

Examen de Technologies Logicielles

Sébastien Morais

30 Avril 2025

Consignes générales

- La durée de l'examen est de 3 heures
- Le rendu doit se faire sur un dépôt **privé** Github, en ajoutant l'enseignant au projet avec un niveau suffisant pour accéder aux sources
- L'utilisateur Github associé à l'enseignant est **SMoraisDev** et a pour adresse **sebastien.morais@proton.me**
- Un fichier README clair et précis sur l'utilisation de votre application doit être fourni
- Vos images doivent être publiées sur Docker Hub. Pour vérifier la réalisation de cette tâche, l'enseignant doit pouvoir accéder à vos images. Veuillez préciser comment le faire dans votre README.
- Votre application doit être fonctionnelle et pouvoir être lancée avec la commande : `docker-compose up`

Barème (approximatif)

- Utilisation de Git et Github : 5 points
- Utilisation de Docker et Docker Hub : 4 points
- Code fonctionnel, avec des **tests unitaires**, des commentaires, des annotations de type valides, une bonne couverture de code, et un code validé par outil de formatage : 11 points

Description du contexte

Afin de valider votre période d'essai, votre directeur technique vous a demandé de réaliser une preuve de concept sur les **technologies** vues pendant votre cours de Technologies Logicielles et de démontrer votre capacité à **organiser votre travail**. Le thème de cette preuve de concept est la gestion de magasins de fleurs.

Implémentation :

- La classe (ou modèle) **Fleur** encapsule une espèce, un identifiant unique, une date de coupe, une qualité et un prix.

- La classe **Facture** encapsule le nom d'un client, un identifiant unique, une date de vente, une liste de fleur composant un bouquet et un prix de vente.
- La classe **CarteFidelite** encapsule le nom d'un client et la liste de ses factures. Cette classe dispose de méthodes pour ajouter une facture, remettre à zéro l'historique des factures, calculer le niveau de fidélité, retourner la liste des factures entre deux dates données.

Architecture souhaitée :

- Deux **bases de données** pour enregistrer les fleurs disponibles. La première correspond au magasin FloraDestock et est composée de fleurs de qualité standard ou dégradée. La seconde correspond au magasin FloraPrestige et est composée de fleurs de qualité supérieure.
- Une API Rest, exposée sur **le port 89236**, avec de **nombreux points de terminaison** permettant d'interagir avec les deux magasins (en python + FastAPI). Par exemple : ajouter/modifier/supprimer une fleur, rechercher des fleurs entre un prix minimum et maximum, rechercher des fleurs par date de coupe, ...
- Un script remplissant automatiquement les bases de données avec des données **valides** prédéfinies par vos choix (en python)

Spécificités du directeur :

- Les **tests unitaires** doivent être réalisés sur la classe **CarteFidelite**
- Une facture associée à un bouquet ne peut pas avoir une date de vente antérieure à la date de coupe des fleurs
- Le prix d'une facture est égal à la somme des prix des fleurs utilisées à laquelle on ajoute une TVA de 20%
- Le prix d'une fleur doit être un montant se terminant par .99 (par exemple 1.99, 4.99, ...)
- Les factures associées à une carte de fidélité doivent appartenir au client propriétaire de la carte
- Le niveau de fidélité est Bronze (respectivement Argent et Or) si la somme du prix des factures est inférieur à 200 (respectivement 500 et 2000)