

Course Management Support Application-iMaster.Report

Bechir Belhaj - a42707

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.Report

Bechir Belhaj - a42707

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

To my dear parents for their considerable support, to my brothers, to all my family, to my colleagues, comrades and friends, to all the people I love, I say thank you and I dedicate this work to them which is the culmination of several years of work..

Acknowledgment

I would like to thank all the people who contributed to the success of my final project and who helped me in writing this report. I would like to thank my academic supervisor Professor José Eduardo Fernandes for guiding me during this project with his valuable advice and sharing of his expertise. I express my sincere gratitude to my technical supervisor, Professor Paulo Alves, for his remarks, his invaluable advice, his kindness, his availability and above all for his confidence. Many thanks to my family and friends who supported me throughout this project. Finally, I thank all the members of the jury for the honor they have given us to participate in the review of our work.

Abstract

Nowadays databases are becoming bigger and bigger and the number of data records increases after the strong growth of investments in information and communication technologies, this is the case of the IPB. That is the reason why this platform is created. Our main goal was to simplify interpretation data through out academies and schools. The project's goal is to design and implement Dashboard called Imaster.reporting which made him able to visualize informations about the master and allowed the director to generate reports. For the realization of this module we used the Frameworks ReactJS and .net Core. The solution also allows the user to make reports instantly with multiple extension PDF and CSV.

Keywords: Imaster.reporting, Dashboard, ReactJS, .net Core.

Resumo

Hoje em dia, as bases de dados estão se tornando cada vez maiores e o número de registros de dados aumenta após o forte crescimento dos investimentos nas tecnologias de informação e comunicação. É por isso que se criou esta plataforma. O principal objetivo era simplificar a interpretação de dados em nossas academias e escolas. O objetivo do projeto é criar e implementar um painel chamado Imaster.reporting que o tornou capaz de visualizar informações sobre o curso de mestrado e permitiu ao diretor gerar relatórios. Para a realização deste módulo utilizamos os Frameworks ReactJS e .net Core. A solução também permite que o utilizador faça relatórios instantaneamente com múltiplas extensões PDF e CSV.

Palavras-chave: Imaster.reporting, Dashboard, ReactJS, .net Core.

Contents

1	Introduction	1
2	State of the Art	4
2.1	Comparative analysis	4
2.1.1	Infocursos	4
2.1.2	Idashboards	5
2.2	Proposed solution	6
3	Requirement Analysis and specification	9
3.1	Analysis and specification of needs	9
3.1.1	Non-functional requirements	9
3.1.2	Functional requirements	10
3.2	Global Conception	11
3.2.1	Actors identification	11
3.2.2	General Use case diagram	11
3.3	Detailed Conception	12
3.3.1	Use case « Authentication »	12
3.3.2	Use case «Show Dashboard»	14
3.3.3	Use Case "Generate report"	16
3.3.4	Use Case "Manage reports"	17
3.3.5	Generate Custom report	19
3.4	The web site map	20

3.4.1	The Director site structure	20
3.4.2	The dashboards site structure	21
3.4.3	The reports history interface structure	21
3.4.4	The generate reports interface structure	22
3.5	Database Schema	22
4	Technologies and Development Tools	25
4.1	Front-end frameworks	25
4.1.1	Angular	26
4.1.2	Vue	26
4.1.3	React	26
4.2	Back-end frameworks	27
4.2.1	Ruby on Rails	28
4.2.2	Laravel	28
4.2.3	Django	29
4.2.4	ASP.NET Core	29
4.3	Technologies choices	29
4.3.1	Front-end choice	30
4.3.2	Back-end choice	30
4.3.3	Other technologies	30
4.3.3.1	Microservices	31
4.3.3.2	Entity framework	31
4.3.3.3	StarUML	32
4.3.3.4	ASP.NET Core Identity	32
4.4	General Structure	32
4.4.1	MVC architecture	32
5	Development of iMaster Reports platform	35
5.1	iMaster Reports Implementation	35
5.1.1	Authentication	35

5.1.2	Dashboard visualization	36
5.1.2.1	General Dashboard	36
5.1.2.2	Students Dashboard	37
5.1.2.3	Professors Dashboard	39
5.1.2.4	Thesis/Project/Internship Dashboard	40
5.1.3	Generate Special Report	41
5.1.4	General PDF Report	42
5.1.5	Students PDF Report	43
5.1.6	Professors PDF Report	44
5.1.7	Thesis/Project/Internship PDF Report	45
5.1.8	Special Report PDF	47
6	Conclusions	49

List of Tables

3.1	Use Case «Authentication» details	13
3.2	Use Case «Show Dashboards» details	14
3.3	Use Case «Generate report » details	16
3.4	Use Case «Manage reports» details	18
3.5	Use Case «Generate Custom report » details	19

List of Figures

2.1	Infocursos Home Page	5
2.2	Idashboard charts	6
3.1	General use case diagram	12
3.2	Sequence Diagram «Authentication»	13
3.3	Show Dashboards Sequence Diagram	15
3.4	Generate report Sequence Diagram	17
3.5	Manage reports Sequence Diagram	18
3.6	Generate Custom report Sequence Diagram	20
3.7	The General site structure	20
3.8	The dashboards site structure	21
3.9	The reports history site structure	21
3.10	Graphical generate reports interfaces structure	22
3.11	Data Base Schema	23
4.1	Front-End Frameworks downloads in the past two years	25
4.2	ReactJS Architecture	27
4.3	Back-End Frameworks downloads in the past two years	28
4.4	microservices architecture	31
4.5	MVC Model	33
5.1	Login window screenshot	36
5.2	General Dashboard view screenshot	37

5.3	Student Dashboard view screenshot	38
5.4	Student Dashboard view screenshot	38
5.5	Professors Dashboard view screenshot	39
5.6	Thesis/Project/Internship Dashboard view screenshot	40
5.7	Thesis/Project/Internship Dashboard view screenshot	41
5.8	Special Report view screenshot	42
5.9	General PDF example	43
5.10	Students PDF example	44
5.11	Professors PDF Report	45
5.12	Thesis/Project/Internship PDF Report	46
5.13	Thesis/Project/Internship PDF Report	47

Acronyms

API Application Programming Interface.

DB Data Base.

DOM Document Object Model.

ESTiG Escola Superior de Tecnologia e Gestão.

HTTP HyperText Transfer Protocol.

IPB Instituto Politécnico de Bragança.

MVC Model View Controller.

SQL Structured Query Language.

UML Unified Modeling Language.

Chapter 1

Introduction

The Polytechnic Institute of Bragança (IPB) is a public higher education institution whose mission is to create, transmit and disseminate technical and scientific knowledge and professional knowledge, through the articulation of study, teaching, oriented research and experimental development. It was founded in in 1983 and its current president is Orlando Isidoro Afonso Rodrigues, it is made up of 5 schools which makes the diversity of the courses that IPB offers for 9000 students who are from several countries of the world because of mobility programs and collaboration with other Erasmus educational establishments.

Nowadays databases are becoming bigger and bigger and the number of data records increases after the strong growth of investments in information and communication technologies.

This is where human has to invent the graphical dashboard, a dashboard is a type of graphical interfaces linked with the database which displays all the data useful for the user and allows the report to be constantly updated. So the directors of IPB masters courses need to visualize the progress of their work whether they are performing well or not during the entire period of their feed backs mandate, they need to know all the information about their students, professors and also thesis and projects made by the students for each semester and also they need this information to present them in meetings which

is for this moment it take long time to have this information because they must have it manually from the database. So this is why the Polytechnic Institute of Bragança started with the development of a platform called Imaster which contains a dashboard called Imaster.Reports. The goal of Imaster.Reports is to facilitate the interpretation of all the data from the IPB database which can be useful and display them in a dashboards also offer the possibility of making reports instantly with multiple extension. The main user of this application is the director of the masters to report on their work as director and have feedbacks of their works. Imaster.Report is also integral with all the other modules of the Imaster platform like Imaster.Calendar and Imaster.Process.

First chapter The first chapter will describe the motivation to develop this project and the main goals.

Second chapter The second chapter will describe the state of the art.

Third chapter The third chapter will describe the Requirement specification and analysis

The Fourth chapter The fourth chapter will describe the technologies and development tools

Fifth chapter The fifth chapter will describe the development of iMaster Reports platform

Finally the report is going to be closed through a general conclusion allowing to synthesize all the work carried out while evoking the future perspectives of the project.

Chapter 2

State of the Art

This chapter presents the general context, it is a really important step for the knowledge of the environment in which the work took place. First a comparative study will be made likewise present the solution for the propose.

2.1 Comparative analysis

The study of an existing solution will allow us to deduce a diagnosis that will help us in the development of the own solution. The final goal is to develop a dashboard that provides the most useful information for the director of the master. For this Infocursos and Idashboards are chosen.

2.1.1 Infocursos

Infocursos is a national Portuguese website that provide data and statistics on Higher Professional Technician (Te-SP), 1st Cycle License, Integrated Master and 2nd Cycle Master courses taught in Portuguese higher education institutions[1].

We will present the Infocursos Home Page in figure 2.1

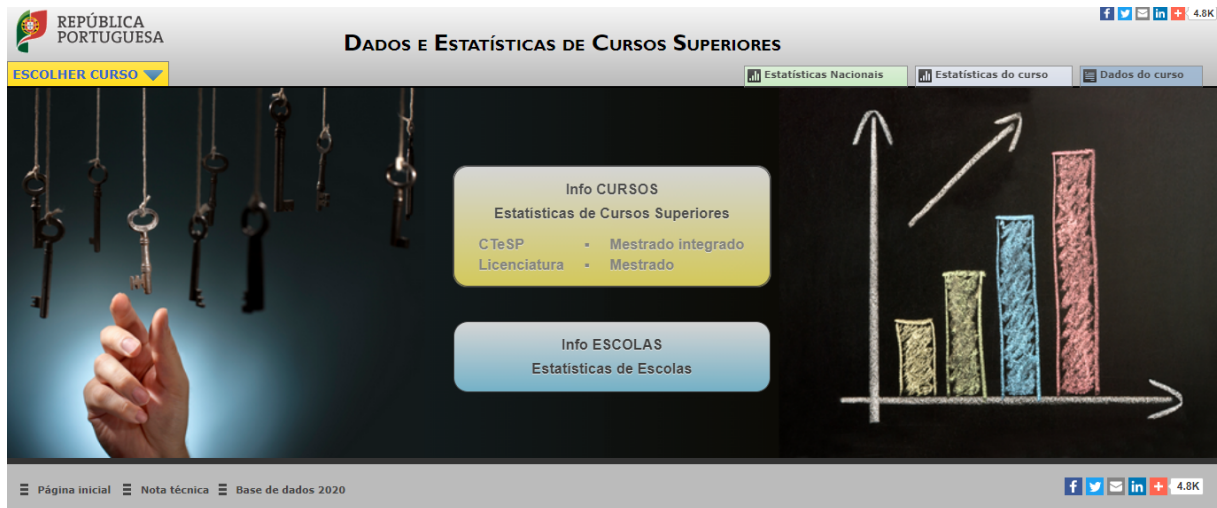


Figure 2.1: Infocursos Home Page

Advantage:

- Infocursos provides information on all Portuguese universities .
- Infocursos provides information on all university cycles .
- Infocursos is free web site .

Disadvantages:

- Infocursos is accessible For everyone who wants to see the informations
- The information available in Infocursos is just for the current year
- Infocursos do not provide much informations on masters
- Infocursos do not generate reports for displayed informations
- Infocursos has a bad designs

2.1.2 Idashboards

IDashboards software is a self-service dashboard solution that allows you to connect disparate data sources to create and visualize personalized dashboards then generate dashboards for higher education[2]. We will present the Idashboard Dashboard in figure 2.2

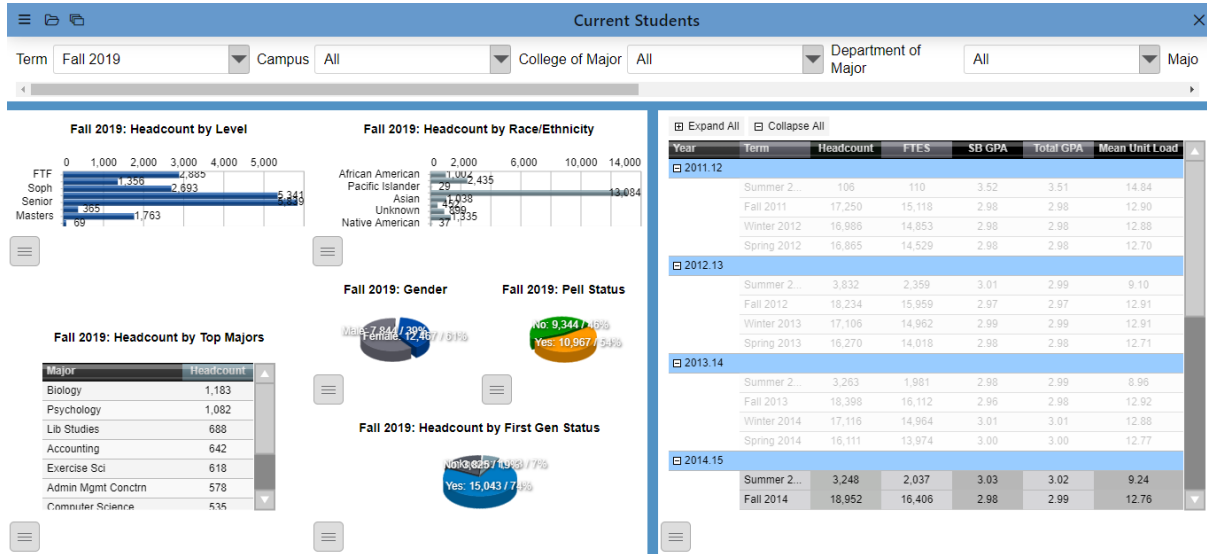


Figure 2.2: Idashboard charts

Advantage:

- Idashboards provides information for all the previous years.
- Idashboards provides many types of charts.
- Idashboards generate reports for each chart.

Disadvantages:

- Idashboards is not for free.
- Idashboard contain a lot of not useful Data.

2.2 Proposed solution

For the solution a general dashboard that contains general informations will be implemented. Three other specified dashboards that contain more detailed informations than the General dashboard. One dashboard is dedicated for students' informations of this

master, the second is dedicated for professors and the third contains Thesis/Project/Internship informations.

For each dashboard the director of the master can generate a report that contain the information of that dashboard and he can also generate a custom report that contain combined information that he need from all the dashboards.

Conclusion

This chapter, have studied the different existing solutions to identify their limits and propose the suitable solution for the case. The next chapter will be dedicated for Requirement analysis and specification.

Chapter 3

Requirement Analysis and specification

This chapter will present all the functional and non-functional requirements needed to develop the application then the global conception, the detailed conception and the web site map and the database schema.

3.1 Analysis and specification of needs

Functional and non-functional requirements will be the subject of this part. It will allow us to understand the expectations of the host organization. The requirement analysis and specification phase is the first formal phase in the development of a web application because the persuasiveness of a product cannot be achieved perfectly without prior specification developed needs and requirements.

3.1.1 Non-functional requirements

Non-functional requirements present internal requirements for the system and hidden from the users. The most immediate requirements of the application are:

- **Ergonomic** : From an ergonomic point of view, the application must be compatible with any operation while being easy to handle. These interfaces must be understandable, and well organized.
- **Extensible**: The system must allow the integration of new functionalities. used to develop Imaster.Report, having several advantages besides the possibility of rendering modular the project maintenance and development of the project in the future.
- **Data security**: Securing data amounts to applying an identification strategy, authentication, authorization and control of each attempt to access this data. In our system access to personal information is only allowed to owners and according to a privilege which determines the access rights. As with authentication, the access to data resources is controlled by OAUTH2. The system uses the access token obtained from the following user authentication for any communication with the data server which makes the exchanges very secure.

3.1.2 Functional requirements

The functional requirements represent the expectations of each actor of the application to be developed. Any conceptual solution must first satisfies functional requirements in order to delimit the functional scope of the application and monitor the traceability of requirements during the development phase.

- **Authentication** : Imaster.Report has a OAUTH2 type authentication. oAuth2 (version 2) is a protocol that allows third-party applications to gain limited access to a service available over HTTP with prior authorization from the resource owner. Access is requested by what is called a "customer", which can be a website or a mobile application for example. If the resources are not owned by the client, then the client must obtain authorization from the end user, otherwise they can directly gain access by authenticating with their own credentials. So all exchanges with the resource server must go through the authorization server

- **Dashboard visualization** : The director will have the possibility of viewing several dashboards (General dashboard, Students dashboard, Professors dashboard, Thesis/Project/Internship dashboard).
- **Report generating**: The director will have the possibility of generating a report for each dashboard (General dashboard, Students dashboard, Professors dashboard, Thesis/Project/Internship dashboard)
- **Custom Report generating** : The manager will have the possibility to generate a report with the information that chooses them from all available dashboards.

3.2 Global Conception

A use case diagram captures the behavior of a system, a subsystem, an class or component as an outside user sees it. It divides the functionality of the system into coherent units, use cases. All this shows that this step is important to produce software that meets users expectations.

3.2.1 Actors identification

We precede by determining the actors and the use cases: An actor is any entity that interacts with the system in order to achieve added value and which always has the same behavior. A use case is the description of a set of sequences of operations that a system performs to meet the need of an actor. The use case diagram model as well as the interactions between the components of the system are important to identify the main feature of the system. The only user was identify is the master director.

3.2.2 General Use case diagram

The general use case diagram is described by following figure:

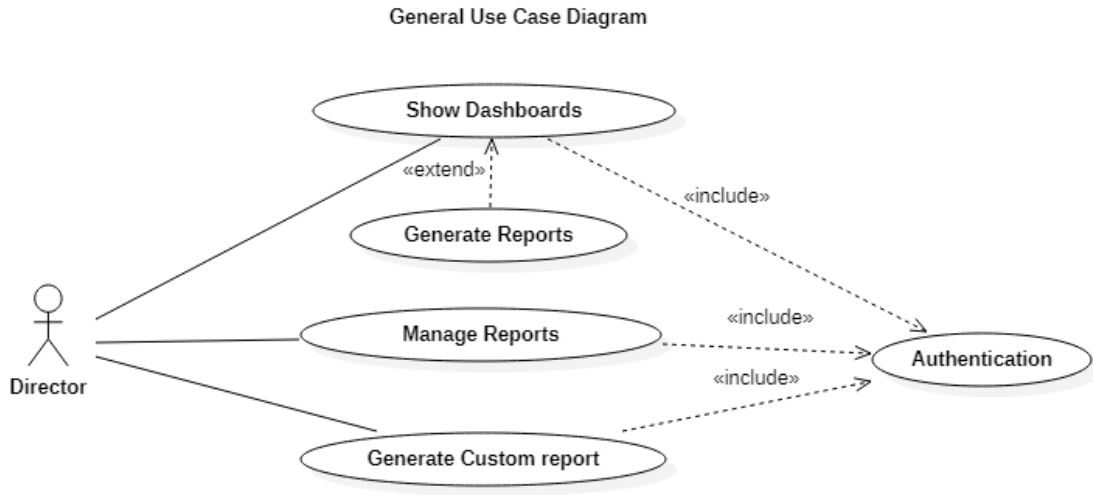


Figure 3.1: General use case diagram

The general use case diagram indicated that the director can visualize the dashboards and after that he have the option of generating reports, he can also manage the reports and he can generate custom reports but before that he have the access for this functions he have to authenticate.

After presenting the General use case diagram, each use case will be refined with some informations and sequence diagrams to understand more the process of each use case.

3.3 Detailed Conception

This part will represent all the detailed use cases and explain each one better by a Sequence Diagram.

3.3.1 Use case « Authentication »

The textual description of the “Authentication” use case is illustrated by the following table:

Use Case	Authentication
Actors	Director
Goal	This use case allows the Director to access his space
Scenario	<ol style="list-style-type: none"> 1. The Directors enters their connection parameters 2. The Director clicks on the "Connect" button
Exception:	If the username or password are wrong, the system displays an error message and the system redirects it to the start of the main scenario

Table 3.1: Use Case «Authentication» details

The system sequence diagram of the "Authentication " use case is described by following figure:

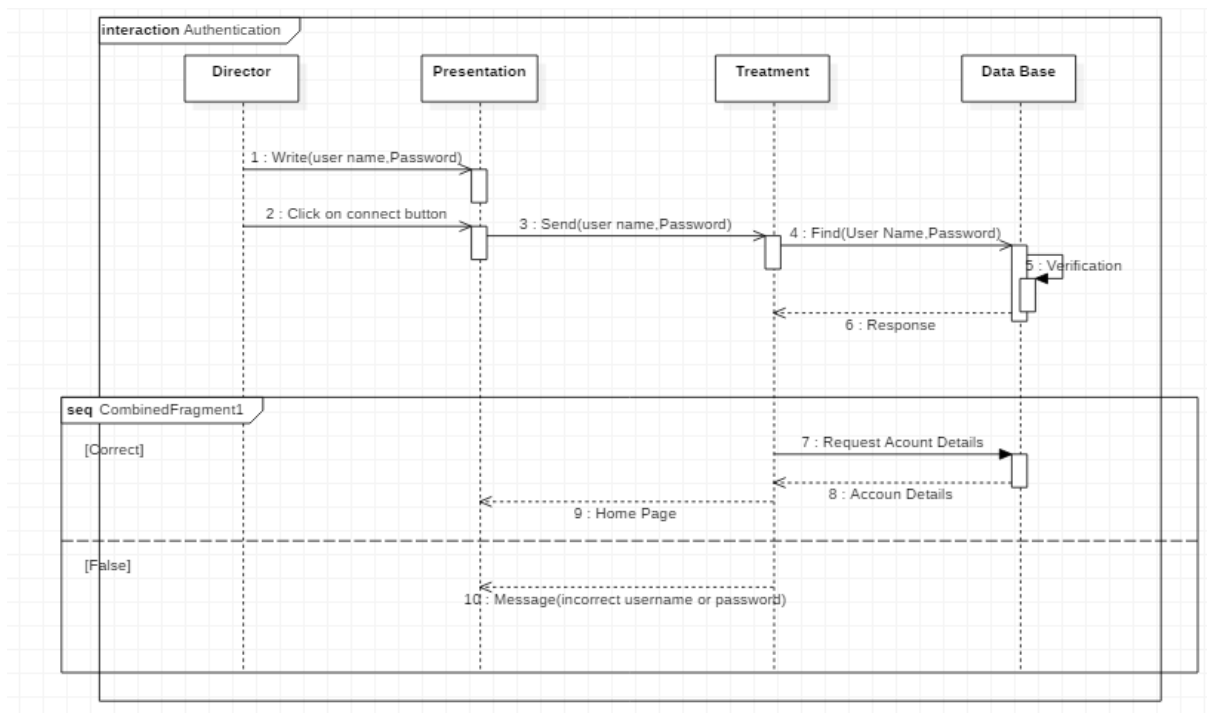


Figure 3.2: Sequence Diagram «Authentication»

By authenticating to the system the user accesses his account via the password and the

username. This data has already been recorded in the table titled user and that is why the controller performs a read operation on this table to find it for the user in question before allowing access or denying it.

3.3.2 Use case «Show Dashboard»

The textual description of the “Show Dashboards” use case is illustrated by the following table:

Use Case	Show Dashboard
Actors	Director
Goal	This use case allows the Director to view all dashboards (General Dashboard, Student Dashboard, Professors Dashboard and Thesis/Project/Internship Dashboard)
Preconditions	The Director Must be connected
Scenario	<ol style="list-style-type: none"> 1. The Director click on "Menu" Button. 2. The System show the dashboards list. 3. The Director click "Dashboard" Button. 4. The system display the specific dashboard.
Exception:	If the user is not connect, the system denied the access to the resource

Table 3.2: Use Case «Show Dashboards» details

The "Show Dashboards" use case is for the director of the master to allow him to view all types of dashboards (General Dashboard, Student Dashboard, Professors Dashboard and Thesis/Project/Internship Dashboard) but to do that he must be authenticated to the system because the system allow only the director of the master to display the information of his master, if the user is not authenticated or don't have the role of the master he will not get the access to that resource.

this use case, he can choose between many Dashboards (General Dashboard, Students dashboards, Professors Dashboards, Thesis Dashboards) But before that he have to authenticate to the system.

3.3.3 Use Case "Generate report"

The textual description of the “Generate report” use case is illustrated by the following table:

Use Case	Generate report
Actors	Director
Goal	This use case allows the Director to generate a report for each type of Dashboard
Preconditions	<ul style="list-style-type: none"> • The Director Must be connected. • The Director must be in the Dashboard Page.
Scenario	<ol style="list-style-type: none"> 1. The Director Choose a Semester. 2. The Director click on "PDF report ". 3. The system save the report in the database. 4. The system generate the report in PDF format.
Exception:	If the user is not connect, the system denied the access to the resource

Table 3.3: Use Case «Generate report » details

The "Generate report" is for the user "Director" to allow him generating reports for each type of dashboard in PDF format but to do that the director of the master need to be in the dashboard page and to do that he need to be authenticated first.

The sequence diagram of "Generate report" use case is described by following figure:

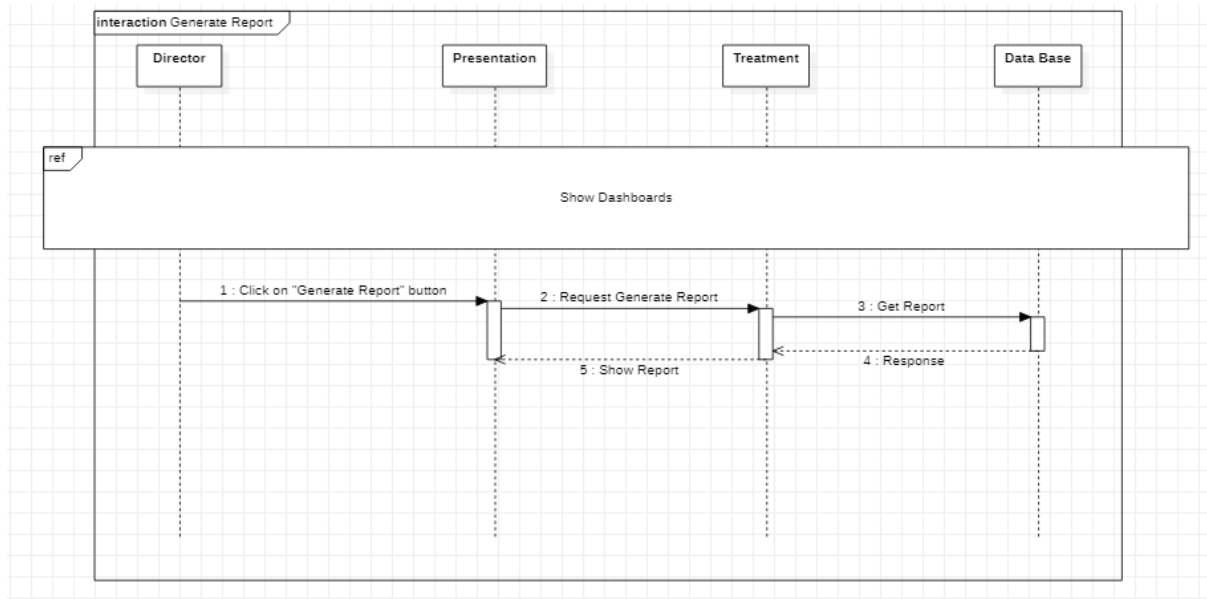


Figure 3.4: Generate report Sequence Diagram

3.3.4 Use Case "Manage reports"

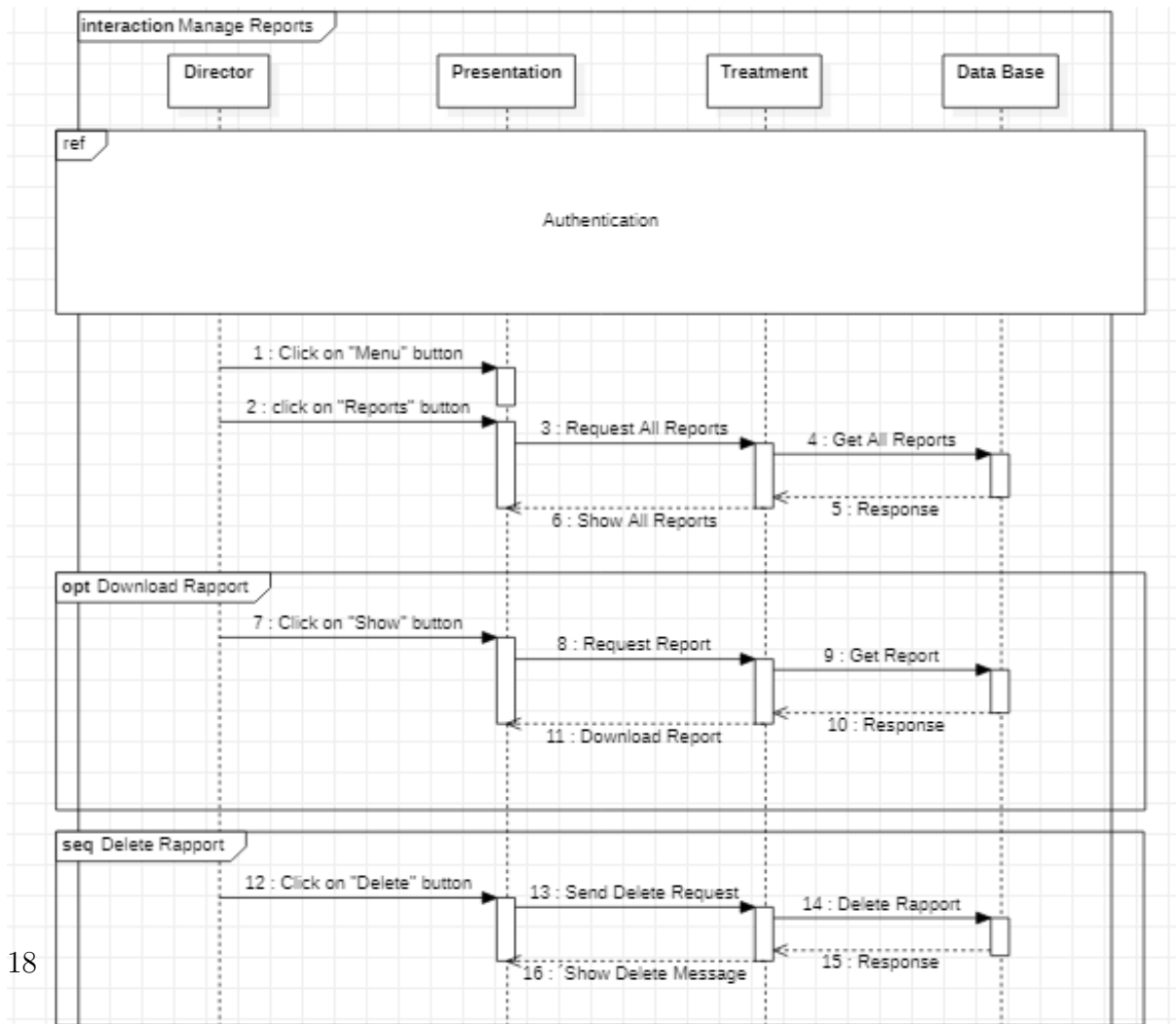
The textual description of the “Manage reports” use case is illustrated by the following table:

The Director of the master can download reports or delete them from the database.

The system sequence diagram of "Manage reports" use case is presented by following figure:

Use Case	Manage reports
Actors	Director
Goal	This use case allows the Director to Download or Delete his previous reports from database.
Preconditions	The Director Must be connected.
Scenario	<ol style="list-style-type: none"> 1. The Director click open "Menu". 2. The Director click on "report History" . 3. The System Display all the previous reports. 4. The director can Delete or show again any report from the displayed list.
Exception:	If the user is not connect, the system denied the access to the resource

Table 3.4: Use Case «Manage reports» details



3.3.5 Generate Custom report

The "Generate Custom report" use Case is to allow the director generating custom reports PDF format that contains only the information that he want to display in his report from all dashboards available,the Director can choose only 4 options from each Dashboard.

Use Case	Generate Custom report
Actors	Director
Goal	This use case allows the Director to generate a custom report with the information that he need from all the dashboards.
Preconditions	The Director Must be connected.
Scenario	<ol style="list-style-type: none">1. The Director open the "Menu".2. The Director click on "Generate Custom report".3. The Director Must fill out the form.4. The Director click on "generate report" .5. The System display the custom report.
Exception:	If the user is not connect, the system denied the access to the resource

Table 3.5: Use Case «Generate Custom report » details

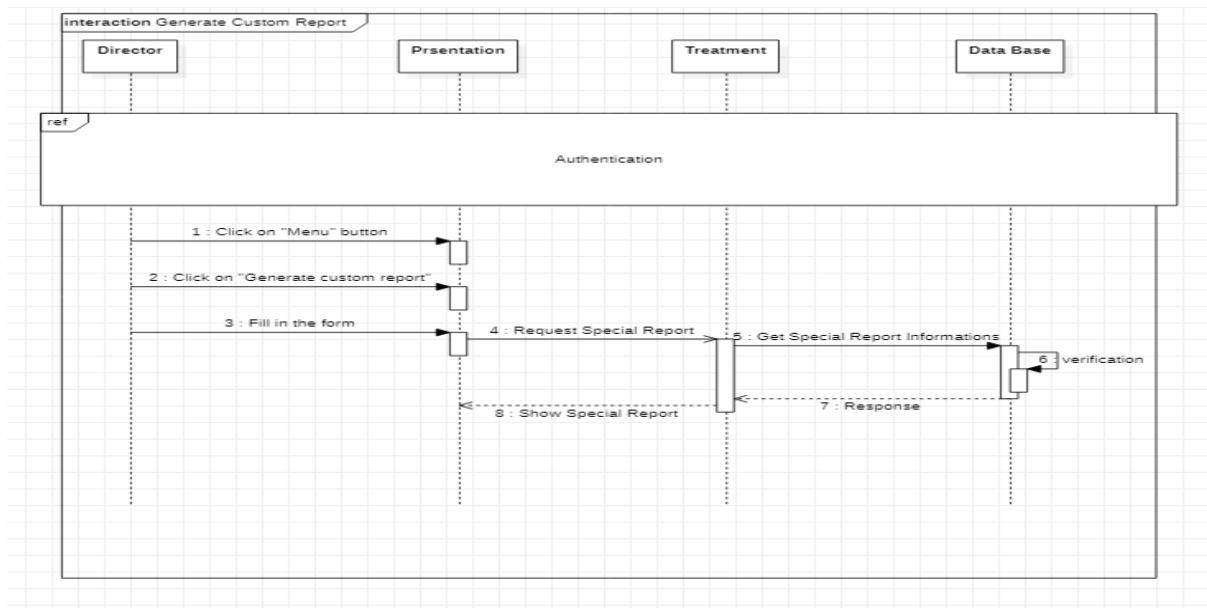


Figure 3.6: Generate Custom report Sequence Diagram

3.4 The web site map

This section will present the general site structure then it will be detailed.

3.4.1 The Director site structure

The Graphical Director interface structure is described by following figure:



Figure 3.7: The General site structure

As the general hierarchy graph of the proposed solution presents to us, the director must

authenticate this is the first level of hierarchy the second level after authentication he can view the dashboards, reports history and generate a special report and for the third level generating dashboard reports.

3.4.2 The dashboards site structure

The Dashboards hierarchy is described by following figure:

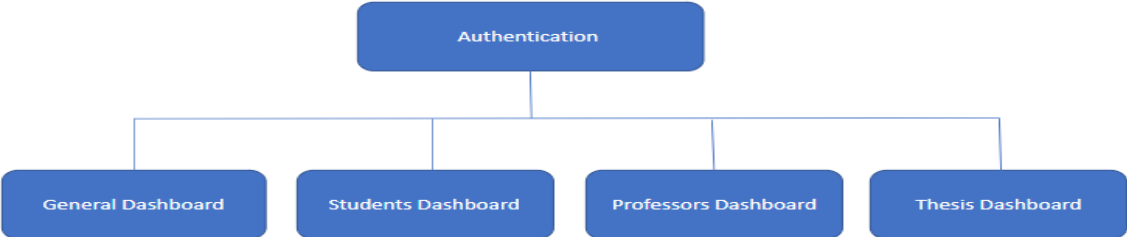


Figure 3.8: The dashboards site structure

As the Dashboards hierarchy graph presents the dashboards that are divided in general dashboard, students dashboards, professors dashboards and thesis dashboards.

3.4.3 The reports history interface structure

The Reports hierarchy is described by following figure:

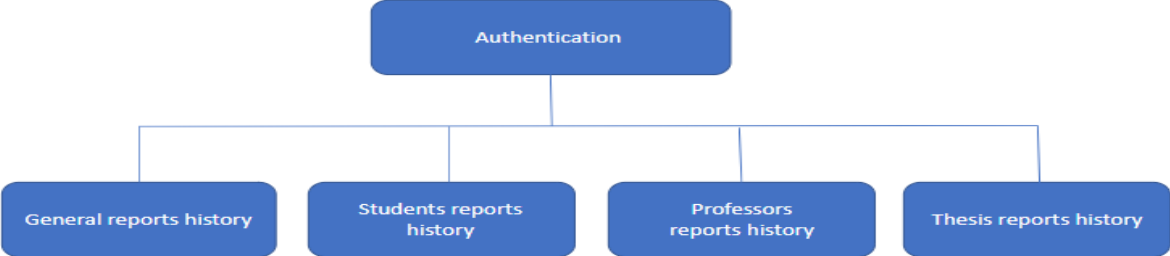


Figure 3.9: The reports history site structure

As the Reports History hierarchy graph presents to us, the Reports History are divided in general Reports History, students Reports History, professors Reports History and thesis

Reports History.

3.4.4 The generate reports interface structure

The Generate Reports hierarchy is described by following figure:

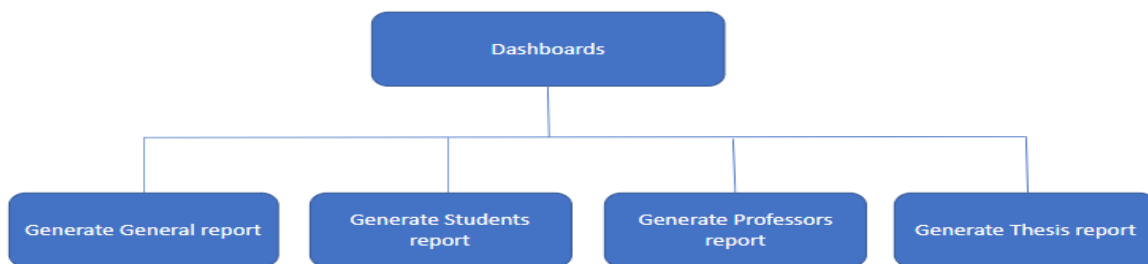


Figure 3.10: Graphical generate reports interfaces structure

As the Generate Reports hierarchy graph presents to us, the Generate Reports is divided in generate general Reports, generate students Reports, generate professors Reports and generate thesis Reports.

3.5 Database Schema

This figure will present the database schema for the tables that the database contains, each table of the database is explained.

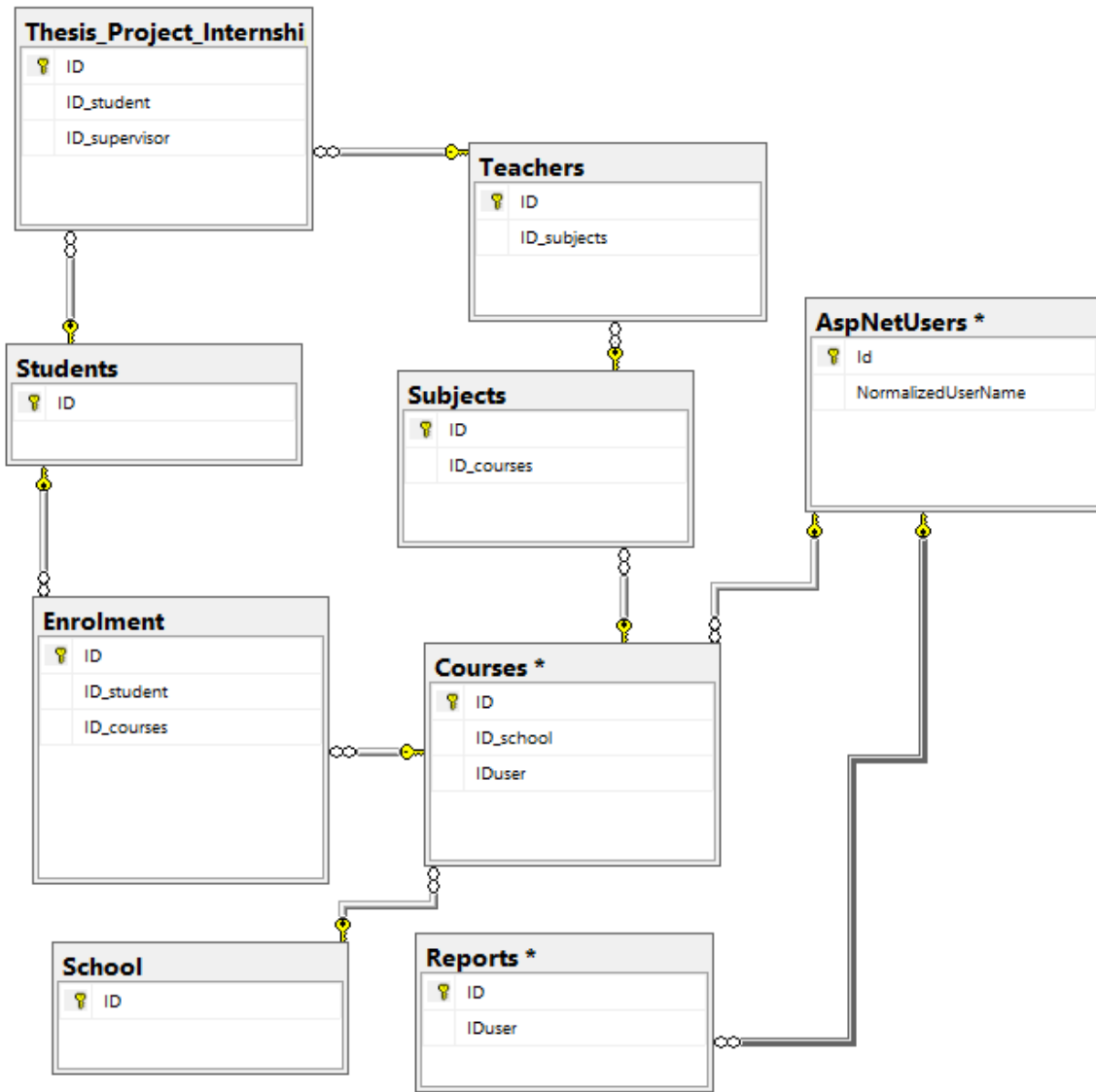


Figure 3.11: Data Base Schema

- School : Represents the IPB Schools list.
- Courses : Represents the Courses of each school.
- Enrolment : Represents the Student Enrolment.
- Enrolment : Represents the Student Enrolment in the courses .
- Student : Represents the Student informations.

- Teachers : Represents the Professors informations.
- Reports : Represents the Reports generated informations.
- Thesis Project Internship : Represents the informations of all the thesis,projects and Internships.

Conclusion

This chapter have present the analysis and specification of platform, the Non-functional requirements and the Functional requirements. Then a global conception and a detailed conception was made after the web site map and the database schema.

The next chapter will be dedicated for the Technologies and Development Tools.

Chapter 4

Technologies and Development Tools

This chapter presents the Technologies and Development Tools also the Front-end and the Back-end frameworks then the adequate technologies to use.

4.1 Front-end frameworks

Consistently, the front-end structures improve and add new functionalities to discover a spot inside the yearly front-end challenges. To identify the main front-end structures and libraries in 2020, GitHub is a reference since it is the largest Git-repository hosting service globally. We will present the Frameworks downloads diagram in figure 2.3 [3]

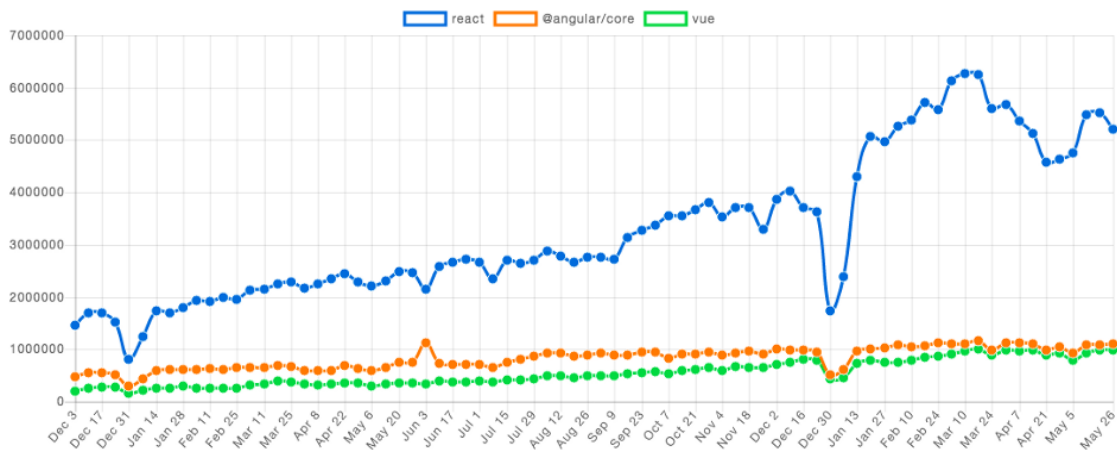


Figure 4.1: Front-End Frameworks downloads in the past two years

4.1.1 Angular

Angular is a known JavaScript framework and an open source developed by Google in 2010. The center idea of Angular depends on two-way information official in internet browsers to decrease the back-end's information handling obligation in web workers. The first goal of Angular is to help web designer in creating a persistent and efficient web forms.[4]

4.1.2 Vue

Vue is a scalable framework for building user interfaces. Unlike other monolithic frameworks, Vue has been designed and engineered to be incrementally adoptable. The core of the library is focused only on the view part, and it's really easy to integrate with other existing libraries or projects. On the other hand, Vue is quite capable of running single-page web applications when coupled with modern tools and complementary libraries. We will present the Vue.js data-driven concept

- 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.

What's more, that is the way Vue can give both two-way and single direction restricting alternative in the information cycle. Furthermore, contrasted with React and Angular, Vue recognized by its negligible volume and effective delivering and preparing. In spite of the fact that its front-end prevalence, Vue's specialized help isn't solid because of its restricted group scale improvement with the surprising refreshed plans. Its little volume implies that it incorporates less highlights than different systems.[5]

4.1.3 React

React (also called ReactJS or React.JS) is an open source, front-end JavaScript library developed by Facebook in 2013 it is used for handling the view layer for web or mobile

applications. The main goal of this library is to facilitate the creation of a single page web application and to build a fast, scalable and fluid website. The specific use of the ReactJS library specializes in helping to build user interfaces, or UIs. In terms of websites and web applications, UIs are the collection of on-screen menus, search bars, buttons, and anything else someone interacts with to use a website or app.

Many web applications was developed with ReactJs like Instagram, Netflix, Whats App, New York Times and many other famous applications. We will present the ReactJS architecture in figure 2.4 [6]

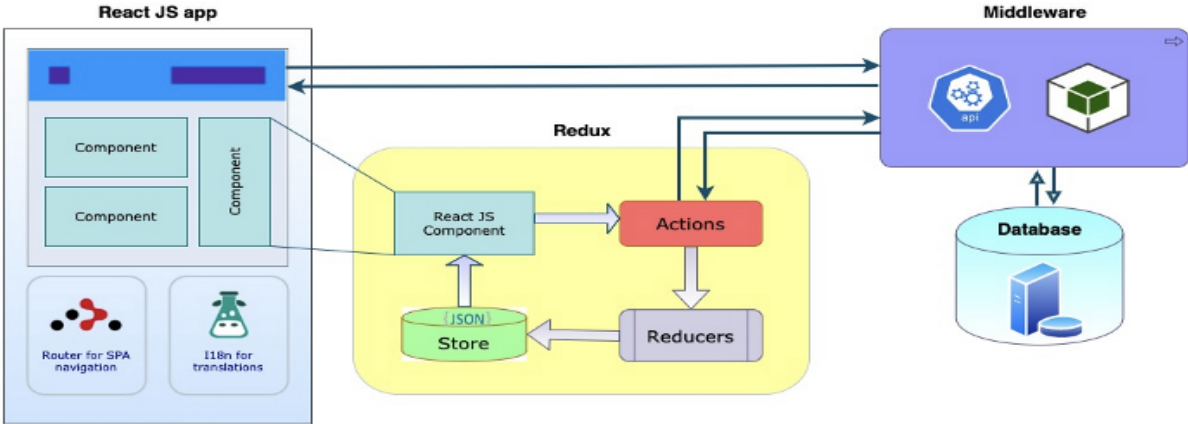


Figure 4.2: ReactJS Architecture

4.2 Back-end frameworks

Beside the front-end frameworks, Back-end frameworks also are improving by time. Through some researches about the well-known back-end frameworks lead to extract these data displayed in the below figure based on GitHub and Stack Overflow scores. To confirm the popularity of the ASP.NET through the time, figure 2.5 will present a graph of the web framework popularity. [7]

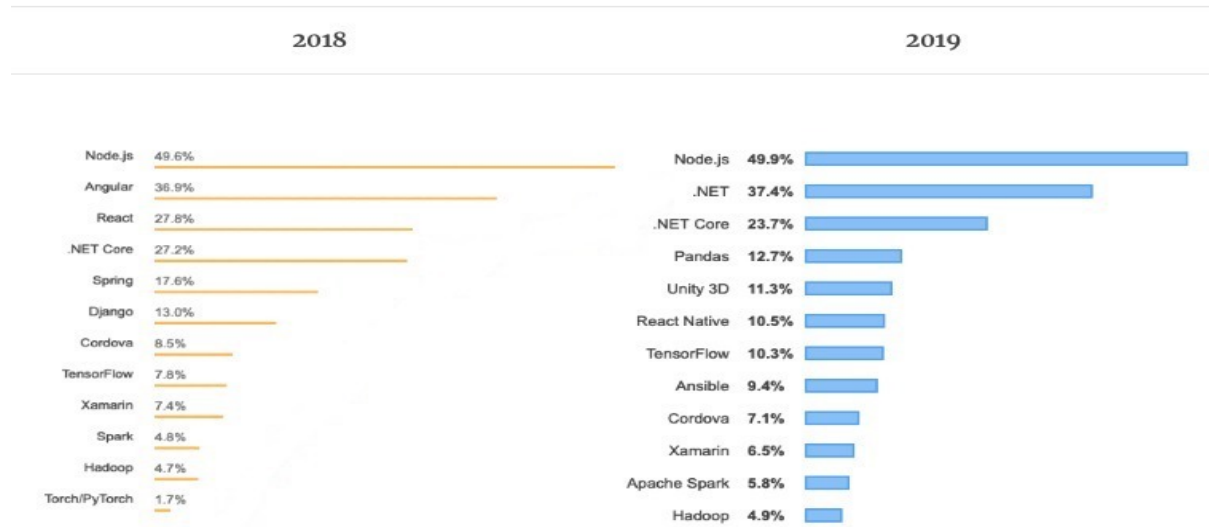


Figure 4.3: Back-End Frameworks downloads in the past two years

4.2.1 Ruby on Rails

Ruby on Rails, additionally called RoR or Rails, is a free web structure written in Ruby. It follows the model-view-controller(MVC) design pattern. It offers a structure that permits to develop an application rapidly and naturally. Nonetheless, it forces an extraordinary degree of reflection in the programming which acquires in return the economy of writing oneself most of the obligatory routines of a web application. Rails is based mainly on two principals[8]:

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

4.2.2 Laravel

Laravel is a web application framework released by Taylor Otwell based on MVC structure. It is based on Preprocessor Hypertext (PHP). Laravel is a structure that is straightforward, simple and well known. In relation to Laravel authentication offers a very easy implementation for developers, since Almost everything is out of the box and configured. The system offers consistent documentation in terms of documentation, so it helps

Developers to quickly learn and develop.[9]

4.2.3 Django

Django is an elevated level Python Web structure. It was made in 2003 when the developers working for the Lawrence Journal-World paper began utilizing Python to assemble applications. Furthermore, on 2005, the software engineers chose to deliver the system as an open source furthermore, it was named after the celebrated jazz guitarist Django Reinhardt. Django is a significant level Python Web structure. Django is a cross-stage supporting the MVC design engineering. The MVC programming worldview causes designers to isolate the both of business rationale and UI which make the web applications quicker and more simple. In addition, this structure gives a very much reported starter guides for amateurs, and since it is identified with Python, help is truly simple to get while creating web applications utilizing Django.[10]

4.2.4 ASP.NET Core

ASP.NET Core is a free, open-source web framework developed by the Microsoft community and designed to allow run time components, APIs compilers to evolve quickly, while still providing a stable platform to keep apps running. ASP.NET Core is more efficient than ASP.NET and works with both the .NET framework on windows and .NET core on the cross-platform it unites ASP.NET MVC and ASP.NET Web API in a single model of programming. For all these reasons the ASP.net core web was chosen to build the API web application with to ensure an efficient and secure back-end for the solution.[11]

4.3 Technologies choices

We need to choose two frameworks to build the web application. the first one for the front-end and the second one for the back-end. The front-end framework allows the programmer to build the interface of the web application, while the back-end framework is the core

of the web application, it includes all the methods built on the front-end to provide an API. In the next pages, the chosen frameworks and technologies for the application are addressed .

4.3.1 Front-end choice

For several factors, after a detailed analysis, ReactJs is chosen as a front-end platform for the program. The number one front-end system nowadays is Reactjs. The Simulated DOM is used by Reactjs in terms of tactics, which gives the ReactJs many strong reasons to be the number one. The virtual DOM doesn't store observables in the memory not like other structures. So, by maximizing the memory consumption, it provides the application. In addition, since the data binding is not entangled with the program, the virtual DOM enables the workflow for developers because the JavaScript code updates the React components while the virtual DOM is modified by React. The application is modified by Virtual DOM, which boosts performance and makes the application more efficient.

4.3.2 Back-end choice

ASP.NET is one of the most stable back-end systems, according to the statistics mentioned in the 2.5 segment. ASP.NET Core was chosen to build the back-end for the program. In particular, with its latest functionality and frequent updates, the ASP.NET platform is able to provide high-performance applications that really allow developers to provide optimized code. ASP.NET Core offers a lower version of code, so the necessary code is lower when the coding is lower and the time for coding would also be smaller, which makes ASP.NET Core even more time-efficient. Less coding, in reality, also implies simpler maintenance.

4.3.3 Other technologies

It is now time to identify the other methods and technology to be used in the application after choosing the front-end and back-end frameworks.

4.3.3.1 Microservices

Microservices are presented as a Microsoft-developed architecture pattern focused on the separation of programs into small independent modules communicating with each other. Unlike other architectures, decomposition makes it much simpler to deploy. In addition to facilitating the code base for the dev team, Microservices also makes it easy to verify and correct vulnerabilities without disrupting the other components. In addition, microservices allow developers to use multiple programming languages as long as the services connect with each other.

We will present the microservices architecture in figure 2.6 [12]

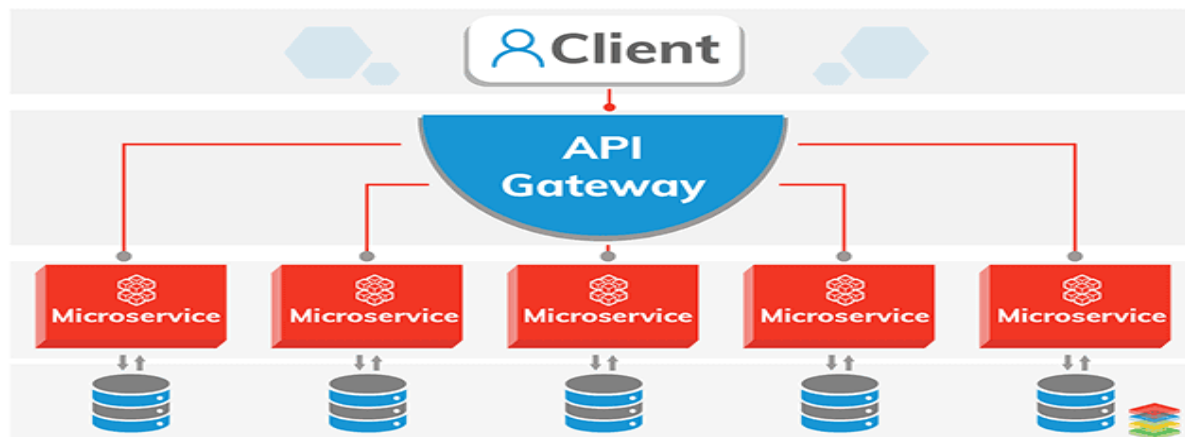


Figure 4.4: microservices architecture

4.3.3.2 Entity framework

Entity Framework is an open source framework for .net applications from Microsoft. It gives us access to work with data using domain-specific class objects without having to worry about database tables.

With Entity Framework, developers can work at a higher level of abstraction when they are dealing with data, and can create applications with less and more efficient code compared with traditional applications.[13]

4.3.3.3 StarUML

StarUML is also an open source tool that allows to quickly draw, import or export UML diagrams from a plain text language. It was used for the realization of the diagrams, the class diagram, use case diagrams, and finally sequence diagrams.

4.3.3.4 ASP.NET Core Identity

ASP.NET Core Identity is an API that supports user interface (UI) login functionality ,Manages users, passwords, profile data, roles, claims, tokens, email confirmation, and more.Identity is typically configured using a SQL Server database to store user names, passwords, and profile data. Alternatively, another persistent store can be used, for example, Azure Table Storage.[14]

4.4 General Structure

Our application is mainly based on two big parts. The first is concerned about the intern system of the application. The second is presenting the user part that contain the interfaces of the application. The system architecture choice will be justified in the next parts

4.4.1 MVC architecture

The MVC model "Model View Controller" which is characterized by a separation of logic three-part code that we find in separate files. This model brings several advantages for developers: Firstly, the ability to reuse the code in other similar applications. Secondly, there is time-saving for the implementation and the development of the applications and it help developers to collaborate and work together. Thirdly is the greater team integration and division of tasks and finally, the reduced complexity in the code.

The figure 4.1 will present the MVC Model: [15]

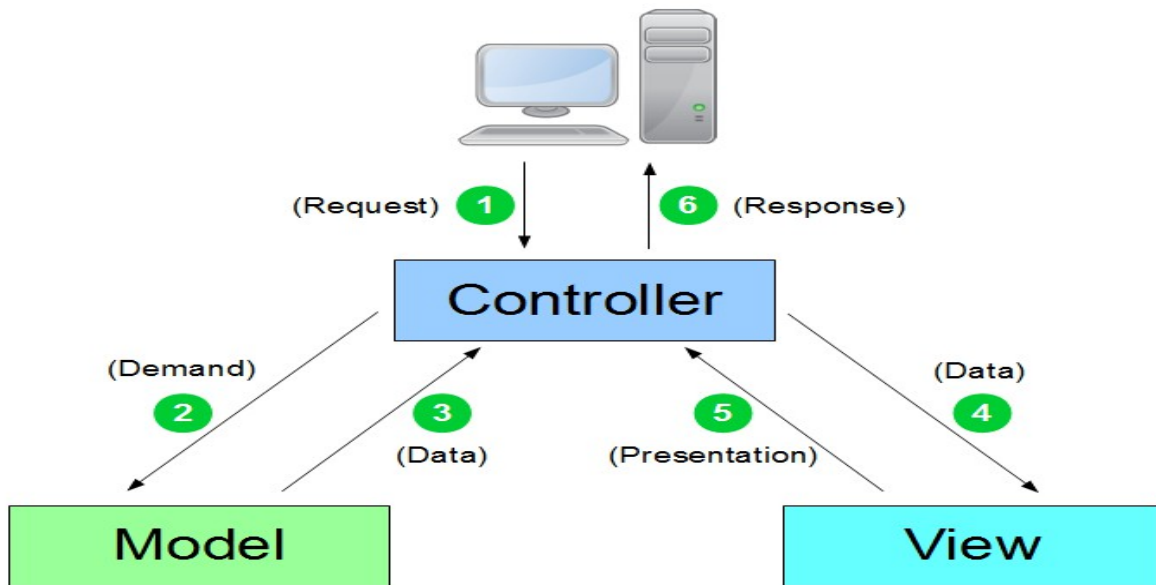


Figure 4.5: MVC Model

- **Model:** this layer manages the data of the application. It groups together functions for recover, inserting and updating information in the database.
- **View:** this is the user interface, it receives variables from the controller to know what to display.
- **Controller:** this is the layer that manages the logic of the code and makes decisions. This is a set of files that provides synchronization between views and models.

Conclusion

This chapter have present the Technologies and Development Tools, front-end frameworks and the back-end framework and the technologies choices.

The next chapter will be dedicated for the Development of iMaster Reports platform.

Chapter 5

Development of iMaster Reports platform

This chapter will focus on the implementation of the application's interfaces by presenting the Authentication interface, the dashboards visualisation interfaces, generate special report interface and some generated PDF examples .

5.1 iMaster Reports Implementation

The user interfaces are important to present the application in a clear way. This Part will present the implementation results with some screenshots and descriptions for the application Imaster.report.

5.1.1 Authentication

When the user launch the application, the first thing that he have to do is that he have to authenticate to the system to get the Token from the API to access to the home page, so the user will be redirected to the login window .

The login window is presented in the down figure :

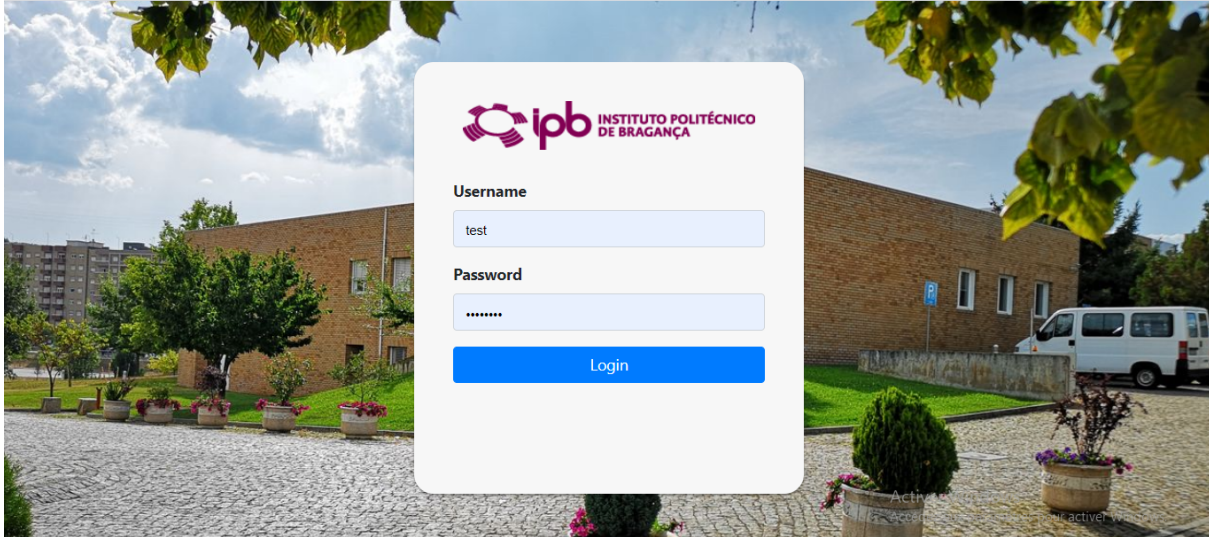


Figure 5.1: Login window screenshot

5.1.2 Dashboard visualization

This part will represent all the dashboards interfaces that was implement in the application that the Director can use (General dashboard, Student dashboard, Professors dashboard, Thesis/Project/Internship dashboard)

5.1.2.1 General Dashboard

The General Dashboard view represent general informations from the other dashboards (Student dashboard, professors dashboard, Thesis/Project/Internship dashboard) and compare these information with the information of the other semesters also he have the possibility to choose the semester that he want to show the informations also the director can generate PDF report or CSV reports .

The informations represented in the general dashboard view are :

- **Students Informations** : Total of students, Total of international students, Total of Portuguese Students, Total of Double Diploma Students, Total of Graduate Students .

- **Professors Informations** : Total of Professors, Total of International Professors, Total of Portuguese Professors, Total of Temporary Professors, Permanent Professors.
- **Thesis/Projects/Internships Informations** : Total of Thesis, Total Projects, Total Internships, Total of Internal Internships.

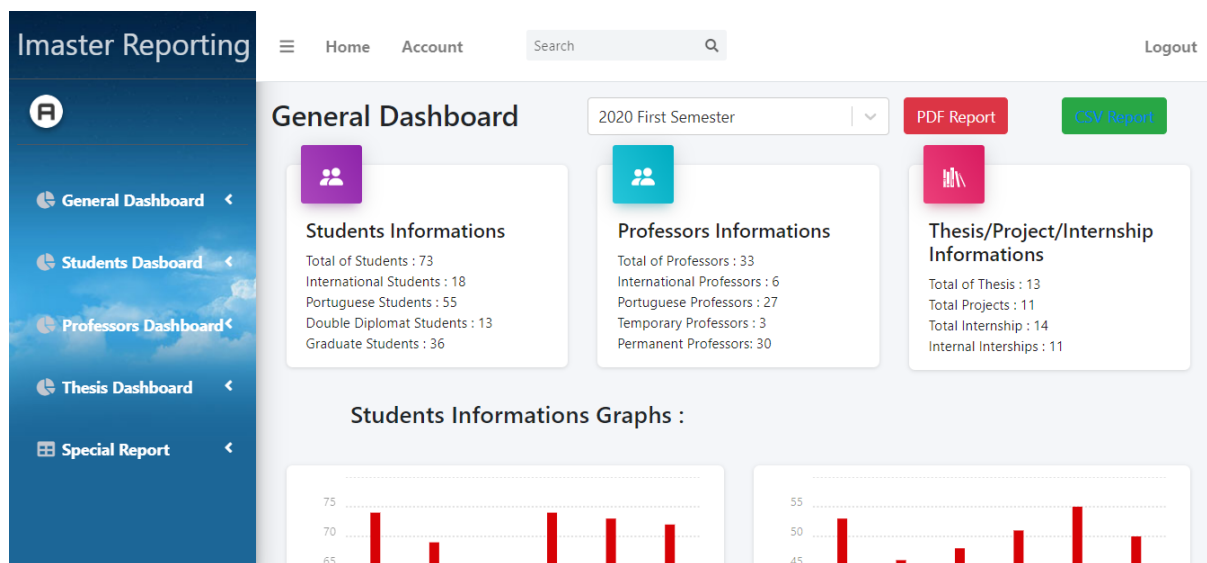


Figure 5.2: General Dashboard view screenshot

5.1.2.2 Students Dashboard

The Students Dashboard view represent students informations (Students By Gender, Students By Master Year, International Students By Nationality and Other Informations) and give the possibility to choose the semester that he want to show the informations and also the director can generate PDF report.

The informations represented in the Students Dashboard view are :

- **Students By Gender** : Total of Male Students, Total of Female Students.
- **Students By Master Year**: Students enrolled in the first year, Students enrolled in the second year.

- **International Students By Nationality:** Total of Tunisian Students, Total of Brazilian Students, Total of Spanish Students, Total of Cap Verde Students, and Total of French Students.
- **Other Informations:** Total of International Students, Total of Portuguese Students, Total of Double Diplomat Students, Total of worker Students and Total of Graduate Students.

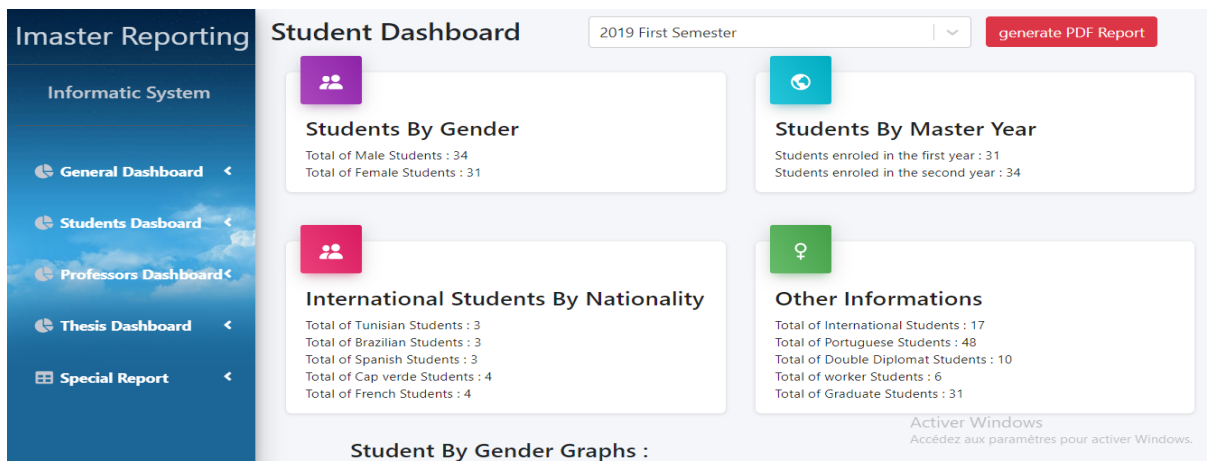


Figure 5.3: Student Dashboard view screenshot

The Students Dashboard view also compare these semester informations with the information of the other semesters also it give to the director the possibility to generate CSV reports.

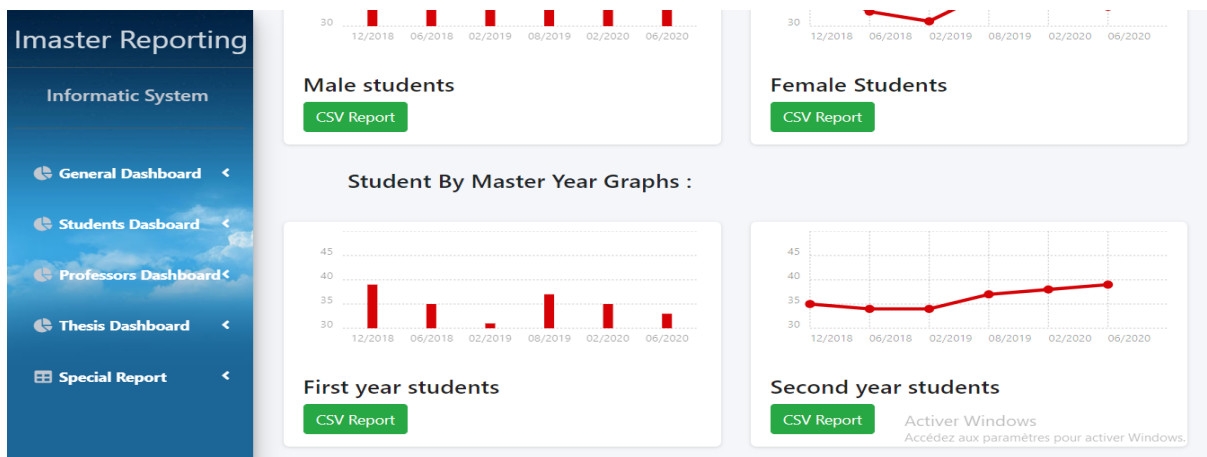


Figure 5.4: Student Dashboard view screenshot

5.1.2.3 Professors Dashboard

The Professors Dashboard view represent Professors informations (Professors By Contract Type, Professors By Grade, Professors By Nationality and Professors By Gender) also it give the possibility to choose the semester that he want to show the informations also the director can generate PDF report.

The informations represented in the Professors Dashboard view are:

- **Professors By Contract Type:** Total of Temporary Professors, Total of Permanent Professors.
- **Professors By Nationality:** Total of Portuguese Professors, Total of International Professors.
- **Professors By Grade:** Total of Full Professors, Total of Associate Professors and Total of Assistant Professors.
- **Professors By Gender:** Total of Professors, Total of Male Professors and Total of Female Professors.

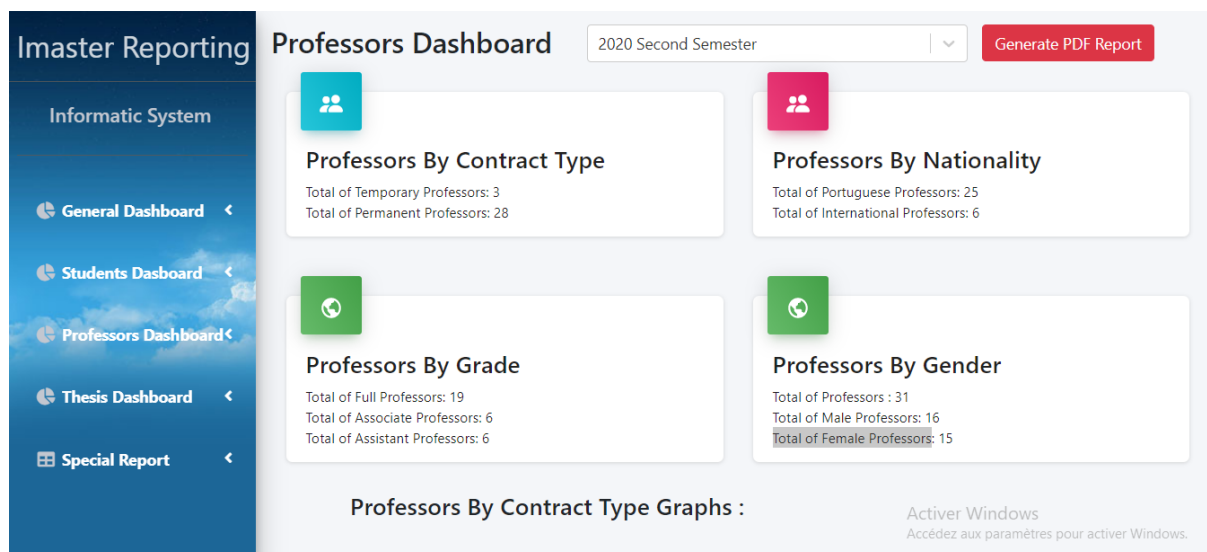


Figure 5.5: Professors Dashboard view screenshot

The Professors Dashboard view also compare Professors information with the information of the other semesters also it give to the director the possibility to generate CSV reports.

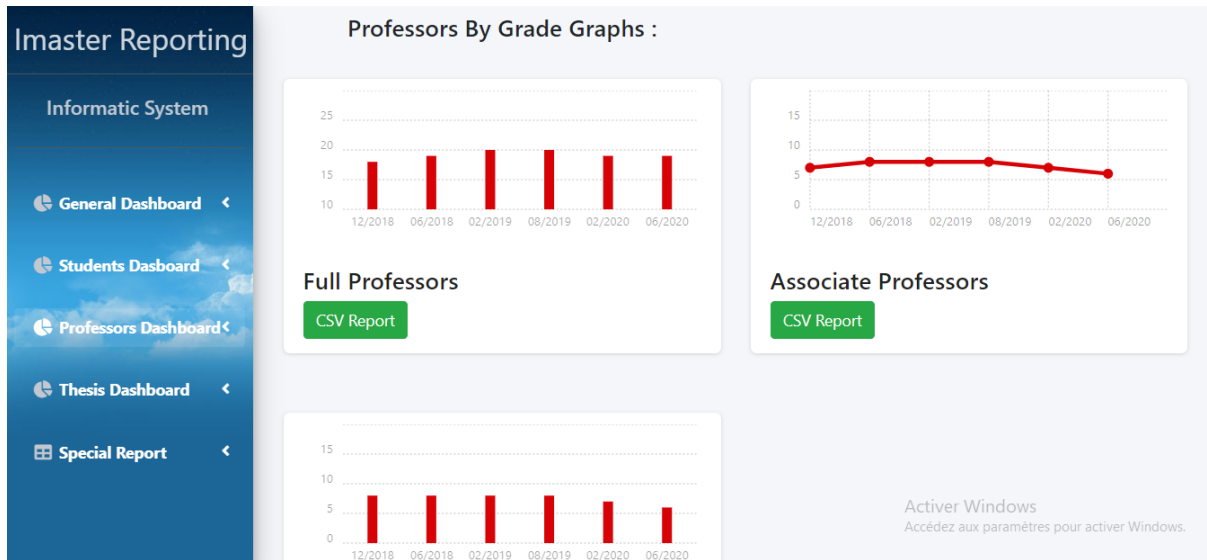


Figure 5.6: Thesis/Project/Internship Dashboard view screenshot

5.1.2.4 Thesis/Project/Internship Dashboard

The Thesis/Project/Internship Dashboard view represent Thesis/Project/Internship informations (Thesis Informations, Projects Informations, Internships Informations and Other Informations) also it give the possibility to choose the semester that he want to show the informations and also the director can generate PDF report.

The informations represented in the Professors Dashboard view are :

- **Thesis Informations:** Total of Thesis, Total of Validated Thesis, Total of Thesis proposed By scientific community, Total of Thesis proposed By Professors and Total of No choosed Thesis.
- **Projects Informations :** Total of Projects, Total of Validated Projects, Total of Projects proposed By scientific community, Total of Projects proposed By Professors and Total of No choosed Projects.
- **Internships Informations :** Total of Thesis, Total of Validated Thesis, Total of

Thesis proposed By scientific community, Total of Thesis proposed By Professors and Total of No choosed Thesis.

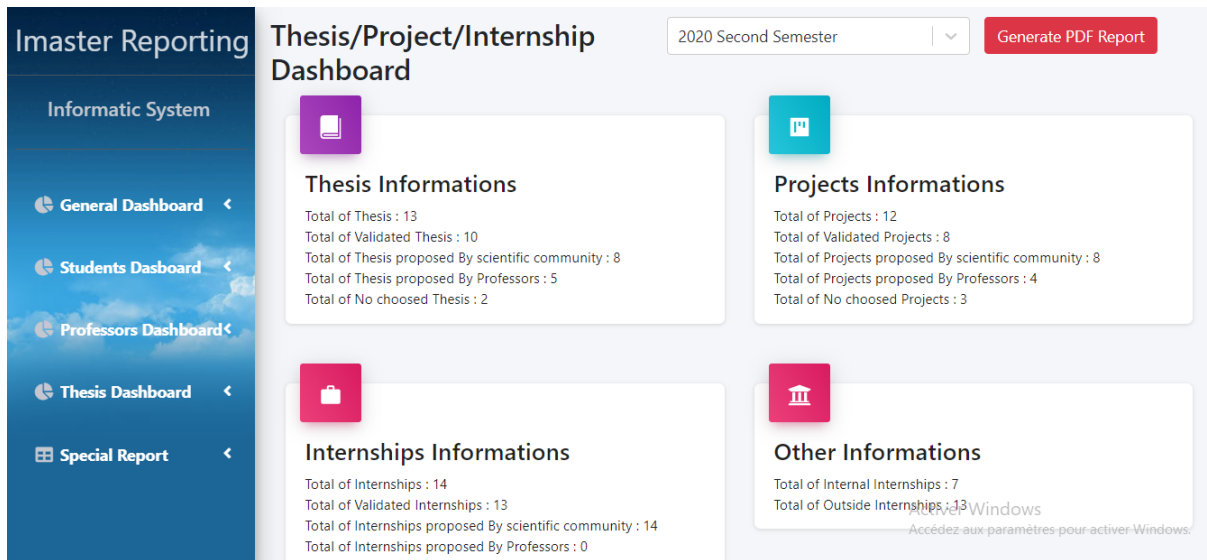


Figure 5.7: Thesis/Project/Internship Dashboard view screenshot

The Thesis/Project/Internship Dashboard also compare the informations with the informations of the other semesters also it give to the director the possibility to generate CSV reports.

5.1.3 Generate Special Report

The Generate Special Report view represent a form that is divided in 3 parts the first part in for Student informations the second part is for Professors informations and the third part is for Thesis/Project/Internship informations, for each part there is 4 Input that the director can select which informations he want to generate .

It give the possibility for the director to generate a report in PDF format with the information chosen in the form and he can filter the informations generated in the report by semesters.

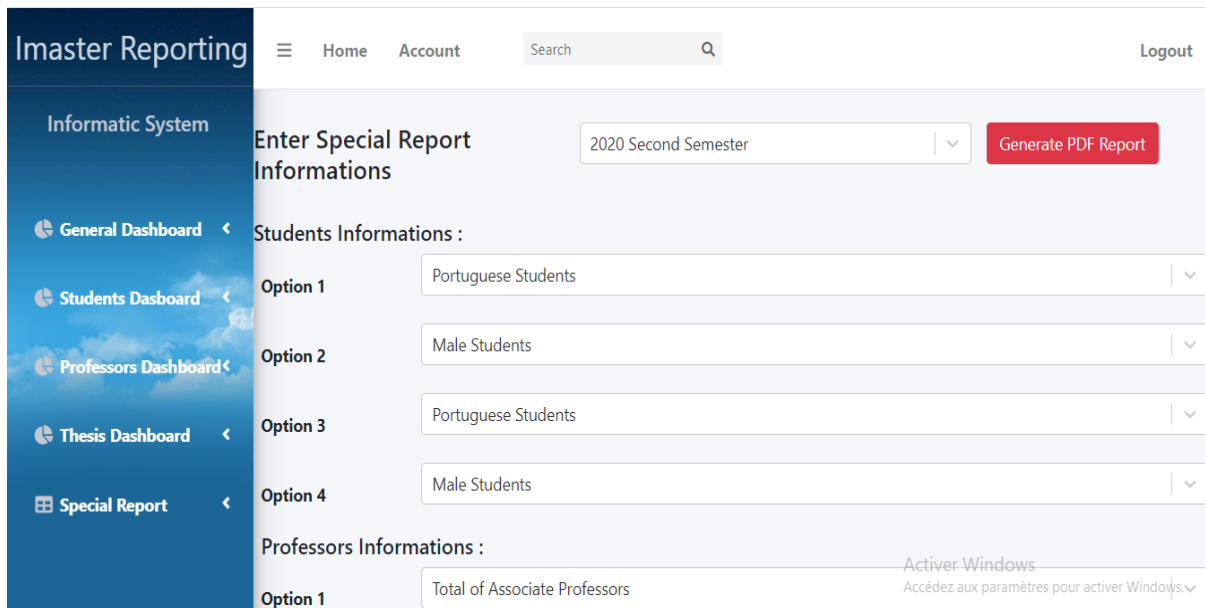


Figure 5.8: Special Report view screenshot

5.1.4 General PDF Report

The General PDF Report view represent the informations of one semester from the General Dashboard.

The informations represented in the general PDF report view are:

- **Students Informations:** Total of students, Total of international students, Total of Portuguese Students, Total of Double Diploma Students, Total of Graduate Students .
- **Professors Informations :** Total of Professors, Total of International Professors, Total of Portuguese Professors, Total of Temporary Professors, Total of Permanent Professors.
- **Thesis/Projects/Internships Informations:** Total of Thesis, Total Projects, Total Interships, Total of Internal Interships.

General Report

2020 Second Semester
Informatic System

Students Informations:

Total of Students : 72

Total of International Students : 22

Total of Portuguese Students: 50

Total of Double Diploma Students : 13

Total of Graduate Students : 36

Professors Informations:

Total of Professors : 31

Total of International Professors : 6

Total of Portuguese Professors: 25

Total of Temporary Professors: 3

Total of Permanant Professors: 28

Figure 5.9: General PDF example

5.1.5 Students PDF Report

The Students PDF Report view represent the informations of one semester from the Students Dashboard.

The informations represented in the Students PDF report view are:

- **Students By Gender:** Total of Male Students, Total of Female Students.
- **Students By Master Year:** Students enrolled in the first year, Students enrolled in the second year.
- **International Students By Nationality:** Total of Tunisian Students, Total of Brazilian Students, Total of Spanish Students, Total of Cap Verde Students and Total of French Students.

- **Other Informations** : Total of International Students, Total of Portuguese Students, Total of Double Diplomat Students, Total of worker Students and Total of Graduate Students.

~ Created By Imaster_Reporting ~

Students Report

2020 Second Semester

Total of Students : 72

Students By Gender

Total of Male Students : 38

Total of Female Students :34

Total of students By enrolment Year :

Total of First year students : 33

Total of Second Second students : 39

International Students By Nationality

Total of Tunisian Students : 2

Total of Brazilian Students : 6

Total of Spanish Students : 6

Figure 5.10: Students PDF example

5.1.6 Professors PDF Report

The Professors PDF Report view represent the informations of one semester from the Professors Dashboard.

The informations represented in the Professors PDF report view are :

- **Professors By Contract Type**: Total of Temporary Professors,Total of Permanent Professors.

- **Professors By Nationality:** Total of Portuguese Professors, Total of International Professors.
- **Professors By Grade:** Total of Full Professors, Total of Associate Professors and Total of Assistant Professors.
- **Professors By Gender:** Total of Professors, Total of Male Professors and Total of Female Professors.

~ Created By Imaster_Reporting ~

Professors Report 2020 Second Semester

Total of Professors : 31

Total of Professors By Contract Type :

Total of Temporary Professors: 3

Total of Permanent Professors: 28

Total of Professors By Grade :

Total of Full Professors: 19

Total of Associate Professors: 6

Total of Assistant Professors : 6

Total of Professors By Nationality :

Total of International Professors : 6

Total of Portuguese Professors: 25

Figure 5.11: Professors PDF Report

5.1.7 Thesis/Project/Internship PDF Report

The Thesis/Project/Internship PDF Report view represent the informations of one semester from the Thesis/Project/Internship Dashboard

The informations represented in the Thesis/Project/Internship PDF report view are:

- **Thesis Informations:** Total of Thesis, Total of Validated Thesis, Total of Thesis proposed By scientific community, Total of Thesis proposed By Professors and Total of No choosed Thesis.
- **Projects Informations:** Total of Projects, Total of Validated Projects, Total of Projects proposed By scientific community, Total of Projects proposed By Professors and Total of No choosed Projects.
- **Internships Informations:** Total of Thesis, Total of Validated Thesis, Total of Thesis proposed By scientific community, Total of Thesis proposed By Professors and Total of No choosed Thesis.

~ Created By Imaster_Reporting ~

Thesis/Project/Internship Report

2020 Second Semester

Thesis Infomations :

Total of Thesis : 13

Total of Validated Thesis : 10

Total of Thesis proposed By scientific community : 8

Total of Thesis proposed By Professors : 5

Total of No choosed Thesis : 2

Projects Infomations :

Total of Projects : 12

Total of Validated Projects : 8

Total of Projects proposed By scientific community : 8

Total of Projects proposed By Professors : 4

Figure 5.12: Thesis/Project/Internship PDF Report

5.1.8 Special Report PDF

The Special Report PDF view represent the informations of one semester from the Form Filled in the Generating Special Report view. The informations represented in the Special Report PDF view can be any information from all the other dashboards.

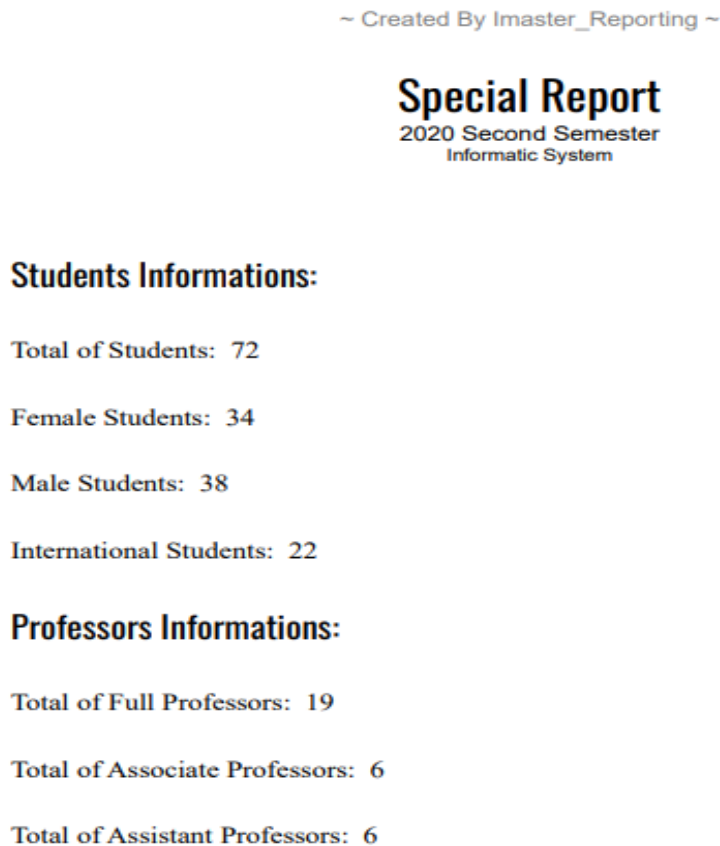


Figure 5.13: Thesis/Project/Internship PDF Report

Conclusion

This chapter, have present the Development of iMaster Reports platform, all the interfaces that the iMaster Reports contain. The next chapter will be the general conclusion.

Chapter 6

Conclusions

The management dashboard is a tool for evaluating the organization of a company or a institution made up of several indicators of its performance at given times or over periods data. It is an effective way to have a real or deferred view of the challenges of your activity. Aggregating key data helps you to gain efficiency and make better decisions. In the academic sector, data visualization plays an important role in every university. It is therefore essential to invest in the production of dashboards for data visualization. under different ways to display them as tables, charts, reports and more.

The work that was carried out during this project aimed to provide concrete solutions to the mentioned problems through the development of an application that contains several dashboards and which generates reports.

To achieve this goal, first the state of the art was presented while by the project and its issues. Second, it was studied the existing situation and identified the consequent criticisms that led us to the solution. Then, the requirements analysis and specification part of the functionalities offered by the system by specifying the different use cases. Following this step, the Design and Structure of the application has been prepared. Finally, the application was made in an incremental and iterative manner to deliver a product that meets the needs. At the moment, the application is ready for use, so we can state that the purpose which was set for us was achieved despite the technical difficulties we encountered, and the time constraint. The mission entrusted to us was very enriching and

instructive in point from the perspective of acquired skills. Apart from the technical side, this project was an opportunity to strengthen the creativity, the determination and the capacity for exchange and collaborative work. In conclusion, this work fully accomplished its objectives that are display Dashboards with useful information for the director of the master course, generate reports that contains the dashboards informations in PDF format and generate special reports with informations from several dashboards. This is only the start of a long process, we may consider other features for Imaster.reports such as adding other dashboards for the director and add other users like the students or the professors of the university .

Bibliography

- [1] *Infocursos*, <http://infocursos.mec.pt/>.
- [2] *Idashboards*, <https://www.idashboards.com/>.
- [3] K. Educa, *Top 8 trends and tools front-end javascript for 2020*, <https://morioh.com/p/f27d9edc8c01>, Accessed: 2020-10-12, 2020.
- [4] M. Wadood Majid and G. Mirzaei, *Mastering Angular JS for .NET Developers*. Apr. 2015, ISBN: 987-1-78355-398-3.
- [5] H. X. J Song's M Zhang, *Design and implementation of a vue.js-based college teaching system*, https://www.researchgate.net/publication/334468164_Design_and_Implementation_of_a_Vuejs-Based_College_Teaching_System, Jul. 2019.
- [6] S. Shah, *React js— architecture + features + folder structure + design pattern*, <https://saurabhshah23.medium.com/react-js-architecture-features-folder-structure-design-pattern-70b7b9103f22>, Aug. 2020.
- [7] R. Rickards, *Which back-end frameworks better to learn in 2020*, <https://morioh.com/p/d024b656ccc2>, Jan. 2020.
- [8] T. Reis, *Node.js and ruby on rails compared (2020)*, <https://www.imaginarycloud.com/blog/node-vs-rails/>, Aug. 2018.
- [9] E. S. Soegoto, *Implementing laravel framework website as brand image in higher-education institution*, https://www.researchgate.net/publication/327896683_Implementing_Laravel_framework_website_as_brand_image_in_higher-education_institution, Sep. 2018.

- [10] *Présentation de django*, <https://python.doctor/page-django-introduction-python>.
- [11] *What is asp.net core?* <https://dotnet.microsoft.com/learn/aspnet/what-is-aspnet-core>.
- [12] *Microservices architecture and design patterns*, <https://www.xenonstack.com/insights/microservices/>, Dec. 2018.
- [13] B. Gorman, *Introduction to entity framework*, Aug. 2020. DOI: 10.1007/978-1-4842-6044-9_1.
- [14] *Introduction to identity on asp.net core*, <https://docs.microsoft.com/en-us/aspnet/core/security/authentication/identity?view=aspnetcore-5.0&tabs=visual-studio>, 2020.
- [15] *Le mvc - model view controller*, <https://adventy.org/fr/mvc>, 2020.