
1 INFORMATIONS GENERALES

Lieu de travail :	ETML / VENNES / 1010 LAUSANNE
Dates de réalisation :	Du mercredi 19.03.2025 au vendredi 30.05.2025
Horaire de travail :	5 périodes par semaine, 9 semaines (+2 de vacances)
Temps total :	45 périodes

2 PROCÉDURE

- Tous les apprentis réalisent le projet sur la base d'un cahier des charges.
- Le cahier des charges est présenté, commenté et discuté en classe.
- Les apprentis sont entièrement responsables de la sécurité et sauvegarde de leurs données.
- En cas de problèmes graves, les apprentis avertissent le client au plus vite.
- Les apprentis ont la possibilité d'obtenir de l'aide externe, mais ils doivent le mentionner.

3 TITRE

- **Dockeriser et déployer une application web**
- **Continuer le développement de l'application « FlashCards »**

4 SUJET

Découvrir et comprendre l'utilisation de **Docker** en « Dockerisant » et en **déployant** une application **AdonisJS**.

Continuer le développement de l'application « Flashcards ».

5 MATÉRIEL ET LOGICIEL À DISPOSITION

- Un PC ETML
- Accès à Internet
- Visual Studio Code
- Node.js
- Git et github
- Docker

6 PRÉREQUIS

Modules de programmation de base

7 CAHIER DES CHARGES

7.1 Qualité

1. Réaliser un travail de qualité
 - Organisé
 - Compacté (pas de copié/collé,...)
 - Optimisé (utilisation de structures adaptées)
 - Testé
 - Commenté
2. Prouver que vous êtes digne de confiance lorsqu'on vous confie un projet
 - Pro-activité
 - **Poser des questions** au client
 - Faire des démonstrations
 - Utiliser un système de versioning de code (GIT)

7.2 Description du projet

Constat : En tant qu'élève, les concepts de « **serveur de production** », « **serveur de test** » et de « **déploiement** » restent flous car ils sont rarement abordés dans votre formation.

L'idée de ce projet est de tenter de démystifier ces concepts.

Ce projet consiste à :

- **Dockeriser** l'application **adonisJS** que vous avez réalisée précédemment. Pour cela, vous devez découvrir les concepts de base de **Docker**.
- **Déployer votre application** sur 2 environnements
 - Serveur de production sur une plateforme gratuite de votre choix qui permet de déployer votre application **AdonisJS dockerisée**.
 - Serveur de test (« **staging** ») sur une plateforme interne à l'ETML qui permet de déployer votre application **AdonisJS sans Docker**.

Dans les 2 cas, votre application a besoin d'une base de données **MySQL**. Par contre, pour le déploiement sur une plateforme gratuite, vous avez la possibilité d'utiliser un autre SGBDR comme par exemple **PostgreSQL** (ce qui élargi vos possibilités de plateformes gratuites).

Utilisation des IA :

Comme vous n'avez pas encore de réelle connaissance de **Docker**, l'utilisation des IA telle que **ChatGPT** est autorisée. Mais attention, vous devez être capable d'expliquer en détail chaque ligne de vos fichiers **Docker** et toute configuration permettant le déploiement de votre application.

7.3 Dockerisation

Que signifie « Dockeriser mon application AdonisJS » :

Dockeriser une application AdonisJS signifie encapsuler cette application et ses dépendances dans un **conteneur Docker**, afin de la rendre facilement **déployable** et **portable** sur n'importe quelle machine ou serveur prenant en charge Docker.

À travers ce projet, vous devez donc acquérir les compétences suivantes :

- Comprendre et rédiger un **Dockerfile** qui décrit comment construire une image **Docker** pour une application **AdonisJS**.
- Créer et utiliser un **docker-compose.yml** pour orchestrer plusieurs conteneurs (application, base de données, etc.)
- Expliquer chaque ligne des fichiers **Dockerfile** et **docker-compose.yml** dans un fichier markdown lié dans le readme.

Dans cette partie, vous devez expliquer en détail tout ce que vous avez compris du fonctionnement de docker.

7.4 Déployer votre application

7.4.1 Déployer votre application sur un serveur de l'ETML

L'ETML dispose depuis peu d'un hébergement permettant de déployer des applications node.js utilisant une base de données **MySQL**.

Vous devez réussir à déployer votre application sur cet environnement et expliquer en détail dans le Readme toutes les étapes effectuées.

Votre enseignant vous fournira les informations nécessaires pour le déploiement sur cet hébergement.

7.4.2 Déployer votre application dockerisée sur une plateforme de cloud computing

Une plateforme de **cloud computing** est un service en ligne qui permet aux utilisateurs de déployer, héberger et gérer des applications, des bases de données et d'autres ressources informatiques sans avoir à gérer directement le matériel sous-jacent (serveurs physiques, stockage, réseau, etc.).

Voici une liste non exhaustive d'étapes à réaliser :

- Vous devez trouver une plateforme de **cloud computing** qui vous permette de déployer gratuitement votre application **AdonisJS** dockerisée. N'oubliez pas que votre application utilise une base de données **MySQL** et que vous pouvez choisir un autre SGDBR comme par exemple **PostgreSQL**.
- Déployer votre application **Dockerisée** sur la plateforme trouvée à l'étape précédente.
- Dans ce projet, nous allons considérer cette plateforme comme notre environnement de production. Vous devez expliquer dans votre Readme les principales différences de configuration entre un environnement de développement et un environnement de production dans le contexte d'une application d'**AdonisJS**.

7.5 Continuer le développement de l'application « FlashCards »

Vous devez continuer le développement de l'application « **FlashCards** » en suivant les mêmes « règles définies » dans le précédent projet.

Votre mission est d'atteindre le **niveau 8**.

7.6 Contraintes

1. Tous les résultats de votre travail sont enregistrés dans votre repository Flashcards (le même que pour P_BULLE T3)
2. Il n'y a pas de Readme à rendre, tout doit se trouver dans (ou être accessible à partir du) Readme.md de votre repository.

3. Le Readme contient (et/ou référence):

- La description du contexte technique de développement
- Les opérations de mise en place de cet environnement
- La description du contexte technique de validation (staging)
- Les opérations de mise en place de cet environnement
- Les opérations de mise à jour de cet environnement (lorsque le développement a produit un artefact)
- La description du contexte technique de production
- Les opérations de mise en place de cet environnement
- Les opérations de mise à jour de cet environnement (lorsqu'un artefact a été validé)

8 Livrables

A la fin de chaque semaine, au plus tard le vendredi, vous créez une release nommée « Semaine X » dans votre repository. Votre journal de travail en PDF est joint à cette release

9 Évaluation

1. Votre enseignant peut effectuer une ou plusieurs évaluations imprévues à l'oral tout au long du projet.
2. Vous devez être en capacité d'expliquer à votre enseignant chaque ligne de code que vous avez écrites et chaque ligne de votre Readme.

Voici la grille d'évaluation de projet contextualisée.

Attention : il est possible que cette grille fasse l'objet d'amendements en cours de projet.

			LARGEMENT ACQUIS	ACQUIS	INSUFFISANT	NON ACQUIS
COMPÉTENCES	PROFESSIONNELLES	Rythme de travail Rapidité, Efficacité	1. Votre app est disponible en production 2. Votre app est disponible en staging 3. Vous êtes niveau 8 dans WOF	Deux des trois points ci-contre	Moins	Pire
		Conscience professionnelle Qualité du travail	Readme de très bonne qualité dans son contenu. Il est visible que le contenu provient d'une réflexion de votre part. Vous êtes capable d'expliquer à l'oral chaque mot, chaque phrase qu'il contient	Readme de qualité satisfaisante dans son contenu. Il y a très peu de mots ou phrases que vous ne pouvez pas expliquer de manière convaincante	Readme issu trop directement d'une IA ou Il y a plusieurs mots ou phrases de votre Readme que vous ne parvenez pas à expliquer oralement	Pire
		Connaissances professionnelles Techniques enseignées	Vous êtes encore en possession des trois points JSP de départ	Il vous reste au moins un point JSP	Il ne vous reste plus de points JSP	Echec dans les notions de base

	METHODOLOGIQUES	Processus de travail	Journal de travail complet et détaillé Commits Git tous conformes à la convention Toutes les livraisons effectuées selon les directives	Journal de travail acceptable Quelques erreurs de commits Livraisons conformes à un ou deux détails près	Journal insuffisant Trop d'erreurs de commits Livrable non recevable	Pas concerné
		Expression orale et écrite Technique de présentation	Readme de très bonne qualité dans sa forme. Il permet entre autres à un collègue de déployer votre application. Le Readme traite également de cas d'erreurs pouvant se présenter.	Bon Readme, permettant au moins de se débrouiller pour arriver à déployer votre application	Qualité de Readme insuffisant ou Procédure et/ou informations insuffisantes pour permettre de suivre le déploiement	Pire
		Approche écologique et économique				
	SOCIALES	Aptitude au travail en équipe Gestion des conflits Communication	Influence positivement le groupe Réagit de manière réfléchie et cherche des solutions	Maintien les bonnes relations Ne provoque pas de conflit et participe aux solutions	Ne participe pas à la cohésion du groupe Réagit de manière irréfléchie et/ou disproportionnée	Influence négative marquée

	PERSONNELLES	Autonomie Attitude face au travail Faculté d'apprendre	Indépendant Entreprenant Adaptation facile aux changements	Aide justifiée Fiabilité Adaptation aux changements	Souvent besoin d'aide Erreurs minimisées Perturbé par les changements	Dépendant Erreurs cachées Inapte aux changements
--	--------------	---	---	--	--	---

Le 19.03.2025

Antoine Fabre

Signature: 