

**UNIVERSIDADE PAULISTA - UNIP**  
**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**ALLAN OLIVEIRA FRANÇA - G5923I0**  
**EDUARDO SUSUMU NISHIO - R108988**  
**JOSE HENRIQUE MATIOLI - G898044**  
**JOYCE VICTORIA SENA – T518BJ6**  
**VINICIUS ALVES VALENTIM - R091DF0**  
**VICTOR HUGO MARTELLI COSTA - G9858J7**

**SISTEMA PARA GESTÃO DE CHAMADOS**  
**E SUPORTE TÉCNICO COM INTEGRAÇÃO DE IA**

**ARARAQUARA**  
**2025**

**ALLAN OLIVEIRA FRANÇA - G5923I0**  
**EDUARDO SUSUMU NISHIO - R108988**  
**JOSE HENRIQUE MATIOLI - G898044**  
**JOYCE VICTORIA SENA – T518BJ6**  
**VINICIUS ALVES VALENTIM - R091DF0**  
**VICTOR HUGO MARTELLI COSTA - G9858J7**

**SISTEMA PARA GESTÃO DE CHAMADOS  
E SUPORTE TÉCNICO COM INTEGRAÇÃO DE IA**

Projeto Integrado Multidisciplinar - Análise  
e Desenvolvimento de Sistemas

**ARARAQUARA**

**2025**

## RESUMO

O presente projeto tem como objetivo o desenvolvimento do Little Bird System 2.0, um sistema integrado de gestão de chamados e suporte técnico baseado em inteligência artificial (IA). Diante do avanço da transformação digital e da crescente dependência das organizações por sistemas tecnológicos, o projeto busca oferecer uma solução moderna que automatize e otimize o fluxo de chamados, garantindo mais agilidade, organização e eficiência no atendimento técnico.

O sistema conta com uma interface intuitiva e responsiva, além de um assistente virtual capaz de realizar triagens automáticas, sugerir soluções e encaminhar solicitações aos atendentes responsáveis. Com isso, o Little Bird System 2.0 visa aprimorar os processos internos das empresas, reduzir o tempo de resposta e aumentar a produtividade, representando um avanço na modernização do suporte técnico corporativo.

**Palavras-Chave:** Help-Desk, Gestão de Chamados, Software, Inteligência Artificial e Tecnologia.

## **ABSTRACT**

This project aims to develop the Little Bird System 2.0, an integrated system for technical support and ticket management based on artificial intelligence (AI). With the advancement of digital transformation and the growing dependence of organizations on technological systems, the project seeks to offer a modern solution that automates and optimizes the flow of support requests, ensuring greater agility, organization, and efficiency in technical services.

The system features an intuitive and responsive interface, as well as a virtual assistant capable of performing automatic triage, suggesting solutions, and forwarding requests to the appropriate support staff. Thus, Little Bird System 2.0 aims to improve internal processes, reduce response time, and increase productivity, representing a significant step forward in the modernization of corporate technical support..

**Keywords:** Help Desk, Ticket Management, Software, Artificial Intelligence and Technology.

## LISTA DE ILUSTRAÇÕES

Imagem 1 - Diagrama ER Banco de Dados.....	17
Imagem 2 - Diagrama de Casos de Uso.....	22
Imagem 3 - Diagrama de Classes UML.....	24
Imagem 4 - Tela de Login.....	27
Imagem 5 - Dashboard.....	28
Imagem 6 - Tela Criar Novo Chamado.....	29
Imagem 7 - Conversa com o Chat Bot.....	30
Imagem 8 - Interface SablÁ.....	31
Imagem 9 - Dashboard Atendente.....	32
Imagem 10 - Detalhamento de Chamados.....	33
Imagem 11 - Tempo de Resposta.....	34
Imagem 12 - Dashboard Administrador.....	35
Imagem 13 - Relatórios.....	36
Imagem 14 - Botão Exportar.....	37

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>8</b>
1.1	Contextualização.....	8
1.2	Cenário: Situação Problema.....	9
1.3	Objetivos.....	10
1.3.1	Objetivo Geral:.....	10
1.3.2	Objetivos Específicos.....	10
<b>2</b>	<b>ARQUITETURA DO PROJETO.....</b>	<b>11</b>
2.1	Visão Geral.....	11
2.2	Clean Architecture no Back-End.....	12
2.3	CQRS (Command Query Responsibility Segregation).....	12
2.4	Front-End.....	12
2.5	Banco de Dados.....	13
2.6	Controllers e Endpoints.....	14
2.6.1	Principais endpoints:.....	14
2.7	Interfaces (POO).....	14
2.8	Fluxos principais do Sistema.....	15
2.9	Integração com o Google Gemini.....	15
2.10	Segurança.....	16
<b>3</b>	<b>MODELAGEM.....</b>	<b>17</b>
3.1	Entidades e Atributos.....	17
<b>4</b>	<b>DIAGRAMA ER.....</b>	<b>18</b>
4.1	Usuário.....	18
4.2	Solicitante.....	19
4.3	Atendente.....	19
4.4	Chamado.....	19
4.5	Resposta.....	20
4.6	Status.....	21
4.7	Relatório.....	21
4.8	BotInteracoes.....	21
<b>5</b>	<b>DIAGRAMAS UML.....</b>	<b>22</b>
5.1	Diagrama de Casos de Uso.....	22
5.2	Diagrama de Classes.....	24
5.3	Descrição geral do sistema.....	25
5.4	Classes e Responsabilidades.....	26
5.4.1	Relacionamentos Principais.....	27
5.5	Fluxo geral do sistema.....	27
<b>6</b>	<b>MANUAL DE USO.....</b>	<b>28</b>
6.1	Como realizar a abertura de um chamado.....	28
6.2	Como entrar em contato com a Inteligência Artificial sablá:.....	31

6.3	Como atender os chamados.....	33
6.4	Como realizar a emissão de relatórios.....	36
7	FAQ DE SUPORTE DE T.I.....	39
7.1	Meu computador não está ligando. O que devo fazer ?.....	39
7.2	Minha internet está lenta ou não conecta. Como resolver?.....	39
7.3	Não consigo acessar meu e-mail corporativo. O que está errado?.....	40
7.4	Meu software trava ou fecha inesperadamente. Como corrigir?.....	40
7.5	Esqueci minha senha de login no sistema corporativo. Como recupero?.....	40
7.6	Meu computador está muito lento. Como melhorar o desempenho?.....	41
7.7	Não consigo imprimir documentos na impressora da rede. O que fazer?.....	41
7.8	Recebi uma mensagem de erro sobre espaço em disco insuficiente. Como resolver?.....	42
8	IMPLEMENTAÇÃO DE ESTRATÉGIAS DE CONFORMIDADE COM A LGPD NA LITTLE BIRD.....	43
9	CONCLUSÃO.....	44
	REFERÊNCIAS.....	46
	APÊNDICES.....	49
	APÊNDICE A - REPOSITÓRIO DO FRONT-END.....	49
	APÊNDICE B - REPOSITÓRIO DO BACK-END.....	49
	ANEXOS.....	50
	ANEXO A - FICHA DE CONTROLE.....	50

## **1 INTRODUÇÃO**

Com o avanço da transformação digital e a crescente dependência das organizações por sistemas tecnológicos, tornou-se essencial contar com ferramentas inteligentes de gestão e suporte técnico capazes de acompanhar o ritmo das mudanças e atender às novas demandas do mercado. As empresas modernas, independentemente do seu porte ou segmento, têm buscado soluções que promovam maior eficiência operacional, comunicação assertiva e agilidade na resolução de problemas internos.

No entanto, muitas ainda enfrentam dificuldades em manter um atendimento eficaz a seus colaboradores e clientes, especialmente devido à ausência de sistemas integrados que permitam organizar, priorizar e automatizar o fluxo de chamados, resultando em retrabalhos, atrasos e perda de produtividade. Nesse contexto, torna-se evidente a necessidade de adotar tecnologias inovadoras que aliem inteligência artificial, automação e gestão eficiente, contribuindo diretamente para a otimização dos processos corporativos e a melhoria contínua dos serviços prestados.

### **1.1 Contextualização**

Nesse contexto, surge o Sistema Integrado para Gestão de Chamados e Suporte Técnico com integração de Inteligência Artificial (IA), uma solução moderna e escalável voltada para empresas que buscam otimizar o processo de atendimento técnico. O sistema tem como foco principal a automatização e centralização das solicitações de suporte, garantindo mais agilidade, precisão e controle em cada etapa do processo.

Com dois perfis de usuários — Atendentes e Solicitantes —, a plataforma oferece uma experiência personalizada e inteligente. Além disso, a integração com um assistente virtual baseado em IA possibilita o atendimento automatizado, sugerindo soluções, criando chamados de forma autônoma e encaminhando demandas complexas aos atendentes humanos. Essa abordagem promove redução do tempo médio de resposta, melhoria na produtividade da equipe de TI e aumento na satisfação dos usuários.



O projeto representa um avanço significativo na modernização da gestão de suporte técnico, aliando inteligência artificial, automação de processos e análise de dados a uma interface intuitiva e responsiva, preparada para diferentes contextos corporativos.

## 1.2 Cenário: Situação Problema

Empresas de médio porte que possuem setores internos de tecnologia frequentemente enfrentam desafios como excesso de demandas manuais, falta de padronização nos atendimentos e dificuldade em acompanhar métricas de desempenho. Na maioria dos casos, o registro de chamados é feito por e-mail, mensagens ou planilhas, o que gera retrabalho, atrasos na resolução e baixa visibilidade dos processos.

O Sistema Integrado para Gestão de Chamados e Suporte Técnico Baseado em IA surge como resposta a essas dificuldades. Seu objetivo é automatizar o fluxo de atendimento desde a abertura do chamado até o encerramento, utilizando recursos de IA para triagem automática, classificação por prioridade e geração de relatórios analíticos.

O sistema atua em três frentes principais:

1. **Atendimento Inteligente:** O bot de IA realiza a triagem inicial, identifica o tipo de solicitação e sugere soluções com base em interações anteriores;
2. **Gestão Eficiente de Chamados:** O painel administrativo permite que atendentes e gestores monitorem, atribuam e acompanhem chamados em tempo real;
3. **Análise e Aprendizado Contínuo:** O sistema coleta dados sobre o desempenho dos atendimentos, alimentando relatórios que auxiliam na tomada de decisão e aprimoram continuamente o serviço prestado.

Dessa forma, o projeto visa transformar o suporte técnico em um processo automatizado, mensurável e inteligente, eliminando gargalos operacionais e aumentando a eficiência do ambiente corporativo.

## 1.3 Objetivos

### 1.3.1 Objetivo Geral:

Desenvolver um sistema integrado de gestão de chamados e suporte técnico com inteligência artificial, capaz de automatizar o atendimento, melhorar a eficiência dos processos internos e proporcionar uma experiência mais ágil e inteligente aos usuários.

### 1.3.2 Objetivos Específicos

- Criar um assistente virtual baseado em IA capaz de realizar triagem e resposta inicial aos usuários;
- Automatizar o registro e categorização de chamados, definindo prioridade conforme o tipo de solicitação;
- Permitir o acompanhamento em tempo real do status de cada chamado;
- Implementar relatórios gerenciais e dashboards analíticos com métricas de desempenho e SLA;
- Garantir uma interface responsiva e intuitiva, acessível tanto em desktop quanto em dispositivos móveis;
- Otimizar o tempo médio de resolução de chamados, melhorando a produtividade do time de suporte;
- Assegurar segurança e integridade dos dados através de autenticação e criptografia;
- Utilizar aprendizado de máquina (Machine Learning) para aperfeiçoar continuamente as respostas automáticas do sistema;
- Promover inovação tecnológica no processo de suporte e atendimento corporativo.

## 2 ARQUITETURA DO PROJETO

Esta documentação técnica apresenta o projeto PIM Help Desk de forma completa, abrangendo o back-end, front-end e banco de dados. Embora o código-fonte não possua comentários detalhados em todas as partes, este documento foi construído com base na análise estrutural do projeto e descreve todos os pontos cruciais de funcionamento, arquitetura e interação entre as camadas do sistema.

### 2.1 Visão Geral

Este documento apresenta a versão aprimorada da documentação do PIM, integrando práticas modernas como Clean Architecture, CQRS, front-end baseado em Vite + React + TypeScript + ShadCN UI e melhorias arquiteturais no back-end ASP.NET Core.

A solução adota um modelo distribuído em camadas bem definidas, seguindo os princípios da Clean Architecture para alcançar alta modularidade, baixo acoplamento e testabilidade. Comunicação entre front-end e back-end ocorre via API REST segura.

O sistema PIM Help Desk é uma aplicação web full stack baseada no padrão MVC e arquitetura RESTful. O projeto foi desenvolvido em .NET 9 no back-end, com Entity Framework Core para persistência de dados, e interface front-end integrando-se à API via requisições HTTP. O sistema tem como objetivo gerenciar chamados de suporte (tickets), controlando usuários, atendentes, administradores e logs de operação.

A arquitetura é dividida em três camadas principais:

- **Front-end:** Interface gráfica e interação do usuário;
- **Back-end:** Camada lógica e de controle, desenvolvida em ASP.NET Core;
- **Banco de Dados:** Armazenamento relacional em SQL Server, gerenciado via Entity Framework Core.

## 2.2 Clean Architecture no Back-End

O back-end em ASP.NET Core foi reorganizado para seguir Clean Architecture, distribuído nas camadas:

- **Domain:** entidades, enums, interfaces base e regras de negócio puras;
- **Application:** casos de uso, DTOs;
- **Infrastructure:** Entity Framework Core, SQL Server, repositórios, migrations;
- **Presentation:** Controllers e endpoints REST.

Essa separação garante isolamento entre lógica de negócio e detalhes técnicos.

## 2.3 CQRS (Command Query Responsibility Segregation)

É um padrão arquitetural que separa a responsabilidade de escrita (comandos) e leitura (consultas) em sistemas de software. Implementamos por causa dos seguintes benefícios:

- Maior organização dos casos de uso;
- Performance otimizada para consultas complexas;
- Menos efeitos colaterais entre módulos.

À seguir vamos listar exemplos de Query e Command:

- **Query:** Listar tickets, buscar usuários;
- **Command:** Criar ticket, atualizar usuário, autenticar login.

## 2.4 Front-End

O front-end foi projetado para alta performance utilizando:

- Vite: bundler ultrarrápido para desenvolvimento;
- React: Biblioteca declarativa e componentizada;
- TypeScript: Tipagem estática, facilitando integração com API;

- Shadcn UI: Componentes modernos, acessíveis e estilizados com Tailwind CSS.

Essas ferramentas tem benefícios bem específicos como:

- Renderização eficiente;
- Componentização escalável;
- Interface limpa, responsiva e coerente.

## **2.5 Banco de Dados**

Mantém-se SQL Server como banco relacional principal.

Justificativas:

- Integração nativa com .NET e EF Core;
- Migrations versionadas;
- Confiabilidade e escalabilidade.

Estruturas principais:

- Users;
- Administrator;
- Attendant;
- Requester;
- Ticket;
- Ticket\_Responses;
- Log.

## 2.6 Controllers e Endpoints

Cada controller chama seus respectivos casos de uso (Handlers CQRS) em vez de acessar serviços diretamente. Controllers na camada Presentation expõem as funcionalidades da API.

### 2.6.1 Principais endpoints:

- GET/tickets ⇒ Query de listagem;
- POST/tickets ⇒ Command para criação;
- PUT/tickets ⇒ Command de atualização (utiliza o Id);
- DELETE/tickets ⇒ Command de remoção (utiliza o Id).

Esse modelo mantém o padrão REST, mas organizado por CQRS.

## 2.7 Interfaces (POO)

Interfaces são usadas nas camadas Domain e Application para garantir baixo acoplamento. Algumas funções principais são:

- Definir contratos de repositórios;
- Permitir injeção de dependência;
- Facilitar testes unitários.

E alguns exemplos são:

- IUserRepository;
- ITicketRepository;
- IAuthService.

## 2.8 Fluxos principais do Sistema

1. Abertura de chamado via CommandHandler;
2. Consulta de tickets via QueryHandler;
3. Autenticação e geração de JWT;
4. Painéis diferentes para Administrator, Attendant e Requester.

O desenvolvimento do sistema demonstrou a importância de aplicar práticas arquiteturais modernas como Clean Architecture e CQRS. O uso de ferramentas atuais como Vite, React e TypeScript resultou em uma solução eficiente, escalável e com código sustentável. A aplicação final atende aos requisitos do PIM, proporcionando um sistema robusto e bem estruturado.

## 2.9 Integração com o Google Gemini

Com o objetivo de elevar a eficiência operacional do sistema de Help Desk e otimizar o processo de atendimento, foi implementada uma integração direta com o Google Gemini, o modelo de Inteligência Artificial generativa de última geração da Google. Esta integração amplia as capacidades do sistema ao permitir automação inteligente, análise contextual e suporte avançado aos atendentes.

A integração foi implementada respeitando a Clean Architecture. A lógica de IA reside na camada Infrastructure, enquanto os casos de uso são orquestrados pela camada Application (via Handlers CQRS). O fluxo é: Controller ⇒ Handler ⇒ Serviço Gemini ⇒ Resposta Estruturada ⇒ Front-End.

## Funcionalidades habilitadas

- Classificação automática de tickets;
- Sugestão de respostas técnicas para atendentes;
- Resumo automático de históricos;
- Geração de insights e relatórios inteligentes.

### **2.10 Segurança**

Toda comunicação com o Gemini utiliza HTTPS e chaves protegidas em variáveis de ambiente, sem exposição de informações sensíveis.



### **3 MODELAGEM**

A modelagem conceitual do sistema de Help Desk foi construída com base na arquitetura Clean Architecture e no padrão CQRS, conforme descrito na documentação. A entidade User representa os participantes da plataforma, armazenando credenciais, perfis e informações de autenticação. A entidade Ticket centraliza o fluxo de atendimento, incluindo abertura, prioridade, status e encerramento. Para garantir comunicação estruturada, TicketResponse registra todas as interações realizadas durante o atendimento.

A entidade LogSistema oferece rastreabilidade completa das ações executadas no sistema, garantindo conformidade com requisitos de auditoria e segurança. Já a entidade Perfil estabelece níveis hierárquicos de acesso, essenciais para o controle granular de permissões. Essa modelagem se alinha aos princípios de separação de responsabilidades, facilitando a organização entre comandos (escrita) e consultas (leitura), tornando o sistema mais escalável e fácil de manter.

#### **3.1 Entidades e Atributos**

No contexto de modelagem de dados e sistemas, entidades e atributos são conceitos fundamentais para organizar a informação. Uma entidade representa qualquer objeto ou conceito do mundo real sobre o qual queremos guardar informações, ou seja, é um conjunto de informações sobre um determinado assunto.

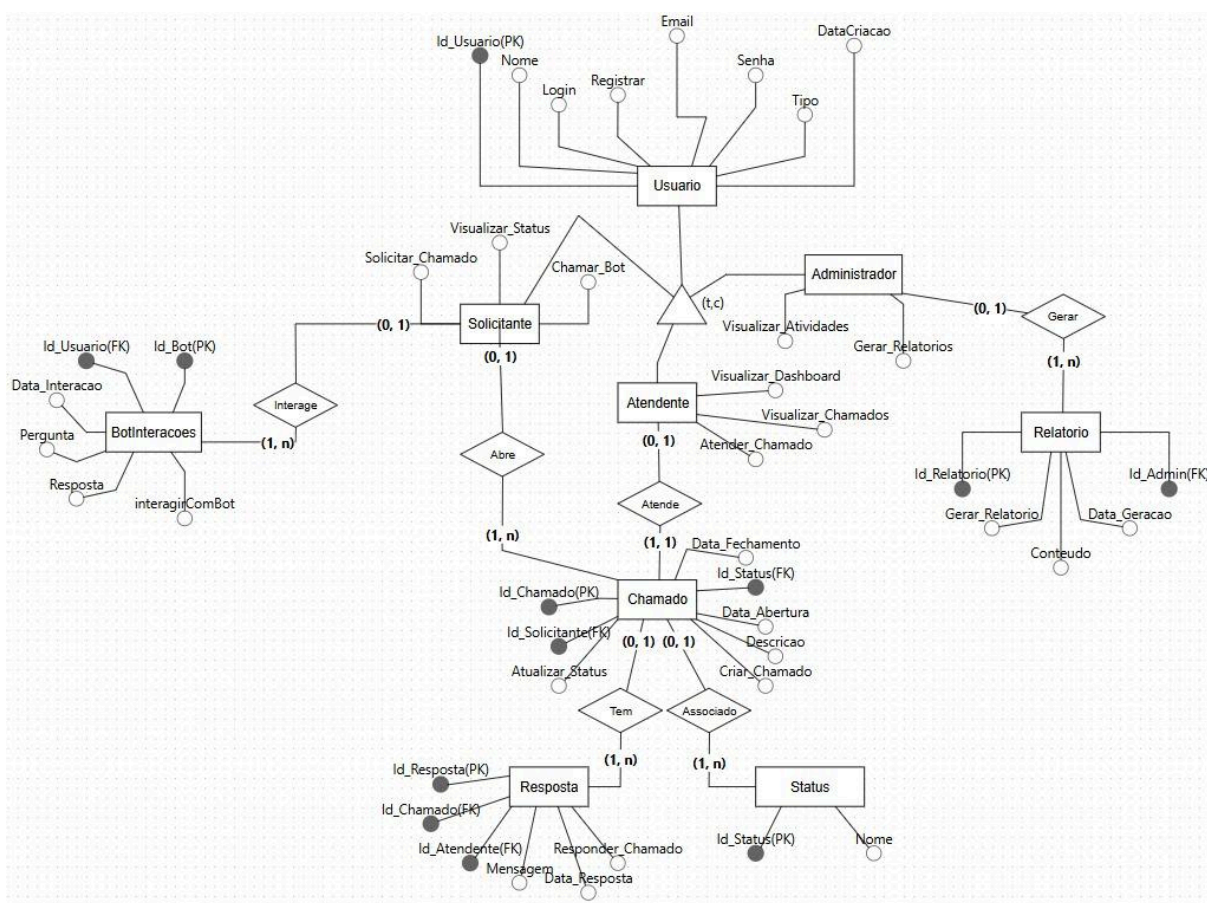
Um atributo é uma característica ou propriedade que descreve aquela entidade, ou seja, são as informações específicas que você quer armazenar sobre aquela entidade.

Agora que foi esclarecido sobre o conceito de Entidades e Atributos, vamos listar quais são as principais em nosso sistema Help Desk.

## 4 DIAGRAMA ER

Felizmente existe uma maneira de ilustrar tudo por meio de imagem, através do Diagrama ER de Banco de Dados (Imagem 1).

Imagem 1 - Diagrama ER Banco de Dados.



Fonte: própria (BR Modelo Web).

### 4.1 Usuário

Atributos:

- Id\_Usuario (PK);
- Nome;
- Email;
- Login;

- Senha;
- DataCriacao;
- Tipo.

**Descrição:** Armazena as informações básicas de todos os usuários do sistema, podendo ser Solicitante, Atendente ou Administrador.

## 4.2 Solicitante

Atributos:

- Solicita chamados;
- Chama BotInteracoes.

**Descrição:** Representa o usuário que solicita a abertura de um chamado ou interage com o bot.

## 4.3 Atendente

Atributos:

- AtenderChamados;
- VisualizarChamados;
- VisualizarDashboard.

**Descrição:** Usuário responsável por atender e atualizar chamados.

## 4.4 Chamado

Atributos:

- Id\_Chamado (PK);
- Id\_Solicitante (FK);

- Id\_Status (FK);
- Data\_Abertura;
- Data\_Fechamento;
- Descricao.

**Descrição:** Registra as solicitações feitas pelos usuários (Solicitantes) e gerenciadas pelos atendentes.

#### 4.5 Resposta

Atributos:

- Id\_Resposta (PK);
- Id Chamado (FK);
- Id\_Atendente (FK);
- Id\_Status (FK);
- Mensagem;
- Data\_Resposta.

**Descrição:** Contém as respostas dadas pelos atendentes aos chamados.

## 4.6 Status

Atributos:

- Id\_Status (PK);
- Nome.

**Descrição:** Define o estado atual de um chamado, como “Aberto”, “Em andamento” ou “Fechado”.

## 4.7 Relatório

Atributos:

- Id\_Relatorio;
- Id\_Admin;
- Data\_Geracao;

**Descrição:** Armazena os relatórios gerados pelo administrador, contendo informações sobre atividades, atendimentos ou chamados.

## 4.8 BotInteracoes

Atributos:

- Id\_Bot (PK);
- Id\_Usuario (FK);
- Data\_Interacao;
- Pergunta;
- Resposta.

**Descrição:** Registra as interações dos usuários com o bot do sistema, armazenando as perguntas e respostas trocadas.

## 5 DIAGRAMAS UML

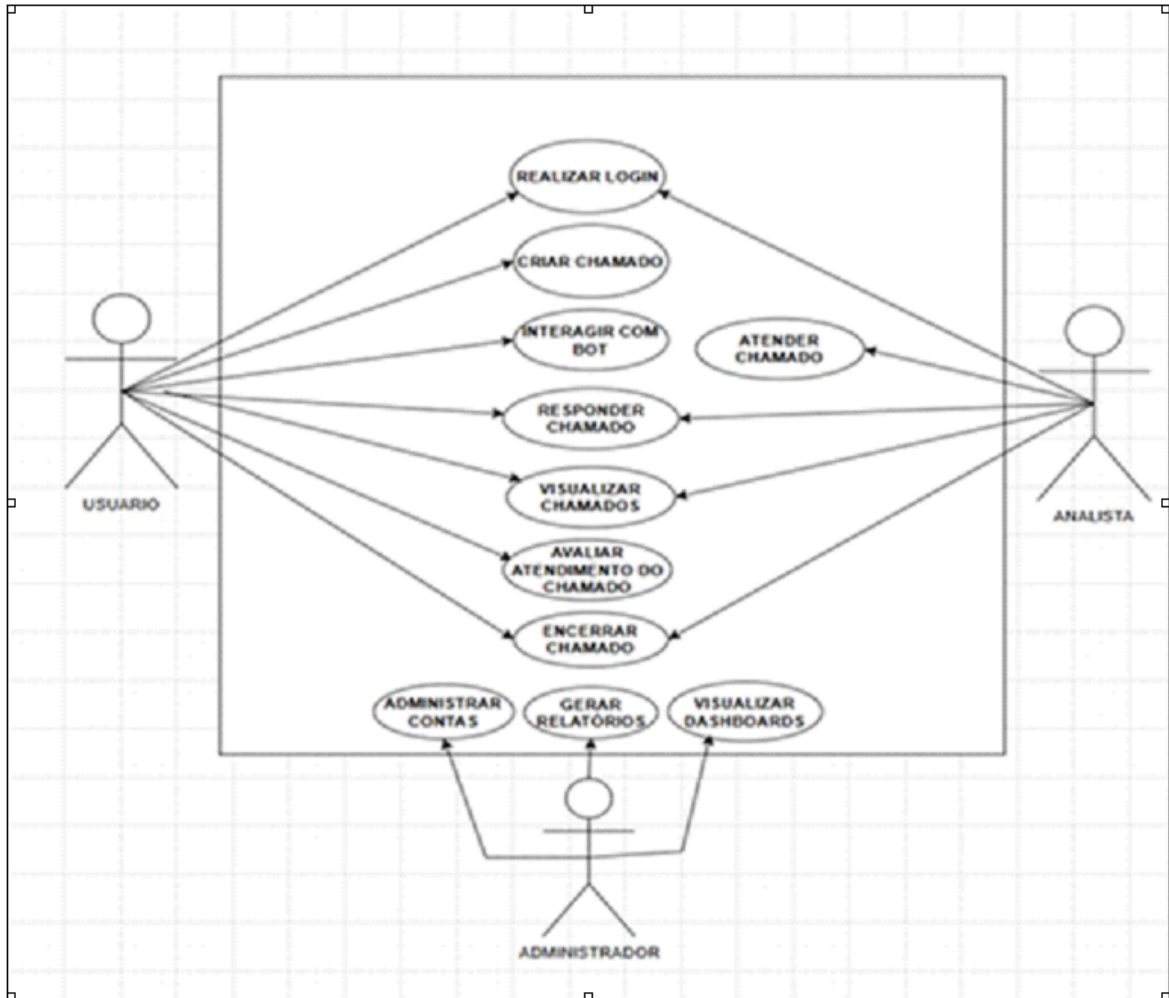
A UML (Unified Modeling Language) é um conjunto padronizado de diagramas usado na engenharia de software. Ela é uma se fosse a “planta” do projeto. Não se trata de uma linguagem de programação, mas sim de uma linguagem visual para:

- **Visualizar:** Criar uma representação gráfica de um sistema;
- **Especificar:** Detalhar os requisitos e a estrutura do sistema;
- **Construir:** Servir como um guia para os desenvolvedores escreverem o código;
- **Documentar:** Registrar o design e o funcionamento do sistema para referência futura.

### 5.1 Diagrama de Casos de Uso

O sistema de chamados tem como objetivo permitir que usuários possam registrar solicitações de suporte, que serão atendidas pelo analista e gerenciadas pelo administrador. O diagrama demonstra as principais interações entre os atores e o sistema, desde o login até a geração de relatórios (Imagem 2).

Imagem 2 - Diagrama de Casos de Uso



Fonte: própria (Lucidchart).

Como podemos visualizar, algumas tarefas são imprescindíveis no contexto do nosso sistema, dentre elas:

- Realizar Login;
- Criar chamado;
- Interagir com Bot;
- Atender Chamado;
- Responder Chamado;
- Visualizar Chamados;
- Avaliar Atendimento;

- Encerrar Chamado;
- Administrar Contas;
- Gerar Relatórios;
- Visualizar Dashboards.

## 5.2 Diagrama de Classes

No âmbito da Engenharia de Software, o diagrama de classes UML se destaca como um dos mais fundamentais e amplamente utilizados. Este diagrama pertence à categoria de diagramas estruturais, ou seja, tem como foco principal descrever a arquitetura estática do sistema. Basicamente ele mostra o que o sistema é, e não como se comporta ao longo do tempo (comportamento dinâmico).

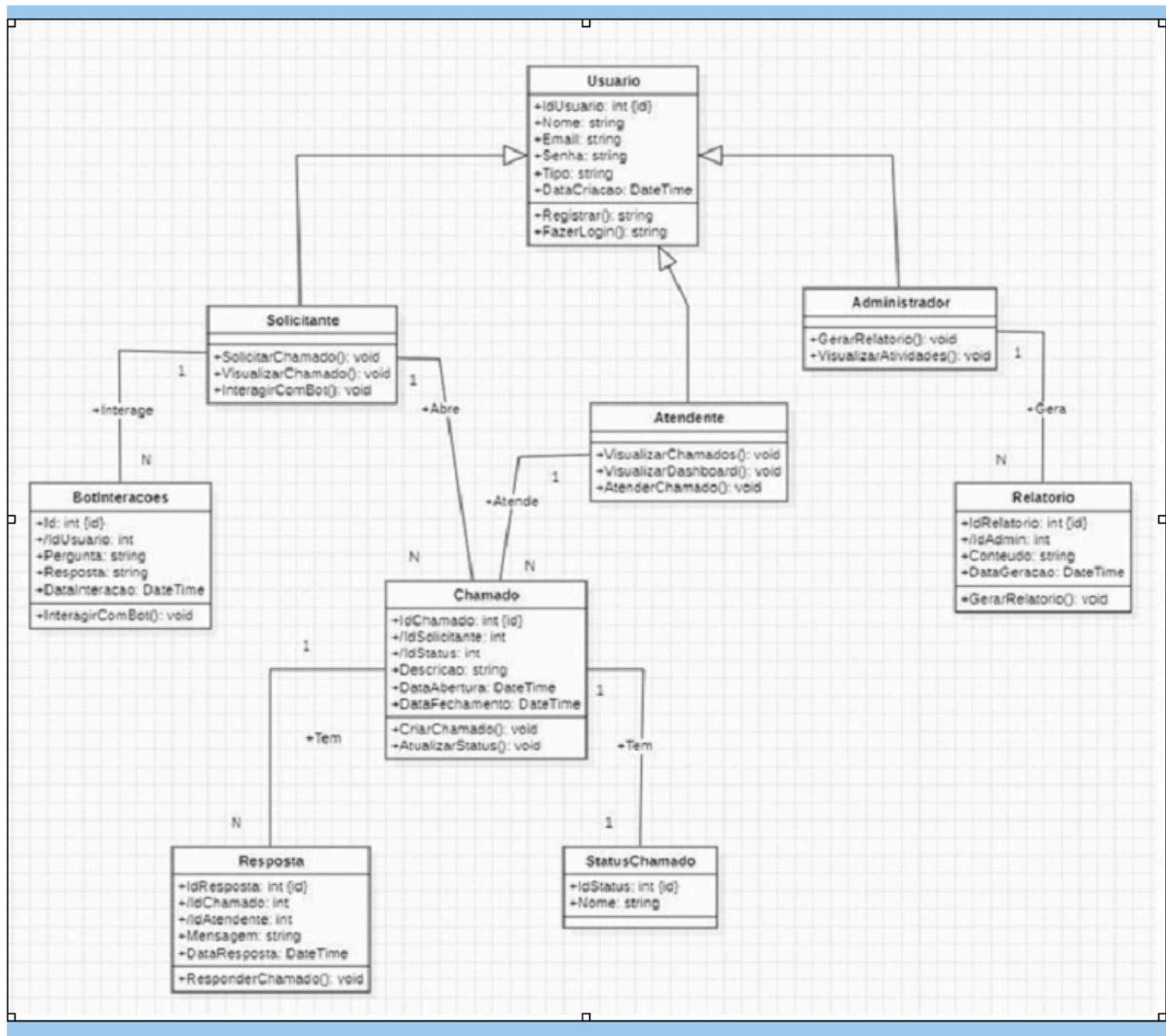
O Diagrama de Classes serve como uma espécie de “planta baixa” do software, detalhando os seguintes componentes essenciais:

- **Classes:** As abstrações das principais entidades ou conceitos do sistema (Cliente, Produto, Pedido). Elas são representadas por retângulos divididos em três seções;
- **Atributos:** As propriedades ou dados que cada classe armazena (no caso de Cliente, tem nome e CPF). Ocupam a segunda posição o retângulo;
- **Métodos:** São os comportamentos (ações) que determinada Classe pode executar. Ocupam a terceira seção.



Obviamente cada software tem uma estrutura e contextos diferentes. Vamos anexar o nosso Diagrama para que fique claro (Imagem 3).

Imagem 3 - Diagrama de Classes UML



Fonte: própria (Draw DB)

### 5.3 Descrição geral do sistema

O Sistema de Chamados tem como objetivo gerenciar solicitações de suporte técnico. Ele permite que usuários registrem chamados, analistas realizem o atendimento e administradores supervisionem e gerem relatórios.

O diagrama de classes representa as entidades do sistema, seus atributos, métodos e relacionamentos, descrevendo de forma clara como os objetos interagem entre si.

## 5.4 Classes e Responsabilidades

A classe Usuário é genérica, pense nela como a base para o restante das classes. Quando criamos a classe Usuário, passamos para ela todos os atributos e métodos que outras classes semelhantes possam usar, como por exemplo um Solicitante, Atendente e até Administrador. Desta forma, reaproveitamos código e aplicamos o conceito de Herança (classe filha herda da classe pai).

Os **atributos** para essa classe pai seriam:

- Id;
- Nome;
- Email;
- Senha;
- Tipo;
- DataCriacao.

Enquanto os **métodos** seriam:

- Registrar;
- FazerLogin.

À partir da classe pai (Usuário), as outras classes podem acessar os mesmos atributos e métodos dado o contexto específico da própria classe. Seja um Solicitante, Atendente e afins.

A classe Chamado representa a solicitação de suporte criada por um solicitante e atendida por um analista.

Temos também no nosso contexto a classe BotInteracoes, que registra as interações entre o solicitante e o bot do sistema.

A classe Resposta serve para armazenar as respostas dadas pelos atendentes aos chamados.

Outra classe de suma importância é a classe “StatusChamado”, que serve para sabermos o estado atual do chamado, se está “Aberto”, “Em andamento” ou “Encerrado”.

Para gerar informações e estatísticas sobre os atendimentos realizados no sistema, criamos a classe “Relatório”.

#### 5.4.1 Relacionamentos Principais

- Solicitante abre chamados;
- Atendente atende e responde chamados;
- Administrador supervisiona o sistema e gera relatórios;
- Chamados possuem Status e Respostas;
- Usuários podem interagir com o Bot para suporte automatizado.

#### 5.5 Fluxo geral do sistema

- O Solicitante abre o chamado e descreve o problema;
- O Atendente recebe o chamado e realiza o atendimento;
- O chamado passa por diferentes Status até ser encerrado;
- O Administrador pode acessar, controlar e gerar relatórios sobre todos os chamados;
- O Bot auxilia o Solicitante em interações automáticas.

Esse é o coração do sistema, a principal ideia. Modelar dados de forma correta ajuda na prevenção de erros durante a codificação, isso garante um código mais limpo e, conseqüentemente, um melhor desempenho. Diagramas ajudam a visualizar e o uso é crucial.

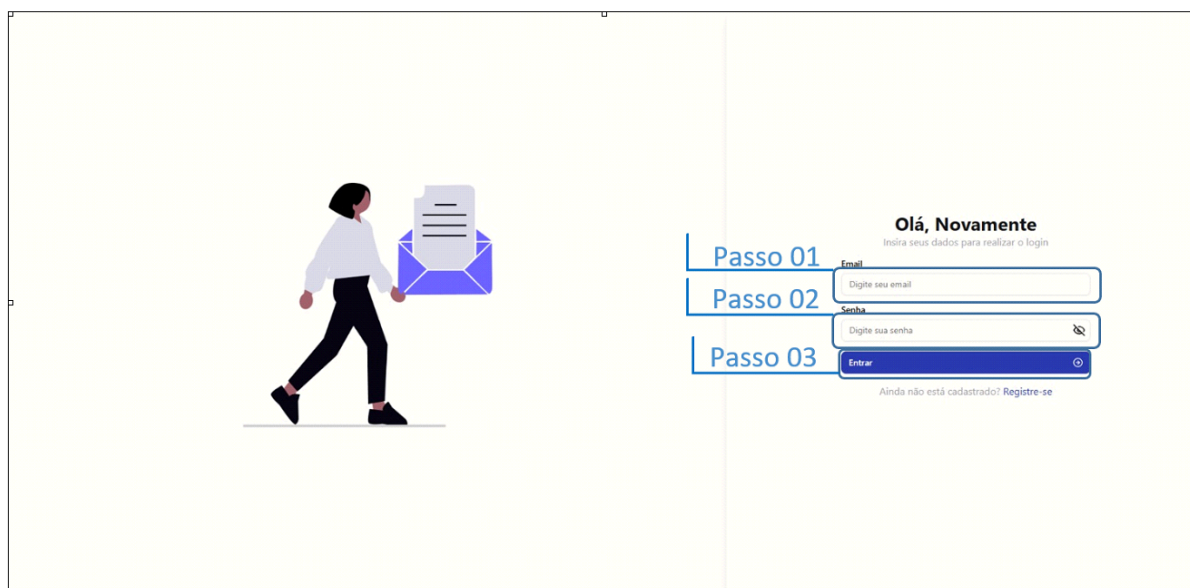
## 6 MANUAL DE USO

### 6.1 Como realizar a abertura de um chamado

Público-alvo: Cliente, Atendente;

1. Faça login com a conta fornecida pelo TI ao ingressar na empresa.  
Insira seu nome de usuário (Passo 1), depois a senha (Passo 2) e pressione Enter ou clique em “Entrar” (Passo 3). (Imagem 4).

Imagem 4 - Tela de Login.



A tela de login é dividida em duas seções principais. À esquerda, há uma ilustração de uma pessoa caminhando e segurando uma pasta azul com documentos. À direita, o formulário de login é intitulado "Olá, Novamente" com o subtítulo "Insira seus dados para realizar o login". O formulário é dividido em três passos: "Passo 01" para o email, "Passo 02" para a senha e "Passo 03" para o botão "Entrar".

**Olá, Novamente**  
Insira seus dados para realizar o login

**Passo 01** Email  
Digite seu email

**Passo 02** Senha  
Digite sua senha

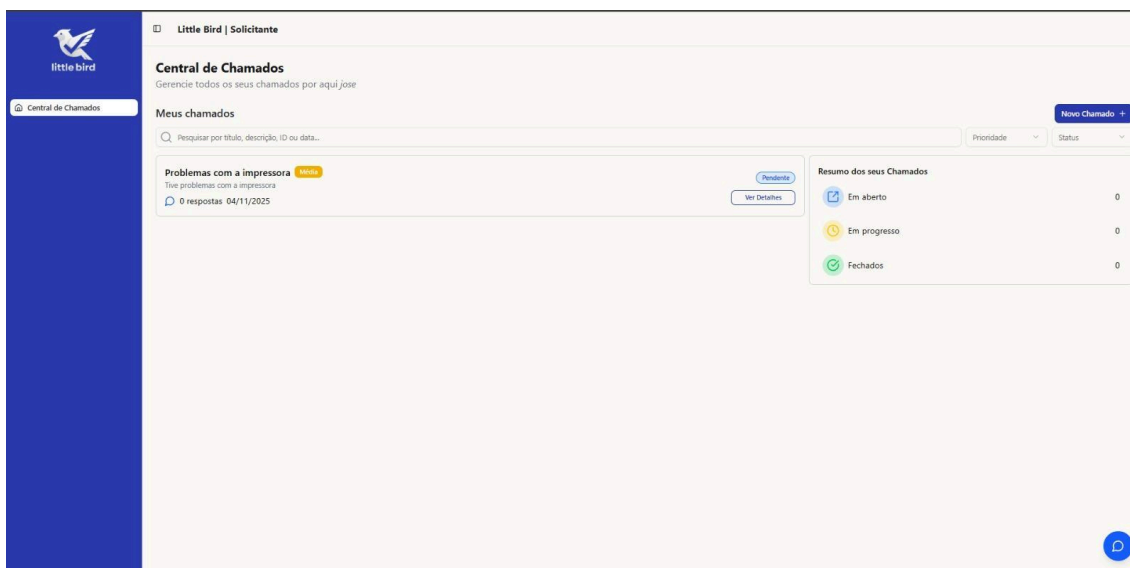
**Passo 03** Entrar

Ainda não está cadastrado? [Registre-se](#)

Fonte: própria (Sistema Web).

1. Quando tiver concluído a etapa anterior, ela vai te levar à tela abaixo, clique em “Novo Chamado” (Imagem 5).

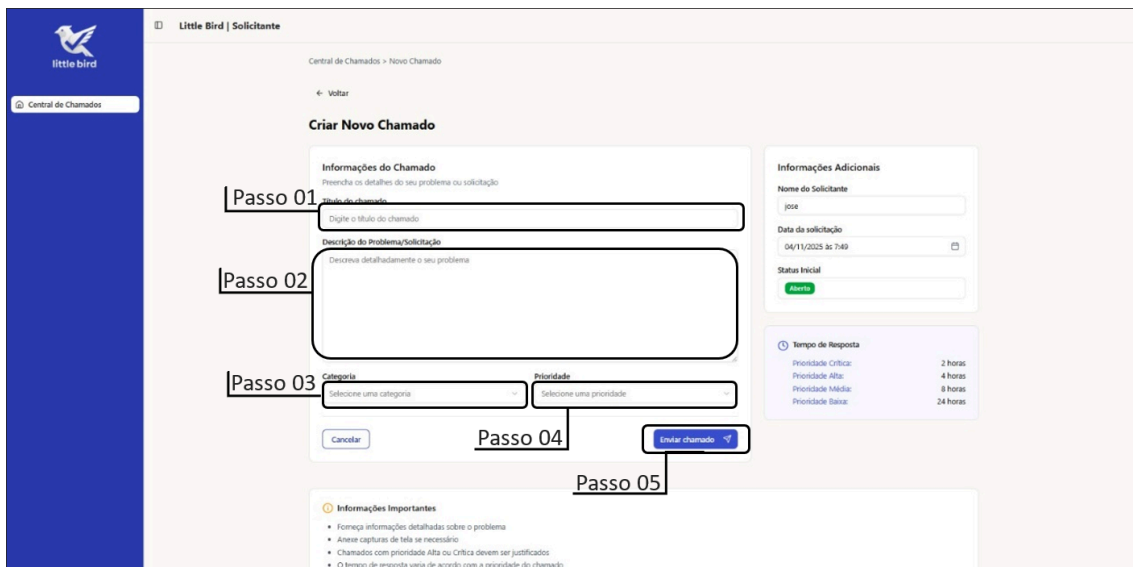
Imagem 5 - Dashboard.



Fonte: própria (Sistema Web).

2. Na próxima tela, preencha o chamado: informe brevemente o problema no título (Passo 1), detalhe-o no campo de descrição (Passo 2), selecione a categoria (Passo 3), defina a prioridade (Passo 4) e clique em “Enviar Chamado” (Passo 5) (Imagem 6).

Imagem 6 - Tela Criar Novo Chamado



**Little Bird | Solicitante**

Central de Chamados > Novo Chamado

[Voltar](#)

### Criar Novo Chamado

**Informações do Chamado**

Preencha os detalhes do seu problema ou solicitação

**Título do chamado**

Digite o título do chamado

**Descrição do Problema/Solicitação**

Descreva detalhadamente o seu problema

**Informações Adicionais**

Nome do Solicitante

jose

Data da solicitação

04/11/2025 às 7:49

Status Inicial

Aberto

**Tempo de Resposta**

Prioridade Crítica:	2 horas
Prioridade Alta:	4 horas
Prioridade Média:	8 horas
Prioridade Baixa:	24 horas

**Categoria**

Selecione uma categoria

**Prioridade**

Selecione uma prioridade

[Cancelar](#)

[Enviar chamado](#)

**Passo 01**

**Passo 02**

**Passo 03**

**Passo 04**

**Passo 05**

**Informações Importantes**

- Forneça informações detalhadas sobre o problema
- Anexe capturas de tela se necessário
- Chamados com prioridade Alta ou Crítica devem ser justificados
- O tempo de resposta varia de acordo com a prioridade do chamado

Fonte: própria (Sistema Web).

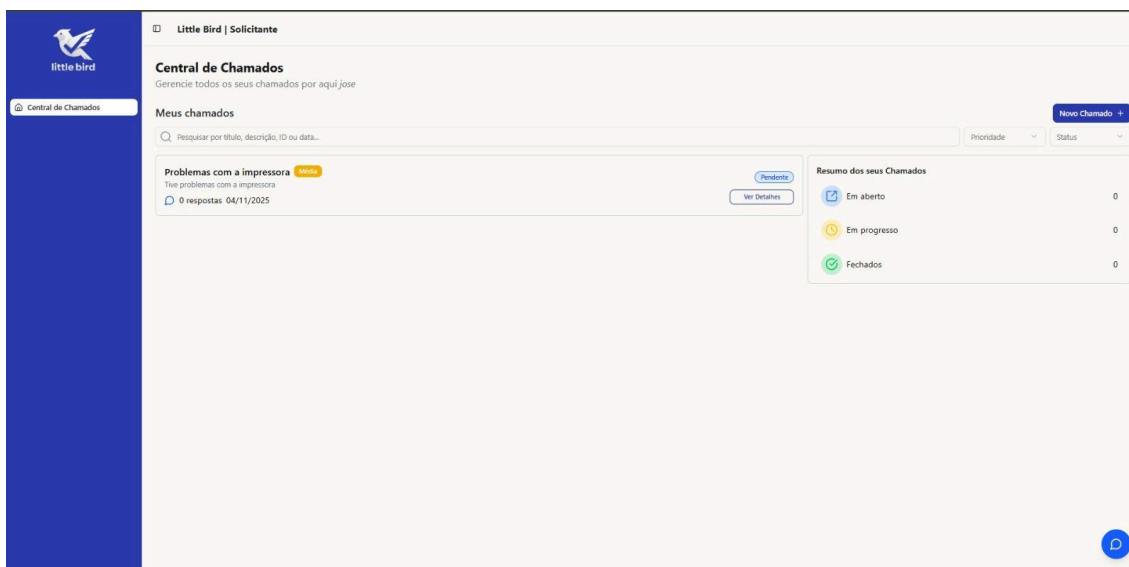
Pronto! Você conseguiu criar um chamado com sucesso.

## 6.2 Como entrar em contato com a Inteligência Artificial sablá:

Público-alvo: Atendentes, Solicitantes;

1. Em qualquer uma das telas que você acessar, você verá um balão azul com o ícone de chat no canto inferior direito, clique nele (Imagem 7).

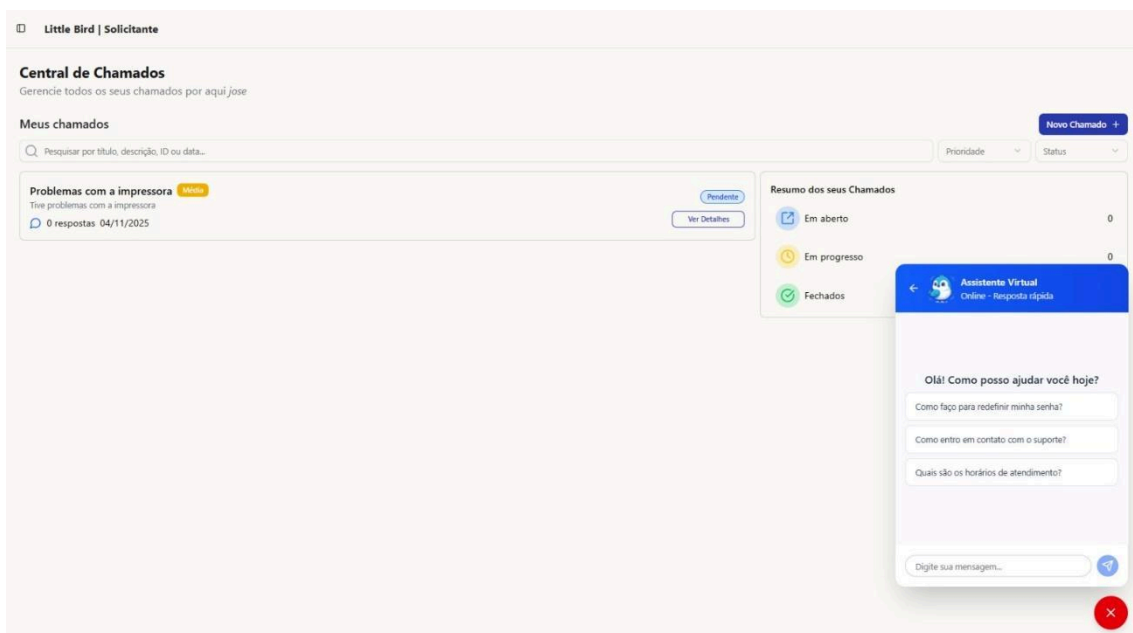
Imagem 7 - Conversa com o Chat Bot.



Fonte: própria (Sistema Web).

2. Ao clicar, será inicializada a conversa com a nossa sablÁ, problemas mais simples poderão ser resolvidos por ela, facilitando suas resoluções em tempo real, selecione uma das opções que aparecem, se nenhuma representar o seu problema, descreva na conversa as suas dificuldades (Passo 01) (Imagem 8).

Imagem 8 - Interface SablÁ.



Fonte: própria(Sistema Web).

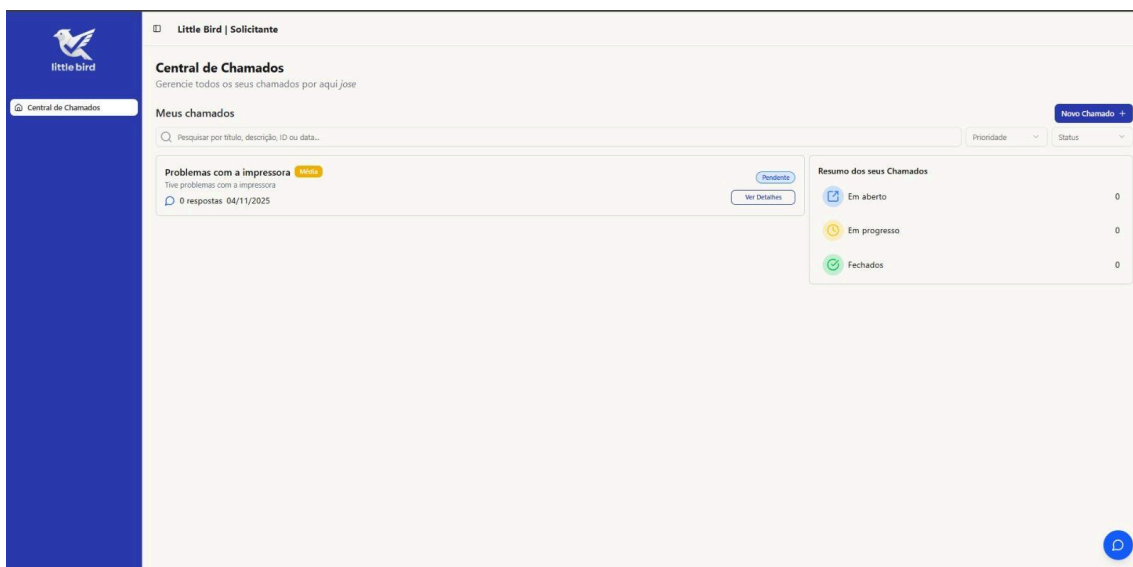


## 6.3 Como atender os chamados

Público-alvo: Atendentes;

1. No menu inicial, será possível ver todos os chamados em aberto, escolha os chamados de acordo com sua prioridade (leve em consideração o SLA) e seu conhecimento do tema e responda eles clicando no campo “ver detalhes” (Imagem 9).

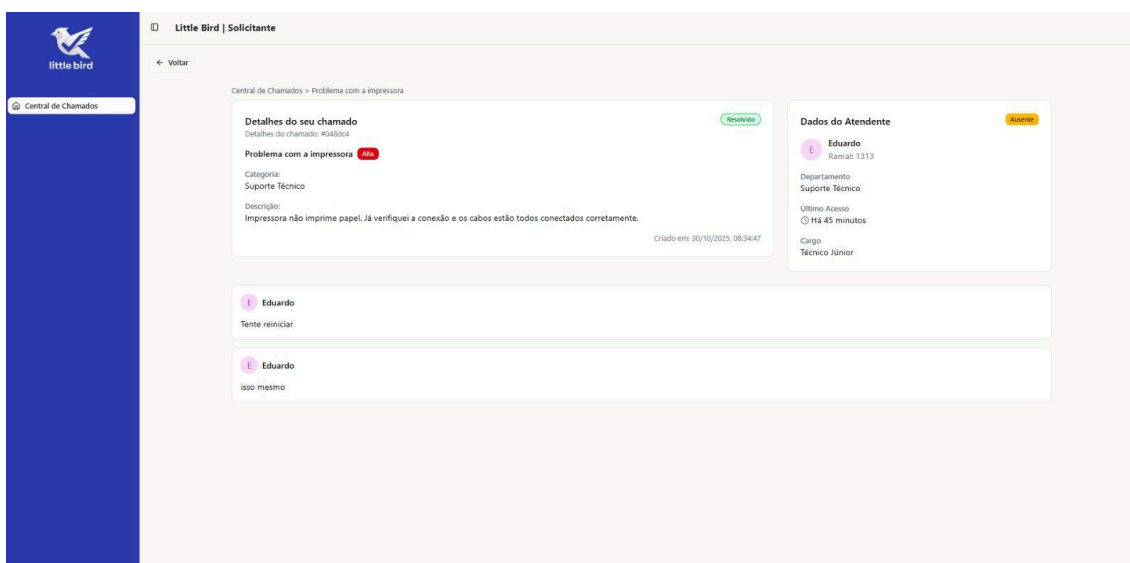
Imagem 9 - Dashboard Atendente.



Fonte: própria (Sistema Web).

2. Ao clicar nele, você conseguirá visualizar todos os detalhes do chamado que foram abertos pela área solicitante, responda o que for necessário, e quando concluído encerre o atendimento, sinalizando o chamado como “concluído” (Imagem 10).

Imagem 10 - Detalhamento de Chamados.

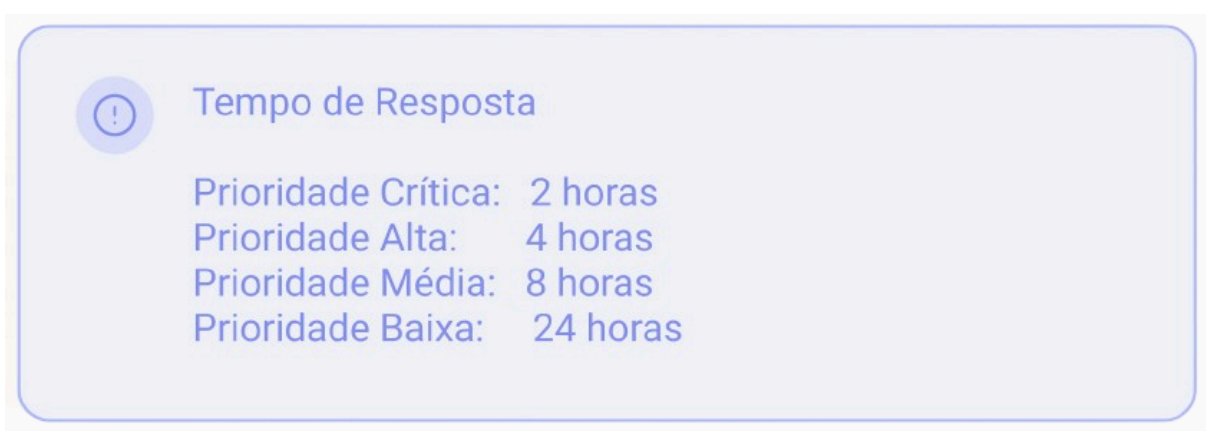


Fonte: própria (Sistema Web).

### **Atenção!**

O prazo do Acordo de Nível de Serviço (SLA) deve ser rigorosamente cumprido. Em caso de descumprimento, é necessário incluir uma justificativa no e-mail emitido pelo setor de Não Conformidades. Para situações recorrentes do mesmo problema, será exigida a criação de um plano de ação (Imagem 11).

Imagem 11 - Tempo de Resposta.



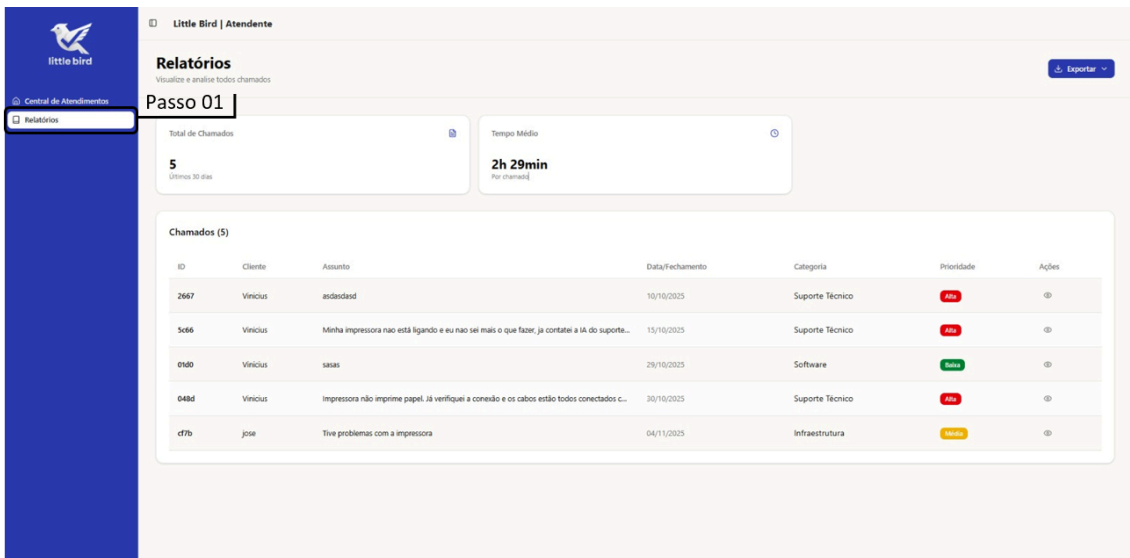
Fonte: própria (Sistema Web).

### 6.4 Como realizar a emissão de relatórios

Público-alvo: Administrador.

- 1. Logo na aba inicial, você deverá acessar a aba “Relatórios” na guia à esquerda (Passo 1) (Imagem 12);

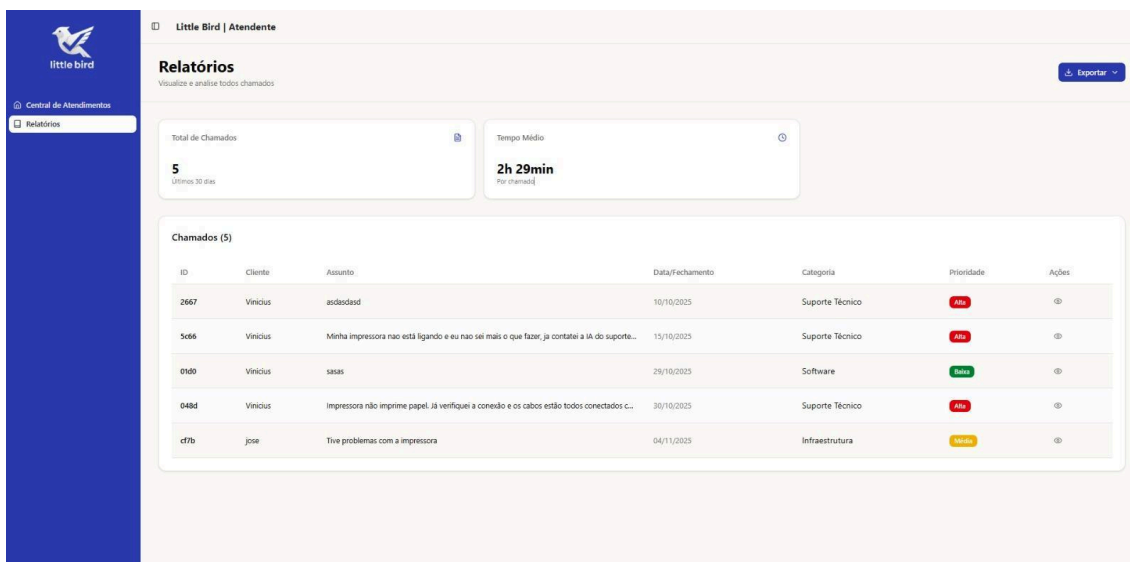
Imagem 12 - Dashboard Administrador.



Fonte: própria (Sistema Web).

2. Depois disso você terá acesso a todos os registros de chamados, caso você queira um relatório simplificado, basta selecionar exportar (Passo 1) e o tipo de arquivo que você vai querer (Passo 2) (Imagem 13).

Imagem 13 - Relatórios.



Fonte: própria (Sistema Web).

Imagem 14 - Botão Exportar.



Fonte: própria (Sistema Web).

Pronto! Agora você já está instruído com relação à ferramenta de extração de relatórios.

## **7 FAQ DE SUPORTE DE T.I**

### **7.1 Meu computador não está ligando. O que devo fazer ?**

**Problema:** O computador não responde ao pressionar o botão de energia, sem sinais de luz ou som.

#### **Soluções:**

- Verifique se o cabo de energia está bem conectado à tomada e ao computador;
- Teste a tomada com outro dispositivo para confirmar que está funcionando;
- Se for um notebook, conecte o carregador e verifique se a bateria está carregada;
- Pressione e segure o botão de energia por 10 segundos para reiniciar o hardware;
- Se o problema persistir, abra um chamado com o suporte de TI informando os detalhes.

### **7.2 Minha internet está lenta ou não conecta. Como resolver?**

**Problema:** A conexão Wi-Fi ou cabeada está instável, lenta ou não funciona.

#### **Soluções:**

- Reinicie o roteador/modem desconectando-o da energia por 30 segundos;
- Verifique se outros dispositivos na rede enfrentam o mesmo problema;
- Conecte-se via cabo Ethernet, se possível, para testar a estabilidade;
- Certifique-se de que não há atualizações ou downloads em segundo plano consumindo banda;
- Contate o suporte de TI se o problema continuar, informando a velocidade atual e testes realizados.

### **7.3 Não consigo acessar meu e-mail corporativo. O que está errado?**

**Problema:** Falha ao fazer login no e-mail da empresa ou mensagens que não carregam.

#### **Soluções:**

- Confirme se o nome de usuário e a senha estão corretos (observe letras maiúsculas/minúsculas);
- Tente acessar o e-mail por outro navegador ou dispositivo para descartar problemas locais;
- Verifique se há notificações de bloqueio de conta ou expiração de senha;
- Limpe o cache do navegador ou use o modo anônimo;
- Abra um chamado com o suporte de TI, especificando a mensagem de erro exibida.

### **7.4 Meu software trava ou fecha inesperadamente. Como corrigir?**

**Problema:** Um aplicativo corporativo apresenta falhas, congela ou fecha sozinho.

#### **Soluções:**

- Feche o aplicativo pelo Gerenciador de Tarefas (Ctrl+Shift+Esc) e reinicie-o;
- Verifique se há atualizações pendentes para o software ou sistema operacional;
- Reinicie o computador para liberar recursos do sistema;
- Confirme se o problema ocorre em outros computadores ou apenas no seu;
- Contate o suporte de TI, informando o nome do software, versão e comportamento exato.

### **7.5 Esqueci minha senha de login no sistema corporativo. Como recupero?**

**Problema:** Não consigo acessar o sistema devido a senha esquecida ou expirada.



### **Soluções:**

- Use a opção “Esqueceu sua senha?” na tela de login para redefinir a senha, se disponível.
- Verifique seu e-mail corporativo ou número de telefone cadastrado para receber o link de recuperação.
- Se não houver opção de recuperação automática, contate o suporte de TI imediatamente.

### **7.6 Meu computador está muito lento. Como melhorar o desempenho?**

**Problema:** O computador demora para abrir programas ou responder a comandos.

### **Soluções:**

- Feche aplicativos desnecessários e verifique o uso de memória no Gerenciador de Tarefas;
- Reinicie o computador para limpar processos em segundo plano;
- Certifique-se de que o disco rígido não está quase cheio (libere espaço, se necessário);
- Execute uma verificação de vírus/malware com o antivírus corporativo;
- Abra um chamado com o suporte de TI para análise de hardware ou otimização do sistema.

### **7.7 Não consigo imprimir documentos na impressora da rede. O que fazer?**

**Problema:** A impressora não responde ou exibe erros ao tentar imprimir.

### **Soluções:**

- Confirme se a impressora está ligada e conectada à rede (verifique luzes indicadoras);
- Verifique se a impressora está selecionada como padrão no seu computador;

- Reinicie a impressora e o computador para reestabelecer a conexão;
- Teste imprimir de outro computador para identificar se o problema é local;
- Contate o suporte de TI, informando o modelo da impressora e a mensagem de erro;

## **7.8 Recebi uma mensagem de erro sobre espaço em disco insuficiente. Como resolver?**

**Problema:** Mensagem indica que o disco rígido está cheio, impedindo salvar arquivos.

### **Soluções:**

- Exclua arquivos desnecessários, como downloads antigos ou arquivos temporários;
- Esvazie a lixeira do sistema para liberar espaço;
- Use a ferramenta de limpeza de disco do sistema operacional (ex.: “Limpeza de Disco” no Windows);
- Transfira arquivos grandes para um armazenamento em nuvem corporativo, se permitido;
- Solicite suporte de TI para análise e possíveis upgrades de armazenamento.

**Nota:** Para qualquer problema não resolvido, abra um chamado no sistema de suporte de TI, informando o título do problema, descrição detalhada, categoria, prioridade e capturas de tela, se aplicável. Siga as diretrizes de SLA para prazos de atendimento.

## 8 IMPLEMENTAÇÃO DE ESTRATÉGIAS DE CONFORMIDADE COM A LGPD NA LITTLE BIRD

A Lei Geral de Proteção de Dados Pessoais (LGPD), instituída pela Lei nº 13.709/2018, é a legislação brasileira que regula o tratamento de dados pessoais, ou seja, qualquer operação realizada com informações que identifiquem ou possam identificar uma pessoa física, como nome, CPF, e-mail, telefone, entre outros.

Para estar em conformidade com a Lei Geral de Proteção de Dados Pessoais (LGPD – Lei nº 13.709/2018), a Little Bird passou a adotar políticas e práticas que visam garantir a privacidade, segurança e transparência no tratamento de dados pessoais inseridos nos chamados criados no portal.

A LGPD será aplicada a todos os dados pessoais tratados no sistema, incluindo informações de identificação, contato e registros de comunicação dos usuários. Dessa forma, a Little Bird reforça o compromisso com a proteção e o uso ético dessas informações.

Para assegurar a conformidade com a legislação, estão sendo implementadas as seguintes estratégias e medidas:

- **Política de Privacidade e Termos de Uso** claros, comunicando como os dados são coletados, armazenados e utilizados;
- **Controle de acesso e perfis de usuário** para restringir a visualização de dados sensíveis apenas a pessoas autorizadas;
- **Anonimização** de informações pessoais quando possível, especialmente em relatórios e análises;
- **Registro e rastreabilidade** das operações realizadas dentro do sistema (logs de acesso e alterações);
- **Treinamento contínuo** da equipe de suporte técnico sobre boas práticas de proteção de dados;
- **Armazenamento seguro** com criptografia e políticas de retenção de dados que limitem o tempo de guarda ao estritamente necessário;

Essas ações reforçam o compromisso da Little Bird em manter um ambiente seguro, transparente e em conformidade com as normas de proteção de dados vigentes.

## 9 CONCLUSÃO

O desenvolvimento do Sistema Integrado para Gestão de Chamados e Suporte Técnico baseado em Inteligência Artificial demonstrou a importância crescente de soluções tecnológicas capazes de otimizar processos internos e responder de forma eficaz às demandas corporativas. Ao longo do projeto, foi possível identificar problemas recorrentes em empresas de médio porte, tais como retrabalhos, ausência de padronização e dificuldade no acompanhamento do fluxo de chamados. A proposta apresentada surgiu justamente como uma resposta a esses desafios, oferecendo uma solução moderna, escalável e alinhada às necessidades atuais do mercado.

A arquitetura proposta evidencia o papel central da automação e da inteligência artificial na melhoria do atendimento técnico. A implementação de um assistente virtual capaz de realizar triagens automáticas, sugerir soluções e encaminhar solicitações representa um avanço significativo na digitalização do suporte interno. Ao integrar funcionalidades como categorização dinâmica, análise de dados e relatórios gerenciais, o sistema não apenas organiza o fluxo de chamados, mas também contribui para uma tomada de decisão mais assertiva e baseada em evidências.

Além disso, a modelagem das entidades, os diagramas UML e a definição clara dos perfis de usuários reforçam a consistência do projeto e demonstram a atenção dedicada à estruturação lógica e funcional do sistema. O uso de interfaces responsivas e intuitivas garante uma experiência fluida, ampliando a acessibilidade e promovendo maior engajamento dos colaboradores que utilizam o sistema. Esses elementos combinados consolidam a proposta como um instrumento robusto para modernização do ambiente corporativo.

A preocupação com a conformidade à LGPD também se mostrou essencial no desenvolvimento do projeto. A implementação de medidas de segurança, controle de acesso, anonimização de dados e políticas transparentes destaca o compromisso com a proteção das informações tratadas dentro da plataforma. Essa abordagem não apenas atende aos requisitos legais, mas também reforça a credibilidade do sistema e sua aderência às boas práticas de governança de dados.

Em síntese, o projeto do Little Bird System 2.0 confirma seu potencial para transformar a gestão de chamados e o suporte técnico nas organizações.

A integração entre automação, inteligência artificial e gestão de processos demonstra que soluções inovadoras podem reduzir o tempo de resposta, aumentar a produtividade da equipe e elevar a qualidade do atendimento. Com uma base sólida e alinhada às demandas contemporâneas, o sistema se apresenta como um importante passo rumo à modernização e eficiência operacional dentro das empresas.

## REFERÊNCIAS

Atlassian. Atlassian. Disponível em:

<<https://www.atlassian.com/pt/itsm/service-desk-vs-help-desk>>. Acesso em: 10 de julho de 2025.

Axelos. Axelos. Disponível em: <<https://www.axelos.com/best-practice-solutions/itil>>.

Acesso em: 05 de agosto de 2025.

BRASIL. Gov.br. Disponível em:

<<https://www.gov.br/cidadania/pt-br/acesso-a-informacao/lgpd>>. Acesso em: 18 de outubro de 2025.

Fowler, Martin. Martin Fowler. Disponível em:

<<https://martinfowler.com/bliki/CQRS.html>>. Acesso em: 01 de novembro de 2025.

Gartner. Gartner. Disponível em:

<<https://www.gartner.com/en/information-technology/glossary/aiops>> Acesso em 30 de agosto de 2025.

Google Cloud. Google Cloud. Disponível em:

<<https://cloud.google.com/learn/what-is-conversational-ai>>. Acesso em 31 de outubro de 2025.

Martin, Robert C. The Clean Coder Blog. Disponível em:

<<https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html>>.

Acesso em: 22 de setembro de 2025.

Meta. React. Disponível em: <<https://pt-br.reactjs.org/>>. Acesso em: 02 de novembro de 2025.

Microsoft. Microsoft .NET. Disponível em:

<<https://dotnet.microsoft.com/pt-br/apps/aspnet>>. Acesso em: 25 de outubro de 2025.

Microsoft. Microsoft Learn. Disponível em:

<<https://learn.microsoft.com/pt-br/azure/architecture/patterns/cqrs>>. Acesso em: 19 de julho de 2025.

Microsoft. Microsoft Learn. Disponível em:

<<https://learn.microsoft.com/pt-br/dotnet/architecture/modern-web-apps-azure/clean-architecture>>. Acesso em: 14 de outubro de 2025.

Shadcn. Shadcn UI. Disponível em: <<https://ui.shadcn.com/>>. Acesso em: 28 de setembro de 2025.

Vite. Vite. Disponível em: <<https://vitejs.dev/>>. Acesso em: 11 de agosto de 2025.

Autoridade Nacional de Proteção de Dados (ANPD). ANPD. Disponível em:

<<https://www.gov.br/anpd/pt-br>>. Acesso em: 10 de setembro de 2025.

Fowler, Martin. Martin Fowler. Disponível em:

<<https://martinfowler.com/bliki/DomainDrivenDesign.html>>. Acesso em: 29 de outubro de 2025.

IBM. IBM. Disponível em: <<https://www.ibm.com/br-pt/topics/chatbots>>. Acesso em: 14 de agosto de 2025.

Microsoft. Microsoft Learn. Disponível em:

<<https://learn.microsoft.com/pt-br/ef/core/>>. Acesso em: 08 de julho de 2025.

Microsoft. TypeScript. Disponível em: <<https://www.typescriptlang.org/>>. Acesso em: 20 de agosto de 2025.

Object Management Group (OMG). Object Management Group (OMG). Disponível em: <<https://www.omg.org/spec/UML/>>. Acesso em: 05 de outubro de 2025.

ServiceNow. ServiceNow. Disponível em:

<<https://www.servicenow.com/br/products/itsm/what-is-sla.html>>. Acesso em: 15 de agosto de 2025.

Tailwind Labs. Tailwind CSS. Disponível em: <<https://tailwindcss.com/>>. Acesso em: 12 de agosto de 2025.



## **APÊNDICES**

### **APÊNDICE A - REPOSITÓRIO DO FRONT-END**

1. [https://github.com/viniciusavalentim/Pim\\_Help\\_Desk\\_Front](https://github.com/viniciusavalentim/Pim_Help_Desk_Front)

### **APÊNDICE B - REPOSITÓRIO DO BACK-END**

1. [https://github.com/viniciusavalentim/PIM\\_Help\\_Desk](https://github.com/viniciusavalentim/PIM_Help_Desk)

## ANEXOS

## ANEXO A - FICHA DE CONTROLE

**FICHA DE CONTROLE DO PIM**

Grupo Nº \_\_\_\_\_ Ano 2º Período: 4º Orientador: Gustavo

Tema: PEM Little Bird 2.0 sistema com integração TA

Alunos:

RA	Nome	E-mail	Curso	Visto do aluno
5898044	José Henrique Matos	josimatos@aluno.unip.br	ADS	
5592310	Allen Oliveira Ramos	Allenoliveira2002@gmail.com	ADS	Allen F.
RJ00888	Eduardo Sampaio Nênis	eduardo.nenis@aluno.unip.br	ADS	
6985837	Victor H. M. Costa	victorA.MARTelli.09@gmail	ADS	Victor H.
T558B16	Joyce Victoria Sena	joyce.sena@aluno.unip.br	ADS	Joyce S.
R09ADFO	Vícius Alves Dos Santos	VICIUS.ALVES@aluno.unip.br	ADS	Vícius V.

Registros:

[illegible]