

WEBQUICK DETECT



**Manuel
d'installation**

Mathis Perrudin, Guillaume Marec et Alxendre Bravo

Table des matières

I.	Informations	3
II.	Image Docker de l'application web	3
III.	Image Docker de la partie IA	3
IV.	Démarrer les 2 images	4
V.	Déploiement	5
VI.	Administration :	5

I. Informations

Notre application est composée de deux images Docker distinctes que vous devrez télécharger sur votre système d'hébergement. Ce manuel vous expliquera comment récupérer et démarrer ces images. Commencez par télécharger les images nécessaires depuis notre dépôt Docker. Ensuite, suivez les instructions pour démarrer les conteneurs Docker, permettant ainsi à l'application de fonctionner correctement sur votre infrastructure.

II. Image Docker de l'application web

Pour installer l'image Docker de l'application Laravel, lancez votre terminal et entrez la commande suivante :

```
docker pull alexsassi/web_quick_detect:app-laravel
```

III. Image Docker de la partie IA

Pour installer l'image Docker qui s'occupe de la partie IA, ouvrez un terminal et exécutez la commande suivante :

```
docker pull alexsassi/web_quick_detect:py_server
```

Attendez que l'installation soit complète. Cette image Docker configure automatiquement deux serveurs distincts : un serveur Flask opérant sur le port 5000 et un serveur websocket sécurisé (WSS) sur le port 5443. Pour utiliser le serveur WSS, il est essentiel de disposer d'un certificat SSL valide. Sans ce certificat, la fonctionnalité d'analyse de webcam de l'application ne sera pas opérationnelle.

IV. Démarrer les 2 images

Une fois les 2 installations terminées, vous pouvez lancer les différentes images docker avec les commandes suivantes :

```
docker network create webquickdetect
```

Cette commande permet de créer un réseau Docker

```
docker run -v pathtoyourkey:/app/ssl/fullchain.pem:ro -v  
pathtoyourkey:/app/ssl/privkey.pem:ro -d --name py_server --network webquickdetect -  
p 5000:5000 -p 5443:5443 alexsassi/web_quick_detect:py_server
```

Cette commande initialise deux serveurs essentiels pour les inférences : un serveur Flask et un serveur WSS, ce dernier étant crucial pour les inférences via la caméra. Assurez-vous de mettre à jour le chemin vers votre clé et votre certificat SSL afin de permettre les inférences sur la webcam. Si vous ne souhaitez pas activer cette fonctionnalité, vous pouvez utiliser la commande alternative suivante pour démarrer les services sans le support de la webcam :

```
docker run -d --name py_server --network webquickdetect -p 5000:5000 -p 5443:5443  
py_server
```

La commande suivante permet de démarrer l'application Laravel sur votre port local 8080 et le serveur de Mail sur le port 8025.

```
docker run -p 8080:80 -p 8025:8025 --network webquickdetect  
alexsassi/web_quick_detect:app-laravel
```

Après l'exécution de ces commandes, vous pourrez accéder à l'application web à l'adresse suivante :

<http://localhost:8080>

Ainsi que la messagerie mail :

<http://localhost:8025>

V. Déploiement

Si vous envisagez de rendre votre site accessible sur Internet, nous recommandons l'utilisation d'un serveur web robuste tel que Nginx ou Apache pour le déploiement. Ces serveurs offrent non seulement une excellente performance et sécurité, mais aussi des options de configuration flexibles qui peuvent être adaptées à vos besoins spécifiques.

VI. Administration :

Un compte administrateur initial est déjà configuré dans la base de données pour faciliter votre première connexion. Vous pouvez accéder à ce compte en utilisant l'adresse email "admin@gmail.com" avec le mot de passe "Admin!1234". Nous vous recommandons fortement de changer ce mot de passe dès votre première connexion pour renforcer la sécurité de votre système.