Smart Contract Application on Blockchain Technology in the Software Industry

Wilson Soto
Politécnico Grancolombiano, Bogotá, Colombia
wsoto@poligran.edu.co

***Abstract*—This paper shows a smart contract application on blockchain in the software industry applied in a Software Development Agreement...**

***Keywords*—Blockchain, Ethereum, Smart Contracts, Software**

I. Introduction

The blockchain is one of the technologies most important in the last years due to its main characteristics: distributed computing infrastructure, decentralization, immutability, and cryptographically sealed.

In the blockchain, miners compete to accumulate transactions, solving a difficult computational problem with the purpose to reach consensus, and adding the transactions as blocks to the blockchain network. The first application of the blockchain technology was Bitcoin, which without a centralized mechanism the users to transfer securely the currency. But several blockchain platforms such as Ethereum or Hyperledger have been proposed in other fields beyond cryptocurrencies, namely, Smart Contracts.

Smart contracts are decentralized programs on the blockchain network. The features of the smart contracts are relevant for generating an impact on diverse areas. The comprehensive survey of smart contract-based applications in [1], provides and explores an important summary: financial applications (currency management, know your customer, escrow service, insurance, lending and borrowing, auditing procedures, and stock trading service); health care related services (health information management, clinical research data protection, automated patient monitoring, and treatment); identity management and access control (identity data protection, decentralized identity management, and security policy in access control); real estate (improved secure transactions process, processing fees and commissions for transactions, and extensive processing time); eGovernment/law (enforcement of law, contractual agreements, public services, and national democracy); internet of things (smart contracts for scalable resource sharing of IoT, smart contracts in edge computing, smart contract for the enforcement of IoT security, unmanned aerial vehicles UAV, and smart cities); telecommunications services (autonomous and intelligent resource sharing in telecommunication, user identity management and access control with smart contracts, and smart contracts for roaming services); logistics management (ensuring sea/air freight supply chain quality and compliance, agricultural supply chain regulatory compliance, and special commodity supply chain traceability); smart contracts in cross industry (smart contracts in enforcement of IT security in the industry, smart contracts in energy trading, smart contracts in automotive industry, smart contracts in environmental protection, smart contracts for construction management, and smart contracts for air traffic management).

In this paper, we presented the design, implementation, and testing of a smart contract-based application for automating a software development agreement. This smart contract has the purpose of getting control of the terms and conditions, as well as transparency and ensuring that they have not been manipulated. Besides, the imposing penalties can convert into programmable codes. In this sense, the smart contracts in contractual agreements are an ideal solution to replace the classical contracts and improve efficiency, effectiveness, and security [1]. In consequence, we can classify this approach as a smart contract-based application set to contractual agreements.

This approach provides a helpful reference to the people who want to design smart contracts or improve processes in the industry of the software.

The paper is structured as follows. The next section shows the background necessary for understanding this paper. Section III explains the methodology used and contains the tools and the implementation of the smart contract as a solution to a software development agreement. Section IV shows the preliminary results. Finally, the last section concludes the paper.

II. Background

Many authors have contributed to systematic reviews, applications, and future research on blockchain technology. For example, [2] shows an exploratory study and descriptive analysis of blockchain technology. In this work is essential the classification of items according to the thematic area, and the taxonomy of blockchain-based applications. A. Ethereum The first platform for developing smart contracts, and currently the most popular. Also, can be used to design several decentralized applications (DApps) [3]. The advantages of the platform Ethereum are an open-source system, a worldwide developer community in contribution, available in private and public mode, and availability of native cryptocurrency. In contrast, the disadvantages are public ledger storage overheads, transaction approval time, transaction cost, single programming language support (solidity), and integration limitations [1].

A. Smart Contracts

In recent years, several surveys have been presented, such as in [4]. This survey shows the literature and online resources about smart contracts in the period 2008-2020. Besides, the paper summarizes several challenges and future research.

In [5] shows a summary of the scenarios, areas, and industries for the smart contract-based applications. The report explores among others the internet of things, smart grid, proofs of origin, supply chain management and purchasing, medical engineering, financial sector, media industry, public sector, legal sector, stock market trading, and asset management.

In [3] presents a comparative summary of the reviews published that have studied the smart contracts in blockchain technology. The paper [6] proposed a study of smart contracts, including frameworks, operating mechanisms, platforms, and programming languages. Moreover, application scenarios, challenges, and future development trends.

Another important survey [7] classified 391 papers related to smart contracts in three topics: dynamic analysis, vulnerability detection, and program correctness in static analysis. A specific use case of smart contract-based application is implemented in [8]. This paper proposed a smart contract for a university examination system.

B. Software Development Agreement

In the software industry, a software development agreement (sometimes referred to as a master services agreement) is a contract between two people, the client and software provider. Generally, in a software development agreement the structure of a contract includes, but is not limited to, the following items:

1) Engagement Of Developer
2) Client Responsibilities
3) Acceptance Of Software
4) Compensation And Payment
5) Term And Termination
6) Confidentiality
7) Intellectual Property Rights
8) Warranties And Disclaimers
9) Limitation Of Liability
10) Non-Solicitation
11) General Conditions

III. Methodology

In this paper, the smart contract for the development agreement is designed for identifying requirements and paying them.

A. Tools

For developing and deploying the smart contract the following tools were selected based on the comparison shown in [9].

1) Metamask is a browser extension, crypto wallet, and designed to make accessing Ethereum's Dapp ecosystem easier.
2) Remix is an IDE that allows developing, deploying, and administering smart contracts for Ethereum.
3) Solidity is a programming language designed for developing smart contracts that run on Ethereum.
4) Brownie is an open-sourced Python development and testing framework for smart contracts targeting the EVM (Ethereum Virtual Machine).
5) Infura is the Web3 backend IAAS (Infrastructure-As-A-Service) provider that have a great diversity of services for blockchain developers.
6) Rinkeby is a Ethereum test network mainly used for blockchain developers before deployment on the Ethereum mainnet.
7) Visual Studio Code is a source code editor for building applications using a development environment.
8) Etherscan is a website for exploring blocks in the decentralized smart contract platform Ethereum.
9) Ganache is a local Ethereum simulator for testing smart contracts in solidity.

B. Implementation

IV. Results

V. Conclusions

References

[1] Fran Casino, Thomas K. Dasaklis, and Constantinos Patsakis. A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telematics and Informatics*, 36:55–81, 2019.

[2] Shuai Wang, Yong Yuan, Xiao Wang, Juanjuan Li, Rui Qin, and Fei-Yue Wang. An overview of smart contract: architecture, applications, and future trends. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 108–113, June 2018.

[3] Giuseppe Destefanis. Design patterns for smart contract in Ethereum. In *2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C)*, pages 121–122, March 2021.

[4] Shinhae Kim and Sukyoung Ryu. Analysis of Blockchain Smart Contracts: Techniques and Insights. In *2020 IEEE Secure Development (SecDev)*, pages 65–73, Sep. 2020.

[5] Shafaq Naheed Khan, Faiza Loukil, Chirine Ghedira-Guegan, Elhadj Benkhelifa, and Anoud Bani-Hani. Blockchain smart contracts: Applications, challenges, and future trends. *Peer-to-Peer Networking and Applications*, 14(5):2901–2925, 2021.

[6] Tharaka Hewa, Mika Ylianttila, and Madhusanka Liyanage. Survey on blockchain based smart contracts: Applications, opportunities and challenges. *Journal of Network and Computer Applications*, 177:102857, 2021.

[7] Ashis Kumar Samanta, Bidyut Biman Sarkar, and Nabendu Chaki. A Blockchain–Based Smart Contract Towards Developing Secured University Examination System. *Journal of Data, Information and Management*, (PMC8284418):1–13, July 2021.

[8] Birgit Weimert, Wolfgang Prinz, Nils Urbach, Steffen Holly, Axel Schulte, Gilbert Fridgen, Thomas Rose, Julian Schütte, Mathias Dalheimer, Markus Wenzel, Boris Otto, Christian Schwede, Ulrich Leiner, Michael Fritz, Michael Kreutzer, Alexander Nouak, Thomas Hoeren, Christian Welzel, Philipp Sprenger, and Nikolas Guggenberger. Blockchain and smart contracts – Technologies, research issues and applications. Technical report, 04 2018.

[9] Bin Hu, Zongyang Zhang, Jianwei Liu, Yizhong Liu, Jiayuan Yin, Rongxing Lu, and Xiaodong Lin. A comprehensive survey on smart contract construction and execution: paradigms, tools, and systems. *Patterns*, 2(2):100179, 2021.

[10] Monika di Angelo and Gernot Salzer. A survey of tools for analyzing ethereum smart contracts. In *2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON)*, pages 69–78, 2019.