1ere approche proposée :

Livrer un MVP (Minimum Viable Product)

Le but est d'avoir un produit fonctionnel, du scraping au site. Faire des tests. On cherche uniquement à avoir un produit fonctionnel (on laisse les performances de côté pour l'instant)

On peut déjà faire une grosse partie du scraping et le preprocessing des données scrapées.

Délais estimés: 2-3 mois ? A débattre. Dépend du state of the art.

State of the art:

--> Faire une revue de la littérature sur ce qui a déjà été fait dans le domaine. Cela nous donnera les pistes de modeles à essayer, les performances atteignables etc.

--> Ca **oriente aussi bcp la partie preprocessing** (tous les modeles n'acceptent pas les memes inputs). Mais **n'affecte a priori pas la partie scraping** puisque le but est de tout scraper puis de selectionner les données utiles dans un second temps.

--> Peut potentiellement écourter la partie collecte de données si le sujet a déjà été traité et que des datasets annotés sont disponibles. Rk: Peu probable car même si le sujet a déjà été traité, ce sera surement aux US et la donnée sera peut etre trop différente pour entrainer notre modele sur des propriétés fr.

Scraping:

--> Extraire autant d'informations que possible de chaque site web. Si, pour une raison ou une autre, 1 ou 2 sites sont plus difficiles à extraire (e.g. captcha) ou posent des problèmes d'ordre juridique, les laisser de côté pour l'instant

Preprocessing:

--> Contiendra a minima une partie NLP pour extraire toutes les infos des annonces. Pour certains sites, peut-être pas nécessaire (e.g. sur Airbnb il y a déjà pleins de champs remplis et ça suffit peut être de les extraire directement) mais c'est toujours un plus.

--> Proposer un ou deux mappings des différentes infos scrapées sur chaque site pour avoir déjà des inputs de prêts pour entrainer le modèle. On pourra proposer plusieurs mappings, quitte à développer plusieurs modèles ensuite si c'est trop compliqué de tout mapper ensemble ou bien laisser 1-2 sites de cotés si les données sont galère à mapper.

Collecte de données:

--> On a besoin de constituer un portfolio de propriétés à protéger avec tous les features correspondant des propriétés à protéger pour entrainer le modèle à reconnaitre les annonces frauduleuses correspondant à ces propriétés --> Surement besoin de **contacter des agences immo** pour ca. Si c'est à nous de le faire, **rajoute des délais**. A voir si possible de trouver de la donnée d'entrainement pour ca mais parait compliqué.

--> Possible de se constituer un petit test set à la main pour tester un peu le modèle et atteindre/établir une baseline performance.

Dev Web: (à préciser)

--> Avoir un site fonctionnel capable de traiter les outputs du modele et d'envoyer un notif mail en cas de résultat positif.

--> L'esthetisme n'a aucune importance à ce stade.

--> En termes de stockage des données, une architecture très simpliste devrait suffire car le volume est faible (on stocke uniquement les features des maisons de notre portfolio dans un csv et les contact details des proprios.

--> Pas besoin de s'occuper de la sécurité à ce stade car on déploie rien ?

Testing and debugging:

--> Pour tester la pipeline, on peut éventuellement proposer un modèle simpliste qui n'utilise pas l'IA (e.g. geomatching d'addresses) mais qui a vocation à vérifier que tout marche bien au niveau du scraping, du preprocessing des données scrapées, et surtout au niveau du site.

Restera à faire:

Améliorer le modèle

Intégrer les sites plus difficiles à scrapper

Implémenter un premier modèle de ML (a priori apprentissage semi-supervisé comme un score de similitude ou un arbre de décision) simple et mesurer ses performances