IJFANS INTERNATIONAL JOURNAL OF FOOD AND NUTRITIONAL SCIENCES

ISSN PRINT 2319 1775 Online 2320 7876

Research paper UGC CARE Listed (Group -I) Journal Volume 11, Iss 7, July 2022

An Blockchain-Based Healthcare Apps

Rohaila Naaz, Assistant Professor

College Of Computing Sciences And Information Technolog, Teerthanker Mahaveer University, Moradabad, Uttar Pradesh, India

Email id- rohailanaaz2@gmail.com

ABSTRACT: Due to characteristics like as decentralization, openness, and immutability, the blockchain, the underlying data structure of the Bitcoin system, has gotten a lot of attention since its creation. These characteristics make blockchains ideal for applications that need disintermediation through trustless exchange, reliable and incorruptible transaction records, and operational models other than money. Blockchain and its smart contract capabilities, in particular, have the potential to address healthcare interoperability issues like enabling effective interactions between users and medical applications, securely delivering patient data to a variety of organizations and devices, and increasing the overall efficiency of medical practice workflow. Despite the growing interest in blockchain for healthcare interoperability, limited information on the specific architectural styles and patterns for utilizing blockchain in healthcare applications is accessible. This paper fills a gap in the market by demonstrating (1) the features and implementation challenges of healthcare interoperability, (2) an end-to-end case study of a blockchain-based healthcare app we're working on, and (3) how foundational software patterns can help address common interoperability challenges faced by blockchain-based healthcare apps.

KEYWORDS: Bitcoin, Blockchain, DASH, Healthcare, Interoperability.

1. INTRODUCTION

Blockchain technology has piqued the attention of computer scientists and domain specialists in a variety of sectors, including banking, real estate, healthcare, and transactive energy, during the last several years [1]. The popularity of Bitcoin and the Bitcoin platform, which is a cryptographic currency foundation and the first use of blockchain, sparked this interest. Bitcoin has become a feasible platform for "trustless" transactions, which may occur directly between any parties without the involvement of a centralized middleman, thanks to the characteristics of blockchain, such as decentralization, transparency, and immutability. Ethereum, another blockchain technology, enhanced the Bitcoin network's capabilities by introducing support for "smart contracts." Smart contracts are computer programs that directly manage the exchanges or redistributions of digital assets between two or more parties based on pre-determined rules or agreements [2]. Decentralized applications (DApps) are autonomously run services cryptographically recorded on the blockchain that allow direct interaction between end users and suppliers, thanks to ethereum's programmable smart contracts.

This article focuses on a hitherto unexplored subject in blockchain, specifically the usage of software patterns to modularize and enable extension of blockchain-based applications aimed at solving healthcare interoperability issues [3]. Interoperability is described as the capacity for various information technology systems and software applications to interact, exchange data, and utilize the data that has been shared in the healthcare setting [4]. Our emphasis on using blockchain technology to assist bridge the gap in communication and information sharing is motivated by the high (and rising) cost of healthcare in many countries. The following is how the rest of the paper is organized: an introduction of blockchain technology and discusses how blockchain-based applications may aid in the resolution of major issues in healthcare interoperability. An end-to-end

ISSN PRINT 2319 1775 Online 2320 7876
Research paper UGC CARE Listed (Group -I) Journal Volume 11, Iss 7, July 2022

case study of a blockchain-based healthcare software we're working on, as well as the implementation difficulties we ran into while expanding it.

It describes foundational software patterns that can be used to address interoperability requirements in the blockchain app covered in and discusses key concerns when implementing these patterns in healthcare-focused blockchain apps; compares our work with existing work on the potential benefits and drawbacks of a health blockchain and summarizes the findings [5]. A blockchain is a decentralized computer architecture that keeps track of a growing list of ordered transactions organized into blocks that are constantly reconciled to keep information current. Only one block may be added to the blockchain at a time, and each block is mathematically validated (using cryptography) to verify that it is put in the correct order to preserve decentralized network consensus. The method is known as "mining" or Proof of Work (PoW), and it enables network nodes (also known as "miners") to compete to be the first to have their block added to the blockchain by solving a computationally costly problem. The winner then broadcasts the answer to the whole network in exchange for bitcoin mining incentives. This method uses a combination of game theory, encryption, and incentive engineering to guarantee that the network agrees on each block in the blockchain and that the transaction history is not tampered with.

The blockchain stores all transaction records and makes them available to all network nodes, guaranteeing transparency, incorruptibility, and resilience (since there is no single point of failure). A blockchain acts as a public record for all financial transactions in bitcoins in the Bitcoin program, promoting trustless finance between individual users and protecting all of their transactions using cryptography. However, when it comes to enabling various kinds of applications requiring contracts, equity, or other information, such as crowdfunding, identity management, and democratic voting registration, the Bitcoin blockchain has limits. Ethereum was developed as an alternative blockchain to meet the demand for a more flexible foundation, providing users with a broad, trustless platform on which to execute smart contracts, which are computer protocols that allow many kinds of distributed systems beyond currency [6].

1.1 Using Blockchain-based Apps to Address Healthcare Interoperability:

Many research and engineering ideas have been proposed and implementation attempts are underway to apply blockchain to healthcare, but few published studies have addressed the software design considerations needed to implement blockchain-based healthcare apps effectively. While it is critical to understand the fundamental properties of blockchain, it is also critical to use sound software engineering practices when programming the blockchain in order to reduce software maintenance effort through code modularity and extensibility, particularly in the fast-growing and in-demand healthcare domain. The rest of this section outlines major healthcare interoperability issues and how blockchain technology may help [7].

- *Challenge 1:* Maintaining Evolvability While Reducing Integration Complexity Many apps are built on the premise that data is easily changeable. However, data in a blockchain application is immutable and impossible to change in bulk. When developing blockchain applications for healthcare, it's crucial to make sure that the data and contracts stored on the blockchain are structured in such a manner that they can evolve as required. Healthcare data must frequently be available from a range of deployed systems that cannot simply be modified over time, despite the fact that evolution must be encouraged. As a result, the

ISSN PRINT 2319 1775 Online 2320 7876
Research paper © 2012 IJFANS. All Rights Reserved, UGC CARE Listed (Group -I) Journal Volume 11, Iss 7, July 2022

evolvability should be built in such a manner that the effect of evolution on clients interacting with data in the blockchain is minimized. The Abstract Factory design may be used in Ethereum contracts to promote evolution while reducing the effect on dependent healthcare application clients [8].

- *Challenge 2:* Minimizing Data Storage Requirements is the second challenge. When all of this data is kept on the blockchain, it may be a huge burden, especially if data normalization and denormalization methods aren't properly considered. Maximizing data sharing while providing adequate freedom to address individual health issues is a key design aspect. The Flyweight design, as described in Section 4.2.2, may be used to guarantee that common intrinsic data is shared across Ethereum contracts while enabling extrinsic data to differ between individual contracts [9].
- *Challenge 3:* Finding a balance between ease of integration and security concerns. The Office of the National Coordination for Health Information Technology (ONC) defines basic technical requirements for achieving interoperability as identifiability and authentication of all participants, ubiquitous and secure infrastructure to store and exchange data, authorization and access control of data sources, and the ability to handle data sources of various structures [ONC]. Due to its inherent design characteristics, such as strong encryption and a robust peer-to-peer network, blockchain technologies are emerging as a viable and cost-effective way to fulfill some of these criteria. Similarly, the characteristics of asset sharing, audit trails of data access, and pseudonymity for user information security, which are important for addressing interoperability problems in healthcare, may help blockchain-based applications in the healthcare sector. Although the interoperability and availability of keeping patient data on the blockchain has enormous potential advantages, it also has major dangers even when encryption is used. In Section we look at how the Proxy pattern may be used to help facilitate blockchain interoperability while preventing sensitive patient data from being directly recorded in the blockchain [10].

1.2 A Case Study of The Smart Health Dapp (Dash):

This section explains the structure and operation of the DASH app, which we're building to test the effectiveness of blockchain technology in the healthcare sector. It also goes through the difficulties of creating blockchain-based applications to enhance healthcare interoperability.

- *DASH Structure and Functionality:* Patients may access and update their medical data, as well as make prescription requests, via DASH's web-based interface. Similarly, depending on patient authorization, the app enables physicians to examine patient data and complete prescription orders. The structure of DASH is shown in Figure 2. DASH is built on an Ethereum test blockchain, with a SMART (Substitutable Medical Apps, Reusable Technology) on FHIR (Fast Healthcare Interoperability Resources) [Bender and Sartipi 2013] schema serving as the standard data format for patient data. A Patient Registry contract is built using Ethereum's smart contract technology to maintain a mapping between unique patient IDs and their corresponding Patient Account contract addresses. A list of health professionals who have read/write access to the patient's medical information is also included in each Patient Account contract. DASH can offer basic access services to two kinds of users at this time: patients and providers.

- *Important Implementation DASH's Challenges:* Although our DASH prototype worked well for its original objectives, as we tried to expand it to accommodate additional kinds of users from different departments within the same company, we ran across the following issues:

  1. Designs that are tightly linked: Because of the strong coupling between numerous variable components in our original architecture, many contracts had to be rewritten, which propagated to changes in client code. For example, information may be kept as member variables or event logs in the internal states of smart contracts, but as better design options become available, new contracts must be instantiated, rendering current contracts outdated owing to blockchain's immutability.

  2. Resources that are duplicated: Despite the fact that healthcare is made up of many different organizations, there is a lot of similarity. Patients allocated to various care management teams, for example, may have the identical insurance plans, which may be copied multiple times if their care management teams are from different departments, using more storage than required. Duplicated resources are particularly complex and expensive to maintain because, in order to prevent misunderstanding, all of the related copies must be updated properly and on time when an update happens. To maintain data integrity, all copies will have to be reevaluated or recollected if some copies are not updated.

  3. There is a lack of scalability: Another issue with blockchain-based healthcare applications is the ease with which events may be broadcast to different groups of people, such as patients, doctors, and billing departments. To keep physicians up to date on a patient's actions, for example, a back-end server must execute computationally intensive operations to continuously monitor events related to that patient's Patient Account contract. Unfortunately, when a large number of parties need to be notified about the actions of other parties with whom they engage, this method does not scale.

1.3 Applying Blockchain-Based Health Apps to Foundational Software Patterns:

This section explains how fundamental software patterns [Gamma 1995; Buschmann et al. 2007] may be used in blockchain-based health applications (such as DASH) to solve the interoperability issues mentioned.

- Ethereum smart contracts are created using Solidity, a Turing-complete programming language. The Ethereum blockchain has become a platform for developing decentralized apps (DApps) as a result of this contract language, potentially addressing healthcare interoperability issues. Solidity is an object-oriented programming language designed mainly for creating Ethereum contracts. In Solidity, a "class" is realized by a "contract," which is a blockchain-based prototype of an object. A contract may be instantiated into a concrete SCA by a transaction or a function call from another contract, much as an object-oriented class can be instantiated into a concrete object at runtime. It is given a unique identifying address upon instantiation, analogous to a reference or pointer in C/C++-like languages, with which it may be called afterwards. Persistent state variables in contracts may also be used to store data. Despite the fact that a single contract may be instantiated into many SCAs, it should be regarded as a singleton to save storage cost. Following the

ISSN PRINT 2319 1775 Online 2320 7876
Research paper UGC CARE Listed (Group -I) Journal Volume 11, Iss 7, July 2022

creation of a contract, the associated SCA address is usually saved somewhere (e.g., a configuration file or a database) and utilized by the application to access the contract's internal states and functionalities. Multiple inheritance, including polymorphism, is supported by Solidity. When a contract inherits from one or more other contracts, it is generated as a single contract by transferring all of the base contract's code into the newly constructed contract instance. In Solidity, declaration headers may be specified without actual implementations using abstract contracts. They can't be combined into a SCA, but they may be used as a starting point. Many fundamental software patterns may be used to smart contracts to solve different design problems, as discussed below, since they are comparable to C/C++-like classes.

- Improving DASH Design by Using Software Patterns. The rest of this part concentrates on four software patterns–Abstract Factory, Flyweight, Proxy, and Publisher Subscriber–that we used in Section 2 to solve healthcare interoperability issues, although these are not the only patterns applicable in this area. In the DASH app's environment, various patterns may interact as seen in Figure 3. Abstract Factory, for example, can help with user account creation based on user types; Flyweight, on the other hand, ensures that accounts on the blockchain are unique and maximizes commonality sharing; and Publisher-Subscriber can be used to notify collaborating users when events of interest occur on the blockchain. In addition, the Proxy design may be used to abstract away storage implementation detail, allowing smooth interactions between different components in the system while still enabling a variety of data storage choices.

2. DISCUSSION

This paper described the motivations for using blockchain technology to solve healthcare interoperability issues, focusing on (1) maintaining evolvability while minimizing integration complexity, (2) minimizing data storage requirements, (3) balancing integration ease with security concerns, and (4) tracking relevant health changes across lanes. We provided a case study of our DApp for Smart Health (DASH) to highlight the implementation difficulties that arose as we tried to expand the app, such as closely linked designs, duplicated resources, and scalability issues. We discovered and implemented four software patterns to DASH to enhance its design and functionality in order to solve interoperability issues while also offering answers to implementation issues.

3. CONCLUSION

We discovered through our research that the blockchain's public, immutable, and verifiable characteristics enable a more interoperable environment that is difficult to create with conventional methods that depend on a centralized server or data storage. To get the most out of the smart contract support while avoiding a lot of compute and storage overhead, important design choices must be taken ahead of time. Combining excellent software design practices with the blockchain's unique characteristics may result in more modular applications and simpler smart contract maintenance. We used the Abstract Factory, Flyweight, Proxy, and Publisher-Subscriber patterns to decouple entity generation and access, optimize resource sharing, and enhance

Research paper © 2012 IJFANS. All Rights Reserved, UGC CARE Listed (Group -I) Journal Volume 11, Iss 7, July 2022

application scalability in DASH. Our next work will build on the app mentioned in Sections 3 and 4 by delving further into the issues and determining the most feasible design approach for a healthcare blockchain architecture. We'll also look at how other software patterns may be used to address issues like security, privacy, reliability, and performance. Experiments to assess the effectiveness of applying software patterns to this architecture to other alternative architectures are one option. We'll also look at blockchain applications in healthcare, such as establishing an alternative health chain that caters only to the healthcare industry.

REFERENCES

[1] P. Zhang, J. White, D. C. Schmidt, and G. Lenz, "Design of blockchain-based apps using familiar software patterns with a healthcare focus," *Proc. 24th Conf. Pattern Lang. Programs*, 2017.

[2] P. Zhang, J. White, D. C. Schmidt, and G. Lenz, "Design of Blockchain-Based Apps Using Familiar Software Patterns to Address Interoperability Challenges in Healthcare," *PLoP-24th Conf. Pattern Lang. Programs*, 2017.

[3] P. Zhang, M. A. Walker, J. White, D. C. Schmidt, and G. Lenz, "Metrics for assessing blockchain-based healthcare decentralized apps," 2017, doi: 10.1109/HealthCom.2017.8210842.

[4] X. Yue, H. Wang, D. Jin, M. Li, and W. Jiang, "Healthcare Data Gateways: Found Healthcare Intelligence on Blockchain with Novel Privacy Risk Control," *J. Med. Syst.*, 2016, doi: 10.1007/s10916-016-0574-6.

[5] A. M. Mosadeghrad, "Factors influencing healthcare service quality," *Int. J. Heal. Policy Manag.*, 2014, doi: 10.15171/ijhpm.2014.65.

[6] N. N. Latt *et al.*, "Healthcare in Myanmar," *Nagoya Journal of Medical Science*. 2016, doi: 10.18999/nagjms.78.2.123.

[7] E. Rojas, J. Munoz-Gama, M. Sepúlveda, and D. Capurro, "Process mining in healthcare: A literature review," *Journal of Biomedical Informatics*. 2016, doi: 10.1016/j.jbi.2016.04.007.

[8] M. Sekhon, M. Cartwright, and J. J. Francis, "Acceptability of healthcare interventions: An overview of reviews and development of a theoretical framework," *BMC Health Serv. Res.*, 2017, doi: 10.1186/s12913-017-2031-8.

[9] R. R. Morubagal, S. G. Shivappa, R. P. Mahale, and S. M. Neelambike, "Study of bacterial flora associated with mobile phones of healthcare workers and non-healthcare workers," *Iran. J. Microbiol.*, 2017.

[10] C. S. Kruse, B. Frederick, T. Jacobson, and D. K. Monticone, "Cybersecurity in healthcare: A systematic review of modern threats and trends," *Technology and Health Care*. 2017, doi: 10.3233/THC-161263.