



# Spécifications projet Architecture|Docker



*Ci-joint la liste de spécifications de votre projet de deuxième années du module Docker. Le projet sera évalué sur la base de 4 critères. Les deux premiers critères sont nécessaires pour avoir la moyenne.*

## Composants techniques de bases :

- ☐ Votre projet contient un frontend avec a minima deux routes (type page d'accueil et route générique).
- ☐ Votre frontend et votre backend vivent dans 2 conteneurs séparés.
- ☐ Votre projet contient un backend avec une base de données.
- ☐ Votre base de données est fonctionnelle et composée d'au moins une table ainsi qu'un schéma explicite.
- ☒ Votre projet contient au moins un Dockerfile.
- ☒ Votre projet contient au moins un fichier docker-compose.yml
- ☒ Vous avez push au moins une image docker sur un container registry (type docker-hub, GCR ...)
- ☒ Votre projet est accessible en ligne sur un gestionnaire de version type GitHub.
- ☐ Votre projet a été builder sans erreur (joindre une capture d'écran dans votre README.md)
- ☐ Vous avez au moins un fichier de test qui :
  - ☐ test la santé de vos conteneurs
  - ☐ test le bon fonctionnement de votre application (front / back)
  - ☐ test du bon fonctionnement de votre base de données

## Architecture

- ☐ Vous avez schématisé l'architecture de votre projet () avec notamment :
  - ☐ les différents composants/services de votre application
  - ☐ les liens entre ces différents composants/services
  - ☐ les ports exposés sont mis en valeur (côté client et coté backend)

## Déploiement & production

- ☐ Votre projet dispose d'un service de reverse proxy type nginx ou traefik.
- ☐ Votre projet dispose d'une documentation auto générée type OpenAPI (formerly Swagger Specification).
- ☐ Vous avez utilisé un orchestrateur type docker-swarm ou kubernetes pour orchestrer vos conteneurs.
- ☒ Votre projet est déployé chez un fournisseur Cloud type (Heroku, GCP, Azure, AWS...) et dispose d'un URL d'accès public.
- ☐ Vous avez automatisé au moins un test dans votre fichier docker-compose.yml

## Clean Code

- ☐ Votre projet dispose d'une documentation claire et concise qui explique les différents endpoints de votre application ainsi que les ports exposés.
- ☐ Il y a des commentaires dans tous vos fichiers de code.

- ☐ Votre README.md à la racine de votre GitHub contient les explications nécessaires pour build et run votre projet.