NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Blockchain and Its Application

Final Year Project Interim Report

Submitted by: Chan De Wei
Matriculation Number: U2020300E
Project Number: A3048-231

Supervisor: Assoc Prof Chua Hock Chuan

School of Electrical & Electronic Engineering

A final year project report presented to the Nanyang Technological University
in partial fulfillment of the requirements of the degree of
Bachelor of Engineering

2023

Table of Contents

1 INTRODUCTION

1.1 Background

In the rapidly evolving landscape of digital finance, there is a need to address the limitations and challenges of current payment systems. As the adoption of various digital payment methods continues to rise, users encounter a myriad of issues that impede the efficiency, security, and overall user experience of these systems, especially during cross-border transaction. [1]

In the realm of traditional banking, intermediaries like banks are always involved, potentially leading to high transaction fees, delays, and security vulnerabilities. [2] To address these issues, cryptocurrencies used in blockchain technology were introduced. Blockchain is a permanent, shared and distributed ledger for recording transactions and tracking assets in a business network.[3] However, while its native cryptocurrencies, such as Bitcoin and Ethers offer decentralization and security, they are plagued by high volatility, rendering them unsuitable for everyday transactions. Additionally, cryptocurrency exchanges are known for their complexity and the security risks associated with centralized systems. For instance, a centralized exchange, FTX filed for bankruptcy due to a liquidity crisis. [4] This serves as a reminder that people needs decentralized alternatives that offer greater control, transparency and security. Hence, Central Bank Digital Currencies (CBDCs) that provide secure and efficient way to make digital transactions could serve as a decentralized alternative. [5]

CBDCs are essentially digital versions of a country's official currency, issued and regulated by the central bank. According to key finding from Atlantic Council, there are 130 countries, representing 98 percent of global GDP, are exploring a CBDC. 19 of the G20 countries are now in the advanced stage of CBDC development and 11 countries have fully launched their CBDC. They aim to provide a secure and efficient way for people to make digital transactions, similar to using physical cash. [5] However, some countries, like the United States, may prefer stablecoins over CBDCs. [6] In fact, stablecoins do not differ much from CBDCs, and the major difference is that CBDCs are distributed by the central bank, whereas stablecoins are issued by private institutions. Therefore, both digital coins should be considered for adoption in the future payment system.

As a well-known fact, e-commerce heavily relies on payment system.[7] While the development of Central Bank Digital Currency (CBDC) payment systems is underway, e-commerce could begin enhancing digital transactions in terms of speed, cost, and transparency by integrating blockchain payment systems into the current infrastructure.

In addition to above, there is another payment method that is often overlooked but widely used in various application, in particularly the e-commerce platform, which is reward or voucher. However, the voucher that users earn are scattered across different platforms due to their spending on different e-commerce platforms. This result in a fragmented user experience and complicating financial management. As the CBDC payment system is being developed, e-commerce platform users could redeem the voucher through the app as Purpose Bound Money (PBM) and store it. PBM refers to digital currencies designated for specific purposes or industries. For instance, Monetary Authority of Singapore (MAS) has created PBM to digitalise voucher, leading to more efficient and affordable services with better user experiences.[8] Once purpose bound money is spent, the logic programmed into it will be released and the status of transferred token will return back to CBDC. Hence, this solution allows for fast and cost-effective settlement but it only can be fully realized after the development of CBDC payment system is complete.

1.2 Literature Review

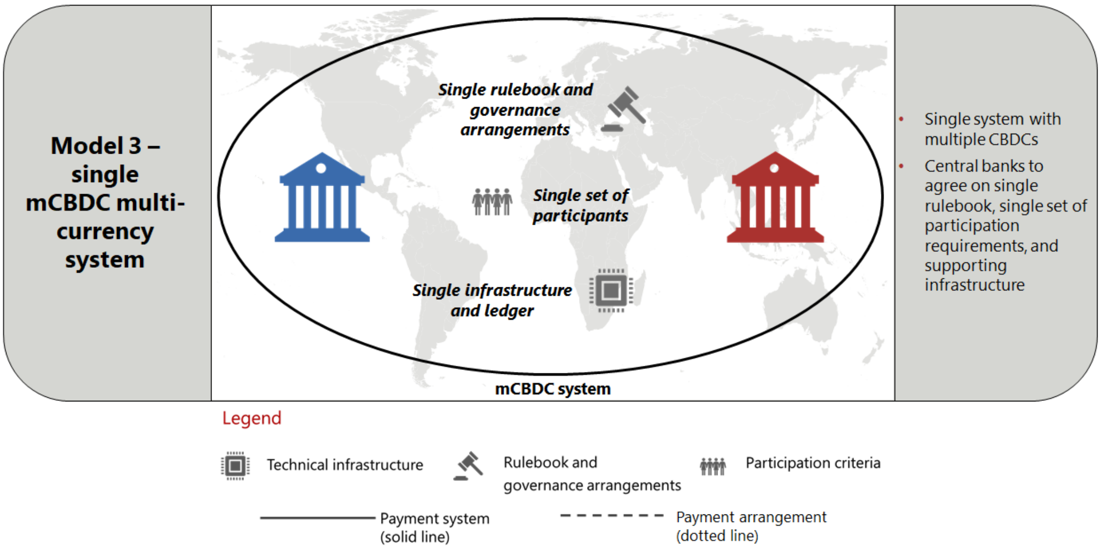
This section is to provide comprehensive analysis of existing research and relevant studies.

The first paper under reviewed, Project Dunbar [2], is aimed to explore using multi-CBDCs (mCBDCs) to manage cross-border transaction in 4 countries, including Singapore. mCBDCs are referred as CBDCs of different countries. In this paper, three different mCBDC designs are presented, and the third model (model 3) is chosen as their design since this model has the capability to connect all the participating banks and utilize a shared platform for international settlements.

However, it should be noted that this model is designed to handle wholesale CBDC transaction, which are typically characterized by giant values and low-frequency use. It is not designed to handle retail CBDC transaction, which are characterized by low values and high volumes.

Retail CBDC allows citizens to hold accounts directly with the central bank and Central bank will directly distribute CBDC without intermediaries. Therefore, wholesale CBDC do not align with the requirements of e-commerce.

Despite this mismatch, this model still can be adopted and improved due to the advantages of no intermediaries involved except central banks and allowance of multiple currencies within single network. In fact, similar models have been proposed in other paper as well, such as Jasper – Ubin Design Paper [9]. As such, this approach could potentially improve the speed, cost and transparency. From a technical aspect, this Corda network that uses permissioned blockchain, meaning that only authorized parties can join the network. Hence, this constrains its applicability within the context of e-commerce.



Model 3 – single mCBDC multi-currency system
Single rulebook and governance arrangements
Single set of participants
Single infrastructure and ledger
mCBDC system
Single system with multiple CBDCs
Central banks to agree on single rulebook, single set of participation requirements, and supporting infrastructure
Legend
Technical infrastructure
Rulebook and governance arrangements
Participation criteria
Payment system (solid line)
Payment arrangement (dotted line)

Figure 1: Single mCBDC multi-currency system proposed in paper of Project Dunbar

According to the Atlantic Council Digital Currency Tracker, Monetary Authority of Singapore has primarily participated in project related to wholesale CBDC only. [5] Furthermore, MAS accessed in 2021 that there was no immediate need for retail CBDCs in Singapore due to governance considerations and existence of SWIFT payment system.[10] In fact, Project Nexus[11], in which MAS has actively participated, aimed at addressing retail cross-border transaction across different countries by establishing a global transaction standard. It introduces an additional alternative, but it's important to note that this initiative does not involve the use of CBDCs. More importantly, the paper published by the Project Nexus team mentions that it is not feasible to mandate every payment system follows this standard, but infrastructure using blockchain, such as CBDC is still allowed to connect with Nexus platform. This provides more choices for user fast and cheap cross-border transaction.

As of March of 2023, Project Icebreaker [12], involving banks from other countries, has begun exploring about the use of retail CBDC to improve cross-border transaction. One participating banks, Norges Bank has even published its code in Github[13], showing that the use of Hardhat framework, Solidity language and Ethereum Public Network. Ethereum's compatibility with public retail CBDC transactions makes it a suitable choice, indicating the potential for further exploration of retail CBDCs as a decentralized alternative for instant cross-border transactions in Singapore.

The second paper under reviewed is Project Orchid [8]. Project Orchid mainly focuses on development of PBM in Singapore. The goal of the project is to digitalize voucher, making the settlement be cheaper and time-saving. Meanwhile, PBM allows various institutions to organize campaign and issue their own voucher for participant to spend on certain products, minimizing the need for system reconfiguration when new campaigns are launched.

In this paper, it is clarified that the PBM token is produced by wrapping the conditional logic on digital form of SGD (DSGD) before distribution. PBM token can only be redeemed back to DSGD if the preset conditionals are met or contract has expired. Therefore, PBM token are issued in advance of issuing DSGD, representing the CBDC of Singapore.

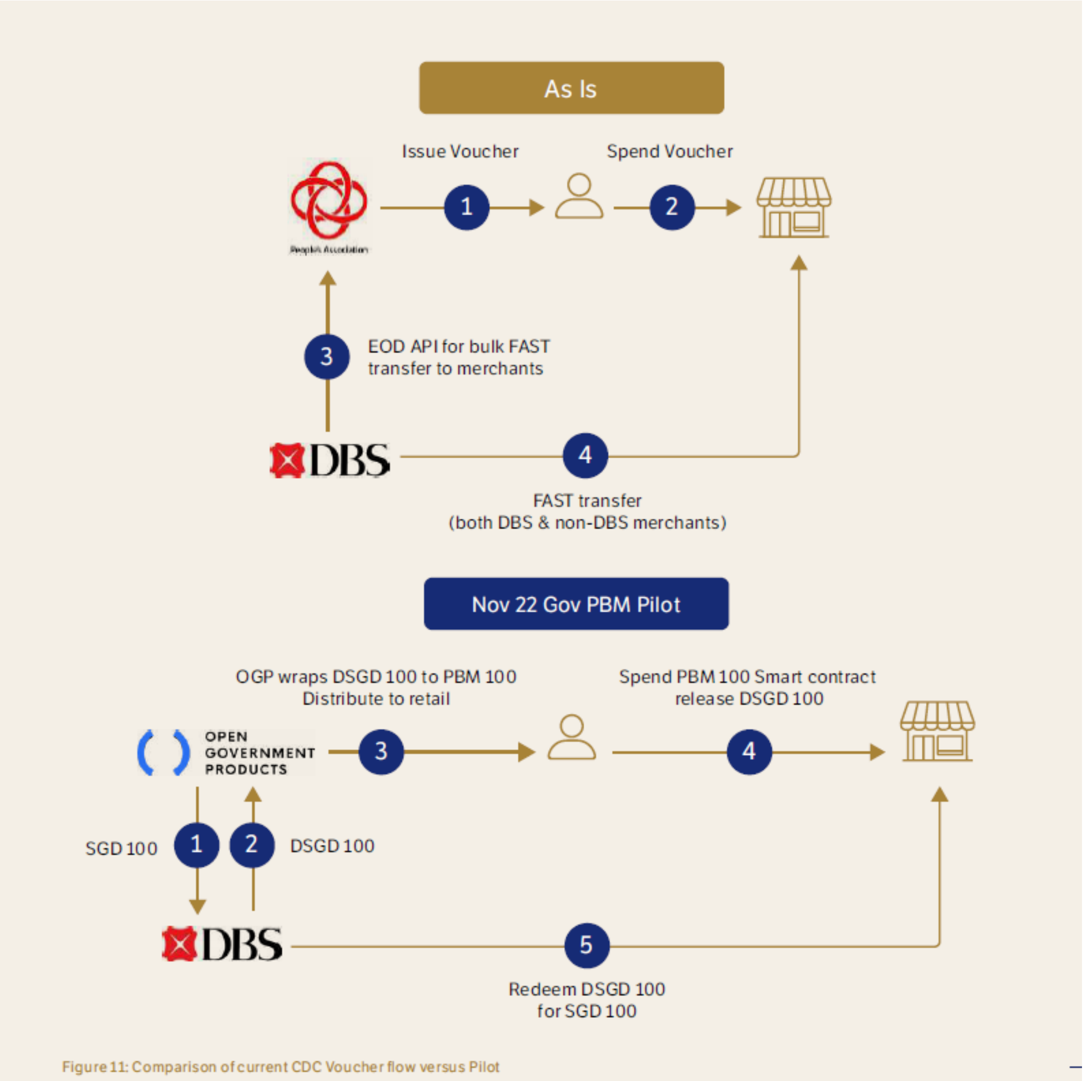

As Is
Issue Voucher
Spend Voucher
People's Association
1
2
3
EOD API for bulk FAST transfer to merchants
DBS
4
FAST transfer (both DBS & non-DBS merchants)
Nov 22 Gov PBM Pilot
OGP wraps DSGD 100 to PBM 100 Distribute to retail
Spend PBM 100 Smart contract release DSGD 100
OPEN GOVERNMENT PRODUCTS
3
4
SGD 100
1
2
DSGD 100
DBS
5
Redeem DSGD 100 for SGD 100
Figure 11: Comparison of current CDC Voucher flow versus Pilot

Figure 2: Comparison of government-issued voucher (CDC voucher) versus PBM

Besides, another PBM technical paper is published by MAS to demonstrate their concept as well.[14] During developing PBM, this report suggests considering permissionless networks, such as Ethereum or Polygon Network since e-commerce should be publicly accessible as well. It also mentions that DSGD token standard to be used is a fungible ERC-20, while the PBM token standard should be a semi-fungible ERC-1155 due to their different purposes, which will be discussed later in this report. Solidity Programming Language and Hardhat framework are chosen is because they are compatible with Ethereum Virtual Machine (EVM) and enable the development of business logic in smart contract.[15]

Besides, the selection of Polygon Mumbai Testnet is motivated by zero cost in testing smart contract deployment, as it offers fake native assets, such as MATIC for testing purposes whereas Ethereum Mainnet requires real Ether to spend on testing smart contract. The native token also not allowed to be transferred between networks.

In the context of e-commerce, the utilization of various vouchers is a common practice. In the past, e-commerce platforms relied on traditional databases to manage the creation, modification, and deletion of voucher configurations.[16] However, with the integration of blockchain technology, terms and regulations can now be encoded within smart contracts, ensuring immutability until the campaign concludes. This implementation holds the potential to enhance transparency and can later seamlessly integrate with retail CBDC systems.

1.3 Objective and Scope

This project aims to develop retail CBDC cross-border payment system and integrate their use of CBDC in e-commerce payment. This project includes developing Oracle that enables smart contract to pull foreign exchange rate from external source through HTTP request. In addition to that, wallet application will be designed to hold the CBDCs and facilitate transactions for product available on e-commerce platforms. Last but not least, this wallet will possess the functionality to redeem and store voucher which issued not only by government entities but also from e-commerce platforms.

The tech tools that taking into consideration to develop the backend are listed below with their functions.

Tools	**Functions**
Solidity	Smart Contract Programming Language
Hardhat	Backend Framework, including local development network and testing
Hardhat-Ethers	Plug-in that brings EtherJS Library into Hardhat for interacting with Ethereum blockchain and its ecosystem, such as deployment of smart contract
Alchemy	Web3 Application Programming Interface (API) Provider for interacting with deployed smart contract, monitoring health of DApps
NodeJS with Axios and Express	Connection between backend and frontend using HTTP request
Openzeppelin	Smart Contracts for various standards, such as ERC-20, ERC-721, ERC1155, Ownable, Access Control and so on
Chainlink	Oracle development to fetch data from outside world to decentralized Smart Contract

React with Ant Design UI Library	Frontend Development
Wagmi	Collections of React Hook for Ethereum
Remix	Integrating Development Environment (IDE) for developing, quick testing smart contract and debugging before being deployed to network
VSCode	Integrating Development Environment (IDE) for working on frontend development and connections

Table 1: Technology Tool Stacks for DApps Development and Testing

After developing and testing, the smart contracts will be deployed in Polygon Mumbai Testnet. Besides, React and Wagmi will be used to develop the front-end to allow user interact with the smart contract.

1.4 Project Timeline

No	Task	Date
1.	Project Planning	16 to 31 Aug (2 weeks)
2.	Literature Review/ Blockchain Learning/ Mini Decentralized Applications (DApps) Development	31 Aug to 10 Oct (6 weeks)
3.	Emulation of code as presented in Literature Review/ Smart Contract Design for Prototype	10 Aug to 25 Oct (2 weeks)
4.	Prototype Development	25 Oct to 8 Nov (2 weeks)
5.	Smart Contracts Design	8 Nov 2023 to 3 Jan 2024 (8 weeks)
6.	User Interface (UI) Redesign	3 Jan to 7 Feb 2024 (5 weeks)
7.	Code Testing and Cleaning	7 Feb to 27 Mar 2024 (7 weeks)
8.	Code Finalization	27 Mar to 8 May 2024 (6 weeks)

Table 2: Project Timeline

2 WORK CONDUCTED

2.1 Blockchain Learning

2.1.1 Online courses - Bitcoin, Blockchain and Cryptoassets, and Smart Contract and Decentralized Finance

In order to learn the basics of blockchain, I registered for two online courses offered by the University of Basel. In the course titled "Bitcoin, Blockchain, and Cryptoassets," I gained knowledge on fundamental blockchain structure, payment systems, hash algorithms, and consensus protocols. The second course, "Smart Contract and Decentralized Finance," provided me with insights into smart contract programming, distinctions between Ethereum and Bitcoin, foundational Ethereum concepts, an overview of DeFi, and the concept of tokenization.

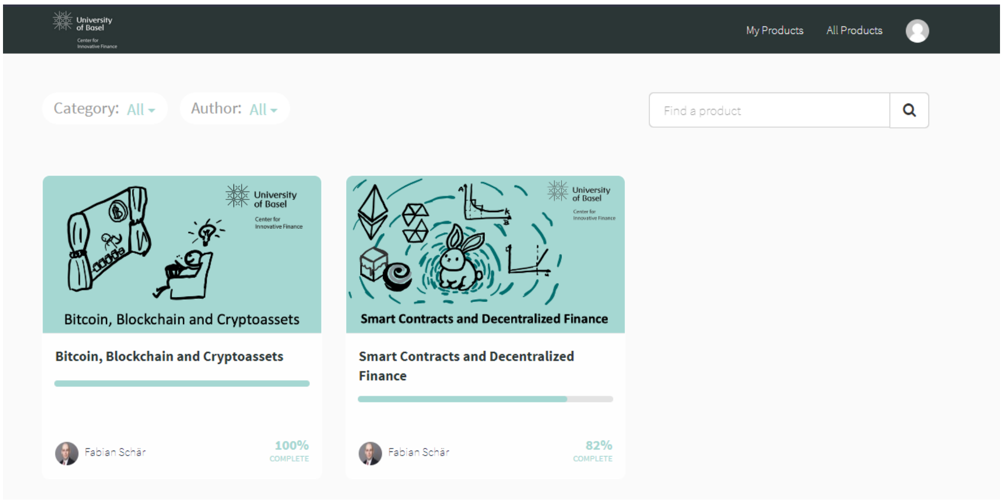
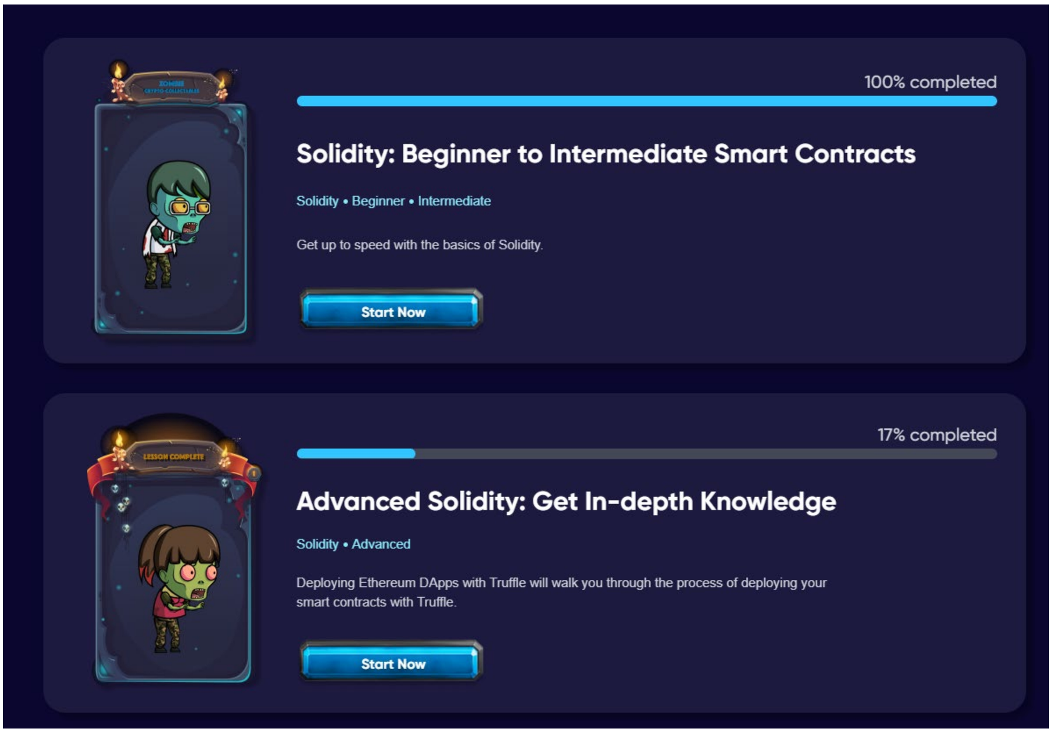

University of Basel
Center for Innovative Finance
My Products
All Products
Category: All
Author: All
Find a product
Bitcoin, Blockchain and Cryptoassets
Bitcoin, Blockchain and Cryptoassets
Fabian Schär
100% COMPLETE
Smart Contracts and Decentralized Finance
Smart Contracts and Decentralized Finance
Fabian Schär
82% COMPLETE

Figure 3: Progress of the Bitcoin, Blockchain and Cryptoassets, and Smart Contract and Decentralized Finance

2.1.2 Cryptozombies

To acquire hands-on experience in developing DApps with Solidity for smart contracts, I decided to enroll in this particular course. The course focused on creating basic games. Unfortunately, I faced a challenge when I discovered that the course was guiding students to use Truffle instead of Hardhat and VueJS instead of ReactJS. Ultimately, I did not end up finishing this course.



100% completed

Solidity: Beginner to Intermediate Smart Contracts

Solidity • Beginner • Intermediate

Get up to speed with the basics of Solidity.

17% completed

Advanced Solidity: Get In-depth Knowledge

Solidity • Advanced

Deploying Ethereum DApps with Truffle will walk you through the process of deploying your smart contracts with Truffle.

Figure 4: Progress of Cryptozombies

2.2 Mini Decentralized Applications (DApps) Development

I learned to build three DApps to become familiar with the development cycle. The table below shows the DApps Name, code repository and its description.

DApps Name	**Description**
Petshop[17], [18]	Allows customers to adopt pets
E-Voting[18], [19]	Allows citizens to vote for elections
Online Movie Ticket System[20]	Allows people to buy tickets online

Table 3: DApps that I learned to develop

Each of these DApps require smart contracts utilizing the Truffle framework and NodeJS for development. Every user-initiated action incurs gas fee for transaction processing. Metamask wallet is integrated here to sign the transaction on behalf of users using the fake account that Ganache network generated, and this provides users with an overview of the transactions they made, including real-time gas fees.

2.3 Emulation of PBM

In my efforts to comprehend the development of PBM, I attempted to execute the code provided by Project Orchid but encountered persistent challenges when attempting deployment on the Polygon Mumbai Testnet.[21], [22]

I soon recognized the presence of multiple issues, including JSON-RPC response errors, contract verification failures, and discrepancies in environmental parameter placement. Resolving these issues required approximately a week of debugging, as I needed to familiarize myself with an additional framework, Hardhat, which was utilized in this particular project.

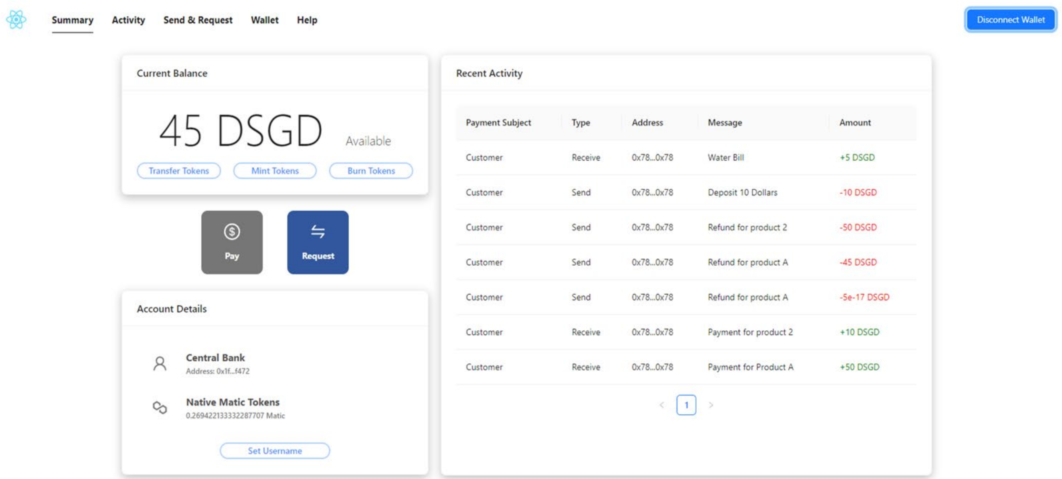
To provide a visual representation of the process and the associated transaction address, I created a final flowchart, accessible via [23]. Please note that to view this SVG file, it should be downloaded in its raw format and access to the Polygon Mumbai Testnet may not be possible through NTUSecure WIFI Network.

Furthermore, it is worth to highlight that Project Orchid only applies ERC-20 Token Standard for both DSGD and PBM [21]. However, the technical paper of PBM published by MAS[14], introduces an innovative approach to extend the functionality of PBM by adopting hybrid approach while retaining ERC-20 as DSGD. This is because PBM is considered to be customized and programmable as well as distributed through various campaigns. As a result, ERC-1155 is taken into consideration to develop the standard of PBM. Ultimately, Open Government Product (OGM), a government technology agency in Singapore, has made the final PBM token standard available in Github.[24]

2.4 Prototype Implementation – Wallet Application

To illustrate the application of CBDC in e-commerce, I built a decentralized wallet DApp using DSGD. The initial inspiration for creating this DApp came from an online tutorial, where my aim was to replicate the functionality of a PayPal wallet. However, I discovered that it also enabled users to request payments, which sparked the idea of implementing this feature in e-commerce transactions. This enhancement enhances the payment process for merchants, as they can ship products only once the user has made the requested payment. The tokens used in this application are sourced from a separate contract that issues Singapore's CBDC, DSGD. This practical implementation aligns with the project's scope.

Through this prototype, I gained valuable experience in Smart Contract Development for payment system and wallet although I encountered various challenges associated with different frameworks. Additionally, I found out there are excellent UI libraries that support React simplifying frontend development. For instance, I improved the frontend by incorporating features such as spinners to enhance the user experience while waiting for transactions to complete, making the app more user-friendly.



Current Balance

45 DSGD Available

Account Details

Central Bank
Address: 0x1f...f472

Native Matic Tokens
0.269422133332287707 Matic

Recent Activity

Payment Subject	Type	Address	Message	Amount
Customer	Receive	0x78...0x78	Water Bill	+5 DSGD
Customer	Send	0x78...0x78	Deposit 10 Dollars	-10 DSGD
Customer	Send	0x78...0x78	Refund for product 2	-50 DSGD
Customer	Send	0x78...0x78	Refund for product A	-45 DSGD
Customer	Send	0x78...0x78	Refund for product A	-5e-17 DSGD
Customer	Receive	0x78...0x78	Payment for product 2	+10 DSGD
Customer	Receive	0x78...0x78	Payment for Product A	+50 DSGD

Figure 5: Wallet Application Home Page

The figure above displays landing page of the basic wallet application. This wallet comprises four key components, “Request and Pay”, “Current Balance”, “Account Details” and “Recent Activity”. These components serve to illustrate the local transfer e-commerce payment process. Within this payment cycle, three distinct roles are involved. They are the Central Bank who responsible for issuing DSGD, the Merchant who initiates payment requests and the user (Customer) who fulfill these requests by making payments. Thus, the current prototype empowers users to execute DSGD transfers, create payment requests, and complete payments for those requests.

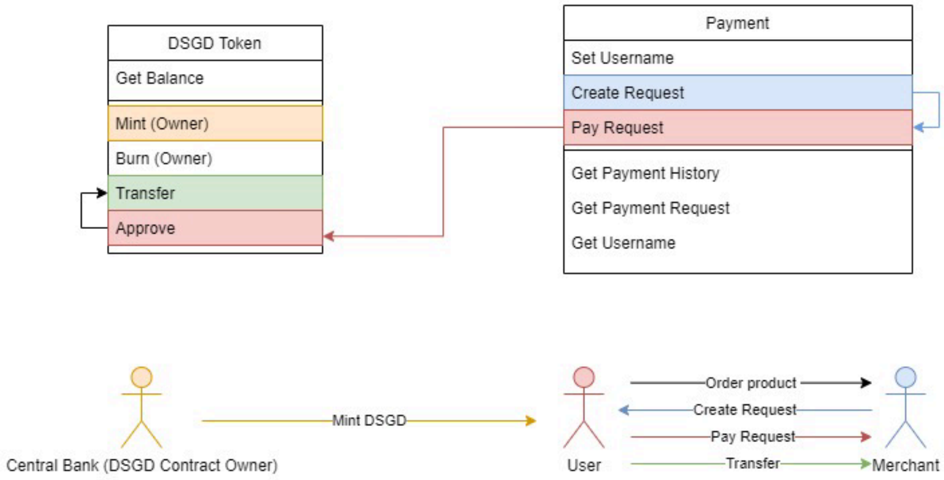
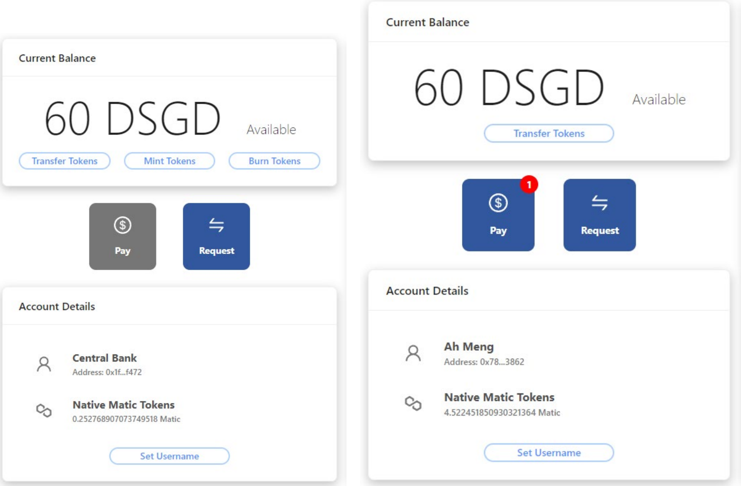


Figure 6: Flow Chart of Simple Payment through Wallet

The tables in the flow chart above show DSGD Token and Payment Smart Contract that place the backend logic. The color of the column indicates the party that initiates the transaction.


Current Balance

60 DSGD Available

Central Bank
Address: 0x1f...f472

Native Matic Tokens
0.252768907073749518 Matic

Current Balance

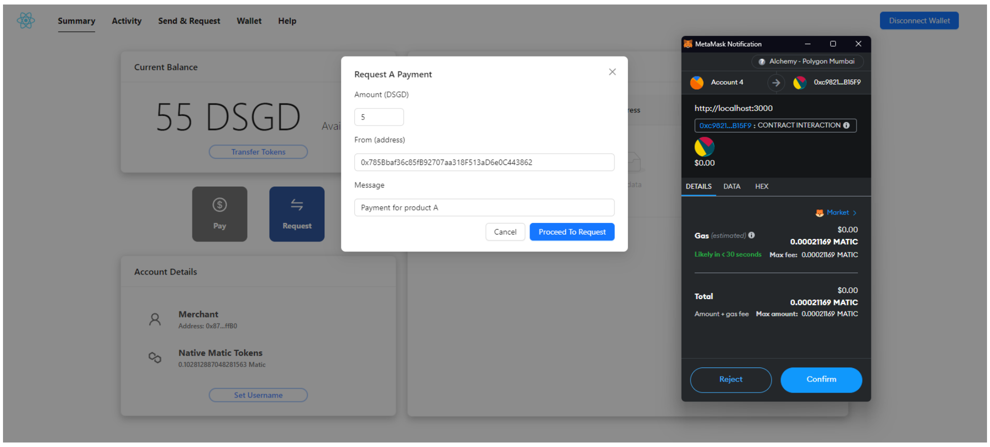
60 DSGD Available

Ah Meng
Address: 0x78...3862

Native Matic Tokens
4.522451850930321364 Matic

Figure 7: User Interface of Central Bank’s Wallet (Left) and Citizen’s Wallet (Right)

The figure above shows that the authority to mint and burn tokens is exclusively held by the Central Bank. The red badge on the pay button signifies that the user has outstanding payment requests that remain unfulfilled by getting payment request.



Current Balance

55 DSGD

Account Details

Merchant
Address: 0x87...ffB0

Native Matic Tokens
0.102812887048281563 Matic

Request A Payment

Amount (DSGD): 5

From (address): 0x785Bbaf36c85fB92707aa318F513aD6e0C443862

Message: Payment for product A

Alchemy - Polygon Mumbai

Account 4 → 0xc9821...B15F9

http://localhost:3000

0xc9821...B15F9 : CONTRACT INTERACTION

$0.00

Gas (estimated) $0.00 0.00021169 MATIC

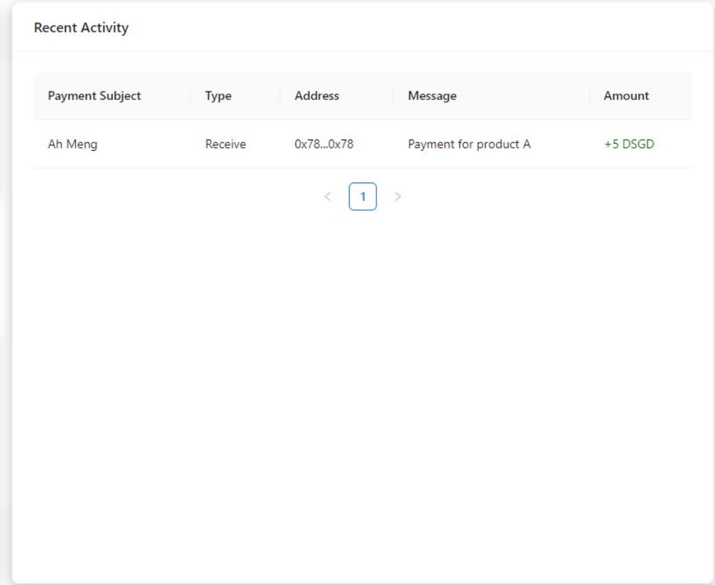
Likely in < 30 seconds Max fee: 0.00021169 MATIC

Total $0.00 0.00021169 MATIC

Amount + gas fee Max amount: 0.00021169 MATIC

Figure 8: User Interface of Merchant Wallet making request

The figure above illustrates the sequence of events where a merchant initiates a payment request from users who have ordered product A. This action prompts the Metamask wallet to request authorization from the merchant to sign the transaction. Only after obtaining this authorization, the relevant data is recorded in the smart contract and subsequently propagated throughout the blockchain.



Recent Activity

Payment Subject	Type	Address	Message	Amount
Ah Meng	Receive	0x78...0x78	Payment for product A	+5 DSGD

Figure 9: User Interface of Merchant’s Wallet Payment History

Upon completion of the payment by a specific party, the transaction history will be displayed in the "Recent Activity" section within both parties’ wallet.

3 FUTURE WORK

3.1 Smart Contracts Design

This task holds the highest priority as the progress of other tasks depends on the establishment of a smart contract. The development of smart contracts is essential, including cross-border transactions, e-commerce payments, and the voucher system. In fact, a prototype featuring basic smart contracts for payments and the distribution of CBDC as DSGD has been already done. Subsequently, the focus will be on creating the cross-border transaction smart contract, drawing insights from Project Icebreaker, Project Dunbar, and Project Orchid. It is important to acknowledge that there may be a minor project delay due to upcoming final exams. There is also a need to have a balanced allocation of time for academic preparation.

In addition to that, the envisioned wallet functionality will allow users to purchase products from international merchants using CBDC, acquire e-commerce vouchers, and store them within the wallet. The potential incorporation of extra features will depend on the availability of time after successfully implementing the anticipated functions.

3.2 User Interface (UI) Redesign

In the context of enabling user interactions with the contracts, UI holds significant importance. As the contracts have undergone upgrades, the existing UI is no longer compatible with the updated contract functionalities. Therefore, it becomes imperative to revise and align the UI with the revised contract functions.

3.3 Code Testing and Cleaning

Upon completion of application development, the subsequent and indispensable phase will be application testing. It becomes important to formulate precise tests, such as unit test, integration test as well as comprehensive end-to-end test of the application's functionality. Apart from testing the code, code cleaning is of great significance. This involves enhancing the readability and ease of managing the code, ensuring that the codebase is well-structured and free from unnecessary complications.

4 CONCLUSION

In my opinion, I might have spent too much time on learning and literature review but I believe this investment is crucial for understanding the technology limitations. The three mini DApps and the wallet prototype will be supportive references during the process of building this project.

Undeniably, making adjustment for quickly transition into the development stage is also important because the challenges that emerge during development are often unexpected and have the potential to disrupt the project's timeline. Nonetheless, it's worth noting that the project is still aligned with its original plan, indicates positive progress.

References

[1] D. Kurnianingrum, Mulyani, and M. Luckieta, "Understanding the Digital Payment Services Through User Experience During the Pandemic Era," IEOM Society International, Mar. 2021. [Online]. Available: http://www.ieomsociety.org/singapore2021/papers/943.pdf

[2] "Project Dunbar - International settlements using multi-CBDCs," BIS, Innovation Hub in partnership with the Reserve Bank of Australia, Central Bank of Malaysia, Monetary Authority of Singapore, and South African Reserve Bank, Research & Publication, Mar. 2022. [Online]. Available: https://www.bis.org/publ/othp47.htm

[3] "What is Blockchain Technology? | IBM." Accessed: Nov. 11, 2023. [Online]. Available: https://www.ibm.com/topics/blockchain

[4] D. Davis, "What Happened To FTX? The Crypto Exchange Fund's Collapse Explained.," Forbes. Accessed: Sep. 27, 2023. [Online]. Available: https://www.forbes.com/sites/darreonnadavis/2023/06/02/what-happened-to-ftx-the-crypto-exchange-funds-collapse-explained/

[5] "Central Bank Digital Currency Tracker," Atlantic Council. Accessed: Sep. 13, 2023. [Online]. Available: https://www.atlanticcouncil.org/cbdctracker/

[6] F. Jin, J. Li, and Y. Xue, "Preferring stablecoin over dollar: Evidence from a survey of Ethereum platform traders," *Journal of International Money and Finance*, vol. 131, p. 102796, 2023, doi: https://doi.org/10.1016/j.jimonfin.2022.102796.

[7] "PAYMENTS are eating THE WORLD | J.P. Morgan." Accessed: Nov. 09, 2023. [Online]. Available: https://www.jpmorgan.com/insights/payments/payment-trends/payments-are-eating-the-world

[8] "Project Orchid," Monetary Authority of Singapore, Singapore, Schemes and Initiatives, Oct. 2022. Accessed: Sep. 13, 2023. [Online]. Available: https://www.mas.gov.sg/schemes-and-initiatives/project-orchid

[9] "Jasper-Ubin Design Paper - Enabling Cross-Border High Value Transfer Using Distributed Ledger," Accenture, 2019. [Online]. Available: https://www.mas.gov.sg/-/media/Jasper-Ubin-Design-Paper.pdf

[10] "A Retail Central Bank Digital Currency: Economic Considerations in the Singapore Context," Monetary Authority of Singapore, Singapore, Nov. 2021. [Online]. Available: https://www.mas.gov.sg/publications/monographs-or-information-paper/2021/retail-cbdc-paper

[11] "Project Nexus - Enabling instant cross-border payments," Bank for International Settlements, Mar. 2023. [Online]. Available: https://www.bis.org/about/bisih/topics/fmis/nexus.htm

[12] "Project Icebreaker -Breaking new paths in crossborder retail CBDC payments," Bank for International Settlements (BIS), Mar. 2023. [Online]. Available: https://www.bis.org/publ/othp61.htm

[13] "Norges Bank," GitHub. Accessed: Nov. 10, 2023. [Online]. Available: https://github.com/norges-bank

[14] "Purpose Bound Money (PBM) Technical Whitepaper," Monetary Authority of Singapore, Singapore, Information Papers, Jun. 2023. [Online]. Available: https://www.mas.gov.sg/-/media/mas-media-library/development/fintech/pbm/pbm-technical-whitepaper.pdf

[15] "Technical Design," GitHub. Accessed: Nov. 10, 2023. [Online]. Available:

https://github.com/opengovsg/cbdc-smart-contracts/wiki/Technical-Design

[16] D. Lim, “What’s an Example of Good E-Commerce Database Design?,” Headless E-Commerce Platform | fabric. Accessed: Nov. 10, 2023. [Online]. Available: https://fabric.inc/blog/commerce/ecommerce-database-design-example

[17] dewchan, “dewchan01/petshop.” Aug. 31, 2023. Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/dewchan01/petshop

[18] “Smart Contract in Ethereum Blockchain.” Accessed: Nov. 12, 2023. [Online]. Available: https://www3.ntu.edu.sg/home/ehchua/programming/blockchain/ethereum.html

[19] dewchan, “dewchan01/evoting.” Aug. 31, 2023. Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/dewchan01/evoting

[20] dewchan, “web3-tickets.” Aug. 31, 2023. Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/dewchan01/web3-tickets

[21] “opengovsg/cbdc-smart-contracts: Smart contracts for the CBDC collaboration.” Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/opengovsg/cbdc-smart-contracts

[22] “FYP/cbdc-smart-contracts at main · dewchan01/FYP,” GitHub. Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/dewchan01/FYP/tree/main/cbdc-smart-contracts

[23] “FYP/PBM-SG.drawio.svg at main · dewchan01/FYP.” Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/dewchan01/FYP/blob/main/PBM-SG.drawio.svg

[24] “PurposeBoundMoney/PBM: Purpose Bound Money proposes a common protocol to specify conditions for the use of digital money.” Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/PurposeBoundMoney/PBM