

Evaluation pratique finale

Individuel - Rendu le 05/06/2020 à 18h - Documents autorisés

L'objectif de cette évaluation est de mettre en pratique les connaissances acquises et les travaux effectués dans le cadre de ce cours de WebServices. Cette évaluation fera appel à toutes les notions abordées dans les différents TPs effectués. Dans cette évaluation, il vous sera demandé de mettre en place une architecture de A à Z dans un cas concret de création d'application. Vous réaliserez aussi une partie client légère dans la technologie de votre choix.

Vous êtes le nouveau responsable produit d'une entreprise qui souhaite se lancer dans la création d'une application de location de vidéos à la demande, plus particulièrement des films. Il vous revient la charge de concevoir et développer un prototype de cette application en vous servant des vos connaissances extrêmement pointues des bonnes pratiques concernant la création de WebServices.

La suite de cette évaluation vous guidera dans la mise en oeuvre de cette application.

Concernant le rendu, vous devrez développer votre code en utilisant l'outil de versionning GIT. Vous rendrez votre code sur GitHub Classroom avec un fichier README qui récapitule les différents choix qui vous sont demandés ainsi que la liste des endpoints de vos WebServices. Tout manquement à cette consigne de rendu entraînera une perte de points.

Cahier des charges

Ce cahier des charges a pour but de préciser les besoins et les solutions choisies concernant le développement de l'application de location de vidéos à la demande. L'application a pour objectif de permettre à des utilisateurs enregistrés de louer des films ou tout autre contenu vidéo sur cette plateforme.

Architecture

La solution développée doit être correctement segmentée afin de limiter la responsabilité de chaque composant de cette plateforme. En d'autres mots, l'architecture Micro Service sera utilisée pour mettre en place cette plateforme en permettant à chaque service de disposer de sa propre base de données. Chaque service du back-end sera réalisé via Node.JS

Sécurité

Tous les services nécessiteront une authentification gérée par un service tier. Cette authentification doit se faire via un token JWT.

Fonctionnalités

Authentification / Inscription

Les utilisateurs doivent pouvoir s'inscrire sur la plateforme via leur adresse mail et un mot de passe. Ils doivent être obligatoirement connectés pour accéder à la plateforme..

Catalogue de films

Les utilisateurs doivent avoir à disposition un catalogue de films classés par nom.

Chaque film doit avoir:

- Un nom
- Une image de couverture
- Une description
- Un prix à la location

Détails d'un film

L'utilisateur peut disposer des détails d'un film avant de le louer. Il peut retrouver toutes les informations précédentes sur la page d'un film mais aussi ajouter un commentaire au film une fois qu'il l'a loué.

Location d'un film

Un utilisateur peut louer un film autant de fois qu'il le souhaite. Chaque location doit durer 24h. Une fois que l'utilisateur a loué un film, il peut le télécharger pendant toute la durée de sa location (Ici, un simple bouton factice suffira).

Le paiement s'effectue à l'aide de sa carte bancaire, pour le moment nous n'implémenteront pas le mécanisme entier de paiement mais simplement une vérification de la longueur du numéro de carte, de la date d'expiration et du cryptogramme visuel.

Sujet

Partie 1 - Architecture

Le premier objectif de ce sujet est de séparer les différents services qui composeront votre architecture.

Réalisez un schéma que vous joindrez dans le README explicitant la liste des services composant votre architecture microservices. Vous identifierez dans cette liste les services les plus critiques, nécessaire au fonctionnement minimal de l'application.

Concernant le service gérant le catalogue, vous distinguerez celui-ci du service servant les informations d'un film.

Partie 2 - Spécification de vos services

Chacun de vos service doit disposer d'une documentation sous forme OpenAPI 3 au format YAML. Vous réaliserez ces spécifications avant de développer le service.

Pour plus de simplicité vous regrouperez ces spécification dans un seul YAML disponible à la racine du projet.

Partie 3 - Développement des services

Vous développerez les différents WebServices conformément au cahier des charges qui a été explicité plus haut. Chaque service doit être isolé comme le veut la pratique des microservices: chaque service doit avoir son propre dossier et son propre cycle de vie.

Les services doivent être capables de discuter entre eux. Ceux que vous avez identifiés comme critiques doivent être capables de fonctionner sans les autres services.

Vous devez implémenter Docker sur votre architecture et mettre à disposition un fichier docker-compose.yml pour lancer votre infrastructure.

Partie 4 - Testing

Vous réaliserez les différents tests unitaires et d'API de vos services. Pensez à ajouter un service de code coverage.

Partie 5 - Frontend

Pour finaliser ce prototype, vous développerez un interface graphique simple, dans la technologie que vous souhaitez. Cette interface doit présenter:

- Une page d'inscription / connexion
- Une page de catalogue listant les différents films disponibles et les films actuellement loués
- Une page de détails d'un film avec la possibilité de le louer