

Ethereum based Marketplace

Mahidhar Gedela 113604363

My website link: <https://mahidhar404.github.io/Blockchain/>

My GitHub link: <https://github.com/mahidhar404/Blockchain>

This project required us to use solidity programming which was new to me. So, I used the help of chatgpt and tutorialspoint to familiarize myself to implement the marketplace smart contract.

Project Design

I started with creating a Marketplace.sol on remix IDE which is used to build and deploy smart contracts. Creating entities looked like creating custom data structures in C++. For security purposes I surfed the internet and came across RentrancyGuard contract which is help in terms of security. So, the contract has to main functions.

1. listItem: This function is used to list Items on the marketplace website. It takes three arguments namely item_name, item_description and price. So once the user enters valid information making sure the price is valid, the itemCount is increased and a new ID for the item is created. ItemListed is a kind of EventListener used to update and notify the user about the listed item.
2. buyItem: This function allows the users to buy the available Items which are not yet sold. The user must give the ItemID and connect his wallet to the UI to make a transaction. To prevent users from not buying already sold items, there is a flag item.sold which is set to true once sold. ItemSold is

again a EventListener used to update and notify the user about the sold item.

So, once the contract is compiled, I first tested it on the Testnet – Sepolia environment to check whether the transactions are being deployed. This can be seen on the metamask website under Etherscan. Metamask is used to store Ether coins and connect the created wallet to the smart contracts.

I then deployed the smart contract on the Injected – Provider Metamask environment. So, the contract address is “0xdBe640E869Fd42103cd355270004e9eCB22578a8”. I also used the ABI - json file created file by the contract in the javascript file to do the mappings for integrating the front-end and back-end.

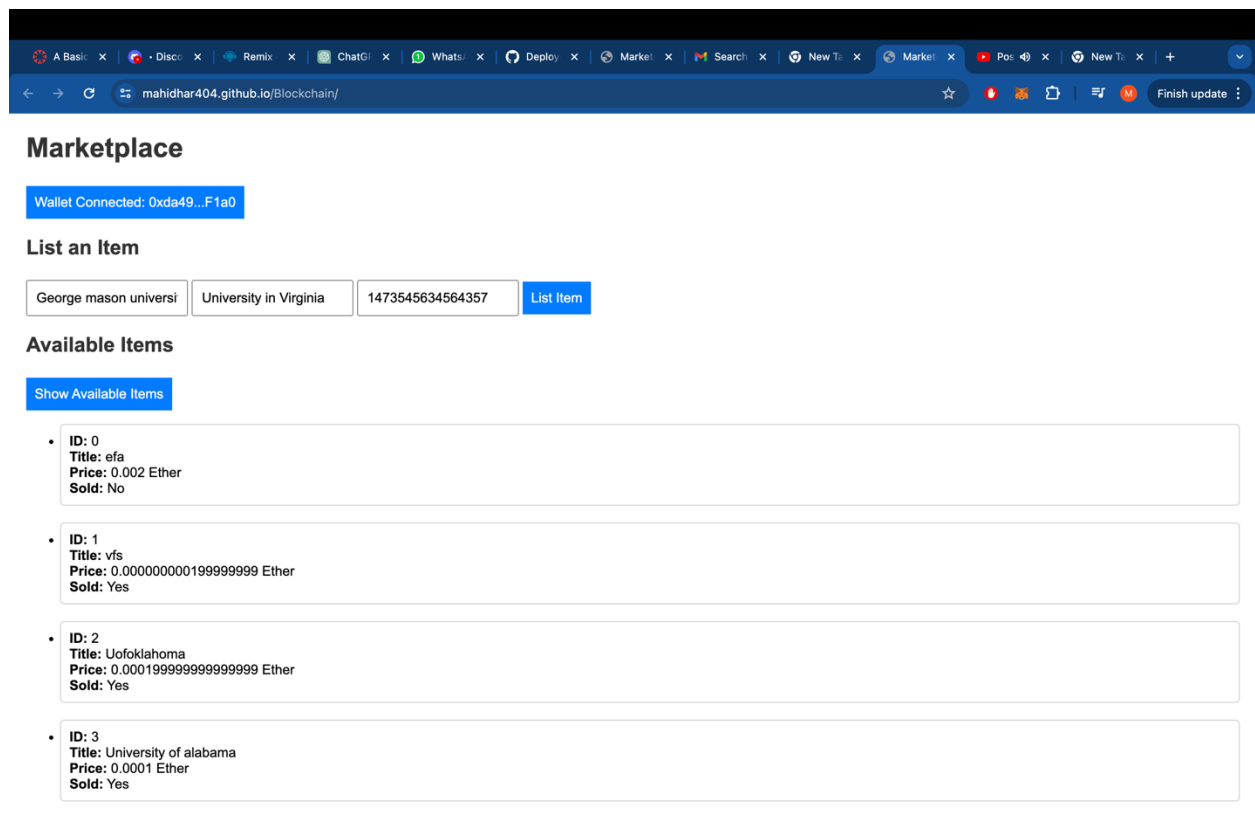
So, the next step so to create a basic working website. The UI has the following functionalities.

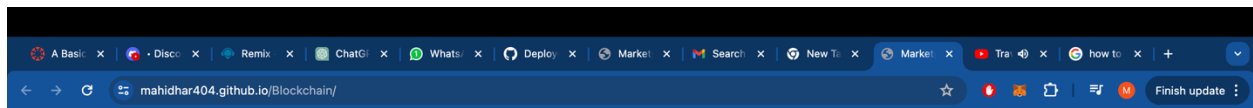
1. **Wallet connection:** There is a button to connect your wallet to the website.
2. **List Items:** Here is where I act as an admin. I gave some universities, their location as the description and random price to list the items. Once the items are listed, a hash is displayed as a pop-up. To see already sold-out items, a button is created which shows all the items.
3. **Buy Items:** The user after connecting their wallet, can look through the available Items and give a valid ID to buy the item with listed price and a small transaction fee is added. Once the item is bought a pop-up shows the hash.

I then deployed the working website along with all the necessary items on the github. Github has a built-in feature to deploy our website to get a https link.

Challenges faced:

I was initially stuck at the point after compiling the code. I deployed it to the Sepolia test net, created the basic UI and used the ABI to connect the connect. However, it took me a while to realize that for the listed items to reflect, I had to connect to the Metamask environment. So, I later took the help of chat gpt to figure out and then I deployed and changed the ABI json which resulted in a working website.





- **ID:** 10
Title: George mason university
Price: 0.001473545634564357 Ether
Sold: No

Sold Items

Show Sold Items

- **ID:** 1
Title: vfs
Description: vsfv
Price: 0.000000000199999999 Ether
Sold: Yes
- **ID:** 2
Title: Uofoklahoma
Description: University
Price: 0.00019999999999999999 Ether
Sold: Yes
- **ID:** 3
Title: University of alabama
Description: University
Price: 0.0001 Ether
Sold: Yes
- **ID:** 4
Title: University of Texas
Description: University
Price: 0.000013423322354 Ether
Sold: Yes
- **ID:** 5
Title: UofWisconsin
Description: Univesity
Price: 0.000019324532245355 Ether
Sold: Yes

Buy Item

Buy Item