UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC
CURSO DE CIÊNCIA DA COMPUTAÇÃO

DÉRICK SOUZA MIRANDA

BLOCKCHAIN NA EDUCAÇÃO: USO DA TECNOLOGIA COMO PROVA DE EXISTÊNCIA DE DIPLOMAS E CERTIFICADOS

CRICIÚMA
2019

DÉRICK SOUZA MIRANDA

BLOCKCHAIN NA EDUCAÇÃO: USO DA TECNOLOGIA COMO PROVA DE EXISTÊNCIA DE DIPLOMAS E CERTIFICADOS

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientador: Prof. Me. Paulo João Martins

CRICIÚMA

2019

DÉRICK SOUZA MIRANDA

BLOCKCHAIN NA EDUCAÇÃO: USO DA TECNOLOGIA COMO PROVA DE EXISTÊNCIA DE DIPLOMAS E CERTIFICADOS

Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do Grau de Bacharel, no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em Segurança da Informação.

Criciúma, 09 de Dezembro de 2019.

BANCA EXAMINADORA

Prof. Me. Paulo João Martins - UNESC - Orientador

Prof. Me. Gustavo Bisognin - UNESC

Prof. Dr. Rogério Antônio Casagrande - UNESC

AGRADECIMENTOS

Agradeço a minha mãe Dirce Souza, que sempre prestou um apoio incondicional durante todo os momentos, e que me mostrou e motivou a enfrentar as dificuldades e conquistar aquilo que desejo.

Um agradecimento em especial, para meus amigos Iury Piva e Jeanluca Fernandes, que estiverem junto a mim desde o início da faculdade. A eles minha gratidão, e que nossa amizade permaneça para toda a vida.

Agradeço também em especial o professor Luciano Antunes, que sempre me orientou e esteve presente para sancionar minhas dúvidas.

Um agradecimento também a meu orientador, Paulo João Martins, que esteve colaborando comigo nesta última etapa da faculdade.

“The Times 03/Jan/2009 Chancellor on brink of second bailout for banks”.

The Times

RESUMO

Desenvolvida para a solução de um meio de pagamento eletrônico, a *Blockchain* surgiu com a criação da criptomoeda Bitcoin. Ela consiste em uma rede P2P de transações, que são mantidas e verificados pelos seus diversos usuários, mantendo a confiança e segurança dos dados, com o uso de diversas técnicas de criptografia. Por manter um registo imutável, a adulteração de dados nessa rede se torna um processo computacional impraticável, dessa forma, seu uso passou a ser explorado em diversas áreas. Neste trabalho, foi realizado um estudo em relação ao uso desta tecnologia no ambiente educacional, como forma de prova de existência de diplomas e certificados, tendo em vista que o atual cenário, envolve processos manuais para seu compartilhamento e verificação, ainda, estabelece uma relação com intermediadores, responsáveis por tais procedimentos. Além disso, os métodos utilizados no momento, são suscetíveis a falhas e fraudes, e são atribuídos a formatos que não podem garantir uma grande longevidade para tais documentos. Ao utilizar a *blockchain* para registrar esses documentos, é possível criar uma prova irrefutável de que eles realmente foram emitidos por uma instituição de ensino, e as técnicas de criptografia utilizadas para sua geração, impedem a adulteração de seus dados. O estudo realizado, descreve o atual cenário de emissão de diplomas no ambiente educacional do Brasil, e então detalha algumas características essenciais para adoção da *blockchain* neste meio, ainda apresenta um cenário favorável para o uso da tecnologia em grande escala. Tendo em vista a necessidade de compreensão das técnicas utilizadas, um estudo detalhado foi realizado com foco no protocolo Bitcoin e da tecnologia de sua rede. Como resultado deste estudo, foi desenvolvido um protótipo, capaz de suprir os passos necessários para efetivar os registros de tais documentos. Com o protótipo finalizado e com os dados obtidos pelo levantamento bibliográfico, foi possível alcançar o objetivo do projeto e identificar melhorias e características necessárias para esse tipo de ferramenta.

***Palavras-chave**:* Prova de Existência. Bitcoin. Blockchain na educação. Criptografia.

ABSTRACT

Developed to solve problems in electronic payment systems, the Blockchain was born with the cryptocurrency Bitcoin. Blockchain is a P2P network made up with transactions which are hold and verified by all members from the network, with this bringing trust and security for all data using a variety of encryption techniques. Because of the immutability characteristic of the records, the adulteration of data in this network becomes an impractical computational process, this way the Bockchain has been explored in many areas. In this paper a research was made to apply the use of this technology in the educational environment, as a proof of existence of diplomas and certificates, considering that the present scenario needs manual processes for their sharing and verification and establishes a relationship with intermediaries responsible for such processes. The current used methods are susceptible to failure and fraud and are assigned to formats that cannot provide these documents for the long-term. With blockchain this type of record allows us to perform an irrefutable proof of which documents were issued by one educational institution and the encryption techniques used for its generation prevent the tampering of this data. The study describes the current environment of issue diplomas in the Brazilian educational system and the details some essential features for its application yet presents a favorable scenario for the use of this technology in a large scale. Given the need to understand the used techniques in this system a detailed study was conducted focusing on the Bitcoin protocol and Blockchain technology. As one of the results of this paper a prototype was developed which can supply all the steps to register such documents in to the blockchain. As results of the developed prototype and with the research results it was possible to achieve the project's objectives and identify improvements and resources for a system that allow to share and manage all data from the educational scenario.

***Palavras-chave**:* Proof-of-Existence. Bitcoin. Blockchain in education. Cryptography.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

API	Interface de programação de aplicações
BTC	Criptomoeda Bitcoin
CPU	Unidade centrais de processamento
DPoS	*Delegated-Proof-of-Stake*
GPU	Unidade de processamento gráfico
HTML	*HyperText Markup Language*
HTTP	*HyperText Transfer Protocol*
IES	Instituição de Ensino Superior
JSON	*JavaScript Object Notation*
MEC	Ministério da Educação
NIST	Instituto Nacional de Padrões e Tecnologia
P2P	*Peer-to-Peer*
P2PK	*Pay-to-Public-Key*
P2PKH	*Pay-to-Public-Key-Hash*
P2SH	*Pay-to-Script-Hash*
PDF	*Portable Document Format*
PKI	Infraestrutura de Chaves Públicas
PoF	*Proof-of-Existence*
PoS	*Proof-of-Stake*
PoW	*Proof-of-Work*
REST	*Representational State Transfer*
SHA	*Secure Hash Standard*
UFPB	Universidade Federal da Paraíba
UFSC	Universidade Federal de Santa Catarina
UNIC	Universidade de Nicosia
UTXO	*Unspent Transaction Outputs*
WIF	*Wallet Import Format*
XML	*Extensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO

Nas últimas décadas, o uso da tecnologia de informação por parte da sociedade, tornou-se cada vez mais inevitável e a necessidade de gerenciar documentos vem crescendo consideravelmente, principalmente, com o surgimento da Internet e consequentemente do fenômeno conhecido como explosão informacional (ARELLANO, 2004; MARCONDES; SAYÃO, 2002), a quantidade de informações geradas e necessárias para as atividades sociais, tornaram-se cada vez maiores.

Apostos em papel, os documentos não mais correspondem às necessidades de rapidez na circulação das informações, ainda existem suas limitações, referentes à conservação, transmissibilidade ou segurança (GANDINI et al., 2016).

Os documentos tradicionais, bem como suas medidas de segurança, se desintegram ou podem se tornar irrecuperáveis e ainda existem efeitos da temperatura, umidade, poluição do ar, danos provocados pelo uso indevido e regular e até mesmo as catástrofes naturais (ARELLANO, 2004).

Com base nos fundamentos apresentados, é possível perceber que o uso de tecnologias, para armazenamento de documentos em formato digital, convém a ser de grande utilidade, onde as atuais ferramentas, além da preservação, permitem fácil catalogação, busca e organização.

A validação de documentos é essencial em qualquer contexto legal, normalmente, documentos físicos são validados por meio de autoridades centrais, onde são aplicados registros mecânicos, os concedendo maior segurança (LEAO; CANEDO; GOMES, 2017).

Quando se trata de documentos digitais, para que eles tenham validade jurídica, é necessário que atendam a requisitos que garantam sua confiabilidade, normalmente, a técnica de assinaturas digitais é uma solução, que é gerada por criptografia assimétrica ou simétrica (GANDINI; SALOMÃO; JACOB, 2001; MARCACINI, 1999).

Protocolos que utilizam criptografia assimétrica, consistem em um par de chaves criptografadas, matematicamente conectadas: uma pública, gerada a partir da

chave privada, a qual é conhecida somente pelo proprietário (GIRAULT, 1991 apud MARTINS, 2018).

Uma característica em comum, tanto para documentos tradicionais e eletrônicos, é a necessidade de um intermediador para sua autenticação. Quando apostos em papel, uma autoridade central é usada para autenticá-lo, por exemplo, um cartório, quanto aos formatos digitais, os envolvidos utilizam seu par de chaves para assiná-lo, também geradas por uma autoridade central. Em ambos os casos há necessidade de uma entidade intermediadora, onde é possível identificar a falta de privacidade.

Na educação, certificados e diplomas, uma forma de “documento”, possuem grande importância, normalmente impressos em papel, eles atribuem capacitações a seus portadores, sejam eles emitidos por Instituições de Ensino Superior (IES) ou instituições educacionais (GRÄTHER et al., 2018, tradução nossa).

Mesmo possuindo um grande valor e qualificações pessoais, em um estudo realizado no Centro Universitário Luterano de Palmas, foram identificadas as seguintes características (LEAO; CANEDO; GOMES, 2017):

a) alto custo na emissão e impressão dos diplomas – devido aos papeis especiais utilizados e aos mecanismos físicos para garantir autenticidade;
b) dificuldade para entrega e desperdício de material;
c) prazo de entrega e produção longos;
d) ineficiência na emissão de 2ª vias.

O estudo prático citado anteriormente, pode se encaixar em qualquer universidade, e tendo em vista que, de acordo com o Ministério de Educação Brasileiro (MEC), em 2016 havia mais de 8 milhões estudantes matriculados, aproximadamente 3 milhões de alunos ingressaram e mais de um 1,1 milhão de estudantes concluíram, cursos de graduação no ensino superior, esses dados, tornam possível idealizar um grande desperdício de dinheiro e tempo, relacionados a emissão dos diplomas.

De acordo com o MEC, todo registro e armazenamento dos dados acadêmicos devem ser mantidos pela IES (BRASIL, 2018). Dispostos a diversos riscos, esses dados estão sujeitos a eventual perda, e até mesmo, o possível

encerramento das atividades da instituição, como o caso da "Gama Filho", uma instituição descredenciada pelo MEC, onde milhares de estudantes perderam todo seu histórico acadêmico, impossibilitando até mesmo, a sua comprovação de graduação, ou transferência para outras universidades (SALLES, 2019).

Outra situação problemática relacionada a emissão de diplomas, é a indústria da fraude, onde existem quadrilhas especializadas na venda de diplomas falsos. O Ministério Público do Paraná investigou casos de mais de 500 fraudes em 2013 (GIMENES, 2013), ainda existem casos, como em Mato Grosso, onde a Polícia Federal investigou uma quadrilha, que revalidava diplomas de medicina de forma fraudulenta (GLOBO, 2017). No estado do Espírito Santo, mais de 100 professores estavam sendo investigados por apresentarem diplomas fraudados, a Secretaria de Educação do estado, também revelou que o prazo referente aos processos administrativos que os investigavam, teriam duração de até 180 dias, com possibilidade de prorrogação (MOURA, 2017).

É com base no cenário abordado, que é proposto o estudo de caso de uma plataforma para emissão, verificação e compartilhamento de diplomas e certificados, por meio do uso da tecnologia *blockchain*, garantindo assim, sua segurança e imutabilidade.

Inicialmente, introduzida como uma tecnologia para meios de pagamentos digitais, por meio da criptomoeda Bitcoin (BTC), (NAKAMOTO, 2008, tradução nossa), ela pode ser aplicada nas mais diversas áreas, como uma forma de suprir segurança e privacidade. Ela também permite a existência de registros cronológicos imutáveis, eliminando a necessidade de terceiros confiáveis, proporcionando transparência e total segurança (COSTA et al., 2018; GRÄTHER et al., 2018, tradução nossa).

1.1 OBJETIVOS

Esta seção, tratará dos objetivos gerais e específicos, a serem alcançados por este trabalho.

1.1.1 Objetivo geral

O objetivo deste trabalho, consiste em explorar o uso da *blockchain*, como mecanismo de emissão e validação de certificados e diplomas, em uma instituição de ensino superior.

1.1.2 Objetivos específicos

Os objetivos específicos deste estudo consistem em:

a) descrever e aplicar os conceitos da tecnologia *blockchain*;
b) demonstrar e aplicar a prova de existência, em um registro de documentos em uma *blockchain*;
c) modelar e especificar um protótipo de uma aplicação, que utilize uma interface de programação de aplicações (API), de forma a registrar arquivos digitais na *blockchain* e fornecer uma prova de existência;
d) realizar a escolha da utilização da *blockchains* públicas, privadas ou híbridas (*consortium*), em instituições de ensino, baseadas em suas limitações do uso.

1.2 JUSTIFICATIVA

Diplomas, certificados e conquistas adquiridos por estudantes ao longo de suas vidas, se tornarão uma importante referência, tanto para suas carreiras profissionais, como, para suas carreiras estudantis, porém, devido à falta de eficácia imposta por meio dos mecanismos de segurança atribuídos a eles (CHENG et al., 2018, tradução nossa), uma série de problemas relacionados ao compartilhamento, verificação e até mesmo fraudes, podem ser ocasionados. Por outro lado, ainda é possível que estudantes venham a perder seus registros escolares ou acadêmicos, entretanto, obter novas cópias é um processo lento, longo e na maioria das vezes ineficaz (GRECH; CAMILLERI, 2017 tradução nossa).

A área da preservação digital é, de certa forma, recente, portanto, os conceitos atribuídos a ela ainda não foram bem estabelecidos, e seu maior desafio envolve sua gestão e preservação, durante todo o seu ciclo de interesse, visando atender a longo prazo as necessidades dos atuais e futuros usuários (SAYÃO; SALES, 2012 apud COSTA et al., 2018; RAMALHO et al., 2007, tradução nossa).

Uma atual proposta para a preservação digital, é manter múltiplas cópias do conteúdo em locais distribuídos (RUUSALEPP; DOBREVA, 2012, tradução nossa; RAMALHO et al., 2007, tradução nossa), a qual propõe níveis de segurança altíssimos, onde, supondo que um dos locais de armazenamento venha a ser comprometido, toda a informação continua armazenada e segura no restante da rede.

Contudo, é economicamente inviável e pouco provável, para uma instituição de ensino manter um sistema de servidores distribuídos, e é neste cenário, no qual torna-se viável explorar o potencial da estrutura da *blockchain*.

Com um serviço de prova de existência, do inglês *Proof-of-Existence* (PoF), por meio de uma *blockchain*, é possível prover as informações necessárias para emissão e validação, e consequentemente realizar a padronização de informação, segurança de armazenamento e estabelecimento de regras para o compartilhamento dos dados. Isto é uma forma de contornar os problemas impostos pelos atuais métodos de emissão de diplomas, como também para qualquer contexto que necessita emitir “documentos como prova”.

Esta tecnologia, é essencialmente um “livro razão público”, ou um banco de dados distribuído, capaz de registrar dados por meio de transações, onde cada uma delas é verificada, pelo consenso da maioria dos participantes e uma vez registrada, ela torna-se imutável, sendo possível sua verificação a qualquer momento (CROSBY et al., 2016, tradução nossa).

De acordo com Crosby et al. (2016), ela pode ser vista como um registro de dados armazenados em blocos, conectados uns aos outros por funções criptográficas, que impedem a adulteração do conteúdo armazenado. Ela é construída de forma descentralizada, em diferentes nós da rede, de modo que cada nó pode ter a cópia completa, ou não, de todos os registros armazenados no sistema (ULRICH, 2014 apud SOBRINHO et al., 2017).

Seu estudo e aplicações são cada vez mais relevantes e seus potenciais de transparência, segurança e descentralização, são gradativamente mais explorados. São inúmeras as áreas que podem tirar proveito da tecnologia, e consequentemente, sua adoção vem sendo considerada em serviços governamentais, como por exemplo na Estônia, onde é estudada desde 2008, no qual, há serviços para registros das atividades judiciais, legislativas e saúde (E-ESTONIA), assim como, na logística (WANG; YUAN, 2016, tradução nossa), saúde (MERTZ, 2018, tradução nossa; METTLER, 2016, tradução nossa), internet das coisas (CONOSCENTI; VETRÒ; MARTIN, 2016, tradução nossa; MILLER, 2018, tradução nossa) e tantas outras possíveis soluções aptas a explorá-la.

Os atuais sistemas para verificação de certificações são lentos, não padronizados e relativamente não confiáveis, com o uso da *blockchain* e seus métodos de criptografia, é possível criar uma infraestrutura adequada, atendendo suas necessidades.

1.3 ESTRUTURA DO TRABALHO

O presente trabalho é formado por 9 capítulos, o qual, no primeiro destes, uma abordagem inicial ao tema é feita, seguido pelo seu objetivo geral, objetivos específicos, justificativa e estrutura.

No segundo capítulo, são apresentadas informações sobre as características que envolvem os diplomas e certificados, bem como seu atual cenário e limitações encontradas.

O terceiro capítulo aborda as técnicas de criptografia mais utilizadas na *blockchain*.

O quarto e quinto capítulos, descrevem respectivamente, informações sobre o Bitcoin e a *blockchain*, relatando seu histórico, estrutura e suas propostas de uso, assim como, as formas de inserção de dados, bem como suas limitações e custos, por fim, descreve o cenário presente entre a emissão de diplomas e essas tecnologias.

No sexto capítulo são descritos os trabalhos correlatos ao objetivo deste estudo.

O sétimo capítulo, refere-se a metodologia do trabalho proposto, onde é descrito detalhadamente todos os procedimentos realizados para seu desenvolvimento.

No oitavo capítulo são apresentados os resultados obtidos e uma análise de dados e sua discussão.

Por fim, o nono capítulo aborda a conclusão alcançada com a pesquisa e desenvolvimento do projeto, bem como as sugestões de trabalhos futuros e então encontram-se as referências utilizadas no desenvolvimento do trabalho.

2 DIPLOMAS E CERTIFICADOS

Um diploma, é caracterizado como prova da graduação e um certificado pode ser identificado como uma prova de capacitação.

Ambos são documentos emitidos por instituições de ensino, no entanto, quando relacionados ao ensino superior, somente são válidos aqueles reconhecidos e aceitos pelo Ministério da Educação, os quais tornam o titular habilitado a exercer determinada profissão, com validade em território nacional, eles podem ser emitidos como forma de comprovação de conclusão em cursos de graduação, bacharelado, licenciatura, pós-graduação, mestrado e doutorado.

No Brasil, a curadoria e gerenciamento desse tipo de registro é mantida pelo MEC, o órgão federal que trata da política nacional de educação em geral.

O foco deste estudo estará voltado a diplomas emitidos por IES, reconhecendo a importância gerada a partir deles.

2.1 EMISSÃO DE DIPLOMAS NO BRASIL

De acordo com Lei de Diretrizes e Bases da Educação, a lei nº 9394/1996 afirma que a emissão dos diplomas é responsabilidade das instituições, incluindo comprovantes e certificados de conclusão de curso. Em caso de instituições não universitárias, tais registros deverão ser feitos por órgãos indicados pelo Conselho Nacional de Educação (BRASIL, 1996).

Quanto a sua preservação, as IES devem manter os registros de todos documentos relacionados e emitidos por ela, e sua a gestão está regulamentada pela Lei 8159/91.

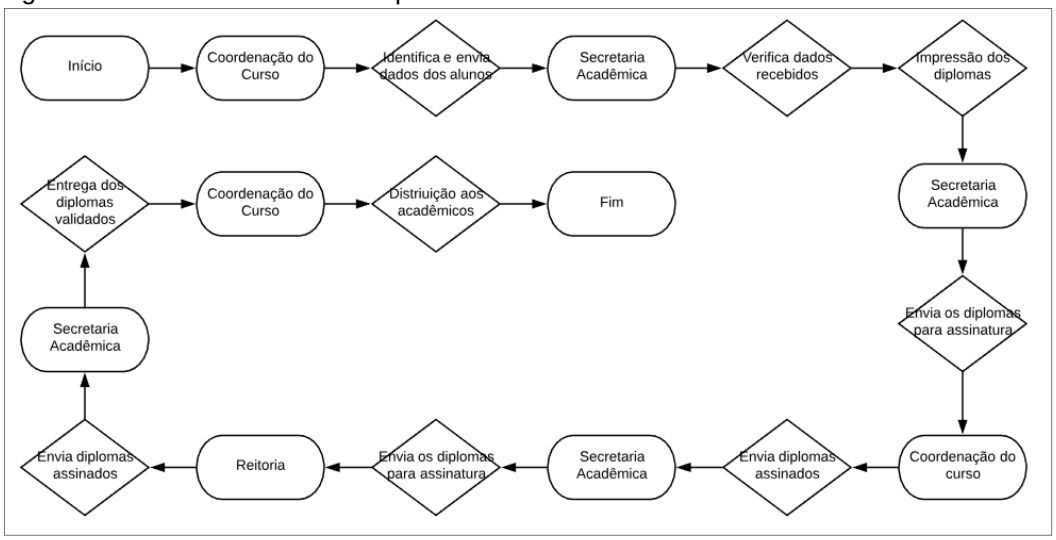
A Portaria nº 1.095, 25 de outubro de 2018, emitida pelo MEC, estabelece as métricas necessárias para emissão de tais documentos, além disso, também faz algumas recomendações relacionados ao fluxo de registro, porém, ele e a manutenção dos arquivos podem ser escolhidos de forma livre.

O fluxo de emissão de diplomas é variável, no entanto, é possível considerar que o passo a passo tomada pelas instituições tenha grande semelhança.

A seguir, são listados os passos (figura 1), considerados como mínimos, para a seleção dos alunos com os requisitos necessários para conclusão do curso (PALMA et al., 2019, tradução nossa):

a) cada curso envia a Secretaria Acadêmica da instituição uma lista de estudantes que atendem os requisitos necessários;
b) os dados são recebidos e verificados, e são identificados se todos estudantes possuem os requisitos para conclusão do curso;
c) a Secretaria Acadêmica registra os dados referentes ao aluno e curso (nome, curso, data e outras informações básicas);
d) é realizada a impressão em papel;
e) os coordenadores assinam os diplomas de seus respectivos cursos;
f) uma vez assinados, eles são enviados pela Secretaria Acadêmica para receber a assinatura do reitor da instituição;
g) os diplomas são assinados pelo reitor e novamente enviados a Secretaria Acadêmica;
h) as secretarias dos cursos recebem os diplomas para realizar a distribuição aos acadêmicos.

Figura 1 - Fluxo de emissão de diplomas



Fonte: Do autor.

Em uma consulta, realizada com membros da administração da universidade em que é feito o presente estudo, o mesmo fluxo de emissão de diplomas foi identificado.

2.2 DIPLOMAS DIGITAIS

Durante o estudo do presente trabalho, o MEC publicou Portaria nº 554, que regulamenta o registro e a emissão de diploma digital, em todas IES que compõem o Sistema Federal de Ensino (BRASIL, 2019).

As instituições terão um prazo de 24 meses, a partir da data de sua publicação, para se adequarem às especificações necessárias para emissão dos diplomas no meio digital. De acordo com o MEC, o objetivo é possibilitar o melhor aproveitamento de recursos, preservando as mesmas condições e garantias dos diplomas físicos e evitar as atuais burocracias relacionados a sua emissão.

O padrão estabelecido para utilização é o *Extensible Markup Language* (XML), uma linguagem de codificação, capaz de armazenar informações de forma estruturada. Para garantir sua segurança, eles serão registrados com certificação digital e carimbo de tempo, com o uso assinatura digital, seguindo as regras da Infraestrutura de Chaves Públicas Brasileira.

O MEC ressalta que o diploma digital será o arquivo XML, devidamente assinado e não sua representação visual, permitindo as instituições a escolha de uma forma de sua representação, a qual deve manter um mecanismo de acesso aos dados XML do documento (BRASIL, 2019).

O diploma deve manter um link de acesso, pelo padrão *HyperText Transfer Protocol Secure* (HTTP), e com ele deve ser possível o acesso a seu formato visual, a visualização dos dados no arquivo, status do diploma e sua validação.

Os processos de emissão e validação, devem seguir a legislação para os demais serviços educacionais prestados pela instituição, e continuam sendo responsabilidade da IES, o armazenamento e fluxo de emissão. Além disso, não é prevista nenhuma cobrança ao acadêmico no decorrer deste processo, a não ser solicitada a impressão do documento pelo estudante (BRASIL, 2019).

Ainda, cabe a IES, enviar ao MEC todos os arquivos emitidos e quaisquer alterações ou fraudes, estarão sujeitas às medidas administrativas, civis e criminais pertinentes (BRASIL, 2019).

2.3 COMPONENTES E PROCESSOS DE UM DIPLOMA

Os diplomas devem possuir informações básicas, necessárias para identificação de seus valores e sua conferência (GRÄTHER et al., 2018, tradução nossa; GRECH; CAMILLERI, 2017 tradução nossa).

Segundo Grech e Camilleri (2017, tradução nossa), eles devem conter os seguintes componentes:

a) reinvindicação: a declaração dos fatos, por exemplo, a conclusão de um curso superior;
b) emissor: uma instituição válida e autorizada, confirmando os fatos da reinvindicação;
c) evidência: informa o procedimento pelo qual a reinvindicação é verificada e traz algumas informações adicionais referentes a ela;
d) condecorado: a pessoa a qual recebe a reinvindicação;
e) certificado: o documento que atesta a identidade do emissor, a identidade do condecorado, a reinvindicação e informa as evidências necessárias. Um certificado deve conter registro que permitam a verificação de todos os dados.

Grech (2017, tradução nossa) ainda identifica os processos envolvidos na "criação" de um destes documentos, sendo eles:

a) emissão: é o processo no qual todos os seus componentes são registrados;
b) verificação: ocorre quando é realizada a apuração de autenticidade de diploma. Ela pode ser realizada pela: confirmação de registros físicos atribuídos ao diploma, por exemplo, selos ou assinaturas; contato com a instituição emissora; comparação com dados em um banco de dados oficial; verificação por meio da assinatura digital;
c) compartilhamento: é quando o condecorado o compartilha.

2.4 LIMITAÇÕES

Embora, a maioria dos diplomas emitidos por instituições ainda são atribuídos por meio físico, e mesmo, considerando aqueles emitidos em meios digitais, não se pode supor um “formato perfeito” em um ambiente, onde existem instituições usando-os de forma híbrida, no qual, os dados apostos em papel, continuam mantendo suas cópias digitais em bancos de dados centralizados.

De qualquer modo, há uma significante limitação em cada meio, gerando a necessidade de uma melhor forma de utilização.

2.4.1 Limitações diplomas em papel

As versões apresentadas em formato físico, podem oferecer um bom nível de segurança e possuem certas vantagens, no estudo realizado por Grech e Camilleri (2017, tradução nossa), elas podem ser identificadas como:

a) dificuldades para forjar, tendo em vista as medidas de segurança atribuídas fisicamente a eles (selos, alto relevo, fitas metálicas, marcas d’agua);
b) manuseio diretamente pelo proprietário;
c) fácil armazenamento;
d) podem ser apresentados a qualquer momento a qualquer pessoa.

Porém, ao mesmo tempo, apresentam desvantagens significantes (GRECH; CAMILLERI, 2017, tradução nossa; PALMA et al., 2019, tradução nossa):

a) as medidas de segurança utilizadas possuem riscos de sua fraude ou forja;
b) a instituição emissora deve manter um registro central de todos diplomas emitidos, como forma de verificar sua autenticidade;
c) embora os certificados sejam válidos, sua capacidade de verificação pode ser perdida;
d) o processo de verificação manual é longo e individual, e é extremamente suscetível a falhas humanas;

e) quanto mais seguro, maior é seu custo de produção.

2.4.2 Limitações diplomas digitais

Os diplomas apresentados em formatos digitais, quando não aplicados pelo uso da *blockchain*, possuem grandes vantagens em relação aqueles concedidos em papel (GRÄTHER et al., 2018, tradução nossa; GRECH; CAMILLERI, 2017 tradução nossa):

a) sua verificação pode ser feita por meio de *softwares*, evitando intervenção humana;
b) sua segurança depende diretamente dos protocolos de criptografia utilizados, tornando sua forja um processo custoso;
c) forjar ou alterar o carimbo de tempo e até mesmo o número de identificação do documento, pode ser um processo impossível de acordo com o modelo utilizado.

Ainda assim, apresentam certas desvantagens (GRECH; CAMILLERI, 2017, tradução nossa):

a) sem o uso de assinaturas digitas, eles continuam sendo fáceis de forjar;
b) o uso de assinatura digitais, normalmente envolve uma relação com uma entidade autorizada, responsável por manter sua integridade, no entanto, ela possui o controle dos aspectos de segurança e verificação, os quais podem vir a ser explorados;
c) os padrões relacionados ao uso de assinaturas digitais sofrem constantes mudanças, o que pode ocasionar o uso de *softwares* proprietários no processo de verificação;
d) não se pode garantir total confiabilidade sobre a entidade autorizada, responsável por manter o processo da assinatura digital seguro;
e) a partir do momento que a entidade encerre suas atividades, o documento torna-se sem valor, uma vez que não é possível verificá-lo;
f) é comum registros de assinaturas digitais estarem associados ao vazamento de dados.

2.5 DESAFIOS DA PRESERVAÇÃO DIGITAL

A quantidade de dados vem crescendo cada vez mais, estudos mostram que aproximadamente 20%, de todos os dados do mundo, foram digitalizados nos últimos anos (ZYSKIND; NATHAN; PENTLAND, 2015, tradução nossa).

Outro fator importante é a conservação de tais documentos, a qual, quando comparada com a preservação por meio de coleções físicas, a preservação digital traz consigo um grande potencial de risco e proteção (SKINNER; SCHULTZ, 2010, tradução nossa). Tal potencial de risco, é caracterizado pela fragilidade do armazenamento digital, suscetível a falhas técnicas e humanas (COSTA et al., 2018).

O armazenamento digital proporciona grande escalabilidade e fácil manutenção, no entanto, ele pode ser facilmente destruído por meio de falhas técnicas ou ainda por falhas humanas, com muito mais facilidade e rapidez quando equiparada a representações físicas. Por outro lado, seu potencial de proteção é caracterizado pela ágil e rápida reprodução e manuseio, fidelidade e integridade (COSTA et al., 2018).

Como visto anteriormente, a preservação digital é uma área de pesquisa recente, ela ainda está nos estágios iniciais de sua formação, e seu aparato tecnológico, metodológico e político ainda estão sendo construídos.

Costa et al. (2018), afirma que ela possui um paradoxo, a descaracterizando do ponto de vista de preservar, onde, envolve o ato de manter imutável e intacto. No entanto, no ambiente digital, são permitidas a mudança, recriação e renovação, onde significam mudar formados, atualizar mídias e substituir *hardware* e *softwares*.

No próximo capítulo, serão descritas as técnicas de criptografia utilizadas na *blockchain,* e que podem oferecer os níveis de segurança ideais para o ambiente de emissão de diplomas.

3 CRIPTOGRAFIA

A criptografia é um dos fatores mais importantes da *blockchain*, é por meio dela que a rede se torna segura e confiável.

Neste capítulo, serão abordados os algoritmos necessários para um melhor entendimento da tecnologia.

3.1 CRIPTOGRAFIA ASSIMÉTRICA

Também conhecido como infraestrutura de chaves públicas, do inglês *public key infrastructure* (PKI), esse tipo de criptografia foi desenvolvido pelos pesquisadores Whitfield Diffie, Martin Hellman e Ralph Merkle, a partir de estudos realizados nos anos 70 (GOYA et al., 2009; MENKE).

Diferente de outros algoritmos, que faziam o uso de uma mesma chave para realizar a criptografia e descriptografia, este sistema solucionou o problema da distribuição das chaves, encontrado nas primeiras versões desta técnica (TANENBAUM, 2003).

A criptografia assimétrica, consiste em um par de chaves matematicamente conectadas: uma privada, somente conhecida pelo proprietário, e uma pública, a qual é distribuída de forma livre (USHMANI, 2019, tradução nossa; ZHENG et al. 2017, tradução nossa).

Nesse sistema, os algoritmos E e D, encriptação e decriptação respectivamente, podem ser declarados da seguinte forma (TANENBAUM, 2003):

a) D(E(P)) = P;
b) é extremamente difícil deduzir D a partir de E;
c) E não pode ser decifrado por um ataque de texto simples escolhido.

A primeira característica diz que, se aplicado D a uma mensagem criptografada, E(P), deve ser obtida o conteúdo original P, sem essa propriedade, um destinatário não poderia verificar um texto codificado. A segunda, diz a respeito da geração das chaves, a qual deve ser um processo simples, a pública é criada a partir da privada, apesar da ligação matemática, a tentativa de obter uma a partir da outra,

é um processo computacionalmente impraticável. A terceira trata da tentativa de obter a mensagem original por meio de força bruta (TANENBAUM, 2003).

Considerando uma mensagem criptografada com uma chave pública, ela somente pode ser de*script*ografada com sua respectiva chave privada. Supondo uma comunicação entre dois usuários, onde Alice envia a Bob uma mensagem criptografada com a chave pública de Bob, mesmo que a mensagem venha a ser recebida por outra pessoa, somente Bob pode verificá-la utilizando sua chave privada para de*script*ografar seu conteúdo (ZHENG et al., 2017, tradução nossa).

.

3.2 FUNÇÕES HASH CRIPTOGRÁFICAS

Funções *hash* criptográficas, se tornaram uma das bases da computação moderna, com seu uso a segurança da maioria das redes é garantida, com elas são possíveis a verificação de identidade, integridade de arquivos e proteção de dados (JIAN-DONG et al., 2010, tradução nossa; WANG et al., 2017, tradução nossa).

Esse tipo de função é capaz de gerar uma saída de tamanho fixo, a partir de uma entrada de qualquer tamanho arbitrário, onde normalmente, a entrada é chamada de mensagem e a saída é chamada de resumo, do inglês *hash.*

Uma função deste tipo, ideal para uso em criptografia deve ser considerada unidirecional, isto é, dada uma mensagem, o cálculo de seu resumo deve ser simples, e reconstruí-la deve ser computacionalmente inviável (GAEINI; GHAFFARI; MOSTAGHIM, 2018, tradução nossa; SHARMA et al., 2018, tradução nossa).

O único meio possível de obter a mensagem a partir do resumo, é por ataque de força bruta, testando diversas possíveis entradas, para encontrar a saída correta. No entanto, qualquer modificação no conteúdo original, deve produzir um resultado totalmente diferente do primeiro, onde os dois não apresentam qualquer relação (SHARMA et al., 2018, tradução nossa).

A segurança destas funções é atribuída a três principais características, a primeira delas é a resistência à pré-imagem: refere-se a dificuldade de encontrar uma mensagem que produza um resumo conhecido, onde dado um resumo *h*, deve ser

complexo encontrar uma mensagem *m* tal que *hash(m) = h*; a segunda propriedade é a resistência à segunda pré-imagem: dada uma mensagem *m*, deve ser impraticável encontrar uma segunda mensagem *m'*, a qual produz o mesmo resultado da primeira, sendo assim *hash(m)* deve ser diferente de *hash(m');* a terceira propriedade é chamada de resistência à colisão: é relacionada a dificuldade de encontrar mensagens diferentes que produzam o mesmo resumo, onde *hash(m) ≠ hash(m')* (GAEINI; GHAFFARI; MOSTAGHIM, 2018, tradução nossa).

Seu uso é amplo, e uma de suas aplicações mais conhecidas é no processo de assinaturas digitais e abordando o contexto de *blockchain*, ao utilizar o protocolo do Bitcoin como exemplo, existem as funções SHA-256 e RIPEMD-160, que geram resumos de 32 bytes e 20 bytes, respectivamente.

3.3 CRIPTOGRAFIA DE CURVAS ELÍPTICAS

A teoria das curvas elípticas foi amplamente estudada durante muito tempo, inicialmente, ela foi utilizada para medir o perímetro e comprimentos das órbitas de planetas (ALMEIDA, 2002). Seu uso pode ser aplicado em diversas áreas, como em geometria diferencial, teorema dos números e geometria algébrica sobre corpos finitos (OLIVEIRA, 2009).

Quando definida sobre corpos finitos, a teoria das curvas elípticas possui uma grande importância para criptografia, o principal motivo é sua eficácia em fornecer uma vasta quantidade de grupos abelianos, e mesmo onde encontram-se um extenso número de elementos, ela permite adequar processos computacionais devido a sua rica estrutura algébrica. Um grupo abeliano ou também conhecido como grupo comutativo, consiste em uma estrutura $(G,*)$, onde $*$ é uma operação comutativa em G (FLOSE, 2011).

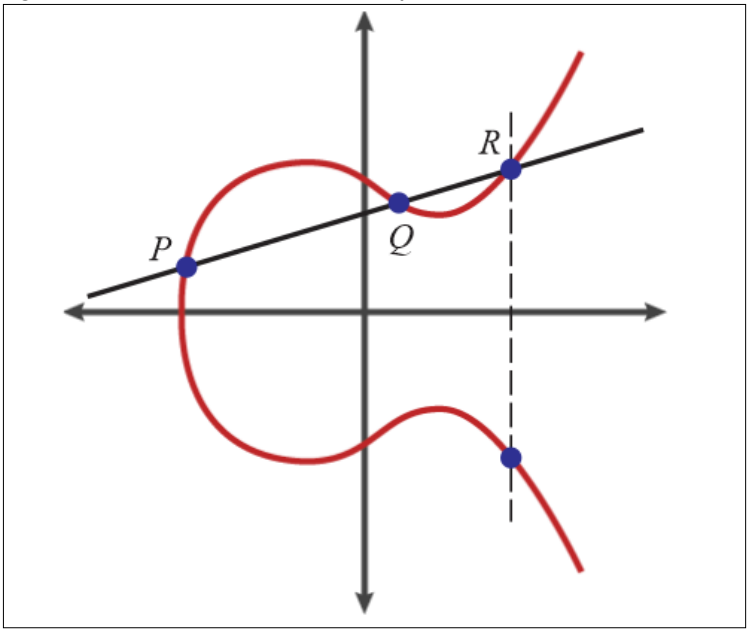
Um corpo, é um conjunto com duas operações, normalmente soma e multiplicação, onde a soma deve ser comutativa, associativa, possuir elemento neutro e um elemento simétrico, e a multiplicação deve ser comutativa, associativa, distributiva, possuir elemento neutro, e todo elemento não-nulo deve possuir um inverso (MARTINS, 2018).

Seu uso na criptografia foi proposto por Miller e Koblitz, no final da década de 80, servindo como uma forma eficiente de implementação de um sistema de chaves públicas. Além de ser mais rápido e ser capaz de utilizar chaves de menor tamanho, garante o mesmo nível de segurança de métodos mais comuns utilizadas na época, como o RSA (ALMEIDA, 2002).

Elas consistem em uma curva algébrica não-singular definida pela equação $y^2 = x^3 + ax + b$, onde seus coeficientes não apresentam cúspides ou auto interseções (OLIVEIRA, 2009). Em um gráfico, cúspides representam “pontas”, e auto interseções, pode sem vistos como cruzamentos na própria curva.

Para somar dois pontos pertencentes a uma curva elíptica, o resultado da soma deve ser um terceiro ponto sobre ela mesmo. O somatório do conjunto de pontos $E(Z_p)$ formam um grupo abeliano, onde o ponto no infinito 0 é o elemento neutro. Sejam, $P = (x1\,;\, y1)$ e $Q = (x2\,;\, y2)$ dois pontos distintos tomados em uma curva elíptica E, a soma de P e Q, resulta em $R = (x3\,;\, y3)$, onde é definida por meio de uma reta traçada de forma tangencial, sobre os pontos P e Q. Ao refletir o ponto R sobre o eixo x, obtém-se $E = P + Q$. Caso a soma dos pontos tenham os mesmos valores em x, a linha deve ser traçada na vertical (PORTNOI, 2005). A figura 2 ilustra o processo da soma dos pontos na curva elíptica.

Figura 2 - Soma em uma curva elíptica



Fonte: Adaptado de Antonopoulus (2014).

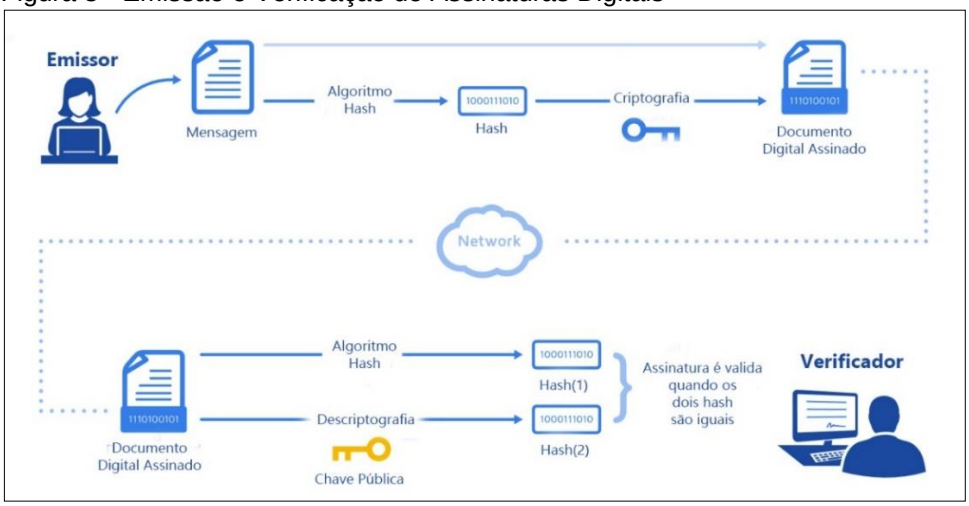
3.4 ASSINATURA DIGITAL

Assinatura digital é um método de autenticação digital, regularmente utilizado, para eliminar a necessidade de uma versão em papel do documento assinado. Ela pode ser empregada para gerar documentos com validade legal, por meio da infraestrutura de chaves públicas.

Ela deve prover autenticidade e uma forma, do receptor, verificar se o documento foi realmente assinado pelo emissor, assim como, garantir sua integridade, fazendo com que a assinatura seja única e não equivalha a mais nenhum outro documento, ou até mesmo, a ele próprio com a menor alteração possível. Ainda a maior de suas características é a irretratabilidade, onde impede que o emissor negue sua autenticidade (RIVEST; SHAMIR; ADLEMAN, 1978, tradução nossa).

O processo de assinatura é realizado da seguinte forma: o emissor deve encontrar o *hash* da mensagem que deseja assinar, por meio de funções criptográficas, como o algoritmo SHA-256, após isso, ele deve criptografá-lo utilizando sua chave privada, gerando uma assinatura digital a qual deve ser anexada ao documento original. Para verificação da assinatura, o receptor deve fazer uso, do mesmo algoritmo utilizado para sua geração, e assim encontrar o resumo da mensagem recebida, em seguida, ele deve descriptografá-la utilizando a chave pública do emissor, dessa forma, ele deve comparar o *hash* obtido por meio da assinatura, com o obtido por meio da mensagem, caso sejam iguais, a assinatura é válida. A figura 3 ilustra o processo de emissão e verificação da assinatura digital.

Figura 3 - Emissão e Verificação de Assinaturas Digitais



Fonte: Adaptado de DocuSign.

3.5 SHA-256

Publicado em 1993, pelo *National Institute of Standards and Technology* (NIST) o *Secure Hash Standard* (SHA), é uma função *hash,* muito utilizada para garantir a segurança na comunicação digital (MCEVOY et al., 2006, tradução nossa).

A primeira família de algoritmos SHA, consistia em uma função de 160-bits, e o maior problema encontrado em seu uso, foi uma forma de causar colisões, durante a execução da etapa de compressão, por meio da análise do processo expansão, o qual consistia somente em operações XOR.

Em relação as falhas do algoritmo SHA, o NIST publicou em 2001 a versão SHA2, a qual inclui os algoritmos SHA-256, SHA-384 e SHA-512, e de acordo com sua publicação, eles são capazes de garantir a integridade da mensagem, atendendo as características descritas na seção 3.2 (NIST, 2002, tradução nossa).

Os algoritmos SHA2 são caracterizados em três estágios, no primeiro são realizados o *padding* e *parsing,* no segundo a expansão e por fim, a compressão (MCEVOY et al., 2006, tradução nossa).

O estágio de *padding* consiste em adicionar um bit de valor 1 ao final da mensagem M e posteriormente, seu preenchimento com k bits de valor 0, até ela possuir um tamanho em bits l, igual à $448\ módulo\ de\ 512$. Após isso, o *parsing* é aplicado, onde, a mensagem é dividida em N blocos de 512-bits, indicados por M^1, M^2, M^3, ... M^N.

O algoritmo SHA-256 opera com palavras de 32-bits, sendo assim, cada bloco $M^{(i)}$ de 512-bits ainda é visto como 16 palavras de 32-bits, onde $M_t^{(i)}, 0 \leq t \leq 15$.

Para realizar a expansão, são necessários valores iniciais, identificados por $H^{(0)}$, quando se trata do algoritmo SHA-256, são necessários oito *hash* de 32-bits, $H_0^{(0)}$... $H_7^{(0)}$, em valores hexadecimais, os quais são estabelecidos e fornecidos pelo padrão NIST.

Cada bloco $M^{(I)}$ é expandido para 64 palavras de 32-bits, W_t, por meio de operações lógicas e binárias, sendo elas (MCEVOY et al., 2006, tradução nossa):

$$\sigma_0(\mathrm{x}) = ROT_7(x) \oplus ROT_{18}(x) \oplus SHF_3(x)$$
$$\sigma_1(\mathrm{x}) = ROT_{17}(x) \oplus ROT_{19}(x) \oplus SHF_{10}(x)$$

$$W_t = \begin{cases} M_i^i & 0 \leq t \leq 15 \\ \sigma_1(W_t - 2) + W_{t-7} + \sigma_0(W_{t-15}) + W_{t-16} & 16 \leq t \leq 63 \end{cases}$$

Onde $ROTn(x)$ é uma rotação circular de x em n posições para a direita e a função $SHFn(x)$ é a descolamento a direita de x em n posições. Todas as adições envolvidas no seu cálculo são módulos de 2^{32}. Os operadores utilizados nos cálculos do algoritmo são descritos na tabela 1.

Tabela 1 - Operações utilizadas no algoritmo SHA-256

Operações	Descrição
$\wedge$	Operação binária AND
$\vee$	Operação binária OR
$\oplus$	Operação binária XOR
$\neg$	Operação binária NOT
<<	Operação binária de deslocamento a esquerda
>>	Operação binária de deslocamento a direita
$\boldsymbol{ROT^n(x)}$	Rotação circular de x em n posições para a direita
$\boldsymbol{SHF^n(x)}$	Descolamento a direita de x em n posições

Fonte: Adaptado de NIST (2002).

Uma vez realizada a função de expansão, as W_t resultantes são enviadas para função de compressão, a qual consiste em 64 iterações, onde as primeiras utilizam os oito *hash* iniciais $\boldsymbol{H_j^{(i)}}$, representados por A, B, C ... H.

As iterações são realizadas utilizando as seguintes funções, onde x, y e z, representam palavras de 32-bits:

$$\mathrm{T1} = \mathrm{H} + \sum_1(\mathrm{E}) + \mathrm{Ch}(\mathrm{E},\mathrm{F},\mathrm{G}) + \mathrm{Kt} + \mathrm{Wt}$$
$$\mathrm{T2} = \sum_0(\mathrm{A}) + \mathrm{Maj}(\mathrm{A},\mathrm{B},\mathrm{C})$$
$$\mathrm{H} = \mathrm{G} \quad \mathrm{G} = \mathrm{F}$$
$$\mathrm{F} = \mathrm{E} \quad \mathrm{E} = \mathrm{D} + \mathrm{T1}$$

$$\mathrm{D} = \mathrm{C} \quad \mathrm{C} = \mathrm{B}$$

$$\mathrm{B} = \mathrm{A} \quad \mathrm{A} = \mathrm{T1} + \mathrm{T2}$$

$$\mathrm{Ch(x, y, z)} = (\mathrm{x\ AND\ y}) \oplus (\overline{\mathrm{x}}\ \mathrm{AND\ z})$$

$$Maj(x, y, z) = (x\ AND\ y) \oplus (x\ AND\ z) \oplus (y\ AND\ z)$$

$$\sum\nolimits_0 (\mathrm{x}) = ROT_2(x) \oplus ROT_{13}(x) \oplus ROT_{22}(x)$$

$$\sum\nolimits_1 (\mathrm{x}) = ROT_6(x) \oplus ROT_{11}(x) \oplus ROT_{25}(x)$$

Os valores K_t são 64 constantes de 32-bits, especificadas pelo NIST, e após realizadas todas interações da função, um *hash* $H^{(i)}$ é obtido por meio do cálculo:

$$\mathrm{H}_0^{(i)} = \mathrm{A} + \mathrm{H}_0^{(i-1)}, \mathrm{H}_1^{(i)} = \mathrm{B} + \mathrm{H}_1^{(i-1)}, \ldots, \mathrm{H}_7^{(i)} = \mathrm{H} + \mathrm{H}_7^{(i-1)}$$

O processo então de repete para todos os N blocos de 512-bits, e ao final de seu processamento, o resumo final $H^{(N)}$ de 256-bits é obtido ao concatenar todos os resultados formados:

$$H^{(N)} = \mathrm{H}_0^{(N)}\ \&\ \mathrm{H}_1^{(N)}\ \&\ \mathrm{H}_2^{(N)}\ \& \ldots \&\ \mathrm{H}_7^{(N)}$$

Os parâmetros e constantes utilizados na execução do algoritmo SHA-256 são representados pela tabela 2.

Tabela 2 - Parâmetros encontrados no algoritmo SHA-256

Parâmetros	Descrição
A, B, C, ... H	Variáveis utilizadas para representar as palavras no cálculo dos valores hash
$\boldsymbol{H^{(i)}}$	Representam os *hash*, onde $\boldsymbol{H^{(0)}}$ é o inicial e $\boldsymbol{H^{(N)}}$ o final
$\boldsymbol{H_j^{(i)}}$	j indica as palavras encontradas nos valores *hash*
$\boldsymbol{K_t}$	Constante utilizada nas iterações
$\boldsymbol{k}$	Quantidade número 0 adicionadas a mensagem durante a função de *padding*
$\boldsymbol{l}$	Tamanho da mensagem em bits
$\boldsymbol{m}$	Número de bits em cada bloco da mensagem
$\boldsymbol{M}$	Mensagem
$\boldsymbol{M^{(i)}}$	Bloco i da mensagem M, com um tamanho em m bits
$\boldsymbol{M_j^{(i)}}$	j indica o número de palavras encontradas no bloco $\boldsymbol{M^{(i)}}$
$\boldsymbol{n}$	Número de bits a serem operadores nas funções de rotação e deslocamento
$\boldsymbol{N}$	Número de bloco na mensagem
$\boldsymbol{T}$	Palavra temporária a ser utilizada no cálculo do *hash*
$\boldsymbol{w}$	Número de bits em uma palavra
$\boldsymbol{W_t}$	t indica número da palavra encontrada na mensagem

Fonte: Adaptado de NIST (2002).

3.6 RIPEMD-160

A família de algoritmos RIPEMD foi originalmente baseada no algoritmo MD4, introduzido em 1990 por Ronald Rivest, em decorrência de algumas falhas encontradas na sua função de compressão.

Esse tipo de algoritmo consiste em uma função *hash* criptográfica, que opera com palavras de 32-bits e suas operações primárias são: operação binária de deslocamento a esquerda; operações binárias AND, NOT, OR e XOR; e dois complementos de módulos de adições de palavras 2^{32} (PRENEEL; DOBBERTIN; BOSSELAERS, 1997, tradução nossa).

A função RIPEMD-160 traz um resultado de 160-bits, e é capaz de tornar qualquer entrada de tamanho arbitrário em blocos de 512-bits. Cada bloco é divido em 16 palavras de 4 bytes cada, as quais ainda são vistas como palavras de 32-bits.

Da mesma forma de operação do algoritmo SHA-256, para garantir que cada bloco possua 512-bits, uma função de *padding* é realizada para adicionar um bit de valor 1 seguido de *n* bits de valor 0. Como resultado, o algoritmo cria 5 palavras de 32-bits, as quais são convertidas em um resumo único de 160-bits (PRENEEL; DOBBERTIN; BOSSELAERS, 1997, tradução nossa).

Ele conta com cinco constantes iniciais, necessárias para realizar a função de compreensão, a qual computa cada estado para um novo bloco. Esta função em uns cinco estados paralelos contendo dezesseis passos cada.

Para iniciá-la, duas cópias do primeiro estado são realizadas, após isso, cada resultado é aplicado no novo estado, por fim, o resultado final é computado ao adicionar as cópias iniciais (PRENEEL; DOBBERTIN; BOSSELAERS, 1997, tradução nossa).

A função executada em cada estado consiste no seguinte cálculo, onde $\ll$ é uma rotação binária sobre s bits, e A, B, C, D, E representam seus estados:

$$A := (A + f(B, C, D) + X + K)^{\ll s} + E \; and \; C := C^{\ll 10}$$

4 BITCOIN

Neste capítulo será apresentado um breve histórico referente às moedas digitais, as quais foram a base para a implementação do protocolo Bitcoin.

4.1 HISTÓRIA

Transações monetárias online, normalmente relacionadas a comércio eletrônico, sempre necessitaram de instituições financeiras como intermediário de confiança (NAKAMOTO, 2008, tradução nossa). Usando transferências bancárias como exemplo, os bancos, são os responsáveis pela verificação do saldo, a ser subtraído de uma para outra conta e em pagamentos com cartões de crédito ou débito, o controle de cobranças é feito pelo emissor do cartão. Ao longo dos anos, algumas soluções passaram a ser utilizadas, como o Paypal (2019) e Mercado Pago (2019), onde auxiliam durante a realização de transferências e pagamentos, de toda forma, sempre se faz necessário, o uso de uma entidade como intermediador para verificação dos dados das transações, afim de evitar a duplicação das moedas digitais, permitindo que um usuário faça cópias de seu saldo e inicie diversas transações simultaneamente, esta prática, é conhecida como técnica do gasto duplo e é um dos maiores desafios a serem contornados nesse tipo de sistema (NAKAMOTO, 2008, tradução nossa).

Um meio de pagamento eletrônico baseado em criptografia, o qual utilizada criptomoedas, ao contrário dos meios mais comuns, baseados em confiança, é necessário para um bom desempenho, permitindo duas partes distintas realizar transações entre si, sem a necessidade de um terceiro, onde elas são computacionalmente impraticáveis de reverter, assegurando o ambiente e solucionando o problema do gasto duplo (NAKAMOTO, 2008, tradução nossa).

As criptomoedas são moedas digitais que usam criptografia para assegurar a validade das transações, buscando a descentralização do sistema de pagamentos, fazendo com que a informação de todas transações, seja distribuída entre todos os usuários participantes, sendo eles responsáveis pela integridade dos dados, portanto,

eliminam a necessidade de confiança em um único mediador (NAKAMOTO, 2008, tradução nossa).

Neste tipo meio de pagamento, o conjunto de usuários é interpretado como uma rede, e cada um deles pode ser visto como um nó. O consenso desse tipo de sistema é obtido por meio do voto da maioria dos nós, porém, ele sofre uma desvantagem, onde um usuário malicioso pode simular milhares deles, e assim, tomar o controle do consenso da rede, essa vulnerabilidade é conhecida como *Sybil Attack* e é com ela que um usuário pode explorar o gasto duplo de suas próprias moedas (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

Na década de 80, David Chaum, publicou um artigo propondo um método chamado pagamento às cegas, do inglês *blind signatures payment*, este seria o primeiro sistema usando criptografia em transações eletrônicas e acabou introduzindo o conceito de criptomoeda (BONNEAU et al., 2015, tradução nossa).

No método proposto, pessoas poderiam realizar pagamentos, os quais, o banco responsável assinaria digitalmente, sem conhecer seu conteúdo, desse modo, provendo a privacidade total aos usuários. David, também propôs as seguintes propriedades para seu sistema de pagamentos: inabilidade de terceiros identificaram o beneficiário, a data e o valor dos pagamentos realizados; capacidade de prover a prova de um pagamento, assim como, aptidão em identificar o beneficiário sobre determinadas circunstâncias; e permissão de cessar o uso a moedas reportadas como roubadas (CHAUM, 1982, tradução nossa).

No início dos anos noventa, Chaum, Fiat e Naor, propuseram uma moeda eletrônica irrastreável, porém, com a propriedade de somente permitir aos bancos o seu rastreamento, afim de evitar o gasto duplo (CHAUM; FIAT; NAOR, 1990, tradução nossa). Ainda na mesma década, vários métodos para pagamentos eletrônicos foram propostos, alguns conceberam grande contribuição para os atuais sistemas, onde eliminavam a necessidade de verificação por parte de bancos, permitiam a divisão de moedas em partes menores e aumentavam eficiência da rede (BONNEAU et al., 2015, tradução nossa). O maior problema encontrada por estes sistemas foi a adoção ao mercado, eles solucionavam um problema ao qual os usuários ainda não enfrentavam ou até mesmo não se importavam (NIMFUEHR, 2018, tradução nossa).

Os métodos de pagamento propostos na época, utilizavam os conceitos de criptografia assimétrica, onde a chave pública é o endereço da carteira o qual as moedas eram depositadas, e a chave privada é a forma de garantir permissão para sua utilização. Além disso, já introduziam o conceito de prova de trabalho, do inglês *Proof-of-Work (*PoW*)*, onde as moedas digitais podem ser geradas por qualquer pessoa, em uma quantidade equivalente ao esforço computacional empregado em sua criação (MARTINS, 2018).

Em 2005, Nick Szabo, publicou um artigo propondo a moeda digital *bit gold*, a qual era capaz de evitar o gasto duplo e garantia a integridade do sistema, por meio de PoW. Ele também introduziu o conceito da mineração, onde os nós da rede, receberiam uma quantidade de moedas em troca do poder computacional, oferecido por eles nas validações de transações (SZABO, 2005, tradução nossa).

No entanto, a primeira criptomoeda com grande sucesso e adoção de usuários foi o Bitcoin. Em 2008, sob o pseudônimo de Satoshi Nakamoto, um artigo foi publicado na Internet, o qual teve grande repercussão por resolver os impasses encontrados nos meios de pagamento eletrônicos.

Nakamoto, propôs uma rede *Peer-to-Peer* (P2P), capaz de solucionar o problema do gasto duplo, a qual formava um registro público de todas transações com um grande poder computacional, e não pode ser modificada, a não ser que toda PoW nela utilizada seja reescrita.

A rede foi chamada de *blockchain*, e consiste em uma corrente de blocos de transações, a qual usa diversas técnicas de criptografia para garantir sua segurança, transparência e imutabilidade. Nessa rede, o consenso é atingido pela solução com maior força computacional e não pelo voto da maioria de seus participantes. Os nós da rede, resolvem problemas matemáticos para participar do processo de validação das transações, deste modo, para um usuário mal-intencionado tomar seu controle, ele precisaria de um poder computacional superior ao restante da rede.

Sua proposta foi tão inovadora que resultou em diversas pesquisas para o aperfeiçoamento de sua arquitetura, e ao longo dos anos, uma variedade de

aplicações começaram a usá-la, como sistemas de votação, contratos inteligentes, armazenamento em nuvem e diversas novas criptomoedas foram criadas.

4.2 REDE

A rede do Bitcoin está estruturada sobre uma arquitetura *Peer-to-Peer*, neste modelo, como não há autenticação de usuários, os participantes, se comunicam diretamente, atuando ao mesmo tempo como clientes e servidores (GRECH; CAMILLERI, 2017, tradução nossa).

Um dos maiores problemas em sistemas distribuídos, é alcançar a integridade e confiança, e por meio do uso de funções *hash* e da rede P2P, Nakamoto, conseguiu garantir um de seus principais objetivos, a descentralização.

Os usuários se conectam à ela executando um determinado *software*, e é importante lembrar que a arquitetura da *blockchain*, somente permite a comunicação dos participantes, caso estejam executando a mesma versão de *software* (NAKAMOTO, 2008, tradução nossa). Alguns usuários, são conhecidos como nós completos, os quais armazenam todos os dados e registros de transações que já ocorreram no sistema (GRECH; CAMILLERI, 2017, tradução nossa).

O cliente utilizado na rede é o Bitcoin Core, originalmente desenvolvido por Satoshi Nakamoto, é um *software* de código aberto, e hoje recebe atualizações por um grupo de desenvolvedores voluntários (WANG; PUSTOGAROV, 2017, tradução nossa).

Um nó completo, executando o Bitcoin Core, é capaz de prover as funções de roteamento, armazenamento de dados, mineração e gerenciamento de carteiras. Outros nós armazenam somente uma parte da *blockchain*, mas para verificar uma transação, eles devem solicitar um conjunto de dados adicionais (ANTONOPOULOS, 2014, tradução nossa).

Uma característica comum de todos nós é armazenar a *mempool,* uma cópia local de todas transações ainda não inclusas a *blockchain*. Assim que uma transação é verificada e adicionada à *mempool,* ela então é transmitida a toda rede (MARTINS, 2018).

4.3 TRANSAÇÕES

Na rede Bitcoin, não existem exatamente o que é chamado de moeda, na verdade, existem somente transações (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa), as quais são estruturas de dados, que comprovam a transferência de valores entre usuários.

Uma transação, possui uma lista de uma ou mais saídas, as quais especificam o valor a ser realizado pela transferência, e uma lista de uma ou mais entradas, que detalham sua origem (BONNEAU et al., 2015, tradução nossa).

Essencialmente, as saídas de uma transação somente podem ser usadas uma única vez, e a tentativa de manipular uma delas pela segunda vez, é caracterizada como um gasto duplo, o qual será negado pelos nós da rede (ANTONOPOULOS, 2014, tradução nossa).

A *blockchain* mantém um conjunto de dados chamado *Unspent Transaction Outputs* (UTXO), o qual armazena todas as saídas da rede ainda não gastas por seus proprietários (VALLOIS; GUENANE, 2017, tradução nossa).

O saldo de um usuário somente existe na camada de aplicação, como uma forma de facilitar a sua utilização, na verdade, essa informação não faz parte do protocolo (BONNEAU et al., 2015, tradução nossa). As transações agregadas a um endereço são verificadas com o uso da UTXO, por meio das saídas ainda não utilizadas, e é dessa forma que um usuário pode fazer novas transações (DECKER; WATTENHOFER, 2013, tradução nossa), sendo assim, o saldo de uma carteira consiste no somatório dos valores encontrados na UTXO, que podem ser gastos pelo proprietário de sua chave privada.

Quando uma transação é realizada, todas moedas encontradas no campo de saída precisam ser consumidas, tendo em vista que sua utilização é permitida uma única vez. O problema é que nem sempre todas moedas disponíveis serão utilizadas em uma determinada saída, no entanto, como citado anteriormente, uma transação pode ter mais de uma saída e entrada, com isso, o usuário pode especificar a quantidade de moedas que ele deseja utilizar, e o restante delas, ele pode direcionar

para seu próprio endereço, e é desta forma que o conceito de troco é aplicado na plataforma (DECKER; WATTENHOFER, 2013, tradução nossa).

Outro problema frequente, são ocasiões em que um usuário não poderá utilizar uma entrada, com o valor total que necessita para realizar uma transação, porém, uma vez que ele tenha recebido várias transações com valores fracionados, ele pode adicionar múltiplas entradas para realizá-la.

Ainda, existem as transações conhecidas como *coinbase*, as quais são referentes a geração de novas moedas quando um bloco é formado (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

As transações *coinbase* possuem uma condição especial, onde suas recompensas não podem ser “gastas”, até que cem blocos após ela sejam validados, isso é uma forma de prevenção, para impedir que um minerador possa receber os valores das taxas e das novas moedas, geradas por um bloco que pode vir a ser considerado como um bloco *stale* (BITCOIN PROJECT, 2019, tradução nossa).

4.4 SCRIPTS

Para garantir a integridades das transações, os seus campos de entradas e saídas são acompanhados de um *script*. Apesar da ambiguidade, a linguagem utilizada no protocolo bitcoin, para codificação das transações é chamada de *Script*.

Ela é uma linguagem procedural, baseada em pilhas de dados, com sintaxe em Notação Polonesa Reversa, dessa forma ela é executada da esquerda para direita. Ela possui escopo limitado, necessita de pouco processamento e foi desenvolvida de forma simples, a fim de permitir sua execução nos mais diversos tipos de *hardware*s (ANTONOPOULOS, 2014, tradução nossa).

Ela também não atende as regras da completude de Turing, portanto, laços e controle de fluxos não são permitidos. Essas limitações garantem que laços infinitos ou outras formas de ataque, como o “*logic bomb*”, venham a ser executados, com o propósito de criar ataques de negação de serviço na rede (ANTONOPOULOS, 2014, tradução nossa).

Os *scripts* são encarregados pela validação de transações, para permitir que determinadas saídas possam ser utilizadas por seus proprietários. As saídas fazem uso de um *script* de bloqueio ou *scriptPubKey*, os quais exigem uma comprovação de posse de chave privada, para liberar o uso da transação, enquanto aqueles utilizados nas entradas, são conhecidos como *script* de desbloqueio ou *scriptSig*, os quais fornecem os dados para execução dos *scripts* de saída das transações (ANTONOPOULOS, 2014, tradução nossa; TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

Para validar transações, os *scripts* de saída e entrada são executados um após o outro. Para cada entrada, o *software* de validação recupera sua devida UTXO, a qual possui as condições necessárias para sua liberação. O *software* então executa o *script* de desbloqueio contido na entrada, utilizando a pilha de chamadas. Caso não encontrados erros na sua execução, a pilha principal é copiada e em seguida o *script* de bloqueio é executado. Eventualmente, se o resultado da execução com os dados da pilha for verdadeiro, ele atendeu as condições necessárias impostas pelo *script* de desbloqueio e dessa forma, a autorização para o uso da UTXO é válida. No entanto, se qualquer outro resultado permanecer na pilha de execução, ele é dado como inválido e não há autorização para o uso (ANTONOPOULOS, 2014, tradução nossa).

O Bitcoin Core vem constantemente atribuindo limitações aos *scripts*, a maioria delas é relacionada ao seu tamanho e ao das transações, e aos dados necessários em cada um de seus campos. A fim de garantir que as restrições sejam atendidas, o método *isStandard()* é executado. Atualmente, existem cinco *scripts* aceitos na rede do Bitcoin: *Pay-to-Public-Key* (P2PK), *Pay-to-Public-Key-Hash* (P2PKH), *Multi-Signature*, *Pay-to-Script-Hash* (P2SH) e OP_RETURN (SWARD; VECNA; STONEDAHL, 2018, tradução nossa).

A maioria dos mineradores, somente minera transações que atendem as restrições impostas pelo método *isStandard()*, porém, é possível criar um *script* que não segue trais regras, no entanto, é necessário encontrar um minerador que valide essa transação a um bloco.

O *script* mais utilizado para criação de transações é o P2PKH, o qual, fornece um endereço Bitcoin como destino, dessa forma, ele trava as moedas e

somente concede permissão de uso, para o usuário que possuir a chave privada relacionada ao endereço especificado (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa). Como prova de autenticação, o destinatário deve informar sua chave pública e uma assinatura de digital, gerada a partir do seu par de chaves, dessa forma, ele tem acesso a saída da transação.

Em uma transação P2PKH, de Alice para Bob, a saída pode conter um *script* no seguinte formato:

$$OP_{DUP}\ \ OP_{HASH160} < Endereço\ de\ Bob > \ OP_{EQUALVERIFY}\ \ OP_{CHECKSIG}$$

Caso Bob desejar utilizar as moedas recebidas por Alice, ele pode utilizar um *script* de desbloqueio conforme abaixo:

$$< Assinatura\ digital\ de\ Bob >< Chave\ pública\ de\ Bob >$$

Os operados utilizados na execução do *script* acima são descritos na tabela 3.

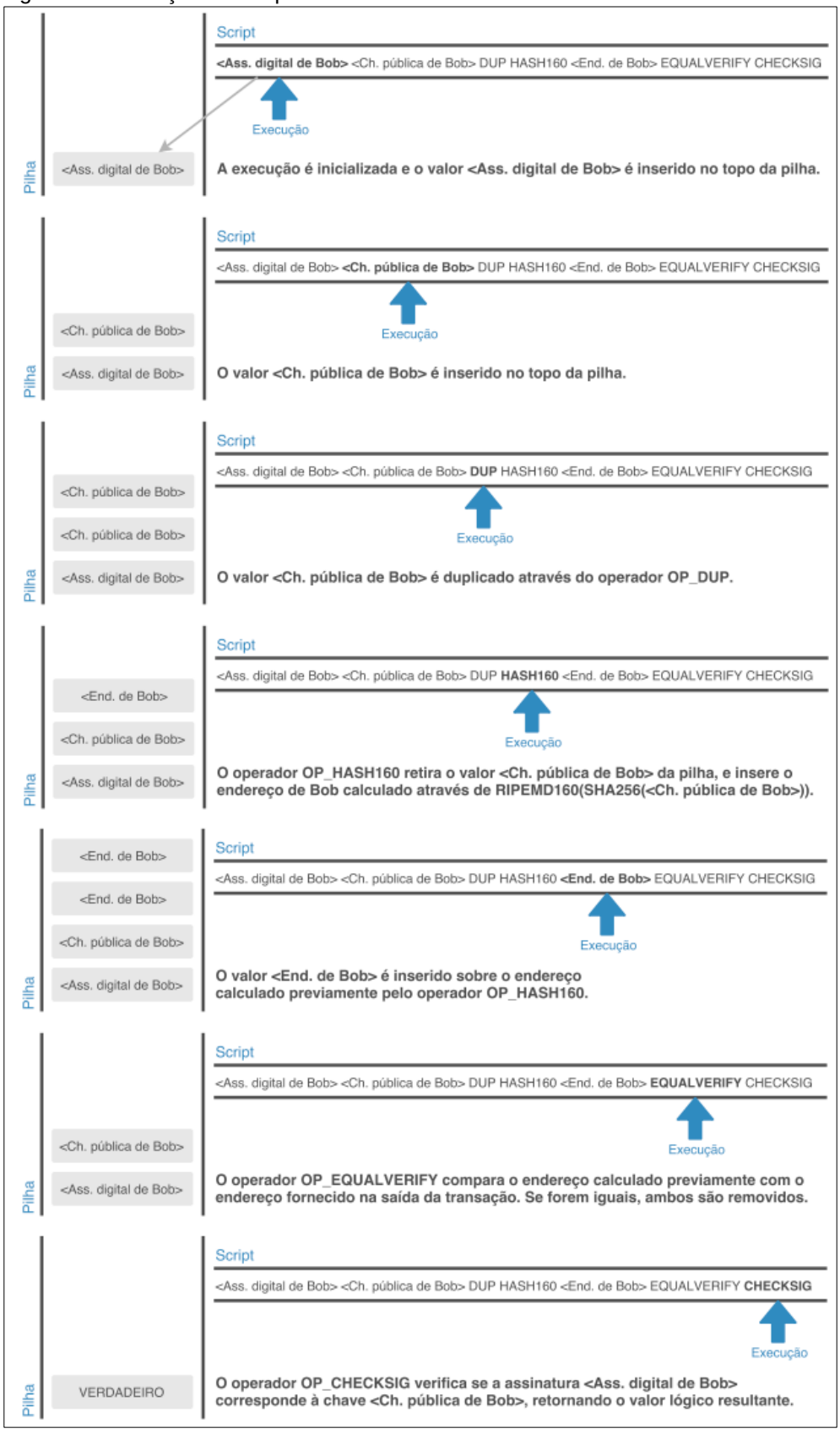
Tabela 3 - Operadores utilizados no P2PKH

Operador	Descrição
OP_CHECKSIG	Retira da pilha dois itens denominados $PubK$ e Sig, e é uma assinatura digital válida e correspondente à chave pública $PubK$. Insere na pilha o valor lógico resultante.
OP_DUP	Insere na pilha uma cópia do último valor inserido previamente.
OP_EQUALVERIFY	Retira e compara dois itens da pilha. Se forem iguais, a execução segue normalmente com os itens removidos da pilha. Se forem diferentes, a execução é interrompida.
OP_HASH160	Retira da pilha um item denominado $PubK$, e calcula $RIPEMD160(SHA256(PubK))$, obtendo o endereço Bitcoin correspondente à chave pública. Insere na pilha o resultado.

Fonte: Adaptado de Antonopoulos (2014).

Quando ambos são executados, a autorização de uso será retornada somente se a assinatura digital e a chave pública fornecidas por Bob, tenham compatibilidade com seu endereço, previamente fornecido por Alice. O passo a passo da execução de validação pode ser visualizado na figura 4.

Figura 4 - Validação do script P2PKH



Fonte: Martins (2018).

4.5 CHAVES, ENDEREÇOS E CARTEIRAS

Para receber e realizar transações, um usuário precisa de um endereço Bitcoin válido, o qual é derivado de sua chave pública.

O principal objetivo do endereço, é proporcionar um manuseio mais simples e seguro, sabendo que eles não são realmente necessários para realizar as transações, onde simplesmente, a chave pública pode ser utilizada diretamente. Porém, é muito recomendado evitar o uso constante de sua chave pública, ou de um mesmo endereço, tal prática permite a usuários maliciosos, realizarem ataques de comparação a assinaturas ou o rastreamento de moedas, portanto, é recomendável o uso de um novo endereço para cada transação (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

Existem *softwares* específicos para o gerenciamento das chaves e endereços de um usuário, os quais são conhecidos como carteiras, do inglês *wallets*. Eles não necessitam de acesso à internet e não são ligadas ao protocolo do Bitcoin, sendo assim, a segurança das chaves preservada por eles, depende totalmente de como um usuário as mantém.

Normalmente as carteiras são divididas em dois grupos: carteiras quentes e carteiras frias, do inglês *hot wallets* e *cold wallets,* respectivamente (MARTINS, 2018). Carteiras quente são conectadas à Internet, e com auxílio de uma interface de fácil utilização, permitem realizar pagamentos ou acompanhar o estado das transações, porém, algumas delas são mantidas por intermediários, os quais mantém a posse dos dados do usuário e consequentemente da sua carteira (GRECH; CAMILLERI, 2017, tradução nossa). Carteiras frias não se conectam a internet, com isso, elas oferecem um nível de segurança maior.

O primeiro passo para conseguir produzir um endereço, é a criação de uma chave privada k de 256 bits, a qual consiste e um número aleatório entre 1 e 2^{256}. O protocolo Bitcoin utiliza o padrão secp256k1 para a geração da chave pública K, onde uma curva elíptica é estabelecida sobre um campo finito de números primos. O padrão também estabelece um ponto G, o qual deve ser multiplicado pela chave privada e resulta em outro ponto na curva. O segundo ponto consiste na chave pública $K = k \cdot$

G. Mesmo com as chaves mantendo uma relação matemática, a chave pública somente pode ser obtida por meio da chave privada, garantindo uma forte segurança. Atualmente, os métodos mais eficientes para se calcular uma chave privada, a partir da chave pública, é pelo uso de força bruta, porém é um processo matematicamente impraticável e custoso (ANTONOPOULOS, 2014, tradução nossa).

Para criar um endereço, é calculado o *hash* da chave pública do usuário, primeiramente, por meio do algoritmo SHA-256 e o seu resultado é usado no algoritmo RIPEMD-160, após isso, um *checksum* (bit adicional, frequentemente utilizado para verificação de integridade de dados) é adicionado ao final do resultado, para evitar erros de digitação. Endereços Bitcoin são codificados com *Base58*, a fim de prevenir que caracteres venham a ser confundidos ou parecer ambíguos, quando exibidos em determinadas fontes (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

5 BLOCKCHAIN

Neste capítulo, serão tratados os conceitos utilizados na sua arquitetura, para garantir a segurança de seus registros. Serão abordadas informações sobre os blocos de transações e suas formas de conexão, bem como sua validação e as técnicas utilizadas para mantê-la como um sistema distribuído, porém, estão mais focadas em relação a rede do protocolo Bitcoin

5.1 HISTÓRICO

Apesar de ter recentemente se tornado o alvo de muitas pesquisas, seus experimentos se iniciaram no começo da década de 90, no entanto, somente em 2008, com a publicação do *white paper* do Bitcoin, realizada pelo pseudônimo Satoshi Nakamoto, a *blockchain* ganhou ampla adoção (GRECH; CAMILLERI, 2017, tradução nossa).

Nakamoto, introduziu ao mundo a primeira *blockchain* bem-sucedida, a qual tem o mesmo nome da primeira criptomoeda amplamente utilizada, o Bitcoin. Bitcoin também é o nome utilizada pelo protocolo da rede.

Ela consiste em um sistema distribuído, o qual permite que dados sejam registrados e compartilhados por todos seus participantes. Seus membros podem ou não, manter uma cópia total dela, no entanto, todos eles são responsáveis pela validação de novos dados inseridos.

Suas principais características são: descentralização, confiança, anonimato e transparência (ZHENG et al., 2017, tradução nossa).

5.2 BLOCOS

Ela é uma estrutura que armazena os dados na forma de transações, onde um grupo delas é conhecido como bloco. Eles são ligados uns aos outros, formando uma lista encadeada, indicando o *hash* referente ao cabeçalho do bloco anterior (CROSBY et al., 2016 , tradução nossa).

A cadeia é projetada, de tal forma que seja inviável realizar modificações em um determinado bloco, a não ser que todos anteriores a ele, também recebam devidas modificações (GRECH; CAMILLERI, 2017, tradução nossa).

O identificador de um bloco consiste em um *hash* de 32 bytes, obtido pela dupla execução do algoritmo SHA-256, no conteúdo encontrado em seu cabeçalho. Esse identificador não é armazenado no próprio bloco ou na rede, na verdade, ele é calculado e mantido como um metadado, com o propósito de facilitar sua indexação e consultas (ANTONOPOULUS, 2014, tradução nossa).

Um bloco pode ser dividido em dois campos, o cabeçalho e o corpo. O corpo, contém um contador e as transações, enquanto o cabeçalho possui o seguinte conjunto de dados (ZHENG et al., 2017, tradução nossa):

a) *block version*: contém o conjunto de regras usados no momento de sua validação;
b) *berkle root*: contém o *hash* de todas transações inclusas no bloco;
c) *timestamp:* data exata, no formato Unix, em que o bloco foi registrado;
d) *nBits ou d*ifficulty target*:* dificuldade alvo do algoritmo de prova-de-trabalho utilizada no bloco;
e) *nonce:* um contador utilizado para o cálculo do cabeçalho do bloco;
f) *previous block hash:* referência ao identificador do bloco anterior.

Na *blockchain*, os blocos são organizados no formato de uma árvore, os quais se iniciam a partir da “raiz” e vão em direção ao “galho” mais longo. O caminho mais longo, a partir do início, é a própria *blockchain* e ramificações encontradas nesse percurso não são legítimas e ainda podem estar armazenando informações que não fazem parte dela (VALLOIS; GUENANE, 2017, tradução nossa).

Cada bloco somente faz referência a um anterior a ele, no entanto, aquele encontrado na “raiz” é o único que esta regra não se aplica, este é o bloco *genesis,* o qual é conhecido por todos nós da rede e está diretamente implementado no software de cada usuário (DECKER; WATTENHOFER, 2013, tradução nossa).

Normalmente, os blocos podem ser classificados em quatro tipos: *normal block,* o qual pertence a rede assim como seu bloco antecessor; *invalid block,* são rejeitados no processo de validação e então deletados; *stale blocks,* os quais não

fazem parte da ramificação principal, no entanto, são válidos e seus pais pertencem a ela; *orphan block,* é um bloco válido, porém com um antecessor desconhecido, normalmente, ocorre quando não houve tempo suficiente para sua informação ser propagada pela rede (VALLOIS; GUENANE, 2017, tradução nossa).

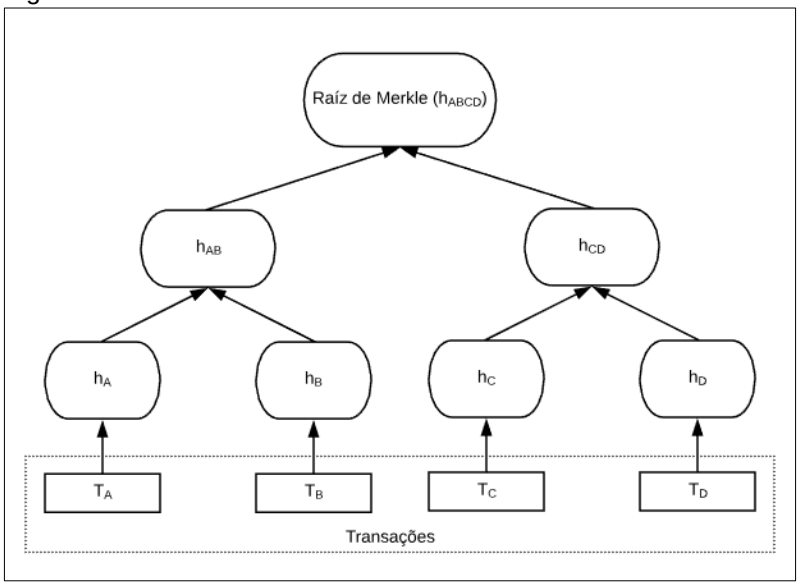
5.3 ÁRVORE DE MERKLE

Árvores de Merkle são árvores binárias, utilizadas para redução e verificação eficiente de grandes estruturas de dados (MERKLE, 1987, tradução nossa).

No protocolo Bitcoin, elas são utilizadas para gerar um *hash* final de um bloco, o qual é atribuído no seu cabeçalho. Para obtê-lo, o resumo de cada transação inclusa em um bloco deve ser calculado e são recursivamente concatenados até chegar a sua raiz, também chamada de Raiz de Merkle (VALLOIS; GUENANE, 2017, tradução nossa).

A figura 5 ilustra o processo de obtenção de sua raiz, onde T_A, T_B, T_C, T_D, representam as transações. Para calcular a Árvore de Merkle, a primeira etapa consiste no cálculo do *hash* h_x de cada transação utilizando duplamente o algoritmo SHA-256, formando as folhas da árvore. Como exemplo, para obter h_A, é calculada a função SHA-256(SHA-256(T_A)). Cada par de folhas origina um nó pai, onde são calculados por meio do mesmo processo, gerando um novo nível da árvore, como exemplo, h_{AB} é obtido com o cálculo SHA-256(SHA-256($h_A + h_b$)).

Figura 5 - Cálculo da Raíz de Merkle



Fonte: Do autor.

O processo é repetido até ser encontrada a Raiz de Merkle, no entanto, por ser uma árvore binária, ela deve ser balanceada, portanto, um número par de transações é necessário, porém, em blocos onde existe um número ímpar delas, o *hash* da última transação é duplicado (KARAME; ANDROUKALI, 2016, tradução nossa).

Por meio da raiz, um cliente é capaz de verificar se uma determinada transação pertence a um bloco, mesmo sem possuir uma cópia completa da *blockchain*, este processo é chamado de Verificação de Pagamento Simplificado. Nele, um cliente utiliza os cabeçalhos dos blocos e um número reduzido de nós da árvore, que são solicitadas a nós completos da rede (NAKAMOTO, 2008, tradução nossa).

Supondo que um usuário deseja verificar se uma transação X , pertence a um determinado bloco, ele deve solicitar a rede os nós essenciais para formar o caminho a partir de X até a raiz. Sendo assim, ele é capaz de realizar seu cálculo, e verificar se ela é a mesma encontrada no cabeçalho do bloco, com isso, caso elas sejam iguais, a transação X pertence a ele (NAKAMOTO, 2008, tradução nossa).

5.4 MINERAÇÃO

É o processo de adicionar novos blocos a *blockchain*, e consequentemente, também é o processo no qual novas moedas são criadas (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

Nos algoritmos de prova de trabalho, a mineração é necessária para validação das transações, onde elas somente são dadas como válidas, a partir do momento em que um minerador inclui ela a um bloco. Quando este processo é concluído, a construção do bloco se encerra e dessa forma todas transações nele inclusas, são consideradas como válidas.

O algoritmo PoW é baseado em valores monetários, portanto, mineradores são recompensados de duas formas, a primeira, é por meio da geração de novas moedas quando um bloco é formado, as quais o minerador recebe por direito, e a segunda é o valor total das taxas, inclusas em cada transação (BONNEAU et al., 2015, tradução nossa).

A geração das novas moedas é dada pela transação conhecida como *coinbase*, ela é a primeira transação inclusa em um bloco e as moedas geradas por ela, são automaticamente enviadas ao endereço de seu minerador (KARAME; ANDROUKALI, 2016, tradução nossa).

A quantidade de moedas neste tipo de transação se altera à medida que a *blockchain* cresce. A partir do bloco gênesis, cada novo bloco injetava 50 BTCs à rede, no entanto, a cada 210.000 novos blocos o valor é reduzido pela metade, esse fenômeno é conhecido como *halving.* Um bloco é aproximadamente construído em dez minutos, dessa forma, é possível estimar que este processo acontece a cada quatro anos (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

A geração de novas moedas decresce exponencialmente, a expectativa para que todas moedas tenham sido geradas é para o ano de 2140, neste momento haverá, aproximadamente 21 milhões de BTCs em circulação, este é o limite de moedas imposto pelo protocolo do Bitcoin. Apesar deste evento, a mineração ainda poderá ocorrer por meio das taxas atribuídas a cada transação, atualmente, elas não representam a maior parte do lucro obtido pelos mineradores, no entanto, a medida do crescimento do efeito de *halving*, elas se tornarão cada vez maiores e

consequentemente mais lucrativas para eles (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

Como citado anteriormente, a mineração é baseada no conceito de PoW, onde um usuário deve provar que realizou determinada tarefa, para receber uma recompensa. Em uma *blockchain*, tal tarefa consiste na solução de um desafio criptográfico, que deve ser computacionalmente custoso (VALLOIS; GUENANE, 2017, tradução nossa).

Usualmente, o desafio consiste em calcular o *hash* do cabeçalho do bloco, variando o valor do campo *nonce,* até que o resultado calculado atenda as condições impostas pelo protocolo da rede. No Bitcoin, o minerador calcula o *hash* utilizando duas vezes o algoritmo SHA-256, e condição que deve ser alcançada é um valor menor ao especificado no campo *difficulty target* (BONNEAU et al., 2015, tradução nossa). Esse desafio é computacionalmente complexo e difícil de ser resolvido, no entanto, uma vez encontrado o valor de *nonce* que atenda as condições, qualquer nó pode facilmente verificar seu resultado.

Quando um minerador alcança o valor esperado, ele deve enviá-lo a rede, onde todos nós são responsáveis por validá-lo, assim, caso o consenso da rede reconheça o valor, ele é adicionado a *blockchain* (ZHENG et al., 2017, tradução nossa). As novas moedas e as taxas de transações, são entregues ao primeiro minerador que concluiu o desafio criptográfico, e assim que o novo bloco é propagado pela rede, todos eles começam a trabalhar na construção dos blocos seguintes.

A dificuldade para solucionar o desafio é ajustada a cada 2016 blocos, em média, aproximadamente a cada duas semanas. O ajuste é necessário, para que a mineração de um bloco se complete em média a cada dez minutos, e caso gerado em um tempo inferior a este intervalo médio, significa que o poder de processamento da rede está muito alto, e dessa forma a dificuldade deve ser ajustada (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

5.5 CONSENSO E FORKS

Caracterizada pela descentralização, a *blockchain* possui o evento conhecido como *fork,* que pode ocorrer ocasionalmente, quando duas versões da rede são “vistas” pelos nós.

Eles são provocados principalmente por duas razões: quando mineradores validam um bloco simultaneamente ou quando um bloco válido no estado de *stale*, tem uma lenta propagação pela rede (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

Em ambos casos, dois blocos válidos passam a ocupar uma mesma posição. A solução para tal problema é simples, ao tratar-se de PoW, a mineração continua na “versão mais longa” da *blockchain*, que possui maior força computacional sendo empregada.

Em dado momento, um *fork* irá sobrepor o outro, isso devido aos blocos validados serem propagados a rede, no entanto, os nós naturalmente validam o primeiro que encontram (BITCOIN PROJECT, 2019), dessa forma, ocasionalmente um dos *forks* será “mais forte” que o outro. Nesse caso, os dois ramos da *blockchain* possuem diferente dificuldades de mineração, com isso, os nós tendem a seguir ao que possui a maior dificuldade (BONNEAU et al., 2015, tradução nossa).

Os blocos presentes no menor *fork* continuam salvos, contudo, deixam de fazer parte da *blockchain*. Eles não são apagados pois podem ser “pais” de alguns dos blocos da ramificação principal, ou ainda, mineradores podem eventualmente vir a adicioná-los a ela (BONNEAU et al., 2015, tradução nossa).

O *software* usado na rede possui determinada especificações as quais devem ser seguidas, dessa forma, todos os nós seguem o mesmo consenso. No entanto, em determinadas situações, quando novas regras são propostas, alguns deles as aceitam, atualizando seu *software*, e outros, continuam com sua antiga versão. Com isso, novos blocos seguindo as regras do novo consenso, são aceito por nós atualizados e rejeitados por aqueles não atualizados, o mesmo acontece para os blocos que não seguem as novas regras, onde são aceitos pelos não atualizados e rejeitados pelos que utilizam a nova versão do *software*, este fenômeno é chamado de *hard fork*, o qual implica em duas versões diferentes da *blockchain* (BITCOIN PROJECT, 2019; BONNEAU et al., 2015, tradução nossa).

Em uma *blockchain* também é possível encontrar um *soft fork*, onde, a maioria dos nós passam a usar uma versão atualizada das regras, portanto, mesmo que existam blocos utilizando regras com versões antigas, eles ainda serão aceitos pelos nós atualizados. Em determinadas circunstâncias, blocos seguindo regras atualizadas, serão aceitos por nós seguindo regras antigas do consenso (BITCOIN PROJECT, 2019; BONNEAU et al., 2015, tradução nossa). Normalmente, *soft forks* acontecem quando pequenas mudanças são propostas ou testadas pela rede.

5.6 TIPOS DE BLOCKCHAIN

Atualmente ela pode ser dividida em três grupos: públicas, privadas e *consortium*. Suas principais diferenças, são as formas em que seus usuários podem participar da rede (TURKANOVIć et al., 2018, tradução nossa).

Em *blockchains* públicas, todos os dados são visíveis e qualquer usuário pode participar no consenso da rede. Ao tratar-se de uma versão *consortium*, somente usuários selecionados podem participar do consenso da rede. E quanto às privadas, normalmente, somente usuários que participam de uma mesma organização, são selecionados para fazer parte do consenso da rede, no entanto, isso traz uma característica única, tornando-a um sistema centralizado, sendo que somente uma organização fica com seu total controle (ZHENG et al., 2017, tradução nossa). A tabela 4 indica as principais diferenças entre os tipos de *blockchain*.

Tabela 4 - Comparação das características entre os tipos de *blockchain*

Propriedade	***Pública***	***Consortium***	***Privada***
Determinação de Consenso	Todos nós	Somente nós selecionados	Organização proprietária
Permissão de leitura	Pública	Pública ou restrita	Pública ou restrita
Imutabilidade	Imutável	Podem ocorrer fraudes	Podem ocorrer fraudes
Eficiência	Baixa	Alta	Alta
Centralização	Descentralizada	Parcial	Centralizada
Participar do Consenso	Sem permissão	Necessita permissão	Necessita permissão

Fonte: Adaptado de Zheng et al. (2017, tradução nossa).

De acordo com Zheng, as principais diferenças e características entre os tipos de *blockchain* podem ser vistos como:

a) determinação do consenso: nas públicas, cada nó faz parte do consenso, nas privadas, o controle é realizado por uma organização e uma *consortium*, pode ser considerada como parcialmente controlada, uma vez que somente nós selecionados fazem parte dela;
b) permissão de leitura: em *blockchains* públicas as transações são visíveis a todos, enquanto nos outros modelos, isso pode variar;
c) imutabilidade: uma vez que todos dados da rede são armazenados em um grande número de nós, é praticamente impossível que transações venham a ser fraudadas em *blockchains* públicas. Diferentemente, transações nas redes *consortium* ou privadas, podem ser mais facilmente fraudadas, sabendo que existem um número limitado de participantes;
d) eficiência: a propagação de dados e blocos pode ser demorada ou prolongada nas versões públicas, onde existem uma grande quantidade de nós, dessa forma, as transações são limitadas e sua latência é alta. No entanto, com poucos nós, os outros modelos, podem oferecer uma maior velocidade no registro de transações e blocos;
e) centralização: normalmente a descentralização é vista como uma das maiores características, no entanto, isso só pode ser alcançado com uma *blockchain* pública;
f) consenso: qualquer usuário pode participar no consenso no modelo público, entretanto, nos outros tipos, para participar, um nó necessita de autorizações.

5.7 ALGORITMOS DE CONSENSO

O consenso da *blockchain* é alcançado por meio de nós executando uma mesma versão de *software*, os quais, seguem determinadas regras para validação das transações e dos blocos. O protocolo do bitcoin utiliza o algoritmo de PoW, contudo, ele não é o único que pode ser utilizado para alcançar o consenso entre os nós da rede (TURKANOVIć et al., 2018, tradução nossa).

Alguns dos algoritmos mais comuns são: *Proof-of-Work*, *Proof-of-Stake* (PoS), *Delegated-Proof-of-Stake* (DPoS), *Proof-of-Importance*, *Proof-of-Activity*, *Proof-of-Burn*, *Proof-of-Deposit* (TURKANOVIć et al., 2018, tradução nossa; ZHENG et al., 2017, tradução nossa). Atualmente os mais utilizados são: PoW, PoS e DPoS.

O PoW, exige grande poder computacional, para solução de desafios criptográficos, onde os mineradores da rede são responsáveis por este processo. Uma vez que um minerador soluciona o desafio, ele propaga o resultado para a rede, e todos os nós o verificam, desta forma o consenso é alcançado (ZHENG et al., 2017, tradução nossa)

O maior problema encontrado nesse tipo de algoritmo, é o custo de operação, onde é exigido *hardware* com alta performance, os quais, normalmente, executam funções *hash* repetidamente.

A princípio, ele era executado por unidades centrais de processamento (CPU), porém, os mineradores, perceberam que esse tipo de *hardware* não era tão eficiente para execução de inúmeras tentativas de solucionar o desafio criptográfico. Pelo fato de processar operações envolvendo computação paralela, a mineração passou a ser executada por unidades de processamento gráfico (GPU), as quais são capazes de executá-las mais rapidamente, e com menor consumo de energia, comparado as CPUs. Contudo, à medida que a dificuldade de mineração aumentou, a eficiência de GPUs passou a se tornar obsoleta (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

Para contornar este problema, os mineradores passaram a usar os circuitos integrados de aplicação especifica, do inglês A*pplication Specific Integrated Circuits* (ASIC), os quais são máquinas dedicadas para execução de funções *hash*, no entanto, possuem alto consumo de energia e seu preço é alto, além disso, uma ASIC dedicada a minerar BTCs, não pode ser usada para mineração, por exemplo, da criptomoeda Ethereum, uma vez que o protocolo de ambas moedas executam diferentes funções *hash* (TSCHORSCH; SCHEUERMANN, 2016, tradução nossa).

À medida que a dificuldade de mineração cresce, ou até mesmo, quando uma criptomoeda alcança seu número máximo de moedas em circulação, o algoritmo

de prova de trabalho torna-se menos eficiente, e uma alternativa para solucionar esse problema é o uso do algoritmo PoS.

Ao contrário do PoW, em que há uma disputa pela mineração, nele, o nó responsável pela validação de um bloco, é escolhido de acordo com sua “participação” na rede. O algoritmo é executado da seguinte forma, um usuário deposita uma quantidade mínima de moedas e assim está participando de um determinado grupo de nós, onde um deles, será randomicamente escolhido para realizar a próxima validação. O PoS, não exige que funções *hash* sejam executadas para validação dos blocos, e sim, somente que o usuário escolhido o assine e assim o tornando válido. O que garante a integridade do sistema, é caso um usuário malicioso tente validar blocos ao quais não foi autorizado, ele perderá todas as moedas depositadas inicialmente. O maior problema desse algoritmo, é quando usuários com maiores quantidades de moedas controlaram a rede, onde possuem mais chances de serem escolhidos para o processo de validação (TURKANOVIć et al., 2018, tradução nossa).

Algumas versões, baseadas no *Proof-of-Stake,* foram criadas para contornar o problema de monopolização da rede, o mais comum entre eles é o DPoS. Nesse tipo de algoritmo, os usuários delegam os responsáveis para validar os blocos. Com um menor número de nós, a validação é muita mais rápida comparada aos outros protocolos, além disso, a segurança da rede também é favorável (ZHENG et al., 2017, tradução nossa).

Nas próximas seções, serão realizadas, uma breve descrição, a respeitos dos métodos para inserção de dados na *blockchain* do Bitcoin, assim como, seus custos e limitações.

5.8 INSERÇÃO DE DADOS NA BLOCKCHAIN

Atualmente, existem inúmeras ferramentas online com capacidade de registrar dados na *blockchain* do Bitcoin. De acordo com Antonopoulus (2014), o armazenamento de dados, não relacionados as transações, é algo muito controverso, uma grande quantidade de usuários e desenvolvedores, veem isso como uma forma

abusiva do uso de sua capacidade, no entanto, muitos acreditam que é uma forma de expor seu potencial e usabilidade. Não existe um meio de prevenir que usuários, venham usá-la para armazenar dados arbitrários, contudo, uma possível solução, seria somente aceitar *scripts* que tornariam o processo muito mais custoso, ainda assim, não seria possível impossibilitar o armazenamento dos dados (SWARD; VECNA; STONEDAHL, 2018, tradução nossa).

O maior problema causado por esta prática, é a forma como os tipos de *scripts* são usados. As primeiras aplicações propostas, utilizavam o *script* de desbloqueio das transações, inserindo informações no campo reservado para o preenchimento do endereço do destinatário, capaz de armazenar somente 20 bytes. Esta era uma solução ineficaz, devido a armazenar uma pequena quantidade de dados e principalmente por criar transações com saídas inválidas.

Todas transações, são inseridas no conjunto de todas as saídas não gastas na rede, a UTXO, contudo, quando não possuem um endereço válido, não é possível “gastar” as transações com esse tipo de saída, causando uma sobrecarga na rede, onde inúmeras transações como essas são armazenadas, este fenômeno é conhecido como *blockchain bloat* (ANTONOPOULOS, 2014, tradução nossa).

Apesar dos diversos métodos, para o armazenamento de dados na *blockchain*, o foco deste estudo é relacionado ao método OP_RETURN, o qual é capaz de armazenar até 80 bytes por transação e possui um custo muito pequeno comparado aos outros existentes (SWARD; VECNA; STONEDAHL, 2018, tradução nossa), além disso, as plataformas mais conhecidas para o registro de diplomas como prova de existência, utilizam este mesmo método.

O OP_RETURN foi inserido na versão 0.9 do Bitcoin Core, devido ao crescente uso do *script* P2FKH para inserir dados, o qual gerava saídas inválidas e armazenadas na UTXO (SWARD; VECNA; STONEDAHL, 2018, tradução nossa). Ele é um operador, que pode ser utilizado nas saídas das transações e permite aos desenvolvedores adicionarem informações diversas, não relacionadas diretamente a transação (ANTONOPOULOS, 2014, tradução nossa).

O operador cria uma saída que explicitamente não pode ser gasta, desta forma, ela não é inserida na UTXO. Seu uso cria um *script* de entrada, que não

corresponde a nenhum *script* de bloqueio, sendo assim, sua utilização impossibilita o uso das moedas atribuídas a saída da transação, porém, é possível criar uma saída com nenhum valor atribuído a ela. O método de padronização das transações, verifica se as saídas contêm apenas um destes operadores, caso ela não atende a essa regra, a transação é dada como inválida.

5.9 LIMITAÇÕES E CUSTOS

A quantidade de dados que podem ser inseridos em uma transação varia de acordo com *script* utilizado, no entanto, todos eles não fornecem uma forma de armazenar grandes quantidades de dados. A tabela 5, mostra a quantidade em bytes que os *scripts* mais comuns podem armazenar.

Tabela 5 – Tamanho máximo do *script* e custo (com uma taxa de 20 satohis/byte)

Script	Quantidade*	Custo**
P2PKH	58,680	.03601624
P2PK	85,280	.02715132
OP_RETURN	80	.00006340
P2MS	92,624	.02522220
P2SH	62,340	.03701302

*Quantidade em bytes **Custo em Bitcoin

Fonte: Adaptado de Sward; Vecna; Stonedahl (2018, tradução nossa).

Com o operador OP_RETURN, é possível inserir dados limitados até a 80 bytes. O protocolo do bitcoin também limita cada bloco à 1 MB (BONNEAU et al., 2015, tradução nossa), dessa forma o tamanho de uma transação normalmente varia entre 250 bytes e 500 bytes.

Outra limitação que ocorre na rede, é o número de transações processadas, sendo possível estimá-lo em sete novos registros a cada segundo. Este é um número relativamente pequeno e muitas vezes é dito como ineficiente, quando comparado ao processamento de transações de outros meios de pagamento (BONNEAU et al., 2015, tradução nossa; GRECH; CAMILLERI, 2017, tradução

nossa), como por exemplo, a rede de pagamentos Visa, tem a capacidade de processar 24000 transações por segundo (VISA, 2019, tradução nossa).

Sua capacidade de processamento pode ser ajustada por meio de melhorias, as quais vem sendo propostas ao protocolo constantemente, contudo elas precisam ser aceitas pela maioria dos nós da rede. Usualmente, elas estão relacionas ao tamanho dos blocos e redução do intervalo de tempo entre sua geração, porém, elas implicam diretamente nas taxas e dificuldades de mineração, ou ainda, necessitam de alterações extremamente complexas na rede, as quais podem afetar diretamente sua segurança (MARTINS, 2018).

A inserção de dados na *blockchain* é realizada por meio de transações, dessa forma, seu custo está diretamente relacionado ao valor da moeda e as taxas de transação entregues aos mineradores. No momento em que o usuário realiza a transação, ele pode escolher qual o valor da taxa, porém, valores muito baixos normalmente são “ignorados” pelos mineradores, dessa forma seu processamento fica sendo adiado até que um minerador venha validá-lo. O valor adequado para a taxa é variável, e ao mesmo tempo, é relativo ao volume de transações sendo processados pela rede.

Usuários podem estimar o tempo de processamento de uma transação, de acordo com o valor da taxa escolhida, por meio de ferramentas online, como a Bitcoin Fees (BITCOIN FEES, 2019).

5.10 BLOCKCHAIN E EMISSÃO DE DIPLOMAS

A infraestrutura fornecida pela tecnologia da *blockchain*, é ideal para armazenar, compartilhar e verificar qualquer tipo de registro. Com ela, o sistema de chaves pública e privada, substituem a necessidade de uma entidade intermediadora para o controle e administração da rede, além do mais, toda ela é reforçada pela descentralização, garantindo maior robustez e segurança.

A descentralização é a razão pela qual a rede possui maior longevidade, devido aos nós que possuem uma cópia de todo histórico de transações, atualmente a *blockchain* do Bitcoin possui mais de 9400 nós (BITNODES, 2019), e com essa

numerosa quantidade de cópias, qualquer possível falha em um dos nós, não promove qualquer fraqueza ou implica na perda de dados. Além do mais, nenhum registro pode ser alterado ou apagado, a menos que toda PoW já realizada seja refeita a partir do início. Outra grande vantagem para esse tipo de sistema, é o fato de que uma rede pública, pode ser acessada por qualquer usuário sem maiores intervenções, e qualquer *software* para registro de arquivos, pode ser usado para qualquer documento independente dos padrões utilizados nele (GRECH; CAMILLERI, 2017, tradução nossa).

Quanto ao registro de diplomas, ele pode ser resumido da seguinte forma: a *blockchain* mantem o seu registro, com a instituição emissora e seu condecorado. O arquivo digital pode ser mantido pelo usuário, em qualquer *hardware* de armazenamento, e até mesmo impresso em papel caso necessário.

O passo a passo para emissão dos diplomas é simples, este processo começa com a criação de um arquivo digital, que contém as informações necessárias para o diploma (Seção 2.2), as quais podem adotar o formato desejado, no entanto, utilizar um padrão é recomendável, e uma vez criado o arquivo, o processo para o registro diretamente na *blockchain* se inicia, onde o emissor deve criar o *hash* do arquivo, criando uma identificação única, e finalmente, usar sua chave privada para cria uma transação, armazenando o resumo com o método escolhido, e uma vez validada, o processo se finaliza.

Os dados presentes no diploma ainda são verificáveis individualmente, mas é necessário lembrar que o mais importante deles, é a chave pública da própria instituição. A verificação do documento é realizada ao comparar seu *hash,* com aquele publicado na *blockchain*, caso forem iguais, o documento é autêntico.

O uso desta tecnologia para o registro diplomas não é somente uma oportunidade de garantir a segurança necessária para os dados, assim como, é uma forma de enriquecer o atual ambiente de certificação digital.

Como formato de padronização, o uso *Open Badges* é de grande importância, ele é eficiente e vem sendo aceito e reconhecido por instituições acadêmicas prestigiosas (GRECH; CAMILLERI, 2017, tradução nossa).

O objetivo de autenticar diplomas em uma *blockchain*, é trazer a versatilidade ao documento, o qual o estudante usualmente mantém em formatos físicos e é dependente de intermediadores, para assegurar sua verificação e segurança, e ao mesmo tempo transformá-lo em um arquivo digital, o qual pode ser consultado em um banco de dados descentralizado e distribuído ao redor do mundo todo, caracterizado pela imutabilidade, sem necessidade de intermediadores para seu acesso e posterior verificação.

5.10.1 Diplomas digitais usando blockchain

O uso da *blockchain* permite uma nova infraestrutura para assegurar, compartilhar e verificar certificações de aprendizados (SMOLENSKI, 2016, tradução nossa).

Certificados digitais usando a *blockchain* possuem vantagens em relação aos demais, sendo elas (GRECH; CAMILLERI, 2017, tradução nossa):

a) eles não podem ser forjados, é possível verificar com total certeza a quem e por quem o certificado foi emitido;
b) a verificação pode ser realizada por qualquer pessoa com acesso a *blockchain*, utilizando *softwares* gratuitos ou ainda ferramentas online;
c) como não existe necessidade de uma entidade intermediadora para validar o certificado, ele pode ser verificado mesmo se a instituição emissora não mais existir;
d) os dados armazenados somente podem ser apagados se todos os nós completos sejam comprometidos;
e) o *hash* publicado pode ser visto somente como uma forma de referenciar o documento original, desta forma, existe um nível de privacidade ainda mais privilegiado aos usuários.

6 TRABALHOS CORRELATOS

Ao decorrer do levantamento bibliográfico, foram realizadas pesquisas no âmbito nacional e internacional de projetos e trabalhos que abordam temas similares aos abordados neste estudo. A partir deles, buscou-se identificar aqueles que tinham maior relação com o objetivo deste projeto e que serviriam como instrumento de estudo para a elaboração do trabalho.

6.1 BLOCKCERTS

É uma plataforma de padrão aberto, capaz de criar, compartilhar, visualizar e verificar certificados utilizando a *blockchain*, os quais são assinados por meio de criptografia e garantem sua segurança. A meta da plataforma é permitir aos usuários obter a propriedade de seus certificados, e os tornar aptos a compartilhar seus documentos de forma instantânea, sem necessitar de nenhuma entidade intermediadora para garantir sua autenticidade (BLOCKCERTS).

O projeto foi realizado pelo MIT Media Lab e pela empresa *Learning Machine*. A comunidade pretende promover sua adoção a um nível global, como forma de padrão para emissão e registro de dados com o uso da *blockchain*.

O padrão proposto permite a qualquer usuário, incluindo estudantes, instituições de ensino e até mesmo órgãos governamentais, o uso do seu código que segue o formato *Open Source*. Ele foi desenvolvido de forma a manter os certificados por meio do padrão *Open Badges* (OPEN BADGES), projetado pela fundação *Mozzila*, dando garantia de confiabilidade aos usuários, aos certificados, ou até mesmo, garantindo os requisitos mínimos para as instituições que os emitem. A comunidade tem crescido, como pode ser observado em seu fórum, onde existe um crescente número de instituições contribuindo e desenvolvendo suas próprias plataformas (GRECH; CAMILLERI, 2017, tradução nossa).

O *Blockcerts* tem suporte as redes, tanto do Bitcoin, quanto do Ethereum, mas a comunidade pode contribuir com outras implementações.

6.2 UNIVERSIDADE DE NICOSIA

Em 2014, a Universidade de Nicosia (UNIC) se tornou a primeira instituição de ensino a emitir certificados acadêmicos, permitindo sua verificação por meio da *blockchain* do Bitcoin. Ela também foi a primeira a explorar o potencial da tecnologia junto à educação, onde passou a aceitar BTC como forma de pagamento para qualquer um de seus cursos de graduação, assim como, foi a primeira a criar um curso de nível universitário sobre criptomoedas, e a oferecer um curso de mestrado em de Moedas Digitais (GRECH; CAMILLERI, 2017, tradução nossa).

Quando a universidade introduziu esses cursos, ela passou a aceitar o BTC como forma de pagamento de seus estudantes, com o intuito de mostrar o comprometimento e interesse da instituição com a tecnologia das criptomoedas. Este foi o grande impulso, para o desenvolvimento de sua própria plataforma para o registro de diplomas com a *blockchain* (UNIC BLOCKCHAIN INITIATIVE).

Em 2015, a UNIC criou seu próprio time de desenvolvimento para emitir e autenticar certificados, utilizando o código *Open Source* disponibilizado pelo *Blockcerts*. Os testes de um sistema para publicar todos os diplomas se iniciaram em junho de 2017, e a partir de outubro do mesmo ano, todos os diplomas passaram a ser emitidos no novo sistema. A universidade também se comprometeu com a criação de *softwares* para o compartilhamento e verificação dos diplomas (GRECH; CAMILLERI, 2017, tradução nossa).

6.3 MIT MEDIA LAB

Em 2015, a universidade *Massachusetts Institute of Technology* (MIT), e seu departamento de pesquisas MIT Media Lab, se juntaram ao desenvolvimento da plataforma *Blockcerts*, e passaram a estudar a adoção da tecnologia dentro da universidade. Neste processo, o MIT conclui que por meio da *blockchain*, havia um maior controle sobre os diplomas, onde foi eliminada a necessidade de intermediadores para a verificação e de funcionários para o armazenando e sua validação (SCHMIDT, 2015, tradução nossa).

Para obter mais dados e conhecimento na área, em 2017, o MIT em colaboração com a empresa *Learning Machine*, criou um projeto teste utilizando o código provido pelo *Blockcerts*, e passou a emitir os diplomas de dois de seus cursos de graduação, com isso, eles foram capazes de identificar os fatores necessários para o desenvolvimento de uma plataforma em larga escala. Ainda assim, o principal motivo do desenvolvimento do projeto de testes foi receber *feedback* dos alunos, a fim de monitorar a relação entre eles e a nova plataforma. Sua principal preocupação, foi perceber se os alunos identificavam as vantagens obtidas por meio do novo sistema, onde um grande nível de segurança foi fornecido, garantindo a eliminação do risco de fraude e da necessidade de interação com intermediadores (GRECH; CAMILLERI, 2017, tradução nossa).

De forma simples, o sistema proposta pelo MIT funciona da seguinte forma: a primeira etapa é a criação de um arquivo digital contendo as informações necessárias, após isso, ele é assinado digitalmente utilizando a chave privada, somente em posse da universidade, o próximo passo, consiste em criar o *hash* do diploma e finalmente ele é registrado na *blockchain*. O projeto implementado pelo MIT, ainda permite verificar por quem e para quem o documento foi emitido, e a validação do conteúdo do diploma (SCHMIDT, 2015, tradução nossa).

6.4 INSTITUIÇÕES DE ENSINO EM MALTA

O país de Malta é um dos pioneiros no desenvolvimento governo eletrônico utilizando *blockchain* e a partir de 2016, o governo do país passou e pesquisá-la como um modelo de adoção nacional para educação.

O projeto vem sendo desenvolvido de forma a adotar credenciais acadêmicas e certificações profissionais, com uma solução para criação, emissão, visualização e verificação certificados. O governo pretende utilizar uma *blockchain* pública, com a intenção de criar uma escalabilidade e flexibilidade maiores. O projeto da plataforma consta com ferramentas necessárias para criação visual dos diplomas e seu compartilhamento (GRECH; CAMILLERI, 2017, tradução nossa).

6.5 BLOCKCHAIN E EDUCAÇÃO NO BRASIL

6.5.1 Universidade Federal de Santa Catarina

Membros da Universidade Federal de Santa Catarina (UFSC), propuseram a emissão de diplomas utilizando a *blockchain* em 2019, como uma solução capaz de prover segurança. Sua proposta, é que a emissão se inicie assim que os estudantes concluam seus cursos, além disso, a emissão de um registro acadêmico, com todo histórico escolar do estudante, é inclusa ao arquivo principal. O projeto ainda tem uma visão de futuro, onde são inclusos todo perfil estudantil do aluno, a partir de suas escolas elementares até seu nível de ensino superior (PALMA et al., 2019, tradução nossa).

Diferentemente deste estudo, o protótipo criado por Palma et al., utiliza *Smart Contracts* e o protocolo do Ethereum. Esta escolha foi feita devido aos inúmeros trabalhos correlatos que utilizam o Bitcoin para o registro de dados arbitrários e devido a muitos membros de sua comunidade considerarem isso como uma prática indevida, assim como, promover e encorajar pesquisadores, a utilizarem novas tecnologia neste ambiente.

Nesta plataforma a emissão dos diplomas não é responsabilidade da universidade, e sim dos *Smart Contracts*, os quais podem ser vistos como uma espécie de contrato auto executável que fica armazenado na *blockchain*.

6.5.2 Universidade Federal da Paraíba

A Universidade Federal da Paraíba (UFPB), se tornou a primeira instituição de ensino superior do Brasil, a emitir diplomas na *blockchain*. No início de 2019, os alunos formandos dos cursos de Ciência da Computação e Engenharia da Computação, foram os primeiros a receber seus diplomas na plataforma (RNP, 2019).

A solução desenvolvida se chama GT-RAP - Serviço de Registro, Autenticação e Preservação Digital de Documentos, e foi financiado pela Rede

Nacional de Ensino e Pesquisa, o projeto foi coordenado por acadêmicos e docentes da própria universidade (COSTA et al., 2018).

De acordo com seus criadores, o projeto surgiu como uma solução tecnológica capaz de suprir a crescente demanda, nos meios universitários, por maior segurança na emissão de diplomas e documentos.

Os estudos se iniciaram em 2015, e a pesquisa foi realizada de forma a emitir os documentos, nas *blockchains* do Bitcoin e Ethereum (COSTA et al., 2018), no entanto, não existem informações atuais sobre qual plataforma vem sendo utilizada.

7 TRABALHO DESENVOLVIDO

Com base no conhecimento adquirido durante a pesquisa e com o uso do material teórico coletado, neste capítulo, serão descritas todas as etapas necessárias para o desenvolvimento, de um protótipo de *software*, capaz de se comunicar com a *blockchain* do protocolo Bitcoin, com a finalidade de registrar e verificar documentos nela inseridos, portanto, criando uma prova de sua existência.

Também serão descritas as características analisadas, como forma de escolha, na adoção do tipo de modelo rede para o desenvolvimento do projeto.

Como resultado, foi projetado uma aplicação web com as tecnologias *Node.JS* e *Express* e para melhor manuseio e confiança dos dados, também foi implementada uma camada de segurança, com objetivo de criar perfis de usuários, como garantia de restringir o acesso total as funcionalidades do protótipo, inserindo permissões, por conta adicionada.

7.1 METODOLOGIA

Para a elaboração deste projeto, o primeiro passo tomado foi o levantamento do material bibliográfico, necessário para o entendimento das tecnologias envolvidas, tendo como maior foco a *blockchain*.

O material encontrado e utilizado foi por meio de livros, teses, publicações da Internet e em repositórios de trabalhos acadêmicos, contudo, teve como principal fonte de pesquisas as bases de dados: Google Acadêmico, *IEEE Xplore* e *Research Gate*, por meio de artigos científicos.

Lida a documentação encontrada, passou-se a escrever o referencial teórico, apresentando os principais pontos e assuntos correlacionados ao objetivo deste trabalho.

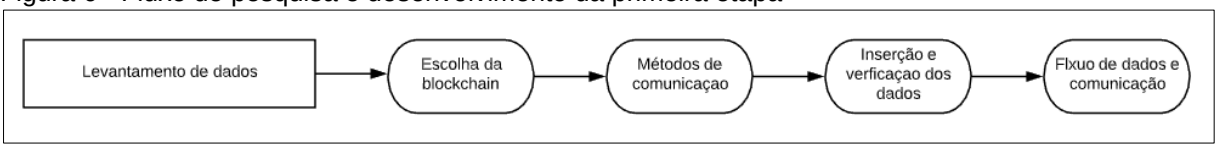
Também foram descritas as características de cada tipo de *blockchain* e suas respectivas qualidades e atributos, tendo em vista sua importância na tomada de decisão na escolha de sua adoção. Esse levantamento, também implica na forma

de manipulação dos dados e requisitos mínimos para caracterização de um projeto de prova de existência.

O processo de desenvolvimento foi dividido em três etapas, onde, durante a primeira, foram levantadas as informações para a aplicação (figura 6), o qual seguiu-se da seguinte forma:

a) escolha da *blockchain*;
b) formas de comunicação com a rede;
c) métodos de inserção e verificação de dados;
   definição do fluxo de dados.

Figura 6 - Fluxo de pesquisa e desenvolvimento da primeira etapa



Fonte: Do Autor.

No decorrer da segunda etapa, foram realizados testes com a biblioteca de abstração *bitcoinjs-lib* e com as API's de comunicação: *ChainPoint*, *Blockcypher* e *Smartbit*, devido a necessidade de encontrar uma plataforma, com integridade e garantia na manipulação e envio dos dados, e que possui suporte a sua busca.

Por fim, na terceira etapa do projeto, foi realizado seu desenvolvimento e respectivos testes com o protótipo desenvolvido, tendo como objetivo, obter sucesso em todos os passos necessários para efetivar um registro na *blockchain*, sendo eles, a criação de uma carteira, transformar um arquivo em resumo, criar uma transação com o *hash* e a carteira selecionados, e então enviar os dados a *blockchain*.

7.2 FERRAMENTAS E RECURSOS UTILIZADOS

Para o desenvolvimento da aplicação e código fontes, o *software Visual Studio Code*[1] foi utilizado, atualmente mantido e distribuído gratuitamente pela Microsoft, ele está disponível para os sistemas operacionais *Windows, Linux e Mac*.

[1] https://code.visualstudio.com/. Acesso em: 20 julho 2019.

Para realização de testes com API's, foi utilizada a versão gratuita do *software Postman*[2], que permite a realização de requisições HTTP, a partir de sua interface, facilitando o consumo e depuração de serviços de transferência representacional de estado, do inglês *Representational State Transfer* (REST).

O ambiente de desenvolvimento do projeto, foi criado por meio da ferramenta *Node.JS*[3], em conjunto com os recursos disponibilizados a partir do framework *Express*[4].

O *Node* é um ambiente de execução, produzido a partir do motor executado no navegador *Google Chrome*, permitindo que desenvolvedores criem código *JavaScript* no lado do servidor. Ele também conta o gerenciador de pacotes *npm*, que por meio de códigos executados por linha de comando, serve como um utilitário para a iteração com repositórios e bibliotecas online, gerenciando versões e auxiliando suas respectivas instalações.

Para criação e manipulação dos dados, que envolvem o protocolo do Bitcoin, foi utilizada a biblioteca *bitcoinjs-lib*[5], desenvolvida em código *TypeScript*, ela permite sua execução no lado do servidor e é distribuída de forma gratuita. O envio dos dados a rede, foi realizado por meio de requisições HTTP, utilizando a API *Smartbit*.

Para o armazenamento dos dados do projeto, foi utilizado o banco de dados *MongoDB*[6], escrito em C++, com suporte a multiplataformas, ele se tornou popular por possuir alta performance e por ser um banco orientado a documentos *JavaScript Object Notation* (JSON), ao contrário dos bancos mais comuns que seguem o modelo relacional.

Todos os códigos fontes e testes envolvendo a aplicação, foram desenvolvidos utilizando o sistema operacional Windows 10 64 bits, em uma máquina *desktop* com as seguintes configurações:

a) processador: AMD Ryzen 7 1700X 3.4Ghz 8-Core 16-Thread;

---

[2] https://www.getpostman.com/. Acesso em: 20 julho 2019.
[3] https://nodejs.org/en/ .Acesso em: 20 julho 2019.
[4] https://expressjs.com .Acesso em: 20 julho 2019.
[5] https://github.com/bitcoinjs/bitcoinjs-lib. Acesso em: 10 agosto 2019.
[6] https://www.mongodb.com/ .Acesso em: 01 setembro 2019.

b) memória: 8GB memória DDR4 2400MHz;
c) armazenamento: SSD 480gb A400 Kingston.

7.3 BUSCAS E DEFINIÇÕES

Para o início da primeira etapa do desenvolvimento, foram levantadas características essenciais ao projeto.

Nesta etapa, foi realizado um levantamento das formas de programação utilizando o cliente original do Bitcoin, no entanto, seu uso necessita de frequentes atualizações e resulta no download de aproximadamente 210GB de arquivos (BITCOINCORE, 2019, tradução nossa), dessa forma, a execução do projeto se tornaria mais difícil e até mesmo inviável quanto ao uso no lado de clientes, tendo em vista que para um usuário acessar essa aplicação, ele teria de possuir um equipamento com grande capacidade de armazenamento, uma boa conexão com a Internet e conhecimentos com execução de códigos via linha de comando.

A programação de *scripts* diretamente com o Bitcoin Core, também implica em cuidados extremamente necessários para evitar o uso indevido de operadores, podendo ocasionar em altas taxas para validação da transação e até mesmo causar o congestionamento da rede, devido a criação de transações inválidas.

Em decorrer deste cenário, foram consultadas formas de criação de *scripts* por meio de ferramentas que façam uma abstração do protocolo. Como resultado, o uso de bibliotecas que atendam essa necessidade foi caracterizado como mais ideal, no entanto, esse tipo de ferramenta somente cria os dados necessários e não faz seu envio a rede.

Dessa forma, o método encontrado para contornar esse problema, foi pelo uso de API's que realizam comunicação com a *blockchain*, porém, esse tipo de ferramenta sofre constantes mudanças, algumas se iniciam como gratuitas e passam a se tornar *software* proprietário, ou ainda, são descontinuadas por falta de resultados, portanto, a necessidade por testes com diferentes plataformas foi apresentada.

Deste modo, a escolha da *blockchain* do Bitcoin foi feita, realizando a criação do *script* com o uso da biblioteca *bitcoinjs-lib,* ao utilizar o operador OP_RETURN, com o envio de dados a rede por meio da *Smartbit* API.

7.3.1 Bitcoin e operador OP_RETURN

Apesar da existência de diversas redes com a arquitetura da *blockchain*, e até mesmo a possibilidade da criação de uma rede direcionada exclusivamente ao projeto, a decisão do uso da *blockchain* da criptomoeda Bitcoin foi feita.

A proposta do trabalho é o uso de um serviço que propõe maior segurança e longevidade para os arquivos registrados, permitindo que toda administração, durante seu ciclo de interesse, possa ser mantida e feita sem a necessidade da confiança em um terceiro, responsável e autorizado para realizar estes processos.

Tendo em mente estes objetivos, o uso de uma rede pública, amplamente conhecida, com milhares de nós sendo executados, são características adequadas, além disso, elas possuem a estrutura que garante maior imutabilidade e transparência, por meio da descentralização, portanto, se adequando a prova de existência (DURANT; TRACHY, 2017, tradução nossa; GRECH; CAMILLERI, 2017, tradução nossa).

Ainda assim, a decisão foi tomada com base em outros casos de sucesso, como o *Blockcerts* e a UNIC, onde, durante a proposta dos respectivos projetos, a escolha mais clara para o registro de longa vida de dados, seria com o uso da *blockchain* do Bitcoin.

Quanto ao uso do operador OP_RETURN, a escolha deste método, também é favorável ao que diz respeito ao uso da *blockchain* para armazenar dados arbitrários, sabendo que outros *scripts* são capazes de inserir quantidades maiores de dados, no entanto, esta prática não é bem vista por grande parte da comunidade de usuários e desenvolvedores da rede (ANTONOPOULOS, 2014, tradução nossa).

A plataforma *Blockcerts* também faz o uso do operador OP_RETURN, para o registro dos certificados, tendo em vista seus resultados obtidos e pelas universidades que utilizam seu código *Open Source*, a escolha foi realizada sem maiores dúvidas.

7.3.2 Busca e definição da API

O uso de uma API para a manipulação dos dados, apresentou grande importância e foi indispensável para o desenvolvimento da aplicação, dessa forma, foram realizadas pesquisas, com o objetivo de encontrar ferramentas com capacidade de realizar algum tipo de operação envolvendo a *blockchain*.

7.3.2.1 ChainPoint

A *ChainPoint*[7] é uma API, voltada para criação de carimbos de tempo na *blockchain*, verificação de integridade de dados e existência de arquivos.

Seu método de operação é simples, somente é necessário enviar a ela o resumo de um dado desejado, e como resposta, ela retorna um carimbo de tempo, e com ele é possível verificar sua integridade. Ela é distribuída por meio de uma API REST, onde basta acessar um de seus endereços e executar os métodos HTTP com o conteúdo a ser enviado ou verificado.

A *ChainPoint* envia os dados de forma gratuita, no entanto, possui um tempo estimado de até duas horas para submeter o conteúdo. Além disso, possui uma outra característica que não se mostrou adequada ao projeto, onde não é possível configurar as chaves do emissor e verificar em qual transação os dados foram enviados.

7.3.2.2 Blockcypher

Blockcypher[8] é uma API REST, capaz de interagir com determinados protocolos, por meio de requisições HTTP. Atualmente ela possui suporte as *blockchains* do Bitcoin, Bitcoin Testnet, Ethereum, Litecoin e Dodgecoin (BLOCKCYPHER, 2019).

---

[7] https://chainpoint.org/. Acesso em: 08 de agosto 2019.
[8] https://www.blockcypher.com/ Acesso em: 08 de agosto 2019.

Ela possui um desempenho excelente, com retorno de objetos em formato JSON, facilitando o entendimento dos dados e trazendo uma maior organização das respostas.

Durante a implementação, ela foi utilizada para criar e enviar transações para a rede de testes do Bitcoin, conhecida como *Testnet*. Os *scritps* implementados foram capazes de registrar as transações na rede, contudo, a API não possui uma forma de verificar dados a partir da identificação de uma transação, ou pelo resumo de um arquivo, dessa forma, não se mostrou adequada para suprir as necessidades do projeto.

7.3.2.3 Smartbit

A *Smartbit*[9] é uma das diversas plataformas conhecidas como explorador de blocos, do inglês *block explorer,* esse tipo de ferramenta proporciona aos usuários informações sobre a rede, blocos, transações, mineração e outros dados.

Ela também possui uma API muito similar ao *Blockcypher*, no entanto, somente tem suporte as redes, principal e de testes do Bitcoin, porém, ao contrário da maioria das outras plataformas, ela fornece suporte as transações OP_RETURN, permitindo a visualização do seu conteúdo.

Essa API se mostrou adequada ao projeto, devido a proporcionar métodos de consulta a *blockchain* por meio de chaves públicas, transações, *hash* e até mesmo texto. Durante a fase de testes do projeto, foi obtido sucesso ao enviar e consultar transações com seu uso.

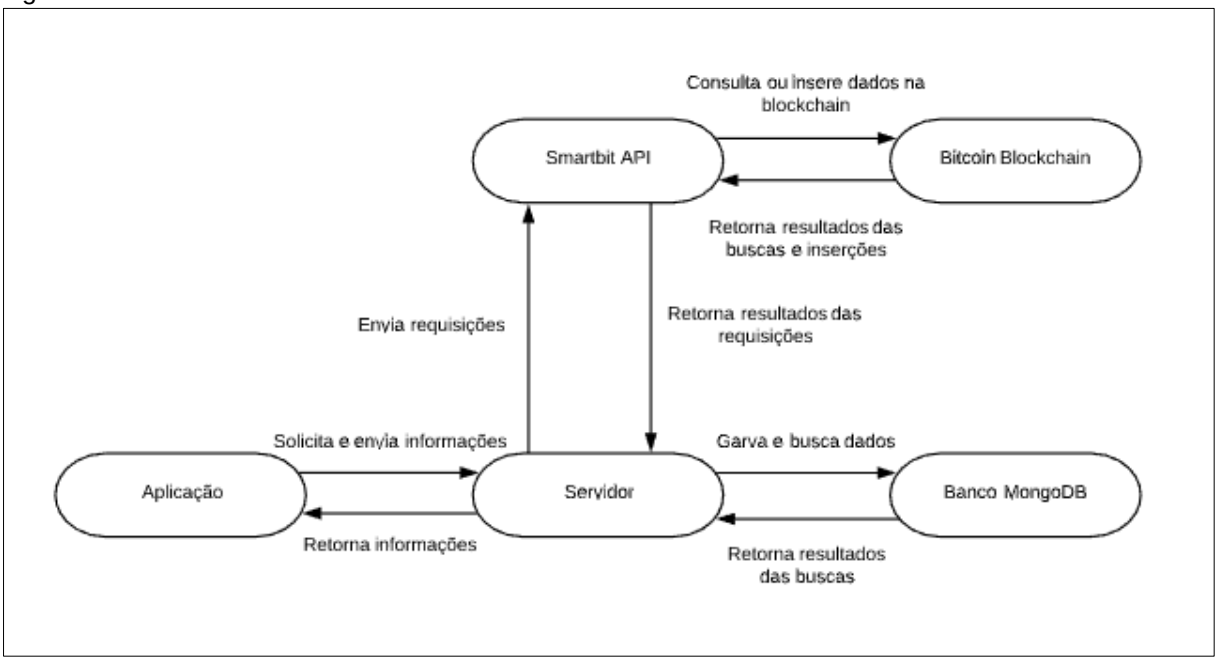
7.3.3 Fluxo de dados e comunicação

Uma vez realizada a escolha da rede e da forma de inserção de dados na *blockchain*, foi estipulado um diagrama de fluxo de dados, com o intuito do mapeamento dos processos e informações necessárias. Com ele foi possível obter

[9] https://www.smartbit.com.au/. Acesso em: 09 de agosto 2019.

uma visão global do projeto, facilitando a percepção de todas etapas a serem implementadas. A figura 7 ilustra o fluxo de dados da aplicação.

Figura 7 - Fluxo de dados



Fonte: Do autor.

O início do fluxo de dados, se dá com as requisições feitas a partir da aplicação para o servidor, o qual pode estar realizando as buscas ou envios para o banco de dados, ou então para a API externa, que é responsável pela comunicação direta com a *blockchain*.

O formato de comunicação estabelecido, ocorre por meio de requisições HTTP, sendo elas tanto para o servidor como para a API, as quais são feitas pelos métodos GET e POST, de acordo com suas necessidades.

O protocolo de transferência utilizado é executado na camada de aplicação, servindo para transmitir diversos tipos de dados. Originalmente foi desenvolvido para comunicação entre navegadores e servidores, ele se baseia no formato de requisição-respostas, e trabalha no modelo cliente-servidor. O método GET, é caracterizado para realização de requisições e por convenção deve somente receber dados, enquanto o método POST, deve ser usado para envio e submissão de dados, os quais devem estar anexados no corpo da requisição (MDN, 2019, tradução nossa).

O formato dos dados, envolvendo todo o projeto, foi definido pela utilização de objetos JSON, tem em vista que a comunicação do servidor com API deve ser realizada com ele, e suas respectivas respostas fazem uso do mesmo padrão, com isso, o uso do banco de dados *MongoDB* se apresentou o mais favorável, tendo em vista que sua orientação se dá por esse tipo de estrutura.

Esse tipo de objeto adota o padrão de chave e valor, possui alta performance com uma formatação de dados leve. Uma outra característica favorável a ele, está relacionada a uma fácil leitura e suporte a diversas linguagens de programação (JSON, 2019, tradução nossa).

7.4 TESTES

Após a conclusão da primeira etapa de desenvolvimento e uma vez que foram definidas todas as características essenciais para o projeto, foi iniciada a fase de testes, com o uso da API e biblioteca de abstração escolhidos.

A fim de evitar gastos e a criação de transações com possíveis erros de implementação, as tornando inválidas e trazendo sobrecarga à rede, as primeiras manipulações de dados, *scripts* e transações foram realizados na *Testnet*, a rede de testes do Bitcoin.

A página principal do *Github* da biblioteca *bitcoinjs-lib,* possui várias orientações para a criação de chaves, endereços e transações, além disso, foram adaptados códigos[10] com instruções do uso da rede, para auxiliar na implementação inicial.

Os testes foram realizados em partes: geração da carteira, importação da carteira, consultas a dados da rede, criação de uma transação, envio a rede, verificação de transações e busca por dados OP_RETURN. A tabela 6 representa os testes realizados.

---

[10] http://blog.willclark.tech/tech/2017/05/27/get-started-with-bitcoin-testnet.html. Acesso em: 15 de agosto 2019.

Tabela 6 - Testes realizados

Função	***Rede***
Geração de carteira	*Testnet*
Importação de carteira	*Testnet*
Criação de transações	*Testnet*
Consultas a rede	*Testnet*
Envio de dados a rede	*Testnet*
Busca por transações	*Testnet*

Fonte: Do Autor.

Os códigos foram desenvolvidos com o uso do *software Visual Studio Code* e sua execução se deu por linha de comando, com a ferramenta *Node.js*. As buscas, consultas e envios dos dados e transações, foram realizados o auxílio da ferramenta *Postman*.

O registro dos dados nesta rede também necessita de uma taxa, dessa forma, para adicionar saldo a carteira criada, foram solicitados *faucets*[11]*,* prática que consiste em pedir uma pequena quantidade de *satoshis*[12], nas plataformas mais comuns e utilizadas por desenvolvedores da área. Por se tratar de uma rede de testes é possível conseguir "*test coins*[13]" gratuitamente, as quais devem e são somente utilizadas para esse tipo de operação e não possuem nenhum valor monetário atribuído (WIKEPEDIA, 2019).

Dessa forma, os primeiros registros foram gerados na *Testnet* e uma vez que sua validação e consulta foram possíveis de se executar, todo o código foi adaptado para ser executado na rede principal.

7.5 IMPLEMENTAÇÃO DO PROTÓTIPO

A partir da conclusão da primeira e segunda etapas do desenvolvimento, foi iniciada a implementação final do protótipo. Essa etapa foi responsável pela criação dos modelos de dados, conforme o levantamento definido, assim como, a

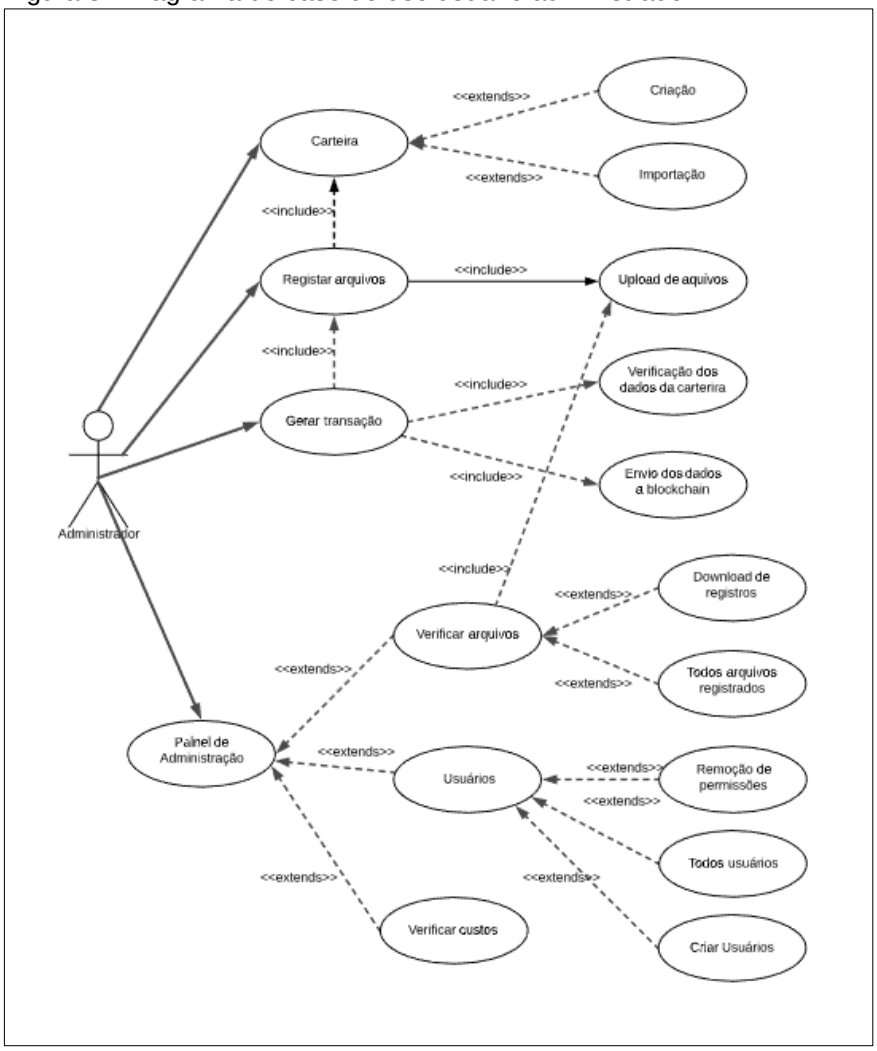
[11] Refere-se a aquisição de *satoshis* de forma gratuita, na *Testnet*, esse é o modo mais comum para adquirir as moedas.
[12] Menor fração de um Bitcoin, corresponde a um centésimo de milionésimo da moeda.
[13] Na rede de testes, as moedas também são chamadas de *"test coins".*

configuração do servidor, desenvolvimento da aplicação para o lado do cliente e toda estrutura para a criação de dados e comunicação com a rede.

De acordo com as características e dados levantados, foram desenvolvidos três diagramas de caso de uso, conforme os diferentes tipos de usuário acessando a aplicação.

A figura 8 representa o diagrama de caso de uso, para um usuário administrador, onde ele tem acesso a todas funcionalidades da aplicação, podendo criar usuários e ter acesso ao painel de administração, assim como, registrar e consultar documentos.

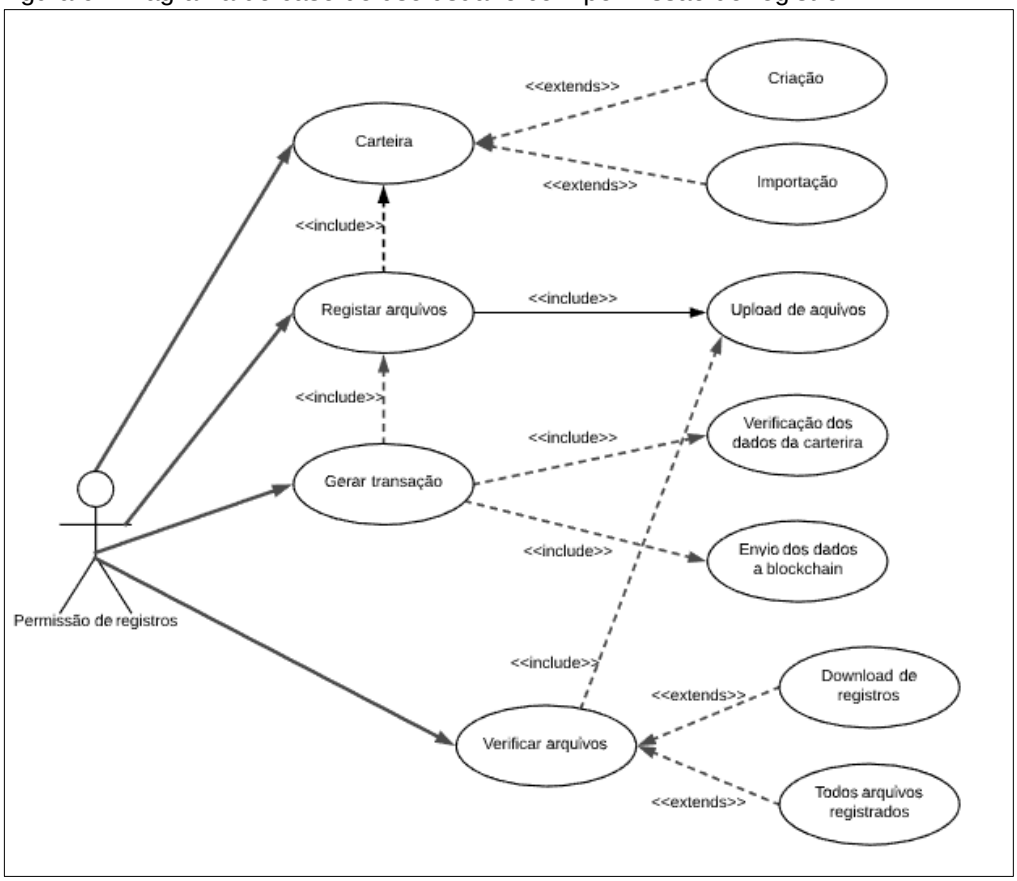
Figura 8 - Diagrama de caso de uso usuário administrador



Fonte: Do autor.

A figura 9 ilustra o diagrama de caso de uso, para um usuário com permissão de registro, dessa forma, ele tem acesso a criação ou importação de uma carteira, registro e consulta de arquivos.

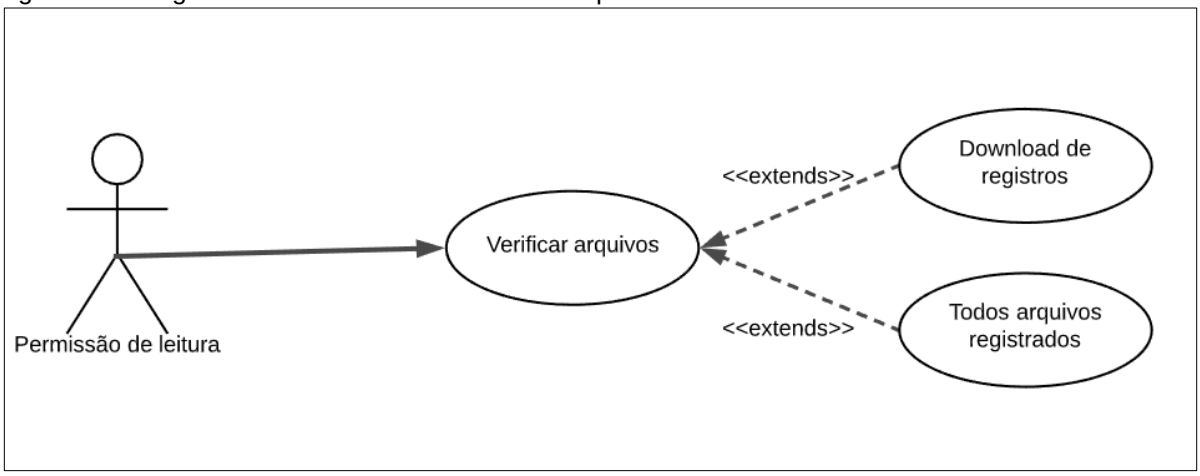
Figura 9 - Diagrama de caso de uso usuário com permissão de registro



Fonte: Do autor.

A figura 10 representa a funcionalidade, que um usuário com permissão de leitura tem na aplicação, sendo ela, a consulta de arquivos.

Figura 10 - Diagrama de caso de uso usuário com permissão de leitura



Fonte: Do autor.

7.5.1 Modelagem de dados

A modelagem do banco de dados, facilita a organização e adaptação dos recursos utilizados, de acordo com o fluxograma de comunicação do projeto.

Para haver um controle de usuários e acessos com permissões na aplicação, foi necessário a criação de uma coleção (refere-se a uma tabela no *MongoDB*) de usuários, para ser registrada no banco de dados. Também foi criada uma segunda coleção para o registro dos arquivos, que contém a informações básicas sobre ele, assim como, informações do usuário e dados referentes ao registro da transação.

A coleção de arquivos possui os seguintes dados (figura 11): nome do arquivo, *hash* do arquivo, identificação da transação, custo em *satohis*, custo em reais, nome do usuário que registrou o arquivo, data, nome do acadêmico, endereço e chave pública utilizados para criar a transação.

Figura 11 - Modelo da coleção de Arquivos

```
var fileSchema = new mongoose.Schema({
    fileName: String,
    fileHash: String,
    txid: String,
    satoshiCost: Number,
    brlCost: Number,
    user: String,
    date: Date,
    name: String,
    publicAddress: String,
    publicKey: String
}, { collection: 'filecollection' }
);
```

Fonte: Do autor.

A coleção de usuários (figura 12), ela armazena o nome completo, o nome de usuário, a senha e o tipo de permissão de acesso.

Figura 12 - Modelo da Coleção de Usuários

```
var userSchema = new mongoose.Schema({
    fullName: String,
    userName: String,
    password: String,
    permission: String
}, { collection: 'usercollection' }
);
```

Fonte: Do autor.

Quanto ao armazenamento dos dados referentes a carteira, ele somente existe em um registro feito pela variável *global*[14]*,* disponibilizada pelo *Node.JS*, a qual é mantida apenas em tempo de execução, tendo em vista que esta prática é a forma adequada de lidar com esse tipo de dado, onde os mantendo salvos, a aplicação estaria tomando posse da carteira do usuário.

7.5.2 Configuração do servidor

O servidor utilizado na aplicação, é criado com o ambiente *Node.JS* e auxílio do *framework Express*, o qual contribui na construção de aplicações *web*.

Para sua configuração, o primeiro passo tomado foi a instalação do *Node.JS*, com isso, é possível utilizar seu gerenciador de pacotes *npm* e fazer a instalação do *Express* e realizar as demais configurações.

Ainda com o *npm*, são instaladas algumas bibliotecas, para melhorar a estrutura do servidor e facilitar a comunicação com o banco de dados e a API, sendo elas:

a) *axios*[15]: ferramenta para realização de requisições HTTP;
b) *body-parser*[16]: *middleware* capaz de converter o corpo das requisições para vários formatos, no projeto, é utilizado para conversão em formato JSON;
c) *consign*[17]: auto carregamento de *scripts*, módulos e pastas;
d) *express-session*[18]: *middleware* responsável pelo gerenciamento das sessões de usuários;
e) *express-validator*[19]: utilizado para validação de dados;

[14] https://nodejs.org/api/globals.html. Acesso em: 30 julho 2019.
[15] https://github.com/axios/axios. Acesso em: 22 julho 2019.
[16] https://github.com/expressjs/body-parser. Acesso em: 20 julho 2019.
[17] https://github.com/jarradseers/consign#readme. Acesso em: 20 julho 2019.
[18] https://github.com/expressjs/session. Acesso em: 07 setembro 2019.
[19] https://express-validator.github.io. Acesso em: 20 julho 2019.

f) *ejs*[20]: é um módulo que permite a utilização de código Java*Script,* com elementos *HyperText Markup Language* (HTML), dessa forma, é possível enviar e exibir objetos, a partir servidor para a interface no lado do cliente.

As configurações do servidor e estrutura do protótipo, são definidas da seguinte forma (figura 13): no início é realizada uma inclusão de todos os módulos utilizados, após isso, uma instância do *Express* é feita e por meio do método *set,* é definido o tipo de arquivo a ser renderizado pela aplicação, sendo ele o *ejs.* Com o método *use,* é definido o diretório de arquivos estáticos do projeto, ele também é utilizado para definir o uso de requisições do tipo JSON e por fim é usado para definir o *middleware express-validator* e para criar o sistema de sessões do projeto. Para encerrar a configuração, o *consign* é usado para incluir o diretório de rotas e controles na instância do *framework Express.*

[20] https://ejs.co/. Acesso em: 20 julho 2019.

Figura 13 - Configuração do servidor

```
let express = require('express');

const expressSession = require('express-session')
let errorHandler = require('../app/helpers/errorHandler');
let consign = require('consign');
let bodyParser = require('body-parser');
let expressValidator = require('express-validator');

let app = express();

app.set('view engine', 'ejs');
app.set('views', './app/views');

app.use(express.static('./app/public'));
app.use(bodyParser.urlencoded({ extended: true }));
app.use(bodyParser.json());
app.use(expressValidator());
app.use(expressSession({
    secret: 'auth-token',
    resave: false,
    saveUninitialized: false
}))
app.use(cors());

global.GLOBAL_WALLET = {}
global.GLOBAL_FILE = {}

consign()
    .include('app/routes')
    .then('app/controllers')
    .into(app);
```

Fonte: Do autor.

7.5.3 Desenvolvimento da aplicação

Com o ambiente configurado, foi iniciado o desenvolvimento da aplicação para gerenciamento de dados, enviados pelo usuário, e comunicação com a *blockchain*.

Todo código relacionado ao *back-end* do protótipo, foi desenvolvido com a linguagem de programação *JavaScript* e para o lado do *front-end,* foram utilizadas as tecnologias HTML, CSS e a biblioteca de estilos *Bootstrap*[21] na versão 4.0.

7.5.3.1 Geração da carteira

O sistema proposto neste trabalho, necessita da posse de um par de chaves e seu respectivo endereço Bitcoin, com fundos para cobrir o custo das taxas. A posse e controle de um endereço são essenciais, sabendo que para o registro de uma transação, serão necessários o acesso a sua chave privada de acordo com as regras do protocolo. Para montar um *script* de uma nova transação, é necessário que o devido endereço selecionado tenha o valor das taxas, referenciado em alguma UTXO, tornando possível sua utilização.

A chave privada consiste em um número aleatório entre 1 e 2^{256}, dessa forma, com o auxílio da biblioteca *bitcoinjs-lib*, ela é gerada (figura 14) a partir de um padrão aleatório previamente estabelecido, a fim de evitar o uso de uma chave criada em um intervalo numérico de fácil acesso, tornando-se inseguro.

Figura 14 - Código da geração da chave privada

```
const bitcoin = require('bitcoinjs-lib')
const axios = require('axios')
let myKeyPair = bitcoin.ECPair.makeRandom()
```

Fonte: Do Autor.

O código desenvolvido, consulta previamente se a chave privada gerada possui saldo em seu endereço, antes de disponibilizá-la ao usuário, caso possua um saldo maior que zero, uma nova chave é gerada e novamente testada. Esta verificação é vista como uma boa prática nas aplicações que utilizam a *blockchain*.

A biblioteca utilizada calcula a chave pública, por meio de multiplicação de curvas elípticas, com o uso do padrão secp256k1, onde são estabelecidos a curva e o ponto gerador.

---

[21] https://getbootstrap.com/. Acesso em: 30 julho 2019.

A curva elíptica deste padrão é definida pela função: $y^2 \, mod \, p = (x^3 + 7) \, mod \, p$, onde $mod \, p$ é um módulo da constante p, e também indica que esta curva é definida por um conjunto finito sobre p e p é um número primo igual á $2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$. Para criá-la, é necessário multiplicar a chave privada pelo ponto G definido no padrão, o qual é uma constante e é utilizado para todas as chaves, portanto, garantindo sempre o mesmo resultado (ANTONOPOULOS, 2014, tradução nossa).

Supondo uma chave privada k, a chave pública K é alcançada ao multiplicar $K = k * G$, onde G é o ponto gerador, por fim, o resultado é obtido ao definir a chave pública como um ponto $K = (x, y)$ (ANTONOPOULOS, 2014, tradução nossa).

Ela também disponibiliza o endereço, no padrão P2PKH correspondente a chave privada, que é criado pela execução das funções criptográficas SHA-256 e RIPEMD-160, respectivamente, sobre a chave pública. A figura 15 ilustra o código desenvolvido para geração do endereço e teste de seu saldo.

Figura 15 - Geração e teste do saldo endereço criado

```
const checkNewKey = (addrs = GLOBAL_ADDRESS.address) =>
    axios.get('https://blockchain.info/rawaddr/' + addrs)
        .then(response => response.data.n_tx)
        .catch((error) => {
            console.error(error);
        });


const createNewKeyPair = (address) => Promise.all([
    checkNewKey(address)
])
    .then(([res]) => {
        if (res != 0) {
            myKeyPair = bitcoin.ECPair.makeRandom()
            GLOBAL_ADDRESS = bitcoin.payments.p2pkh({ pubkey: myKeyPair.publicKey })
            return createNewKeyPair(GLOBAL_ADDRESS.address)
        }
    }).catch((e) => {
        res.render('index', { error: 'Não foi possível criar o endereço, por favor tente novamente' })
    })

createNewKeyPair()
```

Fonte: Do autor.

Por fim, os dados são apresentados ao usuário, conforme representado na figura 16, o qual tem acesso a sua chave privada no formato *wallet import format*

(WIF), que consiste em uma forma codificada de sua representação, tornando mais fácil sua manipulação.

Figura 16 - Resultado da carteira criada

Fonte: Do autor.

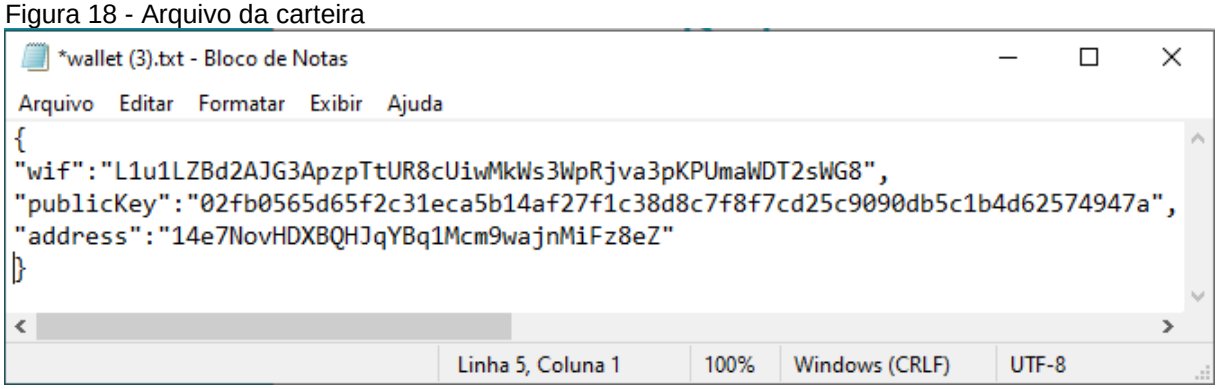
O sistema ainda permite a importação de uma carteira, por meio de um WIF (figura 17), ou então sua devida geração.

Figura 17 - Importação da carteira no formato WIF

Fonte: Do autor.

O usuário ainda pode efetuar o download da sua carteira, a qual é disponibilizada em um arquivo de texto, com a estrutura de um objeto JSON (figura 18).

Figura 18 - Arquivo da carteira



Fonte: Do autor

7.5.3.2 Hash de arquivos

A aplicação consta com o registro de um arquivo na *blockchain*, no entanto, com o objetivo de minimizar custos e permitir o registro de um arquivo de qualquer tamanho, é inserido na rede um identificador único do arquivo, obtido pela execução da função criptográfica SHA-256, que retorna um *hash* no tamanho fixo de 32 bytes.

As funções *hash* criptográficas, possuem propriedades que tornam extremamente difícil encontrar qualquer outro conteúdo, que possua um mesmo identificador, dessa forma, o resultado pode ser visto como exclusivo, e garante sua busca na rede para as possíveis verificações. Além disso, qualquer alteração no arquivo original, resultaria em um resumo totalmente diferente, portanto, a realização de fraudes se torna um processo impraticável.

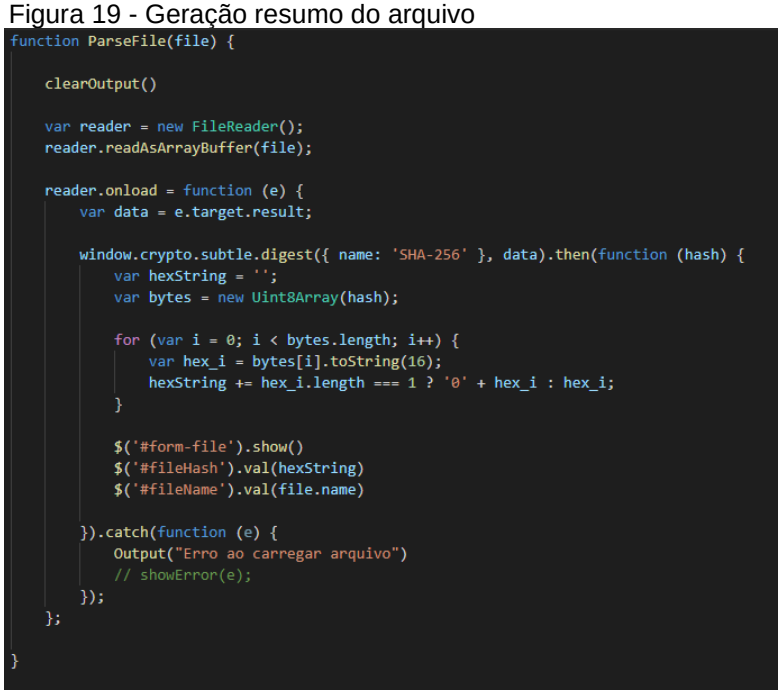
Como a seleção e envio dos arquivos ocorre no lado do cliente, a propriedade responsável por convertê-lo em um resumo, ocorre na execução do *JavaScript* na parte do navegador, utilizando a interface *Web Crypto API*[22], a função *digest()*, se torna responsável pela geração do resumo do arquivo.

Uma vez que selecionado o arquivo, o código o interpreta como um *bufffer* e envia para função de conversão, a qual o utiliza por meio de um *array* em *bytes*, e

[22] https://developer.mozilla.org/pt-R/docs/Web/API/Web_Crypto_APIi. Acesso em: 10 agosto 2019.

retorna como resposta o *hash* do arquivo codificado em sistema hexadecimal (figura 19).

Figura 19 - Geração resumo do arquivo



```
function ParseFile(file) {

    clearOutput()

    var reader = new FileReader();
    reader.readAsArrayBuffer(file);

    reader.onload = function (e) {
        var data = e.target.result;

        window.crypto.subtle.digest({ name: 'SHA-256' }, data).then(function (hash) {
            var hexString = '';
            var bytes = new Uint8Array(hash);

            for (var i = 0; i < bytes.length; i++) {
                var hex_i = bytes[i].toString(16);
                hexString += hex_i.length === 1 ? '0' + hex_i : hex_i;
            }

            $('#form-file').show()
            $('#fileHash').val(hexString)
            $('#fileName').val(file.name)

        }).catch(function (e) {
            Output("Erro ao carregar arquivo")
            // showError(e);
        });
    };

}
```

Fonte: Do Autor

O projeto permite adicionar arquivos (figura 20) no formato *portable document format* (PDF), o qual é atribuído a muitos dos diplomas e certificados. Ainda, conforme proposto o uso do padrão *Open Badges*, o envio de arquivos no formato JSON também são permitidos.

Figura 20 - Upload de arquivos na aplicação

Fonte: Do Autor

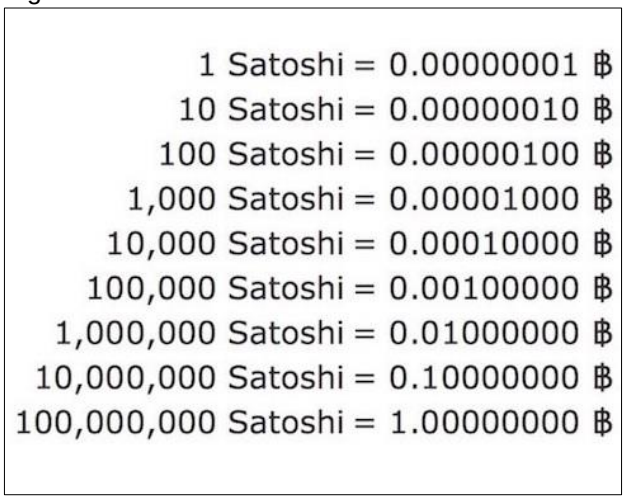
7.5.3.3 Criação da transação

Uma vez gerada a carteira e adicionado o arquivo para registro na *blockchain*, é necessário montar uma transação para realizar o envio a rede, deste modo, a aplicação utiliza o *script null data,* por meio do operador OP_RETURN.

Para montá-lo, deve-se fornecer um endereço e sua chave privada e definir a taxa que será paga ao minerador, responsável por inserir a transação em um bloco.

A taxa é apresentada por valores em *satoshis*, que consiste na forma de representação da menor unidade de uma moeda BTC, no caso, um Bitcoin equivale a cem milhões de *satoshis*. A figura 21 mostra alguns exemplos de conversão da moeda.

Figura 21 - Conversão de BTC em *satoshis*



Fonte: Soares (2019)

A aplicação possui uma opção para o uso da taxa recomendada (figura 22), onde seu cálculo é realizado por meio de uma consulta, no tamanho médio e valor das taxas de transações, que variam de acordo com o número operações sendo realizadas e o valor da moeda BTC. Os valores são obtidos por meio de uma requisição a API da plataforma *Bitcoinfees*[23].

Figura 22 - Código para consulta da taxa recomendada

```
const getRecommendedFee = () => {
    const url = `https://bitcoinfees.earn.com/api/v1/fees/recommended`
    return axios.get(url)
}

const setFee = () =>
Promise.all([
    getRecommendedFee(),
]).then(([recommendedFee]) => {

    const fee = recommendedFee.data.fastestFee * AVERAGE_TRANSACTION_SIZE
    if (fee > parseFloat($("#balance-value").text() * 100000000) ) {
        insufficientBalance()
    } else {
        $("#fee").val(fee)
    }
})
```

Fonte: Do Autor.

O uso do valor recomendado, garante uma propagação dos dados mais rápida a rede, porém, o *software* implementado aceita como valor mínimo 1000

[23] https://bitcoinfees.earn.com/api. Acesso em: 20 agosto 2019.

satoshis, contudo é importante lembrar que valores baixos são “ignorados” pelos mineradores, caracterizando o atraso na sua propagação.

Para compor a transação, a biblioteca *bitcoinlib-js* é usada novamente, onde são adicionadas as entradas para o *script* que será enviado a rede, as quais consistem no valor da taxa, com isso, o *software* realiza uma busca (figura 23), por meio do uso da API *Smartbit*, no conjunto UTXO da rede, verificando se existem saídas não gastas pelo endereço utilizado.

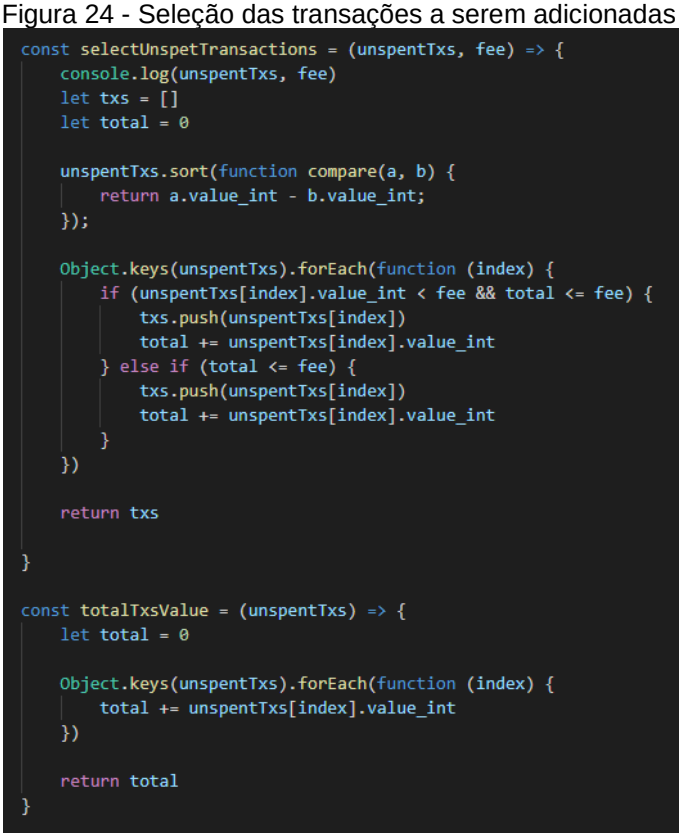
Figura 23 - Consulta a *blockchain* por transações disponíveis ao endereço

```
const getUnspentTransactionsForAddress = address => {
    const url = `https://api.smartbit.com.au/v1/blockchain/address/${address}/unspent`
    return axios
        .get(url)
        .then(response => response.data.unspent)
}

const getBalance = address => {

    const url = `https://api.smartbit.com.au/v1/blockchain/address/${address}`
    return axios
        .get(url)
        .then(response => response.data.address.total.balance_int
        )
}
```

Fonte: Do Autor.

Com a consulta, o *software* permite o agrupamento de uma ou mais entradas, a fim de alcançar o valor total necessário, e caso o endereço não tenha saldo suficiente, um retorno é enviado a ele (figura 24).

Figura 24 - Seleção das transações a serem adicionadas como entradas



```
const selectUnspetTransactions = (unspentTxs, fee) => {
    console.log(unspentTxs, fee)
    let txs = []
    let total = 0

    unspentTxs.sort(function compare(a, b) {
        return a.value_int - b.value_int;
    });

    Object.keys(unspentTxs).forEach(function (index) {
        if (unspentTxs[index].value_int < fee && total <= fee) {
            txs.push(unspentTxs[index])
            total += unspentTxs[index].value_int
        } else if (total <= fee) {
            txs.push(unspentTxs[index])
            total += unspentTxs[index].value_int
        }
    })

    return txs

}

const totalTxsValue = (unspentTxs) => {
    let total = 0

    Object.keys(unspentTxs).forEach(function (index) {
        total += unspentTxs[index].value_int
    })

    return total
}
```

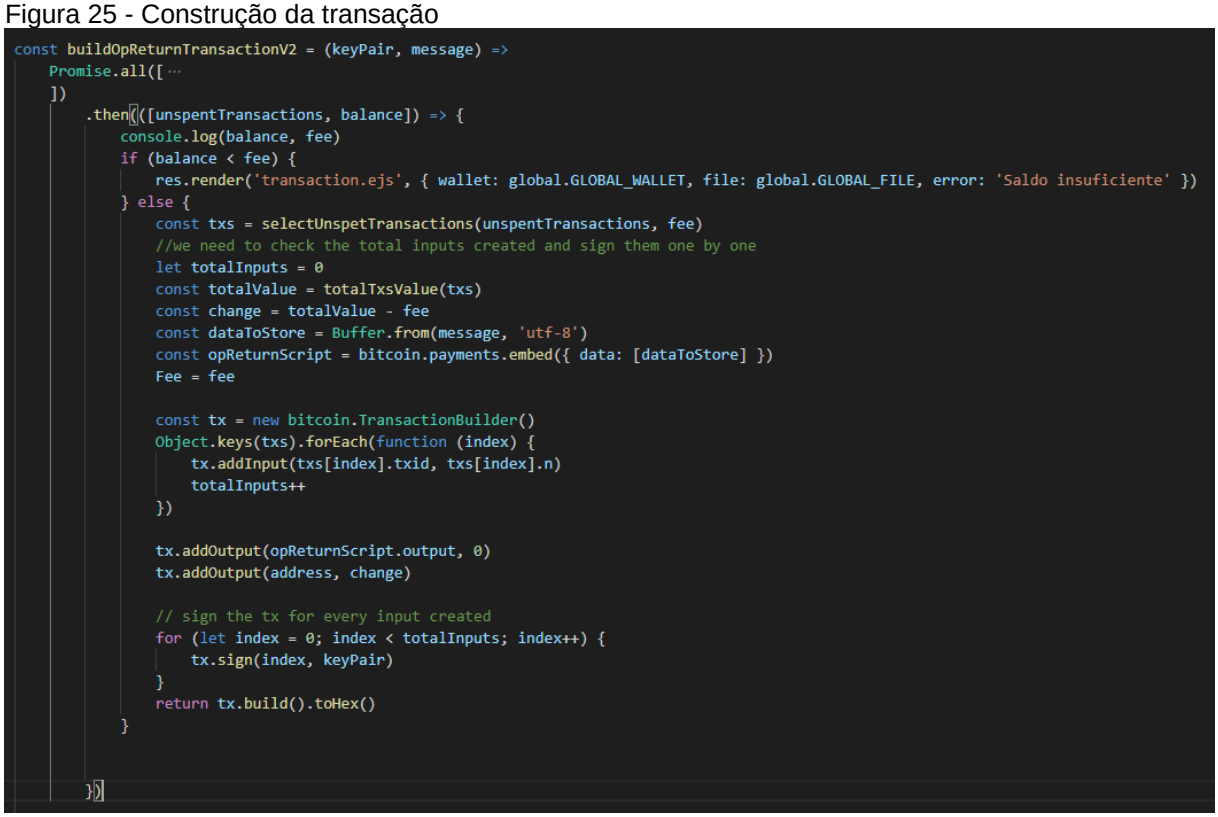
Fonte: Do Autor.

Uma vez que adicionadas as entradas, a aplicação produz suas saídas, neste caso, com o uso do *script null_data*, o operador OP_RETURN deve ser atribuído a pelo menos uma delas, representando o *hash* do arquivo a ser registrado.

O uso de uma segunda saída, se faz necessário quando o endereço fornecido, possua operações encontradas na UTXO com valores maiores que o taxa selecionada, consequentemente, ela simboliza o “troco” para o usuário, onde deve ser informado seu próprio endereço e o valor em *satoshis* que deve ser enviado a ele mesmo. Esse valor é o resultado da subtração entre, o valor total das transações e a taxa escolhida.

Após a inclusão dos *inputs* e *outputs* do *script*, é obrigatório realizar sua assinatura digital, a qual é efetuada com o uso de seu par de chaves. A figura 25 ilustra todo processo de criação da transação.

Figura 25 - Construção da transação



```
const buildOpReturnTransactionV2 = (keyPair, message) =>
    Promise.all([ ···
    ])
        .then(([unspentTransactions, balance]) => {
            console.log(balance, fee)
            if (balance < fee) {
                res.render('transaction.ejs', { wallet: global.GLOBAL_WALLET, file: global.GLOBAL_FILE, error: 'Saldo insuficiente' })
            } else {
                const txs = selectUnspetTransactions(unspentTransactions, fee)
                //we need to check the total inputs created and sign them one by one
                let totalInputs = 0
                const totalValue = totalTxsValue(txs)
                const change = totalValue - fee
                const dataToStore = Buffer.from(message, 'utf-8')
                const opReturnScript = bitcoin.payments.embed({ data: [dataToStore] })
                Fee = fee

                const tx = new bitcoin.TransactionBuilder()
                Object.keys(txs).forEach(function (index) {
                    tx.addInput(txs[index].txid, txs[index].n)
                    totalInputs++
                })

                tx.addOutput(opReturnScript.output, 0)
                tx.addOutput(address, change)

                // sign the tx for every input created
                for (let index = 0; index < totalInputs; index++) {
                    tx.sign(index, keyPair)
                }
                return tx.build().toHex()
            }


        })
```

Fonte: Do Autor.

A biblioteca utilizada para a criação da transação, não a envia diretamente a rede, na verdade, o resultado obtido a partir dela, consiste em dados hexadecimais com todo o conteúdo gerado, dessa forma, para realizar sua propagação, a API da *Smartbit* é utilizada novamente (figura 26).

Figura 26 - Envio da transação a rede

```
const pushTransaction = transaction => {
    const url = 'https://api.smartbit.com.au/v1/blockchain/pushtx'
    const params = { hex: transaction }
    return axios.post(url, params)
}
```

Fonte: Do Autor.

Uma vez que enviada a rede, caso possua a estrutura correta para ser validade pelo método *isStandard(),* ela será inserida em um bloco por um minerador, proporcionando o registro efetivo do arquivo, porém, deve-se ter em mente, que este processo não é efetivado de forma instantânea, existem fatores como a sobrecarga da rede e o próprio valor da taxa escolhida, que implicam no tempo de validação da transação.

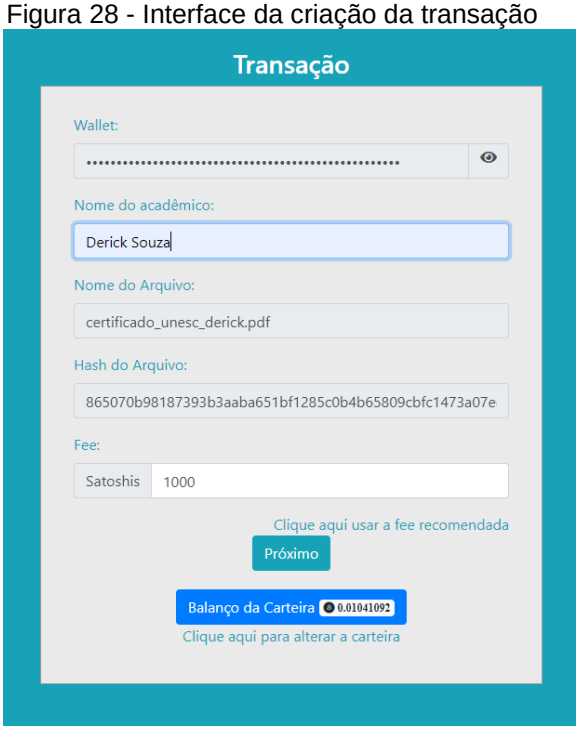
Para evitar registros duplicados, antes de realizar o envio da transação a rede, o *software* faz uma busca pelo identificador do arquivo, com isso, caso o registro seja encontrado, um aviso de confirmação é retornado ao usuário, o informando e solicitando sua autorização para o envio ou seu cancelamento (figura 27).

Figura 27 - Resultado da verificação de um registro do arquivo

Fonte: Do Autor.

A figura 28 representa a interface para a criação da transação.

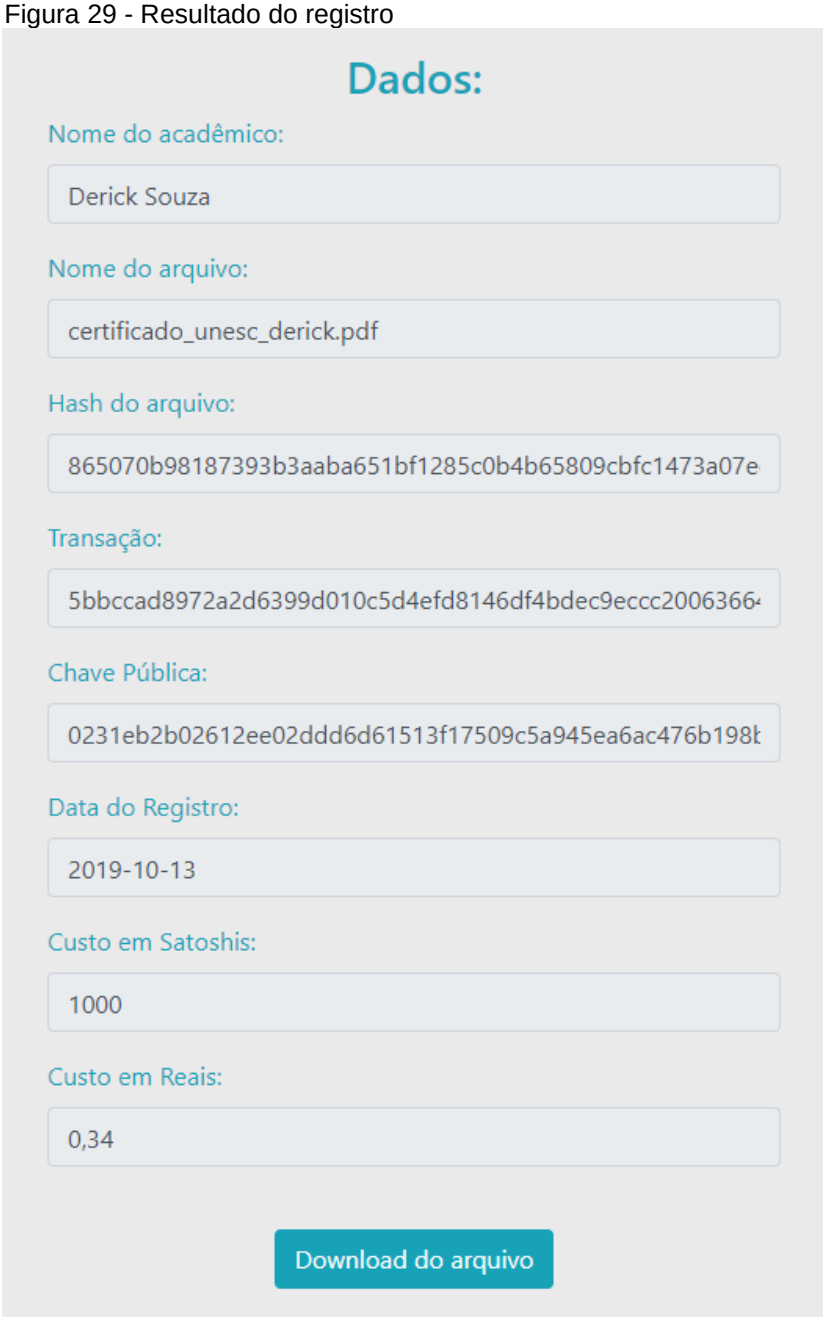
Figura 28 - Interface da criação da transação



Fonte: Do Autor.

Após a conclusão da transação o usuário tem acesso ao registro gerado na *blockchain*, e as suas respectivas informações (figura 29).

Figura 29 - Resultado do registro



Fonte: Do Autor.

O usuário ainda pode efetuar o download de um arquivo em PDF que contém todas as informações em um modelo mais legível (figura 30).

Figura 30 - Arquivo para download

Fonte: Do Autor.

7.5.3.4 Verificação de arquivos

Para verificação, um usuário deve realizar o upload do arquivo original, onde a aplicação, executa o algoritmo para criar seu resumo e então exercer uma varredura pela *blockchain*, e ao obter o resultado é feita uma comparação para garantir a integridade dos dados.

Uma vez que a busca tenha sucesso, o usuário pode acessar a transação (figura 31) em questão para verificar seus dados, ou então, confirmar o conteúdo de sua busca conforme o retorno apresentado a ele.

Figura 31 - Resultado da verificação

Fonte: Do Autor.

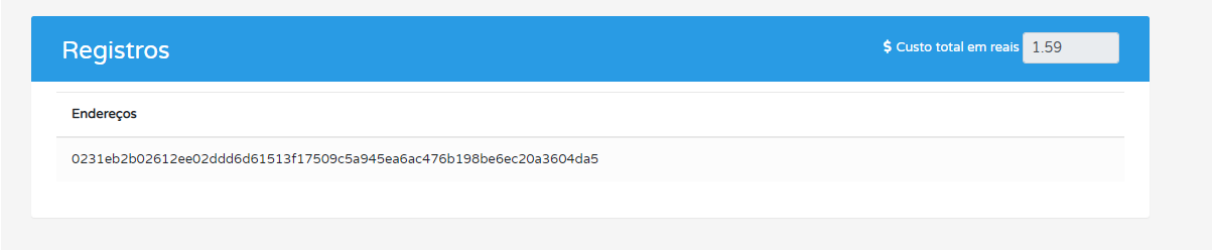
7.5.3.5 Painel de administração

Para o controle de acesso e restrições de funcionalidades foram desenvolvidas três categorias de usuários:

a) *admin:* possui acesso a todas funcionalidades, inclusive a criação de novos usuários e negação de permissões;
b) *writer*: possui permissão para o registro de arquivos, verificação e consulta de arquivos registrados;
c) *reader*: somente tem acesso a verificação e aos arquivos registrados.

O usuário administrador, tem acesso aos endereços utilizados e ao custo de total de todos registros feitos na aplicação (figura 32). O valor apresentado consiste na soma da taxa paga por um registro, convertido em reais na data em que a transação foi criada.

Figura 32 - Endereços e custo total



Fonte: Do Autor.

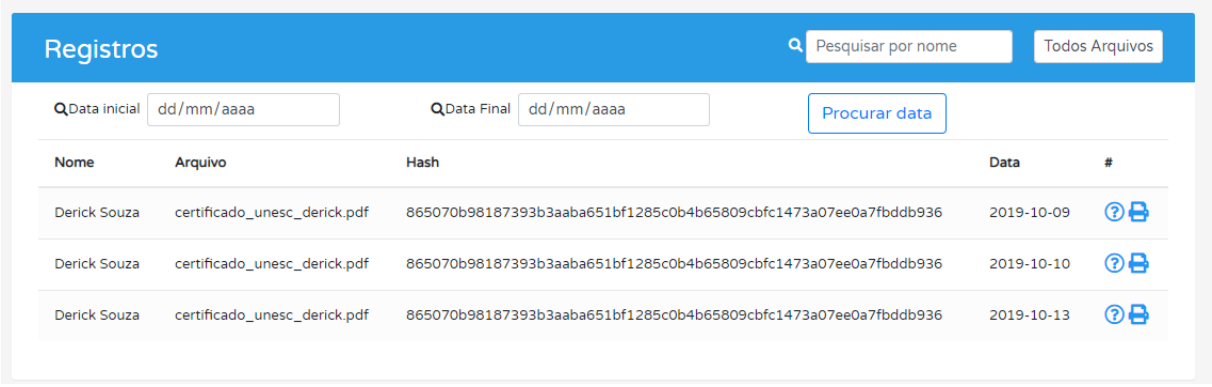
Ele também pode verificar os usuários registrados (figura 33) assim como criar um acesso a plataforma de acordo com a permissão necessária.

Figura 33 - Usuários cadastrados

Fonte: Do Autor.

Todos usuários têm acesso aos arquivos registrados na plataforma, onde é possível realizar uma consulta por data ou pelo nome do acadêmico (figura 34).

Figura 34 - Lista de arquivos registrados



Fonte: Do Autor.

7.5.3.6 Arquivos registrados

Com toda a implementação finalizada, alguns arquivos foram registrados para viabilizar a estrutura desenvolvida.

Foram registrados quatro arquivos referentes a certificados no formato PDF e um arquivo no formato JSON, utilizando a modelo base da plataforma *Blockcerts*.

Os registros foram realizados em diferentes datas e com o valor de taxa de 1000 *satoshis*, como resultado, todas as cinco tentativas foram executadas com sucesso. As transações concluídas podem ser verificadas com a consulta em uma ferramenta *block explorer* com a busca pelo endereço Bitcoin *1LQ5Qhoh8U7VVH1LT752dwuE8jzCmA695V*.

8 APRESENTAÇÃO E ANÁLISE DOS DADOS

Com a conclusão de todas etapas determinadas para o projeto e com os respectivos testes realizados, foi possível realizar uma análise relacionada a seus resultados e dessa forma, verificar sua viabilidade.

Com o protótipo finalizado, os conceitos e tecnologias que envolvem o ambiente de desenvolvimento da *blockchain* foram aplicados, ao utilizar as técnicas e métodos necessários para a área, permitindo o manuseio de uma aplicação que se adequa ao ambiente do protocolo Bitcoin.

Os registros efetivados se adequam ao conceito da prova de existência, garantindo ao emissor e portador do diploma ou certificado, uma prova irrefutável de sua veracidade, além do mais, eliminam a necessidade de terceiros, para manter o processo de verificação desses documentos.

O processo de levantamento de dados realizado, permitiu compor uma estrutura do protótipo, a qual se adequa ao tipo de *blockchain* utilizada, tendo como foco as características de descentralização, longevidade e transparência, as quais foram alcançadas com o uso do protocolo Bitcoin e sua rede pública.

Durante o estudo do projeto *Blockcerts*, o MIT também identificou a *blockchain* do Bitcoin como a mais adequada, pelo fato de ela ser a rede mais testada e confiável, até o presente momento, além disso, com a número quantidade mineradores e com os inúmeros investimentos na moeda, são favoráveis ao que diz respeito a sua longevidade (MIT MEDIA LAB, 2019, tradução nossa).

Com o uso desta rede, também foi possível alcançar alguns dos conceitos vistos como essenciais para a preservação digital, sendo eles, referente ao armazenamento em bancos de dados distribuídos e descentralizados. Ao armazenar registros de forma centralizada, sua seguridade fica comprometida com as possíveis falhas em seu *hardware* de armazenamento, contudo, com o registro na rede do Bitcoin, qualquer falha em um de seus milhares de nós completos, não compromete qualquer problema relacionado a integridade de tais dados. Ainda, como comprometimento da IES que adotar seu uso, um nó completo deveria ser executado por ela.

Uma característica, não levada em conta durante o desenvolvimento, foi a forma de verificação da chave pública utilizada para a criação da transação, sabendo que ela é a principal condição para verificar se o diploma, foi realmente emitido por uma instituição de ensino autorizada. O principal meio de verificação do documento é pelo seu identificador único, no entanto, nada impede que um usuário mal-intencionado registre o mesmo documento na rede, porém, ao identificar a chave pública do registro, seria possível negar a sua autenticidade.

Para realizar este processo de verificação a forma mais adequada seria pela comparação das chaves, porém, no atual momento, é difícil estipular a forma em que uma instituição estaria adotando o uso de seu par de chaves, portanto, um cenário ideal para esta solução é apresentado na seção 8.1.

Um dos primeiros obstáculos encontrados no decorrer do projeto foi a decisão de qual metodologia e padronização a serem utilizadas, tendo em vista que os padrões e convenções para o desenvolvimento de aplicações com o uso da *blockchain* ainda não foram bem estabelecidos, assim como, as boas práticas relacionados ao seu uso para objetivos não financeiros (ANTONOPOULOS, 2014, tradução nossa).

Com este contexto presente, a tentativa do uso do módulo *cert-issuer* [24] do *Blockcerts* foi realizado, por meio das instruções encontradas na página inicial do *GitHub* da plataforma. O teste inicial foi realizado em uma máquina com o sistema operacional *Windows 10 Pro 64 bits*, contudo, ocorreram inúmeros erros, em relação a versões de bibliotecas e conflitos, com a instalação do *Docker*[25] do projeto. Sua utilização traria um ambiente mais enriquecido e confiável, tendo em vista os atuais projetos e instituições que fazem o uso do mesmo código, contudo, em decorrer das dificuldades enfrentadas, houve uma busca por alternativas para o desenvolvimento.

Após o levantamento bibliográfico e consolidação do conhecimento envolvendo a *blockchain*, foram realizadas pesquisas, por formas de implementações capazes de suprir as necessidades do projeto, e nesta fase, foram identificadas

[24] https://github.com/blockchain-certificates/cert-issuer. Acesso em: 20 janeiro 2019.
[25]https://github.com/blockchain-certificates/cert-issuer/blob/master/docs/docker_install.md. Acesso em: 20 janeiro 2019.

algumas API's, as quais se adequavam, no entanto, a partir delas também foi possível perceber a instabilidade envolvendo esses mecanismos. Um exemplo, foi a ferramenta *Tierion*[26], o qual consistia em uma API muito similar a *Smartbit* e a *Blockcypher*, porém, foi descontinuada e alterada para a atual *Chainpoint*.

Apesar das adversidades, ainda existia a possibilidade de uso direto do cliente do Bitcoin, no entanto, além de necessitar um grande espaço de armazenamento, envolve o desenvolvimento de *scripts* diretamente, implicando em códigos complexos e muito suscetíveis a falhas, que proporcionam gastos desnecessários e até mesmo o *blockchain bloat.*

Em decorrer do cenário descrito, a busca de bibliotecas e API's que fazem a abstração do uso da *blockchain*, se mostraram mais viáveis e adaptáveis ao desenvolvimento da aplicação, desta forma, foi possível realizar o desenvolvimento do protótipo e realizar os devidos registros.

As próximas seções descrevem algumas características, identificadas durante a pesquisa e desenvolvimento, que podem prejudicar ou dificultar a viabilidade desse tipo de aplicação e ainda podem se apresentar como cruciais para um seu bom desempenho.

8.1 IDENTIFICAÇÃO DO EMISSOR

Apesar de ser possível realizar um registro de forma imutável na *blockchain*, seus dados ainda precisam de uma verificação individual.

Mesmo após um registro ser apresentado, a única forma de identificar qual instituição realmente o emitiu, é por meio da chave pública utilizada na transação. Deste modo, se torna inevitável para que as instituições as evidenciem em seus *web-sites* ou com a publicação de documentos oficiais (GRÄTHER et al., 2018, tradução nossa).

Uma possível proposta, para melhor busca e segmentos dos registros, é a utilização de determinadas carteiras para cada curso aplicado por ela, desta forma,

[26] https://tierion.com/. Acesso em: 05 junho 2019.

quando um usuário necessitar verificar se determinada pessoa realmente possui um diploma em determinado curso, por exemplo, em Ciência da Computação, ele pode acessar o acessar o site da universidade e verificar a chave pública utilizada para a emissão de diplomas, referentes a graduação em questão.

8.2 AQUISIÇÃO E VALOR DA MOEDA

A prática de adicionar saldo as carteiras desejadas, pode ser vista como um contratempo, sabendo que não é possível adquirir moedas BTC diretamente com bancos, e sim, somente com o uso de *exchanges,* os quais são plataformas direcionadas para venda e compras de moedas, além disso, apesar da diversidade de escolha, ainda existem dificuldades em encontrar plataformas confiáveis em determinados países (ANTONOPOULOS, 2014, tradução nossa).

No Brasil, as plataformas Mercado Bitcoin[27] e Bitcoin Trade[28] são referência no mercado, no entanto, os preços praticados para compra e venda podem divergir entre essas plataformas. A tabela 6 ilustra os valores praticados por diferentes grupos na mesma data.

Tabela 7 – Valores da moeda BTC

Plataforma	Valor em reais	Data
BitcoinTrade	R$ 33.864,04	08/10/2019
Mercado Bitcoin	R$ 33.947,97	08/10/2019
Coinbase	R$ 32.573,02	08/10/2019

Fonte: Do Autor.

As atuais *exchanges* no Brasil, ainda implicam no pagamento de taxas de conversão e até mesmo taxas sobre valores depositados e sacados, com isso, foi identificada a instabilidade em estipular valores a serem pagos pelo BTC.

Outro problema observado em relação ao valor da moeda, são as altas e baixas constantes presentes no mercado. A figura 35 ilustra seu valor no período de outubro de 2017 a outubro de 2019.

[27] https://www.mercadobitcoin.com.br. Acesso em: 08 outubro 2019.
[28] https://www.bitcointrade.com.br/pt-BR/. Acesso em: 08 outubro 2019.

Figura 35 - Valor da moeda BTC no período outubro de 2017 à outubro de 2019

Fonte: TrandingView (2019).

No período em questão a variação do valor moeda foi muito grande, onde, em outubro de 2017, seu valor era de aproximadamente R$ 19.861,63, já no mês de dezembro de 2017 o valor da moeda alcançou o maior registro de sua história, ultrapassando o valor de R$69.000,00, representando uma alta de mais 260% em menos de 2 meses, no entanto, em meados de janeiro de 2018 a moeda chegou à R$26.787,38.

Devido as constantes mudanças no seu valor, é difícil estipular o gasto que a instituição acadêmica possuiria, ainda, existe a possibilidade de em determinada data, ao adicionar um valor *n* em BTC a uma carteira, ocorrer a alta ou baixa da moeda, dessa forma, acarretando possíveis maiores gastos.

8.3 INCONSISTÊNCIA DAS TAXAS

A viabilidade no ponto de vista econômico, do uso da *blockchain* para o registro de arquivos, pode ser comprometida tanto pela instabilidade do valor da

moeda, como das taxas média das transações. As figuras 36 e 37 ilustram o valor médio das transações, em um período aproximadamente de 2 anos.

Diretamente relacionado ao valor da moeda, no mesmo período do ano de 2017 em que o BTC obteve uma grande alta, o valor das taxas adequadas pelo tamanho médio de uma transação se aproximaram à US$55,00, enquanto no início do mesmo ano, o valor médio era de US$0,35 e atualmente consiste em US$0,60.

Figura 36 - Valor médio das transações entre novembro de 2017 e novembro de 2018

Fonte: BitInfoCharts (2019).

Figura 37 - Valor médio das transações entre outubro de 2018 e outubro de 2019

Fonte: BitInfoCharts (2019).

Quando ocorre um constante aumento na demanda pela moeda, a velocidade de processamento da rede passa a decair, e sabendo que sua capacidade é relativamente baixa, comparada a outros meios de pagamento, somente as transações com uma alta taxa tendem a serem processadas, consequentemente, criá-las com baixas taxas, implicam no adiamento de sua inserção em um bloco. Dessa forma, os usuários tendem a aumentar o valor das taxas, para garantir uma rápida propagação de suas operações na rede.

Para solução desse problema, seria possível a criação de uma *blockchain* exclusiva para o registro desse tipo de dados (Seção 8.5), ou ainda, o uso de uma outra rede com valores mais consistentes.

8.4 LIMITE DE DADOS E DE PROCESSAMENTO DA REDE

Os atuais ambientes que envolvem o armazenamento de dados na *blockchain*, enviam somente um identificador único do arquivo para a rede, normalmente ao executar uma função *hash* criptográfica e registrar seu resumo. Outras aplicações ainda usam a árvore de Merkle, para o registro de um *hash* que pode referenciar inúmeros outros dados (GRECH; CAMILLERI, 2017, tradução nossa).

Essa prática somente ocorre para evitar custos e devido a algumas redes, possuírem um grande limite em relação a capacidade de armazenamento de dados nos blocos.

Os gastos são gerados, pelos próprios valores a serem pagos pelas transações e ainda, quanto maior seu tamanho, maior seu custo, além disso, a capacidade de processamento, obtida pelo algoritmo de consenso PoW, resulta em um enorme consumo de energia. O processo de mineração, também consta com os custos relacionados ao uso de *hardware* especifico, com alto custo de aquisição e baixa durabilidade (PALANIVEL, 2019, tradução nossa; TSCHORSCH; SCHEUERMANN, 2016, tradução nossa; ZHENG et al. 2017, tradução nossa).

Em um ambiente ideal, o registro dos dados, deveria ser efetivado diretamente na *blockchain*, eliminando a necessidade por armazenamento em bancos de dados centralizados.

Para contornar este tipo de cenário, o uso de *Smart Contracts* (Seção 8.7), diferentes arquiteturas de rede e algoritmos de consenso se mostram como uma solução eficiente.

8.5 USO DA BLOCKCHAIN E CARACTERÍSTICAS

Como principais propostas do uso da tecnologia da *blockchain*, no ambiente de registros de diplomas, de acordo com as características apresentadas no projeto e conforme os resultados dos estudos de Gräther et al. (2018), Grech e Camilleri (2017), Palanivel (2019), Palma et al. (2019) e Turkanović et al. (2018), foi

possível alcançar a garantia de segurança dos dados e contra fraudes, longevidade, redução de processos manuais e eliminação da centralização de dados e da necessidade de um intermediador de confiança, para realizar verificações e registros.

Contudo, essas características em específico, estão mais atreladas a *blockchain* do Bitcoin, tem em vista sua longa cadeia de blocos, dificuldade de mineração e por possuir uma arquitetura pública.

Uma *blockchain* pública provê máxima descentralização e transparência, mas ao mesmo tempo, possui um armazenamento de dados ineficiente, alto custo com energia elétrica e uma velocidade de operação baixa. Contudo, o uso de outros modelos de *blockchain*, tal como as redes privadas e *consortium,* os problemas citados podem ser solucionados, porém, ao custo da alta transparência e descentralização.

O uso de uma *blockchain* privada resultaria em participantes somente convidados a rede, os quais criam as regras a serem utilizadas entre si. Esse tipo de rede permite que vários usuários possam utilizá-la, mas implica em centralização, dessa forma, acontece uma grande redução da imutabilidade provida pela tecnologia, contudo, produz uma grande eficiência em termos de custos e velocidade de operação.

Enquanto uma rede *consortium* possuiria a característica de convite similar à de uma rede privada, o consenso existe entre todos seus participantes, com isso, os conceitos de descentralização passam a existir neste modelo.

Em um ambiente educacional, existe uma grande variedade de dados a serem armazenados, os quais exigem confiabilidade, segurança e sistemas ante fraudes, sendo alguns deles: pagamentos de mensalidades, bolsas acadêmicas, históricos escolar, diplomas e certificados, cursos e registros de patentes, publicações e artigos. Em uma *blockchain*, com uso dessa grande quantidade de dados, a imutabilidade e segurança da rede se tornariam cada vez mais eficazes, de acordo com a quantidade de registros sendo inseridos (GRECH; CAMILLERI, 2017, tradução nossa; HARTHY, SHUHAIMI; ISMAILY, 2019, traduçao nossa; PALANIVEL, 2019, tradução nossa).

Dessa forma, em um cenário onde diversas universidades, façam a utilização do sistema acadêmico baseado em uma *blockchain*, o possível uso de uma arquitetura *consortium* se mostra o mais ideal.

Supondo uma rede, em que todos usuários com permissão de escrita, constituem nas instituições do país, e onde não existem restrições da permissão de leitura, o conceito de descentralização é alcançado facilmente. Com todos registros sendo aplicados a rede, a imutabilidade cresceria rapidamente e ainda, os custos de operação poderiam ser reduzidos, ao escolher um algoritmo de consenso e processamento de dados eficiente, o qual poderia se adaptar as necessidades do ambiente. No entanto, ao supor um reduzido número de registros, a rede estaria suscetível a falhas, no que se diz a respeito de fraudes.

Uma outra característica ideal para esse modelo, é que a cada novo ingresso de uma instituição à rede, um novo nó completo deve ser implementado por ela, dessa forma, resulta em um sistema de descentralização em grande escala, com segurança a respeito do armazenamento dos dados.

Ao utilizar outros modelos e diferentes tipos de *blockchain*, além do potencial descrito anteriormente, existem outras características que podem se adequar, com as inúmeras necessidades no desenvolvimento de projetos no ambiente educacional, assim como, em outras áreas.

8.6 PADRONIZAÇÃO

Ao iniciar um estudo com a *blockchain*, os pesquisadores devem levar em conta as características essenciais do projeto, as quais consistem em fatores como: quantidade de usuários, transparência, permissões, resistência a fraudes, velocidade de processamento, assim como, os tipos de dados e quantidade a serem registrados e entre outros fatores (PALANIVEL, 2019, tradução nossa).

A adoção em larga escala, de um ambiente educacional utilizando a *blockchain*, implica diretamente em padronizações no sistema, com isso, proporcionando a demanda por acordos, estruturas bem definidas e capacidade de comunicação com outros sistemas.

Porém, definir padrões em uma tecnologia relativamente nova e com grande crescimento, poderia ocasionar em resultados prejudiciais, onde, as diferentes organizações estariam competindo por melhorias e inovação, consequentemente, resultando em fragmentações e cada vez mais numerosas topologias, com isso, os primeiros usos da tecnologia podem ser afetados por futuros padrões e evoluções, os quais podem ocasionar mudanças e divergências, trazendo custos e inconsistência nos modelos previamente utilizados (HANSON, 2017, tradução nossa).

As características citadas acima, foram identificadas e confirmadas nos estudos de Grech e Camilleri (2017), Palanivel (2019) e Turkanović et al. (2018), no entanto, elas não resultam na incapacidade da adoção da tecnologia, e sim, na urgência da necessidade de padronização e definições bem estabelecidas no seu uso, tendo em vista que as atuais plataformas que utilizam esse tipo de sistema, ainda fazem uso de *software* proprietário tanto para o registro, quanto, para verificação de dados emitidos.

Em decorrer da necessidade padronização, deve ser evitado a criação de plataformas com código proprietário, tendo em vista, que no atual cenário, a plataforma *Blockcerts* existe justamente para reduzir essa prática. Ela foi desenvolvida e distribuida como código livre, e um dos principais de seus ideais é promover seu uso para qualquer aplicação que realiza registros na *blockchain*.

Ainda existe a padronização em alguns dos modelos de dados a serem utilizados, em especial, os diplomas e certificados, os quais poderiam ser alcançados com o uso do padrão *Open Badges*.

Ao decorrer do projeto *Blockcerts*, essas características foram levantadas e vistas como cruciais para uma adoção em larga escala. Neste trabalho, uma pesquisa foi realizada para verificar a possibilidade de seu uso, no entanto, ele necessita de licenças. Na pesquisa realizada por Gräther et al. (2018), o uso de dados digitais envolvendo o mesmo padrão também foi adotado.

Esse modelo utiliza objetos no formato JSON, permitindo um fácil consumo e integração com várias plataformas, além disso, possui um conteúdo básico para uso (figura 38), mas permite a inserção de uma lista de dados conforme as diferentes necessidades.

Cada tipo de dado, atribuído ao modelo *Open Badges*, possui um valor agregado de grande importância, dessa forma, seu site oficial possui algumas instruções com a descrição adequada para cada modelo (IMS, 2018, tradução nossa).

Figura 38 – *Open Badges* modelo básico


```
{
  "@context": "https://w3id.org/openbadges/v2",
  "id": "https://example.org/assertions/123",
  "type": "Assertion",
  "recipient": {
    "type": "email",
    "identity": "alice@example.org"
  },
  "issuedOn": "2016-12-31T23:59:59+00:00",
  "verification": {
    "type": "hosted"
  },
  "badge": {
    "type": "BadgeClass",
    "id": "https://example.org/badges/5",
    "name": "3-D Printmaster",
    "description": "This badge is awarded for passing the 3-D printing knowledge and safety test.",
    "image": "https://example.org/badges/5/image",
    "criteria": {
      "narrative": "Students are tested on knowledge and safety, both through a paper test and a supervised performance evaluation
    },
    "issuer": {
      "id": "https://example.org/issuer",
      "type": "Profile",
      "name": "Example Maker Society",
      "url": "https://example.org",
      "email": "contact@example.org",
      "verification": {
        "allowedOrigins": "example.org"
      }
    }
  }
}
```


Fonte: IMS (2018).

8.7 DADOS DIGITAIS E PROCESSOS MANUAIS

O projeto desenvolvido realiza a ancoragem na *blockchain*, de um diploma ou certificado emitido por uma IES para um determinado acadêmico, o qual, foi previamente definido, criado e emitido por um fluxo estabelecido de acordo com a preferência da instituição (Capítulo 2).

Atualmente, a emissão desses documentos em formatos físicos ainda permanece em grande escala, porém, seu ciclo de interesse pode ocasionar discrepâncias, falhas e lentidão (GRÄTHER et al., 2018, tradução nossa).

Uma das propostas do uso da *blockchain*, é a redução e até mesmo eliminação dos processos manuais, envolvendo o clico de vida desses tipos de dados, no entanto, o projeto desenvolvido necessita da prévia criação de um documento digital para ser capaz de realizar seu registro, porém, para possuir a capacidade de reduzir esses processos a um número quase nulo, o uso de *Smart Contracts* se torna inevitável.

Este tipo de contrato, consiste em um *script* auto executável hospedado na *blockchain*, o qual somente é executado quando ambas as partes que o envolvem

aceitam seus termos, normalmente seu conteúdo é verificado e aceito por meio de uma assinatura digital (ZHENG et al. 2017, tradução nossa)

Sua eficácia pode eliminar totalmente a necessidade de processos manuais, que envolvem o ambiente de registro de certificados e diplomas. Sabendo que um contrato bem desenvolvido e estruturado, pode emitir e atualizar dados sobre o acadêmico automaticamente, proporcionando a redução de custos e de processos que consomem muito tempo (GRECH; CAMILLERI, 2017, tradução nossa).

Uma adversidade neste contexto, é que nem toda *blockchain* foi desenvolvida com suporte a essa tecnologia, como o caso do Bitcoin, enquanto outras redes, como a da criptomoeda Ethereum, fornecem um ambiente de desenvolvimento completo a eles (PALMA et al., 2019, tradução nossa).

Palma et al. (2019) também descreve um breve processo da desenvoltura desses contratos, onde, devem ser programados de forma a verificar se um estudante, completou todos requisitos necessários para concluir o curso ou graduação que ele vem seguindo, quando esse contrato é executado e confirmado, um segundo contrato é iniciado para realizar a emissão do diploma, o qual pode vir a ser cancelado pela IES, dessa forma, a correção e prevenção de erros existe nesse tipo de sistema.

Palma ainda especifica uma forma do controle, para autorizar a determinadas instituições, quais tipos de cursos elas estão aptas a realizar graduações. Esse levantamento se dá no âmbito educacional brasileiro, onde as instituições são autorizadas e mantidas por uma agência regulatória, a qual seria responsável por criar *Smart Contracts,* para estabelecer quais contratos podem ser exercidos. Consequentemente, esse mecanismo permite facilmente identificar as universidades, dentro da *blockchain*, por meio de uma chave pública previamente solicitada e obtida.

8.8 DEPENDÊNCIA EM TERCEIROS

De acordo com as características levantadas no Capítulo 2, é possível identificar a interferência humana, em todo ciclo de vida e de interesse no contexto abordado, e essa prática pode gerar falhas, injustiças, fraudes, altos custos e

processos lentos e mal organizados. Em ambos cenários onde uma instituição realiza a emissão dessas dados atribuídos a papel, ou, quando usam diplomas digitais não filiados a *blockchain*, implicam na dependência de terceiros tanto para realizar a verificação, assim como, para manter dados cruciais (GRÄTHER et al., 2018, tradução nossa; GRECH; CAMILLERI, 2017, tradução nossa; PALMA et al., 2019, tradução nossa).

Como explorado neste projeto, o uso de ambiente filiado a *blockchain*, pode reduzir e até eliminar esse tipo dependência, porém, somente com o uso de *Smart Contracts*, no entanto, para as organizações manterem esse tipo de sistema, um problema relacionado a terceiros ainda persiste.

Atualmente as formas mais acessíveis de se adquirir criptomoedas, é com o uso de uma plataforma *exchange,* e ao realizar este tipo de procedimento, o usuário mantém uma relação de submissão a ela, onde, sua carteira é mantida e armezenada, portanto, a plataforma se torna “proprietária” de dados, os quais devem ser particulares e privados.

Dessa forma em um cenário onde ocorra um vazamento de dados, vários registros podem ser realizados com a carteira da instituição.

8.9 CONEXÃO COM USUÁRIOS

Apesar do tipo de sistema, apresentado neste trabalho, possuir vários benefícios para todos presentes no ambiente educacional, um problema, identificado em um projeto similar, pode comprometer sua adoção em grande escala.

Tal problema foi caracterizado na adoção do uso do padrão *Blockcerts* no instituto MIT, onde, identificou-se que apesar das grandes vantagens com o atual sistema utilizado, a eliminação dos processos manuais só poderia ocorrer, a partir do momento, em que os acadêmicos adotem os recursos do meio digital.

Dessa forma, a instituição realizou alguns testes, com o objetivo de levantamento de dados em relação a este cenário. Com isso, é possível perceber que apesar do objetivo deste trabalho solucionar alguns impasses no atual ambiente de

emissão de diplomas, somente com a aprovação e utilização de usuários sua implantação se tornaria adequada.

9 CONCLUSÃO

A partir do levantamento bibliográfico e da implementação do projeto, foi possível reconhecer as principais propostas de uma *blockchain*, a segurança, imutabilidade e descentralização.

Com a finalização do trabalho em questão, foi possível observar uma melhor compreensão, relacionada as tecnologias utilizadas para o desenvolvimento do protótipo, assim como, obter noção da composição e da importância dos dados mantidos em um ambiente educacional.

Durante o levantamento do material bibliográfico, relacionado ao uso da *blockchain* para operações não financeiras e como um serviço de prova de existência, no âmbito educacional, o MEC publicou uma portaria (Seção 2.2), responsável por adaptar o atual ambiente de emissão de diplomas do Brasil ao meio digital.

Por consequência, uma certa insegurança em relação ao propósito deste trabalho foi apresentada, contudo, seu objetivo não é contrapor os atuais cenários presentes nas instituições, e sim, levantar dados e características para mostrar as atuais lacunas e irregularidades presentes, desta forma expondo uma solução com maior eficiência e segurança.

Ao longo do projeto, algumas dificuldades foram encontradas com relação a parte prática, tendo em vista a complexidade de se adequar as convenções de programação presentes na comunidade de desenvolvedores da *blockchain*, assim como, a adaptação das diversas técnicas de criptografia agregadas a esse tipo de rede. As técnicas de criptografias utilizadas na *blockchain* garantem segurança, transparência, longevidade e imutabilidade, fazendo uso de um mecanismo que possui a tendência de cada vez mais, eliminar a possibilidade de adulteração de dados.

Ela também foi desenvolvida, com o propósito de eliminar a necessidade de confiança em um terceiro, para realização de pagamentos eletrônicos, ao transferir essa responsabilidade para criptografia. Dadas suas características, a tecnologia se tornou um objeto de estudo para diversas áreas, não relacionadas a cenários monetários, as quais foram abordadas neste estudo, com o objetivo de reduzir a

interferência humana nos processos de verificação e compartilhamento dos diplomas, assim como, criar um cenário, diferentemente do atual, onde nos atuais métodos, existe a necessidade de confiança em terceiros para poder garantir a segurança destes documentos.

Neste trabalho, o protótipo criado, permite os registros de arquivos da *blockchain* do protocolo Bitcoin, criando uma prova irrefutável de sua existência em determinada data. Este registro consiste na inserção de um identificador único do arquivo em uma transação, o qual é calculado pela função *hash* criptográfica SHA-256, dessa forma, é possível verificar sua existência ao enviar para aplicação um determinado arquivo, e então por meio de seu resumo, uma busca é feita na rede. Caso encontrado, o usuário tem acesso às informações, e ao comparar se o identificador do arquivo existe na transação exibida, é possível afirmar sua existência.

Uma vez que registrado o arquivo, não existe a necessidade de manter processos de verificação centralizados, devido a confiabilidade na rede e na aplicação permitirem que qualquer usuário, com o arquivo original, possa provar sua existência de forma independente. Caso o usuário escolha verificar um arquivo sem o uso da aplicação, ele pode calcular o seu resumo com a função SHA-256 e usar uma plataforma *block explorer,* para realizar sua busca pela *blockchain*.

Os dados relacionados ao registro, armazenados no cabeçalho do bloco em que a transação foi inserida, possuem garantia de imutabilidade, tendo em vista as propriedades criptográficas alcançadas com a *blockchain*. Quanto a integridade de um arquivo, ela é obtida de modo que a menor alteração possível no arquivo original, resultaria em identificador único, sem qualquer relação ao registro feito.

As características obtidas pelos registros, foram alcançadas com o uso da *blockchain* pública do Bitcoin, a qual teve escolha realizada, de acordo com o levantamento bibliográfico, tendo como objetivo encontrar uma rede com características ideais para manter um registro imutável, sem a necessidade de centralização dos dados.

Para realizar o efetivo registro de um arquivo em uma transação, um usuário previamente necessita de uma chave privada, uma chave pública e um endereço Bitcoin válido, portanto, os processos para criação destes dados foram

implementados no protótipo. O endereço utilizado ainda precisa possuir um saldo mínimo de 1000 *sathoshis* para ser pago como taxa para o minerador que incluir a transação gerada em um bloco.

Portanto, para a possível compreensão das técnicas e procedimentos utilizados neste processo, um estudo e detalhamento a respeito do protocolo Bitcoin e consequentemente, da *blockchain*, foi realizado ao longo deste trabalho.

Por fim, os detalhes e características obtidos com o desenvolvimento da aplicação foram apresentados e com relação ao ponto de vista técnico, como um todo e com base nos testes realizados, obteve-se um resultado satisfatório quanto a integração entre todas as tecnologias implementadas.

9.1 TRABALHOS FUTUROS

Embora o presente trabalho possa ser utilizado de forma efetiva, para trabalhos futuros, recomendam-se levantamento de dados específicos dentro de uma instituição educacional e realizar a tentativa de adequação a esses processos, para o uso da *blockchain*.

Ainda, este estudo criou uma plataforma com código proprietário, o que não é a forma mais adequada, para a adoção desta tecnologia no ambiente educacional, portanto, é possível realizar um estudo nas atuais formas de padronização e a aplicar o código desenvolvido pela plataforma *Blockcerts*, consequentemente, contribuindo da forma vista como mais ideal para este tipo de cenário.

O protótipo desenvolvido, necessita de uma API para realizar a comunicação com a *blockchain*, porém, esse tipo de ferramenta é muito instável, o que poderia tornar sua utilização obsoleta, portanto, é proposto a criação de uma API para suprir essa necessidade, ainda, sobre este estudo, há uma necessidade de melhoria na abordagem de dados, como a própria verificação da relação entra a chave pública do registro com a chave da instituição. Além disso, a tentativa de adequar o uso do padrão *Open Badges* para criar o diploma de forma digital, é uma boa forma de reduzir ainda mais os processos manuais que envolvem a criação do diploma ou certificado.

Para conseguir eliminar o envolvimento com interferência humana e dependência por terceiros, em todo ciclo de interesse dos diplomas e certificados, sendo ele, a partir do levantamento de dados referente aos acadêmicos que possuem requisitos para concluírem seus cursos, a própria criação e armazenamento do diploma, e todo seu processo de verificação e compartilhamento, o uso de uma plataforma com suporte a *Smart Contracts* é inevitável, portanto, é proposto um estudo em relação as atuais *blockchain*s que suportam esta tecnologia, assim como o levantamento de características, para verificar qual delas é a mais adequada para esse tipo de processo, e por fim, criar um protótipo que realize estes procedimentos.

Da mesma forma, também podem ser levados em consideração as sugestões abaixo para pesquisa de trabalhos com o uso da *blockchain*:

a) análise da eficiência das diferentes topologias de rede da *blockchain* e o de algoritmos de consenso com melhor adequação a um serviço de prova de existência;
b) busca e levantamento de dados para o escalonamento e adoção de um sistema de registro de diplomas com a *blockchain* em nível nacional;
c) caracterização dos processos praticados dentro de uma instituição e propor soluções com o uso da *blockchain* tendo em vista suas possíveis falhas e irregularidades.

REFERÊNCIAS

ANTONOPOULOS, A. M. **Mastering bitcoin:** unlocking digital cryptocurrencies. 1ª. ed. [S.l.]: O'Reilly Media, 2014.

ARELLANO, M. Á. M. Preservação de Documentos Digitais, 2004. Disponível em: http://revista.ibict.br/ciinf/article/view/1043. Acesso em: 2019 Maio 12.

BITCOIN Fees. **Bitcoin Transaction Fees**, 2019. Disponível em: https://bitcoinfees.info/. Acesso em: 25 Maio 2019.

BITCOINCORE, 2019. Disponível em: https://bitcoincore.org/en/download/. Acesso em: 12 Outubro 2019.

BITINFOCHARTS, 2019. Disponível em: https://bitinfocharts.com/. Acesso em: 08 Outubro 2019.

BITNODES. GLOBAL BITCOIN NODES DISTRIBUTION. Disponível em: https://bitnodes.earn.com/. Acesso em: 19 Maio 2019.

BITCOIN PROJECT. Bitcoin Developer Guide. **Bitcoin Project**, 2019. Disponível em: https://bitcoin.org/en/developer-guide#. Acesso em: 06 Abril 2019.

BLOCKCERTS. Disponível em: https://www.blockcerts.org/. Acesso em: 2019 Abril 13.

BLOCKCYPHER. **Blockcypher**, 2019. Disponível em: https://www.blockcypher.com. Acesso em: 08 Agosto 2019.

BONNEAU, J.; MILLER, A.; CLARK, J.; NARAYANAN, A.; KROLL, J. A.; FELTEN, E. W. SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies. **IEEE Symposium on Security and Privacy**, San José, CA, 2015. 104-121.

BRASIL. Lei n. 8.159, de 8 de jan. de 1991. **Política nacional de arquivos públicos e privados e dá outras providências**, 8 jan. 1991. Disponível em: http://www.planalto.gov.br/ccivil_03/leis/l8159.htm. Acesso em: Abril 27 2019.

BRASIL. Lei, n. 9.394, de 20 de dez. de 1996. **Diretrizes e bases da educação nacional**, 20 dez. 1996. Disponível em: https://www2.senado.leg.br/bdsf/bitstream/handle/id/70320/65.pdf. Acesso em: 27 Abril 2019.

BRASIL, Ministério da Educação. Portaria nº 1.095, 25 de outubro de 2018. **Expedição e o registro de diplomas de cursos superiores de graduação no âmbito do sistema federal de ensino**, 2018. Disponível em: a expedição e o registro de diplomas de cursos superiores de graduação no âmbito do sistema federal de ensino. Acesso em: 27 Abril 2019.

BRASIL, Ministério da Educação. Portaria nº 554, de 11 de março de 2019. **Dispõe sobre a emissão e o registro de diploma de graduação, por meio digital, pelas Instituições de Ensino Superior - IES pertencentes ao Sistema Federal de Ensino**, 2019. Disponível em: http://www.in.gov.br/materia/-/asset_publisher/Kujrw0TZC2Mb/content/id/66544171/do1-2019-03-12-portaria-n-554-de-11-de-marco-de-2019-66543842. Acesso em: 27 Abril 2019.

CHAINPOINT. ChainPoint, 2019. Disponível em: https://chainpoint.org/. Acesso em: 08 Agosto 2019.

CHAUM, D. Blind signatures for untraceable payments. **SPRINGER. Advances in cryptology**, 1982. 199–203.

CHAUM, D.; FIAT, A.; NAOR, M. Untraceable electronic cash. **SPRINGER-VERLAG NEW YORK. Proceedings on Advances in cryptology**, 1990. 319-327.

CHENG, Jiin-Chiou; LEE, Narn-Yih; CHEN, Yi-Hua. **Blockchain and smart contract for digital certificate**. IEEE International Conference on Applied System Invention (ICASI). Chiba: [s.n.]. 2018. p. 1046-1051.

CONOSCENTI, M.; VETRÒ, A.; MARTIN, J. C. D. Blockchain for the Internet of Things: A systematic literature review. **IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)**, Agadir, p. 1-6, 2016.

COSTA, Rostand; FAUSTINO, Daniel; LEMOS, Guido; QUEIROGA, Ademir; DJOHNNATHA, Cláudio; ALVES, Felipe; LIRA, Jordan; PIRES, Mateus. Uso Não Financeiro de Blockchain: Um Estudo de Caso Sobre o Registro, Autenticação e Preservação de Documentos Digitais Acadêmicos. **Workshop em Blockchain: Teoria, Tecnologias e Aplicações (WBlockchain_SBRC)**, v. 1, n. 1/2018, Maio 2018. Disponível em: http://ojs.sbc.org.br/index.php/wblockchain/article/view/2356.

CROSBY, Michael; NACHIAPPAN; PATTANAYAK, Pradhan; VERMA, Sanjeev; KALYANARAMAN, Vignesh. BlockChain Technology. **Applied Innovation Review**, v. 2, 2016. Disponível em: https://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf.

DECKER, C.; WATTENHOFER, R. Information Propagation in the Bitcoin Network. **IEEE P2P 2013 Proceedings**, Trento, 2013. 1-10.

DOCUSIGN. **Understanding digital signatures**. Disponível em: https://www.docusign.com/how-it-works/electronic-signature/digital-signature/digital-signature-faq. Acesso em: 18 Maio 2019.

DURANT, E.; TRACHY, A. MIT News. **Digital Diploma debuts at MIT**, 2017. Disponível em: http://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017. Acesso em: 05 Outubro 2019.

E-ESTONIA. **e-governance**. Disponível em: https://e-estonia.com/solutions/e-governance/. Acesso em: Maio 18 2019.

EXPRESS , 2019. Disponível em: https://expressjs.com/pt-br/. Acesso em: 20 Julho 2019.

FLOSE, V. B. S. **Criptografia e Curvas Elípticas**. Tese de Mestrado - Universidade Estadual Paulista. Campus de Rio Claro, SP. 2011.

GAEINI, A.; GHAFFARI, M. H.; MOSTAGHIM, Z. An Improved Hash Function Based on the Tillich-Zémor Hash Function. **Mathematics Interdisciplinary Research 3**, 2018.

GANDINI, J. A. D.; SALOMÃO, D. P. D. S.; JACOB, C. A SEGURANÇA DOS DOCUMENTOS DIGITAIS, 2001.

GIMENES, E. MP investiga o uso de mais de 500 diplomas falsos em Maringá. **Gazeta do Povo**, 2013. Disponível em: https://www.gazetadopovo.com.br/vida-e-cidadania/mp-investiga-o-uso-de-mais-de-500-diplomas-falsos-em-maringa-045t978j7yjeuhthz32g673gu/ .

GLOBO. Dono de universidade denuncia esquema de venda de diplomas falsos por R$ 550 em MT. **Globo**, 2017. Disponível em: https://g1.globo.com/mato-grosso/noticia/dono-de-universidade-denuncia-esquema-de-venda-de-diplomas-falsos-por-r-550-em-mt.ghtml.

GOYA, D. et al. **Modelos de Criptografia de Chave Pública Alternativos**. [S.l.]. 2009.

GRÄTHER, Wolfgang; KOLVENBACH, Sabine; RULAND, Rudolf; SCHÜTTE, Julian; TORRES, Christof Ferreira; WENDLAND, Florian. Blockchain for Education: Lifelong

Learning Passport. In: (EUSSET), E. S. F. S. E. T. **Proceedings of 1st ERCIM Blockchain Workshop 2018, Reports of the European Society for Socially Embedded Technologies**. Amsterdam: [s.n.], v. 2, 2018.

GRECH, A.; CAMILLERI, A. F. **Blockchain in Education**. Luxemburgo. 2017.

HANSON, R. T. **Distributed Ledgers, Scenarios for the Australian**. Commonwealth Scientific and Industrial Research Organisation. Canberra. 2017.

HARTHY, K. A.; SHUHAIMI, F. A.; ISMAILY, K. K. J. A. **The upcoming Blockchain adoption in Higher-education:** requirements and process. International Conference on Big Data and Smart City (ICBDSC). Muscat: [s.n.]. 2019. p. 1-5.

IMS. IMS Global Learning Consortium. **Open Badges v2.0**, 2018. Disponível em: https://www.imsglobal.org/sites/default/files/Badges/OBv2p0Final/index.html. Acesso em: 13 Outubro 2019.

JIAN-DONG, Liu; YE, Tian; SHU-HONG, Wang; KAI, Yang. A fast new one-way cryptographic hash function. **IEEE International Conference on Wireless Communications, Networking and Information Security**, Beijing, 2010. 302-306.

JUNIOR, A. J. D. Almeida. **Criptossistemas baseados em curvas elípticas estudo de casos e implementação em processador de sinais digitais**. Tese Mestrado - Universidade Estadual de Campinas. [S.l.]. 2002.

JSON, 2019. Disponível em: https://www.json.org/. Acesso em: 05 outubro 2019.

KARAME, G.; ANDROUKALI, E. **Bitcoin and Blockchain Security**. Norwood: Artech House, v. I, 2016.

LEAO, H. A. T.; CANEDO, E. D.; GOMES, A. R. **Proposta de Uso do Blockchain para Validaçao de Documentos de Instituiçoes de Ensino Superior**. Brasília. 2017.

LEDGER. **Ledger Nano S**. Disponível em: https://www.ledger.com/products/ledger-nano-s. Acesso em: 31 Março 2019.

LEMOS, A. H. C.; DUBEUX, V. J. C.; PINTO, M. C. S. Educação, empregabilidade e mobilidade social: convergências e divergências. **Cadernos EBAPE**, Rio de Janeiro, 7, n. 2, Junho 2009. 368-384.

MACHINE Learning. **Issue verifiable digital records**. Disponível em: https://www.learningmachine.com/.

MARCACINI, A. T. R. O DOCUMENTO ELETRÔNICO COMO MEIO DE PROVA, 1999. Disponível em: http://www.egov.ufsc.br/portal/sites/default/files/anexos/13948-13949-1-PB.htm.

MARCONDES, C.; SAYÃO, L. Documentos digitais e novas formas de cooperação entre sistemas de informação em c&t. **Revista Ciência da Informação**, Brasília, v. 31, n. 3, p. 42-54, 2002.

MARTINS, T. F. **Prova de existência de arquivos digitais utilizando a tecnologia blockchain do protocolo Bitcoin**. Trabalho de Conclusão de Curso - Engenharia da Computação, Universidade Federal do Rio Grande do Sul. Porto Alegre, RS. 2018.

MCEVOY, R. P. et al. Optimisation of the SHA-2 Family of Hash Functions on FPGAs, Karlsruhe, 2006.

MDN. MDN web docs. **HTTP**, 2019. Disponível em: https://developer.mozilla.org/en-US/docs/Web/HTTP. Acesso em: 05 Outubro 2019.

MEC e Inep divulgam dados do Censo da Educação Superior 2016. **Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira**, 2016. Disponível em: http://portal.inep.gov.br/artigo/-/asset_publisher/B4AQV9zFY7Bv/content/mec-e-inep-divulgam-dados-do-censo-da-educacao-superior-2016/21206.

MENKE, F. ASSINATURA DIGITAL, CERTIFICADOS DIGITAIS, INFRAESTRUTURA DE CHAVES PÚBLICAS BRASILEIRA E A ICP ALEMA. Disponível em: http://www.egov.ufsc.br/portal/sites/default/files/anexos/4375-4369-1-PB.pdf. Acesso em: 23 Março 2019.

MERCADO Pago, 2019. Disponível em: https://www.mercadopago.com.br/. Acesso em: 24 Março 2019.

MERKLE, R. C. A digital signature based on a conventional encryption. **Proc. 7th Conf. Adv. Cryptol. (CRYPTO'87)**, Agosto 1987. 369-378.

MERTZ, L. (Block) Chain Reaction: A Blockchain Revolution Sweeps into Health Care, Offering the Possibility for a Much-Needed Data Solution. **IEEE Pulse**, v. 9, p. 4-7, 2018.

METTLER, M. **Blockchain technology in healthcare:** The revolution starts here. IEEE 18th International Conference on e-Health Networking, Applications and Services. Munich: (Healthcom). 2016. p. 1-3.

MILLER, D. Blockchain and the Internet of Things in the Industrial Sector. In: IEEE **IT Professional**. [S.l.]: [s.n.], v. 20, 2018. p. 15-18.

MIT MEDIA LAB. **Certificates, Reputation, and the Blockchain**, 2015. Disponível em: https://medium.com/mit-media-lab/certificates-reputation-and-the-blockchain-aee03622426f. Acesso em: 14 Abril 2019.

MIT MEDIA LAB. What we learned from designing an academic certificates system on the blockchain, 2016. Disponivel em: https://medium.com/mit-media-lab/what-we-learned-from-designing-an-academic-certificates-system-on-the-blockchain-34ba5874f196. Acesso em: 1 Novembro 2019.

MIT News. **Digital Diploma debuts at MIT**, 2017. Disponível em: https://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017.

MOURA, T. Secretaria de Educação pode levar até 180 dias para investigar diplomas falsos no ES. **Globo**, 2017. Disponível em: https://g1.globo.com/espirito-santo/educacao/noticia/secretaria-de-educacao-pode-levar-ate-180-dias-para-investigar-diplomas-falsos-no-es.ghtml.

NAKAMOTO, S. Bitcoin: A Peer-to-Peer Electronic Cash System, 2008. Disponível em: https://bitcoin.org/bitcoin.pdf.

NIMFUEHR, M. Hackernoon. **The Amazing Story of Cryptocurrencies Before Bitcoin**, 12 Novembro 2018. Disponível em: https://hackernoon.com/the-amazing-story-of-cryptocurrencies-before-bitcoin-fe1b0e55155b. Acesso em: 24 Março 2019.

NIST. SECURE HASH STANDARD, FIPS PUB 180-2, 2002.

NODE.JS , 2019. Disponível em: https://nodejs.dev/. Acesso em: 20 Julho 2019.

OLIVEIRA, J. G. D. **Curvas Elípticas sobre Corpos Finitos e Criptografia de Chave**. Universidade Federal de Mato Grosso do Sul. [S.l.]. 2009.

OPEN BADGES. **Open Badges**. Disponível em: https://openbadges.org/. Acesso em: 14 Abril 2019.

PALANIVEL, K. Blockchain Architecture to Higher Education Systems. **International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS),** v. III, n. II, Fevereiro 2019.

PALMA, Lucas M.; VIGIL, Martín A. G.; PEREIRA, Fernando L.; MARTINA, Jean E. Blockchain and smart contracts for higher education registry in Brazil. **International Journal of Network Management**, 2019.

PAYPAL , 2019. Disponível em: https://www.paypal.com/br/home. Acesso em: 24 Março 2019.

PRENEEL, B.; DOBBERTIN, H.; BOSSELAERS, A. The Cryptographic Hash Function RIPEMD-160. **Appeared in CryptoBytes 3**, 1997. 9-14.

PORTNOI, M. **CRIPTOGRAFIA COM CURVAS ELÍPTICAS**. UNIVERSIDADE SALVADOR – UNIFACS. Salvador, BA. 2005.

POSTMAN. **About Postman**, 2019. Disponível em: https://www.getpostman.com/about-postman. Acesso em: 20 Julho 2019.

RAMALHO, José Carlos; FARIA, Luís; FERREIRA, Miguel; CASTRO, Rui. **Relational Database Preservation through XML modelling**. Extreme Markup Languages 2007. Montréal: [s.n.]. 2007.

RIVEST, R. L.; SHAMIR, A.; ADLEMAN, L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. **Communications of the ACM**, v. 21, n. 2, p. 120-126, 1978.

RNP - Rede Nacional de Ensino e Pesquisa. **UFPB realiza solenidade de entrega dos primeiros diplomas universitários digitais do país**, 2019. Disponível em: https://www.rnp.br/noticias/ufpb-realiza-solenidade-entrega-primeiros-diplomas-universitarios-digitais-pais. Acesso em: 14 Abril 2019.

RUUSALEPP, R.; DOBREVA, M. Digital Preservation Services: State of the Art Analysis, 2012. Disponível em: https://www.um.edu.mt/library/oar/bitstream/handle/123456789/311/dobreva_Preservation_Services_study.pdf?sequence=1&isAllowed=y. Acesso em: 23 Março 2019.

SALLES, S. Cinco anos após fim da Gama Filho, abandono do campus simboliza decadência do bairro de Piedade. **Blog do Acervo - O Globo**, 2019. Disponível em: https://blogs.oglobo.globo.com/blog-do-acervo/post/cinco-anos-apos-fim-da-gama-

filho-abandono-do-campus-simboliza-decadencia-do-bairro-de-piedade.html. Acesso em: 2019 Abril 27.

SCHMIDT, P. Certificates, Reputation, and the Blockchain. **MIT Media Lab**, 2015. Disponível em: https://medium.com/mit-media-lab/certificates-reputation-and-the-blockchain-aee03622426f. Acesso em: 19 abr. 2019.

SHARMA, A. K.; MITTAL, D. S. K.; MITTAL, D. S. Attacks on Cryptographic Hash Functions and Advances. **INTERNATIONAL JOURNAL OF INFORMATION AND COMPUTING SCIENCE2**, 2018.

SKINNER, K.; SCHULTZ, M. **A Guide to Distributed Digital Preservation**. Atlanta: Educopia Institute, 2010.

SMARTBIT, 2019. Disponível em: https://www.smartbit.com.au/. Acesso em: 09 Agosto 2019.

SMOLENSKI, N. **ACADEMIC CREDENTIALS IN AN ERA OF DIGITAL DECENTRALIZATION**. [S.l.]. 2016.

SOARES, D. Trabalhador Digital. **Bitcoin:** O que é e como funciona? Quanto vale o Satoshi?, 2019. Disponível em: https://www.trabalhadordigital.com/2018/08/bitcoin-satoshi.html. Acesso em: 06 Outubro 2019.

SOBRINHO, Ranulfo Paiva; GARCIA, Junior Ruiz; MAIA, Alexandre Gori; ROMEIRO, Ademar Ribeiro. **TECNOLOGIA 'BLOCKCHAIN': INOVAÇÃO EM PAGAMENTOS POR SERVIÇOS AMBIENTAIS (PSA)**. 55º Congresso da SOBER. Santa Maria, p. 1-17. 2017.

SWARD, A.; VECNA, I.; STONEDAHL, F. Data Insertion in Bitcoin’s Blockchain. **Ledger**, v. 3, 2018.

SZABO, N. Bit Gold, 2005. Disponível em: https://nakamotoinstitute.org/bit-gold/. Acesso em: 24 Março 2019.

TANENBAUM, A. S. **Redes de computadores**. Tradução de Vandenberg D. de Souza. 4. ed. Amsterdam: Campus, 2003.

TRANDINGVIEW , 2019. Disponível em: https://br.tradingview.com/. Acesso em: 08 Outubro 2019.

TSCHORSCH, F.; SCHEUERMANN, B. Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. **IEEE Communications Surveys & Tutorials**, 2016. 2084-2123.

TURKANOVIć, M. et al. EduCTX: A Blockchain-Based Higher Education Credit Platform. **IEEE Access**, 2018. 5112-5127.

UNIC BLOCKCHAIN INITIATIVE. **Academic Certificates on the Blockchain**. Disponível em: https://digitalcurrency.unic.ac.cy/free-introductory-mooc/self-verifiable-certificates-on-the-bitcoin-blockchain/academic-certificates-on-the-blockchain/. Acesso em: 14 Abril 2019.

USHMANI, A. Blockchain Insight. **International Journal of Computer Science Trends and Technology** , Abril 2019.

VALLOIS, V.; GUENANE, F. A. Bitcoin transaction: From the creation to validation, a protocol overview. **1st Cyber Security in Networking Conference (CSNet)**, Rio de Janeiro, 2017. 1-7.

VISA. Visa acceptance for retailers. **Visa**, 2019. Disponivel em: https://usa.visa.com/run-your-business/small-business-tools/retail.html. Acesso em: 04 Novembro 2019.

WANG, Dexi; JIANG, Yu; SONG, Houbing; HE, Fei; GU, Ming; SUN, Jiaguang. Verification of Implementations of Cryptographic Hash Functions. **IEEE Access**, 5, 2017. 7816-7825.

WANG, F.-Y.; YUAN, Y. **Towards blockchain-based intelligent transportation systems**. IEEE 19th International Conference on Intelligent Transportation Systems. Rio de Janeiro: (ITSC). 2016. p. 2663-2668.

WANG, L.; PUSTOGAROV, I. Towards Better Understanding of Bitcoin Unreachable Peers, 2017. Disponível em: https://arxiv.org/pdf/1709.06837.pdf. Acesso em: Março 30 2019.

WIKIPÉDIA. **Testnet**, 2019. Disponível em: https://pt.wikipedia.org/wiki/Testnet#Faucets. Acesso em: 06 Outubro 2019.

ZHENG, Zibin; XIE, Shaoan; DAI, Hongning; CHEN, Xiangping; WANG, Huaimin. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. **International Congress on Big Data**, 2017. 557-564.

ZYSKIND, G.; NATHAN, O.; PENTLAND, A. '. Decentralizing Privacy: Using Blockchain to Protect Personal Data. **IEEE Security and Privacy Workshops**, San José, 2015. 180-184.

APÊNDICE(S)

APÊNDICE A - ARTIGO

Blockhain na Educação: Uso da Tecnologia como Prova de Existência de Diplomas e Certificados

Dérick Souza[1], Paulo João Martins[2]

[1]Acadêmico do Curso de Ciência da Computação - Unidade Acadêmica de Ciências, Engenharias e Tecnologias - Universidade do Extremo Sul Catarinense (UNESC) - Criciúma – SC

[2] Professor do Curso de Ciência da Computação - Unidade Acadêmica de Ciências, Engenharias e Tecnologias - Universidade do Extremo Sul Catarinense (UNESC) - Criciúma - SC

derick_sm@hotmail.com, pjm@unesc.net

Abstract. *The current environment of registration of diplomas and certificates, used in the Brazilian educational environment, is attributed to the physical documents and recently its use with digital systems, assured with digital signatures, was approved. However, the present scenario uses methods and services that are susceptible to fraud, forge and still have security gaps and storage systems that cannot guarantee the administration of the document interest cycle, given the long-term applications of current and future users. Given the need for a more secure service, in this project is made a study and development of a prototype that uses of a proof of existence service through the Bitcoin blockchain, allowing to create immutable records and guarantee total data longevity without the need to maintain centralized record verification processes.*

Resumo. *O atual ambiente de registro de diplomas e certificados, utilizado no ambiente educacional brasileiro, é atribuído ao meio físico e recentemente sua utilização no meio digital, com uso de assinaturas digitais, foi aprovada. No entanto, o presente cenário utiliza métodos e serviços que são suscetíveis a fraudes, forjas e ainda possuem lacunas de segurança no armazenamento de forma que não pode garantir a administração do ciclo de interesse dos documentos, atendendo a longo prazo as necessidades dos atuais e futuros usuários. Tendo em vista a necessidade por um serviço que com capacidade prover maior segurança, no presente é realizado um estudo e desenvolvimento de um protótipo, utilizando um serviço de prova de existência por meio da blockchain do protocolo Bitcoin, permitindo criar registro de formas imutável e garantindo total longevidade para os dados, sem a necessidade manter processos centralizados para verificação dos registros.*

1. Introdução

Apostos em papel, os documentos não mais correspondem às necessidades de rapidez na circulação das informações, ainda existem suas limitações, referentes à conservação, transmissibilidade ou segurança (GANDINI et al., 2016).

A validação de documentos é essencial em qualquer contexto legal, normalmente, documentos físicos são validados por meio de autoridades centrais, onde são aplicados registros

mecânicos, os concedendo maior segurança (LEAO; CANEDO; GOMES, 2017). Os documentos tradicionais, bem como suas medidas de segurança, se desintegram ou podem se tornar irrecuperáveis e ainda existem efeitos da temperatura, umidade, poluição do ar, danos provocados pelo uso indevido e regular e até mesmo as catástrofes naturais (ARELLANO, 2004).

Com base nos fundamentos apresentados, é possível perceber que o uso de tecnologias, para armazenamento de documentos em formato digital, convém a ser de grande utilidade, onde as atuais ferramentas, além da preservação, permitem fácil catalogação, busca e organização.

Quando se trata de documentos digitais, para que eles tenham validade jurídica, é necessário que atendam a requisitos que garantam sua confiabilidade, normalmente, a técnica de assinaturas digitais é uma solução (GANDINI; SALOMÃO; JACOB, 2001; MARCACINI, 1999).

Uma característica em comum, tanto para documentos tradicionais e eletrônicos, é a necessidade de um intermediador para sua autenticação. Quando apostos em papel, uma autoridade central é usada para autenticá-lo, por exemplo, um cartório, quanto aos formatos digitais, os envolvidos utilizam seu par de chaves para assiná-lo, também geradas por uma autoridade central. Em ambos os casos há necessidade de uma entidade intermediadora, onde é possível identificar a falta de privacidade.

Na educação, certificados e diplomas, uma forma de "documento", possuem grande importância, normalmente impressos em papel, eles atribuem capacitações a seus portadores (GRÄTHER et al., 2018, tradução nossa).

Normalmente estes documentos são atribuídos ao meio físico, no entanto, acabam gerando alto custo na sua emissão, devido ao uso papéis e registros especiais para garantir sua segurança, ao mesmo tempo, existe uma grande dificuldade em sua entrega, o que acaba gerando desperdício de material. Ainda, apresentam longos prazos de produção e entrega, e ineficiência no processo de verificação e geração de segundas vias (LEAO; CANEDO; GOMES, 2017).

Eles ainda estão dispostos ao risco de sua eventual perca, e até mesmo encerramento das atividades da instituição que os gerou, o que incapacita a sua verificação (SALLES, 2019). Outro problema frequente, é em relação as indústrias de fraudes, onde cada vez mais, as quadrilhas de venda de diplomas falsos se tornam mais populares.

Com base no cenário abordado e conhecendo a importância pessoal e profissional gerada a partir dos diplomas e certificados, é possível perceber a necessidade por métodos e serviços com capacidade de gerar maior segurança e garantir total integridade para seus registros.

Uma tecnologia, capaz de suprir estas necessidades é a *blockchain*, a qual permite realizar registros imutáveis, trazendo total segurança aos dados, e eliminado a possibilidade de fraude e forjas, e ao mesmo tempo, criando um ambiente em que não há intervenção de terceiros ou autoridades responsáveis por criar e verificar os registros (COSTA et al., 2018; GRÄTHER et al., 2018, tradução nossa).

Com um serviço de prova de existência, do inglês *Proof-of-Existence* (PoF), por meio de uma *blockchain*, é possível prover as informações necessárias para emissão e validação, e consequentemente realizar a padronização de informação, segurança de armazenamento e

estabelecimento de regras para o compartilhamento dos dados. Isto é uma forma de contornar os problemas impostos pelos atuais métodos de emissão de diplomas, como também para qualquer contexto que necessita emitir "documentos como prova".

No presente trabalho, propõe-se a utilização de um serviço de prova de existência com o uso da *blockchain* do protocolo Bitcoin, como proposta para criar um ambiente adequado e seguro para emissão e compartilhamento de diplomas e certificados, dessa forma, permitindo seu armazenamento de forma imutável e garantindo longevidade e integridade dos dados, dadas as características obtidas com o uso da *blockchain*.

2. Diplomas e certificados

Um diploma, é caracterizado como prova da graduação e um certificado pode ser **i**dentificado como uma prova de capacitação. No Brasil, a curadoria e gerenciamento desse tipo de registro é mantida pelo Ministério da Educação (MEC), o órgão federal que trata da política nacional de educação em geral.

De acordo com Lei de Diretrizes e Bases da Educação, a lei nº 9394/1996 afirma que a emissão dos diplomas é responsabilidade das instituições, incluindo comprovantes e certificados de conclusão de curso. Quanto a sua preservação, as IES devem manter os registros de todos documentos relacionados e emitidos por ela, e sua a gestão está regulamentada pela Lei 8159/91.

A Portaria nº 554 estabelecida pelo MEC, regulamenta o registro e a emissão de diplomas no ambiente digital, em todas IES que compõem o Sistema Federal de Ensino De acordo com o MEC, o objetivo é possibilitar o melhor aproveitamento de recursos, preservando as mesmas condições e garantias dos diplomas físicos e evitar as atuais burocracias relacionados a sua emissão. O padrão estabelecido para utilização é o *Extensible Markup Language*, com uso de assinatura digital para garantir sua segurança (BRASIL, 2019).

2.1 Diplomas Digitais

O uso de diplomas em formato digital, possui grandes vantagens, quando comparados ao seu uso em formatos físicos. Com eles é possível prover, uma manipulação e compartilhamento mais adequados, permitindo reduzir seu custo de impressão e sua replicação e distribuição imediata, ilimitada e de forma gratuita, ainda, provê a possibilidade de verificação por mecanismos automáticos, eliminando a necessidade de intermediadores responsáveis por tal procedimento. Além disso, os mecanismos e procedimentos aplicados a ele dificultam as tentativas de fraudes e forjas (GRÄTHER et al., 2018, tradução nossa; GRECH; CAMILLERI, 2017 tradução nossa).

No entanto, o uso de assinaturas digitais como forma de garantia da autenticidade dos documentos apresenta algumas lacunas de segurança que podem implicar diretamente na integridade e armazenamento dos dados, ocasionando um sistema com falhas, que podem comprometer o ciclo de interesse de seus usuários.

O processo de assinatura digital acaba dependendo por uma autoridade responsável por tal procedimento, no entanto, elas possuem controle dos aspectos de segurança e verificação, os quais podem a vir ser explorados por ela, ao mesmo tempo, por ser mantido de forma centralizada, o possível encerramento das atividades da entidade, ou o comprometimento com seu armazenamento de dados, ocasiona na incapacidade da verificação de todos documentos,

uma vez que não é mais possível identificar os envolvidos no procedimento (GRECH; CAMILLERI, 2017 tradução nossa).

Ainda não se pode garantir total confiabilidade sobre a entidade autorizada, responsável por manter o processo da assinatura digital seguro, ainda, é comum esse tipo de entidade estar relacionado ao vazamento de dados (GRECH; CAMILLERI, 2017 tradução nossa).

Tendo em vista as lacunas de segurança e a novamente a dependência por terceiros mesmo com o uso de diplomas validados com o uso de assinatura digital, a *blockchain* permite criar um sistema capaz de suprir as falhas relacionadas ao atual ambiente de emissão de diplomas no meio físico e digital.

Dadas as características de imutabilidade e descentralização desse tipo de tecnologia, é possível explorar e criar um sistema que criar registros com incapacidade de forja e fraudes, ao mesmo tempo em que é eliminada a necessidade de dependência por autoridades para realizar o registro e verificação de tais documentos. A *blockchain* permite o registro e verificação de forma autônoma, dando capacidade de provar a autenticidade do registro e permitindo uma forma de identificar quem o realizou.

No presente momento, a preservação digital é um dos fatores mais importantes para garantir a longevidade dos registros, e então permitir sua administração durante todo ciclo de interesse dos envolvidos. Para realizar tal prática, o uso de armazenamento de dados de forma distribuída é necessário, criando um ambiente com total integridade e segurança de dados, no entanto, ao supor a implementação de tal sistema por uma universidade, existe a necessidade de grandes investimentos (RUUSALEPP; DOBREVA, 2012, tradução nossa; RAMALHO et al., 2007, tradução nossa), com grande espera em seu retorno de investimento, contudo, ao utilizar a *blockchain*, é possível obter tais características de forma simples e gratuita .

3. Bitcoin e Blockchain

O Bitcoin surgiu em 2008, com a publicação de um artigo na internet que rapidamente se popularizou por possuir a proposta de um sistema de pagamentos eletrônicos mais eficiente comparado aos utilizados atualmente. Sua principal proposta era permitir a transferência e pagamentos entre usuários sem a necessidade de um intermediador de confiança, como nos meios de pagamentos mais comuns (NAKAMOTO, 2008, tradução nossa). Esse tipo de prática leva a falta de privacidade, lentidão, e submissão a uma autoridade que não se pode garantir total confiança uma vez que não é possível determinar como os dados são validados por ela. Como proposta para evitar este cenário, o Bitcoin propôs uma rede de pagamentos descentralizada onde usuários se comunicam diretamente entre si, onde a garantia e integridade dos dados e da própria rede, são mantidas por diversas técnicas de criptografia.

Para suprir a necessidade por tal rede de pagamentos, foi desenvolvida a tecnologia *blockchain*, a qual consiste em uma rede *peer-to-peer*, que forma um registro distribuído de todas transações, as quais são validadas em troca de um grande poder computacional, ao utilizar o conceito de prova de trabalho (NAKAMOTO, 2008, tradução nossa).

A arquitetura utilizada nesta rede é o modelo público, portanto não existem restrições de participações, onde as regras e consenso da rede, e o armazenamento de dados, são feitos pelos próprios usuários. Neste modelo, existe a liberdade de realizar transações, sem a necessidade de intermediadores de confiança, uma vez que a integridade dos dados é obtida,

pelas técnicas de criptografia aplicadas em toda plataforma (ANTONOPOULUS, 2014, tradução nossa).

Nesta rede as transaçõe*s,* são criadas com o uso de um *script* que segue as regras do protocolo do Bitcoin. Esse tipo de estrutura representa um conjunto de dados, que é definido pelos desenvolvedores da rede e eles seguem uma linha de execução que garante a validação e integridade das transações, utilizando técnicas de criptografia e são validados pelo algoritmo de consenso utilizado na rede (ANTONOPOULOS, 2014, tradução nossa).

A validação das transações é feita pelo uso do algoritmo de prova de trabalho, do inglês *proof-of-work*, este algoritmo utilizado uma grande força computacional para calcular um desafio matemático estabelecido pelo protocolo da rede (TURKANOVIć et al., 2018, tradução nossa). As características deste algoritmo tornam extremamente difícil a adulteração de dados, tendo em vista que para modificar o conteúdo de algum dado da *blockchain*, a mesma força computacional empregada para gerá-lo deve ser utilizada, portanto, se torna um processo matematicamente complexo e computacionalmente impraticável e custoso.

A *blockchain* é uma cadeia de blocos que possuem uma ligação uns aos outros, obtida a partir do conteúdo presente em cada um deles, essa ligação é obtida pelo uso de técnicas de criptografia e qualquer alteração realizada em seu conteúdo, acaba quebrando a referência mantidas entre os blocos, desta forma, o bloco alterado não faz mais parte da rede, portanto é dito como inválido (CROSBY et al., 2016 , tradução nossa).

3.1 Blockchain e registro de diplomas

A infraestrutura fornecida pela tecnologia da *blockchain*, é ideal para armazenar, compartilhar e verificar qualquer tipo de registro. Com sua estrutura, é possível eliminar a necessidade de uma entidade intermediadora para o controle e administração da rede, portanto, não existe a necessidade de manter processos de registro e verificação dos dados de forma centralizada (GRECH; CAMILLERI, 2017, tradução nossa). Essa rede ainda é reforçada pela descentralização, e dado a forma de armazenamento de ados, é obtido maior robustez, segurança e integridade para os dados.

A *blockchain* foi desenvolvida como uma rede monetária, desta forma, a inserção de dados arbitrários nela, deve ser feita pela manipulação do operador OP_RETURN. Este operador foi criado pelos desenvolvedores da rede, com uma solução para este tipo de prática, sabendo que antes, eram manipulados scripts de pagamento, ocasionando sobrecargas na rede.

Para registrar um diploma ou certificado na *blockchain*, o primeiro passo a ser tomado é a criação de um identificador único a partir do documento original. Esse identificador é gerado pelo uso de uma função *hash* criptográfica, a qual é capaz de gerar uma saída, ou resumo, de tamanho fixo a partir de qualquer entrada de tamanho arbitrário (GAEINI; GHAFFARI; MOSTAGHIM, 2018, tradução nossa), e dadas suas características, qualquer alteração no documento original resultaria em resumo totalmente diferente do original, desta forma, é possível criar um meio de verificar a integridade dos documentos e ao mesmo tempo realizar sua busca pela rede.

Uma vez que gerado o resumo do arquivo, é necessário enviá-lo a rede manipulando um script com o uso do operador OP_RETURN, nele serão informados os dados referentes a taxa de validação da transação, os dados da carteira da universidade e o *hash* do arquivo.

Para realizar o processo de verificação, deve ser gerado o resumo do arquivo original e realizar uma varredura pela rede, caso encontrado o registro, deve ser acessada a transação em que ele foi registrado e verificar a chave pública ou endereço nela utilizada e compará-lo com a utilizada pela universidade, caso sejam iguais, o documento é autêntico.

4. Metodologia

Para a elaboração deste projeto, o primeiro passo tomado foi o levantamento do material bibliográfico, necessário para o entendimento das tecnologias envolvidas, tendo como maior foco a *blockchain*.

O material encontrado e utilizado foi por meio de livros, teses, publicações da Internet e em repositórios de trabalhos acadêmicos, contudo, teve como principal fonte de pesquisas as bases de dados: Google Acadêmico, *IEEE Xplore* e *Research Gate*, por meio de artigos científicos.

Lida a documentação encontrada, passou-se a escrever o referencial teórico, apresentando os principais pontos e assuntos correlacionados ao objetivo deste trabalho.

Foram descritas as características de cada tipo de *blockchain* e suas respectivas qualidades e atributos, tendo em vista sua importância na tomada de decisão na escolha de sua adoção. Esse levantamento, também implica na forma de manipulação dos dados e requisitos mínimos para caracterização de um projeto de prova de existência.

As próximas seções descrevem as características e o etapas adotadas para o desenvolvimento do projeto prático. Tendo como partida o levantamento de requisitos e recursos necessários para se adequar a melhor forma do desenvolvimento envolvendo a *blockchain*, assim como, buscar formas de se adequar as características necessárias para criar um serviço de prova de existência.

4.1. Definições

Apesar da existência de diversas redes com a arquitetura da *blockchain*, e até mesmo a possibilidade da criação de uma rede direcionada exclusivamente ao projeto, a decisão do uso da *blockchain* da criptomoeda Bitcoin foi feita.

Tendo em vista os objetivos deste estudo, o uso de uma rede pública, amplamente conhecida, com milhares de nós sendo executados, são características adequadas, além disso, elas possuem a estrutura que garante maior imutabilidade e transparência, por meio da descentralização, portanto, se adequando a prova de existência (DURANT; TRACHY, 2017, tradução nossa; GRECH; CAMILLERI, 2017, tradução nossa).

O registro de dados na *blockchain* foi definido pelo uso do operador OP_RETURN, tendo em vista que atualmente a forma mais apropriada para inserir dados arbitrários a rede, além disso ele possui um custo muito menor quando comparado aos scripts de pagamento, e no presente momento, ele é a única forma de não criar transações inválidas que trazem sobrecarga a rede.

Para manipulação dos dados relacionados ao protocolo do Bitcoin, foi utilizada a biblioteca de abstração *bitcoinjs-lib*. Essa biblioteca possui suporte para criação de todos scripts da rede e ainda permite a geração e importação de carteiras, no entanto, ela não realiza a comunicação com a *blockchain*, ela somente cria os dados para serem enviados a ela.

Desta forma uso de uma API para a manipulação dos dados, apresentou grande importância e foi indispensável para o desenvolvimento da aplicação. Como resultado a API REST Smartbit foi utilizada para o envio dos dados a rede. Seu uso foi feito devido a ela fornece suporte as transações OP_RETURN, permitindo a visualização do seu conteúdo, ao contrário da maioria das outras plataformas.

4.2. Fluxo de dados da aplicação

Uma vez realizada a escolha da rede e da forma de inserção de dados na *blockchain*, foi estipulado um diagrama de fluxo de dados, com o intuito do mapeamento dos processos e informações necessárias. Com ele foi possível obter uma visão global do projeto, facilitando a percepção de todas etapas a serem implementadas. A figura 1 ilustra o fluxo de dados da aplicação.

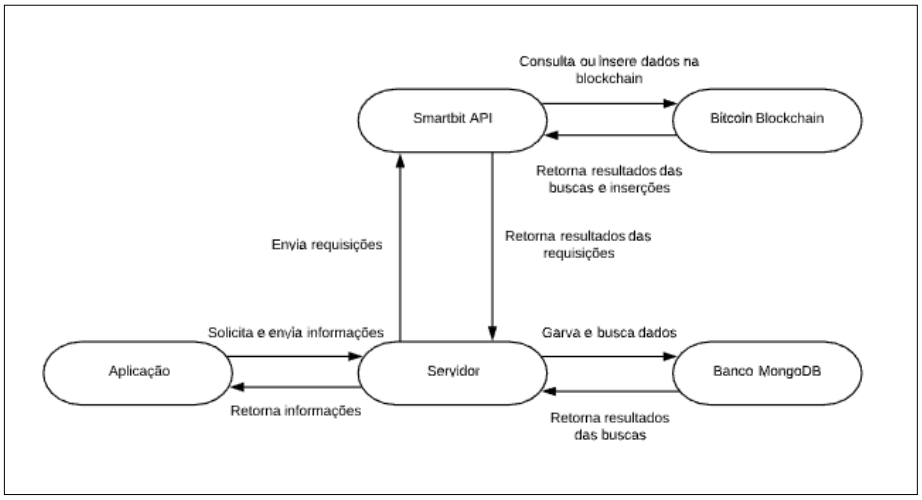


Figura 39. Fluxo de dados

O início do fluxo de dados, se dá com as requisições feitas a partir da aplicação para o servidor, o qual pode estar realizando as buscas ou envios para o banco de dados, ou então para a API externa, que é responsável pela comunicação direta com a *blockchain*.

4.3 Desenvolvimento da aplicação

Para o desenvolvimento da aplicação foi utilizado o banco de dados MongoDB, nele foram inseridas as coleções necessárias para o armazenamento de dados referente aos registros, permitindo assim, futuras verificações.

Um servidor Node.JS com uso do framework Express também foi configurado para permitir a elaboração dos dados a serem enviados para *blockchain* e ainda manter a comunicação entre a ampliação e o banco de dados.

Com o ambiente configurado, foi iniciado o desenvolvimento da aplicação para gerenciamento de dados, enviados pelo usuário, e comunicação com a *blockchain*.

Todo código relacionado ao *back-end* do protótipo, foi desenvolvido com a linguagem de programação *JavaScript* e para o lado do *front-end,* foram utilizadas as tecnologias HTML, CSS e a biblioteca de estilos *Bootstrap* na versão 4.0.

A implementação do código foi dívida em etapas, sendo elas: geração e importação de carteiras, gerar *hash* de arquivos, criar transações, envio de transações a rede e verificação de arquivos.

Para gerar a carteira do usuário a biblioteca *bitcoinjs-lib* foi utilizada, ela permite gerar a chave privada, a chave pública e o endereço de acordo com as regras do protocolo e ainda permite a importação dos dados de uma carteira já criado.

Tendo em vista a necessidade de criar *hash* de arquivos, a aplicação permite seu upload, e por meio das funções nativas de criptografia dos navegadores, o arquivo é executado na função *hash* criptográfica SHA-256.

Uma vez gerada a carteira e adicionado o arquivo para registro na *blockchain*, é necessário montar uma transação para realizar o envio a rede, deste modo, a aplicação utiliza o *script null data,* por meio do operador OP_RETURN. Para montá-lo, deve-se fornecer um endereço e sua chave privada e definir a taxa que será paga ao minerador, responsável por inserir a transação em um bloco.

Os dados relacionados a transações são novamente manipulados pela *bitcoinjs-lib*, no entanto, são enviados a rede pelo uso da *Smartbit* API, por meio de requisições HTTP.

Para verificação, um usuário deve realizar o upload do arquivo original, onde a aplicação, executa o algoritmo para criar seu resumo e então exercer uma varredura pela *blockchain*, e ao obter o resultado é feita uma comparação para garantir a integridade dos dados.

5. Resultados e Discussão

Com a conclusão de todas etapas determinadas para o projeto e com os respectivos testes realizados, foi possível realizar uma análise relacionada a seus resultados e dessa forma, verificar sua viabilidade.

Com o protótipo finalizado, os conceitos e tecnologias que envolvem o ambiente de desenvolvimento da *blockchain* foram aplicados, ao utilizar as técnicas e métodos necessários para a área, permitindo o manuseio de uma aplicação que se adequa ao ambiente do protocolo Bitcoin.

Os registros efetivados se adequam ao conceito da prova de existência, garantindo ao emissor e portador do diploma ou certificado, uma prova irrefutável de sua veracidade, além do mais, eliminam a necessidade de terceiros, para manter o processo de verificação desses documentos.

O processo de levantamento de dados realizado, permitiu compor uma estrutura do protótipo, a qual se adequa ao tipo de *blockchain* utilizada, tendo como foco as características de descentralização, longevidade e transparência, as quais foram alcançadas com o uso do protocolo Bitcoin e sua rede pública.

Com o uso desta rede, também foi possível alcançar alguns dos conceitos vistos como essenciais para a preservação digital, sendo eles, referente ao armazenamento em bancos de dados distribuídos e descentralizados. Ao armazenar registros de forma centralizada, sua seguridade fica comprometida com as possíveis falhas em seu *hardware* de armazenamento, contudo, com o registro na rede do Bitcoin, qualquer falha em um de seus milhares de nós completos, não compromete qualquer problema relacionado a integridade de tais dados.

Ao mesmo tempo que a rede do Bitcoin pode oferecer grandes vantagens em relação a segurança e imutabilidade dos dados, ela pode trazer altos custos para realizar o registro dos diplomas e certificados. Por ser uma rede monetária, o Bitcoin, tende a sofrer constantes altas e baixas de acordo a busca do mercado, por este motivo, a viabilidade financeira do projeto poder ser comprometida, tendo em vista que não é possível estipular o custo para mantê-la e ao mesmo tempo, seu custo de operação em momentos de alta da moeda, se torna inviável para a instituição que utiliza este tipo de sistema.

Para evitar esse tipo de problema, é possível realizar o uso de outras *blockchain*s, com moedas mais estáveis, no entanto as características que elas podem oferecer são diferentes de acordo com seu número de usuários, algoritmo de consenso e tipo de rede.

Ao supor esse tipo de aplicação em larga escala, atendendo todas as necessidades do ambiente educacional e sendo utilizada por várias instituições de ensino, o cenário mais adequado, seria pela criação e uso de uma *blockchain* destinada ao registro de dados arbitrários. Nessa rede, o modelo consortium poderia ser estabelecido, onde é possível determinar os usuários com permissão de escrita, destinada somente as instituições, e a permissão leitura, a qual seria pública. Com todos registros educacionais sendo aplicados a rede, a imutabilidade cresceria rapidamente e ainda, os custos de operação poderiam ser reduzidos, ao escolher um algoritmo de consenso e processamento de dados eficiente, o qual poderia se adaptar as necessidades do ambiente. Uma outra característica ideal para esse modelo, é que a cada novo ingresso de uma instituição à rede, um novo nó completo deve ser implementado por ela, dessa forma, resulta em um sistema de descentralização

6. Conclusão

A partir do levantamento bibliográfico e da implementação do projeto, foi possível reconhecer as principais propostas de uma *blockchain*, a segurança, imutabilidade e descentralização.

Com a finalização do trabalho em questão, foi possível observar uma melhor compreensão, relacionada as tecnologias utilizadas para o desenvolvimento do protótipo, assim como, obter noção da composição e da importância dos dados mantidos em um ambiente educacional.

As técnicas de criptografias utilizadas na *blockchain* garantem segurança, transparência, longevidade e imutabilidade, fazendo uso de um mecanismo que possui a tendência de cada vez mais, eliminar a possibilidade de adulteração de dados. Dadas suas características, a tecnologia se tornou um objeto de estudo para diversas áreas, não relacionadas a cenários monetários, as quais foram abordadas neste estudo, com o objetivo de reduzir a interferência humana nos processos de verificação e compartilhamento dos diplomas, assim como, criar um cenário, diferentemente do atual, onde nos atuais métodos, existe a necessidade de confiança em terceiros para poder garantir a segurança destes documentos.

Neste trabalho, o protótipo criado, permite os registros de arquivos da *blockchain* do protocolo Bitcoin, criando uma prova irrefutável de sua existência em determinada data. Este registro consiste na inserção de um identificador único do arquivo em uma transação, o qual é calculado pela função *hash* criptográfica SHA-256, dessa forma, é possível verificar sua existência ao enviar para aplicação um determinado arquivo, e então por meio de seu resumo,

uma busca é feita na rede. Caso encontrado, o usuário tem acesso às informações, e ao comparar se o identificador do arquivo existe na transação exibida, é possível afirmar sua existência.

Uma vez que registrado o arquivo, não existe a necessidade de manter processos de verificação centralizados, devido a confiabilidade na rede e na aplicação permitirem que qualquer usuário, com o arquivo original, possa provar sua existência de forma independente. Caso o usuário escolha verificar um arquivo sem o uso da aplicação, ele pode calcular o seu resumo com a função SHA-256 e usar uma plataforma *block explorer,* para realizar sua busca pela *blockchain*.

Os dados relacionados ao registro, armazenados no cabeçalho do bloco em que a transação foi inserida, possuem garantia de imutabilidade, tendo em vista as propriedades criptográficas alcançadas com a *blockchain*. Quanto a integridade de um arquivo, ela é obtida de modo que a menor alteração possível no arquivo original, resultaria em identificador único, sem qualquer relação ao registro feito.

As características obtidas pelos registros, foram alcançadas com o uso da *blockchain* pública do Bitcoin, a qual teve escolha realizada, de acordo com o levantamento bibliográfico, tendo como objetivo encontrar uma rede com características ideais para manter um registro imutável, sem a necessidade de centralização dos dados.

Por fim, os detalhes e características obtidos com o desenvolvimento da aplicação foram apresentados e com relação ao ponto de vista técnico, como um todo e com base nos testes realizados, obteve-se um resultado satisfatório quanto a integração entre todas as tecnologias implementadas.

7. Referências

ANTONOPOULOS, A. M. **Mastering bitcoin:** unlocking digital cryptocurrencies. 1ª. ed. [S.l.]: O'Reilly Media, 2014.

ARELLANO, M. Á. M. Preservação de Documentos Digitais, 2004. Disponível em: http://revista.ibict.br/ciinf/article/view/1043. Acesso em: 2019 Maio 12.

BRASIL. Lei n. 8.159, de 8 de jan. de 1991. **Política nacional de arquivos públicos e privados e dá outras providências**, 8 jan. 1991. Disponível em: http://www.planalto.gov.br/ccivil_03/leis/l8159.htm. Acesso em: Abril 27 2019.

BRASIL. Lei, n. 9.394, de 20 de dez. de 1996. **Diretrizes e bases da educação nacional**, 20 dez. 1996. Disponível em: https://www2.senado.leg.br/bdsf/bitstream/handle/id/70320/65.pdf. Acesso em: 27 Abril 2019.

BRASIL, Ministério da Educação. Portaria nº 1.095, 25 de outubro de 2018. **Expedição e o registro de diplomas de cursos superiores de graduação no âmbito do sistema federal de ensino**, 2018. Disponível em: a expedição e o registro de diplomas de cursos superiores de graduação no âmbito do sistema federal de ensino. Acesso em: 27 Abril 2019.

BRASIL, Ministério da Educação. Portaria nº 554, de 11 de março de 2019. **Dispõe sobre a emissão e o registro de diploma de graduação, por meio digital, pelas Instituições de**

Ensino Superior - IES pertencentes ao Sistema Federal de Ensino, 2019. Disponível em: http://www.in.gov.br/materia/-/asset_publisher/Kujrw0TZC2Mb/content/id/66544171/do1-2019-03-12-portaria-n-554-de-11-de-marco-de-2019-66543842. Acesso em: 27 Abril 2019.

COSTA, Rostand; FAUSTINO, Daniel; LEMOS, Guido; QUEIROGA, Ademir; DJOHNNATHA, Cláudio; ALVES, Felipe; LIRA, Jordan; PIRES, Mateus. Uso Não Financeiro de Blockchain: Um Estudo de Caso Sobre o Registro, Autenticação e Preservação de Documentos Digitais Acadêmicos. **Workshop em Blockchain: Teoria, Tecnologias e Aplicações (WBlockchain_SBRC)**, v. 1, n. 1/2018, Maio 2018. Disponível em: http://ojs.sbc.org.br/index.php/wblockchain/article/view/2356.

CROSBY, Michael; NACHIAPPAN; PATTANAYAK, Pradhan; VERMA, Sanjeev; KALYANARAMAN, Vignesh. BlockChain Technology. **Applied Innovation Review**, v. 2, 2016. Disponível em: https://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf.

DURANT, E.; TRACHY, A. MIT News. **Digital Diploma debuts at MIT**, 2017. Disponível em: http://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017. Acesso em: 05 Outubro 2019.

GAEINI, A.; GHAFFARI, M. H.; MOSTAGHIM, Z. An Improved Hash Function Based on the Tillich-Zémor Hash Function. **Mathematics Interdisciplinary Research 3**, 2018.

GANDINI, J. A. D.; SALOMÃO, D. P. D. S.; JACOB, C. A SEGURANÇA DOS DOCUMENTOS DIGITAIS, 2001.

GRÄTHER, Wolfgang; KOLVENBACH, Sabine; RULAND, Rudolf; SCHÜTTE, Julian; TORRES, Christof Ferreira; WENDLAND, Florian. Blockchain for Education: Lifelong Learning Passport. In: (EUSSET), E. S. F. S. E. T. **Proceedings of 1st ERCIM Blockchain Workshop 2018, Reports of the European Society for Socially Embedded Technologies**. Amsterdam: [s.n.], v. 2, 2018.

GRECH, A.; CAMILLERI, A. F. **Blockchain in Education**. Luxemburgo. 2017.

LEAO, H. A. T.; CANEDO, E. D.; GOMES, A. R. **Proposta de Uso do Blockchain para Validaçao de Documentos de Instituiçoes de Ensino Superior**. Brasília. 2017.

MARCACINI, A. T. R. O DOCUMENTO ELETRÔNICO COMO MEIO DE PROVA, 1999. Disponível em: http://www.egov.ufsc.br/portal/sites/default/files/anexos/13948-13949-1-PB.htm.

NAKAMOTO, S. Bitcoin: A Peer-to-Peer Electronic Cash System, 2008. Disponível em: https://bitcoin.org/bitcoin.pdf.

RAMALHO, José Carlos; FARIA, Luís; FERREIRA, Miguel; CASTRO, Rui. **Relational Database Preservation through XML modelling**. Extreme Markup Languages 2007. Montréal: [s.n.]. 2007.

RUUSALEPP, R.; DOBREVA, M. Digital Preservation Services: State of the Art Analysis, 2012. Disponível em: https://www.um.edu.mt/library/oar/bitstream/handle/123456789/311/dobreva_Preservation_Services_study.pdf?sequence=1&isAllowed=y. Acesso em: 23 Março 2019.

SALLES, S. Cinco anos após fim da Gama Filho, abandono do campus simboliza decadência do bairro de Piedade. **Blog do Acervo - O Globo**, 2019. Disponível em: https://blogs.oglobo.globo.com/blog-do-acervo/post/cinco-anos-apos-fim-da-gama-filho-abandono-do-campus-simboliza-decadencia-do-bairro-de-piedade.html. Acesso em: 2019 Abril 27.

SMARTBIT, 2019. Disponível em: https://www.smartbit.com.au/. Acesso em: 09 Agosto 2019.