# CENTRO PAULA SOUZA

**ETEC PROFESSOR CAMARGO ARANHA TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS**

GUILHERME HENRIQUE DO NASCIMENTOS GOMES

GUSTAVO

MELISSA

RICHARD

ENZO

# Estudo de SITE PARA AGENDAMENTO INTELIGENTE EM SERVIÇO DE BARBEARIA INTEGRADO COM O WHARSAPP

SÃO PAULO 2025

Guilherme Gomes

## Site integrado com inteligencia artificial (chatbox) para WhatsApp

Trabalho de conclusão de curso apresentado à Escola Técnica Estadual Professor Camargo Aranha, como requisito obrigatório para a obtenção do diploma de Técnico em Desenvolvimento de Sistemas, sob a orientação de Roberto Itai

# SÃO PAULO

**2025**

### **RESUMO**

O presente Trabalho de Conclusão de Curso tem como objetivo propor o desenvolvimento de um **sistema web integrado a uma chatbox inteligente** voltado para **barbearias de pequeno e médio porte**, com foco na **automação e otimização do processo de agendamento de serviços**. O projeto busca solucionar problemas recorrentes de **falta de organização e dificuldade de comunicação** entre barbeiros e clientes, oferecendo uma ferramenta moderna, prática e acessível.

O sistema proposto é composto por um **site responsivo**, desenvolvido com **HTML, CSS e JavaScript**, contendo as abas **Home, Serviços, Agendamento, Sobre e Contato**, e por uma **chatbox integrada ao WhatsApp**, que permite ao cliente realizar agendamentos de forma rápida, por texto ou áudio, com envio automático das informações para o **banco de dados**.

A arquitetura do sistema segue os princípios do **desenvolvimento web Full Stack**, utilizando **Node.js** no backend e **MySQL** como Sistema Gerenciador de Banco de Dados Relacional (SGBDR). A metodologia adotada foi a **Scrum**, por promover entregas contínuas e colaborativas durante o processo de design, desenvolvimento e testes.

O design da interface foi elaborado com base em práticas de **UI/UX**, priorizando usabilidade, clareza e experiência do usuário. Além disso, o projeto apresenta diagramas de caso de uso (UML), modelo entidade-relacionamento (ER) e protótipos funcionais que representam o funcionamento completo do sistema.

Conclui-se que o desenvolvimento do sistema proposto representa um **avanço tecnológico no gerenciamento de agendamentos em barbearias**, promovendo **eficiência, organização e praticidade** tanto para os profissionais quanto para os clientes.

**Palavras-chave:** Chatbot. Barbearia. Agendamento Online. Desenvolvimento Web. Automação. Sistema de Gestão.

### **ABSTRACT**

This Final Paper aims to propose the development of an **intelligent web system integrated with a WhatsApp chatbot**, focused on **small and medium barbershops**, with the goal of **automating and optimizing service scheduling**. The project addresses common issues such as **lack of organization and communication difficulties** between barbers and clients, providing a modern, practical, and accessible solution.

The proposed system consists of a **responsive website** developed with **HTML, CSS, and JavaScript**, containing the tabs **Home, Services, Scheduling, About, and Contact**, and a **WhatsApp chatbot** that allows clients to schedule appointments quickly via text or audio, with automatic submission of the information to a **MySQL database**.

The system architecture follows **Full Stack web development principles**, and the project adopts the **Scrum methodology** to ensure iterative and collaborative deliveries during design, development, and testing phases.

The user interface design emphasizes **usability, clarity, and user experience (UI/UX best practices)**. Additionally, the study presents use case diagrams (UML), entity-relationship models (ER), and functional prototypes, providing a complete representation of the system’s operation.

It is concluded that the proposed system represents a **technological advance in scheduling management for barbershops**, promoting **efficiency, organization, and convenience** for both professionals and clients.

**Keywords:** Chatbot. Barbershop. Online Scheduling. Web Development. Automation. Management System.

# SUMARIO FIGURA

[Figura 1 - Diagrama de caso 12](#_bookmark20)

[Figura 2 - Diagrama de entidade 13](#_bookmark22)

[Figura 3 -Tela inicial 14](#_bookmark24)

[Figura 4 - Tela de cadastro 15](#_bookmark25)

[Figura 5 - Tela de envio de foto 15](#_bookmark26)

[Figura 6 - Tela com a foto enviada 16](#_bookmark27)

### **LISTA DE SIGLAS**

**API** – Application Programming Interface / Interface de Programação de Aplicações – Integração entre sistemas, como WhatsApp e banco de dados  
**DCU** – Diagrama de Caso de Uso – Engenharia de Software / UML  
**ER** – Diagrama Entidade-Relacionamento – Banco de Dados / Modelagem  
**MVP** – Minimum Viable Product / Produto Mínimo Viável – Desenvolvimento de software  
**RF** – Requisito Funcional – Engenharia de Software  
**RNF** – Requisito Não Funcional – Engenharia de Software  
**SGBDR** – Sistema de Gerenciamento de Banco de Dados Relacional – Banco de Dados  
**SQL** – Structured Query Language / Linguagem de Consulta Estruturada – Banco de Dados  
**UML** – Unified Modeling Language / Linguagem de Modelagem Unificada – Engenharia de Software  
**UX** – User Experience / Experiência do Usuário – Design de Software  
**UI** – User Interface / Interface do Usuário – Design de Software

**SUMÁRIO**

1. [INTRODUÇÃO 6](#_bookmark0)
2. [FUNDAMENTAÇÃO TEÓRICA 7](#_bookmark1)
   1. [Inteligência Artificial e Reconhecimento de Imagens 7](#_bookmark2)
   2. [APIs Nutricionais 7](#_bookmark3)
   3. [Desenvolvimento Web Moderno 7](#_bookmark4)
   4. [Design de Interfaces com Figma (UI/UX) 8](#_bookmark5)
   5. [Frontend: React.js 8](#_bookmark6)
   6. [Backend: Node.js e Express.js 8](#_bookmark7)
   7. [Banco de Dados: MySQL e Modelagem de Dados 9](#_bookmark8)
   8. [Modelagem de Sistemas: UML e ER 9](#_bookmark9)
   9. [Metodologia Ágil: Scrum 9](#_bookmark10)
3. [OBJETIVOS 10](#_bookmark11)
   1. [Objetivo Geral 10](#_bookmark12)
   2. [Objetivos Específicos 10](#_bookmark13)
4. [DESCRIÇÃO DO PROBLEMA 10](#_bookmark14)
5. [SOLUÇÃO PROPOSTA 11](#_bookmark15)
6. [REQUISITOS FUNCIONAIS 11](#_bookmark16)
7. [DIAGRAMAS 11](#_bookmark18)
   1. [Diagrama de Caso de Uso (UML) 12](#_bookmark19)
   2. [Diagrama Entidade-Relacionamento (ER) 13](#_bookmark21)
   3. [Wireframes 14](#_bookmark23)
8. [DICIONÁRIO DE DADOS 17](#_bookmark29)
9. [METODOLOGIA 18](#_bookmark31)
   1. [Estrutura Scrum 18](#_bookmark32)
   2. [Fases do Projeto 18](#_bookmark33)
10. [CRONOGRAMAS 19](#_bookmark34)
    1. [Cronograma da Parte Teórica (setembro a novembro de 2025) 19](#_bookmark35)
    2. [Cronograma de Desenvolvimento (fevereiro a junho de 2026) 20](#_bookmark37)
11. [CONSIDERAÇÕES FINAIS 20](#_bookmark39)
12. [REFERÊNCIAS 21](#_bookmark40)

### **1. INTRODUÇÃO**

A organização eficiente do fluxo de clientes e a gestão de agendamentos tornaram-se prioridades para barbearias de pequeno e médio porte, especialmente diante do crescimento do setor e da demanda por serviços personalizados. Muitas dessas barbearias ainda enfrentam dificuldades significativas na marcação de horários, o que gera **desperdício de tempo, desorganização e insatisfação tanto para os clientes quanto para os barbeiros**. Métodos tradicionais, como registros manuais ou agenda física, são suscetíveis a erros, esquecimentos e limitações de comunicação, comprometendo a eficiência do serviço e a experiência do cliente.

Neste contexto, o desenvolvimento de **sistemas digitais integrados**, como sites responsivos e chatbots, surge como uma solução promissora para otimizar o processo de agendamento e organizar as informações de maneira automatizada. Inspirado por essa necessidade, este Trabalho de Conclusão de Curso propõe o desenvolvimento do sistema **BarberShop**, composto por um **site com abas Home, Serviços, Agendamento, Sobre e Contato**, e uma **chatbox integrada ao WhatsApp**, capaz de receber solicitações de agendamento por texto ou áudio, com envio automático dos dados para um **banco de dados MySQL**.

O sistema será desenvolvido utilizando **HTML, CSS e JavaScript**, garantindo compatibilidade, escalabilidade e responsividade. O design da interface será planejado com foco em **usabilidade e experiência do usuário (UI/UX)**, priorizando clareza, acessibilidade e praticidade no agendamento de horários.

A metodologia ágil **Scrum** será adotada para guiar o desenvolvimento, promovendo entregas iterativas, colaboração entre os membros da equipe e adaptação contínua às necessidades dos usuários. Este estudo de caso não apenas apresenta uma **solução tecnológica inovadora para barbearias**, mas também fornece uma documentação completa, incluindo diagramas de caso de uso (UML), modelo entidade-relacionamento (ER) e protótipos funcionais, consolidando o projeto como um guia técnico-metodológico para sistemas de agendamento digital.

### **1. FUNDAMENTAÇÃO TEÓRICA**

#### **1.1 Sistemas Web e Automação de Agendamento**

Sistemas web desempenham um papel fundamental na automação de processos em empresas de pequeno e médio porte, permitindo a **organização eficiente de informações, interação com clientes e otimização de operações internas**. Em especial, para barbearias, a gestão de agendamentos é um ponto crítico, pois falhas nesse processo podem gerar atrasos, conflitos de horários e insatisfação dos clientes.

O desenvolvimento de **sites responsivos**, utilizando **HTML, CSS e JavaScript**, aliado à integração com **bancos de dados relacionais como MySQL**, possibilita o registro automático de informações e a manutenção de dados atualizados em tempo real. Ferramentas de **API**, como a integração com o WhatsApp, permitem o recebimento de solicitações de agendamento por texto ou áudio, transmitindo esses dados diretamente para o sistema central, promovendo **rapidez e precisão na gestão de horários**.

Conforme destacado em estudos de engenharia de software e desenvolvimento web, a utilização de **interfaces intuitivas, práticas de UX/UI e automação via APIs** contribui significativamente para a melhoria da experiência do usuário, a redução de erros manuais e o aumento da eficiência operacional. A integração de sistemas web com **chatbots inteligentes** tem se mostrado eficaz na comunicação automatizada com clientes, garantindo que informações importantes, como disponibilidade de horários, tipos de serviços e confirmação de agendamentos, sejam transmitidas de forma clara e imediata.

Explorar essas tecnologias, como desenvolvimento web responsivo, bancos de dados relacionais e APIs de comunicação, é essencial para criar soluções práticas e escaláveis que atendam às necessidades de barbearias independentes e promovam **organização, praticidade e satisfação do cliente**.

#### **1.1 Desenvolvimento Web Moderno**

O projeto BarberShop será construído sobre uma **arquitetura web moderna e escalável**, utilizando tecnologias amplamente aplicadas no desenvolvimento front-end, como **HTML, CSS e JavaScript**, e integrando um **banco de dados MySQL** para armazenamento e gerenciamento das informações de agendamentos.

A escolha dessas tecnologias permite a criação de um **site responsivo e intuitivo**, no qual a interface do usuário é clara e acessível, enquanto o backend garante o registro seguro e eficiente dos dados enviados pelo site ou pela chatbox do WhatsApp. Essa abordagem facilita o desenvolvimento e a manutenção do sistema, além de **otimizar a experiência do usuário** e a **colaboração entre os membros da equipe**, que podem trabalhar de forma organizada nas diferentes camadas do projeto (NextAge, 2025).

#### **1.2 Design de Interfaces com Figma (UI/UX)**

O Design de Interface do Usuário (UI) e a Experiência do Usuário (UX) são elementos fundamentais para o sucesso de qualquer aplicação web. No projeto BarberShop, o **design inicial do site foi desenvolvido pelo próprio autor utilizando o Figma**, ferramenta amplamente adotada na indústria para prototipagem e design colaborativo (UX Tools, 2021; Cubos Academy, 2023).

O Figma permite a criação de **wireframes** (esboços de baixa fidelidade), **protótipos interativos** e **design systems** (bibliotecas de componentes reutilizáveis). Sua capacidade de colaboração em tempo real garante o alinhamento entre a concepção do design e a implementação do sistema. A elaboração dos protótipos no Figma assegura que a interface final seja **intuitiva, acessível e alinhada com as melhores práticas de UI/UX**, proporcionando uma experiência de agendamento simples e eficiente tanto para os clientes quanto para os barbeiros.

#### **1.3 Frontend: HTML, CSS e JavaScript**

O frontend do projeto BarberShop foi desenvolvido utilizando **HTML, CSS e JavaScript**, tecnologias tradicionais e amplamente utilizadas para a construção de interfaces web responsivas e interativas. Essa abordagem permite criar **layouts claros e intuitivos**, garantindo que o sistema se adapte a diferentes dispositivos e tamanhos de tela, sem comprometer a experiência do usuário.

O uso combinado dessas tecnologias assegura a **reutilização de componentes de interface**, facilita a manutenção do código e mantém padrões de **acessibilidade e usabilidade** no site. Dessa forma, os clientes podem navegar pelas abas **Home, Serviços, Agendamento, Sobre e Contato** de forma prática e eficiente, enquanto a interface permanece simples e funcional para os barbeiros gerenciarem os agendamento

#### **1.4 Banco de Dados: MySQL e Modelagem de Dados**

O MySQL foi escolhido como o **Sistema de Gerenciamento de Banco de Dados Relacional (SGBDR)** para o projeto BarberShop, devido à sua maturidade, confiabilidade e robustez, garantindo **integridade e segurança dos dados dos clientes e agendamentos** (LuizTools, 2021).

A estruturação do banco de dados será guiada pelo **Modelo Entidade-Relacionamento (ER)**, uma técnica de modelagem de dados conceituais amplamente utilizada na Engenharia de Software (Franck, 2021). O banco armazenará informações sobre **clientes, serviços, horários e confirmações de agendamento**, tanto provenientes do site quanto da chatbox integrada ao WhatsApp, permitindo **consultas rápidas, controle de disponibilidade e histórico de atendimentos**.

#### **1.5 Modelagem de Sistemas: UML e ER**

A modelagem é uma atividade essencial na Engenharia de Software, pois permite a **representação e compreensão de sistemas complexos**, como o projeto BarberShop.

A **Linguagem de Modelagem Unificada (UML – Unified Modeling Language)** é utilizada como notação padrão para **visualização, especificação, construção e documentação** dos artefatos do sistema (Fowler, 2003). Para o projeto, o **Diagrama de Caso de Uso (DCU)** será empregado para mapear os **requisitos funcionais** e as interações do usuário com o sistema, como a realização de agendamentos via site ou WhatsApp, visualização de horários disponíveis e confirmação de serviços.

O **Diagrama Entidade-Relacionamento (ER)** modela a **estrutura do banco de dados**, servindo de base para a implementação da camada de persistência. Ele define **entidades** (tabelas), **atributos** (campos) e os **relacionamentos** (cardinalidade) entre elas, permitindo organizar informações sobre clientes, serviços, horários e confirmações de agendamento de forma eficiente e segura (Franck, 2021).

#### **1.6 Metodologia Ágil: Scrum**

A metodologia ágil **Scrum** é o framework mais utilizado no desenvolvimento de software, especialmente em projetos com **requisitos dinâmicos** e que demandam **entregas contínuas de valor**.

No projeto BarberShop, o Scrum será aplicado para organizar e gerenciar o desenvolvimento do **site de agendamentos e da chatbox integrada ao WhatsApp**, promovendo **colaboração entre os membros da equipe**, adaptação rápida a mudanças nos requisitos e **entregas iterativas e incrementais** das funcionalidades (Schwaber & Sutherland, 2018; PUC Minas, 2024).

# A utilização do Scrum permite que a equipe **priorize tarefas, identifique obstáculos rapidamente e realize ajustes contínuos**, garantindo que o sistema final atenda às necessidades dos clientes e dos barbeiros de forma eficiente, prática e organizada. OBJETIVOS

### **1.1 Objetivo Geral**

Desenvolver um **estudo de caso** sobre o sistema BarberShop, focado na **automatização do agendamento de serviços em barbearias pequenas e médias**, abrangendo análise, design, modelagem de dados e documentação, por meio de um **site responsivo** e de uma **chatbox integrada ao WhatsApp**, com envio automático das informações para o banco de dados MySQL.

### **1.2 Objetivos Específicos**

* Realizar o **levantamento de requisitos** do sistema, identificando necessidades dos barbeiros e clientes.
* Elaborar o **embasamento teórico** sobre desenvolvimento web, design de interfaces, bancos de dados e integração com APIs.
* Desenvolver o **frontend do site** utilizando HTML, CSS e JavaScript.
* Criar o **design de interfaces** no Figma, priorizando usabilidade e experiência do usuário (UI/UX).
* Definir e documentar os **Requisitos Funcionais (RFs)** e **Requisitos Não Funcionais (RNFs)**.
* Elaborar o **Diagrama Entidade-Relacionamento (ER)** para estruturar o banco de dados.
* Criar o **Diagrama de Caso de Uso (UML)**, mapeando interações de usuários com o sistema.
* Desenvolver **protótipos funcionais** das interfaces do site e da chatbox.
* Elaborar o **Dicionário de Dados**, detalhando todas as informações armazenadas no banco.

### **2. DESCRIÇÃO DO PROBLEMA**

Barbearias de pequeno e médio porte frequentemente enfrentam problemas de **desorganização e dificuldade na gestão de agendamentos**. O uso de métodos tradicionais, como registros manuais ou agendas físicas, torna o processo de marcação de horários **lento, sujeito a erros e falhas de comunicação**, gerando **conflitos de horários, atrasos e insatisfação dos clientes**.

Além disso, a ausência de sistemas digitais integrados dificulta que os clientes realizem agendamentos de forma **rápida e prática**, principalmente fora do horário de funcionamento ou em situações onde o contato telefônico não é eficiente. Essa deficiência impacta diretamente na **eficiência operacional da barbearia** e na **experiência do cliente**, comprometendo a competitividade do negócio.

# SOLUÇÃO PROPOSTA

A solução consiste em um Sistema Web Inteligente que automatiza a identificação de alimentos e o cálculo calórico a partir do *upload* de uma foto. Ao integrar a capacidade de reconhecimento visual da IA (ex: Gemini API) com a base de dados confiável das APIs nutricionais, o sistema elimina a necessidade de entrada manual, oferecendo praticidade e precisão para o controle dietético.

# REQUISITOS FUNCIONAIS

***Como documentar Regra de Negócio***

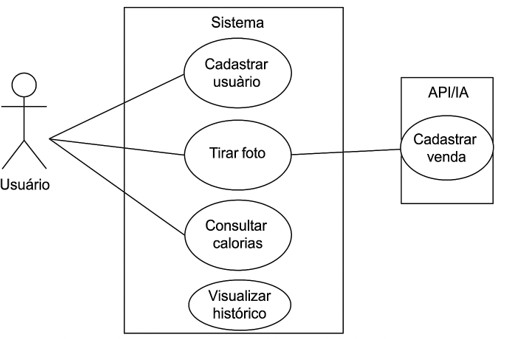
# DIAGRAMAS

Diagramas são representações gráficas utilizadas na engenharia de software para modelar, visualizar e documentar os componentes de um sistema. Eles facilitam a comunicação entre os envolvidos no projeto, como analistas, desenvolvedores e clientes, além de servirem como guia técnico para implementação.

“Diagramas são ferramentas fundamentais para a análise e projeto de sistemas, pois permitem abstrair e organizar informações complexas” (SOMMERVILLE, 2011).

## Diagrama de Caso de Uso (UML)

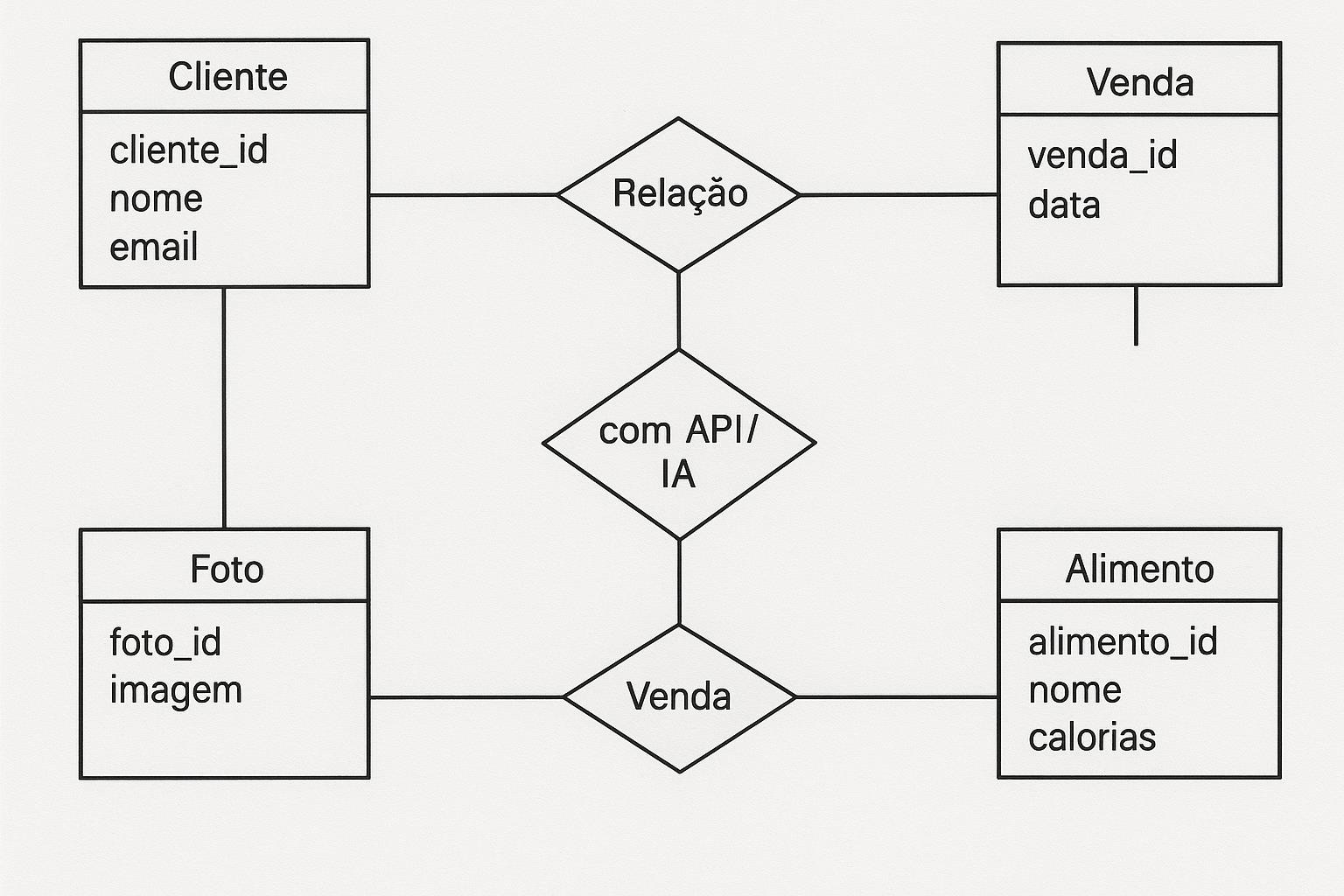
Representa as interações entre os usuários (atores) e o sistema, descrevendo as funcionalidades esperadas. Permite identificar os requisitos funcionais e entender o comportamento do sistema sob a perspectiva do usuário.



*Figura 1 - Diagrama de caso*

## Diagrama Entidade-Relacionamento (ER)

Modela a estrutura do banco de dados, mostrando as entidades (tabelas), seus atributos (campos) e os relacionamentos entre elas. É essencial para garantir a integridade dos dados e orientar o desenvolvimento da camada de persistência.



*Figura 2 - Diagrama de entidade*

## 5. DICIONÁRIO DE DADOS

O Dicionário de Dados é um artefato crítico para a documentação e o gerenciamento de dados em um projeto de software. Ele padroniza e detalha a estrutura do banco de dados, definindo cada tabela, campo, tipo, restrições e finalidade, garantindo a consistência e clareza para a equipe de desenvolvimento (W3C, 2017; USP, 2021).

“O dicionário de dados é uma ferramenta que documenta os elementos de informação de um sistema, promovendo consistência e controle sobre os dados utilizados” (Sommerville, 2011, p. 250).

|  |  |  |  |
| --- | --- | --- | --- |
| **Tabela** | **Campo** | **Tipo** | **Descrição** |
| **clientes** | cliente\_id | INT (PK) | Identificador único do cliente |
| **clientes** | nome | VARCHAR | Nome completo do cliente |
| **clientes** | email | VARCHAR | Email do cliente |
| **refeicoes** | refeicao\_id | INT (PK) | Identificador da refeição |
| **refeicoes** | cliente\_id | INT (FK) | Referência ao cliente |
| **refeicoes** | imagem\_url | TEXT | URL da imagem enviada |
| **alimentos** | alimento\_id | INT (PK) | Identificador do alimento |
| **alimentos** | nome | VARCHAR | Nome do alimento |
| **alimentos** | calorias | FLOAT | Valor calórico por porção |
| **alimentos** | proteinas | FLOAT | Proteínas por porção |
| **alimentos** | carboidratos | FLOAT | Carboidratos por porção |
| **alimentos** | gorduras | FLOAT | Gorduras por porção |
| **refeicao\_alimento** | id | INT (PK) | Identificador da relação |
| **refeicao\_alimento** | refeicao\_id | INT (FK) | Referência à refeição |
| **refeicao\_alimento** | alimento\_id | INT (FK) | Referência ao alimento |
| **refeicao\_alimento** | quantidade | FLOAT | Quantidade em gramas |
| **vendas** | venda\_id | INT (PK) | Identificador da venda |
| **vendas** | cliente\_id | INT (FK) | Referência ao cliente |
| **vendas** | plano | VARCHAR | Tipo de plano adquirido |

*Tabela 2 - Dicionário de dados*

## 5. METODOLOGIA

Para o desenvolvimento prático do sistema, será adotada a **Metodologia Ágil Scrum**, que é a escolha padrão da indústria para projetos de software que buscam flexibilidade e entrega rápida de valor (Schwaber & Sutherland, 2018).

A utilização do Scrum permite que a equipe se adapte rapidamente a feedbacks e requisitos emergentes, comuns em projetos que envolvem **Inteligência Artificial** e integração de APIs.

### 5.1 Estrutura Scrum

* **Sprints:** Ciclos de desenvolvimento curtos e fixos (sugerido 2 semanas).
* **Daily Scrum (Stand-ups):** Reuniões diárias de 15 minutos para alinhamento e identificação de impedimentos.
* **Product Backlog:** Lista priorizada de funcionalidades a serem implementadas.
* **Sprint Backlog:** Subconjunto do Product Backlog selecionado para a Sprint atual.

### 5.2 Fases do Projeto

* **Análise e Design (Início):**  
  Validação de requisitos, modelagem UML/ER, design de interface (Figma) e planejamento inicial (Product Backlog).
* **Desenvolvimento do Sistema (Sprints):**
  + **Backend:** Configuração da API (Node.js/Express), criação da camada de persistência (MySQL) e implementação dos endpoints de cadastro/login.
  + **Frontend:** Construção dos componentes de interface (React.js) para as telas de login, upload e histórico.
  + **Integração:** Conexão do backend com as APIs de IA (Gemini) e nutricionais (FatSecret/Edamam).
  + **Testes e Validação:** Após cada Sprint, serão realizados testes de unidade e testes de integração. O teste de aceitação será feito com a participação de usuários reais, coletando feedbacks cruciais para o ajuste da precisão da estimativa calórica.
* **Implantação e Monitoramento:**  
  O sistema será deployado em ambiente de produção, com monitoramento em tempo real do desempenho (latência) e da confiabilidade da IA.

# CRONOGRAMAS

O cronograma é uma ferramenta essencial para o gerenciamento de projetos, permitindo o controle de prazos e a alocação de recursos, mitigando o risco de atrasos (Pressman, 2016).

## Cronograma da Parte Teórica (setembro a novembro de 2025)

Detalha as fases de elaboração do estudo de caso, como levantamento teórico, definição de requisitos, criação de diagramas e protótipos, revisão e apresentação.

|  |  |
| --- | --- |
| **Atividades** | **Entrega** |
| **Levantamento teórico** |  |
| **Definição de requisitos** | 15 a 20 setembro |
| **Desenvolvimento dos protótipos** | 20 a 31 outubro |
| **Revisão e ajustes** | 01 a 15 novembro |
| **Apresentação** | 24 novembro |

*Tabela 3 - Cronograma da Parte Teórica (Setembro a Novembro de 2025)*

# CONSIDERAÇÕES FINAIS

Este estudo de caso estabelece uma proposta robusta e detalhada para o desenvolvimento de um sistema web inovador de controle nutricional. Ao integrar o poder da Inteligência Artificial com as melhores práticas da Engenharia de Software (Scrum, UML, ER) e tecnologias de ponta (*React.js, Node.js, MySQL*), o projeto oferece um *blueprint* completo. As diretrizes aqui definidas – desde o embasamento teórico, a modelagem de dados e interfaces, até o cronograma de implementação – fornecem o caminho técnico e metodológico necessário para a concretização de uma solução eficiente, escalável e de alto valor para o usuário.

# REFERÊNCIAS

Alura. (2022). *Entenda o Figma: uma solução inovadora para projetos de design*. Disponível em: <https://www.youtube.com/watch?v=9MS3MtC4B28>

Canaltech. (2025). *SnapCalorie: o app de contar calorias que utiliza inteligência artificial do Google*. Disponível em:

[https://canaltech.com.br/apps/app-promete-calcular-calorias-de-comida-pela-foto-com-](https://canaltech.com.br/apps/app-promete-calcular-calorias-de-comida-pela-foto-com-ajuda-de-ia-254192) [ajuda-de-ia-254192](https://canaltech.com.br/apps/app-promete-calcular-calorias-de-comida-pela-foto-com-ajuda-de-ia-254192)

Carvalho, B. V. de; Mello, C. H. P. (2012). Aplicação do método ágil scrum no desenvolvimento de produtos de software em uma pequena empresa de base tecnológica. *Gestão & Produção*, v. 19, n. 3, p. 557-571. Disponível em: <https://scholar.google.com/citations?user=YPNYtw0AAAAJ&hl=pt-BR>

Cubos Academy. (2023). *Figma para iniciantes: Comece com um guia para o design interativo*. Disponível em: <https://github.com/L3MON4D3/LuaSnip/discussions/335>

Fowler, M. (2003). *UML Distilled: A Brief Guide to the Standard Object Modeling Language* (3rd ed.). Addison-Wesley.

Franck, K. M. (2021). Diagrama Entidade-Relacionamento: uma ferramenta para modelagem de dados conceituais em Engenharia de Software. *Research, Society and Development*, v. 10, n. 8, e49510817776. Disponível em: <https://www.grafiati.com/pt/literature-selections/modelagem-entidade-relacionamento/>

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. LuizTools. (2021). *Programação Web com Node.js – Edição MySQL*. Disponível em: <https://www.luiztools.com.br/livro-node-mysql/>

MDN Web Docs. (2025). *Começando com React*. Disponível em: [https://developer.mozilla.org/pt-](https://developer.mozilla.org/pt-BR/docs/Learn_web_development/Core/Frameworks_libraries/React_getting_started) [BR/docs/Learn\_web\_development/Core/Frameworks\_libraries/React\_getting\_started](https://developer.mozilla.org/pt-BR/docs/Learn_web_development/Core/Frameworks_libraries/React_getting_started)

NextAge. (2025). *React + Node.js: por que essa dupla ainda domina o mercado*. Disponível em: <https://nextage.com.br/blog/react-and-node-js/>

PUC Minas. (2024). *Estudo de Caso sobre a Metodologia Scrum no Desenvolvimento de Software*. Disponível em: <http://bib.pucminas.br:8080/pergamumweb/vinculos/000031/0000319f.pdf>

Schwaber, K., & Sutherland, J. (2018). *O Guia do Scrum*. Scrum.org. Sommerville, I. (2011). *Engenharia de Software* (9ª ed.). Pearson Education.

UFC. (2023). *Node.js*. [Monografia]. Disponível em: [https://repositorio.ufsc.br/xmlui/bitstream/handle/123456789/130043/TCC%20Final.pdf?seq](https://repositorio.ufsc.br/xmlui/bitstream/handle/123456789/130043/TCC%20Final.pdf?sequence=1&isAllowed=y) [uence=1&isAllowed=y](https://repositorio.ufsc.br/xmlui/bitstream/handle/123456789/130043/TCC%20Final.pdf?sequence=1&isAllowed=y)

UX Tools. (2021). *Figma: o que é e como usar, suas funcionalidades e vantagens*. Disponível em: <https://ebaconline.com.br/blog/o-que-e-figma-e-como-usar>

W3C. (2017). *Boas Práticas para Dados na Web*. Disponível em:

<https://www.w3.org/Translations/DWBP-pt-BR/>

APIs Referenciadas:

API de Nutrição Edamam: [https://platform.fatsecret.com/platform-](https://platform.fatsecret.com/platform-api?keyword=edamam%20nutrition%20api&device=c&network=g&gad_source=1&gad_campaignid=127963119&gbraid=0AAAAAD97EmHGbv8kX5HLxwlJ9rLEMHpN3&gclid=CjwKCAjwu9fHBhAWEiwAzGRC_wIuYrSeoF1Typsm9P1XkL2U0cPe7p1MLZf3ghf9DX-oYGiBk05AchoC4TYQAvD_BwE) [api?keyword=edamam%20nutrition%20api&device=c&network=g&gad\_source=1&gad\_campa](https://platform.fatsecret.com/platform-api?keyword=edamam%20nutrition%20api&device=c&network=g&gad_source=1&gad_campaignid=127963119&gbraid=0AAAAAD97EmHGbv8kX5HLxwlJ9rLEMHpN3&gclid=CjwKCAjwu9fHBhAWEiwAzGRC_wIuYrSeoF1Typsm9P1XkL2U0cPe7p1MLZf3ghf9DX-oYGiBk05AchoC4TYQAvD_BwE) [ignid=127963119&gbraid=0AAAAAD97EmHGbv8kX5HLxwlJ9rLEMHpN3&gclid=CjwKCAjwu9fHB](https://platform.fatsecret.com/platform-api?keyword=edamam%20nutrition%20api&device=c&network=g&gad_source=1&gad_campaignid=127963119&gbraid=0AAAAAD97EmHGbv8kX5HLxwlJ9rLEMHpN3&gclid=CjwKCAjwu9fHBhAWEiwAzGRC_wIuYrSeoF1Typsm9P1XkL2U0cPe7p1MLZf3ghf9DX-oYGiBk05AchoC4TYQAvD_BwE) [hAWEiwAzGRC\_wIuYrSeoF1Typsm9P1XkL2U0cPe7p1MLZf3ghf9DX-](https://platform.fatsecret.com/platform-api?keyword=edamam%20nutrition%20api&device=c&network=g&gad_source=1&gad_campaignid=127963119&gbraid=0AAAAAD97EmHGbv8kX5HLxwlJ9rLEMHpN3&gclid=CjwKCAjwu9fHBhAWEiwAzGRC_wIuYrSeoF1Typsm9P1XkL2U0cPe7p1MLZf3ghf9DX-oYGiBk05AchoC4TYQAvD_BwE) [oYGiBk05AchoC4TYQAvD\_BwE](https://platform.fatsecret.com/platform-api?keyword=edamam%20nutrition%20api&device=c&network=g&gad_source=1&gad_campaignid=127963119&gbraid=0AAAAAD97EmHGbv8kX5HLxwlJ9rLEMHpN3&gclid=CjwKCAjwu9fHBhAWEiwAzGRC_wIuYrSeoF1Typsm9P1XkL2U0cPe7p1MLZf3ghf9DX-oYGiBk05AchoC4TYQAvD_BwE)

API de Nutrição FatSecret: <https://platform.fatsecret.com/platform-api>

API de Reconhecimento de Imagem Gemini: [https://ai.google.dev/gemini-api/docs/image-](https://ai.google.dev/gemini-api/docs/image-understanding?hl=pt-br) [understanding?hl=pt-br](https://ai.google.dev/gemini-api/docs/image-understanding?hl=pt-br)