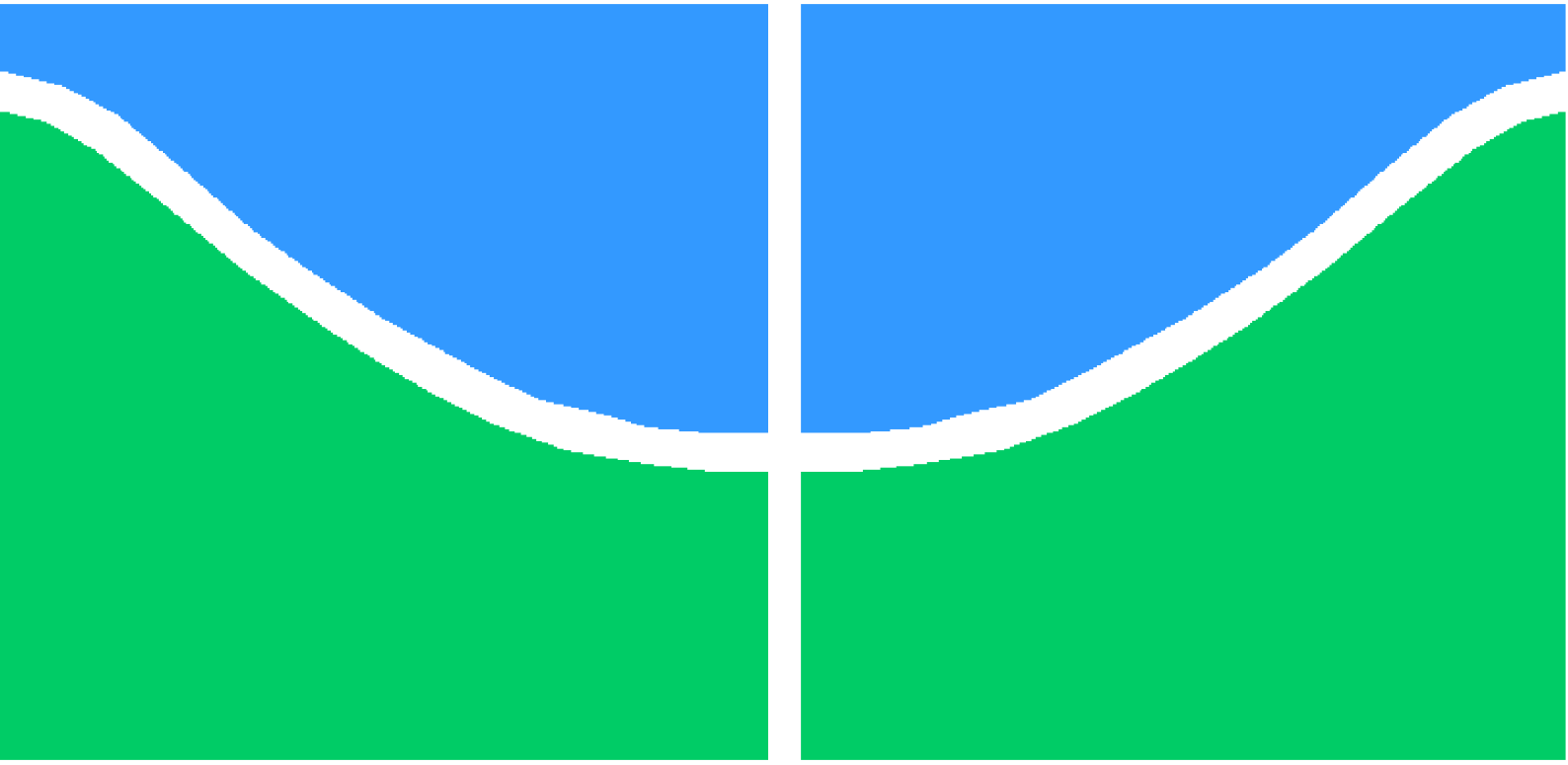

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Engenharia de Software

Blockchain - Tecnologia, Arquitetura e Aplicações

Autor: Kleber Brito Moreira

Brasília, DF

05/12/2019

Autor: Kleber Brito Moreira

Blockchain - Tecnologia, Arquitetura e Aplicações

Projeto submetido ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Prof. Dr. Ricardo Matos Chaim

Brasília, DF

05/12/2019

Autor: Kleber Brito Moreira

Blockchain - Tecnologia, Arquitetura e Aplicações

Projeto submetido ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Trabalho aprovado. Brasília, DF, 05 de Dezembro de 2019:

Prof. Dr. Ricardo Matos Chaim
Orientador

Prof. Dr. Fábio Cordeiro de Lisboa
Convidado 1

Prof. Dr. Leonardo Aguayo
Convidado 2

Brasília, DF
05/12/2019

Agradecimentos

Agradeço primeiramente à Deus por me dar a oportunidade de estar desenvolvendo um Trabalho de Conclusão de Curso em uma Universidade Federal conceituada como a UnB. Devo também, meus sinceros agradecimentos à minha família, em especial à minha mãe Grimalda, ao meu pai José Moreira e as minhas irmãs Luciana e Adriana pelo apoio nos momentos difíceis desta jornada, sempre me incentivando e mostrando o valor dos estudos. A todos os meus amigos que me ajudaram nessa trajetória E não poderia deixar de agradecer ao meu professor orientador Ricardo Chaim por todo apoio, por sempre estar disposto à ajudar e guiar para que o trabalho fosse concluído da melhor maneira possível.

Resumo

Esse trabalho consiste de uma pesquisa experimental descritiva ao qual o foco principal é a Blockchain. Como a Blockchain ganhou popularidade devido ao Bitcoin, as características dessa criptomoeda foram bastante abordadas e exemplificadas. Moeda Digital é uma das aplicações mais populares da Blockchain, e está ganhando cada vez mais notoriedade, mais adeptos no mundo. Devido a esse aumento de interesse global, esse trabalho aborda a construção de uma criptomoeda chamada FGACoin. Utilizando para isso o Ethereum que é a plataforma mais adequada para esse tipo de desenvolvimento. Portanto é abordado ao longo do trabalho fatores como: o que é o Bitcoin, o que é Blockchain, a economia do Bitcoin, a estrutura de uma Blockchain, mineração de Bitcoins, retarget, rede peer-to-peer, carteira de Bitcoins, como funcionam as transações, as chaves, os custos de mineração, diferenças entre Ethereum e Bitcoin, Contratos Inteligentes, token ERC20, ICO entre outros.

Palavras-chaves: *Blockchain*; Bitcoin; Mineração; Criptomoedas; Moedas Criptografadas; Moedas Digitais; Moedas Virtuais; Arquitetura Blockchain; Aplicações Blockchain; Ethereum; Ether; Contratos Inteligentes; FGACoin; Token ERC20; ICO;

Abstract

This work consists of a descriptive experimental research to which the main focus is the Blockchain. As Blockchain gained popularity due to Bitcoin, the characteristics of this cryptocurrency have been extensively addressed and exemplified. Digital Currency is one of Blockchain's most popular applications, and is gaining more and more notoriety, more fans in the world. Because of this increased global interest, this work addresses the construction of a cryptocurrency called FGACoin. Using Ethereum, which is the most suitable platform for this type of development. Therefore it is covered throughout the work factors such as: what is Bitcoin, what is Blockchain, the economy of Bitcoin, the structure of a Blockchain, Bitcoin mining, retarget, peer-to-peer network, Bitcoin portfolio, how transactions, keys, mining costs, differences between Ethereum and Bitcoin, Smart Contracts, token ERC20, ICO among others.

Key-words: *Blockchain; Bitcoin; Mining; Cryptocurrencies; Encrypted coins; Digital coins; Virtual Currencies; Blockchain architecture; Blockchain applications; Ethereum; Ether; Smart Contracts; FGACoin; Token ERC20; ICO;*

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

Ledger	Livro Razão
RS	Revisão Sistemática da Literatura
QP	Questão de Pesquisa
Nonce	N = Número e Once = uma vez
ECC	Criptografia de Curva Elíptica
BACEN	Banco Central do Brasil
TEMAC	Teoria do Enfoque Meta-Analítico Consolidado

Sumário

1 Introdução

Embora a blockchain derive de suas tecnologias introduzidas décadas atrás, ela ganhou popularidade com o Bitcoin. Tudo começou em 2008, quando um indivíduo (ou grupo) anônimo, sob o pseudônimo de Satoshi Nakamoto publicou um white paper apresentando o Bitcoin. O Bitcoin foi o primeiro exemplo de moeda digital generalizada que fornece uma solução para um problema de confiança em um sistema monetário descentralizado (ASTE; TASCA; MATTEO, 2017). Segundo Li e AlexWang (2016), até hoje o Bitcoin é o exemplo mais significativo de criptomoedas baseadas em Blockchain.

Uma blockchain é essencialmente um banco de dados distribuído de registros ou livro razão público de todas as transações ou eventos digitais que foram executados e compartilhados entre as partes participantes. Cada transação no livro razão é confirmada por consenso da maioria dos participantes no sistema. E uma vez inserida a informação não poderá ser apagada. A principal hipótese é que a blockchain estabelece um sistema de criação de um consenso distribuído no mundo digital online. Isso permite que as entidades participantes saibam com certeza que um evento digital aconteceu, criando um registro irrefutável em um livro público. Abrindo assim a porta para o desenvolvimento de uma economia digital democrática, aberta, escalável e descentralizada. Originalmente concebido como a base das criptomoedas, os aspectos da tecnologia blockchain têm um potencial de longo alcance em muitas outras áreas. Há enormes oportunidades nesta tecnologia disruptiva e a revolução neste espaço acaba de começar.

Neste trabalho é descrito conceitos fundamentais da Blockchain, a tecnologia e arquitetura subjacentes, assim como exemplos de aplicações. Fornecendo assim perspectivas reais sobre seus benefícios, riscos, desafios e oportunidades. Como a Blockchain ganhou popularidade através do Bitcoin, o Bitcoin também foi abordado com grande relevância nesse estudo. Tendo assim sido explorada a tecnologia da Blockchain presente em transações envolvendo Bitcoins. Porém ao longo do desenvolvimento desse projeto constatou-se que o Ethereum continha todos os benefícios presentes do Bitcoin, só que ainda melhorados, correspondendo perfeitamente para a necessidade específica em questão. Em outras palavras o Ethereum é uma plataforma que permite ao desenvolvedor a criação da sua própria criptomoeda. Essas diferenças principais são explicadas com maiores detalhes no tópico: Ethereum Versus Bitcoin. Partindo de todo esse pressuposto foi criado a moeda virtual FGACoin, utilizando a plataforma Ethereum através da interação com os contratos inteligentes.

1.1 Justificativa

A moeda como é conhecida atualmente deve ter 3 propriedades específicas: ser reserva de valor, meio de pagamento e unidade de conta. Porém devido ao avanços tecnológicos, presencia-se uma extinção cada vez maior do papel-moeda, muitas pessoas não tem contato físico com o seu próprio dinheiro, utilizando apenas o dinheiro eletrônico presente em contas bancárias. Com o dinheiro físico se tornando cada vez mais raro, inclusive tendo países que já planejam a sua extinção em um futuro próximo, é preciso se preparar para o que virá adiante. Em um mundo cada vez mais conectado e com acesso a smartphones a pergunta que fica é: para que ficar emitindo dinheiro em papel? Tendo isso em mente vários Bancos Centrais ao redor do mundo estão estudando a tecnologia base do Bitcoin, a Blockchain, para assim poder lançar a sua própria moeda digital.

A blockchain surgiu como uma das tecnologias mais promissoras e potencialmente transformadoras dos últimos tempos. Estudos têm postulado a sua capacidade de interromper extensivamente processos de negócios estabelecidos e gerar confiança e integridade, oferecendo ao mesmo tempo desintermediação e imutabilidade. A blockchain ainda é um conceito relativamente imaturo e estudos destacaram que além da viabilidade, existem poucas aplicações no mundo real, o que representa um dilema para as organizações que buscam a tecnologia para entender o impacto em seus processos existentes (HUGHES et al., 2019).

Conforme pode-se observar em Burgos e Batavia (2018), o Banco Central do Brasil tem iniciado estudos referentes a essa área e tem forte interesse em futuramente também lançar a sua própria moeda digital. Seguindo recomendações em FMI (2018) da publicação emitida pelo Departamento de Estatísticas do FMI (Fundo Monetário Nacional), o BACEN passou em Agosto de 2019 a reconhecer a compra e venda de criptoativos (criptomoedas) na balança comercial do País, conforme conferido em seu site oficial BACEN (2019). A recomendação faz distinção entre “ativos não financeiros produzidos” e “tokens digitais”(que podem incluir security tokens, entre outros), mas a nova metodologia exibida pelo BACEN trata genericamente “criptoativos”, sem diferenciá-los. Tendo tudo isso em vista, esse crescente interesse geral (inclusive do Banco Central do Brasil em moedas virtuais), esse trabalho de conclusão de curso visa o desenvolvimento de sua própria criptomoeda chamada FGACoin. Observando especialmente a dinâmica envolvida na criação e distribuição das moedas virtuais.

1.2 Objetivos

1.2.1 Objetivo Geral

Verificar como a moeda digital pode ser implementada em ambientes virtuais.

1.2.2 Objetivos Específicos

Para atingir o objetivo geral, foram definidos os seguintes objetivos específicos:

- Caracterizar a tecnologia blockchain e como é implementada pelas linguagens de programação tradicionais;
- Criar uma criptomoeda a partir da plataforma Ethereum;
- Conhecer a dinâmica e importância dos Contratos Inteligentes;
- Descrever a dinâmica envolvida na produção e distribuição das moedas digitais;

1.3 Metodologia de Pesquisa

Para reunir evidências, responder os questionamentos iniciais e alcançar o objetivo geral deste trabalho optou-se pela realização de uma pesquisa experimental descritiva. Ela tem como um dos objetivo responder as seguintes questões de pesquisa:

QP1 Caracterizar a tecnologia blockchain e como é implementada pelas linguagens de programação tradicionais.

QP2 Criar uma criptomoeda a partir da plataforma Ethereum.

QP3 Conhecer a dinâmica e importância dos Contratos Inteligentes.

QP4 Descrever a dinâmica envolvida na produção e distribuição das moedas digitais.

Foi definido o processo de estudo da Revisão Bibliográfica com base em diretrizes já existentes de Kitchenham e Charters (2007) e Petersen e Feldt (2008). Este processo compreende três fases principais: planejamento, execução e documentação da revisão. Os trabalhos que forem encontrados nas fontes de busca passarão por um processo de seleção feito com base em critérios específicos previamente definidos, assim os artigos que atenderem aos critérios de busca farão parte da base de estudos da revisão bibliográfica.

No Capítulo 3 será descrito o passo a passo para a condução deste trabalho. Além disso, será apresentado o detalhadamente da revisão bibliográfica e as questões de pesquisas abordadas, bem como o detalhamento da execução da mesma.

1.4 Organização do Trabalho

Este trabalho está organizado como segue. No Capítulo 2 é apresentado o Referencial Teórico, contendo os conceitos necessários para o entendimento deste trabalho. As figuras 05 até 09 foram criadas pelo autor no site Brownworth (2019). O Capítulo 3 apresenta a metodologia de desenvolvimento aplicada na elaboração deste trabalho, bem como o detalhamento da revisão bibliográfica e sua execução. O Capítulo 4 apresenta os resultados obtidos para os questionamentos de pesquisa a partir desse estudo. O Capítulo 5 apresenta a conclusão deste trabalho e mostra quais os objetivos alcançados com a elaboração do mesmo. E por último encontrasse no apêndice um roteiro detalhado para a replicação desse experimento.

2 Referencial Teórico

Este capítulo apresenta conceitos importantes a respeito de Bitcoin, Blockchain, Economia do Bitcoin, Estrutura, Arquitetura, Tecnologia e aplicações da Blockchain, Mineração, Retarget, arquitetura peer-to-peer, carteira, transações, custo de produção do Bitcoin, diferenciação do Ethereum para o Bitcoin entre outros. As informações apresentadas foram adquiridas através de pesquisas bibliográficas e tem como objetivo fundamentar a pesquisa e os resultados encontrados bem como prover um melhor entendimento a cerca do objeto de estudo deste trabalho.

2.1 Bitcoin

Zhang e Wen (2016) apresentam o Bitcoin como uma moeda digital eletrônica, criptografada, autônoma e do usuário. O Bitcoin não é emitido por nenhuma nação ou organização, dessa forma não há necessidade dos usuários de Bitcoin se preocuparem com o encerramento de suas contas ou com a depreciação da moeda causado pelo excesso de dinheiro impresso. O advento do Bitcoin cria a era da descentralização. Mas o Bitcoin não é apenas uma moeda, mas também um protocolo, uma rede e uma linguagem de transação. De acordo com Li e AlexWang (2016), o Bitcoin é parte de uma nova geração de sistemas monetários digitais construídos com tecnologia computacional e arquitetura de rede descentralizada (peer to peer). Para servir como meio de pagamento e armazenamento de valor, o Bitcoin cria um sistema de autenticação descentralizado para lidar com problemas de falsificação e duplicidade, enquanto sistemas monetários fiduciários modernos e sistemas de pagamentos digitais exigem das instituições centrais, autenticação em transações e que também sirvam como repositórios.

O Bitcoin revolucionou o campo das moedas digitais e influenciou muitas áreas próximas. Desde o inicio dos anos 80 a visão do dinheiro digital já existia, mas levou mais de 25 anos para que uma solução totalmente distribuída se tornasse realidade. As primeiras tentativas de construir moedas digitais necessitava de um banco, isto é, uma autoridade central. Com o intuito de eliminar o banco, o livro razão (responsável por registrar as operações) também deveria ser distribuído. Após a sua implantação em 2009 o Bitcoin rapidamente se tornou viral. O Bitcoin é de longe a moeda digital mais amplamente conhecida e se tornou responsável por um aumento na área de pesquisa sobre o tema. O enorme sucesso de Bitcoin atraiu uma grande quantidade de usuários. A maioria de usuários de Bitcoin parece ser classificável como nerds, investidores, ideólogos ou criminosos. De qualquer forma a maneira mais fácil de adquirir Bitcoin é comprando moedas em uma exchange. O preço do Bitcoin aumentou rapidamente a medida que foi se tornando mais

popular. 5 anos após o seu lançamento a taxa de câmbio já tinha ultrapassado a marca de U$ 1000,00 por Bitcoin. O Bitcoin transformou muito dos seus primeiros adeptos em milionários e ainda compreende uma economia bilionária (TSCHORSCH; SCHEUERMANN, 2016).

Li e AlexWang (2016) afirmam que a maioria de usuários de Bitcoin não se dedica a mineração para ganhar novas moedas, eles compram Bitcoins de outros com as suas respectivas "moedas locais". As trocas online em que os usuários negociam Bitcoins com outras moedas, são componentes importantes do ecossistema das criptomoedas. Isso é importante pois conectam as criptomoedas com a economia do mundo real, na qual as transações são efetuadas com moedas locais. De acordo com Lischke e Fabian (2016), para evitar a inflação no sistema, uma característica única é que ela tem um número limitado, pré-determinado de 21 milhões em circulação. Esse valor que segundo especialistas pode ser alcançado em 2040.

2.2 Economia do Bitcoin

A figura 1 mostra uma visão geral da economia do Bitcoin contendo os seus principais participantes. Sendo numerados e expostos a seguir, em conformidade com Lischke e Fabian (2016):

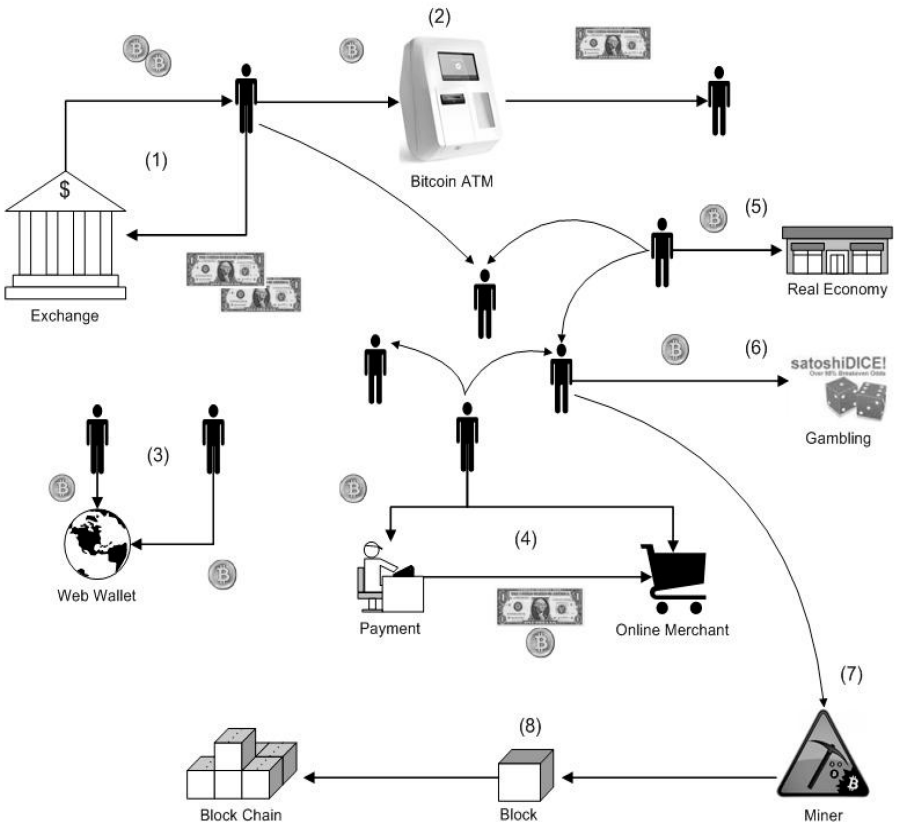

Figura 1 – Economia do Bitcoin. Fonte: (LISCHKE; FABIAN, 2016)

(1) Os usuários podem trocar suas moedas fiduciárias por Bitcoins via plataforma de trocas (Exchange) ou trocas locais;

(2) Retirar dinheiro dos caixas eletrônicos de Bitcoin;

(3) Armazenar Bitcoins em uma carteira online;

(4) Usar serviços de pagamento online em transações com comerciantes;

(5) Pagar com Bitcoins em lojas ou bares locais;

(6) Apostar Bitcoins em plataformas de jogos;

(7) Minerar, ou seja, participar do processo de incorporação de transações em blocos;

(8) Verificar as transações e publicá-las na rede através da cadeia de blocos.

Lischke e Fabian (2016) mostram que há uma grande quantidade de comerciantes e serviços que aceitam Bitcoins como forma de pagamento. Os serviços podem ser categorizados principalmente em trocas, carteiras, mineração, pagamentos, jogos de azar e fornecedores.

• Trocas: as trocas podem ser por moedas fiduciárias, outras moedas criptografadas, e até mesmo ouro por Bitcoins. As plataformas de trocas são principalmente eletrônicas, mas também há trocas locais.

• Carteiras: as carteiras eletrônicas são semelhantes aos bancos na economia real, onde os Bitcoins podem ser armazenados centralmente em plataformas online. A principal vantagem é que os usuários podem acessar Bitcoins de todos os dispositivos conectados a web e têm menos esforço para proteger sua carteira.

• Pagamentos: os serviços de pagamento permitem que os comerciantes on-line aceitem os Bitcoins da mesma forma que aceitam pagamentos com Visa ou Paypal na moeda local. Reduz os custos de transação, evita estornos, riscos de taxa de trocas de Bitcoin e roubos de identidade.

• Jogos de Azar: Jogos de azar: os serviços de jogos de azar oferecem uma ampla variedade de jogos on-line, como jogos de dados, roleta e outros jogos relacionados a cassinos, em que os usuários podem apostar com seus Bitcoins.

• Fornecedores: através de comerciantes on-line, os usuários podem trocar seus Bitcoins por quase todos os tipos de produtos, como conteúdo multimídia, eletrônicos, viagens, cartões de presente, roupas, entre outros. Há também fornecedores que funcionam como mercados, como o Ebay.

• Mineração: bitcoinmining (2014) apresenta a mineração como a contribuição para o processo de geração da moeda, executado principalmente em uma piscina de mineração. Em tais piscinas de mineração, os mineradores compartilham seus recursos de computação

e cada participante recebe uma recompensa pela contribuição específica do poder de computação;

Segundo Tschorsch e Scheuermann (2016), o Bitcoin nos seus primórdios almejava uma solução muito mais ambiciosa do que as moedas tradicionais, uma que se livrasse do banco central. Para esta finalidade é necessário um mecanismo que criem moedas em um ambiente distribuído para armazenar e gerenciar. O principal desafio era obter consenso entre as moedas participantes, sem a necessidade de uma autoridade central e relação de confiança entre os envolvidos. Então foi pensando em como resolver o problema de não precisar de um banco como autoridade central? O Bitcoin resolveu isso utilizando a tecnologia Blockchain, ou seja, todos os participantes mantém uma cópia dos registros que seriam armazenados no banco central.

2.3 Blockchain

Conforme Khan e Salah (2017), a tecnologia Blockchain foi vista pela indústria e pela comunidade científica como uma inovação tecnológica, provocando assim uma ruptura nos modelos ou tecnologias existentes. Uma Blockchain (cadeia de blocos) basicamente é um banco de dados descentralizado, distribuído, compartilhado e imutável que armazena o registro de transações em uma rede Peer-to-peer (P2P). Desempenhando um papel importante no gerenciamento, controle e proteção de dispositivos.

Treleaven, Brown e Yang (2017) afirmam que a Blockchain possuem vários atributos atraentes para os mercados bancário e de serviços financeiros. Os sistemas são resilientes e podem operar como redes descentralizadas que não requerem um servidor central e não possuem um único ponto de falha. Como eles operam usando protocolos de código aberto distribuídos, eles têm integridade e não precisam confiar em terceiros para executar transações. Os sistemas públicos de blockchain são transparentes, porque todas as alterações são visíveis. A Blockchain opera com um alto grau de confiança, pois as transações são imutáveis, ou seja, não pode ser revertida. Em geral, os sistemas Blockchain são capazes de garantir que todas as partes envolvidas tenham registros precisos e idênticos.

Para Aste, Tasca e Matteo (2017), tanto o setor público quanto o setor privado têm grandes expectativas em relação a tecnologia Blockchain porque fornecem a base para o desenvolvimento de plataformas peer-to-peer para troca de informações, ativos e bens digitalizados sem intermediários. A blockchain tem o potencial de mudar radicalmente muitos setores econômicos e melhorar a aplicação de setores regulatórios e de governança de uma maneira completamente inovadora.

De acordo com Khan e Salah (2017), a Blockchain tem um histórico completo com todas as transações e fornece uma confiança global distribuída. Na blockchain cada

transação compartilhada no *ledger* público é verificada pelos nós mineradores, que estão ativamente envolvidos na verificação e validação de transações, de forma que exista um consenso majoritário. Uma vez que as transações são validadas e verificadas por consenso, os dados dos blocos serão imutáveis, ou seja, não poderá ser apagado ou modificado.

2.4 Arquitetura

Para Liang et al. (2017), quando se trata desse tema tem como característica de ser a Blockchain uma solução própria em arquitetura de software que foi pensada e desenvolvida tendo como princípio solucionar problemas característicos que envolviam redes distribuídas, que são aquelas que se contrapõem as redes centralizadas, ou seja, não havendo a necessidade de um servidor central. Dessa forma uma arquitetura Blockchain tem como características a comunicação entre os nós (computadores) presentes na rede P2P sendo assim possível o compartilhamento de dados e serviços, proporcionando uma melhor eficiência. A figura 2 mostra a estrutura típica de uma Blockchain.

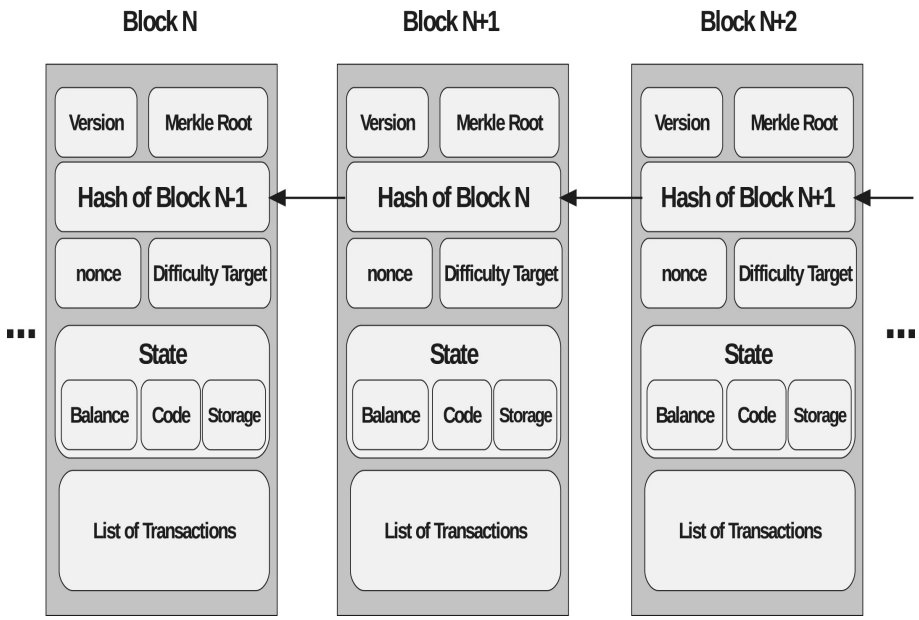

Figura 2 – Estrutura de design de Blockchain mostrando blocos encadeados com campos de cabeçalho e corpo. Fonte: (KHAN; SALAH, 2017)

A estrutura é composta principalmente pelo cabeçalho e corpo do bloco, que contém uma lista de transações (List of Transactions). Os dados presentes em cada bloco contém uma lista de todas as transações e um hash para o bloco anterior. O cabeçalho do bloco contém vários campos, um dos quais é um número de versão (Version) para rastrear software de atualizações de protocolo. O campo Merkle Root representa o valor de hash do bloco atual. Merkle tree hashing é comumente usado em sistemas distribuídos e redes P2P para a verificação eficiente de dados. O campo nonce é usado para o algoritmo de

prova de trabalho e é o valor do contador de teste que produziu o hash com zeros à esquerda. O alvo da dificuldade (Difficulty Target) específica o número de zeros à esquerda e é usado para manter o tempo de bloqueio. O alvo de dificuldade é ajustável periodicamente e é aumentado (com mais zeros à esquerda) à medida que o poder de computação do hardware aumenta ao longo do tempo. O tempo de bloqueio é definido por projeto para levar em conta o tempo de propagação dos blocos para alcançar todos os mineiros e para que todos os mineradores cheguem a um consenso (KHAN; SALAH, 2017).

Para Tschorsch e Scheuermann (2016), é possível formar uma cadeia de blocos porque cada bloco contém um ponteiro para o bloco validado antes dele na cadeia. O ponteiro é implementado contendo o próprio hash e o hash do bloco anterior. Consequentemente a cadeia de blocos tem a estrutura de uma lista encadeada. Por causa da mineração contínua a cadeia de blocos cresce constantemente. Conforme é mostrado na figura 3.

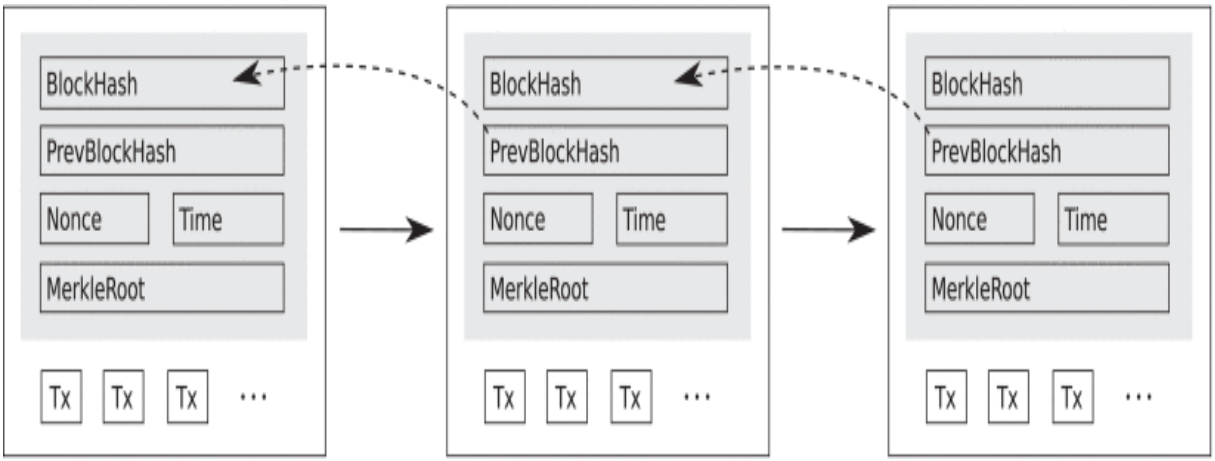

Figura 3 – Encadeamento de Blocos. Fonte: (TSCHORSCH; SCHEUERMANN, 2016)

O hash ele funciona como a impressão digital do dado do respectivo bloco. No número do Bloco (se ele for o primeiro bloco) diz se que é o Bloco Gênese, o bloco de altura 1. A altura do bloco é determinada pela quantidade de blocos que tem da cabeça a cauda. O Bloco Gênese já que é o primeiro bloco da cadeia (não vem nenhum antes dele) então ele tem valor 0 para o (Hash of Block N – 1 ou PrevBlockHash) hash do bloco anterior.

Em se tratando do ponto de vista arquitetural, vale a pena observar que existe diferentes tipos de blockchains que diferem em termos de permissões de leitura ou gravação. Blockchains públicos (como o Blockchain Bitcoin) são Blockchains que podem ser lidos e potencialmente graváveis por todos. Já os Blockchains privados são aqueles que podem ser escritos apenas por membros da organização. As permissões de leitura podem ser públicas ou restritas a organização. Em blockchains de consórcio, um conjunto de nós selecionados pertencentes a diferentes instituições controla a validação, e o blockchain é usado para compartilhar informações entre as instituições participantes. As blockchains

públicas são particularmente úteis quando nenhuma entidade central está disponível para verificar uma transação e a descentralização total é necessária. Os blockchains privados e de consórcio fornecem algumas vantagens, como menores custos de validação e tempos de validação mais curtos (dado o fato de que, devido ao menor número de nós, o problema matemático pode ser simplificado), menor risco de ataques (já que nós que validam transações são conhecidos) e maior privacidade sendo que as permissões podem ser concedidas apenas a nós selecionados). Além disso, em caso de erros ou erros em contratos inteligentes, os blockchains privados ou de consórcio poderiam modificar ou reverter transações anteriores de forma extraordinária (GATTESCHI; LAMBERTI; DEMARTINI, 2018).

2.5 Rede peer-to-peer

Segundo Ouaddah, Elkalam e Ouahman (2017), a rede peer-to-peer forma uma malha fracamente conectada sem uma topologia fixa ou qualquer estrutura, tornando todos os nós iguais, ou seja, todos os nós da rede podem operar tanto como cliente quanto como servidor. As mensagens, incluindo transações e blocos, são propagadas de cada nó para os pares aos quais está conectado. Embora os nós na rede P2P sejam iguais, eles podem assumir diferentes funções, dependendo é claro da funcionalidade ao qual está suportando. Um nó completo na rede é uma coleção de funções: roteamento, banco de dados blockchain, mineração e serviços de carteira. Para Ouaddah, Elkalam e Ouahman (2017), os remetentes não precisam confiar nos nós que usam para transmitir a transação, desde que use mais de 1 nó para garantir a propagação. Os nós não precisam confiar no remetente ou estabelecer a sua respectiva identidade. As transações portanto podem serem transmitidas para a rede P2P por conexões inseguras como: Wifi, Bluetooth, NFC (Near Field Communication) e código de barras. Para se tornar parte da rede peer-to-peer, é necessário ter um software cliente que seja executado no próprio dispositivo ou como serviço de nuvem (OBER; KATZENBEISSER; HAMACHER, 2013).

Figura 4 – Arquitetura distribuída de uma rede Blockchain. Fonte: (PITZ, 2017)

2.6 Mineração

Qualquer alteração realizada em um dos dados do bloco, irá alterar o valor do seu respectivo hash. Alterando o valor do hash, irá alterar o valor de hash que foi para o bloco seguinte (N+1). Tornando o bloco atual e todo seguinte a ele um bloco não minerado, portanto um bloco não válido na rede. Um Bloco é considerado minerado todas as vezes em que o minerador encontrar um nonce que faça que o hash esteja abaixo da dificuldade da rede, tendo assim uma certa quantidade de 0 no valor inicial do hash, ou seja, o valor esteja abaixo do limite pré-estabelecido. Esse é o conceito de prova de trabalho, o computador do minerador vai ter de testar vários nonce até encontrar o valor correspondente designado para minerar aquele bloco específico (TAYLOR, 2017). A Blockchain utiliza criptografia de curva elíptica (ECC) e Hashing SHA-256 para fornecer uma forte prova criptográfica para autenticação e integridade dos dados (KHAN; SALAH, 2017). A figura 5 mostra um exemplo da Blockchain do Peer A devidamente minerada.

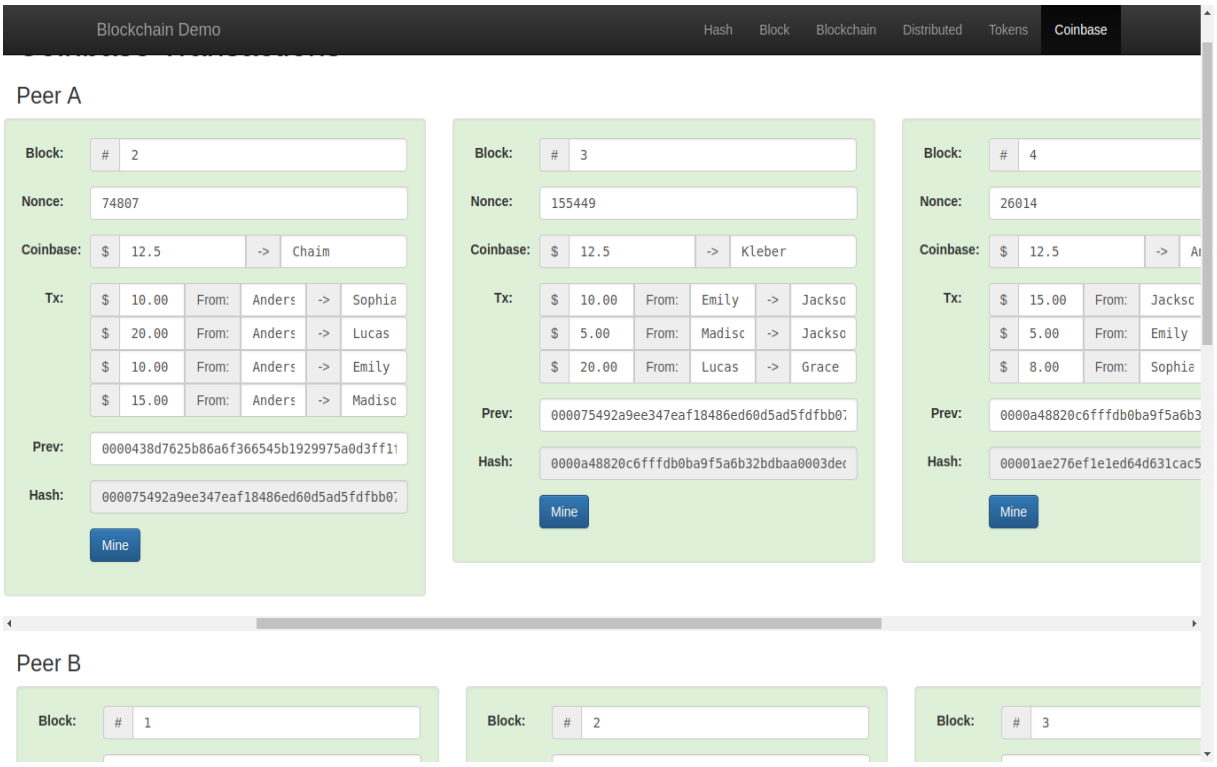

Figura 5 – Blockchain Peer A Minerada. Fonte: Autor

Coinbase: de acordo com Göbel (2016), é o valor ganho pelo minerador por gerar um novo bloco válido para a Blockchain. Essa recompensa se iniciou em 50 BTC (Bitcoins). Diminuindo pela metade a cada 210000 blocos. O minerador recebe atualmente 12,5 bitcoins por realizar esse trabalho, juntamente com uma pequena taxa coletada de cada transação no bloco.

Porém caso algum dado seja modificado no Bloco 2, o Hash será automaticamente modificado, ficando assim fora do limite de dificuldade que nesse exemplo é de 0000

iniciais. Tornando assim o bloco atual e todos subsequentes a ele não válidos. Conforme observado na figura 6.

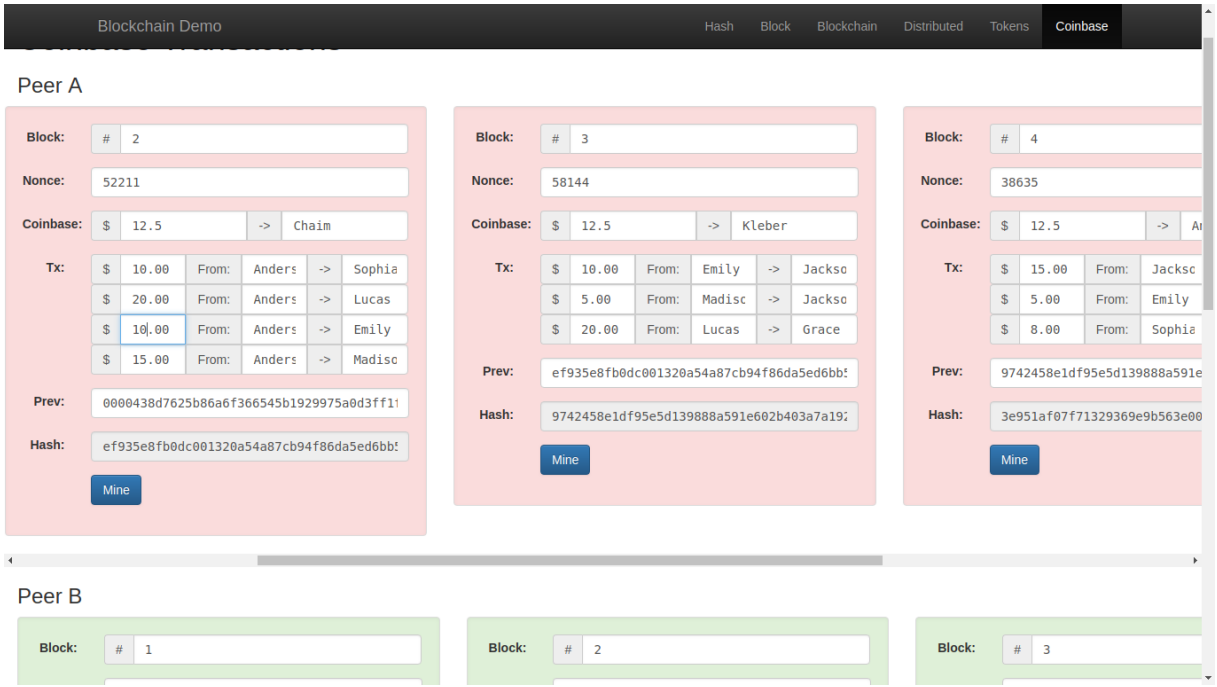

Figura 6 – Blockchain Peer A Incorreta. Fonte: Autor

Minerando somente o bloco 2 não adiantará para arrumar a cadeia conforme vemos na figura 7.

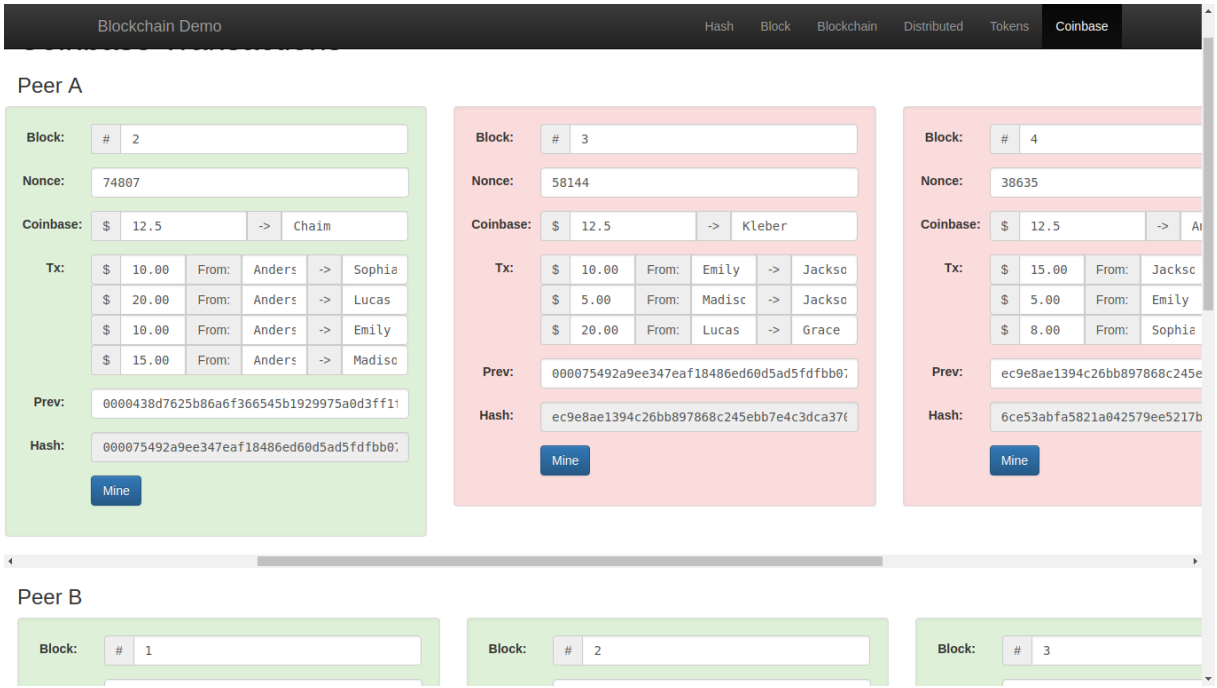

Figura 7 – Bloco 2 Minerado. Fonte: Autor

Devido a isso é fácil observar que é muito difícil modificar um dado já estabelecido na Blockchain. Para arrumar a cadeia, terá de ser minerado o bloco atual e cada bloco subsequente (o que necessita de um grande esforço computacional) e ainda tentar convencer os outros peers que a sua cadeia é a correta, isso tudo em um intervalo de tempo curto.

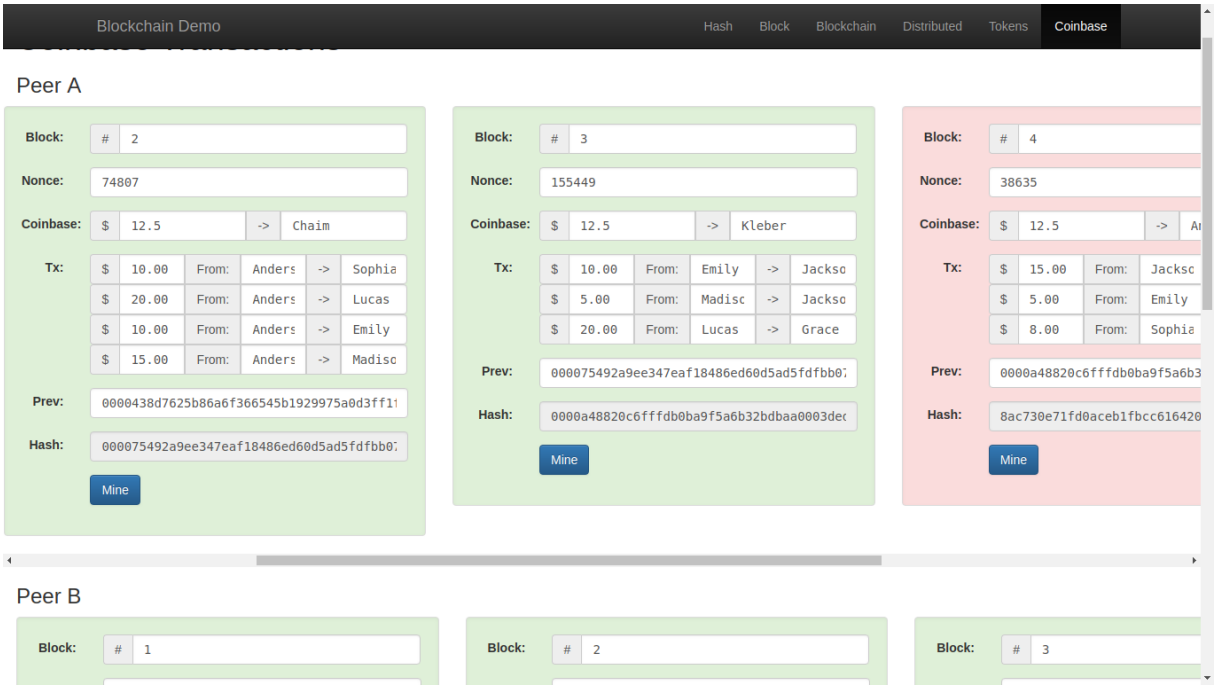

Figura 8 – Bloco 2 e 3 Minerado. Fonte: Autor

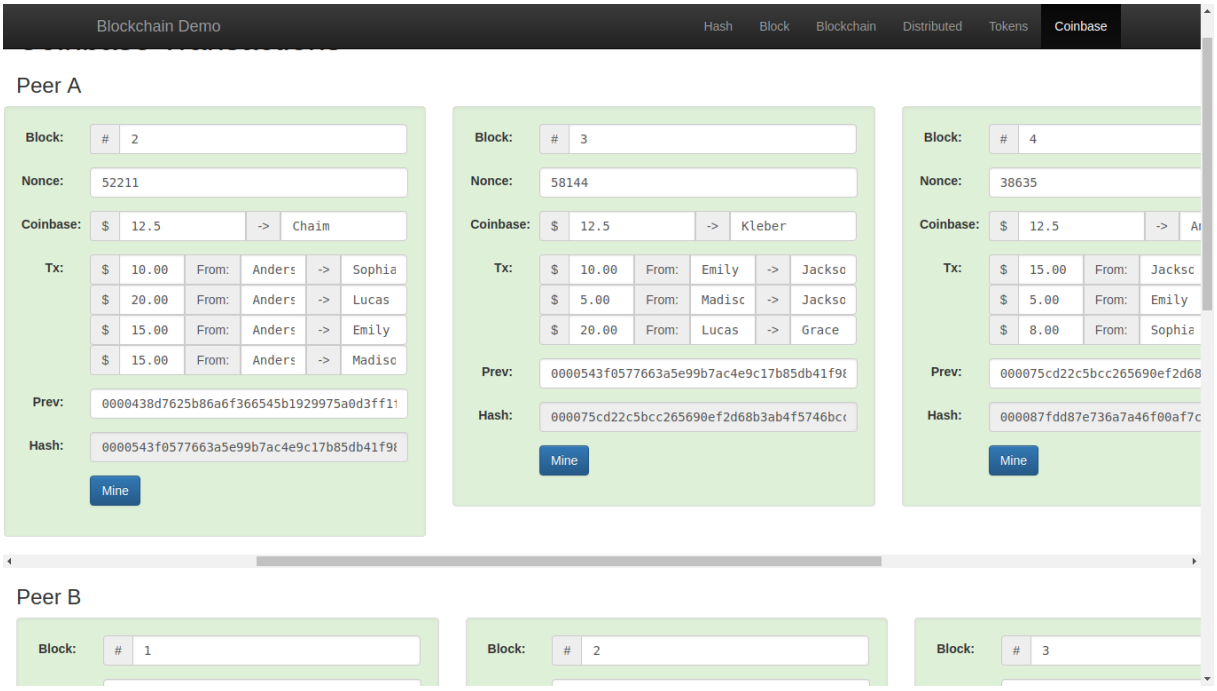

Figura 9 – Toda a Blockchain Minerada. Fonte: Autor

2.7 Retarget

Para S.Hayes (2016), a mineração é bastante competitiva, no sentido de que alguém minerando com mais poder computacional, ou com maior eficiência, tem maiores chances de minerar novos Bitcoins do que alguém com menos. Esforço computacional na produção de criptomoedas são muitas vezes referidas como hashpower, hashing power, mining effort or hashrate.

De acordo com Tschorsch e Scheuermann (2016), por motivos de estabilidade e também de ter um tempo de espera razoável para validação da transação, o valor alvo é ajustado a cada 2016 novos blocos. É então re-escolhido para atender a uma taxa de verificação de aproximadamente 1 bloco a cada 10 minutos. Assim o novo alvo T é dado por:

T = Tprev * (Treal/2016*10 min)

Onde Tprev é o valor alvo antigo, Treal é o tempo que levou para gerar os últimos 2016 novos blocos. Em média a cada duas semanas o valor alvo é recalculado. Porém se 2016 novos blocos foram gerados durante um período de tempo menor do que duas semanas, isso indica em geral que o poder computacional aumentou e com isso consequentemente a sua complexidade (comprovação de trabalho) (TSCHORSCH; SCHEUERMANN, 2016).

Segundo S.Hayes (2016), um bloco de criptomoedas por definição é para ser encontrado por mineração no mesmo intervalo, independentemente da magnitude e do esforço de mineração. Blocos de Bitcoin serão criados 1 vez em média a cada 10 minutos. E essa variação é referida como tempo de bloqueio. A rede verificará se o tempo médio de bloqueio foi atingido, utilizando os blocos minerados anteriormente. O sistema verificará se o tempo médio de criação foi maior ou menor do que 10 minutos. Se for menor do que 10 minutos, o sistema aumentará a dificuldade em encontrar novos blocos, para que assim a média de 10 minutos fosse restaurada. Esse procedimento é chamado de Difficulty Retarget.

S.Hayes (2016) relata que é importante observar que ao empregar um maior poder computacional (hardware na mineração) na rede, isso pode aumentar a probabilidade de sucesso do minerador individual que tenha maior poder na rede, o tornando mais produtivo. No entanto a rede verificará o Difficulty Retarget, e ajustará a dificuldade de acordo com a restauração do tempo de bloqueio. Dessa maneira se alguém colocasse online uma nova tecnologia poderosa, com muito poder computacional, uma vez que a rede detectará que o tempo médio de criação de novos blocos está sendo baixo, ajustaria a dificuldade de acordo com a situação, tornando a nova tecnologia meramente adequada, e tornando a tecnologia de outro mineiro inferior ou até mesmo obsoleta.

2.8 Carteira

De acordo com Ouaddah, Elkalam e Ouahman (2017), a principal função dos aplicativos de carteira é gerenciar o seu par de chaves. Todo usuário tem uma carteira que armazena suas credenciais, endereços e transações relacionadas a eles. Ele contém todas as chaves necessárias para registrar e identificar os seus recursos, assinar as suas transações e solicitar acesso. As principais funcionalidades de uma carteira são:

(1) gerar chaves e endereços;

(2) Transformar as políticas de controle de acesso em transações e as transmitir posteriormente a rede;

(3) Validar transações recebidas na rede.

Os usuários podem enviar e receber eletronicamente Bitcoins utilizando o software Wallet, disponível tanto para dispositivos móveis quanto para computadores (S.HAYES, 2016).

2.9 Chaves

Ober, Katzenbeisser e Hamacher (2013) mencionam que para fornecer algum tipo de anonimato informações pessoais diretamente identificáveis são omitidas da transação. Portanto, os endereços de origem e destino são codificados na forma de chaves públicas. Cada chave pública que serve como um pseudônimo tem uma chave privada que é armazenada na carteira eletrônica. Estes são usados para assinar ou autenticar quaisquer transações. Conforme Göbel (2016) essas chaves são assimétricas, ou seja, elas formam um par perfeito. Se uma encripta, só o par correspondente dela pode desencriptar e vice versa. Por exemplo: suponha que você receba uma transação de Ricardo, se você descriptografou a mensagem de Ricardo usando a sua chave pública, então você confirmou que a mensagem foi criptografada utilizando a chave privada de Ricardo, e portanto a mensagem veio indiscutivelmente de Ricardo.

2.10 Transações

O *ledger* contém o registro de todas as transações. Cada membro da Blockchain pode ter uma cópia sincronizada do *ledger*. Transferências em Bitcoins usam criptografia de chave pública. Os pagadores e beneficiários são identificados pelas chaves públicas de suas identidades de carteira do Bitcoin. Cada transação Bitcoin é criptografada e transmitida pela rede. Por exemplo: suponha que a usuária Alice queira transferir Bitcoins para o usuário Bob. O "aplicativo de carteira" da Alice irá criar uma transação, especificar o valor, inserindo a chave pública dela e do destinatário, assinar com a chave privada dela

e enviar para a Blockchain. Esses dados serão incluídos no *Ledger* e os nós mineradores serão notificados da solicitação de nova transação. O minerador primeiro verifica se Alice tem o saldo necessário. Mas como é verificado se Alice tem Bitcoins suficientes para transferir? Os mineradores de Bitcoin recebem cópias de todas as transações à medida que são geradas. Eles examinam a Blockchain para investigar o histórico dos Bitcoins envolvidos em cada transação. Se a transação proposta tiver crédito de Bitcoin suficiente, ela será aceita para a incorporação no Bloco ao qual o minerador esteja trabalhando atualmente. Após essa validação o minerador executará a prova de trabalho, ou seja, encontrar o valor randômico designado ao nonce para aquele hash. O nó minerador que primeiro conseguir executar esses passos avisará os outros nós mineradores. Os mineradores validarão se os procedimentos estão corretos e caso a maioria concorde que é válido (consenso majoritário), o novo bloco será confirmado e aquele nó minerador que gerou o bloco receberá a recompensa (GöBEL, 2016).

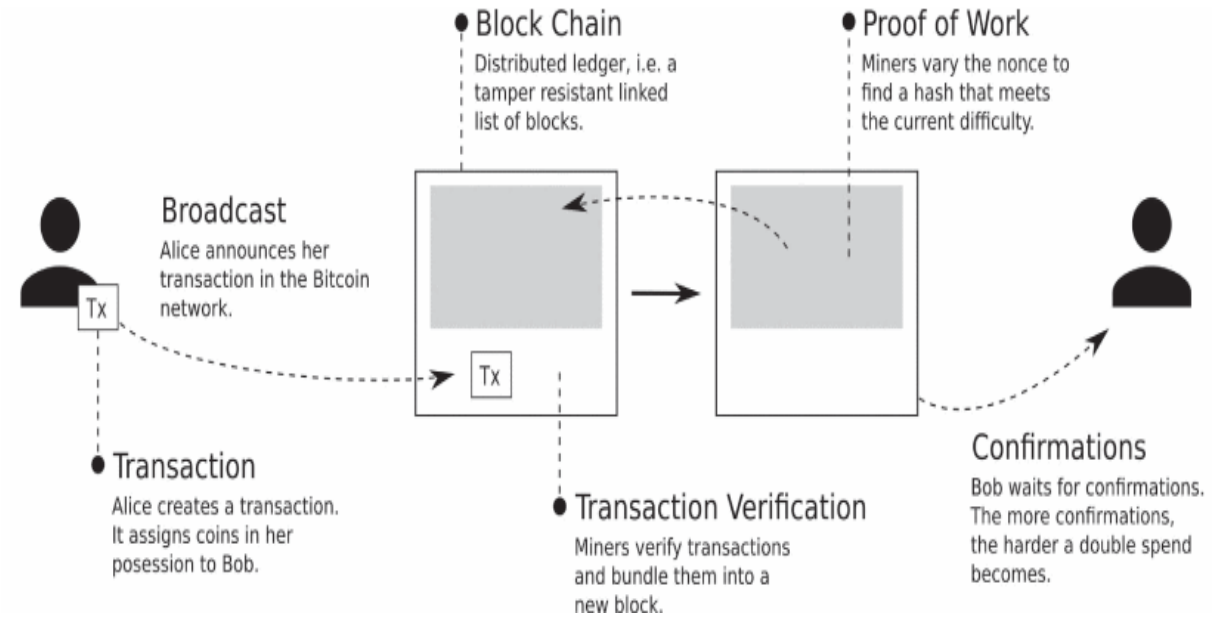

Figura 10 – Etapas de uma transferência de Alice para Bob. Fonte: (TSCHORSCH; SCHEUERMANN, 2016)

2.11 Segurança das transações

A Blockchain é uma tecnologia que busca garantir a segurança e a eficácia das transações. Para atender a esses requisitos de obrigatoriedade e confiabilidade o mecanismo é listado da seguinte maneira (ZHANG; WEN, 2016):

- A Blockchain pode ser construída como: uma rede autorizada (privada), ou sem a permissão (rede pública). (1) Rede autorizada (ou privada) que pode ser restrita a um determinado grupo de participantes. (2) Rede sem a permissão (ou pública) que esteja

aberta para qualquer pessoa participar. Blocos de bloqueios de permissão fornecem mais privacidade e melhor controle de acesso (KHAN; SALAH, 2017)

- Assinatura Eletrônica: Como a chave privada é impossível de falsificar, a identidade dos participantes pode ser verificada se ele assinou a sua chave privada no contrato (ZHANG; WEN, 2016).

- Rede Ponto a Ponto (P2P): Semelhante ao BitTorrent e TCP / IP não pode ser destruída porque opera sem o suporte de uma organização central (ZHANG; WEN, 2016).

- Prova de Trabalho: A prova de trabalho envolve o uso de algoritmos hash como o SHA-256, para encontrar um valor específico (CLACK; BAKSHI; BRAINE, 2016). A ideia principal por trás do sistema de prova de trabalho é tornar caro para um usuário, ou um grupo de usuários reescrever o histórico de transações, uma vez que tenha sido aceito como definitivo. Isso deve impedir que usuários mal intencionados causem fraudes nas transações (OBER; KATZENBEISSER; HAMACHER, 2013).

- Razão Distribuído: O *Ledger* é organizado como uma cadeia de blocos. O Blockchain contém blocos de registros de transações validados para rastrear a propriedade de cada Bitcoin. Cada registro de transação contém a chave pública do destinatário. Em uma transação Bitcoin, o proprietário atual valida a sua propriedade usando a chave privada e envia uma instrução de transação criptografada com sua chave privada. Em seguida o sistema registra a instrução de transação contendo a chave pública do receptor (o novo proprietário) em um novo bloco (LI; ALEXWANG, 2016). O mecanismo Blockchain significa que todos podem examinar a existência de uma determinada transação e verificar a eficácia dela (ZHANG; WEN, 2016). E por fim uma observação importante: nem todos os *ledgers* distribuídos são blockchains, mas todos os blockchains são *ledgers* distribuídos (TRELEAVEN; BROWN; YANG, 2017).

- Algoritmo de Hashing: Toda transação é criptografada pelo algoritmo de Hashing e é adicionada a parte inferior da Blockchain mais recente em tempo real (ZHANG; WEN, 2016). Para proteger a integridade do *ledger* o sistema protege e portanto rótula cada bloco com um hash exclusivo. O hash gerado precisa atender a um critério de "taxa de hash" definido pelo sistema. Um novo bloco contendo transações só é adicionado a blockchain quando um hash válido é encontrado. Gerar um hash utilizando uma chave é fácil. No entanto engenharia reversa de uma chave de hash é criptograficamente difícil. Um hash válido que atenda ao critério de taxa de hash é descoberto por meio de tentativa e erro, o que requer um poder computacional significativo (LI; ALEXWANG, 2016). Se um hacker quiser cancelar ou modificar uma transação, ele precisa descriptografar a Blockchain para reverter isso, enquanto toda a rede está adicionando continuamente novos blocos. Em outras palavras, ele deve oferecer mais de 51% de toda a capacidade computacional da Blockchain. O que é praticamente impossível porque a rede é desenvolvida em grande

escala (ZHANG; WEN, 2016).

2.12 Valor do Bitcoin

O valor do Bitcoin (e de outras moedas criptografadas) pode ser expresso como sua taxa de câmbio em relação a outras moedas. Alguns estudos foram elaborados utilizando análises econométricas com o intuito de observar os fatores determinantes da taxa de câmbio do Bitcoin.

• Kristoufek (2013) estudou o impacto do volume de pesquisas do Google e das visualizações diárias na Wikipédia. O pesquisador encontrou uma correlação significativa entre pesquisa e preço;

• Kristoufek (2015) conduziu uma análise de coerência wavelet para identificar a correlação entre os preços de troca do Bitcoin e outros fatores. O resultado da estimativa revelou que fatores como a relação troca comércio e o comportamento especulativo desempenham um papel significativo nas frequências mais baixas. O pesquisador também sugeriu que o índice do mercado chinês pode ser um dos principais impulsionadores do preço do Bitcoin;

• Garcia et al. (2014) esse estudo se baseou em observar para o efeito boca a boca online nas redes sociais Twitter e Facebook;

• Bouoiyour (2015) foi identificado um conjunto de fatores determinantes, incluindo a pesquisa do Google, a proporção do volume de comércio de câmbio, a taxa de hash e o mercado de ações, utilizando a abordagem de testes de limites de ADRL;

• Ciaian e Rajcaniova (2015) descobriram que o volume de transações, o volume de usuários e a atratividade (medidos pelas postagens no Fórum e nas visualizações da Wikipédia) têm impactos significativos no preço do Bitcoin. Eles também encontraram mudanças significativas ao longo do tempo;

• Polasik (2014) estudou o impacto do volume de notícias, sentimento, pesquisa do Google, quantidade de transações, número de Bitcoins e fatores econômicos (crescimento da produção industrial, desemprego e inflação) no retorno mensal do Bitcoin e descobriu que os retornos são impulsionados principalmente pelo volume de notícias, sentimento de notícias, e o número total de transações;

2.13 Custo de produção

Conforme S.Hayes (2016), cada unidade de esforço de mineração tem um custo fixo irrecuperável envolvido na compra, transporte e instalação do hardware de mineração. As variáveis importantes que devem ser levadas em conta na decisão de minerar são :

(1) O custo da eletricidade, medido em centavos por quilowatt-hora;

(2) O consumo de energia por unidade de esforço de mineração, medido em watts por GH / s (ou Joules por GH), em função do custo da eletricidade e eficiência energética;

(3) O preço monetário do Bitcoin no mercado;

(4) A dificuldade do algoritmo;

A recompensa do bloco minerado é importante, mas essa só muda em média a cada 4 anos. O principal custo na mineração de Bitcoins é o consumo de energia que é necessário para o trabalho computacional na mineração (os outros custos podem ser considerados insignificantes como: serviço de internet, manutenção de hardware, cabos de computador, entre outros) (S.HAYES, 2016).

De acordo com Li e AlexWang (2016), o custo da energia computacional na mineração também depende da tecnologia de mineração. Desde a introdução do Bitcoin, existem quatro gerações de dispositivos de mineração, ou seja, Unidades de Processamento Central (CPUs), Unidades de Processamento Gráfico (GPUs), FPGAs (Field Programmable Gate Arrays) e ASICs (Application-Specific Integrated Circuits), onde cada geração é sucessivamente mais eficiente na mineração de Bitcoin. Os dispositivos baseados em GPU substituíram gradualmente os dispositivos baseados em CPU em 2011. Os dispositivos FPGA foram oferecidos pela primeira vez em janeiro de 2012. Até 2014, os dispositivos de mineração FPGA dominavam o mercado. Os dispositivos ASIC foram introduzidos no início de 2013, mas receberam menos adoção do que os dispositivos FPGA até 2014, porque exigem um investimento inicial maior.

2.14 Aplicações

Para Gatteschi, Lamberti e Demartini (2018), a tecnologia Blockchain pode ser implementada em muitas áreas, sendo claramente o setor bancário ou serviços financeiros aqueles onde tem a maior quantidade de investimentos. Porém com o passar do tempo, os pesquisadores perceberam que a Blockchain poderia ser usada para armazenar outros tipos de ativos, incluindo pedaços de código. Assim foi decretado o nascimento dos contratos inteligentes, isto é, pequenos programas armazenados na blockchain e programados para se comportarem autonomamente de uma determinada maneira quando algumas condições são atendidas. Com um contrato inteligente, uma pessoa poderia codificar sua vontade na blockchain na forma de um conjunto de regras. Por exemplo: em caso de morte, o contrato inteligente poderia transferir automaticamente o dinheiro do testador ou outro tipo de ativos para o beneficiário. O testador também pode fornecer restrições adicionais, como permitir a transferência somente quando o beneficiário atingir a maioridade, ou quando obtém um diploma, entre outros.

Como as condições dos contratos inteligentes são baseadas em dados armazenados na blockchain, eles precisam contar com serviços externos que pegam dados do mundo real (por exemplo os registros de óbitos) e os empurram para a blockchain (ou vice-versa). Estes serviços são referidos como "oráculos". Ao considerar o exemplo do testador, um oráculo poderia inspecionar os registros de morte para identificar se a pessoa faleceu. Em caso afirmativo, ele poderia gravar essas informações na blockchain (por exemplo: alterando o valor de uma variável booleana indicando se a pessoa está viva ou não). O contrato inteligente então acionaria uma instrução condicional (com base no valor da variável) e executaria o bloco de código que iniciaria a transferência de dinheiro (GATTESCHI; LAMBERTI; DEMARTINI, 2018).

Conforme Eyal (2017), para a indústria FinTech no que se refere a toda esta pilha de tecnologia como tecnologia de *ledger* distribuído, essas camadas representam duas oportunidades distintas: utilizando a segurança e a confiabilidade da infraestrutura que vem de baixo e implementando a funcionalidade de contrato inteligente. Para a FinTech, a capacidade do Bitcoin de facilitar transações seguras quando operado por alguns milhares de servidores voluntários é evidência de que talvez o mesmo possa ser feito para transações interbancárias ou bancárias para bancos, que são operadas usando servidores dedicados seguros. Na configuração existente da FinTech, essas transações levam um dia ou vários dias entre a emissão e a liquidação. Usar um blockchain para mediar transações bancárias para bancos poderia melhorar o desempenho de tal forma que um acordo poderia ser alcançado em uma fração desse tempo, comparável aos sistemas nacionais gerenciados centralmente (liquidação bruta em tempo real).

As empresas da FinTech também veem um tremendo potencial para a construção de contratos inteligentes que permitem às instituições financeiras implementar com segurança novos serviços, incluindo a estruturação de títulos a serem negociados e depois decompostos sem supervisão manual, no topo da caixa-preta blockchain. Como o *ledger* distribuído pode facilitar a transação de qualquer coisa que possa ser representada digitalmente, incluindo moeda fiduciária, títulos arbitrários e bens físicos, como ouro, as várias maneiras de usá-lo são uma área rica para exploração e pesquisa para muitas instituições financeiras centrais. Incluindo grandes bancos comerciais, bancos centrais, grandes empresas de contabilidade e gigantes da tecnologia (EYAL, 2017).

A tecnologia do *ledger* distribuído oferece benefícios potenciais para a FinTech em várias áreas importantes. Para uma variedade de casos de uso no mercado de capitais e corporativo, a tecnologia do *ledger* distribuído tem o potencial de reduzir o atraso de transações, o risco operacional, atrito de processos, requisitos de liquidez e muito mais. Além disso, aplicações interessantes surgem no contexto de seguros, monitoramento de cadeia e interação máquina-máquina relacionada à Internet das Coisas. Cada uma dessas áreas tem requisitos diferentes, incluindo política monetária, cumprimento regulatório e

vinculação de tokens digitais a objetos físicos ou outros externos ao blockchain (EYAL, 2017).

Diversos estudiosos dessa tecnologia já propuseram o seu uso em vários setores e contextos, por exemplo:

• No governo para registrar de forma transparente os votos dos cidadãos ou os programas dos políticos (para verificar se as promessas feitas foram cumpridas) ou para permitir sistemas de governação autônomos (HUCKLE; WHITE, 2016);

• Em propriedade intelectual: para atestar a prova de existência e autoria de um documento (GATTESCHI; LAMBERTI; DEMARTINI, 2018);

• Finanças: para transferir dinheiro entre as partes sem ter que depender dos bancos (GATTESCHI; LAMBERTI; DEMARTINI, 2018);

• Comércio: para registrar as características das mercadorias, bem como sua propriedade, especialmente para bens de luxo, reduzindo assim o mercado de itens falsificados ou roubados (GATTESCHI; LAMBERTI; DEMARTINI, 2018);

• Internet das coisas, por exemplo: explorando contratos inteligentes para processar automaticamente dados provenientes de sensores, para permitir que máquinas inteligentes interajam umas com as outras e executem ações autonomamente quando ocorrem situações específicas (HONG et al., 2017);

• Na educação: para armazenar as informações sobre as qualificações adquiridas pelos alunos. Por exemplo: para reduzir as fraudes na aplicação profissional. Nesse contexto vários interessados (universidades, instituições de treinamento, entre outras.) poderiam escrever as qualificações obtidas por uma pessoa na blockchain. A equipe de recursos humanos poderia, então, obter facilmente informações sobre quando e onde uma determinada competência foi obtida (SHARPLES; DOMINGUE, 2016).

2.15 As criptomoedas no Brasil

Muito se perguntam sobre como o Banco Central do Brasil (BACEN) está enxergando as criptomoedas (moedas virtuais) em território nacional. Através do seu comunicado de número 31.379 emitido em 16/11/2017, alertou-se sobre os riscos envolvendo as criptomoedas, decorrentes de operações de guarda e negociações. Nesse comunicado emitido em BACEN (2017) diz que:

"(...) estas não são emitidas nem garantidas por qualquer autoridade monetária, por isso não têm garantia de conversão para moedas soberanas, e tampouco são lastreadas em ativo real de qualquer espécie, ficando todo o risco com os detentores. Seu valor decorre exclusivamente da confiança conferida pelos indivíduos ao seu emissor."

"(...) se utilizadas em atividades ilícitas, podem expor seus detentores a investigações conduzidas pelas autoridades públicas visando apurar as responsabilidades penais e administrativas.

Em BACEN (2017), o BACEN procura esclarecer que as moedas virtuais não são o mesmo que as moedas eletrônicas que são previstas na legislação. Explicitando que as "moedas eletrônicas se caracterizam como recursos em reais mantidos em meio eletrônico que permitem ao usuário realizar pagamentos". No comunicado também deixa bem claro que ainda não há uma regulamentação para esse tipo de atividade no Brasil, esclarecendo que as empresas que negociam ou guardam as criptomoedas (conhecidas como Exchanges) não são reguladas, autorizadas ou supervisionadas pelo Banco Central do Brasil. E enfatiza que as operações de câmbio (operação de troca de moeda de um país pela moeda de outro país) com essas moedas não afasta da obrigatoriedade com a observação das normais cambiais vigentes. Deixando dessa forma o seu público ciente dos riscos de possíveis perdas patrimoniais, ocorridas através da típica variação de preços, decorrentes de fraudes ou outras condutas de negócio inadequadas. Sendo assim as partes envolvidas assumem todo o risco associado com a compra e venda de bens ou serviços, e as formas de pagamento.

2.16 Ethereum Versus Bitcoin

O Ethereum foi lançado em 2015 e é a maior plataforma de software descentralizada e aberta que permite a criação de contratos inteligentes e aplicações distribuídas (Dapps). Se o Bitcoin se destina a servir como moeda digital, o Ethereum representa uma plataforma descentralizada que executa contratos inteligentes. Bitcoin e Ether (ETH) são moedas digitais, mas o principal objetivo do Ether não é estabelecer-se como um sistema monetário alternativo (ao contrário do Bitcoin), e sim para facilitar e monetizar a operação da plataforma de contrato inteligente Ethereum e aplicações descentralizadas (Dapps) (LUU; CHU; OLICKEL, 2016).

Ethereum não é uma moeda, é uma plataforma. Possui sua própria moeda digital chamada Ether. Quando se trata de Bitcoin versus Ethereum, essa é uma das diferenças fundamentais. Porém essa não é a única diferença, outras também serão destacadas a seguir. Como por exemplo: o Ethereum tem um tempo de bloqueio bem mais rápido, ou seja, a quantidade de tempo necessário para validar e gerar um bloco é bem mais rápida. A transação do Ethereum é confirmada em segundos, enquanto a do Bitcoin leva alguns minutos. Bitcoin e Ethereum diferem, no entanto, em seu objetivo geral. Embora o Bitcoin tenha sido criado como uma alternativa às moedas nacionais e portanto seja um meio de troca e uma reserva de valor. O Ethereum foi concebido como uma plataforma para facilitar contratos e aplicações imutáveis e programáticas por meio de sua própria

moeda. Graças a esse conceito de contratos inteligentes, as transações Ethereum podem conter código executável, enquanto os dados nas transações Bitcoin são geralmente apenas para manter a rastreabilidade (LUU; CHU; OLICKEL, 2016).

2.17 Leitura e gravação de dados

Bhargavan, Delignat-Lavaud e Fournet (2016) mostram que a rede Ethereum faz uma distinção entre gravar dados na rede e ler dados dela, e essa distinção desempenha um papel importante na maneira como escrever a aplicação. Em geral, escrever dados é chamado de transação, enquanto ler dados é chamado de chamada. Transações e chamadas são tratadas de maneira muito diferente e têm as seguintes características.

As transações alteram fundamentalmente o estado da rede. Uma transação pode ser tão simples quanto enviar Ether para outra conta, ou tão complicada quanto executar uma função de contrato ou adicionar um novo contrato à rede. A característica definidora de uma transação é que ela grava (ou altera) dados. As transações custam a execução do Ether, conhecido como "gás", e as transações levam tempo para serem processadas. Quando se executa a função de um contrato por meio de uma transação, não pode receber o valor de retorno dessa função porque a transação não é processada imediatamente. Em geral, as funções destinadas a serem executadas por meio de uma transação não retornarão um valor, elas retornarão um ID de transação. Então em resumo as transações:

- Custam Ether (Gás)
- Alteram o estado da rede
- Não são processadas imediatamente
- Não expõe um valor de retorno (apenas um ID de transação).

As chamadas por outro lado são muito diferentes. As chamadas podem ser usadas para executar o código na rede, embora nenhum dado seja alterado permanentemente. As chamadas são gratuitas, e sua característica definidora é que elas lêem dados. Ao executar uma função de contrato por meio de uma chamada, receberá o valor de retorno imediatamente. Em resumo, as chamadas:

- São grátis (não custam gás)
- Não mudam o estado da rede
- São processadas imediatamente
- Exibirá um valor de retorno

Escolher entre uma transação e uma chamada é tão simples quanto decidir se deseja ler ou gravar dados.

2.18 Aplicações Descentralizadas (Dapps)

Segundo Cai, Wang e Ernst (2018), Dapps ou Decentralized Application, é oriundo da plataforma Ethereum, são as aplicações que utilizam contratos inteligentes para processamento sendo assim chamadas de "aplicações descentralizadas"ou "dapps". As interfaces do usuário para esses dapps consistem em linguagens familiares, como HTML, CSS e JavaScript. A aplicação em si pode ser hospedada em um servidor Web tradicional ou em um serviço de arquivos descentralizados, como Swarm ou IPFS.

Dados os benefícios da blockchain Ethereum, um dapp poderia ser uma solução para muitos setores, incluindo:

- Manutenção de registros
- Finanças
- Redes de fornecimento
- Imobiliária
- Marketplaces

2.19 Visão Geral do Truffle

Segundo Truffle Suite (2019b), o truffle é a melhor maneira de criar seu próprio dapp, testá-lo e implantá-lo em uma rede Ethereum. O Truffle é um ambiente de desenvolvimento de classe mundial, estrutura de teste e pipeline de ativos para blockchains usando a Ethereum Virtual Machine (EVM), com o objetivo de facilitar a vida de desenvolvedor. Com o Truffle obtém:

- Compilação inteligente de contratos incorporada, vinculação, implantação e gerenciamento binário.
- Teste de contrato automatizado para desenvolvimento rápido.
- Implantação extensível de script e estrutura de migração.
- Gerenciamento de rede para implantação em qualquer número de redes públicas e privadas.
- Gerenciamento de pacotes com EthPM e NPM, usando o padrão ERC190
- Console interativo para comunicação direta por contrato.
- Pipeline de construção configurável com suporte para uma integração estreita.
- Executor de script externo que executa scripts em um ambiente do truffle.

2.20 Usando o Truffle Develop e o Console

Truffle Suite (2019c) explica que as vezes, é bom trabalhar com os contratos interativamente para fins de teste e depuração ou para executar transações manualmente. O Truffle oferece duas maneiras fáceis de fazer isso por meio de um console interativo, com seus contratos disponíveis e prontos para uso.

• Truffle Console: um console interativo básico que se conecta a qualquer cliente Ethereum

• Truffle Develop: um console interativo que também gera uma blockchain de desenvolvimento

Ter dois consoles diferentes permite escolher a melhor ferramenta para suas necessidades.

Razões para usar o Truffle Develop :

• Está testando o projeto sem intenção de implantar imediatamente

• Não precisa trabalhar com contas específicas (e pode usar as contas de desenvolvimento padrão)

• Não deseja instalar e gerenciar um cliente blockchain separado

Razões para usar o Truffle Console:

• Ter um cliente que já está usando, como Ganache ou geth

• Desejar migrar para uma rede de teste (ou a rede principal do Ethereum)

• Desejar usar uma lista mnemônica ou de conta específica Ao longo desse trabalho foi optado por utilizar o Truffle Console. Todos os comandos exigem que esteja na pasta do projeto. Não precisando estar na raiz. Para iniciar o console:

$ truffle console

Ao carregar o console, verás imediatamente o seguinte prompt:

$ truffle(development)>

2.21 Aprofundando Testes em Solidity

Segundo Truffle Suite (2019d), os contratos de teste de solidity vivem ao lado dos testes de Javascript e tem a extensão de arquivo como .sol. Quando truffle executa os testes, eles serão incluídos como um conjunto de testes separado por contrato de teste. Esses contratos mantêm todos os benefícios dos testes de Javascript. Ou seja, um ambiente de sala limpa por suíte de testes, acesso direto aos contratos implantados e a capacidade de importar qualquer dependência de contrato.

Estrutura de teste: Para entender melhor o que está acontecendo, será mostrado a seguir com mais detalhes algumas particularidades importantes.

Asserções: As funções de asserção Assert.equal() são fornecidas pela biblioteca truffle/Assert.sol. Esta é a biblioteca de asserções padrão, no entanto, pode-se incluir a sua própria biblioteca de asserções, desde que a biblioteca se integre livremente ao executor de testes do Truffle acionando os eventos de asserção corretos. Pode-se encontrar todas as funções de asserção disponíveis em Assert.sol.

Nomes de contrato de teste: Todos os contratos de teste devem começar com Test uma letra maiúscula T. Isso distingue esse contrato, além dos auxiliares de teste e contratos de projeto (ou seja, os contratos sob teste), informando ao executor quais contratos representam os conjuntos de testes.

Nomes de funções de teste: Como os nomes dos contratos de teste, todas as funções de teste devem começar com test minúsculas. Cada função de teste é executada como uma única transação, na ordem em que aparece no arquivo de teste (como seus testes de Javascript). As funções de asserção retornam um booleano que representa o resultado da asserção que você pode usar para retornar cedo do teste para evitar erros de execução (como em erros que Ganache ou Truffle Develop exibirão).

2.22 Testando os Contratos

Estrutura: Conforme Truffle Suite (2019e), o Truffle vem de fábrica com uma estrutura de teste automatizada para facilitar o teste de seus contratos. Essa estrutura permite escrever testes simples e gerenciáveis de duas maneiras diferentes:

• Em Javascript e TypeScript, para exercitar os contratos do mundo exterior, assim como a aplicação.

• No Solidity, para exercitar os contratos em cenários avançados e simples.

Ambos os estilos de testes têm suas vantagens e desvantagens.

Localização: Todos os arquivos de teste devem estar localizados no ./testdiretório. Truffle só irá executar arquivos de teste com as seguintes extensões: .js, .ts, .es, .es6, e .jsx, e .sol. Todos os outros arquivos são ignorados.

Comando: Como já mencionado, para executar todos os testes, basta digitar:

$ truffle test

Como alternativa, pode-se especificar um caminho para um arquivo específico que deseja executar, por exemplo,

$ truffle test ./path/to/test/file.js

3 Metodologia

Segundo Mariano e Santos (2017), a Teoria do Enfoque Meta-Analítico é uma mistura de duas abordagens que são elas: revisão sistemática da literatura e bibliometria. A bibliometria é uma técnica quantitativa e estatística de medição de índice de produção do conhecimento científico, serve para mapear a ciência, as relações da ciência, saber quem está colaborando com quem e é por meio dela que se tem acesso aos conhecimentos mais recentes e de maiores impactos. A revisão sistemática é a pesquisa planejada por meio de ações que permitem diminuir o viés da pesquisa combinando os estudos mais relevantes, por isso, possui alta rigorosidade. A revisão bibliográfica (ou literatura) funciona como um elo de ligação entre o passado e o futuro, é um maneira de conhecer o passado para não recriar o que já foi feito.

Para a elaboração deste trabalho levou se em consideração a Síndrome de Chico Buarque apresentada no TEMAC. A Síndrome de Chico Buarque faz um paralelo com a música. Suponha que um determinado indivíduo queira fazer um estudo sobre MPB (Música Popular Brasileira), nesse trabalho ele cite e tenha como base grandes nomes do MPB, porém esqueça de um dos principais que foi Chico Buarque. Dessa maneira a pesquisa foi incompleta, já que não se percebeu a importância de um dos seus maiores autores dentro do tema. Então essa é a perspectiva do TEMAC, é garantir que os trabalhos mais importantes estarão presentes na pesquisa. Uma Revisão Bibliográfica por meio do enfoque meta analítico não irá dizer o que se tem que usar, e sim dizer o que não se deve faltar (MARIANO; SANTOS, 2017).

A implementação do Temac consistirá de 2 passos:

1. Preparação da pesquisa (múltiplas bases de dados)

Embora possa também utilizar apenas 1 base de dados, ficando assim a critério do pesquisador. Ela é marcada por 4 perguntas que o pesquisador deve-se responder:

• Qual o descritor ou palavra-chave da pesquisa?

Foi utilizado a string: Blockchain AND Bitcoin.

• Qual o campo espaço-tempo da pesquisa?

2008 (Que foi o ano de surgimento do Bitcoin) até 2019. Porém as pesquisas envolvendo esses critérios são recentes, e a pesquisa retornou resultados principalmente de 2015 até 2019

• Quais as bases de dados serão utilizadas?

Foi utilizado a base de dados Scopus (https://www.scopus.com/), por ela ser uma

referência, uma das melhores bases de dados e ter uma excelente cobertura. Ela indexa diversas outras fontes como por exemplo: IEEE Xplore Digital Library, Springer, ACM Digital Library, assim como a base de periódicos da Capes, que contém trabalhos científicos de diversas fontes (conferências e periódicos).

O seu principal objetivo é a pesquisa por autor e assunto. A Scopus tem como vantagens:

- Indexar mais de 18000 títulos de periódicos;

- Inclui títulos em Acesso Aberto, conferências, páginas web, patentes e livros;

- Possui a funcionalidade "more" que permite visualizar rapidamente os registos órfãos;

- Cobertura muito forte ao nível das revistas de ciência e tecnologia;

- Contém ferramentas úteis para identificação dos autores;

- Gera automaticamente o h-índex;

- Tem mais conteúdos europeus que a Web of Science, e inclui mais idiomas para além do Inglês - 60% de cobertura é de fora dos EUA;

• E quais áreas de conhecimento serão utilizadas?

Na pesquisa foram delimitadas as áreas de: Computer Science, Engineering, Mathematics, Economics, Econometrics and Finance

2. Apresentação e inter-relação dos dados

De acordo com aquilo que já foi mencionado o filtro de pesquisa no Scopus ficou da seguinte maneira:

Scopus

Document search: Blockchain AND Bitcoin

Access type: Open Access AND Other

Years: 2008 - 2019

Subject area:

Computer Science,

Engineering,

Mathematics,

Economics, Econometrics and Finance

Document type: Article

Results: 209

Para realizar um trabalho que contivesse um ótimo nível de relevância e não sofresse a síndrome de "Chico Buarque" apresentada pelo TEMAC, foram selecionados os artigos melhores reconhecidos no período, ou seja, aqueles aos quais tivessem as maiores quantidades de citações em outros trabalhos semelhantes.

Na literatura segundo os princípios bibliométricos dos filtros, quando se trata de uma pesquisa em que ela busca os documentos mais citados, ela atinge o princípio da Bibliometria chamada de Lei do Elitismo, Lei do 80/20 e citações. A Lei do elitismo estima o tamanho da elite de determinado conhecimento. As citações atribuem aos documentos importância à medida que são citados por outros autores e a Lei de 80/20 pode ser adaptada para encontrar os 20% dos documentos que equivalem a 80% das citações (MARIANO; SANTOS, 2017).

Dessa forma para obter os artigos mais relevantes, ou seja, com a maior quantidade de citações, foram realizados os seguintes passos. Após fazer o filtro corretamente e os document results serem exibidos. Em sort on foi selecionado a opção: Cited by (highest) dessa forma foi ordenado de maneira decrescente os artigos que contivessem as maiores quantidades de citações. Logo após foi marcada a caixinha do All (para selecionar assim todos os artigos) e depois foi clicado o botão Export. Na janela que irá abrir, foi selecionado CSV para o tipo do documento de exportação. E em Citation information, foi selecionado somente as opções Document title e Citation Count. Logo após ter sido realizado corretamente todos esses procedimentos, foi clicado o botão Export para ter acesso aos dados.

A tabela a seguir exibe o nome dos artigos com as suas respectivas quantidades de citações:

Tabela 1 – Os 40 Artigos mais citados na Scopus.

Título	**Citações**
Bitcoin and beyond: A technical survey on decentralized digital currencies	190
Blockchain: The state of the art and future trends	157
IoT security: Review blockchain solutions and open challenges	86
The IoT eletric business model: using blockchain technology for the internet of things	59
Difficulty control for blockchain-based consensus systems	53
FairAccess: A new blockchain-based access control framework for the internet of things	51
The technology and economic determinants of cryptocurrency exchange rates the case of bitcoin	45
Blockchain technologies the foreseeable impact on society and industry	38
Analyzing the bitcoin network the first four years	35
A Novel Blockchain-Based Product Ownership Management System (POMS) for Anti-Counterfeits in the Post Supply Chain	31
A survey on security and privacy issues of bitcoin	30
Majority is not enough bitcoin mining is vulnerable	28
Cryptocurrency value formation: An empirical study leading to a cost of production model for valuing bitcoin	28
Bitcoin blockchain dynamics the selfish-mine strategy in the presence of propagation delay	28
Blockchain technology in finance	20
The evolution of bitcoin hardware	20
EduCTX: a blockchain-based higher education credit platform	19
Banking on blockchain costs savings thanks to the blockchain technology	19
Blockchain Technology: Transforming Libertarian Cryptocurrency Dreams to Finance and Banking Realities	18
Data integrity protection method for microorganism sampling robots based on blockchain technology	18
Blockchain and Smart Contracts for Insurance: Is the Technology Mature Enough?	17
Blockchain world - Feeding the blockchain beast if bitcoin ever does go mainstream, the electricity needed to sustain it will be enormous	17
Digital enablement of blockchain evidence from hna group	15
Secure and anonymous decentralized bitcoin mixing	14
Beyond bitcoin the rise of blockchain world	13
To blockchain or not to blockchain that is the question	12
An Empirical Study on Modeling and Prediction of Bitcoin Prices With Bayesian Neural Networks Based on Blockchain Information	12
Digital blockchain networks appear to be following Metcalfe's law	12
Contract law 2.0 'smart' contracts as the beginning of the end of classic contract law	12
Scaling properties of extreme price fluctuations in Bitcoin markets	11
Cryptocurrency and business ethics	11
When mobile blockchain meets edge computing	11
Blockchains and the Boundaries of Self-Organized Economies: Predictions for the Future of Banking	11
A survey on anonymity and privacy in bitcoin-like digital cash systems	10
Secure Pub-Sub: Blockchain-Based Fair Payment With Reputation for Reliable Cyber Physical Systems	10
Beyond bitcoin using blockchain technology to provide assurance in the commercial world	10
Socialism and the blockchain	10
Goldstrike 1: CoinTerra's First-Generation Cryptocurrency Mining Processor for Bitcoin	10
Blockchain and Cryptocurrencies: Model, Techniques, and Applications	9

Foi definido um processo de estudo de revisão adaptado de Kitchenham e Charters (2007) e Petersen e Feldt (2008). Este processo consiste em três fases principais: planejar, executar e documentar a revisão, conforme apresentado na Figura 11:

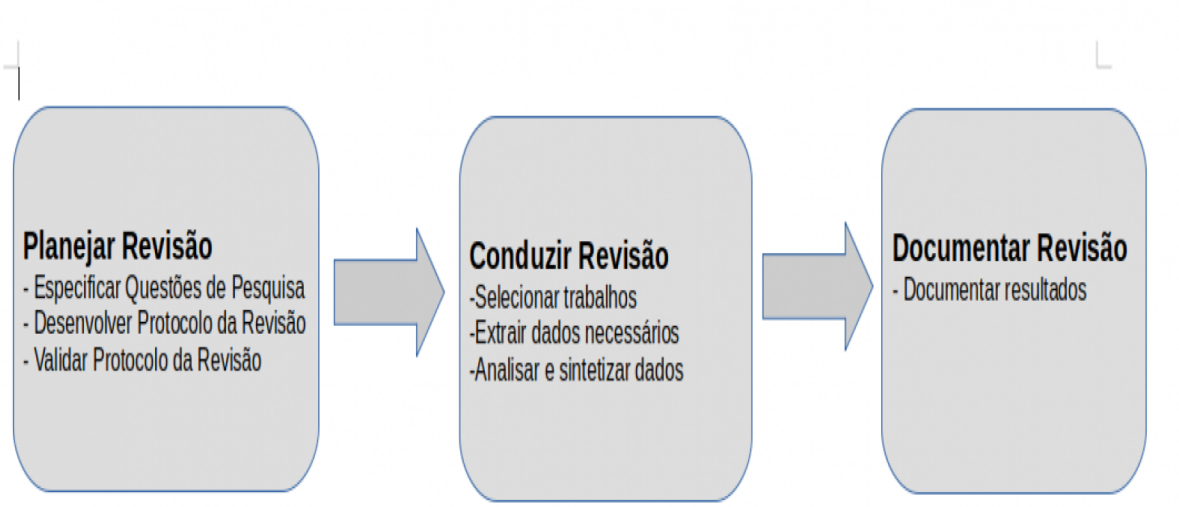

Figura 11 – Visão Geral do Processo de Revisão Bibliográfica. Fonte: Autor

A fase de planejamento da revisão bibliográfica está focada na especificação do protocolo, que descreve atividades para reunir evidências disponíveis. O protocolo é bem detalhado e serve para apoiar os pesquisadores envolvidos durante todo o processo de revisão. Ele fornece uma definição clara de questões de pesquisa, estratégia de busca para reunir estudos relevantes, critérios para inclusão e exclusão de estudos primários, rastreio de processos de documentos, bem como análise de extração de dados e síntese.

A segunda fase, Execução da revisão, envolve a aplicação do protocolo de pesquisa para buscar estudos, extrair dados e sintetizar o conhecimento relevante relacionado ao tema do trabalho. O resultado desta fase é a evidência gerada a partir de todas as atividades do protocolo.

A fase de documentação da revisão relata os resultados do estudo de mapeamento. Nesta fase foram consolidadas as informações, escrito o relatório e os resultados revisados e publicados.

Essa foi a primeira parte da pesquisa, abordando principalmente fatores referentes a parte teórica. Contudo a pesquisa se deu adiante buscando fontes voltadas a parte prática, a parte da implementação do software. E na Scopus prosseguiu a pesquisa repetindo as mesmas condições de ambiente já descrita, os mesmos filtros de cenário, utilizando de maneira isolada e 1 por vez as palavras chaves: Ethereum, Initial Coin Offering, ERC20, Smart Contract. Foram pegos os artigos que foram ordenados pela maior quantidade de citações, ou seja, os mais relevantes da área. Contudo tinham muita teoria e pouca im-

plementação prática. Foram úteis em relação a teoria, porem nem tanto em relação a codificação. Assim foi necessário a busca por outros bons locais de consulta e aprendizado. Dessa forma a consulta, aprendizado e desenvolvimento se deram através do Github oficial da Ethereum / EIPs, disponível para consulta em: ethereum (2019), o site com a documentação oficial do truffle: Truffle Docs (2019), o site com a documentação oficial da linguagem de programação solidity: Solidity Docs (2019) e o site oficial da ethereum, na parte destinada aos seus desenvolvedores Ethereum (2019).

4 Resultados

Os resultados alcançados serão descritos, conforme os objetivos específicos do estudo.

4.1 Caracterizar a tecnologia blockchain e como é implementada pelas linguagens de programação tradicionais

A tecnologia Blockchain e suas particularidades foram caracterizadas ao longo desse trabalho, porém ao se tratar de linguagens de programação tradicionais pode-se observar algumas características. Quando se trata da criação de uma Blockchain, várias linguagens de programação podem ser utilizadas. A seguir será exibido algumas linguagens tradicionais desse ambiente típico de desenvolvimento, e como elas costumam contribuir para o processo de implementação.

• JAVA: Segundo Liu (2018), Java é usado principalmente no design da aplicação, pois é fácil conectar o link entre os blocos de informações. Criar o relacionamento entre os dados e enviá-lo ao usuário é bastante simples com o uso da linguagem Java. Uma das razões pelas quais ele é cobiçado entre os programadores é o fato de que ele pode ser executado em quase todos os formatos de computadores com uma pequena instalação do JRE ou do ambiente de tempo de execução Java. Sendo assim a principal razão para usar o Java como a linguagem de programação blockchain de fato na indústria é pelo fato da sua portabilidade altamente capaz. Os programas escritos em Java são portáveis em qualquer dispositivo computacional, pois eles não dependem da arquitetura específica do sistema, em vez disso, usam a JVM universal (Java Virtual Machine) para execução. Isso faz do Java uma das melhores linguagens de programação para blockchain;

• C++: De acordo com Jindal, Aujla e Kumar (2019), C++ empacota dados e suas funções em objetos, que podem ser chamados e descompactados para uso em outros programas facilmente. Essa linguagem de programação blockchain permite o gerenciamento eficaz de recursos e oferece maior controle sobre a memória. Blockchain requer que muitos usuários e mineradores interajam e operem sistematicamente e simultaneamente. O C ++ cria aplicativos que podem não apenas coordenar entre vários terminais, mas também processar suas interações rapidamente. É por isso que projetos blockchain como Bitcoin, Ethereum e Ripple foram escritos em C ++.

• Python: Conforme Wang, Hsu e Hsiao (2018), Python é uma linguagem de programação simples e minimalista. Principalmente porque você pode executar muitas tarefas com um único comando nesse idioma. Isso faz com que o trabalho de construir o bloco

com as informações relevantes e conectá-las seja muito mais fácil de realizar. Embora as blockchains construídas em cima do Python tendem a ter um baixo desempenho durante operações criptográficas complexas devido à sua natureza interpretada, o Python oferece aos desenvolvedores a capacidade de prototipar suas ideias rapidamente. Além disso, o Python também vem com a capacidade de escrever programas em uma abordagem orientada a objetos, que pode ser utilizada para lidar com muitas de suas despesas gerais relacionadas ao desempenho.

• Solidity: Singla et al. (2019) relata que solidity é uma linguagem orientada a contratos, usada para escrever contratos inteligentes que pode ser implantados no EVM (Ethereum Virtual Machine). Ele segue uma abordagem orientada a objetos e suporta recursos como herança, tipos de dados complexos entre muitos. O EVM é o motor por trás de todo a blockchain Ethereum. Os contratos inteligentes são executados na Máquina Virtual Ethereum (EVM), o computador descentralizado, orientado por consenso, que distingue a Ethereum de Blockchains anteriores. Esta máquina virtual executa sua própria linguagem de bytecode. Por esse motivo, várias linguagens para contratos inteligentes foram desenvolvidas. Destes, a mais popular é a solidity. Solidity é uma linguagem estática que suporta herança, bibliotecas e tipos complexos definidos pelo usuário, entre outros. A sintaxe da solidity é semelhante ao JavaScript, mas se comporta de maneira um pouco diferente devido ao seu caso de uso.

4.2 Criar uma criptomoeda a partir da plataforma Ethereum

Foi construído um site através de uma ICO(Initial Coin Offering, ou seja, Oferta Inicial de Moedas) que conversará com um contrato inteligente de venda coletiva na blockchain Ethereum. Este site terá um campo disponível no qual os usuários podem comprar FGACoin em um procedimento chamado venda coletiva. O site exibirá uma barra de progresso na venda coletiva, exibirá quantas FGACoin o proprietário da conta possui, quantas FGACoin foram compradas por todos os usuários e o número total de FGACoin disponíveis para a venda coletiva. Também será mostrado na tela a conta a qual esta conectado à blockchain naquele momento através de um texto chamado: "sua conta", além de também ser possível enviar Ether entre contas do Ganache através do Metamask. Todo roteiro necessário para a replicação do experimento está disponível no apêndice ao final do trabalho.

4.3 Descrever a dinâmica envolvida na produção e distribuição das moedas digitais

Token ERC-20: A blockchain Ethereum permite que o desenvolvedor crie sua própria criptomoeda, ou token, que pode ser comprada com o Ether, a criptomoeda nativa da blockchain Ethereum. De acordo com Fenu e Marchesi (2018), o ERC-20 é simplesmente um padrão que especifica como esses tokens se comportam, para que sejam compatíveis com outras plataformas, como trocas de criptomoedas. Ethereum é uma blockchain como o Bitcoin. Como o Bitcoin, o Ethereum mantém o controle dos saldos das contas das pessoas que possuem o Ether, a criptomoeda nativa do Ethereum. Ao contrário do Bitcoin, o Ethereum também é uma plataforma que permite criar seu próprio token sem criar uma nova blockchain. Sendo assim pode-se criar um token Ethereum com um contrato inteligente. O ERC-20 é um padrão que especifica como esse contrato inteligente de token deve funcionar.

Exemplificando para assim compreender melhor como um contrato inteligente de token ERC-20 funciona. Suponha que se queira criar um token chamado "FGACoin"com o símbolo "FCN"e que existam 100.000.000 desses tokens. Primeiro, o contrato inteligente de token controla alguns atributos básicos do token. Em outras palavras, ele registra o nome "FGACoin", registra o símbolo em uma troca de criptomoedas e quantos tokens totais existem. Ele também controla quem possui "FGACoin"e quanto.

Os tokens do ERC-20 podem ser transferidos de uma conta para outra como pagamento, assim como qualquer outra criptomoeda. Eles também podem ser comprados em uma venda coletiva, como uma ICO (Initial Coin Offering), que também será explicada logo mais. E também podem ser comprados e vendidos em uma bolsa de criptomoedas. Para encontrar as funções, padrões de desenvolvimento e maiores informações sobre o token ERC-20, consulte o Github oficial: (ETHEREUM, 2019)

Venda coletiva (ICO): Para Adhami, Giudici e Martinazzi (2018) os tokens do ERC-20 podem ser distribuídos de várias maneiras. Um método popular é realizar uma venda em massa ou uma oferta inicial de moedas (ICO). As vendas em massa são uma maneira de uma empresa aumentar o capital de seus negócios criando seu próprio token ERC-20 que pode ser adquirido por investidores com Ether. Sempre que ocorre uma venda coletiva, a empresa obtém capital líquido na forma de Ether, pago pelos investidores, além de manter uma quantidade reservada dos tokens do ERC-20 que foram vendidos na venda coletiva. Para participar de uma venda coletiva, o investidor deve se conectar ao Blockchain Ethereum com uma conta. Essa conta tem um endereço de carteira que pode armazenar Ether, bem como os tokens ERC-20 que são comprados na venda coletiva. O investidor deve visitar um site de venda coletiva que fale com um contrato inteligente. O contrato inteligente rege todas as regras de funcionamento da venda pública.

Sempre que um investidor compra tokens no site de venda coletiva, ele envia o Ether da carteira para o contrato inteligente, e o contrato inteligente distribui instantaneamente os tokens comprados na carteira. O contrato inteligente define o preço do token na venda pública e controla o comportamento da venda pública. Também podem ter listas brancas para restringir quais investidores podem comprar tokens. Podem ter uma quantidade reservada de tokens que não são vendidos na venda em massa. Essas reservas geralmente são reservadas para membros específicos de cada empresa, como fundadores e consultores. Essas reservas podem ser uma quantidade fixa de tokens ou uma porcentagem. Sempre que uma venda coletiva termina, ela pode ser finalizada por um administrador. Sempre que isso acontecer, todos os tokens reservados serão distribuídos nas contas apropriadas e a venda pública será oficialmente encerrada. Algo que acontece no software FGACoin, em que 250000 FGACoin fica retida com o administrador e as outras 750000 FGACoin são comercializadas na ICO. E a venda coletiva quando encerrada, terá o status atual salvo pelo administrador.

4.4 Conhecer a dinâmica e importância dos Contratos Inteligentes

Contratos Inteligentes: Segundo Fenu e Marchesi (2018), os tokens ERC-20 são criados com contratos inteligentes da Ethereum. O Ethereum permite que os desenvolvedores escrevam aplicativos executados na blockchain com contratos inteligentes, que encapsulam toda a lógica de negócios dessas aplicações. Permitindo assim ler e gravar dados na blockchain, bem como executar códigos. No caso de um token ERC-20, o contrato inteligente governa todo o comportamento sobre como o token funciona e mantém o controle da propriedade do token e dos saldos da conta. O ERC-20 é uma especificação de API para como os tokens Ethereum devem ser construídos. É um padrão adotado pela comunidade que permite que os tokens sejam suportados em vários casos de uso. Para assim criar um token que seja compatível com esse padrão para que possa ser amplamente aceito. Se não existisse um padrão como esse, poderia existir diversas maneiras de criar tokens, e elas poderiam não serem compatíveis uma com a outra. (ETHEREUM, 2019)

O uso do padrão ERC-20 garante que um token seja compatível com os seguintes casos de uso:

- Transferências de carteira: enviando tokens de uma conta para outra
- Compra e venda em trocas de criptomoedas
- Compra de tokens em uma ICO (venda coletiva)

A especificação do ERC-20 essencialmente determina a interface à qual contrato inteligente deve responder. Ele especifica a estrutura do contrato inteligente e os tipos de funções que o contrato inteligente deve ter. Também fornece algumas funções sugeridas

que são boas de ter, mas que são opcionais. Determina certos eventos que o token devem ter, como um evento transfer. Contratos inteligentes podem emitir eventos nos quais os consumidores pode se inscrever e, com esse padrão, pode ser assinado eventos que informam quando os tokens são vendidos.

No Ganache a página Contracts contém uma lista dos contratos inteligentes por projeto. À primeira vista, pode-se ver o nome do contrato, endereço, contagem de transações e status de implantação.

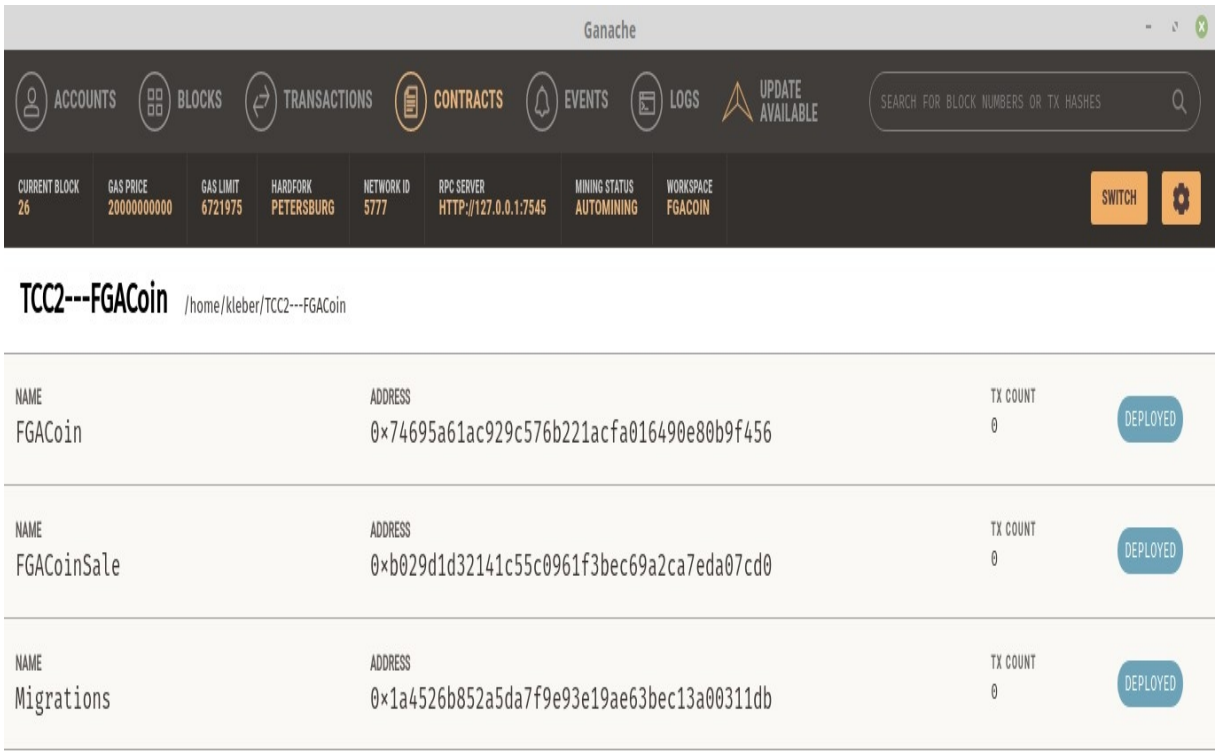

Figura 12 – Exemplo de Contracts no Ganache. Fonte: Autor.

Clicar em um dos contratos mostrará mais detalhes sobre esse contrato, incluindo sua transação de criação, armazenamento(estado), transações e eventos.

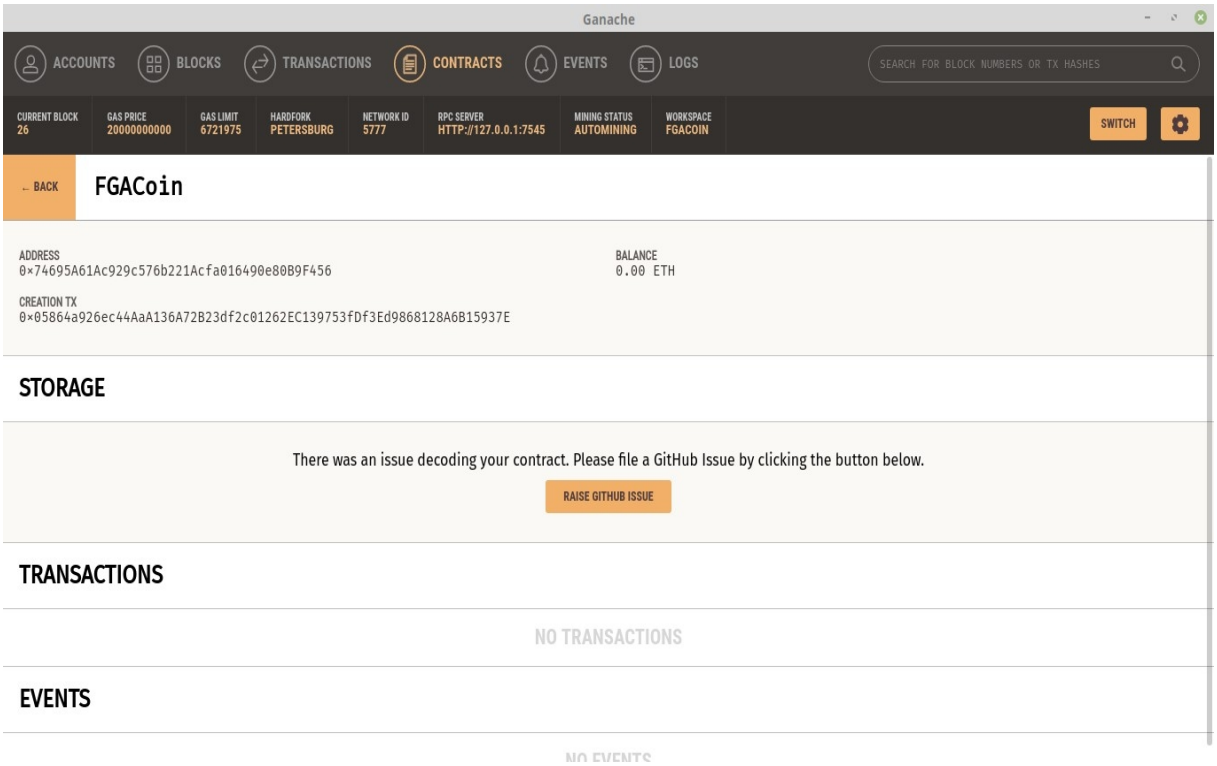

Figura 13 – Exemplo mais detalhado de Contracts no Ganache. Fonte: Autor.

5 Conclusão

Este trabalho de conclusão de curso forneceu um estudo conceitual amplo a respeito de Blockchain, explorando assim as suas origens no contexto de Bitcoin, porém tendo também esse aprofundamento em se tratando de Ethereum. Sendo abordado aspectos relevantes da tecnologia Blockchain, arquitetura e aplicações. E sendo exibido de uma maneira simples, de fácil entendimento e exemplificada passo a passo.

A Blockchain ganhou fama devido ao uso em Bitcoin, porém a Blockchain serve para muitas outras coisas além disso, como guardar algum tipo de informação tal como um contrato, certificado de propriedade, uma declaração de autenticidade ou comprovante de transação financeira de um banco entre muitas outras coisas.

Conforme relatado a Blockchain é uma maneira de armazenar e gravar transações, chega a ser muito parecido com um banco de dados tradicional, porém os blocos estão ligados criptograficamente entre si a fim de certificar que o sistema é a prova de adulteração. Como é um mecanismo muito difícil de ser fraudado, a Blockchain é fortemente pautada na integridade dos dados, na relação de confiança entre os seus adeptos.

Pode-se verificar que apesar de ter algumas vantagens em comparação com o Bitcoin, o Ethereum se difere principalmente no sentido de não ser só uma criptomoeda, e sim uma plataforma que fornece todo o suporte para o desenvolvedor criar a sua própria moeda digital. Fornecendo funções, padrões e dicas de implementação de forma a mostrar o vasto recurso que o Ethereum possui, o tornando assim tão atraente, com tantas qualidades que justificam o fato dessa tecnologia ano após ano estar se destacando e crescendo gradualmente.

E por fim através desse estudo é óbvio perceber que a Blockchain veio para ficar e se expandir cada vez mais em setores financeiros e não financeiros. Há relatos de pesquisadores que chegaram a afirmar que a Blockchain é depois do advento da internet, a maior invenção da computação. Por oferecer segurança, agilidade, transparência, serviços mais baratos, praticidade, automação entre diversas outras vantagens que torna a Blockchain uma tecnologia preciosíssima para ser explorada e usada a fundo.

Referências

ADHAMI, S.; GIUDICI, G.; MARTINAZZI, S. Why do businesses go crypto? an empirical analysis of initial coin offerings. 2018. Disponível em: <https://www-sciencedirect.ez54.periodicos.capes.gov.br/science/article/pii/S0148619517302308?via%3Dihub>. Citado na página 53.

ASTE, T.; TASCA, P.; MATTEO, T. D. Blockchain technologies: The foreseeable impact on society and industry. 2017. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8048633>. Citado 2 vezes nas páginas 17 e 24.

BACEN. A fistful of bitcoins: Characterizing payments among men with no names. 2017. Disponível em: <https://dl.acm.org/citation.cfm?id=2504747>. Citado 2 vezes nas páginas 38 e 39.

BACEN. *BACEN externo*. 2019. Disponível em: <https://www.bcb.gov.br/estatisticas/estatisticassetorexterno>. Acesso em: 02 dec. 2019. Citado na página 18.

BHARGAVAN, K.; DELIGNAT-LAVAUD, A.; FOURNET, C. Formal verification of smart contracts: Short paper. 2016. Disponível em: <https://dl-acm-org.ez54.periodicos.capes.gov.br/citation.cfm?doid=2993600.2993611>. Citado na página 40.

BITCOINMINING. Bitcoinmining: Bitcoin mining pools. 2014. Disponível em: <https://www.bitcoinmining.com/bitcoin-mining-pools/>. Citado na página 23.

BOUOIYOUR, J. What does bitcoin look like? 2015. Disponível em: <https://www.researchgate.net/profile/Refk_Selmi/publication/283676718_What_Does_Bitcoin_Look_Like/links/56432c8c08aef646e6c68bcc/What-Does-Bitcoin-Look-Like.pdf>. Citado na página 35.

BRITO, K. *TCC2—FGACoin*. 2019. Disponível em: <https://github.com/kleberbritomoreira10/TCC2---FGACoin>. Acesso em: 25 nov. 2019. Citado na página 68.

BROWNWORTH, A. *Demo do Blockchain*. 2019. Disponível em: <https://andersbrownworth.com/blockchain>. Acesso em: 25 nov. 2019. Citado na página 20.

BURGOS, A.; BATAVIA, B. Currency in the digital era. 2018. Disponível em: <https://www.bcb.gov.br/htms/public/inovtec/Currency-in-the-Digital-Era.pdf?5>. Citado na página 18.

CAI, W.; WANG, Z.; ERNST, J. B. Decentralized applications: The blockchain-empowered software system. 2018. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8466786>. Citado na página 41.

CIAIAN, P.; RAJCANIOVA, M. The economics of bitcoin price formation. 2015. Disponível em: <https://www.tandfonline.com/doi/full/10.1080/00036846.2015.1109038>. Citado na página 35.

CLACK, C. D.; BAKSHI, V. A.; BRAINE, L. Smart contract templates: Foundations design landscape and research directions. 2016. Disponível em: <https://arxiv.org/abs/1608.00771>. Citado na página 34.

ETHEREUM. *EIPs*. 2019. Disponível em: <https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md>. Acesso em: 02 dec. 2019. Citado 3 vezes nas páginas 50, 53 e 54.

ETHEREUM. *Ethereum*. 2019. Disponível em: <https://www.ethereum.org/developers/#getting-started>. Acesso em: 02 dec. 2019. Citado na página 50.

EYAL, I. Blockchain technology: Transforming libertarian cryptocurrency dreams to finance and banking realities. 2017. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8048646>. Citado 2 vezes nas páginas 37 e 38.

FENU, G.; MARCHESI, L. The ico phenomenon and its relationships with ethereum smart contract environment. 2018. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8327568>. Citado 2 vezes nas páginas 53 e 54.

FMI. Treatment of crypto assets in macroeconomic statistics. 2018. Disponível em: <https://www.imf.org/external/pubs/ft/bop/2019/pdf/Clarification0422.pdf>. Citado na página 18.

GARCIA, D. et al. The digital traces of bubbles: feedback cycles between socio-economic signals in the bitcoin economy. 2014. Disponível em: <https://royalsocietypublishing.org/doi/full/10.1098/rsif.2014.0623>. Citado na página 35.

GATTESCHI, V.; LAMBERTI, F.; DEMARTINI, C. Blockchain and smart contracts for insurance: Is the technology mature enough? 2018. Disponível em: <https://www.mdpi.com/1999-5903/10/2/20/htm>. Citado 4 vezes nas páginas 27, 36, 37 e 38.

GöBEL. Bitcoin blockchain dynamics: The selfish-mine strategy in the presence of propagation delay. 2016. Disponível em: <https://www-sciencedirect.ez54.periodicos.capes.gov.br/science/article/pii/S016653161630089X?via%3Dihub>. Citado 3 vezes nas páginas 28, 32 e 33.

HONG, Z. et al. Blockchain-empowered fair computational resource sharing system in the d2d network. 2017. Disponível em: <https://www.mdpi.com/1999-5903/9/4/85>. Citado na página 38.

HUCKLE, S.; WHITE, M. Socialism and the blockchain. 2016. Disponível em: <https://www.mdpi.com/1999-5903/8/4/49>. Citado na página 38.

HUGHES, L. et al. Blockchain research, practice and policy: Applications, benefits, limitations, emerging research themes and research agenda. 2019. Disponível em: <https://www-sciencedirect.ez54.periodicos.capes.gov.br/science/article/pii/S0268401219302014?via%3Dihub>. Citado na página 18.

JINDAL, A.; AUJLA, G. S.; KUMAR, N. Survivor: A blockchain based edge-as-a-service framework for secure energy trading in sdn-enabled vehicle-to-grid environment. 2019. Disponível em: <https://www-sciencedirect.ez54.periodicos.capes.gov.br/science/article/pii/S138912861831106X?via%3Dihub>. Citado na página 51.

KHAN, M. A.; SALAH, K. Iot security: Review, blockchain solutions, and open challenges. 2017. Disponível em: <https://www-sciencedirect.ez54.periodicos.capes.gov.br/science/article/pii/S0167739X17315765?via%3Dihub>. Citado 6 vezes nas páginas 9, 24, 25, 26, 28 e 34.

KITCHENHAM, B.; CHARTERS, S. Guidelines for performing systematic literature reviews in software engineering. 2007. Disponível em: <https://dl.acm.org/citation.cfm?id=2504747>. Citado 2 vezes nas páginas 19 e 49.

KRISTOUFEK, L. Bitcoin meets google trends and wikipedia: Quantifying the relationship between phenomena of the internet era. 2013. Disponível em: <https://www.nature.com/articles/srep03415?WT.ec_id=SREP-20131210>. Citado na página 35.

KRISTOUFEK, L. What are the main drivers of the bitcoin price? evidence from wavelet coherence analysis. 2015. Disponível em: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0123923&utm_source=rss&utm_medium=rss>. Citado na página 35.

LI, X.; ALEXWANG, C. The technology and economic determinants of cryptocurrency exchange rates: The case of bitcoin. 2016. Disponível em: <https://www-sciencedirect.ez54.periodicos.capes.gov.br/science/article/pii/S0167923616302111?via%3Dihub>. Citado 5 vezes nas páginas 17, 21, 22, 34 e 36.

LIANG, X. et al. Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability. 2017. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/7973733>. Citado na página 25.

LISCHKE, M.; FABIAN, B. Analyzing the bitcoin network: The first four years. 2016. Disponível em: <https://www.mdpi.com/1999-5903/8/1/7>. Citado 3 vezes nas páginas 9, 22 e 23.

LIU, X. A small java application for learning blockchain. 2018. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8614961>. Citado na página 51.

LUU, L.; CHU, D.-H.; OLICKEL, H. Making smart contracts smarter. 2016. Disponível em: <https://dl-acm-org.ez54.periodicos.capes.gov.br/citation.cfm?doid=2976749.2978309>. Citado 2 vezes nas páginas 39 e 40.

MARIANO, A. M.; SANTOS, M. R. Revisão da literatura: Apresentação de uma abordagem integradora. 2017. Disponível em: <https://www.researchgate.net/publication/319547360_Revisao_da_Literatura_Apresentacao_de_uma_Abordagem_Integradora>. Citado 2 vezes nas páginas 45 e 47.

METAMASK. *Metamask*. 2019. Disponível em: <https://metamask.io/>. Acesso em: 02 sep. 2019. Citado na página 68.

NODE.JS. *NODEJS. ORG*. 2019. Disponível em: <https://nodejs.org/en/download/current/>. Acesso em: 10 sep. 2019. Citado na página 67.

OBER, M.; KATZENBEISSER, S.; HAMACHER, K. Structure and anonymity of the bitcoin transaction graph. 2013. Disponível em: <https://www.mdpi.com/1999-5903/5/2/237>. Citado 3 vezes nas páginas 27, 32 e 34.

OUADDAH, A.; ELKALAM, A. A.; OUAHMAN, A. A. Fairaccess: a new blockchain-based access control framework for the internet of things. 2017. Disponível em: <https://onlinelibrary-wiley.ez54.periodicos.capes.gov.br/doi/full/10.1002/sec.1748>. Citado 2 vezes nas páginas 27 e 32.

PETERSEN, K.; FELDT, R. Systematic mapping studies in software engineering. 2008. Disponível em: <https://dl.acm.org/citation.cfm?id=2504747>. Citado 2 vezes nas páginas 19 e 49.

PITZ, R. Blockchain – a arquitetura disruptiva. 2017. Disponível em: <https://www.concrete.com.br/2017/08/25/blockchain-a-arquitetura-disruptiva/>. Citado 2 vezes nas páginas 9 e 27.

POLASIK, M. Price fluctuations and the use of bitcoin: An empirical inquiry. 2014. Disponível em: <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2516754>. Citado na página 35.

SHARPLES, M.; DOMINGUE, J. The blockchain and kudos: A distributed system for educational record, reputation and reward. 2016. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-319-45153-4_48>. Citado na página 38.

S.HAYES, A. Cryptocurrency value formation: An empirical study leading to a cost of production model for valuing bitcoin. 2016. Disponível em: <https://www-sciencedirect.ez54.periodicos.capes.gov.br/science/article/pii/S0736585315301118?via%3Dihub>. Citado 4 vezes nas páginas 31, 32, 35 e 36.

SINGLA, V. et al. Develop leave application using blockchain smart contract. 2019. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8711422>. Citado na página 52.

SOLIDITY DOCS. *Solidity*. 2019. Disponível em: <https://solidity.readthedocs.io/en/latest/>. Acesso em: 02 dec. 2019. Citado na página 50.

TAYLOR, M. B. The evolution of bitcoin hardware. 2017. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8048662>. Citado na página 28.

TRELEAVEN, P.; BROWN, R. G.; YANG, D. Blockchain technology in finance. 2017. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8048631>. Citado 2 vezes nas páginas 24 e 34.

TRUFFLE DOCS. *Truffle Docs*. 2019. Disponível em: <https://www.trufflesuite.com/docs>. Acesso em: 02 dec. 2019. Citado na página 50.

TRUFFLE SUITE. *Ganache*. 2019. Disponível em: <https://www.trufflesuite.com/ganache>. Acesso em: 12 sep. 2019. Citado na página 68.

TRUFFLE SUITE. *Truffle*. 2019. Disponível em: <https://www.trufflesuite.com/docs/truffle/overview>. Acesso em: 02 dec. 2019. Citado na página 41.

TRUFFLE SUITE. *Truffle*. 2019. Disponível em: <https://www.trufflesuite.com/docs/truffle/getting-started/using-truffle-develop-and-the-console>. Acesso em: 02 dec. 2019. Citado na página 42.

TRUFFLE SUITE. *Truffle*. 2019. Disponível em: <https://www.trufflesuite.com/docs/truffle/testing/writing-tests-in-solidity>. Acesso em: 02 dec. 2019. Citado na página 42.

TRUFFLE SUITE. *Truffle*. 2019. Disponível em: <https://www.trufflesuite.com/docs/truffle/testing/testing-your-contracts>. Acesso em: 02 dec. 2019. Citado na página 43.

TSCHORSCH, F.; SCHEUERMANN, B. Bitcoin and beyond: A technical survey on decentralized digital currencies. 2016. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/7423672>. Citado 6 vezes nas páginas 9, 22, 24, 26, 31 e 33.

WANG, S.-Y.; HSU, Y.-J.; HSIAO, S.-J. Integrating blockchain technology for data collection and analysis in wireless sensor networks with an innovative implementation. 2018. Disponível em: <https://ieeexplore-ieee-org.ez54.periodicos.capes.gov.br/document/8644936>. Citado na página 51.

ZHANG, Y.; WEN, J. The iot electric business model: Using blockchain technology for the internet of things. 2016. Disponível em: <https://link-springer-com.ez54.periodicos.capes.gov.br/article/10.1007%2Fs12083-016-0456-1>. Citado 4 vezes nas páginas 21, 33, 34 e 35.

Apêndices

APÊNDICE A – Roteiro para Replicação do ambiente de criação da FGACoin

A.1 Instalação FGACoin

Para criar a FGACoin é necessária a instalação de algumas dependências. A seguir serão mostradas quais são: (Observação: Esse software foi desenvolvido no sistema operacional Linux Mint 19 Cinnamon).

• Node Package Manager (NPM)

A primeira dependência necessária é o NPM, que vem junto com a instalação do Node.js. E para instalar o Node.js vá ao terminal (linha de comando) e digite:

$ brew install node

Ou vá diretamente no Website do Node.js, e faça o seu download e instalação em: NODE.JS (2019)

• Truffle Framework

A próxima dependência é o Truffle Framework, que permite criar aplicações descentralizadas na blockchain Ethereum. Ele fornece um conjunto de ferramentas que permite escrever contatos inteligentes com a linguagem de programação Solidity. O Truffle requer que seja um cliente Ethereum em execução que suporte a API padrão JSON RPC. Há muitos clientes por onde escolher e alguns melhores que outros para desenvolvimento. O truffle também permite testar os contratos inteligentes e implantá-los na blockchain.

Com o comando no terminal:

$ npm install -g truffle

Será obtida a versão mais recente do truffle framework, porém para garantir que o software irá funcionar perfeitamente, deve ser instalada a versão 4.0.4 do truffle que foi a mesma utilizada no desenvolvimento desse projeto. Tendo isso em vista para instalar o truffle framework com o auxílio do NPM, vá até o terminal (linha de comando) e digite:

$ npm install -g truffle@4.0.4 OU npm install truffle@4.0.4

• Ganache

A próxima dependência é o Ganache, em que o mesmo simula o comportamento de uma blockchain local na memória do computador. O Ganache é uma blockchain pessoal para desenvolvimento do Ethereum que roda em seu desktop. O Ganache simplifica

o desenvolvimento descentralizado, colocando seus contratos e transações na frente e no centro. Usando o Ganache, você pode ver rapidamente como seu aplicativo afeta a blockchain e examinar detalhes de como suas contas, saldos, criações de contratos e custos de gás. Também pode ajustar os controles avançados de mineração do Ganache para melhor atender às necessidades desejadas.

O Ganache está disponível para Windows, Mac e Linux. Pode-se instalar o ganache baixando diretamente no site: Truffle Suite (2019a). O Ganache disponibiliza 10 contas externas com endereços em nossa blockchain local Ethereum. Cada conta é pré-carregada com falsos 100 éter. O Ganache quando lançado é executado em http://127.0.0.1:7545. Ele exibirá as 10 primeiras contas e o mnemônico usado para criar essas contas. O mnemônico persistirá nas reinicializações do Ganache, embora possa ser alterado para ser gerado aleatoriamente. Esse mnemônico não deve ser utilizado na rede principal do Ethereum (Rede Ethereum Principal).

- Metamask

A próxima e última dependência é o Metamask. Metamask é a maneira mais fácil de interagir com aplicativos descentralizados em um navegador. É uma extensão, disponível atualmente para o Chrome, Firefox, Opera, Brave, IOS e Android, que se conecta a uma rede Ethereum sem executar um nó completo na máquina do navegador. Ele pode se conectar à rede principal do Ethereum e a qualquer uma das redes de teste (Ropsten, Kovan e Rinkeby) ou a uma blockchain local como a criada por Ganache ou Truffle Develop.

Para o desenvolvimento com o Truffle, isso significa que pode ser usado a aplicação descentralizada da mesma maneira que os usuários irão interagir com ela em uma rede ao vivo. Para usar a blockchain, é preciso se conectar a ela. Assim terá que instalar uma extensão especial do navegador para usar a Blockchain Ethereum. É aí que entra o Metamask. Podendo assim se conectar ao blockchain Ethereum local com a conta pessoal e interagir com o contrato inteligente.

Dessa forma essa extensão poderá ser encontrada em: Metamask (2019)

A.2 Utilização

O projeto FGACoin encontrasse no Github, disponível em: Brito (2019)

Ao rodar o projeto pela primeira vez, deve se instalar as dependências vinculadas ao npm. Dessa forma navegue até a pasta do projeto pelo terminal e digite:

$ npm install

Logo após aperte a tecla Enter e irá iniciar o processo de instalação.

• Testes Referência

Para assegurar o desenvolvimento correto do sistema, durante a elaboração do Back End foram criados testes unitários e alocados na pasta chamada test. E para rodar os testes e verificar se estão passando, vá até a pasta do projeto no terminal e digite:

$ truffle test

Porém caso não esteja com o ganache aberto e um workspace em execução, não será reconhecido como cliente ethereum e será exibido na tela as seguintes mensagens de erro:

Could not connect to your Ethereum client. Please check that your Ethereum client:

- is runningReferencia

- is accepting RPC connections (i.e., --rpc"option is used in geth)

- is accessible over the network

- is properly configured in your Truffle configuration file (truffle.js)

• Servidor

Para iniciar o servidor digite no terminal:

$ npm run dev

Esse comando rodará um script definido no arquivo package.json, carregando assim informações básicas para a implantação do servidor. Nele está descrito o servidor utilizado (lite-server), versão de desenvolvimento, dependências, as portas de redes utilizadas entre outros.

Ao iniciar o servidor, abrirá uma janela de navegação no google chrome com o nome Carregando... O sistema ficará assim pois ainda não sabe quais informações devem ser carregadas e exibidas.

• Configurando o MetaMask

Para usar o Ganache com o MetaMask, clique no ícone MetaMask no seu navegador e esta tela aparecerá:

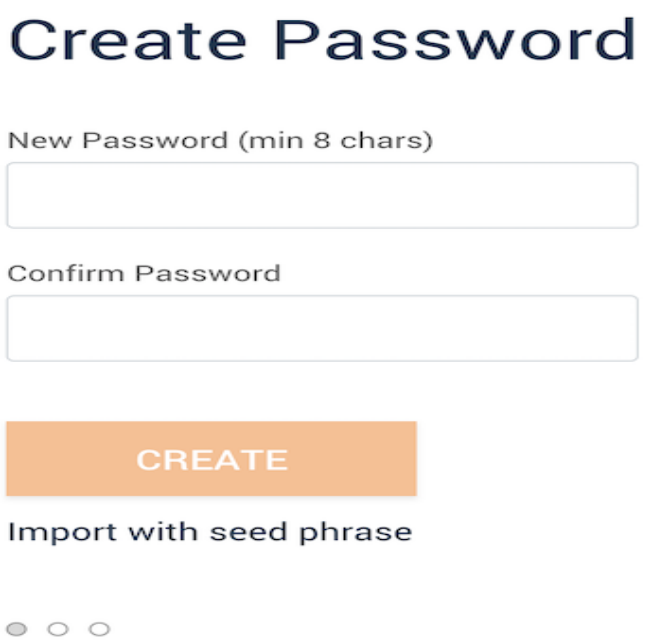

Figura 14 – Criando uma conta no metamask. Fonte: Autor

Clique em Import with seed phrase. Na caixa marcada Wallet Seed, digite o mnemônico exibido ao iniciar o Ganache. Não use este mnemônico na rede principal do Ethereum (rede principal). Certifique-se de definir a rede como "Rede privada"(use a configuração "RPC personalizado"). Veja a seguir para mais detalhes. Digite uma senha abaixo disso e clique em OK.

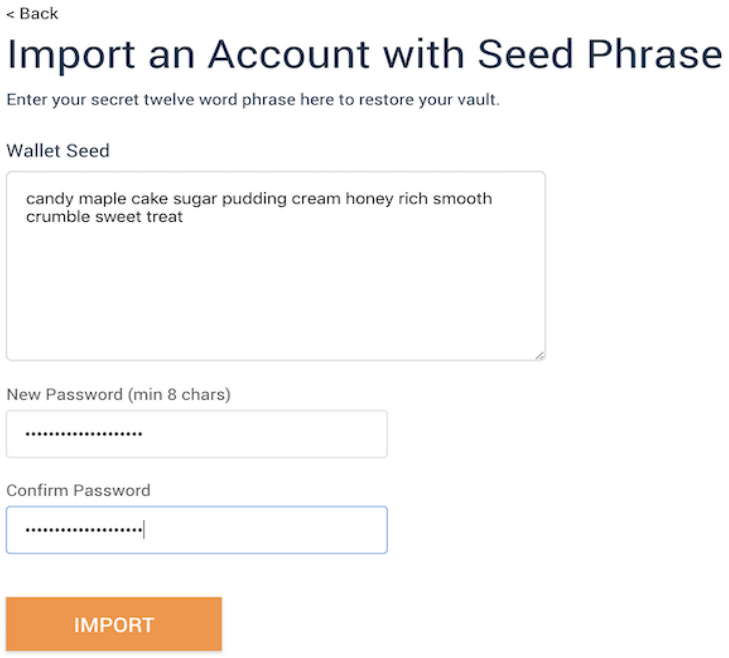

Figura 15 – Importando uma conta com mnemônico. Fonte: Autor

O próximo passo é conectar o ganache com o metamask e assim poder exibir as informações referentes a "carteira" de criptomoedas. Configure assim a sua própria rede no metamask para reconhecer a sua blockchain local e poder executar transações. Comece customizando uma rede RPC, clicando na dropdown das redes e logo em seguida em Customizar RPC

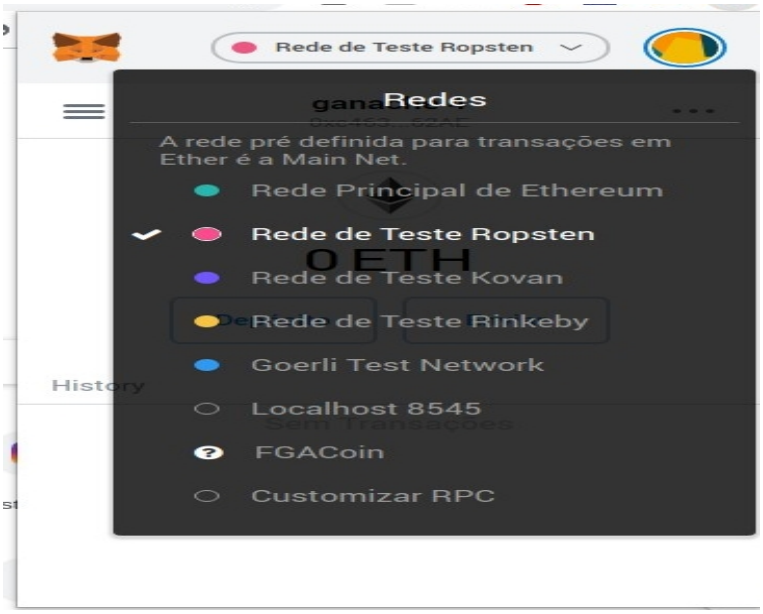

Figura 16 – Selecionando opção Customizar RPC. Fonte: Autor

Após essa etapa, copie o RPC SERVER do ganache (HTTP://127.0.0.1:7545) e cole em New RPC URL do metamask, escolha um Network Name e clique em guardar

Figura 17 – Customizando rede RPC. Fonte: Autor

Feito isso, a página inicial do FGACoin irá carregar, porém com o preço atual da criptomoeda e a quantidade do proprietário zerados. Isso se deve ao fato de não ter sido executada às migrações e não ter conta carregada do Ganache no Metamask.

Para importar as contas será necessário importar as chaves privadas. Vá então no Ganache, clique em Show Keys no address de index 0

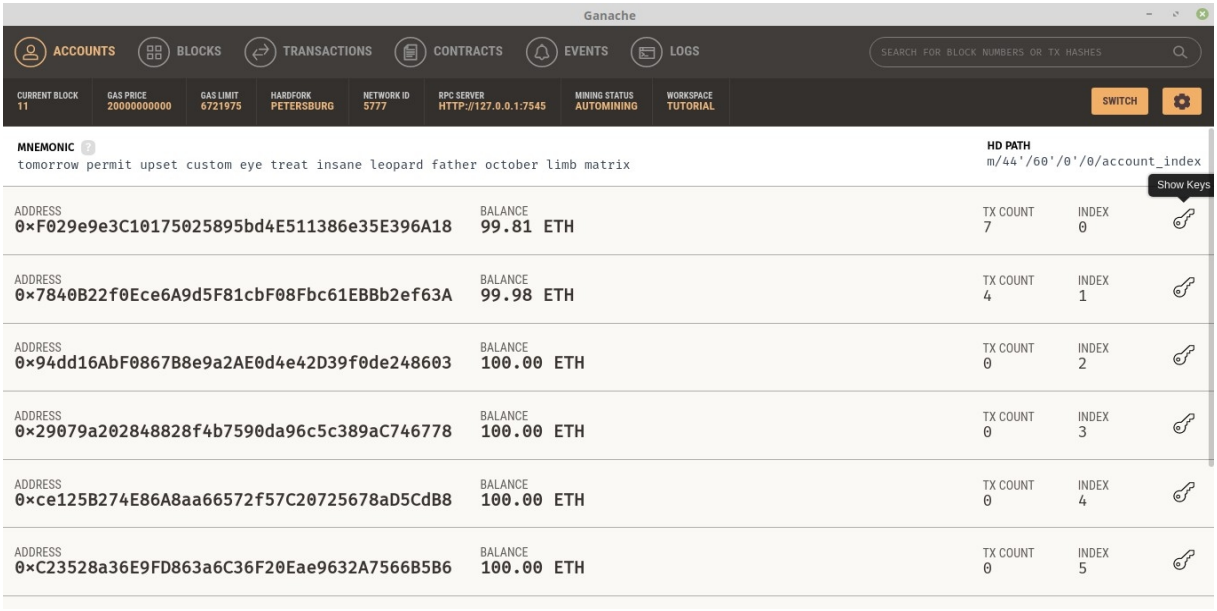

Figura 18 – Clicando Show keys no Ganache. Fonte: Autor

E copie a private key

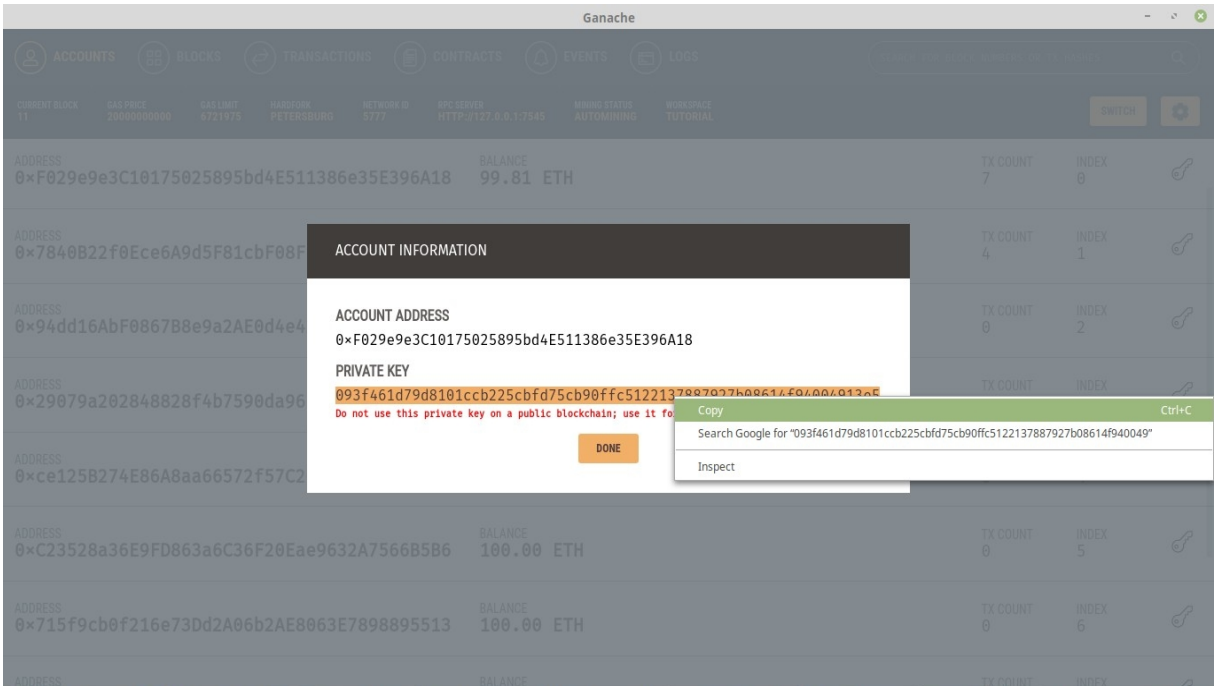

Figura 19 – Copiando chave privada do Ganache. Fonte: Autor

Agora clique no círculo e logo após em Importar Conta

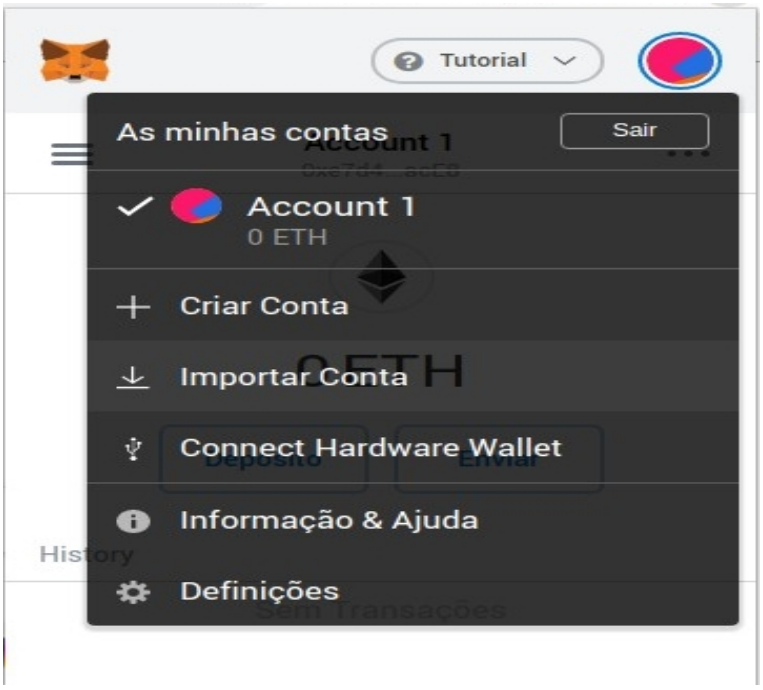

Figura 20 – Clicando em Importar Conta no metamask. Fonte: Autor

Cole a private key e clique em Importar.

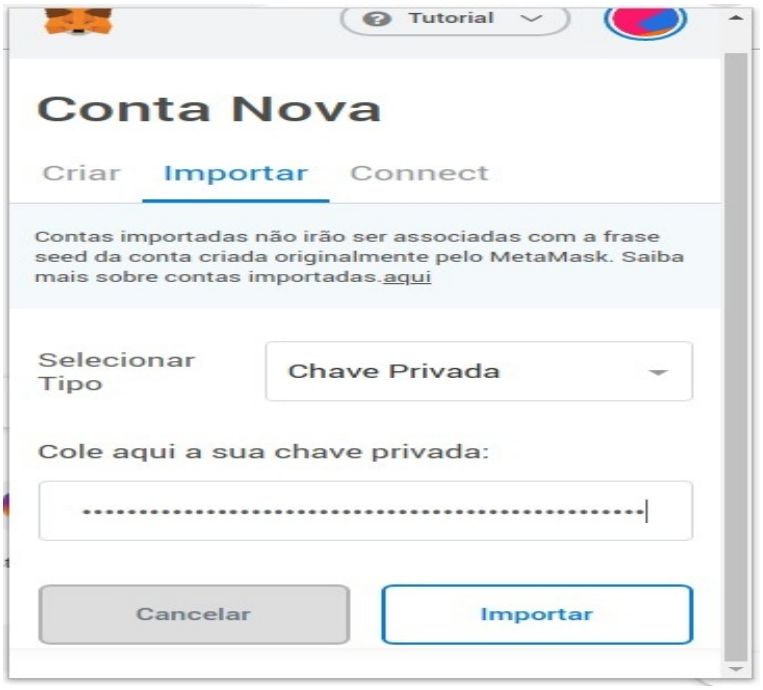

Figura 21 – Colando chave privada no metamask. Fonte: Autor

Repita esse mesmo processo e crie quantas outras contas desejar. Para facilitar a didática da explicação, caso deseje pode-se renomear o nome atribuído a conta. Clique em Menu e logo em seguida no nome da Conta

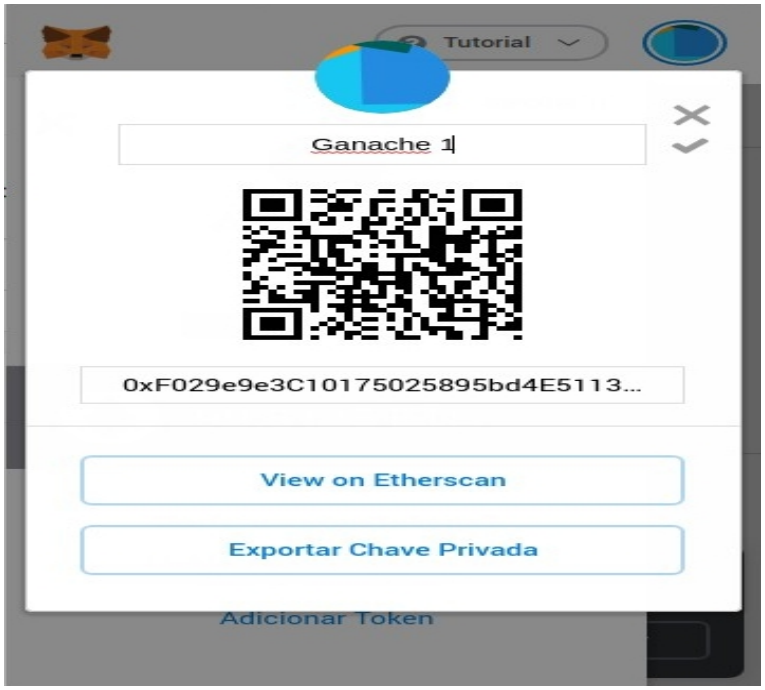

Figura 22 – Renomeando nome da conta no metamask. Fonte: Autor

Nesse exemplo foram importadas 4 contas e renomeadas com Ganache e a sua respectiva numeração

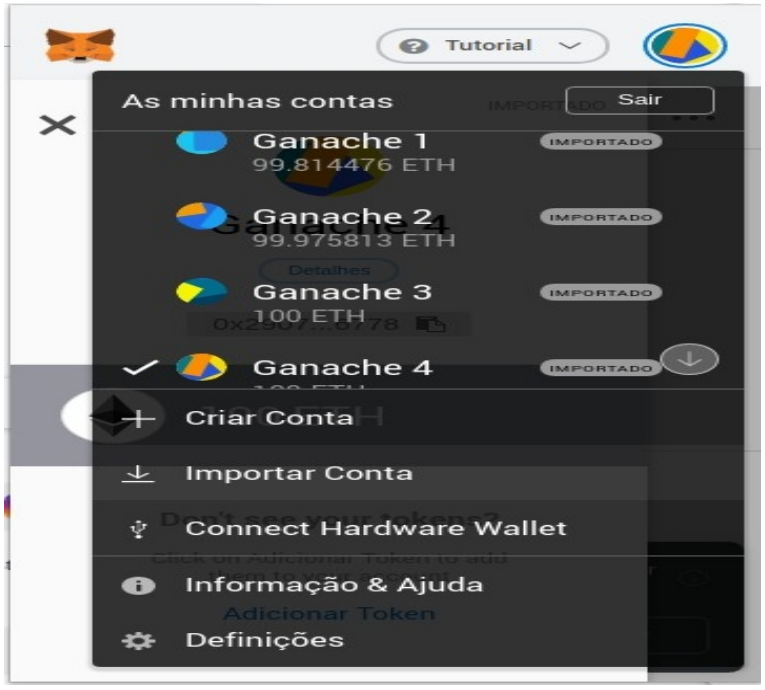

Figura 23 – Exibindo nome das contas renomeadas no metamask. Fonte: Autor

Com as contas desejadas importadas volte ao terminal. Nesse projeto foi designado o valor total de 1000000 criptomoedas (FGACoin). Sendo que 750000 estarão à disposição para venda e as outras 250000 restantes estarão com o administrador. O administrador do

sistema será a conta de index 0, ou seja, a primeira conta do ganache que foi importada no metamask. Essa configuração final será feita no terminal, mas primeiramente será necessário fazer a migração. Digite no terminal:

$ truffle migrate

Com esse comando será executado os arquivos de migração (Migration.sol, 1_initial_migration.js e 2_deploy_contracts.js). A migração é responsável pelo preparo das tarefas de implantação, ou seja, ativa o construtor presente no contratos, inicializando assim características básicas do sistema (como quantidade total de criptomoedas, preço para venda entre outros) além de implantar os smart contracts na rede Ethereum. Com isso ao atualizar a página, pode-se verificar na tela o endereço da sua conta do ganache, além do preço atual da criptomoeda como 0.001 Ether. E a quantidade total de 1000000 FGACoin.

Conforme já mencionado, essa quantidade de FGACoin deverá ser ajustada. E isso é feito através do console do truffle. Digite no terminal:

$ truffle console

Digite na sequência esses comandos:

$ FGACoinSale.deployed().then(function(i) tokenSale=i;)

$ FGACoin.deployed().then(function(i) token=i;)

$ tokensAvailable = 750000;

$ admin = web3.eth.accounts[0]

$ token.transfer(tokenSale.address, tokensAvailable, from: admin)

$ .exit

Ao atualizar a página, poderá ser verificado que os valores estão devidamente ajustados. Com 250000 FGACoin para o administrador, 750000 FGACoin disponíveis para venda e preço da criptomoeda FGACoin = 0.001 * Quantidade de Ether. Clicando no círculo é possível alternar entre as contas. Por exemplo, clicando na conta Ganache 4 e atualizando a página é possível verificar essa permuta entre as contas.

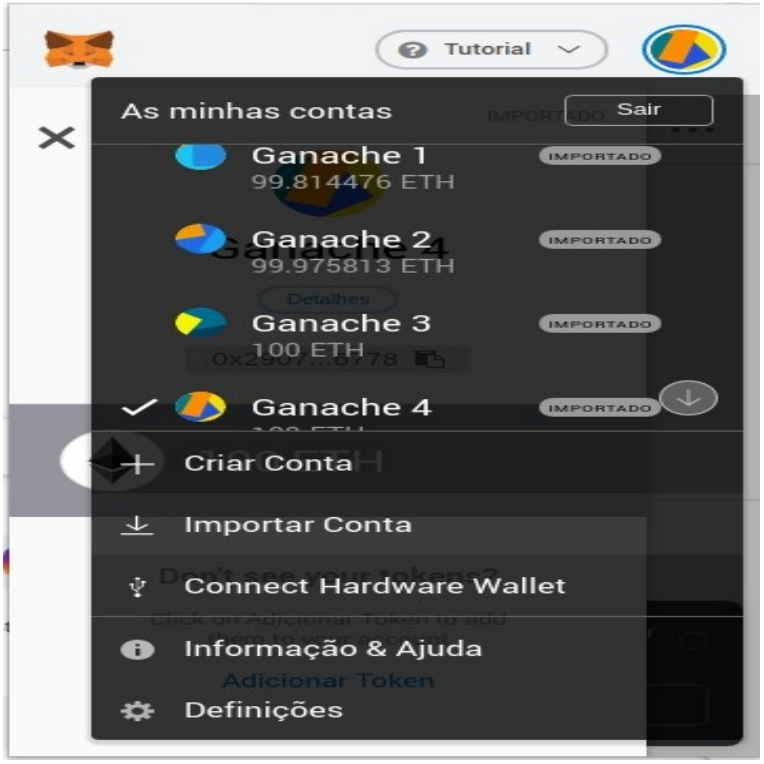

Figura 24 – Alternando entre contas no metamask. Fonte: Autor

Ao atualizar a página irá mostrar que você tem atualmente 0 FGACoin. Para comprar FGACoin, digite o valor desejado e clique em Comprar FGACoin. Uma janela de notificação do metamask irá abrir e perguntar se você deseja Confirmar ou Rejeitar a transação. Nessa janela também é exibida o gas fee (taxa do gás), ou seja, a taxa que você deverá pagar da transação. Sempre que você interage com a blockchain Ethereum, você está lendo dados ou gravando dados nele. A leitura de dados do Ethereum é gratuita, mas a gravação custa caro. Esse dinheiro é chamado de "gás"e é pago em Ether pela conta que inicia a transação.

Como o Ethereum é uma rede de nós ponto a ponto, não um sistema centralizado, as pessoas que executam os nós são incentivadas por pagamentos a participar da rede. Esses nós que participam da gravação de dados no blockchain Ethereum são chamados de "mineradores". Eles são pagos para gravar transações porque incorrem em custos para oferecer seus recursos computacionais à rede e são compensados pelos usuários que pagam para enviar transações. É por isso que você deve pagar para criar transações.

Ao confirmar a transação e atualizar a página poderá ser verificado na tela a quantidade de FGACoin adquirida e indo no metamask ou ganache, a quantidade de Ether subtraído. E na tela da FGACoin também é exibida a quantidade de criptomoedas vendidas em relação ao total, e uma barra de carregamento que irá prosseguir a medida que a quantidade equivalente de criptomoedas forem sendo vendidas.

Após todas essas etapas, será possível verificar a blockchain indo no Ganache e

clicando em Blocks. Será exibida uma tela contendo todos os blocos daquela Blockchain e com as devidas particularidades de cada transação. Conforme pode ser observado (a critério de ilustração) nas imagens a seguir:

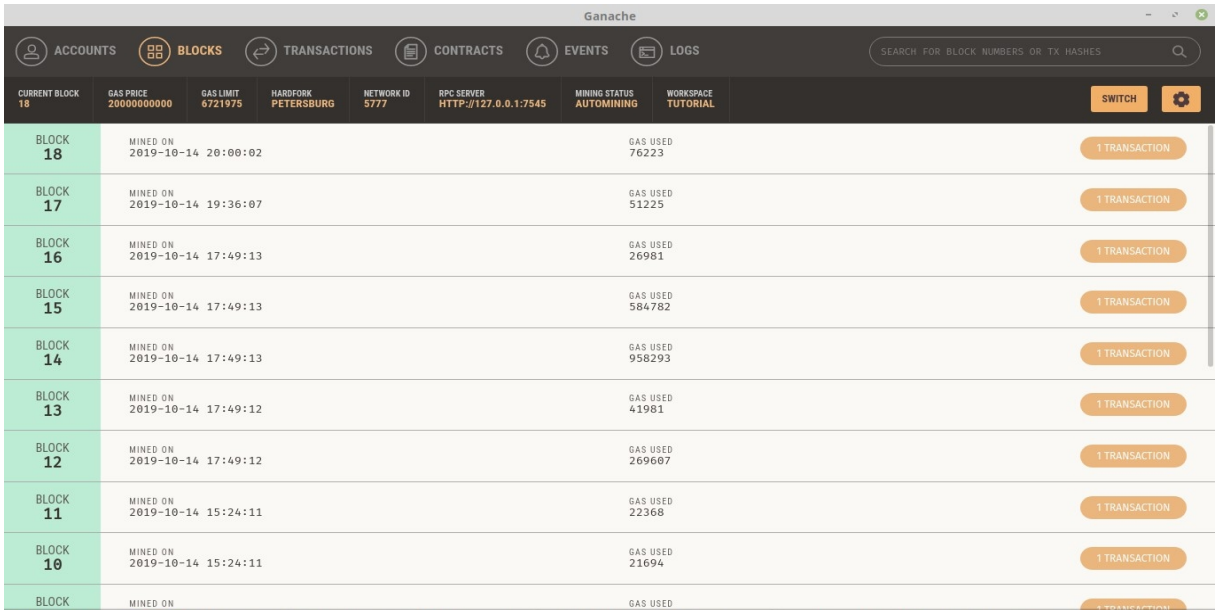

Figura 25 – Blockchain no Ganache. Fonte: Autor

Clicando em Transaction poderá ser observado as informações específicas do bloco em questão

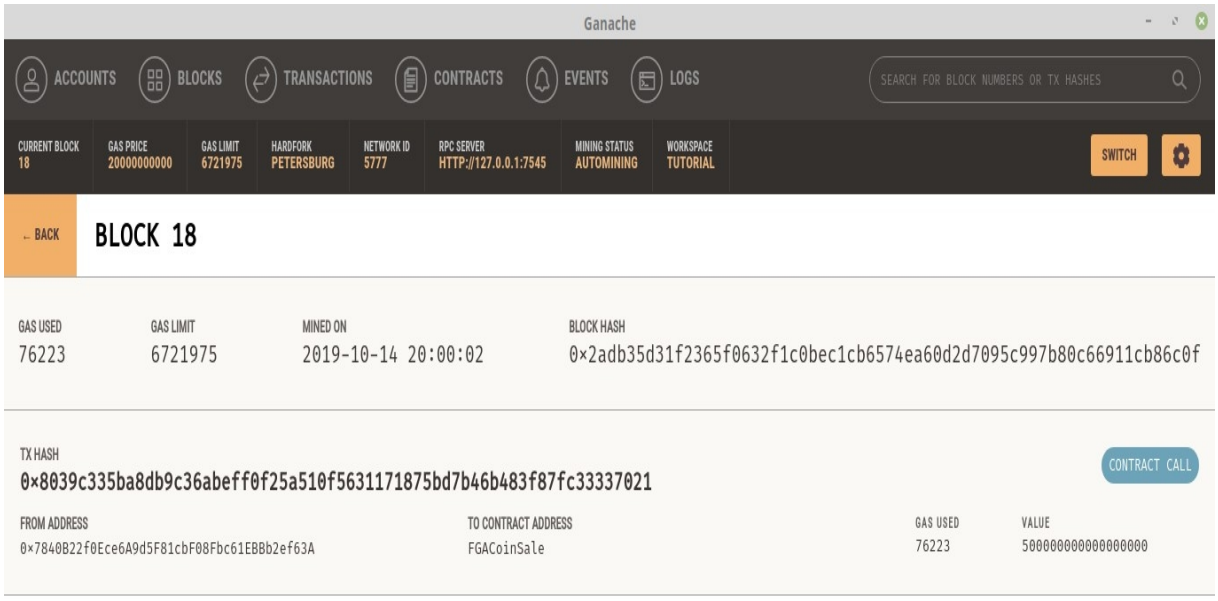

Figura 26 – Especificando a transação de um bloco. Fonte: Autor

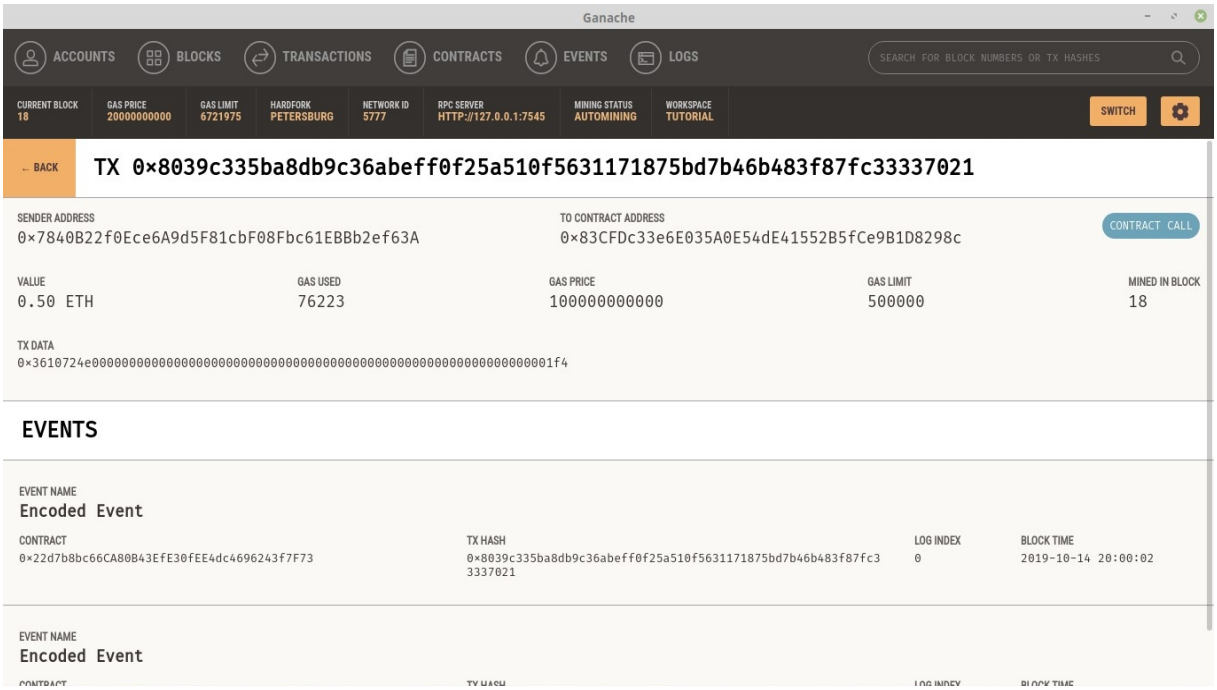

Figura 27 – Especificando valores do Contrato. Fonte: Autor

A.3 Enviando Ether

Para enviar a criptomoeda Ether para outras contas, estando na página inicial clique em Enviar

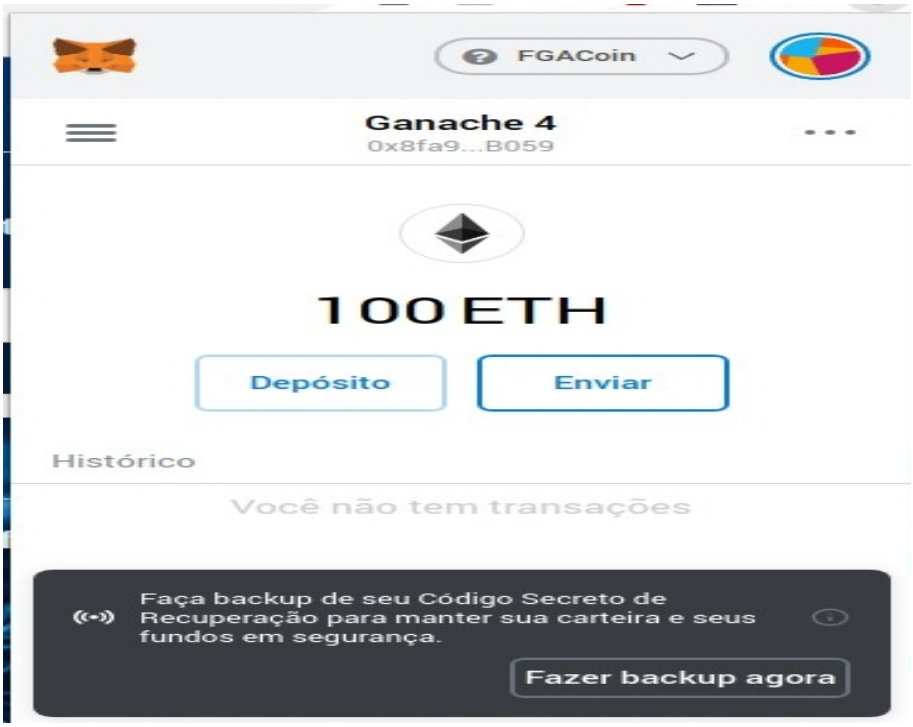

Figura 28 – Clicando em Enviar Ether. Fonte: Autor

Depois adicione o destinatário, pode ser colando o endereço da conta, via QRCode

ou selecionando a opção Transferir entre minhas contas. Clicando em Transferir entre minhas contas, será exibida as suas opções de contas registradas no Metamask.

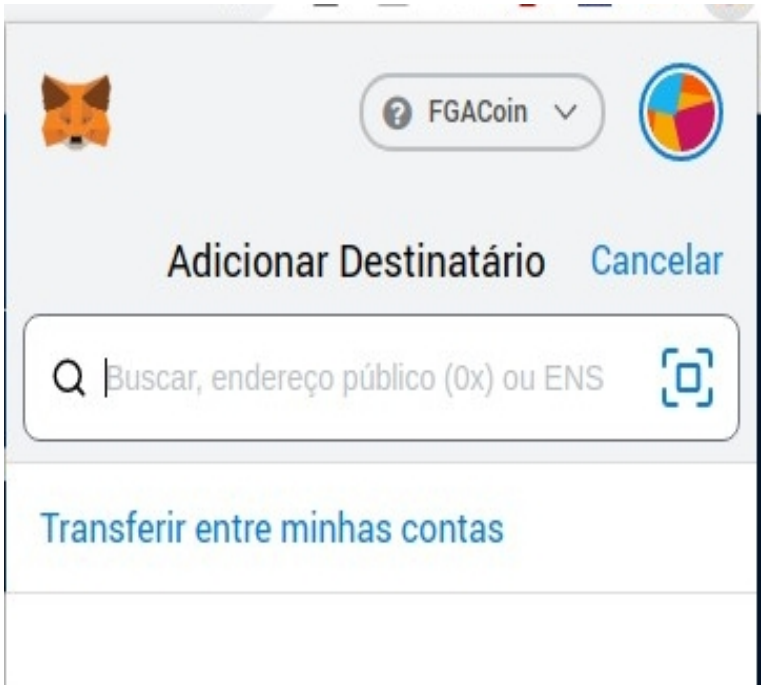

Figura 29 – Adicionando endereço da conta do destinatário. Fonte: Autor

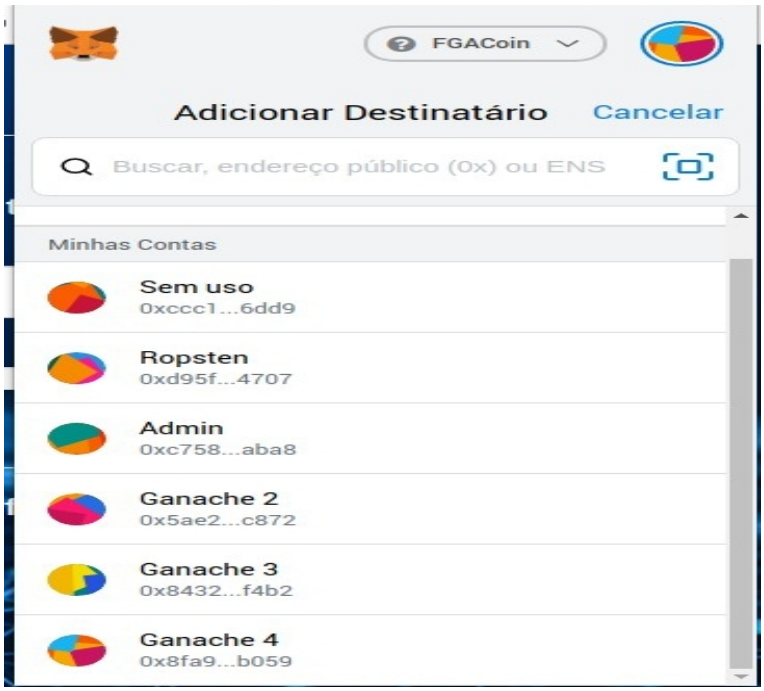

Figura 30 – Selecionando conta de destino. Fonte: Autor

Digite o valor desejado para a transferência, ajuste a taxa da transação caso ache necessário e clique em Próxima

Figura 31 – Definindo o valor da transação. Fonte: Autor

Caso esteja tudo de acordo, clique em Confirmar para a transferência ser efetuada.

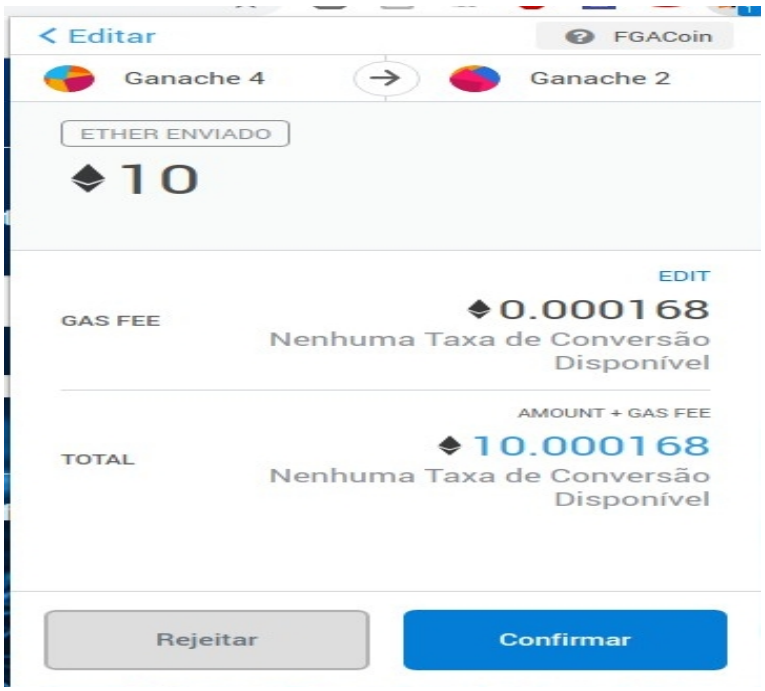

Figura 32 – Confirmando transferência. Fonte: Autor

A.4 Problemas Encontrados

O software FGACoin foi elaborado com o propósito de que o código lê-se as contas presentes no Ganache e exibisse as mesmas no Metamask. Caso esse processo não já tenha sido feito automaticamente durante a instalação, devera ser feito manualmente pelo desenvolvedor. Dessa forma para que o Metamask tivesse acesso a essas informações do Ganache, deve-se primeiro clicar no círculo ao lado da rede FGACoin e ir em configurações do Metamask:

Figura 33 – Clicando em Configurações. Fonte: Autor

Depois clique em connections:

Figura 34 – Clicando em Connections. Fonte: Autor

e manualmente adicionar o site na lista de permissões. Clicando em Conectar-se. Para assim permitir o site ter acesso aos endereços do Ganache.

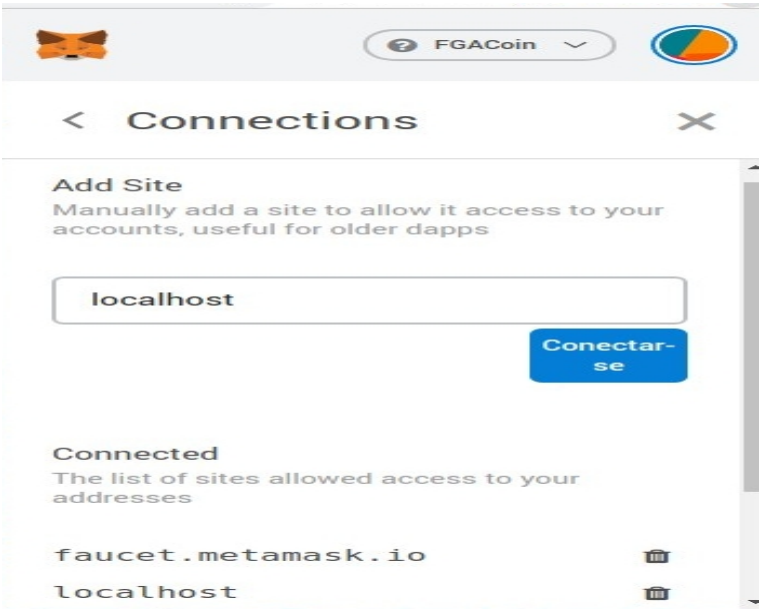

Figura 35 – Clicando em Conectar-se. Fonte: Autor

Durante o processo de desenvolvimento essa etapa não tinha sido realizada. E ao executar o servidor no terminal com:

$ npm run dev

A tela ficava sempre Carregando... e não exibia as informações, conforme mostrado na imagem a seguir:

Figura 36 – Tela principal apenas Carregando. Fonte: Autor

Vários códigos foram feitos e refeitos, tendo como intuito corrigir esse problema de loading, e ao consultar pelo console do javascript no navegador (Fn + F12), foi verificado que as informações referentes a conta do ganache sempre estava chegando como null(nulas, vazias). E somente após habilitar o Metamask, conforme ensinado anteriormente, foi que solucionou esse problema.

A.5 Redes de teste públicas

Os desenvolvedores geralmente usam redes de teste públicas (ou redes de teste) para testar aplicações Ethereum antes da implantação final na rede principal. O Ether nessas redes é usado apenas para fins de teste e não tem valor. Existem três redes de teste públicas em amplo uso:

• Ropsten: A rede de testes oficial, criada pela Fundação Ethereum. Sua funcionalidade é semelhante à MainNet.

• Kovan: uma rede que usa um método de consenso chamado "prova de autoridade". Isso significa que as transações são validadas por membros selecionados, levando a um tempo de bloqueio consistente de quatro segundos. O fornecimento de Ether neste testnet também é controlado para mitigar ataques de spam.

• Rinkeby: uma rede de teste também usando prova de autoridade, criada pela Ethereum Foundation.

Porém durante a elaboração desse trabalho, várias tentativas foram feitos de implantação da FGACoin nessas 3 redes de testes públicas, porém em nenhuma delas foram obtidos êxitos. E fica assim dessa forma (caso o desenvolvedor desejar) como sugestão para trabalhos futuros.