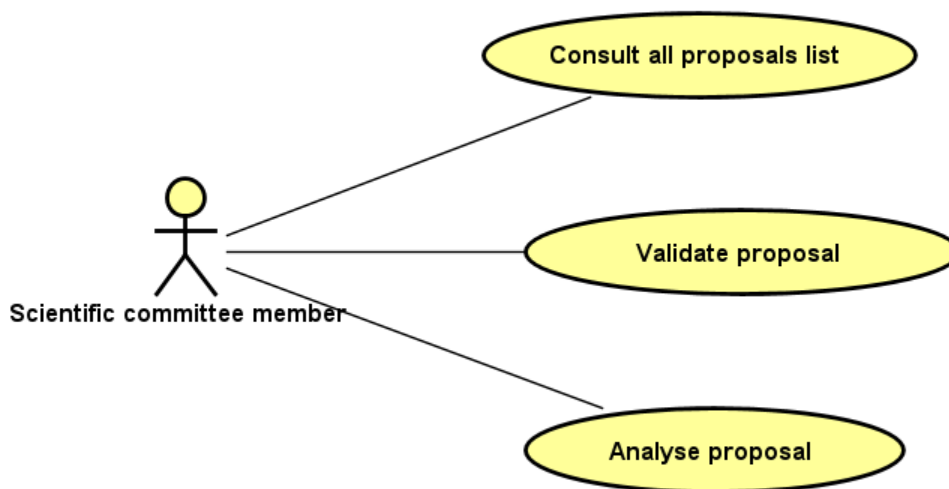


Figure 3.4: Use case for scientific committee member

Table 3.4: Consult all proposals list use case

Use Case	Consult all proposals list use
Actors	Scientific committee member
Goal	This use case allows the scientific committee member to consult all the submitted proposals
Scenario	<ol style="list-style-type: none"> 1. The scientific committee member clicks on "All proposal list" in the side menu bar. A list of the submitted proposals by the supervisors will be displayed to be analysed in further notice.

Table 3.5: Validate proposal use case

Use Case	Validate proposal
Actors	Scientific committee member
Goal	This use case allows the scientific committee member to validate a proposal or ask for a reformulation
Scenario	<ol style="list-style-type: none"> 1. The scientific committee member will receive a proposal list. After analysis, the scientific committee member can accept a proposal to be published or can ask for reformulation. 2. If the scientific committee member validate the submitted proposal, it will be added to the available thesis list 3. If the scientific committee member asks for reformulation, a mail will be sent to the correspondent supervisor to inform him. The reformulation reason should be written by the scientific committee member before sending the mail.
Exception:	<ol style="list-style-type: none"> 1. The proposals list is empty until a supervisor submits a new proposal

3.3.6 For Master Director

Figure 3.5 presents the use case for the master director. Every master director have the right to :

- Manage the final presentation by setting the date, class and the jury members.
- Consult the presentation's schedule.

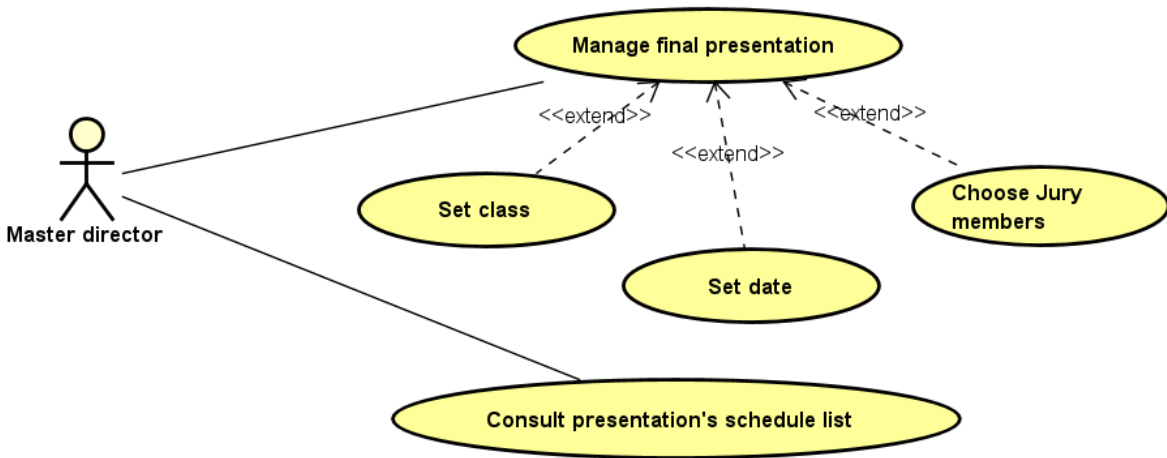


Figure 3.5: Use case for master director

Table 3.6: Consult presentations schedule list use case

Use Case	Consult presentations schedule list
Actors	Master director
Goal	This use case allows the master director to consult the final examination schedule list
Scenario	<ol style="list-style-type: none"> 1. The master director clicks on the "Final Examination Schedule", a list of the confirmed presentations will be displayed 2. The master director can consult the jury members for every examination by clicking on "Check invited jury list"
Exception:	<ol style="list-style-type: none"> 1. The final examination list is empty until the date, place and jury members are assigned to the correspondent thesis.

Table 3.7: Manage final presentation use case

Use Case	Manage final presentation
Actors	Master director
Goal	This use case allows the master director to manage the final presentation
Scenario	<ol style="list-style-type: none"> 1. The Supervisor clicks on "Thesis List" in the side menu bar. A list of all the thesis list will be displayed 2. The master director can press on "Fix Presentation" button in front of every thesis. He will be redirected to another page to fix the place and the date of the final presentation to get a final presentation list. 3. The master director can click on "Final presentation" button to get the list of the defined final presentation with the place and the correspondent date. The supervisor will have the "Add Jury" button. Once clicked, the supervisor will be redirected to another page where he can select jury list for the final presentation.
Exception:	<ol style="list-style-type: none"> 1. The thesis list is empty until the scientific committee accept at least one proposal.

3.4 Activity diagrams

To more understand the applications functionalities, the next parts will present the main activity diagrams of the application.

Activity diagram is a behavior diagram which represents the work flow of our application. It gives us a clear vision about how our applications activities are coordinated to provide the services.

The activity diagrams helps to describe the steps defined in the use cases which helps also to more understand the complicated use cases and to illustrate the business logic and the workflow between the system and the users.

This section will present the main activity diagrams for the work choice and the proposal submission and it will explain the workflow of each activity by describing the diagrams in a coherent paragraph.

3.4.1 Proposal submission activity diagram

Figure 3.6 presents the proposal activity diagram which is accomplished by the interaction between the supervisor and the scientific commission and next part will explain this process with more details.

The diagram is divided into two parts, one part is for the supervisor and the second is for the scientific commission.

The activity of the proposal submission starts when the supervisor submit a proposal.

After submission, the proposal will be analyzed by the scientific commission who have the right to validate or to not validate the proposal.

If the proposal is validated, the scientific commission will publish the proposal and the activity will come to and end.

Otherwise, the supervisor can choose whether to reformulate the proposal or not. If the last decide to reformulate, the proposal activity will start again from the beginning and if not, the proposal activity will end.

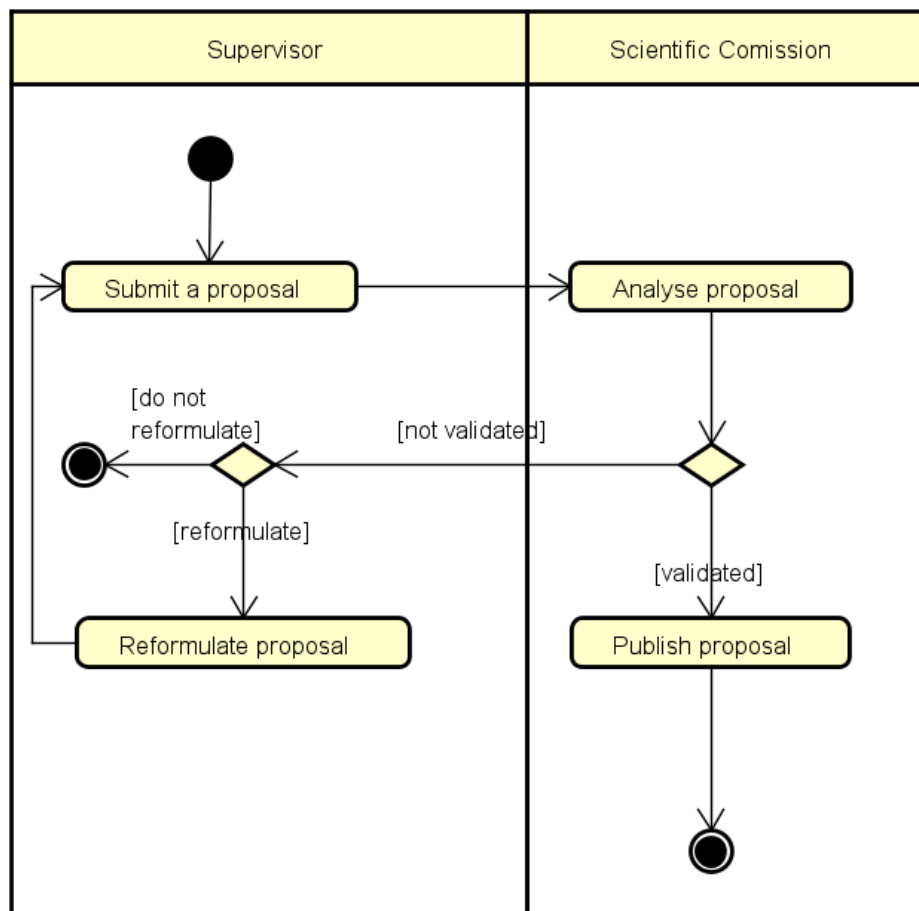


Figure 3.6: Proposal submission activity diagram

3.4.2 Work choice activity diagram

This activity is accomplished by the interaction between the student and the supervisor. The diagram is divided into two parts, one part is for the student and the second is for the supervisor.

The activity of the work choice starts when the student start consulting for the available proposals until he finds a proposal that match his ambitions.

The supervisor has the authority to make a decision whether to assign the work to the student or not.

Once assigned, the work choice activity is finished. Otherwise, the student will start

looking again for a new proposal and the process will restart again.

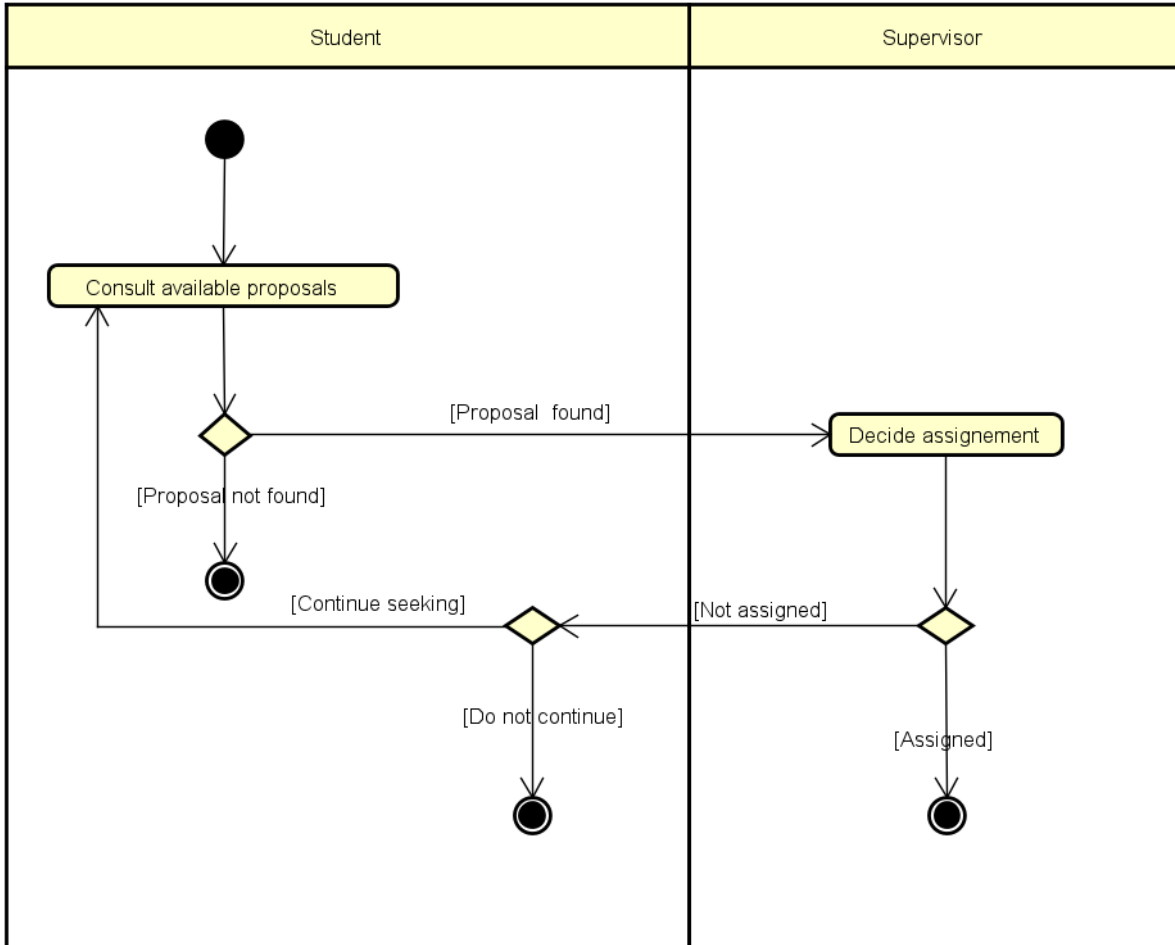


Figure 3.7: Work choice activity diagram

3.5 State machine diagrams

State machine diagram is presented as a behavior diagram using state transitions to express the usage protocol of part of a system. The whole master process is divided into two parts:

- Proposal process
- Work choice process

The next section will present all the parts using state machine diagrams.

3.5.1 State machine diagram for work choice process

Once the work choice process starts, the proposals are consulted, if the correct proposal was found, the state will change to requested and after that it depends on the supervisor's decision.

If the work is assigned to the student the state will change to be In progress and the work will start otherwise it will change to Withdrawn.

But if no proposal was found, the state will change directly to withdrawn.

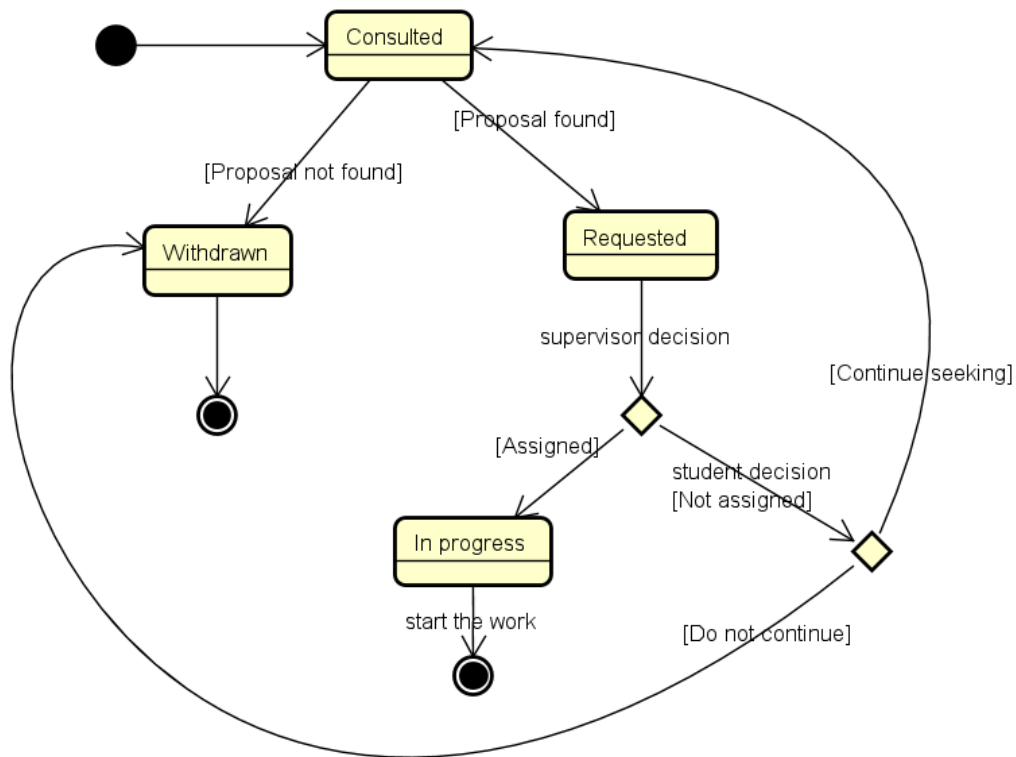


Figure 3.8: State machine diagram for work choice process

3.5.2 State machine diagram for proposal process

In term of state, the proposal process starts when the state is submitted. After that, the state change to be in analysis and depending of the analysis result, the proposal can take three different states. The first state is to be validated which results for the proposal to

be published and available. The second possible state is For Reformulation and the last possible state is Not Accepted which both take the proposal to be Archived. Once validated, the proposal is available and is waiting to be assigned to student, which give us the last state which is Assigned.

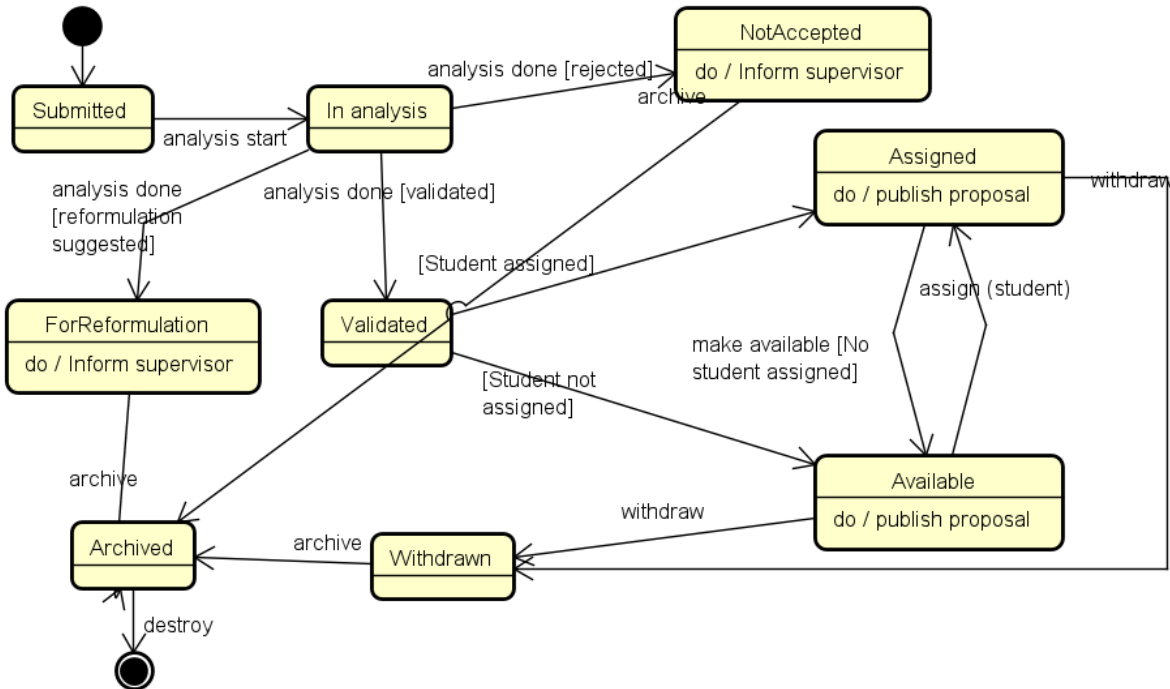


Figure 3.9: State machine diagram for proposal process

3.6 Site maps

In order to provide an easy tracking process, the following sections will develop a site maps for every actor contributing in the application.

3.6.1 Site map for Supervisor

The described diagram in Figure 3.10 presents the different windows for the supervisor in our application.

After a successful login, the supervisor will be logged in into his private area where he will find a side bar menu with all the functionalities that he can do as a supervisor.

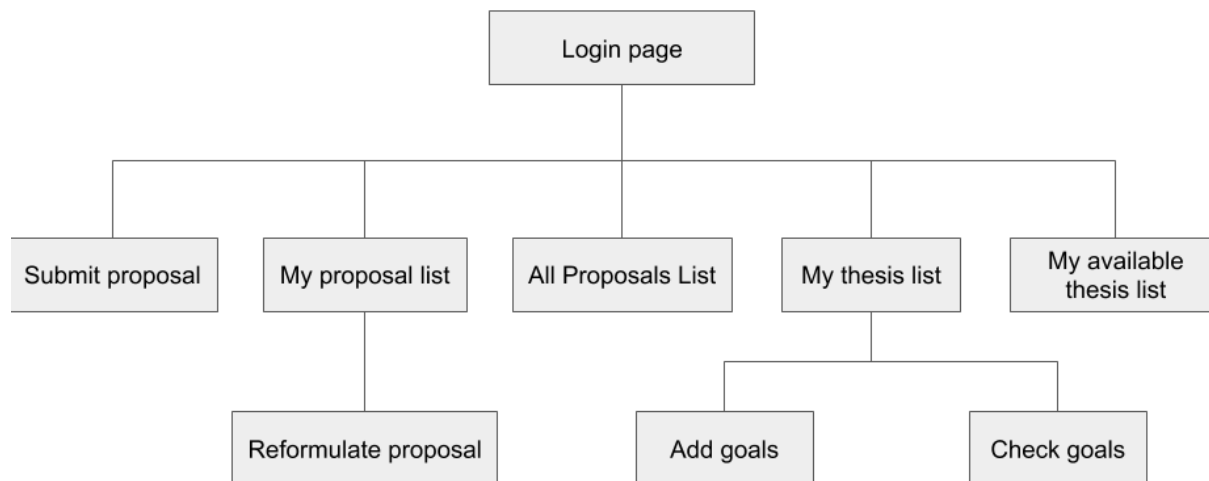


Figure 3.10: Site map for supervisor

3.6.2 Site map for Scientific committee member

The described diagram in Figure 3.11 presents the different windows for the scientific committee member in our application.

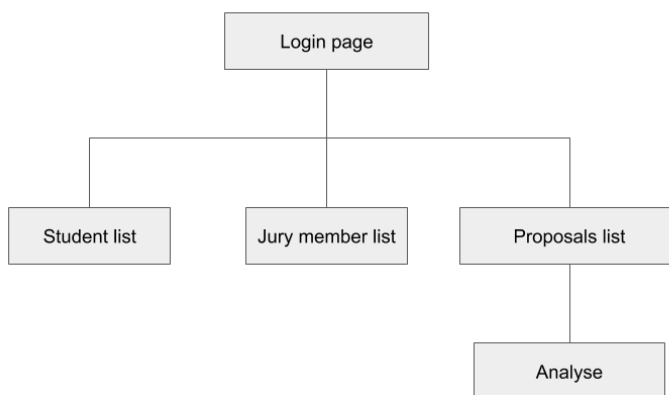


Figure 3.11: Site map for scientific committee member

Once logged in, the scientific committee member will be redirected to his private area where he will find a side bar menu to access to all his defined functionalities in the

application.

3.6.3 Site map for student

The described diagram in Figure 3.12 presents the different windows for the students in our application.

As all the actors, the student will be redirect to his private area after a successful login, he will get a side bar menu with the associated functionalities for his role.

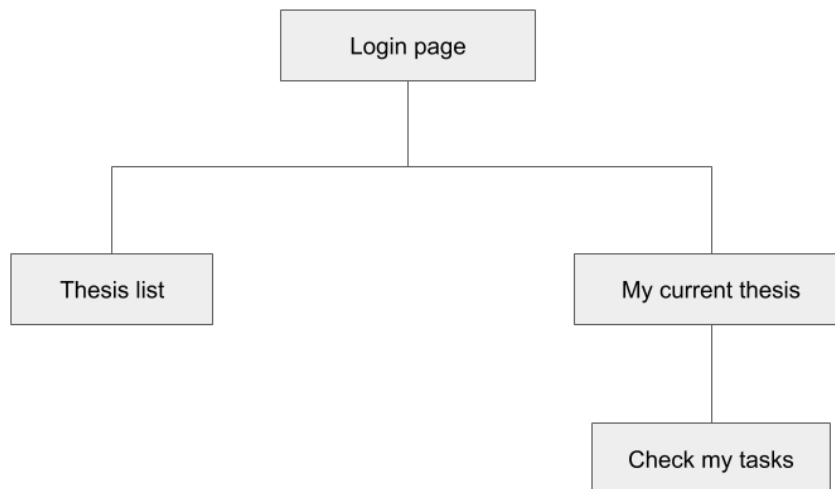


Figure 3.12: Site map for student

3.6.4 Site map for master director

The described diagram in Figure 3.13 presents the different windows for the master director in our application.

Once logged in successfully, the director of the master will be redirected to his private area where he will get a side bar menu filled with his associated functionalities implemented in our application.

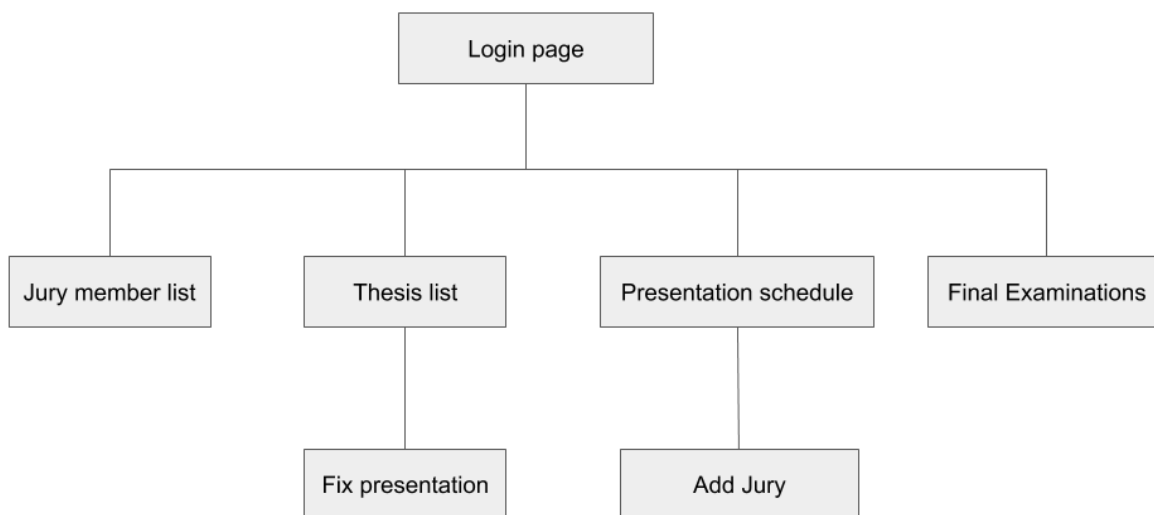


Figure 3.13: Site map for master director

3.7 Database schema

The figure 3.14 presents the database schema.

- AspNetUsers : Represents the user.
- AspNetRoles : Represents a role associated to user.
- AspNetUserClaims : Represents a claim for a specific user.
- AspNetUserTokens : Represents an authentication token for a user.
- AspNetUserLogins : Represents an association between user and login.
- AspNetRoleClaims : Represents a claim which is granted to all the users within a role.
- AspNetUserRoles : An entity that join users and roles.
- Proposal : Represents the submitted proposals.
- Area : Represents the different institute areas.
- Jury : Represents the jury members.
- Thesis : Represents the different thesis.
- Presentation : Represents the final examination.

- Tasks : Represents the goals set between the supervisor and student to monitor the work development.

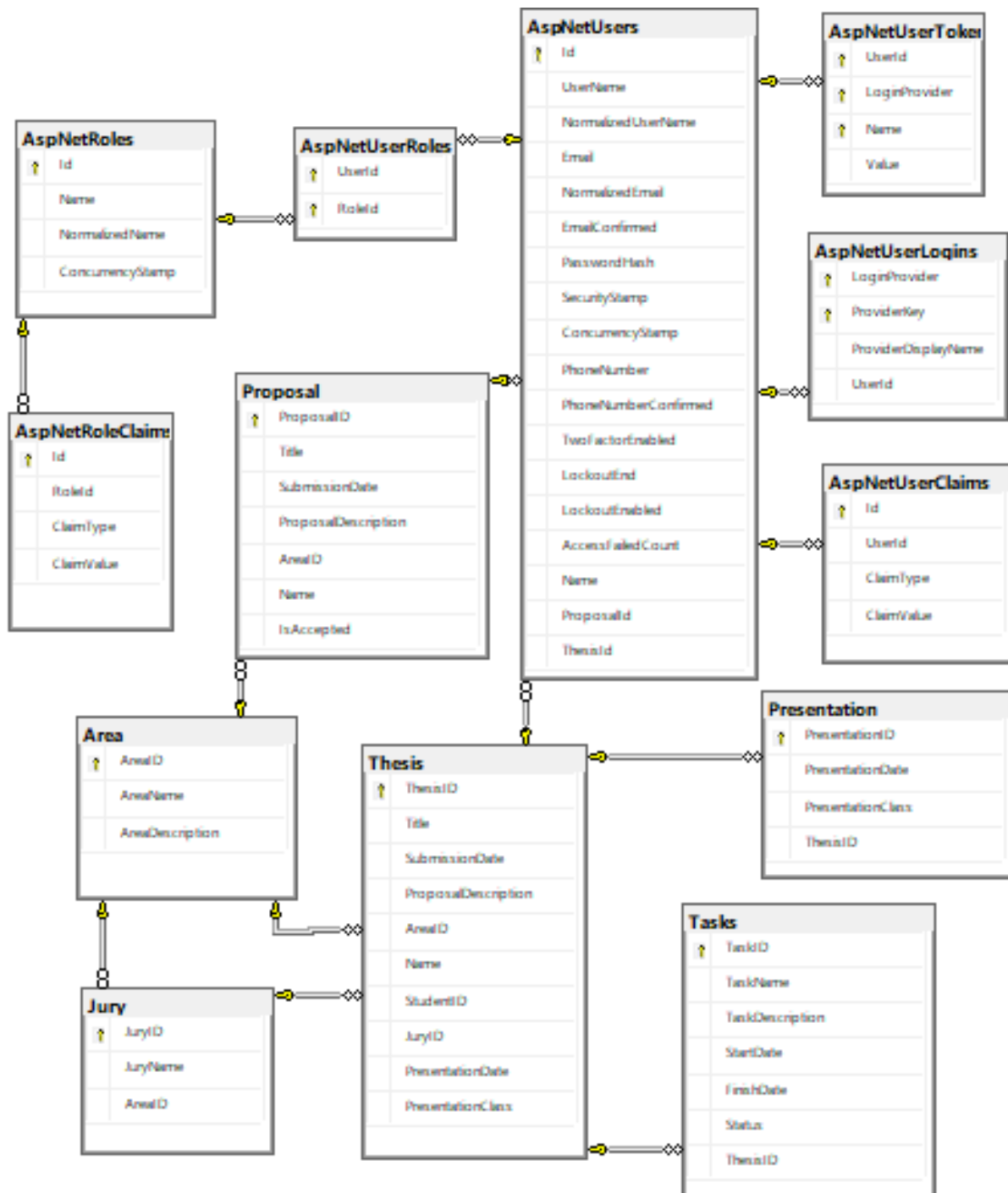


Figure 3.14: Database schema

3.8 Class diagram

The figure 3.15 presents the class diagram which gives us a clear static view of the application and helps us to analyze and describe the responsibilities of the system.

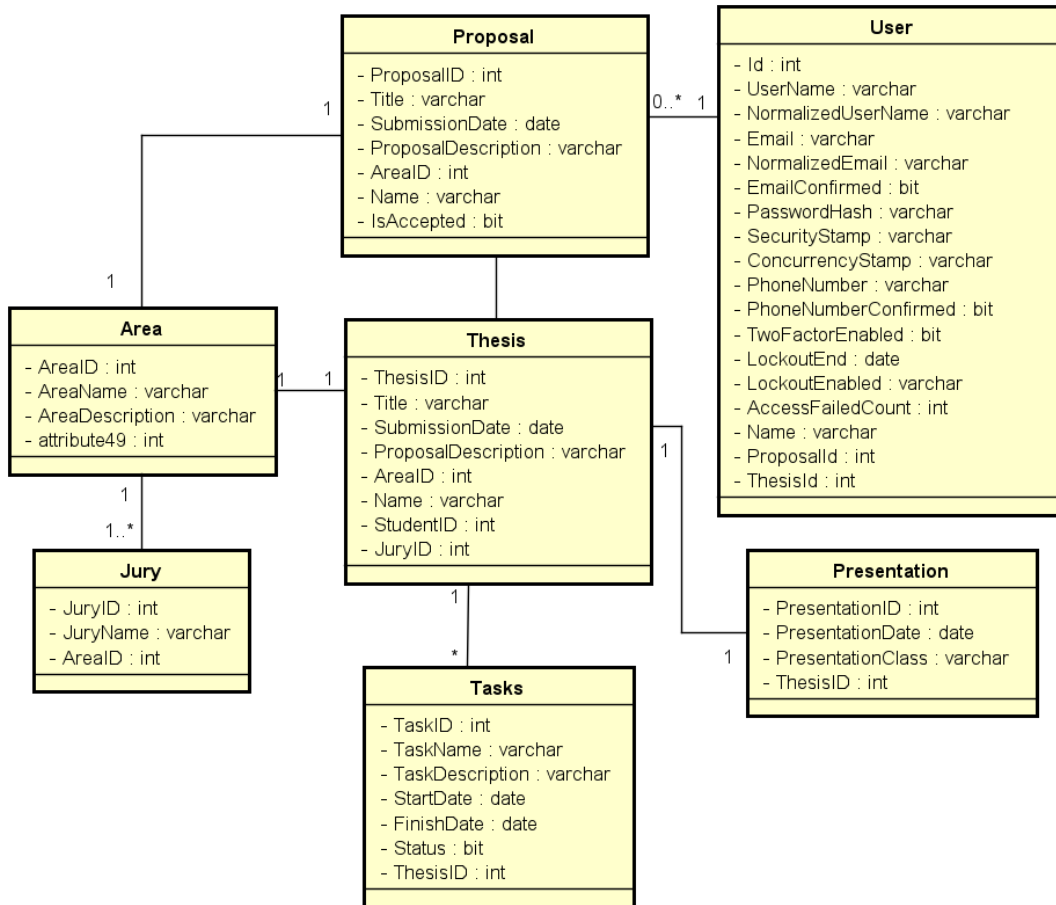


Figure 3.15: Class diagram

Conclusion

This chapter has clearly identified the system actors and the functionalities that the software should offer. The next chapter will describe the technologies and development tools.

Chapter 4

Technologies and development tools

4.1 Front-end frameworks

Every year, the front-end frameworks improve and add new functionalities to find a place within the yearly framework challenge.

In order to find the leading front-end frameworks and libraries in 2020, we took GitHub as a reference since it is the largest Git-repository hosting service globally [15].

So we managed to extract these data displayed in figure 4.1.

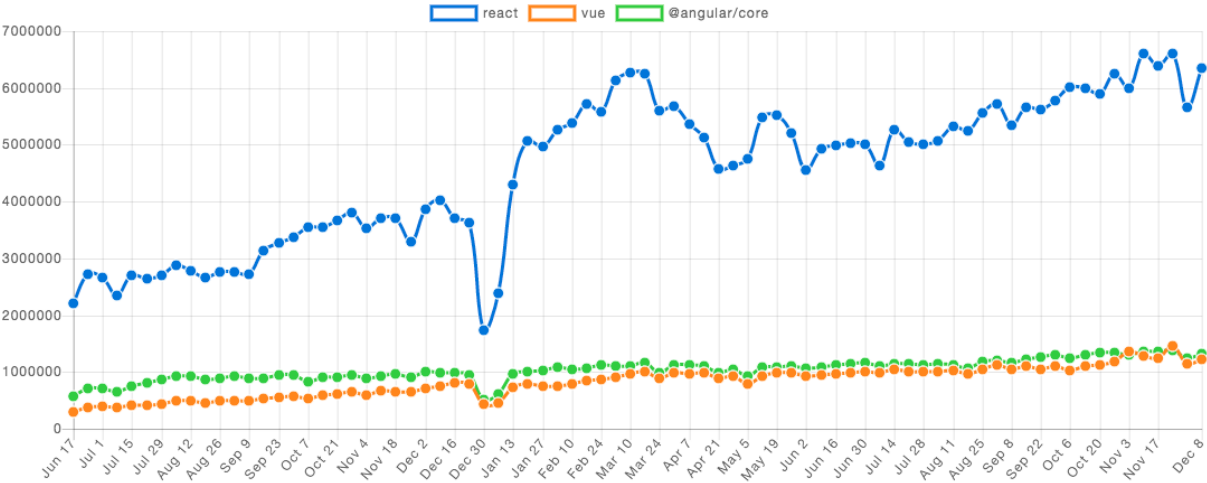


Figure 4.1: Frameworks downloads in the past two years
Source: GitHub[6]

4.1.1 AngularJS

Angular is a known JavaScript framework and an open source developed by Google in 2010. Angular's first development intention is to help web designer in creating a persistent and efficient web forms.

The core concept of Angular is based on two-way data binding in web browsers in order to reduce the back-end's data processing responsibility in web servers. Two-way binding architecture is presented in figure 4.2.

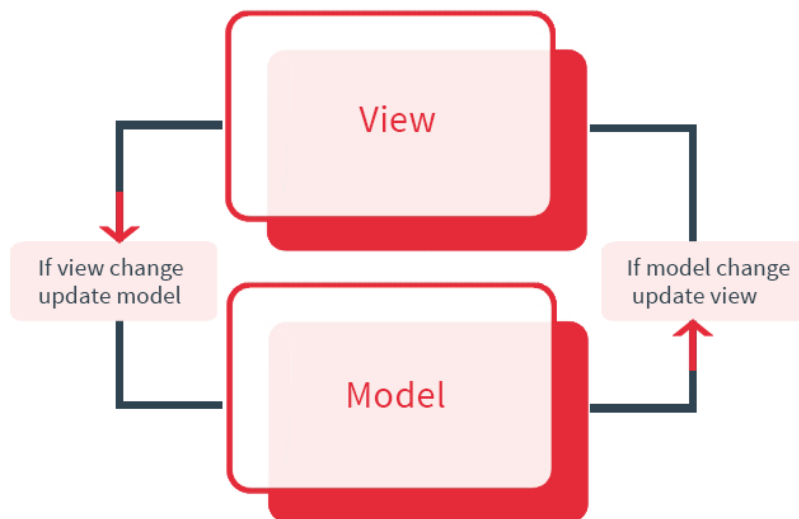


Figure 4.2: Two-way binding architecture
Source: Henrique Custódia [4]

4.1.2 Vue

Vue.js is a known JavaScript ES6 based open source framework developed by Evan You in 2014. Vue's first development intention was to use user interface (UI) components with easy application programming interface in order to achieve responsive data binding.

Figure 4.3 presents the Vue.js data-driven concept:

- View : responsible of displaying the website content.

- View Model : It contains the DOM listeners and data bindings.
- Model : Represents the plain JavaScript objects.

Vue uses DOM listeners with one way binding in order to achieve the result of two-way binding and that's how Vue can provide both two-way and one-way binding option in the data process. Compared to React and Angular , Vue distinguished by its minimal volume and efficient rendering and processing.

Although its front-end superiority, Vue's technical support is not reliable due to its limited team scale development with the unexpected updated plans.

Its small volume means that it includes less features than other frameworks. That means that Vue is suitable for small projects.

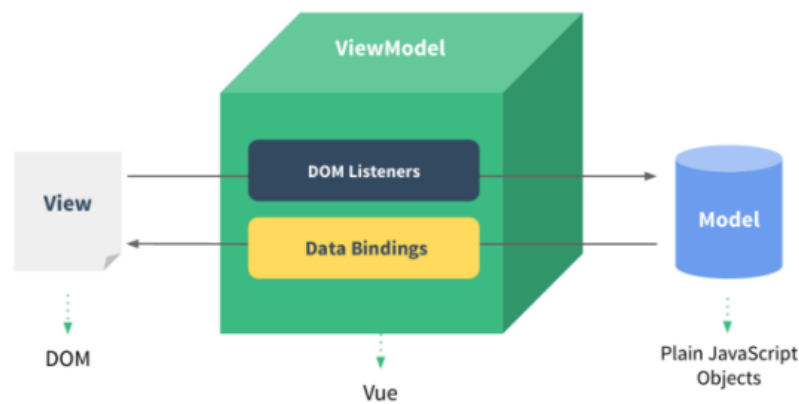


Figure 4.3: Vue.js data-driven concept
Source: Vue.js Official Blog Reactive Data Binding [2]

4.1.3 React

React is an open source JS library developed by Facebook and used for building UIs. It allows developers to modify data without any page reloading. To start with, React is using a specific syntax called JSX that combine HTML with JS which make it simple to use. And unlike other Languages, React is very easy to learn, it requires a basic experience of CSS and HTML. Moreover , this open source supports the code reusability using the Native Approach by that one code can generate Web, Android and iOS applications. So

it's not only easy and simple to learn but also a time saver.

Practically, React is using Flux as architecture which make the debugging more easier because the data flow to components is controlled by one point called the Dispatcher and for testing , ReactJs applications are easy to test.

React is using a virtual DOM with one-way data binding which means that the flow of data is directed towards one direction as presented in figure 4.4. DOM (Document Object

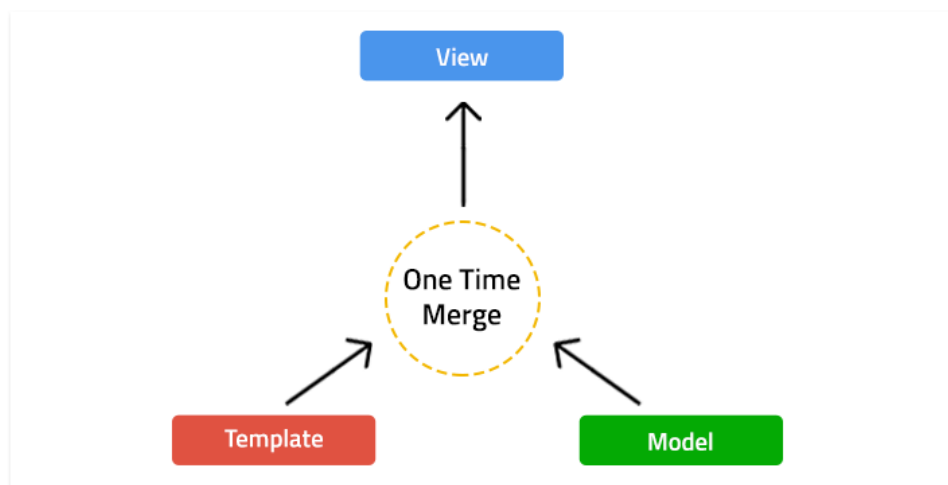


Figure 4.4: One-way data binding architecture
Source: InspironLabs [8]

Model) is an object oriented representation of the web page. It represents all the nodes in an HTML document.

DOM provides a way for JavaScript to interact with every single node in an HTML document. It represents the user interface of the application, so every time there is a change in the state of the application, DOM gets updated.

4.2 Back-end frameworks

Beside the front-end frameworks, back-end frameworks also are improving by time. One did some research about the back-end frameworks leaders and managed to extract these data displayed in the table below and figure based on GitHub and Stack Overflow scores.

Table 4.1: Measuring web framework popularity based on GitHub and Stack Overflow scores

Position	1	2	3	4	5
Framework	ASP.NET	Ruby on Rails	ASP.NET MVC	Django	Laravel

Source: HotFrameworks [7]

To confirm the popularity of the ASP.NET through the time, in figure 4.5 presents a graph of the web framework popularity between the second quarter of 2015 to first quarter of 2019 based on Stack Overflow and GitHub scores. Next sections explores the back-end frameworks leaders in details in order to choose the suitable one for our application.

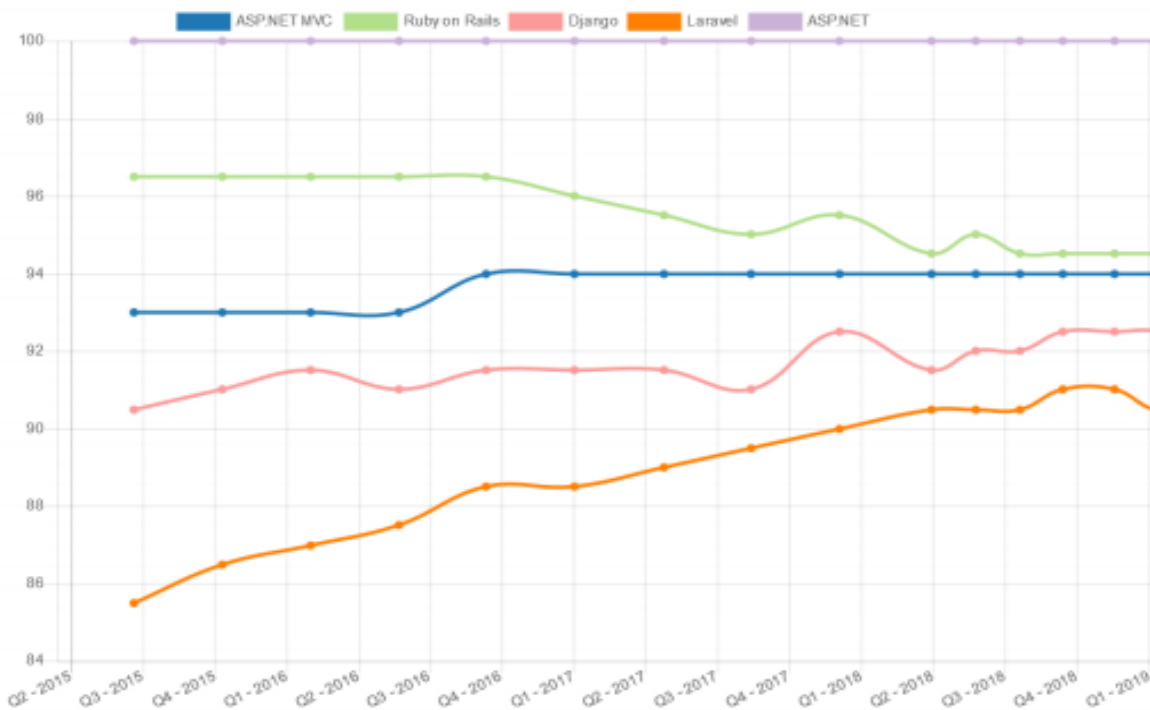


Figure 4.5: Graph of the web framework popularity based on GitHub and Stack Overflow scores

Source: HotFrameworks [7]

4.2.1 ASP.NET and ASP.NET MVC

ASP.NET is a web application framework developed by Microsoft and it's using the Common Language Runtime (CLR). ASP.NET is running on Internet Information Services (IIS).

ASP.NET MVC was released by Microsoft on 2013 and it is based on the Model-View-Controller (MVC) architecture. In fact, it implements the MVC pattern and provides such a great separation between the different layers which is suitable for Web applications.

4.2.2 Ruby on Rails

Rails is an open source launched on 2014, it uses the Model-View-Controller (MVC) pattern. Ruby on Rails official blog provides multiple guides to follow and to getting started with the framework which is perfect for new developers to the framework. Rails is based mainly on two principals:

- Convention over configuration.
- Don't repeat yourselves (DRY)

4.2.3 Django

Django is a high-level Python Web framework. It was created in 2003 when the programmers working for the Lawrence Journal-World newspaper started using Python to build apps. In 2005, the programmers decided to release the framework as an open source and it was named after the famous jazz guitarist Django Reinhardt.

Django is a cross-platform supporting the MVC pattern architecture. The MVC programming paradigm helps developers to separate the both of business logic and user interface which make the web applications faster and more simple.

Moreover, this framework provides a well documented starter guides for beginners, and since it is related to Python, help is really easy to get while developing web applications using Django.

4.2.4 Laravel

Laravel is an MVC framework released by Taylor Otwell. It is based on Hypertext Pre-processor (PHP). Laravel is a clear, simple and well documented framework. In term of authentication, Laravel provides developers with a really simple implementation because almost everything is configured out of the box.

In term of documentation, the framework provides a clear documentation so it helps developers to learn and develop quickly.

4.3 Development tools and technologies definition

To create a web application, every developer needs to choose two frameworks. One for the back-end and another for the front-end. The front-end framework allow the programmer to develop the web application interface whereas the back-end framework represent the heart of the web application, it contains all the methods developed in an API to be consumed on the front-end.

The chosen frameworks and technologies for our application will be discussed in the next sections.

4.3.1 Front-end Framework

After a deep research, ReactJs was selected as a front-end framework for the application for many reasons.

In term of techniques, React js uses the Virtual DOM which gives the ReactJs many good reasons to be the number one.

Compared to other systems, the virtual DOM does not save observables in the memory. So it helps the application to optimize the memory usage.

Beside that, the virtual DOM facilitates the work flow for the developers since the data binding is not intertwined with the application because the JavaScript code updates the components of React while React updates the virtual DOM.

Virtual DOM is updating the application which increases the performance and make our application more efficient.

4.3.2 Back-end Framework

According to the statistics listed in the 3.5 section, ASP.NET is one of the most stable back-end framework during the time. ASP.NET Core was selected to develop the back-end of the application.

The ASP.NET framework is able to provide a high performance applications especially with its new features and regular updates which really helps the developers to provide an optimized code. ASP.NET Core provides a lesser code, and as coding is less, the required coding time will be less also what makes ASP.NET Core much more time-effective.

In fact, less code also means easier maintenance.

4.3.3 Other technologies

After the choice of the front-end and the back-end frameworks, now it is time to define the other tools and technologies to use in our application.

4.3.3.1 Microservices

Microservices are presented as a design pattern developed by Microsoft and based on application division into small independent modules interacting with each other.

The decomposition makes deployment much easier unlike other architectures.

Microservices not only facilitate the codebase for the dev team but also it makes testing more simple as well as fixing bugs without affecting the other parts.

Furthermore, Microservices allow a developer to use different programming languages as long as the services are communicated with each other. Figure 4.6 presents the microservices architecture.

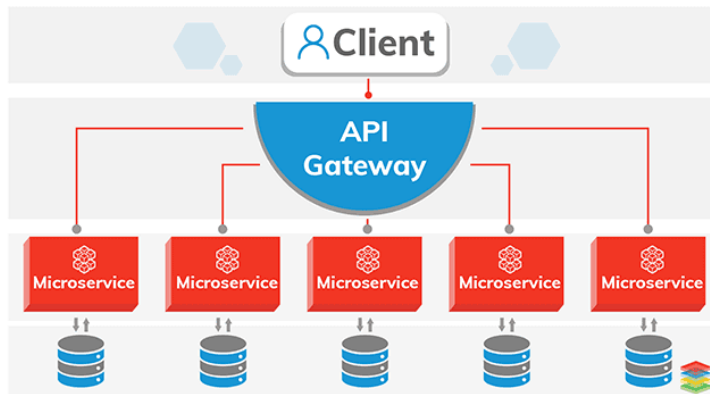


Figure 4.6: Microservices architecture
 Source: Microservices Architecture and Design Patterns [16]

4.3.3.2 Entity framework

Entity Framework is an object-relational mapper that allows ASP.NET developers to use .NET objects in database. It makes life easier for developers especially by eliminating the need for the data-access code. Figure 4.7 illustrates the overall architecture of the Entity Framework : Let's take an individual look at the components of Entity Framework.

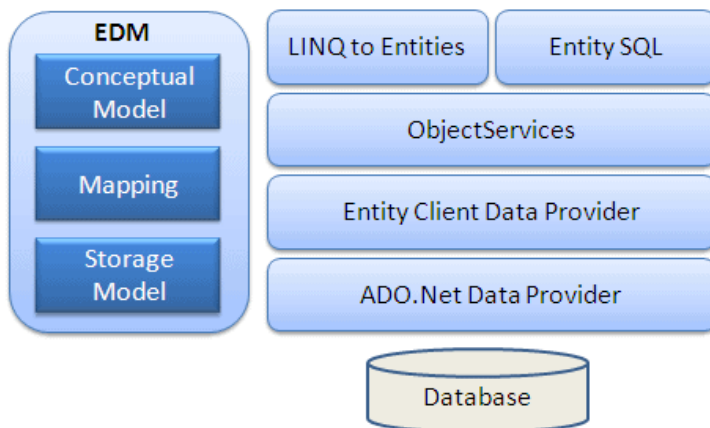


Figure 4.7: Entity Framework architecture
 Source: Entity Framework Architecture [5]

- **EDM (Entity Data Model)** : The EDM is composed of three main parts, Conceptual model, Mapping and Storage model.
- Conceptual model : It contains the model classes and the associated relationships.

- Storage model : This layer presents the database design model including tables, views, procedures and associated relationships.
- Mapping : This layer gives us information about the mapping between the conceptual model and the Storage model.
- **LINQ to Entities** : Can be called also (L2E), It is a query language used to write queries. It returns entities, which are already defined in the conceptual model layer.
- **Entity SQL:** Entity SQL is another query language and it is quite similar to LINQ.
- **Object Service:** This layer presents the principal entry point for access to data from db and return it back. The object service is the responsible of converting data from an entity client data provider (the next layer) into an entity object structure.
- Entity Client Data Provider:** This layer is responsible to convert Entity SQL or LINQ-to-Entities queries to SQL queries which is understood by the associated database. Entity Client Data Provider is in communication with ADO.Net Data Provider (the next layer) which communicates directly with the database.
- ADO.Net Data Provider:** This ADO.Net Data Provider is in direct communication with the database using standard ADO.Net. It can also be used to retrieve results and to execute commands.

In the application, one applied the Database first approach. It creates the models codes (classes, properties, etc.) from an existing database and these classes become the link between the database and the controllers.

4.3.3.3 Astah

Astah is an UML modeling tool that we have used throughout modeling our application.

4.3.3.4 SQL Server Management Studio

Microsoft SQL Server Management Studio is an integrated environment managing a SQL Server infrastructure. It provides several tools with rich script editors that interact with SQL Server.

4.3.3.5 ASP.NET Core Identity

ASP.NET Core Identity is a membership system that provides the ASP.NET Core apps with the login functionality.

The membership system standardizes user store with structure and methods to manage the store. So users can create accounts by providing their personal data or they can also use an external provider, such as Facebook, Google, etc.

ASP.NET Identity allows us to persist the user data into an existing SQL database like we did in our case. We will explain the ASP.NET Core Identity architecture in figure 4.8

- **Identity Manager** : This layer is presented as a high level class which is used by applications to manage Identity models like user, role etc.

- **Identity Store** :The Identity Store present the domain model.ASP.NET Core Identity provides the basic Interfaces for these models

- **Data Access Layer** : This layer provides the logic to store the data from ASP.NET Core Identity to data source.

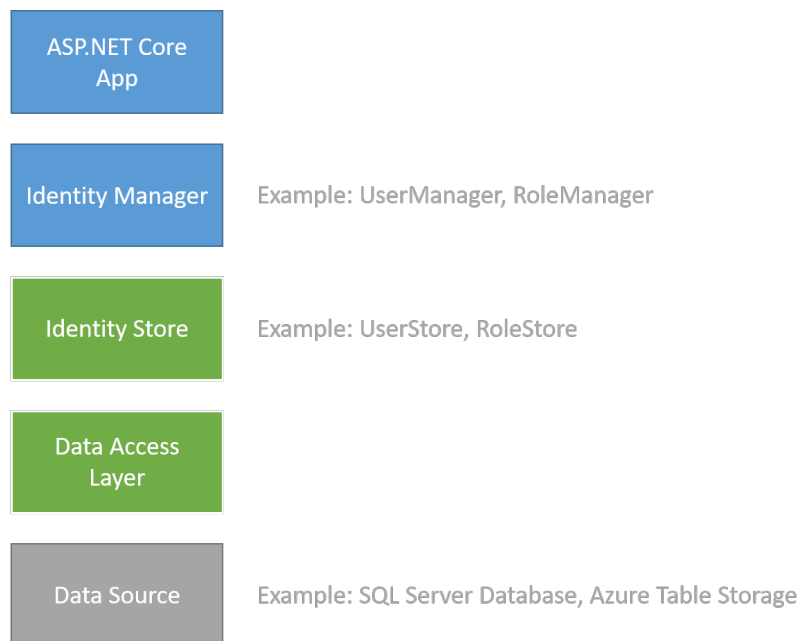


Figure 4.8: ASP.NET Core Identity architecture
Source: Microsoft [9]

4.3.4 MVC architecture

MVC stands for Model, View, and Controller. It's an architecture organizing the Human-Machine interface and it separates an application into three components - Model, View, and Controller

View : Is a user interface. View display model data to the user and also enables them to modify them.

Model : Model represents the data.

Controller : The controller handles the user request. All the actions taken by user in the view are handled by controllers.

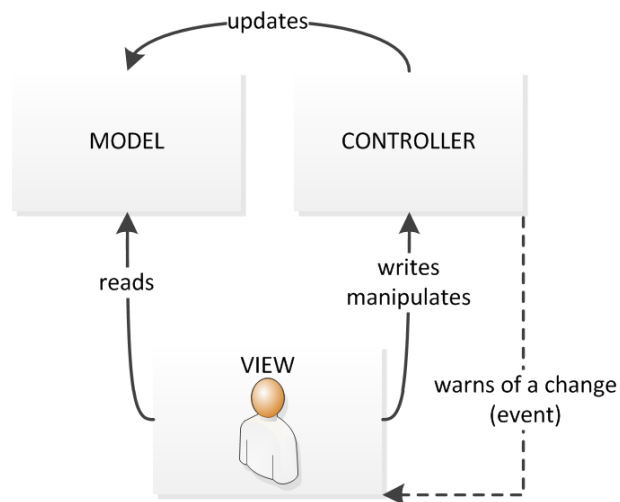


Figure 4.9: MVC architecture

Chapter 5

Development of iMaster Process platform

This chapter focuses on the implementation of the application's components as it presents the accomplished work.

5.1 Results

The user interfaces are really important to present our application in a clear way.

In this section we will present the implementation results with some screenshots for the different users of iMaster.Process application.

5.1.1 Login window

When launching the application, the user will be redirected to the login window presented in figure 5.1.

This window is common for all the user of iMaster(Supervisor, Student, Institute director and Scientific committee member).

The background image is the main entry for the ESTIG department in Instituto Politécnico de Bragança.

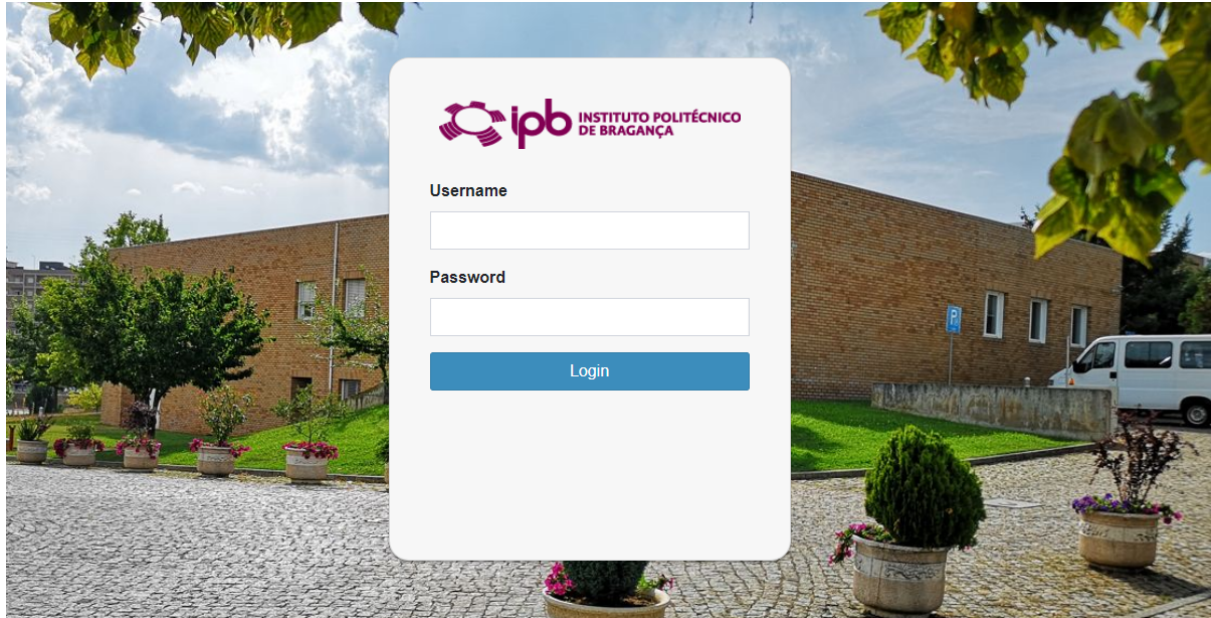


Figure 5.1: Screenshot for the login window

5.1.2 Pages for Student

Every student have the access to consult the available thesis list like presented in figure 5.2.

Title	Submission Date	Area	Supervisor
SAD	2020-08-08T00:00:00	Agraria	Jose
iMaster Schedule	2020-09-24T00:00:00	Agraria	UsernameTester
Pierogi	2020-11-20T00:00:00	ESTIG	Supervisor
Kafteji	2020-12-31T00:00:00	ESTIG	Supervisor
New proposal	2020-12-31T00:00:00	ESTIG	UsernameTester
iMaster Statistics	2020-12-31T00:00:00	ESTIG	Supervisor

Figure 5.2: Screenshot for consulting the available thesis list window

All of the displayed thesis are not assigned yet. The student can consult his current

thesis as the figure 5.3 shows.

The student can check the tasks list uploaded by his supervisor as shown in figure 5.4.

The task list contains the task name, task description, start and finish date, and the tasks status.

The main goal of the tasks is to help the supervisor to monitor the work for every student.

When the task is done, the supervisor will change the status to green.

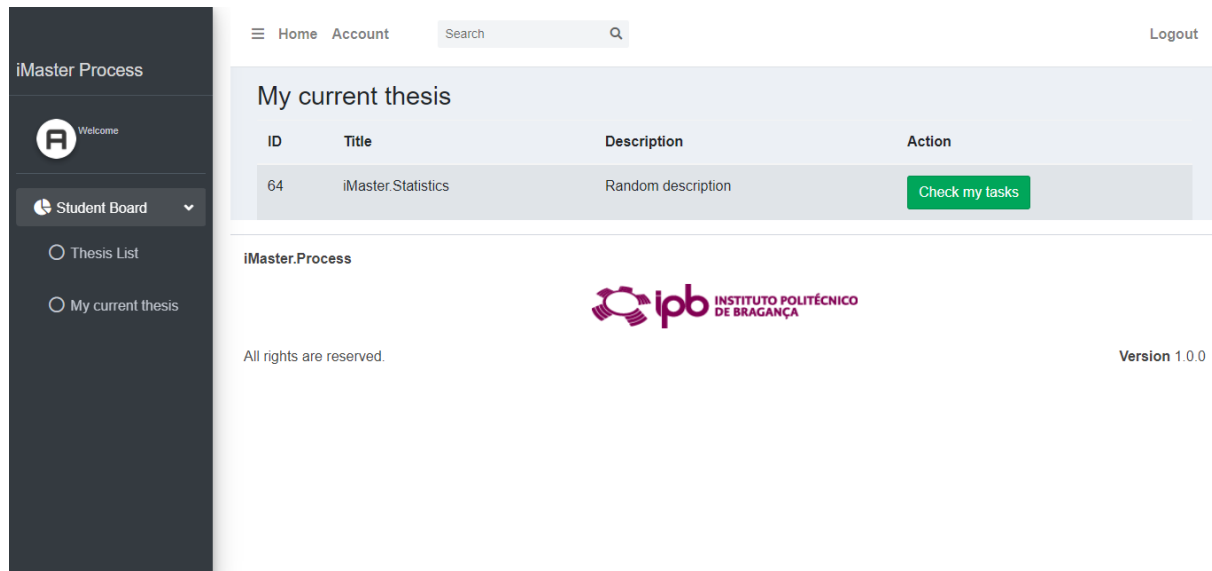


Figure 5.3: Screenshot for consulting the current thesis window

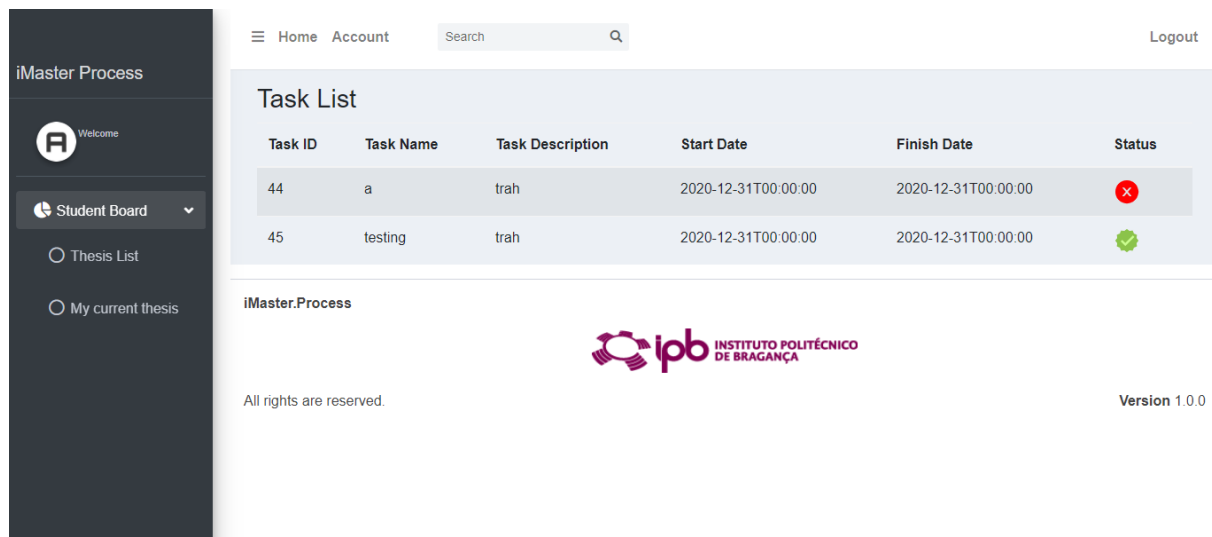


Figure 5.4: Screenshot for consulting the current task list window

5.1.3 Pages for Scientific committee member

Every scientific committee member can consult the students list and the jury members list as shown in figure 5.5 and figure 5.6.

ID	Name	Email
25d8066f-9254-4a89-9b75-d4d6c38004ea	Bechir Belhaj	bechir@gmail.com
34235375-293e-4a4c-8081-d8f8d8b5eff7	student dds	student@gmail.com
66ca4a36-b992-48e6-b2b4-225ca4c7c1ce	Wissem Guiza	wissemguiza@gmail.com
f6b130ce-bd12-4df6-ba89-e4485eb23aa9	Student	Student@gmail.com

Figure 5.5: Screenshot for the student list window

Name	Area
Touati	ESTIG
Asma	ESTIG
Amdouni	Agraria
Yassine Mkadem	Agraria
Pro. Paulo Alves	ESTIG
Prof Jose Fernandes	ESTIG
Paulo Matos	ESTIG

Figure 5.6: Screenshot for the jury list window

The scientific committee member can consult all the submitted proposals by supervisors

to be analysed in further notice like the figure 5.7 shows.

The proposals list is composed by proposals title, submission date, a small description, specified area , supervisors name.

Title	Submission Date	Proposal Description	Area	Supervisor	Action
gerrts	2020-08-08T00:00:00	Uesertest	ESTIG	New Super	Analyse Delete
iMaster.Schedule	2020-12-31T00:00:00	This is a random description	ESTIG	Usenametester	Analyse Delete
gesssss sssrrtds	2020-08-06T00:00:00	Uesertest	Agraria	Jose	Analyse Delete
Sunday Test	2020-10-30T00:00:00	asd asda sd	ESTIG	UsernameTester	Analyse Delete
Pierogi	2020-11-21T00:00:00	Random description as usual	ESTIG	Supervisor	Analyse Delete
Kafteji	2020-12-31T00:00:00	behi	ESTIG	Supervisor	Analyse Delete
New proposal	2020-12-31T00:00:00	a	ESTIG	UsernameTester	Analyse Delete

Figure 5.7: Screenshot for the proposals list window

If the scientific committee member clicks on "Analyse", he will be redirected to figure 5.8 where he will have the duty of analysing the selected proposal.

ID	Title	SubmissionDate	ProposalDescription	Area	Supervisor	Action
1	iMaster.Schedule	2020-12-31T00:00:00	This is a random description	1	Usenametester	Accept Ask for reformulation

Figure 5.8: Screenshot for the analysing proposal window

The scientific committee member have two options, either to accept the proposal or to ask for reformulation. If he asks for reformulation he will be redirected to figure 5.9. The "Name" and "Email" are filled automatically and its for the correspondent supervisor.

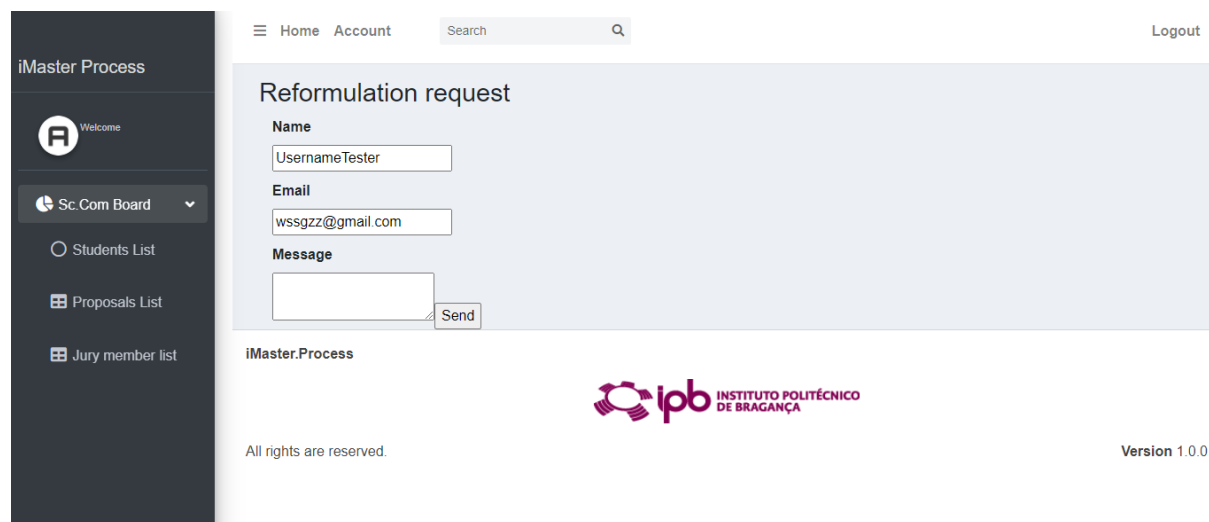


Figure 5.9: Screenshot for asking for reformulation window

Figure 5.10 presents the mail recieved by a supervisor to reformulate a proposal.

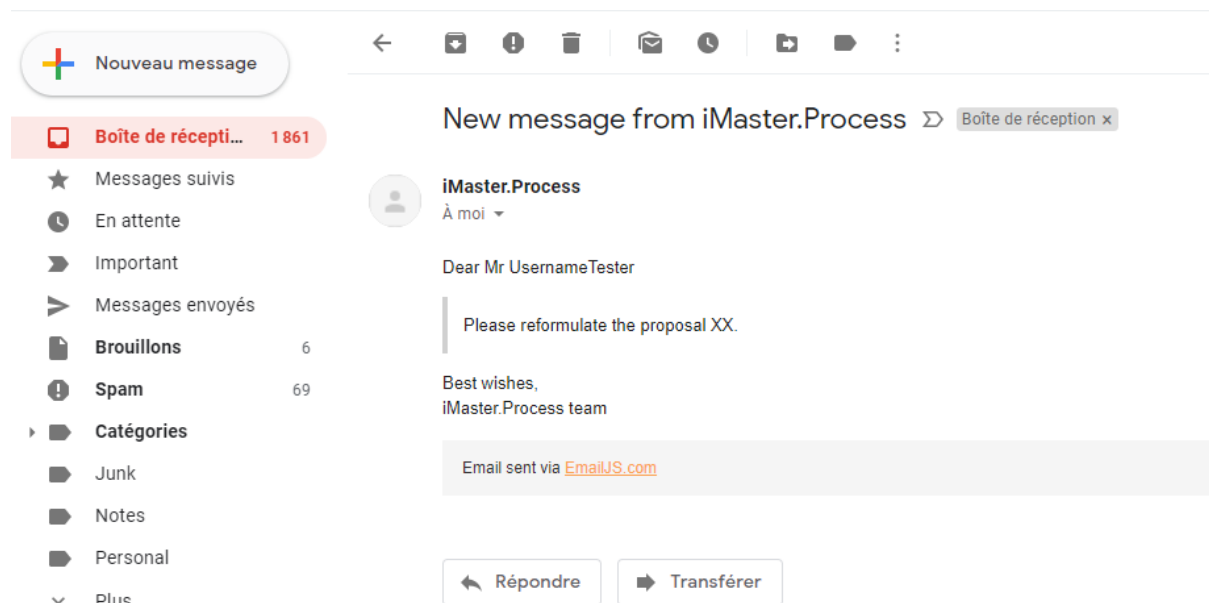


Figure 5.10: Screenshot for the received mail window

He can also consult the accepted thesis list like in figure 5.11

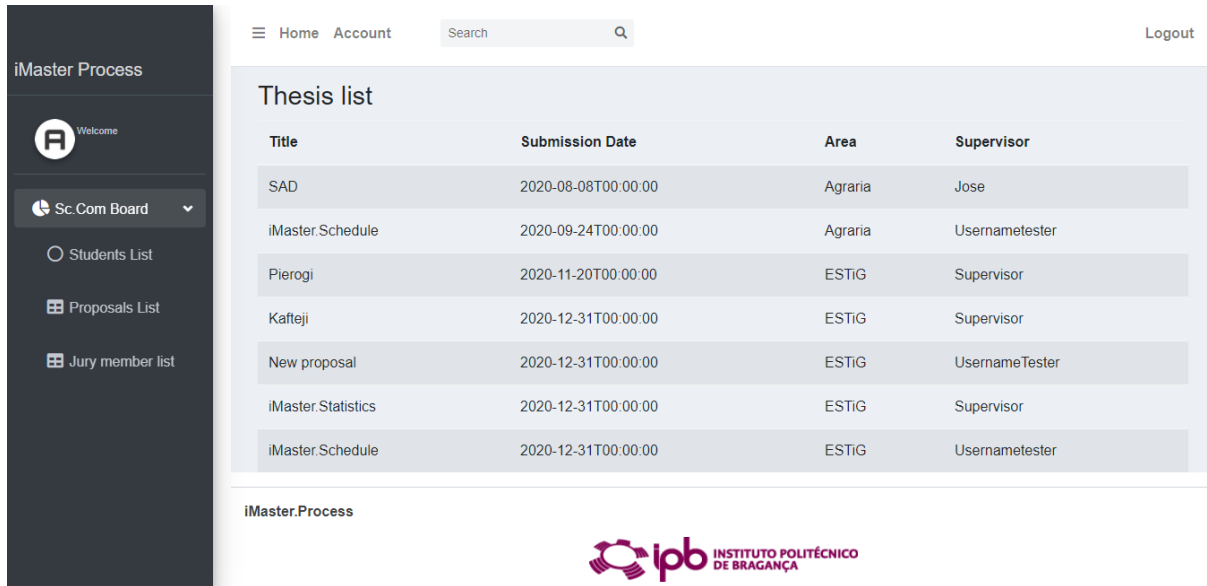


Figure 5.11: Screenshot for the accepted thesis list window

5.1.4 Pages for Master Director

The master director can consult the accepted thesis list which were already assigned to students as shown figure 5.12 in order to fix the presentation for each thesis.

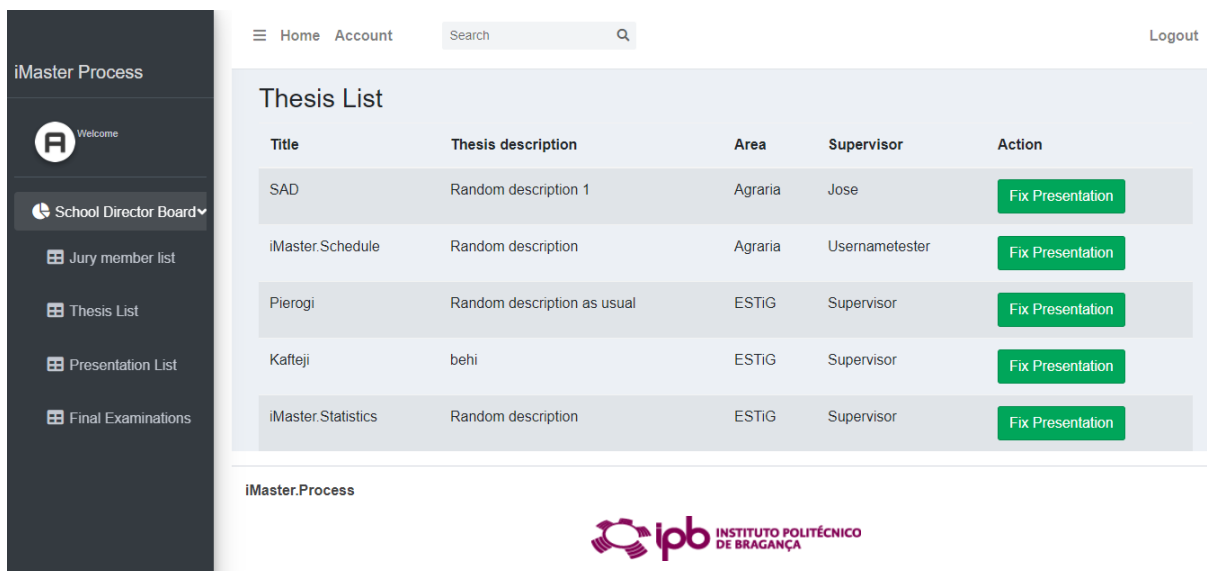


Figure 5.12: Screenshot for consulting the thesis list window

If the master director click on "Fix Presentation" he will be redirected to another page to enter the new presentations details for the selected thesis like the figure 5.13 shows.

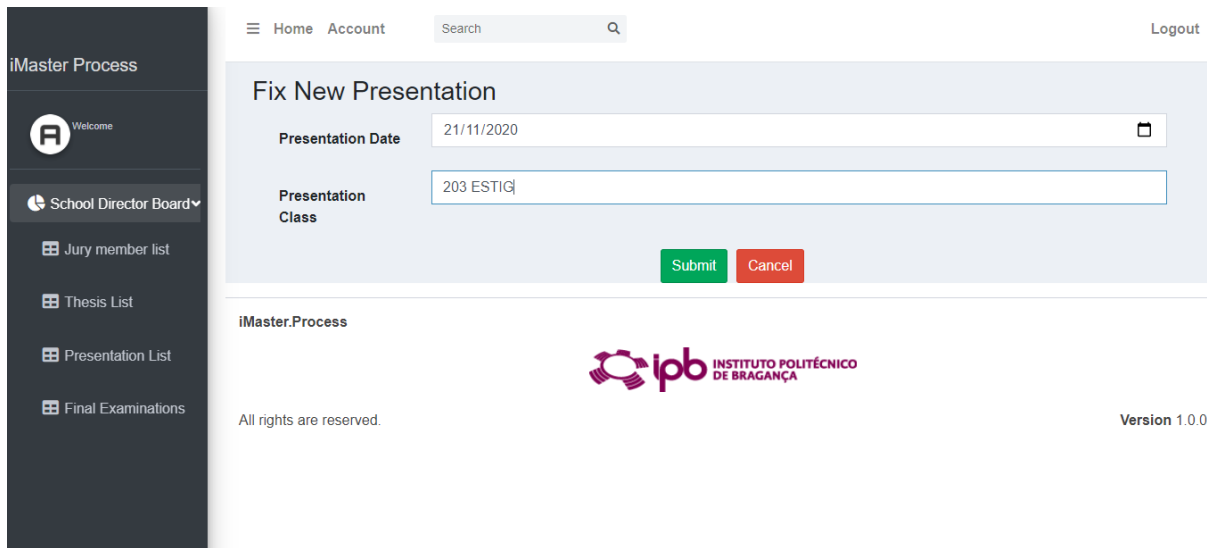


Figure 5.13: Screenshot for fixing the presentation for a thesis window

After fixing the date and class for the selected thesis, the master director will be redirected to another page where he can check the upcoming presentations and where the final examination will be for each thesis like shown in figure 5.14.

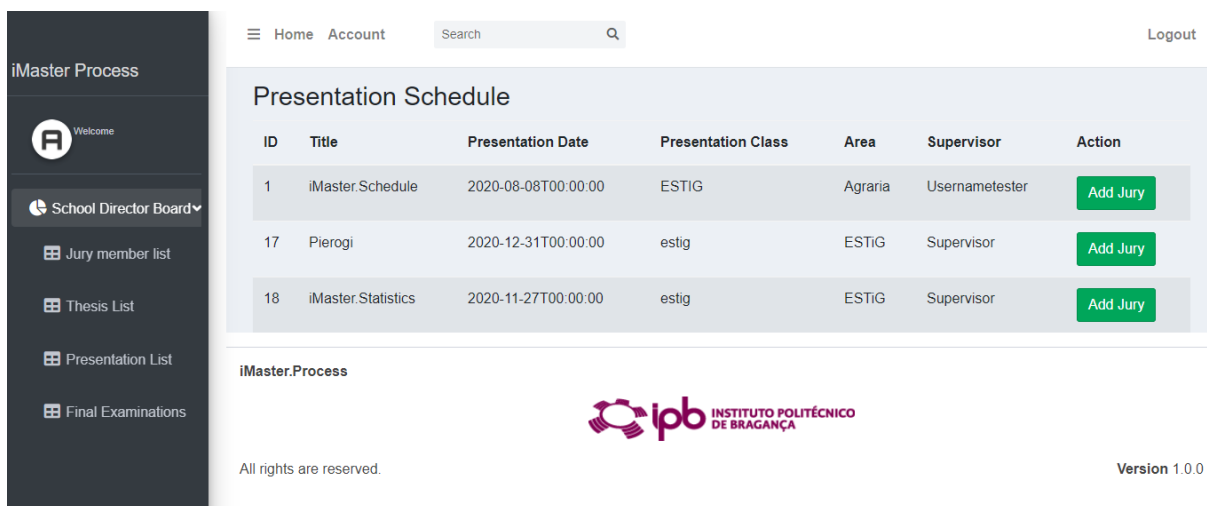


Figure 5.14: Screenshot for consulting the final presentation schedule list window

The master director will have one last duty which is to select the jury members for

the selected presentation. He can do that by clicking on "Add Jury" where he will be redirected to another page to select the jury members from the available list like the figure 5.15 shows.

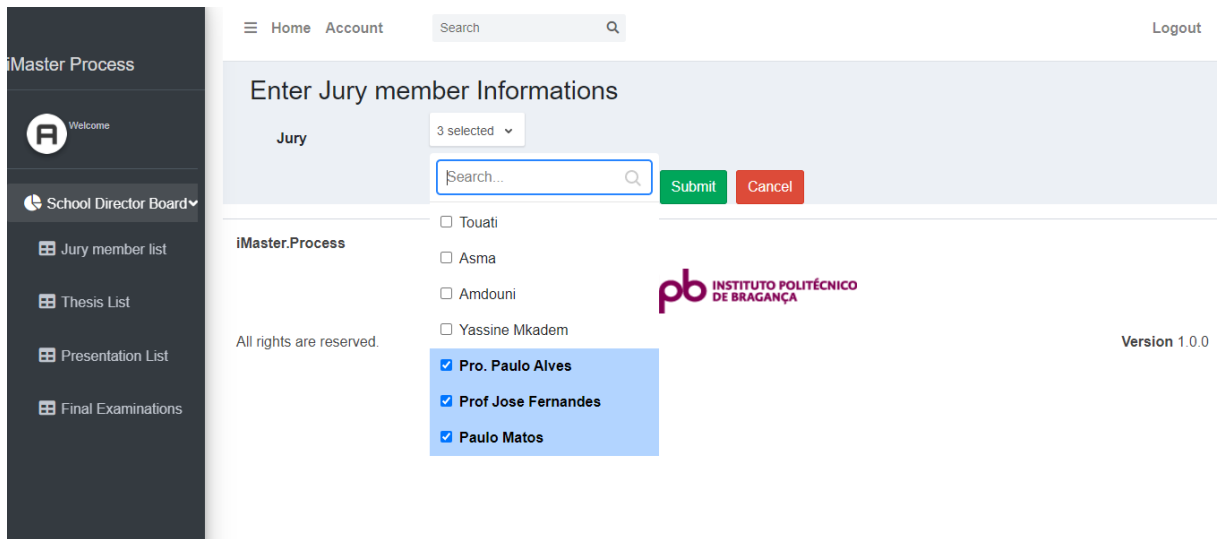


Figure 5.15: Screenshot for assigning jury members for the final examination

After selecting the jury members, the master director will be redirected to the final examination list which is the list of all the confirmed final examinations with defined class, date, and jury members like shown in figure 5.16.

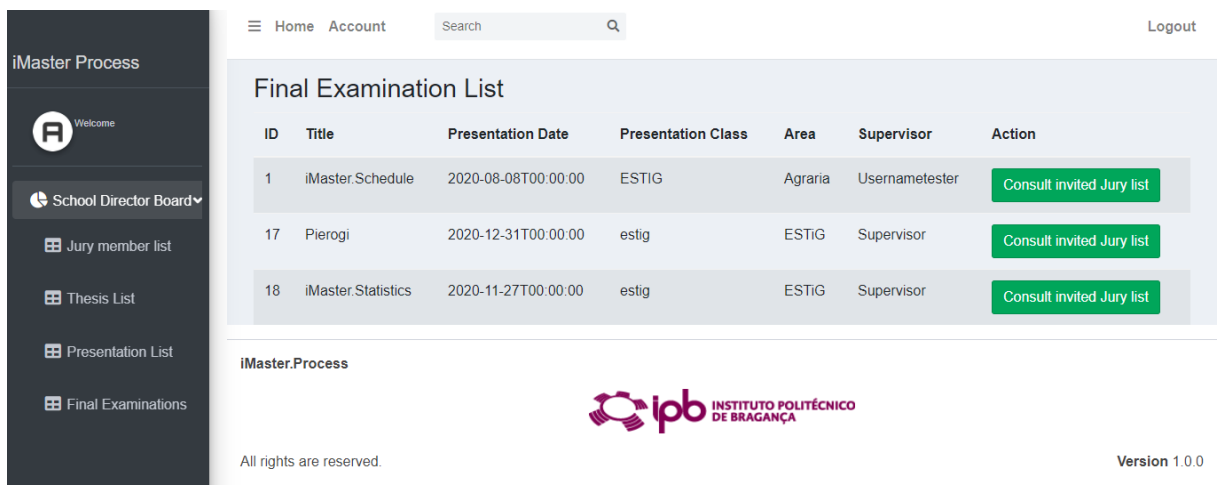


Figure 5.16: Screenshot for the confirmed final examinations list

The master director can always consult the invited jury members for each final examination by clicking on "Consult Invited Jury" like presented in in figure 5.17.

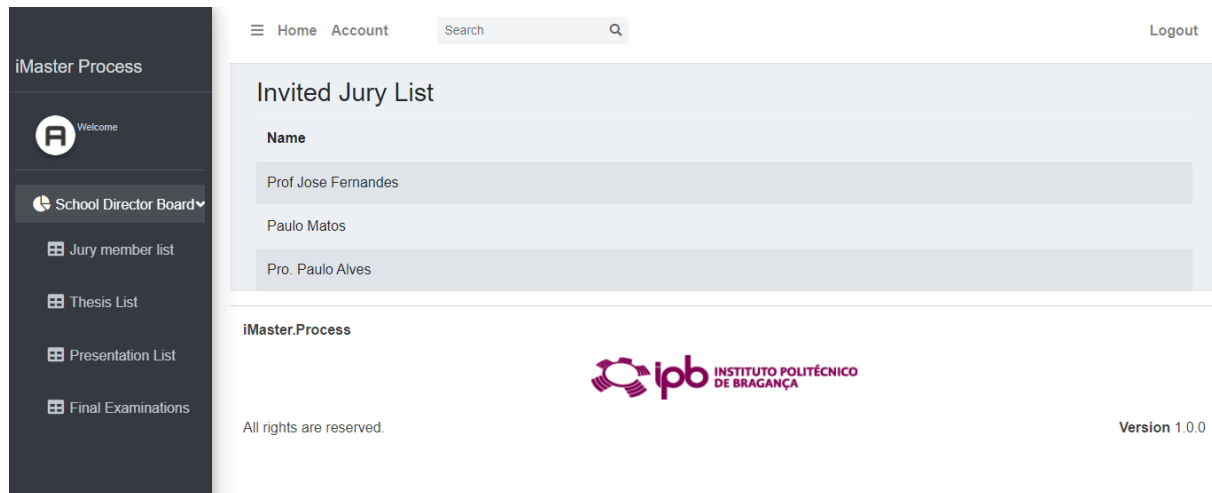


Figure 5.17: Screenshot for the invited jury members for a final examination

5.1.5 Pages for Supervisor

To submit a new proposal, the supervisor have to fill up some informations about the proposal and press "Submit" like the figure 5.18 shows.

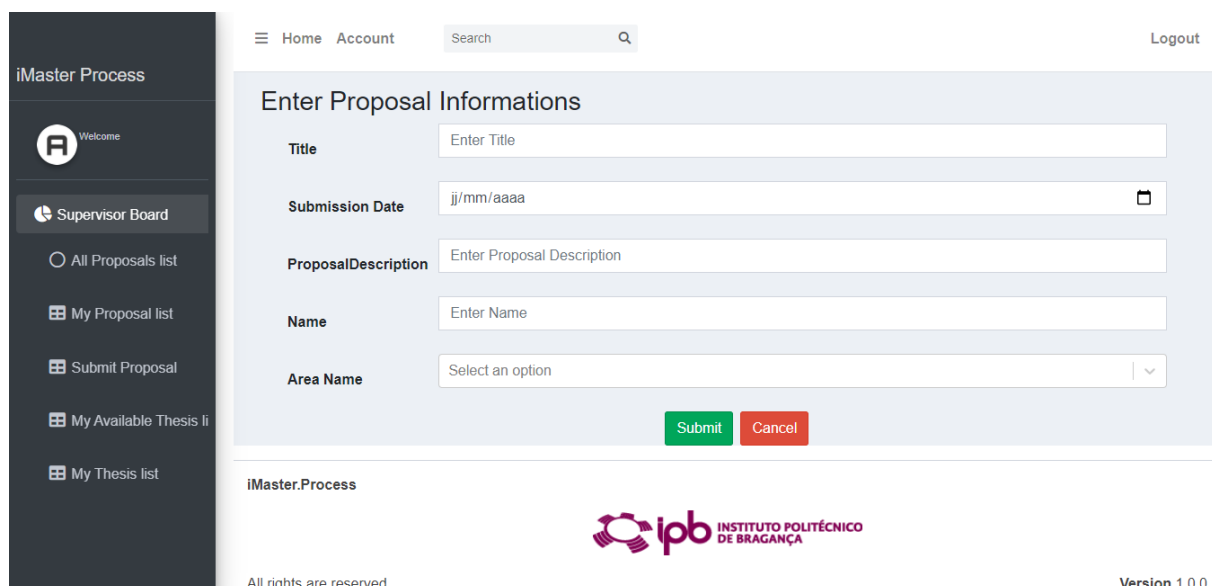


Figure 5.18: Screenshot for the proposal submission window

After submission, the supervisor can consult his own proposals list as presented in figure 5.19 where he has the right to reformulate or delete one of his proposals.

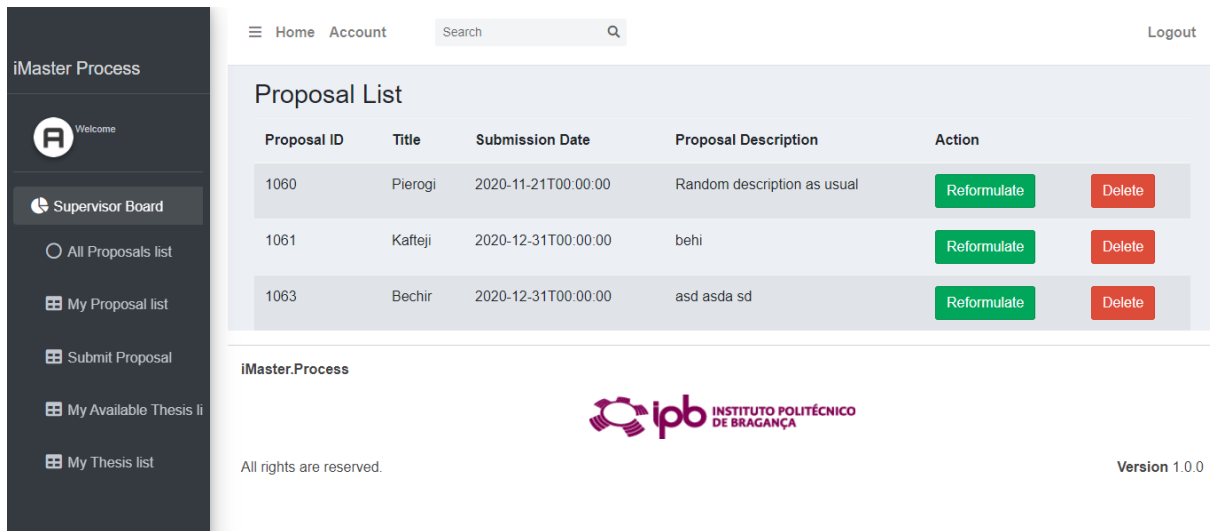


Figure 5.19: Screenshot for the proposals list per supervisor window

If the supervisor chooses to reformulate one of his proposal he will be redirected to a new window like shown in figure 5.20.

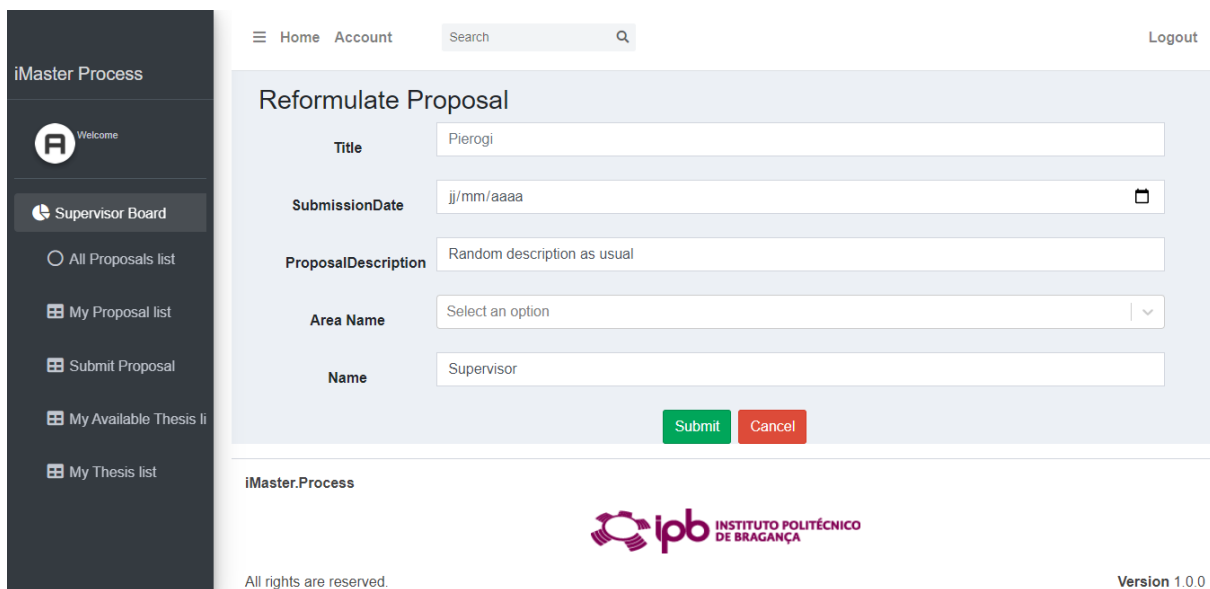


Figure 5.20: Screenshot for the proposals reformulation window

Every supervisor can also check all the submitted proposals by his colleagues as presented

in figure 5.21.

Title	Submission Date	Proposal Description	Area	Supervisor
gerrts	2020-08-08T00:00:00	Usertest	ESTIG	New Super
iMaster.Schedule	2020-12-31T00:00:00	This is a random description	ESTIG	UsernameTester
gesssss sssrrtds	2020-08-06T00:00:00	Usertest	Agraria	Jose
Sunday Test	2020-10-30T00:00:00	asd asda sd	ESTIG	UsernameTester
Pierogi	2020-11-21T00:00:00	Random description as usual	ESTIG	Supervisor
Kaffeji	2020-12-31T00:00:00	behi	ESTIG	Supervisor
New proposal	2020-12-31T00:00:00	a	ESTIG	UsernameTester
Bechir	2020-12-31T00:00:00	asd asda sd	ESTIG	Supervisor

Figure 5.21: Screenshot for the proposals list of all the supervisors window

Once accepted by the scientific committee member, a new thesis will be added to the available thesis and list where the supervisor can assign the work to one of his students like presented in figure 5.22

Thesis ID	Title	Proposal Description	Area	Action
59	Kaffeji	behi	ESTIG	Assign Work Delete

Figure 5.22: Screenshot for the available thesis list window

Once the work assigned to a student, the supervisor will have a list of the in progress thesis where he can monitor the work with the student as shown in figure 5.23

The screenshot displays the 'All Thesis List' interface. At the top, there is a navigation bar with 'Home' and 'Account' links, a search bar, and a 'Logout' button. The main content area features a table with the following data:

Thesis ID	Title	Thesis Description	Area	Student	Action
58	Pierogi	Random description as usual	ESTIG	25d8066f-9254-4a89-9b75-d4d6c38004ea	<input type="button" value="Add Goals"/> <input type="button" value="Check Goals"/>
64	iMaster.Statistics	Random description	ESTIG	f6b130ce-bd12-4df6-ba89-e4485eb23aa9	<input type="button" value="Add Goals"/> <input type="button" value="Check Goals"/>

The footer contains the text 'iMaster.Process', the logo for 'INSTITUTO POLITÉCNICO DE BRAGANÇA', 'All rights are reserved.', and 'Version 1.0.0'.

Figure 5.23: Screenshot for the thesis list in progress window

The supervisor can add a new task for his student after he clicks on "Add Task" where he will be redirected to another window to fill the new task details as presented in figure 5.24

The screenshot shows the 'Add Task Informations' form. The form fields are:

- Task Name:** Enter Task Name
- Task Description:** Enter Submission Date
- Task's Start Date:** jj/mm/aaaa
- Task's Finish Date:** jj/mm/aaaa

At the bottom of the form are 'Submit' and 'Cancel' buttons. The left sidebar contains the following menu items:

- Welcome
- Supervisor Board
- All Proposals list
- My Proposal list
- Submit Proposal
- My Available Thesis li
- My Thesis list

The footer contains the text 'iMaster.Process', the logo for 'INSTITUTO POLITÉCNICO DE BRAGANÇA', 'All rights are reserved.', and 'Version 1.0.0'.

Figure 5.24: Screenshot for the adding task for a thesis window

The supervisor can as well consult the task list for every thesis. The red icon for the status means that the task is not done yet and the green icon means that the task is already

done like presented in figure 5.25 . Only the supervisor can change the tasks status.

The screenshot displays the 'iMaster Process' Supervisor Board interface. The main content area features a 'Task List' table with the following data:

Task ID	Task Name	Task Description	Start Date	Finish Date	Status	Action
44	a	trah	2020-12-31T00:00:00	2020-12-31T00:00:00	✓	Edit Task Enable/Disable
45	testing	trah	2020-12-31T00:00:00	2020-12-31T00:00:00	✗	Edit Task Enable/Disable

The interface also includes a sidebar with navigation options: 'All Proposals list', 'My Proposal list', 'Submit Proposal', 'My Available Thesis li', and 'My Thesis list'. The footer contains the 'iMaster.Process' logo, the text 'All rights are reserved.', and the version 'Version 1.0.0'.

Figure 5.25: Screenshot for monitoring work by supervisor window

Chapter 6

Conclusion

The aim of this project was to facilitate the management of master courses processes in IPB by developing a web application dedicated to the different actors who contribute in this process. The main goals were to handle the submission, approval and assignment process of the masters management and to monitor the development of the work until the final submission and to deal with the jury exam process.

To achieve these goals, one began by reviewing the IPB's master regulations to understand more the process to develop. After an exhaustive research, one could not find any similar applications.

After understanding the master process and choosing the suitable technologies for the application, one started to analyse the requirements for each actor which resulted in the general use case diagram presented in the third chapter of this project.

One continued modeling in the third chapter and presented the activity diagrams as well as the state machine diagram to more explain the workflow and the state transitions of the process to finish modeling with a class diagram to present a clear static view.

After modeling, one started looking for a suitable structure for our application illustrated in the fourth chapter of this project. To finish this chapter one presented site maps of our application to provide an easy tracking process.

To finish, one presented the results in the fifth chapter in form of screenshots of different views for every actor.

Future work include adding functionality for the student to request work from the available thesis list and a functionality of instant messaging between the different users to clarify some points and make things more clear if the available informations were not clear.

During this graduation project, one learned a lot to be ready to pursue new professional challenges. And despite the technical issues that one encountered during the development and the difficult period of the pandemic, the requested goals were achieved with success.

Bibliography

- [1] BetterBuys. *Workzone Review*. <https://www.betterbuys.com/project-management/reviews/workzone/>.
- [2] Vue.js Official Blog Reactive Data Binding. *Reactive Data Binding*. <https://v1.vuejs.org/guide/overview.html>.
- [3] Continu. *The SharePoint Advantages and Disadvantages*. <https://www.continuo.co/blog/sharepoint-disadvantages>.
- [4] Henrique Custódia. *Two-way binding architecture*. <https://medium.com/criciumadev/two-way-data-binding-fd5d71712d28>.
- [5] Entityframeworktutorial. *Entity Framework Architecture*. <https://www.entityframeworktutorial.net/EntityFramework-Architecture.aspx>.
- [6] GitHub. *Frameworks downloads in the past two years*. <https://gist.github.com/tkrotoff/b1caa4c3a185629299ec234d2314e190>.
- [7] HotFrameworks. *HotFrameworks Find your new favorite web framework*. <https://hotframeworks.com/>.
- [8] InspironLabs. *React vs Angular vs Vue JS*. <https://inspironlabs.com/blog/react-vs-angular-vs-vue-js>.
- [9] Microsoft. *The ASP.NET Core Identity architecture*. <https://docs.microsoft.com/en-us/aspnet/core/security/authentication/identity-custom-storage-providers?view=aspnetcore-5.0>.

- [10] Microsoft. *SharePoint in Education*. <https://support.microsoft.com/en-us/office/videos-sharepoint-in-education-7db5f601-2e5f-4dd8-9584-5c82bc504c0e>.
- [11] Software Portal. *Workzone Review*. <https://softwareportal.com/project-management-software/workzone-review/>.
- [12] project-management.com. *Workzone Pros and Cons*. <https://project-management.com/workzone-pros-and-cons-with-3-workzone-alternatives/>.
- [13] BPA Solutions. *SharePoint project management*. <https://www.bpa-solutions.net/sharepoint-project/sharepoint-project1/>.
- [14] TechTerms. *SharePoint Definition*. <https://techterms.com/definition/sharepoint>.
- [15] Wikipedia. *GitHub definition*. <https://en.wikipedia.org/wiki/GitHub>.
- [16] Xenonstack. *Microservices Architecture and Design Patterns*. <https://www.xenonstack.com/insights/microservices/>.

Appendix A

Detailed use cases

Table A.1: Consult all thesis list

Use Case	Consult all thesis list
Actors	Student
Goal	This use case allows the student to consult all the available thesis list
Scenario	<ol style="list-style-type: none">1. The Student clicks on "Thesis List" in the side menu bar. He will receive a list of all the available thesis.
Exception:	<ol style="list-style-type: none">1. The available thesis list is empty until the scientific committee approve one of the supervisors proposals.2. If the thesis is already assigned to one of the students, it will not be on the available list anymore.

Table A.2: Submit project proposal

Use Case	Submit project proposal
Actors	Supervisor
Goal	This use case allows the supervisor to submit a new proposal
Scenario	<ol style="list-style-type: none"> 1. The supervisor have the "Submit Proposal" option in his side menu bar. He just have to provide the necessary informations for his new proposal
Exception:	<ol style="list-style-type: none"> 1. Only supervisors have the option to submit a new proposal.

Table A.3: Manage proposals list

Use Case	Manage proposals list
Actors	Supervisor
Goal	This use case allows the supervisor to manage his own proposal list
Scenario	<ol style="list-style-type: none"> 1. The supervisor have the "My Proposal list" option in his side menu bar. He can reformulate his proposal or delete it.
Exception:	<ol style="list-style-type: none"> 1. Every supervisor can only manage his own proposals.

Table A.4: Consult all proposals list

Use Case	Consult all proposals list
Actors	Supervisor
Goal	This use case allows the supervisor to consult all the proposal list
Scenario	<ol style="list-style-type: none"> 1. The supervisor have the "All Proposal list" option in his side menu bar to consult the proposals submitted by his colleagues.
Exception:	<ol style="list-style-type: none"> 1. The supervisor must be logged in to consult all the proposals list

Table A.5: Analyse proposal

Use Case	Analyse proposal
Actors	Scientific committee member
Goal	This use case allows the scientific committee member to analyse the proposals submitted by the supervisors
Scenario	<ol style="list-style-type: none"> 1. After consulting all the proposals list submitted by the supervisors, the scientific committee member have the authority to analyse every proposals by clicking on "Analyse" 2. After Analysing the proposal, the scientific committee member can either accept the proposal or ask for reformulation if the analysed proposal needs some changes.
Exception:	<ol style="list-style-type: none"> 1. The scientific committee member must be logged in to analyse the proposals