Sustainable Social Bank Development using Blockchain

Nabilah Daud, Amina Abdinasir Ahmed Moallim, Normi Sham Awang Abu Bakar
Department of Computer Science, Kulliyyah of ICT
International Islamic University Malaysia
nabilah.daud.musa@gmail.com, amina.abdinasir.ahmed@gmail.com nsham@iium.edu.my

***Abstract*—** Blockchain has made a big impact as the new rising technology that provides immutability and integrity in data. Attracting software developers to revolutionize Blockchain in their own system, it has proven to be quite difficult to associate traditional software development methodologies to achieve one's goal for a system. In this paper, we will explore whether Agile methodology can be invested as a prime model for development by applying it in a Blockchain mobile application called Sustainable Social Bank. Using time as a conversion currency to exchange services, all processes to incentivize the application will be recorded and Agile will be implemented as the software development method that will gather the requirement, analyzation, design, development, testing module and deployment of the smart contract in the Blockchain. After defining the general process of Agile methodology used as the context of this publication, we will summarize it in the end product of the mobile application.

***Keywords*—** Timebank, Sustainable Social Bank, Blockchain-Oriented Software, Agile Software Development

I. INTRODUCTION

Blockchain technology is one of the fastest growing sectors in the field of information technology, where it can be applied in many fields and applications from IoT to the financial sector. According to [1] Blockchain is a distributed ledger which is shared and synchronized amongst the members of either public or the private peer-to-peer network. the ledger records permanently all the transactions that have been exchanged in the network. Blockchain has many applications other than calculating the number of bitcoins. It is capable of performing complex operations. This paper discusses the concept of Blockchain-oriented Software, and the software engineering practices used in the development of Blockchain applications, as well as their advantages. Furthermore, the development methodology was implemented on a Blockchain-based software known as the Sustainable Social Bank (SSB).

II. BLOCKCHAIN-ORIENTED SOFTWARE

Blockchain-oriented software (BOS) are an application of blockchain technology which includes all the software that implement the technology of the Blockchain [2]. As BOS having rapid growth of applications, there is a need for specialized software engineering practices in order to develop efficient applications [3]. With the decentralized nature of the system, the performance, scalability and security requirements cannot be satisfied without properly designing and testing the system [3]. Errors found after deploying Blockchain are impossible to fix due to its immutable nature [3], thus there is an urgent need to develop software engineering practices that mainly focuses on preventing errors before deploying. In addition to that, the Blockchain sector will need experts with well-defined knowledge in finance and law [2] as Blockchain benefits these fields the most. In the normal workflow of the software engineering process, developers start with a plan and then design the application before coding. On the other hand, Blockchain developers would start coding based on the scenario requirements due the lack of specialized modelling notations [4]. This indicates that there are no specialized methodologies to develop a BOS.

A. *Blockchain-based Software Engineering*

BOS applications have features that are not found in traditional applications, there is a lack of specialized Blockchain-based software engineering practice. Therefore, developers use existing practices that can cover parts of the BOS. Existing software engineering practices such as UML diagrams can be used but only if modified and new ones created specified to address the needs of BOS [2]. Code review and unit testing is used by developers to verify the correctness of the functionality of the application [3]. Manual efforts are being used due to the lack of automated testing suits that address the decentralized nature of Blockchain [3]. JavaScript, Python, Go, C++ and Ruby are the top 5 languages used to develop BOS [3]. Entity–relationship model can easily capture the data of the system but fails to model the functional and structural behaviour [4]. Unified Modelling Language can only model and address the functions and data attributes on smart contracts but have limitations on the whole process design [4]. Business Process Modelling Notation can be used to model the process behaviour but it is difficult to model an overview of the system.

B. Agile Methodology in Software Engineering

A system development methodology refers to the framework that is used to structure, plan and control the process of developing a system. Agile methodology is a famous framework used to manage projects by breaking it down into smaller repetitive and incremental phases.

The Agile Method is a particular approach to project management that is utilized in software development. This method assists teams in responding to the unpredictability of constructing software. It uses incremental, iterative work sequences that are commonly known as sprints.

C. Agile Methodology in BOS development

A system adopting agile methodology requires insight from stakeholders for continual progressive development and the changes are mutable to customize until final requirement of the product has been met. [5] Therefore, the Agile methodology is the best model for quick adaptation of a system's requirement. One of the most famous agile methodologies include Scrum and Kanban methods. Scrum technique as an adoption to the agile methodology seems to fit best for most Blockchain development plans compared to Kanban [6]. Fig. 1 shows the general proceedings of an agile methodology.

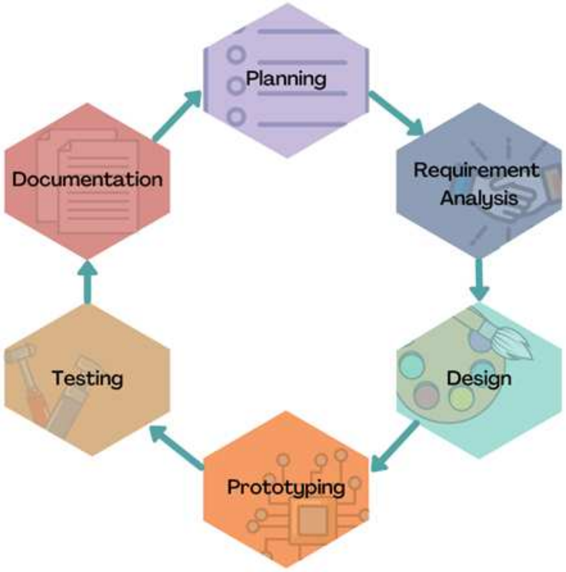


Fig. 1: Agile methodology process

Scrum is a process recognized as daily meetings with the team and the stakeholder. [8] Scrum is done in short meetings where changes are seen and discussed for further development besides supervising the time needed for each task to finish. Scrum masters are responsible for implementing the change in the system from selecting the best solution for a problem to the electing the right person to do the job. The role of the Project Owner (PO) is crucial in several aspects at all stages of the process where the project is constantly presented to them to verify the correctness of the system developed [9]. Scrum and agile methodology in general are used to ensure the functionality of the product and to showcase to the PO within a short interval [10]. Scrum enables the developers to completely develop and test features in short iteration cycle. That is because each team member is in charge of a major reasonability of the project, and ownership of the project is more broadly based [11].

As we depend heavily on Blockchain and its immutability, we must also note the natural state of Blockchain that changes the requirements of the software development stage [7]. Therefore, the process depends heavily on how the Blockchain can be tested and implemented in the project. A part of the agile methodology is elicitation and requirement analysis where the state of Blockchain and its elements needs to be stated in the architectural structure of the software so it will be easier to recognize the next path of development.

III. SUSTAINABLE SOCIAL BANK

Sustainable Social Bank is a mobile application that uses the concept of timebank where time is the currency in the transactions [12]. In order for the user to get balance in their account they need to perform tasks for a certain amount of time, and after that they can exchange services and pay with the time-currency available in their wallet. Sustainable Social Bank is going to help the International Islamic University Malaysia (IIUM) community to get together and help each other and the environment. Money is not going to hinder people from getting help, and instead by performing tasks like community services IIUM community is going to find new ways to connect and improve social inclusion. Besides enhancing

Some members of the IIUM community find finance as an obstacle in getting help. Sustainable Social Bank is going to ease this pain by using Time-banking, where time is the only currency in the transaction by creating an equal environment where each task can scale to the currency standard and ensure the system rewards user respectively regardless of the task [13]. Each user will have a wallet and in order to collect balance the user needs to perform some tasks. After finishing the task, the system is going to validate the transaction by using a smart contract. Users can use their balance to pay for other services provided in the system [14].

A. *Background and Related Work*

BlendTBS is a Blockchain based time banking system made to explore and better understand the social value system in the context of exchanging economies [15]. The system is predicted to encourage people to engage in socially beneficial activities in the community. A Blockchain-based system promises to solve the trust and security issues that can be encountered using the traditional time-banking system [14]. BlendTBS has three main architectural components. Registration is a data structure in a solidity contract that is used by the administrator to store users' identification and profile data. In order for the transaction to be completed the system must delegate the user to send the transaction after the user's identification. The access content verification process is done by the minor to ensure the authorized access to the system. Three components are required to use the system, a web browser to run the user interface. Metamask, a browser plugin used to connect the browser to the Blockchain and a Blockchain server.

The registered users will get 10 hours as an initial time credit deposit. A user can send a service request including the service and the hours needed. A service provider will send a request to perform the service, after completing the user will click check to complete the transaction. After completing the transaction, the time currency will be transferred to the service provider account and a block will be created including the information of the transaction then will be stored in the Blockchain.

IV. PROTOTYPE DEVELOPMENT

To understand further the needs of users from the system, a prototype was built through Scrum sprints. A prototype in the form of mobile application is important because of the variables in sizing for Android and iOS mobile phones. We customized our mobile application using a standard size phone with a screen resolution of 1920×1080. Besides that, prototyping is an important part of iteration in the project whereby the design of the app will change overtime if requirements are changed and need to be met.

To analyse the future application to be built, a class diagram is constructed to observe the static view and responsibilities the system has to do to satisfy the user. The class diagram is depicted in Fig. 2.

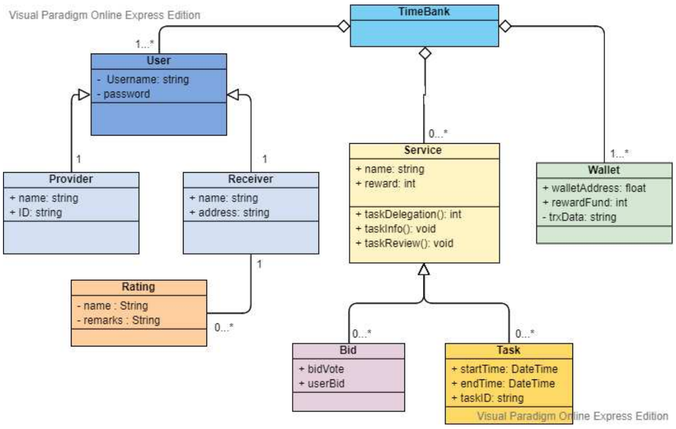


Fig. 2. Class Diagram of SSB

The main component in this environment is affected via the process of any two users interacting and making transactions. Requester and Provider are the main two actors of the system. Both users will be able to login, offer and search for services, and reach a settlement. In order for the Requester to pay for the service, the Provider should finish the service and the provider should have a sufficient amount of time balance in their wallet. Besides the casual transaction of the two main actors, the main goal we are trying to achieve is to ensure that the Requester can get the task that they bid for from the Provider only through the evaluation of bidding for the task and the user's rating. The timebank directly deals with the user, service and the wallet. The payment, also known as reward, is transferred to the Blockchain wallet. Since the application implements Blockchain, all transactions will be permanently stored in this system, improving the transparency and trust between user and system [17].

The attributes of this system are different from most transaction businesses that exist because of the bidding and task related to the service. Bidding is when users, specifically the Provider, have the chance to bid for the task they want to do. Therefore, the Requester will have to post the job description in the app to give a chance to the whole community to see and decide whether they are up for the task and if yes, bid for the job, therefore, giving the Requester many applicants to choose from to do their job.

A. *Software Implementation*

Currently there are two ways to form a timebank and the type that we chose is Type 1 in metaphor known as the "babysitting circle" whereby people can earn time-credits from other members of the circle and these then circulate around other members, who pay it out again when they want help for themselves. This definition is a foundation to guide us to build an app that revolves around Blockchain [3]. The most celebrated programming language for Blockchain is Solidity Programming Language where Blockchain and smart contracts can be deployed even without downloading any software.

To understand the timebank system, we must first dive into the knowledge of creating a smart contract using Solidity which is a contract-oriented high-level language which is designed to run in Ethereum Virtual Machine (EVM). Contract-oriented means that smart contracts are the only way to organize code, store data and other programming logic and the EVM is a global singleton, meaning that it operates as if it were a global, single-instance computer, running everywhere. For this project, we have tested the Solidity language using a browser-based tool called Remix which is an online compiler that can execute Solidity Language.

After experimenting with Remix, we will now configure smart contracts into something broader which is a decentralized application (DApps). DApps provide a feature of Blockchain in terms of interaction and query that can only be accessible through peer-to-peer (P2P) networking and protocols. The main objective of this project is to implement Blockchain as the main server and database sink-in communication between other devices with the application.

B. *Truffle Integrated Development Environment (IDE)*

The main use of Truffle is to provide an end user application that runs a smart contract. Swiss army knife, a name regarded by Truffle, packs together the best modules and tools to streamline smart contract creation, compilation, testing, and deployment onto Blockchain. The end product for this application requires the use of web3.js JavaScript Library which connects the Blockchain and the browser along with an interface to a P2P storage network called Swarm and a P2P messaging service called Whisper. Truffle provides a tool to set up our own Blockchain in our own machine, called Ganache, which can be done through command-line or GUI.

V. PROTOTYPE IMPLEMENTATION

Introducing a new app can be a struggle for some developers if they do not know the demographic audience for the app. Therefore, the user interface that we have adapted uses the design of abstract and geometric art which has a passive energy to approach users of all demographics and bright colours to draw attention besides providing an easy view for users to digest without being distracted by the app's functionality. The interface we have developed as a prototype are shown in Fig. 3:

Main Page: The first page that the user will interact with as the app starts

Login page: Users will have the ability to insert information to log in the system.

Homepage: shows the task available and amount of time-credit earned.

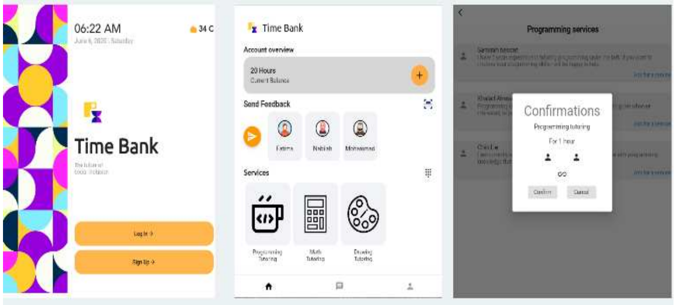
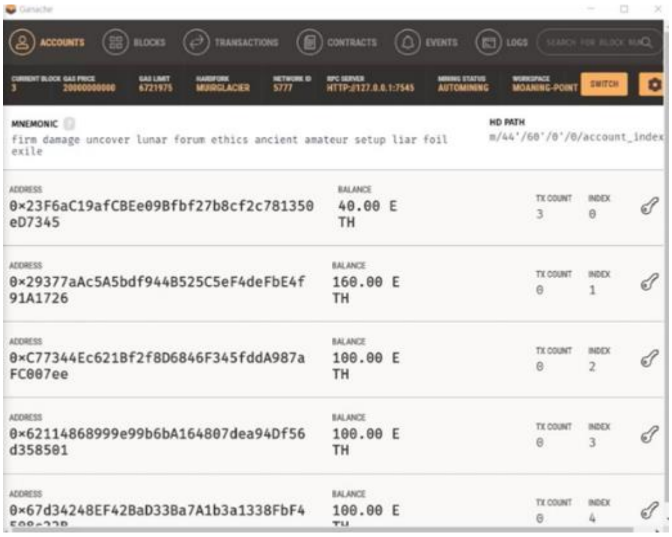


Fig. 3. User Interface of SSB

VI. Testing the Blockchain System

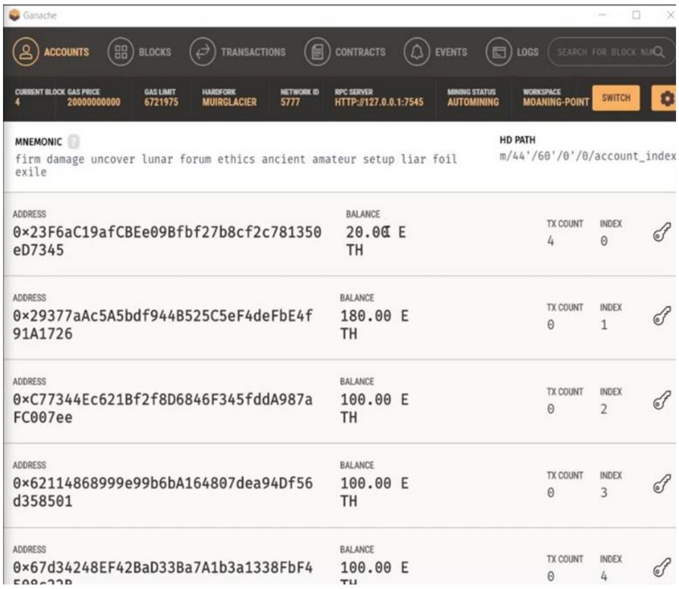
Fixing errors on Blockchain after deploying is impossible, hence the smart contract needs to be tested on an Ethereum smart contract environment to alleviate errors. According to [3], unit testing is one of the practices used to investigate the security and scalability of BOS among some developers. Therefore, we chose to perform unit testing on SSB using Ganache. Unit testing is a software testing practice that is performed by testing the smallest units of the system in an environment to verify the requirements of the system [15]. The smart contract was tested using Ganache which is a personal Blockchain environment set up used to test smart contracts.

After completing the transaction, the time credit is successfully deducted from the service receiver account and sent to the service provider. The account balance before the smart contract transaction is depicted in Fig. 4 and after the transaction is completed, the new balance will be added to the table, which is shown in Fig. 5. In addition, transactions in BOS cannot be tampered with, which makes them secure and protected.



```
CURRENT BLOCK  GAS PRICE    GAS LIMIT  HARDFORK     NETWORK ID  RPC SERVER             MINING STATUS  WORKSPACE
3              20000000000  6721975    MUIRGLACIER  5777        HTTP://127.0.0.1:7545  AUTOMINING     MOANING-POINT

MNEMONIC                                                                   HD PATH
firm damage uncover lunar forum ethics ancient amateur setup liar foil    m/44'/60'/0'/0/account_index
exile

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×23F6aC19afCBEe09Bfbf27b8cf2c781350           40.00 E     3         0
eD7345                                         TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×29377aAc5A5bdf944B525C5eF4deFbE4f            160.00 E    0         1
91A1726                                        TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×C77344Ec621Bf2f8D6846F345fddA987a            100.00 E    0         2
FC007ee                                        TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×62114868999e99b6bA164807dea94Df56            100.00 E    0         3
d358501                                        TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×67d34248EF42BaD33Ba7A1b3a1338FbF4            100.00 E    0         4
```

Fig. 4: Account balance before completing the transaction



```
CURRENT BLOCK  GAS PRICE    GAS LIMIT  HARDFORK     NETWORK ID  RPC SERVER             MINING STATUS  WORKSPACE
4              20000000000  6721975    MUIRGLACIER  5777        HTTP://127.0.0.1:7545  AUTOMINING     MOANING-POINT

MNEMONIC                                                                   HD PATH
firm damage uncover lunar forum ethics ancient amateur setup liar foil    m/44'/60'/0'/0/account_index
exile

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×23F6aC19afCBEe09Bfbf27b8cf2c781350           20.00 E     4         0
eD7345                                         TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×29377aAc5A5bdf944B525C5eF4deFbE4f            180.00 E    0         1
91A1726                                        TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×C77344Ec621Bf2f8D6846F345fddA987a            100.00 E    0         2
FC007ee                                        TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×62114868999e99b6bA164807dea94Df56            100.00 E    0         3
d358501                                        TH

ADDRESS                                        BALANCE     TX COUNT  INDEX
0×67d34248EF42BaD33Ba7A1b3a1338FbF4            100.00 E    0         4
```

Fig. 5 Account balance after completing the transaction

VII. Conclusion

Due to the complexity of the Blockchain-based Software (BOS), many developers have become sceptical of the effectiveness of the existing software development methodology. Software engineers have been working in adapting and modifying the traditional practices for software development, especially with increasing demand for mobile application development. In response, Agile software development methods were introduced with the intention of shifting the paradigm of software engineering.

Agile methodology creates a robust teamwork environment and is very suitable for dynamic projects that signifies interaction with stakeholders, the team and the system. Therefore, this methodology makes it easier to add new requirements of modification or features from the old ones. [16] The rise of new methodologies is yet to be reviewed in conjunction with Blockchain as part of the system. Using Scrum or Kanban are techniques to help adopt the Agile methodology as suited to software development.

This work provides a guideline on how to approach building a BOS using Agile software engineering methodology to ensure a smooth process using Agile Process Model such as Scrum. As a general scheme of the software development process, this work can be useful to follow and build a stronger foundation for a better progression plan in Blockchain-oriented software or mobile app development.

References

[1] R. Koul, "Blockchain Oriented Software Testing - Challenges and Approaches," the 3rd International Conference for Convergence in Technology I2CT, 2018.

[2] S. Porru, A. Pinna, M. Marchesi and R. Tonelli, "Blockchain-oriented software engineering: Challenges and new directions," IEEE/ACM 39th International Conference on Software Engineering Companion ICSE-C, 2017.

[3] H.-P. Halvorsen. "Software Development: A Practical Approach", 2020, https://www.halvorsen.blog/documents/programming/software_engineering/resources/Software%20Development.pdf

[4] H. Rocha and S. Ducasse, "Preliminary steps towards modeling Blockchain oriented software," International Conference on Software Engineering, p.52–57, 2018.

[5] V. Vijayakumar, V. Neelanarayanan, P. Rao, J. Light, "Proceedings of 6th International Conference on Big Data and Cloud Computing Challenges", ICBCC 2019, UMKC, Kansas City, USA, 2020.

[6] M. Marchesi, L. Marchesi, R. Tonelli, " An Agile Software Engineering Method to Design Blockchain Applications," the 14th Central and Eastern European Software Engineering Conference, Russia, Sep 2018.

[7] M. Marchesi, L. Marchesi, R. Tonelli, "ABCDE- Agile block chain DApp engineering," DMI, University of Cagliari, Cagliari, 09124, Italy, 2020

[8] P. Abrahamsson, O. Salo, J. Ronkainen & J. Warsta, "Agile software development methods: Review and analysis," VTT publication 478, Espoo, Finland, 107p, 2017.

[9] S. Ibba, M. Seu,P. Andrea, F.E. Pani . (2017). "*CitySense blockchain-oriented Smart Cities,*" the XP2017 scientific workshops, Germany, May 2017.

[10] C. Udokwu , H. Anyanka , A. Norta (2020)., "*Evaluation of Approaches for Designing and Developing Decentralized Applications*," the 2020 4th nternational Conference on Algorithms, Computing and Systems, 2020.

[11] K. Schawber. (1997) ,"*Scrum Development Process,*" Advanced Development methods, 1997.

[12] A.K. Kakar, "Investigating factors that promote time banking for sustainable community based socio-economic growth and development," Computers in Human Behavior, 2020, Vol.107.

[13] Z.A. Kasim, Y. Ramli, "Feasibility Study: The implementation of Time Bank Volunteering Schemes for Senior Citizens in Malaysia.", Institute of Labour Market and Analysis (ILMIA), Ministry of Human Resource, Malaysia, January, 2020.

[14] X. Lin, R. Xu, Y. Chen and J. K. Lum," A Blockchain-Enabled Decentralized Time Banking for a New Social Value System," 2019 IEEE Conference on Communications and Network Security CNS, 2019.

[15] P. Runseon, "A survey of unit testing practices," *IEEE Software*, no. 4, pp. 22-29, 2006 ,vol. 23,

[16] M.H Miraz, M. Ali ,"Blockchain Enabled Smart Contract Based Applications: Deficiencies with the Software Development Life Cycle Models," Baltica Journal, 2020, Vol.33.

[17] R. Xu , X. Lin & Q. Dong & Y. Chen," Constructing Trustworthy and Safe Communities on a Blockchain-Enabled Social Credits System," the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, 20118.