

**UNIVERSIDADE PAULISTA**

**ICET - INSTITUTO DE CIÊNCIAS EXATAS E TECNOLOGIA**

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

**PROJETO INTEGRADO MULTIDISCIPLINAR**  
**PIM III**

**Sistema integrado para gestão de chamados e suporte técnico baseado em IA**

**Nome**  **R.A**

Agatha Kethyllen Ribeiro de Oliveira. R016DB6

Camilly Vitória Menezes Rodrigues. G9976J0

Guilherme dos Santos Damascena. R086283

João Lucas Alves Gregorio. G9979F0

Micaias Leonardo Costa Viola. G989BE8

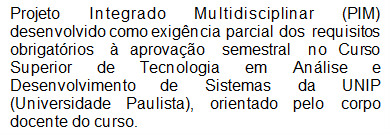
Vanessa Helena Bandeli Scarin. R036499

**SÃO JOSÉ DOS CAMPOS – SP**

**JUNHO/2025**

|  |  |
| --- | --- |
|  | **RA** |
| Aluno 1 Agatha Kethyllen Ribeiro de Oliveira. | R016DB6 |
| Aluno 2 Camilly Vitória Menezes Rodrigues. | G9976J0 |
| Aluno 3 Guilherme dos Santos Damascena. | R086283 |
| Aluno 4 João Lucas Alves Gregorio. | G9979F0 |
| Aluno 5 Micaias Leonardo Costa Viola. | G989BE8 |
| Aluno 6 Vanessa Helena Bandeli Scarin. | R036499 |

**Sistema integrado para gestão de chamados e suporte técnico baseado em IA**



**São José dos Campos – SP**

**MAIO/2025**

**RESUMO**

Um chamado é um registro formal de um pedido de ajuda, suporte ou serviço feito por um usuário para uma equipe responsável, geralmente de suporte técnico. Um sistema de gerenciamento de chamados tem como função centralizar todas as solicitações, priorizar e encaminhar para os departamentos responsáveis por solucionar os problemas, com o objetivo de viabilizar o atendimento as necessidades dos clientes de forma rápida e eficiente. O seguinte projeto tem como finalidade conduzir o levantamento de requisitos para um software de gerenciamento de chamados, com a intenção de otimizar a seção de suporte técnico de uma empresa com soluções que aumentem a eficiência e qualidade do suporte. O objetivo principal é treinar uma Inteligência Artificial (IA) a fim de automatizar os processos de priorização e na solução dos problemas relatados, oferecendo respostas simples e precisas. Foi elaborado um protótipo para a apresentação inicial do produto com o fim de validar os requisitos, com o intuito de validar se o projeto está de acordo com as expectativas do cliente e viabilizar seu desenvolvimento futuro. As metodologias utilizadas foram entrevistas com profissionais que atuam na área de suporte técnico para a elicitação de requisitos, análise documental e pesquisas bibliográficas para compreender o cenário atual de suporte.

**Palavras-chave:** Gerenciamento de chamados, inteligência artificial, suporte técnico.

**SUMÁRIO**

[1. INTRODUÇÃO 4](#_Toc93405477)

[1.1 OBJETIVO GERAL 6](#_Toc1507821438)

[1.2 Objetivos Específicos 7](#_Toc524458543)

[1.3 Atividade de Extensão Universitária 7](#_Toc1892361360)

[2. CARACTERIZAÇÃO DA EMPRESA (SUPORTE TÉCNICO) 7](#_Toc1516731836)

[3. ANÁLISE DE SISTEMAS ORIENTADA A OBJETOS 8](#_Toc971092332)

[4. ENGENHARIA DE SOFTWARE II 10](#_Toc236420135)

[5. PROGRAMAÇÃO ORIENTADA A OBJETOS 12](#_Toc244336160)

[6. PROJETO DE INTERFACE COM O USUÁRIO 14](#_Toc1167430919)

[7. BANCO DE DADOS 16](#_Toc718893412)

[8. ECONOMIA E MERCADO 18](#_Toc1562142004)

[9. GESTÃO ESTRATÉGICA DE RECURSOS HUMANOS 19](#_Toc569007446)

[10. DESENVOLVIMENTO 20](#_Toc1398503017)

[11. CONSIDERAÇÕES FINAIS 48](#_Toc1917642838)

[REFERÊNCIAS BIBLIOGRÁFICAS 50](#_Toc990858177)

[ANEXO 1 53](#_Toc1317292014)

# 1. INTRODUÇÃO

Conforme pesquisa realizada pela TNS Research, empresas que realizam investimento em tecnologia têm uma porcentagem de aumento de receita e de crescimento 60% maior do que aquelas que optam por não investir (ELS SISTEMAS, 2024). Neste cenário, a utilização de sistemas no ambiente de trabalho entra em ascensão, facilitando assim o gerenciamento de fluxo de informações e otimizando processos.

Para aplicar o uso de sistemas informatizados em uma empresa, faz-se necessário a inclusão do setor de gerenciamento de chamados através de uma ferramenta digital. Este irá se concentrar em centralizar, organizar, priorizar e solucionar as solicitações de ajuda realizadas pelos clientes para o departamento de suporte técnico, também conhecidos como chamados (TEIXEIRA, 2024).

Num contexto de mudanças constantes e grande fluxo de informações diárias, surgem divergências no atendimento ao cliente humanizado, por uma questão de tempo de espera, critérios para priorização subjetivos e sobrecarga da equipe especializada em suporte, e como a resolução de problemas de forma rápida e eficiente é a chave de retenção de clientes e posicionamento da marca (SEBRAE, 2024), sugere-se a implementação de uma inteligência artificial na ferramenta digital.

A inteligência artificial é um conjunto de tecnologias que permite que os computadores executem diversas funções avançadas. Através do desenvolvimento de algoritmos modelados conforme os processos de decisões do cérebro humano, é possível que a IA aprenda com os dados disponíveis a classificar, prever e fornecer sugestões cada vez mais precisas, conforme o passar do tempo e a precisão de seu banco de dados (IBM, 2023).

Segundo pesquisa realizada pela Zendesk CX Trends 2023, 72% dos clientes querem receber atendimento imediato, somado a isso, 74% dos entrevistados acreditam que a IA deve melhorar a eficiência do atendimento ao cliente. Em contrapartida, 47% dos entrevistados pela Capterra 2022 acreditam que a IA pode gerar frustrações ao ser repetitiva ou não conseguir solucionar problemas complexos.

Partindo deste pressuposto, desenvolve-se neste projeto um sistema integrado para gestão de chamados e suporte técnico com base em inteligência artificial, com o intuito de proporcionar para este setor otimização ao implementar funções de priorização, categorização e sugestões de resolução realizadas pela IA, mas permitindo que problemas sejam encaminhados para o atendimento humano visando melhorar a experiência do cliente em situações complexas, podendo assim diminuir a sobrecarga do setor de suporte e gerar praticidade para o usuário.

.

## 1.1 OBJETIVO GERAL

Realizar o levantamento e análise dos requisitos para a criação de um sistema de suporte técnico inteligente como sistema interno de uma empresa, que utilize Inteligência Artificial (IA) para triagem e sugestão inicial como solução para o chamado, visando otimizar o tempo de resolução e evitando sobrecarga dos técnicos de suporte da empresa. Os dados pessoas serão armazenados em um banco de dados visando a LGPD.

## 1.2 Objetivos Específicos

* Descrever os requisitos funcionais e não funcionais através da elicitação;
* Desenvolver diagramas UML para análise e modelagem dos requisitos, sendo estes de caso de uso e caso de uso descritivo, classe, sequência e implantação;
* Desenvolver o diagrama de entidade-relacionamento, juntamente com o modelo conceitual descritivo, lógico e dicionário de dados;
* Gerar um script de planejamento do banco de dados em SQL Server;
* Criar o protótipo de interface na versão DESKTOP garantindo a usabilidade;
* Realizar uma estimativa de viabilidade do projeto com base nos recursos disponíveis;
* Especificar os Recursos Humanos e mapear o perfil profissional para a capacitação e desenvolvimento do projeto;
* Criar regras de negócio detalhadas para a empresa;
* Gerar planilha de testes unitários;
* Utilizar uma I.A para triagem e solucionamento inicial de chamados.

## 1.3 Atividade de Extensão Universitária

A descrição das atividades extensionistas encontra-se no Anexo 1.

# 2. CARACTERIZAÇÃO DA EMPRESA (SUPORTE TÉCNICO)

Esfera Contabilidade é uma empresa fictícia de médio porte localizada no Vale do Paraíba, apresentando a seguinte estrutura organizacional: Departamento Pessoal, Recursos Humanos, Área Contábil, Área Fiscal, Financeiro, Comercial, Administrativo e TI.

Aproximadamente 80% de seus funcionários são usuários de microcomputadores interligados em rede, e o perfil do usuário é de baixo a médio conhecimento em informática, utilizando equipamentos e ferramentas para atividades diversas como: softwares de gerenciamento administrativo (ERP), financeiro (contábeis e bancários), comerciais (CRM), de desenvolvimento de artes (Photoshop, Illustrator) e aplicações como editores de texto, planilhas, gráficos e apresentações, navegadores de Internet e serviços de e-mail.

O setor de TI é responsável por toda a estrutura de informática da empresa e há um histórico de alta demanda de atendimentos, agravada em horários de pico e causando atritos como demora no atendimento, ausência de levantamento de dados e relatórios, falha na priorização de tarefas, dúvidas quanto à eficiência do setor e insatisfação geral (tanto dos requerentes quanto da equipe de suporte). O atendimento era solicitado via e-mail e continuado pelo aplicativo de mensagens WhatsApp, ligações telefônicas e presencialmente, se necessário.

Na evidência da necessidade de organizar este processo, a empresa deseja adotar um sistema integrado, no qual os colaboradores possam registrar suas solicitações e utilizar uma ferramenta de Inteligência Artificial (IA) para sugerir soluções automáticas ou encaminhar ao técnico adequado com base no histórico de chamados e complexidade do problema. Partindo desta premissa se inicia o desenvolvimento do sistema integrado para gestão de chamados e suporte técnico baseado em IA descrito neste projeto e denominado “HELPBOX” (podendo ser personalizado e comercializado para outras organizações).

# 3. ANÁLISE DE SISTEMAS ORIENTADA A OBJETOS

De acordo com Sommerville (2011), uma diferença importante entre o desenvolvimento de software profissional e amador é a complementação do programa em si com toda a documentação associada e dados de configurações necessários para fazer esse programa operar corretamente.

Apesar disso, alguns fundamentos se aplicam a todos os tipos de software, dentre eles está a especificação de requisitos, que também funciona como um gerenciamento de expectativas do cliente. Os requisitos de um sistema são as descrições do que um sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento (SOMMERVILLE, 2011).

Wazlawick (2014) expõe a necessidade de, além de organizar a estrutura de um sistema em camadas e módulos, organizar o código implementado dentro deles. Os programadores devem aprender as técnicas que podem levar à construção de código reusável e com baixo acoplamento.

Os conceitos básicos da UML (do inglês Unified Modeling Language) incluem diversos tipos de diagramas que representam diferentes aspectos do sistema em desenvolvimento. Esses diagramas podem ser agrupados em duas categorias principais: diagramas estruturais e diagramas comportamentais, como descrito por Booch et al. (2005).

Diagramas estruturais abordam diagrama de classes, diagrama de objetos, e diagrama de componentes; enquanto diagramas comportamentais seriam o diagrama de casos de uso, diagrama de sequência e diagrama de atividade, por exemplo (BOOCH et al., 2005).

Para Sommerville (2011), casos de uso são uma técnica de descoberta de requisitos, identificando os atores (que podem ser pessoas ou outro sistema) envolvidos em uma interação e nomeando-a. São documentados em seu diagrama com informações adicionais que descrevem a interação com o sistema.

Os casos de uso expandidos podem ser utilizados como comandos (métodos que são ativados por um evento de sistema, ou seja, como uma reação a uma ação do usuário) e consultas (verificação de informação já armazenada) de sistema. A sequência de eventos e respostas em um caso de uso são representados por um diagrama de sequência, com elementos como atores, interfaces do sistema e objetos (WAZLAWICK, 2014).

Já diagramas de classe serão definidos em diversos níveis de detalhamento. Em suma, mostra as classes (definição geral de um tipo de objeto do sistema) e as associações entre elas. Inicialmente os objetos representam algo do mundo real, entretanto enquanto uma aplicação é desenvolvida pode ser necessário definir objetos adicionais de implementação que são usados para fornecer a funcionalidade requerida do sistema. Por fim, os diagramas de implantação da UML mostram como os componentes de software são distribuídos em diferentes plataformas de hardware (SOMMERVILLE, 2005).

Para Wazlawick (2014), nem todos os diagramas devem ser usados durante o desenvolvimento de um sistema, apenas aqueles que representam informações úteis para o projeto são recomendados. Neste projeto foram desenvolvidos diagramas de caso de uso, sequência, classes e implementação.

# 4. ENGENHARIA DE SOFTWARE II

A Engenharia de Software é uma área da computação que trata da aplicação de princípios de engenharia no desenvolvimento, operação e manutenção de sistemas de software. Processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software (SOMMERVILLE, 2019). Envolve uma série de ações estratégicas e bem estruturadas com o objetivo de planejar, criar ou adquirir, testar e implementar o sistema.

A análise de requisitos em si é um processo que engloba o estudo das necessidades do usuário, para que uma definição correta/completa seja aplicada (COOPERSYSTEM, 2023). Os requisitos funcionais referem-se às principais funcionalidades que o software deve oferecer para cumprir seu propósito, isto é, as operações que o usuário precisa realizar. Já os requisitos não funcionais estão relacionados a aspectos como segurança, manutenção e desempenho do sistema.

Qualidade pode ser definida como um método gerencial que, através de procedimentos disseminados por toda a organização, busca garantir um produto que satisfaça às expectativas do cliente, dentro daquilo que foi acordado inicialmente (DEVMEDIA, 2025), como definido na ISO 9126, que rege características da qualidade de produtos de software.

É compreensível que sistemas de software estejam sujeitos aos mais variados tipos de erros e inconsistências. Para evitar que tais erros atinjam os usuários finais e causem prejuízos, é fundamental introduzir atividades de teste em projetos de desenvolvimento de software (VALENTE, 2022). A elaboração de uma tabela de testes permite organizar de maneira estruturada os itens que devem ser verificados, os procedimentos de execução e as evidências necessárias para comprovar o correto funcionamento do sistema. Em casos de falhas (dentre outros), o processo de alteração do software após a entrega é frequentemente chamado de manutenção de software (SOMMERVILLE, 2019). O programa precisa ser atualizado constantemente para corrigir erros, adaptar-se a novas tecnologias e atender a novas demandas.

Uma das alternativas para gerenciamento de equipes é a metodologia ágil SCRUM, que visa o planejamento e o acompanhamento das pessoas envolvidas no projeto, e cuja estrutura descreve um conjunto de reuniões, ferramentas e funções para uma entrega eficiente de projetos (AMAZON, 2025).

# 5. PROGRAMAÇÃO ORIENTADA A OBJETOS

Atualmente existem dois tipos de programação considerados usuais, a estruturada e a orientada a objetos. O primeiro paradigma é centrado em funções e no controle de fluxo com etapas lógicas lineares, enquanto o segundo foca em objetos e classes que interagem entre si através de mensagens, aproximando as estruturas de um programa à estrutura das coisas do mundo real (PROGRAMAE, 2024).

De acordo com dados históricos, os estudos sobre programação orientada a objetos surgiram no final da década de 60 no cenário da “crise do software”. Esse paradigma não possui um único criador, mas sabe-se que a linguagem Simula 67 foi a primeira a introduzir os conceitos de objetos, classes e herança. Esses conceitos foram aprimorados por meio da linguagem Smalltalk, desenvolvida no início da década de 70 pelo Alan Kay, conhecido como um dos pioneiros da POO, no Xerox PARC; mas só ganharam destaque a partir da década de 80 ao serem aplicados nas linguagens Java e C++ (METROPOLE, 2022).

Dentre as vantagens fornecidas por esse paradigma, encontram-se a modelagem do mundo real, que torna o desenvolvimento mais intuitivo e facilita a comunicação interna; a reutilização do código, pois através da herança e do polimorfismo é possível utilizar métodos e classes já existentes em outro contexto; a manutenibilidade simplificada, a possibilidade de desenvolvimento colaborativo, a extensibilidade e a rastreabilidade (BONATINI, 2023).

Em POO, as classes são estruturas que atuam como moldes ao agrupar itens com as mesmas propriedades e funcionalidades e por meio destas criam-se os objetos, que são as instâncias das classes. Os objetos terão os atributos, que correspondem às suas características; os métodos, que representam as ações e os comportamentos que eles poderão realizar e esses apresentam níveis de visibilidade, que determinam o acesso às informações, podendo ser público, privado ou protegido (NORMANDO, 2024).

Neste cenário surgem os seguintes conceitos cruciais, conhecidos como pilares da programação orientada a objetos: abstração, herança, polimorfismo e encapsulamento.

A abstração visa à simplificação ao modelar comportamentos e características concentrando-se apenas nas informações do objeto que são essenciais e relevantes dentro de um contexto. A herança fornece a funcionalidade da reutilização de métodos e atributos a partir de uma hierarquização de classes. O polimorfismo promove a capacidade de um objeto executar diferentes comportamentos a partir do mesmo método em situações distintas. E o encapsulamento visa controlar o acesso aos métodos e atributos da classe através dos níveis de visibilidade, permitindo a ocultação de dados e da estrutura; por meio desses conceitos, são gerados os benefícios da utilização do paradigma abordado (SILVA, 2023).

# 6. PROJETO DE INTERFACE COM O USUÁRIO

Segundo a ISO/IEC 9126, que aborda a temática de qualidade de produto de software, usabilidade se refere a capacidade deste de ser entendido, aprendido, operado e atraente para o usuário, quando este for usado sob determinadas condições específicas (Editora UX, 2024). Essa capacidade refletirá na experiência do usuário (UX), ou seja, nas sensações que existirão durante as interações entre o usuário e o sistema.

Neste cenário, a usabilidade é uma entre as seis principais características de qualidade. Junto a ela encontram-se a portabilidade, confiabilidade, eficiência, manutenibilidade e funcionalidade; isto influenciará diretamente na satisfação do usuário, aumentará a produtividade e a conquista e expansão no mercado. Para avaliar este conceito existem diversas técnicas, como por exemplo avaliações heurísticas e testes de usabilidade (CERQUEIRA, 2024).

A avaliação heurística consiste em uma análise prática e rápida de um produto, interface ou serviço; ela é realizada por um especialista em UX baseando-se nas heurísticas de Nielsen. Essas heurísticas abordam aspectos como a eficiência e a importância dos feedbacks, as "saídas de emergência" para ações indesejadas, a padronização conforme o setor, o uso de modelos conceituais, a clareza das mensagens de erro, a flexibilidade nas formas de realizar ações, entre outros aspectos. E, como citado pelo próprio Jakob Nielsen, as 10 heurísticas em si permaneceram relevantes e inalteradas desde 1994 (TEIXEIRA, 2016).

Em contrapartida, o teste de usabilidade consiste em verificar a consistência e qualidade de um site, aplicativo ou produto através das reações e do comportamento do usuário na naturalidade da utilização; esses usuários são personas que refletem o público-alvo e a partir deste é possível compreender se os requisitos levantados durante a elicitação na análise estão sendo atendidos. Este modelo de teste destaca oportunidades de melhorias no funcionamento, auxilia na otimização da experiência, avalia a velocidade e a facilidade de acesso às informações e a preferência do usuário em relação à forma de como utilizará o sistema (WOEBCKEN, 2021).

Neste contexto, surge o conceito de protótipo que visa identificar e testar a viabilidade técnica de um produto, apresentando uma versão simplificada do que será desenvolvido e podendo ser de baixa, média ou alta fidelidade, além de ser físico ou digital. Ressalta-se que este não é funcional, portanto, não pode ser comercializado.  Além do protótipo, também é possível desenvolver um MVP, Mínimo Produto Viável, este irá se concentrar em analisar a viabilidade do mercado; ele é desenvolvido com requisitos mínimos e pode ser comercializado (CONDE, 2022).

Em suma, a usabilidade é um fator essencial quando analisado pela perspectiva da experiência do usuário, proporcionando um sistema funcional e confiável e gerando a fidelização dos clientes, trazendo como resultado ascensão da marca no mercado.

# 7. BANCO DE DADOS

Todo sistema quando gerado necessita de um local para que os dados futuramente armazenados nele sejam guardados de forma segura, podendo assim manter a integridade e a segurança do software. Esse local é chamado de Banco de Dados, uma forma de armazenar, virtualmente, dados de um sistema que, quando ligado a um Sistema de Gerenciamento de Banco de Dados, fornecem qualidade ao sistema no qual estão ligados, fornecendo as ferramentas necessárias para manutenção e controle das informações e melhora a organização das informações (ORACLE, 2025).

Existem dois tipos de banco de dados: os relacionais e os não relacionais. Os não relacionais não possuem uma estrutura definida, ou seja, é possível guardar informações de diversas formas, como por exemplo por documentos e gráficos. Já o relacional, que é utilizado neste projeto, possui forma, sendo distribuída em linhas e colunas que formam tabelas. Cada tabela é uma entidade e cada linha um registro, sendo criado e manipulado por meio da linguagem SQL e possuindo maior eficiência para o acesso de informações (ORACLE, 2025).

Quando se cria uma entidade para o banco de dados, são especificados atributos dela, como por exemplo “id\_cliente”. Esses atributos são dispostos em colunas no que é chamado de Dicionário de dados, uma tabela que possui várias informações de cada atributo, sendo importante como referência para os desenvolvedores daquele projeto, visando tornar a comunicação mais fácil entre a equipe e diminuindo erros. Esses dicionários são responsáveis por documentar exatamente as informações sobre cada campo, ou atributo, fornecendo uma visão detalhada da estrutura dos dados existentes e facilitando a geração de um script do banco de dados em linguagem SQL. “Uma das importâncias de um DD (Dicionário de Dados) se dá pelo fato de muitas vezes na atividade de modelagem de um PBD (Projeto de Banco de Dados), os diagramas não suprirem informações significativas sobre os dados que devem ser tratados, deixando a interpretação desses diagramas ineficientes, podendo gerar ambiguidades em sua implementação” (BARBOSA, 2012).

Além dele, existe a normalização, é uma forma de normalizar as tabelas para que não ocorram erros durante a elaboração do banco de dados. Existem 6 normalizações, porém as mais utilizadas e necessárias são as três primeiras formas normais, onde a primeira forma normal busca remover atributos que são multivalorados ou compostos, a segunda forma, eliminar dependências parciais, que é quando um atributo depende apenas de uma das Primary Keys, não da chave composta, e existe a terceira, que visa eliminar dependências transitivas, quando um atributo que não é uma chave depende de outro atributo não-chave. Após o processo de normalização, são gerados diagramas de entidade relacionamento (PUIG, 2023).

“Um diagrama entidade relacionamento (ER) é um tipo de fluxograma que ilustra como “entidades”, pessoas, objetos ou conceitos, se relacionam entre si dentro de um sistema” (SILVA, 2021). Eles permitem a criação de scripts de forma mais assertiva e segura, para que posteriormente sejam melhorados e utilizados no SQL server. Existem três tipos de diagrama ER: o conceitual descritivo, que mostra as os atributos de cada tabela, descreve o relacionamento e suas cardinalidades, o conceitual em forma de diagrama, que irá dispor das informações do formato descritivo, porém em imagem, e o lógico, que é o diagrama ER em formato visual porém com as especificações de cada tributo, como por exemplo o tipo, e as chaves existentes em cada.

# 8. ECONOMIA E MERCADO

A análise de economia e mercado em um projeto pode ser empregada em um estudo da estimativa de viabilidade. Conforme definido por Riberiro (2005), estimar consiste em uma avaliação sobre alguém ou sobre alguma circunstância, baseando-se nas evidências ou nos fatos disponíveis; conjectura.

A estimativa de viabilidade analisa se será possível realizar um projeto com base em alguns tópicos, como: estimativa de tempo, estimativa de riscos, Retorno sobre Investimento (ROI), definição de objetivos e análise de requisitos, trazendo uma melhor visão geral do tempo hábil (BORSOI, 2015). Os requisitos foram estabelecidos e estão disponíveis no tópico X, página Y desse projeto. O ROI estabelece e avalia o resultado financeiro de uma empresa ou pessoa, fazendo a relação entre os gastos e lucros, dizendo então se é viável ou não seguir por tal caminho. “Dessa forma, é possível comparar resultados e definir rapidamente os investimentos que valem a pena e aqueles que devem ser repensados ou abandonados” (SEBRAE, 2023).

Dentre as diversas metodologias de planejamento de projetos, para este foi utilizado o método SCRUM, uma ferramenta ágil que gerencia projetos e potencializa o trabalho em equipe de um projeto, possibilitando que todos os integrantes do grupo saibam exatamente o que está acontecendo em cada parte do projeto de períodos em períodos, já que se trata uma ferramenta utilizada em trabalhos de alta complexidade. Por ser um framework, serve como um guia de boas práticas para atingir o sucesso (PEREIRA, 2007).

Com base na análise prévia ao desenvolvimento do sistema, ao tempo hábil estabelecido para realização do projeto, acorda-se a possibilidade de realizar tal projeto sem exceder a data limite.

# 9. GESTÃO ESTRATÉGICA DE RECURSOS HUMANOS

O setor de Recursos Humanos, além de atuar nas atividades de recrutamento e seleção de profissionais qualificados, é encarregado pela mediação das relações entre o negócio e os colaboradores.

Quando se pensa em RH, logo se imagina o modelo tradicional, onde o foco está nas tarefas operacionais e administrativas. Esse departamento tem como principais atividades o recrutamento, seleção e demissão de funcionários, realização de treinamento e suporte na elaboração da folha de pagamento. Sua principal característica é o planejamento limitado e voltado ao curto prazo, é um setor que tende a ser mais reativo, ou seja, age após o surgimento do problema. Como exemplo, a rotatividade de funcionários, quando uma seção da empresa está com um fraco desempenho, a primeira atitude é identificar o funcionário responsável e o substituí-lo, ao invés de averiguar a origem do problema. Esse modelo costuma estar pouco ajustado às estratégicas da empresa (SERASA EXPERIAN, 2023).

Entretanto, o RH estratégico busca alinhar as metas da empresa com os interesses profissionais dos colaboradores, sempre priorizando as pessoas. Ao adotar metodologias voltadas para os objetivos do negócio, o RH deixa de ter apenas uma característica operacional e passa se tornar uma parte fundamental para o desenvolvimento da empresa. O ideal é que haja um alinhamento entres setores, para que o RH não atue de maneira isolada, assim a organização auxilia o RH a ter uma participação mais ativa e relevante na empresa. Seu perfil é proativo, prioriza o desenvolvimento dos funcionários e a retenção de talentos (LECOM, 2023). Utiliza a tecnologia para otimizar seus processos, buscando eficiência e sugere inovações para cada setor da empresa (TWYGO, 2022). Assim, o RH estratégico se consagra um ator transformador dentro da organização, construindo uma cultura organizacional estratégica, inovadora e focada no crescimento contínuo.

Portanto, é importante que o RH compreenda os objetivos da organização e defina qual vai ser o papel do departamento em cada um deles, para que trace um bom plano estratégico que estipule metas de médio a longo prazo. A elaboração de um plano estratégico bem estruturado, juntamente com a capacitação contínua dos funcionários, através de treinamentos, desenvolvimento de competências e programas de formação, valoriza o capital humano da empresa, trazendo mais resultados e inovação.

# 10. DESENVOLVIMENTO

Após a caracterização do cenário (empresa Esfera Contabilidade) onde o sistema de gerenciamento de chamados estará inserido, partiu-se para a elicitação de requisitos funcionais e não funcionais. Para tanto, foram realizadas pesquisas bibliográficas, consultas em versões gratuitas de softwares do ramo já estabelecidos no mercado e entrevistas com profissionais da área de suporte (conforme consta na Atividade de Extensão em anexo), buscando definir quais *features* seriam essenciais, como inserir a atividade da Inteligência Artificial no processo e o quais seriam as etapas do processo na experiência dos usuários.  
 Com isso, utilizando o software Astah UML versão 10.0.0 (lançado em

Tabela 1 – Product Backlog

|  |  |  |  |
| --- | --- | --- | --- |
| **REQUISITOS FUNCIONAIS** | | **REQUISITOS NÃO FUNCIONAIS** | |
| RF001 | Autenticar Usuário | RNF0001 | Reconhecimento e prevenção de erros |
| RF002 | Abrir Chamado | RNF0002 | Ajuda e documentação |
| RF003 | Preencher Formulário | RNF0003 | Manutenibilidade e rastreabilidade |
| RF004 | Avaliar Solução | RNF0004 | Usabilidade |
| RF005 | Verificar Chamado | RNF0005 | Hierarquia de acesso |
| RF006 | Cancelar Chamado | RNF0006 | Implementação de testes |
| RF007 | Categorizar Chamado |  |  |
| RF008 | Sugerir Solução |  |  |
| RF009 | Priorizar Chamado |  |  |
| RF010 | Encaminhar Chamado |  |  |
| RF011 | Diagnosticar Chamado |  |  |
| RF012 | Confirmar Solução |  |  |
| RF013 | Excluir Chamado |  |  |
| RF014 | Gerar Relatório Mensal |  |  |
| RF015 | Gerar Relatório Semestral |  |  |
| RF016 | Gerar Relatório Anual |  |  |
| RF017 | Exportar Relatório |  |  |
| RF018 | Editar Usuário |  |  |
| RF019 | Adicionar Usuário |  |  |
| RF020 | Excluir Usuário |  |  |

Fonte: autoria própria, 2025.

Junto do Product Backlog desenvolveu-se o primeiro diagrama UML, de caso de uso, representando os pacotes de funcionalidades, atores e atividades exercidas.

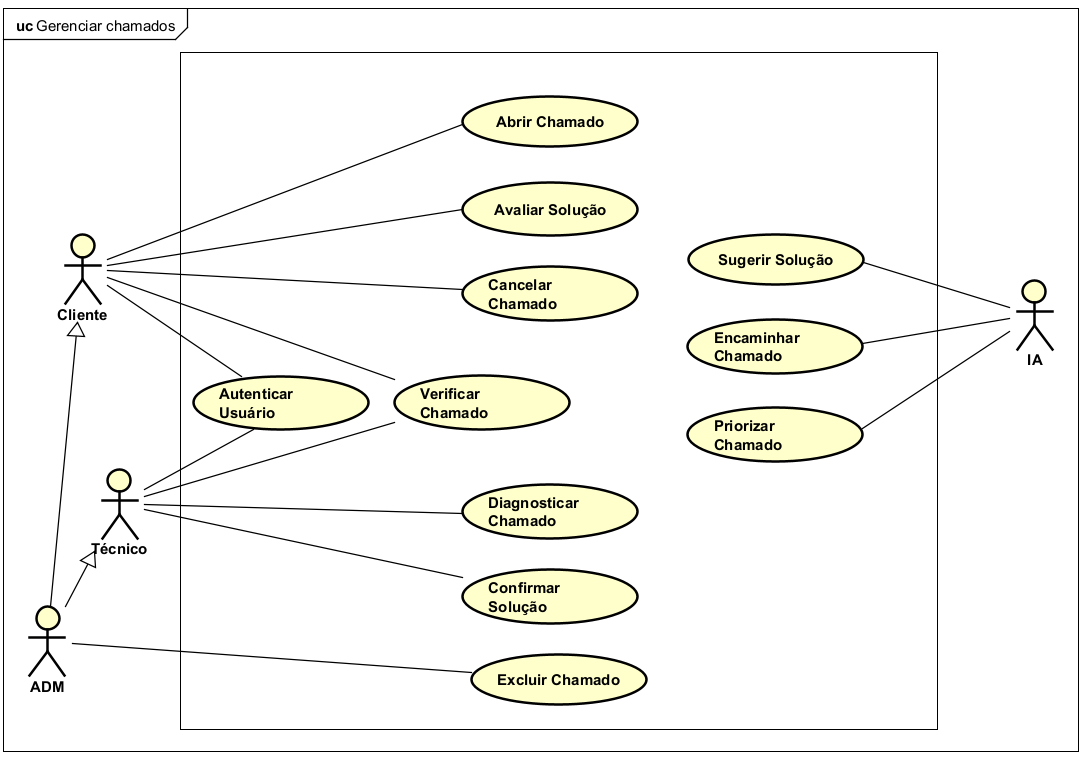
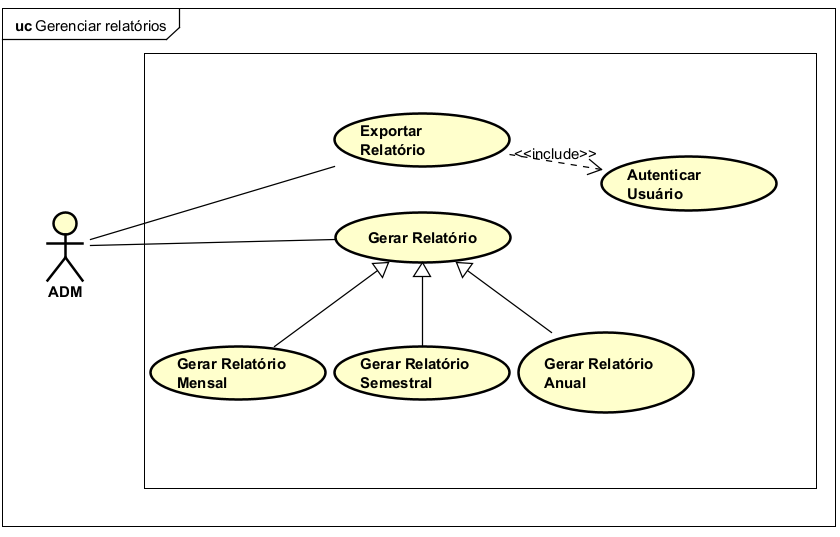
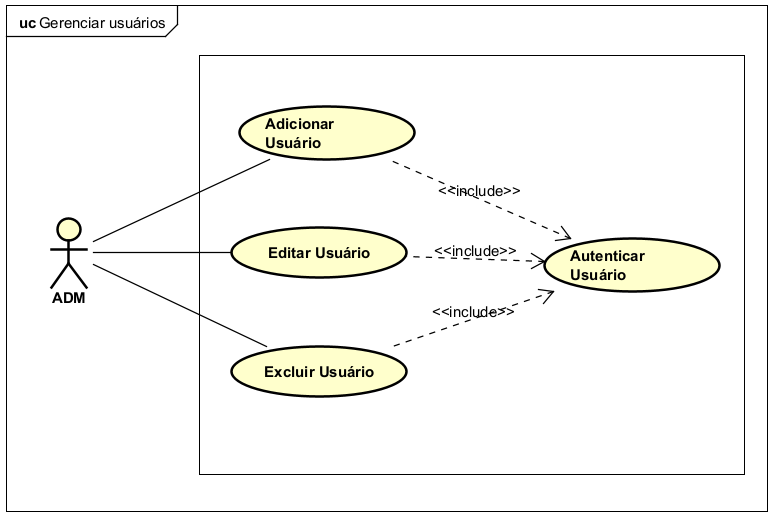
Figura 1 – Diagrama de caso de uso, pacote “Gerenciar Chamados”Fonte: autoria própria, 2025.

Figura 2 – Diagrama de caso de uso, pacote “Gerenciar Relatório

Fonte: autoria própria, 2025

Figura 3 – Diagrama de caso de uso, pacote “Gerenciar Usuários”

Fonte: autoria própria, 2025.

Um caso de uso descritivo é uma forma de documentação na engenharia de software que apresenta, em texto, o comportamento esperado do sistema durante sua interação com usuários ou outros sistemas. Com isso, foram feitos 3 casos de uso descritivos:

Tabela 2 – Caso de uso “Abrir Chamados” descritivo.

|  |  |
| --- | --- |
| **Caso de uso:** | **Abrir chamados** |
| **Ator:** | Cliente |
| **Pré-condições:** | O cliente deve estar autenticado no sistema |
|  | O cliente deve ter acesso à funcionalidade “Abrir Chamado” |
| **Fluxo normal:** | 1. Autenticar usuário |
|  | 1. Cliente seleciona opção “Abrir Chamado” na tela inicial |
|  | 1. Cliente preenche formulário com o máximo de detalhes possíveis |
|  | 1. Cliente confirma envio do formulário |
|  | 1. IA fornece sugestão rápida para o problema informado |
|  | 1. Cliente avalia solução oferecida pela IA |
| **Extensões:** | 1a. Se dados incorretos, solicitar nova tentativa de autenticação |
|  | 1b. Se cliente não recorda seus dados de autenticação, orientar cliente para entrar em contato com o administrador |
|  | 4a. Se formulário vazio ou conter caracteres insuficientes, solicitar novo preenchimento de acordo com a regra não cumprida |
|  | 4b. Se cliente não confirmar envio do formulário, retornar para a tela inicial |
|  | 6a. Se a solução foi útil, chamado é encerrado |
|  | 6b. Se a solução não foi eficiente, IA fará encaminhamento para o técnico do setor. |
| **Pós-condições:** | Um novo chamado é registrado no sistema |
|  | O cliente recebe uma sugestão automática ou tem o chamado encaminhado para análise técnica |

Fonte: autoria própria, 2025.

Tabela 3 – Caso de uso “Verificar Chamado” descritivo.

|  |  |
| --- | --- |
| **Caso de uso:** | **Verificar Chamado (Prioridade)** |
| **Ator:** | Técnico |
| **Pré-condições:** | O técnico deve estar autenticado no sistema. |
|  | Deve haver chamados registrados com níveis de prioridade. |
|  | O técnico deve ter acesso à funcionalidade “Verificar Chamado” |
| **Fluxo normal:** | Autenticar usuário |
|  | Técnico seleciona opção “Verificar Chamado” na tela inicial |
|  | Técnico seleciona opção “Filtrar” na tela de verificar chamado |
|  | Técnico seleciona opção “Prioridade” na função “Filtrar” |
|  | Lista retorna com filtro de prioridade |
|  | Técnico fornece resposta do chamado |
|  | Cliente avalia solução oferecida pelo técnico |
| **Extensões:** | 1a. Se dados incorretos, solicitar nova tentativa de autenticação |
|  | 1b. Se técnico não recorda seus dados de autenticação, orientar cliente para entrar em contato com o administrado |
|  | 7a. Se a solução foi útil, chamado é encerrado |
| **Pós-condições:** | O chamado recebe uma resposta |
|  | O cliente é notificado da resposta e pode avaliá-la |
|  | O chamado pode ser encerrado ou seguir em aberto, conforme avaliação do cliente |

Fonte: autoria própria, 2025.

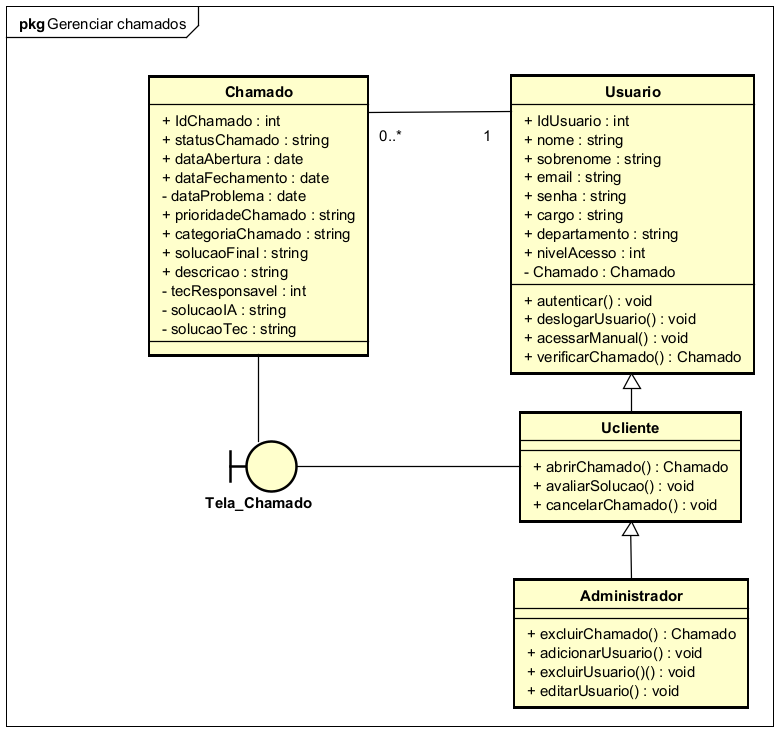
Tabela 4 – Caso de uso “Autenticação de Usuário” descritivo.

|  |  |
| --- | --- |
| **Caso de uso:** | **Autenticação de Usuário** |
| **Ator:** | Cliente e técnico |
| **Pré-condições:** | O sistema deve estar acessível |
|  | O ator deve ter um cadastro válido (e-mail e senha) no sistema |
| **Fluxo normal:** | Abrir Sistema |
|  | Ator preenche e-mail na tela de autenticação |
|  | Ator preenche senha na tela de autenticação |
|  | Ator confirma autenticação |
|  | Abre tela de início do sistema |
| **Extensões** | 1a. Sistema não abre, orientar ator para entrar em contato com o administrador |
|  | 2a. Se e-mail incorreto, solicitar nova tentativa de autenticação |
|  | 2b. Se ator não recorda seu e-mail de autenticação, orientar ator para entrar em contato com o administrador |
|  | 3a. Se senha incorreta, solicitar nova tentativa de autenticação |
|  | 3b. Se ator não recorda sua senha de autenticação, orientar ator para entrar em contato com o administrador |
| **Pós-condições:** | O ator tem acesso ao sistema autenticado, com funcionalidades liberadas |
|  | Caso não autenticado, permanece na tela de login ou é orientado a recuperar os dados |

Fonte: autoria própria, 2025.

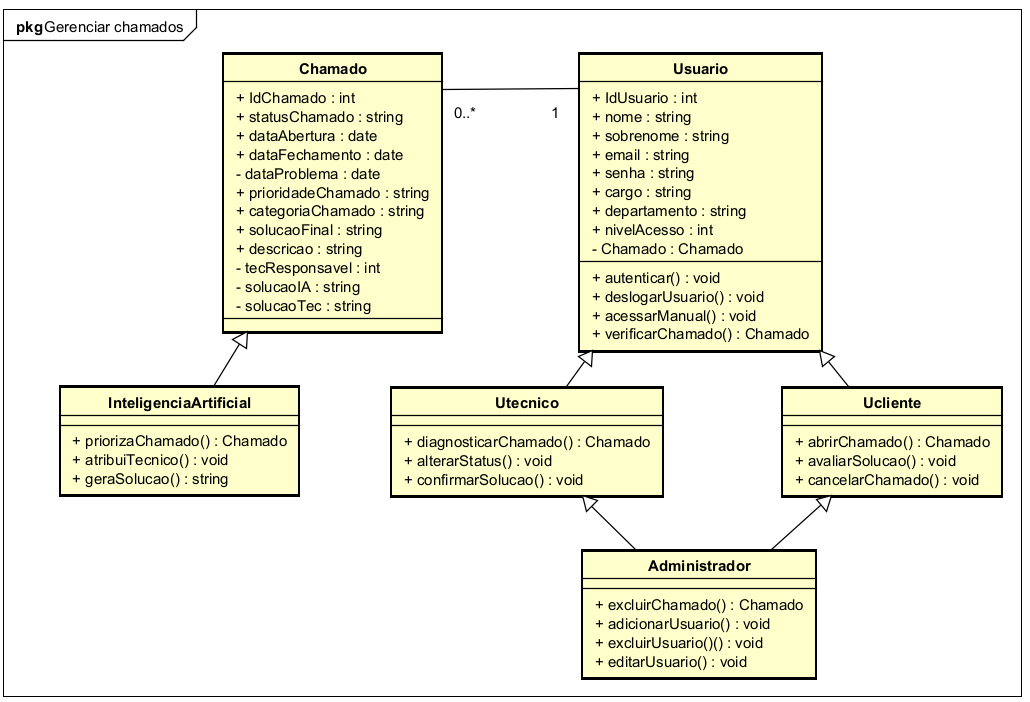
Deste ponto cada caso de uso foi submetido à análise para compreender quais interfaces e classes estariam envolvidas naquela ação, como elas estarão relacionadas e quais seriam seus métodos e atributos. Com isto, tem-se um guia para uma posterior codificação organizada. Estas informações são consolidadas em diagramas de classe para cada caso de uso, como nos dois exemplos abaixo:

Figura 4 – Diagrama de classes para o caso de uso Abrir Chamado

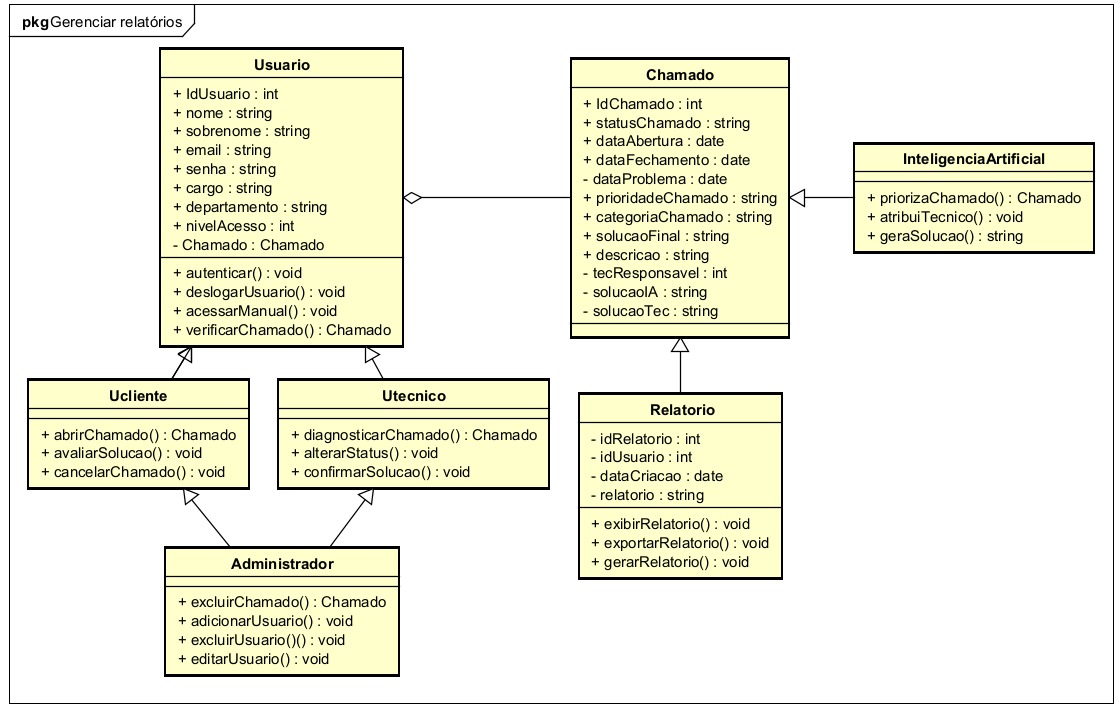


Fonte: autoria própria, 2025.

Figura 5 – Diagrama de classes para o caso de uso Sugerir Solução

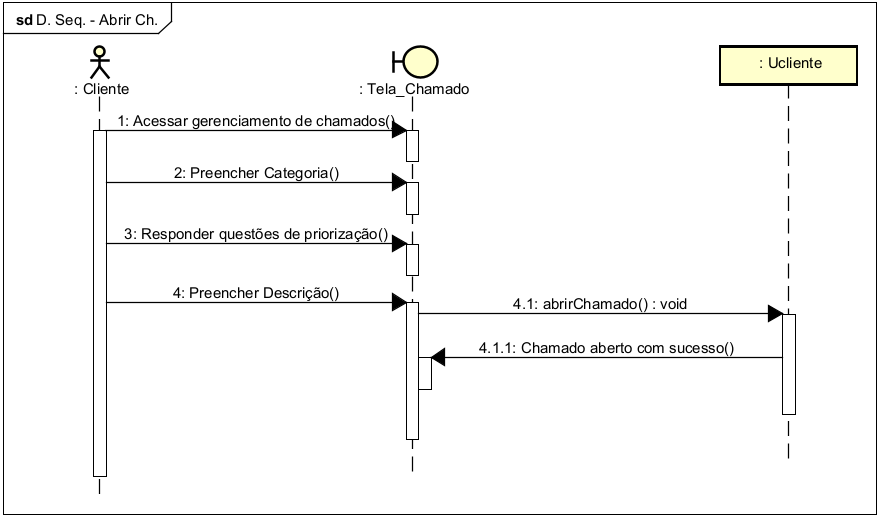
Fonte: autoria própria, 2025.

Todos estes diagramas de classe oriundos de cada caso de uso também se reúnem em um diagrama de classes geral, que contém todos os elementos e associações, conforme figura:

Figura 6 – Diagrama de classes completo.Fonte: autoria própria, 2025.

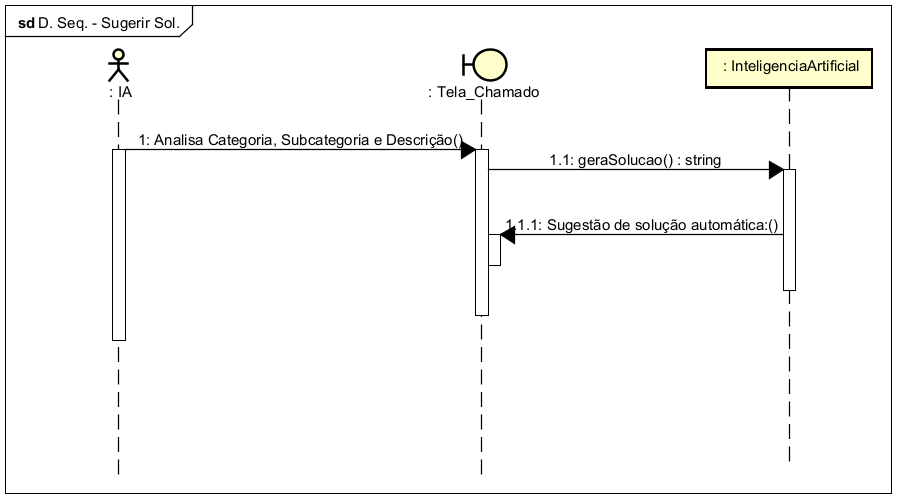
Partindo das classes e interfaces definidas para cada caso de uso, também foram construídos diagramas que sequência que demonstram as interações dos atores com estes e suas devidas respostas. Nas figuras estão exemplos para os mesmos casos de uso utilizados nos diagramas de classe anteriores:

Figura 7 – Diagrama de Sequência para o caso de uso Abrir Chamado.



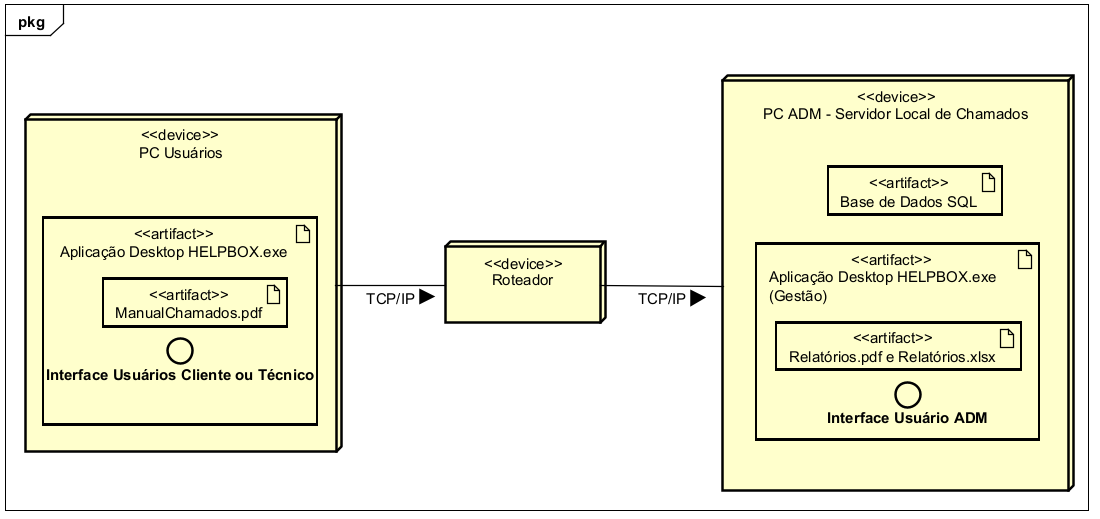
Fonte: autoria própria, 2025.

Figura 8 – Diagrama de Sequência para o caso de uso Sugerir Solução.

Fonte: autoria própria, 2025.

Por fim, descrevendo todos os componentes necessários para funcionamento do software em seus dispositivos - computadores, servidores, roteadores, protocolos de rede, artefatos – arquivos executáveis, documentos internos ou exportáveis, banco de dados, suas associações e interfaces, produziu-se um diagrama de implantação único.

Figura 9 – Diagrama de implantação.

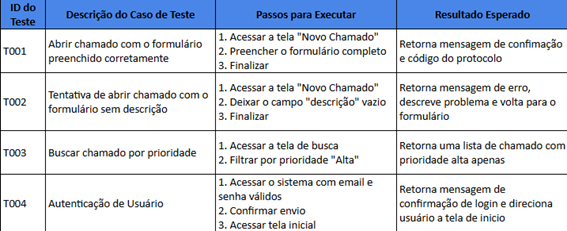
Fonte: autoria própria, 2025.

Em complemento, a elaboração de uma tabela de testes permite organizar de maneira estruturada os itens que devem ser verificados, os procedimentos de execução e as evidências necessárias para comprovar o correto funcionamento do sistema. Essa prática assegura a validação das principais funcionalidades antes da disponibilização do sistema em ambiente produtivo, mitigando riscos de falhas que poderiam resultar em prejuízos, retrabalho ou insatisfação dos usuários. Ademais, a documentação das evidências e das consultas realizadas no banco de dados torna o processo de homologação mais eficiente, seguro e transparente, possibilitando o acompanhamento claro do progresso dos testes por toda a equipe envolvida.

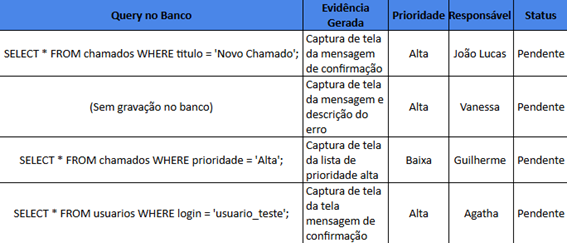
Como primeira parte da tabela temos a numeração do teste, sendo colocado como ID, a descrição de teste que diz qual a ação a ser testada, os passos que deverão ser feitos e o resultado esperado, esse sendo uma especulação do que pode vir a acontecer.

A segunda parte são os comandos ou instruções escritas em linguagem SQL, o envolvimento do banco de dados, as evidências geradas, armazenados como prova do teste, a prioridade do teste a ser feito, o responsável pelo teste e por fim o status.

Figura 10 – Primeira parte da planilha de testes.



Fonte: Autoria própria, 2025.

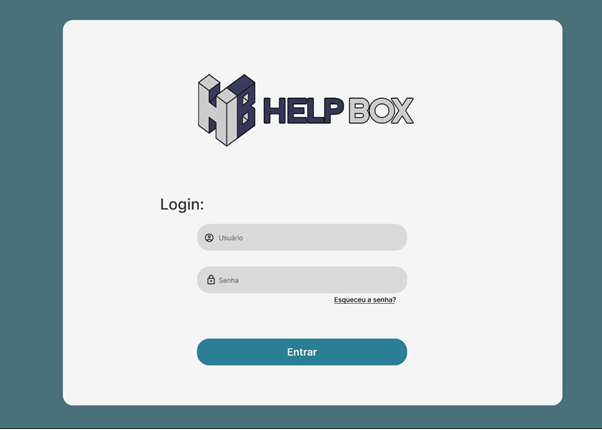
Figura 11 – Segunda parte da planilha de testes.

Fonte: Autoria própria, 2025.

O protótipo do sistema foi desenvolvido na versão DESKTOP, tamanho 1440 x 1024 no Figma e o objetivo foi demonstrar as funcionalidades do sistema a partir do usuário master, que seguindo as regras de negócio, é o administrador. A intenção é que o usuário master tenha acesso a todas as funcionalidades como o maior nível de acesso ao sistema.

A primeira tela que todos os usuários terão acesso será a de login, pois visando ser um sistema interno de uma empresa, as funcionalidades estarão disponíveis apenas para usuários autenticados, sendo este requisito demonstrado como essencial no diagrama de caso de uso. Caso o usuário esqueça ou perca a senha, ao selecionar a opção, será informado que essa alteração deve ser solicitada ao administrador, visando proteção de dados.

Figura 12 - Protótipo: tela de autenticação.

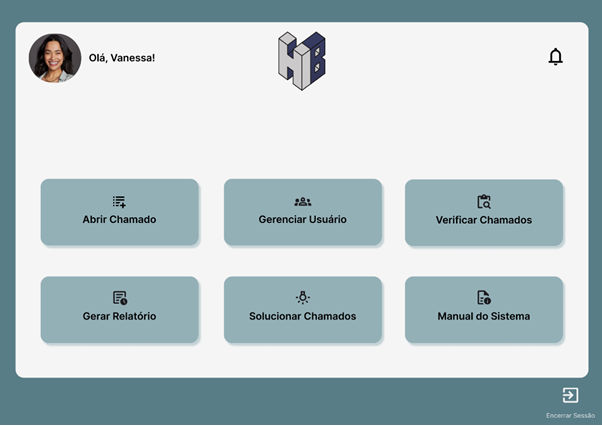
Fonte: autoria própria, 2025.

As funções disponíveis para o administrador são a “Abertura de Chamados” na qual ele preencherá um formulário dividido em 3 partes que visa compreender de maneira efetiva o chamado. Após o envio do formulário, o usuário receberá uma solução inicial gerada automaticamente pela inteligência artificial e terá de dar uma devolutiva se esta foi uma resolução eficiente ou não, em caso de não ter sido, haverá encaminhamento do chamado para um técnico específico.

A função de “Gerenciar Usuário”, na qual o administrador poderá adicionar, excluir ou alterar as informações do usuário. A função de “Verificar Chamados” na qual é possível ter acesso aos chamados abertos e também ao histórico de chamados. A função “Gerar Relatório” na qual seleciona-se os filtros para a construção de um relatório que pode ser exportado em formato PDF.

Há a função “Solucionar Chamados” na qual é possível verificar todos os chamados que estão abertos, visualizar todas as informações sobre ele (incluindo solicitante e solução gerada pela IA) e então solucioná-lo através desta função. Além disso, o usuário consegue ter acesso ao seu perfil ao selecionar a foto no canto superior esquerdo e também visualizar o manual do usuário através do botão “Manual do Usuário” na tela inicial e nas outras telas, há um botão que representa o manual também caso o usuário precise.

Figura 13 – Protótipo: tela inicial



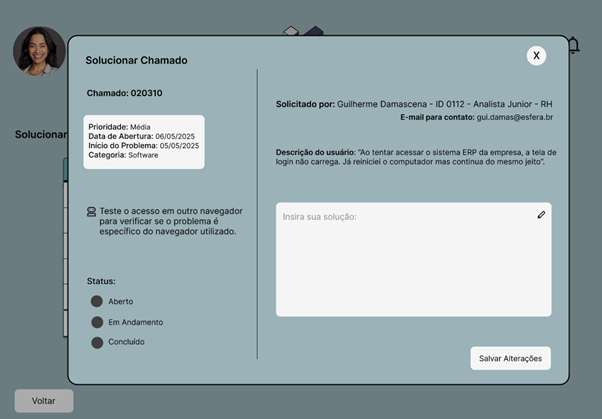
Fonte: autoria própria, 2025.

Figura 14 – Protótipo: tela de sugestão fornecida pela IA



Fonte: autoria própria, 2025.

Figura 15 – Protótipo: tela de resolução de chamado



Fonte: autoria própria, 2025.

O protótipo foi construído visando as heurísticas de Nielsen e prezando pela usabilidade, portanto há prevenção de erros: para todas as ações críticas no sistema, há um pedido de confirmação da ação e também há ajuda e documentação através do manual. O usuário Cliente terá acesso às funções de “Abrir Chamado”, “Verificar Chamado” e Manual do Usuário. Enquanto o usuário Técnico terá acesso às funções de “Solucionar Chamados” e Manual do Usuário.

Para início da modelagem do banco de dados foram definidas as seguintes regras de negócio:

1. O Administrador é um usuário master.
2. O Cliente pode somente gerenciar seus próprios chamados.
3. O Técnico pode somente gerenciar resoluções atribuídas a ele.
4. Somente o administrador pode gerenciar usuários e relatórios.
5. Os atributos do Chamado não podem ser nulos.
6. A priorização dos chamados será definida na abertura (operação da IA).
7. A priorização será definida pela pontuação em três campos do formulário, pela seguinte lógica:
   1. Ocorre ocasionalmente ou de maneira contínua?
      1. Ocasionalmente (+1 ponto)
      2. Continuadamente (+3 pontos)
   2. Qual o impacto na demanda?
      1. Impede a execução do trabalho (+3 pontos)
      2. Causa atraso, mas trabalho continua (+2 pontos)
      3. Impacto mínimo e sem prejuízos operacionais (+1 pontos)
   3. Ocorre com todos os usuários ou apenas com você?
      1. Todos os usuários (+3 pontos)
      2. Atinge apenas um grupo específico (+2 pontos)
      3. Apenas comigo (+1 ponto)

Resultado da somatória de pontos:

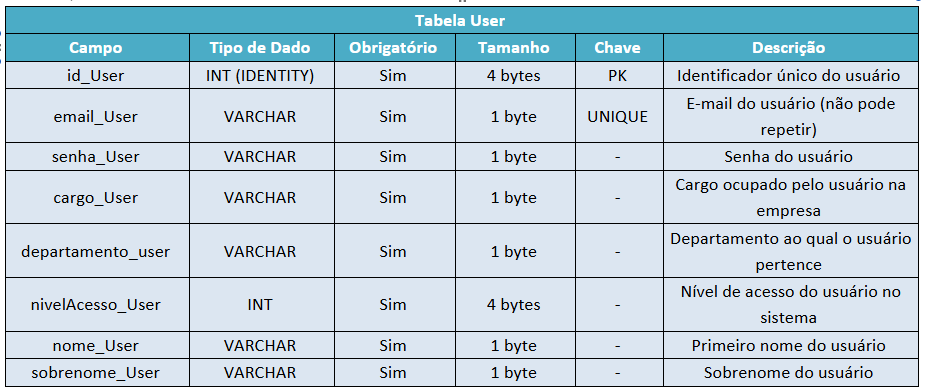
* Alta prioridade: A partir de 7 pontos
* Média prioridade: 4 a 6 pontos
* Baixa prioridade: 3 pontos

1. Os usuários (Cliente ou Administrador) podem ter mais de um chamado em andamento simultaneamente.
2. Os usuários (Técnico ou Administrador) podem resolver mais de um chamado simultaneamente.
3. Todo Chamado novo possui uma solução automática inicial (resposta da IA).
4. Se a solução automática não atender, Chamado será atribuído ao Técnico (operação da IA).
5. Toda resolução de chamado pode gerar uma nova solução automática (alimentação da IA).
6. É obrigatória a justificativa para chamados cancelados ou excluídos.

Com base nas regras de negócio e nos requisitos pré-estabelecidos, foram criadas as tabelas para cada entidade disposta no banco de dados físico, sendo estas: Tabela User, Tabela Cliente, Tabela técnico, Tabela ADM, Tabela Chamado e Tabela Relatório.

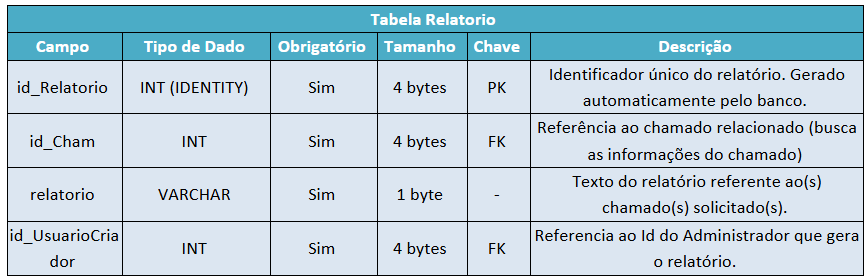
As tabelas foram formalizadas e atendem as 3 primeiras formas normais, ou seja, não há atributos multivalorados, sem dependência da Primary Key ou de dependência transitiva. Após a formalização, foi feito o dicionário de dados de cada tabela, contendo as informações do nome do atributo, tipo de dados, tamanho do dado em bytes, obrigatoriedade, tipo de chave e descrição. A formalização e o dicionário de dados foram feitas para todas as tabelas, porém serão dispostas apenas algumas como exemplo, sendo as mais importantes:

**Tabela 5 – Tabela User**

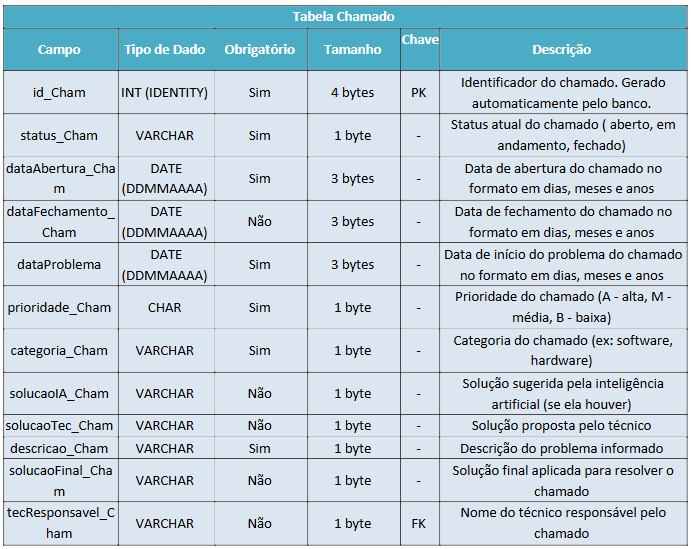


Fonte: autoria própria, 2025.

**Tabela 6- Tabela Relatório.**

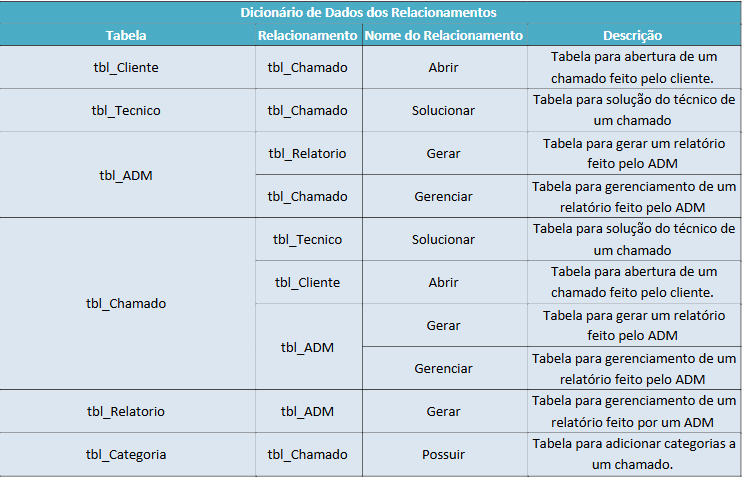
Fonte: autoria própria, 2025.

**Tabela 7 – Tabela Chamado.**

Fonte: autoria própria, 2025.

Além do dicionário de dados de cada tabela, foi feito o dicionário de dados para cada relacionamento: os relacionamentos entre as entidades e o relacionamento entre os relacionamentos:

**Tabela 8 – Relacionamento entre entidades.**

Fonte: autoria própria, 2025.

**Tabela 9 – Relacionamento entre relacionamentos.**Fonte: autoria própria, 2025.

Após o processo de criação de tabelas, foram desenvolvidos os três tipos diagrama de entidade-relacionamento: o conceitual descritivo, o conceitual e o lógico. O conceitual descritivo foi feito para cada tabela e cada relacionamento, sendo este:

**Entidade: User. -** **Atributos:** id\_User (Primary Key), nome\_User, sobrenome\_User, email\_User, senha\_User, cargo\_User, departamento\_User, nívelAcesso\_User, nome\_User

**Entidade Cliente - Atributos:** id\_User (Primary Key e Foreign Key de User).

**Entidade Técnico - Atributos:** id\_User (Primary Key e Foreign Key de User).

**Entidade ADM - Atributos:** id\_User (Primary Key e Foreign Key de User).

**Entidade Chamado -** **Atributos:** id\_Cham (Primary Key), dataProblema, tecResponsavel\_Cham (Foreign Key), solucaoFinal\_Cham, descricao\_Cham, solucaoTec\_Cham, categoria\_Cham, prioridade\_Cham, dataFechamento\_Cham, dataAbertura\_Cham, status\_Cham.

**Entidade** **Relatório -** **Atributos:** id\_Relatorio (Primary Key), id\_Cham (Foreign Key), relatorio, id\_UsuarioCriador.

**Relacionamentos:**

**Abrir (entre Cliente e Chamado):**

Descrição: Um cliente pode abrir nenhum ou vários chamados;

Um chamado é aberto por um único cliente;

Cardinalidade: 1:N (Cliente - Chamado).

**Solucionar (entre Técnico e Chamado)**

Descrição: Um técnico pode solucionar nenhum ou vários chamados;

Um chamado é solucionado por um único técnico;

Cardinalidade: 1:N (Técnico - Chamado)

**Gerar (entre ADM e Relatório)**

Descrição: Um ADM pode gerar nenhum ou mais relatórios;

Um relatório é gerado por um ADM;

Cardinalidade: 1:N (ADM - Relatório).

**Gerencia (entre ADM e Chamado)**

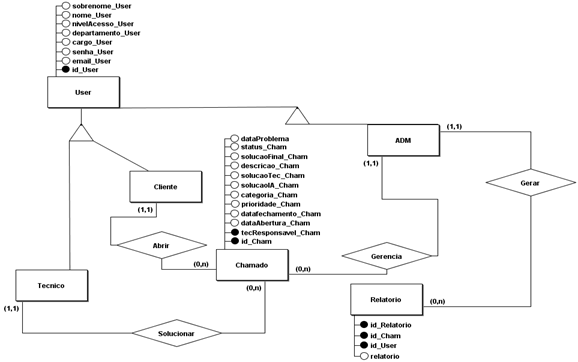
Descrição: Um ADM pode gerenciar nenhum ou mais chamados;

Cada ADM pode gerenciar um chamado;

Cardinalidade: 1:1 (ADM - Chamado).

O DER conceitual visual deve seguir as informações à risca do conceitual descritivo, e após a elaboração do mesmo no aplicativo brModelo, obteve-se tal imagem:

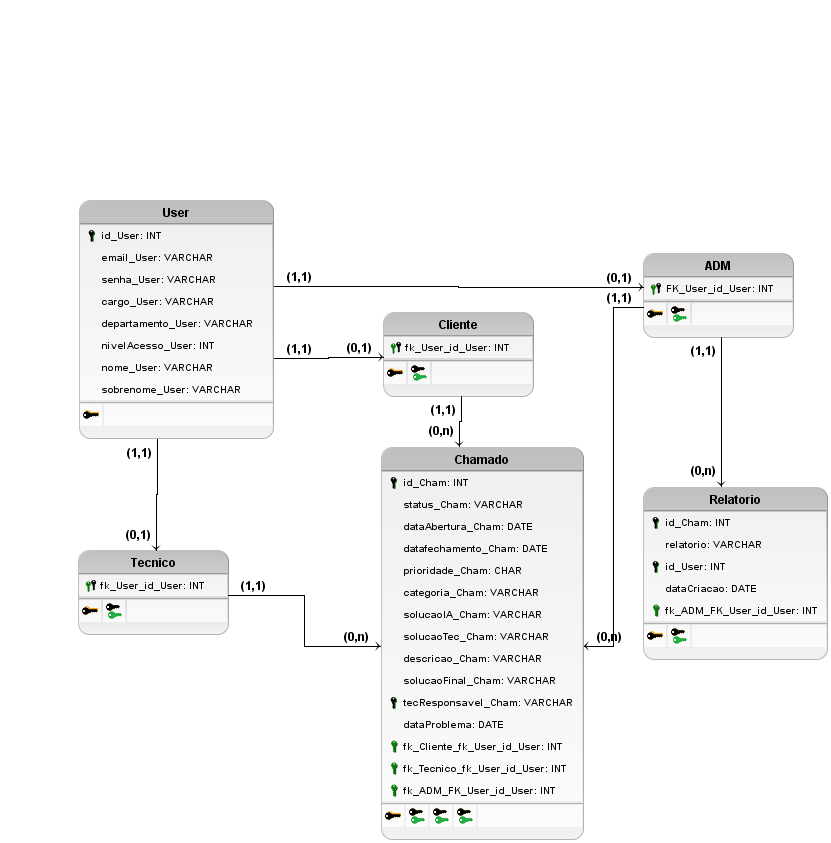
Figura 16 – Diagrama de Entidade-Relacionamento Conceitual Visual.



Fonte: autoria própria, 2025.

Com o DER Conceitual em mãos, foi gerado o modelo lógico, facilitando uma posterior criação do banco de dados físicos, utilizando o SQL Server:

Figura 17 –Diagrama de Entidade-Relacionamento Lógico.

Fonte: autoria própria, 2025.

Todas as etapas anteriores são primordiais para a criação de um código em linguagem SQL, visto que realizando esses processos, haverá uma padronização e formalização do banco de dados, evitando que haja erros no processo de criação ou de execução. Quando um defeito é identificado de forma precoce em uma dar etapas anteriores, é simples ajustar o que estava equivocado e seguir o processo de uma forma linear.

Por fim, foi desenvolvido a parte física do banco de dados: o código em linguagem SQL Server. O principal objetivo do script é criar as tabelas necessárias para o funcionamento do sistema, definindo suas colunas, tipos de dados e relacionamentos, garantindo uma estrutura eficiente e organizada para o armazenamento das informações.

Inicialmente, é criada a tabela Usuario contendo os campos id, email, senha, cargo, departamento, nível de acesso, nome e sobrenome. O campo id atua como chave primária e como ponte entre a tabela Usuario e as demais tabelas especializadas, além disso, duas triggers são definidas:

* trg\_InserirUsuario: insere automaticamente os dados do usuário na tabela correspondente ao seu tipo (Cliente, Técnico ou Administrador) após um INSERT na tabela Usuario.
* trg\_Usuario\_PadronizarCargo: padroniza automaticamente os textos dos campos de cargos, garantindo consistência na entrada de dados.

Em seguida, são criadas as tabelas Cliente, Tecnico e Administrador, cada uma contendo apenas o campo id\_user, que é chave primária da respectiva tabela e chave estrangeira referenciando o id da tabela Usuario. Essa separação por tipo de usuário permite, futuramente, o armazenamento de dados específicos e exclusivos para cada categoria de usuário.

Posteriormente, é criada a tabela Chamado, com os campos status, data\_abertura, data\_fechamento, data\_problema, prioridade, categoria, descricao, solucao\_ia, solucao\_tecnico, solucao\_final e tecnico\_responsavel, sendo este último uma chave estrangeira que referencia o campo id\_user da tabela Tecnico. A tabela Chamado conta com restrições (constraints) para garantir a integridade dos dados:

* CK\_Status\_Chamado: valida que apenas valores de status permitidos sejam inseridos.
* CK\_Prioridade\_Chamado: valida os valores de prioridade permitidos.

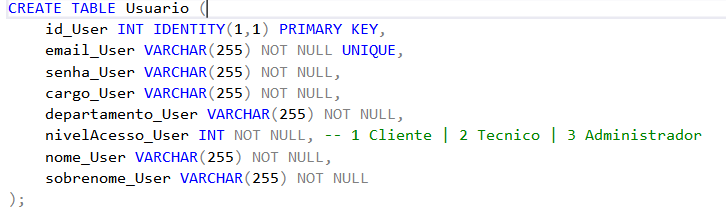
Além disso, foi implementada a trigger trg\_NormalizaStatusChamado, que padroniza automaticamente os textos inseridos nos campos status e prioridade. Por exemplo, se o usuário inserir "EM anDAMENTO", o sistema normaliza e armazena como "Em andamento. Por fim, é criada a tabela Relatorio, composta pelos campos id\_relatorio, id\_chamado, relatorio, data\_criacao e id\_usuario\_criador. Desses, dois são chaves estrangeiras:

* id\_chamado referencia o campo id\_chamado da tabela Chamado;
* id\_usuario\_criador referencia o campo id\_user da tabela Usuario.

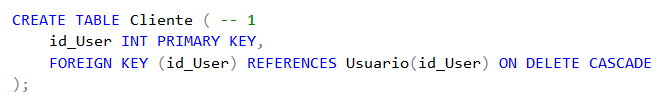
Para garantir que apenas usuários do tipo **Administrador** possam criar relatórios, foi definida a trigger trg\_ValidarRelatorioADM. Essa trigger verifica se o valor de id\_usuario\_criador corresponde a um usuário da tabela Usuario que esteja registrado como Administrador, bloqueando a inserção caso contrário. A seguir estarão imagens do código SQL, todas com autoria própria

**Códigos SQL para a criação das tabelas:**

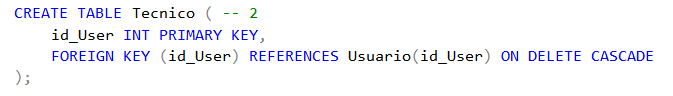
**Figura 18 – Tabela Usuario.**



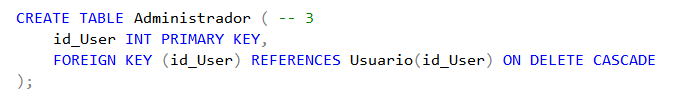
**Figura 19 – Tabela Cliente.**



**Figura 20 – Tabela Tecnico.**



**Figura 21 – Tabela Administrador.**



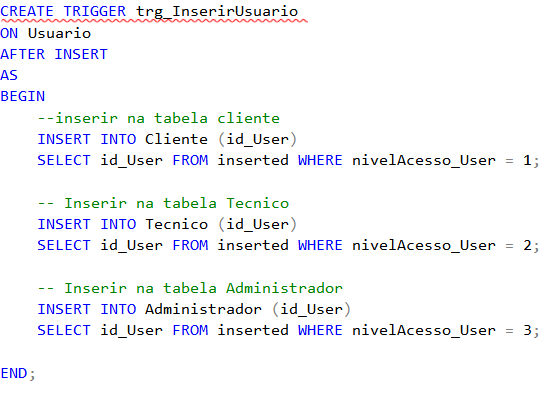
**Figura 22 – INSERT para povoar as tabelas.**



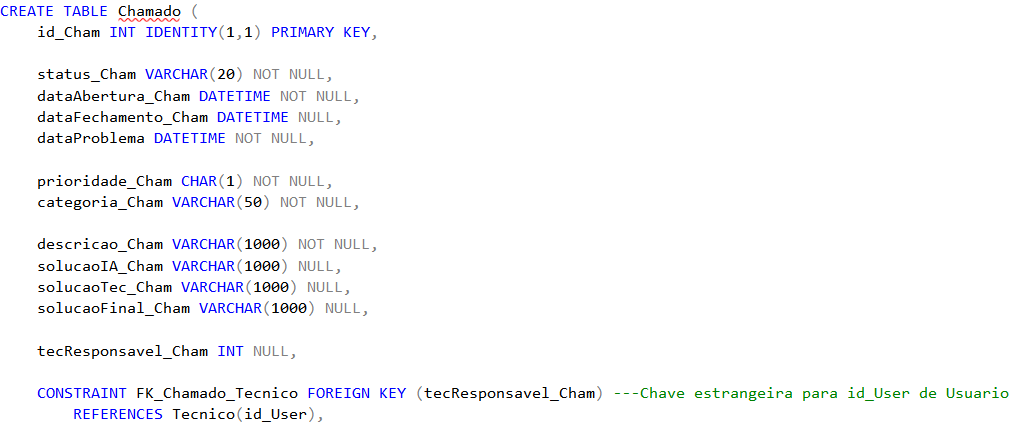
**Figura 23 – Resultado da seleção: SELECT \* FROM Usuario.**



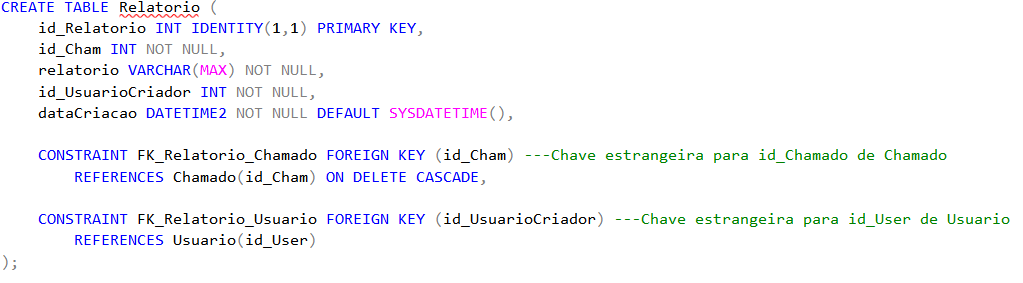
**Figura 24 – Tabela TRIGGER para a inserção automática**



**Figura 25 – Tabela Chamado**



**Figura 26 – Tabela Relatorio**

 No modelo conceitual visual, o atributo da tabela relatório era id\_User, porém foi criado outro nome (id\_UsuarioCriador) referenciando a PK id\_User, melhorando a manutenibilidade do script. A preocupação com a integridade dos dados e a separação por papéis de usuários reforça o compromisso com a escalabilidade e a clareza do sistema. Com essa estrutura sólida, o sistema está preparado para sustentar as funcionalidades previstas e permitir futuras evoluções com consistência e eficiência.

Avaliando o mercado de softwares de gerenciamento de chamados onde este projeto se situaria, é notável o aumento da procura por estes sistemas operacionais. Empresas como Zendesk, Jira e Movidesk possuem crescentes em seus faturamentos ano a ano, deixando claro o aumento da procura por empresas de helpdesk.   
 A viabilidade econômica deste projeto é definida por fatores que vão além do investimento financeiro de seu desenvolvimento, uma vez que não haverá custo na criação dos sistemas. Desta forma, as otimizações que a implantação da plataforma de gerenciamento de chamados de TI trará nos processos da organização podem ser avaliadas.

Com a organização dos chamados e da comunicação entre os funcionários para este fim será centralizada e padronizada dentro do HELPBOX, espera-se que os custos operacionais sejam reduzidos (devido à diminuição de tempo e de recursos dedicados neste processo) e que a produtividade aumente, pela digitalização e automatização das etapas desempenhadas pelos técnicos e pela melhora na resposta dos clientes, que terão suas solicitações atendidas de forma mais eficiente.

Com isto, também se projeta que o Retorno sobre Investimento (ROI) seja calculado em cima da diminuição de tempo ocioso que ocorria devido à erros nos equipamentos ou softwares de informática na empresa. A segurança aplicada no banco de dados utilizado também pode prevenir futuros problemas financeiros ou legais sobre a proteção de informações sensíveis.

Uma última possibilidade no impacto de viabilidade econômico do sistema, este pode ser escalado como produto competitivo no mercado e aplicado em outras empresas, com as devidas adaptações, caso a empresa atual decida por comercializá-lo.

Para o sucesso deste negócio também é essencial selecionar profissionais que possuam não só habilidades técnicas, mas também sintonia com a cultura organizacional e os valores da empresa. A utilização de um perfil profissional no qual a empresa espera encontrar é outro método de recrutamento no qual se alinha a qualidade e desempenho esperado de um profissional que queria ingressar na empresa.

*Perfil Básico - HelpBox:*

Profissional em início de carreira na área de Tecnologia da Informação, com base sólida em informática, lógica de programação e noções de banco de dados. Motivado pelo aprendizado contínuo, busca oportunidades para colocar em prática seus conhecimentos e evoluir tecnicamente em ambientes colaborativos e dinâmicos.

* Formação
  + Cursando ou com ensino técnico/superior em Informática, Sistemas de Informação, Ciência da Computação, Engenharia de Software ou áreas afins.
* Conhecimento Técnico
  + Noções básicas de lógica de programação;
  + Conhecimento introdutório em linguagens como Python, C#, HTML/CSS ou JavaScript;
  + Familiaridade com bancos de dados e comandos básicos de SQL;
  + Conhecimento em pacote Office e sistemas operacionais (Windows e/ou Linux);
  + Conhecimento de ferramentas como GitHub e VS Code; e
  + Desejável - noções de redes, hardware ou suporte técnico.
* Competências Comportamentais
  + Responsabilidade e organização;
  + Boa comunicação e facilidade para trabalhar em equipe; e
  + Proatividade e curiosidade tecnológica.
* Objetivo Profissional
  + Iniciar a carreira na área de TI, contribuindo com tarefas de suporte, desenvolvimento ou manutenção de sistemas, ao mesmo tempo em que desenvolve habilidades práticas e amplia seus conhecimentos técnicos.

E um foco maior em treinamento e desenvolvimento, disponibilização do infográfico de como fazer um chamado e o manual de usuário são as principais ferramentas voltadas ao treinamento da equipe e dos usuários.

# 11. CONSIDERAÇÕES FINAIS

Este projeto surgiu como uma resposta para a otimização de sistemas de gerenciamento de chamados e suporte técnico com base em Inteligência Artificial. Durante a construção deste, ficou evidente a necessidade e utilidade de todos os conteúdos abordados e conhecimentos adquiridos no decorrer do semestre para que a análise e modelagem do sistema fosse concisa e completa para um desenvolvimento eficiente da codificação do software no futuro.

O levantamento de requisitos, a modelagem UML (caso de uso, classe, sequência e implantação), a construção do modelo conceitual, físico e lógico do banco de dados, em conjunto com o dicionário de dados fez com que fosse possível uma modelagem consistente do sistema, demonstrando habilidade de compreensão perante as necessidades e expectativas do cliente, que auxiliarão na rastreabilidade e manutenibilidade futura. As planilhas de testes de software e os casos de uso descritivos facilitaram a busca pela qualidade do software.

O protótipo gerou a possibilidade de validar as ideias, identificar possíveis problemas de design e de funcionalidade, além da eficiência das mesmas, prezando pela usabilidade. Ademais, o estudo de viabilidade econômica e a delimitação do modelo de contratação gerou uma perspectiva de espaço de crescimento de mercado e construção de uma equipe produtiva em um cenário competitivo.

Posteriormente, sugere-se que algumas alterações e adições sejam implementadas, desenvolvendo features de notificações para atualizações sobre os chamados que o usuário abriu ou para avisar um técnico quando ele for atribuído a um novo chamado, adicionar status do usuário (ativo ou inativo) e a data em que o usuário foi criado, além de analisar a viabilidade de incluir um chat entre clientes e técnicos para centralizar totalmente a comunicação.

É pertinente buscar formas de incluir acessibilidade (aumento de fonte, mudança de contraste, integração com a API vLibras), bem como implementar medidas de segurança para garantir robustez no cumprimento da Lei Geral de Proteção de Dados Pessoais (LGPD) de forma robusta, incluindo também uma nova confirmação no sistema do Administrador para a inclusão de senha em ações críticas, visando segurança .

Futuramente este software também poderá ser aplicado em outras empresas além da Esfera Contabilidade descrita nesse projeto, e adaptações para comercialização serão necessárias.

# REFERÊNCIAS BIBLIOGRÁFICAS

ESL SISTEMAS. I**nvestimento em tecnologia faz empresas crescerem 60% a mais no mercado**, 2024. Disponível em: https://eslsistemas.com.br/investimento-em-tecnologia-faz-empresas-crescerem-60-a-mais-no-mercado/ . Acesso em: 27 de março de 2025.

TEIXEIRA, Rafael Fialho. **Sistema de chamados: o que é, como funciona e como implementar**. Desk Manager, 25 de outubro de 2024. Disponível em: https://deskmanager.com.br/blog/sistema-de-chamados/. Acesso em: 27 de março de 2025.

STRIKER, Cole; KAVLAKOGLU, Eda. **O que é inteligência artificial?**. IBM, 09 de agosto de 2024. Disponível em: https://www.ibm.com/br-pt/think/topics/artificial-intelligence . Acesso em: 28 de março de 2025.

GAVA, Marcela. **Pesquisa sobre atendimento ao cliente mostra SAC melhor agora do que nos últimos dois anos**. Capterra, 28 de junho de 2022. Disponível em: https://www.capterra.com.br/blog/2763/pesquisa-atendimento-cliente . Acesso em: 28 de março de 2025.

SEBRAE. **Atendimento ao cliente: o que é e como fidelizar o consumidor**, 2024. Disponível em: https://sebrae.com.br/sites/PortalSebrae/artigos/atendimento-ao-cliente-o-que-e-como-fazer-para-fidelizar-o-consumidor%2Ce4b0af9d324bb810VgnVCM1000001b00320aRCRD . Acesso em: 28 de março de 2025.

SILVA, Douglas da. **Dados sobre o atendimento ao cliente**. ZENDESK, 06 de março de 2025. Disponível em: https://www.zendesk.com.br/blog/dados-estatisticas-atendimento-cliente/ . Acesso em: 28 de março de 2025.

Sommerville, I. **Engenharia de Software.** 9 ed. São Paulo: Pearson Prentice Hall, 2011.

WAZLAWICK, Raul S. **Análise e Design Orientados a Objetos para Sistemas de Informação: Modelagem com UML, OCL e IFML**. 3. ed. Rio de Janeiro: GEN LTC, 2014. Disponível em: <https://app.minhabiblioteca.com.br/reader/books/9788595153653/>. Acesso em: 21 mai. 2025.

BOOCH, G., JACOBSON, I., RUMBAUGH, J. (2005). **UML: guia do usuário.** 2. ed. Rio de Janeiro: Campus.

SOMMERVILLE, Ian. Engenharia de Software. 10ª ed. São Paulo: Pearson Universidades, 2019.

COOPERSYSTEM, **Análise de requisitos: o que é e para que serve?**, 2023 Disponível em: https://www.coopersystem.com.br/analise-de-requisitos-o-que-e-e-para-que-serve/. Acesso em: 07 de maio de 2025.

DEVMEDIA, **Qualidade de Software - Engenharia de Software**, Disponível em: https://www.devmedia.com.br/amp/qualidade-de-software-engenharia-de-software-29/18209. Acesso em: 08 de maio de 2025.

SOMMERVILLE, Ian. **Engenharia de Software**. 10ª ed. São Paulo: Pearson Universidades, 2019.

AMAZON, **O que é o Scrum?**, Disponível em: https://aws.amazon.com/pt/what-is/scrum/. Acesso em: 08 de maio de 2025.

ORACLE. **What is a database?** Disponível em: https://www.oracle.com/br/database/what-is-database. Acesso em: 16 abr. 2025.

PUIG, Luis Ezequiel. N**ormalização em Banco de Dados - Estrutura**. Alura, 7 mar. 2023. Disponível em: https://www.alura.com.br/artigos/normalizacao-banco-de-dados-estrutura. Acesso em: 4 maio 2025.

SILVA, José Carlos da; ALMEIDA, Maria de Fátima Costa; SOUZA, Ana Paula de Lima. **Ratio-Entity Diagram: uma ferramenta para modelagem conceitual de dados em Engenharia de Software**. Research, Society and Development, [S. l.], v. 10, n. 7, p. e44210717776, 2021. DOI: 10.33448/rsd-v10i7.17776. Disponível em: https://rsdjournal.org/index.php/rsd/article/view/17776/15626. Acesso em: 16 abr. 2025.

BARBOSA, Eduardo Pinto; ALVARENGA, Geoflávia Guilarducci de. **Uma proposta para ensino de dicionário de dados em projetos de bancos de dados.** Anais do Workshop sobre Educação em Computação (WEI), 20., 2012, Curitiba/PR. Porto Alegre: Sociedade Brasileira de Computação, 2012. Disponível em: https://sol.sbc.org.br/index.php/wei/article/view/29098/28903. Acesso em: 10 maio 2025.

BORSOI, Beatriz T.; COLAZZOS, Kathya S. L.; ASCARI, Rúbia E. O. S.; TOSCAN, Luiz F.; BOSSOLA, Luiz H.; BOLO, Matheus M.; ARSEGO, Matheus M. **Redes neurais aplicadas na estimativa de prazo de projeto de software.** IV EPAC – Encontro Paranaense de Computação. Pato Branco: Universidade Tecnológica Federal do Paraná, 2015. Disponível em: https://cliqueapostilas.com/Content/apostilas/b1e03f65cc0ee3bbb3bcd0d8bd85b514.pdf. Acesso em: 13 abr. 2025.

PEREIRA, Paulo; TORREÃO, Paula; MARÇAL, Ana Sofia. **Entendendo Scrum para Gerenciar Projetos de Forma Ágil.** 2007. Disponível em: https://faculdadeprojecao.nucleoead.net/pos/pluginfile.php/2123/mod\_resource/content/36/Entendendo\_Scrum\_para\_Gerenciar\_Projetos.pdf. Acesso em: 10 maio 2025.

SEBRAE. **ROI calcula o retorno de seus investimentos.** Sebrae, 6 maio 2023. Disponível em: https://sebrae.com.br/sites/PortalSebrae/artigos/roi-calcula-o-retorno-de-seus-investimentos,17870ffe48437810VgnVCM1000001b00320aRCRD. Acesso em: 10 maio 2025.

Curso de Automação Industrial [Online]. Universidade Federal do Rio Grande do Norte. Disponível em: <https://materialpublic.imd.ufrn.br/curso/disciplina/1/8/1/3>. Acesso em: 22 maio 2025.

BONATINI, B. S. **Quais os benefícios da Programação Orientada a Objetos no desenvolvimento de softwares**, 2023. Disponível em: <https://medium.com/@brunobonatini/quais-os-benef%C3%ADcios-da-programa%C3%A7%C3%A3o-orientada-a-objetos-no-desenvolvimento-de-softwares-c836f7ce275>. Acesso em: 03 maio. 2025.

NORMANDO, Célio. **Orientação a objetos**, 2024. Disponível em: <https://medium.com/@celionormando/orienta%C3%A7%C3%A3o-a-objetos-1cc5cd11abfa> Acesso em: 10 de maio de 2025.

SILVA, Cleuton. **Os 4 Pilares da Programação Orientada a Objetos**, 2023. Disponível em: <https://www.dio.me/articles/os-4-pilares-da-programacao-orientada-a-objetos-SSU4Q9> Acesso em: 02 de maio de 2025.

TWYGO, **RH Estratégico, o que é, qual a importância e como implantar**, 2022. Disponível em: <https://twygo.com/blog/rh-estrategico/>. Acesso em: 04 de maio de 2025.

SERASA EXPERIAN, **RH: Qual a função desse departamento?**, 2023. Disponivel em: < https://www.serasaexperian.com.br/carreiras/blog-carreiras/rh-departamentofuncionarios-e-importancia-para-a-empresa/>. Acesso em: 21 de abril de 2025.

LECOM, **Gestão estratégica de RH: o que é, importância e ferramentas**, 2023. Disponível em: < https://www.lecom.com.br/blog/gestao-estrategica-rh/>. Acesso em: 04 de maio de 2025.

CERQUEIRA, Alex. Qualidade em Usabilidade, 2024. Disponível em: <https://medium.com/editoraux/qualidade-em-usabilidade-48006cbb3a33> Acesso em: 10 de abril de 2025.

TEIXEIRA, Fabricio. **Análise heurística: o que é, como fazer e os benefícios para o projeto**, 2016. Disponível em: <https://brasil.uxdesign.cc/an%C3%A1lise-heur%C3%ADstica-o-que-%C3%A9-como-fazer-e-os-benef%C3%ADcios-para-o-projeto-161f3d94436b> Acesso em: 12 de abril de 2025.

WOEBCKEN, Cayo. **Entenda o que é teste de usabilidade, para que serve e como é feito**, 2021. Disponível em: <https://rockcontent.com/br/blog/teste-de-usabilidade/> Acesso em: 10 de abril de 2025.

CONDE, Natacha. **Protótipo x MVP: vamos entender a diferença?**, 2022 Disponível em: <https://troposlab.com/prototipo-mvp-vamos-entender-a-diferenca/>. Acesso em: 12 de abril de 2025.

# ANEXO 1

**Atividade de Extensão Universitária - ELICITAÇÃO**

**Área Temática e Projetos**

Tecnologia e Produção Desenvolvimento de Aplicativos/Software/Web Site para a comunidade etc.

**Ação**

Capacitação ou treinamento.

**Descrição Geral da Atividade**

Como o Projeto Integrado Multidisciplinar III solicitou que o grupo realizasse as análises e desenvolvimentos de um sistema que gerenciará chamados, utilizando a IA, foi decidido que, para se ter uma base de o que um programa desse calibre precisa possuir, seria realizado entrevistas com funcionários já formados na área da tecnologia que trabalham com resolução de chamados.

Foram selecionadas cinco pessoas que atuam nessa área em diferentes empresas, buscando levantar requisitos para o projeto gerando a base do sistema.

Perguntas iguais foram fornecidas para os funcionários, que dispuseram do seu tempo hábil de forma gratuita para auxiliar no desenvolvimento do projeto. Foi então criado um formulário pelo grupo e fornecido o link para cada um, onde responderam de forma totalmente livre sobre as suas tarefas diárias em suas respectivas empresas.

**Descrição da participação de cada aluno**

Guilherme e Camilly

A dupla foi responsável pela busca das pessoas a serem entrevistadas. Foram selecionadas pessoas que possuem experiência na área de suporte a chamados, tendo como local de referência para a seleção o site *LinkedIn* e comunidades de tecnologia em geral. As pessoas entrevistadas foram selecionadas de empresas diferentes para verificar se haveria alguma diferença muito grande nas tratativas dos chamados ou se também existiria alguma diferença significativa.

João Lucas e Micaías

Ambos ficaram responsáveis inicialmente pela elaboração das perguntas à serem realizadas. Depois de definidas, foi estabelecido pelo grupo que seria interessante receber as respostas através de um *Forms* da empresa *Google*, que então foi criado e elaborado pela respectiva dupla.

Vanessa e Agatha

Ficaram responsáveis pela análise das respostas das pessoas entrevistadas. Foi o último processo, onde foram estudados os pensamentos de cada entrevistado e colocado na ponta do lápis para que, posteriormente, fosse iniciado o projeto, já que foi através dele que foram formalizados os requisitos do projeto do PIM.

**Conclusão**

Com base nas contribuições dos entrevistados, foram desenvolvidos os requisitos para o sistema à ser gerado no futuro, analisando a resposta de cada um e buscando uma generalização do que é relativamente igual entre os processos diários de cada.

Um fato interessante é que os chamados nem sempre são feitos por um sistema, sendo feitos, em mais de um caso dos entrevistados, através de e-mail e WhatsApp, sendo uma forma um pouco mais informal: ela agiliza o processo de abertura, porém os dados do chamado são mais vulneráveis.

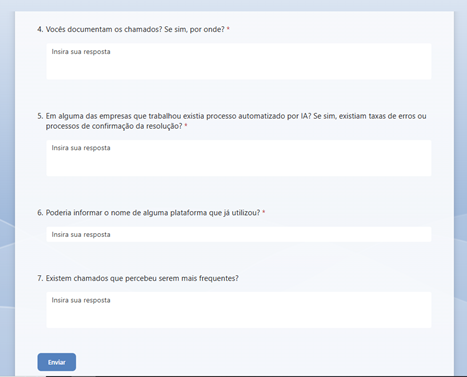
Foi notável que muitos deles não utilizam uma Inteligência Artificial para uma triagem ou resoluções de problemas simples, sendo destacado apenas por uma das pessoas que já utilizou da ferramenta para aumentar o tempo hábil dos técnicos, já que a mesma fornecia soluções para problemas simples e era treinada por uma equipe da própria empresa.

Além disso, fica claro que a maioria dos chamados recebidos são devidos à problemas no software, devido ao mal desempenho e acesso, fato este que já era esperado pelo grupo.

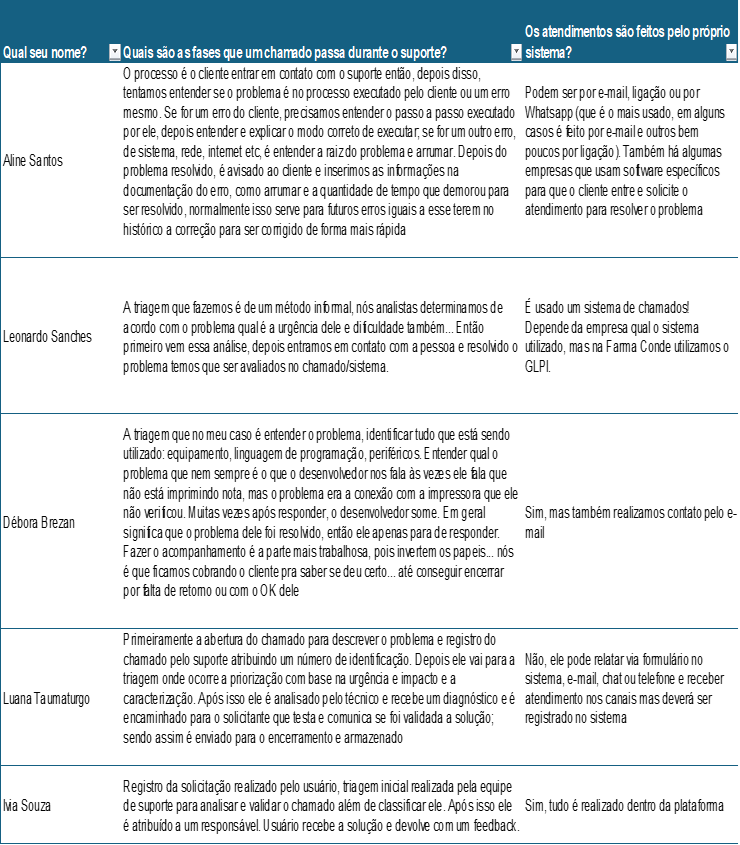
**Comprovação**

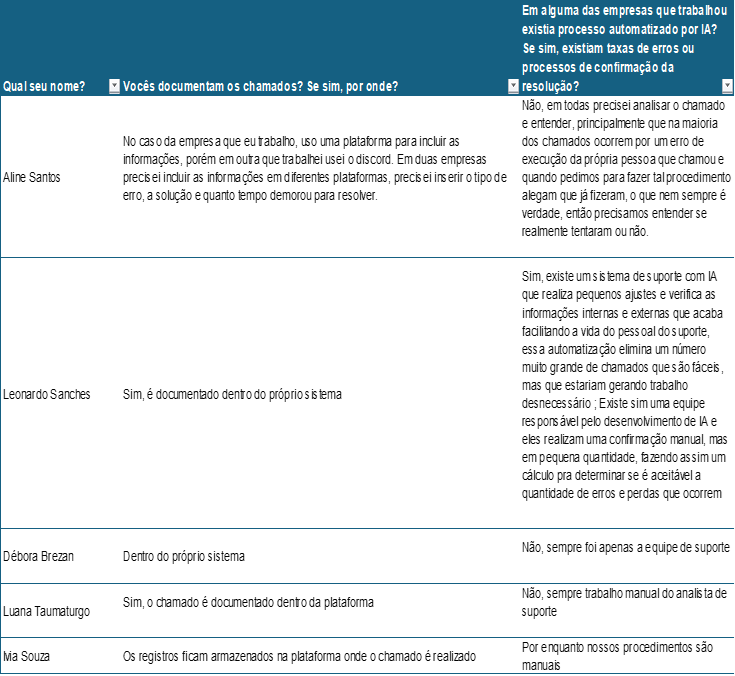
* Prints do formulário do Google

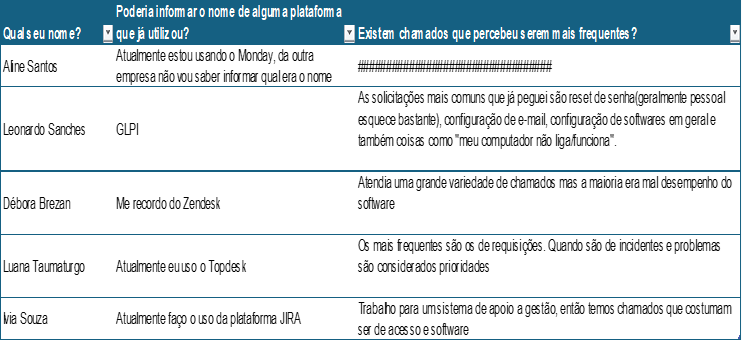




* Respostas do formulário







* Link do formulário: <https://forms.office.com/pages/responsepage.aspx?id=jOaT0T_lEEambVb_MA_semAS-QdqwKZLgZ0EuiIqpOVUQlFUVkhPT0sxR0JYOE1OTTlYR0oxTkhMWi4u&route=shorturl>

Registros

|  |  |
| --- | --- |
| Data do encontro | Observações |
| 16/03/2025 | Leitura e análise do tema proposto pelo PIM |
| 18/03/2025 | Entrevistas realizadas com técnicos de suporte (elicitação) |
| 22/03/2025 | Levantamento dos requisitos funcionais e não funcionais do código |
| 24/03/2025 | Criação do texto de introdução do projeto |
| 26/03/2025 | Desenvolvimento do Diagrama de Caso de Uso |
| 29/03/2025 | Criação das Regras de Negócio e Caracterização da Empresa |
| 30/03/2025 | Desenvolvimento da logo da empresa |
| 05/04/2025 | Criação do texto para a disciplina de Economia e Mercado |
| 08/04/2025 | Início da modelagem do banco de dados e o texto para a disciplina |
| 09/04/2025 | Início da construção do diagrama de classe |
| 12/04/2025 | Criação do texto para a disciplina de Análise de Sistemas Orientada a Objetos |
| 19/04/2025 | Criação do texto de Engenharia de Software e Projeto de Interface com o Usuário |
| 20/04/2025 | Início da criação do protótipo no Figma |
| 21/04/2025 | Criação do texto para a disciplina de Gestão de RH |
| 23/04/2025 | Início da construção do diagrama de sequência |
| 30/04/2025 | Início da construção do diagrama de implantação |
| 04/05/2025 | Verificação das matérias com os professores responsáveis |
| 06/05/2025 | Finalização dos ajustes e criação do arquivo final |
| 22/05/2025 | Finalização do protótipo, revisão de ortografia e referências bibliográficas |
| 24/05/2025 | Envio do projeto |
|  |  |