



**Universidade de Brasília - UnB**  
**Faculdade UnB Gama - FGA**  
**Projeto Integrador de Engenharia II**

## **Seletor de Laranjas**

**Autores:** Beatriz Calheira, Camilla Alves, Giovanna Amorim,  
Guilherme Siqueira, Gustavo Carvalho, João Pedro Gomes,  
Letícia Marinho, Lucas Afonso, Maria Luiza Ferreira, Matheus  
Henrique, Natália Rodrigues, Thiago Ferreira

**Orientadores:** Alex Reis, José Felício, Rhander Viana, Ricardo  
Chaim, Paolo Gessini

**Brasília, DF**

**2020**



Beatriz Calheira, Camilla Alves, Giovanna Amorim, Guilherme Siqueira,  
Gustavo Carvalho, João Pedro Gomes, Letícia Marinho, Lucas Afonso, Maria  
Luiza Ferreira, Matheus Henrique, Natália Rodrigues, Thiago Ferreira

## **Seletor de Laranjas**

Trabalho submetido ao Ponto de Controle 3,  
da disciplina Projeto Integrador de Engenharia II da Universidade de Brasília.

Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA

Orientador: Alex Reis, José Felício, Rhander Viana, Ricardo Chaim,  
Paolo Gessini

Brasília, DF  
2020

# **Lista de ilustrações**

Figura 1 – Morfologia da laranja . . . . .	14
Figura 2 – Especificações da classificação da laranja por diâmetro conforme as normas do CEAGESP (CEAGESP, 2011) . . . . .	14
Figura 3 – Diagrama de atividades representando o fluxo de funcionamento da máquina. . . . .	15
Figura 4 – Estrutura Analítica do Projeto. . . . .	26
Figura 5 – Módulo de esteira de roletes. . . . .	41
Figura 6 – Componentes de uma esteira de roletes. . . . .	42
Figura 7 – Motorredutor DC. . . . .	43
Figura 8 – Sistema de transmissão de corrente e roda dentada. . . . .	46
Figura 9 – Perfil “U” simples. . . . .	47
Figura 10 – Dimensões Perfil “U” simples . . . . .	48
Figura 11 – Perfil retangular . . . . .	48
Figura 12 – Dimensões perfil retangular . . . . .	48
Figura 13 – Rólete selecionado. . . . .	49
Figura 14 – Bomba de água submersível . . . . .	50
Figura 15 – Ventoinha . . . . .	51
Figura 16 – Motor . . . . .	51
Figura 17 – Esquema de um sistema fotovoltaico isolado. . . . .	54
Figura 18 – Resumo dos valores obtidos em simulação. . . . .	58
Figura 19 – Módulo Fotovoltaico 355W Canadian Policristalino . . . . .	59
Figura 20 – Produção normalizada . . . . .	60
Figura 21 – Rendimento global do sistema . . . . .	61
Figura 22 – Especificações do controlador de acordo com a simulação. . . . .	61
Figura 23 – Controlador de carga Epever . . . . .	62
Figura 24 – Especificações do controlador de acordo com a simulação. . . . .	62
Figura 25 – Capacidade de condução de corrente, em ampères. . . . .	64
Figura 26 – Fusível GPV 15A. . . . .	65
Figura 27 – Diagrama unifilar do sistema de operação. . . . .	66
Figura 28 – Suporte para módulos fotovoltaicos. . . . .	66
Figura 29 – Diagrama de Blocos da Solução Eletrônica. . . . .	68
Figura 30 – Sistema Eletrônico. . . . .	69
Figura 31 – Diagrama Circuito Eletrônico. . . . .	70
Figura 32 – Esquemático de Montagem do Módulo de Aquisição de Imagem. . . . .	72
Figura 33 – Diagrama de Controle do Módulo de Aquisição de Imagem. . . . .	73
Figura 34 – Módulo de Câmera. . . . .	74

Figura 35 – Adaptador de Múltiplas Câmeras. . . . .	74
Figura 36 – Módulo Sensor de Obstáculo Infravermelho. . . . .	75
Figura 37 – Esquemático de montagem do módulo de definição de rotas. . . . .	76
Figura 38 – Diagrama de Controle do Módulo de Definição de Rotas . . . . .	77
Figura 39 – Esquemático de montagem do módulo de contagem. . . . .	80
Figura 40 – Diagrama de Controle do Módulo de Contagem . . . . .	81
Figura 41 – Sensor ultrassônico HC-SR04. . . . .	83
Figura 42 – Visão Frontal Display. . . . .	84
Figura 43 – HD Externo. . . . .	85
Figura 44 – Diagrama de Processos Seletor de Laranjas . . . . .	87
Figura 45 – Diagrama de Atuação: Fruto Podre . . . . .	88
Figura 46 – Diagrama de Atuação: Fruto Bom Sem Mancha . . . . .	89
Figura 47 – Diagrama de Atuação: Fruto Bom Com Mancha . . . . .	90
Figura 48 – Algoritmo de Aquisição de Imagem: Simulação 01. . . . .	91
Figura 49 – Algoritmo de Aquisição de Imagem: Simulação 02. . . . .	92
Figura 50 – Algoritmo de Definição de Rotas: Simulação 01 - Laranja Podre. . . . .	93
Figura 51 – Algoritmo de Definição de Rotas: Simulação 02 - Laranja Boa e Sem Mancha. . . . .	93
Figura 52 – Algoritmo de Definição de Rotas: Simulação 03 - Laranja Boa e Com Mancha. . . . .	94
Figura 53 – Algoritmo de Contagem das laranjas boas sem manchas: Simulação 01. .	95
Figura 54 – Algoritmo de Contagem das laranjas boas sem manchas: Simulação 02. .	96
Figura 55 – Algoritmo de Contagem das laranjas boas sem manchas: Simulação 03. .	96
Figura 56 – Tela de cadastro. . . . .	97
Figura 57 – Tela de login. . . . .	98
Figura 58 – Tela de informações quantitativas. . . . .	98
Figura 59 – Tela de informações em forma gráfica (total de laranjas processadas). .	99
Figura 60 – Tela de informações em forma gráfica (total de laranjas boas SEM imperfeições). . . . .	99
Figura 61 – Tela de informações em forma gráfica (total de laranjas boas COM imperfeições). . . . .	100
Figura 62 – Tela de geração de relatório. . . . .	100
Figura 63 – Tela de dados quantitativos da aplicação web. . . . .	101
Figura 64 – Tela do gráfico de total de laranjas por mês da aplicação web. . . . .	102
Figura 65 – Tela do gráfico de total de laranjas por ano da aplicação web. . . . .	102
Figura 66 – Tela do gráfico de total de laranjas boas SEM manchas por dia da aplicação web. . . . .	103
Figura 67 – Tela do gráfico de total de laranjas boas COM manchas por dia da aplicação web. . . . .	103

Figura 68 – Tela de geração de relatório da aplicação web.	104
Figura 69 – Painel da Aplicação Desktop.	104
Figura 70 – Tela de dados quantitativos da aplicação desktop.	105
Figura 71 – Diagrama geral da Arquitetura de Software.	106
Figura 72 – Relatório Composição da Rede Neural	112
Figura 73 – Treinamento Épocas	113
Figura 74 – Gráfico da evolução da acurácia e perda com o passar das épocas	114
Figura 75 – A API retornando todas as classificações possíveis	116
Figura 76 – Imagens classificadas pela API	117
Figura 77 – Representação do banco de dados NoSQL da aplicação Desktop.	119
Figura 78 – Diagrama da representação arquitetural da aplicação <i>desktop</i> .	121
Figura 79 – Representação da utilização do Firebase.	122
Figura 80 – Representação dos dados no Firestore.	122
Figura 81 – Representação da arquitetura da aplicação web.	124
Figura 82 – Algoritmo do servidor do socket.	127
Figura 83 – Algoritmo do cliente do socket.	128
Figura 84 – Conexão entre o algoritmo de contagem e banco de dados local MongoDB.	129
Figura 85 – Diagrama de Integração.	130
Figura 86 – Manual de Operação do Seletor de Laranjas	139

# **Lista de tabelas**

Tabela 1 – Equipe do projeto . . . . .	19
Tabela 2 – Professores da matéria . . . . .	20
Tabela 3 – Riscos Gerais . . . . .	20
Tabela 4 – Riscos de Estrutura / Energia . . . . .	21
Tabela 5 – Riscos de Eletrônica . . . . .	23
Tabela 6 – Riscos de Software . . . . .	24
Tabela 7 – Ponto de controle 1 - Engenharia de Software . . . . .	27
Tabela 8 – Ponto de controle 2 - Engenharia de Software . . . . .	27
Tabela 9 – Ponto de controle 3 - Engenharia de Software . . . . .	28
Tabela 10 – Ponto de controle 1 - Engenharia Eletrônica . . . . .	29
Tabela 11 – Ponto de controle 2 - Engenharia Eletrônica . . . . .	29
Tabela 12 – Ponto de controle 3 - Engenharia Eletrônica . . . . .	30
Tabela 13 – Ponto de controle 1 - Estrutura/Energia . . . . .	31
Tabela 14 – Ponto de controle 2 - Estrutura/Energia . . . . .	31
Tabela 15 – Incidência solar média mensal para a cidade de Avaré. . . . .	56
Tabela 16 – Componentes da carga . . . . .	57
Tabela 17 – Especificações Módulo CS3U 355P. . . . .	59
Tabela 18 – Especificações Controlador de Carga. . . . .	62
Tabela 19 – Especificações Bateria. . . . .	63
Tabela 20 – Especificações Raspberry Pi 4. . . . .	71
Tabela 21 – Especificações Módulo de Câmera. . . . .	74
Tabela 22 – Especificações Sensor Infravermelho. . . . .	75
Tabela 23 – Tabela Verdade de Classificações. . . . .	78
Tabela 24 – Especificações do Servo-motor MG996R. . . . .	78
Tabela 25 – Especificações do Sensor ultrassônico HC-SR04. . . . .	83
Tabela 26 – Especificações do Display. . . . .	84
Tabela 27 – Especificações do HD Externo. . . . .	86
Tabela 28 – Pilha de Tecnologias . . . . .	126
Tabela 29 – Custos Software. . . . .	131
Tabela 30 – Custos Eletrônica. . . . .	131
Tabela 31 – Custos Estrutura/Energia . . . . .	132
Tabela 32 – Perguntas e Respostas do Questionário . . . . .	140
Tabela 33 – Referências das Implementações . . . . .	149
Tabela 34 – Autoavaliação dos Integrantes da Equipe . . . . .	152

# **Lista de abreviaturas e siglas**

CEAGESP	- Companhia de Entrepastos e Armazéns Gerais de São Paulo
CD	- Deploy Contínuo
CI	- Integração Contínua
CNN	- Convolutional Neural Network
CSI	- Camera Serial Interface
DC	- Direct Current
DSI	- Display Serial Interface
EAP	- Estrutura Analítica de Projeto
GND	- Ground
GPIO	- General Purpose Input/Output
HD	- High Definition
HDMI	- High-Definition Multimedia Interface
IPS	- In-Plane Switching
IR	- Infravermelho
LCD	- Liquid Cristal Display
PIB	- Produto Interno Bruto
PWA	- Progressive Web App
PWM	- Pulse Width Modulation
RAM	- Random Access Memory
SD	- Secure Digital
SSR	- Server-Side Rendering
TAP	- Termo de Abertura de Projeto
USB	- Universal Serial Bus
VCC	- Tensão Corrente Contínua
VGA	- Video Graphics Array

# Sumário

<b>1</b>	<b>APRESENTAÇÃO . . . . .</b>	<b>13</b>
<b>1.1</b>	<b>Descrição do Projeto . . . . .</b>	<b>13</b>
<b>1.2</b>	<b>Lista É/Não é . . . . .</b>	<b>15</b>
1.2.1	É . . . . .	15
1.2.2	Não É . . . . .	16
<b>1.3</b>	<b>Premissas . . . . .</b>	<b>16</b>
<b>1.4</b>	<b>Restrições . . . . .</b>	<b>16</b>
1.4.1	Restrições do Produto . . . . .	17
1.4.2	Restrições do Projeto . . . . .	17
<b>1.5</b>	<b>Termo de Abertura do Projeto . . . . .</b>	<b>17</b>
1.5.1	Justificativa de Projeto . . . . .	17
1.5.2	Objetivos do Projeto . . . . .	18
1.5.3	Requisitos de Alto Nível . . . . .	18
<b>1.6</b>	<b>Lista das partes interessadas . . . . .</b>	<b>19</b>
1.6.1	Equipe do Projeto . . . . .	19
1.6.2	Professores . . . . .	19
<b>1.7</b>	<b>Riscos . . . . .</b>	<b>20</b>
1.7.1	Riscos Gerais . . . . .	20
1.7.2	Riscos de Estrutura / Energia . . . . .	20
1.7.3	Riscos de Eletrônica . . . . .	23
1.7.4	Riscos de Software . . . . .	24
<b>2</b>	<b>TEMPO . . . . .</b>	<b>26</b>
<b>2.1</b>	<b>Estrutura Analítica do Projeto . . . . .</b>	<b>26</b>
<b>2.2</b>	<b>Cronograma . . . . .</b>	<b>27</b>
2.2.1	Cronograma de Engenharia de Software . . . . .	27
2.2.2	Cronograma de Engenharia Eletrônica . . . . .	29
2.2.3	Cronograma de Estrutura/Energia . . . . .	31
<b>3</b>	<b>REQUISITOS . . . . .</b>	<b>32</b>
<b>3.1</b>	<b>Requisitos de Usuário . . . . .</b>	<b>32</b>
<b>3.2</b>	<b>Requisitos de Produto . . . . .</b>	<b>32</b>
3.2.1	Requisitos Estruturais . . . . .	32
3.2.2	Requisitos de Pré-Processamento . . . . .	33
3.2.3	Requisitos de Classificação . . . . .	33
3.2.4	Requisitos de Pós-classificação . . . . .	33

<b>3.3</b>	<b>Requisitos de Estrutura / Energia . . . . .</b>	<b>33</b>
3.3.1	Estrutura . . . . .	33
3.3.2	Sistema de Limpeza . . . . .	33
3.3.3	Sistema de Transmissão/Potência . . . . .	34
3.3.4	Sistema de Individualização . . . . .	34
3.3.5	Sistema de processamento de imagem . . . . .	34
3.3.6	Sistema de Classificação . . . . .	34
<b>3.4</b>	<b>Requisitos de Eletrônica . . . . .</b>	<b>35</b>
3.4.1	Câmeras . . . . .	35
3.4.2	Contagem de frutas . . . . .	35
3.4.3	Cancela Mecânica . . . . .	35
3.4.4	Microcontrolador . . . . .	35
<b>3.5</b>	<b>Requisitos de Software . . . . .</b>	<b>36</b>
3.5.1	Requisitos Rede Neural . . . . .	36
3.5.1.1	Requisitos Não Funcionais . . . . .	36
3.5.1.2	Requisitos Funcionais . . . . .	36
3.5.2	Requisitos Aplicação Web . . . . .	37
3.5.2.1	Requisitos Não Funcionais . . . . .	37
3.5.2.2	Requisitos Funcionais . . . . .	37
3.5.3	Requisitos Aplicação Desktop . . . . .	38
3.5.3.1	Requisitos Não Funcionais . . . . .	38
3.5.3.2	Requisitos Funcionais . . . . .	38
<b>4</b>	<b>ARQUITETURA DO PROJETO . . . . .</b>	<b>39</b>
<b>4.1</b>	<b>Arquitetura de Estrutura / Energia . . . . .</b>	<b>39</b>
4.1.1	Solução Geral . . . . .	39
4.1.2	Subsistema de Limpeza . . . . .	39
4.1.3	Subsistema de classificação . . . . .	39
4.1.4	Câmara de processamento de imagem . . . . .	40
4.1.5	Classificação por diâmetro . . . . .	40
<b>4.2</b>	<b>Solução Geral Estrutura . . . . .</b>	<b>41</b>
4.2.1	Sistemas transportadores . . . . .	41
4.2.1.1	Descrição da geometria . . . . .	41
4.2.1.1.1	Base da esteira . . . . .	42
4.2.1.1.2	Perfil lateral . . . . .	42
4.2.1.1.3	Motorreduutor . . . . .	43
4.2.1.1.4	Transmissão . . . . .	45
4.2.1.1.5	Roletes de fricção . . . . .	46
4.2.2	Dimensionamento . . . . .	46
4.2.2.1	Dimensionamento Módulo 1 . . . . .	46

4.2.2.1.1	Perfil lateral . . . . .	47
4.2.2.1.2	Base . . . . .	48
4.2.2.1.3	Rolete . . . . .	49
<b>4.3</b>	<b>Solução Geral Energia . . . . .</b>	<b>49</b>
4.3.1	Alimentação da Estrutura . . . . .	49
4.3.1.1	Alimentação do pré-processamento . . . . .	49
4.3.2	Alimentação dos componentes eletrônicos . . . . .	51
4.3.3	Servos motores . . . . .	52
4.3.4	Sistema Fotovoltaico Autônomo . . . . .	53
4.3.4.1	Componentes . . . . .	53
4.3.4.1.1	Módulo Fotovoltaico . . . . .	54
4.3.4.1.2	Controlador de carga . . . . .	54
4.3.4.1.3	Bateria . . . . .	55
4.3.4.1.4	Condutores . . . . .	55
4.3.4.1.5	Proteção . . . . .	55
4.3.4.2	Desenvolvimento . . . . .	55
4.3.4.2.1	Coleta de dados . . . . .	56
4.3.4.3	Componentes Escolhidos . . . . .	57
4.3.4.4	<i>Softwares</i> utilizados . . . . .	65
4.3.4.5	Diagrama Unifilar . . . . .	66
4.3.4.5.1	Estrutura de fixação . . . . .	66
<b>4.4</b>	<b>Arquitetura de Eletrônica . . . . .</b>	<b>67</b>
4.4.1	Solução Geral de Eletrônica . . . . .	67
4.4.2	Diagramas Eletrônicos . . . . .	69
4.4.2.1	Diagrama de Alimentação . . . . .	69
4.4.2.2	Diagrama Circuito Eletrônico . . . . .	70
4.4.3	Microcontrolador . . . . .	71
4.4.3.1	Raspbian . . . . .	71
4.4.4	Módulo de Aquisição de Imagem . . . . .	71
4.4.4.1	Definição . . . . .	71
4.4.4.2	Esquemático do Módulo de Aquisição de Imagem . . . . .	72
4.4.4.3	Diagrama de Controle do Módulo de Aquisição de Imagem . . . . .	72
4.4.4.4	Câmeras e Adaptador . . . . .	73
4.4.4.5	Sensor Infravermelho . . . . .	74
4.4.4.6	Integração do módulo com o segmento estrutural . . . . .	75
4.4.5	Módulo de Definição de Rotas . . . . .	76
4.4.5.1	Definição . . . . .	76
4.4.5.2	Esquemático do Módulo de Definição de Rotas . . . . .	76
4.4.5.3	Diagrama de Controle do Módulo de Definição de Rotas . . . . .	77

4.4.5.4	Servo-Motor . . . . .	78
4.4.5.5	Integração do módulo com o segmento estrutural . . . . .	79
4.4.6	<b>Módulo de Contagem . . . . .</b>	<b>79</b>
4.4.6.0.1	Definição . . . . .	79
4.4.6.1	Esquemático do Módulo de Contagem . . . . .	79
4.4.6.2	Diagrama de Controle do Módulo de Contagem . . . . .	79
4.4.6.3	Sensor HC-SR04 . . . . .	82
4.4.6.4	Integração do módulo com o segmento estrutural . . . . .	82
4.4.7	<b>Módulo de Disponibilização de Dados . . . . .</b>	<b>83</b>
4.4.7.1	Definição . . . . .	83
4.4.7.2	Display . . . . .	83
4.4.7.3	Integração do módulo com o segmento estrutural . . . . .	84
4.4.8	<b>Periférico Adicional . . . . .</b>	<b>85</b>
4.4.8.1	HD Externo . . . . .	85
4.4.9	<b>Comunicação Serial . . . . .</b>	<b>86</b>
4.4.10	<b>Diagramas . . . . .</b>	<b>86</b>
4.4.10.1	Diagrama de Processos . . . . .	86
4.4.10.2	Diagramas de Atuação . . . . .	88
4.4.11	<b>Algoritmos . . . . .</b>	<b>91</b>
4.4.11.1	Algoritmo de Aquisição de Imagem . . . . .	91
4.4.11.2	Algoritmo de Definição de Rotas . . . . .	93
4.4.11.3	Algoritmo de Contagem . . . . .	95
<b>4.5</b>	<b>Solução Geral de Software . . . . .</b>	<b>97</b>
4.5.1	<b>Arquitetura da Informação . . . . .</b>	<b>97</b>
4.5.1.1	Aplicação Web . . . . .	97
4.5.1.1.1	Protótipo . . . . .	97
4.5.1.1.2	Resultados . . . . .	100
4.5.1.2	Aplicação Desktop . . . . .	104
4.5.1.2.1	Protótipo . . . . .	104
4.5.1.2.2	Resultados . . . . .	104
4.5.2	<b>Arquitetura de Software . . . . .</b>	<b>106</b>
4.5.2.1	Integração de Todos os Sistemas de Software . . . . .	106
4.5.2.2	Machine Learning . . . . .	108
4.5.2.2.1	Código Utilizando o PyTorch . . . . .	109
4.5.2.2.2	Algoritmo utilizando o Keras . . . . .	110
4.5.2.2.3	Resultados e Conclusões . . . . .	114
4.5.2.2.4	Especificações Técnicas . . . . .	115
4.5.2.3	Banco de Dados Local . . . . .	118
4.5.2.3.1	Especificações Técnicas . . . . .	119

4.5.2.4	Desktop APP . . . . .	120
4.5.2.4.1	Especificações Técnicas . . . . .	120
4.5.2.5	<i>Back-end</i> como Serviço - Firebase . . . . .	121
4.5.2.5.1	Especificações Técnicas . . . . .	121
4.5.2.6	Aplicação Web . . . . .	123
4.5.2.6.1	Especificações Técnicas . . . . .	123
4.5.2.6.2	Configuração do Ambiente de Desenvolvimento . . . . .	124
4.5.2.6.3	Implantação . . . . .	125
<b>4.6</b>	<b>Inovação</b> . . . . .	<b>125</b>
<b>4.7</b>	<b>Pilha de Tecnologias Utilizadas</b> . . . . .	<b>126</b>
<b>4.8</b>	<b>Integração Eletrônica e Software</b> . . . . .	<b>126</b>
4.8.1	Integração do Algoritmo de Rotas com o Machine Learning . . . . .	127
4.8.2	Integração do Algoritmo de Contagem com o Desktop App e o Banco de Dados MongoDB . . . . .	129
<b>4.9</b>	<b>Diagrama de Integração</b> . . . . .	<b>130</b>
<b>5</b>	<b>CUSTOS</b> . . . . .	<b>131</b>
<b>5.1</b>	<b>Orçamento do Projeto</b> . . . . .	<b>131</b>
5.1.1	Custos Software . . . . .	131
5.1.2	Custos Eletrônica . . . . .	131
5.1.3	Custos Estrutura/Energia . . . . .	131
<b>6</b>	<b>CONSIDERAÇÕES FINAIS</b> . . . . .	<b>133</b>
<b>REFERÊNCIAS</b> . . . . .		<b>134</b>
<b>APÊNDICES</b> . . . . .		<b>138</b>
<b>APÊNDICE A – MANUAL DE OPERAÇÃO SELETOR DE LARANJAS</b> . . . . .		<b>139</b>
<b>APÊNDICE B – PERGUNTAS E RESPOSTAS DO QUESTIONÁRIO</b> . . . . .		<b>140</b>
<b>APÊNDICE C – DECISÕES IMPORTANTES E APRENDIZADOS (SOFTWARE)</b> . . . . .		<b>143</b>
<b>APÊNDICE D – REFERÊNCIAS DAS IMPLEMENTAÇÕES (SOFTWARE E ELETRÔNICA)</b> . . . . .		<b>149</b>
<b>APÊNDICE E – AUTOAVALIAÇÃO DOS INTEGRANTES</b> . . . . .		<b>152</b>

<b>APÊNDICE F – ESPECIFICAÇÕES TÉCNICAS - MÓDULO CANADIAN</b>	<b>160</b>
<b>APÊNDICE G – MANUAL DE MONTAGEM DO SUPORTE DOS MÓDULOS</b>	<b>162</b>
<b>APÊNDICE H – SIMULAÇÃO PVSYST</b>	<b>165</b>
<b>APÊNDICE I – DIAGRAMA UNIFILAR</b>	<b>172</b>
<b>APÊNDICE J – DESENHOS TÉCNICOS - ESTRUTURA</b>	<b>174</b>
<b>APÊNDICE K – ESPECIFICAÇÕES TÉCNICAS DO MANCAL</b>	<b>201</b>
<b>APÊNDICE L – ESPECIFICAÇÕES TÉCNICAS DO ROLAMENTO</b>	<b>206</b>

# 1 Apresentação

## 1.1 Descrição do Projeto

A máquina seletora de laranjas é capaz de selecionar frutas através de processamento de imagens e classificações mecânicas. Esta classificação baseia-se em padrões comerciais estabelecidos pelas Normas de Classificação do Citrus de Mesa([CEAGESP, 2011](#)). As seleções garantem a transparência na comercialização, com menores perdas de produção e melhor qualidade da laranja.

O seletor de laranjas é dividido em 3 etapas: pré-classificação, classificação e análise de dados.

A primeira etapa, pré-classificação, consiste na preparação das frutas. Nesse momento, ocorre a limpeza das mesmas, retirando as impurezas da superfície. Dessa forma, é possível minimizar a probabilidade de erros no processamento da imagem da fruta.

A segunda etapa é responsável pela seleção dos frutos, através das seguintes classificações:

- Classificação 1 - Frutas ruins: são frutas que apresentam qualquer tipo de lesão de origem mecânica, patológica ou entomológica que atinja o mesocarpo do fruto ([Figura 1](#)). Ou ainda, frutas que apresentam qualquer grau de decomposição, desintegração ou fermentação dos tecidos;
- Classificação 2 - Frutas boas: são frutas que não inviabilizam o consumo ou a comercialização.
  - Classificação 2.1 - Frutas que apresentam manchas difusas, ou seja, são aquelas que não encobrem a cor original da casca da laranja, permitindo a sua perfeita visualização e que não prejudicam a integridade da fruta.
  - Classificação 2.2 - Frutas sem manchas.
    - \* Classificação 2.2.1 - Frutas pequenas
    - \* Classificação 2.2.2 - Frutas médias
    - \* Classificação 2.2.3 - Frutas grandes

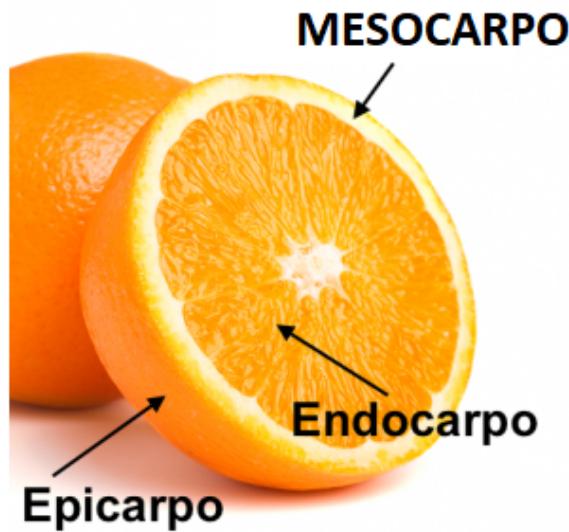


Figura 1 – Morfologia da laranja.

As classificações 2.2.1, 2.2.2 e 2.2.3, que levam em consideração o diâmetro da laranja, baseiam-se no padrão estabelecido pelas normas do CEAGESP ([CEAGESP, 2011](#)). Esses diametros variam conforme a espécie do citrus, e o mesmo pode ser observado na figura 2. Nesta figura é possível visualizar a relação entre o tipo de laranja e os três tamanhos com seus respectivos diâmetros.

Produto	Tamanho	Cotação CEAGESP	Nº dúzias na caixa M	Diâmetro em mm
Laranja de umbigo	Grande	A	5, 6 e 7	maior que 90
	Médio	B	8, 9 e 10	80 a 90
	Pequeno	C	11	menor que 80
Laranja comum e de baixa acidez	Grande	A	6 a 10	maior que 71
	Médio	B	11 a 13	65 a 71
	Pequeno	C	14 a 21	menor que 65

Figura 2 – Especificações da classificação da laranja por diâmetro conforme as normas do CEAGESP ([CEAGESP, 2011](#))

Por fim, a terceira etapa fornece ao usuário informações resultantes do processo de seleção. Essas informações são disponibilizadas por meio de um monitor na própria máquina e através de uma aplicação web.

Levando em consideração as categorizações descritas nos parágrafos acima, o diagrama de atividades representado na figura 3 exemplifica o fluxo de funcionamento do seletor de laranjas. O diagrama mostra a participação dos atores necessários para que o fluxo se complete, sendo eles o usuário e a máquina. O fluxo se inicia no canto superior esquerdo, a partir do círculo de cor sólida preta, e tem como fim os círculos brancos com

pontos pretos ao centro. Os losangos amarelos representam pontos de decisões que se ramificam em outras ações.

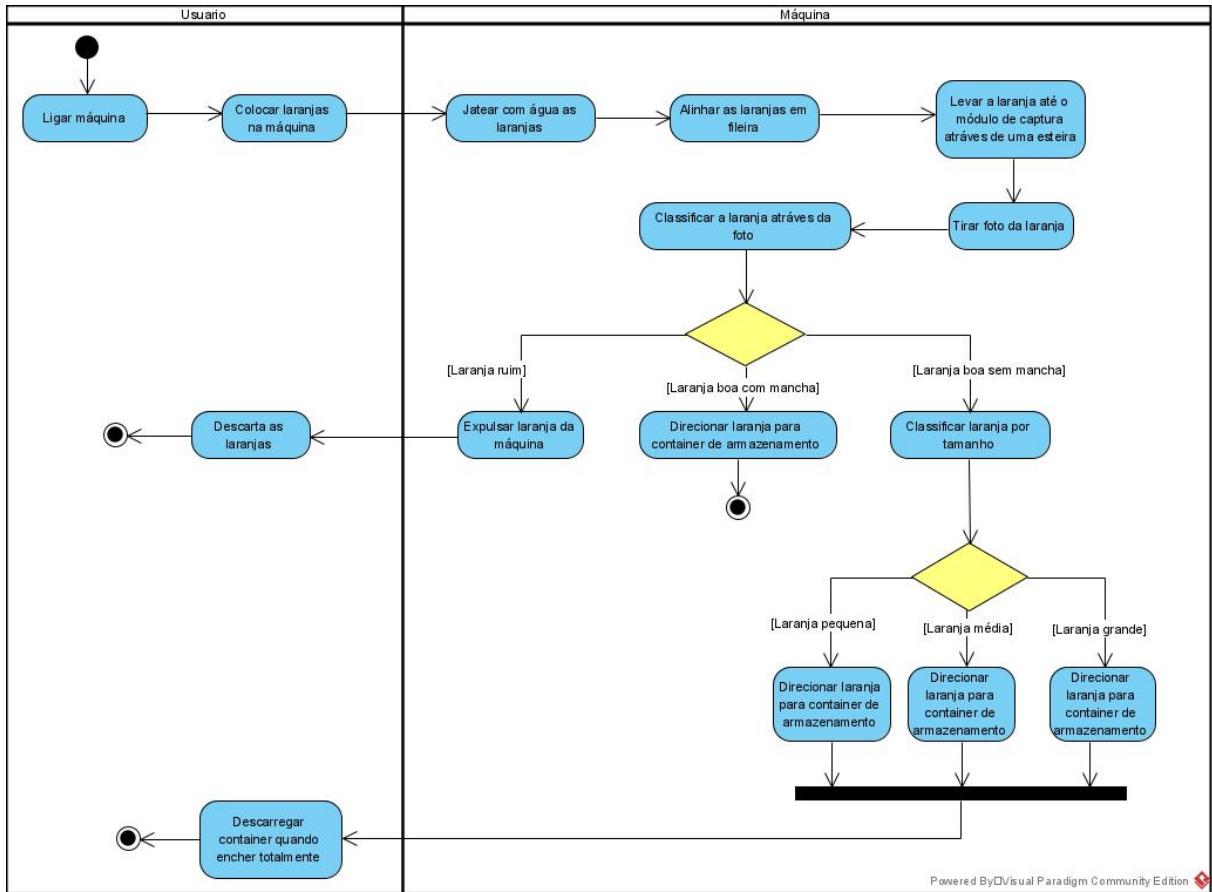


Figura 3 – Diagrama de atividades representando o fluxo de funcionamento da máquina.

## 1.2 Lista É/Não é

### 1.2.1 É

- É capaz de fazer a lavagem das laranjas antes do processamento, afim de eliminar sujeiras;
- É capaz de obter uma imagem nítida para classificação da laranja;
- É capaz de classificar as laranjas através da análise de imagens das mesmas;
- É capaz de separar as laranjas após o processamento por qualidade e tamanho;
- É capaz de exibir informações simples a respeito dos processamentos, offline, por meio de uma aplicação de plataforma embarcada na máquina;
- É capaz de exibir informações mais elaboradas a respeito dos processamentos, online, por meio de uma aplicação web;

- É capaz de gerar relatórios com as informações dos processamentos, online, por meio de uma aplicação web;

### 1.2.2 Não É

- Não é capaz de fazer a classificação de outras frutas ou produtos agrícolas;
- Não é capaz de fazer a classificação por cor das laranjas;
- Não é capaz de captar a casca do fruto em sua totalidade;
- Não é capaz receber interações de usuário na aplicação de plataforma embarcada na máquina;
- Não é capaz de funcionar sem energia elétrica;
- Não é um sistema inteiramente autônomo;

### 1.3 Premissas

Para que o projeto seja executado da melhor maneira possível, algumas premissas foram elaboradas pela equipe de projeto. Essas foram utilizadas no processo de planejamento.

- O local de instalação do maquinário deve possuir energia elétrica;
- O local de instalação do maquinário deve possuir água;
- O local de instalação do maquinário não pode apresentar grandes desníveis;
- O usuário deve possuir meios de armazenar de forma separada as frutas selecionadas;
- É necessário um local coberto para receber o maquinário;
- É necessário 1 (um) operador.

### 1.4 Restrições

Para que o projeto seja executado da melhor maneira possível, algumas restrições foram elaboradas pela equipe de projeto. Essas foram utilizadas no processo de planejamento.

### 1.4.1 Restrições do Produto

As restrições de produto associadas à solução estão descritas a seguir:

- Restrito à classificação de laranjas;
- Restrito à classificação de uma laranja por vez;
- Restrito à classificação de frutos por processamento de imagem e tamanhos;
- A classificação de imagem se restringe a quatro parâmetros: ruim, bom, com manchas e sem manchas;
- A classificação mecânica se restringe a três tamanhos distintos: pequeno, médio e grande.

### 1.4.2 Restrições do Projeto

As restrições do projeto associadas à solução estão descritas a seguir:

- Os integrantes devem trabalhar 12 horas semanais;
- Conhecimento da equipe em relação à determinadas tecnologias;
- O projeto deve seguir as instruções estabelecidas pelo plano de ensino da matéria;
- Tempo de execução limitado à duração do semestre.

## 1.5 Termo de Abertura do Projeto

### 1.5.1 Justificativa de Projeto

A busca constante por maior produtividade e qualidade dos produtos rurais exige que produtores utilizem cada vez mais equipamentos tecnológicos em suas atividades agrícolas, pecuárias e industriais. Aqueles que não utilizam a tecnologia a seu favor acabam perdendo espaço no mercado, uma vez que se tornam menos competitivos.

A automação de processos tem oferecido novas oportunidades de alavancar a produtividade, reduzindo custos e ampliando o uso eficiente dos recursos disponíveis. O processo de detecção, classificação e separação baseada em trabalho manual demanda um alto consumo de tempo e elevado custo de mão de obra, limitando a cadeia de produção de algumas culturas agrícolas ([ABBAS, 2019](#)).

Além disso, a busca por um elevado padrão de qualidade faz com que muitos critérios sejam avaliados. Nos alimentos, a aparência é o principal critério na escolha de

compra dos consumidores. Os produtos são avaliados considerando seus tamanhos, formas, cores e ausência de defeitos visuais.

A laranja é a principal fruta produzida no Brasil em volume, com 16,7 milhões de toneladas saídas dos pomares em 2008, correspondendo a 40,9% das colheitas totais da fruticultura. Dessa forma, grande parte da economia do país é movimentada pela citricultura. O Produto Interno Bruto (PIB) desse setor corresponde a US\$ 6,5 bilhões (2009), sendo US\$ 4,39 bilhões no mercado interno e US\$ 2,15 bilhões no mercado externo ([NEVES, 2008](#)).

No processo pós-colheita de laranjas, uma das maiores dificuldades consiste no método de seleção e padronização das frutas. Sistemas mecânicos baseados em peneiras com diferentes diâmetros são bastante utilizados pela indústria citrícola. Apesar disso, a movimentação intensa das frutas pode gerar grandes perdas, impactando no lucro da produção. Os custos estimados para os procedimentos de pós-colheita aproximam-se de 50% para os produtos perecíveis .

Por esses motivos, o presente projeto visa maximizar o processo de classificação através de técnicas que não comprometam a integridade física do produto, utilizando ferramentas mecânicas e de visão computacional. Isto é, ciência etecnologia que obtêm informações a partir de identificação de padrões e interpretação de imagens

### **1.5.2 Objetivos do Projeto**

O objetivo deste projeto é fornecer uma solução para classificação de laranjas, de forma não destrutiva, eficaz e computacionalmente otimizada. Além disso, visa implementar soluções encontradas na agricultura de precisão, caracterizada pelo alto uso de aparatos tecnológicos no campo, aplicadas de forma modular para pequenos e médios produtores.

Somado a isso, propõe-se a disponibilização de dados sobre os resultados da classificação com o intuito de proporcionar um elevado padrão de qualidade do produto final. Estas informações são relevantes no processo de estudo e produção da cultura.

### **1.5.3 Requisitos de Alto Nível**

Os requisitos de alto nível associados ao projeto estão descritos a seguir:

- A máquina deve ser capaz de realizar a classificação de laranjas em ruins e boas. Entre as boas, separar entre as que estão com manchas e sem manchas, através de imagens em tempo real;
- A máquina deve ser capaz de classificar as laranjas boas sem manchas em tamanhos pequeno, médio e grande;

- A máquina deve ser capaz de direcionar as laranjas em caminhos distintos referentes a sua classificação;
- A máquina deve ser capaz de gerar informações das laranjas que são classificadas.

## 1.6 Lista das partes interessadas

### 1.6.1 Equipe do Projeto

A equipe responsável pelo projeto está disposta na Tabela 1.

Tabela 1 – Equipe do projeto

NOME	MATRÍCULA	CURSO	PAPEL
Lucas Afonso Rodrigues Raimundo	12/0125960	Automotiva	Desenvolvedor
Camilla Alves de Oliveira	16/0003873	Eletrônica	Desenvolvedora
Beatriz Freitas Calheira	16/0003032	Eletrônica	Desenvolvedora
Giovanna Amorim de Farias	16/0007356	Eletrônica	Diretora de Eletrônica
Guilherme Siqueira Brandão	16/0007763	Software	Desenvolvedor
Gustavo Barbosa Carvalho	16/0007780	Software	Coordenador
Natália Maria Rodrigues Queiroz	16/0015839	Software	Desenvolvedora
Maria Luiza Ferreira Assumpção Almeida	16/0014433	Software	Diretora de Software
João pedro Gomes Cabral Ferreira	14/0145842	Software	Desenvolvedor
Thiago Ferreira	15/0149948	Software	Desenvolvedor
Matheus Henrique Almeida de Souza	16/0036569	Energia	Desenvolvedor
Leticia Marinho de Souza	16/0131898	Energia	Diretora de Qualidade

### 1.6.2 Professores

Os docentes responsáveis pela orientação do projeto estão dispostos na Tabela 2.

Tabela 2 – Professores da matéria

DOCENTE	ÁREA
Alex Reis	Engenharia de Energia
José Felício da Silva	Engenharia Eletrônica
Rhander Viana	Engenharia Automotiva
Ricardo Matos Chaim	Engenharia de Software
Paolo Gessini	Engenharia Aeroespacial

## 1.7 Riscos

### 1.7.1 Riscos Gerais

Tabela 3 – Riscos Gerais

CATEGORIA	SUBCATEGORIA	RISCO	MITIGAÇÃO
Organizacional	Recursos	Algum membro do grupo ter problemas em acompanhar a matéria e/ou abandono da matéria	Realocação de membros da equipe, possível redução de escopo e aumento da carga horária dos membros
Organizacional	Recursos	Agravamento da pandemia e mudança de cronograma da matéria	Replanejamento das atividades e realocação de membros da equipe
Organizacional	Recursos	Falta de conhecimento e experiência em relação às tecnologias	Incentivo dos membros para especialização prévia nas áreas a serem trabalhadas
Organizacional	Recursos	Defeito de algum equipamento da equipe	Alocarpareamentos com um membro que tenha disponibilidade

### 1.7.2 Riscos de Estrutura / Energia

Tabela 4 – Riscos de Estrutura / Energia

CATEGORIA	SUBCATEGORIA	RISCO	MITIGAÇÃO
Técnico	Desempenho	Limpeza insuficiente da Fruta	Inspeções periódicas para aferição da qualidade do processo de Limpeza, limpeza periódica dos mecanismos
Técnico	Interação Operador / Máquina	Falha estrutural e/ou mecanismos	Treinamento dos operadores da máquina sobre os limites de operação
Técnico	Interação Operador / Máquina	Riscos Ergonômicos (NR 17)	Treinamento dos Operadores, uso de equipamento de proteção individual (coletes ergonômicos)
Técnico	Interação Operador / Máquina	Riscos Acidentais (NR 12)	Treinamento dos Operadores, emprego de equipamento de proteção individual/coletiva, gerenciamento de um sistema de segurança operacional
Técnico	Integração	Falha na integração dos sistemas	Comunicação constante com os desenvolvedores dos sistemas do maquinário

Continua na próxima página

**Tabela 4 – continuação da página anterior**

CATEGORIA	SUBCATEGORIA	RISCO	MITIGAÇÃO
Técnico	Desempenho	Falha no processo de individualização das laranjas	Existência de um mecanismo que espalce as laranjas a fim de impedir que caia mais de uma fruta em um recipiente
Técnico	Interação Operador / Máquina	Falta de capacidade de armazenamento	Treinamento do operador do seletor sobre os limites de armazenamento e a forma adequada de retirá-los
Técnico	Desempenho	Emperramento do fruto na classificação por diâmetro	O mecanismo deve apresentar ângulo de queda suficiente para que a velocidade do fruto seguinte seja suficiente para desobstruir o caminho
Técnico	Integração	Precariedade na integração com eletrônica	Buscar estar sempre atrelado às alterações da equipe de eletrônica
Técnico	Instrumentação	Falta de Pontência para alimentar os sistemas	Utilização de softwares de simulação para uma escolha mais efetiva da bateria
Técnico	Instrumentação	Possíveis falhas mecânicas no motor e no sistema de transmissão	Utilização de softwares de simulação para uma escolha mais efetiva do motor

Continua na próxima página

**Tabela 4 – continuação da página anterior**

CATEGORIA	SUBCATEGORIA	RISCO	MITIGAÇÃO
Técnico	Instrumentação	Falta de eficiência do mecanismo primário de filtração	Estudo de melhores tecnologias quanto à eficiência dos filtros
Técnico	Conhecimento	Falta de referências e conhecimento técnicas	Estudo aprofundado de funcionamento de cada etapa designada

### 1.7.3 Riscos de Eletrônica

**Tabela 5 – Riscos de Eletrônica**

Categoria	Subcategoria	Risco	Mitigação
Técnico	Instrumentação	Insuficiência na capacidade de processamento causando comunicação falha entre o microcontrolador e os periféricos	Estudo das especificações de microcontroladores comerciais e escolher a opção que mais se encaixa.
Técnico	Instrumentação	Insuficiência na capacidade de processamento causando comunicação falha entre o microcontrolador e os software de processamento de imagens	Estudo das especificações de microcontroladores comerciais e escolher a opção que apresente a capacidade de memória necessária.

Continua na próxima página

**Tabela 5 – continuação da página anterior**

CATEGORIA	SUBCATEGORIA	RISCO	MITIGAÇÃO
Técnico	Instrumentação	Falha na definição da rota do fruto	Estudo das especificações do servomotor e escolher a opção que apresente a rotação necessária para definição e bloqueio de rotas.
Técnico	Instrumentação	Falha na identificação do fruto	Utilização de um sensor que apresente uma boa sensibilidade de detecção.
Técnico	Tecnologia	Atraso no tempo de resposta da câmera gerando ausência de imagens do fruto	Alinhamento com a equipe de estrutura do projeto para que haja sincronismo entre a velocidade da esteira e velocidade de captura da imagem .
Técnico	Tecnologia	Dificuldade em descrever as rotinas dos periféricos	Estudos para evolução dos conhecimentos.

#### 1.7.4 Riscos de Software

**Tabela 6 – Riscos de Software**

CATEGORIA	SUBCATEGORIA	RISCO	MITIGAÇÃO
Técnico	Integração	Falta de integração de software com eletrônica	Alinhamento com a parte de eletrônica do projeto.
Continua na próxima página			

**Tabela 6 – continuação da página anterior**

CATEGORIA	SUBCATEGORIA	RISCO	MITIGAÇÃO
Organizacional	Metodológica	Falhas na aplicação e aderência a metodologia de organização da equipe	Reforço constante, caso a metodologia esteja sendo ineficiente em conjunto com as entregas.
Técnico	Desempenho	Insuficiência do hardware	Caso ocorra, faremos o treinamento utilizando o google cloud, entretanto, os custos do projeto aumentarão.
Técnico	Instrumental	Inexperiência na tecnologia utilizada impactando no tamanho do escopo e da qualidade sobre o produto entregue	Treinamento e preparo dos membros antes do início do desenvolvimento
Organizacional	Requisitos	Atrazo dos prazos propostos por alguma das engenharias causando entrega do produto com requisitos incompletos	Priorização dos requisitos do projeto e acompanhamento do seu ciclo de desenvolvimento, estipulando prazos que condizem com a realidade do contexto do projeto.

# 2 Tempo

## 2.1 Estrutura Analítica do Projeto

A Estrutura Analítica do Projeto (EAP) foi construída seguindo as propostas de entrega para cada ponto de controle da disciplina.

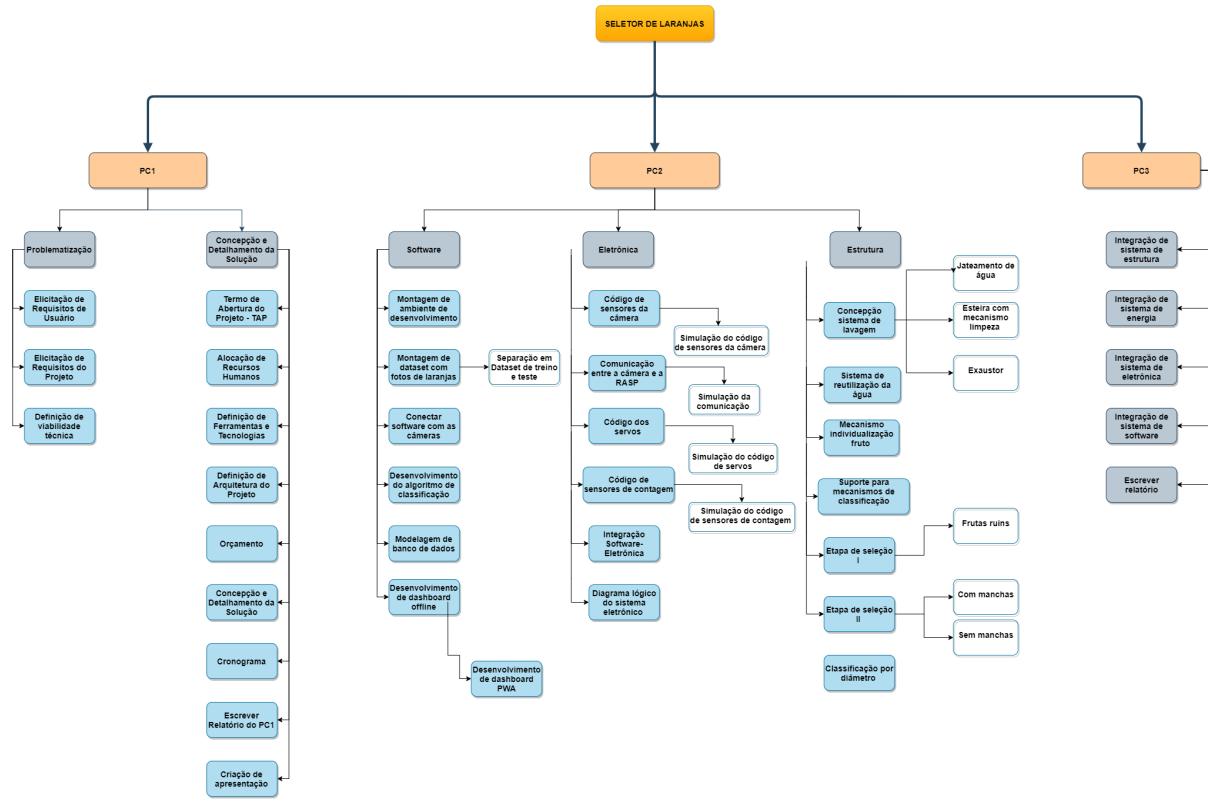


Figura 4 – Estrutura Analítica do Projeto.

## 2.2 Cronograma

### 2.2.1 Cronograma de Engenharia de Software

Atividade	Data de Início	Data Final
Levantamento Requisitos Rede Neural	02/09	10/09
Levantamento Requisitos Aplicativo Plataforma	02/09	10/09
Levantamento Requisitos Front-end aplicação web	02/09	10/09
Definição de Tecnologias	02/09	12/09
Definição de escopo rede neural, web e local	02/09	12/09
Modelagem de Banco de dados	12/09	20/09
Definição da Arquitetura geral	12/09	16/09
Estudo Dataset	12/09	20/09
Criação Protótipo	12/09	20/09
Detalhamento Ambiente de desenvolvimento e produção	12/09	20/09
Detalhamento Testes	12/09	20/09
Definição Riscos, Orçamento, Metodologia e demais	12/09	20/09

Tabela 7 – Ponto de controle 1 - Engenharia de Software

Atividade	Data de Início	Data Final
Configuração ambiente de desenvolvimento	02/10	09/10
Configuração ambiente de desenvolvimento	02/10	09/10
Estruturação API	02/10	09/10
Coleta e construção do Dataset	02/10	09/10
Implementação Inicial Rede Neural	09/10	16/10
Otimização e testagem da rede neural	10/10	24/10
Testes com câmeras para a rede neural	24/10	31/10
Otimização desempenho geral	01/11	15/11
Início do Desenvolvimento front-end local	09/10	24/10
Início do Desenvolvimento front-end web	09/10	24/10
Testagem aplicação web	01/11	15/11
Testagem aplicação local	01/11	15/11
Alimentação dados API com rede neural	17/10	29/10
Deploy API e aplicação web	01/11	15/11
Integração do software com o hardware	29/11	05/12

Tabela 8 – Ponto de controle 2 - Engenharia de Software

Atividade	Data de Início	Data Final
Continuação do desenvolvimento do <i>front-end web</i>	01/11	15/11
Continuação do desenvolvimento do <i>front-end deploy</i>	01/11	20/11
Fazer a API para submissão e classificação das imagens	01/11	20/11
Deploy API e aplicação <i>web</i>	16/11	20/11
Integração do software com eletrônica	18/11	22/11
Refinamento do relatório	16/11	22/11

Tabela 9 – Ponto de controle 3 - Engenharia de Software

### 2.2.2 Cronograma de Engenharia Eletrônica

Atividade	Data de Início	Data Final
Definição dos requisitos de engenharia eletrônica	21/08	05/09
Definição da estrutura geral de eletrônica	06/09	09/09
Desenho da arquitetura de eletrônica	17/09	19/09
Definição do microcontrolador	10/09	6/09
Definição dos servos	19/09	19/09
Definição das câmeras	15/09	15/09
Definir sensores para contagem das frutas	11/09	11/09
Definir sensores para identificar o fruto	11/09	11/09
Levantamento de custos	15/09	15/09
Validação de estrutura com software	16/11	16/11
Escrita do Relatório - Eletrônica	12/09	8/09
Integração Relatório	19/09	20/09
Elaboração da Apresentação	21/09	01/10

Tabela 10 – Ponto de controle 1 - Engenharia Eletrônica

Atividade	Data de Início	Data Final
Revisão dos pontos levantados no PC 1	03/10	05/10
Implementação do código de sensores da câmera	26/09	02/10
Simulação do código de sensores da câmera	03/10	04/10
Implementação do código de servos	03/10	10/10
Simulação do código de servos	11/10	02/10
Implementação do código de sensores de contagem	03/10	10/10
Simulação do código de sensores de contagem	11/10	12/10
Integração Software-Eletrônica	13/10	17/10
Escrita do Relatório - Eletrônica	05/10	16/10
Integração Relatório	17/10	28/10
Elaboração da Apresentação	19/10	29/10

Tabela 11 – Ponto de controle 2 - Engenharia Eletrônica

Atividade	Data de Início	Data Final
<b>Revisão dos pontos levantados no PC 2</b>	30/10	01/11
<b>Integração Geral do Sistema</b>	02/11	20/11
<b>Refinamento dos Códigos</b>	02/11	20/11
<b>Refinamento do Relatório</b>	02/11	20/11
<b>Revisão dos Custos</b>	02/11	20/11
<b>Validar se todos os requisitos foram atendidos</b>	02/11	10/11
<b>Escrita do Relatório - Eletrônica</b>	02/11	21/11
<b>Integração Relatório</b>	21/11	22/11
<b>Elaboração da Apresentação</b>	23/11	26/11

Tabela 12 – Ponto de controle 3 - Engenharia Eletrônica

### 2.2.3 Cronograma de Estrutura/Energia

Atividade	Data de Início	Data Final
Detalhamento do recipiente inicial	06/09	11/09
Sistema de lavagem	06/09	11/09
Esteira Estática	06/09	11/09
Mecanismo de separação das laranjas	06/09	11/09
Definição do sistema de alimentação dos componentes eletrônicos	15/09	18/09
Levantamento de possíveis motores	12/09	14/09
Definição da fonte de alimentação da bateria	15/09	16/09
Levantamento de custos	19/09	19/09
Escrita do relatório	16/09	21/09
Elaboração da apresentação	27/09	01/10

Tabela 13 – Ponto de controle 1 - Estrutura/Energia

Atividade	Data de Início	Data Final
Análise do feedback dado no PC1	02/10	04/10
Cálculos dos reservatórios e tubulações do sistema de lavagem	28/09	02/10
Simulação do sistema de alimentação	03/10	10/10
Elaboração de diagramas elétricos	11/10	15/10
Integração com demais áreas	16/10	18/10
Elaboração do relatório	19/10	25/10
Elaboração da apresentação	24/10	24/10
Especificação componentes e diagramas mecânicos	04/10	10/10
Desenhos mecânicos refinados	11/10	17/10
Elaboração plano de construção dos subsistemas	18/10	24/10
Apresentação	30/10	30/10

Tabela 14 – Ponto de controle 2 - Estrutura/Energia

# 3 Requisitos

Esta seção define os requisitos do sistema para que o projeto fosse executado de acordo com os requisitos de usuário, os quais foram definidos através de um questionário feito com alguns produtores de laranja. Após este levantamento de informações, foram definidos os requisitos de produto para que cada engenharia pudesse executar o projeto de acordo com suas competências e habilidades.

## 3.1 Requisitos de Usuário

Com o propósito de entender melhor as necessidades dos produtores de frutas e adequar ao problema a solução proposta neste projeto, um **questionário** foi elaborado e direcionado a alguns produtores de frutas. No total, 4 pessoas responderam ao questionário, entre elas duas cuidam da produção especificamente de laranjas, o que facilitou a compreensão da classificação da fruta, assim como dos processos pós-colheita. A partir disso, requisitos como a classificação por diâmetro foram inseridos no escopo do projeto, além da retirada de algumas soluções que foram consideradas desnecessárias.

As perguntas e respostas do questionário estão presentes no Apêndice A deste documento.

## 3.2 Requisitos de Produto

Com o objetivo de entender as exigências necessárias para a construção do projeto foram definidos alguns requisitos gerais, divididos em Requisitos Estruturais, de Pré-Processamento, de Classificação e de Pós-Classificação. Após isso foram levantados os requisitos competentes a cada engenharia.

### 3.2.1 Requisitos Estruturais

- A máquina deve possuir uma entrada para a inserção das frutas;
- A área de entrada das frutas não deve possuir altura excessiva;
- A máquina deve ser capaz de manter a fruta no mecanismo até a eliminação da mesma;
- A máquina deve ser capaz de suportar a carga;

### 3.2.2 Requisitos de Pré-Processamento

- A máquina deve ser capaz de organizar as frutas introduzidas sem interferência manual;
- A máquina deve ser capaz de retirar sujeiras, através de lavagem;
- A máquina deve ser capaz de manter um fluxo automatizado de processamento;

### 3.2.3 Requisitos de Classificação

- A máquina deve ser capaz de classificar a qualidade da fruta através de imagens;
- A máquina deve ser capaz de classificar o tamanho da fruta;
- A máquina deve ser capaz de segregar em caminhos diferentes todas as categorizações propostas;

### 3.2.4 Requisitos de Pós-classificação

- A máquina deve ser capaz de direcionar as laranjas em caminhos distintos referentes a sua classificação;

## 3.3 Requisitos de Estrutura / Energia

### 3.3.1 Estrutura

- Apto de suportar as cargas estruturais com um fator de segurança adequado. Comportar os sistemas eletrônicos, elétricos e mecânicos apropriadamente, possibilitando a integração dos sistemas;

### 3.3.2 Sistema de Limpeza

- Capaz retirar impurezas macroscópicas das superfícies das frutas, impurezas como, por exemplo: terra, folhas, galhos, etc. Proporcionando a aquisição/processamento das imagens de forma fiel às características das frutas;
- Otimizar o uso da água no ciclo de limpeza e reduzir o desperdício de água;
- Capaz de coletar as impurezas e armazenar as mesmas até o fim do funcionamento da máquina;
- Capaz de secar as frutas de forma a possibilitar uma captura da imagem fiel ao real;

- O sistema de filtração dos detritos e de reaproveitamento da água necessita de manutenção periódica;

### 3.3.3 Sistema de Transmissão/Potência

- Transmitir a força motriz de forma otimizada com redução de perdas;
- Conduzir as frutas desde o início do processo até o fim da classificação;

### 3.3.4 Sistema de Individualização

- A máquina deve ser capaz de organizar os frutos em recipientes individuais para serem classificados através da captação de imagens;
- Os frutos devem seguir uma velocidade adequada para que seja possível a captação de imagens;
- Os frutos não podem cair do recipiente;

### 3.3.5 Sistema de processamento de imagem

- A câmara deve comportar as câmeras para captar imagens da superfície da laranja;
- A câmara deve fornecer ambiente adequado para a obtenção de imagens com a melhor qualidade possível;
- A câmara deve possuir acesso fácil para manutenção dos componentes eletrônicos interiores;
- A câmara deve ser resistente ao ambiente de trabalho.

### 3.3.6 Sistema de Classificação

- A máquina deve ser capaz de gerir todo o fluxo, sem emperrar ou interromper o percurso;
- A máquina deve conter caminhos distintos referentes a cada classificação proposta no projeto;
- A máquina deve ser capaz de separar os frutos em tamanhos pequeno, médio e grande;
- A separação mecânica obedecerá um determinado intervalo de diâmetro para cada classificação;

- A classificação deve permitir a instalação de sensor para contagem dos frutos;
- É necessário que haja recipiente para armazenar as classificações;

## 3.4 Requisitos de Eletrônica

### 3.4.1 Câmeras

- As câmeras devem ser capazes de captar múltiplas imagens em um curto espaço de tempo;
- As câmeras devem ter alta resolução para que os parâmetros de seleção sejam identificados;
- As câmeras devem ser capazes de captar imagens das frutas em movimento;
- As câmeras devem estar posicionadas de forma que haja a captura máxima da superfície do fruto;
- As câmeras devem ser capazes de interfacear com o microcontrolador escolhido;

### 3.4.2 Contagem de frutas

- Os sensores devem ser capazes de detectar os frutos em movimento e em um curto espaço de tempo;
- Os sensores devem ser capazes de comunicar com o microcontrolador escolhido;

### 3.4.3 Cancela Mecânica

- A cancela deve se movimentar horizontalmente;
- A cancela deve ser capaz de determinar a rota do fruto na esteira;
- A cancela deve ser capaz de receber a informação do microcontrolador;
- A cancela deve ter velocidade suficiente para setar as rotas, sem prejudicar o fluxo da esteira.

### 3.4.4 Microcontrolador

- O microcontrolador deve ter uma alimentação compatível ao sistema;
- O microcontrolador deve ter uma comunicação estável com seus periféricos;

- O microcontrolador deve ser capaz de receber os dados dos sensores e controlar os atuadores;
- O microcontrolador deve ser capaz de enviar e receber os dados para o software;
- O microcontrolador deve ser capaz de realizar a contagem dos frutos selecionados.

## 3.5 Requisitos de Software

### 3.5.1 Requisitos Rede Neural

#### 3.5.1.1 Requisitos Não Funcionais

- Com relação a velocidade da classificação realizada pela rede neural, o algoritmo deve ser capaz de identificar a qualidade das laranjas, de forma mais ágil do que um ser humano treinado.
- A rede neural deverá ser fundamentada e desenvolvida com base em uma rede convolucional.
- O algoritmo deve ser otimizado para não exigir muito do hardware a ser utilizado
- A rede neural deve ser capaz de operar sem conexão a internet, realizando a classificação dos artefatos independente da qualidade ou existência de conexão.

#### 3.5.1.2 Requisitos Funcionais

- A rede neural deve ser capaz de realizar a classificação das laranjas entre as categorias pré-estabelecidas em sua programação.
- A rede neural deve ser capaz de receber imagens do *dataset* e através de aprendizado supervisionado, ser capaz de distinguir a qualidade das laranjas.
- A qualidade das imagens que alimentarão a rede neural, deverão possuir qualidade suficiente para serem convertidas em informação válida e consistente.
- A rede neural deve ser capaz de realizar a classificação, sem cometer falhas grotescas.
- A rede neural deve ser capaz de armazenar informações referentes aos resultados das classificações por ela realizadas.
- A rede neural deve ser capaz de receber imagens, em tempo real, das câmeras de captação.
- O algoritmo deve ser capaz de processar imagens referentes a diferentes câmeras, de forma simultânea e dar uma classificação, com base nestas.

- O algoritmo deve ser capaz de segmentar apenas o objeto de interesse na imagem.
- O algoritmo deve ser capaz de enviar o resultado da classificação para os outros sistemas componentes da máquina.
- Quando ocorrer uma falha ou interrupção no processamento, o algoritmo deve ser capaz de informar os demais componentes, ou paralisar o processo como um todo.
- O algoritmo deve ser capaz de realizar a classificação entre ruins, boas com manchas e boas sem manchas com base nas imagens recebidas.

### 3.5.2 Requisitos Aplicação Web

#### 3.5.2.1 Requisitos Não Funcionais

- O sistema deve ser seguro quanto às informações cadastradas e exibidas, não permitindo o acesso indevido a essas informações.
- O sistema deve ser de fácil manutenibilidade.
- O sistema deve ter um bom desempenho quanto à utilização de suas funcionalidades.
- A interface do sistema deve ser amigável e responsiva.

#### 3.5.2.2 Requisitos Funcionais

- O sistema deve permitir o cadastramento de um usuário a partir do código de identificação de cada máquina.
- O sistema deve permitir que os usuários cadastrados façam login na aplicação para terem acesso ao restante de suas funcionalidades.
- O sistema deve permitir que os usuários recuperem seu login, em caso de perda de senha.
- O sistema deve permitir que os usuários alterem sua senha.
- O sistema deve permitir que os usuários alterem os dados do seu perfil (nome e e-mail).
- O sistema deve permitir que os usuários visualizem os dados quantitativos da colheita (laranjas processadas pela máquina), mostrando as quantidades totais de laranjas processadas e de suas classificações e subclassificações.
- O sistema deve permitir que os usuários alterem o período de apresentação das informações quantitativas, selecionando o intervalo de tempo e o período específico do qual quer que as informações sejam exibidas.

- O sistema deve permitir que os usuários visualizem de forma gráfica as informações da colheita.
- O sistema deve permitir que os usuários selecionem o intervalo e o período específico de visualização das informações no gráfico.
- O sistema deve permitir que os usuários selecionem qual informação de classificação ou subclassificação das laranjas deve ser apresentada no gráfico.
- O sistema deve permitir que os usuários gerem um relatório com as informações quantitativas da colheita.

### 3.5.3 Requisitos Aplicação Desktop

#### 3.5.3.1 Requisitos Não Funcionais

- O sistema deve apresentar os dados de maneira agradável na interface do sistema.

#### 3.5.3.2 Requisitos Funcionais

- O sistema deve apresentar quantas frutas foram classificadas e em que tipos;
- O sistema deve omitir a interface do sistema operacional, mostrando em tela cheia a aplicação desktop;
- O sistema deve desligar a Raspberry quando a máquina for desligada;
- O sistema deve sincronizar o banco de dados local com o banco de dados na nuvem. Quando houver conexão com a internet, periodicamente;
- O sistema deve ser capaz de identificar quando a Raspberry possui conexão com a internet ativa;
- O sistema deve ser capaz de realizar a interpretação dos dados enviados pelos sensores contadores e armazenar a informação no banco de dados;
- O sistema deve ter uma arquitetura que permita a execução de múltiplas tarefas assíncronas;
- A máquina deve possuir um botão de reiniciar todo o mecanismo;
- O banco embarcado no sistema deve ser leve e rápido, não gerando muito esforço de processamento;

# 4 Arquitetura do Projeto

## 4.1 Arquitetura de Estrutura / Energia

### 4.1.1 Solução Geral

A estrutura física do maquinário conta com uma solução prática e eficiente para a problemática. É composta por quatro subsistemas, sendo eles: higienização, estrutura de classificação, separação dos frutos e classificação por diâmetro. O primeiro subsistema higienizará o fruto com água, tendo como ação auxiliar a esteira com mecanismo de escovação e a secagem realizada por um exaustor. A estrutura de classificação consiste no mecanismo de individualizar o fruto para o processamento bem como a estrutura em si que abrigará as câmeras, sensores e demais componentes. O processo de separação se dará por duas cancelas, uma primeira que separa frutos bons e ruins e uma segunda que separa os frutos bons em com manchas e sem manchas. Por fim, a classificação por diâmetro será feita por um mecanismo manual, onde será classificada em diâmetro pequeno, médio e grande.

### 4.1.2 Subsistema de Limpeza

O subsistema de limpeza será uma seção da máquina com o propósito de remover da superfície das frutas impurezas macroscópicas que podem afetar a aquisição de dados ou danificar algum tipo de mecanismo.

As esteiras se constituem de roletes, cilindros vazados que giram, os roletes são movimentados por motor elétrico que estão ligados por uma corrente metálica, as frutas devem ser depositadas em cima da esteira para o início do processo.

O processo de limpeza é feito com o jateamento de água e o atrito das frutas com roletes de escovas, cilindros vazados que em sua superfície possui cerdas. O atrito somado com o solvente deverá remover impurezas macroscópicas, como por exemplo, terra ou folhas. Após a limpeza a esteira tem a sua configuração normal, com roletes e as frutas são direcionadas para o processo de secagem que é feito por uma ventoinha.

### 4.1.3 Subsistema de classificação

Após o fruto sair do sistema de exaustão, inicia-se o processo de individualização dos frutos. Um dos requisitos para processamento é que seja exposto um fruto de cada vez. Por esse motivo, cada fruto estará em um recipiente para ser processado.

Em uma visão superior, observa-se o sistema de afunilamento. Este acontecerá com uma angulação da parede da esteira, ainda não determinada. A saída deste funil possibilita a saída apenas de um fruto por vez, caindo já no recipiente.

Os recipientes estarão bem alinhados e próximos uns aos outros. Para o processamento, não é um requisito que todos os receptáculos tenham um fruto. Este recipiente terá bordas para atender ao requisito do usuário de que o fruto não saia do fluxo da máquina, impossibilitando assim, o seu processamento e classificação.

#### 4.1.4 Câmara de processamento de imagem

A câmara de processamento de imagem é o compartimento fechado que irá receber as laranjas (1 por vez) e onde será realizado o processamento de imagem. Para tanto, a estrutura deve comportar as câmeras de modo a obter imagens cobrindo a maior parte possível da superfície da laranja. A estrutura deve ser revestida internamente por um material anti-reflexivo (por meio de filmes com índices de refração distintos ou o uso de outro material que possua efeito similar) para não gerar prejuízos à imagem captada por meio da iluminação da câmara. A estrutura deve comportar o sistema de iluminação auxiliar à câmara, de modo a maximizar a qualidades das imagens captadas. Por fim, a estrutura da câmara deve ser onstruída de um material resistente à oxidação devido ao ambiente de trabalho.

Logo, a estrutura da câmara será retangular com uma entrada e uma saída. A dimensão exata será determinada pelo tempo de processamento necessário para identificação e classificação da laranja, e a velocidade ao adentrar a câmara. A estimativa inicial é de 0,45 metro de comprimento e 0,30 metro de altura e largura.

As câmeras estarão ancoradas no interior da câmara, assim como as fitas de LED que serão utilizadas para a iluminação.

A câmara será de um material resistente à corrosão e oxidação e revestida internamente por um filme ou material polímero anti-reflexivo.

#### 4.1.5 Classificação por diâmetro

O mecanismo de classificação por diâmetro se dará por esteiras que possuem um certo ângulo de abertura. O intervalo das classificações ainda não foi definido. Haverá um desnível do início do mecanismo até o fim, a fim de auxiliar na velocidade do fruto e também evitar emperramentos.

Abaixo da esteira de classificação por diâmetro haverá um recipiente que receberá as laranjas do intervalo de diâmetro a ser proposto de acordo com as regras comerciais, respeitando a proposta do projeto de tamanhos pequeno, médio e grande([CARTILHAS...](#),

2016). As caixas que receberão as classificações serão de plástico já amplamente utilizadas no manejo de frutas e outros insumos.

## 4.2 Solução Geral Estrutura

### 4.2.1 Sistemas transportadores

Conforme o trabalho de Lamb (LAMB, 2015) podemos definir a função de sistemas transportadores como a condução de objetos de um lugar para outro, dentro de um processo industrial de modo a integrar da melhor maneira possível a planta fabril. Vale destacar , conforme Rudenko (RUDENKO, 1976), que o “fluxo de matéria-prima das empresas depende essencialmente da escolha correta e racional do tipo de máquina de transporte, seus principais parâmetros e eficiência de trabalho.”

O acionamento de tais máquinas pode ocorrer naturalmente por meio da ação da gravidade, ou por meio de motores elétricos e dispositivos hidráulicos e pneumáticos.

#### 4.2.1.1 Descrição da geometria

Devido aos requisitos apresentados para a construção do equipamento, o sistema transportador escolhido é a esteira de roletes. As esteiras de roletes são divididas em módulos de acordo com a função que cada parte da esteira deve exercer. Dessa forma, são definidas dimensões específicas de cada módulo para sua aplicação. A Figura “5” abaixo exibe um módulo de esteira de roletes.

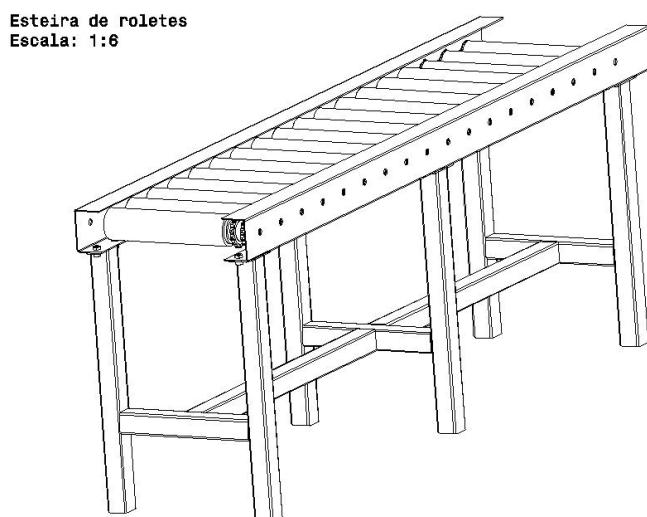


Figura 5 – Módulo de esteira de roletes.

Os componentes de uma esteira de roletes são listados abaixo e exibidos na Figura “6”.

1. Base da esteira;
2. Perfil lateral;
3. Motorredutor;
4. Transmissão;
5. Rolete.

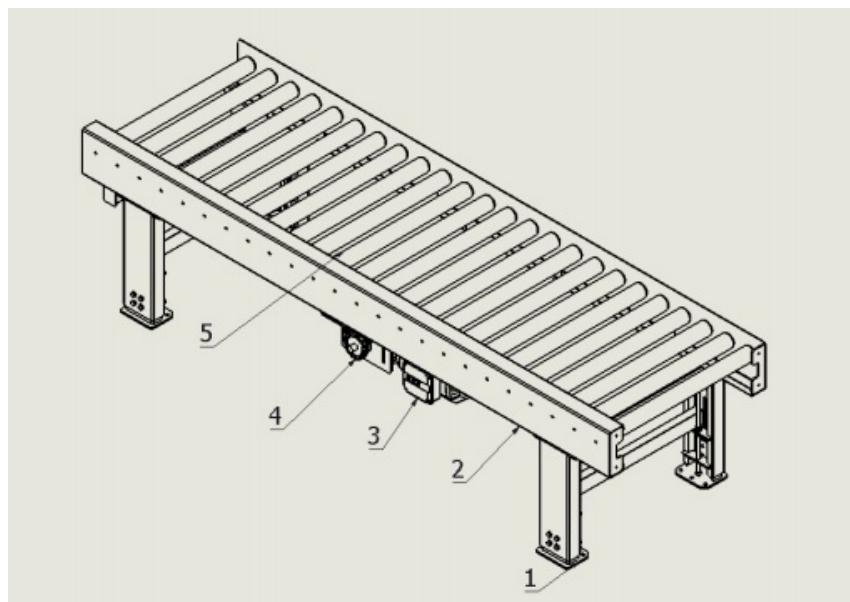


Figura 6 – Componentes de uma esteira de roletes.

#### 4.2.1.1.1 Base da esteira

A base da esteira é o componente responsável por toda a sustentação dos demais componentes, visto que nela são ancorados os perfis laterais. O nivelamento da esteira é determinante para o seu desempenho, afetando diretamente a velocidade do produto transportado. Desse modo, um conceito muito observado nas esteiras disponíveis no mercado é o de ajuste de altura por meio de barra rosada e parafusos.

#### 4.2.1.1.2 Perfil lateral

O perfil lateral é responsável por sustentar os roletes, sendo então submetido à forças cisalhantes resultado da massa do rolete e do produto transportado somadas.

#### 4.2.1.1.3 Motorredutor

O motorredutor é conjunto formado por um motor elétrico acoplado à um sistema de redução utilizando engrenagens de modo a fornecer o torque e potência adequados para cada aplicação específica, conforme Mott (MOTT, 2015).

Na Figura “7” é exibido um modelo de motorredutor da fabricante alemã Technische Antriebselemente GmbH.

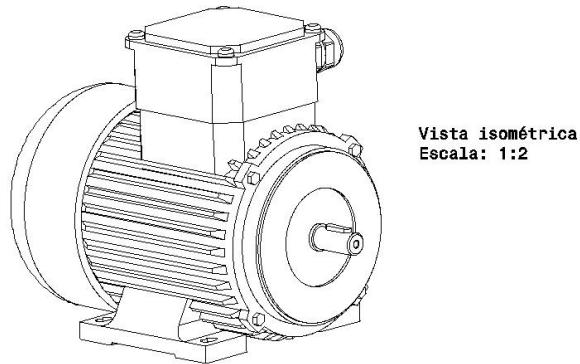


Figura 7 – Motorredutor DC.

Com os dados de potência e rotação, é possível dimensionar corretamente o motorredutor de acordo com a aplicação desejada (EURODRIVE, 2008). O equacionamento necessário para o dimensionamento será apresentado abaixo.

A potência ( $P$ ) exigida para transportar uma determinada massa é definida pela soma da Potência estática ( $P_s$ ) e da Potência Dinâmica ( $P_d$ ).

$$P = P_s + P_d \quad (4.1)$$

A Potência Estática é a potência em velocidade constante, levando em consideração as forças gravitacionais e de atrito. Essa componente da Potência é determinada pela equação 4.2.

$$P_s = \frac{F_f \cdot v}{\eta_g \cdot \eta_2} \quad (4.2)$$

Onde:

$F_f$  = Força de resistência à translação [N];

$v$  = Velocidade máxima da esteira [m/s];

$\eta_g$  = Rendimento do redutor;

$\eta_2$  = Rendimento total do sistema de transmissão.

A Força de resistência à translação é determinada pela equação 4.3.

$$F_f = m \cdot g \cdot \left[ \frac{2}{D} (\mu_L \frac{d}{2} f) + c \right] \quad (4.3)$$

Onde:

$m$  = Massa [kg];

$g$  = Aceleração da gravidade [m/s<sup>2</sup>];

$D$  = Diâmetro da roda livre [mm];

$\mu_L$  = Coeficiente de atrito do mancal;

$d$  = Diâmetro do mancal [mm];

$f$  = Braço de alavanca do atrito rolante [mm];

$c$  = Coeficiente de atrito lateral.

A Potência Dinâmica é definida como a potência necessária para aceleração e desaceleração do conjunto movimentado (EURODRIVE, 2008).

$$P_d = \frac{M_d \cdot n_m}{9550} \quad (4.4)$$

Onde:

$M_d$  = Torque de aceleração dinâmico [Nm];

$n_m$  = Rotação do motor [rpm].

O momento de inercia de um rolete pode ser definido conforme a equação 4.5 (EURODRIVE, 2008).

$$J = \frac{1}{2} \cdot m \cdot (r_2^2 + r_1^2) \quad (4.5)$$

Onde:

$J$  = Momento de inércia do rolete [kgm<sup>2</sup>];

$r_2$  = Raio externo do rolete;

$r_1$  = Raio interno do roelte.

“Para se ter um ponto de referência comum de momento de inércia de massa do motor e momento de inércia de massa externa, o momento de inércia de massa externa deve ser reduzido pela rotação do redutor”. ([EURODRIVE, 2008](#))

$$J_x = J \cdot \left(\frac{n_a}{n_m}\right)^2 \quad (4.6)$$

Onde:

$n_a$  = Rotação da saída do conjunto redutor [rpm].

A rotação de saída do conjunto redutor pode ser calculado por meio da equação [4.7](#).

$$n_a = \frac{v \cdot 1000 \cdot 60}{\pi \cdot D} \quad (4.7)$$

Logo, a relação de transmissão pode ser calculada por meio da equação [4.8](#).

$$I = \frac{n_m}{n_a} \quad (4.8)$$

#### 4.2.1.1.4 Transmissão

Seguindo as definições utilizadas pelos fabricantes de roletes, estes podem ser acionados ou livres. Os roletes acionados possuem seu acionamento, geralmente, por meio de motores elétricos e o conjunto de transmissão é formado por correntes ou correias.

Segundo ([BUDYNAS; NISBETT et al., 2008](#)) o uso de correias, cabos, correntes e outros elementos de máquinas flexíveis são utilizados para a transmissão de potência em sistemas transportadores, substituindo os elementos rígidos, tais como engrenagens e eixos. A utilização de elementos flexíveis simplifica os processos de fabricação e reduz custos.

Na transmissão de potência por correntes podemos destacar as seguintes características: razão constante, longa vida útil e capacidade de acionar mais de um eixo simultaneamente. O uso de correntes de rolos é padronizado pela ANSI, de acordo com suas dimensões.

Para transmissão de potência utilizando correntes, faz-se necessário o uso de uma roda dentada. A Figura “[8](#)” apresenta o sistema de transmissão por corrente e roda dentada.

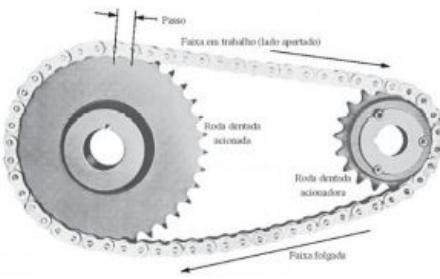


Figura 8 – Sistema de transmissão de corrente e roda dentada.

Para fins de comercialização, a ANSI também padroniza as rodas dentadas disponíveis no mercado.

#### 4.2.1.1.5 Roletes de fricção

Roletes de fricção são comumente utilizados em sistemas transportadores nos quais há acumulo de materiais. Caso o material transportado seja impedido de se movimentar e os roletes continuam a girar, ocorre o aumento de pressão que pode resultar em danos ao material transportado. Roletes de fricção possuem as rodas dentadas ancoradas no rolete, de modo que a força aplicada sobre o rolete deve ser maior que a força de fricção exercida pelo acoplamento do rolete ao seu eixo para que haja movimento da carga.

### 4.2.2 Dimensionamento

De acordo com os requisitos apresentados pelo projeto, os sistemas transportadores foram divididos em dois módulos de acordo com as características que cada um deve possuir.

#### 4.2.2.1 Dimensionamento Módulo 1

A principais grandezas relacionadas à geometria do módulo são:

1. Altura = 600 mm;
2. Comprimento = 1550 mm;
3. Largura = 300 mm.

Para o dimensionamento do primeiro módulo, foram definidas as seguintes variáveis:

- Diâmetro externo do rolete = 50 mm;

- Diâmetro do mancal = 32 mm;
- Braço de alavanca do atrito rolante = 2mm;
- Coeficiente de atrito do mancal = 0.005;
- Rendimento do sistema de transmissão por correntes = 0.267;
- Rendimento da redução = 0.91;
- Rotação do motor = 60 rpm;

De acordo com o equacionamento exibido nos tópicos acima, o primeiro módulo foi dimensionado e obteve os seguintes resultados:

Potência ( $P$ ) = 4.2048 kW

Potência Estática( $P_s$ ) = 4.2033 kW

Potência Dinâmica ( $P_d$ ) = 0.0014425 kW

Rotação na saída do redutor ( $n_a$ ) = 114.56 rpm

Relação de Transmissão ( $I$ ) = 0.52360

#### 4.2.2.1.1 Perfil lateral

O perfil lateral selecionado para a construção do módulo 1 é do tipo perfil “U” simples com as dimensões exibidas na Figura abaixo (espessura de 2mm).

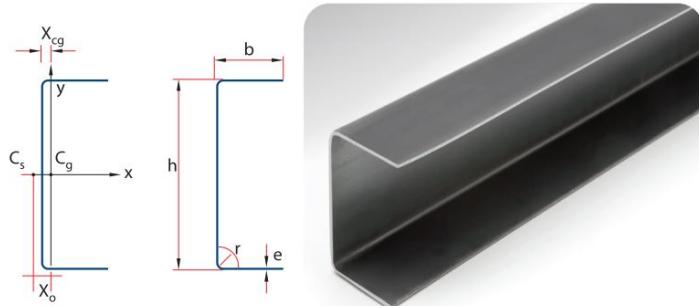


Figura 9 – Perfil “U” simples.

Dimensões Nominais			peso kgf/m	peso kgf/6m	área cm <sup>2</sup>	Direção X				Direção Y				X <sub>o</sub> (cm)	J <sub>t</sub> (cm <sup>4</sup> )	C <sub>w</sub> (cm <sup>6</sup> )	r <sub>o</sub> (cm)
h mm	b mm	esp. mm				J <sub>x</sub> (cm <sup>4</sup> )	W <sub>x</sub> (cm <sup>3</sup> )	i <sub>x</sub> (cm)	X <sub>cg</sub> (cm)	J <sub>y</sub> (cm <sup>4</sup> )	W <sub>y</sub> (cm <sup>3</sup> )	i <sub>y</sub> (cm)					
93	40	* 1,50	1,98	11,87	2,52	33,67	7,24	3,65	0,99	3,92	1,30	1,25	2,33	0,0189	58,20	4,51	
		* 1,80	2,36	14,17	3,01	39,84	8,57	3,64	1,01	4,65	1,55	1,24	2,32	0,0324	68,60	4,49	
		2,00	2,61	15,68	3,33	43,85	9,43	3,63	1,01	5,12	1,71	1,24	2,32	0,0443	75,32	4,48	
		2,25	2,92	17,55	3,73	48,75	10,48	3,62	1,03	5,70	1,92	1,24	2,31	0,0528	83,48	4,47	
		* 2,65	3,42	20,51	4,35	56,32	12,11	3,60	1,04	6,59	2,23	1,23	2,30	0,1018	95,98	4,44	
		* 3,00	3,84	23,05	4,89	62,69	13,48	3,58	1,06	7,35	2,50	1,23	2,30	0,1467	106,38	4,42	
		* 3,35	4,26	25,56	5,43	68,81	14,80	3,56	1,08	8,08	2,76	1,22	2,29	0,2028	116,29	4,40	
		* 3,75	4,73	28,38	6,03	75,52	16,24	3,54	1,09	8,89	3,06	1,21	2,28	0,2821	127,04	4,38	
		* 4,25	5,31	31,83	6,76	83,49	17,95	3,51	1,12	9,85	3,42	1,21	2,27	0,4065	139,64	4,35	
		* 4,75	5,87	35,21	7,48	90,98	19,57	3,49	1,14	10,76	3,76	1,20	2,26	0,5617	151,32	4,32	

Figura 10 – Dimensões Perfil “U” simples

#### 4.2.2.1.2 Base

O perfil da base selecionado para a construção do módulo 1 é do tipo perfil retangular com as dimensões exibidas na Figura abaixo (espessura de 2mm).

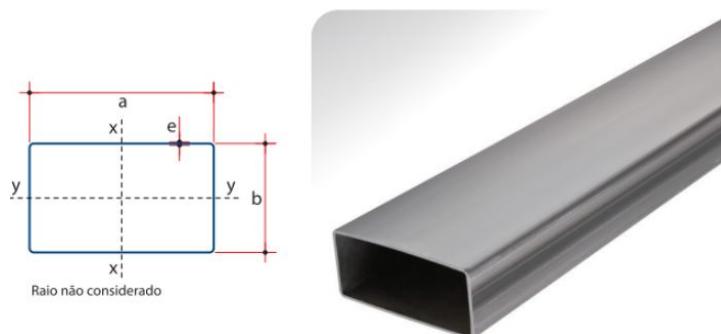


Figura 11 – Perfil retangular

Dimensões Nominais			peso kgf/m	peso kgf/6m	área cm <sup>2</sup>	Direção X				Direção Y				X <sub>o</sub> (cm)	J <sub>t</sub> (cm <sup>4</sup> )	C <sub>w</sub> (cm <sup>6</sup> )	r <sub>o</sub> (cm)
h mm	b mm	esp. mm				J <sub>x</sub> (cm <sup>4</sup> )	W <sub>x</sub> (cm <sup>3</sup> )	i <sub>x</sub> (cm)	X <sub>cg</sub> (cm)	J <sub>y</sub> (cm <sup>4</sup> )	W <sub>y</sub> (cm <sup>3</sup> )	i <sub>y</sub> (cm)					
68	30	* 1,50	1,45	8,69	1,85	13,07	3,85	2,66	0,77	1,61	0,72	0,93	1,76	0,0138	12,62	3,32	
		* 1,80	1,73	10,35	2,20	15,39	4,53	2,65	0,78	1,90	0,86	0,93	1,75	0,0237	14,78	3,31	
		2,00	1,91	11,44	2,43	16,88	4,96	2,64	0,79	2,08	0,94	0,93	1,74	0,0323	16,16	3,29	
		2,25	2,13	12,78	2,71	18,68	5,49	2,62	0,81	2,31	1,05	0,92	1,74	0,0457	17,81	3,28	
		* 2,65	2,48	14,89	3,16	21,42	6,30	2,60	0,82	2,65	1,22	0,92	1,73	0,0739	20,30	3,26	
		* 3,00	2,78	16,69	3,54	23,68	6,97	2,59	0,84	2,94	1,36	0,91	1,72	0,1062	22,33	3,24	
		* 3,35	3,08	18,46	3,92	25,82	7,60	2,57	0,86	3,21	1,50	0,91	1,71	0,1465	24,22	3,22	
		* 3,75	3,41	20,43	4,34	28,12	8,27	2,55	0,88	3,51	1,65	0,90	1,71	0,2031	26,22	3,19	
		* 4,25	3,80	22,83	4,85	30,77	9,05	2,52	0,90	3,86	1,84	0,89	1,70	0,2915	28,49	3,17	
		* 4,75	4,19	25,14	5,34	33,19	9,76	2,49	0,92	4,17	2,01	0,88	1,69	0,4011	30,52	3,14	
75	40	* 1,50	1,77	10,61	2,25	20,50	5,47	3,02	1,10	3,67	1,27	1,28	2,52	0,0169	34,98	4,14	
		* 1,80	2,11	12,64	2,68	24,20	6,45	3,00	1,12	4,34	1,51	1,27	2,52	0,0290	41,16	4,12	
		* 2,00	2,33	13,98	2,97	26,60	7,09	2,99	1,13	4,78	1,66	1,27	2,51	0,0395	45,13	4,11	
		2,25	2,61	15,64	3,32	29,52	7,87	2,98	1,14	5,32	1,86	1,27	2,51	0,0560	49,94	4,10	
		* 2,65	3,04	18,26	3,88	34,01	9,07	2,96	1,16	6,15	2,16	1,26	2,50	0,0907	57,28	4,07	
		* 3,00	3,42	20,51	4,35	37,76	10,07	2,94	1,17	6,85	2,42	1,25	2,49	0,1305	63,34	4,06	
		* 3,35	3,79	22,72	4,82	41,34	11,02	2,93	1,19	7,52	2,68	1,25	2,48	0,1803	69,09	4,04	
		* 3,75	4,20	25,20	5,35	45,23	12,06	2,91	1,21	8,26	2,96	1,24	2,48	0,2505	75,27	4,02	
		* 4,25	4,71	28,23	5,99	49,81	13,28	2,88	1,23	9,14	3,30	1,23	2,47	0,3605	82,46	3,99	
		* 4,75	5,20	31,18	6,62	54,06	14,42	2,86	1,26	9,97	3,64	1,23	2,46	0,4974	89,06	3,96	
		* 1,50	1,74	10,46	2,22	27,39	5,89	3,51	0,65	1,76	0,75	0,89	1,54	0,0166	26,56	3,94	

Figura 12 – Dimensões perfil retangular

#### 4.2.2.1.3 Rlete

O rolete selecionado possui 50 mm de diâmetro externo e 300 mm de comprimento, conforme dados de entrada para o dimensionamento do sistema de redução. O rolete possui a roda dentada acoplada ao seu eixo e seu desenho técnico pode ser observado abaixo.

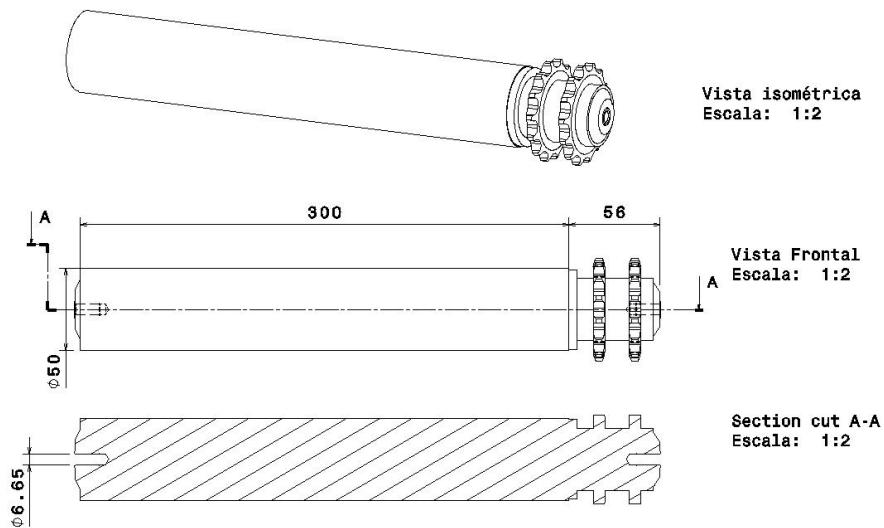


Figura 13 – Rlete selecionado.

## 4.3 Solução Geral Energia

### 4.3.1 Alimentação da Estrutura

Na estrutura do seletor estão presentes um sistema de pré processamento, que consiste em uma lavagem e secagem dos frutos, e um sistema de movimentação dos frutos, dividido em duas etapas: utilizando roletes acionados que realizam rotação em seu próprio eixo e sobre uma correia transportadora. A alimentação desses componentes está apresentada no diagrama unifilar. A alimentação em todos os componentes é realizada, primariamente, pelo painel fotovoltaico que alimenta a bateria e da bateria sai a carga determinada para cada elemento da máquina.

#### 4.3.1.1 Alimentação do pré-processamento

No sistema de pré processamento dos frutos as laranjas são jateadas com água e atritadas contra roletes, os quais possuem cerdas em sua superfície, para retirar material particulado e/ou matéria orgânica aderida em sua superfície, como folhas ou pequenos

galhos que podem interferir no processamento de imagem. Logo após, as laranjas são sopradas para retirar gotículas de água restantes do processo.

Neste sistema estão presentes:

- Bomba de água submersível

Para o sistema de bombeamento de água, optou-se por uma bomba de água submersível de 12 volts, o que exclui a necessidade de usar componentes para alterar a tensão proveniente da bateria, que é de 12 volts. Além disso, optou-se por um equipamento que atendesse a demanda de água para a lavagem dos frutos. Segundo as especificações do componente, este recebe uma corrente máxima de 1 A e possui 12W de potência.



Figura 14 – Bomba de água submersível

- Ventoinha - Insuflador de ar

Para realizar a retirada das gotículas de água dos frutos, foi selecionado um insuflador de ar com 10”, com uma tensão nominal de 12 volts, facilitando sua ligação à bateria e com corrente de 6.5A. Sua potência é de 78 W.



Figura 15 – Ventoinha

- Motor com redução MB26012

O motor encontra-se tanto no pré processamento quanto no restante do sistema de movimentação das cargas. Nos roletes iniciais da máquina, onde a fruta é lavada, há cerdas para limparem os frutos. Esse motor possui tensão nominal de 12 volts, para facilitar a sua ligação à bateria de alimentação, e corrente de 10 A. A sua potência é de 120 W.



Figura 16 – Motor

#### 4.3.2 Alimentação dos componentes eletrônicos

A alimentação dos componentes eletrônicos é dividida em três circuitos: um circuito com 5 volts, para os componentes que possuem tensão nominal de 5 volts; um circuito com 7 volts, para os componentes que possuem demanda de tensão nominal de 7 volts; e

um circuito com 12 volts, para alimentação dos componentes com demanda de 12 volts. A apresentação dos circuitos de alimentação é feita no diagrama unifilar e pode ser vista nos anexos deste relatório.

- 2 Servos Motores MG996R

Os servos motores serão alimentados por uma tensão nominal de 7 volts, a qual será fornecida pela bateria. Possuem uma potência de 6.3 W. São dois servos, conforme pode ser visto no esquemático de estrutura e seu correto dimensionamento.

- 1 Monitor IPS 10.1”

O monitor será alimentado por uma tensão nominal de 12 volts, com uma corrente máxima de 3 A, totalizando em 36 W de potência. Essa alimentação será provida pela bateria.

- 1 Raspberry Pi4 8GB

A Raspberry, a qual é responsável pela alimentação dos sensores, módulo de câmera, entre outros componentes que integram a parte eletrônica do projeto, possui tensão nominal de 5 volts e corrente de 3 A, totalizando 15 W de potência. Será alimentada pela bateria no circuito de 5 volts.

#### 4.3.3 Servos motores

Os motores selecionados para realizar a movimentação dos frutos dependem das demandas de torque dos do sistema frutas+estrutura. Dessa forma, foi selecionado um motor com potência superior ao valor necessário para movimentar o sistema, considerando a margem necessária pelo fator de segurança. As componentes que integram a potência necessária para movimentar o sistema podem ser definidas por:

Os servos motores terão a função de direcionar as frutas, caso seja necessário, ruínas e/ou frutas com manchas para compartimentos separados daquele destinado aos frutos adequados para o consumo *in natura*. Para realizar essa separação, o motor deve ser capaz de resistir ao torque gerado pela fruta no processo de encaminhamento para o reservatório. Para realizar o correto dimensionamento e seleção do componente eletrônico responsável por gerar esse torque resistivo, deve-se realizar o cálculo de ação e reação através da 2º Lei de Newton para cálculo de força, posteriormente encontrando o torque realizado pela fruta sobre o braço mecânico do servo motor. Para tanto, o equacionamento pode ser encontrado abaixo.

A força resultante gerada pela laranja sobre a esteira em contato com o braço mecânico pode ser medida a partir da variação da quantidade de movimento do objeto, de modo que:

$$F = \left( \frac{\Delta Q}{\Delta t} \right) = m \cdot \frac{\Delta v}{\Delta t} \quad (4.9)$$

$V_f = 0$  m/s, pois o fruto, ao entrar em contato com o braço mecânico deixa de se movimentar no eixo da esteira e segue perpendicular em direção ao reservatório.

A velocidade escolhida para a esteira, por critério do processamento de imagem, é de 0,3 m/s. A massa média de uma laranja é de  $m = 0,120$  kg e  $t = 0.1$  s.

Dessa forma, para uma distância relativa ao centro de massa do fruto, com diâmetro médio de 10cm.

Logo, é preciso de um motor com, considerando um fator de segurança de  $FS = 1,5$ , que seja capaz de gerar Torque superior a 2,7 N.cm.

#### 4.3.4 Sistema Fotovoltaico Autônomo

Um sistema fotovoltaico autônomo é bastante útil em regiões que não são atendidas pela rede elétrica da concessionária ou para manter algum equipamento ausente da rede elétrica com operação contínua.

De acordo com Green, um sistema autônomo procura maximizar o aproveitamento da energia solar e o armazenamento da energia de reserva, com o intuito de obter o máximo proveito entre a tecnologia e a economia (GRENN, 2000). Um dos desafios encontrados pelo sistema autônomo é o alto custo de instalação e uma baixa eficiência. Um dos métodos utilizados para reduzir os custos da energia gerada, é garantir que o sistema opere o maior tempo sobre o ponto de máxima potência dos painéis. Entretanto, os painéis são influenciados diretamente pelas condições atmosféricas (MASOUM SEYED MAHDI MOUSAVI BADEJANI, 2004).

Visando atingir por um maior tempo o ponto de máxima potência dos módulos fotovoltaicos, utilizou-se neste seguinte trabalho a tecnologia MPPT (*Maximum Power Point Tracking*), a qual, pode acrescentar entre 15 e 30% da energia produzida (CAVALCANTI K. C. OLIVEIRA, 2007).

##### 4.3.4.1 Componentes

O esquema de um sistema isolado pode ser visto na Figura 17. No presente caso, as cargas serão alimentadas em corrente contínua, ou seja, não há a necessidade de um inversor. Logo, os principais componentes deste sistema são: módulos fotovoltaicos, con-

troladores de carga, baterias, condutores e um sistema de proteção. Serão detalhados esses componentes nos próximos tópicos.

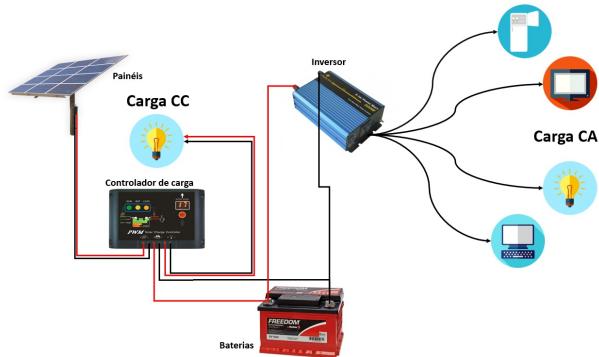


Figura 17 – Esquema de um sistema fotovoltaico isolado.

#### 4.3.4.1.1 Módulo Fotovoltaico

No mercado atual existem basicamente três tipos comerciais de módulos fotovoltaicos: os monocristalinos, os policristalinos e os de filmes finos. Os módulos monocristalinos são obtidos a partir de um único grande cristal mergulhado em silício fundido ([BRITO, 2014](#)). Depois é submetido a altas temperaturas garantindo confiabilidade e eficiência dos produtos ([MACHADO C.; MIRANDA, 2014](#)).

A eficiência dos módulos policristalinos está abaixo dos de silício monocristalino, isso se deve ao fato do processo de fabricação ser diferente, apesar de possuírem a mesma matéria-prima ([RUTHER, 2004](#)).

A tecnologia de filmes finos tem a grande vantagem de produzir um maior número de células com menos matéria-prima que as demais. Entretanto, a conversão de fótons em elétrons é menos eficiente que as fabricadas de cristais de silício ([VILLALVA M.; GAZOLI, 2012](#)).

#### 4.3.4.1.2 Controlador de carga

Também chamado de regulador de carga, este componente é, normalmente, utilizado em sistemas *off grid*, ou seja, que empregam o uso de baterias para o armazenamento de energia. De acordo com Pereira e Oliveira, os controladores de carga têm como principal função proteger os acumuladores, isto é, as baterias de sobrecargas do sistema ([PEREIRA F.; OLIVEIRA, 2011](#)). Além disso, se bem regulados, asseguram que o sistema opere em sua máxima eficiência.

#### 4.3.4.1.3 Bateria

As baterias, entre os diversos sistemas, são mais utilizadas naqueles isolados da rede elétrica, ou seja, *off grid*. Segundo Dazcal e Mello, são dispositivos responsáveis por fazer o armazenamento da energia elétrica gerada pelos módulos, com o intuito de suprir a demanda da mesma na ausência da radiação solar (DAZCAL R.; MELLO, 2008). Com isso, podem ser consideradas de extrema importância, já que a radiação solar não ocorre nos períodos noturnos, e é reduzida em dias nublados. Considerando o fato de que nem todos os modelos são economicamente viáveis, Messenger e Ventre apresentam as baterias de chumbo-ácido como o tipo, até hoje, mais utilizado para os sistemas fotovoltaicos (MESSENGER R.; VENTRE, 2010).

#### 4.3.4.1.4 Condutores

Para dimensionar os condutores, deve-se observar as diversas condições as quais estarão expostas. Os condutores terão dois lados de ligação: o lado que conecta os módulos ao controlador e o lado que conecta o conversor à carga e banco de baterias.

O lado que conecta os módulos ao controlador estará exposto a condições climáticas e outras interferências, por isso, é recomendado que os cabos sejam mais resistentes, como monocondutores com dupla camada de isolação.

Já os condutores que conectam o controlador de carga ao banco de baterias e à carga podem ser menos robustos, pois não estão submetidos a condições degradantes.

#### 4.3.4.1.5 Proteção

A proteção dos sistemas fotovoltaicos deve ser feita tanto para sobrecorrentes quanto para correntes reversas. Os dispositivos mais comuns utilizados são os fusíveis, dispositivos de proteção contra surtos (DPS's) e as chaves seccionadoras. Os fusíveis servem como proteção para sobrecarga e correntes residuais, os DPS's para a proteção de sobretensões e as chaves seccionadoras para desconectar os módulos dos controladores (MOTTA, 2019).

#### 4.3.4.2 Desenvolvimento

O desenvolvimento da solução foi segmentada em três partes:

- Coleta de dados, no qual foram captadas informações geográficas referentes à região a ser implementado o sistema, além de levantamento de informações da carga a ser alimentada;

- Realização de cálculos para dimensionamento do sistema, incluindo os principais componentes já citados;
- Realização de simulações a fim de validar os cálculos e obter relatórios de eficiência completos.

#### 4.3.4.2.1 Coleta de dados

Foram levadas em considerações as incidências de radiação na cidade de Avaré - SP, pois a cidade é considerada uma das maiores produtoras de laranja do país. Os dados referentes à longitude de latitude foram obtidos pelo software *Google Maps*, enquanto dos dados referentes à incidência solar foram coletados na plataforma SunData do CRESESB (??). Além disso, a inclinação adotada foi o valor de 23º em relação à horizontal devido à latitude do local escolhido. A Tabela 15 reúne os dados coletados da incidência solar média mensal durante um período de um ano.

Mês	Irradiação Global [kWh/m².dia]	Irradiação Inclinada [kWh/m².dia]
Janeiro	5,50	4,99
Fevereiro	5,66	5,40
Março	5,02	5,17
Abril	4,54	5,17
Maio	3,61	4,45
Junho	3,36	4,36
Julho	3,48	4,42
Agosto	4,48	5,32
Setembro	4,66	4,97
Outubro	5,25	5,13
Novembro	5,82	5,33
Dezembro	6,10	5,42

Tabela 15 – Incidência solar média mensal para a cidade de Avaré.

A irradiação global pode ser definida como a radiação total que chega à atmosfera terrestre. Já a irradiação inclinada é a radiação que chega perpendicular à superfície receptora para geração de energia.

O consumo de carga no sistema é dividido entre os componentes eletrônica (16.66% da carga total do sistema), componentes de pré processamento (20.44% da carga) e os motores para movimentação dos frutos (62.9% da carga) ao longo das esteiras. Todos

os componentes operam em corrente contínua. A descrição detalhada dos componentes os quais serão alimentados pelo sistema fotovoltaico e de baterias pode ser encontrada na Tabela 16 com as respectivas demandas de tensão, corrente e potência por componente.

Qtde.	Componentes	Tensão Nominal [V]	Corrente [A]	Potência [W]
1	Raspberry Pi4 8GB	5	3	15
2	Servo Motor MG996R	7	0,9	6,3
1	Monitor LCD 10.1"	12	3	36
2	Motor com Redução MB26012	12	10	120
1	Ventoinha 10"	12	6,5	78
1	Bomba	12	1	12

Tabela 16 – Componentes da carga

O sistema é alimentado primariamente, por um painel fotovoltaico, havendo geração de eletricidade da maior parte do período de trabalho referente ao dia. A demanda energética dos componentes é 393,6W, de modo que o painel solar selecionado deve garantir alimentação suficiente ao sistema. Há uma alimentação redundante, através de um conjunto de baterias operando em paralelo, considerando a elevada demanda de corrente dos componentes. O conjunto de baterias deve ter a capacidade de prover carga ao sistema por um período equivalente a 10h de trabalho por dia, para caso de redução de incidência solar ou sombras provenientes de árvores, nuvens ou qualquer outro efeito natural. Além disso, o conjunto de baterias acumula energia elétrica proveniente de fonte solar.

#### 4.3.4.3 Componentes Escolhidos

- Painel Fotovoltaico

A escolha dos módulos fotovoltaicos baseou-se, principalmente, no resultado das simulações, levando em conta seus relatórios de eficiência a fim de escolher o modelo que mais se adéque ao caso em estudo.

A figura 18 contém, de forma resumida, as informações obtidas na simulação, dos componentes necessários para o funcionamento do sistema.

Stand alone system: Detailed User's needs					
Project :	PI2				
Simulation variant :	PI2				
<b>Main system parameters</b>	<b>Stand alone system with batteries</b>				
PV Field Orientation	System type	tilt	23°	azimuth	0°
PV modules	Model	CS3U-355PB-AG 1500V HE Pnom			
PV Array	Nb. of modules	4	Pnom total		355 Wp
Battery	Model	Open 12V / 100 Ah		Technology	<b>1420 Wp</b>
Battery pack	Nb. of units	4	Voltage / Capacity	Lead-acid, vented, plate	
User's needs	Daily household consumers	Constant over the year		Global	<b>12 V / 400 Ah</b>
Daily household consumers, Constant over the year, average = 3.9 kWh/day					

Figura 18 – Resumo dos valores obtidos em simulação.

A Figura 19 e a Tabela 17 apresentam, respectivamente, a imagem e as especificações do módulo fotovoltaico.



Figura 19 – Módulo Fotovoltaico 355W Canadian Policristalino

Especificação	Modelo CS3U 355P
Marca	Canadian
Tensão de máxima potência - Vmp(V)	39,4
Corrente de máxima potência - Imp(A)	9,02
Tensão Circuito Aberto - Voc(V)	46,8
Corrente Curto Circuito - Isc(A)	9,59
Potência Módulo (Wp)	355
Eficiência (%)	17,89
Coeficiente de temperatura da potência (%/°C)	-0,37
Coeficiente de temperatura da tensão (%/°C)	-0,29
Coeficiente de temperatura da corrente (%/°C)	0,05

Tabela 17 – Especificações Módulo CS3U 355P.

Com relação à produção, a figura 20 mostra as produções de energia normalizada por mês, especificando: as perdas na captação solar, na cor **roxo**; a energia útil fornecida efetivamente, na cor **bordô**; as perdas no sistema instalado, na cor **verde**; as perdas de geração quando as baterias estão carregadas, na cor **azul**.

A figura 21 resume a eficiência da geração de cada mês. Como pode-se observar na imagem, a eficiência é de 95%, havendo a baixa nos meses que possuem menor irradiação.

**Normalized productions (per installed kWp): Nominal power 1420 Wp**

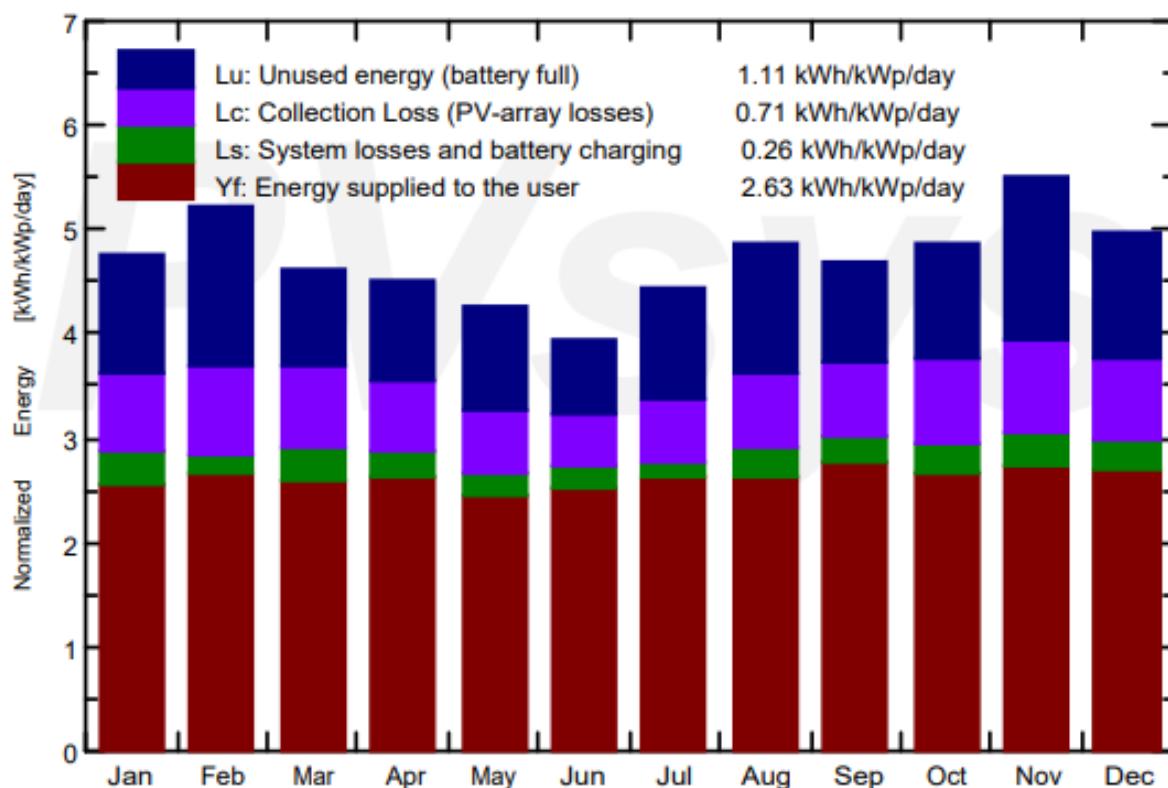


Figura 20 – Produção normalizada

### Performance Ratio PR and Solar Fraction SF

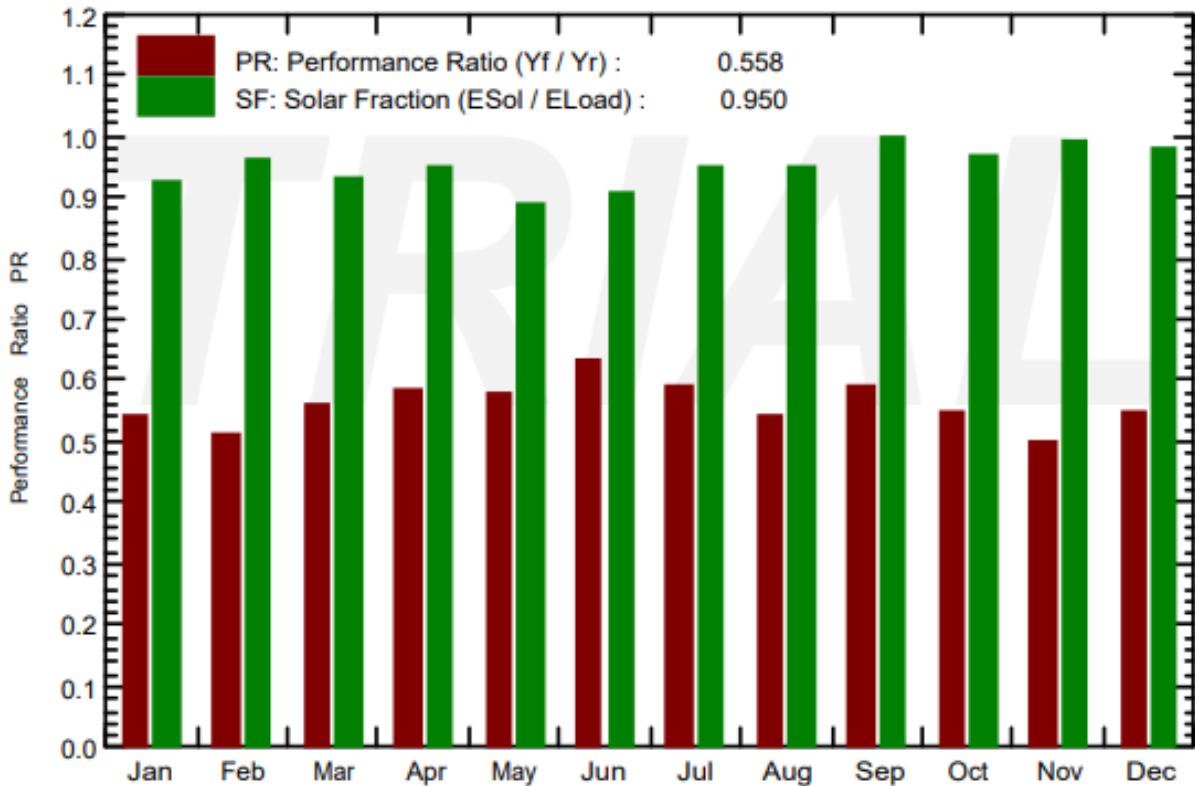


Figura 21 – Rendimento global do sistema

- Controlador de Carga

O controlador de carga também foi definido a partir de simulação. O próprio *software* executa e transparece o modelo mais adequado de acordo com a tecnologia escolhida. No caso em estudo, a tecnologia adotada foi o MPPT, como já dito. As especificações do controlador podem ser encontradas abaixo, assim como sua imagem.

<b>Controller</b>		<b>Model</b>	Universal controller with MPPT converter
Converter	Technology	MPPT converter	Temp coeff. -5.0 mV/°C/Elem.
	Maxi and EURO efficiencies	97.0 / 95.0%	
Battery Management control	Threshold commands as	SOC calculation	
	Charging	SOC = 0.92 / 0.75	approx. 13.8 / 12.4 V
	Discharging	SOC = 0.20 / 0.45	approx. 11.6 / 12.4 V

Figura 22 – Especificações do controlador de acordo com a simulação.



Figura 23 – Controlador de carga Epever

Especificação	Descrição Técnica
Marca	Epever
Modelo	XTRA 3210N
Corrente de carga (A)	30
Corrente de descarga (A)	30
Algoritmo de carga	MPPT
Máxima tensão da bateria (V)	32

Tabela 18 – Especificações Controlador de Carga.

- Bateria

Análogo ao controlador de carga e aos módulos fotovoltaicos, a bateria também foi escolhida de acordo com o resultado da simulação figura 24. Sendo necessária a utilização de 4 baterias em paralelo.

System Parameter	System type	Stand alone system
<b>Battery</b>	Model	<b>Open 12V / 100 Ah</b>
	Manufacturer	Generic
Battery Pack Characteristics	Nb. of units	4 in parallel
	Voltage	12 V
	Discharging min. SOC	20.0%
	Temperature	Average between fixed (25°C) and External
		Nominal Capacity 400 Ah
		Stored energy 3.8 kWh

Figura 24 – Especificações do controlador de acordo com a simulação.

- Condutores

Especificação	Descrição Técnica
Marca	Moura
Modelo	M100qd
Voltagem (V)	12
Capacidade da bateria(Ah)	100

Tabela 19 – Especificações Bateria.

O dimensionamento dos dispositivos de proteção seguiram as normas NBR 16.690 ([NBR...](#), 2020a), NBR 5.410 ([NBR...](#), 2020b) e a norma europeia IEC 60364-7-712. A primeira, exemplifica que os componentes do lado em corrente contínua de sistemas fotovoltaicos, como, por exemplo, módulos fotovoltaicos, caixas de junção, quadros de distribuição e condutores, até os terminais em corrente contínua do inversor, devem possuir isolação dupla ou reforçada (classe II). Para mais, deve ser calculada a máxima corrente (Equação 4.10) que circulará pelos condutores. De acordo com a norma europeia citada, a capacidade de condução de corrente pelos condutores das strings deve ser 25% maior que a corrente do módulo fotovoltaico à temperatura de 80°C. A figura 25 retirada da NBR 5410 mostra os valores de seções nominais dos condutores, com isolamento EPR, de acordo com o valor de corrente máxima.

$$I_{sc(string)} = 1,25 * [I_{sc} + (T_f - T_i) * Cof_{isc} * I_{sc}] \quad (4.10)$$

$$I_{sc(string)} = 1,25 * (9,59 + 55 * 0,0005 * 9,59)$$

$$I_{sc(string)} = 11,10A$$

Observando a figura 25 e o valor de corrente máxima obtido, é notável que a área de seção nominal escolhida será de 2,5mm<sup>2</sup>.

### Capacidade de condução de corrente

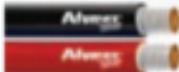
Seção mm <sup>2</sup>	Instalação ao ar livre protegida do sol				Instalação ao ar livre exposta ao sol			
	Modo de instalação [3]							
	1	3	1	3	1	3	1	3
Temperatura de operação no condutor								
	$90^{\circ}\text{C}$ [4]		$120^{\circ}\text{C}$ [5]		$90^{\circ}\text{C}$ [4]		$120^{\circ}\text{C}$ [5]	
2,5	35	40	34	38	31	36	29	35
4	47	53	45	51	41	48	39	46
6	60	68	57	65	51	61	49	59
10	83	95	79	90	71	85	68	81
16	110	125	105	120	93	112	89	107
25	146	166	140	159	123	147	117	141
35	181	207	174	198	151	182	145	174
50	229	260	219	249	189	228	181	218
70	285	325	273	311	234	283	224	271
95	343	390	328	374	279	339	267	324
120	402	458	385	438	325	396	311	379
150	463	527	443	504	371	453	355	434
185	528	600	506	574	420	513	402	491
240	633	719	606	688	499	612	477	586

[1] Resistência ôhmica em corrente contínua à temperatura de  $90^{\circ}\text{C}$ ;

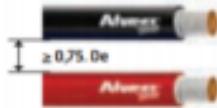
[2] Cálculo da queda de tensão considerando um sistema de corrente contínua;

**Nota:** para a correção do  $\text{R}_{\text{cc}}$  e da queda de tensão a  $120^{\circ}\text{C}$ , multiplicar o  $\text{R}_{\text{cc}}$  e queda de tensão a  $90^{\circ}\text{C}$  pelo fator de 1,0925.

[3] Modo de instalação:



**Modo de instalação 1:** dois cabos unipolares encostados um ao outro na horizontal.



**Modo de instalação 3:** dois cabos unipolares espaçados em pelo menos  $0,75 \times \text{D}_{\text{e}}$  na horizontal.  
 $\text{D}_{\text{e}} =$  Diâmetro nominal externo do cabo completo.

[4] Tabela C.2 da NBR 16612 - Capacidade de condução de corrente para cabos instalados em temperatura ambiente de  $30^{\circ}\text{C}$  e temperatura no condutor em regime permanente de  $90^{\circ}\text{C}$ ;

Figura 25 – Capacidade de condução de corrente, em ampères.

- Proteção

Como já mencionado, o dispositivo de proteção conta com fusíveis, chaves seccionadoras e DPS's. Para a escolha dos fusíveis foi observada a corrente máxima de operação, ou seja, a amperagem do fusível deve ser maior que a corrente máxima de operação adicionando um fator de segurança de 20%. Levando em conta o valor obtido pela Equação 4.10 o fusível escolhido foi com a tecnologia GPV de 15A (Figura 26). Com relação à quantidade, será um fusível para cada *string* de ligação dos módulos.



Figura 26 – Fusível GPV 15A.

#### 4.3.4.4 Softwares utilizados

- *PVSyst*

Devido à imprevisibilidade e variações do recurso solar, optou-se pela utilização dos *software PVSyst* para a simulação e validação dos cálculos realizados no ponto de controle anterior. De acordo com a descrição, o *PVSyst* possui as funcionalidades de especificar e dimensionar os componentes de um sistema fotovoltaico, além de verificar o projeto e validar seus componentes. Caso necessário, o *software* também fornece relatórios de desempenho e análise das principais causas de perdas nos sistemas. Basicamente, as informações inseridas para obter a simulação foram: 1) coordenadas geográficas do local a ser instalado; 2) dados de irradiação; 3) especificações da bateria e do módulo fotovoltaico.

- AutoCAD

O AutoCAD é um *software CAD* (projeto auxiliado por computador) que arquitetos, engenheiros e profissionais de construção utilizam para criar desenhos 2D e 3D precisos (DESCRIÇÃO..., 2020). Para mais, ele auxilia na construção de diagramas elétricos. No presente trabalho, ele foi utilizado como ferramenta para construção dos diagramas unifilares.

#### 4.3.4.5 Diagrama Unifilar

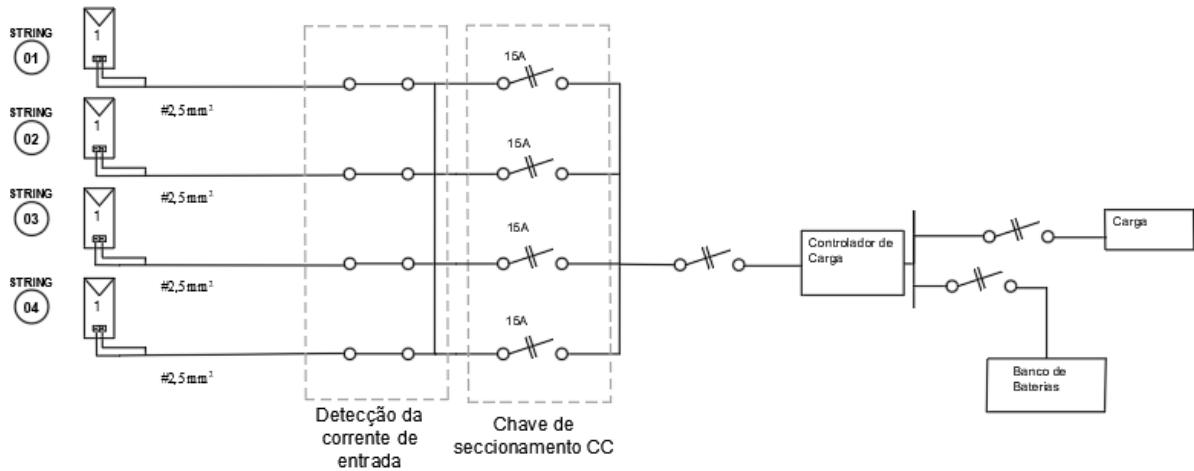


Figura 27 – Diagrama unifilar do sistema de operação.

##### 4.3.4.5.1 Estrutura de fixação

Como suporte dos módulos fotovoltaicos, foram selecionadas estruturas de alumínio como na figura 28 abaixo. A estrutura tem capacidade para 4 módulos com potência



Figura 28 – Suporte para módulos fotovoltaicos.

máxima de 500W e será fixada em uma superfície plana no chão.

## 4.4 Arquitetura de Eletrônica

### 4.4.1 Solução Geral de Eletrônica

A construção da arquitetura de eletrônica foi segmentada em quatro módulos de atuação: aquisição de imagem, definição de rotas, contagem e disponibilização de dados.

- O módulo de aquisição de imagem é constituído por câmeras e é responsável por controlar as câmeras e obter a imagem do fruto a ser classificado.
- O módulo de definição de rotas tem como objetivo orientar o fruto através de duas cancelas automatizadas utilizando servo-motores para separação dos frutos.
- O módulo de contagem contabiliza os frutos de acordo com as classificações determinadas: laranjas boas sem manchas pequenas, médias e grandes.
- O módulo de disponibilização de dados permite a visualização dos dados obtidos, a partir do resultado da classificação dos frutos, como por exemplo, gráficos, tabelas e dados estatísticos.

Após definição do escopo e dos requisitos do sistema, foram levantados dados técnicos de componentes eletrônicos para implementação da solução. A escolha dos componentes e as suas respectivas especificações técnicas, bem como esquemáticos e diagramas de controle estão descritos nos tópicos seguintes.

O diagrama de blocos da figura 29 ilustra os módulos de atuação e apresenta o microcontrolador responsável por gerenciar os periféricos que integram cada módulo, transmitindo e recebendo os dados.

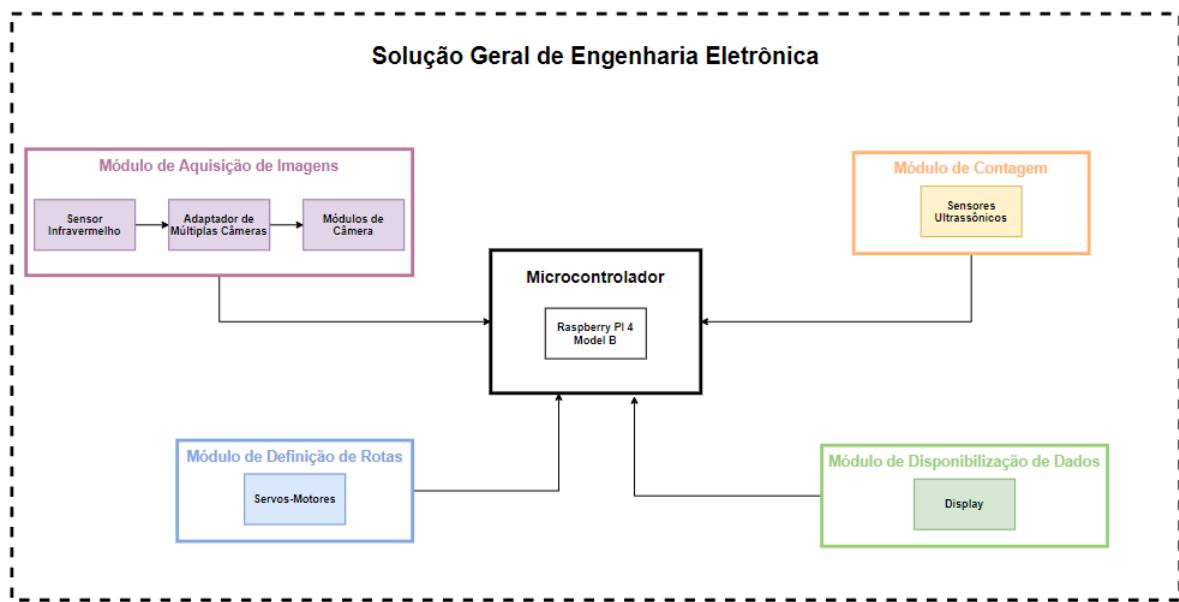


Figura 29 – Diagrama de Blocos da Solução Eletrônica.

## 4.4.2 Diagramas Eletrônicos

### 4.4.2.1 Diagrama de Alimentação

O diagrama abaixo (Figura 30) apresenta a tensão e a corrente de alimentação de cada componente e a integração entre os mesmos.

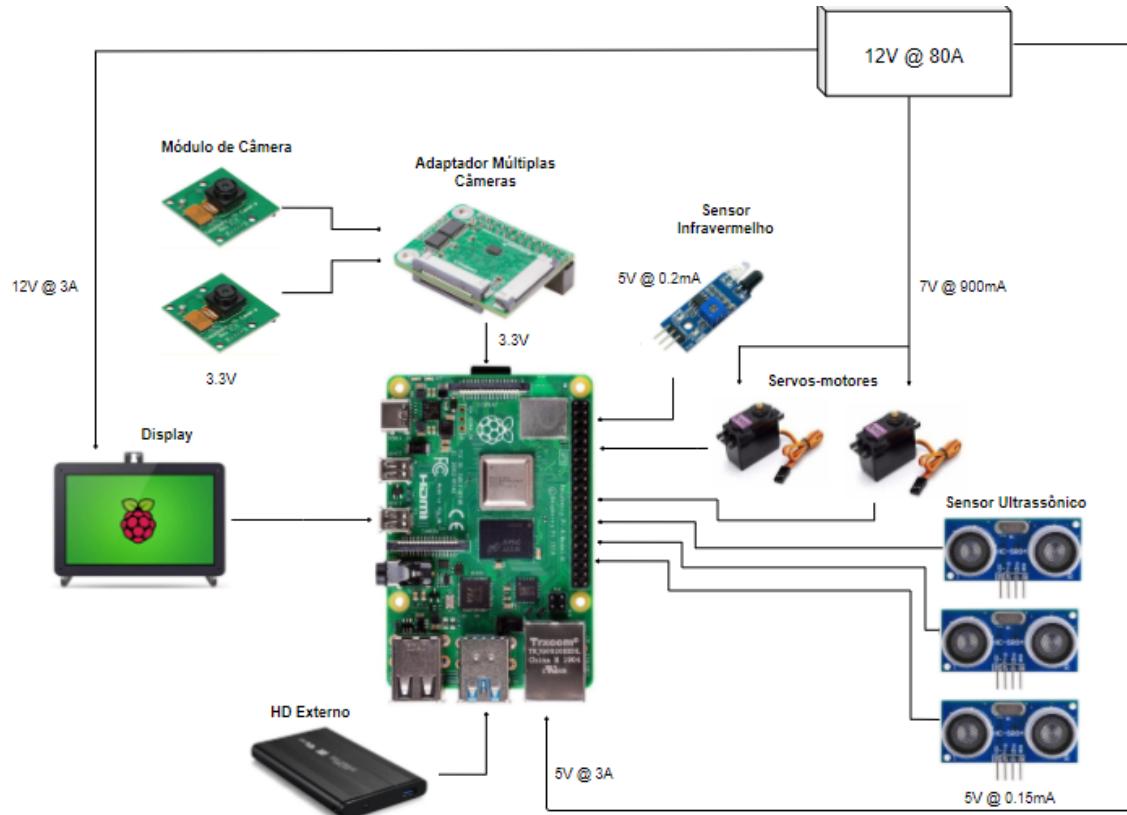


Figura 30 – Sistema Eletrônico.

#### 4.4.2.2 Diagrama Circuito Eletrônico

A figura 31 apresenta o diagrama geral da arquitetura do sistema que será implementado.

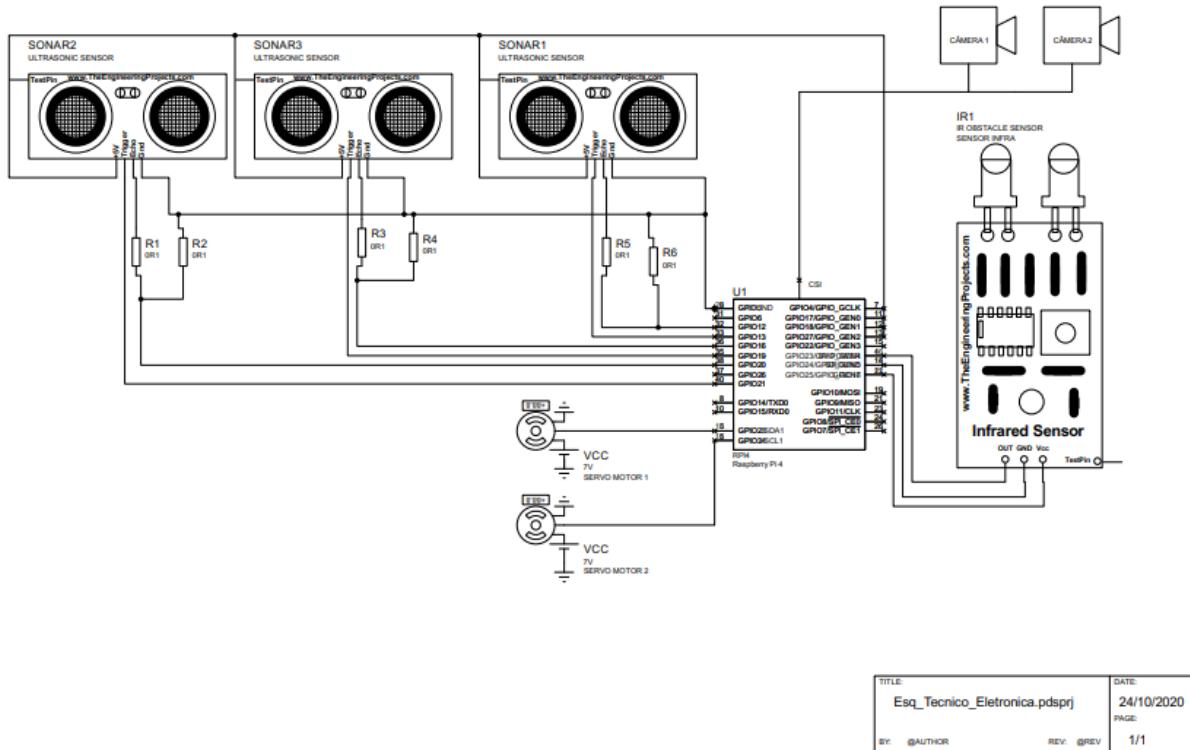


Figura 31 – Diagrama Circuito Eletrônico.

### 4.4.3 Microcontrolador

Para implementar a solução proposta optou-se pelo uso da *Raspberry PI 4 Model B*. Esse modelo é ideal para projetos com processamento de imagem, pois apresenta um rápido processamento, conector CSI (*Camera Serial Interface*) dedicado para conexão de câmeras e troca de dados e interface para display DSI (*Display Serial Interface*). ([RASPBERRY..., 2020](#))

Somado a isso, conta com um armazenamento mais rápido e conexões de rede via *Gigabit Ethernet*, duas saídas HDMI (*High-Definition Multimedia Interface*) que suportam uma resolução de até 4K, 3840 x 2160 pixels, e duas portas USB 3.0. O modelo escolhido para implementação do projeto foi de 8 GB DDR4 de memória RAM. A tabela abaixo apresenta as especificações técnicas do modelo da Raspberry Pi escolhida. ([RASPBERRY..., 2020](#))

Parâmetro	Valor
Frequência do processador	1500 MHz
Memória - RAM	8 GB
USB Ports	2xUSB2, 2xUSB3
Ethernet	1000Base-T
Wireless	802.11ac/n
Bluetooth	5.0
Alimentação	5V DC / 3A
GPIO	40 Pinos

Tabela 20 – Especificações Raspberry Pi 4.

Neste projeto, esse microcontrolador será responsável por compilar e interpretar o *software* de processamento de imagem através dos dados adquiridos pelas câmeras. Além disso, será responsável pelo controle dos periféricos do sistema, como sensores, servos e contadores aplicados no decorrer na estrutura física.

#### 4.4.3.1 Raspbian

Para o sistema operacional da Raspberry Pi 4 Model B será utilizado Raspbian OS, que é uma distribuição da Linux criada para executar os códigos na Raspberry Pi. Derivada do Debian, é considerado o sistema operacional padrão do computador da Raspberry Foundation ([GNU..., 2019](#)).

### 4.4.4 Módulo de Aquisição de Imagem

#### 4.4.4.1 Definição

O módulo de aquisição de imagem tem como objetivo o controle das câmeras e captura das imagens do fruto. Para o acionamento das câmeras, torna-se necessária a

utilização de um detector de presença para informar quando o fruto estiver dentro da etapa de captura.

Este módulo é constituído por um sensor infravermelho, dois módulos de câmera e um adaptador de múltiplas câmeras.

#### 4.4.4.2 Esquemático do Módulo de Aquisição de Imagem

A figura 32 apresenta o diagrama de montagem do módulo de aquisição de imagem. A câmera A e a câmera B são conectadas ao adaptador de múltiplas câmeras e este é conectado à porta de câmera CSI, o sensor infravermelho é conectado aos pinos 5V, *Ground* e GPIO 23 da placa.

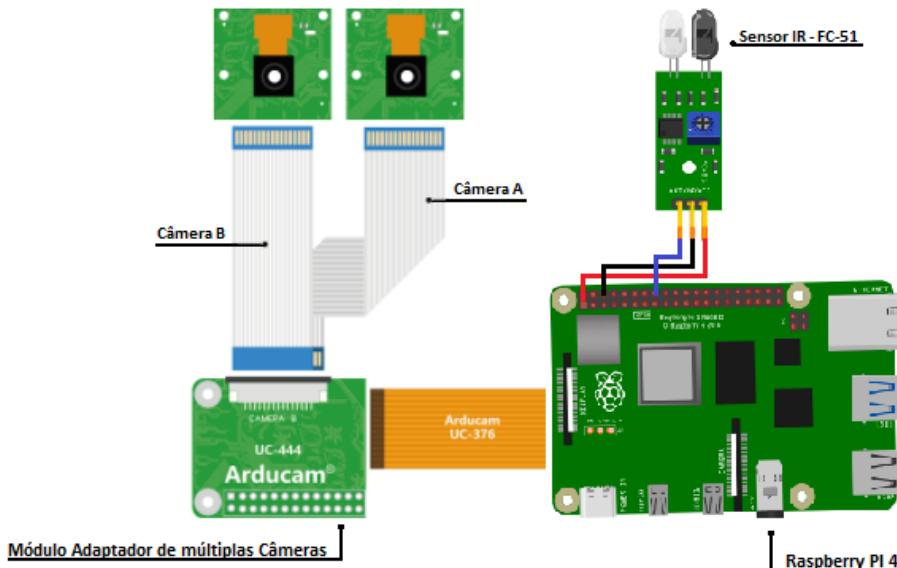


Figura 32 – Esquemático de Montagem do Módulo de Aquisição de Imagem.

#### 4.4.4.3 Diagrama de Controle do Módulo de Aquisição de Imagem

O diagrama abaixo (Figura 33) exemplifica o funcionamento do módulo de aquisição de imagem. Inicialmente, o LED verde é setado com nível lógico baixo e a variável Saída do sensor infravermelho em nível lógico alto. A partir da detecção de um fruto, o nível lógico do sensor infravermelho é setado em zero. Com isso, o LED verde é ativado, bem como a Câmera 1 e a Câmera 2. A ativação das câmeras libera a aquisição das imagens do fruto, que são posteriormente armazenadas e enviadas para o *software* de processamento de imagem.

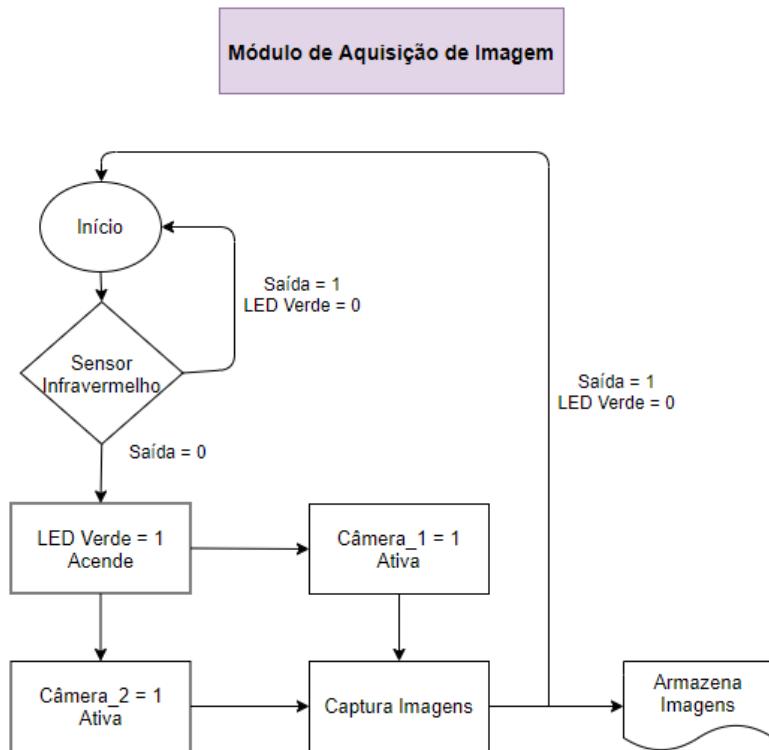


Figura 33 – Diagrama de Controle do Módulo de Aquisição de Imagem.

#### 4.4.4.4 Câmeras e Adaptador

Os estudos para escolha do módulo de câmeras foram guiados a partir da definição do microcontrolador. A utilização da Raspberry PI Model B permitiu a possibilidade de implementar o módulo de câmera lançado oficialmente para este computador de uso geral ([RASPERRY...](#), 2019).

O modelo escolhido para desenvolvimento do projeto foi o Módulo de Câmera 5mp Raspberry Pi V1.3 (Figura 34). Esta versão apresenta o OV5647 como sensor, permitindo imagens com alta resolução 1080p. O ajuste de foco permite regular a distância ao objeto sem alterar o posicionamento da câmera, facilitando o manuseio. Além disso, a Raspberry PI apresenta o conector CSI (Camera Serial Interface), específico para este tipo de equipamento ([SARAVATI...](#), 2019).

Para permitir o uso de mais um módulo, será utilizado um adaptador de múltiplas câmeras Raspberry Pi (Figura 35), projetado para conectar mais de uma câmera a uma única porta de câmera CSI na placa. A adição deste adaptador permite a visualização de um maior percentual da área do fruto que será fotografado. ([ARDUCAM...](#), 2019) ([ARDUCAM...](#), 2020)



Figura 34 – Módulo de Câmera.

Parâmetro	Valor
Resolução da câmera	5MP
Comprimento Focal	3.6mm
Ângulo de visão (diagonal)	75.7°
Sensor	OV5647
CCD Size	1/4 polegadas
Peso	20g
Dimensões	5 x 24 x 23.5 mm

Tabela 21 – Especificações Módulo de Câmera.



Figura 35 – Adaptador de Múltiplas Câmeras.

#### 4.4.4.5 Sensor Infravermelho

O sensor infravermelho reflexivo de obstáculo (Figura 36) funciona através de um sistema de reflexão infravermelho, em que um LED emissor IR (Infravermelho) e um fototransistor IR ficam lado a lado. Quando um obstáculo ou objeto atravessa dentro do raio de ação de ambos, o sensor coloca a saída em nível baixo e um LED verde é aceso na placa indicando presença de um obstáculo ou objeto. ([ATHOSELETRONICS... , 2017](#)).

Este periférico requer uma tensão de 5V e uma corrente de 0.20mA e é capaz de identificar objetos de 2 centímetros a 30 centímetros. Este sensor conta com a seguinte

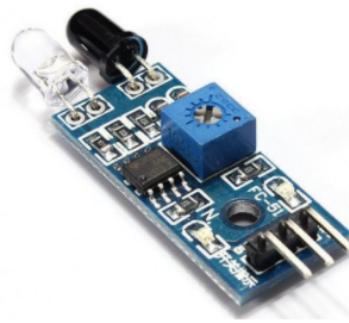


Figura 36 – Módulo Sensor de Obstáculo Infravermelho.

pinagem: Alimentação (VCC), Ground (GND) e OUT. A tabela 22 apresenta as especificações técnicas deste componente ([MASTERWALKER..., 2020](#)).

Parâmetro	Valor
Tensão de Alimentação	3.3V - 5V
Distância de Detecção	2cm - 30cm
Corrente de Operação	0.2mA
Faixa de Detecção	35°
Saída	Digital
Dimensões	37 x 14 x 6 mm

Tabela 22 – Especificações Sensor Infravermelho.

Este sensor foi escolhido levando em consideração o ângulo de detecção para que ocorra o acionamento dos módulos de câmera assim que o fruto estiver na etapa de captura. Além disso, busca a garantia de que a imagem terá o fruto na posição correta para os módulos de câmeras, aumentando a precisão das imagens e posterior classificação. Por fim, o custo e o consumo energético também foram levados em consideração.

#### 4.4.4.6 Integração do módulo com o segmento estrutural

Objetivando um melhor funcionamento dos componentes apresentados nos tópicos acima, foi definido junto à equipe de estrutura que o sistema do seletor de frutas terá um arranjo fechado específico para etapa de captura de imagens. Somado a isso, busca a obtenção de uma imagem nítida e com uma qualidade capaz de fornecer imagens que não possibilitem erros de classificação para o *software* de processamento de imagem.

A distância máxima de detecção do sensor infravermelho é pequena, pois, a partir de um certo valor, a luz infravermelha do ambiente se iguala à luz detectada pelo receptor. Dessa forma, esse sensor é recomendado para aplicações em ambientes mais fechados com pouca iluminação solar. Além de melhorar a atuação do sensor, evita interferências externas à imagem capturada.

Além disso, a velocidade da esteira deve adequar-se ao tempo de processamento e de resposta dos sensores e das câmeras, e o posicionamentos dos periféricos devem seguir as especificações técnicas dos mesmos, como a distância de sensibilidade do sensor e o ângulo de visão da câmera.

#### 4.4.5 Módulo de Definição de Rotas

##### 4.4.5.1 Definição

O módulo de definição de rotas tem como objetivo a detecção, definição e seleção da rota que o fruto será direcionado. Essas rotas são definidas de acordo com a resposta recebida pela etapa de processamento de imagem. As rotas possíveis são: fruto bom sem mancha, fruto bom com mancha e fruto podre. Para a seleção das rotas serão utilizadas duas cancelas automatizadas. As cancelas, por sua vez, serão orientadas por servo-motores.

Este módulo é constituído por dois servo-motores.

##### 4.4.5.2 Esquemático do Módulo de Definição de Rotas

A figura 37 apresenta o diagrama de montagem do módulo de definição de rotas. Sabendo que o servo pode chegar à uma corrente de 900mA quando alimentado com 7V, optou-se por utilizar a alimentação pela fonte de 12V. Os pinos de alimentação e os terras são conectados no regulador de tensão e os pinos de entrada PWM, para controle do giro, são os pinos GPIO 24 e GPIO 25.([SENSOR... , 2020](#))

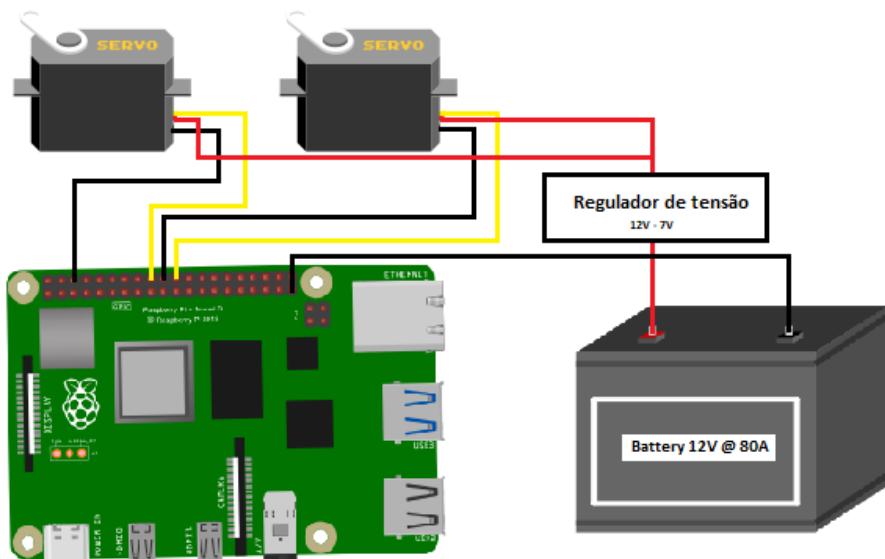


Figura 37 – Esquemático de montagem do módulo de definição de rotas.

#### 4.4.5.3 Diagrama de Controle do Módulo de Definição de Rotas

Para compreensão do controle dos servos-motores foi desenvolvido um diagrama. Este diagrama exemplifica como será realizado a escolha das rotas de cada fruto, após a classificação do *software* de processamento de imagem. Esse diagrama é apresentado na figura 38.

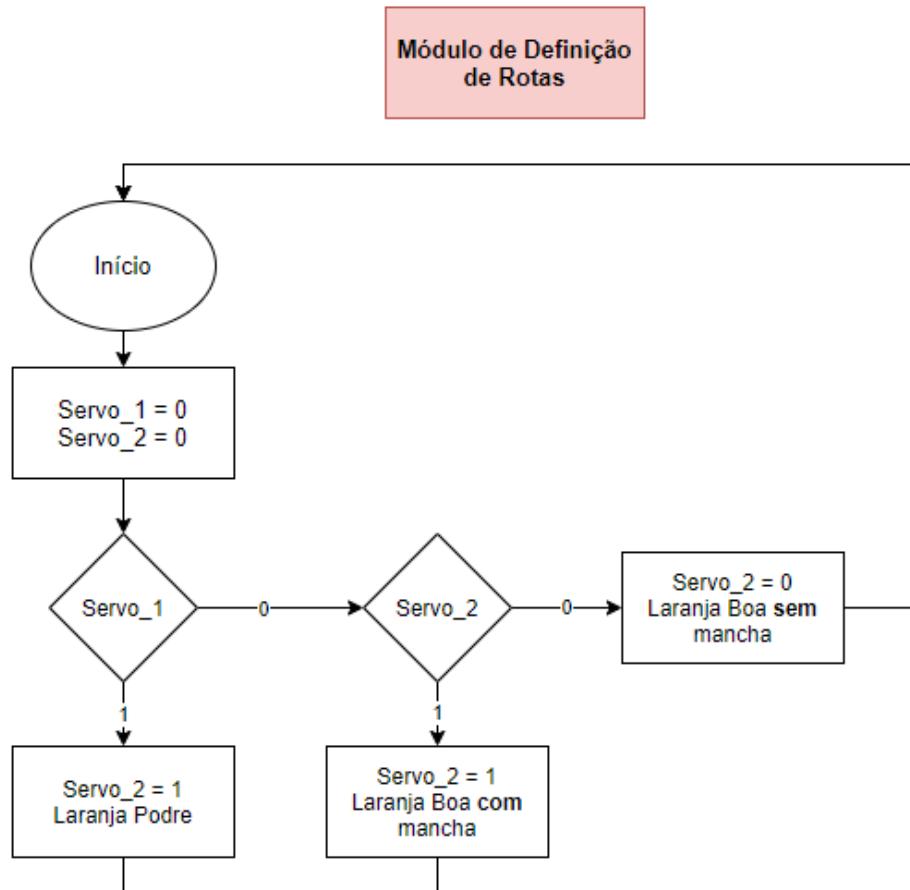


Figura 38 – Diagrama de Controle do Módulo de Definição de Rotas

Inicialmente, o Servo\_1 e Servo\_2 são setados na posição em que as frutas são direcionadas para a rota ideal: laranjas boas sem manchas. Após receber a informação de classificação advinda do processamento de imagem, os frutos são direcionados no decorrer da esteira, de acordo com a tabela 23. A Tabela Verdade demonstra o funcionamento dos servos. Nos casos em que as laranjas são classificadas como podres, recebe-se dois sinais de nível lógico alto nos servos, ou nível lógico alto no Servo\_1 e nível lógico baixo no Servo\_2. Esta última, é uma condição de segurança para que laranjas podres nunca percorram a rota de laranjas boas sem manchas. Nos casos em que as laranjas são boas, mas apresentam manchas em sua superfície, o Servo\_1 recebe sinal de nível lógico baixo e o Servo\_2 recebe sinal de nível lógico alto. Por fim, se as laranjas forem classificadas como boas e sem manchas, a condição inicial se mantém e as laranjas seguem o fluxo ideal.

Este processo é contínuo e ocorre de forma exclusiva para cada fruto classificado. Dessa forma, cada laranja possui uma rota única para sua classificação, garantindo a qualidade do processo.

Servo_1	Servo_2	Classificação
0	0	Boa sem manchas
0	1	Boa com manchas
1	0	Podre
1	1	Podre

Tabela 23 – Tabela Verdade de Classificações.

#### 4.4.5.4 Servo-Motor

Serão utilizados dois servos-motores MG996R, com estrutura e engrenagem de metal. O servo escolhido possui movimento suave, velocidade de reação rápida e pode rotacionar até 180°. O principal diferencial desse servo motor para outros servos está na precisão da rotação e na sua capacidade de suportar estruturas mais robustas. ([SERVO... , 2019](#))

O dimensionamento de componente para separação das laranjas foi realizado a partir da necessidade de torque resistivo à força resultante gerada pelo movimento das laranjas na esteira. Os cálculos para seleção do componente podem ser encontrados no tópico 2.2.3.2.

Este servo requer uma alimentação mínima de 4.8V e corrente de 500mA. O servo possui 3 fios – vermelho, marrom/preto e amarelo – sendo eles alimentação (VCC), ground (GND) e Sinal PWM (*Pulse Width Modulation*), respectivamente. É um servo analógico. De acordo com as especificações e o seu custo acessível este servo foi escolhido para ser acoplado à estrutura da cancela e realizar o movimento da cancela de acordo com os sinais recebidos, direcionando o fruto pela esteira. A tabela abaixo apresenta as especificações técnicas deste componente. ([DATASHEET... , 2019](#))

Parâmetro	Valor
Tensão de alimentação	4.8V - 7.2V
Corrente de Operação	500mA - 900mA
Torque	9.4kgf.cm (4.8V) e 11kgf.cm (6V)
Velocidade	0.17s/60° (5V) e 0.14s/60° (7V)
Peso	55g
Dimensões	55x20x45mm

Tabela 24 – Especificações do Servo-motor MG996R.

#### 4.4.5.5 Integração do módulo com o segmento estrutural

A relação do módulo de definição de rotas com a parte estrutural do projeto se dá na velocidade da esteira, que deve ser adequada ao tempo de processamento e de resposta dos servos. Além disso, o posicionamento dos servos na estrutura definida é de extrema importância para programar um grau de giro correto. Dessa forma, não haverá frutos percorrendo rotas erradas e a classificação não será prejudicada.

### 4.4.6 Módulo de Contagem

#### 4.4.6.0.1 Definição

O módulo de contagem é responsável pela contagem dos frutos selecionados após a classificação. Os frutos que serão contados são os frutos bons sem mancha, separados em pequenos, médios e grandes. Para realizar a contagem dos frutos, faz-se necessário a utilização de um sensor capaz de identificar a passagem do fruto. Para isso, serão utilizados três detectores de presença.

Este módulo é constituído por três sensores ultrassônicos.

#### 4.4.6.1 Esquemático do Módulo de Contagem

Os pinos digitais da Raspberry trabalham com níveis lógicos de 0 a 3.3V, por isso, não podemos conectar o pino ECHO a um pino de entrada da Raspberry, pois o pino ECHO trabalha a 5V quando é acionado. Para resolver esse problema, utilizou-se um divisor de tensão.

A figura 39 apresenta o esquemático de montagem do módulo de contagem. Os sensores são alimentados pelo pino de 5V, os terras são conectados no pino GND, os pinos de TRIG são conectados nos pinos GPIO 5, GPIO 6 e GPIO 26, os pinos de ECHO são conectados nos divisores de tensão e os divisores estão conectados nos pinos GPIO 17, GPIO 22 e GPIO 27.

#### 4.4.6.2 Diagrama de Controle do Módulo de Contagem

O diagrama de blocos da figura 40 descreve o controle do módulo. Inicialmente, as variáveis de contagem - laranja\_pequena, laranja\_media e laranja\_grande - são inicializadas com nível lógico baixo e as variáveis de distância - distancia\_pequena, distancia\_media e distancia\_alta - recebem os valores de distância lidos pelos sensores ultrassônicos.

Se a distancia\_pequena for igual ou menor que 50cm, a variável laranja\_pequena armazena o seu valor atual mais 1, atualiza a contagem no *software* e volta para atualizar a leitura das variáveis de distância, se não, é feito o segundo teste.

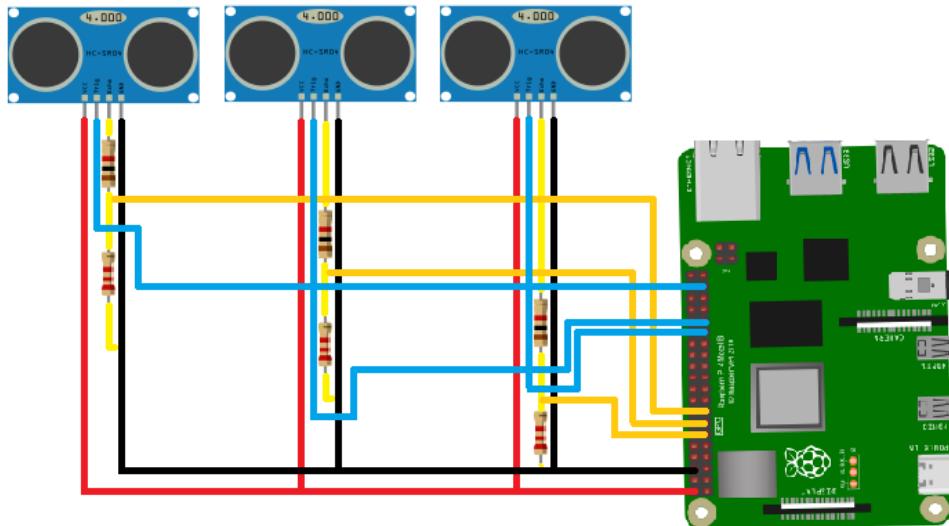


Figura 39 – Esquemático de montagem do módulo de contagem.

Se a `distancia_media` for igual ou menor que 50cm, a variável `laranja_media` armazena o valor atual dela mais 1, atualiza a contagem no *software* e volta para atualizar a leitura das variáveis de distância, se não, é feito o terceiro teste.

Se a `distancia_grande` for igual ou menor que 50cm, a variável `laranja_grande` armazena o valor atual dela mais 1, atualiza a contagem no *software* e volta para atualizar a leitura das variáveis de distância, se não, significa que nenhuma laranja foi detectada e volta para atualizar a leitura das distâncias.

A verificação deve ser feita de forma precisa, portanto, essa rotina é contínua. Assim, as verificações serão sempre feitas, de modo que todos os frutos que passam pela classificação de tamanho sejam contabilizados e apresentados no *software* de controle.

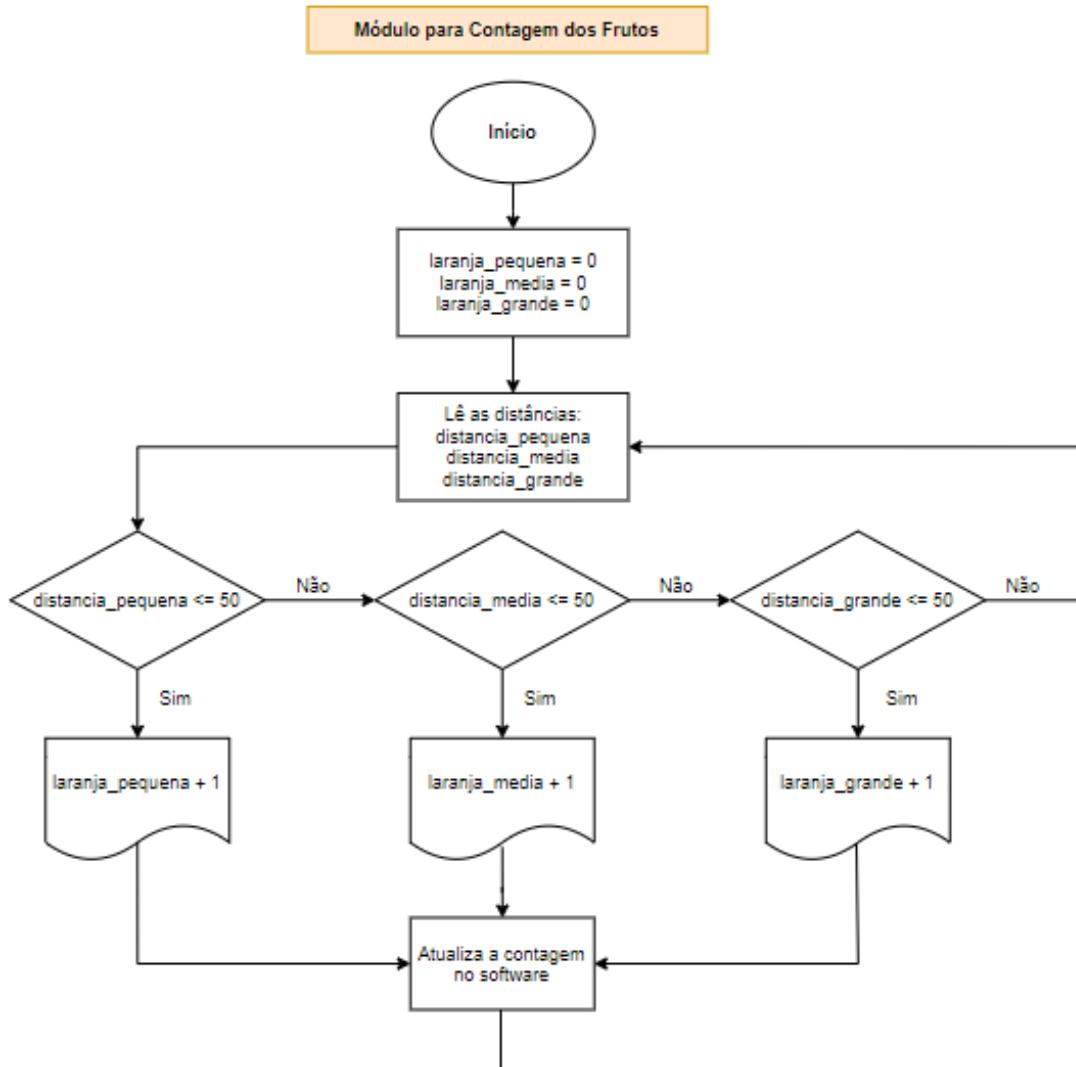


Figura 40 – Diagrama de Controle do Módulo de Contagem

#### 4.4.6.3 Sensor HC-SR04

O modelo escolhido para implementação foi o sensor ultrassônico HC-SR04. Estes sensores são dispositivos que utilizam ondas sonoras para detectar obstáculos e medir distâncias. As ondas sonoras são ondas mecânicas que se propagam em diferentes materiais em forma de energia e utilizam frequência acima da faixa audível do ser humano de forma a evitar possíveis interferências. A velocidade de propagação do som difere de acordo com o meio de propagação, sendo a velocidade de propagação no ar de aproximadamente 340m/s ([MORAIS DOUGLAS; RODRIGUES, 2017](#)).

O funcionamento deste sensor ocorre por dois elementos principais, um transmissor e um receptor de som, posicionados lado a lado. Quando deseja-se fazer uma medida, o transmissor é ativado e produz uma série de pulsos sonoros. Esses pulsos viajam pelo espaço entre o sonar e um obstáculo (no projeto, o fruto a ser contabilizado) na velocidade do som. Quando as ondas sonoras atingem o obstáculo, parte delas sofrem reflexão. Essas ondas refletidas são então captadas pelo receptor.

O sensor HC-SR04 (Figura 41) possui 4 pinos: VCC, GND, TRIG e ECHO ([DATASHEET..., 2020](#)). O TRIG é o pino de ativação do transmissor: quando esse pino recebe um pulso de  $10\mu s$ , o transmissor do módulo produz uma série de pulsos ultrassônicos de 40kHz. O intervalo em que ECHO está em nível lógico alto é o tempo de ida e volta dos pulsos ultrassônicos. A diferença de tempo entre a transmissão dos pulsos sonoros e a captação pelo receptor é usada para calcular a distância até o objeto. A distância é então calculada por:

$$d = \frac{t_{echo} * V_{som}}{2} \quad (4.11)$$

Em que:

- d = distância [metro]
- techo = tempo em que o pulso ficou em nível lógico alto [segundos]
- Vsom = velocidade do som [metro/segundo]

O resultado é dividido por 2 porque deve-se levar em consideração o tempo de transmissão e de reflexão.

#### 4.4.6.4 Integração do módulo com o segmento estrutural

A relação do módulo de contagem com a parte estrutural do projeto se dá no posicionamento dos sensores ultrassônicos e nos diâmetros de classificação de tamanho dos frutos analisados.



Figura 41 – Sensor ultrassônico HC-SR04.

Parâmetro	Valor
Tensão de alimentação	5V
Distância de detecção	2 cm à 40 cm
Corrente de operação	0.15mA
Faixa de detecção	15°
Peso	5g

Tabela 25 – Especificações do Sensor ultrassônico HC-SR04.

Neste módulo, o conhecimento das distâncias de cada sensor para o fruto que será detectado pelo mesmo é de extrema importância, pois este dado irá permitir a programação do periférico para que a solução tenha a identificação precisa de cada fruto e o dado informado para o usuário seja correto.

#### 4.4.7 Módulo de Disponibilização de Dados

##### 4.4.7.1 Definição

Trata-se do módulo responsável pela visualização dos dados obtidos, a partir do resultado da classificação dos frutos. Para fornecimento das informações ao usuário será acoplado um monitor à estrutura do projeto.

##### 4.4.7.2 Display

Para possibilitar a visualização de dados a respeito da classificação das laranjas analisadas, o seletor engloba um *display*. Este periférico permitirá que o usuário acompanhe em tempo real gráficos quantitativos sobre os parâmetros de qualidade e tamanho. Dessa forma, o modelo escolhido para implementação foi um *display* de alta resolução com 10,1 polegadas compatível com a versão de Raspberry PI escolhida previamente. ([SUNFOUNDER... , 2020](#))

Este painel possui tecnologia IPS-LCD. Com isso, o tempo de atualização e a reprodução de cor cumprem os requisitos necessários para uma boa visualização das imagens. Além disso, possui entrada VGA e HDMI. Seu ângulo visual é de 178°, o que garante qualidade da imagem visualizada em qualquer ângulo.([SUNFOUNDER... , 2020](#))

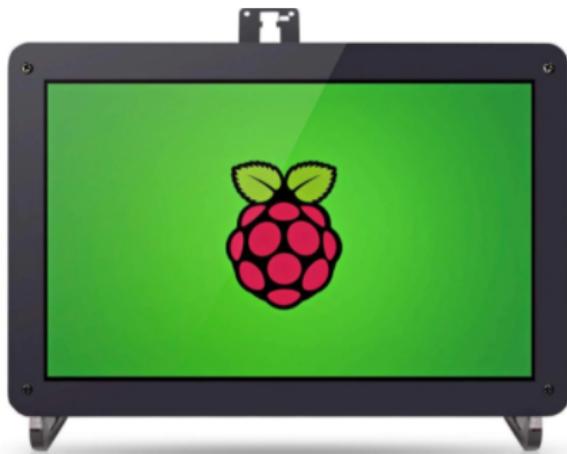


Figura 42 – Visão Frontal Display.

Parâmetro	Valor
Resolução	1280x800
Ângulo Visual	178°
Tamanho	218mm x 137mm
Proporção de Tela	16:10
Tensão de Alimentação	12 V
Corrente de Operação	3 A

Tabela 26 – Especificações do Display.

O usuário poderá visualizar os dados de forma gráfica e quantitativa referentes ao processamento de classificação de laranjas. Será possível visualizar os gráficos por qualidade e tamanho das laranjas que estão sendo analisadas, assim como suas quantidades em forma de tabela.

#### 4.4.7.3 Integração do módulo com o segmento estrutural

A relação do módulo de disponibilização de dados com a parte estrutural do projeto se dá no posicionamento do display na estrutura final do projeto. Devendo estar posicionado em um local adequado para melhor visualização e conforto do usuário.

## 4.4.8 Periférico Adicional

### 4.4.8.1 HD Externo

A Raspberry PI 4, por padrão, trabalha com SD Card - *Secure Digital Card* - que são pequenos cartões de memória não voláteis. Estes cartões de memória possuem vida útil com base em ciclos de gravação.

O *software* para classificação de imagens possui um banco de dados para garantir a seleção correta de cada fruto. Dessa forma, a vida útil do cartão é reduzida, já que para cada laranja acontecerá os seguintes processos:

- 1 - Salvamento da imagem capturada pela câmera em disco de armazenamento;
- 2 - Resultado da classificação é salvo no banco de dados, que por sua vez está armazenado no disco de armazenamento;
- 3 - Após a classificação ser salva, a imagem capturada é apagada do disco de armazenamento.

Outro agravante é que, quando o usuário possui internet e a sincronização entre o banco de dados local e o na nuvem ocorre, os dados do banco de dados local são apagados, evitando armazenamento desnecessário. Caso o usuário nunca tenha acesso à internet no local de produção, a limpeza do banco de dados também acontece, só que com base nas colheitas mais antigas.

Dessa forma, o *SD Card* não é uma solução sustentável devido sua vida útil ser reduzida quando lida com ciclos de gravação e deleção muito intensa. Para evitar possíveis falhas, será adicionado um HD Externo de 120 GB (Figura 43). A Tabela 27 apresenta as especificações técnicas do modelo escolhido. ([MERCADO..., 2020](#))



Figura 43 – HD Externo.

Parâmetro	Valor
Interface	USB 3.0
Armazenamento	HDD
Capacidade	120 GB
Dimensões	12,5 x 7,5 x 1,3 cm

Tabela 27 – Especificações do HD Externo.

#### 4.4.9 Comunicação Serial

O protocolo de comunicação escolhido para transferência de informações entre a Raspberry Pi e os periféricos foi a comunicação serial. A comunicação é efetuada entre dois dispositivos ligando os respectivos sinais de entrada e saída, isso permite que o sinal transmitido por um dispositivo chegue ao receptor do outro, possibilitando a comunicação.

As direções de transferência dos dados serão *simplex* e *full-duplex*. No primeiro modo de ligação, cada elemento é caracterizado como transmissor ou receptor. Dessa forma, um sempre transmite e outro sempre recebe. Já no modo *full-duplex* os dois sistemas são capazes de transmitir e de receber, porém as duas operações não podem ser realizadas simultaneamente.

Deste modo, a comunicação serial é a mais recomendada, pois não utiliza as portas de I/O do microcontrolador e o próprio USB o alimenta, diminuindo a quantidade de fontes para o sistema. Mesmo a Raspberry Pi permitindo a comunicação por *WiFi*, essa comunicação é mais rápida e mais segura.

#### 4.4.10 Diagramas

##### 4.4.10.1 Diagrama de Processos

O diagrama abaixo tem como objetivo permitir a visualização do fluxo de atividades que ocorrem durante a utilização do seletor de laranjas.

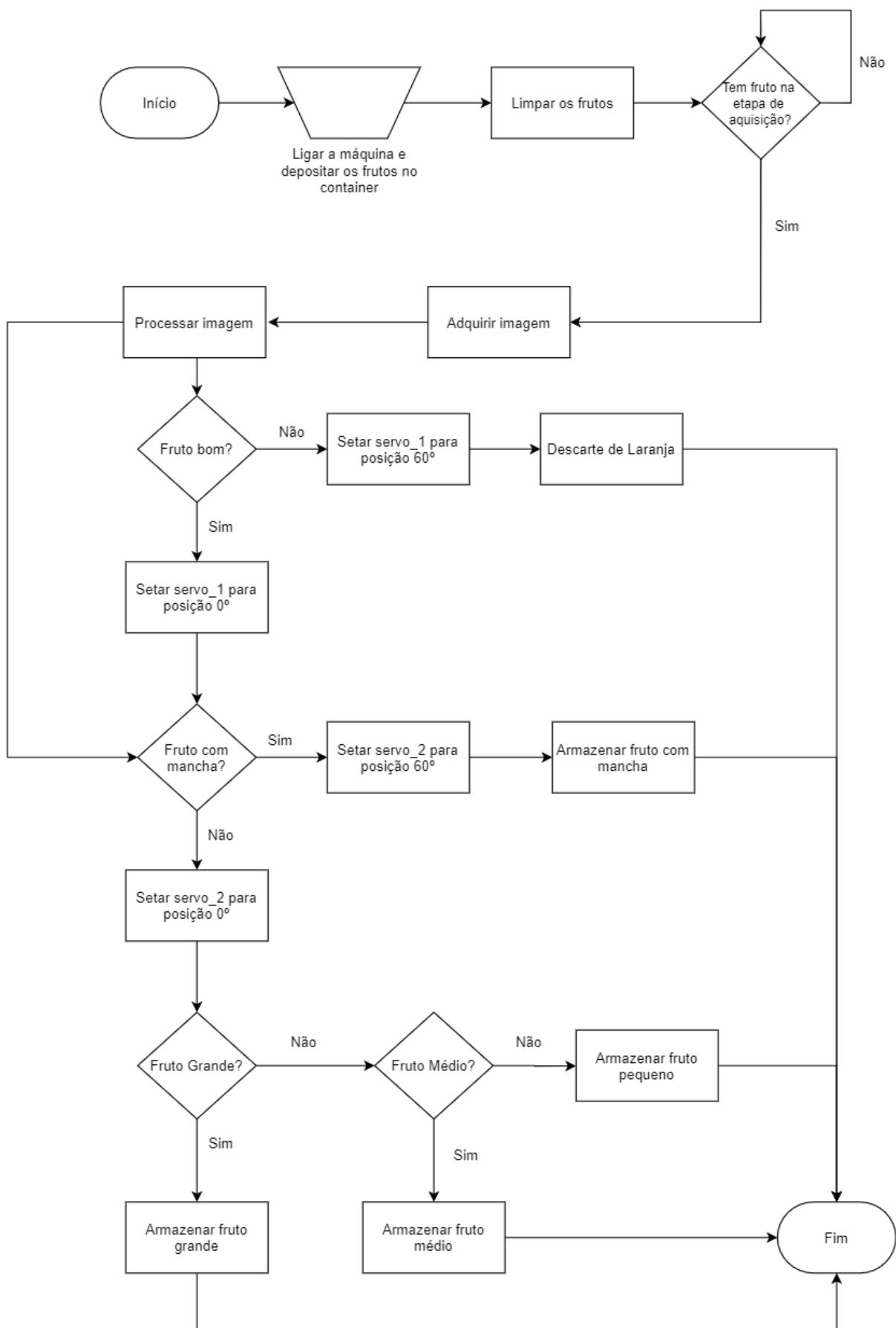


Figura 44 – Diagrama de Processos Seletor de Laranjas

#### 4.4.10.2 Diagramas de Atuação

Os diagramas abaixo visam ilustrar o comportamento da solução a partir da variação do tipo de laranja que será classificada. Sendo a imagem 45 para frutos podres, imagem 46 para frutos bom e sem manchas e, por fim, a imagem 47 para frutos bom e com manchas.

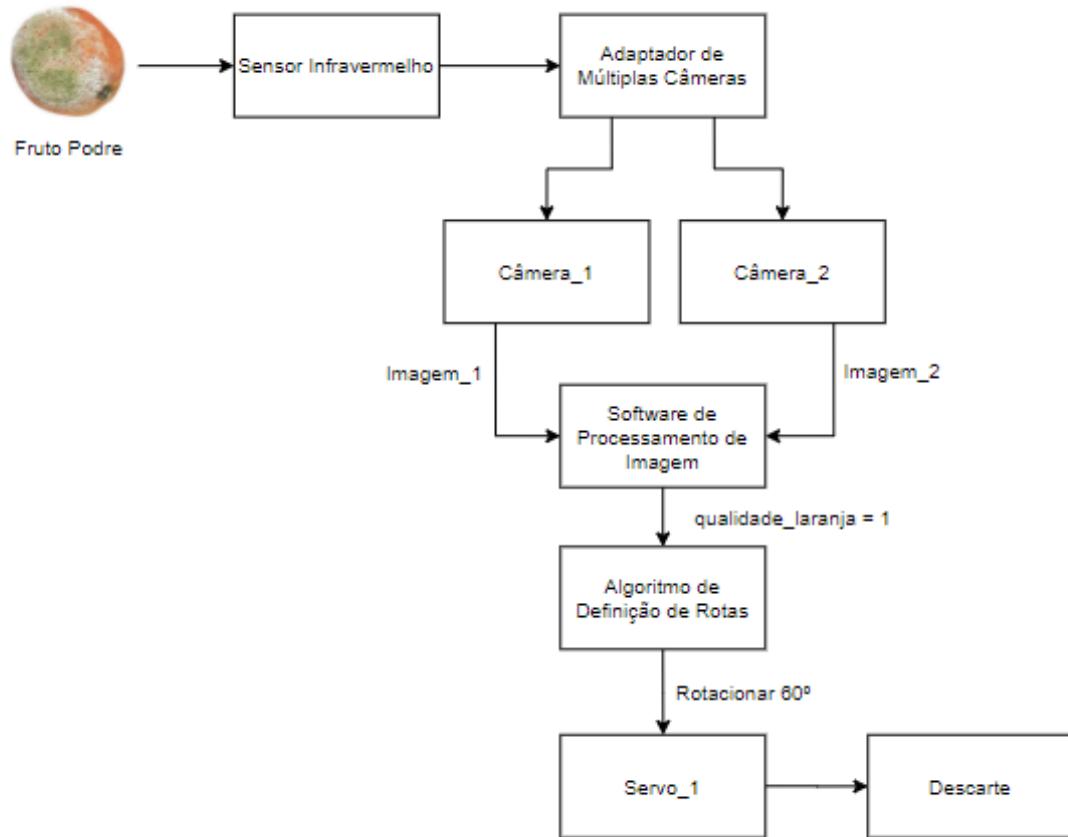


Figura 45 – Diagrama de Atuação: Fruto Podre

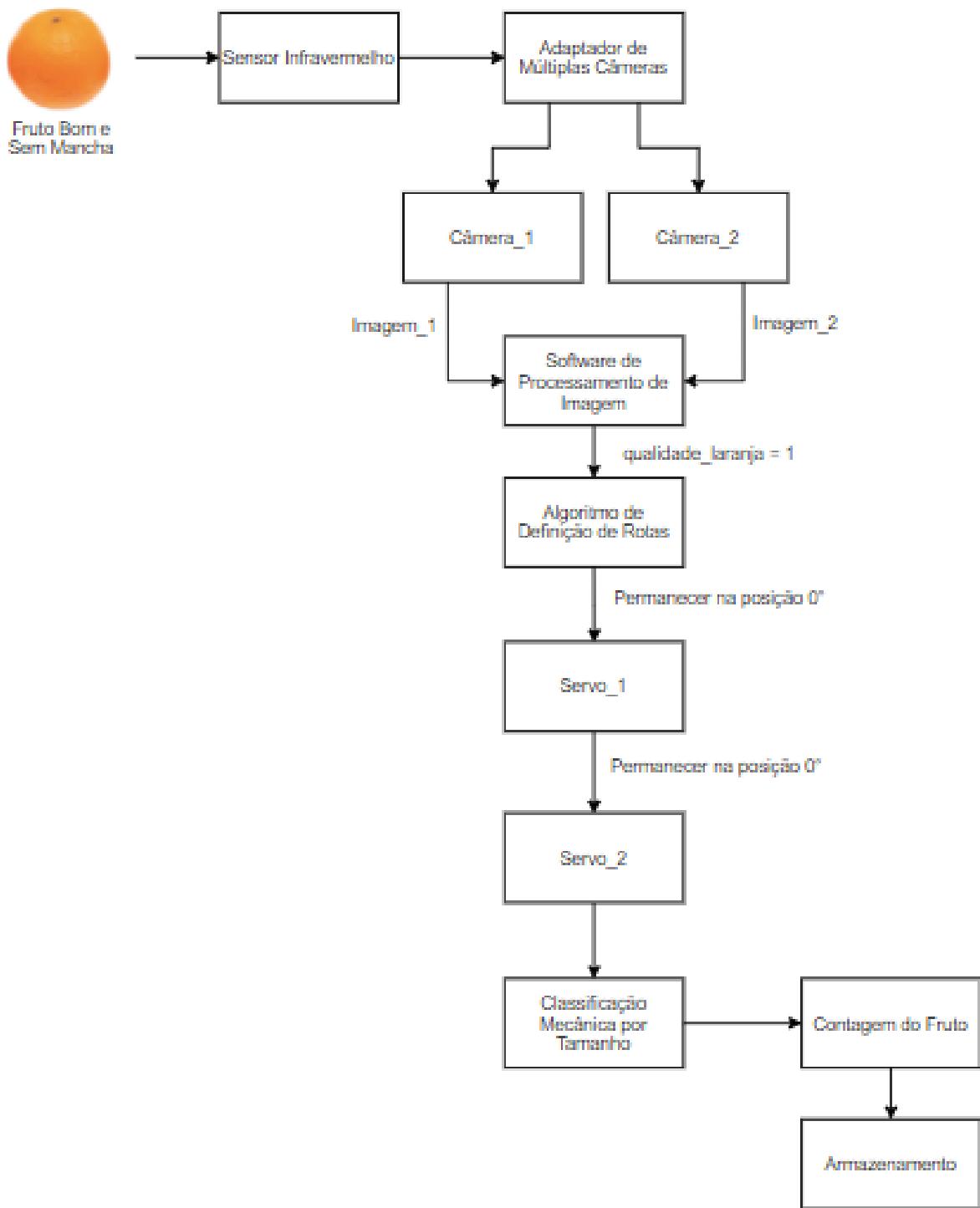


Figura 46 – Diagrama de Atuação: Fruto Bom Sem Mancha

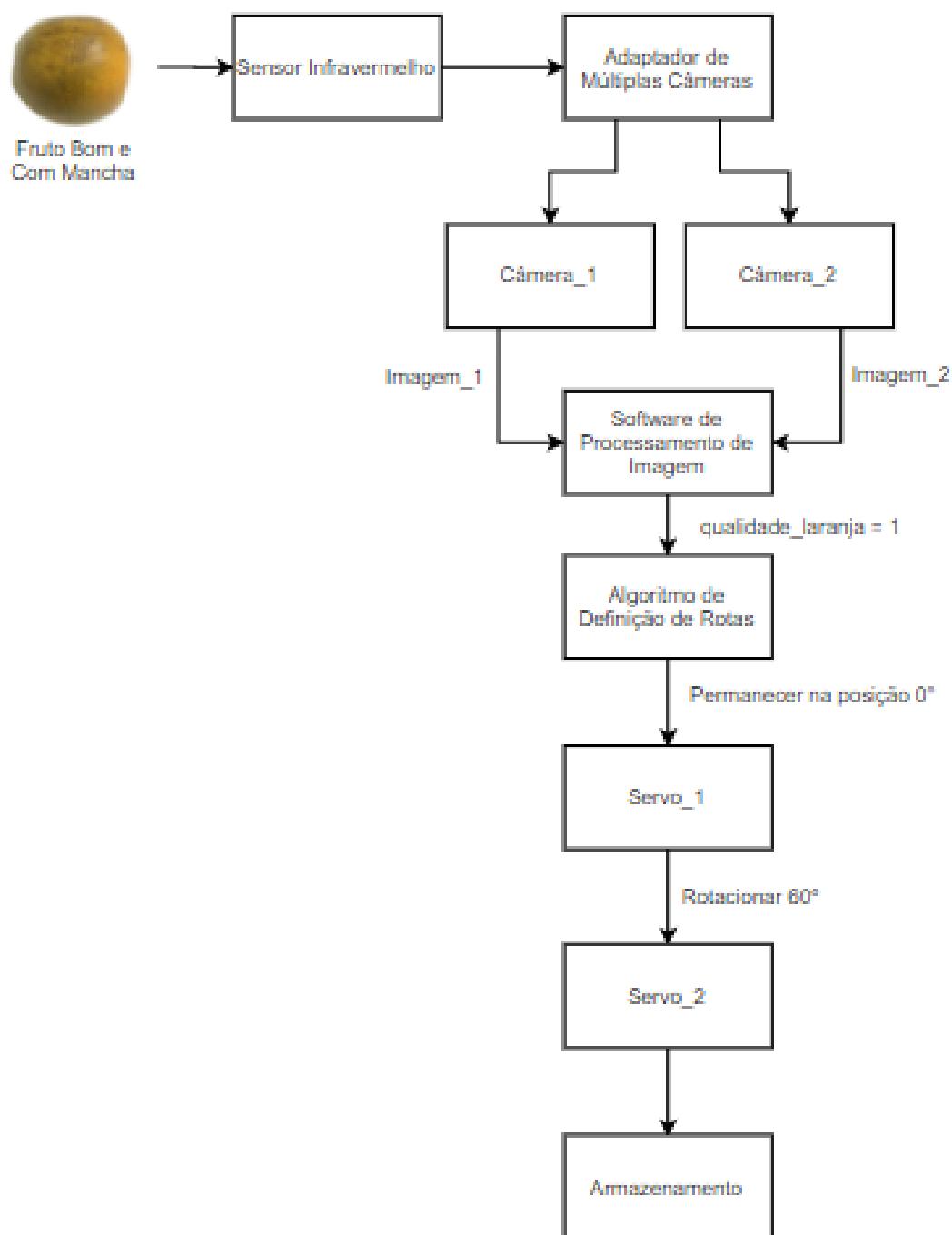


Figura 47 – Diagrama de Atuação: Fruto Bom Com Mancha

### 4.4.11 Algoritmos

Para controle dos periféricos e funcionamento da solução, foram desenvolvidos códigos em *Python 3*.

#### 4.4.11.1 Algoritmo de Aquisição de Imagem

Inicialmente, foram importadas as bibliotecas necessárias para compilar o código: RPi.GPIO, Picamera, time. Depois, criou-se uma função em que a partir da identificação de nível lógico baixo na variável "sensor", as câmeras fossem acionadas e posteriormente, as imagens capturadas e armazenadas. Na sequência, as imagens são enviadas para o banco de dados para realização do processamento e classificação pelo *software*.

Na figura 48 é possível visualizar o resultado da simulação quando a variável "sensor" é setada para 1. Como saída, obteve-se o *print* "Fruto Não Detectado", indicativo de que não há frutos na etapa de aquisição. Portanto, as câmeras não são acionadas e permanecem em espera.

Na figura 49, referente à segunda simulação, a variável "sensor" foi setada para 0, indicando a presença de fruto na etapa de aquisição. Dessa forma, as câmeras são ativadas e as imagens capturadas.

```

11 GPIO.setwarnings(False)
12
13 #Conexão física do pino
14 sensor_ir = 23
15
16 #Configurando o pino como entrada
17 GPIO.setup(sensor_ir, GPIO.IN)
18
19 #Configurando as câmeras
20 camera = PiCamera()
21 camera.start_preview()
22
23 while True:
24     if GPIO.input(sensor_ir)==1:
25         #Não há fruto na etapa de aquisição de imagem
26         time.sleep(0.1)
27         print("Sensor IR: 1")
28         print("Fruto Não Detectado.")
29
30     elif GPIO.input(sensor_ir) == 0:
31         #Detecção de fruto na etapa de aquisição de imagem
32
33         #Ativação dos módulos de câmera para aquisição de imagem
34         #Iniciar câmera A
35         i2c = 'i2cset -y 1 0x70 0x00 0x04'
36         os.system(i2c)
37         capture(1)
38
39         #Iniciar câmera B
40         i2c_ = 'i2cset -y 1 0x70 0x00 0x05'
41         os.system(i2c_)
42         capture(2)
43
44         print("Sensor IR:", sensor_ir)
45         print("Fruto Detectado. Ativação câmera A. Ativação câmera B.")
46
47 #Função de Captura de Imagens
48 def capture(cam):
49     image_1 = "raspistill -o image_Xd.jpg" % cam
50     image_2 = "raspistill -o image_Xd.jpg" % cam
51     os.system(image_1, image_2)
52     print("Imagens capturadas.")
53
54 #Simulação
55 GPIO.input(sensor_ir) = 1

```

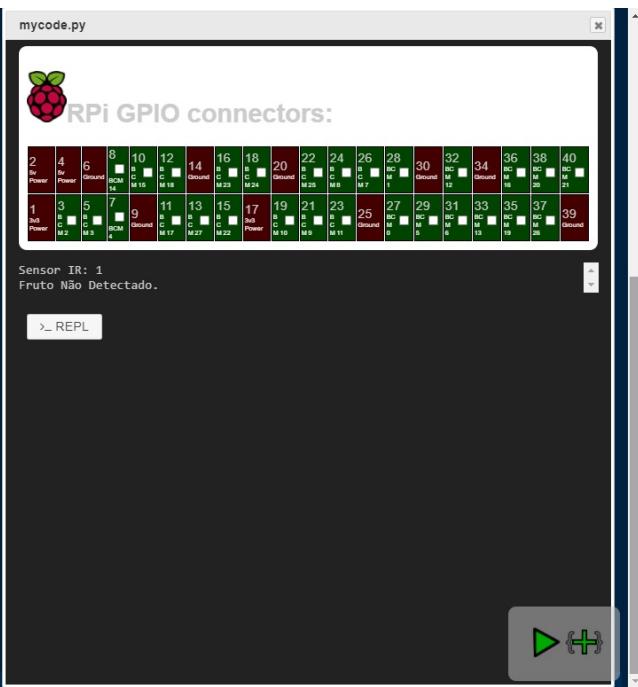


Figura 48 – Algoritmo de Aquisição de Imagem: Simulação 01.

The image shows a terminal window titled "mycode.py" containing Python code for a fruit detection system. The code uses GPIO pins for an infrared sensor and PiCameras for two cameras. It checks the sensor input and performs actions based on the value (1 or 0). It also initializes cameras, sets I2C parameters, and captures images from both cameras. A message is printed indicating the status of the sensor and the cameras.

```

11 GPIO.setwarnings(False)
12
13 #Conexão física do pino
14 sensor_ir = 23
15
16 #Configurando o pino como entrada
17 GPIO.setup(sensor_ir, GPIO.IN)
18
19 #Configurando as câmeras
20 camera = PiCamera()
21 camera.start_preview()
22
23 while True:
24     if GPIO.input(sensor_ir)==1:
25         #Não há fruto na etapa de aquisição de imagem
26         time.sleep(0.1)
27         print("Sensor IR: 1")
28         print("Fruto Não Detectado.")
29
30     elif GPIO.input(sensor_ir) == 0:
31         #Detecção de fruto na etapa de aquisição de imagem
32
33         #Ativação dos módulos de câmera para aquisição de imagem
34         #Inicializar câmera A
35         i2c = 'i2csel -y 1 0x70 0x00 0x04'
36         os.system(i2c)
37         capture(1)
38
39         #Inicializar câmera B
40         i2c = 'i2csel -y 1 0x70 0x00 0x05'
41         os.system(i2c)
42         capture(2)
43
44         print("Sensor IR:", sensor_ir)
45         print("Fruto Detectado. Ativação câmera A. Ativação câmera B.")
46
47 #Função de Captura de Imagens
48 def capture(cam):
49     image_1 = "raspistill -o image_Xd.jpg" % cam
50     image_2 = "raspistill -o image_Xd.jpg" % cam
51     os.system(image_1, image_2)
52     print("imagens capturadas.")
53
54 #Simulação
55 GPIO.input(sensor_ir) = 0

```

Figura 49 – Algoritmo de Aquisição de Imagem: Simulação 02.

#### 4.4.11.2 Algoritmo de Definição de Rotas

Para simular o algoritmo de controle dos servo-motores, foi definida duas variáveis que correspondem às informações resultantes do processamento de imagens. Estas variáveis são qualidade\_laranja e mancha\_laranja. A primeira é responsável por informar se a laranja é boa (0) ou ruim (1). Já a segunda informa se não há manchas (0) ou se há manchas (1) na superfície do fruto.

Dessa forma, estas variáveis são as condições para acionar os servo-motores que irão rotacionar as cancelas responsáveis pela seleção das laranjas. As imagens abaixo apresentam os resultados das simulações a partir da alteração das variáveis mencionadas.



Figura 50 – Algoritmo de Definição de Rotas: Simulação 01 - Laranja Podre.

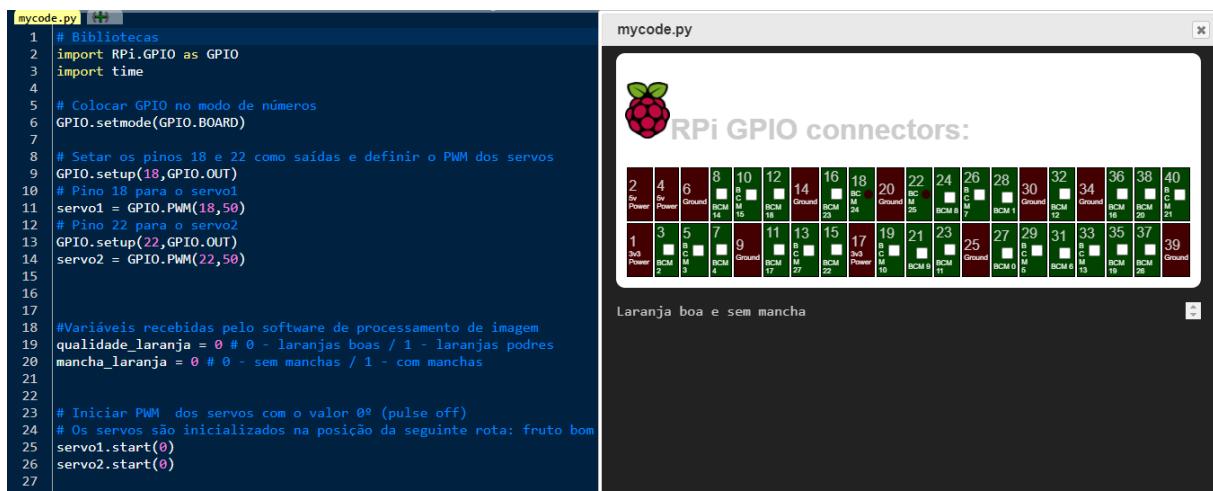


Figura 51 – Algoritmo de Definição de Rotas: Simulação 02 - Laranja Boa e Sem Mancha.

```

mycode.py
1 # Bibliotecas
2 import RPi.GPIO as GPIO
3 import time
4
5 # Colocar GPIO no modo de números
6 GPIO.setmode(GPIO.BCM)
7
8 # Setar os pinos 18 e 22 como saídas e definir o PWM dos servos
9 GPIO.setup(18,GPIO.OUT)
10 # Pino 18 para o servo1
11 servo1 = GPIO.PWM(18,50)
12 # Pino 22 para o servo2
13 GPIO.setup(22,GPIO.OUT)
14 servo2 = GPIO.PWM(22,50)
15
16
17
18 #Variáveis recebidas pelo software de processamento de imagem
19 qualidade_laranja = 0 # 0 - laranjas boas / 1 - laranjas podres
20 mancha_laranja = 1 # 0 - sem manchas / 1 - com manchas
21
22
23 # Iniciar PWM dos servos com o valor 0° (pulse off)
24 # Os servos são inicializados na posição da seguinte rota: fruto bom
25 servo1.start(0)
26 servo2.start(0)
27

```

RPi GPIO connectors:

Figura 52 – Algoritmo de Definição de Rotas: Simulação 03 - Laranja Boa e Com Mancha.

#### 4.4.11.3 Algoritmo de Contagem

O algoritmo do módulo de contagem foi implementado em *Python 3* e o código com as descrições encontra-se no apêndice.

Para simular o comportamento do algoritmo de contagem, utilizaram-se os *RPi GPIO connectors* mostrados nas figuras abaixo para representar as entradas na Raspberry Pi 4 Model B.

Quando colocamos PIN\_ECHO1 (pino 11 - laranjas pequenas), PIN\_ECHO2 (pino 13 - laranjas médias) ou PIN\_ECHO3 (pino 15 - laranjas grandes) em nível lógico alto, a variável de início de pulso é atualizada com o tempo atual, quando voltamos para nível lógico baixo, o tempo de fim de pulso é atualizado com o tempo atual e a distância é calculada com a duração do pulso.

Na primeira simulação (figura 53) o objetivo é mostrar as situações possíveis, que são atualizar as variáveis de contagem de tamanho quando a laranja passar pelo sensor dentro da distância permitida (50 centímetros) e não contabilizar objetos que possam aparecer fora da distância permitida.

Inicialmente, setamos o pino 11 (representando uma laranja pequena) por um curto período de tempo, depois setamos o pino 13 (representando uma laranja média) e depois o pino 15 (representando uma laranja grande) como as distâncias foram menores do que 50 centímetros, o resultado apresentado foi contabilizar uma laranja de cada tamanho. Em seguida, setamos os pinos por um período de tempo maior, como as distâncias foram maiores, as variáveis de contagem não foram atualizadas.

```

1 # Código para contagem das laranjas boas sem manchas classificadas
2
3 #Importando as bibliotecas necessárias
4 import RPi.GPIO as GPIO #Biblioteca que possibilita controlar e interagir com os pinos
5 import time #Biblioteca que permite controlar funções que utilizam o tempo
6
7 #Variáveis para armazenar o valor da contagem
8 laranja_pequena = 0
9 laranja_media = 0
10 laranja_grande = 0
11
12 #Variáveis para armazenar o valor da distância
13 distancia_pequena = 0
14 distancia_media = 0
15 distancia_grande = 0
16
17 #GPIOBOARD configura os pinos na forma de contagem física.
18 GPIO.setmode(GPIO.BCM)
19
20 #Conexão física dos pinos
21 PIN_TRIGGER1 = 29 #Sensor laranja pequena
22 PIN_ECHO1 = 11 #Sensor laranja pequena
23 PIN_TRIGGER2 = 31 #Sensor laranja média
24 PIN_ECHO2 = 13 #Sensor laranja média
25 PIN_TRIGGER3 = 37 #Sensor laranja grande
26 PIN_ECHO3 = 15 #Sensor laranja grande
27
28 #Configurando os pinos como saída e entrada
29 GPIO.setup(PIN_TRIGGER1, GPIO.OUT) #Sensor laranja pequena
30 GPIO.setup(PIN_ECHO1, GPIO.IN) #Sensor laranja pequena
31 GPIO.setup(PIN_TRIGGER2, GPIO.OUT) #Sensor laranja média
32 GPIO.setup(PIN_ECHO2, GPIO.IN) #Sensor laranja média
33 GPIO.setup(PIN_TRIGGER3, GPIO.OUT) #Sensor laranja grande
34 GPIO.setup(PIN_ECHO3, GPIO.IN) #Sensor laranja grande
35
36 while True:
37
38     #Variáveis para guardar o tempo de inicio do pulso e fim do pulso
39     tempo_inicio_pulso1 = time.time()
40     tempo_inicio_pulso2 = time.time()

```

Figura 53 – Algoritmo de Contagem das laranjas boas sem manchas: Simulação 01.

Na segunda simulação (figura 54) o objetivo é mostrar que não precisa seguir uma ordem para apresentar a contagem, sempre que a distância for menor ou igual a 50 o algoritmo é capaz de contabilizar a laranja e enviar o resultado para o *software*.

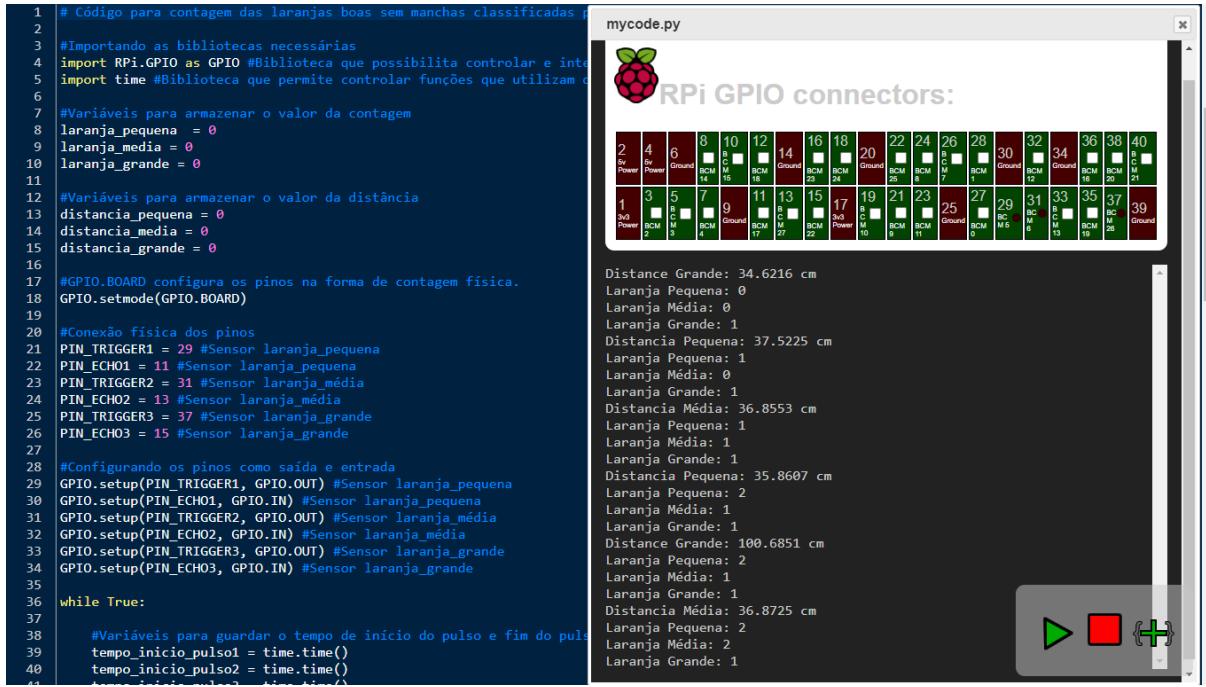


Figura 54 – Algoritmo de Contagem das laranjas boas sem manchas: Simulação 02.

Na terceira simulação (figura 55) o objetivo é mostrar que enquanto a máquina estiver ligada, é possível ter a atualização das variáveis de contagem e o envio dos dados.

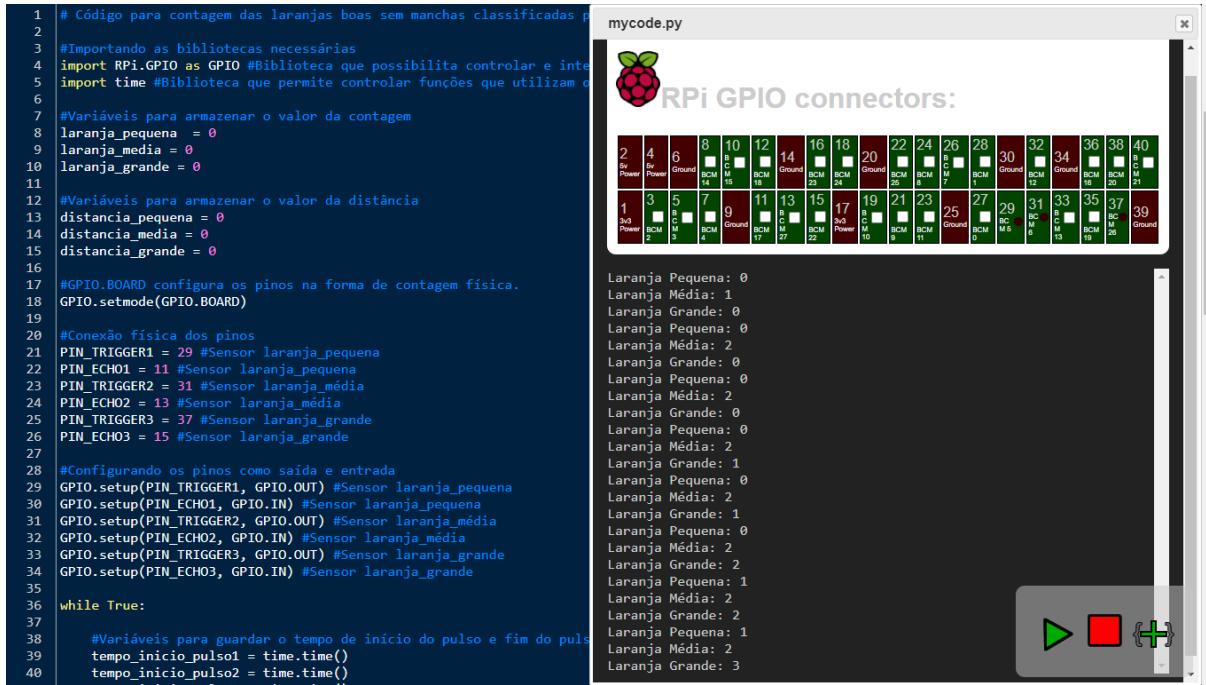


Figura 55 – Algoritmo de Contagem das laranjas boas sem manchas: Simulação 03.

## 4.5 Solução Geral de Software

Com o objetivo de detalhar melhor a solução geral de *software* foi dividida em quatro módulos, os quais são: Arquitetura da Informação, Arquitetura de Software, Diagramas e Inovação.

### 4.5.1 Arquitetura da Informação

A arquitetura da informação é importante para projetos de *software*, principalmente porque ela dá clareza às funcionalidades do projeto (A..., 2020). Além disso, o arquiteto da informação precisa abstrair os requisitos do cliente para materializar os seus desejos quanto ao projeto.

Com base nisso, o grupo fez os protótipos das telas da Aplicação Web e do Desktop APP a fim de organizar as informações que serão mostradas aos usuários. Para realizar estes designs foram utilizadas a ferramenta InVision para o protótipo da Aplicação Web e o Figma para o protótipo do Desktop APP.

#### 4.5.1.1 Aplicação Web

##### 4.5.1.1.1 Protótipo

A Figura 56 representa o design da tela de registro, onde o usuário pode entrar com os seus dados de cadastro, como nome, e-mail, senha e o identificador de sua máquina.

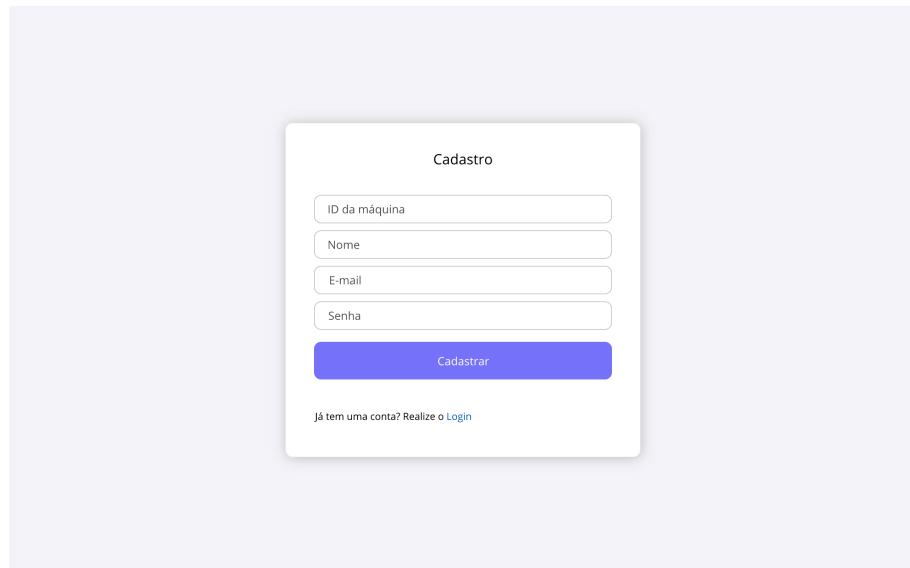


Figura 56 – Tela de cadastro.

A Figura 57 representa o design da tela de login, onde o usuário pode entrar com os seus dados para entrar na aplicação, inserindo e-mail e senha.

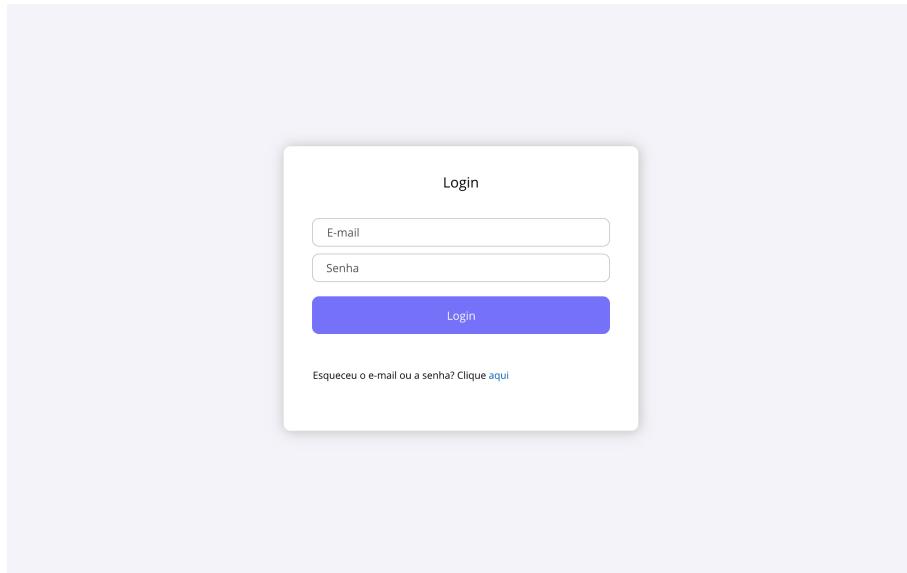


Figura 57 – Tela de login.

A Figura 58 representa o *design* da tela inicial da aplicação, onde o usuário pode visualizar os dados quantitativos referentes a sua produção. É exibido na tela as quantidades totais de laranjas processadas, de laranjas boas e laranjas descartadas, além das subclassificações das laranjas boas, que são os totais de laranjas com manchas e sem manchas, dentre as sem manchas, o usuário ainda consegue visualizar as quantidades das laranjas por classificação de tamanho.

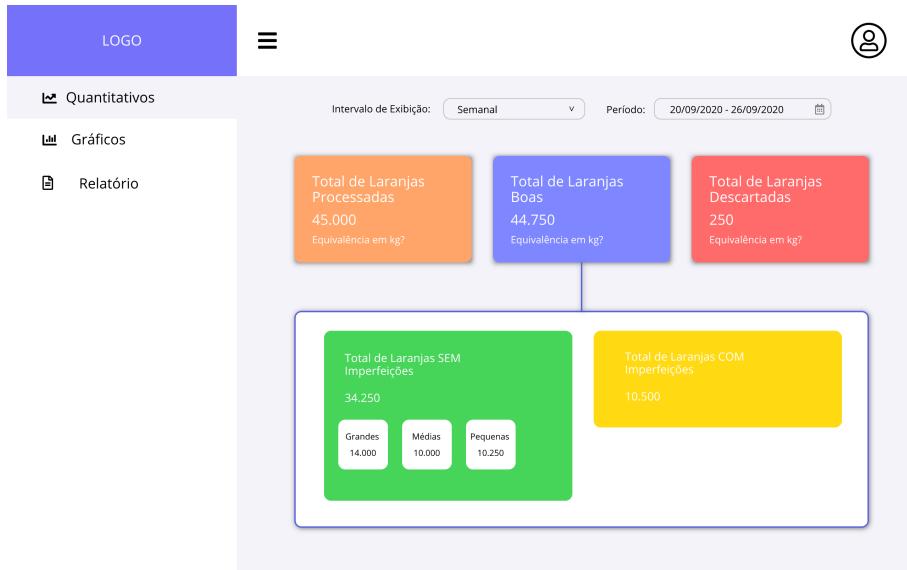


Figura 58 – Tela de informações quantitativas.

As Figuras 59, 60 e 61 representam o design da tela de gráficos, onde o usuário pode visualizar os dados de forma gráfica referentes a sua produção. É exibido na tela os gráficos das laranjas de acordo com o período selecionado pelo usuário e qual o tipo de informação que deseja visualizar. O intervalo de exibição pode ser semanal, mensal ou anual. O usuário

pode escolher o período de acordo com o intervalo selecionado. A informação do gráfico pode ser referente a quantidades de laranjas processadas, quantidades de laranjas boas, quantidades de laranjas boas sem manchas e quantidades de laranjas boas com manchas.

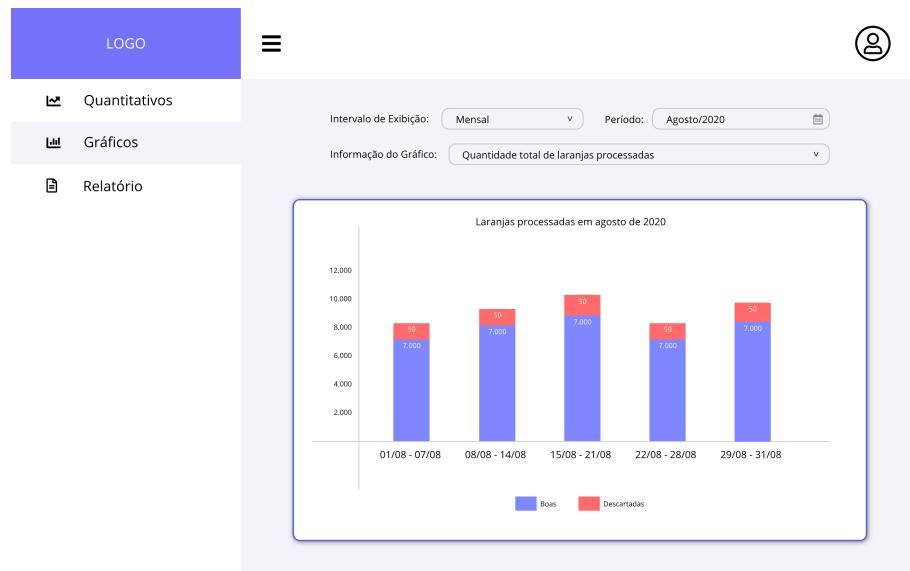


Figura 59 – Tela de informações em forma gráfica (total de laranjas processadas).

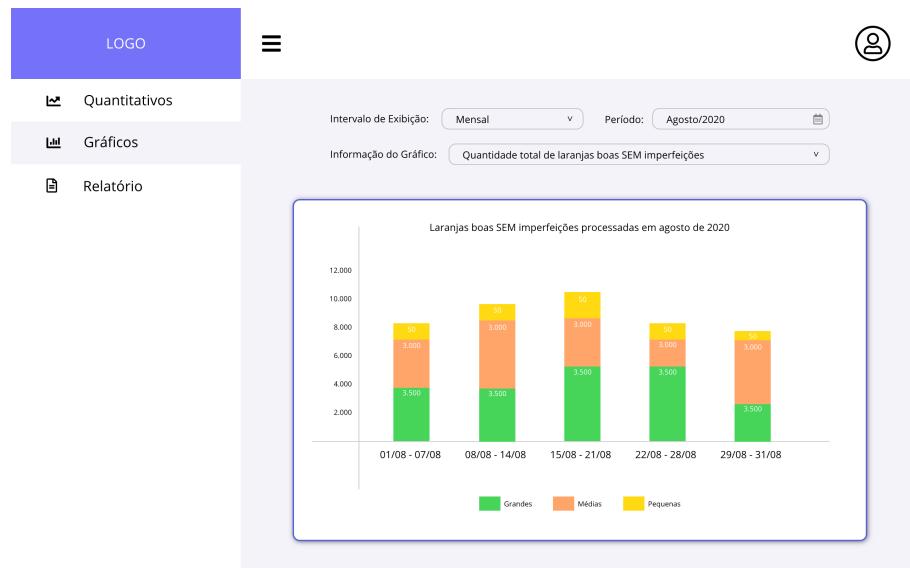


Figura 60 – Tela de informações em forma gráfica (total de laranjas boas SEM imperfeições).

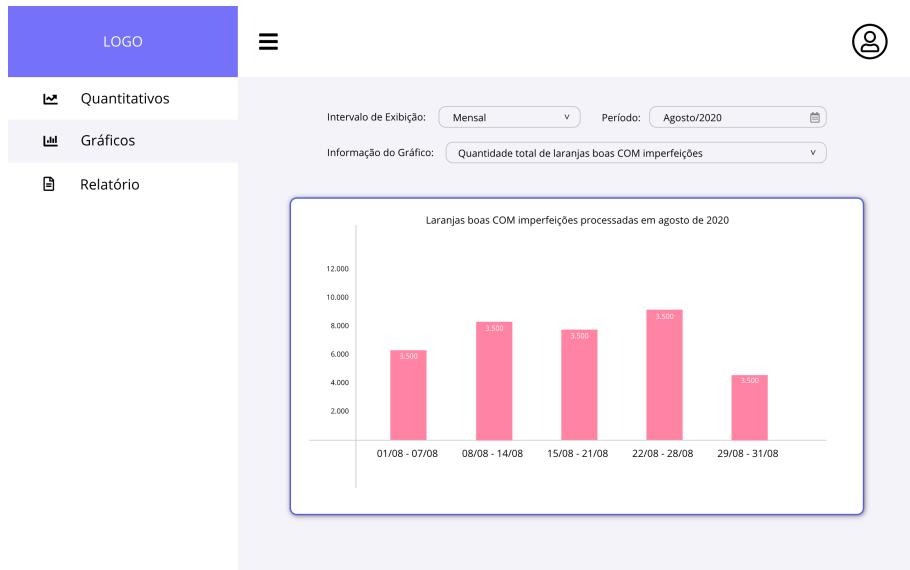


Figura 61 – Tela de informações em forma gráfica (total de laranjas boas COM imperfeições).

A Figura 62 representa o *design* da tela de relatório da aplicação, onde o usuário pode selecionar o intervalo e o período que deseja gerar o relatório.

Gerar Relatório de Dados Quantitativos

Selecione o intervalo e o período dos quais você deseja que os dados sejam recuperados. Você pode adicionar mais de um intervalo e período.

Intervalo:	Mensal	Período:	Agosto/2020
Intervalo:	Mensal	Período:	Agosto/2020

[+]

**Gerar Relatório**

Figura 62 – Tela de geração de relatório.

#### 4.5.1.1.2 Resultados

Como resultado da implementação da aplicação web, as funcionalidades principais previstas para o módulo foram codificadas e estão em funcionamento. As telas de apresentação dos dados na aplicação estão representadas nas figuras a seguir.

A figura 63 mostra a tela de apresentação dos dados quantitativos da colheita na aplicação web. Nela, o usuário pode escolher o intervalo de tempo para o qual deseja visualizar os dados e as quantidades das classificações e subclassificações das laranjas processadas.

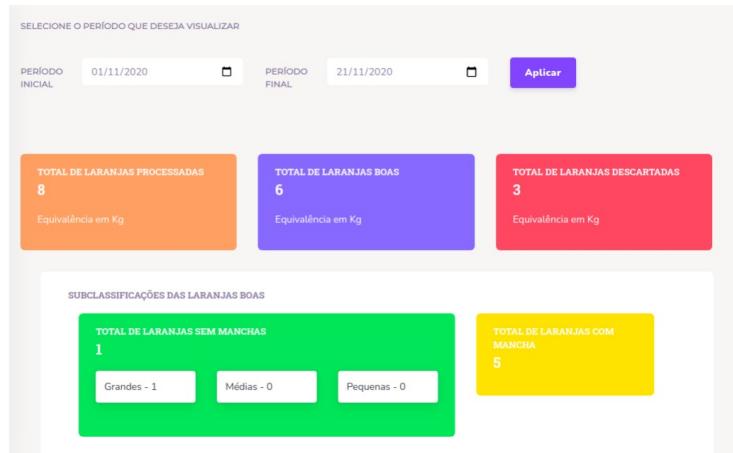


Figura 63 – Tela de dados quantitativos da aplicação web.

A figura 64 mostra o gráfico do total de laranjas por mês apresentado na aplicação web. O usuário pode selecionar a classificação na qual deseja que os dados sejam apresentados. As subclassificações são representadas por cores diferentes no gráfico. No caso da figura, as laranjas boas estão representadas pelas faixas em azul e as ruins pelas faixas em vermelho.

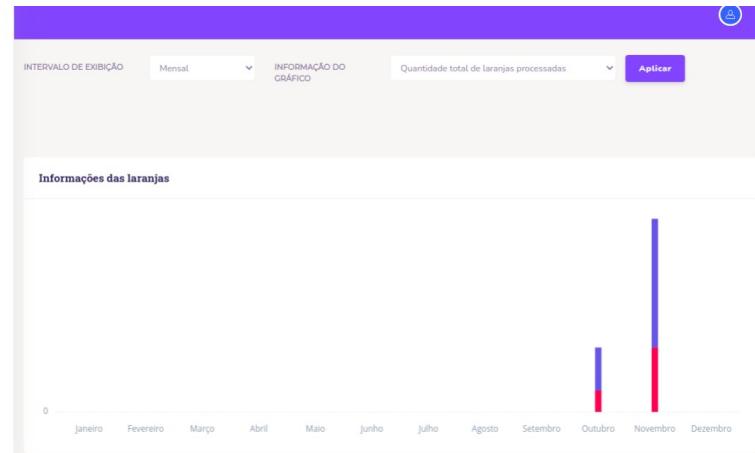


Figura 64 – Tela do gráfico de total de laranjas por mês da aplicação web.

A figura 65 mostra o gráfico do total de laranjas por ano apresentado na aplicação web. O usuário pode selecionar o ano de início e o de fim para os quais deseja visualizar os dados.

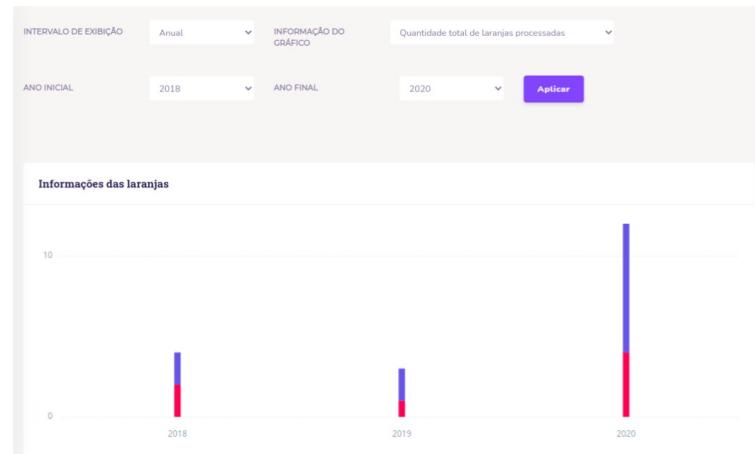


Figura 65 – Tela do gráfico de total de laranjas por ano da aplicação web.

A figura 66 mostra o gráfico do total de laranjas boas SEM manchas apresentado na aplicação web. As subclassificações de tamanho das laranjas estão representadas em cores diferentes e o usuário pode selecionar o mês e o ano para os quais deseja que os dados sejam apresentados.

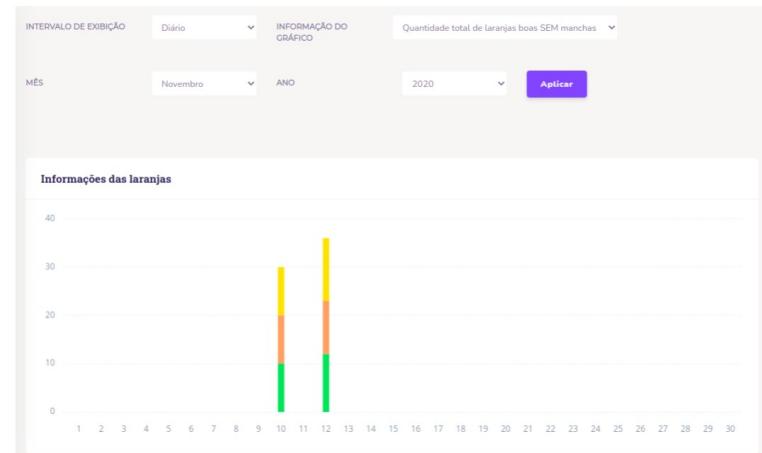


Figura 66 – Tela do gráfico de total de laranjas boas SEM manchas por dia da aplicação web.

A figura 67 mostra o gráfico do total de laranjas boas COM manchas apresentado na aplicação web.

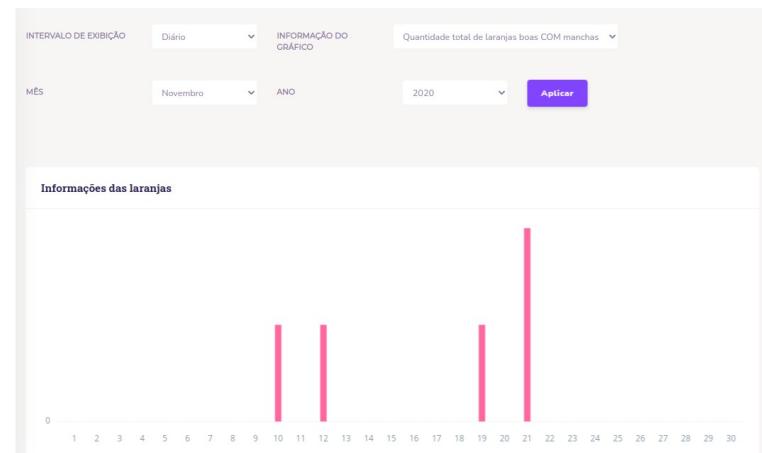


Figura 67 – Tela do gráfico de total de laranjas boas COM manchas por dia da aplicação web.

A figura 68 mostra a tela de geração de relatório da aplicação web. Nela, o usuário pode escolher o intervalo de tempo para o qual deseja que os dados sejam recuperados e, ao clicar em "Gerar Relatório", um arquivo .csv é gerado com os dados quantitativos da colheita para o intervalo de tempo selecionado.

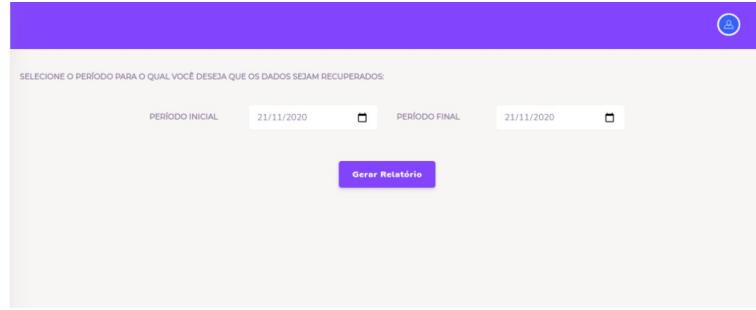


Figura 68 – Tela de geração de relatório da aplicação web.

#### 4.5.1.2 Aplicação Desktop

##### 4.5.1.2.1 Protótipo

A Figura 69 representa o *design* da tela do Desktop App, onde o usuário consegue visualizar os dados de forma gráfica e quantitativa referentes ao processamento. É possível visualizar os gráficos por qualidade e tamanho das laranjas que estão sendo processadas, assim como suas quantidades em forma de tabela.

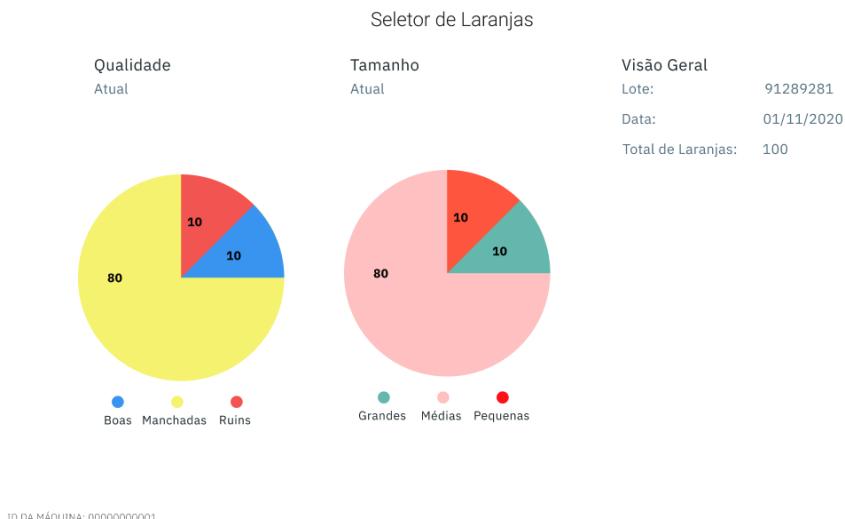


Figura 69 – Painel da Aplicação Desktop.

##### 4.5.1.2.2 Resultados

A figura 70 mostra a tela de apresentação dos dados referentes à colheita do usuário, bem como o lote atual que os dados se referem.

Além dos dados referentes às laranjas em si, a figura 70 traz dados referentes ao *id* da máquina do usuário, bem como total de laranjas que já foram classificadas e a data que o lote está sendo classificado.

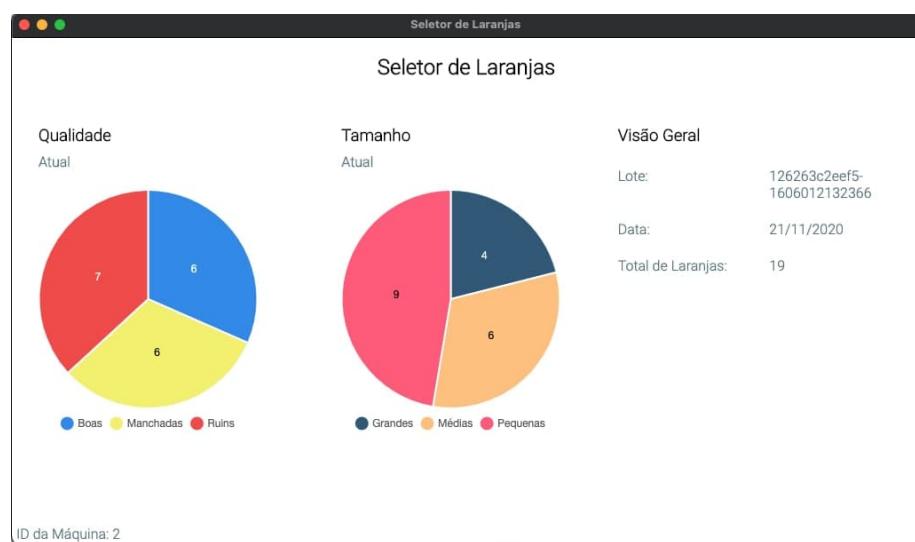


Figura 70 – Tela de dados quantitativos da aplicação desktop.

## 4.5.2 Arquitetura de Software

Para explicar a solução computacional do projeto, esta seção é encarregada de detalhar todos os módulos inseridos na área de *software*.

### 4.5.2.1 Integração de Todos os Sistemas de Software

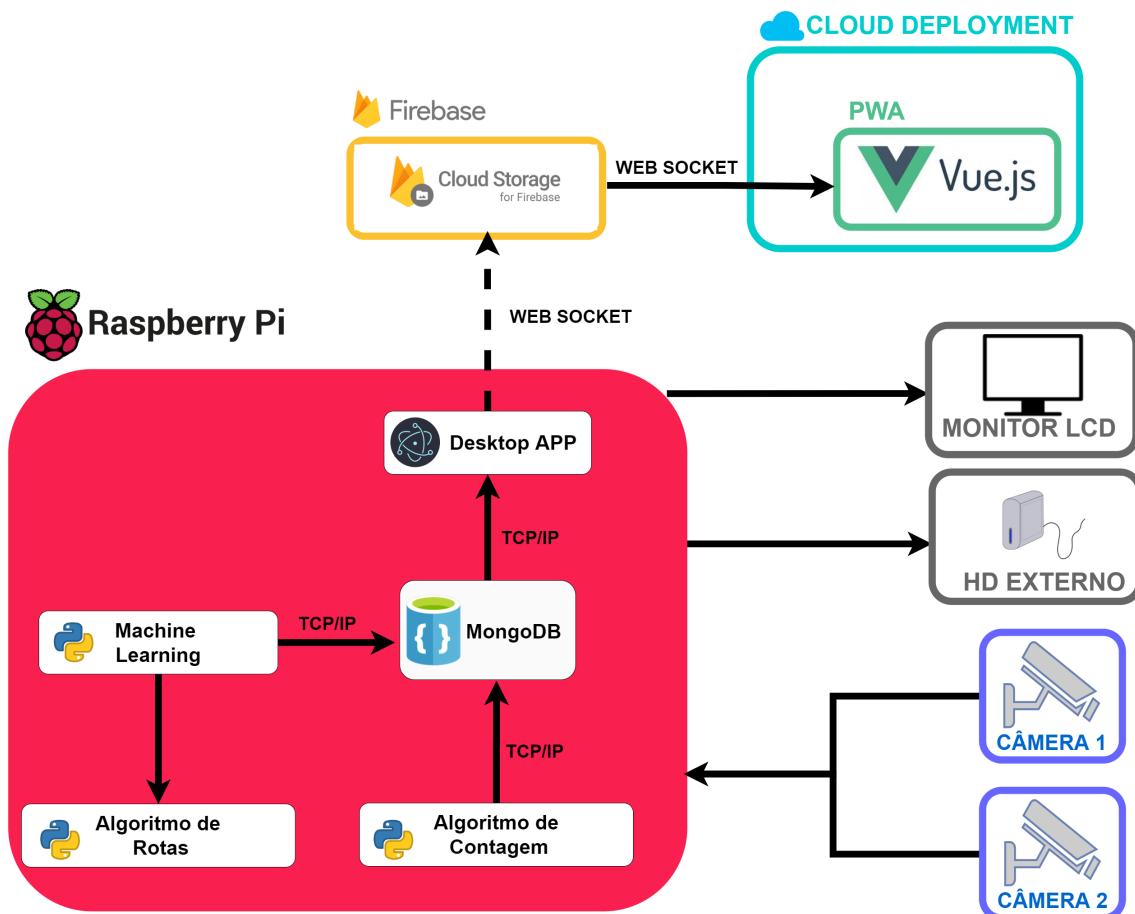


Figura 71 – Diagrama geral da Arquitetura de Software.

Com o objetivo de atender a demanda computacional do projeto a solução de software foi arquitetada em quatro aplicações diferentes, as quais são: *Machine Learning*, *Desktop APP*, *back-end* como serviço (*Backend-as-a-Service*) utilizando Firebase e Aplicação Web, como representado no diagrama da Figura 71. Para o armazenamento na própria Raspberry Pi 4 é utilizado um banco de dados não relacional utilizando MongoDB. O algoritmo de contagem e de rotas está sob responsabilidade da equipe de eletrônica. Estes diagramas foram representados em nossa representação, pois eles interagem com software com trocas de dados. Será descrito abaixo a responsabilidade de cada módulo.

O primeiro módulo consiste em uma rede neural artificial, a qual é treinada por meio de imagens de laranjas, a fim de reconhecer as frutas que estão boas e ruins e, dentre as frutas boas, também é capaz de classificar entre as que estão com manchas e sem

manchas. Este módulo é composto de um algoritmo de *machine learning* (representado na figura 71) utilizando a linguagem de programação *Python*, compondo a Rede Neural Convolucional, que é previamente treinada com imagens referentes a laranjas boas e ruins, para que a máquina entenda a diferença e consiga discernir as imagens que serão captadas pelos dois Módulos de Câmera 5mp Raspberry Pi V1.3 (Figura 35). Este algoritmo de treinamento da Rede Neural é feito separadamente da Raspberry PI 4 (Figura ??), devido a necessidade de alta performance de processamento. Após o treinamento, o algoritmo é inserido na Raspberry PI 4 (Figura ??).

Ao receber as imagens dos Módulos de Câmera 5mp Raspberry Pi V1.3 (Figura 35), a partir de duas imagens o algoritmo é capaz de classificar as laranjas entre laranjas com manchas, sem manchas ou ruins. Após a classificação, os dados são armazenados em um banco de dados não relacional, MongoDB (representado na figura 71), dentro da Raspberry PI 4 (Figura ??) e, além disso, envia o dado de classificação para o algoritmo de rotas de eletrônica, a fim de realizar o movimento dos servos.

O segundo módulo, que está sendo representado na figura 71 como *Desktop APP*, consiste numa solução que possui um aplicativo *desktop*, que permite ao produtor visualizar os dados quantitativos sobre a remessa de laranja que está sendo processada pela máquina por meio de um monitor conectado com Raspberry PI 4 (Figura ??), não demandando conexão com a internet. Este módulo é feito utilizando o *framework Electron*, que consome os dados do banco de dados local *MongoDB* e apresenta os dados de forma quantitativa e de forma gráfica no monitor conectado com a *Raspberry PI 4* (Figura ??). Este módulo foi pensado para que o produtor pudesse ter acesso aos dados mesmo não tendo acesso à internet em sua área de produção. Porém, caso a internet seja acessível, este módulo é capaz de enviar as informações processadas para a nuvem (*Firebase Cloud Firestore*).

O terceiro módulo está representado pelo *Firebase* que será utilizado como um serviço do projeto, onde será possível armazenar dados em nuvem. Será utilizado o *Cloud Firestore*, que é um banco de dados orientado a documentos, onde serão armazenados os dados das classificações das laranjas para que seja possível construir um *dashboard web* que disponibilize as informações. Este módulo possui uma comunicação via *WebSocket*, onde a comunicação é bidirecional.

Já o quarto módulo foi pensado para caso o produtor tenha acesso à internet no local de sua produção e consiga enviar os dados para o *Firebase*. Este módulo consistirá em um aplicativo *Progressive Web App (PWA)*, o qual é um aplicativo *Web* adaptável a qualquer dispositivo *mobile* e *desktop*, que permitirá ao produtor visualizar de forma gráfica os dados de sua colheita, além disso é possível gerar relatórios referentes a datas pré-selecionadas pelo usuário. Será feito utilizando as tecnologias *Vue.js*, *VueX* e *Nuxt*. Este módulo se comunicará com o *Firebase*, através do *SDK* disponibilizado pela plata-

forma, a fim de coletar os dados da colheita e persistir informações de usuários. Ele será responsável por apresentar os dados de forma mais robusta e com uma melhor experiência de visualização para o produtor. Serão apresentados os dados de forma quantitativa e de forma gráfica, conforme a classificação e os períodos pré-definidos pelo usuário. Além da visualização, o usuário poderá gerar relatórios sobre a colheita, de acordo com o período desejado.

O banco de dados local está representado na figura 71 por *MongoDB*, que é um banco de dados orientado a documentos, que armazena as informações das classificações e das imagens capturadas. Estas imagens serão importantes para retreinar a rede neural, a fim de aprimorar sua capacidade de classificação. As classificações que serão armazenadas são: laranjas ruins, laranjas boas com manchas, laranjas boas sem manchas e as quantidades de laranjas sem manchas classificadas por tamanho. Esta última informação será recebida do Algoritmo de Contagem de eletrônica. A comunicação feita dos algoritmos com o banco de dados local é feita através do protocolo *TCP/IP*.

Os demais componentes são periféricos necessários que estão conectados com a *Raspberry PI 4* (Figura ??). O monitor é necessário para exibir as informações quantitativas e gráficas do algoritmo do *Desktop APP* para o operador da máquina. O HD Externo se fez necessário, pois o *SD Card* que vem junto com a *Raspberry* não é uma solução sustentável devido sua vida útil ser reduzida quando lida com ciclos de gravação e deleção muito intensos, como ocorre quando se utiliza banco de dados. Além disso, possui as câmeras de capturas das imagens e sensores (não representados, pois entra no escopo de eletrônica) conectados com a *Raspberry*.

#### 4.5.2.2 Machine Learning

Para o desenvolvimento do módulo de *machine learning* do projeto, são necessários alguns passos, sendo eles:

1. Obtenção de um *dataset* pra treinar o código;
2. tratamento e importação dos dados para uso;
3. importação de um modelo para treinamento;
4. incrementação das camadas no modelo;
5. treinamento do modelo desenvolvido;
6. avaliação da qualidade do modelo através da precisão;
7. uso da rede neural;
8. armazenamento dos dados gerados no banco de dados *MongoDB*;

## 9. envio dos dados de classificação para o Algoritmo de Rotas de eletrônica.

Inicialmente foi idealizado testar o trabalho com diversas ferramentas dentro do escopo do projeto, objetivando métricas comparativas para seleção da melhor ferramenta no quesito de desempenho. Diante disso, foi levantado algumas bibliotecas com as quais poderíamos desenvolver o algoritmo, sendo estas: *Keras*, *Tensorflow* e *Pytorch*.

Conforme foi desenvolvido, foi notado que o *Keras* se tratava de uma biblioteca evoluída do *Tensor Flow*, reduzindo as alternativas a apenas duas. Essas duas são o *Keras* e o *Pytorch*. Inicialmente, pensava-se que o *Pytorch* se mostraria mais eficiente para o projeto. Entretanto, após serem desenvolvidas as duas redes neurais, isso não se mostrou verdade. O *Pytorch* apresentou uma acurácia de apenas 91%. Enquanto o *Keras*, inicialmente mostrou uma acurácia de 97%. Após a evolução do algoritmo, esta se encontra em aproximadamente 99% apresentando apenas 1 erro a cada 100 imagens processadas.

A seguir foi feita uma descrição dos procedimentos seguidos e uma apuração dos resultados, assim como o algoritmo de escolha para integrar o sistema definitivamente.

### 4.5.2.2.1 Código Utilizando o PyTorch

A fim de realizar uma análise de desempenho para comparação com a biblioteca *Keras*, foi realizada uma sequência de tarefas e protocolos utilizando a biblioteca *PyTorch*. Para isso, utilizou-se as bibliotecas *os*, *glob*, *openCV*, *matplotlib*, *tdqm*, *datetime*, *time*, *torch* e *warning*.

Após a definição das bibliotecas, foi necessária a criação de funções para realizar os testes com a biblioteca *PyTorch*. A primeira função faz uso da OpenCV, uma biblioteca utilizada para a conversão de espaço de cores. A segunda função é uma parte do código que acessa as imagens, através do caminho de acesso ao arquivo, sendo a obtenção deste auxiliada pela biblioteca *os*. Após isso, fazemos uma plotagem das imagens obtidas a serem utilizadas na classificação.

Criou-se então uma classe abstrata *FruitsDataset*, onde essa herda o atributo *Dataset*. Essa classe é iniciada com 4 atributos: Altura da Imagem, Largura da Imagem, Rótulo e *Path* da Imagem.

Estes são então tratados de acordo com a classificação e, com base no resultado, atribui-se um valor entre 0 e 1, com base na qualidade da fruta.

Agora, antes de iniciar o processamento das imagens, necessitamos, por obrigação dos próprios algoritmos, de realizar um pré-processamento/tratamento dos dados de input. Diante disso, a primeira abordagem é realizar uma reestruturação da dimensão das imagens a serem tratadas, para que todas se encontrem no mesmo tamanho. Para isso, utilizamos a função *transforms.ToTensor()* e *transforms.Normalize()*, onde a primeira

executa uma conversão das imagens de um *array numpy* para imagens *torch* e o segundo normalizamos as imagens para estas terem uma média zero e variância zero também. Feito isso, criamos uma classe *FruitDataset()* com as imagens já tratadas.

Feito o pré-processamento, partimos então para a etapa de preparação do *dataset* para o processamento, fazendo uma iteração preparatória nestes e preparando os dados de teste e de treino.

Os resultados do treino para o algoritmo do *PyTorch* são explicitados abaixo:

- Treinamento utilizando *pytorch* 25 épocas: 175.96667766571045 segundos
- Média: 7 segundos por época.
- Dataset: 3061 imagens
- Treino (1500 laranjas ruins / 800 boas)
- Memoria máxima durante treinamento 4.0GB
- Memoria maxima GPU 763 MB

Já com os dados treinados, partimos para a validação do modelo, que resultou em uma acurácia de 91% com os dados de treino e 93% com os dados de teste.

#### 4.5.2.2.2 Algoritmo utilizando o Keras

O *Keras* consegue fazer uso direto de uma GPU para maior agilidade e eficiência no treinamento. Para desenvolver o algoritmo utilizamos as seguintes bibliotecas: *numpy*, *Pandas*, *Os*, *OpenCV*, *matplotlib*, *tqdm*, *random*, *keras*, *pickle* e *tensorflow*.

Foi necessário o tratamento das imagens. Para isso, foi utilizado a biblioteca de visão computacional *OpenCV*, esta a qual permite a conversão das imagens em matrizes de formato RGB de dimensão 100x100 pixels. Este valor foi obtido através de testes práticos, levando ao limite o hardware utilizado para desenvolver e treinar o algoritmo. Feita essa transformação, as imagens foram transformadas em um arranjo do *Numpy*, de dimensões (36,100,100,3), sendo o 36 representativo da quantidade de linhas de imagem existentes, 100 e 100 das dimensões da imagem e 3 sendo o espectro RGB, sendo o R vermelho, G verde e B azul.

Ao converter as imagens, foi realizado o mesmo procedimento de visão computacional; foram convertidas as novas imagens carregadas; e foi montado o *set* de treino e teste. O resultado obtido foi de 2283 imagens de frutas boas com manchas, 1595 de ruins e 1466 de laranjas boas sem manchas para treino. Já para o teste foi obtido um resultado de 1180 imagens de boas com manchas, 389 de imagens boas sem manchas, 406 ruins.

Iniciando a utilização do Keras, após termos realizado o tratamento dos dados em questão, partimos para a estruturação da rede neural e o treinamento do algoritmo, assim como a construção do modelo.

Do *Keras*, fizemos uso de diversas funções, entre elas, o *Dense()*, *Dropout()*, *Conv2D()*, *MaxPooling()*, *Activation()*, *Flatten()*, *BatchNormalization()*, *SeparableConv2D*, *Sequential()*, *Model()* e *loadmodel()*.

Após feito o tratamento das imagens e com os componentes já importados. Foi utilizado o modelo *Sequential()* do *Keras*, sendo este um modelo que se sobressai quando se trata de uma pilha de camadas simples, tendo cada camada exatamente um tensor de entrada e outro de saída. Neste modelo adicionamos um total de 20 camadas, estas camadas abrangendo diversas utilidades dentro do treinamento.

A primeira camada foi uma camada de convolução, onde, essa realiza a extração dos recursos da imagem de entrada e cria 32 recursos convolutos 3x3 *ReLU* ou, também conhecido como mapa de recursos, posteriormente adicionamos outras duas dessa mesma camada. A rede neural possui também em sua composição 3 camadas de Normalização, 3 camadas de *Pooling*, que criam uma subamostragem do mapa de recursos, 3 camadas de convolução separáveis 2d, outras 3 camadas de *dropout*, 2 de densidade e também outra de achatamento. Mais informações a respeito dessas camadas e outras funcionalidades do *Keras* pode ser encontrada na documentação oficial Keras ([DOCUMENTAÇÃO...](#)).

Abaixo podemos ver um relatório da composição da rede gerada:

```
Model: "sequential"
```

Layer (type)	Output Shape	Param #
<hr/>		
conv2d (Conv2D)	(None, 100, 100, 32)	896
batch_normalization (BatchNo	(None, 100, 100, 32)	128
separable_conv2d (SeparableC	(None, 100, 100, 32)	1344
max_pooling2d (MaxPooling2D)	(None, 50, 50, 32)	0
batch_normalization_1 (Batch	(None, 50, 50, 32)	128
dropout (Dropout)	(None, 50, 50, 32)	0
separable_conv2d_1 (Separabl	(None, 50, 50, 64)	2400
separable_conv2d_2 (Separabl	(None, 50, 50, 64)	4736
batch_normalization_2 (Batch	(None, 50, 50, 64)	256
max_pooling2d_1 (MaxPooling2	(None, 25, 25, 64)	0
dropout_1 (Dropout)	(None, 25, 25, 64)	0
conv2d_1 (Conv2D)	(None, 25, 25, 128)	73856
conv2d_2 (Conv2D)	(None, 25, 25, 128)	147584
batch_normalization_3 (Batch	(None, 25, 25, 128)	512
max_pooling2d_2 (MaxPooling2	(None, 12, 12, 128)	0
dropout_2 (Dropout)	(None, 12, 12, 128)	0
flatten (Flatten)	(None, 18432)	0
dense (Dense)	(None, 128)	2359424
dropout_3 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 1)	129
<hr/>		
Total params:	2,591,393	
Trainable params:	2,590,881	
Non-trainable params:	512	

Figura 72 – Relatório Composição da Rede Neural

Após a composição da rede neural, partimos para realizar o *fitting* do modelo, resultando na representação a seguir:

```

Epoch 1/25
154/154 [=====] - ETA: 0s - loss: 0.6168 - accuracy: 0.8703
Epoch 00001: val_loss improved from inf to 11.57578, saving model to ./rotten.h5
154/154 [=====] - 10s 670ms/step - loss: 0.6168 - accuracy: 0.8703 - val_loss: 11.5757 - val_accuracy: 0.4893
Epoch 2/25
154/154 [=====] - ETA: 0s - loss: 0.1974 - accuracy: 0.9271
Epoch 00002: val_loss did not improve from 11.57578
154/154 [=====] - 18s 664ms/step - loss: 0.1974 - accuracy: 0.9271 - val_loss: 12.2643 - val_accuracy: 0.4893
Epoch 3/25
154/154 [=====] - ETA: 0s - loss: 0.1673 - accuracy: 0.9402
Epoch 00003: val_loss improved from 11.57578 to 8.80041, saving model to ./rotten.h5
154/154 [=====] - 7s 488ms/step - loss: 0.1673 - accuracy: 0.9402 - val_loss: 8.8004 - val_accuracy: 0.5811
Epoch 4/25
154/154 [=====] - ETA: 0s - loss: 0.1285 - accuracy: 0.9579
Epoch 00004: val_loss improved from 8.80041 to 0.18174, saving model to ./rotten.h5
154/154 [=====] - 9s 594ms/step - loss: 0.1285 - accuracy: 0.9579 - val_loss: 0.1817 - val_accuracy: 0.9597
Epoch 5/25
154/154 [=====] - ETA: 0s - loss: 0.0951 - accuracy: 0.9670
Epoch 00005: val_loss improved from 0.18174 to 0.14052, saving model to ./rotten.h5
154/154 [=====] - 18s 664ms/step - loss: 0.0951 - accuracy: 0.9670 - val_loss: 0.1405 - val_accuracy: 0.9668
Epoch 6/25
154/154 [=====] - ETA: 0s - loss: 0.1821 - accuracy: 0.9681
Epoch 00006: val_loss improved from 0.14052 to 0.12835, saving model to ./rotten.h5
154/154 [=====] - 18s 668ms/step - loss: 0.1821 - accuracy: 0.9681 - val_loss: 0.1283 - val_accuracy: 0.9761
Epoch 7/25
154/154 [=====] - ETA: 0s - loss: 0.0921 - accuracy: 0.9713
Epoch 00007: val_loss improved from 0.12835 to 0.06441, saving model to ./rotten.h5
154/154 [=====] - 7s 461ms/step - loss: 0.0921 - accuracy: 0.9713 - val_loss: 0.0644 - val_accuracy: 0.9774
Epoch 8/25
154/154 [=====] - ETA: 0s - loss: 0.1319 - accuracy: 0.9530
Epoch 00008: val_loss did not improve from 0.06441
154/154 [=====] - 7s 484ms/step - loss: 0.1319 - accuracy: 0.9530 - val_loss: 0.5187 - val_accuracy: 0.9157
Epoch 9/25
154/154 [=====] - ETA: 0s - loss: 0.0998 - accuracy: 0.9637
Epoch 00009: val_loss did not improve from 0.06441
154/154 [=====] - 7s 488ms/step - loss: 0.0998 - accuracy: 0.9637 - val_loss: 0.0966 - val_accuracy: 0.9648
Epoch 10/25
154/154 [=====] - ETA: 0s - loss: 0.0639 - accuracy: 0.9794
Epoch 00010: val_loss did not improve from 0.06441
154/154 [=====] - 8s 534ms/step - loss: 0.0639 - accuracy: 0.9794 - val_loss: 0.1535 - val_accuracy: 0.9736
Epoch 11/25
154/154 [=====] - ETA: 0s - loss: 0.0729 - accuracy: 0.9755
Epoch 00011: val_loss improved from 0.06441 to 0.05930, saving model to ./rotten.h5
154/154 [=====] - 9s 585ms/step - loss: 0.0729 - accuracy: 0.9755 - val_loss: 0.0593 - val_accuracy: 0.9849
Epoch 12/25
154/154 [=====] - ETA: 0s - loss: 0.0482 - accuracy: 0.9804
Epoch 00012: val_loss improved from 0.05930 to 0.02524, saving model to ./rotten.h5
154/154 [=====] - 18s 659ms/step - loss: 0.0482 - accuracy: 0.9804 - val_loss: 0.0252 - val_accuracy: 0.9899
Epoch 13/25
154/154 [=====] - ETA: 0s - loss: 0.0658 - accuracy: 0.9804
Epoch 00013: val_loss did not improve from 0.02524
154/154 [=====] - 18s 655ms/step - loss: 0.0658 - accuracy: 0.9804 - val_loss: 0.1916 - val_accuracy: 0.9635
Epoch 14/25
154/154 [=====] - ETA: 0s - loss: 0.0634 - accuracy: 0.9794
Epoch 00014: val_loss did not improve from 0.02524
154/154 [=====] - 18s 655ms/step - loss: 0.0634 - accuracy: 0.9794 - val_loss: 0.1189 - val_accuracy: 0.9686
Epoch 15/25
154/154 [=====] - ETA: 0s - loss: 0.2458 - accuracy: 0.9350
Epoch 00015: val_loss did not improve from 0.02524
154/154 [=====] - 8s 549ms/step - loss: 0.2458 - accuracy: 0.9350 - val_loss: 0.1092 - val_accuracy: 0.9597
Epoch 16/25
154/154 [=====] - ETA: 0s - loss: 0.1836 - accuracy: 0.9668
Epoch 00016: val_loss did not improve from 0.02524
154/154 [=====] - 18s 655ms/step - loss: 0.1836 - accuracy: 0.9668 - val_loss: 0.1328 - val_accuracy: 0.9673
Epoch 17/25
154/154 [=====] - ETA: 0s - loss: 0.0741 - accuracy: 0.9745
Epoch 00017: val_loss did not improve from 0.02524
154/154 [=====] - 18s 653ms/step - loss: 0.0741 - accuracy: 0.9745 - val_loss: 0.0416 - val_accuracy: 0.9874
Epoch 18/25
154/154 [=====] - ETA: 0s - loss: 0.0439 - accuracy: 0.9887
Epoch 00018: val_loss did not improve from 0.02524
154/154 [=====] - 9s 588ms/step - loss: 0.0441 - accuracy: 0.9887 - val_loss: 0.1569 - val_accuracy: 0.9673
Epoch 19/25
154/154 [=====] - ETA: 0s - loss: 0.0884 - accuracy: 0.9693
Epoch 00019: val_loss did not improve from 0.02524
154/154 [=====] - 9s 586ms/step - loss: 0.0884 - accuracy: 0.9693 - val_loss: 0.0992 - val_accuracy: 0.9698
Epoch 20/25
154/154 [=====] - ETA: 0s - loss: 0.0882 - accuracy: 0.9745
Epoch 00020: val_loss did not improve from 0.02524
154/154 [=====] - 18s 653ms/step - loss: 0.0882 - accuracy: 0.9745 - val_loss: 0.0788 - val_accuracy: 0.9786
Epoch 21/25
154/154 [=====] - ETA: 0s - loss: 0.0452 - accuracy: 0.9843
Epoch 00021: val_loss did not improve from 0.02524
154/154 [=====] - 8s 572ms/step - loss: 0.0452 - accuracy: 0.9843 - val_loss: 0.0915 - val_accuracy: 0.9811
Epoch 22/25
154/154 [=====] - ETA: 0s - loss: 0.0375 - accuracy: 0.9846
Epoch 00022: val_loss did not improve from 0.02524
154/154 [=====] - 18s 653ms/step - loss: 0.0375 - accuracy: 0.9846 - val_loss: 0.1751 - val_accuracy: 0.9673
Epoch 23/25
154/154 [=====] - ETA: 0s - loss: 0.0292 - accuracy: 0.9912
Epoch 00023: val_loss did not improve from 0.02524
154/154 [=====] - 18s 658ms/step - loss: 0.0292 - accuracy: 0.9912 - val_loss: 0.0716 - val_accuracy: 0.9811
Epoch 24/25
154/154 [=====] - ETA: 0s - loss: 0.0376 - accuracy: 0.9869
Epoch 00024: val_loss did not improve from 0.02524
154/154 [=====] - 9s 619ms/step - loss: 0.0376 - accuracy: 0.9869 - val_loss: 0.0515 - val_accuracy: 0.9862
Epoch 25/25
154/154 [=====] - ETA: 0s - loss: 0.0481 - accuracy: 0.9827
Epoch 00025: val_loss did not improve from 0.02524
154/154 [=====] - 7s 470ms/step - loss: 0.0481 - accuracy: 0.9827 - val_loss: 0.1242 - val_accuracy: 0.9748

```

Figura 73 – Treinamento Épocas

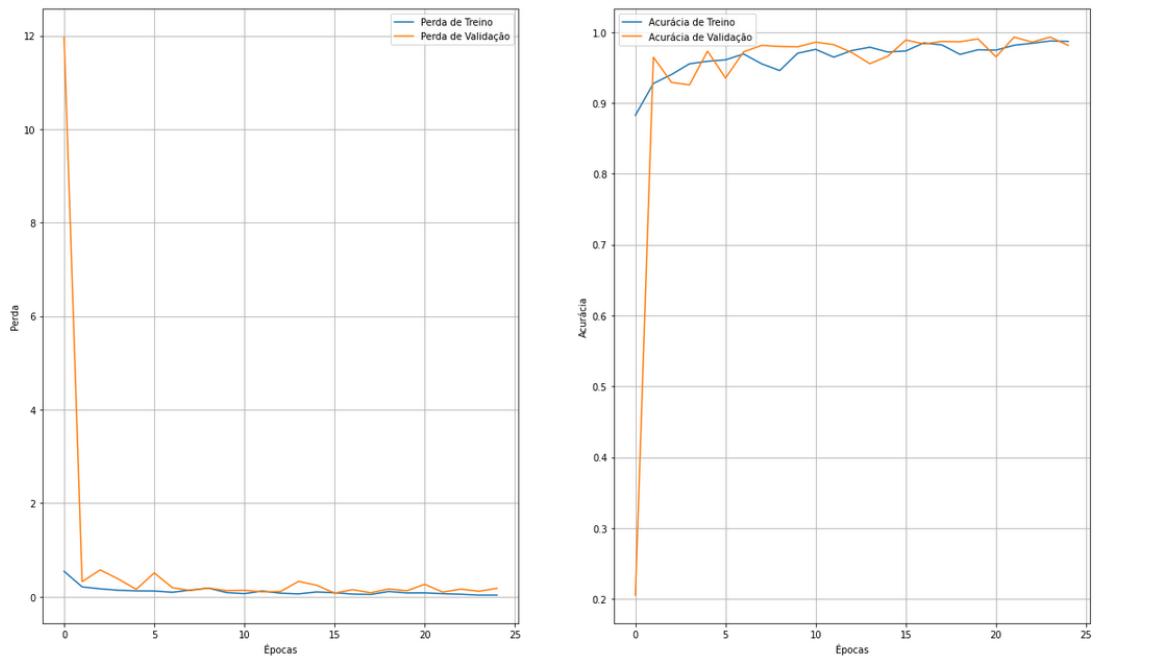


Figura 74 – Gráfico da evolução da acurácia e perda com o passar das épocas

Feita a estruturação da rede neural, foi continuado o algoritmo para o ponto de treinamento, onde foi realizada a diminuição dimensional das imagens, dividindo a matriz por 255, valor esse representativo da quantidade de colunas e linhas existentes numa matriz RGB. Com isso feito, realizamos o *fitting* do *dataset* no modelo.

Com o desenvolvimento das épocas, pode-se observar um salto da acurácia e diminuição da perda considerável, estando a acurácia na detecção da qualidade das laranjas em 89% na primeira época. Ao final da vigésima quinta época, a última, mostra-se uma acurácia de treino de 98.56% assim como um *loss* de treino de 54% na primeira época e esse valor sendo reduzido para 3.34% na última época. A acurácia de teste evoluiu de 20.56% para 98.13% no final do treinamento.

#### 4.5.2.2.3 Resultados e Conclusões

Diante das soluções apresentadas anteriormente, obtivemos os resultados da acurácia para o Pytorch 93% de Acurácia e Keras 98.56% de Acurácia. Os resultados de tempo de treinamento para 25 épocas para o PyTorch foi de 172.81 segundos (imagens 100x100 pixels) e para o Keras de 300 segundos (imagens 100x100 pixels). Os resultados de tempo de classificação com o modelo já carregado para o PyTorch foi de 100ms (por imagem) e para o Keras foi de 110ms (por imagem).

Com base nestes resultados, optamos pela utilização do Keras. Este permite implementar, treinar e testar suas redes rapidamente, com depuração fácil.

#### 4.5.2.2.4 Especificações Técnicas

O algoritmo foi todo testado e treinado em uma máquina com 12 núcleos, 16GB de RAM e uma GPU com 4GB de RAM.

Com o modelo já treinado e salvo no formato de serialização h5, o algoritmo vai receber um *input* com fotos da laranja e vai classificar foto por foto sendo capaz de fazer um *reduce* do resultado aumentando ainda mais sua capacidade de acertar a classificação correta. Para cada imagem que for classificada, o algoritmo de forma assíncrona será capaz de salvar no formato base64 para que o modelo seja treinado outras vezes se o acesso à internet for possível. As principais bibliotecas utilizadas na formatação e no pré-processamento foram a biblioteca opencv para redimensionar o tamanho das imagens e fazer *data augmentation* (aumento dos dados para treinamento), *pyplot* para visualizar e trabalhar melhor com os dados. O *dataset* utilizado foi salvo no Kaggle e está disponível em ([DATASET, 2020](#)).

Desenvolvido o modelo de classificação, foi necessário que este modelo fosse integrado aos demais sistemas e se mostrar disponível para uso. Com isso a equipe desenvolveu uma micro-aplicação em Flask, a qual, é acessível, atualmente por *web* ou rodando o *container docker* desta.

A API desenvolvida é responsável por dar uma interface ao mecanismo de classificação, visto as condições extraordinárias deste semestre, não haveria forma de demonstrar o mecanismo em ação, utilizando toda a estrutura teoricamente idealizada, por isso foi feita essa interface para poder alimentar o sistema.

A aplicação faz uso do modelo salvo para classificar imagens recebidas pelo sistema, sendo possível, na aplicação *mobile*, utilizar diretamente a câmera do dispositivo para fazer a captura da imagem. A aplicação pode dois *inputs* de imagens e classifica cada uma delas individualmente. Estes inputs seriam as imagens obtidas pelos dois módulos de câmeras. Com base nos *inputs*, a aplicação retorna valores 0 para laranjas boas sem manchas, 1 para laranjas ruins e 2 para laranjas boas com manchas. Diante disso, o resultado final fará uso da seguinte lógica:

Se a classificação contar com um retorno 2 e o outro retorno não for 1, a classificação será 2. Se a classificação retornar 1 em qualquer uma das imagens aplicadas, o valor do retorno será consequentemente 1, independente dos demais resultados. A laranja somente será considerada boa sem manchas se ambos os retornos forem iguais a 0.

A seguir são mostradas as imagens da API de classificação dos frutos e as laranjas que foram classificadas na API:



Figura 75 – A API retornando todas as classificações possíveis

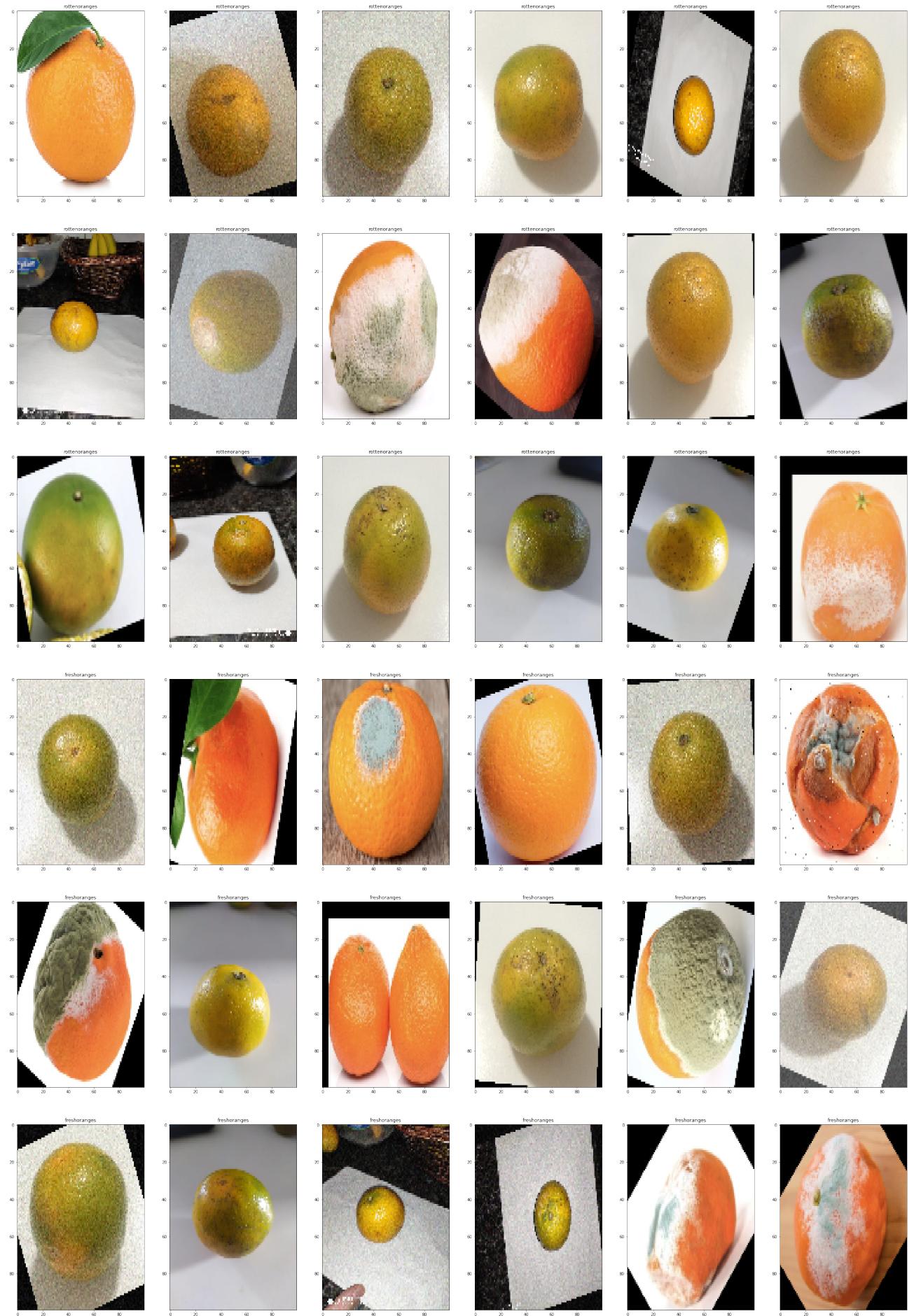


Figura 76 – Imagens classificadas pela API

Internamente, a aplicação faz a coleta dos dados com base nas imagens fornecidas e o resultado destas classificações. Ao receber as imagens dos frutos os seguintes dados são salvos no banco de dados MongoDB: as próprias imagens, a classificação feita pelo algoritmo, a remessa e a data do processamento. Esses dados que são gerados, são utilizados para alimentar as demais partes do software da aplicação, como a Aplicação Web e o Desktop App.

Contudo, além da ligação com os mecanismos de software, também foi utilizado dentro do código *sockets* para realizar a comunicação TCP com o software da parte eletrônica do projeto, enviando as informações de classificação com base nas imagens recebidas. Portanto, é enviado um sinal ao código de eletrônica, que age de acordo com a classificação obtida para movimentar os periféricos.

#### 4.5.2.3 Banco de Dados Local

Para este projeto não há necessidade de um banco de dados relacional, pois não há relacionamentos entre tabelas. Exposto isso, analisou-se três opções de bancos de dados não relacionais (*NoSQL*): Redis, TinyDB e MongoDB. Todas essas três opções possuem prós e contras, os quais serão listados a seguir:

- Redis: é um banco de dados de código aberto que trabalha com estrutura chave-valor. Redis é recomendável para aplicações que necessitam de alta performance, entretanto, essa alta performance consome uma elevada quantidade de memória RAM ([PANDEY, 2017](#)).
- TinyDB: é um banco de dados leve orientado a documentos, otimizado para dispositivos que possuem menores recursos de hardware. Por ser um banco que não tem finalidade de atender grandes aplicações o TinyDB não possui recursos que são necessários para aplicações modernas, como é o caso da concorrência. Para esta aplicação se faz necessário o acesso simultâneo para leitura e modificação de um mesmo documento armazenado no banco, já que uma mudança não pode sobreescriver a outra. O TinyDB não oferece suporte para esse tipo de operação ([TINYDB..., 2016](#)).
- MongoDB: é um banco de dados orientado a documentos, distribuído, criado para atender aplicações modernas e com suporte para a nuvem. O MongoDB possui as propriedades ACID ao realizar transações, com suporte à transações distribuídas de vários documentos com isolamento. O problema de concorrência que o TinyDB possui é sanado utilizando MongoDB ([PANDEY, 2017; KEEP; CABRAL, 2018](#)).

Feita essa breve explanação sobre os bancos de dados em questão e levando em consideração às necessidades do projeto, descarta-se o uso do Redis, devido ao seu alto

consumo de memória RAM, recurso esse que não há em abundância na Raspberry PI 4. O TinyDB também é descartado devido ao problema de concorrência, pois a aplicação utiliza acesso simultâneo ao mesmo documento. Por fim, resta o MongoDB, que é um banco que equilibra performance com robustez, sendo ideal para essa aplicação, possuindo um baixo consumo de recursos e suporte à concorrência na leitura e gravação de documentos.

#### 4.5.2.3.1 Especificações Técnicas

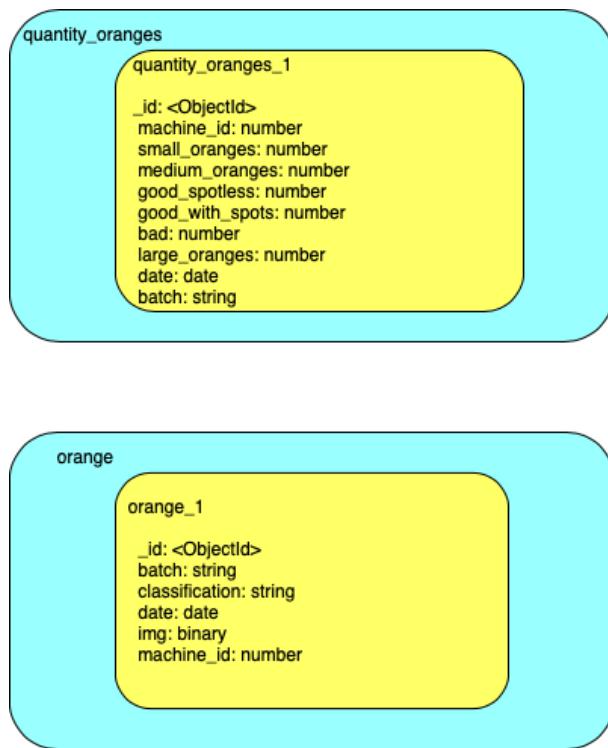


Figura 77 – Representação do banco de dados NoSQL da aplicação Desktop.

A figura 77 apresenta a organização dos dados no MongoDB. Há duas coleções (*collections*), *quantity\_oranges* e *orange*. A coleção *quantity\_oranges* representa a quantidade total de laranjas classificadas em um lote (representado pelo atributo *batch* no banco de dados) de uma determinada data (representada pelo atributo *date* no banco de dados). Essa coleção é necessária para controle de tamanho das laranjas por lote, essa de responsabilidade dos atributos *small\_oranges*, *medium\_oranges* e *large\_oranges*, além de grande importância para a geração de gráficos de tamanho da laranja e de qualidade da laranja, através dos atributos *good\_with\_spots*, *good\_spotless* e *bad*.

A segunda coleção, *oranges*, possui também o atributo *batch* para armazenamento do lote que a laranja integra, além da data que ocorreu o processamento, bem como sua classificação e a foto da laranja. Essa coleção é necessária para o aprendizado constante do algoritmo de *machine learning*, que sempre possuirá fotos novas de laranjas e a classificação do fruto, melhorando constantemente a acurácia do algoritmo.

#### 4.5.2.4 Desktop APP

Para o desenvolvimento da aplicação Desktop, foi ponderado a utilização de tecnologias que conseguissem entregar uma interface gráfica rodando em cima do próprio sistema operacional da placa, o Raspeberry OS. Levando isso em consideração, foi cogitado a escolha das seguintes ferramentas, já apresentado seus pontos negativos e positivos:

- Electron: É um *framework* para criação de aplicativos multiplataformas, utilizando JavaScript, HTML e CSS. Ele permite a construção de aplicações *desktop* utilizando conhecimentos de desenvolvimento *Web*. Dois dos seus pontos positivos é a facilidade em criar interfaces intuitivas utilizando bibliotecas gráficas e o tempo necessário para criar uma aplicação é bem menor quando comparado a outras ferramentas da mesma categoria, como PyQt e linguagens de baixo nível. Sua principal desvantagem é um maior consumo de recursos de *hardware* como memória e processamento, visto que ele utiliza um *browser Web* para executar a aplicação. Nos testes realizados uma aplicação Electron utilizou cerca de 100 MB de memória RAM ([NALEGAVE, 2018](#)).
- PyQt: É uma biblioteca *Python* para o desenvolvimento de aplicações para ambientes em *desktop*. Seu principal ponto positivo é a pouca utilização de recursos computacionais quando comparado a *frameworks* baseados em tecnologia *Web*. Suas desvantagens se encontram na pouca documentação, resultado num processo mais prolongado de aprendizado, e a flexibilidade oferecida pela linguagem *Python* acaba tornando suscetível a criação de códigos sem padrão ([TKINTER..., 2020](#)).
- Tkinter: É a biblioteca padrão que vem junto com o *Python* para desenvolvimento de interfaces gráficas. Entretanto, possui componentes de baixa complexidade, o que impede o uso neste projeto ([TKINTER..., 2020](#)).

Levando em consideração a escassez de recursos de hardware e a necessidade de desenvolvimento acelerado, o Electron se destacou entre as duas opções analisadas, devido ao seu consumo de memória moderado e a menor curva de aprendizado, além da facilidade de comunicação com o banco de dados local MongoDB.

##### 4.5.2.4.1 Especificações Técnicas

O Electron é um *framework* de código aberto para desenvolvimento de aplicações *desktop* utilizando tecnologias *Web* (HTML, CSS e JS). Na figura 78 está representada a arquitetura da aplicação *desktop*, sendo essa uma arquitetura *Model-View-Controller - MVC*, na qual a *model* é responsável pela manipulação de dados, leitura e escrita de dados no banco de dados. A *view* é a camada de interação com o usuário, possuindo interface gráfica para exibição dos dados. Por fim, a *controller* recebe as requisições do usuário,

atuando como um intermediário entre a *model* e a *view*, sendo responsável por controlar qual *model* e por qual *view* será utilizada.

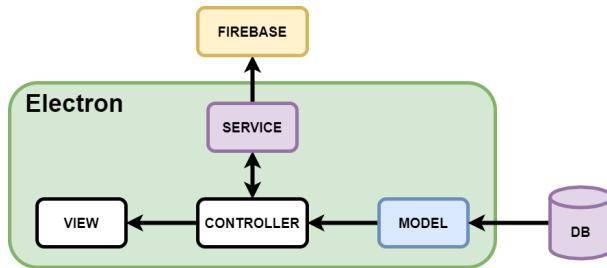


Figura 78 – Diagrama da representação arquitetural da aplicação *desktop*.

Com o diagrama representado na Figura 78, a *model* é responsável pela manipulação dos dados da laranja armazenados no banco de dados local MongoDB, a *controller* possui toda a lógica de negócio da aplicação, fornecendo os dados recebidos da *model* para a *view* apresentar os dados ao usuário. Além disso, a *controller* se comunica com uma *service* que possui um *Web Socket* aberto para comunicação constante com *back-end* como serviço, o Firebase, para sincronizar os dados do banco de dados local com a nuvem.

#### 4.5.2.5 Back-end como Serviço - Firebase

O armazenamento em nuvem se dará por meio da utilização do Firebase, plataforma da Google voltada para o desenvolvimento de aplicações *web* e *mobile*, selecionada para o projeto por conta de ser um *back-end* que atua como serviço, fornecendo serviços como o armazenamento e recuperação de dados em tempo real, além da autenticação completa de usuários. Ou seja, o Firebase une o *back-end* com o banco de dados, além de poupar recursos e esforço com manutenção de infraestrutura em nuvem que seria necessária caso se utilizasse um *back-end* padrão, com banco de dados separado.

Os dados serão armazenados no Cloud Firestore, tecnologia de banco de dados integrada ao Firebase, que consiste em um banco de dados *NoSQL*, orientado a documentos. Nele, serão registrados os dados oriundos do Banco de Dados Local, além dos dados de usuários, que serão consumidos pelo Aplicativo Web para a apresentação das informações a respeito das classificações das laranjas, conforme demonstrado na figura 58.

##### 4.5.2.5.1 Especificações Técnicas

A figura 79 representa como será a comunicação entre os sistemas com o Firebase. O dado que for atualizado no Banco de Dados Local (MongoDB) vai ser enviado em tempo real para o Firebase através da Aplicação Desktop. Após o envio, o Firebase salva essa informação no seu banco de dados, que é o Cloud Firestore. Assim que a informação é salva, o Firebase é responsável por encaminhar a notificação de atualização para o dispositivo de destino e processar a entrega da mesma.

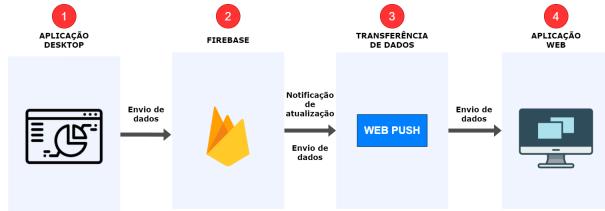


Figura 79 – Representação da utilização do Firebase.

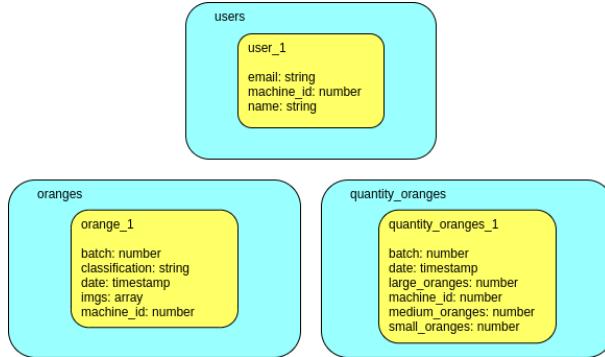


Figura 80 – Representação dos dados no Firestore.

A figura 80 apresenta a organização dos dados no Cloud Firestore. A coleção *users* armazenará os documentos com os dados de cada usuário cadastrado no Aplicativo Web, possibilitando a recuperação do nome, e-mail e ID da máquina cadastrado. Por sua vez, assim como no Banco de Dados Local, as coleções *oranges* e *quantity\_oranges* serão destinadas ao armazenamento de documentos com os dados das classificações das laranjas. A primeira, *oranges*, conterá um documento para cada laranja processada pelo algoritmo da rede neural, possibilitando a recuperação do lote de processamento, da classificação atribuída à laranja pelo algoritmo, da data de processamento, das strings das imagens utilizadas pelo algoritmo para a sua classificação e do ID da máquina associado. Enquanto isso, a segunda, *quantity\_oranges*, conterá um documento para cada lote de processamento, indicando o número do mesmo, a data, a quantidade de laranjas pequenas, médias e grandes classificadas mecanicamente e o ID da máquina associado.

A autenticação de usuários, incluindo cadastro e login, será feita por meio do módulo de autenticação do Firebase, que oferece suporte ao registro de e-mail e senha, além de outros atributos relativos ao perfil de usuário, como nome e foto de perfil, por exemplo. Esse módulo também é responsável por fazer em background todas as validações referentes aos dados do usuário, isentando o cliente desse processo e configurando mais um motivo para a escolha da utilização do Firebase no projeto.

Por isso, mesmo com a autenticação sendo feita por meio do módulo do Firebase, houve a necessidade da criação de uma coleção específica para os usuários dentro do Firestore, representada na figura 80 pela coleção *users*. Isso devido ao fato de que, para cada usuário, deve haver o número de identificação da máquina associado, dado este

que não pode ser guardado no módulo de autenticação e que possibilita a consulta aos documentos corretos dentro das outras coleções presentes no banco, para a apresentação das informações na Aplicação Web.

#### 4.5.2.6 Aplicação Web

A Aplicação Web será feita utilizando o *framework* Vue.js, pois é um *framework* flexível na hora de construir aplicações Web. Este *framework* é capaz de criar interfaces reativas utilizando componentes, além de utilizar o conceito de *Single File Components*, que dá liberdade na hora de criar os componentes, englobando os códigos de HTML, CSS e Javascript em um único arquivo, isso facilita o isolamento e a manutenção de funcionalidades. Ademais, Vue.js tem suporte para *Progressive Web Apps (PWA)*, que são aplicações web responsivas que se adéquam às telas de celulares, funcionando como uma aplicação *mobile* ao mesmo tempo como uma aplicação web comum.

Pelo Vue.js ser um *framework* poderoso na hora de criar aplicações PWA e com uma curva de aprendizado baixa, o grupo optou por utilizar essa tecnologia para a construção da Aplicação Web.

##### 4.5.2.6.1 Especificações Técnicas

A arquitetura é baseada no modelo cliente-servidor, o que configura uma aplicação distribuída, ou seja, na rede existem os fornecedores de recursos ou serviços a rede, que são chamados de servidores, e existem os requerentes dos recursos ou serviços, denominados clientes. No caso do projeto, o cliente é a própria Aplicação Web (feita utilizando o *framework* Vue.js) e o servidor é o Firebase, que fornecem os serviços ao cliente. Além de ter um modelo cliente-servidor, a comunicação é via *Socket*, onde o Firebase (servidor) manda as informações para o cliente sempre que há atualização.

A arquitetura do código em Vue.js utiliza duas ferramentas que auxiliam na construção, gerenciamento e manutenção do código, que são o Vuex e o Nuxt.

O Vuex é um gerenciador de estados da aplicação em Vue.js. Ele funciona da seguinte maneira: É disparada uma *action*, e com isso, há uma mudança no estado através das *mutations* e posteriormente os componentes podem obter os estados através dos *getters*. Tudo isso é denominada a *store* do sistema, a qual gerencia os estados e envia para os componentes que disparam novas *actions*. A escolha pela utilização do Vuex é a facilidade para gerenciar os estados das variáveis da aplicação como um todo. Está representado na Figura 81.

O Nuxt é um *framework* JavaScript, mantido pela comunidade Vue.js, que auxilia na criação de um projeto flexível. Ele pré configura toda a configuração necessária para tornar mais agradável o desenvolvimento, tanto do lado do cliente, como do lado do

servidor, como dados assíncronos, *middleware*, *layouts*, etc. Além disso, ele utiliza do conceito de *SSR* (*Server-Side Rendering*), ou seja, essa abordagem define que toda a renderização de nosso site ocorrerá do lado do servidor, o que torna a aplicação mais rápida, pois quando o *browser* requisitar alguma informação, ela já estará sido montada pelo servidor. Está representado na Figura 81.

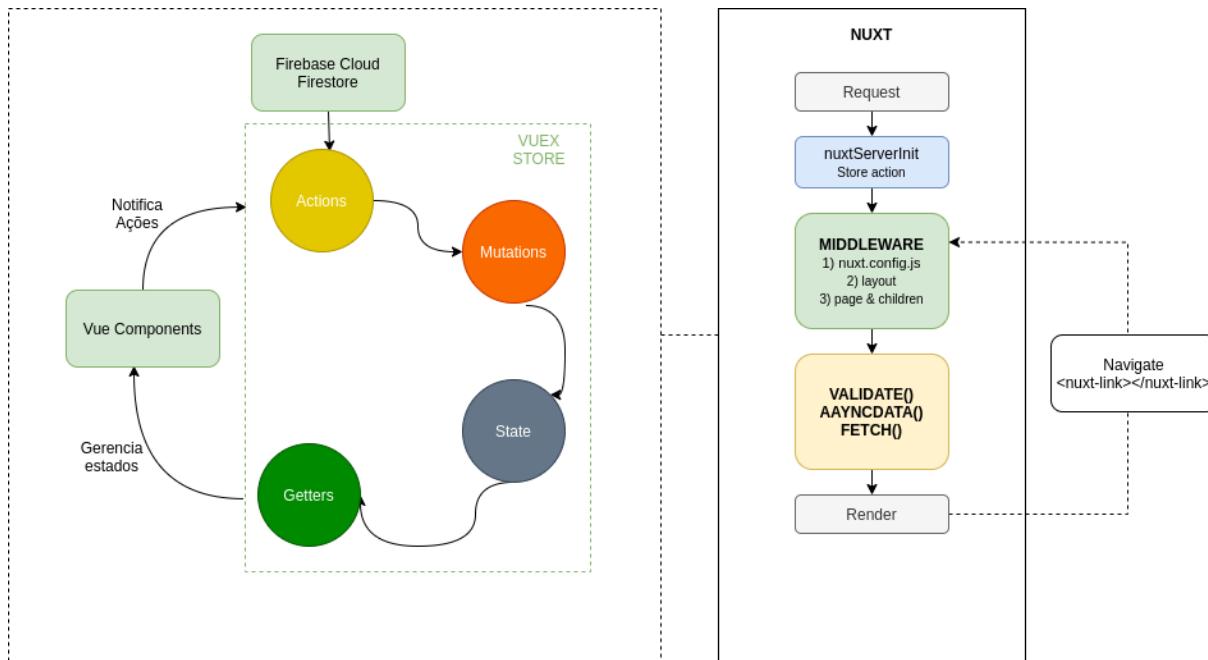


Figura 81 – Representação da arquitetura da aplicação web.

#### 4.5.2.6.2 Configuração do Ambiente de Desenvolvimento

Para isolar o ambiente de desenvolvimento, a fim de evitar problemas relacionados a diferenças em variáveis de ambiente e facilitar a testagem da aplicação localmente, a aplicação foi conteneirizada, ou, em outras palavras, distribuída de maneira compartimentada, portável e autossuficiente (FERNANDES, 2018), por meio de *containers* Docker. Segundo (KARAKAN, 2020), Docker é uma das ferramentas que usou a ideia dos recursos isolados para criar um conjunto de ferramentas que permite que os aplicativos sejam empacotados com todas as dependências instaladas e executadas onde desejar. A ideia de *containers* para a ferramenta é de que eles sejam unidades padrão de software que empacotam o código e todas as suas dependências para que a aplicação seja executada de forma rápida e confiável em um ambiente de computação para outro (DOCKER, 2020). Por fim, para gerenciar os *containers* da aplicação, foi utilizado o Docker Compose, que se trata de uma ferramenta simples, mas poderosa, usada para executar múltiplos *containers* como um único serviço (ZAMAN, 2019).

#### 4.5.2.6.3 Implantação

Para a disponibilização da aplicação online, utilizamos o serviço de computação em nuvem oferecido pelo Heroku. A ferramenta disponibiliza o que é chamado de plataforma como serviço (*PaaS*), que vem ganhando espaço no mercado por, segundo (STORI, 2013), abstrair o desenvolvedor dos detalhes de infraestrutura ao disponibilizar *containers* para instalação das aplicações, facilitando a manutenção, extensão e escalabilidade, além de oferecer maior agilidade para disponibilizar uma aplicação na web, com menor custo inicial.

## 4.6 Inovação

Na parte do projeto que cabe a Software, pode-se considerar inovador em relação à pilha de tecnologias definidas para compor a arquitetura do sistema de forma geral a utilização do processamento de imagens e classificação das laranjas por meio de uma Rede Neural, que aplica algoritmos de aprendizado de máquina (*Machine Learning*) para fazer a seleção de cada uma das laranjas utilizando os dados das imagens recebidas.

O Aprendizado de Máquina vem ganhando força na área da Engenharia e da Ciência da Computação e, embora ainda não esteja presente na grande maioria dos sistemas convencionais, está sendo cada vez mais disseminado em decorrência do grande volume de dados gerados atualmente. Segundo (GOMES, 2019), com a crescente quantidade de dados disponíveis, há boas razões para acreditar que a análise inteligente de dados se tornará cada vez mais difundida como um ingrediente necessário para o progresso tecnológico.

Dessa forma, o Aprendizado de Máquina utilizado no projeto configura um passo a mais em relação ao estado atual da maioria dos sistemas presentes no mercado, ao mesmo tempo que se mostra ser o futuro da tecnologia da informação, se tornando cada vez mais indispensável quando se pretende alcançar resultados singulares com o emprego da tecnologia.

Além do Aprendizado de Máquina, toda a pilha da aplicação Web apresenta elementos inovadores, como o uso do Firebase atuando como um *back-end* como serviço (*Backend-as-a-Service - BaaS*), substituindo um *back-end* comum às aplicações que possuem arquitetura cliente-servidor, mantendo os serviços do Firebase e de armazenamentos em um só local. Além do uso do *BaaS*, outro elemento inovador do projeto é o uso de *Progressive Web App - PWA*, possibilitando desenvolver uma única aplicação que poderá ser renderizada em computadores, *tablets* e celulares de forma responsiva em cada um desses dispositivos.

## 4.7 Pilha de Tecnologias Utilizadas

Tabela 28 – Pilha de Tecnologias

NOME	ENDEREÇO	DESCRIÇÃO
Keras	<a href="https://keras.io/">https://keras.io/</a>	Biblioteca de aprendizado de máquina utilizada na Rede Neural
MongoDB	<a href="https://www.mongodb.com/">https://www.mongodb.com/</a>	Banco de dados orientado a documentos utilizado pelos sistemas locais da máquina
Electron	<a href="https://www.electronjs.org/docs">https://www.electronjs.org/docs</a>	Framework JavaScript utilizado no desenvolvimento do Desktop APP
Firebase	<a href="https://firebase.google.com/">https://firebase.google.com/</a>	Plataforma de <i>back-end</i> como serviço utilizada para sincronização de dados na nuvem
Vue.js	<a href="https://vuejs.org/">https://vuejs.org/</a>	Framework <i>JavaScript</i> utilizado no desenvolvimento da Aplicação Web
VueX	<a href="https://vuex.vuejs.org/">https://vuex.vuejs.org/</a>	Gerenciador de estados da Aplicação Web
Nuxt	<a href="https://nuxtjs.org/">https://nuxtjs.org/</a>	Framework <i>JavaScript</i> que auxilia no desenvolvimento da Aplicação Web
Docker e Docker Compose	<a href="https://www.docker.com/">https://www.docker.com/</a>	Ferramentas utilizadas na conteinerização da aplicação web.
Heroku	<a href="https://www.heroku.com/">https://www.heroku.com/</a>	Ferramenta de <i>PaaS</i> utilizada para a implantação da aplicação web.

## 4.8 Integração Eletrônica e Software

Esta seção tem como objetivo explicar como foi realizada a integração das soluções de *software* e eletrônica, pois existe a transferência de dados entre estes dois módulos. O

algoritmo de Machine Learning que faz a classificação de imagens por meio de processamento de imagens precisa enviar os dados dessa classificação para o Algoritmo de Rotas para que seja realizado o controle dos servos. A contagem feita por sensores controlado pelo Algoritmo de Contagem precisa enviar os dados para que seja armazenado no banco de dados local, MongoDB.

#### 4.8.1 Integração do Algoritmo de Rotas com o Machine Learning

A comunicação entre o Algoritmo de Rotas e o algoritmo de *Machine Learning* é necessária para que exista a troca de informação sobre a classificação do fruto entre os dois algoritmos. O algoritmo de *Machine Learning* após classificar as imagens recebidas pelos módulos de câmera precisa informar a classificação para o Algoritmo de Rotas para que este controle os servos.

A comunicação foi realizada por meio de *socket* do Python, que é utilizado para enviar mensagens através de uma rede. Ele fornece uma comunicação entre processos. Foi utilizado o protocolo padrão, o Protocolo de Controle de Transmissão ([SOCKET..., 2020](#)).

Foi criada uma API cliente-servidor utilizando o *socket* do Phyton. O servidor se encontra no Algoritmo de Rotas, que fica em espera de alguma mensagem enviada pelo cliente que foi utilizado no Algoritmo de *Machine Learning* para enviar a classificação.

Os algoritmos estão evidenciados e detalhados nas figuras 82 e 83. A figura 82 mostra melhor como é criado um servidor utilizando *socket* do Python. Já a figura 83 mostra como é criado um cliente do *socket* para envio de mensagens.

```
# Criação de um objeto socket
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    # Associa um socket a uma interface de rede
    s.bind((HOST, PORT))
    # Permite que o servidor aceite conexões
    s.listen()
    # Aceita a conexão
    conn, addr = s.accept()
    with conn:
        while True:
            # Recebe a mensagem do cliente
            data = conn.recv(1024)
```

Figura 82 – Algoritmo do servidor do socket.

```
# Cria um objeto socket
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    # Conecta ao servidor
    s.connect((HOST, PORT))
    # Envia a mensagem - LABEL ('BOA COM MANCHAS', 'BOA SEM MANCHAS', 'RUIM')
    s.sendall(str.encode(LABEL))
    # Lê a resposta do servidor
    data = s.recv(1024)
```

Figura 83 – Algoritmo do cliente do socket.

### 4.8.2 Integração do Algoritmo de Contagem com o Desktop App e o Banco de Dados MongoDB

O algoritmo de contagem é responsável por coletar os dados dos sensores que fazem a classificação por tamanho da laranja, portanto, se torna imprescindível a comunicação com os sistemas de software. Para essa comunicação acontecer, foi necessário implementar uma conexão com o banco de dados local MongoDB, com o fim de persistir no disco rígido os dados da contagem de laranja por tamanho e posteriormente enviar para o banco de dados em nuvem.

A conexão com o banco de dados local MongoDB é a ponte entre o subsistema de eletrônica e de software, após essa conexão ser estabelecida, é feita uma busca pelo último documento inserido na coleção *quantity\_oranges* presente banco de dados local, quando esse documento é recuperado, é realizada a recuperação do lote no qual o documento recuperado anteriormente pertence, sendo esse lote o *id* da sessão do MongoDB em que o documento foi inserido na coleção. Após a recuperação do lote, é feita atualização do documento recuperado, armazenamento dos valores de contagem nos campos *small\_oranges*, *medium\_oranges* e *large\_oranges*.

```
# Conecta com o banco de dados (mongodb) local
cliente_prod = MongoClient('mongodb://localhost:27017/orange_classification')
db_prod      = cliente_prod.orange_classification
# Estabelece conexão com a collection quantity_oranges
coll_quantity = db_prod.quantity_oranges
# Recupera o último documento atualizado pelo algoritmo de Machine Learning
ultimo_quantity_oranges = coll_quantity.find({}).sort('_id', -1).limit(1)
# Recupera o número do lote do último documento atualizado pelo algoritmo de Machine Learning
for doc in ultimo_quantity_oranges:
    lote = doc['batch']
#Atualizando a contagem e armazenando os valores no banco de dados
if distancia_pequena <= 50 and distancia_pequena > 0:
    laranja_pequena = laranja_pequena + 1
    coll_quantity.update_one({ 'batch': lote }, {"$set": { "small_oranges": laranja_pequena }})
if distancia_media <= 50 and distancia_media > 0:
    laranja_media = laranja_media + 1
    coll_quantity.update_one({ 'batch': lote }, {"$set": { "medium_oranges": laranja_media }})
if distancia_grande <= 50 and distancia_grande >0:
    laranja_grande = laranja_grande + 1
    coll_quantity.update_one({ 'batch': lote }, {"$set": { "large_oranges": laranja_grande }})
```

Figura 84 – Conexão entre o algoritmo de contagem e banco de dados local MongoDB.

## 4.9 Diagrama de Integração

O diagrama apresentado na imagem abaixo apresenta a integração entre as soluções propostas por cada engenharia.

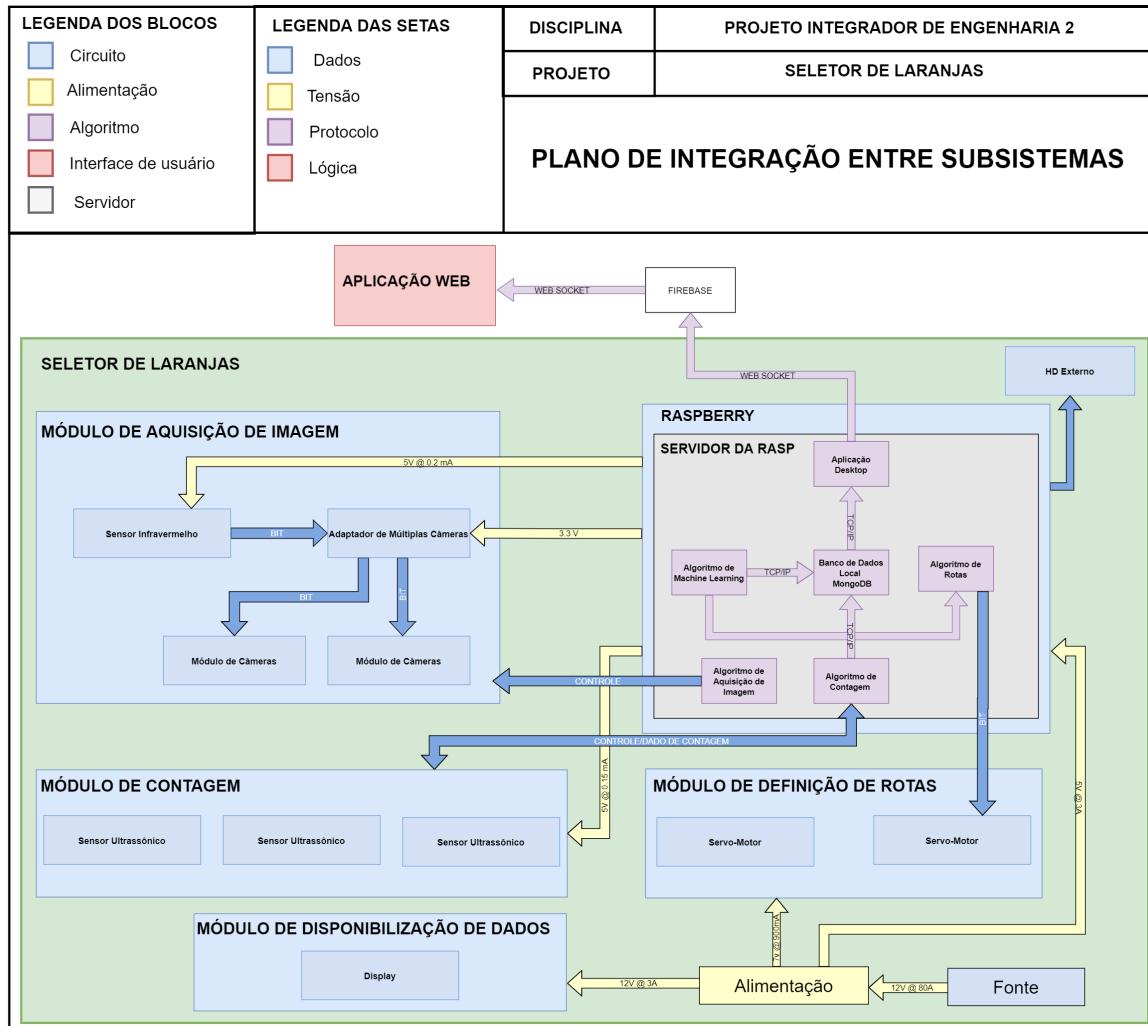


Figura 85 – Diagrama de Integração.

# 5 Custos

## 5.1 Orçamento do Projeto

### 5.1.1 Custos Software

Quantidade	Componente	Custo Unitário	Custo Total	Fornecedor
1	Hospedagem Aplicação Web	R\$ R\$178,24 mensais	R\$ 356,48	Google Cloud

Tabela 29 – Custos Software.

### 5.1.2 Custos Eletrônica

Quantidade	Componente	Custo Unitário	Custo Total	Fornecedor
1	Raspberry PI 4	R\$ 800,00	R\$ 800,00	Mercado Livre
2	Servo Motor TowePro MG-996R	R\$ 52,58	R\$ 105,16	HU Infinito
3	Sensor Ultrassônico HC-sr04	R\$ 10,50	R\$ 31,50	HU Infinito
2	Módulo Câmera 5mp	R\$ 44,99	R\$ 89,98	Mercado Livre
1	Adaptador de Múltiplas Câmeras	R\$ 269,44	R\$ 269,44	Amazon
1	Módulo Sensor Infravermelho	R\$ 12,90	R\$ 12,90	Mercado Livre

Tabela 30 – Custos Eletrônica.

### 5.1.3 Custos Estrutura/Energia

Quantidade	Componente	Custo Unitário	Custo Total	Fornecedor
<b>1</b>	Tubo PVC	R\$ 130,00	R\$ 130,00	Mercado Livre
<b>1</b>	Conexão de PVC	R\$ 98,99	R\$ 98,99	Mercado Livre
<b>1</b>	Pacote parafuso sextavado	R\$ 167,00	R\$ 167,00	Mercado Livre
<b>5</b>	Chapa de aço inox	R\$ 189,00	R\$ 945,00	Mercado Livre
<b>1</b>	Tubo PVC	R\$ 130,00	R\$ 130,00	Mercado Livre
<b>40</b>	Rolamentos	R\$ 14,12	R\$ 564,80	Mercado Livre
<b>20</b>	Rolete	R\$ 41,00	R\$ 820,00	Mercado Livre
<b>20</b>	Relação de moto	R\$ 45,00	R\$ 900,00	Mercado Livre
<b>10</b>	Viga galvanizada	R\$ 82,33	R\$ 823,30	Aladim Metais
<b>3</b>	Motor redutor esteira	R\$ 250,00	R\$ 750,00	Mercado Livre
<b>1</b>	Bomba d'água 19W 12V	R\$ 103,09	R\$ 103,09	Mercado Livre
<b>1</b>	Tubo PVC	R\$ 130,00	R\$ 130,00	Mercado Livre
<b>4</b>	Painel solar 355W	R\$ 725,00	R\$ 2900,00	Energia Total
<b>1</b>	Ventoinha	R\$ 269,90	R\$ 269,90	Mercado Livre
<b>1</b>	Tubo PVC	R\$ 130,00	R\$ 130,00	Mercado Livre
<b>4</b>	Bateria 12V 100Ah	R\$ 365,00	R\$ 1460,00	Mercado Livre
<b>1</b>	Tubo PVC	R\$ 130,00	R\$ 130,00	Mercado Livre
<b>4</b>	Tarugo barra aço inox Aisi 304	R\$ 40,00	R\$ 160,00	Mercado Livre
<b>1</b>	Estrutura solar para 4 Placas - solo	R\$ 999,00	R\$ 999,00	Energy Shop
<b>4</b>	Fusíveis GPV 15A	R\$ 47,00	R\$ 188,00	Energia Total

Tabela 31 – Custos Estrutura/Energia

## 6 Considerações Finais

A partir do desenvolvimento de pesquisas e a definição da arquitetura para implementação do projeto, foi possível notar a viabilidade do seletor de laranjas. Além disso, esta solução possui potencial comercial e tecnológico devido ao seu custo e aos resultados das simulações realizadas.

A utilização de tecnologias na agricultura já é extremamente difundida no mercado. No entanto, o custo é alto e sua aplicação é feita apenas por grandes produtores rurais. Dessa forma, o projeto seletor de laranjas pode ser comercializado e utilizado por pequenos e médios produtores.

# Referências

A importância da Arquitetura da Informação. 2020. <<https://jera.com.br/blog/5403/design-ux/a-importancia-da-arquitetura-da-informacao>>. Citado na página 97.

ABBAS, H. M. T. Automated sorting and grading of agricultural products based on image processing. Paquistão, 2019. Citado na página 17.

ARDUCAM. Arducam Multi Camera Adapter Module V2.2 for Raspberry Pi 4 B. 2019. <<https://www.arducam.com/product/multi-camera-v2-1-adapter-raspberry-pi/>>. Citado na página 73.

ARDUCAM. Multi Camera Adapter Module for Raspberry Pi. 2020. <<https://www.arducam.com/multi-camera-adapter-module-raspberry-pi/>>. Citado na página 73.

ATHOSELETRONICS. Sensor infravermelho de obstáculo reflexivo e Arduino. 2017. <<https://athoselectronics.com/sensor-infravermelho-de-obstaculo-reflexivo-e-arduino/>>. Citado na página 74.

BRITO, C. . D. P. D. E. E. C. . D. R. P. E. S. E. E. S. *Manual de Engenharia para Sistemas Fotovoltaicos*. Rio de Janeiro, 2014. Citado na página 54.

BUDYNAS, R. G.; NISBETT, J. K. et al. *Shigley's mechanical engineering design*. [S.l.]: McGraw-Hill New York, 2008. v. 8. Citado na página 45.

CARTILHAS de Frutas: Laranja. 2016. <<http://www.faep.com.br/comissoes/frutas/cartilhas/frutas/laranja.htm>>. Citado na página 41.

CAVALCANTI K. C. OLIVEIRA, G. M. S. A. F. A. S. N. M. C. Comparative study of maximum power point tracking techniques for photovoltaic systems. Brazilian Journal of Power Electronics, v. 12, 2007. Citado na página 53.

CEAGESP. *MANUAL DE CLASSIFICACAO DE CITROS DE MESA*. São Paulo, Brasil, 2011. Citado 3 vezes nas páginas 2, 13 e 14.

DATASET. 2020. <<https://www.kaggle.com/jppgomes/dataset-laranjas>>. Citado na página 115.

DATASHEET: Sensor Ultrassônico HC-SR04. 2020. <<https://www.alldatasheet.com/datasheet-pdf/pdf/1132203/ETC2/HC-SR04.html>>. Citado na página 82.

DATASHEET. Servo motor. 2019. <<https://www.alldatasheet.com/datasheet-pdf/pdf/1131873/ETC2/MG996R.html>>. Citado na página 78.

DAZCAL R.; MELLO, A. Estudo da implementação de um sistema de energia solar fotovoltaica em um edifício da universidade presbiteriana mackenzie. Abenge –Associação Brasileira de Educação de Engenharia, Fortaleza, 2008. Citado na página 55.

DESCRIÇÃO autocad. 2020. <<https://www.autodesk.com.br/products/autocad/overview?plc=ACDIST&term=1-YEAR&support=ADVANCED&quantity=1>>. Citado na página 65.

- DOCKER. *What is a Container?* 2020. <<https://www.docker.com/resources/what-container>>. Citado na página 124.
- DOCUMENTAÇÃO do Keras. <<https://keras.io/about/>>. Citado na página 111.
- EURODRIVE, S. Catálogo de motoredutores. *SEW Eurodrive. São Paulo*, 2008. Citado 3 vezes nas páginas 43, 44 e 45.
- FERNANDES, A. *O que é Conteinerização de aplicação?* 2018. <<https://vertigo.com.br/o-que-e-conteinerizacao-de-aplicacao/>>. Citado na página 124.
- GNU LINUX BRASIL. 10 sistemas operacionais que você pode executar no Raspberry Pi em 2019. 2019. <<https://gnulinuxbrasil.com.br/2019/05/02/10-sistemas-operacionais-que-voce-pode-executar-no-raspberry-pi-em-2019/>>. Citado na página 71.
- GOMES, P. C. T. *Aprendizado de Máquina*. 2019. <<https://www.datageeks.com.br/aprendizado-de-maquina/>>. Citado na página 125.
- GRENN, M. Photovoltaics: technology overview. *Energy Policy*, v. 28, 2000. Citado na página 53.
- KARAKAN, B. *What exactly is Docker?* 2020. <<https://medium.com/swlh/what-exactly-is-docker-1dd62e1fde38>>. Citado na página 124.
- KEEP, M.; CABRAL, A. *MongoDB 4 Update: Multi-Document ACID Transactions*. 2018. <<https://www.mongodb.com/blog/post/mongodb-multi-document-acid-transactions-general-availability>>. Citado 2 vezes nas páginas 118 e 146.
- LAMB, F. *Automação Industrial na Prática-Série Tekne*. [S.l.]: AMGH Editora, 2015. Citado na página 41.
- MACHADO C.; MIRANDA, F. Energia solar fotovoltaica: Uma breve revisão. Revista virtual de química, Niterói, RJ, v. 7, 2014. Citado na página 54.
- MASOUM SEYED MAHDI MOUSAVI BADEJANI, E. F. F. M. S. Microprocessor-controlled new class of optimal battery chargers for photovoltaic applications. *IEEE Transactions on energy conversion*, v. 19, 2004. Citado na página 53.
- MASTERWALKER. Como usar com Arduino – Sensor Infravermelho Reflexivo de Obstáculo. 2020. <<https://blogmasterwalkershop.com.br/arduino/como-usar-com-arduino-sensor-infravermelho-reflexivo-de-obstaculo/>>. Citado na página 75.
- MERCADO LIVRE. Hd Externo 120gb - Usb 3.0. 2020. <[https://produto.mercadolivre.com.br/MLB-1645500319-hd-externo-120gb-usb-30-novo-e-completo-um-luxo-\\_JM?matt\\_tool=90090532&matt\\_word=&matt\\_source=google&matt\\_campaign\\_id=6542484841&matt\\_ad\\_group\\_id=84209438291&matt\\_match\\_type=&matt\\_network=u&matt\\_device=c&matt\\_creative=385102491763&matt\\_keyword=&matt\\_ad\\_position=&matt\\_ad\\_type=&matt\\_merchant\\_id=246755736&matt\\_product\\_id=MLB1645500319&matt\\_product\\_partition\\_id=342889296875&matt\\_target\\_id=pla-342889296875&glcid=CjwKCAjwlbr8BRA0EiwAnt4MTo4w8u9SYYG\\_xclzaYR3gbFezxW66711eeowhM-XQ7tom-OwzvqShoCRKgQAvD\\_BwE](https://produto.mercadolivre.com.br/MLB-1645500319-hd-externo-120gb-usb-30-novo-e-completo-um-luxo-_JM?matt_tool=90090532&matt_word=&matt_source=google&matt_campaign_id=6542484841&matt_ad_group_id=84209438291&matt_match_type=&matt_network=u&matt_device=c&matt_creative=385102491763&matt_keyword=&matt_ad_position=&matt_ad_type=&matt_merchant_id=246755736&matt_product_id=MLB1645500319&matt_product_partition_id=342889296875&matt_target_id=pla-342889296875&glcid=CjwKCAjwlbr8BRA0EiwAnt4MTo4w8u9SYYG_xclzaYR3gbFezxW66711eeowhM-XQ7tom-OwzvqShoCRKgQAvD_BwE)>. Citado na página 85.

MESSENGER R.; VENTRE, J. Photovoltaic systems engineering. CRC Press, Boca Raton, 2010. Citado na página 55.

MORAIS DOUGLAS; RODRIGUES, D. L. P. D. C. F. C. Ondas ultrassônicas: teoria e aplicações industriais em ensaios não-destrutivos. 2017. Citado na página 82.

MOTT, R. L. Elementos de maquinas em projetos mecânicos. São Paulo, 2015. Citado na página 43.

MOTTA, J. Projeto elétrico da instalação de um sistema fotovoltaico no prédio ued. Brasília, DF, 2019. Citado na página 55.

NALEGAVE, S. *Electron / Pros And Cons*. 2018. <<https://medium.com/@nalegaveshardul40/electron-pros-and-cons-8f58fd6313d5>>. Citado na página 120.

NBR 16690. 2020. <<https://www.solarize.com.br/downloads/manual-energia-solar/NBR-16690-2019-consulta-publica.pdf>>. Citado na página 63.

NBR 5410. 2020. <<https://docente.ifrn.edu.br/jeangaldino/disciplinas/2015.1/installacoes-elettricas/nbr-5410>>. Citado na página 63.

NEVES, M. F. O retrato da citricultura brasileira. Brasil, 2008. Citado na página 18.

PANDEY, V. *Redis vs. MongoDB: Comparing In-Memory Databases with Percona Memory Engine*. 2017. <<https://scalegrid.io/blog/comparing-in-memory-databases-redis-vs-mongodb-percona-memory-engine/>>. Citado 2 vezes nas páginas 118 e 145.

PEREIRA F.; OLIVEIRA, M. Curso técnico instalador de energia solar fotovoltaica. Porto, 2011. Citado na página 54.

RASPBERRY Pi 4. 2020. <<https://www.filipeflop.com/produto/raspberry-pi-4-model-2gb-4gb>>. Citado na página 71.

RASPBERRY PI. Raspberry Pi High Quality Camera. 2019. <<https://www.raspberrypi.org/products/raspberry-pi-high-quality-camera>>. Citado na página 73.

RUDENKO, I. *Máquinas de elevação e transporte*. [S.l.]: Livros Técnicos e Científicos, 1976. Citado na página 41.

RUTHER, R. Edifícios solares fotovoltaicos: o potencial de geração solar fotovoltaica integrada a edificações urbanas e interligadas a rede elétrica pública no brasil. Labsolar, Florianópolis, SC, 2004. Citado na página 54.

SARAVATI. Módulo de Câmera 5mp Raspberry PI. 2019. <<https://www.saravati.com.br/modulo-de-camera-5mp-raspberry-pi>>. Citado na página 73.

SENSOR de obstáculos reflexivo infravermelho. 2020. <[https://produto.mercadolivre.com.br/MLB-689768524-sensor-de-obstaculo-reflexivo-infravermelho-ir-arduino-pic-\\_JM?matt\\_tool=79246729&matt\\_word=&gclid=Cj0KCQjwtZH7BRDzARIIsAGjbK2ZCs\\_yVuZFJskm1oNY8rqn3P9BcbGkBM4VWB6yPCMC6na6kfgse5oaAgeCEALw\\_wcB](https://produto.mercadolivre.com.br/MLB-689768524-sensor-de-obstaculo-reflexivo-infravermelho-ir-arduino-pic-_JM?matt_tool=79246729&matt_word=&gclid=Cj0KCQjwtZH7BRDzARIIsAGjbK2ZCs_yVuZFJskm1oNY8rqn3P9BcbGkBM4VWB6yPCMC6na6kfgse5oaAgeCEALw_wcB)>. Citado na página 76.

SERVO Motor.Veja como Funciona e Quais os Tipos. 2019. <<https://www.citisystems.com.br/servo-motor/>>. Citado na página 78.

SOCKET Programming in Python (Guide). 2020. <<https://docs.python.org/3/howto/sockets.html>>. Citado na página 127.

STORI, D. “*Primeiros passos em PaaS com Heroku*. 2013. <<https://www.devmedia.com.br/primeiros-passos-em-paas-com-heroku/29465>>. Citado na página 125.

SUNFOUNDER.MONITOR Display for Raspberry Pi 4B. 2020. <<https://www.sunfounder.com/10-1-inch-hdmi-lcd.html>>. Citado 2 vezes nas páginas 83 e 84.

TINYDB: Why Not Use TinyDB? 2016. <<https://tinydb.readthedocs.io/en/stable/intro.html#why-not-use-tinydb>>. Citado 2 vezes nas páginas 118 e 145.

TKINTER vs PYQT. 2020. <[https://www.slant.co/versus/16724/22768/~tkinter\\_vs\\_pyqt](https://www.slant.co/versus/16724/22768/~tkinter_vs_pyqt)>. Citado na página 120.

VILLALVA M.; GAZOLI, J. Energia solar fotovoltaica: conceitos e aplicações. São Paulo, 2012. Citado na página 54.

ZAMAN, T. “*What, Why, How” Docker Compose?* 2019. <<https://medium.com/faun/what-why-how-docker-compose-482e8aa021b>>. Citado na página 124.

# Apêndices

# APÊNDICE A – Manual de Operação

## Seletor de Laranjas



### 1. Ligar a máquina

A máquina precisa estar conectada a energia e um cabo de ethernet rj45.



### 2. Ligar a torneira local

Para melhor classificação, o sistema de limpeza de frutos é necessário.



### 3. Verificar se os containers de armazenamento estão posicionados nas saídas corretas

Os frutos serão armazenados separadamente de acordo com a classificação



### 4. Colocar as laranjas no container de entrada

Os frutos que serão classificados deverão estar na máquina para início da seleção.



### 5. Acompanhar os resultados da classificação e desligar a máquina ao final do uso

Os resultados da classificação serão disponibilizados por meio de gráficos.

**SELETOR DE LARANJAS**

PROJETO INTEGRADOR 2

Figura 86 – Manual de Operação do Seletor de Laranjas

# APÊNDICE B – Perguntas e Respostas do Questionário

Tabela 32 – Perguntas e Respostas do Questionário

PERGUNTA	RESPOSTA		
Qual a sua identificação?	R1	-	
	R2	-	
	R3	Cláudio - Fazenda Marfim	
	R4	José Luiz	
Com qual produto agrícola você trabalha hoje?	R1	Limão	
	R2	Laranja e milho.	
	R3	Laranja	
	R4	Mamão	
Qual a sua escala de produção?	R1	Pequena	
	R2	Média	
	R3	Grande	
	R4	Média	
Qual é o processo pós colheita do seu produto?	R1	Benefício em máquina de seleção	
	R2	Venda do mesmo.	
	R3	As laranjas são classificadas com o auxílio de uma máquina de classificação. As menores e mais manchadas são enviadas para o processamento industrial (Granel). As outras, que irão para o mercado interno da região, passam por uma máquina onde são lavadas, enceradas e polidas para serem, em seguida, en-sacadas (em sacos de 20kg).	
	R4	-	
Continua na próxima página			

**Tabela 32 – continuação da página anterior**

PERGUNTA	RESPOSTA		
Quais as formas de classificação você utiliza para separar os produtos bons para comercialização daqueles que não podem ser comercializados?	R1	Limão de primeira, segunda e o danificado é descartado.	
	R2	Nenhuma classificação	
	R3	Na própria colheita, feita manualmente, os trabalhadores são orientados a identificar as frutas em um bom estado de maturação. Depois disso, as laranjas colhidas passam por uma máquina de classificação por tamanho e quantidade de manchas.	
	R4	É feito a mão	
Existe algum tipo de classificação intermediária para esse produto (além de bom ou ruim)? Se sim, qual(is)?	R1	Não	
	R2	Sim, visualmente apenas.	
	R3	Tamanho, manchas e maturação	
	R4	Não ou está bom ou ruim	
Qual a sua maior necessidade ao fazer a seleção dos produtos? (Ex: padronização das frutas, evitar que frutas ruins estraguem as boas, etc.)	R1	Padronização	
	R2	Não é feita a seleção.	
	R3	Padronizar as frutas. Para a comercialização em mercados (in natura), elas devem ser maiores, estar em um bom estado de maturação, além de estar polidas e enceradas.	
	R4	Sim seria ótimo padronizar	
Como é feito o descarte dos produtos estragados?	R1	Vai descartado para o lixo	
	R2	Não descartamos, a empresa que compra que descarta.	
	R3	São poucas as laranjas descartadas. As menores e mais manchadas são mandadas para a indústria para serem processadas.	
	R4	Levamos para os animais	
Você acha interessante a aquisição de uma máquina que automatizaria o processo de seleção, fazendo a distinção entre os produtos bons e estragados da sua produção? Se sim, por quê?	R1	Sim, agrupa valor a mercadoria	
	R2	Eu não faço seleção.	
	R3	Sim! A automatização da classificação das laranjas (por tamanho e quantidade de manchas) é, hoje, essencial no processo como um todo.	
	R4	Sim seria ótimo facilitaria o trabalho e ganhariámos mais tempo	

Continua na próxima página

**Tabela 32 – continuação da página anterior**

PERGUNTA	RESPOSTA		
A visualização dos dados da colheita (quantidade total de produtos, quantidade de produtos bons e ruins, etc.) te beneficiaria? Seria interessante a visualização de estatísticas e gráficos a partir desses dados?	R1	Beneficiaria muito, pois através destes dados eu teria uma noção de como está a produtividade e sanidade da lavoura.	
	R2	Não	
	R3	Como a máquina que faz a classificação não dá nenhum tipo de informação, esse trabalho de geração de relatórios é feito manualmente pelos funcionários encarregados (eles contabilizam, normalmente, pelo peso das remessas). Então, seria interessante que essas informações fossem geradas automaticamente.	
	R4	Sim	
Existe acesso à internet no local onde acontece sua produção?	R1	Sim	
	R2	Não	
	R3	Sim	
	R4	Não	

# APÊNDICE C – Decisões Importantes e Aprendizados (Software)

A seguir, está apresentada uma lista de tópicos discutidos entre a equipe de Software, que configuraram decisões importantes a respeito do projeto/produto e lições aprendidas para os membros da equipe.

## 1. Apresentação dos dados da colheita

Visto que a conexão com a Internet é um dos fatores com os quais devemos lidar, apresentaram-se duas opções quanto à exibição de informações da colheita ao produtor:

- Com conexão à internet - Duas aplicações na camada de apresentação dos dados, uma aplicação *desktop* e outra aplicação Web. A aplicação *desktop* apresentará os dados localmente e enviará os dados para o Firestore, enquanto a aplicação Web disponibilizaria funções mais avançadas de visualização dos dados, sendo disponibilizada também na versão PWA (Progressive Web App). Neste caso, adiciona-se ao projeto a complexidade de detecção da conexão com a Internet, além da sincronização dos dados do banco de dados local (NoSQL) com o banco de dados da nuvem (Firestore).
- Sem conexão à internet - Uma aplicação desktop para apresentação dos dados referentes à colheita atual. Os dados são armazenados em banco de dados local NoSQL, para caso um dia o usuário tenha conexão com a internet, seus dados serão sincronizados com a nuvem.

Ao analisar a situação e, levando em conta a fase de problematização e levantamento de requisitos, vimos a necessidade de geração de dados mais elaborados acerca da colheita, além da automatização desse processo, uma vez que os produtores entrevistados na primeira fase descreveram uma situação na qual eles não têm acesso a tais informações de forma automatizada. Além disso, a conexão com a Internet já é uma realidade mais presente do que se imaginava. Dos produtores entrevistados, 50% deles possuem acesso à Internet nas suas propriedades.

## 2. Escolha das tecnologias de banco de dados

Em relação à escolha de qual banco de dados seria utilizado no microcomputador (Raspberry PI 4, Figura ??), apresentou-se a problemática da definição de qual tipo de implementação de banco de dados seria adotado. A princípio, nós optaríamos por

uma solução com banco de dados local SQL (o SQLite), por ser um banco de dados já pensado para aplicações que rodam localmente e ser leve. No entanto, levando em consideração a necessidade de sincronização dos dados locais com os dados em nuvem (caso haja conexão com a Internet), decidiu-se optar por uma solução com banco de dados NoSQL, por conta do suporte para transmissão de informações em tempo real e para evitar maiores consumos de hardware, convertendo dados relacionais para não relacionais, caso fosse utilizada uma combinação com um banco de dados SQL no microcomputador sincronizando com um banco de dados NoSQL na nuvem. Foi decidido que será utilizado o banco de dados que dados em nuvem chamado Firebase Cloud Firestore em conjunto com um banco de dados local NoSQL. A escolha por usar um banco de dados NoSQL na nuvem se deu por conta do Firestore ser um banco de dados que atua como um Backend Como Serviço (BaaS), que funciona basicamente com um banco de dados em nuvem onde todo o tratamento de dados é feito nesta mesma camada, sendo ideal para projetos onde há separação de usuários através de logins e dados não relacionais de propriedade desse usuário. Como o Firestore é um banco de dados orientado a documentos e não relacional, foi, então, definido que o banco de dados utilizado na Raspberry PI 4 (Figura ??) será NoSQL, ou seja, não relacional, para que a transmissão de informações deste banco para o Firebase seja feita de forma mais rápida e exija menos do hardware da placa ao não exigir conversão de dados.

### 3. Treinamento da Rede Neural fora da Raspberry

Foi decidido que o treinamento da Rede Neural será fora da Raspberry PI 4 (Figura ??) e será inserido somente o resultado do treinamento neste microcomputador. Essa decisão foi tomada devido a necessidade de grande de processamento para o treinamento da Rede Neural. Este processo de treinamento do algoritmo para reconhecimento das laranjas pode sobrecarregar o processamento da Raspberry PI 4 (Figura ??), já que ela contém o Desktop App e ainda se comunica com a nuvem, o algoritmo de Machine Learning para diferenciar as imagens das frutas, o banco de dados local NoSQL, para armazenar estes dados e, além disso, alguns algoritmos de Eletrônica.

### 4. Sobre os Módulo de Câmera

A decisão sobre os módulos de câmera ficou confusa entre quem iria decidir, pois não estava muito claro de quem seria a responsabilidade, de Eletrônica ou de Software. Então, foi decidido que a escolha dos módulos de câmera (Figura 34) ficaria com eletrônica, já que esta área ficou responsável por administrar a conexão de todos os periféricos eletrônicos com a Raspberry PI 4 (Figura ??). Porém, a comunicação das câmeras seria diretamente com Software, para que a transmissão das imagens não tenha intermediário e o processamento seja lento. Assim, as imagens são enviadas

---

diretamente ao algoritmo de Machine Learning, para que seja possível a classificação das frutas.

## 5. Limitação de Hardware da Rapsberry

Com a decisão do grupo de centralizar os componentes de gerenciamento de eletrônica e software em uma só placa, a Raspberry PI 4 (Figura ??), surgiu um novo desafio de balancear o poder de processamento oferecido pelo microcomputador e as necessidades de processamento das atividades realizadas. Toda a arquitetura de software teve que ser repensada para se adequar ao contexto de racionamento de hardware, desde a escolha do banco de dados e dos algoritmos da rede neural, até a aplicação de saída de dados para o usuário. Um exemplo citado acima é a tecnologia escolhida para o desenvolvimento da aplicação desktop. O grupo teve que desconsiderar frameworks e linguagens que já eram familiares aos membros, mas de grande consumo de recursos computacionais, para tecnologias pouco utilizadas, que trariam o benefício de serem mais leves e otimizadas em relação à utilização de recursos de hardware. Todo esse contexto foi consequência da problemática levantada pela equipe de eletrônica em relação à utilização de mais de uma placa, o que resultou por impactar diretamente as soluções arquiteturais propostas pela equipe de software, trazendo uma maior complexidade na concepção das mesmas.

## 6. Banco de Dados Local

Outra decisão chave que precisamos tomar foi a de qual banco de dados não relacional dentro da Raspberry Pi 4 utilizaríamos, pois teria que ser uma tecnologia que consumisse uma quantidade moderada de recurso de *hardware*, aceitasse concorrência no acesso de leitura e gravação de documentos, além de desempenho que não compromettesse a aplicação. Dentre as possíveis escolhas, selecionamos as tecnologias Redis, TinyDB e MongoDB:

- Redis: é um banco de alto desempenho, superior aos outros dois, entretanto consome uma grande quantidade de memória RAM, que é um recurso escasso dentro da realidade do projeto ([PANDEY, 2017](#)).
- TinyDB: é um banco de dados extremamente leve, mas não possui suporte à concorrência no acesso de leitura e gravação de documentos, sendo inclusive desencorajado pela própria documentação da tecnologia a não utilizá-la quando houver múltiplos acessos a documentos por processos ou *threads* ([TINYDB..., 2016](#)).
- MongoDB: mostrou-se ser a escolha ideal, pois possui suporte à concorrência de acesso de leitura e gravação nos documentos do banco de dados, além de não consumir grande quantidade de recursos de *hardware* ([PANDEY, 2017](#)).

Foi escolhido o MongoDB para o banco de dados local, que é um banco de dados orientado a documentos, distribuído, criado para atender aplicações modernas e com suporte para a nuvem. O MongoDB possui as propriedades ACID ao realizar transações, com suporte à transações distribuídas de vários documentos com isolamento, equilibrando performance com robustez (KEEP; CABRAL, 2018).

## 7. Armazenamento em Nuvem

Foi definida a utilização do Firebase como plataforma de armazenamento de dados em nuvem. Por ser voltada para o desenvolvimento de aplicações web e mobile, a plataforma foi selecionada para o projeto por conta do oferecimento de serviços de gerenciamento, em tempo real, de dados armazenados em um banco NoSQL orientado a documentos (o Firestore), integrado à plataforma, além da autenticação completa de usuários. A autenticação de usuários, incluindo cadastro e login, oferecida pela plataforma é também responsável por fazer, em background, todas as validações referentes aos dados do usuário, isentando o cliente desse processo e configurando mais um motivo para a escolha da utilização do Firebase no projeto.

## 8. Desktop APP

Para o Desktop APP foi discutido se seria utilizado o *framework* JavaScript Electron, a biblioteca *Python* Tkinter ou outra biblioteca *Python* chamada PyQt. O Electron JS de primeira aparentou ser a opção mais atraente, devido a familiaridade com as tecnologias JavaScript, HTML e CSS, que são as utilizadas por este *framework*, entretanto, nos testes realizados o Electron consumiu cerca de 100 MB de memória RAM. Pouco acima do consumo do PyQt, que foi cerca de 50 MB, mas, o PyQt possui a problemática de não ser uma ferramenta que possui uma boa documentação, além de ser apenas uma ponte para a biblioteca de Qt. Por fim, restou a Tkinter, que é a biblioteca padrão para interfaces gráficas que o *Python* utiliza, mas ela é uma biblioteca que possui componentes que possui grande limitação, sendo inviável para fazer os gráficos necessários neste projeto. Portanto, o Electron se mostra uma melhor alternativa, devido menor curva de aprendizado e moderado consumo de recursos de *hardware*.

## 9. Framework da Aplicação Web

Foi escolhido o Framework Vue.js devido a familiaridade de alguns membros do grupo com a tecnologia e a curva de aprendizagem ser baixa, já que o projeto tem um limite de tempo pequeno para o planejamento, levantamento de requisitos e execução do projeto. Além disso, o *framework* é flexível na hora de construir aplicações web e permite criar interfaces reativas utilizando componentes. Ademais, o VueJS dá suporte para desenvolver aplicações Progressive Web App (PWA), que

---

são aplicações web responsivas que se adequam às telas de celulares e funcionam como uma aplicação mobile ao mesmo tempo que uma aplicação web comum.

## 10. Escolha de Biblioteca para aprendizado de máquina

De forma geral, obtivemos resultados aceitáveis para a classificação das laranjas, mesmo com o dataset, até então, limitado, sendo estes: 97% para o algoritmo em Keras. 93% para o algoritmo em PyTorch. Diante deste resultado, por ambos possuírem um desempenho de velocidade de classificação considerado ótimo, resultando em menos de meio de segundo de processamento cada laranja, como podemos ver no resultado da Figura 74, optamos por aquele que obteve mais precisão em sua classificação, entretanto, ao final do projeto, planejamos uma configuração de acurácia ainda maior, conforme a construção de um dataset de laranjas mais robusto. Além disso, o Keras tem uma curva de aprendizado menor e não há preocupação com a configuração da GPU. Além de poder implementar, treinar e testar suas redes rapidamente com depuração fácil.

## 11. Integração de Software com Eletrônica

Devido à limitação que o grupo enfrentou ao realizar o trabalho sem suporte da Raspberry, foi necessário desenvolver métodos que permitissem simular a integração dos trabalho de ambas as engenharias, software e eletrônica, de modo a garantir a funcionalidade do que estava sendo desenvolvido. A equipe de software teve que atuar ativamente dentro do código de eletrônica para desenvolver soluções que permitissem a comunicação entre os módulos projetados, mesmo que a nível de simulação.

- Algoritmo de *machine learning* (software) e algoritmo de rotas (eletrônica): para esta comunicação foi necessário pensar em algo que permitisse os dois códigos conversarem de modo rápido e sem interferência. Foi considerado o uso de requisições HTTP através de uma conexão local dentro placa, mas esse tipo de solução resultaria em uma quantidade enorme de requisições em um curto período de tempo. Para poder contornar este tipo de problema, foi implementado uma conexão *socket* entre os dois algoritmos, visto que, esse tipo de comunicação permitiria um canal de fluxo de dados abertos e sem necessidade de ficar criando requisições de envio e recebimento de informações.
- Algoritmo de contagem (eletrônica) e banco de dados (software): para permitir que os dados de contagem dos frutos pudessem aparecer de forma interativa para usuário, se fez necessário o desenvolvimento de uma solução que pudesse captar os sinais dos periféricos contadores de eletrônica. Tendo isso mente, foi feita uma intervenção no código de eletrônica para que os sinais dos contadores

fossem convertidos em dados para popular o banco não relacional hospedado na placa Raspberry.

# APÊNDICE D – Referências das Implementações (Software e Eletrônica)

Tabela 33 – Referências das Implementações

IMPLEMENTAÇÃO	DESCRIÇÃO	ENDEREÇO
Tela de Quantitativos Web	Tela de informações quantitativas do processamento.	<a href="#">Código no GitHub</a>
Tela de Gráficos Web	Tela de informações gráficas do processamento.	<a href="#">Código no GitHub</a>
Tela de Relatórios	Tela de geração de relatório com as informações de processamento.	<a href="#">Código no GitHub</a>
Comunicação com o Firebase Web	Consumo de dados referentes ao usuário.	<a href="#">Código no GitHub</a>
Tela de Login Web	Login do usuário na aplicação web.	<a href="#">Código no GitHub</a>
Tela de Cadastro Web	Cadastro de usuário na aplicação web.	<a href="#">Código no GitHub</a>
Algoritmos Classificação de Laranjas	Repositório contendo o código fonte dos algoritmos de aprendizado de máquina do projeto.	<a href="#">Código no GitHub</a>
API para envio das imagens das laranjas	Código feito para enviar imagem e obter a classificação; conectar com o banco MongoDB para enviar as informações da classificação; conexão via socket com o Algoritmo de Rotas de eletrônica	<a href="#">Código no GitHub</a>
Função que faz a classificação da qualidade da laranja	Código feito para classificar a qualidade das laranjas com base no modelo de <i>Machine Learning</i> já treinado	<a href="#">Código no GitHub</a>

Continua na próxima página

**Tabela 33 – continuação da página anterior**

IMPLEMENTAÇÃO	DESCRIÇÃO	ENDEREÇO
Função que faz a conexão do algoritmo de <i>Machine Learning</i> com o banco de dados local MongoDB	Função que realiza a comunicação entre o algoritmo de classificação de laranjas que utiliza <i>Machine Learning</i> e o banco de dados MongoDB	<a href="#">Código no GitHub</a>
Integração do Desktop App com banco de dados local MongoDB	Função que realiza a comunicação entre o Desktop APP e o banco de dados MongoDB.	<a href="#">Código no GitHub</a>
Algoritmo de Aquisição de Imagem (Eletrônica)	Realizar o acionamento dos módulos de câmeras a partir da detecção de um fruto na etapa de aquisição de imagem. Controle do sensor infravermelho.	<a href="#">Código no GitHub</a>
Algoritmo de Definição de Rotas (Eletrônica)	Definir o posicionamento dos servo-motores a partir do recebimento dos dados resultantes da classificação de imagem.	<a href="#">Código no GitHub</a>
Código do gráfico de qualidade - Desktop APP	Código que realiza montagem do gráfico de qualidade da laranja na aplicação <i>desktop</i>	<a href="#">Código no GitHub</a>
Código do gráfico de tamanho - Desktop APP	Código que realiza montagem do gráfico de tamanho da laranja na aplicação <i>desktop</i>	<a href="#">Código no GitHub</a>
Código de sincronização de dados - Desktop APP	Código que realiza a sincronização de dados locais com a nuvem na aplicação <i>desktop</i>	<a href="#">Código no GitHub</a>
Código de monitoramento de conexão - Desktop APP	Código que monitora conexão com a nuvem na aplicação <i>desktop</i>	<a href="#">Código no GitHub</a>
Continua na próxima página		

**Tabela 33 – continuação da página anterior**

IMPLEMENTAÇÃO	DESCRIÇÃO	ENDEREÇO
Algoritmo de Contagem das laranjas (Eletrônica)	Realizar a contagem dos frutos selecionados por diâmetro pequeno, médio e grande.	<a href="#">Código no GitHub</a>
Integração software-eletrônica no Algoritmo de Contagem	Equipe de software realiza a comunicação do algoritmo de contagem de laranjas por tamanho feito pela equipe de eletrônica com os subsistemas de software.	<a href="#">Código no GitHub</a>
Código para gerar sessão única pra cada vez que a máquina for ligada	Neste código é feita a recuperação da sessão MongoDB para toda vez que a máquina for ligada ser gerado um lote ( <i>batch</i> ) único.	<a href="#">Código no GitHub</a>

# APÊNDICE E – Autoavaliação dos Integrantes

Tabela 34 – Autoavaliação dos Integrantes da Equipe

INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Camilla Alves de Oliveira	Desenvolvimento e construção da arquitetura eletrônica; Participação em reuniões de brainstorming, que ajudaram a definir a solução proposta pela equipe no projeto; Elaboração de esquemáticos eletrônicos; Elaboração de diagramas de controle; Pesquisa de especificações de componentes; Elaboração do cronograma da equipe de engenharia eletrônica; Pesquisa de referências para concepção da solução proposta; Elaboração dos códigos e rotinas para controle; simulações dos módulos; Refinamento do relatório; Revisão do documento de eletrônica; Participação da construção do manual de operação.
Beatriz Freitas Calheira	Desenvolvimento e construção da arquitetura eletrônica; Participação em reuniões de brainstorming, que ajudaram a definir a solução proposta pela equipe no projeto; Elaboração de esquemáticos eletrônicos; Elaboração de diagramas de controle; Pesquisa de especificações de componentes; Elaboração do cronograma da equipe de engenharia eletrônica; Pesquisa de referências para concepção da solução proposta; Elaboração dos códigos e rotinas para controle; simulações dos módulos; Refinamento do relatório; Revisão do documento de eletrônica; Participação da construção do manual de operação.
Continua na próxima página	

**Tabela 34 – continuação da página anterior**

INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Giovanna Amorim de Farias	Gerenciamento da equipe de eletrônica; Divisão de tarefas entre os membros da equipe; Desenvolvimento e construção da arquitetura eletrônica; Participação em reuniões de brainstorming, que ajudaram a definir a solução proposta pela equipe no projeto; Elaboração de esquemáticos eletrônicos; Elaboração de diagramas de controle; Pesquisa de especificações de componentes; Elaboração do cronograma da equipe de engenharia eletrônica; Elaboração do diagrama de solução geral do projeto; Pesquisa de referências para concepção da solução proposta; Elaboração dos códigos e rotinas para controle; simulações dos módulos; Refinamento do relatório; Revisão do documento de eletrônica; Participação da construção do manual de operação.
Continua na próxima página	

**Tabela 34 – continuação da página anterior**

INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Guilherme Siqueira Brandão	<p>Pesquisa de referências para o entendimento do problema e concepção da solução; participação em reuniões de brainstorming, que ajudaram a definir a solução proposta pela equipe do projeto; elaboração do Termo de Abertura do Projeto (TAP); elaboração da Estrutura Analítica do Projeto (EAP); elaboração de requisitos de produto e de projeto; elaboração dos requisitos da aplicação desktop; elaboração da arquitetura da aplicação desktop; elaboração do banco de dados da aplicação desktop; prototipação em alto nível da aplicação desktop; revisão e correção de partes do relatório de software; descrição de tecnologias de software; escrita do tópico de Arquitetura de Software em geral, principalmente nos tópicos Integração de Todos os Sistemas, Banco de Dados Local, Desktop APP e Inovação; desenvolvimento da aplicação desktop, API de Machine Learning, toda cadeia do banco de dados local, incluindo conexão com algoritmo de contagem feito pela equipe de eletrônica.</p>
Continua na próxima página	

**Tabela 34 – continuação da página anterior**

INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Gustavo Barbosa Carvalho	<p>Coordenação das reuniões do grupo e reuniões das diretorias; elaboração da proposta de projeto; pesquisa de referências e participação de debates para o entendimento do problema e concepção da solução; ; elaboração do Termo de Abertura do Projeto (TAP); auxílio na elaboração da Estrutura Analítica do Projeto (EAP); elaboração dos riscos gerais; elaboração de requisitos de produto e de projeto; elaboração dos requisitos da aplicação desktop; elaboração da arquitetura geral de software; elaboração da arquitetura da aplicação desktop; elaboração do banco de dados da aplicação desktop; revisão e correção de partes do relatório de software; descrição de tecnologias de software; escrita do tópico de Arquitetura de Software em geral, principalmente nos tópicos Integração de Todos os Sistemas, Banco de Dados Local, Desktop APP e Inovação; desenvolvimento da aplicação desktop e banco de dados local.</p>

Continua na próxima página

**Tabela 34 – continuação da página anterior**

INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Natália Maria Rodrigues Queiroz	<p>Pesquisa de referências para o entendimento do problema e concepção da solução; participação em reuniões de brainstorming, que ajudaram a definir a solução proposta pela equipe no projeto; elaboração do questionário utilizado na fase de elicitação de requisitos do projeto; contato direto com produtores para entrevistá-los, afim de coletar respostas para o questionário; elaboração dos requisitos da aplicação web; elaboração de textos e diagramas inseridos na descrição da arquitetura de software; revisão e correção de partes do relatório, incluindo a formatação e padronização de tabelas; Estudo sobre a utilização do VueJS e do Firebase dentro do projeto; Participação na configuração da Aplicação Web; Implementação das funcionalidades da Aplicação Web (código); Configuração do Firestore; Estruturação dos dados no Firestore; Elaboração do protótipo da aplicação web; Elaboração dos textos referentes à aplicação web na seção de Arquitetura da Informação; Participação na elaboração dos textos e diagramas referentes à arquitetura da aplicação web; Elaboração do tópico Inovação; Deploy da aplicação web;</p>
Continua na próxima página	

**Tabela 34 – continuação da página anterior**

INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Maria Luiza Ferreira Assumpção Almeida	Pesquisa de referências para o entendimento do problema e concepção da solução; participação em reuniões de brainstorming, que ajudaram a definir a solução proposta pela equipe no projeto; elaboração do TAP; elaboração do questionário utilizado na fase de elicitação de requisitos do projeto; elaboração dos requisitos da aplicação web; elaboração de textos e diagramas inseridos na descrição da arquitetura de software; revisão e correção do relatório como um todo; Realização dos códigos da Aplicação Web; Comunicação com Firestore para obtenção dos dados; Organização das atividades da equipe de Software, com um roadmap do que deveria ser entregue até o PC3; Realização da integração entre software e eletrônica para o envio da classificação dos frutos; Refinamento do relatório; Revisão de todo o documento de software; Realização do cronograma de software até o PC3; Participação da construção do manual de operação
João Pedro Gomes Cabral Ferreira	Implementação da rede neural, algoritmo de pre-processamento de dados, estudo dos dados, responsável pela coleta dos dados. Desenvolvimento de protótipos (notebooks) utilizando keras e pytorch com fim de comparar o desempenho dos dois. Desenvolvimento da api responsável por classificar imagens e da arquitetura da rede neural.
Continua na próxima página	

**Tabela 34 – continuação da página anterior**

INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Thiago Ferreira	<p>Implementação e criação dos canais de ferramentas organizacionais Github e Teams, além da integração da equipe nestas. Desenvolvimento do cronograma de atividades da equipe, elicitação de tarefas a serem realizadas, auxílio de coordenação nas reuniões, participação nos brainstorms de elaboração de escopo e tomada de decisão da equipe; iniciativa em mudanças com relação a pontos de falha no projeto; auxílio na coordenação das reuniões da equipe; trabalho conjunto com a equipe de engenharia de software para a elicitação de escopo, arquitetura e necessidades macro do projeto; desenvolvimento da documentação referente a recursos humanos; Todos os tópicos e artefatos referentes a rede neural; Levantamento de custos para a parte de software; descrição das tecnologias a serem utilizadas no projeto, na parte de software. Incrementação e coleta de dados para o dataset, participação desenvolvimento do código da rede neural, escrita relatório machine learning.</p>
Matheus Henrique Almeida de Souza	<p>Dimensionamento de componentes de alimentação e motores; Levantamento de custos de componentes da estrutura e elaboração de tabela; Elaboração de Arquitetura de Estrutura / Energia com colaboração dos membros; Levantamento de soluções de energia, diagrama unifilar, dimensionamento de componentes de regulação de tensão, servos motores e alimentação secundária de componentes de eletrônica.</p>
Continua na próxima página	

**Tabela 34 – continuação da página anterior**

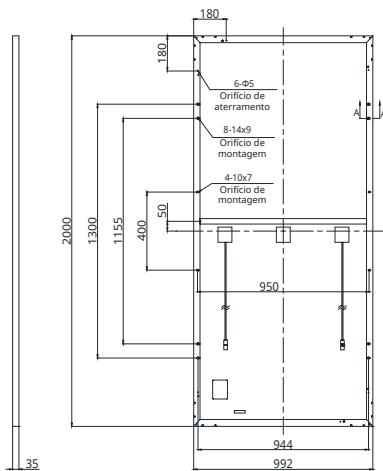
INTEGRANTE	CONTRIBUIÇÃO INDIVIDUAL
Letícia Marinho de Souza	Dimensionamento do sistema autônomo fotovoltaico; Dimensionamento dos componentes de alimentação; Escrita do relatório; Elaboração da solução de Energia. Revisão do relatório; Elaboração do Diagrama unifilar, dimensionamento de componentes de regulação de tensão. Realização de simulações do sistema fotovoltaico, bem como a escolha dos componentes.
Lucas Afonso	Contribuição acerca do funcionamento e arquitetura da máquina em todos seus processos e subsistemas. Definição da arquitetura da câmara de processamento de imagem e integração com os sistemas eletrônicos que deverão ser instalados no interior da câmara; Levantamento de informações relacionadas à disposição da iluminação e processamento de imagem, bem como o uso de filmes e polímeros anti-reflexivos como forma de maximizar a qualidade da imagem captada. Levantamento de requisitos, riscos e viabilidade técnica do sistema de processamento de imagem.

# APÊNDICE F – Especificações Técnicas -

## Módulo Canadian

## DESENHO DE ENGENHARIA (mm)

Vista traseira



Seção transversal do quadro A-A



## DADOS ELÉTRICOS | STC\*

CS3U	355P	360P	365P	370P	375P
Máx. Potência Nominal (Pmax)	355 W	360 W	365W	370W	375W
Opt. Tensão de Operação (Vmp)	39,4 V	39,6 V	39,8 V	40,0 V	40,2 V
Opt. Corrente de Operação (Imp)	9,02 A	9,10 A	9,18 A	9,26 A	9,34 A
Tensão de Circuito Aberto (Voc)	46,8 V	47,0 V	47,2 V	47,4 V	47,6 V
Corrente de Curto-Círcuito (Isc)	9,59 A	9,67 A	9,75 A	9,83 A	9,91 A
Eficiência de Módulo	17,9%	18,2%	18,4%	18,7%	18,9%
Temperatura de Operação	-40°C ~ +85°C				
Máx. Tensão de Sistema	1500V (IEC/UL) ou 1000V (IEC/UL)				
Desempenho antichamas do módulo	TIPO 1 (UL 1703) ou Classe C (IEC 61730)				
Máx. valor nominal do fusível de série	30 A				
Classe da aplicação	Classe A				
Tolerância de potência	0 ~ + 10 W				

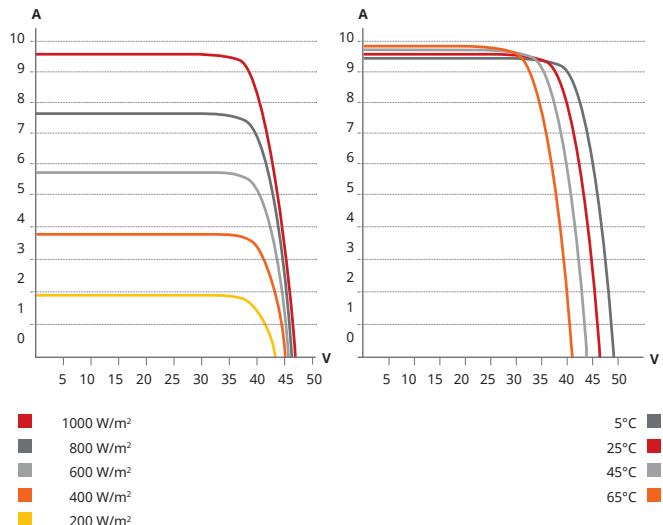
\* Sob condições de teste padrão (STC) de irradiância de 1000 W/m<sup>2</sup>, espectro AM 1.5 e temperatura de célula de 25°C.

## DADOS ELÉTRICOS | NMOT\*

CS3U	355P	360P	365P	370P	375P
Máx. Potência Nominal (Pmax)	264 W	268 W	272 W	276 W	279 W
Opt. Tensão de Operação (Vmp)	36,7 V	36,9 V	37,0 V	37,2 V	37,4 V
Opt. Corrente de Operação (Imp)	7,21 A	7,27 A	7,34 A	7,40 A	7,46 A
Tensão de Circuito Aberto (Voc)	44,0 V	44,1 V	44,3 V	44,5 V	44,7 V
Corrente de Curto-Círcuito (Isc)	7,74 A	7,80 A	7,87 A	7,93 A	7,99 A

\* Sob temperatura nominal de operação do módulo (NMOT), irradiância de 800 W/m<sup>2</sup>, espectro AM 1.5, temperatura ambiente de 20°C, velocidade do vento de 1 m/s.

## CS3U-365P / CURVAS I-V



## DADOS MECÂNICOS

Especificação	Dados
Tipo de célula	Policristalina
Arranjo de células	144 [2 X (12 X 6) ]
Dimensões	2000 X 992 X 35 mm (78,7 X 39,1 X 1,38 pol.)
Peso	22,5 kg (49,6 lbs)
Capa dianteira	Vidro temperado de 3,2 mm
Quadro	Liga de Alumínio Anodizado, com barra transversal
Caixa J	IP68, 3 diódos de bypass
Cabo	4 mm <sup>2</sup> (IEC), 12 AWG (UL)
Comprimento do cabo (Incluindo conector)	Retrato: 400 mm (15,7 pol.) (+) / 280 mm (11,0 pol.) (-); paisagem: 1250 mm (49,2 pol.); conexão leapfrog: 1670 mm (65,7 pol.)*
Conector	Série T4 ou H4 UTX ou MC4-EVO2
Por palete	30 peças
Por contêiner (40' HQ)	660 peças

\* Para informações detalhadas, contate seu Representante de Vendas ou Representante Técnico local.

## CARACTERÍSTICAS DE TEMPERATURA

Especificação	Dados
Coeficiente de Temperatura (Pmax)	-0,36 % / °C
Coeficiente de Temperatura (Voc)	-0,28 % / °C
Coeficiente de Temperatura (Isc)	0,05 % / °C
Temperatura Nominal de Operação do Módulo	42 ± 3°C

## SEÇÃO DE PARCEIROS

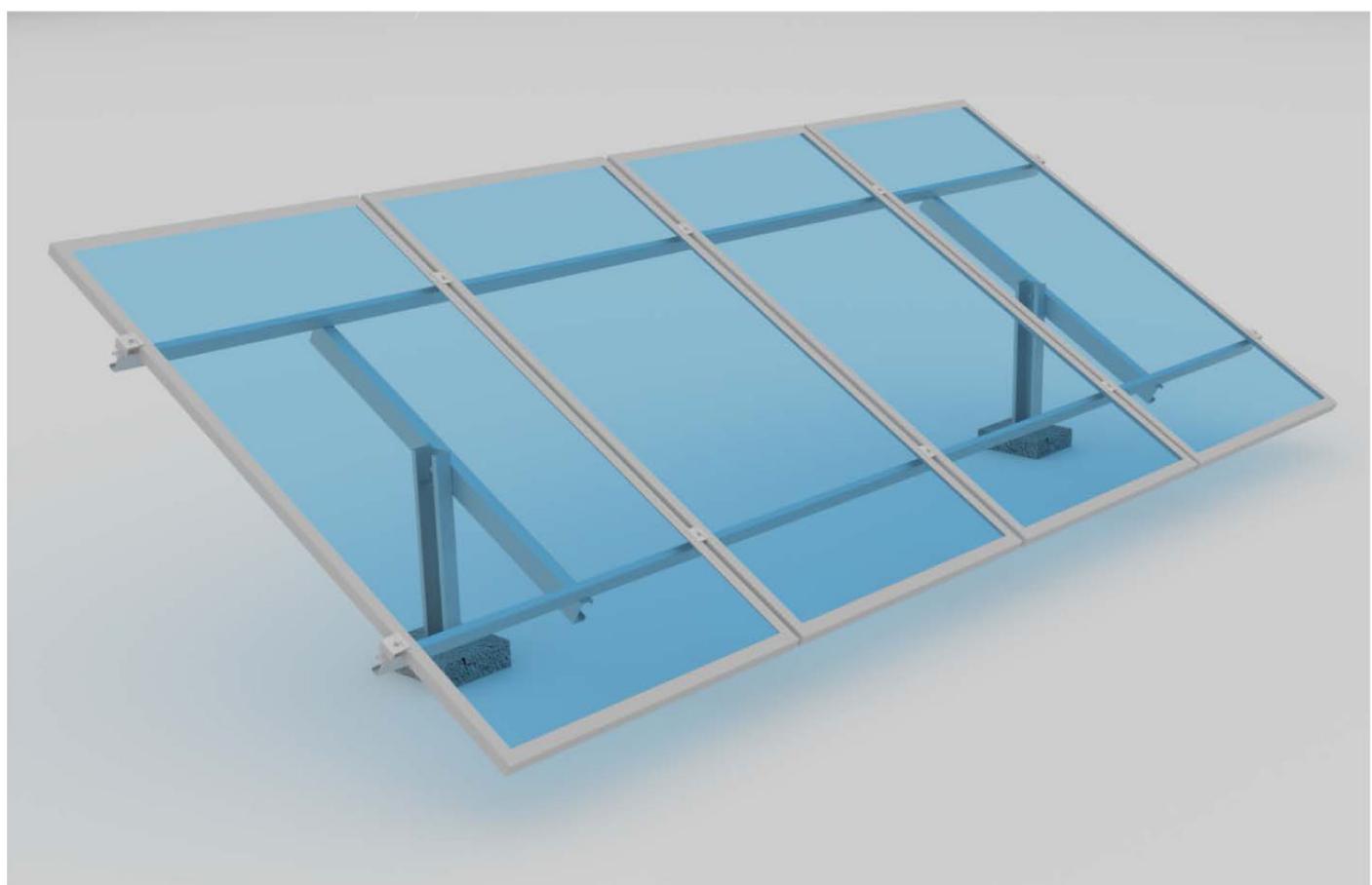


\* As especificações e características principais contidas nesta ficha de dados poderão diferir ligeiramente das dos produtos reais, devido à inovação e melhorias efetuadas continuamente nestes produtos. A Canadian Solar Inc. se reserva o direito de fazer quaisquer ajustes necessários nas informações descritas aqui, a qualquer tempo, sem aviso. Alertamos que os módulos PV devem ser manuseados e instalados por pessoas profissionalmente qualificadas e que as instruções de segurança e de instalação devem ser lidas, cuidadosamente, antes do uso de nossos módulos PV.

# APÊNDICE G – Manual de montagem do suporte dos módulos

# ESTRUTURA SOLO

INSTRUÇÃO DE MONTAGEM



## FERRAMENTAS / EQUIPAMENTOS

- Chave de boca para parafuso sextavado M8
- Chave de boca para parafuso sextavado M12
- Parafusadeira/furadeira com bocal para parafuso auto atarrachante 5/16".
- Transferidor de graus ou equipamento que faça função similar.

## TORQUE DE APERTO

- Parafuso M12 - 5daN.m
- Parafuso M8 – 1,5daN.m

## CUIDADOS

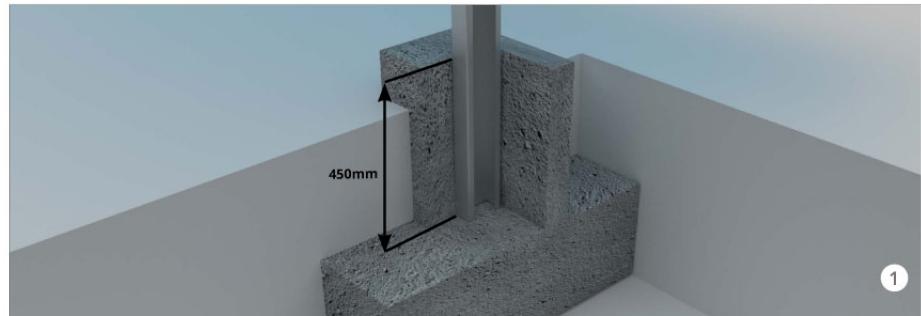
- Verificar o correto esquadrejamento da estrutura;
- Ao final da instalação realizar inspeção visual, a fim de verificar a correta fixação de todos os elementos.

# ESTRUTURA SOLO

*Instrução de Montagem*

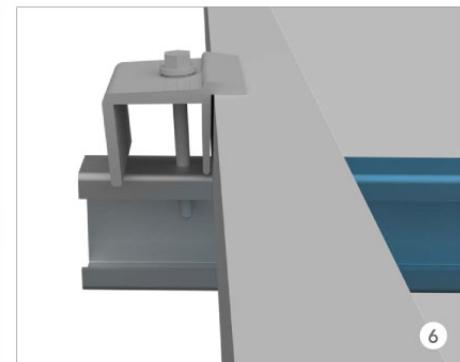
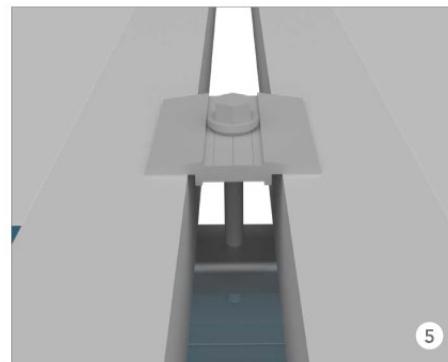
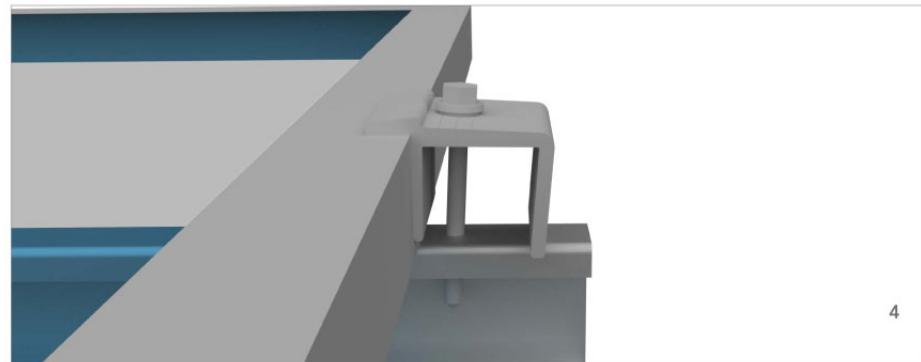
## CONECTANDO A ESTRUTURA

- Verifique no manual do módulo quais as distâncias de fixação;
- Concretar as pilastras (1) com 450mm de profundidade obedecendo as distâncias entre pés; [figura 1]
- Conectar as travessas utilizando o parafuso M12; [figura 2]
- Instalar as longarinas utilizando parafusos M8. Regular o ângulo da estrutura utilizando o transferidor ou equipamento que faça função similar; [figura 3]
- Travar a estrutura apertando o parafuso M8 do oblongo de regulagem;



## INSTALAÇÃO DAS PLACAS

- Coloque o módulo sobre as longarinas de aço;
- Faça primeiro a fixação da lateral do módulo através do grampo final. O furo em que o parafuso é fixado, é feito através da ponta de brocagem do mesmo; [figura 4]
- Coloque o segundo módulo e faça a fixação utilizando o grampo do meio. O furo em que o parafuso é fixado, é feito através da ponta de brocagem do mesmo; [figura 5]
- Após a instalação de todos os módulos no trilho, coloque o grampo final. [figura 6]



*Obs 1. Os grampos de fixação final e intermediário, atendem a módulos com altura de 35 a 50mm.*

*Obs 2. Proteção garantida por zincagem a fogo, permitindo áreas descobertas de zinco de até 1cm<sup>2</sup>, sem perda de proteção (isto inclui furos necessários para fixação dos módulos, arranhões e descascados acidentais).*

## APÊNDICE H – Simulação *PVSyst*

## Stand alone system: Simulation parameters

<b>Project :</b>	<b>PI2</b>		
<b>Geographical Site</b>	<b>Avaré</b>	<b>Country</b>	<b>Brazil</b>
<b>Situation</b>	Latitude -23.11° S	Longitude -48.95° W	
Time defined as	Legal Time zone UT-3	Altitude 782 m	
<b>Meteo data:</b>	<b>Avaré</b> Meteonorm 7.3, Sat=100% - Synthetic		
<b>Simulation variant :</b>	<b>PI2</b>		
	Simulation date	21/11/20 18h59	
<b>Simulation parameters</b>	System type	<b>Stand alone system with batteries</b>	
<b>Collector Plane Orientation</b>	Tilt 23°	Azimuth 0°	
<b>Models used</b>	Transposition Perez	Diffuse Circumsolar	Perez, Meteonorm separate
<b>User's needs :</b>	Daily household consumers average	Constant over the year 3.9 kWh/Day	
<b>PV Array Characteristics</b>			
<b>PV module</b>	Si-poly	Model <b>CS3U-355PB-AG 1500V HE</b>	
Original PVsyst database		Manufacturer Generic	
Number of PV modules	In series	1 modules	In parallel
Total number of PV modules	nb. modules	4	4 strings
Array global power	Nominal (STC)	<b>1420 Wp</b>	Unit Nom. Power
Array operating characteristics (50°C)	U mpp	35 V	At operating cond.
Total area	Module area	<b>8.0 m²</b>	I mpp 37 A Cell area 7.1 m²
<b>System Parameter</b>	System type	<b>Stand alone system</b>	
<b>Battery</b>	Model <b>Open 12V / 100 Ah</b>		
Battery Pack Characteristics	Manufacturer Generic		
	Nb. of units 4 in parallel		
	Voltage 12 V	Nominal Capacity 400 Ah	
	Discharging min. SOC 20.0%	Stored energy 3.8 kWh	
	Temperature Average between fixed (25°C) and External		
<b>Controller</b>	Model Universal controller with MPPT converter		
Converter	Technology MPPT converter	Temp coeff. -5.0 mV/°C/Elem.	
Battery Management control	Maxi and EURO efficiencies 97.0 / 95.0%		
	Threshold commands as	SOC calculation	
	Charging SOC = 0.92 / 0.75	approx. 13.8 / 12.4 V	
	Discharging SOC = 0.20 / 0.45	approx. 11.6 / 12.1 V	
<b>PV Array loss factors</b>			
Thermal Loss factor	Uc (const) 20.0 W/m²K	Uv (wind) 0.0 W/m²K / m/s	
Wiring Ohmic Loss	Global array res. 16 m	Loss Fraction 1.5 % at STC	
Serie Diode Loss	Voltage drop 0.7 V	Loss Fraction 1.8 % at STC	
Module Quality Loss		Loss Fraction -0.3 %	
Module mismatch losses		Loss Fraction 2.0 % at MPP	
Strings Mismatch loss		Loss Fraction 0.10 %	

## Stand alone system: Simulation parameters

Incidence effect (IAM): User defined profile

10°	20°	30°	40°	50°	60°	70°	80°	90°
1.000	1.000	1.000	0.990	0.990	0.970	0.920	0.760	0.000

## Stand alone system: Detailed User's needs

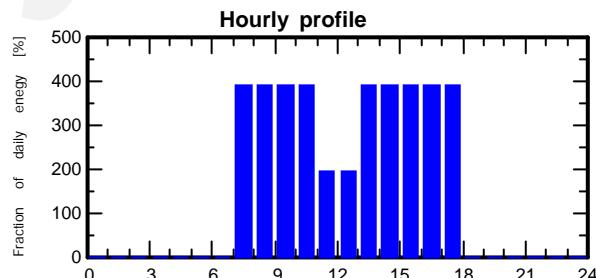
**Project :** PI2  
**Simulation variant :** PI2

<b>Main system parameters</b>	System type	<b>Stand alone system with batteries</b>		
		tilt	azimuth	
PV Field Orientation		23°	0°	
PV modules	Model	CS3U-355PB-AG 1500V HE	Pnom	355 Wp
PV Array	Nb. of modules	4	Pnom total	<b>1420 Wp</b>
Battery	Model	Open 12V / 100 Ah	Technology	Lead-acid, vented, plates
Battery pack	Nb. of units	4	Voltage / Capacity	<b>12 V / 400 Ah</b>
User's needs	Daily household consumers	Constant over the year	Global	1437 kWh/year

**Daily household consumers, Constant over the year, average = 3.9 kWh/day**

Annual values

	Number	Power	Use	Energy
Raspberry	1	15W/lamp	10H/day	150Wh/day
Monitor LCD	1	36W/app	10H/day	360Wh/day
Ventoinha	1	78W/app	10H/day	780Wh/day
Bomba	1		10Wh/day	120Wh/day
Servo Motor	2	6W tot	10H/day	126Wh/day
Motor Redução	2	120W tot	10H/day	2400Wh/day
Total daily energy				3936Wh/day



## Stand alone system: Main results

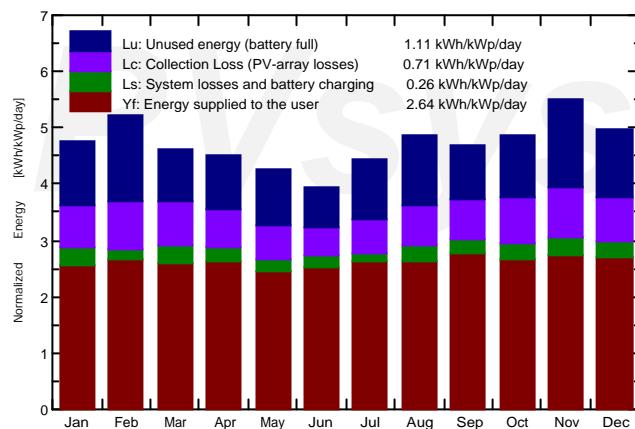
**Project :** PI2  
**Simulation variant :** PI2

<b>Main system parameters</b>		<b>System type</b>	<b>Stand alone system with batteries</b>		
PV Field Orientation		tilt	23°	azimuth	0°
PV modules		Model	CS3U-355PB-AG 1500V HE	Pnom	355 Wp
PV Array		Nb. of modules	4	Pnom total	<b>1420 Wp</b>
Battery		Model	Open 12V / 100 Ah	Technology	Lead-acid, vented, plates
Battery pack		Nb. of units	4	Voltage / Capacity	<b>12 V / 400 Ah</b>
User's needs	Daily household consumers		Constant over the year	Global	1437 kWh/year

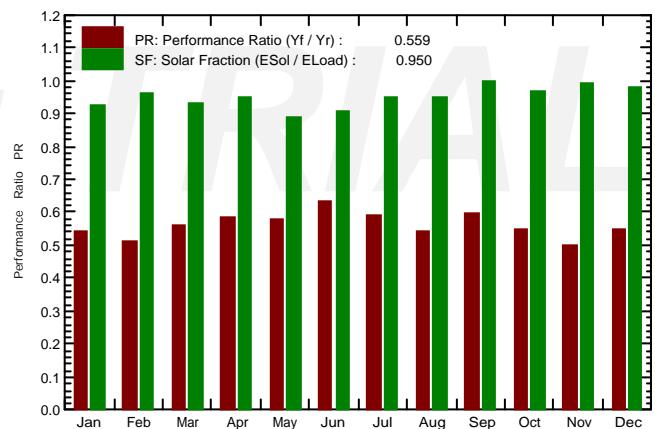
### **Main simulation results**

System Production	<b>Available Energy</b>	<b>2002 kWh/year</b>	Specific prod.	1410 kWh/kWp/year
	Used Energy	1366 kWh/year	Excess (unused)	575 kWh/year
	Performance Ratio PR	55.90 %	Solar Fraction SF	94.98 %
Loss of Load	Time Fraction	5.3 %	Missing Energy	72 kWh/year
Battery aging (State of Wear)	Cycles SOW	93.7%	Static SOW	90.9%
	Battery lifetime	11.0 years		

Normalized productions (per installed kWp): Nominal power 1420 Wp



Performance Ratio PR and Solar Fraction SF



## PI2 Balances and main results

	GlobHor kWh/m <sup>2</sup>	GlobEff kWh/m <sup>2</sup>	E_Avail kWh	E_UNUSED kWh	E_Miss kWh	E_User kWh	E_Load kWh	SolFrac ratio
January	160.4	143.9	169.3	49.23	8.91	113.2	122.1	0.927
February	149.9	143.1	168.0	59.70	4.24	106.1	110.3	0.962
March	137.3	140.9	164.3	41.43	8.26	113.9	122.1	0.932
April	119.3	133.1	156.8	40.24	5.81	112.4	118.2	0.951
May	107.0	130.6	157.2	45.03	13.44	108.7	122.1	0.890
June	93.0	116.9	141.7	30.82	11.04	107.2	118.2	0.907
July	107.2	136.0	163.7	46.59	6.03	116.1	122.1	0.951
August	126.5	148.3	177.0	53.99	6.45	115.7	122.1	0.947
September	130.1	137.7	163.6	40.23	0.00	118.2	118.2	1.000
October	151.0	147.8	173.5	49.42	4.32	117.8	122.1	0.965
November	176.5	160.6	189.6	65.89	1.27	116.9	118.2	0.989
December	169.8	150.3	177.1	52.32	2.37	119.8	122.1	0.981
Year	1628.0	1689.3	2001.8	574.90	72.14	1366.0	1438.1	0.950

Legends: GlobHor Global horizontal irradiation  
GlobEff Effective Global, corr. for IAM and shadings  
E\_Avail Available Solar Energy  
E\_UNUSED Unused energy (battery full)  
E\_Miss Missing energy  
E\_User Energy supplied to the user  
E\_Load Energy need of the user (Load)  
SolFrac Solar fraction (EUsed / ELoad)

## Stand alone system: Special graphs

Project : PI2

Simulation variant : PI2

### Main system parameters

PV Field Orientation

PV modules

PV Array

Battery

Battery pack

User's needs

### System type

tilt

23°

azimuth

0°

Model

CS3U-355PB-AG 1500V HE

Pnom

355 Wp

Nb. of modules

4

Pnom total

1420 Wp

Model

Open 12V / 100 Ah

Technology

Lead-acid, vented, plates

Nb. of units

4

Voltage / Capacity

12 V / 400 Ah

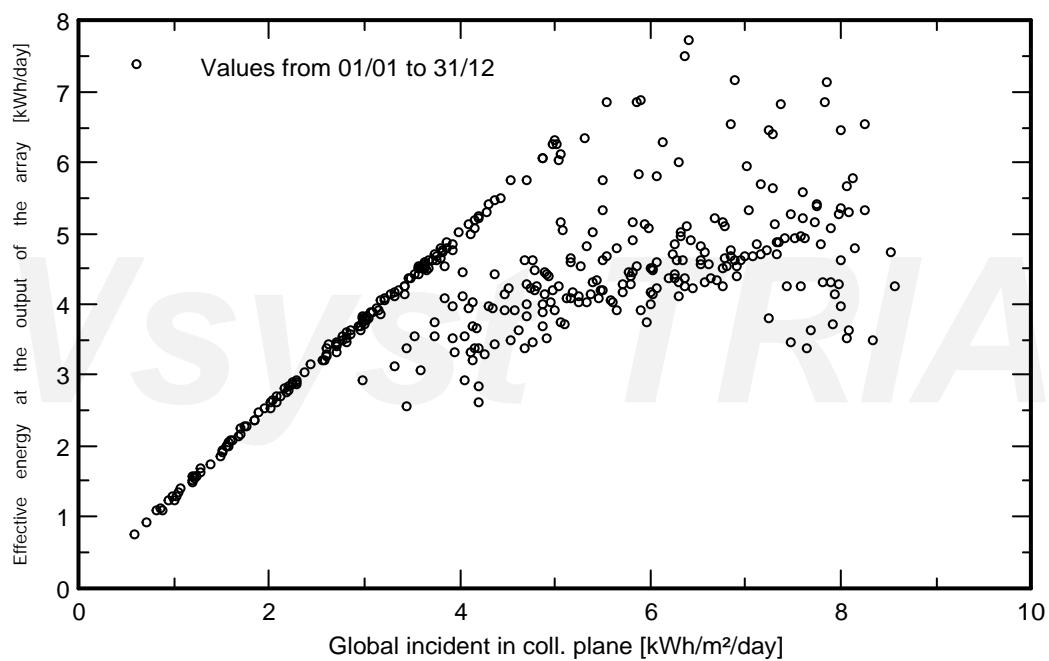
Daily household consumers

Constant over the year

Global

1437 kWh/year

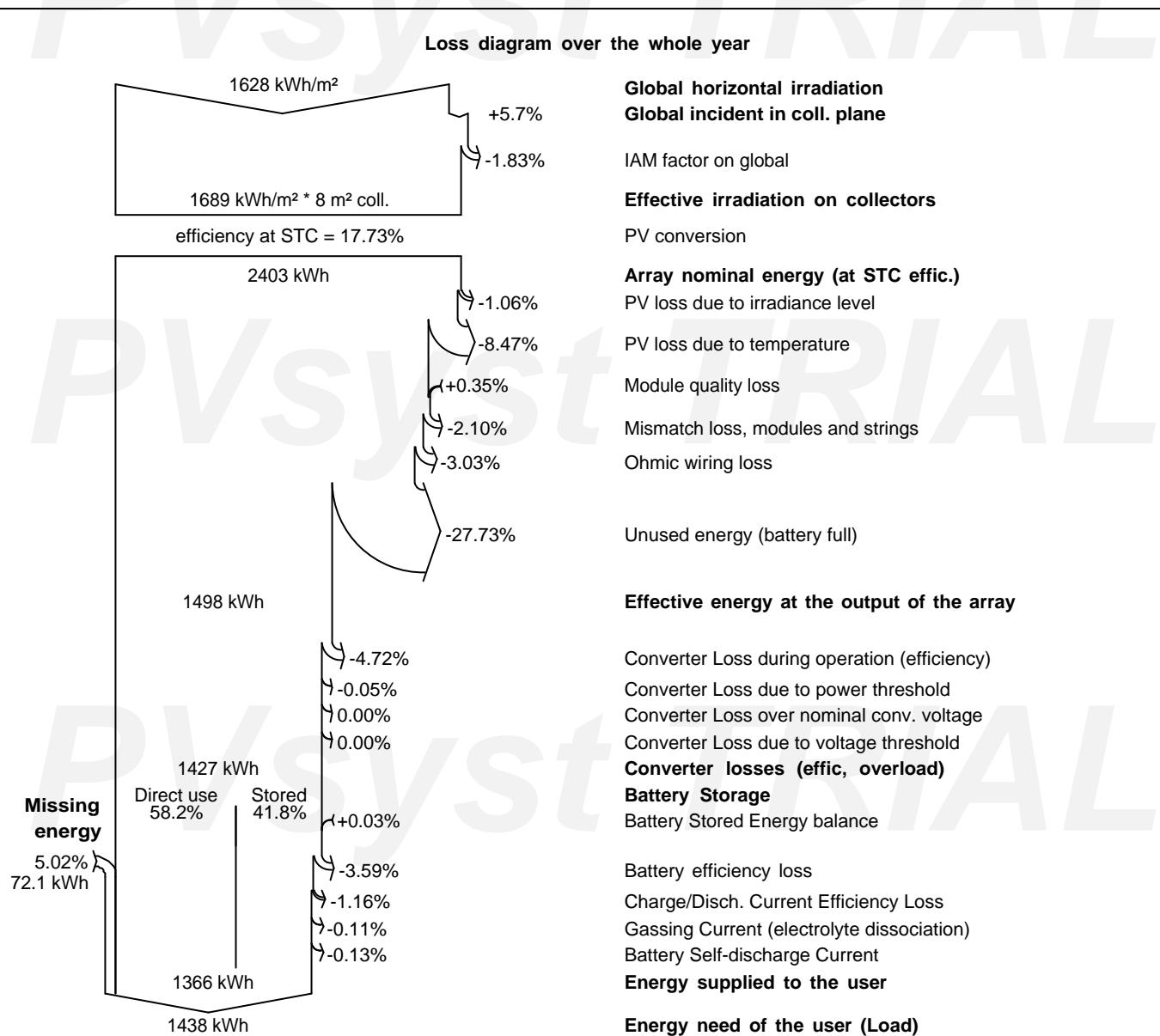
## Daily Input/Output diagram



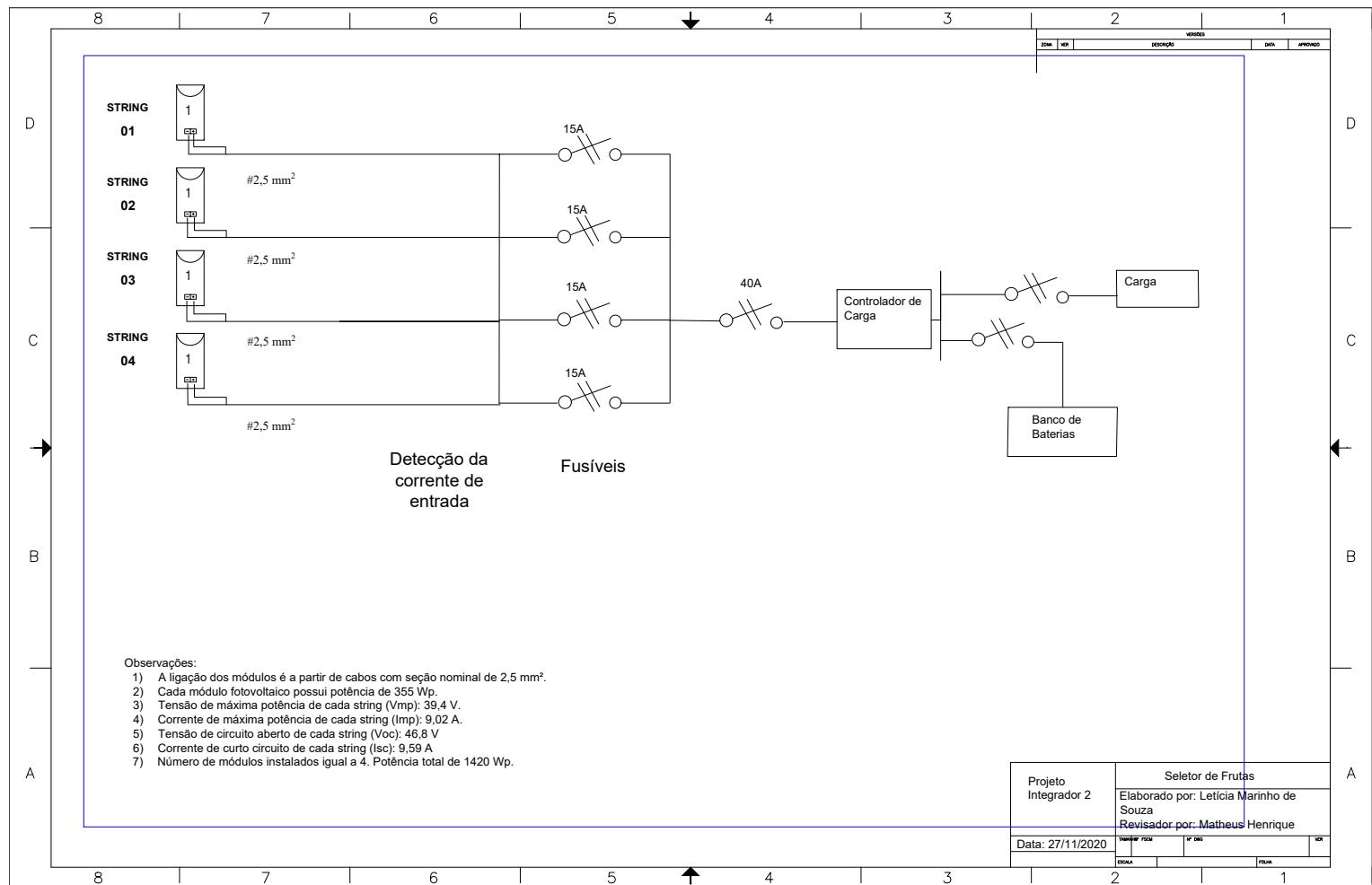
## Stand alone system: Loss diagram

**Project :** PI2  
**Simulation variant :** PI2

<b>Main system parameters</b>		<b>Stand alone system with batteries</b>			
PV Field Orientation		System type	tilt 23°	azimuth 0°	
PV modules		Model	CS3U-355PB-AG 1500V HE	Pnom 355 Wp	
PV Array	Nb. of modules	4		Pnom total 1420 Wp	
Battery	Model	Open 12V / 100 Ah	Technology Lead-acid, vented, plates		
Battery pack	Nb. of units	4	Voltage / Capacity 12 V / 400 Ah		
User's needs	Daily household consumers	Constant over the year	Global	1437 kWh/year	



# APÊNDICE I – Diagrama Unifilar



# APÊNDICE J – Desenhos Técnicos - Estrutura

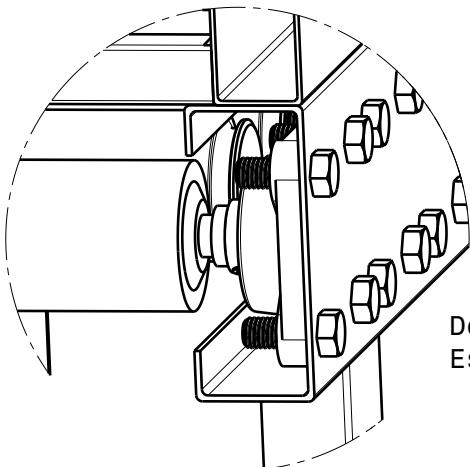
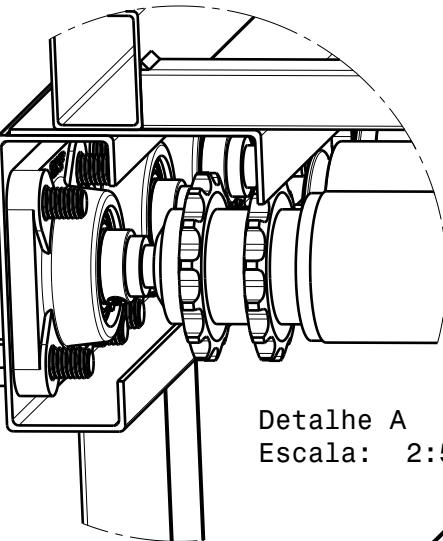
D

C

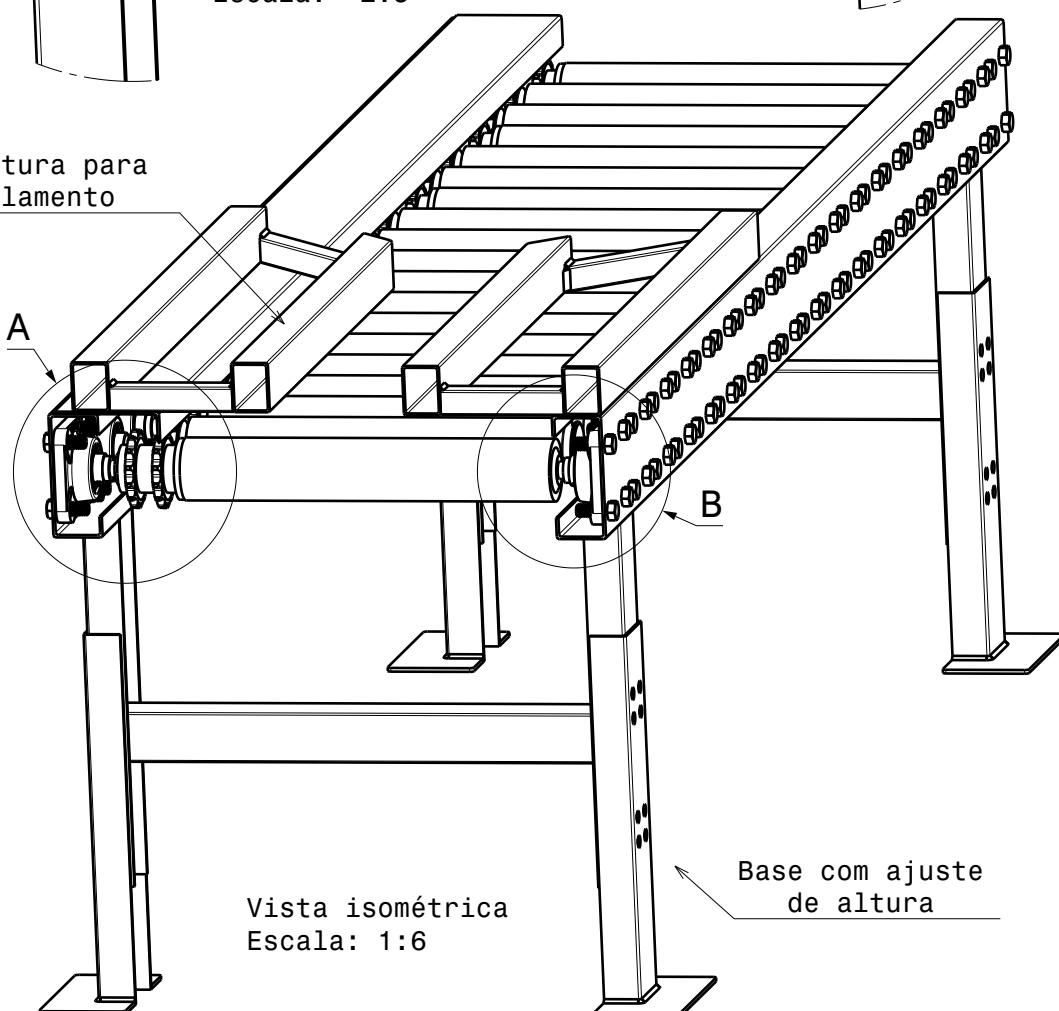
B

A

4



Estrutura para afunilamento



MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
22/11/2020

TÍTULO DO TRABALHO:

# ESTEIRA SELETORA

FOLHA

A4



Projeto Integrador 2 - UnB

UNIDADE  
- MASSA (kg)  
- 77,389

NOME DO COMPONENTE:

ESTEIRA SELETORA - MÓDULO

CÓDIGO

-

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

This drawing is our property; it can't be reproduced or communicated without our written agreement.

D

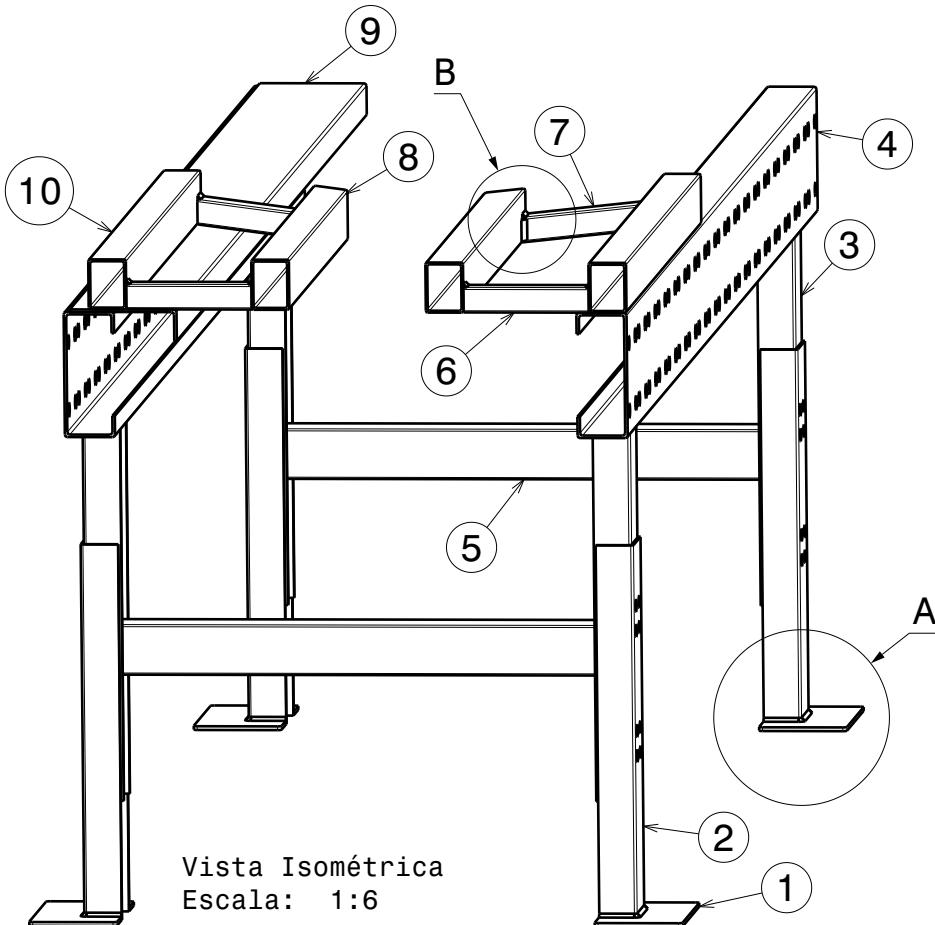
A

4

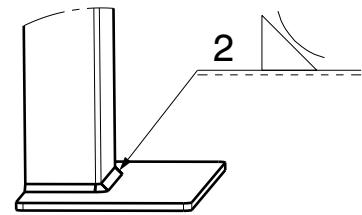
3

2

1



Detalhe B  
Cordão de Solda  
Escala: 1:3



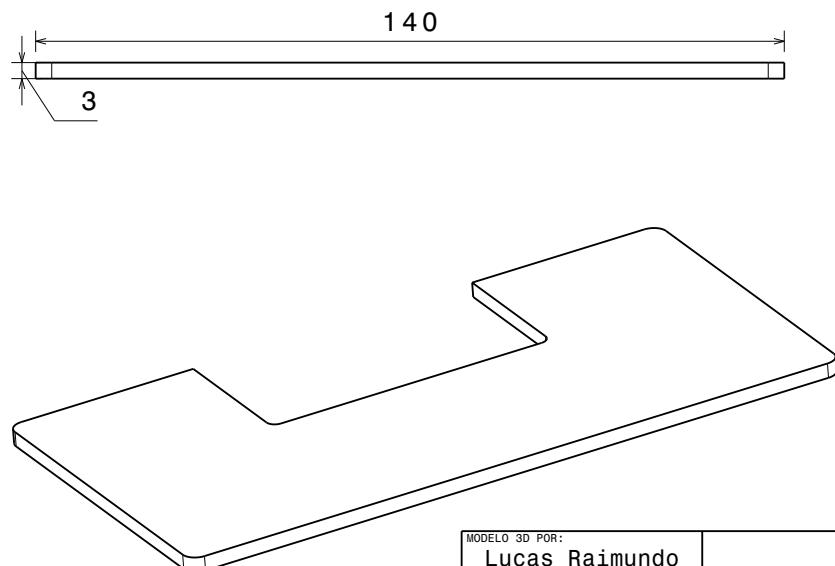
Detalhe A  
Cordão de Solda  
Escala: 1:3

Vista Isométrica  
Escala: 1:6

Nº	CÓDIGO	Nome do componente	Quantidade
1	EST01	Chapa de apoio da base	4
2	EST02	Base	4
3	EST03	Ajuste de altura da base	4
4	EST04	Perfil longitudinal	2
5	EST05	Barra de reforço	2
6	EST06	Barra de reforço afunilamento	2
7	EST07	Barra diagonal afunilamento	2
8	EST08	Barra longitudinal	2
9	EST09	Anteparo	1
10	EST10	Barra longitudinal 2	2

MODELO 3D POR: <b>Lucas Raimundo</b>		TÍTULO DO TRABALHO:	I	-
DATA: <b>20/11/2020</b>			H	-
REVISADO POR: <b>Lucas Raimundo</b>		MATERIAL:	G	-
DATA: <b>21/11/2020</b>			F	-
FOLHA		<b>Projeto Integrador 2 - UnB</b>		
UNIDADE	MASSA (kg)	NOME DO COMPONENTE:	CÓDIGO	
-	<b>15,535</b>	<b>Estrutura da esteira</b>	<b>EST</b>	A
This drawing is our property; it can't be reproduced or communicated without our written agreement.				

Vista frontal  
Escala: 1:1



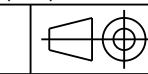
Vista isométrica  
Escala: 1:1

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE

A4



## SELETOR DE LARANJA

MATERIAL:

AÇO SAE 1008

Projeto Integrador 2 - UnB

UNIDADE

mm

WEIGHT (kg)

0,1865

NOME DO COMPONENTE:

CHAPA DE APOIO DA BASE

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

This drawing is our property; it can't be reproduced or communicated without our written agreement.

D

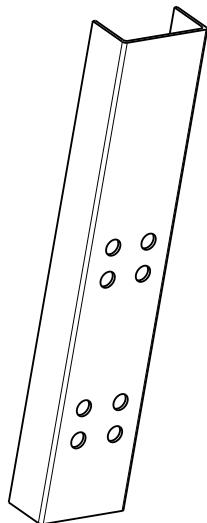
A

D

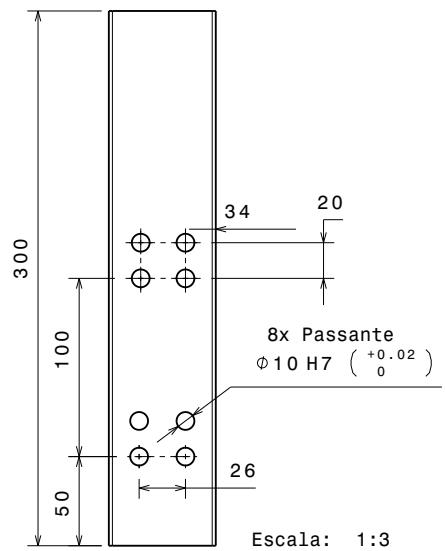
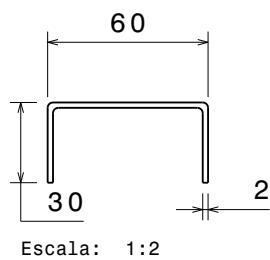
C

B

A



Vista isométrica  
Escala: 1:3



MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE  
A4

UNIDADE  
mm WEIGHT (kg)  
0,532

## SELETOR DE LARANJA

MATERIAL:

AÇO SAE 1008

Projeto Integrador 2 - UnB

This drawing is our property; it can't be reproduced or communicated without our written agreement.

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

D

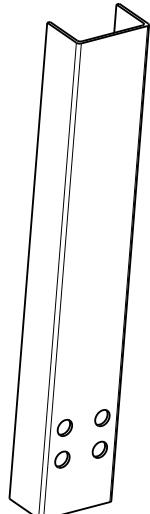
A

D

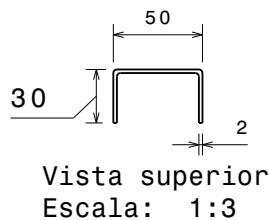
C

B

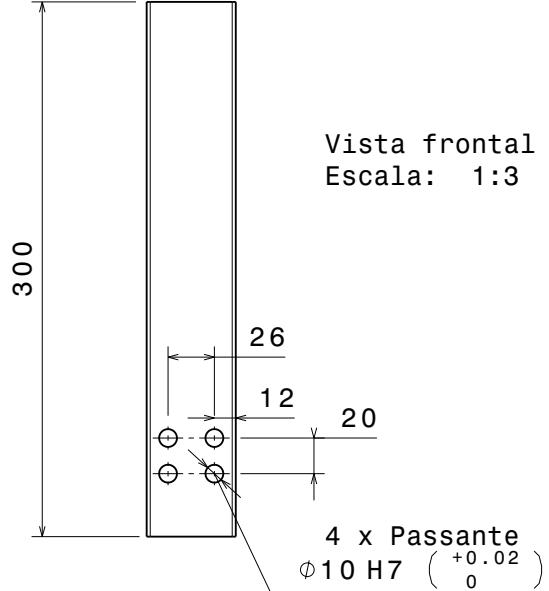
A



Vista Isométrica  
Escala: 1:3



Vista superior  
Escala: 1:3



Vista frontal  
Escala: 1:3

4 x Passante  
 $\phi 10 \text{ H7 } (+0.02 \text{ } 0)$

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE  
A4



## SELETOR DE LARANJA

MATERIAL:

AÇO SAE 1008

Projeto Integrador 2 - UnB

UNIDADE  
mm

WEIGHT (kg)

NOME DO COMPONENTE:  
AJUSTE DE ALTURA DA BASE

CÓDIGO  
EST03

This drawing is our property; it can't be reproduced or communicated without our written agreement.

D

A

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

4

6

三

A

4

15 80

76 x Passante  
Ø10 H6

5

Vista superior  
Escala: 1:6

1535

4

8

**MODELO 3D POR:**  
Lucas Raimundo  
**DATA:**  
20/11/2020  
**REVISADO POR:**  
Lucas Raimundo  
**DATA:**  
MATERIAIS

Vistal frontal  
Escala: 1:3

100

A technical drawing showing two parallel vertical lines with a horizontal dimension line above them labeled '17'. A leader line points from the right side of the drawing to a separate dimension line labeled '2'.

1

Vista isométrica  
Escala: 1:10

MODELO 3D POR:	
DATA:	Lucas Raimundo
	20/11/2020
REVISADO POR:	
DATA:	Lucas Raimundo
SIZE	
A4	
UNIDADE	WEIGHT (kg)
mm	1,385

This drawing is our property; it

# SELETOR DE LARANJA

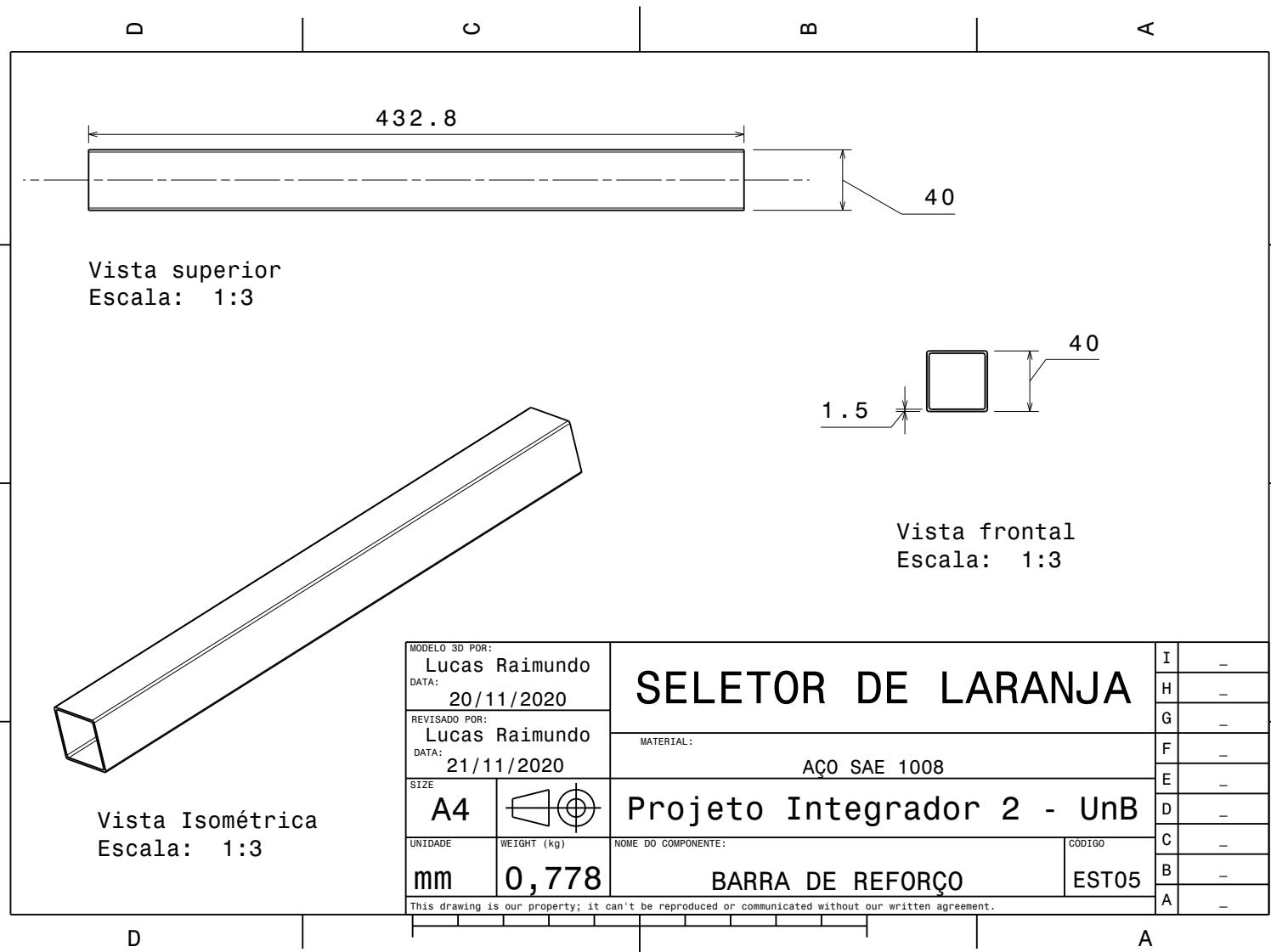
10 of 10

ACO SAE 1008

Projeto Integrador 2 - UnB

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

This drawing is our property; it can't be reproduced or communicated without our written agreement.



D C B A

4

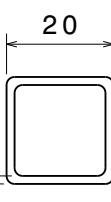
C

B

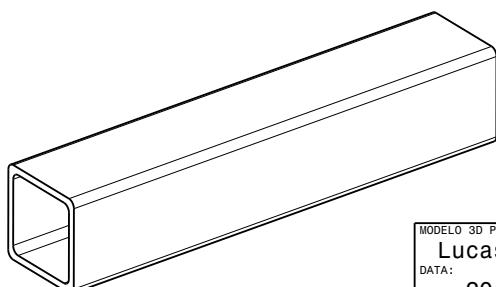
A

3

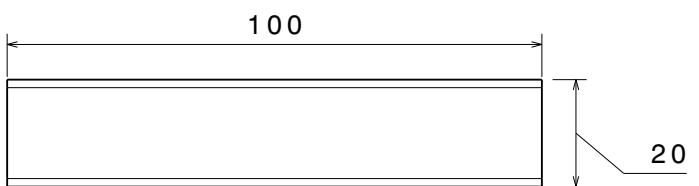
Vista frontal  
Escala: 1:1



2



Vista Isométrica  
Escala: 1:1



Vista Lateral  
Escala: 1:1

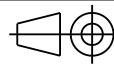
D

A

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE  
A4



UNIDADE  
mm

WEIGHT (kg)

0,086

# SELETOR DE LARANJA

MATERIAL:

AÇO SAE 1008

Projeto Integrador 2 - UnB

This drawing is our property; it can't be reproduced or communicated without our written agreement.

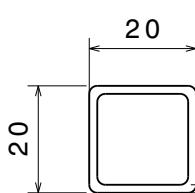
I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

D

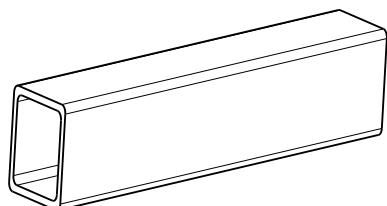
C

B

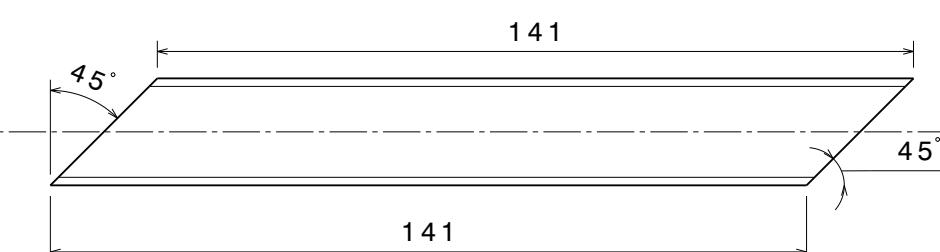
A



Vista frontal  
Escala: 1:1



Scale: 1:1



MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE  
A4

UNIDADE  
mm

# SELETOR DE LARANJA

MATERIAL:

AÇO SAE 1008

Projeto Integrador 2 - UnB

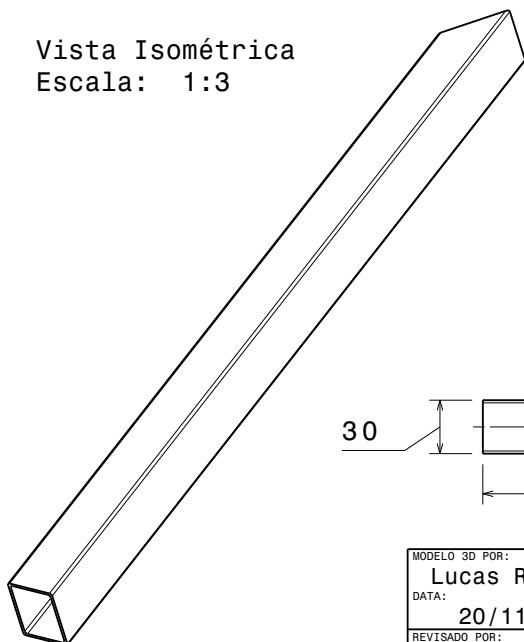
I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

This drawing is our property; it can't be reproduced or communicated without our written agreement.

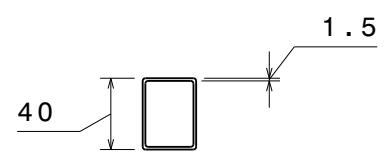
D

A

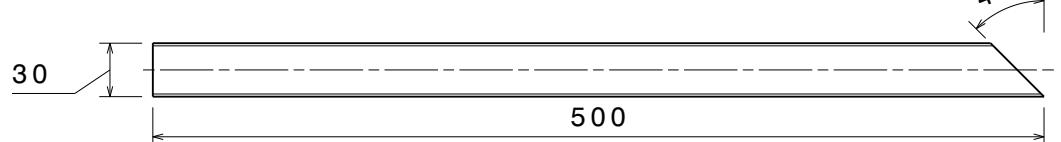
Vista Isométrica  
Escala: 1:3



Vista frontal  
Escala: 1:3



Vista superior  
Escala: 1:3

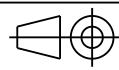


MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE

A4



## SELETOR DE LARANJA

MATERIAL:

AÇO SAE 1008

Projeto Integrador 2 - UnB

UNIDADE

mm

WEIGHT (kg)

0,759

NOME DO COMPONENTE:

Barra longitudinal

CÓDIGO

EST08

This drawing is our property; it can't be reproduced or communicated without our written agreement.

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

D

C

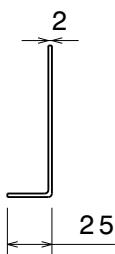
B

A

Vista isométrica  
Escala: 1:6

1525

Vista lateral  
Escala: 1:6



Vista frontal  
Escala: 1:3

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE  
A4

UNIDADE  
mm

# SELETOR DE LARANJA

MATERIAL:  
AÇO SAE 1008

Projeto Integrador 2 - UnB

NOME DO COMPONENTE:

ANTEPARO

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

This drawing is our property; it can't be reproduced or communicated without our written agreement.

D

A

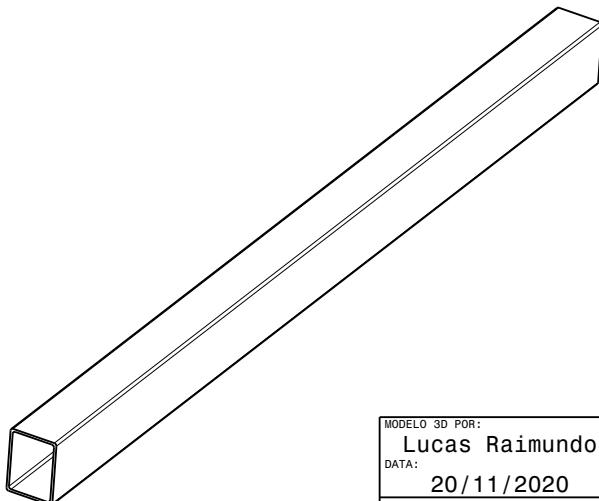
D C B A

4

600

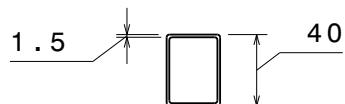
4

3



Vista Isométrica  
Escala: 1:3

Vista superior  
Escala: 1:3



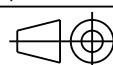
Vista frontal  
Escala: 1:3

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE

A4



## SELETOR DE LARANJA

MATERIAL:

AÇO SAE 1008

Projeto Integrador 2 - UnB

UNIDADE

mm

WEIGHT (kg)

0,939

NOME DO COMPONENTE:

BARRA LONGITUDINAL 2

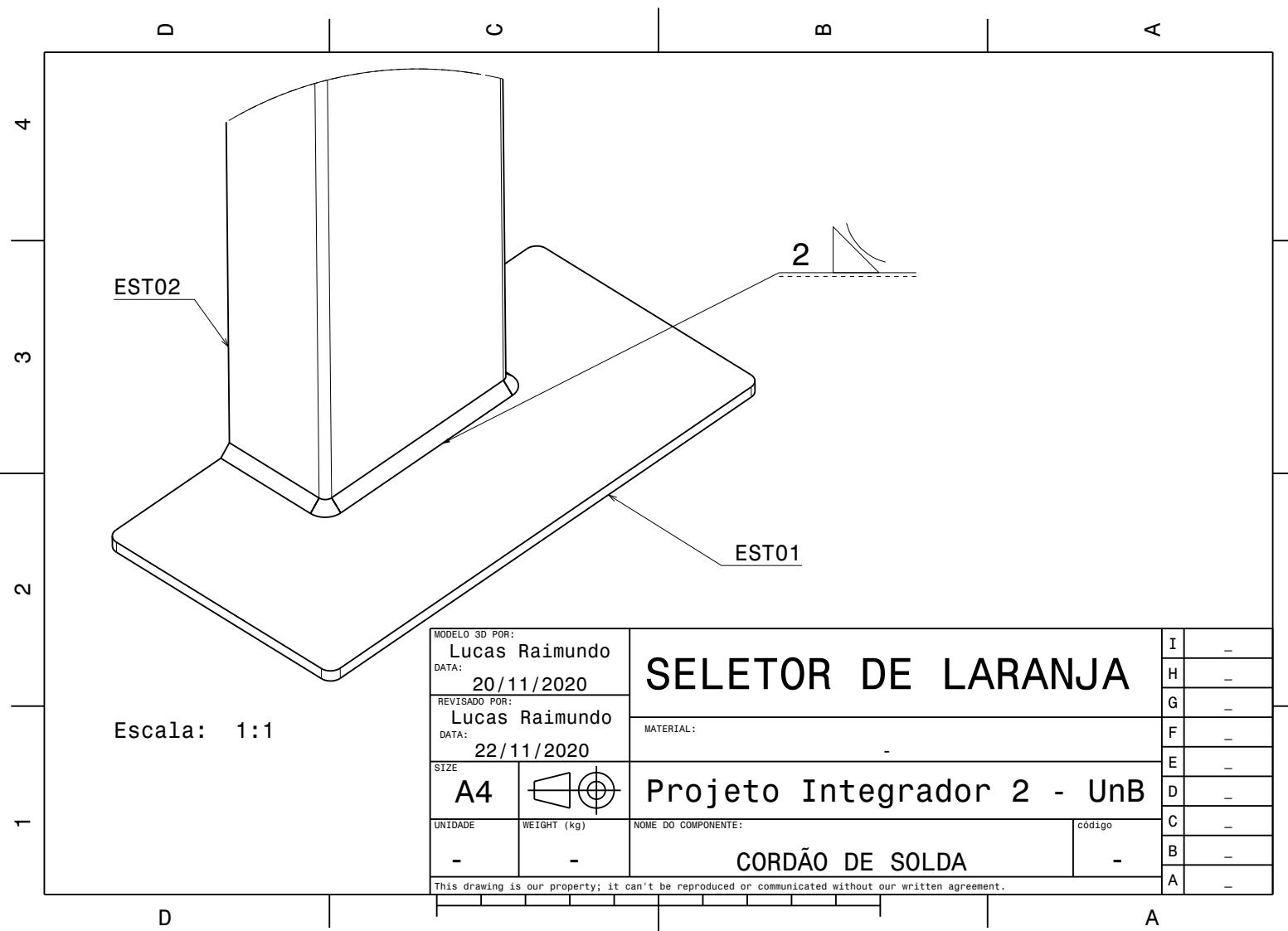
CÓDIGO

EST10

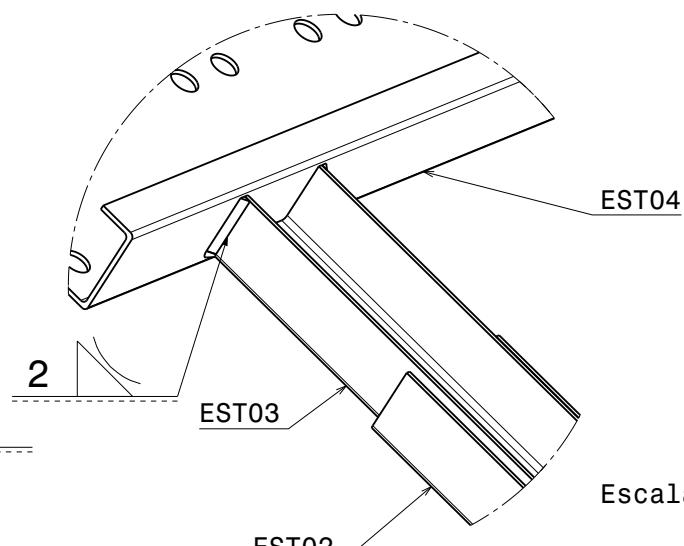
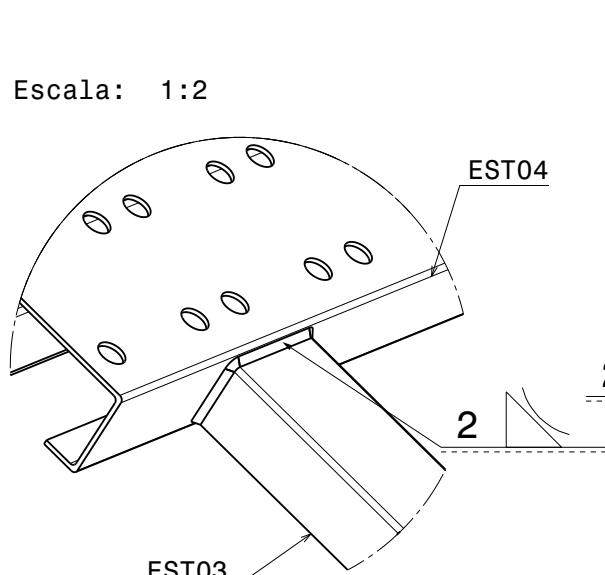
This drawing is our property; it can't be reproduced or communicated without our written agreement.

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

D A



Escala: 1:2



Escala: 1:2

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
22/11/2020

SIZE  
A4



## SELETOR DE LARANJA

MATERIAL:

Projeto Integrador 2 - UnB

UNIDADE  
-

WEIGHT (kg)  
-

NOME DO COMPONENTE:  
CORDÃO DE SOLDA

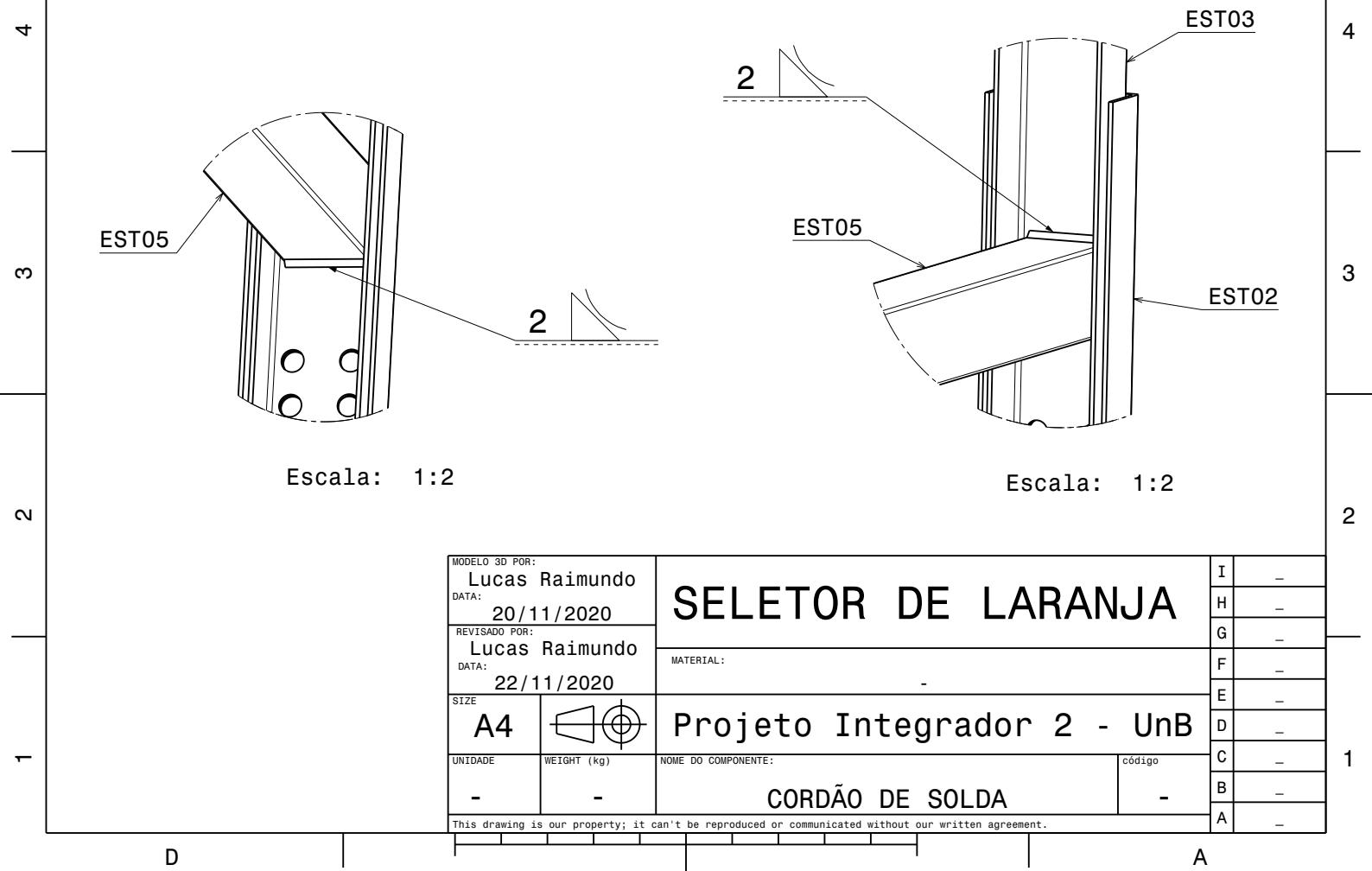
I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

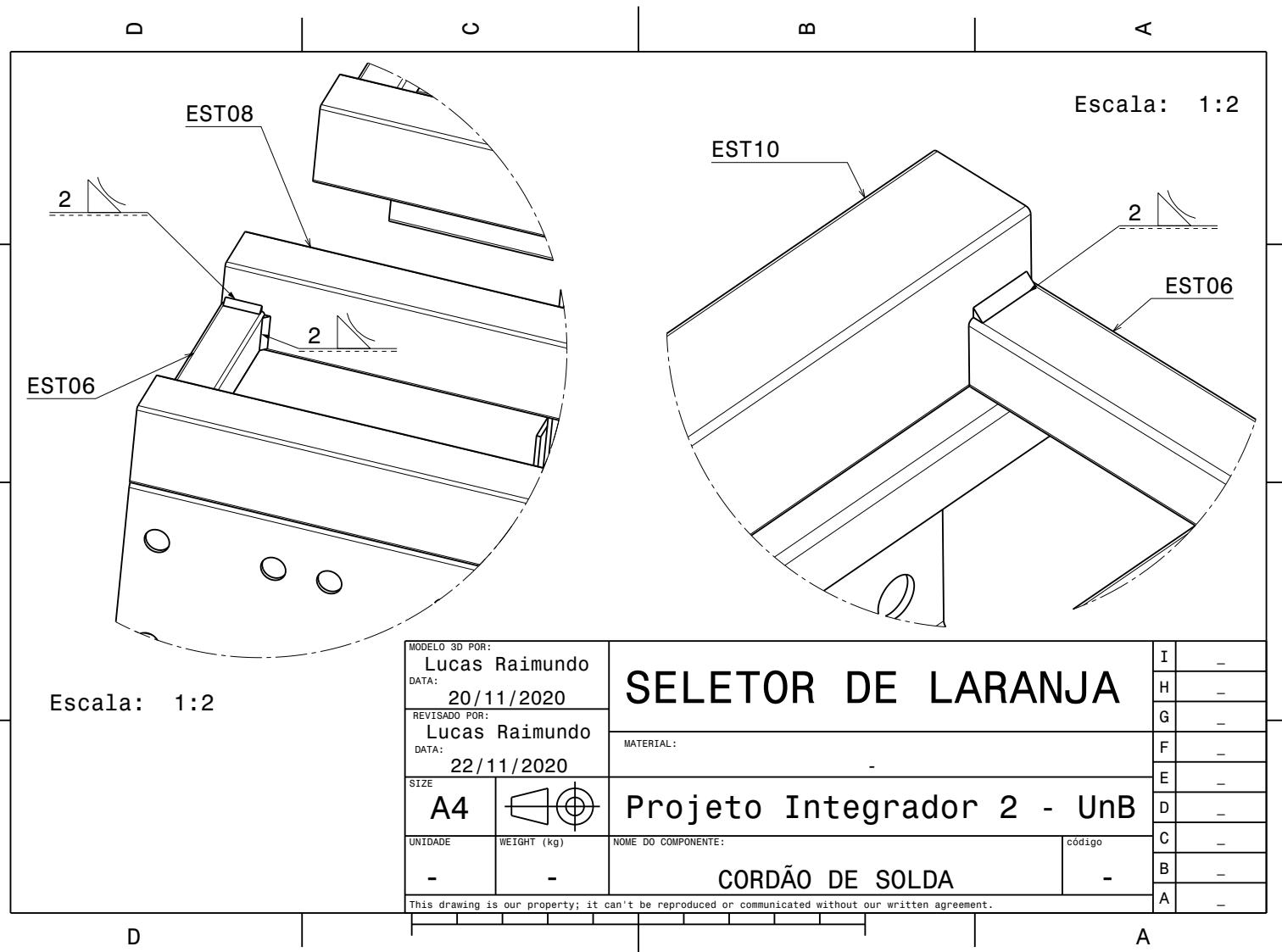
This drawing is our property; it can't be reproduced or communicated without our written agreement.

D

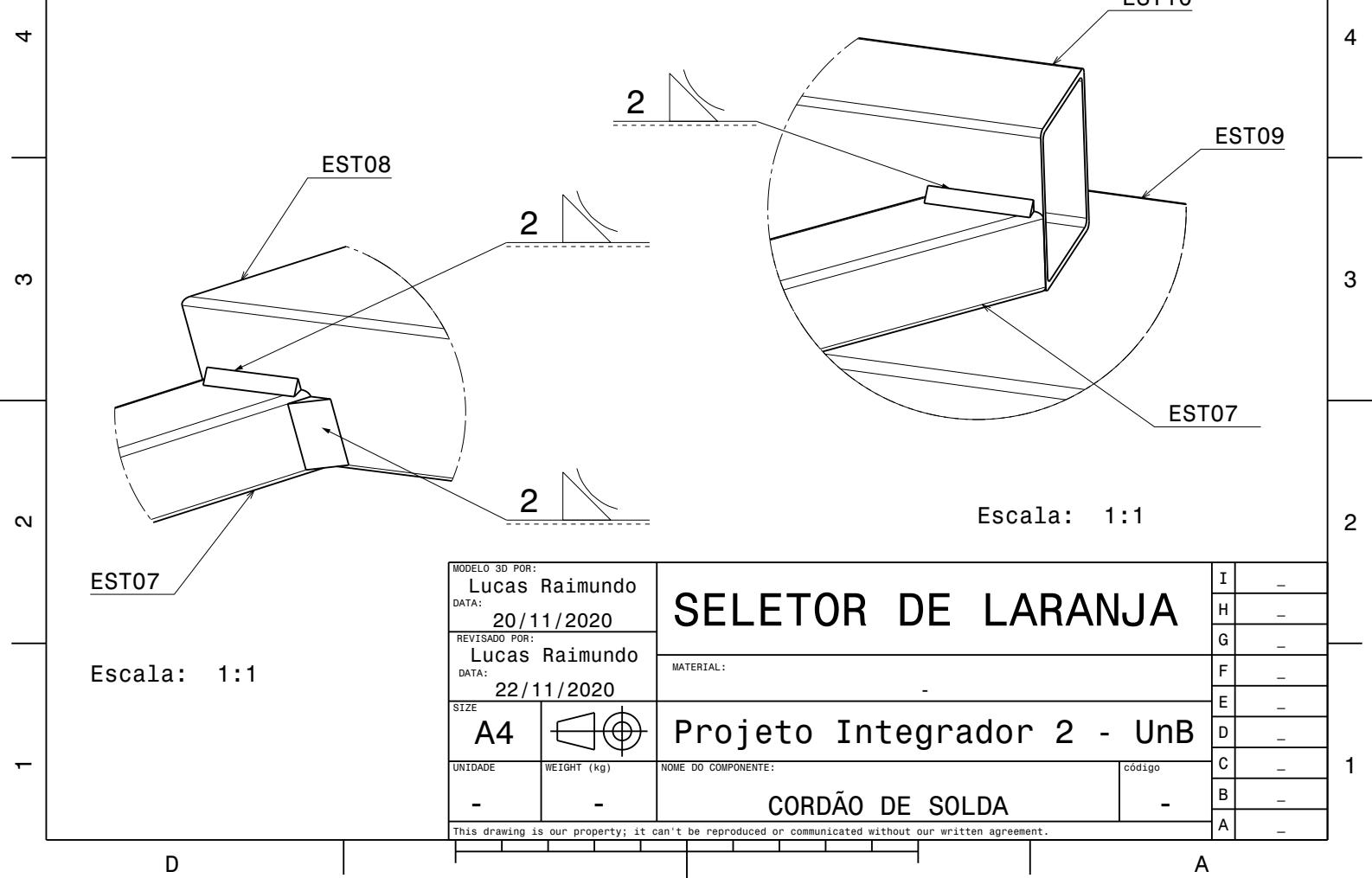
A

D C B A





D C B A



D C B A

4

D

C

B

A

3

4

3

2

2

1

1

0

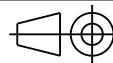
ROLETE DE AÇO TRACIONADO  
POR RODAS DENTADAS

Vista Isométrica  
Escala: 1:2

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
22/11/2020

SIZE  
A4



## SELETOR DE LARANJA

MATERIAL:

Projeto Integrador 2 - UnB

UNIDADE  
-

WEIGHT (kg)  
6,042

NOME DO COMPONENTE:

SISTEMA DE ROLETE ACIONADO

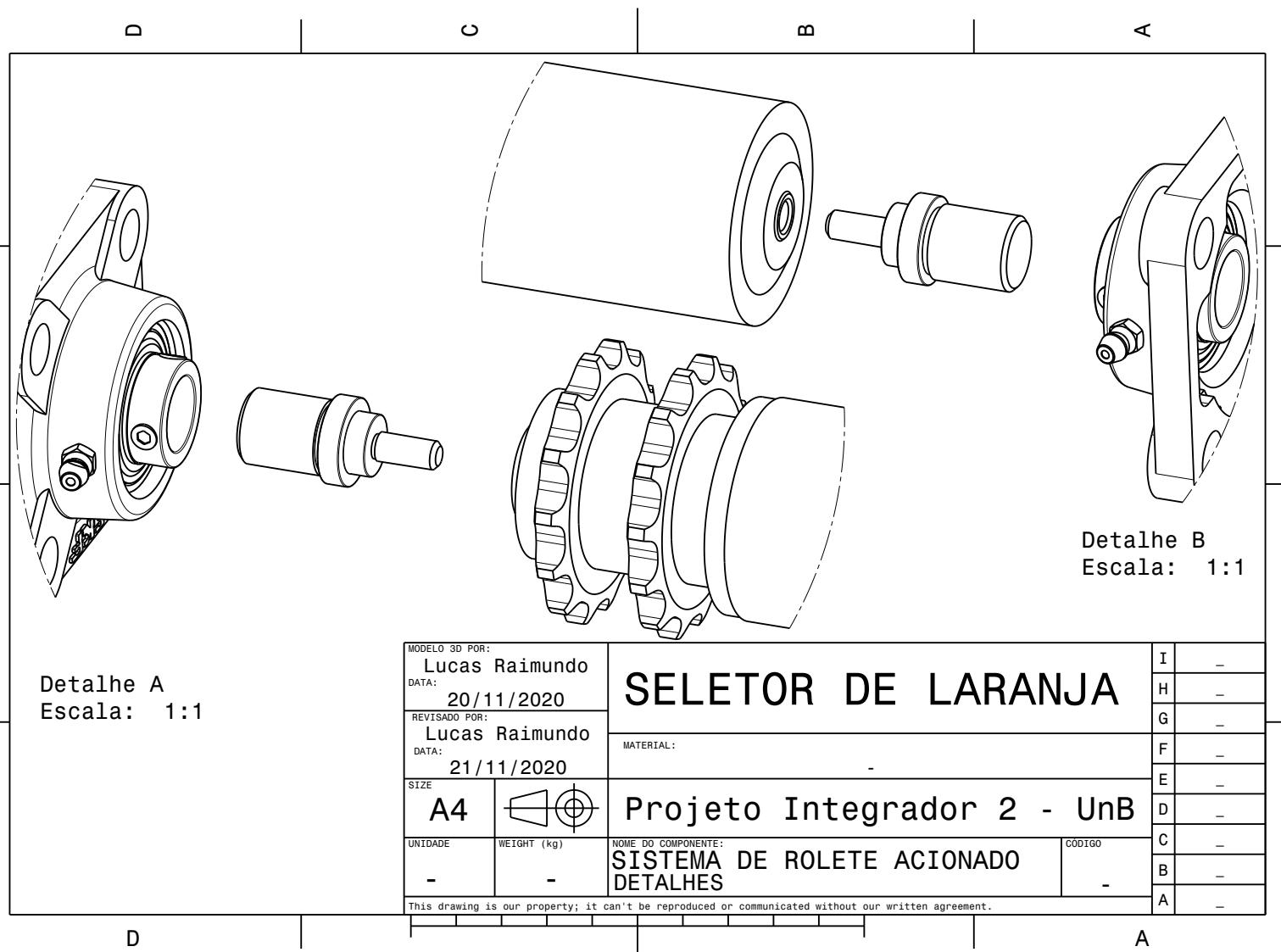
CÓDIGO  
TR

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

This drawing is our property; it can't be reproduced or communicated without our written agreement.

D

A



D C B A

4

3

2

1

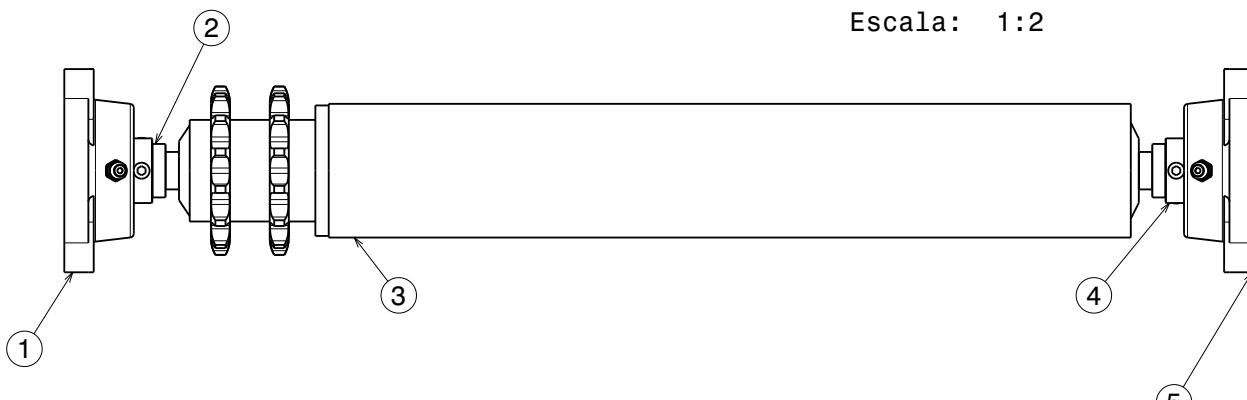
4

3

2

1

Vista Frontal  
Escala: 1:2



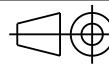
CONEXÃO ENTRE TR01- TR02  
PARAFUSO M10 SEXTAVADO  
DIN 931 - PASSO 1.5 mm

Nº	CÓDIGO	COMPONENTE
1	TR01	MANCAL
2	TR02	ÁRVORE
3	TR03	ROLETE
4	TR02	ÁRVORE
5	TR01	MANCAL

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
22/11/2020

SIZE  
A4



## SELETOR DE LARANJA

Projeto Integrador 2 - UnB

UNIDADE  
-

WEIGHT (kg)  
6,042

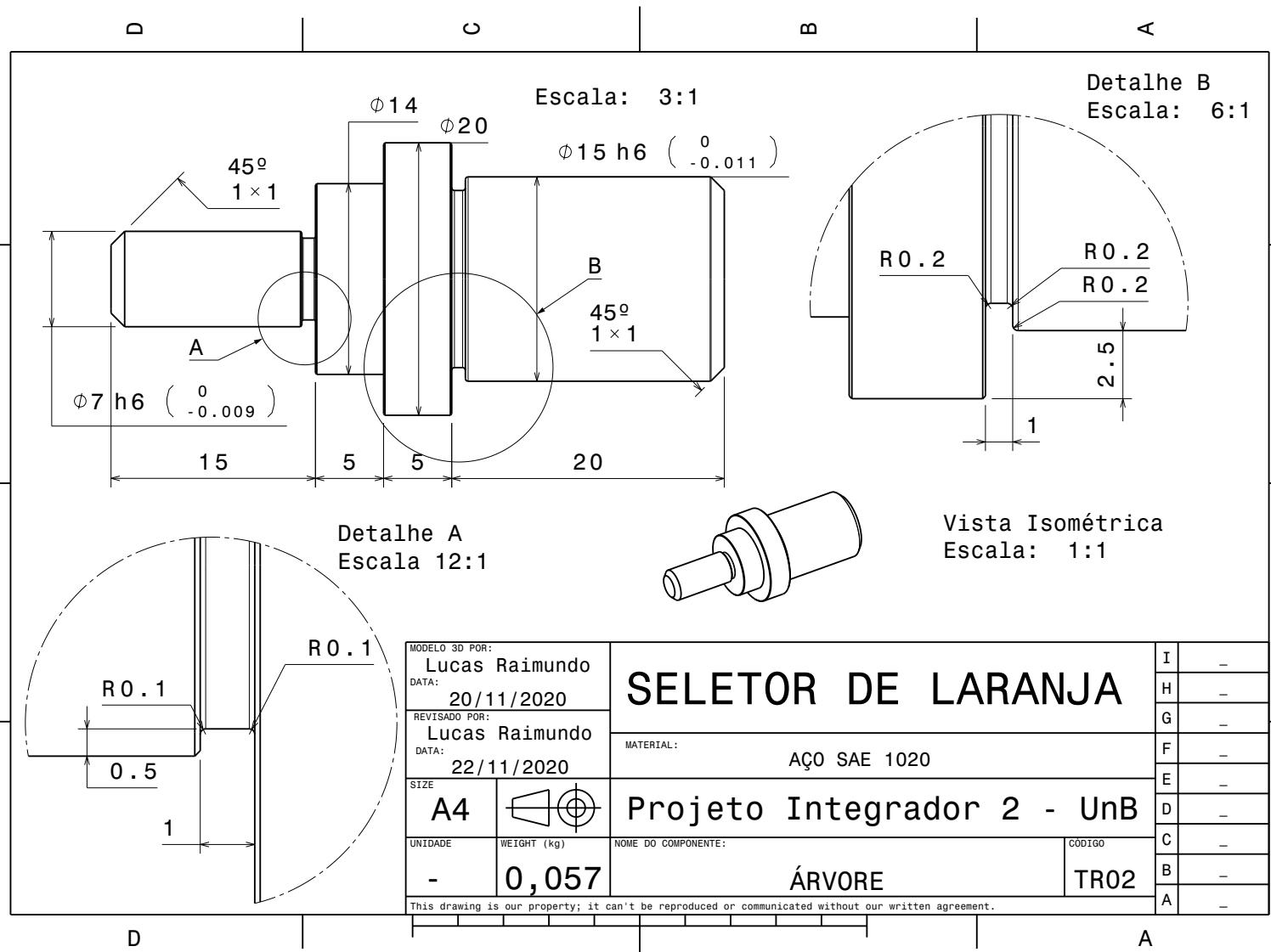
NOME DO COMPONENTE:  
SISTEMA DE ROLETE ACIONADO  
COMPONENTES

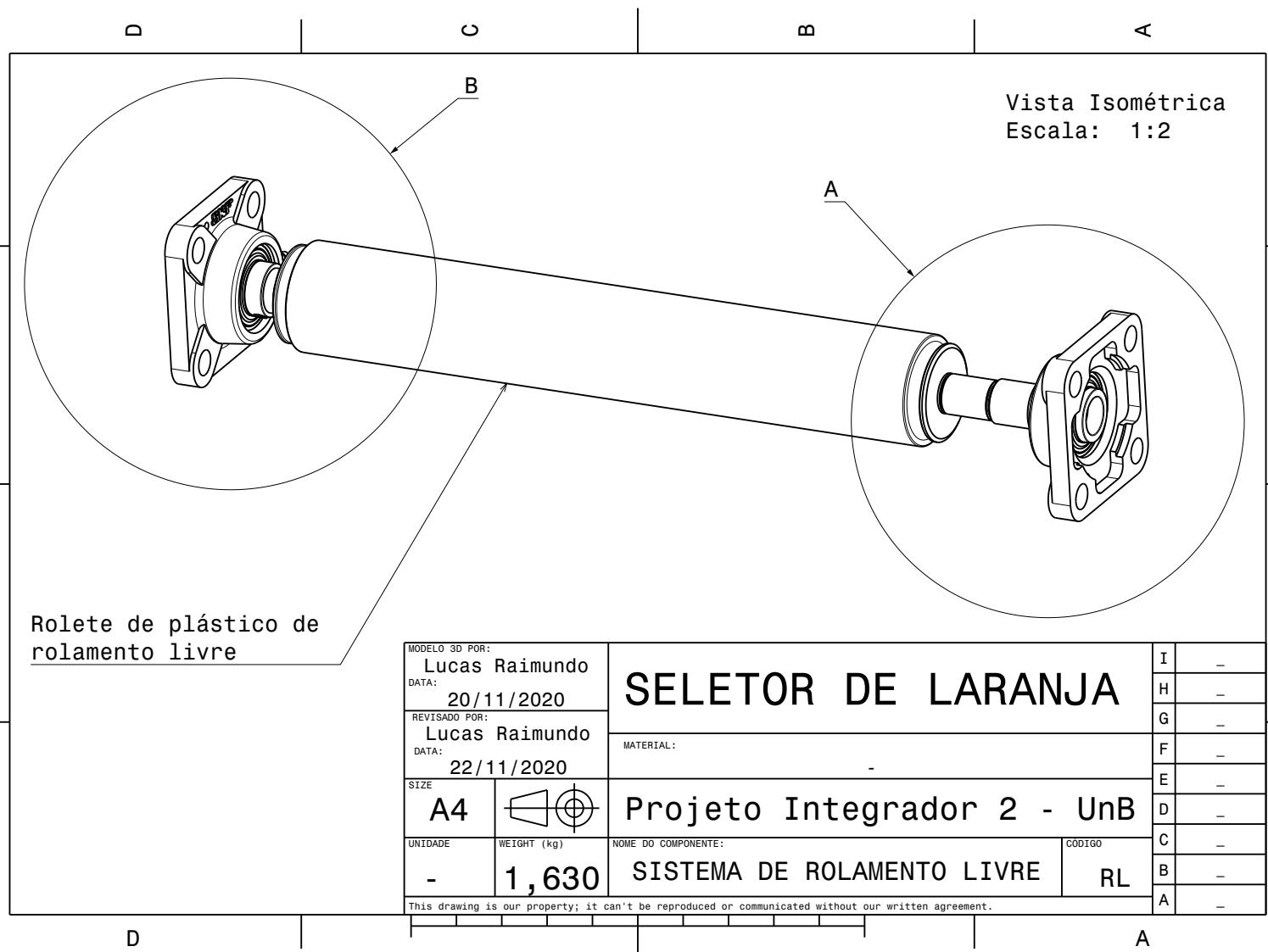
CÓDIGO  
TR

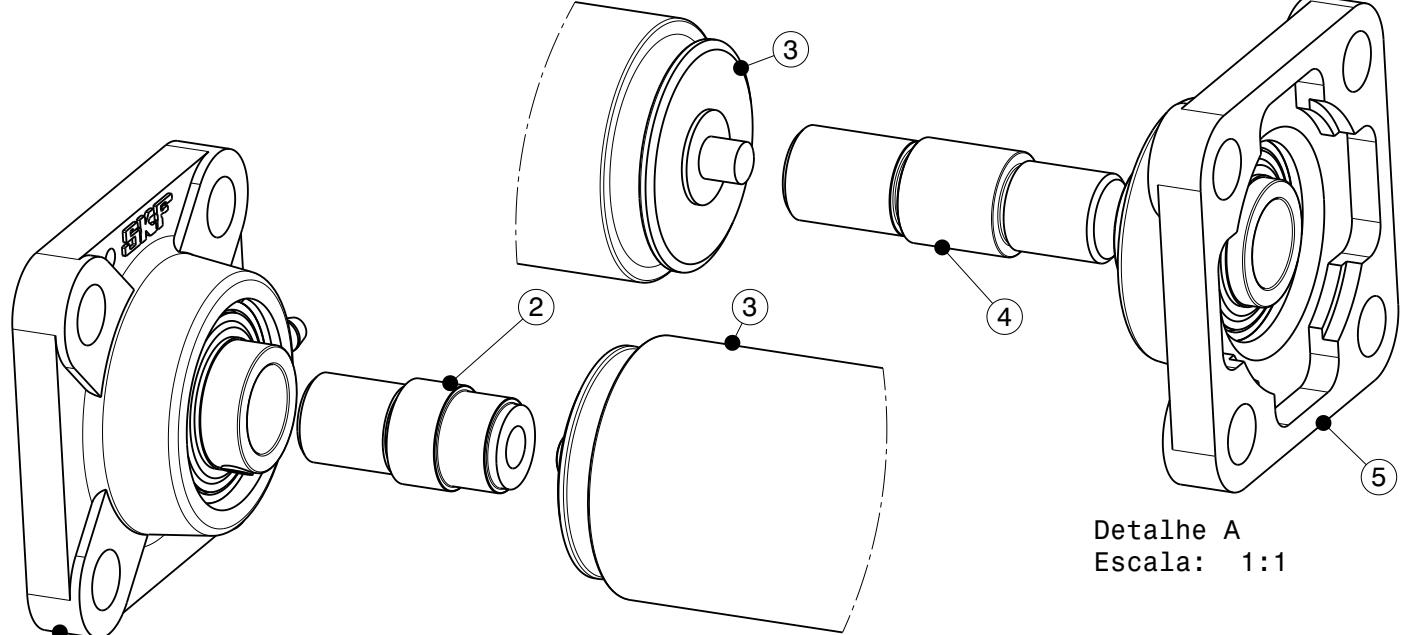
I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

This drawing is our property; it can't be reproduced or communicated without our written agreement.

D A







Detalhe A  
Escala: 1:1

Detalhe B  
Escala: 1:1

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
21/11/2020

SIZE  
A4

UNIDADE  
-

WEIGHT (kg)  
-

## SELETOR DE LARANJA

MATERIAL:

Projeto Integrador 2 - UnB

DETALHES DO SISTEMA  
DE ROLAMENTO LIVRE

CÓDIGO  
-

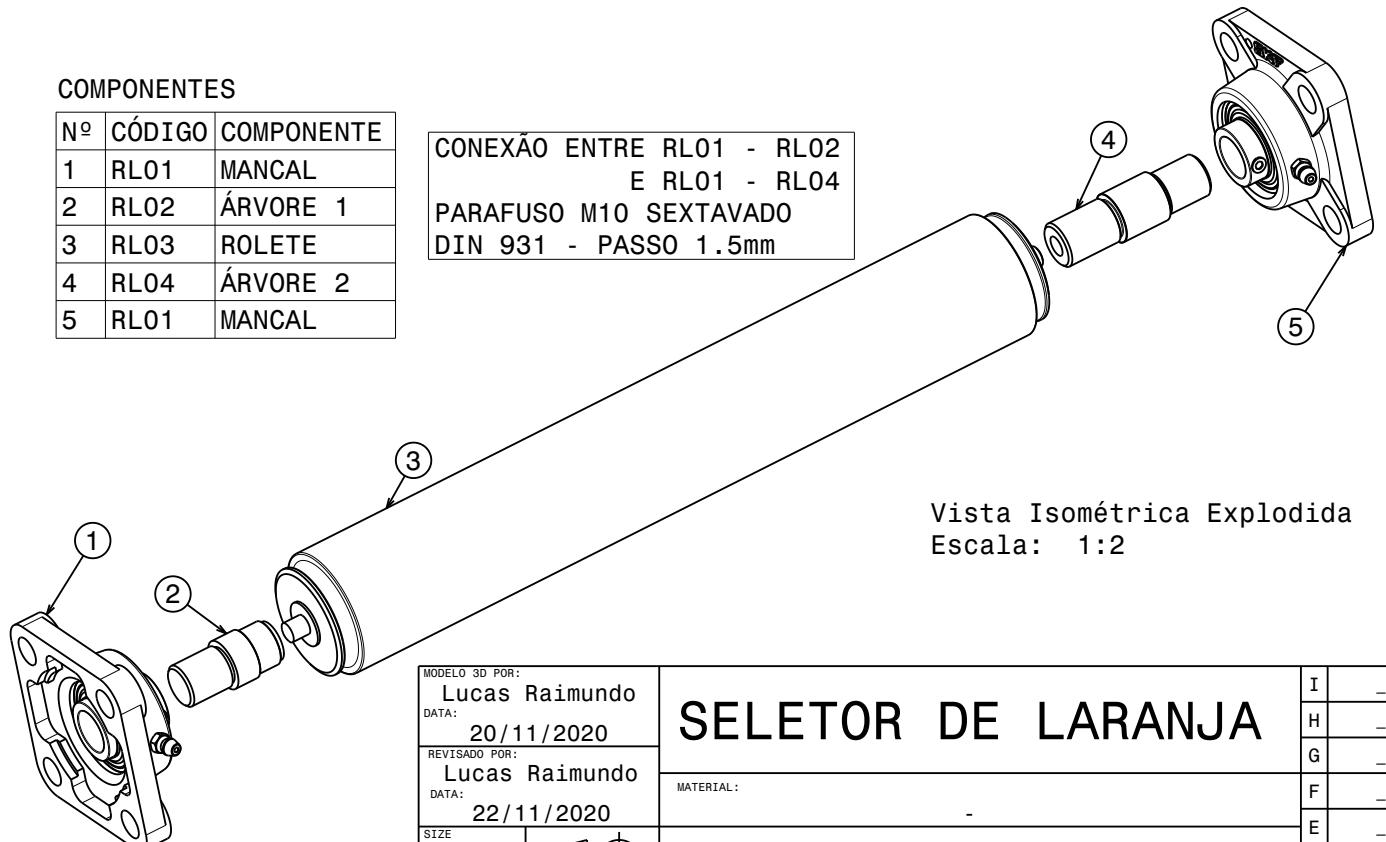
This drawing is our property; it can't be reproduced or communicated without our written agreement.

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-

### COMPONENTES

Nº	CÓDIGO	COMPONENTE
1	RL01	MANCAL
2	RL02	ÁRVORE 1
3	RL03	ROLETE
4	RL04	ÁRVORE 2
5	RL01	MANCAL

CONEXÃO ENTRE RL01 - RL02  
E RL01 - RL04  
PARAFUSO M10 SEXTAVADO  
DIN 931 - PASSO 1.5mm



Vista Isométrica Explodida  
Escala: 1:2

MODELO 3D POR:  
Lucas Raimundo  
DATA:  
20/11/2020

REVISADO POR:  
Lucas Raimundo  
DATA:  
22/11/2020

SIZE  
A4

### SELETOR DE LARANJA

MATERIAL:

Projeto Integrador 2 - UnB

UNIDADE  
- WEIGHT (kg)  
- 1,630

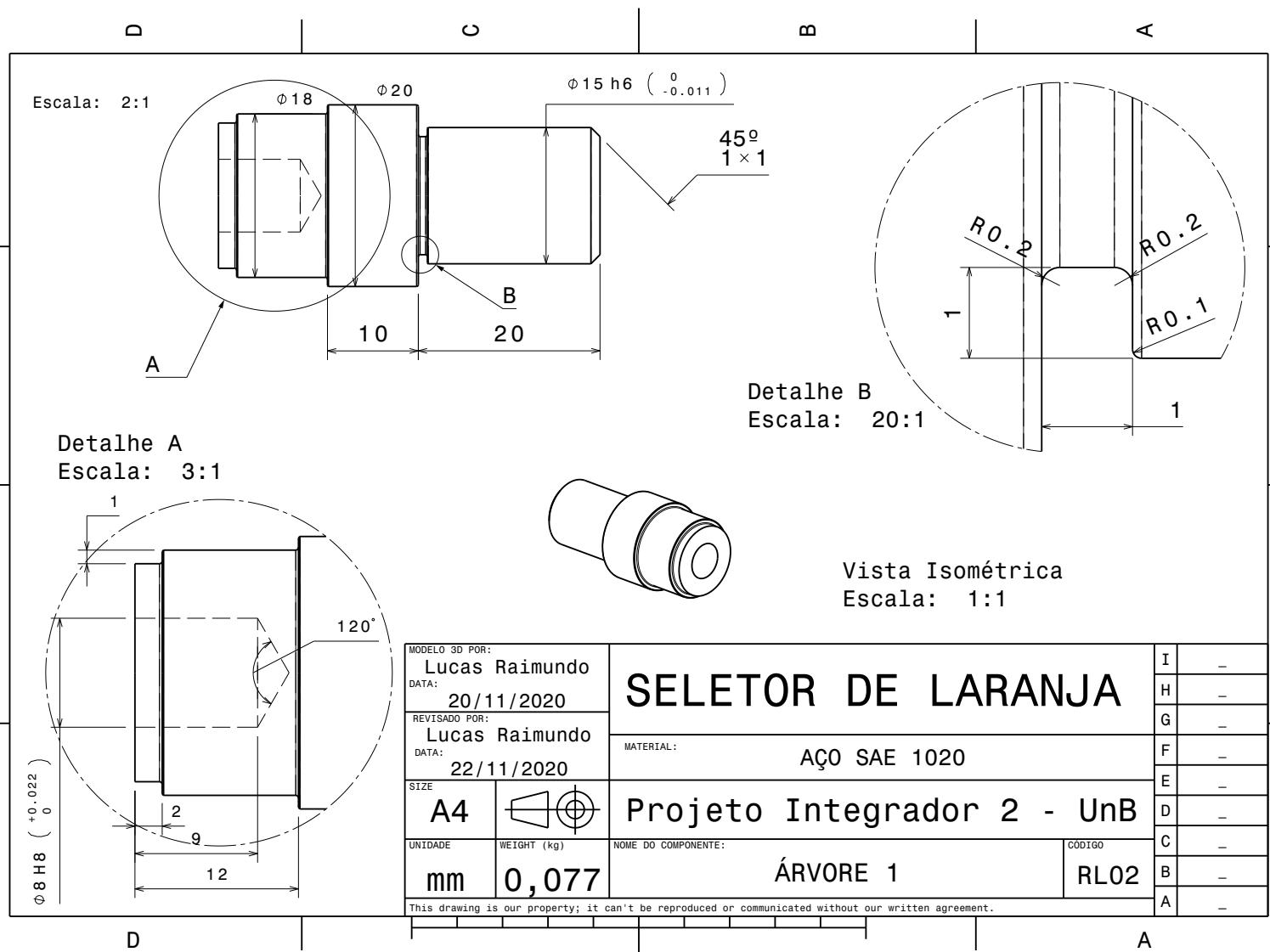
NOME DO COMPONENTE:

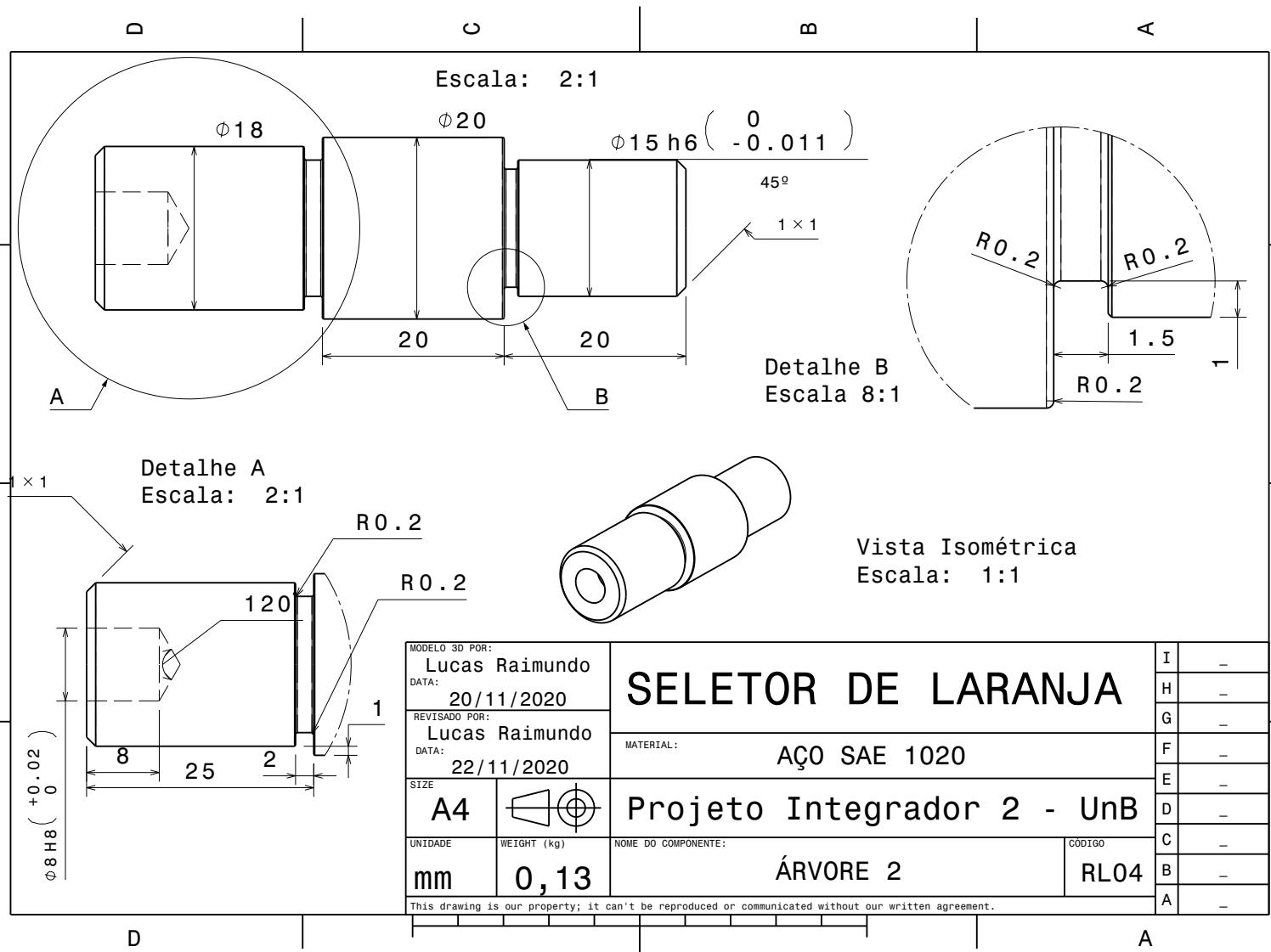
ROLETE DE ROLAMENTO LIVRE

CÓDIGO  
RL

This drawing is our property; it can't be reproduced or communicated without our written agreement.

I	-
H	-
G	-
F	-
E	-
D	-
C	-
B	-
A	-





# APÊNDICE K – Especificações técnicas do mancal



## FY 503 M

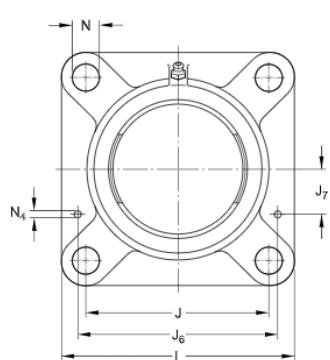
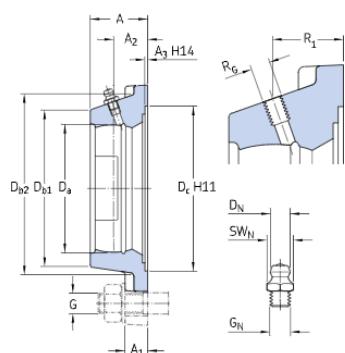
Caixas de mancal flangeadas quadradas para rolamentos série Y

### Especificações técnicas

Material

Ferro fundido

### DIMENSÕES



D <sub>a</sub>	40 mm
D <sub>b1</sub>	50 mm
D <sub>b2</sub>	53 mm
D <sub>c</sub>	55.6 mm
A	26 mm
A <sub>1</sub>	11 mm
A <sub>2</sub>	17 mm
A <sub>3</sub>	3.2 mm
J	54 mm
L	76 mm
N	11.5 mm

### FURO ROSCADO

R <sub>G</sub>	1/4-28 UNF
R <sub>1</sub>	19 mm

## PINO GRAXEIRO

D <sub>N</sub>	6.5 mm
S <sub>W</sub> <sub>N</sub>	7 mm
G <sub>N</sub>	1/4-28 SAE-LT

## PINOS DE AJUSTE

J <sub>6</sub>	66 mm
J <sub>7</sub>	14.5 mm
N <sub>4</sub>	4 mm

## MASSA

Massa do mancal	0.28 kg
-----------------	---------

## INFORMAÇÕES DE MONTAGEM

Recommended diameter for attachment bolts, mm	G	10 mm
Recommended diameter for attachment bolts, inch	G	0.375 in

## More information

Product details	Engineering information	Tools
<a href="#">Flanged units to ISO standards</a>	<a href="#">Principles of ball bearing units selection and application - Download catalogue (4.9 MB)</a>	<a href="#">Bearing Select</a>
<a href="#">Flanged units to North American standards</a>	<a href="#">Principles of selecting mounted bearing solutions</a>	<a href="#">Engineering Calculator</a>
<a href="#">Flanged units to Japanese Industrial Standards (JIS)</a>	<a href="#">Falha do rolamento e como evitá-la</a>	<a href="#">Instruções para montagem e desmontagem de rolamentos</a>
<a href="#">Flanged units, not standardized (SKF Food Line)</a>		
<a href="#">Ball bearing unit data</a>		
<a href="#">Loads</a>		
<a href="#">Temperature limits</a>		
<a href="#">Permissible speed</a>		
<a href="#">Lubrication</a>		
<a href="#">Mounting and dismounting</a>		
<a href="#">Designation system</a>		

# Termos e condições

Ao acessar e utilizar este site/aplicativo do Grupo SKF, o que inclui a AB SKF e/ou qualquer uma de suas afiliadas ("SKF"), você concorda com os seguintes termos e condições

## Exclusão da garantia e limitação de responsabilidade

Embora todo cuidado tenha sido tomado para assegurar a precisão das informações deste site/aplicativo, a SKF fornece essas informações "NO ESTADO" e SEM QUAISQUER GARANTIAS, EXPRESSAS OU IMPLÍCITAS, INCLUINDO, ENTRE OUTRAS, GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E ADEQUAÇÃO PARA UM DETERMINADO PROPÓSITO. Você reconhece que o uso deste site/aplicativo é um risco unicamente seu, que você assume total responsabilidade por todos os custos associados ao uso do site/aplicativo e que a SKF não será responsável por quaisquer danos diretos, incidentais, consequentes ou indiretos de qualquer espécie decorrentes de seu acesso ou uso das informações ou software disponibilizados no site/aplicativo. Quaisquer garantias e representações neste site/aplicativo em relação a produtos ou serviços da SKF adquiridos ou utilizados por você estarão sujeitas aos termos e condições acordados no contrato do referido produto ou serviço. Além disso, para sites/aplicativos que não sejam da SKF e que sejam referidos em nosso site/aplicativo ou onde haja um hiperlink, a SKF não dá garantias relativas à precisão ou confiabilidade das informações desses sites/aplicativos, não assumindo qualquer responsabilidade por materiais criados ou publicados por terceiros ali contidos. Ademais, a SKF não garante que este site/aplicativo ou outros sites/aplicativos vinculados não contenham vírus ou outros elementos nocivos.

## Copyright

Os direitos autorais deste site/aplicativo e os direitos autorais das informações e software disponibilizados neste site/aplicativo pertencem à SKF ou seus licenciadores. Todos os direitos são reservados. Todo o material licenciado faz referência ao licenciador que cedeu à SKF o direito de utilizar o material. As informações e o software disponibilizados neste site/aplicativo não podem ser reproduzidos, duplicados, copiados, transferidos, distribuídos, armazenados, modificados, transferidos por download ou explorados de qualquer outra forma, para qualquer uso comercial, sem aprovação prévia por escrito da SKF. No entanto, eles podem ser reproduzidos, armazenados e transferidos por download para uso por pessoas, sem a aprovação prévia, por escrito, da SKF. Sob nenhuma circunstância, essas informações ou esse software podem ser fornecidos a terceiros.

Este site/aplicativo inclui determinadas imagens usadas sob licença da Shutterstock, Inc.

## Marcas e patentes

Todas as marcas comerciais, nomes de fantasia e logotipos corporativos exibidos no site/aplicativo são propriedade da SKF ou de seus licenciadores, não podendo ser utilizados de qualquer forma sem a aprovação prévia por escrito da SKF. Todas as marcas comerciais licenciadas publicadas neste site/aplicativo fazem referência ao licenciador que cedeu à SKF o direito de utilizar a marca comercial. O acesso a este site/aplicativo não concede ao usuário qualquer licença sob quaisquer patentes pertencentes ou licenciadas à SKF.

## Alterações

A SKF reserva-se o direito de fazer alterações ou acréscimos neste site/aplicativo a qualquer momento.

# APÊNDICE L – Especificações técnicas do rolamento



# 6002-2Z

Item popular  
SKF Explorer

## Rolamentos rígidos de esferas

### Especificações do rolamento

#### Tolerâncias,

Normal (métrico), P6, P5, Normal (polegadas),

#### Folga interna radial,

Pares de rolamentos pareados, Aço inoxidável d < 10 mm, Outros rolamentos

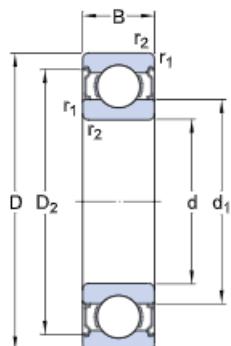
### Interfaces do rolamento

#### Tolerâncias de assento para condições padrão,

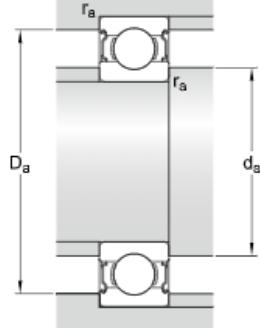
#### Tolerâncias e ajustes resultantes

## Especificações técnicas

### DIMENSÕES



d	15 mm
D	32 mm
B	9 mm
d <sub>1</sub>	≈ 20.5 mm
D <sub>2</sub>	≈ 28.2 mm
r <sub>1,2</sub>	min. 0.3 mm



### DIMENSÕES DO ENCOSTO

d <sub>a</sub>	min. 17 mm
d <sub>a</sub>	max. 20.4 mm
D <sub>a</sub>	max. 30 mm
r <sub>a</sub>	max. 0.3 mm

## DADOS DE CÁLCULO

Classificação de carga dinâmica básica	C	5.85 kN
Classificação de carga estática básica	$C_0$	2.85 kN
Limite de carga de fadiga	$P_u$	0.12 kN
Velocidade de referência		50000 r/min
Velocidade-limite		26000 r/min
Fator de cálculo	$k_r$	0.025
Fator de cálculo	$f_0$	14

## MASSA

Massa do rolamento	0.032 kg
--------------------	----------

## Mais informações

Detalhes do produto	Informações de engenharia	Ferramentas
<a href="#">Rolamentos rígidos de uma carreira de esferas</a>	<a href="#">Princípios da seleção de rolamentos</a>	<a href="#">Bearing Select</a>
<a href="#">Rolamentos rígidos de esferas de aço inoxidável</a>	<a href="#">Conhecimentos gerais sobre rolamentos</a>	<a href="#">SimPro Quick</a>
<a href="#">Rolamentos rígidos de uma carreira de esferas com rasgos de entrada</a>	<a href="#">Processo de seleção de rolamentos</a>	<a href="#">Engineering Calculator</a>
<a href="#">Rolamentos rígidos de duas carreiras de esferas</a>	<a href="#">Interfaces do rolamento</a>	<a href="#">LubeSelect para graxas SKF</a>
<a href="#">Especificações do rolamento</a>	<a href="#">Tolerâncias de assento para condições padrão</a>	<a href="#">Ferramenta para seleção de aquecedor</a>
<a href="#">Cargas</a>	<a href="#">Seleção de folga interna</a>	<a href="#">Instruções para montagem e desmontagem de rolamentos</a>
<a href="#">Limites de temperatura</a>	<a href="#">Lubrificação</a>	
<a href="#">Velocidade permitida</a>	<a href="#">Vedações, montagem e desmontagem</a>	
<a href="#">Sistema de designação</a>	<a href="#">Falha do rolamento e como evitá-la</a>	

# Termos e condições

Ao acessar e utilizar este site/aplicativo do Grupo SKF, o que inclui a AB SKF e/ou qualquer uma de suas afiliadas ("SKF"), você concorda com os seguintes termos e condições

## Exclusão da garantia e limitação de responsabilidade

Embora todo cuidado tenha sido tomado para assegurar a precisão das informações deste site/aplicativo, a SKF fornece essas informações "NO ESTADO" e SEM QUAISQUER GARANTIAS, EXPRESSAS OU IMPLÍCITAS, INCLUINDO, ENTRE OUTRAS, GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E ADEQUAÇÃO PARA UM DETERMINADO PROPÓSITO. Você reconhece que o uso deste site/aplicativo é um risco unicamente seu, que você assume total responsabilidade por todos os custos associados ao uso do site/aplicativo e que a SKF não será responsável por quaisquer danos diretos, incidentais, consequentes ou indiretos de qualquer espécie decorrentes de seu acesso ou uso das informações ou software disponibilizados no site/aplicativo. Quaisquer garantias e representações neste site/aplicativo em relação a produtos ou serviços da SKF adquiridos ou utilizados por você estarão sujeitas aos termos e condições acordados no contrato do referido produto ou serviço. Além disso, para sites/aplicativos que não sejam da SKF e que sejam referidos em nosso site/aplicativo ou onde haja um hiperlink, a SKF não dá garantias relativas à precisão ou confiabilidade das informações desses sites/aplicativos, não assumindo qualquer responsabilidade por materiais criados ou publicados por terceiros ali contidos. Ademais, a SKF não garante que este site/aplicativo ou outros sites/aplicativos vinculados não contenham vírus ou outros elementos nocivos.

## Copyright

Os direitos autorais deste site/aplicativo e os direitos autorais das informações e software disponibilizados neste site/aplicativo pertencem à SKF ou seus licenciadores. Todos os direitos são reservados. Todo o material licenciado faz referência ao licenciador que cedeu à SKF o direito de utilizar o material. As informações e o software disponibilizados neste site/aplicativo não podem ser reproduzidos, duplicados, copiados, transferidos, distribuídos, armazenados, modificados, transferidos por download ou explorados de qualquer outra forma, para qualquer uso comercial, sem aprovação prévia por escrito da SKF. No entanto, eles podem ser reproduzidos, armazenados e transferidos por download para uso por pessoas, sem a aprovação prévia, por escrito, da SKF. Sob nenhuma circunstância, essas informações ou esse software podem ser fornecidos a terceiros.

Este site/aplicativo inclui determinadas imagens usadas sob licença da Shutterstock, Inc.

## Marcas e patentes

Todas as marcas comerciais, nomes de fantasia e logotipos corporativos exibidos no site/aplicativo são propriedade da SKF ou de seus licenciadores, não podendo ser utilizados de qualquer forma sem a aprovação prévia por escrito da SKF. Todas as marcas comerciais licenciadas publicadas neste site/aplicativo fazem referência ao licenciador que cedeu à SKF o direito de utilizar a marca comercial. O acesso a este site/aplicativo não concede ao usuário qualquer licença sob quaisquer patentes pertencentes ou licenciadas à SKF.

## Alterações

A SKF reserva-se o direito de fazer alterações ou acréscimos neste site/aplicativo a qualquer momento.