

RevealStake dNFT Collection

Buy revealable dynamic NFT collection with staking rewards.

Documentation

Just run this command to open the book in your browser:

```
$ mdbook serve
```

Currently, the mermaid diagrams are not supported in mdbook. So, in order to preview the diagrams either prefer this [VSCode extension](#) or view in Github.

Usage

Build

```
$ forge build
```

Test

```
$ forge test
```

To view gas-usage in table format:

```
$ forge test --gas-report
```

Fork testing

On Sepolia testnet:

```
$ forge test --mp test/sepolia/SPNFTTest.t.sol --fork-url $SEPOLIA_RPC_URL
```

Format

```
$ forge fmt
```

Contract size

```
$ forge build --sizes
```

Gas Snapshots

```
$ forge snapshot
```

Deploy

To any network:

```
$ forge script script/SPNFT.s.sol:SPNFTScript --rpc-url <your_rpc_url> --private-key <your_private_key> --broadcast
```

Deploy to Anvil (local network):

```
# Run local network
$ anvil
# Deploy
$ forge script script/SPNFT.s.sol:SPNFTScript --rpc-url
http://localhost:8545 --private-key
0xac0974bec39a17e36ba4a6b4d238ff944bacb478cbed5efcae784d7bf4f2ff80 --broadcast
```

Deploy to **Sepolia** and verify on Etherscan:

Set the `.env` as per the `.env.example` file.

```
$ source .env
$ forge script script/SPNFT.s.sol:SPNFTScript --rpc-url $SEPOLIA_RPC_URL --private-key $DEPLOYER_PRIVATE_KEY --broadcast --verify
```

With logs enabled using `-vvvvv`, you can see the transaction hash and the etherscan link.

Flatten

```
$ forge flatten src/SPNFT.sol -o flatten/src/SPNFTFlattened.sol
$ forge flatten src/RevealedSPNFT.sol -o flatten/src/RevealedSPNFT.sol
```

Bindings

Generate bindings for Rust EVM client SDK:

```
$ forge bind
```