

Projet OPA CryptoBot avec Binance

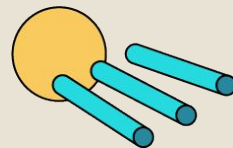
Le but principal de ce projet est de créer un bot de trading basé sur un modèle de Machine Learning et qui investira sur des marchés crypto

Oualid Lemtine

Samuel Vicat



DataScientest



Nov23_Continu_de - 2024

SOMMAIR

E

1

Présentation du projet

2

Étapes importantes du projet

3

La création d'équipes

4

Étapes importantes du projet

5

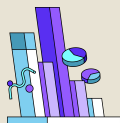
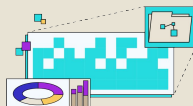
L'encadrement du projet



Présentation du projet

Se confronter à un réel
problème de Data
Engineering

Démontrer toute la
panoplie de
compétences



Traduire une problématique **métier** en un problème de **Data Engineer**.

Identifier et choisir les données nécessaires à sa **résolution**.

Modéliser le problème (données, choix des algorithmes).

Apporter une **expertise** métier et un regard critique sur des résultats.

Restituer, présenter et illustrer ses résultats à tout type **d'audience**.



La proposition de sujets

1/ Choisir un sujet dans le catalogue

En rapport avec une activité que vous connaissez.

Basé sur des données disponibles.

Faire valider par le Chef de Cohorte J-10 avant la fin du sprint 2.

2/ Proposer un sujet



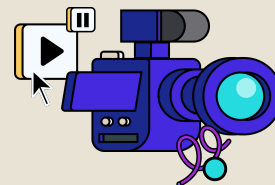
La création d'équipes

Les groupes sont composés de 3 à 4 personnes.

Comment choisir son projet ?

Au cours du sprint 2, vous émettez vos vœux en remplissant le formulaire qui vous sera communiqué par votre CDC via Slack.

L'allocation des groupes se fait par l'équipe pédagogique.



Étapes 1 : Récolte des données



Etape	Description	Objectif	Modules / Masterclass / Templates	Conditions de validation du projet
1	Récolte des données	<p>Récolter deux types de données en passant par l'API Binance en passant par une architecture de streaming.</p> <ol style="list-style-type: none">Grâce à cette API, on peut aller récupérer des informations sur les cours des différents marchés (BTC-USDT, BTC-ETH, ...). Le but sera de créer une fonction de récupération de données générique afin de pouvoir avoir les données de n'importe quel marché. Il faudra aussi créer un script de pré-processing pour réorganiser les données sortant du streaming afin qu'elles soient propres.Récupérer les données historiques, pré-processées pour pouvoir entraîner notre futur modèle		<p>Fichier explicatif du traitement (doc / pdf)</p> <p>Un fichier json d'exemple de récupération.</p>



Étapes 2 : Architecture de la donnée



Etape	Description	Objectif	Modules / Masterclass / Templates	Conditions de validation du projet
2	Architecture de la donnée	<p>Il s'agit de choisir la solution de stockage la plus adaptée:</p> <ul style="list-style-type: none">- 2 tables SQL, une pour les données historiques et une autre pour les données streaming.- Une DB Mongo/Elastic comportant 2 collections: une pour les données stream et une autre pour les données historiques.	<p>142 - SQL (Architecture des données)</p> <p>Elasticsearch</p> <p>143 - MongoDB</p>	<p>Une base de données relationnelle</p> <p>Un fichier de requête SQL pour montrer que c'est bien fonctionnel</p> <p>Même rendu mais exemples de requêtes Elastic/Mongo</p>



Étapes 3 : Consommation de la donnée



Etape	Description	Objectif	Modules / Masterclass / Templates	Conditions de validation du projet
3	Consommation de la donnée	Implémenter un Dashboard qui permettrait de requêter sur la bdd streaming pour suivre une évolution en direct du cours OU utiliser un algo de Machine Learning appliqué à la finance qui permettra de retourner une décision d'achat ou non. (Quelle stratégie pour la vente ?) Aller plus loin: prédiction de gains.	ElasticSearch, Kibana, Dash (module complémentaire) DE120 - ML pour les DE DE121	Notebook ou tableau de bord Dash



Étapes 4 : Déploiement



Etape	Description	Objectif	Modules / Masterclass / Templates	Conditions de validation du projet
4	Déploiement	<p>Faire une API pour:</p> <ul style="list-style-type: none">- <u>Si choix du dashboard</u>: Être capable de requêter des données dans les BDDs créées. Ex: Le cours actuel de la paire de trading, les prix moyens au cours des dernières années, l'évolution du cours de la monnaie par rapport à une moyenne mobile... aucune limite sur les données que vous souhaiteriez mettre en avant- <u>Si choix de ML</u>: Mettre en avant les prédictions du modèle de ML <p>Implémenter des tests unitaires pour tester votre API.</p>	FastAPI, Flask	Fichiers de l'API
		<p>Dockeriser tout le projet pour qu'il soit reproductible sur n'importe quel machine</p> <p>Docker-Compose des différents conteneurs + réseau + ports pour API.</p>	Docker	<p>Fichier Yaml du docker-compose</p> <p>Possible de faire un setup.sh pour la création des images</p>



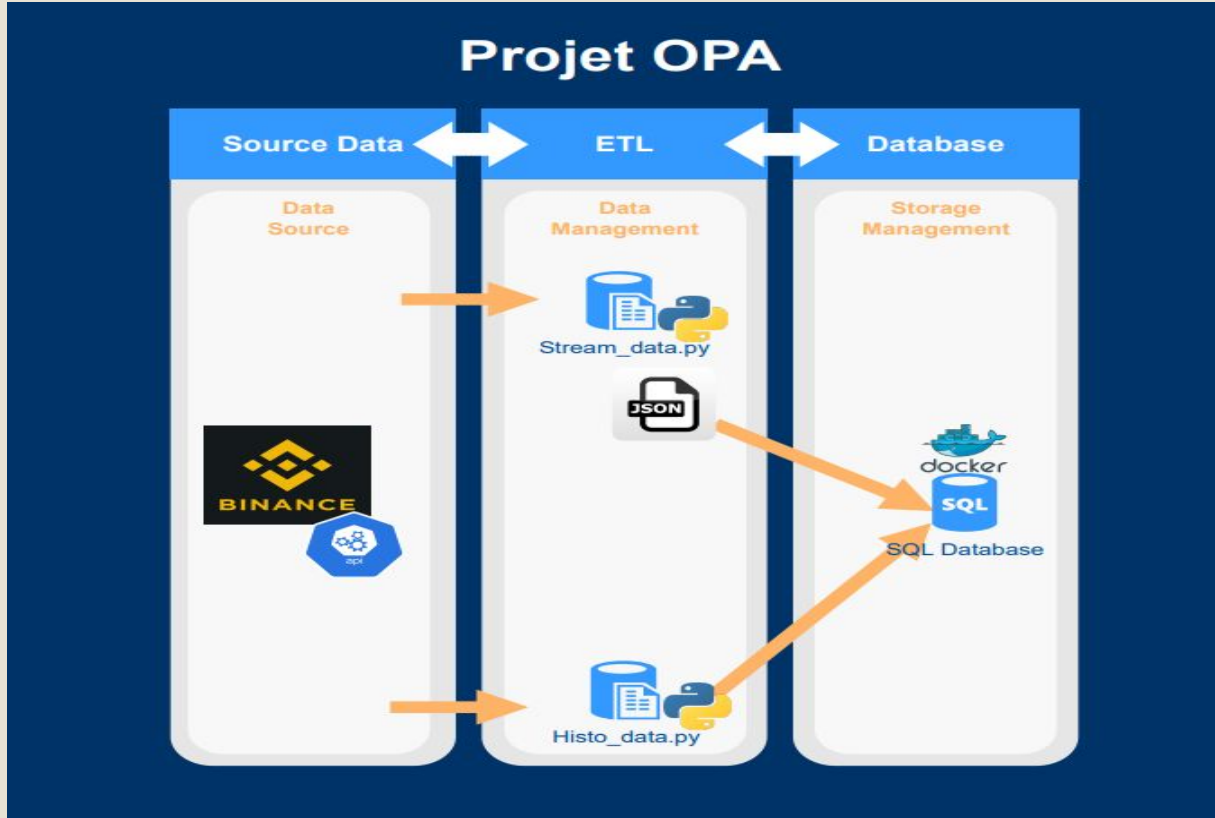
Étapes 5 : Automatisation des flux (facultatif)



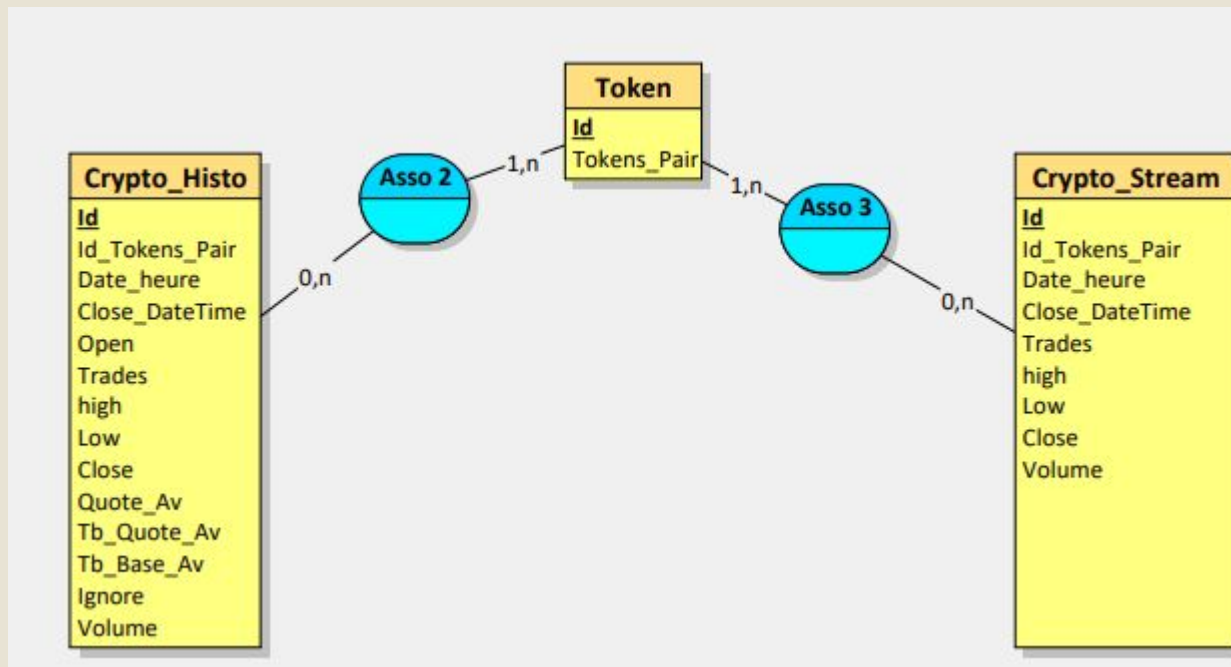
Etape	Description	Objectif	Modules / Masterclass / Templates	Conditions de validation du projet
5	Automatisation des flux (facultatif)	Il faudra requêter l'API quotidiennement via Airflow. Nous ne pouvons pas utiliser Airflow pour les données en streaming, mais vous pouvez tester Nifi.	Airflow, (Nifi)	Fichier python du DAG



Étapes 5 : Schéma Architecture



Étapes 5 : Schéma Table



Étapes importantes du projet

Récolte des données

Identification d'une source de données

Connexion et début d'importation des datas

Architecture & Stockage

Choix d'un SGBD adapté au problème

Modélisation des tables/collections/index



Remise d'un document expliquant l'architecture choisie ainsi que la modélisation



Étapes importantes du projet

Consommation des données

Dashboard ou Algorithme de ML

Etape normalement incombant au Data Scientist/Analyst

Déploiement

Création de l'API de votre projet

Dockerisation de tout le projet



Remise d'un rapport final du projet.

Rapport expliquant tout le workflow décisionnel du projet

Un Github propre sur lequel le groupe aura travaillé



DES QUESTIONS ?

