Pedagogia

Opening Pedagogia platform

Alexander Mazur

(version traduite en français et tronquée)

30.05.2023

L'ARCHITECTURE DE LA PLATEFORME PEDAGOGIA

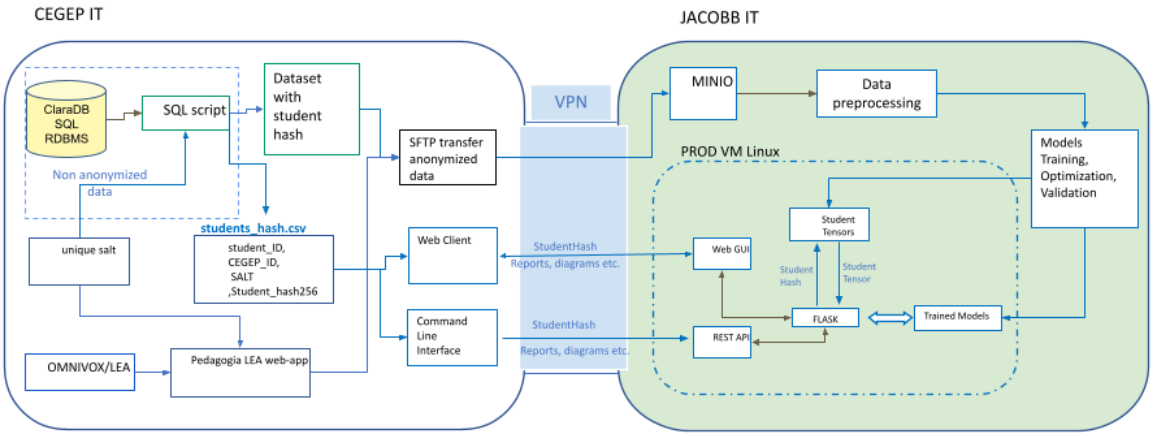

Fig. 1. Schéma de communication entre le client (FEDERAT) et le serveur (JACOBB) à travers la plateforme Pedagogia

La plateforme Pedagogia se compose de deux parties principales : le côté client (CEGEP) et le côté serveur (JACOBB). Le côté client est situé dans l'environnement informatique du CEGEP et est responsable de la collecte/extraction des informations des étudiants stockées dans une source de donnée, de l'anonymisation de ces données, et de leur transmission via un canal sécurisé (VPN) vers le côté serveur. Du côté serveur, ces données seront nettoyées, vérifiées et prétraitées de manière appropriée pour effectuer l'entraînement d'un modèle spécialement conçu et vérifier les résultats prévus (voir figure 1).

La technologie de la plateforme

La plateforme Pedagogia a été mise en œuvre en utilisant Python 3.8.5, avec la version 2.4.1 de la bibliothèque Keras et TensorFlow 2.4.0 comme programme secondaire. Deux applications web pour web GUI et REST API ont été développées pour fournir des connexions entre le côté client destiné aux enseignants (interface web) ou au pipeline d'analyse interne du CEGEP (API REST) avec le côté serveur (JACOBB).

Anonymisation des données côté client

L'anonymisation des données consiste à supprimer de manière permanente et complète les identifiants personnels des données, notamment en transformant les informations personnelles identifiables en données agrégées. Les données anonymisées sont des données qui ne peuvent plus être associées à un individu de quelque manière que ce soit. Une fois que ces données ont été dépourvues d'éléments permettant l'identification personnelle, ces éléments ne

peuvent jamais être ré-associés aux données ou à l'individu sous-jacent.

Nous proposons une méthode pour anonymiser le jeu de données côté client en utilisant notre service de hachage sécurisé. Le service de hachage de JACOBB est une application qui s'exécute dans l'environnement informatique côté client et offre des interfaces REST API et web-GUI. L'entrée de ce service est un fichier non anonymisé et la sortie sera un fichier anonymisé comme suit:

- Nom, prénom, numéro de téléphone: supprimés
- Code permanent: haché par l'algorithme SHA256
- Date de naissance: convertie en année de naissance
- Code postal: converti en code régional (suppression des 3 derniers caractères)

L'algorithme SHA256 permet de crypter une chaîne de caractères de manière irréversible. Cet algorithme sera appliqué à la combinaison du code permanent et d'un mot de passe temporaire et spécifique à un utilisateur côté client. De cette manière, JACOBB ou un utilisateur non autorisé de la plateforme sera incapable d'identifier les étudiants dont les données anonymisées sont transmises à la plateforme et ce, même en possession de la liste des codes permanents du client. L'algorithme sera fourni au client sous forme de script exécutable en mode *bash*.

L'exécution du script de hachage sur les fichiers contenant les données SPEC sera réalisée par un technicien du côté client et les données anonymisées seront exportées vers des fichiers texte (séparés par des tabulations). Les ensembles de données de sortie seront anonymisés et ne contiendront aucune information d'identification personnelle.

Transfert de données côté client et vérifications d'intégrité

La procédure de transfert de données du côté client vers le côté JACOBB peut être organisée soit avec une interaction humaine, soit en mode automatisé. Du côté client, les éléments suivants doivent être préinstallés :

1. Application WireGuard (publiquement disponible) pour établir une connexion VPN sécurisée entre le client et JACOBB.
2. Script Python SFTP client (JACOBB) pour transférer de manière sécurisée des ensembles de données anonymisées du côté client vers Jacobb.

Pour sécuriser la transmission des ensembles de données anonymisées vers le côté de JACOBB, nous fournissons une connexion VPN via WireGuard vers le service de stockage cloud MINIO de JACOBB (compatible avec l'API du service de stockage cloud Amazon S3, voir figure 1). Les ensembles de données clients provenant de MINIO sont vérifiés, nettoyés et

prétraités. L'étape suivante consiste à entraîner, valider et optimiser les hyperparamètres du modèle. JACOBB propose plusieurs interfaces pour la visualisation des données prévues via une interface web-GUI et via une API REST à utiliser dans les pipelines automatisés des clients.

Vérifications de cohérence pour les procédures côté client

Voici une liste d'étapes d'assurance qualité qui garantissent le bon fonctionnement complet du protocole de transfert de données :

- Vérification de l'état d'exécution du service de hachage sécurisé (conteneur Docker)
    - Vérification de la validité de la paire nom d'utilisateur/mot de passe pour l'utilisateur.
    - Test de la procédure de hachage avec des fichiers de test.
- Tester l'application WireGuard pour ouvrir et fermer la connexion VPN sécurisée vers JACOBB
- Tester la connexion du client SFTP avec le service MINIO (JACOBB) via le VPN:
    - Vérification de la validité de la paire nom d'utilisateur/mot de passe pour l'utilisateur SFTP
    - Envoi d'un fichier de test à MINIO et réception du statut de transfert
    - Simulation de la transmission d'un fichier d'erreur et vérification du service d'alerte par e-mail (les e-mails d'alerte doivent être envoyés aux personnes responsables)
    - Le service de journaux doit être mis à jour pour toutes les transmissions de données avec les champs date-heure, statut (OK ou code/message d'erreur), nom de fichier, taille du fichier