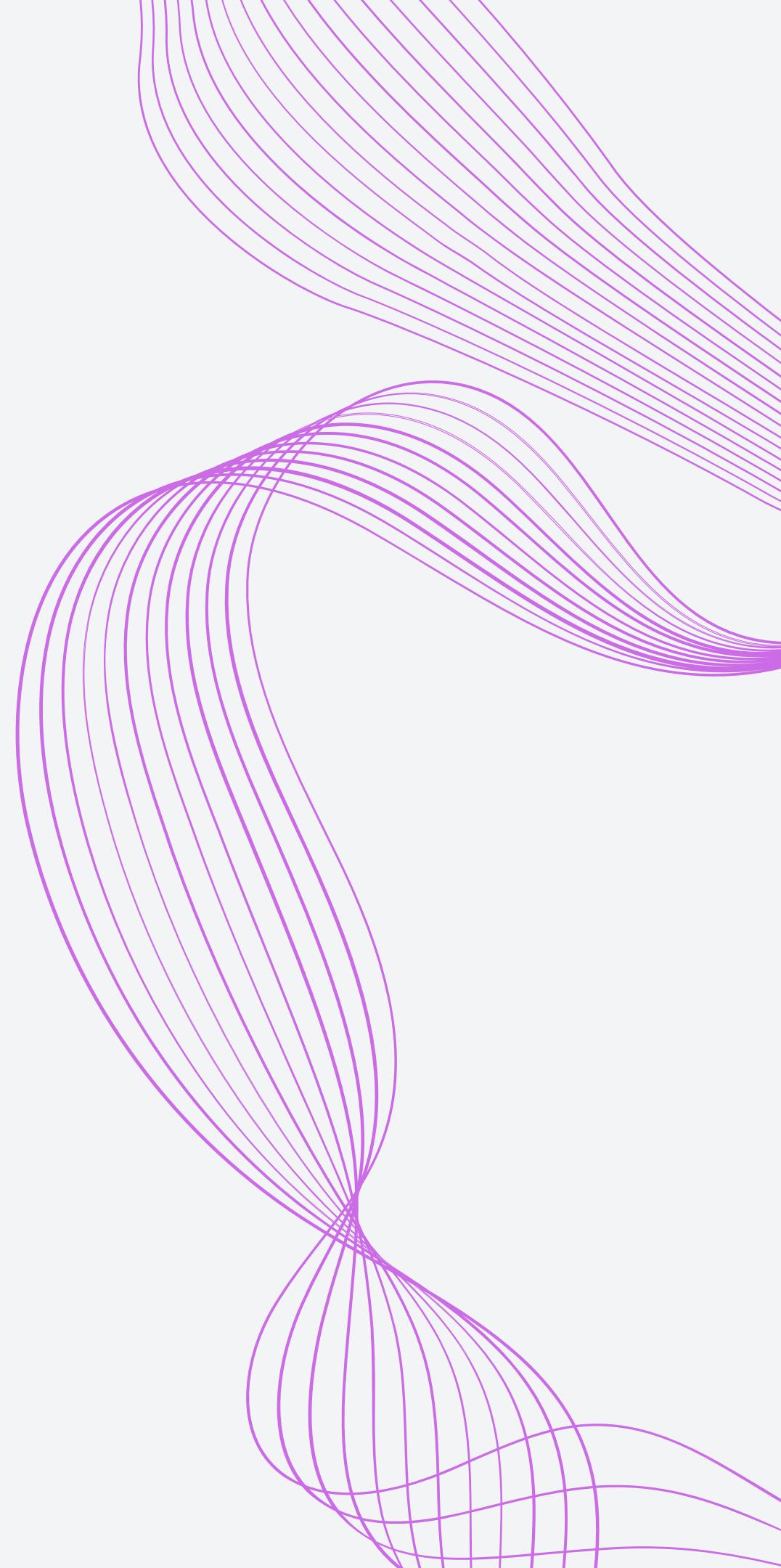


# **ENERGYCHAIN**

**Hostel 65**

# CONTENT

- 
- 
- 01** THE PROBLEMS WE AIM TO SOLVE
  - 02** OUR SOLUTION
  - 03** INTRODUCTION
  - 04** TECH STACKS
  - 05** SYSTEM ARCHITECTURE
  - 06** SET-UP INSTRUCTIONS
  - 07** DAPP FLOW SCREENSHOTS
  - 08** IOT INTEGRATION FLOW
  - 09** HOW WE USED ZK-SNARKS
  - 10** IMPORTANT LINKS RELATED TO THE PROJECT

# THE PROBLEMS WE AIM TO SOLVE

- Right now, the energy sector has **too many middlemen**, which makes trading electricity and carbon credits slow, expensive, and unfair.
- Small renewable energy producers (like someone with solar panels) **can't join** the market easily.
- Trading carbon credits (a way to offset pollution) isn't very **transparent or reliable**.
- Safety Issues.

# OUR SOLUTION

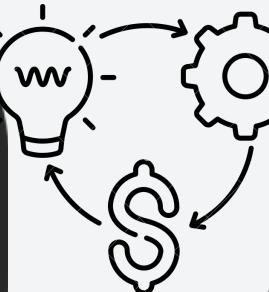
We are building a decentralized system (using blockchain) that:

- Directly connects users who have extra energy (like solar producers) with people who need energy.
- Tracks real-time energy use with smart devices (like electricity meters).
- Makes carbon credit trading fair and automated.
- Blockchain ensures an immutable ledger for verifying carbon credits and transactions, eliminating double spending.
- Ensures privacy for business-sensitive data using ZK-SNARKs (a privacy tech).

# INTRODUCTION

## PURPOSE

This project leverages blockchain technology integrated with real-time data processors such as IoT devices to create a platform for carbon credit and electricity trading. By utilizing privacy-preserving technologies like zk-SNARKs, the platform ensures transparent transactions while maintaining user privacy. The system facilitates secure, traceable, and privacy-preserving trading to promote sustainability and efficient energy markets.

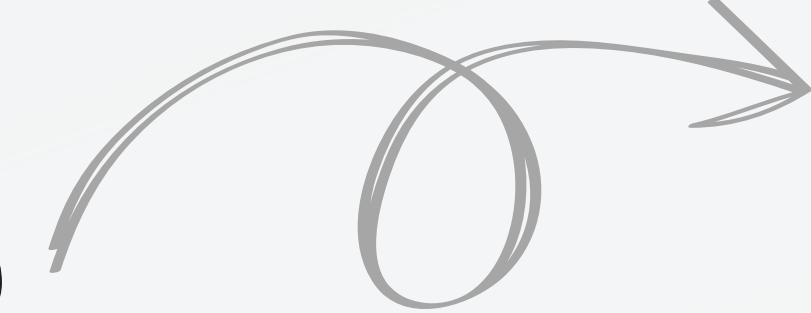


## MISSION

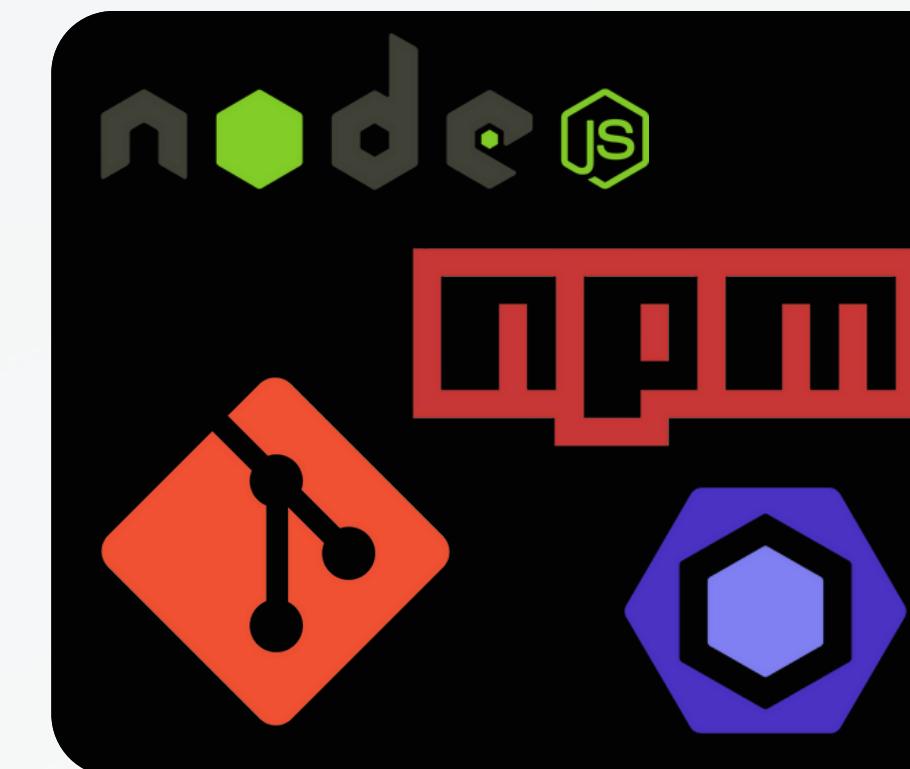
- **Facilitating Transparent and Secure Energy Trading:** Our platform leverages blockchain and smart contracts to enable trustless, tamper-proof, and automated carbon credit and electricity trading, ensuring fairness and reliability for all participants.
- **Ensuring Privacy and Data Security:** By integrating zk-SNARKs and decentralized storage, we protect sensitive transaction data, allowing users to trade with full confidentiality while maintaining the integrity of the system.
- **Promoting Sustainability and Energy Equity:** We empower both individuals and businesses to actively participate in carbon markets and renewable energy trading, fostering a greener, more inclusive, and efficient global energy ecosystem.



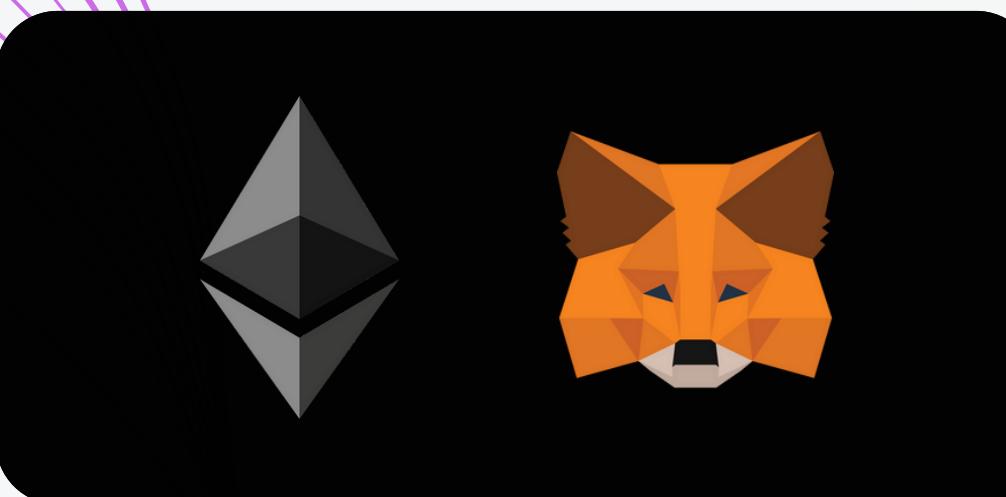
# TECH STACKS



FRONTEND



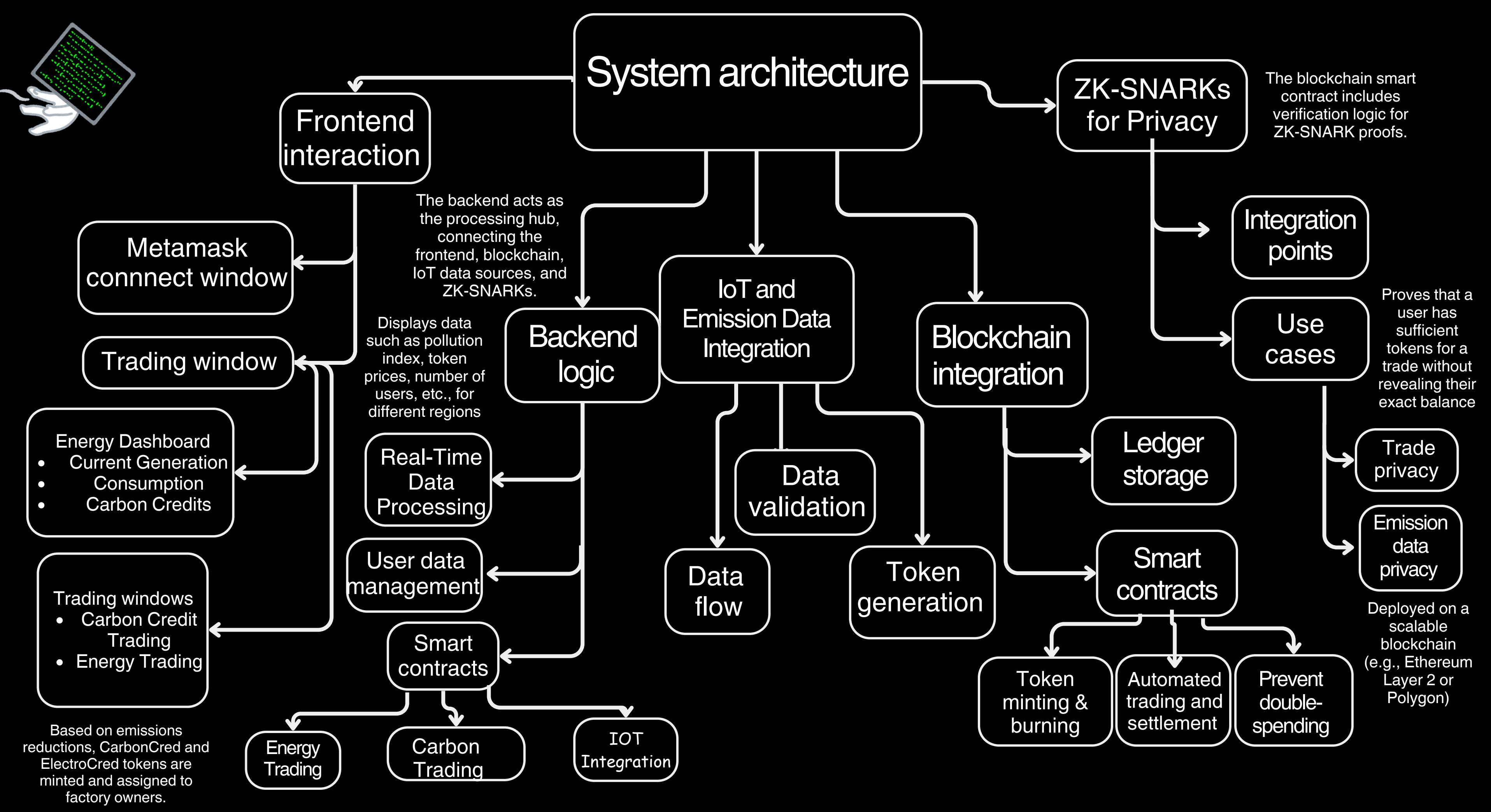
BACKEND (BLOCKCHAIN)



BLOCKCHAIN

DEVELOPMENT TOOLS

STORAGE



# SET-UP INSTRUCTIONS

## CLONE THE REPOSITORY

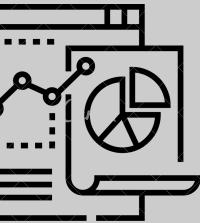
```
git clone https://github.com/musk1n/webss
```

## INSTALL DEPENDENCIES

Navigate to the project directory and install required modules:

```
cd webss
```

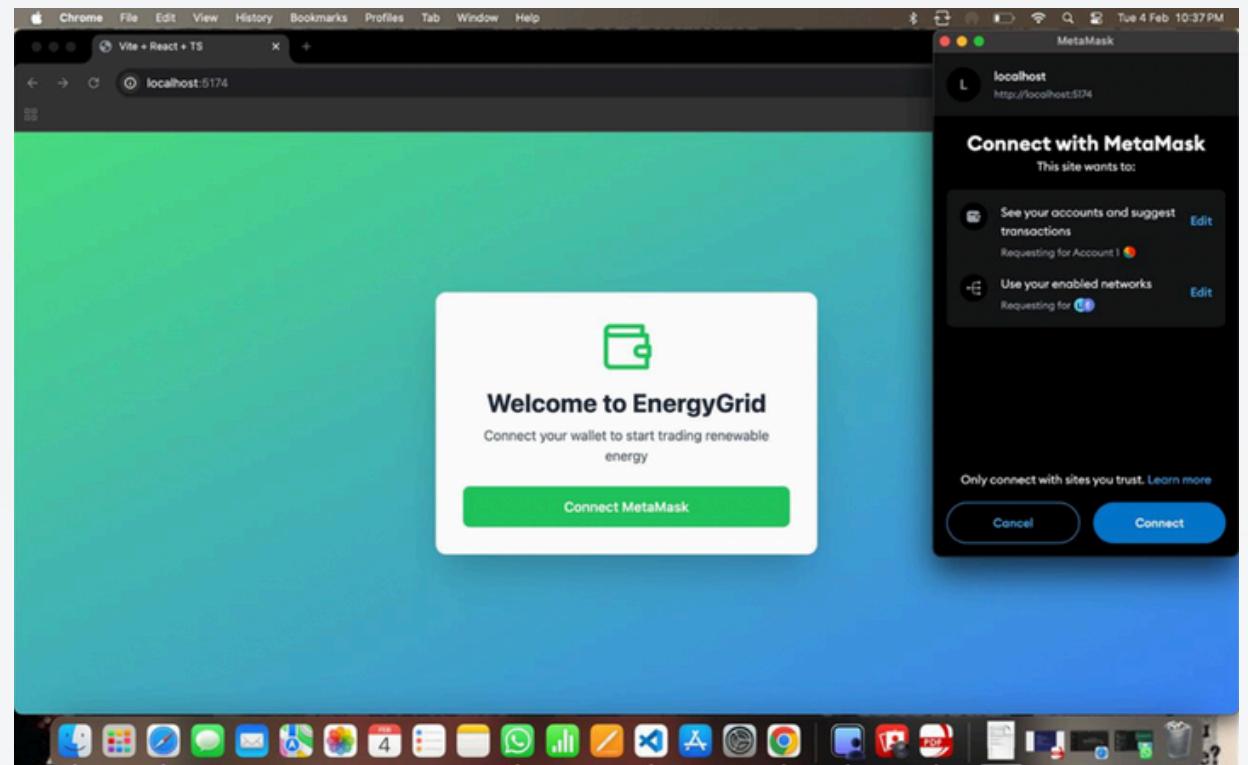
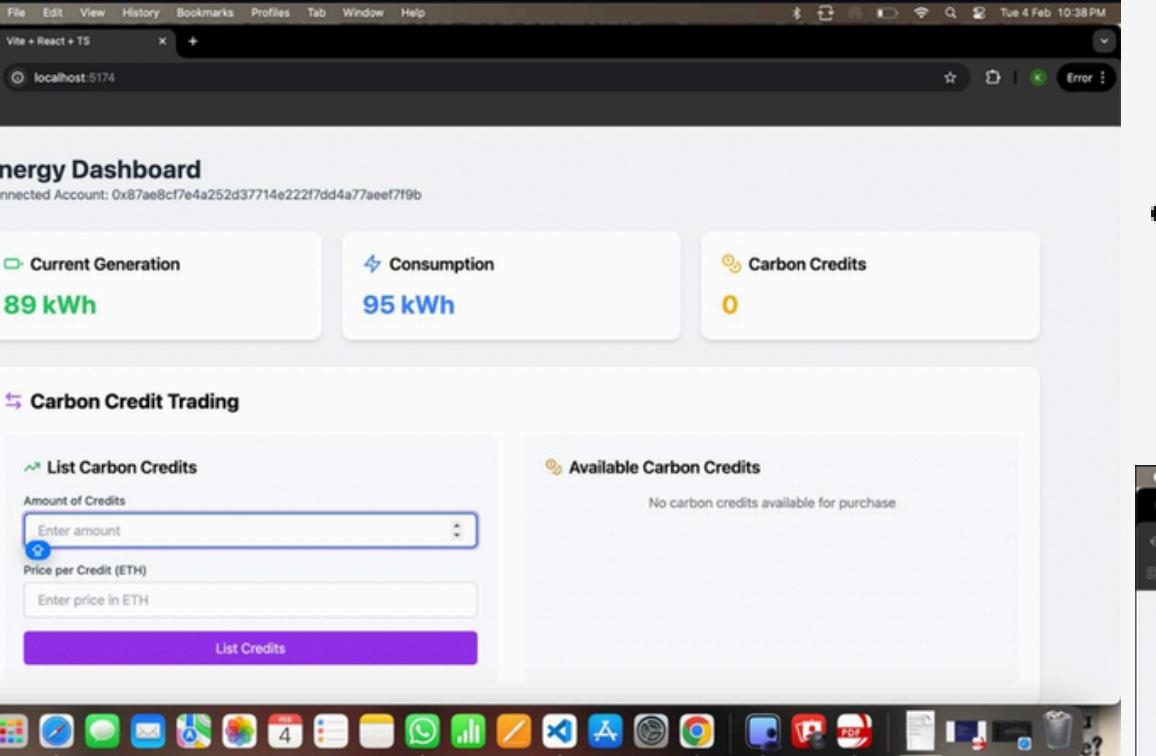
```
npm install
```



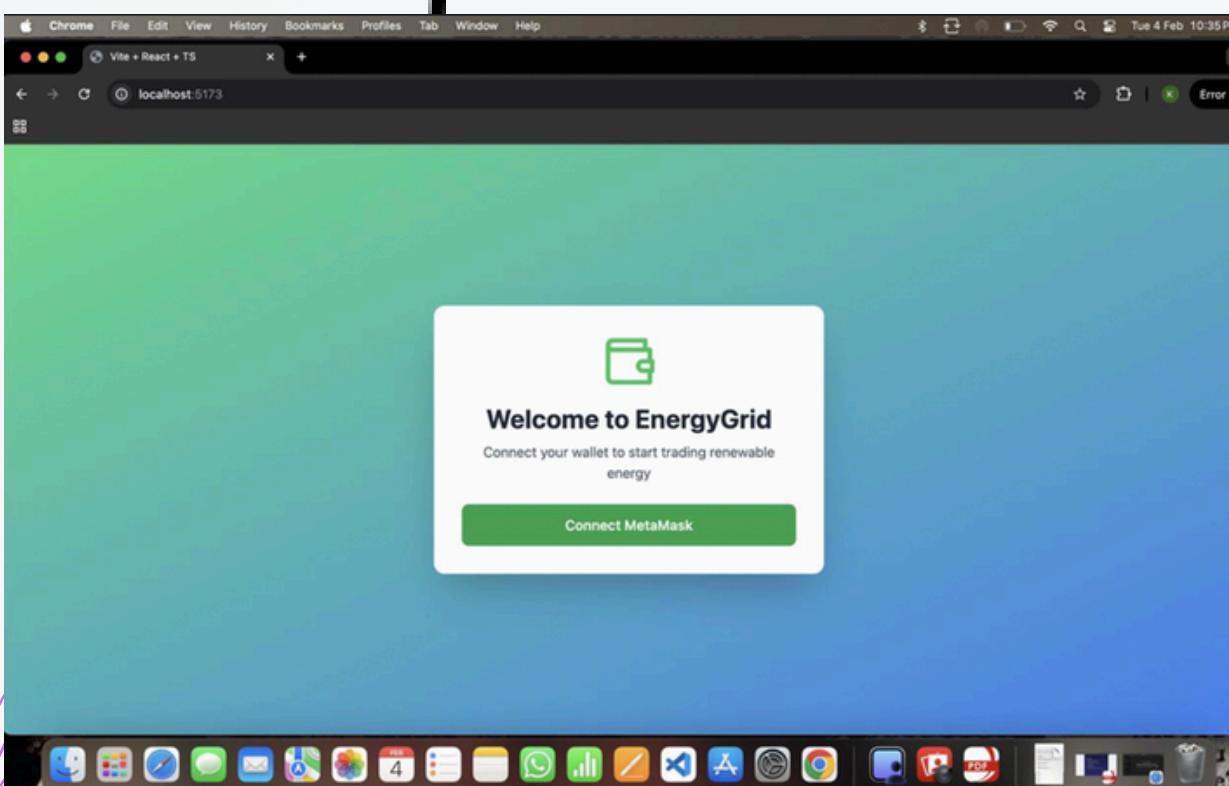
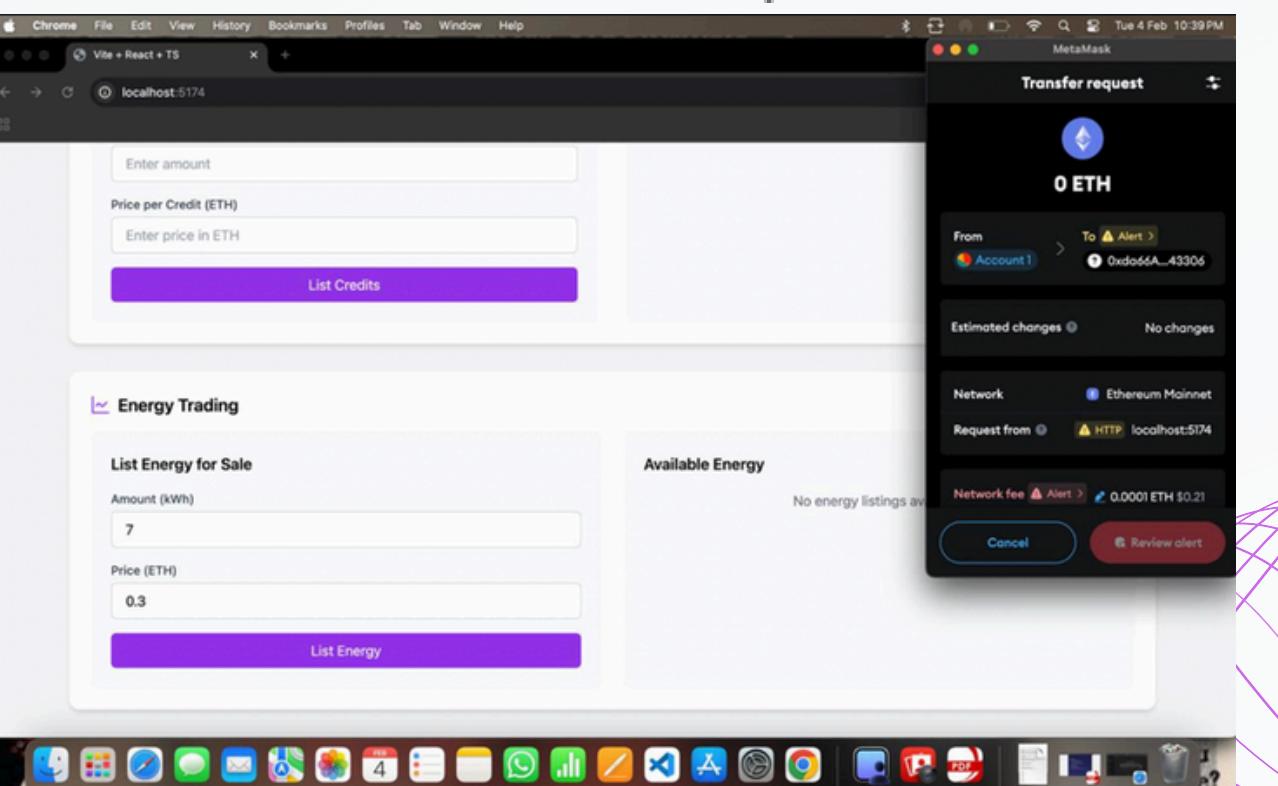
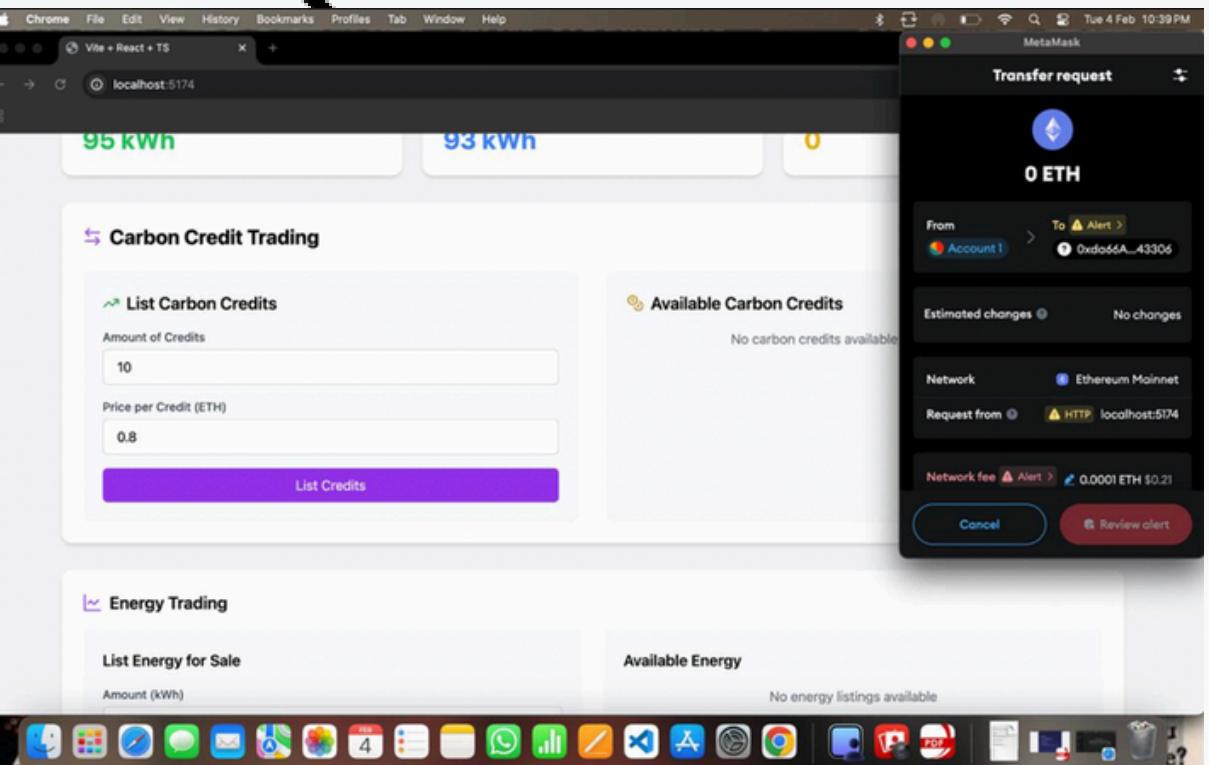
## RUN THE APPLICATION

Start the development server:

```
npm run dev
```



# DAPP FLOW SCREENSHOTS



# IOT INTEGRATION FLOW

## DATA COLLECTION FROM IOT DEVICES AND SMART METERS

- IoT devices and smart meters continuously monitor and collect data.
- This data is formatted and saved into a JSON file.

## STORING DATA ON IPFS

- The JSON file is uploaded to the InterPlanetary File System (IPFS).
- A unique Content Identifier (CID) or hash is generated after successful storage.

## STORING THE HASH ON BLOCKCHAIN

- The generated CID is stored on the blockchain.
- We utilize Alchemy to handle blockchain transactions and storage operations.

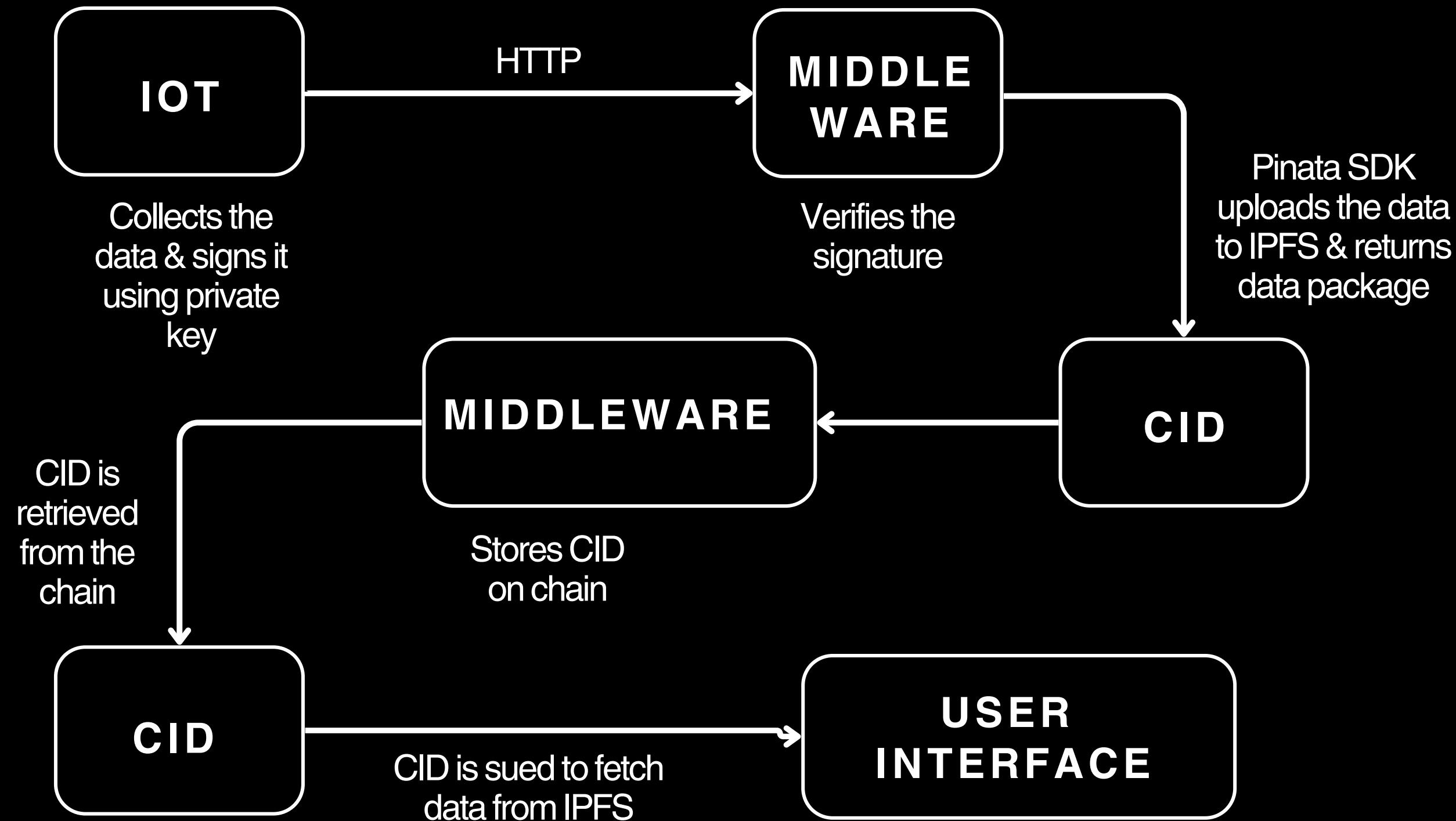
## DATA RETRIEVAL PROCESS

To retrieve and access the data:

1. Fetch the CID: Retrieve the hash stored on the blockchain.
2. Retrieve Data: Use the CID to fetch the actual JSON data file from \*\*IPFS.

For now we are using mock data for IoT which is getting updated in the real-time but our plan is collecting the data from IoT devices and updating it in on a fixed timeframe like everyday, which resolves the problem of high gas fees.

# IOT INTEGRATION WORKFLOW



# HOW WE USED ZK-SNARKS

- **Privacy-Preserving Transactions:** it allows users to prove they own carbon credits or energy assets without revealing sensitive data.
- **Secure and Trustless Verification:** The system verifies ownership and transaction validity without exposing details, ensuring a trustless and efficient trading process.
- **Enhancing Blockchain Scalability:** zk-SNARKs enable lightweight proofs, reducing the computational overhead on the blockchain while maintaining high security and efficiency.
- **Preventing Fraud and Double Spending:** By cryptographically verifying transactions off-chain, zk-SNARKs help prevent issues like double spending of carbon credits while keeping the process decentralized.

## Alchemy

Blockchain infrastructure provider for managing transactions.

## IOT and Smart-Meters

Real-time data sources that collect and monitor system metrics.

## KEY COMPONENTS

### Blockchain

Immutable ledger for securely storing IPFS CIDs (hashes).

### IPFS (InterPlanetary File System)

Decentralized, peer-to-peer storage system for large-scale data.

RESEARCH

DEPLOYED  
WEBSITE  
LINK



# THANK YOU

