



UNIVERSIDADE PAULISTA

CAUAN RAMOS DE LIMA

MATEUS CAETANO RIBEIRO

LUANN GALLEGOS DE SOUZA

SISTEMA INTELIGENTE DE SUPORTE DE TI:
AUTOMAÇÃO E OTIMIZAÇÃO COM INTELIGÊNCIA
ARTIFICIAL

SÃO JOSÉ DO RIO PRETO - SP

2025

CAUAN RAMOS DE LIMA
MATEUS CAETANO RIBEIRO
LUANN GALLEGOS DE SOUZA

Sistema Inteligente de Suporte de TI: Automação e Otimização com Inteligência Artificial

Trabalho de projeto integrado multidisciplinar para obtenção do título em graduação em Analise e Desenvolvimento de Sistema apresentado à Universidade Paulista – UNIP. Disciplina de Desenvolvimento de Projeto.

SÃO JOSÉ DO RIO PRETO - SP

2025

RESUMO

Este trabalho tem como objetivo desenvolver o HelpDesk, um sistema de chamados com integração de Inteligência Artificial (IA), fundamentado nos conhecimentos adquiridos no curso de Análise e Desenvolvimento de Sistemas. A disciplina de Engenharia de Software é essencial para a estruturação do sistema, pois define seus requisitos e assegura a qualidade em todas as etapas do desenvolvimento, desde a concepção até a entrega do produto final. A Análise Orientada a Objetos possibilita a modelagem de casos de uso, a criação de diagramas de classe e a organização eficiente do código, facilitando a manutenção e evolução do sistema. O uso de um Banco de Dados garante o armazenamento seguro e eficiente das informações, preservando sua integridade, confiabilidade e disponibilidade para os usuários. A disciplina de Programação Orientada a Objetos é fundamental para a construção do software, que utilizará a linguagem (C SHARP). Essa abordagem permite a criação de um código mais modular, reutilizável e escalável, garantindo maior flexibilidade para futuras atualizações e aprimoramentos. O Projeto de Interface com o Usuário viabiliza o desenvolvimento de interfaces intuitivas e acessíveis para web, mobile e desktop, proporcionando uma experiência agradável e eficiente para os usuários. Além disso, a disciplina de Economia de Mercado auxilia na análise de custos e na viabilidade financeira do projeto, permitindo um planejamento adequado dos investimentos necessários. Já a Gestão de Recursos Humanos define funções e responsabilidades dentro da equipe de desenvolvimento e suporte, assegurando a eficiência na operação e na manutenção contínua do sistema.

Palavras-chave: Sistema de chamados, Engenharia de Software, Análise Orientada a Objetos, Programação Orientada a Objetos, Banco de Dados, Economia de Mercado, Gestão de Recursos Humanos, Interface com o Usuário.

ABSTRACT

This work aims to develop Digisic, a call system with Artificial Intelligence (AI) integration, based on the knowledge acquired in the Systems Analysis and Development course. The Software Engineering discipline is essential for structuring the system, as it defines its requirements and ensures quality throughout all stages of development, from conception to the final product delivery. Object-Oriented Analysis enables the modeling of use cases, the creation of class diagrams, and the efficient organization of the code, facilitating system maintenance and evolution. The use of a Database ensures secure and efficient storage of information, preserving its integrity, reliability, and availability for users. The Object-Oriented Programming discipline is fundamental for software development, utilizing the (C SHARP) language. This approach allows for the creation of a modular, reusable, and scalable codebase, ensuring greater flexibility for future updates and improvements. User Interface Design enables the development of intuitive and accessible interfaces for web, mobile, and desktop, providing users with a pleasant and efficient experience. Furthermore, the Market Economics discipline assists in cost analysis and the financial viability of the project, allowing for adequate planning of the necessary investments. Human Resources Management defines roles and responsibilities within the development and support team, ensuring efficiency in the operation and continuous maintenance of the system.

Keywords: Call System, Software Engineering, Object-Oriented Analysis, Object-Oriented Programming, Database, Market Economics, Human Resources Management, User Interface.

SUMÁRIO

1	INTRODUÇÃO	7
1.1	Proposta do trabalho	7
1.2	Objetivos	8
1.2.1	Objetivo Geral	8
1.2.2	Objetivos específicos	8
1.3	Justificativa	8
1.4	Regras de Negócios	8
1.5	Glossário do sistema	9
1.6	Ciclo de vida do Software: Kanban	9
2	DESENVOLVIMENTO	11
2.1	Processos de Negócio	11
2.1.1	Processos Finalísticos	11
2.1.2	Processos de Apoio	11
2.1.3	Processos de Gestão	12
2.2	Requisitos e UML	12
2.2.1	Regras de Negócio (RN)	12
2.2.2	Requisitos de Usuário (RU)	12
2.2.3	Requisitos do Sistema (RS) com BDD	13
2.3	Diagrama de Domínio	14
2.3.1	Diagrama de Caso de Uso: Login e Autenticação	16
2.3.2	Diagrama de Caso de Uso: Registrar Chamado	17
2.3.3	Diagrama de Caso de Uso: Classificar e Encaminhar Chamado (IA)	19
2.3.4	Diagrama de Caso de Uso: Consultar Status de Chamado	20
2.3.5	Diagrama de Caso de Uso: Abrir e Encerrar Chamado	22
2.3.6	Diagrama de Caso de Uso: Gerenciar Usuários e Técnicos	23
2.3.7	Diagrama de Caso de Uso: Emitir Relatórios	24
2.3.8	Visão Geral dos Casos de Uso	26
2.4	Diagramas de sequência	27
2.4.1	Diagramas de Sequência: Login e Autenticação	27
2.4.2	Diagramas de Sequência: Registrar Chamado	28
2.4.3	Diagramas de Sequência: Consultar Status de Chamado	28
2.4.4	Diagramas de Sequência: Abrir e Encerrar Chamado	29
2.4.5	Diagramas de Sequência: Gerenciar Usuários e Técnicos	29
2.4.6	Diagramas de Sequência: Emitir Relatórios	30

2.5	Diagramas de Atividade	30
2.5.0.1	Login e Validação	30
2.5.0.2	Registrar Chamado	31
2.5.0.3	Consultar Status de Chamado	31
2.5.0.4	Abrir e Fechar Chamado	32
2.5.0.5	Gerenciar Usuários e Técnicos	32
2.5.0.6	Emitir Relatórios	33
2.5.1	Diagrama de Classes	33
2.5.2	Diagrama de Implantação	35
2.6	Requisitos Funcionais (RF)	36
2.7	Requisitos Não Funcionais (RNF)	37
2.8	Banco de Dados	37
2.8.1	Scripts do Banco de Dados	37
2.8.2	Scripts de Testes	39
2.8.3	Diagrama Entidade e Relacionamento	42
2.8.4	Dicionário de Dados: Tabela tb_admin	42
2.8.5	Dicionário de Dados: Tabela tb_usuario	42
2.8.6	Dicionário de Dados: Tabela tb_tecnico	43
2.8.7	Dicionário de Dados: Tabela Tickets	44
2.8.8	Dicionário de Dados — Tabela Mensagens	45
2.8.9	Dicionário de Dados — Tabela ChatFiles	46
2.8.10	Planilha de Teste do Sistema de Chamados	47
2.9	Interface Humano-Computador	49
2.9.1	Protótipo de Tela: Mobile	49
2.9.2	Protótipo de Tela: Web	52
2.9.3	Protótipo de Tela: Desktop	54
3	CONCLUSÃO	55
	REFERÊNCIAS	56

1 INTRODUÇÃO

1.1 Proposta do trabalho

O propósito deste trabalho é aprofundar os conceitos práticos e teóricos abordados nas disciplinas do curso de Análise e Desenvolvimento de Sistemas da Universidade Paulista de São José do Rio Preto, além de aplicar esses conceitos em um estudo de caso em um sistema de chamados para suporte técnico de TI. O principal objetivo é desenvolver um sistema de chamados com integração de inteligência artificial, o qual permite que a equipe de suporte técnico possa responder aos clientes de forma mais rápida e eficiente. Para alcançar esse fim, serão aplicados os conhecimentos adquiridos nas disciplinas de engenharia de software, análise orientada a objetos, banco de dados, interface com o usuário, economia de mercado e gestão de recursos humanos.

Na disciplina de engenharia de software, os alunos exploram os princípios e técnicas fundamentais para o desenvolvimento de sistemas, com foco na definição dos requisitos funcionais e não funcionais que garantem o correto funcionamento do software. Essa base oferece o suporte necessário para o planejamento e a organização deste projeto.

Em programação orientada a objetos, os conceitos de algoritmos e estruturas de dados formam o alicerce da construção do sistema, que será desenvolvido na linguagem C(SHARP) com o framework .NET.

No banco de dados, a estruturação e organização das informações do sistema se baseiam no uso do SQL Server, garantindo armazenamento seguro e eficiente.

Na disciplina de análise orientada a objetos, o uso da UML é aprofundado com foco nos diagramas que representam o funcionamento interno do software, facilitando sua compreensão e manutenção.

Na interface com o usuário, os conhecimentos adquiridos permitem criar protótipos de telas para desktop, dispositivos móveis e web, promovendo uma experiência mais fluida e intuitiva para o usuário.

Na gestão de recursos humanos, os conceitos ajudam a entender a importância do trabalho em equipe, da liderança e da comunicação eficiente durante o desenvolvimento do projeto, além de contribuir para o gerenciamento das demandas e do tempo da equipe.

1.2 Objetivos

1.2.1 Objetivo Geral

Este trabalho tem como objetivo aprofundar os conceitos aprendidos no curso de Análise e Desenvolvimento de Sistemas da Universidade Paulista de São José do Rio Preto, além de aplicar na prática os conhecimentos adquiridos ao longo do semestre. A finalidade é construir um sistema de suporte técnico inteligente.

1.2.2 Objetivos específicos

Aplicar levantamento de requisitos do sistema.

Desenvolver artefatos UML.

Criação de interfaces gráficas para usabilidade no sistema.

Utilização da inteligência artificial para boas práticas.

1.3 Justificativa

Este trabalho tem como objetivo melhorar o gerenciamento de chamados para o suporte técnico, ajudando no controle da entrada e saída de informações. Além disso, permite a aplicação e o aprofundamento dos conhecimentos adquiridos ao longo deste semestre no curso de Análise e Desenvolvimento de Sistemas.

1.4 Regras de Negócios

Para que a equipe compreenda adequadamente a solicitação, cada chamado deve conter informações detalhadas sobre o ocorrido, incluindo: assunto, explicação e tipo. O registro de chamados no sistema somente será permitido mediante o fornecimento completo dessas informações por parte do usuário.

A classificação dos chamados será realizada manualmente por um atendente. A prioridade (baixa, média, alta ou crítica) será definida pelo atendente responsável, com base nas informações disponibilizadas.

Após a finalização do chamado, o sistema deve registrar a solução adotada e disponibilizar uma opção para que o usuário possa fornecer feedback.

Os chamados devem ser atribuídos a uma equipe ou atendente responsável, sendo que o sistema somente permitirá o andamento do chamado caso ele esteja devidamente atribuído.

Técnicos e atendentes não possuem permissão para excluir chamados, sendo autorizados apenas a alterar seu status para “aberto”, “em andamento” ou “resolvido”.

Apenas usuários devidamente cadastrados no sistema poderão registrar chamados, garantindo, assim, a segurança, autenticidade e rastreabilidade dos registros.

1.5 Glossário do sistema

Abertura de Chamado: Processo inicial de registro de um problema ou solicitação no sistema de suporte técnico.

Chamado: Registro formal de um problema ou solicitação feita por um usuário ou cliente para o suporte de TI. Pode ser um incidente ou uma solicitação de serviço.

Incidente: Qualquer evento que interrompa o funcionamento normal de um serviço de TI, como falhas de hardware, software ou rede.

Solicitação de Serviço: Pedido feito por um usuário para obter um serviço específico, como instalação de software, criação de contas ou modificações em configurações de TI.

Status do Chamado: Representa a fase atual do chamado no processo de atendimento, como por exemplo status em andamento, aberto, resolvido.

Categoria: Tipo de problema ou serviço solicitado. Exemplos de categorias incluem Hardware, Software, Rede, Suporte ao Usuário, Infraestrutura, entre outros.

Técnico de Suporte: Profissional responsável por resolver os chamados registrados.

Resolução: Resolve o problema ou atende à solicitação do usuário.

Histórico do Chamado: Registro completo de todas as interações, soluções e atualizações realizadas durante o ciclo de vida de um chamado, desde sua abertura até o fechamento.

Escalonamento: Processo de encaminhar um chamado para um nível mais avançado de suporte, quando não é possível resolvê-lo no nível atual.

Notificação: Alerta enviado ao usuário ou técnico para informar sobre atualizações, status ou resolução do chamado.

1.6 Ciclo de vida do Software: Kanban

Foi optado pela utilização da metodologia ágil Kanban para o desenvolvimento do sistema de suporte de TI.

Visualização clara do fluxo de trabalho: o kanban permite visualizar todas as etapas do processo de desenvolvimento deste sistema, como novo, em andamento, teste, concluído, o

que facilita a organização e permite saber exatamente o que está sendo realizado.

Identificar gargalos: Facilita a percepção de onde os processos estão travando ou acumulando tarefas, permitindo ações corretivas rápidas.

Transparência para todos os envolvidos: Toda a equipe pode acompanhar o andamento dos chamados, promovendo confiança e clareza.

Adaptação e melhoria contínua: O Kanban é flexível e pode ser ajustado constantemente, conforme as necessidades da equipe e do sistema.

Priorização de chamados: o kanban permite que os chamados mais urgentes sejam atendidos primeiro.

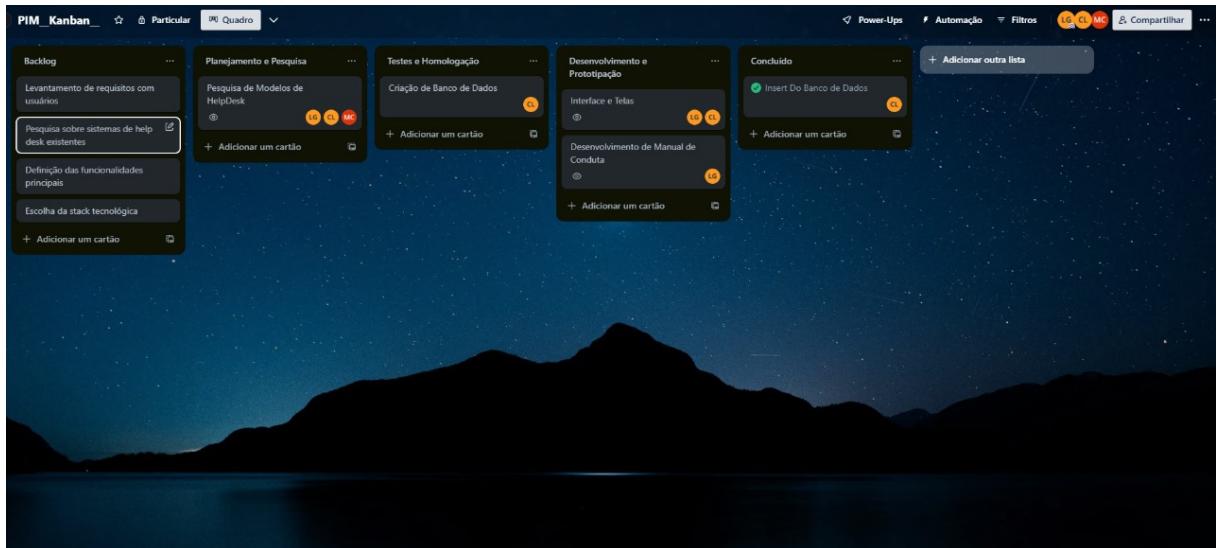
Feedback rápido e constante: Como o processo de desenvolvimento e suporte é transparente, os membros da equipe recebem feedback contínuo sobre o status dos chamados, facilitando a tomada de decisões rápidas e informadas.

Facilidade de gestão e controle: A utilização de Kanban proporciona uma gestão visual e simples, permitindo que líderes de equipe e gestores monitorem a performance do time e o progresso das tarefas em tempo real.

Ajuste contínuo e responsivo: O processo contínuo de ajustes no fluxo de trabalho permite que o Kanban se encaixe perfeitamente no ritmo e nas necessidades da equipe, tornando-se uma ferramenta ágil de fácil adaptação.

Redução do tempo de ciclo: Ao visualizar o progresso e identificar gargalos, o Kanban ajuda a reduzir o tempo de ciclo de cada tarefa, permitindo que os chamados sejam resolvidos mais rapidamente e com mais qualidade.

Como é mostrado abaixo.



2 DESENVOLVIMENTO

2.1 Processos de Negócio

2.1.1 Processos Finalísticos

- **Criação de Chamados (Usuário):** O usuário acessa o sistema e registra um novo chamado, informando o problema encontrado. Assim que o chamado é criado, o sistema abre automaticamente um chat entre o usuário e o técnico responsável, permitindo a comunicação em tempo real durante o atendimento.
- **Encaminhamento Automático (IA):** A inteligência artificial analisa o conteúdo do chamado e direciona automaticamente para o técnico mais adequado, considerando o tipo de problema e o histórico de atendimentos.
- **Atendimento e Resolução (Técnico):** O técnico recebe o chamado, interage com o usuário pelo chat, analisa as informações e aplica as ações necessárias para solucionar o problema. Quando o técnico finaliza o atendimento, o sistema registra a conclusão e envia uma notificação ao usuário informando que o chamado foi encerrado.
- **Acompanhamento de Status (Usuário):** O usuário pode acompanhar o andamento dos seus chamados em tempo real, verificando atualizações no chat e no painel de status, como “aberto”, “em andamento” e “finalizado”.

2.1.2 Processos de Apoio

- **Base de Conhecimento:** O sistema mantém um conjunto de categorias pré-definidas de problemas. Esse filtro inicial auxilia a inteligência artificial a identificar o tipo de ocorrência durante o registro do chamado, facilitando a classificação e o encaminhamento automático para o técnico mais adequado.
- **Suporte e Manutenção do Sistema:** A equipe de TI realiza manutenções preventivas, correções e atualizações para garantir que o sistema funcione de forma estável e segura.
- **Controle de Acesso:** O sistema gerencia o login e as permissões de cada usuário, garantindo segurança nas informações e respeitando os diferentes níveis de acesso (usuário, técnico e administrador).

2.1.3 Processos de Gestão

- **Gerenciamento de Usuários e Técnicos (Administrador):** O administrador pode cadastrar, editar ou excluir contas de usuários e técnicos, controlando perfis.
- **Geração de Relatórios (Administrador):** O administrador pode emitir relatórios de desempenho, como número de chamados abertos por mês, status de todos os chamados e relatório dos atendimentos.
- **Melhoria Contínua:** Com base nos relatórios e feedback dos usuários, a equipe gestora analisa resultados e define ajustes para aprimorar a eficiência e a qualidade dos atendimentos.

2.2 Requisitos e UML

2.2.1 Regras de Negócio (RN)

- **RN01:** Todo chamado deve conter informações obrigatórias: título, descrição.
- **RN02:** A IA classifica automaticamente o chamado e sugere o técnico mais adequado.
- **RN03:** Somente usuários autenticados podem abrir chamados pelo Mobile ou pelo Web.
- **RN04:** Técnicos podem apenas alterar o status do chamado (aberto, em andamento, finalizado), mas não podem excluir chamados.
- **RN05:** Apenas administradores podem gerar relatórios e gerenciar usuários/técnicos.
- **RN06:** Todos os dados devem estar em conformidade com a LGPD, garantindo segurança, privacidade e rastreabilidade.

2.2.2 Requisitos de Usuário (RU)

- **RU01:** O usuário deve conseguir abrir um chamado pelo Mobile.
- **RU02:** O usuário deve conseguir abrir um chamado pelo Web.
- **RU03:** O usuário deve conseguir acompanhar o status de seus chamados.
- **RU04:** O técnico deve conseguir receber, atender e encerrar chamados.
- **RU05:** O administrador deve conseguir cadastrar, editar e remover usuários e técnicos.
- **RU06:** O administrador deve poder gerar relatórios gerenciais.

2.2.3 Requisitos do Sistema (RS) com BDD

- **RS01:** O sistema deve validar credenciais de acesso.

Cenário: Login válido

Dado que a persona esteja na tela de login.

Quando inserir usuário e senha corretos.

Então o sistema deve liberar o acesso.

Cenário: Login inválido

Dado que a persona esteja na tela de login.

Quando inserir login e senha incorretos.

Então o sistema deve exibir mensagem de erro.

- **RS02:** O sistema deve armazenar e classificar automaticamente os chamados via IA.

Cenário: Registro de chamado com sucesso

Dado que o usuário esteja autenticado.

Quando registrar um chamado.

Então o sistema deve salvar.

E a IA deve classificá-lo.

Cenário: Falha na IA

Dado que um chamado foi registrado.

Quando a IA não conseguir classificar.

Então o chamado não deve ser encaminhado.

- **RS03:** O sistema deve encaminhar o chamado ao técnico mais adequado.

Cenário: Encaminhamento automático

Dado que a IA classificou o chamado.

Quando o sistema identificar o técnico.

Então o chamado deve ser atribuído ao técnico responsável.

Cenário: Falha no encaminhamento

Dado que a IA não conseguiu classificar.

Quando o chamado for criado.

Então ela não deve encaminhar.

- **RS04:** O sistema deve permitir consulta de status em tempo real.

Cenário: Consulta com resultados

Dado que o usuário tenha chamados registrados.

Quando acessar 'Meus Chamados'.

Então o sistema deve exibir a lista de chamados com seus status.

Cenário: Nenhum chamado encontrado

Dado que o usuário não tenha chamados.

Quando acessar 'Meus Chamados'.

Então o sistema deve informar que não há registros.

- **RS05:** O sistema deve emitir relatórios de gráficos de desempenho, volume e PDF de chamados registrados.

Cenário: Relatório gerado

Dado que o administrador esteja na tela de gerenciamento.

Quando solicitar um relatório.

Então o sistema deve gerar e exibir o relatório.

Cenário: Falha na geração

Dado que o administrador esteja na tela de gerenciamento.

Quando o sistema não encontrar os dados.

Então o sistema deve informar indisponibilidade.

- **RS06:** O sistema deve notificar os usuários sobre encerramento de chamados.

Cenário: Notificação de encerramento

Dado que o chamado foi atualizado.

Quando o técnico finalizar o chamado.

Então o sistema deve enviar uma notificação ao usuário.

2.3 Diagrama de Domínio

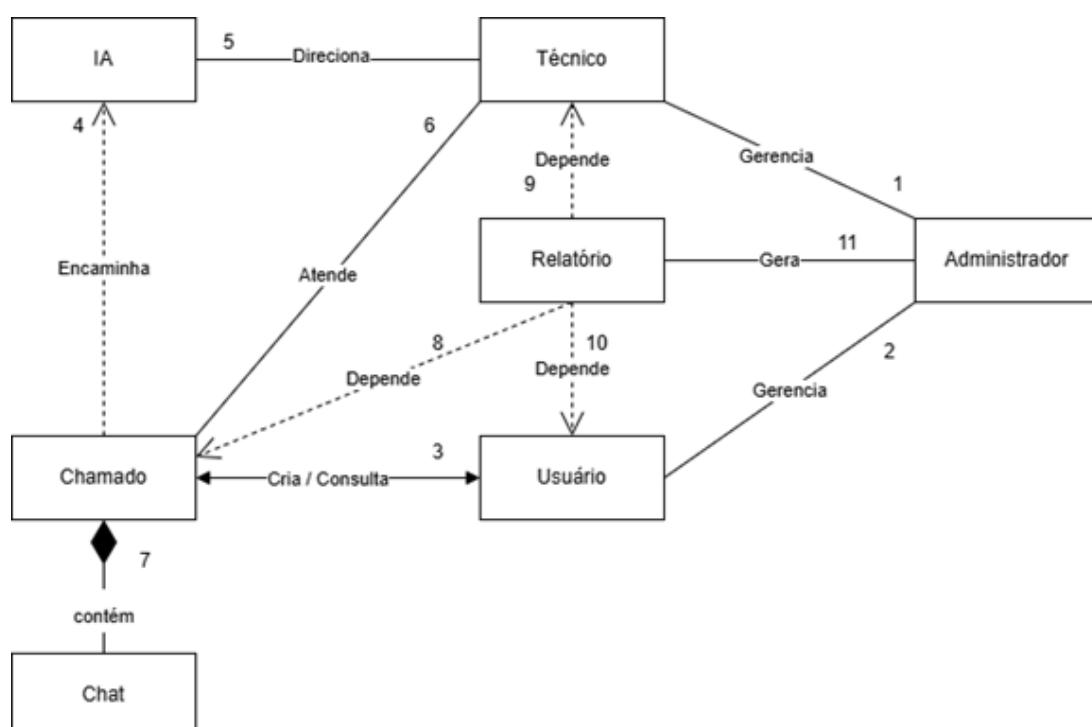


Figura 1 – Diagrama de Domínio do Sistema

1. **Administrador gerencia Técnico:** O administrador cadastra, edita ou remove técnicos no sistema, controlando quem pode atender chamados.
2. **Administrador gerencia Usuário:** O administrador cadastra, edita ou remove usuários, definindo permissões e acessos.
3. **Usuário cria / consulta Chamado:** O usuário abre um novo chamado ou acompanha o andamento de chamados anteriores via web ou mobile.
4. **Chamado é encaminhado para IA:** Assim que criado, o sistema envia o chamado para a IA analisar e classificar o tipo de problema.
5. **IA direciona o Chamado para o Técnico:** A IA escolhe automaticamente o técnico mais adequado com base no tipo de problema.
6. **Técnico atende o Chamado:** O técnico recebe o chamado, conversa com o usuário pelo chat e realiza o atendimento até resolver o problema.
7. **Chamado contém Chat:** Cada chamado tem um chat próprio, usado para comunicação entre o técnico e o usuário durante o atendimento.
8. **Chamado fornece dados para Relatório:** Os dados dos chamados resolvidos alimentam os relatórios do sistema.
9. **Relatório depende do Técnico:** Os relatórios mostram o desempenho de cada técnico (quantidade e tempo de atendimento, por exemplo).
10. **Relatório depende do Usuário:** Os relatórios também trazem informações sobre os usuários, como quantidade de chamados gerados no total.
11. **Administrador gera Relatórios:** O administrador emite relatórios de desempenho e controle, analisando o funcionamento geral do sistema.

2.3.1 Diagrama de Caso de Uso: Login e Autenticação

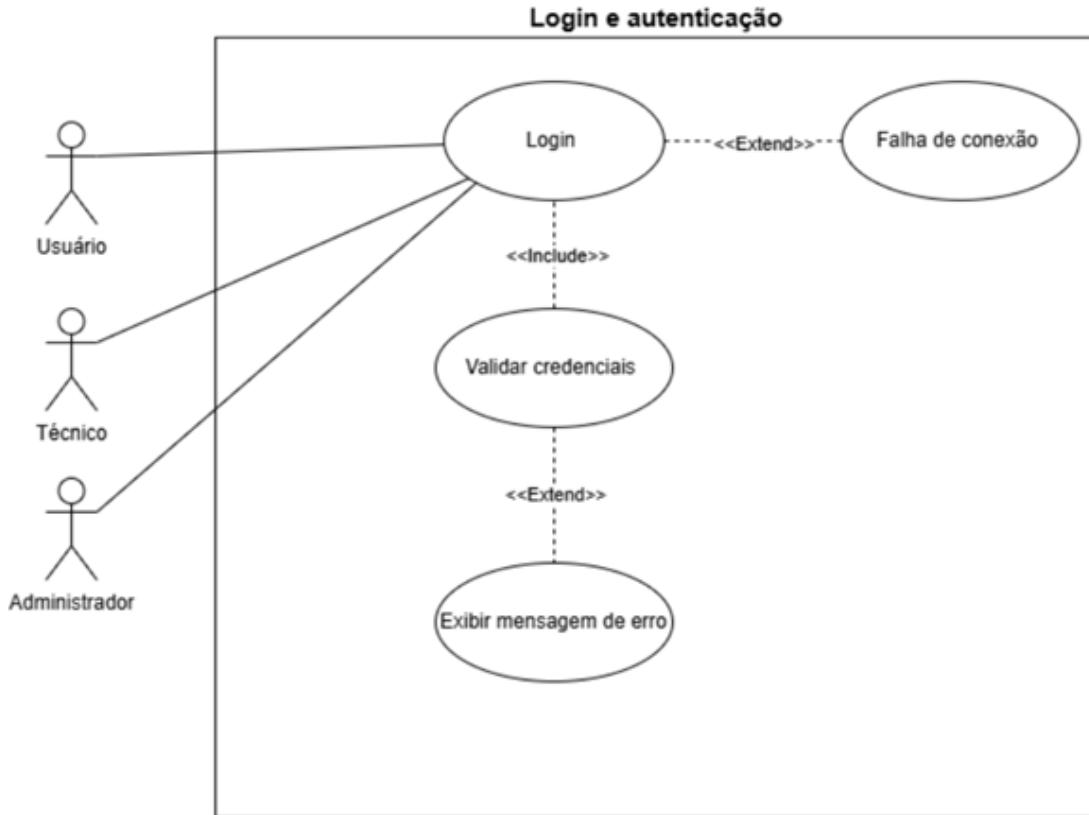


Figura 2 – Diagrama de Caso de Uso: Login e Autenticação

Atores: Usuário, Técnico e Administrador

Descrição: Permite que o usuário realize login de forma segura no sistema, informando suas credenciais (usuário e senha). O sistema valida os dados e concede acesso conforme o perfil de permissão.

Fluxo Principal:

1. O usuário acessa a tela de login.
2. O sistema solicita usuário e senha.
3. O usuário informa os dados.
4. O sistema valida as credenciais.
5. O sistema libera o acesso ao painel correspondente (usuário, técnico ou administrador).

Fluxo Alternativo: Dados Inválidos

1. Caso os dados estejam incorretos, o sistema exibe mensagem de erro.
2. O usuário pode tentar novamente.

Fluxo de Exceção: Falha de Conexão com o Servidor de Autenticação

1. O sistema exibe mensagem de erro.
2. O sistema orienta o usuário a tentar mais tarde.

Pré-condição: O usuário precisa ter uma conta válida cadastrada no sistema.

Pós-condição: Usuário autenticado com sucesso e acesso liberado conforme seu perfil.

2.3.2 Diagrama de Caso de Uso: Registrar Chamado

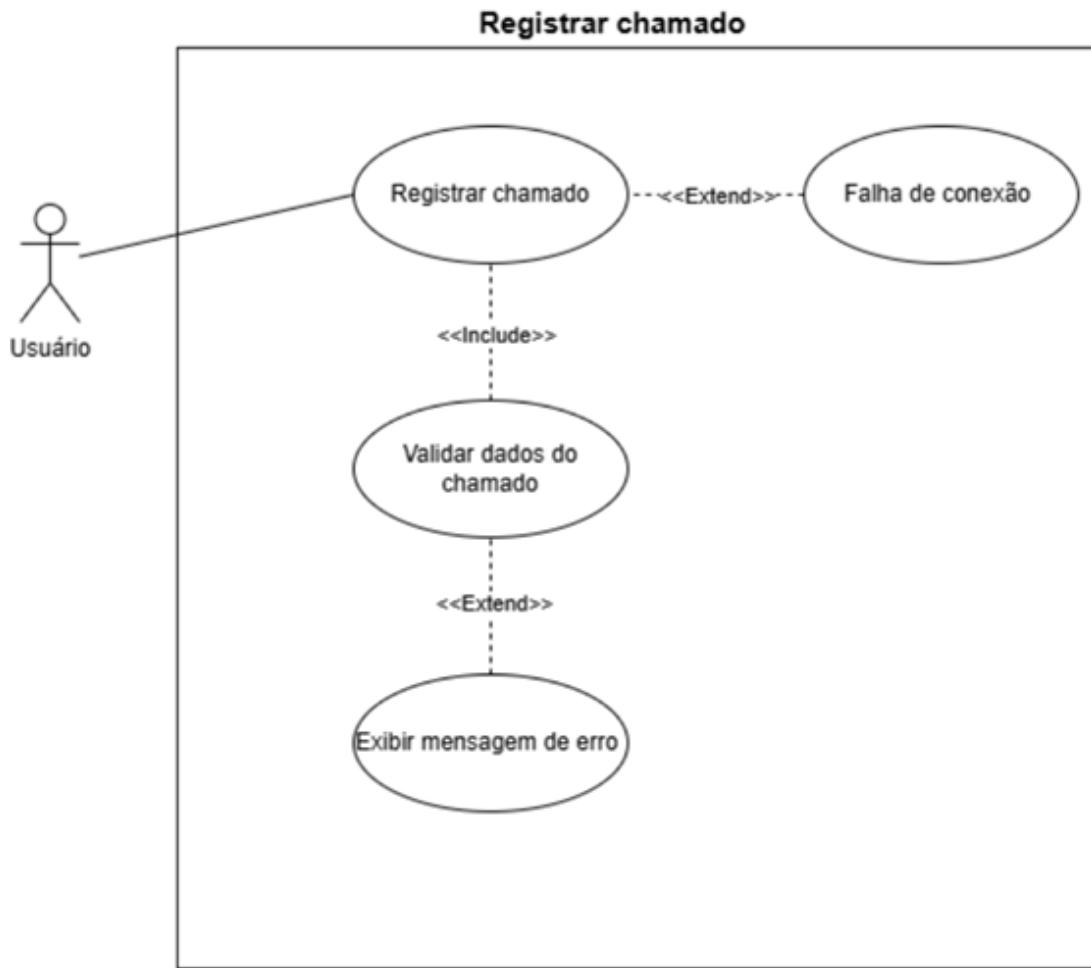


Figura 3 – Diagrama de Caso de Uso: Registrar Chamado

Atores: Usuário

Descrição: Permite que o usuário registre um chamado técnico através do aplicativo mobile ou via web, descrevendo o problema e anexando informações relevantes.

Fluxo Principal:

1. O usuário acessa o aplicativo mobile ou web e seleciona "Enviar chamado".
2. O sistema solicita o preenchimento de informações sobre o problema.
3. O usuário preenche os dados e anexa imagens (se necessário).
4. O usuário confirma o envio.
5. O sistema registra o chamado e encaminha para a IA para classificação e encaminhamento ao técnico.

Fluxo Alternativo: Dados Incompletos

1. O sistema alerta sobre campos obrigatórios não preenchidos.
2. O usuário preenche as informações faltantes.

Fluxo de Exceção: Falha no Envio do Chamado

1. O sistema exibe mensagem de erro.
2. O sistema orienta o usuário a tentar novamente.

Pré-condição: Usuário autenticado e logado no aplicativo mobile ou via web.

Pós-condição: Chamado registrado e encaminhado para classificação pela IA.

2.3.3 Diagrama de Caso de Uso: Classificar e Encaminhar Chamado (IA)

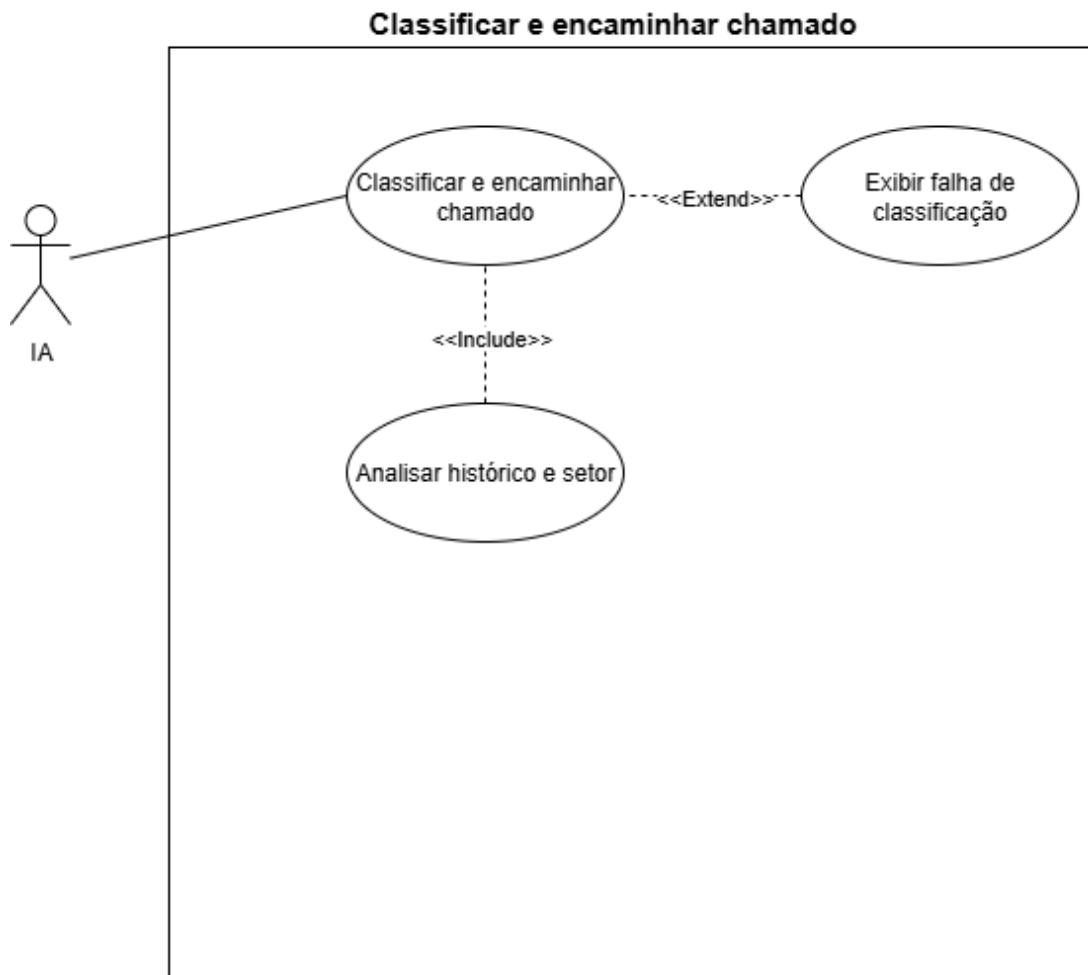


Figura 4 – Diagrama de Caso de Uso: Classificar e Encaminhar Chamado (IA)

Atores: IA

Descrição: Após o chamado ser registrado, a IA analisa o conteúdo do chamado e, com base em histórico e complexidade, encaminha automaticamente ao técnico mais adequado.

Fluxo Principal:

1. Chamado é registrado no sistema.
2. A IA analisa o texto e metadados.
3. A IA identifica o técnico mais capacitado.
4. O sistema encaminha automaticamente o chamado ao técnico selecionado.

Fluxo Alternativo: Dúvida na Classificação

1. Caso a IA não encontre correspondência clara, o chamado não é encaminhado.

Fluxo de Exceção: Falha na Comunicação com o Serviço de IA

1. O sistema salva o chamado.
2. O sistema aguarda o serviço de IA restabelecer para processar novamente.

Pré-condição: Chamado registrado no sistema.

Pós-condição: Chamado atribuído ao técnico responsável.

2.3.4 Diagrama de Caso de Uso: Consultar Status de Chamado

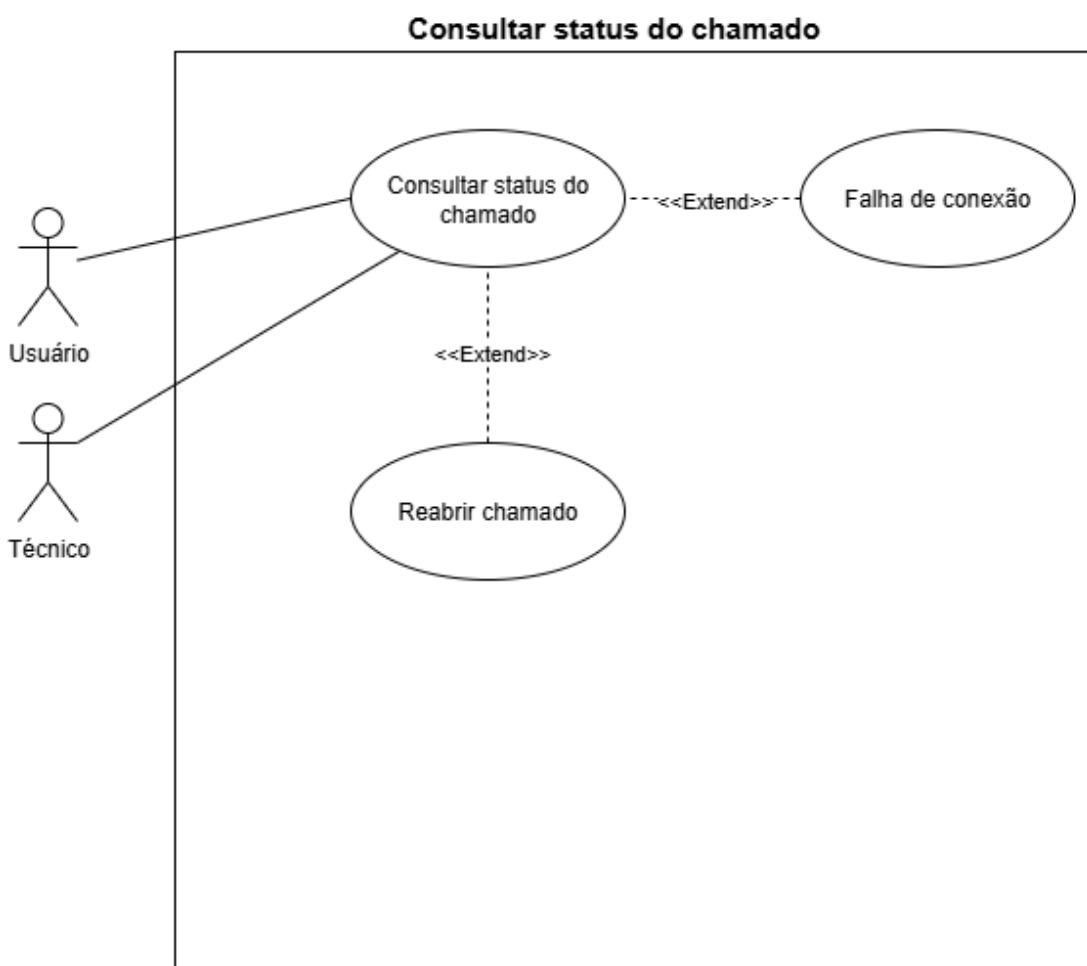


Figura 5 – Diagrama de Caso de Uso: Consultar Status de Chamado

Atores: Usuário e Técnico

Descrição: Permite ao usuário consultar, via aplicativo mobile ou web, o status de seus chamados registrados, e ao técnico visualizar os status dos chamados via web.

Fluxo Principal:

1. O usuário acessa a área "Meus Chamados".
2. O técnico acessa o menu com os chamados.
3. O sistema exibe a lista de chamados com seus respectivos status.
4. O usuário e o técnico podem visualizar detalhes de cada chamado.

Fluxo Alternativo: Nenhum Chamado Encontrado

1. O sistema exibe mensagem indicando que não existem chamados.

Fluxo de Exceção: Falha de Conexão

1. O sistema exibe mensagem de erro.

Pré-condição: Usuário e técnico autenticados.

Pós-condição: O usuário e o técnico visualizam as informações de status dos chamados.

2.3.5 Diagrama de Caso de Uso: Abrir e Encerrar Chamado

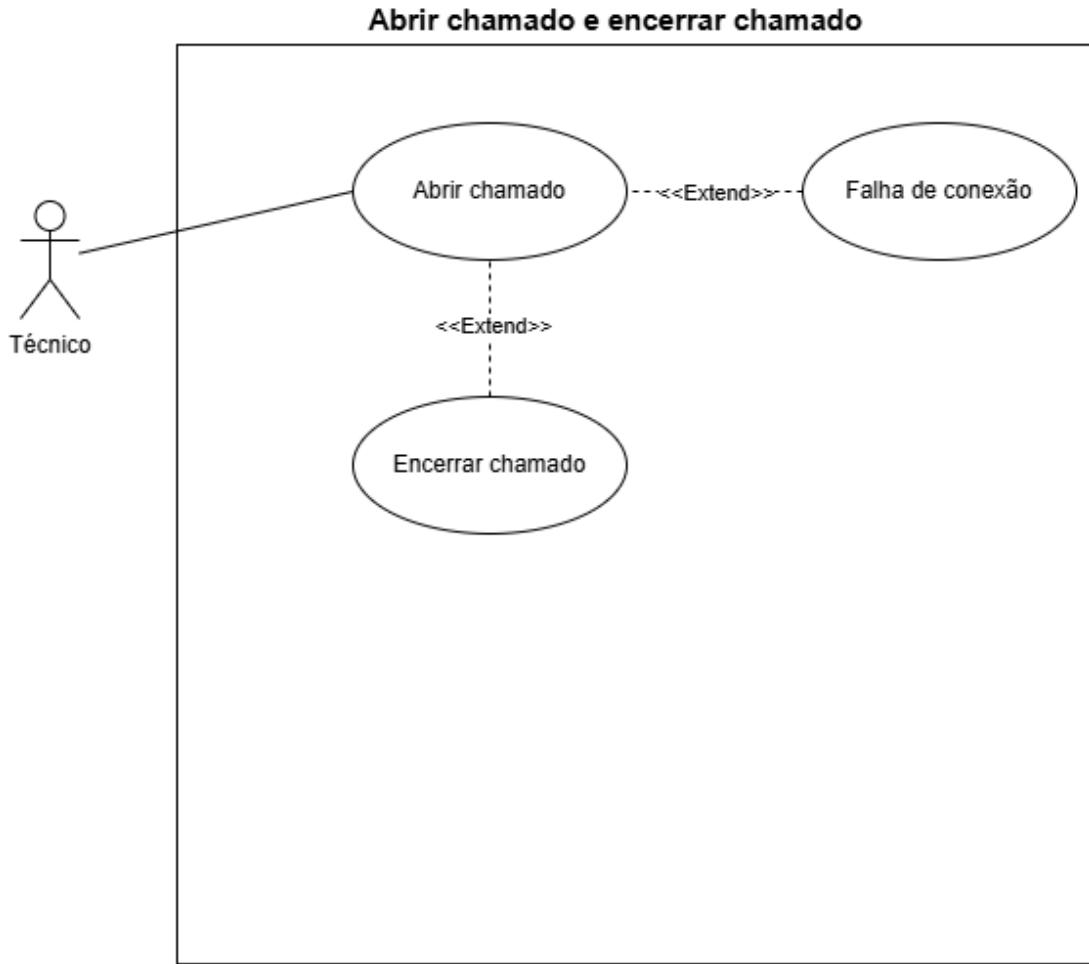


Figura 6 – Diagrama de Caso de Uso: Abrir e Encerrar Chamado

Atores: Técnico

Descrição: Permite que o técnico abra e encerre um chamado, registrando observações e finalizando o atendimento.

Fluxo Principal:

1. O técnico acessa a lista de chamados atribuídos.
2. O técnico seleciona o chamado.
3. O técnico insere observações finais e conclui o encerramento.
4. O sistema registra o encerramento e notifica o usuário.

Fluxo Alternativo: Chamado não resolvido

1. O chamado permanece na lista de chamados em andamento.

Fluxo de Exceção: Falha ao encerrar o chamado

- O sistema exibe mensagem de erro.

Pré-condição: Chamado atribuído ao técnico.

Pós-condição: Chamado encerrado e usuário notificado.

2.3.6 Diagrama de Caso de Uso: Gerenciar Usuários e Técnicos

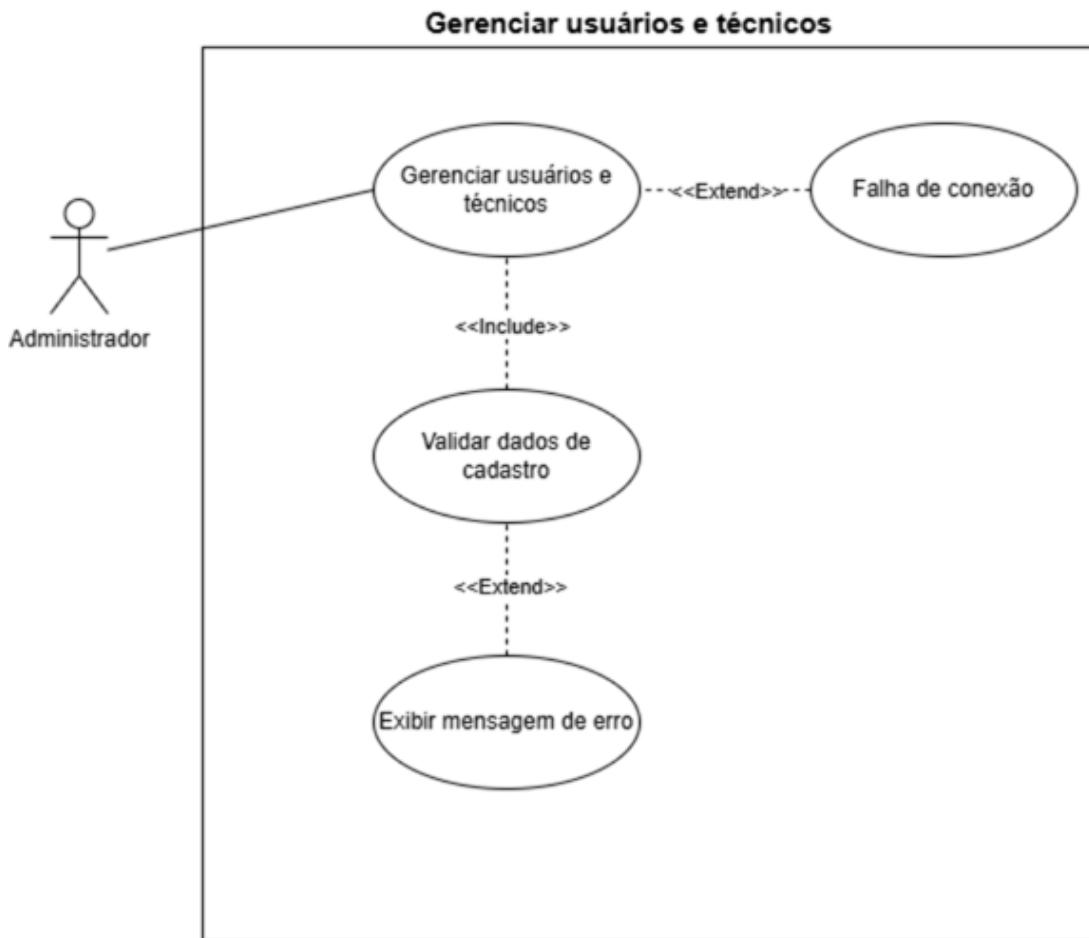


Figura 7 – Diagrama de Caso de Uso: Gerenciar Usuários e Técnicos

Atores: Administrador

Descrição: Permite ao administrador cadastrar, editar e excluir contas de usuários e técnicos, mantendo o controle de acessos e perfis no sistema.

Fluxo Principal:

- O administrador acessa o módulo de gestão.
- O administrador cria, edita ou remove cadastros de usuários e técnicos.

3. O sistema salva as alterações no banco de dados.

Fluxo Alternativo: Dados incompletos

1. O sistema alerta sobre campos obrigatórios não preenchidos.

Fluxo de Exceção: Falha na conexão com o banco de dados

1. O sistema exibe mensagem de erro.

Pré-condição: Administrador autenticado no sistema.

Pós-condição: Usuários e técnicos cadastrados e atualizados corretamente no sistema.

2.3.7 Diagrama de Caso de Uso: Emitir Relatórios

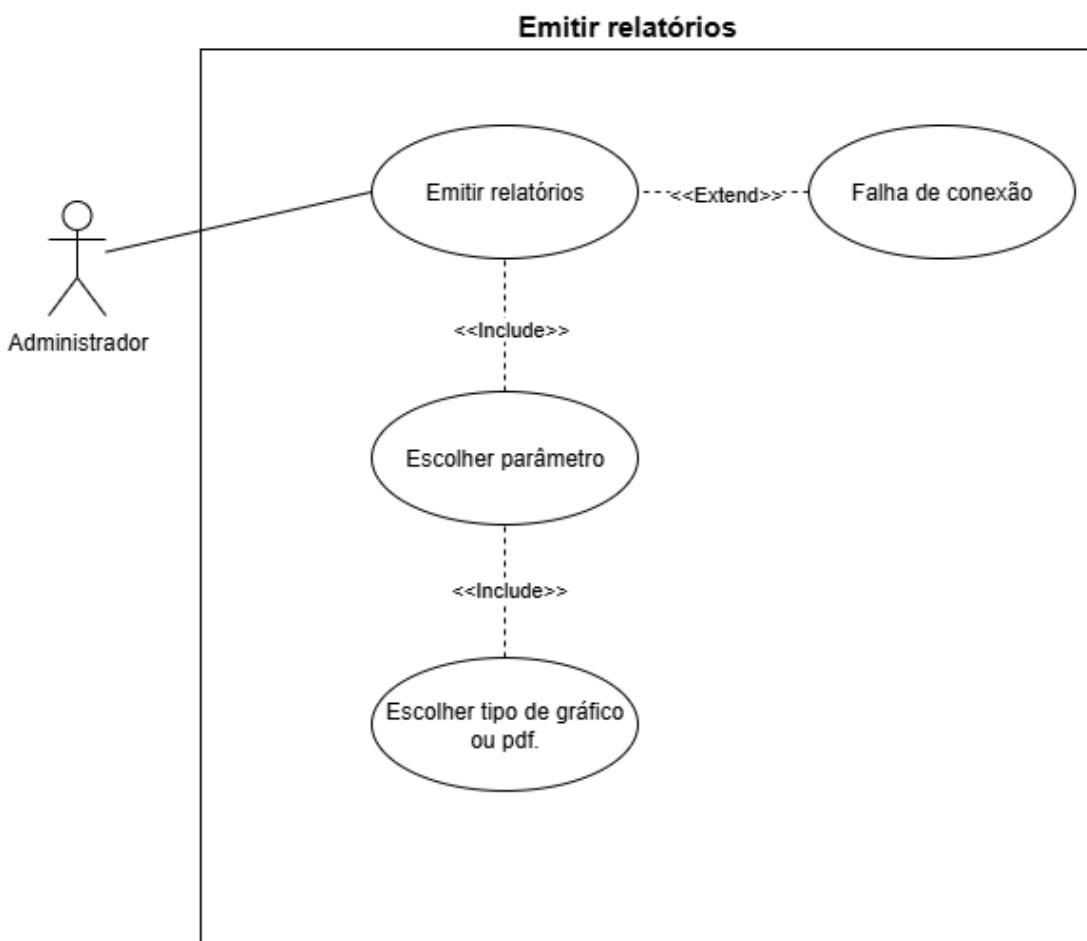


Figura 8 – Diagrama de Caso de Uso: Emitir Relatórios

Atores: Administrador

Descrição: Permite ao administrador emitir relatórios de gráficos de desempenho, volume e PDF de chamados registrados, possibilitando análise gerencial do sistema.

Fluxo Principal:

1. O administrador acessa o módulo de relatórios.
2. O sistema oferece filtros e critérios de consulta.
3. O administrador seleciona as opções desejadas.
4. O sistema gera e exibe o gráfico ou relatório.

Fluxo Alternativo: Nenhum dado disponível

1. O sistema informa que não há dados para o período selecionado.

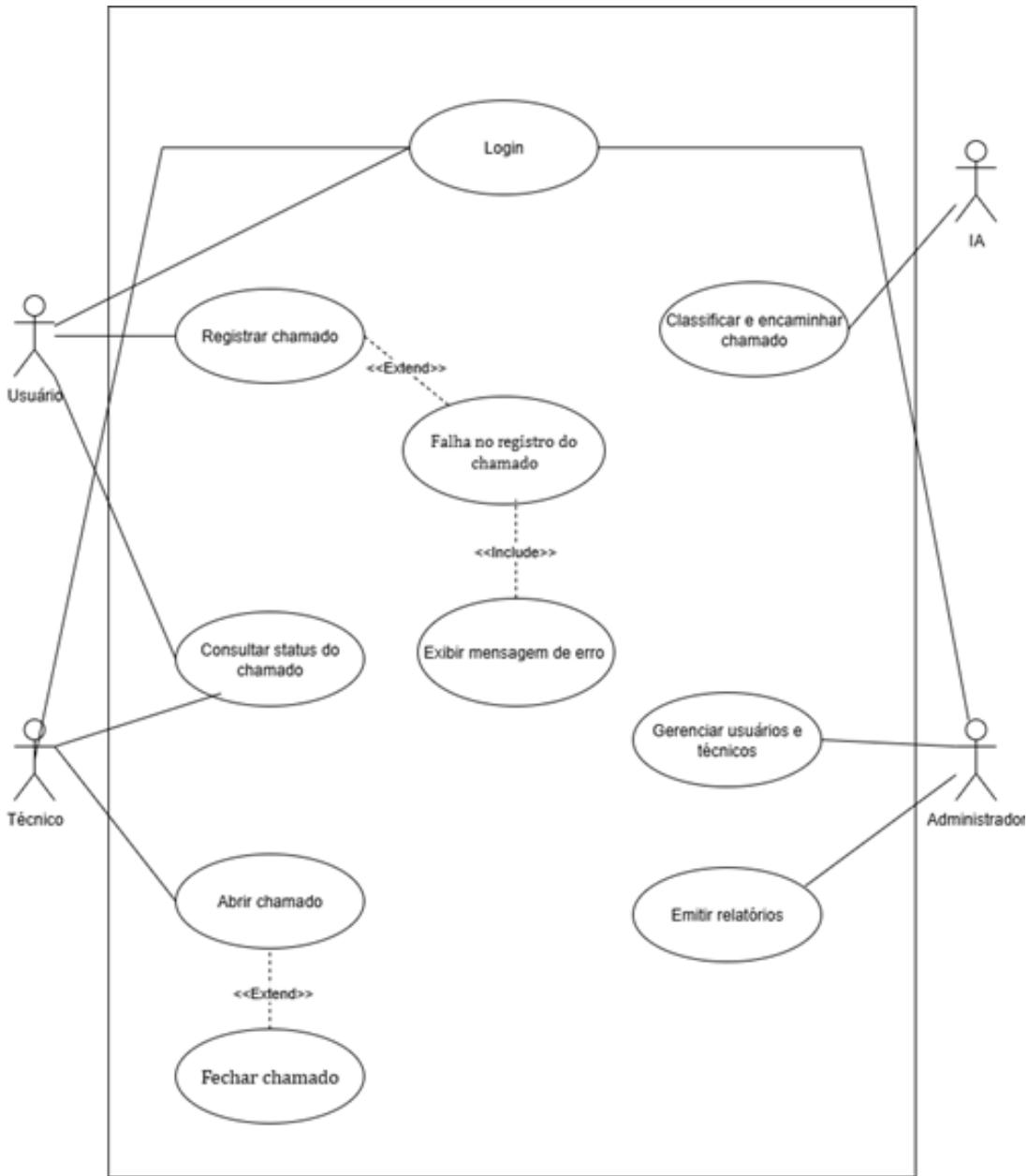
Fluxo de Exceção: Falha ao gerar relatório

1. O sistema exibe mensagem de erro e orienta nova tentativa.

Pré-condição: Administrador autenticado.

Pós-condição: Relatório ou gráfico gerado e disponibilizado corretamente.

2.3.8 Visão Geral dos Casos de Uso



Atores: Usuário, IA, Técnico e Administrador

Descrição: Permite que o usuário registre um chamado via aplicativo mobile ou web. O sistema utiliza Inteligência Artificial para analisar o conteúdo do chamado e encaminhá-lo automaticamente ao técnico mais adequado, considerando histórico, setor e complexidade do problema. O técnico realiza o atendimento e encerra o chamado. O usuário é notificado sobre a finalização do chamado. O administrador pode gerenciar os usuários e técnicos e gerar relatórios.

Fluxo Principal:

1. O usuário acessa o aplicativo ou plataforma web e realiza o login.

2. O usuário registra o chamado.
3. O sistema armazena o chamado.
4. A IA analisa o conteúdo e classifica o chamado.
5. A IA encaminha o chamado ao técnico mais adequado.
6. O técnico resolve o problema e encerra o chamado.
7. O sistema notifica o usuário sobre a conclusão.

Fluxo Alternativo:

1. Dúvida na classificação: caso a IA não encontre correspondência clara, o chamado não é encaminhado.

Fluxo de Exceção:

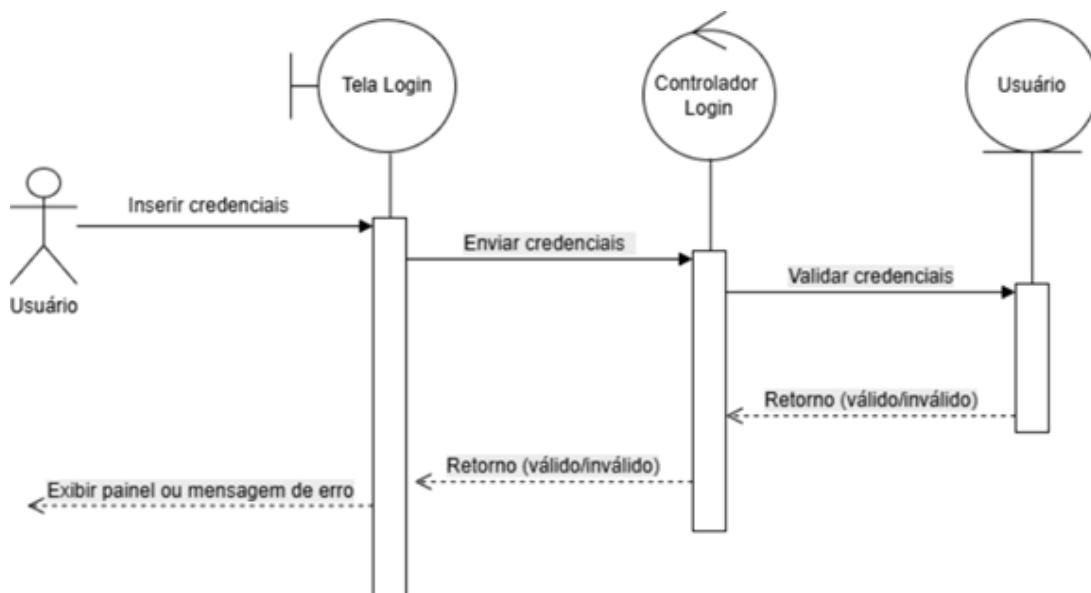
1. Falha no Registro do Chamado: o sistema exibe uma mensagem de erro e orienta o usuário a tentar novamente.

Pré-condição: Usuário autenticado no sistema.

Pós-condição: Chamado classificado, atribuído ao técnico responsável e finalizado com notificação ao usuário.

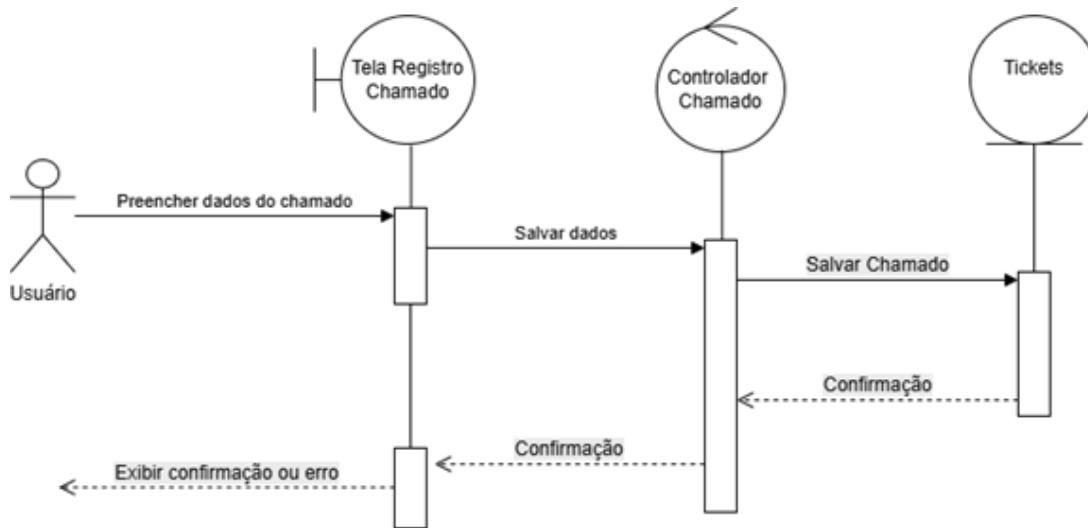
2.4 Diagramas de sequência

2.4.1 Diagramas de Sequência: Login e Autenticação



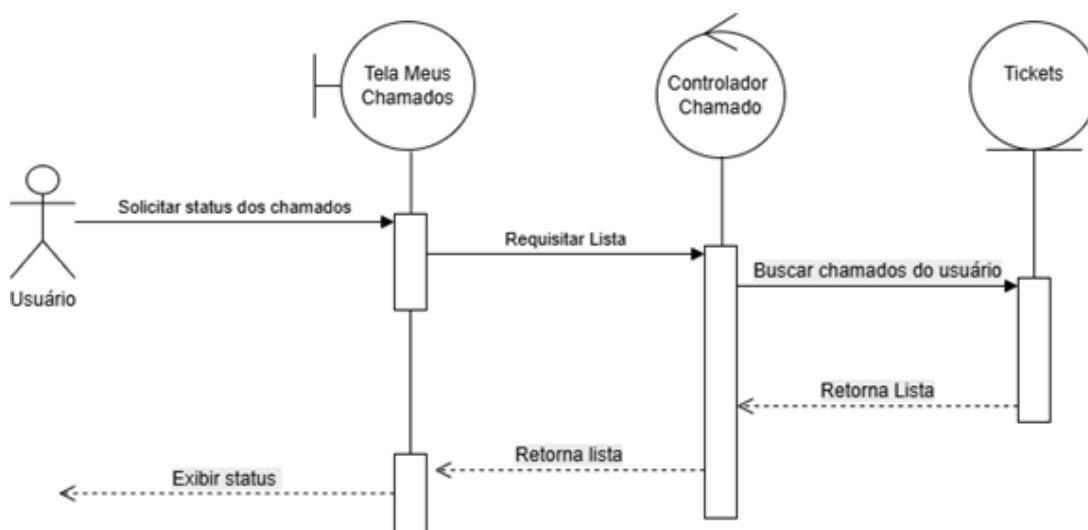
Descrição: Este diagrama representa o fluxo de autenticação de um usuário. A interface de login coleta as credenciais e as envia para o controlador, que consulta a entidade de usuários para validar os dados. Em caso de sucesso, o sistema concede acesso conforme o perfil; caso contrário, exibe mensagens de erro ou falha de conexão.

2.4.2 Diagramas de Sequência: Registrar Chamado



Descrição: Este diagrama ilustra o processo de registro de um chamado técnico via aplicativo mobile ou web. O usuário preenche os dados, e o controlador valida os dados, encaminhando-os à entidade de tickets. O sistema armazena o chamado e aciona a IA para classificação automática.

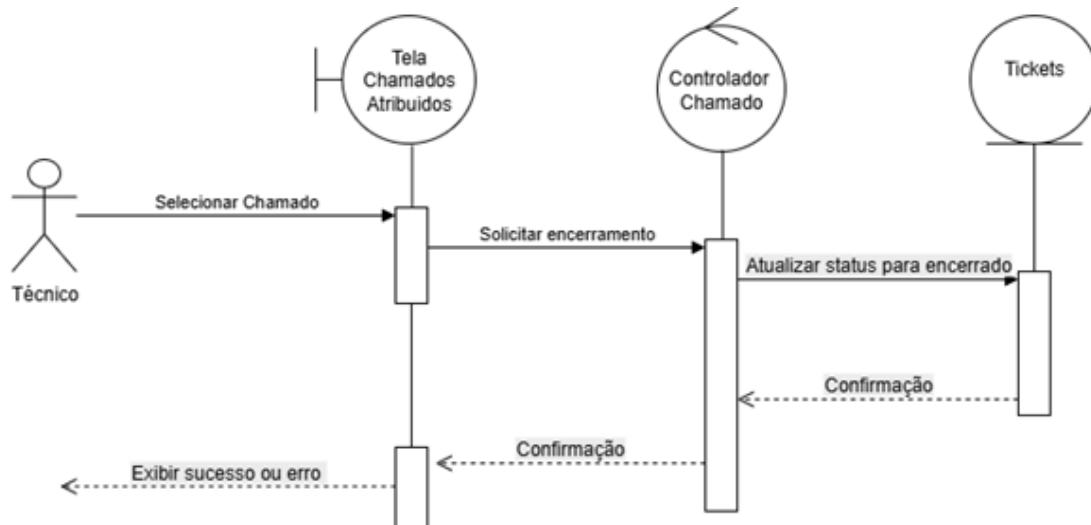
2.4.3 Diagramas de Sequência: Consultar Status de Chamado



Descrição: Este diagrama descreve como o usuário verifica o status dos chamados

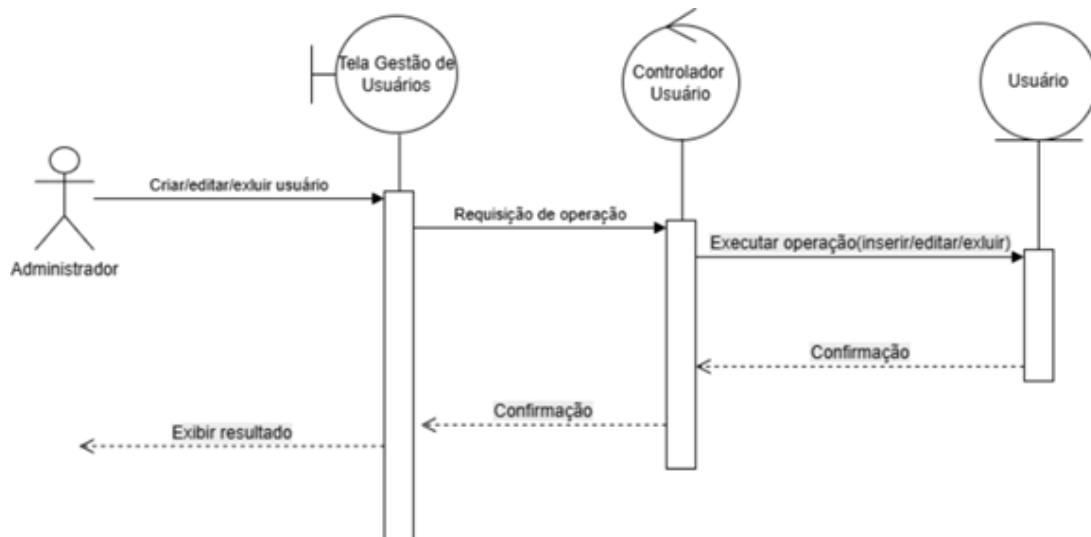
registrados. A interface solicita a lista de chamados do controlador, que recupera os dados da entidade de tickets e retorna os detalhes ao usuário.

2.4.4 Diagramas de Sequência: Abrir e Encerrar Chamado



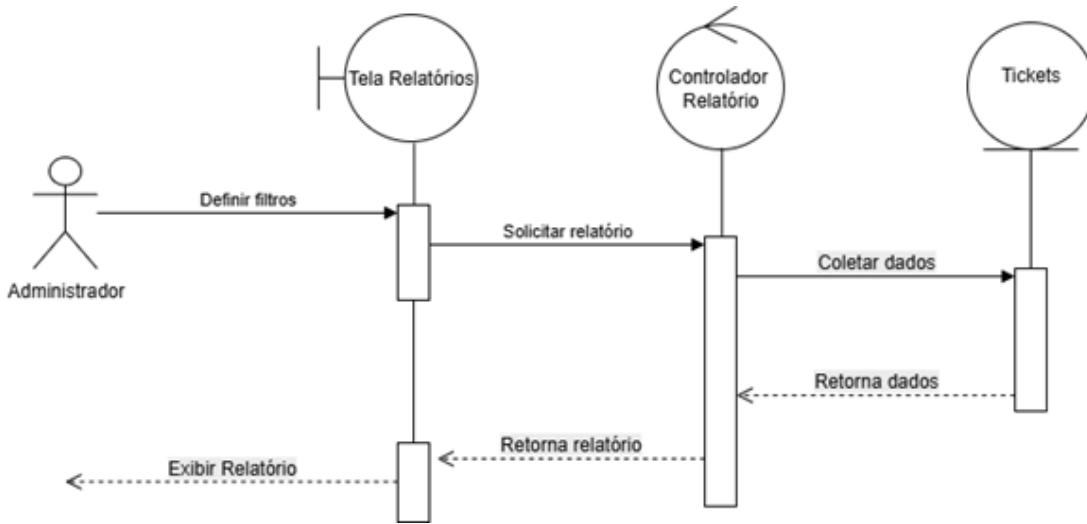
Descrição: Este diagrama mostra o fluxo em que o técnico encerra um chamado após resolver o problema. O técnico acessa os chamados atribuídos, seleciona o resolvido e finaliza. O controlador atualiza a entidade de tickets e o sistema notifica o usuário.

2.4.5 Diagramas de Sequência: Gerenciar Usuários e Técnicos



Descrição: Este diagrama descreve como o administrador gerencia perfis de usuários e técnicos. A interface administrativa envia comandos ao controlador, que consulta e atualiza a entidade de usuário conforme os dados inseridos ou alterados.

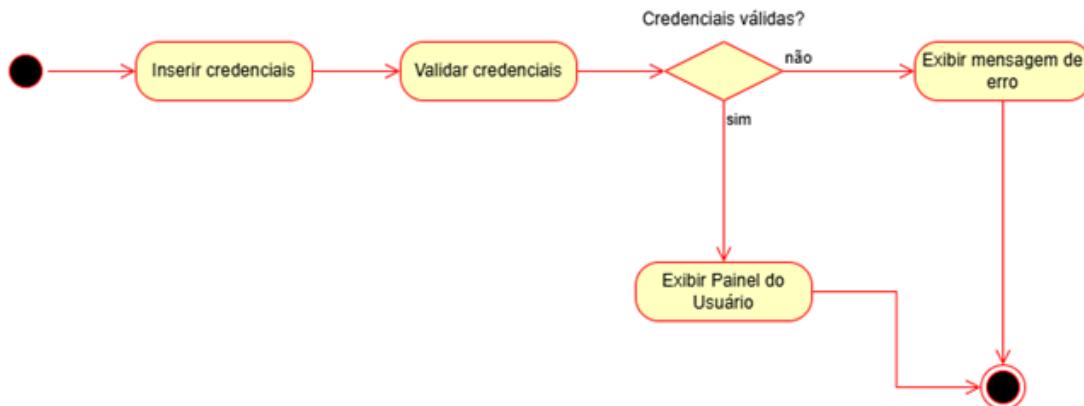
2.4.6 Diagramas de Sequência: Emitir Relatórios



Descrição: Este diagrama mostra o processo de emissão de relatórios gerenciais. O administrador define os filtros, o controlador coleta os dados da entidade tickets e o sistema gera o relatório com os resultados solicitados.

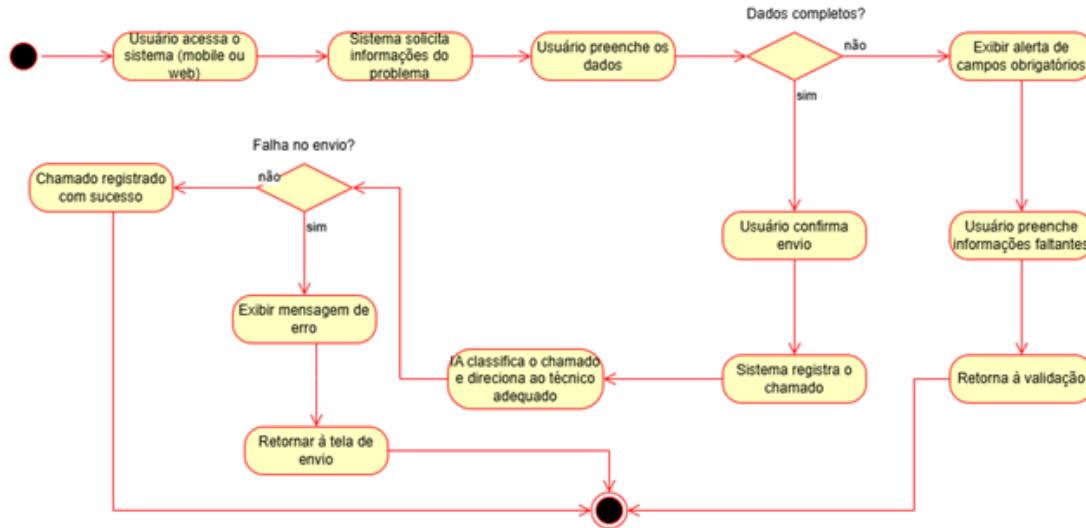
2.5 Diagramas de Atividade

2.5.0.1 Login e Validação



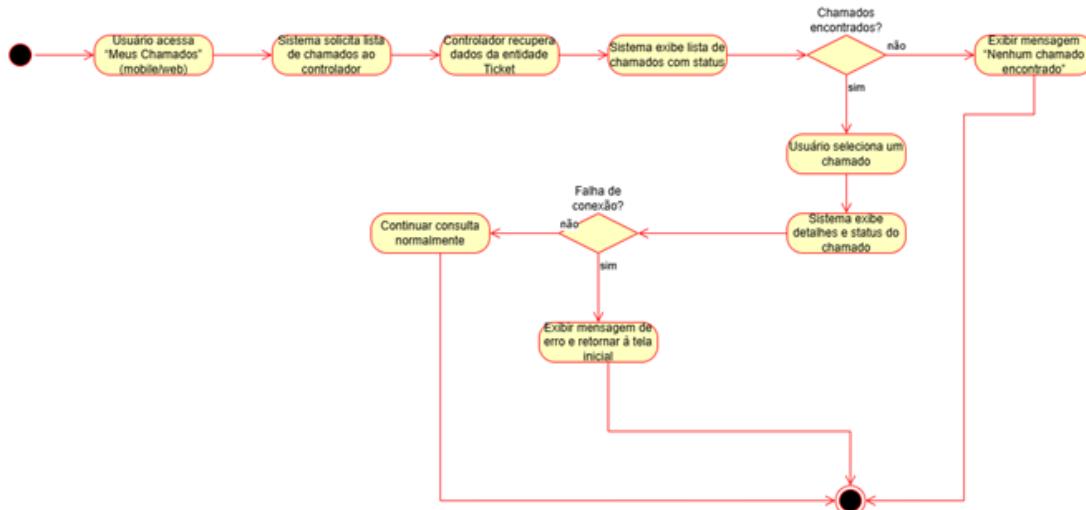
Descrição: Fluxo do login no sistema: o usuário insere as credenciais, o sistema valida e decide o resultado. Se corretas, acessa o painel; se incorretas, exibe erro.

2.5.0.2 Registrar Chamado



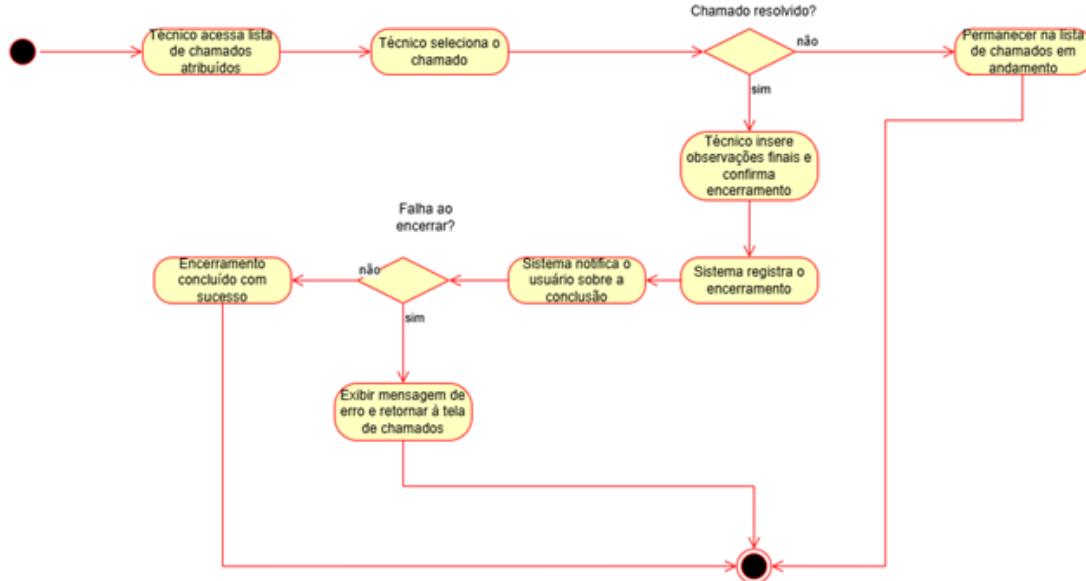
Descrição: O fluxo para registrar um chamado. O usuário preenche e envia as informações; o sistema valida os dados, registra o chamado e aciona a IA para direcionamento. Em caso de erro ou dados incompletos, o sistema exibe alertas e retorna à etapa anterior.

2.5.0.3 Consultar Status de Chamado



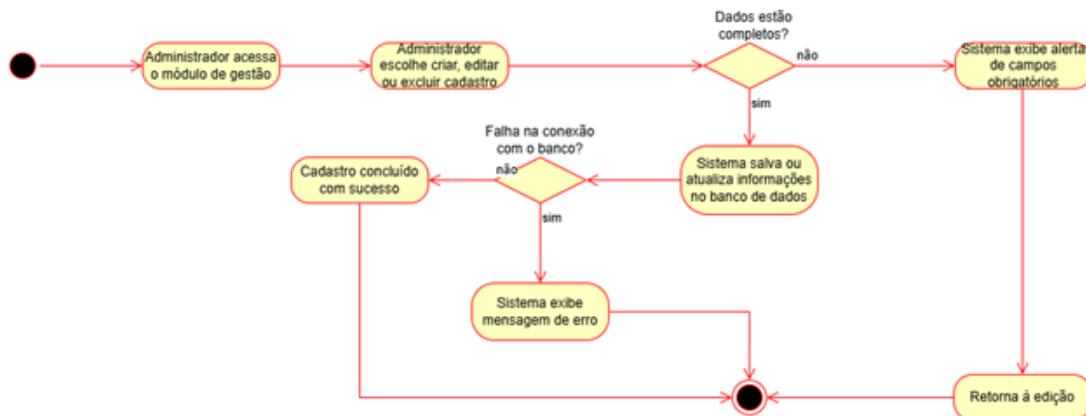
Descrição: O fluxo representa o processo em que o usuário acessa seus chamados, visualiza o status e os detalhes de cada um. Caso não existam chamados ou ocorra falha de conexão, o sistema exibe mensagens de aviso.

2.5.0.4 Abrir e Fechar Chamado



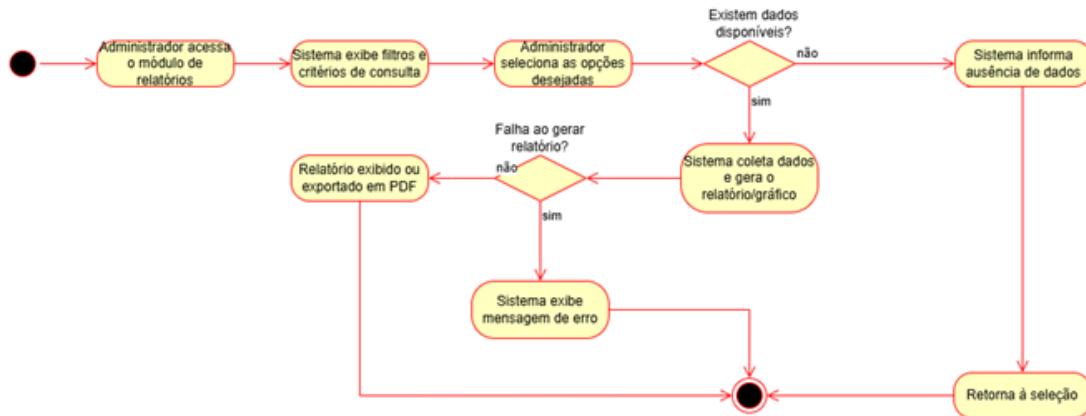
Descrição: Mostra o processo em que o técnico encerra um chamado resolvido. O sistema atualiza o status e notifica o usuário. Caso o chamado não esteja resolvido ou ocorra falha no encerramento, o processo é interrompido.

2.5.0.5 Gerenciar Usuários e Técnicos



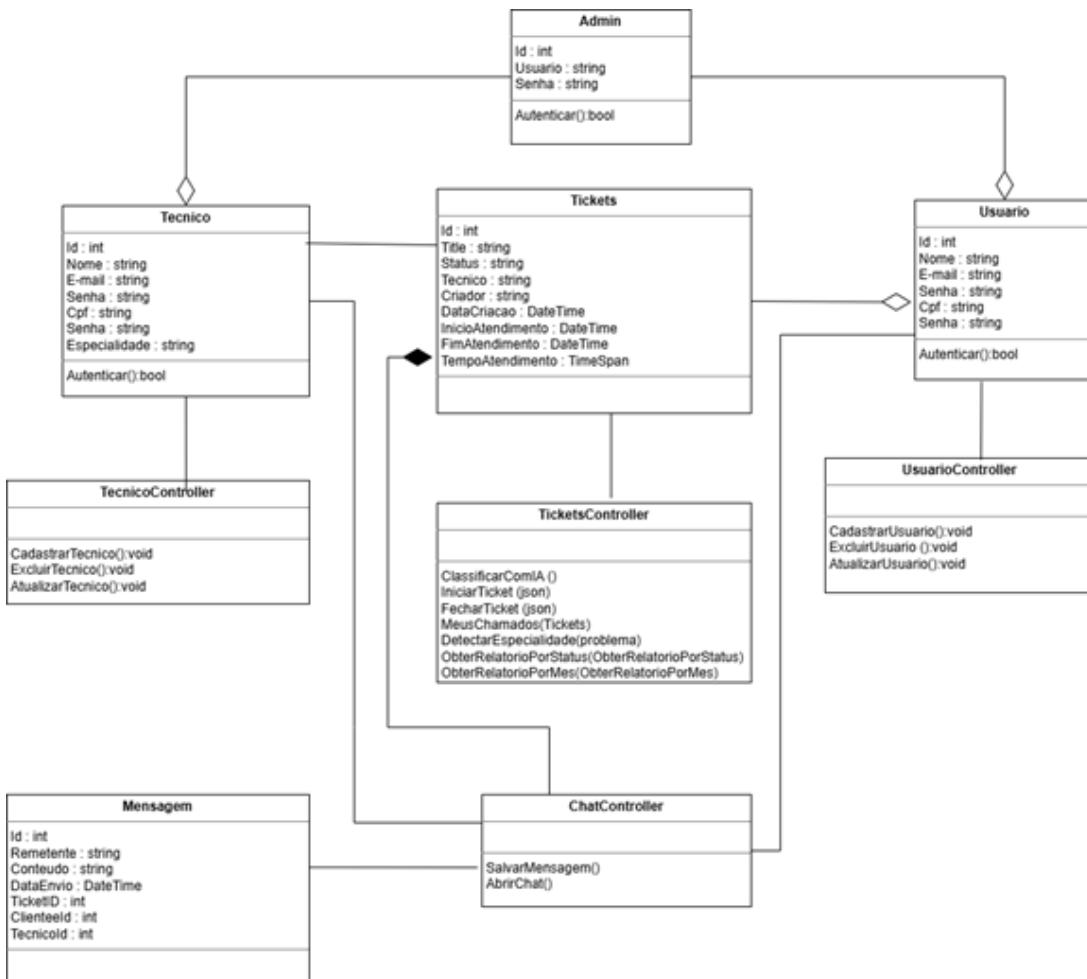
Descrição: Representa o processo em que o administrador gerencia contas de usuários e técnicos, podendo criar, editar ou excluir cadastros. O sistema valida os dados e salva as alterações, exibindo mensagens em caso de erro ou campos faltantes.

2.5.0.6 Emitir Relatórios



Descrição: Representa o processo em que o administrador define filtros e gera relatórios de desempenho ou volume de chamados. O sistema valida os dados, gera os resultados e trata falhas ou ausência de informações.

2.5.1 Diagrama de Classes



Descrição: O diagrama de classes apresentado descreve a estrutura principal do sistema de gestão de chamados com suporte à inteligência artificial. Nele, estão representadas as principais classes envolvidas nas funcionalidades do sistema, como login, registro e encerramento de chamados, classificação automática por IA, acompanhamento de status, geração de relatórios e gestão de usuários.

Cada classe define seus atributos, que representam as informações que ela armazena, e seus métodos, que representam as ações ou comportamentos que ela executa. As relações entre as classes também foram modeladas para demonstrar como os elementos do sistema interagem entre si.

Dentre os principais relacionamentos modelados, destacam-se:

Agregação: As classes *Admin*, *Usuario* e *Tecnico* estão conectadas por meio de relações de agregação, indicando que o administrador é responsável por gerenciar os usuários e técnicos cadastrados, sem que estes dependam diretamente de sua existência. De forma semelhante, a classe *Usuario* agrupa *Tickets*, o que demonstra que um usuário pode criar vários tickets, mas os tickets podem permanecer no sistema mesmo após a exclusão do usuário.

Associação: O *Tecnico* está associado à classe *Tickets*, representando que um técnico pode atender diversos chamados, mas o chamado não pertence ao técnico. Os controladores (*UsuarioController*, *TecnicoController*, *TicketsController* e *ChatController*) mantêm associações simples com as classes correspondentes, sendo responsáveis por intermediar a comunicação entre as regras de negócio e a interface do sistema, conforme o padrão MVC.

Composição: A relação entre *Tickets* e *Mensagem* é de composição, pois uma mensagem existe exclusivamente dentro de um ticket. Dessa forma, ao excluir um ticket, todas as mensagens relacionadas a ele são automaticamente removidas do sistema.

Esse diagrama oferece uma visão clara e estruturada dos elementos centrais do sistema, sendo essencial para o entendimento da lógica do projeto e servindo de base para a implementação em linguagem de programação orientada a objetos, como C(SHARP).

2.5.2 Diagrama de Implantação



Figura 9 – Diagrama de Implantação do Sistema

Manual de Implantação do Sistema

1. Estrutura do sistema:

O sistema é composto por três interfaces principais:

- **Aplicativo Mobile:** Utilizado pelos usuários para registrar novos chamados.
- **Aplicativo Web:** Utilizado pelos técnicos para visualizar, atender e encerrar chamados, e também pelos usuários para registrar novos chamados.
- **Aplicativo Desktop:** Utilizado pelo administrador para gerenciar usuários, técnicos e relatórios.

Todos esses aplicativos se conectam ao servidor principal, onde estão instalados:

- O sistema (backend);
- O banco de dados (Microsoft SQL Server).

A comunicação entre os componentes ocorre através dos seguintes protocolos:

- **HTTP:** Para a troca de informações entre os aplicativos e o servidor;
- **TCP/IP:** Para garantir a conectividade entre os dispositivos na rede.

2. Configuração do servidor:

Para funcionar corretamente, o servidor deve conter:

- Um sistema operacional compatível;
- O Microsoft SQL Server instalado para armazenar os dados;
- O backend do sistema instalado e configurado, preparado para receber e processar requisições dos usuários.

3. Instalação do banco de dados:

No servidor, após a instalação do SQL Server, deve-se:

- Criar ou importar a estrutura do banco de dados conforme o projeto do sistema;
- Definir as tabelas, colunas e relacionamentos;
- Configurar as permissões básicas de acesso para os usuários do sistema.

4. Instalação dos aplicativos:

Após a configuração do servidor e banco de dados, os aplicativos devem ser instalados:

- **Aplicativo Mobile:** Instalado em smartphones via loja de aplicativos ou pacote instalador (APK);
- **Aplicativo Web:** Acessado diretamente via navegador de internet utilizando o endereço do servidor (via HTTP);
- **Aplicativo Desktop:** Instalado em máquinas administrativas usando um instalador apropriado.

Durante a instalação ou no primeiro acesso, será necessário configurar os aplicativos com o endereço do servidor (IP ou domínio) para que possam se conectar corretamente.

5. Comunicação entre os componentes:

A comunicação entre os aplicativos e o servidor é feita por meio de:

- **HTTP:** Protocolo para envio e recebimento de dados;
- **TCP/IP:** Protocolo de rede que permite a conexão entre dispositivos da rede local (intranet) ou da internet.

6. Manutenção e atualizações:

Após a instalação e funcionamento do sistema, é necessário realizar manutenções periódicas, como:

- Monitoramento do servidor;
- Aplicação de atualizações de sistema;
- Correções de erros.

2.6 Requisitos Funcionais (RF)

- **RF01:** O sistema deve permitir login de usuário, técnico e administrador.
- **RF02:** O sistema deve realizar validação de credenciais.

- **RF03:** O sistema deve permitir registro de chamado pelo Mobile.
- **RF04:** O sistema deve permitir registro de chamado pelo Web.
- **RF05:** O sistema deve realizar classificação automática de chamados por IA.
- **RF06:** O sistema deve realizar encaminhamento automático ao técnico responsável.
- **RF07:** O sistema deve permitir consulta de status do chamado pelo usuário e técnico.
- **RF08:** O sistema deve permitir encerramento de chamados pelo técnico.
- **RF09:** O sistema deve permitir gerenciamento de usuários e técnicos (administrador).
- **RF10:** O sistema deve permitir geração de relatórios gerenciais (administrador).

2.7 Requisitos Não Funcionais (RNF)

- **RNF01:** A interface deve ser responsiva (desktop, mobile e web).
- **RNF02:** O sistema deve ter tempo de resposta inferior a 2 segundos para operações críticas.
- **RNF03:** O sistema deve realizar a classificação de chamados por IA em até 5 segundos.
- **RNF04:** O sistema deve suportar pelo menos 100 usuários simultâneos.
- **RNF05:** O sistema deve suportar até 1.000 chamados ativos.
- **RNF06:** O sistema deve garantir conformidade com a LGPD.
- **RNF07:** A navegação deve ser intuitiva e acessível.

2.8 Banco de Dados

2.8.1 Scripts do Banco de Dados

```
1 CREATE DATABASE ProjetoDB;
2 GO
3
4 USE ProjetoDB;
5 GO
```

Listing 2.1 – Criação do banco de dados ProjetoDB

```

1 CREATE TABLE dbo.tb_admin (
2     id_admin INT PRIMARY KEY IDENTITY(1,1) NOT NULL,
3     usuario VARCHAR(50) NOT NULL,
4     senha VARCHAR(100) NOT NULL
5 );

```

Listing 2.2 – Criação da tabela Admin

```

1 CREATE TABLE dbo.tb_tecnico (
2     id INT NOT NULL PRIMARY KEY,
3     nome NVARCHAR(100) NOT NULL,
4     cpf NVARCHAR(14) NOT NULL,
5     email NVARCHAR(100) NOT NULL,
6     senha NVARCHAR(255) NOT NULL,
7     especialidade NVARCHAR(100) NOT NULL
8 );

```

Listing 2.3 – Criação da tabela tb_tecnico

```

1 CREATE TABLE dbo.tb_usuario (
2     id INT NOT NULL PRIMARY KEY,
3     nome NVARCHAR(100) NOT NULL,
4     cpf NVARCHAR(14) NOT NULL,
5     email NVARCHAR(100) NOT NULL,
6     senha NVARCHAR(255) NOT NULL
7 );

```

Listing 2.4 – Criação da tabela tb_usuario

```

1 CREATE TABLE dbo.Tickets (
2     Id INT NOT NULL PRIMARY KEY,
3     Title NVARCHAR(255) NOT NULL,
4     Description NVARCHAR(MAX) NULL,
5     Status NVARCHAR(50) NOT NULL,
6     DataCriacao DATETIME NULL,
7     InicioAtendimento DATETIME NULL,
8     FimAtendimento DATETIME NULL,
9     TempoAtendimento NVARCHAR(50) NULL,
10    CriadorId INT NOT NULL,
11    TecnicoId INT NULL,
12    FOREIGN KEY (CriadorId) REFERENCES tb_usuario(id),
13    FOREIGN KEY (TecnicoId) REFERENCES tb_tecnico(id)
14 );

```

Listing 2.5 – Criação da tabela Tickets

```

1 CREATE TABLE dbo.Mensagens (
2     Id INT NOT NULL PRIMARY KEY,

```

```

3     Remetente NVARCHAR(100) NOT NULL,
4     Destinatario NVARCHAR(100) NOT NULL,
5     Conteudo NVARCHAR(MAX) NOT NULL,
6     DataEnvio DATETIME NULL,
7     TicketId INT NOT NULL,
8     ClienteId INT NULL,
9     TecnicoId INT NULL,
10    FOREIGN KEY (TicketId) REFERENCES Ticket(Id),
11    FOREIGN KEY (ClienteId) REFERENCES Cliente(Id),
12    FOREIGN KEY (TecnicoId) REFERENCES Tecnico(Id)
13 );

```

Listing 2.6 – Criação da tabela Mensagens

```

1 CREATE TABLE ChatFiles (
2     Id INT IDENTITY(1,1) PRIMARY KEY,
3     FileName NVARCHAR(255) NOT NULL,
4     FileUrl NVARCHAR(500) NOT NULL,
5     UploadedAt DATETIME NULL,
6     UploadedById INT NOT NULL,
7     UploadedToId INT NOT NULL,
8     TicketId INT NOT NULL,
9     UploadedByName NVARCHAR(150) NULL,
10    UploadedByTecnicoId INT NULL,
11
12    CONSTRAINT FK_ChatFiles_UploadedBy FOREIGN KEY (UploadedById) REFERENCES
13        AspNetUsers(Id),
14    CONSTRAINT FK_ChatFiles_UploadedTo FOREIGN KEY (UploadedToId) REFERENCES
15        AspNetUsers(Id),
16    CONSTRAINT FK_ChatFiles_Ticket FOREIGN KEY (TicketId) REFERENCES Tickets(Id)
17 );

```

Listing 2.7 – Criação da tabela ChatFiles

2.8.2 Scripts de Testes

```

1 INSERT INTO dbo.tb_admin (usuario, senha)
2 VALUES (
3     'admin_cauan',
4     'senhaSegura123'
5 );

```

Listing 2.8 – Inserção de um usuário admin na tabela tb_admin

Listing 2.9 – Scripts de inserção de dados para testes no banco de dados ProjetoDB.

```

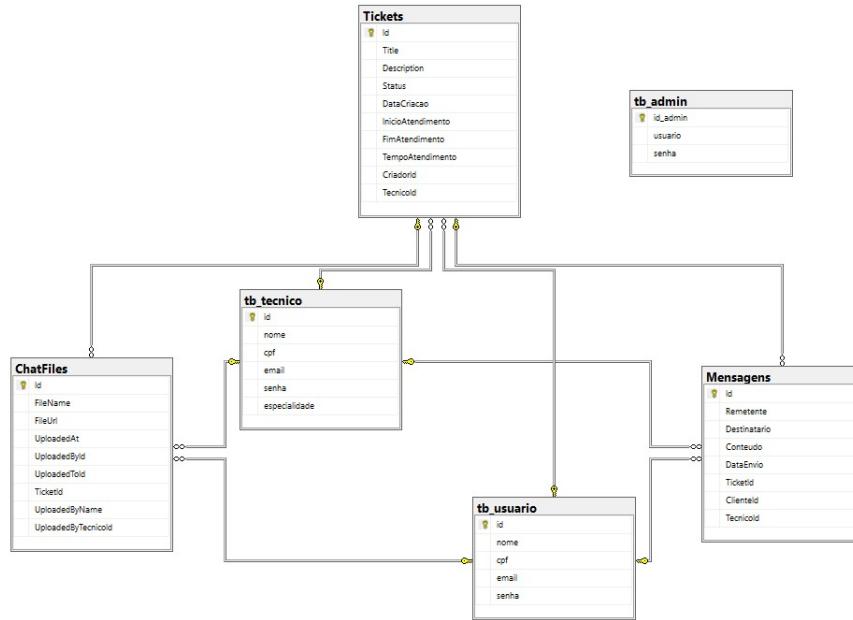
1 -- Inserção de um usuário
2 INSERT INTO dbo.tb_usuario (nome, cpf, email, senha)

```

```
3 VALUES ('João Silva', '123.456.789-00', 'joao.silva@email.com', '
    senhaSegura123');
4
5 -- Inserção de um técnico
6 INSERT INTO dbo.tb_tecnico (nome, cpf, email, senha, especialidade)
7 VALUES ('Maria Oliveira', '987.654.321-00', 'maria.oliveira@empresa.com'
    , 'senhaForte456', 'Redes e Infraestrutura');
8
9 -- Inserção de um ticket de suporte
10 INSERT INTO dbo.Tickets (
11     Title,
12     Description,
13     Status,
14     DataCriacao,
15     InicioAtendimento,
16     FimAtendimento,
17     TempoAtendimento,
18     CriadorId,
19     TecnicoId
20 )
21 VALUES (
22     'Erro ao acessar sistema',
23     'Usuário relata que não consegue acessar o sistema após
        atualizar o.',
24     'Aberto',
25     GETDATE(),
26     NULL,
27     NULL,
28     NULL,
29     1, -- ID do usuário criador
30     1 -- ID do técnico responsável
31 );
32
33 -- Inserção de uma mensagem vinculada ao ticket
34 INSERT INTO dbo.Mensagens (
35     Remetente,
36     Destinatario,
37     Conteudo,
38     DataEnvio,
39     TicketId,
40     ClienteId,
41     TecnicoId
42 )
43 VALUES (
44     'joao.silva@email.com',
45     'maria.oliveira@empresa.com',
46     'Olá Maria, estou com dificuldades para acessar o sistema desde
```

```
        ontem. ,
47      GETDATE(),
48      1,    -- ID do ticket relacionado
49      1,    -- ID do cliente (se aplic vel)
50      1    -- ID do t cnico (se aplic vel)
51  );
52
53  -- Inser o de um arquivo de chat
54  INSERT INTO ProjetoDB.dbo.ChatFiles (
55      FileName,
56      FileUrl,
57      UploadedAt,
58      UploadedById,
59      UploadedToId,
60      TicketId,
61      UploadedByName,
62      UploadedByTecnicoId
63  )
64  VALUES (
65      'manual_sistema.pdf',
66      'https://empresa.com/arquivos/manual_sistema.pdf',
67      GETDATE(),
68      1,    -- ID do usu rio que enviou
69      2,    -- ID do destinat rio
70      1,    -- ID do ticket relacionado
71      'Maria Oliveira',
72      1    -- ID do t cnico que enviou
73  );
```

2.8.3 Diagrama Entidade e Relacionamento



2.8.4 Dicionário de Dados: Tabela tb_admin

Nome do Campo	Tipo de Dado	Tamanho	Nulo	Chave	Descrição
id_admin	INT	-	NÃO	PK	Identificador único do administrador na tabela, auto-incrementado.
usuario	VARCHAR	50	NÃO	-	Nome de usuário utilizado para login.
senha	VARCHAR	100	NÃO	-	Senha do usuário, armazenada de forma segura (hash).

2.8.5 Dicionário de Dados: Tabela tb_usuario

Nome do Campo	Tipo de Dado	Tamanho	Nulo	Chave	Descrição
id	INT	-	NÃO	PK	Identificador único do usuário na tabela.
nome	NVARCHAR	100	NÃO	-	Nome completo do usuário.
cpf	NVARCHAR	14	NÃO	-	CPF do usuário (formato com pontos e traço, ex: 000.000.000-00).
email	NVARCHAR	100	NÃO	-	Endereço de e-mail do usuário, usado para login e contato.
senha	NVARCHAR	255	NÃO	-	Senha criptografada utilizada para autenticação no sistema.

2.8.6 Dicionário de Dados: Tabela tb_tecnico

Nome do Campo	Tipo de Dado	Tamanho	Nulo	Chave	Descrição
id	INT	-	NÃO	PK	Identificador único do técnico na tabela.
nome	NVARCHAR	100	NÃO	-	Nome completo do técnico.
cpf	NVARCHAR	14	NÃO	-	CPF do técnico (formato com pontos e traço, ex: 000.000.000-00).
email	NVARCHAR	100	NÃO	-	Endereço de e-mail do técnico, utilizado para login e comunicação.
senha	NVARCHAR	255	NÃO	-	Senha criptografada usada para autenticação do técnico.
especialidade	NVARCHAR	100	NÃO	-	Área de atuação ou especialidade do técnico (ex: redes, hardware, software).

2.8.7 Dicionário de Dados: Tabela Tickets

Nome do Campo	Tipo de Dado	Tamanho	Nulo	Chave	Descrição
Id	INT	-	NÃO	PK	Identificador único do ticket.
Title	NVARCHAR	255	NÃO	-	Título ou assunto principal do chamado/ticket.
Description	NVARCHAR(MAX)	-	SIM	-	Descrição detalhada do problema ou solicitação registrada.
Status	NVARCHAR	50	NÃO	-	Situação atual do ticket (ex: “Aberto”, “Em andamento”, “Finalizado”, “Cancelado”).
DataCriacao	DATETIME	-	SIM	-	Data e hora em que o ticket foi criado.
InicioAtendimento	DATETIME	-	SIM	-	Data e hora em que o técnico iniciou o atendimento.
FimAtendimento	DATETIME	-	SIM	-	Data e hora em que o atendimento foi concluído.
TempoAtendimento	NVARCHAR	50	SIM	-	Tempo total gasto no atendimento (geralmente calculado automaticamente).
CriadorId	INT	-	NÃO	FK	Identificador do usuário que criou o ticket (referência à tabela tb_usuario).
TecnicoId	INT	-	SIM	FK	Identificador do técnico responsável pelo ticket (referência à tabela tb_tecnico).

Restrições e Relacionamentos:

- **PRIMARY KEY:** Id — garante a unicidade de cada ticket.
- **FOREIGN KEY:** CriadorId → *tbusuario(Id)* — identifica o usuário que criou o ticket.
- **FOREIGN KEY:** TecnicoId → *tbtecnico(Id)* — identifica o técnico responsável pelo atendimento.

2.8.8 Dicionário de Dados — Tabela Mensagens

Campo	Tipo de Dado	Tamanho	Permite Nulo	Descrição
Id	INT	—	Não	Identificador único da mensagem (chave primária).
Remetente	NVARCHAR	100	Não	Nome de quem enviou a mensagem (cliente ou técnico).
Destinatario	NVARCHAR	100	Não	Nome de quem recebe a mensagem.
Conteudo	NVARCHAR(MAX)	—	Não	Texto completo da mensagem enviada.
DataEnvio	DATETIME	—	Sim	Data e hora em que a mensagem foi enviada.
TicketId	INT	—	Não	Identificador do ticket ao qual a mensagem pertence.
ClienteId	INT	—	Sim	Identificador do cliente associado à mensagem.
TecnicoId	INT	—	Sim	Identificador do técnico associado à mensagem.

Restrições e Relacionamentos:

- **PRIMARY KEY:** Id — garante a unicidade de cada mensagem.

- **FOREIGN KEY:** TicketId → *Ticket(Id)* — relaciona a mensagem ao ticket correspondente.
- **FOREIGN KEY:** ClienteId → *Cliente(Id)* — identifica o cliente associado à mensagem.
- **FOREIGN KEY:** TecnicoId → *Tecnico(Id)* — identifica o técnico associado à mensagem.

2.8.9 Dicionário de Dados — Tabela ChatFiles

Campo	Tipo de Dado	Tamanho	Permite Nulo	Descrição
Id	INT (IDENTITY)	—	Não	Identificador único do arquivo enviado (chave primária, auto-incremento).
FileName	NVARCHAR	255	Não	Nome do arquivo enviado (com extensão).
FileUrl	NVARCHAR	500	Não	Caminho completo (URL) onde o arquivo está armazenado.
UploadedAt	DATETIME	—	Sim	Data e hora do envio do arquivo.
UploadedById	INT	—	Não	Identificador do usuário que fez o upload do arquivo.
UploadedToId	INT	—	Não	Identificador do usuário que recebeu o arquivo.
TicketId	INT	—	Não	Identificador do ticket ao qual o arquivo está vinculado.
UploadedByName	NVARCHAR	150	Sim	Nome do usuário que realizou o upload.
UploadedByTecnicoId	INT	—	Sim	Identificador do técnico que enviou o arquivo (se aplicável).

Restrições e Relacionamentos:

- **PRIMARY KEY:** Id — garante a unicidade de cada arquivo enviado.
- **FOREIGN KEY:** UploadedById → *AspNetUsers(Id)* — identifica o usuário que fez o upload.
- **FOREIGN KEY:** UploadedToId → *AspNetUsers(Id)* — identifica o usuário que recebeu o arquivo.
- **FOREIGN KEY:** TicketId → *Tickets(Id)* — relaciona o arquivo ao ticket correspondente.

2.8.10 Planilha de Teste do Sistema de Chamados

A planilha de testes a seguir apresenta os cenários verificados durante a validação do sistema de chamados, abrangendo desde o login até o encerramento dos atendimentos. Cada teste foi executado para garantir o correto funcionamento das funcionalidades principais do sistema, bem como a integridade das informações armazenadas no banco de dados.

Tabela 1 – Planilha de Testes Funcionais do Sistema de Chamados

ID do Teste	Funcionalidade	Descrição do Teste	Resultado Esperado	Resultado Obtido	Status
T01	Login de Cliente	Inserir e-mail e senha válidos do cliente e acessar o sistema.	Sistema redireciona o cliente para a tela inicial.	Conforme esperado.	Aprovado
T02	Login de Técnico	Inserir e-mail e senha válidos do técnico.	Sistema redireciona o técnico para o painel de chamados.	Conforme esperado.	Aprovado
T03	Login Inválido	Inserir e-mail ou senha incorretos.	Exibir mensagem “Credenciais inválidas”.	Conforme esperado.	Aprovado
T04	Abertura de Chamado	Cliente informa o problema e cria um novo chamado.	Chamado é salvo no banco e exibido na lista de tickets.	Conforme esperado.	Aprovado
T05	Redirecionamento Automático (IA)	Após o chamado ser criado, o sistema usa a IA para identificar a especialidade e atribuir ao técnico adequado.	Chamado é direcionado automaticamente ao técnico da área correspondente.	Conforme esperado.	Aprovado
T06	Início de Atendimento	Técnico acessa o ticket e clica em “Iniciar Atendimento”.	Status muda para “Em andamento” e chat é aberto.	Conforme esperado.	Aprovado
T07	Envio de Mensagem	Cliente envia mensagem de texto ao técnico.	Mensagem aparece instantaneamente no chat e é salva no banco.	Conforme esperado.	Aprovado
T08	Resposta do Técnico	Técnico responde no chat.	Mensagem é exibida em tempo real e registrada no banco.	Conforme esperado.	Aprovado
T09	Upload de Arquivo	Técnico ou cliente envia um arquivo anexado ao chat.	Arquivo é salvo na tabela ChatFiles e aparece no histórico.	Conforme esperado.	Aprovado
T10	Finalizar Atendimento (Técnico)	Técnico marca o ticket como finalizado.	Status muda para “Aguardando Confirmação do Cliente”.	Conforme esperado.	Aprovado
T11	Confirmação de Encerramento (Cliente)	Cliente recebe notificação perguntando se deseja encerrar o atendimento.	Caso confirme, status muda para “Finalizado” e o chat é encerrado.	Conforme esperado.	Aprovado
T12	Recusa de Encerramento (Cliente)	Cliente seleciona “Não” ao ser perguntado sobre encerramento.	Atendimento continua ativo e chat permanece aberto.	Conforme esperado.	Aprovado
T13	Atualização de Status no Banco	Verificar no banco de dados se o status muda conforme ações (em andamento, finalizado, etc.).	As atualizações ocorrem corretamente nas tabelas Tickets e Mensagens.	Conforme esperado.	Aprovado

2.9 Interface Humano-Computador

2.9.1 Protótipo de Tela: Mobile



Tela em que o usuário preenche os campos para acessar o aplicativo.

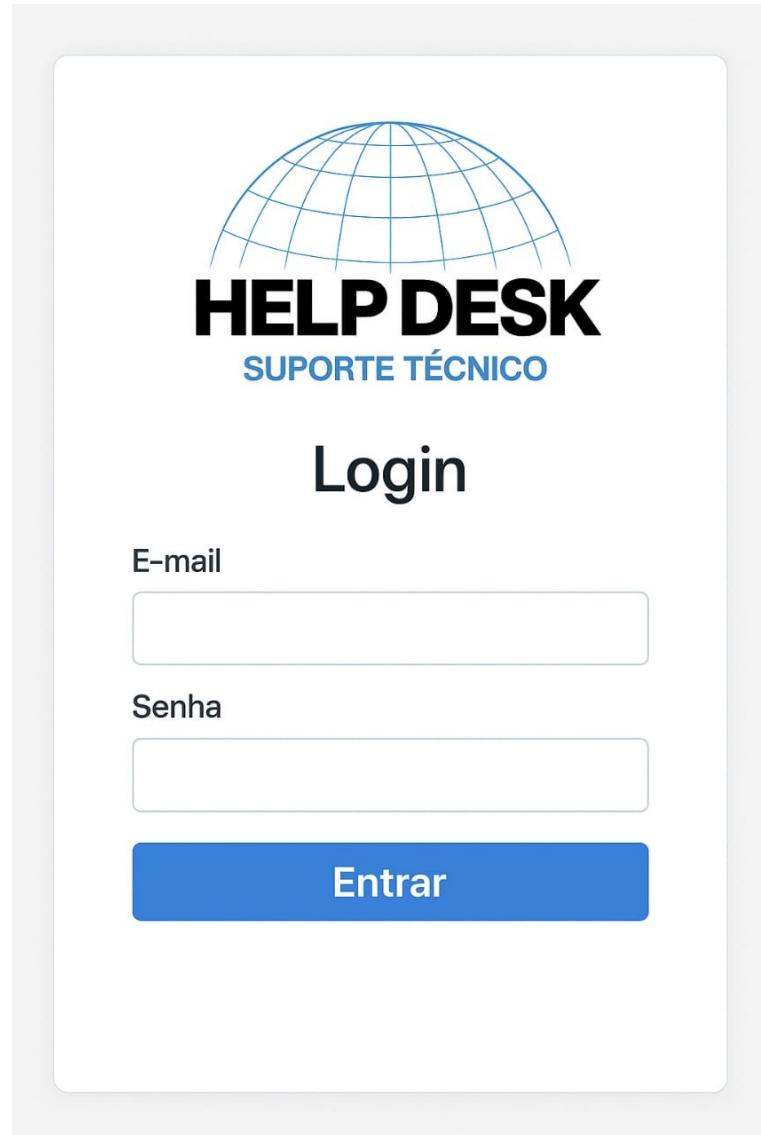


Tela em que o usuário insere o problema e a prioridade; o sistema o encaminha para o especialista.



Tela que registra os chamados feitos e mostra o andamento de cada um como histórico.

2.9.2 Protótipo de Tela: Web



Tela onde os técnicos realizam o login.



Chamado

Usuário

Assunto

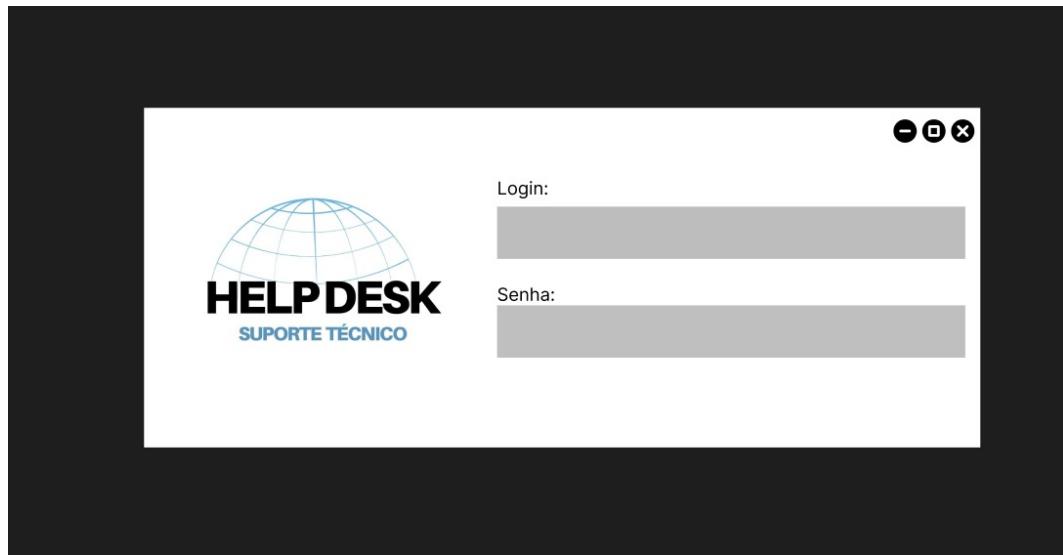
Descrição

Encerrar Chamado

Reabrir Chamado

Tela utilizada pelos técnicos para preencher os campos e encerrar os chamados.

2.9.3 Protótipo de Tela: Desktop



Tela de login da versão Desktop.

3 CONCLUSÃO

REFERÊNCIAS