

1. Cài môi trường

Cài đặt **Java JDK 21**: : [Amazon Corretto 21 Downloads](#).

Cài đặt **Node.js** và npm : <https://nodejs.org/>

Cài đặt và Cấu hình **PostgreSQL**: [PostgreSQL Downloads](#)

Cài git: <https://git-scm.com/>

Các gói install/dependencies khác đã bị xóa khi upload github nên cần phải tải lại

- Vào **Postgres** thiết lập tài khoản:

tk: postgres

mk: admin

Databases -> Create -> Database

Đặt tên database là: energymarket

Chủ sở hữu (Owner): postgres.

Tạo DB bằng pgAdmin

1. Mở pgAdmin 4

2. Kết nối server:

- user: **postgres**

- password: **admin**

3. Tạo DB:

- Mở cây bên trái → **Databases** → chuột phải → **Create** → **Database...**
 - **Database name:** **energymarket**
 - **Owner:** **postgres**
 - Save

Register - Server

General Connection Parameters SSH Tunnel Advanced Post Connection SQL Tags

Name

energymarket

Server group

Servers

Background

X

Foreground

X

Connect now?

☒

Comments

Close

Reset

Save

Register - Server

General Connection Parameters SSH Tunnel Advanced Post Connection SQL Tags

Host name/address

postgres

Port

5432

Maintenance database

postgres

Username

postgres

Kerberos authentication?

☐

Password

In edit mode the password field is enabled only if Save Password is set to true.

Save password?

☒

Role

Close

Reset

Save

⇒ address chuyển thành localhost

1) Kiểm tra DB **energymarket** đã có chưa

Ở khung trái:

- Mở rộng: **Servers** → **energymarket**
- Mở rộng tiếp: **Databases**
 - Nếu bạn thấy trong danh sách có **database tên energymarket** → **xong phần DB.**
 - Nếu **chưa thấy energymarket** → làm bước 2.

2) Tạo DB **energymarket** (nếu chưa có)

- Chuột phải **Databases** → **Create** → **Database...**
- Tab **General**:
 - Database:** **energymarket**
 - Owner:** **postgres**
- Save**
- Chuột phải **Databases** → **Refresh**

• Vào Git Bash:

```
curl -L https://foundry.paradigm.xyz | bash
```

2

Tắt đi mở lại, rồi chạy: foundryup

script đã cài **foundryup** và đã thêm vào PATH, nhưng **nạp lại shell** rồi mới chạy **foundryup**.

⇒ làm tiếp đúng 2 bước này trong **Git Bash** (ngay cửa sổ đang mở):

Bước 1: nạp lại bashrc (khởi cần đóng mở)

Copy dán:

```
source ~/.bashrc
```

(hoặc đúng như màn hình gợi ý:)

```
source /c/Users/Hang/.bashrc
```

Bước 2: cài Foundry binaries

Chạy:

```
foundryup
```

Bước 3: kiểm tra

Chạy:

```
forge --version
```

Nếu hiện version → **thành công**.

Chuyển file vào ổ C, không để ở mục downloads

cd "C:\energymarket\smartcontract"

git init (hiện chữ master bên cạnh)

mẫu

```
forge script script/Deployment.s.sol:Deployment \  
--rpc-url https://eth-sepolia.g.alchemy.com/v2/hjXDhwZQgf-JnFUpkrbCP \  
--private-key xxx \  
--broadcast \  
--slow
```

hjXDhwZQgf-JnFUpkrbCP: cái này là API key - phần tạo tài khoản alchemy bên dưới

Nếu mà chạy cái deploy bị lỗi thì mới phải install
rm -rf lib
forge install foundry-rs/forge-std
forge install OpenZeppelin/openzeppelin-contracts

Copy output này lưu vào docx

```
EnergyNFT deployed at: 0x.....  
LoyaltyProgram deployed at: 0x.....  
EnergyMarketplace deployed at: 0x.....  
Code mới có thêm deploy lending  
Marketplace address set in NFT contract  
Marketplace authorized in LoyaltyProgram
```

Chạy backend:

Vào trang này: <https://www.alchemy.com/faucets/ethereum-sepolia>

Tạo tài khoản,

Copy API Key, thay vào địa chỉ

<https://eth-sepolia.g.alchemy.com/v2/hjXDhwZQgf-JnFUprbcP>

Sửa cả Nft VÀ Marketplace

cd "C:\energymarket\backend"

sửa file backend/src/main/resources/application.yml trước khi chạy, các thông tin sẽ khác nhau giữa mọi người

```
spring:  
  datasource:  
    url: jdbc:postgresql://localhost:5432/energymarket  
    username: postgres  
    password: 'admin '  
    driver-class-name: org.postgresql.Driver  
  jpa:  
    hibernate:  
      ddl-auto: update  
    show-sql: true
```

```

    properties:
      hibernate:
        format_sql: true
        dialect: org.hibernate.dialect.PostgreSQLDialect

logging:
  level:
    org.springframework.web: INFO

web3:
  rpc-url: "https://eth-sepolia.g.alchemy.com/v2/hjXDhwZQgf-JnFUpkrbcP"

contract:
  nft:
    address: "0x7a1de3C1FA96dD0D01e5036652541d8b25587773"
  marketplace:
    address: "0x7CF0b116FC492eb89F81884888ED2371CFc30740"
  lending:
    address: ".."

ipfs:
  gateway:
    url: https://ipfs.io/ipfs/

app:
  cache:
    ttl: 3600 # Cache TTL in seconds
  pagination:
    default-page-size: 20
    max-page-size: 100

cache:
  evict:
    rate: 3600000 # 1 hour

blockchain:
  node:
    url: "https://eth-sepolia.g.alchemy.com/v2/hjXDhwZQgf-JnFUpkrbcP"

```

Chạy

```
./gradlew bootRun
```

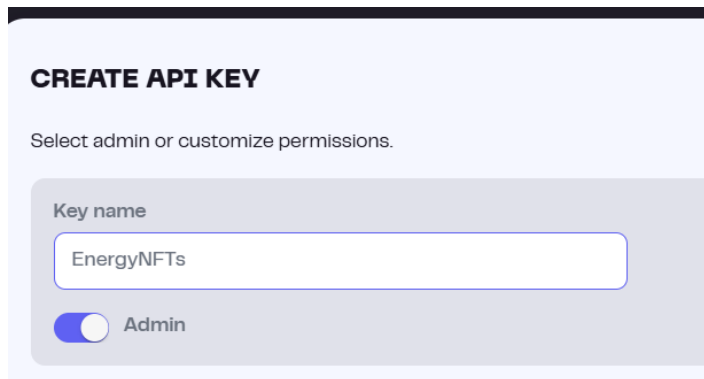
Chạy frontend:

cd "C:\energymarket\frontend"

Vào <https://app.pinata.cloud/register> tạo một tài khoản miễn phí

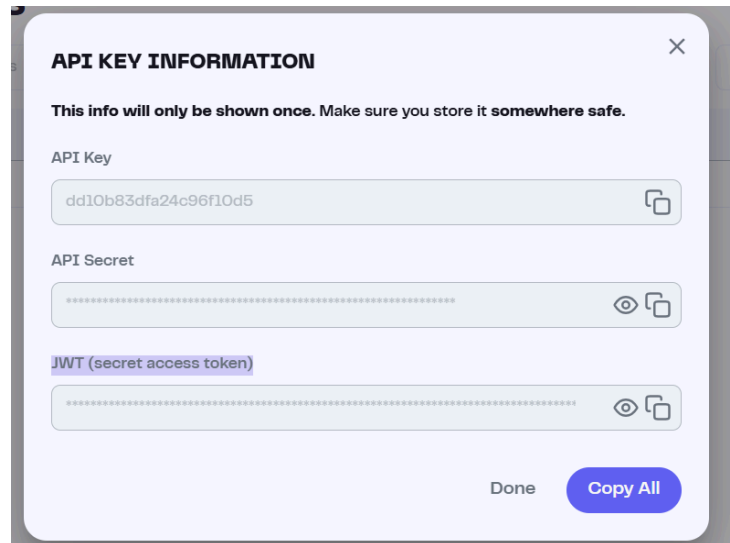
Sau khi đăng nhập, nhìn thanh menu bên trái, chọn vào Group, bấm + New group -> đặt tên EnergyMarket).

1. Copy Group ID vào **PINATA_GROUP_ID**
2. **NEXT_PUBLIC_PINATA_GATEWAY**=<https://gateway.pinata.cloud>
3. **PINATA_JWT**, vào API Keys, đặt tên tiếp EnergyNFTs , bấm quyền Admin



Bấm Create

Copy cái JWT (secret access token) (RẤT DÀI) vào **PINATA_JWT**



Sửa file .env.local

[illegible]

PINATA_GROUP_ID=2110ed6c-ba02-4e7a-aa39-577285b556d9

Vào lại cái cd "C:\energymarket\frontend"

⇒ Mở terminal mới (Git Bash khác) giữ Gitbash backend đang process

npm install

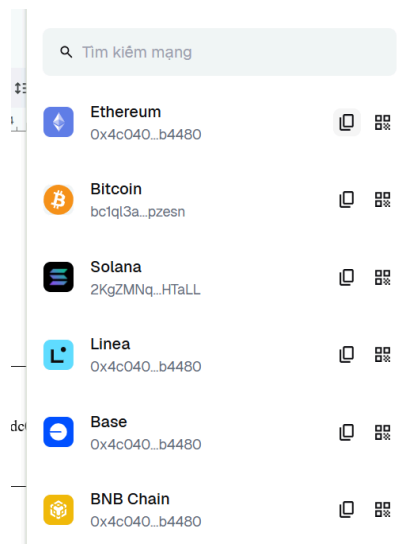
npm run dev

Vào gg gõ: <http://localhost:3000>

Từ metamask, vào tài khoản, xem chi tiết tài khoản

Vào khóa riêng tư, sao chép khóa

Vào Mạng



Sao chép địa chỉ ví

Chuyển tiền từ tiền ở metamask vào NFT Web, như kiểu mình cần nạp tiền vào web để thực hiện mua bán, còn tiền ở metamask cần thêm để trả phí gas thôi

```
cast send \  
  --rpc-url https://eth-sepolia.g.alchemy.com/v2/hjXDhwZQgf-JnFUpkrbcP \  
  --private-key xxx \  
  "produceEnergy(address,uint256)" \  

```


0x4c040e708ff5daf50fd39a11ac86ff03206b4480 \

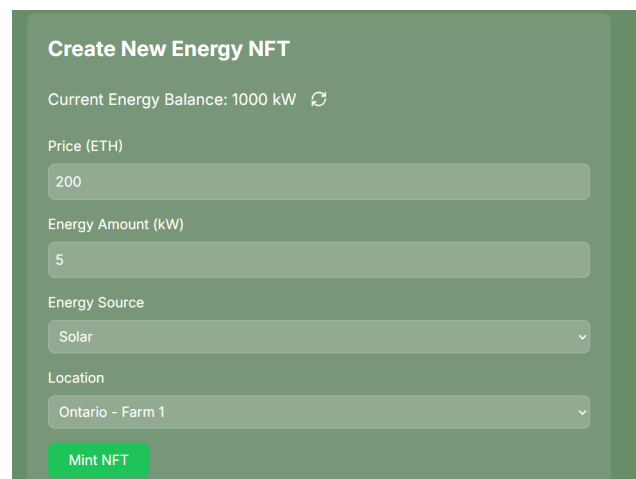
2000

hxDhwZQgf-JnFUprbcP: này là API Key ở trang web, mỗi ng có 1 key khác
private-key xxx ở metamask

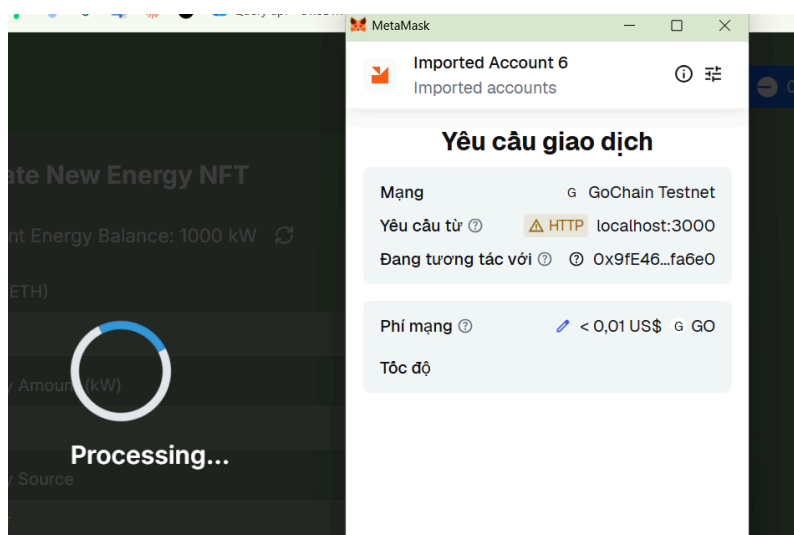
0x4c040e708ff5daf50fd39a11ac86ff03206b4480: địa chỉ ví cần nhận tiền, lưu ý: nếu cần chuyển energy balance vào 1 tài khoản khác thì thay địa chỉ ví nhận này thôi; mọi cái private key giữ nguyên (vì nó là private key của admin, và chỉ admin mới có quyền gửi đi năng lượng (vì là năng lượng giả mà))

Máy toai chờ tầm 2 phút, rồi load lại web, hoặc bấm nút xoay cạnh Current Balance, thấy update lên 1000 là được.

⇒ dùng mạng testnet => ở advanced của setting metamask bật Hiển thị các mạng thử nghiệm



Nhấn mint thấy trở ra cửa sổ metamask là dc



Vào port bank

```
cd "C:\energymarket\bank-portal"  
npm run dev
```

<http://localhost:3001>

Nếu kết nối với ví mà deploy phần đầu thì nó sẽ hiện giao diện quản lý thanh khoản

Nếu kết nối với các account ví bình thường khác thì sẽ hiện giao dịch cho vay, chấm điểm tín dụng, hạn mức, thời hạn lãi suất,

Rồi sẽ chuyển thẳng energy vào web marketplace cho mình, và lưu vào blockchain số tiền mình còn nợ, mình sẽ bắt buộc phải trả nợ, thông tin sẽ không thể sửa đổi