2025

Franck Giet

Studi Graduate développeur PHP/Symfony

18/01/2025

Projet Zoo Arcadia

****

**SOMMAIRE**

[1 Introduction 3](#_Toc189753120)

[1.1 Présentation personnelle et rôle dans le projet 3](#_Toc189753121)

[2 La liste des compétences mises en oeuvre dans le cadre du projet 4](#_Toc189753122)

[2.1 Activité Type 1 : développer la partie front-end d’une application web ou web mobile sécurisée RNCP37674BC01 4](#_Toc189753123)

[2.2 Activité Type 2 : développer la partie back-end d'une application web ou web mobile sécurisée RNCP37674BC02 5](#_Toc189753124)

[3 L’expression des besoins du projet pour définir les objectifs et les limites du projet 6](#_Toc189753125)

[3.1 Analyse des besoins 6](#_Toc189753126)

[3.1.1 Résumé du projet 6](#_Toc189753127)

[3.1.2 Présentation de l’entreprise et du service 6](#_Toc189753128)

[3.1.3 Environnement technique 7](#_Toc189753129)

[3.1.4 Résumé des besoins 7](#_Toc189753130)

[3.1.5 Besoins identifiés 7](#_Toc189753131)

[3.2 Objectifs principaux du site 7](#_Toc189753132)

[3.2.1 Processus de développement 8](#_Toc189753133)

[4 L’environnement technique : 8](#_Toc189753134)

[4.1 Front-End 8](#_Toc189753135)

[4.2 Back-End 9](#_Toc189753136)

[4.3 Outils de Développement et Déploiement 10](#_Toc189753137)

[4.4 Sécurité 10](#_Toc189753138)

[4.5 Environnement de Travail 11](#_Toc189753139)

[5 L’environnement technique 11](#_Toc189753140)

[5.1 Front-End 11](#_Toc189753141)

[5.2 Back-End 12](#_Toc189753142)

[5.3 Outils de Développement et Déploiement 12](#_Toc189753143)

[5.4 Sécurité 13](#_Toc189753144)

[5.5 Environnement de Travail 13](#_Toc189753145)

[6 Les réalisations permettant la mise en œuvre des compétences 14](#_Toc189753146)

[6.1 Conception de la base de données 14](#_Toc189753147)

[6.1.1 2. Schémas UML, MCD, MPD, MLD (Modélisation de la base de données et des interactions) 14](#_Toc189753148)

[6.1.2 4. Captures de scripts SQL (Requêtes et Base de Données) 14](#_Toc189753149)

[6.2 Développement de l'interface utilisateur 15](#_Toc189753150)

[6.2.1 1. Maquettes (Mobile & Desktop) – Figma ou autre outil utilisé 15](#_Toc189753151)

[6.2.2 3. Captures de codes HTML, CSS, JavaScript 15](#_Toc189753152)

[6.3 Gestion des fonctionnalités dynamiques 16](#_Toc189753153)

[6.4 Mise en œuvre de la sécurité et gestion des accès 16](#_Toc189753154)

[6.4.1 5. Captures des Scripts de Sécurité (Bscript, Sécurité des accès, Validation des formulaires) 16](#_Toc189753155)

[6.5 Déploiement et hébergement 17](#_Toc189753156)

[6.6 Collaboration et gestion de projet 17](#_Toc189753157)

[6.6.1 6. Capture des Outils Utilisés (FileZilla, XAMPP, GitHub) 17](#_Toc189753158)

[**7** **Conclusion** 17](#_Toc189753159)

[**8** **Annexes** 18](#_Toc189753160)

# Introduction

## Présentation personnelle et rôle dans le projet

Bonjour, je m'appelle Franck Giet, et je suis actuellement en formation de reconversion professionnelle en développement web, spécialisé en **PHP et Symfony**. Dans le cadre de cette formation, je réalise ce projet en tant que **développeur full-stack**, avec la responsabilité de concevoir et de développer une application web pour le **Zoo Arcadia**.

1.1 Présentation personnelle et rôle dans le projet

Je m'appelle Franck Giet, et je suis en reconversion professionnelle dans le domaine du développement web, spécialisé en PHP et Symfony. Dans le cadre de ma formation, je réalise ce projet en tant que développeur full-stack, avec l’objectif de concevoir et développer une application web pour le Zoo Arcadia.

Ce projet représente pour moi une opportunité d'appliquer les compétences acquises tout en continuant à apprendre et à progresser dans le développement web.

Mon rôle couvre plusieurs aspects du développement :

**Analyse des besoins** : Comprendre les attentes des différents utilisateurs (visiteurs, vétérinaires, employés, administrateurs) et définir les fonctionnalités clés.

**Conception et intégration** : Réalisation des maquettes et mise en place des interfaces utilisateurs en utilisant HTML, CSS, Bootstrap et JavaScript.

**Développement back-end** : Mise en place de la base de données, gestion des accès via PHP/PDO et premières fonctionnalités interactives. Étant en apprentissage, j’avance progressivement en m’aidant de ressources comme ChatGPT et la documentation officielle.

**Tests et amélioration continue** : Vérification du bon fonctionnement des fonctionnalités développées, correction des erreurs et amélioration des performances en fonction des retours et des besoins.

Grâce à ce projet, je consolide mes compétences en développement full-stack et en gestion de projet, tout en approfondissant mes connaissances en sécurité et bonnes pratiques du web.

# La liste des compétences mises en oeuvre dans le cadre du projet

## Activité Type 1 : développer la partie front-end d’une application web ou web mobile sécurisée RNCP37674BC01

* Installer et configurer son environnement de travail en fonction du projet web ou web mobile,
* Maquetter des interfaces utilisateur web ou web mobile,
* Réaliser des interfaces utilisateur statiques web ou web mobile,
* Développer la partie dynamique des interfaces utilisateur web ou web mobile.

Dans le cadre du projet **Zoo Arcadia**, j’ai travaillé sur la conception et le développement de l’interface utilisateur en utilisant **HTML, CSS et Bootstrap** pour assurer un affichage responsive et une navigation fluide.

Étant encore en apprentissage, j’ai utilisé des ressources en ligne et des outils comme **ChatGPT** pour comprendre et appliquer les bonnes pratiques en front-end. J’ai notamment appris à structurer une page web avec des composants réutilisables et à améliorer l’expérience utilisateur avec des styles CSS adaptés.

J’ai également intégré des **boutons interactifs** et des sections dynamiques en JavaScript pour offrir une meilleure interactivité aux utilisateurs. Par exemple, un système de filtrage permet d’afficher les animaux en fonction de leur habitat.

Grâce à cette partie du projet, j’ai renforcé mes compétences en intégration web et en responsive design, tout en découvrant progressivement l’importance des bonnes pratiques en accessibilité et sécurité.

## Activité Type 2 : développer la partie back-end d'une application web ou web mobile sécurisée RNCP37674BC02

* Mettre en place une base de données relationnelle,
* Développer des composants d'accès aux données SQL et NoSQL,
* Développer des composants métier coté serveur,
* Documenter le déploiement d'une application dynamique web ou web mobile.

Étant au début de mon apprentissage en **back-end**, j’ai commencé à comprendre les bases du développement côté serveur, notamment l’utilisation de **PHP et MySQL** pour stocker et gérer les données de l’application.

Avec l’aide de ressources pédagogiques et d’outils comme **ChatGPT**, j’ai appris à établir une connexion sécurisée avec une base de données en utilisant **PDO**. Pour l’instant, j’ai mis en place les premières requêtes permettant d’afficher les informations des animaux depuis la base de données et de les afficher sur le site.

Je me familiarise progressivement avec des concepts essentiels comme la gestion des utilisateurs, l’authentification et la protection contre les failles de sécurité (injections SQL, XSS). Mon objectif est de continuer à approfondir ces notions pour rendre l’application plus robuste et sécurisée.

En parallèle, j’ai commencé à documenter mon travail pour garder une trace des solutions mises en place et faciliter l’évolution du projet.

Ce projet me permet de mettre en pratique les compétences acquises tout en relevant un défi concret : créer une application web qui incarne les valeurs écologiques du **Zoo Arcadia** tout en répondant aux besoins variés de ses utilisateurs.

Ce projet me permet de consolider ces compétences tout en en explorant de nouvelles, afin de livrer une application web complète et conforme aux attentes du **Zoo Arcadia**.

# L’expression des besoins du projet pour définir les objectifs et les limites du projet

## Analyse des besoins

### Résumé du projet

Le Zoo Arcadia, situé près de la forêt de Brocéliande, souhaite développer une application web pour renforcer sa notoriété et refléter ses valeurs écologiques. Cette application offrira aux visiteurs un accès aux informations sur les animaux, leurs états de santé, les habitats, les services proposés (restauration, visites guidées), ainsi qu’un espace pour laisser des avis.

Elle comportera également des interfaces dédiées aux employés, vétérinaires et administrateurs afin de gérer les données du zoo. Par exemple :

* Les vétérinaires pourront saisir des comptes rendus sur les animaux.
* Les employés géreront les avis des visiteurs et enregistreront les consommations alimentaires des animaux.
* L’administrateur aura un tableau de bord pour superviser l’ensemble des fonctionnalités.

Le projet repose sur une interface ergonomique avec un menu clair, des pages dynamiques et un système de gestion de contenu efficace. Il intégrera des bases de données relationnelles et NoSQL et devra être accessible sur différents supports numériques.

### Présentation de l’entreprise et du service

Le **Zoo Arcadia**, situé près de la légendaire forêt de Brocéliande en Bretagne, est une institution emblématique depuis 1960. Engagé pour le bien-être animal et l’indépendance énergétique, il se distingue par une gestion minutieuse des habitats et des pratiques respectueuses de l’environnement.

Face à l’évolution du numérique, le Zoo Arcadia souhaite améliorer sa visibilité en ligne et proposer une plateforme intuitive répondant aux besoins de ses différents utilisateurs : visiteurs, vétérinaires, employés et administrateurs.

Dans le cadre de mon **examen final encadré par mon école**, j’ai pour mission de concevoir et développer une application web complète qui incarne ces ambitions. Cette plateforme mettra en avant l’engagement écologique du zoo et proposera une navigation fluide et optimisée pour chaque groupe d’utilisateurs.

### Environnement technique

Le projet repose sur les technologies suivantes :

* **Frontend** : HTML5, CSS (Bootstrap), JavaScript
* **Backend** : PHP (PDO) pour la gestion des données
* **Base de données** : MySQL/MariaDB (relationnelle) et MongoDB (NoSQL)
* **Déploiement** : Solutions comme Fly.io, Heroku ou un hébergement mutualisé (PlanetHoster)

Ce projet est une **opportunité unique** pour démontrer mes compétences en développement web tout en contribuant à l’essor numérique d’un acteur local engagé dans l’écologie.

### Résumé des besoins

L’objectif du projet est de créer une application web permettant de **refléter les valeurs écologiques** du zoo tout en offrant une **interface intuitive et fonctionnelle** pour ses différents utilisateurs.

### Besoins identifiés

* **Pour les visiteurs** : Consulter les informations du zoo, les animaux, les horaires, les services, et laisser des avis.
* **Pour les vétérinaires** : Saisir des comptes rendus détaillés sur la santé des animaux.
* **Pour les employés** : Gérer les avis visiteurs et suivre la consommation alimentaire des animaux.
* **Pour l’administrateur** : Superviser l’ensemble des données et des utilisateurs via un tableau de bord.

## Objectifs principaux du site

L’application web doit permettre de :

* **Offrir une interface ergonomique** pour une navigation fluide.
* **Valoriser le zoo et ses spécificités** (expositions, animaux, valeurs écologiques).
* **Améliorer l’expérience utilisateur** grâce à une recherche avancée, des filtres dynamiques et une galerie d’images.
* **Faciliter la gestion des contenus** via un tableau de bord pour les administrateurs et employés.

### Processus de développement

Le projet suivra plusieurs étapes clés :

* **Analyse des besoins** : Identification des attentes des utilisateurs et des fonctionnalités nécessaires.
* **Maquettage** : Création des maquettes interactives pour valider l’interface utilisateur.
* **intégration** : Développement des pages statiques et dynamiques en HTML, CSS et JavaScript.
* **Développement** : Programmation du backend avec PHP/PDO et mise en place des bases de données.
* **Déploiement** : Mise en ligne, tests d’utilisation et ajustements finaux.

# L’environnement technique :

L’environnement technique de ce projet est choisi de manière à assurer une solution web fonctionnelle, évolutive, et performante, tout en prenant en compte les contraintes du projet (**coût, délais,** compatibilité avec différentes plateformes, etc.). Le projet repose sur une architecture **full-stack** avec une partie front-end et une partie back-end, intégrant des technologies modernes et des solutions open-source adaptées aux besoins du Zoo Arcadia.

## Front-End

Pour le développement de l’interface utilisateur, l’objectif est de garantir une expérience fluide et agréable pour tous les utilisateurs, en particulier les visiteurs du zoo. Les choix technologiques suivants ont été retenus :

* **HTML5 et CSS3** : Langages fondamentaux du développement web, utilisés pour la structuration des pages et le design de l’interface. Le choix de HTML5 permet une meilleure gestion des multimédias (audio, vidéo) et des fonctionnalités avancées comme le stockage local.
* **CSS avec Bootstrap** : Framework CSS pour garantir un design responsive et moderne, adapté à tous les types d’écrans (mobiles, tablettes, ordinateurs). Bootstrap permet de gagner en productivité et de s'assurer que l'application est bien optimisée pour tous les supports.
* **JavaScript** : Langage permettant de dynamiser les pages web (animations, interactions utilisateur). Ce choix permet d'ajouter des fonctionnalités telles que des filtres dynamiques et des galeries interactives.
* **React** (ou React Native si une version mobile est envisagée) : Framework JavaScript permettant de développer une interface dynamique et réactive. Cela permettra également une gestion plus performante des états et des interactions avec l’utilisateur.

Ces technologies assurent une grande flexibilité pour concevoir des interfaces interactives, et un design adaptable aux contraintes d’écran des visiteurs.

## Back-End

Le back-end de l'application est conçu pour gérer les demandes des utilisateurs, la gestion des données du zoo (animaux, vétérinaires, avis, etc.), ainsi que la sécurité des échanges d'informations. Pour cela, les technologies suivantes ont été retenues :

* **PHP avec PDO** : Le choix de PHP est motivé par sa robustesse, sa large adoption et sa compatibilité avec les systèmes de gestion de bases de données (SGBD) relationnels comme MySQL/MariaDB. PDO permet une gestion sécurisée des accès aux bases de données via des requêtes paramétrées, protégeant ainsi contre les injections SQL.
* **MySQL/MariaDB** : Ces SGBD relationnels sont utilisés pour gérer les données structurées liées au zoo, comme les informations sur les animaux, les avis des visiteurs, les consultations vétérinaires, etc. Leur performance et leur flexibilité sont idéales pour ce type de projet.
* **MongoDB (NoSQL)** : Pour certaines fonctionnalités nécessitant des données non structurées ou moins rigides, comme les logs d’activité ou les commentaires des visiteurs, MongoDB sera utilisé pour sa flexibilité et sa capacité à gérer des documents variés.

Ce choix d'architecture garantit une séparation claire entre la logique métier (back-end) et l'interface utilisateur (front-end), tout en permettant une gestion efficace des données.

## Outils de Développement et Déploiement

Pour garantir une gestion optimale du projet, un ensemble d'outils de développement et de déploiement est utilisé :

* **Git** (versionnage du code) : Outil essentiel pour la gestion des versions du code, permettant une collaboration efficace et un suivi des évolutions du projet.
* **Visual Studio Code** : Éditeur de code léger et performant, avec des extensions adaptées pour le développement PHP, JavaScript et la gestion de bases de données.
* **Docker** (si nécessaire) : Pour créer un environnement de développement isolé, permettant de tester l’application dans des conditions proches de la production.
* **Solutions de déploiement** :
  + **Heroku / Fly.io / Azure** : Des solutions de cloud pour héberger l’application et garantir sa disponibilité. Ces plateformes permettent une mise en ligne rapide et une gestion simplifiée des ressources.
  + **Planethoster (hébergement mutualisé)** : Solution pour une gestion à moindre coût, idéale pour un projet en phase de développement ou à petite échelle.

## Sécurité

Le projet met également l'accent sur la sécurité des données, particulièrement pour les informations sensibles telles que les comptes utilisateurs, les avis des visiteurs, et les détails médicaux des animaux. Voici les principales mesures de sécurité adoptées :

* **Authentification et autorisation des utilisateurs** : Mise en place de mécanismes de connexion sécurisés avec gestion des rôles (administrateur, vétérinaire, employé, visiteur).
* **Chiffrement des données sensibles** : Utilisation de techniques de chiffrement pour sécuriser les informations personnelles et médicales stockées.
* **Protection contre les attaques courantes** : Mise en place de protections contre les attaques XSS (Cross-site Scripting) et CSRF (Cross-Site Request Forgery), et validation des entrées utilisateur pour prévenir les injections SQL.

## Environnement de Travail

Enfin, l’environnement de développement est configuré en tenant compte des besoins de productivité et de collaboration. L’utilisation de plateformes collaboratives et d’outils de gestion de projet (comme GitHub, GitLab, ou Trello) permet un suivi constant de l’avancement du projet et une répartition claire des tâches.

# L’environnement technique

L’environnement technique du projet a été conçu pour garantir une solution web fonctionnelle, sécurisée et évolutive. L’accent a été mis sur l’utilisation de technologies éprouvées et de ressources open-source adaptées aux besoins du projet. Le choix des outils et des technologies prend en compte les contraintes du projet, telles que les délais et la compatibilité avec les différents navigateurs.

## Front-End

L’objectif est d’offrir une interface utilisateur fluide et agréable pour les utilisateurs du site. Les technologies retenues pour le développement front-end sont :

* **HTML5 et CSS3** : Ces langages permettent une structuration claire des pages et une présentation moderne et optimisée. HTML5 facilite l’intégration de multimédias et de fonctionnalités avancées comme le stockage local.
* **CSS avec Bootstrap** : Utilisation du framework CSS Bootstrap pour garantir un design responsive, permettant une adaptation automatique de l'interface à différents types d’écrans (mobiles, tablettes, ordinateurs).
* **JavaScript** : Ce langage est utilisé pour dynamiser l’interactivité de l’application, en intégrant des fonctionnalités comme des filtres dynamiques, des galeries interactives, et des animations.
* **React** : Bien que prévu pour une évolution future, ce framework JavaScript pourra être utilisé pour rendre l’interface plus dynamique et performante si une gestion avancée des états devient nécessaire.

## Back-End

Le back-end est responsable de la gestion des données et de la logique métier de l’application. Les choix techniques suivants ont été effectués :

* **PHP avec PDO** : PHP est utilisé pour sa robustesse et sa compatibilité avec les bases de données relationnelles comme MySQL/MariaDB. Le recours à PDO garantit des connexions sécurisées à la base de données grâce à des requêtes paramétrées, protégeant contre les injections SQL.
* **MySQL/MariaDB** : Ces systèmes de gestion de bases de données relationnelles sont utilisés pour organiser les données structurées du projet, telles que les informations sur les véhicules, les utilisateurs et les transactions.
* **XAMPP** : XAMPP a été utilisé comme environnement de développement local. Cette solution complète offre un serveur Apache, MySQL/MariaDB, PHP et d'autres outils nécessaires pour développer, tester et déployer l’application.

## Outils de Développement et Déploiement

L’environnement de développement repose sur plusieurs outils qui facilitent la gestion du code, le suivi des versions, et l'hébergement des fichiers :

Dans le cadre du développement de ce projet, des outils collaboratifs ont été utilisés pour garantir une gestion fluide du projet, une communication efficace, ainsi qu’une meilleure gestion des aspects visuels et techniques :

* **FileZilla** : Utilisé pour le transfert de fichiers entre l’environnement local et le serveur distant. FileZilla permet une gestion facile des fichiers sur le serveur d’hébergement.
* **Git** : Le contrôle de version est effectué avec Git, permettant de suivre l’évolution du code, de collaborer efficacement et de maintenir une trace des modifications importantes.
* **Visual Studio Code** : Un éditeur de code léger et performant qui offre une riche extension pour le développement PHP, JavaScript et la gestion des bases de données.
* **ChatGPT** : Un outil d’assistance pour la génération de code, la rédaction de documentation et la résolution de problèmes techniques rencontrés tout au long du projet.

## Sécurité

La sécurité est primordiale pour protéger les données sensibles. Plusieurs mesures ont été mises en place pour sécuriser l’application :

* **Bscript** : Utilisation de Bscript pour l'implémentation des fonctionnalités de sécurité. Bscript permet de renforcer la gestion des connexions utilisateurs, la validation des données, et la protection contre les vulnérabilités courantes telles que les attaques XSS et CSRF.
* **Authentification et autorisation** : Mise en œuvre d'un système d’authentification sécurisé avec des rôles d’utilisateur (administrateur, employé, visiteur), garantissant que seuls les utilisateurs autorisés peuvent accéder à certaines parties de l’application.
* **Chiffrement des données sensibles** : Des techniques de chiffrement sont appliquées pour protéger les données personnelles et sensibles stockées dans la base de données.
* **Validation des entrées utilisateurs** : Pour éviter les injections SQL et d'autres vulnérabilités, toutes les entrées utilisateur sont systématiquement validées et assainies.

## Environnement de Travail

Pour assurer une gestion optimale du projet et une collaboration efficace, des outils de travail collaboratifs ont été utilisés :

* **Figma** : Outil de design collaboratif permettant de créer des maquettes, des prototypes interactifs et de définir les éléments visuels de l'interface utilisateur. Figma est utilisé pour concevoir les wireframes, les interfaces, et les éléments graphiques du site, tout en permettant à l’équipe de collaborer en temps réel sur ces aspects. Cela facilite la validation des choix de design et l'intégration des éléments visuels dans le développement front-end.
* **GitHub / GitLab / Trello** : Ces outils permettent de suivre l’avancement du projet, de collaborer efficacement sur le code, et de gérer les tâches et les responsabilités au sein de l’équipe.
* **Slack / Plateformes de communication** : Des plateformes de communication en temps réel ont été mises en place pour faciliter la collaboration entre les membres de l’équipe, permettant un échange rapide d’informations et une résolution immédiate des problèmes.

# Les réalisations permettant la mise en œuvre des compétences

Cette section présente les étapes et les réalisations spécifiques ayant permis d’appliquer les compétences techniques et fonctionnelles acquises tout au long de la gestion du projet **Zoo Arcadia**. Chaque tâche ou réalisation a contribué à l'acquisition de nouvelles compétences dans différents domaines, allant du développement web à la gestion de la sécurité des données, tout en tenant compte des spécificités et des besoins du projet.

## Conception de la base de données

* **Compétence en gestion de bases de données relationnelles (MySQL/MariaDB)** : La création et la gestion des tables users2, animaux, avis\_visiteurs, et d'autres entités liées à la gestion du zoo ont permis de renforcer ma compréhension des bases de données relationnelles et des bonnes pratiques d’intégrité des données.
* **Mise en œuvre des clés étrangères** : L'intégration des relations entre les différentes tables, telles que les informations sur les animaux, les visites et les avis des visiteurs, a consolidé mes compétences en gestion des relations entre les entités dans une base de données relationnelle.

### 2. Schémas UML, MCD, MPD, MLD (Modélisation de la base de données et des interactions)

🔹 **MCD (Modèle Conceptuel des Données)** – Pour illustrer les relations entre les entités

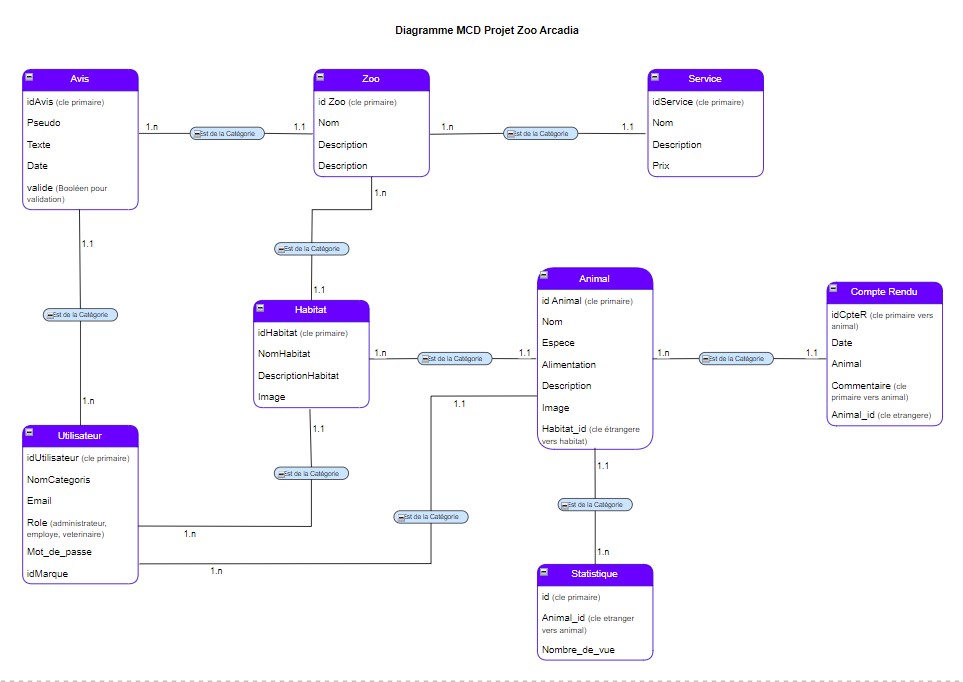


Figure 1 Diagramme MCD du zoo arcadia

🔹 **MPD (Modèle Physique des Données)** – Pour montrer comment la base de données a été implémentée.

🔹 **MLD (Modèle Logique des Données)** – Optionnel, mais utile pour montrer la transition entre MCD et MPD.

🔹 **Diagramme UML (Cas d’utilisation, classes, séquences)**

* Cas d’utilisation : Qui fait quoi dans le site (admin, employé, visiteur, etc.).
* Séquences : Comment se déroule un processus (exemple : ajout d’un avis, achat d’un billet en ligne).

### 4. Captures de scripts SQL (Requêtes et Base de Données)

🔹 **Requêtes SQL** importantes :

* La création des tables users2, animaux, avis\_visiteurs.
* Une requête d’insertion de données.
* Une requête de sélection avec jointure (ex : récupération des avis des visiteurs pour un animal).

## Développement de l'interface utilisateur

* **Compétences en HTML/CSS et JavaScript** : La création de pages web interactives pour le zoo, telles que la galerie des animaux et les fiches d’information détaillées, m'a permis d’appliquer mes connaissances en HTML5 pour structurer le contenu, en CSS3 pour la mise en forme et le design responsive, et en JavaScript pour ajouter des fonctionnalités dynamiques comme les filtres de recherche d'animaux et les interactions avec les utilisateurs.
* **Utilisation de frameworks** : L'intégration des maquettes réalisées sur **Figma** dans le développement front-end avec des outils comme **Bootstrap** a permis de garantir un design réactif et fonctionnel, essentiel pour une bonne expérience utilisateur, notamment pour les visiteurs du zoo.

### 1. Maquettes (Mobile & Desktop) – Figma ou autre outil utilisé

📱 Page d'accueil mobile

📱 Page d’un animal mobile

📱 Formulaire de réservation mobile

🖥️ Page d’accueil desktop

🖥️ Liste des animaux desktop

🖥️ Page de détails d’un animal desktop

### 3. Captures de codes HTML, CSS, JavaScript

🔹 **HTML** – Exemple de structure d’une page importante (ex : carte d’un animal).

🔹 **CSS** – Une feuille de style montrant la gestion responsive ou le design d’un élément clé.

🔹 **JavaScript** – Une fonction clé (ex : filtre dynamique, affichage dynamique des animaux).

## Gestion des fonctionnalités dynamiques

* **Mise en place de filtres dynamiques** : Le développement de filtres interactifs pour la recherche des animaux ou des événements au zoo, ainsi que l'intégration de fonctionnalités de recherche avancée, ont renforcé mes compétences en JavaScript, notamment en matière d’interactions utilisateur et de gestion dynamique des contenus affichés.
* **Gestion des formulaires et des données d’entrée** : La création de formulaires pour l’ajout d’animaux, d'avis des visiteurs ou d'événements m’a permis de travailler sur la validation des données côté serveur, l’insertion dans la base de données et la gestion des retours d’informations.

## Mise en œuvre de la sécurité et gestion des accès

* **Sécurisation des données** : L'utilisation de **Bscript** pour protéger les données sensibles (comptes utilisateurs, avis des visiteurs, informations médicales des animaux) et l'intégration de mécanismes de **gestion des utilisateurs et des rôles** (administrateur, vétérinaire, employé, visiteur) m'ont permis d'approfondir mes connaissances en sécurité informatique et en gestion des accès dans une application web.
* **Prévention des attaques courantes** : La gestion des sessions et la mise en place de protections contre les attaques telles que **les injections SQL**, **le Cross-Site Scripting (XSS)** et **les attaques CSRF** ont permis d’améliorer mes compétences en sécurité des applications web.

### 5. Captures des Scripts de Sécurité (Bscript, Sécurité des accès, Validation des formulaires)

📍 **Où ?** Dans la section **"Mise en œuvre de la sécurité et gestion des accès"**

* 🔹 **Hachage des mots de passe avec Bscript** (ex : partie du script d’inscription/login).
* 🔹 **Vérification et filtrage des entrées utilisateur** (ex : protection contre les injections SQL/XSS).
* 🔹 **Gestion des rôles (admin/employé/visiteur)** avec un extrait de code.

## Déploiement et hébergement

* **Utilisation de XAMPP pour l’environnement local** : La mise en place de l’environnement de développement local avec XAMPP m'a permis de maîtriser l'installation et la configuration des serveurs locaux pour le développement web, en particulier pour tester et valider les fonctionnalités avant la mise en production du site du zoo.
* **Utilisation de FileZilla pour le déploiement** : Le transfert des fichiers vers le serveur de production avec FileZilla m’a donné une expérience pratique de la gestion des fichiers et du déploiement d’une application web en ligne, ce qui est essentiel pour garantir une mise en ligne fiable de la plateforme.

## Collaboration et gestion de projet

* **Utilisation de Git et GitHub** : Le versionnage du code via **Git** et la gestion collaborative avec **GitHub** m'ont permis de gérer les différentes versions du code, de collaborer efficacement avec d'autres développeurs et de suivre l'évolution du projet **Zoo Arcadia**. Cela a renforcé mes compétences en gestion de versions et en travail collaboratif à distance.

### 6. Capture des Outils Utilisés (FileZilla, XAMPP, GitHub)

📍 **Où ?** Dans la section **"Déploiement et hébergement"** et **"Collaboration et gestion de projet"**

* 🔹 **FileZilla** – Transfert des fichiers vers le serveur.
* 🔹 **XAMPP** – Serveur local en fonctionnement.
* 🔹 **GitHub** – Référentiel du projet, historique des commits

1. **Conclusion**

 est une synthèse des points saillants du projet mettant en avant les accomplissements majeurs et les apprentissages tirés de l'expérience

* **Prise de recul** : Ce que tu as appris pendant le projet, les défis rencontrés et comment tu les as surmontés.
* **Apports du projet** : Ce que ce projet t'a apporté professionnellement.
* **Projets futurs** : Ce que tu prévois pour le futur après ce projet.

1. **Annexes**

* Images, code source, diagrammes, etc.

devront contenir les extraits de code, les captures d'écran, les schémas, les diagrammes et tout autre élément qui soutient et enrichit vos explications (la taille de chaque élément dépasse la demi-page).

8 Remerciement (optionnel)