DESENVOLVIMENTO DE APLICAÇÃO WEB PARA GESTÃO E AUTOMAÇÃO DO FLUXO DE APROVAÇÃO DE SOLICITAÇÕES DE COMPRAS

Raphael Nicolas Leite Santos, Vinicius Alendes Sousa, Lucas Ferreira Stabile David, Wagner dos Santos Clementino de Jesus.

Colégio Técnico Antônio Teixeira Fernandes / Fundação Vale Paraibana de Ensino, Avenida Paraibuna, 75, Centro - 12245-020 - São José dos Campos-SP, Brasil, lucas.stabileunivap@gmail.com, vinicius.alendes.souza@gmail.com, santoos.univap30@gmail.com, wsantoscj@gmail.com.

Resumo

O processo de compras em organizações envolve solicitações, seleção de produtos e fornecedores e, em determinados casos, aprovações definidas conforme níveis hierárquicos. Quando realizado manualmente, esse processo pode gerar burocracia e dificultar o rastreamento. Este trabalho apresenta um sistema web para gerenciamento de compras com autorizações pré-definidas, implementado com HTML, CSS, JavaScript, Bootstrap, PHP, Slim Framework e MySQLn. A metodologia foi baseada no Design Sprint, apoiando o levantamento das necessidades, modelagem, prototipação, implementação e validação das funcionalidades. Como resultados, foram obtidos módulos para gerenciamento de usuários, produtos, fornecedores e pedidos, um fluxo de aprovação hierárquica conforme o valor das solicitações e o vínculo entre produtos e fornecedores, com histórico completo das movimentações. A aplicação atende à proposta de centralizar e organizar o processo de compras, contribuindo para maior controle, rastreabilidade e agilidade.

Palavras-chave: Gestão de compras. Automação. Aprovação hierárquica. PHP. MySQL.
Área do Conhecimento: Ciências Exatas e da Terra / Técnico em Informática, Ciência da Computação.

Introdução

O setor de compras exerce papel relevante no funcionamento das organizações, pois está relacionado à aquisição de materiais na quantidade, no momento e nas condições adequadas. Um processo de compras desorganizado pode provocar atrasos, aquisição inadequada de itens e dificuldades de comunicação entre os setores envolvidos. O planejamento e o controle das atividades de aquisição são importantes para o funcionamento eficiente da organização (Francischini; Gurgel, 2013).

Em processos internos que dependem de solicitações e autorizações, outro desafio está na definição de quem deve analisar cada pedido. Quando essa decisão é realizada manualmente, podem ocorrer atrasos, falta de rastreabilidade e concentração de solicitações em determinados responsáveis (Barroso, 2013). Nesse contexto, a utilização de sistemas de informação pode centralizar os dados e automatizar regras de negócio, reduzindo intervenções desnecessárias e tornando o fluxo mais organizado (Pressman; Maxim, 2021).

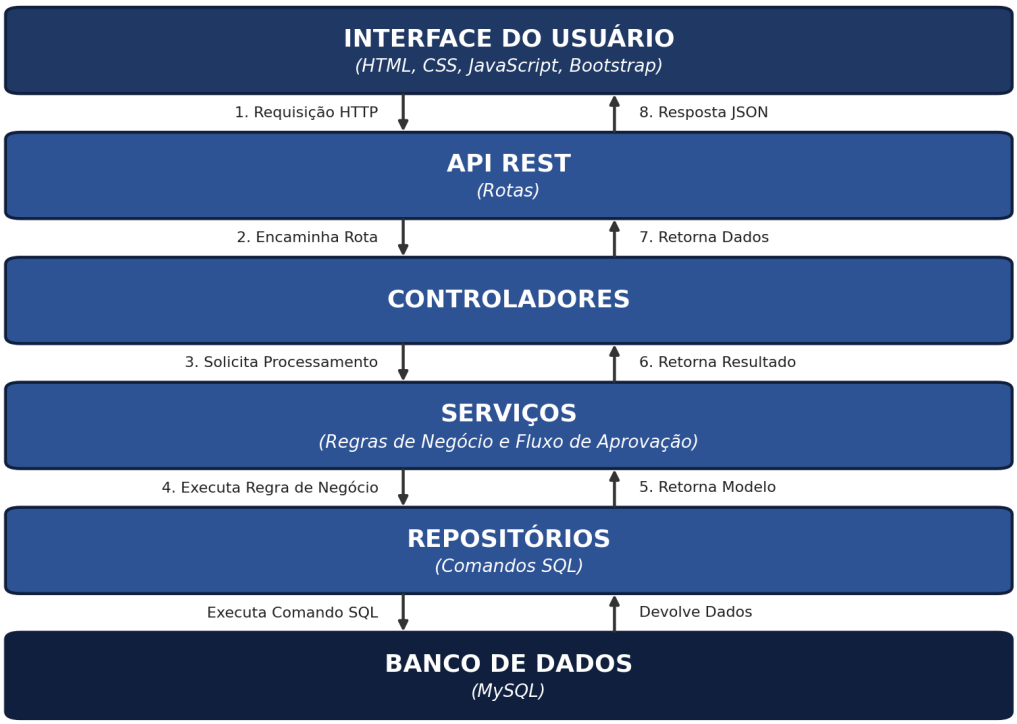
Diante desse cenário, este trabalho teve como objetivo desenvolver uma aplicação web para gerenciamento de solicitações de compras, permitindo o cadastro de usuários, produtos e fornecedores e o registro de pedidos de compra. A solução também foi projetada para encaminhar as solicitações conforme regras de autorização previamente definidas, registrar as decisões tomadas e manter o histórico das movimentações, buscando proporcionar maior controle e transparência ao processo.

Metodologia

O desenvolvimento do sistema foi conduzido como um projeto aplicado de desenvolvimento tecnológico, utilizando a metodologia Google Design Sprint para organizar as etapas de compreensão do problema, geração de ideias, tomada de decisões, prototipação e testes (Knapp; Zeratsky; Kowitz, 2017). A abordagem permitiu estruturar as funcionalidades e validar os fluxos da aplicação ao longo do desenvolvimento, reduzindo retrabalho e facilitando o refinamento da solução.
A aplicação foi construída em arquitetura web, dividida entre front-end e back-end. A interface foi desenvolvida com HyperText Markup Language (HTML), Cascading Style Sheets (CSS) e JavaScript (Robbins, 2025), com apoio do framework Bootstrap (Silva, 2023; Marques, 2022). O back-end foi implementado em PHP, utilizando o Slim Framework, um micro-framework PHP direcionado à criação

de APIs REST (Sunardi; Suharjito, 2019), e o Composer para o gerenciamento das dependências. O armazenamento foi realizado em banco de dados relacional MySQL, estruturado em tabelas de usuários, hierarquia, produtos, fornecedores, vínculo produto-fornecedor, pedidos, itens de pedido, aprovações, solicitações de acesso e observações privadas, relacionadas por meio de chaves primárias e estrangeiras e manipuladas por meio de SQL (Silberschatz; Korth; Sudarshan, 2020). Do ponto de vista da engenharia de software, a aplicação foi estruturada em camadas de rotas, controladores, serviços e repositórios (Pressman; Maxim, 2021), conforme ilustrado na Figura 1. A interface do usuário envia requisições por meio da API REST, que são encaminhadas às camadas de controle e de serviço responsáveis pelas regras de negócio, como o fluxo de aprovação, enquanto a camada de repositórios isola os comandos SQL executados no banco de dados.

Figura 1 – Arquitetura em camadas do sistema.

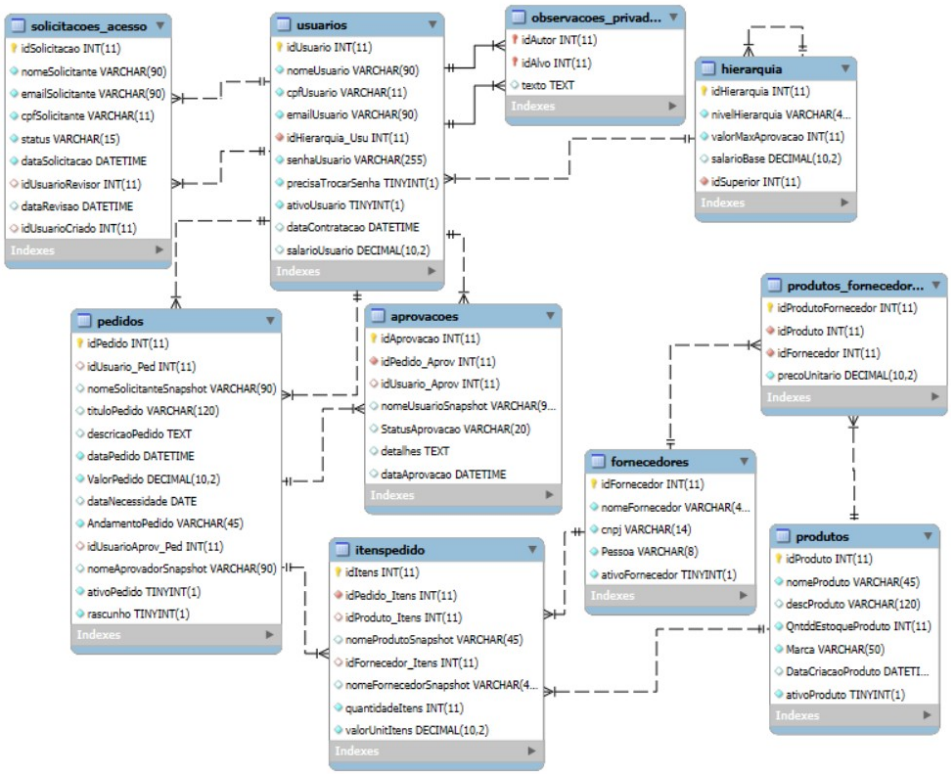


Fonte: Autores, 2026.

A solução foi organizada em módulos de Produtos, Fornecedores, Usuários e Pedidos, disponibilizados por meio dessa API REST e consumidos pela interface web. Essa organização em camadas contribuiu para separar responsabilidades e facilitar a manutenção do sistema.

O modelo de dados, ilustrado na Figura 2, é composto por dez tabelas relacionadas. Os usuários são vinculados à hierarquia de aprovação, que define os níveis, seus limites de valor e o respectivo nível superior. Os pedidos mantêm vínculos com o usuário solicitante, o aprovador responsável, as aprovações e os itens do pedido. Para preservar o histórico mesmo após alterações nos cadastros, são armazenados dados de referência, como os nomes do solicitante, do aprovador, dos produtos e dos fornecedores. O vínculo entre produtos e fornecedores também permite definir um preço específico para cada combinação.

Figura 2 – Modelo de dados do sistema.



Fonte: Autores, 2026.

O fluxo de pedidos foi implementado considerando os níveis hierárquicos, os limites de aprovação e a relação entre níveis superiores definida na hierarquia. Ao criar uma solicitação, o sistema identifica o menor nível hierárquico cujo limite seja suficiente para o valor total do pedido. Caso o valor ultrapasse o limite permitido para determinado nível, a solicitação é escalonada para o nível superior correspondente. Essa relação também é utilizada em regras de autorização que exigem a atuação do superior responsável. Quando existem usuários aptos no mesmo nível, a distribuição entre esses usuários considera a quantidade de pedidos pendentes, reduzindo a concentração de solicitações em um único responsável. Usuários de níveis superiores também podem intervir na solicitação, de acordo com as regras de autorização definidas no sistema.

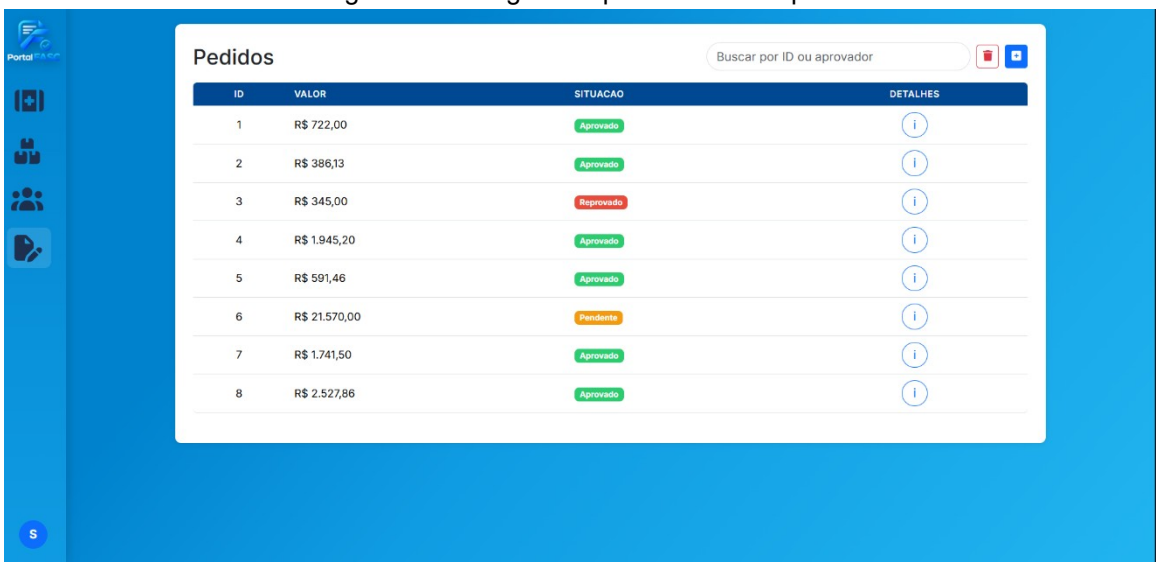
Também foi implementado o vínculo entre produtos e fornecedores, permitindo que cada combinação possua seu próprio preço e que um pedido reúna itens de diferentes fornecedores, com o valor total calculado a partir dos preços cadastrados. A validação foi realizada por meio de testes funcionais em ambiente de desenvolvimento local, utilizando dados simulados com elevado volume de usuários, produtos e pedidos. Foram verificados os cadastros, a criação e edição de pedidos, o direcionamento automático conforme o nível hierárquico, o histórico de aprovações e as operações de exclusão e restauração. Uma funcionalidade era considerada validada quando produzia o resultado esperado sem erros de execução.

Resultados

O desenvolvimento resultou em uma aplicação funcional para centralização do processo de compras, composta pelos módulos de Produtos, Fornecedores, Usuários e Pedidos. Os registros são disponibilizados por uma API REST em PHP com *Slim Framework* e apresentados por uma interface web em HTML, CSS e JavaScript.

O sistema diferencia dois perfis principais de utilização. O funcionário pode se autenticar, consultar seus pedidos e criar novas solicitações. O administrador possui recursos de gerenciamento e análise dos pedidos, podendo atuar no fluxo de aprovação conforme suas permissões. Essa separação organiza as responsabilidades e evita que operações administrativas fiquem disponíveis indistintamente a todos os usuários.

Figura 3 – Listagem de pedidos de compra.



Fonte: Autores, 2026.

Ao acessar os detalhes de um pedido, o sistema apresenta informações do solicitante, valor, data, aprovador atual e situação, além dos itens solicitados e do histórico de aprovação. Essa organização concentra em uma única tela as informações necessárias para acompanhamento da solicitação, evitando que o usuário precise consultar diferentes áreas do sistema para compreender o estado do pedido.

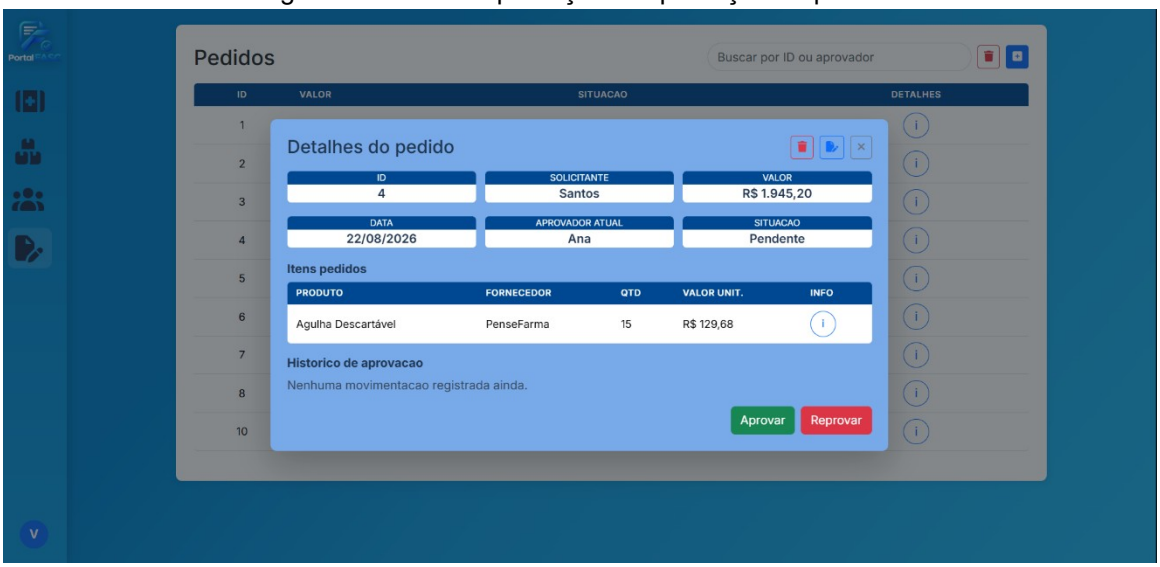
Figura 4 – Detalhes de um pedido e histórico de movimentações.



Fonte: Autores, 2026.

A Figura 4 apresenta o pedido pendente no momento da análise pelo responsável. Nessa etapa, ficam disponíveis as ações de aprovação ou reprovação, enquanto as informações do pedido permanecem visíveis para apoiar a decisão. Após uma decisão, a movimentação é registrada no histórico, permitindo acompanhar as alterações de situação ocorridas durante o fluxo.

Figura 5 – Tela de aprovação e reprovação de pedido.



Fonte: Autores, 2026.

Outro resultado relevante foi a implementação do vínculo entre produtos e fornecedores com preços independentes. Cada item do pedido mantém o fornecedor e o preço utilizados no momento da solicitação, permitindo reunir produtos de fornecedores diferentes e calcular automaticamente o valor total. Também foram implementadas operações de exclusão em duas etapas para produtos, fornecedores, usuários e pedidos, com envio inicial para uma lixeira e possibilidade de restauração ou remoção definitiva.

O fluxo de pedidos mantém um histórico de aprovação, reprovação, edição e reabertura, registrando o usuário responsável por cada ação. A solução também utiliza autenticação por tokens JWT, armazenamento de senhas com *hash* e validação das regras de acesso no próprio back-end, protegendo informações sensíveis de acordo com as permissões definidas.

Discussão

Os resultados obtidos indicam que a centralização das solicitações em uma aplicação web pode reduzir a dependência de controles dispersos e facilitar o acompanhamento do processo de compras. A divisão da aplicação em *front-end*, *back-end* e banco de dados favoreceu a separação das responsabilidades e permitiu organizar as regras de negócio de forma independente da interface, aspecto importante para manutenção e evolução do sistema (Pressman; Maxim, 2021).

O mecanismo de encaminhamento dos pedidos conforme o nível hierárquico e o valor da solicitação constitui um dos principais resultados funcionais da solução. A definição automática do responsável reduz a necessidade de distribuição manual das solicitações, enquanto a distribuição entre usuários do mesmo nível ajuda a evitar a concentração de pedidos pendentes em um único responsável. A possibilidade de intervenção de níveis superiores acrescenta flexibilidade ao processo.

A manutenção do histórico de aprovações, reprovações, edições e reaberturas amplia a rastreabilidade do processo. Em vez de registrar somente a situação atual, a aplicação preserva as movimentações realizadas, permitindo identificar as ações efetuadas sobre cada pedido. Esse recurso é particularmente relevante em processos administrativos, nos quais a recuperação do contexto de uma decisão pode ser necessária para acompanhamento ou auditoria.

A estrutura de produtos, fornecedores e preços também contribui para a organização das solicitações. Como os valores ficam associados às combinações entre produto e fornecedor, o pedido mantém o preço utilizado no momento da solicitação e não depende de um valor digitado livremente pelo usuário. Essa abordagem reduz inconsistências e mantém maior coerência entre os dados cadastrados e os valores utilizados no processo de compra.

Apesar dos resultados positivos, a validação foi realizada em ambiente de desenvolvimento local e não incluiu testes automatizados, testes de carga ou uso simultâneo por múltiplos usuários em produção. Assim, os resultados demonstram a viabilidade funcional da proposta, mas avaliações

futuras em condições reais podem fornecer evidências adicionais sobre desempenho, escalabilidade e comportamento do sistema diante de maior volume de solicitações.

Conclusão

O desenvolvimento da aplicação web alcançou o objetivo de facilitar o processo de solicitação e aprovação de compras em uma organização, com potencial para reduzir a burocracia associada aos processos manuais. A automatização da atribuição de aprovadores conforme os níveis hierárquicos, os limites de aprovação e o valor dos pedidos tornou o fluxo mais organizado, enquanto o histórico das movimentações ampliou a rastreabilidade das decisões.

A utilização de PHP, *Slim Framework*, JavaScript e MySQL mostrou-se adequada ao desenvolvimento da aplicação, enquanto a modelagem relacional permitiu organizar os dados e preservar suas relações. Os recursos de histórico e lixeira acrescentaram rastreabilidade e segurança operacional ao fluxo.

Como continuidade, recomenda-se ampliar os testes automatizados e de carga, além de desenvolver relatórios gerenciais e notificações para pedidos pendentes. Dessa forma, o sistema poderá evoluir para oferecer indicadores mais completos sobre o processo de compras e apoiar ainda mais a tomada de decisão da organização.

Referências

BARROSO, Regiscélio Fonteles. **Análise estratégica da gestão de compras em uma empresa do ramo da construção civil no estado do Ceará**. Fortaleza: Faculdade Cearense, 2013.

FRANCISCHINI, Paulino G.; GURGEL, Floriano do Amaral. **Administração de materiais e do patrimônio**. 2. ed. São Paulo: Cengage Learning, 2013.

KNAPP, Jake; ZERATSKY, John; KOWITZ, Braden. **Sprint: o método usado no Google para testar e aplicar novas ideias em apenas cinco dias**. Rio de Janeiro: Intrínseca, 2017.

MARQUES, D. **Bootstrap 5: guia rápido para iniciantes.** 1. ed. Lagoa Santa: Alfahelix, 2022.

PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de software: uma abordagem profissional**. 9. ed. Porto Alegre: AMGH, 2021.

ROBBINS, Jennifer Niederst. **Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Images.** 6. ed. Sebastopol: O'Reilly Media, 2025.

SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. **Sistema de Banco de Dados**. 7. ed. Rio de Janeiro: LTC, 2020.

SILVA, Maurício Samy. **Bootstrap: um guia completo para construir aplicativos responsivos, modernos e eficientes**. São Paulo: Novatec Editora, 2023.

SUNARDI, Andri; SUHARJITO. **MVC Architecture: A Comparative Study Between Laravel Framework and Slim Framework in Freelancer Project Monitoring System Web Based**. Procedia Computer Science, v. 157, p. 134-141, 2019. Disponível em: https://www.sciencedirect.com/science/article/pii/S1877050919310683. Acesso em: 24 ago. 2026.