



#### MARKETPLACE IMMOBILIÈRE

L'objectif du projet est de mettre en place une place de marché ou s'échange des biens immobiliers (maisons, appartements, etc) au travers d'un smart contract sur Ethereum.

Il devra probablement posséder des fonctions comme :

- Mettre en vente
- Acheter

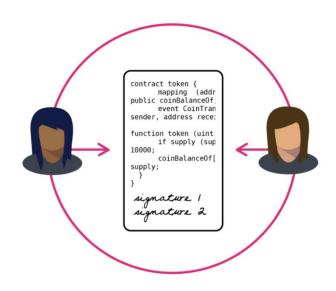
Il faudra donc développer une application web avec une interface permettant de mettre en vente un bien sur le smart contract et acheter un bien qui a été mis en vente.



#### SMART CONTRACT

#### Créer un **smart contract** gérant :

- La mise en vente avec le prix, l'adresse, le nombre de m², la description, le hash des documents, l'adresse publique du vendeur, et autres (date, nombre de chambre, etc).
- L'achat qui prend en paramètre l'id de la maison mise en vente, l'adresse de l'acheteur,



# FRONT

### Créer un **front end** permettant de gérer :

- Visualiser les propriétés mises en vente
- Cliquer sur une propriété et voir ses infos
- Acheter la propriété
- Mettre en vente une propriété



# WEB3.JS

Connecter le front avec le smart contract en utilisant des appels web3.js

- Instancier l'objet web3
- Instancier l'objet **contract**
- Faire appel aux fonctions du smart contrat en utilisant des **.send** ou **.view** (paramètres comme ether envoyé, adresse avec qui on appelle la fonction)



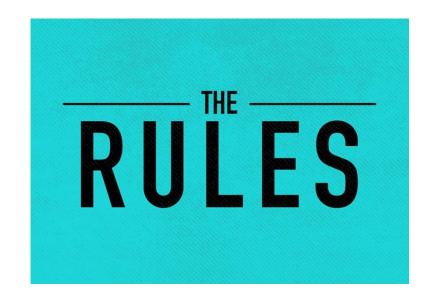
#### CONSIGNES

Cette application peut être hébergé en local (m'envoyer le code ou le lien du git) ou sur un serveur distant.

Un livrable présentant votre projet, son histoire, ses limites et ses possibles améliorations en PDF à rzgoda@myges.fr

L'utilisation de git ou svn est conseillé.

Attention à ne pas copier collé du code présent sur internet pour le smart contract.



### EXEMPLE

You are connected has: 0x8FAC2BcCA1f5397BF2Bc96aA4aE8735728882761

#### CoinHouse

Take & Share your house with Ethereum

Share house

### Travelling around the world











# EXEMPLE

CoinHouse Home Share my house

You are connected has: 0x6FAC2BcCA1f5397BF2Bc96aA4aE8f35728882761





#### A beautiful appartment

Delete this house

27 Rue Lonnesse, 75019 PARIS, France

Owner : Rafal Price : 1 Ethereum

#### About

Microwaves

The newest coworking space around, with breathtaking views of the Thames! Us&Co is a spacious and comfortable workspace with fantastic amenities including a coffee bar, outdoor patio seating for those sunny days, and of course super fast wifi.

#### Tsss... You can't buy your own house!

	SIZE (M2) NB BEDROOM		NB ROOM	
	60	4	4	
Others				
Coffee		Unlimited coffee and assorted tea. You're welcome to grab a cup in the kitchen and make yourself a cup of coffee.		
Printing		Unlimited coffee and assorted tea. You're welcome to grab a cup in the kitchen and make yourself a cup of coffee.		

Unlimited coffee and assorted tea. You're welcome to grab a cup in the kitchen and make yourself a cup of coffee.

# CRITÈRES DE NOTATION

#### <u>Critères de notation :</u>

-	Le fonctionnement du smart contract	/5pts
-	La qualité du code, commmentaires, documentation	/3pts
-	L'utilisation de réseau de test comme Koban, Rinkeby ou Ropsten	/2pts
-	L'utilisation d'une application avec web3.js	/4pts
-	L'utilisation de frameworks JS (React/VueJS)	/1pts
-	L'utilisation d'un gestionnaire de version (git, svn)	/1pts
-	Soutenance	/5pts
-	Livrable	/3pts

Les critères de notation sont données à titre indicatif, une réévaluation des critères est fortement envisageable.



Projet:

Vote en ligne

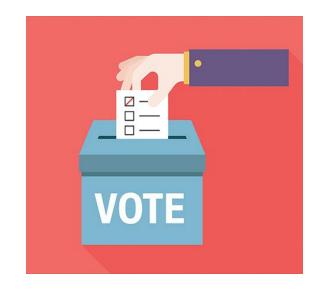
#### DÉMOCRATIE ET VOTE

Communément, la méthode de dialogue avec les citoyens dans une démocratie se fait au travers du vote.

Il est important de noter que les règles qui régissent le fonctionnement du vote peuvent avoir une influence drastique sur le résultat d'une élection.

L'établissement des règles régissant le vote au travers de smart contract peut permettre d'empêcher la fraude, d'avoir un système plus transparent et simple.

Comment les règles du scrutin peut avoir un impact sur le résultat final d'une élection et comment les smart contract peuvent révolutionner ce secteur?



#### LES MODES DE SCRUTIN EXISTANT

ScienceEtonnante:

Réformons l'élection présidentielle! — Science étonnante #35

https://www.youtube.com/watch?v=ZoGH7d51bvc

La statistique expliquée à mon chat :

Monsieur le président, avez-vous vraiment gagné cette élection ?

https://www.youtube.com/watch?v=vfTJ4vmIsO4

Wikipedia:

Les différents modes de scrutin

https://fr.wikipedia.org/wiki/Syst%C3%A8me %C3%A9lectoral#Les diff%C3%A9rents modes de scrutin et syst%C3%A8mes %C3%A9lectoraux

PROJET: IMPLÉMENTATION D'UN MODE DE SCRUTIN AU CHOIX

AVEC SOLIDITY

#### CONSIGNES

L'objectif du projet est de réaliser un smart contract permettant de **simuler une élection** dans le cadre de votre démocratie idéale (choix d'un mode de scrutin). Il devra probablement posséder des fonctions comme :

- Se présenter à l'élection
- Voter
- Clôture de l'élection
- Récupérer les résultats de l'élection

#### L'objectif du projet n'est pas de faire un simple scrutin majoritaire à un tour!

Il vous sera aussi demandé de développé une application web avec une interface permettant de voter au travers du smart contract et regarder les résultats de l'élection. Cette application peut être hébergé en local (m'envoyer le code ou le lien du git) ou en ligne.

Un livrable présentant votre projet, son histoire, ses limites et ses possibles améliorations en PDF à rzgoda@maltem.com

L'utilisation de git ou svn est conseillé. Attention à ne pas copier collé du code présent sur internet pour le smart contract.



# WEB3.JS

Connecter le front avec le smart contract en utilisant des appels **web3.js** 

- Instancier l'objet web3
- Instancier l'objet **contract**
- Faire appel aux fonctions du smart contrat en utilisant des .send



# CRITÈRES DE NOTATION

#### <u>Critères de notation :</u>

-	Le fonctionnement du smart contract	/5pts
-	La qualité du code, commentaire, documentation	/3pts
-	L'utilisation de réseau de test comme Koban, Rinkeby ou Ropsten	/2pts
-	L'utilisation d'une application web avec web3.js	/4pts
-	L'utilisation de frameworks JS (React/VueJS)	/1pts
-	L'utilisation d'un gestionnaire de version (git, svn)	/1pts
-	L'originalité du projet	/1pts
-	Complexité du mode de scrutin	/3pts
-	Soutenance	/5pts
-	Livrable	/3pts

Les critères de notation sont données à titre indicatif, une réévaluation des critères est fortement envisageable.





Application mobile de compétition de trading

### **TRADING**

Aujourd'hui, on peut trouver de nombreuses applications de trading de cryptomonnaie sur l'AppStore et le PlayStore.

En revanche on ne trouve aucun jeu permettant de mettre en compétition plusieurs traders sur une période donnée.

Pourtant, de nombreux traders aiment comparer leurs performances et se vanter de leurs superbes trades.

Ainsi, l'endroit parfait pour comparer ses performances serait une application mobile où deux traders pourrait s'affronter sur une période donnée.



#### LES RESSOURCES POUR LE PROJET

Ox Radar:

Protocole pour créer facilement un exchange décentralisé (mise en place du trading)

https://0x.org/

Solidity DateTime Contract

Gestion du temps sur le smart contract

https://github.com/pipermerriam/ethereum-datetime

**Fortmatic** 

Alternative à Metamask pour mobile (user-friedly et custody)

https://fortmatic.com

PROJET: IMPLÉMENTATION D'UN APPLICATION MOBILE DE COMPÉTITION DE TRADING

#### CONSIGNES

L'objectif du projet est de réaliser un smart contract permettant de **mettre en place une compétition de trading.** 

Il devra probablement posséder des fonctions comme :

- Créer une nouvelle compétition
- Lancer la compétition
- Récupérer les résultats d'une compétition

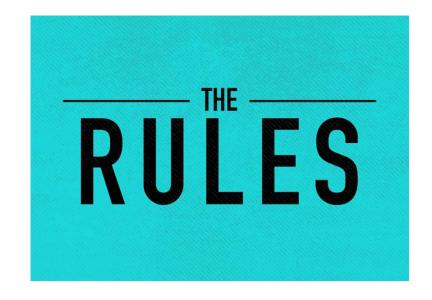
Il vous sera aussi demandé de développé une application mobile avec une interface permettant de lancer la compétition au travers du smart contract, trader et regarder les résultats de la compétition.

Le but de l'application est de mettre en place un pari sur un smart contract et payer automatiquement le gagnant à la fin de la compétition.

Un livrable présentant votre projet, son histoire, ses limites et ses possibles améliorations en PDF à rzgoda@maltem.com

L'utilisation de git ou svn est conseillé.

Attention à ne pas copier collé du code présent sur internet pour le smart contract.



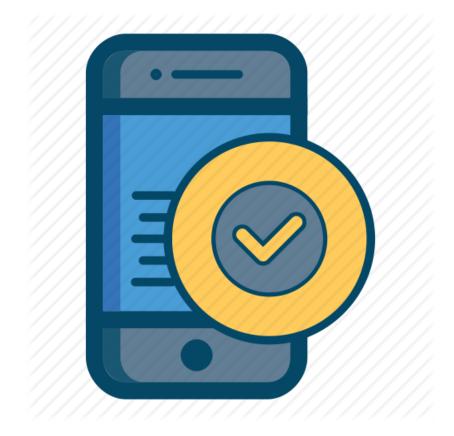
# WEB3.JS

Utiliser la doc de Fortmatic pour faire des appels au smart contract



# APP MOBILE

Utilisation de Cordova, Ionic ou autres framework pour faciliter le développement mobile



# **D**IFFICULTÉS

#### Dans un soucis de simplicité :

- Qu'une seule currency est tradé et vous recevrez une fonction qui récupère les prix de ce marché (ETH/USD)
- La fin de la compétition n'est pas automatiquement close, mais uniquement lorsque quelqu'un appelle la fonction getWinPrice

# Influe

# Plateforme web de tokenisation d'un influenceur



Face au changement continuel des méthodes de communication, les marques doivent faire preuve de créativité pour développer des procédés efficaces capables d'atteindre les consommateurs exigeants et averses aux publicités intrusives.

Passer par l'intermédiaire d'un influenceur permet de crédibiliser la campagne par un message plus authentique et proche, relayé par un leader d'opinion.

Aujourd'hui, une marque passe principalement par une plateforme d'influence marketing (exemple : https://www.influence4you.com/) où elle sélectionne un influenceur pour véhiculer sa campagne marketing au travers de post sponsorisés, vidéos, tweet etc. Le fameux « placement de produit ».

# L'objectif de ce projet est de développer une plateforme web d'influence marketing sur laquel des influenceurs peuvent capitaliser sur leur image.

Sur cette plateforme, des influenceurs peuvent vendre leurs services (comme des posts, tweets, photos, etc.). Afin de vendre leurs services, ils doivent au préalable créer leur propre cryptomonnaie. Des investisseurs peuvent investir dans leur cryptomonnaie dans l'espoir d'une future plus-value. Les influenceurs quant à eux, peuvent vendre leur cryptomonnaie afin de lever des fonds pour développer leur audience et leur activité (achat caméra, ordinateur, logiciel, monteur, etc.). Cette cryptomonnaie est cotée sur des marchés comme Uniswap (https://info.uniswap.org/home)

Des marques peuvent acheter un post sponsorisé à cet influenceur.

L'influenceur peut ainsi lever des fonds en début de carrière et peut aussi investir sur sa propre image. Il devient ainsi son propre actionnaire. Des investisseurs peuvent acheter la cryptomonnaie de cet influenceur et valoriser l'image de l'influenceur et donc avoir un impact sur le prix de la cryptomonnaie

**Source:** https://tokentuesdays.substack.com/p/tokenizing-human-capital

# Dans la première version de cette application web, on doit pouvoir retrouver les fonctionnalités suivantes :

- un influenceur peut s'inscrire avec son compte Facebook http://www.passportjs.org/packages/passport-facebook/
- un influenceur peut émettre sa cryptomonnaie (jeton ERC20 https://github.com/ConsenSys/Tokens/blob/master/contracts/eip20/EIP20.sol et le déploiement avec https://web3js.readthedocs.io/en/v1.3.0/web3-eth-contract.html#deploy
- un influenceur peut créer un marché pour sa cryptomonnaie https://uniswap.org/docs/v2/smart-contracts/factory#createpair
- un investisseur peut visualiser l'ensemble des influenceurs avec le prix de leur cryptomonnaie sous la forme de *cards*
- un investisseur peut acheter ou vendre la cryptomonnaie d'un influenceur (avec son compte *Metamask*)



#### **Technologies:**

- Ethereum (la blockchain)
- Solidity (ERC20 est le standard à copier coller pour créer la cryptomonnaie)
- Web3.js (pour interagir avec les smart contracts)
- Infura (pour interagir avec la blockchain)
- Metamask (extension Chrome qui permet d'interagir avec la Blockchain)
- VueJS/ReactJS (front end)
- NodeJS (back end)
- NoSQL Database (la base de donnée)



#### Contact

Société Nice Mining Rig SAS ZGODA Rafal- Co-fondateur - 06.64.92.48.42 contact@niceminingrig.com