

DESENVOLVIMENTO DE UM SOFTWARE DE GESTÃO PARA UMA EMPRESA MADEIREIRA NO SUL DO PARANÁ

Guilherme Prigol Kramer¹
Alexandre Ruan Base²
Luiz Odair Paulek Filho³
Claudio Jorge Gomes da Rocha Junior⁴

RESUMO: O trabalho teve como tema o desenvolvimento de um *software* de gestão personalizado para uma empresa madeireira no sul do Paraná, com foco no atendimento às necessidades específicas da organização. Tratou-se de um estudo aplicado que utilizou tecnologias atuais, como *TypeScript*, *NestJS*, *React*, *Vite*, *Electron* e *PostgreSQL*, para a construção de uma aplicação *desktop* multiplataforma. A metodologia envolveu o levantamento de requisitos funcionais e não funcionais, elaboração de diagramas *UML*, prototipagem no *Figma* e codificação no *Visual Studio Code*. A solução foi planejada para operar localmente, sem dependência da internet, permitindo o gerenciamento de produtos, negociações, contas e usuários. Os resultados demonstraram um sistema funcional e validado, com índices de satisfação superiores a 86% entre os usuários em múltiplas instâncias conectadas ao mesmo banco de dados, demonstrando a aplicação prática dos conhecimentos de desenvolvimento de *software* e engenharia de requisitos na criação de soluções sob medida.

Palavras-chave: Software personalizado. Setor madeireiro. Aplicação desktop.

ABSTRACT: The study focused on the development of custom management software for a lumber company in southern Paraná, focusing on meeting the specific needs of the organization. It was an applied study that used current technologies, such as *TypeScript*, *NestJS*, *React*, *Vite*, *Electron*, and *PostgreSQL*, to build a cross-platform desktop application. The methodology involved gathering functional and non-functional requirements, creating *UML* diagrams, prototyping in *Figma*, and coding in *Visual Studio Code*. The solution was designed to operate locally, without internet dependency, allowing for the management of products, negotiations, accounts, and users. The results demonstrated a functional and validated system, with satisfaction rates exceeding 86% among users across multiple instances connected to the same database, showcasing the practical application of software development and requirements engineering knowledge in creating tailored solutions.

Keywords: Custom software. Lumber industry. Desktop application.

INTRODUÇÃO

A digitalização de processos se mostra como um grande auxílio para a gestão de empresas. Ela permite automatizar tarefas que antes eram manuais, reduzindo o tempo de execução das atividades e aumentando a eficiência dos processos. Além disso, contribui para a

¹Tecnólogo em Análise e Desenvolvimento de Sistemas, IFPR.

²Estudante de Tecnologia em Análise e Desenvolvimento de Sistemas, IFPR.

³Estudante de Tecnologia em Análise e Desenvolvimento de Sistemas, IFPR.

⁴Foutor em Sociologia, professor orientador, IFPR.

diminuição de custos operacionais, facilita a análise de dados em tempo real, eleva a produtividade e pode reduzir ou até eliminar a necessidade de intervenção humana. Com isso, as empresas podem concentrar seus esforços em tarefas estratégicas de maior valor. Outros benefícios possíveis são a melhoria na experiência do cliente, a viabilização da personalização de soluções, o aumento da segurança da informação e, como consequência de todos os benefícios citados, o crescimento da receita (SEBRAE, 2023).

Em virtude da evidente exigência do mercado por agilidade e de todos os benefícios citados ao redor do uso de tecnologias, uma empresa madeireira no sul do Paraná estava em busca de um *software* para digitalizar alguns de seus processos: compras, vendas, contas a pagar, contas a receber e controle de produtos. Porém, foi relatado que os sistemas de gestão disponíveis no mercado não a satisfaziam, além de contemplarem diversas funcionalidades que não seriam realmente utilizadas, tornando o uso desses sistemas em uma tarefa desgastante, frustrante e cheia de elementos desnecessários.

Diante desse contexto, o presente trabalho teve como objetivo desenvolver um *software* de gestão empresarial personalizado, que atendesse aos requisitos definidos pelos responsáveis da empresa. O *software* consistiu numa aplicação *desktop*⁵, feita com a linguagem de programação *TypeScript*, utilizando as tecnologias *Vite*, *NestJS* e *Electron*. Este trabalho teve como objetivos específicos: identificar os requisitos desejados pelos usuários da empresa, desenvolver o *software* atendendo aos requisitos levantados, implementar o *software* na empresa e, mensurar a satisfação dos representantes em relação ao produto final.

SOFTWARE ERP

ERP é a sigla de “*Enterprise Resource Planning*” (Planejamento de Recursos Empresariais). Um *software* ERP tem como objetivo centralizar as informações dos setores de uma empresa, facilitando o acesso em tempo real a informações relevantes (Senior Sistemas, 2025). Cada setor da empresa previsto no sistema pode ter uma área dentro dele para a sua gestão, como estoque, vendas, relacionamento com o cliente, entre outros. Essa integração de setores é o que caracteriza uma aplicação do tipo ERP, a diferenciando dos demais tipos de sistemas de gestão, que são especialistas em um único setor ou processo da empresa.

Quando o assunto de *softwares* ERP é abordado, existem duas linhas de frente principais: genéricos e personalizados. O ERP genérico é feito para atender vários tipos de empresas e

⁵ Aplicações *desktop* são sistemas que funcionam sendo instalados em computadores. Se diferem de aplicações *web*, que são acessadas pela internet e de aplicações *mobile*, que são instaladas em *smartphones*.

setores, com funcionalidades que servem para diferentes modelos de negócio (Milani, 2024). De acordo com Pires (2023), um ERP genérico se adapta a vários tipos de empresas sem depender de um nicho de mercado específico. Já um ERP personalizado, por sua vez, é desenvolvido com base nas necessidades e processos da única empresa que o utilizará.

A implementação de um ERP gera reestruturações em processos já estabelecidos, o que é positivo, pois abre espaço para reorganizar processos que estavam desorganizados (Santos, 2024). Porém, isso exige um conhecimento adequado dos processos da empresa. Ao implementar um ERP, é crucial entender as necessidades específicas da organização e dos setores envolvidos. Isso exige o entendimento das regras de negócio existentes, das políticas da empresa e das expectativas dos usuários finais (Damo e Santander, 2024). Balieiro e Pinto (2024) destacam, com base em dados apresentados por Bambazek, Groher e Senff (2023), que 45% dos clientes consideram que os *softwares* entregues não atendem às suas necessidades, principalmente devido a um levantamento de requisitos inadequado. Isso evidencia o impacto negativo de sistemas que não acompanham as demandas reais da organização.

Armstrong (2024), em seu estudo sobre fatores de estresse no uso de sistemas ERP, afirma que a complexidade é a maior causa de *technostress* entre todas as variáveis analisadas na sua pesquisa. Com base nesse resultado, a autora destaca a importância de interfaces simples e intuitivas, bem como a remoção de elementos desnecessários, para impedir o aumento do estresse ocupacional dos usuários. A sobrecarga de informações também é apontada como fator determinante para o nível de estresse enfrentado, o que reforça mais uma vez o impacto negativo de elementos desnecessários na interação do usuário com o sistema.

METODOLOGIA

Este trabalho se caracterizou como uma pesquisa aplicada, voltada ao desenvolvimento prático do projeto, a partir da identificação do problema, utilizando tecnologias disponíveis no mercado. A empresa selecionada para o estudo permitiu a execução da pesquisa, mediante acesso às informações do negócio, a fim de personalizar o *software*, incluindo a condição de manter a confidencialidade de sua identificação e dados sensíveis.

Para o planejamento e documentação do *software*, foram utilizados os conceitos da UML “Diagrama de Casos de Uso” e “Diagrama de Entidade e Relacionamento”, feitos com a ferramenta “Draw.io”. Também foram aplicados conceitos da engenharia de requisitos como “requisitos funcionais” e “requisitos não funcionais”. Além disso, criou-se um protótipo visual com a ferramenta Figma, para ajudar a definir o design estético do sistema, planejar o fluxo de

navegação entre as páginas e, ainda, auxiliar a validar as funcionalidades planejadas e os requisitos definidos.

O *software* foi construído com tecnologias e conceitos de desenvolvimento web, como a linguagem de programação *TypeScript*, *Backend* com o *framework* *NestJS*, *Frontend* com a ferramenta *Vite* e banco de dados *PostgreSQL*. Ademais, foi utilizado o *framework* “*Electron*” para empacotar e executar a aplicação como um *software desktop*.

As ferramentas escolhidas foram selecionadas com base em critérios de eficiência, compatibilidade, comunidade ativa, afinidade com o desenvolvedor e alinhamento com os objetivos do projeto, como principais ferramentas utilizadas no projeto podemos citar: *Draw.io* que é um editor de diagramas e fluxogramas gratuito que funciona de maneira online no navegador e *offline* com aplicativo próprio; *Figma* é uma ferramenta de design gráfico que ajuda equipes a criar produtos de forma mais eficiente; *Git* é um sistema de controle de versões *open-source* projetado para lidar com projetos, desde pequenos a enormes, com velocidade e eficiência; *Typescript* é uma linguagem de programação de código aberto fortemente tipada baseada em *JavaScript*; *NestJS* é um *framework* de *backend* para *Node.js* que facilita a construção de aplicações *server-side* de forma organizada, eficiente e escalável; *PostgreSQL* foi usado como banco de dados do sistema proposto; *Vite* é uma *build tool* de *Frontend* com foco em promover uma experiência de desenvolvimento mais rápida e simples para projetos *web* modernos; *Electron* é um *framework* que permite o desenvolvimento de aplicações *desktop* multiplataforma utilizando tecnologias *web*.

RESULTADOS

Os requisitos levantados com os representantes da empresa foram a base do desenvolvimento do sistema, a fim de concluir o objetivo específico 2 deste trabalho.

Requisitos não funcionais descrevem como o sistema deve se comportar. Eles não são diretamente ligados a funcionalidades específicas, mas influenciam a experiência do usuário, a performance, a segurança, a escalabilidade, entre outros aspectos. Em seguida estão listados os requisitos não funcionais do sistema: RNF_01: O sistema deve ser uma aplicação desktop compatível com o *Windows 11*; RNF_02: Deve funcionar de maneira desconexa e independente da internet, comunicando-se apenas através da rede interna da empresa; RNF_03: Deve ser multiusuário; RNF_04: Deve utilizar a mesma base de dados em todas as instâncias do aplicativo, mesmo quando executadas em computadores diferentes; RNF_05: As senhas dos usuários devem ser armazenadas de maneira criptografada; RNF_06: Deve permitir a divisão

dos usuários em cargos, incluindo “Gestor de produtos”, “Gestor comercial” e “Administrador”.

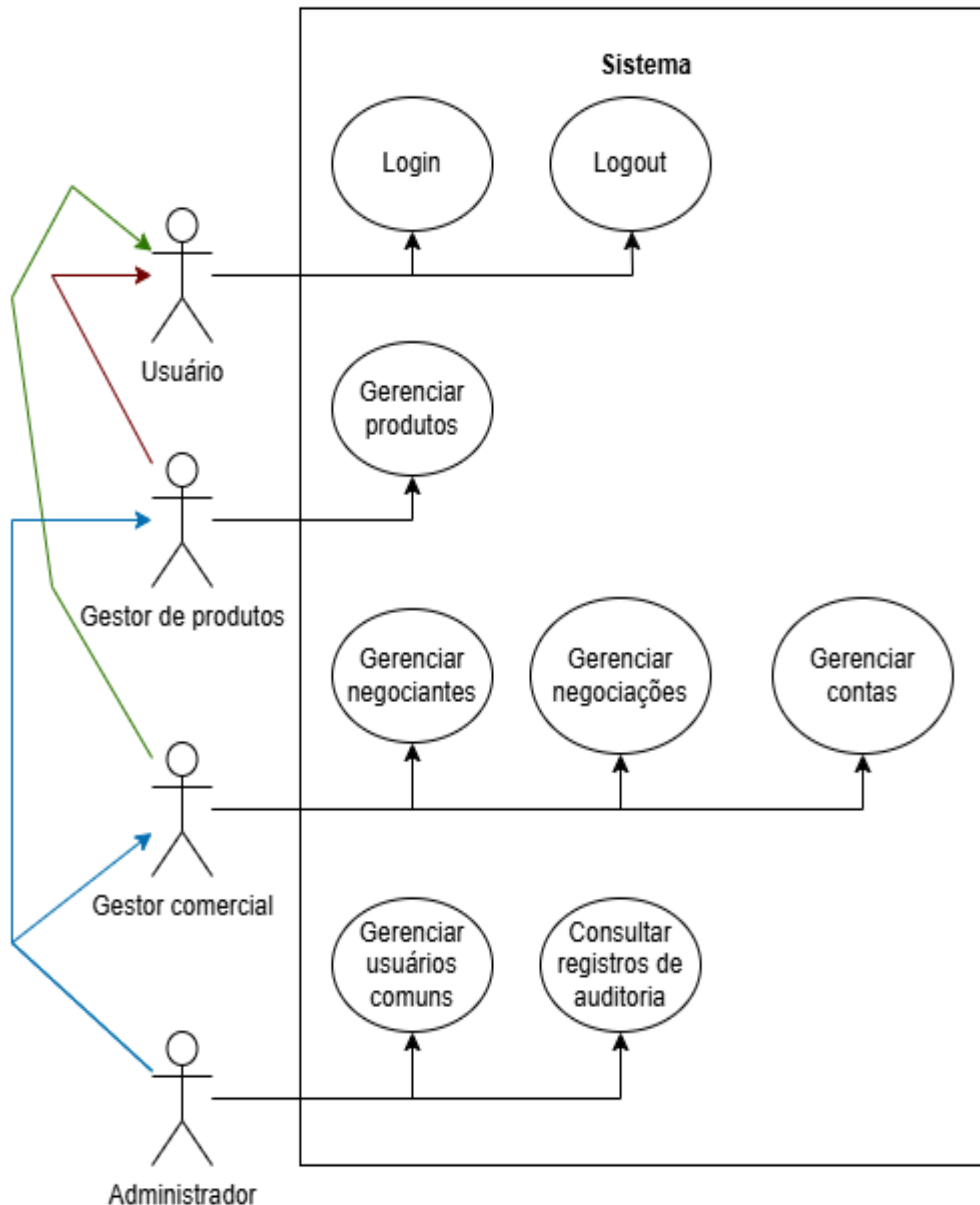
Os requisitos funcionais descrevem o que o sistema deve fazer, representando suas funcionalidades e comportamentos esperados. Eles estão diretamente relacionados aos objetivos do *software* e definem as tarefas que ele deve ser capaz de realizar. Em seguida estão listados os requisitos funcionais do sistema: RF_USU_01: Permitir que administradores cadastrem usuários; RF_USU_02: Permitir que administradores visualizem os usuários cadastrados; RF_USU_03: Permitir que administradores editem usuários; RF_USU_04: Permitir que administradores desativem usuários, impedindo o login e invalidando sessões ativas; RF_USU_05: Permitir que administradores reativem usuários; RF_USU_06: Permitir que cada usuário edite as informações do próprio perfil, com exceção do cargo, como nome, nome de usuário e senha; RF_AUT_01: Permitir que o usuário realize *login* com nome de usuário e senha; RF_AUT_02: Permitir que o usuário realize *logout*; RF_PROD_01: Permitir o cadastro de produtos; RF_PROD_02: Permitir a visualização de produtos; RF_PROD_03: Permitir a edição de produtos; RF_PROD_04: Permitir a exclusão de produtos, incluindo todas as variações e negociações vinculadas; RF_NEGT_01: Permitir o cadastro de negociantes; RF_NEGT_02: Permitir a visualização dos negociantes cadastrados; RF_NEGT_03: Permitir a edição dos negociantes; RF_NEGT_04: Permitir a exclusão de negociantes, removendo também as negociações vinculadas; RF_NEGC_01: Permitir o cadastro de negociações; RF_NEGC_02: Permitir a visualização das negociações cadastradas; RF_NEGC_03: Permitir a edição das negociações cadastradas; RF_NEGC_04: Permitir a exclusão das negociações cadastradas; RF_CONT_01: Permitir o cadastro de contas avulsas; RF_CONT_02: Permitir a visualização das contas cadastradas; RF_CONT_03: Permitir a edição das contas cadastradas; RF_CONT_04: Permitir a exclusão das contas cadastradas; RF_BKP_01: Permitir que administradores realizem *backup* dos dados do sistema; RF_BKP_02: Permitir que administradores repovoe o banco de dados a partir de um *backup* previamente realizado.

Após a criação do protótipo com a ferramenta *Figma*, uma reunião de alinhamento foi realizada para validar as funcionalidades previstas para o sistema e também sua interface. Logo após, iniciou-se a codificação do aplicativo.

MODELAGENS

O diagrama de caso de uso deste trabalho, conforme exposto na figura 01 abaixo, demonstra que o sistema tem quatro papéis: Usuário, Gestor de produtos, Gestor comercial e Administrador.

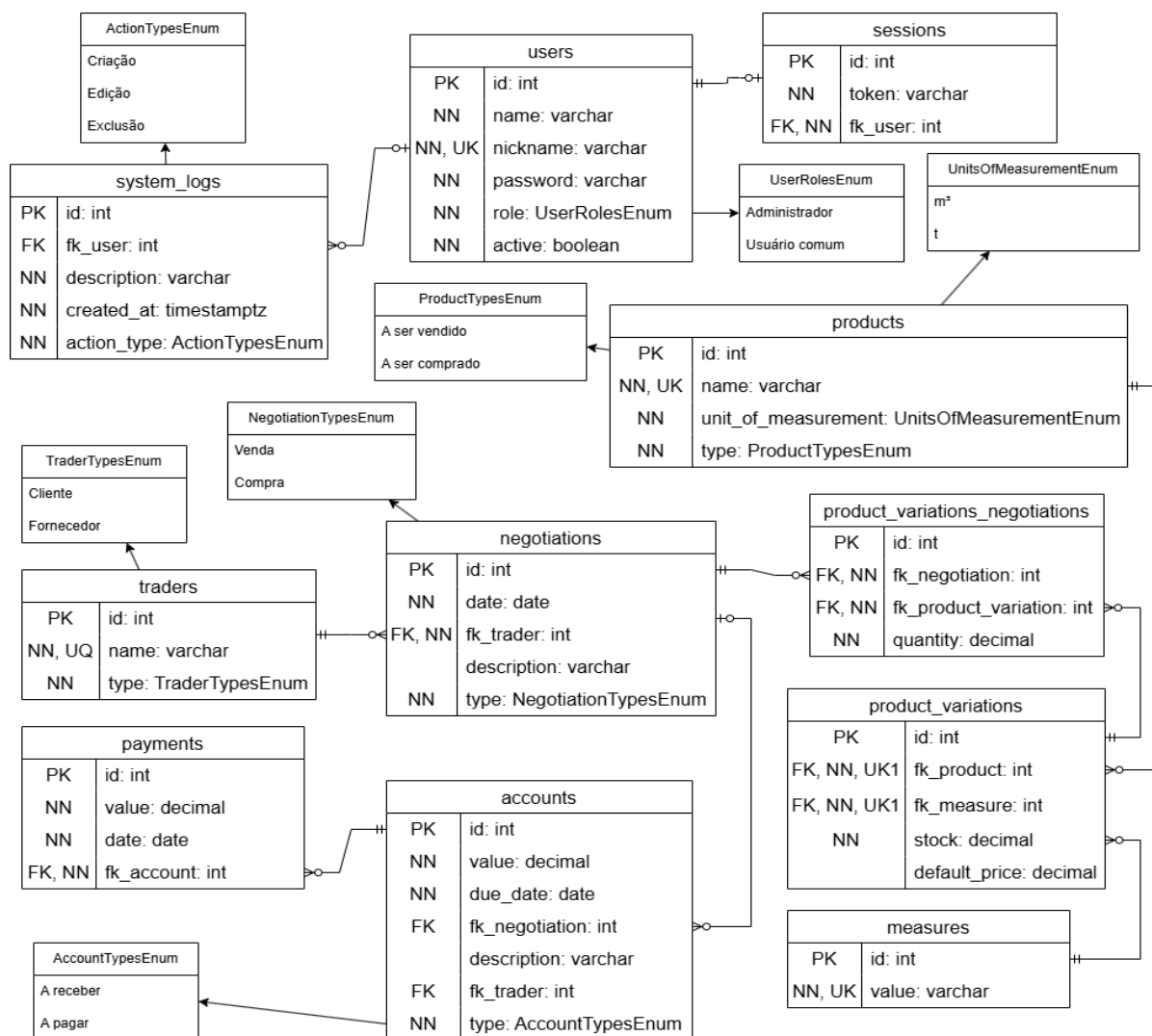
Figura 01 - diagrama de casos de uso



Fonte: os autores (2026).

Na figura 02 consta a estruturação do modelo, em torno do fluxo operacional do negócio, começando pelos produtos, passando pelas negociações e chegando às contas e pagamentos.

Figura 02 - diagrama de entidade e relacionamento

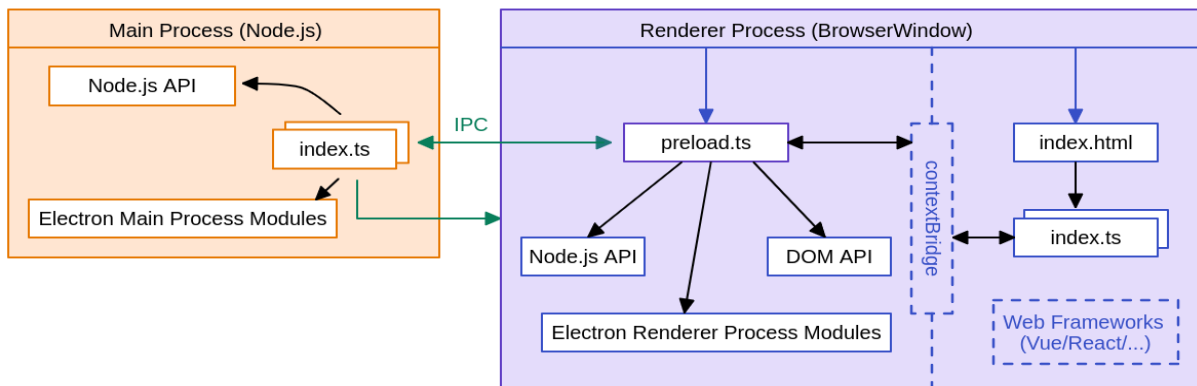


Fonte: os autores (2026).

O aplicativo propriamente dito funciona com base em dois subprocessos do Electron: o *Main Process* (Processo principal), que cuida da lógica do *backend* e da comunicação com funcionalidades nativas, sendo o local onde o *Node.js* será executado; e o *Renderer Process* (Processo de Renderização), responsável pela renderização do *frontend*.

Na figura 03 está a demonstração dos dois processos do *Electron* e sua comunicação via *IPC (Inter-Process Communication)*.

Figura 03 - Processos do electron

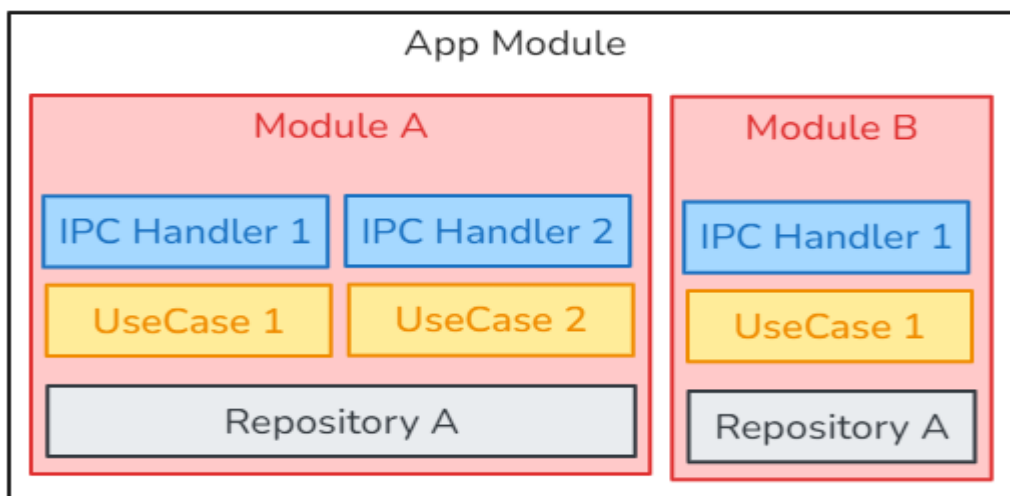


Fonte: electron-vite (2024).

O *backend* foi executado no modo *standalone* do NestJS, que permite usar o *framework* sem expor portas HTTP. Isso foi essencial para o projeto, já que a comunicação com o *frontend* acontece exclusivamente pelo protocolo IPC do *Electron*.

Com essa arquitetura, a camada de controladores não foi utilizada. No lugar dela, entram os *handlers* de IPC, que funcionam como pontos de entrada do *backend*. Cada *handler* recebe a mensagem enviada pelo *frontend* através do *contextBridge*, valida o *payload* e chama seu respectivo *use case*. Depois disso, o *handler* retorna a resposta ao *renderer*. Essa abordagem mantém a separação de responsabilidades do NestJS, mas adaptada ao fluxo interno da aplicação *desktop*, conforme descrito na figura 04.

Figura 04 - exemplificação da estrutura de camadas do backend



Fonte: os autores (2026).

O *frontend* do sistema foi desenvolvido com *React*, utilizando *Vite* como ferramenta de *build* e *Tailwind CSS* para estilização, juntamente com componentes do *HeroUI*. Essa combinação permitiu uma interface moderna, além de agilizar o desenvolvimento.

O *Vite* atua como SPA sendo CSR, gerando apenas arquivos estáticos. Isso se encaixa bem no contexto do *Electron*, onde toda a renderização ocorre no *renderer process*, responsável pela interface gráfica da aplicação *desktop*.

Como exemplos de interface gráfica, nas figuras 05 a 08, temos as telas de listagem de usuários, a listagem de negociações, listagem de contas e listagem de produtos. Nessas telas é possível acessar os modais de criação de usuários, negociações, contas e produtos.

A figura 05 demonstra a tela de usuários, que só pode ser acessada por administradores. Os usuários estão ordenados por nome e podem ser filtrados por nome, cargo ou estado de ativação (ativo ou desativado). A busca é paginada de maneira manual, através dos botões numerados. É possível editar as informações de um usuário, desativá-lo ou reativá-lo. Os usuários desativados são mostrados em vermelho. Nessa tela, é possível acessar o modal de criação de usuários e a página de registro de auditoria.

Figura 05 - listagem de usuários

ID	Nome completo	Nome de usuário	Cargo	Editar	Ativado / Desativado
77	Betina	betina123	Usuário comum		
76	Gabriel A	red	Usuário comum		
78	Fernanda	fernanda@1	Administrador		
79	Bárbara	baprigel	Usuário comum		
81	Diogo	diogo	Usuário comum		
82	João	joao	Usuário comum		

Fonte: os autores (2026).

Na figura 06 está demonstrada a tela de negociações, nas quais são representadas por *cards* agrupados e ordenados por data. A busca pode ser filtrada por tipo (compra ou venda) e por período. Cada negociação pode ser diferenciada entre compra ou venda a partir do ícone no canto superior direito do seu respectivo *card*. A busca é paginada de maneira automática

conforme o usuário rola a tela.

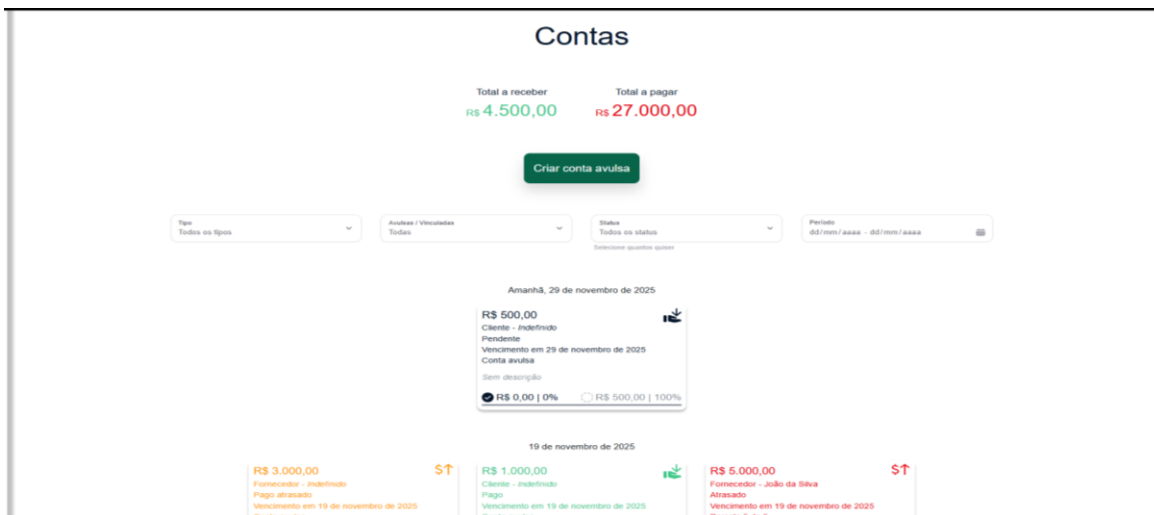
Figura 06 - listagem de negociações



Fonte: os autores (2026).

A figura 07 apresenta a página de contas, exibindo um valor total a receber e um valor total a pagar. As contas listadas são agrupadas e ordenadas por data e podem ser filtradas por tipo (a pagar ou a receber), avulsa ou vinculada (à uma negociação), status (paga, atrasada, paga com atraso e pendente) e período. A busca é paginada de maneira automática conforme o usuário rola a tela.

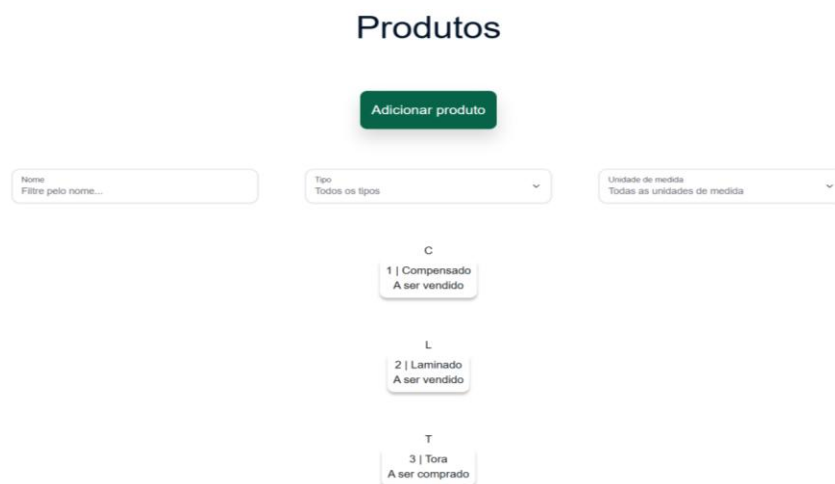
Figura 07 - listagem de contas



Fonte: os autores (2026).

A figura 08 demonstra a tela de produtos, onde é possível acessar o modal de criação de produtos. A listagem de produtos é agrupada pela primeira letra do nome do produto e ordenada alfabeticamente pelo mesmo. A busca pode ser filtrada pelo nome do produto, tipo (a vender ou a comprar) e pela unidade de medida (tonelada ou metro cúbico).

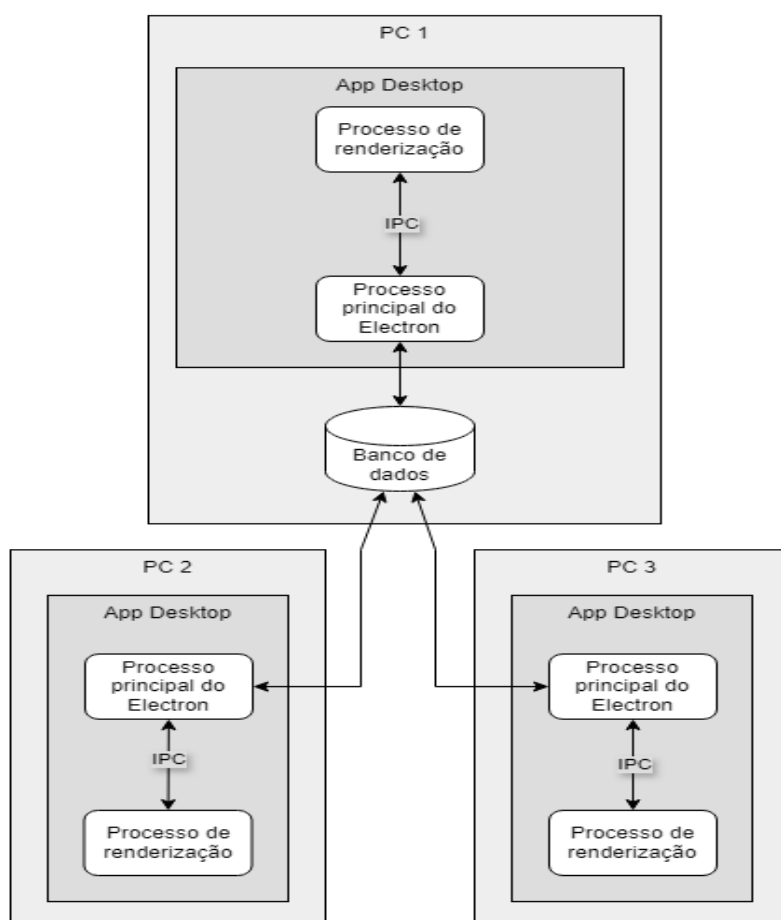
Figura 08 - listagem de produtos



Fonte: os autores (2026).

O aplicativo foi instalado nos computadores dos usuários utilizando o pacote de instalação gerado pelo *Electron* através do processo de *build*. Na figura 09 está a exemplificação da arquitetura de comunicação entre as instâncias e o banco de dados

Figura 09 - diagrama de arquitetura de hospedagem local



Fonte: os autores (2026).

Os requisitos foram considerados atendidos do ponto de vista técnico porque todas as funcionalidades levantadas no início do projeto foram implementadas, testadas manualmente e executadas corretamente dentro do fluxo real de uso da empresa. Ou seja, cada requisito funcional listado no documento foi checado individualmente e validado conforme sua regra de negócio, garantindo que o sistema faça exatamente o que foi definido.

Já na percepção dos usuários, foi aplicado um formulário. Como três usuários responderam ao questionário, cada módulo pode atingir no máximo 15 pontos (3 usuários x nota máxima 5) e no mínimo 3 pontos (3 usuários x nota mínima 1). Com base nisso, os módulos de usuários, produtos e *backup* atingiram 13/15, e os módulos de autenticação, negociantes, negociações e contas alcançaram a nota 15/15.

Mesmo com essas variações, todas as pontuações encontram-se acima de 86% de satisfação, o que reforça que o sistema, além de tecnicamente completo, apresenta boa aceitação entre seus usuários.

Em síntese, os achados revelam que o sistema cumpriu com sucesso tanto os requisitos técnicos originalmente propostos quanto às expectativas práticas dos usuários. A alta satisfação observada sugere que o processo de desenvolvimento foi eficaz em alinhar requisitos, implementação e experiência de uso. Esses resultados também indicam que eventuais ajustes futuros podem se concentrar em pontos específicos, como os módulos de usuários e de *backup*, a fim de otimizar ainda mais a interação e a percepção de valor por parte dos usuários.

CONCLUSÃO

O trabalho realizado teve como principal objetivo o desenvolvimento de um aplicativo de computador que auxiliasse na gestão de uma empresa madeireira do sul do Paraná, de forma totalmente personalizada às suas necessidades específicas. O desenvolvimento do *software* representou a aplicação prática dos conhecimentos adquiridos ao longo do curso de Análise e Desenvolvimento de Sistemas, abrangendo desde o levantamento de requisitos, planejamento, modelagem, desenvolvimento prático e implementação até a validação do sistema junto à empresa, além de oferecer uma solução tecnológica concreta para uma demanda real de mercado. A escolha por tecnologias modernas, como *TypeScript*, *NestJS*, *Vite* e *Electron*, proporcionou uma arquitetura robusta, modular e de fácil manutenção, capaz de ser expandida.

Os objetivos definidos para o projeto foram alcançados. Os requisitos levantados junto aos representantes da empresa foram atendidos, o sistema foi desenvolvido conforme o planejado, implementado no ambiente real e validado pelos usuários, que demonstraram

satisfação com o resultado final. Os resultados obtidos evidenciam que a plataforma atende de maneira satisfatória aos requisitos estabelecidos, confirmando que o *software* cumpre seu propósito como ferramenta digital e se apresenta como uma solução viável para a gestão da empresa, reforçando o impacto positivo da tecnologia na modernização e otimização de processos.

AGRADECIMENTOS

Ao IFPR como instituição apoiadora da pesquisa.

REFERÊNCIAS BIBLIOGRÁFICAS

ARMSTRONG, Ruth Selina. **Fatores de estresse no uso de sistemas de gestão ERP em indústrias brasileiras**. 2024. 121 f. Dissertação (Mestrado em Administração) – Escola de Administração, Universidade Federal Do Rio Grande Do Sul, Porto Alegre, 2024. Disponível em:

<https://lume.ufrgs.br/bitstream/handle/10183/281875/001217030.pdf?sequence=1&isAllowed=y>
Acesso em: 20 abr. 2026.

BALIEIRO, Anderson Fernandes; PINTO, Giuliano Scombatti. A importância do levantamento de requisitos no desenvolvimento de softwares. **Interface Tecnológica**, Taquaritinga, v. 21, n. 1, p. 115-126, 2024. Disponível em: <https://revista.fatectq.edu.br/interfacetecnologica/article/view/1845/1012>. Acesso em: 18 abr. 2026.

BAMBAZEK, P., GROHER, I.; SEYFF, N. Requirements engineering for sustainable software systems: a systematic mapping study. **Requirements Eng**, 28, 481-505, 2023. Disponível em: <https://doi.org/10.1007/s00766-023-00402-1>. Acesso em: 19 abr. 2026.

DAMO, Marco Antonio; SANTANDER, Victor Francisco Araya. **Investigando o Processo de Documentação de Requisitos em Sistemas Enterprise Resource Planning (ERP)**. In: WORKSHOP ON REQUIREMENTS ENGINEERING, 27., 2024, Rio de Janeiro. Anais... Rio de Janeiro: PUC-Rio, 2024. Disponível em: https://www.inf.puc-rio.br/~wer/WERpapers/artigos/artigos_WER24/WER2024_paper_18.pdf. Acesso em: 18 abr. 2026.

Electron-vite. **Development**: Using Preload Scripts. 2024. Disponível em: <https://electron-vite.org/guide/dev>. Acesso em: 20. abr. 2026.

MILANI, Isabela. **Diferença entre ERP Padrão e ERP Personalizado**: compare as opções e descubra qual melhor atende às suas necessidades empresariais. *Gestão.dev*, [S.l.], ago. 2024. Disponível em: <https://gestao.dev.br/blog/erp/diferenca-entre-erp-padrao-e-erp-personalizado>. Acesso em: 19 abr. 2026.

PIRES, Luiz. **ERP Industrial vs. ERP Genérico**: descubra qual é a melhor opção para sua empresa!. *Sensio Blog*, [S.l.], nov. 2023. Disponível em: <https://www.sensio.com.br/blog/erp-o-que-e>. Acesso em: 19 abr. 2026.

SANTOS, Elisandra Regina. **A relevância dos sistemas Enterprise Resource Planning (ERP) para a análise de negócios em uma empresa ou organização.** Tecnia: Revista de Educação, Ciência e Tecnologia do IFG, Goiânia, v. 9, n. 1, 2024. Disponível em: <https://periodicos.ifg.edu.br/tecnica/article/view/950/1770>. Acesso em: 20 abr. 2026.

SEBRAE. **Como a digitalização de processos impacta os resultados da empresa?** Sebrae Artigos, Pernambuco, ago. 2023. Disponível em: <https://sebrae.com.br/sites/PortalSebrae/ufs/pe/artigos/como-a-digitalizacao-de-processos-impacta-os-resultados-da-empresa,833be1541664a810VgnVCM1000001b00320aRCRD>. Acesso em: 14 abr. 2026.

SENIOR SISTEMAS. **O que é ERP e como funciona?** Blog Senior, Blumenau, 2025. Disponível em: <https://www.senior.com.br/sistema-erp-o-que-e-e-como-funciona>. Acesso em: 18 abr. 2026.