Blockchain-based reputation system for peer reviewing

Master's Thesis

Viktor Jacynycz García

Directed by

Samer Hassan Collado and Antonio Sánchez Ruiz-Granados

Máster en ingeniería informática

Master's degree in software engineering

Facultad de Informática

Universidad Complutense de Madrid

2017/2018 academic year

Document made with TEXIS v.1.1+. modified by Viktor Jacynycz García

This document is prepared for duble-side printing.

Blockchain-based reputation system for peer reviewing

Memoria que presenta para optar al master en ingeniería informática

Viktor Jacynycz García

Directed by

Samer Hassan Collado and Antonio Sánchez Ruiz-Granados

Máster en ingeniería informática
Master's degree in software engineering
Facultad de Informática
Universidad Complutense de Madrid

2017/2018 academic year

Copyright ©Viktor Jacynycz García

El/la abajo firmante, matriculado/a en el Máster en Ingeniería Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: *Blockchain-based reputation system for peer reviewing*, realizado durante el curso académico 2017-2018 bajo la dirección de Samer Hassan Collado y Antonio Sanchez Ruiz-Granados en el Departamento de ISIA, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en internet y garantizar su preservación y acceso a largo plazo.

A mi madre, porque sí

A mi hermano, porque también

A jenny, poque tl

A mi padre, porque yokse

Agradecimientos

I find your lack of faith disturbing

Darth Vader, Star Wars: A New Hope.

I thank you.

Abstract

...

...

... Resumen chulo

Index

Index of figures

Index of tables

Chapter 1

Introduction

Who Watches the Watchmen?

Watchmen - Alan Moore

Scientific research nowadays is based on publications in journals with a high impact factor [1], the most well-known is the Journal Citation Reports (*JCR*). This factor was originally determined by the Science Citation Index, born in 1955 [2] but nowadays managed by a private company dedicated to benefit from the work of researchers [3]. This poses two important problems when it comes to entering the world of academic research:

The first one is that scientific journals that want to maintain their impact factor have to make sure that the articles that come out in their issues have a large number of citations, so they are always going to look for novel and high-impact articles. Therefore, the editors of these journals will have a network of reviewers on which they can trust to review the article. But sometimes these reviews are not entirely objective, since there are many cases of unfavorable reviews due to gender causes, especially in scientific fields [4].

Besides, it is necessary to consider that the time of revision for an article is excessively long, causing the process of academic investigation to be quite slow [5].

The second problem is that the benefits of scientific distribution are centralized in publication systems, nor the authors, the reviewers or the readers get money from it. Today, with electronic paper distribution, universities purchase site licenses for online access to journal contents. This system implies an additional cost for the universities who want to advance in their research fields and do not have enough money for it. However, site licenses are not always disadvantageous. Some journals issued by

private companies and universities adjust their prices to maximize subscriptions [6]. But generally, people who earn money from this paper-based system only act as an intermediary between the authors, the reviewers and the readers.

The internet offers the possibility to meet people all around the world, and when it comes to trust total strangers, you should have a system in which you can rely on to deposit your trust in them. Reputation systems are the solution to these problems, since they offer a good first impression about an unknown person [7].

Editors who want to assign the review of a paper to a series of reviewers have to rely on them beforehand. Thus, limiting the spectrum of fields that can be revised to the fields in which those reviewers are experts. If you want to broaden the scope of reviewers with more fields of expertise, you need to contact new reviewers. But there is no easy way to predict reviewer quality from their training and experience factors [8], so a rating system of reviewers would be useful for journals to select the best reviewers. The solution is a reviewer reputation network, in which reviewers get rated based on their reviews and build up their reputation based on good practices and helpful reviews. In this network, publishers who have to find new reviewers for their papers do not have to know them beforehand, since trust is placed in the reputation network instead of in the person itself.

Science publication and peer review are based on a paper-based paradigm, with only a few changes in the last centuries [9]. Critics to current science publication and peer review systems include concerns about its fairness [4], quality [10], performance [5], cost [6], and accuracy of its evaluation processes [1], among others.

The development of the Internet enabled the proposal of alternatives for science dissemination [11] and evaluation [12]. The reduction of distribution costs enabled a wider access to scientific knowledge, and questioned the role of traditional publishers [13]. It is acknowledged that the Open Access and Open Science movements have successfully reduced the economic cost of accessing knowledge to readers [14]. However, it has not successfully challenged traditional publishers' business models [15], who are now combining charging readers and charging authors [16].

Peer review has suffered multiple criticism, and yet only marginal alternatives have gathered success [17]. The literature provides multiple proposals around open peer review [18], and proposals of reputation networks for reviewers [19]. In fact, a start-up, Publons[1], provides a p latform to acknowledge reviews and open them up.

[1] https://publons.com/

1.1. Objective

We aim to challenge middlemen such as traditional publishers in science publication. Particularly, we propose a decentralized publication system for open science, allowing 1) paper submissions, 2) assignment of reviewers, 3) peer review and, as a novelty, 4) the rating of peer reviews. With this distributed system, we aim to improve the quality and efficiency of reviews and knowledge distribution, helping editors, authors, and reviewers:

- Editors and journals will be able to find the best peer reviewers in their fields of interest, and also those that respond quickly. Thus, reducing time-to-publish and publishing costs.
- Authors will be able to submit papers to time-responsive, free, open access journals, and forget about slow, unfair and unaccountable anonymous reviews.
- Reviewers will finally have their work recognized.

We are interested in exploring the following challenges, that could be dealt with our technology:

- Reduce time-to-review by rewarding on-time reviewers.
- Measure and prevent sexism, nepotism and other abuses in peer review.
- Develop fully autonomous decentralized journals.
- Explore fully free publication systems for Open Access science, while enabling innovative business models.
- Explore alternative and open metrics for papers, journals and reviewers.

1.2. In this work

In this work there will be the following sections:

- **State of the art:** This chapter is about what methods, systems and technologies try to change the actual publication systems and how good or bad they are.

- **Methodology and Technology:** This chapter is about the methodology I followed during the realization of this work, and the technologies I have used to face the challenges and why I decided to use them.
- **Platform description:** This is the main chapter of this work. It contains the platform description, its implementation, how it works, why it is better than the actual publication systems, the challenges I have faced during the realization and the internal structure of the final system.
- **Results and discussion:** This chapter is about the results obtained after the realization of the work proposed in this project, how it will affect the scientific community and how I measured the potential impact if it becomes a wide-used publication system.
- **Conclusion and future work:** This chapter is about the implications of this work in the scientific community and settles the next steps to follow to create an ecosystem of autonomous publication systems without the need of middlemen such as journals or editors, proposing a future PhD about this subject.

Chapter 2

State of the art

The needs of the many outweigh the needs of the few

Spock - The Wrath of Khan

2.1. Alternative Publication systems

Publication systems, as seen on section 1 are vampirizing the industry. However, there are some attempts to change this paradigm on behalf of science dissemination.

Open journal systems [20] is an open software designed to facilitate the publishing process. This project was created by the Public Knowledge Project[1] and it targets open-access online journals that want to speed up the publication processes. The system provides tools to control the whole publishing process from article submission, through peer reviewing to the final publication issue.

Mega-journals (or Multi-journals) [21; 22] combine multiple journals into a single journal, allowing the publication of open-access papers, which have gone through a peer review process. The first journal to adopt this idea was the *PLOS ONE* Journal[2] as of the project *Public Library of Science*. This project aims to create a library of scientific journals under the values of open access and creative commons licenses. As a result of the success of the *PLOS ONE* journal, other publishers have started their own mega-journals. Featuring alternative impact metrics, reusability of figures and data, post-publication discussions and portable reviews from other journals [23].

[1] https://pkp.sfu.ca/about/

[2] http://journals.plos.org/plosone/

The continuous publication model is based on publishing individual papers migrating from the previous issue-based model [24]. This method is seen as an altenative for open-access journals as it speeds up the publication process [25]. *DPSOS*[3] adopts this model by design (see section 3.2.2.2) as it publish automatically papers that meet certain preconditions that are written in the blockchain.

Preprints are scientific papers that have not yet gone through the peer review process [26]. Formerly, the preprints that were sent to the journals were private, and only accessible by the editors and assigned reviewers. But nowadays it is common to publish a preprint before sending it to a journal, uploading it to specialized platforms like arXiv[4] or Preprints[5] [27]. In fact there is a correlation between the upload of a preprint and early citations after the publication of the paper [28]. This system is a possible solution to the cold-start problem that papers of new researchers who enter the academic career have [29].

Social networks have also made a dent in the academic world, creating platforms to contact other researchers and encouraging them to share open access papers. Some of the well-known are Research Gate[6], Mendeley[7] or Academia[8]. But despite the good intentions of the creators of these platforms, many of the journals demand the copyright of the papers they publish, preventing the authors from sharing them through these services.

Decentralized alternatives, in spite of their promises [30], are still in their infancy. A few proposals, none of them functional to date, have appeared recently.

One of them is a peer review proposal that tries to solve some of the peer review socio-technical problems using cryptocurrencies [31]. It needs a critical threshold of research community engagement, changing the actual processes and platforms, to start being implemented.

Blockchain-enabled apps have also been proposed, with voting and storage of publications. This is the case of Aletheia [32], a software for getting open access papers published. This platform idea aims to use blockchain as a decentralized and distributed database as a publishing platform.

Peer review quality control through blockchain-based cohort trainings [33] have been also proposed, with the promise of transparency and decentralization using a

[3]Decentralized Publication System for Open Science
[4]https://arxiv.org/
[5]https://www.preprints.org/
[6]https://www.researchgate.net
[7]https://www.mendeley.com
[8]http://academia.edu

distributed ledger. Research labs can use this training network to test their technology and reduce the risk for private investment opportunities.

Finally, some of the off-chain journals are adapting to the demands of the current scientific community like Ledger[9], a cryptocurrencies and blockchain-based journal that records the publication timestamps in the Bitcoin blockchain.

2.2. Reputation systems

Reputation systems today arise from the need to trust unknown individuals [7]. Many of the big internet communities like Stackexchange[10] or reddit[11] have their own reputation system. Reputation systems behavior may vary depending on the platform [34], but the most usual is the one where users get a score based on certain interaction with the community.

Reputation systems also have a very large niche in e-commerce webs such as Ebay[12], in which people pay for a product sold by an unknown vendor. There must be a previous trust in the vendor before buying any product, so a reputation system offers a score given by other users that encourages you to trust or not that certain seller [35].

Reputation systems vary widely in scope, such as one for peer-to-peer computing [36], vehicle ad-hoc [37], web services [38] and even Wikipedia [39]. All of them are based on an exchange of trust between users who use these services.

This same concept was intended to be transferred within the blokchain using a token as a trust unit, which users exchanged as a sign of trust deposits among them [40].

But reputation systems also have problems when it comes to defend the users from attacks to individuals [41] and unfair ratings [42], so the architecture chosen for it must consider these weak points and try to mitigate them.

This paper proposes the development of a decentralized publication system for open science. It aims to challenge the technical infrastructure that supports the middlemen role of traditional publishers. Due to the successes of the Open Access movement, some of the scientific knowledge is today freely provided by the publishers. However, the content is still mostly served from their infrastructure (i.e. servers,

[9] https://ledgerjournal.org
[10] https://stackexchange.com/
[11] https://www.reddit.com/
[12] https://www.ebay.com

web platforms). This ownership of the infrastructure gives them a position of power over the scientific community which produces the contents [43]. Such central and oligopolistic position in science dissemination allows them to impose policies (e.g. copyright ownership, Open Access prices) and concentrate profits.

The proposed system aims to move the infrastructure control from the publishers to the scientific community. It entails the decentralization of three essential functions of science dissemination: 1) the peer review process, 2) the selection and recognition of peer reviewers, and 3) the distribution of scientific knowledge. The following section provides an overview of the system features, while the final section discusses its challenges.

Chapter 3

Methodology & Technology

The needs of the many outweigh the needs of the few

Spock - The Wrath of Khan

3.1. Methodology

The idea of this project came up in a Hackathon in September 2017. We were a group of 4 developers with one month to create an idea and a small prototype to implement using blockchain technologies.

To give birth to the idea of a decentralized publication system for open science we used agile methodologies.

3.1.1. Brainstorming

Brainstorming was born as a method to increase creativity in groups and organizations. There are only few rules on this method: do not criticize any of the given ideas, quantity is desired over quality, try to combine suggested ideas and give all the ideas that come to mind, no matter if they are possible or not [44].

This method is used nowadays in companies and work groups as part of the process of the creatfile uploadsion of a product, although there are some critics about brainstorming and sometimes instead of encouragind creativity, inhibits it [45; 46]

Leaving apart these problems, we decided to make a brainstorming session to define what we were going to do. Many ideas emerged and were capture into a white board without discrimination, no matter how hard or easy to implement they were.

Figura 3.1: Image of the value proposition canvas after the session

After saying enough ideas to fill the board we filtered the ones that were impossible to achieve. Then, each one voted the best three, making a ranking of the 3-4 best projects to start working on. Some of the ideas were creating a distributed wikipedia with governance models, an application to contact people from minority groups in countries where they are persecuted collectives, a distributed and community driven NGO, and a crowdfunding platform for *whisteblowers*.

Finally we decided to create an approach to a distributed platform for open science.

3.1.2. Value proposition canvas

A value proposition canvas is a tool to create, design and implement a product idea. Is commonly used by businesses and entrepreneurs to find the balance between customer profile and product design, but there are other cases of use for this tool outside business scope [47; 48].

The process is divided in two parts, customer profile and value map, each of these

divided in other three parts: [49]:

- **Customer profile:** This step is to identify the profile of the final user of the platform. This section is divided in three parts: 1) *Customer jobs:* things the customer are trying to get done, 2) *Customer pains:* undesired costs and situations, 3) *Customer gains:* benefits, social gains and cost savings expected.

- **Value Map:** This section is about what the final product has to have and what does not, and its also divided in: 1) *Product and services:* which products and services are offered that help the customer get a job done, 2) *Pain relievers:* how the customer pains are going to be alleviated, 3) *Gain creators:* how the products and services create customer gains

We decided to use this methodology for the definition of the final platform, since it established the general development framework of the application.

3.1.3. Agile methodologies

Traditional software development methodologies are being eclipsed by new light or agile methodologies. These methodologies are characterized by continuous integration, iterative development and the ability to assume changes in business requirements [50; 51].

One of the most popular is known as Extreme Programming [52] based on a series of basic concepts when carrying out the development of a program: code simplicity and rapid prototyping, continuous customer communication with the development team, responsibility of the code of all the members of the group, short and quick meetings, refactoring and continuous integration [53].

Another well-known method within agile methodologies is scrum [54], which uses two-week frameworks to perform development sprints and planning meetings. The use of these methodologies allows developers to create better quality software in shorter periods of time and are designed for small teams from three to nine developers.

I used these two methodologies to develop the platform that we were defining. Every Saturday we had a weekly meeting where we were adding or removing functionality to the platform. Extreme programming allowed me to meet design demands without sacrificing development time. At the end of each session, we planned the whole week and set the date for the next meeting.

3.2. Technology

The proposed system relies upon two emerging distributed technologies. On the one hand, the Blockchain [55] provides a public decentralized ledger to record the system's interactions. On the other hand, IPFS [56] is a distributed file system to store all the papers and reviews sent to the platform. This ensures that all the information is persistent, free and accessible, and does not rely on a centralized server.

3.2.1. IPFS

IPFS stands for Interplanetary File System. It is a peer-to-peer file-sharing protocol that uses a cryptographic hashes to store files in a distributed network. IPFS works very similar to HTTP protocol but in a BitTorrent way. It can be seen as a giant git repository where everyone can store, share and exchange files[57].

IPFS merges four main ideas: Distributed Hash Tables, BitTorrent, Git and Self-Certified Systems.

3.2.1.1. Distributed Hash Tables

A distributed hash table(*DHT*) is a decentralized structure that works very similar to a hash table. Hash tables are used to identify items in a database. The table performs simple mathematical operations generating a random string called hash. The hash acts as a pointer that directs to the data, this allows the user to find data directly instead of looking through the entire database[58].

In a distributed hash table, any node can use a hash as a key to retrieve data. This system includes a data structure called "keyspace" that is a set of all possible keys, which is split up across the nodes in the system. The mapping of the keys is made by another function that describes the distance from one key to another. All the nodes have and identifier and a set of identifiers pointing to all its neighbors nodes. If a node is removed from the network, only a small portion of the data must be recovered by other nodes[58].

This system makes *DHTs* scalable, fast and robust. It is used by frameworks such as Tapestry [59], Chord [60], Kelips [61], Kademlia [62] and IPFS [57]. These platforms are similar in cost and performance if they are tested in a large enough network. They behave very fast when it comes to searching for a key through massive networks of nodes[63], that's why it is used by IPFS to create its distributed file

system.

3.2.1.2. BitTorrent - File sharing

BitTorrent [64] is a P2P file sharing system used worldwide. In this system, files are divides into very small chucks of data, and are shared in a peer-to-peer network. Each peer aims to maximize its download rate by connecting to low latency peers. In BitTorrent's network, peers with high upload rate will get higher download rate, so the key is balancing the network bandwidth between downloading and uploading files[65].

IPFS uses three main features from BitTorrent's protocol[57]:

- BitTorrent's data exchange protocol rewards nodes who contribute to the network, and punishes the ones who don't.
- BitTorrent tracks the availability of file chunks, sending the rarest first rather than sending the most common ones.
- IPFS uses PropShare[66] bandwidth allocation strategy to improve BitTorrent's behavior facing exploitable scenarios.

3.2.1.3. Git - Version control system

Git is a distributed version control system (*DVCS*)[67]. Git was born in 2005 when the development process of the Linux kernel lost its version control system. The Linux kernel is one of the biggest free software projects nowadays, it has a great team of developers behind and the code usually changes very frequently. In 2002 the team used BitKeeper as VCS since they had a free license. But in 2005 when this license was over, Linus Torvalds decided to develop his own VCS[68].

Git was designed to be scalable and distributed, and the most important factors that IPFS inherits from Git are: [57]:

- Git implements a Merkle Directed Acyclic Graph [69], an object that reflects changes in a file system in a distributed way.
- Objects are identified by the cryptographic hash of their contents.
- Version changes only update preferences and add objects. To broadcast version changes, git only needs to transfer the new objects and update the remote references.

3.2.1.4. Self-Certified File Systems

A self-certified file system (*SCFS*) is a secure file system that avoids internal key management, using public keys to map file names, making self-certifying pathnames. Key management occurs outside the system, letting the user choose the desired procedure to generate file names [70].

The name of an self-certified file system assure its server. Users can verify the public key offered by a *SCFS* server and negotiate a shared secret to secure all traffic.

IPFS tries to connect these ideas into a cohesive, trustful and decentralized file system. It is build on top of a peer-to-peer network, so no nodes are privileged, and all of those store IPFS objects in local storage. These objects represent files or other data structures.

inar explicar

3.2.2. Ethereum

Ethereum [55] is a very novel technology that allows the creation of distributed applications that run in an arbitrary large and trust-less network of nodes. Ethereum's strength rely on three main concepts: blockchain, smart contract and transactions.

3.2.2.1. Blockchain

In the decade of the 80s and the 90s, decentralized forms of payment began to appear, such as ecash [71] that offered a currency with a high level of privacy; It was then that the concept of "anonymous electronic money" began to emerge.

Wei Dai in 1998 published his proposal for electronic money called B-money [72] and from this idea, other proposals have emerged such as Bit gold [73], improving the implementation of a cryptocurrency using RPOW [74], an extension of the Hashcash work test system [75].

In 2009, the idea of a decentralized currency first emerged, when Satoshi Nakamoto published the first version of Bitcoin [76]. The purpose of this currency was to create a fully decentralized electronic payment system, using cryptographic tests instead of trust through a concept called proof of work (*POW*).

All monetary transactions of the system are stored in a data block until reaching a specific size. The idea of *POW* is to add a random number (called "nonce") to that data so that when performing a hash function of the whole block, it has a certain number of leading zeros. Once a block is created, it is used as reference to the next

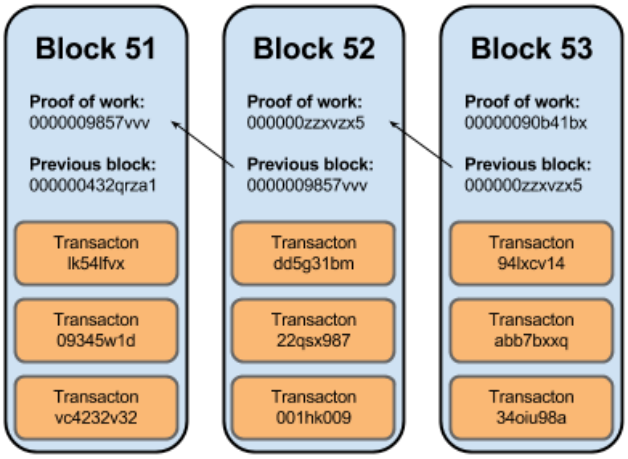

Figura 3.2: Blockchain representation diagram

block of data, adding the hash of the previous block to the next one. This system is called the blockchain [77] (see figure 3.2).

All the nodes compete to find the next block, because when one of them manages to find the nonce to create the hash with leading zeros, notifies it to all the nodes of the network and gains 12.5 bitcoins [78].

This system makes it practically impossible to falsify a transaction in the blockchain, since the minimum change would cause an totally different hash from the blockchains of the other nodes, provoking a desyncronization to the peer-to-peer network. However, this system is still vulnerable to attacks aimed at specific users, such as man in the middle attacks [79].

Ethereum uses this technology, not only for monetary exchanges, but for the execution of small fragments of code called “smart contracts”.

3.2.2.2. Smart Contracts

Ethereum was born inspired by the Bitcoin concept to offer any user a tool to develop decentralized and secure applications in a simple way [55]. These applications are called “smart contracts” and are written in the Ethereum’s blockchain.

Ethereum’s cryptocurrency is called “ETH” and not only works like Bitcoin, to exchange money between users, but to *fuel* the smart contracts’ execution, running its source code for a small amount of ETH.

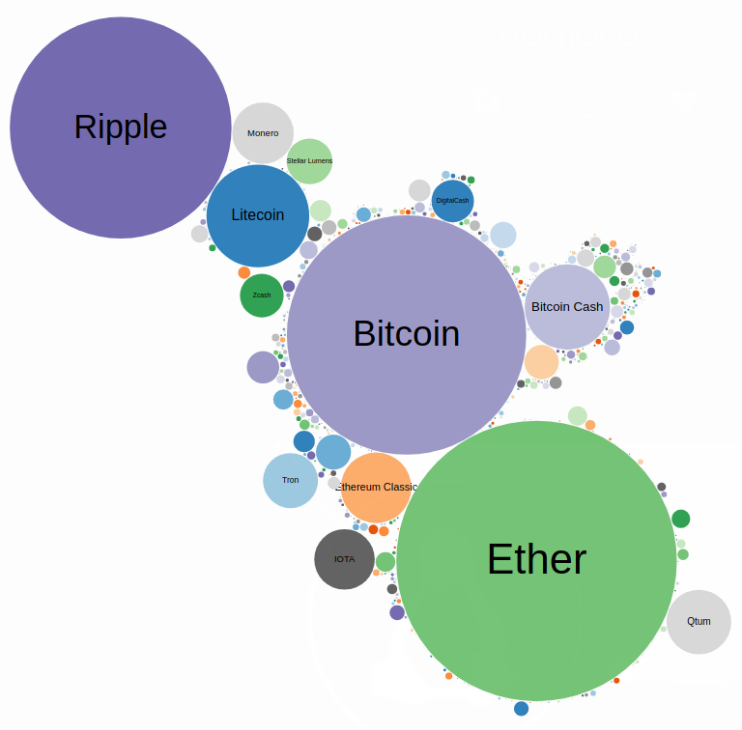

Figura 3.3: Existing cryptocurrencies represented by purchase-sale volume

Smart contracts are written in a programming language called Solidity[1] provided by the Ethereum's developers. This language is called contract-oriented and it was influenced by C++, Python and JavaScript. Solidity offers the possibility to create a wide range of decentralized applications in the blockchain in which users do not have to trust a centralized organization.

These contracts also have the capacity to store an transfer money, making them the perfect tool to implement a wide range of decentralized applications like: gambling games [80], voting systems [81], crowdfunding [82], prediction markets [83], transparency systems [84] and so on. Today it is the most exchanged currency within cryptocurrencies (see figure 3.3 [85]).

Smart contracts offer us a framework to design distributed platforms like the one proposed in this work, in which all transactions that interact with the scientific publication process can be cryptographically signed (see section **??**).

[1] http://solidity.readthedocs.io/en/develop/

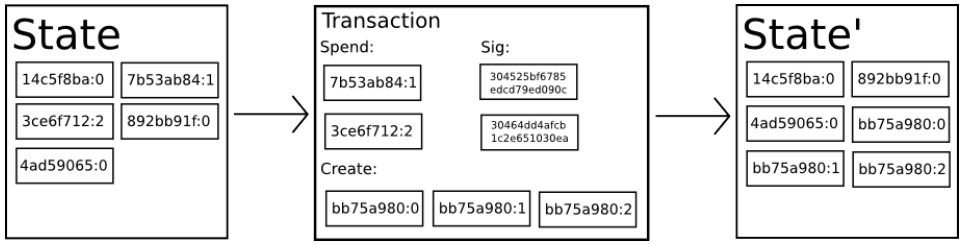

Figura 3.4: Blockchain's state change through transactions

3.2.2.3. Accounts and Transactions

An address in ethereum is a 32-byte string that symbolizes a person's account or an smart contract:

- **Personal accounts:** These are the accounts of the users who want to interact with the Ethereum network. Each one has its address and a balance.
- **Contract accounts:** A smart contract also is identified by an Ethereum address. It contains the source code, a balance with the available money, and its own internal memory where it saves all the contract's information.

In this way, users and contract behave very similar in the blockchain, and to communicate these addresses, Ethereum uses transactions.

Smart contracts behavior is transaction-based [86]. once a contract is deployed in the blockchain, users (or other contracts) may send transactions to run its code. Each transaction has a payload, containing the data required to execute the desired part of the contract. This execution has a fee that users have to pay to the network based on how complex is the code they want to run.

Ethereum's transaction-based smart contracts have changed the paradigm of modern software development, since the priority when developing a smart contract is to reduce the transaction costs of each interaction [87].

3.2.3. Remix

Remix[2] es un compilador online de contratos inteligentes diseñado por la comunidad de Ethereum que permite crear, compilar y desplegar contratos inteligentes tanto en una testnet como en la blockchain.

[2]https://remix.ethereum.org/

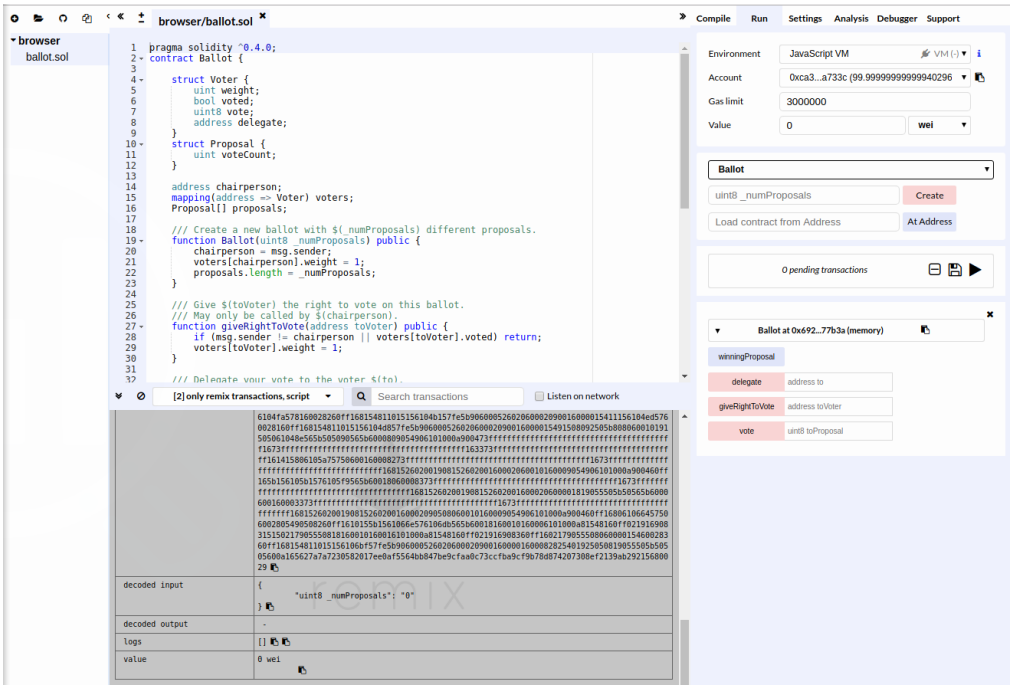

Figura 3.5: Remix smart contract web compiler

Este compilador ofrece la posibilidad de poder probar los contratos a través de una máquina virtual que simula la blockchain de Ethereum [88]. Tras desplegar un contrato a través de Remix, la plataforma genera elementos HTML que simulan botones de interacción con el código, sin tener que desarrollar un frontend para probar las funciones de este.

Además Remix ofrece un debugger para poder seguir el proceso de ejecución de una transacción, un estimador del gas necesario para poder realizarla, un analizador de código para detectar posibles vulnerabilidades y un sistema de exportación para generar ABIs (ver seccion 3.2.4).

Esta herramienta facilita mucho el proceso de desarrollo, ya que se pueden probar diferentes implementaciones realizando minimos cambios en el código y sin tener que instalar ningun programa (ver figura 3.5).

3.2.4. JavaScript and Metamask

Ethereum dispone de un sistema para conectar usuarios a la blockchain a través de los navegadores web. Existen varias formas de poder conectarse: mediante nave-

gadores especiales como AlethZero o Mist, o mediante la utilización de Metamask.

Metamask es una extensión de los navegadores web más comunes (como Firefox o Chrome) que permite al usuario conectarse tanto a la red de Ethereum como a una testnet personalizada. Contiene los mecanismos necesarios para poder realizar transacciones y comunicarse con los contratos inteligentes que están en la blockchain. Para poder utilzarlo simplemente hay que instalar un pequeño programa en el navegador y disponer de una dirección con fondos para poder realizar estas transacciones.

JavaScript se utiliza para conectar con la dirección de un contrato inteligente y poder llamar a funciones específicas de este. Para ello el JavaScript ha de disponer de dos datos importantes:

- **La dirección del contrato:** Al cargar la página web que se conecta al contrato se debe disponer de la dirección de Ethereum en el que está alojado, ya que para todas las interacciones con la plataforma se han de realizar transacciones a dicha dirección.
- **ABI del contrato:** ABI significa Aplication Biary Interface y es una estructura de datos en la que se encuentran todas las funciones a las que se puede llamar en un contrato inteligente. Es la forma que tiene javascript de poder construir las transacciones para llamar a dichas funciones. La generación del ABI suele ser automática cuando se crea un contrato inteligente.

Estas dos tecnologías son las utilizadas para realizar pruebas en el contrato inteligente a través de un HTML personalizado que simulaba la plataforma. Tniendo en cuenta como serían las interacciones de todos los usuarios de la plataforma si migraran de los sistemas actuales.

Chapter 4

Platform Description

The needs of the many outweigh the needs of the few

Spock - The Wrath of Khan

We propose a blockchain-enabled decentralized publication system for open science. It consists of three main components that decentralize and try to improve three different aspects related to scientific publication:

1) Peer review governance communication is traditionally centralized and controlled by editors and publishers. Our proposal opens and decentralizes these communications making the process more transparent.

2) Peer reviewer quality and reliability information is difficult to predict [8], and it is usually hold private by publishers and journals. The system proposes to open this information through a decentralized reputation network of peer reviewers over a blockchain.

3) Scientific papers are traditionally obtained or bought from a centralized publisher. We propose a decentralized network to distribute academic works and promote free access to science.

These ideas are further discussed in the following sections.

4.1. Transparent Peer Review Governance

The system provides a platform for the peer review process communication, from paper submission to paper acceptance or rejection. It registers all the interactions into a blockchain based distributed ledger.

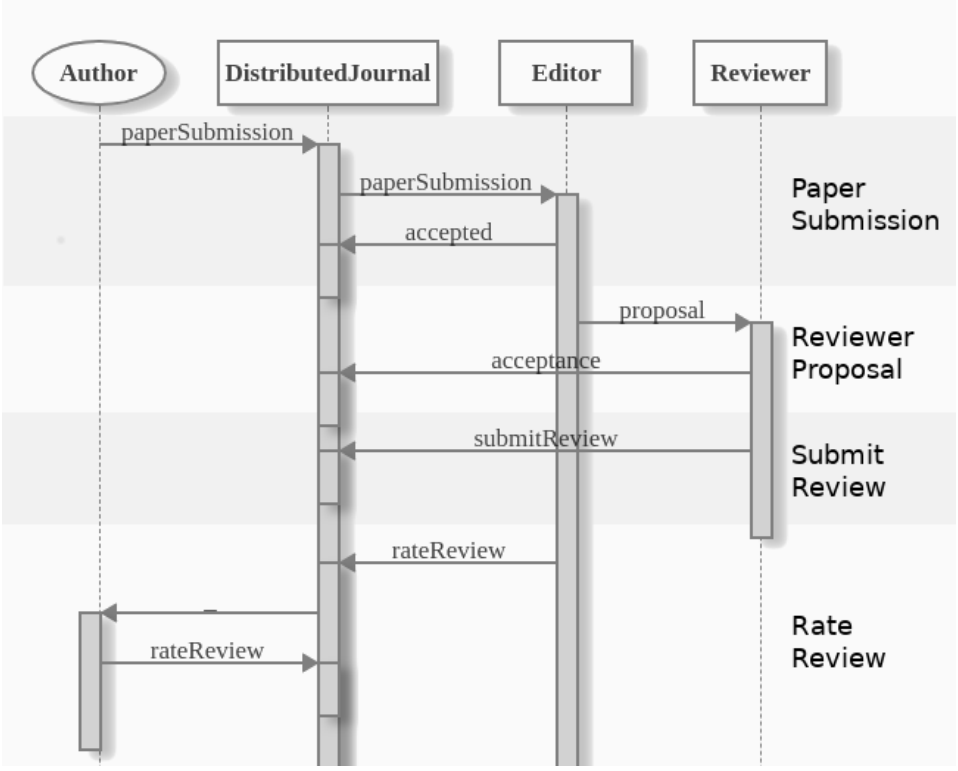

Figura 4.1: Sequence diagram of platform interaction

The interaction diagram of the system (Figure **??**) describes the interactions of the supported peer review governance. Following, this interactions and their implementation are described.

A paper submission is registered by submitting the IPFS address of the paper to an Ethereum contract. Then, the Ethereum sender address is recorded as the corresponding author, and the submission is timestamped in the blockchain.

A journal editor may invite a peer reviewer to review a specific paper. The transaction will record the Ethereum address of the reviewer and optionally, a deadline to submit the review.

An invited reviewer may accept or reject the review of a paper. The response will be recorded into the blockchain.

A reviewer should make a transaction to deliver the review. The transaction will record the acceptance/rejection and the IPFS address of the detailed review.

A novelty of the system further discussed in Section 4.2 is the rating system for

reviews. The transaction will record the sender address and the rating as well as the rated review and reviewer addresses.

4.2. The Peer Review Reputation network

The system proposes the use of a peer review reputation network were the quality of peer reviews is rated by the authors, editors and reviewers of the system. The work extends traditional peer review governance with the possibility of rating the reviews, building a reputation system for reviewers [7]. Reviewers get rewarded for worthy, fair, an timely reviews, or penalized otherwise.

This network of peer reviewers would enable a better reviewer selection, a fair recognition of reviewers work and a protection against unfair reviews for authors. However, it could also rise privacy concerns for both reviewers and raters [89; 90]. We consider these privacy issues in section 4.4.

4.3. Distributed Open Access infrastructure

Open Access focuses in the free access to scientific knowledge. While publishers provide free of charge their Open Access content, their control of the science dissemination infrastructure allows them to impose certain rules, such as charging authors unreasonable fees to offer their work as Open Access [91] (Gold Open Access) or the temporal embargo and restrictions on the dissemination of the final version (Green Open access) [92], among others.

The system proposes a decentralized infrastructure for science publication. Academic documents - from first drafts to final versions, including peer reviews- are shared in IPFS, an open P2P network [56]. Thus, the system inherently grants Open Access by the design of its distributed infrastructure and circumvents the publishers dominant role.

4.4. Privacy Settings of Open Peer Review and Rating

Anonymity of reviewers and authors in peer reviews is traditionally used to improve the fairness of the process. Thanks to single blind reviews, anonymous reviewers can honestly critic a paper without fearing the reactions of the authors. Double blind reviews also allow to reduce the impact of personal biases. Finally, open review

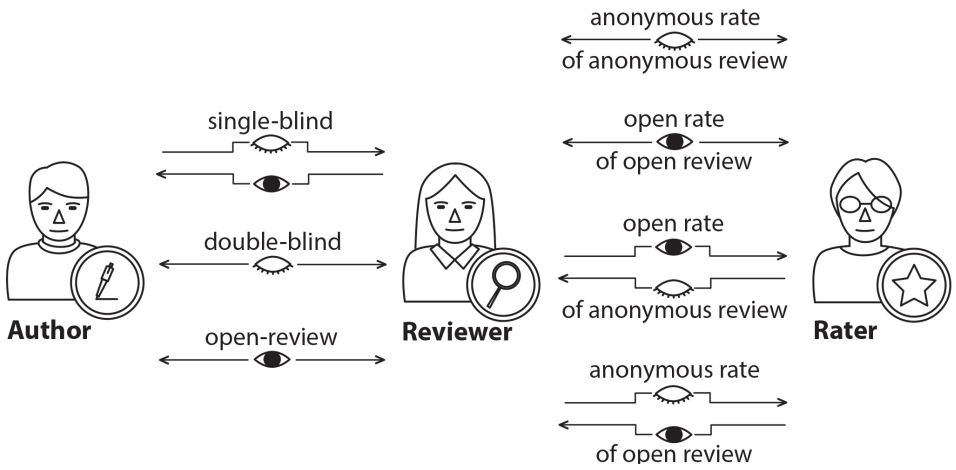

Figura 4.2: Review and Rating privacy models

models propose that both authors and reviewers know each other. These different privacy settings are shown in the left part of Figure 4.2.

Note, however, that the anonymity of the reviewers can be also abused. Unfair and low quality reviews are not discouraged by the system due to the lack of consequences. In order to alleviate this problem, our system proposes the construction of a reputation network of peer reviewers so that reviewers are awarded or criticized according to their work. This reputation network can also adopt different privacy settings, allowing both anonymous and signed ratings of either signed or anonymous reviews as depicted on the right side of Figure 4.2.

The implementation of these different privacy settings in a blockchain requires different approaches. The question of whether we can keep the benefits of blind review while providing accountability and recognition to reviewers deserves special consideration. Next, we discuss the different privacy models for review and privacy settings.

Blind peer review

Blind review is the protection of the identity of reviewers in the peer review process. In a blockchain, this protection could be easily achieved by using single-use addresses previously agreed with the editor.

Double blinded peer review

A double blinded review is a blind review that additionally protects the authors identity to prevent social bias [93] [94]. Authors could protect their identities prior to publication by providing a single-use public address on submission. Later they can reveal their real identity since they are the only ones with access to that address.

Open peer review

Open evaluation proposes the de-anonymization of all the parties involved in the peer review process [18]. While studies found effect on the percentage of reviewers declining to review [89] other implications remain open to debate [95].

Open Rating

Similarly to open reviews, open ratings are easy to implement by maintaining a public identity for the raters.

4.4.1. Anonymous Rating

Protecting the identity of raters is interesting in several reputation systems. We can support this anonymity feature using *blinded tokens* [90] that grant permission to rate without revealing the identity of the rater. People authorized to rate a review, such as authors, editors and other reviewers involved in the process, may each get one of these tokens.

Rating anonymous reviews

In a system that support voluntary signing of reviews, unsigned reviews would not affect the reviewers reputation unless they acknowledge their authorship. Thus, reviewers may only reveal their identities for well rated reviews, reducing the desired accountability for poor quality, unfair or late reviews.

A system allowing anonymous, yet accountable, reputation system for peer reviewing is therefore of great interest. Following, we discuss the feasibility of adopting different anonymity approaches to realize this system.

Collateral models are widely used in blockchain technology to ensure that an actor assumes negative consequences of an interaction in order to avoid the greater consequence of loosing the collateral. A similar strategy can be used for the anonymous

reputation network. If a reputation collateral is requested from the reviewers, they would be encouraged to claim even negative ratings. This model can be combined with anonymity measures to ensure accountable yet anonymous, peer reviews.

Coin mixing protocols are designed to obfuscate the relation between senders and receivers of Bitcoin payments by mixing in a single transaction many senders and receivers [96]. We can not directly apply this approach to rate reviews since the receiver identity is known. However it can be used in collaboration with other techniques discussed below.

Reusable payment codes enable the possibility of using a large amount of addresses to receive a payment [97; 98]. Reviewers may share one of this addresses with each of the actors with permission to rate and then collect the reputation probably using an anonymity layer such as coin mixing. Using a collateral model would encourage the acceptance of bad ratings.

ZK-Snarks are a cryptographic tool enabling to prove a statement without revealing anything else than the statement is in fact true (Zero-Knowledge Proof of Knowledge) [99; 100]. They also provide this property in a succinct and non-interactive fashion (i.e. using a relatively small proof and not requiring further communication between prover and verifier). Zcash uses this technology to build an anonymous cryptocurrency [101]. A similar approach could be used to manage anonymous ratings. A reviewer could also receive the rating of a review she did without revealing from which review or which rating the reputation comes. As before, a collateral model can be used to encourage the acceptance of low ratings.

4.5. Architecture

The platform's architecture consists in two main parts as explained in section 3.2, a decentralized file system in which users can upload all the files using IPFS (see section 3.2.1), and a smart contact to register all interactions of the users with the platform (see section 3.2.2.2).

As a decentralized technology, anyone can run a node locally, connecting to the IPFS network and to the Ethereum's blockchain to interact with the platform, but this technology is not used commonly, and not all users have the knowledge to install and run these programs. As a solution, a "gateway server" to test the platform's implementation was created, using an web browser extension called Metamask[1] to interact with the blockchain and running an IPFS node to upload the files as explained

[1] https://metamask.io/

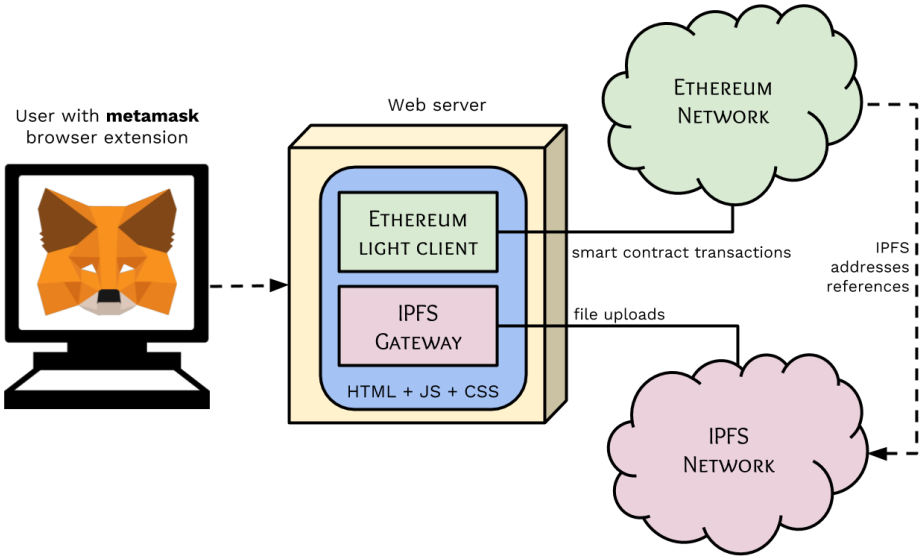

Figura 4.3: Architecture diagram of a node with IPFS and an Ethereum light client

in the diagram of the figure 4.3.

4.5.1. Smart Contract Architecture

Smart contracts in Ethereum can interact with each other, creating an ecosystem of programs that resemble object oriented programming. The contract structure and source code is crucial because once a contract is in the blockchain, there is no way to change it.

Uno de los retos más importantes para diseñar un contrato es reducir el coste de las transacciones, ya que el coste de estas puede ser muy alto si el diseño del contrato es ineficiente [102]. Insertar datos en la blockchain es muy caro, por eso es recomendable utilizar las estructuras de datos que nos ofrece Ethereum para reducir el coste de transacción. El problema se plante a cuando se utilizan direcciones en IPFS, las cuales están en BASE58, un tamaño que sólo es representable por el tipo "string" de Ethereum, el cual es de los tipos de datos maś costosos para trabajar.

Como solucion, la plataforma realiza la funcion hash en base32 de la dirección

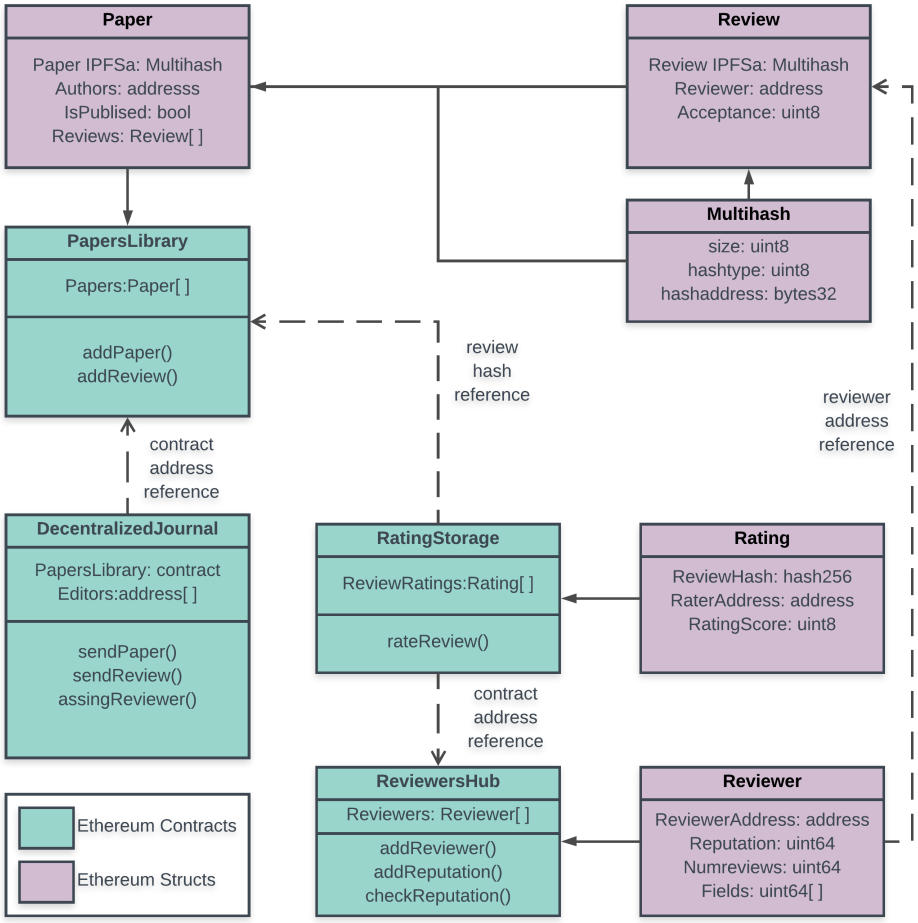

Figura 4.4: Diagrama UML del la estructura de contratos

IPFS, convirtiendo el tipo de dato a uno mucho más barato para reducir en un 30 % el coste de transacción.

La estructura dentro de Ethereum está separada en cuatro contratos inteligentes como se muestra en la figura 4.4.

Decentralized Journal:

Es el contrato desde el que se controlan todas las interacciones con los autores y los editores. Los autores interactuan con este contrato para enviar los papers, y los editores asignan a revisores para el proceso de revision por pares.

Este contrato tiene una serie de direcciones ethereum asociadas a los editores (que

son los que pueden asignar revisores) y una referencia a la dirección del contrato de la librería donde se envían los papers. Además a través de este contrato, los revisores envían las revisones que les han sido asignadas.

Decentralized Library

Este contrato es el que almacena las direcciones IPFS de los papers y controla si están aceptados por el journal o no.

Es el que se encarga de convertir una dirección de base58 a bytes32 para que sea manejada por la plataforma. Esta conversión se realiza en dos pasos: primero el navegador web a través de javaScrpt deshace la base58 obteniendo los datos de identificación de un archivo IPFS (ver seccion 3.2.1). Luego almacena estos datos en el struct "Multihash" y realiza una función hash para obtener un dato mucho más eficiente para trabajar en Ethereum. Este hash es el que se propagará por el sistema de contratos como idetificador del paper que se ha enviado.

Además cada paper almacena todas las revisiones que tiene a trasvés de un struct "Review". Este struct tiene una dirección IPFS con la review realizada, la dirección del reviewer y un entero representando si el paper es aceptado o no.

RatingStorage

Este contrato tiene como finalidad almacenar los ratings de las reviews que han hecho los reviewers, y se encarga de dar la reputacion a estos.

Cada ratrings es represetnado por un struct que tiene: un hash que identifica univocamente la review de un paper hecha por un revisor, la dirección de la persona que realiza el rating y un score que representa la reputacion que se le da al revisor.

Respecto al sistema de reputación se decidió adoptar un sistema de reputación de cinco estrellas explicado en la sección **??**

ReviewersHub

Se encarga de almacenar las direcciones, los campos de investigación, y la reputación de los revisores que etán registrados en la plataforma.

Con este contrato interactuan el contrato de Rating storage para dar las puntuaciones de reputación a los revisores, los nuevos revisores que quieran registrarse en la plataforma y los editores que quieran encontrar a un revisor para realizar el proceso de revisión por pares.

El diagrama 4.4 intenta ilustrar la estructura de los contratos a través de un UML, si asemejamos los contratos inteligentes de ethereum como objetos.

4.6. Reputation system

De los muchos sistemas de reputación comentados en la sección 2.2 se ha decidido para la plataforma, un basado en un rating de cinco estrellas [103]. Este sistema de reputación esta presente en muchas plataformas online como Tripadvisor[1], Amazon[2], Google Play[3] y otras grandes plataformas en las que los productos y servicios son votados por los usuarios.

Al estar implementado directamente en la blokchcain, todas las interacciones son totoalemtne públicas y auditables, con lo que cualquier usuario puede ver quien ha votado a qué revisión y con qué puntuacion, por lo que es un sistema en el que no existe de base el anonimato. Esto permite disuadir los problemas comentados en la sección 2.2 ya que los usuarios que realizan ataques dirigidfos o rating injustas se ven expuestos públicamente en la red.

El funcionamiento interno tiene varios sencillos pasos:

1. Al hacer una revisión, se crea en el sistema un hash (SHA-3 [104]) con: la dirección del revisor, la dirección del paper al que revisa y la dirección del journal que le asignó la revisión. Este hash identifica unívocamente a la revisión dentro del sistema.

2. Tanto los autores, como los editores y los otros revisores del paper tienen la capacidad de asignar una puntuación del 1 al 5 indicando si piensan que la review es justa o no.

3. Por cada votación, el sistema registra al votante y le envía la reputación al revisor, realizando un alisado exponencial de todas las votaciones que ha recibido [105].

A la hora de decidir si una review es justa o no hay varios puntos de vista [106; 107] y no todas las personas estarán de acuerdo, pero si se puede ofrecer unas guía de actuación para la comunidad para intentar que todas las votaciones del sistema sean lo más justas posibles para todos los que la utilicen.

[1] https://www.tripadvisor.com/
[2] https://www.amazon.com/
[3] https://play.google.com/

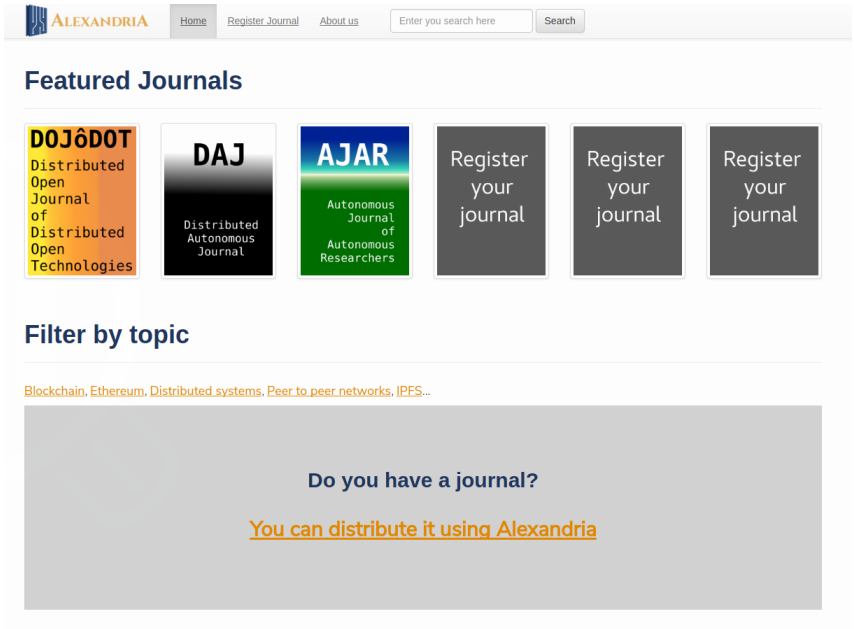

Figura 4.5: Diagrama UML del la estructura de contratos

Dentro de un sistema de reputación siempre existirá cierta controversia y siempre existirán métodos para poder aplacarla [108], pero el diseño se presenta en este trabajo es una prueba de concepto que irá evolucionando a medida que se diseñen nuevos sistemas para aplacar los problemas que surgen.

4.7. Proof of concept showcase

La prueba de concepto presentada bajo el nombre de “Alexandría” es una aplicación web que engloba todo el proceso de publicación academica actual. Esta demo se conecta al contrato descrito en la seccion **??**

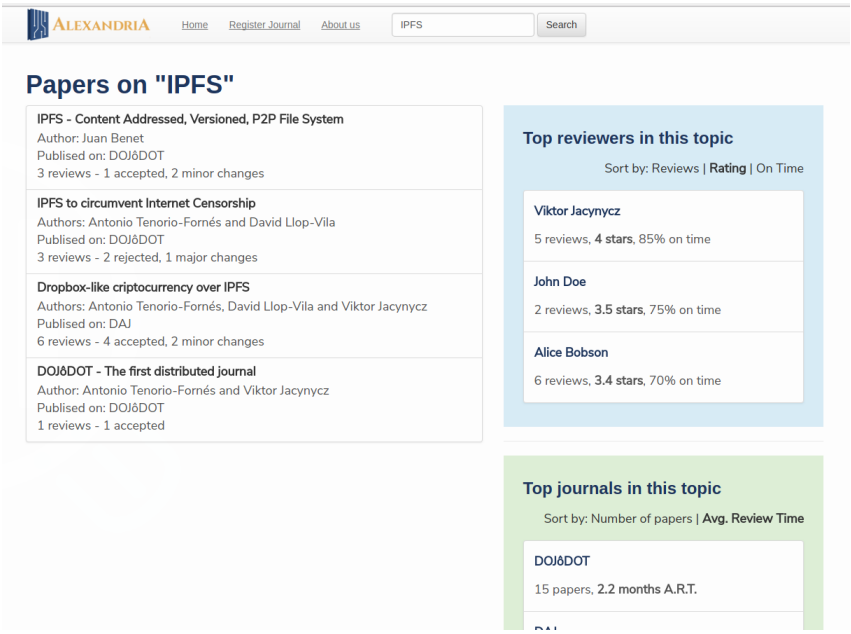

Figura 4.6: Diagrama UML del la estructura de contratos

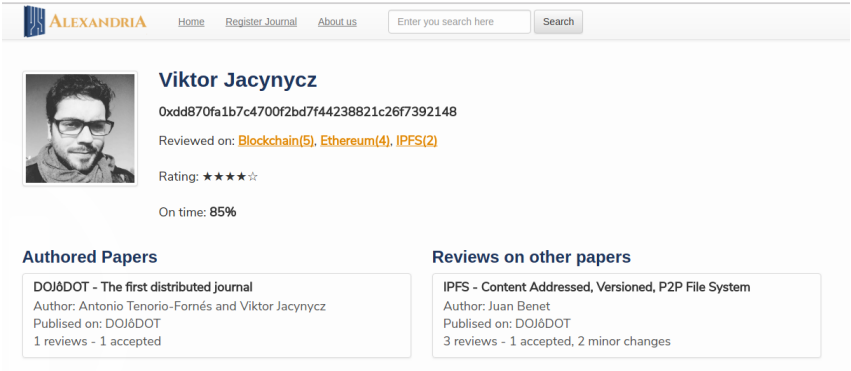

Figura 4.7: Diagrama UML del la estructura de contratos

Chapter 5

Discussion

The needs of the many outweigh the needs of the few

Spock - The Wrath of Khan

Como resultado del desarrollo de esta plataforma, el escenario ideal sería que algunos journals que ya hayan ido migrando al sistemas de publicación alternativos como los explicados en la sección 2.1 se adapten a esta plataforma. Al realizar un pequeño análisis del posible impacto de este trabajo hay dos puntos importantes a destacar.

5.1. Monetary Impact

El impacto monetario sería uno de los más notables tras la implantación de este sistema en sistemas de publicación científica de hoy en día. Según datos de investigaciones al respecto, el coste de publicación de un artículo en una revista de imacto varía de entre 1000$ hasta los 5000$ [16; 109], coste muchas veces inviable para investigadores que quieran avanzar en la investigación científica.

El coste de la publicación y el acceso a la ciencia a través de el trabajo propuesto sería únicamente variable en función del precio del ETH[1]. Tras realizar un análisis con varias versiones de la plataforma desplegada se determina que el coste de una transacción varía entre 100000 y 150000 de gas[2].

[1] La criptomoneda de Ethereum explicada en la sección 3.2.2

[2] Gas es lo que pagas como comisión a la red de Ethereum por ejecutar una transacción

Teniendo en cuenta que para que se publique un paper han de realizarse como mínimo 5 transacciones se puede determinar que el precio actual para publicar un paper ronda entre los 4$ y 6$ segun datos de Ethereum Gas Station[3], más de 250 veces más barato que los sistemas de publicación actual en el mejor de los casos.

5.2. Review Time and Quality Impact

Otro de los impactos importantes sería la reducción del tiempo y el aumento de la calidad en el proceso de revision por pares.

Si se dispone de una masa crítica de usuarios de la plataforma propuesta, se crea un ecosistema de usuarios que alimentan tanto la red de reputación de revisores como los contratos de publicación científica (ver section **??**). El proceso de Peer review se vería afectado de dos maneras:

1. **El tiempo de revisión:** Los contratos inteligentes permiten establecer tiempos límites para la revisión de un artículo, suponiendo una penalización a los revisores que no cumplan estos plazos (ver sección 4.2). Si un Decentralized Journal tiene unos tiempos de revisión establecidos, y los revisores que asignan aceptan las revisiones, probablemente se experimente una mejoría en el tiempo de entrega de las revisiones y por lo tanto en el proceso de publicación, con respecto a los tiempos muchas veces excesivos de los sistemas de publicacion actual [5].

2. **La calidad de las revisones:** Todas las revisiones son rateables por la comunidad y afectan directamente a la reputación del revisor, así que es altamente probable que la calidad de las revisiones en el sistema sufra una mejoría y desaparezcan muchos de los problemas respecto a la revisión por pares comentados en la sección 1.

5.3. Science Distribution

Toda interacción con la plataforma ha de realizarse mediante una cuenta Ethereum (ver section 3.2.2) y quedan grabadas en la blockchain de este. Esto implicaría que a través de la dirección de un investigador científico se puedan obtener dato de todos los papers que ha publicado y revisado. Todo la comunidad científica podría sufrir una mejor gracias a este sistema. Además, los nuevos investigadores que

[3]Precio de una transacción en Ethereum https://ethgasstation.info/

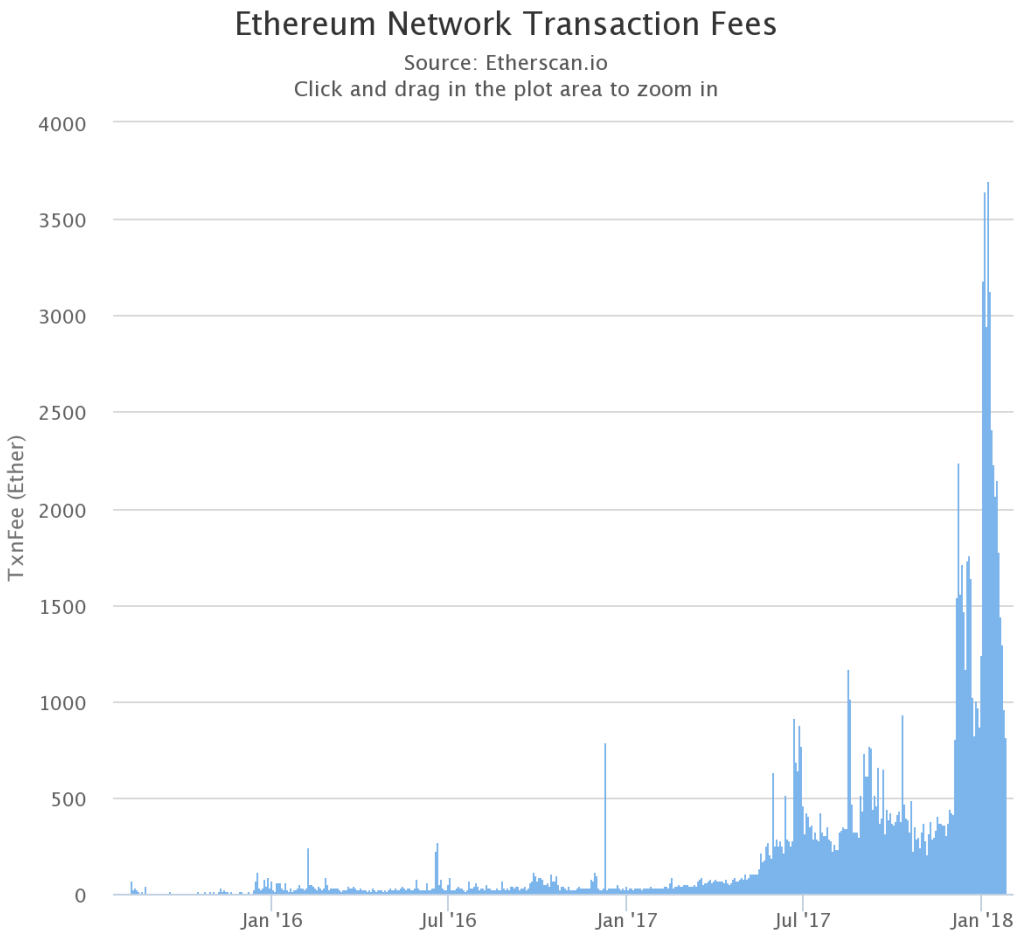

Figura 5.1: Ethereum transaction fees evolution

quieren empezar su carrera en el mundo académico pueden obtener visibilidad si los revisores que revisan los papers que envían a los Distributed Journals tienen alta reputación o no.

Si el sistema se implantara de manera exitosa, la comunidad científica empezaría a cuestionarse la existencia de los publishers, y si estos desaparecieran, se encontrarían nuevas formas de financiación de proyectos.

5.4. Problems

Hay varios problemas para implantar este proyecto hoy en día, ya que las tecnologías propuestas todavía tienen poda expansión y son poco conocidas por el usuario medio.

La primera es el cambio de plataforma para los investigadores de la comunidad, ya que la costumbre de utilizar las plataformas de hoy en día es dificil de cambiar, por lo que proponer un cambio en los sistemas de comunicación y revisión puede suponer un gran rechazo inicial, pudiendo llevar al proyecto a un punto muerto, sobretodo si la metodología de conexión a la blockchain de Ethereum es compleja actualmente.

La segunda es el precio de las transacciones. Ethereum es una moneda que fluctúa bastante, y ultimamente se ha experimentado una gran crecida de todas los precios de las criptomonedas con respecto a hace seis meses. Las subidas en el precio provocan subidas en las transacciones, que a su vez provoca que la interacción sea más cara para todos los usuarios que la utilizan.

Chapter 6

Conclusions and future work

The needs of the many outweigh the needs of the few

Spock - The Wrath of Khan

Decentralizing and opening academic publishing and peer review infrastructure enable many possibilities to foster the Open Access and Open Evaluation vision of a free and fair science publication. The paper has introduced the decentralization of three essential functions of science dissemination: 1) the peer review process, 2) the selection and recognition of peer reviewers, and 3) the distribution of scientific knowledge.

The transparency provided by opening the peer review allows the construction of a reputation system of reviewers, but also rise concerns about privacy and fairness. The paper studies the different privacy settings in the peer review process and the reputation network.

Blockchain technology enables the introduction of a new review model that support the accountability of peer reviewing proposed by open peer review models while keeping the anonymity of blind and double blind reviews to improve fairness.

This challenging proposal raise many issues. The implications of such accountable, open and anonymous review models are still to be revealed. Moreover, a deeper study of the strategies to grant anonymity to different actors is also to be done. Nevertheless, we believe the proposal opens a debate worth having.

This paper proposes the opening and decentralization of three of the peer review and publication functions: 1) the peer review process communication, 2) the reputation of reviewers, and 3) the distribution of papers and peer reviews. Arguably,

this decentralization of the infrastructure could help to challenge the central role of middlemen such as traditional publishers.

Distributed technologies such as Blockchain and IPFS may finally realize the promise of Open Access, while enabling new not-for-profit models of science dissemination. Opening and decentralizing the infrastructure enhances the transparency and accountability of the system, and fosters innovation.

Note the proposed system does not rely on the use of cryptocurrencies, since it is focused on a not-for-profit approach, far from the startup-driven commercial approaches common in the blockchain space.

This challenging proposal raises multiple issues. The opening of the peer review process may reduce the privacy of current closed system. Blind review relies in such privacy, and a lack of this protection can cause a great rejection by the community. Recent technical cryptographic innovations may be used to circumvent this issue [99] and allow transparency while still allowing double blind reviews.

The introduction of a new public metric (reviewers' reputation) may also affect researcher careers, adding pressure to the already straining processes for academic survival [110].

Additionally, the proposed system's infrastructure relies in new technologies with their own challenges. Blockchain technologies face scalability, transaction costs, inclusiveness and usability problems that remain open and under discussion. On the other hand, distributed file systems such as IPFS may be more resilient, but they still need somebody in charge of preserving and providing the data, since without that responsible actor, it may result in unpredictable loss of content.

Other open issues that may be explored in future work are the exploration of different copyright regimes, the challenging of traditional journal-centered metrics to rate publication quality, different reputation algorithms, different levels of openness, and the exploration of decentralized autonomous journals.

Despite the existing challenges, we are confident that decentralizing the processes that Science relies on, would open up a whole new playing field, with implications we cannot possibly foresee now. Will its benefits outweigh its risks? We believe it is a conversation worth having.

6.1. Future Work

Uno de los primeros problemas a atacar sería el anonimato en la revisiones por pares, ya que por muy utópico que parezca, un sistema totalmente público en el que

no exista el anoniumato de los revisores y los autores es bastante dificil de implantar. Este problema se podría paliar implementando algína de las soliciones propuestas en la sección **??**, incluso pudiendo crear una cadena de bloques interna para todas las universidades del mundo en la que ya estén implementados protocolos de anonimato.

A raiz de este trabajo se extablecen unas nuevas lineas de desarrollo para un proyecto mucho más grande y con un alto impacto. Pero para poder realizarlo hay que tener en cuenta los siguientes hitos.

La incorporación de un sistema para poder reenviar papers ya aceptados, intentando ampliar la investigación que proponen, incluso poder seguir las líneas de investigación de otros autores completando sus papers para construir una comunidad científica colaborativa y basada en la ayuda mutua.

La automatización de la elección de los revisores que ha de tener un artículo recién enviado hará que el papel formal del editor desaparezca, ya que la elección dependerá enteramente del contrato inteligente del journal al que corresponde. Esta elección se basa en la confianza que tiene la comunidad en la red de reputation de revisores, pudiendo eliminar uno de los intermediarios que del sistema.

Luego habría que eliminar el intermediario de los journals, ya que si la elección de revisores y publicación de los artículos es automática y está en la blokchcain, la existencia de los journal sería cuestionada por la comunidad científica, pudiendo convertir la diseminación de ciencia en una gran biblioteca en la que los autores publican sus papers y son revisados por revisores aleatorios que elige el sistema.

Esta úlitma idea abre la posibilidad de poder crear colecciones determinadas dentro de la biblioteca de papers que funcionen como los actuales “issues”.

Bibliografía

Y así, del mucho sleer y del poco dormir, se le secó el celebro de manera que vino a perder el juicio.

Miguel de Cervantes Saavedra

[1] G. E, "The history and meaning of the journal impact factor," *JAMA*, vol. 295, no. 1, pp. 90–93, 2006. [Online]. Available: +http://dx.doi.org/10.1001/jama.295.1.90

[2] E. Garfield, "The evolution of the science citation index," *International microbiology*, vol. 10, no. 1, p. 65, 2007.

[3] E. G. Toledo and D. T. Salinas, "Book citation index: nueva historia sobre big science y little science," *Anuario ThinkEPI*, no. 1, pp. 195–202, 2011.

[4] C. Wenneras and A. Wold, "Nepotism and sexism in peer-review," *Women, sience and technology: A reader in feminist science studies*, pp. 46–52, 2001.

[5] J. Huisman and J. Smits, "Duration and quality of the peer review process: the author's perspective," *Scientometrics*, pp. 1–18, 2017.

[6] C. T. Bergstrom and T. C. Bergstrom, "The costs and benefits of library site licenses to academic journals," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 3, pp. 897–902, 2004.

[7] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation systems," *Communications of the ACM*, vol. 43, no. 12, pp. 45–48, 2000.

[8] M. L. Callaham and J. Tercier, "The relationship of previous training and experience of journal peer reviewers to subsequent review quality," *PLoS medicine*, vol. 4, no. 1, p. e40, 2007.

[9] R. Spier, "The history of the peer-review process," *TRENDS in Biotechnology*, vol. 20, no. 8, pp. 357–358, 2002.

[10] S. Goldbeck-Wood, "Evidence on peer review—scientific quality control or smokescreen?" *BMJ: British Medical Journal*, vol. 318, no. 7175, p. 44, 1999.

[11] G. Eysenbach, "Citation advantage of open access articles," *PLoS biology*, vol. 4, no. 5, p. e157, 2006.

[12] R. Walker and P. Rocha da Silva, "Emerging trends in peer review—a survey," *Frontiers in Neuroscience*, vol. 9, May 2015.

[13] B. Whitworth and R. Friedman, "Reinventing academic publishing online. part i: Rigor, relevance and practice," *First Monday*, vol. 14, no. 8, 2009. [Online]. Available: http://firstmonday.org/ojs/index.php/fm/article/view/2609

[14] J. A. Evans and J. Reimer, "Open access and global participation in science," *Science*, vol. 323, no. 5917, pp. 1025–1025, 2009.

[15] V. Larivière, S. Haustein, and P. Mongeon, "The oligopoly of academic publishers in the digital era," *PloS one*, vol. 10, no. 6, p. e0127502, 2015.

[16] R. Van Noorden *et al.*, "The true cost of science publishing," *Nature*, vol. 495, no. 7442, pp. 426–429, 2013.

[17] M. Ware, "Peer review: benefits, perceptions and alternatives," *Publishing Research Consortium*, vol. 4, pp. 1–20, 2008.

[18] E. Ford, "Defining and characterizing open peer review: A review of the literature," *Journal of Scholarly Publishing*, vol. 44, no. 4, pp. 311–326, 2013.

[19] P. Frishauf, "Reputation systems: a new vision for publishing and peer review," *Journal of Participatory Medicine*, 2009.

[20] J. Willinsky, "Open journal systems: An example of open source software for journal management and publishing," *Library hi tech*, vol. 23, no. 4, pp. 504–519, 2005.

[21] P. Binfield, "Open access megajournals: have they changed everything?" 2013.

[22] R. Wellen, "Open access, megajournals, and moocs: on the political economy of academic unbundling," *SAGE Open*, vol. 3, no. 4, p. 2158244013507271, 2013.

[23] B.-C. Björk, "Have the "mega-journals" reached the limits to growth?" *PeerJ*, vol. 3, p. e981, 2015.

[24] S. Anderton and L. Harvey, "Continuous publication: ready, set, cite!" 2013.

[25] R. Hayman, "View the continuous publication model as a satisfactory alternative for open access lis journals. evid based libr inf pract. 2014; 9."

[26] S. Harnad, "Electronic preprints and postprints," in *Encyclopedia of library and information science*. Marcel Dekker, 2003, vol. 67, no. 4.

[27] C. Brown, "The e-volution of preprints in the scholarly communication of physicists and astronomers," *Journal of the Association for Information Science and Technology*, vol. 52, no. 3, pp. 187–200, 2001.

[28] X. Shuai, A. Pepe, and J. Bollen, "How the scientific community reacts to newly submitted preprints: Article downloads, twitter mentions, and citations," *PloS one*, vol. 7, no. 11, p. e47523, 2012.

[29] K. Sugiyama and M.-Y. Kan, "Scholarly paper recommendation via user's recent research interests," in *Proceedings of the 10th annual joint conference on Digital libraries*. ACM, 2010, pp. 29–38.

[30] S. Bartling and B. Fecher, "Blockchain for science and knowledge creation. zenodo," *Publisher Full Text*, 2016.

[31] J. P. Tennant, J. M. Dugan, D. Graziotin, D. C. Jacques, F. Waldner, D. Mietchen, Y. Elkhatib, L. B. Collister, C. K. Pikas, T. Crick *et al.*, "A multidisciplinary perspective on emergent and future innovations in peer review," *F1000Research*, vol. 6, 2017.

[32] A. C. Kade Morton, "Aletheia: blockchain for scientific knowledge with a community management framework," 2017. [Online]. Available: https://github.com/aletheia-foundation/aletheia-whitepaper/

[33] V. Dhillon, "From bench to bedside: Enabling reproducible commercial science via blockchain," *Bitcoin Magazine*, 2016.

[34] A. Josang and R. Ismail, "The beta reputation system," in *Proceedings of the 15th bled electronic commerce conference*, vol. 5, 2002, pp. 2502–2511.

[35] P. Resnick and R. Zeckhauser, "Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system," in *The Economics of the Internet and E-commerce.* Emerald Group Publishing Limited, 2002, pp. 127–157.

[36] R. Zhou and K. Hwang, "Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing," *IEEE Transactions on parallel and distributed systems*, vol. 18, no. 4, 2007.

[37] F. Dotzer, L. Fischer, and P. Magiera, "Vars: A vehicle ad-hoc network reputation system," in *World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005. Sixth IEEE International Symposium on a.* IEEE, 2005, pp. 454–456.

[38] G. Moore and L. G. Meredith, "Reputation system for web services," Dec. 16 2008, uS Patent 7,467,206.

[39] B. T. Adler and L. De Alfaro, "A content-driven reputation system for the wikipedia," in *Proceedings of the 16th international conference on World Wide Web.* ACM, 2007, pp. 261–270.

[40] M. Sharples and J. Domingue, "The blockchain and kudos: A distributed system for educational record, reputation and reward," in *European Conference on Technology Enhanced Learning.* Springer, 2016, pp. 490–496.

[41] K. Hoffman, D. Zage, and C. Nita-Rotaru, "A survey of attack and defense techniques for reputation systems," *ACM Computing Surveys (CSUR)*, vol. 42, no. 1, p. 1, 2009.

[42] A. Whitby, A. Jøsang, and J. Indulska, "Filtering out unfair ratings in bayesian reputation systems," in *Proc. 7th Int. Workshop on Trust in Agent Societies*, vol. 6, 2004, pp. 106–117.

[43] M. Fuster Morell, "Governance of online creation communities: Provision of infrastructure for the building of digital commons," Ph.D. dissertation, European University Institute, 2010.

[44] A. F. Osborn, "Applied imagination." 1953.

[45] R. I. Sutton and A. Hargadon, "Brainstorming groups in context: Effectiveness in a product design firm," *Administrative Science Quarterly*, pp. 685–718, 1996.

[46] B. Mullen, C. Johnson, and E. Salas, "Productivity loss in brainstorming groups: A meta-analytic integration," *Basic and applied social psychology*, vol. 12, no. 1, pp. 3–23, 1991.

[47] J. Pokorná, L. Pilar, T. Balcarová, and I. Sergeeva, "Value proposition canvas: Identification of pains, gains and customer jobs at farmers' markets," *AGRIS on-line Papers in Economics and Informatics*, vol. 7, no. 4, p. 123, 2015.

[48] L. O. Meertens, M. E. Iacob, L. J. M. Nieuwenhuis, M. J. van Sinderen, H. Jonkers, and D. Quartel, "Mapping the business model canvas to archimate," in *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, ser. SAC '12. New York, NY, USA: ACM, 2012, pp. 1694–1701. [Online]. Available: http://doi.acm.org/10.1145/2245276.2232049

[49] A. Osterwalder, Y. Pigneur, G. Bernarda, and A. Smith, *Value proposition design: How to create products and services customers want.* John Wiley & Sons, 2014.

[50] B. Boehm and R. Turner, "Management challenges to implementing agile processes in traditional development organizations," *IEEE software*, vol. 22, no. 5, pp. 30–39, 2005.

[51] J. A. Livermore, "Factors that significantly impact the implementation of an agile software development methodology," *Journal of software*, vol. 3, no. 4, pp. 31–36, 2008.

[52] L. Lindstrom and R. Jeffries, "Extreme programming and agile software development methodologies," *Information systems management*, vol. 21, no. 3, pp. 41–52, 2004.

[53] W. Theunissen, A. Boake, and D. G. Kourie, "In search of the sweet spot: agile open collaborative corporate software development," in *Proceedings of the 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries.* South African Institute for Computer Scientists and Information Technologists, 2005, pp. 268–277.

[54] L. Rising and N. S. Janoff, "The scrum software development process for small teams," *IEEE software*, vol. 17, no. 4, pp. 26–32, 2000.

[55] V. Buterin, "Ethereum: A next-generation smart contract and decentralized application platform," 2014.

[56] J. Benet, "IPFS-content addressed, versioned, P2p file system," *arXiv preprint arXiv:1407.3561*, 2014. [Online]. Available: https://arxiv.org/abs/1407.3561

[57] ——, "Ipfs-content addressed, versioned, p2p file system," *arXiv preprint arXiv:1407.3561*, 2014.

[58] A. Kaluszka, "Distributed hash tables," 2010.

[59] B. Y. Zhao, L. Huang, J. Stribling, S. C. Rhea, A. D. Joseph, and J. D. Kubiatowicz, "Tapestry: A resilient global-scale overlay for service deployment," *IEEE Journal on selected areas in communications*, vol. 22, no. 1, pp. 41–53, 2004.

[60] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 4, pp. 149–160, 2001.

[61] I. Gupta, K. Birman, P. Linga, A. Demers, and R. Van Renesse, "Kelips: Building an efficient and stable p2p dht through increased memory and background overhead," *Peer-to-Peer Systems II*, pp. 160–169, 2003.

[62] P. Maymounkov and D. Mazieres, "Kademlia: A peer-to-peer information system based on the xor metric," in *International Workshop on Peer-to-Peer Systems*. Springer, 2002, pp. 53–65.

[63] J. Li, J. Stribling, T. M. Gil, R. Morris, and M. F. Kaashoek, "Comparing the performance of distributed hash tables under churn." in *Iptps*, vol. 4. Springer, 2004, pp. 87–99.

[64] B. Cohen, "Incentives build robustness in bittorrent," in *Workshop on Economics of Peer-to-Peer systems*, vol. 6, 2003, pp. 68–72.

[65] J. Pouwelse, P. Garbacki, D. Epema, and H. Sips, "The bittorrent p2p file-sharing system: Measurements and analysis," in *IPTPS*, vol. 5. Springer, 2005, pp. 205–216.

[66] D. Levin, K. LaCurts, N. Spring, and B. Bhattacharjee, "Bittorrent is an auction: analyzing and improving bittorrent's incentives," in *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 4. ACM, 2008, pp. 243–254.

[67] L. Torvalds and J. Hamano, "Git: Fast version control system," *URL http://git-scm. com*, 2010.

[68] D. Spinellis, "Version control systems," *IEEE Software*, vol. 22, no. 5, pp. 108–109, 2005.

[69] D. Bleichenbacher and U. M. Maurer, "Directed acyclic graphs, one-way functions and digital signatures," in *Annual International Cryptology Conference*. Springer, 1994, pp. 75–82.

[70] D. D. F. Mazières, "Self-certifying file system," Ph.D. dissertation, Massachusetts Institute of Technology, 2000.

[71] Digicash - an introduction to ecash. [Online]. Available: https://web.archive.org/web/19971009044558/http://digicash.com/publish/ecash_intro/ecash_intro.html

[72] W. Dai. B-money proposal. [Online]. Available: http://www.weidai.com/bmoney.txt

[73] N. Szabo. Unenumerated: Bit gold. [Online]. Available: http://unenumerated.blogspot.com.es/2005/12/bit-gold.html

[74] H. Finney, "Rpow: Reusable proofs of work," *CodeCon 2005*, 2005.

[75] A. Back *et al.*, "Hashcash-a denial of service counter-measure," 2002.

[76] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[77] A. M. Antonopoulos, *Mastering Bitcoin: unlocking digital cryptocurrencies*. .º'Reilly Media, Inc.", 2014.

[78] S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better—how to make bitcoin a better currency," in *International Conference on Financial Cryptography and Data Security*. Springer, 2012, pp. 399–414.

[79] T. Moore and N. Christin, "Beware the middleman: Empirical analysis of bitcoin-exchange risk," in *International Conference on Financial Cryptography and Data Security*. Springer, 2013, pp. 25–33.

[80] P. J. Piasecki, "Gaming self-contained provably fair smart contract casinos," *Ledger*, vol. 1, pp. 99–110, 2016.

[81] P. McCorry, S. F. Shahandashti, and F. Hao, "A smart contract for boardroom voting with maximum voter privacy," in *International Conference on Financial Cryptography and Data Security*. Springer, 2017, pp. 357–375.

[82] V. Jacynycz, A. Calvo, S. Hassan, and A. A. Sánchez-Ruiz, "Betfunding: A distributed bounty-based crowdfunding platform over ethereum," in *Distributed Computing and Artificial Intelligence, 13th International Conference*. Springer, 2016, pp. 403–411.

[83] J. Peterson and J. Krug, "Augur: a decentralized, open-source platform for prediction markets," *arXiv preprint arXiv:1501.01042*, 2015.

[84] J. Bonneau, "Ethiks: Using ethereum to audit a coniks key transparency log," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 95–105.

[85] C. Eisenberg. Graphical comparison of all cryptocurrencies. [Online]. Available: https://www.cryptocoincharts.info/coins/graphicalComparison

[86] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, pp. 1–32, 2014.

[87] K. Delmolino, M. Arnett, A. Kosba, A. Miller, and E. Shi, "Step by step towards creating a safe smart contract: Lessons and insights from a cryptocurrency lab," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 79–94.

[88] E. Hildenbrandt, M. Saxena, X. Zhu, N. Rodrigues, P. Daian, D. Guth, and G. Rosu, "Kevm: A complete semantics of the ethereum virtual machine," Tech. Rep., 2017.

[89] S. Van Rooyen, F. Godlee, S. Evans, N. Black, and R. Smith, "Effect of open peer review on quality of reviews and on reviewers' recommendations: a randomised trial," *Bmj*, vol. 318, no. 7175, pp. 23–27, 1999.

[90] A. Schaub, R. Bazin, O. Hasan, and L. Brunie, "A trustless privacy-preserving reputation system," in *IFIP International Information Security and Privacy Conference*. Springer, 2016, pp. 398–411.

[91] D. J. Solomon and B.-C. Björk, "A study of open access journals using article processing charges," *Journal of the Association for Information Science and Technology*, vol. 63, no. 8, pp. 1485–1495, 2012.

[92] B.-C. Björk, M. Laakso, P. Welling, and P. Paetau, "Anatomy of green open access," *Journal of the Association for Information Science and Technology*, vol. 65, no. 2, pp. 237–250, 2014.

[93] C. J. Lee, C. R. Sugimoto, G. Zhang, and B. Cronin, "Bias in peer review," *Journal of the Association for Information Science and Technology*, vol. 64, no. 1, pp. 2–17, 2013.

[94] A. E. Budden, T. Tregenza, L. W. Aarssen, J. Koricheva, R. Leimu, and C. J. Lortie, "Double-blind review favours increased representation of female authors," *Trends in ecology & evolution*, vol. 23, no. 1, pp. 4–6, 2008.

[95] T. Groves and K. Khan, "Is open peer review the fairest system," *BMJ (Clinical research ed.)*, vol. 341, pp. 1082–1083, 2010.

[96] S. Meiklejohn and C. Orlandi, "Privacy-enhancing overlays in bitcoin," in *International Conference on Financial Cryptography and Data Security*. Springer, 2015, pp. 127–141.

[97] M. Harrigan and C. Fretter, "The unreasonable effectiveness of address clustering," in *Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), 2016 Intl IEEE Conferences*. IEEE, 2016, pp. 368–373.

[98] J. Ranvier, "Reusable payment codes for hierarchical deterministic wallets." [Online]. Available: https://github.com/bitcoin/bips/pull/159

[99] M. Blum, P. Feldman, and S. Micali, "Non-interactive zero-knowledge and its applications," in *Proceedings of the twentieth annual ACM symposium on Theory of computing*. ACM, 1988, pp. 103–112.

[100] N. Bitansky, A. Chiesa, Y. Ishai, O. Paneth, and R. Ostrovsky, "Succinct non-interactive arguments via linear interactive proofs," in *Theory of Cryptography*. Springer, 2013, pp. 315–333.

[101] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *Security and Privacy (SP), 2014 IEEE Symposium on*. IEEE, 2014, pp. 459–474.

[102] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer *et al.*, "On scaling decentralized blockchains," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 106–125.

[103] M. Kinateder and K. Rothermel, "Architecture and algorithms for a distributed reputation system," in *International Conference on Trust Management*. Springer, 2003, pp. 1–16.

[104] J.-P. Aumasson, L. Henzen, W. Meier, and R. C.-W. Phan, "Sha-3 proposal blake," *Submission to NIST*, 2008.

[105] E. S. Gardner, "Exponential smoothing: The state of the art," *Journal of forecasting*, vol. 4, no. 1, pp. 1–28, 1985.

[106] H.-D. Daniel and W. E. Russey, *Guardians of science: Fairness and reliability of peer review*. Wiley Online Library, 1993.

[107] J. R. Cole, "Fair science: Women in the scientific community," 1979.

[108] C. Dellarocas, "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior," in *Proceedings of the 2nd ACM conference on Electronic commerce*. ACM, 2000, pp. 150–157.

[109] R. Russel, "The business of academic publishing: A strategic analysis of the academic journal publishing industry and its impact on the future of scholarly publishing," *Electron J Acad Spec Librarianship. http://southernlibrarianship. icaap. org/content*, 2008.

[110] M. De Rond and A. N. Miller, "Publish or perish: bane or boon of academic life?" *Journal of Management Inquiry*, vol. 14, no. 4, pp. 321–329, 2005.

–¿Qué te parece desto, Sancho? – Dijo Don Quijote –
Bien podrán los encantadores quitarme la ventura,
pero el esfuerzo y el ánimo, será imposible.

Segunda parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes

–Buena está – dijo Sancho –; fírmela vuestra merced.
–No es menester firmarla – dijo Don Quijote–,
sino solamente poner mi rúbrica.

Primera parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes