

Benad
Kilian
BTS-SIO

Dossier E6

BTSSiO

*

cci
campus
ALSACE

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS		SESSION 2025
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle (recto)		
Épreuve E6 - Conception et développement d'applications (option SLAM)		

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 2
Nom, prénom : Benad kilian		N° candidat : 02442761565
Épreuve ponctuelle <input type="checkbox"/>	Contrôle en cours de formation <input checked="" type="checkbox"/>	Date : 29 / 04 / 2025
Organisation support de la réalisation professionnelle Interface graphique pour Multipass		
Intitulé de la réalisation professionnelle : Interface graphique Multipass		
Période de réalisation : 20/01/2025 au 25/01/2025 Lieu : CCI campus		
Modalité : <input checked="" type="checkbox"/> Seul(e) <input type="checkbox"/> En équipe		
Compétences travaillées <input checked="" type="checkbox"/> Concevoir et développer une solution applicative <input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative <input checked="" type="checkbox"/> Gérer les données		
Conditions de réalisation¹ (ressources fournies, résultats attendus) L'objectif du projet est de concevoir et développer une application JAVA capable d'interagir Multipass, afin de permettre aux utilisateurs de : Lister, démarrer, arrêter, suspendre, crée, supprime des instance Multipass. Les résultats attendus comprennent une application JAVA intuitive, capable de communiquer efficacement avec Multipass, ainsi qu'une documentation technique détaillant l'architecture, l'installation et l'utilisation de la solution.		
Description des ressources documentaires, matérielles et logicielles utilisées² Document fournie : <ul style="list-style-type: none"> - Enoncer projet 3 : liste des fonctionnalités - Documentation AP 4 - Documentation utilisateur Ressource matériel et logiciel : Environnement de programmation : Visual Studio Code, IntelliJ IDEA. Langage utiliser : <ul style="list-style-type: none"> - JAVA - FXML, CSS 		
Modalités d'accès aux productions³ et à leur documentation⁴ Lien Gitlab : https://gitlab.com/kilian67/ap4-multipass.git Portfolio : https://kilianbenad.fr/		

¹ En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

² Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

³ Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

⁴ Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)****Épreuve E5 - Conception et développement d'applications (option SLAM)****Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

Ce projet ayant pour but de réaliser une interface graphique à Multipass, j'avais défini les besoins suivants :

- Liste des interfaces
- Possibilité de démarrer les instances Multipass
- Possibilité de stopper les instances Multipass
- Possibilité de suspendre les instances Multipass
- Possibilité de crée des instances Multipass
- Possibilité de supprimer des instances Multipass
- Possibilité d'avoir les détails des instances Multipass

Dans la réalisation de ce projet, j'ai eu la chance de réaliser les éléments suivants :

La documentation :

- Les maquettes
- La documentation
- Les branches Gitlab et la gestion du projet étapes par étapes

La partie graphique donc :

- La page de créations des instances
- La page de gestion des instance (liste des instance Multipass, option démarrage, suspension, arrêt et suppression)
- La page détail contenant quelque détail sur l'instance dont on a cliqué

Mais également le back avec :

- Le contrôleur base pour la base du logiciel
- Le contrôleur pour la page création
- Le contrôleur pour la page détail
- Le contrôleur pour la page avec liste instance
- Le contrôleur pour chaque ligne d'instance

Mais également :

- La gestion des différentes scènes
- La gestion de l'exécution des commandes Multipass
- La gestion des threads

Lors de la réalisation de ce projet un cahier des charges a pu être établie vis-à-vis des demandes attendues dans l'énoncer du projet :

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)****Épreuve E5 - Conception et développement d'applications (option SLAM)****Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

Résumer des commandes à exécuter dans Multipass :

Listing des instances : `multipass list`

Démarrage d'une instance : `multipass start` (nom de l'instance Multipass)

Suspension d'une instance : `multipass suspend` (nom de l'instance Multipass)

Création d'une instance : `multipass launch -n` (nom de l'instance multipass)

L'option `-n` signifie le fait de vouloir ajouter un nom à l'instance créée sinon le nom sera aléatoire

Suppression d'une instance : `multipass delete -p` (nom de l'instance)

L'option `-p` signifie que l'on veut purger par la même occasion l'instance en même temps que la suppression

Arrêt d'une instance : `multipass stop` (nom de l'instance)

Détail d'une instance : `multipass info` (nom de l'instance)

Architecture du projet :

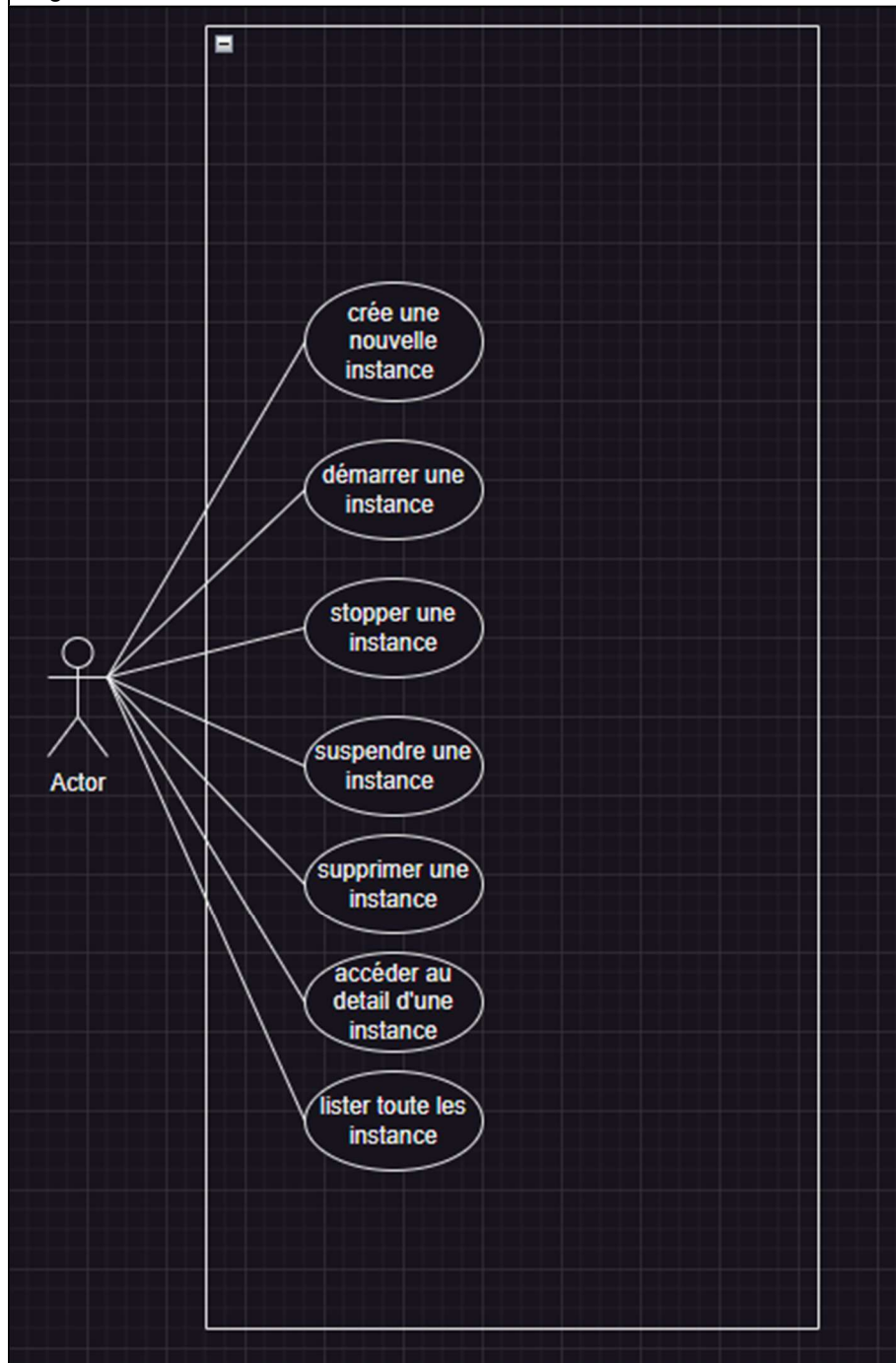
- Main
 - Java
 - Main
 - controller
 - BaseController.java
 - DetailController.java
 - HomeController.java
 - InstanceController.java
 - NewInstanceController.java
 - handler
 - SceneManager.java
 - InstanceSelectionManager.java
 - LoggerHandler.java
 - MultipassManager.java
 - MultipassMonitor.java
 - Observer.java
 - TaskManager.java
 - ressources
 - icon
 - instance.png
 - newInstance.png
 - logo.png
 - style
 - style.css
 - views
 - home
 - home.fxml
 - instanceRow.fxml
 - base.fxml
 - detail.fxml
 - newInstance.fxml

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)****Épreuve E5 - Conception et développement d'applications (option SLAM)**

Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs

Durant la production de l'application j'ai pu définir des choses à mettre en place pour l'utilisateur :

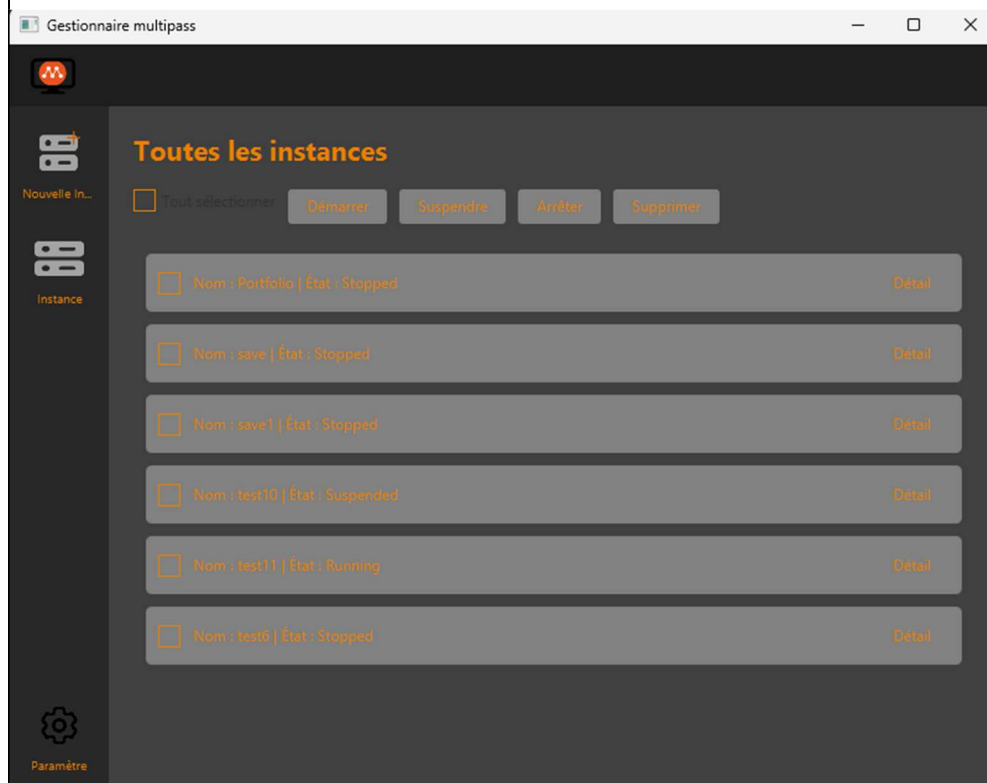
Diagramme de cas d'utilisation :



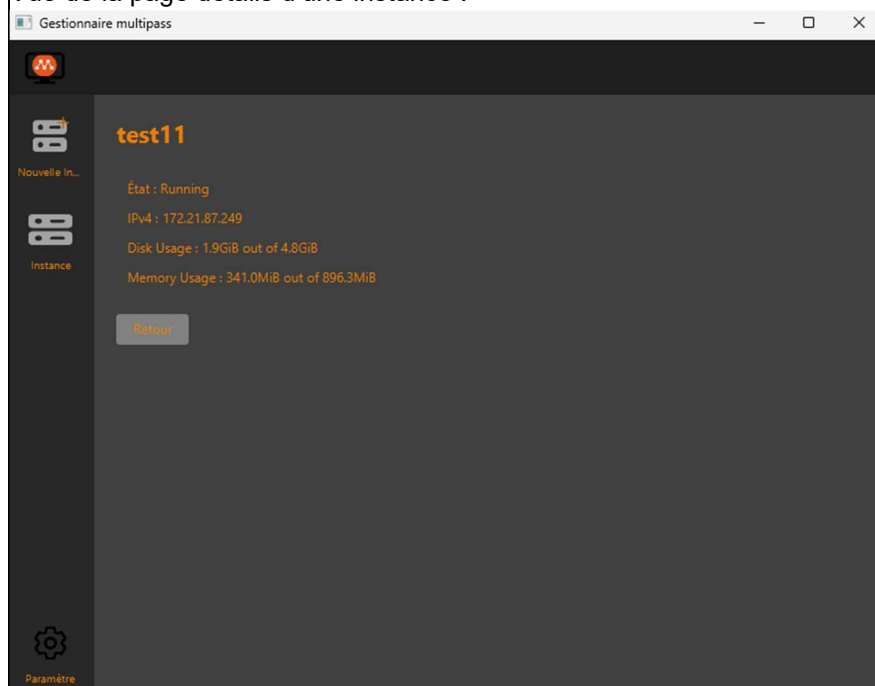
**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)****Épreuve E5 - Conception et développement d'applications (option SLAM)****Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

J'ai également les différentes vues de chaque page préparer en amont grâce aux maquettes :

Vue de la page de liste des instances :



Vue de la page détails d'une instance :

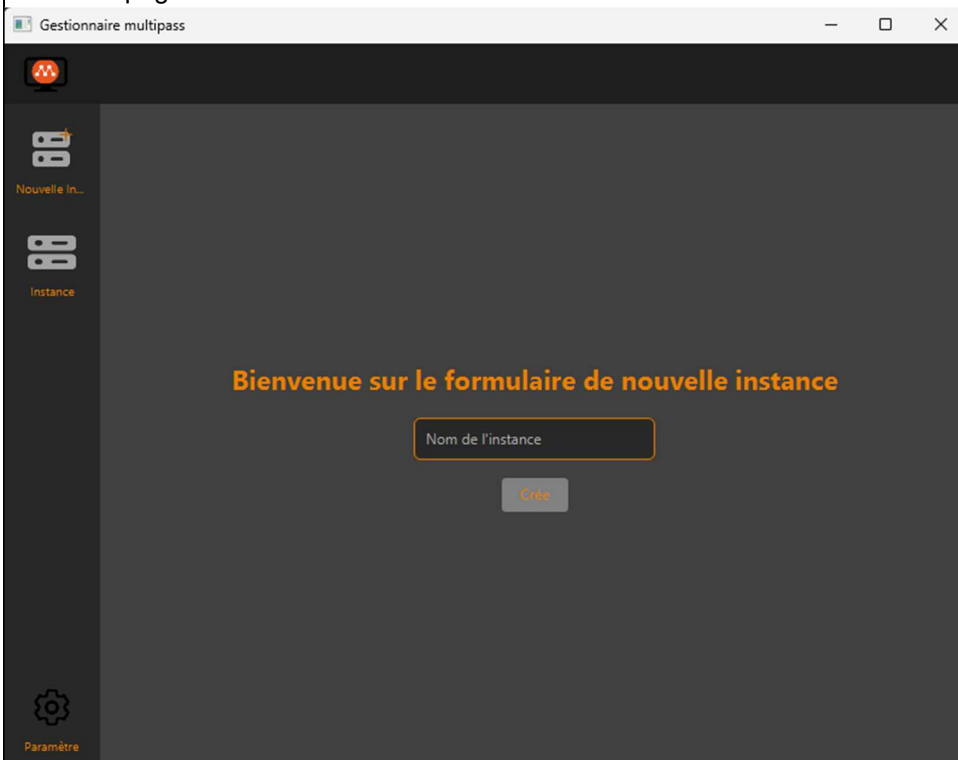


**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)**

Épreuve E5 - Conception et développement d'applications (option SLAM)

Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs

Vue de la page de création d'une instance :



The screenshot shows a web browser window titled "Gestionnaire multipass". The interface has a dark theme. On the left, there is a sidebar with three icons: a red one at the top, a grey one in the middle, and a gear icon at the bottom. The middle icon is highlighted with an orange glow and labeled "Nouvelle in..." and "Instance". The main content area is dark grey and displays the text "Bienvenue sur le formulaire de nouvelle instance" in orange. Below this text is a text input field with the placeholder "Nom de l'instance" and a "Crée" button.

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)**

Épreuve E5 - Conception et développement d'applications (option SLAM)

Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2024
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle (recto)	
Épreuve E5 - Conception et développement d'applications (option SLAM)	

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 1
Nom, prénom : Benad kilian		N° candidat : 02442761565
Épreuve ponctuelle <input type="checkbox"/>	Contrôle en cours de formation <input checked="" type="checkbox"/>	Date : 29 / 04 / 2025
Organisation support de la réalisation professionnelle Du code exécuté sur le système d'exploitation d'un serveur.		
Intitulé de la réalisation professionnelle Atelier de professionnalisation 3		
Période de réalisation : Septembre 2024 – Janvier 2025		Lieu : Cci Campus Alsace Strasbourg
Modalité : <input type="checkbox"/> Seul(e) <input checked="" type="checkbox"/> En équipe		
Compétences travaillées X Concevoir et développer une solution applicative X Assurer la maintenance corrective ou évolutive d'une solution applicative X Gérer les données		
Conditions de réalisation¹ (ressources fournies, résultats attendus) Lister, démarrer, arrêter, modifier et supprimer des conteneurs. Gérer les images Docker. Visualiser les logs et les statistiques des conteneurs. Les résultats attendus comprennent une application web intuitive, robuste et sécurisée, capable de communiquer efficacement avec l'API Docker, ainsi qu'une documentation technique détaillant l'architecture, l'installation et l'utilisation de la solution.		
Description des ressources documentaires, matérielles et logicielles utilisées² Ressources documentaires : <ul style="list-style-type: none"> • Cahier des charges et spécifications fonctionnelles fournies Ressources matérielles : <ul style="list-style-type: none"> • Poste de travail avec configuration adaptée pour le développement et les tests Ressources logicielles : <ul style="list-style-type: none"> • Environnement de développement intégré (EDI) : Visual Studio Code, PhpStorm • Langages et frameworks : <ul style="list-style-type: none"> ○ PHP ○ SQL ○ Twig • Technologies et outils : <ul style="list-style-type: none"> ○ Docker ○ Systèmes de gestion de base de données Mariadb ○ Outils de gestion de versions Git et hébergement sur GitLab ○ Serveur web pour déploiement Apache ○ Mermaid et PlantUML pour la documentation 		
Modalités d'accès aux productions³ et à leur documentation⁴ L'accès aux productions et documentations se font sur gitlab à l'url suivante : https://gitlab.com/DamienReichhart/CCI-BTS-SIO-23-25-Atelier-Professionnalisation-3 La documentation du projet est également accessible à cette url : https://kilianbenad.fr/		

¹ En référence aux conditions de réalisation et ressources nécessaires du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

² Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

³ Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

⁴ Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

Table des matières

Cahier des charges fonctionnel.....	3
Formalisme	3
Gestion de projet.....	3
Réalisations.....	4
Architecture	5
Documentation technique	6
Diagrammes de classe	6
Diagramme de cas d'utilisation	6
Diagramme de contexte statique.....	7
Modèle entité association	7
Diagrammes de séquence	9
Diagramme de séquence des réseaux docker	9
Diagramme de séquence des requêtes http	9
Diagramme des volumes docker.....	10
Diagramme du lancement de l'application	11
Diagramme des couches de sécurité docker	12
Cycle de vie de la requête http.....	13
Intégration docker dans php	14
Diagramme des flux de données.....	15
Diagramme des étapes des middlewares	16
Diagramme de représentation du MVC.....	17
Captures d'écrans de l'application	18
Page de connexion.....	18
Gestion d'un conteneur.....	19
Gestion d'un serveur.....	Erreur ! Signet non défini.
Gestion des serveurs	19
Gestion des dockerfiles.....	20
Gestion des utilisateurs	20

Cahier des charges fonctionnel.

Une application web intuitive, robuste et sécurisée pour docker, capable de communiquer efficacement avec l'API Docker, ainsi qu'une documentation technique détaillant l'architecture, l'installation et l'utilisation de la solution.

Les besoins fonctionnels de la solution ont été définis comme tel :

- Gestion du conteneur
 - Possibilité de lancer des conteneurs
 - Possibilité de stopper des conteneurs
 - Possibilité de modifier/supprimer/ajouter des attribues aux conteneurs
 - Possibilité de visualiser les conteneurs en cours / arrêtés / en erreur
 - Possibilité de supprimer un conteneur
 - Possibilité de modifier un conteneur
- Possibilité de gérer les comptes / utilisateur pour le super admin
- Possibilité d'ajouter / modifier des dockerfiles personnalisés destinés à la solution
- Possibilité de visualiser le log du conteneur
- Possibilité d'ajouter un serveur à manager

Formalisme

Pour ce projet, un formalisme des commits et des tickets a été mis en place afin de garantir un suivi de projet de qualité.

Pour cela, le formalisme suivant a été défini avec les préfixes suivants :

[FIX]	Définit une correction dans un message de commit
[FEATURE]	Définit que le ticket représente une fonctionnalité ou que le commit représente l'ajout de Fonctionnalité
[RELEASE]	Définit que le commit représente une release
[DOC]	Définit que le ticket ou le commit représente un ajout ou une modification de documentation
[HOTFIX]	Définit que le ticket ou le commit a réalisé une correction de bug rapide et urgente sur une des branches de production (preprod / main)

Gestion de projet

Le projet a été mené selon la méthodologie Waterfall, ou cascade en français.

Dans ce cadre, l'ensemble des tâches et des étapes du projet ont été planifiées en amont, puis structurées sous forme de tickets et de jalons (Milestones).

Chaque membre de l'équipe s'est vu attribuer des tickets correspondant aux fonctionnalités à développer ou aux bugs à corriger. Cela permettait un suivi clair de l'avancement, une activité en cours était facilement identifiable grâce aux tickets assignés, assurant ainsi une bonne coordination au sein de l'équipe.

Réalisations

Dans la réalisation de ce projet, j'ai eu la chance de réaliser les éléments suivants :

- Création des maquettes
- Documentation UML sur les conteneurs
- cahier des charges
- Gestion des utilisateurs :
 - Ajout / suppression / modification des utilisateurs
- page d'accueil
- page gestion des utilisateurs
- page gestion des serveurs
- page gestion des conteneurs
- page gestion des docker files

Architecture

Voici l'architecture des fichiers du projet :

Racine du projet

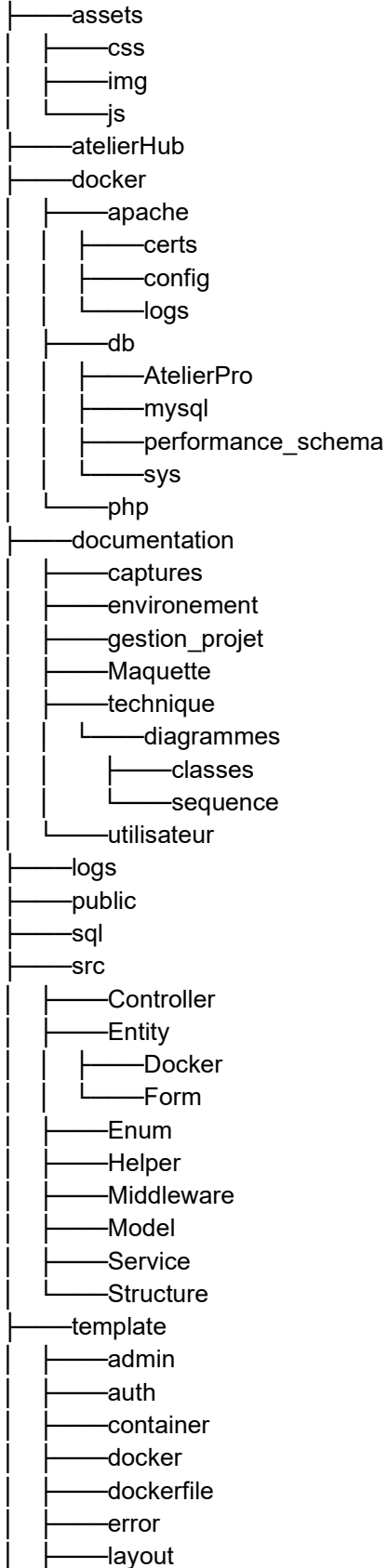
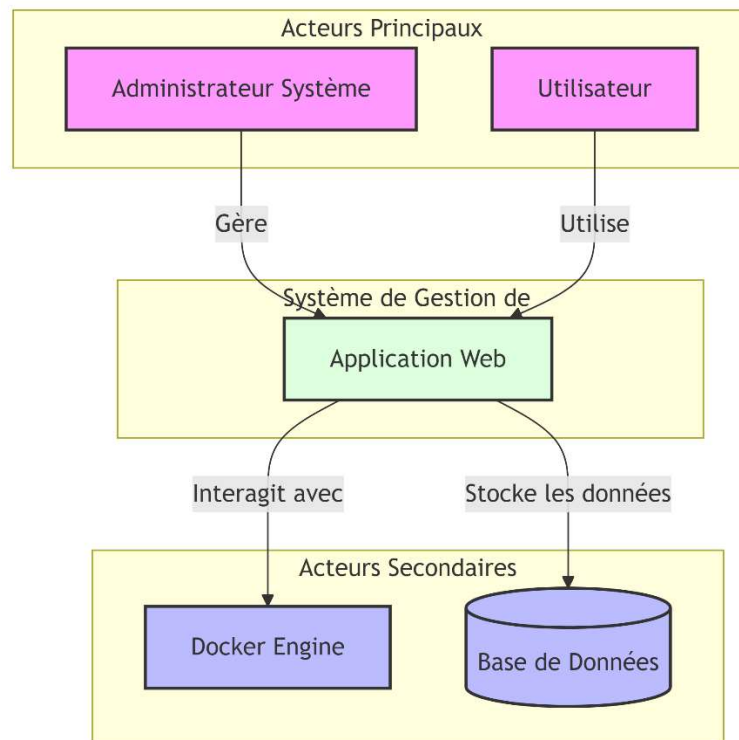
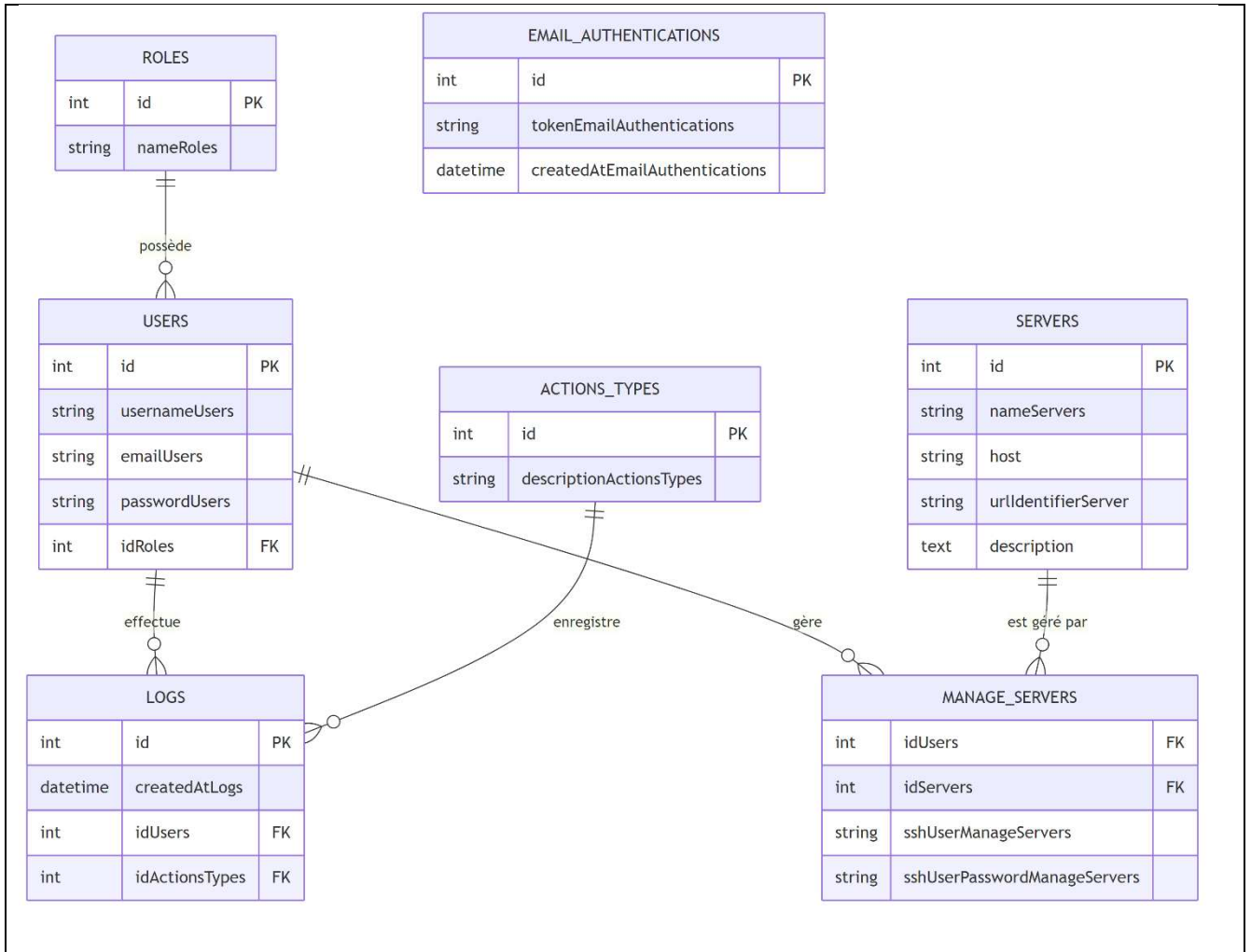


Diagramme de contexte statique



Modèle entité association



Diagrammes de séquence

Diagramme de séquence des réseaux docker

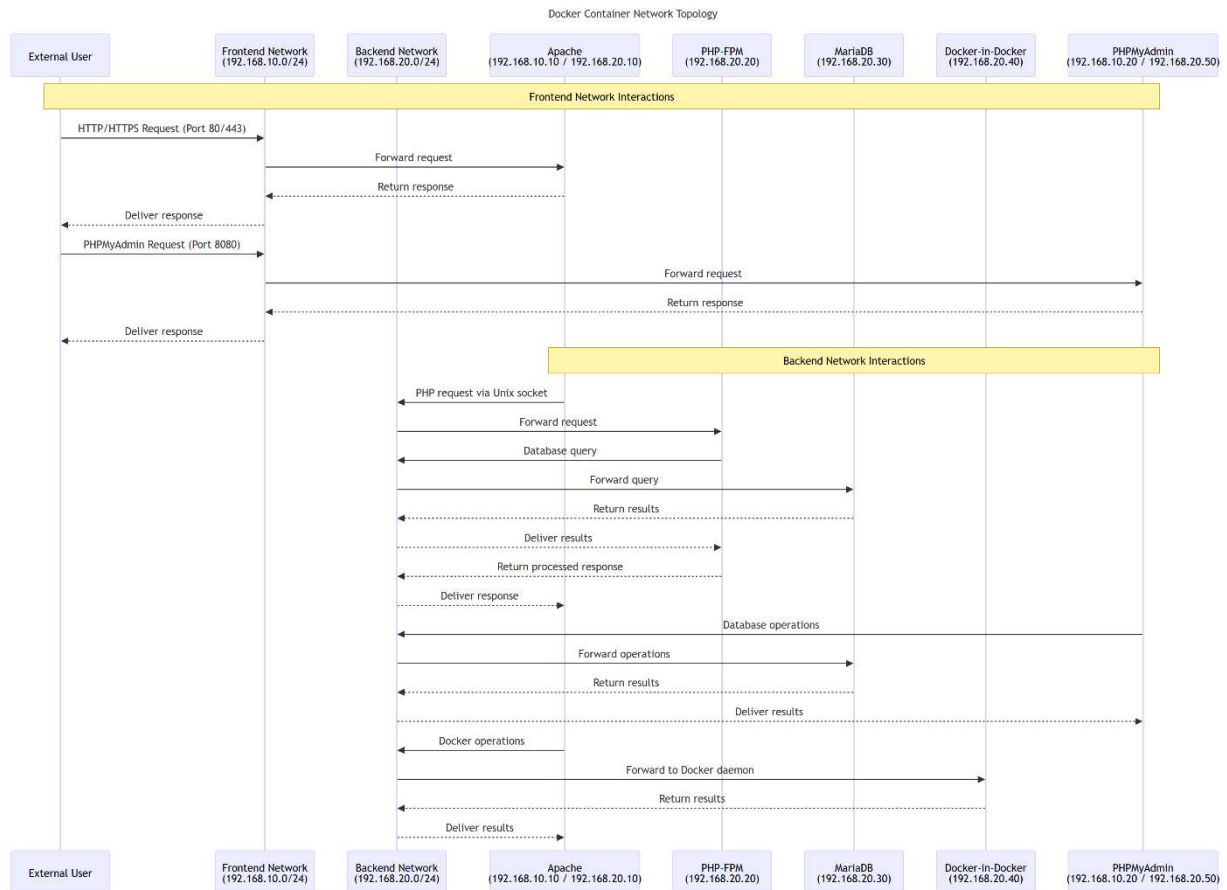


Diagramme de séquence des requêtes http

HTTP Request Flow Through Docker Infrastructure

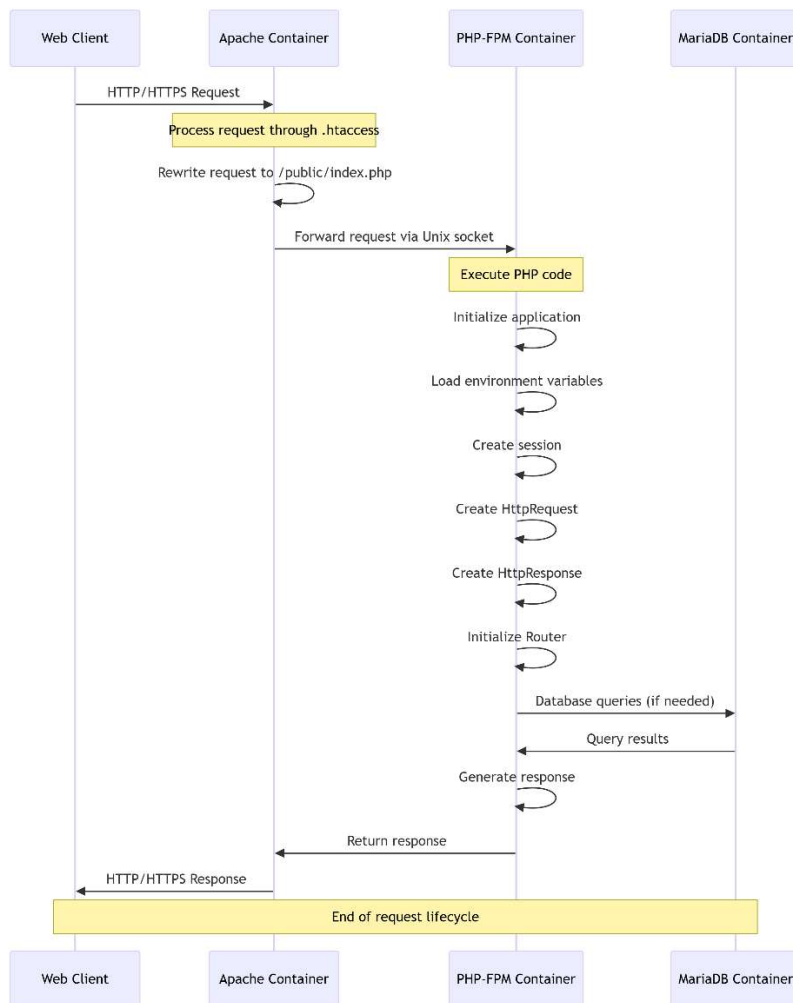


Diagramme des volumes docker

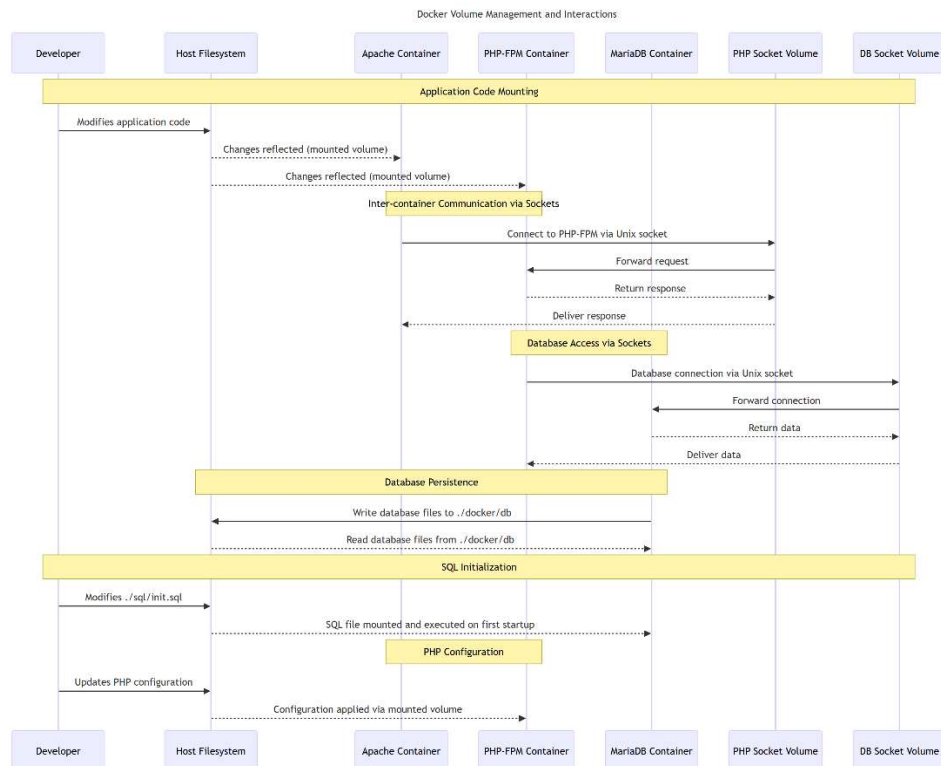


Diagramme du lancement de l'application

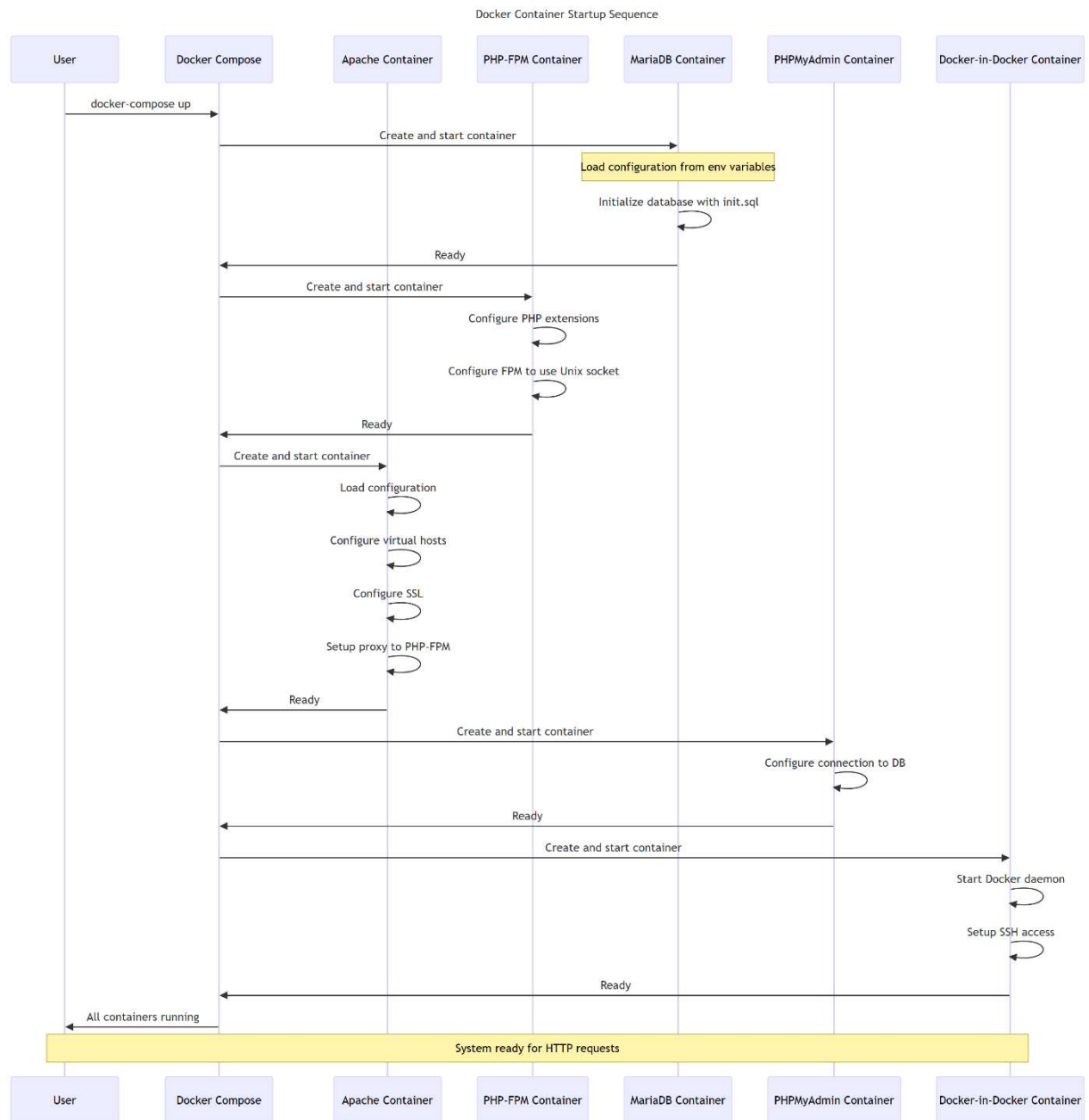
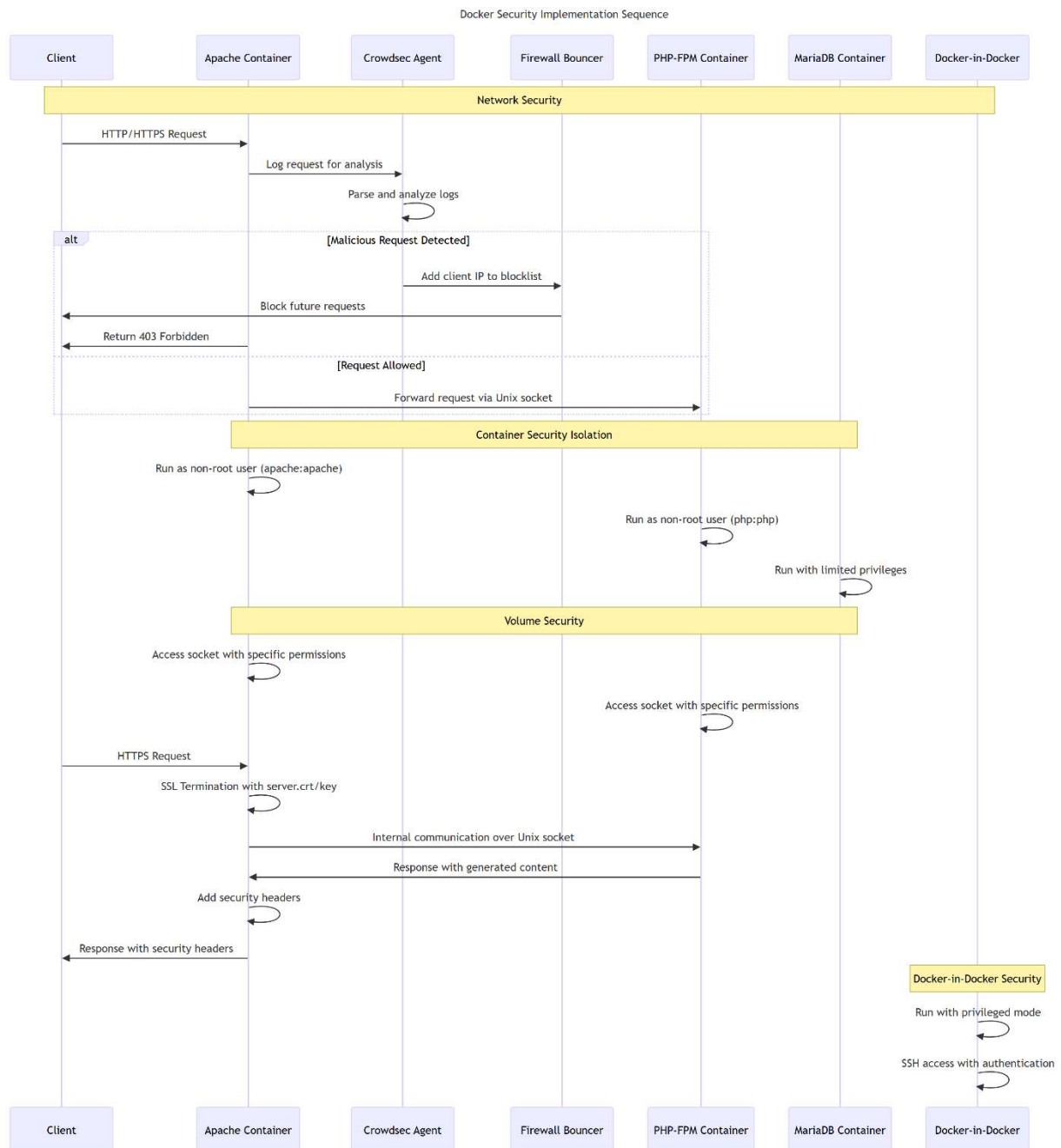
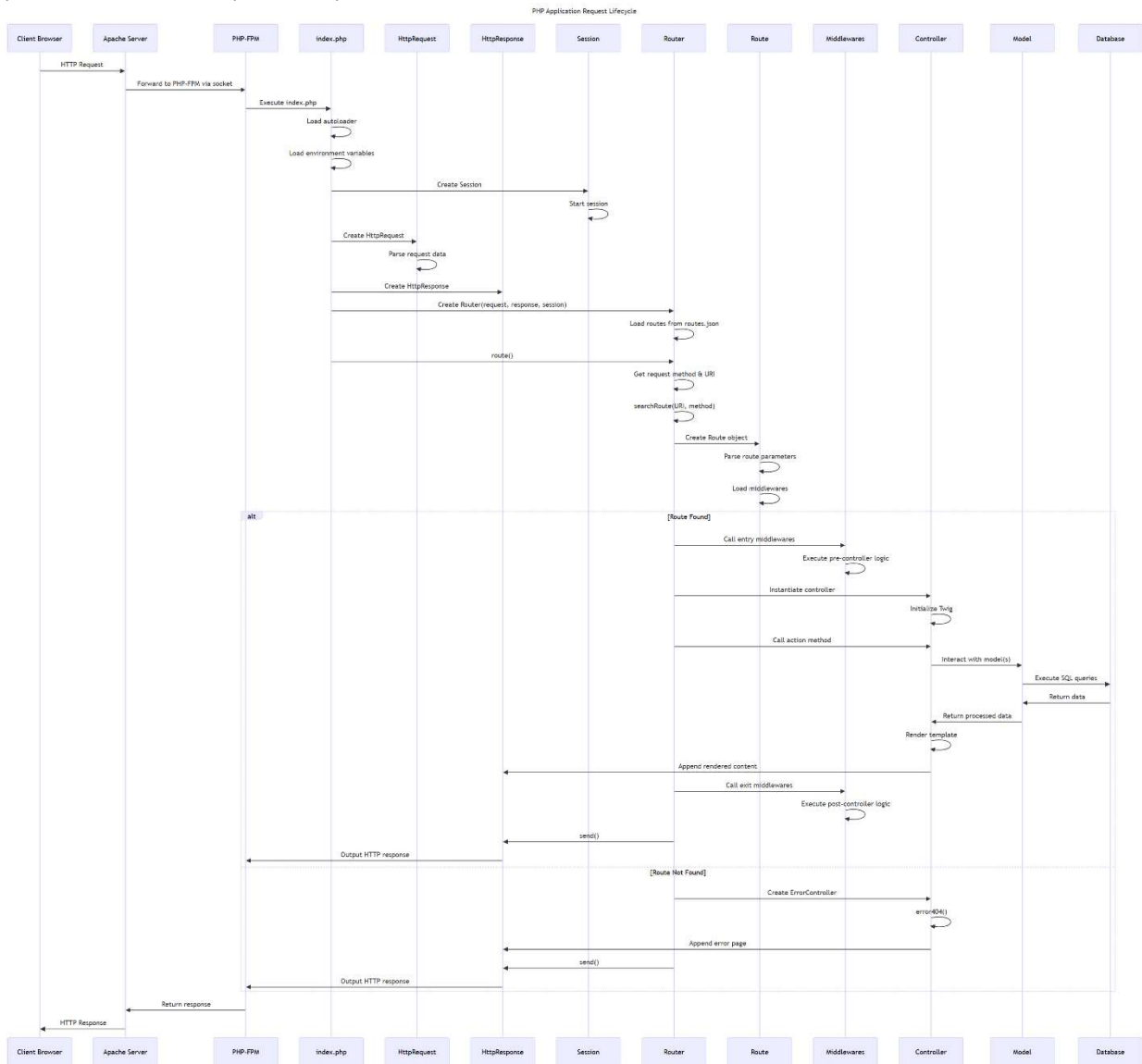


Diagramme des couches de sécurité docker



Cycle de vie de la requête http



Intégration docker dans php

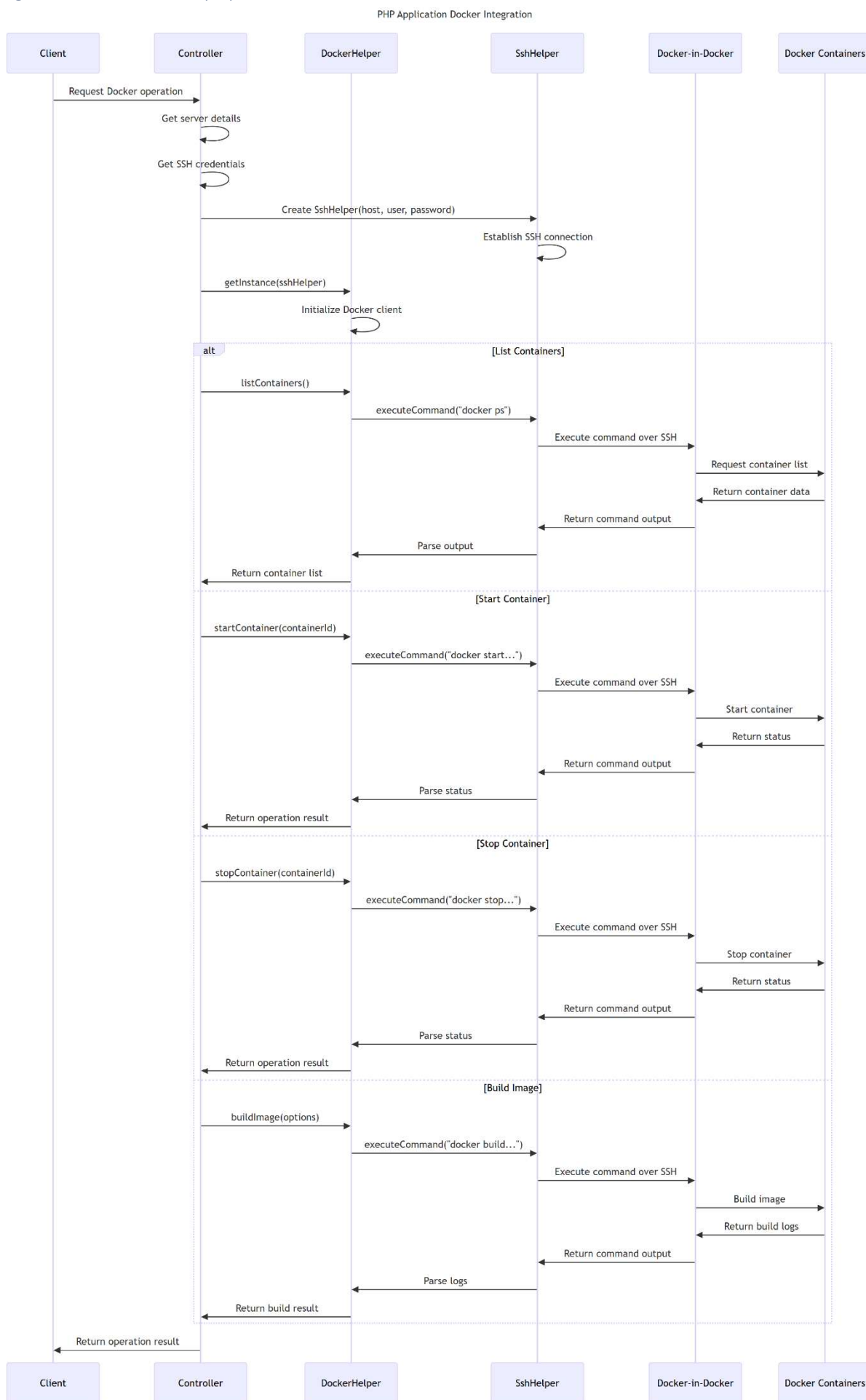
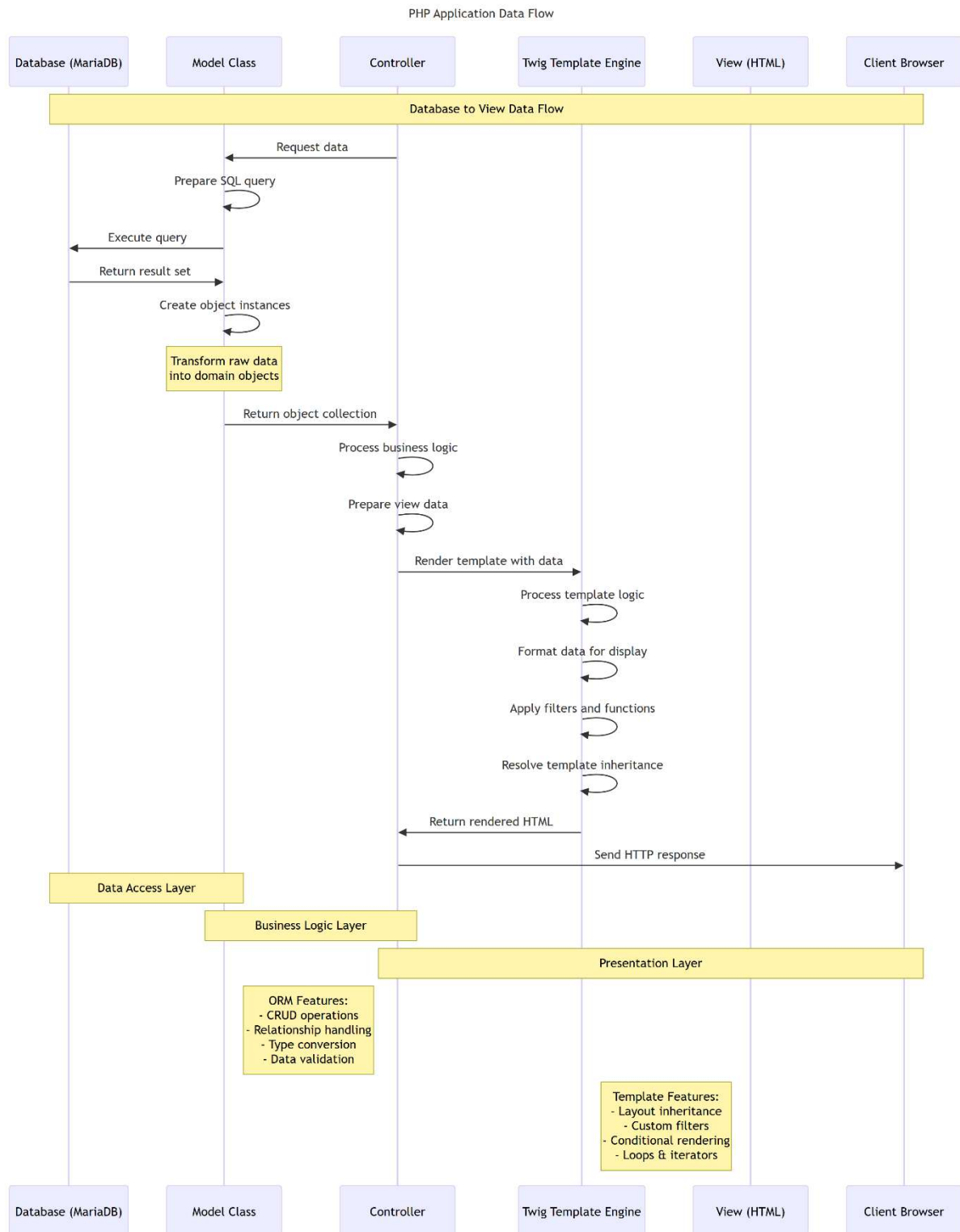


Diagramme des flux de données



Digramme des étapes des middlewares

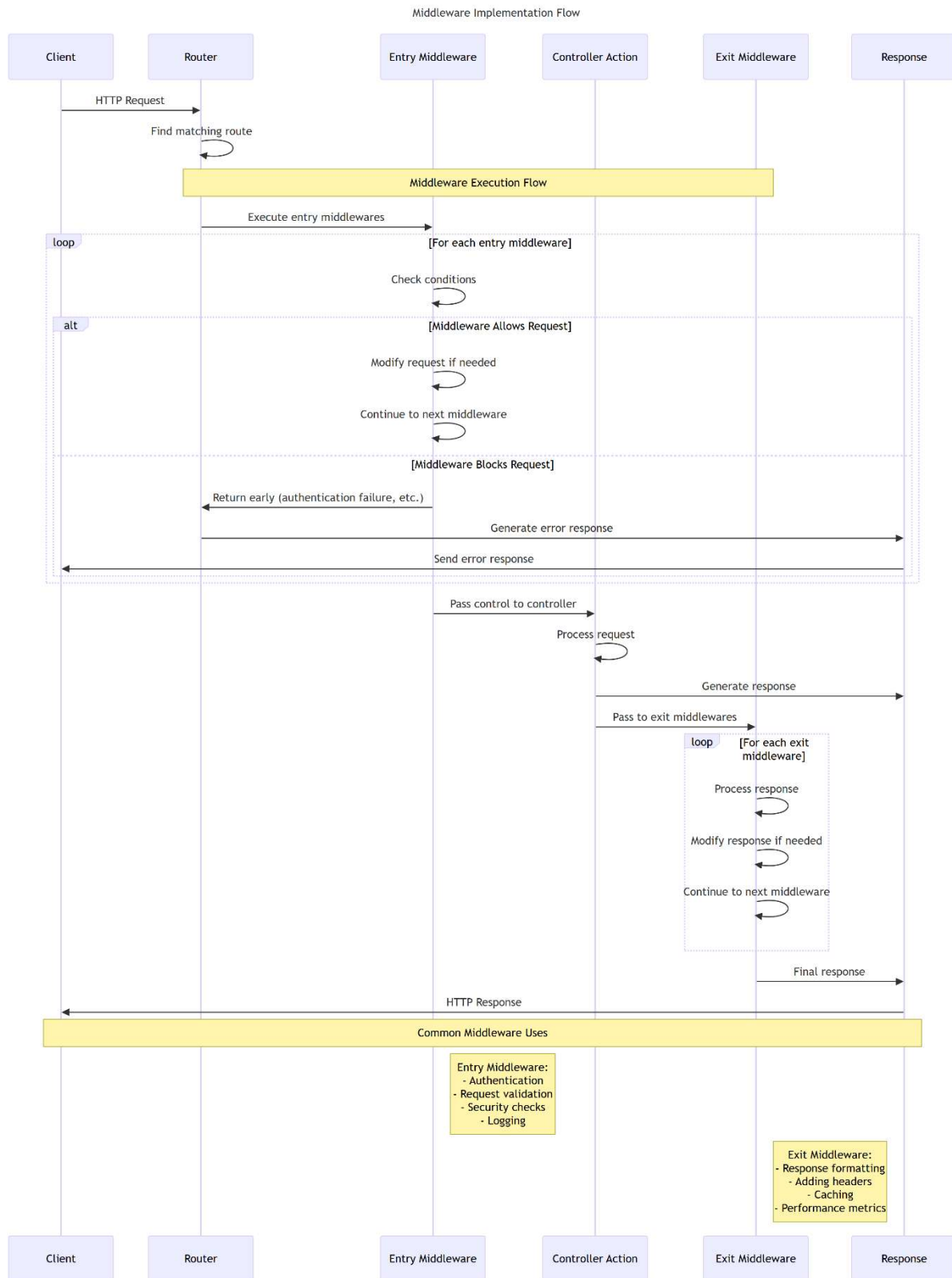
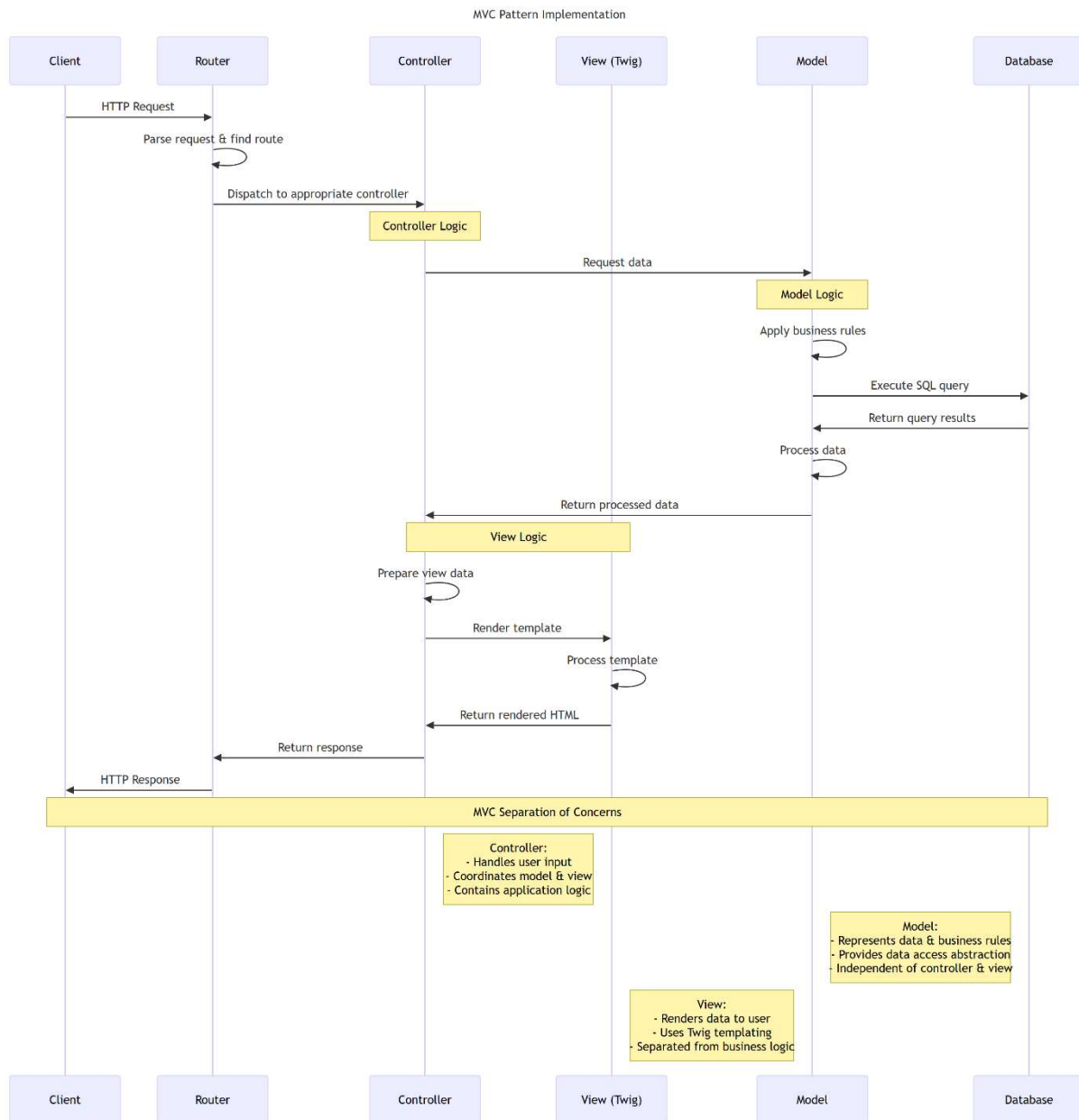


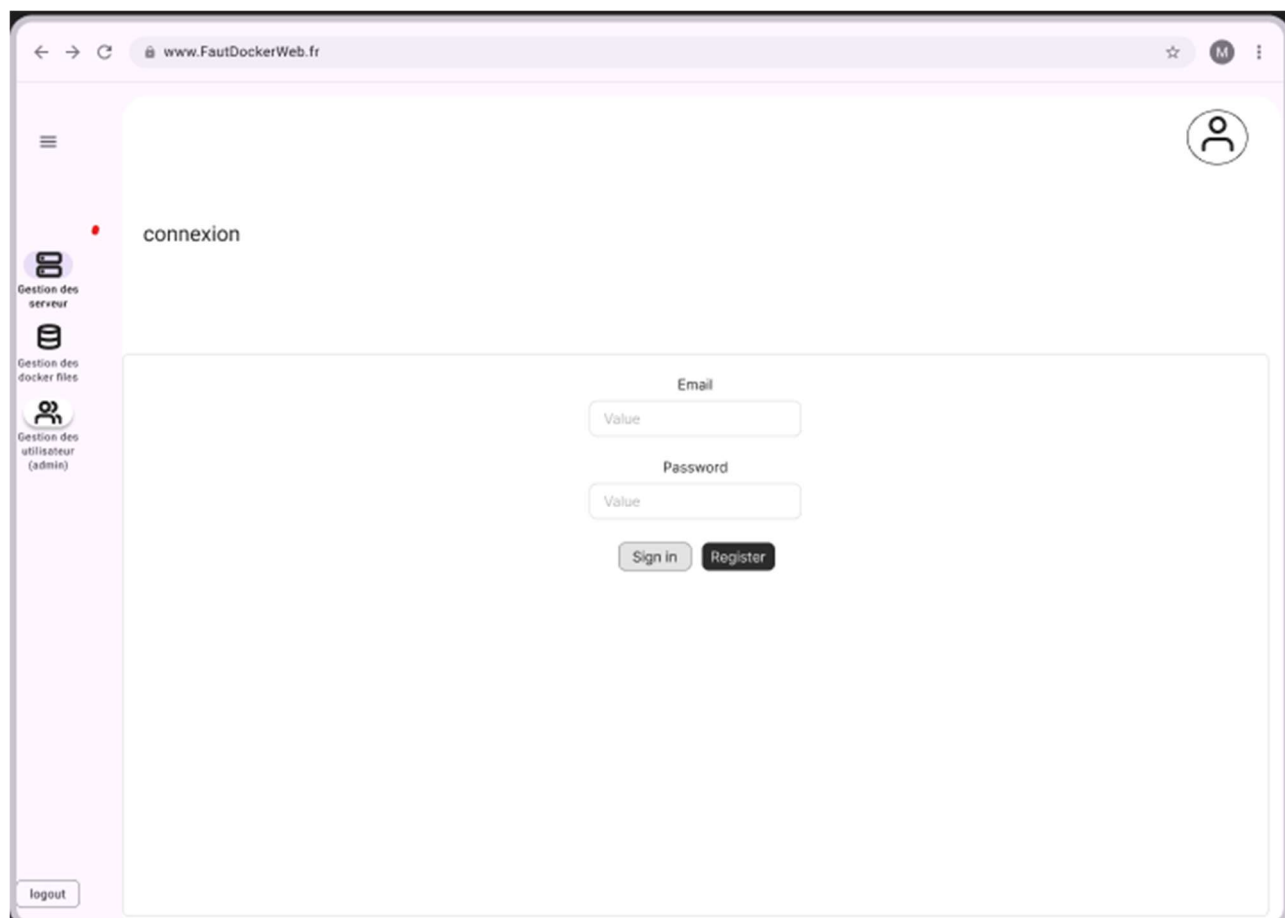
Diagramme de représentation du MVC



Captures d'écrans de l'application

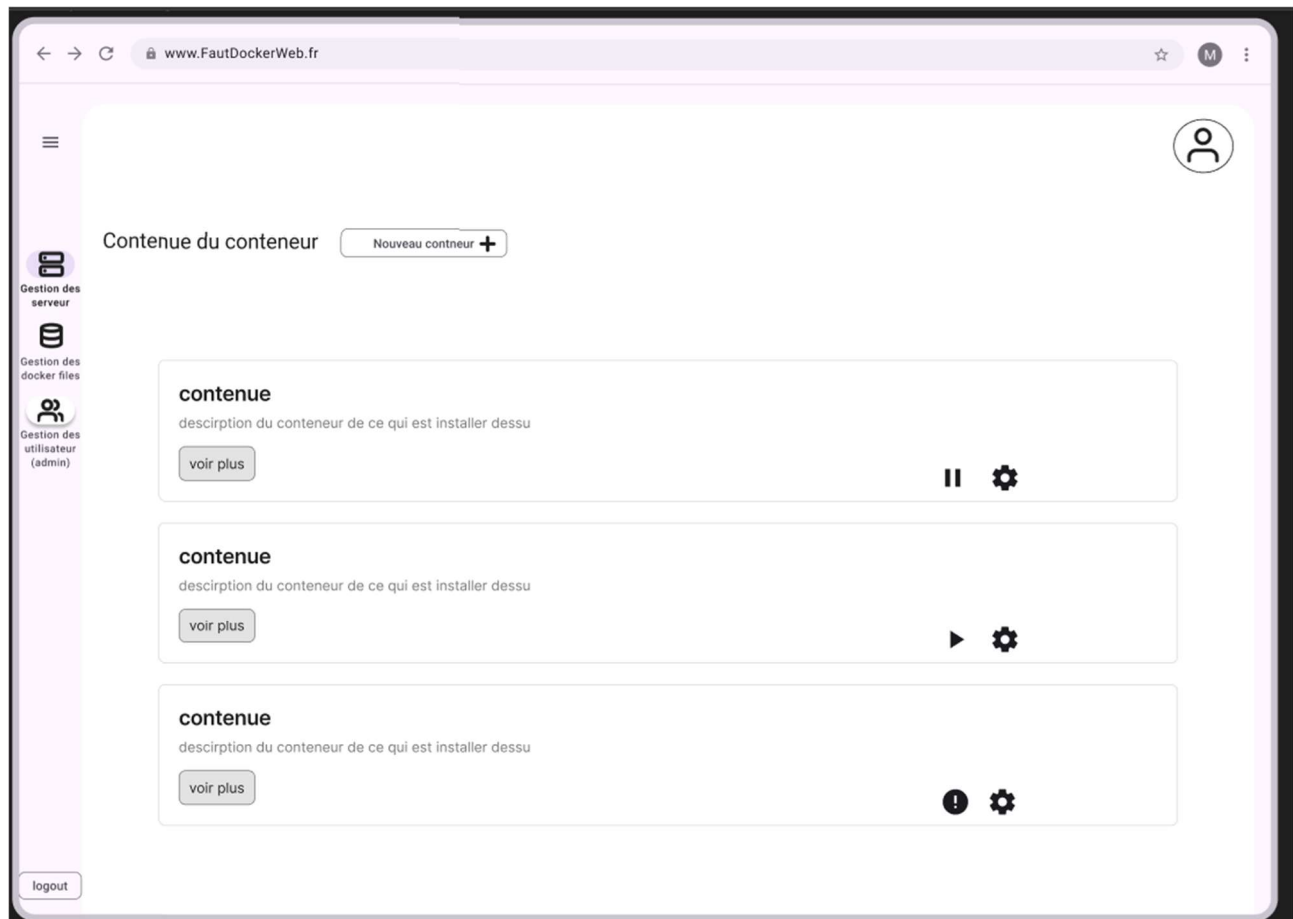
Vous trouverez ci-dessous des captures d'écran de l'application :

Page de connexion

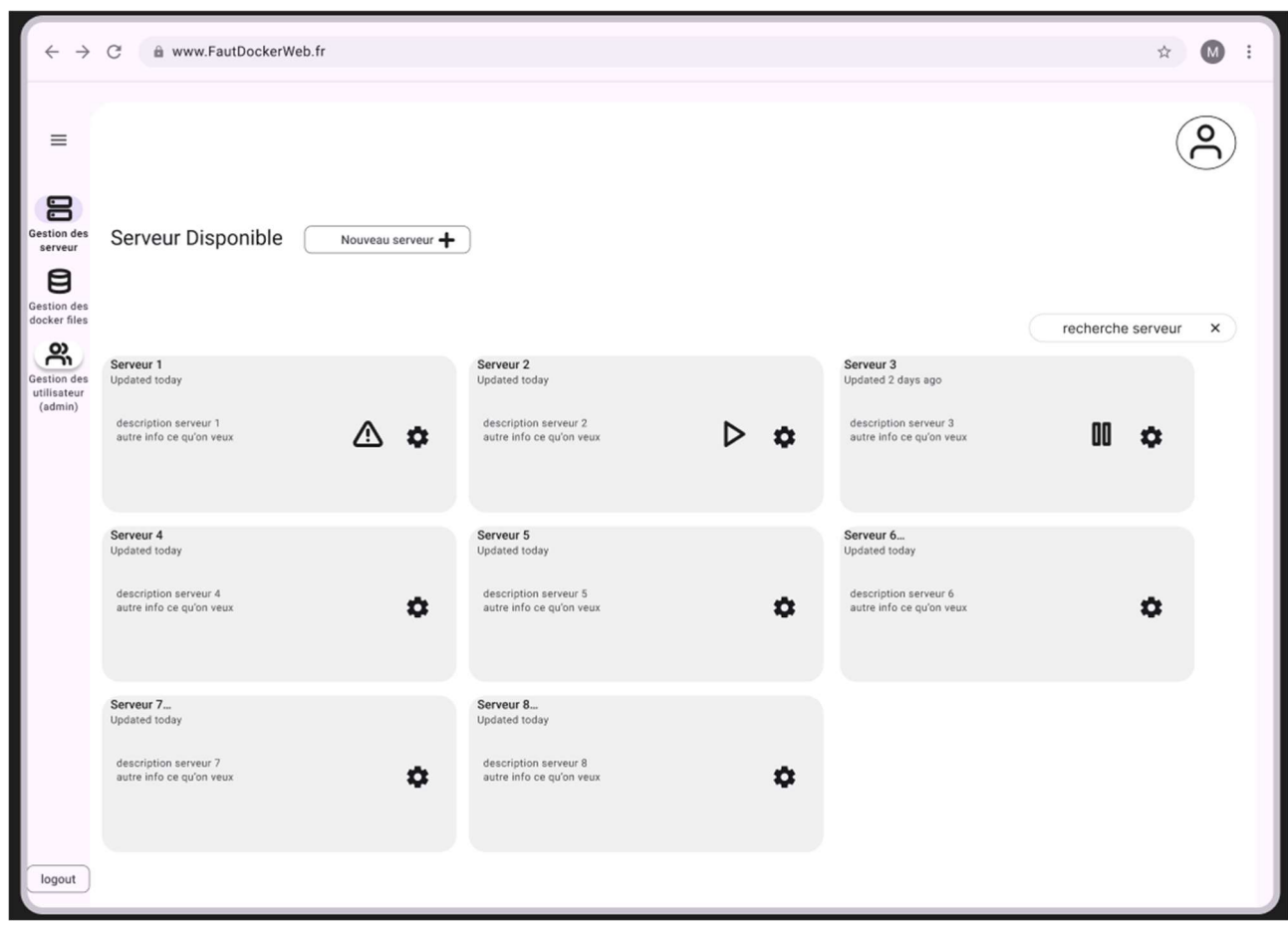


The screenshot shows a web browser window with the address bar displaying `www.FautDockerWeb.fr`. The page has a light pink header and a sidebar on the left. The sidebar contains three menu items: "Gestion des serveur" (with a server icon), "Gestion des docker files" (with a Docker icon), and "Gestion des utilisateur (admin)" (with a user icon). The main content area is titled "connexion" and features a login form. The form has two input fields: "Email" and "Password", both with placeholder text "Value". Below the fields are two buttons: "Sign in" and "Register". A "logout" button is located in the bottom left corner of the sidebar.

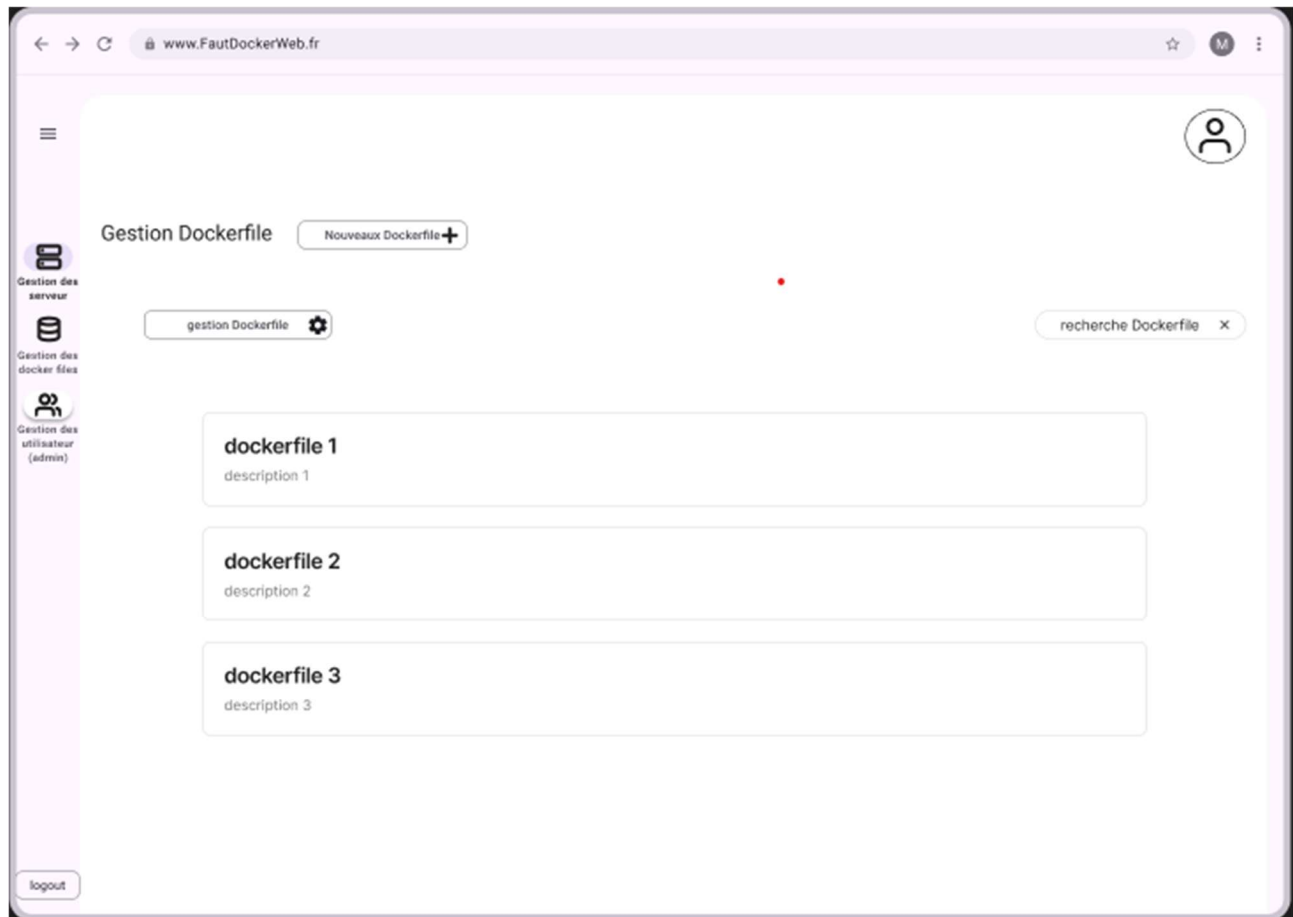
Gestion d'un conteneur



Gestion des serveurs



gestion des dockerfiles



Gestion des utilisateurs

