

**SERVIÇO NACIONAL DE APRENDIZAGEM COMERCIAL**

**SENAC**

**CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE  
SISTEMAS**

**PROJETO INTEGRADOR IV: DESENVOLVIMENTO DE SISTEMAS  
ORIENTADO A DISPOSITIVOS MÓVEIS E BASEADOS NA WEB**

Integrantes do grupo:

BRUNO PASSARELI COSTA

GUSTAVO LUIZ PEREIRA LIMA

IZABEL RARIDJA SANTOS GUIMARÃES

JOAO PEDRO DE OLIVEIRA GOULART

PRISCILLA FERREIRA SILVA

Integrantes do grupo:

BRUNO PASSARELI COSTA

GUSTAVO LUIZ PEREIRA LIMA

IZABEL RARIDJA SANTOS GUIMARÃES

JOAO PEDRO DE OLIVEIRA GOULART

PRISCILLA FERREIRA SILVA

**PROJETO INTEGRADOR IV: DESENVOLVIMENTO DE SISTEMAS  
ORIENTADO A DISPOSITIVOS MÓVEIS E BASEADOS NA WEB**

Carlos William Ferreira de Lima

**TRABALHO PARA APROVAÇÃO EM DISCIPLINA**

# Resumo

**(obrigatório)**

Aqui você deve fazer um resumo, em um único parágrafo (150 a 500 palavras)

Palavras-chave: <<Web>>, <<Análise e Desenvolvimento de Sistemas>> ,  
<<Dispositivos Móveis>>

## Sumário

1	VISÃO GERAL DA SOLUÇÃO PROPOSTA.....	5
1.1	CONTEXTUALIZAÇÃO E MOTIVAÇÃO.....	5
1.2	OBJETIVOS.....	6
2	PLANEJAMENTO PARA O DESENVOLVIMENTO DA SOLUÇÃO PROPOSTA.....	7
2.1	CICLO DE VIDA DE DESENVOLVIMENTO:.....	7
2.2	PREMISSAS:.....	8
2.3	REQUISITOS (DESCRIÇÃO TEXTUAL E CLASSIFICAÇÃO) / HISTÓRIAS DO USUÁRIO:.....	9
2.3.1	Requisitos Funcionais:.....	9
2.3.2	Requisitos Não Funcionais:.....	10
2.3.3	Histórias do Usuário:.....	11
2.4	PLANEJAMENTO (ETAPAS, TAREFAS, CICLOS DE DESENVOLVIMENTO):.....	11
2.4.1	Levantamento de Requisitos.....	11
2.4.2	Planejamento da Arquitetura.....	12
2.4.3	Desenvolvimento.....	12
2.4.4	Testes e Validação.....	12
2.4.5	Manutenção e Melhorias.....	12
3	DESENVOLVIMENTO DA SOLUÇÃO PROPOSTA.....	14
3.1	PROJETO DO MODELO DE DADOS.....	14
3.2	IMPLEMENTAÇÃO DO WEBSERVICE:.....	15
3.3	IMPLEMENTAÇÃO DO FRONT-END BASEADO EM PADRÃO WEB.....	16
3.4	JUSTIFICATIVA DO USO DE SOLUÇÕES DE COMPUTAÇÃO EM NUVEM.....	17
3.4.1	Utilização de Serviços em Nuvem para Manutenção da Operação....	17
3.4.2	Hospedagem e Escalabilidade.....	17
3.4.3	Armazenamento e Gerenciamento de Dados.....	18
3.4.4	Distribuição de Conteúdo e Performance.....	18
3.5	PERSONAS.....	19
3.6	JORNADAS DE USUÁRIOS.....	19
3.7	SEGUNDA ENTREGA.....	19
3.7.1	Revisitação do Projeto.....	19
3.7.2	Ambiente de Desenvolvimento do Projeto.....	19
	CONCLUSÃO.....	20
	REFERÊNCIAS.....	21

# **1 VISÃO GERAL DA SOLUÇÃO PROPOSTA**

Nesta seção será abordado a contextualização e motivações do projeto, dando-se assim uma visão geral do mercado em que o produto se insere, assim como o problema que a aplicação desenvolvida procura resolver, por fim se estabelece quais os objetivos do projeto.

## **1.1 CONTEXTUALIZAÇÃO E MOTIVAÇÃO**

O mercado de livros no Brasil faturou em 2023 mais de 4 bilhões de reais, que apesar da queda em relação ao ano anterior continua sendo um dos principais acesso das pessoas a cultura, em 2025 cerca de 16% da população com mais de 18 anos consome pelo menos um livro anualmente, e 40% desses compram no mínimo 3 livros por ano. (CÂMARA BRASILEIRA DO LIVRO, 2023) (CÂMARA BRASILEIRA DO LIVRO, 2025).

Com o advento da tecnologia tem a necessidade de poder registrar uma quantidade cada vez maior de informações de maneira mais segura e eficiente, para isso tem se utilizado aplicações que diminuem o risco de falha e o tempo necessário para registrar informações. Outro ponto a se considerar é o compartilhamento de informações relevantes com possíveis clientes, o que é vital hoje com a ampla utilização de tecnologias web e dispositivos móveis.

Tendo em vista este cenário do mercado o objetivo geral do projeto é o desenvolvimento de uma aplicação baseada em web e mobile com o intuito de auxiliar no registro de livros e afins e suas respectivas informações para então se montar um catálogo que pode ser compartilhado com seus potenciais clientes, tendo um foco em livrarias de pequeno e médio porte, aprimorando assim o desempenho dessas empresas.

## 1.2 OBJETIVOS

Entre os objetivos que a aplicação deve alcançar pode-se citar:

- A criação de uma interface de registro de livros e afins de fácil manuseio para uso das livrarias.
- A possibilidade de interagir com os sistemas através da web e dispositivo móveis.
- O compartilhamento de catálogos de livros das lojas com seus possíveis clientes.

## **2 PLANEJAMENTO PARA O DESENVOLVIMENTO DA SOLUÇÃO PROPOSTA**

Nesta seção serão abordados segmentos referentes: ao ciclo de desenvolvimento do projeto; bem como as principais premissas que do projeto; além dos principais requisitos que a aplicação terá que atender bem como a história de usuário para o entendimento das dores do público alvo da aplicação; e por fim será demonstrado o planejamento da aplicação, com as etapas de desenvolvimentos.

### **2.1 CICLO DE VIDA DE DESENVOLVIMENTO:**

Para garantir um desenvolvimento eficiente e adaptado às necessidades da solução proposta, será utilizado o ciclo de vida de desenvolvimento iterativo e incremental. Esse modelo permite um processo dinâmico e flexível, no qual a aplicação será desenvolvida em módulos, possibilitando ajustes ao longo do tempo com base em feedbacks obtidos durante a implementação.

A escolha do ciclo iterativo e incremental se justifica pela necessidade de desenvolver uma aplicação que atenda às demandas específicas de livrarias de pequeno e médio porte, permitindo a evolução contínua da solução sem comprometer a usabilidade ou a estabilidade do sistema.

Esse modelo também possibilita a priorização de funcionalidades essenciais no início do projeto, garantindo entregas parciais que agreguem valor ao usuário final de maneira progressiva.

As principais fases desse ciclo incluem:

- Levantamento de requisitos: Entendimento das necessidades dos usuários e documentação das funcionalidades essenciais da aplicação.
- Planejamento e design: Estruturação do projeto, definição da arquitetura da aplicação e criação dos protótipos de interface.
- Desenvolvimento incremental: Construção da aplicação em etapas, permitindo testes e melhorias contínuas.
- Testes e validação: Avaliação da interface e funcionalidades para garantir usabilidade e eficiência.
- Implantação e monitoramento: Disponibilização da aplicação para os usuários finais e acompanhamento para futuras otimizações.

Esse modelo favorece uma abordagem ágil, permitindo que os ajustes necessários sejam feitos ao longo do processo, garantindo que o produto final atenda às necessidades do público-alvo. Além disso, essa abordagem minimiza riscos e reduz o tempo de desenvolvimento, já que funcionalidades podem ser entregues gradualmente, assegurando uma adaptação mais rápida às demandas do mercado.

## 2.2 PREMISSAS:

Para garantir que a aplicação atenda às expectativas dos usuários e proporcione uma experiência intuitiva e eficiente, algumas premissas fundamentais foram estabelecidas:

- Facilidade de uso: A interface da aplicação deverá ser intuitiva, permitindo que os usuários cadastrem e gerenciem seus catálogos de livros sem dificuldades.
- Acessibilidade e multiplataforma: A solução deverá ser compatível com dispositivos móveis e navegadores web, garantindo acesso flexível.



- Segurança e confiabilidade: O armazenamento de informações deverá ser seguro, protegendo dados sensíveis e garantindo a integridade das informações.
- Interatividade e compartilhamento: A aplicação deverá possibilitar o compartilhamento de catálogos de forma prática e eficiente.
- Metodologia de validação: Para assegurar a qualidade da experiência do usuário, serão realizados testes de usabilidade com um grupo piloto de livrarias, coletando feedbacks para melhorias antes do lançamento oficial.

Com base nessas premissas, o desenvolvimento será direcionado para proporcionar uma solução eficiente, que atenda às necessidades das livrarias de pequeno e médio porte, otimizando seus processos de registro e compartilhamento de informações sobre livros.

## 2.3 REQUISITOS (DESCRIÇÃO TEXTUAL E CLASSIFICAÇÃO) / HISTÓRIAS DO USUÁRIO:

Para garantir que a aplicação atenda às necessidades dos usuários e cumpra seus objetivos, foram definidos requisitos funcionais e não funcionais, além de histórias do usuário para ilustrar a experiência esperada.

### 2.3.1 Requisitos Funcionais:

Os requisitos funcionais descrevem as funcionalidades que a aplicação deverá oferecer.

- O sistema deve permitir o cadastro, edição e exclusão de livros no catálogo.
- O sistema deve permitir o cadastro de informações detalhadas sobre cada livro, incluindo título, autor, editora, preço e disponibilidade.

- O usuário deve poder buscar livros por título, autor, categoria ou palavra-chave.
- O sistema deve permitir a geração e o compartilhamento de um link do catálogo.
- A aplicação deve possibilitar o gerenciamento de diferentes categorias de livros.
- O sistema deve oferecer uma interface responsiva para uso em dispositivos móveis e desktop.
- O usuário deve poder visualizar estatísticas básicas sobre os livros cadastrados (ex.: quantidade de livros disponíveis por categoria).

### 2.3.2 Requisitos Não Funcionais:

Os requisitos não funcionais definem características técnicas e de desempenho da aplicação.

- A aplicação deve garantir tempo de resposta inferior a **2 segundos** para consultas no catálogo.
- O sistema deve seguir princípios de **usabilidade**, garantindo uma interface intuitiva e acessível.
- O banco de dados deve garantir a **persistência segura** dos dados, evitando perdas e inconsistências.
- O sistema deve permitir **acesso simultâneo** de múltiplos usuários sem degradação de desempenho.
- A arquitetura da aplicação deve ser **escalável**, possibilitando a adição de novas funcionalidades futuramente.

### 2.3.3 Histórias do Usuário:

As histórias do usuário ilustram como diferentes tipos de usuários interagem com a aplicação.

#### 1. Cadastro de Livro:

- o "Como **funcionário de livraria**, quero cadastrar um novo livro no sistema informando título, autor e preço, para que ele apareça no catálogo da loja."

#### 2. Busca por Livros:

- o "Como **cliente**, quero pesquisar livros pelo título ou gênero, para encontrar mais rapidamente o que desejo comprar."

#### 3. Gerenciamento do Catálogo:

- o "Como **dono da livraria**, quero editar e remover livros do catálogo quando necessário, para manter a lista sempre atualizada."

#### 4. Compartilhamento do Catálogo:

- o "Como **livreiro**, quero gerar um link compartilhável do meu catálogo, para divulgar os livros para meus clientes."

### 2.4 PLANEJAMENTO (ETAPAS, TAREFAS, CICLOS DE DESENVOLVIMENTO):

O planejamento do desenvolvimento do projeto será dividido em etapas, com tarefas específicas e ciclos de desenvolvimento bem definidos.

#### 2.4.1 Levantamento de Requisitos

##### Tarefas:

- Reuniões com stakeholders para entender necessidades

- Documentação dos requisitos funcionais e não funcionais
- Criação de protótipos iniciais da interface

## **2.4.2 Planejamento da Arquitetura**

### **Tarefas:**

- Definição da arquitetura da aplicação (frontend e backend).
- Escolha das tecnologias a serem utilizadas (linguagens, frameworks, banco de dados).
- Planejamento da estrutura do banco de dados

## **2.4.3 Desenvolvimento**

- Utilização de metodologias ágeis
- Divisão do desenvolvimento em sprints de 2-4 semanas
- Implementação das funcionalidades do sistema (cadastro, busca, gerenciamento de categorias)

## **2.4.4 Testes e Validação**

- realização de testes funcionais e não funcionais
- Testes de usabilidade com usuários reais
- Correção de bugs e ajustes baseados no feedback

## **2.4.5 Manutenção e Melhorias**

- Monitoramento do desempenho da aplicação
- Coleta de feedback dos usuários para melhorias
- Planejamento de novas funcionalidades e correções de problemas identificados

O planejamento é essencial para garantir que todas as etapas do projeto sejam executadas de forma organizada e eficiente. A utilização de ciclos de desenvolvimento ágeis permitirá adaptações rápidas às necessidades dos usuários.

Link do Github: <https://github.com/BunnyCute99/PTI-Web>

### **3 DESENVOLVIMENTO DA SOLUÇÃO PROPOSTA**

A solução será construída com uma arquitetura eficiente, escalável e de fácil uso, permitindo que os usuários realizem o cadastro, organização e compartilhamento do catálogo de forma intuitiva.

Para isso, serão utilizadas tecnologias modernas no desenvolvimento do back-end e front-end, garantindo uma experiência fluida e responsiva. Além disso, a aplicação contará com a integração a um banco de dados seguro e estruturado, permitindo o armazenamento e a recuperação eficiente das informações.

#### **3.1 PROJETO DO MODELO DE DADOS**

A modelagem de dados será fundamental para garantir a eficiência no armazenamento, recuperação e manipulação das informações, permitindo que os usuários registrem seus produtos de forma detalhada e organizada.

Para isso, será necessário definir entidades que representam os principais elementos do sistema, como livros, autores, editoras e categorias, além de informações relacionadas aos clientes e suas interações com o catálogo. Cada livro terá atributos que descrevem suas características, incluindo:

- Título
- Autor
- Editora
- Ano de Publicação
- Gênero
- Número de páginas

- Idioma
- Preço

Também será essencial considerar a possibilidade de múltiplos autores para uma única obra, o que exigirá a criação de uma relação adequada entre as tabelas.

Além disso, será implementado um sistema de usuários, possibilitando que diferentes perfis tenham acesso à plataforma com permissões específicas, como:

- Administradores
- Funcionários
- Clientes

Com um modelo de dados bem estruturado, a aplicação proporcionará um ambiente eficiente para o gerenciamento de livros, contribuindo para a otimização das operações e a melhoria do desempenho das livrarias.

### 3.2 IMPLEMENTAÇÃO DO WEBSERVICE:

A implementação do Web Service será fundamental para permitir a comunicação entre o front-end e o back-end da aplicação, garantindo a integração dos dados de forma segura e eficiente. O serviço será desenvolvido utilizando uma API RESTful, que permitirá operações como cadastro, consulta, atualização e remoção de livros e usuários.

O back-end será responsável por processar as requisições e interagir com o banco de dados, garantindo a consistência das informações. As principais rotas da API incluirão:

- /livros – Gerenciamento de livros (listagem, criação, edição e exclusão)

- /autores – Cadastro e consulta de autores
- /editoras – Registro e gerenciamento de editoras
- /usuarios – Controle de usuários e autenticação
- /catalogo – Consulta de livros disponíveis para os clientes

A API será estruturada para garantir alta performance e escalabilidade, utilizando um modelo que assegure a segurança nas transações e a integridade dos dados. A comunicação será feita de forma eficiente e segura, utilizando práticas de autenticação e autorização, como tokens para garantir acesso controlado e seguro.

Além disso, serão implementadas medidas de segurança, como proteção contra ataques de injeção SQL, controle de acesso entre diferentes origens (CORS), e estratégias de autenticação adequadas para proteger os dados e as interações com o sistema.

Com essa implementação, a aplicação poderá fornecer um ambiente robusto e confiável para o gerenciamento do catálogo de livros, facilitando a interação entre livrarias e clientes.

### 3.3 IMPLEMENTAÇÃO DO FRONT-END BASEADO EM PADRÃO WEB

Será baseado nas melhores práticas da web moderna, com foco em criar uma interface intuitiva e responsiva, capaz de se adaptar a diferentes dispositivos e tamanhos de tela, como desktops, tablets e smartphones. A aplicação será estruturada utilizando componentes reutilizáveis, o que facilita a manutenção e garante uma experiência de usuário consistente. A responsividade será um dos principais pilares, utilizando técnicas como consultas de mídia e layouts fluidos, para que o conteúdo se ajuste de forma otimizada em qualquer dispositivo.



Além disso, a navegação será simplificada, com menus adaptáveis e elementos interativos otimizados para dispositivos móveis. A interatividade será aprimorada com animações e transições suaves, oferecendo uma experiência agradável ao usuário. A usabilidade será uma prioridade, com a interface sendo organizada de forma lógica e intuitiva para facilitar a navegação, enquanto a acessibilidade será garantida por meio de práticas como navegação por teclado e contraste adequado de cores.

Por fim, a performance será otimizada com técnicas de compressão de recursos, como imagens e scripts, além da implementação de recursos como lazy loading, garantindo que a aplicação carregue rapidamente e proporcione uma experiência eficiente e ágil para todos os usuários.

### 3.4 JUSTIFICATIVA DO USO DE SOLUÇÕES DE COMPUTAÇÃO EM NUVEM

#### **3.4.1 Utilização de Serviços em Nuvem para Manutenção da Operação**

A solução proposta pode se beneficiar significativamente do uso de serviços em nuvem, garantindo alta escalabilidade, flexibilidade e disponibilidade contínua para manter a aplicação em operação de forma eficiente. A seguir, são detalhadas as principais formas de utilizar a nuvem para suportar a infraestrutura da aplicação.

#### **3.4.2 Hospedagem e Escalabilidade**

A hospedagem da aplicação será realizada em plataformas de nuvem, como AWS, Google Cloud ou Microsoft Azure. Estas oferecem recursos como máquinas virtuais e containers para executar a aplicação de forma escalável. A principal vantagem é a capacidade de escalar automaticamente os recursos computacionais conforme a demanda de

tráfego, garantindo que a aplicação continue a funcionar de maneira eficiente mesmo durante picos de acesso.

A escalabilidade na nuvem permite ajustar o desempenho de acordo com a quantidade de usuários e o volume de dados processados, sem a necessidade de intervenções manuais, o que reduz custos e melhora a eficiência operacional.

### **3.4.3 Armazenamento e Gerenciamento de Dados**

O banco de dados será hospedado na nuvem, utilizando serviços gerenciados como Amazon RDS ou Google Cloud SQL. Estes serviços oferecem a vantagem de backup automático, alta disponibilidade e escalabilidade, garantindo que a aplicação possa armazenar, acessar e processar grandes volumes de dados de forma confiável.

Além disso, a nuvem permite realizar replicação de dados em diferentes regiões geográficas, o que aumenta a segurança e a confiabilidade, minimizando riscos de perda de dados em caso de falhas.

### **3.4.4 Distribuição de Conteúdo e Performance**

Para melhorar o desempenho da aplicação, será utilizado uma CDN (Content Delivery Network), como Amazon CloudFront ou Google Cloud CDN. Essas redes de distribuição de conteúdo ajudam a reduzir o tempo de carregamento da aplicação, distribuindo o conteúdo estático, como imagens e arquivos JavaScript, em servidores localizados em diferentes regiões geográficas.

Essa abordagem assegura que os usuários, independentemente de sua localização, tenham uma experiência de navegação rápida e fluida, o que é especialmente importante para dispositivos móveis e conexões de internet mais lentas.

### 3.5 PERSONAS

#### 2 PERSONAS DA APLICAÇÃO

### 3.6 JORNADAS DE USUÁRIOS

#### 1 JORNADAS DE USUÁRIOS

### 3.7 SEGUNDA ENTREGA

Nesta secção será abordado a revisita do projeto com suas modificações e prova de conceito da aplicação, assim como a descrição do ambiente de desenvolvimento da aplicação, detalhando as tecnologias que foram utilizadas na construção da aplicação, tanto no backend quanto no frontend e banco de dados.

#### **3.7.1 Revisitação do Projeto**

##### REVISÕES DO PROJETO NA SEGUNDA ENTREGA

#### **3.7.2 Ambiente de Desenvolvimento do Projeto**

No desenvolvimento do projeto uma variedade de tecnologias foram utilizadas, do banco de dados, passando pelo backend até o frontend. Esta secção busca elencar estas diversas tecnologias dando uma dimensão da importância das mesmas para a realização do projeto.

Para o armazenamento de dados relevantes da aplicação foi escolhido o PostgreSQL, um banco de dados que possibilita relacionar entidades umas com as outras e organizar dados em tabelas para a leitura humana.

Já no frontend foi utilizado o *HyperText Markup Language* (HTML) para a construção das páginas web, com a utilização do *Cascading Style Sheets* (CSS) para estilizar as páginas e deixá-las intuitivas para os

usuários finais, também foi utilizado o Thymeleaf para mostrar dados do backend para no frontend.

No Backend foi utilizado o framework Spring Boot em conjunto com java para realizar as operações no lógicas da aplicação, como criar, editar, listar e excluir livros e estantes do banco de dados, além de utilizar essas ferramentas para elaborar o programa na arquitetura *Model-View-Controller* (MVC) com adição de camadas *Service*, segmentando assim as funções da aplicação.

Também foram utilizados o site Github para fazer o versionamento do projeto assim como seu compartilhamento, disponível no seguinte link (<https://github.com/BunnyCute99/PTI-Web>). Além da utilização do Visual Studio Code como ambiente de desenvolvimento integrado.

## **CONCLUSÃO**

(resultados e conclusões sobre o projeto)

## REFERÊNCIAS

CÂMARA BRASILEIRA DO LIVRO. **PANORAMA DO CONSUMO DE LIVROS.** 2025. Disponível em: <[https://cbl.org.br/wp-content/uploads/2025/02/2024\\_Panorama-do-Consumo-de-Livros\\_Imprensa\\_V2.pdf](https://cbl.org.br/wp-content/uploads/2025/02/2024_Panorama-do-Consumo-de-Livros_Imprensa_V2.pdf)> Acessado em: 22 de Fevereiro de 2025.

CÂMARA BRASILEIRA DO LIVRO. **DESEMPENHO REAL DO MERCADO EDITORIAL BRASILEIRO.** 2024. Disponível em: <[https://cbl.org.br/wp-content/uploads/2024/07/Serie\\_Historica\\_PCR\\_2023v2.pdf](https://cbl.org.br/wp-content/uploads/2024/07/Serie_Historica_PCR_2023v2.pdf)> Acessado em: 22 de Fevereiro de 2025.