

did-besu-contract

`did-besu-contract` 는 OpenDID EVM Contract를 구현한 프로젝트로, 블록체인 상에서 DID Document(DID 문서)와 Verifiable Credential Metadata(VC Meta) 정보를 저장하고 관리하는 기능을 제공합니다. 이 프로젝트는 분산 신원(DID) 및 검증 가능한 자격 증명(VC)을 지원하기 위해 설계되었습니다.

S/W 사양

- 스마트 컨트랙트: Solidity
- 개발 환경: Hardhat
- 테스트: Mocha, Chai
- 블록체인 네트워크: Ethereum 기반 네트워크

Contract 주요 기능

- DID Document 관리: DID 문서를 블록체인에 저장, 조회, 업데이트, 상태 변경 등의 기능을 제공합니다.
- VC Metadata 관리: Verifiable Credential의 메타데이터를 저장하고 관리할 수 있습니다.
- ZKP Data 관리: ZKP 관련 데이터를 블록체인에 저장, 조회 기능을 제공합니다.
- 확장성: OpenZeppelin 라이브러리를 활용하여 업그레이드 가능한 스마트 컨트랙트를 지원합니다.

설치 및 배포

1. 의존성 설치

```
npm install
```

2. 환경 설정

`.env` 파일을 생성하고 필요한 환경 변수를 설정합니다. 예:

```
PRIVATE_KEY=<your-private-key>  
RPC_URL=<your-rpc-url>
```

3. 컴파일

스마트 컨트랙트를 컴파일하려면 다음 명령어를 실행하세요:

```
npx hardhat compile
```

4. 배포

스마트 컨트랙트를 배포하려면 다음 명령어를 실행하세요:

```
npx hardhat run scripts/deploy.js --network <network-name>
```

5. 테스트

테스트를 실행하려면 다음 명령어를 사용하세요:

```
npx hardhat test
```

디렉토리 구조

```
did-besu-contract
├── contracts/      # 스마트 컨트랙트 코드
│   ├── data/      # 데이터용 컨트랙트 라이브러리
│   ├── storage/   # 저장용 컨트랙트 코드
│   └── utils/     # 유틸성 컨트랙트 라이브러리
├── test/          # 테스트 코드
├── scripts/       # 배포 스크립트
├── artifacts/     # 컴파일된 아티팩트
├── cache/         # Hardhat 캐시
├── data/          # 샘플 데이터 및 스키마
└── doc/          # 문서화 자료
```

각 폴더와 파일에 대한 설명은 다음과 같습니다: