

RESUME

Levio is a consulting firm that aims to energize and manage its resources and the partners it works with through a web application. The goal of our project is to develop two web applications: one is done in C Sharp language via the .Net framework, and some in JAVA language using the JEE platform, in order to visualize each resource and the project it works, consults the mandates of each resource, the interviews of new employees, etc..

This report presents the details of the project from a technical point of view, where the first chapter will be devoted to the description of the company we are working with, as well as the problem and the solution. The second chapter will be reserved for the functional study accompanied by diagrams such as the use of case diagram, class diagram as well as the architects chosen for our solution and finally, the last chapter will contain the tools used.

Contents

GENERAL INTRODUCTION	1
I GENERAL CONTEXT	2
Introduction	3
I.1 Company presentation	3
I.2 Problematic	3
I.3 Study of the existing	4
I.4 Project description	5
I.4.1 Identification of actors	6
I.4.2 Functional requirements study	6
I.4.3 Non-Functional requirements study	8
Conclusion	9
II FUNCTIONAL STUDY	10
Introduction	11
II.1 Global use case diagram	11
II.2 System sequence diagrams	13
II.2.1 System sequence diagram of the use case "Register"	13
II.2.2 System sequence diagram for use case "Assign a resource to a project"	14
II.3 Global class analysis diagram	14
II.4 Graphic interfaces and scenarios	16
II.4.1 Register	16
II.4.2 Add skills	17
II.4.3 Apply for a job	17
II.4.4 Assign a resource to a project	18
II.5 Global architecture of the application	19
II.5.1 Logical architecture	19
II.5.2 Physical architecture	20
Conclusion	22
III IMPLEMENTATION	23
Introduction	24
III.1 Description of proposed methods	24
III.2 Roadmap	25
III.3 Presentation of the development tools used	29
III.3.1 JBoss	29
III.3.2 Visual Studio	29
III.3.3 WildFly	29
III.3.4 Microsoft SQL Server	30
III.3.5 TexMaker	30
III.3.6 GitHub	30

Conclusion	31
CONCLUSION & PERSPECTIVES	33

List of Figures

I.1	Identity card of the company	3
I.2	Company "LEVIO"	3
I.3	Company "COFOMO"	4
I.4	Company "Sintegra Consulting"	5
II.1	Global use case diagram	12
II.2	System sequence diagram of the use case "Register"	13
II.3	System sequence diagram of the use case "Assign a resource to a project" . .	14
II.4	Global class analysis diagram	15
II.5	Fill out the registration form	16
II.6	Add skills	17
II.7	Apply for a job	17
II.8	Projects List	18
II.9	Assign a resource to a project	18
II.10	Logical architecture of the application	19
II.11	Physical architecture of the application	20
III.1	Diagram illustrating the steps of the scrum method	24
III.2	Gantt chart	25
III.3	October's planning	26
III.4	November's planning	27
III.5	December's planning	28
III.6	JBoss	29
III.7	Visual Studio	29
III.8	WildFly	29
III.9	Microsoft SQL Server Management Studio	30
III.10	TexMaker	30
III.11	GitHub	30
III.12	Setting up and configurion of the github	31
III.13	List of collaborators	31

List of Tables

I.1	Advantages and disadvantages of the competitor website	4
I.2	Identification of actors	6
I.3	Identification of the Functional requirements	7
I.4	Identification of technical requirements	8
I.5	Identification of the ergonomic requirements	9

GENERAL INTRODUCTION

During the academic year 2018-2019, we are asked to realize an application as part of the Project Integration Development module. This project puts into practice the skills acquired in different programming languages (JEE, .Net) as well as discovering the professional world of complex applications development.

In the same context, we are honored today to apply our knowledge in these modern technologies to embody the idea of our client. The aim is to create an application for Levio Company to foster the development and retention of its resources in a context of labor shortages. The main goal is to ensure that it has a global view of the assignment of all its employees into the different projects.

This report will therefore discuss the different phases of our project, namely the preliminary study, the in-depth study and the design. It then comprises three chapters: The first chapter will be dedicated to the presentation of Project. We will first introduce our project. Then we will study and critic other similar applications in order to propose an adequate and better solution.

The second chapter will be devoted to the functional study, where we will present the main UML diagrams of the application: use case diagrams, class diagrams and sequence diagrams, specify the choice of architectures used in our application and present graphic interfaces of certain scenarios through screenshots.

The third and the last chapter is dedicated to the implementation. We will present the appropriate method to properly manage this project as well as the development tools that we will use.

Chapter I

GENERAL CONTEXT

Introduction

In this first chapter, we first introduced the company followed by a study of the existing. Then, we focused on the different functional as well as non-functional requirements.

I.1 Company presentation

Figure I.1 illustrates the identity card of the company.



Figure I.1: Identity card of the company

LEVIO is a company specializing in support when implementing programs or major projects. Its strength is represented in its vision focused on achieving the expected benefits. Its leverage is the ability of customers to operationalize changes on the ground. [6]



Figure I.2: Company "LEVIO"

Levio's main areas of activities are insurance, finance, health and government services.

I.2 Problematic

The issue with this project is that the company needs a powerful tool able to manage all of its activities; from managing and assigning resources to assuring the communication between all actors. The application also needs to offer a global view of the company's activities.

I.3 Study of the existing

The digital transformation has become a strategic issue for companies. To succeed in this transformation, organizations must face the challenge of mastering all dimensions such as the evolution of their business model, improving the customer experience, redesigning business processes, setting up a culture of innovation and the exploitation of digital technologies.

Here are some examples of consulting firms:



Figure I.3: Company "COFOMO"

IT consulting firm recognized for the quality of its expertise, the excellence of its services and its professionally human approach.[1]

❖ **Location:** 1000 Rue de la Gauchetière O, Montréal, QC H3B 4X5, Canada

❖ **Website:** <https://www.cofomo.com/fr/>

Advantages	Disadvantages
High traceability	Non-optimized response time.
	Complicated and congested interfaces.

Table I.1: Advantages and disadvantages of the competitor website



Figure I.4: Company "Sintegra Consulting"

Sintegra Consulting specializes in recruiting Maghreb engineers for more than 50 renowned French IT companies.

❖ **Location:** Rue des Entrepreneurs, Tunis

➤ This company does not have a website; it exposes its services of consultants on social networks for example: LinkedIn, Facebook, etc.

I.4 Project description

Desiring to promote the development and retention of its resources in a context of labor shortage, Levio wants to ensure that it has a global view on the allocation of all its employees, as well as the duration of their mandate, i.e date of entry and date of exit. However, this is not currently possible with the current tools.

In this sense, Levio wants to set up a resource allocation tool that will allow it to have an overview of these resources, as well as other relevant information that will reflect the profile of the employee.

I.4.1 Identification of actors

Actor	Description
Visitor	To benefit from the features of our platform the visitor should register.
Resource	The resource is generally employed by Levio who must be assigned to a business area of the organization.
Client	Our application allows customers to buy the different services of consultants offered by Levio.
Administrator	The administrator handles the administrative tasks; includes recruitment, accounting, etc..

Table I.2: Identification of actors

I.4.2 Functional requirements study

These are the features to be provided by the application. These are the requirements specifying the input-output behavior. The application must allow to:

Requirement	Description
Register	This feature allows the visitor to create a profile to benefit of the services offered by Levio.
Authenticate	This feature allows a resource to connect.
Customers management	The administrator will have the privilege to accept the partnership request, modify and ban a client as well as assign a project to him/her.
Projects management	The customer will have the privilege of creating, modifying and deleting a project under the agreement of the administrator.
Mandates management	The administrator will designate the appropriate resource(s) for a mandate and will manage the assignment history.
Resources management	A resource has the right to modify its profile and deactivate it. It can manage its skills (add, modify and delete) as it has to define its availability.
Communication management	The manager will deal with requests from different customers as well as messages.
Dashboard management	The administrator can generate statistics as well as detailed reports.
International recruitment management	A resource must prepare a hiring file and send a request. The administrator will manage the requests and will take care of the process of reception.

Table I.3: Identification of the Functional requirements

I.4.3 Non-Functional requirements study

These are the features that characterize the system. These are requirements related to performance and the type of design. These needs may relate to implementation constraints. As part of this work, the application must be able to:

❖ Technical requirements:

Requirement	Description
Performance	Queries must be optimized to ensure minimal response time. High availability and tolerance to breakdowns. Avoid the problem of scalability.
Maintainability	The application must evolve over time as it must be closed to modification and open to extension.
Security	Ensure account security through encrypted passwords at the database. The application requires authentication while managing access controls.
Capacity	The ability to communicate with other remote applications and provide the service to different types of client.

Table I.4: Identification of technical requirements

❖ Ergonomic requirements:

Requirement	Description
Design	Navigation between interfaces must be simple, homogeneous, light and fluid.
Ease-of-use	The user must be guided when entering certain information in order to respect the formats of the fields of our database.

Table I.5: Identification of the ergonomic requirements

Conclusion

In this chapter, we have developed a general outline of the project context and the major objectives to be addressed. We will continue with the next chapter on the functional study

Chapter II

FUNCTIONAL STUDY

Introduction

The analysis and requirements specification phase is an important step in the development cycle of a project. Indeed, it provides a better understanding of the work required by identifying the needs of the different users that the system must accomplish.

II.1 Global use case diagram

We are reaching a key stage of the process. It is thanks to this process that the study realized in the preceding part will emphasize the role of each actor of the system as well as the functionalities presented above.

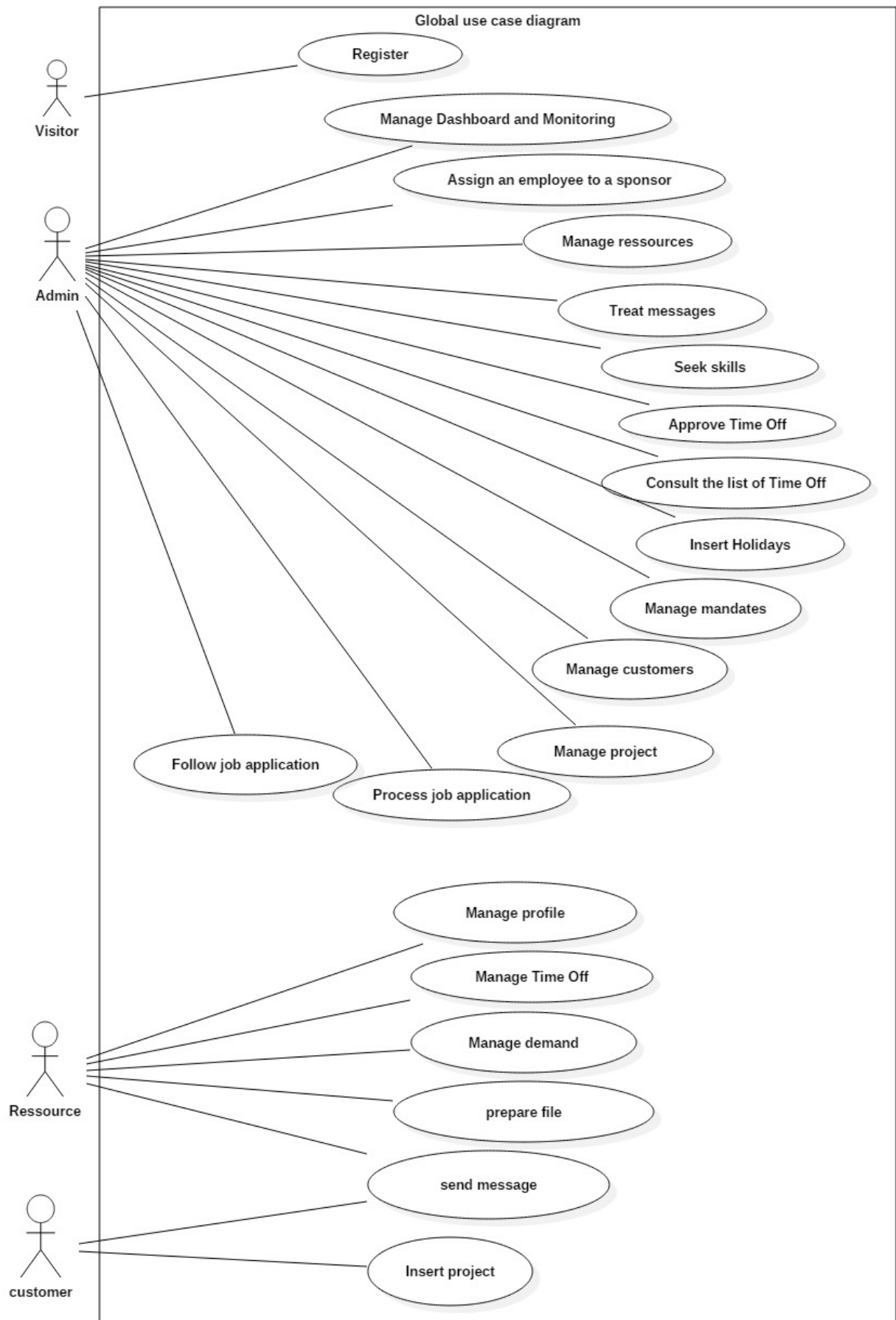


Figure II.1: Global use case diagram

II.2 System sequence diagrams

II.2.1 System sequence diagram of the use case "Register"

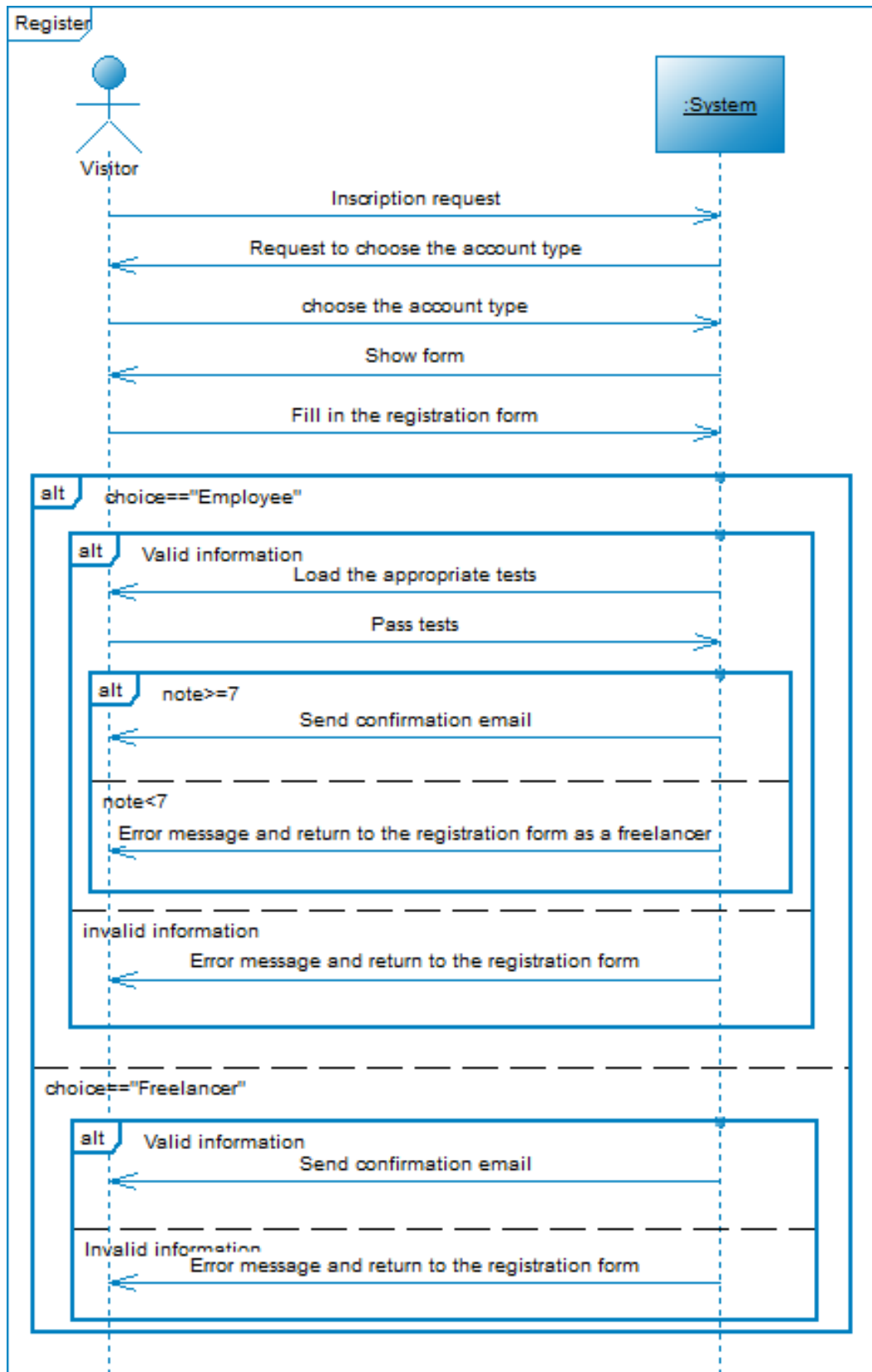


Figure II.2: System sequence diagram of the use case "Register"

II.2.2 System sequence diagram for use case "Assign a resource to a project"

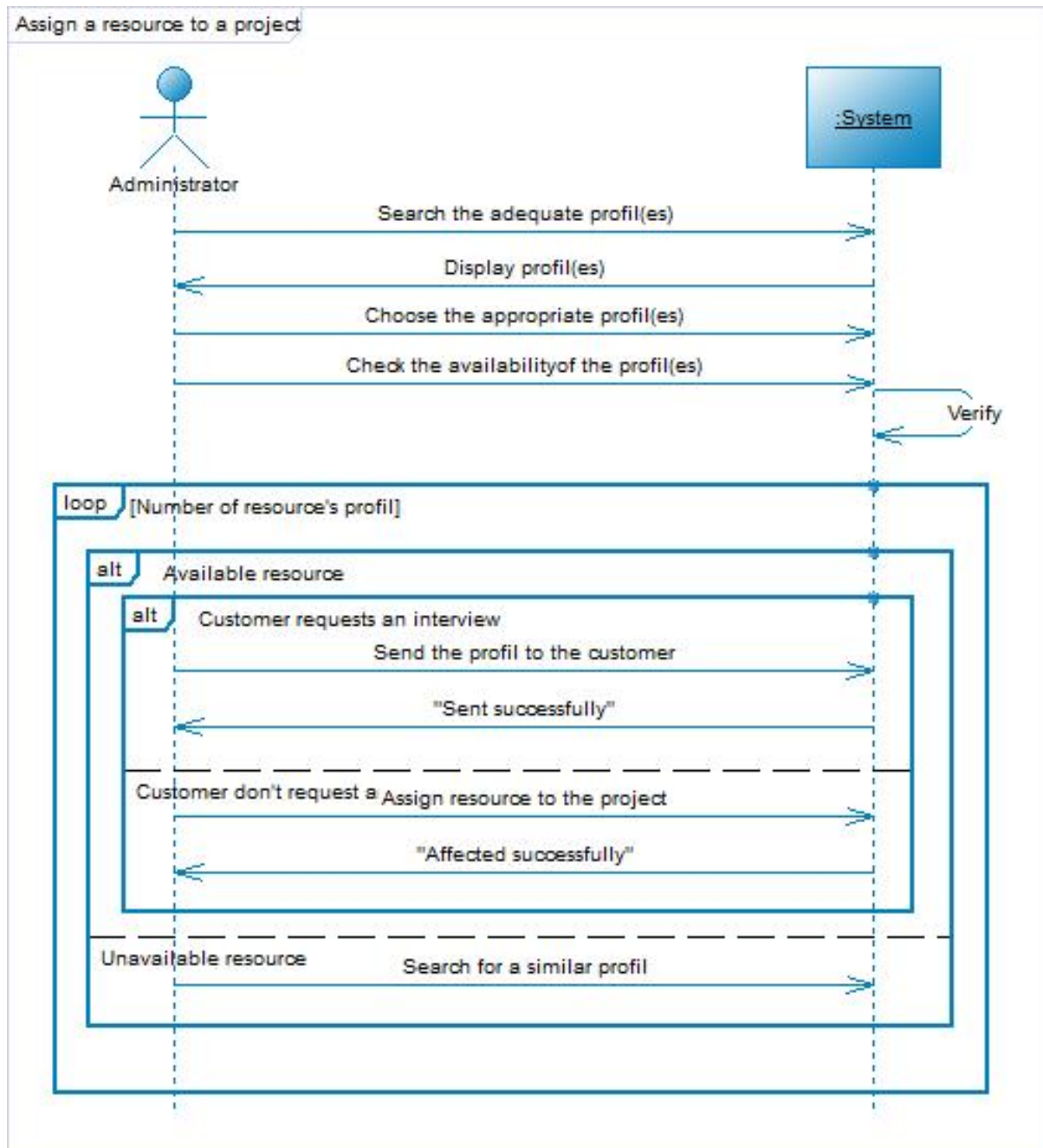


Figure II.3: System sequence diagram of the use case "Assign a resource to a project"

II.3 Global class analysis diagram

The UML class diagram is intended to describe the objects and information structures used on our application, both internally and in communication with its users. It describes the information without referring to a particular implementation.

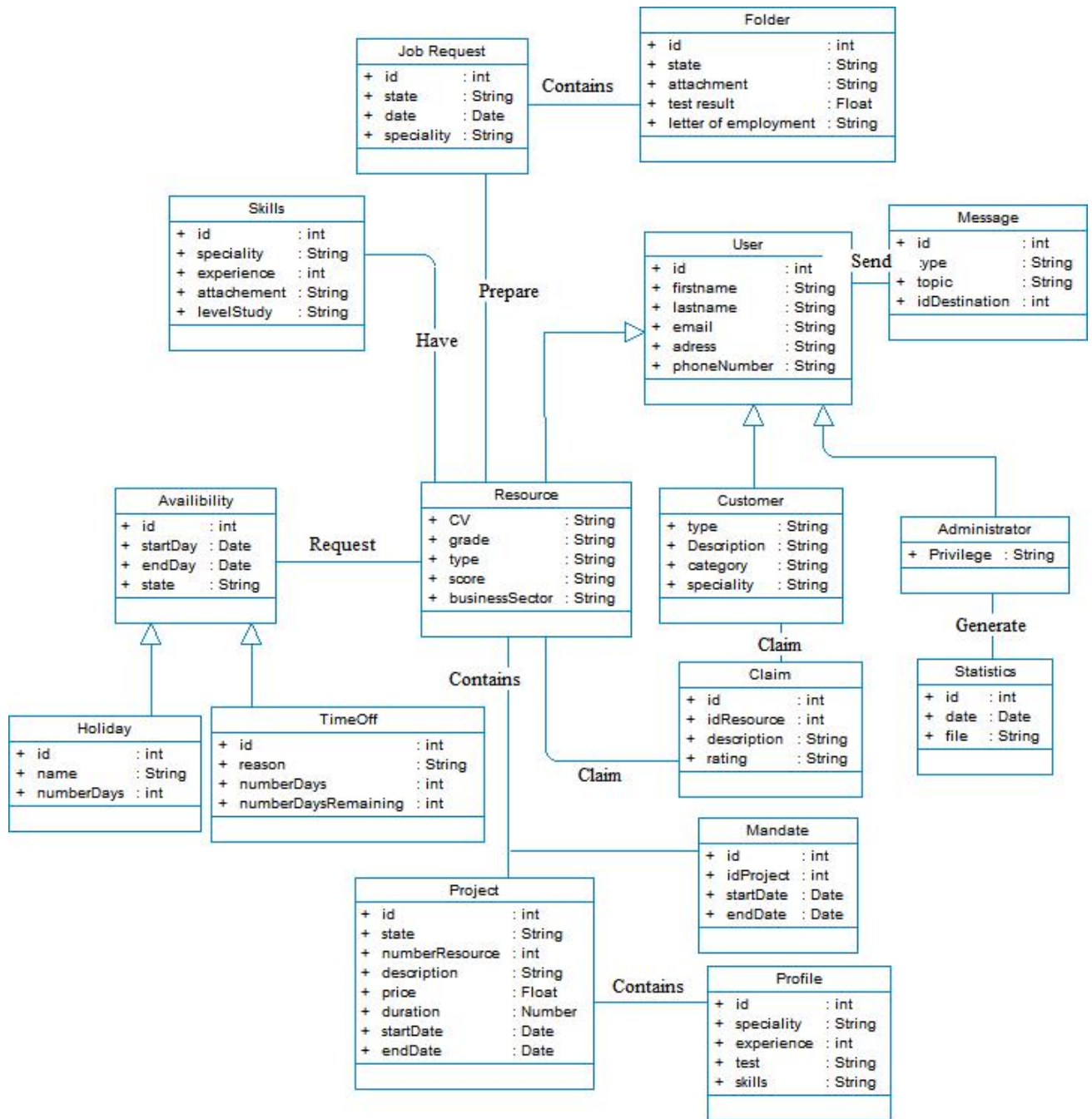


Figure II.4: Global class analysis diagram

II.4 Graphic interfaces and scenarios

II.4.1 Register

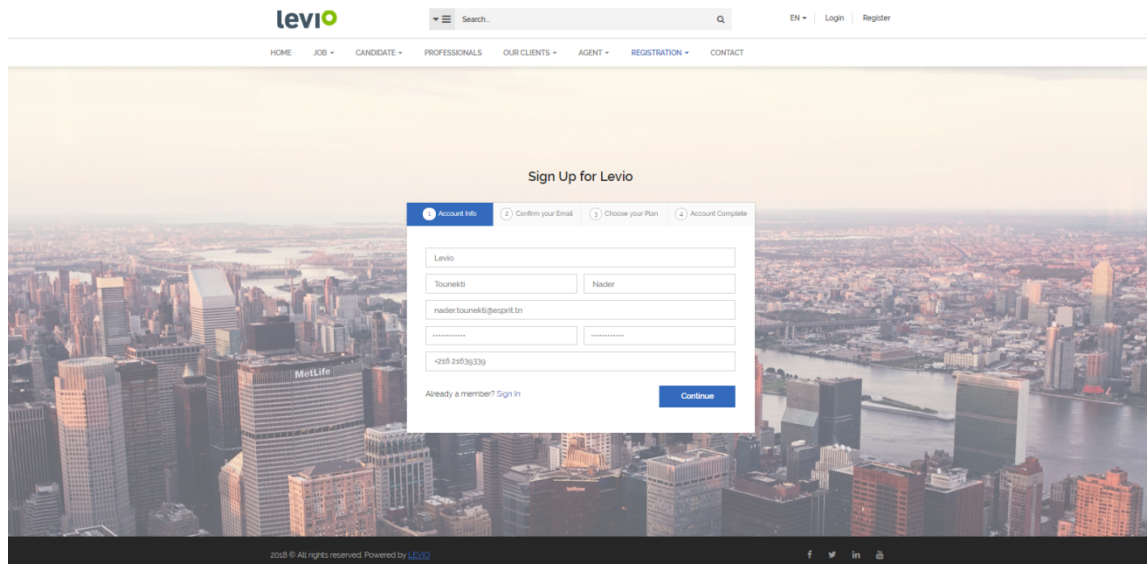
The image shows a web browser displaying the Levio registration page. The background is a cityscape at sunset. The page has a navigation bar with the Levio logo, a search bar, and links for EN, Login, and Register. Below the navigation bar is a menu with links: HOME, JOB, CANDIDATE, PROFESSIONALS, OUR CLIENTS, AGENT, REGISTRATION, and CONTACT. The main content area is titled "Sign Up for Levio" and features a four-step progress bar: 1. Account Info (active), 2. Confirm your Email, 3. Choose your Plan, and 4. Account Complete. The "Account Info" step contains a registration form with the following fields: a text field for "Levio" (containing "Levio"), a text field for "Tounekdi" (containing "Tounekdi"), a text field for "Nader" (containing "Nader"), a text field for "nader.tounekdi@esprit.tn", a password field (containing "*****"), and a phone number field (containing "+216 24536339"). Below the form is a link "Already a member? Sign In" and a blue "Continue" button. The footer contains the text "2018 © All rights reserved. Powered by LEJO" and social media icons for Facebook, Twitter, LinkedIn, and YouTube.

Figure II.5: Fill out the registration form

This interface will be used by any visitor to sign up, the concerned one will have to fill out a form with his basic informations, if the form isn't valid, an error alert will inform him about the field in question.

II.4.2 Add skills

Preview **Import CV** **EN** **Add Language**

About The Candidate

UPLOAD A PICTURE

GENERAL INFORMATION [Edit](#)

NAME:	Tounekti Nader
GENDER:	MALE
DATE OF BIRTH:	27-05-1994
RESIDENCE:	Ben Arous, MOUROUJ
TEL:	+21658865370
MOB:	+21658865370
EMAIL:	nadertounekti@esprit.tn
SOCIAL NETWORK:	Web Development
YEARS OF EXP:	7 Years
NATIOANLITY:	TUNISIEN
WORK PERMIT:	GENERAL WORK PERMIT
MARITAL STATUS:	SINGLE
CHILDREN:	0
DRIVING LICENSE:	[driving license]
AVAILABILITY:	[availability]
EMPLOYMENT STA:	EMPLOYED

Skills

Skills List:

- php** (Progress: 75%)
 - Description: [Text Area]
 - Buttons: Edit, Delete, Save
- html** (Progress: 77%)
 - Description: [Text Area]
 - Buttons: Edit, Delete, Save
- javascript** (Progress: 30%)
 - Description: [Text Area]
 - Buttons: Edit, Delete, Save

Additional Skills

Add your skills separated by comma

Add a Skill

Figure II.6: Add skills

After the registration, the visitor needs to set up his profile by adding his skills with a description, importing a CV and uploading his picture.

II.4.3 Apply for a job

Sr. Quality Engineer
Published: 10 nov 2014 | job ID: 29930
San Francisco, California

IT Auditor/ IT Risk Analyst
Published: 10 nov 2014 | job ID: 29930
San Francisco, California

Applicants For: Sr. Quality Engineer **Sort by**

Tounekti Nader
Web Developer
LOCATION: BEN AROUS, EZZAHRA
NATIONALITY: TUNISIE
YEARS OF EXPERIENCE: 7
DEGREE: MBA
EMPLOYMENT STATUS: EMPLOYED
GENDER: MALE
WORK PERMIT: GENERAL WORK PERMIT
More Info

Rania Midani
freelancer
LOCATION: BEN AROUS, EZZAHRA
NATIONALITY: TUNISIENNE
YEARS OF EXPERIENCE: 7
DEGREE: MBA
EMPLOYMENT STATUS: FISCISTE
GENDER: FEMALE
WORK PERMIT: GENERAL WORK PERMIT
More Info

Figure II.7: Apply for a job

Then in this interface, the ressource can browse the projects list which is containing all the details to apply for.

II.4.4 Assign a resource to a project


Client	Name Project	Description	competences	price	Assignment
	ERP	In order to strengthen our project teams, OOREDOO Tunisia is recruiting: - HTML Development Engineers - PHP Development Engineers	PHP-HTML-JAVASCRIPT-SQL	\$1500	<button>ASSIGN</button>
	WEBSITE RH	We are looking for Juniors and experienced Javascript developers to work on a new generation of B2B web platform.	ANGULAR 6 - NODE - MONGODB	\$2500	<button>ASSIGN</button>
	Website E-commerce	To support our strong growth, we seek to surround ourselves with talented, passionate, innovation-driven profiles and a Start-up spirit!EY Consulting is recruiting a DEVELOPPEUR BACK END	JAVA - SQL -	\$3000	<button>ASSIGN</button>

Figure II.8: Projects List







No	Name	Email	Phone	Role	Age	Skills	Salary	Action
1	 Midani Rania	rania.midani@esprit.tn	+123 456 789	<button>Developer</button>	23	PHP-HTML-JAVASCRIPT-SQL	\$3200	<button>ASSIGN</button>
2	 Tounekti Nader	nader.tounekti@esprit.tn	+234 456 789	<button>Developer</button>	26	PHP-HTML-JAVASCRIPT-SQL	\$1800	<button>ASSIGN</button>
3	 Didi Doha	didi.doha@esprit.tn	+345 456 789	<button>Developer</button>	28	PHP-HTML-JAVASCRIPT-SQL	\$2200	<button>ASSIGN</button>
4	 Boudali Anis	anis.boudali@esprit.tn	+456 456 789	<button>Developer</button>	25	PHP-HTML-JAVASCRIPT-SQL	\$2200	<button>ASSIGN</button>
5	 Guerfali Bechir	bechir.guerfali@esprit.tn	+678 456 789	<button>Developer</button>	29	PHP-HTML-JAVASCRIPT-SQL	\$1500	<button>ASSIGN</button>
6	 Hadhri Jasser	jasser.hadhri@esprit.tn	+123 456 789	<button>Developer</button>	35	PHP-HTML-JAVASCRIPT-SQL	\$3200	<button>ASSIGN</button>

Figure II.9: Assign a resource to a project

Those two interfaces demonstrate the ability of the admin to assign a ressource for a project, starting by choosing the project (Figure II.8) and then choosing the ressource concerned (Figure II.9). This step can't be done if the ressource doesn't sign up and create his own profile.

II.5 Global architecture of the application

II.5.1 Logical architecture

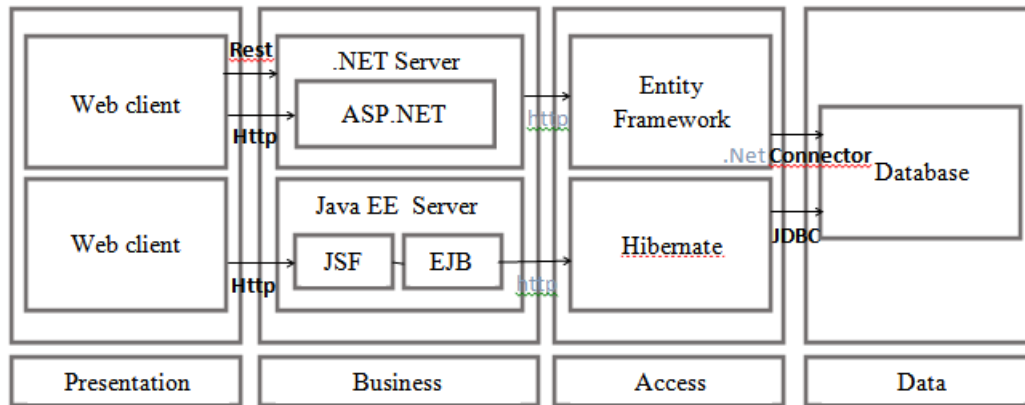


Figure II.10: Logical architecture of the application

- ❖ Presentation Layer is in touch with the user of the Web application. The user interacts with the Web application through Web pages visualized on the browser .
- ❖ The business layer aims to answer to the requests of the client. It implements the rules of the application, such as the calculation of expenses-mandates. This layer uses data coming from the user via the Presentation layer and the DBMS via the Data layer. The business layer is divided into two components: Java EE Server and .Net Server. They are connected by a restful decouple file.
 - **Java EE Server** is composed of:
 - **Web service:** it represents communication mechanism between application distances within the Internet independently from any programming language and any execution platform. For example, Wildfly is web service for our application.
 - **Java Server Faces (JSF)** is a Java Framework to develop web application.
 - **Enterprise JavaBeans (EJB)** is a software components' architecture on the server for the development platform Java EE
 - **.Net Server:** it is composed of ASP.NET. The last is a framework that generates Web pages, launched by Microsoft in July 2002, and used to create Web applications. This technique is incorporated in Microsoft .NET4

- ❖ The Access Layer, in the form of the Hibernate and Entity Framework, manages the data access to DBMS. They play the role of the ORM (Object Relational Mapper). It is considered as a bridge between manipulated objects in the Access Layer and the rows of the relational database.
- ❖ The Data layer represents the database tier. The data are stocked there externally. It is not saved in the System. The database uses the capacity of the DBMS SQL Server.
- ❖ The integration of the layer is held by the EJB3 container (Enterprise Java Bean).

II.5.2 Physical architecture

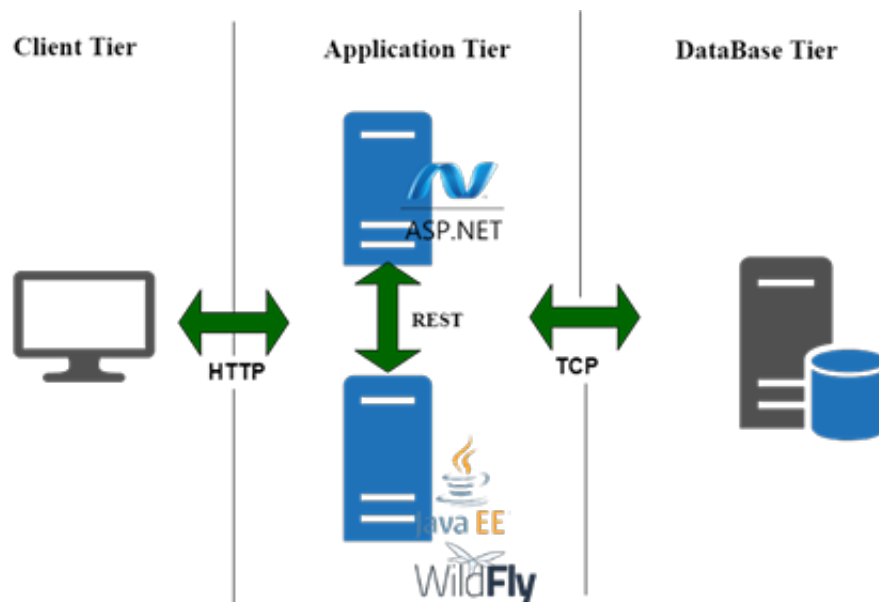


Figure II.11: Physical architecture of the application

A 3-tier architecture is a type of software architecture which is composed of three “tiers” or “layers” of logical computing. They are often used in applications as a specific type of client-server system. 3-tier architectures provide many benefits for production and environments development by modularizing the user interface, business logic, and data storage layers. Doing so gives greater flexibility to development teams by allowing them to update a specific part of an application independently of the other parts.

This added flexibility can improve overall time-to-market and decrease development cycle times by giving development teams the ability to replace or upgrade independent tiers without affecting the other parts of the system.

- ❖ **Tier Presentation:** The tier presentation is the front end layer in the 3-tier system and consists of the user interface. This user interface is often a graphical one accessible through a web browser or web-based application and which displays content and information useful to an end user. This tier is often built on web technologies or through other popular web development frameworks, and communicates with others layers through API calls.
- ❖ **Tier Application:** The tier application contains the functional business logic which drives an application's core capabilities. We deploy Wildfly for Java EE and IIS for the .Net part of the layer.
- ❖ **Tier Data:** The tier data comprises of the database/data storage system and data access layer. We will use for this project Microsoft SQL Server. Data is accessed by the application layer via API calls.
- ❖ **Advantages:**

There are many benefits to using a 3-layer architecture including speed of development, scalability, performance, and availability. As mentioned, modularizing different tiers of an application gives development teams the ability to develop and enhance a product with greater speed than developing a singular code base because a specific layer can be upgraded with minimal impact on the other layers. It can also help improve development efficiency by allowing teams to focus on their core competencies. Many development teams have separate developers who specialize in front- end, server back-end, and data back-end development, by modularizing these parts of an application you no longer have to rely on full stack developers and can better utilize the specialties of each team.

Scalability is another great advantage of a 3-layer architecture. By separating out the different layers you can scale each independently depending on the need at any given time.

By having disparate layers you can also increase reliability and availability by hosting different parts of your application on different servers and utilizing cached results. With a full stack system you have to worry about a server going down and greatly affecting performance throughout your entire system, but with a 3-layer application, the increased independence created when physically separating different parts of an application minimizes performance issues when a server goes down.

Conclusion

We have specified the different needs that our application must meet. This chapter was useful to show our purpose, our needs and to clarify our approach.

Chapter III

IMPLEMENTATION

Introduction

In this chapter, we have presented the necessary methods for setting up and linking the project, the various development tools used accompanied by a short description.

III.1 Description of proposed methods

Agile methods characterize a management mode of IT projects that favors dialogue between all the stakeholders, customers, users, developers and other project professionals, flexibility in progress, the ability to modify plans and speed of delivery. It is a question of breaking with more traditional practices that are far too rigid and too demanding in terms of (contractual) specifications. Therefore, it is important to give priority to relational and extended communication on development processes.

And for that purpose we chose to work with the scrum method; a diagram of complex product development organization. It is defined by its creators as an "iterative holistic framework that focuses on common goals by productively and creatively delivering products of the greatest possible value." [8]

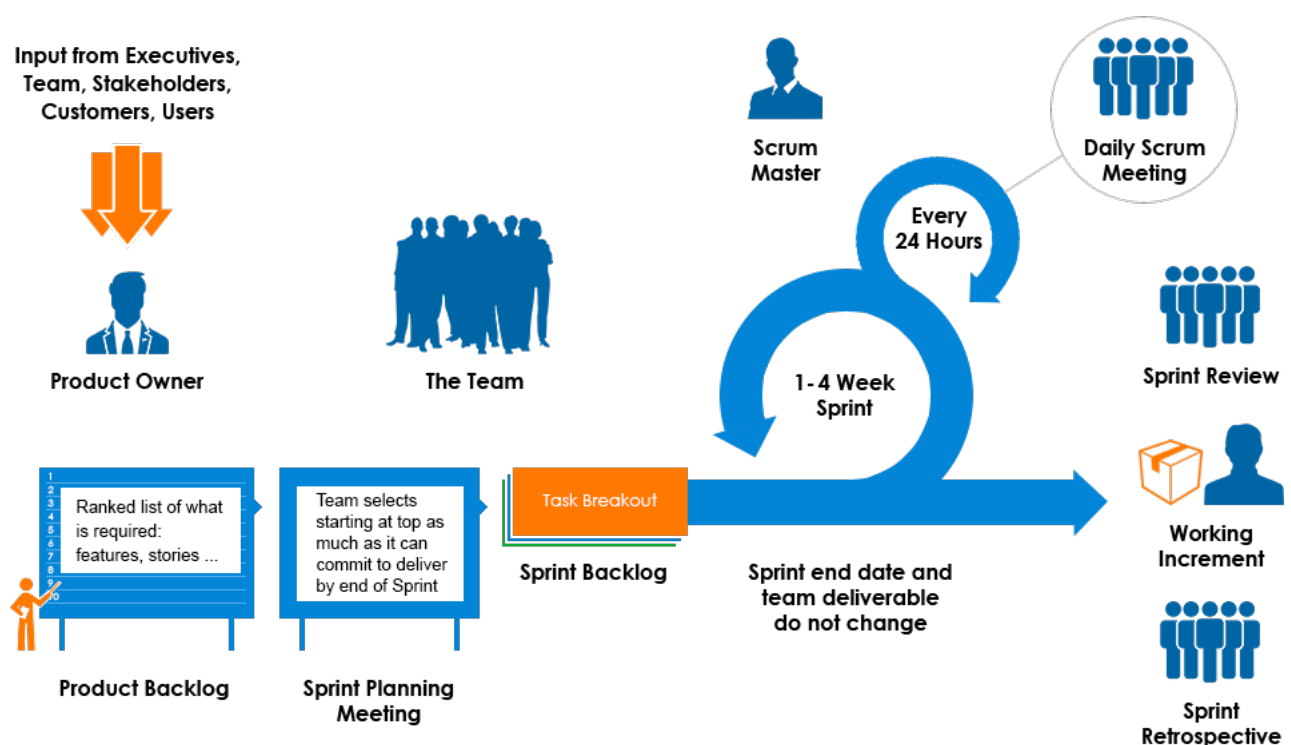


Figure III.1: Diagram illustrating the steps of the scrum method

III.2 Roadmap

Roadmap is a simplified graphic representation that communicates and effectively shares a strategic intent to mobilize, align and coordinate the efforts of stakeholders to achieve one or more goals. [7]

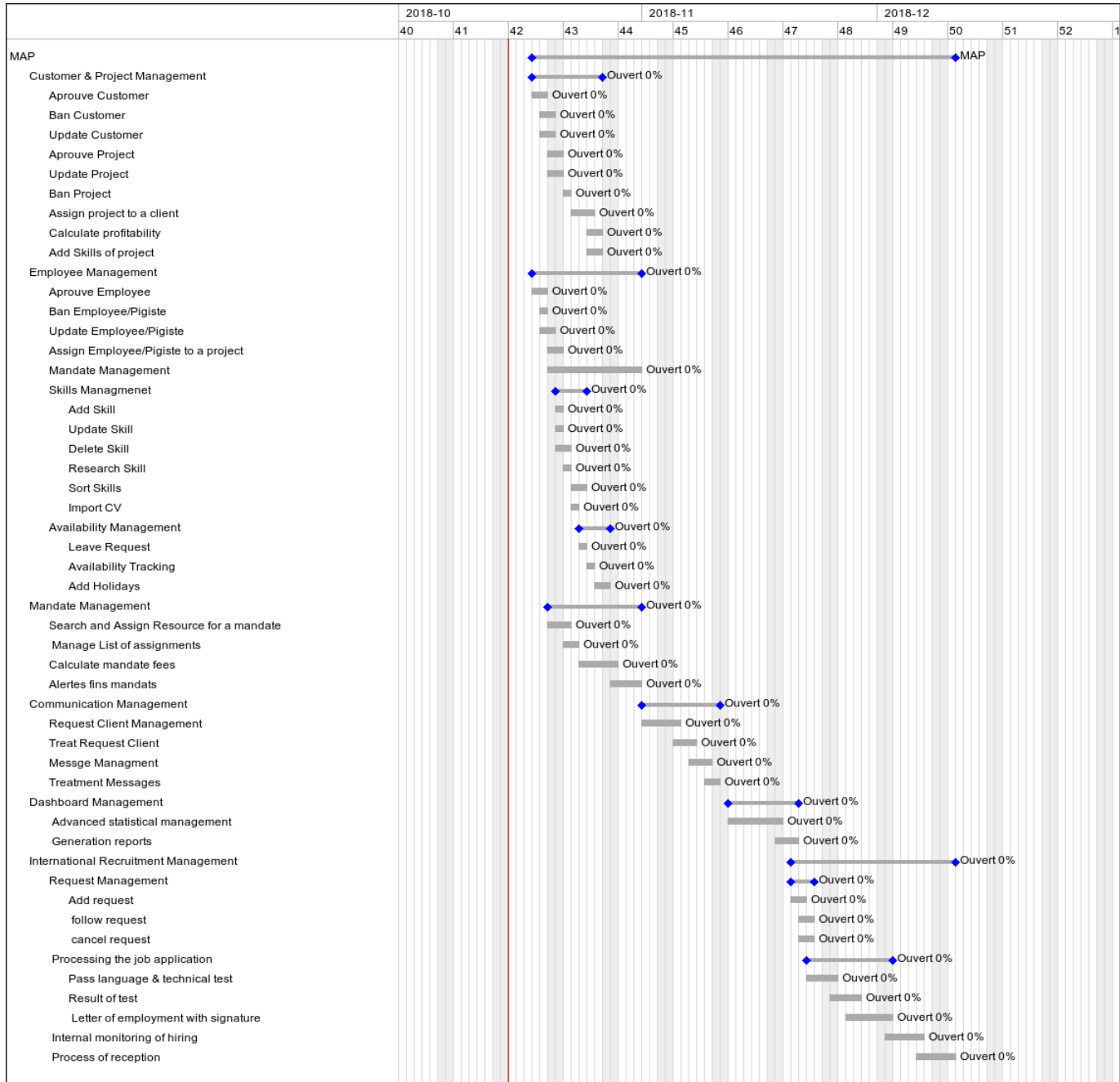


Figure III.2: Gantt chart

The Gantt chart, commonly used in project management, is one of the most effective tools for visually representing the status of the various activities (tasks) that constitute a project. [2]

	lundi	mardi	mercredi	jeudi	vendredi	samedi	dimanche
40	1	2	3	4	5	6	7
41	8	9	10	11	12	13	14
42	15	16	17	18	19	20	21
				<ul style="list-style-type: none"> Tâche #8: Customer & Project Management 	<ul style="list-style-type: none"> Tâche #12: Approuve Customer 	<ul style="list-style-type: none"> Tâche #9: Ban Customer 	<ul style="list-style-type: none"> Tâche #14: Approuve Project
				<ul style="list-style-type: none"> Tâche #12: Approuve Customer 	<ul style="list-style-type: none"> Tâche #21: Approuve Employee 	<ul style="list-style-type: none"> Tâche #13: Update Customer 	<ul style="list-style-type: none"> Tâche #15: Update Project
				<ul style="list-style-type: none"> Tâche #20: Employee Management 	<ul style="list-style-type: none"> Tâche #22: Ban Employee/Project 	<ul style="list-style-type: none"> Tâche #23: Update Employee/Project 	<ul style="list-style-type: none"> Tâche #24: Assign Employee/Project to ...
				<ul style="list-style-type: none"> Tâche #21: Approuve Employee 	<ul style="list-style-type: none"> Tâche #9: Ban Customer 	<ul style="list-style-type: none"> Tâche #14: Approuve Project 	<ul style="list-style-type: none"> Tâche #25: Add Skill
					<ul style="list-style-type: none"> Tâche #13: Update Customer 	<ul style="list-style-type: none"> Tâche #15: Update Project 	<ul style="list-style-type: none"> Tâche #27: Update Skill
				<ul style="list-style-type: none"> Tâche #23: Update Employee/Project 		<ul style="list-style-type: none"> Tâche #24: Assign Employee/Project to ... 	<ul style="list-style-type: none"> Tâche #25: Skills Management
						<ul style="list-style-type: none"> Tâche #38: Mandate Management 	<ul style="list-style-type: none"> Tâche #26: Delete Skill
						<ul style="list-style-type: none"> Tâche #37: Mandate Management 	
						<ul style="list-style-type: none"> Tâche #38: Search and Assign Resource ... 	
43	22	23	24	25	26	27	28
	<ul style="list-style-type: none"> Tâche #16: Ban Project 	<ul style="list-style-type: none"> Tâche #31: Import CV 	<ul style="list-style-type: none"> Tâche #25: Skills Management 	<ul style="list-style-type: none"> Tâche #17: Assign project to a client 	<ul style="list-style-type: none"> Tâche #5: Customer & Project Management 	<ul style="list-style-type: none"> Tâche #32: Availability Management 	<ul style="list-style-type: none"> Tâche #40: Calculate mandate fees
	<ul style="list-style-type: none"> Tâche #28: Delete Skill 	<ul style="list-style-type: none"> Tâche #33: Manage List of assignments 	<ul style="list-style-type: none"> Tâche #30: Sort Skills 	<ul style="list-style-type: none"> Tâche #34: Availability Tracking 	<ul style="list-style-type: none"> Tâche #18: Calculate profitability 	<ul style="list-style-type: none"> Tâche #35: Add Holidays 	<ul style="list-style-type: none"> Tâche #41: Alerter fins mandats
	<ul style="list-style-type: none"> Tâche #29: Research Skill 	<ul style="list-style-type: none"> Tâche #17: Assign project to a client 	<ul style="list-style-type: none"> Tâche #33: Leave Request 	<ul style="list-style-type: none"> Tâche #15: Calculate profitability 	<ul style="list-style-type: none"> Tâche #19: Add Skills of project 		
	<ul style="list-style-type: none"> Tâche #35: Search and Assign Resource ... 	<ul style="list-style-type: none"> Tâche #30: Sort Skills 	<ul style="list-style-type: none"> Tâche #32: Availability Management 	<ul style="list-style-type: none"> Tâche #15: Add Skills of project 	<ul style="list-style-type: none"> Tâche #35: Add Holidays 		
	<ul style="list-style-type: none"> Tâche #35: Manage List of assignments 	<ul style="list-style-type: none"> Tâche #40: Calculate mandate fees 					
44	29	30	31	1	2	3	4
		<ul style="list-style-type: none"> Tâche #20: Employee Management 	<ul style="list-style-type: none"> Tâche #42: Communication Management 	<ul style="list-style-type: none"> Tâche #42: Communication Management 			
		<ul style="list-style-type: none"> Tâche #38: Mandate Management 	<ul style="list-style-type: none"> Tâche #43: Request Client Management 	<ul style="list-style-type: none"> Tâche #43: Request Client Management 			
		<ul style="list-style-type: none"> Tâche #37: Mandate Management 					
		<ul style="list-style-type: none"> Tâche #41: Alerter fins mandats 					

- Tâche commençant ce jour
- Tâche finissant ce jour
- Tâche commençant et finissant ce jour

Figure III.3: October's planning

	lundi	mardi	mercredi	jeudi	vendredi	samedi	dimanche
44	29	30	31	1	2	3	4
			<div>Tâche #20: Employee Management</div> <div>Tâche #36: Mandate Management</div> <div>Tâche #37: Mandate Management</div> <div>Tâche #41: Alertas fins mandats</div>	<div>Tâche #42: Communication Management</div> <div>Tâche #43: Request Client Management</div>			
45	5	6	7	8	9	10	11
	<div>Tâche #43: Request Client Management</div> <div>Tâche #44: Treat Request Client</div>	<div>Tâche #44: Treat Request Client</div> <div>Tâche #45: Messge Management</div>			<div>Tâche #45: Messge Management</div> <div>Tâche #46: Treatment Messages</div>	<div>Tâche #42: Communication Management</div> <div>Tâche #46: Treatment Messages</div>	
46	12	13	14	15	16	17	18
	<div>Tâche #47: Dashboard Management</div> <div>Tâche #48: Advanced statistical manag...</div>					<div>Tâche #48: Advanced statistical manag...</div> <div>Tâche #49: Generation reports</div>	
47	19	20	21	22	23	24	25
	<div>Tâche #47: Dashboard Management</div> <div>Tâche #49: Generation reports</div> <div>Tâche #50: International Recruitment Ml...</div> <div>Tâche #51: Request Management</div> <div>Tâche #52: Add request</div>	<div>Tâche #52: Add request</div> <div>Tâche #53: follow request</div> <div>Tâche #54: cancel request</div>	<div>Tâche #51: Request Management</div> <div>Tâche #53: follow request</div> <div>Tâche #54: cancel request</div> <div>Tâche #55: Processing the job applica...</div> <div>Tâche #56: Pass language & technical test</div>			<div>Tâche #56: Pass language & technical test</div> <div>Tâche #57: Result of test</div>	
48	26	27	28	29	30	1	2
	<div>Tâche #58: Letter of employment with ...</div>	<div>Tâche #58: Letter of employment with ...</div>	<div>Tâche #57: Result of test</div>			<div>Tâche #55: Processing the job applica...</div> <div>Tâche #58: Letter of employment with ...</div> <div>Tâche #59: Internal monitoring of hiring</div>	

tâche commençant ce jour
 tâche finissant ce jour
 tâche commençant et finissant ce jour

Figure III.4: November's planning

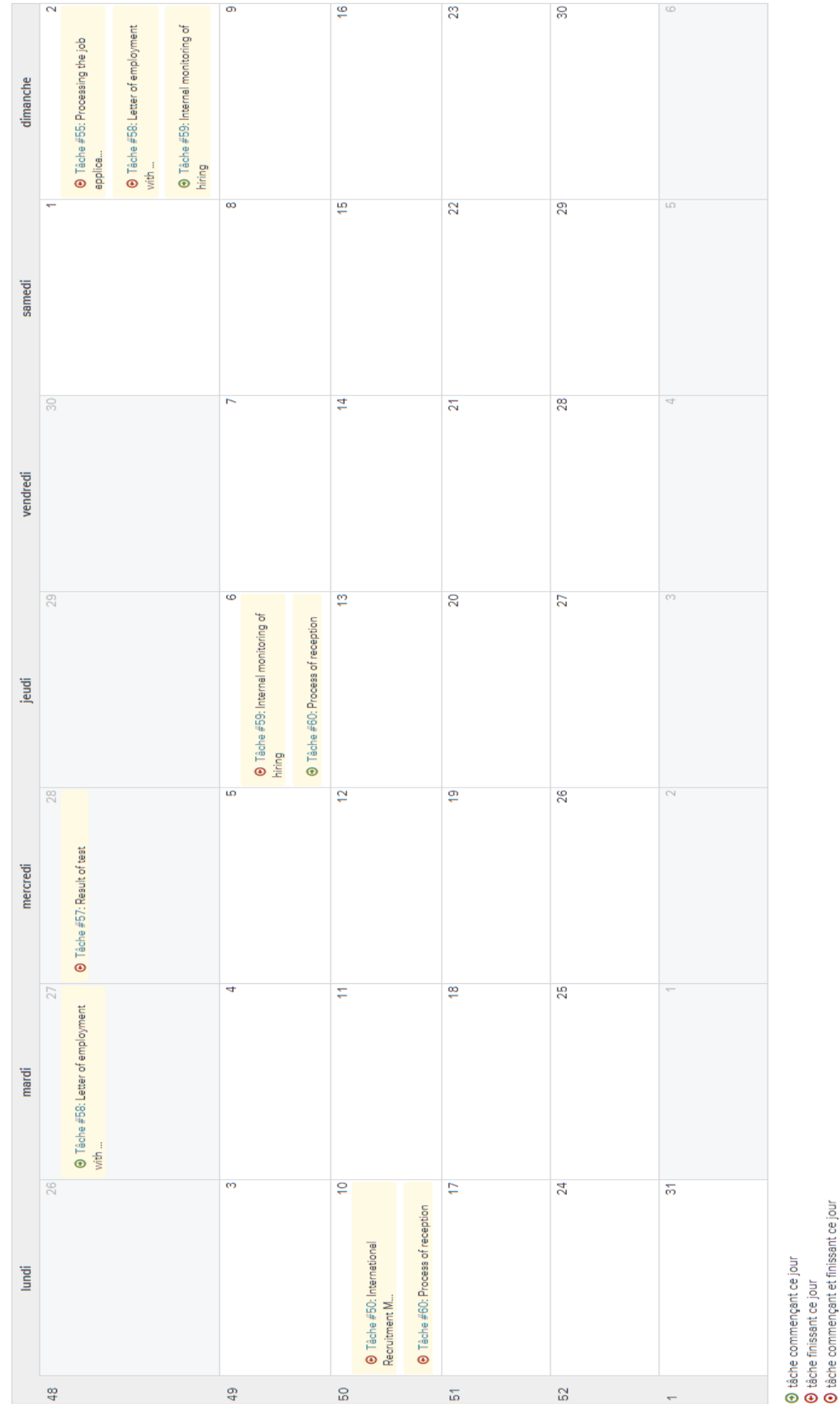


Figure III.5: December's planning

III.3 Presentation of the development tools used

III.3.1 JBoss



Figure III.6: JBoss

JBoss is a division of Red Hat that supports JBoss open source application server program and related middleware services, marketed under the JBoss Enterprise Middleware brand. JBoss is an open source solution designed to replace commercial offerings such as IBM WebSphere. [5]

III.3.2 Visual Studio



Figure III.7: Visual Studio [10]

Visual Studio .NET is a Microsoft-integrated development environment (IDE) that can be used for developing consoles, graphical user interfaces (GUIs), Windows Forms, Web services and Web applications.

III.3.3 WildFly



Figure III.8: WildFly

WildFly WildFly formerly known as JBoss AS, or simply JBoss, is an application server authored by JBoss, now developed by Red Hat. WildFly is written in Java and implements the Java Platform, Enterprise Edition (Java EE) specification. It runs on multiple platforms. [11]

III.3.4 Microsoft SQL Server



Figure III.9: Microsoft SQL Server Management Studio

Microsoft SQL Server is a relational database management system, or RDBMS, that supports a wide variety of transaction processing, business intelligence and analytics applications in corporate IT environments. It's one of the three market-leading database technologies, along with Oracle Database and IBM's DB2. [4]

III.3.5 TexMaker



Figure III.10: TexMaker

TexMaker Texmaker is a free, modern and cross-platform LaTeX editor for Linux, macOS and Windows systems that integrates many tools needed to develop documents with LaTeX, in just one application. [9]

III.3.6 GitHub



Figure III.11: GitHub

GitHub is an open source versioning and collaboration platform for software developers. Delivered as software on demand (SaaS, Software as a Service), the GitHub solution was launched in 2008. It is based on Git, an open source code management system created by Linus Torvalds with the aim of accelerating software development. [3]

➤ Setting up and configurion of the github

Branch: maître	PiMap2018 / Intégration /	Créer un nouveau fichier	Télécharger des fichiers	Rechercher un fichier	Historique
TounektiNader FirstCommit Dernier commit 1a61f2a 22 minutes ago					
..					
App_Start	FirstCommit	22 minutes ago			
Zones / Clients	FirstCommit	22 minutes ago			
Contenu	FirstCommit	22 minutes ago			
ContenuA	FirstCommit	22 minutes ago			
Contrôleurs	FirstCommit	22 minutes ago			
Propriétés	FirstCommit	22 minutes ago			
Des vues	FirstCommit	22 minutes ago			
des scripts	FirstCommit	22 minutes ago			
ApplicationInsights.config	FirstCommit	22 minutes ago			
Global.asax	FirstCommit	22 minutes ago			
Global.asax.cs	FirstCommit	22 minutes ago			
Integration.csproj	FirstCommit	22 minutes ago			
Web.Debug.config	FirstCommit	22 minutes ago			
Web.Release.config	FirstCommit	22 minutes ago			
Web.config	FirstCommit	22 minutes ago			
packages.config	FirstCommit	22 minutes ago			

Figure III.12: Setting up and configuion of the github

TounektiNader / PiMap2018

Regarder 0
Étoile 0
Fourche 0

Problèmes liés au code 0
Demandes d'extraction 0
Projets 0
Paramètres de
Wiki
Insights

Options
Collaborateurs
Branches
Webhooks
Intégrations et services
Déployer des clés

Modération
Limites d'interaction

Collaborateurs
Accès Push au référentiel

RaniaMidani

En attente de la réponse de Jasserhadhi
Copier le lien d'invitation
Annuler l'invitation

En attente de la réponse d'Anni
Copier le lien d'invitation
Annuler l'invitation

En attente de la réponse de BechirGuerfa
Copier le lien d'invitation
Annuler l'invitation

En attente de la réponse de dohadi
Copier le lien d'invitation
Annuler l'invitation

Recherche par nom d'utilisateur, nom complet ou adresse email

Vous ne pourrez trouver un utilisateur GitHub par son adresse e-mail que s'il a choisi de le lister publiquement. Sinon, utilisez leur nom d'utilisateur à la place.

Ajouter un collaborateur

Figure III.13: List of collaborators

Conclusion

We have specified the different needs that our application must meet. This chapter was

useful to show our purpose, our needs and to clarify our approach.

CONCLUSION & PERSPECTIVES

After achieving the analysis chapter, we have a much clearer idea about the product that we are to deliver and how to proceed. We also identified the different and numerous tasks that must be developed at the end of each sprint.

We also conceived the UML diagrams in order to implement our application and most importantly, we now have a concrete architecture that will serve to the realization of this project.

Bibliography

- [1] comofo. <https://www.cofomo.com/fr/>. (visit  le 13/10/2018).
- [2] Gantt. https://fr.wikipedia.org/wiki/Diagramme_de_Gantt. (visit  le 12/10/2018).
- [3] Github. <https://www.lemagit.fr/definition/GitHub>. (visit  le 12/10/2018).
- [4] Introduction to microsoft sql server. <https://searchsqlserver.techtarget.com/definition/SQL-Servers>. (visit  le 12/10/2018).
- [5] Jboss. <https://www.lemagit.fr/definition/JBoss>. (visit  le 12/10/2018).
- [6] Levio. <https://www.levio.ca/>. (visit  le 12/10/2018).
- [7] Roadmap. <https://fr.wikipedia.org/wiki/Roadmap>. (visit  le 12/10/2018).
- [8] Scrum. [https://fr.wikipedia.org/wiki/Scrum_\(d%C3%A9veloppement\)](https://fr.wikipedia.org/wiki/Scrum_(d%C3%A9veloppement)).. (visit  le 12/10/2018).
- [9] Tex maker. <https://en.wikipedia.org/wiki/Textmaker>. (visit  le 14/10/2018).
- [10] Visual studio. <https://www.techopedia.com/definition/15740/visual-studio-net>. (visit  le 12/10/2018).
- [11] Wildfly. <https://en.wikipedia.org/wiki/WildFly>. (visit  le 12/10/2018).