

**2021 / 2022**

**SPECIALITY : TWIN**

**TOPIC OF THE INTERNSHIP REPORT:**  
**Design and implementation of a human resources module in an ERP**

Realized By: Mohamed Amine Laribi

Supervised By:

School Supervisor: Wafa Gaigi

Company Supervisor: Kaouther Allani



## Company Supervisor Signature

**Mrs. Kaouther Allani**



## Academic Supervisor Signature

**Mrs. Wafa Gaigi**

## Dedications

*I dedicate this humble work with all my heart to all the people who helped and encouraged me from near and far.*

*I dedicate this work more precisely:*

*To the soul of my dear father*

*To my dear mother who never stopped encouraging me and praying for me.*

*To my two sisters for their encouragement and advice.*

*To Syrine Jlassi, the most courageous and loving person I have seen. Thank you for your support, your encouragement, and your advice to carry out this project.*

*To the team I worked with.*

*May God grant them health and prosperity.*



Mohamed Affine

## Acknowledgements

*It is with my most sincere feelings that I thank everyone who helped me to carry out this work.*

*First, I would like to thank Mr. Khaled Junior Hafaiedh for his welcome and the trust he gave me by granting me this project.*

*My thanks are also addressed to Mrs. Kaouther Allani, supervisor of the project, for her time and her advice throughout this period.*

*My thanks also go to Mrs. Wafa Gaigi, my supervisor at the Private Higher School of Engineering and Technology, for her efforts during the internship period, her time and the quality of the supervision provided. It is with gratitude that I would like to thank her for her availability and her most pertinent advice.*

*I would like also to thank Mr. Skander Saadaoui, my technical supervisor at ESPRIT, for his advice and recommendations to improve my work.*

*My thanks go to the members of the jury, wishing that they find in this report the clarity they expect.*

# Table of content

<b>General Introduction .....</b>	<b>1</b>
<b>Chapter 1: General Presentation .....</b>	<b>3</b>
Introduction: .....	3
I. Scope of the internship.....	3
II. Host Organization .....	3
1. Mission.....	3
2. Values.....	3
3. Services .....	4
III. Problematic .....	4
IV. Benchmarking.....	5
1. Description of existent .....	5
2. Criticism .....	6
V. Proposed Solution .....	7
VI. Methodology & Teaching Strategy.....	8
1. Selected methodology .....	8
2. Scrum .....	9
3. Teaching Strategy.....	10
Conclusion.....	10
<b>Chapter 2 - Sprint 0: Project Analysis .....</b>	<b>11</b>
Introduction .....	11
I. Context Study.....	11
II. Study of the requirements .....	11
1. Functional Needs.....	11
2. Non-Functional Needs.....	12
III. Use Case diagram .....	13
1. Presenting actors.....	13
2. Global use case diagram .....	14
IV. General Database Modeling.....	14
V. System Architecture .....	16
1. Physical Architecture.....	16
2. Logical Architecture .....	17
VI. Product Backlog.....	18

VII. Work environment .....	19
1. Tools.....	19
2. Languages, Frameworks & Database.....	19
Conclusion.....	20
<b>Chapter 3 – Sprint 1: Authentication, Organigram, Employees &amp; Departments Management .....</b>	<b>21</b>
Introduction .....	21
I. Analyze and specification of needs.....	21
1. Sprint Backlog.....	21
2. General Use Case Diagram of Sprint One .....	26
II. Conceptual Study .....	27
1. System Sequence Diagrams .....	27
2. Object-Oriented Sequence Diagrams .....	27
III. Implementation .....	28
1. Employee on boarding Workflow .....	28
2. Generate Organigram .....	29
Conclusion.....	29
<b>Chapter 4 – Sprint 2: Contracts, Leaves, Releases &amp; Recruitment Management .....</b>	<b>30</b>
Introduction .....	30
I. Analyze and specification of needs.....	30
1. Sprint Backlog.....	30
2. General Use Case Diagram of Sprint Two.....	37
II. Conceptual Study .....	38
1. Object-Oriented Sequence Diagrams .....	38
a. Object-Oriented Sequence Diagram “Leave Request” case .....	38
b. Object-Oriented Sequence Diagram “Interviews Management” case.....	39
2. Activity Diagrams .....	40
III. Implementation .....	41
1. Leave Request Workflow.....	41
2. Follow-up Requests Calendar .....	42
3. Create a meeting.....	42
Conclusion.....	42
<b>Chapter 5 – Sprint 3: Skills &amp; Training Management .....</b>	<b>43</b>
Introduction .....	43

<b>I.</b>	<b>Analyze and Specification of needs .....</b>	<b>43</b>
1.	Sprint Backlog.....	43
2.	General Use Case Diagram of Sprint Three.....	46
<b>II.</b>	<b>Conceptual Study .....</b>	<b>47</b>
1.	System Sequence Diagrams .....	47
2.	Activity Diagram.....	48
<b>III.</b>	<b>Implementation .....</b>	<b>48</b>
1.	View employee skills .....	48
2.	Training creation workflow.....	49
	Conclusion.....	50
	<b>Chapter 6 – Sprint 4: Payroll, Appraisals &amp; Expenses Management .....</b>	<b>51</b>
	Introduction .....	51
<b>I.</b>	<b>Analyze and Specification of needs .....</b>	<b>51</b>
1.	Sprint Backlog.....	51
2.	General Use Case Diagram of Sprint Four.....	53
<b>II.</b>	<b>Conceptual Study .....</b>	<b>54</b>
1.	System Sequence Diagram.....	54
2.	Object-Oriented Sequence Diagram.....	54
<b>III.</b>	<b>Implementation .....</b>	<b>55</b>
1.	Consult The Pay Sheet .....	55
2.	Create an evaluation test .....	55
	Conclusion.....	55
	<b>Chapter 7 – Sprint 5: Administration Management &amp; Statistics Generation.....</b>	<b>56</b>
	Introduction .....	56
<b>I.</b>	<b>Analyze and Specification of needs .....</b>	<b>56</b>
1.	Sprint Backlog.....	56
2.	General Use Case Diagram of Sprint Five .....	57
<b>II.</b>	<b>Conceptual Study .....</b>	<b>58</b>
1.	System Sequence Diagrams .....	58
2.	Activity Diagrams .....	58
<b>III.</b>	<b>Implementation .....</b>	<b>59</b>
1.	Send Request .....	59
2.	Human Capitals Dashboard.....	59
	Conclusion.....	59

Chapter 8: SEO & Deployment.....	60
Introduction .....	60
I.    Search Engine Optimization .....	60
1.    Study of existing.....	60
2.    Keywords Validation.....	61
3.    Implement a SEO strategy.....	61
a.    Meta-Title .....	61
b.    Meta-description .....	62
c.    Meta-keywords .....	62
d.    Customization of the 404 Error Page .....	62
e.    Setting Robots.txt file .....	63
II.    Deployment.....	63
1.    Tools.....	63
2.    Explication .....	63
3.    Realization.....	64
General Conclusion .....	65
Netography .....	66

## List of illustrations

<b>Figure 1 IPACT Logo .....</b>	3
<b>Figure 2 Gusto software logo.....</b>	5
<b>Figure 3 Justworks software logo .....</b>	6
<b>Figure 4 ZenHR software logo .....</b>	6
<b>Figure 5 OnPay software logo .....</b>	6
<b>Figure 6 Marketing Positioning .....</b>	7
<b>Figure 7 Scrum Agile .....</b>	10
<b>Figure 8 Jigsaw strategy .....</b>	10
<b>Figure 9 General HCM Use Case Diagram .....</b>	14
<b>Figure 10 HR Database Modeling.....</b>	15
<b>Figure 11 Physical Architecture.....</b>	16
<b>Figure 12 Logical Architecture .....</b>	17
<b>Figure 13 Sprint 1 use case .....</b>	26
<b>Figure 14 Refinement Of "Manage Employees" use case .....</b>	26
<b>Figure 15 System sequence diagram of the "Create Department" use case.....</b>	27
<b>Figure 16 Object sequence diagram of the "Generate Organigram" use case .....</b>	28
<b>Figure 17 on boarding employee - General Info.....</b>	28
<b>Figure 18 on boarding employee - Contact.....</b>	28
<b>Figure 19 on boarding employee - Health .....</b>	29
<b>Figure 20 on boarding employee - Social .....</b>	29
<b>Figure 21 on boarding employee - Department.....</b>	29
<b>Figure 22 View Organigram.....</b>	29
<b>Figure 23 Sprint 2 use case .....</b>	37
<b>Figure 24 Refinement Of "Manage Leaves" use case.....</b>	37
<b>Figure 25 Refinement Of use case "Manage Recruitment" .....</b>	38
<b>Figure 26 Object Sequence Diagram of use case "Leave Request" .....</b>	39
<b>Figure 27 Object Sequence Diagram of "Applications Management" use case .....</b>	40
<b>Figure 28 Leave Management Activity Diagram .....</b>	41
<b>Figure 29 Leave Request -Motif.....</b>	42

<b>Figure 30 Leave Request -Comment .....</b>	42
<b>Figure 31 Leave Request-Interval .....</b>	42
<b>Figure 32 Leaves Calendar .....</b>	42
<b>Figure 33 Interview .....</b>	42
<b>Figure 34 Use Case Diagram Sprint 3 .....</b>	46
<b>Figure 35 Refinement of use case "Manage Training" .....</b>	47
<b>Figure 36 System Sequence Diagram of "Create Training" use case .....</b>	47
<b>Figure 37 Activity Diagram of "Create Training" use case.....</b>	48
<b>Figure 38 Employee Skills - Progress Mode .....</b>	48
<b>Figure 39 Employee Soft Skills - Radar .....</b>	49
<b>Figure 40 Employee Hard Skills - Radar .....</b>	49
<b>Figure 41 Create Training - General Info.....</b>	49
<b>Figure 42 Create Training - Cost.....</b>	50
<b>Figure 43 Create Training - Program .....</b>	50
<b>Figure 44 Sprint 4 Use Case .....</b>	53
<b>Figure 45 Refinement of use case "Appraisals Management" .....</b>	53
<b>Figure 46 System Sequence Diagram of use case "Create Evaluation".....</b>	54
<b>Figure 47 Object Sequence Diagram of use case "Create Evaluation" .....</b>	54
<b>Figure 48 Employee Pay Slip.....</b>	55
<b>Figure 49 Create Evaluation Test.....</b>	55
<b>Figure 50 Email Confirmation Of Evaluation.....</b>	55
<b>Figure 51 Sprint 5 Use Case .....</b>	57
<b>Figure 52 Send Request Object Sequence.....</b>	58
<b>Figure 53 Activity Diagram of use case "Handle Request" .....</b>	58
<b>Figure 54 Send Request .....</b>	59
<b>Figure 55 Human Resources Dashboard .....</b>	59
<b>Figure 56 Meta-title.....</b>	61
<b>Figure 57 Meta-description .....</b>	62
<b>Figure 58 Meta-Keywords .....</b>	62
<b>Figure 59 404 Not Found .....</b>	62
<b>Figure 60 robots.txt .....</b>	63
<b>Figure 61 Portainer Interface.....</b>	64

## List of tables

<b>Table 1 Critical benchmark .....</b>	7
<b>Table 2: Comparative study of methodologies .....</b>	9
<b>Table 3 Product Backlog.....</b>	19
<b>Table 4 Tools.....</b>	19
<b>Table 5 Languages &amp; Frameworks.....</b>	20
<b>Table 6 Sprint 1 Backlog .....</b>	25
<b>Table 7 Sprint 2 Backlog .....</b>	37
<b>Table 8 Sprint 3 Backlog .....</b>	46
<b>Table 9 Sprint 4 Backlog .....</b>	52
<b>Table 10 Sprint Backlog 5 .....</b>	57
<b>Table 11 Optimization on-page.....</b>	61
<b>Table 12 KeyWords Validation - Part 1.....</b>	61
<b>Table 13 Keywords Validation - Part 2 .....</b>	61
<b>Table 14 Deployment Tools .....</b>	63

## List of Acronyms

**ERP:** Enterprise Resource Planning

**HR:** Human Resources

**HCM:** Human Capitals Management

**HRM:** Human Resources Management

**SEO:** Search Engine Optimization

**BI:** Business Intelligence

# General Introduction

Human capital is the heart of corporate management, strategy and even growth. HCM treats the workforce, as more than just an operational cost as it is the know-how asset of a company and often determines its economic activity. Nowadays, the term HCM becomes important where companies need to attract and retain skills in an evolving set of roles.

Therefore, the Human Resources (HR) Department should prove its efficiency and relevance and should contribute to the overall performance of the company. For those reasons, the HR functions requires a pivotal and often complex role to achieve their objectives.

Nowadays, digitizing is the perfect solution to support the company management and governance. Furthermore, digitizing supports the company's growth, development, and economic success through tracking, data integrity, process automation, historical data handling, data collection and analysis, integrity and data collection and analysis.

Clearly, technology is transforming the way of work, changing strategies, removing jobs, and creating new ones. However, as traditional hierarchical structures change, teams are becoming more agile, mobile, international, and diverse.

This fact driven us to develop our HCM module as a part of an Enterprise Resources Planning (ERP) Software solution in order to meet in a more effective and efficient way the human resources needs and satisfaction. Our HCM module allows the digitizing of the traditional administrative functions of HR departments such as recruiting, trainings, payrolls, etc.

Moreover, our HCM Software offers the possibility to manage and track the employee's activities. Compared to existing software, some (Enterprise Resource Planning) ERP ignore the human resources component, while others concentrate on a single activity (e.g., Payroll).

Our solution is an efficient standard that includes most employee's activities. Therefore, our added value is to provide a fairly complete solution which can be hosted with an ERP software.

The first chapter will be devoted to the general presentation in which we will start by presenting the framework of the internship, the host company, setting up the problematic of our project, followed by a study of the different software, tracked by the proposed solution and then, the methodology adopted for this project.

The second chapter will be dedicated to the sprint 0. We start with the study of context, then presenting the global use case diagram. We continue by presenting the database modeling diagram for our system, tracked by the explanation of the system architecture. Then, we move to present the product backlog. Finally, we prepare the working environment.

The next chapters will contain the definition and specification of the requirements. Then we present and explain the conceptual study. Finally, we present some graphical interfaces.

The last chapter related to the Search Engine Optimization in which we make a study to know the keywords to improve the visibility of our application followed by a deployment in which we set up the environment required to deploy our application.

# Chapter 1: General Presentation

## Introduction:

In this first chapter, we begin by presenting the framework of the internship, the host organization, followed by an analysis of the problem, define, and criticize what exists on the current market, then we present our solution and the appropriate methodology, and we end with a conclusion.

## I. Scope of the internship

As part of our training at the private engineering and technology school ESPRIT, we must realize a final project to put into practice our skills acquired during our studies. I had the opportunity to do my internship at IPACT Consult Canada for a period of 6 months. The project consisted in the design, development, and implementation of a web application for human resources management.

## II. Host Organization

IPACT Consult is a group of experts with expertise in coaching, human development, technology, and communication. It works on a global scale to assist clients with the execution of their projects. The narrative starts in 2019, the year when IPACT Consult was developed.



Figure 1 IPACT Logo

### 1. Mission

By utilizing its technological and human resources in conjunction with a strategic action plan, IPACT Consult seeks to evaluate and improve the organization's performance.

### 2. Values

- **Respect:** Every client, candidate, and employee are treated fairly and courteously by IPACT Consult.

- **Accountability:** To complete each project on schedule and with the desired results, IPACT Consult puts their all into it.
- **Integrity:** It provides context for his behavior by producing tangible outcomes.
- **Flexibility:** It provides context for his behavior by producing tangible outcomes.

### 3. Services

- **Coaching:** IPACT Consult provides both individual and group support through hypnosis, mental training, and neuroscience coaching.
- **Consulting:** IPACT Consult offers advisory services centered on the possibilities and challenges of clients: marketing plan, technology.
- **Training:** An effective training program is provided by IPACT Consult. This training program helps the team perform better and achieve important goals faster.
- **Acceleration:** IPACT aids in establishing a sizable investor community in the business world. It can become involved at any point in the growth of a firm, from the idea stage to revenue generation.

## III. Problematic

In this section, we mention the different problems related to HCM.

- **Recruitment process management**

To retain the best talent, the recruitment process within the HR department requires a good strategy and more effort. With changing corporate culture and expectations, this task becomes increasingly difficult to manage. It can also be exhausting to target the right profiles for the company's needs. The interview strategy is a human resources issue. The topic is that most company focuses on how the work should be done rather than on the smooth integration of the profiles into the organization.

- **Adaptability and acceptance of change**

One of the issues confronting the human resources department is employee adaptation to changes in activity, position. Human resources are concerned with maintaining morale, employee satisfaction, and continuous skill improvement to respond to and meet the business requirements.

- **Identification of training needs**

Identifying training needs in the human resources sector is a real challenge in terms of setting up training for employees based on their skills. It is difficult to investigate and pinpoint the

areas in which employees need to improve. This is becoming increasingly difficult for the company's culture to develop.

- **Compliance and standardization of payroll regulations.**

Payroll regulation compliance is a critical issue in payroll management. While this payroll section is hugely affected by sociopolitical changes, the payroll manager is responsible for ensuring the consistency and conformity of the processed elements. This task becomes increasingly difficult as companies must be kept up to date on changes and legal developments.

- **Correct Pay slips**

Payroll nonconformity can have serious consequences for both the organization and its employees. These consequences may impact the elements of the gross salary, the right to paid vacation, employee and employer contributions, the net amount to be paid, and the tax deductions. The absence of a digital system may raise the risk of pay slip errors. Payroll mismanagement affects the company, the employees, the managers, and the state. This exposes businesses to legal and financial risks.

- **Payment of employees' salaries**

The payment of the employees is a very important element of payroll that must be carried out on regular basis and at a fixed date that cannot exceed one month. The lack of such an automated system makes the payment method very difficult in terms of the number and grade of employees to be paid.

## IV. Benchmarking

### 1. Description of existent

- **Gusto**

Gusto [1] is a human resources software designed for team management. It provides companies with personal management tools for payroll, integration, and team commitment monitoring. Gusto is a cloud software available as a free trial and with a paid version that varies according to the type of plan selected.



Figure 2 Gusto software logo

- **Justworks**

Justworks [2] is a human resources software destined for small and medium-sized businesses with a support solution for payroll, benefits, HR, and compliance. Features include payroll, employee profiles, workforce management information, remote or face-to-face team management and tax management. Available in cloud, software as a service (SAAS) and web versions. It offers a free demonstration upon request to the vendor. Its paid version can be customized according to the number of employees in the company.



**Figure 3 Justworks software logo**

- **ZenHR**

ZenHR [3] is a human resources software which is adapted for professional structures of different sizes. It offers some functionalities to automate the management of human resources department: employee database, employee profiles, applicant tracking, payroll, performance and recruitment management are some examples. It offers for their users a free demonstration, upon request to the vendor and a paid version.



**Figure 4 ZenHR software logo**

- **OnPay**

OnPay [4] is a payroll and personnel management tool. It makes it possible to automate the payment of taxes, deposits, payroll, and human resources management. In terms of integration, payroll can be added to accounting solutions. OnPay is a cloud software available as a free trial and with a paid version.



**Figure 5 OnPay software logo**

## 2. Criticism

The Human Resources software is a service who helps to manage employee's including their leaves requests, remunerations, skills, and training, employees' appraisals, etc.

The existed Human Resources software are Gusto, ZenHR, Justworks and onPay.

The following table shows the drawback of software's according to selected criteria.

Software	Features	User Friendly	Ergonomic	Performance
<b>gusto</b>	● MEDIUM	● HIGH	● HIGH	● LOW
<b>zenHR</b>	● MEDIUM	● HIGH	● HIGH	● LOW
<b>Justworks</b>	● MEDIUM	● HIGH	● HIGH	● HIGH
<b>onpay</b>	● MEDIUM	● HIGH	● HIGH	● MEDIUM

Table 1 Critical benchmark

## V. Proposed Solution

Manazello HCM is intended to manage human capitals to respond to employee's needs. It represents an efficient standard management solution that allows managers to better manage employees, track them, and their compensation, skills, training, absence, expenses and to automate the recruitment with a better and simple management of applications.

Manazello HCM contains several modules: employees management, organigram management, statistics generation, training management, appraisals management, payroll management, human resources contracts management, absence management, releases management, recruitment management, skills management, expenses management.

Our added value is to create a standard Human Resources Management web application that is scalable according to the client's needs. Our application should guarantee to the user the ergonomics of the interfaces with a simple and fast user experience.

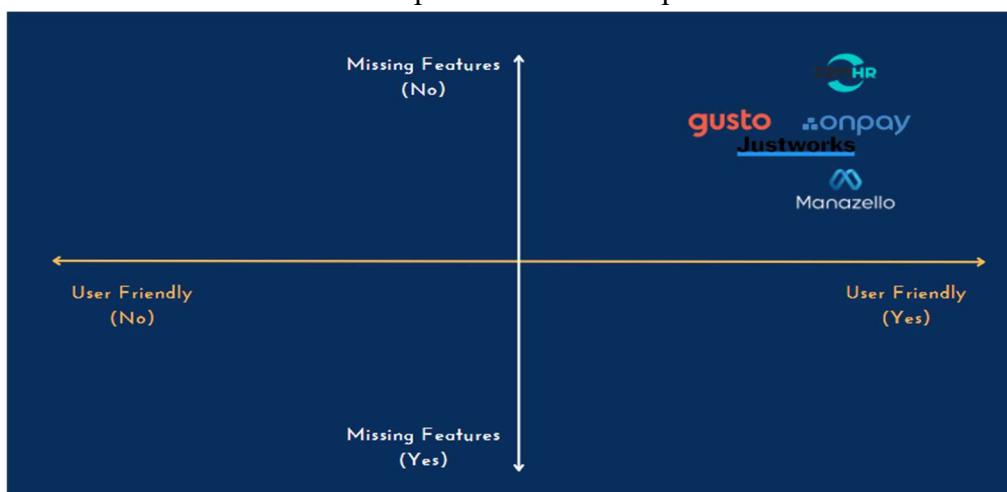


Figure 6 Marketing Positioning

In order to distinguish Manazello HCM at this level from the competition while taking into account our own strengths and shortcomings as well as those of our competitors, as shown

in the image below, we have concentrated on two axes that are useful in our application: Functionality and ergonomics.

## VI. Methodology & Teaching Strategy

It is necessary to adhere to a process while the project is in the development stage. To do that, it is necessary to adhere to a specific procedure to make this development meet the needs of the consumer.

### 1. Selected methodology

Before opting for a development method, we suggest must first establish a comparative study, analyzing the advantages and disadvantages of each one to be able to determine the method that best suits the context of our project.

We cannot use traditional methods because they were developed for small projects, do not guarantee client involvement in the project, and do not encourage needs to change throughout the project because the context of our project demands a thorough understanding of the business, client involvement in the project, and needs changing throughout the project.

We therefore start a comparative study of the different agile methodologies to choose the most appropriate one for our project. Considering the important number of methodologies, we do a study between the agile methods in particular scrum, the unified process, and Kanban.

Methodology	Benefits	Disadvantages
Scrum [5]	<ul style="list-style-type: none"> <li>- To maximize predictability, control predictability, and manage risk, an incremental and iterative strategy is used.</li> <li>- Acceptance of changing requirements as they arise during change.</li> <li>- The task must be completed collaboratively.</li> <li>- Role required.</li> <li>- The project is categorized as sprint.</li> <li>- Customer participation.</li> </ul>	-Limited documentation
Unified	<ul style="list-style-type: none"> <li>- Iterative and incremental approach.</li> </ul>	- Very long process.

<b>Process [6]</b>	- Favors communication. - Detailed documentation.	- Very detailed planning.
<b>Kanban [5]</b>	- Focused on visualizing the work - Continuous flow - flexible processes	-No role required - No sprint

**Table 2: Comparative study of methodologies**

Our decision is based on the comparison table.

We observe that, given the circumstances of our project, we are unable to select Kanban because it does not support sprints and has no role requirements in our situation.

On the other side, the unified process is not an option since it provides an extensive and in-depth documentation.

Finally, we choose the SCRUM technique, which the comparative table indicates is the best option for our project because it necessitates adapting to the customer's changing needs during each phase and requiring the client to be present while the project is being developed.

## 2. Scrum

Agile processes like SCRUM deliver the most business value in the least amount of time. The primary goal of SCRUM is to create functional software.

We can define the architecture of our team in the following way:

- **The product owner:** The person in charge of the project and in relation with the customer. He defines the **Product Backlog**.
- **Scrum planning:** The scrum team inspect the work of the product backlog into sprint backlog.
- **Sprint:** Each sprint takes between one to four weeks.
- **Scrum Master:** It represents the team leader and the project management. He intervenes in case of a situation that can delay the progress of the work planned during the sprint.
- **Scrum Team:** Eight of the ten members of our team are developers, with a consultant for information systems and a consultant for business intelligence.
- Our Scrum Team has a **Daily Meeting** which take fifteen minutes, during which each member is called to sum up what has been done, what is being done and what they plan to do next. The following figure illustrate the scrum process

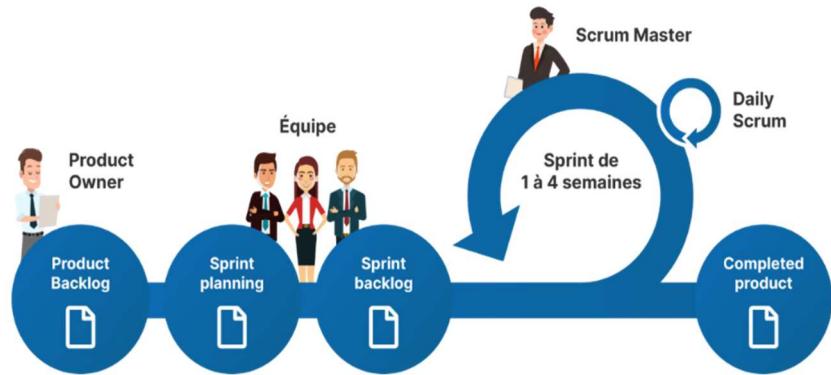


Figure 7 Scrum Agile

### 3. Teaching Strategy

We decided to use an educational strategy called "Jigsaw" to ensure a good collaboration between different team members and to develop different aspects such as listening, communication and problem solving. This is a collaborative education strategy that allows each member of the group to specialize in a particular area. Associates meet with members of other groups assigned the same aspects, then master materials and teach them to the home group. In this strategy, each collaborator in the "Home" group acts as a piece of the puzzle in question and works together to create a complete puzzle.

The following figure presents the jigsaw strategy.



Figure 8 Jigsaw strategy

### Conclusion

This section is treated as an introductory part of our project where we present the general framework of our host organization, the problematic, the benchmarking, then the solution and finally the working method. In the next chapter, we will discuss the sprint 0 which will present the study and the global analysis of the project needs.

# Chapter 2 - Sprint 0: Project Analysis

## Introduction

In this chapter, we start with the project analysis in which we focus on the context study and the functional requirements of our application. Then we present the global use case diagram chained with the database modeling diagram. Then we present the planning of each sprint and the product backlog. We close this chapter by presenting the working environment.

### I. Context Study

A project in information technology is given oversight to demonstrate it for the advantage of realizing it through requirements analysis. After doing a preliminary analysis of the risks and changes that could have occurred in the middle of the sprints, we determine the various users and their needs, which are the added value of our web application, at the completion of this phase. At this stage, we define the functional requirements that respond to the question of what the objective of our application is.

## II. Study of the requirements

Our goal is to set up a software human resources web application.

### 1. Functional Needs

We present in this section the functional needs which represent the processing that the application must do. Our application must offer the following functional requirements:

- **Authentication:** our application shall guarantee the access control and ensure the security.
- **Departments Management:** our application shall allow an effective structuring of the departments.
- **Employees Management:** our application shall allow tracking the employees' activities since their integration.
- **Generate Organigram:** our application shall allow the generation of an organigram in order to have a global vision of the departments' structure.
- **Contracts Management:** our application shall provide the ability to manage contracts including their types and track the contract status of each employee.

- **Payroll Management:** our application shall allow tracking the pay sheet of each employee.
- **Leaves Request Management:** Our application shall allow tracking a process from sending request, then the treatment of the request and finally sending an email in case of an approved request and persist it in the calendar.
- **Releases Request Management:** The application shall allow tracking a process as soon as an exit request is sent for a specific time, which must be treated by his manager.
- **Recruitment Management:** our application shall allow tracking a specific process from sharing a job offer, then getting the data of applications and finally handle request of each application.
- **Appraisals Management:** our application shall allow following up a process to create and assigning an appraisal test with sending an email.
- **Training Management:** our application shall allow following up a process to create and assigning a training with sending an email.
- **Skills Management:** our application shall allow tracking the skills of employees.
- **Skills Matrix:** our application shall allow to generate a matrix to compare the skills of employee's.
- **Mckenzie 7S:** our application shall allow to visualize a model that will help you study the interactions between the different components of your organization.
- **Expenses Management:** our application must give the possibility to manage and track the expenses assigned to employees.
- **Administration Management:** our application shall allow managing the process of purchasing requests to use our system.

## 2. Non-Functional Needs

The non-functional needs are the requirements that characterizes our system. In our project case, we have:

- **Ergonomics:** The elements should be well placed; the colors should be consistent and comfortable for user's eyes. In our case, our template present different mode (light, dark, cosmic, and corporate) where each theme has its own colors. The elements are put in their placement and in the same order to guarantee a reciprocity between all the components.
- **User friendly:** Our application should be ease of use, easy of understanding and easy of learn. In our project case, in each interface, we use help icons placed next to each field to

describe its use. To have a coherent graphical interface, we use steppers where in each step we group the fields that have the same context to simplify the process for the user.

- **Reliability:** our application should guarantee a high availability. We use microservices that are distributed and not strongly coupled, so that the changes made in one microservice X do not affect the other microservices and the whole application. In our case, our application contains 3 microservices: employees management, recruitment management and payroll management.
- **Maintainability:** our application is easy of analysis, of update and tested. In our case, our application should be open to corrective and evolutionary improvements to ensure its sustainability.
- **Performance and efficiency:** we use Gzip algorithm to reduce the size of the data to guarantee a data response in a short time.

In the next part, we identify from the requirement, the actors, and the use cases of the application.

### III. Use Case diagram

To specify in an explicit and very clear way, we define the use case diagram to guarantee a good comprehension of needs.

#### 1. Presenting actors

The actors are some external entities, which interact with our system. In our case, we have:

- **Super Administrator:** It's an actor who is the owner of the application.
- **Administrator:** It's an actor that have all rights and access in the application. It has a global view of the functioning of the application.
- **Manager:** It's an actor in the system that have some rights and access to manage the application.
- **Employee:** It's an actor included in the system that has rights an access to his space.
- **Prospect:** It's an actor who will buy the application.

After presenting the static context diagram and mentioned the actors, we move now to present our global use case diagram.

## 2. Global use case diagram

The Following diagram illustrate the global use case diagram.

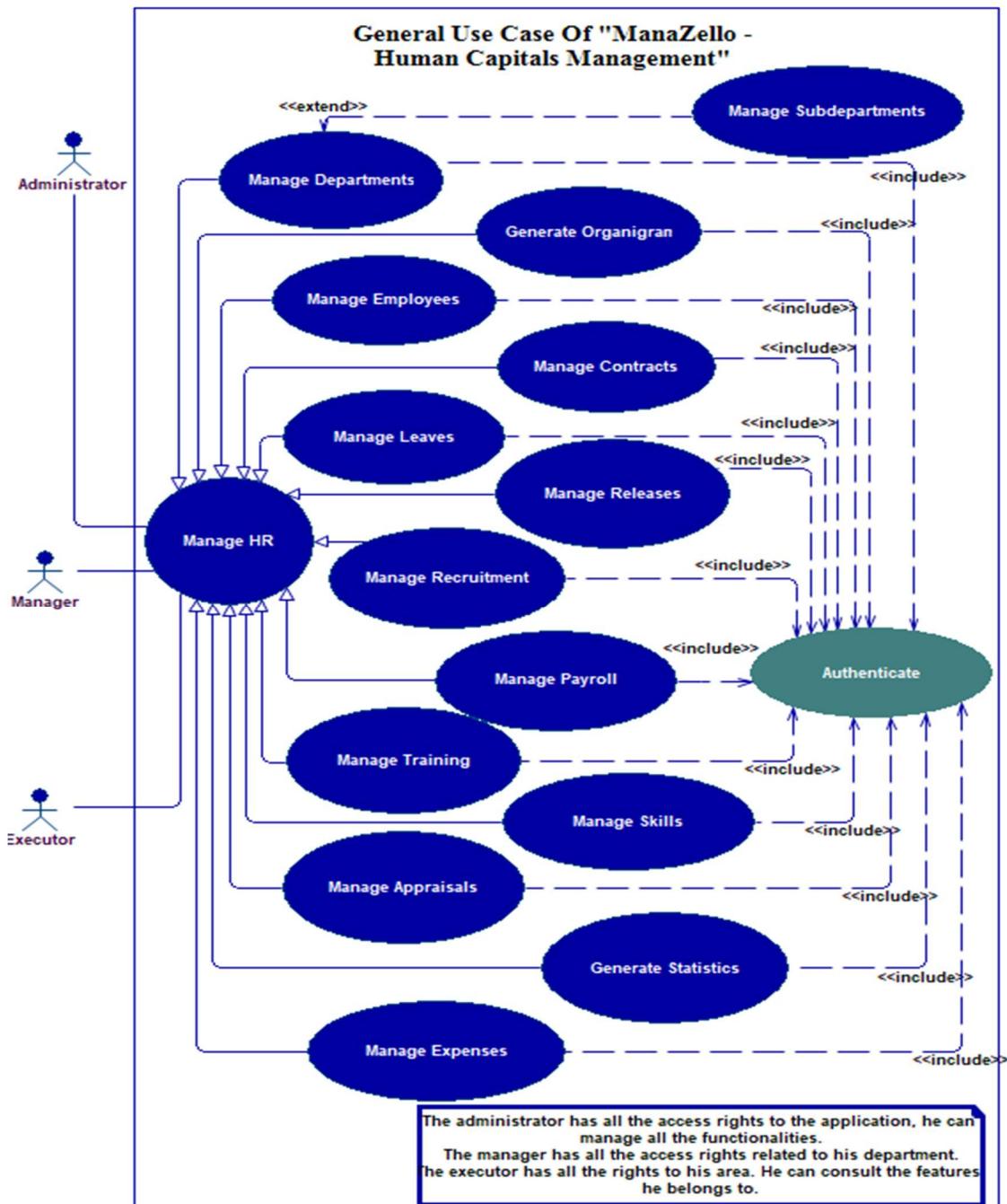


Figure 9 General HCM Use Case Diagram

## IV. General Database Modeling

In this step, we present our database modeling which gives a global vision of our system. The following illustrates the database modeling diagram.



Figure 10 HR Database Modeling

## V. System Architecture

In this part, we talk about the architecture of our system.

### 1. Physical Architecture

A physical architecture is an architecture who present the concrete components: machines, network, ...

The following figure illustrates the technical architecture of our system.



Figure 11 Physical Architecture

The physical structure of a system and its components in a diagram is its physical organization. It is a representation of the structure or organization of the physical components. The breakdown of our three-tier architecture is as follows:

A single **front-end tier** is the presentation part.

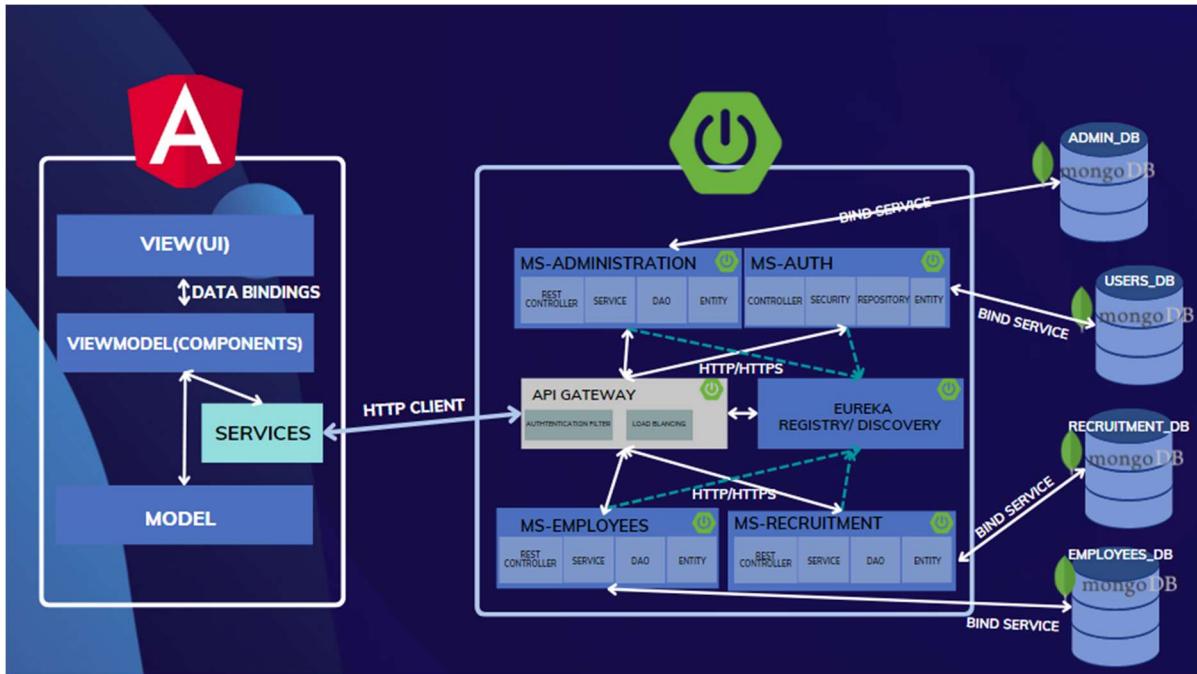
A **logical tier**: dedicated to backend operations. This tier consists of microservices including authentication, employee management, recruitment management and administration management. Through a gateway, which acts as an intermediary between them and the front-end and manages the rerouting and filtering of the different microservices and their path from the Eureka server; the discovery of all these different microservices is managed by a gateway, which acts as an intermediary between them and the front-end.

A **data tier**: contain the databases of each microservice.

## 2. Logical Architecture

A logical architecture is an architecture who describe the abstract components and their interactions.

The following figure illustrates the logical architecture of our system.



**Figure 12 Logical Architecture**

When the user initiates an action in the **View** (user interface), it is parsed and processed in the **View Model** which is responsible for acting as an intermediary between the user interface and the **Model** (representing data) and responsible for processing and manipulating data (Data Bindings). To communicate with the backend, the View Model must consume **Services** using the application module **Http Client** that contains the rest of the API to use.

This action taken directly in a **Gateway** to filter requests, verifying the existence of the relevant microservice by going through a **Discovery server** (Eureka Server) to be propagated to the controller. This action propagates from the **controller** to the service containing the business logic implementation of the operation. This service should use a **DAO** (Mongo Repository) class that represents a design pattern that will ensure interaction with the database. The result will be returned in JSON format which will go through the gateway to ensure data redirection to the relevant element that will notify the view of the modifications made.

## VI. Product Backlog

The following table present the product backlog.

ID	Features	User Story	Estimation	Priority
1	Authenticate	As a user, I want to have access to the application	8 days	1
2	Department Management	As a user, I want to manage departments	6 days	1
3	Employees Management	As a user, I want to manage employees	10 days	1
4	Generate Organigram	As a user, I want to show the structure of departments	6 days	1
5	Leaves Management	As a user, I want to manage and track leaves request.	7 days	2
6	Releases Management	As a user, I want to manage, and track exits request.	7 days	2
7	Contracts Management	As a user, I want to manage employees' contracts	4 days	2
8	Recruitment Management	As a user, I want to follow a process to integrate new profile.	12 days	2
9	Skills Management	As a user, I want to manage and track employee's skills.	4 days	3
10	Training Management	As a user, I want to assign training	6 days	3
11	Evaluation Management	As a user, I want to manage evaluations.	5 days	3
12	Payroll Management	As a user, I want to display the employee pay sheet.	15 days	3
13	McKinsey7S	As a user, I want to analyze firm's organizational design.	1 day	4
14	Expenses Management	As a user, I want to manage travel expenses	7 days	4

15	Administration Management	As a super admin, I want to manage prospects.	18 days	4
----	---------------------------	---	---------	---

**Table 3 Product Backlog**

## VII. Work environment

In this step, we present the different tools, languages et frameworks used to build our project.

### 1. Tools

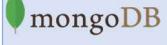
The following table present the different tools used in the project.

Logo	Tool	Description
 Visual Studio Code	<b>Visual Studio Code</b>	A code editor to develop the frontend application. It offers some extensions which facilitate writing code.
	<b>IntelliJ IDE</b>	An IDE permits to develop the backend of the application using java language.
	<b>Git</b>	A version control system that allows to commit changes and to push the code to a remote repository.
	<b>GitLab</b>	A collaborative and continuous integration platform that allows to manage the source code of the application
	<b>Postman</b>	A rest API testing software
	<b>MongoDB Compass</b>	An interactive system for analyzing mongo DB records.
	<b>Trello</b>	A task tracker management software to manage any type of projects between the team.
	<b>Power Designer</b>	A data modeling tool that allows to create diagrams.

**Table 4 Tools**

### 2. Languages, Frameworks & Database

The following table presents the languages, frameworks and database used in the project.

Logo	Framework/Language	Description
	<b>Angular</b>	A client framework based on TypeScript language with a high scalability.
	<b>Spring</b>	A backend application builder.
	<b>MongoDB</b>	A non-relational database with a high, flexible, horizontal scalability of data.
	<b>Java</b>	An object-oriented programming language because spring boot is based on Java.
	<b>Type Script</b>	A programming language allows to improve the productivity of the code.

**Table 5 Languages, Frameworks & Database**

## Conclusion

This second chapter was devoted to the analysis of the project. We presented the context study of our project and the requirements of our application, which allowed us to draw the global use case diagram. Then, we presented the database modeling diagram and the system's architecture. Next, we focused on the planning of each sprint, the product backlog, and the working environment. The next chapter will be dedicated to the first sprint.

# Chapter 3 – Sprint 1: Authentication, Organigram, Employees & Departments Management

## Introduction

In this chapter, we present the first sprint which contains: analyze and specification of needs, the conceptual study, and the implementation. The goal in this chapter is to develop the authentication part, employee's management, departments management (including sub-departments management) and generate the organigram.

## I. Analyze and specification of needs

### 1. Sprint Backlog

The following table present a list of the features to realize, to be completed and terminate throughout the sprint to have a deliverable conformity with what has been mentioned in the product backlog.

ID	User Story	Task ID	Task	Estimation
			Task	
1.1	As a user, I want to authenticate	1.1.1	Implement and test service in backend	3 days
			Consume service in frontend	
			Create the interface	
			Test feature	
1.1	As a user, I want to reset my password	1.1.2	Implement and test service in backend	2 days
			Consume service in frontend	
			Create the interface	
1.2	As a user, I want to create a department	1.2.1	Implement and test service in backend	1 day
			Consume service in frontend	
			Create add department interface	
			Test feature in frontend	

1.2	As a user, I want to update the information of an existent department	1.2.1	Implement and test service in backend	
		1.2.2	Consume service in frontend	
		1.2.3	Create edit department interface	
		1.2.4	Test feature in frontend	
1.2	As a user, I want to archive a department	1.2.1	Implement and test service in backend	1 day
		1.2.2	Consume and test service in frontend	
1.2	As a user, I want to restore a department	1.2.1	Implement and test service in backend	
		1.2.2	Consume and test service in frontend	
1.2	As a user, I want to display all departments	1.2.1	Implement and test service in backend	2 days
		1.2.2	Consume service in frontend	
		1.2.3	Create department list interface	
1.2	As a user, I want to display the history departments.	1.2.1	Implement and test service in backend	1 day
		1.2.2	Consume service in frontend	
		1.2.3	Create history interface	
1.2	As a user, I want to delete a department.	1.2.1	Implement and test service in backend	1 day
		1.2.2	Consume and test service in frontend	
1.2	As a user, I want to export data in pdf.	1.2.1	Implement and test in frontend	2 days
1.2	As a user, I want to export data in excel.	1.2.2	Implement and test service in backend	
1.2	As a user, I want to create a new sub department.	1.2.1	Implement and test service in backend	1 day
		1.2.2	Consume service in frontend	
		1.2.3	Create sub department add interface	
		1.2.4	Test feature	

1.2	As a user, I want to display sub departments by department.	1.2.1	Implement and test service in backend	1 day
		1.2.2	Consume service in frontend	
		1.2.3	Create tags to display subdepartments in each department	
1.2	As a user, I want to remove a sub department from department item.	1.2.1	Implement and test service in backend	2 hours
		1.2.2	Consume service in frontend	
1.2	As a user, I want to do an advanced search of the data	1.2.1	Implement and test service in frontend	4 hours
1.2	As a user, I want to do a filter on the data	1.2.1	Implement and test service in frontend	
1.2	As a user, I want to edit sub department	1.2.1	Implement and test service in backend	1 day
		1.2.2	Consume service in frontend	
		1.2.3	Create Edit interface	
		1.2.4	Test feature	
1.2	As a user, I want to archive sub department	1.2.1	Implement and test service in backend	1 day
			Consume service in frontend	
			Create Archive Button	
			Test Feature	
1.2	As a user, I want to restore sub department	1.2.1	Implement and test service in backend	1 day
			Consume service in frontend	
			Create Restore Button in frontend	
			Test feature	
1.2	As a user, I want to display sub departments in	1.2.1	Implement and test service in backend	1 day
		1.2.2	Consume service in frontend	
		1.2.3	- Create sub departments interface	

	different mode (table, grid(2x2), grid(3x3)).			
<b>1.2</b>	As a user, I want to export data in pdf.	1.2.1	Implement frontend service with pdf personalization	1 day
<b>1.2</b>	As a user, I want to export data in excel.	1.2.1	Implement frontend service excel file	
<b>1.2</b>	As a user, I want to do an advanced search of the data.	1.2.1	Implement frontend search service	1 hour
<b>1.3</b>	As a user, I want to create a new employee.	1.3.1	Implement and test service in backend	1 day
		1.3.2	Consume service in frontend	
		1.3.3	Create add employee interface	
		1.3.4	Test feature	
<b>1.3</b>	As a user, I want to edit an employee.	1.3.1	Implement and test service in backend	1 day
		1.3.2	Consume service in frontend	
		1.3.3	Create Edit interface	
		1.3.4	Test feature	
<b>1.3</b>	As a user, I want to view details of an employee.	1.3.1	Implement and test service in backend	2 days
		1.3.2	Consume service in frontend	
		1.3.3	Create view details interface	
<b>1.3</b>	As a user, I want to display employees in different mode	1.3.1	Implement and test service in backend	2 days
		1.3.2	Consume service in frontend	
		1.3.3	Create interface	
<b>1.3</b>	As a user, I want to archive an employee.	1.3.1	Implement and test service in backend	1 day
		1.3.2	Consume service in frontend	

1.3	As a user, I want to restore an employee	1.3.1	Implement and test service in backend	
		1.3.2	Consume service in frontend	
		1.3.3	Test Feature	
1.3	As a user, I want to show history of employees	1.3.1	Implement and test service in backend	1 day
		1.3.2	Consume service in frontend	
		1.3.3	Create history interface	
1.3	As a user, I want to search data	1.3.1	Implement frontend search service	12 hours
1.3	As a user, I want to filter data	1.3.1	Implement frontend filter service	
1.4	As a user, I want to view organigram	1.4.1	Consume department service in frontend	2 days
		1.4.2	Consume sub department service in frontend	
		1.4.3	Create organigram interface	
1.4	As a user, I want to export organigram in pdf.	1.4.1	- Implement frontend service with pdf personalization	1 day

**Table 6 Sprint 1 Backlog**

After presenting the sprint backlog, we present the use case related to this sprint.

## 2. General Use Case Diagram of Sprint One

The following figure present the general use case diagram of the first sprint.

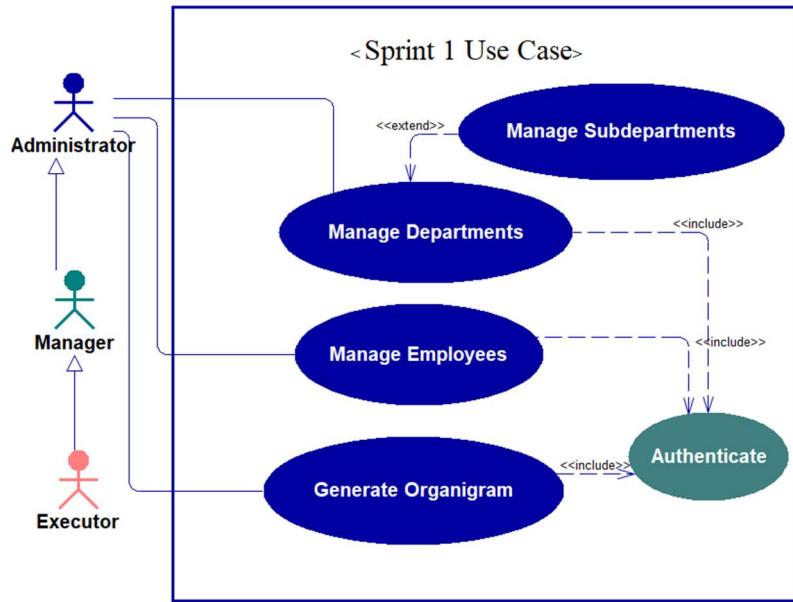


Figure 13 Sprint 1 use case

The following figure presents the refinement of “manage employees” use case.

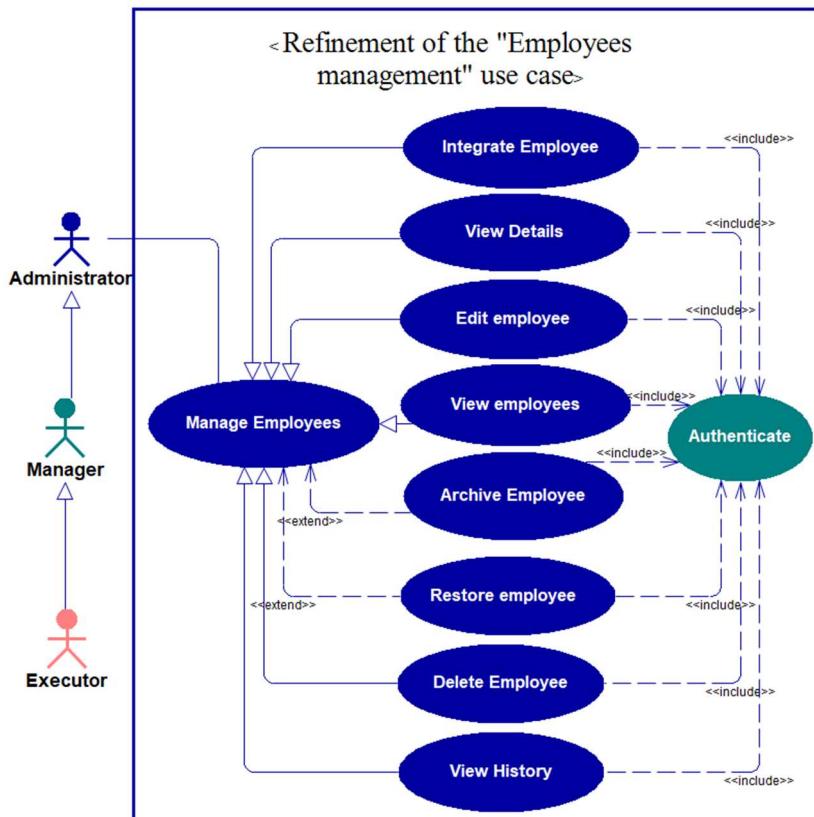


Figure 14 Refinement Of "Manage Employees" use case

After presenting the use case of the first sprint, we move to the conceptual study.

## II. Conceptual Study

In this section, we present and explain the conceptual study for sprint 1.

### 1. System Sequence Diagrams

The following figure illustrates the sequence system diagram of create department use case.

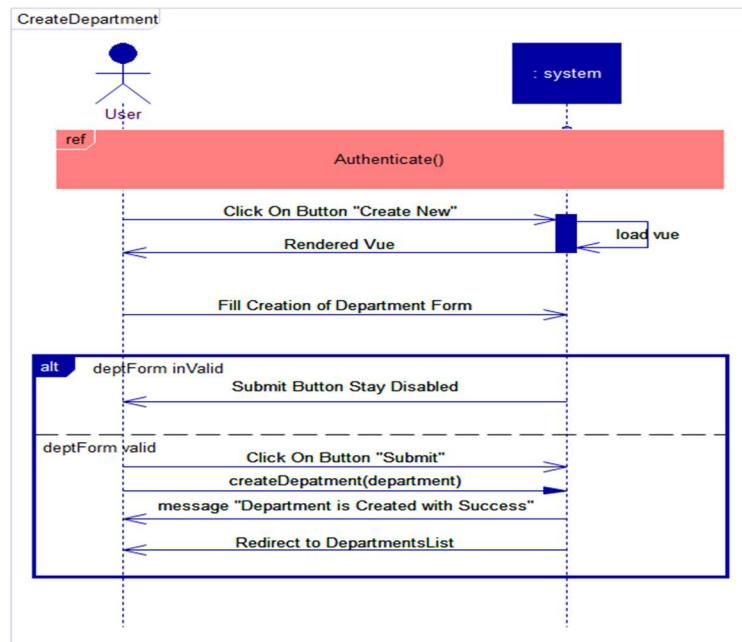
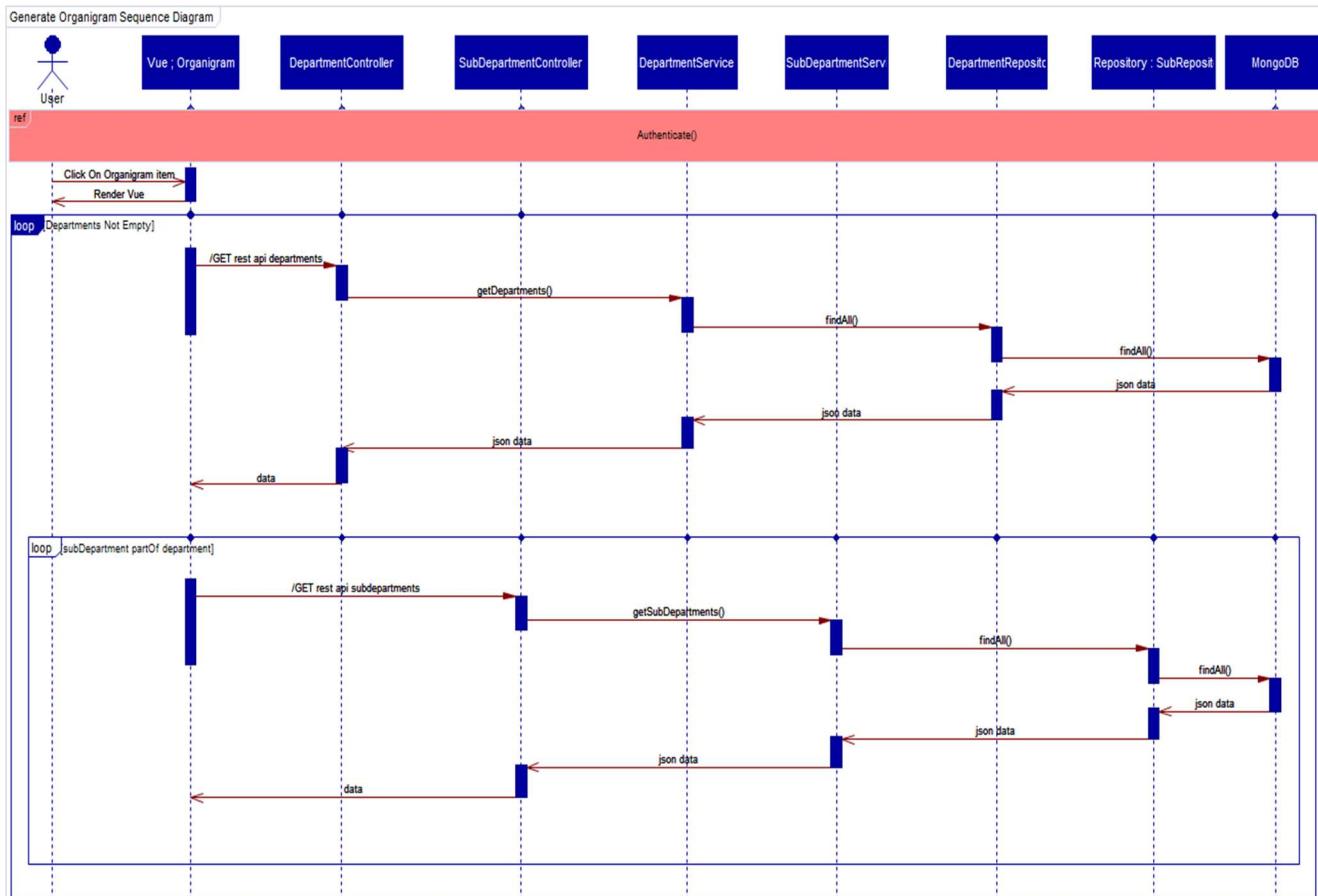


Figure 15 System sequence diagram of the "Create Department" use case

### 2. Object-Oriented Sequence Diagrams

To better understand the interactions between objects of our system, we established some sequence diagrams that we find interesting.



**Figure 16 Object sequence diagram of the "Generate Organigram" use case**

The following figure illustrates the object sequence diagram of use case “Generate Organigram”.

To display the organigram, the user should be authenticated. Once logged, the user has a vision on the different departments as well as the sub-departments.

In the next section, we present some different graphical interfaces.

### III. Implementation

#### 1. Employee on boarding Workflow

The following interface present the workflow of integration a new employee in the application.

This screenshot shows the 'Create Employee' form in its first step, 'General'. The form includes fields for Personal information (Employee Name, Birth Date, City, Note), Contact details (Email Address, Phone number), and other personal details like Gender, Nationality, and Driving License. A 'NEXT' button is at the bottom right.

**Figure 17 on boarding employee - General Info**

This screenshot shows the 'Create Employee' form in its second step, 'Contact'. It displays the email address and phone number entered in the previous step. Navigation buttons 'PREV' and 'NEXT' are visible at the bottom.

**Figure 18 on boarding employee - Contact**

Figure 21 on boarding employee - Department

Figure 20 on boarding employee - Social

Figure 19 on boarding employee - Health

To integrate an employee, the manager must follow a specific sequence. He must carefully fill out each phase starting with general information, followed by contact information, followed by department assignment, social security information and finally health information.

## 2. Generate Organigram

The following interface present the organigram.



Figure 22 View Organigram

## Conclusion

During this first sprint, we presented the sprint backlog, followed by the use case diagram and the database modeling diagram. Finally, we have some graphical interfaces. In the next chapter, we discuss about the sprint 2 which contains the requirements for our next deliverable.

# Chapter 4 – Sprint 2: Contracts, Leaves, Releases & Recruitment Management

## Introduction

In this chapter, we present the second sprint which contains: analyze and specification of needs, conceptual study, and implementation. The goal of this chapter is to develop the HR contracts, leaves request, releases request and the process of the recruitment management

### I. Analyze and specification of needs

In this first step, we specify and present our requirements.

#### 1. Sprint Backlog

The following table contains a list of items to be achieved and closed during the sprint to have a release in conformity with what was specified in the product backlog.

ID	User Story	Task ID	Task	Estimation
1.1	As a user, I want to assign a contract to an employee	1.1.1	Implement and test backend service	1 day
		1.1.2	Consume service in frontend	
		1.1.3	Create interface assign contract to an employee	
		1.1.4	Test feature	
1.1	As a user, I want to view the contract of the employee	1.1.1	Implement and test backend service	3 hours
		1.1.2	Consume service in frontend	
		1.1.3	Create interface view contract	
1.1	As a user, I want to display the contracts	1.1.1	Implement and test backend service	4 hours
		1.1.2	Consume service in frontend	
		1.1.3	Create view contracts interface	
1.1	As a user, I want to update an	1.1.1	Implementation and testing backend service	2 hours
		1.1.2	Consume service in frontend	

	existent contract	1.1.3	Create update contract interface	
		1.1.4	Testing feature	
<b>1.1</b>	As a user, I want to do an advanced search	1.1.1	Implement and test frontend service	1 hour
<b>1.1</b>	As a user, I want to do a filter of the data contracts	1.1.1	Implement and test frontend service	8 hours
<b>1.1</b>	As a user, I want to archive a contract	1.1.1	Implement and test backend service	2 hours
		1.1.2	Consume service in frontend	
<b>1.1</b>	As a user, I want to restore a contract	1.1.1	Implement and test backend service	
		1.1.2	Consume service in frontend	
<b>1.1</b>	As a user, I want to delete a contract	1.1.1	Implement and test backend service	1 hour
		1.1.2	Consume service in frontend	
<b>1.1</b>	As a user, I want to export data as pdf	1.1.1	Implement pdf service in frontend	12 hours
<b>1.1</b>	As a user, I want to export data as excel.	1.1.1	Implement excel service in frontend	
<b>1.1</b>	As a user, I want to	1.1.1	Implement and test backend service	3 hours
		1.1.2	Consume service in frontend	
		1.1.3	Create history interface	

	history of contracts			
<b>1.1</b>	As a user, I want to update the number of items	1.1.1	Implement frontend service	2 hours
<b>1.2</b>	As a user, I want to apply for a leave of absence	1.2.1	Implement and test backend service	1 day
		1.2.2	Consume service in frontend	
		1.2.3	Create apply interface	
		1.2.4	Test feature	
<b>1.2</b>	As a user, I want to save my leave request as a draft	1.2.1	Implement and test backend service	4 hours
		1.2.2	Consume and test feature in frontend	
<b>1.2</b>	As a user, I want to display my requests	1.2.1	Implement and test backend service	4 hours
		1.2.2	Consume service in frontend	
		1.2.3	Create requests interface	
<b>1.2</b>	As a user, I want to display my drafts requests.	1.2.1	Implement and test backend service	5 hours
		1.2.2	Consume service in frontend	
		1.2.3	Create drafts interface	
<b>1.2</b>	As a user, I want to display all leave requests	1.2.1	Implement and test backend service	10 hours
		1.2.2	Consume service in frontend	
		1.2.3	Create leaves interface	
<b>1.2</b>	As a user, I want to edit	1.2.1	Implement and test backend service	5 hours
		1.2.2	Consume service in frontend	
		1.2.3	Create edit interface	

	the request status	1.2.4	Test feature	
<b>1.2</b>	As a user, I want to see leave balance	1.2.1	Implement and test backend service	2 days
		1.2.2	Consume data in frontend	
<b>1.2</b>	As a user, I want to export data in pdf.	1.2.1	Implement pdf service in frontend	1 day
<b>1.2</b>	As a user, I want to export data in excel.	1.2.1	Implement excel service in frontend	
<b>1.2</b>	As a user, I want to search data	1.2.1	Implement and test service in frontend	1 hour
<b>1.2</b>	As a user, I want to archive data.	1.2.1	Implement and test backend service	3 hours
		1.2.2	Consume service in frontend	
<b>1.2</b>	As a user, I want to restore data.	1.2.1	Implement and test backend service	
		1.2.2	Consume service in frontend	
<b>1.2</b>	As a user, I want to view the requests history	1.2.1	Implement and test backend service	4 hours
		1.2.2	Create history interface	
<b>1.3</b>	As a user, I want to apply for a request of release	1.3.1	Implement and test backend service	2 days
		1.3.2	Consume service in frontend	
		1.3.3	Create apply service	
		1.3.4	Test feature	
<b>1.3</b>		1.3.1	Implement and test backend service	

	As a user, I want to save my release request as a draft	1.3.2	Consume and testing service in frontend	
1.3	As a user, I want to display my own releases	1.3.3	Implement and test backend service	5 hours
			Consume service in frontend	
			Create interface	
1.3	As a user, I want to display my own drafts releases.	1.3.1	Implement and test backend service	1 day
		1.3.2	Consume service in frontend	
		1.3.3	Create interface	
1.3	As a user, I want to display releases requests	1.3.1	Implement and test backend service	1 day
		1.3.2	Consume service in frontend	
		1.3.3	Create interface	
1.3	As a user, I want to filter data	1.3.1	Implement and test service in frontend	2 hours
1.3	As a user, I want to search data	1.3.1	Implement and tes service in frontend	12 hours
1.3	As a user, I want to archive data.	1.3.1	Implement and test backend service	3 hours
		1.3.2	Consume and test service in frontend	
1.3	As a user, I want to restore data.	1.3.1	Implement and test backend service	3 hours
		1.3.2	Consume and test service in frontend	
1.3		1.3.1	Implement and test backend service	4 hours

	As a user, I want to view the history of the releases	1.3.2	Consume service in frontend	
1.3	As a user, I want to delete data.	1.3.1	Implement and test backend service	2 hours
		1.3.2	Consume service in frontend	
1.4	As a user, I want to post a job offer	1.4.1	Implement and test backend service	2 days
		1.4.2	Consume service in frontend	
		1.4.3	Create add job offer interface	
		1.4.4	Test feature	
1.4	As a user, I want to display job offers	1.4.1	Implement and test backend service	1 day
		1.4.2	Consume service in frontend	
		1.4.3	Create offers interface	
1.4	As a user, I want to edit a job offer	1.4.1	Implement and test backend service	1 day
		1.4.2	Consume service in frontend	
		1.4.3	Create edit job interface	
		1.4.4	Test feature	
1.4	As a user, I want to export data in pdf file	1.4.1	Implement and test service in frontend	1 day
1.4	As a user, I want to export data in excel file	1.4.1	Implement and test service in frontend	12 hours
1.4	As a user, I want to archive data	1.4.1	Implement and test backend service	4 hours
		1.4.2	Consume and test service in frontend	
1.4		1.4.1	Implement and test backend service	4 hours

	As a user, I want to restore data	1.4.2	Consume service in frontend	
1.4	As a user, I want to display the history of offers	1.4.1	Implement and test backend service	2 hours
		1.4.2	Consume service in frontend	
		1.4.3	Create history interface	
1.4	As a user, I want to do an advanced search	1.4.1	Implement and test frontend service	2 hours
1.4	As a user, I want to do a filter of the data	1.4.1	Implement and test frontend service	12 hours
1.4	As a user, I want to display all applications	1.4.1	Implement and test backend service	4 days
		1.4.2	Consume service in frontend	
		1.4.3	Create applications list	
1.4	As a user, I want to screen applications.	1.4.1	Implement and test backend service	5 days
		1.4.2	Consume service in frontend	
1.4	As a user, I want to reject application	1.4.1	Implement and test backend service	6 hours
		1.4.2	Consume service in frontend	
1.4	As a user, I want to approve an application	1.4.1	Implement and test backend service	6 hours
		1.4.2	Consume service in frontend	
1.4	As a user, I want to fix an	1.4.1	Implement and test backend service	2 days
		1.4.2	Consume service in frontend	

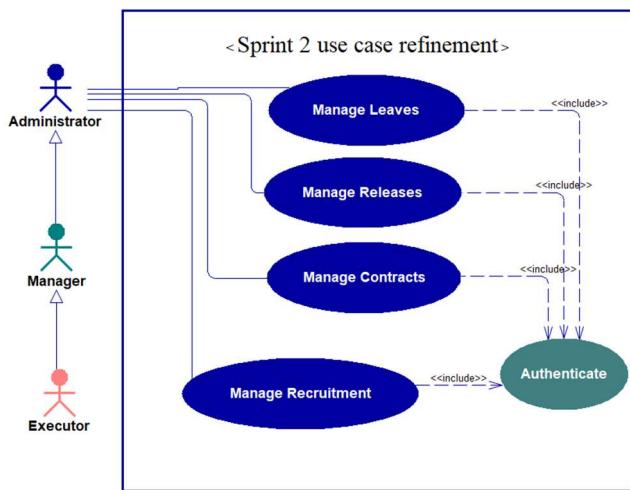
	interview with the candidate	1.4.3	Test feature in frontend	
		1.4.2	Consume service in frontend	
		1.4.3	Testing feature in frontend	

**Table 7 Sprint 2 Backlog**

After presenting the sprint backlog, now we present the use case diagram.

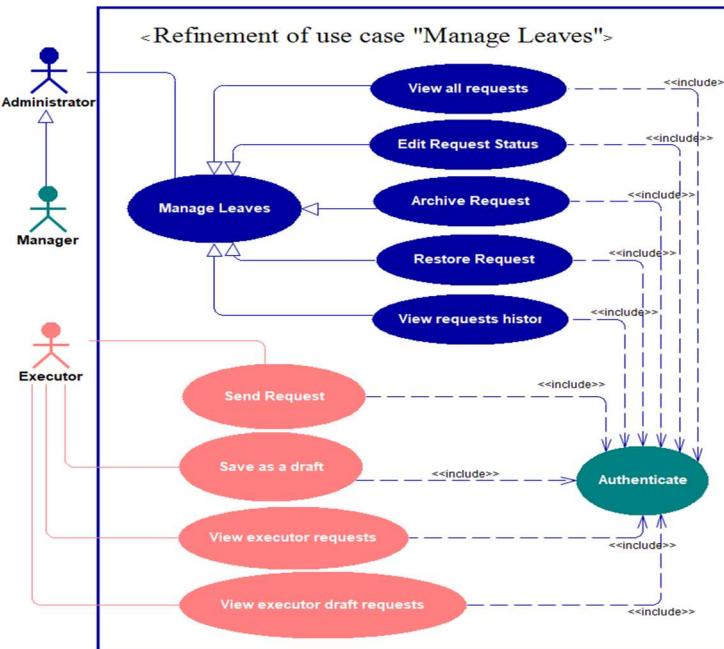
## 2. General Use Case Diagram of Sprint Two

The following figure illustrates the general use case related to the second sprint.



**Figure 23 Sprint 2 use case**

The following figure shows the refining of the "Manage Leaves" use case.



**Figure 24 Refinement Of "Manage Leaves" use case**

The following figure shows the refining of "Manage Recruitment" use case.

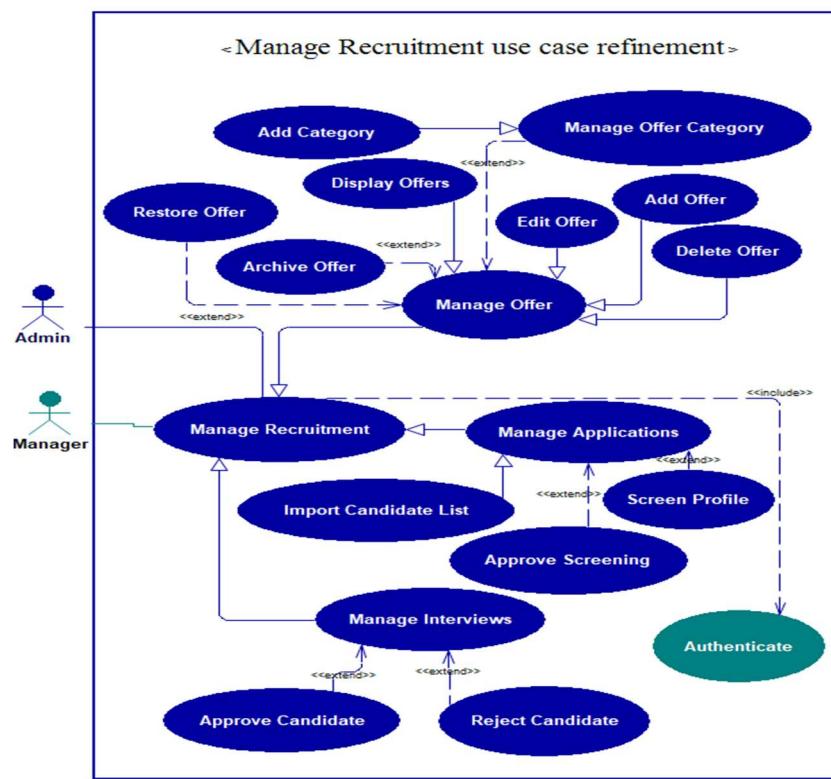


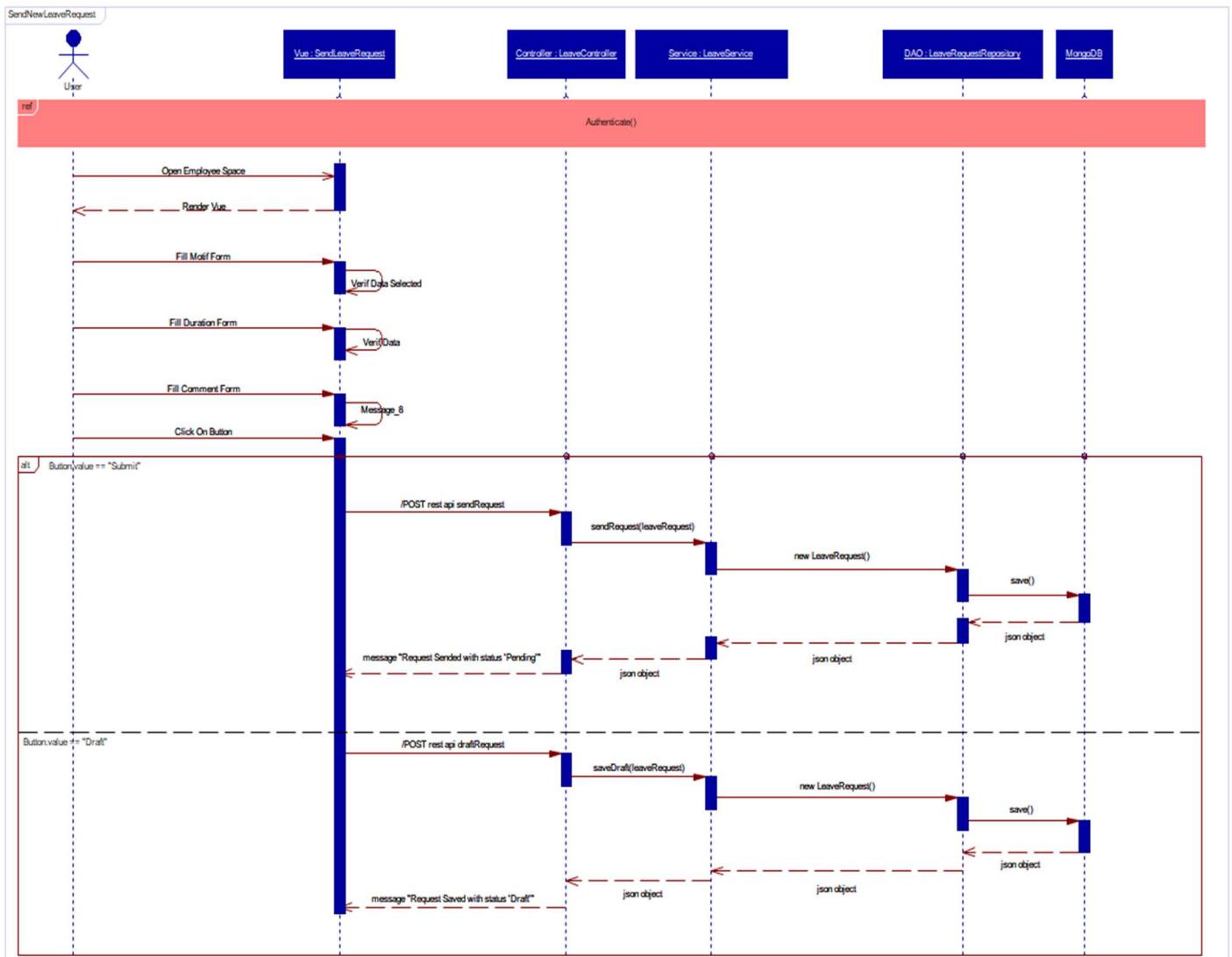
Figure 25 Refinement Of use case "Manage Recruitment"

## II. Conceptual Study

### 1. Object-Oriented Sequence Diagrams

#### a. Object-Oriented Sequence Diagram “Leave Request” case

The following figure shows the object sequence diagram of the "Leave Request" use case.



**Figure 26 Object Sequence Diagram of use case "Leave Request"**

To make a request for leave, the employee is supposed to access his space to send a request for absence in which he specifies the reason for the absence, the duration, and a comment if necessary. If the employee clicks on the submit button, then his request will be sent directly to his manager to manage it. If not, the employee saves his request for a later date, while clicking on the draft button.

### b. Object-Oriented Sequence Diagram “Interviews Management” case

The following figure shows the sequence diagram of the "Application management" use case.

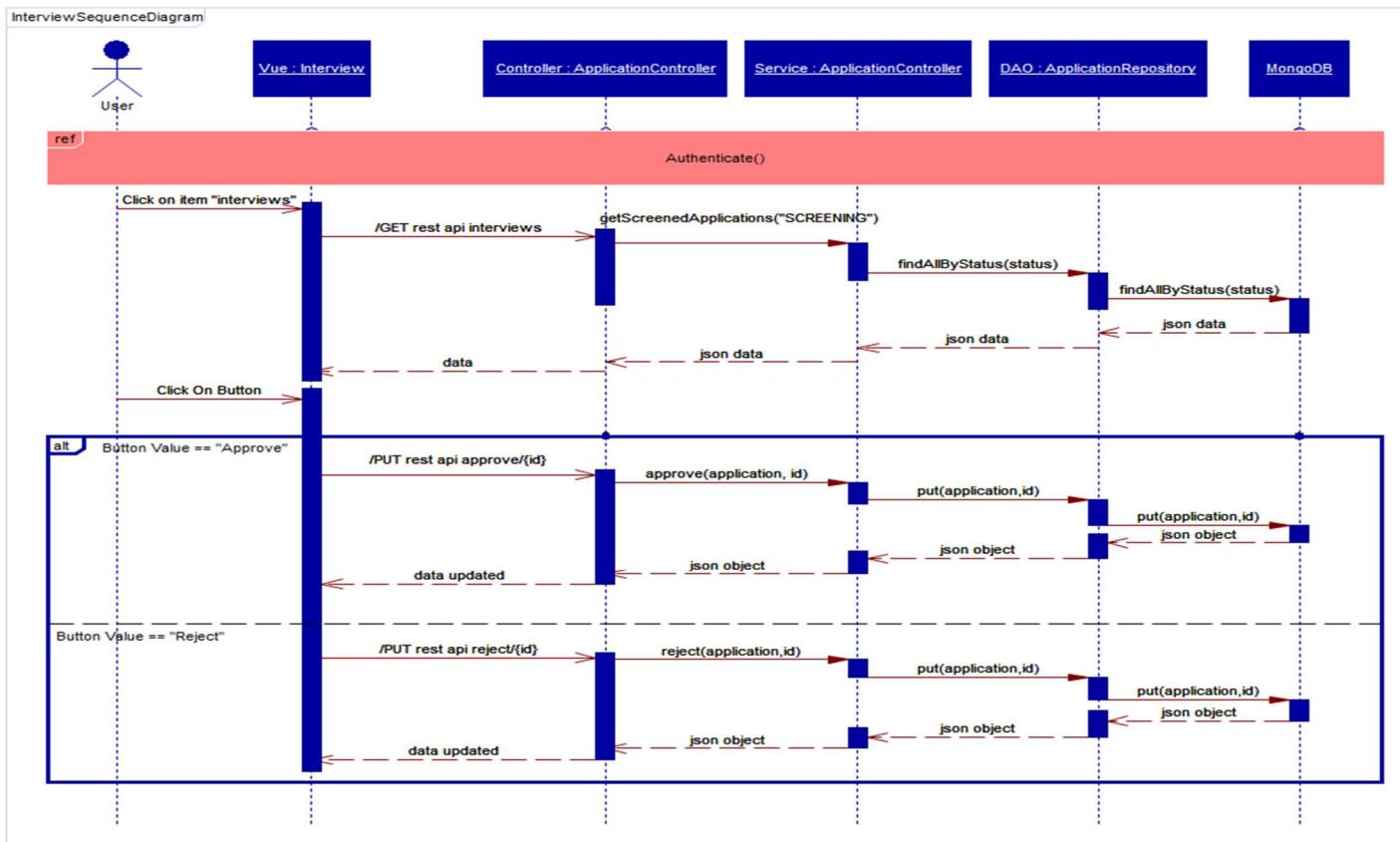


Figure 27 Object Sequence Diagram of "Applications Management" use case

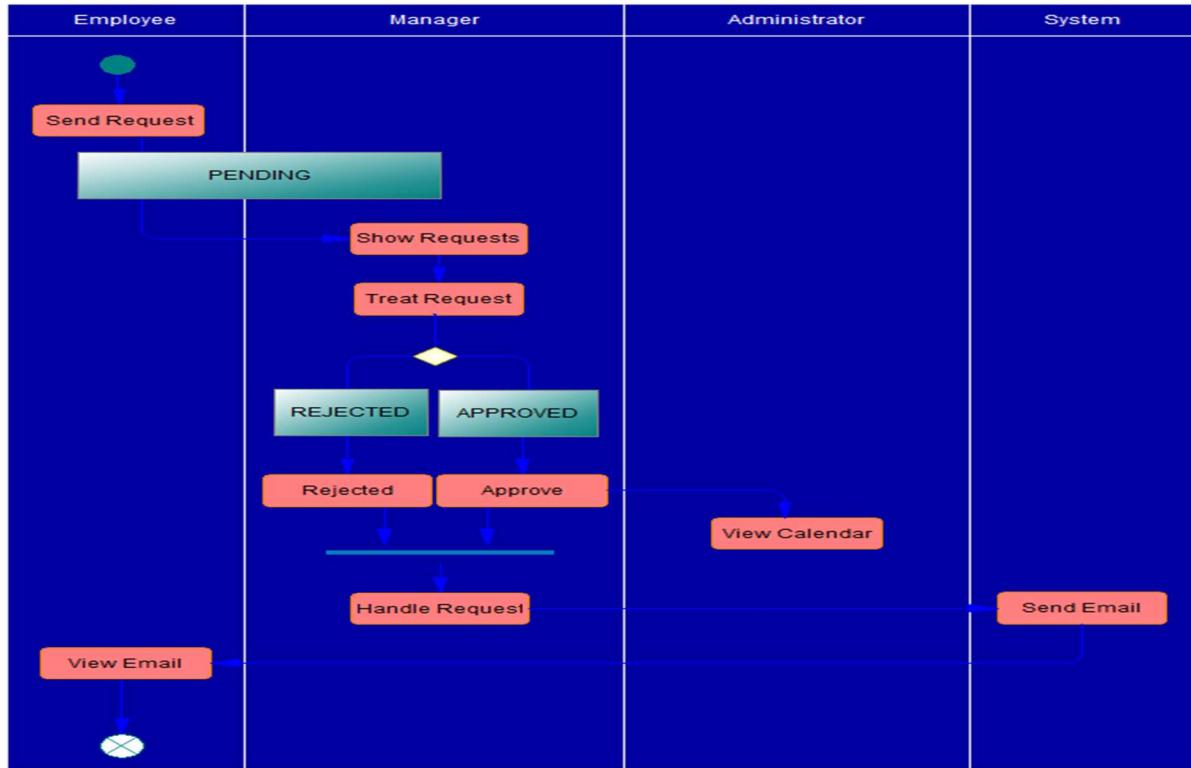
Once the manager has confirmed the screening phase of the different profiles, the manager is supposed to consult the interview interface in which he has a view on the list of confirmed candidates and a meeting room. The manager can send an invitation to each candidate to validate or reject in a final way. This scenario can only be validated after being authenticated.

In the next part, we present to the activity diagram.

## 2. Activity Diagrams

An activity diagram is an UML diagram that describes the workflow.

The following figure illustrates the activity diagram of “Manage Leave Request”.



**Figure 28 Leave Management Activity Diagram**

Once the employee initiates a leave request, the request is in pending status.

The manager consults the list of requests and will process it. If the manager has not validated the request, it will go to a reject status; an email will be sent to the employee to notify him that his request has not been approved. If the manager has validated the request, it goes to the approved status and in the same way, an email will be sent to the employee indicating that the request has been validated. The administrator on his part can view the calendar of validated requests.

Next, we present some different graphical interfaces that we find interesting.

### III. Implementation

In this part, we present some user interfaces of this sprint.

#### 1. Leave Request Workflow

The following interface presents and explain the workflow of sending new leave request

The figure consists of three side-by-side screenshots of a leave request form. Each screenshot shows a header with 'Absences', 'NEW REQUEST', 'LIST REQUESTS', and 'DRAFT REQUESTS'. 1. The first screenshot shows the 'Send Leave Request' section with three steps: 'Motif' (selected as 'Legal Paid Leave'), 'Period' (selected as 'May 24, 2022' to 'May 27, 2022'), and 'Comments' (empty). 2. The second screenshot shows the same steps but with the 'Comments' field populated with 'This Is A Comment ...'. 3. The third screenshot shows the 'Comments' field populated with 'This Is A Comment ...' and includes a 'Draft Mode' toggle switch and a 'CONFIRM' button.

Figure 29 Leave Request -Motif    Figure 31 Leave Request-Interval    Figure 30 Leave Request -Comment

## 2. Follow-up Requests Calendar

The next interface presents the follow-up of absence requests in a calendar.

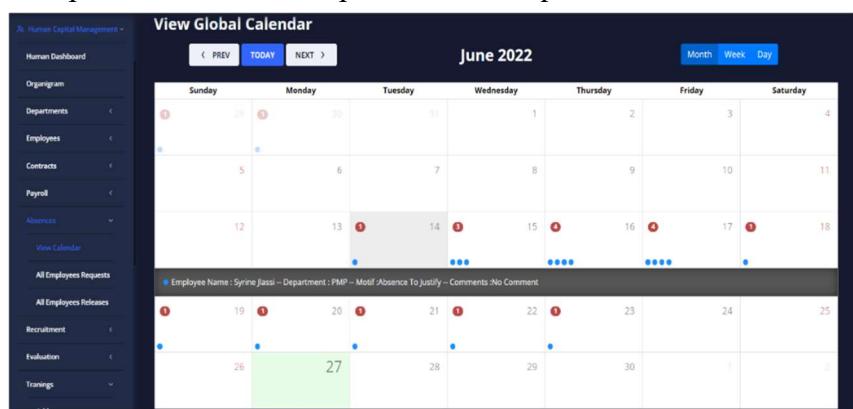


Figure 32 Leaves Calendar

## 3. Create a meeting

The next interface presents the meeting room.

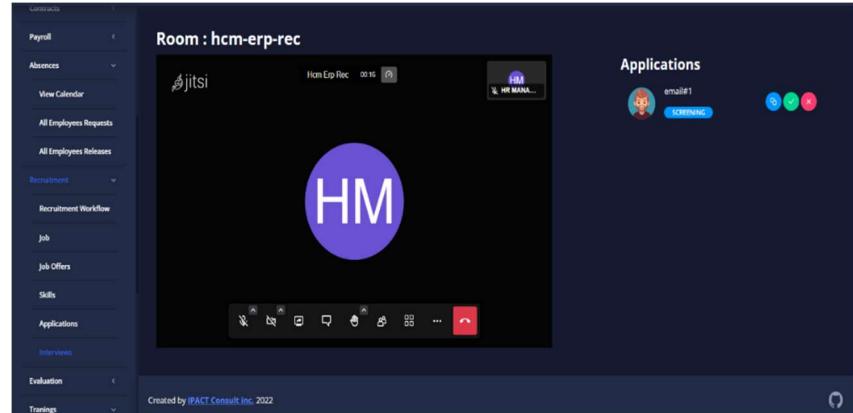


Figure 33 Interview

## Conclusion

Throughout this sprint, we presented the backlog for this sprint, followed by the overall use case diagram and the database modeling diagram. Finally, we demonstrated some graphical interfaces. In the next chapter, we will discuss about the Sprint 3 which includes the requirements for our next deliverable.

# Chapter 5 – Sprint 3: Skills & Training Management

## Introduction

This chapter covers the third sprint in which we present the requirements analysis and specification followed by the conceptual study and implementation phase. The objective in this sprint is to develop the functionalities for employee skills management and training management.

### I. Analyze and Specification of needs

#### 1. Sprint Backlog

The next table illustrates the sprint backlog of this sprint.

ID	User Story	Task ID	Task	Estimation
3.1	As a user, I want to create a new skill.	3.1.1	Implement and test service in backend	1 day
		3.1.2	Consume backend service in frontend	
		3.1.3	Create interface create skill	
		3.1.4	Test feature	
3.1	As a user, I want to edit an existent skill.	3.1.1	Implement and test service in backend	1 day
		3.1.2	Consume backend service in frontend	
		3.1.3	Create interface edit skill	
		3.1.4	Test feature	
3.1	As a user, I want to display all skills.	3.1.1	Implement and test service in backend	2 days
		3.1.2	Consume backend service in frontend	
		3.1.3	Create interface display skills (grid2x2, grid3x3, table)	
		3.1.4	Test feature	
3.1	As a user, I want to assign a skill to an employee	3.1.1	Implement and test service in backend	1 day
		3.1.2	Consume backend service in frontend	
		3.1.3	Add field employee	
		3.1.4	Test feature	

<b>3.1</b>	As a user, I want to archive a skill	3.1.1	Implement and test service in backend	1 day
		3.1.2	Consume backend service in frontend	
		3.1.3	Add Button Archive into data skills.	
		3.1.4	Test feature	
<b>3.1</b>	As a user, I want to restore a skill	3.1.1	Implement and test service in backend	1 day
		3.1.2	Consume backend service in frontend	
		3.1.3	Add button Restore in data history skills.	
		3.1.4	Test feature	
<b>3.1</b>	As a user, I want to export skills as pdf file.	3.1.1	Install pdf dependency	3 days
		3.1.2	Create function in type script component	
		3.1.3	Add Action in Pdf Button.	
		3.1.4	Test feature	
<b>3.1</b>	As a user, I want to export skills as csv file	3.1.1	Install csv dependency	1 day
		3.1.2	Create function in type script component	
		3.1.3	Add Action in csv button	
		3.1.4	Test feature	
<b>3.1</b>	As a user, I want to filter data	3.1.1	Create filter function in type script	2 days
		3.1.2	Add filters in html component	
		3.1.3	Test filters	
<b>3.1</b>	As a user, I want to delete skill	3.1.1	Implement and test service in backend	1 day
		3.1.2	Consume backend service in frontend	
		3.1.3	Test feature	
<b>3.1</b>	As a user, I want to search a skill	3.1.1	Add search field in type script component	12 hours
		3.1.2	Call search attribute in html component	
<b>3.2</b>	As a user, I want to create a training	3.2.1	Implement and test service in backend	2 days
		3.2.2	Consume backend service in frontend	
		3.2.3	Create Training interface	
		3.2.4	Test feature	
<b>3.2</b>		3.2.1	Implement and test service in backend	1 day
		3.2.2	Consume backend service in frontend	

	As a user, I want to edit a training	3.2.3 3.2.4	Create Edit Interface Test feature	
3.2	As a user, I want to display all trainings	3.2.1	Implementation and testing service backend	1 day
		3.2.2	Consume backend service in frontend	
		3.2.3	Create list training interface	
		3.2.4	Testing feature	
3.2	As a user, I want to assign a training to employee	3.2.1	Implement and test service in backend	1 day
		3.2.2	Consume backend service in frontend	
		3.2.3	Add employee field in create training	
		3.2.4	Test feature	
3.2	As a user, I want to archive a training	3.2.1	Implement and test service in backend	12 hours
		3.2.2	Consume backend service in frontend	
		3.2.3	Add Action in training list	
		3.2.4	Test Feature	
3.2	As a user, I want to restore a training	3.2.1	Implement and test service in backend	12 hours
		3.2.2	Consume backend service in frontend	
		3.2.3	Add Action in training list	
		3.2.4	Test Feature	
3.2	As a user, I want to delete a training	3.2.1	Implement and test service in backend	12 hours
		3.2.2	Consume backend service in frontend	
		3.2.3	Add Action in training list	
		3.2.4	Test Feature	
3.2	As a user, I want to display training history	3.2.1	Implement and test service in backend	1 day
		3.2.2	Consume backend service in frontend	
		3.2.3	Create training list interface	
		3.2.4	Test Feature	
3.2	As a user, I want to	3.2.1	Install pdf dependency	3 days
		3.2.2	Create function in type script component	

	export data as pdf file	3.2.3 3.2.4	Add Action in Pdf Button Test Feature	
3.2	As a user, I want to export data as csv file.	3.2.1	Install csv dependency	1 day
		3.2.2	Create function in type script component	
		3.2.3	Add Action in csv button	
		3.2.4	Test feature	
3.2	As a user, I want to filter trainings	3.2.1	Create filter function in type script	2 days
		3.2.2	Add fields in html component	
		3.2.3	Test Features	
3.2	As a user, I want to search training	3.2.1	Create filter functions in type script	12 hours
		3.2.2	Add fields in html component	
		3.2.3	Test Features	

Table 8 Sprint 3 Backlog

## 2. General Use Case Diagram of Sprint Three

The following figure present and illustrates the use case diagram of third sprint.

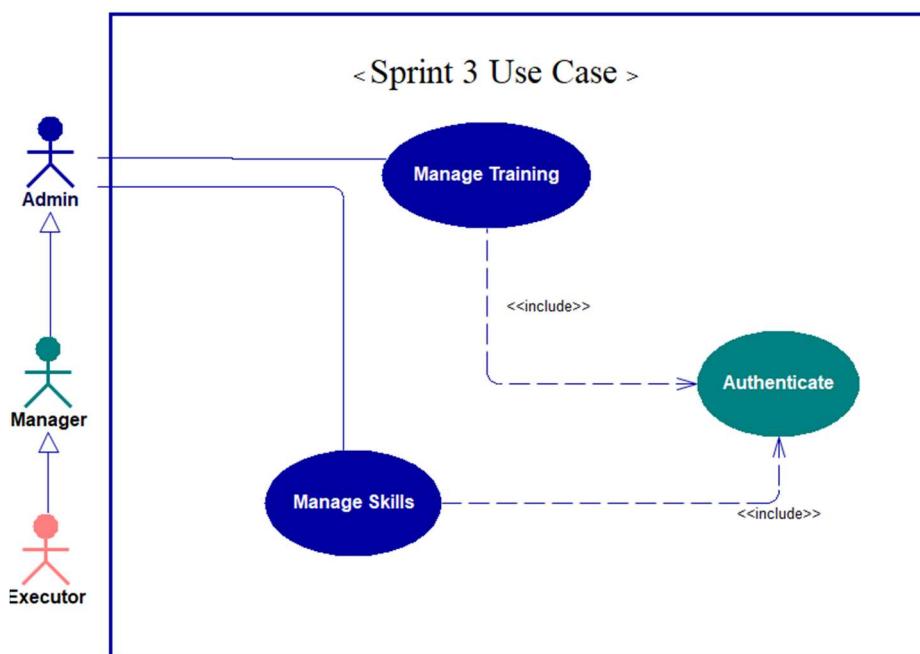


Figure 34 Use Case Diagram Sprint 3

The following diagram illustrates the refinement of use case “Manage Training”.

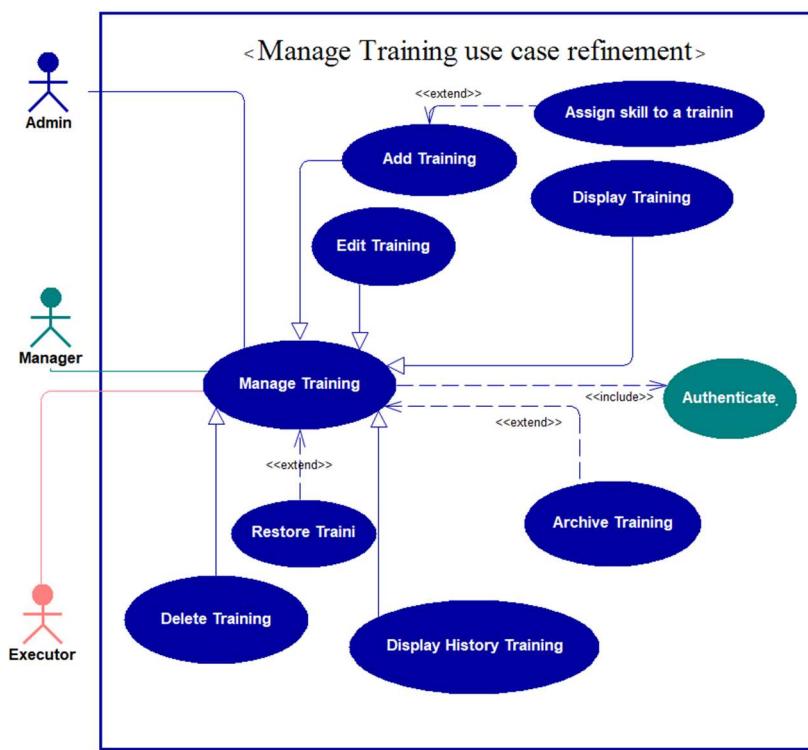


Figure 35 Refinement of use case "Manage Training"

## II. Conceptual Study

### 1. System Sequence Diagrams

The following figure illustrates the sequence diagram of “create Training” use case.

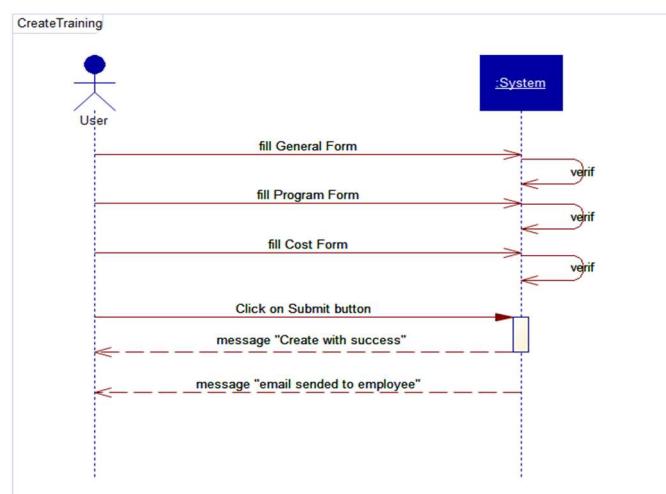


Figure 36 System Sequence Diagram of "Create Training" use case

## 2. Activity Diagram

The following figure illustrates the activity diagram of “create training” use case.

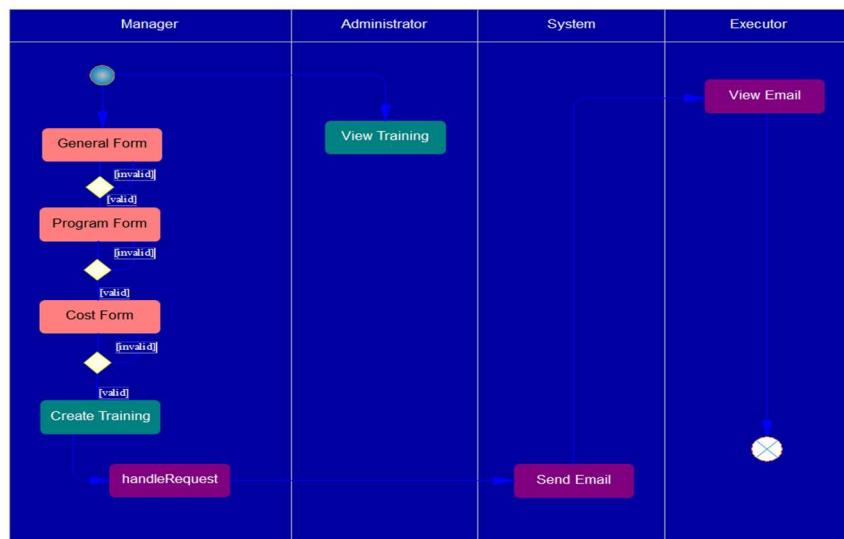


Figure 37 Activity Diagram of "Create Training" use case

To create a new training, the manager must follow a detailed process. He should start with the general information, then the program and then the cost information. The manager must assign an employee to this training. Once the process is completed and validated, an email is sent to the employee to inform him of the training. The administrator on his part can follow the training schedule.

## III. Implementation

### 1. View employee skills

The following figure presents the interface “View Employee Skills”.

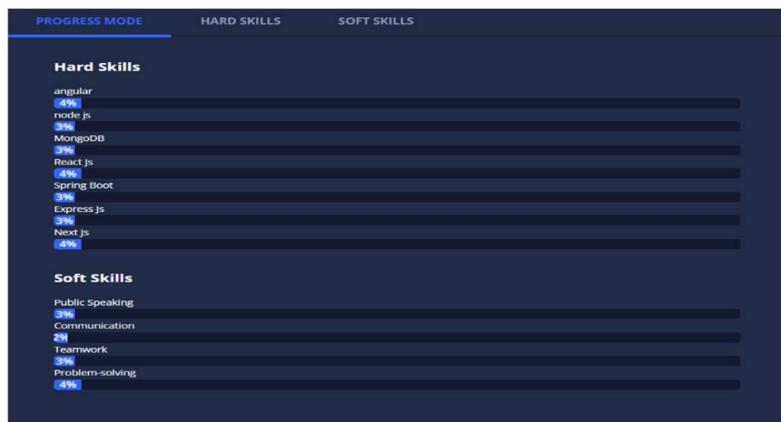


Figure 38 Employee Skills - Progress Mode

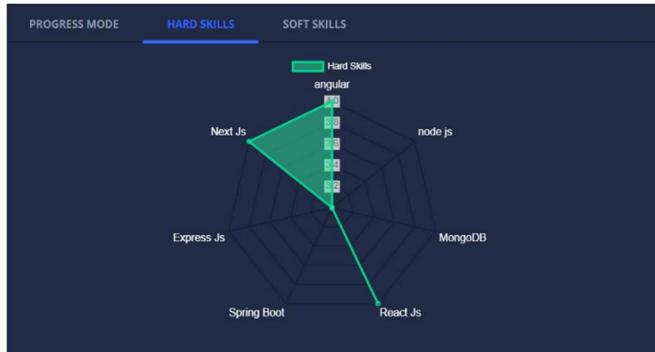


Figure 40 Employee Hard Skills - Radar

## 2. Training creation workflow

The following figure illustrates the workflow of creating a new training.



Figure 39 Employee Soft Skills - Radar

A screenshot of a "New Training" form. On the left is a sidebar with navigation links like "All Employees Requests", "Recruitment", "Job", "Skills", etc. The main area shows a progress bar with three steps: 1. General Info (highlighted), 2. Program, and 3. Cost. The "General Info" step contains fields for "Object" (dropdown), "Number Of Hours" (1000), "Start Date" (PICK Date), "End Date" (PdI Date), and "Description" (richtext editor). Buttons for "PREV" and "NEXT" are at the bottom.

Figure 41 Create Training - General Info

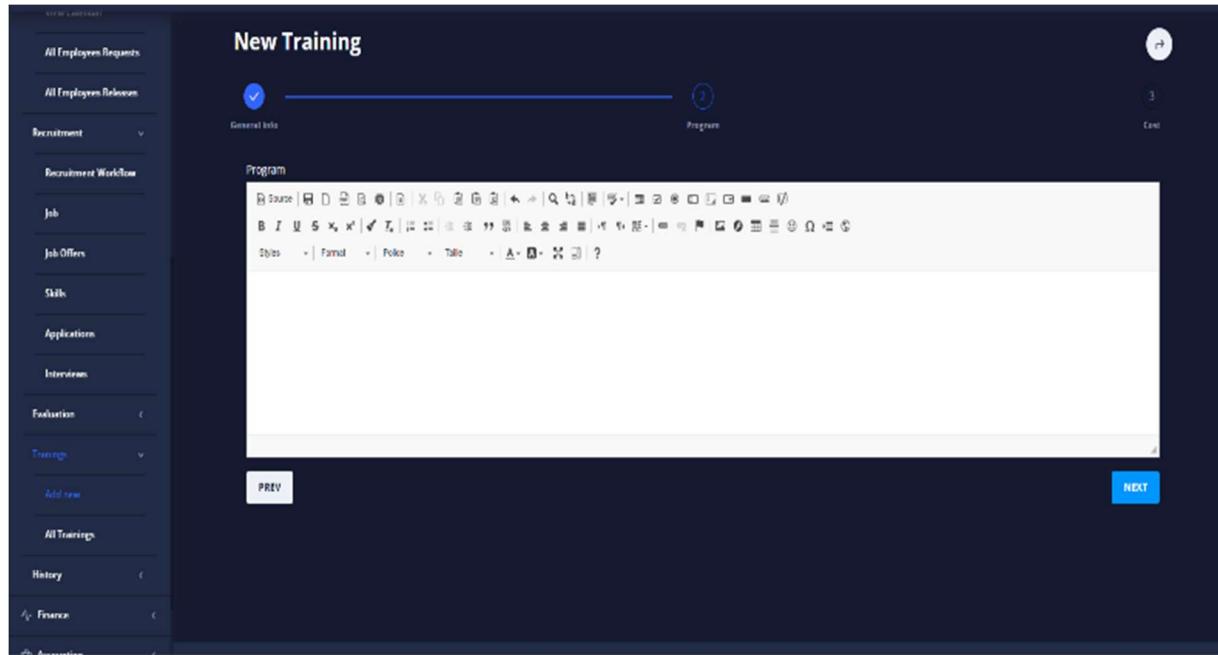


Figure 43 Create Training - Program

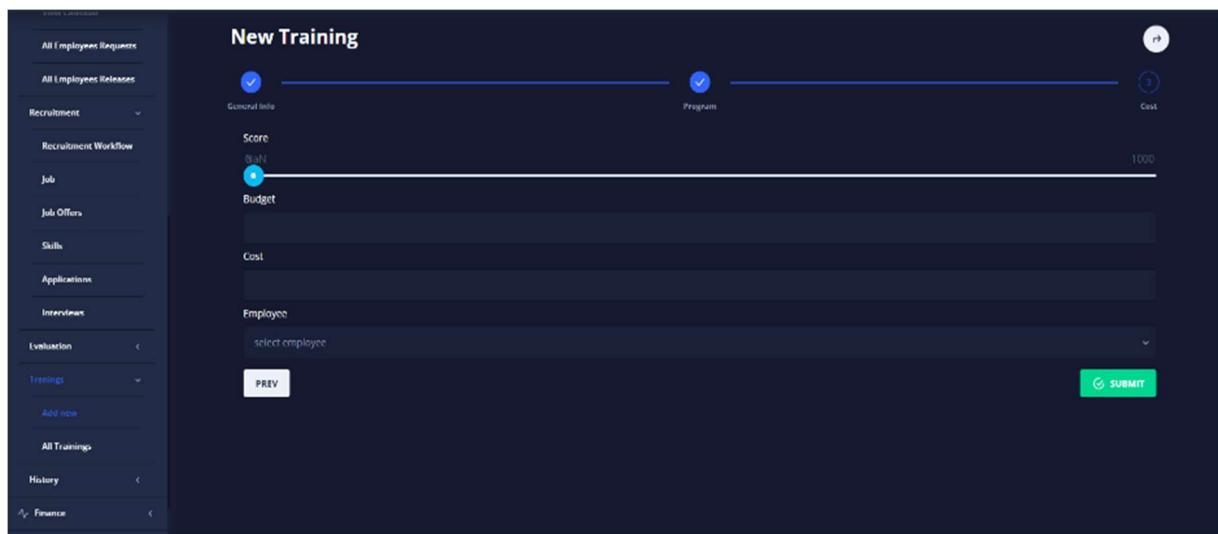


Figure 42 Create Training - Cost

## Conclusion

This sprint was dedicated to the presentation of the backlog, the use case diagram related to this sprint followed by the conceptual study in which we have presented the database modeling diagram and some sequence and activity diagrams. At the end of this chapter, we present the different graphical interfaces. In the next chapter, we will talk about the next sprint which includes the deliverable requirements.

# Chapter 6 – Sprint 4: Payroll, Appraisals & Expenses Management

## Introduction

This chapter presents the fourth sprint in which we present the requirements analysis and specification, followed by the conceptual study and implementation phase. The target of this sprint is to develop the payroll functionality and the management of employee's appraisals.

### I. Analyze and Specification of needs

#### 1. Sprint Backlog

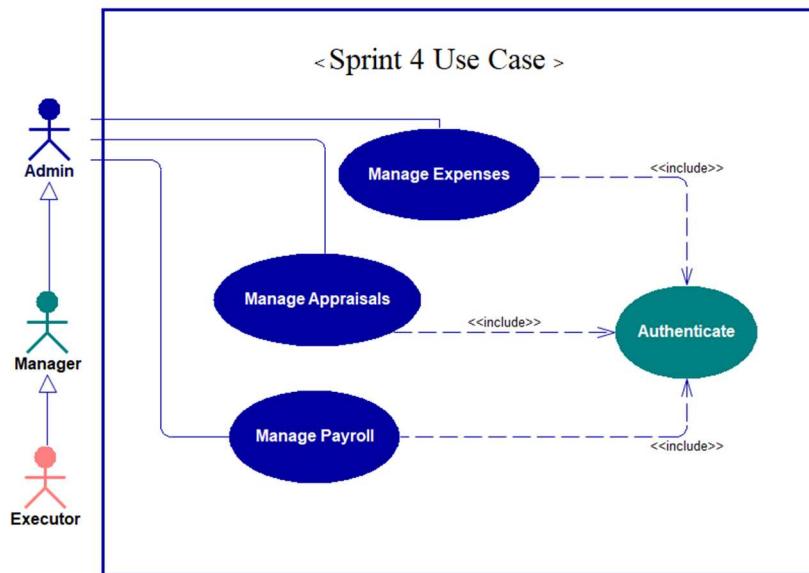
ID	User Story	Task ID	Task	Estimation
4.1	As a user, I want to create a new evaluation test.	4.1.1	Implement and test service in backend	1 day
		4.1.2	Consume service in frontend	
		4.1.3	Create Evaluation Test Interface	
		4.1.4	Testing Feature	
4.1	As a user, I want to assign an employee to an evaluation Test	4.1.1	Implement and test service in backend	12 hours
		4.1.2	Consume service in frontend	
		4.1.3	Add employee field	
		4.1.4	Testing Feature	
4.1	As a user, I want to receive an email after assign.	4.1.1	Add Email Dependency in pom.xml file	2 days
		4.1.2	Create Email Template in backend	
		4.1.3	Implement and test service in backend	
4.1	As a user, I want to display all evaluations	4.1.1	Implement and test service in backend	1 day
		4.1.2	Consume backend service in frontend	
		4.1.3	Create interface list evaluations	
		4.1.4	Testing Feature	

<b>4.1</b>	As a user, I want to archive evaluation	4.1.1	Implement and test service in backend	12 hours
		4.1.2	Consume 1service in frontend	
		4.1.3	Create Action Button in evaluation list	
		4.1.4	Test Feature	
<b>4.1</b>	As a user, I want to restore evaluation	4.1.1	Implement and test service in backend	12 hours
		4.1.2	Consume backend service in frontend	
		4.1.3	Create Action Button in evaluation list	
		4.1.4	Test Feature	
<b>4.1</b>	As a user, I want to delete evaluation	4.1.1	Implement and test service in backend	12 hours
		4.1.2	Consume backend service in frontend	
		4.1.3	Create Delete Button in list	
		4.1.4	Test Feature	
<b>4.1</b>	As a user, I want to search evaluation	4.1.1	Add search attribute in type script component	6 hours
		4.1.2	Read this variable in html component	
<b>4.1</b>	As a user, I want to filter evaluation	4.1.1	Add filters functions in type script component	1 day
		4.1.2	Call these functions in html component	
<b>4.1</b>	As a user, I want to export data as pdf file	4.1.1	Call dependency in type script component	1 day
		4.1.2	Add source code related to export as pdf	
		4.1.3	Call pdf attribute in html component	
<b>4.1</b>	As a user, I want to export data as csv file	4.1.1	Call dependency in type script component	1 day
		4.1.2	Add source code related to export as csv	
		4.1.3	Create button and call csv function in html component.	
<b>4.1</b>	As a user, I want to follow my evaluations.	4.1.1	Implement and test service in backend	2 days
		4.1.2	Consume backend service in frontend	
		4.1.3	Create component	
		4.1.4	Test Feature	

**Table 9 Sprint 4 Backlog**

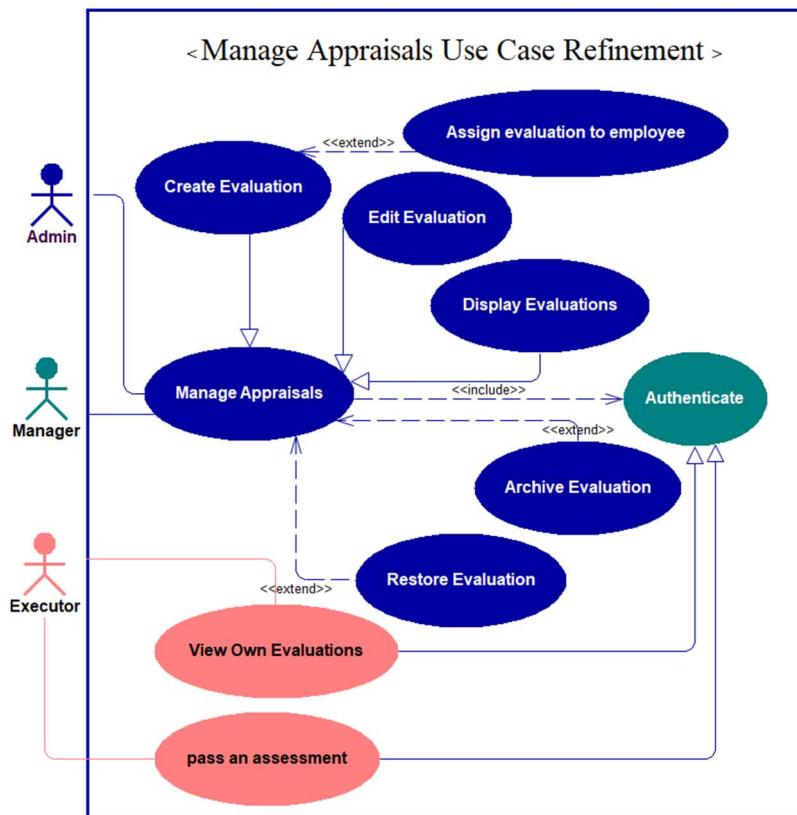
## 2. General Use Case Diagram of Sprint Four

The following figure illustrates and presents the use case diagram of fourth sprint.



**Figure 44 Sprint 4 Use Case**

The following figure illustrates the refinement of use case “Appraisals Refinement”.



**Figure 45 Refinement of use case "Appraisals Management"**

## II. Conceptual Study

### 1. System Sequence Diagram

The following figure illustrates the system sequence diagram of “Create Evaluation” use case.

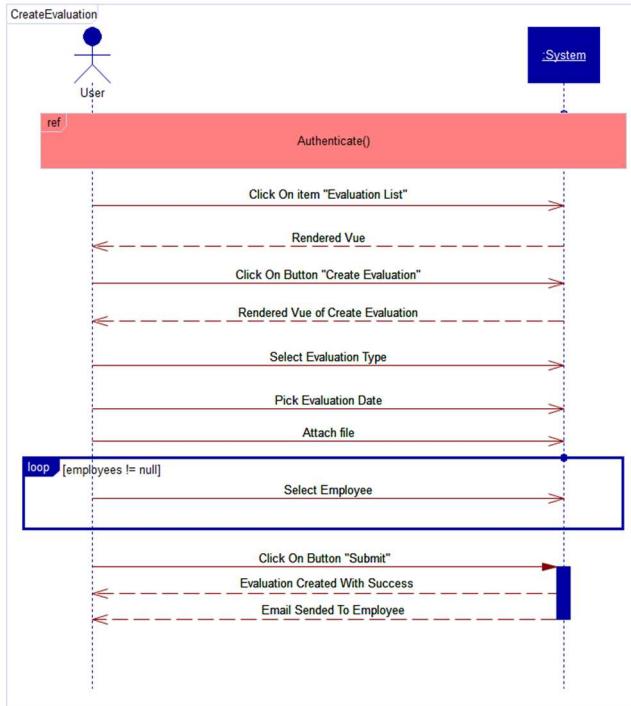


Figure 46 System Sequence Diagram of use case "Create Evaluation"

### 2. Object-Oriented Sequence Diagram

The following figure illustrates the object sequence diagram of “Create Evaluation” use case.

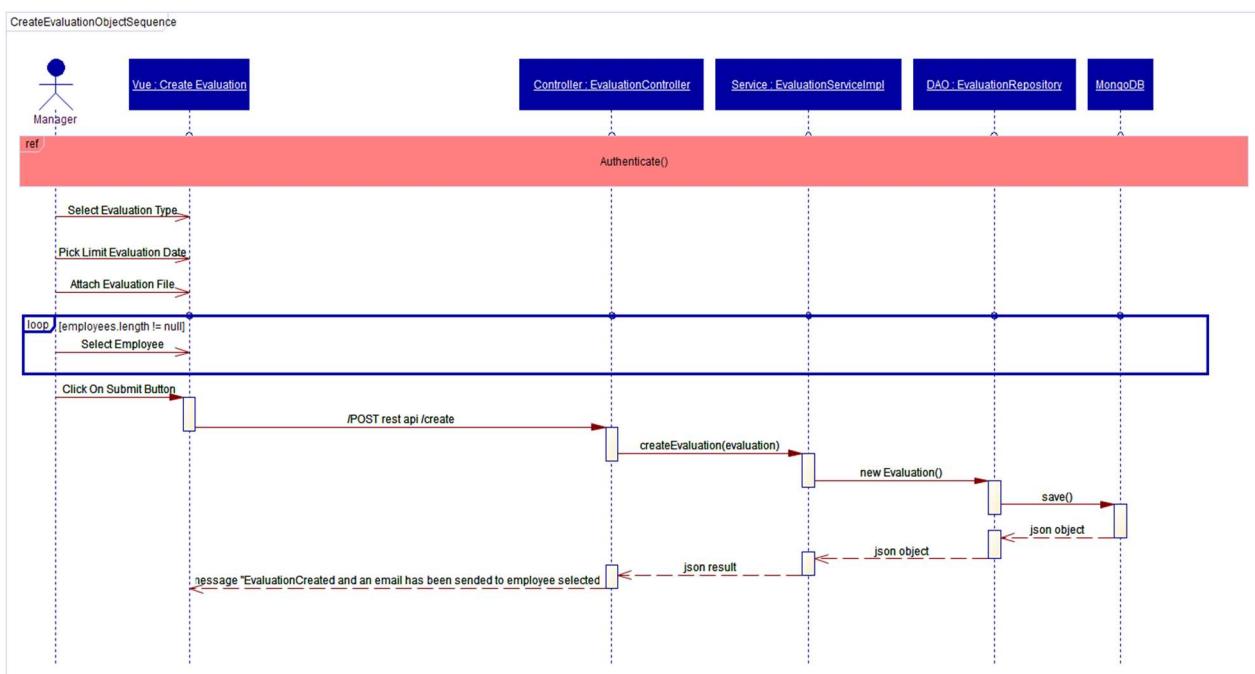


Figure 47 Object Sequence Diagram of use case "Create Evaluation"

### III. Implementation

At this level, we present some user interfaces of this sprint.

#### 1. Consult The Pay Sheet

The following figure illustrates the pay sheet of an employee.

Pay Sheet Employee			
Company Name : Ipcact Consult			Pay Sheet
Company Address : ipactconsult@gmail.com			Matriculate N CNSS/CNRPS : ERP-EMZ-HR-3
N CNSS : 00112233IC			Marital Status : Married
Position : ROLE_HR_MANAGER		Employee Name : Amine Laribi	Number of kids : 2
Department : HR		Employee Email : aminlaribi18@gmail.com	
N	Payroll element	Percentage	Amount
10	Base Salary		1200000 DT
100	Conventional Bonus		100000 DT
150	Non-Conventional Bonus		100000 DT
Total Gross			1400000 DT
200	CNSS	9,18%	128520 DT
600	Net Imposed		861980 DT
700	IRPP		456000 DT
800	Net Salary		1033980 DT
900	CNAM		28000 DT
			Net to pay: 1061980DT

Figure 48 Employee Pay Slip

#### 2. Create an evaluation test

The following figure illustrates the creation of an evaluation test with sending an email to employee.

Figure 49 Create Evaluation Test

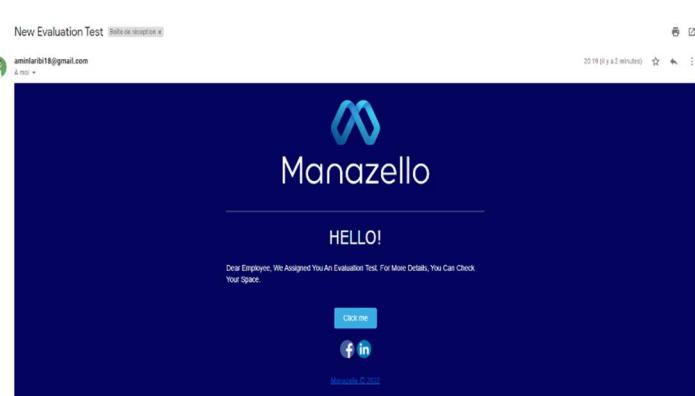


Figure 50 Email Confirmation Of Evaluation

### Conclusion

This sprint was dedicated to the presentation of the backlog, the use case diagram related to this sprint followed by the conceptual study to present the database modeling diagram and some sequence diagrams. At the end of this chapter, we present the different graphical interfaces. In the next chapter, we will talk about the next sprint which includes the deliverable requirements.

# Chapter 7 – Sprint 5: Administration

## Management & Statistics Generation

### Introduction

This chapter presents the fifth sprint in which we present the requirements analysis and specification, followed by the design and implementation phase. The target of this sprint is to develop the payroll functionality and the management of employee evaluations.

### I. Analyze and Specification of needs

#### 1. Sprint Backlog

The following table illustrates the sprint backlog of sprint 5.

ID	User Story	Task ID	Task	Estimation
<b>5.1</b>	As a user, I want to send a rental request	5.1.1	Implement and test service in postman that permits to user to send a rental request	1 day
		5.1.2	Consume rest api in frontend	
		5.1.3	Create rental request process	
<b>5.1</b>	As a user, I want to check the list of requests	5.2.1	Implement and test service in postman that permits to display list of requests	2 days
		5.2.2	Consume rest api in frontend	
		5.2.3	Create requests list interface	
<b>5.1</b>	As a user, I want to see request details	5.1.1	Implement and test service in postman that permits to see request details	1 day
		5.1.2	Consume rest api	
		5.1.3	Create request details interface	
<b>5.1</b>	As a user, I want to decline a request	5.1.1	Implement and test service in postman that permits to decline a request, sending e-mail, and consuming in frontend	3 hours

5.1	As a user, I want to approve a request	5.1.1	Implement and test service in postman that permits to approve a request, sending e-mail, and consuming in frontend	3 hours
5.1	As a user, I want to change the status of the request to in progress	5.1.1	Implement and test service in postman that permits to change to in progress a request	3 hours
5.1	As a user, I want to guarantee access to an employee	5.1.1	Implement and test service in postman that permits to give access to an employee	1 day

Table 10 Sprint Backlog 5

## 2. General Use Case Diagram of Sprint Five

The following figure illustrates the use case diagram of fifth sprint.

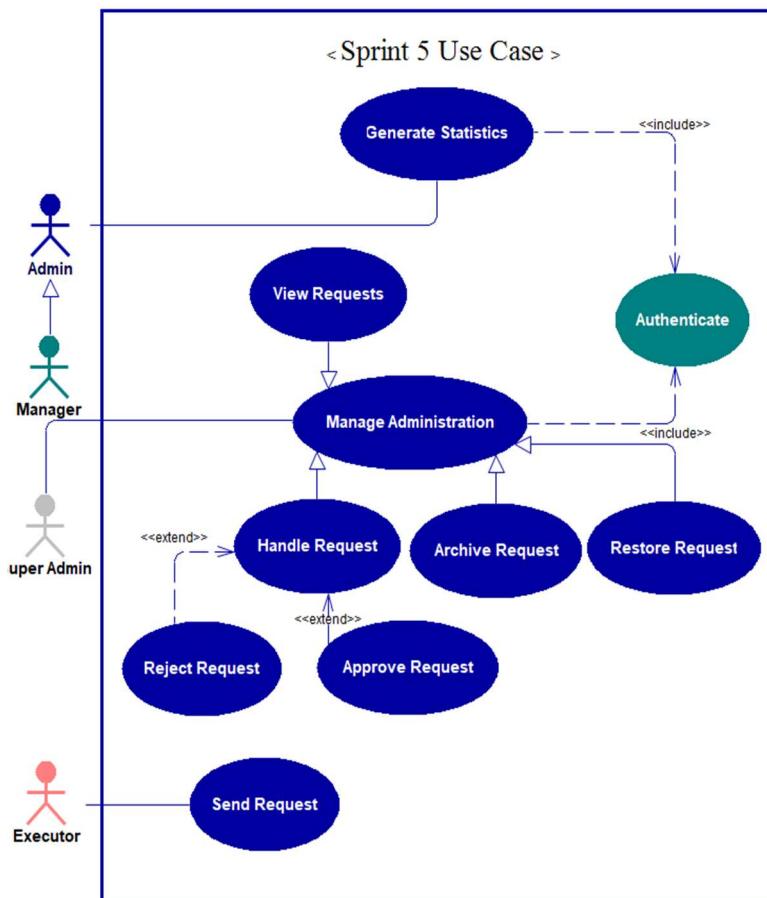


Figure 51 Sprint 5 Use Case

## II. Conceptual Study

### 1. System Sequence Diagrams

The following figure illustrates the sequence diagram of “send request” use case.

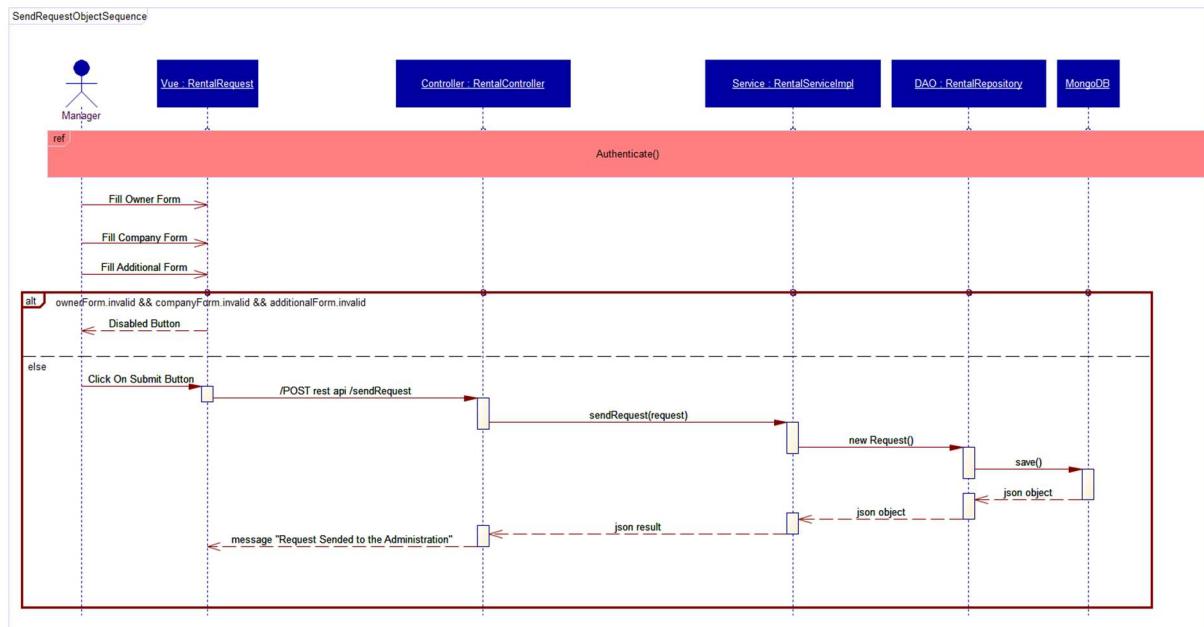


Figure 52 Send Request Object Sequence

### 2. Activity Diagrams

The following figure illustrates the activity diagram of “Handle Request” use case.

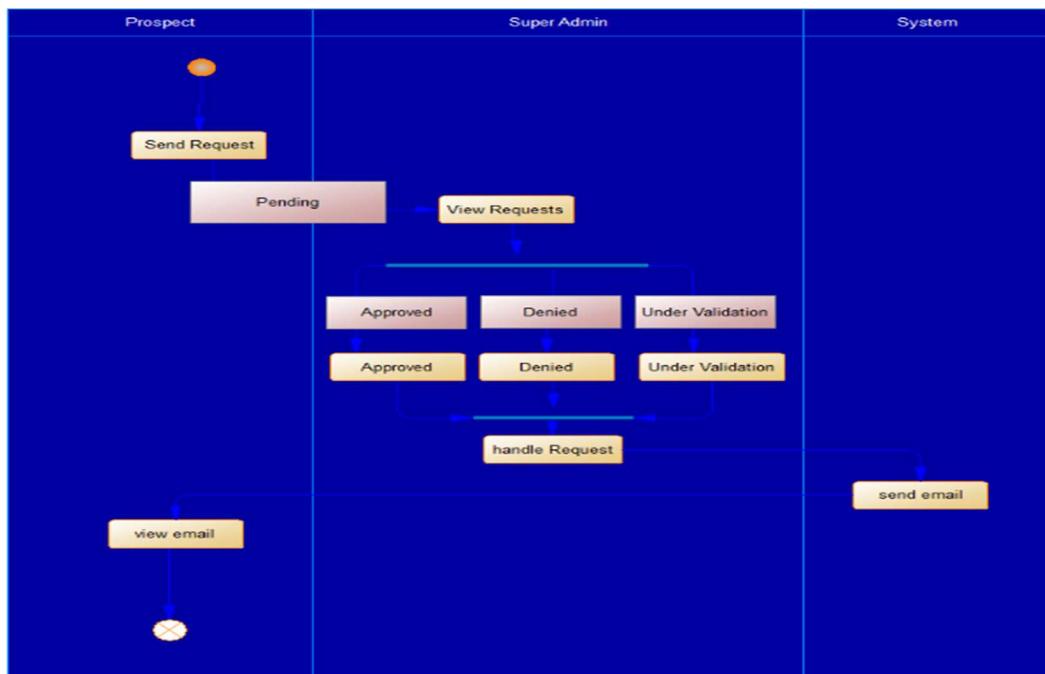


Figure 53 Activity Diagram of use case "Handle Request"

### III. Implementation

#### 1. Send Request

The following figure illustrates send new request interface.

The screenshot shows a dark-themed web form titled "Be Our Customer". On the left, there's a "Welcome" message: "If you want to become a customer in our application, please fill in this form, we will do our best to answer you as soon as possible." The form is divided into sections: "Owner" and "Company". The "Owner" section contains fields for "First Name", "Last Name", "Email address", and "Phone Number". The "Company" section contains fields for "First Name", "Last Name", "Email address (Optional)", and "Phone Number (Optional)". Below these are "Additional" fields. At the bottom are two large blue arrows pointing left and right, and a copyright notice: "All Rights Reserved © 2022 – 2023".

Figure 54 Send Request

#### 2. Human Capitals Dashboard

To ensure a lot of technical heterogeneity and to improve the quality of our project, we opt for the integration of a personalized dashboard that summarizes the activity of the human resources department in terms of estimated training costs compared to the actual budget, the number of employees per department, the number of men and women per department, the rate of employee absenteeism, etc.

The following figure shows the statistics of our application, integrated by the Power BI tool.

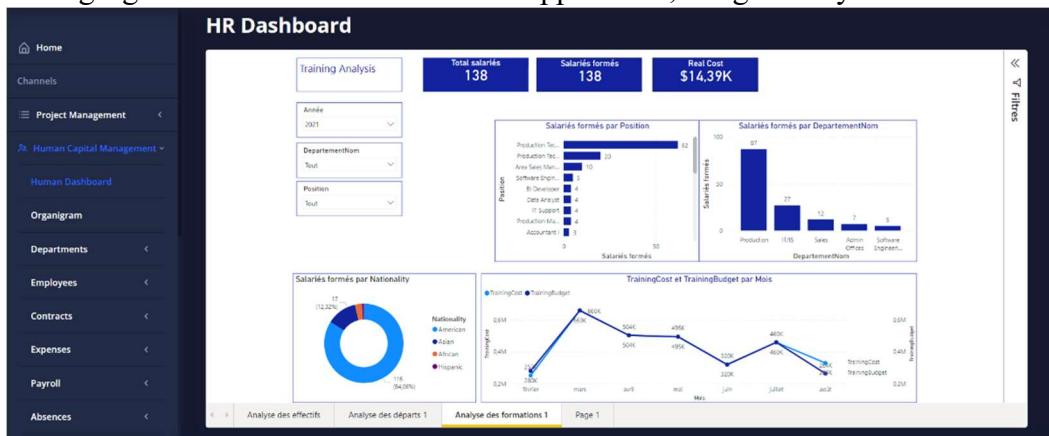


Figure 55 Human Resources Dashboard

### Conclusion

This sprint was dedicated to the presentation of the backlog, the use case diagram related to this sprint followed by a conceptual study. At the end of this chapter, we present the different graphical interfaces. In the next chapter, we will talk about the next sprint which includes the deliverable requirements.

# Chapter 8: SEO & Deployment

## Introduction

In this last chapter, we are interested in the part of referencing in which we make research to have an idea on the words necessary to have a better visibility of our application and the deployment of the application.

### I. Search Engine Optimization

The natural referencing contains the practices used by search engines which grades and guides the results they take back to users who have applied for the demands.

#### 1. Study of existing

At this stage, we do an inspection of the competitors to examine the keywords submitted by the user through the searches. We quote:

			
Title	Payroll Services by OnPay   Full-Service Online Payroll, HR,	ZenHR   HR   Software  HRIS   HRMS   HR System	Payroll, Benefits, HR, and Compliance All in One Place
Description	Simplify your payroll, taxes, HR, and benefits. OnPay top-rated online payroll services bring it all together.	ZenHR is a cloud-based Human Resources Management System, HRMS, localized to manage onboarding, offboarding, payroll, attendance and time tracking, performance and more	Run your business with confidence with Justworks. Get simple software + expert support for payroll, benefits, HR, and compliance.

Hierarchization	Payroll and HR that move you in the right direction Payroll HR Benefits Rated “Editors’ Choice” Top Rated Best Value Every business. Every milestone. Designed to be accurate for everyone	Technology with the power to transform HR Tackling your HR needs from Acquire to Retire. 53 60 60 Let our customers do the talking	Run Your Business with Confidence Simple software + expert support for payroll, benefits, HR, and compliance. Payroll Pay your people with ease. Benefits
-----------------	--	---	---

Table 11 Optimization on-page

## 2. Keywords Validation

The following figures presents the validated keywords mentioned by Google Trends.

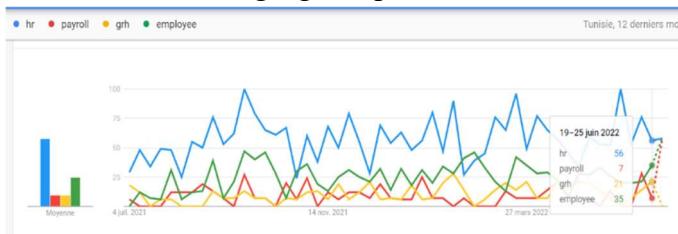


Table 12 KeyWords Validation - Part 1



Table 13 Keywords Validation - Part 2

The words validated are HR, Employee, GRH (English: HRM [Human Resources Management]) and recruitment.

## 3. Implement a SEO strategy

### a. Meta-Title

The following figure illustrates the rate given the meta-title.



Figure 56 Meta-title

## b. Meta-description

The following figure illustrates the rate given the meta-description.

Audit SEO en ligne pour mieux optimiser et référencer son site web

Audit de l'URL : <https://manazello-client-henna.vercel.app/#/> (07/07/2022 - 13:45:36)

Titre Meta Description Meta Keywords URL Structure H1 Contenu Textuel Autres Contenus Popularité Données Techniques

Le contenu de votre balise Meta Description est le suivant :

**Manazello is ERP Software. It contain servel parts (CRM, HR, PM, CmKg, RL, ACC, FIN, PIM).HR make to manage employees including their remuneration, skills, training, evaluation, ... .CRM ensure the customer loyalty.PM make to manage project, planning and tracking tasks.**

La balise "Meta Description" de votre page contient 270 caractères et 40 mots.

★★★★★

Votre description a une bonne taille. Bravo !

Code HTML détecté :

```
<meta name="description" content="Manazello is ERP Software. It contain servel parts (CRM, HR, PM, CmKg, RL, ACC, FIN, PIM).HR make to manage employees including their remuneration, skills, training, evaluation, ... .CRM ensure the customer loyalty.PM make to manage project, planning and tracking tasks.">
```

DigitalOcean® Cloud Hosting Deploy Faster & Scale Easier With an Application Server That Saves Your Team Time & Money!

Figure 57 Meta-description

## c. Meta-keywords

The following figure illustrates the rate given of the meta-keywords.

Audit SEO en ligne pour mieux optimiser et référencer son site web

Audit de l'URL : <https://manazello-client-henna.vercel.app/#/> (07/07/2022 - 13:45:36)

Titre Meta Description Meta Keywords URL Structure H1 Contenu Textuel Autres Contenus Popularité Données Techniques

Le contenu de votre balise meta Keywords est le suivant :

**ERP, Software, Project Management, Tasks, Tracking, Planning, Schedule, Planning, Customer relationships Management, Customer, Loyalty, Productivity, Employee, Training, Skills, Evaluations, Recruitment, Remuneration, Financial, Litigation and Recovery**

La balise "Meta Keywords" de votre page contient 251 caractères et 24 mots.

★★★★★

Votre balise a une bonne longueur (20 mots ou plus). Bravo !

DigitalOcean® Cloud Hosting Deploy Faster & Scale Easier With an Application Server That Saves Your Team Time & Money!

Figure 58 Meta-Keywords

## d. Customization of the 404 Error Page

The following figure illustrates the 404 page not found.

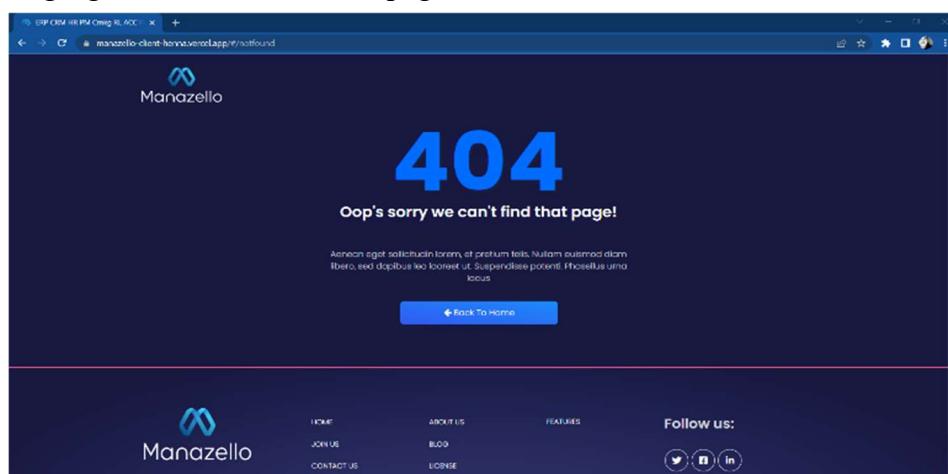


Figure 59 404 Not Found

### e. Setting Robots.txt file

The following figure present the robots.txt file which used to manage the traffic of crawlers on the website.

```
User-agent: *
Disallow:
Sitemap: https://manazello-client-app.vercel.app/#/sitemap.xml
```

Figure 60 robots.txt

## II. Deployment

After having developed the application, we now proceed to the deployment stage which consists in migrating the hardware or software elements during the realization phase to a progressive implementation phase of a new system.

It is a migration from a local server to a test server before deploying it to the end user.

### 1. Tools

The following table presents the work environment required to establish the deployment of the application.

Logo	Name	Description
 portainer.io	Portainer	Smart open-source management interface to visualize the containerized applications.
 ubuntu	Ubuntu	Operating system open source to run commands needed to deploy the application.
 docker	Docker	Technology that allows running containers separately and also image control.
 VMware	VMware	Virtualization Tool

Table 14 Deployment Tools

### 2. Explication

To ensure a correct scenario for the deployment we:

- Separate the application from the host environment using Docker.
- Bundle each microservices into a Docker container because we use a microservices architecture.

- Create a YAML file with name of dockercompose.yaml to define the microservices and to share multi-container docker applications.
- Build, maintain, deliver, and deploy the application using resource-isolated environments called Docker containers.
- Use Portainer as an interface that permits to easily manage volumes of container images in the Docker stack.

### 3. Realization

The following figure presents the Portainer interface which contains all images.

The screenshot shows the Portainer interface running in a Firefox browser window. The URL is `localhost:9000/#!/2/docker/containers`. The left sidebar has a 'LOCAL' section with 'Containers' selected. The main area is titled 'Container list' and shows a table of running containers:

Name	State	Quick Actions	Stack	Image	Created	IP Address	Published Ports	Ownership
angular	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	nginx.alpine	2022-07-04 12:56:18	172.19.0.2	[4200:80]	administrators
recrutement	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	openjdk:11	2022-07-04 12:56:12	172.18.0.2	[8071:8071]	administrators
hcm	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	openjdk:11	2022-07-04 12:56:11	172.18.0.3	[8060:8060]	administrators
spring-boot	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	openjdk:11	2022-07-04 12:56:11	172.18.0.4	[8070:8070]	administrators
eureka-server	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	openjdk:11	2022-07-04 12:56:09	172.18.0.5	[8761:8761]	administrators
gateway	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	openjdk:11	2022-07-04 12:56:09	172.18.0.6	[8088:8088]	administrators
mongo	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	mongo	2022-07-04 12:56:09	172.18.0.7	[27017:27017]	administrators
portainer	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	-	portainer/portainer-ce	2022-07-02 08:03:34	172.17.0.2	[8000:8000] [9000:9000]	administrators
recovery	stopped	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	openjdk:11	2022-07-04 12:56:11	-	-	administrators
crm	stopped	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	mnt	openjdk:11	2022-07-04 12:56:11	-	-	administrators

Figure 61 Portainer Interface

## General Conclusion

The goal of our final study project is to design and create a microservice of human resources management in which the objective is to offer a fluid management of employees, their remuneration, skills, leaves, etc in our application.

This internship was beneficial on several levels:

The first aspect is the teamwork: the methodology adopted for this project was consistent in the sense of being active and a good communicator within the team.

The second aspect is the management of the team: the fact of being not only a member of the team but also the person in charge of the graphic charter and the ergonomics aspect of the application, which is not an obvious thing to manage in the sense that all the members of the team must follow the same charter.

The third aspect is conflict management: conflicts are always present in any project, but the main thing is to have the sense to manage them and find a solution.

The fourth component is meeting deadlines: we are required to provide a deliverable at the end of each sprint.

The fifth part is the technical aspect: this internship allowed us to improve our skills and knowledge in writing code using good practices as well as to improve in everything that is application deployment.

The human resource part is still in need of improvement on two levels:

- On the strategic level, we can look for solutions that can improve the productivity of the employee.
- On the functional level, we can integrate the part of the prediction of the costs of the trainings as well as to analyze if this employee is productive or not.

# Netography

- [1] <https://www.capterra.fr/software/131882/gusto>
- [2] <https://www.capterra.fr/software/149405/justworks>
- [3] <https://www.capterra.fr/software/157129/zenhr>
- [4] <https://www.capterra.fr/software/167109/onpay>
- [5] <https://www.atlassian.com/fr/agile/kanban/kanban-vs-scrum>
- [6] <https://www.reactive-executive.com/processus-unifie-ou-methode-agile/>
- [7] <https://code.visualstudio.com/docs>
- [8] <https://www.jetbrains.com/help/idea/discover-intellij-idea.html>
- [9] <https://git-scm.com/>
- [10] <https://www.postman.com/>
- [11] <https://www.mongodb.com/products/compass>
- [12] <https://support.atlassian.com/trello/docs/what-is-trello/>
- [13] <https://www.datensen.com/data-modeling/moon-modeler-for-databases.html>
- [14] <https://www.powerdesigner.biz/EN/powerdesigner/powerdesigner-features.html>
- [15] <https://angular.io/guide/architecture>
- [16] [https://www.tutorialspoint.com/spring\\_boot/spring\\_boot\\_introduction.htm](https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm)
- [17] [https://www.tutorialspoint.com/spring\\_boot/spring\\_boot\\_introduction.htm](https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm)
- [18] <https://www.lemagit.fr/definition/Java>
- [19] <https://www.typescriptlang.org/>

## **Abstract**

This work is part of the engineering cycle final project in which we designed and developed a human resources module of an ERP.

**Keywords:** Human Resources, Collaborator, Employee, Department, Payroll, Recruitment, Training, Skills, Contracts, Angular, Spring Boot, MongoDB, Microservices, Deployment, Natural referencing.

## **Résumé**

Ce travail fait partie du projet de fin d'études de cycle d'ingénieur dans lequel nous avons conçu et développé un module de ressources humaines dans un ERP.

**Les mots clés :** Ressources Humaines, Collaborateur, Employé, Département, Paie, Recrutement, Formations, Compétences, Contrats, Angular, Spring Boot, MongoDB, Microservices, Déploiement, Référencement Naturel.

## **ملخص**

يعد هذا العمل جزءاً من مشروع دراسة نهاية دورة الهندسة حيث صمممنا وطورنا وحدة موارد بشرية في تخطيط موارد المؤسسات (ERP).

**الكلمات المفتاحية:** الموارد البشرية، الموظف، القسم، الرواتب، التوظيف، التدريب، المهارات، العقود

**ESPRIT SCHOOL OF ENGINEERING**

**www.esprit.tn - E-mail : contact@esprit.tn**

**Siège Social :** 18 rue de l'Usine - Charguia II - 2035 - Tél. : +216 71 941 541 - Fax. : +216 71 941 889

**Annexe :** 1-2 rue André Ampère - 2083 - Pôle Technologique - El Ghazala - Tél +216 70 250 000 - Fax +216 70 685454