

## Token ERC20

**Biên soạn:** Nhóm Nghiên Cứu Blockchain Khoa HTTT

---

### 1. Định nghĩa Token<sup>[1], [2]</sup>

- Token là một **đơn vị tiền ảo** được tạo dựa trên **nền tảng** của một loại tiền ảo khác **đã có sẵn** trong **blockchain** để hoạt động.
- So sánh Token và Coin**

	Token	Coin
Tính năng	<ul style="list-style-type: none"><li>Có mục đích sử dụng rộng hơn coin.</li><li>Thường dùng cho các mục đích tiện ích cụ thể (GAS, CMT,...)</li></ul>	<ul style="list-style-type: none"><li>Coin được tạo ra với mục đích sử dụng như tiền tệ.</li><li>Có thể lưu trữ.</li><li>Có thể giao dịch được.</li></ul>
Nền tảng	<ul style="list-style-type: none"><li>Phụ thuộc vào nền tảng của coin.</li></ul>	<ul style="list-style-type: none"><li>Có nền tảng riêng.</li></ul>

- Fungible Token:** (FT) Không duy nhất và có thể phân chia. Ví dụ đô la (\$) phát hành không duy nhất và có thể phân chia nhỏ ra thành nhiều đơn vị: 1\$, 2\$, 5\$, 10\$,.... Trong Ethereum tiêu chuẩn: ERC-20 (Năm 2016)
- Non-fungible Token:** (NFT) Là duy nhất và không thể phân chia. Ví dụ vé máy bay là duy nhất và không thể phân chia nhỏ ra được. Trong Ethereum tiêu chuẩn: ERC-721(Năm 2017)
- Multiple Token:** (MT) Là sự kết hợp giữa Fungible Token, Non-fungible Token và các loại token khác. Trong Ethereum có tiêu chuẩn ERC-1155 (Năm 2018)

### 2. Cài đặt Truffle

- Để bắt đầu cho Project TokenCreation bước đầu tiên **cài đặt Truffle**

```
npm install -g truffle
```

- Khởi tạo Project

```
> mkdir <project_name>
> cd <project_name>
> truffle unbox react
```

```
PS E:\Solidity\ProjectResearch\TokenCreation\ERC20> mkdir example

Directory: E:\Solidity\ProjectResearch\TokenCreation\ERC20

Mode                LastWriteTime         Length Name
----                -
d-----          7/3/2022  11:44 PM             example

PS E:\Solidity\ProjectResearch\TokenCreation\ERC20> cd example
PS E:\Solidity\ProjectResearch\TokenCreation\ERC20\example> |
```

### 3. Fungible Token (ERC-20)

- **ERC20 Smart Contract Installation**

Cài đặt bằng dòng lệnh sau đây:

```
npm install --save @openzeppelin/contracts@v3.0.0
```

- **ERC20 Interface**

- **Các đặc tính của ERC20**

- **ERC20** cung cấp cơ chế chuyển token từ một tài khoản sang tài khoản khác.
- Lấy số lượng token hiện tại đang có của một tài khoản.
- Lấy tổng số lượng token đang sẵn có trong mạng.
- Chấp thuận nếu số lượng token gửi của một tài khoản có thể sang bên thứ 3.

- **Giải thích cụ thể về ERC20:** <https://eips.ethereum.org/EIPS/eip-20>

#### FUNCTIONS

```
constructor(name, symbol, decimals)
name()
symbol()
decimals()
```

```
totalSupply()
balanceOf(account)
transfer(recipient, amount)
allowance(owner, spender)
approve(spender, amount)
transferFrom(sender, recipient, amount)
```

IERC20

**ERC20 Interface gồm được định nghĩa như sau:**

<b>FUNCTIONS</b>
<code>totalSupply()</code>
<code>balanceOf(account)</code>
<code>transfer(recipient, amount)</code>
<code>allowance(owner, spender)</code>
<code>approve(spender, amount)</code>
<code>transferFrom(sender, recipient, amount)</code>
<b>EVENTS</b>
<code>Transfer(from, to, value)</code>
<code>Approval(owner, spender, value)</code>

#### **4. Tài liệu tham khảo**

- [1] <https://200lab.io/blog/fungible-va-non-fungible-token-su-khac-biet-la-gi/>,  
[Online] [Thời gian truy cập: 24/06/2022]
- [2] <https://ethereum.org/en/developers/docs/standards/tokens/erc-1155/>,  
[Online] [Thời gian truy cập: 24/06/2022]
- [3] [https://www.tutorialspoint.com/solidity/solidity\\_error\\_handling.htm](https://www.tutorialspoint.com/solidity/solidity_error_handling.htm), [Online]  
[Thời gian truy cập: 23/06/2022]
- [4] <https://ethereum.org/en/developers/docs/standards/tokens/erc-20/>, [Online]  
[Thời gian truy cập: 07/04/2022]