



République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la

Recherche Scientifique

Université de sciences et de la technologie de Houari Boumediene

Faculté d'Informatique

Département IA & SD

Mémoire de Licence

Spécialité:

Licence Informatique Générale

Thème

**Conception et réalisation d'une application web
pour la gestion d'un parc zoologique**

Sujet encadré par:

Mr : BOUDLAA Ayoub

Soutenu le : 12/06/2022

Présenté par:

M^{elle} : CHABNI Hanane

M^{elle} : FERKOUS Sarah

Devant le jury composé de:

M^{me} BOUGHACI Dalila

Président

M^r KHENNAK Ilyes

Membre

Binôme n°:ACAD_I_027/2022

Remerciements

Nous remercions tout d'abord dieu tout puissant de nous avoir donné la force et la connaissance pour accomplir une action qui lui plaise.

Nous tenons à remercier tout particulièrement notre promoteur **MONSIEUR BOUDLAA**, pour nous avoir encadrés et guidés dans la réalisation de notre projet.

Nous exprimons aussi nos gratitudeles plus sincères à tous ceux qui nous ont apporté leur aide et qui ont accepté de répondre à nos questions avec gentillesse.

Nous tenons ensuite à remercier **les membres du jury** d'avoir accepté d'honorer par leur jugement notre travail ainsi que pour le temps qu'ils consacreront pour donner leurs avis et corrections...

Dédicace 1

Je dédie ce modeste travail :

A mes très chers parents qui m'ont encouragé et m'ont conseillé pendant mes plus pénibles moments. Que DIEU vous garde pour moi.

A mes chères sœurs Thanina et Maroua et mon petit frère Ghiles pour leurs encouragements permanents, et leur soutien moral.

A toute ma famille, en particulier ma chère tante Karima qui m'a aidé beaucoup.

A mon binôme Sara, Merci pour ton soutien, ta patience et ta compréhension.

A tous ceux qui me sont chers.

Hanane...

Dédicace 2

Je dédie ce modeste travail :

*A mes chers parents qui m'ont encouragé et qui
Ont veillé à ce que je réussisse dans mes études*

*A mes deux frères Abd El Rahime et Amine, et ma belle sœur Ikrame, que
j'aime beaucoup.*

A mon binôme Hanane, merci pour votre soutien.

A Tous ceux qui me sont chers

Je dédie le fruit de mes 3 ans d'études.

Sarah...

Résumé : Les applications Web actuelles ont connu une grande révolution qui permet de présenter des informations de différentes façons dans plusieurs domaines.

Dans ce travail, nous souhaitons créer une application web pour la gestion d'un parc zoologique et qui offre plusieurs services aux différents utilisateurs. Notamment la gestion des employés et des animaux.

Pour la création de cette application, nous avons passé par plusieurs étapes, en commençant d'abord par une étude de l'existant, par la suite nous avons suivi une phase conceptuelle détaillée en utilisant le langage de modélisation « UML ». Enfin, afin de réaliser l'outil, différentes technologies ont été utilisées: python avec le Framework "flask", angular, html5, css...

Abstract : Today's web applications have a great revolution which can present information in different ways in several areas.

In this work, we want to create a web application for the management of a zoological park that offers several services to different users, including employees and animals management.

For the creation of this application, we went through several stages. First, we started with a study of the existing systems, then we followed a detailed conceptual phase using the language of modelling «UML». Finally, in order to realize this application, different technologies have been used:python with framwork"flask", angular, html5, css ...

Sommaire

| | |
|--|----|
| INTRODUCTION GENERALE | 1 |
| Chapitre 1 | |
| 1.1. Introduction | 2 |
| 1.2. Définitions d'un parc zoologique | 2 |
| 1.3. Objectif d'un parc zoologique | 2 |
| 1.4. Définition des mots clés | 3 |
| 1.5. Description générale du parc ELHAMMA | 3 |
| 1.5.1. Définition du parc ELHAMMA | 3 |
| 1.5.2. Ses animaux | 3 |
| 1.5.3. Ses services | 5 |
| 1.6. Etude de l'existence | 5 |
| 1.7. Analyse et spécification des besoins | 5 |
| 1.7.1. La problématique | 5 |
| 1.7.2. Identification des acteurs | 6 |
| 1.7.3. Les besoins fonctionnels | 7 |
| 1.7.4. Les besoins non fonctionnels | 7 |
| 1.8. Introduction au web | 8 |
| 1.8.1. La définition d'une application web | 8 |
| 1.8.2. Les avantages d'une application web | 8 |
| 1.8.3. <i>Objectif de notre application web</i> | 8 |
| 1.9. Conclusion | 8 |
| Chapitre 2 | |
| 2.1. Introduction | 9 |
| 2.2. Unified Modeling Language | 9 |
| 2.2.1. Définition | 9 |
| 2.2.2. Le Choix d'UML comme outil de modélisation | 9 |
| 2.3. Diagramme de cas d'utilisation | 9 |
| 2.3.1. Définition | 9 |
| 2.3.2. Identification des cas d'utilisation | 10 |
| 2.3.2.1. Fonctionnalités du directeur | 10 |
| 2.3.2.2. Fonctionnalités du chef animalier | 11 |
| 2.3.2.3. Fonctionnalités du vétérinaire et le soigneur animalier | 11 |
| 2.3.2.4. Fonctionnalités du visiteur | 11 |

| | |
|--|-----------|
| 2.4. Diagramme de séquence | 12 |
| 2.4.1. Définition | 12 |
| 2.4.2. Spécification des scénarios à décrire | 12 |
| 2.5. Diagramme de Classe | 22 |
| 2.5.1. Définition | 22 |
| 2.5.2. Représentation de diagramme de classe | 23 |
| 2.5.3. Les règles de gestion | 24 |
| 2.6. Modèle relationnelle | 24 |
| 2.6.1. Définition | 24 |
| 2.6.2. Les règles de passage | 24 |
| 2.7. Codification proposé | 26 |
| 2.7.1. Pour l'animal | 26 |
| 2.7.2. Pour l'employé | 26 |
| 2.8. Conclusion | 26 |

Chapitre 3

| | |
|--|-----------|
| 3.1. Introduction | 27 |
| 3.2. Environnement de travaille | 27 |
| 3.2.1. Environnement matériel | 27 |
| 3.2.2. Environnement logiciel | 27 |
| 3.3. Outils de développement | 27 |
| 3.4. Interface de l'application | 28 |
| 3.4.1. La page d'accueil | 28 |
| 3.4.2. Espace d'authentification | 29 |
| 3.4.3. Les interfaces des acteurs | 29 |
| 3.4.3.1. L'interface de directeur | 30 |
| 3.4.3.2. L'interface de chef animalier | 31 |
| 3.4.3.3. L'interface de vétérinaire | 33 |
| 3.4.3.4. L'interface de soigneur animalier | 33 |
| 3.4.4. Page de contact | 34 |
| 3.5. Conclusion | 34 |
| Conclusion générale | 35 |
| Bibliographie | 36 |

Liste des figures

| | |
|--|----|
| Figure2.1- Diagramme de cas d'utilisateur de directeur | 10 |
| Figure2.2- Diagramme de cas d'utilisateur des acteurs | 12 |
| Figure2.3- Diagramme de cas d'utilisateur de visiteur | 12 |
| Figure2.4- Diagramme de séquences « authentification » | 13 |
| Figure2.5- Diagramme de séquences « ajouter animal» | 14 |
| Figure2.6- Diagramme de séquences « modification d'un animal » | 15 |
| Figure2.7- Diagramme de séquences « suppression d'un animal» | 16 |
| Figure2.8- Diagramme de séquences « recherche d'un animal» | 17 |
| Figure2.9- Diagramme de séquences «la gestion des acteurs» | 19 |
| Figure2.10- Diagramme de séquences « la gestion des employés» | 21 |
| Figure2.11- Diagramme de séquences « recherche d'un employé» | 22 |
| Figure2.12- Diagramme de séquences « diagramme de classe» | 23 |
| Figure3.1 –Interface d'accueil | 28 |
| Figure3.2 - Interface de connexion (login) | 29 |
| Figure3.3 - Interface d'un acteur | 29 |
| Figure3.4 - Interface de profil d'un acteur | 30 |
| Figure3.5 - Interface de la gestion des acteurs | 30 |
| Figure3.6 - Interface d'ajouter un nouvel acteur | 31 |
| Figure3.7 - Interface de la modification d'une activité | 31 |
| Figure3.8 - Interface de la gestion des animaux | 32 |
| Figure3.9 - Interface de détail d'un animal (chef animalier) | 32 |
| Figure3.10 - Interface d'ajouter un nouveau employé | 33 |
| Figure3.11 - Interface de la gestion des états des animaux | 33 |
| Figure3.12 - Interface de détail d'un animal (vétérinaire) | 34 |
| Figure3.13 -Interface de contact | 34 |

Liste des tableaux

| | |
|--|----|
| Tableau1.1-Table des animaux de parc | 5 |
| Tableau2.1-Table d'identification des cas d'utilisateurs | 10 |
| Tableau2.2-Table de spécification des scénarios | 12 |

Introduction générale

Depuis la découverte de l'informatique, de nombreuses activités de la vie courante ont été simplifiées. Actuellement, les individus peuvent facilement traiter des informations en se servant des logiciels et des réseaux informatiques. Compte tenu de son évolution, ce système caractérise la majorité des grandes entreprises quel que soit le secteur d'activité. En effet, elle offre des solutions qui facilitent la vie de l'humanité. Parmi ces solutions, on cite le développement des applications web qui a connu un grand succès dans les dernières années.

Le parc zoologique HAMMA fait partie intégrante des jardins que l'informatique pourra beaucoup aider. En effet, la croissance de nombre d'animaux et des employés nécessite la mise en place d'une gestion rationnelle et rapide.

Le suivi quotidien des animaux, employés, activités... est important pour le parc, le directeur peut donc consulter les listes de tous les acteurs, activités, services. Le chef animalier consulte également les animaux, et partage les informations avec le vétérinaire et le soigneur animalier.

Le suivi a pour objectif de donner du sens au règlement et à la sécurité. Elle prend en compte un certain nombre d'éléments : les informations détaillées sur l'animal et la responsabilité du vétérinaire (suivi médical), et du soigneur animalier (suivi quotidien) tout est enregistré.

Dans la plupart des parcs, le contrôle ou le suivi des employés ou animaux se fait d'une manière manuelle, ce qui engendre un certain nombre de problèmes tels que la perte du temps, masse de stockage importante etc... Afin de pallier ces problèmes, ce présent projet permet d'offrir une solution qui consiste à concevoir et à implémenter une application web permettant d'automatiser toutes les informations.

Pour atteindre les objectifs de ce projet de fin d'études, nous avons procédé comme suit :

Chapitre 1 : Dans ce chapitre « Etude de l'Existant », on présentera les parcs zoologiques en général et puis identifie les acteurs de notre parc et la solution proposée qui est la création d'une application web pour régler tous nos problèmes.

Chapitre 2 : Dans le chapitre « Conception » on présente la conception qui a été mise en œuvre tout au long de la réalisation de ce projet et qui répond aux besoins de la gestion des animaux et employés.

Chapitre 3 : Le dernier chapitre « Réalisation » présente l'environnement de travail ainsi que les outils et les langages de programmation utilisés, qui nous ont permis de réaliser notre système.

Chapitre 1 : Etude de l'existant et analyse des besoins

1.1. Introduction

Un parc est une zone délimitée d'un territoire, maintenue dans son état naturel (dans un but de conservation de la nature ou dans un but de loisir). Il existe plusieurs types des parcs par exemple les parcs des loisirs, et dans notre cas, on va étudier un type du parc qui est le parc zoologique.

1.2. Définitions d'un parc zoologique

Un parc zoologique, aussi appelé jardin zoologique, ou plus communément zoo, est un espace où se sont réunies de nombreuses espèces animales, pour la plupart sauvages, vivant dans des espaces clos. Ils ont pour but le divertissement, la conservation des espèces, la pédagogie et la recherche scientifique. On recense plus de 2 000 zoos dans le monde, attirant un ensemble de près de 350 millions de visiteurs par année.

1.3. Objectif d'un parc zoologique

Les parcs zoologiques ont quatre fonctions fondamentales à remplir : la récréation, l'éducation, la recherche, la conservation et la préservation.

- **La récréation** : Il est primordial de satisfaire le plaisir des visiteurs citadins, puisque les parcs zoologiques sont des lieux de détente, de divertissement et de découverte. Ainsi, plusieurs parcs ont décidé d'installer une partie attraction touristique, à travers un manège, un train ou un bateau promenade, au sein de leur parc, pour élargir les services proposés. D'autres, ont préféré associer à leurs activités une mise en valeur patrimoniale, paysagère ou encore botanique.
- **L'éducation** : C'est un aspect important, car par le biais des panneaux d'information concernant les espèces et les milieux naturels menacés, les parcs transmettent et sensibilisent un maximum de visiteurs, dans le but que ces derniers prennent conscience de l'action (bénéfique et plus souvent maléfique) de l'homme sur la faune sauvage menacée et plus largement sur la nature. Bien souvent, les parcs ont également des programmes pédagogiques à destination des écoles et autres groupes d'enfants dans le but de les sensibiliser dès le plus jeune âge.
- **La recherche** : La zoologie, l'écologie, la parasitologie font partie des recherches menées au sein des parcs animaliers dans le but d'acquérir des connaissances précises et indispensables. Ces dernières doivent contribuer à la préservation des espèces, à la surveillance de leur bon état de santé, ainsi qu'au bon déroulement des programmes de reproduction.
- **La conservation et la préservation** : Les parcs animaliers doivent contribuer à la protection de la nature, mais également à la défense des droits des animaux et à la conservation des espaces naturels intacts. Pour éviter l'extinction, la seule solution pour l'instant est l'élevage

en captivité, en attendant de pouvoir réintégrer les individus dans leurs biotopes naturels et de pouvoir renforcer la population naturelle. Les parcs animaliers n'ont donc pas pour but officiellement d'exploiter, mais de conserver.

1.4. Définition des mots clés

- **Animal** : Être vivant non végétal, ne possédant pas les caractéristiques de l'espèce humaine.
- **Aliment** : Toute substance qui peut servir de nourriture à un être vivant.
- **Espèce** : Nature propre à plusieurs êtres vivants ou à plusieurs choses, qui permet de les faire entrer dans une classe.
- **Cage** : Endroit clos (par des barreaux, du grillage) servant à tenir enfermés des animaux vivants.
- **Enclos** : Un enclos est un espace de terrain entouré d'une clôture qui sert à contenir des animaux domestiques, à délimiter un monument commémoratif. En archéologie, un enclos fossoyé est un espace correspondant à un système funéraire ou agro-pastoral.
- **Zone** : Partie importante d'une surface sphérique comprise entre deux plans parallèles.
- **Zoo** : Un zoo est un parc ou jardin zoologique réunissant de nombreuses espèces animales et étant destiné à être visité.

1.5. Description générale de parc ELHAMMA

1.5.1. Définition de parc ELHAMMA

Le jardin d'essai d'ELHamma : est l'un des jardins botaniques les plus célèbres au monde, situé dans le quartier d'ELHamma à Alger, est un jardin luxuriant, qui s'étend en amphithéâtre, au pied du Musée National des Beaux-Arts d'Alger, de la rue Mohamed Belouizdad à la rue Hassiba Ben Bouali, sur une superficie de 58 hectares (38 hectares de jardin et 20 hectares d'arboretum). Créé en 1832, il est considéré comme l'un des jardins d'essai et d'acclimatation les plus importants au monde.

Le Jardin d'essai n'est pas seulement un centre de production botanique ou horticole, mais aussi un centre d'enseignement et un lieu de promenade fort apprécié des Algérois. On y dénombre plus de 1 200 espèces végétales.

1.5.2. Ses animaux

Notre parc zoologique accueille l'une des plus grandes diversités d'espèces animales en Algérie, le parc contient plus de 90 animaux de différents continents.

Les animaux qui existent dans le parc zoologique sont représentés dans le tableau suivant :

| Animal | Nombre | Famille | Description |
|---------|--------|---------|---|
| Lion | 5 | Félin | Le Lion est une espèce de mammifères carnivores. Il pèse en moyenne 175 kg et mesure entre 136 à 198 cm le lion possède un pelage de couleur sable ou ocre foncé. |
| Guépard | 2 | Félin | Le guépard est un grand mammifère carnassier vivant en Afrique et en Asie de l'Ouest. Le guépard a une allure svelte et fine, avec de longues pattes élancées, et une face au museau court marquée par deux traces noires partant des yeux. |

| | | | |
|-------------------------|-----------|-----------------------|---|
| Tigre d'inde | 3 | Félin | Le Tigre est une espèce de mammifère carnivore. Il possède des rayures verticales noires sur un pelage orangé. Le mâle pèse 180 à 260 kg, tandis que la tigresse ne pèse que 130 à 160 kg. Disposant d'une longue queue qui peut aller jusqu'à 1 mètre, cet animal peut atteindre une taille de 2,80 m. |
| Gazelle | 3 | Antilopes | Les gazelles sont des mammifères élégants et délicats. Leur taille et leur poids varient selon les espèces. Elles mesurent 80 à 200 cm de longueur, 50 à 120 cm au garrot et pèsent entre 12 et 75 kg. |
| Lama | 1 | Camélidés | Le lama est un mammifère domestique, de la même famille que le dromadaire et le chameau. Il mesure entre 100 et 190 cm de la tête aux sabots, pour un poids variant entre 130 et 200 kg. |
| Dromadaire | 2 | Camélidés | Le dromadaire, ou chameau d'Arabie, est une espèce de chameau, mammifère. Il pèse 400 à 800 kg, il mesure 1.8 m à 2m de longueur. |
| Zèbre | 1 | Equidés | Le zèbre est un animal d'Afrique centrale et australe voisin de l'âne. Il est de couleur blanche avec des rayures noires .il mesure 1,10 à 1,25 m en moyenne au garrot, pour une longueur entre 2,20 et 2,70 m et un poids d'environ 300 kg. |
| Loup | 5 | Canidé | Le loup est un mammifère qui mesure entre 70 centimètres et 1m50 cm, sa taille varie selon les espèces. Ses petits yeux sont brillants, de couleur foncée. Sa queue mesure entre 30 et 50 cm. A l'âge adulte, il pèse environ 40 kg |
| Fennec | 2 | Canidé | Le fennec, nommé aussi renard des sables est un petit renard qui vit dans le désert du Sahara et la péninsule du Sinaï il pèse 0.68 kg à 1.6 kg, il mesure 30 cm à 41 cm de longueur. |
| Ours brun | 1 | Ursidés | C'est un animal mammifère, carnivore à forte corpulence, qui peut atteindre une masse corporelle de 130 à 700 kg. Il pèse entre 120 et 700 kg et mesure entre 1,5 et 3,5 m de hauteur. |
| Crocodile | 2 | Reptiles | Le crocodile est un reptile qui possède une grande gueule et des puissantes mâchoires. C'est un redoutable animal qui peut atteindre plus de 7 mètres de longueur. Son corps est moins court que sa queue. |
| Kangourou | 1 | macropodidé | Le kangourou est un marsupial qui possède une poche abdominale (appelée également poche marsupiale) chez les femelles. Cette poche leur permet d'abriter leurs petits. Il mesure entre 1,10 m et 1,80 m selon le sexe, et sa queue peut atteindre jusqu'à 1 mètre de longueur. |
| Eléphant d'Asie | 2 | Éléphantidés | Les éléphants sont des mammifères proboscidiens, ils pèsent jusqu'à 8 tonnes et mesurent 3 à 4 mètres de hauteur, pour environ 7 mètres de longueur. |
| Poney | 2 | Equidé | Un poney, appelé cheval nain en Louisiane, est, selon la définition courante, un cheval de petite taille, avec une conformation et un tempérament particuliers et mesure entre 1,07 m et 1,48 m. |
| Singe magot | 10 | Hominidé-singe | Un mâle qui mesure environ 64 cm de long pour 17 kg, et la femelle 58 cm pour 13 kg. Il est le seul. macaque vivant hors d'Asie. |
| MANGABEY | 3 | Hominidé-singe | C'est une espèce herbivore, Grand singe peuplant les forêts d'Afrique et arborant une couleur brune dorée, grise ou noir ainsi qu'une longue queue généralement plus grande que leur propre corps. |
| CERF DE BARBARIE | 2 | Cervidé | Aussi appelé cerf de l'Atlas, est une sous-espèce du cerf élaphe. Ce cerf au bord de l'extinction, est le seul représentant de Cervidé en Afrique. Il possède des bois qui peuvent atteindre environ 80 cm. Il pèse entre 150 à 225 kg (mâle) |
| Oryx | 1 | Cervidé | Les Oryx forment un genre d'antilopes. Ils sont répandus en Afrique, ainsi que sur la péninsule Arabique, ils portent typiquement des cornes longues et droites, pointant vers l'arrière. |
| Oie | 3 | Anatidé | L'oie sauvage se présente avec un plumage gris-beige, une grosse tête, et des pattes roses et palmées. Elle mesure entre 70 et 90 cm pour un poids variant de 2,5 à 3,5 kg, et son envergure varie de 140 à 185 cm. |

| | | | |
|------------------|-----------|-------------------|---|
| Cacatoès | 5 | Psittacidé | Le Cacatoès est un bel oiseau originaire d'Indonésie et d'Australie, la longévité moyenne est entre 50 et 60 ans. |
| Canard | 20 | Anatidé | Les canards sont des oiseaux aquatiques au bec caractéristique, c'est un animal d'élevage que l'on trouve un peu partout. Dans la nature, il apprécie les rivières, étangs et marais. Le corps d'environ 65 cm pour une envergure de 140 cm et un poids moyen de 3 à 3,5 kg. |
| Perroquet | 2 | Psittacidé | Ce sont des oiseaux souvent très colorés. Ils ont un gros bec crochu, certains perroquets sont également capables d'imiter la voix humaine, et de répéter des mots, ou des sons. |
| Paon | 3 | Poulets | Les paons sont des grands oiseaux connus pour leur aspect majestueux. Leur poids varie entre 3,5 et 6 kg. Sans la queue, le mâle mesure entre 90 et 110 cm, et la femelle, légèrement plus petite, mesure 86 cm. Le paon le plus répandu est le paon bleu aux plumes et ocelles bleue et verte. |

Table 1.1 : Table des animaux de parc

1.5.3. Ses services

Le jardin d'essai d'ELHamma se compose de plusieurs unités :

- Un jardin botanique, qui comprend le jardin français et le jardin anglais. Dans chacun des deux jardins, il y a un certain nombre de couloirs et un bassin et fontaines d'eau.
- Un zoo, qui comprend des centaines d'espèces d'animaux, d'oiseaux et de poissons : la reproduction en captivité est une des actions importantes des parcs animaliers. De nombreuses actions de conservation sont entreprises et certains parcs se spécialisent dans la reproduction d'un groupe précis d'espèces.
- École d'horticulture agricole, créée en 1918 pour enseigner le jardinage.
- École d'éducation environnementale, pour sensibiliser et éduquer les enfants, en particulier les adultes, aux problèmes environnementaux.
- Le Musée national des beaux-arts, un grand musée présentant à la fois de l'art moderne et ancien, est situé en face de l'entrée sud du parc.

1.6. Etude de l'existant

L'analyse de l'existant ou analyse détaillée a pour but l'étude du système existant, son diagnostic, un aperçu de l'ensemble des problèmes qui seront résolus suivant des objectifs bien précis à déterminer. Pour ce faire, il faut d'abord étudier le système existant lui-même ainsi que son environnement, le critiquer et faire sortir les insuffisances et les manques, puis proposer une solution, en détaillant les fonctionnalités requises par l'utilisateur pour se placer dans l'optique du démarrage de réalisation d'un projet réel.

1.7. Analyse et spécification des besoins

1.7.1. La problématique

- **Notre problème** : Le jardin d'ELHamma fait partie des plus grands jardins du monde, sa gestion nécessite donc une grande efficacité, en plus d'une organisation coordonnée, que ce soit par son directeur ou ses ouvriers, et ce pour garder sa valeur, en particulier en préservant ses organismes vivants de la diversité des plantes et assurer la sécurité des animaux qui ont

besoin de soins et de contrôles particuliers. D'autre part, le parc d'ELHamma d'un poste prestigieux, son objectif est d'attirer le plus de visiteurs possible de l'intérieur et de l'extérieur du pays.

Malgré les efforts fournis par l'ensemble des employés pour assurer une bonne gestion des animaux, certains problèmes restent toujours posés et qui peuvent être résumés comme suit :

- Une grande perte du temps dans l'extraction de différentes informations concernant les animaux ou les employés
- Le suivi manuel de toutes les tâches issues d'ajout, d'extraction, et les modifications.
- Le risque de perte d'information en cas de destruction des documents.
- Une mal gestion des animaux et des employés.

Afin de pouvoir gérer ce jardin et répondre à toutes ses exigences, il faut penser à une solution qui répond à tous ces besoins et les facilite de la meilleure façon.

- **Les solutions** : Pour résoudre les problèmes rencontrés qui résident essentiellement dans la gestion du parc, nous proposons de développer une application Web, qui permettra au gestionnaire de gérer plus facilement son parc et de suivre ses employés, de déterminer les insuffisances, ainsi de permettre aux employés d'améliorer la qualité de leur travail. Aussi, cette application sera une guide pour les visiteurs, qui leur permettra de visualiser le jardin et de faciliter leurs déplacements à l'intérieur du jardin, en connaissant les emplacements de toutes ses installations.

1.7.2. Identification des acteurs

- **Le directeur** : Le Directeur s'occupe de réunir et de manager les moyens techniques, financiers et humains de son parc, afin d'atteindre les buts fixés. Le développement de l'image du parc fait également partie de ses fonctions :
- Il est responsable de la gestion et de l'administration du site.
 - La gestion des équipes, services, et activités.
 - La communication et le marketing du parc.
 - La sécurité des visiteurs comme celle des animaux.
 - L'avancement des projets avec la rénovation des enclos ou la construction de nouveaux espaces.
- **Le vétérinaire** :
- Il a pour mission de veiller à la bonne santé des animaux. Pour cela, il y a bien sûr les traitements à apporter en cas de plaie ou d'incident ponctuel (désinfection, suture, anesthésie, opération).
 - Un vétérinaire de parc animalier peut également donner des instructions aux soigneurs pour les rations alimentaires ou l'entraînement des animaux, notamment dans le cas du médical training, c'est-à-dire l'entraînement des animaux à être

manipulés dans le cas où le vétérinaire devrait un jour faire un soin.

- **Soigneur animalier:** Parmi les postes les plus représentés dans un parc zoologique, on retrouve la profession du soigneur animalier :
 - Ce professionnel est chargé de soigner et de nourrir les animaux, sans oublier sa plus grande mission, l'entretien et le nettoyage de leur espace de vie (enclos, bâtiments, cages, aquariums...).
 - Il travaille la plupart du temps à l'extérieur, quelles que soient les conditions climatiques (canicule, vent, neige, pluie.....).
- **Chef animalier :** ce professionnel exerce différentes tâches, en plus de ses missions d'encadrement :
 - Le responsable de la gestion des employés et des animaux.
 - Capture et contention des animaux avec le vétérinaire, aménagement des enclos, suivi de la réalisation de certains travaux ;
 - Il peut également avoir la fonction de responsable de collection, c'est à dire choisir les animaux qui peuvent cohabiter ensemble, se reproduire ou être échangés.
- **Le visiteur :** Il peut faire les fonctionnalités suivantes :
 - Consulter les informations des animaux du parc ainsi leurs emplacements.
 - Consulter la description générale du parc et envoyer des messages.

1.7.3. Les besoins fonctionnels

Le système doit assurer les fonctions suivantes:

- La gestion des animaux, des employés, des services et des activités.

De plus, il offre d'autres fonctionnalités additionnelles :

- Décrit le parc pour les visiteurs, ainsi que son histoire.
- Le visiteur peut connaître les types d'animaux présents.

1.7.4. Les besoins non fonctionnels

A part les besoins fondamentaux, notre système doit répondre aux critères suivants :

- **La rapidité de traitement** : En effet, vu le nombre important des animaux, employés, toutes les gestions, il est impérativement nécessaire que la durée d'exécution des traitements s'approche le plus possible du temps réel.
- **La performance** : Un logiciel doit être avant tout performant c'est-à-dire, à travers ses fonctionnalités, répond à toutes les exigences des usagers d'une manière optimale.
- **La convivialité** : Le futur logiciel doit être facile à utiliser. En effet, les interfaces utilisateur doivent être conviviales c'est-à-dire simples, ergonomiques et adaptées à l'utilisateur.

1.8. Introduction au web

1.8.1. La définition d'une application web

Une application Web est en général basée sur HTML, JavaScript ou CSS. Etant donné qu'une telle application est chargée depuis un serveur Web et exécutée via un navigateur Web, vous n'avez donc aucune installation à prendre en charge, contrairement aux applications mobiles. Néanmoins, l'utilisation d'un signet ou raccourci permet l'accès direct à l'application depuis le bureau ou l'écran d'accueil d'un appareil mobile.

1.8.2. Les avantages d'une application web

Aujourd'hui et de plus en plus, il est possible de s'abonner à des applications web. Nous pouvons donc résumer les principaux avantages d'une application web de la manière suivante :

➤ **Maitrise du budget et diminution des couts :**

- La mise de départ est inexistante, les frais d'adaptation, de maintenance et d'améliorations sont inclus.

➤ **Gain de temps :**

- La mise en œuvre et le déploiement sont plus rapides, et l'utilisation des applications est plus intuitive et plus facilement.

➤ **Accessibilité optimisée :**

- Accès universel depuis n'importe quel type de poste : PC, portables, téléphone mobile, tablette... Et données sont disponibles 24h sur 24 et 7j sur 7.

➤ **Meilleure gestion de la sécurité :**

- L'accès aux données est contrôlé par identification et certificats.

➤ **Evolution et innovation continue :**

- Aucun ne risque d'obsolescence, et la mise à niveau des applications est automatique.

1.8.3. Objectif de notre application web

- Faciliter la consultation des données.
- Faciliter la gestion de parc.
- Sécuriser les informations internes contre les pertes et les modifications erronées.
- Création d'une base de données exploitable.

1.9. Conclusion

Dans ce chapitre, nous avons présenté quelques notions sur les applications web et les parcs zoologiques, qui ont besoin les bien gérer, publier, et attirer de nouveaux visiteurs. Après avoir fixé nos objectifs, on doit suivre des étapes bien précises afin de les atteindre. Dans le chapitre suivant, on commence par l'étude conceptuelle de notre application web, pour la gestion du parc zoologique d'Elhamma.

Chapitre2 : Conception et modélisation

2.1. Introduction

La phase de la modélisation et de conception, est la phase clé de la création et de l'implémentation de notre projet. En effet, elle représente une étape importante dans le cycle de développement logiciel après l'étape de l'étude de l'existant, l'étape d'analyse et conception consiste à analyser, spécifier et effectuer les choix de conception du système, grâce aux diagrammes UML. Dans ce deuxième chapitre, nous procédons à l'analyse des besoins, décrits dans le chapitre précédent, afin de livrer une description complète de la solution à réaliser et de produire un système qui répond aux besoins.

2.2. Unified Modeling Language (UML)

2.2.1. Définition

UML (Unified Modeling Language) est un langage graphique de modélisation des données et des traitements. Il est constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent à chacun une vision différente du projet à traiter. Il fournit 13 diagrammes qui se répartissent en deux grands groupes :

- Diagrammes structurels ou diagrammes statiques (UML Structure) : dans lesquels nous avons choisi d'utiliser le diagramme de classes (Class diagram).
- Diagrammes comportementaux ou diagrammes dynamiques (UML Behavior) : dans lesquels nous avons choisi le diagramme de cas d'utilisation (Use-Case diagram). Parmi les diagrammes dynamiques, on trouve :
 - ✓ Les diagrammes d'interaction: Dans lequel nous avons choisi d'utiliser le diagramme de séquence (Sequencediagram).

2.2.2. Choix d'UML comme outil de modélisation

UML est l'approche moderne des logiciels de modélisation et de documentation la plus populaire. Nous avons utilisé UML pour ses nombreux avantages :

- Il permet d'établir une structuration cohérente des fonctionnalités et des données.
- Il apporte une compréhension rapide du programme.
- Il permet de documenter très clairement les besoins exprimés par les utilisateurs.

2.3. Diagramme de cas d'utilisation

2.3.1. Définition

Le diagramme de cas d'utilisation est un diagramme représentant la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. Il englobe et assure toutes les relations

entre l'utilisateur et les objets que le système met en œuvre.

2.3.2. Identification des cas d'utilisation

Le tableau ci-dessous illustre les cas d'utilisation dans notre système :

| Numéro | Scénario | Acteur |
|--------|--|---|
| 1 | S'authentifier | Directeur, vétérinaire, soigneur animalier, chef animalier, |
| 2 | Gérer l'acteur | Directeur |
| 3 | Gérer les services | |
| 4 | Gérer les activités | |
| 5 | Consulter la liste des employés | Directeur, chef animalier |
| 6 | Consulter les informations des animaux | Directeur, chef animalier, vétérinaire, soigneur animalier. |
| 7 | Gérer animal | Chef animalier |
| 8 | Gérer employé | Chef animalier |
| 9 | Gérer état animal | Vétérinaire |
| 10 | Gérer nourritures | Soigneur animalier |
| 11 | Consulter service, activité | Visiteur |
| 12 | Consulter animal | |
| 13 | Contacteur administration | |

TABLE 2.1 – Table d'identification des cas d'utilisateur

2.3.2.1. Fonctionnalité du directeur

Les fonctionnalités du directeur se résument dans la gestion des acteurs, services, activités que ce soit l'ajout, la suppression ou la modification, mais aussi, de consulter la liste des employés et des animaux.

Le directeur est apte à visionner les données et les modifier à tout moment après l'authentification.

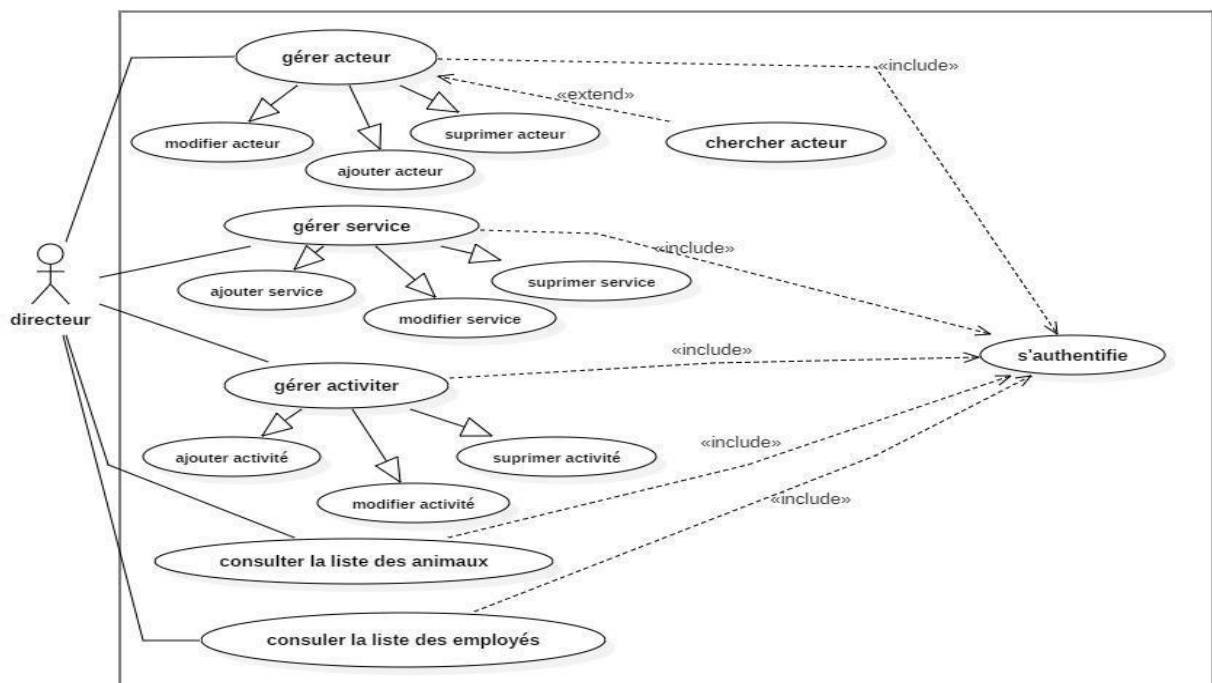


Figure 2.1 – Diagramme de cas d'utilisateur de directeur

2.3.2.2. Fonctionnalités du chef animalier

Le chef animalier a des différentes tâches à accomplir :

- La gestion des employés et des animaux.
- La consultation de la liste des employés, des animaux, et les détails sur les animaux.

Le chef animalier est apte à visionner les données et les modifier à tout moment après l'authentification.

2.3.2.3. Fonctionnalités du vétérinaire et le soigneur animalier

- Le vétérinaire s'occupe de la gestion d'état des animaux, que soit l'ajout, la suppression et la modification, il peut aussi consulter la liste d'animaux.
- Le soigneur animalier s'occupe de la gestion des nourritures de l'animal, il peut aussi consulter la liste des animaux.

Ils sont aptes à visionner les données et les modifier à tout moment après l'authentification.

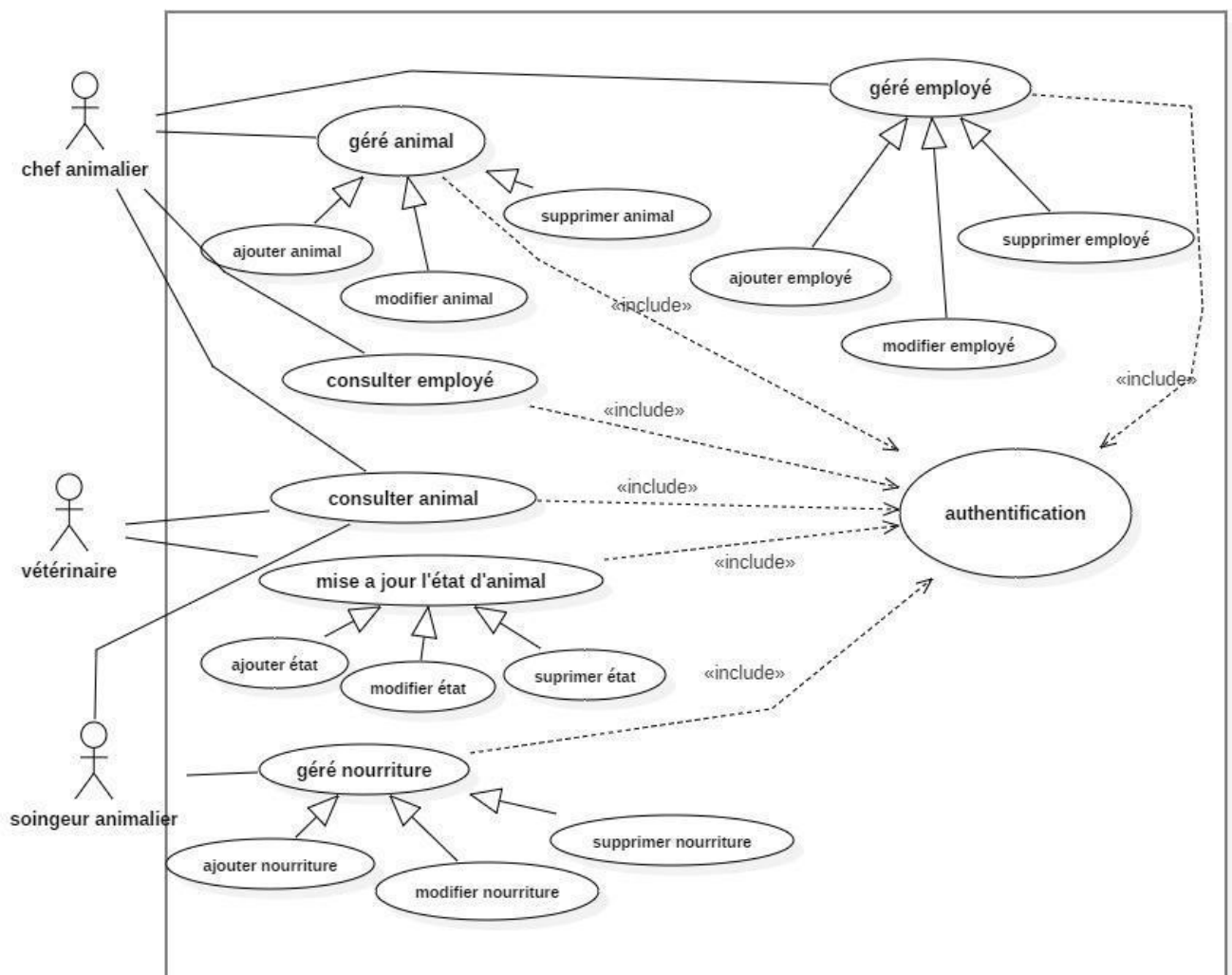


Figure 2.2 – Diagramme de cas d'utilisateur des acteurs

2.3.2.4. Fonctionnalités du visiteur

Les fonctionnalités du visiteur sont de consulter que soit la liste des animaux, des services, et des activités, ainsi que, de contacter l'administration.

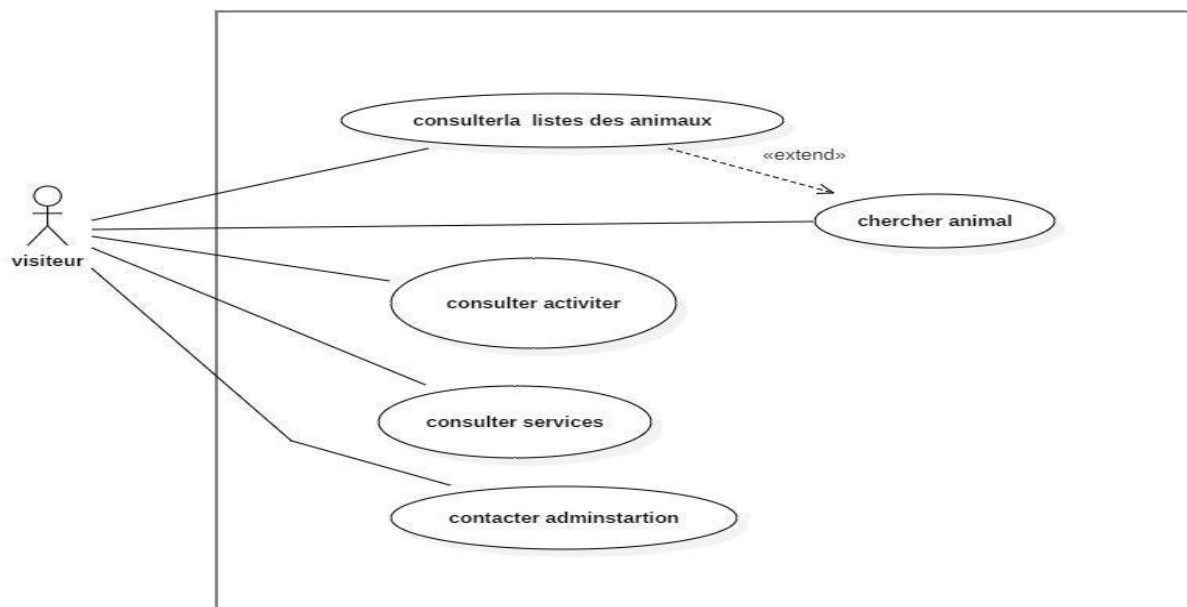


Figure 2.3 – Diagramme de cas d'utilisateur de visiteur

2.4. Diagramme de séquence

2.4.1. Définition

Un diagramme de séquence est un diagramme UML, qui représente la succession chronologique des opérations réalisées par un acteur. Il comprend un groupe d'objets que l'acteur va manipuler, représentés par des lignes de vie, et des messages que ces objets échangent lors de l'interaction.

2.4.2. Spécification des scénarios à décrire

| Numéro | Scénario | Acteur |
|--------|-----------------------------|--|
| 1 | Authentification | directeur, vétérinaire, chef animalier, soigneur animalier |
| 2 | L'ajout d'un animal | Chef animalier |
| 3 | La modification d'un animal | Chef animalier, vétérinaire, soigneur animalier |
| 4 | La suppression d'un animal | Chef animalier |
| 5 | La recherche d'un animal | Chef animalier, vétérinaire, soigneur animalier |
| 6 | La gestion des acteurs | Directeur |
| 7 | La gestion des employés | Chef animalier |
| 8 | La recherche d'un employé | Chef animalier |

TABLE 2.2 – Table de spécification des scénarios

2.4.2.1. Authentification

Cette tâche concerne le directeur, vétérinaire, chef animalier et le soigneur animalier. Quand l'un de ces derniers veut utiliser le système, il doit d'abord, introduire son matricule, nom d'utilisateur, son rôle, et son mot de passe, si ces quatre entrées sont correctes, le système affichera alors la page selon l'utilisateur sinon, il affichera un message d'erreur.

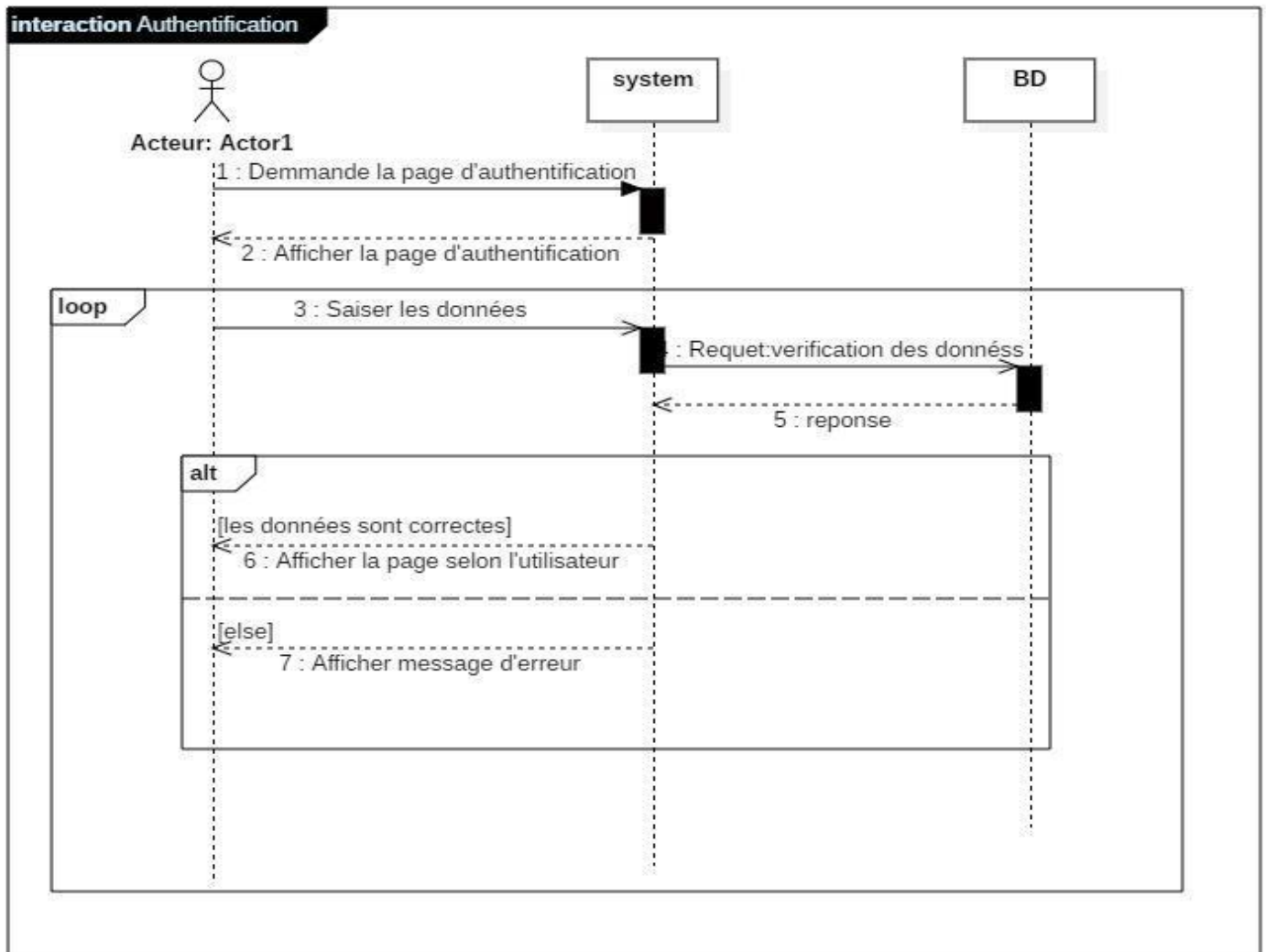


Figure 2.4 : Diagramme de séquences (authentification)

- **Diagramme de séquence «authentification»**

| | |
|---------------------|--|
| But | Permettre aux utilisateurs ayant un compte de se connecter à leurs comptes et de bénéficier des différents services offerts. |
| Acteur | Directeur, vétérinaire, chef animalier, soigneur animalier |
| Scénario normal | -L'utilisateur fait une demande d'authentification. -Le système lui affiche l'interface d'authentification. -L'utilisateur saisit ses informations et son adresse email et le mot de passe. -Le système vérifie les champs saisis. -Le système redirige l'utilisateur vers son interface |
| Scénario alternatif | -Saisie incorrecte (email et/ou mot de passe et/ou rôle) -Le système affiche un message d'erreur. |

2.4.2.2. L'ajout d'un animal

Le chef animalier est le responsable de cette tâche. En effet, si cet animal existe déjà dans la base de données, un message sera affiché : « Cet animal existe déjà », sinon : le système indique que cet animal a été ajouté avec succès dans la base de données.

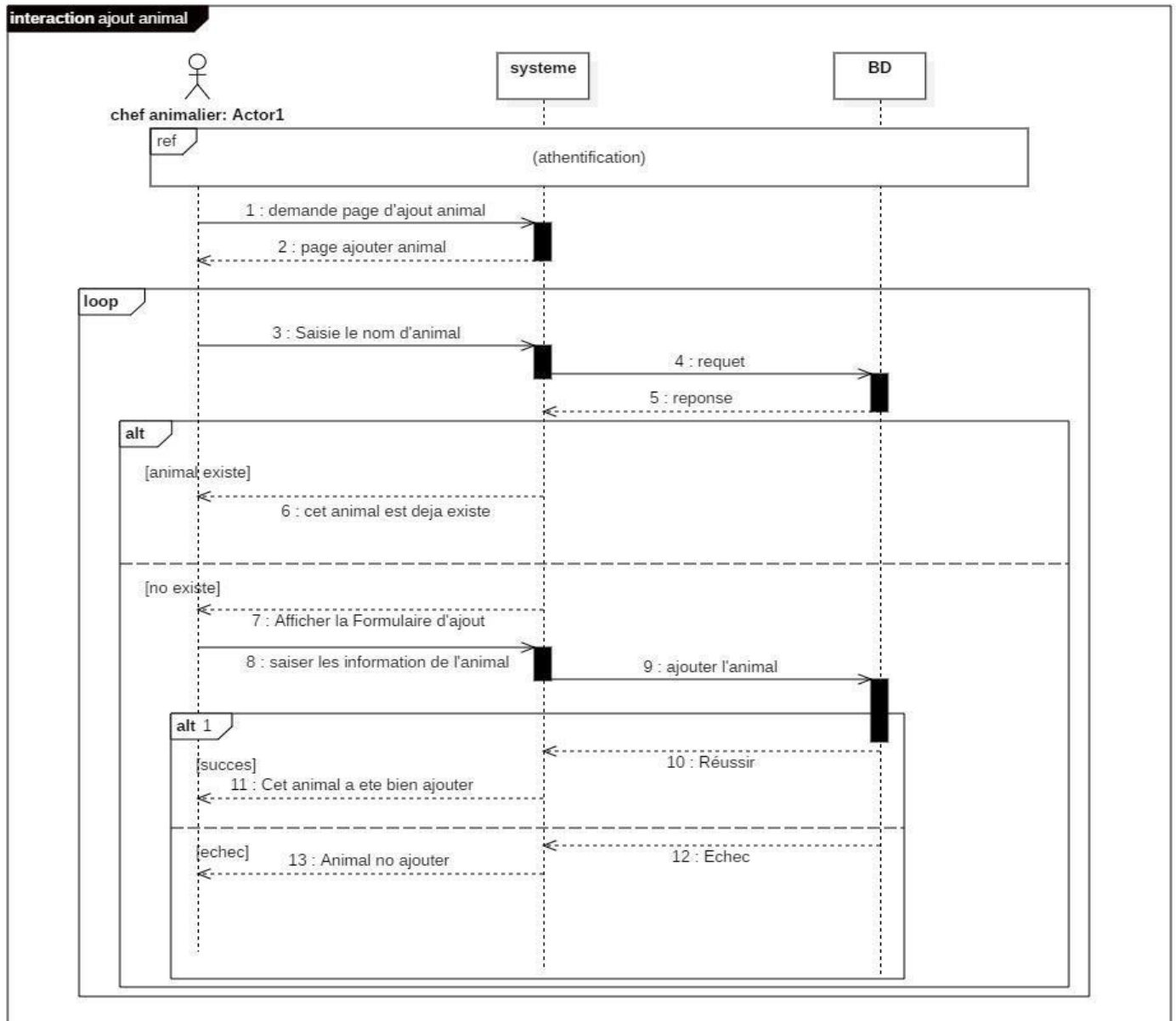


Figure 2.5: Diagramme de séquences (ajout animal)

- **Diagramme de séquence «ajout animal»**

| | |
|---------------------|---|
| But | Permettre au chef animalier d'ajouter un animal qui n'existe pas dans la BD. |
| Acteur | chef animalier |
| Scénario normal | <ul style="list-style-type: none"> -Le chef animalier fait l'authentification -il demande la page d'ajout d'un animal -Le système redirige le chef animalier vers la page -il remplit le formulaire de l'animal à ajouter -le system affiche un message : «cet animal est ajouté » |
| Scénario alternatif | <ul style="list-style-type: none"> -Saisie un code ou un nom d'un animal existe déjà -Le système affiche un message : « cet animal existe déjà ». -Les informations saisies sont incorrectes ou le formulaire n'est pas rempli complètement. - Le système affiche un message d'erreur. - L'employé quitte le site ou annule l'opération. -un problème de connexion : «Echec » |

2.4.2.3. La modification d'un animal

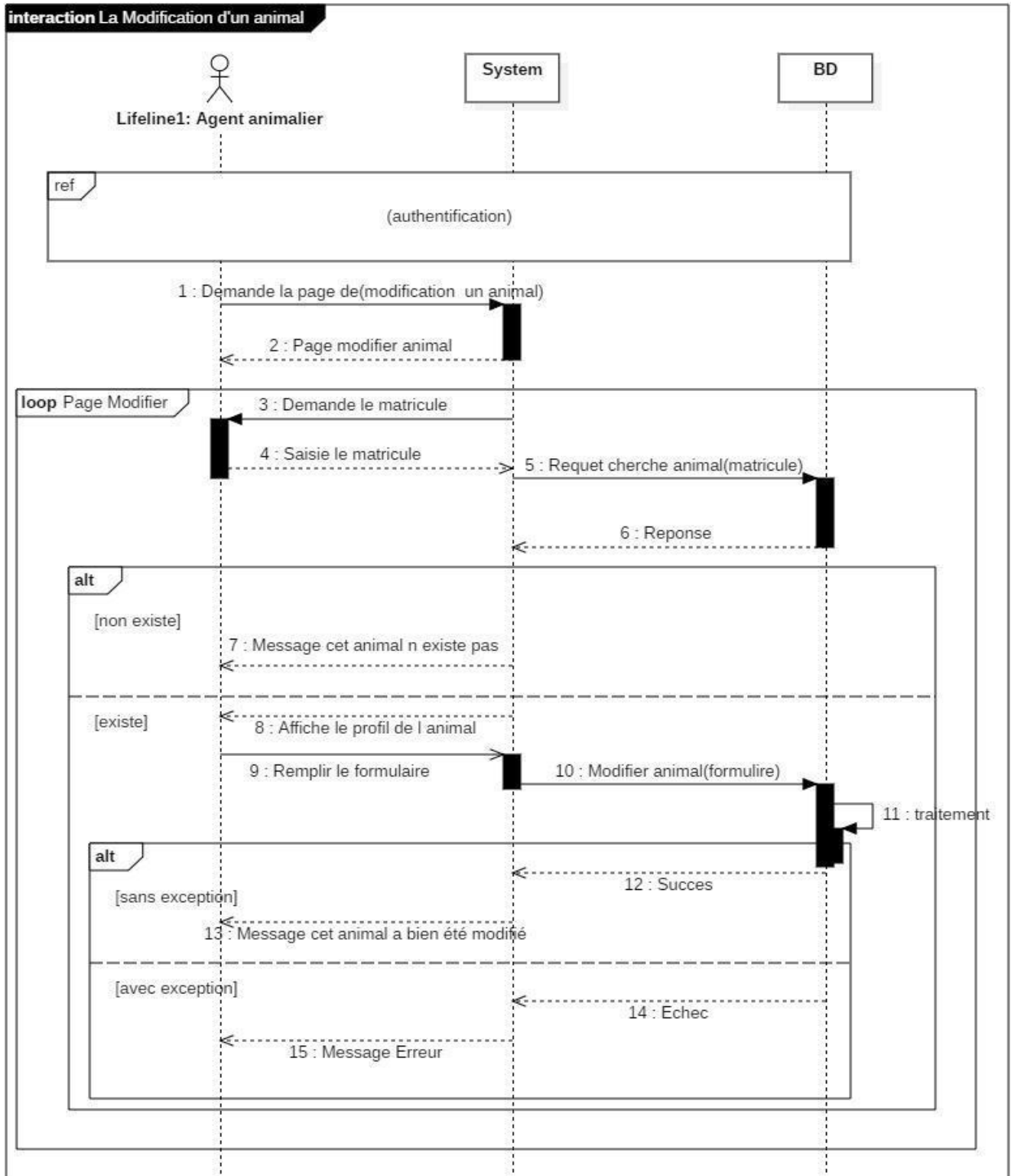


Figure 2.6 : Diagramme de séquences (modification d'un animal)

- **Diagramme de séquence «modification animal»**

Ce diagramme de séquence décrit la modification d'un animal. Le chef animalier, le vétérinaire ou le soigneur animalier sont les responsables. Si une modification s'impose et les informations saisies sont correctes, alors la modification aura lieu, sinon elle sera annulée.

| | |
|---------------------|--|
| But | Autoriser l'utilisateur à éditer un animal existant. |
| Acteur | chef animalier, vétérinaire, soigneur animalier |
| Scénario normal | -L'acteur fait l'authentification et accès à son page -Il demande la page de modification d'un animal -Le système redirige l'acteur vers la page -Il remplit le formulaire de l'animal à modifier -Le système affiche un message : «cet animal est modifié » |
| Scénario alternatif | -Les informations saisies sont incorrectes ou le formulaire n'est pas rempli complètement, le système affiche un message d'erreur. -L'acteur quitte le site ou annule l'opération. -un problème de connexion le système affiche : «Echec » |

2.4.2.4. La suppression d'un animal

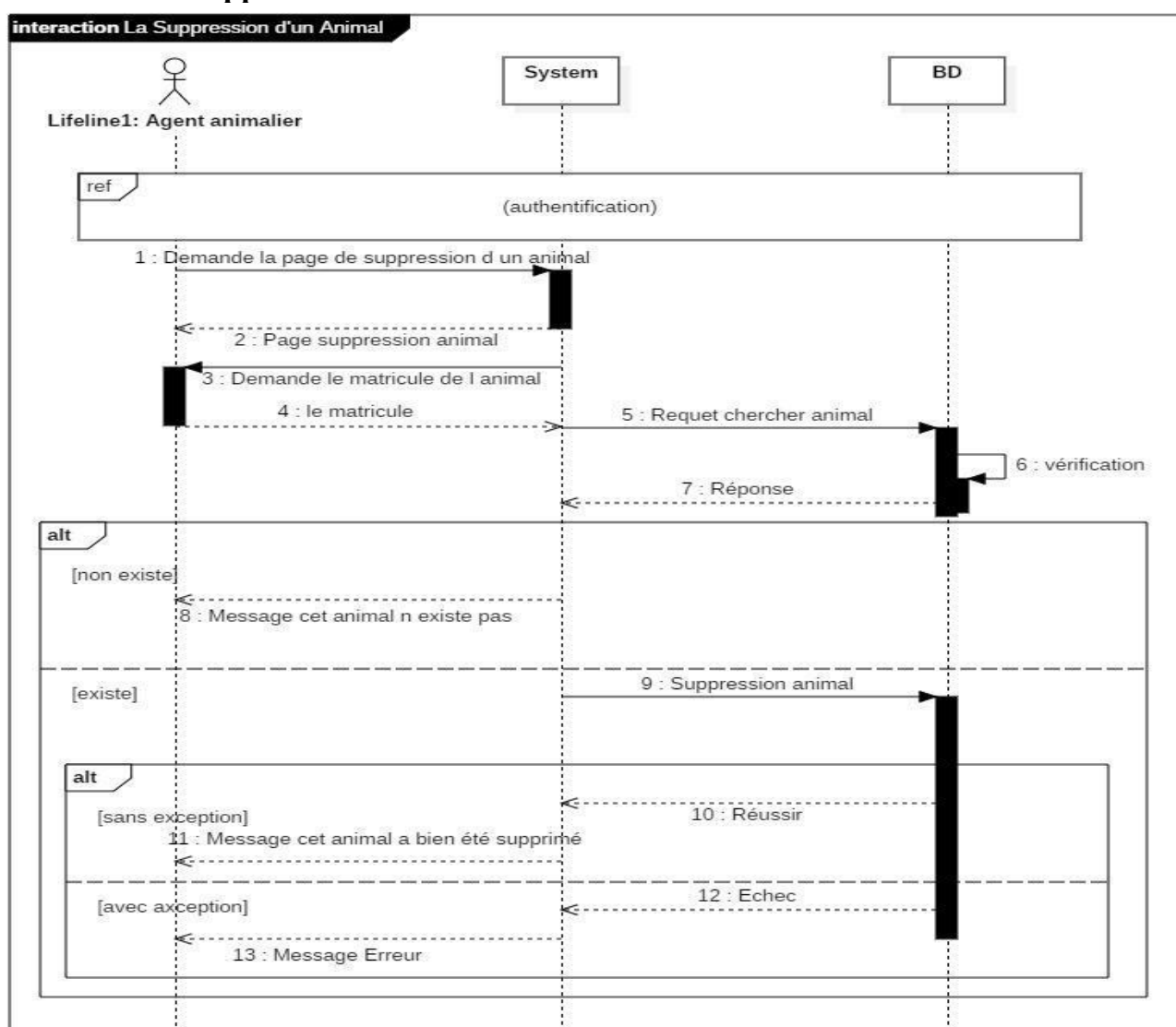


Figure 2.7 : Diagramme de séquences (suppression d'un animal)

● Diagramme de séquence «suppression animal»

Ce diagramme de séquence décrit le cas d'utilisation de la suppression d'un animal. Le chef animalier est le responsable. Si l'animal existe dans la base de données alors la suppression aura

lieu, sinon elle sera annulée.

| | |
|---------------------|--|
| But | Permettre au chef animalier de supprimer un animal. |
| Acteur | chef animalier |
| Scénario normal | -Le chef animalier fait l'authentification - Le système redirige le chef animalier vers sa page -Le chef animalier choisit l'animal qu'il veut supprimer et clique sur le bouton [Supprimer]. -Le système affiche un message de confirmation -Le chef animalier clique sur [oui]. -Le système affiche le message qui confirme la suppression. |
| Scénario alternatif | -L'employé quitte le site ou annule l'opération -un problème de connexion le système affiche : «Echec » |

2.4.2.5. La recherche d'un animal

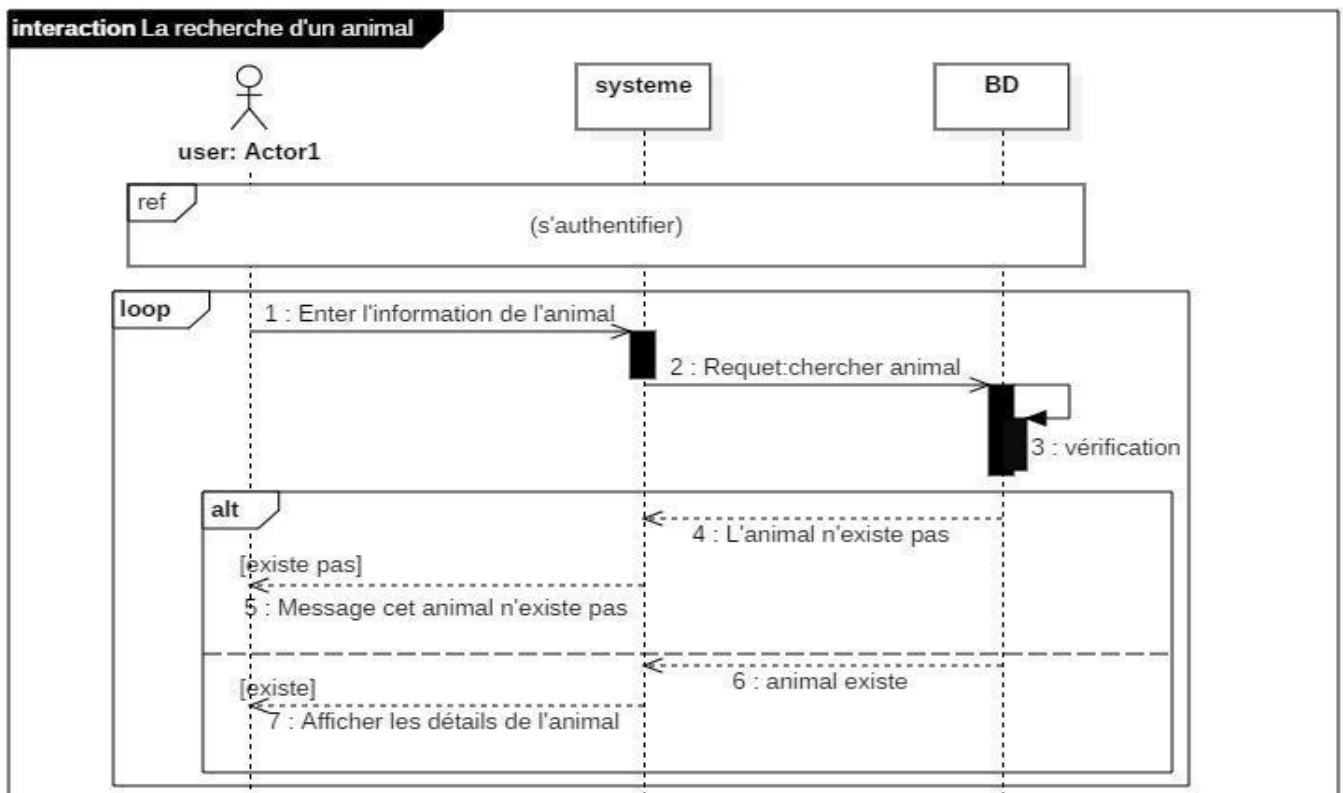


Figure 2.8: Diagramme de séquences (recherche d'un animal)

● Diagramme de séquence «recherche animal»

Le chef animalier, le vétérinaire et le soigneur animalier sont les responsables de cette tâche. En effet, si cet animal existe dans la base de données, ses informations seront affichées, sinon le système indique que cet animal n'existe pas.

| | |
|---------------------|--|
| But | Permettre au chef animalier, vétérinaire de chercher un animal dans la BD. |
| Acteur | chef animalier, vétérinaire, soigneur animalier |
| Scénario normal | <ul style="list-style-type: none"> -L'acteur fait l'authentification -Le système redirige l'acteur vers sa page -Il entre l'information de l'animal à rechercher -Le système affiche le profil de l'animal |
| Scénario alternatif | <ul style="list-style-type: none"> -Saisie l'information d'un animal qui n'existe pas, le système va afficher un message : « cet animal n'existe pas » -L'acteur quitte le site ou annule l'opération -un problème de connexion le système affiche : «Echec » |

2.4.2.6. La gestion des acteurs

Cette tâche concerne le directeur, après l'authentification, il demande la page de gestion des acteurs, et saisit son choix (ajout, suppression, modification), pour les trois cas, le système vérifié d'abord l'existence de l'acteur dans la BD, après, il exécute la tâche et affiche un message de succès, sinon il affiche un message d'échec.

interaction La Gestion des acteurs

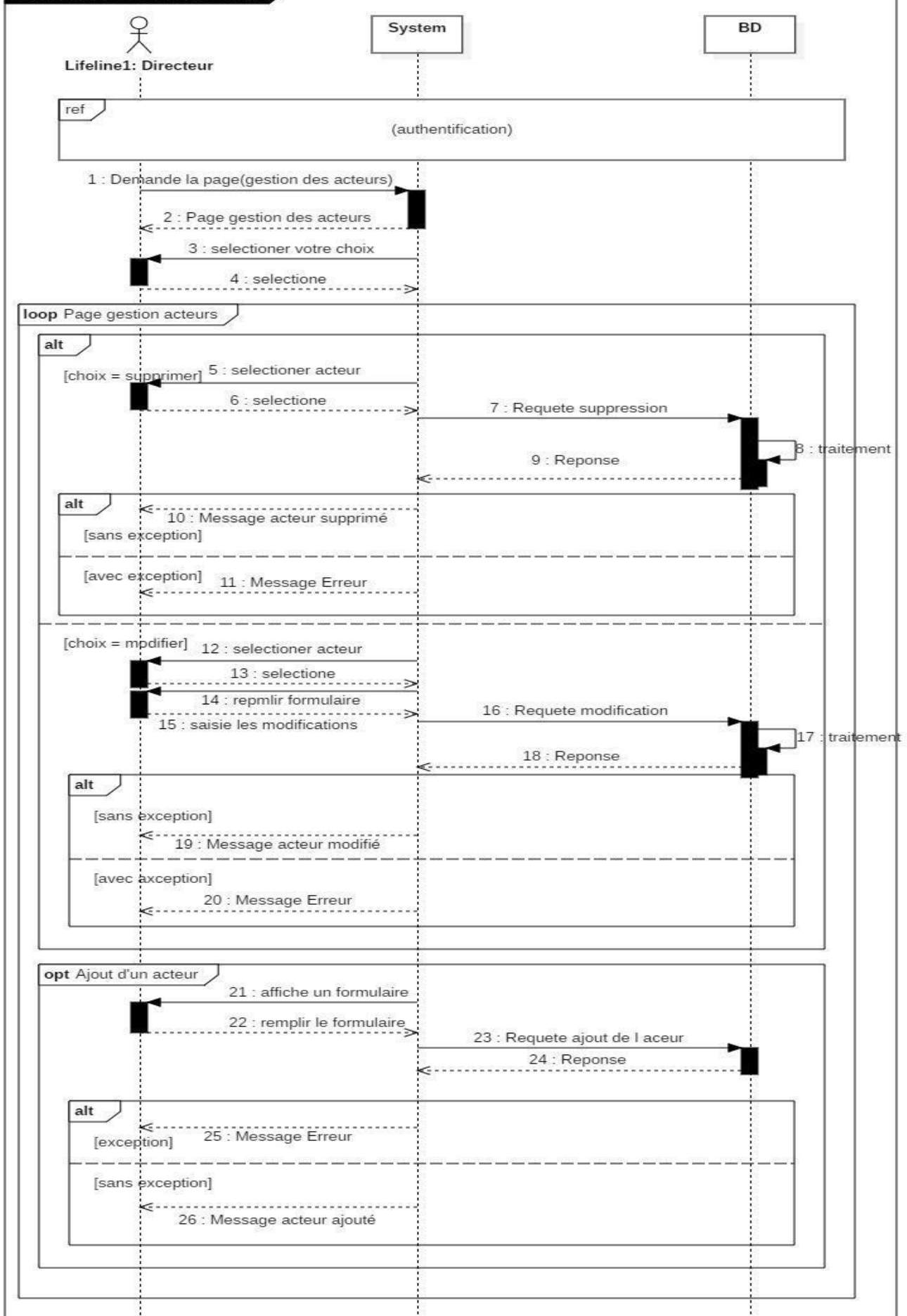


Figure 2.9: Diagramme de séquences (La gestion des acteurs)

● **Diagramme de séquence «gestion des acteurs**

| | |
|---------------------|---|
| But | Permettre au directeur de gérer les acteurs (ajout, suppression, modification) |
| Acteur | Directeur |
| Scénario normal | <ul style="list-style-type: none"> -Le directeur fait l'authentification -Le système redirige le directeur vers sa page -Le directeur choisit la gestion des acteurs -Il peut modifier un acteur avec le remplissage de son formulaire -il peut ajouter un nouvel acteur ou le supprimer -un message de succès sera affiché |
| Scénario alternatif | <ul style="list-style-type: none"> -pour la suppression et la modification si l'acteur n'existe pas dans la BD le système affiche un message d'erreur -pour l'ajout si la matricule de l'acteur saisi existe déjà, le système affiche un message d'erreur -Les informations saisies sont incorrectes ou le formulaire n'est pas rempli complètement -un problème de connexion le système affiche : «Echec » |

2.4.2.7. La gestion des employés

Cette tâche concerne le chef animalier, après l'authentification, il demande la page de gestion des employés et saisie son choix (ajout, suppression, modification), pour les trois cas le système vérifié d'abord l'existence de l'employé dans la BD, après il exécute la tâche et affiche un message de succès sinon il affiche un message d'échec.

● **Diagramme de séquence «gestion employé»**

| | |
|---------------------|---|
| But | Permet au chef animalier de géré les employés |
| Acteur | chef animalier |
| Scénario normal | <ul style="list-style-type: none"> -Le chef animalier fait l'authentification -Le système redirige le chef animalier vers sa page -Il choisit la gestion des employés -Il peut modifier un employé avec le remplissage de son formulaire -Il peut ajouter un nouvel employé ou le supprimer -un message de succès sera affiché |
| Scénario alternatif | <ul style="list-style-type: none"> - Pour la suppression et la modification si l'employé n'existe pas, le système affiche un message d'erreur. - Les informations saisies sont incorrectes ou le formulaire n'est pas rempli complètement. - pour l'ajout si la matricule de l'employé saisi existe déjà, le système affiche un message d'erreur - le chef quitte le site ou annule l'opération. - un problème de connexion, le système affiche message d'erreur |

interaction La Gestion des employés

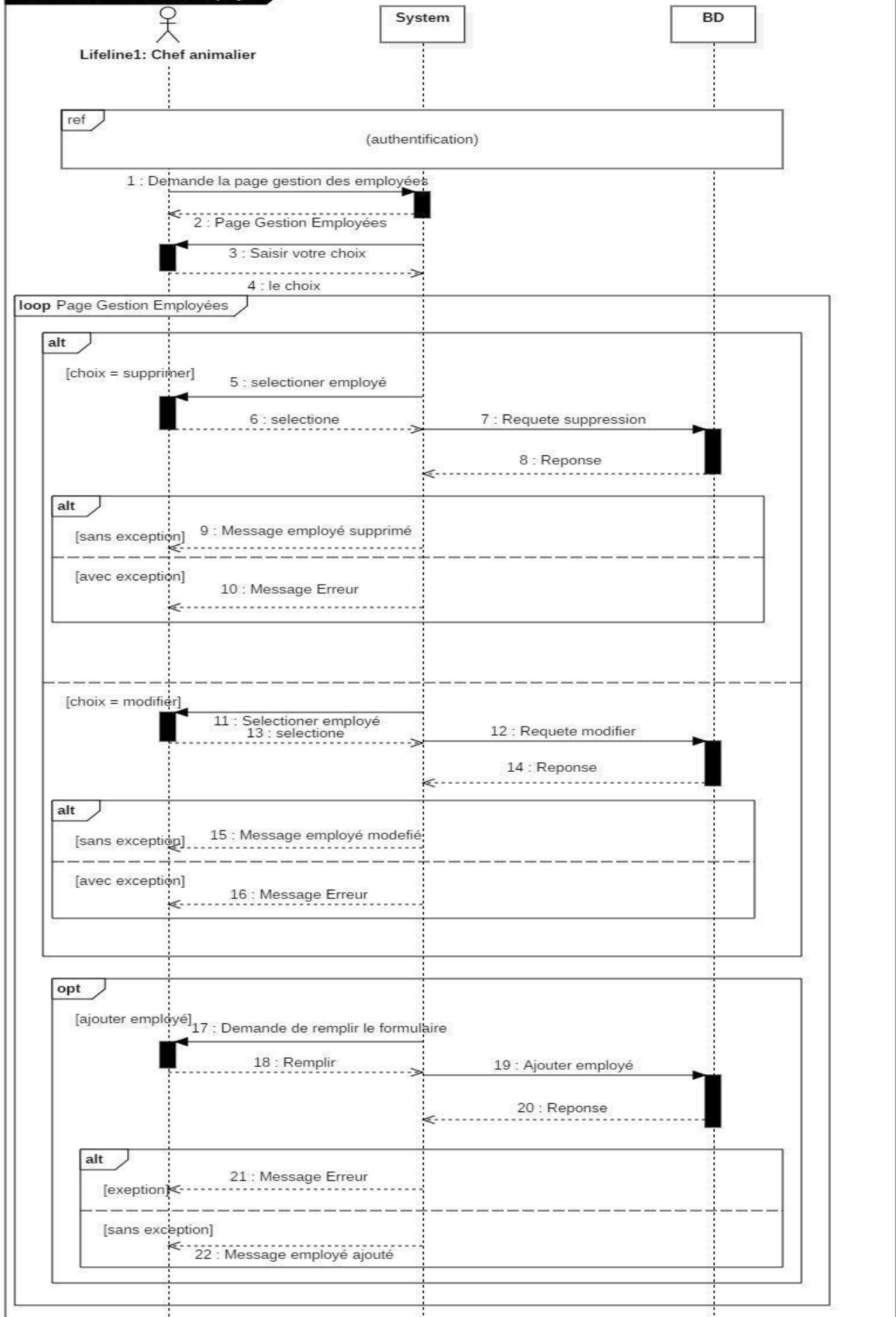


Figure 2.10: Diagramme de séquences (La gestion des employés)

2.4.2.8. La recherche d'un employé

Le chef animalier est le responsable de cette tâche. En effet, si l'employé existe dans la base de données, ses informations seront affichées, sinon le système indique que l'employé n'existe pas.

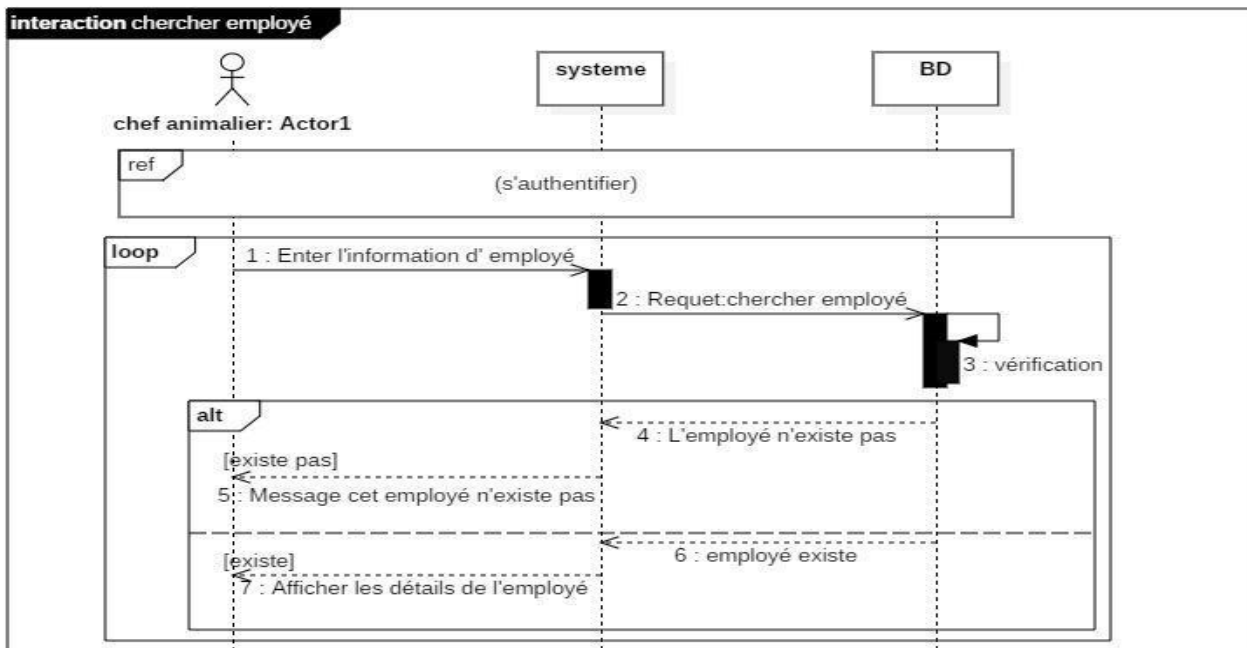


Figure 2.11 : Diagramme de séquences (La recherche d'un employé)

● Diagramme de séquence «rechercher d'un employé»

| | |
|---------------------|---|
| But | Permettre au chef animalier de rechercher un employé dans la BD. |
| Acteur | chef animalier |
| Scénario normal | -Le chef animalier fait l'authentification, le système lui redirige vers sa page -Il lance la recherche (entre l'information de l'employé) -Le système affiche les détails de l'employé |
| Scénario alternatif | -Saisie une information qui n'existe pas, le système va afficher un message d'erreur : « cet employé n'existe pas » -un problème de connexion le système affiche : «Echec » |

2.5. Diagramme de classe

2.5.1. Définition

Le diagramme de classes est le point central dans un développement orienté objet. Pour cela, il faut identifier les classes ainsi que, les associations pertinentes entre celles-ci. Les cardinalités (ou multiplicités) fournissent une indication sur le nombre d'objets d'une même classe participant à l'association. Elles sont extraites des règles de gestion du système étudié données. La notation des cardinalités est la suivante :

- 1 : Obligatoire (un et un seul).
- 0..1 : Optionnel (0 ou 1).
- 0..* ou * : Quelconque.
- N...* : Au moins N.

2.5.2. Représentation de diagramme de classe

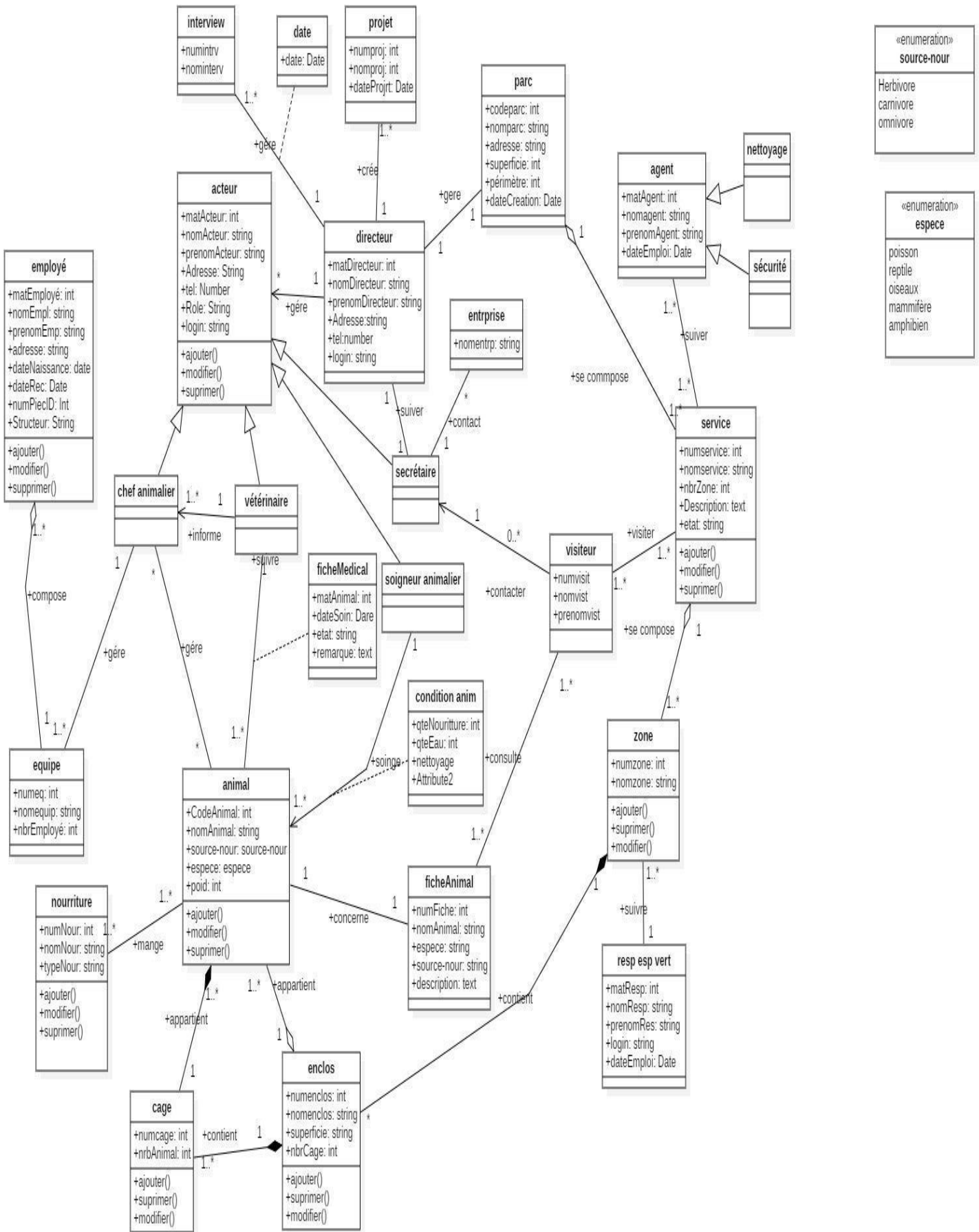


Figure 2.12: Diagramme de classe

2.5.3. Les règles de gestion

RG1 : Un acteur est ajouté par un seul directeur.

RG2 : un acteur peut être soit un chef animalier, vétérinaire ou soigneur animalier.

RG4 : Un directeur peut créer plusieurs projets, un projet est créé par un seul directeur.

RG4 : Un chef animalier peut gérer un ou plusieurs équipes, une équipe est gérée par un chef animalier.

RG6 : un employé peut gérer un ou plusieurs animaux, un animal peut être géré par un ou plusieurs employés.

RG7 : un vétérinaire peut suivre plusieurs animaux, un animal est suivi par un seul vétérinaire.

RG8 : Un visiteur peut visiter un ou plusieurs services qui contiennent plusieurs zones.

RG9 : Une zone peut contenir un ou plusieurs enclos, un enclos appartient à une seule zone.

RG10: Un enclos contient plusieurs cages, une cage appartient à un seul enclos.

RG11 : Une cage peut contenir plusieurs animaux.

2.6. Modèle relationnel

2.6.1. Définition

Ce modèle trie les données dans des tables, appelées aussi des relations, dont chacune se compose de colonnes et de lignes. Chaque colonne contient un attribut de l'entité en question.

L'ensemble de ces attributs est appelé domaine. La clé primaire est constituée d'un ou de plusieurs attributs. On peut y faire référence dans d'autres tables, elle devient alors une clé étrangère.

- **les clés primaires** : une clé primaire d'une relation est simplement un identificateur unique de cette relation.
- **les clés étrangères** : ce sont les attributs qui font référence à une ligne dans une autre table.
- **les clés composées** : ce sont la combinaison des clés des classes participantes.

2.6.2. Les règles de passage :

- **Règle 1** : Toute classe devient une relation dans laquelle :
 - les attributs de relation constituent les attributs de la classe.
 - Si aucun des attributs de la classe ne peut jouer le rôle d'un identifiant, on ajoute un qui sera la clé primaire de la relation.
- **Règle 2** : Une association 1 à 1 se traduit en portant dans une des relations la clé primaire de l'autre. L'attribut ainsi rajouté est appelé «clé étrangère».
- **Règle 3**: Une association * à 1 se traduit en migrant la clé de la classe où la multiplicité de son côté est à 1 vers l'autre classe en clé étrangère.
- **Règle 4** : Une relation * à * se traduit en :
 - Créant une relation particulière qui contient comme attribut les identifiants des deux classes associées (après l'application de la règle 1).
 - Les attributs constituent à deux la clé primaire de la relation.

- Ils sont individuellement la clé étrangère.
- Ajoutant là où les éventuelles propriétés de l'association à cette relation (le cas d'une classe d'association).

➤ **Règle 5:** Les relations d'héritages dans le modèle objet ne sont pas interprétées dans le modèle relationnel, il existe deux solutions :

- Toutes les classes filles sont regroupées dans une même relation, et les attributs sont rajoutés pour distinguer entre les filles.
- Soit chaque file sera interprétée par une relation et dans ce cas il faut prendre en compte les attributs et les relations de la classe mère qui les possède

En appliquant ces règles de transformation sur notre diagramme de classe vers le modèle relationnel, nous avons abouti le à schéma relationnel suivant :

- chefAnimalier(matActeurC*, nomActeur, prenomActeur, nomUtilisateur, role, #matActeurV)
- equipe(numeq*, nomeq, nbrEmployé, #matActeurC)
- vétérinaire(matActeurV*, nomActeur, prenomActeur, adresse, role)
- animal(matAnimal*, nomAnimal, source-nour, espece, poids, #matActeurV, numEnclos, #matActeurS, #numcage, #numFiche)
- nourriture(NumNour*, NomNour, TypeNour)
- cage(numcage*, nbranimal, #numenclos)
- enclos(numenclos*, nomenclos, superficie, nbrcage, #numzone)
- zone(numzone*, nomzone, #numservice, #matResp)
- service(numservice*, nomservice, nbrZone, description, etat, #codeparc)
- serviceVisiteur(numservice*, numvisiteur*)
- entreprise(nomentreprise*, #numservice)
- agent(matAgent*, nomAgent, prenomAgent, dateRec)
- serviceAgent(#matAgent, #numservice)
- parc(codeparc*, nomparc, adresse, superficie, périmetre, dateCreation, #matDirecteur)
- soigneurAnimalier(matActeurS*, nomActeur, prenomActeur, login, adresse, role)
- directeur(matDirecteur*, nomDirecteur, prenomDirecteur, loginDirecteur, #matActeurSe)
- Projet(numProjet*, nomProjet, dateProjet, # matDirecteur)
- Interview(numInterview*, nom Interview, # mat Directeur)
- Employé(matEmployé*, nomEmployé, prenomEmployé, dateNaiss, dateRec, numPieID, #numEquip)
- animalEmployé(matEmployé*, matAnimal*)

2.7. Codification proposé

2.7.1. Pour l'animal

| Z1 | Z2 | Z3 |
|----|----|----|
|----|----|----|

➤ Z1 : 3 positions pour l'espèce de l'animal :

- Poisson : POI
- Reptile : REP
- Oiseau : OIS
- Mammifère : MAM
- Amphibie : AMP

➤ Z2 : 2 positions pour l'année d'arrivé :

- 2021—>21, 2019 —>19

➤ Z3 : 5 positions pour un numéro séquentiel

Exemple : MAM2000022, OIS1900111

2.7.2. Pour l'employé

| Z1 | Z2 | Z3 |
|----|----|----|
|----|----|----|

➤ Z1 : 3 positions pour les 3 premières lettres de son nom

➤ Z2 : 2 positions pour l'année d'arrivé

➤ Z3 : 2 positions pour spécifier sa fonctionnalité :

- Chef animalier : CA
- Vétérinaire : VE
- Soigneur animalier : SA
- Responsable des espaces verts : RV
- Agent de nettoyage : AN
- Agent de sécurité : AS

Exemple : FER18VE, CHA17CA

2.8. Conclusion

Nous avons présenté dans ce chapitre une modélisation de notre système en utilisant trois types de diagrammes UML, commençant par le diagramme de cas d'utilisation (use-case) qui nous a permis de montrer les fonctionnalités de notre application associées à chacun des acteurs recensés. Ensuite, nous avons enchaîné avec quelques diagrammes de séquences relatives aux cas d'utilisation, en montrant des scénarios qui illustrent l'interaction entre l'utilisateur, les interfaces de l'application et la base de données. Finalement, nous avons présenté le diagramme de classes qui a exposé la structure statique de notre application, sans oublier le schéma relationnel des données, qui nous permet de créer la base de données de notre système.

Chapitre3 : Réalisation

3.1. Introduction

Ce chapitre représente la dernière étape de notre projet, qui consiste à implémenter les différentes fonctionnalités du système.

Nous allons présenter ce chapitre sur deux parties, la première partie, présentera les différents outils et logiciels utilisés pour le développement de l'application et la création de la base de données et le serveur utilisé. Tandis que la 2ème partie présentera le test de l'application et les résultats d'exécution illustrée par des captures d'écrans.

3.2. Environnement de travail

3.2.1 Environnement matériel

Pc1: Lenovo, Intel(R) Core(TM) i3-6006U 4Go RAM

Pc2: DELL, Intel Core i5-7200U 8Go RAM

3.2.2. Environnement logiciel

Système d'exploitation : Windows10

L'éditeur de texte : Visuel studio code.

3.3. Outils de développement

- Angular : est un Framework de développement d'applications web qui permet de créer des applications dynamiques complètes.
- TypeScript : est un langage primaire pour le développement d'applications JavaScript utilisé par Angular. C'est un super-ensemble de JavaScript avec un support de conception-temps pour la sécurité de type et l'outillage.
- Angular Material : est une bibliothèque de composants d'interface utilisateur (IU) que les développeurs peuvent utiliser dans leurs projets Angular pour accélérer le développement d'interfaces utilisateur élégantes et cohérentes.
- Bootstrap un framework de frontend gratuit, c'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs.
- HTML/CSS : Le HyperText MarkupLanguage, généralement abrégé HTML ou, dans sa dernière version, HTML5, est le langage de balisage conçu pour représenter les pages web. Et le CSS permet de gérer l'apparence de la page web (agencement, positionnement, décoration, couleurs, taille du texte...). Ce langage est venu compléter le HTML.
- Flask : Ce dernier est un micro Framework de développement Web open source en Python, il a pour but de rendre le développement Web simple et rapide.



- **Json:** JavaScript Object Notation est une extension simple qui ajoute un meilleur support JSON à l'application Flask. Elle aide à gérer les demandes basées sur **JavaScript**, il est un format largement utilisé pour échanger des données sur le Web.



- **REST-API :** signifie Transfert d'Etat Représentatif. C'est un type d'API qui représente un style d'architecteur pour la conception d'application connectées.



Il utilise un HTTP simple pour rendre la communication entre les machines possibles.

Ainsi, au lieu d'utiliser une URL pour manipuler certaines informations utilisateur, REST envoie une requête HTTP comme GET, POST, DELETE, etc. à une URL pour manipuler des données.

- **PhpMyAdmin :** est une application Web de gestion pour le système de gestion de base de données. MySQL réalisée principalement en PHP. Il s'agit de l'une des plus célèbres Interfaces pour gérer une base de données MySQL sur un serveur PHP. Cette interface pratique permet d'exécuter, très facilement, des requêtes SQL. Ce système permet de sauvegarder commodément une base de données sous forme de fichier SQL.



3.4. Interface de l'application

Dans ce qui suit, nous allons exposer les différentes situations que peuvent rencontrer les utilisateurs et les visiteurs sur notre application web :

3.4.1. LA PAGE D'ACCUEIL

Cette page accueille le visiteur et lui présente le concept du site, au haut de la page il y a un menu principal qui contient :

- Page service-animal
- Page contact
- Page about
- Page d'authentification

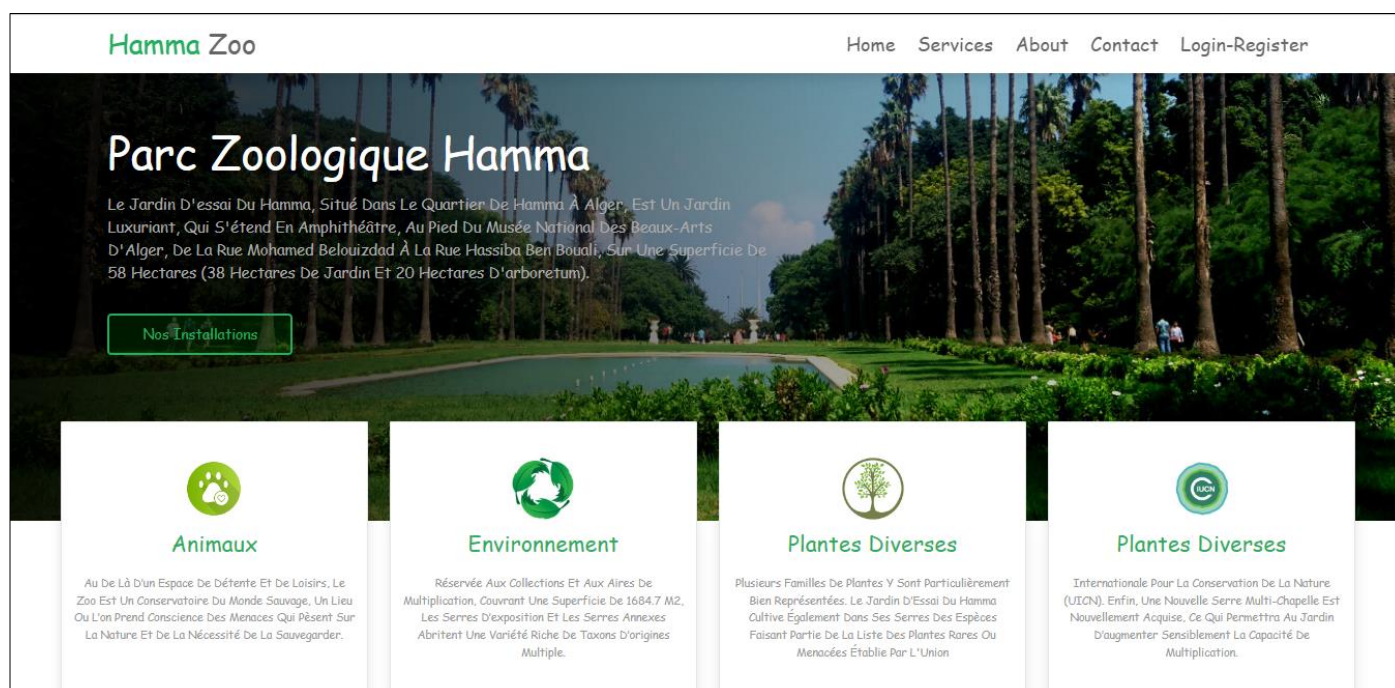


Figure 3.1 : Interface d'accueil

➤ ESPECE D'AUTHENTIFICATION

➤ Login

L'utilisateur doit se connecter afin d'effectuer les taches spécifiées par l'admin, en plus de ce dernier. Il existe 3 types d'utilisateurs :

- Chef animalier
- Soigneur animalier
- vétérinaire

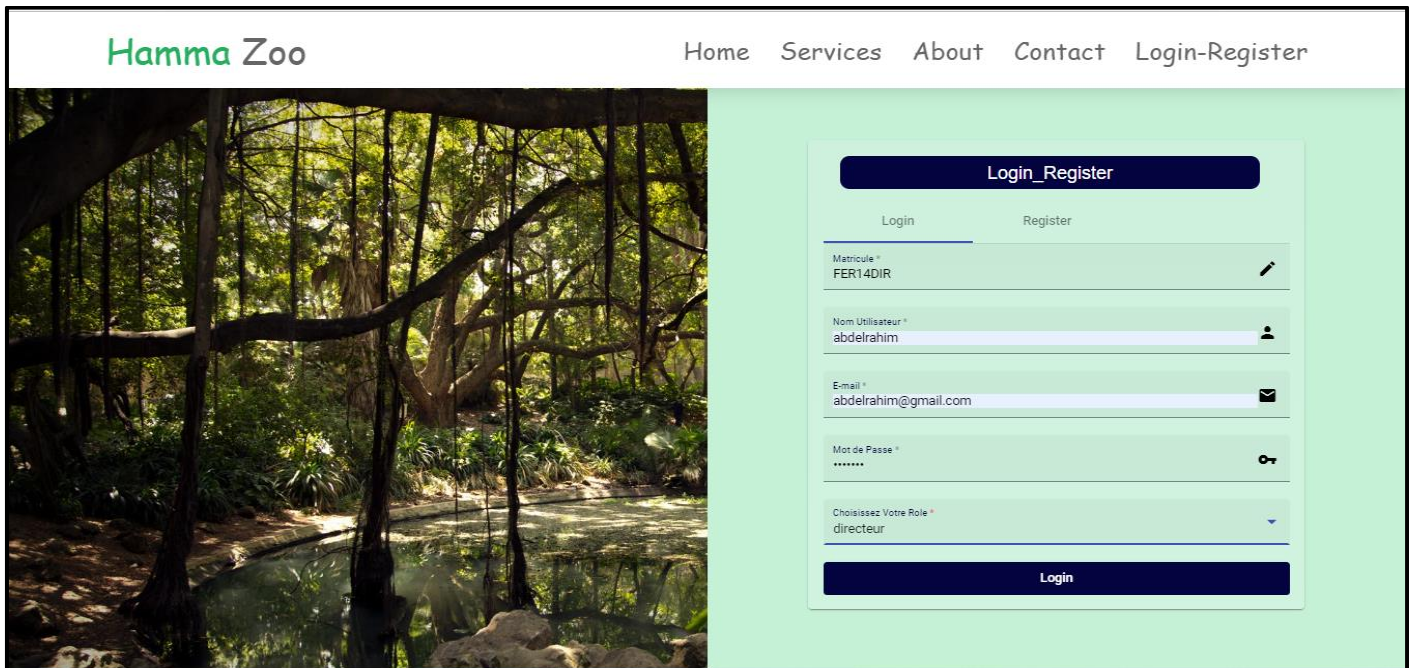


Figure 3.2 : Interface de connexion (login)

3.4.2. Les interfaces des acteurs

Chaque acteur dispose son propre espace, qui contient la page Dashboard et un menu principal situé sur l'extrémité gauche de chaque page. On trouve le nom et le rôle de l'utilisateur connecté, des boutons pour accéder aux taches, ou le profil de cet acteur. En haut de la page, se trouve un bouton de déconnexion pour que l'acteur ferme sa session et revienne à la page de login.



Figure 3.3 : Interface d'un acteur

3.4.2.1. L'interface de directeur

Le directeur peut gérer les acteurs, les services, et les activités, et de consulter son profil.

➤ Consulter le profil

The screenshot shows the 'Hamma Zoo' application interface. On the left is a dark sidebar with a user profile icon and the name 'abdelrahim Directeur'. The main content area has a light gray background. At the top left of this area is a profile card for 'abdelrahim ferkous' with a placeholder image and an 'About' section. To the right is a 'Personal Details' form with fields for First Name, Last Name, Email, Phone, Role, Address (Street, City), and buttons for 'Update' and 'Ajouter'.

| Field | Value |
|------------|----------------------|
| First Name | abdelrahim |
| Last Name | ferkous |
| Email | abdelrahim@gmail.com |
| Phone | 88822 |
| Role | directeur |
| Street | alger |
| City | kkkda |

Figure 3.4 : Interface de profil d'un acteur

➤ La gestion des acteurs :

Cette page affiche une liste d'acteurs avec leurs informations (Matricule, Nom, Prénom, Adresse, numéro de téléphone, rôle), ainsi que, la possibilité de la recherche dynamique.

The screenshot shows the 'Hamma Zoo' application interface for 'La Gestion Des Acteurs'. The page has a sidebar with navigation links (Dashboard, Gérer Acteur, Gérer Activité, Gérer Service, Profil) and a main content area. At the top of the main area is a yellow header bar with the title 'La Gestion Des Acteurs' and a green 'Ajouter Acteur' button. Below the header is a table with 7 columns: Matricule, Nom, Prenom, Adresse, Tel, role, and Action. The table contains 5 rows of actor data. At the bottom of the table is a pagination bar showing 'Items per page: 5' and '1 - 5 of 15'.











| Matricule | Nom | Prenom | Adresse | Tel | role | Action |
|-----------|-----------|------------|---------|---------|--------------------|---|
| ALL20SOI | allach | ikrame | bijaya | 444412 | soigneur animalier |   |
| BEN09CHE | bensallam | amine | hidra | 746208 | chef animalier |   |
| FER14DIR | ferkous | abdelrahim | alger | 8257528 | directeur |   |
| FER22VE | ferkous | sarah | alger | 8627 | vétérinaire |   |
| GG1919 | hayate | jaddi | tindouf | 12345 | chef animalier |   |

Figure 3.5 : Interface de gestion des acteurs

➤ Ajouter un nouvel acteur

La «Figure 3.6 » représente l'interface permettant d'ajouter un nouvel acteur en remplissant un formulaire sans aucune erreur.

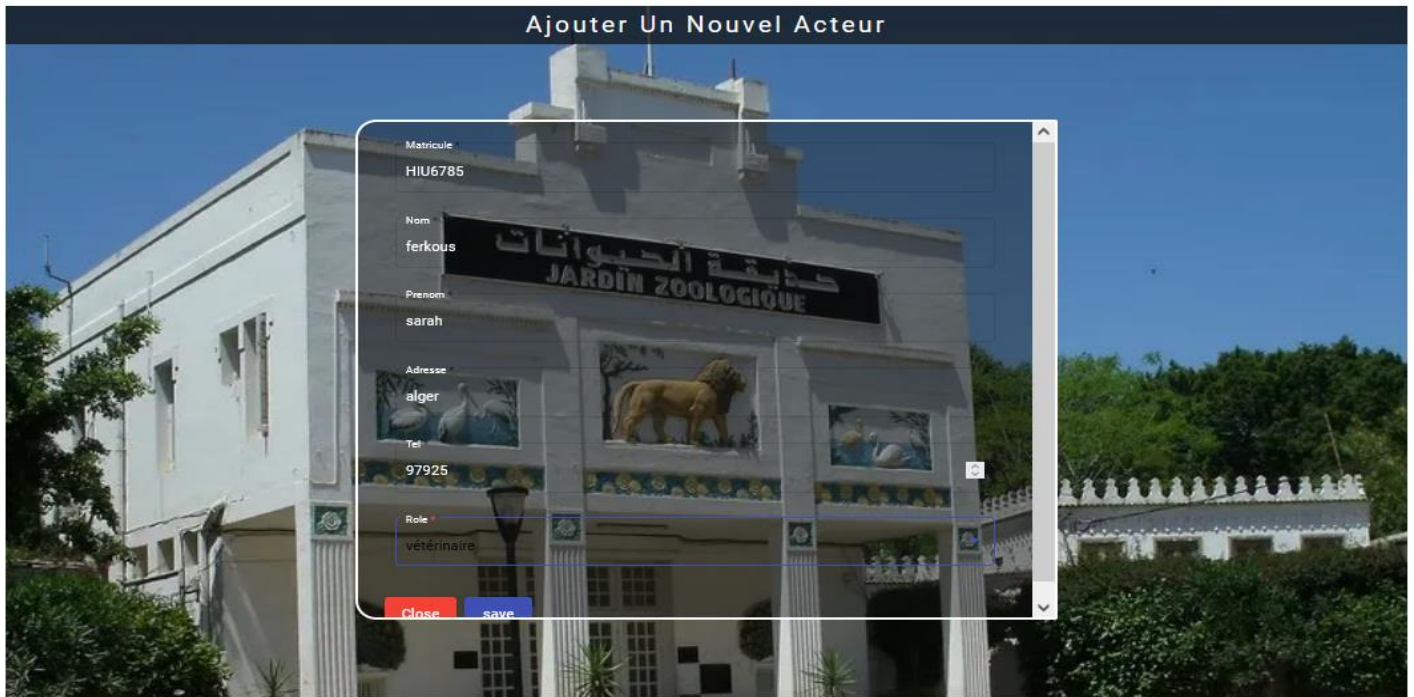


Figure 3.6 : Interface d'ajouter un nouvel acteur

➤ La suppression d'un acteur :

Lorsque le directeur supprime un acteur, ce dernier perd son compte.

➤ La gestion des activités :

Le directeur peut ajouter, modifier ou supprimer une activité.

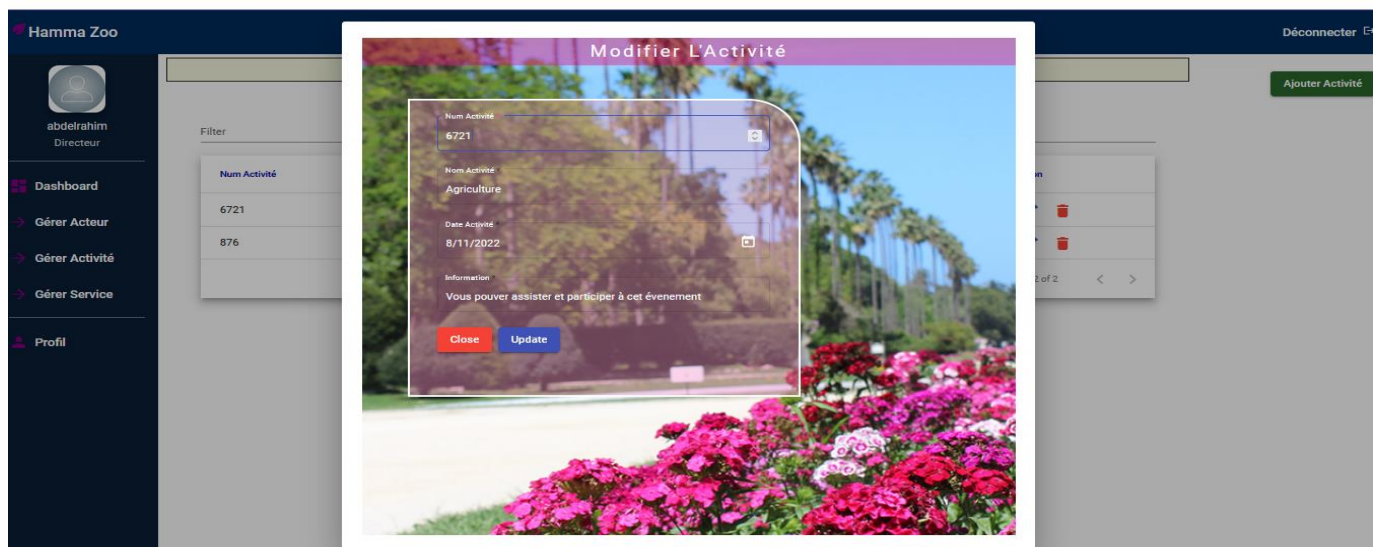


Figure 3.7 : Interface de la modification d'une activité

➤ La gestion des services :

Cette page affiche une liste des services du parc avec leurs informations et offre la possibilité d'ajouter, supprimer ou de modifier un service.

3.4.2.2. L'interface de chef animalier

Le chef animalier peut gérer les animaux, les employés, et de consulter son profil.

➤ Gérer animal :

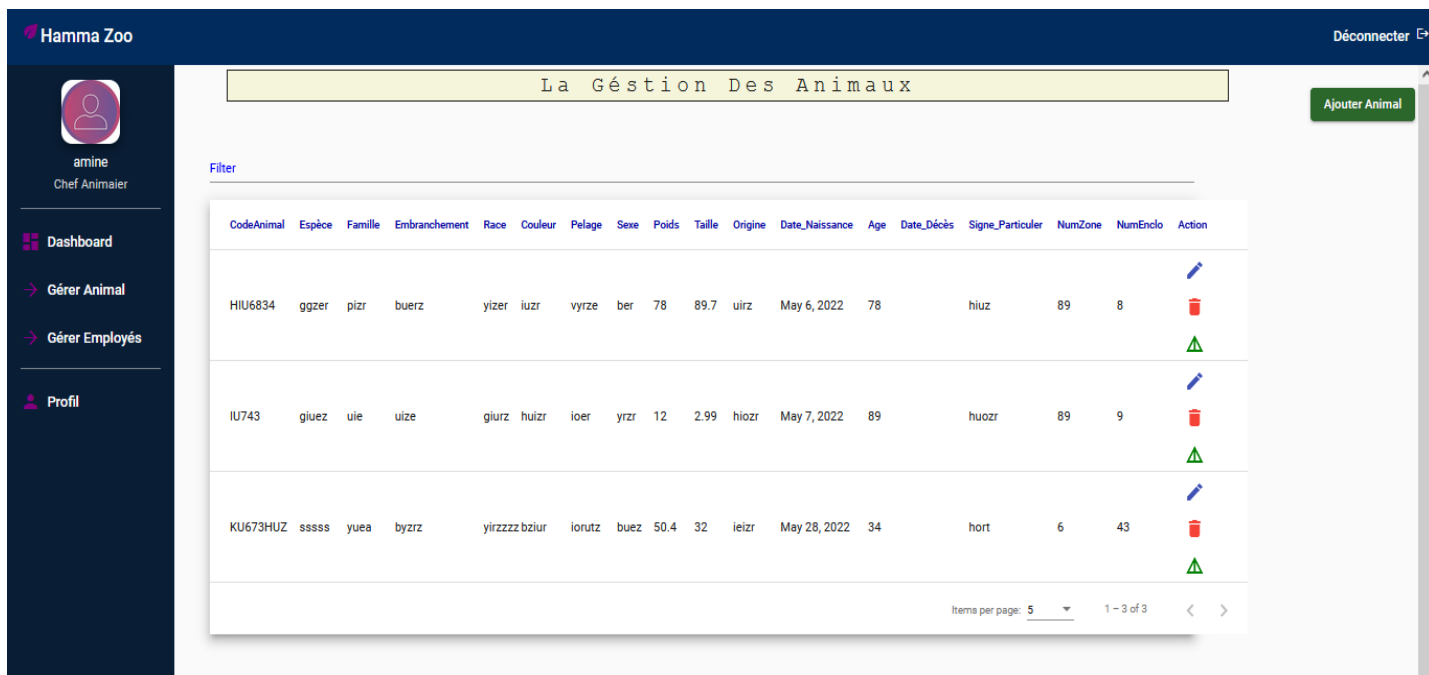


Figure 3.8 : Interface de la gestion des animaux

Cette page affiche une liste des animaux avec leurs informations nécessaires et offre la possibilité d'ajouter, supprimer ou de modifier un animal, la recherche dynamique des animaux, et aussi de consulter les détails d'un animal envoyé par le vétérinaire et le soigneur animalier.

➤ Consulter les détails d'un animal :

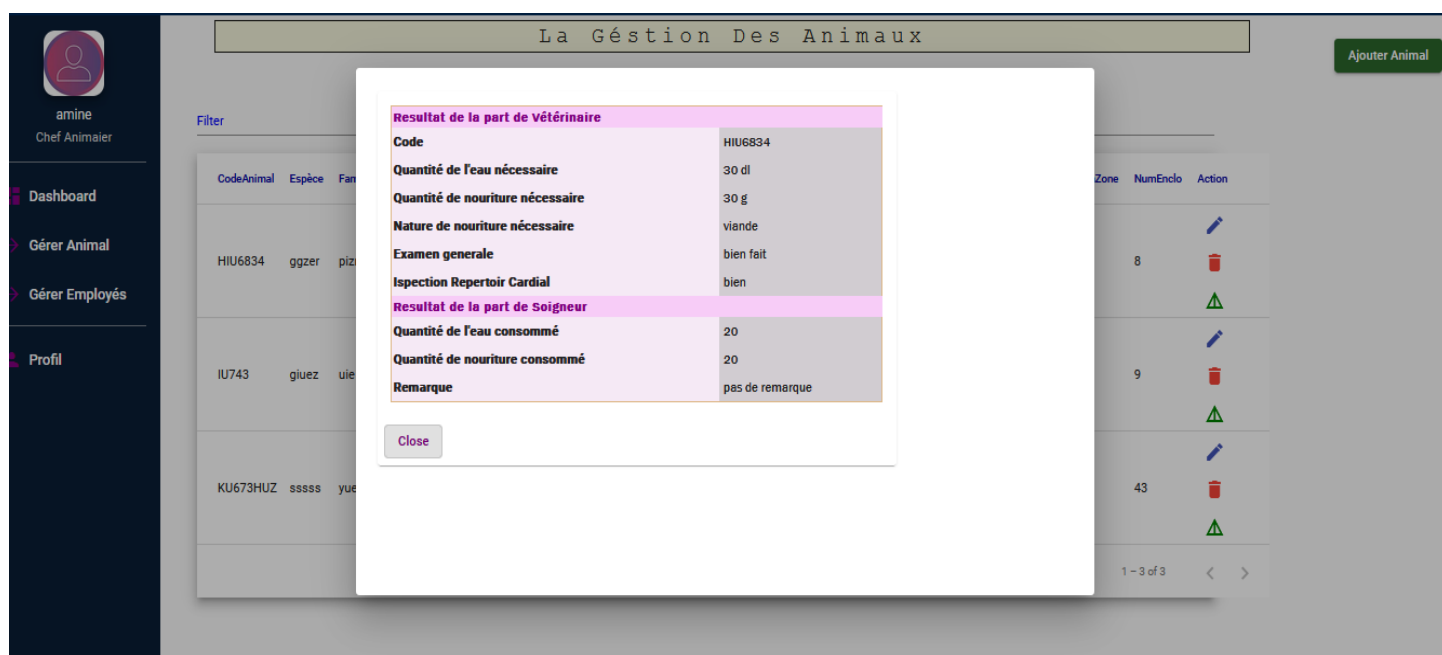


Figure 3.9 : Interface de détail d'un animal(chef animalier)

➤ Gérer employés

Cette interface contient une liste des employés avec leurs informations nécessaires et offre la possibilité d'ajouter, supprimer ou de modifier un employé, et la recherche dynamique des employés.

➤ L'ajout d'un nouvel employé



Figure 3.10 : Interface d'ajouter un nouveau employé

S'il y a une erreur dans un champ du formulaire, le système affiche un message qui explicite l'erreur, sinon le système affiche un message qui confirme l'ajout.

3.4.2.3. L'interface de vétérinaire

Le vétérinaire peut gérer les états des animaux et de consulter les détails sur les animaux.

➤ Gérer L'Etat d'animal :

Le vétérinaire est le responsable de cette tâche, le chef animalier ajoute un nouvel animal qui sera automatiquement affiché dans l'interface de vétérinaire. Alors, le vétérinaire sélectionne l'animal puis, remplit ses informations (la quantité d'eau et de nourriture nécessaire pour cet animal ainsi que, l'examen général). Aussi, il peut consulter les détails de chaque animal envoyé par le chef animalier et le soigneur animalier. (regardez la figure 3.12)




Figure 3.11 : Interface de la gestion d'état d'un animal

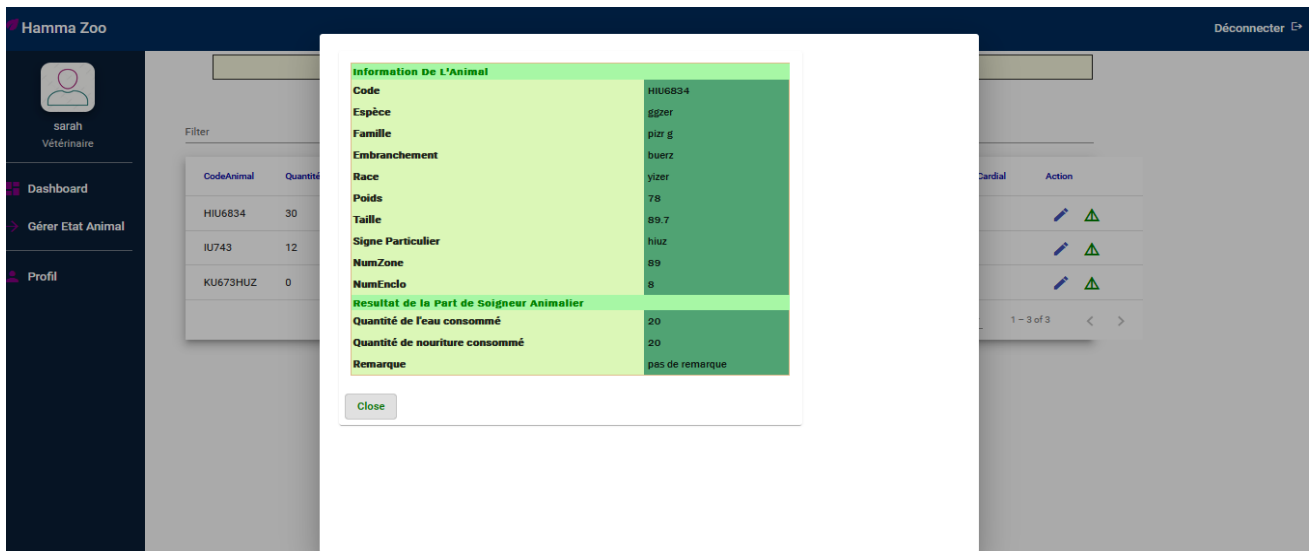


Figure 3.12 : Interface de détail d'un animal (vétérinaire)

3.4.2.4. L'interface de Soigneur

La fonctionnalité de soigneur est la gestion des nourritures (la quantité consommée par jour de chaque animal) et aussi, de consulter les détails de chaque animal.

3.4.3. Page de contact

La présente page permet aux visiteurs de communiquer avec l'administration du parc :

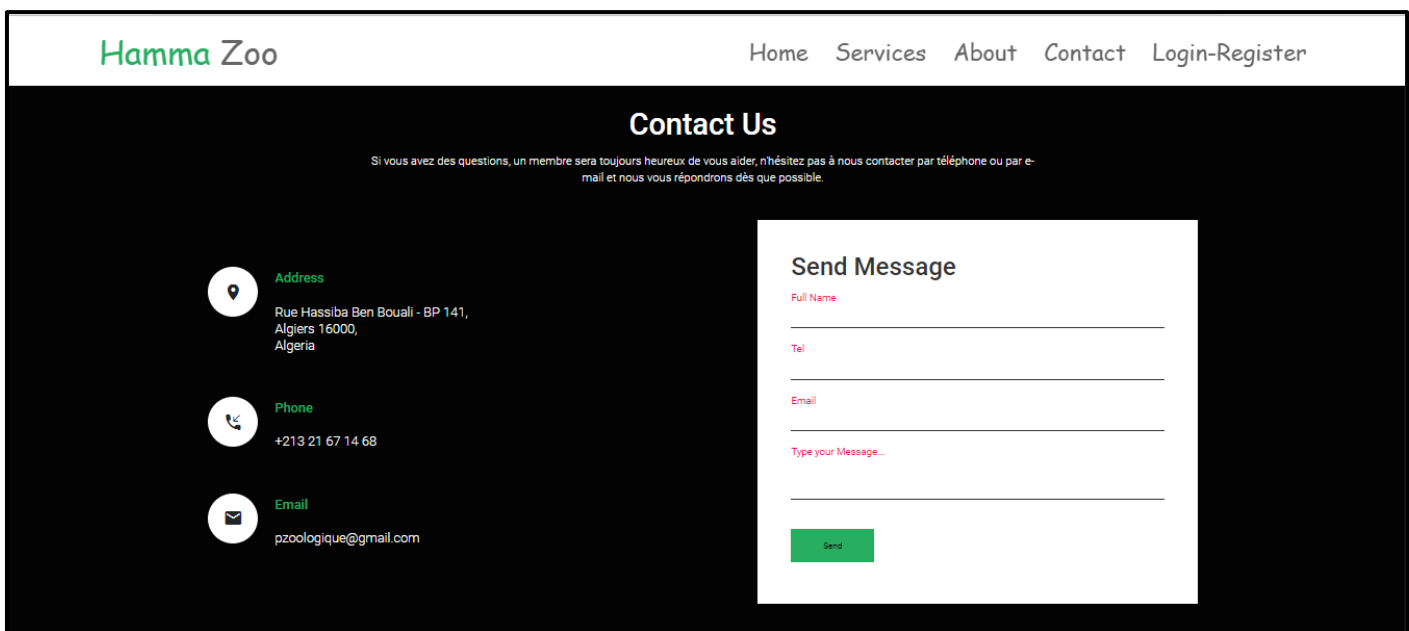


Figure 3.13 : Interface de contact

3.5. Conclusion

Dans ce dernier chapitre, nous avons présenté premièrement, les différents outils et les langages de programmation, que nous avons utilisés pour réaliser notre projet de fin d'études, par la suite, on a terminé par la présentation des différentes interfaces utilisateur de notre système.

Conclusion générale

L'objectif de notre projet était de créer et de développer une application web qui a pour objectif, la gestion du parc zoologique, qui offre la possibilité aux utilisateurs de gérer le fonctionnement du parc. Ainsi, de veiller sur la gestion des différentes espèces animales présentes dans le parc. En appliquant un processus de développement qui a pour but de spécifier les différentes phases d'utilisation et de fonctionnement, en utilisant des outils et des techniques informatiques pour faciliter la tâche d'implémentation.

A travers ce mémoire, nous avons présenté les différentes étapes de la conception et la réalisation de notre application web. Pour cela, nous avons dans un premier temps, présenté le parc, ses composants (administration, espèces animales ...) suivi par la présentation du langage UML et ses différents diagrammes utilisés dans la modélisation de notre système.

Enfin, nous avons présenté dans la phase de réalisation, les différentes fonctionnalités du site à travers des interfaces.

Cette étude a fait l'objet d'une expérience intéressante, elle nous a permis l'acquisition des connaissances multiples dans le domaine informatique et d'améliorer nos compétences dans le domaine de la programmation. Ainsi, cette étude va servir son utilisateur comme étant un outil d'aide à la décision.

Bibliographie

- [1] <https://www.ideematic.com/dictionnaire-digital/application-web>
- [2] <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/quest-ce-quune-application-web-aperçu-des-formats/>
- [3] <https://www.kherdja.com/detail-guide/7150-jardin-d-essai-el-hamma>
- [4] <http://angieangers.centerblog.net/3-objectif-des-parc-animalier>
- [5] <https://www.jaitoutcompris.com/animaux>
- [6] <https://www.elwatan.com/regions/centre/alger/el-hamma-formations-pour-tous-au-jardin-dessai-11-08-2014>
- [7] <https://www.aps.dz/societe/67879-jardin-d-essai-d-el-hamma-plus-de-1-8-millions-de-visiteurs-en-2017>
- [8] [https://fr.wikipedia.org/wiki/Tarzan,_l'homme_singe_\(film,_1932\)](https://fr.wikipedia.org/wiki/Tarzan,_l'homme_singe_(film,_1932))
- [9] <https://fr-fr.facebook.com/jardindessaiduhamma.1832/>
- [10] <http://alarabi.com/2018/04/>
- [11] <https://www.zooactu.com/metiers-du-zoo>
- [12] <https://www.cours-animalia.com/actualites/travailler-dans-un-zoo-161.html>
- [13] https://www.memoireonline.com/07/08/1363/m_conception-realisation-application-gestion-centre-kinesie9.html
- [14] <https://fr.wikipedia.org/wiki/Angular>
- [15] <https://runebook.dev/fr/docs/angular/guide/typescript-configuration>
- [16] <https://www.uptrends.fr/qu-est-ce-que/rest-api>
- [17] <https://pythonhosted.org/Flask-JSON/>
- [18] [https://fr.wikipedia.org/wiki/Bootstrap_\(framework\)](https://fr.wikipedia.org/wiki/Bootstrap_(framework))
- [19] [https://fr.wikipedia.org/wiki/Flask_\(framework\)](https://fr.wikipedia.org/wiki/Flask_(framework))
- [20] <https://auth0.com/blog/creating-beautiful-apps-with-angular-material/>
- [21] <https://fr.wikipedia.org/wiki/PhpMyAdmin>