

## Controle Gerencial Londricostura

Versão: 1.1

## Especificação Complementar

Data: 26/02/2025

---

### Histórico da Revisão

Data	Versão	Descrição	Autor
26/02/2025	1.0	Preenchimento inicial	Victor Cottar Marçal Silva
08/03/2025	1.1	Ajustes no documento	Victor Cottar Marçal Silva

---

### Índice

1. Introdução
  - 1.1 Objetivo
  - 1.2 Escopo
  - 1.3 Definições, Acrônimos e Abreviações
  - 1.4 Referências
  - 1.5 Visão Geral
2. Funcionalidade
  - 2.1 Processar dados
  - 2.2 Dados
  - 2.3 Alertas
  - 2.3 Indicadores
  - 2.3 Relatórios
  - 2.3 Visualização de dados do Marketplace
3. Utilidade
  - 3.1 <Requisito de Utilidade Um>
4. Confiabilidade
  - 4.1 <Requisito de Confiabilidade Um>

- 5. Desempenho
    - 5.1 <Requisito de Desempenho Um>
  - 6. Suportabilidade
  - 7. Documentação do Usuário On-line e Requisitos do Sistema de Ajuda
  - 8. Interfaces
    - 8.1 Interfaces com o Usuário
    - 8.2 Interfaces de Hardware
    - 8.3 Interfaces de Software
  - 9. Observações Legais, sobre Direitos Autorais e Outras Observações
  - 10. Padrões Aplicáveis
- 

## Introdução

Essa seção tem por objetivo introduzir de uma maneira sucinta o escopo do documento.

---

## Objetivo

**Este documento de Especificação Complementar tem como objetivo detalhar requisitos não funcionais, restrições, regras de negócio e outras informações adicionais que complementam a Especificação de Requisitos do Sistema.**

## Escopo

Este documento especifica os atributos relacionados ao desenvolvimento do projeto Controle Gerencial Londricostura. O sistema em questão será responsável por modernizar os processos da empresa, que hoje utiliza planilhas Excel para realizar algumas ações.

---

## **Definições, Acrônimos e Abreviações**

**Pode ser visualizado no documento de Glossário.**

## **Referências**

Não foram utilizadas referências para a criação deste documento.

---

## **Visão Geral**

Este documento contém uma introdução sobre a Especificação Suplementar que será apresentada. Após, descreve os requisitos funcionais, desempenho, suportabilidade, usabilidade, confiabilidade,

---

## **Funcionalidade**

Essa seção apresenta os requisitos não funcionais do projeto em questão.

### **Processar dados**

O sistema deverá processar e padronizar os dados recebidos via arquivo .xlsm, onde esses dados serão inseridos no banco de dados. Além disso, o sistema permitirá a visualização e alteração dos dados inseridos. Caso o usuário não queira utilizar a planilha, ele também poderá inserir, visualizar e modificar os dados diretamente no sistema, sem a necessidade do arquivo .xlsm.

---

### **Dados**

O sistema deverá permitir visualizar os dados enviados via arquivo .xlsm. No entanto, caso o usuário opte por não utilizar a planilha, ele também poderá visualizar os dados diretamente no sistema, mantendo a flexibilidade de uso.

---

### **Alertas**

A aplicação terá alertas de quando uma ação foi concluída com sucesso, melhorando a usabilidade e acessibilidade do sistema.

---

#### Indicadores

A aplicação terá indicadores que facilitará a tomada de decisão, realização de ações para recuperação de clientes etc.

---

#### Relatórios

A aplicação terá a funcionalidade de impressão de relatórios, permitindo que os usuários gerem e imprimam relatórios personalizados sobre os dados registrados no sistema.

---

#### Visualização de dados de MarketPlace

A aplicação terá a funcionalidade de visualização de dados das vendas feitas pelo Marketplace.

---

### Utilidade

Usuários necessitaram de um treinamento curto, para demonstração das telas e funcionalidades do sistema.

---

### Confiabilidade

O sistema deve estar disponível 24 horas por dia, 7 dias na semana, se o servidor local tiver acesso a internet e energia elétrica. O tempo máximo aceitável para recuperação após uma falha crítica não deverá ultrapassar 20 minutos. Logs de erros e eventos críticos serão registrados para análise e melhoria contínua da estabilidade do sistema.

---

## Desempenho

O desempenho do sistema deve ser avaliado continuamente durante a fase de implementação, garantindo eficiência no processamento dos dados e tempos de resposta adequados para as operações realizadas pelos usuários.

---

### Otimização do código

Desenvolver o sistema utilizando algoritmos eficientes, evitando a execução de operações redundantes ou desnecessárias, a fim de garantir uma performance ágil e otimizada.

---

### Banco de dados

Priorizar consultas otimizadas e a estruturação adequada das tabelas, garantindo que o acesso e manipulação de dados sejam rápidos e eficazes, mesmo em cenários de alta carga.

---

### Velocidade da internet

Considerar diferentes tipos de conexão à internet e otimizar o carregamento de recursos, como imagens, scripts e outros elementos, para garantir uma navegação fluida e responsiva, mesmo em conexões mais lentas.

---

## Suportabilidade

O usuário deverá ser capaz de acessar o sistema por meio do navegador web. O suporte ao sistema é vitalício e realizado no mesmo dia ou semana.

---

## **Documentação do Usuário On-line e Requisitos do Sistema de Ajuda**

Será disponibilizado um Manual para o Usuário, onde o mesmo terá explicações visuais sobre as telas e o funcionamento das mesmas.

---

## **Interfaces**

Essa seção tem por objetivo descrever os três tipos de interface do sistema.

### **Interfaces com o Usuário**

O sistema deve oferecer uma interface gráfica web intuitiva e responsiva, permitindo a interação eficiente do usuário final. Além disso, deverá proporcionar uma navegação clara, com elementos visuais bem organizados e feedback adequado para as ações realizadas pelo usuário.

### **Interfaces de Hardware**

O sistema de hardware do cliente deve contar com uma tela para exibição do sistema, além de periféricos para navegação, entrada de texto e acesso à internet. O hardware do servidor deve ser capaz de fornecer o sistema, por meio de comunicação via internet.

### **Interfaces de Software**

O sistema será desenvolvido com os Frameworks e Bibliotecas: NextJS, React, Tailwind e Shadcn, será alimentado pela API que vai ser desenvolvida com o Framework NestJS que é um framework para Node.js que facilita o desenvolvimento de aplicações escaláveis e eficientes do lado do servidor, utilizando o ORM TypeORM, o banco de dados que será utilizado é o PostgreSQL e a API será hospedada em containeres do Docker.

---

## **Observações Legais, sobre Direitos Autorais e Outras Observações**

Sistema desenvolvido para empresa Londricostura e o código não será aberto.

---

## **Padrões Aplicáveis**

LGPD (Lei Geral de Proteção de Dados - Brasil): Regula o tratamento de dados pessoais de usuários.

---

## **Confidencialidade**

© Londricostura, 2025