UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

Alexandre Pereira Back

UMA ARQUITETURA PARA MICRO TRANSAÇÕES FINANCEIRAS EM UM CONTEXTO DE ECONOMIA DAS COISAS

Florianópolis

2019

Alexandre Pereira Back

UMA ARQUITETURA PARA MICRO TRANSAÇÕES FINANCEIRAS EM UM CONTEXTO DE ECONOMIA DAS COISAS

Trabalho de conclusão de curso apresentado como parte dos requisitos para a obtenção do Grau de Bacharel em Sistemas de Informação.
Orientador: Prof. Dr. Alex Sandro Roschildt Pinto
Coorientador: Me. Bruno Machado Agostinho

Florianópolis

2019

Alexandre Pereira Back

UMA ARQUITETURA PARA MICRO TRANSAÇÕES FINANCEIRAS EM UM CONTEXTO DE ECONOMIA DAS COISAS

Trabalho de conclusão de curso apresentado como parte dos requisitos para a obtenção do grau de "Bacharel em Sistemas de Informação".

Florianópolis, de 2019.

Banca Examinadora:

Prof. Dr. Alex Sandro Roschildt Pinto
Orientador

Me. Bruno Machado Agostinho
Coorientador

Profa. Dra. Carla Merkle Westphall

Prof. Dr. Frank Augusto Siqueira

Este trabalho é dedicado a meus pais, meus orientadores, meus amigos e minha namorada, que me apoiou incondicionalmente nesta reta final da conclusão do curso

AGRADECIMENTOS

Ao meu coorientador e amigo Bruno Machado Agostinho, que no momento que mais precisei de ajuda na vida acadêmica, me estendeu a mão e me orientou no caminho correto.

Ao Professor Alex Sandro Roschildt Pinto, me orientou e deu conselhos essenciais na conclusão deste trabalho.

Aos Professores Carla Merkle Westphall e Frank Augusto Siqueira por fazerem parte da minha banca avaliadora.

A todos os amigos feitos na vida acadêmica, sem cada um deles eu não chegaria aonde cheguei, seja em 4 ou 12 anos.

Aos meus pais Alvete e Manoel, por sempre acreditarem na minha capacidade de concluir esta etapa de minha vida, por mais longa que ela tenha sido.

E especialmente, à minha namorada Marjorie Miranda, por ter tornado esta conclusão possível. Sem seu incessante apoio, eu jamais teria a resiliência de enfrentar as dificuldades de regressar a UFSC e concluir o curso.

Quando se navega sem destino, nenhum vento é favorável

Sêneca

RESUMO

A Internet das coisas (IoT) revolucionou a forma como vivemos e trabalhamos. Sua capacidade de captar, compartilhar e utilizar informações para tomadas de decisão inteligentes, qualificam um grande potencial de mudança atrelado a esta tecnologia. Estes dispositivos estão cada vez mais presentes no dia-a-dia das pessoas, desde pulseiras inteligentes informando o batimento cardíaco, até casas e carros inteligentes. Da aliança desta tecnologia com Blockchain, desenvolvida para uso com a criptomoeda BitCoin e cada vez mais explorada no mundo acadêmico, nasce a Economia das Coisas (EoT). A EoT é a disponibilização de serviços digitais fornecidos por dispositivos IoT em marketplaces baseados em Blockchains. Há ainda necessidade de uma proposta de controle para as transações neste universo e é para este fim que é proposta neste trabalho, uma arquitetura de microsserviços. A versatilidade alcançada por sistemas baseados na arquitetura de microsserviços, viabiliza fácil integração de diferentes tecnologias focadas na resolução de um problema em comum. Os microsserviços podem internamente possuir das mais diversas abordagens de desenvolvimento, sem que afete a funcionalidade do sistema como um todo, podendo ser moldado na melhor forma proposta para realização de suas próprias tarefas. Assim, será apresentada uma proposta para micro transações financeiras num contexto de EoT, através de uma arquitetura de microsserviços, com apresentação de resultados obtidos em ambiente experimental desenvolvido para validar a proposta.

Palavras-chave: IoT. Economia das Coisas. Microsserviços. Blockchain

ABSTRACT

IoT has revolutionized the way we live. Its ability to capture, share and use information for smart decision-making qualifies for a great potential for change linked to this technology. The alliance of this technology with Blockchain, developed for use with BitCoin cryptocurrency, gave birth to the Economy of Things (EoT). EoT is the delivery of digital services provided by IoT devices on Blockchains-based marketplaces. In this paper we will present a proposal for the control of micro financial transactions in an EoT context, through a microservices architecture, with presentation of results obtained in an experimental environment developed to validate the proposal.

Keywords: IoT. EoT. Microservices. Blockchain

LISTA DE FIGURAS

LISTA DE ABREVIATURAS E SIGLAS

SUMÁRIO

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

A Internet das Coisas (IoT) vem revolucionando a forma como vivemos e trabalhamos. A forma como estes dispositivos captam informações do ambiente com seus sensores para compartilharem entre si e com humanos possibilitando tomadas de decisão inteligentes, que visam beneficiar o ecossistema como um todo, é o que qualifica seu grande potencial de mudanças. O crescimento na quantidade de dispositivos IoT é notável nas atividades do dia-a-dia. Há poucos anos não se imaginava o quão populares se tornariam, nas mais variadas formas como pulseiras e relógios inteligentes que monitoram o sono ou nossas atividades físicas diárias. A facilidade ao se lidar com assistentes inteligentes, que nos informam a previsão do tempo com lembretes em nossas agendas com um mero comando de "bom-dia", até ligam e desligam máquinas de café, lâmpadas e diversificados dispositivos *smarts*. A quantidade de dispositivos conectados Machine-to-Machine (M2M) deve ultrapassar o número de pessoas utilizando serviços de inscrição como telefones, computadores e tablets até 2020 e até 2024 a indústria IoT no geral deve movimentar até 4.3 trilhões de dólares (Sanchez-Gomez; Sanchez-Iborra; Skarmeta, 2017).

A IoT demanda soluções leves e escaláveis, com garantias de segurança e privacidade. A tecnologia *blockchain*, tem o potencial de atender a estas demandas, como resultado de sua natureza distribuída, segura e privada. Essencialmente, *blockchain* é o banco de dados distribuído de registros ou um livro-razão público de todas as transações ou eventos digitais que foram executados e compartilhados entre seus participantes. Cada transação do livro-razão é verificada por um consenso da maioria dos participantes do sistema e, uma vez aceitas, não podem ser apagadas. Esta tecnologia estabelece um sistema de criação de consenso distribuído no mundo digital online - o qual permite que entidades participantes tenham certeza de que um evento digital aconteceu ao criar um registro irrefutável em um livro-razão público.

O *blockchain* abriu as portas para o desenvolvimento de uma economia digital democrática, aberta e escalável, a partir de uma economia centralizada. Desde sua criação, com a criptomoeda Bitcoin, outras formas de dinheiro eletrônico com estruturas semelhantes surgiram. Ao mesmo tempo, diferentes aplicativos usando *blockchain* foram desenvolvidos ao longo dos anos para implementar outros cenários além

das criptomoedas: novos conceitos, como contratos inteligentes e propriedades inteligentes, entraram em cena.

A economia das coisas (EoT) é a monetização das coisas, a disponibilização de ativos digitais fornecidos por dispositivos IoT em *marketplaces*. A integração entre blockchains e IoT pode trazer inúmeras vantagens a este movimento. Com o recente lançamento feito pela IOTA *Foundation* do *Industry Marketplace* - um *marketplace* descentralizado voltado para a indústria de IoT (a indústria 4.0), este conceito ganha visibilidade, evidenciando o potencial por trás desta união. Contudo, isto não implica em inexistência de desafios a serem ultrapassados, pelo contrário. A interoperabilidade e heterogeneidade em IoT, suas restrições computacionais e energéticas, estão entre estes desafios.

Diante estes desafios, é proposta - neste trabalho - uma arquitetura de microsserviços para superar a heterogeneidade entre dispositivos e protocolos, num contexto de microtransações financeiras de EoT. A flexibilidade de microsserviços em comparação a abordagens mais monolíticas torna este tipo de arquitetura atraente para solucionar os problemas originados neste contexto.

1.2 OBJETIVOS

O objetivo principal deste trabalho é propor uma arquitetura para controle de micro transações financeiras em um contexto de Economia das Coisas (EoT). Tal arquitetura deverá ser flexível, de forma que sua integração com diferentes protocolos e tecnologias seja otimizada.

1.2.1 Objetivos Específicos

Os objetivos específicos que podem ser citados são:

1. Desenvolver *gateway* de saldo para controlar a efetivação e confirmação de transações com os dispositivos IoT
2. Desenvolver *gateway* de comunicação entre os dispositivos IoT e marketplaces
3. Desenvolver *gateways* de comunicação com criptomoedas.
4. Desenvolver *gateway* de monitoramento de transações dos dispositivos

5. Desenvolver módulo de autenticação dos dispositivos participantes da arquitetura

6. Validar a utilização da arquitetura proposta através da utilização de tipos diferentes de criptomoedas e dispositivos em um ambiente experimental.

1.3 METODOLOGIA

Este trabalho foi desenvolvido com os seguintes critérios de pesquisa:

- Bibliográfica, sobre estudos existentes do estado da arte e pesquisas correlatas aos temas aqui abordados;

- Experimental, através elaboração de caso de testes para validar a proposta.

Para garantir estes critérios, foram seguidas as etapas abaixo descritas:

- (a) Fundamentação bibliográfica;

- (b) Elaboração da proposta de arquitetura do trabalho;

- (c) Desenvolvimento dos módulos para validação da arquitetura;

- (d) Simulação do funcionamento dos módulos;

- (e) Análise dos resultados obtidos na simulação;

- (f) Elaboração de relatório descritivo da proposta e sua validação.

1.4 ORGANIZAÇÃO

Objetivando melhor compreensão e organização dos conteúdos, este trabalho está organizado em 6 capítulos. Este primeiro, que apresenta a introdução, esclarecendo as motivações e os objetivos do trabalho.

O capítulo 2, com a exposição da fundamentação teórica, trazendo as definições dos conceitos que permeiam esta proposta. Neste ainda, há uma seção com trabalhos correlatos ao aqui proposto.

O capítulo 3 traz a proposta em si, apresentando a arquitetura desejada e o objetivo pretendido com cada decisão tomada.

O capítulo 4 apresenta o ambiente experimental e como ele foi desenvolvido para validação da arquitetura.

O capítulo 5 é dedicado à apresentação dos resultados obtidos com esta proposta e o experimento realizado.

O capítulo 6, traz as conclusões do trabalho e identificação de trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 MICROSSERVIÇOS

Microsserviço é o termo utilizado para definir aplicações projetadas como um conjunto de serviços autônomos (BARROZO, 2016). De acordo com Butzin, Golatowski e Timmermann (2016), a arquitetura de microsserviços não foi inventada, mas surgiu de experiências e boas práticas essenciais elencadas por usuários de SOA (BACK, 2016).

Apesar de não haver uma definição formal de microsserviços, de maneira informal, pode-se dizer que são uma abordagem para desenvolver uma aplicação como um conjunto de serviços menores, focados em tarefas pequenas com tecnologias leves (Krylovskiy; Jahn; Patti, 2015; NEWMAN, 2015; Jamshidi et al., 2018).

Acima de tudo, esta arquitetura se tornou uma proposta para suportar modelos de negócios altamente escaláveis e mutáveis, além de seu elevado grau de manutenibilidade, visto que serviços podem ser escritos em diferentes linguagens de programação, proporcionando grande heterogeneidade.

Cada microsserviço é implementado e operacionalizado como um pequeno - porém independente - sistema. E através de uma interface de rede adequada, disponibiliza acesso a sua lógica e dados, aumentando a agilidade do software, pois cada microsserviço é uma unidade autônoma de desenvolvimento, implantação, operação, versionamento e escalabilidade (Jamshidi et al., 2018).

Dentre suas características, Lewis e Fowler (2014) listam que as mais comuns são sua organização, capacidade comercial, implantação automatizada, inteligência nas chamadas da interface do sistema e controle descentralizado de idiomas e dados.

Em comparação a uma arquitetura monolítica, Lucio (2017) considera que as arquiteturas em microsserviços ganham destaque em aplicações de maior porte, onde seus benefícios são facilmente identificados, como os listados por Back (2016):

- Alinhamento organizacional com projetos e equipes menores;
- Independência do restante do sistema;
- Facilidade de entrega;
- Entrega contínua;

- Liberdade para escolha de tecnologias heterogêneas;
- Liberdade para substituição e composição de serviços.

A versatilidade alcançada por sistemas baseados na arquitetura de microsserviços, viabiliza fácil integração de diferentes tecnologias focadas na resolução de um problema em comum. Os microsserviços podem internamente possuir das mais diversas abordagens de desenvolvimento, sem que afete a funcionalidade do sistema como um todo, podendo ser moldados na melhor forma proposta para realização de suas próprias tarefas.

2.2 IOT

A Internet das Coisas (IoT) consiste em objetos conectados, que detectam e coletam dados de seus arredores, que então são utilizados na realização de tarefas automatizadas visando ajudar as pessoas.

Estima-se que quantidade de dispositivos inteligentes no mundo ultrapasse os 250 bilhões até o final da década (FAMILIAR, 2015). Esta constante taxa de crescimento de dispositivos conectados, capazes de interagirem e comunicarem entre si, onde tudo está conectado é conhecida como IoT.

Conceitualmente, a IoT permite que estes dispositivos se comuniquem, compartilhem informações, coordenem decisões e desempenhem atividades (AL-FUQAHA et al., 2015). Segundo Atzori, Iera e Morabito (2010), a IoT consiste na presença ubíqua destes dispositivos ou *coisas* a nossa volta, nas suas mais variadas formas, de sensores de temperatura até *smartphones* e carros.

Diante de sua ampla aplicabilidade, existem fábricas - a chamada Indústria 4.0 (SCHWAB, 2017), prédios inteligentes, redes de sensores militares, até cidades - chamadas *smart cities*.Desta forma, a IoT possui enorme potencial de mudanças impactantes no dia a dia de seus usuários (ATZORI; IERA; MORABITO, 2010).

Tamanhas aplicabilidades e crescimento, resultam em pesquisas sobre padrões de uso e comunicação dos dados gerados e trafegados por estes dispositivos, bem como propostas de arquiteturas, como exemplificado em Agostinho et al. (2018), Al-Fuqaha et al. (2015), Atzori, Iera e Morabito (2010), Ngu et al. (2017), Butzin, Golatowski e Timmermann (2016), Sheng et al. (2013).

Fatores que impulsionam este crescimento vão desde o crescimento da rede mundial de computadores, com a expansão da internet

e de tecnologias sem fio, tais como as redes celulares até a introdução de dispositivos vestíveis como relógios inteligentes, o progresso das tecnologias de armazenamento e a *cloud computing* (AL-FUQAHA et al., 2015).

As restrições em termos de recursos são uma das caraterísticas marcantes dos dispositivos IoT: baixo poder de processamento e pequena capacidade energética, são algumas que podem ser citadas. Da necessidade de sobrepor restrições como essas, são propostos padrões e protocolos, como visto nas *Low-power wide area networks* e no próprio *Bluetooth Low Energy*, dentre outros conforme serão apresentados na seção 2.3.

2.3 PROTOCOLOS PARA IOT

Não há consenso quanto a comunicação em um contexto de IoT, assim, existem diversos protocolos em uso atualmente, cada qual buscando eficiência em diferentes aspectos. Sheng et al. (2013) analisa através dos protocolos existentes, como propor melhorias nestes protocolos, com tentativas de padronização na área, enquanto Granjal, Monteiro e Sá Silva (2015), faz um levantamento sob uma perspectiva de segurança na sua comunicação.

Já o autor Al-Fuqaha et al. (2015) os classifica em 4 grandes categorias: protocolos de aplicação; protocolos de descoberta de serviço; protocolos de infraestrutura e outros protocolos influentes. É possível detalhar além destas categorias, identificando em nível de camadas,tais como: de roteamento, de rede, física, de dados, de infraestrutura e de comunicação.

Constrained Application Protocol (CoAP), tem funcionamento a nível de aplicação, atua sobre o protocolo HTTP utilizando o protocolo UDP e o padrão REST para comunicação. Sua a conexão ativa entre nós proporciona um tempo de resposta inferior (ROTTA; DANTAS, 2017).

Apesar de WiFi (IEE 802.11.x) ser um dos padrões mais utilizados na comunicação sem fio, dispositivos IoT são limitados e necessitam de mecanismos de economia de energia para operar satisfatoriamente por longos períodos. Desta forma, o WiFi foi criado para redes com grande largura de banda e alcance de transmissão, mas estas funcionalidades necessitam de alto suprimento de energia e a incapacidade de Wifi operar com eficiência nessas condições, levou a pesquisa por novos padrões de comunicação sem fio (SIDDIQUI et al., 2019).

Bluetooth Low Energy (BLE) e ZigBee operam na comunicação

sem fio. Ambos são propostos para uso de comunicação com baixo consumo energético, embora ZigBee ainda não esteja difundido no mercado de *smartphones* e dispositivos móveis (AGOSTINHO et al., 2018). Mesmo com o grande potencial de uso em IoT por BLE, seu uso indiscriminado pode prejudicar a visão de sua concepção e por isto já existem propostas para adequar-se a ambientes amplamente ocupados por dispositivos utilizando BLE (Harris III et al., 2016).

Mesmo que ambos sejam considerados viáveis para implementação de serviços IoT, devido ao baixo consumo energético, a limitada cobertura oferecida por eles é um obstáculo, principalmente em situações que considerem um perímetro urbano, como em *Smart Cities* (Centenaro et al., 2016).

Low-power wide area networks (LPWANs) são soluções alternativas, que operam entre as soluções de longo e curto alcance. Redes deste tipo exploram frequências não licenciadas de bandas, têm como características topologias estrela e conexões de rádio de longo alcance, com uma arquitetura feita para cobrir uma grande área, garantindo conectividade entre os nós presente nos mais adversos ambientes (Centenaro et al., 2016). São protocolos LPWAN, a SigFox e LoRa/LoRaWan.

SigFox foi fundada em 2009 e permite a transferência de uma pequena quantidade da dados - 10 a 1000 bits por segundo, com a tecnologia de *Ultra Narrow Band* (UNB) . Capaz de rodar com uma pequena bateria (Al-Sarawi et al., 2017), foi a primeira tecnologia proposta para o mercado em LPWAN, e diz suportar até um milhão de objetos conectados com uma cobertura de 30-50km em zonas rurais e 3-10km em perímetros urbanos (Centenaro et al., 2016).

Inicialmente, a SigFox possuía comunicação unidirecional partindo dos dispositivos, mas atualmente a comunicação bidirecional é suportada. Não há documentos públicos sobre os protocolos da camada de rede de SigFox, pois são proprietários.

Já a LoRa, é a a camada física da LPWAN, desenvolvida e patenteada pela Semtech Corporation - o qual possui duas camadas: LoRa PHY e LoRaWAN, sendo a LoRa PHY a camada proprietária, que suporta múltiplos canais, o que torna possível a maiores trocas de dados, com mensagens mais longas ou com maior alcance (Centenaro et al., 2016).

Enquanto LoRaWAN, é aberta e seu desenvolvimento é feito pela LoRa Alliance, liderado pela IBM, Actility, Semtech, e Microchip. Este utiliza o identificador único estendido IEEE 64-bit(EU) para associar automaticamente endereços IPv6 com nós LoRa. Portanto, protocolos IPv6 podem ser implantados em redes LoRaWAN, permitindo intero-

perabilidade transparente com as comunicações baseadas em IP.

2.4 BLOCKCHAIN E CRIPTOMOEDAS

Blockchain é uma estrutura de dados distribuída, replicada e compartilhada entre membros de uma rede P2P, foi inicialmente desenvolvida para uso com a criptomoeda BitCoin - visando acabar com o problema do gasto duplo, mas suas aplicações vão além disto (FERNáNDEZ-CARAMéS; FRAGA-LAMAS, 2018).

O seu funcionamento é basicamente uma cadeia de blocos, aonde cada um deles possui uma lista de transações e o *hash* do bloco anterior. Cada bloco é um nó de uma rede que possui um par de chaves pública/privada, utilizadas na leitura e validação de transações. Quando um nó desta rede recebe uma transação ela é validada; havendo êxito a transação é assinada e retransmitida aos blocos adjacentes e assim sucessivamente, caso contrário ela é descartada (CHRISTIDIS; DEVETSIKIOTIS, 2016).

Criptomoedas são um esquema de troca digital P2P que gera e distribui valores monetários usando criptografia (FARELL, 2015). Este processo necessita um verificação distribuída de transações, sem uma autoridade central. As verificações da transações confirmam seus valores e se o pagador detém a moeda que ele deseja gastar, garantindo que unidades não sejam gastas em duplicidade. Este método de verificação é chamado de mineração (MUKHOPADHYAY et al., 2016), e diferentes criptomoedas utilizam diferentes métodos, conforme suas necessidades.

2.4.1 Bitcoin

Bitcoin é a primeira criptomoeda implementada inteiramente descentralizada (MUKHOPADHYAY et al., 2016). Publicada por um grupo de programadores ou um programador desconhecido sob o pseudônimo de Satoshi Nakamoto em 2008 e implementada em 2009, Bitcoin não é uma empresa nem um produto (Manimuthu et al., 2019).

Segundo Nakamoto et al. (2008), o Bitcoin é na verdade uma plataforma que permite dois indivíduos efetuarem transações financeiras sem custos de mediação relacionados a comércio eletrônico e sem o envolvimento de terceiros. Seu sistema de pagamentos é baseado em provas criptográficas ao invés de uma entidade de confiança, como um governo ou organização. Atualmente, estima-se que existam entre 13 e

32 milhões de usuários de Bitcoin.

Nos parágrafos seguintes seu funcionamento será explicado, embasados no whitepaper publicado por Nakamoto et al. (2008).

Nakamoto define uma moeda eletrônica como uma cadeia de assinaturas digitais. Cada transferência é um conjunto de uma assinatura de um *hash* da transação anterior e a chave pública do próximo dono da moeda. As assinaturas podem ser verificadas através da cadeia de propriedade da moeda.

Um grande problema deste sistema é a impossibilidade de verificação da existência de gasto duplo, desta forma, para resolver isto, foi proposto utilizar um servidor de *timestamp* P2P que funciona tomando um *hash* de um bloco de itens a registrar o *timestamp* e publicar o *hash* amplamente. Esse prova que os dados existiam naquele momento, para entrarem no *hash* gerado. Cada *timestamp* inclui o *timestamp* anterior, formando uma cadeia que com cada ciclo adicional, reforça os *hashes* anteriores. A chave pública é armazenada em uma carteira, que pode ser implementada em software, hardware ou online.

Todas as transações são registradas no livro-razão do Bitcoin, além de conter as propriedades na rede e cada nó mantém uma cópia dos registros do livro-razão. Para enviar uma quantidade de moedas de um usuário para outro, é necessário anunciar publicamente a transação desejada e caberá a rede validar sua exatidão. Cada transação é definida por seu valor em *hash* representando um identificador e um conjunto de entradas e saídas. Assim, cada saída da transação só pode ser usada uma vez como entrada em toda a *blockchain*. A tentativa de desviar desta característica é proibida na rede, pois implica no problema de gasto duplo.

Objetivando prevenir esta situação citada anteriormente, é utilizado o *proof-of-work* - PoW. Com o PoW a rede Bitcoin exige que cada nó faça consideráveis trabalhos computacionais para provar que são membros válidos da rede.

Desta forma, enquanto a força computacional dos nós honestos da rede for maior que a dos maliciosos, a rede continuará consistente e todas as transações legítimas acontecerão.

Em suma, os passos da rede ocorrem da seguinte forma: uma nova transação é transmitida a todos os nós, então cada nó coleta novas transações em um bloco. Cada nó trabalha para encontrar uma PoW difícil para seu bloco e quando ele é encontrado, é transmitido para todos os nós. Os nós então aceitam o bloco apenas se todas as transações nele forem válidas e ainda não gastas (para evitar o gasto duplo). Por fim, os nós expressam o aceite do bloco, trabalhando na

criação do próximo bloco da rede, usando o *hash* do bloco aceito como o *hash* anterior.

2.4.2 Iota

Criada em 2015, Iota é uma criptomoeda de microtransações otimizada para IoT. Diferente das demais *blockchains*, complexas e pesadas como Bitcoin, criadas para diferentes usos, Iota foi criada para ser o mais leve possível, voltada especificamente para IoT.

Ela foi concebida com a expectativa de 50 bilhões de dispositivos IoT conectados, acreditando que um dos obstáculos deste futuro seria a necessidade de realizar microtransações, Iota foi criada para que estes dispositivos pudessem pagar automaticamente quantias minúsculas uns para os outros.

Buscando oferecer uma solução aos problemas de escalabilidade e altas taxas sobre as tecnologias de *blockchain*, sua principal diferença das criptomoedas tradicionais é utilizar um Grafo Acíclico Dirigido (DGA) - chamado de *Tangle*, ao invés de um *blockchain* global (TENNANT, 2017). Funcionalmente, Tangle se distingue no tratamento de novas transações: ao invés de uma transação precisar ser validada por toda a cadeia para ser aceita, quando uma nova transação chega, ela deve validar duas transações anteriores para poder ser inserida na rede.

Isto, garante que novos usuários precisem trabalhar validando transações antes de poder emitir uma, assim contribuindo para a segurança da rede (POPOV, 2018).

2.4.3 Ripple

Ripple é o nome utilizado tanto para referenciar ao protocolo de pagamento Ripple quanto a criptomoeda XRP. O XRP Ledger é um sistema distribuído de pagamento que permite a fácil transferência de valores pelo mundo (CHASE; MACBROUGH, 2018). Operado dentro de uma rede P2P distribuída, enfrenta os mesmos desafios que outras moedas digitais, como a prevenção de gasto duplo e garantia da consenso por toda a rede sobre o estado das contas de usuários e saldos.

Proposto em Schwartz et al. (2014), o algoritmo enfrenta estes desafios através de um protocolo Bizantino de acordo tolerante a falhas, sob sub-redes coletivamente confiáveis, chamado de *XRP Ledger*

Consensus Protocol (XRP LCP). Posteriomente, Chase e MacBrough (2018) apresentou um detalhamento do algoritmo e também de condições para segurança em sua utilização.

O código por trás de Ripple é *open source*, o que facilita para que qualquer pessoa possa criar uma instância Ripple. Os nós da rede Ripple podem ser de três tipos distintos: usuários, capazes de fazer e receber pagamentos; *market makers* (formadores de mercados), facilitadores de transações, geram e suportam liquidez monetária; e servidores de validação, que executam o XRP LCP para verificar e validar todas as transações da rede (ARMKNECHT et al., 2015).

Para chegar ao consenso, são utilizadas listas de nós únicos (UNL) no XRP LCP. Um nó necessita consultar somente a sua UNL ao invés de a rede inteira para chegada de consenso (CHRISTIDIS; DEVETSIKIOTIS, 2016).

O funcionamento do XRP LCP de uma maneira geral pode ser descrito como uma máquina de estados replicada (CHASE; MACBROUGH, 2018). O estado replicado é o livro-razão mantido por cada nó da rede e a mudança de estados corresponde às transações submetidas por usuários. Quando os nós definem o conjunto de transações a serem aplicadas ao estado, o protocolo de processamento de transações define regras determinísticas para ordenação das transações e de como aplicá-las para gerar um novo estado do livro-razão. Assim, o XRP LCP deve apenas fazer com que a rede chegue em um consenso quanto ao conjunto de transações, não quanto ao conteúdo ou resultado delas. Desta forma, ele garante que o livro-razão seja consistente em todos os nós da rede.

2.4.4 Criptomoedas com foco em IoT

Iota não é a única proposta de criptomoeda com foco em IoT, é apenas a de maior sucesso dentre elas. A seguir, serão listadas algumas delas, trazendo suas características e objetivos.

- IoT Chain - IoT Chain é um sistema operacional seguro e leve para a IoT, alimentado pelo *blockchain*. Ele combina a *blockchain* com a DGA e o uso de criptografia assimétrica, para operar dispositivos IoT com baixo poder computacional. Usando essa mistura de tecnologias, o grande objetivo da IoT Chain é fornecer a estrutura para conectividade segura e de baixo custo para IoT (CHAIN, 2018);

- IoTeX - Projeto *open-source* de 2017, promete um *blockchain* auto-escalável e centrado em privacidade para IoT. IoTeX almeja criar uma malha de confiança descentralizada para a era da colaboração e troca de dados entre dispositivos, aplicações e pessoas. Ainda em desenvolvimento, está uma arquitetura de *blockchain*-em-*blockchain* para computação heterogênea, um algoritmo de consenso rápido e robusto de *Roll-Delegated Proof of Stake* (Roll-DPoS), e protocolos computacionais confiáveis (TEAM, 2018);

- LightChain - Proposto em 2019, LightChain é um sistema *blockchain* leve, eficiente na atualização de recursos e adequável para cenários com restrição de energia na IoT Industrial (IIoT). Utiliza um mecanismo de consenso *green*, chamado de *Synergistic Multiple Proof*, para estimar a cooperação entre dispositivos IIoT e também uma estrutura de dados leve chamada de LightBlock para otimizar a transmissão de conteúdos. Além disso, conta com um filtro de transferência de blocos não relacionados, para evitar o crescimento ilimitado do livro-razão, sem afetar a rastreabilidade da *blockchain* (Liu et al., 2019);

- RuffChain - Publicada em 2017, está em implementação com planejamento de lançamento em 2019. Ruff possui uma arquitetura *blockchain open-source* descentralizada, focada em alta eficiência para o desenvolvimento de aplicações IoT. Sua proposta é resolver o problema de operações confiáveis e operações onerosas entre sistemas IoT em diferentes domínios, criando assim um ecossistema aberto Ruff Chain. A garantia de disponibilidade se faz através da combinação de Edge Computing e *blockchain* (CHAIN, 2017).

2.5 ECONOMIA DAS COISAS

A Economia das Coisas é um conceito recente que descreve a monetização das coisas. A disponibilização de produtos ou serviços digitais fornecidos por dispositivos IoT em *marketplaces*, é o que constitui a EoT.

Lougee e Pureswaran (2015) demonstra com um estudo de caso, como a EoT implicará na liquidez de diversos mercados, exemplificando através do imobiliário e de pequenas e médias empresas, fazendo uma relação direta com a transformação na aviação norte americana originada pela digitalização do mercado de assentos das aeronaves. Este impacto foi analisado por todo um ciclo de aproximadamente 50 anos,

com início entre os anos 1960 e 1970 até 2013.

Em Huckle et al. (2016), através da identificação em dados reais, são abordadas as vantagens da associação entre IoT e *Blockchain* para beneficiar aplicações de economia compartilhada - que, em suma, são são aplicações para monetizar coisas, como um quarto de sua casa(airbnb) ou seu carro(Uber).

Outro exemplo, como em um dos cenários mencionados, aonde com sensores IoT em um carro, ele teria a capacidade de sugerir alterações em rota, com o objetivo de abastecer o combustível, além de já efetuar as devidas transações financeiras necessárias para pagamento através de uma *blockchain* vinculada a utilizados do veículo.

Recentemente, a IOTA Foundation lançou o *Industry Marketplace* (FOUNDATION, 2019), que nada mais é do que um *marketplace* voltado diretamente para a indústria 4.0, oferecendo a rede Tangle como base para comunicação e infraestrutura computacional, capaz de suportar todos os atores envolvidos na cadeia de valores do segmento.

O *Industry Marketplace* estabelece um mercado neutro para os componentes da indústria 4.0 para compra e venda de mercadorias, dados e serviços, através de uma plataforma autônoma e descentralizada para oferecer e pesquisar dados e serviços, de forma gratuita e aberta.

A trabalho conjunto de *Blockchains* e IoT é objetivo de pesquisas, Fernández-Caramés e Fraga-Lamas (2018), Dai, Zheng e Zhang (2019) elencam alguns desafios que necessitam ser abordados, buscando maneiras de adaptar a tecnologia *blockchain* às necessidades de IoT, dentre eles: a descentralização, baixa interoperabilidade, vulnerabilidades de privacidade e segurança.

Skwarek (2017) busca um método para permitir que dispositivos IoT alcancem um nível industrial de confiabilidade para transferência de informações de sensores sem fio para sistemas de produção, através de mecanismos de *blockchain* que assegurem comunicação segura de uma forma leve e escalável.

Segundo Zheng et al. (2018), *blockchain* pode ter diversificadas aplicações, muito além de criptomoedas, considerando como permite que pagamentos sejam finalizados sem nenhum banco ou intermediários, podendo ser utilizada tanto em serviços financeiros quanto pagamentos online.

2.6 TRABALHOS CORRELATOS

Para elaboração da proposta deste trablho, foram analisados diversos trabalhos acadêmicos, alguns com abordagens semelhantes a aqui propostas, que serão melhor descritos em sequência.

Krishnamachari et al. (2018) encara os desafios da integração de dispositivos IoT no atendimento de larga escala, como um cidade. Para isto, ele propõe um Integrador de IoT Inteligente (I3). O foco do I3 é em dados dinâmicos, para construção de comunidades de dispositivos inteligentes, aonde o fluxo de dados se torno como um rio, permitindo que diferentes entidades se unam para analisar, processar e agir de forma a suportar um conjunto diverso de aplicações. I3 atua sobre a camada de troca de dados acima da camada de transporte e propõe que donos de dispositivos possam vender dados enquanto desenvolvedores de aplicações podem comprar dados para poder melhorar suas aplicações. Uma prova de conceito foi realizada na Universidade do Sul da Califórnia e em novembro de 2017 foi aberto um consórcio de empresas aonde diversas empresas se uniram ao projeto e estão apoiando sua visão.

Em Mišura e Žagar (2016), partiu-se do mesmo ideal de monetização dos dados gerados por IoT. Neste trabalho é proposta uma arquitetura com interface web, para o registro de dispositivos e consumidores. Com os dados dos registros web feitos, possíveis consumidores podem pesquisar por dados que lhes tragam maior benefício, baseados em parâmetros do sistema, como localização, orçamento e tempo de vida do dado em si. A credibilidade dos dados disponibilizados, viria de uma medida sobre dados vendidos serem válidos para quem os comprou. Em um experimento realizado, foi observado que a performance de buscas segue uma complexidade linear quanto ao número de dispositivos registrados no sistema. Em conclusão, acreditam que um *marketplace* voltado a dados de dispositivos IoT poderia fomentar a compra e uso de dispositivos IoT pelas pessoas, visto que poderiam monetizar estes dispositivos, o que ocasionaria uma maior disponibilização de dados causando uma melhora nas aplicações IoT, com uso destes dados.

Bröring et al. (2017) acredita no potencial que *marketplaces* de dados de IoT tem, mas não deixa de lado um grande desafio do setor: a interoperabilidade de dispositivos IoT. Para enfrentar este desafio, neste trabalho é proposta uma "Ponte para a abertura da interoperabilidade de IoT"(*BIG IoT - Bridging the Interoperability Gap of the IoT*). A proposta é criar uma arquitetura de um ecossitema IoT que

quebre essa barreira, conectando provedores de *things*, serviços e usuários, aproveitando da Web Semântica. O grande desafio atualmente enfrentado por esta proposta é que desenvolvedores comecem a utilizá-lo, visto que já há uma API para uso da BIT IoT.

3 ARQUITETURA PROPOSTA

Neste trabalho, é proposta uma arquitetura para o controle de micro transações financeiras em um contexto de EoT. Esta arquitetura, deve ser adequada a comportar quaisquer criptomoedas desejadas, com uma abstração da comunicação entre dispositivos fornecedores de serviços ou produtos, e os consumidores, fazendo então o controle das microtransações financeiras.

Uma situação de uso da arquitetura, seria com um carro inteligente. Este carro teria sensores que identificariam a quantidade de combustível remanescente no carro e a necessidade de abastecimento, pois também saberia que seu dono faria um trajeto até sua casa ao fim da jornada de trabalho. Este carro com a informação do trajeto e distância que o carro conseguiria percorrer com o combustível restante, verificaria através da arquitetura, postos de combustíveis no trajeto, selecionando com melhor custo e menor alteração no trajeto, para sugerir uma rota ao motorista. Ao chegar ao posto, através de uma carteira de criptomoeda já vinculada ao carro, o combustível poderia ser pago com ela.

A Figura 1 foi elaborada para representar como estes atores estariam na arquitetura proposta, havendo os carros inteligentes com seus sensores, comunicações com criptomoedas e outros dispositivos fornecendo seus produtos, como postos de combustível.

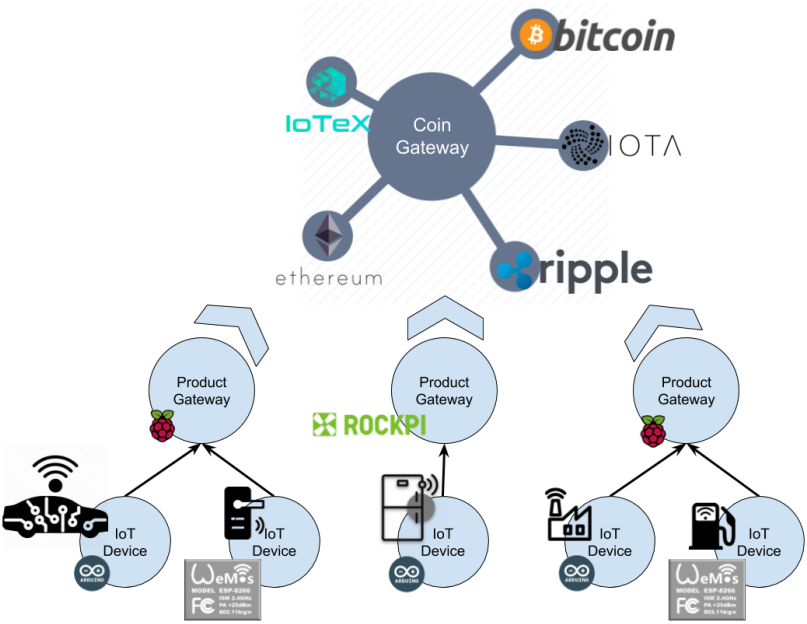

Figura 1 – Abstração de exemplo da arquitetura proposta

Após análise dos benefícios do uso de microsserviços, optou-se por sua utilização para enfrentar os desafios de interoperabilidade e heterogeneidade em ambientes IoT, além de sua flexibilidade no que diz respeito à possibilidade de melhorias contínuas com o incremento ou até substituição de microsserviços sem um impacto sistêmico.

Na arquitetura proposta, foram imaginados microsserviços, com sua organização apresentada na Figura 2 e descritos em sequência.

- *Service Registry*
- *IoT Devices*
- *Product Gateway*
- *Coin Gateways*
- *Balance Control*
- *Transaction Watcher*
- *Marketplace*

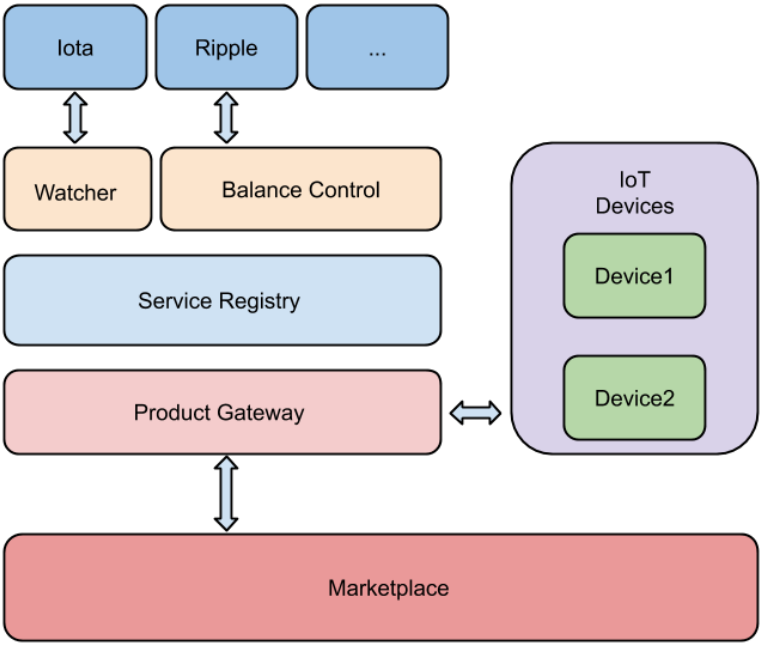

Figura 2 – Organização da arquitetura proposta

Assim, tem-se a estrutura principal da arquitetura - no qual apresenta os pontos de comunicação entre os módulos.

3.1 SERVICE REGISTRY

Módulo responsável por manter o registro dos serviços do sistema. Nele, serão armazenados os endereços e tipos dos demais serviços que integram o sistema. A base de seu funcionamento é: os serviços se cadastram nele e requisitam os dados de outros serviços nele cadastrados para poderem se comunicar.

Nesta troca, o Service Registry fornece as chaves de autenticação para se comunicarem. Este módulo é essencial ao sistema por manter quais serviços pertencem a ele afim de propiciar a integração entre eles.

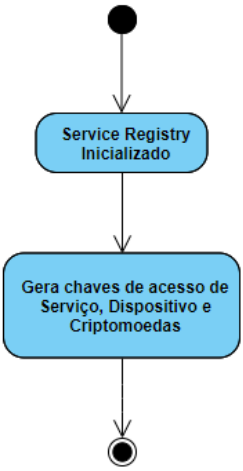

Figura 3 – Fluxo de inicialização do módulo Service Registry

Em sua inicialização - apresentada na figura 3, o serviço gera essas chaves de acesso: uma do próprio serviço, uma dos dispositivos IoT e outra de criptomoedas.

A medida que cada tipo de serviço se registra no Service Registry, será disponibilizada a respectiva chave de acesso. A figura 4 representa este fluxo de registro.

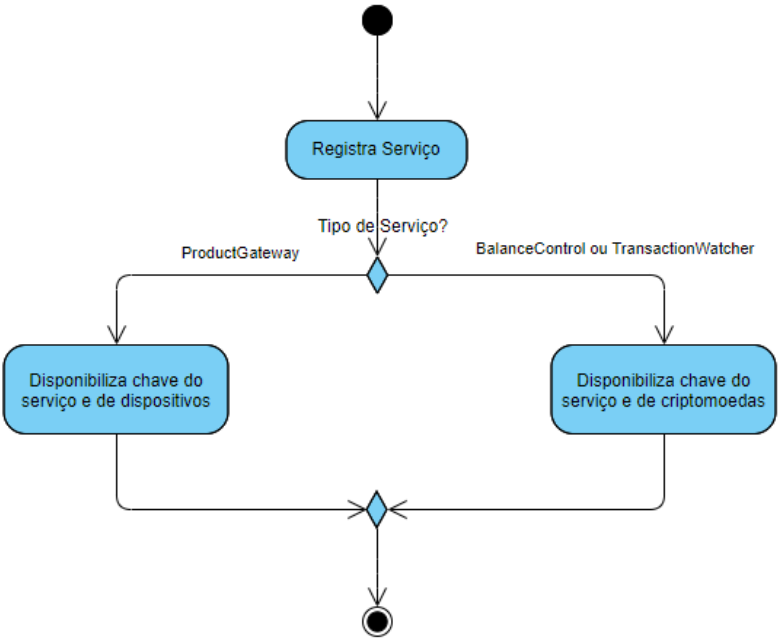

Figura 4 – Fluxo de registro de serviços do módulo Service Registry

A chave de acesso é a mesma disponibilizada aos dispositivos e as criptomoedas quando eles se registram.

Quando os demais serviços requisitarem deste serviço as listagens de dispositivos ou criptomoedas, os mesmos deverão autenticar-se com a chave recebida no seu registro, conforme ilustrado na figura 5.

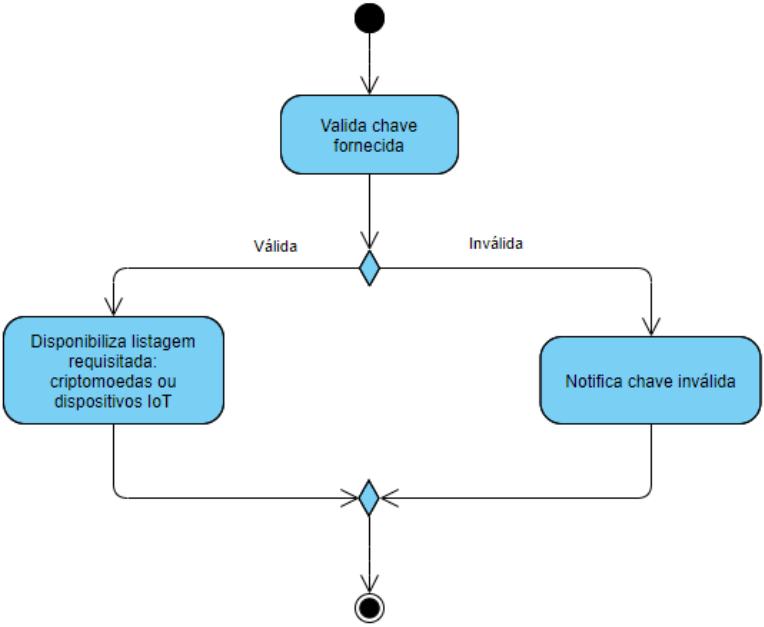

Figura 5 – Fluxo genérico de listagem do módulo Service Registry

Ressalta-se que cada tipo de serviço deverá ter seu meio de acesso exclusivo.

3.2 IOT DEVICES

No módulo de Dispositivos IoT estão configurados os produtos fornecidos por ele. Em sua inicialização, representada na Figura 6, o dispositivo registra-se no Service Registry.

Figura 6 – Fluxo de inicialização do módulo IoT Device

Os produtos registrados serão utilizados pelo Product Gateway, quando este comunicar-se com o dispositivo para realizar reservas e confirmação de pedido.

Ao registrar-se no Service Registry, o dispositivo informa seu produto e recebe sua chave de autenticação.

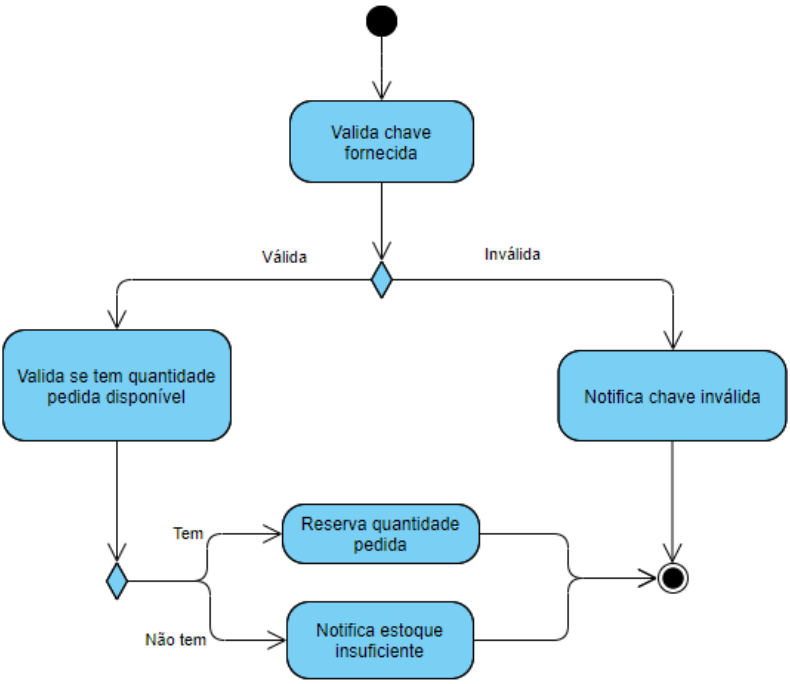

Figura 7 – Fluxo de reserva do módulo IoT Device

O fluxo de reserva apresentado na figura 7, a reserva só ocorre caso o dispositivo possua estoque suficiente do produto reservado.

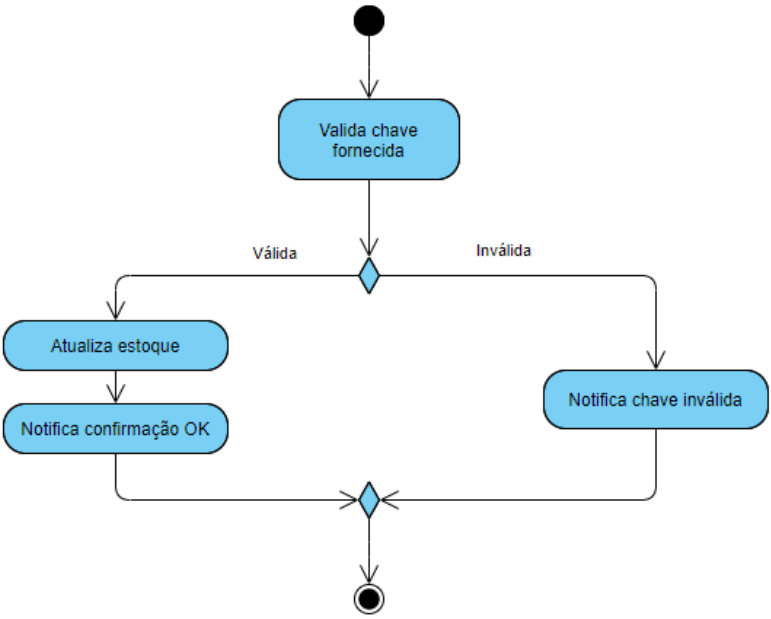

Figura 8 – Fluxo de confirmação do módulo IoT Device

No momento da confirmação, ilustrado na figura 8, a confirmação acontece somente se a devida chave de autenticação for fornecida.

3.3 PRODUCT GATEWAY

Product Gateway é uma abstração dos produtos fornecidos pelos dispositivos IoT. Neste módulo, estarão disponíveis as informações destes, além de métodos para interação com os produtos por ele fornecidos. Dentre as operações, possíveis de se realizar, estão a consulta de dados dos produto, suas formas de pagamento, reservas de produtos com os dispositivos, registro de pagamento e confirmação de transação referente ao pagamento.

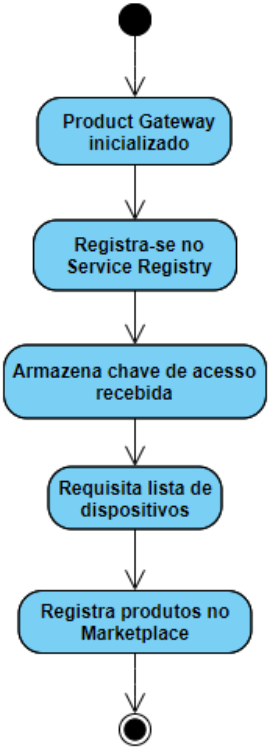

Figura 9 – Fluxo de inicialização do módulo Product Gateway

Na Figura 9 é apresentado o fluxo de inicialização do serviço.

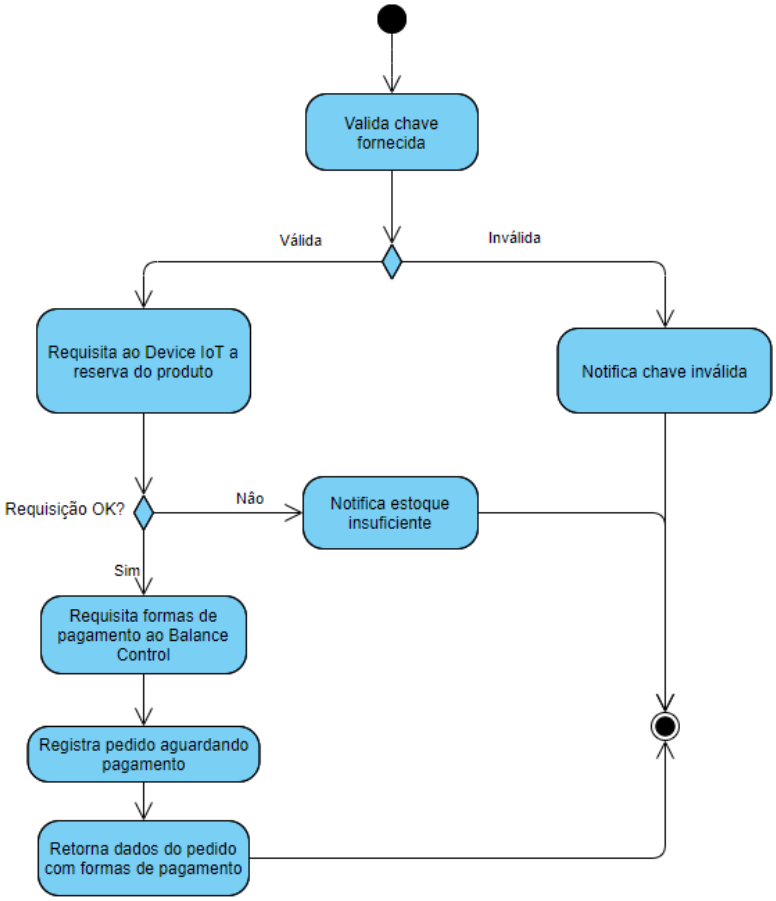

Figura 10 – Fluxo de reservas do módulo Product Gateway

O fluxo de reserva de um produto é ilustrado nas Figura 10.

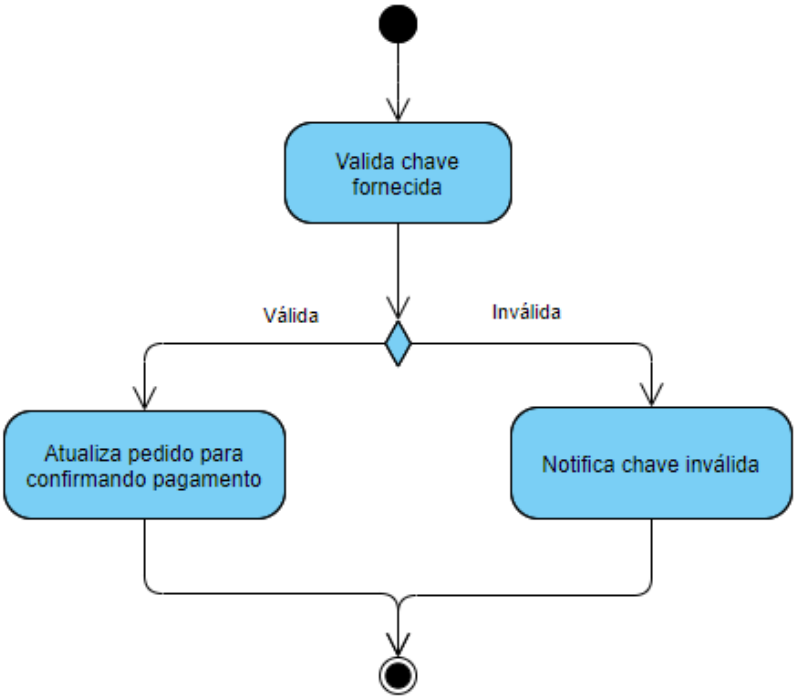

Figura 11 – Fluxo de processamento de pagamento do módulo Product Gateway

A figura 11 ilustra o fluxo de execução do processamento de pagamento no Product Gateway.

3.4 COIN GATEWAY

O módulo Coin Gateway representa a generalização dos serviços de comunicação com criptomoedas. Toda comunicação com as criptomoedas passará por estes serviços, como fornecer o endereço da carteira, realizar transferências, listar as transações de uma carteira além de validar o status das transações.

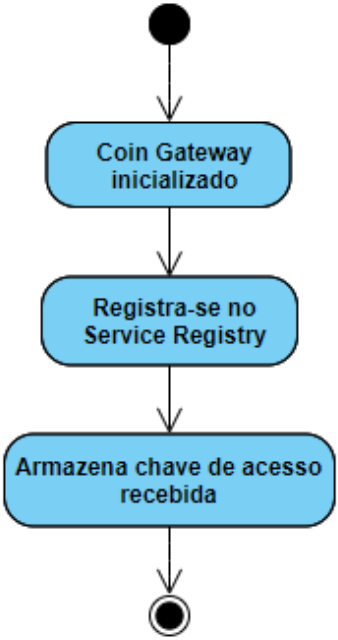

Figura 12 – Fluxo de inicialização do módulo Coin Gateway

Em sua execução, o módulo consistirá basicamente em registrar-se no Service Control e responder a requisições feitas pelo módulo de Balance Control. Seu fluxo de inicialização é representado Figura 12 e os demais fluxos estão contidos nos fluxos de Balance Control, no tópico 3.5 deste trabalho.

3.5 BALANCE CONTROL

É através do módulo Balance Control que acontecerá o controle das micro transações financeiras. Este módulo será responsável por disponibilizar as formas de pagamento para o Product Gateway, também podendo validar os preços dos produtos em suas criptomoedas e realizar transferências com os Coin Gateway.

A Figura 13, apresenta como o módulo Balance Control se comporta na sua inicialização, desde o registro no Service Registry a requisição da lista de criptomoedas.

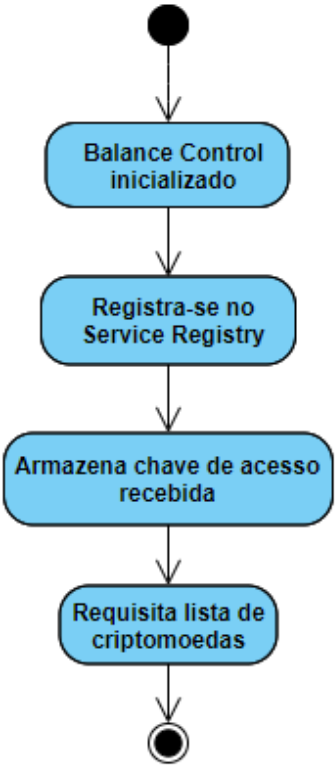

Figura 13 – Fluxo de inicialização do módulo Balance Control

Abaixo, nota-se o fluxo de listagem de métodos de pagamento no módulo Balance Control, vide Figura 14.

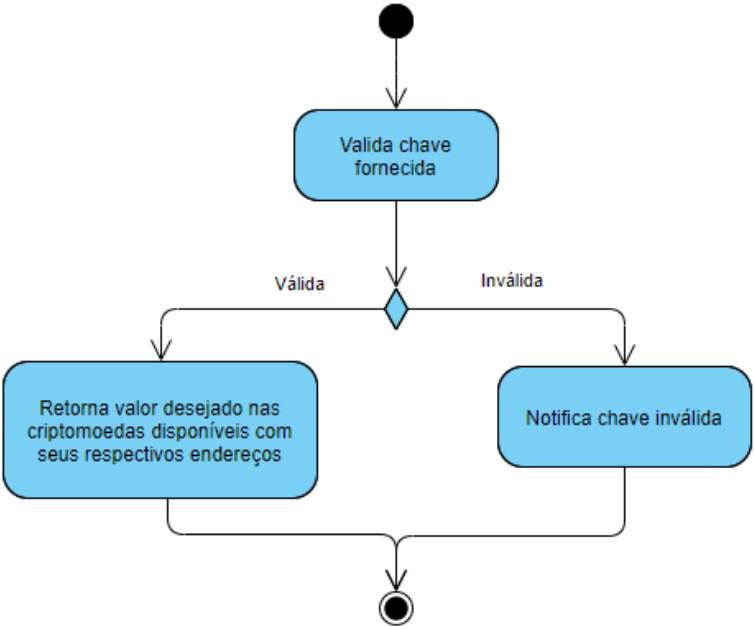

Figura 14 – Fluxo de listagem de métodos de pagamento do módulo Balance Control

Conforme ilustrado na Figura 14, uma das operações deste fluxo é a listagem de métodos de pagamento - o qual deverá fornecer uma lista nas criptomoedas disponíveis os valores praticados naquelas moedas.

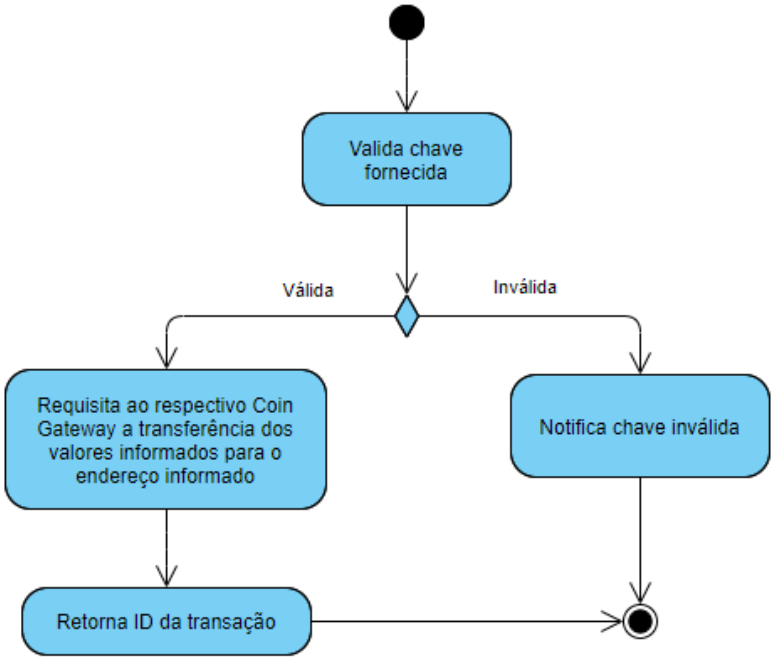

Figura 15 – Fluxo de operação de transferência do módulo Balance Control

A figura 15 representa como o módulo processará uma requisição de transferência.

3.6 TRANSACTION WATCHER

O módulo Transaction Watcher servirá para monitorar as transações registradas pelo Product Gateway. As transações que estiverem em status de confirmando pagamento, serão monitoradas periodicamente por este serviço, de forma que quando a transação for confirmada, o Product Gateway seja devidamente notificado. A Figura 16 representa o fluxo de execução periódico do módulo.

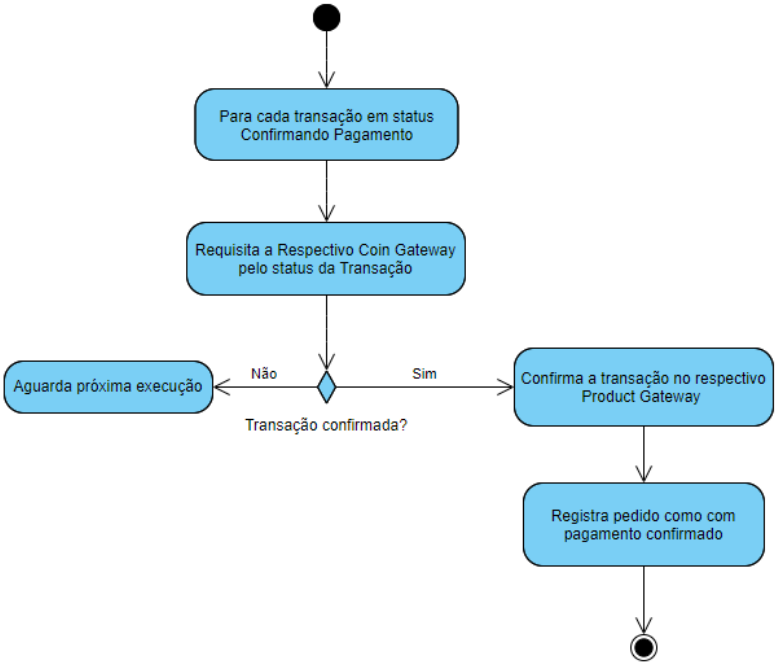

Figura 16 – Fluxo de execução de rastreamento do módulo Transaction Watcher

Assim, verifica-se que as transações que não estiverem confirmadas aguardam nova verificação, enquanto aquelas com status confirmados, geram a notificação confirmando o pagamento.

3.7 MARKETPLACE

Marketplace é o módulo aonde os produtos ficarão disponíveis para serem comprados. Através do marketplace será possível verificar seus produtos disponíveis, buscar por produtos específicos e cadastrar novos. Aplicações que desejarem efetuar a reserva/compra de algum produto, deverá buscá-lo antes no marketplace. Não possui fluxo de operação, pois são todas operações diretas, visto que ele apenas terá os produtos cadastrados pelos Product Gateways.

4 AMBIENTE EXPERIMENTAL

Nas seções abaixo, será descrito como cada módulo da arquitetura foi desenvolvido para um ambiente experimental da proposta, com descrição do funcionamento dos endpoints de cada um deles. O ambiente de testes contou com uma placa Wemos D1 para representar os dispositivos IoT, um Raspberry Pi os Coin Gateways, Balance Control, Service Registry e Product Gateway e um outro Raspberry Pi para o Marketplace. A figura 17 apresenta a disposição do ambiente experimental.

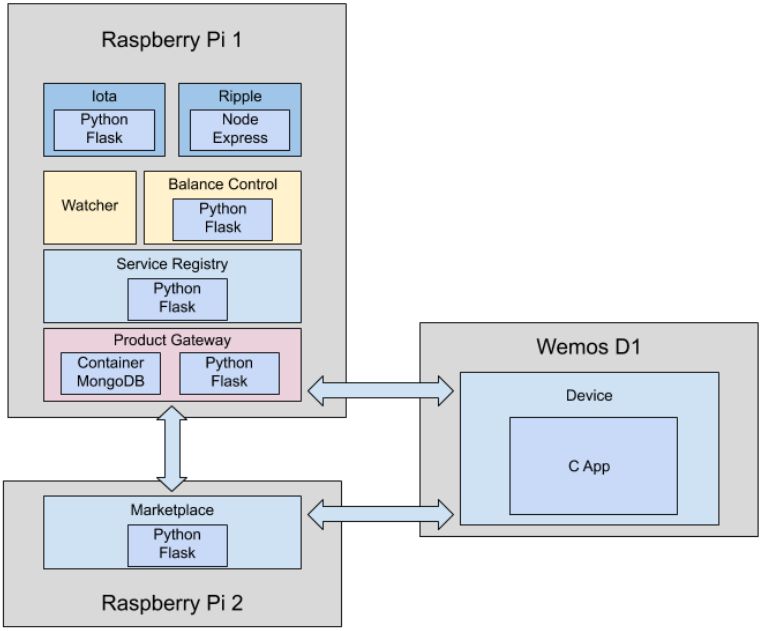

Figura 17 – Arquitetura do ambiente de experimental

4.1 SERVICE REGISTRY

O módulo Service Registry foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. As bibliotecas utilizadas foram Flask e json. Devido as características experimentais de teste, foi escolhido armazenamento em memória das

chaves de autenticação do serviço, de criptomoedas e dos dispositivos, apesar de não ser a melhor opção, como observado em (Jamshidi et al., 2018). Também devido as características experimentais, as chaves utilizadas são chaves simples hashes hexadecimais. Abaixo, as descrições dos endpoints do módulo:

- registerservice - Recebe os dados do serviço a ser registrado como parâmetro - ip, porta, tipo e parametros, e registra o serviço em uma estrutura de lista de serviços. Conforme o tipo de serviço que se registrou, são retornadas as devidas chaves de acesso: para Product Gateway, chaves de serviço e dispositivos; Balance Control, chaves de serviço e criptomoedas; Transaction Watcher, serviço e criptomoedas.

- registercrypto - Recebe os dados de Coin Gateway a ser registrado como parâmetro - ip, porta, tipo e parametros, e registra a criptomoeda em uma lista de criptomoedas. É então retornada a chave de autenticação de criptomoedas.

- registerdevice - Recebe os dados do dispositivo a ser registrado como parâmetro - ip, porta, tipo e parametros, e registra o dispositivo em uma lista de dispositivos. É então retornada a chave de autenticação de dispositivos.

- getservice - Recebe como parâmetros a chave de autenticação do serviço e o tipo de serviço que se deseja. A chave sendo válida, é retornado o serviço em formato json, caso contrário, é retornada mensagem de erro.

- listcryptos - Recebe como parâmetro a chave de autenticação de criptomoedas. A chave sendo válida, são retornadas as criptomoedas em formato json, caso contrário, é retornada mensagem de erro.

- listdevices - Recebe como parâmetro a chave de autenticação de dispositivos. A chave sendo válida, são retornados os dispositivos em formato json, caso contrário, é retornada mensagem de erro.

4.2 IOT DEVICES

O módulo de IoT Device foi desenvolvido em C. As bibliotecas utilizadas foram ArduinoJson, ESP8266WiFi, ESP8266WebServer e ESP8266HTTPClient. O módulo foi implementado de maneira que o

registro no Service Registry seja feito em sua inicialização. Além disto, também são inicializadas suas quantidade de produtos em estoque, reserva e produzidos. Abaixo, as descrições dos endpoints do módulo:

- book - Recebe como parâmetro a chave de autenticação do serviço e a quantidade. A chave sendo válida e possuindo estoque suficiente, atualiza as quantidades em estoque e reservados. Caso a quantidade seja superior a quantidade em estoque, é retornada uma mensagem notificando estoque insuficiente.

- confirm - Recebe como parâmetro a chave de autenticação do serviço e a quantidade. A chave sendo válida, atualiza as quantidades em reservados e produzidos, além de retornar uma mensagem notificando a confirmação.

4.3 PRODUCT GATEWAY

O módulo Service Registry foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. Para armazenamento de dados, foi optado pela utilização do MongoDB dentro de um container docker. As bibliotecas utilizadas foram Flask, json e pymongo. Em sua inicialização, o módulo registra-se no Service Registry, em seguida requisita os dispositivos para poder registrar no Marketplace os produtos. Abaixo, as descrições dos endpoints do módulo:

- book - Para efetuar reservas, recebe como parâmetros o tipo, a quantidade e um identificador do cliente. Com base nestas informações, requisita do IoT Device a reserva da quantidade informada. É utilizada então a chave de acesso do Service Registry para buscar o serviço do Balance Control, e por fim requisitar do Balance Control as formas de pagamento para a quantidade desejada. Registra-se então o pedido no banco de dados em status de "Aguardando Pagamento".

- pay - Recebe como parâmetros o ID do pedido, o ID da transação, a criptomoeda utilizada e o endereço da carteira. Com base nestes parâmetros, são registrados no BD as informações de pagamento no pedido, e o registro tem seu status alterado para "Confirmando Pagamento", de forma que o módulo Transaction Watcher possa rastreá-lo.

4.4 COIN GATEWAY - IOTA

O gateway para comunicação com a criptomoeda IOTA foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições e a biblioteca PyOTA para realizar a conexão com a rede Tangle. Após o desenvolvimento, para testar o módulo foi utilizada a rede de testes da Tangle para realizar transações entre nós.

Através de ferramentas fornecidas pela PyOTA, foram gerados SEEDs para fazer a troca de mensagens. Um SEED é o identificador de um usuário dentro da rede Tangle. A partir de um SEED é possível gerar vários endereços que tem o seu funcionamento parecido com uma carteira, registrando a entrada e saída de tokens IOTA. Como é possível realizar transações sem enviar tokens, foram realizadas algumas trocas de mensagens como teste entre os SEEDs gerados. Inicialmente obteve-se uma média de 1 minuto e meio para confirmação das transações.

Abaixo, as descrições dos endpoints do módulo:

- transactionstatus - Utilizada para buscar o status de uma transação, recebe como parâmetros o ID da transação, o endereço e o valor. Com estes dados, é utilizada a api IOTA para buscar as transações do endereço fornecido e caso a transação informada esteja entre as transações do endereço, utiliza a API para buscar e retornar o status da última inclusão nesta transação.
- transfer - Utilizada para realizar uma transferência, recebe como parâmetros o endereço e o valor. É então montada uma transação para ser proposta para a rede Tangle. Através da api IOTA, a proposta de transação é enviada para a rede Tangle, e o Id da transação é retornado.
- getaddress - Utilizado para gerar um endereço na criptomoeda. Utiliza diretamente a api IOTA para gerar um endereço e retorná-lo.

4.5 COIN GATEWAY - RIPPLE

O serviço de comunicação para a criptomoeda Ripple foi desenvolvido em JavaScript (Node.js). Ele utiliza o framework Express para receber e devolver as requisições. O motivo para escolher uma linguagem de programação diferente é que a Ripple oferece uma API de desenvolvimento com bibliotecas já prontas para Node, a ripple-lib.

Para validar a utilização do gateway desenvolvido foi utilizada uma rede de desenvolvimento da Ripple. Diferente da IOTA, para se conectar a essa rede foi necessário conseguir credenciais de acesso. Essas são geradas de forma automática pela API. Outra diferença, é que cada nova conta desta rede de desenvolvimento inicia com 10 mil tokens XRP, viabilizando uma simulação real de transferência de recursos.

Diferente da Tangle, não parece ser necessária uma funcionalidade de verificação de status de transação, uma vez que ao que tudo indica esse status é definitivo após o termino de uma chamada na rede XRP e já retorna como resposta nas chamadas da API mas, como o Coin Gateway requer a existência deste tipo de requisição, foi implementado um endpoint para verificar o status. As transações realizadas como teste obtiveram uma média de duração de aproximadamente 4 segundos. Abaixo, as descrições dos endpoints do módulo:

- transactionstatus - Utilizada para buscar o status de uma transação, recebe como parâmetros o ID da transação, o endereço e o valor. Com estes dados, é utilizada a api Ripple para validar se a transação desejada está já está na última versão do livro-razão e retornar seu resultado, se foi para o endereço e possui status de sucesso.

- transfer - Utilizada para realizar uma transferência, recebe como parâmetros o endereço e o valor. A transação é preparada com a api Ripple, com sua devida assinatura e configuração de expiração através da versão do livro-razão. Com a transação preparada, ela é submetida para a rede, e retorna o Id da transação.

- getaddress - Utilizado para gerar um endereço na criptomoeda. No contexto do ambiente experimental, devido a necessidade de credencias de acesso, foi utilizado um endereço fixo.

4.6 BALANCE CONTROL

O módulo Balance Control foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. Em sua inicialização, após se registrar no Service Registry, é buscada a listagem de criptomoedas disponíveis nele através do endpoint listcryptos, armazenando os dados de serviço delas em memória. Abaixo, as descrições dos endpoints do módulo:

- listpaymentoptions - Recebe como parâmetro a quantidade desejada e retorna após uma busca nas criptomoedas em memória,

seus endereços e preços.

- transfer - Registra uma transfência, recebe como parâemtro o endereço, o valor e a criptomoeda desejada. Requisita então para o respectivo Coin Gateway a realização da transferência, retornando o ID da transação.

4.7 TRANSACTION WATCHER

Módulo para rastreamento das transações, foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. As bibliotecas utilizadas foram Flask, json, threading e time. Para armazenamento de dados, foi optado pela utilização do banco já utilizado pelo Product Gateway, dada a natureza experimental do ambiente e ambiente restrito. Apesar desta reutilização, o módulo não ficará indisponível caso o serviço Product Gateway esteja fora do ar, apenas se comportará como se não houvessem transações sendo rastreadas no momento. Em sua inicialização, após se registrar no Service Registry, é buscada a listagem de criptomoedas disponíveis nele através do endpoint listcryptos, armazenando os dados de serviço delas em memória.

Em sua inicialização, também é iniciada uma thread de rastrear as transações. O rastreio funciona buscando no BD todas os pedidos com status "Confirmando Pagamento"e para cada pedido retornado, é validado o status de sua transação através do módulo Coin Gateway da criptomoeda utilizada no pedido. Caso o status da transação seja de que ela foi concluída, o pedido tem seu status atualizado no BD para "Confirmado"e o endpoint de confirmtransaction do Product Gateway é utilizado. Esta thread dispara o processo de rastreio a cada 1 segundo.

Este módulo não possui endpoints por ter seu funcionamento automático pela thread de sua inicialização.

4.8 MARKETPLACE

O módulo Marketplace é a porta de acesso aos produtos da arquitetura. Foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. As bibliotecas utilizadas foram Flask e json. Os produtos nele registrados são armazenados em memória. Abaixo, as descrições dos endpoints do módulo:

- register - Utilizado para registrar produtos, recebe como parâme-

tro os dados do serviço de Product Gateway.

- list - Sem parâmetros, tras a listagem de todos os produtos registrados.
- search - Recebe como parâmetro o tipo do produto e retorna os produtos registrados que forem daquele tipo.

5 RESULTADOS OBTIDOS

Para análise do ambiente experimental, concebeu-se uma forma de testá-lo, obtendo dados e os analisando. O teste foi efetuar 100 transações com cada uma das criptomoedas que tiveram um Coin Gateway implementado ou seja, 100 transações com IOTA e 100 transações com Ripple. Depois sucedeu a análise do tempo dispendido nas transações.

Para executar o teste, utilizou-se um pequeno projeto em Python, que representou uma instância da arquitetura que fez uso somente de dpois módulos, Balance Control e Coin Gateway. Essa medida foi adotada para provar a diversidade possível com a arquitetura, pois uma instância dela não necessita de todos os módulso para operar. Esta intância comunicou-se com outra, através dos Marketplace e do Balance Control para realização de operações. Houveram duas execuções, uma para as transações em IOTA e outra para as transações em Ripple. Abaixo, quais operações eram feitas pelo script em cada uma das 100 repetições:

1. Requisita ao marketplace através da operação *search* por um produto específico.

2. Ordena os produtos retornados por preço e seleciona o de menor preço.

3. Requisita a reserva do produto de menor preço através da operação. *book* de seu Product Gateway, recebendo como retorno os dados do pedido.

4. Utiliza o endereço de IOTA contido nos dados do pedido e requisita ao Balance Control o pagamento do preço do pedido, através da operação *transfer*.

5. Com os dados da transação retornados do Balance Control, requisita ao Product Gateway a confirmação do pagamento através da operação *pay*.

A Figura 18 representa as trocas de comunicações entre as instancias da arquitetura no momento em que é feita a operação de buscar de produtos. Para isto, TestScript executa a operação *search* do módulo marketplace, que retornar os produtos, com o endereço dos seus Balance Control, além de demais informações dos produtos.

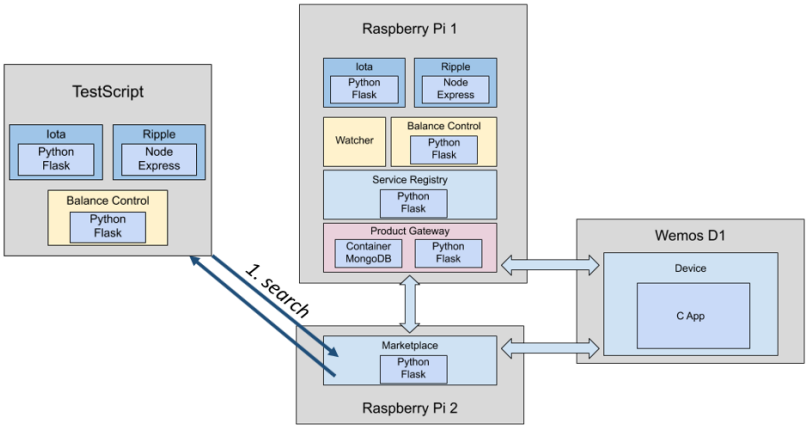

Figura 18 – Troca de requisições na operação *search*

A Figura 19 representa as trocas de comunicações entre as instancias da arquitetura no momento em que é feita a operação de reserva de produto. Para isto, TestScript executa a operação 1.*book* do módulo Balance Control do produto desejado. O pedido de reserva é então feito ao módulo do dispositivo IoT, com sua operação 2.*book*, que retorna o sucesso ou não do pedido de reserva. No caso de sucesso, é executada a operação 3.*listpaymentoptions* do Balance Control, que por sua vez requisita de cada Coin Gateway os endereços de suas carteira através das operações 4. e 5. *getaddress*, devolvendo ao Balance Controle, que por fim encerra as requisições retornando ao TestScript o identificador do pedido e os endereços das carteiras

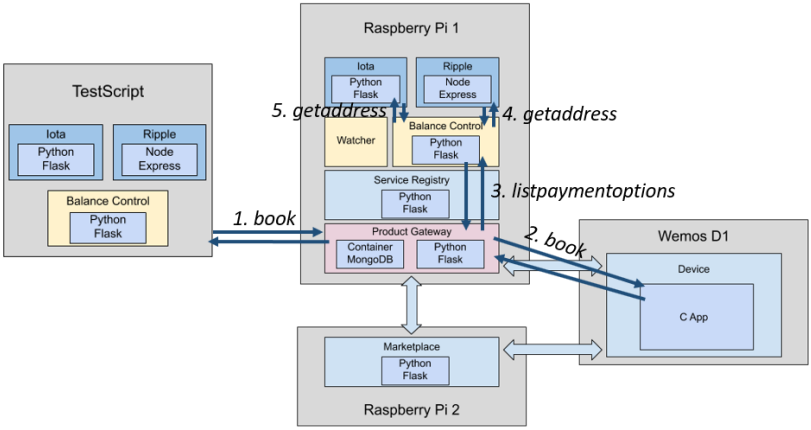

Figura 19 – Troca de requisições na operação *book*

A Figura 20 representa as trocas de comunicações dentro da arquitetura do TestScript no momento em que é feita a operação transferência para a carteira recebida com seu pedido. Para isto, TestScript executa a operação 1.*transfer* do seu próprio módulo Balance Control, que por sua vez executa a operação 2.*transfer* a criptomoeda da carteira informada

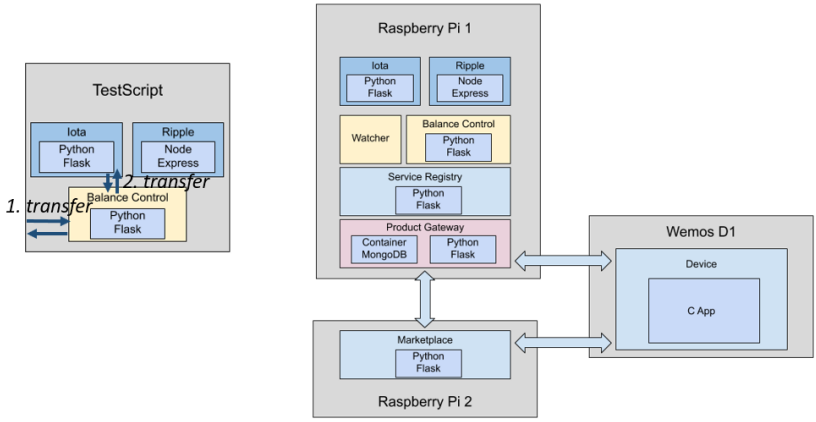

Figura 20 – Troca de requisições na operação *transfer*

A Figura 21 representa as trocas de comunicações entre as instancias da arquitetura no momento em que é feita a operação de infor-

mar o pagamento. Para isto, TestScript executa a operação 1.*pay* do módulo Balance Controle, que registra a mudança de status do pedido informado e o id da transação.

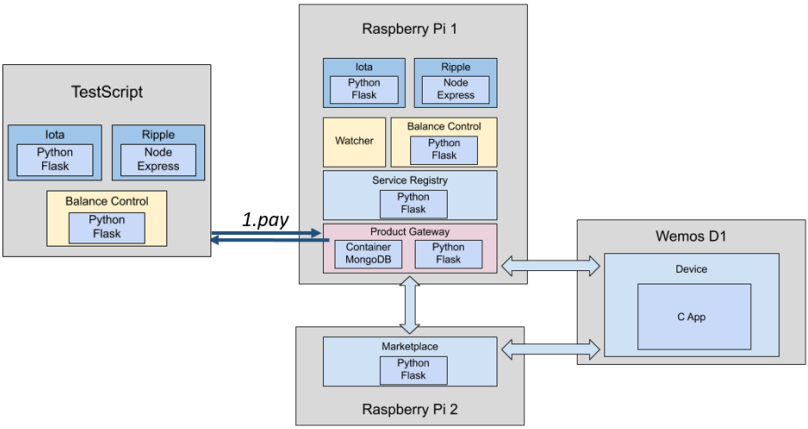

Figura 21 – Troca de requisições na operação *pay*

A Figura 22 representa as trocas de comunicações quando o módulo Transaction Watcher valiad os status das transações que tiveram seu pagamento informado. Para isto, o módulo executa a operação 1.*transactionstatus* do módulo Coin Gateway das tranações que tiveram pagamento informado. Com o status confirmado pelo coin gateway, o transaction watcher então utiliza a operação 2.*confirmtransaction* do módulo Product Gateway, que por sua vez executa a operação 3.*confirm* do dispositivo IoT que aguardava a transação confirmada.

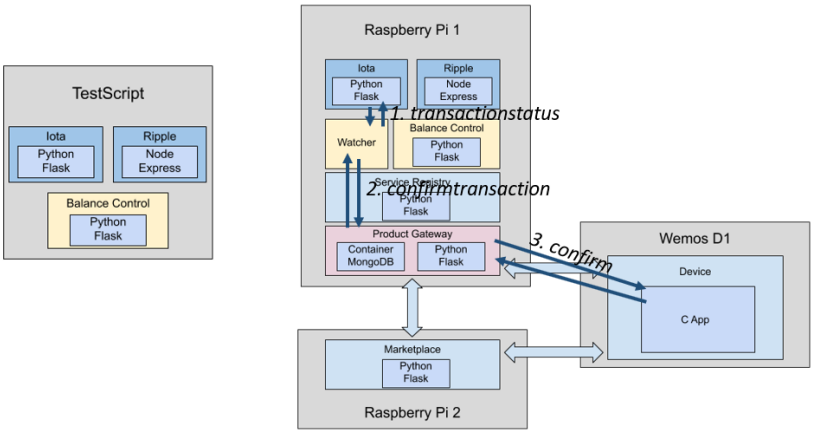

Figura 22 – Troca de requisições na operação *transactionstatus*

Após a execução, os dados dos tempos das transações foram extraídos, para geração dos gráficos que serão apresentados a seguir. Os termos utilizados no gráfico, são o do tempo de uma transação, que consiste na soma do tempo levado para fazer um pagamento com o tempo levado para confirmar o pagamento.

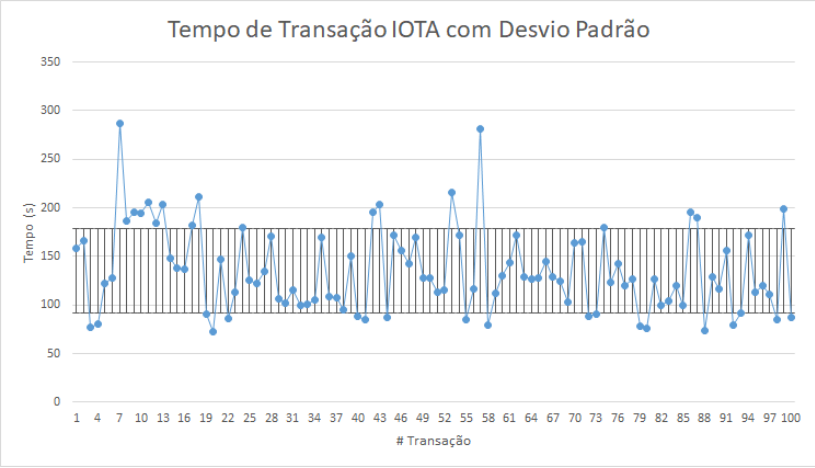

Figura 23 – Dispersão das transações IOTA

Para IOTA, o tempo total de execução foi de 49 minutos e 50 segundos. A duração média de transação foi de aproximadamente 135

segundos sendo 13% do tempo gasto com o pagamento e 87% com a confirmação do pagamento. Analisando a dispersão dos dados, em conjunto com seu desvio padrão na figura 23, 36% das transações ficam além do desvio, indicando que os tempos de transação de IOTA têm considerável variação.

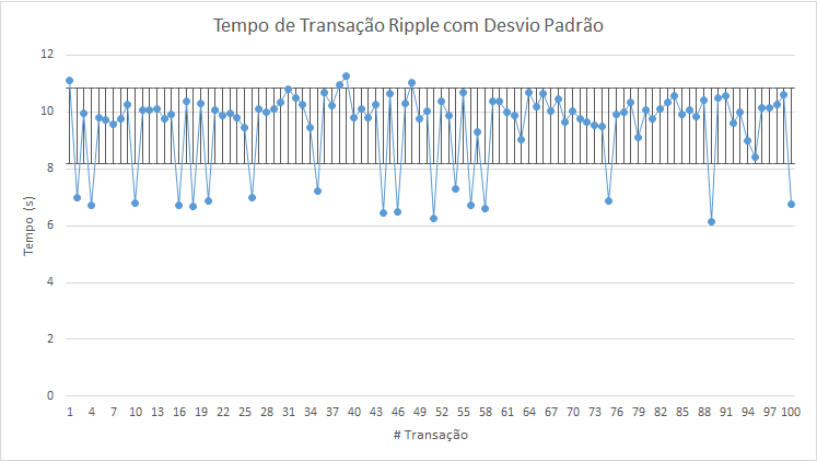

Figura 24 – Dispersão das transações Ripple

Ao realizar a mesma análise sobre as transações de Ripple, que possuem o tempo de execução de 17 minutos e 39 segundos, tendo um tempo médio por transação de 9,5 segundos e um desvio padrão de 1,33 segundos, com 21% das transações ficando foram do intervalo compreendido pelo desvio padrão sobre a média, ilustrado na figura 24. O tempo de cada transação em Ripple, ficou com uma divisão de 33,38% em pagamento e 66,62% para pagamento.

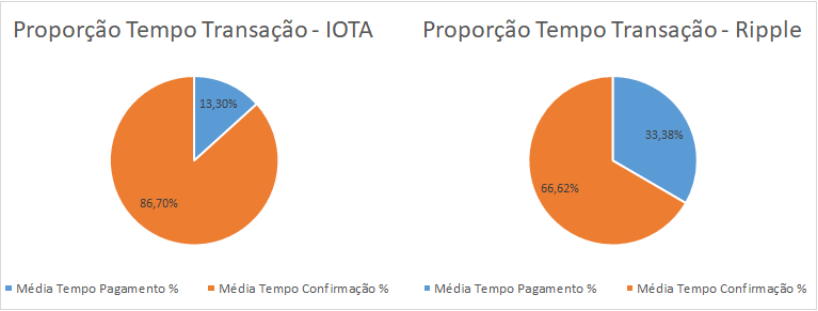

Figura 25 – Proporção do tempo de transação de cada criptomoeda

Na figura 25 é possível comparar a diferença na proporção de tempo utilizada por cada criptomoeda nos esforço para realização de pagamento e para confirmação da transação. IOTA gasta próximo de um décimo de seu tempo com pagamento, quanto Ripple chega a um terço.

6 CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho teve como proposta, conceber uma arquitetura para controle de micro transações financeiras em um contexto de EoT, baseada em microsserviços. A fim de alcançar este objetivo, foram pesquisados e analisados conceitos, propostas e utilizações de IoT, Blockchains, Microsserviços e EoT, ora integrados, ora isolados. A arquitetura proposta possuía gateways de comunicação para criptomoedas, dispositivos IoT, e monitoramento de transações destes dispositivos, além de um módulo para autenticação dos participantes da arquitetura. A flexibilidade necessária em um ambiente aonde haja variedade de tecnologias envolvidas, que estão em constante evolução, como IoT e blockchains, levou a utilização de microsserviços.

Para validação da proposta, foi desenvolvido um ambiente experimental. Nele, o gateway de comunicação com criptomoedas (Coin Gateway), realizava as consultas e transferências com criptomoedas, transferindo fundos e consultando status de transações submetidas através dele. Já o gateway de comunicação com os dispositivos IoT, chamado de Product Gateway, é o serviço responsável por realizar as comunicações que os dispositivos necessitassem, consultando os produtos que o dispositivo tinha, formas de pagamento, efetuando reservas de itens e confirmando transações. O Balance Control realiza o controle das micro transações, sendo responsável por informar as formas de pagamento para o Product Gateway e validando os preços nas criptomoedas que serão transferidos para os Coin Gateway. O módulo de monitormanento de transações (Transaction Watcher), é incumbido de constantemente validar as transações passadas pela arquitetura, notificando aos Product Gateway quando confirmadas. O módulo de autenticação - Service Registry, valida os elementos da rede, encaminhando para cada um as chaves de acesso aos serviços que necessitassem.

O ambiente experimental, consistiu de uma placa Wemos D1 para representar os dispositivos IoT, um Raspberry Pi para os Coin Gateways, Balance Control, Service Registry e Product Gateway e um outro Raspberry Pi para o Marketplace. Foram utilizadas as linguagens Python, JavaScript e C, a fim de demonstrar a flexibilidade da arquitetura. A comunicação era realizada através da internet, com requisições utilizando o framewerk Flask. As criptomoedas selecionadas para os testes foram IOTA e Ripple, aonde para cada uma, foram realizadas 100 transações e analisados seus tempos de execução.

Embora os tempos de transações com a criptomoeda IOTA te-

nham tido uma dispersão grande, os resultados apresentados foram satisfatórios, levando a crer que a arquitetura é apta a atender a necessidade da proposta em um contexto real.

Como trabalhos futuros, foi identificada a necessidade de um teste em um ambiente experimental mais complexo, com formas de comunicação distintas e mais elementos como outras criptomoedas e diversificados tipos de dispositivos IoT. Além de uma melhora que torne possível analisar o tempo dispendido em comunicação entre os módulos para análise da performance da arquitetura, considerando que o teste feito não diferenciou o tempo das transações neste aspecto. A necessidade de um dispositivo precisar efetuar transações de compra através de outro marketplace dando inicio a um novo ciclo de interações, também parece uma boa maneira de validação. Também se faz necessária uma abordagem mais segura na forma da comunicação entre os serviços.

REFERÊNCIAS

AGOSTINHO, B. M. et al. Smart comm: A smart home middleware supporting information exchange. *44th Annual Conference of the IEEE Industrial Electronics Society, IECON.*, Oct 2018. ISSN 2577-1647.

AL-FUQAHA, A. et al. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communication Surveys & Tutorials.*, v. 17, n. 4, 2015.

Al-Sarawi, S. et al. Internet of things (iot) communication protocols: Review. In: *2017 8th International Conference on Information Technology (ICIT)*. [S.l.: s.n.], 2017. p. 685–690.

ARMKNECHT, F. et al. Ripple: Overview and outlook. In: SPRINGER. *International Conference on Trust and Trustworthy Computing*. [S.l.], 2015. p. 163–180.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 54, n. 15, p. 2787–2805, out. 2010. ISSN 1389-1286. <http://dx.doi.org/10.1016/j.comnet.2010.05.010>.

BACK, R. P.

Análise Comparativa de Técnicas de Integração entre Microsserviços — Universidade Federal de Santa Catarina, Florianopolis, SC, Brasil, 2016.

BARROZO, L. M.

Descoberta semântica de microservices em contêineres — Universidade Federal de Santa Catarina, Florianopolis, SC, Brasil, 2016.

Bröring, A. et al. Enabling iot ecosystems through platform interoperability. *IEEE Software*, v. 34, n. 1, p. 54–61, Jan 2017.

Butzin, B.; Golatowski, F.; Timmermann, D. Microservices approach for the internet of things. p. 1–6, Sep. 2016.

Centenaro, M. et al. Long-range communications in unlicensed bands: the rising stars in the iot and smart city scenarios. *IEEE Wireless Communications*, v. 23, n. 5, p. 60–67, October 2016.

CHAIN, I. *IoT Chain - A high-security lite IoT OS - White Paper*. 2018. <https://iotchain.io/whitepaper/ITCWHITEPAPER.pdf>. Acessado em 12/10/2019.

CHAIN, R. *Ruff IoT Blockchain Whitepaper*. 2017. <https://github.com/RuffNotes/RuffChain/blob/master/WhitePaper.md>. Acessado em 12/10/2019.

CHASE, B.; MACBROUGH, E. Analysis of the xrp ledger consensus protocol. *ArXiv*, abs/1802.07242, 2018.

CHRISTIDIS, K.; DEVETSIKIOTIS, M. Blockchains and smart contracts for the internet of things. *IEEE Access*, v. 4, p. 2292–2303, 2016. ISSN 2169-3536.

DAI, H.-N.; ZHENG, Z.; ZHANG, Y. Blockchain for internet of things: A survey. *ArXiv*, abs/1906.00245, 2019.

FAMILIAR, B. *Microservices, IoT, and Azure: Leveraging DevOps and Microservice Architecture to Deliver SaaS Solutions*. Berkeley, CA: Apress, 2015. 133-163 p. ISBN 978-1-4842-1275-2.

FARELL, R. *An analysis of the cryptocurrency industry*. 2015. <https://repository.upenn.edu/>. Acessado em 25/06/2019.

FERNáNDEZ-CARAMéS, T. M.; FRAGA-LAMAS, P. A review on the use of blockchain for the internet of things. *IEEE Access*, v. 6, p. 32979–33001, 2018. ISSN 2169-3536.

FOUNDATION, I. *The Industry Marketplace*. 2019. <https://industry.iota.org/industry$_m$*arketplace*>.*Acessadoem*14/10/2019.

Granjal, J.; Monteiro, E.; Sá Silva, J. Security for the internet of things: A survey of existing protocols and open research issues. *IEEE Communications Surveys Tutorials*, v. 17, n. 3, p. 1294–1312, thirdquarter 2015. ISSN 1553-877X.

Harris III, A. F. et al. Bluetooth low energy in dense iot environments. *IEEE Communications Magazine*, v. 54, n. 12, p. 30–36, December 2016. ISSN 0163-6804.

HUCKLE, S. et al. Internet of things, blockchain and shared economy applications. *Procedia computer science*, Elsevier, v. 98, p. 461–466, 2016.

Jamshidi, P. et al. Microservices: The journey so far and challenges ahead. *IEEE Software*, v. 35, n. 3, p. 24–35, May 2018. ISSN 0740-7459.

KRISHNAMACHARI, B. et al. I3: An iot marketplace for smart communities. In: ACM. *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. [S.l.], 2018. p. 498–499.

Krylovskiy, A.; Jahn, M.; Patti, E. Designing a smart city internet of things platform with microservice architecture. p. 25–30, Aug 2015.

LEWIS, J.; FOWLER, M. Microservices. *martinfowler. com*, 2014.

Liu, Y. et al. LightChain: A lightweight blockchain system for industrial internet of things. *IEEE Transactions on Industrial Informatics*, v. 15, n. 6, p. 3571–3581, June 2019.

LOUGEE, R.; PURESWARAN, V. *The Economy of Things - Extracting new value from the Internet of Things*. 2015. <https://www.ibm.com/downloads/cas/AVRE308E>. Acessado em 06/06/2019.

LUCIO, J. P. D.

ANÁLISE COMPARATIVA ENTRE ARQUITETURA MONOLÍTICA E DE MICROSSERVIÇOS — Universidade Federal de Santa Catarina, Florianopolis, SC, Brasil, 2017.

Manimuthu, A. et al. A literature review on bitcoin: Transformation of crypto currency into a global phenomenon. *IEEE Engineering Management Review*, v. 47, n. 1, p. 28–35, Firstquarter 2019. ISSN 0360-8581.

Mišura, K.; Žagar, M. Data marketplace for internet of things. In: *2016 International Conference on Smart Systems and Technologies (SST)*. [S.l.: s.n.], 2016. p. 255–260.

MUKHOPADHYAY, U. et al. A brief survey of cryptocurrency systems. p. 745–752, Dec 2016.

NAKAMOTO, S. et al. Bitcoin: A peer-to-peer electronic cash system. 2008.

NEWMAN, S. *Building microservices: designing fine-grained systems*. [S.l.]: "O'Reilly Media, Inc.", 2015.

NGU, A. H. et al. Iot middleware: A survey on issues and enabling technologies. *IEEE Internet of Things Journal.*, v. 4, n. 1, February 2017.

POPOV, S. *The Tangle.* 2018. <https://iota.org/IOTA$_W$*hitepaper.pdf*>.*Acessadoem*26/06/2019.

ROTTA, G.; DANTAS, M. A. R. Um estudo sobre protocolos de comunicação para ambientes de internet das coisas. *Escola Regional de Alto Desempenho.*, 2017.

Sanchez-Gomez, J.; Sanchez-Iborra, R.; Skarmeta, A. Transmission technologies comparison for iot communications in smart-cities. In: *GLOBECOM 2017 - 2017 IEEE Global Communications Conference.* [S.l.: s.n.], 2017. p. 1–6.

SCHWAB, K. *The Fourth Industrial Revolution.* Crown Publishing Group, 2017. ISBN 9781524758875. <https://books.google.com.br/books?id=ST_FDAAAQBAJ>.

SCHWARTZ, D. et al. The ripple protocol consensus algorithm. *Ripple Labs Inc White Paper*, v. 5, p. 8, 2014.

Sheng, Z. et al. A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities. *IEEE Wireless Communications*, v. 20, n. 6, p. 91–98, December 2013. ISSN 1536-1284.

SIDDIQUI, F. et al. Secure and lightweight communication in heterogeneous iot environments. *Internet of Things*, p. 100093, 2019. ISSN 2542-6605. <http://www.sciencedirect.com/science/article/pii/S2542660519301921>.

SKWAREK, V. Blockchains as security-enabler for industrial iot-applications. *Asia Pacific Journal of Innovation and Entrepreneurship*, Emerald Publishing Limited, v. 11, n. 3, p. 301–311, 2017.

TEAM, T. I. *IoTeX - A Decentralized Network for Internet of ThingsPowered by a Privacy-Centric Blockchain.* 2018. <https://v1.iotex.io/white-paper>. Acessado em 12/10/2019.

TENNANT, L. *Improving the Anonymity of the IOTA Cryptocurrency.* 2017. <https://assets.ctfassets.net/r1dr6vzfxhev/6StLLAy9b26eyUG8SGQqeu/e30c20f91e77

ZHENG, Z. et al. Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*, Inderscience Publishers (IEL), v. 14, n. 4, p. 352–375, 2018.

APÊNDICE A – Artigo SBC

Uma arquitetura para micro transações financeiras em um contexto de economia das coisas

Alexandre P. Back[1]

[1]Departamento de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)
Campus Reitor João David Ferreira Lima, s/n - Trindade, Florianópolis - SC, 88040-900 - Brazil

alexandre.pb@grad.ufsc.br

Abstract. *IoT has revolutionized the way we live. Its ability to capture, share and use information for smart decision-making qualifies for a great potential for change linked to this technology. The alliance of this technology with Blockchain, developed for use with BitCoin cryptocurrency, gave birth to the Economy of Things (EoT). EoT is the delivery of digital services provided by IoT devices on Blockchains-based marketplaces. In this paper we will present a proposal for the control of micro financial transactions in an EoT context, through a microservices architecture, with presentation of results obtained in an experimental environment developed to validate the proposal.*

Resumo. *A IoT revolucionou a forma como vivemos. Sua capacidade de captar, compartilhar e utilizar informações para tomadas de decisão inteligentes, qualificam um grande potencial de mudança atrelado a esta tecnologia. A aliança desta tecnologia com Blockchain, desenvolvida para uso com a criptomoeda BitCoin, nasce a Economia das Coisas (EoT). A EoT é a disponibilização de serviços digitais fornecidos por dispositivos IoT em marketplaces baseados em Blockchains. Neste artigo será apresentada uma proposta para controle de micro transações financeiras num contexto de EoT, através de uma arquitetura de microsserviços, com apresentação de resultados obtidos em ambiente experimental desenvolvido para validação da proposta.*

1. Introdução

A Internet das Coisas (IoT) vem revolucionando a forma como vivemos e trabalhamos. A forma como estes dispositivos captam informações do ambiente com seus sensores para compartilharem entre si e com humanos possibilitando tomadas de decisão inteligentes, que visam beneficiar o ecossistema como um todo, é o que qualifica seu grande potencial de mudanças. O crescimento na quantidade de dispositivos IoT é notável nas atividades do dia-a-dia. Há poucos anos não se imaginava o quão populares se tornariam, nas mais variadas formas como pulseiras e relógios inteligentes que monitoram o sono ou nossas atividades físicas diárias. A facilidade ao se lidar com assistentes inteligentes, que nos informam a previsão do tempo com lembretes em nossas

agendas com um mero comando de "bom-dia", até ligam e desligam máquinas de café, lâmpadas e diversificados dispositivos smarts. A quantidade de dispositivos conectados Machine-to-Machine (M2M) deve ultrapassar o número de pessoas utilizando serviços de inscrição como telefones, computadores e tablets até 2020 e até 2024 a indústria IoT no geral deve movimentar até 4.3 trilhões de dólares.

A IoT demanda soluções leves e escaláveis, com garantias de segurança e privacidade. A tecnologia blockchain, tem o potencial de atender a estas demandas, como resultado de sua natureza distribuída, segura e privada. Essencialmente, blockchain é o banco de dados distribuído de registros ou um livro-razão público de todas as transações ou eventos digitais que foram executados e compartilhados entre seus participantes. Cada transação do livro-razão é verificada por um consenso da maioria dos participantes do sistema e, uma vez aceitas, não podem ser apagadas. Esta tecnologia estabelece um sistema de criação de consenso distribuído no mundo digital online - o qual permite que entidades participantes tenham certeza de que um evento digital aconteceu ao criar um registro irrefutável em um livro-razão público.

O blockchain abriu as portas para o desenvolvimento de uma economia digital democrática, aberta e escalável, a partir de uma economia centralizada. Desde sua criação, com a criptomoeda Bitcoin, outras formas de dinheiro eletrônico com estruturas semelhantes surgiram. Ao mesmo tempo, diferentes aplicativos usando blockchain foram desenvolvidos ao longo dos anos para implementar outros cenários além das criptomoedas: novos conceitos, como contratos inteligentes e propriedades inteligentes, entraram em cena.

A economia das coisas (EoT) é a monetização das coisas, a disponibilização de ativos digitais fornecidos por dispositivos IoT em marketplaces. A integração entre blockchains e IoT pode trazer inúmeras vantagens a este movimento. Com o recente lançamento feito pela IOTA Foundation do Industry Marketplace - um marketplace descentralizado voltado para a indústria de IoT (a indústria 4.0), este conceito ganha visibilidade, evidenciando o potencial por trás desta união. Contudo, isto não implica em inexistência de desafios a serem ultrapassados, pelo contrário. A interoperabilidade e heterogeneidade em IoT, suas restrições computacionais e energéticas, estão entre estes desafios.

Diante estes desafios, é proposta - neste trabalho - uma arquitetura de microsserviços para superar a heterogeneidade entre dispositivos e protocolos, num contexto de microtransações financeiras de EoT. A flexibilidade de microsserviços em comparação a abordagens mais monolíticas torna este tipo de arquitetura atraente para solucionar os problemas originados neste contexto.

2. Fundamentação Teórica

Microsserviço é o termo utilizado para definir aplicações projetadas como um conjunto de serviços autônomos [Barrozo 2016]. De acordo com [Butzin 2016], a arquitetura de microsserviços não foi inventada, mas surgiu de experiências e boas práticas essenciais elencadas por usuários de SOA [Back 2016].

Acima de tudo, esta arquitetura se tornou uma proposta para suportar modelos de negócios altamente escaláveis e mutáveis, além de seu elevado grau de

manutenibilidade, visto que serviços podem ser escritos em diferentes linguagens de programação, proporcionando grande heterogeneidade.

A Internet das Coisas (IoT) consiste em objetos conectados, que detectam e coletam dados de seus arredores, que então são utilizados na realização de tarefas automatizadas visando ajudar as pessoas.

Estima-se que quantidade de dispositivos inteligentes no mundo ultrapasse os 250 bilhões até o final da década [Familiar 2015]. Esta constante taxa de crescimento de dispositivos conectados, capazes de interagirem e comunicarem entre si, onde tudo está conectado é conhecida como IoT.

Conceitualmente, a IoT permite que estes dispositivos se comuniquem, compartilhem informações, coordenem decisões e desempenhem atividades [Al-Fuqaha 2015]. Segundo [Aztori 2010], a IoT consiste na presença ubíqua destes dispositivos ou coisas a nossa volta, nas suas mais variadas formas, de sensores de temperatura até smartphones e carros.

Blockchain é uma estrutura de dados distribuída, replicada e compartilhada entre membros de uma rede P2P, foi inicialmente desenvolvida para uso com a criptomoeda BitCoin - visando acabar com o problema do gasto duplo, mas suas aplicações vão além disto [Fernández-Caramés 2018].

O seu funcionamento é basicamente uma cadeia de blocos, aonde cada um deles possui uma lista de transações e o hash do bloco anterior. Cada bloco é um nó de uma rede que possui um par de chaves pública/privada, utilizadas na leitura e validação de transações. Quando um nó desta rede recebe uma transação ela é validada; havendo êxito a transação é assinada e retransmitida aos blocos adjacentes e assim sucessivamente, caso contrário ela é descartada [Christidis 2016].

Criptomoedas são um esquema de troca digital P2P que gera e distribui valores monetários usando criptografia [Farell 2015]. Este processo necessita um verificação distribuída de transações, sem uma autoridade central. As verificações da transações confirmam seus valores e se o pagador detém a moeda que ele deseja gastar, garantindo que unidades não sejam gastas em duplicidade. Este método de verificação é chamado de mineração [Mukhopadhyay 2016], e diferentes criptomoedas utilizam diferentes métodos, conforme suas necessidades.

A Economia das Coisas é um conceito recente que descreve a monetização das coisas. A disponibilização de produtos ou serviços digitais fornecidos por dispositivos IoT em marketplaces, é o que constitui a EoT.

Em [Huckle 2016], através da identificação em dados reais, são abordadas as vantagens da associação entre IoT e Blockchain para beneficiar aplicações de economia compartilhada - que, em suma, são são aplicações para monetizar coisas, como um quarto de sua casa(airbnb) ou seu carro(Uber).

Recentemente, a IOTA Foundation lançou o Industry Marketplace [Foundation 2019], que nada mais é do que um marketplace voltado diretamente para a indústria 4.0, oferecendo a rede Tangle como base para comunicação e infraestrutura computacional, capaz de suportar todos os atores envolvidos na cadeia de valores do segmento.

3. Arquitetura Proposta

Neste trabalho, é proposta uma arquitetura para o controle de micro transações financeiras em um contexto de EoT. Esta arquitetura, deve ser adequada a comportar quaisquer criptomoedas desejadas, com uma abstração da comunicação entre dispositivos fornecedores de serviços ou produtos, e os consumidores, fazendo então o controle das microtransações financeiras.

Uma situação de uso da arquitetura, seria com um carro inteligente. Este carro teria sensores que identificariam a quantidade de combustível remanescente no carro e a necessidade de abastecimento, pois também saberia que seu dono faria um trajeto até sua casa ao fim da jornada de trabalho. Este carro com a informação do trajeto e distância que o carro conseguiria percorrer com o combustível restante, verificaria através da arquitetura, postos de combustíveis no trajeto, selecionando com melhor custo e menor alteração no trajeto, para sugerir uma rota ao motorista. Ao chegar ao posto, através de uma carteira de criptomoeda já vinculada ao carro, o combustível poderia ser pago com ela.

A Figura 1 foi elaborada para representar como estes atores estariam na arquitetura proposta, havendo os carros inteligentes com seus sensores, comunicações com criptomoedas e outros dispositivos fornecendo seus produtos, como postos de combustível.

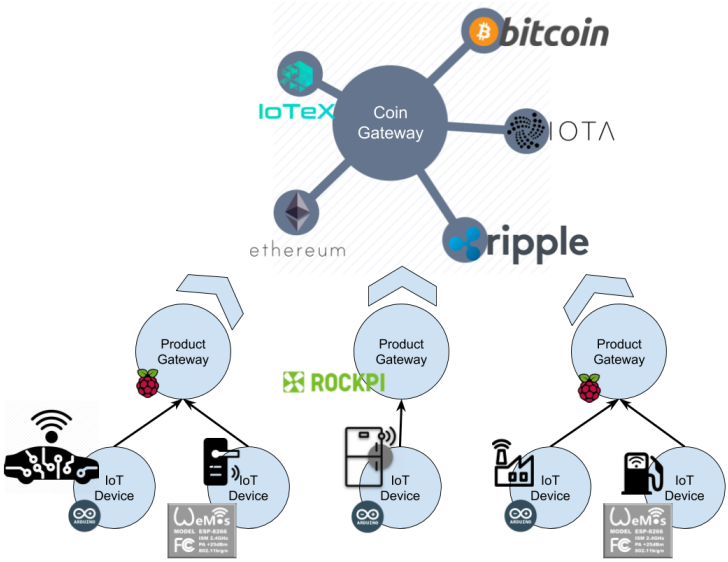

Figure 1. Exemplificando arquitetura proposta

Após análise dos benefícios do uso de microsserviços, optou-se por sua utilização para enfrentar os desafios de interoperabilidade e heterogeneidade em ambientes IoT, além de sua flexibilidade no que diz respeito à possibilidade de melhorias

contínuas com o incremento ou até substituição de microsserviços sem um impacto sistêmico. A Figura 2 apresenta a organização dos microsserviços da arquitetura.

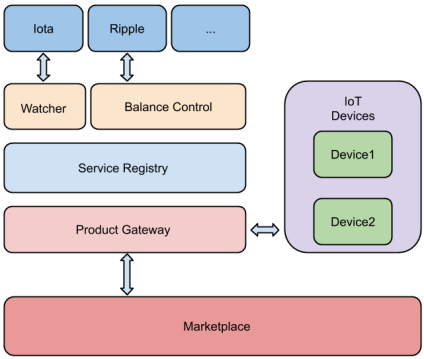

Figure 2. Organização arquitetura

Service Registry é o módulo responsável por manter o registro dos serviços do sistema. Nele, serão armazenados os endereços e tipos dos demais serviços que integram o sistema. A base de seu funcionamento é: os serviços se cadastram nele e requisitam os dados de outros serviços nele cadastrados para poderem se comunicar.

IoT Devices é o módulo de Dispositivos IoT, aonde serão configurados os produtos fornecidos por ele. Estes produtos serão utilizados pelo Product Gateway, quando este comunicar-se com o dispositivo para realizar reservas e confirmação de pedido.

Product Gateway é uma abstração dos produtos fornecidos pelos dispositivos IoT. Neste módulo, estarão disponíveis as informações destes, além de métodos para interação com os produtos por ele fornecidos. Dentre as operações, possíveis de se realizar, estão a consulta de dados dos produto, suas formas de pagamento, reservas de produtos com os dispositivos, registro de pagamento e confirmação de transação referente ao pagamento.

O módulo Coin Gateway representa a generalização dos serviços de comunicação com criptomoedas. Toda comunicação com as criptomoedas passará por estes serviços, como fornecer o endereço da carteira, realizar transferências, listar as transações de uma carteira além de validar o status das transações.

É através do módulo Balance Control que acontecerá o controle das micro transações financeiras. Este módulo será responsável por disponibilizar as formas de pagamento para o Product Gateway, também podendo validar os preços dos produtos em suas criptomoedas e realizar transferências com os Coin Gateway.

O módulo Transaction Watcher servirá para monitorar as transações registradas pelo Product Gateway. As transações que estiverem em status de confirmando pagamento, serão monitoradas periodicamente por este serviço, de forma que quando a transação for confirmada, o Product Gateway seja devidamente notificado.

E por fim, o Marketplace é o módulo aonde os produtos ficarão disponíveis para serem comprados. Através do marketplace será possível verificar seus produtos disponíveis, buscar por produtos específicos e cadastrar novos. Aplicações que desejarem efetuar a reserva/compra de algum produto, deverá buscá-lo antes no marketplace.

4. Ambiente Experimental

O ambiente de testes contou com uma placa Wemos D1 para representar os dispositivos IoT, um Raspberry Pi os Coin Gateways, Balance Control, Service Registry e Product Gateway e um outro Raspberry Pi para o Marketplace. A figura 3 apresenta a disposição do ambiente experimental.

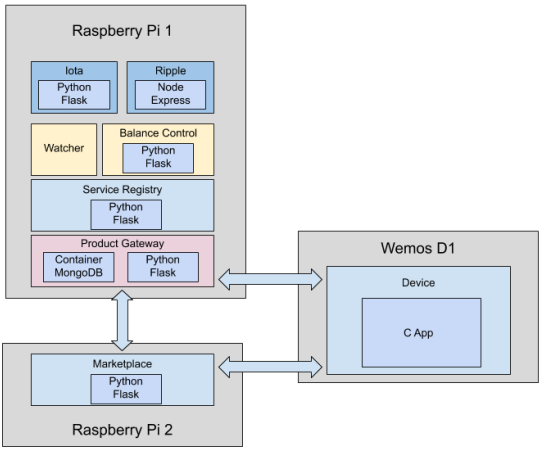

Figure 3. Ambiente experimental

O módulo Service Registry foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. As bibliotecas utilizadas foram Flask e json.

O módulo de IoT Device foi desenvolvido em C. As bibliotecas utilizadas foram ArduinoJson, ESP8266WiFi, ESP8266WebServer e ESP8266HTTPClient.

O módulo Service Registry foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. Para armazenamento de dados, foi optado pela utilização do MongoDB dentro de um container docker. As bibliotecas utilizadas foram Flask, json e pymongo.

O gateway para comunicação com a criptomoeda IOTA foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições e a biblioteca PyOTA para realizar a conexão com a rede Tangle. Após o desenvolvimento, para testar o módulo foi utilizada a rede de testes da Tangle para realizar transações entre nós.

O serviço de comunicação para a criptomoeda Ripple foi desenvolvido em JavaScript (Node.js). Ele utiliza o framework Express para receber e devolver as requisições. O motivo para escolher uma linguagem de programação diferente é que a Ripple oferece uma API de desenvolvimento com bibliotecas já prontas para Node, a ripple-lib. Para validar a utilização do gateway desenvolvido foi utilizada uma rede de desenvolvimento da Ripple.

O módulo Balance Control foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições.

Módulo para rastreamento das transações, foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. As bibliotecas utilizadas foram Flask, json, threading e time. Para armazenamento de dados, foi optado pela utilização do banco já utilizado pelo Product Gateway, dada a natureza experimental do ambiente e ambiente restrito. Apesar desta reutilização, o módulo não ficará indisponível caso o serviço Product Gateway esteja fora do ar, apenas se comportará como se não houvessem transações sendo rastreadas no momento.

O módulo Marketplace é a porta de acesso aos produtos da arquitetura. Foi desenvolvido em Python utilizando o framework web Flask para envio e recebimento de requisições. As bibliotecas utilizadas foram Flask e json.

4.1. Resultados Obtidos

Para análise do ambiente experimental, concebeu-se uma forma de testá-lo, obtendo dados e os analisando. O teste foi efetuar 100 transações com cada uma das criptomoedas que tiveram um Coin Gateway implementado ou seja, 100 transações com IOTA e 100 transações com Ripple. Depois sucedeu a análise do tempo dispendido nas transações.

Para executar o teste, utilizou-se um pequeno projeto em Python, que representou uma instância da arquitetura que fez uso somente de dpois módulos, Balance Control e Coin Gateway. Essa medida foi adotada para provar a diversidade possível com a arquitetura, pois uma instância dela não necessita de todos os módulso para operar. Esta intância comunicou-se com outra, através dos Marketplace e do Balance Control para realização de operações. Houveram duas execuções, uma para as transações em IOTA e outra para as transações em Ripple.

Após a execução, os dados dos tempos das transações foram extraídos, para geração dos gráficos que serão apresentados a seguir. Os termos utilizados no gráfico, são o do tempo de uma transação, que consiste na soma do tempo levado para fazer um pagamento com o tempo levado para confirmar o pagamento.

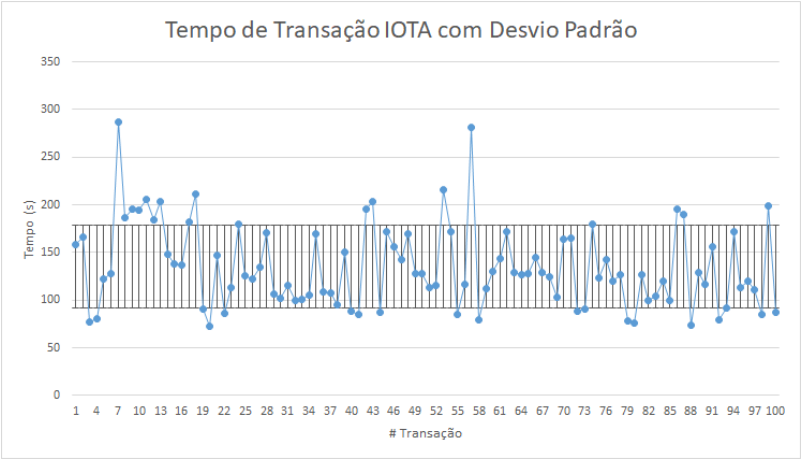

Figure 4. Dispersão das transações IOTA

Para IOTA, o tempo total de execução foi de 49 minutos e 50 segundos. A duração média de transação foi de aproximadamente 135 segundos sendo 13% do tempo gasto com o pagamento e 87% com a confirmação do pagamento. Analisando a dispersão dos dados, em conjunto com seu desvio padrão na figura 4, 36% das transações ficam além do desvio, indicando que os tempos de transação de IOTA têm considerável variação.

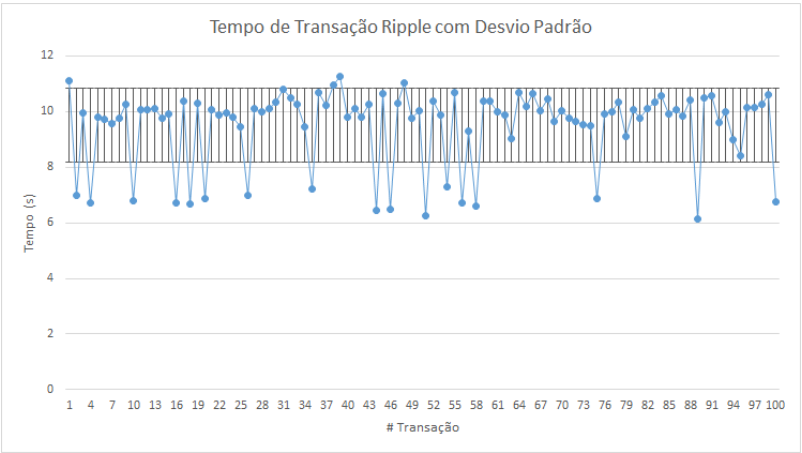

Figure 5. Dispersão das transações Ripple

Ao realizar a mesma análise sobre as transações de Ripple, que possuem o tempo de execução de 17 minutos e 39 segundos, tendo um tempo médio por transação de 9,5 segundos e um desvio padrão de 1,33 segundos, com 21% das transações ficando foram do intervalo compreendido pelo desvio padrão sobre a média, ilustrado na figura 5. O tempo de cada transação em Ripple, ficou com uma divisão de 33,38% em pagamento e 66,62% para pagamento.

5. Conclusões e Trabalhos Futuros

Este trabalho teve como proposta, conceber uma arquitetura para controle de micro transações financeiras em um contexto de EoT, baseada em microsserviços. A fim de alcançar este objetivo, foram pesquisados e analisados conceitos, propostas e utilizações de IoT, Blockchains, Microsserviços e EoT, ora integrados, ora isolados. A arquitetura proposta possuía gateways de comunicação para criptomoedas, dispositivos IoT, e monitoramento de transações destes dispositivos, além de um módulo para autenticação dos participantes da arquitetura. A flexibilidade necessária em um ambiente aonde houvesse variedade de tecnologias envolvidas, que estão em constante evolução, como IoT e blockchains, levou a utilização de microsserviços.

Para validação da proposta, foi desenvolvido um ambiente experimental. Nele, o gateway de comunicação com criptomoedas (Coin Gateway), realizava as consultas e transferências com criptomoedas, transferindo fundos e consultando status de transações submetidas através dele. Já o gateway de comunicação com os dispositivos IoT, chamado de Product Gateway, é o serviço responsável por realizar as comunicações que os dispositivos necessitassem, consultando os produtos que o dispositivo tinha, formas de pagamento, efetuando reservas de itens e confirmando transações. O Balance Control realiza o controle das micro transações, sendo responsável por informar as formas de pagamento para o Product Gateway e validando os preços nas criptomoedas que serão transferidos para os Coin Gateway. O módulo de monitormanento de transações (Transaction Watcher), é incumbido de constantemente validar as transações passadas pela arquitetura, notificando aos Product Gateway quando confirmadas. O módulo de autenticação - Service Registry, valida os elementos da rede, encaminhando para cada um as chaves de acesso aos serviços que necessitassem.

O ambiente experimental, consistiu de uma placa Wemos D1 para representar os dispositivos IoT, um Raspberry Pi para os Coin Gateways, Balance Control, Service Registry e Product Gateway e um outro Raspberry Pi para o Marketplace. Foram utilizadas as linguagens Python, JavaScript e C, a fim de demonstrar a flexibilidade da arquitetura. A comunicação era realizada através da internet, com requisições utilizando o framewerk Flask. As criptomoedas selecionadas para os testes foram IOTA e Ripple, aonde para cada uma, foram realizadas 100 transações e analisados seus tempos de execução.

Embora os tempos de transações com a criptomoeda IOTA tenham tido uma dispersão grande, os resultados apresentados foram satisfatórios, levando a crer que a arquitetura é apta a atender a necessidade da proposta em um contexto real.

Como trabalhos futuros, foi identificada a necessidade de um teste em um ambiente experimental mais complexo, com formas de comunicação distintas e mais elementos como outras criptomoedas e diversificados tipos de dispositivos IoT. Além de

uma melhora que torne possível analisar o tempo dispendido em comunicação entre os módulos para análise da performance da arquitetura, considerando que o teste feito não diferenciou o tempo das transações neste aspecto.

A necessidade de um dispositivo precisar efetuar transações de compra através de outro marketplace dando inicio a um novo ciclo de interações, também parece uma boa maneira de validação. Também se faz necessária uma abordagem mais segura na forma da comunicação entre os serviços.

6. Referências

BARROZO, L. M.Descoberta semântica de microservices em contêineres— UniversidadeFederal de Santa Catarina, Florianopolis, SC, Brasil, 2016.

Butzin, B.; Golatowski, F.; Timmermann, D. Microservices approachfor the internet of things. p. 1–6, Sep. 2016.

BACK, R. P.Análise Comparativa de Técnicas de Integração entre Microsserviços— Universidade Federal de Santa Catarina, Florianopolis, SC, Brasil,2016.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things:A survey.Comput. Netw., Elsevier North-Holland, Inc., New York,NY, USA, v. 54, n. 15, p. 2787–2805, out. 2010. ISSN 1389-1286.<http://dx.doi.org/10.1016/j.comnet.2010.05.010>.

CHRISTIDIS, K.; DEVETSIKIOTIS, M. Blockchains and smartcontracts for the internet of things.IEEE Access, v. 4, p. 2292–2303,2016. ISSN 2169-3536.

FARELL, R.An analysis of the cryptocurrency industry. 2015.<https://repository.upenn.edu/>. Acessado em 25/06/2019.

FERNáNDEZ-CARAMéS, T. M.; FRAGA-LAMAS, P. A review onthe use of blockchain for the internet of things.IEEE Access, v. 6, p.32979–33001, 2018. ISSN 2169-3536.

FOUNDATION, I.The Industry Marketplace. 2019.

<https://industry.iota.org/industrymarketplace>.Acessado em 14/10/2019.

HUCKLE, S. et al. Internet of things, blockchain and shared economyapplications.Procedia computer science, Elsevier, v. 98, p. 461–466,2016.

FAMILIAR, B. Microservices, IoT, and Azure: Leveraging DevOpsand Microservice Architecture to Deliver SaaS Solutions. Berkeley,CA: Apress, 2015. 133-163 p. ISBN 978-1-4842-1275-2.

AL-FUQAHA, A. et al. Internet of things: A survey on enablingtechnologies, protocols, and applications.IEEE CommunicationSurveys & Tutorials., v. 17, n. 4, 2015.

MUKHOPADHYAY, U. et al. A brief survey of cryptocurrencysystems. p. 745–752, Dec 2016.