UNIP - UNIVERSIDADE PAULISTA

Projeto Integrado Multidisciplinar

Cursos Superiores De Tecnologia

**Erick Oliveira Abreu Santos - RA F09392-1**

**Fabio Emanoel Dias dos Reis - RA F079IB-4**

**Gustavo da Silva Ferreira - RA T95554-8**

**Maria Lucia da Silva - RA F064EI-5**

**Matheus de Andrade - RA N51061-7**

**Otavio Teixeira Assis da Silva - RA N5058H-4**

**DESENVOLVIMENTO E ANÁLISE DE UM SISTEMA DE CONTROLE OPERACIONAL DE UMA EMPRESA ESPECIALISTA EM TRANSAÇÕES COM BLOCKCHAIN**

**SÃO PAULO**

**2020**

**Erick Oliveira Abreu Santos - RA F09392-1**

**Fabio Emanoel Dias dos Reis - RA F079IB-4**

**Gustavo da Silva Ferreira - RA T95554-8**

**Maria Lucia da Silva - RA F064EI-5**

**Matheus de Andrade - RA N51061-7**

**Otavio Teixeira Assis da Silva - RA N5058H-4**

**DESENVOLVIMENTO E ANÁLISE DE UM SISTEMA DE CONTROLE OPERACIONAL DE UMA EMPRESA ESPECIALISTA EM TRANSAÇÕES COM BLOCKCHAIN**

Trabalho de conclusão de curso para obtenção do título de graduação em Análise e Desenvolvimento de Sistemas apresentado à Universidade Paulista – UNIP.

Orientadora:

Profa. Aparecida de Fátima Castello Rosa

**SÃO PAULO**

**2020**

**Erick Oliveira Abreu Santos - RA F09392-1**

**Fabio Emanoel Dias dos Reis - RA F079IB-4**

**Gustavo da Silva Ferreira - RA T95554-8**

**Maria Lucia da Silva - RA F064EI-5**

**Matheus de Andrade - RA N51061-7**

**Otavio Teixeira Assis da Silva - RA N5058H-4**

**DESENVOLVIMENTO E ANÁLISE DE UM SISTEMA DE CONTROLE OPERACIONAL DE UMA EMPRESA ESPECIALISTA EM TRANSAÇÕES COM BLOCKCHAIN**

Trabalho de conclusão de curso para obtenção do título de graduação em Análise e Desenvolvimento de Sistemas apresentado à Universidade Paulista – UNIP.

Orientadora:

Profa. Aparecida de Fátima Castello Rosa

Aprovado em:

BANCA EXAMINADORA

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_/\_\_\_

Professora: Aparecida de Fátima Castello Rosa

Universidade Paulista – UNIP

**SÃO PAULO**

**2020**

**RESUMO**

O Objetivo desse trabalho é desenvolver um sistema de gerenciamento e controle para uma empresa especialista em *Blockchain,* tecnologia que surgiu por volta de 2008, junto com o Bitcoin. Porém, sua disponibilização ao público só foi no ano de 2009. Esse sistema se trata de uma validação de informações por blocos, neste PIM foi desenvolvido, e aplicado os conhecimentos adquiridos através das aulas e atividades propostas, tais como ferramentas computacionais para modelagem de negócios, técnicas usadas pela UML (*Linguagem de modelagem unificada*), foi abordado sobre requisitos funcionais e não funcionais usabilidade e normas e por fim, executado o trabalho em equipe com divisão de tarefas do projeto.

As disciplinas utilizadas como base de estudos foram: Gerenciamento de Projeto de Software, Desenvolvimento de Software para Internet, Programação Orientada a Objetos II, Projeto de Sistemas Orientado a Objetos, Tópicos Especiais de Programação Orientada a Objetos e as complementares são Empreendedorismo e Gestão da Qualidade.

Foi elaborado um sistema interno de Gerenciamento das operações na empresa, para o controle de informações de seus clientes, como cadastro de clientes e usuários, relatórios executivos para tomada de decisões. Esse sistema contém versões Web, Mobile e Desktop.

Palavras chaves: Bitcoin, modelagem, sistema, TI.

**ABSTRACT**

The objective of this work is to develop a management and control system for a company specializing in Blockchain, technology that emerged around 2008, together with Bitcoin. However, it was made available to the public only in 2009. This system is a validation of information by blocks, in this PIM it was developed, and the knowledge acquired through the classes and activities proposed, such as computational tools for business modeling, was applied. , techniques used by UML (Unified Modeling Language), was addressed on functional and non-functional requirements, usability and standards and finally, team work was carried out with the division of project tasks.

The disciplines used as the basis of studies were: Software Project Management, Internet Software Development, Object Oriented Programming II, Object Oriented Systems Design, Special Object-Oriented Programming Topics and the complementary ones are Entrepreneurship and Quality Management.

An internal operations management system was developed in the company, to control the information of its customers, such as customer and user registration, executive reports for decision making. This system contains Web, Mobile and Desktop versions.

Keywords: Bitcoin, modeling, system, IT.

**SUMÁRIO**

[1.INTRODUÇÃO 7](#_Toc57421383)

[2.GERENCIAMENTO DE PROJETO DE SOFTWARE 10](#_Toc57421384)

[2.1.Definir atividades 13](#_Toc57421385)

[2.2.Estimar duração das atividades 13](#_Toc57421386)

[2.3.Sequenciar atividades 13](#_Toc57421387)

[2.4.Estimar recursos 13](#_Toc57421388)

[2.5.Desenvolver o cronograma 13](#_Toc57421389)

[3. DEFINIÇÃO DO NEGÓCIO 14](#_Toc57421390)

[3.1. Outras Soluções Disponíveis 16](#_Toc57421391)

[4. REFERENCIAL TEÓRICO 16](#_Toc57421392)

[4.1 - Arquitetura MVC 16](#_Toc57421393)

[4.1.1 - Camada Módel 17](#_Toc57421394)

[4.1.2 - Camada View 18](#_Toc57421395)

[4.1.3 - Camada Controller 18](#_Toc57421396)

[5. FUNÇÕES DO NEGÓCIO 18](#_Toc57421397)

[6. REQUISITOS 32](#_Toc57421398)

[7. DIAGRAMA DE SEQUÊNCIA 34](#_Toc57421399)

[8. DIAGRAMA DE CLASSE 35](#_Toc57421400)

[9. DIAGRAMA ENTIDADE RELACIONAMENTO 35](#_Toc57421401)

[10. O SISTEMA 40](#_Toc57421402)

[10.1. Protótipo De Telas 40](#_Toc57421403)

[10.1.1 Versão Mobile 40](#_Toc57421404)

[10.1.2 – Versão Desktop 44](#_Toc57421405)

[10.1.3 Versão Web 45](#_Toc57421406)

[11. DIAGRAMA DE IMPLANTAÇÃO 47](#_Toc57421407)

[12. RELATORIO DE CLIENTES 51](#_Toc57421408)

[12.1 Tipos de Bloqueio 53](#_Toc57421409)

[12.2. Relatório De Cobrança Por Serviços Prestados 53](#_Toc57421410)

[13. EMPREENDEDORISMO 55](#_Toc57421411)

[13.1 A Tomada De Decisâo 55](#_Toc57421412)

[13.2. Empreendedor – Empreendedorismo 57](#_Toc57421413)

[14. ESTRATÉGIA DE NEGÓCIO 58](#_Toc57421414)

[15. GESTÃO DE QUALIDADE 59](#_Toc57421415)

[15.1 . Ferramentas De Qualidade 60](#_Toc57421416)

[15.2 Gráfico De Pareto 61](#_Toc57421417)

[15.3 Diagrama De Causa E Efeito De Ishikawa 61](#_Toc57421418)

[15.4 Planejamento Pdca 62](#_Toc57421419)

[15.5 Normas E Iso 62](#_Toc57421420)

[16. CONCLUSÃO 63](#_Toc57421421)

[17. REFERÊNCIAS BIBLIOGRÁFICAS 65](#_Toc57421422)

[APÊNDICE 1 66](#_Toc57421423)

[APÊNDICE 2 80](#_Toc57421424)

[APÊNDICE 3 81](#_Toc57421425)

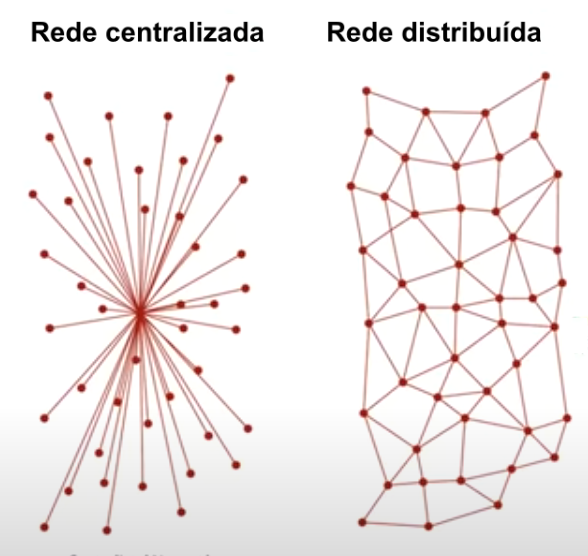
[APÊNDICE 4 85](#_Toc57421426)

[APÊNDICE 5 89](#_Toc57421427)

# 1.INTRODUÇÃO

A tecnologia blockchain surgiu em meados de 2008 junto com o bitcoin, mas só em 2009 foi disponibilizado ao público, blockchain se trata de um sistema de validação de informações por blocos, esse sistema faz a validação de forma distribuída, em um sistema distribuído todos os pontos estão conectados entre si e não existe um ponto central, não tem uma instituição que faz o controle ou registro de tudo, nesse sistema distribuído todos os agentes dessa rede tem a responsabilidade verificar as informações nela inseridas, diferente dos bancos por exemplo que utilizam uma validação centralizada, partindo de um ponto focal para a validação de informações, um órgão regulador e terceiras partes intermediarias.

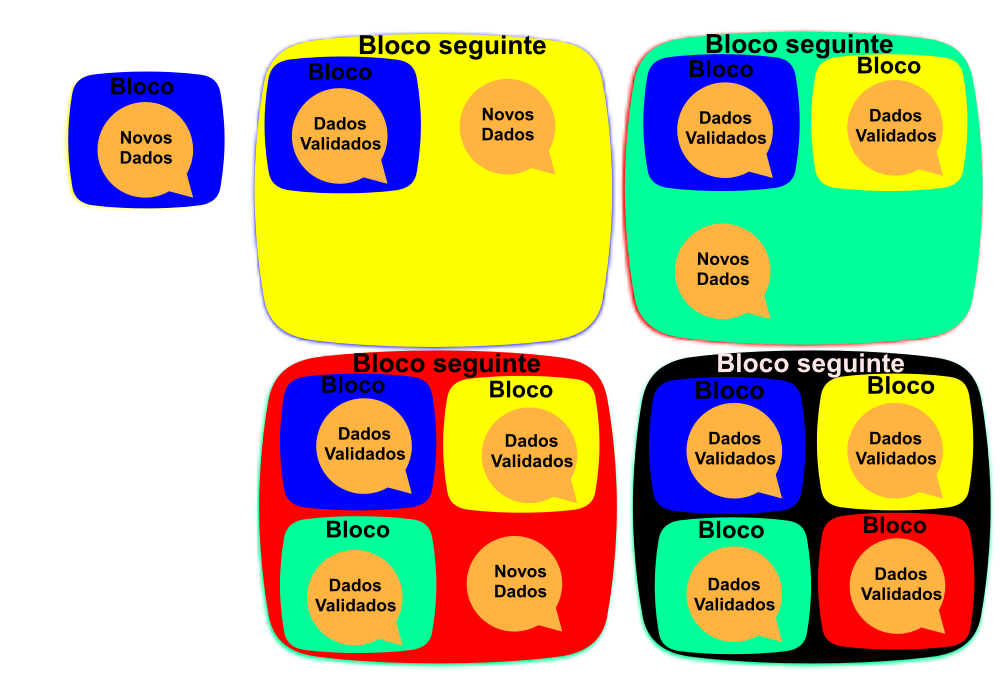
Exemplo de rede centralizada e rede distribuída:



Fonte: [www.infowester.com](http://www.infowester.com)

Blockchain se trata de uma cadeia de blocos que são validados e revalidados, sendo que cada bloco dessa cadeia possui uma quantidade de informações, podendo por exemplo no uso com bitcoins armazenar transações financeiras, cada bloco armazena um histórico dessas transações através de um hash, que é a identidade desse bloco, comprovando que ele é único, esse bloco também armazena outros hash dos blocos anteriores, para um novo bloco ser adicionados existe um processo de mineração, que é o processo de validação de um bloco novo, vários computadores espalhados no mundo todo estão empenados em resolver problemas matemáticos e quando este problema é solucionado, a informação é disseminada entre os outros computadores envolvidos, que validam a informação e no caso de ser válida, um novo bloco é criado e pode ser armazenado na rede de blockchain e quando um novo bloco é armazenado ele é validado e também valida todos os blocos anteriores.

Figura 1 - Exemplo de inclusão e validação de dados no blockchain



Fonte: o autor

Desta forma quanto mais antigo o bloco mais seguro as informações pois já foram validadas diversas vezes, no caso dos bitcoins a cada dez minutos um novo bloco é adicionado ao sistema, todas as vezes que um bloco é adicionado todos os membros da rede recebem esse bloco e adicionam no seu blockchain, esses membros são conhecidos com “full nodes”, que possuem todo o histórico de transações do blockchain, eles são responsáveis por fazer essa validação.

Os termos dessa validação são dados através de “*smart contracts”, ou contratos inteligentes, que se trata de contratos auto executáveis mediante a uma ação ou circunstâncias, um código que armazena parâmetros e quando são atingidos executa uma função, as pessoas e objetos podem ser representadas dentro dos smart contracts através de chaves públicas, sendo sua identificação e a ela atrelada termos que compõem este contrato auto executável.*

Outras formas de aplicação de blockchain pode ser o controle de criptoativos e tokens que se trata de ilustração digital de ativos físicos ou digitais, ou compartilhamento de dados, garantindo que estejam atualizados, também podem ser usado como plataforma de workflows descentralizados para digitalização e automação de processos.

Nesse projeto temos o objetivo geral o detalhamento da especificação do sistema de gerenciamento e controle administrativo das operações realizadas pelos clientes. De forma a atingir esse objetivo, temos os seguintes objetivos específicos que irão nortear o desenvolvimento do trabalho:

* 1. Definição das regras de negócio e processos;
  2. Prototipação das principais telas;
  3. Delimitação dos requisitos funcionais e não funcionais;
  4. Criação dos casos de uso e fluxos;
  5. E por fim, manuais de instalação, cobrança, entre outros

Devemos levar em conta principalmente que o sistema poderá ser utilizado por qualquer perfil de pessoa, inclusive pessoas idosas ou com deficiências, dessa forma faz-se essencial que ele seja fácil de usar, intuitivo e que procure, dentro do possível, permitir que essas pessoas consigam utilizar a aplicação.

Como a aplicação também será utilizada por pessoas físicas, é essencial que esteja de acordo com a LGPD (Lei geral de proteção de dados), que irá entrar em vigor entre agosto de 2020 e janeiro de 2021. Dessa forma a lei deixa claro quais dados serão coletados, por quanto tempo serão mantidos e qual será seu uso.

# 2.**GERENCIAMENTO DE PROJETO DE SOFTWARE**

O gerenciamento de projeto de software consiste em uma parte muito importante do desenvolvimento de sistemas, pois através dele prevenimos erros, criamos ações para correções e mantemos um ambiente saudável entre a equipe para ter o melhor resultado final, algumas características são essenciais no gerenciamento de projetos, tais como, conhecimento, habilidades, ferramentas, técnicas e atividades.

Baseando-se nesses ponto o gerenciamento tenta compreender idéias e viabiliza-las, trabalhando com o balanceamento de custos junto com os prazos de entrega e empenho da equipe, atingindo metas com sprints semanais por exemplo, com entregas frequentes tendo sempre o retorno do usuário para possíveis correções e até mesmo mudanças no percurso que o projeto tem como previsto, podem ter modificações nas datas finais de entrega e custo final, isso se da através de reuniões com os stakeholder.

Os stakeholders são usuários chaves do sistema como um todo ou daquela parte especifica que envolve a sprint que está sendo desenvolvida, através deles é feita a colheita de dados pelos analistas de requisitos, para um posterior refinamento técnico do levantamento da dados, por onde a equipe de desenvolvimento terá as informações para desenvolver o projeto.

PMO, o Project Management Office, criado em 1996 é a área responsável pela definição de padrões de projetos nas empresas, tendo como função chave a elaboração e continuidades desses padrões, podendo ser em toda a empresa ou partes chaves apenas.

PMBOK ou Project Management Body of Knowledge, tem como base a padronização para identificar e conceituar processos, área de conhecimento, técnicas e ferramentas, o PMBOK atua como um guia, para facilitar o gerenciamento de projetos, consistindo das melhores práticas e tendo o respaldo de ter sido implantado em diversas empresas, sendo algo já testado, é a base na qual o gerente de projetos se apoia, o PMBOK utiliza um vocabulário comum em suas práticas possibilitando uma melhor troca de informações entre os gerentes de projetos, não pode ser considerado uma metodologia, pois não oferece uma abordagem diferente em cada tipo de projeto, apenas fornece uma visão geral do gerenciamento de projetos.

PMI, ou Project Management Institute, é um instituto de gerenciamento de projetos, a maior associação sem fins lucrativos do mundo voltada para profissionais de gerenciamento de projetos, fundada em 1969, a missão do PMI é divulgar melhores praticas de gestão de projetos e impulsionar habilidades de pessoas de quase todos os países, o trabalho do PMI é realizado através de mais de 250 capítulos e 30 comunidades de práticas, essas comunidades são voltadas pra diversas áreas da gestão de projetos, incluindo as metodologias ágeis, regularmente o PMI fornece novas diretrizes e regras para desenvolvimento de profissionais de projetos, além disso o PMI oferece oito certificações que atestam conhecimento e competência para profissionais de gestão de projetos, na década de 80, o Brasil foi o primeiro pais a ganhar um capitulo do PMI fora dos EUA, anos mais tarde essa iniciativa foi encerrada e retomada posteriormente em 1998, em 2018 o país já contava om 15 capítulos.

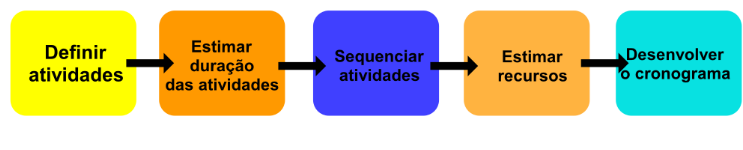
PMP, ou Project Management Professional, é um título de certificação na área de gerenciamento de projetos, concedida desde 1984, o profissional com esse título tem experiência tanto profissional quanto acadêmica, na área de gerenciamento de projetos, competência para dirigir projetos e as equipes responsáveis pelo seu desempenho, é uma certificação essencial para trabalhar na área de gerenciamento de projetos, sendo reconhecida como a certificação mais importante da área, empresas contam com esses profissionais para garantir o sucesso de seus projetos.

Nesse projeto teremos um total de 640 horas para a realização de todo o projeto, divididas em 160 horas para o levantamento de dados e 480 horas para o desenvolvimento.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Horas** | **Valor hora** | **Colaborador** | **Custo total** | **Função** | **Total projeto** |
| 300 | R$ 75,00 | Gustavo | R$ 22.500,00 | Desenvolvimento |  |
| 180 | R$ 75,00 | Maria | R$ 13.500,00 | Desenvolvimento |  |
| 40 | R$ 75,00 | Otavio | R$ 3.000.00 | Analista de requisitos |  |
| 40 | R$ 75,00 | Fabio | R$ 3.000.00 | Analista de requisitos |  |
| 40 | R$ 75,00 | Matheus | R$ 3.000.00 | Analista de requisitos |  |
| 40 | R$ 75,00 | Erick | R$ 3.000.00 | Analista de requisitos | R$ 48.000,00 |

Tabela elaborada pelo autor

Ao desenvolver um projeto de gerenciamento de software é necessário a elaboração de um cronograma para o sucesso nas entregas do projeto, baseando-se no PMBOK de boas práticas, deve-se desenvolver um cronograma com no máximo cinco dias utilizando o EAP como parâmetro para isso, o PMBOK trás cinco atividades para auxilio na elaboração do cronograma

Fonte: o autor

## 2.1.Definir atividades

Inicialmente, é necessário usar a EAP como base para definir as atividades para identificar todas as atividades. Com a ajuda da declaração de escopo e informações de projetos anteriores, nossa ideia é desenvolver um conjunto de metas que tendem a ser o escopo e as metas do projeto e são, em primeiro lugar, administráveis.

## 2.2.Estimar duração das atividades

Como o prazo é um dos principais fatores e fundamentos para o sucesso do projeto, onde determinamos cada atividade e estima-se o tempo necessário para a conclusão das mesmas. Geralmente é estimado em dias e sempre é baseado na lista de atividades. Projeto e conhecimento técnico da equipe de informação.

## 2.3.Sequenciar atividades

Nesse ponto determinamos a atribuição e interligações das atividades, para ter uma maior harmonia durante seu desenvolvimento e com isso uma maior satisfação do usuário final, ressaltando que todas as novas atividades ou mudanças nas atividades previstas poderá acarretar no acréscimo no prazo final de entregar e custo do projeto

## 2.4.Estimar recursos

Nessa fase o gerente de projetos junto de sua equipe deve estimar com precisão os recursos a serem empenhados no projeto, tanto humanos quanto financeiros, será estimado quanto o projeto irá custar para o cliente, fase fundamental para o andamento do projeto e também para a confiança do cliente, pois ninguém quer no meio do projeto se deparar com custos não previstos.

## 2.5.Desenvolver o cronograma

Parte fundamental na elaboração de um projeto é desenvolver seu cronograma, onde será feito a estimativa das entregas, as sprints semanais, prioridades e alocação de recursos humanos e materiais.

# 3. DEFINIÇÃO DO NEGÓCIO

Uma rede blockchain atua sem a necessidade de intermediários, ou seja, todos os usuários tem uma cópia de todas as transações, dessa forma qualquer pessoa na rede pode verificar se determinada transação é legítima, aumentando o nível de transparência também.

A Figura 1 contém uma representação visual de uma transação em blockchain, onde os usuários realizaram uma transação, que foi incluída em um bloco, teve sua validade confirmada pelos nós da rede e então o bloco foi adicionado a cadeia de blocos antes da transação ser confirmada a realizada.

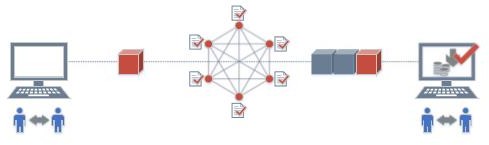


Figura 2: Funcionamento de uma transação em blockchain (Andoni et al., 2019).

Fonte: o autor

Blockchain: O blockchain é um livro-razão distribuído que permite que os participantes escrevam e atualizem registros no livro-razão e na criptografia garante que o armazenamento de registros permaneça o mesmo uma vez adicionado. Os registros são adicionados ao livro-razão na forma de transações, e essas transações são agrupadas em blocos. Cada bloco está criptograficamente ligado ao próximo bloco. A árvore de Markel ou árvore de hash é um método criptográfico que garante que as transações armazenadas em blockchain estejam ligadas a hastes matemáticas. Isso garante que nenhuma modificação pode invalidar todo o registro. Os hashes fornecem um método eficiente para verificar qualquer transação na blockchain. Com este método, os registros podem ser verificados sem passar por todos os dados armazenados na rede.

A rede blockchain é representada pelos nós que estão conectados em pares e cada nó participante tem uma cópia do livro razão [6]. Os nós são executados por máquinas virtuais, por exemplo, um nó blockchain é alimentado pela Máquina Virtual Ethereum (EVM). Uma vez que um novo bloco é aceito na rede, cada nó atualiza seu registro adicionando e novo bloco. As transações são carimbadas e enviadas dos nós participantes. Todos os nós da rede concordam com um método de consenso para adicionar novos registros ao livro. As transações são agrupadas em blocos e uma vez que um bloco é aceito pela rede, todos os nós participantes adicionam o novo bloco à sua cópia do livro-razão.

Um método de consenso é um método acordado para adicionar novos registros ao blockchain pelos nomes participantes. Os métodos de consenso são agrupados em dois - consenso baseado em votação e consenso baseado em prova. Proof-of-Work (PoW) é um método de consenso baseado em prova e está sendo usado atualmente pelo Bitcoin e pela blockchain no momento desta escrita. PoW é um mecanismo de consenso que permite que todos os nós participantes resolvam um problema matemático difícil e recompensa o primeiro nó que resolve o problema selecionando o nó para adicionar o próximo bloco.

O Proof-Of-Stake (PoS) é outro exemplo de método de consenso baseado em prova. No entanto, enquanto o PoW motiva uma centralização do poder computacional, o PoS move a base de decisão do poder de computação para a posse de participação no sistema, como uma quantidade de criptomoedas.

Solidity é um exemplo de uma linguagem de programação que fornece um método para executar um código de computador em nós blockchain. Programas de computador que verificam digitalmente, executam contratos e executam em uma rede blockchain um reto e referido como Smart Contract. Os Smart Contract são armazenados e executados em nós blockchain. Com o acesso certo, qualquer usuário pode executar e executar funções de contrato inteligente saem de qualquer nó participante na rede.

A empresa PIM foi então chamada de forma a permitir que os clientes possam gerenciar e ter acesso parcial e consolidado de seus investimentos e criptoativos em relatórios e gráficos, e também que a empresa transmissora possua esses dados de seus clientes.

## 3.1. Outras Soluções Disponíveis

A primeira pesquisa acadêmica encontrada na literatura sobre o uso de blockchain, nos mercados é de Mihaylov (2014), e seu objetivo foi o de utilizar as criptomoedas para a negociação. As moedas são produzidas de acordo com as condições de oferta demanda no momento em que são injetadas na rede, de modo que o custo real reflita no seu preço. As moedas podem ser negociadas na rede ou utilizadas para compra de criptoativos.

Outra solução foi a de Foti et al. (2018), que implementou uma rede privada com 306 nós, Cada um dos participantes tinha acesso a uma conexão externa utilizando uma senha, e também tinha uma chave privada e uma criptografada, que estavam instaladas no blockchain. Todo o sistema foi implementado em blockchain, sendo criado um contrato para cada cenário simulado, onde foram especificadas as regras em que o mercado deveria operar.

Foi utilizada a rede privada da Ethereum, onde os nos foram instalados em 3 máquinas virtuais, com cerca de 100 nós em cada máquina, tendo cada uma 244 G de RAM e cada CPU com 16 núcleos. O GridLAB-D foi instalado na quarta e última máquina virtual, sendo que essa tinha 4G de RAM e um CPU com 4 núcleos. O sistema foi desenvolvido em C++ e foi integrado à biblioteca GridLAB-D, utilizada para permitir que cada consumidor pudesse utilizar seu nó exclusivo para envio de transações aos contratos inteligentes através do JSON.

Dessa forma, temos que as demais soluções encontradas focam no gerenciamento da rede blockchain e nos contratos inteligentes, deixando para segundo plano o gerenciamento dos clientes e da parte administrativa, foco desse trabalho e da solução proposta.

# 4. REFERENCIAL TEÓRICO

## 4.1 - Arquitetura MVC

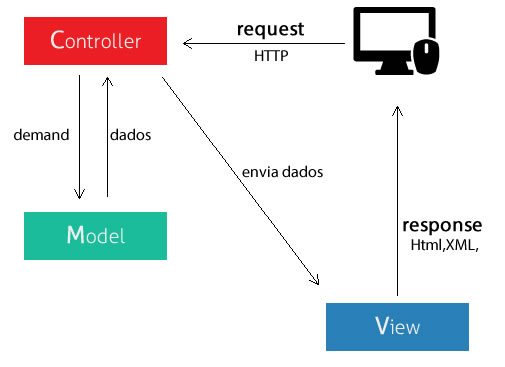
MVC (Model, View, Controller) é um padrão de arquitetura de software criado

na década de 70 e desenvolvido para ser usado em projetos de interface visual em

Smaltalk, linguagem de programação que ganhou grande reconhecimento junto com

o C ++ na época, mesmo após esses anos de sua criação ainda é um pattern

aplicável em diversas aplicações, principalmente web.



Fonte: Disponível em https://tableless.com.br/mvc-afinal-e-o-que/

## 4.1.1 - Camada Módel

Essa Camada é responsável pela Lógica, pela regra do negócio, ela é o software em si. Essa camada representa e manipula seguindo as regras do negócio.

O modelo encapsula o estado e comportamento da aplicação além de ser o único componente do MVC que faz interface da aplicação frente à fonte de dados, que normalmente é representada pelo banco de dados da aplicação.

## 4.1.2 - Camada View

Essa camada é responsável pela apresentação, é a interface de representação do modelo, ou seja, trata-se da fronteira entre usuário e o sistema em si. A view pode dar forma mais conveniente, exibir alguns atributos e ocultar outros, atuando como um filtro para os dados do modelo. É papel do controlador definir a view apropriada para a exibição da resposta obtida pela requisição feita.

## 4.1.3 - Camada Controller

A camada Controller ou controlador é a camada responsável pela intermediação entre as camadas Módel e View, essa camada é que define o comportamento da aplicação, gerenciando e interpretando as requisições dos usuários e retornando uma interação entre as camadas módel e View.

# 5. FUNÇÕES DO NEGÓCIO

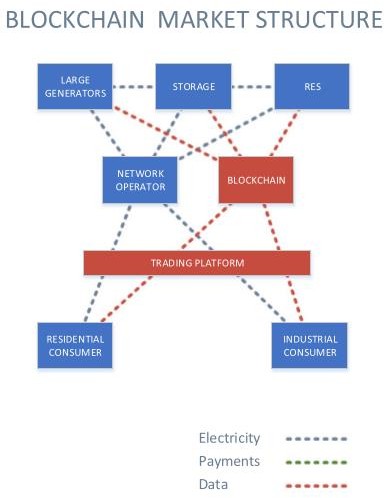
De forma geral, o funcionamento do sistema é como consta na Figura 3, e o sistema proposto tem como objetivo fazer a ligação entre os consumidores, sejam eles pessoa física ou jurídica, a rede operadora e a rede blockchain. De forma a manter disponível para todos os envolvidos na rede as informações centralizadas

Figura 3: Rede de distribuição baseado na tecnologia blockchain (Andoni et al., 2019).

Abaixo apresentamos as principais funcionalidades do sistema:

* + - Cadastro dos clientes.
    - Os clientes deverão dar aceite no documento que lhes informa como será feito o tratamento de seus dados, ou seja, por quanto tempo serão armazenados, como serão destruídos ou anonimizados, entre outros.
    - O sistema deverá ser desenvolvido em C# com banco de dados SQL Server.
    - O sistema deverá apresentar em tempo real um relatório parcial aos clientes quanto ao valor de cada um no final do mês.
    - O preço será computado conforme um leilão pela rede blockchain com os contratos inteligentes, e apenas informará o valor para esse sistema.
    - Esse sistema não terá nenhuma ação sobre a rede blockchain implementada, limitando-se a consumir os valores enviados.
    - A empresa prestadora do serviço terá acesso a relatórios gerais e por cliente.
    - Os clientes terão acesso a relatórios mensais de valores já consolidados.
    - Os clientes poderão configurar um alerta para quando o preço atingir determinado valor, ou quando seus criptoativos atingirem determinado valor.
    - Os valores mensais e diários poderão ser apresentados em forma de gráfico.
    - A empresa prestadora poderá ver uma lista com o status de pagamento dos clientes.

O sistema funcionaria então a partir dos seguintes princípios e tarefas:

* + - Empresa: Acesso somente via desktop. Teria acesso a todos os clientes, produto gerado e consumido de cada um deles. Controlaria também os pagamentos e promoções. Relatórios gerais e por cliente.
    - Clientes: Acesso somente via mobile ou web através de layout responsivo, sendo que o aplicativo seria desenvolvido apenas emulando a versão web, de forma a se economizar recursos. Cada cliente teria acesso a seus gastos, ou seja, consumo. Relatórios mensais consolidados e diários parciais. De forma a atender usuários com deficiência visual, o sofware pode conter as tags “alt” nas imagens, permitindo dessa forma que softwares específicos leiam as imagens como texto, bem como conter botões grandes e de fácil identificação via cores para facilitar o uso por pessoas com problemas de visão.

Apresentamos o caso de uso referente aos processos que serão executados pelo sistema na Figura 4, onde o <<extend>> demonstra a obrigação de cada cliente ter ao menos um produto cadastrado. A descrição de cada caso de uso, bem como seus fluxos encontram-se nas Tabelas a seguir.

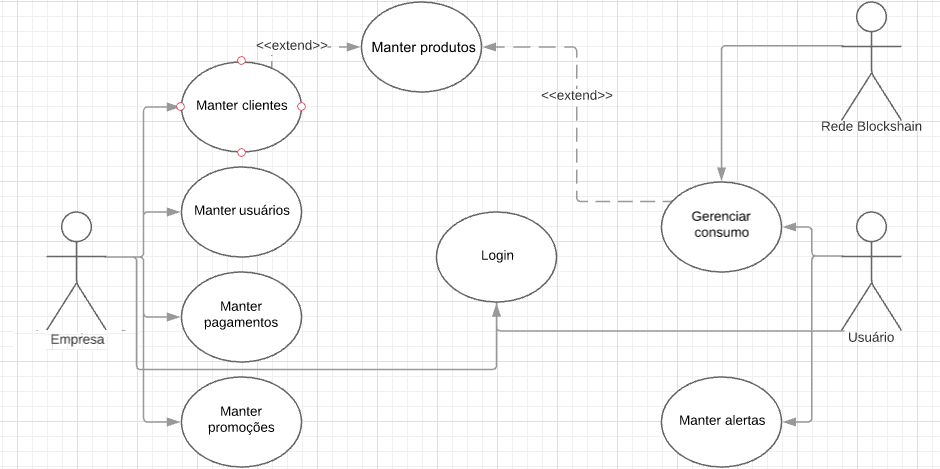


Figura 3: Estudo de caso da solução proposta (do Autor).

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Manter clientes** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Empresa |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de manter os clientes |
| **Pré-condições** | Empresa autenticada no sistema |
| **Pós-condições** | Ação efetuada com o cliente |
| **Fluxo Principal** |  |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Consultar cliente |  |
| 2. Verifica se cliente existe |  |
|  | 3. Caso não exista, cadastrar cliente |
|  | 1. Caso exista, escolher a ação que deseja:    * editar, excluir |
| **Restrições/ Validações** | 1. Não podem haver 2 clientes com o mesmo CPF |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar que cliente não existe 2. Informar que dados de cadastro estão incorretos (cpf, data de nascimento, entre outros). |

Tabela 1: Manter clientes (do usuário).

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Manter Proditos** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Empresa |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de manter os clientes |
| **Pré-condições** | Empresa autenticada no sistema, cliente cadastrado |
| **Pós-condições** | Ação efetuada |
| **Fluxo Principal** |  |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Consultar cliente |  |
| 2. Verifica se esse produto existe para o cliente |  |
|  | 3. Caso não exista, cadastrar |
|  | 1. Caso exista, escolher a ação que deseja:    * editar, excluir |
| **Restrições/ Validações** | 1. Não é possível cadastrar um produto sem o mesmo estar vinculado a um cliente já cadastrado |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar que cliente não existe 2. Informar que dados de cadastro estão incorretos (cep, endereço, entre outros). |

Tabela 2: Manter investimentos (do usuário).

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Manter promoções** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Empresa |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de manter as promoções |
| **Pré-condições** | Empresa autenticada no sistema |
| **Pós-condições** | Ação efetuada com a promoção |
| **Fluxo Principal** |  |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Consultar promoção |  |
| 2. Verifica se promoção existe |  |
|  | 3. Caso não exista, cadastrar promoção |
|  | 1. Caso exista, escolher a ação que deseja:    * editar, excluir |
| **Restrições/ Validações** | 1. Uma promoção em uso por algum cliente não pode ser excluída |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar que promoção não existe 2. Informar que dados de cadastro estão incorretos (data de inicio, data de fim, entre outros). |

Tabela 3: Manter promoções (do usuário).

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Manter alertas** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Usuário |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de manter os alertas |
| **Pré-condições** | Usuário autenticado no sistema |
| **Pós-condições** | Ação efetuada com o alerta |
| **Fluxo Principal** |  |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Consultar alerta |  |
| 2. Verifica se alerta existe |  |
|  | 3. Caso não exista, cadastrar alerta |
|  | 1. Caso exista, escolher a ação que deseja:    * editar, excluir |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar que alerta não existe 2. Informar que intervalo de cadastro está incorreto. |

Tabela 4: Manter alertas (do usuário).

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Manter pagamentos** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Empresa |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de manter os pagamentos dos clientes. |
| **Pré-condições** | Empresa autenticada no sistema, produto cadastrado |
| **Pós-condições** | Ação efetuada com o pagamento |
| **Fluxo Principal** |  |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Consultar produto |  |
| 2. Verifica se existe pagamento para o produto |  |
|  | 3. Caso não exista, cadastrar pagamento |
|  | 1. Caso exista, escolher a ação que deseja:    * editar valor, dar baixa |
| **Restrições/ Validações** | 1. Não é possível excluir uma pagamento 2. Toda pagamento deve estar associado a um um produto |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar esse produto não existe 2. Informa que a cobrança não existe |

Tabela 5: Manter pagamentos (do usuário).

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Manter usuários** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Empresa |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de manter os usuários |
| **Pré-condições** | Empresa autenticada no sistema |
| **Pós-condições** | Ação efetuada com o usuário |
| **Fluxo Principal** |  |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Consultar usuário |  |
| 2. Verifica se usuário existe |  |
|  | 3. Caso não exista, cadastrar usuário |
|  | 1. Caso exista, escolher a ação que deseja:    * editar, excluir |
| **Restrições/ Validações** | 1. Cadastro de usuário válido apenas para uso interno (administrativo) do sistema |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar que usuário não existe 2. Informar que dados de cadastro estão incorretos (telefone, email, entre outros). |

Tabela 6: Manter usuários (do usuário).

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Login** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Usuário e Empresa |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de logar no sistema |
| **Pré-condições** | Usuário não autenticado no sistema |
| **Pós-condições** | Login e acesso ao sistema realizado |
| **Fluxo Principal** |  |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Digitar login e senha |  |
|  | 2. Sistema válida os dados informados |
|  | 3. Sistema libera acesso ao sistema |
| **Restrições/ Validações** | 1. O usuário deve ter sido cadastrado |
|  | 2. Necessário preencher campo de login e senha |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar que dados informados não constam na base de de dados. 2. Informar que campo login deve ser preenchido. 3. Informar que campo senha deve ser preenchido. |

Tabela 7: Login (do usuário)

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **Gerenciar consumo** |
| **Caso de Uso Geral** |  |
| **Ator Principal** | Usuário, rede blockchain |
| **Atores Secundários** |  |
| **Resumo** | Descreve a etapa de incluir consumo de determinado produto |
| **Pré-condições** | Produto cadastrado no sistema |
| **Pós-condições** | Consumo eretorno para o cliente |
| **Fluxo Principal** |  |
| **Ações da Rede Blockchain** | **Ações do Sistema** |
| 1. Seleciona produto |  |
| 2. Verifica o valor de utilização para aquele momento de acordo com a oferta/ demanda |  |
| 3. Envia o valor via json |  |
|  | 4. Verifica se o registro já não existe, se não existir cadastra |
| **Restrições/ Validações** | 1. Caso o consumo atinja o alerta programado pelo cliente dispara e- mail para o mesmo |
| **Fluxo Exceção** |  |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Informar que dado já foi cadastrado |

Tabela 8: Gerenciar consumo (do usuário).

# 6. REQUISITOS

Quanto aos requisitos, esses podem ser definidos como funcionais e não funcionais, sendo que os funcionais são os referentes as funcionalidades e regras de negócio do sistema, e os não funcionais são referentes a qualidade e usabilidade do sistema, geralmente não impactando nas funções de utilização, e sim na experiência do usuário.

Os requisitos funcionais são apresentados na Tabela 9, sua matriz de rastreabilidade é apresentada na Tabela 10 e os não funcionais estão na Tabela 11.

|  |  |  |
| --- | --- | --- |
| **ID** | **Descrição** | **Prioridade** |
| RF01 | Login | Alta |
| RF02 | Manter usuários | Alta |
| RF03 | Manter clientes | Alta |
| RF04 | Manter produtos | Alta |
| RF05 | Manter investimentos | Média |
| RF06 | Manter promoções | Baixa |
| RF07 | Manter alertas | Baixa |
| RF08 | Incluir consumo/ geração de lucros | Alta |
| RF09 | Relatórios | Média |

Tabela 9: Requisitos funcionais (do Autor).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **01** | **02** | **03** | **04** | **05** | **06** | **07** | **08** | **09** |
| **01** |  | X | X | X | X | X | X |  | X |
| **02** |  |  |  |  |  |  |  |  |  |
| **03** |  |  |  | X |  |  |  |  | X |
| **04** |  |  |  |  |  |  | X | X | X |
| **05** |  |  |  |  |  |  |  |  | X |
| **06** |  |  |  |  |  |  |  |  | X |
| **07** |  |  |  |  |  |  |  |  |  |
| **08** |  |  |  |  |  |  | X |  | X |
| **09** |  |  |  |  |  |  |  |  |  |

Tabela 10: Matriz de rastreabilidade (do Autor).

|  |  |  |
| --- | --- | --- |
| **ID** | **Descrição** | **Prioridade** |
| RNF01 | A versão web deve ser responsiva | Alta |
| RNF02 | O aplicativo a ser disponibilizado deve emular a versão web | Alta |
| RNF03 | Os botões e texto devem ser grandes e intuitivos | Médio |
| RNF04 | As imagens devem conter a tag “alt” para que leitores de tela utilizados por pessoas com problemas de visão possam ler a imagem | Alta |
| RNF05 | O sistema deve apresentar um resumo de todas as informações (*dashboard*) após logar | Média |

Tabela 11: Requisitos não funcionais (do Autor).

# 

# 7. DIAGRAMA DE SEQUÊNCIA

O diagrama de sequência proposto na Figura 4 demonstra o caso de uso Login, e seu objetivo é demonstrar a interação entre as camadas model, view e controller do sistema.

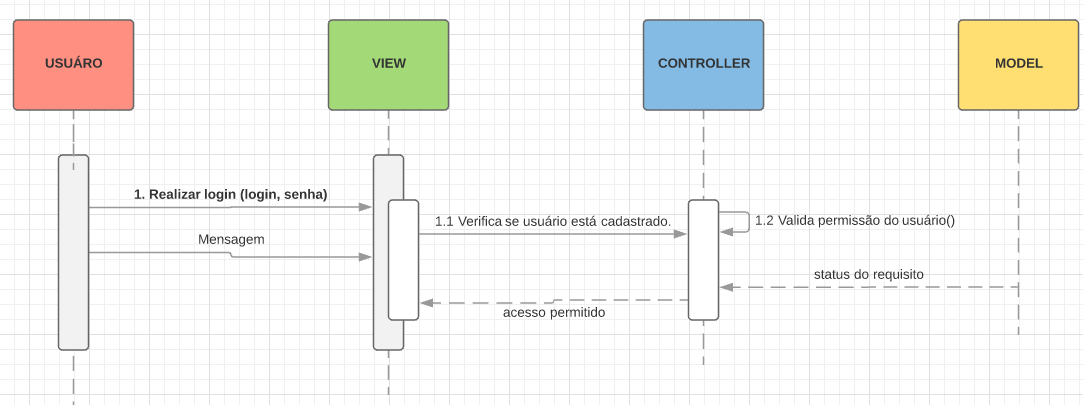


Figura 4: Diagrama de sequência do caso de uso Login (do Autor).

# 

# 8. DIAGRAMA DE CLASSE

A Figura 5 contém o diagrama de classes.

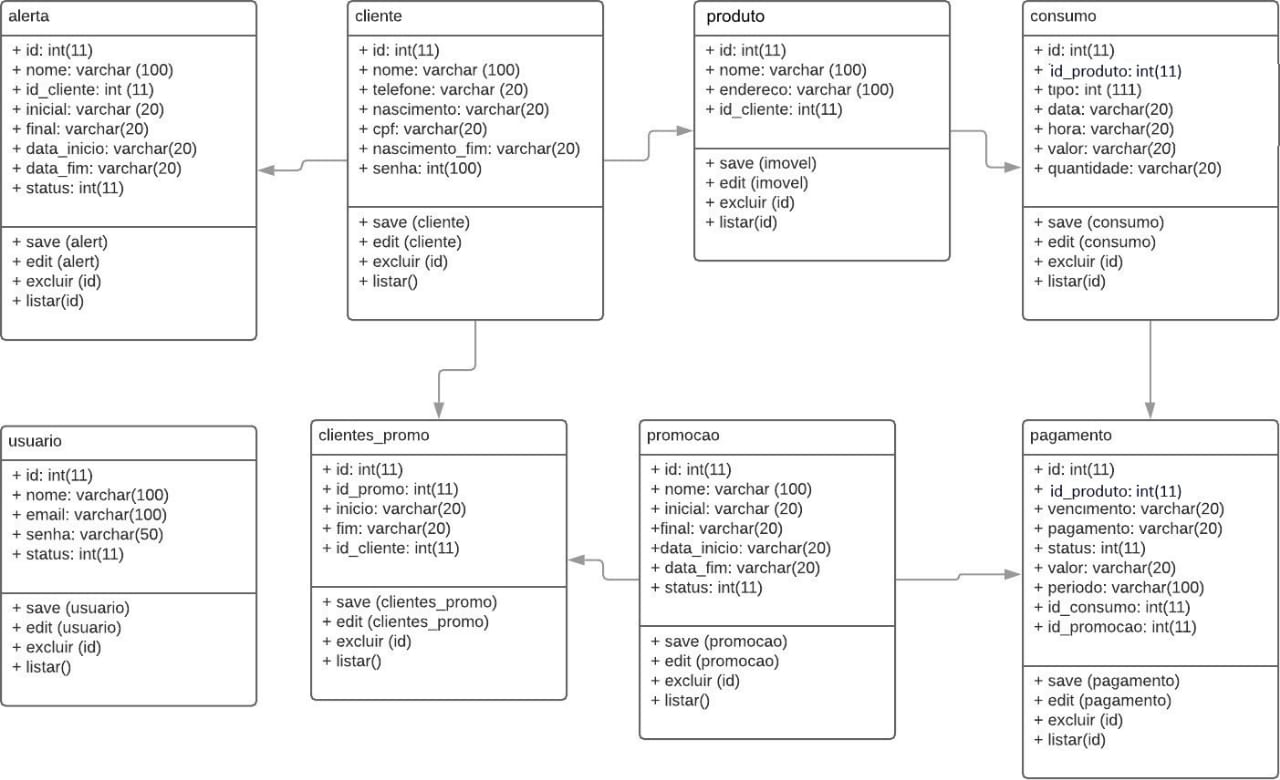
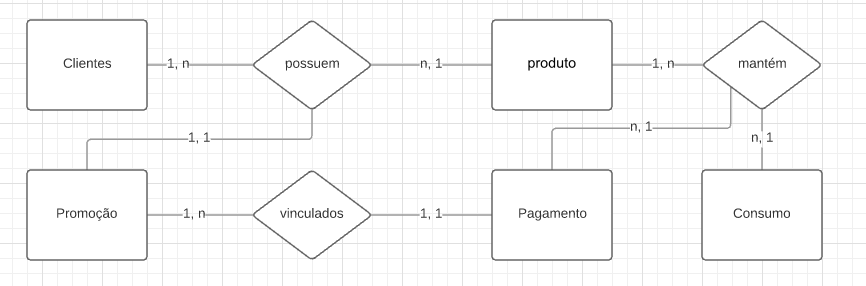


Figura 5: Diagrama de classe (do Autor).

# 9. DIAGRAMA ENTIDADE RELACIONAMENTO

Na Figura 6 apresentamos o diagrama entidade relacionamento, e a geração do script de criação do banco de dados e do primeiro usuário encontra-se no Apêndice 1.

Figura 6: Diagrama de classe (do Autor).

**Modelo Relacional**

create table alerta

(

id int primary key,

nome varchar(100),

inicial varchar(20),

data\_inicio datetime,

data\_fim datetime,

status int,

id\_cliente int,

FOREIGN KEY (id\_cliente) references cliente(id)

)

create table cliente

(

id int primary key,

nome varchar(100),

telefone varchar(20),

cpf varchar(20),

nascimento varchar(20),

senha varchar(50)

)

create table produto

(

id int primary key,

nome varchar(100),

endereco varchar(100),

id\_cliente int,

FOREIGN KEY (id\_cliente) references cliente(id)

)

create table consumo

(

id int primary key,

id\_produto int,

tipo int,

data datetime,

hora varchar(10),

quantidade int,

FOREIGN KEY (id\_produto) references produto(id)

)

create table usuario

(

id int primery key,

nome varchar(100),

email varchar(100),

senha varchar(50),

status int

)

create table clientes\_promo

(

id int primary key,

id\_promo int,

inicio datetime,

fim datetime,

id\_cliente int,

FOREIGN KEY (id\_cliente) references cliente(id),

FOREIGN KEY (id\_promo) references promocao(id)

)

create table promocao

(

id int primary key,

nome varchar(100),

inicial varchar(20),

final varchar(20),

data\_inicio datetime,

data\_fim datetime,

status int

)

create table pagamento

(

id int,

id\_produto int,

vencimento datetime,

pagamento datetime,

status int,

valor decimal(10,2),

periodo varchar(100),

id\_comsumo int,

id\_promocao int,

FOREIGN KEY (id\_consumo) references consumo(id),

FOREIGN KEY (id\_promocao) references promocao(id),

FOREIGN KEY (id\_produto) references produto(id) )

# 10. O SISTEMA

## 10.1. Protótipo De Telas

## 10.1.1 Versão Mobile

A versão móbile do sistema foi desenvolvida utilizando a ferramenta App Inventor, o aplicativo permite que o usuário acesse e faça o cadastro, porém enquanto está em fase de desenvolvimento, para acessar o sistema, após o cadastro, o administrador envia um login e senha de acesso.

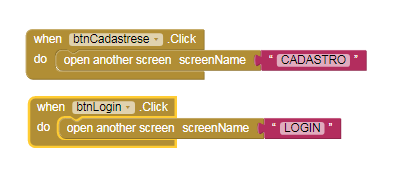
Com o login e a senha fornecidos pelo administrador do sistema, o usuário tera acesso a todas as funcionalidades oferecidas pela empresa e poderá cadastrar transações, realizar consultas etc.

Na figura abaixo, apresentamos a tela principal do aplicativo, ao clicar em cadastre-se, o usuário será direcionado para a tela de cadastro da figura 7 e ao clicar em login será direcionado para a segunda tela da figura 7.



Figura 6: Protótipo da tela principal (do Autor).

Bloco de programação da tela principal



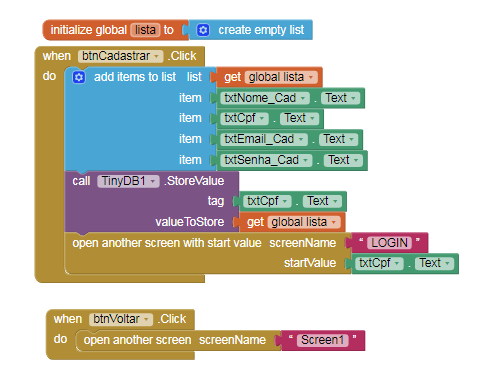
Fonte: autor - desenvolvimo em <http://appinventor.mit.edu/>

Na Figura 7 é apresentado 2 telas responsivas, do aplicativo móbile, sendo a da esquerda e tela de cadastro e a da direita a tela de login de usuários.

|  |  |
| --- | --- |
|  |  |

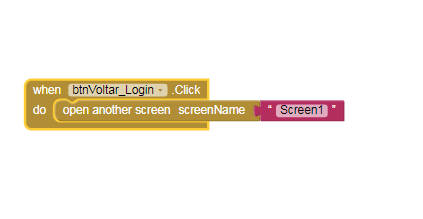
Figura 7: Protótipo de telas responsivas (do Autor).

Blocos de programação da tela de cadastro



Fonte: autor - desenvolvido em <http://appinventor.mit.edu/>

Blocos de programação da tela de login



Fonte: autor - desenvolvido em <http://appinventor.mit.edu/>

## 10.1.2 – Versão Desktop

Nas figuras a seguir são apresentados alguns protótipos de telas de gerenciamento administrativo do sistema.

A versão desktop foi desenvolvimento na linguagem C#, utilzando MVN e entity framework. Essa versão so poderá ser acessada pelo administrador do sistema.

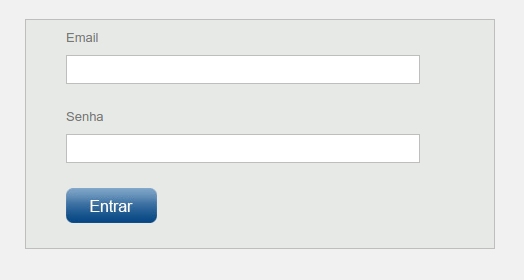


Figura 8: Tela de login (do Autor)

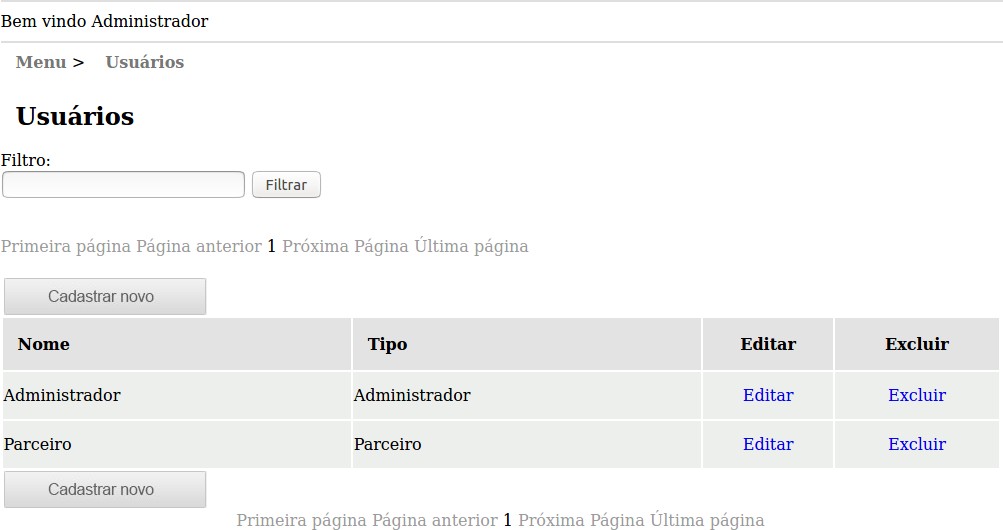


Figura 9: Tela de listagem de usuários (do Autor).

## 10.1.3 Versão Web

A versão web do sistema foi desenvolvida em HTML5, C# e css, para que o usuário tenha acesso às transações e funcionalidades do site, deverá fazer o cadastro e aguardar os dados de acesso. Apresentamos abaixo lgumas telas de interação do usuario com o sistema web.



Figura 10: Protótipo de tela de login (do Autor).

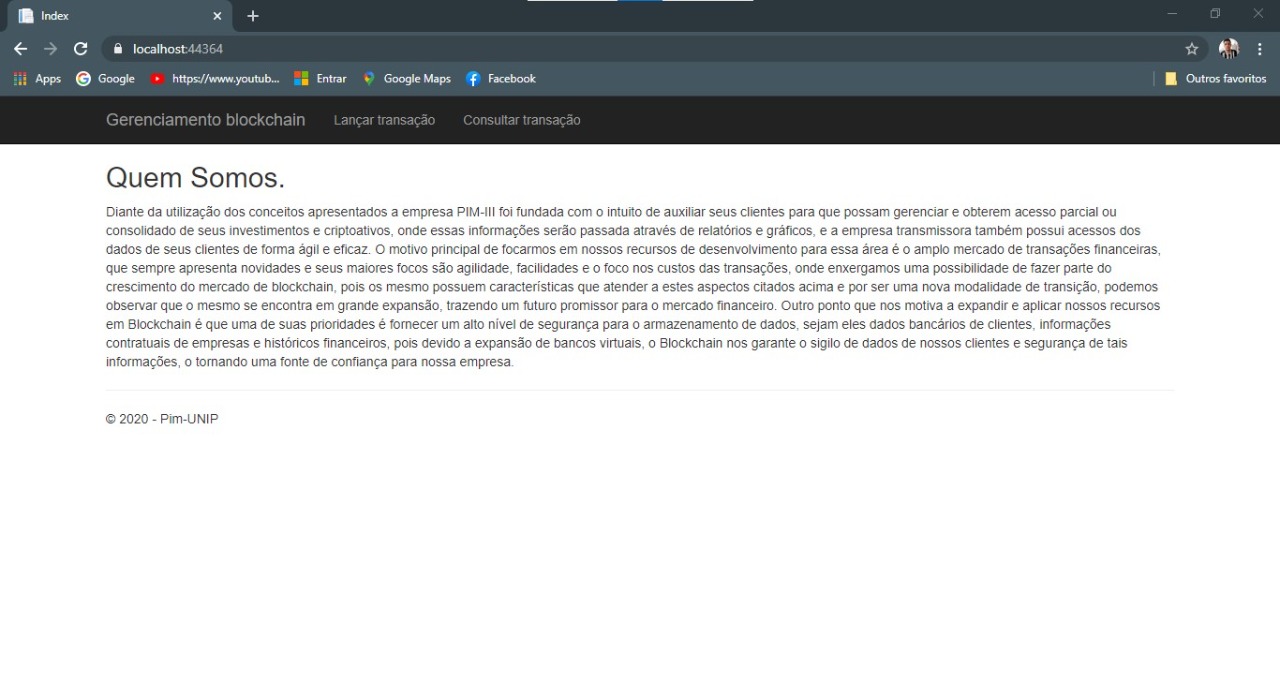


Figura 11: Protótipo de tela quem somos (do Autor).

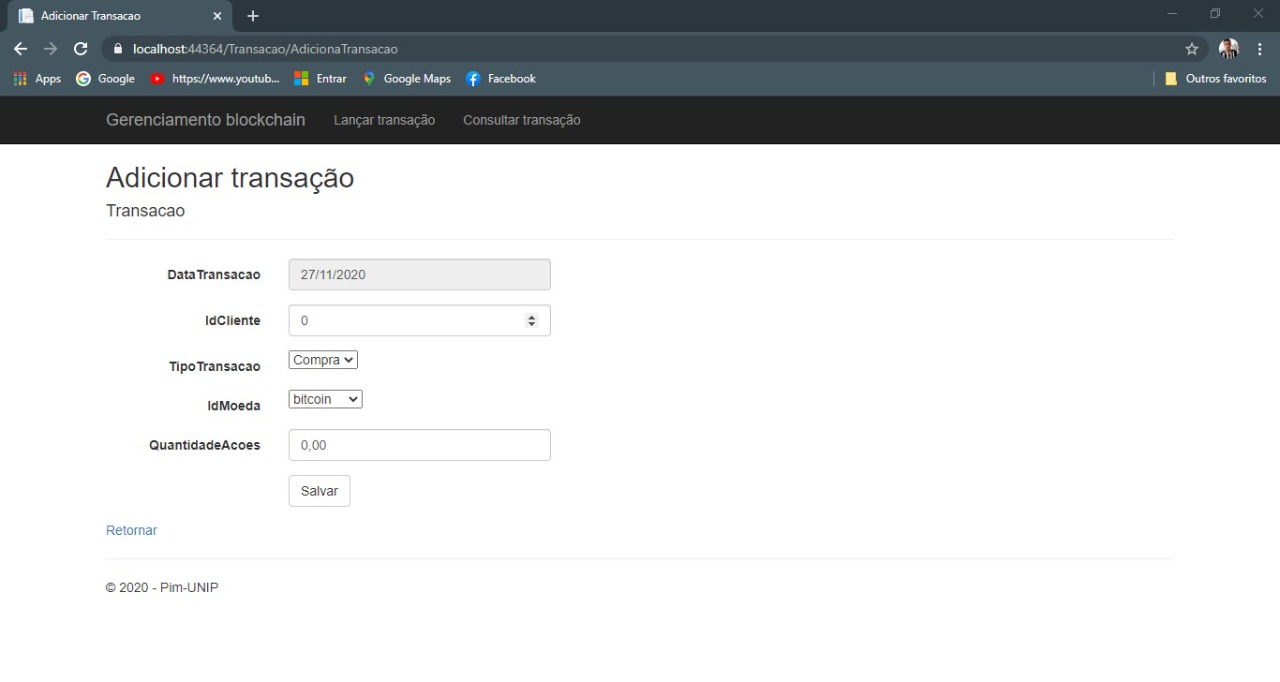


Figura 12: Protótipo de tela adicionar transação (do Autor).

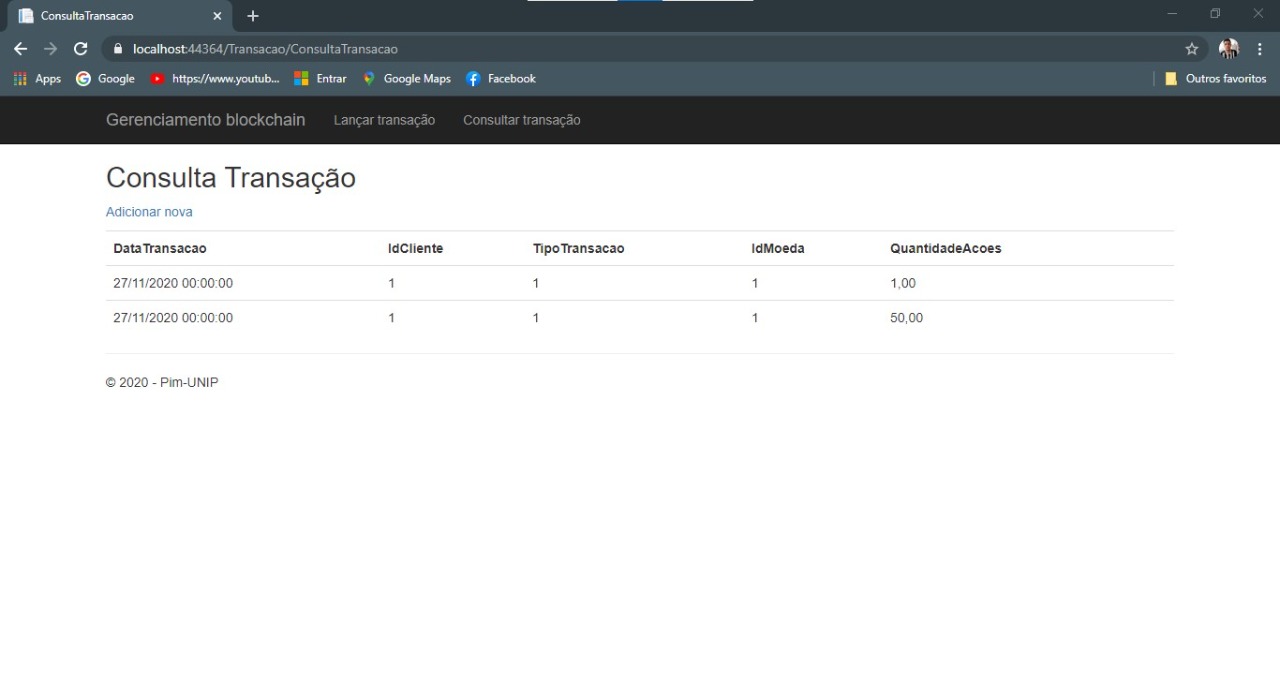


Figura 13: Protótipo de tela consultar transação (do Autor).

# 11. DIAGRAMA DE IMPLANTAÇÃO

O diagrama de implantação é apresentado na Figura 14, onde faz-se necessário um servidor com sistema operacional Windows Server, devido a escolha do banco de dados (SQL Server) e linguagem de programação (C#). O servidor conterá uma base de dados SQL Server, que se comunicará com os sistemas desktop, web e mobile.

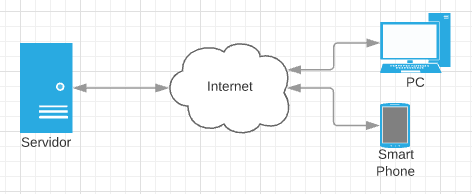


Figura 14: Diagrama de implantação (do Autor).

A instalação do sistema se fará através de um cd com o sistema, sendo que o manual de instalação do sistema pode ser encontrado no Apêndice 3, enquanto que o treinamento de uso é feito via video-aulas e cobre os seguintes itens:

* + - Cadastro de usuário
    - Cadastro de produto
    - Cadastro de promoções
    - Gerenciar pagamento dos usuários

Ao final de cada video aula o usuário é convidado a executar uma tarefa relativa ao módulo proposto, somente assim podendo dar sequência ao treinamento. Um profissional também ficará a disposição online para auxiliar e tirar dúvidas.

Ao final da instalação do sistema, o sistema executa de forma automática os testes apresentados na Tabela 12, de forma a verificar se a instalação foi bem sucedida.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Funcionalidade** | | | **Resultado Esperado** | **Query** | |
| Criação do primeiro usuário | | | Usuário criado | INSERT INTO `tcc`.`usuario` (`us\_cod`,  `nome`, `senha`, `status`, `email`) VALUES (NULL, 'Administrador', '336699', '2', 'admin@pimiii.com.br'); | |
| Criação de cliente | | | Cliente criado | INSERT INTO `tcc`.`cliente` (  `cl\_cod` , `nome` , `cpf` , `endereco` ,  `profissao` , `telefone` , `email` ,  `nascimento` ) VALUES (  NULL , 'Jimmy Silva', '876.454.545.-33',  'Rua Principal, 22. Bairro centro. Cep 83.546-88. São Paulo/ SP', 'Empresário', '(49) 98763-8989', 'jime@company.com.br', '25/12/1990' | |
|  | | |  | ); | |
| Edição de cliente | | | Cliente editado | UPDATE `tcc`.`cliente` SET `telefone` = '(49) 9833-3454' WHERE  `cliente`.`cl\_cod` =1; | |
| Inclusão de produto | | | produto incluído | INSERT INTO `tcc`.`produto` (`id`,  `id\_cliente`, `nome`, `endereco`) VALUES (NULL, '1', 'produto principal', 'Rua dos pombos, 44. Centro. Cidade/ Estado. Cep. 43.346-34. '); | |
| Login | | | Usuário logado | SELECT count(\*) AS total, us.\* FROM usuario us WHERE email = “[jime@company.com.br](mailto:jime@company.com.br)” AND senha = “336699” LIMIT 1 | |
| Busca imóveis do cliente | | | Listagem de produtos do cliente indicado | SELECT \* FROM produtos id\_cliente = 1 | WHERE |
| Inclusão de alerta | | | Alerta incluído | INSERT INTO `tcc`.`alerta` (`id`,  `nome`, ‘id\_cliente’, `inicial`, `final`,  `data\_inicio`, `data\_fim`, `status`) VALUES (NULL, 'Máximo consumo dia', '1’, '150', '12/02/2020', '', '2'); | |
| Edição promoção | de | | Promoção editada | UPDATE `tcc`.`promocao`  `data\_fim` = '12/02/2021'  `promocao`.`id` =1; | SET WHERE |
| Listagem de consumo | | | Consumos listados | SELECT \* FROM consumo WHERE id\_produto = 3 | |
| Listagem promoção | da do | | Promoção listada | SELECT pr.nome AS promocao FROM promocao pr INNER JOIN clientes\_promo cp ON (cp.id\_promo = | |
| cliente | | |  | pr.id) WHERE cp.id\_cliente = 1 | |
| Inclusão | | de | Pagamento incluído | INSERT INTO `tcc`.`pagamento` (`id`, | |
| pagamento | | para |  | `id\_produto`, `id\_consumo`, | |
| um consumo | |  |  | `vencimento`, `pagamento`, `status`, | |
|  | |  |  | `valor`, `periodo`, `id\_promocao`) | |
|  | |  |  | VALUES (NULL, '1', '1', '20/04/2020', '', | |
|  | |  |  | '3', '110,67', '10/03/2020 - 10/04/2020', | |
|  | |  |  | '1'); | |
| Baixa  pagamento | | de | Pagamento consta  como pago | UPDATE `tcc`.`pagamento` SET  `pagamento` = '13/04/2020', | |
|  | |  |  | `status` = '2' WHERE `pagamento`.`id`  =1; | |
| Listagem pagamentos aberto | | dos em | Pagamentos pendentes listados | SELECT \* FROM pagamento WHERE status = 3 | |
| Exclusão de todos os pagamentos | | | Pagamentos excluídos | DELETE FROM pagamento | |
| Exclusão consumos cliente | | de do | Consumos excluídos | DELETE FROM consumo WHERE id\_produto = 1 | |
| Exclusão promoções | | de | Promoções excluídas | DELETE FROM promocao | |
| Exclusão de todos os alertas de um cliente | | | Alertas excluídos | DELETE FROM produto WHERE  id\_cliente = 1 | |
| Exclusão de alerta | | | Alerta excluído | DELETE FROM alerta WHERE id = 4 | |
| Exclusão de produto | | | Produto excluído | DELETE FROM produto WHERE id = 1 | |
| Exclusão de cliente | | | Cliente exluído | DELETE FROM `tcc`.`cliente` WHERE  `cliente`.`cl\_cod` = 1" | |

Tabela 12: Planilha de testes para homologação no sistema (do Autor).

# 

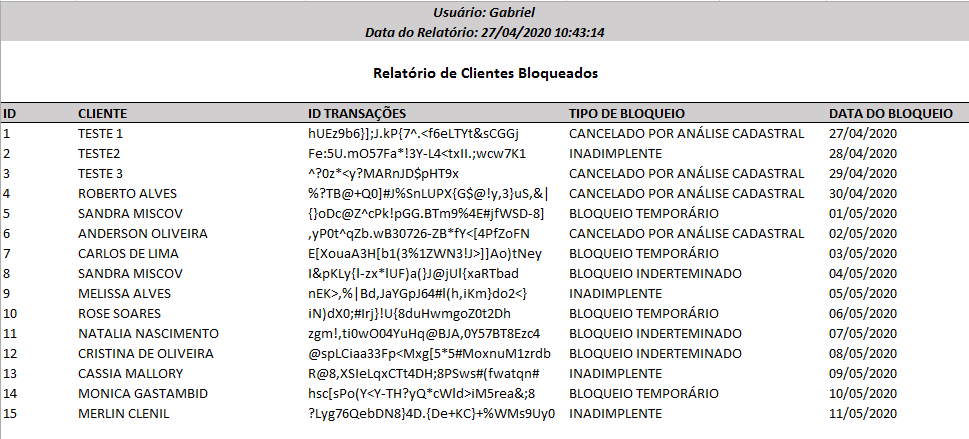
# 12. RELATORIO DE CLIENTES

Aderindo as informações do RF9, a seguir teremos a geração de relatórios de clientes no qual foram bloqueados por uso indevido da plataforma, cliente inadimplentes e que foram cancelados por solicitação do mesmo, será apresentado a tela de solicitação de clientes bloqueados e suas respectivas funções.



Fonte: o autor

Em seguida o sistema reconhecendo a solicitação, vai gerar o relatório de acordo com as ordens requisitadas.



Fonte: o autor

## 12.1 Tipos de Bloqueio

* Cancelado por análise cadastral

Geralmente o cliente é cancelado por análise cadastral por usar a plataforma indevidamente, pode ser casos de fraude e por atividade maliciosa dentro da máquina do cliente.

* Inadimplente

Nesse caso é quando o cliente está com algum debito em aberto na conta, sendo assim só poderá ser resolvido diretamente com o setor financeiro.

* Bloqueio Temporário

Quando o cliente tentar acessar a conta por mais de 3 vezes com a senha incorreta, o sistema bloqueia o acesso no período de 1 hora. Sendo assim terá que aguardar este tempo para solicitar o acesso a conta.

* Bloqueio Indeterminado

São os clientes que até o momento não tem nenhum acesso a conta, geralmente acontece por ele utilizar a plataforma diversas vezes de forma indevida.

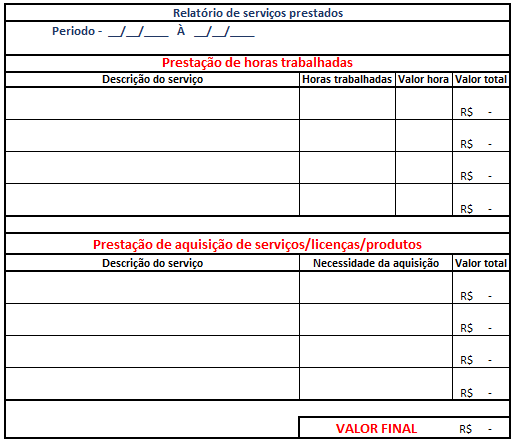
De acordo com a solicitação, o usuário poderá ver o motivo do bloqueio do cliente e assim encaminhar diretamente para o setor especifico para procurar alguma possível solução.

## 

## 12.2. Relatório De Cobrança Por Serviços Prestados

O relatório de cobrança tem como objetivo detalhar a geração do valor da cobrança e atualizar o cliente sobre os pontos desenvolvidos no projeto em um determinado período de tempo, assim como, quaisquer tipos de aquisições técnicas necessárias para o desenvolvimento do software.

No relatório temos a *Prestação de horas trabalhadas*, que engloba : descrição dos pontos desenvolvidos, as horas gastas em desenvolvimento do mesmo, assim como o valor da hora trabalhada. Temos também a *Prestação de aquisição de serviços/licenças/produtos-*trata-se de possíveis aquisições de licenças, servidores, perifericos e afins *-,* que engloba: Descrição do produto adiquirido, a necessidade desse produto(onde afeta no projeto), assim como o valor da aquisição.



Fonte: o autor

# 13. EMPREENDEDORISMO

Antes de entrarmos oficialmente na abordagem de empreendedorismo é necessário que citemos outros conceitos relativos à administração, para que posteriormente conseguimos contextualizar os conceitos de empreendedorismo.

Iniciando pela administração que é um processo que visa às ações de tomadas de decisões de planejamentos, organizações, direções e controles de recursos humanos, tecnológicos, físicos, financeiros, de informação e de conhecimento, entre outros, com o intuito de alcançar os objetivos das organizações (BERNARDI, 2012; COSTA NETO; CANUTO, 2010).

A Administração Clássica e Científica fornece um grande legado para as organizações, porém peca em considerar pouco ou quase nada o elemento humano. Segundo Costa Neto e Canuto (2010), os grandes pensadores da Administração começaram a perceber e apontar a importância do fator humano nas corporações. Os autores mencionam os elementos que podem influenciar no comportamento humano é o condicionamento pelo sistema social e pela biologia; necessidade de participação em grupos informais; comportamentos complexos e afetividade e sociabilidade.

Diante de todo o histórico, passando por todas as escolas e a partir do legado deixado por Taylor, Fayol, Ford, Mayo e Druker, entre outros, é possível entender que a Administração possui quatro funções:

* Planejamento;
* Organização;
* Direção;
* Controle.

## 13.1 A Tomada De Decisâo

Mesmo sendo algo natural aos administradores e empreendedores, a tomada de decisão precisa ser um processo tratado de forma racional, onde devem ser utilizadas técnicas por meio das quais se atingem objetivos organizacionais. Alguns fatores devem ser considerados para que as decisões sejam tomadas com alto grau de qualidade. Costa Neto (2007), estabelece alguns desses fatores, como por exemplo:

* **Decisões tomadas racionalmente:** é consequência de um cuidadoso processo de reflexão e do uso de metodologias apropriadas. No processo de tomada de decisão, as fortes emoções têm de ser extintas e desconsideradas, dando espaço a um processo racional;
* **Decisões baseadas em fatos e dados:** preza pela tomada de decisão com base em fatos e dados, ao invés de palpites ou opiniões subjetivas sem qualquer embasamento técnico. No processo de tomada de decisão, é importante transformar os dados em informações, bem como as informações em conhecimento;
* **Decisões economicamente pesadas:** considera que é de suma importância pesar os aspectos econômicos na tomada de decisão;
* **Decisões baseadas na experiência:** retrata a importância da experiência adquirida sobre determinados assuntos como fator de sucesso para a tomada de boas decisões;
* **Decisões visando ao futuro:** considera que se deve sempre pensar estrategicamente no longo prazo. Numa corporação, a visão como norteador estratégico é importantíssima no processo de tomada de decisão; 47 Revisões: Nome do revisor - Diagramação: Nome do diagramador - data EMPREENDEDORISMO.
* **Decisões consensuais**: quando há uma enorme complexidade no processo, em que diversos envolvidos enxergam ângulos distintos da problemática, o melhor é obter o consenso;
* **Decisões baseadas na consolidação de indicadores:** considera que as decisões precisam ser baseadas em indicadores que surgem por meio de métodos quantitativos adequadamente escolhidos;
* **Decisões justas e legais:** estabelece que as decisões precisam ser pautadas em critérios éticos, justos e que observam disposições legais;
* **Decisões criativas e inovadoras:** no processo de tomada de decisão, é necessária a capacidade de fugir dos moldes comuns, das soluções batidas, das determinações padronizadas, das decisões conhecidas de antemão;
* **Decisões corajosas x prudentes**: é necessário balancear a coragem e a prudência no processo de tomada de decisões;
* **Decisões inspiradas em paradigmas:** considera a prática do benchmarking para adoção de melhores práticas de mercado. A presteza na tomada de decisão é fundamental no dia a dia de um empreendedor, devido a toda incerteza associada aos cenários com os quais ele se depara.

## 13.2. Empreendedor – Empreendedorismo

O empreendedor é a pessoa que dá início a um negócio para realizar uma idealização de uma ideia ou projeto. Dito isso, o mesmo assumi os riscos e responsabilidades de seu negocio, sendo necessário estar inovando continuamente para que o projeto tenha longevidade. Essa definição abrange não somente os fundadores de empresas, mas também os membros da segunda ou terceira geração de empresas familiares e os gerentes‑proprietários que optam por adquirir empresas já existentes de seus fundadores.

Quando tratamos de empreendedorismo, podemos dizer que se trata de uma junção de pessoas e processos que tem como objetivo levar à transformação de ideias em oportunidades. A perfeita aplicação destas oportunidades pode levar à criação de negócios de sucesso.

Diante da utilização dos conceitos apresentados a empresa PIM-III foi fundada com o intuito de auxiliar seus clientes para que possam gerenciar e obterem acesso parcial ou consolidado de seus investimentos e criptoativos, onde essas informações serão passada através de relatórios e gráficos, e a empresa transmissora também possui acessos dos dados de seus clientes de forma ágil e eficaz.

O motivo principal de focarmos em nossos recursos de desenvolvimento para essa área é o amplo mercado de transações financeiras, que sempre apresenta novidades e seus maiores focos são agilidade, facilidades e o foco nos custos das transações, onde enxergamos uma possibilidade de fazer parte do crescimento do mercado de *blockchain*, pois os mesmo possuem características que atender a estes aspectos citados acima e por ser uma nova modalidade de transição, podemos observar que o mesmo se encontra em grande expansão, trazendo um futuro promissor para o mercado financeiro.

Outro ponto que nos motiva a expandir e aplicar nossos recursos em Blockchain é que uma de suas prioridades é fornecer um alto nível de segurança para o armazenamento de dados, sejam eles dados bancários de clientes, informações contratuais de empresas e históricos financeiros, pois devido a expansão de bancos virtuais, o Blockchain nos garante o sigilo de dados de nossos clientes e segurança de tais informações, o tornando uma fonte de confiança para nossa empresa.

# 14. ESTRATÉGIA DE NEGÓCIO

A estratégia de negócios está alinhada com a utilização do sistema, onde cada cliente terá acesso a uma base de dados independente, pois caso um cliente apresente uma demanda muito alta pode ser movido para um servidor independente sem impacto para os demais.

A principal estratégia de crescimento é o envolvimento das moedas virtuais, como o Bitcoin, sendo utilizado por todo o mundo, sendo uma moeda segura e com grande valor amarzenada nela, sendo interligada ao Blockchain. Primeiramente iremos voltar ao passado e relembrar onde surgiu o Blockchain.

O Blockchain surgiu em 30 de agosto de 2011 para ser um seviço de explorador de blocos do Bitcoin, na qual surgiu para ser uma estrutura de dados de todas as transações feitas com o Bitcoin, onde cria-se o registro atual dependendo das informações ateriores, sendo uma forma impossivel de alterar os registros passado sem que seja dectado e permitido, sendo uma forma mais confiavel e segura, sendo reforçada a autenticação e validação.

Podemos utilizar o Blockchain para outras empresas, na qual solicitarem um serviço para estrutura de dados de registro publico, atender cada vez mais os usuarios que optaram por este meio. Por tanto o meio dessas buscas o Blackchain e sua demanda ira começar a crescer e ter mais espaço no meio de empresas, sendo adptadas conforme a solicitações do mesmo.

Sendo assim, o Blockchain é mais facil de ser encontrado nos bancos, sendo transmitidos por toda sua rede de banco de dados, uma revolucionaria forma de gerar informações reforçadas e impossivieis de alterações.

Os preços também são competitivos e de acordo com o mercado, sendo que são definidos conforme o número de clientes. Para os clientes empresariais também há uma promoção de 20% de desconto mensal no 1 ano, para contratos firmados para mais de 5 anos.

Para a estratégia de negócio, utilizaremos a seguintes ferramentas como, Promoção e Desconto, na qual o intuito é nos trazer clientes por nossa competência, e beneficiar nós como empresa e também nossos clientes. De primeira situação iremos usar o desconto, na qual o cliente fechar 100% da documentação e Software, no final do produto, ficar como o esperado pelo cliente, o mesmo ira receber 2% de desconto no valor final do produto.

Como queremos que o cliente sempre nos procure para realizarmos mais trabalhos, a cada pessoa que ele indicar para gerarmos a documentação e Software, ao retornar para realizar melhoria no Software na qual criamos para ele, o mesmo ira receber uma promoção proporcional por cada indicação fechada conosco, Isso será permanecido também para o Cliente por indicação, e assim repentinamente para todos os clientes.

# 15. GESTÃO DE QUALIDADE

Gestão é o ato de organizar e monitorar um sistema para fazê-lo chegar aos resultados desejados, ou seja, direcionar e sincronizar os recursos para um único objetivo. Para realizar uma gestão, precisamos de um trabalho que envolva pessoas empenadas, construção de autonomia, definição de responsabilidades, gerenciamento de recursos, processos e definição de metas e objetivos.

Para definirmos qualidade, precisamos levar em conta o seu conceito teórico, visto que qualidade possui inúmeras definições e pode mudar sua definição, dependo do contexto empregado. Basicamente um produto ou serviço de qualidade é aquele que cumpre o que se propõe a resolver. Esse conceito foi evoluindo com o passar dos anos, e conseguimos notar que muitos produtos vendidos no passado deixaram de ser comercializados, por não possuírem as qualidades exigidas nos dias atuais, como por exemplo, vender pizza no Mcdonalds, visto que sua marca principal é a agilidade da entrega de seus produtos e linha de montagem de seus hamburgueses serem extremamente parecidas e quando tentaram implantar as vendas das pizzas em 1990, as pizzas não possuíam os mesmos padrões de qualidade de seus hamburgueses e tão pouco a agilidade de entrega dessas pizzas.

O conceito de qualidade é milenar, mas somente em meados do século XX a qualidade passou a ser, o foco das organizações. O aumento do consumo e do mercado literalmente os obrigou as tratarem de gestão de qualidade com mais cuidado e critério. Fazendo-se surgir às normas e padrões, com bases teóricas em grandes autores que tiveram papel fundamental estabelecendo as diretrizes para a qualidade.

## 15.1 . Ferramentas De Qualidade

Podemos considera para medição da qualidade todos os recursos que melhoram problemas e identificam defeitos, essas ferramentas não servem apenas para solucionar problemas, elas podem e devem ser usadas como um diferencial pela empresa, tornando a qualidade como um ponto estratégico e competitivo para a organização.

No âmbito empresarial, para que possamos criar estabilidade entre os processos, podemos utilizar métodos de estratégia, como por exemplo, desenvolver processos empresariais para controlar a variação das ferramentas adequadas, com o objetivo de reduzir falhas e melhorar de forma significante o entendimento e até mesmo a visão dos processos.

Também podemos citar as diretrizes que permitem classificar os objetivos das ferramentas da qualidade segundo Oliveira (1995):

* Facilitar a visualização e o entendimento dos problemas.
* Sintetizar o conhecimento e as conclusões.
* Desenvolver a criatividade.
* Permitir o conhecimento do processo.
* Fornecer elementos para o monitoramento dos processos.

## 15.2 Gráfico De Pareto

A ferramenta conhecido como Diagrama de Pareto ou apenas Gráfico mostra quais são os itens e quais as classes que possuem ordens nos números de ocorrências apresentando a soma acumulada, mostrando a prioridade e a urgência de cada item. Ele é considerado por muitos especialistas como uma das melhores ferramentas para diagnostico de problemas, e possui como característica a capacidade de fazer a descrição das causas que ocorrem na natureza de forma geral e do comportamento humano em suas mais diferentes facetas. Suas usabilidades são:

* Identificar problemas.
* Encontrar as causas que atuam em um determinado defeito.
* Identifica problemas e suas possíveis causas.
* Pode melhorar a visualização de uma ação ou atividade.
* Pode priorizar ações mais necessárias.
* Mostra os resultados obtidos por meio da melhoria aplicada a processos.
* Detalha as maiores causas do problema, mas em partes específicas, causando a eliminação destas.
* Estratifica uma ação.
* Identifica os fatores-chave pelos maiores impactos em uma empresa.
* Mostra as melhorias de um projeto.

## 15.3 Diagrama De Causa E Efeito De Ishikawa

Podemos dizer que é uma representação gráfica que organiza as informações da forma mais funcional possível, também chamado de “espinha de peixe”, esse recurso mostra as causas fundamentais de uma ação, bem como a forma pela qual é dirigida e como se formam as subcausas, levando ao resultado. Podemos usar o diagrama de Ishikawa nas seguintes situações, quando temos a necessidade de identificar quais são as possíveis causas de um determinado problema. Para obter uma visualização mais completa e ampla, bem como as causas e os efeitos dela decorrentes. Para classificar todas as possíveis causas, decompondo-as em subcausas. Assim, podemos verificar quais são os efeitos ou resultados consequentes. Tornou-se muito útil para identificar precisamente e mostrar as relações entre causas e efeitos de uma situação, bem como suas prioridades e também serve como ferramenta para analisar as causas dos defeitos, com o objetivo de melhorar o desempenho dos produtos, bem como para identificar possíveis falhas e poder aperfeiçoá-lo.

## 15.4 Planejamento Pdca

Planejamento PDCA (*Plan-Do-Check-Act)* é uma técnica utilizada para atingir a principal meta, ou mesmo o objetivo que precisa ser alcançado pela empresa. É necessário que a meta seja sempre muito claro, preciso, quantificável e realista. A ausência da definição especifica da meta é uma das razoes mais pertinentes e frequentes do fracasso de muitos projetos. Um plano de ação eficiente para a empresa precisa ter o método 5W2H, que é essencial para que esse plano tenha especificações de:

* Como fazer.
* O que fazer.
* Quem deve fazer.
* Quando fazer.
* Onde fazer.
* Por que fazer.
* Quanto investir.

## 15.5 Normas E Iso

As normas que constituem a NBR **ISO 9000** são partes de uma serie de técnicas que tratam a gestão da qualidade com exclusividade, como uma espécie de marca de qualidade. Desde então, o cliente final, que presta cada vez mais atenção nos quesitos relacionados a segurança e qualidade dos produtos e serviços, já que o símbolo NBR ISO é um sinônimo de qualidade, segurança e confiabilidade. Vale lembrar que a certificação não garante que a empresa seja perfeita já que existem falhas, problemas como qualquer outra, mas possuem um controle rígido sobre seus processos tento assim a capacidade de realizar um gerenciamento mais efetivo.

A NBR **ISO 9001** é parte de uma série de outras normas que compõem a série ISO 9000. Essas normas foram divulgadas no Brasil no final do ano 2000, e têm como base três outras normas:

* NBR ISO 9000 – Sistema de Gestão de Qualidade – fundamentos e vocabulário: traz a definição dos conceitos fundamentais usados na série de normas ISO 9000.
* NBR ISO 9001 – Sistema de Gestão da Qualidade – requisitos: mostra os principais requisitos necessários para que possamos implementar um Sistema de Gestão da Qualidade. Essa norma tem certificação.
* NBR ISO 9004 – Sistema de Gestão da Qualidade – diretrizes para a melhoria do desempenho: mostra as principais diretrizes para o constante aperfeiçoamento do sistema da gestão da qualidade. Pode também fornecer uma espécie de base conceitual para que possamos implementar as práticas que têm relação com a melhoria da qualidade.

Gestão da qualidade através da ISSO 9000.

# 16. CONCLUSÃO

Esse trabalho foi desenvolvido com base nas disciplinas Gerenciamento de Projetos, Programação Orientada a Objetos II, Projeto de Sistemas Orientado a Objetos, Desenvolvimento de Software Para Internet, e as disciplinas complementares Empreendedorismo e Gestão da Qualidade.

Foi realizado uma pesquisa sobre a tecnolgia Blockchain e desenvolvida uma especificação para um sistema de controle de Blockchain. Utilizamos diagramas UML e tabelas para demonstrar o comportamento, desenvolvimento e implantação do sistema, e desenvolvido o sistema nas versões web, mobile e desktop.

A tecnologia Blockchain vem conquistando espaço nos mercados, sobretudo de criptoativos, mas tambem já se faz presente em diversas modalidades de negócio, como o caso dos Smart Contracts, que além de praticidade, traz segurança para o cliente.

Devido a isso esse trabalho tem a proposta de criar um sistema administrativo para gerenciar a rede, os clientes, contratos, investimentos, entre outros. E temos como trabalho a continuidade do projeto, fazendo a implementação das especificações aqui apresentadas.

# 17. REFERÊNCIAS BIBLIOGRÁFICAS

Andoni, Merlinda & Robu, Valentin & Flynn, David & Abram, Simone & Geach, Dale & Jenkins, David & McCallum, Peter & Peacock, Andrew. **Blockchain technology in the energy sector: A systematic review of challenges and opportunities.** Elsevier, 2019.

D’Oriano, Luigi & Mastandrea, Giuseppe & Rana, Giuseppe & Raveduto, Giuseppe & Croce, Vincenzo & Verber, Matteo & Bertoncini, Massimo. **Decentralized blockchain flexibility system for Smart Grids: Requirements engineering and use cases.** IEEE International Conference and Workshop in Óbuda on Electrical and Power Engineering, Budapest, Hungary, 2018.

Walport M. **Distributed ledger technology: beyond blockchain.** Disponível em

<https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attach ment\_data/file/492972/gs-16-1-distributed-ledger-technology.pdf>. Acessado em 5 de março de 2019. 2016.

**Aplicativo mobile**

https://appinventor.mit.edu/

# APÊNDICE 1

**SCRIPTS DE CRIAÇÃO SQL**

create database db\_blockchain

use db\_blockchain

create table tb\_login

(

Id int identity,

IdCliente int,

usuario varchar(50),

senha varchar(50),

FOREIGN KEY (IdCliente) REFERENCES tb\_cliente(IdCliente)

)

create table tb\_plano

(

IdPlano int identity primary key,

NomePlano varchar(50),

Descricao varchar(500),

ValorPlano decimal(10,2)

)

create table tb\_cliente

(

IdCliente int identity primary key,

Email varchar(100),

Nome varchar(200),

DataNascimento datetime,

Cpf varchar(15),

IdPlano int,

FOREIGN KEY (IdPlano) REFERENCES tb\_plano(IdPlano)

)

create table tb\_endereco

(

Id int identity,

Logradouro varchar(300),

cep varchar(10),

IdCliente int,

FOREIGN KEY (IdCliente) REFERENCES tb\_cliente(IdCliente)

)

create table tb\_carteira

(

Id int identity,

IdCliente int,

IdMoeda int,

QuantidadeAcoes decimal(10,2),

FOREIGN KEY (IdCliente) REFERENCES tb\_cliente(IdCliente)

)

create table tb\_transacao

(

Id int primary key identity,

DataTransacao datetime,

IdCliente int,

TipoTransacao int,

IdMoeda int,

QuantidadeAcoes decimal(10,2),

FOREIGN KEY (IdCliente) REFERENCES tb\_cliente(IdCliente),

FOREIGN KEY (TipoTransacao) REFERENCES tb\_tipo\_transacao(Id),

FOREIGN KEY (IdMoeda) REFERENCES tb\_moeda(Id)

)

create table tb\_moeda

(

Id int identity primary key,

NomeMoeda varchar(50),

ValorMoeda decimal(10,2)

)

create table tb\_tipo\_transacao

(

Id int identity primary key,

Descricao varchar(50)

)

create table tb\_telefone

(

Id int identity primary key,

IdCliente int,

telefone varchar(16),

FOREIGN KEY (IdCliente) REFERENCES tb\_cliente(IdCliente)

)

insert into tb\_tipo\_transacao(Descricao) values('Compra')

insert into tb\_tipo\_transacao(Descricao) values('Venda')

insert into tb\_login(usuario, senha) values('gusta','12345')

insert into tb\_plano(NomePlano,Descricao,ValorPlano) values('Standard','plano mais básico',50.99)

insert into tb\_plano(NomePlano,Descricao,ValorPlano) values('Premiun','plano Intermediário',75.99)

insert into tb\_plano(NomePlano,Descricao,ValorPlano) values('Gold','plano mais completo',89.99)

CREATE TABLE [usuario] (

us\_cod integer(11) NOT NULL, nome varchar(100) NOT NULL, senha varchar(100) NOT NULL, status integer(11) NOT NULL, email varchar(100) NOT NULL,

CONSTRAINT [PK\_USUARIO] PRIMARY KEY CLUSTERED (

[us\_cod] ASC

) WITH (IGNORE\_DUP\_KEY = OFF)

) GO

CREATE TABLE [cliente] ( cl\_cod integer NOT NULL,

nome varchar(100) NOT NULL, cpf varchar(100) NOT NULL, endereco text NOT NULL,

) GO

CREATE TABLE [alerta] (

id integer(11) NOT NULL, nome varchar(100) NOT NULL,

id\_cliente integer(11) NOT NULL, inicial varchar(20) NOT NULL, final varchar(20) NOT NULL, data\_inicio date(20) NOT NULL, data\_fim date(20) NOT NULL, status integer(11) NOT NULL,

CONSTRAINT [PK\_ALERTA] PRIMARY KEY CLUSTERED (

[id] ASC

) WITH (IGNORE\_DUP\_KEY = OFF)

id integer(11) NOT NULL, nome varchar(100) NOT NULL, inicial varchar(20) NOT NULL, final varchar(20) NOT NULL,

data\_inicio date(20) NOT NULL, data\_fim date(20) NOT NULL, status integer(11) NOT NULL,

CONSTRAINT [PK\_PROMOCAO] PRIMARY KEY CLUSTERED (

[id] ASC

) WITH (IGNORE\_DUP\_KEY = OFF)

) GO

CREATE TABLE [produto] (

id integer(11) NOT NULL, id\_cliente integer(11) NOT NULL, nome varchar(100) NOT NULL, endereco varchar(100) NOT NULL,

CONSTRAINT [PK\_PRODUTO] PRIMARY KEY CLUSTERED (

[id] ASC

) WITH (IGNORE\_DUP\_KEY = OFF)

) GO

CREATE TABLE [consumo] ( id integer(11) NOT NULL,

id\_produto integer(11) NOT NULL, tipo integer(11) NOT NULL,

data date(20) NOT NULL, hora time(20) NOT NULL,

quantidade varchar(20) NOT NULL, valor varchar(20) NOT NULL,

CONSTRAINT [PK\_CONSUMO] PRIMARY KEY CLUSTERED (

[id] ASC

) WITH (IGNORE\_DUP\_KEY = OFF)

) GO

CREATE TABLE [pagamento] ( id integer(11) NOT NULL,

id\_produto integer(11) NOT NULL, id\_consumo integer(11) NOT NULL, vencimento date(20) NOT NULL, pagamento varchar(20) NOT NULL, status integer(11) NOT NULL,

valor varchar(20) NOT NULL, periodo varchar(100) NOT NULL, id\_promocao integer(11) NOT NULL,

CONSTRAINT [PK\_PAGAMENTO] PRIMARY KEY CLUSTERED (

[id] ASC

) WITH (IGNORE\_DUP\_KEY = OFF)

) GO

CREATE TABLE [clientes\_promo] ( id integer(11) NOT NULL,

id\_cliente integer(11) NOT NULL, id\_promo integer(11) NOT NULL, inicio date(20) NOT NULL,

fim date(20) NOT NULL,

CONSTRAINT [PK\_CLIENTES\_PROMO] PRIMARY KEY CLUSTERED (

[id] ASC

) WITH (IGNORE\_DUP\_KEY = OFF)

) GO

ALTER TABLE [alerta] WITH CHECK ADD CONSTRAINT [alerta\_fk0]

FOREIGN KEY ([id\_cliente]) REFERENCES [cliente]([cl\_cod]) ON UPDATE CASCADE

GO

ALTER TABLE [alerta] CHECK CONSTRAINT [alerta\_fk0] GO

ALTER TABLE [produto] WITH CHECK ADD CONSTRAINT [produto\_fk0]

FOREIGN KEY ([id\_cliente]) REFERENCES [cliente]([cl\_cod]) ON UPDATE CASCADE

GO

ALTER TABLE [produto] CHECK CONSTRAINT [produto\_fk0] GO

ALTER TABLE [consumo] WITH CHECK ADD CONSTRAINT [consumo\_fk0] FOREIGN KEY ([id\_produto]) REFERENCES [produto]([id])

ON UPDATE CASCADE GO

ALTER TABLE [consumo] CHECK CONSTRAINT [consumo\_fk0] GO

ALTER TABLE [pagamento] WITH CHECK ADD CONSTRAINT

[pagamento\_fk0] FOREIGN KEY ([id\_produto]) REFERENCES [produto]([id])

ON UPDATE CASCADE GO

ALTER TABLE [pagamento] CHECK CONSTRAINT [pagamento\_fk0] GO

ALTER TABLE [pagamento] WITH CHECK ADD CONSTRAINT

[pagamento\_fk1] FOREIGN KEY ([id\_consumo]) REFERENCES [consumo]([id]) ON UPDATE CASCADE

GO

ALTER TABLE [pagamento] CHECK CONSTRAINT [pagamento\_fk1] GO

ALTER TABLE [pagamento] WITH CHECK ADD CONSTRAINT

[pagamento\_fk2] FOREIGN KEY ([id\_promocao]) REFERENCES [promocao]([id]) ON UPDATE CASCADE

GO

ALTER TABLE [pagamento] CHECK CONSTRAINT [pagamento\_fk2] GO

ALTER TABLE [clientes\_promo] WITH CHECK ADD CONSTRAINT

[clientes\_promo\_fk0] FOREIGN KEY ([id\_cliente]) REFERENCES [cliente]([cl\_cod]) ON UPDATE CASCADE

GO

ALTER TABLE [clientes\_promo] CHECK CONSTRAINT [clientes\_promo\_fk0] GO

ALTER TABLE [clientes\_promo] WITH CHECK ADD CONSTRAINT

[clientes\_promo\_fk1] FOREIGN KEY ([id\_promo]) REFERENCES [promocao]([id])

ON UPDATE CASCADE GO

ALTER TABLE [clientes\_promo] CHECK CONSTRAINT [clientes\_promo\_fk1] GO

INSERT INTO `tcc`.`usuario` (`us\_cod`, `nome`, `senha`, `status`, `email`) VALUES (NULL, 'Administrador', '336699', '2', 'admin@pimiii.com.br');

# APÊNDICE 2

**MANUAL DE INSTALAÇÃO**

1. Insira o cd/ dvd com o sistema no leitor de cd/ dvd. Se a instalação não começar automaticamente, procure no disco o arquivo de instalação do programa, normalmente chamado Setup.exe ou Install.exe. Abra o arquivo para iniciar a instalação.
   * Insira o disco no computador e siga as instruções na tela. Talvez você receba uma solicitação de senha de administrador.
   * Se a instalação não começar automaticamente, verifique as configurações de Reprodução Automática.
     + Na caixa de pesquisa na barra de tarefas, digite **Configurações de Reprodução Automática** e selecione-as nos resultados
     + Certifique-se de que a Reprodução Automática esteja **Ativada**.
   * Você também pode escolher os padrões de Reprodução Automática para unidades removíveis e cartões de memória.
2. Siga os passos indicados na instalação.
3. Será necessária conexão com a internet para prosseguir com a instalação.

# APÊNDICE 3

**CONTRATO DE DESENVOLVIMENTO**

Pelo presente instrumento particular de contrato de Prestação de Serviços, as partes abaixo qualificadas, através de seus representantes legais no final assinados, resolvem ajustar as cláusulas seguintes:

CONTRATANTE:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,

Pessoa jurídica de direito privado inscrita no CNPJ/MF sob o nº. \_\_.\_\_\_.\_\_\_/\_\_\_\_-\_\_, Endereço: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, número \_\_\_\_, Bairro: \_\_\_\_\_, CEP \_\_\_\_\_-\_\_\_, Cidade: \_\_\_\_\_\_\_\_ . São Paulo

CONTRATADA:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,

Pessoa jurídica de direito privado inscrita no CNPJ/MF sob o nº. \_\_.\_\_\_.\_\_\_/\_\_\_\_-\_\_, Endereço: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, número:\_\_\_\_\_\_, Bairro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_, CEP \_\_\_\_\_\_-\_\_\_, Cidade: \_\_\_\_\_\_\_\_\_ - São Paulo.

**OBJETO DO CONTRATO**

O presente contrato tem por objetivo o desenvolvimento de um sistema de web de gerenciamento específico.

O período de vigência do contrato está a ser combinado com o CONTRANTE, podendo após ser definido também ser prorrogado automaticamente por períodos iguais e sucessivos, salvo em caso de comunicação por escrito por uma parte à outra com no mínimo 30 dias de antecedência.

**1 - A CONTRATANTE**

1.1 - Será responsável por definir os projetos a que se refere o item 10.1, especificando sua abrangência e prazo.

1.2 - Os projetos serão iniciados pela CONTRATADA em até 07 dias, contando da emissão de ADITIVOS a este Contrato nos quais deverão constar o escopo do projeto e prazo para a sua execução.

1.3 - Os trabalhos serão realizados pela CONTRATADA, de acordo com as especificações técnicas fornecidas pela CONTRATANTE.

**2 - CONDIÇÕES COMERCIAIS**

2.1 - O valor total do desenvolvimento objeto desse contrato será de xxxx (xxxx reais) que será dividido da forma abaixo:

2.2 - Pelo projeto pactuado no item “1”, os CONTRATANTES pagarão à CONTRATADA o valor de 40% no início de cada Modulo 40% na entrega do Modulo e 20% de cada módulo no final do projeto, após os testes e a integração com a completa documentação dos programas fontes dos módulos e de todas as interfaces utilizadas no projeto. O valor será depositado na conta indicada pela contratante mediante recibo ou Nota Fiscal.

2.2.1 - O valor mensal de manutenção do sistema será cobrado após a entrega de forma satisfatória do aplicativo, de acordo com entregas e escopo pré-definidos e aprovados por ambas as partes

2.3 - As faturas serão emitidas no primeiro dia útil de cada mês e terão o seu vencimento cinco dias útil após a respectiva entrega à CONTRATANTE, mediante protocolo.

2.3.1 Os preços serão reajustados anualmente pelo IGP-M (Índice Geral de Preços para o Mercado), ou na falta de referido índice, por outro que venha a substituí-lo.

2.3.2. O reembolso das despesas aludidas no item 2.5, será feito dentro em 5 (cinco) dias úteis, contados da apresentação dos respectivos comprovantes pela CONTRATADA à CONTRATANTE.

**3 - OBRIGAÇÕES DA CONTRATADA**

3.1. Executar os projetos de trata o item 1, com estrita observância das especificações técnicas indicadas pela CONTRATANTE.

3.1.1. Submeter a prévia aprovação, por escrito, de responsável indicado pela CONTRATANTE, de qualquer despesa adicional.

3.1.2. Garantir que a instalação, a elaboração de novos programas, objeto deste ajuste, não infrinjam quaisquer direitos de propriedade intelectual, tais como patentes e direitos.

3.1.3. Manter o mais absoluto sigilo sobre as informações, documentos e técnica transmitidos pela CONTRATANTE, para possibilitar a execução de projetos ora pactuados.

3.2. O suporte sera dado ao sistema diariamente apos as 18:00 para correções, exceto em casos de emergencias tais como parada do sistema.

**4. OBRIGAÇÕES DA CONTRATANTE**

4.1. Colocar à disposição da CONTRATADA todas as informações que esta necessite, para obtenção do objetivo do presente contrato.

4.2. Fornecer material para o desenvolvimento dos projetos contratados.

4.3. Não contratar direta ou indiretamente qualquer empregado da CONTRATADA envolvido na atividade objeto deste contrato, a contar da assinatura deste instrumento durante sua vigência e por um período de 24 meses após o término do contrato.

4.4. Responsabilizar-se por eventuais atrasos que possam ocorrer no projeto, decorrentes do não atendimento dos itens sob sua responsabilidade.

4.5. O desrespeito à clausula 4.3. prevista neste item do presente contrato, sujeitará a multa equivalente às seis últimas faturas mensais em favor da CONTRATADA, sem prejuízo de eventual ação de indenização.

**5.CONDIÇÕES GERAIS**

5.1. Serão de propriedade da CONTRATANTE todos os programas, módulos, manuais, documentação associada e sistemas, resultantes deste contrato, sendo proibido seu uso ou comercialização pela CONTRATADA.

5.2. É defeso à CONTRATADA ceder ou transferir, total ou parcialmente, os direitos e OBRIGAÇÕES decorrentes deste contrato

5.3. O presente contrato poderá ser rescindido pelas partes mediante notificação extrajudicial e escrito, com prazo de 30 dias permanecendo inalteradas as cláusulas e condições deste contrato. Poderá ainda o contrato ser rescindido independentemente de notificação por acordo entre as partes.

5.4. A CONTRATANTE poderá rescindir o presente contrato, independentemente de qualquer notificação judicial ou extrajudicial, nas seguintes hipóteses:

inadimplência reiterada de qualquer cláusula ou condição do presente contrato;

decretação de falência/insolvência ou pedido de concordata da CONTRATADA;

nos demais casos previstos na legislação em vigor.

5.5. Fica convencionado que a parte que reiteradamente infringir qualquer cláusula constante deste contrato pagará à outra parte multa equivalente aos três últimos faturamentos emitidos, sem prejuízo da rescisão automática ora pactuado.

5.6. Nenhuma das partes será responsável perante a outra por qualquer falha ou atraso no cumprimento das obrigações constantes do presente contrato, causados por casos fortuitos ou força maior.

E por estarem assim justas e acertadas, as partes firmam o presente instrumento em 2 (duas) vias de igual teor e forma, tudo na presença das duas testemunhas abaixo:

São Paulo, 30 de Junho de 2020

*CONTRATANTE \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,*

*CONTRATADA \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,*

*TESTEMUNHA 1:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_TESTEMUNHA2:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

# APÊNDICE 4

**CONTRATO DE PRESTAÇÃO DE SERVIÇOS**

**CONTRATANTE: XXXX** regularmente constituída, com sede na cidade de XXX, no Estado do XXXX, na Rua XXXX, CEP XXXXXX, inscrita no CNPJ/MF sob o nº. **XXXXX**, registrado na Junta Comercial do Estado do XXXXX.

**CONTRATADA: XXXXX**, regularmente constituída, com sede na cidade de XXXXX, no Estado do XXXX, na XXXXX, CEP XXXXX, inscrita no CNPJ/MF sob o nº. **XXXXX**, registrado na Junta Comercial do Estado do XXXXX.

As partes acima identificadas têm, entre si, justo e acertado o presente Contrato de Prestação de Serviços, que se regerá pelas cláusulas seguintes e pelas condições de preço, forma e termo de pagamento descritas no presente.

**DO OBJETO DO CONTRATO**

***Cláusula Primeira-*** O presente contrato implica na manutenção de sistemas informáticos por parte da **CONTRATADA** atendendo aos interesses do **CONTRATANTE** a partir da data da assinatura deste contrato.

**§1º:**O serviço ora contratado compreende a manutenção dos sistemas informáticos contratados, hospedagem e backup dos serviços. O suporte corresponde a 30h mensais via online, e 5h presenciais. Caso a **CONTRATANTE** deseje serviços ou módulos adicionais, deverá ser cobrado via valor hora definido no contrato.

**§2º:** No caso da realização eventual de algum serviço, que em razão da sua natureza, estiverem além da compatibilidade do valor ora contratado, ou não estiverem na especificação deste contrato, o valor será fixado por um novo contrato, em remuneração compatível com o trabalho.

**DA REMUNERAÇÃO**

***Cláusula Segunda-*** A **CONTRATANTE** se compromete ao pagamento dos serviços contratados na seguinte forma;

O valor de R$ 3.000 (três mil) reais mensais sempre até o dia 15 de cada mês.

**§1º:** Ocorrendo atraso, ficará o contratante sujeito ao pagamento de multa de 2% (dois por cento) calculado sobre o valor devido, mais juros de mora de 5.9% (cinco ponto nove por cento) ao mês.

**DOS DIREITOS E DEVERES**

***Cláusula Terceira -*** A **CONTRATADA** se responsabiliza, pela manutenção de sigilo de quaisquer informações ou documentos, de caráter técnico ou não, que venha obter em função deste contrato, sejam elas de interesse da **CONTRATANTE** ou de terceiros. Tais obrigações permanecem mesmo quando extinto este contrato

***Cláusula Quarta –*** A **CONTRATADA** não poderá terceirizar o serviço ora contratado, sem a aprovação prévia por escrito da **CONTRATANTE**.

***Cláusula Quinta -*** A **CONTRATADA** submeterá à prova da **CONTRATANTE**

todos os materiais que serão publicados.

***Cláusula Sexta –*** Todo o código fonte desenvolvido será de propriedade da **CONTRATANTE** e será via email após a finalização do pagamento, a contar da assinatura do contrato.

***Cláusula Quinta -*** A **CONTRATANTE** possui direito de exclusividade sob o código fonte.

***Parágrafo Primeiro -*** A **CONTRATANTE** perderá todos os direitos sobre o material criado pela **CONTRATADA** em caso de rescisão do presente contrato.

**DA RESCISÃO**

***Cláusula sexta–*** O presente contrato poderá ser desfeito antes de sua expiração, mediante aviso prévio, por escrito, de qualquer uma das partes, com o prazo de 30 (trinta) dias, contados do recebimento e expirados os quais, estará

Resolvido de pleno direito sem que caiba qualquer indenização às partes contratantes.

***Parágrafo Primeiro-***Em caso de desrespeito a qualquer termo previsto neste contrato e também da Legislação em vigor no Brasil será exigível ainda o pagamento de uma **multa em valor correspondente a 60% do valor do contrato**, além da possível responsabilização sobre o dano causado àquele que desmotivadamente teve sua expectativa de direito tolhida (parte prejudicada).

***Parágrafo Segundo-***O inadimplemento de qualquer das cláusulas por qualquer das partes constituirá em mora o inadimplente, sendo que aplicadas as sanções estabelecidas na legislação civil em vigor, até a rescisão do presente contrato, com ônus civil ao causador.

**DAS CONDIÇÕES GERAIS**

***Cláusula oitava–***A **CONTRATADA** não possuirá horários fixos de entrada e saída com a **CONTRATANTE**, uma vez que não existirá vínculo empregatício.

***Cláusula nona–***É livre ao **CONTRATADO** prestar serviços a outras pessoas ou empresas, fora do âmbito deste contrato.

**DO FORO**

***Cláusula décima–***As partes elegem o Foro de XXXXX, Estado do XXXX, para dirimir quaisquer dúvidas advindas do presente contrato.

Assim, por estarem justas e contratadas, as partes assinam o presente instrumento em duas vias de igual teor e forma, na presença de duas testemunhas abaixo identificadas.

XXXX, XX de XXX de 20XX.

*CONTRATANTE \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,*

*CONTRATADA \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,*

*TESTEMUNHA 1:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_TESTEMUNHA2:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

APÊNDICE 5

**GLOSSÁRIO DE DADOS**

* 1. Blockchain: Sistema distribuído e descentralizado que permite que todos os usuários saibam o que está acontecendo em tempo real e tenham acesso a todas as informações.
  2. Contratos inteligentes: Usualmente utilizados juntamente com o blockchain, podem ser usados para definir regras na rede.
  3. C#: linguagem de programação de propriedade da Microsoft.
  4. Rede centralizada: Rede onde todos os dados estão armazenados em um único local, caso apresente problema todos os demais que consomem o serviço são impactados.
  5. Rede descentralizada: Rede onde os dados são armazenados em vários locais, caso um dele apresente problema não causa impacto na rede ou serviço.
  6. Servidor: Local onde um sistema fica instalado e que, através de conexão com a internet disponibiliza acesso a um sistema, serviço e ou banco de dados.
  7. SQL Server: linguagem de banco de dados.