

Curso de Análise e Desenvolvimento de Sistemas

Amanda S. N. Brandino

Mateus Gonçalves Franco

Simão Pedro da Silva

Victor Seiichi Aguiar Shiira

Documentação de Desenvolvimento de Software

Sistema de Contas a Pagar

Sorocaba/SP

2021



**Amanda S. N. Brandino**

**Mateus Gonçalves Franco**

**Simão Pedro da Silva**

**Victor Seiichi Aguiar Shiira**

Documentação de Desenvolvimento de Software

Título: Sistema de Contas a Pagar

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Sorocaba, como parte dos pré-requisitos para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas

Orientador: Denilce de Almeida O. Veloso

Sorocaba/SP

2021

**Dedicatória**

Exemplo de dedicatória

Dedico este trabalho aos meus pais que com muita perseverança e paciência sempre incentivaram meus estudos e proporcionaram a educação que hoje tenho.

**Agradecimento**

Exemplo de agradecimento

Agradeço a todos que de uma forma ou de outra colaboraram para com este trabalho,

em especial aos meus grandes amigos Luke Skywalker e seu pai Darth Vader que me ajudaram com a força para conclusão deste trabalho.

Resumo

Este texto é um exemplo, use como modelo para a formação do seu TG, não leve em consideração o conteúdo do texto aqui apresentado, trata-se apenas de um exemplo. O resumo deve ser composto de uma sequência de frases concisas e afirmativas. Inicie com uma frase significativa, explicando o tema e o problema abordado, a seguir ressalte o objetivo geral, as técnicas e ferramentas, os resultados e a conclusão. Lembre-se que o trabalho será inicialmente conhecido pelo resumo aqui escrito, portanto, ele deve ser fidedigno ao trabalho desenvolvido. Note que não há abertura de parágrafo. O conteúdo deve ser disposto em parágrafo único. O resumo não deve ser muito longo, tente escrever no máximo 250 palavras, o objetivo do resumo é mostrar para o leitor qual é o escopo do TG. A ABNT recomenda que os resumos sejam escritos no impessoal, assim como todo o texto do TG. Por exemplo, deve-se utilizar as seguintes expressões: “verifica-se que", “conclui-se que", "percebe-se que pelos testes", "é válido supor", etc. Não é adequado, dizer: "conforme vimos no item anterior". Diz-se: "conforme visto no item anterior", ou, em vez de "dissemos que", "foi dito que". No caso do TG em grupo, é permitido usar a primeira pessoa do plural. Não indique referências.

Palavras-chave: Resumo; Pesquisa; Escopo

# Lista de Figuras

Figura 1 – Visão geral da gestão de pessoas 10

Figura 2 – Enfoque sistêmico nas organizações 12

Figura 3 – Esquema da estratégia empresarial e gestão de pessoas 13

Figura 4 – Fronteiras virtuais nas organizações 24

Figura 5 – Gestão das tecnologias da informação e gestão de pessoas 25

# Lista de Tabelas

Tabela 1 – Principais mudanças na Área de Recursos Humanos (PricewaterhouseCoopers) 17

Tabela 2 – Principais mudanças na Área de Recursos Humanos (Chiavenato) 22

Tabela 3 – Resumo do Estudo de Caso 1 29

Tabela 4 – Ferramentas de T.I. e seu relacionamento com os processos da ARH (estudo de caso 1) 30

Tabela 5 – Ferramentas de T.I. e seu relacionamento com os processos da ARH (estudo de caso 2) 36

# Índice: Inserir o índice dos títulos e subtítulos.

# Introdução(nos títulos use letra arial, 14, negrito)

A introdução deve começar com uma apresentação geral do assunto do trabalho. Descreva o caso a ser estudado de maneira sucinta, descreva a organização onde será aplicado o estudo. Justifique a escolha do tema. Comente quais as ferramentas, tecnologias, metodologias, técnicas, modelos, etc. que serão utilizados para solucionar o problema apresentado.

Descreva também o objetivo do trabalho de forma clara. Evite colocar objetivos que não poderão ser atingidos ou mensurados, deixe claras as limitações (fronteiras) do seu trabalho, porém não justifique possíveis problemas encontrados. Os problemas deverão ser relatados nas seções seguintes, mas não devem aparecer no objetivo. Use letra Arial, tamanho 12, parágrafo com espaçamento 1,5.

# Embasamento teórico.

Esta seção deve ser utilizada quando o software desenvolvido tenha como escopo um tema que necessite uma explicação mais abrangente para o leitor. Por exemplo, no caso em que o software desenvolvido utilizar IoT (Internet das Coisas) para a solução do problema, apresentar os conceitos de IoT, indicando as referências utilizadas conforme padrão ABNT.

O nível de detalhamento desta seção deve ser discutido com o orientador. O título da seção pode ser o assunto a ser apresentado por exemplo: “Conceitos de IoT”.

Obs: Ao inserir uma imagem é obrigatório numerar, descrever e indicar a fonte como no exemplo abaixo:

*Figura 1 - Site Amazom.com em 1995*



*[[1]](#footnote-1)Fonte: Disponível em: https://www.businessinsider.com. Acesso em: 13 set. 2019.*

Caso a figura tenha sido retirada de livros ou artigo, indicar a fonte e não esqueça de citá-la no texto.

# 3. Análise de Requisitos

# 3.1 Visão geral do Produto

# O objetivo deste projeto é desenvolver uma aplicação web responsiva para automatizar e auxiliar nos processos do setor de contas a pagar da empresa Ferramentaria Sorocaba, que atualmente é pouco informatizado e, na prática, não é sistematizado, minimizando o tempo gasto e reduzindo a incidência de erros causados pela repetição destes processos. Um dos principais focos é a intuitividade e facilidade de uso, pois o principal problema relatado pelos usuários do sistema atual é a sua baixa eficiência e dificuldade de utilização.

As funcionalidades propostas para o novo sistema inclui:

- Gerenciamento de Fornecedores

- Gerenciamento de Categorias de Contas

- Gerenciamento de Categorias de Pagamento (Dinheiro, Cheque, Banco)

- Lançamentos de Contas a Pagar;

- Relatórios Diversos

- Monitoramento de Contas em Atraso;

# 3.2 Descrição e Delimitação do problema

A Ferramentaria Sorocaba conta um setor de contas a pagar com 3 funcionários, sendo 1 líder e 2 auxiliares, divididos em dois turnos - manhã e tarde. No atual sistema de controle, os lançamentos são realizados em uma planilha Excel, onde são registradas todas as contas pendentes com seus respectivos dados como, número do documento, que pode ser de uma nota fiscal, pedido ou boleto, o dia do lançamento dos dados na planilha, a descrição dos documentos, o valor, a data de vencimento e a situação deste documento.

O controle mensal das movimentações são feitos utilizando planilhas individuais para cada mês. Também é possível de visualizar um resumo do ano separado por meses, em que é apresentado o total geral, os valores que foram pagos e o total pendente de pagamentos.

A utilização de planilhas para controle das atividades da empresa causa problemas de controle, pois os dados não são confiáveis visto que pode ocorrer a alteração de informações e até mesmo exclusão de dados essenciais causando prejuízos com multas por atraso em pagamentos.

3.3 Descrição da técnica utilizada para levantamento dos requisitos

A Metodologia para levantamento de dados foi entrevista com o cliente. Nessas entrevistas foram levantas as informações necessárias para que pudesse chegar aos requisitos essenciais para o funcionamento do aplicativo, e que foram essenciais para a elaboração de protótipos para posterior validação junto ao cliente.

# 3.4 Requisitos de Software

O Desenvolvimento de um software demanda o planejamento e em especial as definições previas dos requisitos do sistema. Deste modo é importante saber distinguir a diferença entre requisitos funcionais e requisitos não funcionais.

De acordo com Sommerville (2011) requisitos funcionais são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a

entradas específicas e de como o sistema deve se comportar em determinadas situações. Em alguns casos, os

requisitos funcionais também podem explicitar o que o sistema não deve fazer.

por outro lado, requisitos não funcionais são restrições aos serviços ou funções oferecidos pelo sistema. Incluem restrições de timing, restrições no processo de desenvolvimento e restrições impostas pelas normas. Ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, muitas vezes, aplicam-se ao sistema como um todo.

3.4.1 Requisitos Funcionais

A seguir estão enumerados os requisitos funcionais definidos pelo nosso levantamento e pelo pedido do cliente, a empresa Ferramentaria Sorocaba. Para isso foram realizadas visitas e levantado questionamentos à empresa para se ter uma visão geral do sistema a ser desenvolvido e os requisitos necessários para seu desenvolvimento.

**RF01 - Gerenciar Funcionário**

O sistema deve permitir o supervisor cadastrar, editar ou excluir uma ou mais contas de usuários, preenchendo os campos com seus dados. Apenas o Supervisor tem acesso a essa funcionalidade.

**RF02 – Login do Funcionário**

Uma interface de login deverá ser exibida ao usuário, caso o usuário já esteja cadastrado ele poderá acessar a tela principal do sistema e demais funcionalidades.

**RF03 – Fazer logout**

O sistema deve redirecionar o usuário à interface de login após ser encerrado o seu acesso.

**RF04 – Recuperar Senha**

O usuário poderá recuperar sua senha através do e-mail cadastrado.

**RF05 – Emitir Relatórios**

O sistema exibirá uma tela de relatório de contas a pagar, permitindo a exibição na tela ou impressão dos relatórios.

**RF06 – Cadastrar Contas a Pagar**

O funcionário ou supervisor poderão fazer o cadastro das contas a serem pagas. Para isso será exibida uma tela para cadastro de contas onde serão inseridos os dados das contas.

**RF07 – Gerenciar Tipos de Contas**

O sistema deverá permitir a consulta e visualização das contas a pagar por categoria.

**RF08 – Gerenciar Fornecedores**

O sistema deve permitir que o usuário cadastre os fornecedores e controle os valores que deverão ser gastos com eles.

**RF09 – Atualizar Contas a Pagar**

A atualização das contas a serem pagas poderão ser feitas por qualquer usuário cadastrado. Será exibida uma interface para que o usuário atualize e/ou corrija os dados das contas.

**RF10 – Pagamento das Contas a Pagar**

O sistema deverá exibir uma tela de pagamento de contas onde aparecerá as contas próximas ao vencimento, onde poderá ser feita a impressão do boleto dessas contas.

**RF11 - Gerenciar Tipos de Pagamentos**

O sistema deverá permitir a consulta e visualização dos pagamentos por categoria.

**RF12 - Confirmação de Pagamento**

Ao tentar finalizar o processo de pagamento o sistema deverá exibir uma tela de confirmação da efetuação do pagamento.

3.4.2 Requisitos Não Funcionais

Os requisitos não funcionais foram levantados pela nossa equipe e revisados pelo cliente, com foco em estabelecer restrições e as características do sistema.

**RNF01 - Desempenho**

O sistema deverá responder em tempo aceitável, e ter acesso rápido às informações, tendo um tempo de resposta de no máximo 1 segundo.

**RNF02 - Integridade/segurança**

Apenas usuários com privilégios de acesso de Administrador poderão cadastrar e atualizar dados de funcionários cadastrados.

**RNF03 - Facilidade de Uso**

A interface deverá ser de fácil entendimento e autoexplicativa ao usuário, facilitando o uso e compreensão das funcionalidades oferecidas pelo sistema ao usuário.

**RNF04 – Portabilidade**

O sistema poderá ser acessado em mais de uma plataforma (Desktop e mobile).

**RNF05 – Confiabilidade**

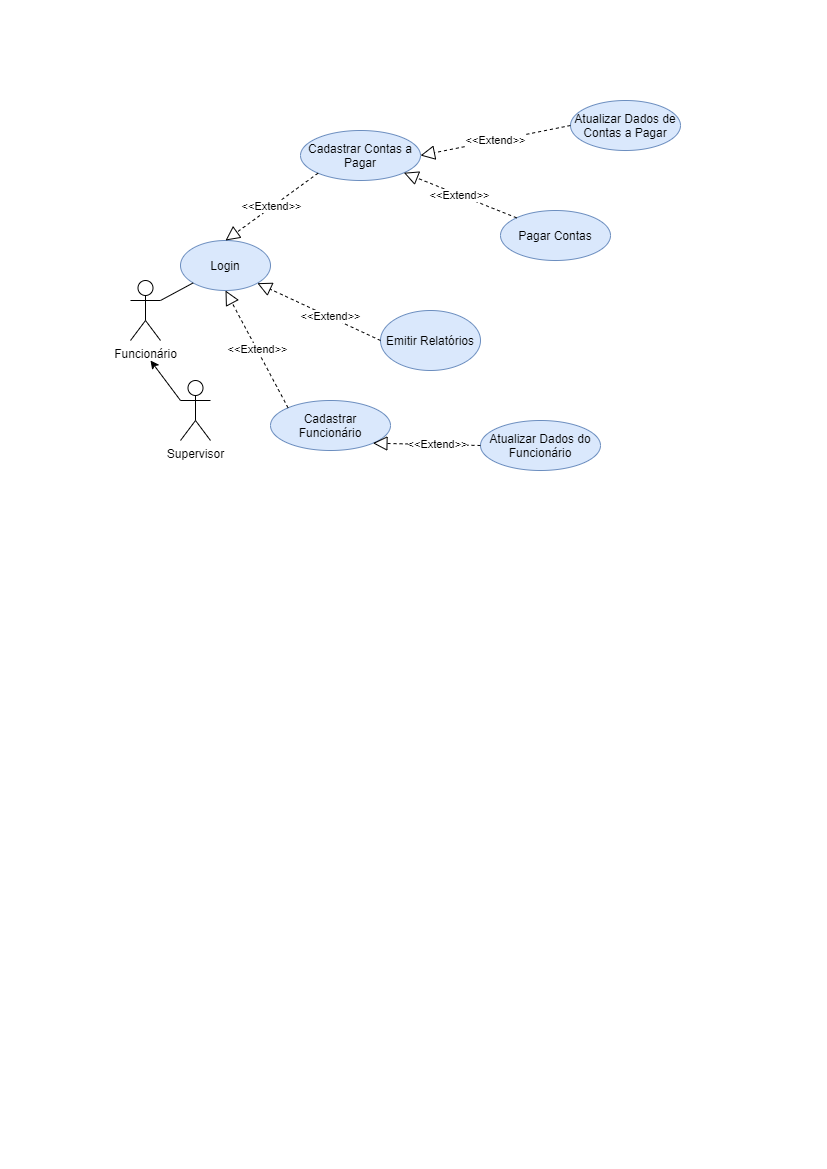
O sistema deverá apresentar uma taxa baixa de erros e falhas a um nível aceitável ao usuário.

**RFN06 – Cache de Senha**

O sistema não permitirá cache de senha. Deverá ser feita uma autenticação a cada acesso novo.

**3.4.3 Diagrama de Casos de Uso e Descrição dos Casos de Uso**

Figura 1 - Diagrama de Casos de Uso

****

Fonte: Elaborada pelos autores.

**Descrição dos Casos de Uso**

Quadro 1. Caso de uso – Login Funcionário

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF02: LOGIN FUNCIONÁRIO | |
| **Ator Principal** | FUNCIONÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Funcionário deve estar cadastrado no sistema | |
| **Pós-Condição** | Funcionário acessa o sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – Inserir o nome de usuário e senha. | |  |
|  | | 2 – Verificar nome de usuário e senha. |
|  | | 3 – Se nome de usuário e senha estiverem corretos: Passo 7; Senão: Passo 4. |
|  | | 4 – Informar mensagem de erro. |
| 5 – Informar nome de usuário e senha novamente. | |  |
|  | | 6 – Repetir passo 2. |
|  | | 7 – Mostrar tela inicial. |

Quadro 2. Caso de uso – Cadastrar Funcionário

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF01: CADASTRAR FUNCIONÁRIO | |
| **Ator Principal** | SUPERVISOR | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Estar com todos os dados do funcionário em mãos. Possuir nível de acesso de Supervisor. | |
| **Pós-Condição** | Funcionário é cadastrado | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – Após efetuar login no sistema, o funcionário acessa a opção Usuários/Cadastro de Usuário. | |  |
|  | | 2 – Exibir tela de cadastro de usuário. |
| 3 – Inserir os dados do funcionário a ser cadastrado e clicar em salvar. | |  |
|  | | 4 – Sistema verifica se todos os campos obrigatórios estão preenchidos. |
|  | | 5 – Se todos os campos obrigatórios estiverem preenchidos: Passo 9. Senão: Passo 6. |
|  | | 6 – Exibir mensagem de erro. |
| 7 – Preencher campos obrigatórios não preenchidos e clicar em salvar. | |  |
|  | | 8 – Repetir passo 4. |
|  | | 9 – Salvar as informações. |
|  | | 10 – Exibir mensagem de confirmação. |

Quadro 3. Caso de uso – Atualizar Dados do Funcionário

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF03: ATUALIZAR DADOS DO FUNCIONÁRIO | |
| **Ator Principal** | FUNCIONÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Funcionário deve estar cadastrado no sistema. | |
| **Pós-Condição** | Dados do funcionário são atualizados. | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – Após efetuar o login no sistema, o usuário acessa a opção Usuários/Alterar dados | |  |
|  | | 2 – Exibir tela com os dados do funcionário. |
| 3 – Alterar os dados necessários. | |  |
|  | | 4 – Sistema verifica se todos os campos obrigatórios estão preenchidos. |
|  | | 5 – Se todos os campos obrigatórios estiverem preenchidos: Passo 9. Senão: Passo 6. |
|  | | 6 – Exibir mensagem de erro. |
| 7 – Preencher campos obrigatórios não preenchidos. | |  |
|  | | 8 – Repetir passo 6. |
|  | | 9 – Salvar as alterações. |
|  | | 10 – Exibir mensagem de confirmação. |

Quadro 4. Caso de uso – Emitir Relatórios

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF04: EMITIR RELATÓRIOS | |
| **Ator Principal** | FUNCIONÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Estar logado no sistema. | |
| **Pós-Condição** | Relatório é exibido na tela ou impresso. | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – Após efetuar login no sistema, o funcionário acessa a opção “Emitir Relatório”. | |  |
|  | | 2 – Exibir tela de emissão de relatórios. |
| 3 – Funcionário seleciona de qual tabela deseja que seja emitido o relatório, caso deseje filtra os dados das tabelas e seleciona exibir. | |  |
|  | | 4 – Sistema exibe as informações. |
| 5 – Caso seja necessária a impressão do relatório, usuário clica em Imprimir. | |  |
|  | | 6 – Se a opção imprimir foi selecionada, imprimir o relatório. |

Quadro 5. Caso de uso – Cadastrar Contas a Pagar

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF05: CADASTRAR CONTAS A PAGAR | |
| **Ator Principal** | FUNCIONÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Estar com todos os dados da conta em mãos. | |
| **Pós-Condição** | Conta é cadastrada. | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – Após efetuar login no sistema, o usuário acessa a opção Contas/Cadastro de Conta | |  |
|  | | 2 – Exibir tela de cadastro de contas a pagar. |
| 3 – Funcionário insere os dados da conta e clica em salvar. | |  |
|  | | 4 – Sistema verifica se todos os campos obrigatórios estão preenchidos. |
|  | | 5 – Se todos os campos obrigatórios estiverem preenchidos: Passo 9. Senão: Passo 6. |
|  | | 6 – Exibir mensagem de erro. |
| 7 – Preencher campos obrigatórios não preenchidos e clicar em salvar. | |  |
|  | | 8 – Repetir passo 4. |
|  | | 9 – Salvar as informações. |
|  | | 10 – Exibir mensagem de confirmação. |

Quadro 6. Caso de uso – Atualizar dados de Contas a Pagar

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF06: ATUALIZAR DADOS DE CONTAS A PAGAR | |
| **Ator Principal** | FUNCIONÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Conta deve estar cadastrada. | |
| **Pós-Condição** | Conta é atualizada. | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – Após efetuar login no sistema, o usuário acessa a opção Contas/Atualizar Dados | |  |
|  | | 2 – Exibir tela de contas cadastradas. |
| 3 – Funcionário seleciona a conta que deseja alterar o cadastro. | |  |
|  | | 4 – Sistema exibe os dados da conta selecionada. |
| 5 – Usuário altera os dados da conta e clica em salvar. | |  |
|  | | 6 – Sistema verifica se todos os campos obrigatórios estão preenchidos. |
|  | | 7 – Se todos os campos obrigatórios estiverem preenchidos: Passo 11. Senão: Passo 8. |
|  | | 8 – Exibir mensagem de erro. |
| 9 – Preencher campos obrigatórios não preenchidos e clicar em salvar. | |  |
|  | | 10 – Repetir passo 4. |
|  | | 11 – Salvar as informações. |
|  | | 12 – Exibir mensagem de confirmação. |

Quadro 7. Caso de uso – Pagar Contas

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF07: PAGAR CONTAS | |
| **Ator Principal** | FUNCIONÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Informações referentes à(s) conta(s) terem sido lançadas no sistema. | |
| **Pós-Condição** | Gerar arquivo com informações das contas a serem pagar. | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – Após efetuar login no sistema, o usuário acessa a opção “Pagar Contas”. | |  |
|  | | 2 – Exibir tela de contas em aberto. |
| 3 – Usuário seleciona as contas que deseja pagar e clica em salvar. | |  |
|  | | 4 – Gerar um arquivo que contém todas as contas selecionadas e informações necessárias para realizar o pagamento. |

4. Projeto Detalhado do Software

Este item poderá ter suas seções alteradas com a autorização do orientador. As modificações podem ser decorrentes do emprego de um Método de Processo de Software específico. Por exemplo, se o desenvolvimento for na área de jogos/jogos educativos o aluno poderá seguir outras metodologias por ex. Extreme Game Development(XGD) ou alguma sistemática indicada por algum especialista no assunto. O mesmo pode ocorrer com desenvolvimento ágil para aplicações móveis ou web.

4.1 Arquitetura da aplicação Atual

Apresentar de maneira sucinta, qual foi o modelo arquitetural ou o padrão de projeto escolhido para o projeto. Por exemplo, foi utilizado o padrão MVC (model, view, controller), etc. É interessante incluir figuras facilitando o entendimento dos componentes.

4.2 Tecnologias utilizadas e APIs

Descrever as tecnologias que serão utilizadas para desenvolvimento da aplicação, principalmente se for uma tecnologia nova. Exemplo: nova linguagem, framework, banco de dados, API ou hardware. Se necessário podem ser incluídas subseções. Indicar referências.

Exemplo:

* **OpenCV**

OpenCV[[2]](#footnote-2), também chamado de Open Source Computer Vision, é uma biblioteca de visãocomputacional. Inicialmente, foi desenvolvida pela Intel, mas hoje é mantida por uma ampla comunidade de programadores independentes, empresas e universidades, sob a licença aberta BSD. O desenvolvimento está ativo, com o último lançamento estável em julho de 2019.

* YouTube API

A YouTube API permite adicionar funcionalidades do YouTube em sites e aplicativos através de um serviço REST. A figura 5 mostra a Try this API, um console que se comunica com a YouTube API. Inserindo-se a url e os parâmetros a API apresenta a resposta......etc etc....

4.3 Modelo de dados

**4.3.1 Modelo Conceitual**

Apresentar o modelo de dados que foi utilizado na aplicação indicando o tipo de banco de dados utilizado para prover a persistência dos dados (relacional, não relacional). Poderá ser usado o Diagrama Entidade-Relacionamento (DER).

4.3.2 Modelo Lógico

Definir as entidades, atributos, relacionamentos domínios e validações. Se for necessário incluir um dicionário de dados com detalhamento dos atributos, abrir uma nova subseção. O Script das tabelas pode ser colocado no Apêndice.

Se o modelo de banco de dados não for o relacional (NoSQL) apresentar a estrutura do documento agregado.

4.3.3 Diagrama de Classes

Deverá ser utilizado se o desenvolvimento utilizar orientação a objetos.

4.4 Diagrama de Sequência

É um diagrama de comportamento dinâmico que procura determinar a sequência de eventos que ocorrem em um determinado processo, identificando quais mensagens devem ser disparadas entre os elementos envolvidos e em que ordem. Somente os processos mais relevantes na aplicação deverão ser representados.

4.5 Diagrama de Atividades

O Diagrama de Atividades é um diagrama comportamental (que especifica o comportamento do software), e através dele podemos modelar partes do comportamento de um software. Este diagrama deverá ser utilizado para documentar o aspecto funcional (não estrutural) do software, quando é necessário representar o fluxo da informação que o software trabalhará.

4.6 Diagrama Estado e Diagrama de Pacotes

Estes diagramas devem ser incluídos caso o orientador solicite.

4.7 Interfaces com o usuário

Apresentar aqui as interfaces com o usuário acompanhada de uma pequena explicação esclarecendo aspectos do uso. Pode ser *printscreen* das telas ou layout elaborado por alguma ferramenta.

4.8 Relatórios e documentos

Descrever e/ou apresentar imagem dos relatórios ou documentos gerados pelo software.

5. Implantação

Indicar o repositório onde o código-fonte pode ser acessado. Fornecer informações sobre a instalação do software desenvolvido, assim como dos softwares complementares a serem instalados para o funcionamento do sistema.

Aqui também podem ser especificadas informações adicionais sobre o software, informações sobre sua utilização, backups, monitoramento, etc.

# 6. Conclusão

# Este item é muito importante. Faz o fechamento, concluindo as ideias. Esta etapa sintetiza todo o trabalho realizado e fornece uma resposta para a questão apresentada. Pode também levantar hipóteses e refletir sobre cada objetivo proposto.

A conclusão deverá apresentar um resumo de tudo o que foi feito. Poderão ser inseridos argumentos que mostrem quais objetivos foram atingidos e os resultados obtidos.

Referências

< Este é um item obrigatório. Lista numerada em ordem alfabética **>**

**IMPORTANTE UTILIZAR A FERRAMENTA MORE (Mecanismo Online para Referências) da UFSC baseada nas normas ABNT –** [**www.more.ufsc.br**](http://www.more.ufsc.br)

Obs: Listar somente as referências que tem autoria e que foram efetivamente citados no texto. As referências sem autoria, representadas apenas por uma URL (Ex. http://pmkb.com.br/sig/padroes-frameworks/pmbok-pmi/) devem ser apresentadas ao longo do texto, em notas de rodapé, de acordo com o exemplo a seguir:

**¹ Conforme disponível em: < http://pmkb.com.br/sig/padroes-frameworks/pmbok-pmi/>. Acesso em: 10 jul. 2020.**

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed. rev. São Paulo: Pearson, 2011. 529 p.

Glossário

É um item opcional. Trata-se de uma listagem que contém as palavras ou termos técnicos desconhecidos utilizados no texto, com seus significados. A lista deve ser em ordem alfabética.

**Exemplo:**

**SGBD** – Sistema Gerneciador de Banco de Dados. Software que gerencia e proporciona o armazenamento de dados, permitindo consultas aos dados armazenados e garantindo sua integridade.

**Sistemas de Informação Gerencial** ou **ERP** – **E**nterprise **R**esource **P**lanning ou software de planejamento de recursos empresariais. É um software que procura integrar todas as áreas da empresa, desde o chão de fábrica até a alta administração, procurando otimizar processos e garantir confiabilidade das informações.

**Workflow** – Software que procura gerenciar e descrever o fluxo de dados entre as tarefas e processos da organização.

Apêndice

É opcional – São documentos de agregados à obra para fins de apoio à argumentação. São documentos elaborados pelo autor. Nesta parte são incluídos os questionários, entrevistas, tabulação de dados, etc.

Anexos

É opcional. Documentos agregados à obra para fins de comprovação de dados ou ilustração.

# Padrões de formatação s serem utilizados:

# 1. Títulos use letra Arial ou Times New Roman, 14, negrito

* 1. **Subtítulos, Arial ou** Times New Roman**, tamanho 12, negrito**

**Corpo do texto:** Todo o corpo do texto deverá estar formatado com letra Arial ou Times New Roman tamanho 12. Espaçamento entre linhas 1,5.

**Itálico:** Deve ser usado nas palavras de outros idiomas. Esta orientação não se aplica às expressões latinas apud e et al.

**Formatação da página:** Margens: Direita e inferior: 2cm / Esquerda e superior: 3cm

Espaçamento entre linhas 1,5

**Referências para elaboração deste documento**

**IFSC,2018 -** Dicas para escrita de texto cientifico. Disponível em **:** [https://wiki.sj.ifsc.edu.br/wiki/index.php/Dicas\_para\_escrita\_de\_texto\_cient%C3%ADfico](about:blank) Acesso em: 25/04/2018

**Medeiros, Ernani Sales de.** Desenvolvendo Software com UML. Makron Books – São Paulo, 2004

**Normas ABNT.** Disponível em [https://www.normaseregras.com/normas-abnt/](about:blank)  Acesso em: 17/04/2018

**Sommerville, Ian*.*** Engenharia de Software. Ed. Addison Wesley - São Paulo, 2003

1. Imagem retirada de: https://www.businessinsider.com/what-amazon-looked-like-when-it-launched-2015-3?IR=T#heres-a-look-at-the-whole-homepage-from-august-16-1995-2 [↑](#footnote-ref-1)
2. Disponível em <[https://opencv.org](about:blank)> [↑](#footnote-ref-2)