UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

Vinícius Gamarra Andrade Sousa

Um Aplicativo de Gerenciamento de Carteira de Criptomoedas

São Luís - MA
2022

Vinícius Gamarra Andrade Sousa

Um Aplicativo de Gerenciamento de Carteira de Criptomoedas

Trabalho de Conclusão de Curso apresentado ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.. .

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

Universidade Federal do Maranhão

Orientador: Prof. Dr. <Mário Antônio Meireles Teixeira >

São Luís - MA

2022

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Diretoria Integrada de Bibliotecas/UFMA

Andrade Sousa, Vinícius Gamarra.
Um Aplicativo de Gerenciamento de Carteira de Criptomoedas / Vinícius Gamarra Andrade Sousa. - 2022.
49 p.

Orientador(a): Mário Antônio Meireles Teixeira.
Curso de Ciência da Computação, Universidade Federal do Maranhão, São Luís Maranhão, 2022.

1. Aplicativo. 2. Criptomoeda. 3. Mobile. 4. Tecnologia. I. Meireles Teixeira, Mário Antônio. II. Título.

Vinícius Gamarra Andrade Sousa

Um Aplicativo de Gerenciamento de Carteira de Criptomoedas

Trabalho de Conclusão de Curso apresentado ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.. .

Banca Examinadora

Prof. Dr. <Mário Antônio Meireles Teixeira >
Orientador
Universidade Federal do Maranhão

Prof. Dr. <Anselmo Cardoso de Paiva>
Examinador Interno
Universidade Federal do Maranhão

Prof. Dr. <Samyr Béliche Vale>
Examinador Interno

São Luís - MA
2022

Agradecimentos

Primeiramente gostaria de agradecer a Deus, pois sem ele nada seria possível.

Gostaria de agradecer aos meus pais, que sempre foram cuidadosos, me deram condições de atingir as minhas metas e apoio sempre que precisei.

Agradeço também ao meu orientador Professor Mário Meireles, por ter me guiado durante essa missão,mostrando o caminho a ser seguido em momentos de dúvida.

"A riqueza não consiste em ter grandes posses, mas em ter poucas necessidades."

(Epicteto)

Resumo

A evolução tecnológica , tem trago uma grande mudança na área monetaria, surgindo novas formas de troca e soluções no meio digital. As Criptomoedas e a BlockChain fazem parte dessa mudança e ter conhecimento delas é necessario, pois economizam tempo e agilizam processos de forma simples do ponto de vista dos usuários comuns. Dessa forma buscou-se estudar e criar uma aplicativo móvel que pudesse ser utilizada da forma mais simples possível de modo a facilitar o processo de obtenção de informações sobre as criptomoedas disponíveis no mercado. O objetivo deste trabalho é discorrer sobre Criptomoedas e BlockChain, bem como seus conceitos e processos necessários. Para isso foi pesquisado ferramentas e maneiras para trazer uma proposta de Visualização de criptomoedas, além disso criou- se uma aplicação para demonstrar de forma prática o funcionamento de todo o processo

necessário.

Palavras-chave: Criptomoedas , BlockChain, Aplicativo móvel.

Abstract

Technological evolution has brought a great change in the monetary area, emerging new forms of exchange and solutions in the digital environment. Cryptocurrencies and BlockChain are part of this change and having knowledge of them is necessary, as they save time and streamline processes in a simple way from the point of view of common users. Of that In this way, we sought to study and create a mobile application that could be used as simply as possible in order to facilitate the process of obtaining information about cryptocurrencies available on the market. The objective of this work is to discuss Cryptocurrencies and BlockChain, as well as their necessary concepts and processes. For this, tools and ways to bring a proposal for Visualization of cryptocurrencies, in addition, created- if an application to demonstrate in a practical way the operation of the whole process

Keywords: Cryptocurrency, BlockChain, Mobile App

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

JVM	*Javascript Virtual Machine*
P2P	*peer to peer*
KYC	Know-Your-customer

Sumário

1 Introdução

1.1 Contextualização

O mercado de criptomoedas cresceu de forma irregular e com grande velocidade desde sua recente criação com o Bitcoin, lançada em 2009. Com o passar dos anos, inúmeras criptomoedas foram criadas e até o presente momento existem aproximadamente vinte mil criptoativos segundo coinmarketcap.com.

Existem dois tipos de usuários Bitcoin: usuários comuns e assim chamados mineradores de Bitcoin. Usuários comuns de Bitcoin usam uma carteira digital semelhante aos aplicativos bancários eletrônicos. A carteira é software para uma gestão de dinheiro Bitcoin, assim para enviar e receber pagamentos em bitcoins. Bitcoins existem apenas como informações em arquivos em um computador ou dispositivo móvel. Acesso a esses arquivos é restrito ao detentor da chave privada, que é usado para garantir o dinheiro.

1.2 Justificativa

Com o crescimento acelerado do mercado de criptoativos, muitos podem ficar perdidos em ter de acompanhar tantas informações sendo geradas diariamente. Dessa forma, websites e aplicativos que oferecem visualizações e simulações de criptomoedas, se mostram bastante úteis para uma boa tomada de decisão dos interessados no tema.

1.3 Objetivo geral

Este trabalho tem como foco disponibilizar de forma simples, a simulação de gerenciamento de uma carteira de criptoativos por meio de um aplicativo mobile, aplicando métodos de desenvolvimento e padrões de projeto com a finalidade de tornar a aplicação de fácil manutenção e uso. Além disso, utilizar métodos de engenharia de software para auxiliar durante e após o desenvolvimento para eventuais consultas.

1.3.0.1 Objetivos Específicos

- Estudar os conceitos de blockchain e criptomoedas.
- Desenvolver habilidades de programação mobile.
- Entender como uma carteira simples de criptomoeda funciona.
- Aprender o ciclo de desenvolvimento de um aplicativo móvel.

1.4 Organização do Trabalho

Esta introdução faz parte do Capítulo 1, onde são abordadas as justificativas, objetivos e organização do trabalho.

No Capítulo 2, são apresentados os conceitos que dão base ao trabalho, são eles: blockchain, criptomoedas, aspectos de rede,segurança e aplicações destas tecnologias. Também é feita uma análise gráfica que mostra o crescimento das criptomoedas no decorrer do tempo.

O Capítulo 3 discorre sobre a metodologia utilizada no desenvolvimento do aplicativo. Além disso, também são apresentadas as funcionalidades planejadas, protótipos e diagramas UML.

O aplicativo em si é abordado no Capítulo 5, sendo apresentadas a tecnologia, arquitetura utilizada, códigos de cada tela e funcionalidades executas nelas, a API utilizada para o consumo de dados e capturas de tela do aplicativo em execução.

2 Fundamentação Teórica

Nesse capítulo será apresentado o conceito de blockhain e criptomoedas, abordando suas definições, as principais características, pontos positivos no uso dessas tecnologias, além de listar e explicar sobre as principais aplicações, aspectos de rede e segurança.

2.1 Blockchain

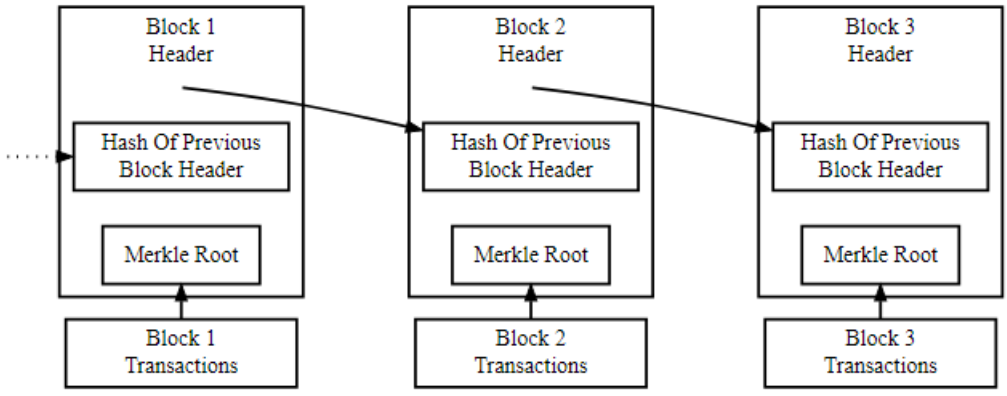
A blockhain é um banco de dados distribuido, descentralizado de registros, que permite transações rápidas e confiaveis sem o monitoramento de um ente centralizado (SHETH; DATTANI, 2019).

2.1.1 Contexto Histórico

A idéia principal por trás da blockchain surgiu por volta de 1980 e 1990. Em seu artigo *The Part-Time Parliament* (LAMPORT, 2019), Leslie descrevia um modelo de consenso para atingir um resultado em uma rede onde os computadores ou a rede podiam ser não confiáveis.

Esse conceito foi aplicado em um dinheiro eletronico em 2008 e mostrado no artigo *Bitcoin: A Peer to Peer Electronic Cash System* (NAKAMOTO, 2008), que foi publicado pseudoanonimamente por Satoshi Nakamoto, e mais tarde foi estabelecida a rede de blockchain da criptomoeda bitcoin. Esse artigo continha diagramas que a maioria das criptomoedas modernas seguem(cada uma com suas modificações). O bitcoin foi apenas a primeira de muitas aplicações da blockchain.

Figura 1 – Bitcoin blockchain overview



Fonte: <https://developer.bitcoin.org/devguide/block$_c$*hain.html* >

Muitos esquemas de dinheiro eletrônico existiram antes do Bitcoin (por exemplo, ecash e NetCash), mas nenhum deles alcançaram amplo uso. O uso de uma blockchain

permitiu que o Bitcoin fosse implementado em uma distribuição de forma que nenhum usuário único controlasse o dinheiro eletrônico e que nenhum ponto único de falha exista; isso promoveu seu uso. Seu principal benefício era permitir transações diretas entre usuários sem a necessidade de um terceiro confiável. Também possibilitou a emissão de novas criptomoeda de forma definida para aqueles usuários que conseguem publicar novos blocos e manter cópias do livro-razão; esses usuários são chamados de mineradores em Bitcoin. O pagamento automatizado dos mineradores possibilitou a administração distribuída do sistema sem a necessidade de organização. Ao usar uma blockchain e validação baseada em consenso, foi criado um mecanismo de autopoliciamento que garantiu que apenas transações e blocos válidos fossem adicionados ao blockchain (YAGA et al., 2019).

Em 2013, várias propostas e companhias estavam promovendo a idéia de usar a tecnologia da blockchain sem se prender à moeda digital. Em tese, nas "permissioned blockchains", somente membros pré-aprovados poderiam adicionar dados na blockchain, que existe como um registro compartilhado entre as partes participantes. Sem cálculo de prova de trabalho, com a veracidade das transaçõess sendo realizadas por membros identificáveis e responsáveis uns com os outros. No próximo tópico, serão examinadas propostas comuns de uso da tecnologia da blockchain, e também apresentados os tipos de blockchain.

2.1.2 Tipos de blockchain

Um sistema pode seguir dois modelos, centralizado ou distribuido. A tecnologia de Blockchain segue o método distribuido, onde diversos nós são conectados entre si, sem a necessidade de um nó central conectando todos os nós da rede. As criptomoedas estão entre os exemplos mais famosos de sua aplicabilidade, mas também pode ser utilizada em outras áreas como a financeira. A blockchain é um arquivo aberto onde cada transação é registrada e todos estão conectados um com o outro.Ela implementa um banco de dados distribuído P2P que permite armazenar, verificar e auditar a transação pelos peers presentes na rede. Uma vez que a transação foi adicionada na blockchain, é impossível mudar, deletar ou adulterar. Para a transação ser registrada, ela tem de ser validada pela blockchain através de mecanismos de consenso (SHETH; DATTANI, 2019).

Permissionless Blockchain

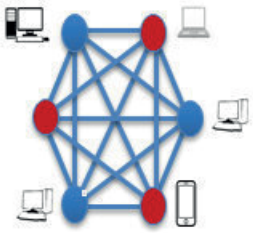
Uma Permissionless blockchain é aquela em que qualquer um pode rodar um nó ou software de mineração.Qualquer um pode acessar uma carteira, escrever dados nas transações desde que siga as regras da blockchain. Esses tipos de blockchain são abertas e qualquer um pode auditar a qualquer momento. Podem ser chamadas de blockchains publicas e exemplos desse tipo são o bitcoin e lite coin (SHETH; DATTANI, 2019).

Permissioned Blockchain

Uma Permissioned blockchain é aquela em que é necessária a permissão de alguma autoridade(centralizada ou descentralizada) para publicar os blocos. Desde que apenas usuários autorizados mantenham a blockchain, é possível restringir o acesso de leitura e assinatura das transações. Assim como podem restringir, também podem liberar o acesso a todos ou apenas à usuários autorizados. Podem ser código aberto ou fechado. Elas devem ter a mesma rastreabilidade de ativos digitais, assim como, um banco de dados resiliente, redundante e distribuído de uma permissionless blockchain. Além disso, também usam modelos de consenso para publicar blocos, mas não são tão custosos como na permissionless, pelo fato de a publicação ser feitas apenas por pessoas autorizadas e não toda uma rede competindo par assinar a autenticidade do bloco (YAGA et al., 2019).

Um resumo do que acabou de ser explicado na figura abaixo:

Figura 2 – Blockhain permissionado e não permissionado

Nó simples somente inicia ou recebe uma transação Nó validador valida, inicia ou recebe uma transação	 Privado Permissionado	 Público Não Permissionado
Acesso à rede	Necessita de autorização	Acesso aberto
Legislação e regulação	Conforme legislação e regulações	Pode ter regras próprias
Quem são os validadores	Grupo pré-selecionado	Anônimos
Potenciais aplicações	Ambientes corporativos fechados	Aplicações de acesso aberto

Fonte: (BRAGA; MARINO; SANTOS, 2017)>

2.1.3 Potenciais aplicações da tecnologia de blockchain

Pagamentos digitais: Sistemas atuais de pagamento requerem a confiança em um terceiro, que geralmente são entes centralizados, registrando todas as transações e ativos de cada usuário. A logistica de transação parte de um dos lados para o intermediário, que valida e efetua a transação para a outra parte. Na blockchain, as transações são transmitidas para toda a rede, que envolve muitas transmissões e mais poder de processamento e tempo. A transação se torna parte da blockchain quando copiada para todos os nós da rede. Isso explica o motivo de as transações bancárias serem mais rápidas que a do bitcoin por exemplo (AMMOUS, 2016).

Contratos: Atualmente, contratos são esquematizados por advogados, julgado por tribunais e reforçados pela policia. Um "smart contract"é um programa simples que roda na blockchain da Ethereum. É uma coleção de código e dados que residem em um endereço específico na blockchain da Ethereum (ETHEREUM, 2022).

Sistemas criptográficos de contrato inteligente, como o Ethereum, codificam contratos em uma blockchain para fazer autoexecutáveis, sem possibilidade de recurso ou reversão, e fora do alcance dos tribunais e polícia. “O código é a lei” é um lema usado por programadores de contratos inteligentes (YAGA et al., 2019).

Banco de dados e gerenciamento de registros Blockchain é um banco de dados e registro de ativos confiável e à prova de modificações, mas apenas para a moeda nativa da blockchain, apenas se a moeda valer o suficiente para que a rede tenha poder de processamento suficiente para resistir à um ataque. Para qualquer outro ativo, físico ou digital, a blockchain é confiável apenas como responsável por estabelecer uma conexão entre o ativo e o que referi-lo na blockchain. Não existem ganhos de eficiência ou transparência por utilizar uma permissioned blockchain aqui, como ela somente é confiàvel se o participante garantir permissão para escrever. A introdução do blockchain na manutenção de registros dessa parte só vai torná-lo mais lento, sem adicionar segurança ou imutabilidade, uma vez que não há prova de trabalho. Confiar em terceiros intermediários devem permanecer, enquanto o poder de processamento e o tempo necessário para executar o banco de dados aumenta. Um blockchain protegido por um token pode ser usado como um serviço de cartório, onde contratos ou documentos são criptografados em um bloco de transações, permitindo que qualquer parte acesse o contrato e certifique-se de que a versão exibida é a que foi hash no momento. Esse serviço fornecerá um mercado para espaço de bloco escasso, mas é impraticável com qualquer blockchain sem moeda (AMMOUS, 2016).

2.1.4 Desafios economicos da tecnologia de blockchain

Ao analisar os tres potenciais de aplicação da tecnologia de blockchain, quatro obstáculos principais para larga adoção são identificados. Segundo (AMMOUS, 2016), esses obstáculos seriam a redundancia, escalabilidade, conformidade regulatória e irreversibilidade.

Redundância: Ter cada transação armazenada por cada membro da rede é uma redundância custosa para o objetivo de remover uma intermediação. Para cada intermediario, seja financeira ou legal não teria sentido adicionar essa redundância enquanto permanecer intermediário. Não existe uma boa razão para qualquer banco querer ter registros completos ’lidando um com o outro’. Essa redundância oferece custos aumentados sem ter benefícios concebíveis.

Escalabilidade: Uma rede distribuída onde todos os nós registram todas as transações vão ter em comum o crescimento exponencial do livro-razão mais rápido que

o número de membros da rede. Portanto o armazenamento e o peso computacional dos membros da rede vão eventualmente se tornar muito pesados para os membros da rede poderem lidar conforme a rede cresce. Blockchains sempre enfrentarão essa barreira para o dimensionamento eficaz, e isso explica por que, à medida que os desenvolvedores de bitcoin procuram soluções para dimensionamento, eles estão se afastando do modelo de blockchain descentralizado puro para ter pagamentos liberados por intermediários fora do blockchain. Há um claro trade-off entre escala e descentralização. Caso um blockchain seja feito para acomodar volumes maiores de transações, os blocos precisam ser maiores, o que aumentaria o custo de adesão à rede e resultaria em menos nós, tornando a rede mais centralizada. A maneira mais econômica de ter um grande volume de transações é a centralização em um nó.

Conformidade regulatória: Blockchains com sua própria moeda, como Bitcoin, existem ortogonalmente à lei, pois não há nada que qualquer autoridade governamental possa fazer para afetar ou alterar sua operação, e o presidente do Federal Reserve disse que não tem autoridade para regular o Bitcoin. As transações serão compensadas se válidas, e não serão compensadas se não forem válidas, e não há nada que os reguladores possam fazer para derrubar o consenso do poder de processamento da rede. A aplicação da tecnologia blockchain em setores altamente regulamentados, como direito ou finanças, com moedas diferentes do Bitcoin resultará em problemas regulatórios e complicações legais. Os regulamentos foram projetados para uma infraestrutura muito diferente da blockchain e as regras não podem ser facilmente adaptadas para se adequar à operação blockchain, com a abertura radical de ter todos os registros distribuídos a todos os membros da rede. Além disso, as blockchains operam online em jurisdições com diferentes regras regulatórias, dificultando a garantia da conformidade com todas as regras.

Irreversibilidade: Com pagamentos via intermediários, erros humanos ou de software podem ser facilmente revertidos apelando para o intermediário. Em uma blockchain, as coisas são infinitamente mais complicadas. Uma vez que um bloco foi confirmado e novos blocos estão sendo anexados a ele, só é possível reverter qualquer uma de suas transações empacotando 51 porcento do poder de processamento da rede para se engajar em um 'hard fork' da rede, onde todos esses nós concordam em se mover simultaneamente para uma blockchain alterada. A tecnologia Blockchain, afinal, destina-se a replicar transações em dinheiro on-line e, portanto, replicará a irreversibilidade das transações em dinheiro e não trará nenhum dos benefícios da intermediação de custódia. Muito provavelmente, esse fork nunca terá sucesso se for tentado com o Bitcoin, pois exigiria que muitos atores díspares concordassem e gastassem recursos sem ganho. Após o incidente do DAO, tornou-se evidente que, para qualquer blockchain que não seja bitcoin, o hashrate da rede é pequeno o suficiente e os designers da moeda influentes o suficiente para derrubar partes do blockchain que eles não gostam. Isso significa que a alegação de "imutabilidade" da tecnologia blockchain só é realmente válida no caso do Bitcoin. Para qualquer outro

blockchain, os operadores do blockchain, ou uma autoridade regulatória, podem de fato alterar o registro. Um blockchain que é alterável é um exercício completamente inútil de sofisma de engenharia: ele usa um método muito complexo e caro de liberação para remover intermediários e estabelecer imutabilidade, mas concede a um intermediário a capacidade de derrubar essa imutabilidade. A melhor prática atual nesses campos contém reversibilidade e supervisão por autoridades legais e reguladoras, mas emprega métodos mais baratos, rápidos e eficientes.

2.2 Criptomoedas

Após a análise de aplicabilidade, percebe-se que a única aplicação comercialmente bem-sucedida da tecnologia blockchain até agora é o dinheiro digital e, em particular, o Bitcoin. Os aplicativos potenciais mais comuns divulgados para tecnologia blockchain, pagamentos, contratos e registro de ativos só são viáveis na medida em que são executados usando a moeda descentralizada do blockchain. Todas as blockchains sem moedas não passaram do estágio de protótipo para a implementação comercial porque não podem competir com as melhores práticas atuais em seus mercados. O design do Bitcoin está disponível gratuitamente online por mais de 8 anos, e os desenvolvedores podem copiá-lo e melhorá-lo para introduzir produtos comerciais, mas nenhum desses produtos apareceu. O teste de mercado mostra que as redundâncias de registro de transações e prova de trabalho só podem ser justificadas para fins de produção de dinheiro digital e rede de pagamentos sem intermediação de terceiros (AMMOUS, 2016).

Em suma, uma criptomoeda é um sistema de cunhagem virtual que funciona como uma moeda padrão, permitindo que os usuários realizem pagamento virtual para bens e serviços sem uma autoridade central confiável. As criptomoedas dependem da transmissão de informações digitais, utilizando métodos para garantir transações legítimas e exclusivas. Bitcoin tomou o mercado de moedas digitais um passo adiante, descentralizando a moeda e liberando-a do poder hierárquico estruturas. Em vez disso, indivíduos e empresas fazem transações com a moeda eletronicamente em um rede ponto a ponto. Ele chamou muita atenção a partir de 2011, e várias altcoins - um nome geral para todas as outras criptomoedas pós-Bitcoin – logo apareceram (FARELL, 2015).

2.2.1 Aspectos de rede e segurança

Quando se trata de aspecto de segurança, é necessário que certos pontos sejam atendidos. Segundo (FARELL, 2015), a moeda deve ser capaz de mudar de proprietário, registrar transações combinando duas assinaturas de cada lado da operação e um marcador de data e hora, validando o registro. Dessa forma, o código representa a moeda e o caminho entre a rede. Ocorrerá um broadcast do código para todos os nós (computadores conectados

e rodando o software de rede da criptomoeda) na rede. Entretando, é necessário que a maioria dos nós concordem sobre as transações que ocorreram, de outra forma, ataques de gasto duplo e negação de serviço podem ocorrer. O mecanismo usado para consenso entre os nós coloca a integridade no sistema verificando se a transação é legítima. Dessa forma, as transações são verificadas e o sistema se tornou seguro, implementando mecanismos que tornou custosa a violação da integridade do sistema. As transações são registradas combinando as assinaturas digitais de cada parte e um timestamp, para que a data da transação seja registrada. Este novo código representa a moeda e seu caminho pela rede. Este código é então transmitido para todos os nós (computadores conectado e executando o software de rede de criptomoeda) na rede. No entanto, é necessário que a maioria dos nós concorde com transações que tenham Caso contrário, podem ocorrer ataques de gastos duplos e de negação de serviço (DoS). o mecanismo usado para chegar a um consenso entre os nós coloca a integridade no sistema verificando se a transação é realmente legítima. Assim, as transações são verificadas, e o sistema tornou-se seguro, implementando certos mecanismos que tornam muito caro violar a integridade do sistema. Várias criptomoedas desenvolveram novos recursos para usar como meio de segurança de rede. O recurso pode ser uma combinação de eletricidade, tempo, ou entrega temporária de moedas, e representa o custo para proteger a rede. Os mineradores - aqueles que possuem o recurso subjacente e, portanto, o gastam - protegem a rede, e são remunerados por seu trabalho na forma de taxas de transação ou moedas. O mecanismo usado para proteger a rede determina o recurso escolhido e o método usado para pagar os mineiros. Assim, o mecanismo de segurança de rede subjacente de cada moeda tem um impacto significativo na economia subjacente da moeda.

2.2.2 Exchanges

Uma exchange de criptomoedas é um negócio que permite clientes trocarem criptomoedas. Elas podem ser criadoras de mercado, normalmente usando a variação de bid-ask como comissão pelos serviços, ou plataforma de ordem, cobrando taxas (FANG et al., 2022).

Vantagens de usar exchanges e criptomoedas

Grandes flutuações: A volatilidade das criptomoedas geralmente são um atrativo par aqueles que gostam de especular e investidores. As rápidas flutuações dos preços durante o dia podem prover grandes retornos financeiros, mas tambem grandes riscos.

Mercado 24H: O mercado de criptomoeda está disponível 24 horas por dia, pelo fato de ser descentralizado. Diferente de comprar e vender ações e fundos imobiliarios, o mercado de criptomoedas não é operado fisicamente de uma única localização.

Perto da anonimidade: Comprar produtos e serviços usando criptomoedas pode

ser realizado online e não ser necessário que sua identidade seja publicada. Diversas exchanges tem medidas de KYC para identificar seus usuários e clientes. O KYC(Know Your Client) permite que instituições financeiras reduzam o risco financeiro enquanto maximizam a anonimidade do dono da carteira.

Recursos "inteligentes" programáveis: Algumas criptomoedas podem trazer outros benefícios aos titulares, incluindo propriedade limitada e direitos de voto. As criptomoedas também podem incluem uma participação acionária parcial em ativos físicos, como obras de arte ou imóveis e em ativos digitais, como alugar criptomoedas em uma plataforma como a blockfi.com.

Desvantagens de usar exchanges e criptomoedas

Problema de escalabilidade: Antes da expansão massiva da infraestrutura tecnológica, o número de transações e a velocidade das transações não podiam competir com o comércio tradicional de moedas. Embora seja uma desvantagem, estão surgindo as soluções de segunda camada, como a Lightning Network, que buscam resolver esse problema de escalabilidade. Lightning é uma rede descentralizada que usa a funcionalidade de contrato inteligente no blockchain para permitir pagamentos instantâneos em uma rede de participantes (LIGHTNINGNETWORK, 2020).

Problemas de segurança cibernética: Como tecnologia digital, as criptomoedas estão sujeitas a ataques cibernéticos falhas de segurança e podem cair nas mãos de hackers. Recentemente, mais de US 600 milhões de ethereum e outras criptomoedas foram roubadas em agosto de 2021 na plataforma baseada em blockchain Poly Network (FORBES, 2021).

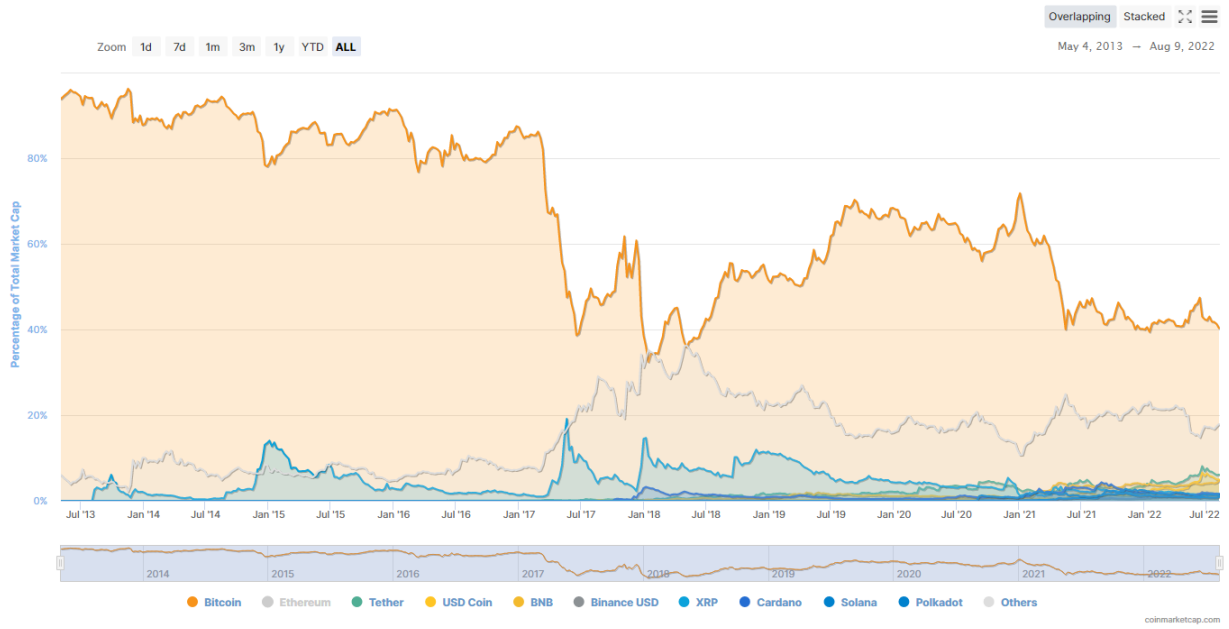
Regulamentação A situação das criptomoedas em relação à regulamentação ainda não está totalmente resolvida, pelo fato de existirem moedas que não são regulamentadas tão facilmente, como o Bitcoin, bastando que os usuários tenham uma carteira física e transacionem de forma P2P e a solução até o momento foi regulamentar os intermediários, as exchanges, como a Binance. Para uns, a falta de regulamentação é um atrativo, para outros, um ponto negativo. Segundo (VEJAČKA, 2014), Permissão mais ampla de uso de criptomoeda definitivamente suporta seu uso e gera maior confiança nele. No entanto, a maioria dos países já permite o uso de criptomoeda e lhes concederam o status de moeda. Esta ação aumentou a possibilidade de seu uso em economia jurídica. Porém na economia ilegal mantém seu status de meio de troca seguro e anônimo

2.2.3 Criptomoedas no decorrer do tempo

Conforme o tempo passa, o mercado de criptoativos cresce e segundo dados encontrados no coinmarketcap.com no dia 9 de agosto, existem 20.529 criptomoedas ativas, sendo que em 2013, o Bitcoin representava cerca de 93% do mercado. No decorrer do tempo, novas moedas foram surgindo e atualmente o Bitcoin representa aproximadamente

40% do mercado e em segundo lugar a moeda Tether, com 6%. O gráfico abaixo exemplifica a variância deste mercado desde 2013 até os dias atuais.

Figura 3 – Major Cryptoassets By Percentage of Total Market Capitalization (Bitcoin Dominance Chart)



Fonte: <(COINMARKETCAP, 2022)>

3 Metodologia

3.1 Metodologia

3.1.1 Visão Geral

Serão abordadas nos próximos tópicos as técnicas utilizadas para a construção do aplicativo, tendo como base o ciclo de desenvolvimento, que ajuda a lidar com a produção de funcionalidades, criação de protótipos e diagramas.

3.1.2 Ciclo de desenvolvimento de aplicativos móveis

O ciclo de vida de desenvolvimento não é diferente em termos práticos, tanto para web quanto para mobile, ambos compartilham, geralmente, cinco etapas principais no processo. São elas, **Concepção**, a ideia que gera o aplicativo, **Design**, a parte em que a interface de usuário é planejada, sua experiência enquanto utiliza. **Desenvolvimento**, a fase que mais se utiliza recursos, geralmente onde o aplicativo é construído. **Estabilização**, um aplicativo desenvolvido suficientemente para que seja testado, podendo entrar em fase beta para que um público possa usá-lo e dar suas impressões acerca dele. **Implantação** (DAVIDORTINAU et al., 2022). Este método foi escolhido por se encaixar de forma mais específica na criação de aplicativos móveis, enquant que as outras formas, como o modelo em cascata são mais genéricos.

3.1.2.1 Concepção

A concepção é a fase em que se define e melhora a ideia que irá gerar um aplicativo. A criação de funcionalidades podem ser facilitadas criando-se atores, que são os usuários, e casos de uso, que são as ações realizadas e objetivos.

Exemplificando, um aplicativo de simulação de carteira de criptomoedas tem como ator o usuário, este pode ver uma lista de criptomoedas e pesquisar por uma, caso queira, pode ver mais informações sobre alguma em específico. Sendo assim, os casos de uso nessa situação seriam mostrar as moedas, pesquisar por uma e ver informação da cripto. Com esse cenário em mente, é possível saber quais telas devem ser feitas para que esses casos se concretizem.

3.1.2.2 Design

Geralmente, o design da interface é feito por meio de softwares específicos para tal, como o Figma, que permite design de telas de forma livre. É importante personalizar

a experiência do usuário para o uso do aplicativo para diferentes plataformas caso o aplicativo seja multiplataforma, pois um tablet tem maior espaço que um smartphone e o Android utiliza componentes diferentes do IOS .

3.1.2.3 Estabilização

É o método de corrigir os bugs do aplicativo, não apenas na parte funcional mas também na usabilidade e desempenho. É aconselhável começar a estabilização desde cedo, pois evita chegar no ponto em que os problemas já escalaram, tornando a correção trabalhosa. Geralmente os aplicativos passam por níveis. Protótipo, apenas funcionalidades básicas Alfa, as funcionalidades básicas estão prontas mas ainda não foram testadas, as funcionalidades secundárias podem ainda não estarem presentes, Beta,houve testes, algumas correções de bugs e problemas conhecidos ainda podem estar presentes, e a Versão Release Candidate, quando a aplicação está pronta, livre de bugs conhecidos e apta para o uso.

3.1.3 Funcionalidades

A aplicação mobile apresenta de forma simples e direta, escolhas ao usuário que o permita procurar informações rápidas, simulações de cripto ativos em uma carteira e também informações um pouco mais específicas, como um gráfico de preços.

É mostrado abaixo a lista de funcionalidades da aplicação:

- Não funcionais:

  Multiplataforma: A aplicação deverá rodar tanto no ios quanto no Android. (Prioridade: Essencial, Esforço: Médio)

  Internet: Para que o usuário tenha acesso à funcionalidades cruciais é necessária a conexão com a internet, sendo assim, um requisito obrigatório. (Prioridade:Desejável , Esforço: Médio)

  Fluidez: O sistema deve rodar de forma leve e rápida, sem a ocorrência de travamentos para que não atrapalhe na experiência do usuário. (Prioridade: Essencial, Esforço: Alto).

- Funcionais:

  Listar criptomoedas: Quando selecionada a opção de listar criptomoedas, esta funcionalidade apresenta uma lista de criptomoedas de forma paginada, sendo que cada página contém seis criptos, apresentando informações como, a marca, o código da moeda, o preço atual em dólar, a variação do preço nas últimas 24h , opção de adicionar à carteira e ver informações sobre tal.

Informações sobre uma criptomoeda específica: Ao clicar em uma das criptomoedas listas, esta funcionalidade aparecerá na tela do usuário, informando em uma tela dedicada não apenas as informações previamente conhecidas mas também as mídias sociais em que o ativo pode ser encontrado, um gráfico mostrando a taxa de variação do preço em relação ao dólar nos últimos 7 dias, opções de adicionar e remover o cripto ativo da carteira e informações básicas no market cap, como sua posição, volume de troca e preço.

Gerenciar carteira de cripto ativo: Essa funcionalidade resume um CRUD no banco de dados local (Create, Read, Update, Delete). Ou seja, é responsável por adicionar, remover, mostrar e atualizar o(s) dados(s) da aplicação neste módulo.

3.1.4 Protótipo de baixa fidelidade(mockup)

A confecção de um protótipo da aplicação se deu pela plataforma figma, que possibilita criar interfaces de forma fácil e intuitiva, oferecendo diversas ferramentas que auxiliam de forma eficiente na criação.

Os protótipos são mostrados abaixo:

Figura 4 – Tela de início

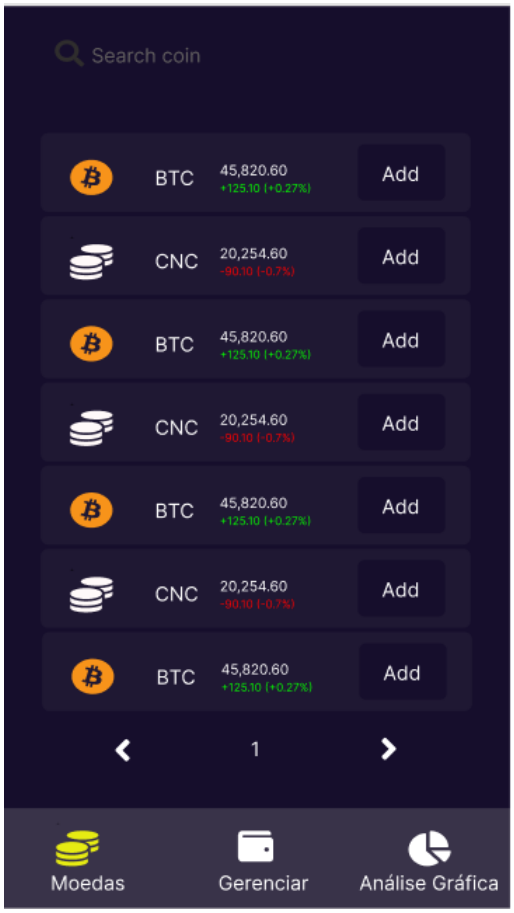


Figura 5 – Lista de moedas

Figura 6 – Informações da moeda

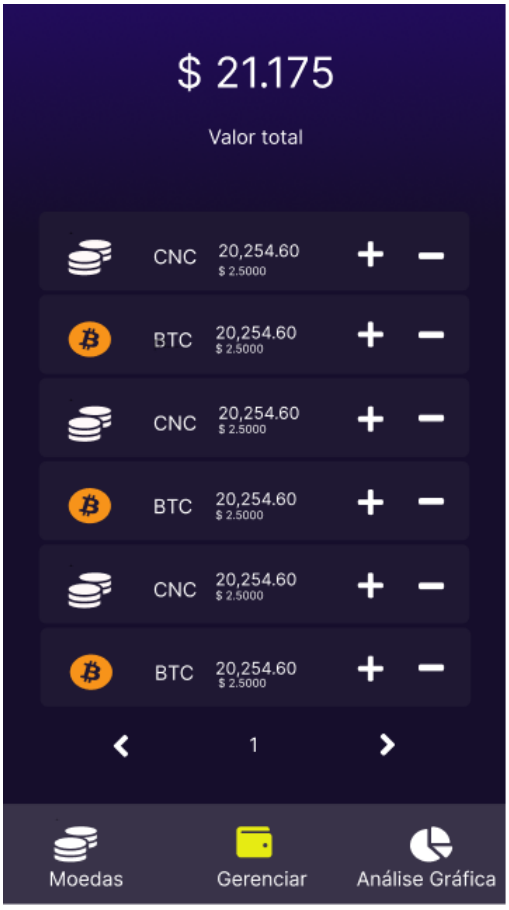


Figura 7 – Carteira ocm ativos

3.1.5 Diagrama UML

A modelagem da aplicação utilizando a linguagem UML para a criação dos diagramas. A UML é tida como uma linguagem unificada de modelagem, é uma linguagem gráfica para a visualização, especificação, construção e documentação de artefatos de sistemas complexos de software (Rumbaugh, Jacobson, Booch, Addison- Wesley, 2005).

3.1.5.1 Diagrama de casos de uso

O Diagrama de Caso de Uso mostra os relacionamentos entre atores e casos de uso em interações com um sistema (FURLAN,1998,p.299).

UC01 - Listar criptomoedas: O aplicativo deverá mostrar ao usuário uma lista vazia caso ocorra algum erro na listagem, caso contrário, mostra uma lista paginada de 6 em 6 itens com um limite de 10 páginas. Se o usuário preferir, pode pesquisar pelo código específico de uma criptomoeda e caso exista, será mostrada em uma lista com apenas o resultado encontrado. Tanto na listagem paginada quanto no resultado da pesquisa, é possível adicionar à carteira e acessar informações específicas da criptomoedas, desde cotação atual, taxa de variação nos últimos 7 dias em forma gráfica ou em porcentagem para as últimas 24h, permitindo nesta tela também adicionar e remover da carteira caso

queira.

UC02 - Manter carteira simulada: Este caso de uso permite ao usuário ver uma lista de criptomoedas em carteira, o preço de cada uma e o agregado em dólar, adicionar e remover o ativo da carteira.

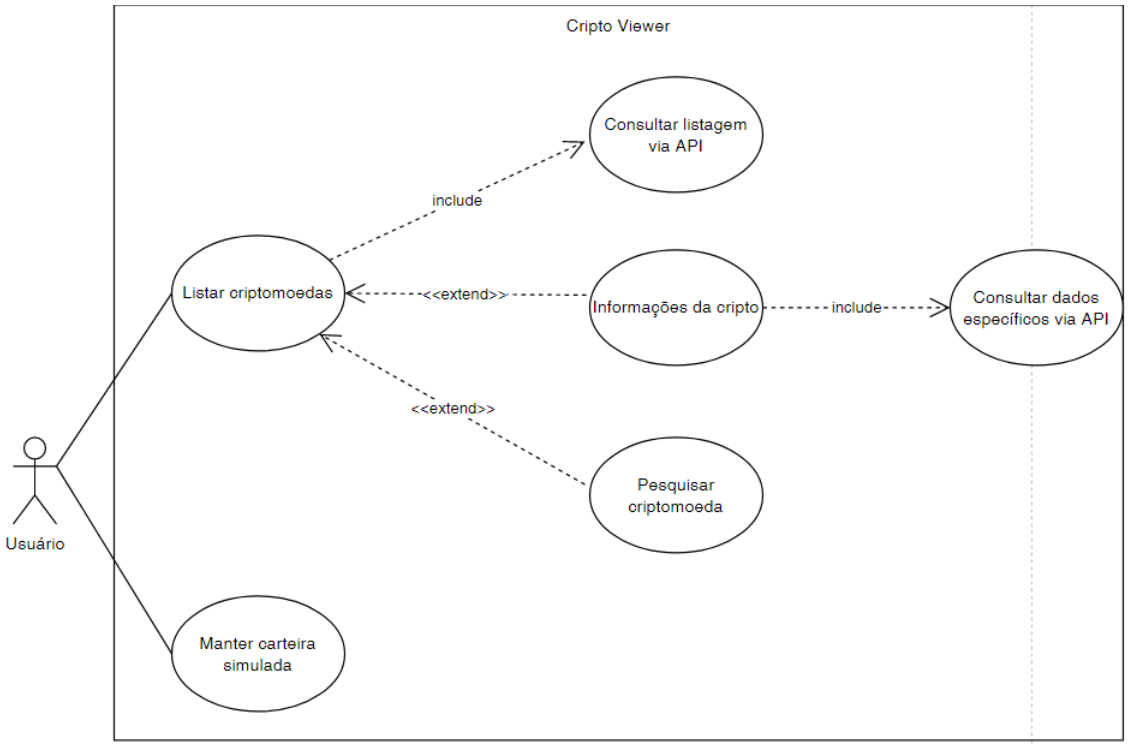


Figura 8 – Diagrama de casos de uso

3.1.5.2 Diagramas de sequência

O diagrama de sequência tem como objetivo mostrar mais detalhadamente as funcionalidades do sistema, especificando a sequência das mensagens trocadas entre os objetos participantes do sistema.

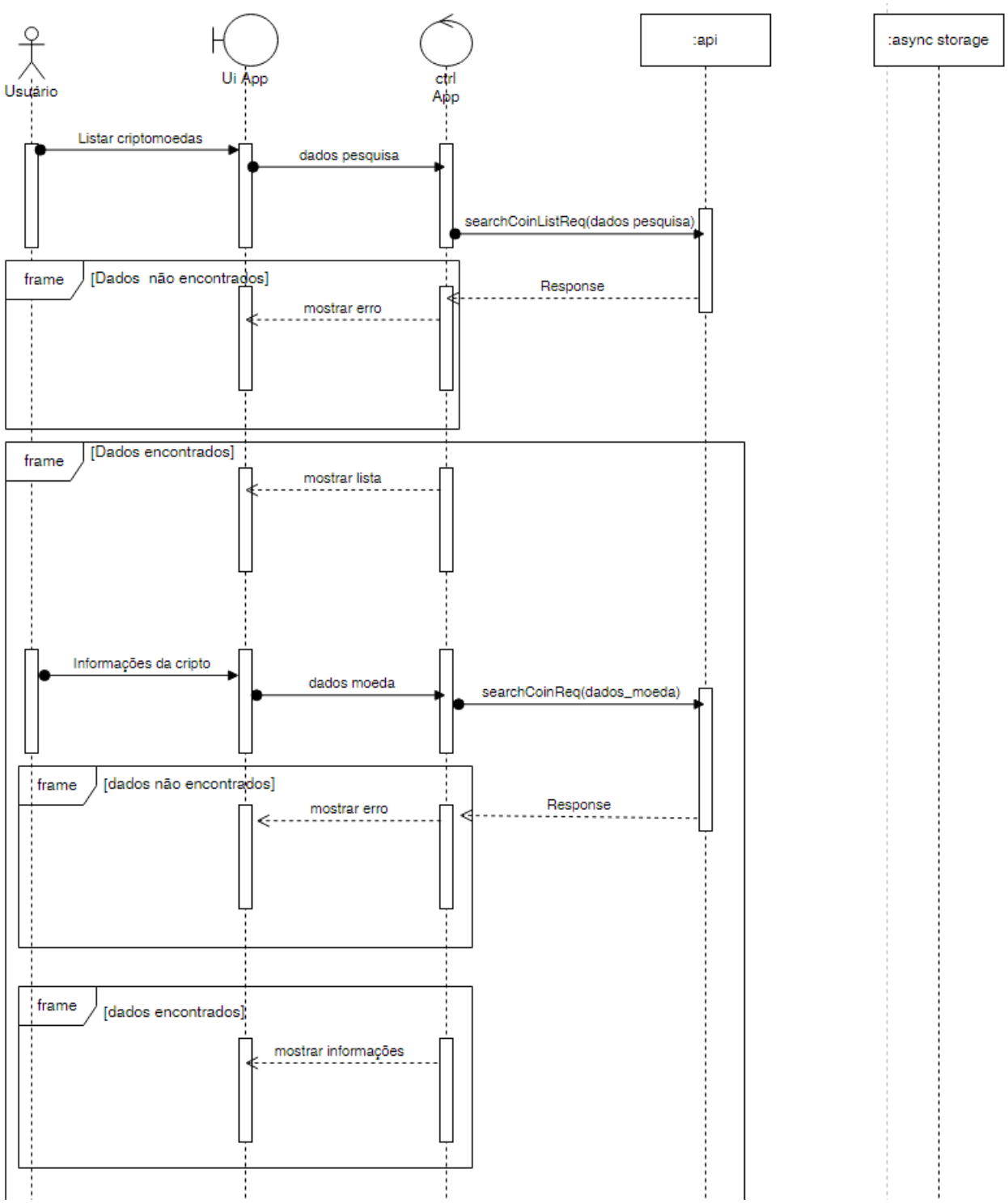


Figura 9 – Diagrama de sequência - Listar criptomoedas

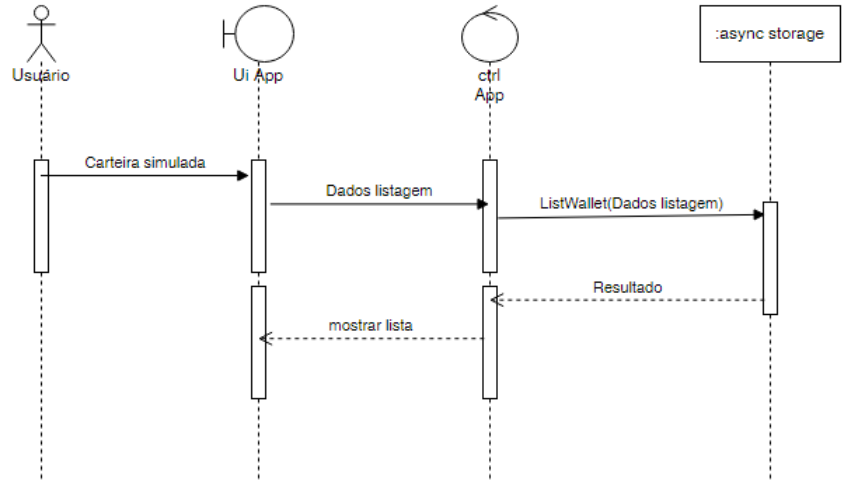


Figura 10 – Diagrama de sequência - Mostrar carteira

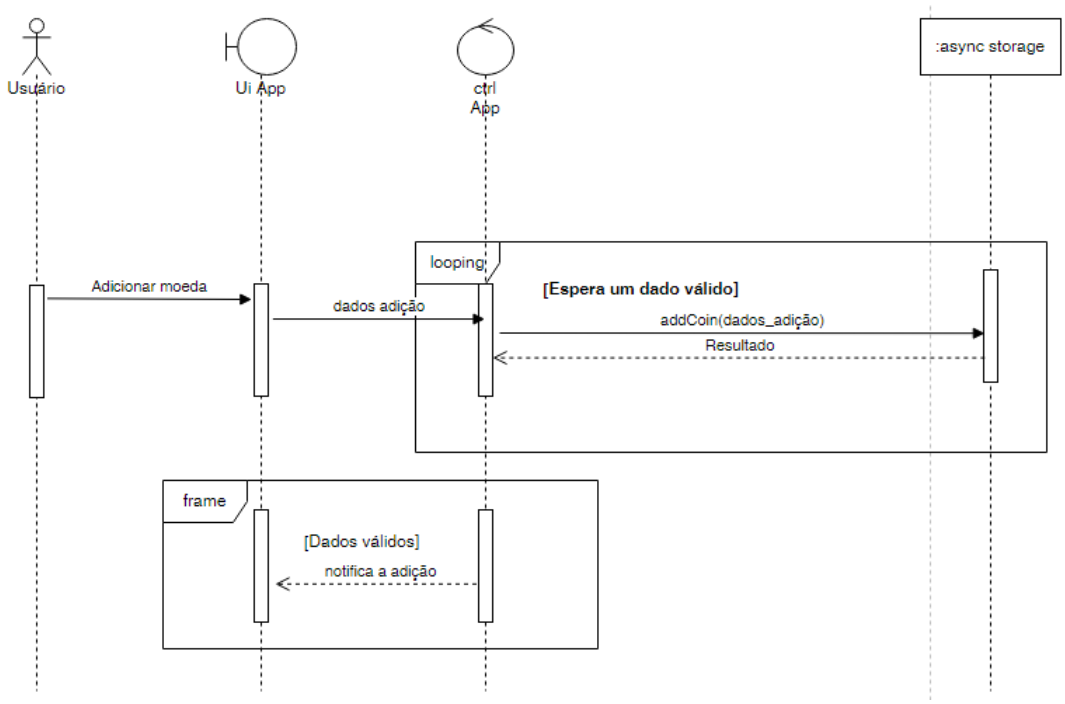


Figura 11 – Diagrama de sequência - Adicionar/Remover ativo

3.1.5.3 Diagrama de classes

Os diagramas de classes a seguir foram desenvolvidos com a finalidade de nortear a implementação das funcionalidades referentes ao Async storage, obtenção de dados da API, formatação do gráfico a ser construído e mídias sociais referentes às moedas.

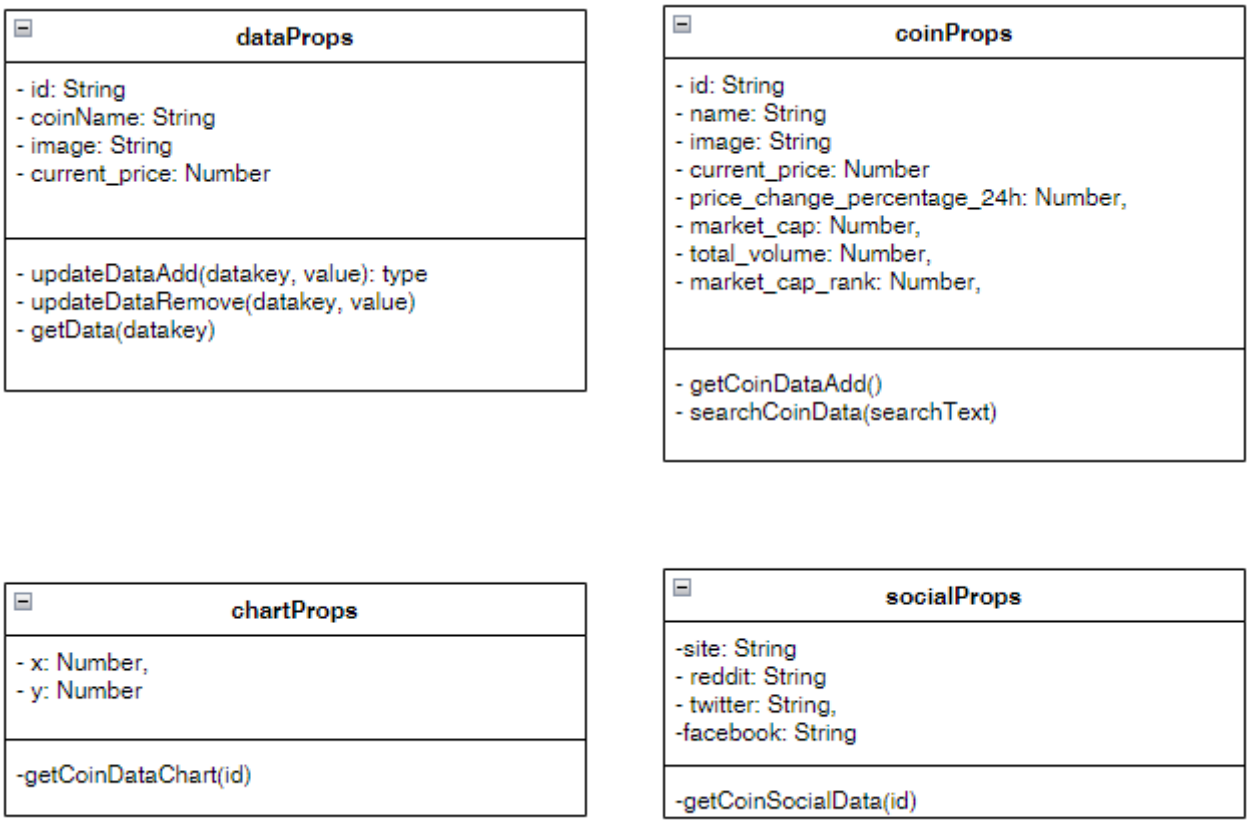


Figura 12 – Diagrama de classes

4 Aplicativo de gerenciamento de carteira de criptomoedas

Nesse capítulo, serão abordadas as características gerais da aplicação, desde quais tecnologias foram utilizadas , linguagem e arquiteturas, até a aplicação como um todo, mostrando funções desenvolvidas, capturas de tela e a API utilizada para obter os dados.

4.1 React Native

É um framework javascript para escrever e renderizar nativamente aplicações mobile para IOS e Android. É baseada no React, biblioteca javascript do facebook para construir interfaces, mas em vez de almejar o browser, visa plataformas mobile (EISENMAN, 2015).

Essa tecnologia foi escolhida pelo fato de ser versátil, em vez de aprender a arquitetura de cada ambiente, Android e IOS, é possível desenvolver algo para ambas com a mesma base de código com ela.

4.1.1 Arquitetura

A arquitetura do React se baseia no novo sistema de renderização **Fabric**, onde a característica principal é unificar a lógica de renderização em C++, melhorando a interoperabilidade com a plataforma nativa(Android, IOS, macOS, Windows). A explicação da arquitetura é dada a seguir:

Render: É o responsável por rendereziar a lógica desenvolvida em react em lógica da plataforma nativa. Ele executa a lógica e cria uma React Element Tree em javascript(Elas descrevem o que deve aparecer na tela). Dessa arvore, é criada uma React Shadow Tree em C++(Consiste em Shadow Nodes, que por sua vez representam componentes react nativos para serem montados).

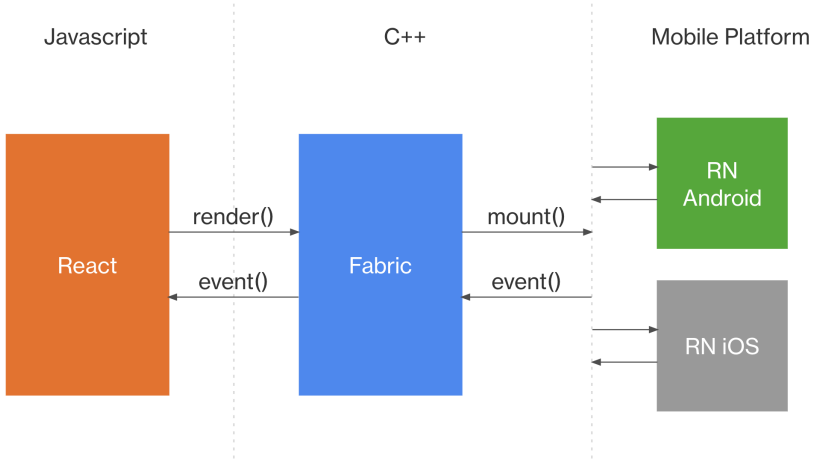
Commit: Depois que a React Shadow Tree está totalmente criada, o render aciona o commit, que promove a React Element Tree e a nova Shadow Tree criada como "proxima arvore"a ser montada.

Mount: A React Shadow Tree, agora com os resultados calculados, são transformados em uma Host View Tree.

Event: São funções como, onLayout, onKeyPress, touc, etc). Que acionam a comunicação entre o React e o render.

A figura 13 ilustra o que foi dito acima:

Figura 13 – Diagrama de implementação



Fonte: https://reactnative.dev/architecture/xplat-implementation, React

4.2 Aplicativo Crypto Viewer

4.2.1 Início

A primeira tela a ser vista da aplicação dá a opção de escolher entre ver uma lista de moedas e a carteira. A figura 14 mostra o código da tela inicial:

```
export function Start({navigation}){
    return (
        <Container>
            <Bitcoin_logo name = {'bitcoin-circle'}></Bitcoin_logo>
            <TitleContainer>
                <Coin_logo name = {'coins'}></Coin_logo>
                <Title>Crypto Viewer</Title>
            </TitleContainer>
            <InfoText>Veja e simule sua carteira de criptoativos</InfoText>

            <ButtonsContainer>
                    <Button
                        title="Moedas"
                        handleFunction = {() => navigation.navigate('CoinList', { name: 'CoinList' })}
                    />
                    <Button
                        title = {'Carteira simulada'}
                        handleFunction ={()=> navigation.navigate('Wallet', {name: 'Wallet'})}
                    />

            </ButtonsContainer>

            <Footer>
                <FooterText>Aplicativo de código aberto</FooterText>
            </Footer>
        </Container>
    )
}
```

Figura 14 – Listagem de criptomoedas

4.2.2 Arquitetura geral do App

Utilizando como base o padrão MVC, foi criada uma arquitetura simples, que mostra a interação entre o view, controller e as bases de dados utilizadas. O MVC é um padrão difundido no mercado onde a divisão em camadas facilita a manutenção do código (BARROS; SILVA; ESPÍNOLA, 2007). As descrições das camadas do cliente servidor se encontram abaixo.

Visão: Camada que contém a interface do usuário, onde é possível interagir com ela, fazendo requisições dos serviços para o servidor.

Controle: É responsável pela execução das regras de negócio da aplicação, controlando o fluxo de informações, quais devem ser recebidas, quais devem ser geradas e para onde devem ir.

Modelo: Responsável por acessar o banco de dados , utiliza modelos para fazer consultas, cálculos e regras de acesso ao banco.

Async storage

Os dados da carteira simulada do usuário deveriam ficar apenas armazenados localmente. Para isso foi utilizada uma biblioteca do React native chamada async storage, que é um sistema de armazenamento de chave-valor não criptografado, assíncrono, persistente e global para o aplicativo. (Figura 15)

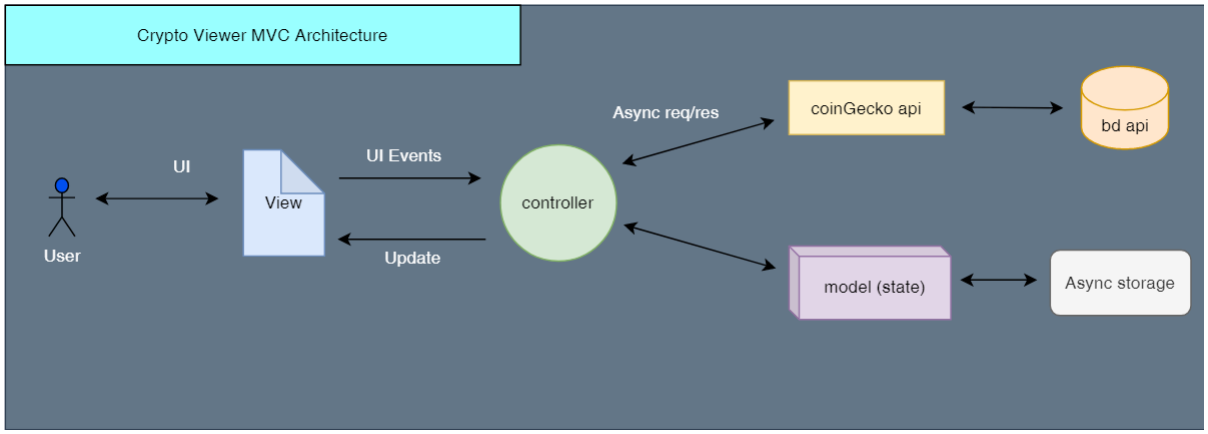


Figura 15 – Arquitetura do app

4.2.3 Listagem

Ao selecionar a opção da lista de moedas, o usuário é redirecionado para outra página, que faz as requisições para a API, recebendo como resultado, caso bem sucedida, um response com a lista de moedas. (Figura 16)

```
const getCoinData = async() =>{
    let temp = page.toString()
    return await fetch('https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&order=market_cap_desc&per_page=6&page='
    +temp+'&sparkline=false')
    .then((response)=> response.json())
    .then((json) =>{
        handleCoinsList(json)

        setIsLoading(false)
    })
    .catch((error) =>{
        console.error(error)
    })

}
```

Figura 16 – Requisição de dados da api

Na página de listagem é possível procurar pelo código de uma moeda específica e receber o resultado. (Figura 17)

```
const searchCoinData = async () =>{
    let url_req = 'https://api.coingecko.com/api/v3/coins/'+ searchText.toLowerCase() +
    '?tickers=false&market_data=true&community_data=false&developer_data=false&sparkline=true'
    setSearchText('');
    setflagreq(0);
    return await fetch(url_req)
    .then((response)=> response.json())
    .then((json)=>{
        const result:coinProps = {
            id: json.id,
            symbol: json.symbol,
            name: json.name,
            image: json.image.small,
            current_price: json.market_data.current_price.usd,
            price_change_percentage_24h: json.market_data.market_cap_change_percentage_24h,
            market_cap: json.market_cap,
            total_volume: json.total_volume,
            market_cap_rank: json.market_cap_rank,


        }
        handleCoinsList([result])

    })
    .catch((error)=>{
        Alert.alert('Moeda não encontrada')
    })
}
```

Figura 17 – Requisição de dados de uma moeda específica

4.2.4 Informações detalhadas

Ao selecionar uma moeda para ver informações mais específicas, é feito um redirecionamento para outra página, que faz requisições para criar um gráfico e mostrar as redes sociais dela. Os dados do gráfica precisam, após recebidos, ser formatados para que seja possível envia-los como entrada da biblioteca que renderiza. (Figura 18)

```
const getCoinDataChart = async() =>{
    let temp = id.toString().toLowerCase();
    let days = 7;
    return await fetch('https://api.coingecko.com/api/v3/coins/'
    + temp +'/market_chart?vs_currency=usd&days='+days+'&interval=daily')
    .then((response)=> response.json())
    .then((json) =>{
        let temp = json.prices
        let chartFormated:chartProps[] = []
        for(let i = 0 ; i < days; i++){
            chartFormated.push({
                x:i,
                y: parseInt(temp[i][1])
            })

        }

        setChartData(chartFormated)
        setIsloading(false)

    })
    .catch((error) =>{
        console.error(error)
    })

}
```

Figura 18 – requisição e formatação gráfica

Na mesma seleção, também são coletados dados sobre quais redes sociais a moeda está inserida e possui um link para sua página. (Figura 19)

```
const getCoinSocialData = async () =>{
    let temp = id.toString().toLowerCase();
    return await fetch('https://api.coingecko.com/api/v3/coins/'
    +temp+
    '?localization=false&tickers=false&market_data=false&community_data=true&developer_data=false&sparkline=false')
    .then((response)=> response.json())
    .then((json)=>{
        const data:socialProps = {
            site : '',
            reddit : '',
            twitter : '',
            facebook :''
        }
        if (json.links.hasOwnProperty('homepage') ){
            data.reddit = json.links.homepage[0]
        }
        if (json.links.hasOwnProperty('subreddit_url') ){
            data.reddit = json.links.subreddit_url
        }
        if (json.links.hasOwnProperty('facebook_username') ){
            data.facebook = json.links.facebook_username
        }
        if (json.links.hasOwnProperty('twitter_screen_name') ){
            data.twitter = json.links.twitter_screen_name
        }

        setSocialUrls(data)
    })
    .catch((error)=>{
        console.error(error)
    })
}
```

Figura 19 – Requisição de dados de mídias sociais

:

4.2.5 Carteira

Ao acessar a área de carteira, a página irá pegar os dados do que já foram armazenados e os mostrará na tela cada ativo em posse, o preço individual e o agregado. (Figura 20)

```
const handleCoinsData = async () =>{
    return await getData(datakey)
    .then((data)=>{
        const dataResolved = data?JSON.parse(data): []
        if(dataResolved.length > 0){

            setCoins(dataResolved)

        }setcoinDataloading(false)
    })

}
const getCurrentUSDvalue = () =>{

    let total = 0;

    coins.map((item)=>{
        total += ((item.current_price) * (item.valueOwned));
    })
    settotalUSDvalue(Number(total.toFixed(1)))
    setIsLoading(false)
}
```

Figura 20 – Carteira

A operação de inserção/remoção de um ativo, primeiramente checa se já existe um valor armazenado, caso exista e o valor de inserção/remoção seja válido, a operação é realizada com sucesso. Caso o valor subtraido zere o ativo na carteira, este é removido. (Figura 21, 22 e 23)

```
export const getData = async (key:string) => {
    // get Data from Storage
    try {
      const data = await AsyncStorage.getItem(key);
      if (data !== null) {

        return data;
      }
    } catch (error) {
      console.log(error);

    }
    return null;
  };

export const removeData = async (datakey:string) =>{
  await AsyncStorage.removeItem(datakey)
}
```

Figura 21 – Receber dados dos ativo

```
export const updateDataAdd = async (datakey:string, value:dataProps) => {

    try {

      const data = await AsyncStorage.getItem(datakey);
      const currentData = data ? JSON.parse(data) : [];

      let alreadyIn = false
      const result = currentData.map((item:dataProps)=>{
        if (item.coinName === value.coinName){
          item.valueOwned += value.valueOwned;
          alreadyIn = true
        }
        return item
      })
      let dataFormatted = []

      if(alreadyIn){
        await AsyncStorage.setItem(datakey, JSON.stringify(result))
        Alert.alert('Valor adicionado')
        return true
      }else{
        dataFormatted = [
          ...currentData,
          value
        ]
        await AsyncStorage.setItem(datakey, JSON.stringify(dataFormatted))
        Alert.alert('Valor adicionado')
        return true
      }
```

Figura 22 – Adicionar/Atualizar Ativo

```
export const updateDataRemove = async (datakey:string, value:dataProps) => {
    try {
      const data = await AsyncStorage.getItem(datakey);
      const currentData = data ? JSON.parse(data) : [];
      let alreadyIn = false;
      let toRemove = false;
      const result = currentData.map((item:dataProps)=>{
        if ((item.coinName === value.coinName) && (item.valueOwned-value.valueOwned) > 0){
          item.valueOwned -= value.valueOwned;
          alreadyIn = true;
        }else if((item.valueOwned-value.valueOwned) === 0){
          alreadyIn = true;
          toRemove = true;
        }
        return item
      })
      let dataFormatted = [];

      if(alreadyIn && !toRemove){
        await AsyncStorage.setItem(datakey, JSON.stringify(result))
        Alert.alert('Valor removido')
        return true
      }else if(alreadyIn && toRemove){
        dataFormatted = result.filter((item:dataProps)=>{
          return item.coinName != value.coinName;
        })
        await AsyncStorage.setItem(datakey, JSON.stringify(dataFormatted))
        Alert.alert('Valor removido')
        return true
      }else{
        Alert.alert('Valor invalido2')
        return false
      }
    } catch (e) {
      return false
    }
  }
```

Figura 23 – Remover/Atualizar Ativo

4.3 API REST

Como foi mostrado na arquitetura anteriormente, a API utilizada para aquisição de informações sobre criptomoedas foi a da (COINGECKO, 2020) pois oferece de forma clara e objetiva os dados necessários para a construção da aplicação.

API REST, também chamada de API RESTful, é uma interface de programação de aplicações (API ou API web) que está em conformidade com as restrições do estilo de arquitetura REST, permitindo a interação com serviços web RESTful. REST é a sigla em inglês para "Representational State Transfer", que em português significa tansferência de estado representacional. Essa arquitetura foi criada pelo cientista da computação Roy Fielding (REDHAT, 2020). A figura 24 mostra a documentação da API:

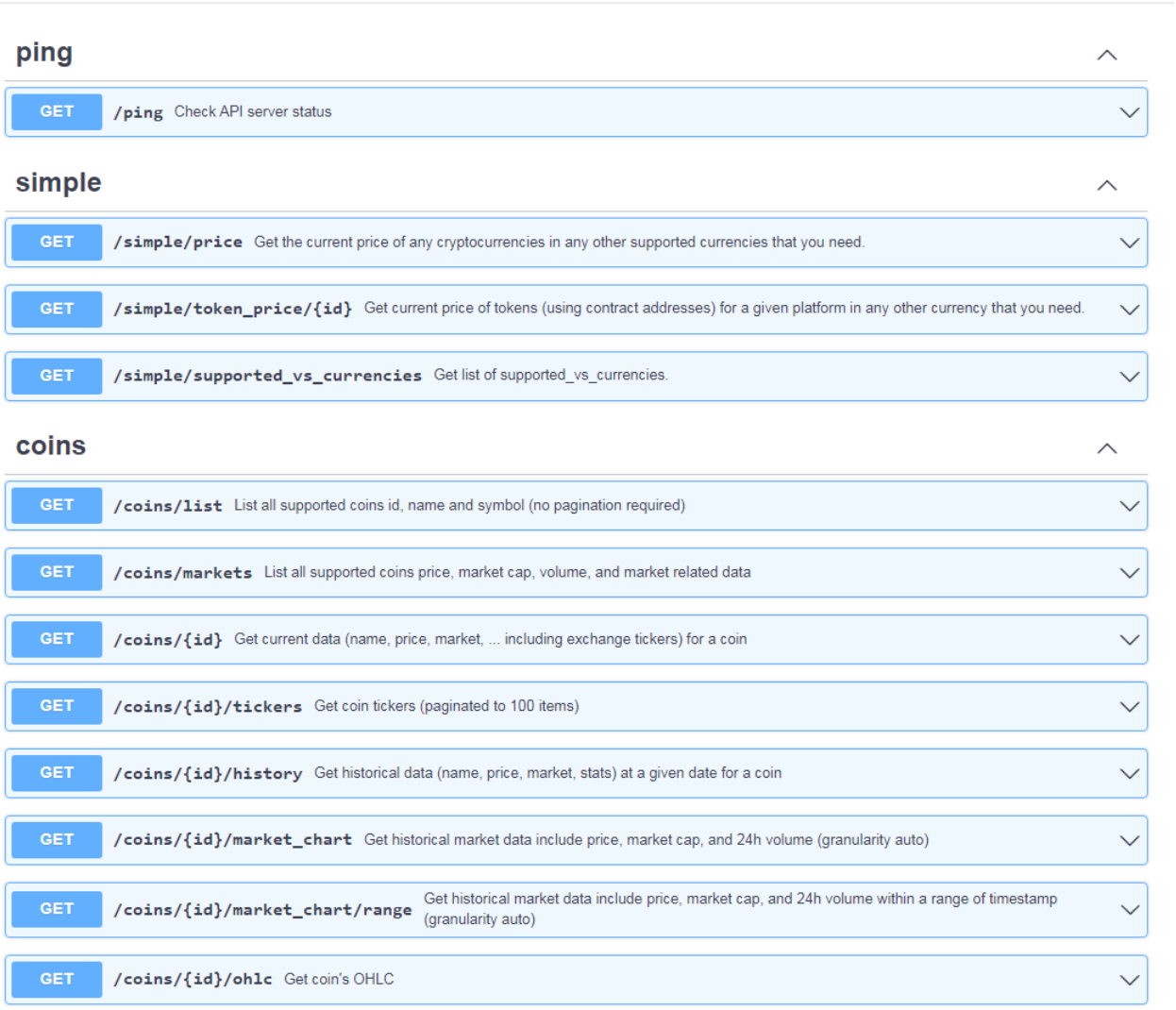


Figura 24 – CoinGecko - Documentação

Ao selecionar uma das opções, um série de parâmetros a serem configurados.

Abaixo foi mostrada a configuração utilizada na aplicação para o retorno da lista de criptomoedas.As configurações feitas foram retornar as 6 primeiras moedas ranqueadas no market cap com o valor em USD, ao alterar a pagina para 2 por exemplo, serão enviadas as moedas ranqueadas entre 6 e 12 e assim por diante. (Figura 25 e 26)

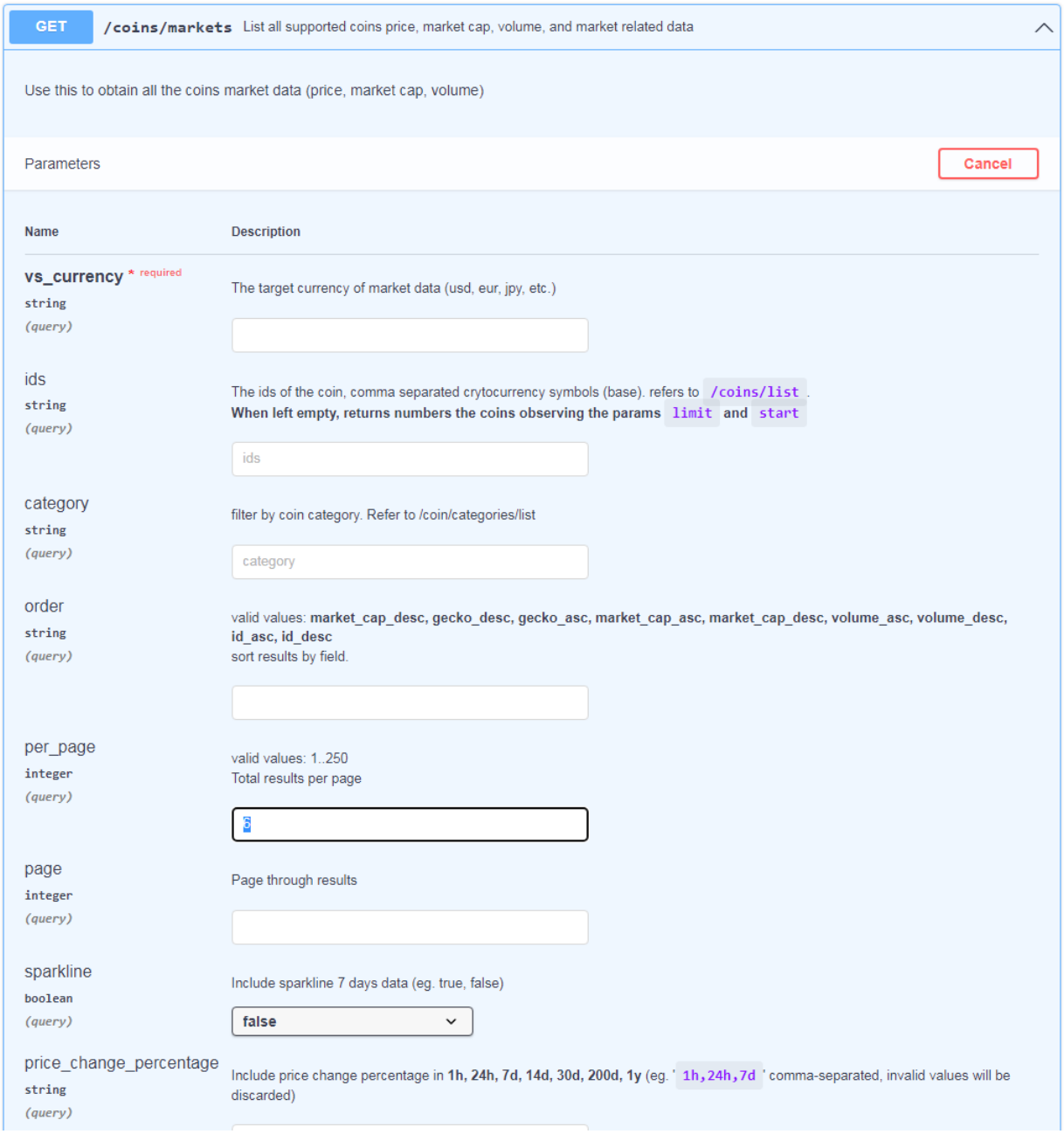


Figura 25 – CoinGecko - Parametros para busca

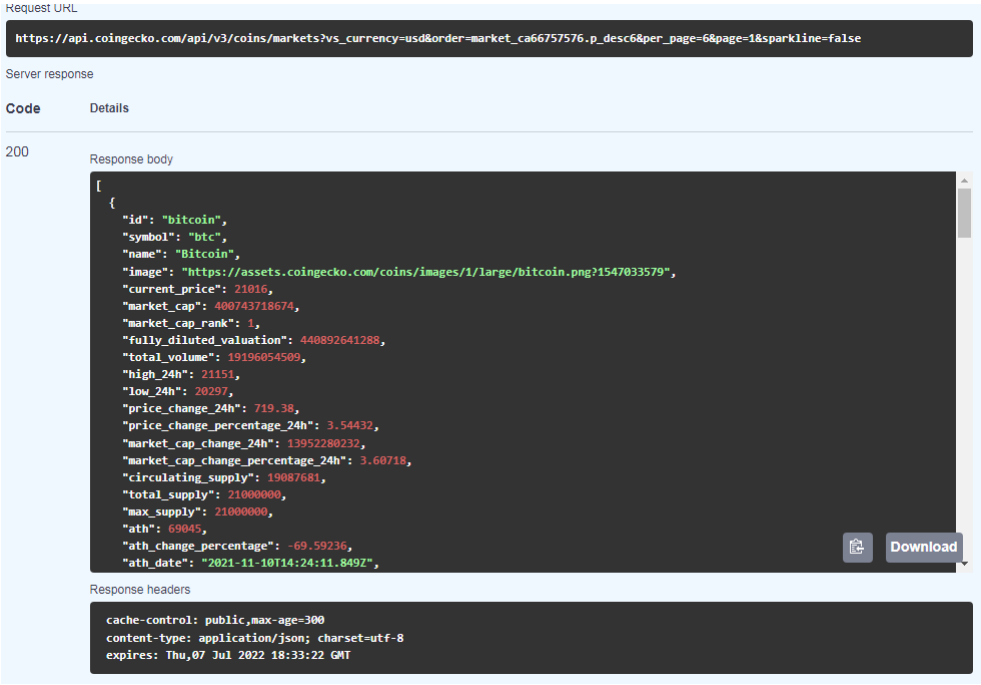


Figura 26 – CoinGecko - Response

4.3.1 Aplicativo em funcionamento

Nesta seção, serão apresentadas capturas de tela feitas do resultado final do aplicativo, como tela inicial, listagem de moeda etc. (Figura 27, 28, 29 e 30)

Figura 27 – Tela de início

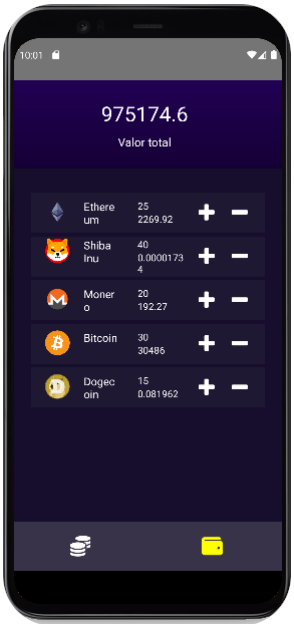


Figura 28 – Carteira simulada

Foi possível implementar os protótipos(mockup) com um bom grau de fidelidade, certas alterações se mostraram plausíveis conforme o andamento do projeto, como remover

um elemento da nav bar, visto que seria um redundância com o gráfico apresentado na captura de tela que mostra as informações específicas.

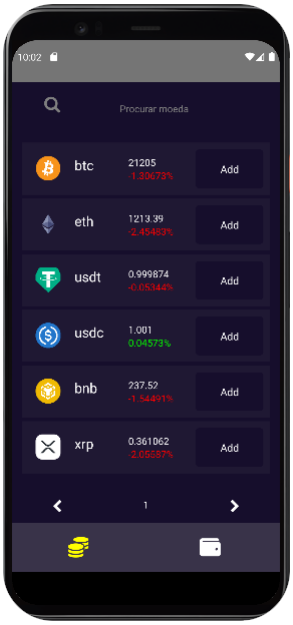


Figura 29 – Lista de moedas

Figura 30 – Informações específicas

5 Conclusão

Este trabalho abordou os conceitos de criptomoeda e blockchain e como esses conceitos provocaram uma grande mudança desde suaS criações como eles podem ser utilizado nos dias atuais para garantir transações rapidas e seguras, além de outras aplicações. As criptomoedas apesar de suas enormes vantagens e comodidade, ainda são pouco utilizadas pela maioria dos usuários dos meios digitais. Pouco conhecimento sobre esse mundo gera um sentimento de desconfiança por meio dos usuários comuns, que acabam não utilizando e não confiando na segurança das criptomoedas e blockchain, deixando de aproveitar da comodidade, rapidez e facilidade na hora de transacionar ativos financeiros. Este trabalho tem por objetivo implantar um Aplicativel móvel para visualizar criptomoedas do mercado de modo simples, prático, rápido e de fácil acesso e utilização do ponto de vista da experiência dos usuários, visando facilitar o processo e providencias informações a cerca das criptomoedas bem como difundir os conceitos de Criptomoedas de modo a quebrar o paradigma e a desconfiança que ainda se tem sobre o mundo cripto. Vale lembrar que tal aplicação não foi testada com usuários e portanto, não foi possível apresentar resultados práticos de métricas de avaliação.

Referências

AMMOUS, S. Blockchain technology: What is it good for? *Available at SSRN 2832751*, 2016. Citado 3 vezes nas páginas 16, 17 e 19.

BARROS, T.; SILVA, M.; ESPÍNOLA, E. State mvc: Estendendo o padrão mvc para uso no desenvolvimento de aplicações para dispositivos móveis. In: *Sexta Conferência Latino-Americana em Linguagens de Padrões para Programação*. [S.l.: s.n.], 2007. Citado na página 33.

BRAGA, A. M.; MARINO, F. C. H.; SANTOS, R. R. dos. Segurança de aplicaçoes blockchain além das criptomoedas. *Sociedade Brasileira de Computação*, 2017. Citado na página 16.

COINGECKO. *Documentação*. 2020. Url https://www.coingecko.com/pt/api/documentation. Citado na página 41.

COINMARKETCAP. *charts*. 2022. Urlhttps://coinmarketcap.com/charts/. Citado na página 22.

DAVIDORTINAU; DAVIDBRITCH; DCTHEGEEK; CONCEPTDEV; VIVEKNATARAJAN; TIMEYOUTAKEIT. *O ciclo de vida do desenvolvimento de software móvel*. 2022. Urlhttps://docs.microsoft.com/pt-br/xamarin/cross-platform/get-started/introduction-to-mobile-sdlc. Citado na página 23.

EISENMAN, B. *Learning react native: Building native mobile apps with JavaScript*. [S.l.]: "O'Reilly Media, Inc.", 2015. Citado na página 31.

ETHEREUM. *INTRODUCTION TO SMART CONTRACTS*. 2022. Urlhttps://ethereum.org/en/developers/docs/smart-contracts/. Citado na página 17.

FANG, F.; VENTRE, C.; BASIOS, M.; KANTHAN, L.; MARTINEZ-REGO, D.; WU, F.; LI, L. Cryptocurrency trading: a comprehensive survey. *Financial Innovation*, SpringerOpen, v. 8, n. 1, p. 1–59, 2022. Citado na página 20.

FARELL, R. An analysis of the cryptocurrency industry. 2015. Citado na página 19.

FORBES. *More Than 600 Million Stolen In Ethereum And Other Cryptocurrencies—Marking One Of Crypto's Biggest Hacks Ever*. 2021. Url https://www.forbes.com/sites/jonathanponciano/2021/08/10/more-than-600-million-stolen-in-ethereum-and-other-cryptocurrencies-marking-one-of-cryptos-biggest-hacks-ever/?sh=63680b87f623. Citado na página 21.

LAMPORT, L. The part-time parliament. In: *Concurrency: the Works of Leslie Lamport*. [S.l.: s.n.], 2019. p. 277–317. Citado na página 14.

LIGHTNINGNETWORK. *Documentação*. 2020. Url http://lightning.network/how-it-works/. Citado na página 21.

NAKAMOTO, S. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, p. 21260, 2008. Citado na página 14.

REDHAT. *O que é API REST?* 2020. Urlhttps://www.redhat.com/pt-br/topics/api/what-is-a-rest-api. Citado na página 41.

SHETH, H.; DATTANI, J. Overview of blockchain technology. *Asian Journal For Convergence In Technology (AJCT) ISSN-2350-1146*, 2019. Citado 2 vezes nas páginas 14 e 15.

VEJAČKA, M. Basic aspects of cryptocurrencies. *Journal of Economy, Business and Financing*, v. 2, n. 2, p. 75–83, 2014. Citado na página 21.

YAGA, D.; MELL, P.; ROBY, N.; SCARFONE, K. Blockchain technology overview. *arXiv preprint arXiv:1906.11078*, 2019. Citado 3 vezes nas páginas 15, 16 e 17.