

# **Documentation de gestion de projet pour le site du zoo Arcadia**

## **1. Cas d'utilisation :**

Les cas d'utilisation représentent les différentes interactions entre les utilisateurs et le site du zoo. Chaque scénario est conçu pour comprendre comment les visiteurs et le personnel du zoo interagiront avec le site. Voici quelques exemples de cas d'utilisation :

Visiteur : Consulter les informations sur les animaux, les horaires, les services, laisser un avis.

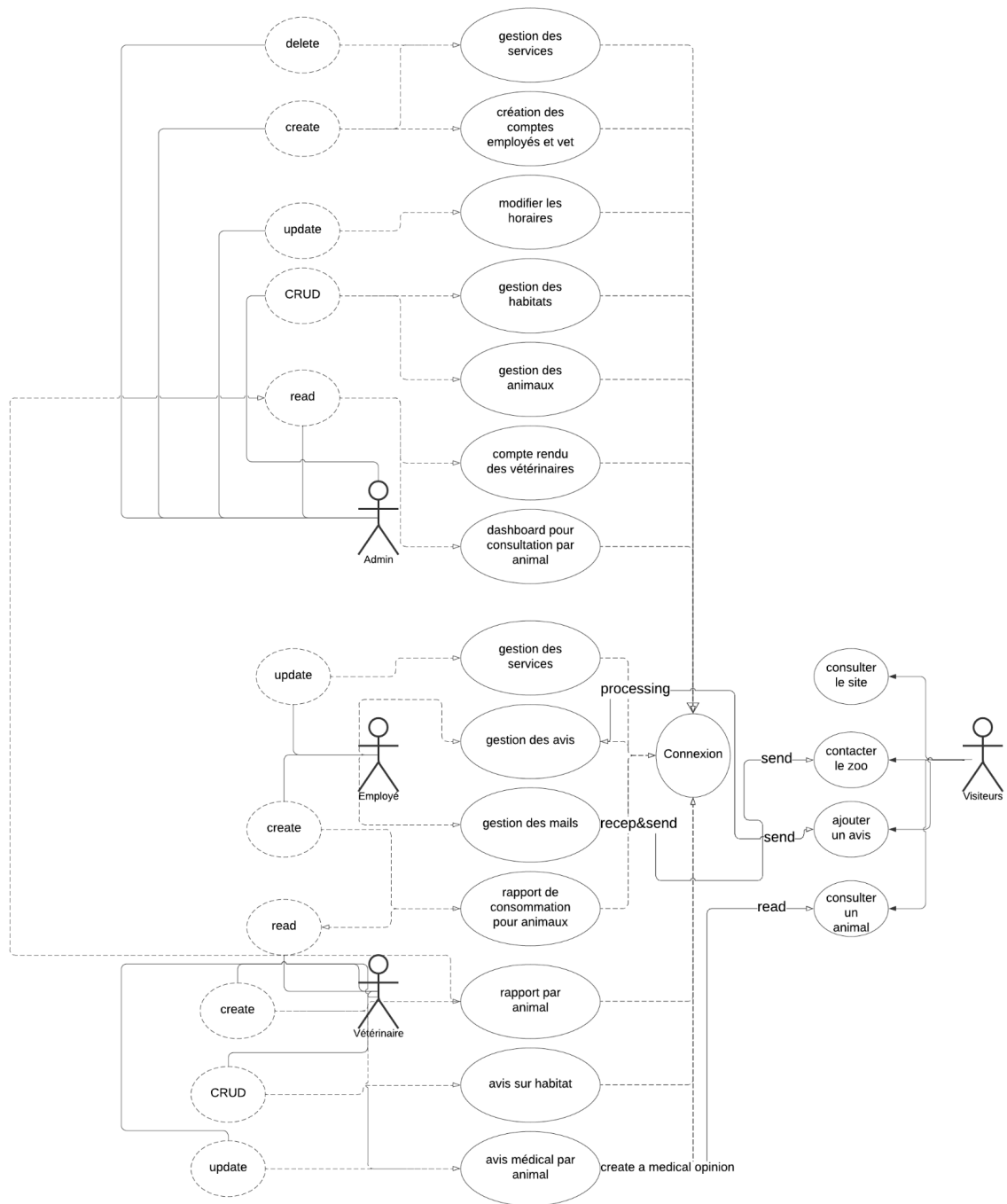
Administrateur : Ajouter/modifier/supprimer les services, habitats et animaux du zoo, gérer les horaires, recevoir les comptes rendus des vétérinaires et dashboard de consultation par animal.

Employé : Modifier les services, valider ou rejeter les avis, répondre aux mails des visiteurs, ajouter une consommation par animal.

Vétérinaire : Ajouter/modifier un avis médical par animal, ajouter/modifier/supprimer un commentaire par habitat, voir la saisie de consommation laissé par l'employé, créer un compte rendu par animal pour l'administrateur.

**Outil utilisé** : *Lucidchart*. permet de créer des diagrammes clairs pour visualiser ces interactions et scénarios.

Schéma de cas d'utilisation :

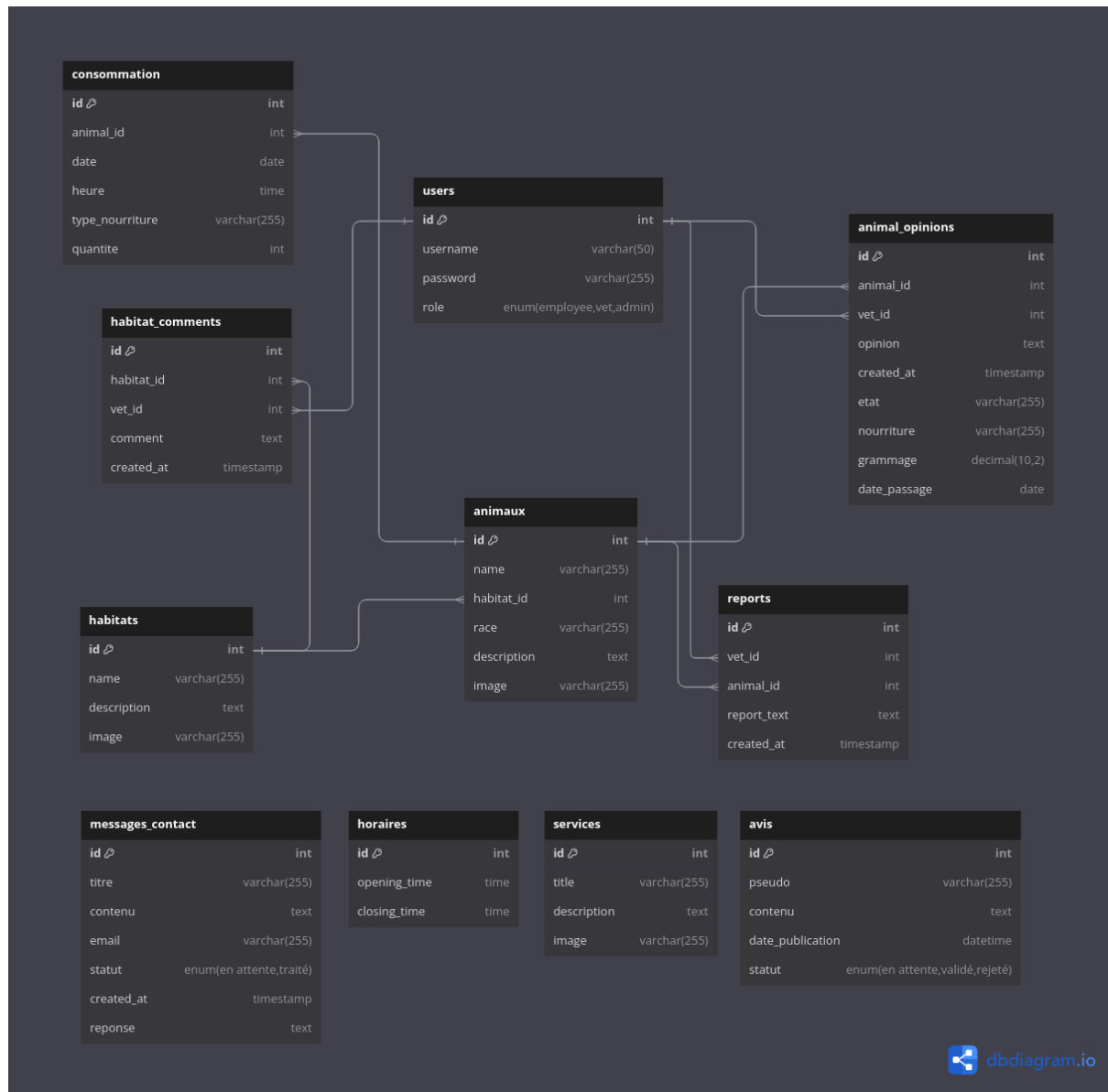


## 2. Structure de la base de données :

La base de données organise les informations du zoo de manière structurée. Elle contient plusieurs tables pour gérer les données des animaux, du personnel, etc.

**Outil utilisé :** *dbdiagram.io* pour concevoir la structure de la base de données. Cette structure permet de gérer efficacement toutes les informations nécessaires pour le bon fonctionnement du site.

### Schéma du diagramme ER :



### **3. Maquette du site :**

La maquette du site représente l'interface utilisateur (UI). Elle montre à quoi ressemblera le site avant le développement. Cela permet de visualiser l'agencement des pages et de tester l'expérience utilisateur.

**Outil utilisé :** *Figma* pour créer des maquettes interactives des pages principales du site, comme la page d'accueil, la page des animaux, la page des services, etc.

Les maquettes incluent des éléments de design comme les menus, les boutons, les couleurs et les typographies.

### **4. Mise en place de l'environnement de développement :**

L'environnement de développement est configuré pour travailler sur le projet. Cela inclut l'installation de logiciels et d'outils nécessaires.

IDE (Integrated Development Environment) : Nous utilisons *PhpStorm* comme éditeur de code principal pour écrire le code PHP et autres fichiers.

Composer : Outil de gestion des dépendances PHP. Il permet de gérer facilement les bibliothèques externes nécessaires au projet (mongodb, phpmailer, etc.).

Librairies externes : Nous utilisons des bibliothèques comme *Bootstrap* pour le design.

## **5. Développement du site :**

Le développement se fait en plusieurs étapes, avec l'utilisation de diverses technologies pour chaque aspect du site.

PHP & MySQL : Pour la logique serveur et la gestion de la base de données. Nous utilisons PHP pour interagir avec MySQL et afficher les données dynamiquement sur le site.

HTML & CSS : Pour la structure et le design de chaque page. HTML est utilisé pour la structure de base (titres, textes, images), tandis que CSS est utilisé pour la mise en forme et le style.

JavaScript : Pour l'interactivité. Par exemple, pour les messages pop-up (style events) lors d'un clic de suppression ou pour les images qui défilent.

Bootstrap : Pour un design responsive et professionnel. Il nous aide à rendre le site compatible avec tous les types d'appareils (ordinateurs, tablettes, mobiles).

Git : Pour le contrôle de version. Nous utilisons Git pour gérer l'historique des modifications et travailler en équipe sur le code sans risques de conflits.

## **6. Déploiement du site :**

Une fois le site développé, il faut le déployer sur un serveur pour qu'il soit accessible en ligne.

Hébergement (Hostinger) : Le site est déployé sur la plateforme d'hébergement *Hostinger*. C'est une solution d'hébergement fiable et facile à configurer.

Export de la base de données et des fichiers : Nous exportons la base de données MySQL et les fichiers du projet pour les transférer sur le serveur.

Service mail : Pour gérer l'envoi d'e-mails (envoyer des mails et répondre aux mails), nous utilisons un service de messagerie externe.

Base de donnée MongoDB : Pour certaines fonctionnalités, comme la gestion des avis des visiteurs ou des logs, nous utilisons *MongoDB Atlas*.

Une fois le déploiement terminé, le site est accessible à tous les utilisateurs.

Ce projet de site pour le zoo combine plusieurs technologies pour offrir une expérience fluide et agréable pour les visiteurs, tout en permettant une gestion facile pour les administrateurs du zoo. Grâce à des outils comme PhpStorm, Composer, Git, et des services externes, le projet peut évoluer et être maintenu facilement sur le long terme.