



Construction d'un Système de Traitement de Commandes Asynchrone avec FastAPI

Introduction :

Dans ce TP, vous allez développer un système de traitement de commandes asynchrone en utilisant FastAPI. Le système se compose de deux composants principaux : un processus client pour passer des commandes et un processus fournisseur pour gérer les demandes de commande, générer des devis et confirmer les commandes.

Objectifs :

1. Implémenter des points d'accès asynchrones pour passer des commandes, vérifier des commandes, générer des devis et confirmer des réalisations.
2. Utiliser des tâches en arrière-plan et une logique asynchrone pour gérer la communication asynchrone entre les processus client et fournisseur.
3. Implémenter des tâches d'intervention humaine pour vérifier des commandes, confirmer des devis et confirmer des réalisations.
4. Gérer l'état et la session des instances de processus et stocker les données de processus dans une base de données.
5. Créer des tableaux de bord avancés pour surveiller les états des processus et analyser l'exécution des processus.

Tâches :

Tâche 1 : Configuration du Projet FastAPI

1. Créer un nouveau projet FastAPI.
2. Configurer une base de données pour stocker les données de processus.

Tâche 2 : Modélisation BPMN

1. Établir un modèle BPMN décrivant le flux du processus de traitement des commandes. Identifiez les points d'interaction entre le client et le fournisseur, ainsi que les tâches asynchrones.
2. Utiliser le modèle BPMN comme référence pour l'implémentation.

Tâche 3 : Implémentation des Processus Client et Fournisseur

1. Créer des points d'accès asynchrones pour passer des commandes et gérer les événements de confirmation dans le processus client.
2. Implémenter des points d'accès asynchrones pour vérifier des commandes, générer des devis et gérer les événements de réalisation dans le processus fournisseur.

Tâche 4 : Implémentation des Tâches d'Intervention Humaine

1. Créer des tâches d'intervention humaine pour vérifier des commandes, confirmer des devis et confirmer des réalisations.
2. Utiliser des tâches en arrière-plan ou des tâches asynchrones pour gérer l'intervention humaine.

Tâche 5 : Gestion de l'État et de la Session

1. Implémenter des mécanismes pour gérer l'état et la session des instances de processus.
2. Stocker les données de processus dans la base de données pour permettre la consultation de l'état de n'importe quel processus à tout moment.

Tâche 6 : Implémentation des Tableaux de Bord

1. Créer des tableaux de bord avancés pour surveiller les états des processus.
2. Implémenter des tableaux de bord d'analyse statistique pour analyser l'exécution des processus et l'achèvement des tâches.

Ressources :

- Documentation FastAPI : <https://fastapi.tiangolo.com/>
- Guide de configuration de la base de données : <https://docs.sqlalchemy.org/en/20/>
- Tableaux de bord avec FastAPI : <https://fastapi.tiangolo.com/tutorial/security/oauth2-jwt/> ; ou Grafana : <https://grafana.com>

Soumission :

Soumettez le code source, y compris les fichiers du projet FastAPI, les scripts de base de données et tout fichier supplémentaire lié aux tableaux de bord. Fournissez une documentation expliquant les choix de conception, les détails de l'implémentation et comment exécuter et tester le système de traitement des commandes.

Évaluation :

Votre projet sera évalué en fonction des critères suivants :

1. **Fonctionnalité** : Le système répond-il aux exigences spécifiées ? Les processus asynchrones sont-ils implémentés correctement ?
2. **Modélisation BPMN** : Le modèle BPMN représente-t-il correctement le flux du processus et les interactions entre les tâches ?
3. **Qualité du Code** : Le code est-il bien organisé, lisible et correctement documenté ?
4. **Gestion de la Base de Données** : La base de données est-elle configurée correctement, et stocke-t-elle efficacement les données de processus ?
5. **Tableaux de Bord** : Les tableaux de bord créés permettent-ils de surveiller les états des processus et d'analyser l'exécution des processus ?
6. **Gestion de l'Intervention Humaine** : Les tâches d'intervention humaine sont-elles implémentées efficacement à l'aide de tâches en arrière-plan ou asynchrones ?

Note :

Il est recommandé d'explorer des fonctionnalités supplémentaires et des améliorations au-delà des tâches spécifiées. La créativité et les fonctionnalités supplémentaires seront prises en compte pour des points supplémentaires. Bonne chance !