# SISTEMA WEB DE ABERTURA E GERENCIAMENTO DE CHAMADOS PARA A MANUTENÇÃO DE COMPUTADORES

**Henrique Gouveia Rodrigues, Matheus Borges Rodrigues, Hélio Lourenço Esperidião**

Colégio Técnico Antônio Teixeira Fernandes, R. Paraibuna, 75 - Jardim São Dimas - 12245-020 - São José dos Campos - SP, Brasil, henriq.gouveia@gmail.com, mathborgesrod@gmail.com, helioesperidiao@gmail.com

**Resumo**

A utilização de laboratórios de informática funcionais em instituições de ensino técnico é indispensável para o aprendizado prático. Contudo, a ausência de um sistema estruturado para a gestão de ordens de serviço pode comprometer a produtividade acadêmica, acarretando atrasos nos reparos e inatividade dos equipamentos. O objetivo deste trabalho é apresentar o desenvolvimento de um sistema web voltado ao diagnóstico e suporte à manutenção de computadores para o Colégio Técnico “Antônio Teixeira Fernandes”. A metodologia empregou o padrão de arquitetura *REST API*, *backend* em *Node.js* com *Express*, *frontend* dinâmico, padrão MVCS (*Model-View-Controller-Service*), banco de dados relacional *MySQL* gerenciado via *SQL* e rigorosas validações de *software*. Os resultados demonstraram a centralização e otimização do fluxo de chamados, a eliminação da perda de informações e o fornecimento de acompanhamento em tempo real para o coordenador. Conclui-se que o sistema melhora o gerenciamento dos laboratórios e contribui diretamente para a continuidade pedagógica.

**Palavras-chave**: Manutenção de Computadores, Abertura e Gerenciamento de Chamados, Laboratórios de Informática

**Área do Conhecimento:** Técnico em Informática

## Introdução

No cenário contemporâneo, a tecnologia da informação deixou de ser apenas um suporte operacional para se tornar o motor das estratégias organizacionais. A utilização da informática é fundamental para impulsionar a eficiência e o alinhamento das metas corporativas (Graelm, 2000). No ambiente educacional, especificamente em escolas técnicas, essa premissa se traduz na necessidade de laboratórios de informática funcionais, onde a manutenção preventiva e corretiva torna-se um pilar indispensável para o aprendizado (Santana, 2026).

A relevância da adoção de sistemas de gestão é corroborada por dados da IDG Research Services, que indicam que empresas que utilizam essas ferramentas crescem 35% mais rápido e apresentam uma produtividade 10% superior. No contexto de um laboratório de ensino, a ausência de um sistema de gestão de ordens de serviço compromete diretamente essa produtividade acadêmica, pois resulta no aumento dos custos de manutenção, equipamentos inativos por longos períodos e menor agilidade no processo de reparo (Martins, 2024). A manutenção é um processo fundamental para evitar falhas e prolongar a vida útil dos equipamentos (Ricce, 2023), dividindo-se em corretiva, preventiva e preditiva.

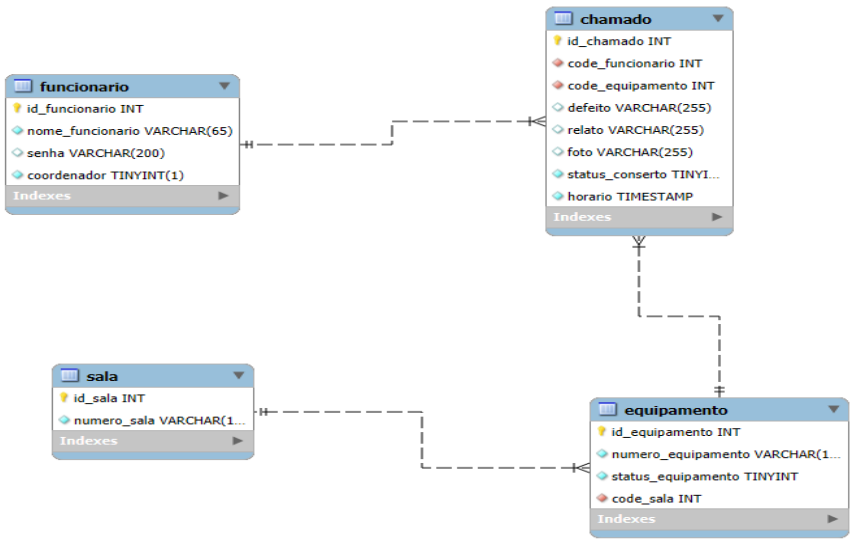
No Colégio Técnico "Antônio Teixeira Fernandes", o curso técnico em informática oferece aos alunos infraestrutura para aprendizado prático em diversas linguagens de programação. Entretanto, a instituição não possuía um sistema capaz de acompanhar as demandas de manutenção dos laboratórios e auxiliar o processo de abertura de chamados, resultando em atrasos e comprometendo a disponibilidade das máquinas.

Diante do exposto, o objetivo deste projeto é desenvolver um sistema web que proporcione aos professores do Colégio Técnico “Antônio Teixeira Fernandes” a oportunidade de reportar problemas de *hardware* e *software* nos laboratórios de informática do colégio ao coordenador do curso técnico em informática de maneira mais rápida e precisa, oferecendo maior controle e ciência dos problemas técnicos de cada equipamento e laboratório, facilitando o processo de abertura de chamados para a manutenção destes equipamentos.

## Metodologia

Para a estruturação da comunicação e integração do sistema, foi adotado o padrão de arquitetura *REST (Representational State Transfer*). As *REST APIs* utilizam restrições arquitetônicas padronizadas e métodos do protocolo *HTTP* (como *GET, POST, PUT* e *DELETE*) para interagir com os recursos da aplicação, cada qual identificado por um URI exclusivo (Goodwin, 2024). Essa abordagem garantiu a criação de *endpoints* de comunicação seguros, independentes e eficientes entre o cliente e o servidor.
No desenvolvimento da aplicação, o *backend* foi construído em *JavaScript* utilizando o ambiente de execução *Node.js*, otimizado pela máquina virtual V8, em conjunto com o *framework Express*. Essa combinação permitiu a construção de rotas dinâmicas e o processamento assíncrono de requisições.
Enquanto no *frontend*, foram empregadas linguagens web padrão (*HTML*, *CSS* e *JavaScript*) para gerar interfaces dinâmicas, responsivas e amigáveis, assegurando aos usuários telas intuitivas para abertura e gerenciamento de solicitações.
A organização do código-fonte seguiu o padrão de arquitetura *MVCS* (*Model-View-Controller-Service*). A camada Model trata do mapeamento e da estrutura dos dados, a View é responsável pela renderização da interface com o usuário, a camada *Controller* intermediou as requisições *HTTP* e direciona o fluxo da aplicação, e a camada *Service* concentra as regras de negócio complexas e a lógica de processamento do sistema. Essa separação em camadas garante alta reusabilidade, testabilidade e facilidade de manutenção.
Para a persistência de dados, utilizou-se o modelo de banco de dados relacional, seguindo a linguagem *SQL*. A estrutura compreende tabelas para funcionários, salas, equipamentos e chamados, conforme a figura 1. A Tabela “equipamento” relaciona-se com “sala” através de chave estrangeira, e a Tabela “chamado” estabelece vínculos relacionais que registram o funcionário solicitante, o equipamento com defeito, relato, foto e status do conserto.

Figura 1: Diagrama do Banco de Dados



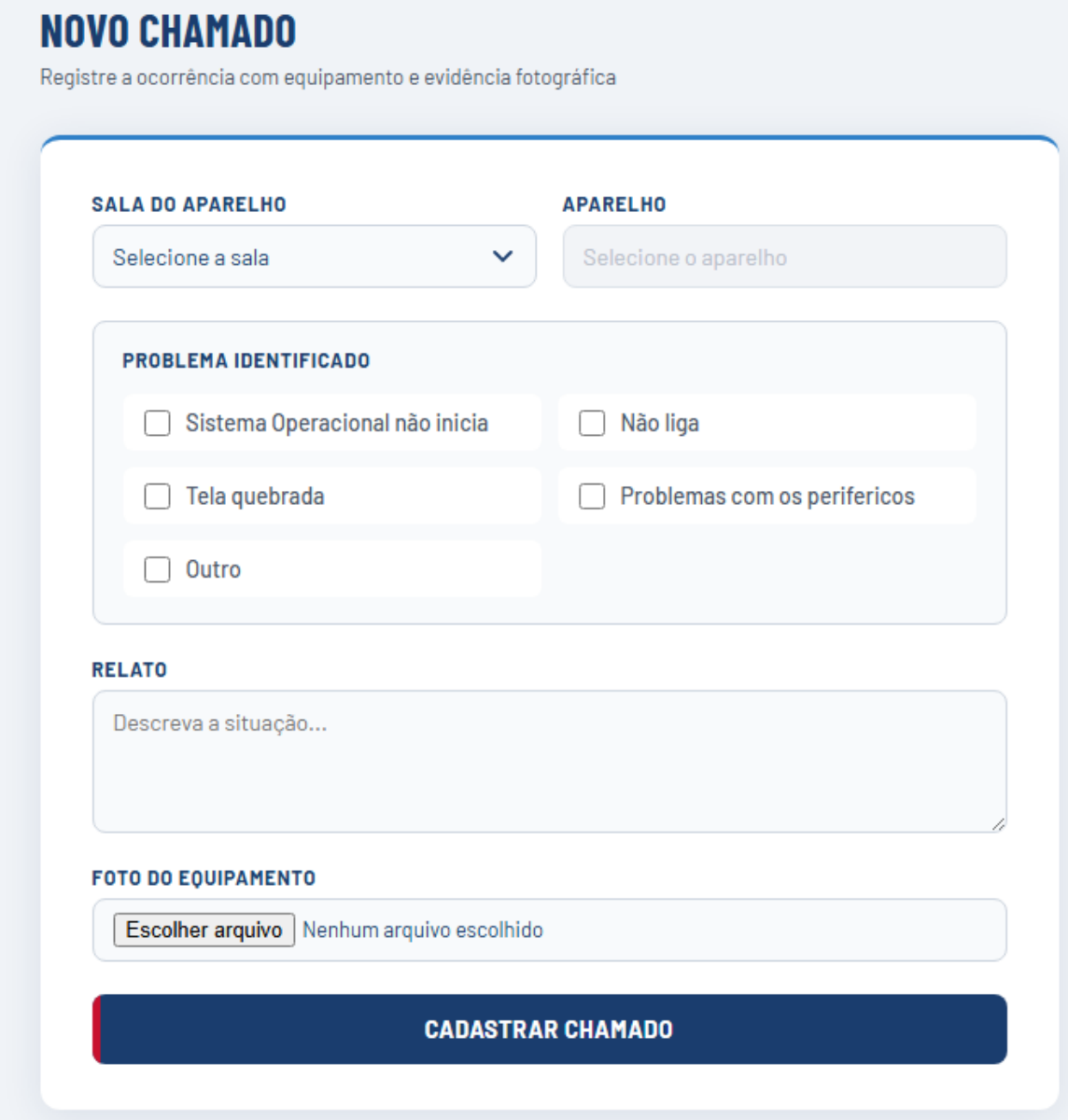

Fonte: Autores (2026)

Para garantir a integridade dos dados e a segurança operacional, foram implementadas validações de software rigorosas no *frontend* e no *backend*. No lado do cliente, formulários possuem verificação em tempo real quanto ao preenchimento de campos obrigatórios e formatos de entrada. No *backend*, middlewares inspecionam o *payload* das requisições, realizam a sanitização de entradas contra injeção *SQL* e aplicam travas de segurança por perfil de acesso (Autenticação e Autorização), restringindo funções administrativas exclusivamente a usuários com perfil de Coordenador.

**Resultados**

A implementação da plataforma web permitiu centralizar e organizar integralmente as solicitações de manutenção dos laboratórios. Antes da adoção do sistema, os alertas de falhas eram realizados por meio de comunicações informais, o que provocava perda sistemática de informações e atrasos. Com a ferramenta ativa, o processo de abertura de chamados passou a exigir o cadastro preciso do laboratório, equipamento e tipo de defeito, demonstrado na figura 2. Essa padronização eliminou a necessidade de triagem presencial prévia pelo suporte técnico.

Figura 2: Tela de Abertura de Chamado




Fonte: Autores (2026)

Adicionalmente, a tela de gerenciamento de chamados, apresentada na figura 3, permitiu ao coordenador acompanhar os status das solicitações em tempo real (Aberto, Em Andamento e

Concluído) e aplicar filtros por sala, professor ou equipamento.

Figura 3: Tela de Gerenciamento de Chamados




Fonte: Autores (2026)

## Discussão

Ao comparar os resultados obtidos com projeto de Rosa et al. (2023), destacam-se avanços relevantes na arquitetura e no escopo do sistema. Enquanto a solução proposta por Rosa et al. (2023) utiliza PHP no backend, o padrão MVC tradicional e possui foco genérico em ativos escolares, a presente proposta foi desenvolvida sob a arquitetura MVCS e focada 100% nos laboratórios de informática. Sob a perspectiva de desempenho, a adoção do ecossistema Node.js com Express supera o PHP ao utilizar um modelo assíncrono e orientado a eventos (I/O não bloqueante), lidando melhor com requisições simultâneas e evitando gargalos do processamento síncrono. Além disso, o uso do Node.js unifica o desenvolvimento ao utilizar JavaScript no frontend e no backend (CODIFICAR, 2023). No aspecto arquitetural, a evolução para o padrão MVCS traz uma vantagem crucial sobre o MVC convencional: a adição da camada Service isola as regras de negócio complexas do ambiente educacional dos controladores, garantindo maior modularidade, facilidade de manutenção e segurança nas validações de acesso por perfil.

## Conclusão

O presente trabalho atingiu o objetivo desenvolver um sistema *web* para facilitar e agilizar o processo de identificação de problemas nos equipamentos dos laboratórios de informática do Colégio Técnico “Antônio Teixeira Fernandes”, bem como otimizar a abertura e o gerenciamento dos chamados de manutenção, proporcionando aos professores a oportunidade de relatar falhas técnicas e abrir solicitações para os equipamentos de maneira mais rápida e precisa, assim como oferecer ao coordenador um acompanhamento amplo e em tempo real sobre a situação dos chamados, dos laboratórios e do estado de cada equipamento, apresentando gráficos e relatórios referentes a situação dos mesmos.
Em trabalhos futuros, sugere-se a integração de QR Codes fixados em cada computador dos laboratórios, permitindo que o professor abra o chamado de maneira instantânea ao escanear o código do equipamento pelo celular

## Referências

GRAEML, Alexandre Reis. Sistemas de Informações: O alinhamento da estratégia de TI com a estratégia corporativa. São Paulo: Atlas, 2000.

EMPRESAS com sistema de gestão têm mais chance de crescer. Terra, [S. l.], 2015. Dino. Disponível em: https://www.terra.com.br/noticias/dino/empresas-com-sistema-de-gestao-tem-mais-chance-de-crescer,dea5cc85a26380c03a68d3d8b789e352ezy728s0.html. Acesso em: 25 ago. 2026.

MARTINS. Gestão eficiente de laboratórios de informática em escolas e universidades: ganhos de tempo, custos, segurança e agilidade. ENG, 5 jul. 2024. Disponível em: https://www.eng.com.br/artigo.cfm?id=7791&post=gestao-eficiente-de-laboratorios-de-informatica-em-escolas-e-universidades:-ganhos-de-tempo,-custos,-seguranca-e-agilidade. Acesso em: 25 ago. 2026.

*RICCE.* Qual é a importância da manutenção e seu impacto na qualidade de produtos e serviços. *Blog Sismetro, 2023. Disponível em:* https://blog.sismetro.com/post/17/qual-e-a-importancia-da-manutencao-e-seu-impacto-na-qualidade-de-produtos-e-servicos. *Acesso em: 25 ago. 2026.*

GOODWIN, Michael. What is an API (application programming interface)? IBM, 9 apr. 2024. Disponível em: https://www.ibm.com/br-pt/think/topics/api. Acesso em: 25 ago. 2026.

ROSA, [*et al.* [SISTEMAS DE GESTÃO DE MANUTENÇÃO DE ATIVOS PARA ESCOLAS]. In: ENCONTRO DE INICIAÇÃO CIENTÍFICA DA UNIVAP, XXVII., 2023, São José dos Campos. Anais [...]. São José dos Campos: Universidade do Vale do Paraíba, 2023. Disponível em: https://www.inicepg.univap.br/cd/INIC_2023/anais/arquivos/RE_0376_0751_01.pdf. Acesso em: 25 ago. 2026.