Ministry of Higher Education and Scientific Research

Manouba University

Higher Institute of digital economy

Code: 1514

HR BOT MANAGER

Fekr Society

Made by

Mohamed Gundia

Supervised by

Rafik naccahe: Technical supervisor – FEKR

Dhouha maatar: Technical supervisor – ESEN

Academic Year: 2022/2023

I authorize the submission of the internship report for the purpose of a defense.

, Dhouha Maatar

Signature

I authorize the submission of the internship report for the purpose of a defense.

, Rafik Naccahe

Signature and seal

Dedication

We dedicate this modest work :

To my loving parents, for their continuous support during my End of Studies project, I would like to sincerely thank them.

To my friends and loved ones, whom I was unable to have completed my job as successfully and efficiently as in the manner I could without their assistance, compassion, and encouragement.

To my professors and supervisors.

For everything they have done for me, I sincerely thank them and will always be appreciative.

Mohamed Gundia

Acknowledgments

I desire to share my appreciation to the entire Fekr's personnel for providing a supportive and cooperative work environment. I was able to thrive both professionally and personally thanks to the open and accepting environment. I am appreciative for the chance of being a part of such an exciting community.

Furthermore, I would like to thank Mrs. Dhouha Maatar, who was essential to my web development process, from the bottom of my heart. Her in-depth expertise, great problem-solving skills, and constant mentoring have all been essential to my growth as a web developer. My technical skills have been strengthened thanks to Mrs. Maatar's tolerance and readiness to offer me helpful feedback, and she has also given me the confidence to take on tough difficulties. Being able to learn from someone as devoted and skilled as Mrs. Maatar has been a true blessing for me.

I am very appreciative of Rafik Naccahe, my technical supervisor, for his important advice and support throughout my end-of-study internship at Fekr. My educational experience has been greatly shaped by his considerable knowledge, skill, and passion.

I want to express our sincere appreciation to the jury for taking the time to analyze and assess my proposal and for playing an important part in the evaluation procedure.

Mohamed Gundia

Table of content

Table of figures

List of tables

List of acronyms

HTML	*HyperText Markup Language*
MVC	*Model View Controller*
HTTP	*HuperText Transfer Protocol*
API	*Application Programming Interface*
CSS	*Cascading Style Sheets*
DB	*Data Base*
US	*User Story*
UML	*Unified Modeling Language*
VSCode	*Visual Studio Code*

General Introduction

In today's rapidly evolving business landscape, the effective management of human resources and recruitment processes stands as a pivotal element in ensuring the growth and sustainability of organizations. For Small and Medium-sized Enterprises (SMEs), which constitute a significant portion of the global economy, the challenges and opportunities in managing their human capital are particularly unique. These enterprises often face resource constraints, limited budgets, and a need to compete with larger corporations to attract and retain top talent.

In response to these challenges, the development of a web-based Human Resources and Recruitment Management Application tailored specifically for SMEs has emerged as a promising solution. Such an application can empower SMEs with the tools and capabilities required to streamline their HR and recruitment processes, enhance employee management, and ultimately drive business success.

The importance of this project lies not only in its practical implications for SMEs but also in its relevance to the broader discourse on technological innovation and its impact on business operations. By harnessing the capabilities of modern web-based applications, SMEs can not only improve their HR and recruitment processes but also compete more effectively in the global marketplace. The journey from conceptualization to realization of this application represents a testament to the power of technology in addressing the unique challenges faced by businesses, regardless of their size.

This report details the various phases of the implementation of our project and is organized into six chapters as follows :
We begin by presenting the overall context of the project in the first chapter titled "General Project Framework." In the second chapter, titled "Sprint 0 : Project Foundation Preparation," we focus on defining the business requirements of our application, identifying the technologies used, and presenting the adopted working methodology. From the third to the fifth chapter, we describe the implementation of the work carried out during each sprint, presenting the sprint backlog, use case diagrams, sequence diagrams, and associated accomplishments each time. Finally, we conclude this report with a general summary highlighting the main contributions of this project and possible future prospects.

CHAPTER 1

PROJECT OVERVIEW

Plannig

Introduction

Before delving into the in-depth study of our end-of-study internship, we will briefly describe, in this first chapter, the organization where our internship takes place. Then, we will present a critique of the current situation to unveil the proposed solution. Finally, to conclude, we will devote the last part to the presentation of the methodology adapted for managing this work.

1.1 Presentation of the Host Organization

1.1.1 Description of the Host Organization (Fekr)

Fekr Technologies, is a consultancy founded in 2016 that provides services in software craftsmanship, Devops, AI and Big Data. Its offering is to mind the gap between startups and high-end talent. Fekr makes it possible for emergent structures or incumbents alike thrive in their technological challenges by providing assitance, development and training.

FIGURE 1.1 : Logo of the orgnisation

1.1.2 Description of the IT Service

Figure 1.2 illustrates the different IT services at Fekr :

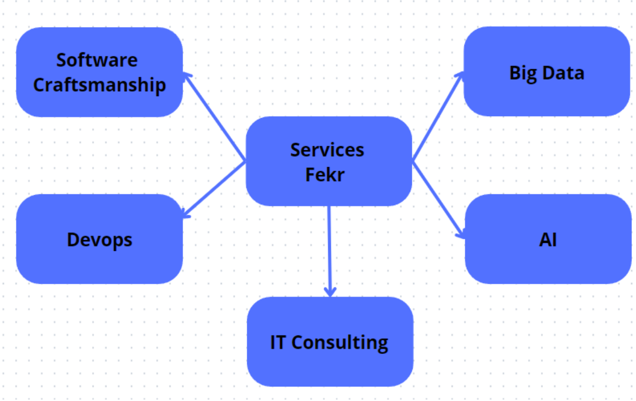


FIGURE 1.2 : Fekr Services

1.1.3 Work Methodology

— **Custom Development :**

- Design custom websites or applications.
- Create specifications.
- Develop custom services/products.
- Make immediate corrections or improvements.

— **Effective IT Consulting :**

- MProvide the skills of IT consultants to conduct a detailed diagnosis of the current situation.
- Propose the most appropriate solutions to address bottlenecks and progress rapidly towards objectives.

— **Practical e-Learning Courses :**

- Offer flexible online learning methods, supervised by competent trainers.
- Provide close monitoring to quickly obtain certification.
- Access all training modules for free with a free account.

1.2 Problematic

While SMEs play a vital role in the global economy, they often face formidable hurdles in effectively managing their Human Resources (HR) and Recruitment processes. The core problematic lies in the inefficiencies and limitations of existing HR and Recruitment Management systems, which are typically designed for larger enterprises. SMEs are left grappling with manual, time-consuming processes, and a lack of affordable, tailored solutions. This situation not only impedes their ability to attract and retain top talent but also hampers their overall competitiveness and growth potential in an increasingly dynamic business environment.

1.3 Existing Situation Analysis

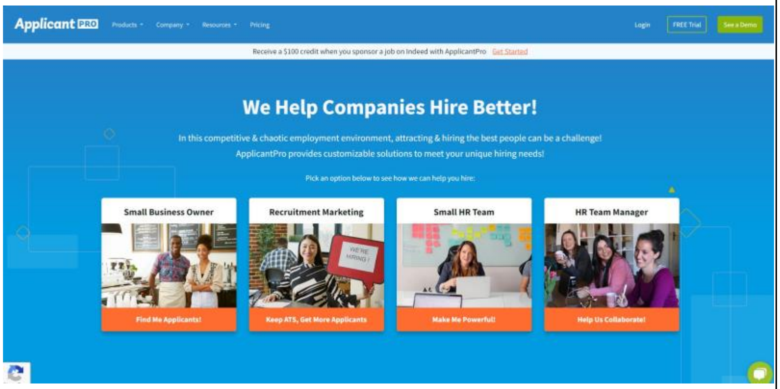
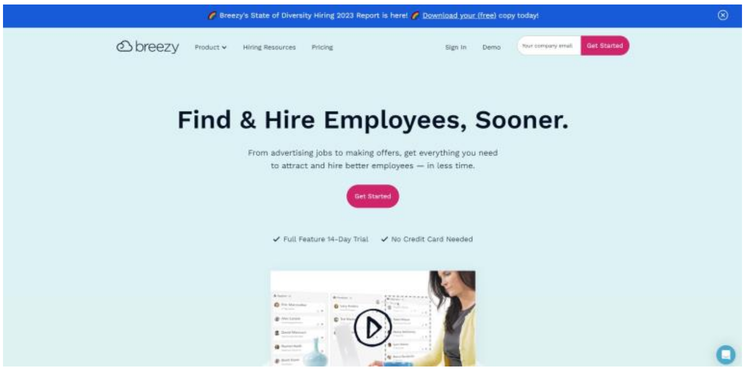
This chapter aims to place our subject in its general context. We will start with an overview of the existing situation, an analysis of similar applications, and a critical evaluation of the current situation. Then, we will propose solutions to improve this workflow.

1.3.1 Description of the Existing Situation :

After analyzing the market, we noticed the existence of web applications dedicated to manage job opportunities, such as "ApplicantPro ," "Breezy HR "etc.

Table 1.1 provides an overview of existing applications and websites that offer services to users.

TablE 1.1 : Overview of Existing Applications and Websites

Captures	**Description**
 Receive a $100 credit when you sponsor a job on Indeed with ApplicantPro Get Started We Help Companies Hire Better! In this competitive & chaotic employment environment, attracting & hiring the best people can be a challenge! ApplicantPro provides customizable solutions to meet your unique hiring needs! Pick an option below to see how we can help you hire: Applicant PRO Products Company Resources Pricing Login FREE Trial See a Demo Small Business Owner Find Me Applicants! Recruitment Marketing Keep ATS, Get More Applicants Small HR Team Make Me Powerful! HR Team Manager Help Us Collaborate!	**ApplicantPro** is a simple and customized applicant tracking solution that is ideal for small and medium-sized businesses.
 Breezy's State of Diversity Hiring 2023 Report is here! Download your (free) copy today! Find & Hire Employees, Sooner. From advertising jobs to making offers, get everything you need to attract and hire better employees — in less time. ✓ Full Feature 14-Day Trial ✓ No Credit Card Needed breezy Product Hiring Resources Pricing Sign In Demo Your company email Get Started Get Started	**Breezy HR :** is a fantastic alternative for small and medium-sized enterprises because it has an intuitively designed applicant tracking system with collaborative hiring capabilities.

1.3.2 Critical Review of the Existing Situation :

The critical review of the existing situation is a crucial phase that comes after studying the current situation. This step aims to identify the problems caused by the use of traditional or classic methods.

- **Inefficient Recruitment Processes :** Many SMBs continue to rely on manual and paper-based recruiting processes, which results in delays, increased costs, and a lack of agility in hiring qualified employees.
- **Heavy Administrative Burden :** HR staff at small and medium-sized businesses frequently have a large administrative effort, from publishing job advertisements to organizing interviews and monitoring applicant data, which can detract from strategic HR initiatives.
- **Lack of Data-Driven Insights :** Many SMBs fail to use data to make educated recruitment decisions, missing out on opportunities to enhance hiring methods and review overall HR performance.
- **Compliance and Diversity Issues :** Maintaining labor law compliance and fostering diversity and inclusion can be difficult, particularly for SMBs with limited resources and expertise in these areas.
- **Integration and Scalability :** SMEs may face technical hundles in integrating HR and recruitment systems with other tools and assuring scalability to meet corporate expansion.
- **Cost Constraints :** SMEs with limited funds may be unable to invest in innovative HR and recruitment solutions, putting them at a competitive disadvantage in attracting top personnel.
- **Subjective Candidate Evaluation :** Conventional hiring processes may include subjective candidate evaluations, which introduce bias and inconsistency into the hiring process.
- **Restricted Access to Talent :** Small and medium-sized businesses (SMBs) may encounter difficulties in contacting a diverse pool of possible candidates, perhaps resulting in missed opportunities to identify the best fit for their job openings.

Table 1.2 provides a comparison of existing applications and websites.

TablE 1.2 : Comparison of Existing Sites

Existing Sites	Advantages	Disadvantages
ApplicantPro	• Effective applicant monitoring. • Talent sourcing tools. • Automation, and compliance management. • Low cost.	• Low scalability. • Changing cost. • Reliance on internet connectivity.
Breezy HR	• Ease of use. • Candidate sourcing tools. • Customization, and integration possibilities.	• Scaling constraints. • Different pricing schemes. • Need on consistent internet connectivity.

1.4 Proposed Solution

In response to the identified problematic related to HR and Recruitment Management for Small and Medium-sized Enterprises (SMEs), we propose the development of a comprehensive web-based HR and Recruitment Management Application tailored specifically to the unique needs and constraints of SMEs. This solution aims to address the challenges outlined in the problematic statement and empower SMEs to effectively manage their HR and recruitment processes. It embodies the commitment to efficiency, affordability, and adaptability, ultimately contributing to the growth and sustainability of SMEs in a dynamic business landscape.

1.5 Work Methodology

Conducting a software project in a team working context requires an understanding of the main work methodologies to ensure an organized and optimal life cycle. It is essential to refer to a unified modeling language to formalize the proposed solution using known concepts and rules that lead to an abstract and simplified representation of the project. Some projects are based on classical methodologies, while others are based on agile methodologies.

1.5.1 Traditional vs. Agile Work Methodologies

Table 1.3 illustrates the comparison between traditional and agile work methodologies.

TablE 1.3 : Tableau de comparaison des méthodes de développement

Theme	**Traditional Methods**	**Agile Methods**
Lifecycle	Sequential phases in a waterfall or V-shaped model, with no possibility of feedback, sequential phases.	Iterative and incremental.
Planning	Predictive, characterized by detailed plans based on a defined and stable scope and requirements at the project's outset.	Adaptive with multiple levels of planning (macro and micro-planning) with adjustments as needed based on changes that occur.
Documentation	Produced in large quantities as a means of communication, validation, and contractualization.	Reduced to the bare minimum in favor of operational functional increments to obtain client feedback.
Team	A team with specialized resources, led by a project manager.	An empowered team where initiative and communication are emphasized, supported by the project manager.
Quality	Quality control at the end of the development cycle. The client discovers the finished product.	Early and ongoing quality control at both the product and process levels. The client visualizes results earlyand frequently.
Change	Resistance and opposition to change. Heavy processes for managing accepted changes.	Favorable reception of inevitable change, integrated into the process.
Progress Tracking	Measurement of compliance with initial plans. Analysis of deviations.	A single progress indicator : the number of implemented features and the remaining work.
Risk Management	Distinct and rigorous risk management process.	Risk management integrated into the overall process, with everyone responsible for identifying and resolving risks.

1.5.2 The Four Fundamental Values of Agile Methodology

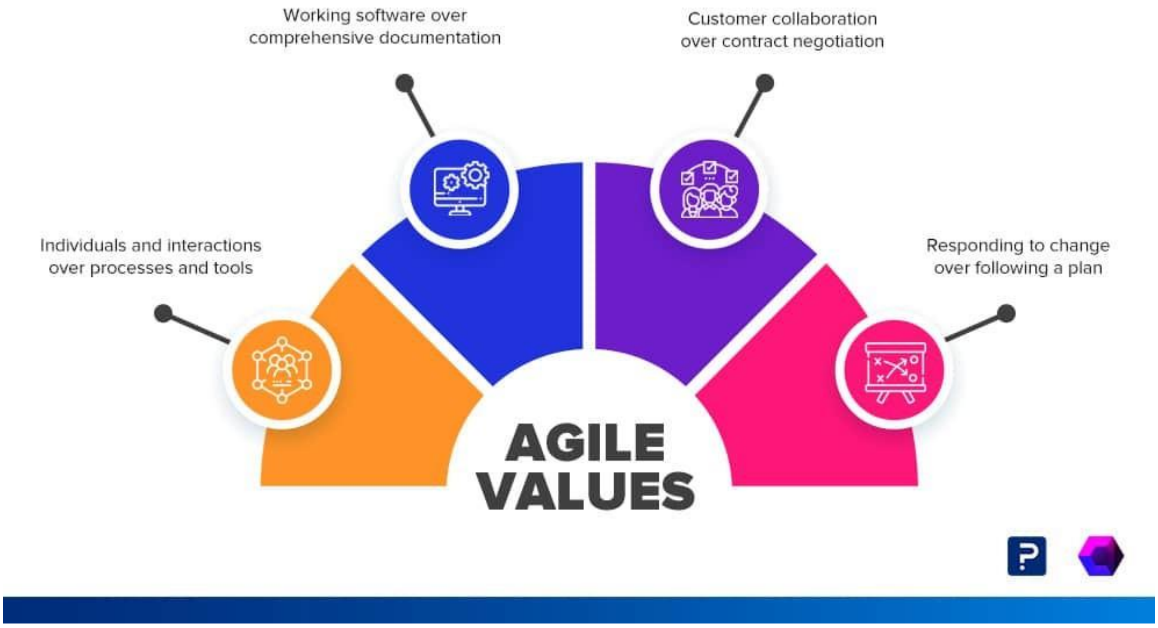

Working software over comprehensive documentation
Customer collaboration over contract negotiation
Individuals and interactions over processes and tools
Responding to change over following a plan
AGILE VALUES

FIGURE 1.3 : Agile Values

1.5.3 The Twelve Agile Principles

Table 1.4 lists the twelve Agile principles :

TablE 1.4 : The Twelve Agile Principles

1	Prioritize customer satisfaction.
2	Embrace changes.
3	Deliver working software frequently.
4	Foster collaboration between the project team and business stakeholders.
5	Build projects around motivated individuals.
6	Favor direct communication.
7	Measure project progress primarily through working software.
8	Maintain a sustainable pace of work for all team members.
9	Ensure continuous attention to technical excellence and good design.
10	Minimize unnecessary work.
11	Empower and trust team members.
12	Regularly reflect on team performance and adjust behavior accordingly.

1.5.4 Key Agile Methodologies

The most popular Agile methodologies today include :

— Extreme Programming (XP)

— Scrum

— Feature Driven Development (FDD)

— Agile Unified Process (Agile UP or AUP)

— Dynamic Systems Development Method (DSDM)

After studying various methodologies, and to ensure the successful execution of different phases of our project, we have chosen Scrum as the preferred methodology for design and development.

1.5.5 Scrum as the Adopted Agile Method

Presentation and Description

"SCRUM is considered a project management framework. This framework consists of role definitions, meetings, and artifacts. SCRUM is by far the most widely used methodology among existing Agile methods." [1] At this stage, we have chosen the SCRUM method because it is best suited to our project, based on its advantages :

- Adaptation to change
- Self-organizing teams
- Collaboration
- Transparency
- Task decomposition and understanding
- Adherence to deadlines
- Flexibility
- Speed of execution and delivery
- Client involvement

Roles in SCRUM

The SCRUM method defines three roles :

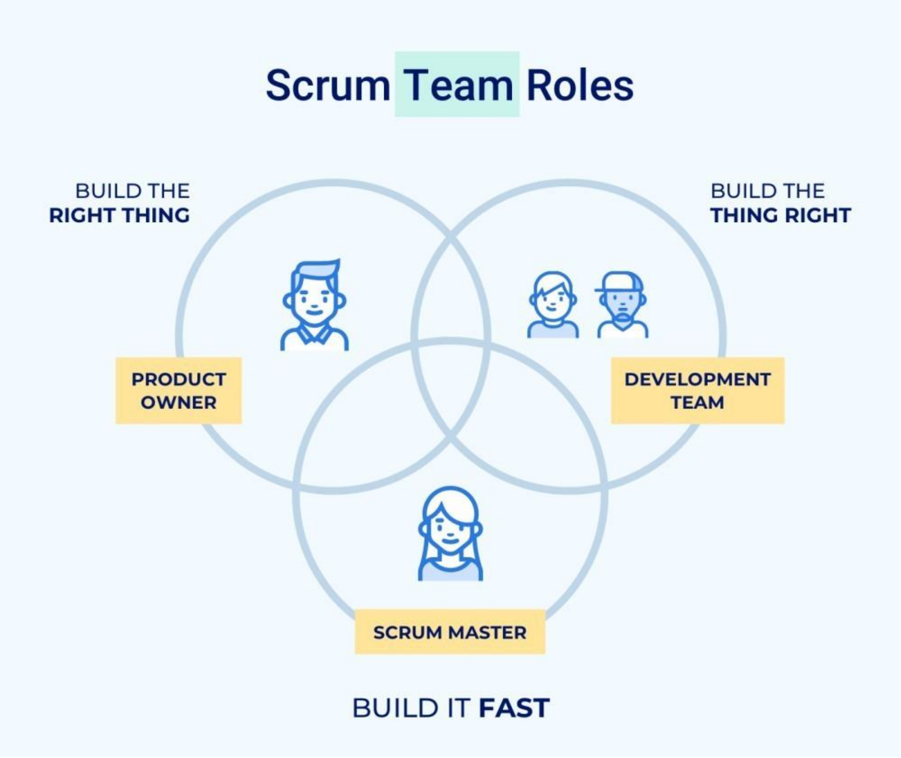


Figure 1.4 : SCRUM Team

- **Product Owner :** The representative of the client in the organization, responsible for defining the main needs and characteristics of the product and overseeing its functional quality. The Product Owner should also have a clear vision of the project and validate its progress to ensure successful completion.
- **SCRUM Master :** Responsible for ensuring that the SCRUM methodology is followed and guiding the team in its application. The SCRUM Master assists team members and helps them improve to ensure maximum team productivity.
- **SCRUM Team (Development Team) :** The development team should consist of all the expertise necessary for project realization. Team members are responsible for executing the work as defined by the Product Owner. During sprints or iterations, they develop and test various user stories to produce a technically sound product.

SCRUM Artifacts

SCRUM artifacts involve creating information definitions that are understandable by everyone. There are three SCRUM artifacts :

- **Product Backlog :** These are the features demanded by the Product Owner of the project, organized as a list of tasks to be completed, ordered by priority. The Product Owner is responsible for continuously updating this list to ensure alignment with ongoing client needs.
- **Sprint Backlog :** This can be defined as a commitment from the team since it represents the objective of the current sprint. It contains a subset of the Product Backlog that includes the features to be completed by the end of the sprint.
- **Burndown Chart :** This is a document that should be continuously updated, presenting the project's progress and allowing visualization of the project's state at any given time. This document is managed by the SCRUM Master.

Sprint Activities

- **Daily SCRUM (Daily Standup) :** A daily meeting that gathers the development team with the SCRUM Master to synchronize work progress every 24 hours. This meeting should not exceed 15 minutes and is typically held at the beginning of the day.
- **Sprint Planning :** A meeting that brings together the development team and the Product Owner to choose the set of features, a subset of the Product Backlog, to be executed during the sprint.
- **Sprint Review :** A meeting that gathers the entire SCRUM team at the end of each sprint to ensure the deliverable's compliance with the Product Owner's requirements.
- **Sprint Retrospective :** A meeting held at the end of each sprint following the review, with the goal of learning from the completed sprint to improve future ones.

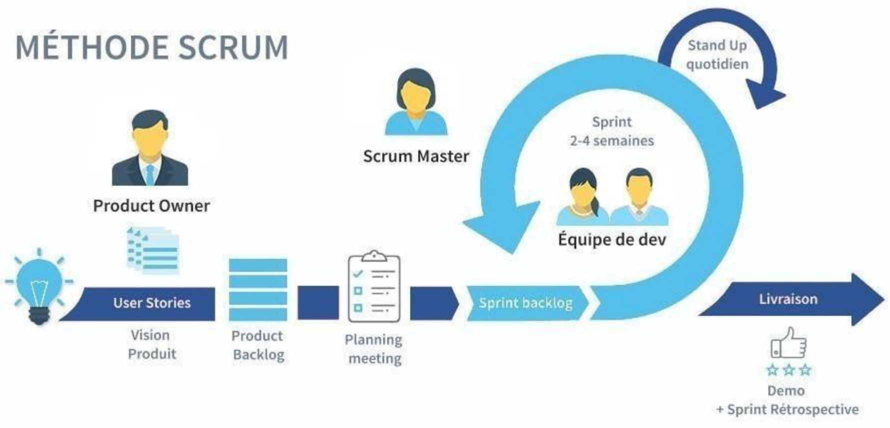


FIGURE 1.5 : SCRUM Process

1.6 Adopted Modeling Language

To make the system understandable to professionals and other users, software developers need a visual representation that facilitates information transmission. Therefore, they use modeling and diagramming using the UML (Unified Modeling Language) tool.

"UML, Unified Modeling Language, is a standardized, precise, and stable visual language consisting of a set of diagrams used to provide different views of the project being developed. UML provides diagrams to represent how the software functions, its deployment, actions the software can perform, and more. Creating these diagrams is a way of modeling the software's requirements." [3]

Not only does UML offer standardization, precision, and stability, but it also provides a variety of diagrams. This is why we have chosen UML as a modeling language to improve the understanding and simplification of our project.

Conclusion

In the first chapter, we described the general framework of our project and the organization that has welcomed us for our end-of-study internship. Then, we conducted an analysis of the existing system, which led to the proposal of our solution, as described earlier. We also presented our working method and the modeling language. In the following chapter, we will introduce the actors who will interact with the proposed system, as well as the functional and non-functional requirements of our project.

CHAPTER 2

SPRINT 0 : PROJECT FOUNDATION PREPARATION

Plannig

Introduction

As mentioned in the previous chapter, we have chosen to adopt the Scrum methodology for the design of our platform. In this chapter, we will carry out the first step of the Scrum methodology, during which we will identify user

2.1 Requirements Specification

In this section, we will present the actors, their roles, and the privileges they have. We will introduce each actor along with their definition.

2.1.1 Actors Identification

An actor represents a person or another computer system that benefits from one or more services offered by an access interface and interacts with the system by sending or receiving messages. The actors identified in the context of our project are as follows :

- **SME HR Managers :** are responsible for overseeing the entire HR and recruitment process within their organizations. They are pivotal in creating and managing job listings, evaluating candidate applications, and making hiring decisions.
- **Applicants :** are individuals seeking job opportunities within SMEs. They come from diverse backgrounds and submit job applications through the application platform.

2.1.2 Static Context Diagram

A static context diagram allows us to identify the relationship between different actors in relation to our system, specifying the number of instances of each actor connected to the system at a given moment.

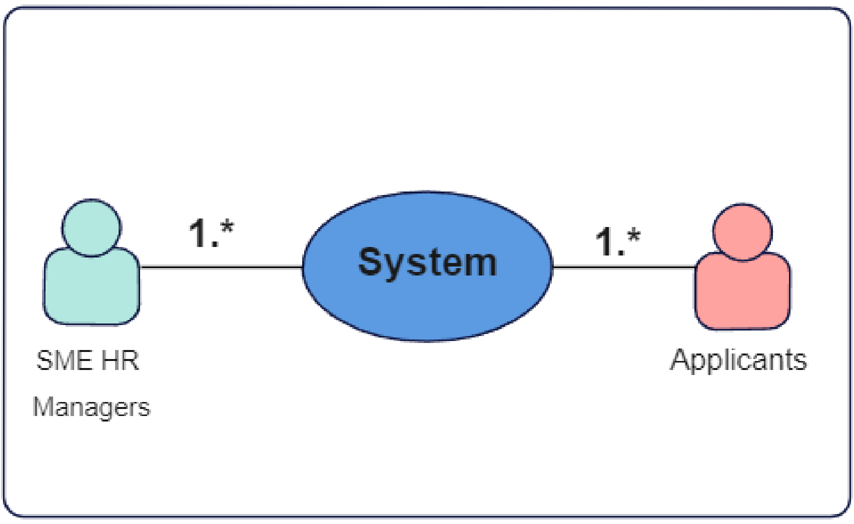


FIGURE 2.1 : Static Context Diagram

2.1.3 Description of Functional Requirements

In this section, we will present the various functionalities and services provided by our system for the different actors identified in the previous section :

— **Functional Requirement for the "SME HR Managers" Actor :**

- Create and post detailed job listings, including job titles, descriptions, responsibilities, qualifications, location, and application deadlines.
- Preview job listings before publishing to ensure accuracy.
- Sort candidate applications by date, name, and score.
- Access candidate profiles with contact information, resume uploads.
- Visualize scores assigned automatically to applicant by the AI-driven resume parsing.
- Make application decision by approving or rejecting application based on the scores
- view, modify, and desactivate user accounts.
- Access and review detailed job listings.
- Access and view statistical reports on a dashboard.

— **Functional Requirement for the "Applicants" Actor :**

- Register to the application.
- View and update profile informations.
- Search and view job listings.
- modify profile's personnal informations.

  - Upload resumes and other required documents.
  - Receive notifications regarding application status updates.
  - apply to a job opportunity.

— **Common Features Among the Four Actors :**

  - Authentication : Our system provides each user with a login and password to securely access resources.

2.1.4 Description of Non-Functional Requirements

The non-functional requirements of our system are as follows :

- **Security Requirements :** All access to different areas must be protected by a password and access privilege. Therefore, it is necessary to ensure data encryption at the database level.
- **Usability Requirements :** The user interface must be clear and easy to use.
- **Reliability :** The application must always provide correct results to users.
- **Organization :** The code must be clear to allow for possible future evolutions or improvements.
- **Speed :** Reasonable execution and processing times.

2.2 Global Use Case Diagram

The global use case diagram illustrates and defines the requirements for a complete system. In this context, we generally present the functionalities offered by our application. Figure 2.1 illustrates the global use case diagram.

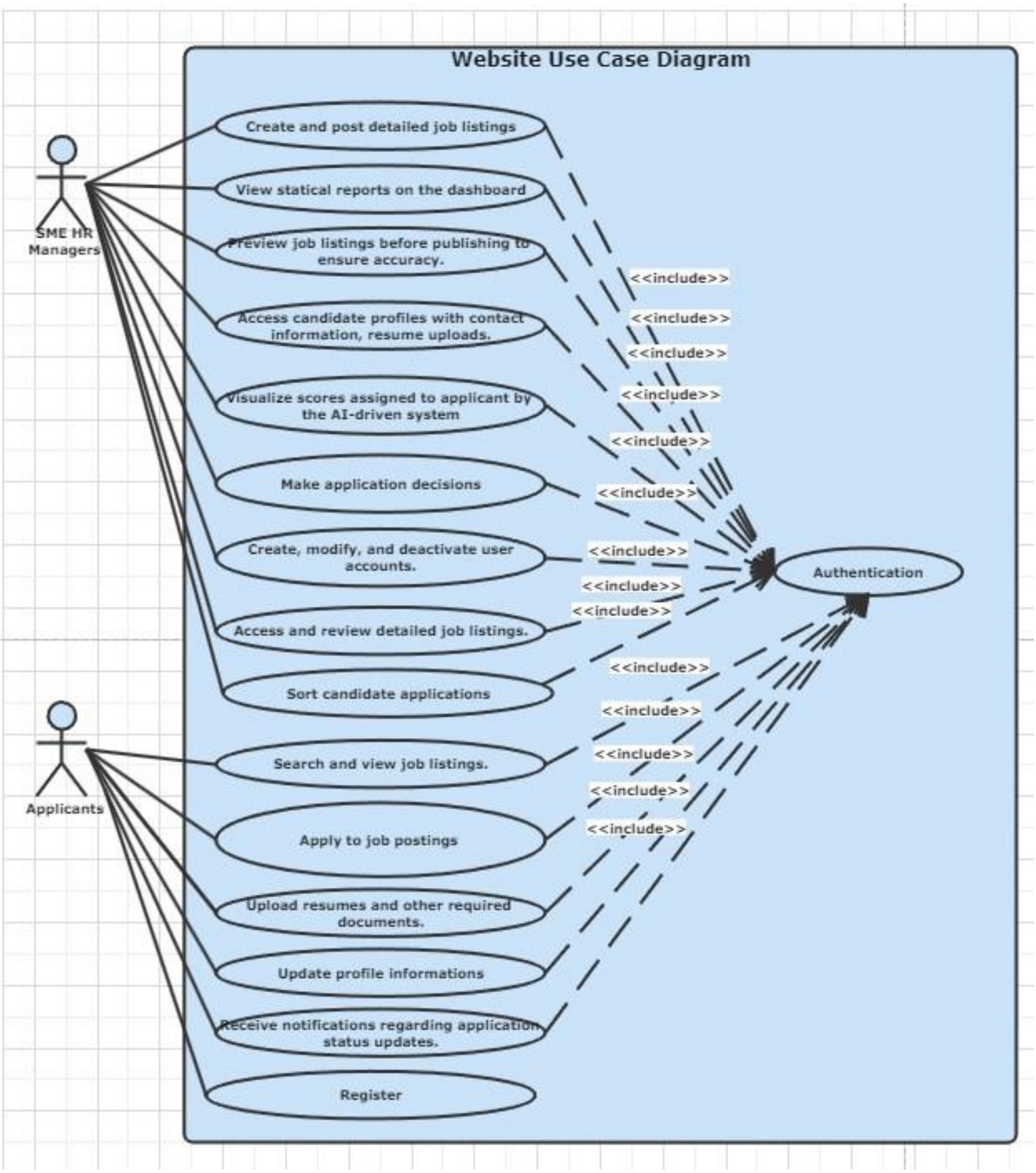

Website Use Case Diagram
SME HR Managers
Applicants
Create and post detailed job listings
View statical reports on the dashboard
Preview job listings before publishing to ensure accuracy.
Access candidate profiles with contact information, resume uploads.
Visualize scores assigned to applicant by the AI-driven system
Make application decisions
Create, modify, and deactivate user accounts.
Access and review detailed job listings.
Sort candidate applications
Search and view job listings.
Apply to job postings
Upload resumes and other required documents.
Update profile informations
Receive notifications regarding application status updates.
Register
Authentication
<<include>>

FIGURE 2.2 : Global Use Case Diagram

2.3 Project Management with Scrum

First, we start by designating the SCRUM team. Then, we continue by presenting the Product Backlog. Finally, we finish by planning the Releases.

2.3.1 Identifying the Scrum Team

The SCRUM method is a software development methodology that relies on team collaboration. In the context of our project, the SCRUM team is as follows :

Roles are distributed as follows :

- **Scrum Master :** Supervisor Ms. Dhouha maatar
- **Product Owner :** Supervisor Mr. Rafik naccahe
- **Development Team :** Mohamed Gundia

2.3.2 Product Backlog

As we defined in the first chapter, the product backlog is a very important artifact that represents the expected requirements and functionalities.

Table 2.1 summarizes the Product Backlog for our solution :

Table 2.1 : Product Backlog

User Story	Description	Estimation	Priority	Risk	Sprint
US 1	As an applicant, I should be able to register.	High	1	3	Sprint No.1
US 2	As an SME HR Manager or applicant, I should be able to authenticate.	High	1	2	Sprint No.1
US 3	As an SME HR Manager, I should be able to visualize scores assigned automatically to applicant by the AI-driven resume parsing.	High	1	2	Sprint No.1
US 4	As an SME HR Manager, I should be able to make application decision by appro- ving or rejecting application based on the scores.	High	2	2	Sprint No.1
US 5	As an applicant, I should be able to Upload resumes and other required documents.	High	2	3	Sprint No.1
US 6	As an SME HR Manager, I should be able to create and post detailed job listings, including job titles, descriptions, responsibilities, qualifications, location, and application deadlines.	Medium	3	3	Sprint No.2
US 7	As an SME HR Manager, I should be able to preview job listings before publishing to ensure accuracy.	Low	2	3	Sprint No.2
US 8	As an SME HR Manager, I should be able to sort candidate applications by date, name, and score.	Medium	2	3	Sprint No.2
US 9	As an applicant, I should be able to update my profile.	High	2	3	Sprint No.1
US 10	As an applicant, I should be able to search and view job listings.	Medium	2	3	Sprint No.2
US 11	As an applicant, I should be able to receive notifications regarding application status updates.	Medium	4	2	Sprint No.2

US 12	As an SME HR Manager, I should be able to modify, and desactivate user accounts.	Medium	3	3	Sprint No.3
US 13	As an SME HR Manager, I should be able to access candidate profiles with contact information, resume uploads.	Medium	2	3	Sprint No.3
US 14	As an HR manager, I should be able to access and view statistical reports on my dashboard.	Medium	2	3	Sprint No.3

2.3.3 Sprint Structure

Sprint Planning

In order to ensure the success of a project, sprint planning is of paramount importance for the smooth progress of work. The planning phase allows for a clear and organized schedule and approach to optimally divide and assign tasks. The table below illustrates the sprint planning.

Table 2.2 : Sprint Backlog 1

Sprint No.	Start Date	End Date	Sprint Name
1	25/07/2023	15/08/2023	- Authentification - Application scoring
2	16/08/2023	07/09/2023	-job opportunities Management
3	08/09/2023	30/09/2023	Users Management

2.3.4 Project Implementation Schedule

The key to the success of a project is a good schedule. Indeed, the schedule helps to divide the work effectively and separate the tasks to be performed, providing a better estimate and management of the time required for each task. Furthermore, it provides enough visibility to approximately estimate the completion date of each task. In our project, we have estimated that our application will be developed in an approximate duration of 4 months.

The figure 2.5 below shows the schedule we have adapted to successfully implement the various parts of the project.

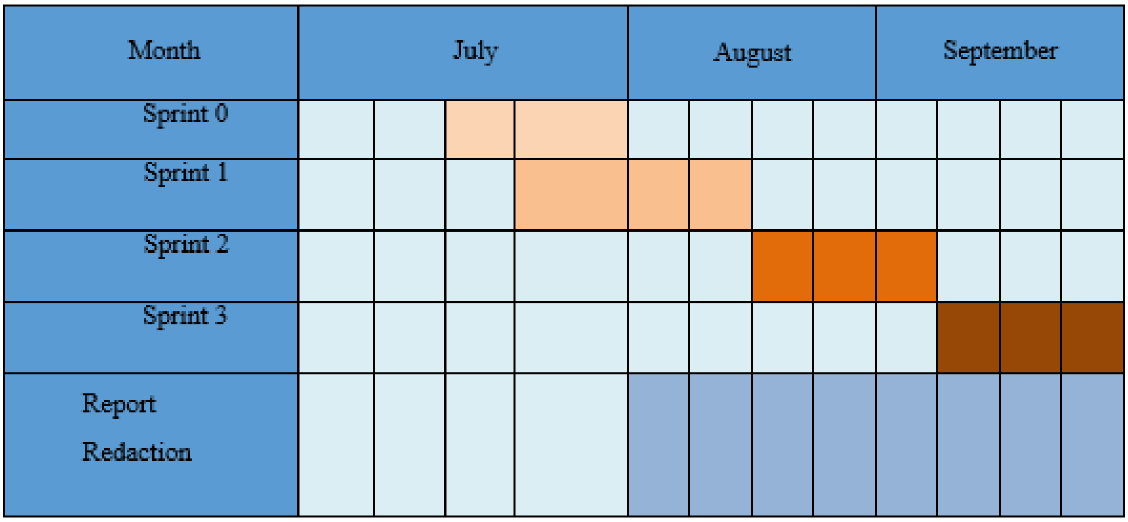


Figure 2.3 : Sprint Planning Gantt Chart

2.4 Working Environment

In this section, we focus on our working environment. Firstly, we present our hardware environment, and secondly, our technological choices related to the software environment by detailing the different technologies and tools used during the development of our solution.

2.4.1 Hardware Environment

We will present the hardware environment in which our application will be developed. The characteristics of the machine are listed in the table below.

Table 2.3 : Machine Characteristics

Machine	**Characteristics**
Lenovo Ideapad Gaming PC	**Operating System :** Windows 11 **Processor :** 11th Gen Intel(R) Core(TM) i5-11320H **Memory :** 16 GB **Hard Drive :** 474 GB **Graphics Card :** NVIDIA Geforce RTX

2.4.2 Software Environment

In this part, we present the software environment of our application in terms of languages and software.

Languages Used

In table 2.3, there is a description of the languages used during development, such as HTML, CSS for the graphical aspect of the application.

Table 2.4 : Programming Languages Used

Languages	Description	Logo
HTML	Markup language used for creating web pages, allowing the definition of hyperlinks.[4]	HTML
CSS	Cascading Style Sheets (CSS) is a document in which we define color choices, font types, positioning, etc.[5]	CSS
Python	Python is a high-level programming language known for its simplicity and readability. It's commonly used in various applications, including web development, data analysis, machine learning, and more.[6]	python

Used Software

TablE 2.5 : Used Software

Software	Description	Logo
Visual Studio Code	Visual Studio Code is an extensible code editor developed by Microsoft for Windows, Linux, and MacOS. [7]	
Diagrams.Net	Diagrams.net (draw.io) is a free online solution that allows you to create diagrams and flowcharts. [8]	
Postman	Postman is a free software that allows you to make API requests without coding. [9]	POSTMAN

The table above provides a description of the software used during the development of the application.

Frameworks & Technologies

Table 2.5 illustrates the frameworks and technologies used.

TablE 2.6 : Frameworks & Technologies

Technologies	Description	Logo
React JS	React is an open-source JavaScript library developed and maintained by Facebook. It is primarily used for building user interfaces (UI) for web applications. [10]	React JS
Django	Django is a high-level Python web framework that simplifies the process of building web applications. It follows the "batteries-included" philosophy, providing developers with a set of pre-built components and tools for common web development tasks. [11]	django
PostgreSQL	PostgreSQL is a powerful open-source relational database management system. PostgreSQL is known for its reliability, robustness, and SQL standards compliance. It can handle large volumes of data and supports a wide range of advanced features. [12]	PostgreSQL

2.5 System Architecture

2.5.1 Logical Architecture

Our application has been developed using the MVC (Model View Controller) architecture. It is an architectural model that aims to clearly separate the presentation layer (UI : User Interface), the business layer (BLL : Business Logic Layer), and data access (DAL : Data Access Layer). The goal is to minimize dependencies between different layers of the application, so changes made to any layer do not affect the others.

- **Model :** This is the core of the application that manages data, retrieves information from the database, and organizes it for processing by the controller.
- **View :** It is a graphical component of the interface that presents the data from the model to the user.
- **Controller :** It is responsible for decision-making, managing the code logic that makes decisions, and acts as an intermediary between the model and the view.

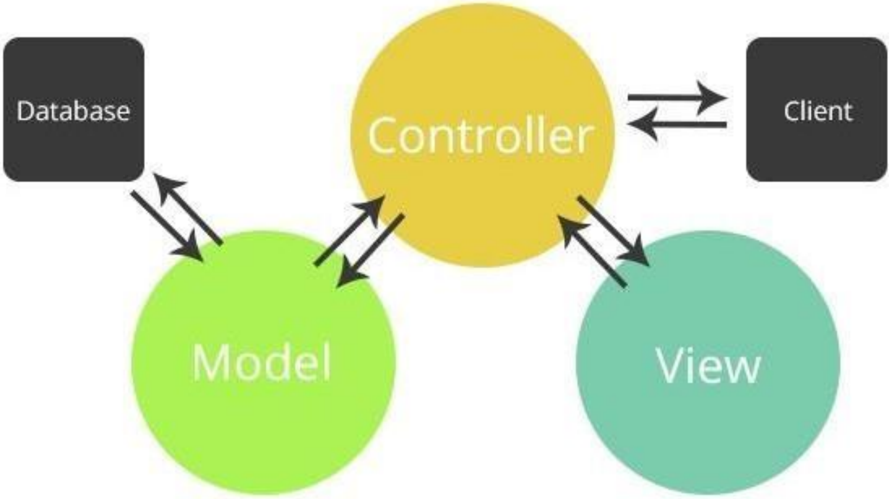


FIGURE 2.4 : The MVC Model

2.5.2 Physical Architecture

To implement our project, we adopted a 3-tier architecture. This is a logical model of application architecture that models an application as a stack of three software layers, each with a clearly defined role :

- **Data Presentation :** This corresponds to displaying and rendering data on the user's workstation, as well as interacting with the user.
- **Business Data Processing :** This corresponds to implementing all business rules and application logic.
- **Access to Persistent Data :** This corresponds to data that is intended to be stored for the long term, possibly permanently.

Figure 2.7 illustrates a more detailed 3-tier architecture.

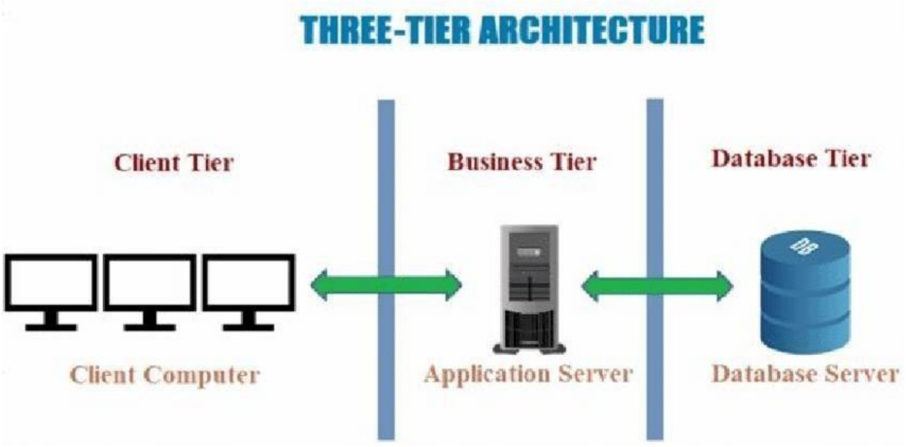


FIGURE 2.5 : 3-tier architecture

2.5.3 Database Architecture

Our database structure is designed to store information about users, job opportunities, job applications made by users, and associated documents. It establishes relationships between the tables through foreign keys to maintain data integrity. The primary keys ensure that each record in a table has a unique identifier.

The table bellow represents the database structure.

Table 2.7 : Database Tables

Table	Column	Data Type	Key

User	id
	full_name
	username
	password
		INT
		VARCHAR(255)
		VARCHAR(30)
		VARCHAR(128)
			Primary Key

Opportunity	id title select_value description due_date created_at	INT VARCHAR(255) VARCHAR(10) TEXT DATE DATETIME	Primary Key
Application	id user_id opportunity_id	INT INT INT	Primary Key Foreign Key to User Foreign Key to Opportunity
Document	id created_date document_url title user_id	INT DATETIME VARCHAR(200) VARCHAR(255) INT	Primary Key Foreign Key to User

Conclusion

In this section, we identified the actors, functional requirements, and non-functional requirements of the project in order to plan the sprints. Additionally, we created the global use case diagram and established the Product Backlog. In the next chapter, we will present our first sprint, which represents the initial delivery of our project.

CHAPTER3

SPRInT 1 : PROfILE ManagemenT

Plannig

Introduction

In the previous chapter, we examined in detail the various functional requirements related to our project. Then, we proceeded to break down the project to effectively plan the different work phases. This chapter focuses on the first sprint of our project, interfaces developed for validation, and the retrospective of each sprint.

3.1 Sprint 1 Backlog

The various tasks related to the completion of the first Sprint are described in the Sprint 1 Backlog (see Table 3.1).

TablE 3.1 : Sprint 1 Backlog

ID	User story	Sub-tasks
Us 1	As an applicant, I must be able to register.	**1.** Create detailed use case, class, and sequence diagrams for the "Registration" case. **2.** Design the registration interface (Front-end). **3.** Implement the code for registration within the application (Back-end). **4.** Test the registration process.
Us 2	As an SME HR Manager or applicant, I must be able to authenticate.	**1.** Create detailed use case, sequence, and class diagrams for the "Authenticate" case. **2.** Design the authentication interface (Front-end). **3.** Implement the code for authentication within the application (Back-end). **4.** Test the authentication process.
Us 3	As an SME HR Manager, I must be able to visualize scores assigned automatically to applicant by the AI-driven resume parsing .	**1.** Design the scoring interface (Front-end). **2.** Implement the code for scoring within the application (Back-end). **3.** Test the scoring process.

Us 4	As an SME HR Manager, I must be able to make application decision by approving or rejecting application based on the scores.	**1.** Design the decision interface (Front-end). **2.** Implement the code for the decision within the application (Back-end). **3.** Test the decision process.
Us 5	As an applicant, I must be able to Update my profile	**1.** Create detailed use case, sequence, and class diagrams for the "Profile Modification" case. **2.** Design the modification interface (Front-end). **3.** Implement the code for the modification within the application (Back-end). **4.** Test the modification process.

3.2 Detailed Use Case Diagram of the First Sprint

Detailed use cases are more specific and provide more detailed information compared to the overall diagram. They describe the structure of essential features for system users.

Figure 3.1 presents the use case diagram for the first sprint.

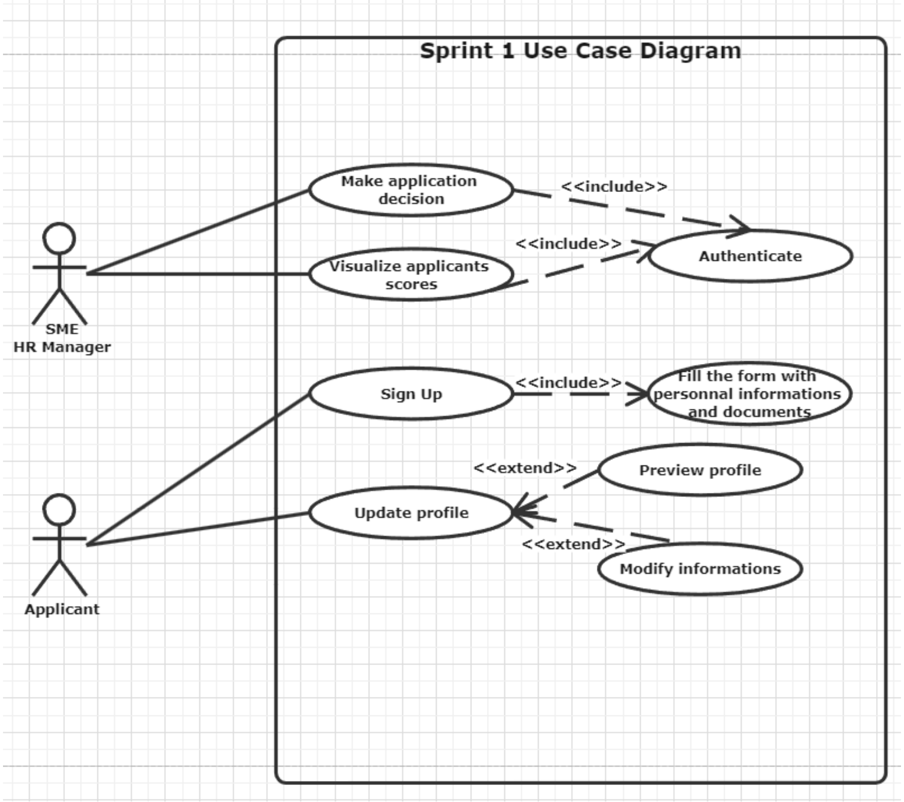


FIGURE 3.1 : First Sprint Use Case Diagram

3.3 Class Diagram for Sprint 1

A class diagram is a graphical representation of the static structure of a software system, showing classes, attributes, methods, and relationships between classes. It allows for modeling key concepts and interactions between different parts of the system.

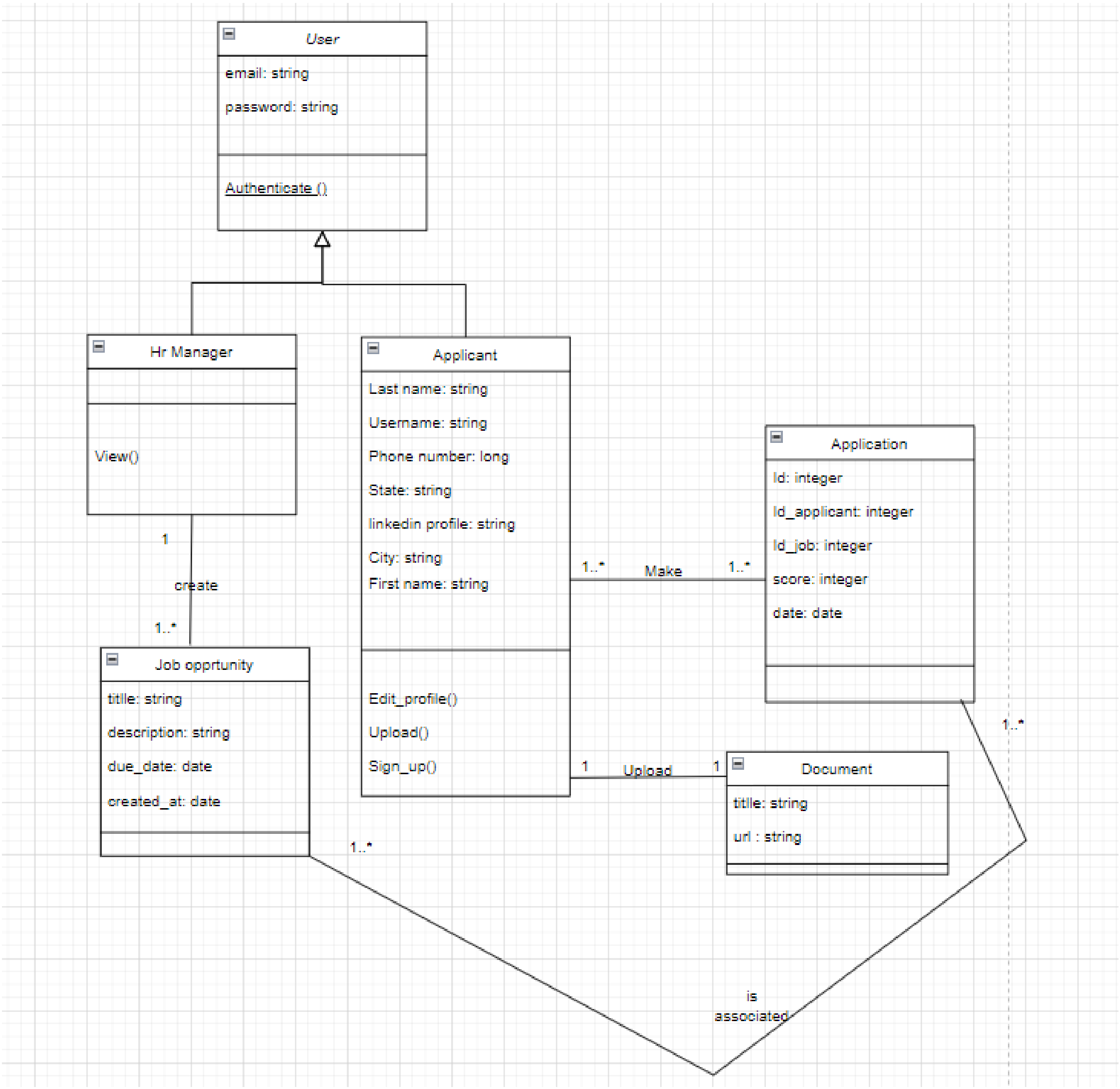


FIGURE 3.2 : Class Diagram Sprint 1

3.4 Sequence Diagrams for Sprint 1

Sequence diagrams are often used to visualize and understand the control flow and interactions between different elements of a system. They can be used at various stages of the software development process, from initial design to documenting and communicating system features.

3.4.1 Sequence Diagram for "Sign Up" Use Case

- **Textual Description of Use Case**

Table 3.2 : Textual Description of Use Case : Sign Up

Use Case	Sign Up
Actor	Applicant
Precondition	The applicant accessed to the registration interface.
Post-condition	The applicant is registred.
Description of the Main Scenario	**1.** The Applicant fill the registartion form with his personnal inforamtion. **2.** The applicant upload his cv. **3.** The applicant click on the register button.
Exception	**Exception 1 :** If some fields are left empty, the system displays an error message next to each empty field.

Figure 3.3 illustrates the sequence diagram for the "Sign Up" use case.

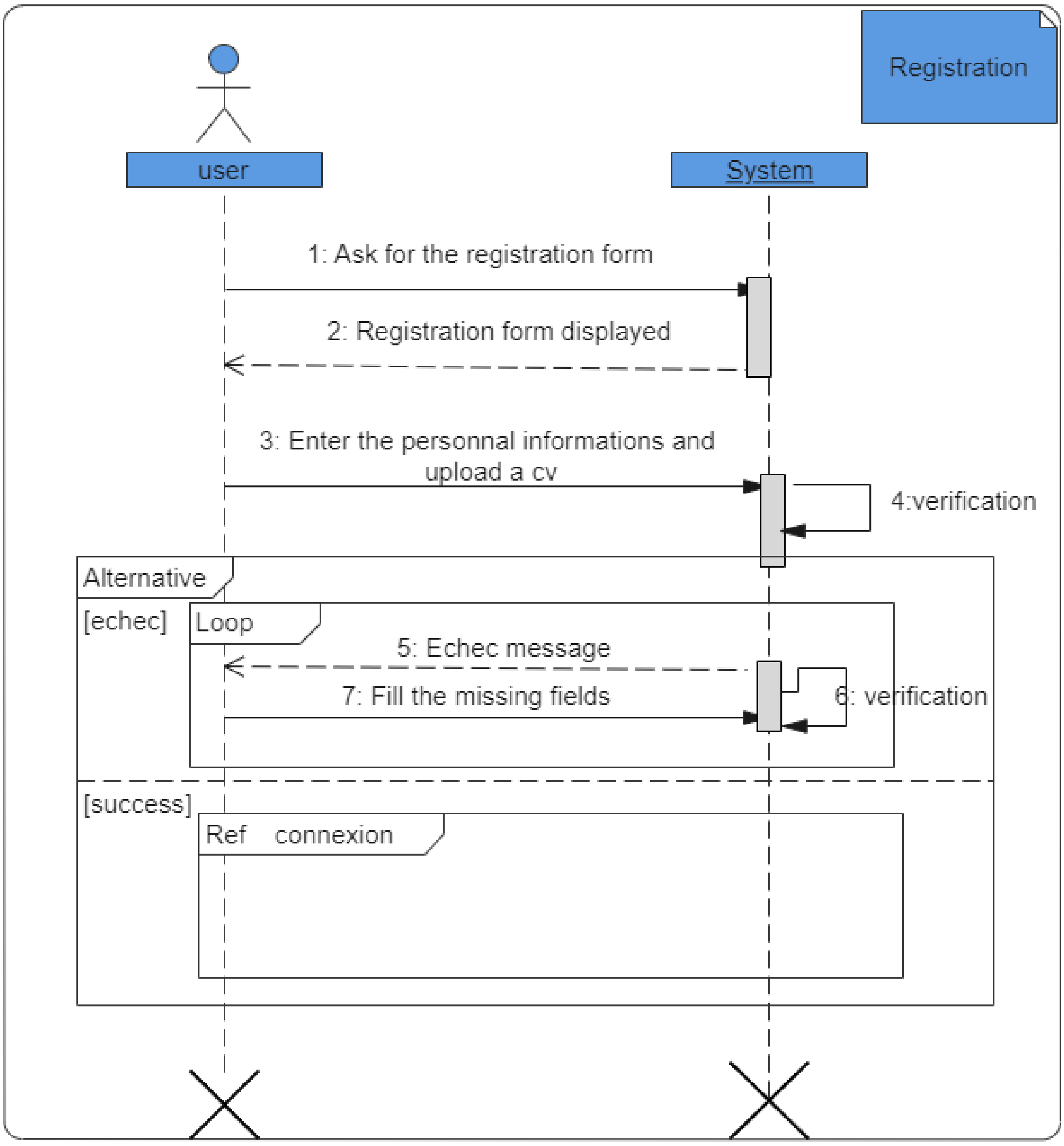

Registration
user
System
1: Ask for the registration form
2: Registration form displayed
3: Enter the personnal informations and upload a cv
4:verification
Alternative
[echec]
Loop
5: Echec message
7: Fill the missing fields
6: verification
[success]
Ref connexion

FIGURE 3.3 : Sequence Diagram for "Sign in" Use Case

3.4.2 Sequence Diagram for "Authentication" Use Case

- **Textual Description of Use Case**

Figure 3.4 illustrates the sequence diagram for the "Authenticate" use case.

Table 3.3 : Textual Description of Use Case : Sign In

Use Case	Sign In
Actor	Applicant
Precondition	The applicant is registred.
Post-condition	The applicant is authenticated.
Description of the Main Scenario	**1.** The Applicant enter the email and password of his account. **2.** The applicant click on the Sign in button.
Exception	**Exception 1 :** If the applicant's login credentials are incorrect, the system displays an error message and prompts the applicant to retry. **Exception 2 :** If some fields are left empty, the system displays an error message next to each empty field.

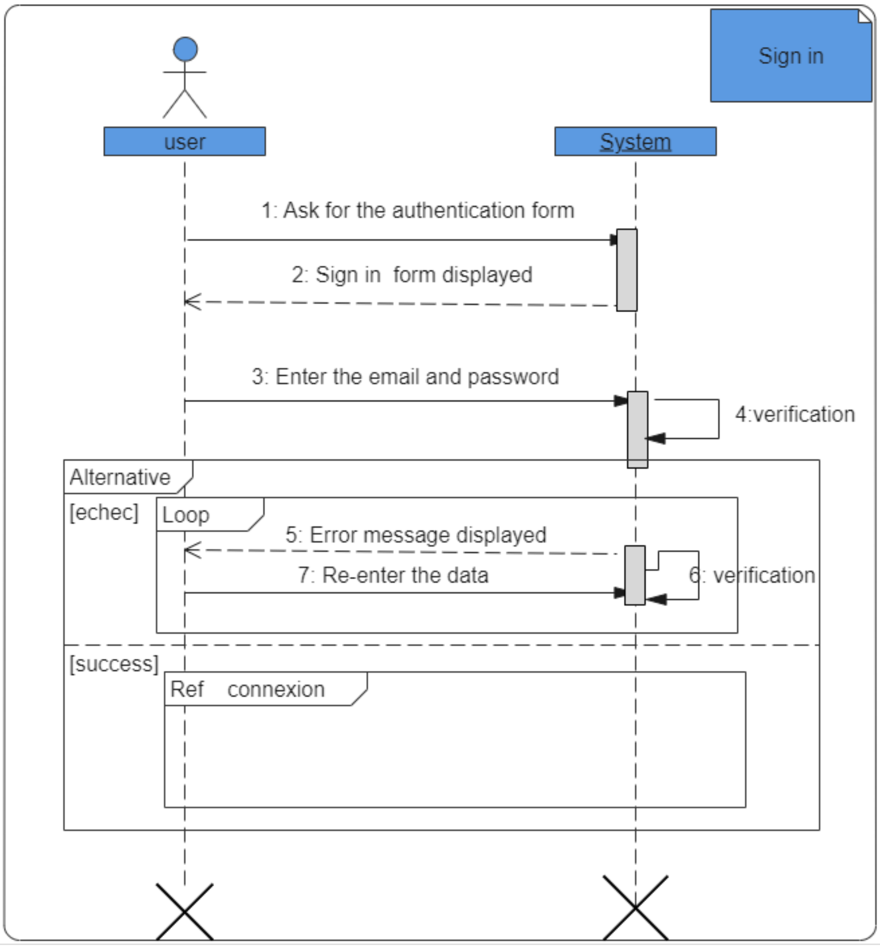

Sign in
user
System
1: Ask for the authentication form
2: Sign in form displayed
3: Enter the email and password
4:verification
Alternative
[echec]
Loop
5: Error message displayed
7: Re-enter the data
6: verification
[success]
Ref connexion

Figure 3.4 : Sequence Diagram for "Authenticate" Use Case

3.4.3 Sequence Diagram for "Edit Profile" Use Case

- **Textual Description of Use Case**

TablE 3.4 : Textual Description of Use Case : Edit profile

Use Case	Edit profile
Actor	Applicant
Precondition	The applicant is authenticated.
Post-condition	The profile applicant is updated.
Description of the Main Scenario	**1.** The Applicant access to his profile. **2.** The applicant click on the edit button. **3.** The applicant changed the desired informations. **2.** The applicant click on the button "Submit".

Figure 3.5 illustrates the sequence diagram for the "Edit Profile" use case.

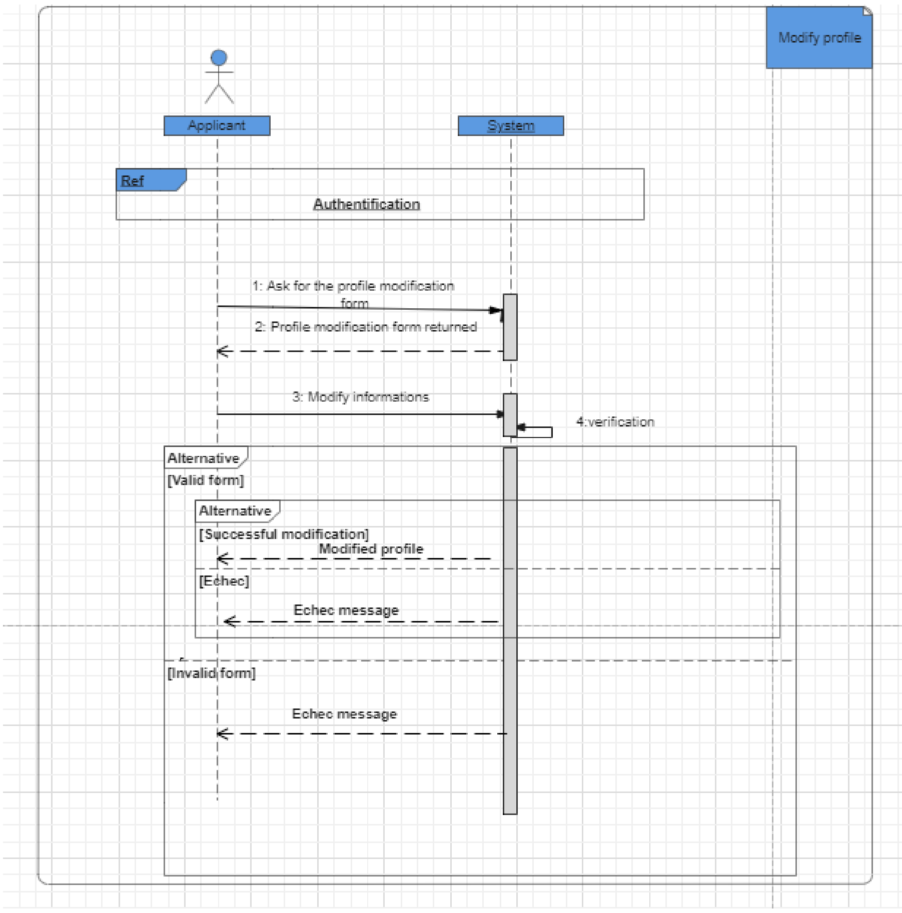


FIGURE 3.5 : Sequence Diagram for "Edit Profile" Use Case

3.5 Detailed sequence diagrams

3.5.1 Detailed sequence diagram for "Registration" use case

Figure 3.4 illustrates the detailed sequence diagram for the "Registration" use case.

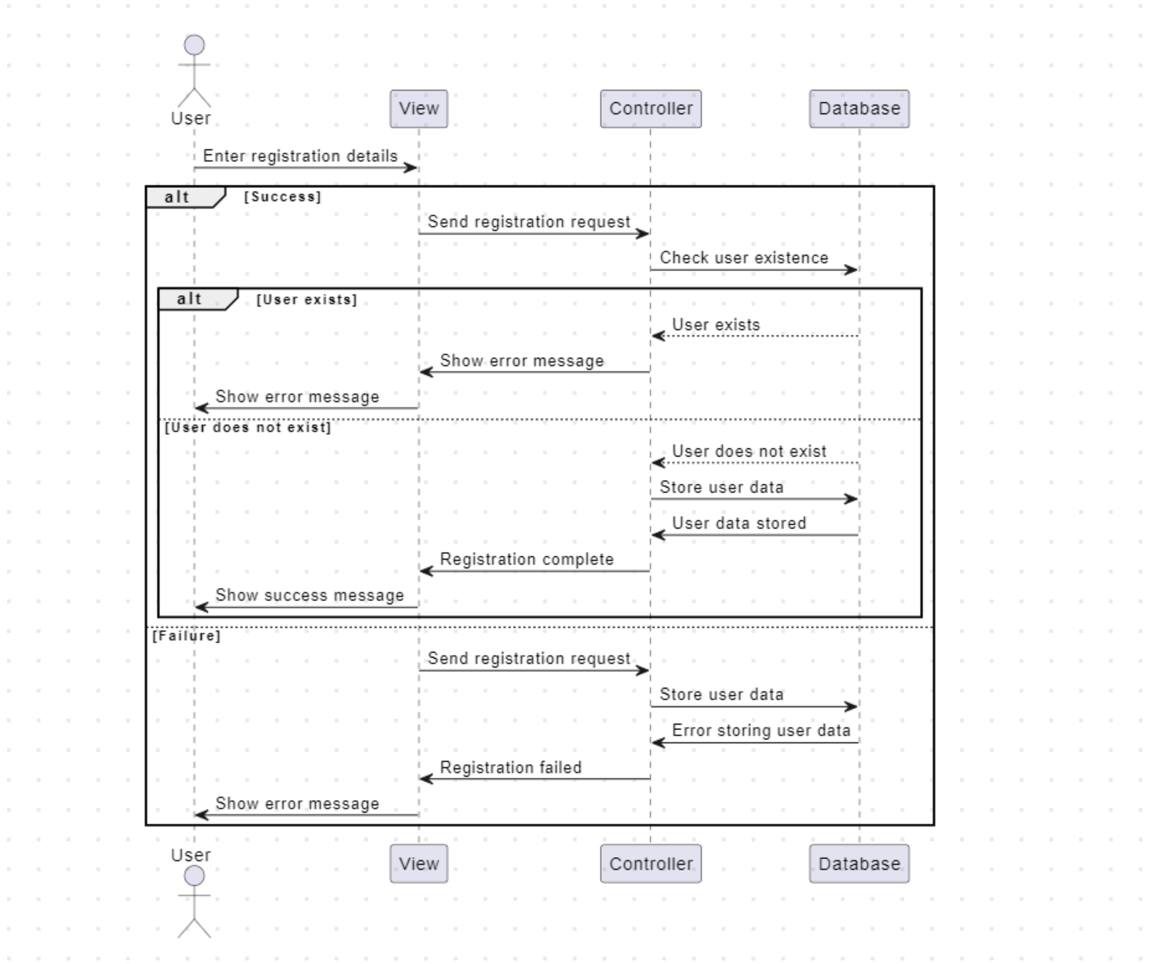


FIGURE 3.6 : Sequence Diagram for "Registarion" Use Case

3.5.2 Detailed sequence diagram for "Authentication" use case

Figure 3.4 illustrates the detailed sequence diagram for the "Authentication" use case.

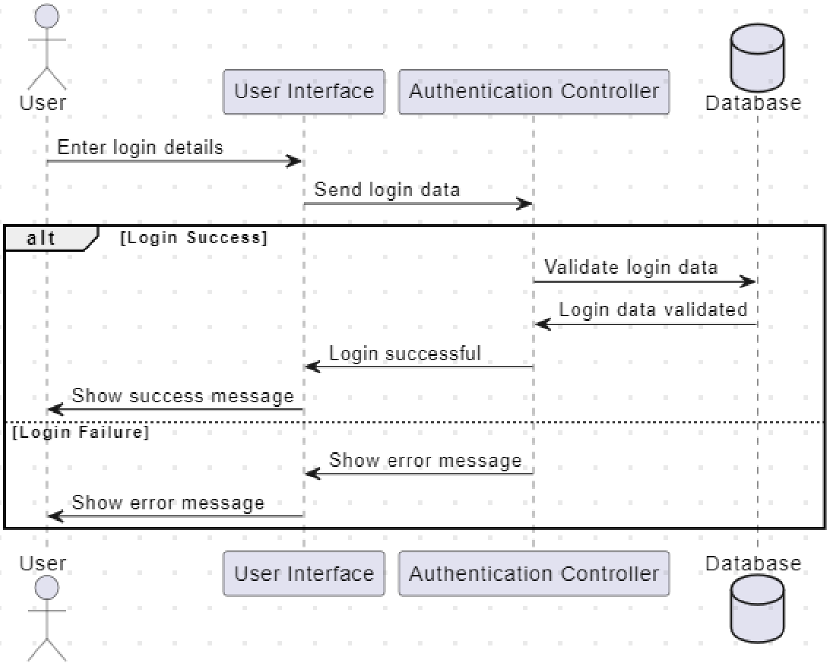


FIGURE 3.7 : Sequence Diagram for "Authentication" Use Case

3.5.3 Detailed sequence diagram for "Update profile" use case

Figure 3.4 illustrates the detailed sequence diagram for the "Update profile" use case.

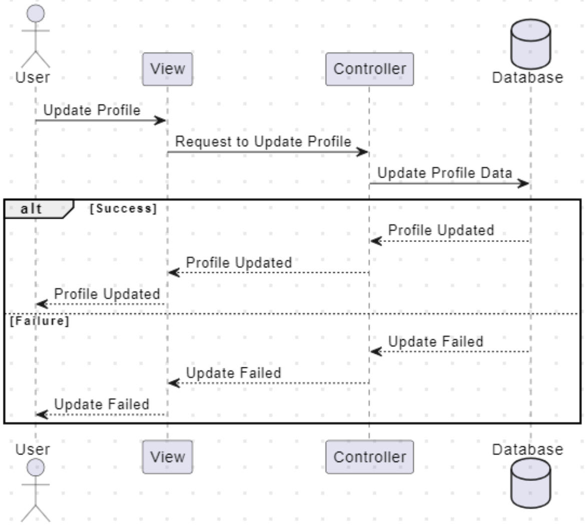


FIGURE 3.8 : Sequence Diagram for "Update profile" Use Case

3.6 The scoring process

The scoring logic employed in our system plays a pivotal role in evaluating the suitability of a candidate's resume concerning a given job description. The primary objective of this scoring mechanism is to quantitatively measure the similarity between the contents of a candidate's resume and the expectations outlined in the job description. This quantification is presented as a percentage, providing a clear and intuitive indicator of the level of alignment.

Here are the scoring process steps :

— **Document Retrieval :** The process begins with the retrieval of two documents : a job description and a candidate's resume.

— **Text Processing :** The content from both documents is processed using the docx2txt library. This library converts the content of Word documents into plain text for analysis.

— **Count Vectorization :** The text content from the job description and resume is converted into numerical representations using the CountVectorizer from the scikit-learn library. This process involves counting the frequency of each term (word) in the text.

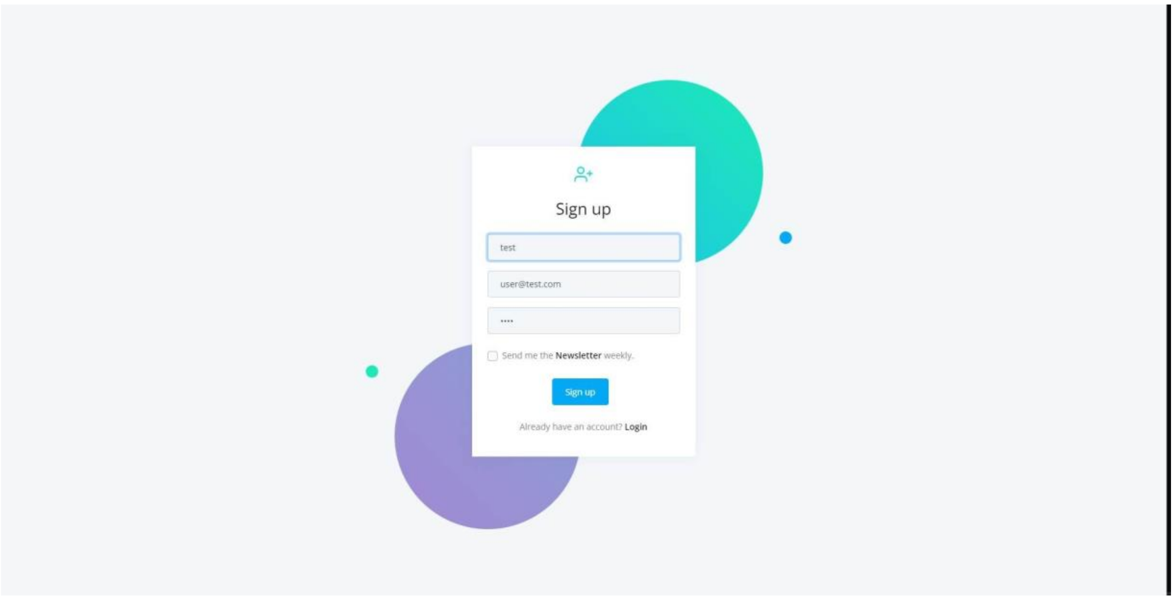
— **Cosine Similarity Calculation :** The cosine similarity is calculated between the term frequency vectors of the job description and the resume. Cosine similarity is a measure of similarity between two non-zero vectors and ranges from 0 (no similarity) to 1 (perfect similarity). In this case, it is represented as a percentage.

— **Result Presentation :** The result of the cosine similarity calculation is presented as a percentage, indicating how closely the content of the resume matches the content of the job description.

— **API Endpoint :** This process is implemented as an API endpoint that responds to HTTP GET requests. When a GET request is made to this endpoint, it returns the calculated cosine similarity as a response.

3.7 Implementation

In this sprint, we move on to the most critical phase of our development, which is implementation. We now present a significant part of the application. To this end, we have completed the development phase and are proud to present screenshots illustrating this part of the application. Figure 3.6 and 3.7 represents the applicant registration interfaces in our application.



Sign up

test

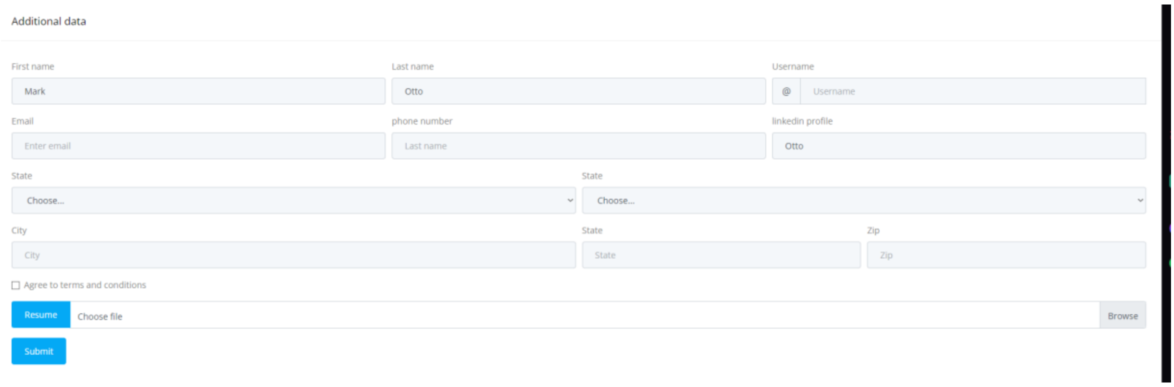
user@test.com

••••

□ Send me the **Newsletter** weekly.

Already have an account? **Login**

FIGURE 3.9 : Registration Interface_1



Additional data

First name: Mark

Last name: Otto

Username: @ Username

Email: Enter email

phone number: Last name

linkedin profile: Otto

State: Choose...

State: Choose...

City: City

State: State

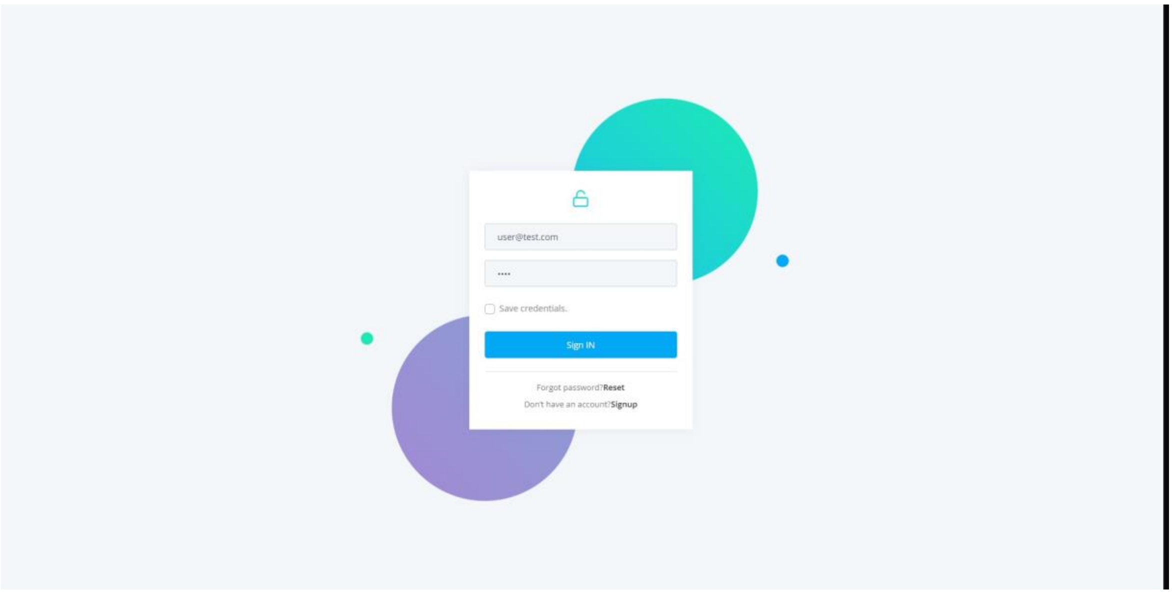
Zip: Zip

□ Agree to terms and conditions

Resume: Choose file

FIGURE 3.10 : Registration Interface_2

Figure 3.8 represents the authentication interface in our application.



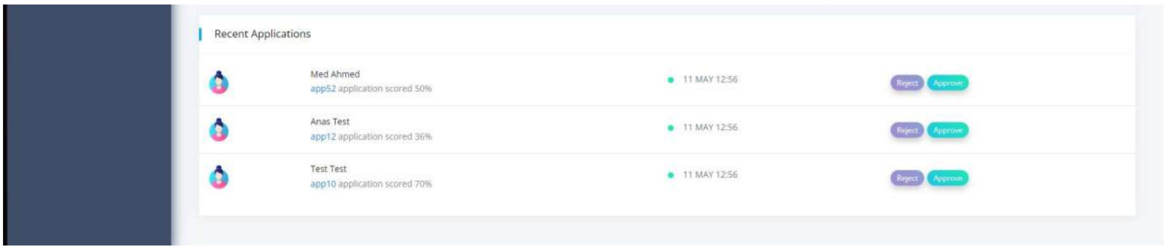
user@test.com

····

□ Save credentials.

Figure 3.11 : Authentication Interface

Figure 3.9 represents the interface designed for the HR manager to visualize applicants scores and approve or disapprove them.



Recent Applications

Med Ahmed
app52 application scored 50%
11 MAY 12:56

Anas Test
app12 application scored 36%
11 MAY 12:56

Test Test
app10 application scored 70%
11 MAY 12:56

Figure 3.12 : Profile Modification Interface

Conclusion

In this first chapter of sprint no. 1, we first organized our Sprint Backlog to determine the user stories to be completed for this sprint. Then, we created the corresponding use case, class, and sequence diagrams for this sprint. We also described the scoring logic and process. Finally, we included screenshots of the interfaces developed during this sprint for the implementation part. In the next chapter, we will discuss sprint no. 2, which will focus on managing job opportunities.

CHAPTER 4

SPRINT 2 : Job Management

Plannig

Introduction

This chapter is dedicated to describing the analysis, design, and implementation of the second sprint, focusing on job listings and applications.

4.1 Sprint 2 Backlog

The various tasks related to the realization of the second sprint are described in the Sprint 2 backlog (see Table 4.1).

TablE 4.1 : Sprint 2 Backlog

ID	User Story	Sub-Tasks
US 6	As an SME HR Manager, I must be able to create and post detailed job listings, including job titles, descriptions, responsibilities, qualifications, location, and application deadlines.	**1.** Create detailed use case, class, and sequence diagrams for the "Add job opportunity" case. **2.** Design the interface (Front-end). **3.** Implement the code for the "add opportunity" within the application (Back-end). **4.** Test the registration process.
US 7	As an SME HR Manager, I should be able to preview job listings before publishing to ensure accuracy.	**1.** Create detailed use case, sequence, and class diagrams for the "Preview job listings" case. **2.** Design the interface for "Preview job listings" (Front-end). **3.** Implement code for previewing job listings (Back-end). **4.** Test the "Preview job listings" case.
US 8	As an SME HR Manager, I should be able to sort candidate applications by date, status, and relevance.	**1.** Create detailed use case, sequence, and class diagrams for the "sort candidate applications" case. **2.** Design the interface for "sort candidate applications " (Front-end). **3.** Implement code for sorting candidate applications (Back-end). **4.** Test the " sort candidate applications" case.

US 10	As an applicant, I should be able to search and view job listings.	**1.** Create detailed use case, sequence, and class diagrams for the "search and view job listings" case. **2.** Design the interface for "search and view job listings" (Front-end). **3.** Implement code for searching and viewing job listings (Back-end). **4.** Test the " search and view job listings" case.
US 11	As an applicant, I should be able to receive notifications regarding application status updates.	**1.** Create detailed use case, sequence, and class diagrams for the "Receive notifications" case. **2.** Design the interface for "Receive notifications " (Front-end). **3.** Implement code for receiving notifications (Back-end). **4.** Test the "Receive notifications" case.

4.2 Detailed Use Case Diagram of Sprint 2

Figure 4.1 illustrates the detailed use case diagram for the second sprint.

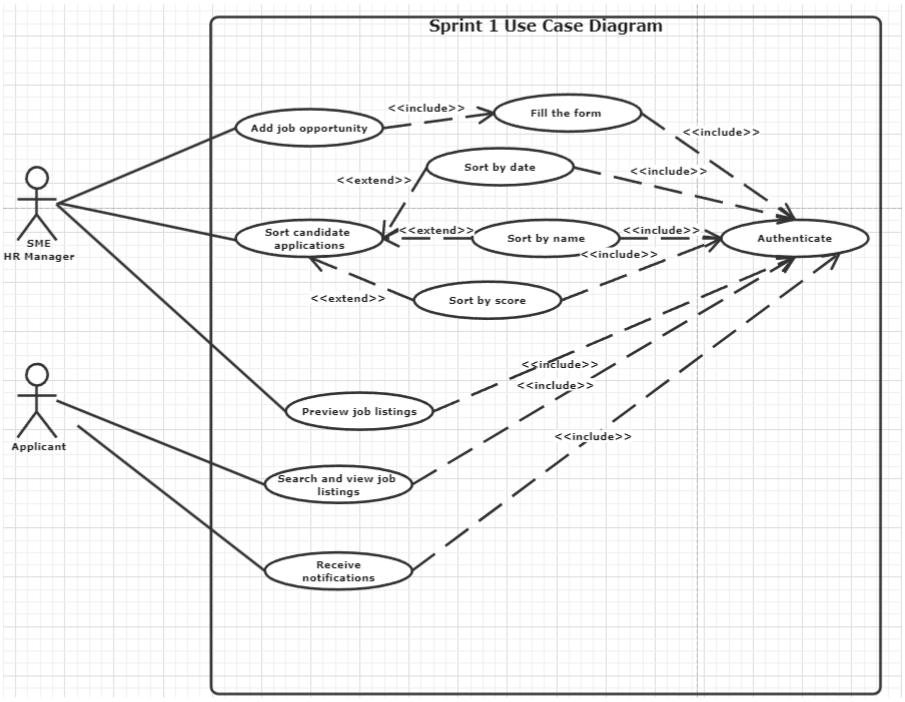


FIGURE 4.1 : Second Sprint Use Case Diagram

4.3 Class Diagram for Sprint 2

Figure 4.4 illustrates the class diagram for Sprint 2.

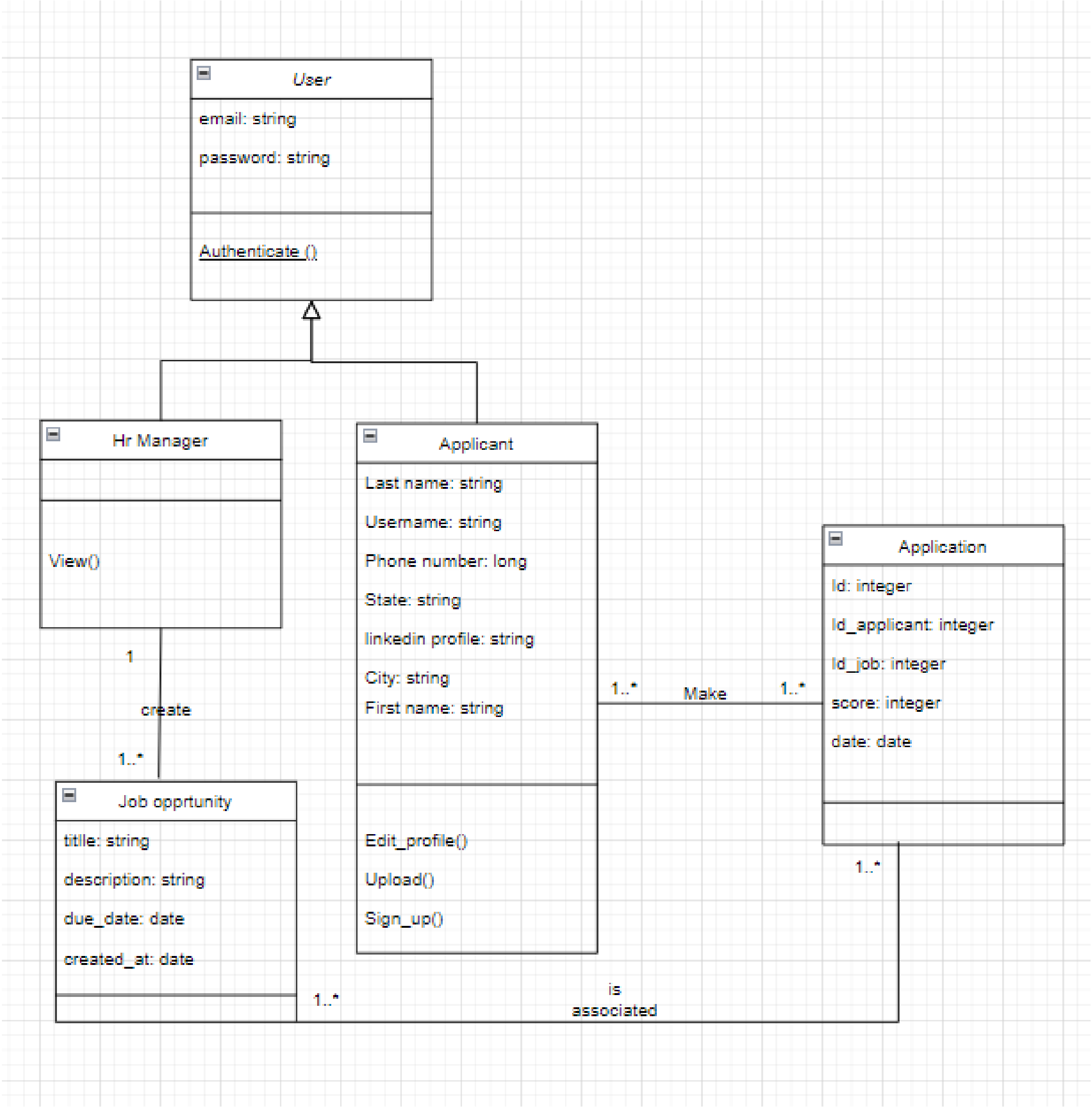


FIGURE 4.2 : Class Diagram

4.4 Sequence Diagrams for Sprint 2

4.4.1 Sequence Diagram : "Add job opportunity"

- **Textual Description of Use Case**

Figure 4.5 illustrates the sequence diagram for the "Add job opportunity" case.

Table 4.2 : Textual Description of Use Case : Add job opportunity

Use Case	Add job opportunity
Actor	Applicant
Precondition	The applicant accessed to the add job interface.
Post-condition	The HR manager is authenticated.
Description of the Main Scenario	**1.** The Applicant fill the add form with the job opportunity details . **3.** The applicant click on the submit button.
Exception	**Exception 1 :** If some fields are left empty, the system displays an error message next to each empty field.

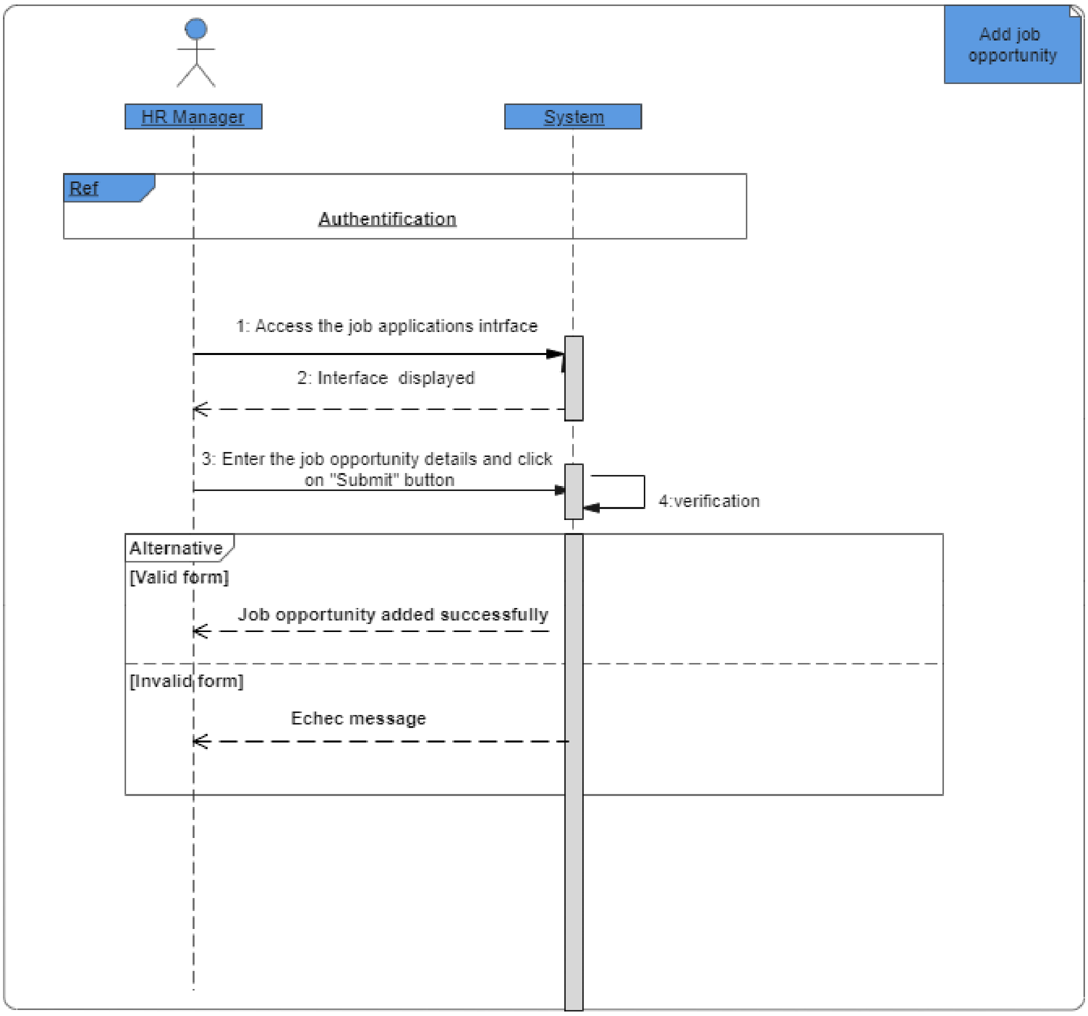


Figure 4.3 : Sequence Diagram : "Add job opportunity"

4.4.2 Sequence Diagram : "Search job opportunities"

Figure 4.6 illustrates the sequence diagram for the "Search job opportunities" case.

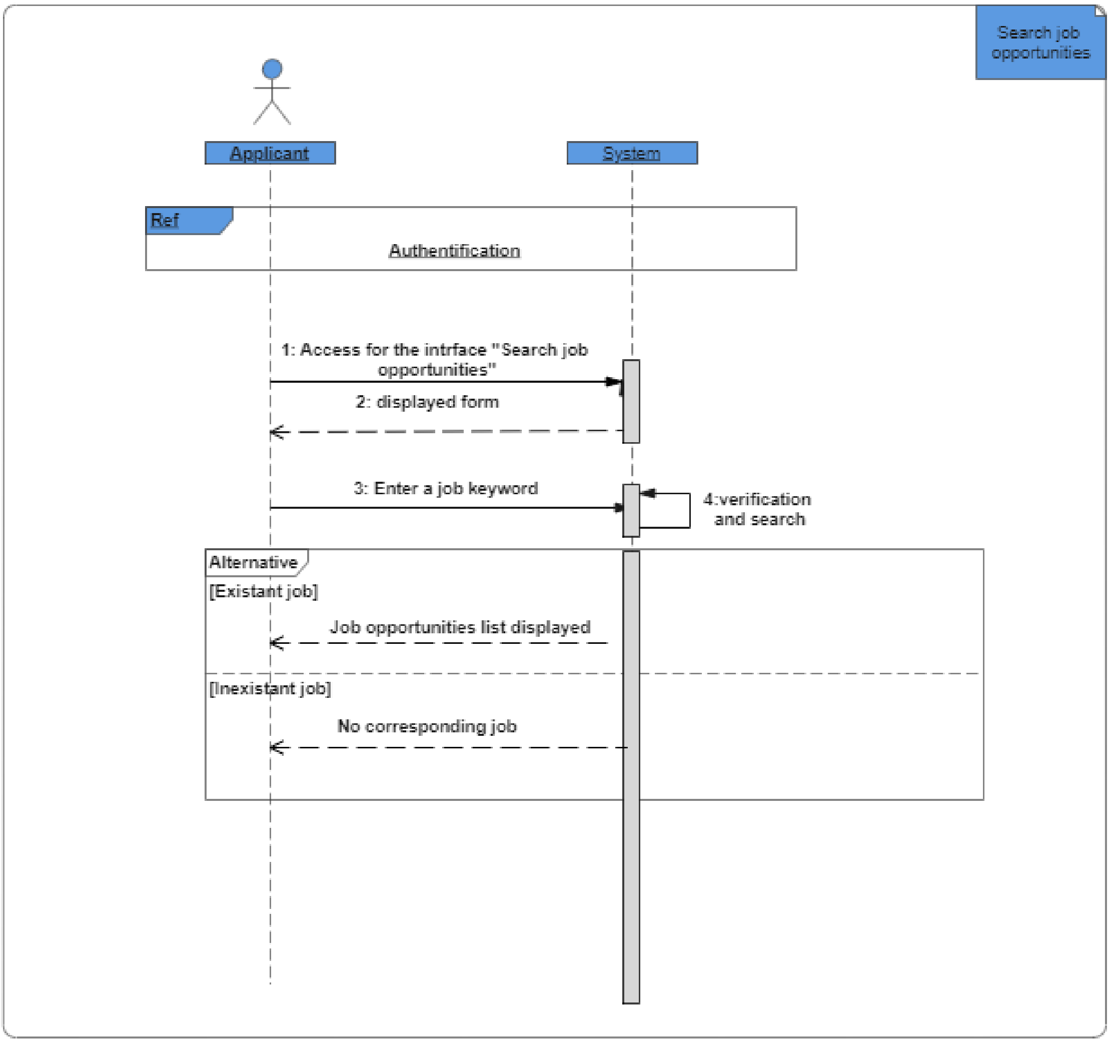


FIGURE 4.4 : Sequence Diagram : "Search job opportunities"

4.4.3 Sequence Diagram : "Sort job applications"

Figure 4.7 illustrates the sequence diagram for the "Sort job applications" case.

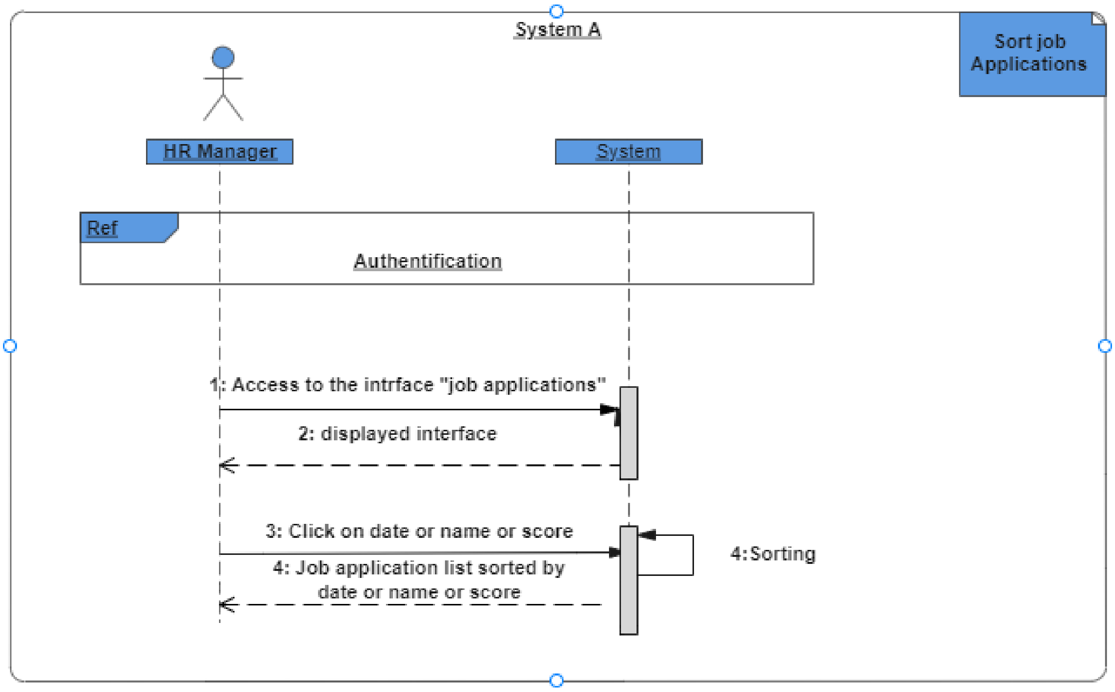

System A
Sort job Applications
HR Manager
System
Ref
Authentification
1: Access to the intrface "job applications"
2: displayed interface
3: Click on date or name or score
4: Job application list sorted by date or name or score
4:Sorting

Figure 4.5 : Sequence Diagram : "Sort job applications"

4.5 Detailed sequence diagrams

Detailed sequence diagram for "Apply to job" use case

Figure 3.4 illustrates the detailed sequence diagram for the "Apply to job" use case.

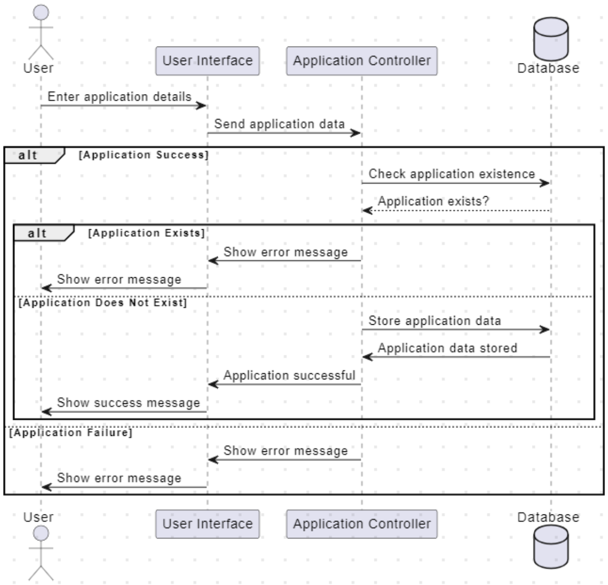
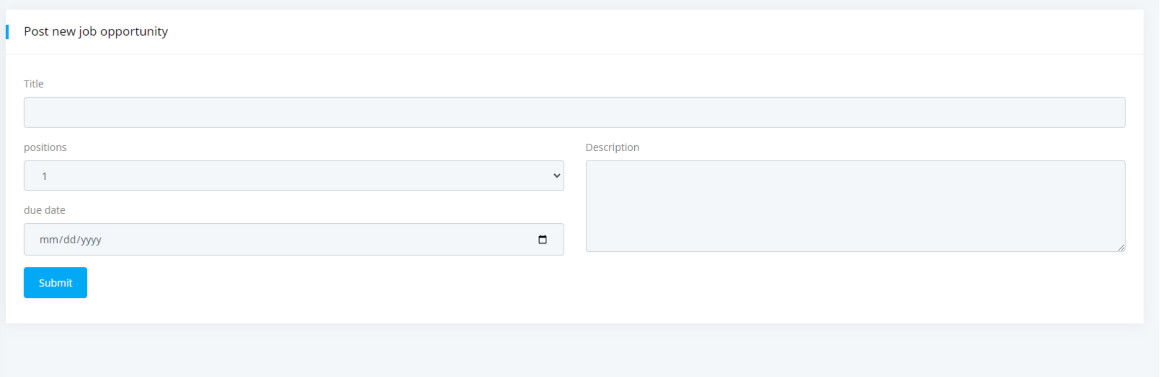


Figure 4.6 : Sequence Diagram for "Apply to job" Use Case

4.6 Implementation

Figure 4.7 represents the HR manager interface for adding job opportunities.



Post new job opportunity

Title

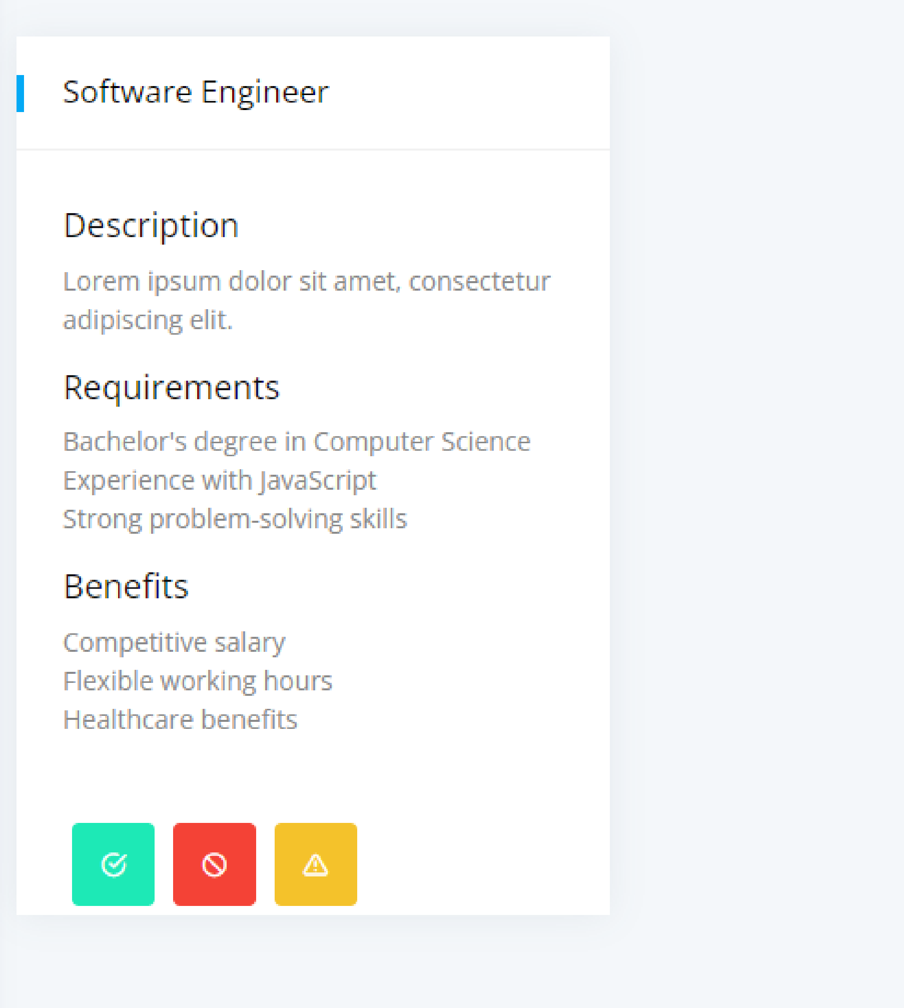
positions: 1

Description

due date: mm/dd/yyyy

FIGURE 4.7 : HR manager interface for adding job opportunities

Figure 4.8 represents the HR manager interface for previewing job listings before publishing them.



Software Engineer

Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Requirements

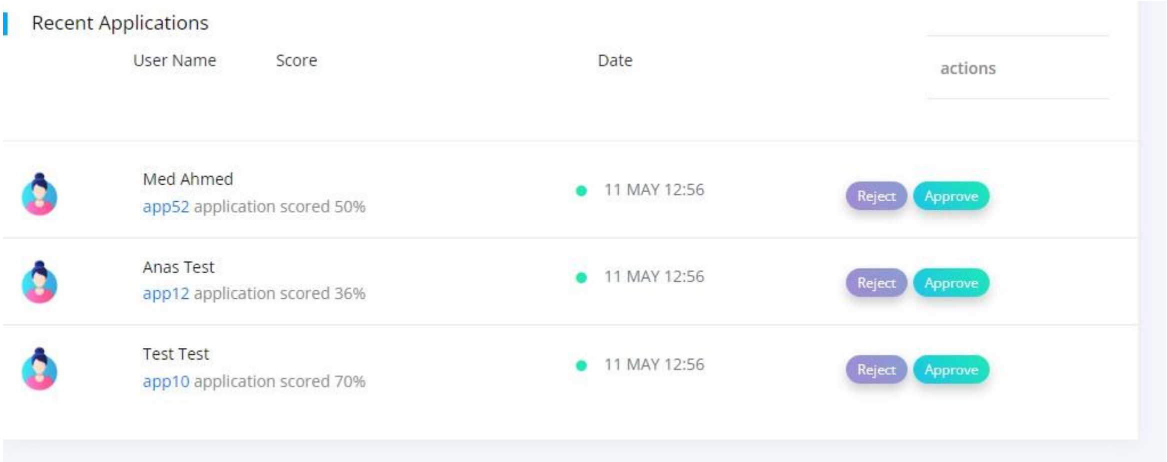
Bachelor's degree in Computer Science
Experience with JavaScript
Strong problem-solving skills

Benefits

Competitive salary
Flexible working hours
Healthcare benefits

FIGURE 4.8 : HR manager interface for previewing job listings

Figure 4.9 represents the HR manager interface for sorting job application by date .

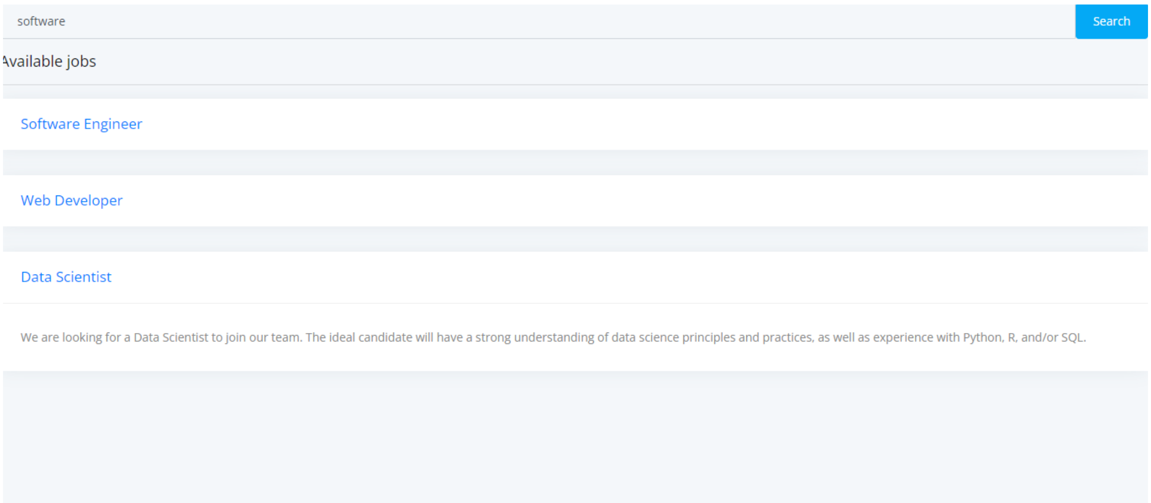


Recent Applications

User Name	Score	Date	actions
Med Ahmed app52 application scored 50%		● 11 MAY 12:56	Reject Approve
Anas Test app12 application scored 36%		● 11 MAY 12:56	Reject Approve
Test Test app10 application scored 70%		● 11 MAY 12:56	Reject Approve

Figure 4.9 : User Interface for Adding a Participant

Figure 4.10 represents the applicant interface for searching job opportunity by keywords.



software

Available jobs

Software Engineer

Web Developer

Data Scientist

We are looking for a Data Scientist to join our team. The ideal candidate will have a strong understanding of data science principles and practices, as well as experience with Python, R, and/or SQL.

Figure 4.10 : User Interface for Viewing the List of Events and Participants

Conclusion

In this chapter of Sprint 2, we first organized our Sprint Backlog to define the user stories to be implemented. Then, we created detailed use case, class, and sequence diagrams for the respective cases. Lastly, we showcased screenshots of the interfaces developed during this sprint in the implementation section. In the following chapter, we will commence Sprint 3, dedicated to profile management.

CHAPTER 5

SPRINT 3 : PROFILE Management

Plannig

Introduction

This chapter is dedicated to describing the analysis, design, and implementation of the third sprint. The primary objective of this sprint is to enable HR managers to manage profiles.

5.1 Sprint 3 Backlog

The various tasks related to the implementation of the third sprint are described in the sprint 3 backlog (see table 5.1).

TablE 5.1 : Sprint 3 Backlog

ID	User Story	Sub-Tasks
US 12	As an SME HR Manager, I must be able to manage user accounts by modifing or blocking them.	**1.** Create the "manage user accounts" interface (Front-end). **2.** Code to manage user accounts (Back-end). **3.** Test the "manage user accounts " case.
US 13	As an SME HR Manager, I must be able to access candidate profiles with contact information, resume uploads.	**1.** Create detailed use case, class, and sequence diagrams for the "Access Profiles" case. **2.** Create the "Access profiles" interface (Front-end). **3.** Implement code for accessing profiles (Back-end). **4.** Test the "Access profiles" case.
US 14	As an HR manager, I should be able to access and view statistical reports on my dashboard.	**1.** Create detailed use case, class, and sequence diagrams for the "View dashboard" case. **2.** Create the "View dashboard" interface (Front-end). **3.** Implement code for viewing dashboard (Back-end). **4.** Test the "View dashboard" case.

5.2 Detailed Use Case Diagram for Sprint No. 3

Figure 5.1 illustrates the detailed use case diagram for the third sprint.

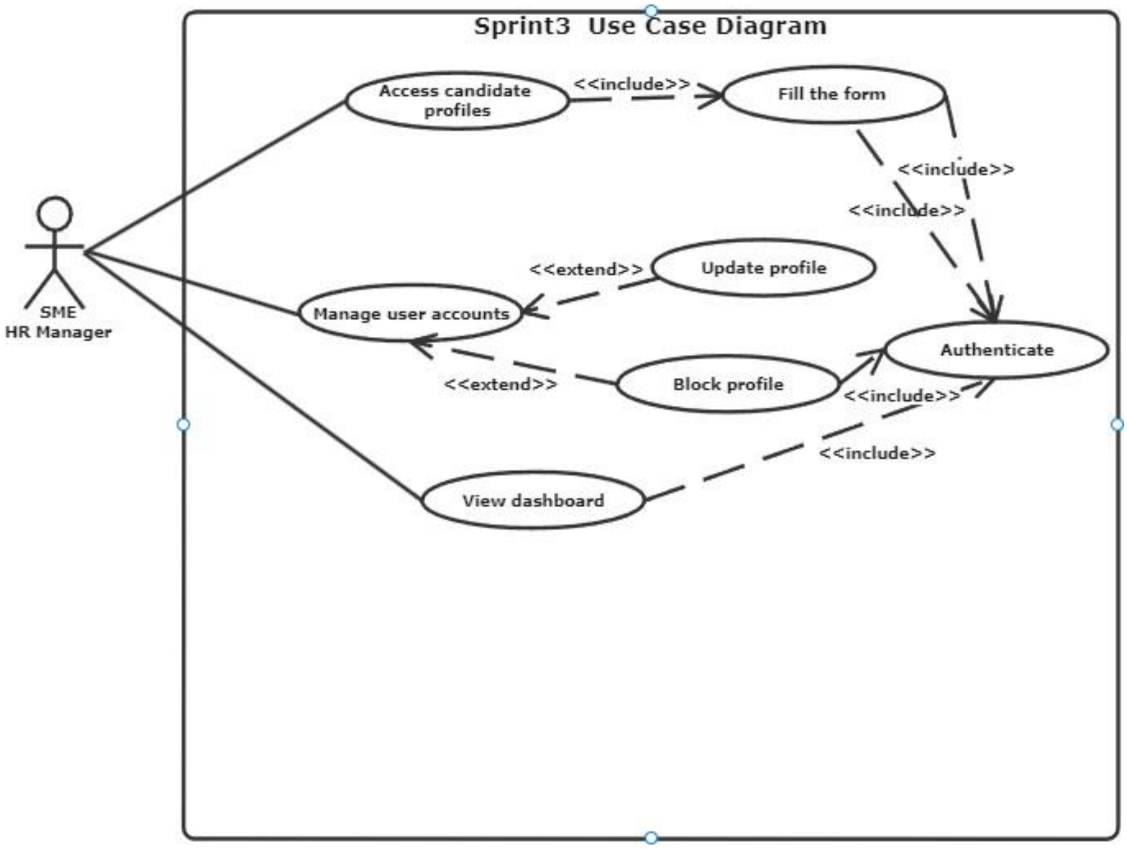


FIGURE 5.1 : Sprint 3 Detailed Use Case Diagram

5.3 Class Diagram for Sprint 3

Figure 5.2 illustrates the class diagram for sprint 3.

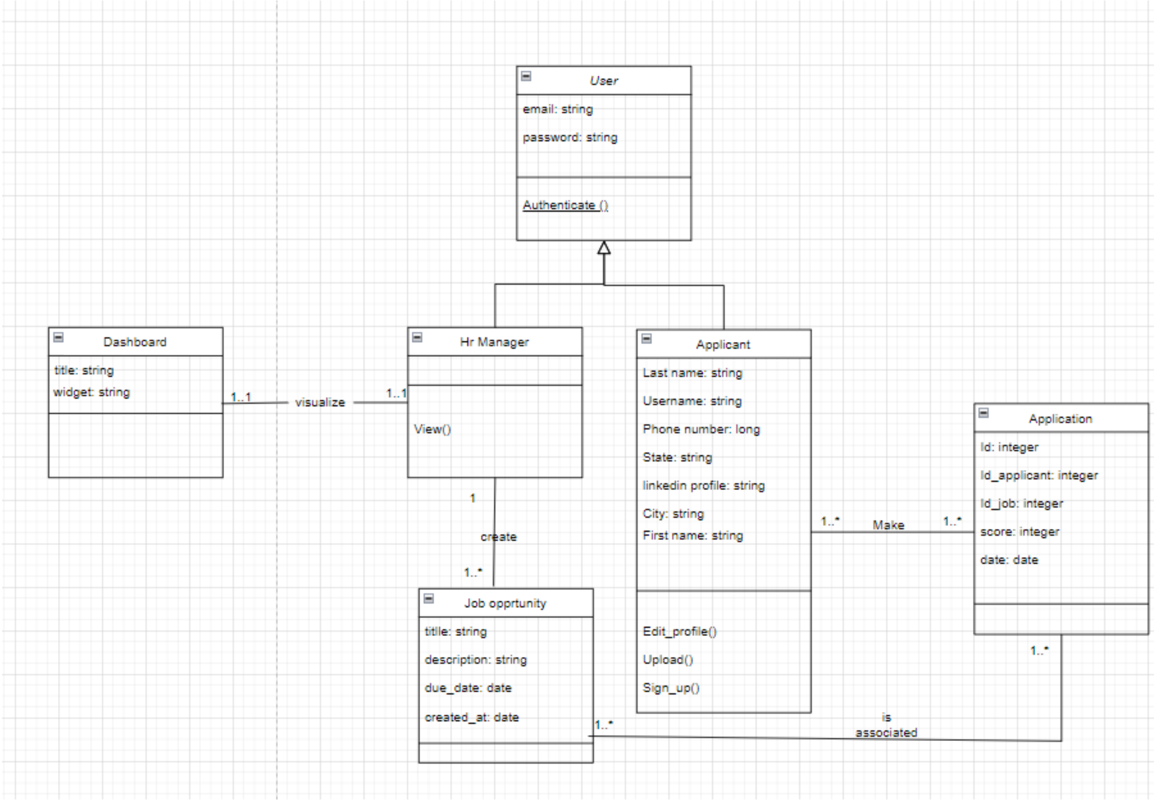


FIGURE 5.2 : Class Diagram

5.4 Sequence Diagrams for Sprint 3

5.4.1 Sequence Diagram for "Modify applicant profile"

Figure 5.4 illustrates the sequence diagram for the "Modify applicant profile" case.

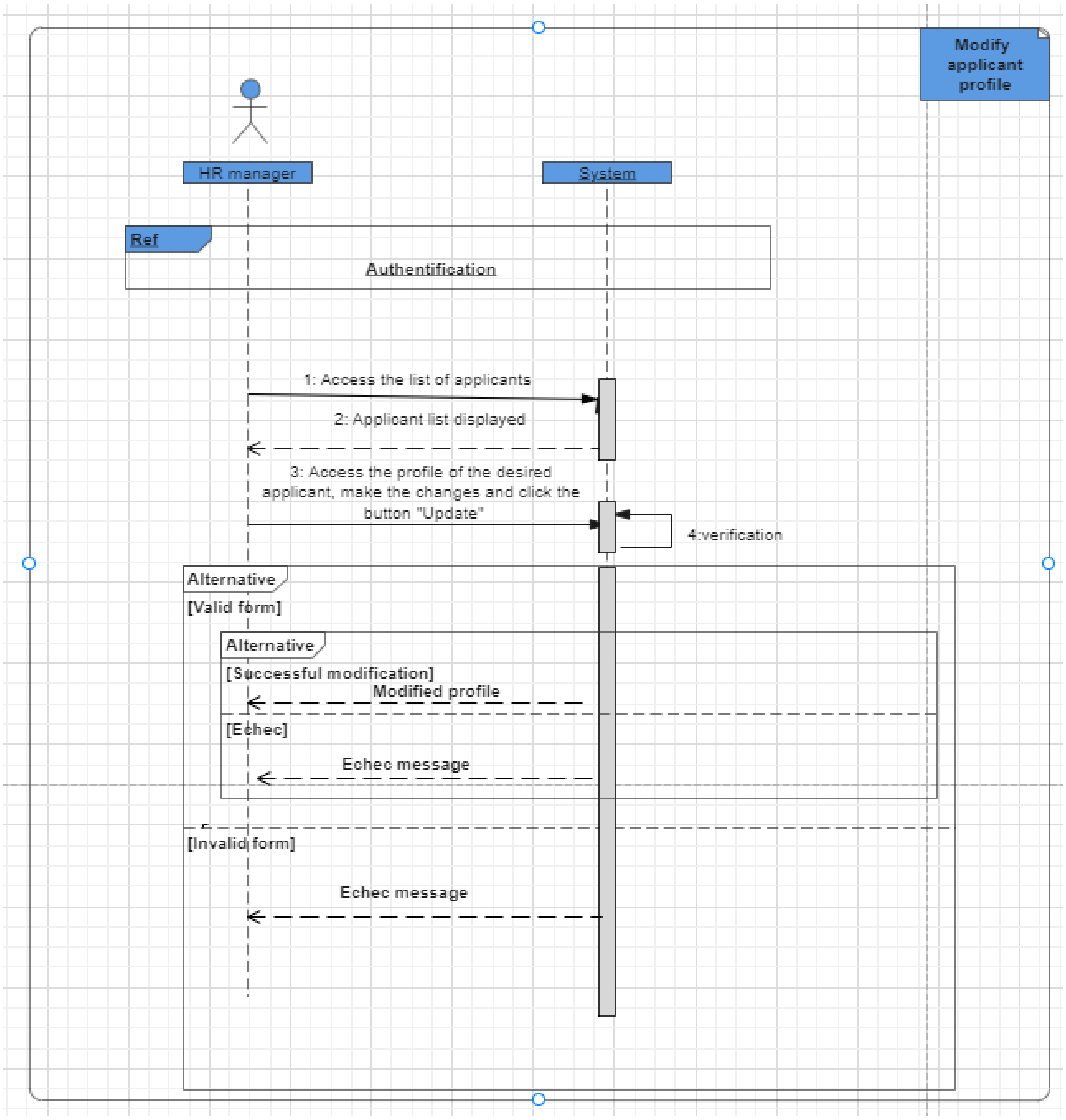

Modify applicant profile
HR manager
System
Ref
Authentification
1: Access the list of applicants
2: Applicant list displayed
3: Access the profile of the desired applicant, make the changes and click the button "Update"
4:verification
Alternative
[Valid form]
Alternative
[Successful modification]
Modified profile
[Echec]
Echec message
[Invalid form]
Echec message

Figure 5.3 : Sequence Diagram for "Modify applicant profile"

5.4.2 Sequence Diagram for "Block an applicant"

Figure 5.3 illustrates the sequence diagram for the "Block an applicant" case.

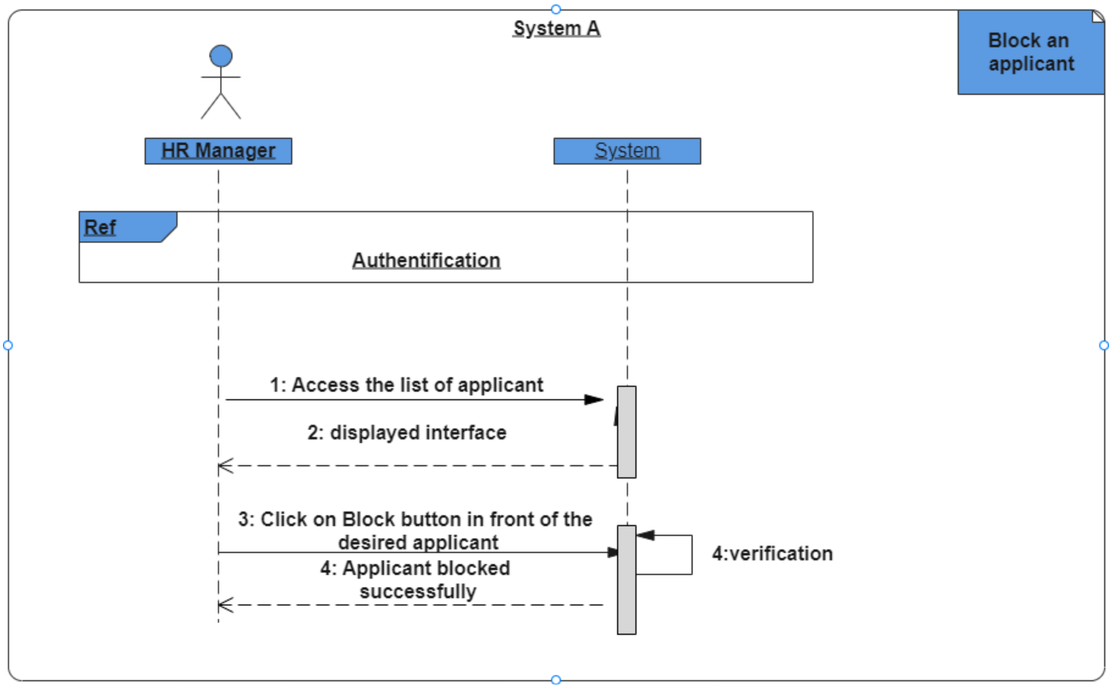

System A
Block an applicant
HR Manager
System
Ref
Authentification
1: Access the list of applicant
2: displayed interface
3: Click on Block button in front of the desired applicant
4:verification
4: Applicant blocked successfully

Figure 5.4 : Sequence Diagram for "Block an applicant"

5.5 Detailed sequence diagrams

5.5.1 Detailed sequence diagram for "Blocking an applicant" use case

Figure 3.4 illustrates the detailed sequence diagram for the "Blocking an applicant" use case.

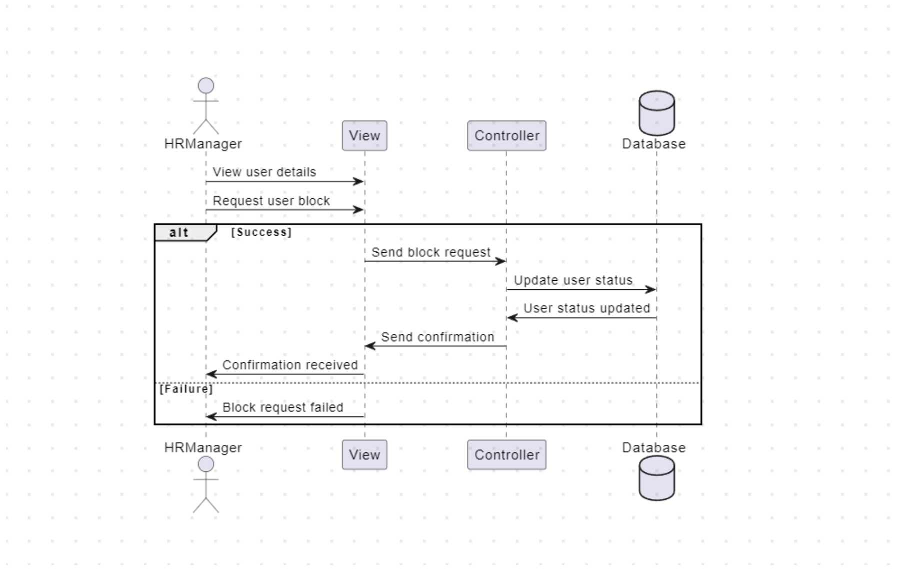


Figure 5.5 : Sequence Diagram for " Blocking an applicant" Use Case

5.5.2 Detailed sequence diagram for "View dashboard" use case

Figure 3.4 illustrates the detailed sequence diagram for the "View dashboard" use case.

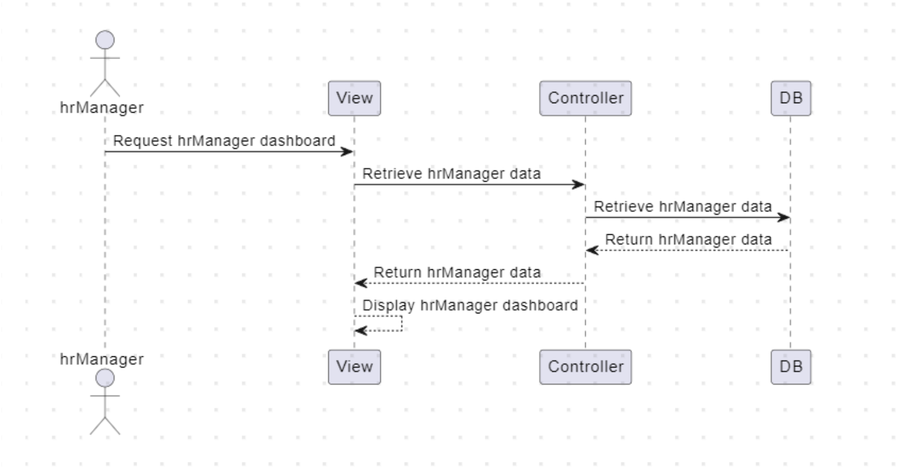
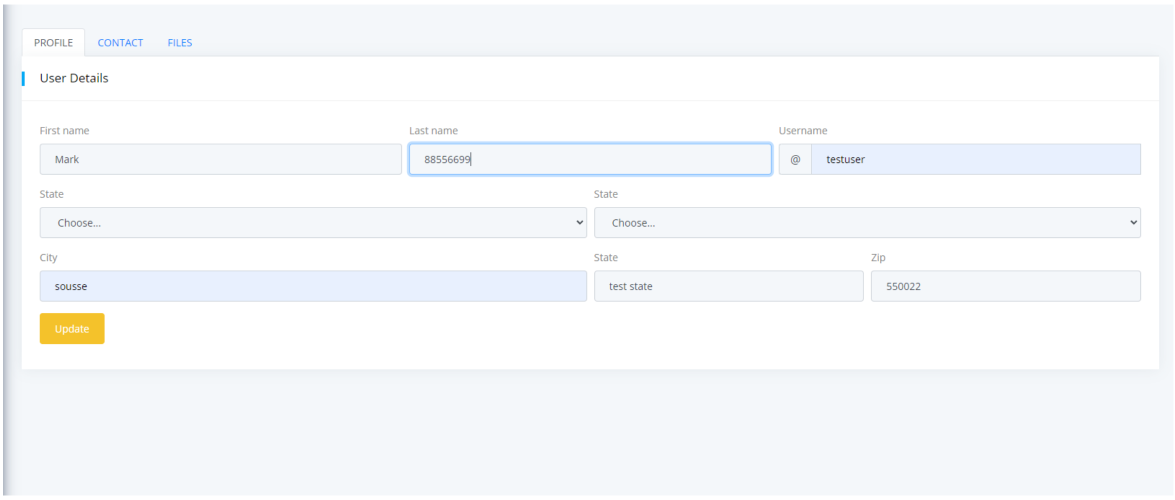


FIGURE 5.6 : Sequence Diagram for "View dashboard" Use Case

5.6 Implementation

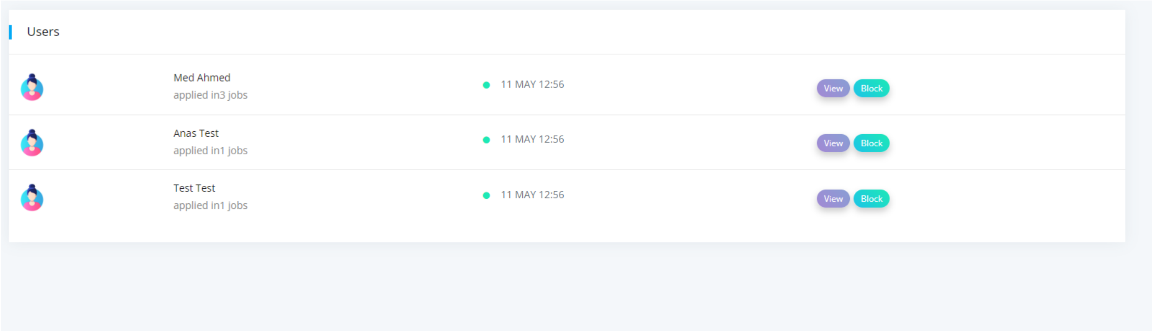
Figure 5.5 represents the HR interface for updating an applicant profile.



First name: Mark
Last name: 88556699
Username: @ testuser
State: Choose...
State: Choose...
City: sousse
State: test state
Zip: 550022

FIGURE 5.7 : Update Profile Interface

Figure 5.6 represents the HR interface for viewing or blocking an applicant.



Users

Med Ahmed applied in3 jobs	● 11 MAY 12:56	View	Block
Anas Test applied in1 jobs	● 11 MAY 12:56	View	Block
Test Test applied in1 jobs	● 11 MAY 12:56	View	Block

Figure 5.8 : Block Profile Interface

Figure 5.7 represents the HR interface for visualizing the dashboard.

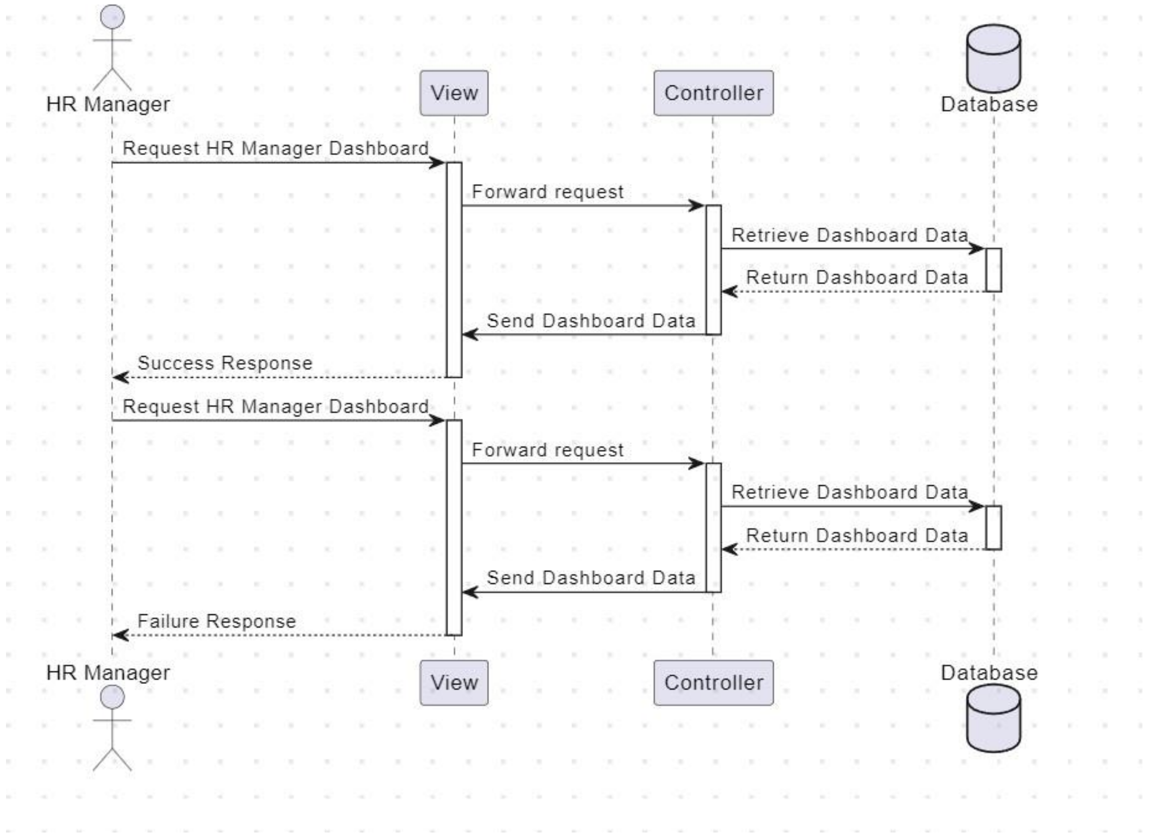


Figure 5.9 : Dashboard Interface

Conclusion

In this chapter of Sprint No. 3, we first organized our Sprint Backlog to define the user stories to be implemented in this sprint. Secondly, we created detailed use case, class, and sequence diagrams corresponding to this sprint. Furthermore, we provided screenshots of the interfaces created during this sprint in the implementation section.

General conclusion and perspetives

At the end of our final year project, we have successfully designed, developed, and implemented a Web-Based Human Resources and Recruitment Management Application for SMEs which represent a significant step toward addressing the unique challenges faced by small and medium-sized enterprises in managing their human resources and recruitment processes. This project has strived to provide an efficient, customizable, and cost-effective solution that empowers SMEs to streamline their HR operations, attract top talent, and remain competitive in a dynamic business landscape.

We adopted the Scrum methodology throughout our work, building our module incrementally. We began with a deep understanding of the overall project context and its justification. Next, we conducted a preliminary study to identify the techniques and tools to be used, as well as the methodology to follow. We then proceeded with a thorough analysis of the requirements, which allowed us to derive the Product Backlog. We studied the different Sprints in our Backlog, and for each sprint, we detailed the main tasks performed during the development phase.

Throughout the realization of this project, we faced technical challenges that contributed to our acquisition of new knowledge. This project provided us with the opportunity to implement innovative and scalable techniques while solidifying the knowledge we gained at out school.

Looking forward, there are several perspectives and potential enhancements to consider for the future of this project such as advanced AI techniques by explore the integration of more advanced AI and machine learning techniques, such as natural language processing (NLP) and deep learning, for more sophisticated candidate assessment and matching.

Webography

[1] Scrum,https://www.piloter.org/projet/methode/scrum, accessed on 17/09/2023.

[2] Concept,https://agiliste.fr/introduction-methodes-agiles/, accessed on 17/09/2023.

[3] UML, https://www.lucidchart.com/pages/fr/langage-uml, accessed on 17/09/2023.

[4] HTML, https://fr.wikipedia.org/wiki/Hypertext_Markup_Language, accessed on 18/04/2023.

[5] CSS, ,https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade, accessed on 18/04/2023.

[6] Python, https://fr.wikipedia.org/wiki/python, accessed on 08/10/2023.

[7] Visual , https://fr.wikipedia.org/wiki/Visual_Studio_Code, accessed on 01/10/2023.

[8] Diagrams.net , https://en.wikipedia.org/wiki/Diagrams.net, accessed on 01/10/2023.

[9] Postman , https://fr.wikipedia.org/wiki/Postman_(logiciel), accessed on 01/10/2023.

[10] React , https://fr.wikipedia.org/wiki/React, accessed on 01/10/2023.

[11] Django , https://fr.wikipedia.org/wiki/Django, accessed on 01/10/2023.

[12] PostgreSQL , https://fr.wikipedia.org/wiki/PostgreSQL,r, accessed on 01/10/2023.

Résumé

Ce travail vise à obtenir une Licence en Commerce Électronique Appliqué à l'École Supérieure de Commerce Électronique à Manouba. L'objectif de ce travail était de concevoir et développer une application de gestion des ressources humaines et du recrutement basée sur le web pour les PME. Nous avons cherché à simplifier le processus de recrutement et à rester compétitifs dans un environnement commercial en constante évolution.

Mots clés : React Js, django, Scrum, NLP.

Abstract

This work aims to obtain a Bachelor's Degree in applied E-Commerce in the Higher School of E-commerce in Manouba. The objective of this work was to design and develop a Web-Based Human Resources and Recruitment Management Application for SMEs . We aimed to simplify the recuitement process and remain competitive in a dynamic business landscape.

Keywords : React Js, django, Scrum, NLP.