



Ministère de l'enseignement supérieur de Tunisie
Ecole Nationale d'ingénieurs de Sfax



Projet Blockchain

Prédiction et Prévention du diabète IA - Blockchain

Elaboré par :
RACHEH Chayma
BEN ALI Houda
BEN FATMA Mahmoud



Encadré par :
M.FRIKHA Tarek

18/01/2025

PLAN

Introduction

Contexte Generale

Notions Théoriques

Architecture de projet

Réalisation

Conclusion & perspectives



Introduction



Introduction



La Surveillance efficace et en temps réel du diabète, en combinant des technologies avancées telles que l'intelligence artificielle et la blockchain

- Détection et identification rapide de la maladie grâce à l'analyse de données médicales (tests, signes cliniques).
- Notification des parties prenantes concernées, comme les médecins et les patients, pour une prise en charge rapide.
- Garantie de traçabilité et de sécurité des données collectées via l'utilisation de la blockchain, assurant l'intégrité et la confidentialité des informations de santé.



Contexte Général



Contexte General



**Le patient saisit ses données médicales.
Notifications via une application Flutter (web et
mobile) pour une action rapide.**

**Détection précise des patients diabétiques grâce à l'IA,
évitant les confusions avec d'autres non diabétiques.**

- Sécurisation des données dans une blockchain pour garantir traçabilité et intégrité.**
- Le patient et le médecin peuvent consulter l'historique sécurisé des bilans.**

Notions théoriques



Notions théoriques



définition de la blockchain



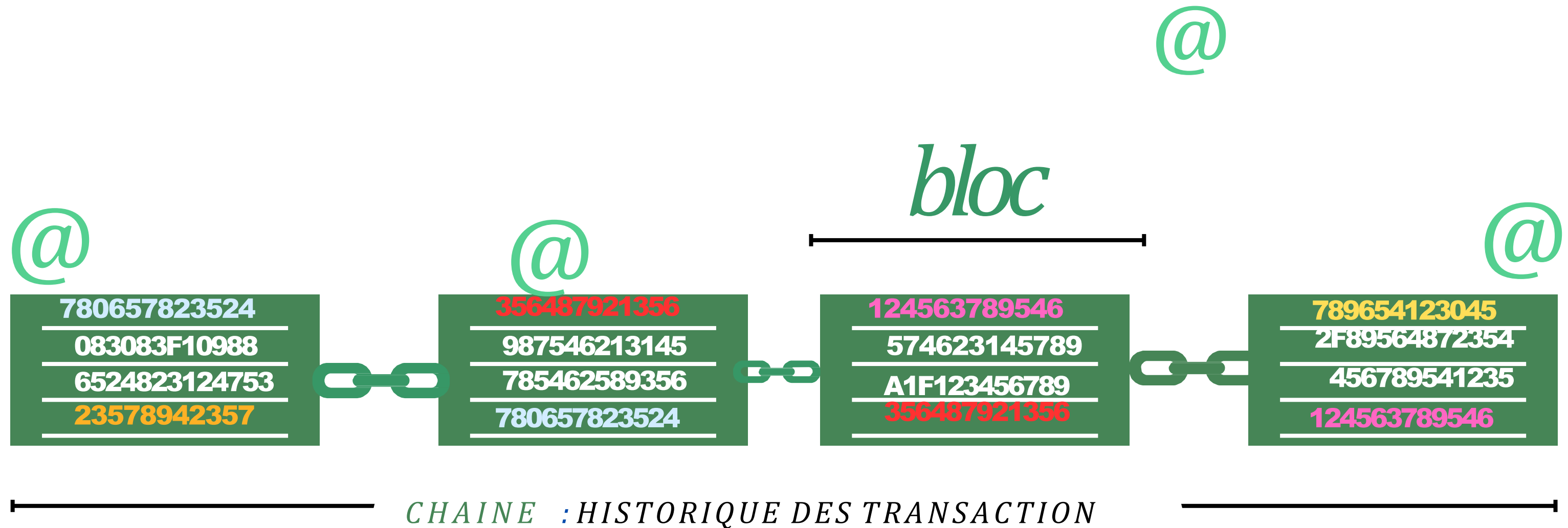
Stockage Décentralisé

+

Transmission d'information

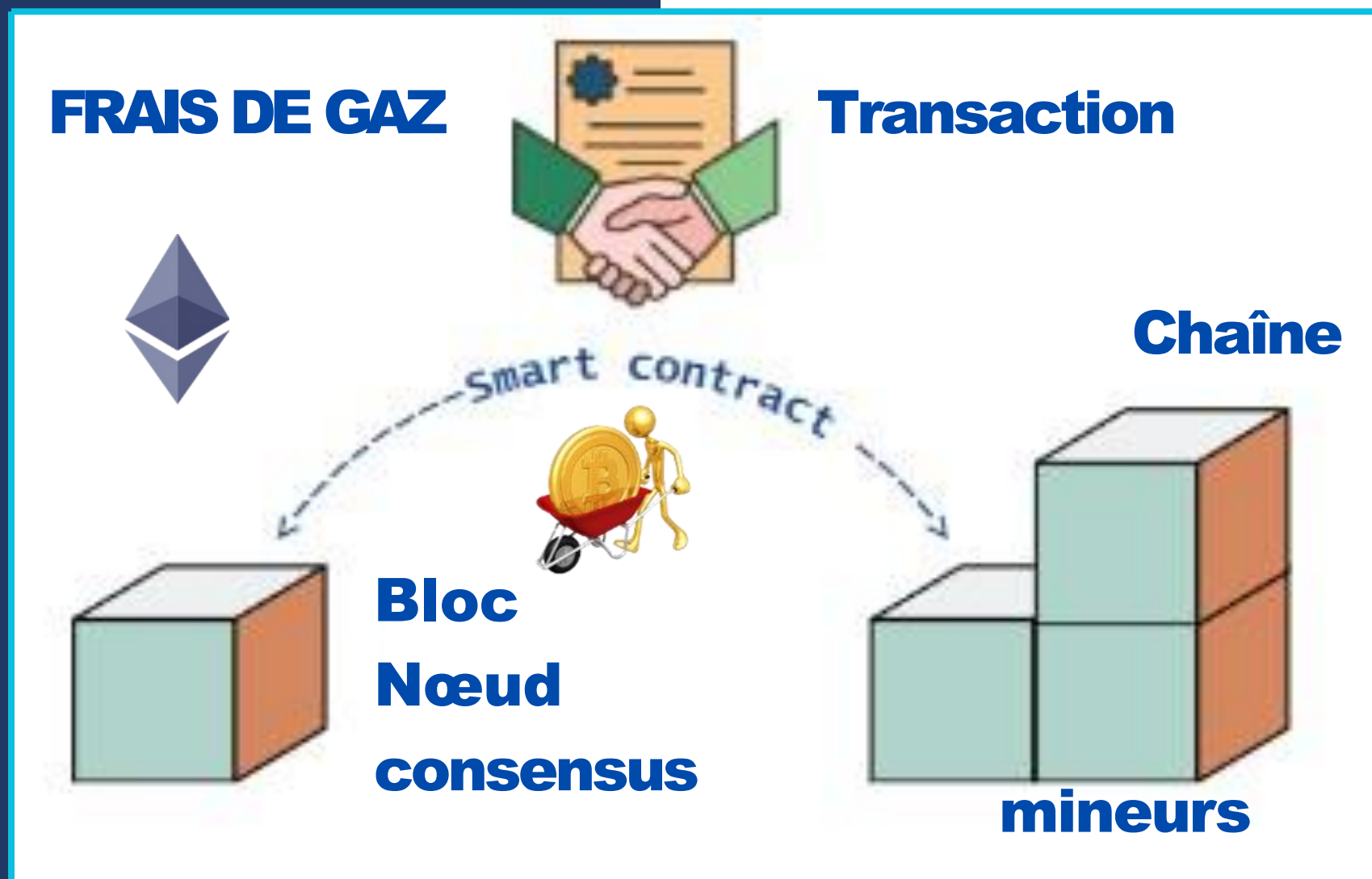


2008



Notions Theoriques

Architecture de la blockchain



- • • Calcul de frais de gaz (puissance de transaction)
- • • **Bitcoin** : Fonctionnement sur le principe de preuve de travail (**POW**).
- • • Exécution de nos smart contracts
- • • Application décentralisée : Basée sur les smart contracts.
- • • Demande d'exécution via Ethereum .

Notions theoriques

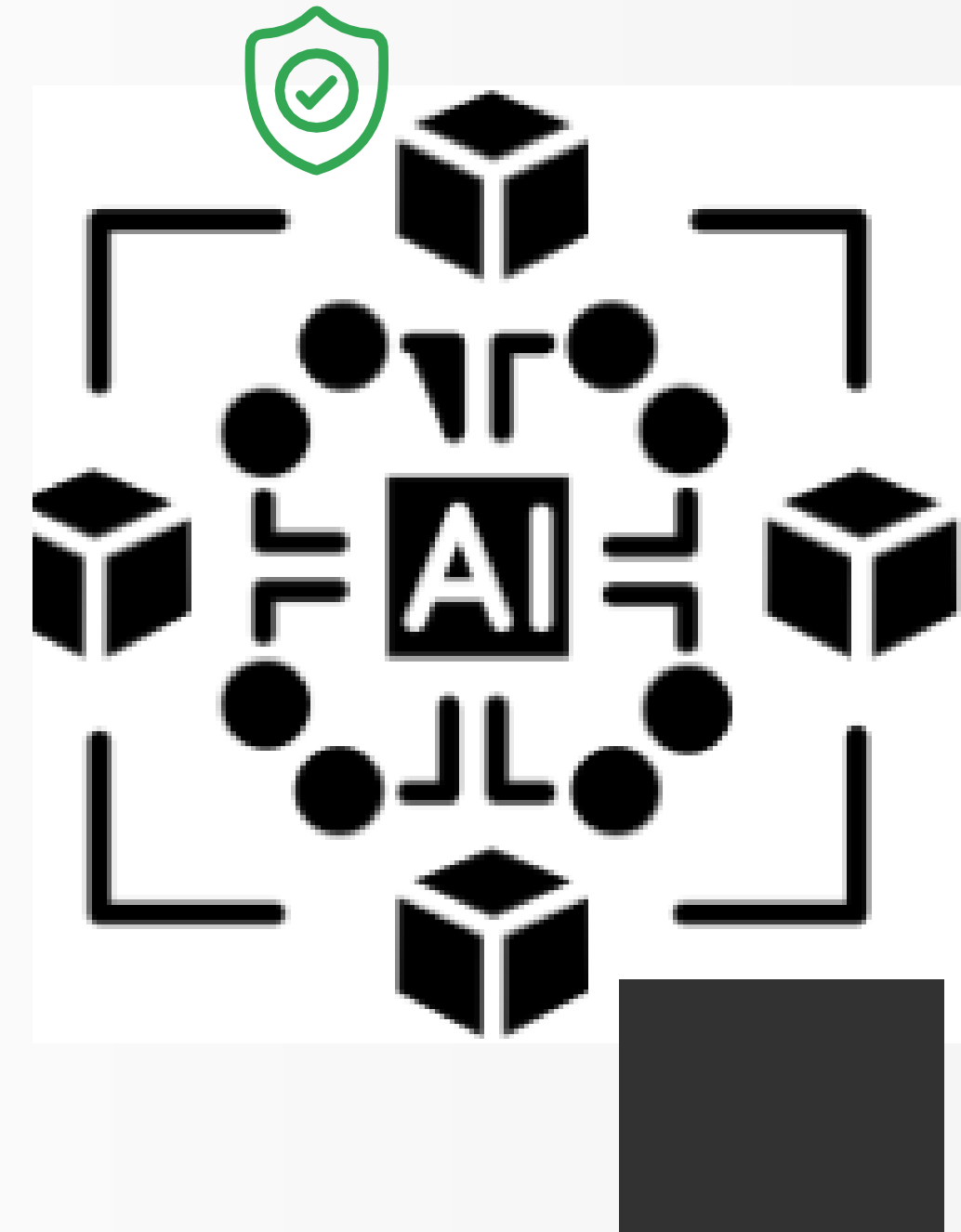
IA et Blockchain : Importance et Intégration

Pourquoi l'IA est essentielle ?

- **Automatisation** des tâches complexes.
- **Analyse des données** pour des décisions optimales.
- **Efficacité et personnalisation** dans divers secteurs (santé, finance, industrie).

Intérêt de l'intégration avec la Blockchain ?

- **Sécurité renforcée** : stockage immuable des modèles IA.
- **Efficacité** : Données sécurisées pour entraîner les IA, smart contracts autonomes.
- **Transparence** : Explication des décisions et renforcement de la confiance.



Réalisation



1/Modèle IA pour la détection du diabète 🧠

- Ce modèle prédit si le patient est **diabétique ou non** en fonction des paramètres d'entrée (glycémie, IMC, âge, etc.).

1 Collecte et préparation des données 📊

2 Choix du modèle d'intelligence artificielle 🧠

On a Testé plusieurs algorithmes (RandomForest, xgboost, Decision Tree, SVM)

meilleur résultat => xgboost

```
0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 0 1 1 1  
1 2 1 1 1 1 1 1 1 1 1 2 0]  
accuracy: 0.996969696969697
```

3 Entraînement et validation du modèle

4 Déploiement du modèle IA

```
[ ] import pickle  
  
# Sérialisation et sauvegarde du modèle dans un fichier  
with open("model_diabetic_final.pkl", "wb") as file:  
    pickle.dump(clfxgboost, file)
```

Réalisation

Choix technologiques & Justification

Ganache

- Environnement de test permettant de simuler une blockchain Ethereum en local.
- Facilite le développement et le débogage des contrats intelligents sans frais de transaction.
- Création de comptes Ethereum préfinancés pour les tests.
- Affichage des transactions en temps réel et exploration des blocs générés.



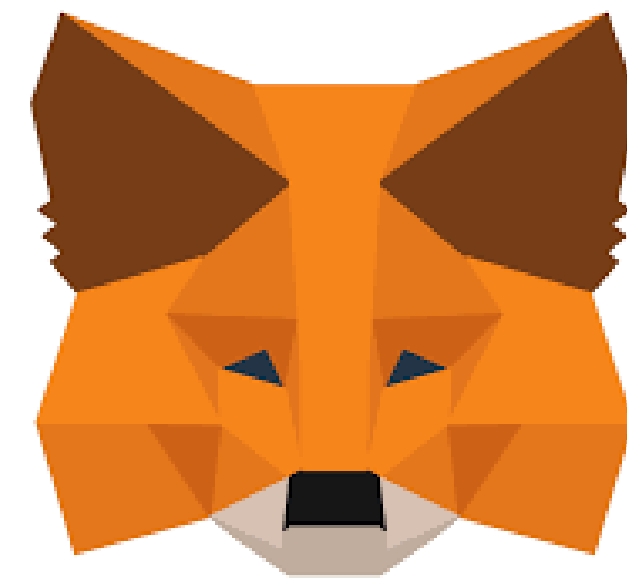
Réalisation

Choix technologiques & Justification



MetaMask local.

- Portefeuille numérique (wallet) permettant d'interagir avec la blockchain Ethereum.
- Connexion avec Ganache pour tester les transactions en environnement local avant un déploiement sur le réseau réel.
- Interaction avec les contrats intelligents directement depuis l'interface.

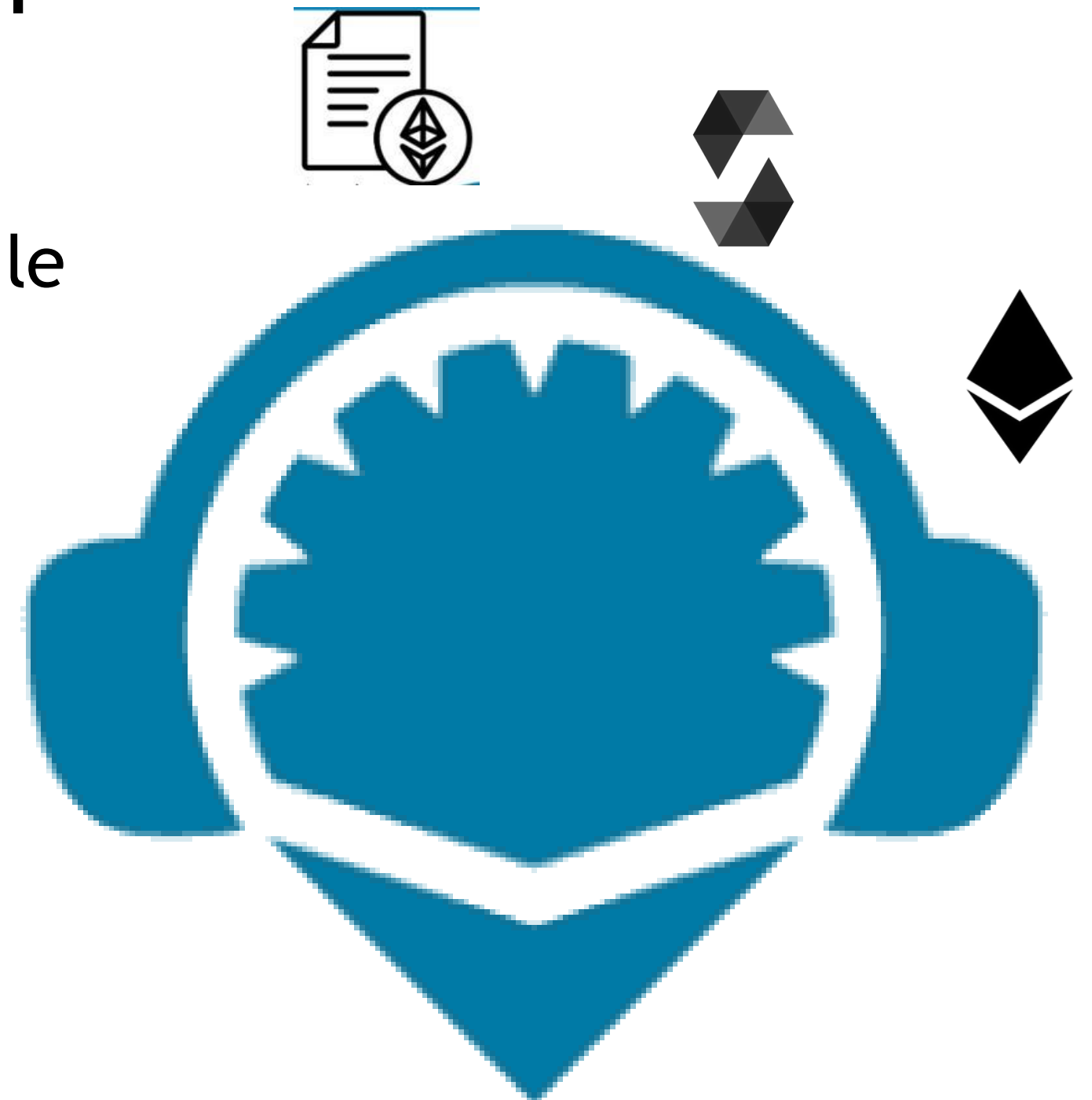


Réalisation

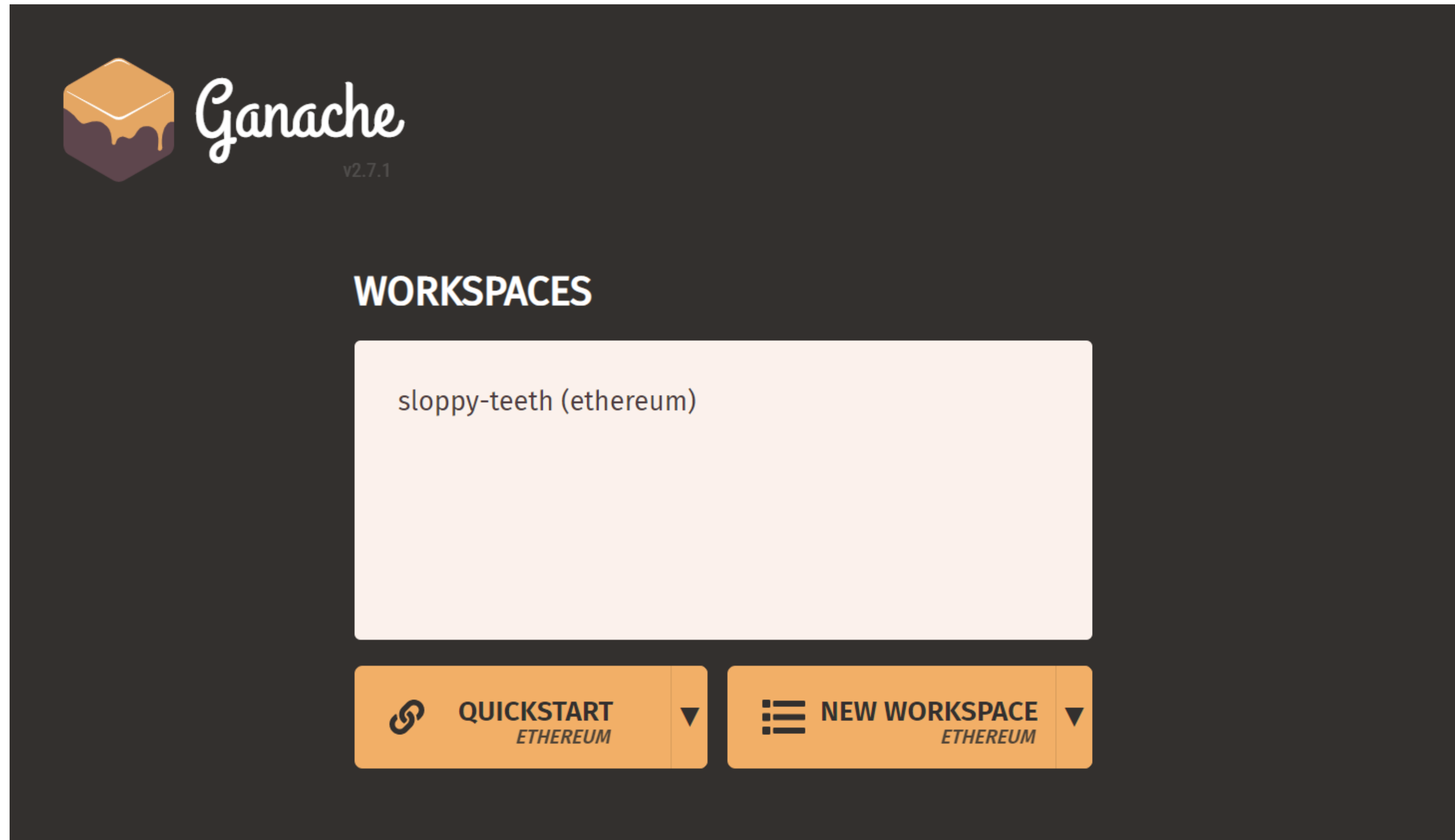
Choix technologiques & Justification

Remix

- Environnements de développement intégrés (IDE) pour le développement de contrats intelligents Ethereum.
- Éditeur de solidité.
- Compilation et déploiement de contrats intelligents sur Ethereum.
- Capacité de Remix à réaliser des tests intégrés.



2/ Lancement du Ganache pour la gestion de la blockchain locale



ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK0

GAS PRICE2000000000

GAS LIMIT6721975

HARDFORKMERGE

NETWORK ID5777

RPC SERVERHTTP://127.0.0.1:7545

MINING STATUSAUTOMINING

WORKSPACE QUICKSTART

SAVE

SWITCH

MNEMONIC ?

swim repeat brain tiny use short animal bachelor domain ketchup when response

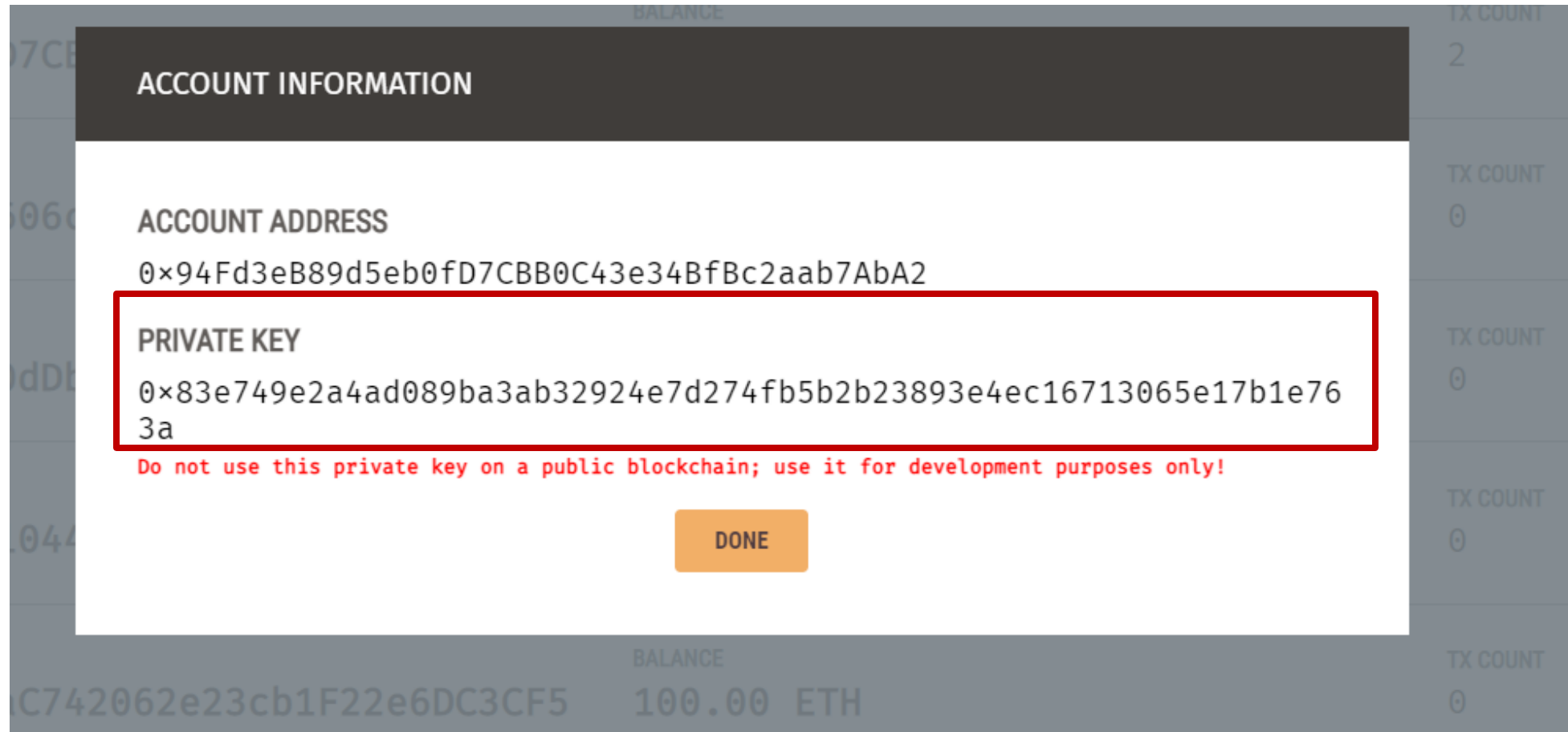
HD PATH

m44'60'0'0account_index

ADDRESS	BALANCE	TX COUNT	INDEX	
0x84c482ecC2cBFE49741c580542bacED331C8dE9a	100.00 ETH	0	0	
ADDRESS	BALANCE	TX COUNT	INDEX	
0xc0615214a29C812029Ccb68Ec16b5D411E7e307e	100.00 ETH	0	1	
ADDRESS	BALANCE	TX COUNT	INDEX	
0x041BD7DEe5F3fD1a429a99FB219a5F585901e9a5	100.00 ETH	0	2	
ADDRESS	BALANCE	TX COUNT	INDEX	
0xdEE700ACAd91B80252d99A0226fF4F7255d41E1A	100.00 ETH	0	3	
ADDRESS	BALANCE	TX COUNT	INDEX	
0xE8c5D26Ed95DF0f723A47EC34d34e069c09Aaa40	100.00 ETH	0	4	
ADDRESS	BALANCE	TX COUNT	INDEX	
0x02325867A7c3dd9412B2d70D03a4f24E4437150c	100.00 ETH	0	5	
ADDRESS	BALANCE	TX COUNT	INDEX	
0xd169798b504E22622a0399773434AFD5Af075233	100.00 ETH	0	6	

Comptes Ethereum préconfigurés : Chaque adresse dispose d'un solde de **100 ETH** pour tester les transactions sans coût réel.

3/ Connecter MetaMask à Ganache en important les clés privées des comptes test.



MetaMask connecté à Ganache.



4/ Création, Déploiement du Smart Contract et exécution avec confirmation Metamask dans Remix

The screenshot displays the Remix IDE interface during the deployment of a smart contract. The central editor shows the Solidity code for `IncidentContract`, which includes a struct for incident data, a mapping of incidents, a constructor, and a `recordIncident` function. The left sidebar, titled "DEPLOY & RUN TRANSACTIONS", shows the environment set to "Injected Provider - MetaMask" on a "Custom (5777) network". The account address is `0x94F...7AbA2` with a balance of 99.98721748 ETH. The gas limit is set to 3,000,000 and the value is 0 Wei. The contract to be deployed is `IncidentContract - blockchain.sol` using the "evm version: london". A "Deploy" button is visible. Below the deployment settings, it shows "Transactions recorded: 2" and "Deployed Contracts: 1". The bottom status bar indicates the transaction was successful: "[block:1 txIndex:-] from: 0x94f...7aba2 to: IncidentContract.(constructor) value: 0 wei data: 0x608...10033 logs: 0 hash: 0xfaa...34e68". The right sidebar, titled "SOLIDITY COMPILER", shows the compiler version as `0.8.17+commit.8df45f5f` and includes buttons for "Compile blockchain.sol", "Compile and Run script", "Run Remix Analysis", "Run SolidityScan", "Publish on IPFS", "Publish on Swarm", and "Compilation Details".

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.17;
3
4 contract IncidentContract {
5     struct Incident {
6         uint id;
7         uint No_Patient;
8         uint Gender;
9         uint AGE;
10        uint[10] medicalData; // Stocke les valeurs Urea, Cr, HbA1c, etc.
11    }
12
13    mapping(uint => Incident) private incidents;
14    uint[] private incidentIds;
15    address public owner;
16
17    event IncidentRecorded(uint id, uint No_Patient, uint Gender, uint AGE);
18
19    modifier onlyOwner() {
20        require(msg.sender == owner, "Access denied: Not contract owner");
21        _;
22    }
23
24    constructor() {
25        owner = msg.sender;
26    }
27
28    function recordIncident(
```

Transactions recorded 2 i >

Deployed Contracts 1

INCIDENTCONTRACT AT 0X8D

[block:1 txIndex:-] from: 0x94f...7aba2 to: IncidentContract.(constructor) value: 0 wei data: 0x608...10033 logs: 0 hash: 0xfaa...34e68

Debug

SOLIDITY COMPILER

COMPILER +

0.8.17+commit.8df45f5f

☐ Include nightly builds

☐ Auto compile

☐ Hide warnings

Advanced Configurations >

Compile blockchain.sol

Compile and Run script i

CONTRACT

IncidentContract (blockchain.sol)

Run Remix Analysis

Run SolidityScan

Publish on IPFS

Publish on Swarm

Compilation Details

ABI Bytecode

Ganache

ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK
2

GAS PRICE
20000000000

GAS LIMIT
6721975

HARDFORK
MERGE

NETWORK ID
5777

RPC SERVER
HTTP://127.0.0.1:7545

MINING STATUS
AUTOMINING

WORKSPACE
QUICKSTART

SAVE

SWITCH

TX HASH

0x125efb63bae9c3889f6f84c9808e01ca10f8c3561bea9e51d79aea5549b0df90

CONTRACT CREATION

FROM ADDRESS

0x94Fd3eB89d5eb0fD7CBB0C43e34BfBc2aab7AbA2

CREATED CONTRACT ADDRESS

0xCb5FDa2DF43bC0A9d6b59081c377B1A579d5ac2a

GAS USED

639126

VALUE

0

TX HASH

0xa93dfc299d249518e4fd6ed50b0aef7ec0d635d87e463a906c71712796442f65

CONTRACT CREATION

FROM ADDRESS

0x94Fd3eB89d5eb0fD7CBB0C43e34BfBc2aab7AbA2

CREATED CONTRACT ADDRESS

0x8d9664622b245F65E1Cc4d93e038f58c07a3AFb0

GAS USED

639126

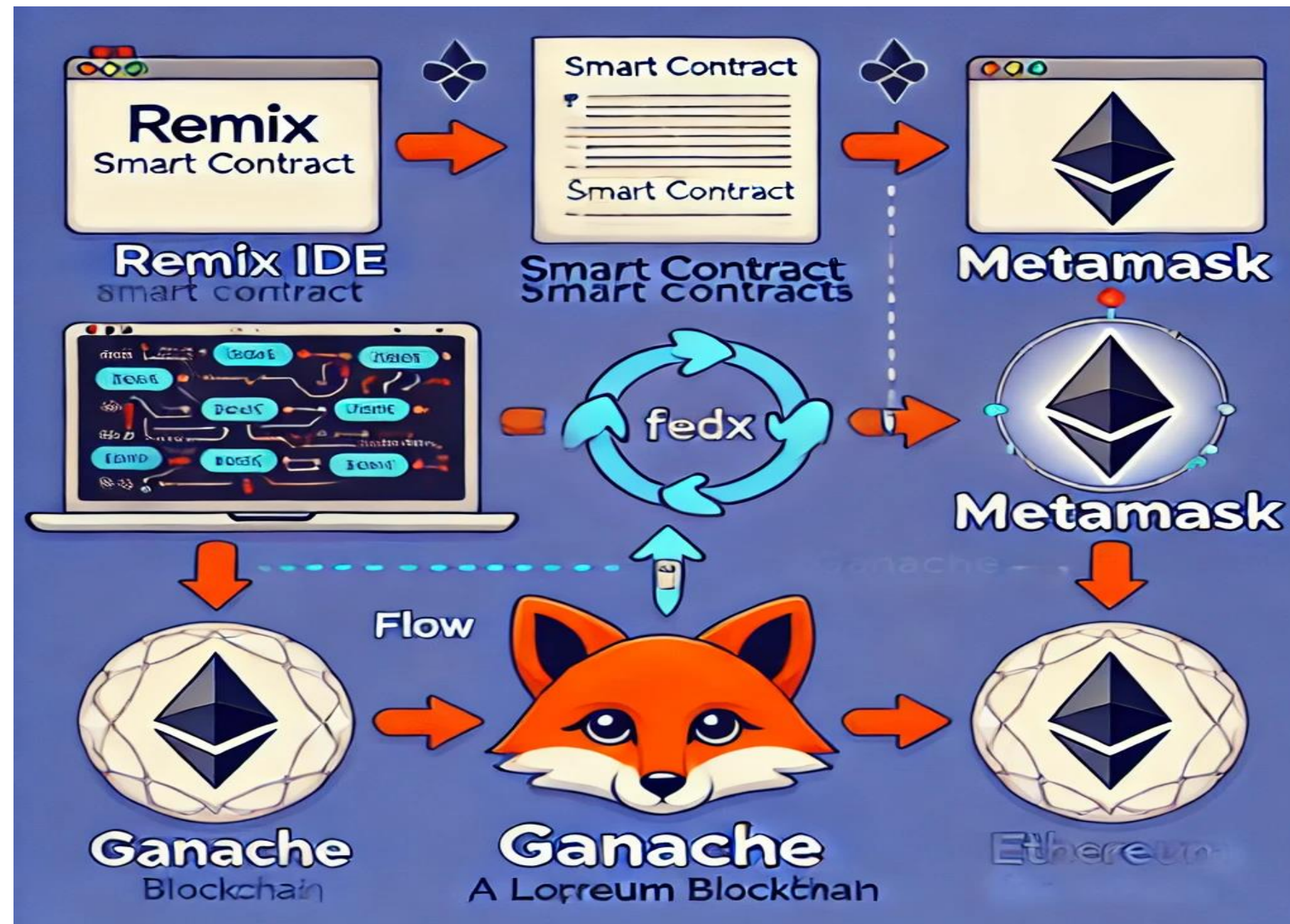
VALUE

0

=> le contrat est bien **déployé et actif** sur la blockchain simulée, prêt à être testé et utilisé.

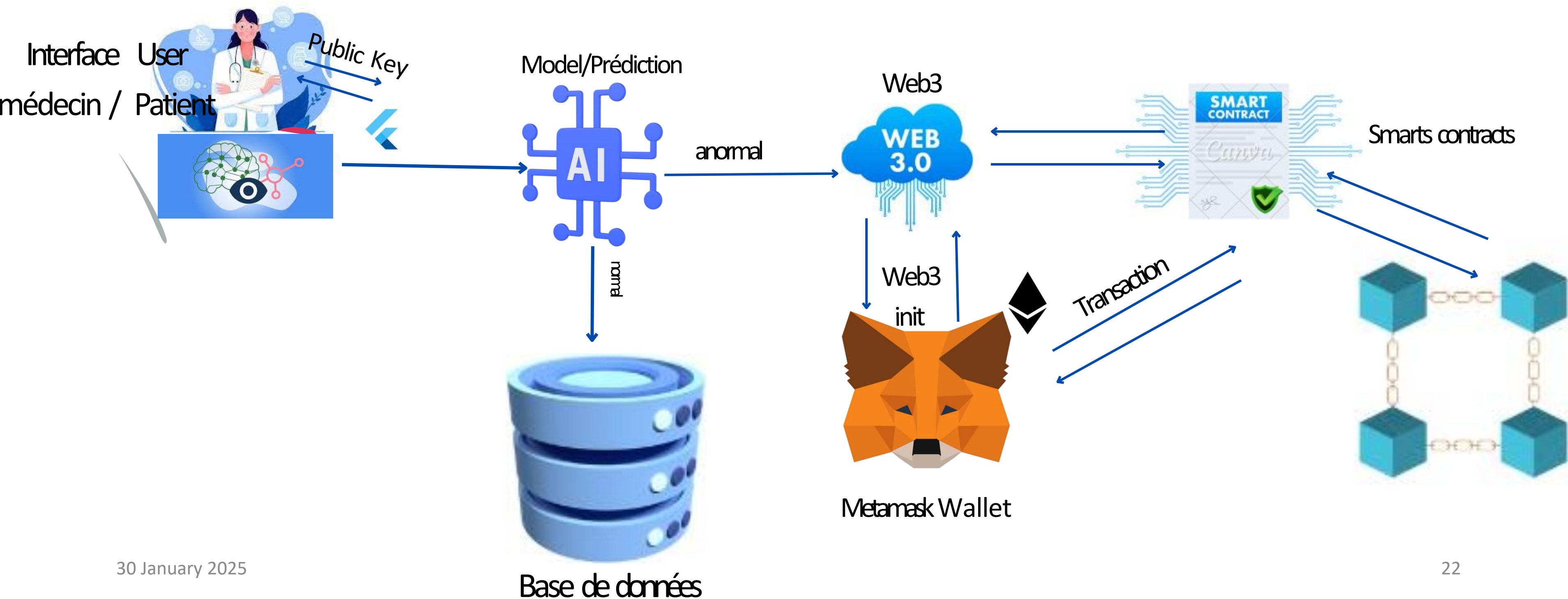
Réalisation

Les outils utilisés



Réalisation

ARCHITECTURE DE NOTRE APPLICATION



Conclusion



Conclusion

Ce projet représente une avancée significative dans l'amélioration du diagnostic et de la gestion du diabète grâce à la synergie de ces technologies modernes.

Une solution qui pourrait non seulement prédire le diabète de manière fiable, mais aussi renforcer la confiance des utilisateurs grâce à un système sécurisé et transparent.

merci de Votre Attention