BlockChain Final Project

—

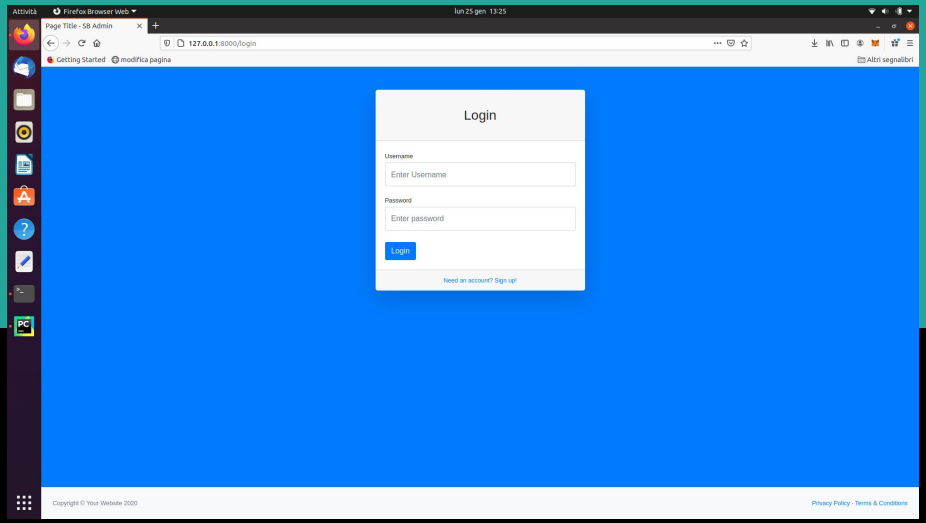
by Giovanni Maria Mazza

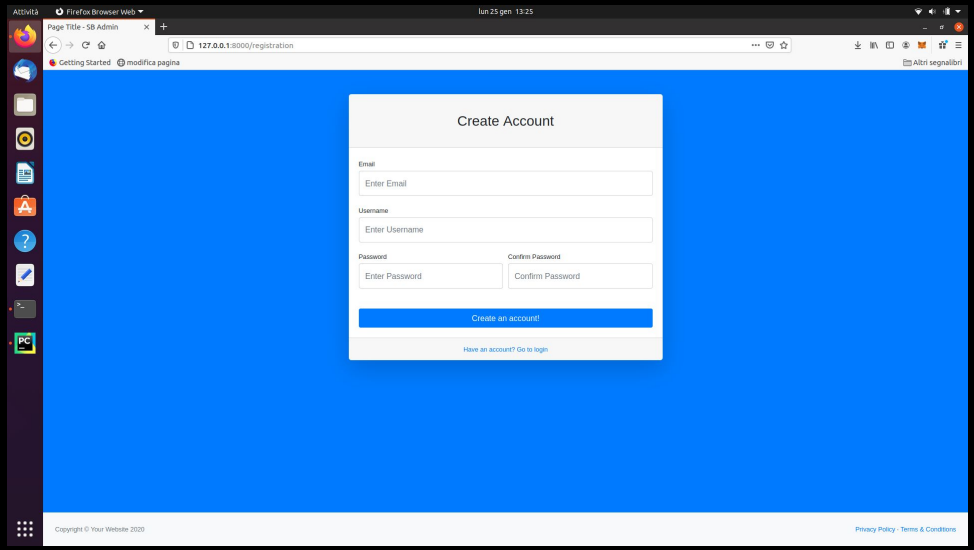
Introduction

This is my auction site in which auctions completed are stored in relational database and generate an hash from a JSON file having all the info about it which will be written on the Ropsten Ethereum BlockChain.

Login / Register Page

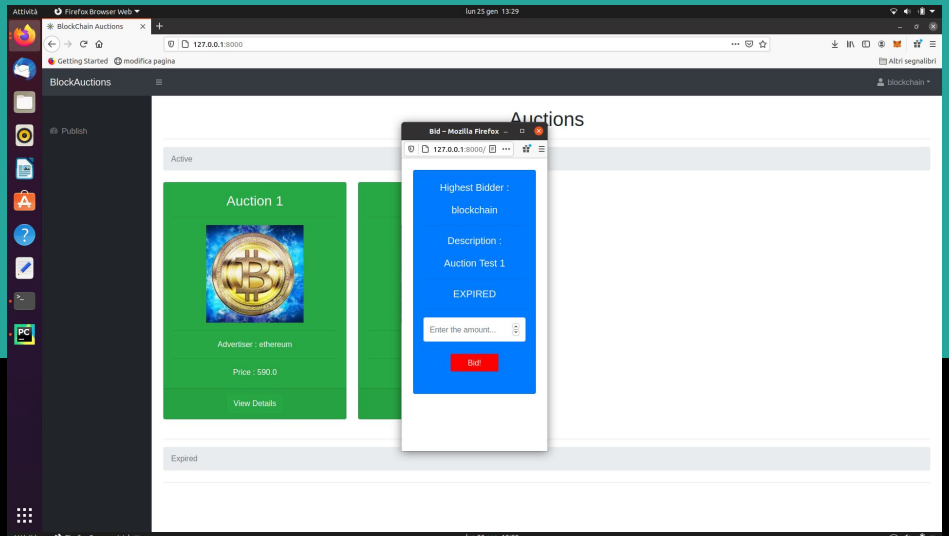
You can create an account which generates a profile user in which are stored informations about won and published auctions.

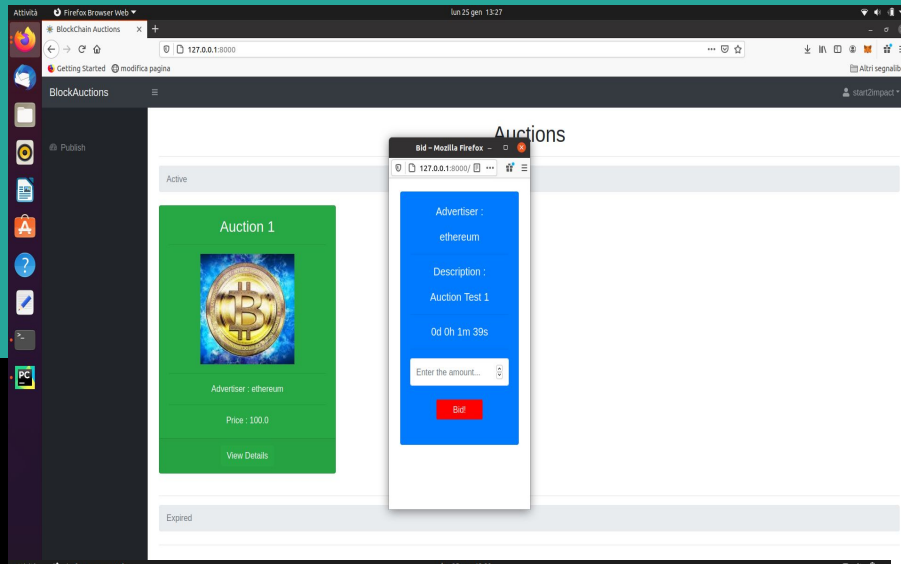

Attività
Firefox Browser Web
lun 25 gen 13:25
Page Title - SB Admin
127.0.0.1:8000/login
Getting Started
modifica pagina
Altri segnalibri
Login
Username
Enter Username
Password
Enter password
Login
Need an account? Sign up!
Copyright © Your Website 2020
Privacy Policy · Terms & Conditions

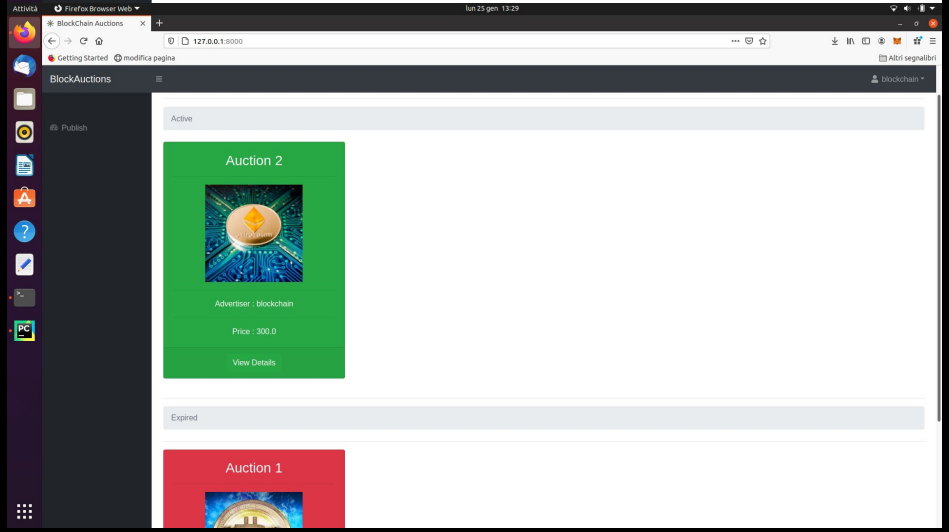

Attività
Firefox Browser Web
lun 25 gen 13:25
Page Title - SB Admin
127.0.0.1:8000/registration
Getting Started
modifica pagina
Altri segnalibri
Create Account
Email
Enter Email
Username
Enter Username
Password
Enter Password
Confirm Password
Confirm Password
Create an account!
Have an account? Go to login
Copyright © Your Website 2020
Privacy Policy · Terms & Conditions

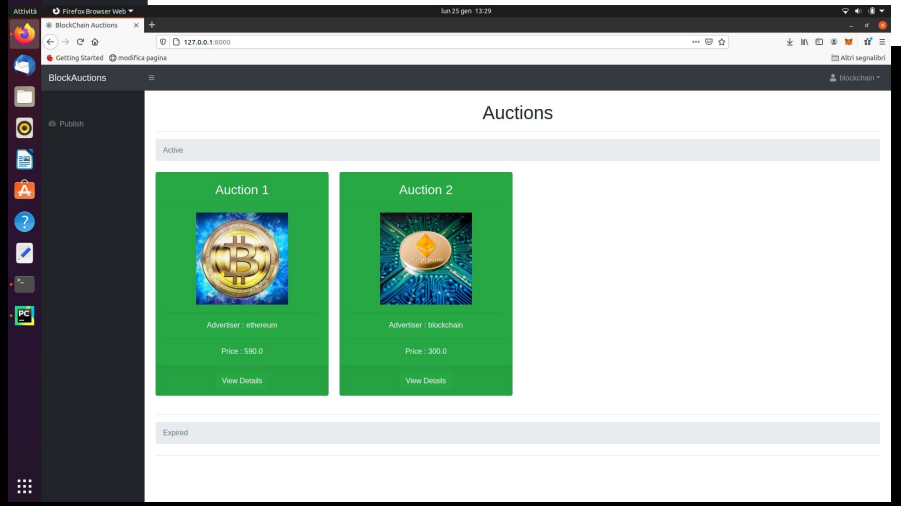
Homepage

In the homepage you can see active and expired auctions and , if logged in, you can interact looking for info and bidding up on them thanks to popup windows.


BlockAuctions
Auctions
Active
Auction 1
Advertiser : ethereum
Price : 590.0
View Details
Bid – Mozilla Firefox
Highest Bidder :
blockchain
Description :
Auction Test 1
EXPIRED
Enter the amount...
Bid!
Expired

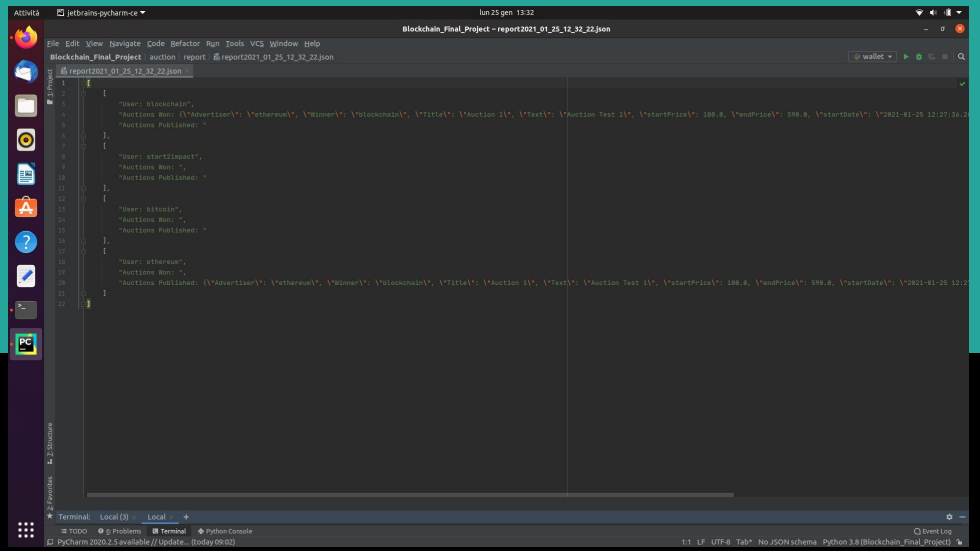

BlockAuctions
Auctions
Active
Auction 1
Advertiser : ethereum
Price : 100.0
View Details
Bid – Mozilla Firefox
Advertiser :
ethereum
Description :
Auction Test 1
0d 0h 1m 39s
Enter the amount...
Bid!
Expired

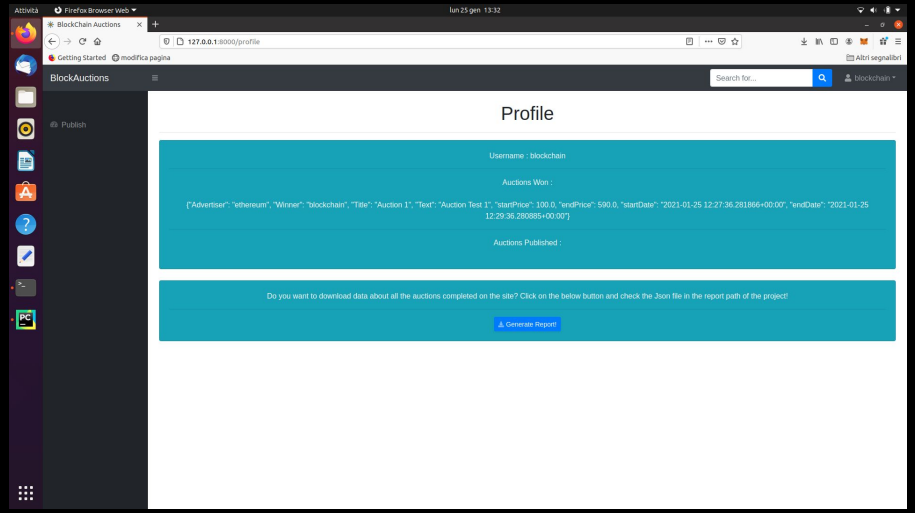

BlockAuctions
Active
Auction 2
Advertiser : blockchain
Price : 300.0
View Details
Expired
Auction 1


BlockAuctions
Auctions
Active
Auction 1
Advertiser : ethereum
Price : 590.0
View Details
Auction 2
Advertiser : blockchain
Price : 300.0
View Details
Expired

Profile

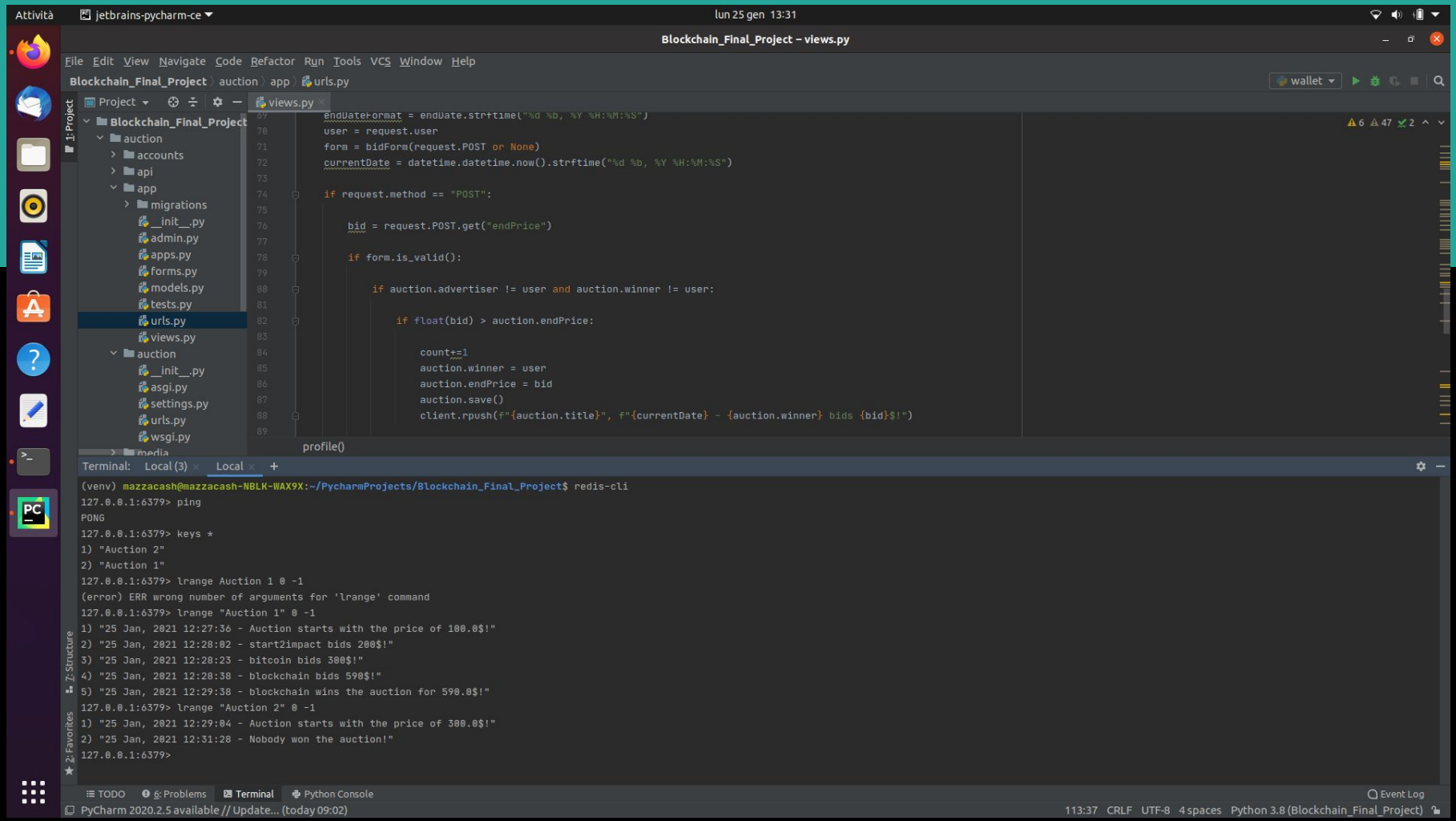
In this page you can watch informations about auctions won and published and generate a report of all the auctions completed on the site.


Attività
jetbrains-pycharm-ce
lun 25 gen 13:32
Blockchain_Final_Project – report2021_01_25_12_32_22.json
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Blockchain_Final_Project › auction › report › report2021_01_25_12_32_22.json
wallet
report2021_01_25_12_32_22.json
1 Project
2: Structure
2 Favorites
Terminal: Local (3) Local
TODO Problems Terminal Python Console Event Log
PyCharm 2020.2.5 available // Update... (today 09:02)
1:1 LF UTF-8 Tab* No JSON schema Python 3.8 (Blockchain_Final_Project)


Attività
Firefox Browser Web
lun 25 gen 13:32
BlockChain Auctions
127.0.0.1:8000/profile
Getting Started
modifica pagina
Altri segnalibri
BlockAuctions
Search for...
blockchain
Publish
Profile
Username : blockchain
Auctions Won :
{"Advertiser": "ethereum", "Winner": "blockchain", "Title": "Auction 1", "Text": "Auction Test 1", "startPrice": 100.0, "endPrice": 590.0, "startDate": "2021-01-25 12:27:36.281866+00:00", "endDate": "2021-01-25 12:29:36.280885+00:00"}
Auctions Published :
Do you want to download data about all the auctions completed on the site? Click on the below button and check the Json file in the report path of the project!
Generate Report!

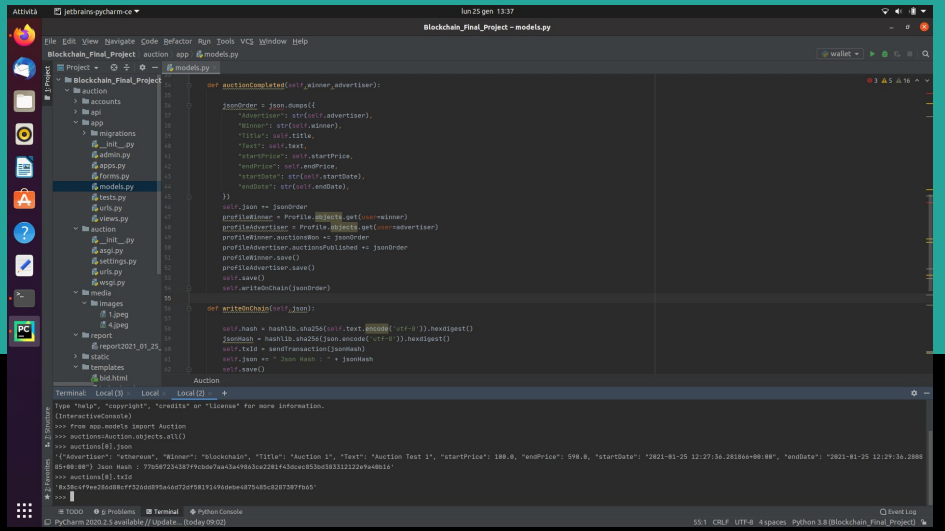
Redis

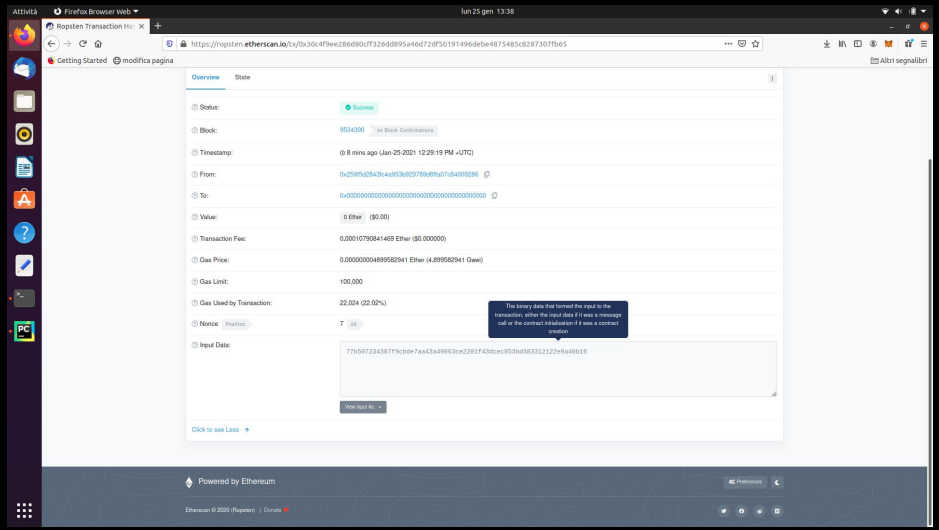
Informations about bids are stored in Redis so server can’t slow down in case of large influx of users.


Attività
jetbrains-pycharm-ce
lun 25 gen 13:31
Blockchain_Final_Project – views.py
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Blockchain_Final_Project 〉 auction 〉 app 〉 urls.py
wallet
Project
views.py
Blockchain_Final_Project
auction
accounts
api
app
migrations
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
auction
__init__.py
asgi.py
settings.py
urls.py
wsgi.py
media
endDateFormat = endDate.strftime("%d %b, %Y %H:%M:%S")
user = request.user
form = bidForm(request.POST or None)
currentDate = datetime.datetime.now().strftime("%d %b, %Y %H:%M:%S")
if request.method == "POST":
bid = request.POST.get("endPrice")
if form.is_valid():
if auction.advertiser != user and auction.winner != user:
if float(bid) > auction.endPrice:
count+=1
auction.winner = user
auction.endPrice = bid
auction.save()
client.rpush(f"{auction.title}", f"{currentDate} - {auction.winner} bids {bid}$!")
profile()
Terminal: Local (3) Local
(venv) mazzacash@mazzacash-NBLK-WAX9X:~/PycharmProjects/Blockchain_Final_Project$ redis-cli
127.0.0.1:6379> ping
PONG
127.0.0.1:6379> keys *
1) "Auction 2"
2) "Auction 1"
127.0.0.1:6379> lrange Auction 1 0 -1
(error) ERR wrong number of arguments for 'lrange' command
127.0.0.1:6379> lrange "Auction 1" 0 -1
1) "25 Jan, 2021 12:27:36 - Auction starts with the price of 100.0$!"
2) "25 Jan, 2021 12:28:02 - start2impact bids 200$!"
3) "25 Jan, 2021 12:28:23 - bitcoin bids 300$!"
4) "25 Jan, 2021 12:28:38 - blockchain bids 590$!"
5) "25 Jan, 2021 12:29:38 - blockchain wins the auction for 590.0$!"
127.0.0.1:6379> lrange "Auction 2" 0 -1
1) "25 Jan, 2021 12:29:04 - Auction starts with the price of 300.0$!"
2) "25 Jan, 2021 12:31:28 - Nobody won the auction!"
127.0.0.1:6379>
TODO 6: Problems Terminal Python Console Event Log
PyCharm 2020.2.5 available // Update... (today 09:02)
113:37 CRLF UTF-8 4 spaces Python 3.8 (Blockchain_Final_Project)

BlockChain Storing

When auction is completed, a script generates a JSON file with all the info about it and store relative hash in a transaction on the Ethereum Ropsten.


Attività
jetbrains-pycharm-ce
lun 25 gen 13:37
Blockchain_Final_Project – models.py
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Blockchain_Final_Project auction app models.py
wallet
Project
models.py
Blockchain_Final_Project
auction
accounts
api
app
migrations
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
auction
__init__.py
asgi.py
settings.py
urls.py
wsgi.py
media
images
1.jpeg
4.jpeg
report
report2021_01_25_
static
templates
bid.html
def auctionCompleted(self,winner,advertiser):
jsonOrder = json.dumps({
"Advertiser": str(self.advertiser),
"Winner": str(self.winner),
"Title": self.title,
"Text": self.text,
"startPrice": self.startPrice,
"endPrice": self.endPrice,
"startDate": str(self.startDate),
"endDate": str(self.endDate),
})
self.json += jsonOrder
profileWinner = Profile.objects.get(user=winner)
profileAdvertiser = Profile.objects.get(user=advertiser)
profileWinner.auctionsWon += jsonOrder
profileAdvertiser.auctionsPublished += jsonOrder
profileWinner.save()
profileAdvertiser.save()
self.save()
self.writeOnChain(jsonOrder)
def writeOnChain(self,json):
self.hash = hashlib.sha256(self.text.encode('utf-8')).hexdigest()
jsonHash = hashlib.sha256(json.encode('utf-8')).hexdigest()
self.txId = sendTransaction(jsonHash)
self.json += " Json Hash : " + jsonHash
self.save()
Auction
Terminal: Local (3) Local Local (2)
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from app.models import Auction
>>> auctions=Auction.objects.all()
>>> auctions[0].json
'{"Advertiser": "ethereum", "Winner": "blockchain", "Title": "Auction 1", "Text": "Auction Test 1", "startPrice": 100.0, "endPrice": 590.0, "startDate": "2021-01-25 12:27:36.281866+00:00", "endDate": "2021-01-25 12:29:36.288085+00:00"} Json Hash : 77b507234387f9cbde7aa43a49863ce2201f43decc853bd383312122e9a46b16'
>>> auctions[0].txId
'0x38c4f9ee286d88cff326dd895a46d72df5019149debe4875485c8287307fb65'
>>>
TODO Problems Terminal Python Console Event Log
PyCharm 2020.2.5 available // Update... (today 09:02)
55:1 CRLF UTF-8 4 spaces Python 3.8 (Blockchain_Final_Project)


Attività
Firefox Browser Web
lun 25 gen 13:38
Ropsten Transaction Hash
https://ropsten.etherscan.io/tx/0x38c4f9ee286d80cff326dd895a46d72df5019149debe4875485c8287307fb65
Getting Started
modifica pagina
Altri segnalibri
Overview
State
Status:
Success
Block:
9534390
14 Block Confirmations
Timestamp:
8 mins ago (Jan-25-2021 12:29:19 PM +UTC)
From:
0x259f5d2843fc4a953b829789d8ffa07c840009286
To:
0x0000000000000000000000000000000000000000
Value:
0 Ether ($0.00)
Transaction Fee:
0.000107909841469 Ether ($0.000000)
Gas Price:
0.000000004899582941 Ether (4.899582941 Gwei)
Gas Limit:
100,000
Gas Used by Transaction:
22,024 (22.02%)
Nonce
7
Input Data:
77b507234387f9cbde7aa43a49863ce2201f43decc853bd383312122e9a46b16
View Input As
Click to see Less
The binary data that formed the input to the transaction, either the input data if it was a message call or the contract initialization if it was a contract creation
Powered by Ethereum
Preferences
Etherscan © 2020 (Ropsten)
Donate

Conclusion

This is my auction site, I hope you enjoy it... Below that's my GitHub repository, thanks for watching it!

https://github.com/mazzacash94/Final-Project-BlockChain---Auction-Site