

SISTEMA WEB PARA GERENCIAMENTO DE CHAMADOS TÉCNICOS EM PEQUENAS EMPRESAS

Nelson Tadeu Vaz de Oliveira¹

Julio Cesar Carou Felix de Lima²

Olinda Nogueira Paes Rizzo³

Resumo

Introdução

A transformação digital é uma realidade em empresas de todos os portes, contudo, Micro e Pequenas Empresas (MPEs) frequentemente enfrentam obstáculos para implementar e estruturar processos de atendimento técnico informatizados. É comum que essas empresas utilizem ferramentas genéricas e desconexas, como planilhas eletrônicas, e-mails e mensageiros instantâneos, para gerenciar solicitações de suporte. Essa abordagem fragmentada gera desafios operacionais significativos, incluindo a perda de informações, a dificuldade de priorização de demandas e a baixa visibilidade do status dos chamados para os clientes, impactando negativamente a satisfação e a eficiência (Sommerville, 2011).

Este trabalho busca investigar o seguinte problema de pesquisa: Como a informatização do fluxo de chamados técnicos impacta a comunicação, a visibilidade e a eficiência operacional no contexto específico das Micro e Pequenas Empresas (MPEs)? Parte-se da hipótese de que a implementação de um sistema web centralizado, desenvolvido sob medida para as necessidades e restrições das MPEs, pode reduzir o tempo médio de resolução de chamados, aumentar a transparência para o cliente e melhorar a organização interna da equipe de suporte.

O desenvolvimento do sistema seguiu uma abordagem incremental, com cycles curtos de implementação e validação, permitindo ajustes contínuos baseados no feedback dos usuários. A solução foi construída utilizando tec-

nologias web modernas e padrões de arquitetura consolidados, visando atender às necessidades específicas das MPEs.

A Engenharia de Software oferece princípios e práticas para abordar tais problemas, abrangendo desde a especificação de requisitos até a manutenção do software, com foco na qualidade do produto e do processo (PRESSMAN, 2011). Neste contexto, a justificativa deste trabalho reside no potencial de uma solução sob medida para reduzir o esforço operacional, fortalecer a gestão do conhecimento tácito da equipe e melhorar a experiência do cliente, preenchendo uma lacuna entre soluções de mercado complexas e caras e a realidade operacional dessas empresas.

Este artigo está organizado da seguinte forma: além desta introdução, a seção 2 detalha os objetivos gerais e específicos; a seção 3 descreve a metodologia aplicada, incluindo a base teórica que fundamenta as escolhas de desenvolvimento; a seção 4 expõe o processo de desenvolvimento com detalhes sobre arquitetura e implementação; a seção 5 apresenta e discute os resultados obtidos através de testes sistemáticos; e, finalmente, a seção 6 traz as conclusões e propostas de trabalhos futuros, destacando as contribuições do estudo para o contexto das MPEs.

Objetivo

Objetivo Geral

Desenvolver um sistema web para gerenciamento de chamados técnicos em pequenas empresas e avaliar se os objetivos de eficiên-

¹Graduando em Engenharia de Software da Universidade Santo Amaro, SP. E-mail: nvaz264@gmail.com.

²Professor Mestre, Universidade Santo Amaro, SP. E-mail: jclima@prof.unisa.br.

³Professora Mestra, Universidade Santo Amaro, SP. E-mail: orizzo@prof.unisa.br.

cia, usabilidade e desempenho são atendidos.

Objetivos Específicos

- Elicitar e documentar requisitos funcionais e não funcionais do sistema.
- Modelar a solução por meio de diagramas de casos de uso e de classes.
- Implementar um protótipo funcional do sistema.
- Conduzir testes funcionais, de usabilidade e de desempenho no protótipo.
- Avaliar os resultados obtidos e propor melhorias e trabalhos futuros.

Metodologia

A revisão bibliográfica sistemática foi conduzida entre janeiro e março de 2025, nas bases IEEE Xplore, ACM Digital Library, Scopus e Google Scholar, utilizando combinações dos descritores: ('ticket system' OR 'help desk') AND ('SME' OR 'small business') AND ('usability' OR 'user experience') AND ('web development'). Foram selecionados 18 artigos publicados entre 2018 e 2023, priorizando estudos aplicados ao contexto de pequenas empresas, mediante critérios de inclusão como: abordagem de desenvolvimento web, avaliação de usabilidade e aplicabilidade em MPEs. A pesquisa caracteriza-se como aplicada, com abordagem mista (qualitativa e quantitativa) e estratégia de estudo de caso único. O estudo foi conduzido em parceria com uma pequena empresa do setor de serviços de TI, localizada em São Paulo, com aproximadamente 15 colaboradores. A escolha por um estudo de caso único permitiu uma investigação profunda e contextualizada do problema. A base teórica para este estudo foi construída a partir de uma revisão bibliográfica inicial, que orientou as escolhas metodológicas e técnicas. Foram consultados autores clássicos da Engenharia de Software (como Pressman e Sommerville) para fundamentar o processo de desenvolvimento, normas internacionais como a ISO 25010 para embasar os requisitos de qualidade, e literatura específica sobre usabilidade

(Nielsen, 1993; Krug, 2014) e métodos ágeis (Beck et al., 2001) para guiar a avaliação e a gestão do projeto.

O processo de elicitação de requisitos foi conduzido através de duas técnicas principais: entrevistas semiestruturadas e observação direta do processo atual. Foram realizadas cinco entrevistas com os stakeholders principais: um gestor, dois técnicos de suporte e dois clientes-chave da empresa parceira. As entrevistas foram gravadas (com consentimento prévio) e transcritas para análise. O roteiro continha 10 questões, abordando desde problemas operacionais até expectativas sobre funcionalidades e usabilidade. O roteiro de entrevista abordou pontos como as principais dificuldades operacionais no processo manual, funcionalidades essenciais desejadas em um sistema de chamados e expectativas em relação a usabilidade e tempo de resposta.

Cada entrevista teve duração média de 45 minutos e foi documentada em forma de relatórios e mapas mentais para posterior análise. Os requisitos coletados foram documentados e priorizados utilizando a técnica MoSCoW (Must have, Should have, Could have, Won't have), em uma sessão de workshop com a equipe de gestão da empresa parceira. Essa priorização foi crucial para definir o escopo mínimo viável (MVP) para o protótipo, garantindo que as funcionalidades mais críticas para a operação fossem implementadas primeiro.

Para a coleta de dados durante a fase de avaliação, foram elaborados instrumentos específicos para cada tipo de teste. Para os testes de usabilidade, utilizou-se um protocolo composto pelo questionário SUS acrescido de três questões abertas sobre aspectos específicos da interface (clareza visual, facilidade de navegação e utilidade das funcionalidades). Os testes de desempenho foram conduzidos mediante script automatizado no Apache JMeter, simulando 10, 20 e 50 usuários concorrentes executando as operações principais do sistema (abertura de chamado, listagem e atualização de status) durante um período de 15 minutos para cada cenário.

Os dados quantitativos foram analisados através de estatística descritiva (média, desvio padrão) para os tempos de resposta e pontuação

SUS. Os dados qualitativos das entrevistas iniciais e dos testes de usabilidade foram categorizados tematicamente seguindo a análise de conteúdo (Bardin, 2011), identificando padrões recorrentes como 'facilidade de uso', 'transparência do processo', 'agilidade na comunicação' e 'organização do fluxo de trabalho'.

O método de desenvolvimento adotado foi incremental, com iterativas curtas e validações frequentes, permitindo ajustes contínuos baseados em feedback. O estudo foi conduzido em quatro etapas principais: (1) Levantamento de Requisitos: realizado por meio de entrevistas semiestruturadas e observação do processo atual da empresa parceira, para compreender a fundo a problemática; (2) Modelagem: utilizou-se a linguagem UML para criar diagramas de casos de uso e de classes, definindo a funcionalidade e a estrutura estática do sistema; (3) Implementação: desenvolvimento de um protótipo funcional seguindo a arquitetura em camadas (Model-View-Controller (MVC)); (4) Testes e Avaliação: foram realizados testes funcionais para verificar a atendimento aos requisitos, testes de usabilidade aplicando o questionário System Usability Scale (SUS) com uma amostra de 5 usuários potenciais, e testes de desempenho para medir tempos de resposta sob carga básica.

Ferramentas Utilizadas

Para o apoio ao desenvolvimento e gestão do projeto, foram utilizadas as seguintes ferramentas: Astah UML para modelagem, Trello para gestão ágil do backlog, XAMPP como ambiente de desenvolvimento integrado, e Apache JMeter para a realização dos testes de desempenho.

Justificativa das Escolhas Técnicas

A escolha pelo padrão arquitetural Model-View-Controller (MVC) justifica-se pela separação de concerns (preocupações), que facilita a manutenção e escalabilidade do sistema. O framework Laravel foi selecionado por sua ro-

bustez, curva de aprendizado favorável e ampla comunidade, fatores alinhados ao contexto das MPEs. Por fim, o Bootstrap garantiu a construção de uma interface responsiva e acessível com agilidade, atendendo ao requisito não funcional de usabilidade (RNF02).

Aspectos Éticos

Este trabalho foi conduzido seguindo os preceitos éticos da pesquisa acadêmica. Para garantir o anonimato e a privacidade dos envolvidos, a empresa parceira é referida de forma genérica ao longo do texto. Todos os participantes envolvidos nas etapas de elicitação de requisitos e testes de usabilidade concordaram com a participação por meio de um Termo de Consentimento Livre e Esclarecido (TCLE), que lhes garantia o direito de desistir da participação a qualquer momento, conforme diretrizes da Universidade Santo Amaro. Os dados coletados foram utilizados exclusivamente para os fins desta pesquisa.

Resultados e Discussão

Os resultados dos testes demonstraram a aderência do protótipo aos objetivos propostos. Nos testes funcionais, 28 dos 30 casos de teste criados foram aprovados, atingindo um índice de sucesso de 93%. Os dois casos reprovados relacionavam-se a validações secundárias de formulário, que foram ajustadas posteriormente. A avaliação de usabilidade resultou em uma pontuação média de 81 na escala SUS, superando a meta estabelecida de 75 pontos. O feedback qualitativo dos usuários destacou positivamente a clareza na visualização do status dos chamados e a utilidade do histórico completo de interações, que proporciona transparência ao processo. Os testes de desempenho, realizados sob carga simulada de usuários, apresentaram um tempo médio de resposta de 1,7 segundo para o carregamento das páginas principais, ficando abaixo do limite de 2 segundos definido como requisito não funcional (RNF03).

Para uma visão consolidada do desempenho

do sistema, Para uma visão consolidada do desempenho do sistema frente aos objetivos traçados, os principais resultados das avaliações realizadas são sumarizados na Tabela 1. A Taxa de Sucesso de Testes Funcionais mede a porcentagem de casos de teste que passaram sem erros, validando se as funcionalidades essenciais do sistema foram implementadas corretamente. A Pontuação Média SUS (System Usability Scale) reflete a percepção dos usuários sobre a facilidade de uso, onde

uma pontuação acima de 80 é considerada 'excelente'. Já o Tempo Médio de Resposta avalia a eficiência do sistema, garantindo que as interações principais ocorram dentro de um limite de 2 segundos, conforme recomendado para manter a fluidez da experiência do usuário. Os dados apresentados indicam que o sistema não apenas atendeu, mas superou as expectativas iniciais em todos os critérios mensurados.

Tabela 1 - Resultados da avaliação do protótipo do sistema

Métrica de Avaliação	Meta Estabelecida	Resultado Obtido
Taxa de sucesso de testes funcionais	90%	93%
Pontuação média SUS	75	81
Tempo médio de resposta (s)	< 2	1,7

Fonte: Autor (2025).

Além dos dados quantitativos, coletou-se feedback qualitativo através de questões abertas aplicadas juntamente com o questionário SUS. Os usuários destacaram como pontos positivos: 'interface intuitiva e de fácil aprendizado' (mencionado por 4 dos 5 usuários), 'histórico completo das interações em linha única do tempo' (3 usuários) e 'agilidade no carregamento das páginas' (3 usuários). Como sugestões de melhoria, foram apontados: 'incluir upload de múltiplos arquivos em um único chamado' (2 usuários), 'notificações por WhatsApp integradas' (2 usuários) e 'aplicativo mobile para acompanhamento offline' (1 usuário). Estes insights foram catalogados no backlog do produto para consideração em versões futuras.

Análise Detalhada dos Testes e Métricas Complementares

Para além das métricas gerais apresentadas na Tabela 1, uma análise mais granular dos resultados obtidos fornece insights valiosos sobre o desempenho do sistema em cenários específicos. Os testes funcionais, que alcan-

çaram 93% de sucesso, foram distribuídos por diferentes módulos do sistema, conforme detalhado na Tabela 2. Esta distribuição permite identificar quais áreas do sistema apresentaram maior maturidade e quais requereram ajustes pós-testes.

Complementarmente, buscou-se uma avaliação qualitativa mais estruturada e ampla, para além dos testes quantitativos de usabilidade e desempenho. Para isso, adotou-se como referência o modelo de qualidade de produto de software definido pela norma internacional ISO/IEC 25010, que permite uma análise sistemática de características críticas para a aceitação e longevidade de um sistema. Para representar de forma clara e visual o nível de atendimento a cada um desses critérios de qualidade, utilizou-se uma escala simplificada com três círculos, onde ●●● = Plenamente Atendido, ●● = Parcialmente Atendido e ● = Não Atendido. A Tabela 3 consolida essa avaliação, proporcionando uma visão holística da qualidade do software desenvolvido.

Tabela 2 - Taxa de sucesso por módulo funciona

Módulo do sistema	Casos de Teste Executados	Casos Aprovados	Taxa de Sucesso
Autenticação e Segurança	5	5	100%
Gestão de Chamados	12	11	92%
Comunicação (Notificações)	6	5	83%
Relatórios e Dashbord	4	4	100%
Administração do Sistema	3	3	100%

Fonte: Autor (2025).

Os dois casos reprovados concentraram-se no módulo de Comunicação, especificamente na validação de formatos de e-mail e na robustez do sistema de filas em cenários de concorrência extrema. Estes problemas foram identificados como de média criticidade e prontamente resolvidos através da implementação de validações adicionais no front-end e da otimização de queries no back-end.

No que tange aos testes de usabilidade, a pontuação SUS de 81 pontos foi decomposta para análise individual de cada uma das dez questões que compõem a escala. As questões relacionadas à facilidade de uso do sistema (questões 1, 3, 5, 7 e 9) obtiveram pontuação média agregada de 4,2 em uma escala de 1 a 5, enquanto as questões relacionadas à complexidade percebida e consistência (questões 2, 4, 6, 8 e 10) obtiveram pontuação média de 4,1. Esta distribuição homogênea indica que os usuários não identificaram pontos de dificuldade exacerbada ou inconsistências gritantes na interface, corroborando o feedback qualitativo que elogiou a intuitividade e clareza do fluxo de trabalho.

Os testes de desempenho expandiram-se além do tempo de resposta médio, avaliando também o consumo de recursos do servidor sob carga. Em um servidor com configuração modesta (2 vCPUs, 4GB RAM), o sistema manteve consumo de CPU abaixo de 60% e utilização de memória estável em torno de 512MB durante os picos de teste com 50 usu-

ários concorrentes. Estes resultados são particularmente relevantes para o contexto das MPEs, que tipicamente operam com infraestrutura de TI limitada, demonstrando que a solução desenvolvida é tecnicamente viável e economicamente sustentável para este porte de empresa.

A avaliação de desempenho incluiu ainda testes de resistência (endurance tests), onde o sistema foi submetido a uma carga constante de 20 usuários concorrentes por um período prolongado de 60 minutos. Neste cenário, o tempo de resposta manteve-se estável em 1.7s ($\pm 0.2s$), não sendo observados vazamentos de memória ou degradação de performance, o que atesta a robustez da arquitetura implementada e a eficiência das escolhas tecnológicas, notadamente o uso do framework Laravel com Eloquent ORM para operações de banco de dados e do Redis para gerenciamento de filas de e-mail.

Complementarmente, buscou-se uma avaliação qualitativa mais estruturada e ampla, para além dos testes quantitativos de usabilidade e desempenho. Para isso, adotou-se como referência o modelo de qualidade de produto de software definido pela norma internacional ISO/IEC 25010, que permite uma análise sistemática de características críticas para a aceitação e longevidade de um sistema. Os atributos de Functional Suitability (Adequação Funcional), Performance Efficiency (Eficiência de Performance) e Usability (Usabilidade) fo-

ram plenamente atendidos, conforme demonstrado pelos testes realizados. Os atributos de Compatibility (Compatibilidade) e Maintainability (Manutenibilidade) foram validados através da adoção de padrões abertos (HTTP, REST, SMTP) e da arquitetura em camadas MVC, que facilita a evolução e manutenção do código. O atributo de Security (Segurança) foi parcialmente validado através dos testes de autenticação e controle de acesso, sendo recomendado para trabalhos futuros testes de penetração mais aprofundados. A Tabela 3 consolida essa avaliação, proporcionando uma visão holística da qualidade do software desenvolvido. Para indicar o nível de atendimento de cada sub-atributo, utiliza-se uma escala de três pontos representada por círculos preenchidos (●), onde três círculos (●●●) significam "Plenamente Atendido", dois círculos (●●) indicam "Parcialmente Atendido" e um círculo (●) significa "Não Atendido". Esta escala simplificada permite uma visualização clara e direta do nível de maturidade do sistema em relação a cada critério de qualidade.

Tabela 3 – Avaliação dos atributos de Qualidade (ISSO/IEC 25010)

Atributo	Status
Funcionalidade	●●●
Desempenho	●●●
Usabilidade	●●●
Compatibilidade	●●
Manutenibilidade	●●●
Segurança	●●

Fonte: Autor (2025).

Nota: ●●● = Plenamente atendido; ●● = Parcialmente atendido; ● = Não atendido. Os gestores da empresa parceira enfatizaram a vantagem da visibilidade completa do fluxo de trabalho, mencionando que 'pela primeira vez temos dados concretos sobre tempo médio de resolução e volume de chamados por categoria, o que permite tomar decisões baseadas em dados em vez de suposições'. Esta fala corrobora a hipótese inicial de que a informati-

zação traria maior transparência e controle gerencial.

A discussão dos resultados evidencia que o sistema desenvolvido atingiu seus objetivos principais. A abordagem incremental permitiu ciclos rápidos de feedback e validação, reduzindo riscos e retrabalho. A pontuação SUS satisfatória de 81 pontos, classificada como "excelente" e acima da média da literatura (Bangor et al., 2008), indica que a interface proposta é adequada para o perfil dos usuários-alvo. Este resultado corrobora os achados de Nielsen (1993) sobre a importância de interfaces focadas no fluxo de trabalho principal para a adoção de ferramentas por usuários finais.

O tempo médio de resposta de 1,7s para o carregamento das páginas principais está alinhado com as recomendações de Nielsen (1993) e com o requisito RNF03, que estabelece o limite de 2s para feedback do sistema, considerado ideal para manter o fluxo de pensamento do usuário. Contudo, é importante reconhecer as limitações do estudo. Além do ambiente controlado e com uma amostra pequena de usuários, outra limitação reside na validade ecológica do estudo. O protótipo foi testado com dados sintéticos e carga simulada, que podem não capturar completamente a complexidade e os imprevistos de um ambiente operacional real. Além disso, o estudo não investigou longitudinalmente a satisfação do cliente final após a implantação, focando-se na usabilidade para a equipe interna. Em um cenário de implantação real, fatores como volume maior de dados, integrações com outros sistemas (ERP, CRM) e políticas de segurança mais rigorosas poderiam impactar o desempenho e exigir ajustes nos requisitos. Futuros estudos poderiam implementar o sistema em produção e coletar métricas de satisfação do cliente (ex: Net Promoter Score - NPS) ao longo de um período determinado.

Considerações Finais

Conclui-se que o objetivo geral de desenvolver um sistema web para gerenciamento de chamados técnicos em pequenas empresas e

avaliar seus indicadores de eficiência, usabilidade e desempenho foi atingido. Os objetivos específicos também foram cumpridos: os requisitos foram eliciados e priorizados com a técnica MoSCoW; a modelagem foi realizada com diagramas UML; o protótipo foi implementado com sucesso; e os testes funcionais, de usabilidade e de desempenho foram conduzidos e validaram a aderência da solução ao contexto das MPEs.

Os resultados positivos sustentam a premissa inicial de que a informatização desse processo crítico pode gerar ganhos tangíveis para MPEs, validando a hipótese central deste trabalho. Como trabalhos futuros, recomenda-se a implementação de funcionalidades adicionais, como autenticação multifator, desenvolvimento de um aplicativo mobile para notificações em tempo real, construção de uma base de conhecimento integrada para defletir chamados simples e a exploração de técnicas de aprendizado de máquina para classificação e roteamento automático inicial de chamados. Adicionalmente, um estudo longitudinal em ambiente real de produção seria valioso para validar os benefícios operacionais aqui identificados.

Palavras-chave

Gestão de Suporte Técnico; Desenvolvimento Ágil. Qualidade de Software; PHP Laravel; Testes de Usabilidade.

Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 14724: informação e documentação – trabalhos acadêmicos – apresentação. Rio de Janeiro: ABNT, 2011.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6023: informação e documentação – referências – elaboração. Rio de Janeiro: ABNT, 2018.

BANGOR, A.; KORTUM, P. T.; MILLER, J. T. An empirical evaluation of the System Usability

Scale. *International Journal of Human-Computer Interaction*, v. 24, n. 6, p. 574-594, 2008.

BARDIN, L. *Análise de conteúdo*. São Paulo: Edições 70, 2011.

BECK, K. et al. *Manifesto para Desenvolvimento Ágil de Software*. 2001. Disponível em: <https://agilemanifesto.org/iso/ptbr/manifesto.html>. Acesso em: 10 ago. 2025.

BROOKE, J. SUS: a quick and dirty usability scale. In: JORDAN, P. W. et al. (Org.). *Usability Evaluation in Industry*. London: Taylor & Francis, 1996. p. 189-194.

FWLER, M. *Patterns of Enterprise Application Architecture*. Boston: Addison-Wesley, 2002.

ISO/IEC. ISO/IEC 25010: Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuRE) – System and software quality models. Geneva: ISO, 2011.

KRUG, S. *Não me faça pensar: uma abordagem de bom senso à usabilidade na web*. 3. ed. Rio de Janeiro: Alta Books, 2014.

LAKER, J. Implementing Role-Based Access Control in Laravel. *Laravel News*, 2023. Disponível em: <https://laravel-news.com/role-based-access-control-laravel>. Acesso em: 10 ago. 2025.

NIELSEN, J. *Usability Engineering*. San Francisco: Morgan Kaufmann, 1993.

PRESSMAN, R. S. *Engenharia de Software: uma abordagem profissional*. 7. ed. Porto Alegre: AMGH, 2011.

RUMBAUGH, J.; BOOCH, G.; JACOBSON, I. *The Unified Modeling Language Reference Manual*. 2. ed. Boston: Addison-Wesley, 2005.

SCHWABER, K.; SUTHERLAND, J. *Guia do Scrum: As Regras do Jogo*. 2020. Disponível em: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-PortugueseBR->



2.0.pdf. Acesso em: 10 ago. 2025.

SOMMERVILLE, I. Engenharia de Software. 9.
ed. São Paulo: Pearson, 2011