

Arcadia Zoo, Web Application - Technical documentation

by lorenzo-florenty.com - **Lorenzo FLORENTY**-Web Developer

Documentation Technique

Arcadia Zoo - 2024



INFORMATIONS

Project Name	Arcadia Zoo
Document Type	Technical Documentation
Date	May 22, 2024
Author	Lorenzo FLORENTY
GitLab Repository URL	https://gitlab.com/ecf3/arcadia-zoo
Website URL	https://arcadiazoo-972.fly.dev
Project Management link	https://gitlab.com/ecf3/arcadia-zoo/-/boards
FIGMA Graphic Chart	https://www.figma.com/design/3rlLaLNMiJ4ykZKmPJX2oV/Arcadia-Zoo?m=auto&t=AWhgEDzwJp0Bb4hA-6

TABLE DES MATIÈRES

Informations	2
Table des matières	2
1 - Résumé du document	4
2 - Rappel des principales informations sur le projet	5
2.1 - Description du projet	5
2.2 - Décomposition du projet	6
3 - Réflexions initiales sur les technologies utilisées	8
4 - Description de l'environnement de travail	9
5 - Description de l'interface de gestion de projet	10
6 - Manuel utilisateur	11
7- Documentation du déploiement de l'application	12

1 - RÉSUMÉ DU DOCUMENT

Ce document est la documentation technique officielle de l'application web de gestion des zoos d'Arcadia, Arcadia Zoo. Il est divisé en parties :

- Réflexions initiales sur les technologies utilisées
- Description de l'environnement de travail
- Description de l'interface de gestion de projet

2 - RAPPEL DES PRINCIPALES INFORMATIONS SUR LE PROJET

2.1 - Description Du Projet

Le projet se concentre sur la création d'une application web conviviale et écologique pour le Zoo Arcadia en France.

L'application permettra aux visiteurs d'explorer les différentes sections du zoo, de s'informer sur les services disponibles et d'accéder aux horaires. Les principales caractéristiques de l'application comprennent une page d'accueil qui présente le zoo, les habitats, les services et les commentaires. Le menu de l'application facilitera la navigation et permettra d'accéder aux services, aux habitats, à la connexion et à une option de contact.

La vue des services affichera divers services avec des options de personnalisation, tandis que la vue des habitats fournira des détails sur les habitats et les animaux associés. Les visiteurs pourront également laisser des commentaires, sous réserve de validation, dans la section des critiques.

L'espace administrateur permettra de gérer les comptes, les services, les horaires, les habitats, les animaux et les rapports vétérinaires. L'espace employé permettra la validation des avis, la modification des services et l'enregistrement des aliments pour animaux. L'espace vétérinaire permettra de saisir des rapports sur les animaux et des commentaires sur les habitats.

L'application disposera également d'une fonctionnalité de connexion réservée aux administrateurs, aux vétérinaires et aux employés.

Enfin, les visiteurs auront la possibilité de contacter le zoo au moyen d'un formulaire, ainsi que d'une fonction statistique permettant de suivre les consultations d'habitats et de fournir des données analytiques.

2.2 - Décomposition Du Projet

Le projet du zoo d'Arcadia se décompose comme suit :

FRONT-END :

I. User Interface :

Page d'accueil présentant le zoo, les habitats, les services et les commentaires.

Menu de navigation pour accéder aux différentes sections de l'application.

II. Affichage des données :

Affichage des services : Affiche les différents services du zoo avec les options de configuration.

Vue des habitats : Présentation des habitats et des animaux associés avec détails.

III. Interaction de l'utilisateur :

Laisser des commentaires : Les visiteurs peuvent soumettre des commentaires, sous réserve de validation.

IV. Contact :

Formulaire de contact permettant aux visiteurs de contacter le zoo.

BACK-END :

I. Gestion des utilisateurs :

Création de comptes d'administrateurs, de vétérinaires et d'employés.

Authentification et gestion des sessions utilisateurs.

II. Gestion du contenu :

Gestion des services, des habitats et des animaux.

Validation des commentaires soumis par les visiteurs.

III. Fonctionnalités vétérinaires :

Saisie de rapports sur les animaux et les habitats.

Consultation des données relatives à l'alimentation des animaux.

IV. Fonctionnalités des employés :

Valider les commentaires soumis par les visiteurs.

Modification des services du zoo.

Enregistrement de l'alimentation quotidienne des animaux.

V. Statistiques :

Suivi des consultations d'animaux pour fournir des données analytiques.

GESTION DES DONNÉES :

I. Stockage des données :

Utilisation d'une base de données pour stocker des informations sur les utilisateurs, les avis, les services, les habitats et les animaux.

II. Traitement des données :

Traitement des données pour les différentes fonctionnalités de l'application.

III. Sécurité :

Mise en œuvre de mesures de sécurité pour protéger les données sensibles et l'accès à l'application.

3 - RÉFLEXIONS INITIALES SUR LES TECHNOLOGIES UTILISÉES

Le projet consiste à créer une application Web pour le zoo d'Arcadia, avec une interface d'administration conviviale et un Front-End dynamique. La programmation orientée objet sera utilisée pour structurer efficacement les différents acteurs (administrateurs, employés, vétérinaires) et entités (animaux, habitats, avis des utilisateurs, etc.). L'utilisation d'un Back-End robuste est nécessaire pour alimenter le Front-End dynamique avec des données mises à jour en temps réel.

En termes de bases de données, l'idée est de combiner une base de données relationnelle pour les données sensibles et une base de données non relationnelle pour les données simples. Cette approche permet de minimiser les redondances et d'assurer une sécurité adéquate des données.

Le choix des frameworks s'est porté sur Symfony pour le Back-End, en raison de sa flexibilité, de son architecture robuste et de sa compatibilité avec EasyAdmin pour simplifier le développement de l'interface d'administration. Bien que React JS ait été envisagé pour le Front-End, une approche plus traditionnelle avec HTML, Twig et Tailwind CSS a été préférée pour une meilleure séparation des préoccupations et une gestion plus simple du Front-End.

En termes de bases de données, PostgreSQL a été choisi pour sa facilité d'implémentation et ses performances. Ces choix technologiques ont été faits en tenant compte des besoins spécifiques du projet, avec un souci constant de sécurité, de fiabilité et de performance.

4 - DESCRIPTION DE L'ENVIRONNEMENT DE TRAVAIL

Comme indiqué le projet est basé sur le framework PHP SYMFONY. L'ensemble du back-end est donc géré par du PHP. Le bundle Easy-Admin a été utilisé pour l'interface d'administration également.

En ce qui concerne la base de donnée, elle est hébergé par un conteneur docker, pareil pour l'interface de mailing (Webmail).

Le front end lui est géré principalement par TWIG. Il est épaulé par Tailwind CSS pour le style ainsi que par du JavaScript pour le dynamisme et la gestion des promesses.

5 - DESCRIPTION DE L'INTERFACE DE GESTION DE PROJET

Pour des raisons de facilités et de practicité, l'interface retenue est GitLab Issues Board (<https://gitlab.com/ecf3/arcadia-zoo/-/boards>).

Pour commencer, il est directement intégré au dépôt GitLab ce qui centralise les outils, de plus les issues peuvent être converties en Merge request facilitant le monitoring de l'avancée du projet.

Le mien est organisé comme suit.

En ce qui concerne mes panneaux il y a :

Open où l'ensemble des issues se trouvent à leur création,

Sprint pour les issues qui doivent être traitées prochainement

To be corrected pour les bugs à régler

In Progress pour les issues en cours de traitement

To be deployed in Pr-Production pour les issues terminées à mettre en Pré prod

Pre-Production pour les issues en Pre-Production (Version Bêta)

To be deployed in Production pour toutes les issues stables pas encore déployées

Closed pour toutes les issues déployées en production.

De plus les issues sont labelisées (Back, Front, Development, Animation, Conception, Features, INFRA, Documentation, Technical Debt, CI/CD, FIX, Web Design, Writing, ...) ce qui permet de comprendre rapidement sur quoi portera le travail à réaliser.

6 - MANUEL UTILISATEUR

Un compte administrateur Web MASTER (webmaster@arcadia.fr) existe déjà.

Il a pour mot de passe « webmaster ».

Il permet d'accéder à l'interface d'administration (/admin)

De là vous pourrez éditer, ajouter et supprimer vos différentes entités.

Vos Employés et Vétérinaires pourront accéder via le bouton « Employé.e.s » à une page de connexion les renvoyant vers leur espace personnel après connexion, où ils pourront accomplir les différentes tâches journalières.

7- DOCUMENTATION DU DÉPLOIEMENT DE L'APPLICATION

Le déploiement a été prévu su [fly.io](#) .

Donc il a fallu faire :

- fly apps create arcadia-zoo-972
- fly launch
- Gérer les informations de launch
- Rédiger le fly.toml

L'application est en ligne mais non fonctionnelle pour le moment.

Les bugs sont à régler.