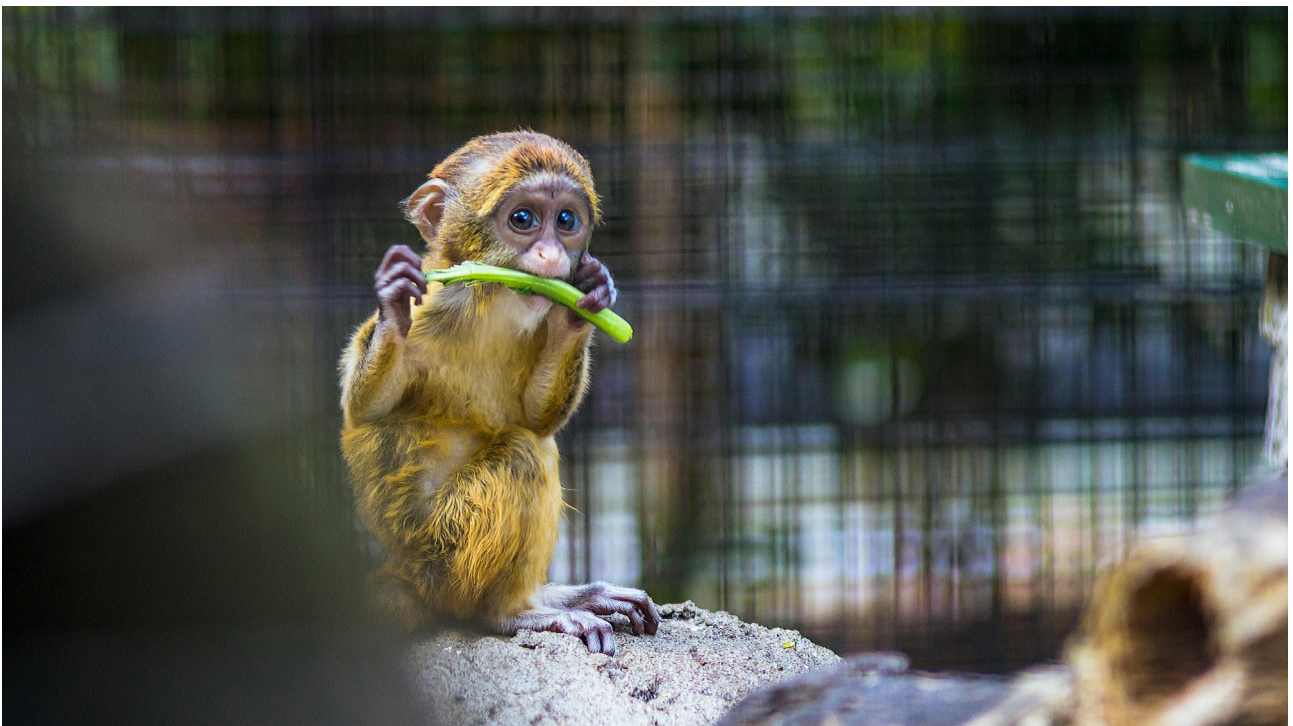


## Document technique



Rédigé par Lucy Rayé

Graduate développeur Angular 2023-2029  
Promotion janvier / février 2025

# Table des matières

<b>Thèmes</b>	<b>Pages</b>
Réflexion initiale technologique du projet	3-4
Configuration de l'environnement de travail	5-7
Diagramme de classe	8
Diagramme d'utilisation	9
Diagrammes de séquence	10-11
Démarche de déploiement de l'application	12-21
Annexe 1 : tests	22-fin

# Réflexion initiale technologique du projet

## 1/ le frontend, langage et sécurité

**HTML 5** : Le HTML est le langage de programmation le plus utilisé sur internet. Il est à la base de la structure de toute page web. Le HTML5 est sa dernière version. C'est d'ailleurs la plus sécurisée.

L'utilisation des attributs type et pattern dans une balise input permet notamment un premier niveau de sécurité afin de réduire les risques des attaques XSS. Tout d'abord une bonne définition du type d'un input va permettre de réduire les possibilités de saisies de l'utilisateur. De plus l'usage de pattern permet de réduire à nouveau les caractères autorisés lors d'une saisie par l'utilisateur.

**CSS 3** : Le CSS permet de dissocier la structure de l'esthétisme. Il rend le projet plus lisible et donc plus maintenable. Le CSS3 est sa dernière version. Elle est donc la plus adaptée aux nouveaux projets.

**Javascript** : JavaScript permet de créer des fonctionnalités interactives pour les utilisateurs, d'animer des éléments sur une page web et de manipuler le contenu HTML et CSS. La bibliothèque JQuery est également intéressante afin notamment de valider automatiquement un formulaire.

Il peut également être utilisé comme deuxième niveau de sécurité pour éviter les attaques XSS. Il peut également permettre de refuser la saisie de certains caractères par l'utilisateur.

## 2/ le backend, langage et sécurité

**MySQL** : Le projet nécessite l'utilisation d'un système de gestion de base de données (SGBD). L'un des plus populaires est MySQL (2<sup>e</sup> position selon <https://db-engines.com>). De plus il s'agit d'un SGBD open source avec une communauté active. Il est aussi compatible avec tous les systèmes d'exploitation. A cela s'ajoute qu'il est performant et dispose de fonctionnalités intégrées de sauvegarde des données, de réplication et de contrôle d'accès.

L'une des informations ciblées par les cybercriminels n'est autre que les accès aux données. Pour cela il est nécessaire de sécuriser les mots de passe par différentes manières. Tout d'abord ils ne doivent pas directement être accessibles via la base de données. Pour ce projet ils seront hashés avant d'être enregistré dans la base de données. De plus un mot de passe fort est moins facilement trouvable. La CNIL préconise au minimum 12 caractères comprenant des majuscules, des minuscules, des chiffres et des caractères spéciaux. L'utilisateur sera donc contraint de répondre à ces préconisations. Enfin il existe une attaque nommée par force brute qui consiste à essayer des mots de

passer jusqu'à obtenir le bon. Pour contrer cette attaque plusieurs choses seront mises en place. Tout d'abord il n'y aura pas d'indication sur l'élément faux entre l'identifiant et le mot de passe ; une phrase générique sera utilisée du type « votre identifiant et/ou mot de passe est erroné. » . De plus le compte sera bloqué à partir de trois tentatives infructueuses.

**MongoDB** : Le cahier des charges impose l'utilisation d'un système de base de données non relationnelles. Ce système a l'avantage d'un déploiement rapide et d'un bon support technique. C'est d'ailleurs le SGBD noSQL classé numéro 1 sur <https://www.ambient-it.net/top-meilleures-db-nosql/>.

**PHP 8.2.12** : PHP est l'un des langages côté serveur les plus populaires. Il a été conçu pour le web. De plus l'utilisation de PHP est open source avec une communauté très active. Il est aussi puissant et facile d'accès avec les bases de données et les images. La librairie PDO sera un outil incontournable pour faire la liaison avec la base de données relationnelle. La bibliothèque PHP MongoDB sera quant à elle utilisée pour la base de données NoSQL.

Le développement côté serveur doit porter attention à différents types d'attaques. PHP via des requêtes préparées aide à prévenir les attaques par injection de SQL en éliminant le besoin de protéger manuellement les paramètres.

L'utilisation d'une base de données nécessite une attention particulière aux droits d'utilisation. En effet il existe une attaque dénommée élévation des privilèges qui consiste à un utilisateur d'obtenir des privilèges auxquels il n'a normalement pas le droit. Pour éviter cela les URL du site n'auront aucun lien avec les privilèges. De plus, les éléments visibles seulement par certains utilisateurs ne seront pas disponibles dans le HTML/javascript. La sécurisation des mots de passe utile pour lutter contre la force brute sera également un bon usage pour lutter contre l'élévation des privilèges.

L'un des risques est aussi présent lors du chargement d'image par l'utilisateur. Il peut en effet charger un fichier malveillant. Pour ce projet le chargement sera limité à 5000ko et aux formats png et jpg.

### 3/ Absence de framework : pourquoi ?

Dans un projet professionnel l'utilisation de frameworks est indispensable. Des frameworks tel que Symfony et Bootstrap auraient certainement été utilisés.

Cependant l'objectif étant de montrer ma capacité à développer, aucun framework ne sera exploité.

# Configuration de l'environnement de travail

## Éditeur de code : VS code 1.90.1

Il est confortable d'utiliser un éditeur de code. En effet cela permet notamment la coloration syntaxique afin de voir plus aisément les boucles, les balises ainsi que l'utilisation des variables. Il permet également la complétion automatique ou encore le débogage.

VS Code est open source et léger ce qui en fait un bon outil.

Pour l'installer il suffit de le télécharger et suivre les instructions de l'assistant d'installation.

J'ai également installer les extensions suivantes :

- Composer : permet d'utiliser composer directement sur VSCode.
- Git History : permet d'utiliser GIT directement depuis VSCode notamment la création, le choix de branche, l'envoi de commit ainsi que la différence entre commit.
- PHP : prérequis pour la plupart des extensions installées
- PHP Debug : permet de déboguer du code php
- PHP Server : permet d'héberger localement un programme PHP
- PHP IntelliSense : permet la saisie semi-automatique du code, donne des informations sur les paramètres.
- IntelliPHP : fait de la prédiction de code et permet de présélectionner un élément dans une liste de suggestion (utile notamment pour l'utilisation des variables).

## Distribution de logiciels serveur : XAMPP v3.3.0

Utiliser cette distribution permet d'obtenir rapidement et facilement une site de logiciels indispensable pour le développement du projet :

**PHP** : qui permet de pouvoir coder en PHP 8.2.12

**Serveur local** : **Apache** permettant ainsi de visualiser le site en local

Pour l'installation il suffit de télécharger le logiciel et suivre l'assistant d'installation. J'ai également modifier le php.ini afin de pouvoir envoyer des mails :

```
SMTP=smtp.gmail.com
smtp_port=587
sendmail_from = monadresse@gmail.com
sendmail_path = [MON REPERTOIRE XAMPP]sendmailsendmail.exe -t
```

Ainsi que le sendmail.ini :

```
smtp_server=smtp.gmail.com
smtp_port=587
error_logfile=error.log
debug_logfile=debug.log
auth_username= monadresse@gmail.com
auth_password= monMotdePasse
force_sender= monadresse@gmail.com
```

### Interface graphique permettant l'administration d'un serveur

**MySql** : **phpMyAdmin**, permet de créer rapidement une base de données SQL et de vérifier simplement, sans terminal, le bon fonctionnement des requêtes réalisées depuis PHP.

**Interface graphique pour manipuler MongoDB** : **MongoDB Compass**, permet de pouvoir créer rapidement, sans terminal, une base de données noSQL et de vérifier simplement le bon fonctionnement des requêtes réalisées depuis PHP.

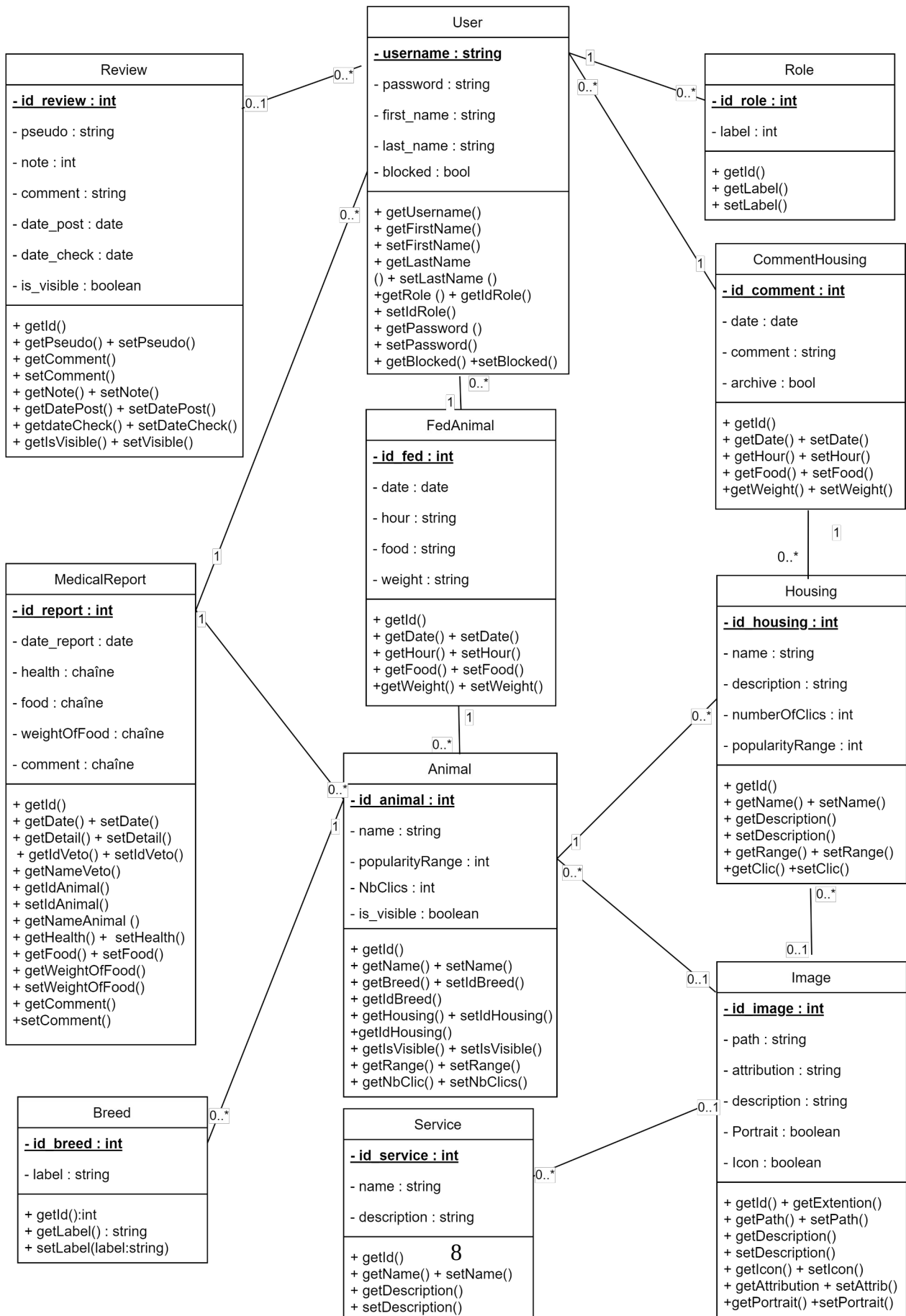
**Système de gestion de version** : **git 2.45.2.windows.1**, il s'agit d'un système distribué. Autrement dit, contrairement à un système centralisé, le développeur dispose d'une copie complète du dépôt sur sa machine locale et peut travailler hors ligne et indépendamment. De plus c'est le plus utilisé sur le marché.

**Plateforme de gestion de version** : **github**, permet l'hébergement de projet et de ses versions en ligne. Il offre d'autres fonctionnalités non utilisé pour ce projet tel que le partage de codes open sources, la gestion de projet et facilite le travail en groupe. Sauf si le projet nécessite un outil particulier il est préférable d'utiliser ceux qui sont les plus maîtrisés. C'est pour cela que cette plateforme a été validée pour ce site.

**Hébergement :** **Virtual Private Server (VPS) d'OVH**, OVH est un acteur populaire de l'hébergement français. De plus un VPS est un investissement pour l'avenir. Il permet de pouvoir héberger plusieurs sites et surtout permet une maîtrise totale de la configuration de l'environnement. Les avantages comparés à un serveur physique sont la flexibilité et le coût. Il est donc idéal pour des projets ne nécessitant pas de ressources importantes.

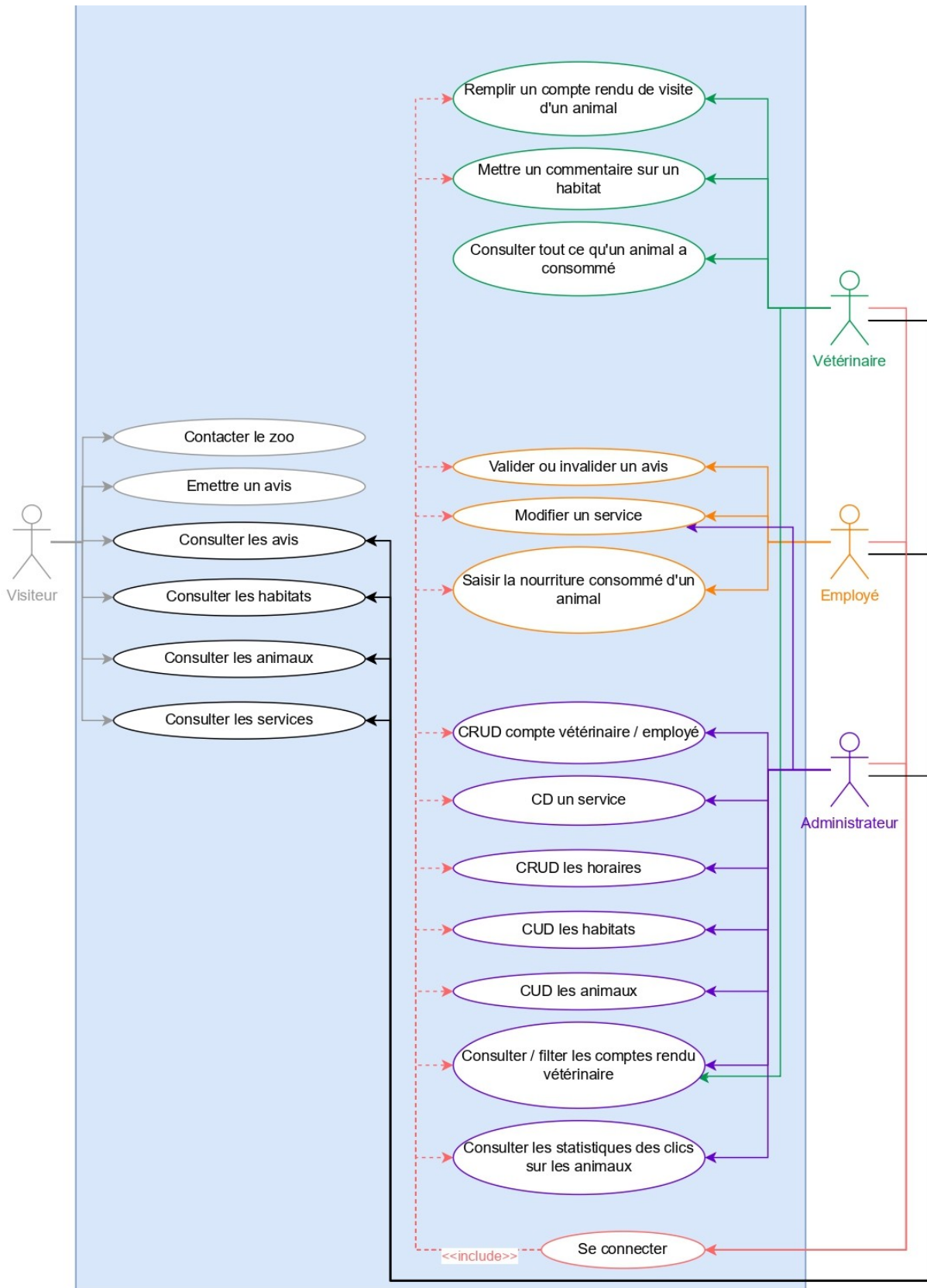


# Diagramme de classe





# Diagramme d'utilisation



# Diagrammes de séquence

Diagramme de séquence : connexion

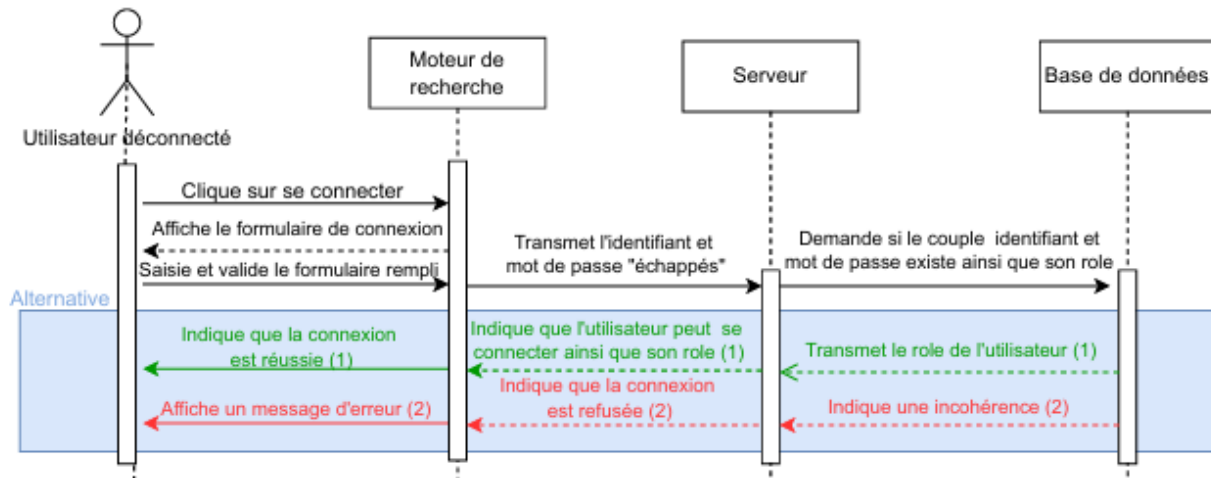
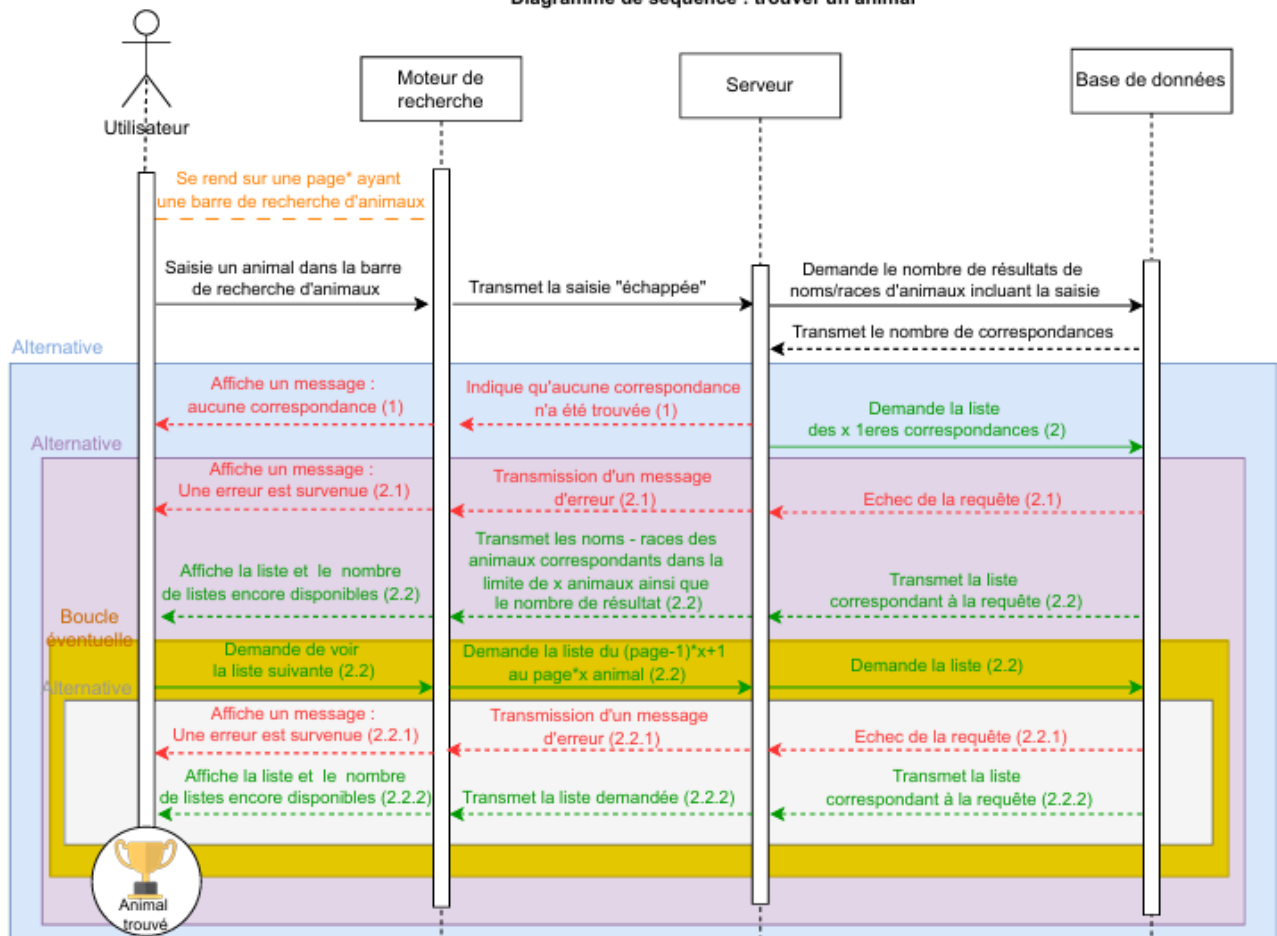
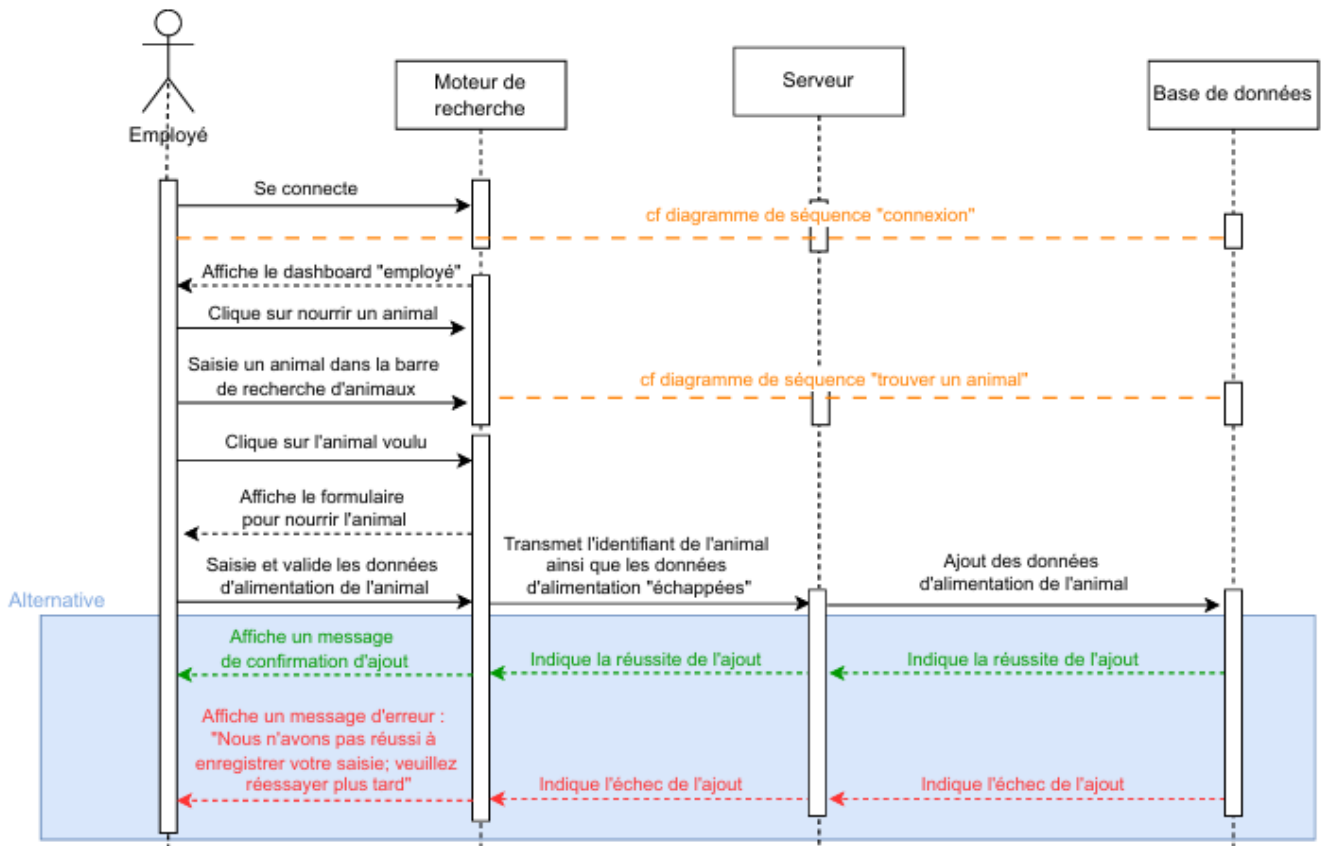


Diagramme de séquence : trouver un animal



\*Exemples de page : habitats, animaux, nourrir un animal, comptes-rendus d'animaux...

Diagramme de séquence : nourrir un animal



# Démarche pour le déploiement de l'application

Avant de déployer le site j'ai réalisé **une série de tests** que vous trouverez en annexe.

## Le choix du dispositif :

J'ai tout d'abord commencé par regarder les différentes possibilités d'hébergement que proposaient OVH. J'avais donc le choix entre l'hébergement web, un serveur et un serveur privée virtuel.

L'hébergement web aurait été la solution de facilité. OVH gère l'ensemble des technologies (SGBD, serveur web, langage...) ainsi que la sécurité, l'utilisateur dépose et gère son ses fichiers / données. Seulement l'utilisateur n'a pas accès aux versions utilisées ni aux éléments de sécurité. Il ne maîtrise pas l'environnement.

Le serveur physique permet qu'en à lui de gérer entièrement l'environnement mais est onéreux et nécessite une maintenance importante. Il est idéal pour de gros projets.

Le serveur privé virtuel (VPS) est un entre deux idéal. Il permet de gérer complètement son environnement sans le coût imposant d'un serveur physique.

## Le nom de domaine :

Il permet de rendre plus facilement accessible le site en évitant d'imposer à l'utilisateur une suite de chiffres plus difficilement retenable.

Beaucoup étaient déjà pris, j'ai tout de même fait ce choix : **arcadiaecf.fr**

Pour le lier à mon VPS j'ai suivi l'aide en ligne d'OVH :

[https://help.ovhcloud.com/csm/fr-vps-secondary-dns-vps?](https://help.ovhcloud.com/csm/fr-vps-secondary-dns-vps?id=kb_article_view&sysparm_article=KB0047511)

[id=kb\\_article\\_view&sysparm\\_article=KB0047511](https://help.ovhcloud.com/csm/fr-vps-secondary-dns-vps?id=kb_article_view&sysparm_article=KB0047511)

[https://help.ovhcloud.com/csm/fr-dns-edit-dns-zone?](https://help.ovhcloud.com/csm/fr-dns-edit-dns-zone?id=kb_article_view&sysparm_article=KB0051684)

[id=kb\\_article\\_view&sysparm\\_article=KB0051684](https://help.ovhcloud.com/csm/fr-dns-edit-dns-zone?id=kb_article_view&sysparm_article=KB0051684)

## La configuration du VPS :

J'ai choisi le système d'exploitation Debian 12 (la plus récente proposée) qui était livré avec Docker.

### 1 étape : la sécurité

## Mise en place du protocole HTTPS :

S'agissant d'un site « pédagogique » qui n'aura pas réellement de trafic j'ai choisi de passer par le SSL Gateway d'OVH qui prend en charge toutes les démarches et configurations : <https://www.ovh.com/fr/ssl-gateway/>

### Mise à jour des paquets :

Après m'être connecté avec le protocole SSH, j'ai commencé par faire une mise à jour des paquets :

```
sudo apt update  
sudo apt upgrade
```

### Modification du port d'écoute SSH par défaut :

Par défaut le port d'écoute SSH est le **22**, les tentatives de hack du serveur par des robots vont ainsi cibler ce dernier en priorité. Sa modification est donc une mesure simple pour renforcer la protection du VPS contre les attaques automatisées.

Pour ce faire il faut modifier le fichier de configuration du service :

```
sudo nano /etc/ssh/sshd_config
```

et modifier :

```
#Port 22
```

Par un nombre **compris entre 49152 et 65335** non utilisé (en enlevant le #).  
Il faut ensuite faire `crtl X` et entrer.

Il est alors nécessaire de redémarrer le service :

```
sudo systemctl restart sshd
```

Par la suite la connexion via ssh impliquera le paramètre port :

```
ssh -p XXXXX user@IP
```

### Paramétrer le pare-feu :

Les distributions GNU/Linux sont fournies avec le service de pare-feu « **iptables** » que j'ai paramétré ainsi :

Vérification de sa bonne installation (déjà réalisé par ovh) :

```
sudo iptables -L
```

Autoriser le trafic de le système interne (le localhost) :

```
sudo iptables -A INPUT -i lo -j ACCEPT
```

Autoriser le trafic Internet HTTPS :

```
sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

Autoriser le trafic Internet HTTP :

```
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

Autoriser uniquement le trafic SSH (Secure Shell) entrant :

```
sudo iptables -A INPUT -p tcp --dport n°port -j ACCEPT
```

Sauvegarder les modifications :

```
sudo -s iptables-save -c
```

### Installation de Fail2ban :

Fail2ban est un **framework de prévention contre les intrusions** dont le but est de bloquer les adresses IP depuis lesquelles des bots ou des attaquants tentent de pénétrer dans le système.

Ce paquet est recommandé pour protéger le VPS des **attaques de types *Brute Force* ou *Denial of Service***.

Pour l'installer il faut utiliser la commande suivante :

```
sudo apt install fail2ban
```

Comme le recommande Fail2ban, j'ai créé un fichier de configuration local de vos services en copiant le fichier "jail" :

```
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
```

### Activation et configuration de Edge Network Firewall :

Pour protéger les services des clients exposés sur les adresses IP publiques, OVHcloud propose un pare-feu sans état. Il permet de limiter l'exposition des services aux attaques DDoS, en supprimant des flux réseau spécifiques qui peuvent provenir de l'extérieur du réseau OVHcloud.

Pour le paramétrer j'ai suivi l'aide en ligne : [https://help.ovhcloud.com/csm/fr-dedicated-servers-firewall-network?id=kb\\_article\\_view&sysparm\\_article=KB0043455](https://help.ovhcloud.com/csm/fr-dedicated-servers-firewall-network?id=kb_article_view&sysparm_article=KB0043455)

### Etape 2 : l'installation des technologies et du routage

Pour que mon site fonctionne correctement j'ai besoin :

- Un serveur WEB (choix de Nginx)
- PHP (avec pdo et librairie mongodb)
- MySQL
- mongoDB

- composer

J'ai trouvé ce tutoriel qui correspondait grandement à mon besoin :  
<https://www.tutorials24x7.com/devops/containerize-php-with-nginx-mysql-and-mongodb-using-docker-containers> (attention à la petite coquille dans le dockerfile de php il est nécessaire d'ajouter -fpm)

Au préalable j'ai dû finaliser l'installation de Docker :  
J'ai d'abord vérifié l'installation réalisé par OVH :

```
docker run hello-world
```

Puis j'ai installé docker-compose :

```
sudo apt install docker-compose
```

J'ai remis au goût du jour le docker-compose proposé par ce tutoriel et j'ai enlevé les conteneurs des interfaces graphiques des bases de données. J'ai également conçu une fichier .env afin d'extraire les données sensibles :

*version: "3.8"*

*services:*

*nginx:*

*container\_name: nginx*

*build: ./docker/nginx*

*command: nginx -g "daemon off;"*

*ports:*

*- "80:80"*

*volumes:*

*- ./src:/var/www/html*

*depends\_on:*

*- php*

*php:*

*container\_name: php*

*build: ./docker/php*

*ports:*

*- "9000:9000"*

*volumes:*

*- ./src:/var/www/html*

*- ./php/php.ini:/usr/local/etc/php/php.ini*

*working\_dir: /var/www/html*



*depends\_on:*

- *mysql*
- *mongo*

*mysql:*

*image: mysql:8.0.27*

*container\_name: mysql*

*environment:*

*MYSQL\_ROOT\_PASSWORD: \${MYSQL\_ROOT\_PASSWORD}*

*MYSQL\_DATABASE: \${MYSQL\_DATABASE}*

*MYSQL\_USER: \${MYSQL\_USER}*

*MYSQL\_PASSWORD: \${MYSQL\_PASSWORD}*

*ports:*

- *"3306:3306"*

*volumes:*

- *./database/mysql:/var/lib/mysql*

*mongo:*

*image: mongo:5.0*

*container\_name: mongo*

*environment:*

*MONGO\_INITDB\_ROOT\_USERNAME: \${MONGO\_INITDB\_ROOT\_USERNAME}*

*MONGO\_INITDB\_ROOT\_PASSWORD: \${MONGO\_INITDB\_ROOT\_PASSWORD}*

*restart: unless-stopped*

*ports:*

- *"27017:27017"*

*volumes:*

- *./database/mongodb/db:/data/db*
- *./database/mongodb/dev.archive:/Databases/dev.archive*
- *./database/mongodb/production:/Databases/production*

*networks:*

*default:*

*driver: bridge*

J'ai modifié également de dockerfile du conteneur PHP afin d'intégrer mes librairies et particularités du php.ini :

*FROM php:8.1-fpm*

*# Installation des dépendances nécessaires (zip, unzip, git, et d'autres)*

*RUN apt-get update && apt-get install -y*

*RUN docker-php-ext-install pdo pdo\_mysql mysqli*

*RUN apt-get install -y autoconf pkg-config libssl-dev*

*# Installation de l'extension MongoDB pour PHP via PECL*

*RUN pecl install mongodb |*

*&& docker-php-ext-enable mongodb*

*#RUN echo "extension=mongodb.so" >> /usr/local/etc/php/conf.d/mongodb.ini*

*# Installation de Composer (gestionnaire de dépendances PHP)*

*COPY --from=composer:latest /usr/bin/composer /usr/bin/composer*

*#personnalisation du php.ini*

*COPY config/custom.ini /usr/local/etc/php/conf.d/*

*# Configuration du répertoire de travail*

*WORKDIR /var/www/html*

*# Donne les bons droits d'exécution*

*RUN chown -R www-data:www-data /var/www/html*

*# Expose le port 80 pour Nginx*

*EXPOSE 80*

*# Lancement à la fois de PHP-FPM et Nginx*

*CMD php-fpm*

J'ai du personnaliser le php.ini pour que PHP accepte de télécharger des fichiers dépassant 2Mo via ce contenu de fichier (custom.ini) :

*upload\_max\_filesize = 6M*

*post\_max\_size = 6M*

J'ai également personnalisé le arcadiaecf.fr.nginx.conf afin de pouvoir rediriger toutes les urls sur index.php et permettre le téléchargement d'une image de 5M (éviter l'erreur 502 bad gateway) :

*# docker/nginx/arcadiaecf.fr.nginx.conf*

*server {*

*listen 80;*

*index index.php;*

*root /var/www/html;*

```

error_log /var/log/nginx/error.log;
access_log /var/log/nginx/access.log;
client_max_body_size 10M; # taille des fichiers téléchargés max

location / {
    try_files $uri $uri/ /index.php?$query_string;
}
location ~ \.php$ {
    include fastcgi_params;
    fastcgi_pass php:9000;
    fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
}
location ~ /\.ht {
    deny all;
}
}

```

J'ai ensuite redémarrer nginx pour la prise en compte de ces modifications :

```
docker exec -it nginx nginx -s reload
```

### **Etapas 3 : les imports des fichiers et bases de données**

J'ai **déposé mon projet dans le dossier src** (ainsi que le dossier vendor et le composer.json pour que mongodb puisse fonctionner). Pour ce faire j'ai utilisé fileZilla avec le protocole sftp.

J'ai ensuite **importé la base de données SQL**.

Pour ce faire j'ai d'abord exporté ma base locale depuis mon phpMyAdmin afin d'obtenir un fichier .sql .

J'ai ensuite mis ce fichier dans mon vps à l'adresse : mon\_projet/database/mysql

Une fois le conteneur en route via la commande :

```
docker-compose up -d
```

J'ai établi une session de terminal à l'intérieur de mon conteneur mysql :

```
docker exec -it mysql /bin/bash
```

J'ai ensuite pu faire mon import :

```
mysql --user=root --password='pwd' arcadia_zoo < /var/lib/mysql/arcadia_zoo.sql
```

Pour finir, j'ai vérifié la réussite de l'import :

```
Mysql -u root -p
Enter password : ****
Use arcadia_zoo ;
Show tables ;
```

J'ai ainsi pu voir les tables de ma base de données.

Enfin j'ai **créé la base de données MongoDB**.

Une fois le conteneur en route via la commande :

```
docker-compose up -d
```

J'ai établi une session de terminal à l'intérieur de mon conteneur mongo :

```
docker exec -it mongo /bin/bash
```

Je me suis ensuite connectée :

```
mongosh "mongodb://mongo:27017" --username root --authenticationDatabase admin
Enter password : *****
```

J'ai ensuite créé ma base de données :

```
use Arcadia ;
db.createCollection('schedules') ;
db.createCollection('popularity') ;
db.schedules.insertOne({Text : « horaire a définir »}) ;
```

Pour vérifier la réussite de cette création :

```
Show collections ;
```

J'ai ainsi vu les collections.

```
Db.schedules.find()
```

J'ai ainsi vu le document des horaires.

#### Étape 4 : adaptation du projet

J'ai commencé par **modifier le fichier config.php** du projet avec les nouvelles informations des bases données et url.

J'ai continué avec la modification de la variable url dans le fichier CONSTANTES\_inputs.js

J'ai également adapté le fichier router.php notamment les fonctions optionPage(\$request) et loadContentPage(\$request).

```

function optionPage($request){
    $requestExplode = explode('/', $request, 2);
    if(count($requestExplode)>1 && !str_contains($request, SITE_URL)){
        return true;
    }
    else if(count($requestExplode)>2 && str_contains($request, SITE_URL)){
        return true;
    }
    else{
        return false;
    }
}

function loadContentPage($request){
    $option = optionPage($request);
    $request = explode('/', $request, 2);
    $actualRoute = getRouteByUrl($request[1], $option);
    if($actualRoute->getPathController()!=null)
    {
        include $actualRoute->getPathController();
    }
    if($actualRoute->getPathHtml()!=null){
        include $actualRoute->getPathHtml();
    }
}

```

j'ai du adapter le code du fait de passer par PHPMailer pour le formulaire de contact.  
J'ai ajouté des constantes incluant mes identifiants / mot de passe ovh.

Voici le nouveau ManageContact :

```

use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

```

[... gestion des \_POST inchangé]

```

//si on a pu tout récupéré on envoi le mail à Arcadia
if($goodForSend){

    // code utilisé pour le déploiement local (SMTP via XAMPP)
    //destinataire présent dans les constantes
    /* $recipient = MAIL_CONTACT;

    $subject = "Nouveau message : ".$obj;
    $message = "Vous avez un nouveau message :". "<br>";
    $message .= "Emetteur : ".$firstName.' '.$lastName."<br>";
    if($company != "") $message.= "Entreprise : ".$company."<br>";
    $message .= "Mail : ".$mail."<br>";
    $message .= "Téléphone : ".$tel."<br>";

```

```

$message .= "Objet : ".$obj."<br>";
$message .= "Message : ".$msg."<br>";
$headers = "MIME-version: 1.0\r\n"."Date: '.date("r").'\r\n";

//émetteur du mail le visiteur permettant ainsi à Arcadia de faire directement répondre
$headers .= "From: ".$firstName." ".$lastName." <".$mail.">". "\r\n"."Reply-To: ".$firstName." ".
$lastName." <".$mail."> \r\n";
$headers .= "Content-type: text/html; charset=utf-8" . "\r\n";
if(mail($recipient,$object,$message,$headers)) $msg=true;
else $msg=false;*/

require 'vendor/autoload.php';

$mail = new PHPMailer(true);
try {
    // Configuration du serveur
    $mail->isSMTP();
    $mail->Host      = HOST_SMTP; // Serveur SMTP
    $mail->SMTPAuth  = true;
    $mail->Username  = MAIL_CONTACT; // Nom d'utilisateur
    $mail->Password  = SMTP_PWD; // Mot de passe
    $mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS; // Activer le TLS
    $mail->Port      = SMTP_PORT; // Port

    // Destinataires
    $mail->setFrom(MAIL_CONTACT, 'Zoo');
    $mail->addAddress( MAIL_CONTACT, 'contact ');

    // Contenu de l'email
    $mail->isHTML(true); // Envoyer au format HTML

    $mail->Subject = "Nouveau message : ".$obj;
    $mail->Body = "<b>Vous avez un nouveau message :". "<br>";
    $mail->Body .= "Emetteur : ".$firstName.' '.$lastName."<br>";
    if($company != "") $message.= "Entreprise : ".$company."<br>";
    $mail->Body .= "Mail : ".$adrMail."<br>"; //modif du nom de variable mail en adrMail
    $mail->Body .= "Téléphone : ".$tel."<br>";
    $mail->Body .= "Objet : ".$obj."<br>";
    $mail->Body .= "Message : ".$msg."<b>";
    $mail->AltBody = 'This is the body in plain text for non-HTML mail clients';

    $mail->send();
    $msg = true ;
} catch (Exception $e) {
    echo "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";
}

```

# ANNEXE 1

## Tests :

Action	Attendu	Résultat
<p><i>Menu : cahier des charges :</i>  <i>Le menu est essentiel à l'application, il permet de faciliter la navigation et de fluidifier le trafic, ainsi, il doit composer au minimum :</i>  <i>Retour vers la page d'accueil</i>  <i>Accès à tous les services</i>  <i>Accès à tous les habitats</i>  <i>Connexion (la connexion sera possible uniquement pour les vétérinaires, employé et administrateur)</i>  <i>Contact</i></p>		
Menu : clic sur tous les liens	Envoi vers la page indiquée	Idem attendu
Menu : voir l'intitulé de la page actuelle	page actuelle en bleu	Idem attendu
Menu : clic sur me connecter	Ouverture popup de connexion	Idem attendu
Menu : clic sur le nom de l'utilisateur	aucun	Idem attendu
Menu : vérification des possibilités après connexion	Espaces différents suivant utilisateur	Idem attendu
Menu : clic sur déconnexion	Déconnexion	Idem attendu
Menu : Voir le menu sur un écran de téléphone	Affichage d'un menu burger qui se déplie au clic	Idem attendu
Footer : clic sur les liens	Envoi vers la page indiquée	Idem attendu
<p><i>Accueil : cahier des charges :</i>  <i>La page d'accueil doit comporter :</i>  <i>Présentation du zoo en y incorporant quelques images</i>  <i>Mentionnez les différents habitats, services ainsi que les animaux que possède le zoo</i>  <i>Les avis du Zoo</i></p>		
Accueil : clics sur tous les boutons	Envoi vers la page indiquée	Idem attendu
Accueil (section habitats - desktop) : Attendre sur la section	Les habitats doivent défiler	Idem attendu



Accueil (section habitats - mobile) : voir tous les habitats et animaux	Les habitats ne doivent pas défiler et une nouvelle section animaux avec un carroussel doit-être présente	30/08 : le changement de taille engendre la perte des img 01/09 : Idem attendu
Accueil (section habitats) : voir un habitat / animal sans photo	Une image par défaut doit s'afficher	Idem attendu
Accueil (section habitats – desktop) : voir un habitat avec moins de 3 animaux	Les animaux disponibles doivent s'afficher	Idem attendu
Accueil (section avis) : voir les avis	Seul les avis validé doivent apparaître	Idem attendu
Accueil (section avis) : voir la section après connexion	Le bouton donner mon avis ne doit plus être accessible	Idem attendu
<p><i>Page services : cahier des charges :</i></p> <p><i>Utilisateur concerné : Visiteur</i></p> <p><i>Une vue globale est nécessaire afin de proposer une interface simple et récapitulative de tous les services que propose le parc. Les services doivent être configurable depuis l'espace "Administrateur", un service dispose des caractéristiques suivantes :</i></p> <p><i>Un nom</i></p> <p><i>Une description</i></p> <p><i>Pour le moment, le parc dispose de plusieurs services : restauration, visite des habitats avec un guide (gratuit), visite du zoo en petit train</i></p>		
Services : accès à la page sans connexion	Voir tous les services sans icone ni bouton ajout	Idem attendu
Services : modifier la taille de la fenêtre	La mise en page doit s'adapter	Idem attendu
Services : accès à la page avec la connexion admin	Icônes de modification et suppression s'affichent ainsi que l'ajout d'un service	Idem attendu
Services : accès à la page avec la connexion vétérinaire	La page est la même que sans connexion	Idem attendu
Services : clic sur l'icône de suppression d'un service (fait sur au moins 2 services)	Une popup de confirmation doit s'afficher avec le nom du service	Idem attendu

Services : clic sur non sur la popup de confirmation de suppression	La popup se ferme, la page est identique à avant le clic sur l'icône	Idem attendu
Services : clic sur oui sur la popup de suppression	Le service ne doit plus être visible et doit aussi être supprimé de la BD ainsi que ses images	Idem attendu
Services : rechargement de la page après une suppression	Rien	Idem attendu
Services : clic sur ajouter un service	Envoi vers une nouvelle page pour l'ajout d'un service	Idem attendu
Ajouter : ne rien renseigner	Message d'erreur sans ajout dans bd ni dossier	Idem attendu
Ajouter : ne renseigner qu'un élément (hors attributions, descriptions d'images)	Message d'erreur sans ajout dans bd ni dossier	Idem attendu
Ajouter : mettre un fichier autre que png/jpg	Message d'erreur sans ajout dans bd ni dossier	Idem attendu
Ajouter : mettre une image trop volumineuse	Message d'erreur sans ajout dans bd ni dossier	Idem attendu
Ajouter : renseigner tous les champs	Ajout dans la base de données, création d'un dossier avec les images et visible depuis services	Idem attendu
Services / mise à jour : clic sur l'icône mise à jour	Le formulaire de MAJ doit apparaître dans le service concerné	Idem attendu
Services / mise à jour : clic sur Annuler	Le formulaire disparaît	Idem attendu
Services / mise à jour : suppression des textes dans les champs obligatoires	Message d'erreur	03/09 (1) : le formulaire se valide avec une erreur 03/09 (2) : Idem attendu
Services / mise à jour : changement des textes	Le service doit s'afficher avec les nouveaux textes	Idem attendu
Services / mise à jour :	L'ancienne photo doit être	03/09 : rien ne se passe...

modification de la photo	supprimée du fichier, la nouvelle doit être incluse et le service doit s'afficher avec la nouvelle photo	03/09 (2) : idem attendu
Services / mise à jour : modification de l'icône	L'ancienne icône doit être supprimée du fichier, la nouvelle doit être incluse et le service doit s'afficher avec la nouvelle icône (dans accueil)	03/09 : img du dossier ok, path de la photo dans la bd modifiée... 03/09 (2) : Idem attendu

*Habitats : cahier des charges*

*Utilisateur concerné : Visiteur Cette page, doit mentionner tous les habitats que propose le zoo et les animaux associés. José vous propose une vue tout d'abord de tous les habitats (en affichant que l'image et le nom), puis, au clic sur celui-ci, on affiche le détail avec les animaux ainsi que la description. Un habitat est caractérisé par :*

*Un nom*

*Une ou des images*

*Une description de l'habitat*

*Une liste d'animaux*

*Un animal est caractérisé par :*

*Un Prénom*

*Une race*

*Une ou des images*

*Un habitat où il est affecté Le vétérinaire, passe régulièrement et saisie des informations depuis son espace sur un animal donné en mentionnant :*

*L'état de l'animal*

*La nourriture proposée*

*Le grammage de la nourriture*

*Date de passage*

*Détail de l'état de l'animal (information facultative)*

*Il est très important pour José de mentionner l'état de l'animal sur cette page. Enfin, au clic sur un animal de l'habitat, le visiteur doit pouvoir visualiser le détail de l'animal, ce qui inclue ses propriétés mais également l'avis du vétérinaire.*

Habitats : changement de taille d'écran	La mise en page doit s'adapter	Idem attendu
Habitats (hero) : clic sur un habitat	Le détail de l'habitat apparaît	Idem attendu
Habitat (détail) : clic sur un animal	Le détail de l'animal apparaît	Idem attendu
Habitat : dans savane saisi	Seul les lions doivent	Idem attendu

de lion dans recherche animal	apparaître dans la liste	
Habitat : dans savane saisi de chien dans recherche animal	Un message indiquant qu'il n'y a pas cet animal apparaît	Idem attendu
<p>Avis : cahier des charges :</p> <p>Utilisateur concerné : Visiteur</p> <p>Un visiteur, peut laisser un commentaire s'il le désire. Ce commentaire, contiendra juste un pseudo ainsi qu'un champ "avis" texte. Cet avis sera ensuite soumis à validation par l'employé. Depuis l'espace employé, il pourra ensuite autoriser ou non l'avis à apparaître sur la page d'accueil. La soumission d'un avis se fait par cette même page.</p>		
Accueil et Avis : clic sur donner mon avis	Une popup apparaît	Idem attendu
Accueil et Avis : aucune saisie et validation	Un message d'erreur s'affiche	Idem attendu
Accueil et Avis : saisi seulement du pseudo ou du commentaire	Un message d'erreur s'affiche	Idem attendu
Accueil et Avis : validation sans clic sur une étoile	Rien de doit se passer	Idem attendu (amélioration possible : message s'affiche)
Accueil et Avis : clic sur chacune des étoiles	L'étoile cliqué et les précédentes changent de couleur	Idem attendu
Accueil et Avis : après avoir tout renseigné clic sur soumettre	L'avis doit apparaître dans la BD	Idem attendu
Avis : voir les avis (sans connexion)	l'avis crée précédemment ne doit pas apparaître	Idem attendu
Avis : clic sur le pouce levé d'un avis ainsi que sur valider vérifications	L'avis apparaît désormais sans être connecté et le nombre d'avis a augmenté de 1	03/09 : un echo a été oublié... sinon idem attendu
Avis : clic sur le pouce baissé d'un avis ainsi que sur valider vérifications	L'avis n'apparaît pas sans être connecté et le nombre d'avis modéré a augmenté de 1	Idem attendu
Avis : valider un avis en	La moyenne doit changer	Idem attendu

dessous de 5 étoiles	ainsi que « les barres »	
Avis : décocher des filtres	Seul les avis au statut coché apparaissent	Idem attendu
Avis : changer le tri	Les avis s'affichent en fonction du tri	Idem attendu
<p><i>Espace Administrateur : cahier des charges :</i></p> <p><i>Utilisateur concerné : Administrateur</i></p> <p><i>L'admin, peut créer un compte de type "employé" et "vétérinaire", il doit pour cela, fournir un courriel (qui sera l'username) ainsi qu'un mot de passe. L'utilisateur en question, va recevoir un mail avec son username, cependant, le mot de passe n'est pas communiqué dans le mail. Il devra se rapprocher de l'administrateur afin de l'obtenir. Il peut également modifier les services, horaire, habitat et animaux du zoo (création, mise à jour et suppression). Ensuite, l'espace administrateur doit avoir un emplacement avec tous les comptes rendus du vétérinaire, des filtres sont à positionner afin de pouvoir trier et filtrer les comptes rendus sur un animal ou une date. Enfin, il doit pouvoir avoir un Dashboard lui montrant le nombre de consultation par animal (ce besoin est décrit en US 11) José précise que vous devez lui créer ce compte et qu'il ne doit pas être possible de créer un compte Administrateur depuis l'application.</i></p>		
Comptes utilisateurs : clic sur « comptes utilisateurs » dans menu	Affichage de la page comptes utilisateurs	Idem attendu
Comptes utilisateurs : modification de la taille de l'écran	Mise en page s'adapte	06/09 : pas top pour mobile 07/09 : idem attendu
Comptes utilisateurs (ajout) : mauvaise saisie mail et mot de passe	Input en rouge / pas d'ajout dans base de données	Idem attendu
Comptes utilisateurs (ajout) : saisi d'un mail déjà existant	Message indiquant que le mail existe déjà / pas d'ajout en BD	06/09 : pas de message mais pas d'ajout en bd 07/09 : idem attend
Comptes utilisateurs (ajout) : clic sur la liste déroulante des rôles	possibilité de sélectionner seulement employé ou vétérinaire	Idem attendu
Comptes utilisateurs (ajout) : saisir les champs avec un mail accessible	Ajout de l'utilisateur dans la liste (et base de données) + envoi d'un mail	Idem attendu
Comptes utilisateurs (modification) : bloquer un	L'utilisateur ne peut plus se connecter	Idem attendu

utilisateur		
Comptes utilisateurs (modification) : débloquent un utilisateur	Un popup s'affiche demandant de saisir un mot de passe. Une fois le mdp validé l'utilisateur doit pouvoir se connecter avec.	Idem attendu
Comptes utilisateurs (modification) : modification d'un élément	L'élément dans être modifié dans la liste	Idem attendu
Comptes utilisateurs : filtrer suivant rôle	Seul les utilisateurs avec le rôle cocher doivent apparaître	Idem attendu
Comptes utilisateurs : filtrer suivant adresse mail	Seul les utilisateurs contenant les caractères saisis doivent apparaître	Idem attendu
Modifier horaires : modifier le texte puis valider	Modifie le footer	Idem attendu
Modifier horaire : saisir un caractère interdit (<)	Input en rouge et le caractère n'apparaît pas	Idem attendu
Habitats : connexion avec tous les rôles, affichage de la page habitat	Les boutons d'ajout et les icônes de modification / suppression n'apparaissent que sur la connexion admin	Idem attendu
Habitats (modifier) : modifier le nom/description puis valider	Un message de réussite s'affiche en bas du formulaire. Le nom/description est modifié	Idem attendu
Habitats (modifier) : modifier la description/nom en saisissant un <	L'input s'affiche en rouge le < ne s'est pas saisi	Idem attendu
Habitats (modifier) : ajouter une image avec description et attribution	Un message de réussite s'affiche en bas du formulaire. L'image apparaît dans l'habitat et l'attribution dans les mentions légales	Idem attendu
Habitats (modifier) :	Un message de réussite	Idem attendu

supprimer une image	s'affiche en bas du formulaire. L'image n'apparaît plus dans l'habitat, ni dans le dossier image, l'attribution n'apparaît plus dans les mentions légales	
Habitats (créer) : saisie de tous les éléments puis valider	Un message de réussite s'affiche en bas du formulaire et l'habitat apparaît dans les habitat	Idem attendu
Habitats (créer) : saisie des mêmes éléments que précédemment puis valider	Un message d'erreur s'affiche. L'habitat n'est pas crée.	Idem attendu
Habitats (créer) : saisir la description/nom en saisissant un <	L'input s'affiche en rouge le < ne s'est pas saisi	Idem attendu
Habitats (supprimer) : clic sur l'icône supprimer d'un habitat	Un popup s'ouvre en demandant la confirmation de la suppression de l'habitat et des animaux y étant présent.	Idem attendu
Habitats (supprimer) : clic sur non	Le popup se ferme rien n'est supprimé	Idem attendu
Habitats (supprimer) : clic sur oui	Le popup se ferme, les animaux et l'habitat sont supprimés (base de données et dossiers images)	Idem attendu
Animaux (modifier) : modification de chaque élément, ajout d'une photo puis valider	Un message de réussite s'affiche en bas du formulaire. Les modifications sont réalisées	Idem attendu
Animaux (créer) : saisir les éléments puis valider	Un message de réussite apparaît en dessous du formulaire. L'animal apparaît dans animaux et dans l'habitat.	Idem attendu
Animaux (créer) : saisir les	Un message d'erreur	Idem attendu



éléments, ajout d'une image dans la taille et le format ne correspond pas aux attentes puis valider	apparaît en bas du formulaire. Les éléments n'ont pas été créés.	
Animaux (supprimer) / dans habitat: clic sur l'icône supprimer	Un popup de confirmation s'affiche	Idem attendu
Animaux (supprimer) / dans habitat: clic sur non	Le popup se ferme. L'animal est toujours là.	Idem attendu
Animaux (supprimer) / dans habitat: clic sur oui	Le popup se ferme. L'animal est supprimé (bd et dossier image). L'habitat ne l'affiche plus.	07/09 : l'habitat tente d'afficher l'animal avec une erreur sur la photo... 07/09 (2) : Idem attendu
Animaux (archiver) : clic sur archiver puis oui	L'animal apparaît avec écrit archiver. Au rechargement de la page il n'apparaît plus dans habitat mais dans animaux.	Idem attendu
Comptes rendus vétérinaire : clic sur la page en étant connecté avec compte admin	La page s'affiche sans possibilité d'ajout de compte rendu	Idem attendu (autres tests voir partie vétérinaire)
Dashboard : clic sur tableau de bord	La page s'affiche avec les éléments dont l'admin a accès	Idem attendu
<i>Espace Employé : cahier des charges</i> <i>Utilisateur concerné : employé</i> <i>Un employé depuis son espace, peut valider un avis ou l'invalidier. Il peut également modifier les services du zoo. L'employé, va donner l'alimentation quotidienne aux animaux, il va ainsi depuis son espace, pouvoir sélectionner un animal et lui ajouter une consommation de nourriture, pour cela il va devoir donner la date ainsi que l'heure, puis, la nourriture donnée ainsi que la quantité.</i>		
Services : accès à la page avec connexion employé	Icônes de modification s'affichent	Idem attendu
Habitats et animaux : clic sur un animal	Icônes pour voir les repas de l'animal et pour le nourrir s'affichent	Idem attendu
Habitat et animaux / animal cliqué : clic sur nourrir	Accède à une page pré-remplie contenant les	Idem attendu

	éléments permettant d'alimenter l'animal	
Ajouter alimentation : vérification des input pré-saisi	Les éléments suivants sont pré-saisi : animal, date, heure	Idem attendu
Ajouter alimentation : saisie de la nourriture et de la quantité puis clic sur soumettre	Ajout des éléments dans la base de données + message de réussite	04/09 : ajout dans la bd ok mais pas de message de réussite 07/09 : idem attendu
Ajouter alimentation : clic sur soumettre sans nourriture ni alimentation	Message indiquant de saisir les champs s'affiche	Idem attendu
Ajouter alimentation : clic sur retour habitat / retour animaux	Doit permettre d'aller sur la bonne page	Idem attendu
Avis : voir la partie avis		
<i>Espace Vétérinaire : cahier des charges :</i> <i>Utilisateur concerné : vétérinaire</i> <i>Un vétérinaire passe quotidiennement dans le zoo, ainsi, depuis son espace, il remplira les comptes rendus par animaux. Le détail est disponible en US 4 Il peut en plus, mettre un commentaire sur les habitats afin qu'il/elle puisse donner son avis son état et s'il faut l'améliorer ou non. Le vétérinaire voit également sur son espace et par animal, tout ce que l'animal a pu manger via la saisie de l'employé sur son espace.</i>		
Habitat ou animaux : clic sur un animal (connexion vétérinaire)	Les icônes des repas, des comptes rendus doivent être visible	Idem attendu
Fiche animal : clic sur nouveau rapport	Nouvelle page avec un formulaire prérempli	Idem attendu
Nouveau rapport : aucune saisie clic sur soumettre	Un message demandant de saisir les champs s'affiche	Idem attendu
Nouveau rapport : saisie de tout sauf le détail puis clic sur soumettre	Ajout dans la bd et un message de réussite s'affiche	04/09 : ajout dans bd ok, impossible de saisir '/' dans le grammage, pas de message de réussite 07/09 : idem attendu
Fiche animal : clic sur rapports médicaux (RM)	Une nouvelle page s'affiche avec les rapports médicaux de l'animal	Idem attendu

RM animal : filtre suivant des dates	Seul les RM entre ces dates doivent s'afficher	04/09 : idem attendu mais le placeholder du nombre à afficher est bizarre 07/09 : idem attendu
RM animal : clic sur en savoir plus d'un RM	Ouverture d'une popup avec le détail	Idem attendu
RM : clic sur la page comptes-rendus vétérinaires depuis menu	Affichage de la page	04/09 : Message d'erreur filtre vétérinaire 07/09 : idem attendu
RM : filtre en fonction de dates	Affichage des comptes rendus seulement entre ses dates	Idem attendu
RM : filtre en fonction vétérinaire	Affichage des RM seulement des véto sélectionnés	Idem attendu
RM : filtre sur une race	Affiche des RM seulement de cette race	06/09 : Idem attendu mais dans races sélectionnés il y a « toutes » 07/09 : idem attendu
RM : clic sur les icônes à gauche	Change de page ou popup nouveau rapport	Idem attendu
Fiche animal : clic sur les repas	Nouvelle page avec la liste des repas pris pour l'animal en question	Idem attendu
Repas animal : filtre sur les dates	Seul les repas pris entre les dates doivent s'afficher	04/09 : idem attendu, attention un var_dump a été oublié ! 04/09 (2) : ok
Repas animal : clic sur les différents icônes en haut à gauche	Doit ouvrir un popup ou amener vers une autre page suivant icônes	Idem Attendu
Repas (all) : clic sur nourrir un animal puis saisi dans l'input animal	Une liste des animaux répondant à la saisie doit s'afficher	Idem attendu
Habitat (commentaires) : voir un habitat après s'être connecté avec Vété	La partie commentaire est présente avec l'icône d'ajout, de page commentaire et les icônes de suppression et d'archivage	Idem attendu

Habitat (commentaires) : voir un habitat après s'être connecté avec Admin/Employé	La partie commentaire est présente avec l'icône de page commentaire et les icônes d'archivage	Idem attendu
Habitat (commentaires) : voir un habitat après sans connexion	La partie commentaire n'est pas visible	Idem attendu
Habitat (commentaires) : clic sur l'icône d'ajout	Une popup s'ouvre avec des éléments pré-rempli	Idem attendu
Habitat (commentaires-ajout) : saisi du commentaire et clic sur ajouter	La popup se ferme et le commentaire est visible à la fin de la liste	Idem attendu
Habitat (commentaires-ajout) : saisie d'un élément interdit : <	L'input devient rouge le caractère n'apparaît pas	Idem attendu
Habitat (commentaires-supprimer) : clic sur l'icône supprimer	Une popup de confirmation s'ouvre	Idem attendu
Habitat (commentaires-supprimer) : clic sur non	La popup se ferme. Le commentaire est toujours présent	Idem attendu
Habitat (commentaires-supprimer) : clic sur oui	La popup se ferme. Le commentaire n'apparaît plus sur le site ni dans la base de données	Idem attendu
Habitat (commentaires-archivage) : clic sur l'icone	Une popup de confirmation s'ouvre.	Idem attendu
Habitat (commentaires-archivage) : clic sur non	La popup se ferme. Le commentaire est toujours présent.	Idem attendu
Habitat (commentaires-archivage) : clic sur oui	La popup se ferme. Le commentaire n'apparaît plus dans habitat mais seulement dans la page commentaires.	Idem attendu
Habitat (commentaires) : clic sur voir tous les commentaires	Affichage d'une nouvelle page « commentaires des vétérinaires »	Idem attendu

Commentaires (ajouter) : clic sur l'icône ajouter	Une popup s'ouvre avec des éléments pré-rempli	Idem attendu
Commentaires : archiver un commentaire puis désarchiver	L'icône doit changer et le commentaire désarchivé ré-apparaît dans la page habitat	Idem attendu
Commentaires : appliquer des filtres sur les habitats et dates	Seul les commentaires concernés doivent s'afficher	Idem attendu
<p>Connexion : cahier des charges</p> <p>Utilisateur concerné : Administrateur, vétérinaire, employé</p> <p>Seul une personne de type administrateur, vétérinaire ou encore employé peut se connecter. Un visiteur ne peut pas devenir utilisateur, et donc, ne peut pas se créer de compte. Pour se connecter, il devra juste saisir son username (mail) suivi de son mot de passe.</p>		
Connexion : saisi d'un mauvais mot de passe	Un message indique le mail ou le mdp est faux	Idem attendu
Connexion : saisi 3x d'un mauvais mot de passe puis saisi du bon mot de passe	Un message indiquant que le compte est bloqué s'affiche. Il n'est plus possible de se connecter.	07/09 : au 4e essai toujours pas bloqué... 07/09 (2) : idem attendu
Connexion : saisi 3x mauvais mot de passe en sortant du popup à chaque fois	Le compte est bloqué	Idem attendu
<p><i>Contact : cahier des charges :</i></p> <p><i>Utilisateur concerné : Visiteur</i></p> <p><i>Un visiteur peut contacter le zoo s'il le souhaite, pour cela, il devra accéder à la page contact depuis le menu applicatif. Pour donner suite à cela, il aura accès à un formulaire qui va lui demander un titre, une description ainsi que son mail afin qu'il puisse obtenir une réponse. Pour donner suite à cet envoi, la demande est envoyée par mail au zoo. L'employé peut répondre à la demande directement par mail.</i></p>		
Contact : saisi de caractères interdit dans les inputs	L'input est rouge et le caractère n'apparaît pas	Idem attendu
Contact : saisi de tous les éléments sauf structure et validation	Un message userfriendly apparaît à la place du formulaire. Un mail est envoyé à l'adresse du zoo.	Idem attendu
Contact : modifier la taille de	La mise en page s'adapte	Idem attendu

l'écran		
<i>Statistique sur la consultation des habitats : cahier des charges</i> <i>Utilisateur concerné : Visiteur</i> <i>Un visiteur, quand il cliquera sur l'animal, va augmenter la consultation de 1 pour l'animal donné. Cette information devra être stocké dans une base de données non relationnelle. Par exemple, si un visiteur consulte l'animal portant le prénom "Médor", alors le compteur de "Médor" passera à 2 (s'il avait eu 1 avant). Ces données sont exploitées dans le Dashboard administrateur afin que José puisse visualiser quels animaux plaisent le plus. José conclue sur le fait que vous êtes dans l'obligation de déployer votre / vos applications. Un schéma de données possible vous a été communiqué en annexe 1, vous pouvez l'implémenter, ou, faire le vôtre.</i>		
Pages habitat/animal : clic sur un animal sans être connecté	Les stats de l'animal augmente de 1	Idem attendu
Pages habitat/animal : clic sur un animal en étant connecté	Les stats de l'animal ne bougent pas	Idem attendu
Dashboard : affichage de la page sans être connecté	Erreur 404	Idem attendu
Dashboard ; affichage de la page en étant connecté	Affichage de la page	Idem attendu J'aurai demandé à José si cela lui convenait car ce n'était pas prévu comme ça mais je trouvais ça plus sympa.
Dashboard : clic sur plus de popularité	La liste des animaux apparaît avec le nombre de clic par animal	Idem attendu
Dashboard : modification de la taille de l'écran	La mise en page s'adapte	07/09 : 1 image et 2 textes dépassent des tuiles 07/09 (2) : idem attendu
Dashboard : clic sur chaque tuile	Doit permettre d'accéder à la page nommée	Idem attendu