

# COMPOSANT 5 : SIGNATURE

Version 1.0

Groupe E :

HENRI AYCARD, AMINE BENNANI, THEOPHILE DANO, DO THU HANG, THIBAUD MOUTSITA-GOULO

## DESCRIPTION

- **Contexte**

Ce document intervient dans le projet de Programmation par composants de l'Université Paris-Dauphine, encadré par José Luu [[jose.luu@dauphine.psl.eu](mailto:jose.luu@dauphine.psl.eu)] et réalisé par les élèves du Master 2 IF-App.

La blockchain est une technologie assez récente de stockage et de transmission d'informations. Cette technologie offre de la transparence et de la sécurité car elle fonctionne sans organe central de contrôle. Plus concrètement, la blockchain permet à ses utilisateurs de partager des données sans intermédiaire. Elle a surtout été développée à partir de 2008 avec l'apparition du Bitcoin et d'autres cryptomonnaies.

Ce projet consiste à créer une blockchain contenant un ensemble de transaction et permettant d'en ajouter de nouvelles. Dans ce projet différents composants interagissent entre eux. Ce document explique les spécifications concernant le composant 5 : "La signature".

Les objectifs du composant sont :

- Recevoir un TXI et créer une clé privée et publique
- Signer une donnée avec une clé privée
- valider une signature en utilisant la clé publique correspondante

- **Schéma bloc incluant les composants connexes**

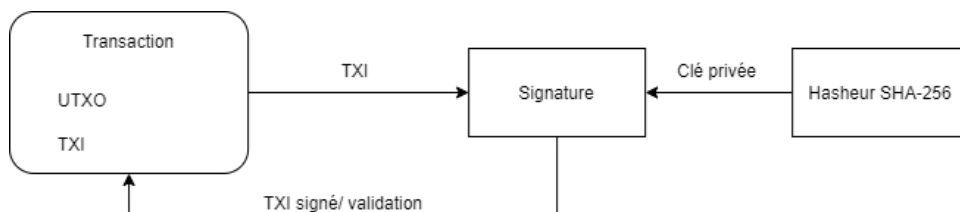
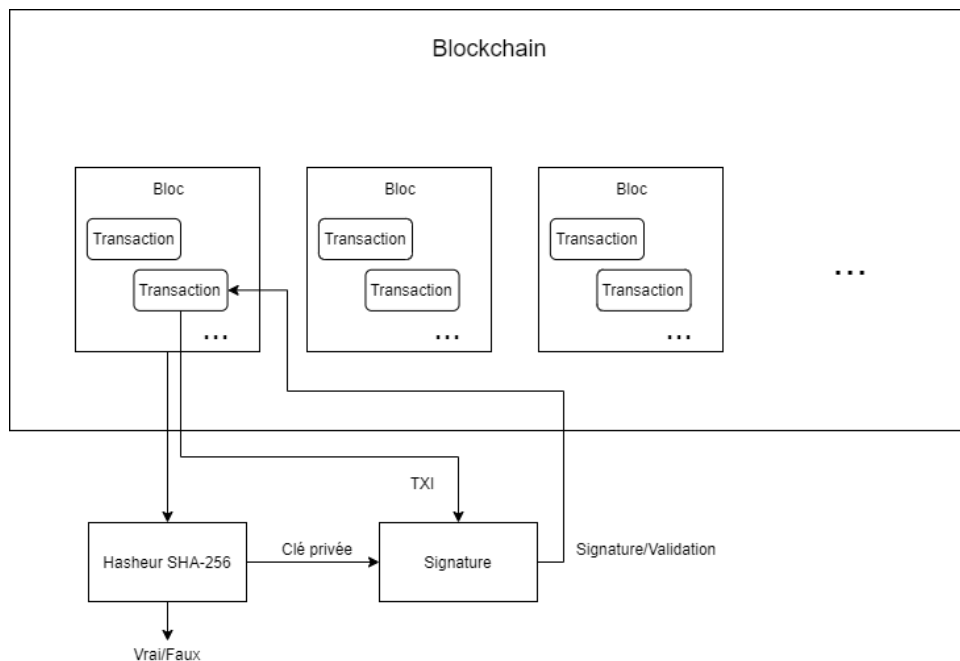


Schéma plus global :



- **Interface et interaction avec chaque autre composant**

Pareil que schéma bloc mais avec noms de fonctions ?

- **Résumé: déclarations de fonctions python d'interface et leurs arguments**

#### **Signature signerDonnee(String donnee, String cle\_privee)**

Cette fonction prend en paramètre une chaîne de caractère correspond à la transaction à signer et une clé privée.

Cette fonction retourne un objet de type Signature qui correspond à la signature de la transaction passée en paramètre.

#### **Bool validerDonnee(String donnee, String cle\_public, String signature)**

Cette fonction prend en paramètre trois chaînes de caractère correspondant à la donnée transmise, une clé publique et la signature pour authentifier

Cette fonction retourne un booléen correspond à la validation ou non de la signature.

- **Cas d'erreurs (NB: on utilisera une exception python)**

Conditions	Retour valeurs d'erreurs
Un des paramètres est nul	Null Value
Moins de paramètre attendu	Not Enough Argument
Plus de paramètre attendu	Too Much Argument
Mauvais type d'argument attendu	Invalid Type Argument

## TEST

Afin de vérifier le bon fonctionnement de notre composant nous avons écrit un code de test (cf. Test.cpp).

Ainsi ce code permet de tester les méthodes :

<Signature> signerDonnee(String donnee, String cle\_privée)

<Bool> validerDonnee(String donnee, String cle\_public, string signature)

Pour chacune des méthodes nous allons tester le bon fonctionnement (le retour de la valeur dans le cas normal) et le retour d'exception dans le cas d'erreurs.

Pour le cas normal :

La méthode validerDonnee() devra retourner TRUE si la signature est valide et FALSE sinon.

La méthode signerDonnee() devra retourner une signature issue du message haché récupérer du composant 4 et de la clé privée.

Pour le cas d'exception :

- Si l'un des paramètres est des fonctions est nul alors l'exception "Null Value" est levée.
- Si la fonction est appelée sans ou avec un nombre insuffisant de paramètres, alors le message d'erreur "Not Enough Argument" est retourné.
- Si la fonction est appelée avec plus de paramètre qu'elle requiert, alors l'exception "Too much argument" est levée.
- Si la fonction est appelée avec un type ne correspondant pas à celui avec lequel elle a été déclarée, alors le message d'erreur "Invalid Type Argument" est retourné.