

SMB111

Projet

DocHub

MERCI DE NE PAS IMPRIMER CE  
DOCUMENT

<b>1. Présentation du projet</b>	<b>4</b>
1. Objectifs	4
2. Prérequis	4
3. Infrastructure (3 Machines)	5
<b>2. Étapes du projet</b>	<b>6</b>
1. Phase 1 : Préparation des Machines	6
2. Phase 2 : Installation et Configuration de Docker	6
3. Phase 3 : Déploiement de Kubernetes	7
4. Phase 4 : Déploiement du Site Web	7
5. Phase 5 : Automatisation avec Ansible	7
6. Phase 6 : Maintenance et Supervision	7
<b>3. Exigences du projet</b>	<b>8</b>
1. Exigences	9
2. Exigences sur les rendus attendus	10

# DocHub: Déploiement Automatisé d'un Site de Documentation avec MCO/MCS

## 1. Présentation du projet

Ce projet consiste à déployer automatiquement un **site web statique** contenant la **documentation complète du projet**, avec Docker, Ansible et Kubernetes sur 3 machines. L'objectif est de fournir un espace accessible via un navigateur pour consulter les spécifications, les guides et les instructions du projet.

### 1. Objectifs

- Un site statique généré automatiquement en HTML
- Utilisation de Markdown pour rédiger la documentation
- Déploiement avec Docker Compose
- Orchestration avec Kubernetes
- Accès web via Traefik/Nginx
- La création de 3 VM
- Intégration du MCO : supervision, logs et monitoring
- Intégration du MCS : mise à jour, durcissement et gestion des accès

### 2. Prérequis

- Trois machines physiques ou virtuelles
- Un accès root ou sudo
- Docker et Kubernetes installés
- Ansible configuré pour la gestion des machines

### 3. Infrastructure (3 Machines)

Machine	Rôle	Services
Master	Orchestration	Kubernetes API, Ansible, Load Balancer, Supervision
Worker Node 1	Serveur Web	Site web statique (Nginx), Monitoring
Worker Node 2	Stockage & Sécurité	Serveur de fichiers, Système de gestion des logs, Audits de sécurité

## 2. Étapes du projet

### 1. Phase 1 : Préparation des Machines

Installation de l'OS (facultatif, si les machines ne sont pas déjà installées)

1. Choix de l'OS (Ubuntu/Debian recommandé)
2. Installation de base avec SSH activé
3. Mise à jour des paquets

Configuration initiale des machines

1. Ajout des utilisateurs nécessaires
2. Configuration réseau et hostname
3. Activation de l'accès SSH sans mot de passe via des clés publiques

Installation des prérequis

1. Installation de Python et des dépendances nécessaires
2. Vérification de la connectivité entre les machines

### 2. Phase 2 : Installation et Configuration de Docker

1. Installation de Docker sur chaque machine
2. Ajout des utilisateurs à Docker pour éviter l'utilisation de sudo
3. Vérification du bon fonctionnement de Docker

### 3. Phase 3 : Déploiement de Kubernetes

1. Installation de Kubernetes (kubeadm, kubelet, kubectl)
2. Initialisation du cluster Kubernetes
3. Ajout des nœuds au cluster
4. Vérification du bon fonctionnement du cluster

### 4. Phase 4 : Déploiement du Site Web

1. Écriture d'un Dockerfile pour contenir l'application web statique
2. Création d'un docker-compose.yml pour l'orchestration locale
3. Déploiement initial avec Docker Compose pour validation
4. Création des manifests Kubernetes pour le déploiement sur le cluster
5. Déploiement final sur Kubernetes

### 5. Phase 5 : Automatisation avec Ansible

1. Création des playbooks Ansible pour automatiser l'installation et la configuration
2. Exécution des playbooks pour provisionner l'ensemble des machines
3. Validation du bon fonctionnement

### 6. Phase 6 : Maintenance et Supervision

1. Mise en place de la supervision des conteneurs et du cluster Kubernetes
2. Déploiement de métriques et logs pour surveiller l'infrastructure
3. Automatisation de la gestion des mises à jour avec Ansible

## 3. Exigences du projet

Niveaux de priorité des exigences

### Priorité 1 - Critique :

- Exigences indispensables à la validation du travail (ex : respect des consignes spécifiques du sujet, plagiat).

### Priorité 2 - Élevée :

- Exigences importantes qui impactent significativement la qualité du rendu (ex : structure du document, clarté et cohérence de l'argumentation).

### Priorité 3 - Moyenne :

- Exigences qui améliorent la présentation et la qualité du rendu (ex : utilisation de données chiffrées, qualité des références).

### Priorité 4 - Faible :

- Exigences supplémentaires qui ne sont pas essentielles mais recommandées (ex : style de présentation, esthétique du document).



# 1. Exigences

Exigences	Description	Priorité
EXG GEN 01	Nombre de machines : Minimum 3 mais vous pouvez en avoir plus	1
EXG GEN 02	OS recommandé : Ubuntu/Debian	4
EXG GEN 03	Technologies utilisées : Docker, Kubernetes, Ansible	2
EXG GEN 04	Automatisation avec Ansible pour installation et gestion	2
EXG GEN 05	Orchestration avec Kubernetes	2
EXG GEN 06	Déploiement d'un site web statique	1
EXG GEN 07	Déploiement initial avec Docker Compose	2
EXG GEN 08	Supervision avec métriques et logs	2
EXG FCT 01	Le site web doit être accessible via une URL spécifique	2
EXG FCT 02	Mises à jour (MCO) via Ansible	3
EXG FCT 03	Utilisation d'un Load Balancer pour la répartition du trafic	3

Exigences	Description	Priorité
EXG TEC 01	Les conteneurs doivent redémarrer automatiquement en cas de crash	2
EXG TEC 02	Sécurisation des accès avec des certificats (Auto Signé)	3

## 2. Exigences sur les rendus attendus

Exigences	Description	Priorité
EXG REN 01	Format du document : <ul style="list-style-type: none"><li>- Format PDF.</li><li>- Taille de police : 12 pt pour le texte, 14 pt pour les titres.</li></ul>	2
EXG REN 02	Structure du document : <ul style="list-style-type: none"><li>- Page de couverture : titre, <b>nom de l'étudiant</b>, date de soumission.</li></ul>	1
EXG REN 03	Contenu : <ul style="list-style-type: none"><li>- Inclusion de données chiffrées et graphiques si nécessaire, avec sources citées.</li><li>- Liste de références complète à la fin du document.</li><li>- Vérification de l'orthographe et de la grammaire.</li></ul>	3

