Introduction to Blockchain and cryptocurrency

Semester Project

Submitted By:

Aleezeh Usman 18I-0529

Areesha Tahir 18I-1655

Sophia Syed 18I-0835

Introduction to Blockchain and cryptocurrency 1

Project Title 3

Project description 3

Scope: 4

Functionality: 4

Use Case Diagram 5

Functionality Demonstration 6

Scenario 1: Less Money in bid 6

Scenario 2: Ownership transferred 6

User Interfaces 7

Home Page 8

Search Item Page 8

Add Item Page 9

Place Bid Page 10

Project Title

Jinjja - Product verification through blockchain

Project description

Jinjja means ‘real’ in Korean, which is what our application stands for. Jinjja aims to counter the pirated and fake copies of priceless unique artifacts present in the world using a more transparent, reliable and trustworthy platform. Jinjja also aims to make a level plainfield for buying and selling unique artifacts where anyone can place a bid and the seller can accept any bids. It is a platform to verify the authenticity of an artifact and check its credentials and ownership.

Scope:
Jinjja will be used to authenticate unique artifacts, however it is built primarily keeping in mind the art industry where each artifact is extremely unique and exists as a lone piece. Each piece will have a unique private code that will only be accessible to the owner of the contract and the owner of the object.

Functionality:

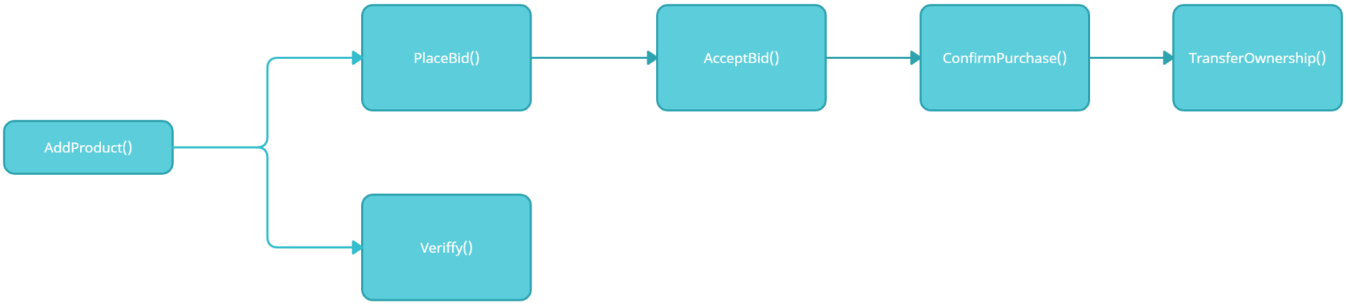
The following are the two main classes that the smart contract will be using:

```
struct Product{
    string product_title;
    bytes32 product_qrcode;
    string date_created;
    string product_desc;
    uint price;
    address original_owner;
    address payable curr_owner;
    string priv_code;       //Only visible to the owner of the product
    //Will keep a log of all the owners of the product and a corresponsing log of the dates the ownership was
    transferred to that owner. latest owner will be the current owner
    address[] prevowner;
    string[] ownershipdate;
}

struct Bid{
    uint BidID;
    string code;
    uint offer;
    address buyer;
    bool status;
}
```

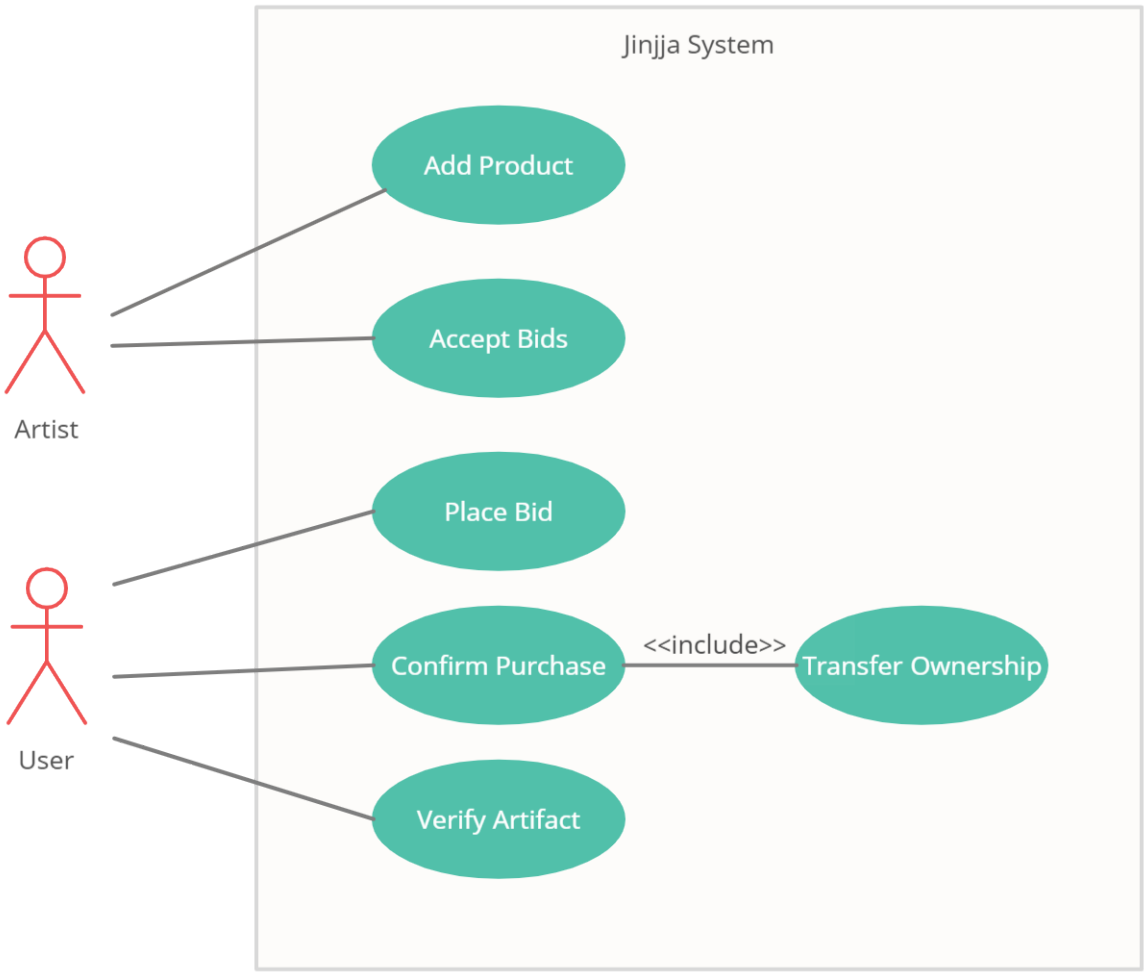
Product will store a title, *a private code* which only the owner of the artifact or owner of contract will have access to and a hash of that code which will be visible to everyone. In addition the actual owner that created the art piece and the date of creation will be stored along with the current owner and an entire log of all the owners and corresponding dates the transfer of ownership occurred on.

Following is a sample workflow of the project:



A creator/artists will first add a product to the blockchain which can then be used to verify if a product that seems to be the same artifact is original or not using the unique code and hash on the blockchain. Furthermore, bids can be placed to purchase the object, the owner can choose to accept any bid, bids can only be placed if the offer is higher than the artists minimum demand. Once the artist has accepted a bid, the buyer can confirm purchase by paying and the smart contract will automatically transfer the ownership to the buyer once the amount has been transferred.

Use Case Diagram



Functionality Demonstration

Here is a working test run of our backend code (smart contract) using ganache and truffle.
Initial list of products:

```
truffle(ganache)> jj.GetHiddenList().then(value=>value.toString())
'MONA LISA,0x058da274e74a84e635f90a5d8fb2cabf96bc5d456b696705d6a8e9439b0b7c3b,12/3/2019,--,80,0x8D6e755D18e00bEC9B3
8172A60Bd4BCC6D4204a9,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,ML0001,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,
12/3/2019,STARRY NIGHT,0x7e2c9c859da6b1bb8356dbd5a9940197cb8198e3f69320fd05994b87a67bfbcf,2/5/2019,--,70,0x8D6e755D
18e00bEC9B38172A60Bd4BCC6D4204a9,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,SN0001,0x8D6e755D18e00bEC9B38172A60Bd4B
CC6D4204a9,2/5/2019,BLOCKCHAIN,0xefb70922b76f142641b333bb9e1dcabcf645a778e9edcc7c51788dcbb37b2482,1/2/2021,--,40,0x
8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,BC0001,0x8D6e755D18e00bEC9B3817
2A60Bd4BCC6D4204a9,1/2/2021'
```

Scenario 1: Less Money in bid

As we can see in one of the following bids the money that the buyer is trying to bid is less than the owner's requirement therefore it will not be entertained.

```
truffle(ganache)> jj.PlaceBid('MONA LISA',79,{from:accounts[3]})
```

```
truffle(ganache)> jj.PlaceBid('MONA LISA',81,{from:accounts[1]})
```

```
truffle(ganache)> jj.viewBids().then(value=>value.toString())
'0,ML0001,81,0x37f737025b5FD12B619Fd1A3D0bbB0213CE8ACe9,false,1,BC0001,45,0xC252eac1eA7cE3a999471F11ddA639B1B6D0a33
D,false,2,BC0001,43,0xfd4B140DA7611C0968644657ddCEf86200fB16b6,false'
```

In the above list of bids, the bid for 'MONA LISA' that has ID 'ML0001' with less money has not been added and can not even be considered.

Scenario 2: Ownership transferred

In the below scenario two buyers place bids for an artifact which the owner choses from the bids and accepts one of them which changes the status of that bid to 'true' after which the buyer can confirm the purchase and pay for the artifact which is then automatically transferred to the owner of the artifact.
Here we can see the list of bids that have been places for 'BLOCKCHAIN' artifact:

```
truffle(ganache)> jj.viewBids().then(value=>value.toString())
'0,BC0001,43,0xC252eac1eA7cE3a999471F11ddA639B1B6D0a33D,true,1,BC0001,45,0xfd4B140DA7611C0968644657ddCEf86200fB16b6
,false'
```

Here we can see the previous owner of the artifact i.e account[0] in our Ganache blockchain:

```
truffle(ganache)> jj.GetHiddenList().then(value=>value.toString())
'BLOCKCHAIN,0xefb70922b76f142641b333bb9e1dcabcf645a778e9edcc7c51788dcbb37b2482,1/1/1,--,40,0x8D6e755D18e00bEC9B3817
2A60Bd4BCC6D4204a9,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,BC0001,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,1/1
/1'
```

Here we can see that the previous owner accepted one of the bids and therefore the artifact has been transferred to the buyer:

```
truffle(ganache)> jj.getCurrentOwnerfromName('BLOCKCHAIN').then(value=>value.toString())
'0xC252eac1eA7cE3a999471F11ddA639B1B6D0a33D'
```

Thus, we can see a successful transfer of ownership from one account to another.

Scenario 3: Verifying an Artifact

```
truffle(ganache)> jj.GetHiddenList().then(view=>view.toString())
'MONA LISA,0x813ee6a3576f84ebe7ad746eebe12fb0ec04cd3fecf070ea0a6cf7d8e05c0d0b,1/1/1,--,20,0x8D6e755D18e00bEC9B38172
A60Bd4BCC6D4204a9,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,ML001,0x8D6e755D18e00bEC9B38172A60Bd4BCC6D4204a9,1/1/1
'
truffle(ganache)> jj.VerifyProduct('ML0001').then(value=>value.toString())
'FAKE'
truffle(ganache)> jj.VerifyProduct('ML001').then(value=>value.toString())
'REAL'
```

Using the special hash code that is generated using the hidden private code an item can be verified to prove whether it is authentic or not.

User Interfaces

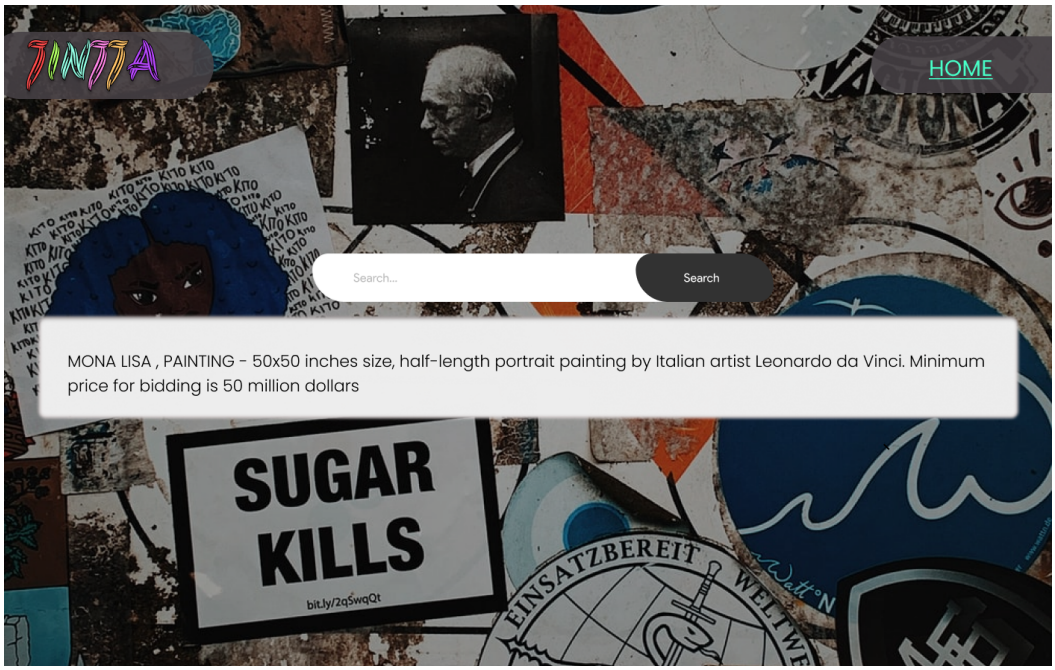
Home Page

The Home Page consists of a navigation bar on the top right corner from which we can go to the search page and the create page respectively.
It also has our logo in the center.

Search Item Page

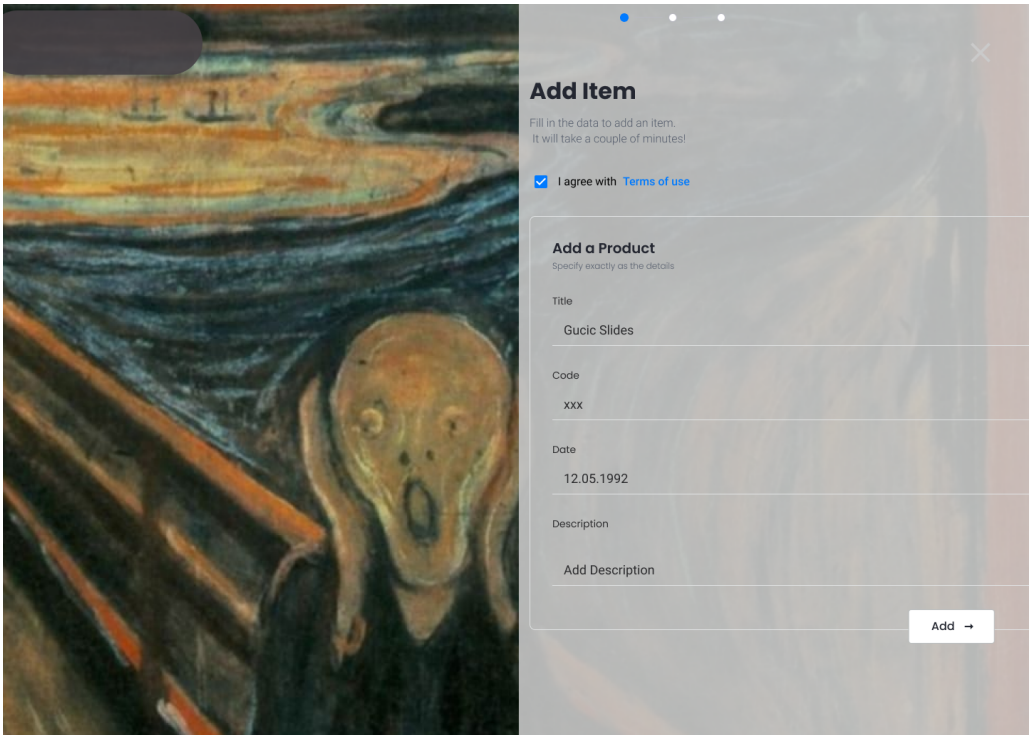
Our Search Item Page has a search bar from which we can validate the authenticity of the artifacts by inserting the cryptographically hashed QR Code.

On the top right corner we have the Home Button which will take us to the Home Page(See above)



Add Item Page

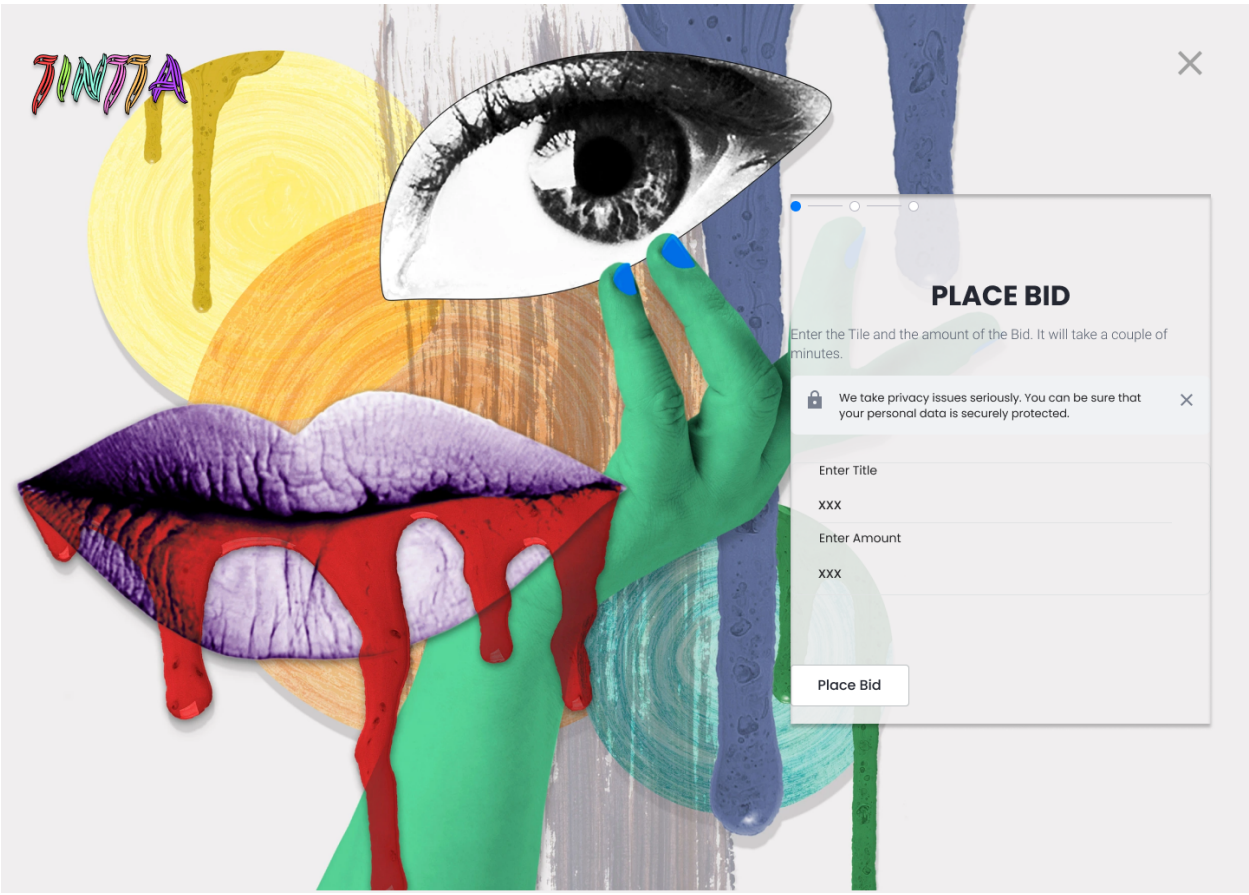
The Add Item Page has a form which takes in the Title,Code,Date and the Description of the Product we want to add. The item will then be added onto the blockchain through the smart contract.



The art piece used in the add item page was literally us throughout the whole project :))

Place Bid Page

In our Place Bid screen, we have made a form which takes in the Title of the product we want to place the bid for, and the amount of the bid.



Disclaimer: front has been designed but not connected to the backend code due to connectivity issues.