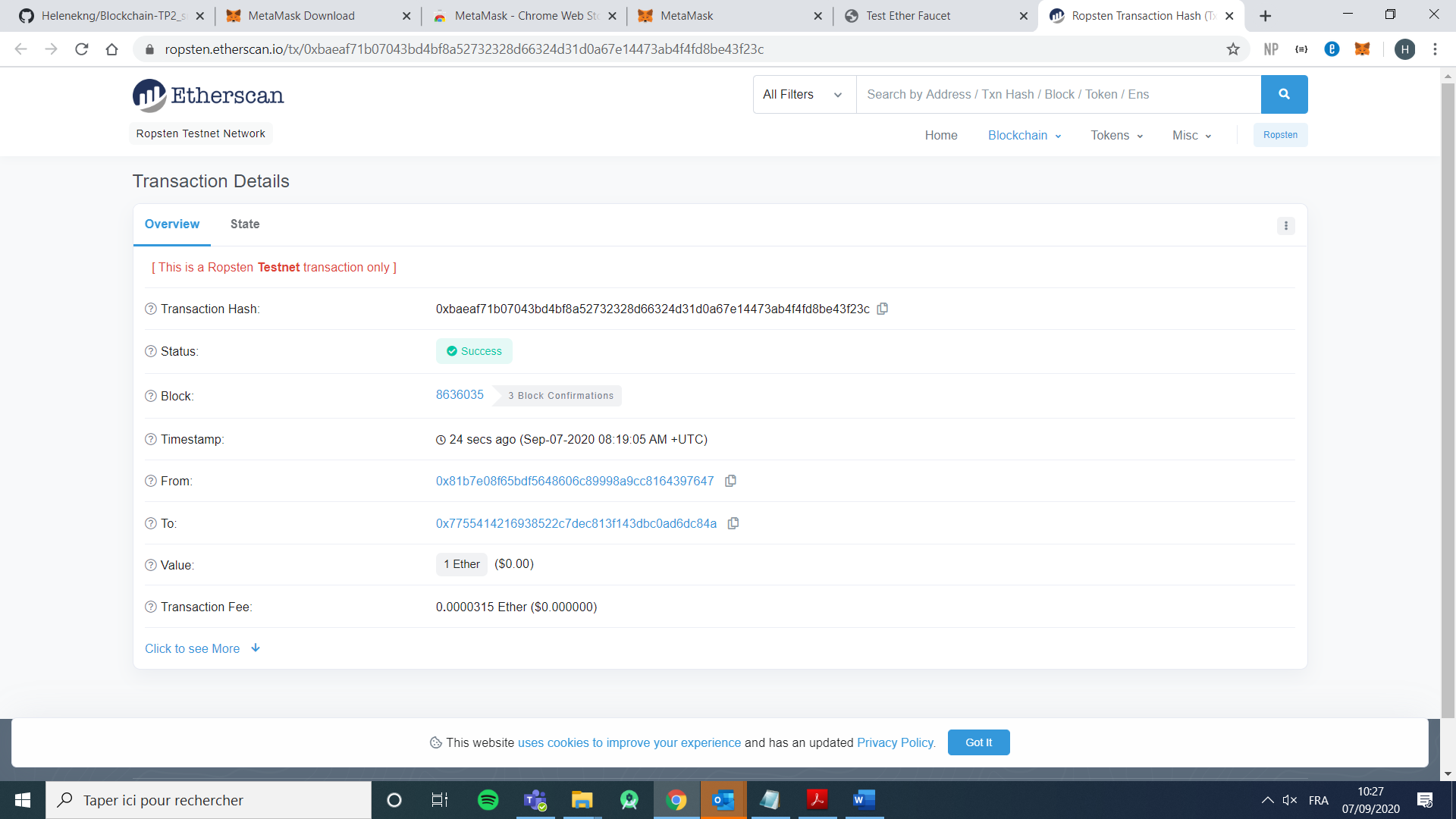
Hélène KAING A2I

**TP Blockchain :**

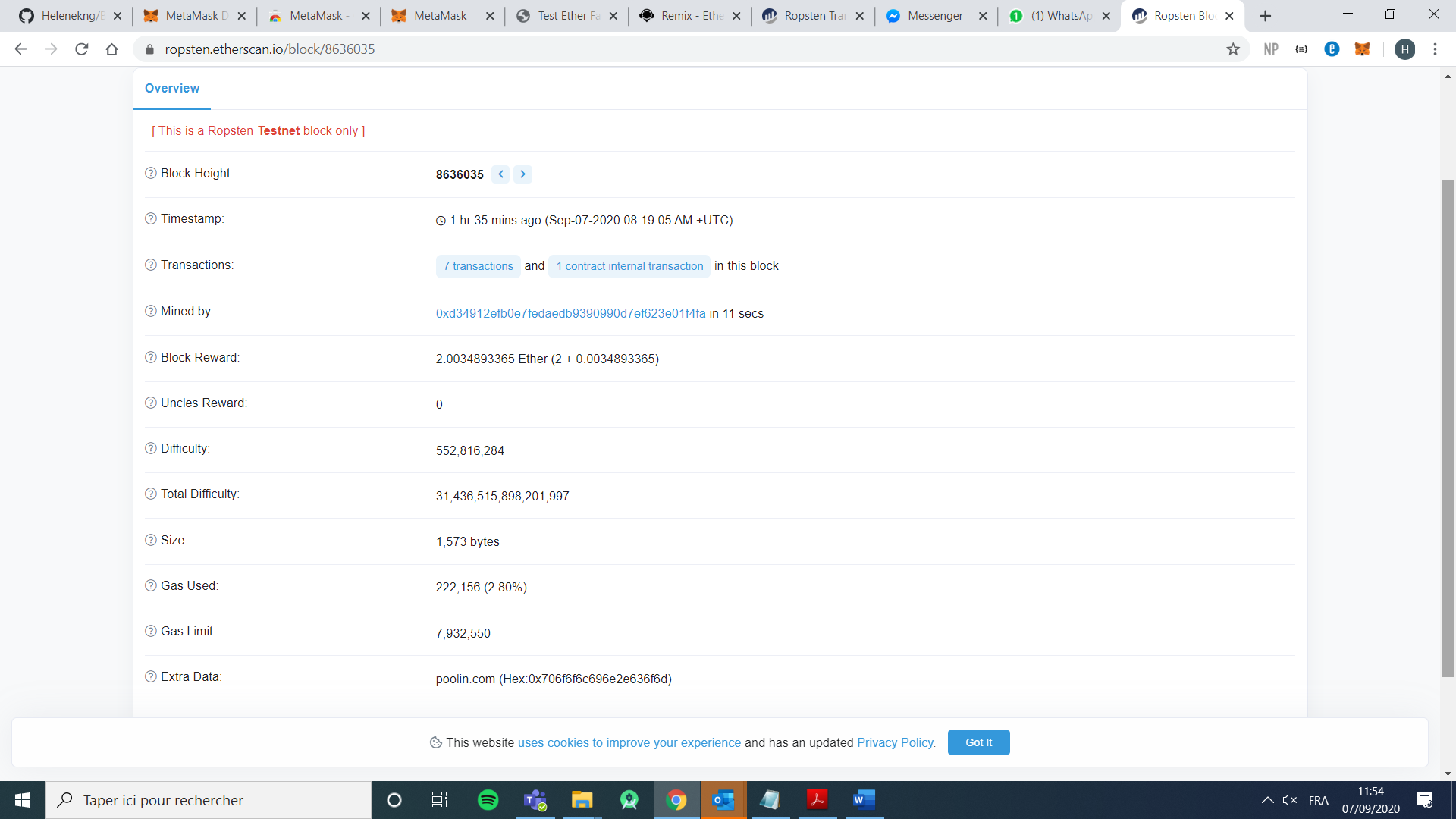
e. Réception d’1 éther :

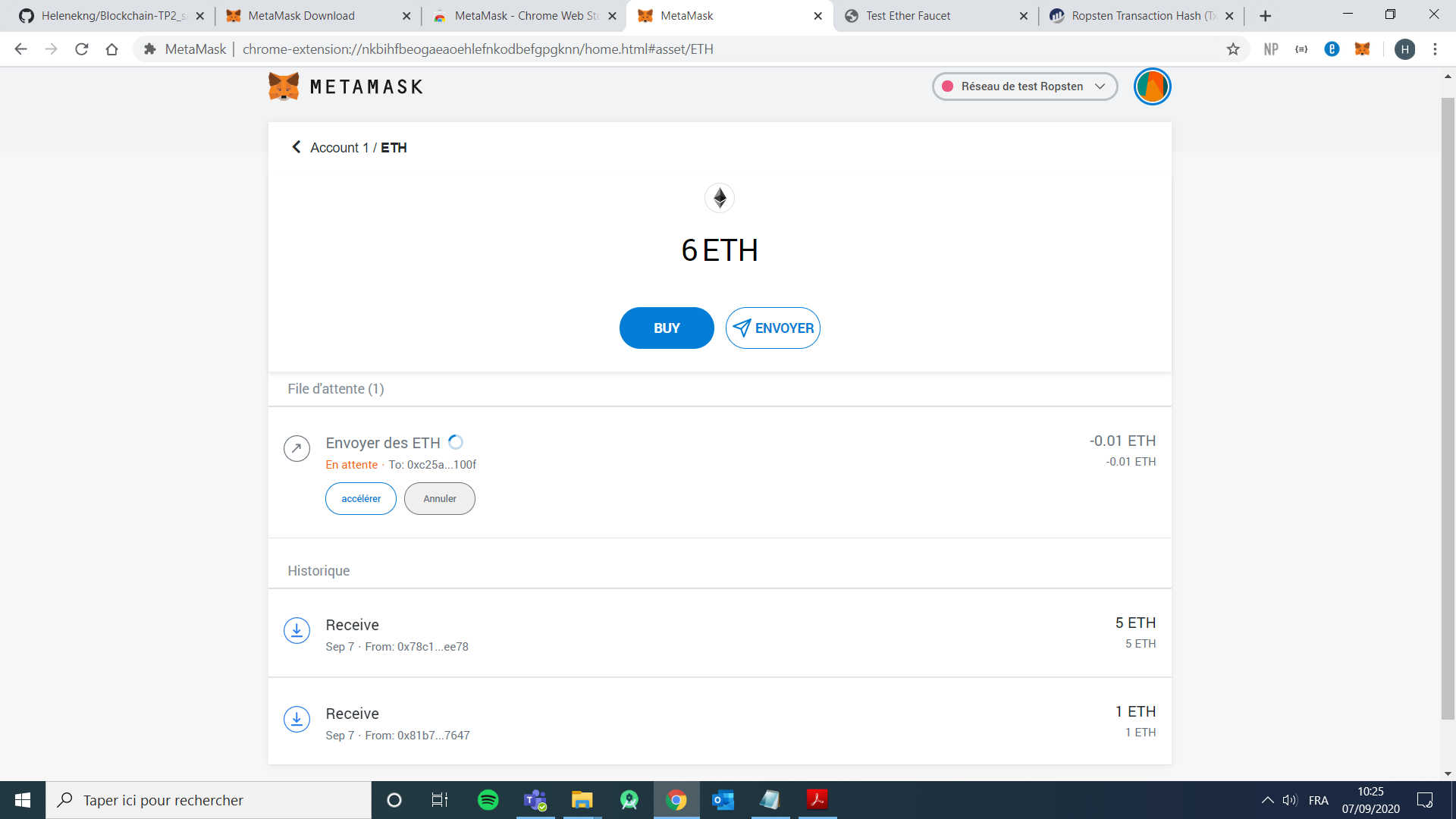


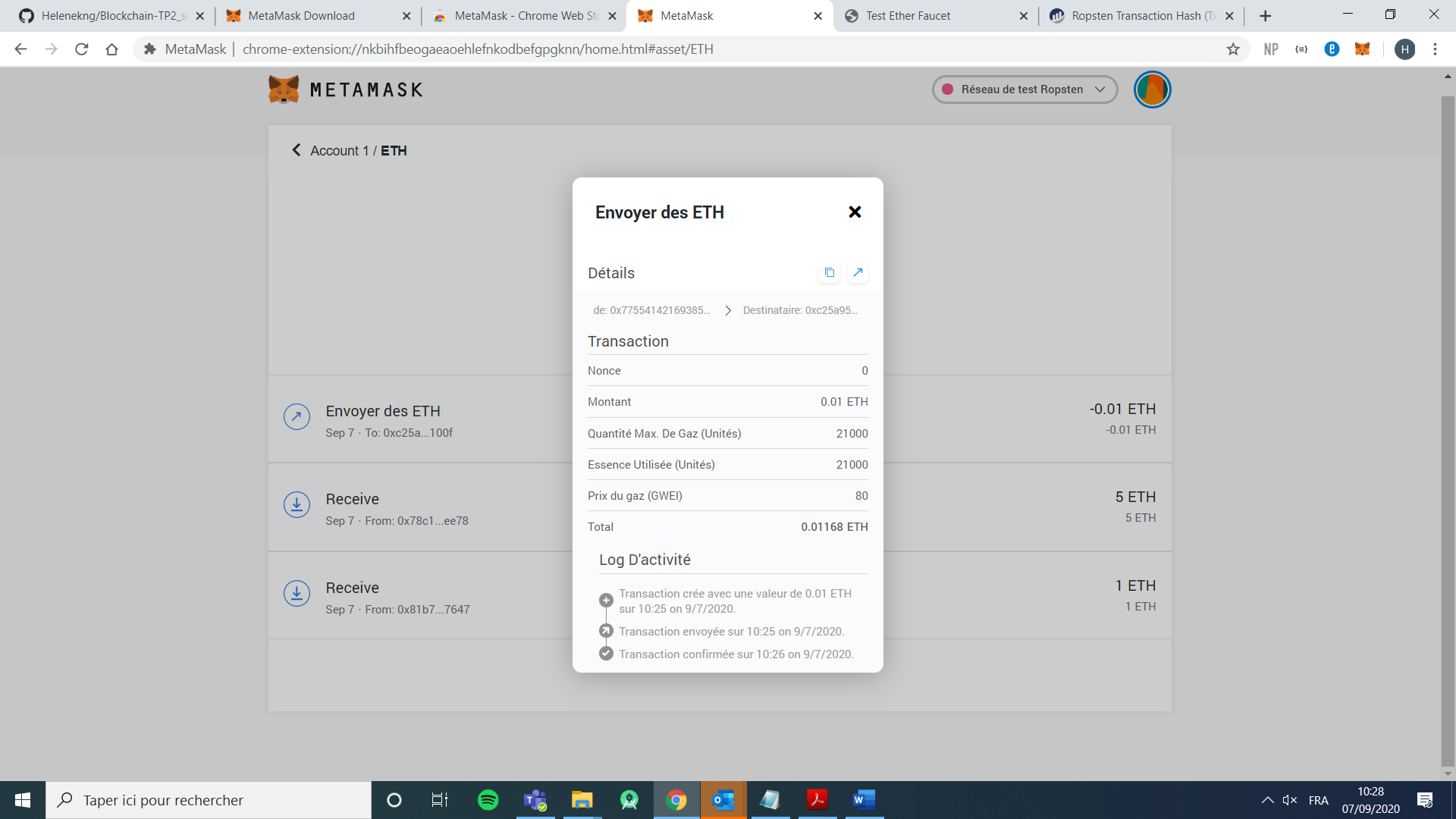
Montant transféré

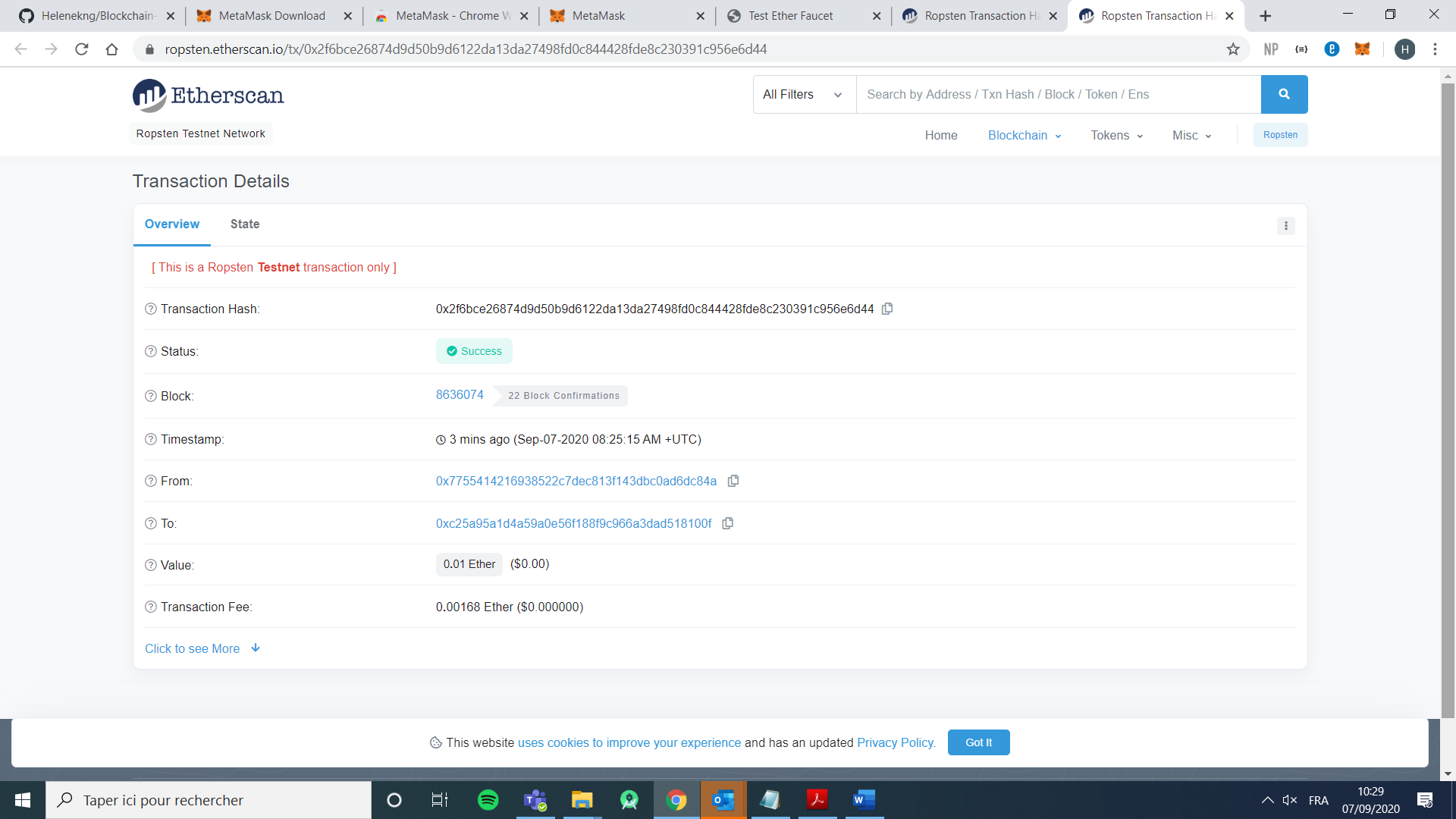
Adresse publique de mon portefeuille

f. Détail du bloc :

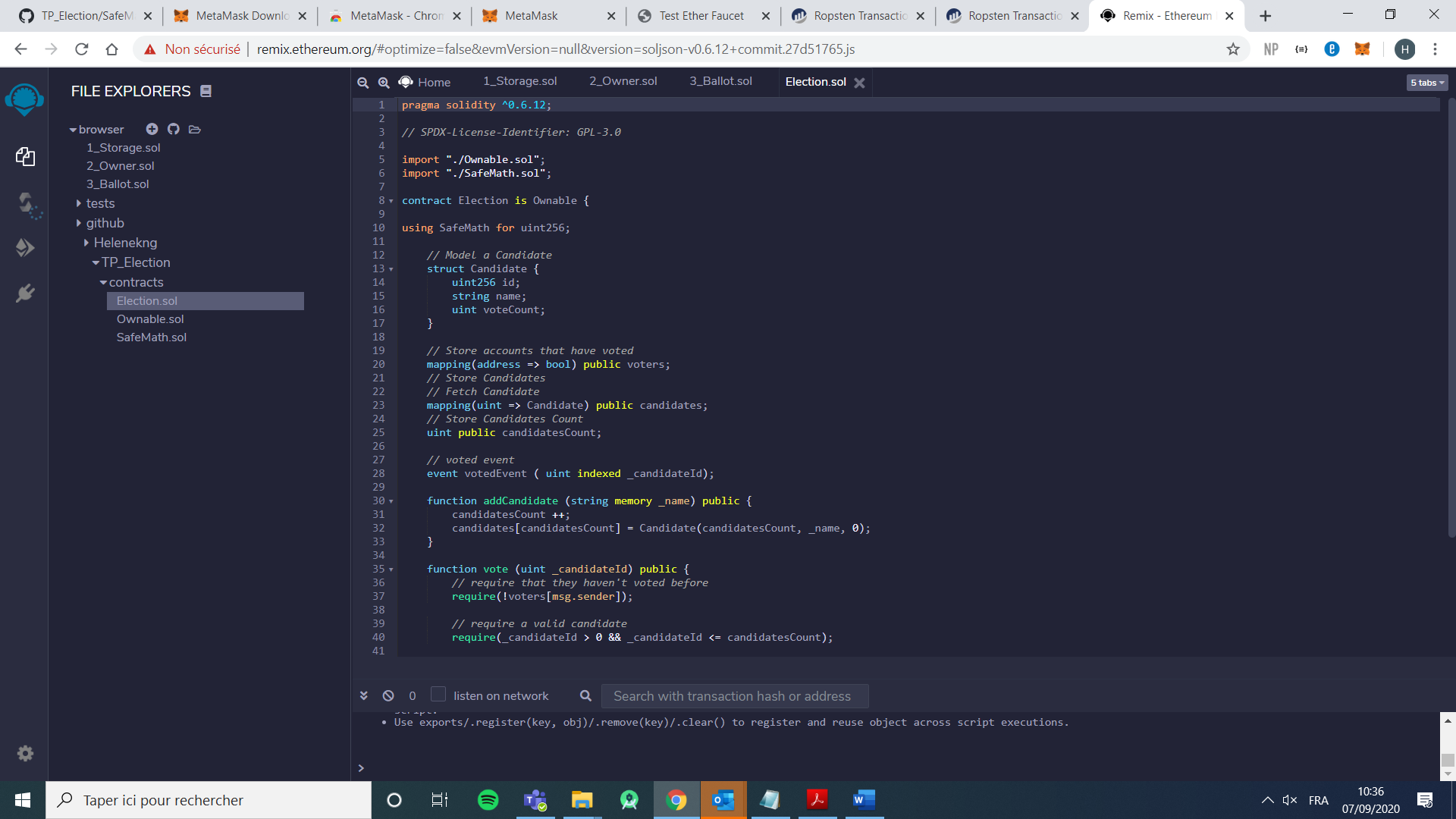


g. Envoie d’ethereum



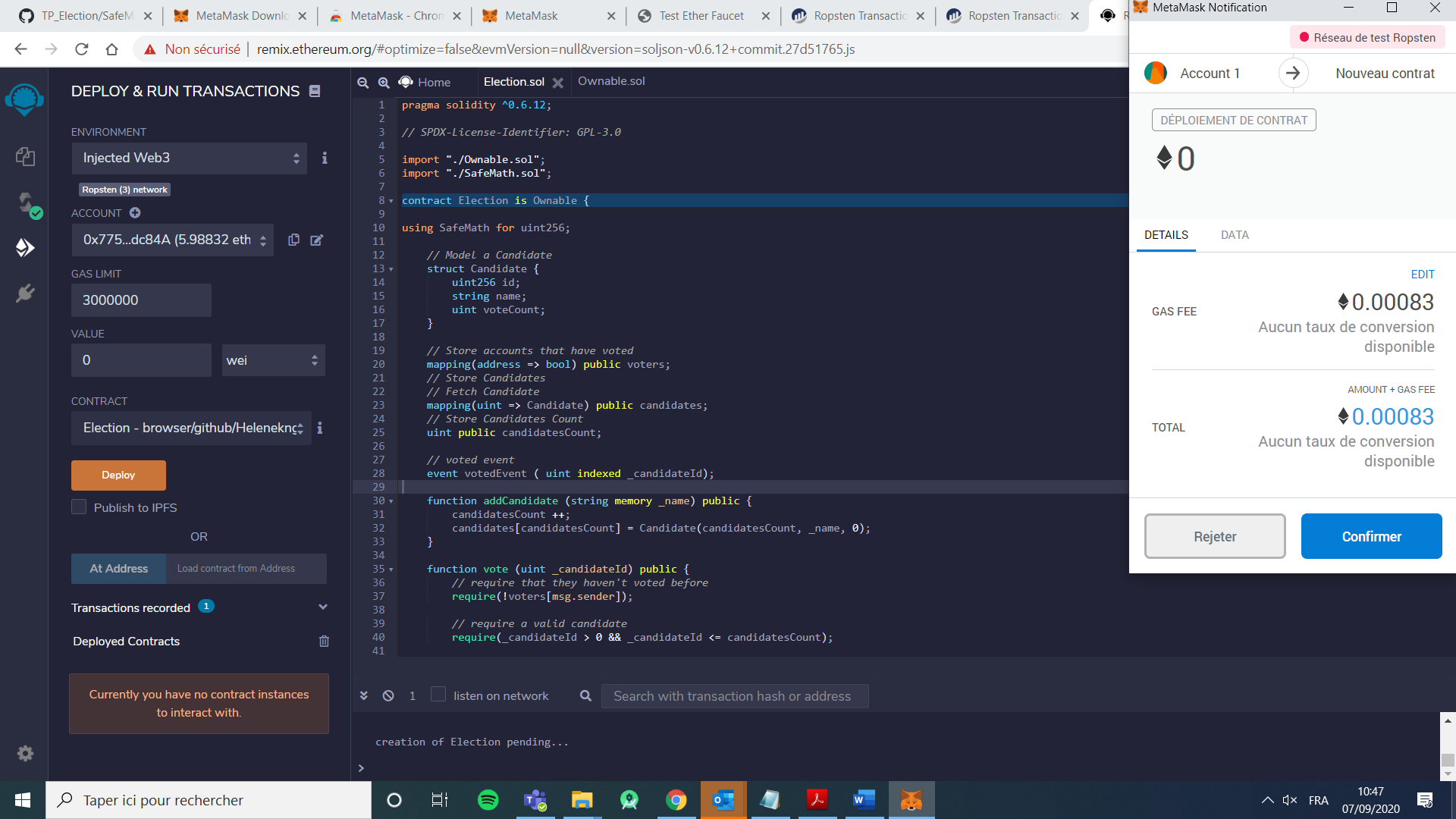


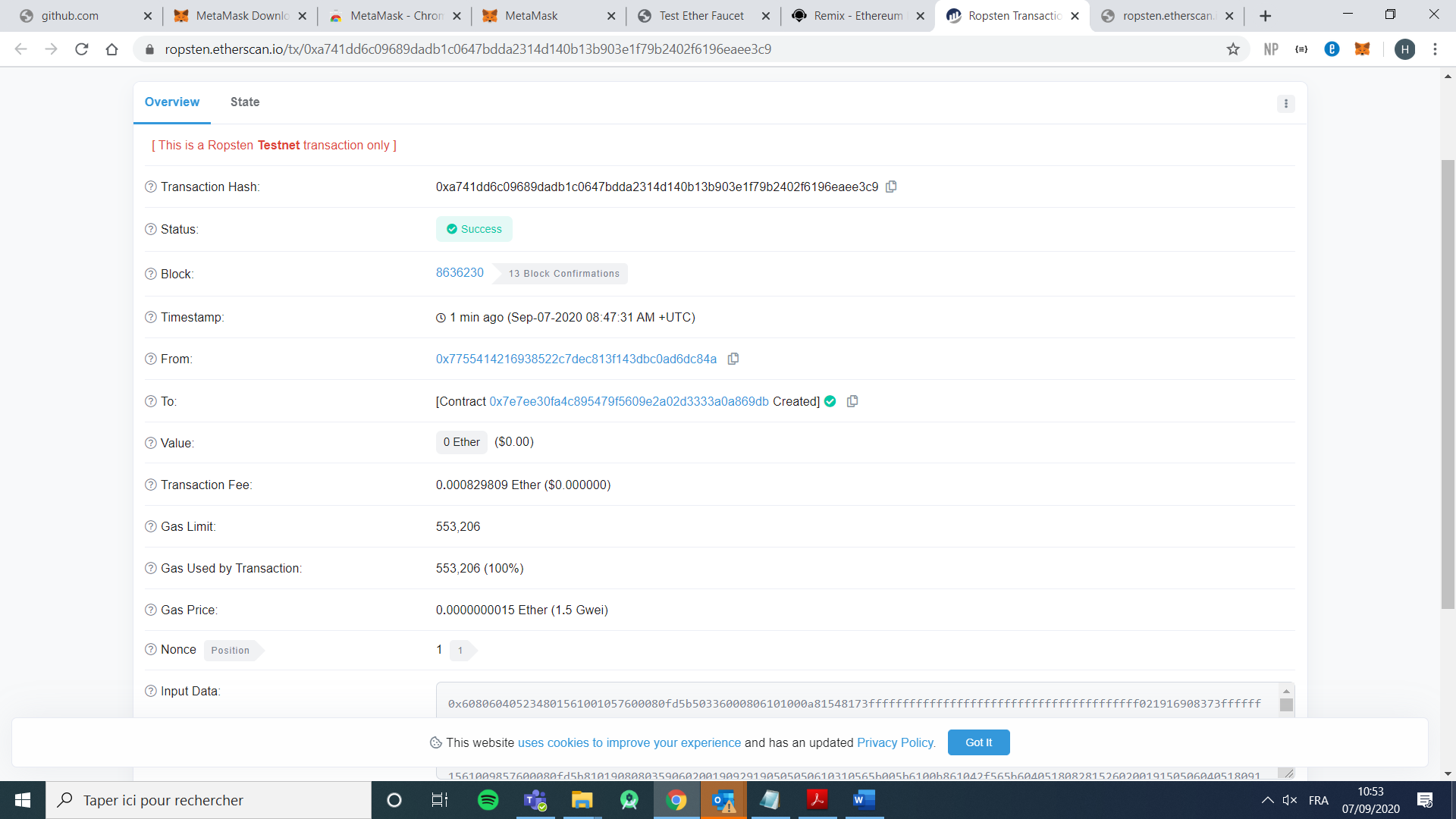
j.



k. Voir fichiers Election\_ABI.json et Election\_Bytecode.json : <https://github.com/Helenekng/Blockchain-TP2_smartcontract1>

m. Déploiement du smart contract



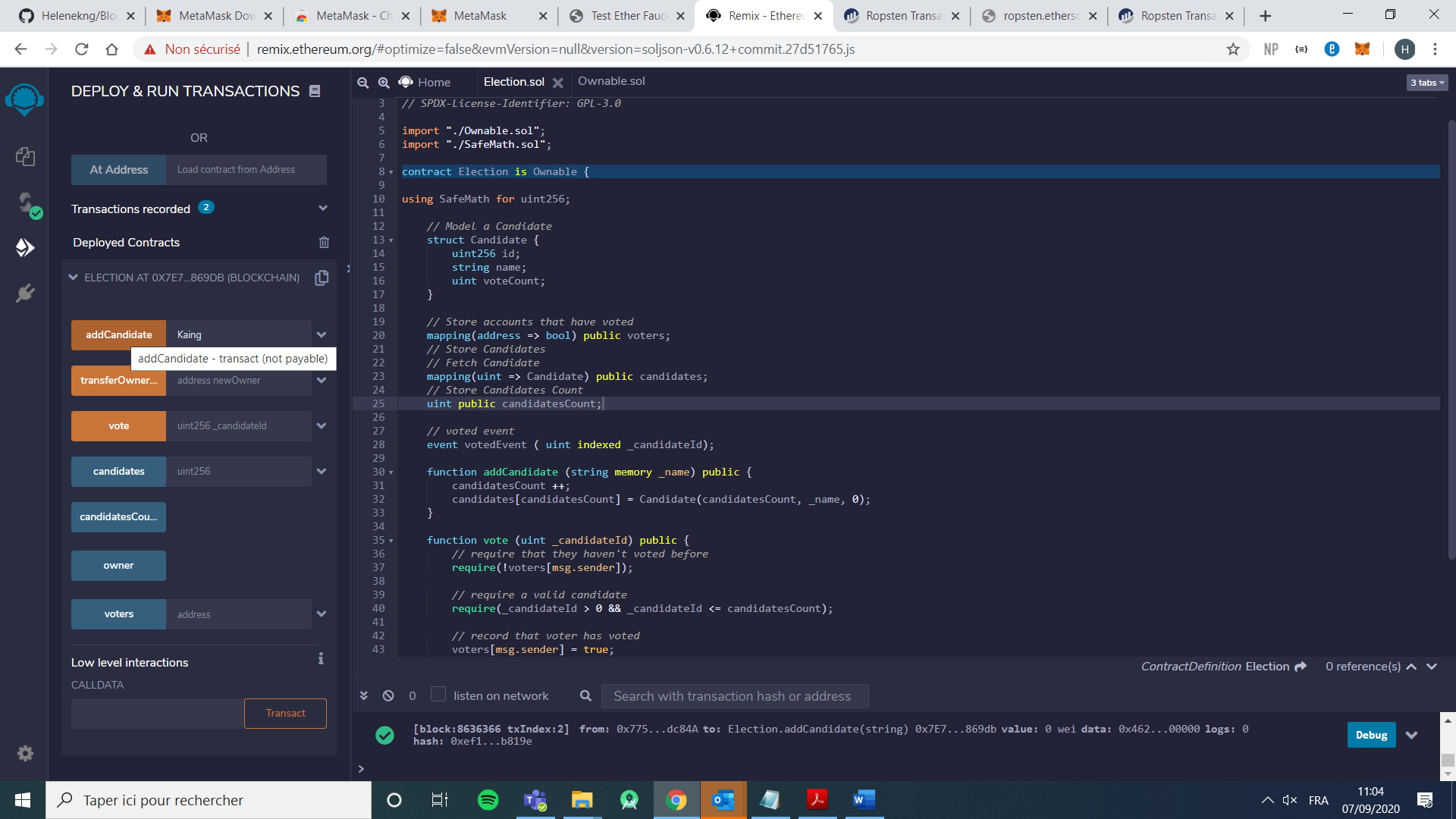
l. 

m.

Les frais de transactions sont différents car ceux évolue en fonction du nombre de transactions/de la saturation sur le réseau à l’instant T (voir ethereum.gasstation pour voir les frais à l’instant et estimer le bon moment, en fonction de la date de livraison également)

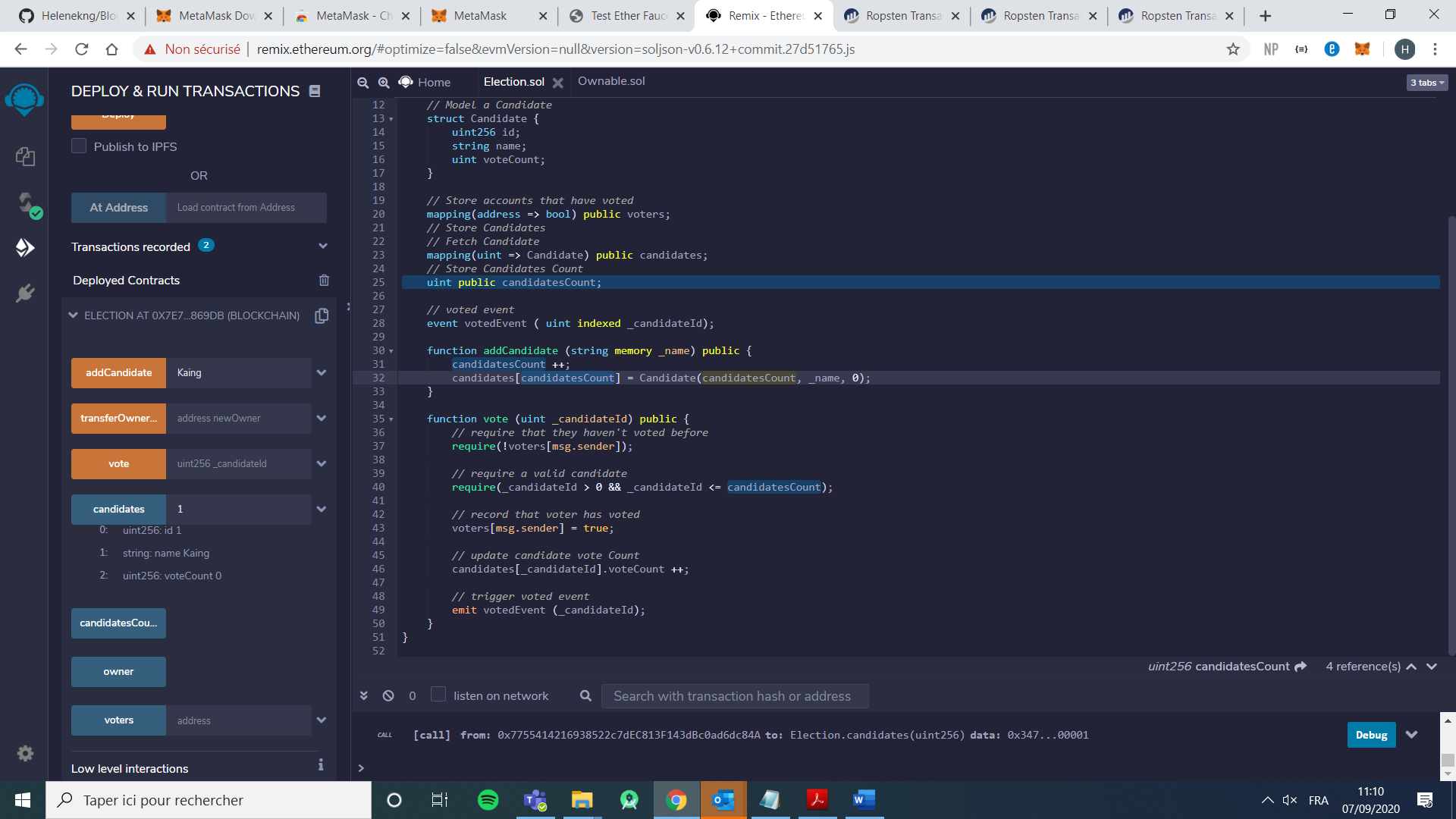
Adresse publique smartcontract : 0x7E7EE30Fa4c895479F5609e2a02d3333A0a869db

n. Add Canditates :

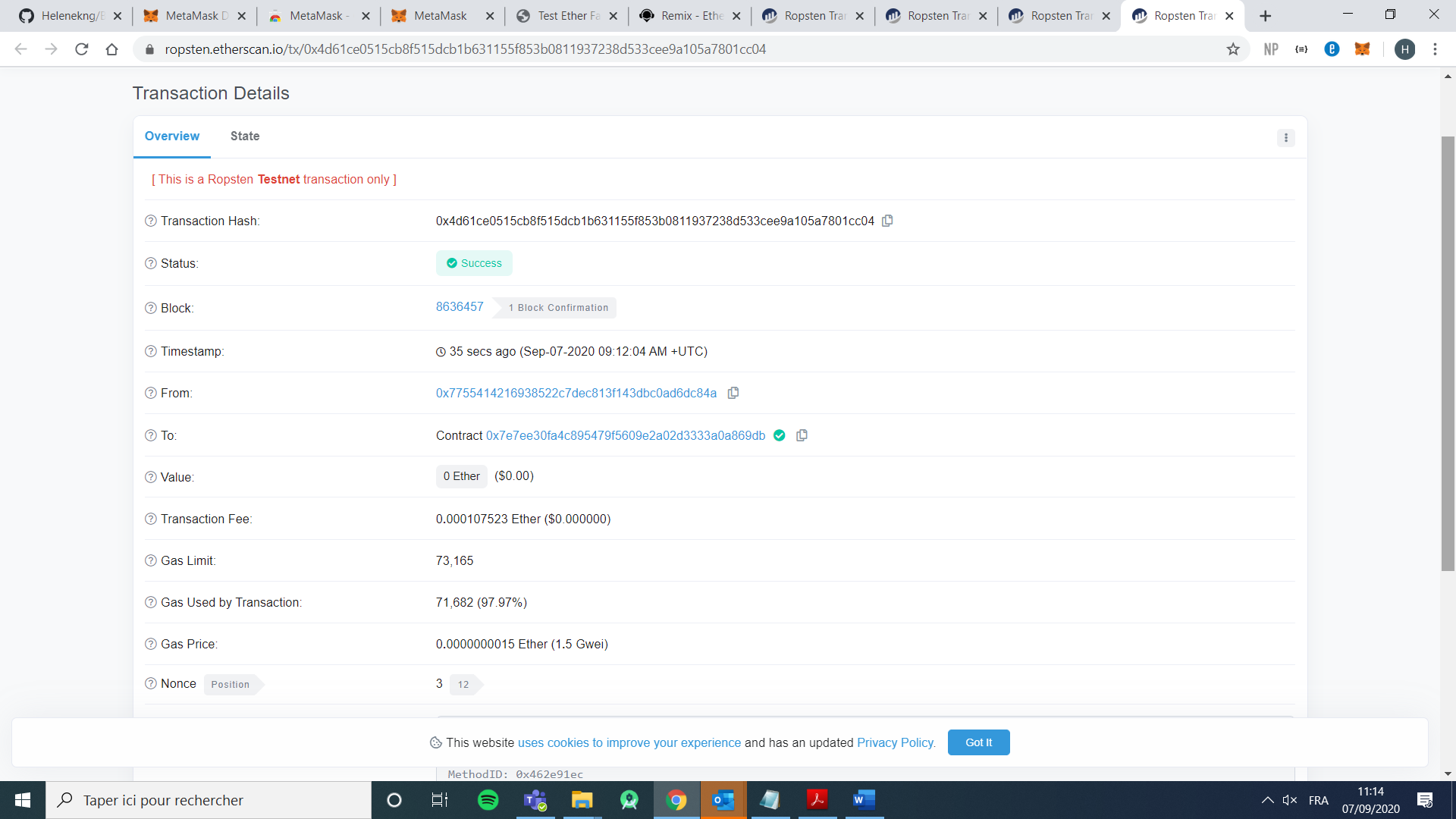


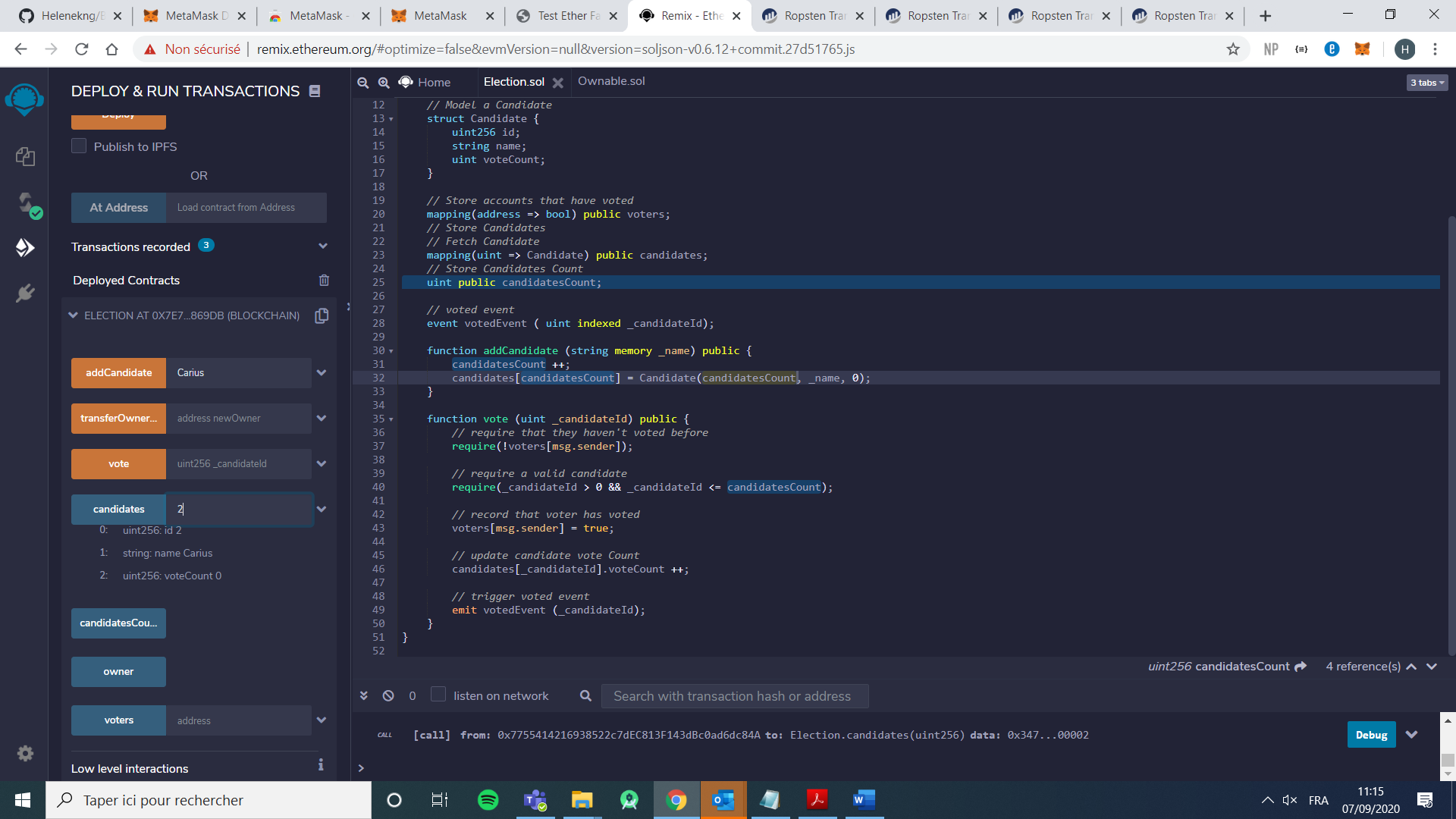


p. CandidateID = 1 :

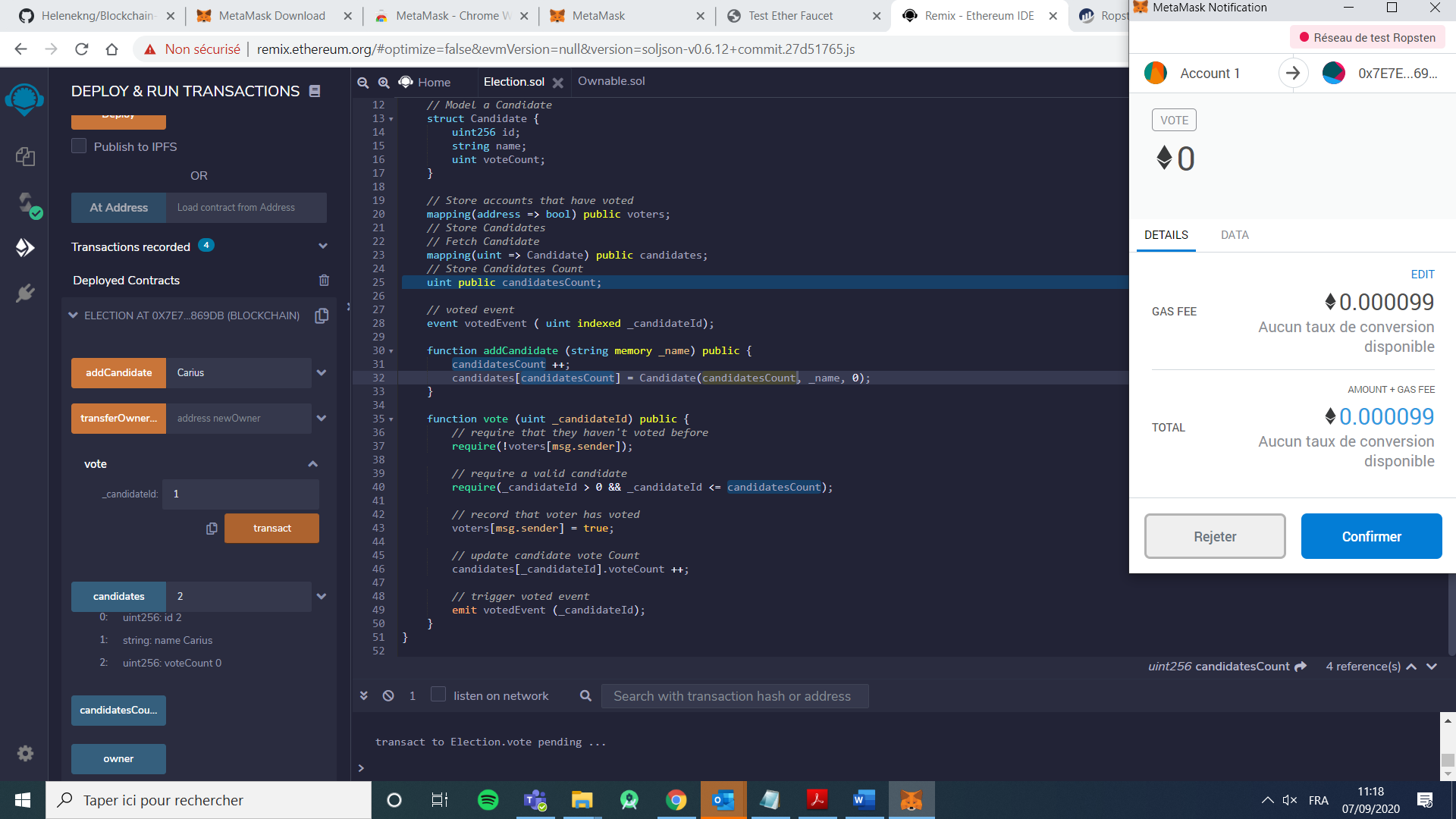


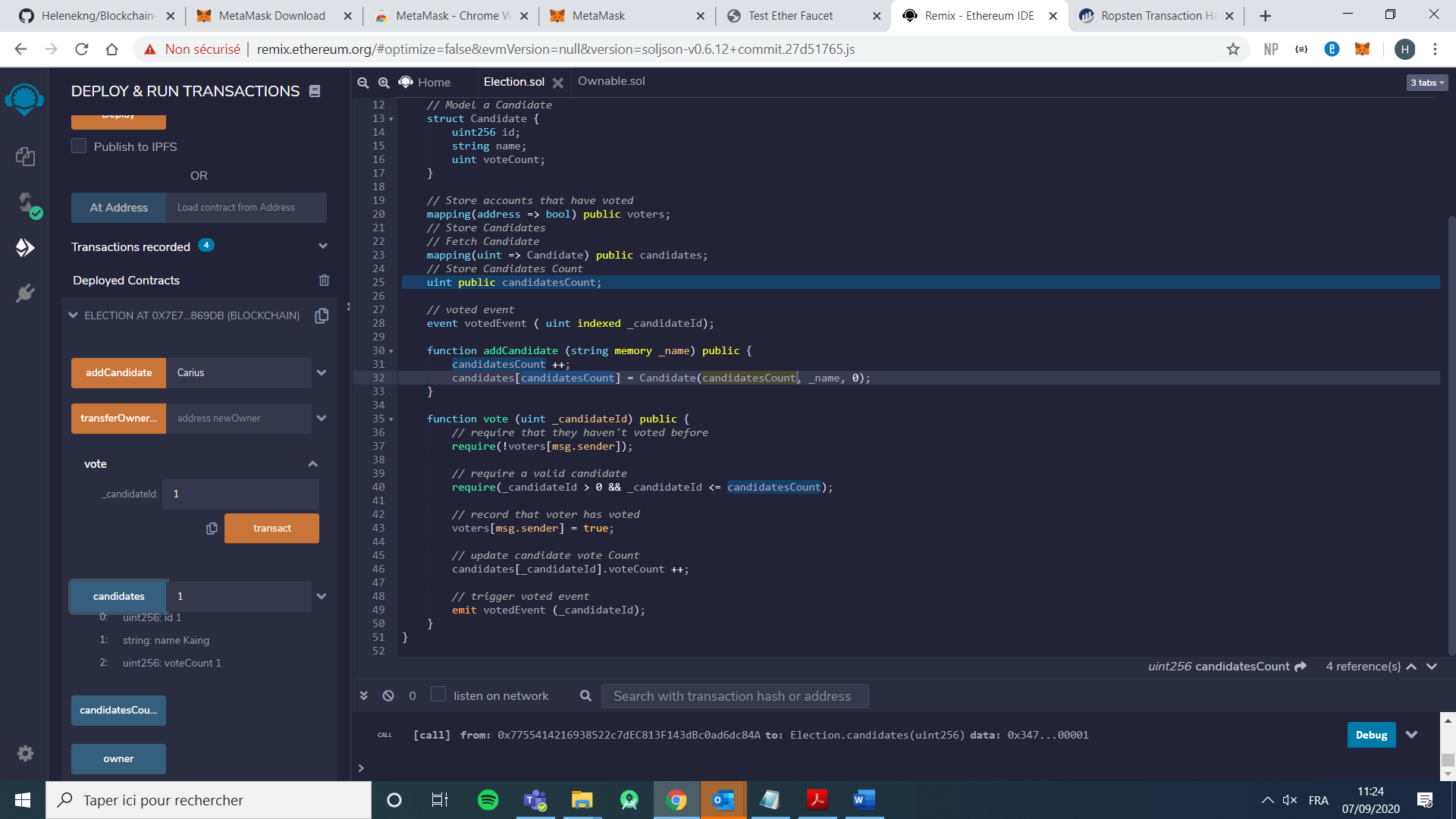
q. Ajout 2ème candidat :



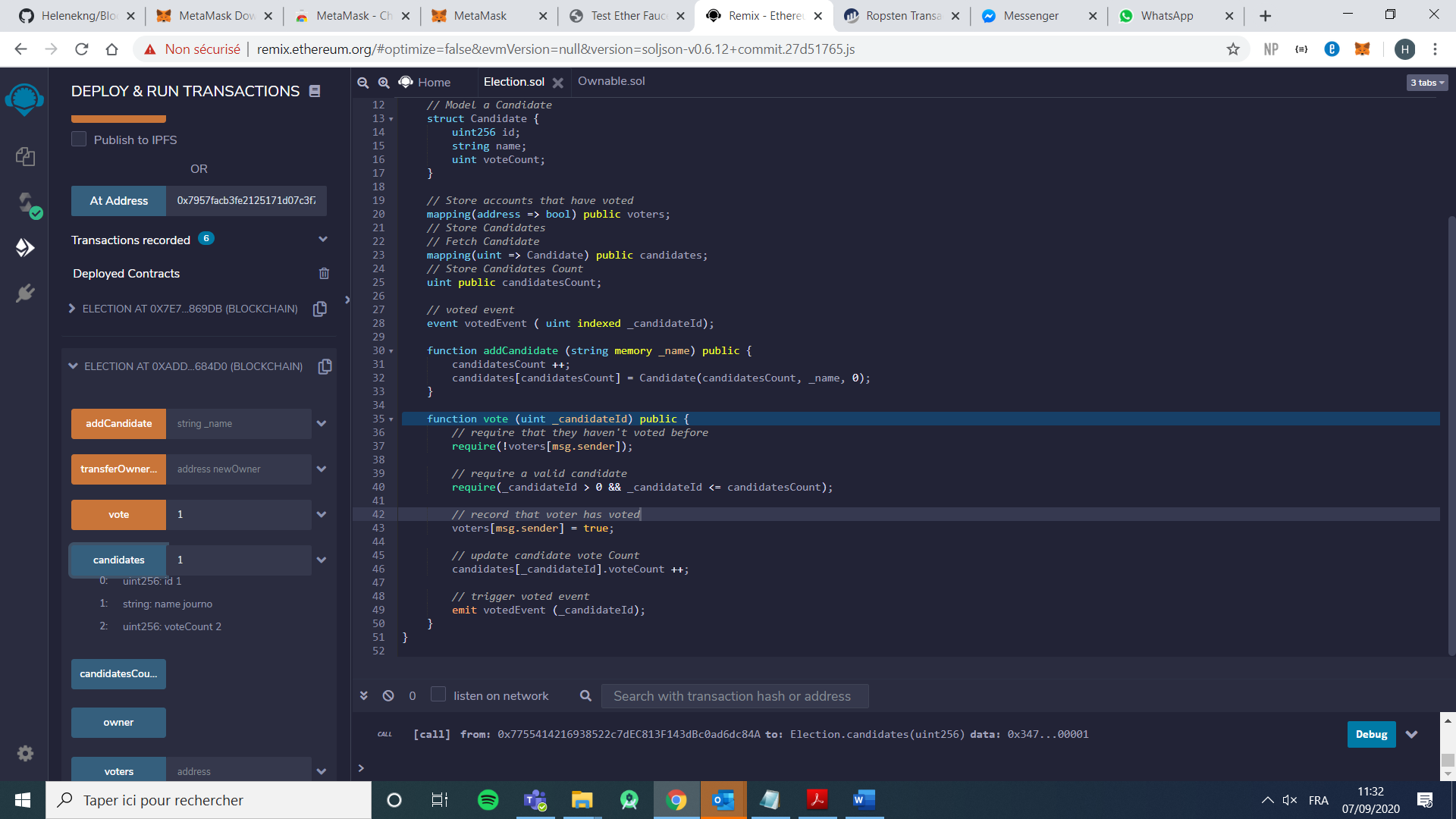
r. CandidateID : 2 

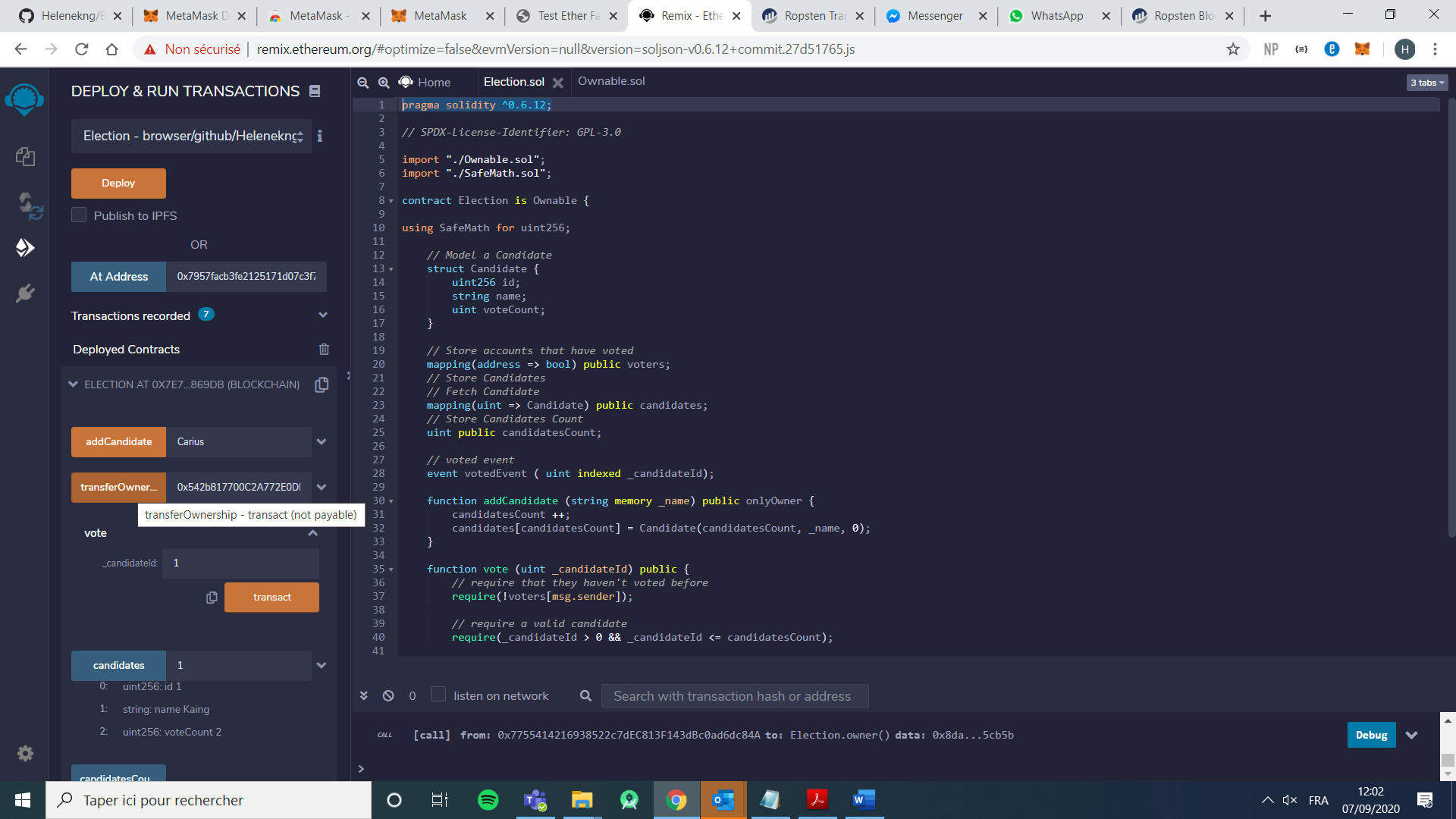
s. Adresse propriétaire contrat = 0x7755414216938522c7dec813f143dbc0ad6dc84a (moi)

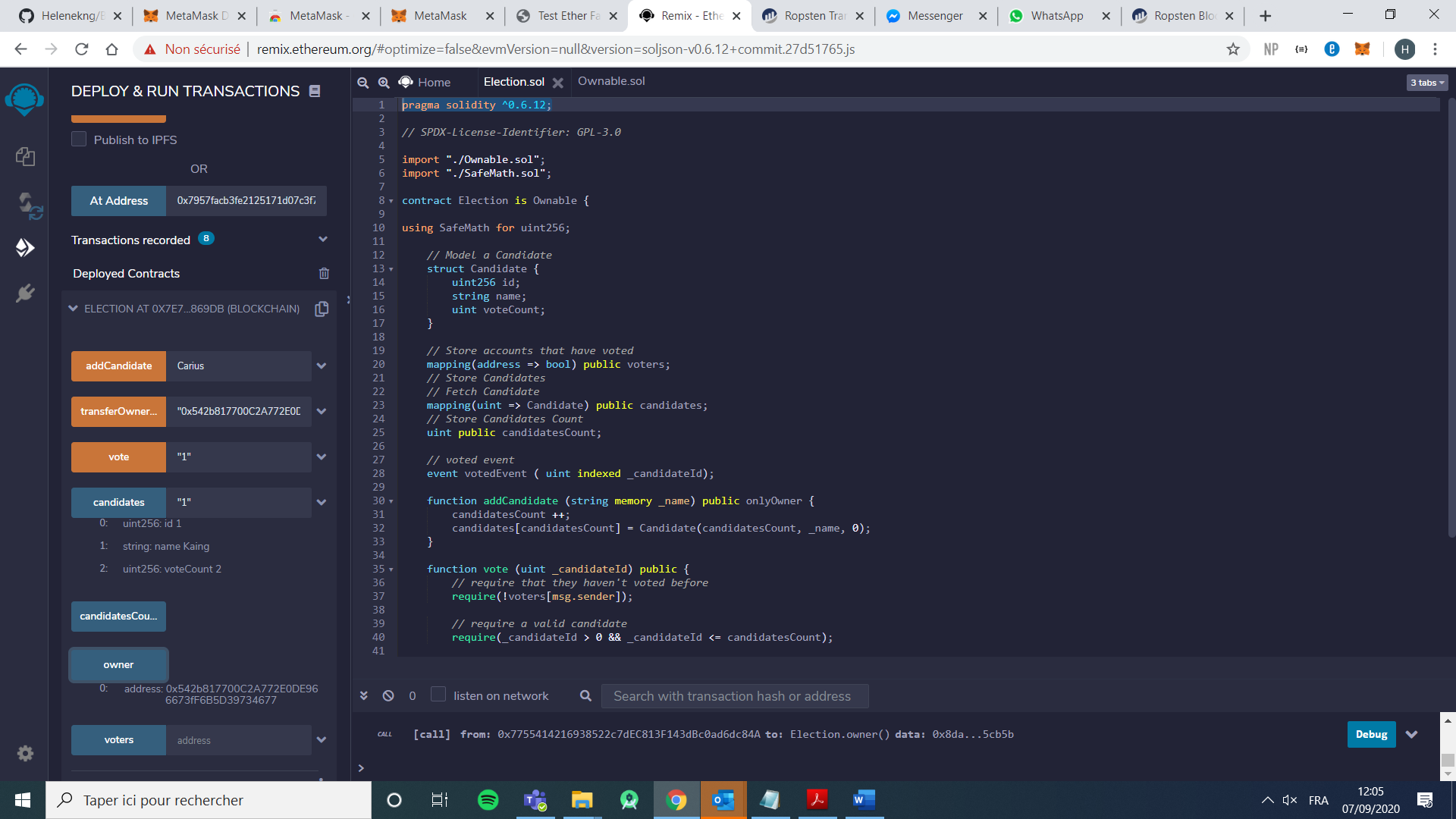
t. Vote : 



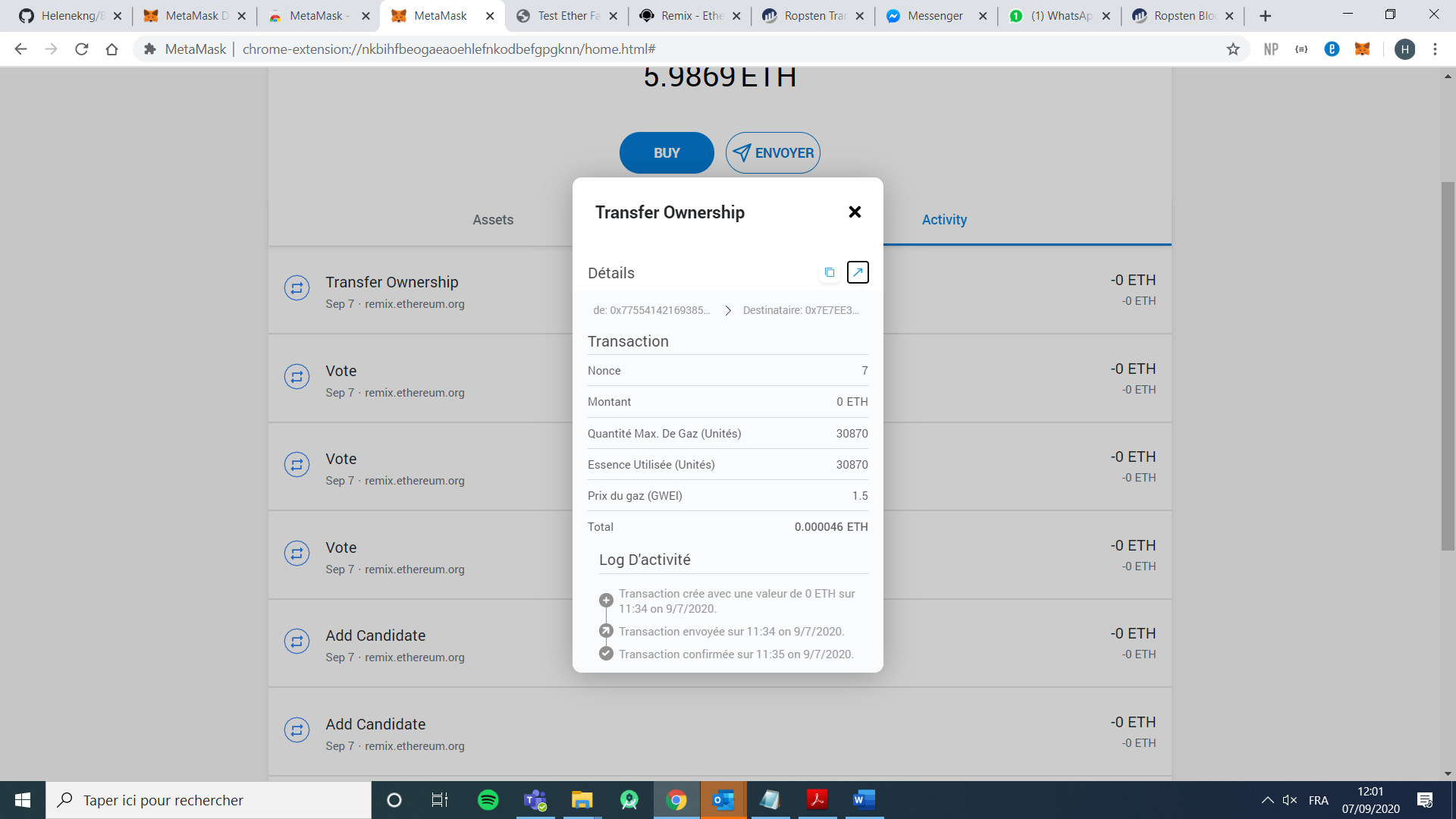
v. Vote sur le contrat d’un camarade :



w. Transfert de propriété à 0x542b817700C2A772E0DE966673fF6B5D39734677 :

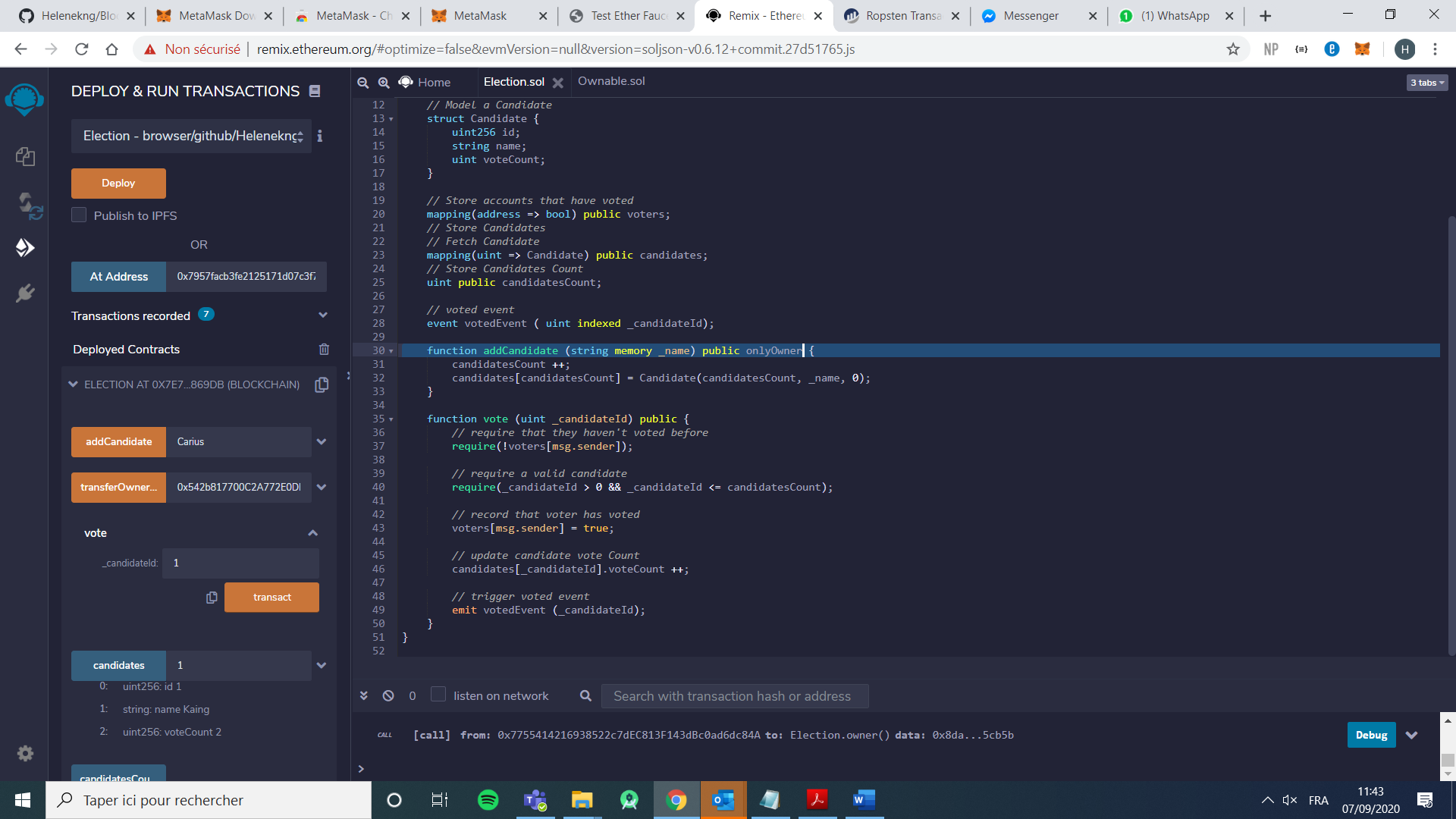


**Adresse publique du nouveau propriétaire**



x . De la même manière que la fonction tranferOwnership(), on doit faire appel au modifier onlyOwner, présent Ownable.sol, dans la déclaration de la fonction.

y. Modification du code pour sécuriser la fonction addCandidate :



On recompile le contrat modifié, et on redéploie :

