

# LABORATORIUM TECHNOLOGII SIECIOWYCH:

## Blockchain i kryptowaluty

Opracował Marcin Augustynowicz

### Ścieżka 1 (Java+Maven)

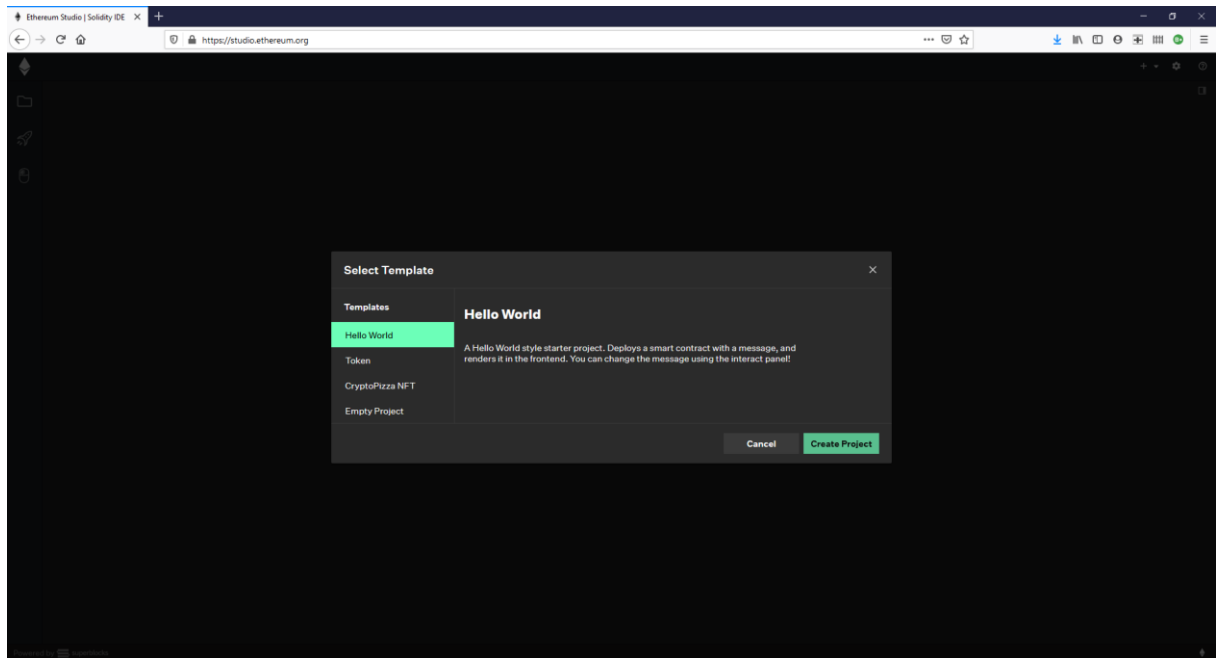
1. Sklonować repozytorium: <https://github.com/ronni7/BlockchainLab>
2. Otworzyć projekt w dowolnym IDE umożliwiającym tworzenie projektów w języku Java i obsługującym narzędzie Maven.
3. Zbudować projekt za pomocą polecenia mvn clean/mvn build
4. Przejść do katalogu currentState ścieżka (<NazwaProjektu> (domyślnie Blockchainlab) /src/main/currentState) i usunąć plik blockchain.json
5. Uruchomić projekt i sprawdzić czy plik blockchain.json został ponownie utworzony. W przypadku błędów związanych z odczytem/zapisem plików dostosować ścieżki do pliku blockchain.json w kodzie (plik Main.java)
6. W pliku Main.java pod napisem //TODO add transactions here zmienić wartość transakcji (pierwszy parametr o nazwie amount) na dowolną inną poprawną wartość, skompilować i uruchomić projekt
7. Jeżeli plik blockchain.json został poprawnie zaktualizowany, zrobić zrzut ekranu (do sprawozdania)
8. W pliku Main.java pod napisem //TODO add transactions here dopisać jedną lub więcej własnych transakcji (można również zatwierdzać wiele bloków za pomocą metody blockchain.acceptCurrentState()) i ponownie skompilować i uruchomić projekt.
9. Jeżeli plik blockchain.json został poprawnie zaktualizowany, zrobić zrzut ekranu (do sprawozdania)
10. Zaproponować (słownie, lub implementując) dodatkowe informacje, które może przechowywać blok i transakcja i umieścić w sprawozdaniu ( w przypadku implementacji, wkleić też zrzut ekranu reprezentujący dodatkową funkcjonalność)

### Ścieżka 2 (.Net)

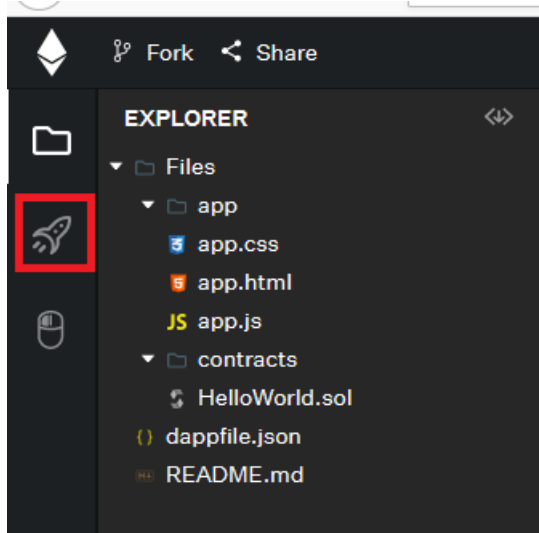
1. Sklonować repozytorium: <https://github.com/plipowczan/MySimpleBlockchain>
2. Otworzyć projekt w dowolnym IDE umożliwiającym tworzenie projektów w języku C# i uruchomić projekt
3. Jeżeli projekt uruchomił się poprawnie, załączyć zrzut ekranu poprawnie uruchomionego programu
4. Dodać możliwość zapisywania i odczytywania blockchain do pliku
5. Postępować jak w Ścieżce Javowca, od punktu 6 (oczywiście odpowiednie pliki będą miały inne nazwy np. Main.java -> Program.cs )

### Ścieżka 3 (Online, Platforma Ethereum)

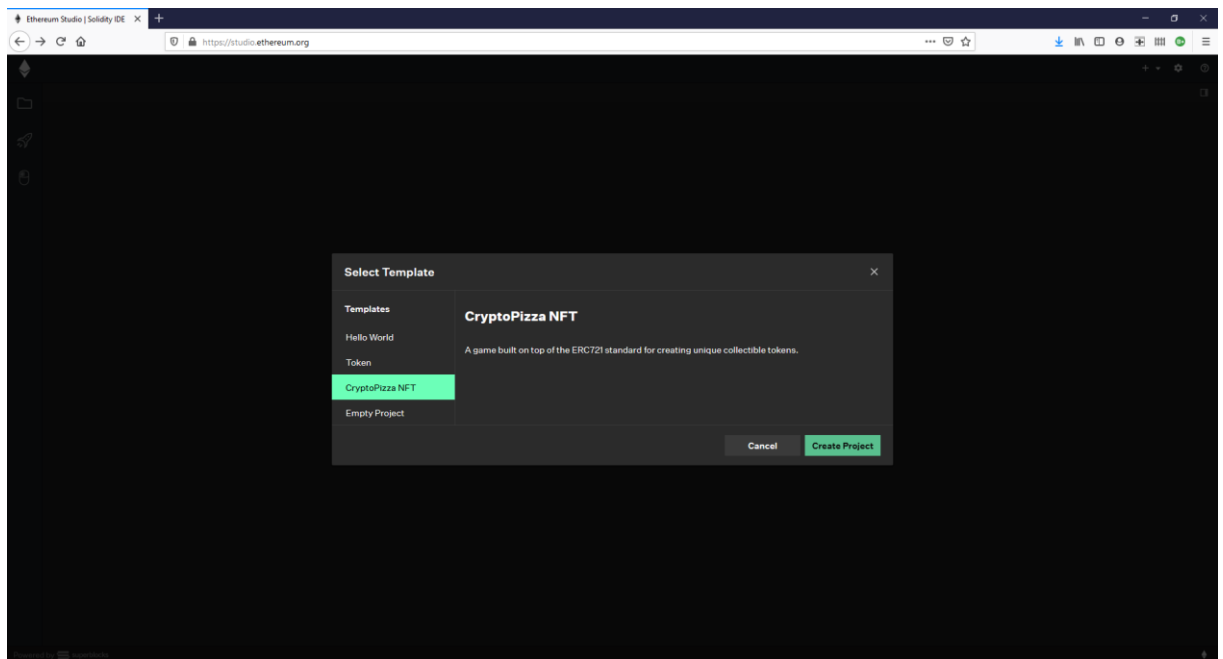
1. Wejść w przeglądarce pod adres <https://studio.ethereum.org/>
2. Wybrać projekt o nazwie Hello World z listy po lewej



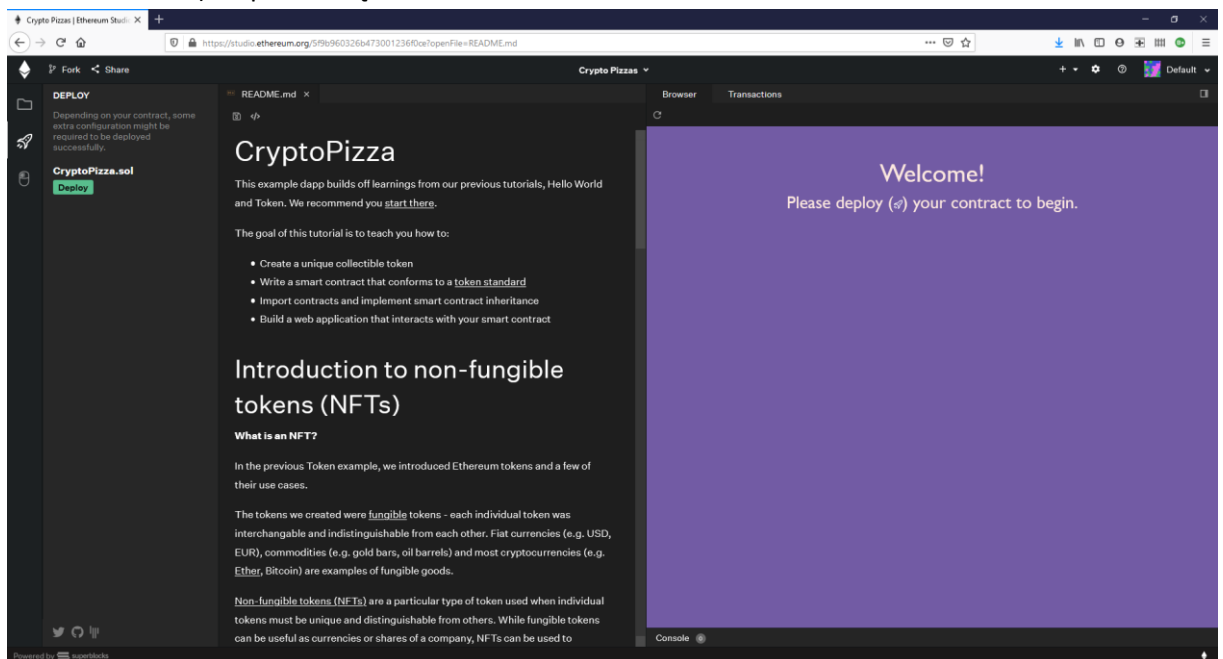
3. Zapoznać się z edytorem online i uruchomić projekt (Ikona rakiety z paska narzędzi po lewej stronie ekranu)



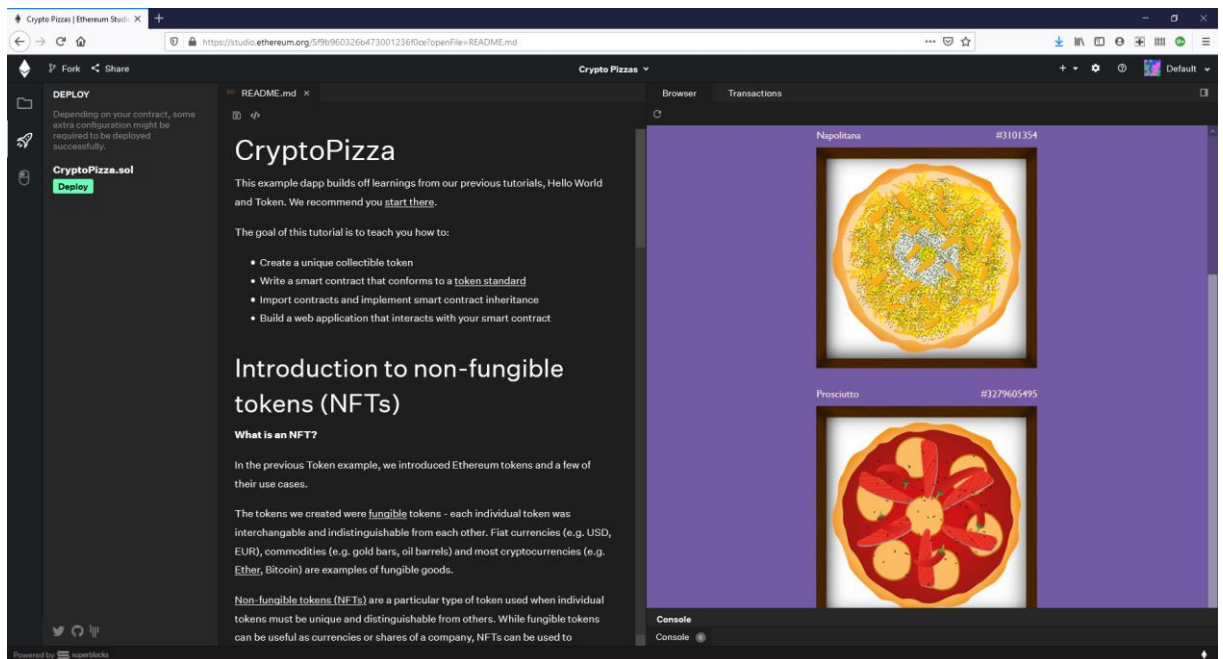
4. Zamieścić zrzut ekranu poprawnie wykonanego programu do sprawozdania
5. Zamknąć kartę i ponownie wejść pod adres <https://studio.ethereum.org/>
6. Wybrać projekt o nazwie CryptoPizza NFT



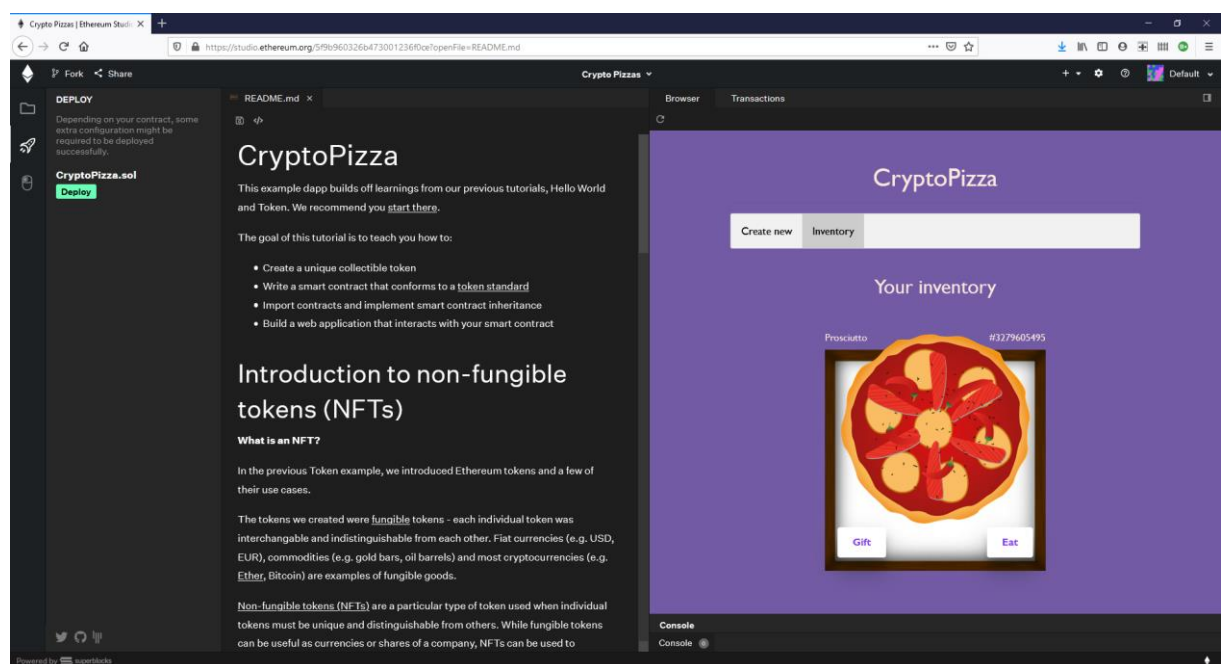
7. Uruchomić projekt (rakietą => przycisk Deploy) i zapoznać się z Readme.MD oraz strukturą pliku CryptoPizza.sol (jest to plik napisany w języku Solidity pozwalający pisać programy (tzw. Smart Contracts) na platformę Ethereum).



8. W prawej części ekranu uruchomiła się aplikacja. Dodać dwie CryptoPizze o nazwach Napolitana i Prosciutto (składniki i tak pojawiają się losowo)
9. Przejsć na zakładkę Inventory i zamieścić zrzut ekranu do sprawozdania



10. Zjeść jedną pizzę (najechać kursorem na obrazek z pizzą i w prawym jego dolnym rogu kliknąć przycisk 'Eat'), drugą spróbować podarować dowolnemu użytkownikowi poprzez przycisk Gift w lewym dolnym rogu obrazka z pizzą (transakcja nie musi się powieść).



11. Przejść na zakładkę Transactions w górnej części ekranu i zamieścić screen listy transakcji (dla chętnych, można spróbować przeanalizować i opisać znajdujące się tam informacje).

Default

Browser

Transactions

CryptoPizza burn

From:

0xa48f2e0be8ab5a04a5eb1f86ead1923f03a207fd

To:

Value: 0 ether

Age: 3 min

Block #4 (Index 0)

Gas used: 18440

Function arguments: 0

Show more

CryptoPizza createRandomPizza

From:

0xa48f2e0be8ab5a04a5eb1f86ead1923f03a207fd

To:

Value: 0 ether

Age: 5 min

Block #3 (Index 0)

Gas used: 97700

Function arguments: Prosciutto

Show more

CryptoPizza createRandomPizza

From:

0xa48f2e0be8ab5a04a5eb1f86ead1923f03a207fd

To:

Value: 0 ether

Age: 5 min

Block #2 (Index 0)

Gas used: 126135

Function arguments: Napolitana

Show more

CryptoPizza deploy

Creator:

0xa48f2e0be8ab5a04a5eb1f86ead1923f03a207fd

Contract address:

0xa9e73bb65b54c445081dae9d67f08ccbcce8bcb7

Age: 5 min

Block #1 (Index 0)

Console

Console