Présentation de Flask-MonitoringDashboard :

Un dispositif de monitorage applicatif spécifique aux application Flask est Flask-MonitoringDashboard. Il permet la collecte des données (la quantité de données collectées par le tableau de bord varie pour chaque endpoint), et le niveau de surveillance peut etre spécifié pour chaque endpoint. Le fichier de configuration (config.cfg) permet d’ajuster certaines proprietes telles que la version d’application, lien du repo git, niveaux de monitorage, période d’echantillonage, ect. Une base de données est créée automatiquement pour sauvegarder les logs d’évènements et un outil de visualisation est également mis à disposition.

Présentation des outils de monitorage :

Il existe 3 niveaux d'alerte qui permet d’enregistrer les évènements de performance et utilisation, les outliers qui correspondent aux requêtes les plus lentes, et un profilage statistique de toutes les requêtes sur l’API. La détection automatique d’évènement permet par exemple de connaitre la date et l’heure ainsi que l’IP qui fait la requête.

Un incident pourrait ainsi être le mauvais chargement du modèle ML qui permet d’effectuer les prédictions. Ce genre d’évènement peut être enregistré grâce au module logging de python et un block try /except.

Debugging l’application:

Différentes erreurs sont survenues avec l’exécution du script de l’application. Il peut s’agir de IndentationError, SyntaxError, erreurs de nom de colonnes et de noms de endpoints

(jinja2.exceptions.TemplateNotFound: home.html), des TypeError: de type ‘positional argument’ ou il est nécessaire d’ajouter les arguments dans l’appel de la fonction. Une erreur de type ValueError sur l’utilisation du modele pour la prediction à également été debuggée ainsi que l’insta,ntiation de l’array pour récupérer les prédictions.

lien vers le repo de l’app débeuggée :

https://github.com/RCaz33/Simplon\_RenduE5