

Projet de session

Historique

#	Date	Description
1	25 septembre	Version initiale
2	25 septembre	Précisions sur les remises
3	25 septembre	Correction au diagramme de séquence
4	26 septembre	Instructions de remise sur Github
5	7 octobre	Ajout d'une exigence lors du paiement d'une commande et ajout de précisions sur la récupération des produits
6	11 octobre	Rajout d'une exigence pour la création d'une commande et quelques précisions
7	18 novembre	Deuxième remise

Informations générale

L'objectif du projet de session est de développer et déployer une application Web responsable du paiement de commandes Internet.

Le projet est divisé en deux étapes pour chacune des remises.

Objectifs

- Se familiariser avec le développement Web
- Développer une API REST
- Utiliser des services Web distants
- S'assurer de la résilience et de la performance d'une application Web
- Déployer une application Web

Équipe

Le travail se fait individuellement ou en équipe de 2.

Échéances

Date	Description	Pondération
4 novembre 21h	Première remise	15%
16 décembre 21h	Remise finale	15%

Le projet de session représente 30% de la note globale du cours.

Évaluation

L'évaluation se fera à distance sans la présence du ou des membres de l'équipe.

Le code de l'application doit être hébergé sur Github.

Aux deux dates de la remise, à 21h, le dépôt sera cloné et c'est ce qui sera utilisé pour l'évaluation.

Remise

La création du projet sur Github est automatisé grâce à Github Classroom.

Pour créer un projet, vous devez suivre les instruction suivantes :

1. Créer un compte Github si ce n'est déjà pas le cas.
2. Rendez-vous sur la page du projet de session :
<https://classroom.github.com/g/oH9-4nj2>
3. Acceptez le travail, et identifier vous dans la liste en choisissant votre code permanent **Assurez-vous de ne pas prendre le code de quelqu'un d'autre!**
4. Créer une équipe individuelle, ou rejoignez l'équipe de votre collègue si vous faites la remise à 2.
5. Votre dépôt Github privé sera créé, vous pouvez commencer à l'utiliser.

Langage de programmation

Le langage de programmation pour le projet de session est Python. La version minimale qui sera utilisé est 3.6. Vous pouvez utiliser la version 3.7 si vous le désirez.

Le cadriciel de développement Web est Flask 1.1.

Le démonstrateur de laboratoire sera en mesure de vous apporter du support et du soutien technique pour ces deux technologies.

Le projet

Le projet consiste à développer une application Web responsable de prendre des commandes Internet. Cette application devra répondre à une API REST, mais

devra également être utilisée à travers des pages HTMLs.

Le projet est séparé en deux. Les informations pour la première remise sont archivés et disponible sur Github.. La section pour la deuxième remise est disponible ci-bas.

Il s'agit de remises incrémentiels sur le même projet. Pour la deuxième remise, vous continuerez à utiliser le projet.

Deuxième remise

La deuxième partie du projet de session se concentre sur la maintenance de l'application : déploiement, ajout de fonctionnalités, performance et observabilité.

Base de donnée

Vous devez changer la base de donnée `sqlite` vers PostgreSQL. Les informations de connexion à PostgreSQL sont transmises par les variables d'environnements suivantes :

- `DB_HOST` : l'hôte pour se connecter à la base de donnée
- `DB_USER` : le nom d'utilisateur de la base de donnée
- `DB_PASSWORD` : le mot de passe de la base de donnée
- `DB_PORT` : le port de connexion de la base de donnée
- `DB_NAME` : le nom de la base de donnée

N.B.: Vous n'avez pas à gérer la création de la base de donnée pour cette remise. Seulement la création des tables.

Initialisations des tables

Lors de la correction de cette remise, les tables de la base de donnée seront créées avec la commande suivante :

```
$ FLASK_DEBUG=True FLASK_APP=inf5190 DB_HOST=localhost  
→ DB_USER=user DB_PASSWORD=pass DB_PORT=5432 DB_NAME=inf519  
→ flask init-db
```

Les informations de la base de donnée (hôte, utilisateur, etc) sont données à titre d'exemple seulement.

Dockerfile

Vous devez produire un fichier `Dockerfile` valide à la racine du projet. Ce fichier doit produire une image Docker de votre application avec toutes les dépendances requises (Python, Flask).

Ce fichier doit pouvoir bâtir l'image Docker avec la commande suivante :

```
$ docker build -t inf5190 .
```

Et l'application doit pouvoir être lancée avec la commande suivante :

```
$ docker run -e DB_HOST=localhost -e DB_USER=user -e  
→ DB_PASSWORD=pass -e DB_PORT=5432 -e DB_NAME=inf519 inf5190
```

Déploiement

Votre application doit être déployée sur Heroku. L'URL complet de l'application (i.e.: <https://<application>.herokuapp.com>) doit être fournie dans un fichier **HEROKU** à la racine de votre projet.

L'application déployée doit utiliser PostgreSQL (i.e.: **Heroku Postgres**)

Vous n'êtes pas obligé de payer pour Heroku, les forfaits de base pour l'application, ainsi que pour PostgreSQL et Redis sont suffisant pour cette remise.

Vous devez obligatoirement rajouter mon compte (jean-philippe.caissy@uqam.ca) en tant que collaborateur à votre application Heroku. Sinon je ne pourrai pas corriger l'objectif de déploiement et vous aurez 0% pour cette exigence.

Les instructions pour rajouter un collaborateur sont disponible ici : <https://devcenter.heroku.com/articles/collaborating#add-collaborators>

Première remise

Les informations sur la première remise sont disponibles sur Github : <https://github.com/jpcaissy/INF5190/blob/b55de3f413234b3217d6df2f0cf9f5f52880f2dd/travail-de-session/enonce.md#premi%C3%A8re-remise>.