



CRYZELLE

Convenient Way to send Crypto

Our Team



Drew Herrera



Matthew Stream



Agnes Maria



Theena Dang

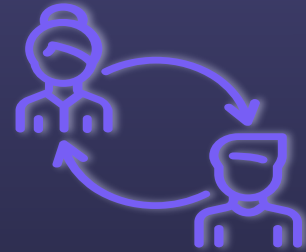


Nayana Narayanan



Hugo Velazquez

Overview



New to cryptocurrency? Want to send cryptocurrency to someone but you haven't memorized their 64 character hexadecimal wallet address? Have no fear Cryzelle is here! Using our system, registered users would only need to know the other person's phone number or email address and with a click of a button they can send or receive cryptocurrency instantly from each other.

Introduction

The movement of a cryptocurrency to an external wallet or another exchange via blockchain transaction using the user's Ethereum wallet registered phone number or email address as "destination address" which indicates where you would like to send your crypto. We are also incorporating Ganache with our project app and using AWS to create conversation that will turn into data input for transaction details.



Features



Registration

Phone number and email address



Send

Send payment to your (trusted) friend



Request

Requesting payment



Transaction History

Confirmed and past transaction



Loan

Borrow or lend capability using smart contract



Find a Friend

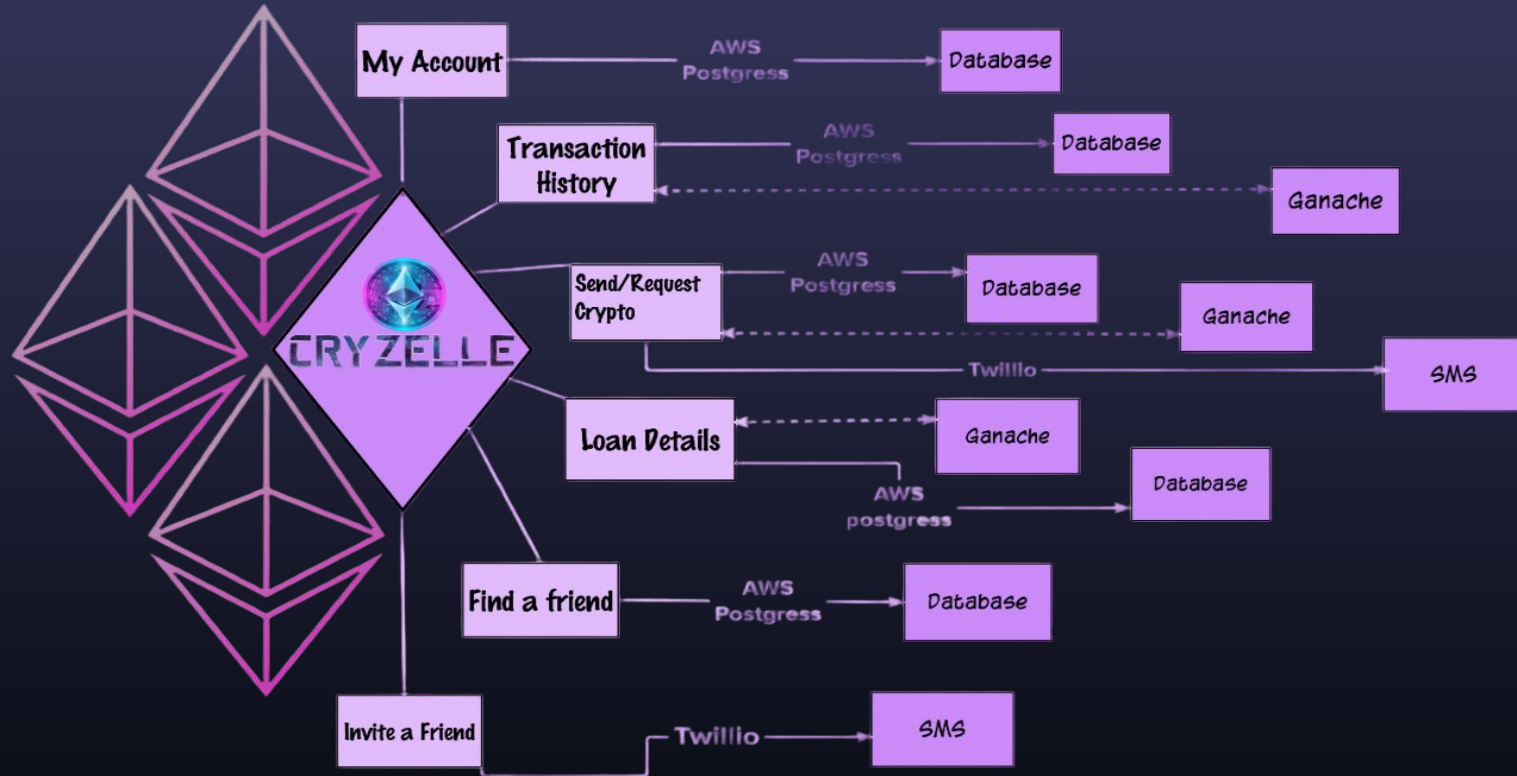
Find your trusted friend



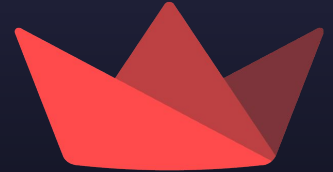
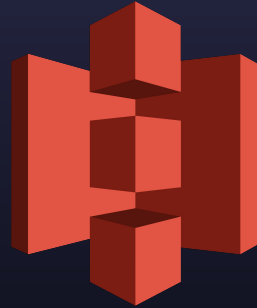
Invite a Friend

Referral Program

Flowchart



Technologies



Ganache Blockchain on AWS

```
from PIL import Image
import os
from dotenv import load_dotenv
import streamlit as st
import datetime as dt
from twilio.rest import Client
from .account import Account
from streamlit_authenticator.db_utils import get_default_active_user_wallet_by_phone
from web3.gas_strategies.time_based import medium_gas_price_strategy
from web3 import Web3
w3 = Web3(Web3.HTTPProvider('HTTP://52.87.239.109:8545'))
```

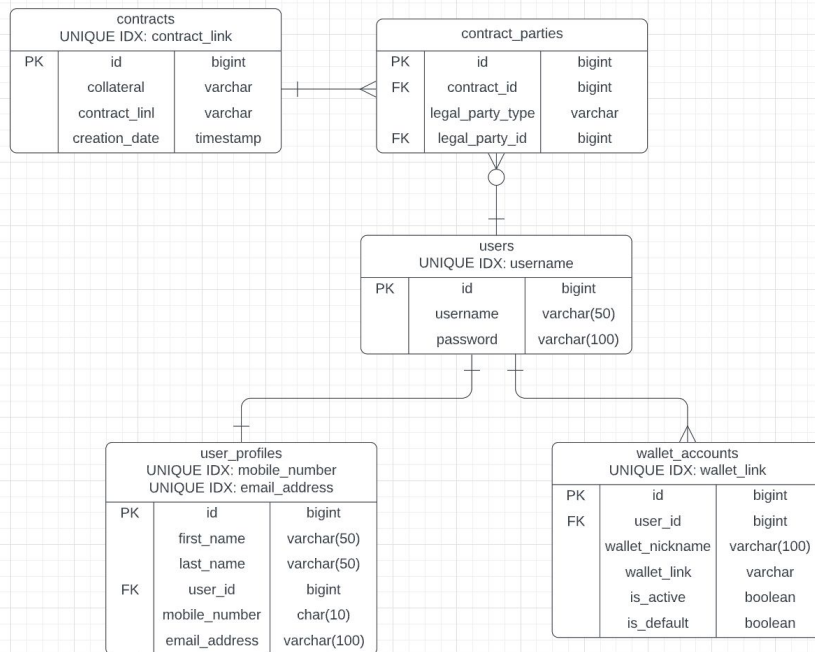
```
class Transaction:
    from_account = Account()
    to_account = Account()
    tx_amount = 0
    tx_date = dt.datetime

    def send(self, location: str = 'main'):
        # Image Logo
        image = Image.open('images/cryzelle1.jpeg')
        st.image(image)
        # Load .env environment variables
        load_dotenv()

        account_sid = os.environ['TWILIO_ACCOUNT_SID']
        auth_token = os.environ['TWILIO_AUTH_TOKEN']
        messaging_service_id = os.environ['TWILIO_MESSAGING_SERVICE_ID']
        client = Client(account_sid, auth_token)
        placeholder = st.empty()
        placeholder.empty()
        friend_information = None

        if location == 'main':
            st.markdown("# Send Crypto from a Friend")
            st.text("")
```


Postgres DB Model on AWS



Twilio Integration

```
import os
from dotenv import load_dotenv
from twilio.rest import Client
import streamlit as st
from streamlit_authenticator.db_utils import get_user_profile_by_phone

class SMS:
    def invite(self, control_name: str, location: str = 'main'):
        # Load .env environment variables
        load_dotenv()
        account_sid = os.environ['TWILIO_ACCOUNT_SID']
        auth_token = os.environ['TWILIO_AUTH_TOKEN']
        twilio_phone_number = os.environ['TWILIO_PHONE_NUMBER']
        client = Client(account_sid, auth_token)
        placeholder = st.empty()
        placeholder.empty()
```

Transaction History

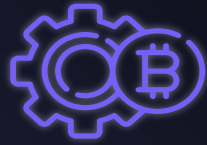
```
ubuntu@ip-172-31-82-124: ~  
eth_getBalance  
eth_getBalance  
eth_getBalance  
eth_getBalance  
eth_getTransactionCount  
eth_chainId  
eth_estimateGas  
eth_getBlockByNumber  
eth_getBlockByNumber  
eth_getBlockByNumber  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByHash  
eth_getBlockByNumber  
eth_chainId  
eth_sendTransaction  
  
Transaction: 0x04f2b18facae27d590c046584ba25bbb5c71b2846151d2224412dbfc351819d1  
Gas usage: 21000  
Block Number: 10  
Block Time: Thu Nov 24 2022 04:03:42 GMT+0000 (Coordinated Universal Time)  
[0] 0:node* "ip-172-31-82-124" 04:18 24-Nov-22
```

Demo of Cryzelle



<https://chorizoburrito.streamlit.app/>

Future Upgrades



- Change raw SQL queries into Object Relational Mapper.
- Adding multiple wallet addresses for an account.
- To show all the wallet transactions.
- IOUcoins and loans.
- User friendly system.
- Request/Send actionable items from their friends directly.

Thank you!

<https://github.com/drew94591/cryzelle.git>
<https://chorizoburrito.streamlit.app/>
<https://jollibeechicken.streamlit.app/>