



PIDEV REPORT

SAE

Realized by:

Cherif Yasmine

Saadaoui Ilef

El Jery Fyras

Abed Jawher

Klila Karim

Hmoudi Mouwaffak

«Internship service management platform»

Academic Year 2023/2024

Table of Contents

Chapter 1 Study of the organism:	7
1. Introduction	7
2. Study of the existing	7
a. Critique of the existing	7
b. Proposed solution	9
3. Specification of needs	10
a. Identification of actors	10
b. Functional requirements	10
c. Non - functional requirements	11
4. Global Use Case Diagram	12
5. Global class diagram	14
6. Software environment	14
7. Mockups	15
8. Development Method	17
9. Global Architecture	17
a. Logic Architecture	17
b. Physical Architecture	18
10. Product Backlog	18
11. Conclusion	20
Chapter 2 Sprint 1:	21
1. Introduction	21

2. Use case diagram -----	21
3. Realization -----	25
a. Subject management -----	25
b. Postulation management -----	30
c. journal and evaluation -----	32
d. Complaint and response management-----	33
e. Users management -----	36
4. Conclusion -----	37
Chapter 3 Sprint 2: -----	38
1. Introduction -----	38
2. Use case diagram -----	38
3. Realization -----	41
a. Confirm a postulation -----	41
b. Manage files -----	42
c. Manage convention -----	42
d. Manage journal-----	44
e. User management -----	44
4. Conclusion -----	46
<u>Chapter 4 Sprint 3:</u> -----	47
1. Introduction -----	47
2. Realization -----	47
a. Regional Patterns: Statistics & Clustering Analysis -----	47
b. Classification - determining if a comment/answer is correct based on the training data Model.--	52
c. Classification Logistic Regression: in convention -----	53

d. Classification: Extracting Statistics -----	53
e. KNN: Domain prediction based on student skills -----	55
3. Conclusion -----	58

Table of Figures

Figure 1 –Simplicity Platform -----	8
Figure 2 –Esprit PlatformFigure -----	9
Figure 3 –Global Use Case Diagram -----	13
Figure 4 – Global class diagram -----	14
Figure 5 – student interface -----	16
Figure 6 – company interface -----	16
Figure 7 – Company interface -----	16
Figure 8 – Scrum methodology -----	17
Figure 9 – Logic Architecture -----	18
Figure 10 – Physical Architecture -----	18
Figure 11 – Product backlog -----	20
Figure 12 – Use case diagram of sprint 1 -----	21
Figure 13 – refinement internship Management -----	22
Figure 14 – refinement Postulation Management -----	23
Figure 15 – refinement Forum Management -----	24
Figure 16 – refinement Journal and evaluation management -----	24
Figure 17 – refinement Complaint and Response management -----	25
Figure 18 – Add a subject interface (admin) -----	26
Figure 19 – View subject details (admin) -----	27
Figure 20 – View subject Postulations (admin) -----	27
Figure 21 – View All Postulations (admin) -----	28
Figure 22 – View subjects (Student) -----	29
Figure 23 – Postulate for a subject (Student) -----	29
Figure 24 – View postulations (Student) -----	30
Figure 25 –Posts management -----	30
Figure 26 –Comments management -----	31
Figure 27 –Manage posts -----	31
Figure 28 –Journal section -----	32
Figure 29 –Journal section -----	32
Figure 30 –Manage journal tasks -----	33
Figure 31 –view complaints -----	33
Figure 32–add a response -----	34
Figure 33–edit a complaint -----	34
Figure 34 –statictics -----	35
Figure 35 –manage complains -----	35
Figure 36 – List of users -----	36
Figure 37 – Add user -----	37
Figure 38 – Update user -----	37
Figure 39 – Use case diagram of sprint 2 -----	38
Figure 40– refinement Postulation Management -----	39
Figure 41– refinement File Management -----	39
Figure 42– refinement convention Management -----	40

Figure 43– refinement journal Management -----	40
Figure 44– Confirm postulation -----	41
Figure 45– Confirmation mail -----	41
Figure 46– View uploaded files -----	42
Figure 47– upload files -----	42
Figure 48– upload convention -----	43
Figure 49– validate or archive convention -----	43
Figure 50– archive convention -----	44
Figure 51– Confirmation mail -----	44
Figure 52– Upload students list -----	44
Figure 53 – Update temporary password -----	45
Figure 54 – Reset password mail -----	45
Figure 55– Region dataset -----	47
Figure 56– Kmeans code -----	48
Figure 57– Kmeans output -----	49
Figure 58– stats code -----	50
Figure 59– stats output 1 -----	51
Figure 60– stats output 2 -----	52
Figure 61– logistic regression output -----	52
Figure 62– logistic regression dataset -----	52
Figure 63– Code 1 -----	53
Figure 64– Code 2 -----	53
Figure 65– test 1 -----	54
Figure 66– test 2 -----	54
Figure 67– code 3 -----	55
Figure 68– test 3 -----	55
Figure 69– Code loading and encoding skills dataset -----	56
Figure 70– Code training KNN model -----	56
Figure 71– Code predicting domains for a new student -----	57
Figure 72– Code main function to use in Flusk endpoint -----	57
Figure 73– Code Flask app and endpoint -----	57

Chapter 1: Study of the organism

1. Introduction

This report presents the creation of a comprehensive internship management platform, designed to streamline the internship process for students, companies, and academic institutions. By leveraging modern technology, the platform aims to enhance efficiency, communication, and collaboration throughout the internship lifecycle.

2. Study of the existing

The existing internship management system suffers from significant drawbacks, including manual application processes, time loss, complexity, and limited access to information. These issues underscore the necessity for a more streamlined and efficient solution.

a. Critique of the existing

This functional aspect involves studying and analyzing similar projects such as Symplicity platform on an international scale and Esprit platform on a national scale in Tunisia.

i. Simplicity platform

Simplicity SIM (Simplicity Internship Management) was designed to manage the operational process of approval, monitoring, and evaluation of internships. SIM ensures the management of the signing and approval process of agreements between employers and universities.



Figure 1 –Simplicity Platform

- **Strengths**

- Its focus on simplifying administrative processes related to internships.
- Tracking capabilities for applications, internship offer management, and communication with companies.

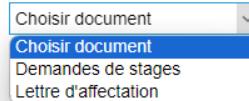
- **Weaknesses**

- A user interface that lacks intuitiveness and navigation difficulties.
- Unstable performance and limited features, such as integration with other tools.
- Customer support may be unsatisfactory due to slow response times and ineffective problem resolution.

ii. Esprit platform

In the Esprit platform, we can simply download an internship application or assignment letter through the form below. Only one assignment letter can be obtained.

Documents de stage



"Vous pouvez télécharger une demande de stage ou Lettre d'affectation à travers le formulaire ci-dessous.

Vous ne pouvez obtenir qu'une seule et unique Lettre d'affectation "

Pour toutes informations relatives aux Stages, veuillez contacter : contact.stage@esprit.tn"

Figure 2 –Esprit Platform

- **Weaknesses**

- Students manually submit internship applications.
- Time Loss (Back and Forth of Students)
- Complexity of Application Process
- Limited Access to Feedback
- Delays in receiving responses from companies/administration
- Lack of centralized information about available internships opportunities
- Difficulty retrieving any document in case of loss
- Administration Paperwork Overload
- Storage and Organization of Paper Documents

b. Proposed solution

In response to the demands and to broaden our reach, we have embarked on the development of a platform that responds and offers these solutions:

- Automated Application (easy submission)
- Use automated notifications for real-time status updates (Timesaving)

- Facilitates real-time communication between companies/ students /administration
- Create an approach for feedback
- Secures data
- Document archiving (in case of loss)
- Simplify internship search
- Make the application process easier
- Document archiving for easy retrieval
- Easy access to company and student information
- Responsive platform

3. Specification of needs

In this part, we will reveal the actors and the functionalities of our future system.

a. Identification of actors

The actor is the end user who requests the system to have the information he needs.

The actors interacting with our system are organized as follows:

- **Student:** Individual enrolled in an educational institution seeking practical experience through internships.
- **Company Agent:** Representative within a company responsible for internship-related matters.
- **Administration Agent:** Staff member managing administrative tasks related to internships within the educational institution.
- **Super Admin:** Privileged user with comprehensive access and control over the internship management platform.

b. Functional requirements

The university's administration recognized the need for a more streamlined and efficient system to manage internships for students. To address this, they decided to implement a comprehensive internship management platform with various functionalities.

Firstly, administrators needed robust user account management capabilities. This included the ability to create, update, and delete accounts, and provide secure login, logout, and password reset options for users.

Students were given the autonomy to customize their profiles by adding, modifying, or deleting information as needed. They also had access to search and view available internships directly from the platform.

When students found internship opportunities outside the platform or from the list of subjects provided, they could prepare their internship applications using the site's admin interface.

Upon receiving confirmation from the company, students could generate internship agreements with all necessary details. The platform facilitated communication by automatically sending an email to the company, creating an account if required, and informing them about the student's internship agreement.

Throughout their internship, students had daily access to an internship journal to record tasks performed, with detailed descriptions. These entries were subject to approval by the company agent.

At the end of the internship, students submitted a fully completed internship journal, validated by academic and professional supervisors. Additionally, students were required to submit an internship report, which was evaluated by the supervisors, and a grade was assigned based on the assessment.

The platform also provided a dedicated space for students to ask questions and communicate with each other.

Furthermore, the system allowed both students and company agents to submit complaints. Administrators were responsible for reviewing and addressing these complaints promptly, ensuring transparency and accountability. Users were notified of the status and actions taken by the administrators regarding their complaints.

c. Non - functional requirements

These are the constraints to be considered to guarantee better operation of the application to be delivered:

- Security:**

- Encryption of sensitive data

- **Performance:**

- Fast response time
- Ability to handle many simultaneous users

- **System availability 24/7**

- **Scalability:**

- Ability to grow and adapt as the number of users increases without affecting system performance.

- **User-Friendliness:**

- Simple and easy-to-use interface for students, companies, and administration

4. Global Use Case Diagram



Figure 3 –Global Use Case Diagram

5. Global class diagram

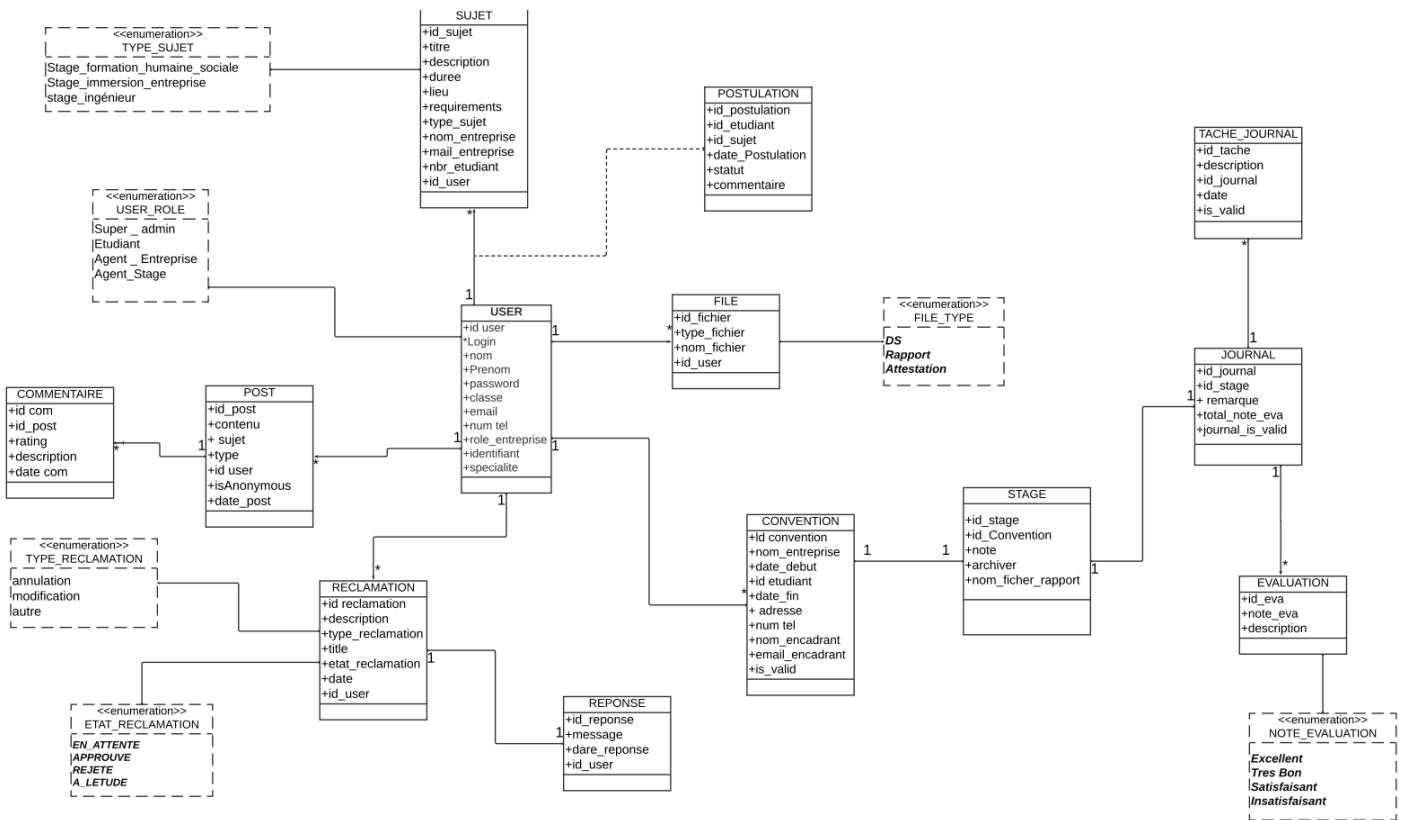


Figure 4 – Global class diagram

6. Software environment

Table 1 presents the software that we used throughout the realization of our project

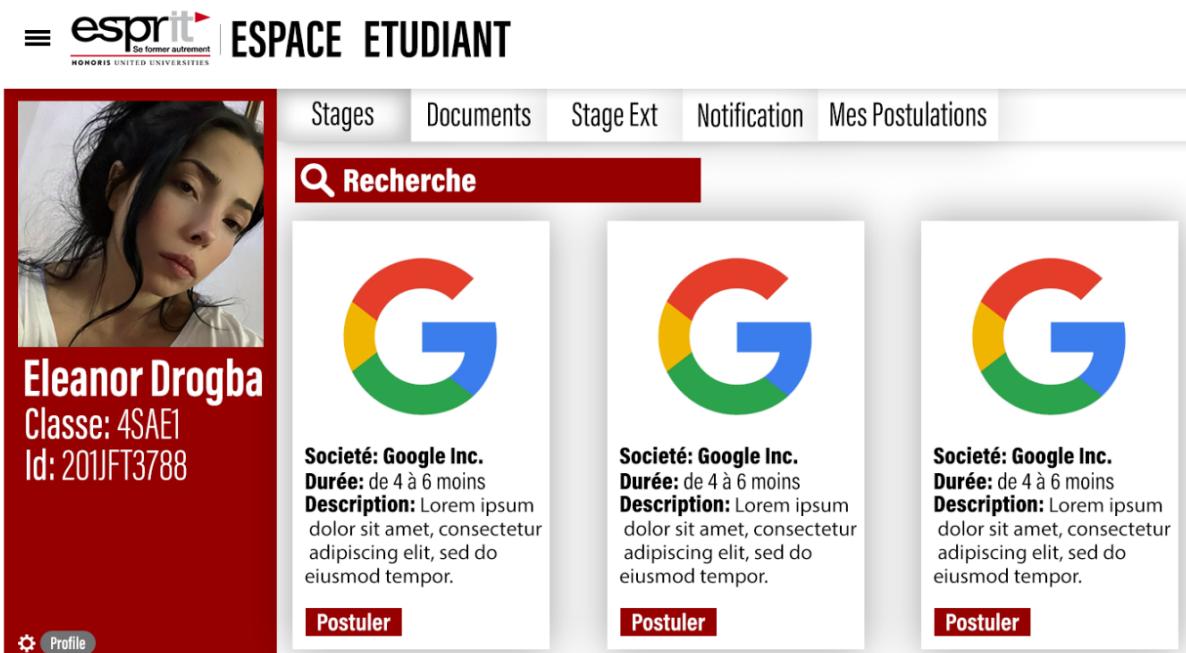
Table 1– Software Environment

Logo	Description
	Visual Studio Code is a lightweight and versatile code editor developed by Microsoft, popular for its extensive language support and various extensions for enhanced development workflows.

	<p>MongoDB Atlas is a fully managed cloud database service provided by MongoDB that allows users to deploy, manage, and scale MongoDB databases effortlessly, providing high availability and automated backups.</p>
	<p>IntelliJ IDEA is a powerful integrated development environment (IDE) for software developers, offering advanced features and tools to enhance productivity and streamline the coding process.</p>
	<p>Star UML is a modeling tool used for creating UML (Unified Modeling Language) diagrams to visualize and design software systems and processes.</p>

7. Mockups

Here is the student part interface:



The student interface features a header with the logo "esprit" and "ESPACE ETUDIANT". Below the header, there's a navigation bar with tabs: Stages, Documents, Stage Ext, Notification, and Mes Postulations. A search bar labeled "Recherche" is positioned above a grid of job posting cards. Each card displays a company logo (Google), the company name ("Société: Google Inc."), duration ("Durée: de 4 à 6 mois"), and a brief description. A "Postuler" button is at the bottom of each card. On the left, there's a sidebar with a profile picture of a woman and text: "Eleanor Drogba", "Classe: 4SAE1", and "Id: 201JFT3788". A "Profile" button is also present.

Figure 5 – student interface

Here is the admin part interface:

The screenshot shows the 'Espace Admin' interface. On the left, there is a profile picture of a woman and her details: **Eleanor Drogba**, Id: 201JFT3788. Below this is a 'Profile' button. At the top, there are four navigation tabs: 'Creation des comptes', 'Listes des stages', 'Listes des etudiants', and 'Listes des entreprises'. Under 'Creation des comptes', there are three sections: 'Cree compte classe' (with an 'Upload excel' button and a 'CRÉE' button), 'Cree compte entreprise' (with an 'Upload excel' button and a 'CRÉE' button), and 'Cree compte admin' (with an 'Upload excel' button and a 'CRÉE' button).

Figure 6 – company interface

Here is the company part interface:

The screenshot shows the 'Espace Entreprise' interface. On the left, there is a profile picture of a company and its details: **Google Inc.**, Id: 201JFT3788. Below this is a 'Profile' button. At the top, there are two navigation tabs: 'Ajout Stages' and 'Les Postulations'.

Figure 7 – Company interface

8. Development Method

In our project, where product dimensions aren't fixed from the start and needs may change, collaboration for adjusting requirements is essential. Given the dynamic nature and size of our project, prioritizing an agile method like Scrum is crucial for successful outcomes.

Scrum fosters a culture of learning from mistakes and continuous improvement. It emphasizes adaptation to variables and ongoing learning from errors, allowing teams to adjust to changing elements and requirements throughout development cycles.

To effectively utilize SCRUM, a sprint lifecycle is essential. Here's an overview:

1. The Product Owner specifies user stories in the Product Backlog.
2. The team defines the User Stories comprising the Sprint Backlog.
3. The team implements short-term sprint tasks over 2-4 weeks.
4. Daily, the team and Scrum Master review progress, set daily goals, and identify potential hurdles.
5. After completing work, a Sprint review is held to check the increment and adjust the Product Backlog if needed. A final Sprint retrospective discusses potential improvements.

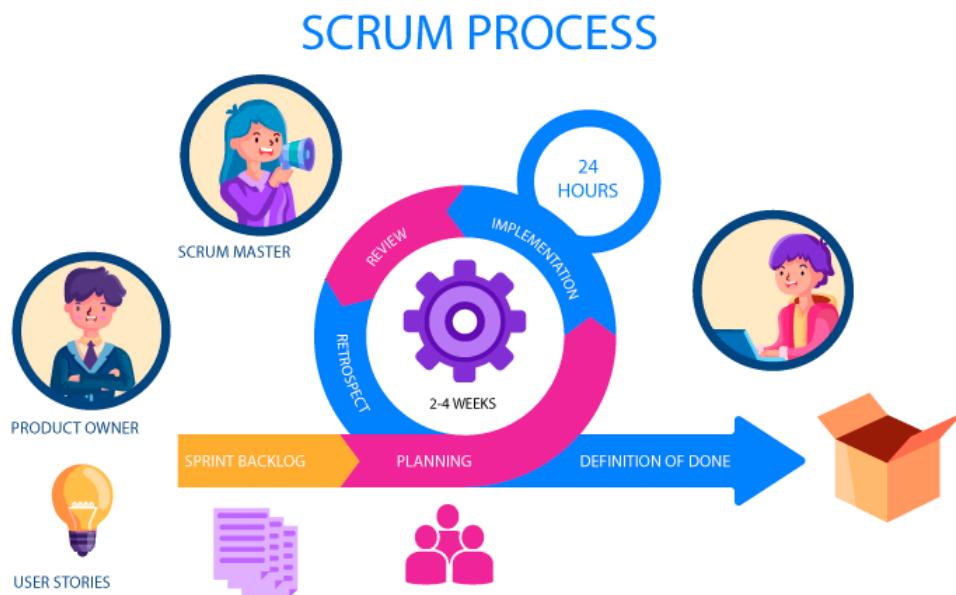


Figure 8 – Scrum methodology

9. Global Architecture

a. Logic Architecture:

Here is the logic architecture:

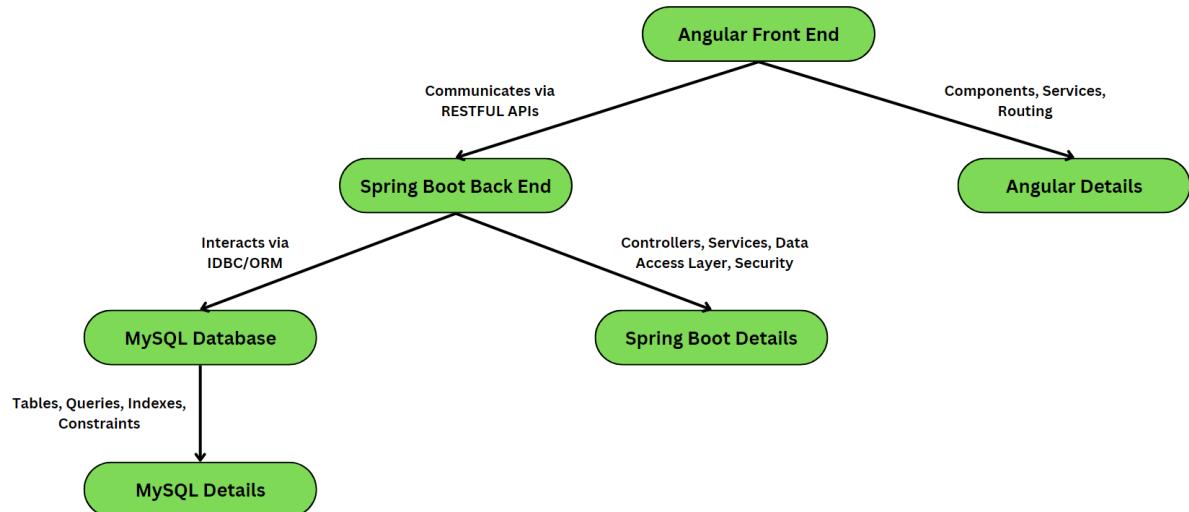


Figure 9 – Logic Architecture

b. Physical Architecture:

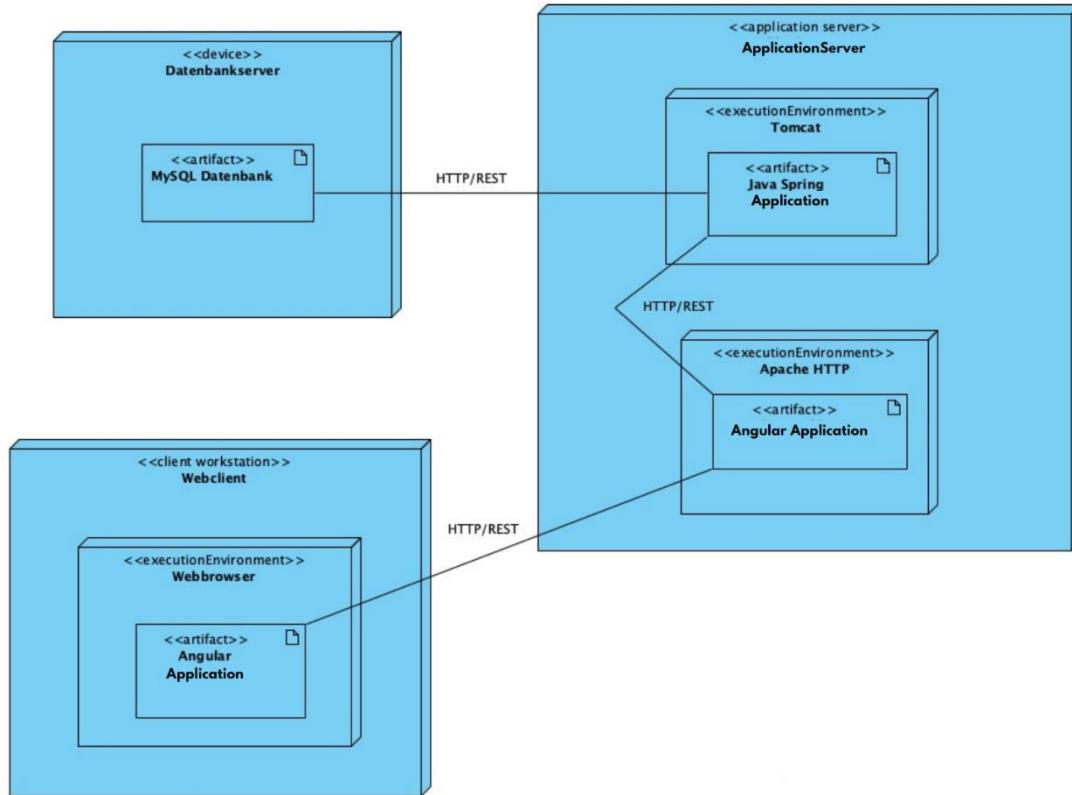


Figure 10 – Physical Architecture

10. Product Backlog

Here is the functional product backlog:

<input type="checkbox"/>	▼ Product backlog 9 mai – 30 juin (51 tickets)	0	0	0	Terminer le sprint	...
<input checked="" type="checkbox"/>	PGS-74 As an admin, i want to connect to my account	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-72 As an administrator agent, i want to connect to my account	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-73 As a company agent, i want to connect to my account	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-24 As a student, i want to reset my password	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-4 As a company agent, i want to reset my password	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-25 As a student, i want to get a temporary password	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-69 As a company agent, i want to get a temporary password	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-70 As an admin, i want to create all accounts	TERMINÉ(E) ▾	-	F		
<input checked="" type="checkbox"/>	PGS-3 As a student, i want to add, edit , delete a complaint	TERMINÉ(E) ▾	-	MH		
<input checked="" type="checkbox"/>	PGS-59 As a company agent, i want to add, edit , delete a complaint	TERMINÉ(E) ▾	-	MH		
<input checked="" type="checkbox"/>	PGS-60 As an Administration agent , i want to consult, add, edit, delete a complaint	TERMINÉ(E) ▾	-	MH		
<input checked="" type="checkbox"/>	PGS-30 As an admin , i want to consult, add, edit, delete a complaint	TERMINÉ(E) ▾	-	MH		
<input checked="" type="checkbox"/>	PGS-8 As an admin, i want to add, edit, delete a subject	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-7 As an company agent, i want to add, edit, delete a subject	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-5 As an administration agent, i want to add, edit, delete a subject	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-9 As a student, i want to display the subjects	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-10 As a student, i want to postulate	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-11 As an admin, i want to accept/decline a postulation	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-61 As a company agent, i want to accept/decline a postulation	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-14 As an admin, i want to display the applications for each subject	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-15 As an administration agent, i want to display the applications for each subject	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-16 As a company agent, i want to display the applications for each subject	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-26 As a student, i want to receive an email in each application response	TERMINÉ(E) ▾	-	YC		
<input checked="" type="checkbox"/>	PGS-18 As an admin, i want to add, edit, delete a post	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-29 As a company agent, i want to add, edit, delete a post	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-31 As a student, i want to add, edit, delete a post	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-62 As an administration agent, i want to add, edit, delete a post	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-19 As a student, i want to like/dislike a post/comment	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-63 As a company agent, i want to like/dislike a post/comment	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-64 As an administration agent, i want to like/dislike a post/comment	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-20 As a student, i want to add, edit, delete a comment to a post	TERMINÉ(E) ▾	-	I		
<input checked="" type="checkbox"/>	PGS-21 As a company agent, i want to add, edit, delete a comment to a post	TERMINÉ(E) ▾	-	I		

<input checked="" type="checkbox"/> PGS-28	As an administration agent, i want to add, edit, delete a comment to a post	TERMINÉ(E) ▾	-	I
<input checked="" type="checkbox"/> PGS-43	As an admin, i want to see all posts	TERMINÉ(E) ▾	-	I
<input checked="" type="checkbox"/> PGS-44	As an admin, i want to see all the comments to all the posts	TERMINÉ(E) ▾	-	I
<input checked="" type="checkbox"/> PGS-42	As a student, i want to report post	TERMINÉ(E) ▾	-	I
<input checked="" type="checkbox"/> PGS-65	As a company agent, i want to report post	TERMINÉ(E) ▾	-	I
<input checked="" type="checkbox"/> PGS-66	As an administration agent, i want to report post	TERMINÉ(E) ▾	-	I
<input checked="" type="checkbox"/> PGS-23	As a student, i want to add, edit a convention	TERMINÉ(E) ▾	-	KK
<input checked="" type="checkbox"/> PGS-39	As a student, i want to add, edit a journal	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-35	As a student, i want to add, edit, delete a journal task	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-36	As an admin, i want to add, edit, delete an evaluation	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-34	As an company agent, i want to add, edit, delete an evaluation	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-37	As a company agent, i want to validate/refuse a journal task	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-67	As a company agent, i want to validate/refuse a journal	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-45	As an admin, I want the bad words to be automatically removed	TERMINÉ(E) ▾	-	I
<input checked="" type="checkbox"/> PGS-46	As a student, I want to receive an internship confirmation sms	TERMINÉ(E) ▾	-	KK
<input checked="" type="checkbox"/> PGS-48	As a student, i want to upload my report	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-49	As a student, I want to obtain the convention in pdf form	TERMINÉ(E) ▾	-	KK
<input checked="" type="checkbox"/> PGS-50	As an admin, i want to get the uploaded files of all the students	TERMINÉ(E) ▾	-	JE
<input checked="" type="checkbox"/> PGS-54	As a student i want to receive a confirmation email when validating or rejecting...	TERMINÉ(E) ▾	-	JE

Figure 11 – Product backlog

11. Conclusion

In conclusion, the implementation of an internship management platform provides an effective solution to address the current challenges of the internship process, offering students, companies, and university administration a centralized and user-friendly tool to facilitate internship management, communication, and evaluation. By integrating advanced features and focusing on process optimization, this platform aims to enhance the overall internship experience while promoting student success and development in both academic and professional endeavors.

Chapter2: Sprint 1

1. Introduction:

In this section, we will present how we worked on the first Sprint. Indeed, Sprint planning is a Scrum event that initiates the sprint. This event aims to define what can be delivered within the sprint. Sprint planning is done collaboratively with the entire Scrum team.

2. Use case diagram:

We present the overall use case diagram for sprint 1 in the following figure

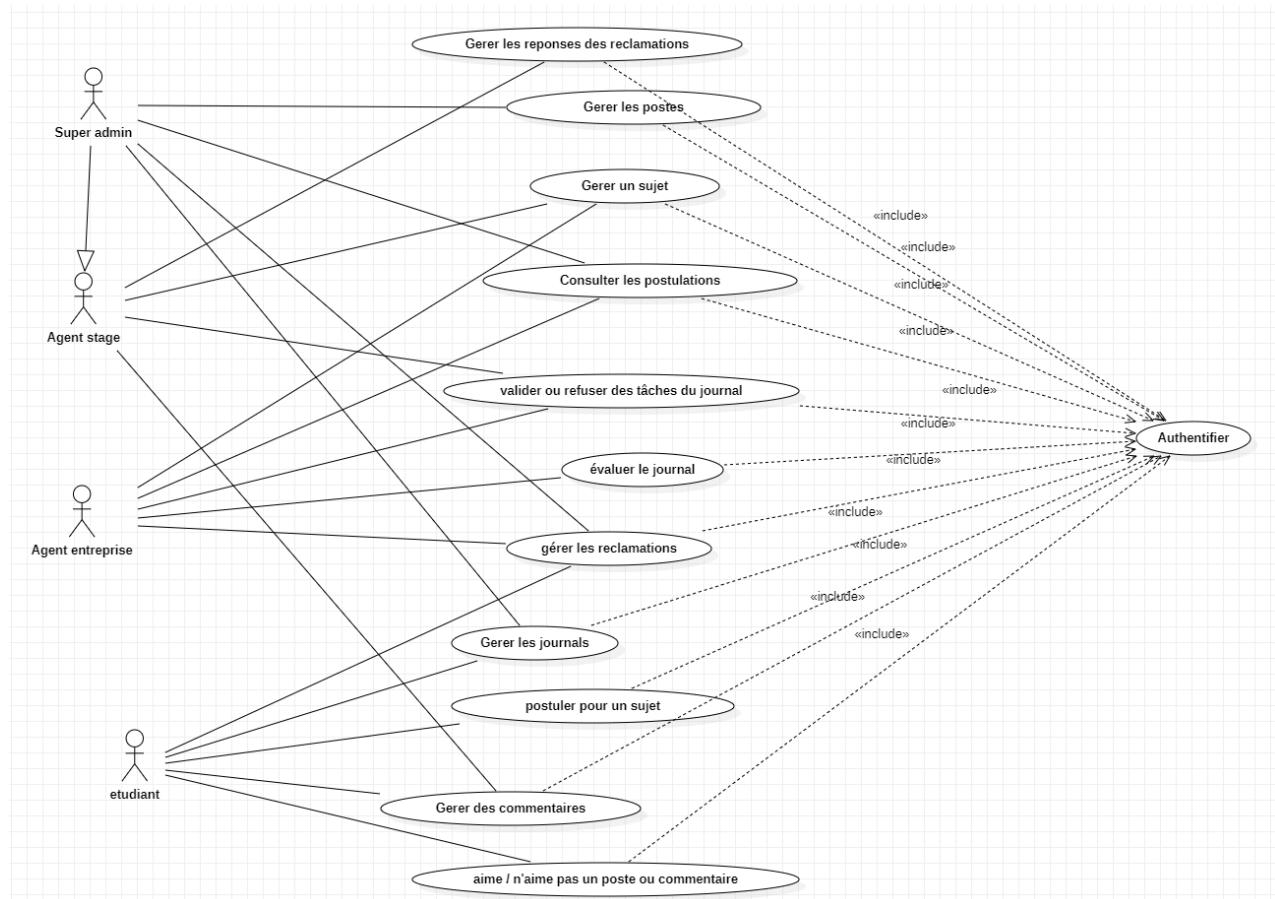


Figure 12 – Use case diagram of sprint 1

- **Use case refinement internship Management**

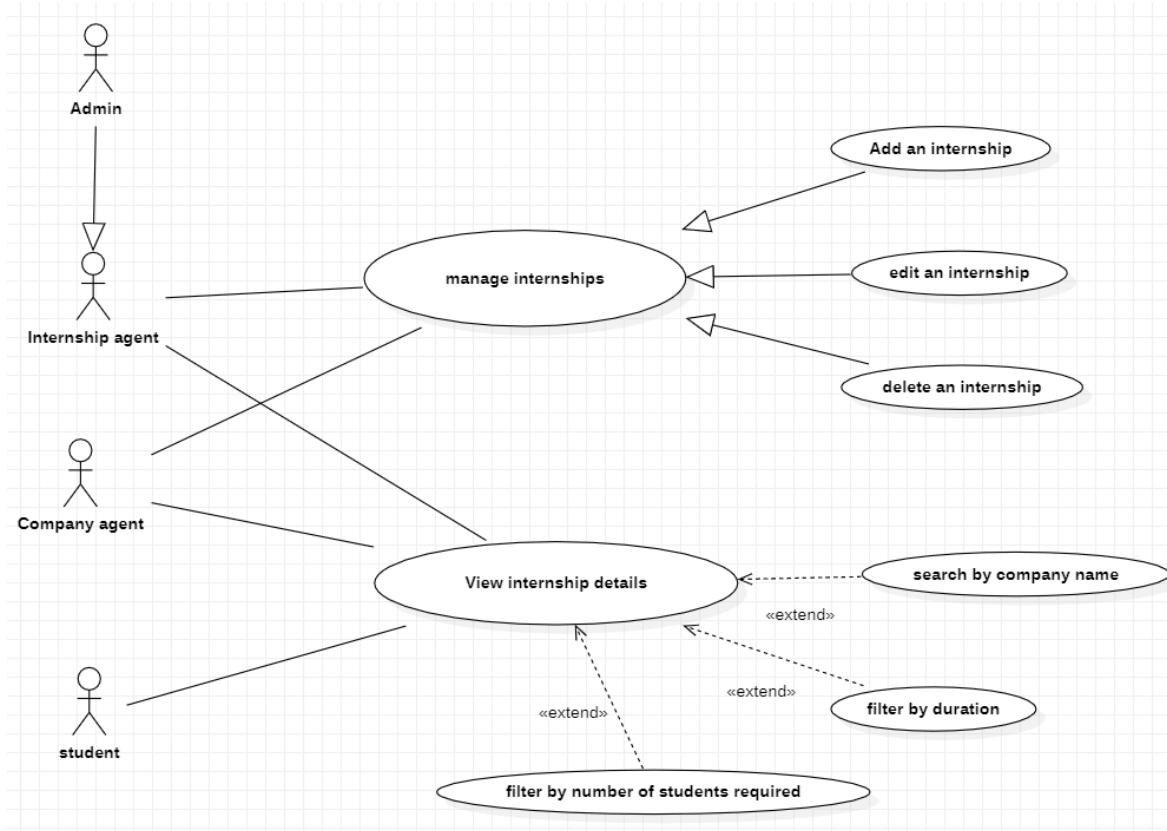


Figure 13 – refinement internship Management

- **Use case refinement Postulation Management**

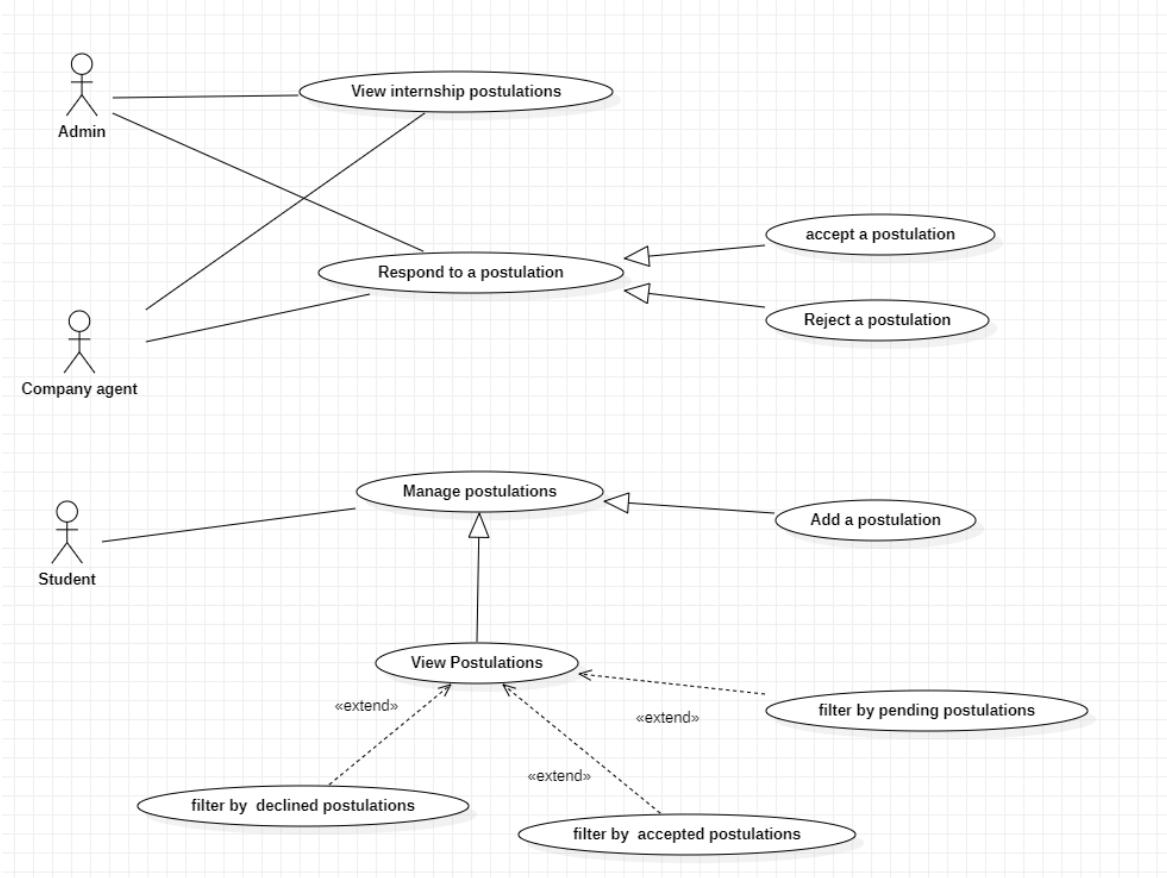


Figure 14 – refinement Postulation Management

- Use case refinement Forum Management

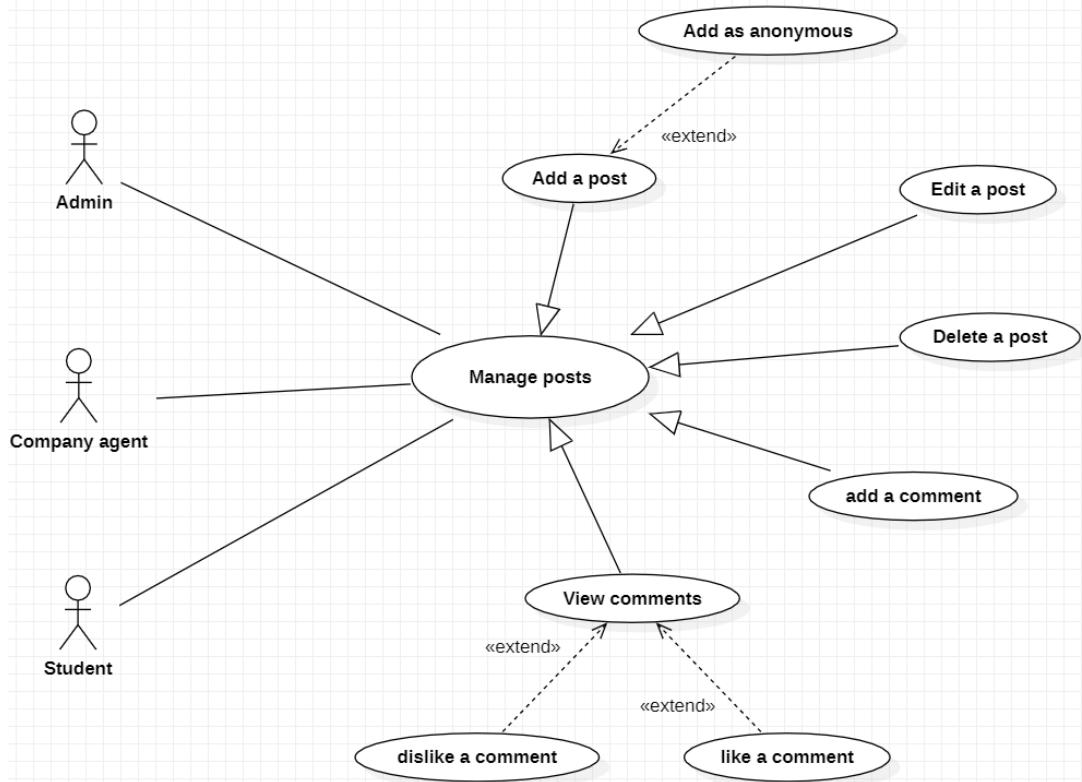


Figure 15 – refinement Forum Management

- Use case refinement Journal and evaluation management

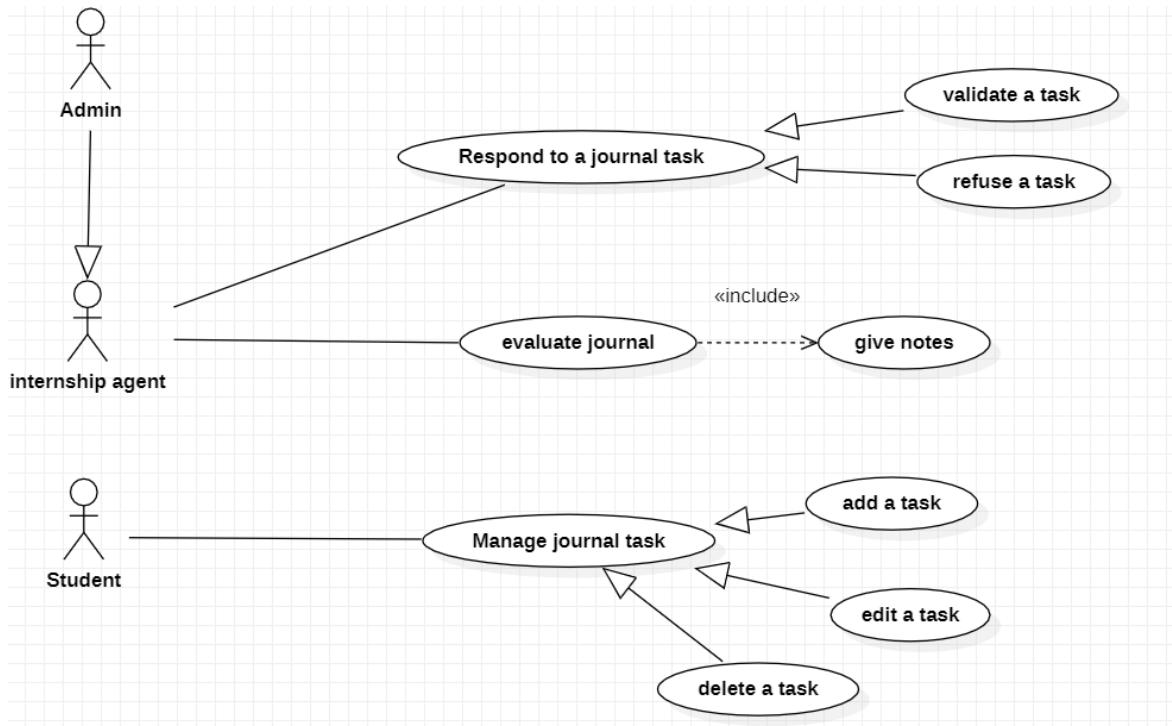


Figure 16 – refinement Journal and evaluation management

- Use case refinement Complaint and Response management

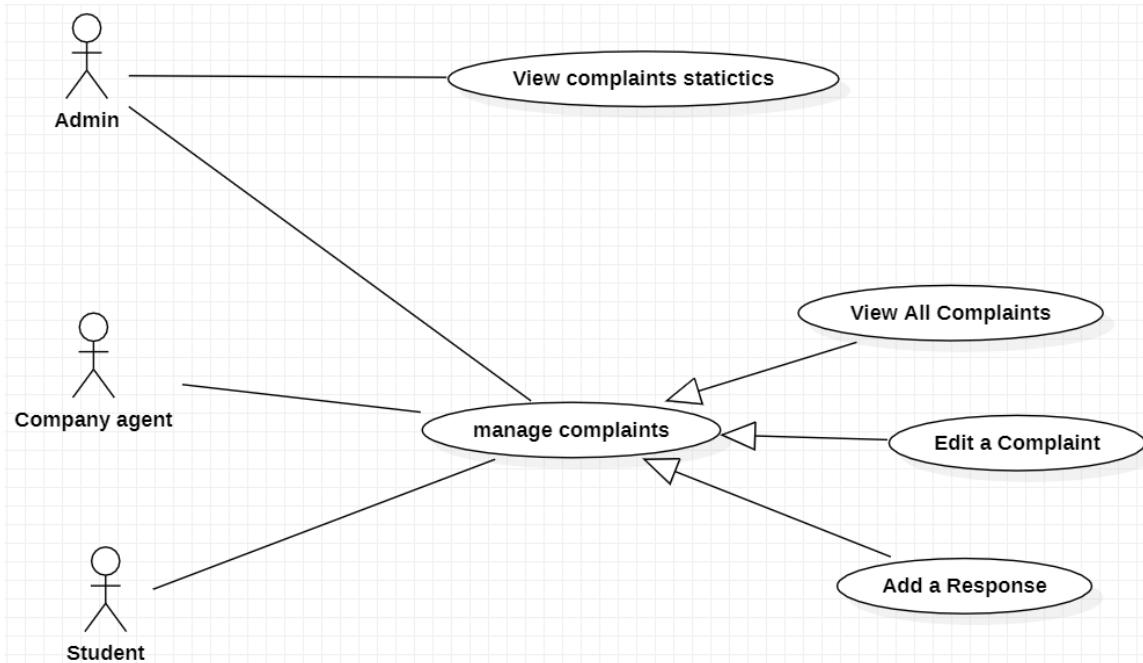


Figure 17 – refinement Complaint and Response management

3. Realization:

a. Subject and Postulation Management

Back Office part: (Admin / Company agent / Internship agent)

In the subject management section, you can perform various actions related to subjects.

- **Add a New Subject**

To add a new subject, click on "Sujet" in the navigation bar. You will be prompted to enter details such as the title, description, duration, location, number of required students, contact email, and any additional requirements for the subject.

The screenshot shows the Admin Dashboard interface. On the left is a sidebar with the 'esp-it' logo and a list of menu items: Comptes, Conventions, Journal, Sujet (which is highlighted in blue), Postulations, Forum, Reclamation, and Account. The main content area is titled 'Admin Dashboard Content'. Below this, there is a sub-section titled 'Content' with a 'Liste des sujets' button. To the right of this is a large form titled 'Ajouter un sujet'. The form fields are as follows:

- Titre du sujet: [Text input field]
- Description: [Text input field]
- Durée: [Text input field]
- Lieu: [Text input field]
- Nombre d'étudiants requis: [Text input field]
- Exigences: [Text input field]
- Type de stage: [Select dropdown]
- Nom de l'entreprise: [Text input field]
- Email de l'entreprise: [Text input field]

Figure 18 – Add a subject interface (admin)

- **View Subject Details:**

You can view the details of each subject by clicking on "Liste des sujets". This action will display all the information related to the subject, including its description, duration, location, contact email, and any additional requirements. Additionally, for each subject, there are buttons to update, delete, and view postulations. You can also search for subjects by the name of the company.

The screenshot shows the Admin Dashboard interface. On the left, a sidebar lists navigation options: Comptes, Conventions, Journal, Sujet, Postulations, Forum, Reclamation, and Account. The main content area is titled "Sujets" and "Gérer les sujets". A specific subject entry for "Esprit" is displayed, titled "Developpement de plateforme de stage". The details include:

- Description:** Developpement de plateforme de stage pour l'automatisation des papiers
- Durée:** 2
- Lieu:** Online
- Contact:** esprit@esprit.tn
- Nombre d'étudiants requis:** 3
- Exigences:** Spring Boot - Angular - SQL - UML
- Type de sujet:** STAGE_IMMERSION_ENTREPRISE

Buttons for "Modifier", "Supprimer", and "Afficher Postulations" are present.

Figure 19 – View subject details (admin)

- **View Subject Postulations:**

When I click on "Afficher Postulations", it displays the postulations for that subject. This allows the admin or agent to accept or decline the offers.

The screenshot shows the Admin Dashboard interface. On the left, a sidebar lists navigation options: Comptes, Conventions, Journal, Sujet, Postulations, Forum, Reclamation, and Account. The main content area displays two postulation entries for the subject "Esprit".

Postulation Details	Status
Titre: Developpement de plateforme de stage Nom entreprise: Esprit Sujet: Candidature stage esprit online Durée: 07-03-2024 - 30-04-2024 Lieu: Online Lettre de motivation: Jeune motivé en informatique Status: En attente	Confirmer Refuser
Titre: Developpement de plateforme de stage Nom entreprise: Esprit Sujet: Candidature khalil Durée: 01-03-2024 - 01-05-2024 Lieu: Online Lettre de motivation: Jeune motivé Status: En attente	Confirmer Refuser

Figure 20 – View subject Postulations (admin)

- **View All Postulations:**

When I click on "Postulations" in the navigation bar, it displays all the postulations that are pending. This allows the admin or agent to accept or decline the offers.

The screenshot shows the Admin Dashboard interface. On the left, there is a sidebar with the 'esprit' logo and a navigation menu containing links: Comptes, Conventions, Journal, Sujet, Postulations, Forum, Reclamation, and Account. The main content area is titled 'Admin Dashboard Content'. It displays two postulation entries, each with a 'Candidature' title, subject details, duration, location, motivation letter, status, and two buttons: 'Confirmer' (green) and 'Refuser' (red). The first postulation is for 'Candidature EY' with the subject 'Récupération/mise à disposition des données can', duration '01-03-2024 - 31-03-2024', location 'Ariana', and motivation 'Étudiant passionné par le développement'. The second postulation is for 'Candidature Herbioty' with the subject 'site e commerce Herbioty', duration '06-03-2024 - 14-03-2024', location 'Sfax', and motivation 'Étudiant passionné par le développement'.

Candidature	Sujet	Durée	Lieu	Motivation	Status	Action
EY	Récupération/mise à disposition des données can	01-03-2024 - 31-03-2024	Ariana	Étudiant passionné par le développement	En attente	Confirmer Refuser
Herbioty	site e commerce Herbioty	06-03-2024 - 14-03-2024	Sfax	Étudiant passionné par le développement	En attente	Confirmer Refuser

Figure 21 – View All Postulations (admin)

Front Office part: (Student)

• View Subjects:

When I click on "Subjects" in the navigation bar, it shows all the information related to the subjects. Additionally, you can search for a specific subject or filter by the number of students or the internship period.

The screenshot shows a web interface for viewing job subjects. At the top, there is a search bar labeled "Chercher par nom d'entreprise" with a dropdown menu set to "par durée". Below the search bar, there are two job listings:

- Telecom**
Smart Diagnoser
Description: Implémenter une solution qui permet de faire des opérations intelligentes de diagnostic
Durée: 10
Lieu: Hammamet
Contact : Telecom@gmail.com
Nombre d'étudiants requis: 2
Exigences: C/C++, SHELL, FLUTTER, PYTHON
Type de sujet: STAGE_INGENIEUR
Postuler
- EY**
Récupération/mise à disposition des données can
Description: Développement d'une application (canmanager) qui permet la lecture/écriture des données can.
Durée: 6
Lieu: Ariana
Contact : EY@gmail.com
Nombre d'étudiants requis: 4

Figure 22 – View subjects (Student)

- **Postulate for a subject:**

When I click on the "Postulate" button, it opens a form where I can enter the necessary information to apply.

The screenshot shows a "Postuler" (Apply) form. The form fields include:

- Titre de la candidature:** (Input field)
- Région:** (Dropdown menu with placeholder "Sélectionnez une région")
- Date de début:** (Input field with placeholder "mm/dd/yyyy"))
- Date de fin:** (Input field with placeholder "mm/dd/yyyy"))
- Lettre de motivation:** (Text area)
- Commentaire:** (Text area)
- Postuler** (Blue button at the bottom)

Figure 23 – Postulate for a subject (Student)

- **View Postulations:**

When I click on "Postulations" in the navigation bar, it displays all the postulations specific to that student. You can filter them by postulations that are pending, accepted, or declined.

Figure 24 – View postulations (Student)

b. Forum Management

Front Office part: (Student/company/teacher)

Here is the front office interface where users can interact with each other adding feedback sharing experiences or asking questions

They can delete it edit it and add comments in comment sections

Subject	Content	Date	Actions
sharing_experience	zazaz	3/7/24, 12:02 AM	Edit Delete View Comments
sharing_experience	azzaza	3/7/24, 12:08 AM	Edit Delete View Comments
feedback	i had a wonderful exp	3/7/24, 8:32 AM	Edit Delete View Comments

Figure 25 –Posts management

In this interface it's where the user sees his own posts and add new ones

The screenshot shows a list of posts on a platform called "esprit". There are three visible posts:

- sharing_experience** (zadaz) - posted at 3/7/24, 12:02 AM. Has 1 Like and 1 Dislike.
- sharing_experience** (azzaza) - posted at 3/7/24, 12:08 AM. Has 1 Like and 0 Dislikes.
- feedback** - posted at 3/7/24, 8:32 AM. Has 1 Like and 0 Dislikes.

Each post includes a "View Comments" button, an "Add Comment" button, a "Report Post" button, and "Like" and "Dislike" buttons. A "My Posts" button is also present.

Figure 26 –Comments management

As a user, you can also like and dislike posts. Additionally, you have the ability to dislike or undislike them. Furthermore, there is a report button where you can report posts, adding a reason. If a post receives a certain number of reports, it will be deleted.

Back Office part: (Admin)

In the back-office part, the admin can control add and manage all posts

The screenshot shows the back-office management interface for posts. On the left, a sidebar menu includes "Comptes", "Conventions", "Journal", "Sujet", "Forum", "Reclamation", and "Account".

The main area is titled "Posts" and displays a table of existing posts:

Subject	Content	Date	Actions
sharing_experience	zadaz	3/7/24, 12:02 AM	Edit Delete View Comments
sharing_experience	azzaza	3/7/24, 12:08 AM	Edit Delete View Comments
feedback	i had a wonderful exp	3/7/24, 8:32 AM	Edit Delete View Comments

Below the table, there is a section titled "Add New Post" with fields for "Subject", "Content", and a checkbox for "Anonymous". A "Add Post" button is located at the bottom of this section.

Figure 27 –Manage posts

c. Journal and evaluation management

Front Office part: (internship agent)

As an internship agent, you can validate or refuse each task in the journal

The screenshot shows the 'Journal Details' section of a web application. It includes a remark 'Stage 4eme', a total evaluation score of 19.55, and a status 'Journal Validé'. Below this is a 'Tache List' table:

Description	Date	Status	Actions
Architecture et Conception	2024-03-29T15:46:11	Validé	Valider A Refaire
Installation Framework	2024-03-29T15:46:58	Validé	Valider A Refaire
Integration APIo	2024-03-29T15:47:25	Validé	Valider A Refaire
ARARAR	2024-03-31T02:20:47	Validé	Valider A Refaire

Figure 28 –Journal section

In the end, he can have access to the evaluation part so he can evaluate and give notes

The screenshot shows the 'Evaluations' section of a web application. It displays five evaluation items with their notes and appreciation levels:

Description	Note	Appréciation
Sait être attentif(ve)	3	TRESBON
Capacité de communiquer par écrit	4	EXCELLENT
Motivation personnelle de l'étudiant	4	EXCELLENT
Réalise un travail avec soin et précision	4	EXCELLENT
Est capable de travailler seul	4	EXCELLENT

Figure 29 –Journal section

Front Office part: (student)

The student can add, update, or delete the journal tasks

Journal Details

Remark: Stage 4eme

Total Evaluation Score: 19.55

Etat Journal: Journal Validé

Tache List

Tache List			
Description	Date	Status	Actions
Architecture et Conception	Mar 29, 2024	Validé	Modifier Supprimer
Installation Framework	Mar 29, 2024	Validé	Modifier Supprimer
Integration APIo	Mar 29, 2024	Validé	Modifier Supprimer
ARARAR	Mar 31, 2024	Validé	Modifier Supprimer

Figure 30 –Manage journal tasks

d. Complaint and Response management

Back Office part: (Admin)

In the Complaint and Response management, you can perform various actions related to either the Complaint or the Response:

- View All Complaints**

When I click the “Reclamation” button it shows a list with all the complaints received and I have various actions that I can do (Edit, Delete and Reply) to these complaints

Liste des Réclamations							
ID	Titre	Type de Réclamation	Description	Date de Réclamation	Statut	Réponse	Actions
2	test	AUTRES	tesssstttt	2024-03-04T14:57:06	REJETE	Répondre Supprimer	Modifier
3	une autre fois	MODIFICATION	AAAAAAAAAAAAAA	2024-03-04T14:57:19	APPROUVE	Répondre Supprimer	Modifier
5	tessst 2	MODIFICATION	tessst 2	2024-03-06T01:29:45	REJETE	Répondre Supprimer	Modifier
7	AAAAAA	ANNULATION	AAAAAAAAAA	2024-03-06T01:38:10	ENATTENTE	Répondre Supprimer	Modifier
8	zzzzz	AUTRES	zzzzzz	2024-04-04T03:05:17	ENATTENTE	Répondre Supprimer	Modifier
9	ssss	ANNULATION	ssss	2024-04-15T16:02:59	ENATTENTE	Répondre Supprimer	Modifier
10	ssssssssss	ANNULATION	ssssssssss	2024-04-15T16:03:54	ENATTENTE	Répondre Supprimer	Modifier
11	TTTTTTTTTT	ANNULATION	TTTTTTTTTT	2024-04-15T16:04:11	ENATTENTE	Répondre Supprimer	Modifier

Figure 31 –view complaints

In this interface you can see all the complaints with the time the complaint was sent and their status (By default “En Attente”) till you reply or reject it, you can search for a specific complaint by title, delete a specific complaint, edit it, and reply to this complaint.

- Add a Response (Reply to a specific Complaint):**

When I click on the “Répondre” button in the Complaints page it shows a pop up for the Response form to reply to the specific Complaint

	ID	Type de	Date de	Status	Action
8	zzzzz	AUTRES	xxxxxx	2024-04-04T03:05:17	ENATTENTE Répondre Modifier
9	ssss	ANNULATION	ssss	2024-04-05T16:02:59	ENATTENTE Répondre Modifier
10	ssssssssss	ANNULATION	ssssssssss	2024-04-05T16:03:54	ENATTENTE Répondre Modifier
11	TTTTTTTTTT	ANNULATION	TTTTTTTTTTTT	2024-04-05T16:04:11	ENATTENTE Répondre Modifier

Figure 32–add a response

- Edit a Complaint:**

When I click on the “Modifier” button in the Complaints page it redirects you to the selected complaint with all her values and you have the access to modify it, then click “Mettre à jour” button to save your changes

Figure 33–edit a complaint

- Statistics:**

When I click on the “Statistics” button It counts complaint and shows a statistic of the complaints by type (Type de réclamation)

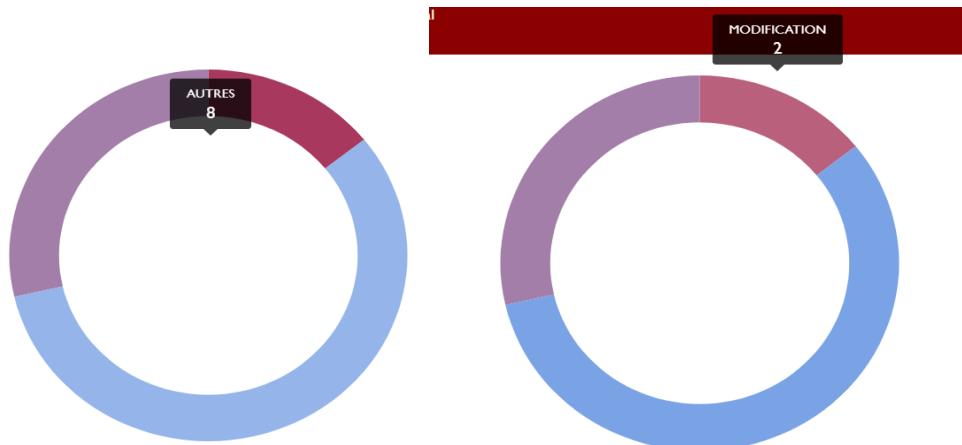


Figure 34 –statictics

Front Office part: (Student)

The students can add, edit, and delete their complaints.

With the necessary input control for the vulgar words or empty fields

Réclamation

Title:	<input type="text"/>
Type de Réclamation	<input type="text" value="Type de Réclamation*"/> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Annulation <input type="checkbox"/> Modification <input type="checkbox"/> Autres
Description:	<input type="text"/>
<input type="button" value="Annuler"/> <input type="button" value="Confirmer"/>	

Figure 35 –manage complains

e. Users management

- **Authentication and Authorizations:**

All users are able to login securely through the login page provided by Keycloak.



Once the user is logged in, he will be automatically redirected, based on their role, to the front or the back office.

Super Admin and admins(esprit agents) are redirected to the back office.

Students and enterprise agents are redirected to the front office.

Back Office part: (Super-Admin and admins)

Super Admin and admins will be able to view the list of users, add accounts (Super-Admin only), update user data and delete accounts (Super-admin only).

The screenshot shows the 'esprit' back office interface. On the left, there's a red sidebar with various menu items like 'Comptes', 'Conventions', 'Journal', etc. The main area has a header 'Welcome super-admin' and a section titled 'List of Users'. A table displays five user entries:

Login	Email	First Name	Last Name	Role	Num Tel	Role Entreprise	Identifiant	Classe	Specialite	Action	Action
firas	firas.jery@gmail.com	firas	jery	etudiant	0		223JMT0955	4SAE1	GL	Update	Delete
admin	admin@gmail.com	admin	admin	Agentesprit	0					Update	Delete
encadrant	encadrant@esprit.tn	encadrant	encadrant	Agententreprise	0					Update	Delete
RhEntreprise	RH@esntreprise.tn	RH	RH	Agententreprise	0					Update	Delete

Figure 36 –List of users

Login:

Email:

First Name:

Last Name:

Password:

Role:

Submit

Upload list of students

No file chosen

Figure 37 – Add user

Login:

Email:

First Name:

Last Name:

Role:

Identifiant:

Classe:

Specialite:

Submit

Figure 38 – Update user

4. Conclusion:

In conclusion, the subject and postulation management system provide an effective platform for streamlined administration and interaction. By offering intuitive interfaces for users and comprehensive functionalities for handling subjects, applications, and forum interactions, it enhances efficiency and communication in academic and professional settings.

Chapter 3 Sprint 2:

1. Introduction:

During the second Sprint, we honed user experience and functionality in our system. Collaborative planning guided efforts to enhance efficiency for administrators, agents, and students. Key features like confirming postulations and managing agreements were improved to streamline operations. Let us explore our progress in detail.

2. Use case diagram:

We present the overall use case diagram for sprint 2 in the following figure

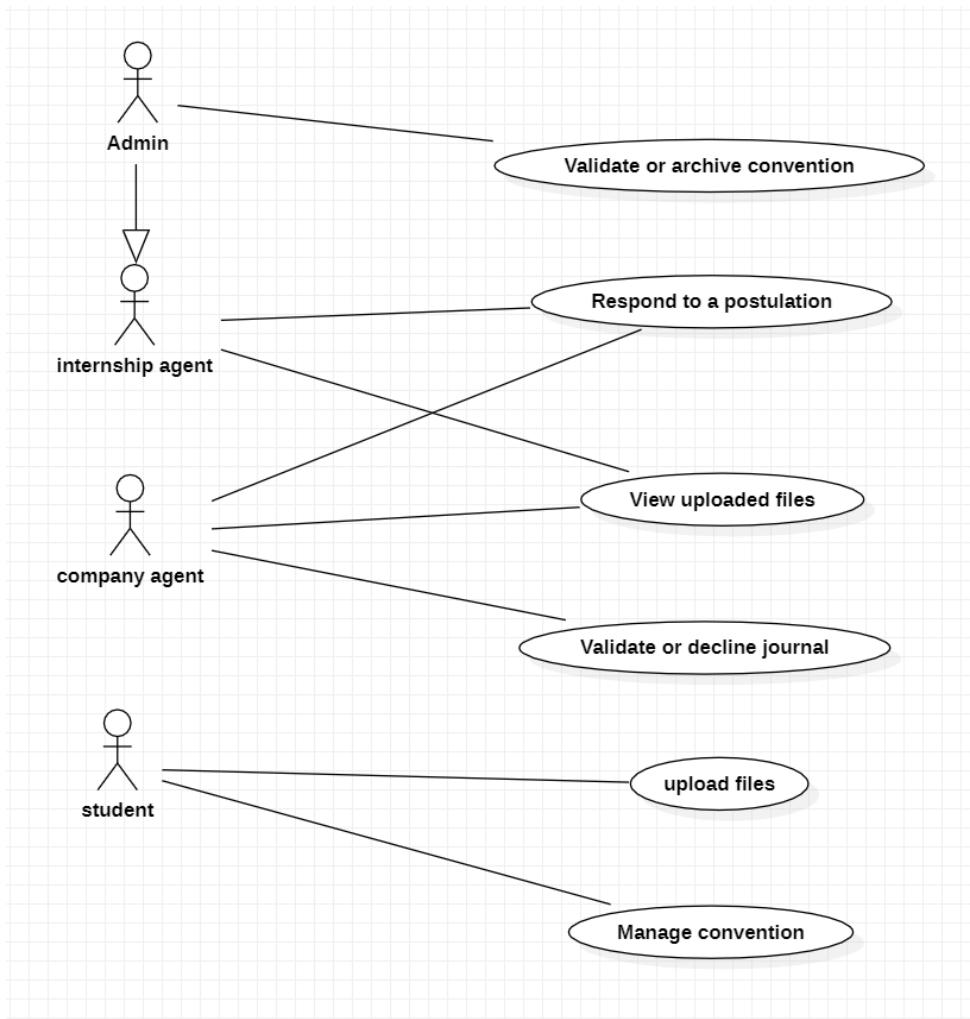


Figure 39 – Use case diagram of sprint 2

- Use case refinement Postulation Management

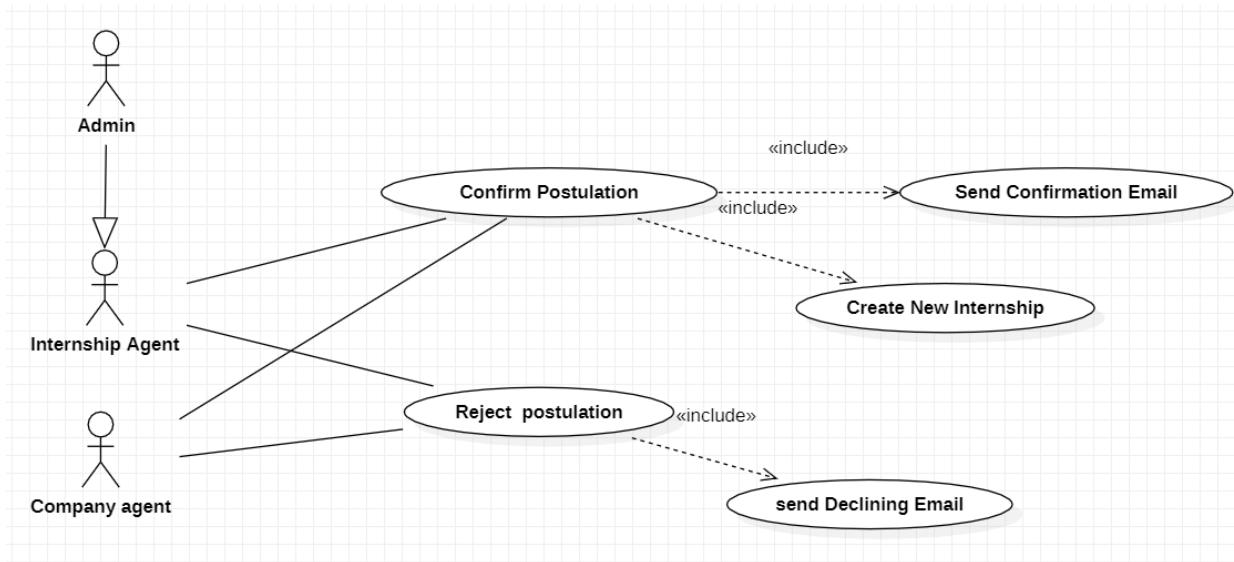


Figure 40 – refinement Postulation Management

- Use case refinement File Management

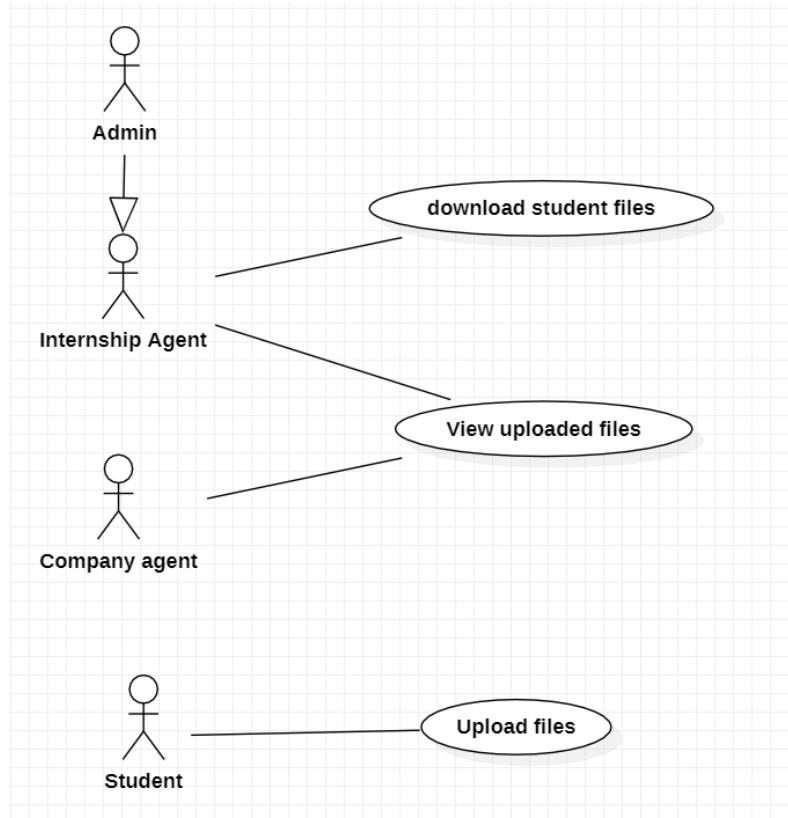


Figure 41– refinement File Management

- Use case refinement convention Management

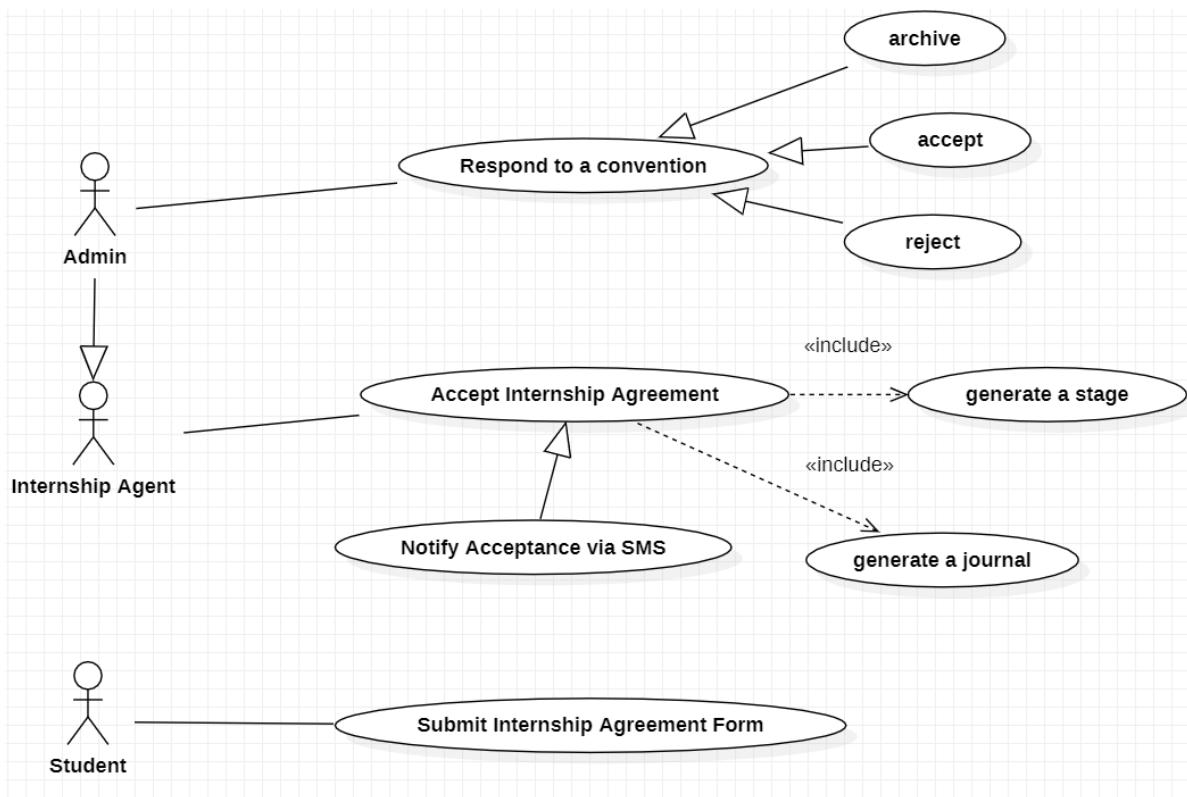


Figure 42– refinement convention Management

- **Use case refinement journal Management**

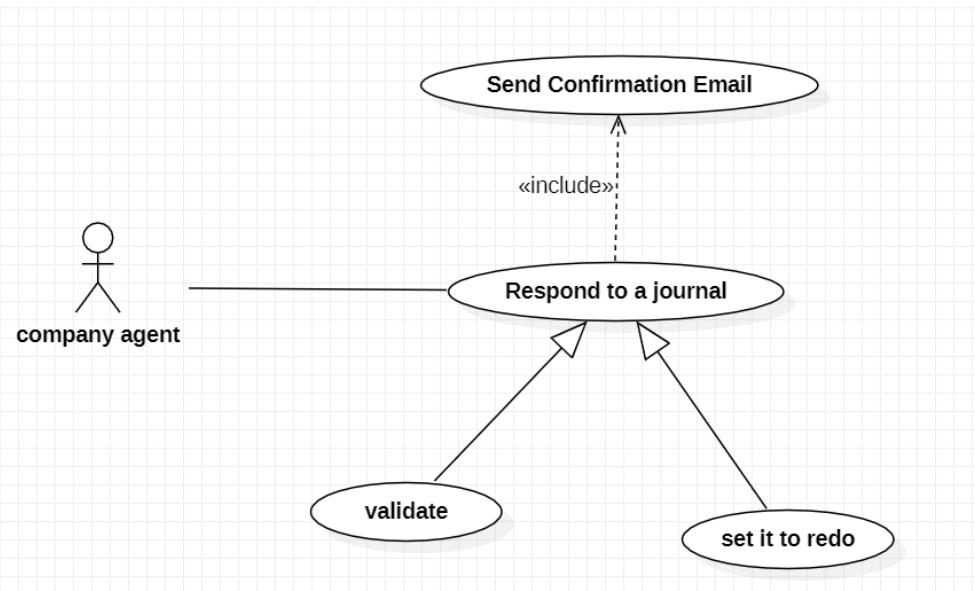


Figure 43– refinement journal Management

3. Realization:

a. Manage postulations (Admin):

In the postulation management section, you can perform various actions related to postulations.

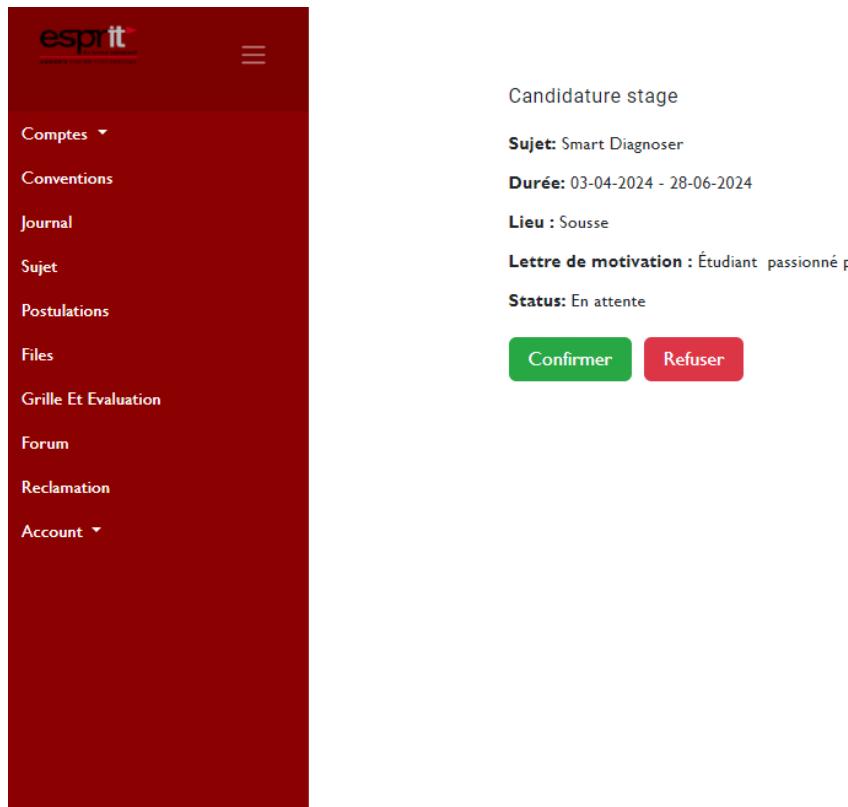


Figure 44– Confirm postulation

To confirm a postulation, both Company agent and internship agent should accept it, so a confirmation mail will be sent to the student and a new internship will be created for that student.

Confirmation of your internship application Inbox x



indila205@gmail.com

to me ▾

Hello Cherif Yassmine, your internship application for the company Esprit has been accepted.

--

Cet e-mail a été vérifié par le logiciel antivirus d'Avast.

www.avast.com

Reply

Forward



Figure 45– Confirmation mail

b. Manage files:

- View uploaded files

The Company agent and internship agent can view the uploaded files of each student.

The screenshot shows a sidebar menu on the left with categories like Comptes, Conventions, Journal, Sujet, Postulations, Files, Grille Et Evaluation, Forum, Reclamation, and Account. The main content area is titled "Liste des fichiers envoyés". It displays a table with three rows:

Nom de l'étudiant	Type	Lien vers le fichier
223AFT0324	DEMANDE_STAGE	Lien
223AFT0324	ATTESTATION	Lien
223AFT0324	DEMANDE_STAGE	Lien

Figure 46– View uploaded files

- **Upload files:**

The student can upload his own files (Report /attestation internship / internship application)

The screenshot shows a navigation bar at the top with links to Home, Convention, Sujets, Mes postulations, Forum, Mes fichiers, Reclamation, Journal, Stage En Cours, and Mes Convention. Below the navigation bar is a large white form area with three sections for file uploads:

- A section labeled "Ajouter votre demande de stage" with a "Choose File" button, a "No file chosen" message, and a "Télécharger" button.
- A section labeled "Ajouter votre attestation" with a "Choose File" button, a "No file chosen" message, and a "Télécharger" button.
- A section labeled "Ajouter votre Rapport" with a "Choose File" button, a "No file chosen" message, and a "Télécharger" button.

Figure 47– upload files

c. **Manage convention:**

Submission of Internship Agreement Forms: Students complete and submit their internship agreement forms online. Upon submission, the form is automatically downloaded to the student's computer.

Acceptance and Notification: Once an internship agreement is accepted by an administrator, the system automatically: Generates a stage and journal for the student to track their internship progress.

Sends an SMS notification to the contact number provided in the internship agreement. This message

confirms the acceptance of the internship and provides any necessary next steps or details. During the Internship: Students can upload their reports directly to their assigned stage. This allows for easy documentation and review by administrators.

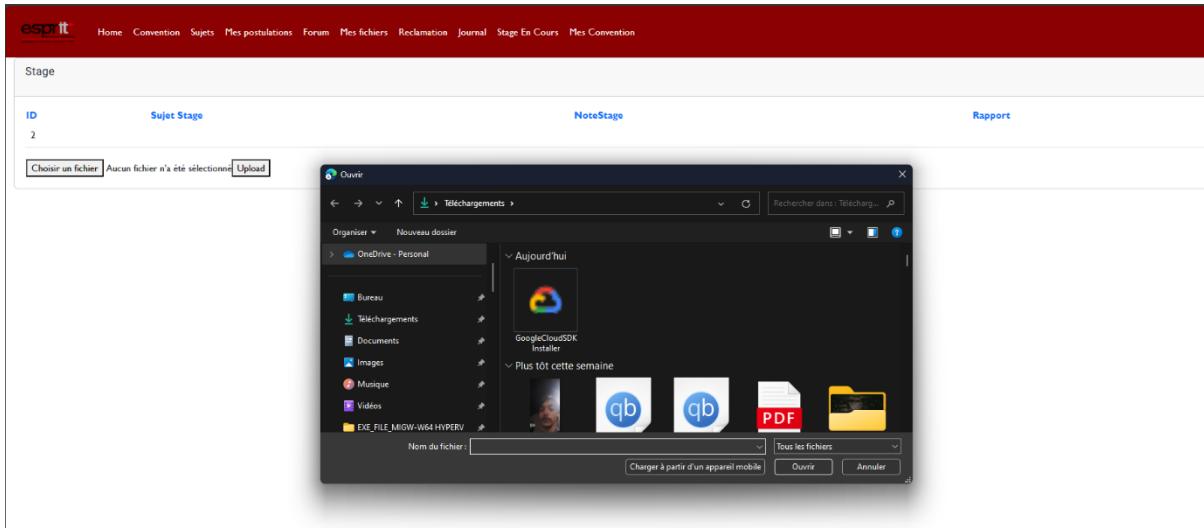


Figure 48– upload convention

- **Validate or archive conventions:**

Administration Management: On the administrative side, all submitted agreements are visible in a dashboard where administrators can accept, reject, or archive these agreements as needed once accepted an sms will be sent to the students .

ID	Nom Entreprise	Date Début	Date Fin	Adresse	Num Téléphone	Nom Encadrant	Email Encadrant	Validité	Actions
2	add	7/18/24	7/18/24	add	21456548	add	add@add.add	Not Validated	<button>Validate</button> <button>Archive</button>
1	azazaza	7/20/24	7/20/24	azzaza	21456548	zaaz	zazaz@azaz.com	Validated	<button>Validated</button> <button>Archive</button>

The screenshot shows a table titled 'Convention' with a header 'List of Conventions for Validation'. The table contains two rows of data. The first row has ID 2, enterprise name 'add', start date '7/18/24', end date '7/18/24', address 'add', phone number '21456548', supervisor name 'add', supervisor email 'add@add.add', and status 'Not Validated'. The second row has ID 1, enterprise name 'azazaza', start date '7/20/24', end date '7/20/24', address 'azzaza', phone number '21456548', supervisor name 'zaaz', supervisor email 'zazaz@azaz.com', and status 'Validated'. Below the table, there is a section titled 'Archived Conventions'.

Figure 49– validate or archive convention

Archive Convention							
List of Archived Conventions							
ID	Nom Entreprise	Date Début	Date Fin	Adresse	Num Téléphone	Nom Encadrant	Email Encadrant
1	azzazzazza	7/20/24	8/23/24	azzaza	21456548	azzazzazza	zazaz@azaz.com

[Get back from Archive](#)

[Back To Conventions](#)

Figure 50– archive convention

d. Manage journal:

The company agent has the access to validate or set it to redo and a confirmation mail will be sent to the student



Figure 51 – Confirmation mail

e. Users management

- **Upload a list of students:**

The super admin will be able to select a list of students within a csv file, the file will be parsed and accounts are automatically created and sent by email to each student.

Upload list of students

No file chosen
Upload

Figure 52 – Upload students list

- **Temporary password:**

All created accounts have a temporary password, so when the user logs in for the first time, they are prompted to set their password.

The screenshot shows a web-based password update form. At the top, a yellow warning box contains the text: "You need to change your password to activate your account." Below this, there are two input fields: "New Password" and "Confirm password", each accompanied by a small "eye" icon to show the typed characters. Underneath the fields is a checked checkbox labeled "Sign out from other devices". At the bottom of the form is a large blue "Submit" button.

Figure 53 – Update temporary password

- **Reset password:**

In case a user forgot their password, a reset password link is provided in the login page. All users can click the link and proceed to the reset password process, which starts with sending an email containing the reset link. Once the link is clicked, the user can enter their new password.

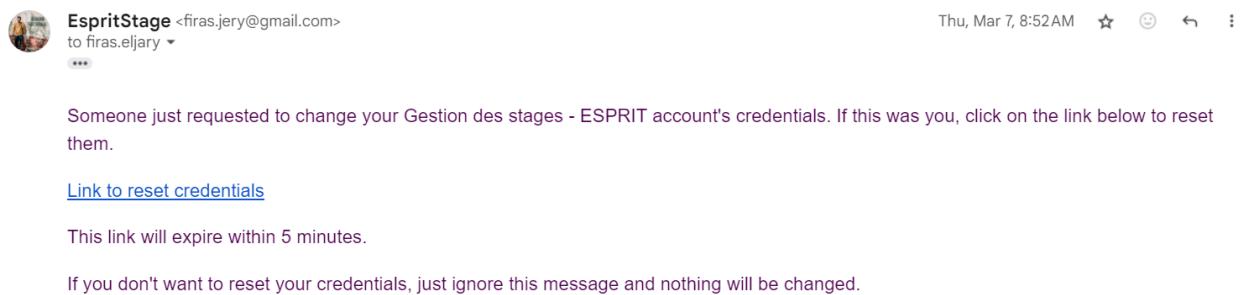


Figure 54 – Reset password mail

4. Conclusion

In summary, the second Sprint concentrated on refining user experience and functionality, facilitating smooth operations for administrators, agents, and students. Collaborative planning ensured clear objectives, driving effective implementation of key features like confirming postulations, managing files, and handling internship agreements. Efforts were directed towards optimizing usability and productivity, enhancing the system's efficiency in managing internships and interactions.

Chapter 4 Sprint 3:

1. Introduction

During the third sprint, we shift our focus to data analysis and prediction, a pivotal phase in our project. Over three weeks, we'll delve into data mining, aiming to extract insights and forecast trends. Sessions will cover formulating business goals, selecting algorithms, and preparing data. Coaching will ensure effective implementation, culminating in individual validation sessions. Our aim is to enhance decision-making through data-driven insights. Let's dive into this exploration together.

2. Realization

a. Regional Patterns: Statistics & Clustering Analysis

This figure presents a portion of the dataset utilized.

region	classe
Tunis	4SAE5
Gafsa	4BI8
Mahdia	4SIM9
Tataouine	4TWIN10
Medenine	4NIDS5
Kebili	4SLEAM4
Tunis	4SE2
Sidi Bouzid	4ArcTIC3
Monastir	4DS7
Bizerte	4Infini6
Ariana	4SAE3
Jendouba	4BI9
Ben Arous	4SIM2
El Kef	4TWIN1
Sfax	4NIDS8
Kairouan	4SLEAM10
Beja	4SE7
Sousse	4ArcTIC4
Nabeul	4DS6
Kasserine	4Infini5
Siliana	4SAE10
Tozeur	4BI1
Tunis	4SIM3
Tunis	4TWIN8
Medenine	4NIDS7
Sidi Bouzid	4SLEAM6
Monastir	4SE4
El Kef	4ArcTIC9
Kebili	4DS2

Figure 55– Region dataset

With K-means clustering, we analyzed student internship applications to identify distinct groups based on shared characteristics such as geographical region and specialization preferences. By assigning each application to a cluster, we were able to uncover patterns and trends in the distribution of applications, providing valuable insights into regional and specialization preferences among students. This clustering approach helps educational institutions and employers better understand student career aspirations and tailor internship opportunities to match their interests and needs.

These figures depict the code of the K-means algorithm:

```

# Import necessary libraries
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
import seaborn as sns # Import Seaborn for visualization
import warnings

# Load the dataset
dataset = pd.read_csv('/content/region_classe.csv')

# Preprocess the data
# Encode categorical variables: Regions and Classes
X = dataset[['region', 'classe']]
X_encoded = X.apply(lambda x: pd.factorize(x)[0]) # Assign numerical labels to categories
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X_encoded) # Standardize the features to have a mean of 0 and a standard deviation of 1

# Initialize lists to store inertias and number of clusters
inertias = []
num_clusters_range = range(1, 11) # Test different numbers of clusters from 1 to 10

# Calculate inertia for different numbers of clusters
for k in num_clusters_range:
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(X_scaled)
    inertias.append(kmeans.inertia_)

# Plot the elbow curve to find the optimal number of clusters
plt.plot(num_clusters_range, inertias, marker='o')
plt.title('Elbow Method for Optimal Number of Clusters')
plt.xlabel('Number of Clusters')
plt.ylabel('Inertia')
plt.xticks(num_clusters_range)
plt.show()

# Preprocess the data
# Encode categorical variables: Regions and Classes
X = dataset[['region', 'classe']]
X_encoded = X.apply(lambda x: pd.factorize(x)[0]) # Assign numerical labels to categories
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X_encoded) # Standardize the features to have a mean of 0 and a standard deviation of 1

# Choose the number of clusters (K)
n_clusters = 4

# Apply K-means clustering
kmeans = KMeans(n_clusters=n_clusters, random_state=42)
clusters = kmeans.fit_predict(X_scaled)

# Add cluster labels to the dataset
dataset['cluster'] = clusters

# Set plot size
plt.figure(figsize=(20, 18))

# Visualize the clusters using a scatter plot
sns.scatterplot(data=dataset, x='region', y='classe', hue='cluster', palette='viridis', s=150)
plt.title('K-means Clustering of Regions and Classes')
plt.xlabel('Region')
plt.ylabel('Class')
plt.xticks(rotation=45, ha='right')
plt.legend(title='Cluster')
plt.show()

```

Figure 56– Kmeans code

In this figure, we observe the outcome/result:

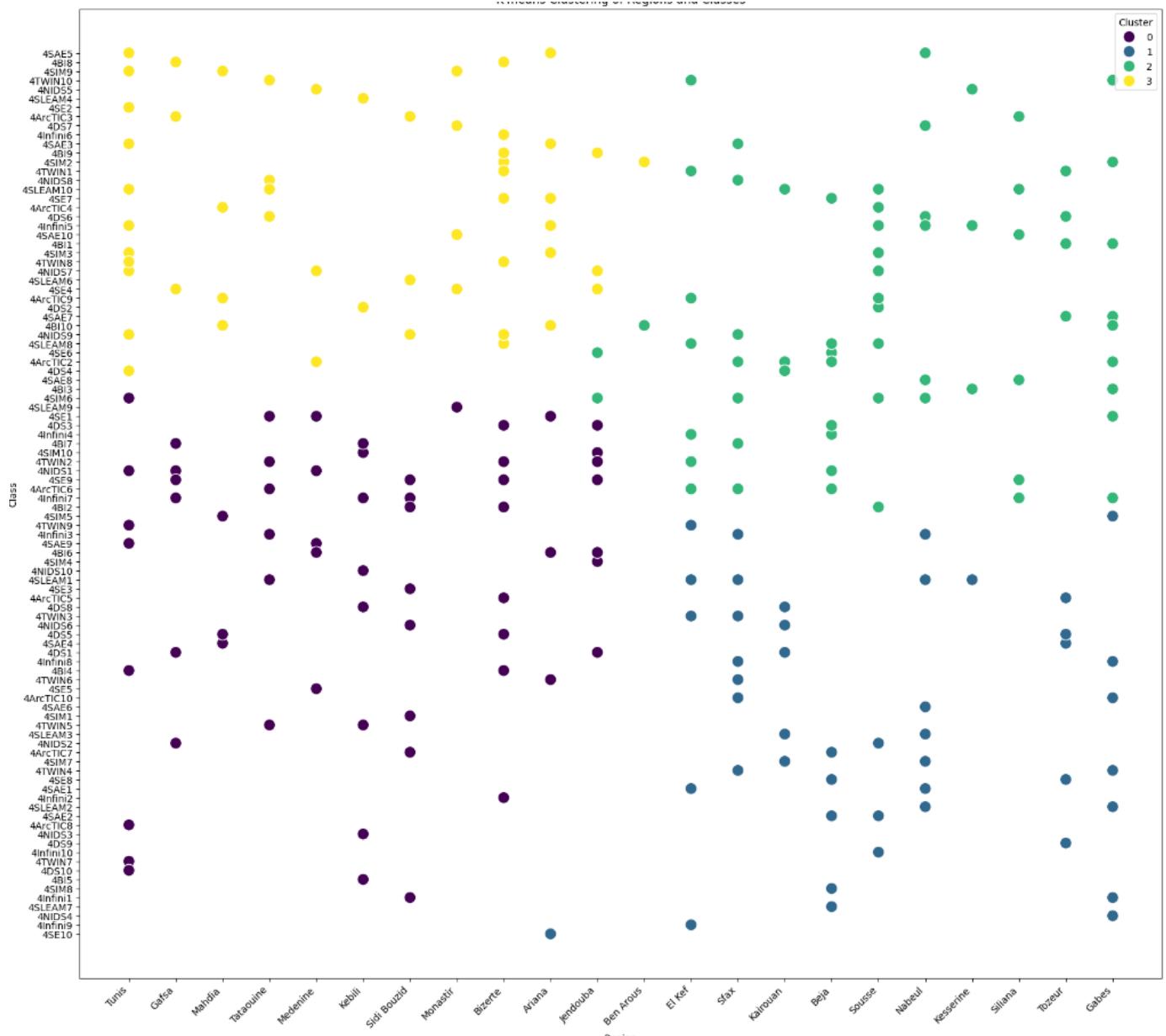


Figure 57– Kmeans output

We also examined the dataset of student internship applications. After checking its structure and ensuring completeness, we visualized the distribution of applications across regions using bar plots and pie charts. These visualizations offer insights into regional preferences among students, aiding institutions, and employers in optimizing internship opportunities.

These figures illustrate the statistical analysis code:

```

import warnings
warnings.filterwarnings("ignore")

# Load the dataset
dataset = pd.read_csv('/content/region_classe.csv')

# Display the first few rows of the dataset
print("First few rows of the dataset:")
print(dataset.head())

# Display basic information about the dataset
print("\nBasic information about the dataset:")
print(dataset.info())

# Check for any missing values
print("\nMissing values in the dataset:")
print(dataset.isnull().sum())

# Calculate basic statistics for the 'region' column
# Title Comment: Calculate region counts
region_counts = dataset['region'].value_counts()
print("\nRegion Counts:")
region_counts_df = pd.DataFrame(region_counts)
region_counts_df.columns = ['Number of Applications']
print(region_counts_df)

# Visualize the distribution of regions using a bar plot
# Title Comment: Visualize region distribution with a bar plot
plt.figure(figsize=(10, 6))
region_counts.plot(kind='bar', color='skyblue')
plt.title('Distribution of Regions')
plt.xlabel('Region')
plt.ylabel('Number of Applications')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()

# Sort region_counts by values
# Title Comment: Sort region counts
sorted_region_counts = region_counts.sort_values(ascending=False)

print("\n\n")

# Visualize the distribution of regions using a pie chart with sorted percentages
# Title Comment: Visualize region distribution with a pie chart
plt.figure(figsize=(10, 8))
plt.pie(sorted_region_counts, labels=sorted_region_counts.index, autopct='%.1f%%', startangle=140)
plt.title('Distribution of Regions (Sorted)')
plt.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.
plt.tight_layout()
plt.show()

# Sort region_counts by values
# Title Comment: Sort region counts
sorted_region_counts = region_counts.sort_values(ascending=False)

print("\n\n")

# Visualize the distribution of regions using a horizontal bar chart
# Title Comment: Visualize region distribution with a horizontal bar chart
plt.figure(figsize=(10, 8))
sorted_region_counts.plot(kind='barh', color='orange')
plt.title('Distribution of Regions (Sorted)')
plt.xlabel('Number of Applications')
plt.ylabel('Region')
plt.tight_layout()
plt.show()

```

Figure 58– stats code

In these figures, we observe the outcome/result:

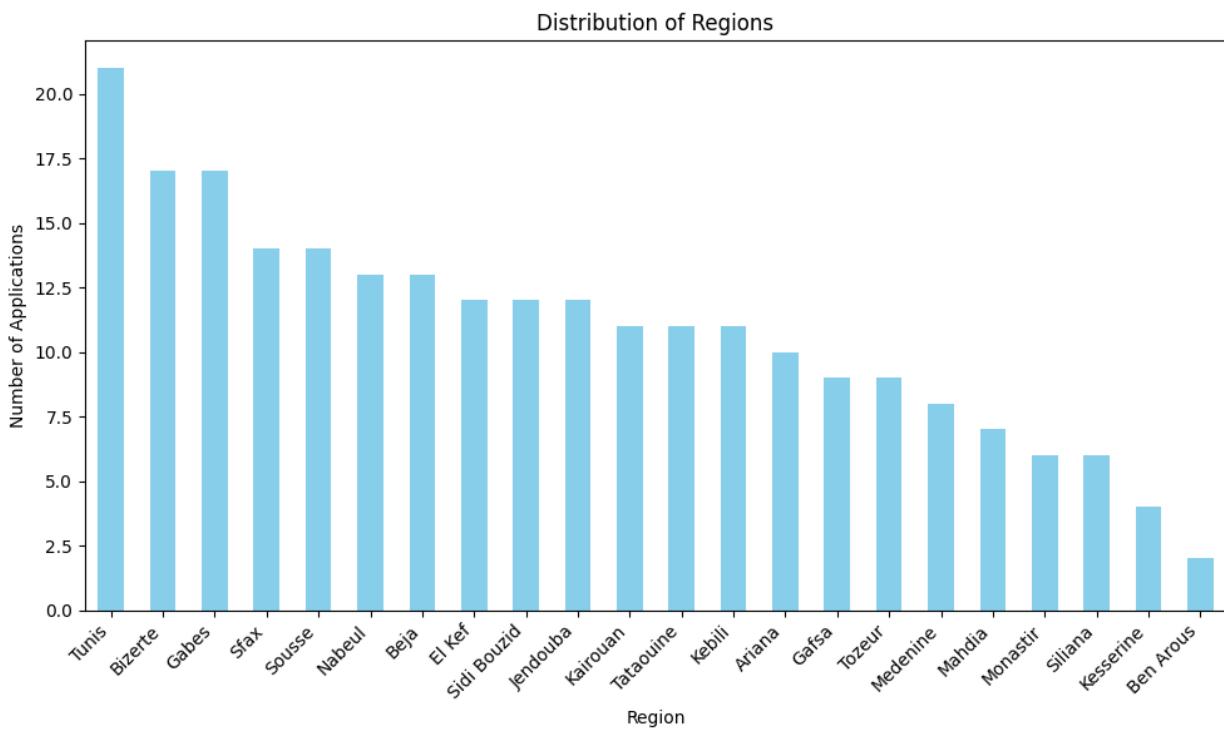


Figure 59– stats output 1

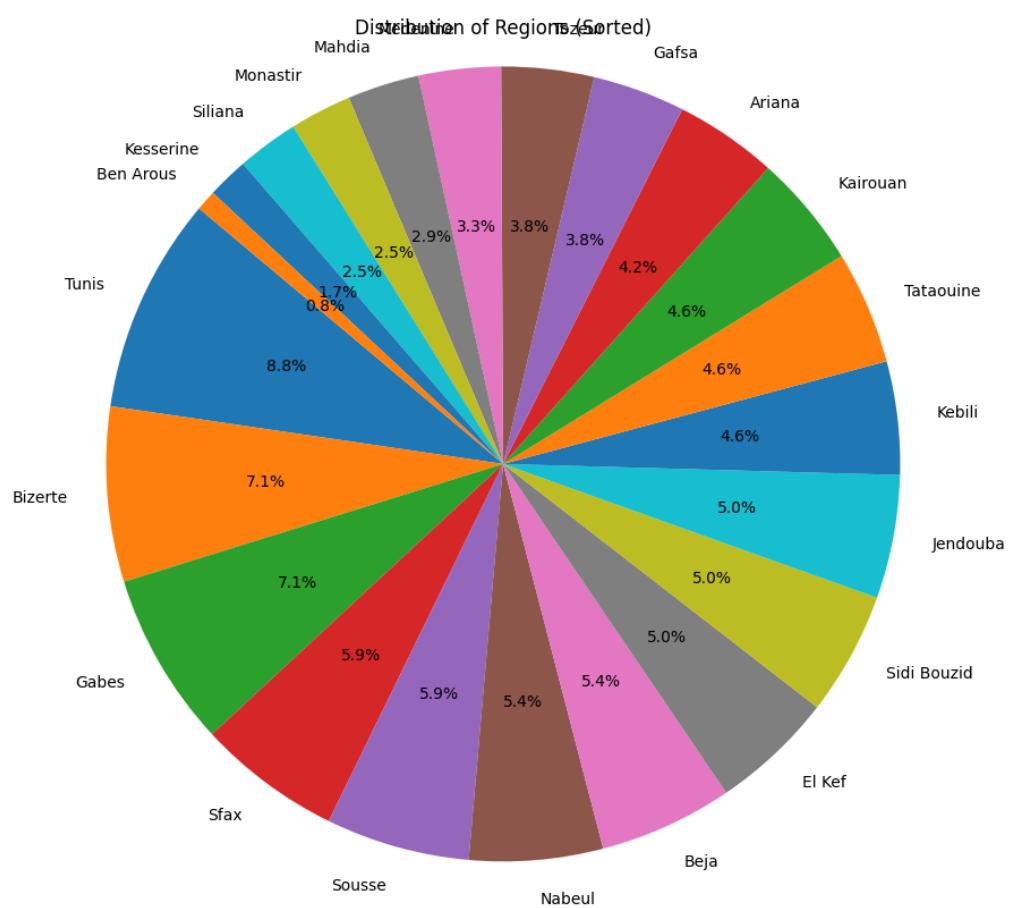


Figure 60– stats output 2

b. Classification - determining if a comment/answer is correct based on the training data Model:

Logistic Regression - a linear model that predicts the probability of a binary outcome (correct or incorrect in this case).

The screenshot shows a forum post titled "Internships" with a timestamp of "6/10/24, 1:28 AM". The post contains the following text:

```

All internships are unpaid. - 6/10/24, 1:01 AM - false
You must be a junior or senior undergraduate student. - 6/10/24, 9:37 PM - true
Internships can be combined with online courses, especially if the internship is remote. - 6/10/24, 9:44 PM - true
Students can apply for as many internships as they are qualified for. - 6/10/24, 9:44 PM - true
Anyone can apply, regardless of their academic level. - 6/10/24, 9:44 PM - false
  
```

Below the text are buttons for "Report Post", "Like", and "Unlike".

Figure 61– logistic regression output

- **Logistic Regression: in forum**

It works by establishing a relationship between your features (processed answer text) and the labels (correct or incorrect).

It calculates odds of a particular answer being correct based on the features.

A threshold is set to classify the answers - values above the threshold are considered correct, and vice versa.

Question ID	Question Text	Answer ID	Answer Text	Correct
1	What are the eligibility criteria for the summer internship program?	1	You must be a junior or senior undergraduate student.	0
1	What are the eligibility criteria for the summer internship program?	2	Anyone can apply, regardless of their academic level.	0
2	How many credits are required to apply for an internship?	3	Students need at least 60 credits to qualify.	0
2	How many credits are required to apply for an internship?	4	There are no credit requirements.	0
3	Are international students allowed to apply for internships?	5	International students can apply if they have a valid student visa.	0
3	Are international students allowed to apply for internships?	6	No, any local students are eligible.	0
4	What is the application deadline for the fall internship?	7	The deadline is June.	1
4	What is the application deadline for the fall internship?	8	Applications are accepted throughout the year.	0
5	Can first-year students apply for internships?	9	First-year students are not eligible for internships until their second year.	0
5	Can first-year students apply for internships?	10	First-year students are eligible for internships.	0
6	Do internships require a minimum GPA?	11	A minimum GPA of 3.0 is necessary to apply for internships.	1
6	Do internships require a minimum GPA?	12	GPA is not a criterion for Internship applications.	0
7	Are there internships available for part-time students?	13	Part-time students are eligible for certain internships depending on their programs.	1
7	Are there internships available for part-time students?	14	Internships are only available for full-time students.	0
8	Is prior work experience required for applying to internships?	15	Some internships may require prior related work experience.	1
8	Is prior work experience required for applying to internships?	16	Other internships require no prior work experience.	0
9	Are internships paid or unpaid?	17	Internships can be either paid or unpaid, depending on the employer.	1
9	Are internships paid or unpaid?	18	All internships are unpaid.	0
10	Do internships count towards academic credit?	19	Internships may count as academic credit with prior approval from the department.	1
10	Do internships count towards academic credit?	20	Internships may count as academic credit with prior approval from the university's internship portal.	0
11	Can international students do internships during the semester?	21	Students can register for internships through the university's internship portal.	1
11	Can international students do internships during the semester?	22	Students need to report any internships taken during the semester.	0
12	What types of internships are available for engineering students?	23	Engineering students can find internships in software, mechanical, civil, and electrical fields.	1
12	What types of internships are available for engineering students?	24	Engineering students are eligible for internships in any of these fields.	0
13	How long do typical internships last?	25	Most internships last between 2-6 months.	1
13	How long do typical internships last?	26	Some internships last up to a year.	0
14	What is the process to apply for an internship?	27	Students can apply for internships through the university's internship portal.	1
14	What is the process to apply for an internship?	28	Students need to apply directly to the company without university involvement.	0
15	Is there a deadline for reporting internship offers?	29	Students need to report any internships taken during the semester.	1
15	Is there a deadline for reporting internship offers?	30	Students need to report internships to the university at the end of their internship.	0
16	Are students required to complete an evaluation after their internship?	31	Students are required to complete an evaluation at the end of their internship.	1
16	Are students required to complete an evaluation after their internship?	32	Students are required to complete an evaluation at the end of their internship.	0
17	Can students do multiple internships during their degree?	33	Students are allowed to do multiple internships throughout their entire degree.	1
17	Can students do multiple internships during their degree?	34	Students are allowed to do multiple internships throughout their entire degree.	0
18	Do internships require departmental approval before application?	35	Departmental approval is not necessary for Internship applications.	0
18	Do internships require departmental approval before application?			

Figure 62– logistic regression dataset

- **Key Points:**

This is a supervised learning approach where the model learns from labeled data (correct/incorrect).

The TF-IDF vectorizer transforms textual data into numerical features suitable for the logistic regression model.

- classifying the correctness of comments/answers.

```
[24]: import joblib

# Load the pre-trained model and vectorizer
classifier = joblib.load('C:/Users/user/Downloads/classifier.pkl')
vectorizer = joblib.load('C:/Users/user/Downloads/vectorizer.pkl')

# Sample comment for testing
sample_comment = "Students need at least 60 credits to qualify."

# Transform the comment using the vectorizer
X_new = vectorizer.transform([sample_comment])

# Predict with the classifier
prediction = classifier.predict(X_new)
is_correct = prediction[0] == 1

# Print out the prediction result
print(f'Is the comment correct? {is_correct}')


Is the comment correct? True
```

Figure 63– Code 1

c. Classification Logistic Regression: in convention

```
[28]: loaded_model = joblib.load('C:/Users/user/Desktop/logistic_regression_model.joblib')

loaded_vectorizer = joblib.load('C:/Users/user/Desktop/tfidf_vectorizer.joblib')

def loaded_predict_text(text):
    # Transform the input text to TF-IDF using the Loaded vectorizer
    tfidf = loaded_vectorizer.transform([text])
    prediction = loaded_model.predict(tfidf)
    return "AI-generated" if prediction[0] == 1 else "Human-written"

example_text = "The code you provided utilizes libraries commonly found"
print(loaded_predict_text(example_text))

AI-generated
```

Figure 64– Code 2

- **TfidfVectorizer**: Converts text data into numerical features based on word importance within the documents (TF-IDF).
- **LogisticRegression**: Classification algorithm used to predict if a text is AI-generated or human-written based on the TF-IDF features.
- **SVC**: Support Vector Machine (SVM) - another classification algorithm, used here for comparison with Logistic Regression.

d. Classification: Extracting Statistics

Following the classification of feedback entries using a sentiment analysis model (e.g., Gaussian Naive Bayes), we extracted statistical data to understand the overall sentiment distribution. This analysis revealed that [number]% of the feedback entries were classified as positive, while [number]% were classified as negative. Further analysis of word count within the feedback showed that positive entries tended to have [average word count] words on average, compared to [average word count] words for negative entries.

The screenshot shows the Postman interface with the 'Body' tab selected. The request body is set to 'Text' and contains the following JSON:

```

1 {
2   "sentence": "bad experience"
3 }

```

Below the body, the status bar indicates 'Status: 200 OK' and 'Size: 260 B'. The 'Test Results' section shows a single entry labeled '1 Negative'.

Figure 65– test 1

The screenshot shows the Postman interface with the 'Body' tab selected. The request body is set to 'Text' and contains the following JSON:

```

1 {
2   "sentence": "best experience"
3 }

```

Below the body, the status bar indicates 'Status: 200 OK' and 'Size: 260 B'. The 'Test Results' section shows a single entry labeled '1 Positive'.

Figure 66– test 2

```
In [13]: from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=0)
classifier = GaussianNB()
classifier.fit(X_train, y_train)
```

Figure 67– code 3

The screenshot shows the Postman interface with the 'Body' tab selected. The request body is set to 'Text' and contains the JSON payload: {"sentence": "good work"}. The response status is 200 OK, time taken is 884 ms, and size is 260 B. The test results show one positive test case.

Figure 68– test 3

e. KNN: Domain prediction based on student skills:

I. Data Preparation:

The code loads a CSV file containing student data (skills and corresponding career domains). It separates skill features from labels (domain names).

Categorical domain labels are converted into numerical values using Label Encoding for the KNN model.

The data is split into training and testing sets for model training.

```
import pandas as pd
from sklearn.neighbors import KNeighborsClassifier
from sklearn.preprocessing import LabelEncoder
import random
from sklearn.model_selection import train_test_split
import json

def load_and_encode_data(data_path):

    skill_data = pd.read_csv(data_path)
    features = skill_data.drop(["domain", "name"], axis=1)
    X = features.copy()

    encoder = LabelEncoder()
    skill_data["domain"] = encoder.fit_transform(skill_data["domain"])
    y = skill_data["domain"]

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    return X_train, X_test, y_train, y_test, encoder
```

Figure

69– Code loading and encoding skills dataset

II. KNN Model Training and Evaluation:

A KNeighborsClassifier model is created with a specified number of neighbors (e.g., 80). This determines how many similar students the model considers for predictions.

The model is trained on the training data, learning the relationships between skills and career paths.

The trained model can be evaluated on the testing set to assess its performance (accuracy) in predicting career domains for unseen data.

```
def train_and_evaluate_model(X_train, X_test, y_train, y_test, encoder):  
  
    knn = KNeighborsClassifier(n_neighbors=80)  
    knn.fit(X_train, y_train)  
  
    y_pred = knn.predict(X_test)  
    predicted_domain_names = encoder.inverse_transform(y_pred)  
  
    return knn
```

Figure 70– Code training KNN model

III. Prediction for New Students:

The web application accepts new student data in JSON format through a POST request to the /predict endpoint.

This data likely represents a user's skillset.

The model predicts the most suitable career domains for the new student based on the trained model and encoded data.

The top three most probable domains are returned as the predicted results.

```
def predict_new_student(encoder, knn, new_student_data):  
  
    new_student_df = pd.DataFrame([new_student_data])  
    new_student_prediction = knn.predict(new_student_df)  
    predicted_domain_probabilities = knn.predict_proba(new_student_df)[0]  
  
    # Sort Domains and Probabilities Together by Descending Probability  
    sorted_predictions = sorted(zip(encoder.inverse_transform(range(len(predicted_domain_probabilities))), predicted_domain_probabilities))  
    top_three_domains = sorted_predictions[:3]  
    return top_three_domains
```

Figure 71– Code predicting domains for a new student

IV. Flask Integration:

Flask, a Python web framework, is used to create a web application.

Users can interact with the application through a web interface to provide their skillsets as input. The Flask app calls the main function to process the data, make predictions using the trained KNN model, and return the recommended career paths in JSON format.

```

def main(data):
    # Load data, encode, and split into training and testing sets
    data_path = "610.csv"
    X_train, X_test, y_train, y_test, encoder = load_and_encode_data(data_path)

    # Train the KNN model
    knn = train_and_evaluate_model(X_train, X_test, y_train, y_test, encoder)

    predicted_domains = predict_new_student(encoder, knn, data)
    print(f"Predicted Domain for the New Student: {predicted_domains}")

    domain_names = [domain for domain, _ in predicted_domains] # Extract only domain names
    return json.dumps(domain_names) # return top 3 domains as JSON

```

Figure 72– Code main function to use in Flusk endpoint

```

# Flask app setup
app = Flask(__name__)

@app.route("/predict", methods=["POST"])
def predict():
    data = request.get_json()
    if not data:
        return jsonify({"error": "Missing data in request body"}), 400
    return main(data)

if __name__ == "__main__":
    app.run(debug=True)

```

Figure 73– Code Flask app and endpoint

3. Conclusion

In wrapping up Sprint 3, we've advanced significantly in data analysis and prediction. Through strategic planning and execution, we've extracted valuable insights and laid the groundwork for informed decision-making. From formulating objectives to validating results, each step has propelled us towards our goal of enhancing project outcomes. Armed with newfound knowledge, we're ready to tackle future challenges with confidence. Let's maintain this momentum as we strive for excellence in data-driven decision-making.