

Implementation Plan for DeEnergy Platform (MERN Stack Developer)

Project Overview

DeEnergy is a decentralized solar energy trading platform that allows producers (e.g., Ravi) to tokenize their surplus solar power and sell it directly to consumers (e.g., Anu) using blockchain-based smart contracts.

System Components & Tech Stack

Layer	Technology
Frontend	Next.js, TailwindCSS, RainbowKit, React Hook Form, TypeScript
Backend	Node.js (Express/Fastify), MongoDB, MQTT
Smart Contracts	Solidity (ENGY ERC-20 token, Escrow), Polygon Chain, Chainlink
IoT Integration	Raspberry Pi, Smart Meter, Mosquitto MQTT Broker
Indexing & Storage	The Graph, IPFS/Filecoin, TimescaleDB
Analytics & DAO	Grafana, Aragon/Snapshot (optional)

Phase 1: MVP (1–2 Months)

1. User Management (Producer / Consumer)

- [] Implement MongoDB schemas: User , Wallet , EnergyOffer , Transaction
- [] Wallet-based login using RainbowKit
- [] Role-based dashboards (Producer vs Consumer)

2. Manual Token Minting & Trade (Mocked IoT)

- [] Producer submits surplus kWh via form
- [] Mint ENGY tokens (1 token = 1 kWh) to wallet
- [] Store listing in MongoDB and optionally IPFS

3. Buy Token (Frontend + Escrow)

- ☐ Consumer browses offers
- ☐ Purchase triggers escrow smart contract
- ☐ Funds held until energy is verified

Phase 2: IoT & Blockchain Integration (1–2 Months)

4. Smart Meter & MQTT Bridge

- ☐ Deploy Mosquitto MQTT Broker
- ☐ Raspberry Pi pushes energy data
- ☐ Node.js backend receives and logs to TimescaleDB
- ☐ Push real delivery data to Chainlink Oracle

5. Escrow Contract & Delivery Confirmation

- ☐ Modify smart contract to hold payment
- ☐ Chainlink verifies delivery via IoT data
- ☐ Release funds to producer upon confirmation

Phase 3: Indexing, Analytics, DAO (Optional)

6. Analytics Dashboard

- ☐ Use Grafana or Chart.js for:
 - Tokens Minted
 - Live Energy Delivery Logs
 - Purchase Metrics

7. Governance & DAO

- ☐ ENGY staking for DAO voting
- ☐ Aragon or Snapshot for proposal governance
- ☐ UI for community voting and upgrades

Project Structure (Suggested)

/DeEnergy/

— frontend/	# Next.js with wallet integration
— backend/	# Express/Fastify APIs, MQTT bridge
— contracts/	# Solidity smart contracts (Hardhat/Foundry)
— graph/	# The Graph protocol (subgraph definition)
— pi-client/	# MQTT client for Raspberry Pi (smart meter)
— docs/	# Architecture docs, API specs, DAO guides