

# 區塊鏈技術期末專題報告

---

Bonus Point System

紅利積分系統

資工三甲 李友升 09360724

資工三甲 傅崇浩 09360680

# 目錄

第壹章	動機 .....	3
第貳章	相關技術 .....	4
第壹節	Remix IDE.....	4
第貳節	MetaMask.....	4
第參節	Mumbai Test Net .....	5
第肆節	RSA Hash 值 .....	5
第參章	開發方法 .....	6
第壹節	撰寫及測試智能合約 .....	6
第貳節	撰寫及測試 Python UI 介面.....	6
第肆章	成果展示 .....	7
第伍章	結論 .....	8
第陸章	課堂回饋及心得 .....	9
李友升	.....	9
傅崇浩	.....	10
第柒章	文獻參考 .....	11

# 第壹章 動機

現在網路發達，不管是進行交易或是任何事都非常方便。但是也不免產生了許多的問題，例如交易糾紛等。因此在進行交易功能時，我們為了確保商家是否有給予正確的回扣點數，而建立了這個小系統，方便我們去觀察和監督。



## 紅利回饋折

---

每筆商品訂購金額

**100%回饋**

# 第貳章 相關技術

## 第壹節 Remix IDE

Remix 是一個開源的網絡編譯器和調試工具，用於開發以太坊智能合約。它提供了一個簡單而強大的界面，使開發人員能夠在瀏覽器中編寫、編譯和調試智能合約，來達成合約的部署。



## 第貳節 MetaMask

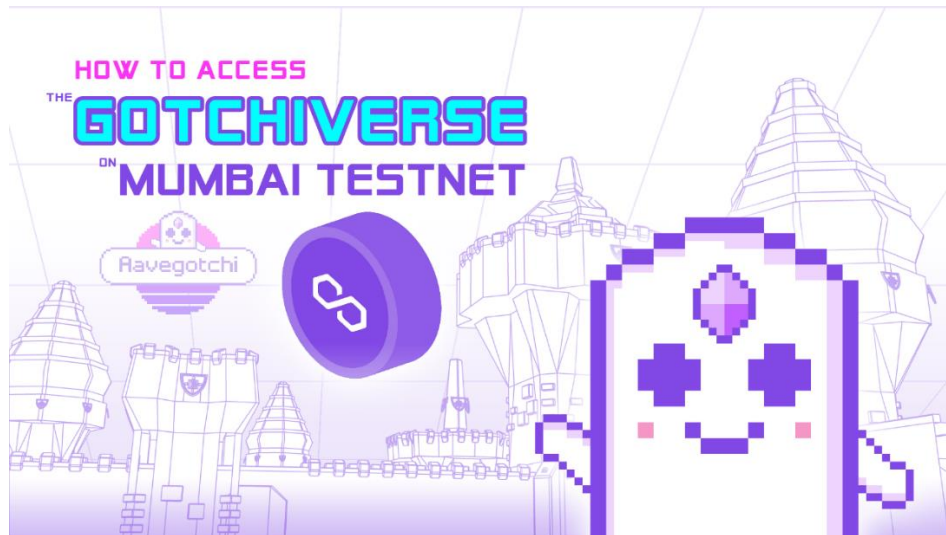
作為一個瀏覽器擴展程式（插件）的形式存在，提供了一個安全、便捷的方式來管理以太坊資產和與去中心化應用（DApps）進行交互。



# METAMASK

## 第參節 Mumbai Test Net

Mumbai Test Net 是一個用於開發和測試以太坊 DApps 和智能合約的測試網絡，它提供一個安全且可靠的環境，供開發人員進行實驗和測試，以確保他們的應用能夠在真實的以太坊網絡上正常運行。



## 第肆節 RSA Hash 值

在 RSA 演算法中，Hash 值的作用是對要簽名的數據進行摘要計算，以確保數據的完整性和不可否認性。Hash 值在 RSA 演算法中扮演著重要的角色，用於確保數據的完整性和驗證簽名的有效性。



# 第參章 開發方法

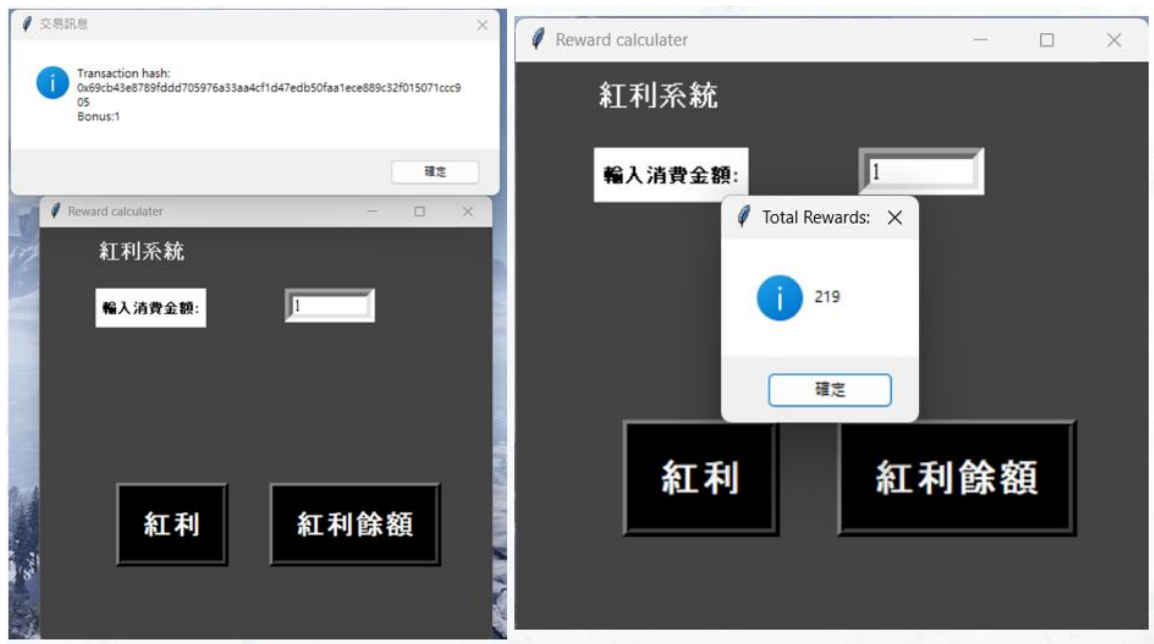
## 第壹節 撰寫及測試智能合約

```
1
2  pragma solidity 0.5.17;
3
4  contract RewardsCalculator {
5      address public Server;
6      address public Costomer;
7      uint public Bonus;
8      uint public TotalRewards;
9      // constructor function
10     constructor(address costomer) public {
11         Server = msg.sender;
12         Costomer = costomer;
13         TotalRewards = 0;
14     }
15     // call this function to add miles
16     function AddBonus(uint cost) public {
17         Bonus = cost;
18         ComputeTotalRewards();
19     }
20     function ComputeTotalRewards() private {
21         TotalRewards += Bonus;
22     }
23 }
24
```

## 第貳節 撰寫及測試 Python UI 介面

```
1 from tkinter import *
2 from tkinter import messagebox
3 from web3 import Web3
4 import json
5
6 def send_transaction_to_contract():
7     try:
8         cost = int(ENTRY1.get())
9         except ValueError:
10             messagebox.showinfo("提示", "請輸入正確金額!")
11     else:
12         transaction = contract.functions.AddBonus(cost).build_transaction({
13             'from': server,
14             'gas': 2000000,
15             'gasPrice': web3.to_wei('40', 'gwei'),
16             'nonce': web3.eth.get_transaction_count(server),
17         })
18         signed_transaction = web3.eth.account.sign_transaction(transaction, private_key)
19         transaction_hash = web3.eth.send_raw_transaction(signed_transaction.raw_transaction)
20         bonus = cost
21         messagebox.showinfo("交易信息", "Transaction hash: " + str(transaction_hash.hex()) + "\n" + "Bonus: " + str(bonus))
22
23 def get_contract_data():
24     total_rewards = contract.functions.TotalRewards().call()
25     messagebox.showinfo("Total Rewards", str(total_rewards))
26
27 if __name__ == "__main__":
28     private_key = "732140c96442695643d2dfe0b4e082770b1d4a20d0c4492d99d6ff19"
29     contract_address = "0x047694f03b7894177b01470d4e3bEF3af21"
30     rpc_url = "http://rpc.mimblex.com"
31     web3 = Web3(Web3.HTTPProvider(rpc_url))
32     abi = json.loads([{"inputs": [{"internalType": "address", "name": "costomer", "type": "address"}], "stateMutability": "nonpayable", "type": "constructor"}, {"inputs": [{"internalType": "uint256", "name": "cost", "type": "uint256"}], "name": "AddBonus", "outputs": [], "stateMutability": "nonpayable", "type": "function"}, {"inputs": [], "name": "TotalRewards", "outputs": [{"internalType": "uint256", "name": "TotalRewards", "type": "uint256"}], "type": "function"}])
33     contract = web3.eth.contract(address=contract_address, abi=abi)
34     server = contract.functions.Server().call()
35     costomer = contract.functions.Costomer().call()
36     TOP = Tk()
37     TOP.title("Reward Calculator")
38     TOP.configure(background="#f4f4f4")
39     TOP.resizable(0, 0)
40     LABEL = Label(TOP, bg="#f4f4f4", fg="#ffffff", text="紅利系統", font=("Helvetica", 15, "bold"), pady=10)
41     LABEL.place(x=5, y=5)
42     LABEL2 = Label(TOP, bg="#ffffff", text="輸入消費金額:", bd=1, font=("Helvetica", 10, "bold"), pady=5)
43     LABEL2.place(x=5, y=40)
44     ENTRY1 = Entry(TOP, bd=1, width=40, font="Roboto 11")
45     ENTRY1.place(x=240, y=40)
46     BUTTON = Button(TOP, bg="#000000", fg="#ffffff", bd=1, text="支付", padx=10, pady=10, command=send_transaction_to_contract, font=("Helvetica", 20, "bold"))
47     BUTTON.grid(row=4, column=0, sticky=W)
48     BUTTON2 = Button(TOP, bg="#000000", fg="#ffffff", bd=1, text="查詢總額", padx=10, pady=10, command=get_contract_data, font=("Helvetica", 20, "bold"))
49     BUTTON2.grid(row=5, column=0, sticky=W)
50     BUTTON2.place(x=225, y=250)
51
52 TOP.mainloop()
```

# 第肆章 成果展示



Contract 0xcc4769A4F0367d884177b041A7cd4E3bEF5Afa21

Contract Overview

Balance: 0 MATIC

More Info

My Name Tag: Not Available

Contract Creator: 0xa4f27b416b85232965... at txn 0xce2491445e5d53b407...

Transactions ERC-20 Token Txns Contract Events

Latest 7 from a total of 7 transactions

Txn Hash	Method	Block	Age	From	To	Value	[Txn Fee]
0xa012a6d5258ba2982f...	Add Bonus	36789498	1 day 23 hrs ago	0xa4f27b416b85232965...	IN 0xcc4769a4f0367d8841...	0 MATIC	0.001265
0x7c38454694cd2842ba...	Add Bonus	36758550	2 days 17 hrs ago	0xa4f27b416b85232965...	IN 0xcc4769a4f0367d8841...	0 MATIC	0.001265
0x7cbaf8b9a1f48a1b02f...	Add Bonus	36758540	2 days 17 hrs ago	0xa4f27b416b85232965...	IN 0xcc4769a4f0367d8841...	0 MATIC	0.001153
0x7a56b1023367acec13...	Add Bonus	36758534	2 days 17 hrs ago	0xa4f27b416b85232965...	IN 0xcc4769a4f0367d8841...	0 MATIC	0.001265
0x4d48d5ec6b1bcdcb9...	Add Bonus	36756508	2 days 18 hrs ago	0xa4f27b416b85232965...	IN 0xcc4769a4f0367d8841...	0 MATIC	0.001265
0xb90b6d497695f0f8d3...	Add Bonus	36756400	2 days 18 hrs ago	0xa4f27b416b85232965...	IN 0xcc4769a4f0367d8841...	0 MATIC	0.000164562501
0xce2491445e5d53b407...	0x60806040	36756378	2 days 18 hrs ago	0xa4f27b416b85232965...	IN Contract Creation	0 MATIC	0.000553167503

## 第伍章 結論

這次的期末專案雖然略顯簡陋，成效卻十分明瞭，而這專案著重於監督交易的部分。

利用雜湊函數可以幫助我們避免交易上的衝突，也可防止商家沒有給予合理的回扣。

雖然本次專案簡單卻也將此學期的所學(RSA、智能合約和 Python 串接)都有使用，並且我們是用 Mumbai 的測試幣去進行合約簽署。總結來說這學期也學到許多有用且有趣的知識。





# 第陸章 課堂回饋及心得

李友升

這學期知曉到以前只聽聞卻從未接觸過的新知-區塊鏈。從學期初的 RAS 演算法，私密鑰加解密在程式上的應用，還有學期末的以太坊和私網挖礦，這些知識都讓我感到非常新奇及有趣。

而在這次的期末專案中，我們建立了一個小系統，是與紅利積分有些相關，不僅讓我再次了解虛擬網路與虛擬貨幣間的關係，還能夠親自去測試利用智能合約去達成交易。也很感謝王家輝老師不遺餘力地教導我們，分享很多額外的知識，想讓我們能夠更加瞭解區塊鏈的一切。

## 傅崇浩

從很久之前就已經聽過區塊鏈這個東西了，但是一直沒有了解這方面的知識，例如 NFT，很多網紅都在發布 NFT，但是我完全不了解甚麼是 NFT，所以剛好看到學校有這門課，想在這門課多了解一點區塊鏈。在修完這門課後，學到了很多區塊鏈的知識，例如：區塊鏈的加密、區塊鏈為什麼能夠去中心化、怎麼挖礦、智能合約、發行 NFT。在本次課程中，感覺比較難的是使用 geth 測試網，雖然我的作業都有做完，但是在使用 geth 的時候，很多出錯都不知道是甚麼原因，都需要自己一直重複嘗試，常常都要做到下午第一堂課前才離開。在本學期中，最喜歡的應該是 MetaMask，一直很想買看看、玩看看虛擬貨幣還有發行 NFT。

# 第柒章 文獻參考

## 資料網址

Frequent Flyer Rewards Calculator :

<https://github.com/Azure-Samples/blockchain/blob/master/blockchain-workbench/application-and-smart-contract-samples/frequent-flyer-rewards-calculator/readme.md>

交易紀錄 :

<https://mumbai.polygonscan.com/address/0xcc4769A4F0367d884177b041A7cd4E3bEF5Afa21>