



THE NFT BAR

Have your cocktail NFT!
Kit, Pauline, Jason



Summary



Our goal was to build a user-friendly web application for the users to sell or buy the Artwork NFT.

The front end (UI) of our application is powered by Streamlit.

The back end layers are enabled by:

- Seller Features: a Solidity smart contract, which allows them to add their Artwork and set the price of their listing.
- Buyer Features: an executable python file, which allows the customers to browse, buy the artworks and pay with their crypto wallet.



Functionalities



Seller Page

We utilized Solidity and Streamlit to build the web-based interface with three main sections:

- Register New Artwork
- Edit the Product Listing
- Get Price History

Buyer Page

We utilized Python and various libraries, such as os, requests, dotenv, bip44, web3, Account, and Wallet, to compose four main functions:

- Shopping Cart
- Generate Account
- Get Balance
- Send Transaction



- Front-end layer - a web application accessible for users.
- A Solidity smart contract enables sellers to add and set the price of their Artworks, edit the price of their listings and get the price history of a specific listing.
- Web3 API provides the connection between the web front end and smart contract
- The blockchain encrypts and stores transactional data.



Seller Features

- We deploy the ArtToken smart contract to a local blockchain by using Remix and Ganache
- Perform development and functional testing by using test accounts:
 - To registry a new piece of Artwork and review the price history of the Artwork
 - Edit the price and review the price history of the Artwork

Seller Page Demo

The screenshot shows a Streamlit application running at `localhost:8501`. The title bar indicates the window is titled "seller_page - Streamlit". The main content area has a dark background with white text.

Welcome to NFT Bar

Choose an account to get started

Select Account

0xd3f1D2749d5Ec49ADceD3D3a27658836515531B2

Register New Artwork

Enter the name of the artwork

Enter the artist name

Enter the price (ETH)

Enter the URI to the artwork

Register Artwork

The right side of the browser window features a vertical sidebar with various icons, likely for navigation or settings, typical of a Streamlit application's interface.

Seller Page Result

Welcome to NFT Bar

Choose an account to get started

Select Account

0xB63d7110B0CF92bd45a9AA1C0F5d12Af17C51B3C

Register New Artwork

Enter the name of the artwork

Manhattan

Enter the artist name

Nick Barclay Design

Enter the price (ETH)

20

Enter the URI to the artwork

<https://www.solopress.com/blog/wp-content/uploads/2014/09/manhattan-minimalist-cocktail-poster>

Register Artwork

TokenId: `0`

Transaction receipt mined:

```
{  
  "transactionHash" :  
    "HexBytes('0x30d68344abec78cc34a52d2f820af1fa18b51adfd5b1480fc20216a64f603b7')"  
  "transactionIndex" : 0  
  "blockHash" :  
    "HexBytes('0xf745d86286548ebb525a897d47260af8bf02e17a75b2cd2b08127658850a4b9b')"
```

Get the price history

Artwork ID

1

Get Price Reports Details

```
AttributeDict({'tokenId': 1, 'price': 4, 'reportURI': '20% discount'})
```

Edit the product listing

Choose an Art Token ID

1

Enter the new price (ETH)

4

Enter notes about the change

20% discount

Update price

```
b'\xe8\xf3\xb8!\x9c\xcd\xab.\x7f\xbc\x8\xfe\x9e\x085RZ\xd7\xc5\xbc\x98TRFV\xb3\x1ea>\x8e'
```

- Front End: User interface built with streamlit code
- Back End:
 - Main page: `Classic_Cocktails_NFT.py`
 - Multi Page functionality: a `Page` folder for the other pages, `1_Elegant_Cocktails_NFT.py` and `2_Long_Drinks_NFT.py`
 - Functions: `crypto_wallet.py` contains the functions to generate account, get balance and send transaction. The functions are imported into the Streamlit code.
 - Env file contains the mnemonic key for the `buyer's account`.



Buyer Page Demo

Classic_Cocktails_NFT - Stream

localhost:8501

Fintech Bootcamp GitHub DataCamp CopyRich Academy LinkedIn Gmail YouTube KUCas myKU ChatGPT AlgoTrading101 GO Markets 調酒進階課 Other bookmarks

NFT BAR

Have a Classic Cocktail NFT!

Classic Cocktails NFT

Elegant Cocktails NFT

Long Drinks NFT

Client Account Address and Ethernet Balance in Ether

0x6577Ec0Cb35898EeC49befEE9E05Cb2D30e9c3d6

Your Balance of Ether

84.625

Select a Cocktail

Negroni

Cocktail Name, Price, Transaction Fee and Ethereum Address

Negroni

5

18°C Mostly cloudy

Search

Old-Fashioned

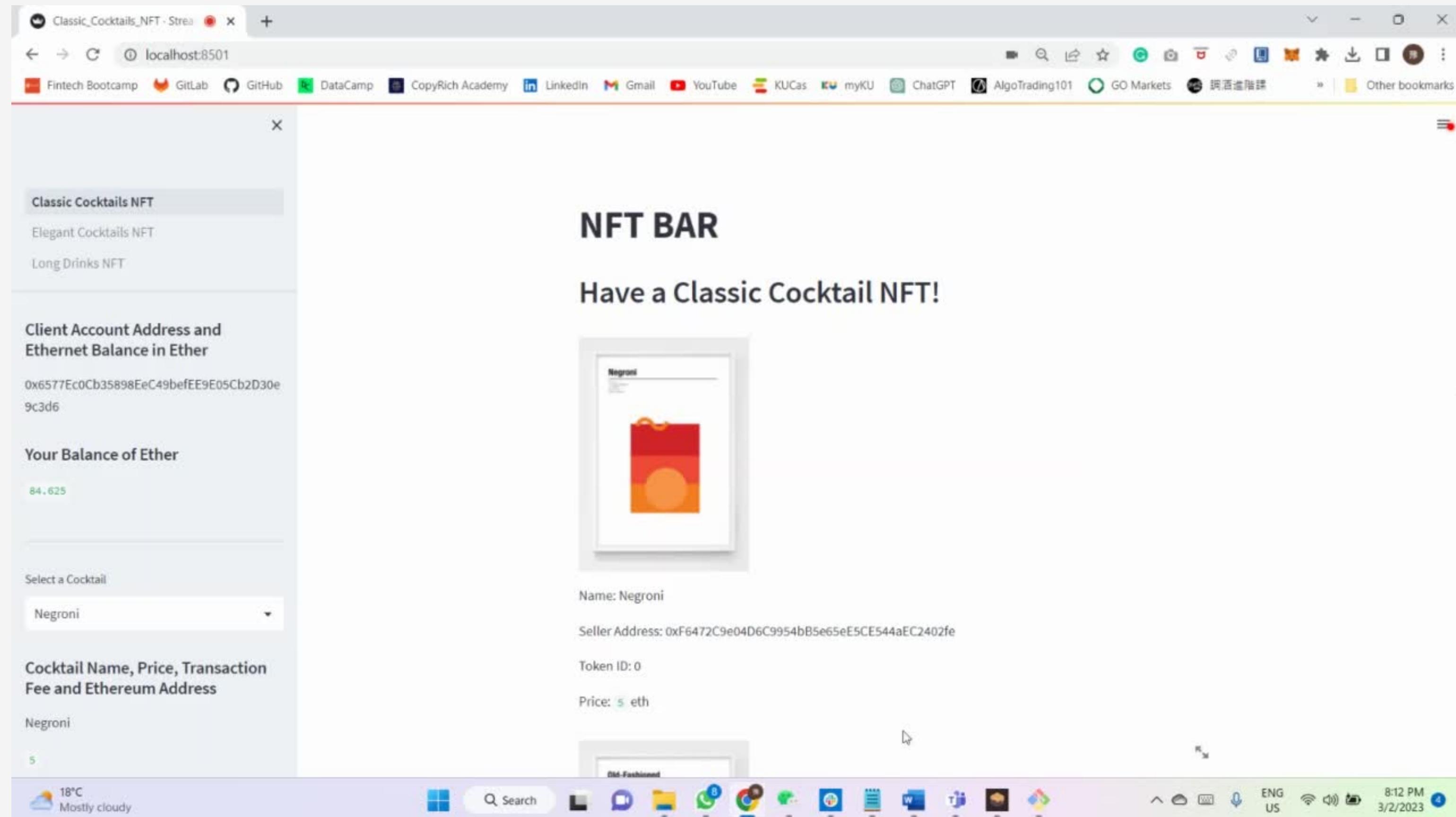
Name: Negroni

Seller Address: 0xF472C9e04D6C9954bB5e65eE5CE544aEC2402fe

Token ID: 0

Price: 5 eth

8:12 PM 3/2/2023



Testing the code

ADDRESS	BALANCE	TX COUNT	INDEX	
0x6577Ec0Cb35898EeC49befEE9E05Cb2D30e9c3d6	93.85 ETH	1	0	🔑
0xF6472C9e04D6C9954bB5e65eE5CE544aEC2402fe	100.00 ETH	0	1	🔑
0x28b7905b2c924Fe5fA8555bEC98243Acf3eD555C	100.00 ETH	0	2	🔑
0x17d2682a6Eb41504CFF977E315fe7e0f6d9bE85d	100.00 ETH	0	3	🔑
0xcab67550CeCAa7cd34D81C29B832E0fbC2a89E01	100.00 ETH	0	4	🔑
0x2157032A1dA319f967FaAf2dABa07E379CAF2Fdb	100.00 ETH	0	5	🔑
0x05F98e8b028380266455aCee5C0183746B449ad3	106.15 ETH	0	6	🔑
← BACK	TX 0xfe56ef3a05d9e216312118382430868dcc29a0ae3dd51e1b5e8b66bf66100503	CONTRACT CALL		
SENDER ADDRESS	0x6577Ec0Cb35898EeC49befEE9E05Cb2D30e9c3d6	TO CONTRACT ADDRESS	0x05F98e8b028380266455aCee5C0183746B449ad3	
VALUE	6.15 ETH	GAS USED	21000	GAS LIMIT
			0	21000
TX DATA	0x			MINED IN BLOCK
				1

Buyer Page Result

NFT BAR

Have a Classic Cocktail NFT!



Name: Negroni
Seller Address: 0xF6472C9e04D6C9954bB5e65eE5CE544aEC2402fe
Token ID: 0
Price: 5 eth

Client Account Address and Ethernet Balance in Ether
0x6577Ec0Cb35898EeC49befEE9E05Cb2D30e9c3d6
Your Balance of Ether
100

Select a Cocktail
Negroni

Cocktail Name, Price, Transaction Fee and Ethereum Address
Negroni
5

NFT BAR

Have a Elegant Cocktail NFT!



Name: Cosmopolitan
Seller Address: 0xcab67550CeC9a7cd34D81C29B832E0fbC2a89E01
Token ID: 3
Price: 8 eth

Client Account Address and Ethernet Balance in Ether
0x6577Ec0Cb35898EeC49befEE9E05Cb2D30e9c3d6
Your Balance of Ether
100

Select a Cocktail
Cosmopolitan

Cocktail Name, Price, Transaction Fee and Ethereum Address
Cosmopolitan
8

NFT BAR

Have a Long Drink NFT!



Name: Bloody Mary
Seller Address: 0x1Ab81C688086507157e9c71345Bd14BcA0db696E
Token ID: 7
Price: 10 eth

Client Account Address and Ethernet Balance in Ether
0x6577Ec0Cb35898EeC49befEE9E05Cb2D30e9c3d6
Your Balance of Ether
100

Select a Cocktail
Bloody Mary

Cocktail Name, Price, Transaction Fee and Ethereum Address
Bloody Mary
10

Total Price in Ether
6.15

Send Transaction

Validated Transaction Hash

```
b'\xfeV\xef:\x05\xd9\xe2\x161!\x188$0\x86\x8d\xcc)\xa0\xae=\xd5\x1e\x1b^\x8bf\xbff\x10\x05\x03'
```

Shopping Cart

nftbar - Streamlit

localhost:8501

Your Account Address and Ethernet Balance in Ether
0xd2Cd7d41A97584ecbC811cF8213aB482cc62c481

Your Balance of Ether
100

My Shopping Cart:

Name	Price	Seller_Address
empty		

[Clear cart](#)

Total Payable in Ether: 0

Amount payable grouped by seller:

Seller_Address	Price
empty	

[Checkout](#)

Welcome to NFT Bar

Cocktail

TokenID: 0
Name: Manhattan
Artist: N. Barclay
Price: 20 eth
Seller address: 27D9F2238EfA446

Coffee

TokenID: 4
Name: Cappuccino
Artist: N. Barclay
Price: 20 eth
Seller address: 366b53

Tea

TokenID: 8
Name: Black Tea
Artist: N. Barclay
Price: 20 eth
Seller address: 0xc9b0A36b22DDBaCddfb366b53

Stop sharing Hide

10:57 PM 1/03/2023



Unanticipated insights



- We would like to have different categories of NFT: We implemented the `Multipage` structure on the Streamlit Web App.
- The shopping bag kept overwriting itself: resolve the shopping cart issue using `streamlit.session_state`.
- We would like the item to show `sold` once it is being sold: added the feature where once the nft is added to cart, the button "add to cart" will be disabled.

Future Development

- We could consider using external data storage for the user generated content
- Combining the seller page and buyer page and allow them to communicate with each other
- Adding a sold status to show what has sold on our website
- Extensive transaction history
- Product review
- Sale history showing who owned the art before.

Unwind at The NFT BAR.

Come and get a cocktail
NFT of your own today!

