

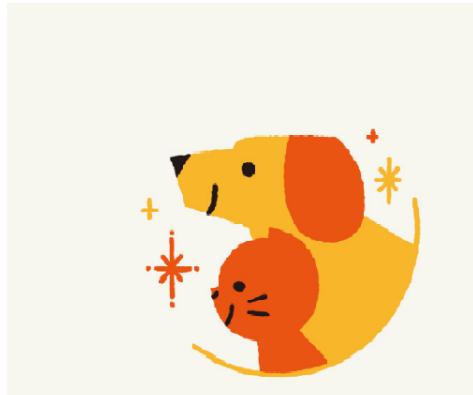
RAPPORT DE MINI PROJET

Présenté en vue de l'obtention de la
LICENCE EN SCIENCES DE L'INFORMATIQUE

Spécialité : Génie Logiciel et Systèmes d'Information

PetMatch

Par KHAMESSI TAHA , SAIDI MARAM



03/05/2023

Année Universitaire: 2022 - 2023

PetMatch

May 3, 2023

Contents

1 Specification des besoins	2
1.1 Introduction	2
1.2 Identification des besoins fonctionnels	2
1.3 Identification des cas d'utilisation	2
1.4 Identification des besoins non fonctionnels	3
1.5 Identification des acteurs	3
1.6 Diagramme de cas d'utilisation	4
1.7 Backlog de produit	5
1.8 Environnement de travail	5
1.8.1 La méthode SCRUM	5
1.8.2 Les outils logiciels	5
1.9 Conclusion	7
2 sprint 0	7
2.1 Introduction	7
2.2 Identification de back log de sprint 0	7
2.3 Raffinement du sprint 0	7
2.3.1 Raffinement de cas d'utilisation « S'authentifier » :	7
2.3.2 Raffinement de cas d'utilisation « Traiter l'ajout des animaux » :	8
2.3.3 Raffinement de cas d'utilisation « Enregistrer » :	9
2.4 Conception du sprint 0	10
2.4.1 Conception de cas d'utilisation « S'authentifier » :	11
2.4.2 Conception de cas d'utilisation « Traiter l'ajout des animaux » :	11
2.4.3 Conception de cas d'utilisation « Enregistrer » :	12
2.5 Réalisation du sprint 0 :	13
2.5.1 Interface du cas d'utilisation « S'authentifier » :	13
2.5.2 Interface du cas d'utilisation « Traiter l'ajout d'animal » :	14
2.5.3 Interface du cas d'utilisation « Enregistrer » :	15
2.6 Conclusion	15
3 Sprint 1	15
3.1 Introduction	15
3.2 Identification de back log de sprint 1	15
3.3 Raffinement du sprint 1	16
3.3.1 Raffinement de cas d'utilisation « MAJ» :	16
3.3.2 Raffinement de cas d'utilisation « Consulter » :	17
3.3.3 Raffinement de cas d'utilisation « Adopter » :	17
3.4 Conception du sprint 1	18
3.4.1 Conception de cas d'utilisation « MAJ» :	18
3.4.2 Conception de cas d'utilisation « consulter » :	19
3.4.3 Conception de cas d'utilisation « Adopter » :	20
3.5 Réalisation du sprint 1	21
3.5.1 Interface du cas d'utilisation « MAJ»	21
3.5.2 Interface du cas d'utilisation « Consulter »	21

3.5.3 Interface du cas d'utilisation « Adopter »	23
3.6 Conclusion	24

Introduction Générale

Le déplacement d'animaux domestiques dans notre société est une question importante qui concerne de nombreux propriétaires d'animaux de compagnie. Malheureusement, il y a de nombreuses situations où les propriétaires ne peuvent plus prendre soin de leur animal et sont obligés de chercher une nouvelle maison pour leur ami à fourrure. Les raisons du déplacement des animaux de compagnie peuvent être diverses, comme la perte d'un emploi, un déménagement dans un endroit où les animaux ne sont pas autorisés, des problèmes de santé ou même la mort du propriétaire. Les refuges pour animaux jouent un rôle important dans la réduction du nombre d'animaux domestiques déplacés . C'est dans ce contexte qu'on va représenter notre site web « PetMatch » qui permet essentiellement d'adopter et d'abandonner les animaux de compagnie (chats et chiens) afin d'aider à réduire ce phénomène. Pour la modélisation de notre projet, nous avons suivi la méthodologie de conception itérative et incrémentale la méthode SCRUM. Notre rapport est organisé de la manière suivante :

chapitre 1 : intitulé Spécification de besoins qui présente l'identification des besoins fonctionnel, non fonctionnels, les acteurs et les cas d'utilisation.

chapitre 2 : intitulé "sprint 0" est une description détaillée du premier release à étudier.

chapitre 3 : intitulé "sprint 1" est une description détaillée du deuxième release à étudier.

Enfin, nous synthétisons notre rapport par une conclusion générale et perspectives.

1 Specification des besoins

1.1 Introduction

Dans ce chapitre nous allons étudier les besoins et représenter les objectifs de notre projet en précisant les besoins fonctionnels et les besoins non fonctionnels d'une part, d'une deuxième part nous allons mentionner le diagramme de cas d'utilisation globale et le Backlog produit.

1.2 Identification des besoins fonctionnels

Un besoin fonctionnel est un besoin spécifiant une action qu'un système doit être capable d'effectuer, sans considérer aucune contrainte physique. C'est un besoin du point de vue de l'utilisateur. Notre projet consiste à répondre aux besoins suivants :

- Consulter;
- S'inscrire;
- S'authentifier;
- Enregistrer;
- Traiter l'ajout d'animal (accepter ou refuser);
- MAJ;
- Adopter.

1.3 Identification des cas d'utilisation

- **Consulter:** Notre site web permet à l'utilisateur de consulter les divers animaux avec leurs descriptions.
- **S'inscrire:** Dans ce stade là l'utilisateur est anonyme mais pour passer à un stade plus rigoureux, il faut qu'il s'inscrire comme "utilisateur".

- **S'authentifier:** Après la phase d'inscription l'utilisateur a le droit de s'authentifier et selon son choix il sera guidé comme un adopteur ou un abandon sinon il à un accés en tant qu'administrateur.
- **Enregistrer:** Après l'inscription l'abandon à le droit de poster(Ajouter) son animal à adopter sous forme d'une demande à traiter par l'admin.
- **Traiter l'ajout d'animal :** c'est le rôle de l'admin de refuser ou d'accepter les demandes des abandons afin d'être supprimées du l'interface de l'admin (en cas de refus) ou ajoutées à l'accueil (en cas d'acceptation).
- **MAJ:** C'est le rôle de l'admin de mettre à jour les animaux dans le fil d'actualité.
- **Adopter:** Après l'inscription l'adopteur a l'accès d'adopter des animaux.

1.4 Identification des besoins non fonctionnels

Les besoins non fonctionnels sont des indicateurs de qualité de l'exécution des besoins fonctionnels, importants car ils agissent de façon indirecte sur le résultat et sur le rendement de l'utilisateur, ce qui fait qu'ils ne doivent pas être négligés. Pour cela il faut répondre aux exigences suivantes :

- **L'aptitude à la maintenance et la réutilisation :** Le système doit être conforme à une architecture standard et claire permettant sa maintenance et sa réutilisation.
- **Fiabilité :** L'application doit fonctionner de façon cohérente sans erreur et doit être satisfaisante.
- **Sécurité :** Notre solution doit respecter surtout la confidentialité des données personnelles des utilisateurs qui reste l'une des contraintes les plus importantes dans les applications mobiles.

1.5 Identification des acteurs

Un acteur est une personne physique ou morale prenant part ou affectée par l'action ou le projet en question. Il faut donc commencer par bien préciser par rapport à quelle action ou suite d'actions on cherche à déterminer qui sont et ce que sont les acteurs. [3] Notre site web contient 3 acteurs qui agissent directement avec le système :

- L'adopteur : C'est un utilisateur qui peut accéder au site consulter et adopter les animaux.
- L'abandon : C'est un utilisateur qui peut accéder au site consulter et Enregistrer les animaux.
- L'administrateur : C'est le superviseur qui possède toutes les permissions de contrôle pour gérer les animaux.

ACTEUR	ROLE
Admin	<ul style="list-style-type: none"> • Traiter l'ajout d'animal • MAJ • S'authentifier
Abandon	<ul style="list-style-type: none"> • S'authentifier • S'inscrire • Enregistrer • Consulter
Adopteur	<ul style="list-style-type: none"> • S'authentifier • S'inscrire • Consulter • Adopter

1.6 Diagramme de cas d'utilisation

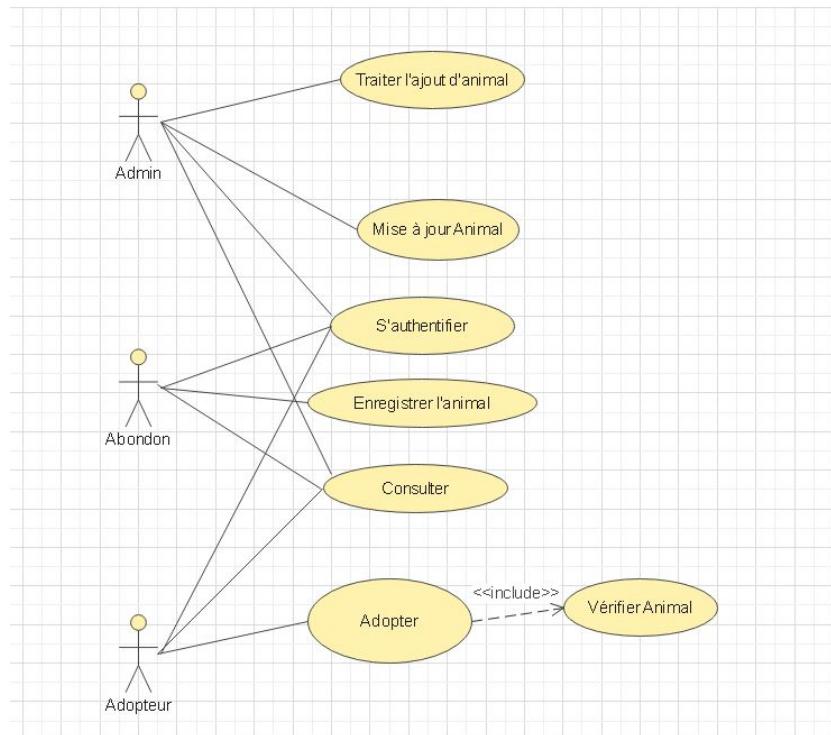


Figure 1: Diagramme de cas d'utilisation

1.7 Backlog de produit

Backlog de produit	Priorité	Estimation	Planification
En tant qu'admin, je peux traiter l'ajout d'animal.	1	Moyen	Sprint 0
En tant qu'admin, je peux m'authentifier.	1	Moyen	Sprint 0
En tant qu'adopteur, je peux m'authentifier.	1	Moyen	Sprint 0
En tant qu'abandon, je peux m'authentifier.	1	Moyen	Sprint 0
En tant qu'abandon, je peux enregistrer l'ajout d'animal	1	Moyen	Sprint 0
En tant qu'admin, je peux mettre à jours Animal.	2	Moyen	Sprint 1
En tant qu'abondon, je peux consulter Animal.	2	Moyen	Sprint 1
En tant qu'adopteur, je peux consulter Animal.	2	Moyen	Sprint 1
En tant qu'adopteur, je peux adopter Animal.	2	Moyen	Sprint 1

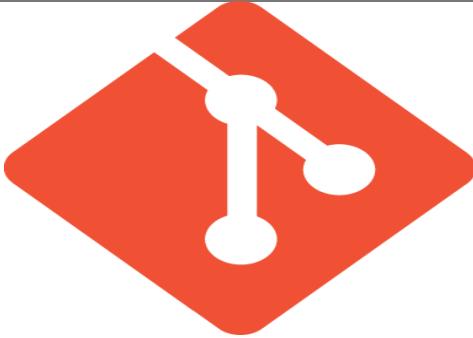
1.8 Environnement de travail

1.8.1 La méthode SCRUM

La méthode Scrum est une approche de gestion de projet agile et itérative qui se concentre sur la collaboration de l'équipe, la livraison régulière de produits fonctionnels et la réponse rapide aux changements. Elle est basée sur des principes tels que la transparence, l'inspection et l'adaptation. Le cadre Scrum est composé d'événements, de rôles et d'artefacts, qui travaillent ensemble pour fournir une structure claire et efficace pour la gestion de projet.

1.8.2 Les outils logiciels

Dans notre mini projet nous avons utiliser les logiciels suivants:

Logiciel	Definition
 <p>Visual Studio Code</p>	<p>Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS.</p>
	<p>Node.js est un environnement d'exécution JavaScript construit sur le moteur JavaScript V8 de Chrome.</p>
	<p>MySQL Workbench (anciennement MySQL administrator) est un logiciel de gestion et d'administration de bases de données MySQL créé en 2004. Via une interface graphique intuitive, il permet, entre autres, de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données</p>
	<p>Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre et gratuit, créé en 2005 par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2.</p>

1.9 Conclusion

Dans ce chapitre, nous avons fait un découpage fonctionnel détaillé de notre future solution par le biais de diagramme de cas d'utilisation et le backlog de produit.

2 sprint 0

2.1 Introduction

Dans ce chapitre, nous allons présenter le premier release du projet qui est : "sprint 0" qui permet de détailler les cas d'utilisation de priorité 1. L'étude de ce sprint comprend le raffinement, la conception, et la réalisation.

2.2 Identification de back log de sprint 0

Dans ce backlog du premier release que nous présentons ci-dessous contient une liste des éléments backlog qui devra être réalisé dans le sprint 0 :

Backlog de produit	Priorité	Estimation	Planification
En tant qu'admin, je peux traiter l'ajout d'animal.	1	Moyen	Sprint 0
En tant qu'admin, je peux m'authentifier.	1	Moyen	Sprint 0
En tant qu'adopteur, je peux m'authentifier.	1	Moyen	Sprint 0
En tant qu'abandon, je peux m'authentifier .	1	Moyen	Sprint 0
En tant qu'abandon, je peux enregistrer l'ajout d'animal	1	Moyen	Sprint 0

2.3 Raffinement du sprint 0

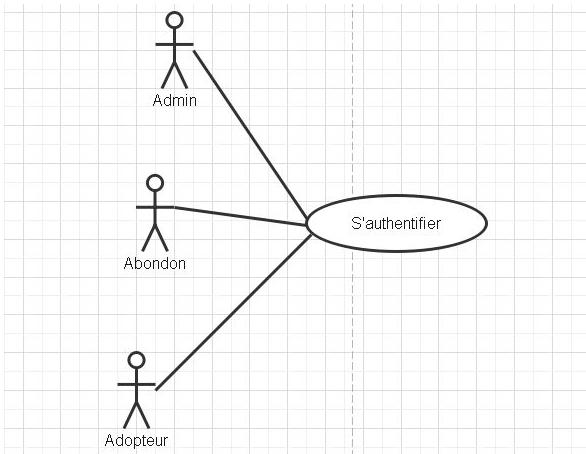
Dans cette partie, nous nous intéressons aux cas d'utilisation suivants :

- Traiter l'ajout d'animal (accepter ou refuser).
- S'authentifier.
- Enregistrer.

2.3.1 Raffinement de cas d'utilisation « S'authentifier » :

L'authentification est le besoin primordial pour le traitement et la sécurité des autres cas d'utilisation. Pour que les acteurs puissent exécuter leurs propres besoins, ils sont obligés de passer par l'authentification.

La figure 2.3.1 nous illustre le diagramme de cas d'utilisation « S'authentifier » :

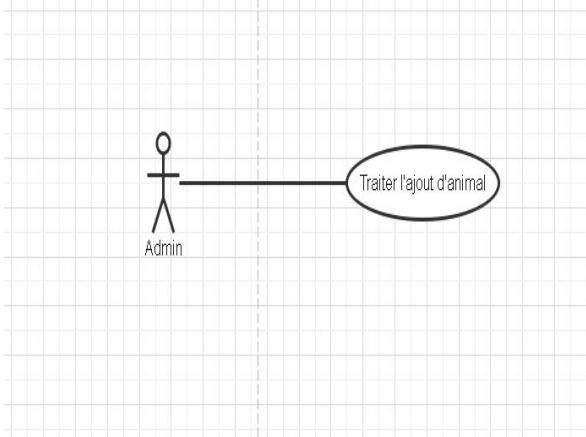


le tableau 2.3.1 présente le raffinement de cas d'utilisation "s'authentifier":

Cas d'utilisation	S'authentifier
Acteur	Admin, Abandon, Adopteur
Pré-condition	système est en marche
Post-Condition	Utilisateur (Admin, Abandon , Adopteur) authentifié
Description de scénario principal	<ul style="list-style-type: none"> Le système affiche l'interface de l'authentification L'utilisateur saisit son address mail et son mot de passe. L'utilisateur clique sur le bouton "Login". Le système vérifie le couple address (ou nom d'utilisateur) et mot de passe. Le système passe à la page d'accueil de l'application.
Scénario d'exception	Le système affiche un message d'erreur si les données sont erronées

2.3.2 Raffinement de cas d'utilisation « Traiter l'ajout des animaux » :

La figure 2.3.2 nous illustre le diagramme de cas d'utilisation « Traiter l'ajout d'animal» :

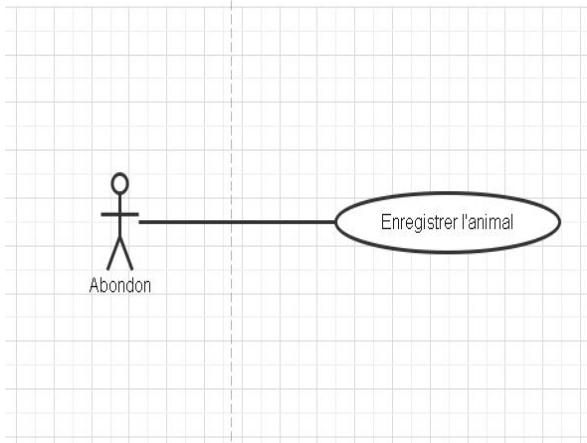


Le tableau 2.3.2 présente le raffinement de cas d'utilisation « Traiter l'ajout des animaux » :

Cas d'utilisation	Traiter l'ajout des animaux
Acteur	Admin
Pré-condition	Admin doit etre authentifié
Post-Condition	Demande d'ajout Traité
Description de scénario principal	<ul style="list-style-type: none"> Le système affiche l'interface de traitement d'ajout L'admin traite la demande d'ajout. L'admin clique sur le bouton "Accepte" ou "Reject". Le système vérifie la reponse d'admin et l'enregistrer. Le système recharge la page d'admin.
Scénario d'exception	le système affiche l'échec de traitement par un message d'erreur.

2.3.3 Raffinement de cas d'utilisation « Enregistrer » :

La figure 2.3.3 nous illustre le diagramme de cas d'utilisation « Enregistrer » :



Le tableau 2.3.2 présente le raffinement de cas d'utilisation « Enregistrer » :

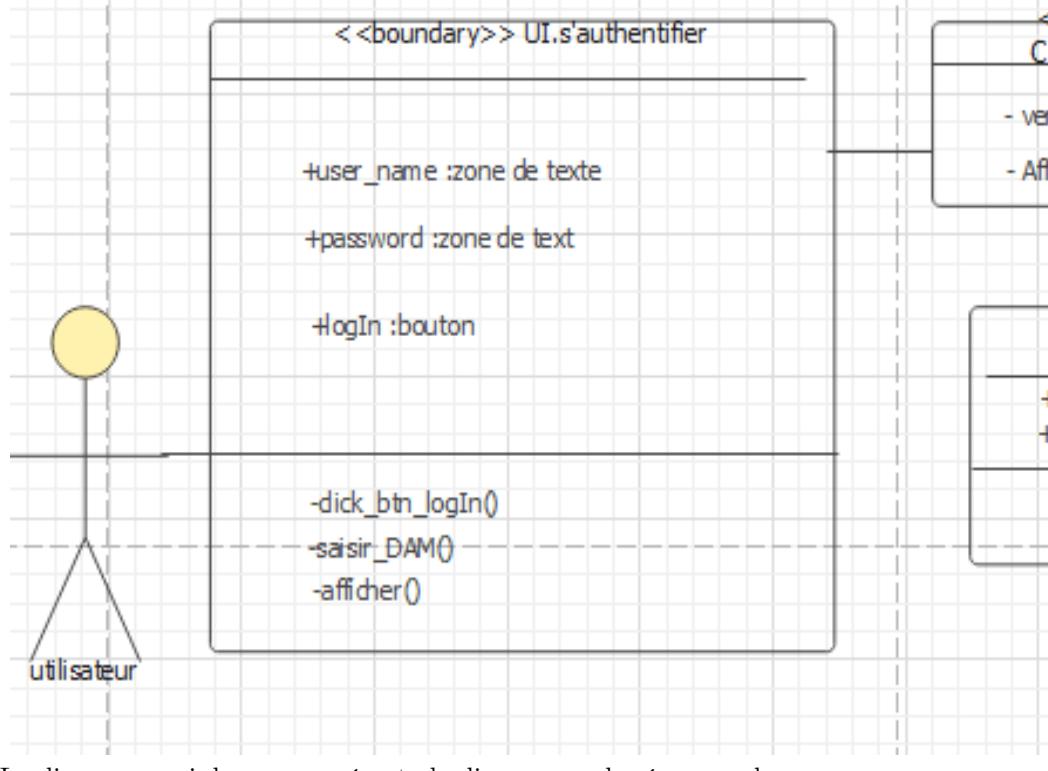
Cas d'utilisation	Enregistrer
Acteur	Abandon
Pré-condition	Abandon doit etre authentifié
Post-Condition	Demande d'ajout enregistré
Description de scénario principal	<ul style="list-style-type: none"> • Le système affiche la page "Accueil" • L'abandon clique sur le bouton "Abandonner". • Le système affiche la formulaire d'enregistrement • L'abandon choisit s'il va enregistrer un chat ou un chien, son sexe, sa race et son age et saisit son nom, couleur, poid et une description(si nécessaire). • L'abandon clique sur le bouton "Send". • Le système enregistre la demande. • Le système affiche la page d'accueil.
Scénario d'exception	le système affiche l'échec d'enregistrement par un message d'erreur.

2.4 Conception du sprint 0

La conception est une activité significative pour comprendre le développement d'un système afin de le rendre plus fiable et fidèle aux besoins du client. Nous allons donc spécifier notre système avant de le réaliser.

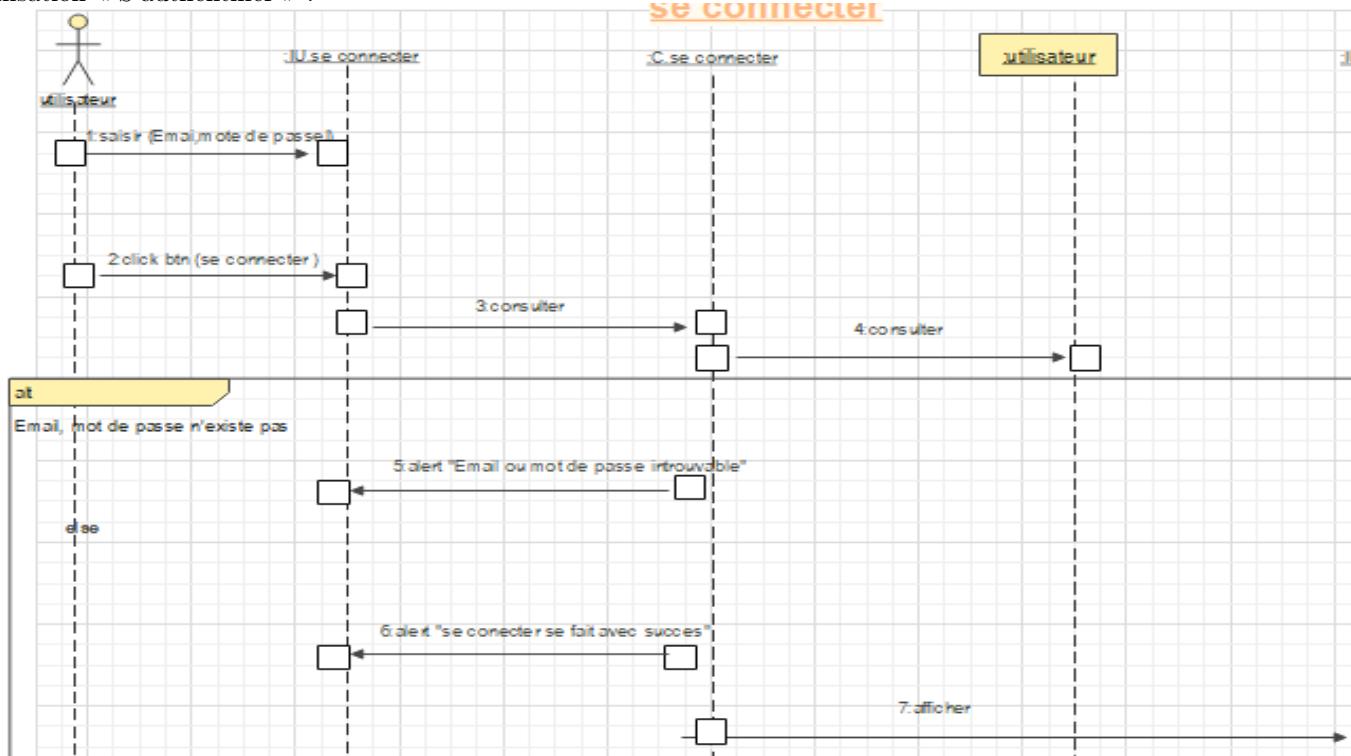
2.4.1 Conception de cas d'utilisation « S'authentifier » :

Diagramme de classes Le diagramme ci-dessous représente le Diagramme de classes relatives au cas d'utilisation « S'authentifier » :



d'utilisation « S'authentifier » :

Diagramme de séquence : Le diagramme ci-dessous représente le diagramme de séquence du cas d'utilisation « S'authentifier » :



2.4.2 Conception de cas d'utilisation « Traiter l'ajout des animaux » :

Diagramme de classes Le diagramme ci-dessous représente le Diagramme de classes relatives au cas d'utilisation « Traiter l'ajout des animaux » :

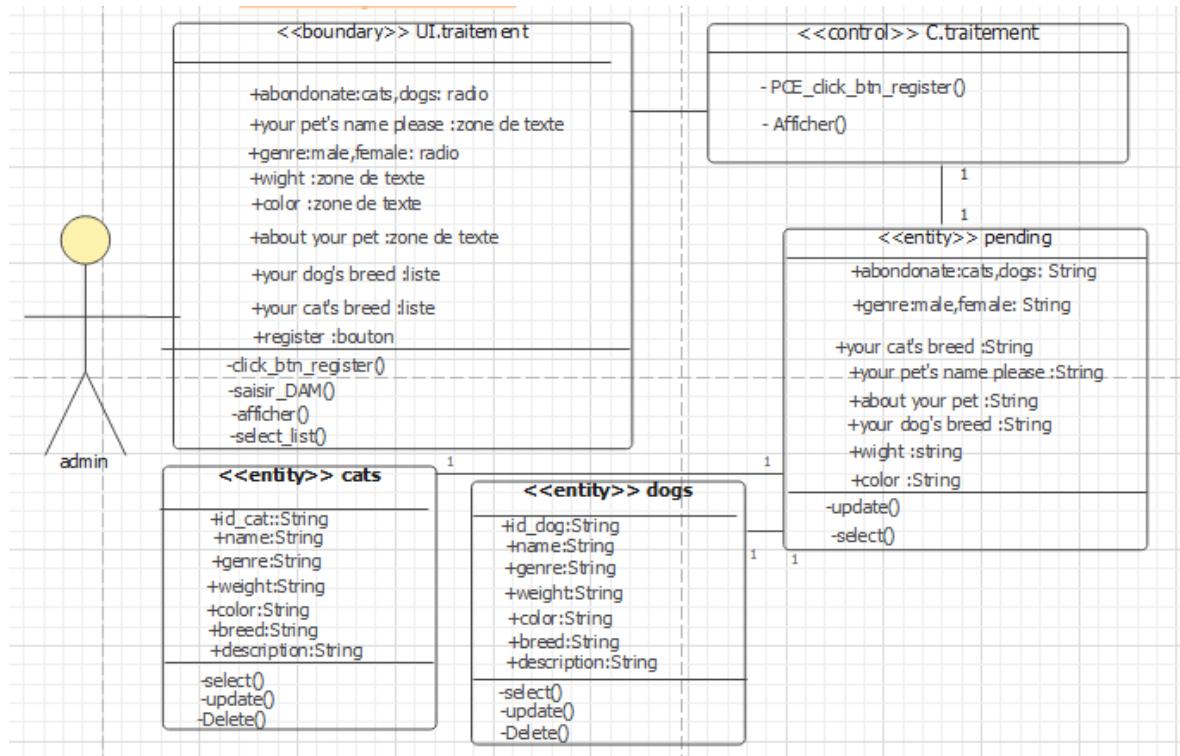
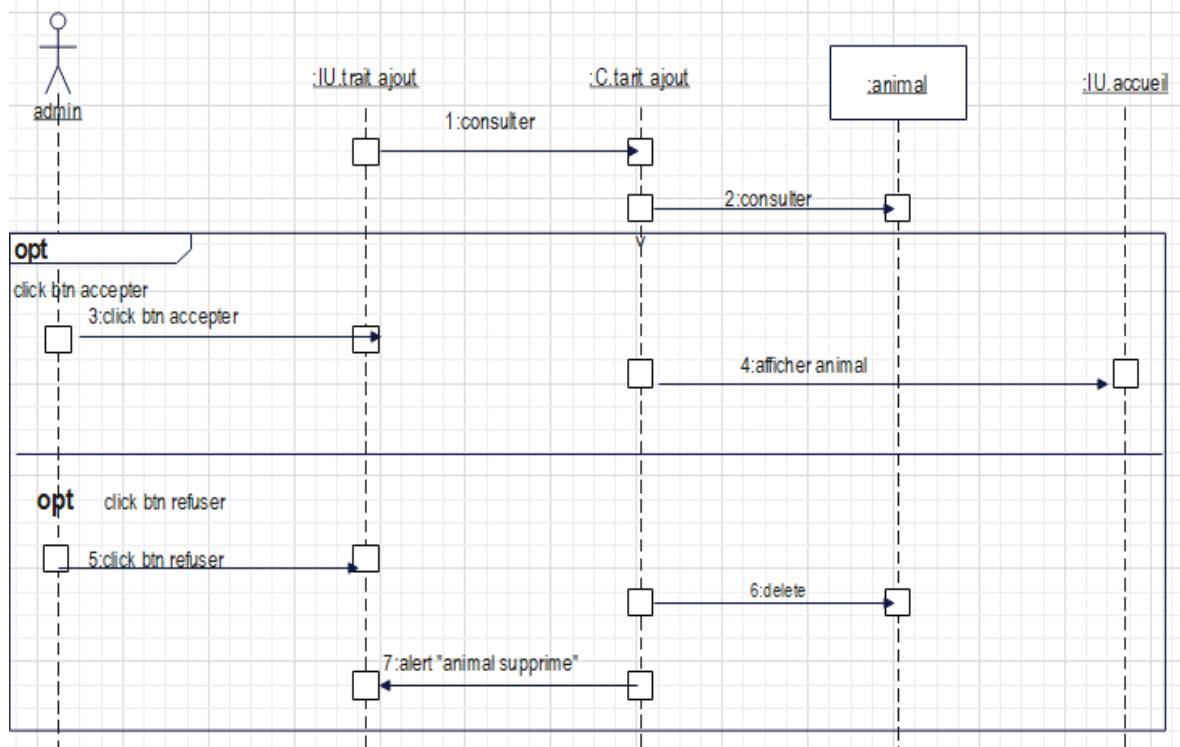


Diagramme de séquence : Le diagramme ci-dessous représente le diagramme de séquence du cas d'utilisation « Traiter l'ajout des animaux » :



2.4.3 Conception de cas d'utilisation « Enregistrer » :

diagramme de classes Le diagramme ci-dessous représente le Diagramme de classes relatives au cas d'utilisation « Enregistrer » :

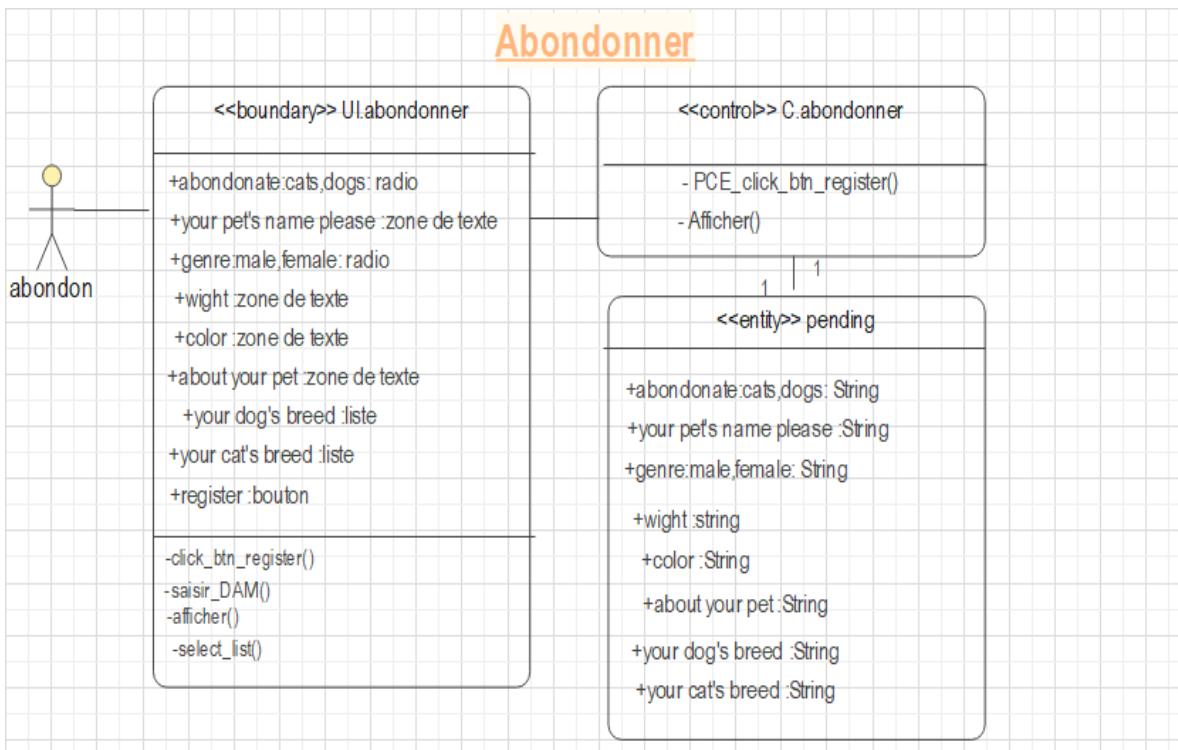
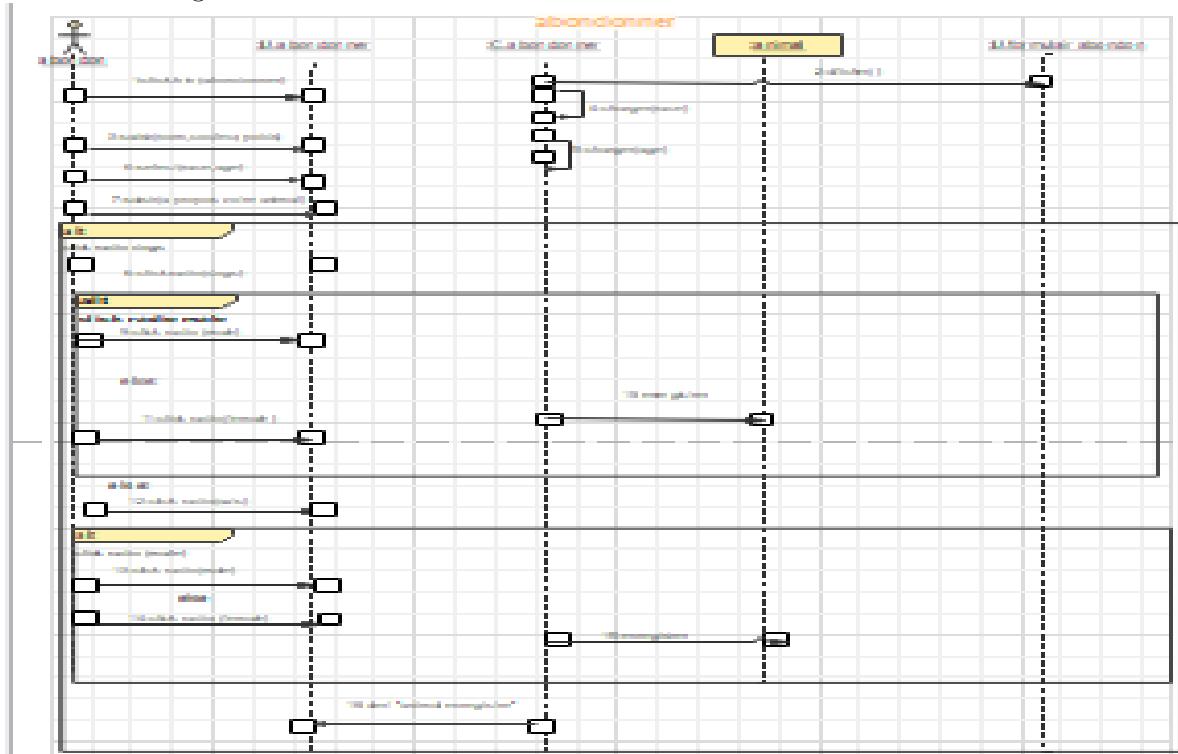


Diagramme de séquence : Le diagramme ci-dessous représente le diagramme de séquence du cas d'utilisation « Enregistrer » :

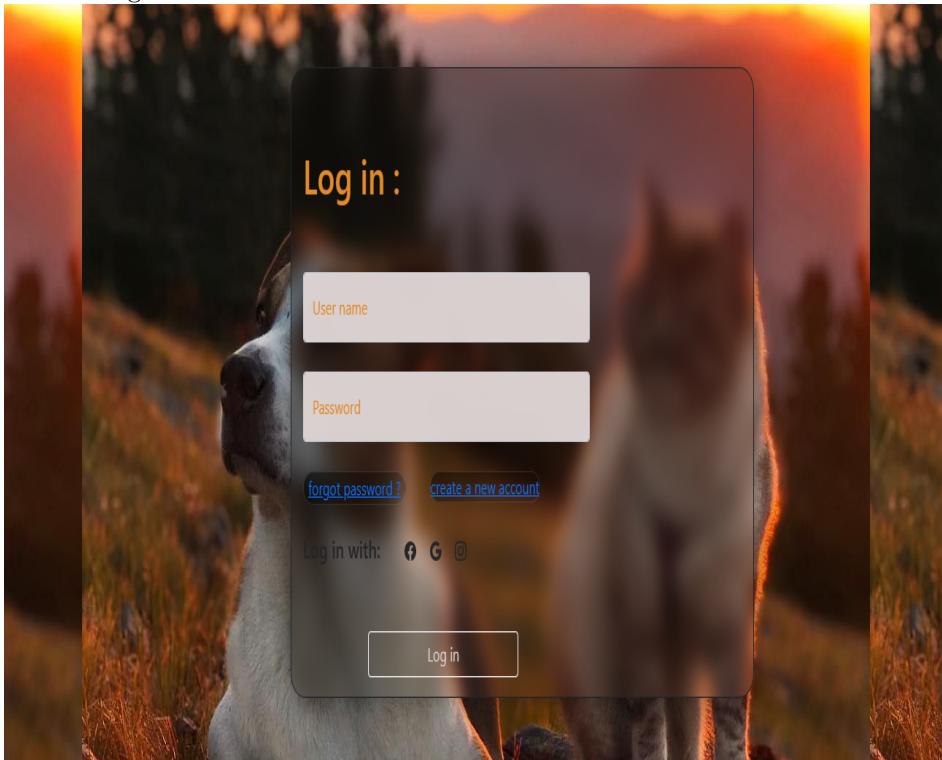


2.5 Réalisation du sprint 0 :

2.5.1 Interface du cas d'utilisation « S'authentifier » :

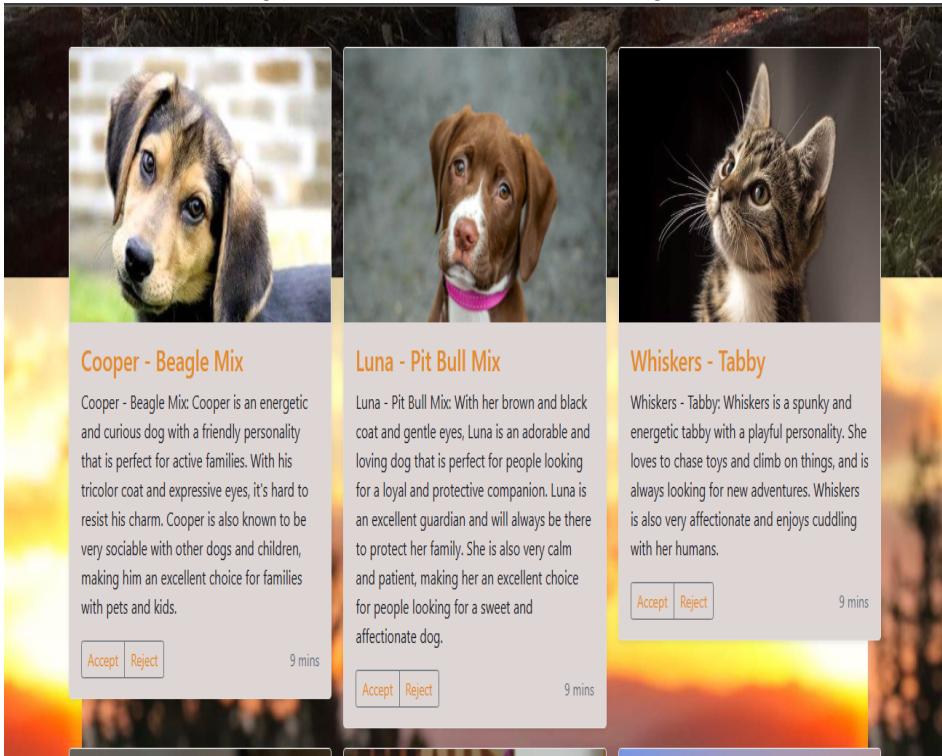
Ceci l'interface d'authentification, l'utilisateur (client / Concessionnaire) saisie son email ou son identifiant et son mot de passe et clique sur le bouton «Se Connecter» pour accéder à son compte crée précédemment, comme

le montre la figure :



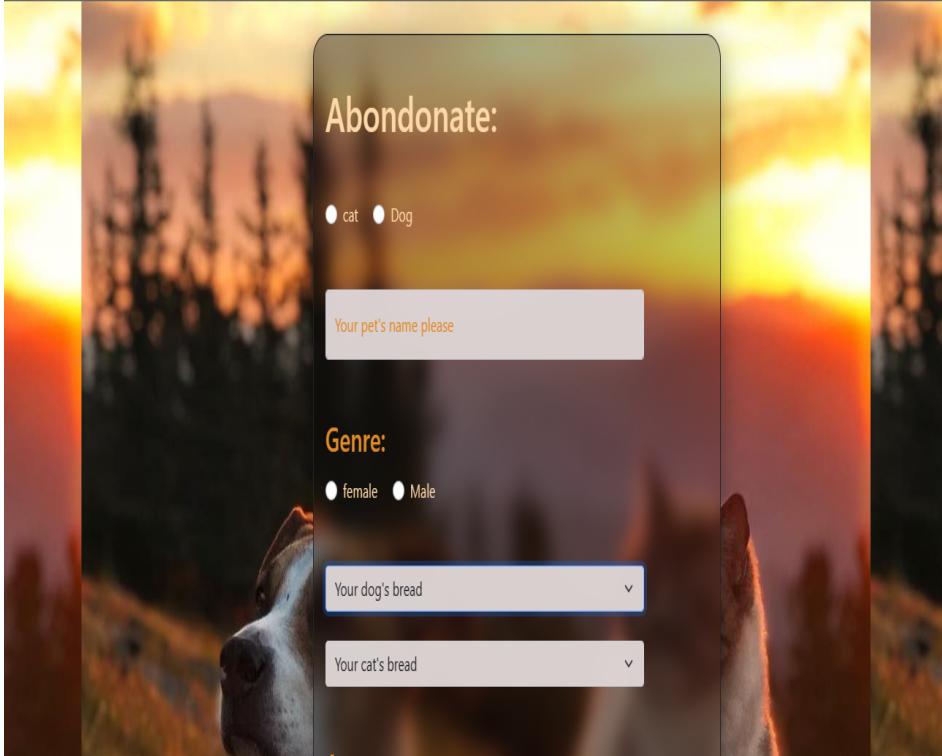
2.5.2 Interface du cas d'utilisation « Traiter l'ajout d'animal » :

Ceci l'interface d'enregistrement d'un animal , l'admin clique sur le bouton "Accept" ou "Reject" pour traiter la demande d'enregistrement ,comme le montre la figure :



2.5.3 Interface du cas d'utilisation « Enregister » :

Ceci l'interface d'abandonner un animal , l'utilisateur clique sur le bouton "Register" pour abandonner l'animal choisi, ,comme le montre la figure :



2.6 Conclusion

Dans ce chapitre, nous avons réalisé les cas d'utilisation de Sprint 0 par la suite nous avons présenté quelques interfaces ainsi que leurs analyses. Dans le chapitre suivant, nous présentons le 2ème Sprint consacré au "sprint 1".

3 Sprint 1

3.1 Introduction

Dans ce chapitre, nous allons présenter le dixième release du projet qui est : "sprint 1" qui permet de détailler les cas d'utilisation de priorité 2. L'étude de ce sprint comprend le raffinement, la conception, et la réalisation.

3.2 Identification de back log de sprint 1

Dans ce backlog du dixième release que nous présentons ci-dessous contient une liste des éléments backlog qui devra être réalisé dans le sprint 1 :

Backlog de produit	Priorité	Estimation	Planification
En tant qu'admin, je peux mettre à jours Animal.	2	Moyen	Sprint 1
En tant qu'abondon, je peux consulter Animal.	2	Moyen	Sprint 1
En tant qu'adopteur, je peux consulter Animal.	2	Moyen	Sprint 1
En tant qu'adopteur, je peux adopter Animal.	2	Moyen	Sprint 1

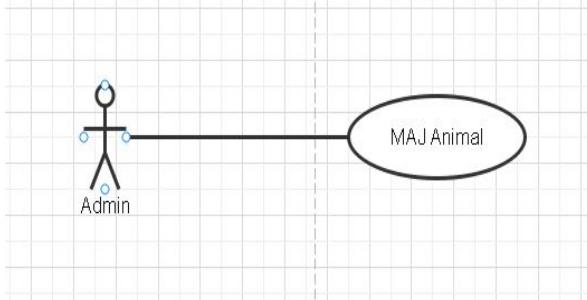
3.3 Raffinement du sprint 1

Dans cette partie, nous nous intéressons aux cas d'utilisation suivants :

- MAJ.
- Consulter.
- Adopter.

3.3.1 Raffinement de cas d'utilisation « MAJ » :

La figure 3.1 nous illustre le diagramme de cas d'utilisation « MAJ » :

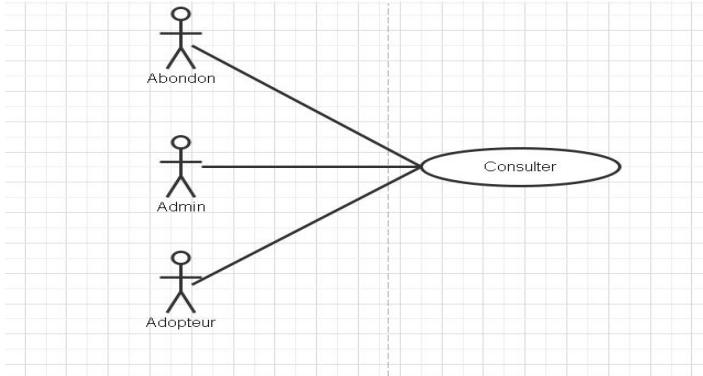


le tableau 3.1 presente le raffinement de cas d'utilisation "MAJ":

Cas d'utilisation	MAJ
Acteur	Admin
Pré-condition	Animal non mis à jours
Post-Condition	Animal mis à jours
Description de scénario principal	<ul style="list-style-type: none"> Le système affiche l'interface d'admin L'admin traite les demande d'ajout et d'adoption . Le système vérifie la reponse d'admin et mettre à jours la table d'animaux. Le système recharge la page d'admin.
Scénario d'exception	Le système affiche un message d'erreur si les données sont erronées

3.3.2 Raffinement de cas d'utilisation « Consulter » :

La figure 3.2 nous illustre le diagramme de cas d'utilisation « Consulter » :

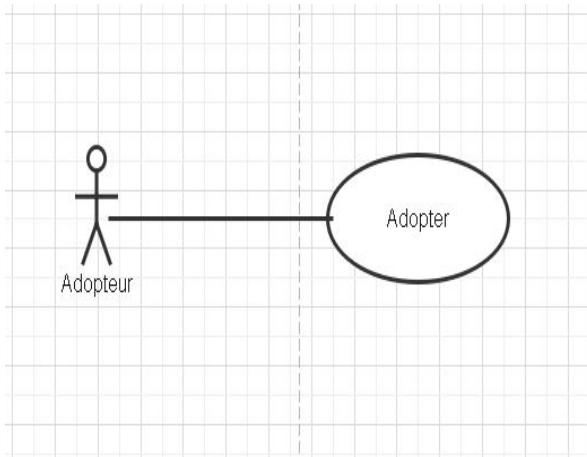


Le tableau 3.2 présente le raffinement de cas d'utilisation « Consulter » :

Cas d'utilisation	Consulter
Acteur	Admin, Abandon, Adopteur
Pré-condition	Animal non consulté
Post-Condition	Animal consulté
Description de scénario principal	<ul style="list-style-type: none"> Le système affiche l'interface d'accueil L'acteur consulte l'album des animaux . L'acteur clique sur le bouton "Cats" ou "Dogs" . Le système affiche l'interface demandé.
Scénario d'exception	Le système affiche un message d'erreur si les données sont erronées

3.3.3 Raffinement de cas d'utilisation « Adopter » :

La figure 2.3.4 nous illustre le diagramme de cas d'utilisation « Adopter» :



Le tableau 2.3.4 présente le raffinement de cas d'utilisation « Adopter » :

Cas d'utilisation	Adopter
Acteur	Adopteur
Pré-condition	Adopteur doit etre authentifié
Post-Condition	Demande d'adopter enregistré
Description de scénario principal	<ul style="list-style-type: none"> • Le système affiche la page "Accueil". • L'adopteur clique sur le bouton "Cats" ou "Dogs". • Le système affiche un catalogue des chats ou des chiens selon son choix. • L'adopteur choisit un animal. • L'adopteur clique sur le bouton "Adopter". • Le système affiche une formulaire d'adoption. • l'adopteur saisit son nom, prénom, ID D'animal, son adresse et choisir une date de rendez-vous puis il clique sur le bouton "Envoyer". • le système enregistre l'adoption. • Le système affiche la page d'accueil.
Scénario d'exception	le système affiche l'échec d'enregistrement ou du date du rendez-vous (si elle est réservée déjà) par un message d'erreur.

3.4 Conception du sprint 1

La conception est une activité significative pour comprendre le développement d'un système afin de le rendre plus fiable et fidèle aux besoins du client. Nous allons donc spécifier notre système avant de le réaliser.

3.4.1 Conception de cas d'utilisation « MAJ » :

Diagramme de classes: Le diagramme ci-dessous représente le Diagramme de classe relatives au cas d'utilisation « MAJ » :

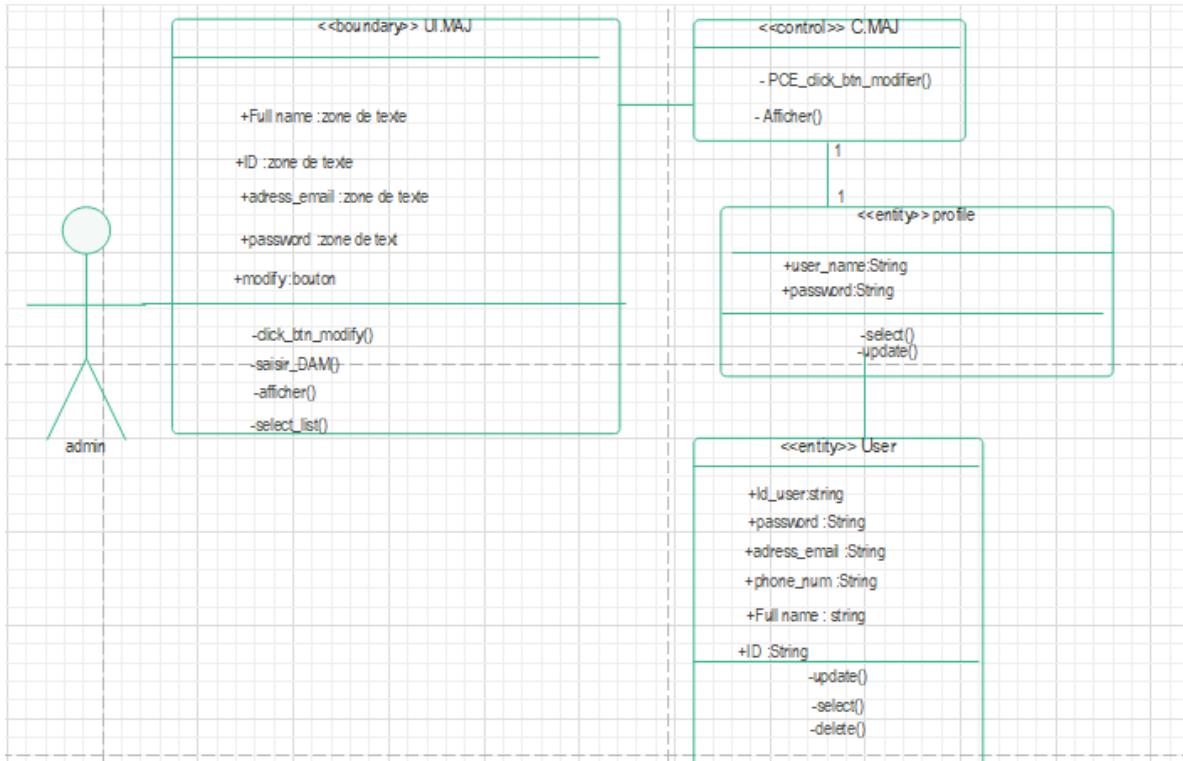
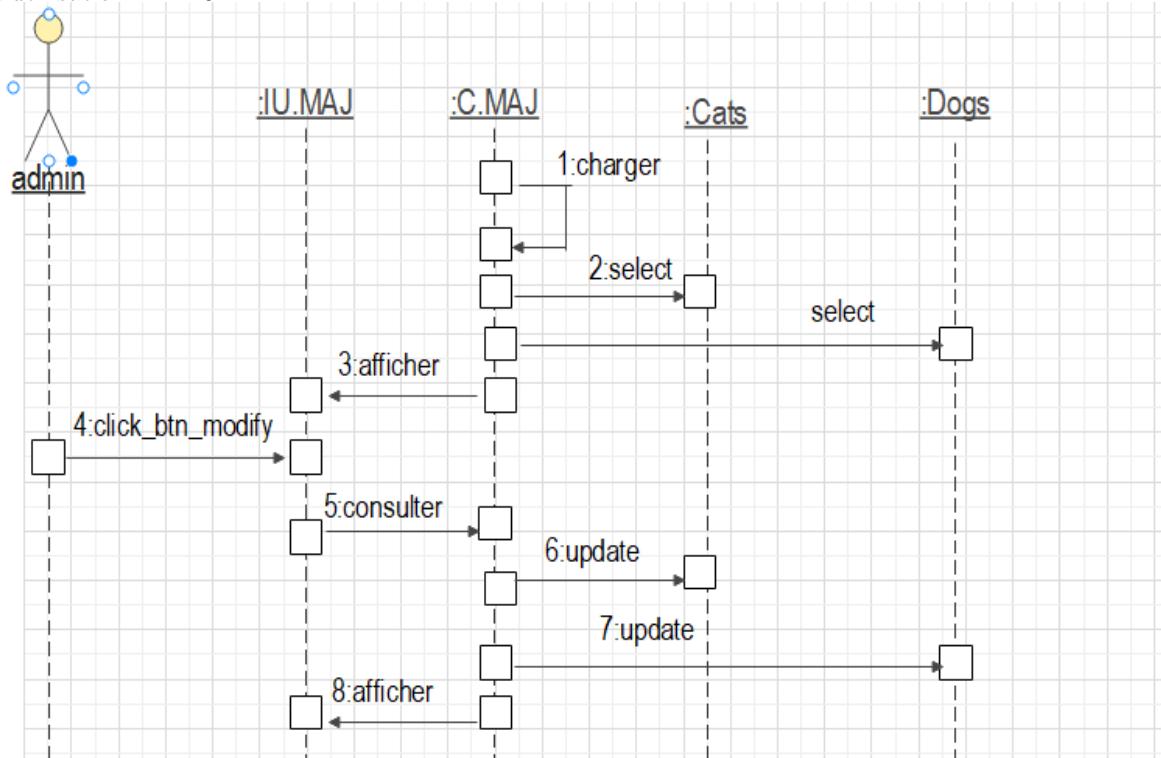


Diagramme de séquence : Le diagramme ci-dessous représente le diagramme de séquence du cas d'utilisation « MAJ » :



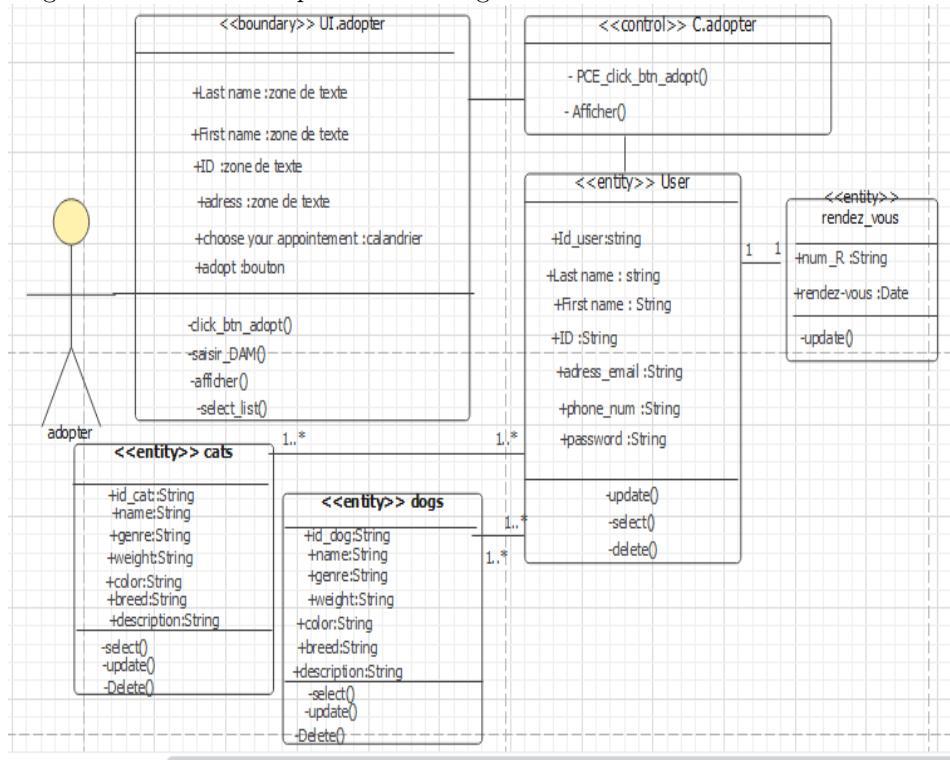
3.4.2 Conception de cas d'utilisation « consulter » :

Diagramme de classes Le diagramme ci-dessous représente le Diagramme de classe relatives au cas d'utilisation « consulter » :

Diagramme de séquence Le diagramme ci-dessous représente le diagramme de séquence du cas d'utilisation « consulter » :

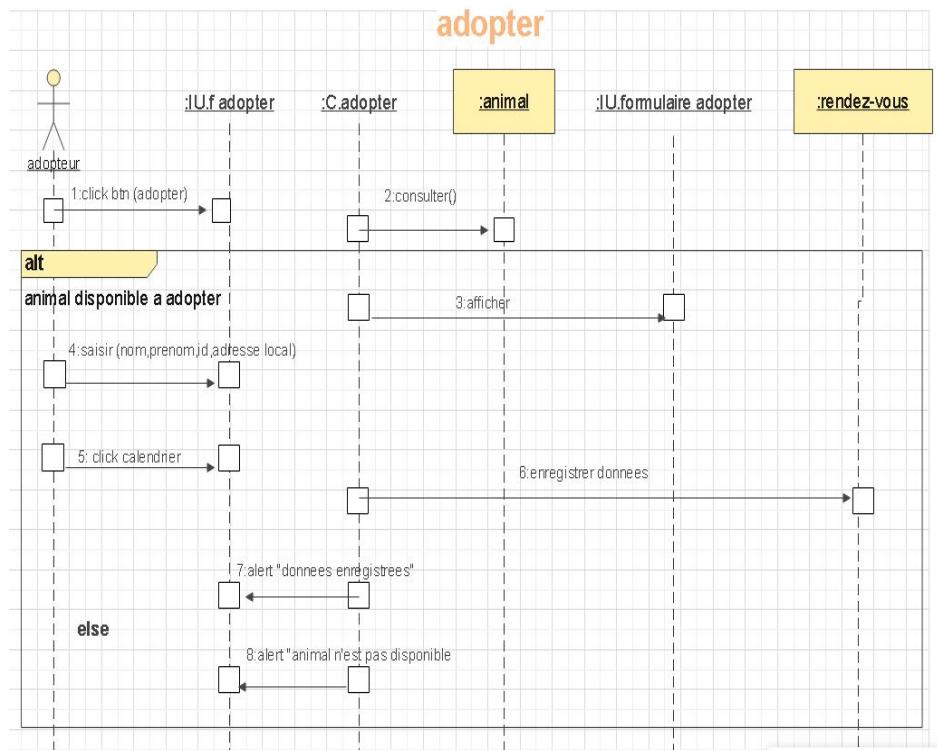
3.4.3 Conception de cas d'utilisation « Adopter » :

Diagramme de classe Le diagramme ci-dessous représente le Diagramme de classe relatives au cas d'utilisation « Adopter » :



d'utilisation « Adopter » :

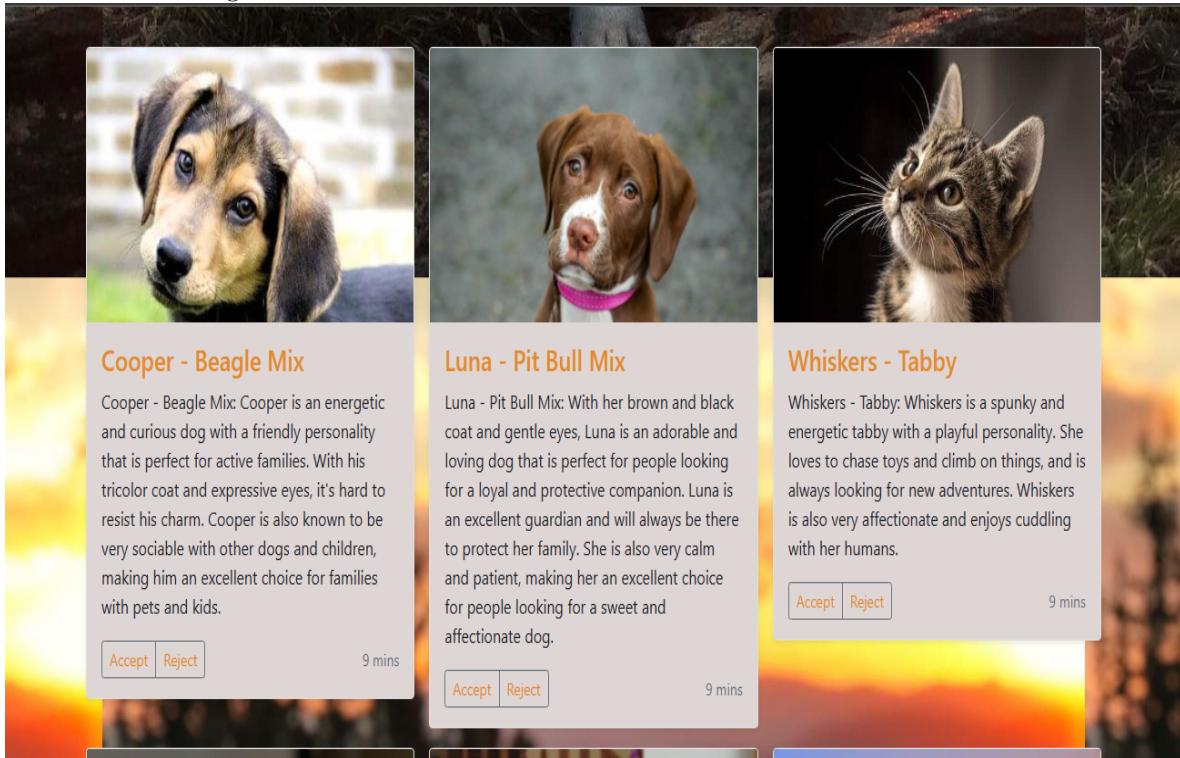
Diagramme de séquence Le diagramme ci-dessous représente le diagramme de séquence du cas d'utilisation « Adopter » :



3.5 Réalisation du sprint 1

3.5.1 Interface du cas d'utilisation « MAJ »

Ceci l'interface de MAJ, l'admin clique sur le bouton «modify» pour mettre à jour les demandes ,comme le montre la figure :



3.5.2 Interface du cas d'utilisation « Consulter »

Ceci l'interface de consultr les animaux , l'utilisateur défile l'interface pour choisir un animal soit a adopter pour l'adopteur ,soit a traiter l'ajout d'animal pour l'admin ,comme le montre la figure :

"Welcome to our pet adoption site 'PetMatch', where you can find your new best friend and give them a loving forever home."

"PetMatch is where we are committed to helping pets in need find their forever homes. Our goal is to connect you with the perfect companion who will bring joy, love, and companionship into your life. By adopting a pet, you not only gain a new family member, but you also give a second chance to an animal in need. Thank you for considering adoption and for joining us in our mission to make a difference in the lives of these wonderful creatures."



Simba - Maine Coon

"Simba - Maine Coon: Calm, affectionate, and sociable, the Maine Coon is an excellent choice for families."

[adopter](#)

9 mins



Luna - Siamois

"Luna - Siamese: Intelligent, affectionate, and vocal, the Siamese is an excellent choice for people who love cats with a strong personality."

[adopter](#)

9 mins

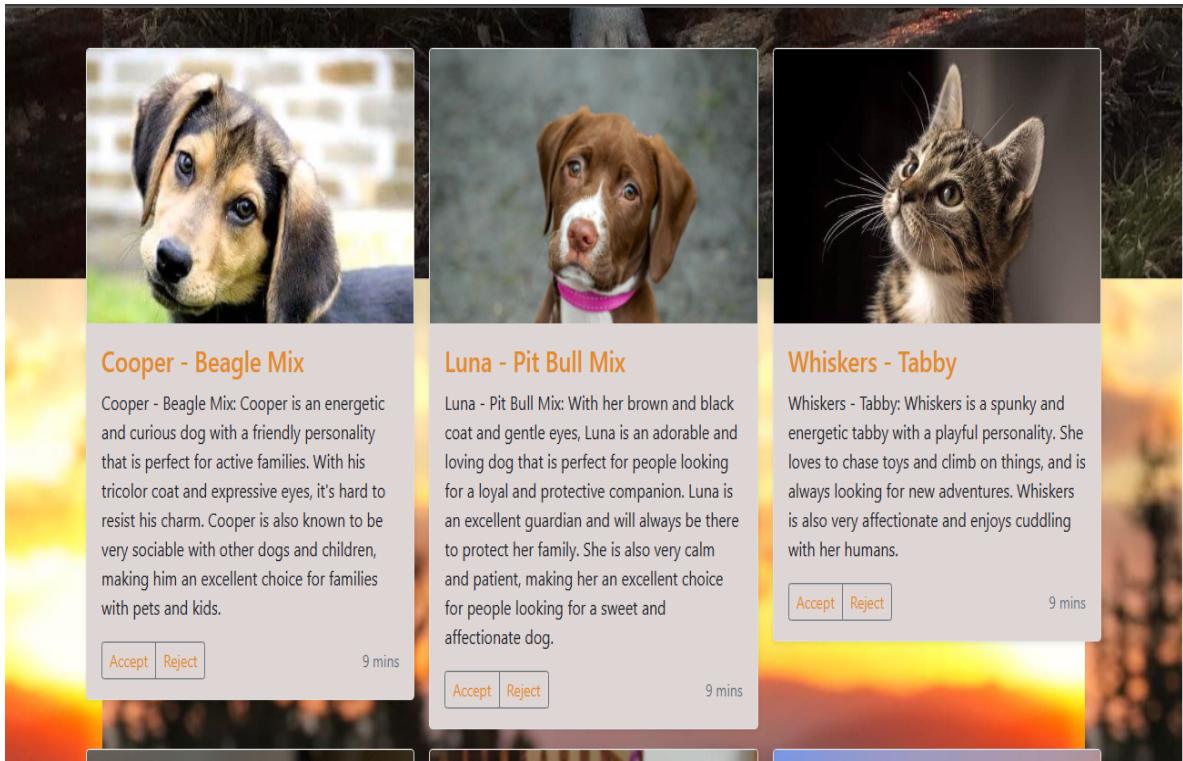


Oliver - British Shorthair

"Oliver - British Shorthair: Calm, affectionate, and easygoing, the British Shorthair is an excellent choice for people looking for a companion cat."

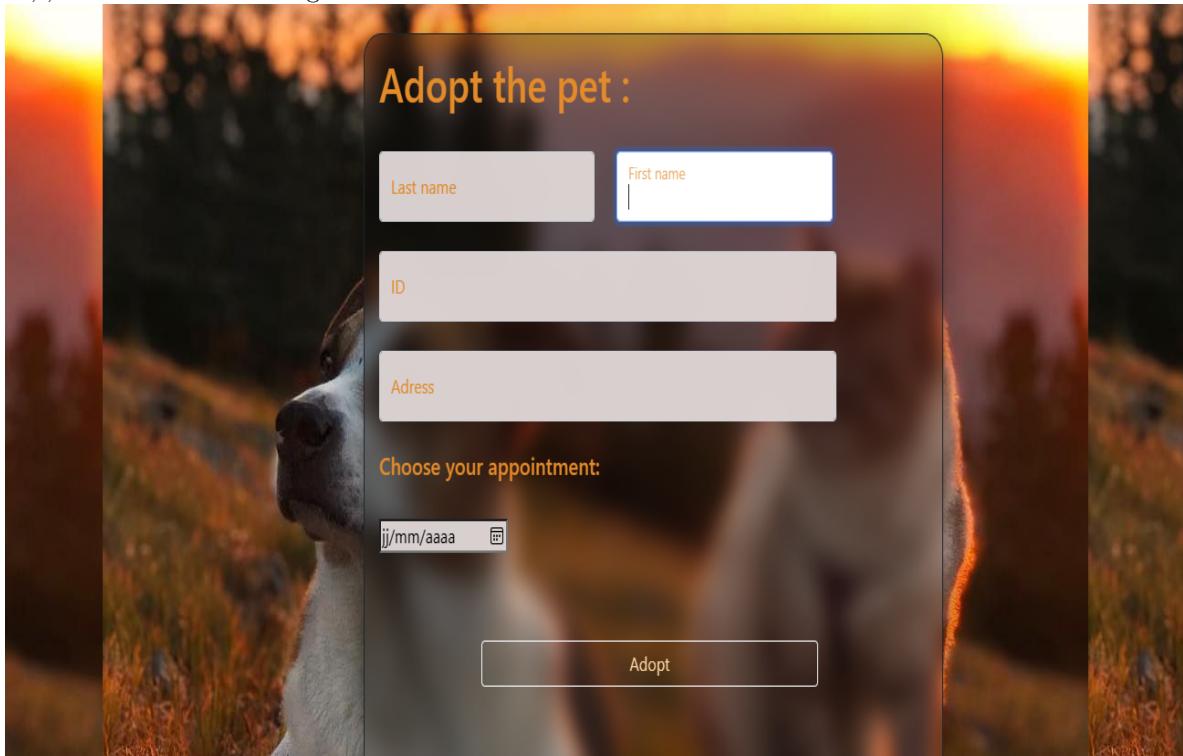
[adopter](#)

9 mins



3.5.3 Interface du cas d'utilisation « Adopter »

Ceci l'interface d'adopter un animal , l'utilisateur clique sur le bouton "Adopter" pour adopter l'animal choisi , comme le montre la figure :



3.6 Conclusion

Dans ce chapitre, nous avons réalisé les cas d'utilisation de sprint 1 par la suite nous avons présenté quelques interfaces ainsi que leurs analyses.

Conclusion Generale

Au terme de ce rapport, nous pouvons conclure que ce projet nous a donné une occasion opportune qui permet de confronter l'acquis théorique à l'environnement pratique.

En effet, ce projet nous a permis de prendre certaines responsabilités, ce qui nous a aidé à consolider de plus en plus nos connaissances pratiques et théoriques.

Du point de vue technique, ce projet nous a permis de nous adapter avec l'environnement professionnel du développement informatique, aussi il nous a permis de maîtriser la méthode de développement SCRUM et les technologies Node.js, git ...

Notre projet est toujours extensible et ouvert à l'évolution où nous pouvons ajouter la recherche par reconnaissance vocale, transformer l'application en mobile avec PWA.