****

**AUTORES**

**TÍTULO**

Subtítulo (se houver)

**LONDRINA**

**2021**

**AUTORES**

**TÍTULO**

Subtítulo (se houver)

Documento apresentado como requisito parcial de aprovação da disciplina de Trabalho de Conclusão do Curso Superior em Tecnologia e Análise e Desenvolvimento de Sistemas da Faculdade Positivo Londrina.

Orientador: Prof. Me. Helton de Azevedo

**LONDRINA**

**2021**

OBS.: Esta folha deverá ser impressa no verso da folha de rosto (folha anterior)

Sobrenome, Nome.

Título: subtítulo / Nome Sobrenome. – Londrina, 2021.

xx f. *(quantidade de folhas da monografia)*

Orientador: XXXXXXX.

Trabalho de conclusão de curso (Graduação – Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas) – Faculdade Positivo Londrina – Escola de Tecnologia da Informação, Londrina, 2021.

1.Assunto. 2. Assunto. 3. Assunto. I. Título.

CDU

**AUTORES**

**TÍTULO**

Subtítulo (se houver)

Trabalho de conclusão de curso apresentado como requisito parcial para obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas no curso de Análise e Desenvolvimento de Sistemas da Faculdade Positivo Londrina, pela seguinte banca examinadora:

Londrina, \_\_\_\_ de \_\_\_\_\_\_\_\_\_\_\_ de 2021.

Banca Examinadora

|  |  |  |
| --- | --- | --- |
| Orientador(a): |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Prof. Me.  Faculdade Positivo Londrina |
|  |  |  |
| Coordenador(a) do Curso: |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Prof. Me.  Faculdade Positivo Londrina |
|  |  |  |
| Professor(a) Convidado(a): |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Prof. Me.  Faculdade Positivo Londrina |

Dedico este trabalho .....

**AGRADECIMENTOS**

Agradecemos....

**Epígrafe**

*“O futuro tem muitos nomes. Para os fracos é o inalcançável. Para os temerosos, o desconhecido. Para os valentes, é a oportunidade.”*

*Victor Hugo*

**RESUMO**

O estudo destaca O presente trabalho tem como objetivo. O objeto da pesquisa fala sobre o... Diante disso o quadro teórico busca definir. A metodologia desse estudo pode ser classificada como. Neste estudo como técnica para obtenção das informações analisadas foi utilizado um questionário baseado nos estudos, considerando a aplicação do questionário ocorreu, sendo que foram respondidos (xx) questionários. Quanto às conclusões. Diante disso

**Palavras-chave:** Contabilidade digital; Avanço tecnológico; Nota fiscal eletrônica;

**ABSTRACT**

The work developed and the procedures executed for the development of digital accounting software, focusing on service providers. The development is organized in: business process modeling, requirements gathering, application of construction methodology and software construction. The work is a set of methodologies and technologies with procedures used in the software development market. However, the software was developed for the issuance of accounts and expenses, reports, documents and launches, and with the differential in the issuance of notes. This software offers, it is possible to issue the note directly without needing to contact the city hall, thus optimizing the service and costs for a company that uses the software.

**Keywords**: digital Accounting; Technological advancement; Electronic invoice; Accounting software

**LISTA DE TABELAS**

[Quadro 1 – Exemplo Tabela de Requisitos 2](#_Toc10157555)

[Quadro 2 – Requisitos de Usuário Fonte: Autoria Própria 2](#_Toc10157556)

[Quadro 3 – Requisito Cadastrar Empresa 2](#_Toc10157557)

[Quadro 4 – Requisito Cadastrar Usuário 2](#_Toc10157558)

[Quadro 5 – Requisito Emissão 2](#_Toc10157559)

[Quadro 6 – Requisito Exibir Dashboard 2](#_Toc10157560)

[Quadro 7 – Requisito Cadastrar Guias e Boletos 2](#_Toc10157561)

[Quadro 8 – Requisito Cadastrar Documentos 2](#_Toc10157562)

**LISTA DE FIGURAS**

[Figura 1 - Exemplo de Diagrama de classe 2](#_Toc10157527)

[Figura 2 - Exemplo de Caso de uso 2](#_Toc10157528)

[Figura 3 – Planejamento do Projeto 2](file:///C:\Users\Marcus%20Vinicius\Downloads\TCC_2.5.docx#_Toc10157529)

[Figura 4 – BPMN do projeto 2](file:///C:\Users\Marcus%20Vinicius\Downloads\TCC_2.5.docx#_Toc10157530)

[Figura 5 – Diagrama de Caso de Uso do projeto 2](#_Toc10157531)

**LISTA DE ABREVIATURAS E SIGLAS**

CFC – Conselho Federal de Contabilidade

CETIC – Centro Regional de Estudos para o Desenvolvimento da Sociedade da Informação

SPED – Sistema Público de Escrituração Digital

PAC – Programa de Aceleração do Crescimento do Governo Federal

NFS-e – Nota Fiscal Eletrônica de Serviço

ISSQN – Imposto Sobre Serviço de Qualquer Natureza

INSS – Instituto Nacional do Seguro Social

SAAS – *“Software as a Service”*, em português “Software como Serviço”

BMPN – *“Business Process Model and Notation”*, em português “Modelo de Processo de Negócios e Notação”

OMG – *Object Management Group*

B2B – “*Business to business”*, em português “Empresa para outras Empresas”

UML – “Unified Modeling Language”, em português “Linguagem de Modelagem Unificada”

JSON – *JavaScript Object Notation*

DARPA – *Defense Advanced Research Projects Agency*

SLQ – “Structure Query Language”, em português “Linguagem de Consulta Estruturada”

**SUMÁRIO**

[1 PROBLEMA 14](#_Toc72921526)

[2 JUSTIFICATIVA 14](#_Toc72921527)

[2 OBJETIVO GERAL 15](#_Toc72921528)

[2.1 Objetivos Específicos 15](#_Toc72921529)

[3 PLANEJAMENTO 16](#_Toc72921530)

[3.1 Modelagem de Processos 16](#_Toc72921531)

[3.2 Tabelas de Requisitos 16](#_Toc72921532)

[3.3 Detalhamento dos Requisitos 17](#_Toc72921533)

[3.3.1 Cadastrar Empresa 17](#_Toc72921534)

[3.3.2 Emissão 17](#_Toc72921535)

[3.3.3 Exibir Dashboard 18](#_Toc72921536)

[3.4 CRONOGRAMA 18](#_Toc72921537)

[3.5 ORÇAMENTO 19](#_Toc72921538)

[4 ESPECIFICAÇÓES TÉCNICAS 20](#_Toc72921539)

[4.1 TECNOLOGIAS 20](#_Toc72921540)

[4.1.1 AngularJS 20](#_Toc72921541)

[4.1.2 NodeJS 20](#_Toc72921542)

[4.1.3 PostgreSQL 20](#_Toc72921543)

[4.1.4 GitLab 21](#_Toc72921544)

[4.2. ESPECIFICAçÕES E MODELAGEM 21](#_Toc72921545)

[4.2.1 Levantamento de Requisitos 21](#_Toc72921546)

[4.2.2 Tipos de Requisitos 21](#_Toc72921547)

[4.2.3 Modelagem de Software 22](#_Toc72921548)

[4.2.4 Diagrama de Classe 22](#_Toc72921549)

[4.2.5 Diagrama de Casos de Uso 23](#_Toc72921550)

[5 RESULTADOS E DISCUSSÃO 24](#_Toc72921551)

[REFERÊNCIAS 25](#_Toc72921552)

# 1 PROBLEMA

Em 2020, foi declarada pela OMS, uma pandemia global causada por um vírus que transmite a Covid-19. Este vírus é transmitido por uma pessoa infectada para um outra que está saudável, através do contato próximo por meio de tosse, catarro, espirro, superfície de objetos ou lugares que estejam contaminados, gotículas de saliva e diversos outros fatores. *[ref. 1]*

Dentre as várias medidas de prevenção recomendadas pela OMS, existe o distanciamento social mínimo entre as pessoas em um mesmo ambiente, tanto em lugares públicos quanto em lugares privados. Outro fator que também é de extrema importância para a prevenção da infecção, seria manter uma quantidade reduzida de pessoas no ambiente, permitindo assim uma circulação mais segura e controlada entre as pessoas no local. *[ref. 2]*

# 2 JUSTIFICATIVA

Sendo assim, observando este cenário apresentado anteriormente, surge o intuito do projeto que será o de auxiliar na realização do monitoramento dessas normas preventivas.

O Sistema desenvolvido em Python irá realizar a leitura de pessoas em um ambiente fechado e controlado, por meio de imagens de vídeo em tempo real capturadas pela webcam de um computador. O Software irá realizar o reconhecimento e a detecção da quantidade de pessoas que estão e que passaram pelo ambiente, verificando se houve ou não uma aglomeração de pessoas no local. Caso houver, será informado exibindo alertas visuais em tempo real de execução, essas informações também serão listadas em um relatório final após o término da execução do monitoramento.

# 2 OBJETIVO GERAL

Desenvolver um Sistema de monitoramento de pessoas, para o controle de aglomeração em ambientes fechados nos tempos de pandemia.

## 2.1 Objetivos Específicos

1. Monitorar imagens da webcam dentro de um ambiente controlado;
2. Realizar a detecção das pessoas nas imagens da webcam;
3. Emitir alertas visuais para o usuário administrador em caso de aglomeração;
4. Realizar a contagem de pessoas no ambiente controlado;
5. Emitir um relatório apresentando os dados capturados durante a execução do software;
6. Desenvolver o sistema conforme as especificações;

# 

# 3 PLANEJAMENTO

No presente capítulo se apresenta as fases de planejamento adotadas para o desenvolvimento do trabalho, assim fornecendo base para argumentos no mesmo apresentados.

## 3.1 Modelagem de Processos

A partir das necessidades do cliente, foi possível efetuar uma análise de como é realizado os processos de uma empresa de contabilidade, e a partir de entrevistas com o cliente, foi identificado que os processos não acontecem em tempo real.

3.1.1 Coleta de Requisitos

Contudo, após a coleta de todas as informações necessárias, a equipe de desenvolvimento, juntamente com o cliente por intermédio de entrevistas, realizou uma coleta de requisitos funcionais para o planejamento de um software, com a proposta de otimização dos processos contábeis, e levar a informação mais rapidamente para seus usuários, consequentemente reduzindo custos para o cliente e seus usuários.

## 3.2 Tabelas de Requisitos

No presente subcapitulo são descritos em forma de tabelas os requisitos coletados para o desenvolvimento do sistema.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Código | Requisitos | Descrição | Partes Interessadas | Prioridades |
| R1 | Cadastrar Empresa | O sistema deverá ser capaz de cadastrar empresas mantendo as de informações Razão Social, Nome Fantasia, CNPJ, Regime, E-mail, Endereço e telefone | Empresa que utilizará do serviço | Alta |
| R2 | Cadastrar Usuário | O sistema deverá ser capas de realizar o cadastro de Usuário mantendo as informações de Nome, Senha, Empresa, Perfil e E-mail | Setor financeiro que utilizara sistema | Alta |
| R3 | Emissão | O sistema poderá emitir relatórios com todas as informações de Emissão de Notas Fiscais, Despesas, Receita, Faturamento e DRE (Demonstração de Resultados do Exercício) | Gestor da empresa | Alta |
| R4 | Exibir Dashboards | O sistema deverá exibir Dashboards com informações relacionadas a relatórios de Receita, Despesas e DRE | Gestor da empresa | Média |
| R5 | Cadastro Guias Boletos | O sistema deverá ser capaz de realizar o cadastro de Guias e Boletos mantendo informações de Mês, Ano e anexo de arquivo em PDF | Empresa que utilizará o sistema | Alta |
| R6 | Cadastro Documentos | O sistema deverá ser capaz de realizar o cadastro de Documentos mantendo as informações de Tipo de Documento, Mês, Ano e anexo do Documento em PDF | Empresa que utilizará o sistema | Alta |

Quadro 2 – Requisitos de Usuário

Fonte: próprio autor (2021)

## 3.3 Detalhamento dos Requisitos

### 3.3.1 Cadastrar Empresa

O detalhamento do requisito Cadastrar Empresa, descrevendo sua função principal e seus dados de ação que o sistema deve realizar para que o requisito seja atendido com sucesso, demonstrando também seus efeitos colaterais.

|  |  |
| --- | --- |
| Cadastrar Empresa | |
| Função | Cadastrar Empresa |
| Descrição | Realiza o cadastramento de uma nova empresa no sistema a partir dos dados informados no ato do cadastro |
| Entradas | Razão Social, Nome Fantasia, CNPJ, Regime, Situação, Endereço, E-mail e Telefones |
| Fonte | Dados coletados a partir da empresa que utilizará o sistema |
| Saídas | Registro de cadastro de empresa na tabela EMPRESA no banco de dados |
| Ação | O cadastro da empresa deverá validar os campos de CNPJ, Razão Social e e-mail evitando o cadastro de duplicadas ou dados incorretos. Se houver dados duplicados o sistema deverá informar mensagem para o usuário que o CPF já existe no sistema. Se os dados de e-mail estiverem fora dos padrões o mesmo deverá informar ao usuário o formato padrão de e-mails aceitos. |
| Efeitos colaterais | Empresa não poderá ser cadastrado caso não houver CNPJ |

Quadro 3 – Requisito Cadastrar Empresa

Fonte: próprio autor (2021)

### 3.3.2 Emissão

O detalhamento do requisito Emissão, descrevendo sua função principal e seus dados de ação que o sistema deve realizar para que o requisito seja atendido com sucesso.

|  |  |
| --- | --- |
| Emissão | |
| Função | Emissão |
| Descrição | Realiza a emissão de Notas Fiscais a partir dos dados informados |
| Entradas | Tipo de relatório, Data de Início e Data de término |
| Saídas | Conjunto de informações de Notas Fiscais |
| Ação | O sistema deverá realizar uma busca no banco de dados trazendo informações de acordo com os dados informados. |

Quadro 5 – Requisito Emissão

Fonte: próprio autor (2021)

### 3.3.3 Exibir Dashboard

O detalhamento do requisito Exibir Dashboard descrevendo sua função principal e seus dados de ação que o sistema deve realizar para que o requisito seja atendido com sucesso.

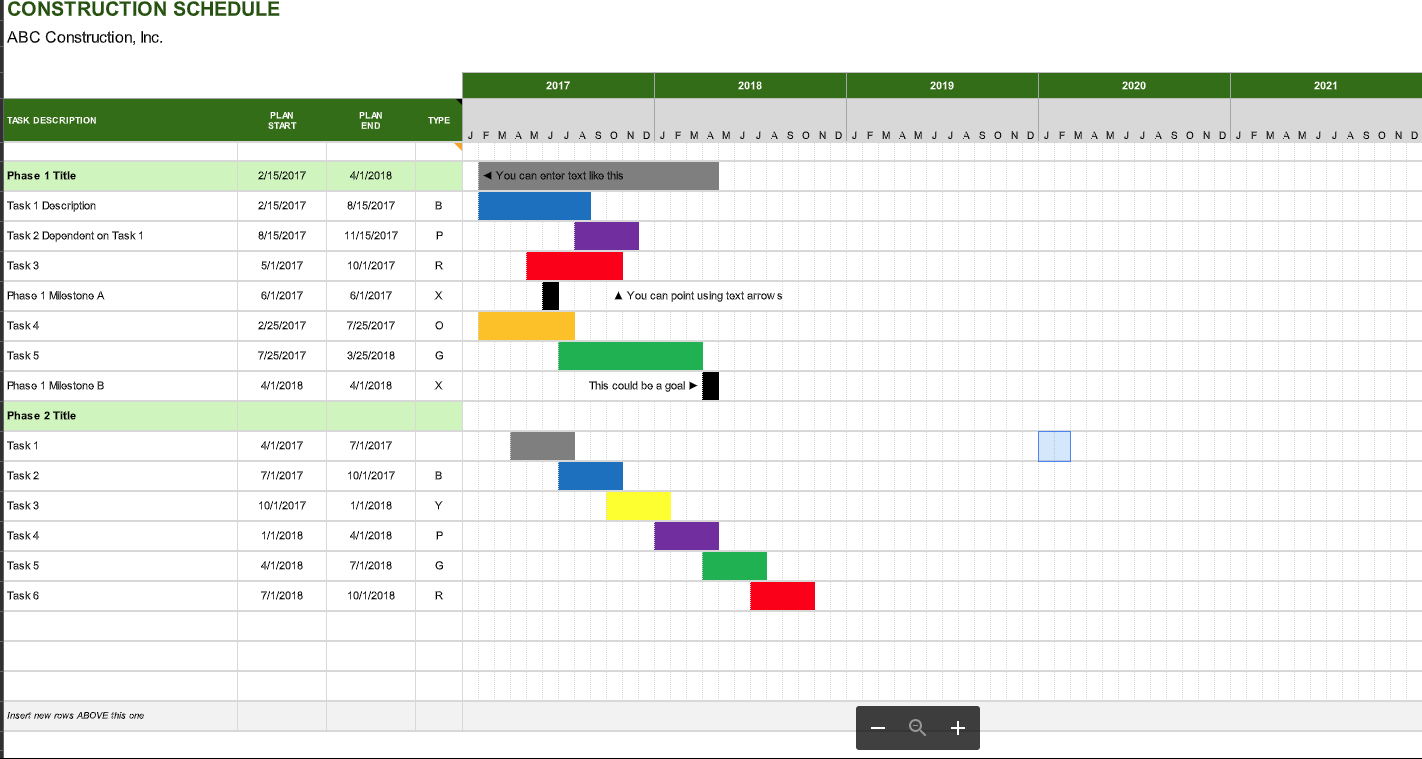
|  |  |
| --- | --- |
| Exibir Dashboard | |
| Função | Exibir Dashboard |
| Descrição | Exibir dashboards de Receitas e Despesas e Despesas por Categoria |
| Saídas | Conjunto de informações de Notas Fiscais |
| Ação | O sistema deverá mostrar os gráficos de Receitas e Despesas e Despesas por Categoria ao carregar a tela. |

Quadro 6 – Requisito Exibir Dashboard

Fonte: próprio autor (2021)

## 3.4 CRONOGRAMA

Descrever as atividades necessárias para a realização do projeto e os marcos principais de prazos para sua realização. Utilizar as datas do calendário geral de TCC para definir as entregas.



https://drive.google.com/file/d/0B\_jOMNZuPd-FUnFxSXFFTkprRlU/view

## 3.5 ORÇAMENTO

Para a realização do desenvolvimento e demonstração do projeto, serão necessários um Computador contendo uma webcam, que irá fornecer as imagens de vídeo, por fim, pessoas de exemplo e um ambiente fechado e controlado para testar e demonstrar a execução do software de monitoramento.

# 4 ESPECIFICAÇÓES TÉCNICAS

No presente capítulo é demonstrado de forma sucinta as tecnologias utilizadas na plataforma desenvolvida, e qual foram as metodologias aplicadas.

## 4.1 TECNOLOGIAS

Descrever tudo que for utilizado para a produção do TCC, incluindo softwares de código aberto ou comerciais, bibliotecas, frameworks e algoritmos de terceiros, bem como a sequência de atividades que serão realizadas para a construção da solução. Para cada tecnologia, a equipe deverá indicar aonde e de que forma está sendo utilizada dentro do projeto.

### 4.1.1 NumPy

Para o suporte de Arrays e matrizes multidimensionais.

O NumPy é uma biblioteca Python utilizada para a realização de cálculos em arrays e matrizes multidimensionais, muito utilizadas nas áreas da matemática e na ciência de dados.

Essa Biblioteca oferece várias funcionalidades que facilitam a execução de diversos cálculos numéricos, que são utilizados nas tarefas de Machine Learning, processamento de imagem, computação gráfica e tarefas matemáticas.

### 4.1.2 Time

Para realizar a medição de tempo de execução do programa.

Essa tecnologia basicamente irá realizar a medição do tempo de execução do nosso programa.

### 4.1.3 OpenCV

O OpenCV (Open Source Computer Vision Library) é uma biblioteca multi plataforma Open Source, ou seja, totalmente libre para a utilização para fins acadêmicos e comerciais para desenvolver aplicações relacionadas a área de visão computacional.

O OpenCV possui recursos de processamento de imagens e vídeo I/O, estrutura de dados, álgebra linear, GUI (Graphical User Interface ou Interface Gráfica do Usuário), controle de teclado e mouse, além de diversos algoritmos de visão computacional, como: filtros de imagem, calibração de câmera, reconhecimento de objetos e etc.

Essa biblioteca também possui suporte para ser utilizado nas tecnologias Java, Python e Visual Basic.

### 4.1.4 Math

O Math é uma biblioteca que irá auxiliar no cálculo da distância entre as pessoas que estarão sendo monitoradas pelas imagens de vídeo da webcam do computador, posicionados em um ambiente fechado e controlado.

### 4.1.4 Imutils

A biblioteca Imutils será utilizada tendo como responsabilidade realizar a detecção de um deslocamento na imagem e de traduzir uma imagem em OpenCV.

### 4.1.4 YOLO

A tecnologia YOLO (*You Only Look Once*) é uma ferramenta de visão computacional utilizada na detecção e classificação de objetos em tempo real. O YOLO também é totalmente de código aberto e livre, ou seja, pode ser usado por qualquer pessoal, de qualquer forma.

Visão Computacional é a ciência e tecnologia das máquinas que enxergam. É construído uma teoria e tecnologia para o desenvolvimento de sistemas artificiais que captam informações referentes a imagens ou qualquer outro dado de multidimensão.

A visão computacional estuda e descreve sistemas de visão artificial, implementados por hardware ou software.

Para exemplificar, as áreas que mais utilizam a visão computacional atualmente são: aplicações de controle de processos (robôs industriais, veículos autônomos), detecção de eventos, modelagem de objetos ou ambientes, reconstrução de cenas, reconhecimento de objetos, aprendizagem de máquina e restauração de imagens.

Neste método, todos os objetos que forem detectados, a região de interesse irá ser cercada por retângulos, com uma cor que indicará o tipo do objeto. Esse sistema é capaz de detectar até 80 classes de objetos diferentes em apenas uma imagem ou até mesmo em um frame de vídeo. É extremamente rápido e preciso, permitindo realizar ajustes de velocidade e precisão.

O YOLO é um método de detecção de objetos por passada única (*single pass*) que utiliza uma rede neural convolucional para extrair as devidas características (*features*). Ao contrário dos algoritmos concorrentes de detecção de objetos, no YOLO é necessário olhar pela imagem uma única vez para enviar esses dados à rede neural. Devido a essa característica ele recebeu esse nome (**Y**ou **O**nly **L**ook **O**nce – *“Você só olha uma vez”*), e também foi capaz de conseguir uma velocidade na detecção de objetos muito maior do que as tecnologias concorrentes, sem perder sua precisão.

A profunda rede neural que o YOLO utiliza é chamada de “Darknet” (*arquitetura*). Mesmo nome do framework utilizado para a implementação do detector. E também foi desenvolvido pelo próprio criador do YOLO, Joseph Redmon.

O YOLO consegue identificar os objetos na imagem com um simples problema de regressão.

Primeiramente ele divide a imagem em um grid de células, cada uma dessas células é responsável por fazer a predição de 5 caixas delimitadoras. Para cada caixa, a célula também faz a previsão de uma classe. Por fim, o valor de confiança para a caixa delimitadora e a predição da classe são combinados em uma pontuação final, que vai nos informar a probabilidade dessa caixa conter aquele objeto em específico.

Temos uma imagem abaixo exemplificando o funcionamento desses passos:

[IMAGEM – FONTE: ?]

Essa tecnologia, assim como outros sistemas de detecção de objetos, é de uso geral, ou seja, pode ser utilizado para vários propósitos diferentes, basta treinar para qualquer categoria de imagem desejada. O mesmo código utilizado para encontrar pessoas ou animais, também pode ser utilizado para outras finalidades, como encontrar outros objetos, por exemplo: carros, células cancerígenas em um tecido por meio de uma biópsia e etc.

### 4.1.4 Tkinter

## 4.2. ESPECIFICAçÕES E MODELAGEM

Este item deverá versar sobre a maneira que o software pensado. Em conjunto com o orientador a equipe deverá obrigatoriamente incorporar ao texto, diagramas que melhor descrevem o projeto. Exemplo: Modelo Entidade Relacionamento (MER), Diagrama Entidade Relacionamento (DER), Diagrama de Caso de Uso, Diagrama de Sequência, Diagrama de Classe, Diagrama de Implantação.

### 4.2.1 Levantamento de Requisitos

O levantamento de requisitos se consolida em técnicas aplicadas entre a equipe de desenvolvimento e seus clientes, para que assim facilitando a compreensão das necessidades, os desenvolvedores, juntamente com os clientes, tentam levantar e definir as necessidades dos futuros usuários do sistema a ser desenvolvido” (BEZERRA, 2015). Assim se tornando uma das etapas indispensáveis para o projeto.

### 4.2.2 Tipos de Requisitos

Os requisitos são divididos da seguinte forma, requisitos funcionais, requisitos não funcionais, requisitos de sistema e requisitos normativos, para que dessa forma apresente corretamente os dados necessários.

Os requisitos funcionais são aqueles que afetam de forma direta ou indireta a funcionalidade do sistema em seus processos funcionais.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Código | Requisitos | Descrição | Partes Interessadas | Prioridades |
| R1 | Cadastrar Empresa | O sistema deverá ser capaz de cadastrar empresas mantendo as de informações Razão Social, Nome Fantasia, CNPJ, Regime, E-mail, Endereço e telefone | Empresa que utilizará do serviço | Alta |
| R2 | Cadastrar Usuário | O sistema deverá ser capas de realizar o cadastro de Usuário mantendo as informações de Nome, Senha, Empresa, Perfil e E-mail | Setor financeiro que utilizara sistema | Alta |

Quadro 1 – Exemplo Tabela de Requisitos

Fonte: próprio autor (2021)

Com a divisão correta dos requisitos e a criação do documento de requisitos é possível realizar uma boa modelagem de software.

### 4.2.3 Modelagem de Software

A modelagem de software é realizada por meio de registros gráficos e textuais com o objetivo de representar as partes essenciais do sistema e considerando várias perspectivas (BEZERRA, 2015).

### 4.2.4 Diagrama de Classe

Na figura 1 ilustra um diagrama de classe que é a parte mais importante da linguagem UML (Guedes, 2018), representa de forma mais abstrata a estrutura de um sistema, apresentando suas classes, atributos, operações e as relações entre os objetos. É importante este tipo de representação pois é definido todas as classes do sistema e serve de base na construção de outros diagramas.

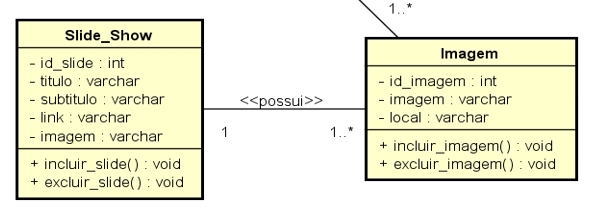


Figura 1 - Exemplo de Diagrama de classe

Fonte: próprio autor (2021)

### 4.2.5 Diagrama de Casos de Uso

Os casos de uso são utilizados para obter os requisitos funcionais do sistema, assim como funcionalidades, tarefas e serviços identificados como necessários, que os atores que interagem com o sistema poderão utilizar (Guedes, 2018). Desta maneira, documentando os comportamentos esperados para as funções do sistema.

Os casos de uso são representados por Elipses que contém dentro a ação a ser prestada por aquele caso, que sempre são representados com verbos denotando a ação que será realizada quando executado. A figura 2 apresenta um exemplo de caso de uso de um sistema de Pizzaria:



Figura 2 - Exemplo de Caso de uso

Fonte: próprio autor (2021)

# 

# 5 RESULTADOS E DISCUSSÃO

Além das conclusões pertinentes ao trabalho, os autores devem obrigatoriamente indicar possibilidades de Trabalhos Futuros derivados do desenvolvimento de seu protótipo.

Após a analise dos fatos descritos neste trabalho foi possível desenvolver para o cliente uma plataforma de....

O novo software desenvolvido, trouxe para o cliente uma plataforma inovadora para os processos...

O software traz para seu usuário informações detalhadas em tempo real das ...

O resultado obtido foi positivo, de acordo com o feedback recebido do cliente, com o software próprio, pode-se utilizar o ...

Os usuários reportaram que após a utilização da plataforma, foi possível poder analisar melhor as informações da sua empresa, identificando onde está ...

Observando todo o contexto, a visão do futuro é ....

Trabalhos futuros sobre o tema podem explorar...

# REFERÊNCIAS

CORTELLA, Mario Sergio**. Por que fazemos o que fazemos?**: 32.ed.São Paulo: Planeta, 2017

CETIC. **PANORAMA SETORIAL DA INTERNET Acesso a internet no Brasil**.Disponivel em:<[https://www.cetic.br/media/docs/publicacoes/6/Panorama\_Setorial\_11.pdf>. Acesso](https://www.cetic.br/media/docs/publicacoes/6/Panorama_Setorial_11.pdf%3e.%20Acesso) em: 23 mar. 2019

PostgreSQL. Disponivel em: <https://www.postgresql.org/about/>. Acessado em:

VMWARE. **Virtualização.** Disponivel em: <https://www.vmware.com/br/solutions/virtualization.html>. Acessado em: 01 abr. 2019

CONTAAZUL. **Sobre**. Disponivel em: <https://contaazul.com/sobre/>. Acessado em: 01 abr. 2019

BPMN, Business Process and Notation. Disponivel em: <http://www.bpmn.org/>. Acessado em: 04 abr. 2019

OMG, Object Management Group. Disponivel em: <https://www.omg.org/spec/BPMN/2.0/>. Acessado em: 04 abr. 2019

**BRASIL. Lei 13.429, de 31 de março de 2017. Lei Ordinária.** Diário Oficial [da] República Federativa do Brasil. Brasília, DF, pg. 1, 31 mar., 2017, Seção 1, pt 1.

NODEJS. Disponivel em: <https://nodejs.org/en/about/>. Acessado em: 15 abr. 2019.

UNICAMP, Universidade Estadual de Campinas, Studium.**Referências bibliográficas (nbr 6023):notas explicativas.** Disponivel em: <https://www.studium.iar.unicamp.br/biblio/guia.html>. Acessado em: 01 jun. 2019.