
TOWARDS RELIABLE AND TRANSPARENT VACCINE PHASE III TRIALS WITH SMART CONTRACTS

A PREPRINT

Ivan da Silva Sendin
School of Computer Science
Federal University of Uberlandia
sendin@ufu.br

Rodrigo Sanches Miani
School of Computer Science
Federal University of Uberlandia
miani@ufu.br

February 16, 2021

ABSTRACT

Transforming a vaccine concept into a real vaccine product is a complicated process and includes finding suitable antigens and regulatory, technical, and manufacturing obstacles. A relevant issue within this scope is the clinical trial process. Monitoring and ensuring the integrity of trial data using the traditional system is not always feasible. The search for a vaccine against the coronavirus SARS-CoV-2 illustrates this situation. The scientific credibility of findings from several vaccines' clinical trials contributed to distorted perceptions concerning the benefits and risks of the drug. This scenario is ideal for applying technologies such as Blockchain and Smart Contracts in healthcare issues. This paper proposes a protocol based on Smart Contracts, named VaccSC, to enable transparency, accounting, and confidentiality to Phase III of vaccine experiments. The protocol was implemented in Solidity language, and results show that the VaccSC enables double-blindness, randomization, and the auditability of clinical data, even in the presence of dishonest participants.

1 Introduction

The COVID-19 pandemic that hit the world in 2019/2020 [6] is undoubtedly a disruptive event. Basically, all aspects of our society have been affected - and still will be - due to this pandemic. During the COVID-19 pandemic, vaccine development underwent irreversible changes. Before 2019, it was expected that a vaccine would take a few years or even a decade to develop and be considered safe, efficient, and made available to the market [4]. Contrary to initial expectations, several laboratories [7] released advanced studies of COVID-19 vaccine candidates in less than a year of research. This speed may raise doubts about the honesty and rigor of these scientific experiments.

Before regulatory approval, a vaccine candidate usually experiences three phases of development in humans: Phase I, Phase II, and Phase III. During Phase III large-scale trials, the human subjects are divided into two groups: i) those who received the vaccine and ii) those who received a control, a placebo, for example. The main outcome of this phase is the vaccine efficiency which is calculated by comparing the incidence of disease in vaccinated subjects and the incidence of disease in unvaccinated subjects (control groups). Unfortunately, this process is not transparent to society as a whole, which can compromise acceptance of the vaccine by some individuals [14, 11].

While transparency is important, the confidentiality of emerging data is also relevant: vaccine developers should not have access to them with the risk of changes in the trial design and compromising the final research results [12]. The recent technologies of Blockchain and Smart Contracts are primarily known for adding a layer of "trust" on the Internet services, as they offer decentralization, immutability, and public access to stored data [25, 29].

The use of Blockchain and Smart Contracts in the health area is not new. Several studies point out the possibility of using these technologies in healthcare systems[30, 22], diagnostics tracking[5], vaccine and medication trial tracking[27, 2]. More recently, the so-called "COVID-19 passports" were addressed by these technologies [8], [10] [23]. The previously cited approaches maintain trust "out of the contract". In other words, the data is stored in the

contract, and participants use it with a notary system. Therefore, the contract cannot interfere with the integrity of the stored information.

In this work we present a protocol based on smart contracts and Commitment Schemas to enable transparency, accounting and confidentiality to Phase III of vaccine experiments. Our work’s main contribution for the use of Smart Contracts in vaccine trials is the use of Commitment Schemas to enable the properties of double-blind, randomization and the auditability of clinical data, even in the presence of dishonest participants. Also, the Smart Contracts can determine whether the experiment has reached the desired efficiency and decide on the vaccine approval.

The rest of paper is organized as follows. In Section 2 we outline the theoretical background of our proposal which includes a description of Commitment Schemas and Smart Contracts. We briefly describe the vaccine development process in Section 3. Section 4 details the proposed protocol called **VaccSC**. Section 5 summarizes related work and Section 9 concludes the paper.

2 Background

2.1 Commitment Schemes and Coin Flipping

Commitment Schemas emerged in the early 1980s [3] and are used when a party \mathcal{A} needs to commit to a certain amount v with another party \mathcal{B} without revealing the value of v immediately.

A common construction of Commitment Schemas is done using cryptographic hashing functions and has two phases:

Commit Phase In this phase, \mathcal{A} commits to the value v generating a nonce¹ n and producing

$$C = H(v|n),$$

where $|$ indicates the concatenation operation and H is a cryptographic hash function. The value C - the commit - is then made public;

Reveal Phase To reveal the committed amount, \mathcal{A} sends to \mathcal{B} v and n , which calculates the hash value of the received data, and checks whether the generated hash is equal to C .

We say that C has the properties of **bidding**, since \mathcal{A} cannot create v' and n' that produce the same C ; and **hidding** because from C it is not possible to determine v . The value C can be seen as a *sealed* version of the value n in the sense that once C has been published, \mathcal{B} does not know v and \mathcal{A} can no longer change v without being discovered. See [9] for a review and security recommendations.

Using the Commitment Schemas, one can build a protocol to establish a pseudo-random number between parties that do not trust each other. This protocol is known as "coin flipping over telephone". To generate a pseudo-random bit, parts \mathcal{A} and \mathcal{B} follow the following steps:

- \mathcal{A} (and \mathcal{B}), privately, generate a pseudo-random bit $R_{\mathcal{A}}(R_{\mathcal{B}})$;
- $\mathcal{A}(\mathcal{B})$ commits and reveals $R_{\mathcal{A}}(R_{\mathcal{B}})$ using a Commitment Scheme;
- Now, \mathcal{A} and \mathcal{B} know both $R_{\mathcal{A}}$ and $R_{\mathcal{B}}$, and both can calculate $R = R_{\mathcal{A}} \oplus R_{\mathcal{B}}$;
- The resulting bit R is a pseudo-random bit shared by \mathcal{A} and \mathcal{B} .

One of the parties acting honestly is enough for the bit produced by the protocol to be random[3].

2.2 Smart Contracts

Introduced by Nick Szabo in [26], Smart Contracts (**SC**) are programs whose correct execution cannot be prevented. A **SC** can be compared to vending machines: with a coin and the press of a button, the user gets the expected product with a certain degree of confidence without dependence on other entities.

Currently, the **SC** platforms are obtained through P2P networks that execute contracts - also called *dapps* - redundantly. The participating nodes in this network receive financial incentives for the correct execution of these contracts. The high degree of redundancy and the economic stimulus produces an extremely reliable virtual computer.

The main **SC** platform currently available is Ethereum [28], which uses its own cryptocurrency, called Ether, in its transactions and to reward participating nodes. Ethereum is permissionless, meaning that anyone can access the

¹Nonce is a pseudo-random number used only once in some cryptographic protocol.

network. Users at the Ethereum platform are identified by an address that is essentially a public key. Access to the system, such as transferring values or invoking methods in contracts, is done through signed messages using the private key corresponding to Ethereum addresses. The confidence in the correct execution of programs and the extensive use of cryptographic protocols allow **SC** to be secure, transparent, and auditable.

The applications of **SC** are countless: financial services, games, betting, and network infrastructure are some examples². The capability to enforce rules, accurately described by a programming language, allows the adoption of **SC** in other application scenarios such as public administration [13] and regulated activities [1].

3 Vaccine development process

The vaccine development goes through an elaborate development process after the discovery stage [24]. It is common to divide this process into preclinical and clinical stages. The clinical-stage is defined by several clinical trials in human subjects organized by three phases. The objective of phase I is to evaluate the vaccine’s safety in a small sample of human volunteers. During phase II, the goal is to demonstrate immunogenicity vaccine by assessing the impact of multiple variables on immune response, such as age, ethnicity, gender, and presence of maternal or pre-existing antibodies. The sample of human volunteers in phase II varies from hundreds to thousands of subjects. Phase III trials are large-scale clinical trials and essential for registration and approval to license the vaccine. If the phase III results demonstrate efficacy and safety, the vaccine manufacturer can apply to the national regulatory authority to approve and market the product. The primary outcome is vaccine efficiency (VE), which can be defined as [16]:

$$VE = (1 - RR) * 100\% \quad (1)$$

where RR represents risk reduction (or relative risk). RR is the ratio between the incidence of disease in vaccinated subjects and disease incidence in unvaccinated subjects. Considering AR_1 the number of infected cases among the subjects in the investigational vaccine group and AR_0 the number of infected cases among the control subjects then

$$RR = \frac{AR_0 - AR_1}{AR_0}. \quad (2)$$

This can be seen as a comparison of who received the vaccine and who received a placebo. To illustrate how the RR is calculated we will use the [21]: a threshold of 164 contamination confirmations was determined for this experiment. If, for example, after 164 infections it is found that 120 were from people who received the control and 44 from people who received the vaccine, the RR of this vaccine would be approximately 0.63.

The disease’s occurrence is the most common endpoint; however, the trial may be based on other clinical endpoints, such as incidence of infection or immunological correlations of protection. After completing Phase III trials and following licensure of the product, Phase IV studies, also referred to as postmarketing surveillance studies (PMS), are used to monitor the vaccine for safety and population effectiveness.

The well-accepted standard for carrying out vaccination experiments is conducted in the randomized controlled, double-blind trial system [15]. In this system, the person responsible for applying the vaccine randomly chooses what will be applied to each individual, not knowing its content (vaccine or control). Also, the subject who receives the shot does not know what he/she is receiving. This procedure aims to avoid bias in the vaccination process. Eventually, a dishonest vaccine company would like to improve its product’s efficiency by selecting a healthier subpopulation to receive the vaccination and a less healthy population to receive control. Similarly, the person receiving the shot should not know its content to avoid the experiment’s placebo effect.

4 Proposed Protocol

In this session, we present **VaccSC**: a **SC** based protocol to track a Phase III vaccine trial providing transparency and double-blind behavior.

For the sake of simplicity, the proposed protocol models the participation of three types of entities:

Vaccine Developer Responsible for distributing vaccines to clinics and protecting information about their content. He is also responsible for deploying the contract, being the only participant who can distinguish the real vaccine from the control, creates the commits for the shots, which will be used as identifiers in the contract;

Vaccine Clinic Apply the shots and updates the **VaccSC** bidding the shot with a specific patient;

²See <https://etherscan.io/dapp> for an extensive list of *dapps*.

Information	Type	Access and Control
Commit	hash	Immutable, crated by Vaccine Developer
Clinic	Address	Each clinic is added by Vaccine Developer
Patient	Address	Clinic and patient
gotSick	Boolean	Patient
Vaccine Type	Vaccine/Placebo	VaccSC fills field with the <i>reveal</i> provided by Vaccine Developer

Table 1: Data structure used by **VaccSC** to keep vaccine trial information.

Patient Receives the shot and notifies the contract when he/she becomes ill.

We assume that all participants have an Ethereum address and keep the corresponding private key securely. This step can be achieved using a smartphone, for example.

4.1 Random Patient/Shot Assignment

As stated earlier, the random association between the shot and the patient prevents bias. The generation of random numbers on **SC** environment is complicated and must be done carefully. In this protocol, we chose to use the approach in which each participant generates a random number (R_1 and R_2). The **SC** combines these numbers using bitwise **XOR** operation. With this approach, one part being honest is enough for the number to be random. This process is described in Figure 1.

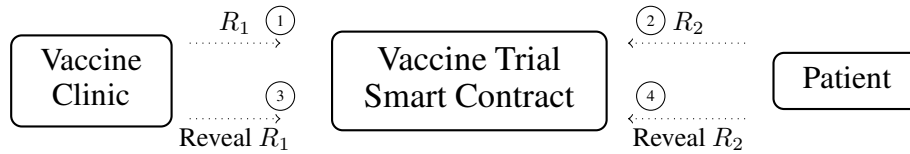


Figure 1: Each participant privately generates a random number and commits this number. After both numbers are committed to the contract, each participant reveals the value (steps 3 and 4). The **SC** combines these numbers - $R_1 \otimes R_2$ - to associate one of the shots available to the patient.

4.2 Double Blindness

Here we describe how the **SC** guarantees Double Blindness property. The Vaccine Clinic does not know what he is applying as much as the patient does not know if he receives the vaccine or a control shot. Initially, Vaccine Developer protects the information on the content of each shot that will be part of Phase III as described in Section 2.1:

$$C_i = H(n_i|v_i),$$

the value C_i is used as a shot identifier by the contract and must be physically associated with the shot. The values n_i and v_i are saved and used later to reveal the shot's content and calculate the vaccine effectiveness.

Also, when deploying the contract, Vaccine Developer informs:

Number of participants: corresponding to the number of available shots;

Infected Threshold: number of patients who need to be infected before the committed values are revealed and the vaccine's efficiency is calculated;

Target efficiency: minimum VE required in order to vaccine be approved;

Vaccination Clinics: the Ethereum address of clinics that will apply the shots and update the **VaccSC**.

The data structure that stores the necessary information to calculate vaccine efficiency are shown in Table 1. The fields are presented in chronological order in which they are changed during the execution of the contract.

4.3 Execution

Once the Vaccine Developer deployed the **VaccSC**, the protocol follows the following steps:

1. The Vaccine Developer delivers the shots to each Vaccination Clinic and makes the association between the shots and the clinics in the contract;
2. The clinic applies the shot and associates the shot identifier with a specific patient. The patient confirms this to contract;
3. If the patient becomes ill, he communicates the fact to the contract;
4. When the infected threshold is reached, the contract creates an event informing the Vaccine Developer;
5. Using the Ethereum addresses of infected patients, Vaccine Developer selects the corresponding commits. After that, the Vaccine Developer sends information about patients who **received the control** to the contract. The contract calculates the number of patients who received the vaccine;
6. Vaccine efficiency can be determined (using Equation 2), and its approval status can be obtained directly from the contract by anyone.

An overview of the protocol is shown in Figure 2.

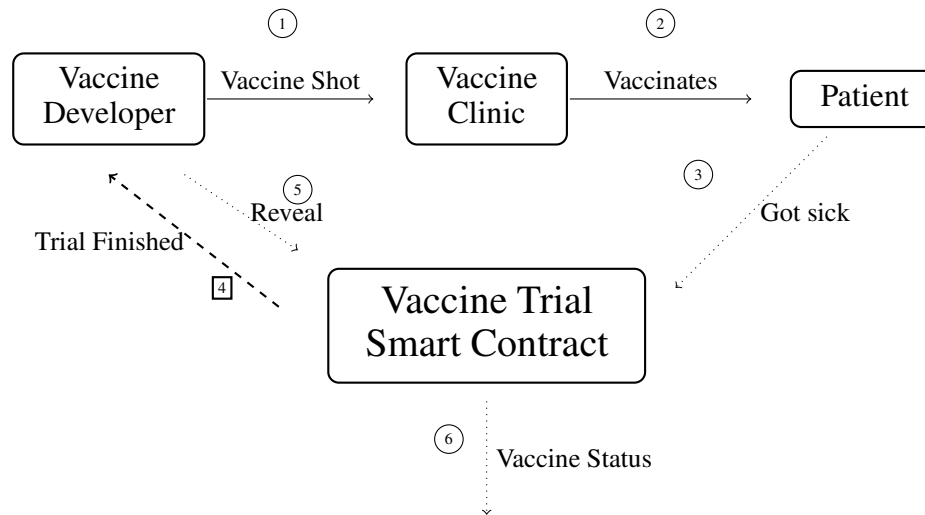


Figure 2: Protocol events associated with one shot. Step four occurs only once. Solid lines indicates physical events which contract updates, and dashed lines indicates **VaccSC** interactions.

4.4 Security analysis

In the associating vaccines to patients phase, both participants are equally responsible for generating the random number, and one of the participants acting honestly is sufficient to produce a random number. If the two conspire to choose a specific shot, they still cannot choose between vaccine or control, as the Vaccine Developer protected this information.

The double blindness property has the following analysis:

Patient The action expected by the patient is to communicate his own disease event to the contract. Patients do not know about their vaccination status, so they cannot effectively change the results of the contract;

Vaccine Clinic As he applies the shot but does not know its content, he is also unable to defraud the protocol;

Vaccine Developer A dishonest developer can improve the efficiency of the vaccine by decreasing the number of people who have been effectively vaccinated and have become ill. Using the proposed protocol, he cannot do that because his action will precisely reveal the patients who received a placebo, so the omission is harmful to him. Therefore, the forging of "control patients" does not occur because the shot status is committed.

4.5 Availability

The **VaccSC** protocol was implemented in Solidity language, the client side applications were developed using the Brownie suite³ and Python language. The contract is available for download at

³<https://github.com/eth-brownie>

<https://github.com/ivansendin/Vaccine>. Commit Scheme and Coin Tossing libraries are available at <https://github.com/ivansendin/CommitExamples>.

5 Related Work

Early attempts to apply **SC** to conduct clinical experiments use **SC** to store data securely, guaranteeing the data's authenticity and integrity through the access control and digital signatures provided by the platform. There are several works available with this approach [17, 2, 18, 27, 19, 20].

[17] used **SC** as a solution to data manipulation issues common to clinical trials. The authors propose a hierarchical arrangement of two core types of smart contracts: i) a regulator contract, holding a data structure containing clinical trial authorization (CTA) details, and ii) a trial contract deployed by contract research organizations (CROs). The trial contract's logic effectively enforces aspects of the trial protocol, securing that neither subjects nor measurements are added outside the trial timelines. At the same time, the tamper-resistant characteristics of the blockchain prevent data manipulation. Authors deployed contracts using a private Ethereum blockchain with synthetic data representing the clinical trials of Tamiflu, an influenza drug. They were able to query the state of data, such as the number of trials underway, the number of subjects recruited, and the timestamp at which the transaction was processed.

[2] attempted to clarify the three main principles of using blockchain in clinical trials: timestamping, time-ordering and smart-contracting. Regarding **SC**, authors focus on providing examples of how it can be applied to clinical trials. According to them, **SC** can be applied in the entire process. The case of randomization of a patient in a clinical trial before written consent is briefly discussed as a potential application. The paper does not provide any experiments or **SC** implementations.

Omar et al. [18] and [19] proposed a **SC** using Ethereum to tackle data management challenges in clinical trials. The proposed **SC** initially captured three stages of a clinical trial process: new drug application, clinical trial initiation, and patient enrollment stages. [19] discussed patient monitoring and severe adverse event occurrence. The smart contract was written in Solidity and captures interactions among clinical trial stakeholders such as the regulatory agency, drug sponsors, physicians, and patients. For instance, for the new drug application scenario, the smart contract implements the process in which the sponsor obtains permission from the FDA (Federal Drug Agency) to test their drug in humans. The contract has a function that requests the sponsor to upload the required documentation. When the documentation status is set to true, the FDA announces the approval or rejection of the request.

[27] developed a blockchain-based web portal to tackle challenges in the current clinical trial process. The designed portal is accessible to all interested parties, facilitating the interaction between patient and clinical investigator. The portal also enables expediting adverse event reporting. The authors modeled a phase II clinical trial during their experiments. They simulated how a previously completed clinical trial (omalizumab drug) would behave using blockchain software technologies. Despite arguing that forcing all participating parties to use a service like that is still a challenge, they showed that data entry, storage, and adverse event reporting could be performed more robustly and securely.

[20] addressed the issue of centralized vaccine production supervision. The authors proposed a decentralized blockchain-based method to enforce privacy protection of vaccine production records. The blockchain structure is divided into two levels. The first one is the private data of the pharmaceutical company and consists of production records. The second one is public data. A public blockchain is used to store this data. Every transaction in the block represents a process for each batch of vaccine within the production process. Each transaction also has its timestamp and signature of the company and the hash value of the previous transaction to prevent production data fraud. The authors were able to protect the privacy of the company's production of vaccines using this structure. The proposed framework was implemented using the Hyperledger Fabric blockchain. The authors only evaluated the spatio-temporal efficiency (impact of the transaction against latency, throughput, and blockchain size) of the method.

6 Conclusion

We presented **VaccSC**: a protocol based on Smart Contracts that brings transparency and reliability to Phase III of vaccine trials while maintaining the double-blind property present in existing protocols. The protocol prevents the Vaccine Developer from hiding relevant information to vaccine trials and allows anyone to audit the trial data. The protocol achieves these conflicting requirements using Commitment Schemas.

The presented contract models a simplified version of the real world, other scenarios - such as the presence of observers to ratify the events - can be easily added to the contract. Ultimately, the proposed protocol captures the essence of Smart Contracts - "Code is law" - and can be used as a deciding element on the approval of a vaccine, whether by a country or by an individual.

References

- [1] Merlinda Andoni, Valentin Robu, David Flynn, Simone Abram, Dale Geach, David Jenkins, Peter McCallum, and Andrew Peacock. Blockchain technology in the energy sector: A systematic review of challenges and opportunities. *Renewable and Sustainable Energy Reviews*, 100(October 2018):143–174, 2019.
- [2] Mehdi Benchoufi and Philippe Ravaud. Blockchain technology for improving clinical research quality. *Trials*, 18(1):1–5, 2017.
- [3] Manuel Blum. Coin flipping by telephone a protocol for solving impossible problems. *ACM SIGACT News*, 15(1):23–27, 1983.
- [4] Barry C Buckland. The process development challenge for a new vaccine. *Nature medicine*, 11(4):S16–S19, 2005.
- [5] Hiten Choudhury, Bidisha Goswami, and Sameer Kumar Gurung. CovidChain: An Anonymity Preserving Blockchain Based Framework for Protection Against Covid-19. *arXiv*, 2020.
- [6] Ensheng Dong, Hongru Du, and Lauren Gardner. An interactive web-based dashboard to track COVID-19 in real time. *The Lancet Infectious Diseases*, 20(5):533–534, may 2020.
- [7] Peter Doshi. Covid-19 vaccine trial protocols released. *The BMJ*, 371:10–11, 2020.
- [8] Marc Eisenstadt, Manoharan Ramachandran, Niaz Chowdhury, Allan Third, and John Domingue. Covid-19 antibody test / vaccination certification there’s an app for that. *arXiv*, 1:148–155, 2020.
- [9] Oded Goldreich. *Foundations of Cryptography: Basic Tools*. Cambridge University Press, USA, 2001.
- [10] Harry Halpin. A Critique of Immunity Passports and W3C Decentralized Identifiers. 6736(20), 2020.
- [11] Emily A. Harrison and Julia W. Wu. Vaccine confidence in the time of COVID-19. *European Journal of Epidemiology*, 35(4):325–330, 2020.
- [12] Philip R Krause, Thomas R Fleming, Susan S Ellenberg, Ana Maria Henao-Restrepo, PR Krause, T Fleming, MM Alejandria, B Bhargava, S Ellenberg, PJ Garcia, E Halloran, AM Henao-Restrepo, I Longini, MC Miranda Montoya, M Roses, and H Sommerfelt. Maintaining confidentiality of emerging results in COVID-19 vaccine trials is essential. *The Lancet*, 6736(20):1611–1613, 2020.
- [13] Magnus Krogsbøll, Liv Hartoft Borre, Tijs Slaats, and Søren Debois. Smart Contracts for Government Processes: Case Study and Prototype Implementation (Short Paper). *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12059 LNCS:676–684, 2020.
- [14] Elisabeth Mahase. Covid-19: Vaccine trials need more transparency to enable scrutiny and earn public trust, say experts. *The BMJ*, 371:1–3, 2020.
- [15] John N.S. Matthews. *An introduction to randomized controlled clinical trials*. Chapman & Hall/CRC, 2006.
- [16] Jozef Nauta. *Statistics in Clinical Vaccine Trials*. Springer-Verlag Berlin Heidelberg, 2011.
- [17] Timothy Nugent, David Upton, and Mihai Cimpoesu. Improving data transparency in clinical trials using blockchain smart contracts. *F1000Research*, 5, 2016.
- [18] IA Omar, R Jayaraman, K Salah, and MCE Simsekler. Exploiting ethereum smart contracts for clinical trial management. In *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–6. IEEE, 2019.
- [19] Ilhaam A Omar, Raja Jayaraman, Khaled Salah, Mecit Can Emre Simsekler, Ibrar Yaqoob, and Samer Ellahham. Ensuring protocol compliance and data transparency in clinical trials using blockchain smart contracts. *BMC Medical Research Methodology*, 20(1):1–17, 2020.
- [20] Shaoliang Peng, Xing Hu, Jinglin Zhang, Xiaolan Xie, Chengnian Long, Zhihui Tian, and Hongbo Jiang. An efficient double-layer blockchain method for vaccine production supervision. *IEEE Transactions on NanoBio-science*, 19(3):579–587, 2020.
- [21] Pfizer. Pf-07302048 (bnt162 rna-based covid-19 vaccines) protocol c4591001. https://pfe-pfizercom-d8-prod.s3.amazonaws.com/2020-09/C4591001_Clinical_Protocol.pdf, 2020.
- [22] Gaganjeet Singh Reen, Manasi Mohandas, and S. Venkatesan. Decentralized patient centric e-Health record management system using blockchain and IPFS. *2019 IEEE Conference on Information and Communication Technology, CICT 2019*, 2019.
- [23] Kiarash Shamsi, Koosha Esmailzadeh Khorasani, and Mohammad Javad Shayegan. A Secure and Efficient Approach for Issuing KYC Token As COVID-19 Health Certificate Based on Stellar Blockchain Network. 2020.

- [24] K Singh and S Mehta. The clinical development process for a novel preventive vaccine: An overview. *Journal of postgraduate medicine*, 62(1):4, 2016.
- [25] M. Swan. *Blockchain: Blueprint for a New Economy*. O’Reilly Media, 2015.
- [26] Nick Szabo. Formalizing and Securing Relationships on Public Networks. *First Monday*, 2(9), 1997.
- [27] Daniel R Wong, Sanchita Bhattacharya, and Atul J Butte. Prototype of running clinical trials in an untrustworthy environment using blockchain. *Nature communications*, 10(1):1–8, 2019.
- [28] Gavin Wood et al. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151(2014):1–32, 2014.
- [29] Xiwei Xu, Ingo Weber, and Mark Staples. *Architecture for Blockchain Applications*. Springer Publishing Company, Incorporated, 1st edition, 2019.
- [30] Rengeng Zou, Xixiang Lv, and Jingsong Zhao. Spchain: Blockchain-based medical data sharing and privacy-preserving ehealth system, 2020.