REVISTA TÓPICOS
ISSN 2965-6672 | Qualis A2

THUR: ARQUITETURA DE ASSISTENTE FINANCEIRO CONVERSACIONAL INTEGRADO POR APIS PARA MICRO E PEQUENAS EMPRESAS

THUR: A CONVERSATIONAL FINANCIAL ASSISTANT ARCHITECTURE INTEGRATED THROUGH APIS FOR MICRO AND SMALL ENTERPRISES

Ciências Exatas e da Terra • 25/09/2026

REGISTRO DOI: 10.70773/revistatopicos/790303755

Arthur Barros de Carvalho[1]

Wanderson Alexandre da Silva Quinto[2]

Armando José de Sá Santos[3]

Antonio Marcos Cardoso Silva[4]

Marilia Epaminondas Martins e Martins[5]

Luis Felipe de Melo Mendes[6]

RESUMO

A gestão financeira estruturada é determinante para a sobrevivência das micro e pequenas empresas, mas segue informal na maioria delas apesar da digitalização dos serviços financeiros. Este trabalho apresenta o Thur, protótipo de assistente financeiro conversacional cuja arquitetura integra um Modelo de Linguagem de Grande Escala a interfaces de programação de aplicações, com acesso previsto por aplicativos de mensagens. A arquitetura delega a consulta à fonte por chamadas de função, de modo que o valor apresentado ao usuário não é produzido pelo modelo, e substitui a navegação por menus pela formulação da pergunta em linguagem comum. Em cem perguntas submetidas três vezes cada, a acurácia de intenção foi de 98,00% e a de parametrização, de 90,91%, sem resposta indevida a pergunta fora do escopo e sem qualquer valor monetário gerado pelo modelo. A contribuição é uma arquitetura modular para consulta, em linguagem natural, dos dados de vendas e custos do próprio negócio.

Palavras-chave: assistente conversacional; modelos de linguagem de grande escala; chamada de função; interfaces de programação de aplicações; micro e pequenas empresas.

ABSTRACT

Structured financial management is decisive for the survival of micro and small enterprises, yet remains informal in most of them despite the digitalization of financial services. This work presents Thur, a prototype conversational financial assistant whose architecture integrates a Large Language Model with application programming interfaces, with access designed for messaging applications. The architecture delegates data retrieval to the source via function calls, so that the value presented to the user is not produced by the model, and replaces menu navigation with a

question asked in plain language. Across one hundred questions submitted three times each, intent accuracy reached 98.00% and parameterization accuracy 90.91%, with no improper answers to out-of-scope questions and no monetary value generated by the model. The contribution is a modular architecture for natural language querying of a business's own sales and cost data.
Keywords: conversational assistant; large language models; function calling; application programming interfaces; micro and small enterprises.

1. INTRODUÇÃO

A transformação digital tem redefinido processos no setor financeiro. Sperancini e Alexandre (2024) evidenciam que a expansão da tecnologia móvel, as mudanças comportamentais e as inovações regulatórias ampliaram a inclusão financeira, enquanto Silva, Garcia Junior e Araújo (2022) destacam que as fintechs impulsionam a modernização ao oferecerem serviços ágeis que reconfiguram a atuação das instituições tradicionais. Contudo, a oferta de infraestrutura digital não garante a democratização dessas ferramentas entre todos os atores econômicos.

Essa assimetria torna-se crítica no segmento das micro e pequenas empresas brasileiras. Conforme Fidelis e Moraes (2025), representam mais de 99% das empresas do país e respondem por cerca de 70% dos empregos formais. Os mesmos autores e Silva et al. (2009) ressaltam que a gestão financeira estruturada é vital para sua sobrevivência, e apontam que a maioria dos pequenos empreendedores ainda lida com registros imprecisos, controle insuficiente de caixa e baixa maturidade gerencial. Como reflexo, o

acompanhamento financeiro passa a ser percebido como processo burocrático e distante da realidade do negócio.

Diante desse quadro, a Inteligência Artificial (IA), sobretudo os Modelos de Linguagem de Grande Escala (LLMs), amplia a capacidade de processar linguagem natural. Segundo Vaswani et al. (2017), a arquitetura Transformer constitui sua base. Sobre ela, Lee et al. (2024) destacam modelos especializados em finanças (FinLLMs), e Kolt (2025), a evolução dos modelos generativos para agentes. Conforme Lewis et al. (2020) e Wei et al. (2022), a geração aumentada por recuperação (RAG) e a explicitação do raciocínio tornam mais confiável o que o modelo produz, o que não basta quando a resposta precisa reproduzir um valor exato.

Essa capacidade, contudo, exige integração às fontes de dados financeiros. Segundo Fielding (2000), as Interfaces de Programação de Aplicações (APIs) estabelecem uma interface uniforme, que faz sistemas distintos interoperarem sem conhecer a implementação alheia. Conforme o Banco Central do Brasil (2026), o Open Finance regulou esse compartilhamento, restrito a produtos bancários e de investimento e dependente de autorização no aplicativo detentor, escopo que não alcança a adquirência, atividade de processar pagamentos com cartão em nome do lojista, nem preserva a continuidade da conversa. Adota-se, por isso, a integração direta por API. Quanto à interface, McTear (2021) e Soares e Silva (2025) mostram a consolidação dos ambientes conversacionais, e Oliveira, Centeno e Diniz (2023), o WhatsApp pela penetração e pela API oficial. Definem-se, assim, as duas pontas que o trabalho liga: a fonte do dado e o canal do empreendedor.

Embora o avanço seja notável em cada frente, a convergência entre elas permanece parcial. Já existem assistentes conversacionais que consultam dados financeiros reais, mas o fazem pela via do Open Finance, que, pelas razões já expostas, deixa descoberta a adquirência, onde ficam as vendas processadas e a agenda de recebíveis. Revela-se, assim, uma lacuna quanto a uma arquitetura única que reúna um modelo de linguagem, as interfaces da instituição que processa as vendas do empreendedor e um aplicativo de mensagens, para responder em linguagem natural sobre o desempenho do negócio.

Diante dessa lacuna, esta pesquisa busca responder ao seguinte problema: de que forma uma arquitetura de assistente financeiro conversacional, integrada via APIs e acessível por aplicativos de mensagens, pode interpretar consultas em linguagem natural e devolver ao pequeno empreendedor dados exatos sobre o seu negócio?

Como resposta provisória a esse questionamento, parte-se da hipótese de que uma arquitetura baseada em chamadas de função e em separação por camadas é capaz de interpretar intenções financeiras expressas em linguagem natural e de devolver dados exatos provenientes da interface consultada, sem que o modelo generativo produza ou altere valores monetários. A conjectura é deliberadamente técnica e verificável: sustentá-la exige medir se a intenção foi corretamente identificada, se os parâmetros da consulta foram corretamente preenchidos e se nenhum valor apresentado ao usuário teve origem no modelo.

Esta pesquisa tem como objetivo projetar e avaliar a arquitetura de software de um protótipo de assistente financeiro conversacional,

aqui denominado Thur, concebido para integração via APIs financeiras e para acesso por aplicativos de mensagens, com foco na interpretação de consultas em linguagem natural sobre dados do próprio negócio. Propõe-se, para tanto, uma arquitetura modular que separa a interpretação da pergunta do acesso ao dado, de modo que o modelo de linguagem responda pela compreensão e pela verbalização, e a fonte consultada responda pelo valor. O aplicativo de mensagens é a interface-alvo do projeto, e não requisito de teste nesta etapa.

2. REVISÃO DA LITERATURA

2.1. Gestão Financeira em Micro e Pequenas Empresas

Neste trabalho, entende-se gestão financeira como o conjunto de decisões sobre o dinheiro que entra e sai do negócio, na acepção de Cheng e Mendes (1989), para quem essas decisões buscam equilibrar rentabilidade e liquidez, ou seja, gerar lucro sem deixar de pagar as contas na data certa. Os mesmos autores observam que essa responsabilidade se espalha por todas as áreas da empresa, e não cabe apenas ao setor financeiro.

Em uma micro ou pequena empresa (MPE), porém, essas áreas raramente existem separadas: quem vende costuma ser quem compra, quem paga e quem decide. A qualidade das decisões financeiras passa a depender, então, do tempo e do preparo de uma única pessoa, quase sempre o próprio dono.

Seria razoável esperar que essa pessoa tivesse instrumentos simples à mão, e ferramentas não faltam. Conforme Oliveira e Nascimento (2024), em revisão sistemática sobre gestão financeira em MPE, há demonstrações contábeis, controles de rotina e índices econômico-

financeiros, embora só uma minoria dos gestores os utilize. Silva et al. (2009) dão apoio empírico à constatação: mesmo entre pequenas empresas tidas como profissionalizadas, as vendas são anotadas de qualquer jeito e o dinheiro da empresa se mistura ao do dono. Falta, portanto, não a ferramenta, e sim o caminho que a leva da teoria contábil para a rotina de quem toca o negócio sozinho.

Sobre porque esse caminho não se completa, os autores divergem. Fidelis e Moraes (2025) apontam a falta de planejamento e a fragilidade no controle de caixa, e propõem educação financeira e ferramentas tecnológicas. Contra essa expectativa pesa Sant'anna et al. (2011), que já descreviam gestores cercados de dados brutos, com pouca informação útil para a decisão, e sugeriam exatamente software analítico e relatórios estruturados. Se o mesmo diagnóstico se repete há mais de quinze anos, treinar o gestor talvez não baste: é possível que o problema esteja também na forma como essas ferramentas pedem para ser usadas.

2.2. Digitalização dos Serviços Financeiros e os Limites da Inclusão

Enquanto esse diagnóstico permanecia o mesmo, o setor financeiro brasileiro passava por uma transformação que prometia aproximar os serviços de quem estava fora deles. Segundo Silva, Garcia Junior e Araújo (2022), as fintechs trouxeram menos burocracia e atendimento mais personalizado, pressionando os bancos tradicionais a acelerar seus canais digitais. Sperancini e Alexandre (2024) descrevem movimento semelhante, em que o celular aproximou um público antes não atendido e tornou o digital o principal meio de transação, mas acrescentam uma ressalva decisiva ao separarem a inclusão financeira em quatro dimensões, acesso,

uso, qualidade e destino, sendo a última a da inserção produtiva das empresas menores. A separação permite enxergar até onde a transformação chegou e onde parou.

É nas duas últimas dimensões que o pequeno empreendedor segue descoberto. Ferreira e Pepece (2023) lembram que ter conta em banco não significa estar financeiramente incluído, já que a conta aberta não garante efeito sobre o negócio. Na mesma direção, Darôs e Rosa (2023) observam que o acesso à tecnologia cresceu sem que crescesse a habilidade de aproveitá-la, e registram que trocar mensagens segue sendo o principal uso da internet no país. O empreendedor paga e recebe pelo celular sem dificuldade, mas continua sem enxergar como o negócio anda. Resolvido o acesso e não o uso, vale olhar para tecnologias que mudem a forma de conversar com a informação, e não apenas a de chegar até ela.

2.3. Modelos de Linguagem, Agentes e Uso de Ferramentas

Entre as tecnologias capazes de mudar essa forma de interação, segundo Silva et al. (2026), estão os Modelos de Linguagem de Grande Escala, ou Large Language Models (LLM), redes neurais treinadas para prever qual texto vem a seguir e que, a partir de certo tamanho, passam a resolver problemas que ninguém lhes ensinou explicitamente. Segundo Vaswani et al. (2017), esse tamanho só se tornou possível com a arquitetura Transformer, cujo mecanismo de atenção, por não depender da ordem da frase, permitiu dividir o treinamento entre muitas máquinas. Desse ganho de escala nasce a habilidade de entender uma pergunta escrita em linguagem comum, única razão pela qual a arquitetura é aqui mencionada.

Segundo Quinto (2025), compreender a pergunta resolve metade do problema, já que a resposta depende de informações que o modelo não guarda. No setor financeiro, Lee et al. (2024) observam que aplicações apoiadas em interfaces de programação seguem escassas. Entende-se aqui por agente, na acepção de Kolt (2025) e, o sistema que combina um modelo de linguagem a recursos de planejamento, memória e uso de ferramentas, esta última a categoria que aciona tais interfaces para buscar dados ausentes. Mialon et al. (2023) chegam à mesma divisão ao organizarem o campo em dois eixos, o raciocínio, que quebra a tarefa, e o uso de ferramentas, que chama programas externos. Fica clara a divisão de trabalho que orienta a arquitetura proposta: parte o modelo resolve sozinho, e parte precisa pedir a quem tem a informação.

Buscar o dado fora ainda não garante que a resposta esteja correta, garantia indispensável quando a informação vai orientar uma decisão de negócio. Segundo Lewis et al. (2020), a geração aumentada por recuperação, ou Retrieval-Augmented Generation (RAG), junta o que o modelo aprendeu no treinamento a uma base externa consultada na hora de responder, o que permite atualizar informações sem retreinar o sistema e reduzir a invenção de conteúdo. Wei et al. (2022) demonstram que pedir ao modelo os passos do raciocínio melhora seu desempenho em contas. Ambas as técnicas atuam sobre aquilo que o modelo gera por conta própria, buscando torná-lo mais confiável.

Existe, entretanto, linha de pesquisa que parte da premissa oposta e prefere não entregar ao modelo aquilo que ele faz mal. Schick et al. (2023) observam que os modelos falham em contas e em busca de fatos, tarefas em que programas simples se saem melhor, e propõem delegá-las por chamadas de função, entendidas neste

trabalho como comandos estruturados que acionam outro programa e devolvem um resultado que o modelo não precisa inventar. Morais (2025) traz evidência nessa direção: em experimento com base de dados restrita, usar o modelo direto pela interface de programação superou a recuperação aumentada e o ajuste fino, em qualidade e em custo. A escolha depende do que está em jogo: quando basta uma estimativa, melhorar o raciocínio resolve; quando o número precisa estar certo, é mais seguro ir buscá-lo na fonte.

2.4. Interfaces de Programação e Ambientes Conversacionais

Buscar o dado na fonte pressupõe um caminho técnico até ela, previsível o bastante para os dois lados se entenderem. Segundo Fielding (2000), as interfaces de programação de aplicações, ou Application Programming Interfaces (API), estabelecem uma interface uniforme, conjunto padronizado de operações que permite a dois programas trocarem dados sem conhecer o funcionamento interno alheio. Zachariadis e Ozcan (2017) observam que a abertura dessas interfaces cria oportunidade para modelos de plataforma no setor bancário. Cada parte evolui, assim, no seu ritmo, e trocar o fornecedor dos dados não obriga a refazer o sistema, característica retomada na arquitetura proposta.

Quando os dados são financeiros, essa troca não acontece livremente, e no Brasil tem regra própria. Conforme o Banco Central do Brasil (2026), o Open Finance abrange produtos bancários e de investimento e só libera o acesso depois que o usuário autoriza, dentro do aplicativo da instituição que guarda seus dados. Duas características o afastam do caso aqui estudado: o alcance, que não chega à adquirência, atividade de processar pagamentos com cartão em nome do lojista e onde ficam as vendas e a agenda de

recebíveis; e a autorização, que obriga o usuário a sair para outro aplicativo e interrompe a conversa. Por essas razões, este trabalho opta por falar diretamente com a interface que detém o dado.

Resolvido o acesso ao dado, resta decidir como apresentá-lo a quem não tem familiaridade com sistemas de gestão. McTear (2021) define sistemas de diálogo como programas que conversam com pessoas, separa os baseados em regras, que percorrem caminhos previstos pelo projetista, daqueles que interpretam linguagem natural, e observa que os hospedados em aplicativos de mensagens poupam o usuário de instalar um programa novo para cada serviço. Soares e Silva (2025) acrescentam uma ressalva: em revisão de cinquenta e cinco estudos, o ponto fraco do campo não é o modelo escolhido e sim a bagunça das bases que os alimentam, em geral documentos soltos e desatualizados. O achado interessa a esta pesquisa, que não depende de documentos avulsos, já que seus dados vêm de uma interface mantida por quem detém o registro.

Resta saber se o arranjo funciona fora do papel, e trabalhos recentes indicam que sim. Oliveira, Centeno e Diniz (2023) descrevem um assistente em aplicativo de mensagens construído pela interface oficial, e Pantoja et al. (2025) relatam resultado convergente, com assistente que reduziu dúvidas repetidas entre os usuários atendidos. Nos dois casos, porém, o assistente responde a partir de documentos da própria instituição, e não de dados financeiros guardados por terceiro. É essa ligação, entre uma interface de conversa e uma fonte financeira externa, que segue sem exemplo na literatura examinada.

2.5. Soluções Existentes no Mercado

Já existem, no mercado brasileiro, assistentes financeiros em aplicativos de mensagens, distinguíveis em três abordagens. A primeira é o atendimento por menu, que percorre caminhos predefinidos sem interpretar linguagem natural. A segunda interpreta a pergunta, mas opera sobre lançamentos que o próprio usuário informa por texto, áudio ou foto. A terceira consulta dados reais por Open Finance, com as limitações já apontadas. Nenhuma chega à adquirência, onde estão as vendas processadas e a agenda de recebíveis. Este trabalho investiga recorte distinto: interpreta a pergunta e a responde consultando a API de quem processa as vendas do negócio.

3. METODOLOGIA

3.1. Caracterização da Pesquisa

Esta pesquisa é de natureza aplicada, uma vez que se volta à construção de um artefato destinado a resolver um problema concreto, e adota abordagem qualitativa, já que interpreta o comportamento desse artefato diante de perguntas formuladas em linguagem comum, e não a distribuição estatística de um fenômeno. Quanto aos objetivos, caracteriza-se como exploratória e descritiva. Quanto aos procedimentos, o trabalho se organiza em duas frentes complementares, tratadas nas subseções seguintes.

3.2. Procedimento de Levantamento Bibliográfico

A fundamentação teórica resultou de revisão narrativa, e não sistemática, o que se declara para que os achados sejam lidos na medida certa. A busca foi conduzida no Google Acadêmico e complementada por pesquisa aberta na internet, a partir de seis descritores: gestão financeira, fintechs, agentes de IA, LLMs,

WhatsApp e o nome da instituição utilizada na validação. Os quatro primeiros orientaram o levantamento acadêmico da revisão, e os dois últimos, o reconhecimento de documentação técnica e de soluções de mercado, base da subseção que as examina. A seleção obedeceu à adequação temática aos quatro blocos da revisão, com fontes em português e em inglês, entre artigos de periódico e de evento, tese, monografia, livro, preprint e documentação institucional.

Quanto ao período, privilegiaram-se publicações de 2020 em diante, e as exceções obedecem a duas razões distintas. A primeira reúne obras de referência cujo conteúdo não é substituível por trabalho recente, caso de Cheng e Mendes (1989), Fielding (2000), Vaswani et al. (2017) e Peffers et al. (2007). A segunda preserva estudos empíricos anteriores porque sua data importa ao argumento: Silva et al. (2009) e Sant'anna et al. (2011) mostram que o mesmo diagnóstico sobre a gestão financeira das pequenas empresas se repete há mais de quinze anos. Em ambos os casos, a data foi avaliada pelo papel da fonte no argumento, e não em termos absolutos.

Adotou-se, ainda, o critério de leitura integral de cada fonte antes de sua citação, o que exclui a prática de reproduzir referências vistas apenas na bibliografia de terceiros. Sua aplicação teve efeito verificável sobre o corpus: duas referências foram substituídas por não se confirmar a existência delas, e uma afirmação atribuída a Silva et al. (2009) foi corrigida após conferência no artigo original.

3.3. Método de Desenvolvimento do Artefato

O desenvolvimento segue o Design Science Research, paradigma voltado à criação e à avaliação de artefatos que resolvem problemas identificados. Conforme Peffers et al. (2007), o método se organiza em seis atividades, identificação do problema, definição dos objetivos da solução, projeto e desenvolvimento, demonstração, avaliação e comunicação, e admite quatro pontos de entrada distintos. Este trabalho adota a entrada centrada no problema, uma vez que parte de uma dificuldade observada na rotina dos pequenos empreendedores e não de uma tecnologia à procura de aplicação. A escolha desse paradigma decorre da própria natureza do estudo, que não busca descrever um fenômeno existente, mas verificar se um arranjo técnico proposto se sustenta.

As seis atividades correspondem, neste trabalho, às seguintes etapas. A identificação do problema e a definição dos objetivos estão desenvolvidas nas seções anteriores, que estabelecem a lacuna e delimitam o que a solução precisa entregar. O projeto e desenvolvimento consistem na construção de uma arquitetura em quatro camadas, detalhada na seção seguinte. A demonstração consiste na operação do protótipo sobre um conjunto de perguntas representativas. A avaliação aplica as métricas descritas adiante. A comunicação se realiza neste artigo.

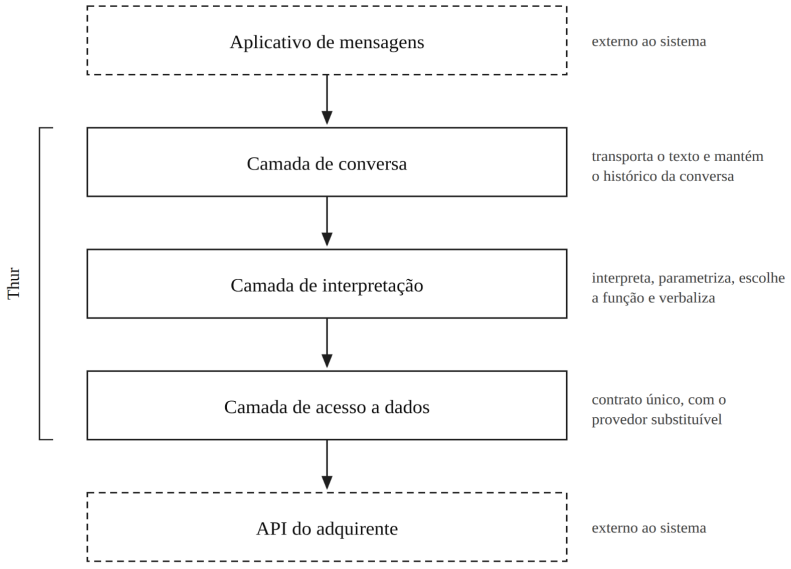
4. DESENVOLVIMENTO DO PROTÓTIPO

4.1. Arquitetura em Camadas

O artefato, aqui denominado Thur, foi organizado em quatro camadas (Figura 1), cada uma com responsabilidade única e dependente apenas da vizinha. A camada de conversa recebe a mensagem e devolve a resposta. A de interpretação identifica o que

foi perguntado e traduz a pergunta em chamada estruturada. A de acesso converte essa chamada em requisição à interface que detém o registro. A de dados é externa e pertence ao provedor. A separação atende a uma exigência do desenho: o modelo de linguagem pode ser trocado sem afetar o acesso aos dados, e o provedor pode ser trocado sem alterar a forma como o sistema conversa.

Figura 1. Arquitetura em camadas do Thur.

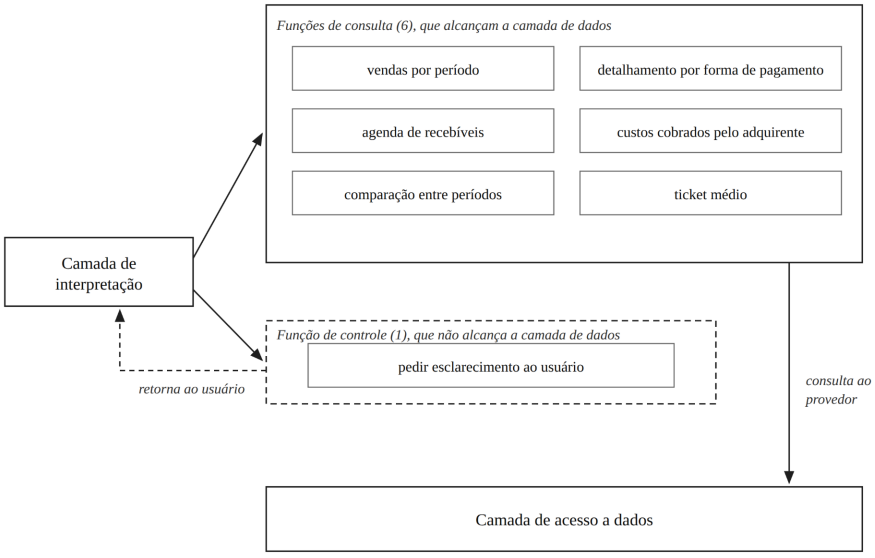


Fonte: elaborado pelo autor.

4.2. Camada de Interpretação e Funções

À camada de interpretação cabe descobrir o que o usuário quis saber e com quais parâmetros, delegando o restante por chamadas de função. Seis funções de consulta atendem ao recorte (Figura 2): vendas por período, agenda de recebíveis, comparação entre períodos, detalhamento por forma de pagamento, custos cobrados pelo adquirente e ticket médio, três delas com parâmetro opcional. As expressões temporais mais frequentes chegam ao modelo já convertidas em datas concretas, pela mesma razão que o impede de calcular valor, uma vez que aritmética de data também é conta; as demais continuam a seu cargo.

Figura 2. Funções declaradas à camada de interpretação.



Fonte: elaborado pelo autor.

Às seis soma-se uma função de controle, que solicita esclarecimento ao usuário. Ela não integra o contrato da camada de dados e foi declarada como função para que a decisão de perguntar se torne sinal explícito e mensurável. Seu acionamento é restrito: exige que a pergunta admita mais de uma leitura, que ao menos uma delas esteja no escopo e que as leituras conduzam a respostas distintas. Pergunta genérica sobre gasto ou custo, portanto, não a aciona, sendo respondida com o que o adquirente cobra e a declaração do que ficou de fora. A restrição responde a um risco conhecido, o de um sistema que pergunta sempre e assim nunca erra a função escolhida.

4.3. Camada de Acesso a Dados

A camada de acesso concentra todo o conhecimento sobre o provedor, dos endereços ao tratamento de erro, sem que nada disso alcance as demais. Conforme Fielding (2000), é a interface uniforme que permite a dois programas trocarem dados sem conhecer a implementação alheia, do que decorre sua evolução independente.

Aplicado aqui, o princípio faz com que trocar de provedor signifique reescrever um componente, e não o sistema, o que sustenta a afirmação, retomada na delimitação, de que o provedor utilizado é instância de teste da arquitetura e não seu objeto.

4.4. Origem do Número e Estado da Implementação

O princípio que organiza o conjunto é que o número vem da interface consultada, e não do modelo, que apenas o transporta para dentro de uma frase. Quatro mecanismos o sustentam no código: o modelo não conhece a interface, apenas um catálogo de funções; o valor monetário lhe chega já formatado em texto, para que apenas o transcreva; todo indicador derivado, como o ticket médio, é calculado na camada de acesso; e uma verificação automática confronta cada número da resposta final com os devolvidos pelas funções. Essa verificação origina uma das métricas descritas adiante.

Quanto ao estado da construção, a camada de interpretação está completa, assim como o contrato da camada de acesso, ocupado nesta etapa por um substituto programado; a camada de conversa opera em terminal e a integração ao aplicativo de mensagens não foi realizada. A camada expõe interface própria para o provedor de modelo, com adaptadores para três serviços comerciais, e a avaliação aqui relatada empregou o modelo gemini-3.5-flash-lite. A linguagem de implementação é Python.

4.5. Projeto da Interface Conversacional

Por ser conversacional, a interface do Thur não se organiza em torno de elementos dispostos numa tela. O problema de projeto é decidir o que o sistema informa, quando pergunta de volta e quando recusa. Segundo Nielsen (1994), heurísticas de usabilidade são um

conjunto reduzido de princípios amplos com que avaliadores inspecionam uma interface, formulados a partir da análise de problemas reais de uso. Este trabalho as emprega em função distinta da que lhes foi dada, como diretriz de projeto da arquitetura de interação e não como instrumento de avaliação de usabilidade. Quatro das nove heurísticas ali apresentadas orientaram as decisões a seguir.

A primeira consequência do formato diz respeito à navegabilidade, e é de supressão. Os dados que o Thur devolve já se encontram disponíveis no aplicativo do adquirente, alcançáveis por um percurso de menus que o empreendedor precisa conhecer. Ao substituir esse percurso por uma pergunta, a interface não simplifica a navegação, e sim a dispensa. O ganho pretendido não está no acesso ao dado, que já existia, mas na eliminação do caminho até ele.

Duas heurísticas orientaram a forma das respostas. A visibilidade do estado do sistema levou à decisão de que toda resposta declare o período efetivamente consultado, como em "nos últimos 30 dias, de 21/07/2026 a 19/08/2026, você vendeu R$ 35.596,70". A escolha tem efeito verificável: quando o sistema resolve mal uma expressão temporal, o equívoco fica visível a quem lê, ao passo que uma resposta que informasse apenas o valor o ocultaria. A correspondência com o mundo real, por sua vez, orientou o vocabulário, que emprega os termos do próprio comerciante, como maquininha e taxa, e admite referências temporais coloquiais em lugar de datas formais.

Outras duas orientaram o comportamento diante do que o sistema não sabe. A prevenção de erros sustenta a recusa explícita de

perguntas fora do escopo, preferida à tentativa de resposta aproximada, e sustenta também o impedimento de que o modelo produza qualquer número por conta própria. O reconhecimento em lugar da memorização sustenta a função de esclarecimento: diante de pergunta ambígua, o sistema oferece as leituras possíveis, como em "você quer saber sobre o aluguel das maquininhas ou o aluguel do imóvel?", em vez de exigir que o usuário conheça de antemão o termo que o sistema espera. A mesma orientação aparece na recusa que indica a parte atendível do pedido, apontando o que pode ser consultado quando o todo não pode. Registre-se que essas decisões não foram submetidas a avaliadores independentes nem verificadas junto a usuários, procedimentos que permanecem como trabalho futuro.

4.6. Procedimento de Avaliação

A avaliação é exclusivamente técnica e não envolve usuários reais. Os testes correm contra um substituto que implementa o mesmo contrato da interface real e devolve dados fictícios determinísticos, de modo que a mesma pergunta produza sempre o mesmo número e o que varia seja apenas o comportamento sob exame. Fixou-se como referência o dia 19 de agosto de 2026, uma quarta-feira, escolha deliberada porque expressões como "essa semana" e "semana passada" degeneram quando a data de referência cai numa segunda. As rodadas foram conduzidas entre agosto e setembro de 2026, com essa data mantida constante em todas elas.

O conjunto de teste reúne cem perguntas, construídas antes da execução e organizadas por função, com várias formulações da mesma intenção, variando vocabulário, ordem e referência temporal. Vinte delas estão deliberadamente fora do escopo, sem as quais a

respectiva taxa não teria denominador, e três admitem mais de uma leitura, destinadas à função de esclarecimento. Dezesseis aceitam mais de um gabarito, por comportarem leitura dupla defensável. Para cada item registra-se de antemão a resposta esperada, isto é, a função que deveria ser acionada e seus argumentos, o que permite comparação automática entre o esperado e o obtido.

Cada pergunta é submetida três vezes, porque modelos de linguagem podem produzir saídas distintas para uma mesma entrada. As métricas incidem sobre o total de execuções, e não sobre a decisão majoritária de cada pergunta. A escolha é consequente: apurar por maioria esconderia a instabilidade justamente na métrica que o leitor examina, e uma pergunta que acerta em duas de três execuções entrega resposta errada a um terço de quem a fizer.

Sete métricas orientam a avaliação, definidas operacionalmente a seguir.

Métrica	Definição
Acurácia de intenção	Execuções em que a função acionada foi a esperada, computadas como acerto a recusa correta e o esclarecimento correto, sobre o total de execuções.
Acurácia de parametrização	Entre os acertos de intenção em perguntas de consulta, execuções com todos os argumentos iguais ao gabarito ou a alternativa aceita.
Taxa de resposta fora de escopo	Entre as execuções de perguntas fora de escopo, quantas acionaram função de consulta em vez de recusar.
Taxa de esclarecimento indevido	Entre as execuções de perguntas não ambíguas, quantas geraram pedido de esclarecimento.

Estabilidade	Proporção de perguntas que produziram a mesma decisão em todas as repetições.
Violações de origem do número	Ocorrências de número na resposta final sem correspondência nos resultados devolvidos pelas funções.
Latência	Média, mediana e percentil noventa e cinco do tempo entre pergunta e resposta, excluída a espera por cota do provedor.

As duas últimas merecem justificativa. A taxa de esclarecimento indevido existe porque pedir esclarecimento é a forma mais simples de burlar a avaliação. As violações de origem do número verificam empiricamente a garantia que a arquitetura promete, por meio de verificação heurística, que compara valores sem rastrear a proveniência de cada símbolo, e é por existirem que a exatidão do valor não figura entre as métricas: como o número vem da fonte e é apenas verbalizado, sua correção está garantida por construção, e o que cabe medir é se essa construção se sustentou.

4.7. Delimitação do Estudo

Delimita-se o escopo à movimentação que o adquirente registra sobre o negócio, o que abrange as vendas processadas, a agenda de recebíveis, os custos por ele cobrados e os indicadores deles derivados. O critério do recorte é a origem do dado, e não uma categoria contábil. Ficam de fora as despesas com terceiros e as tarifas de contas mantidas em outras instituições, que o adquirente não registra e cuja obtenção exigiria declaração do usuário ou compartilhamento por Open Finance, caminhos contra os quais este trabalho argumenta. Nesta versão, o assistente apenas apresenta informações. A emissão de recomendações, o controle de despesas

e a projeção de fluxo de caixa permanecem como possibilidades de evolução, retomadas ao final deste trabalho.

Registre-se, por fim, que a instituição financeira adotada constitui instância de validação da arquitetura proposta, e não seu objeto: a camada de acesso a dados é concebida como componente substituível, de modo que a solução permanece aplicável a outros provedores de serviços financeiros. Declara-se, ainda, que o autor mantém vínculo profissional com a instituição utilizada na validação, o que facilitou o acesso ao ambiente de homologação e não interfere nos critérios de avaliação, definidos previamente e aplicáveis a qualquer provedor que exponha os mesmos dados.

5. RESULTADOS E DISCUSSÃO

5.1. Visão Geral das Rodadas

O protótipo foi submetido a quatro rodadas de avaliação, todas com o modelo gemini-3.5-flash-lite. As rodadas não são repetições de uma mesma medida, e sim etapas de um ciclo de correção, o que exige declarar o que mudou entre elas antes de compará-las.

Métrica	R1	R2	R3	R4
Perguntas	98	100	100	100
Repetições por pergunta	1	1	1	3
Execuções válidas	97	100	100	100
Acurácia de intenção	96,91%	98,00%	98,00%	98,00%

Acurácia de parametrização	86,30%	76,62%	92,21%	90,91%
Taxa de resposta fora de escopo	5,26%	0,00%	0,00%	0,00%
Taxa de esclarecimento indevido	0,00%	0,00%	0,00%	0,00%
Estabilidade	Não medida	Não medida	Não medida	96,00%
Violações de origem do número	18	2	1	4

As três primeiras rodadas empregaram uma execução por pergunta e a quarta, três, o que perfaz 97, 100, 100 e 300 execuções válidas; a taxa de esclarecimento indevido foi nula nas quatro. Adota-se a rodada 4 como referência, por ser a única com estabilidade efetivamente medida. Com execução única, a estabilidade aparece como total nos relatórios, resultado que é artefato do procedimento e não medida do comportamento, razão pela qual não se reporta nas três primeiras. A latência mediana caiu de 6,17 s na primeira rodada para cerca de 2 s nas três seguintes, e o percentil noventa e cinco, de 106,28 s para 22,96 s.

Três ressalvas qualificam a leitura da tabela. Entre a primeira e a segunda rodada o conjunto passou de 98 para 100 perguntas, quatro gabaritos foram corrigidos e o texto de instrução mudou em cinco pontos, de modo que a comparação entre elas mistura efeitos. Entre a segunda e a terceira mudou apenas a redação da instrução de ano, e entre a terceira e a quarta apenas o número de repetições, o que torna esses dois pares mais informativos. A quarta rodada foi

executada em dois dias, por esgotamento da cota gratuita do provedor, mantidos o mesmo modelo, o mesmo código e o mesmo conjunto.

O desempenho por categoria de pergunta, na rodada de referência, mostra diferenças que a leitura agregada esconde. A escolha da função foi correta em todas as categorias de consulta e em todas as perguntas fora de escopo, o que indica que a identificação do que está sendo perguntado, e do que não pode ser respondido, é a parte estável do comportamento. A dificuldade concentra-se na parametrização, e dentro dela na comparação entre períodos, cuja acurácia de parametrização foi de 58,3%, única categoria abaixo de 87%. As três perguntas ambíguas não sustentam conclusão e são apresentadas como indício.

5.2. Verificação Empírica da Exatidão por Construção

A arquitetura promete que o valor apresentado ao usuário provém da interface consultada e não do modelo. A promessa foi verificada, e não apenas afirmada, pela auditoria de origem do número descrita na metodologia.

Na primeira rodada, a auditoria localizou uma resposta em que o modelo executou uma soma por conta própria:

Somando crédito à vista (R$ 11.999,68) e parcelado (R$ 7.886,50), o total foi de R$ 19.886,18.

A operação está aritmeticamente correta, o que não atenua o problema: o número exibido não veio da fonte. A correção foi feita na camada de acesso, que passou a devolver o total do crédito já agregado, e um teste automatizado passou a falhar caso o agregado

seja removido. Nas trezentas execuções da rodada de referência, nenhum valor monetário foi produzido pelo modelo, e as quatro violações registradas correspondem a descrição de período, não a valor.

O episódio dialoga diretamente com Schick et al. (2023), para quem modelos de linguagem apresentam desempenho fraco em aritmética, razão pela qual tais operações devem ser delegadas a programas externos. O que aqui se acrescenta é de outra ordem: não a constatação de que o modelo erra a conta, mas a de que ele a executa mesmo quando não lhe foi pedido, e a de que apenas um mecanismo de verificação explícito torna essa iniciativa visível.

5.3. A Natureza dos Erros de Parametrização

Em todas as execuções em que o modelo escolheu período incorreto, o valor devolvido estava correto para o período que ele efetivamente solicitou. O exemplo a seguir, da rodada de referência, ilustra o padrão:

Comparando o mês passado (01/07/2025 a 31/07/2025) com este mês (01/08/2026 a 19/08/2026), o faturamento passou de R$ 29.881,98 para R$ 22.554,67.

O ano está errado e os valores estão certos para o intervalo consultado. O erro permaneceu confinado à camada em que a arquitetura admite que ele ocorra, sem contaminar o dado, o que corresponde na prática à divisão que Mialon et al. (2023) propõem entre raciocínio e uso de ferramentas. Convém notar ainda que a resposta declara o período utilizado, o que torna o equívoco perceptível a quem lê; uma resposta que informasse apenas a variação do faturamento o tornaria invisível.

A distribuição desses erros é igualmente informativa. Das vinte e uma execuções com falha de parametrização, dezoito envolvem a mesma expressão, "mês passado", resolvida com o ano incorreto. A acurácia de 90,91% subestima, portanto, o desempenho no restante do conjunto, e indica que o ganho disponível está concentrado em um único ponto do tratamento temporal.

5.4. Sensibilidade da Camada de Interpretação à Redação do Prompt

Entre a primeira e a segunda rodada foi acrescentada ao texto de instrução a seguinte orientação: "Atenção ao ano: salvo quando o usuário disser explicitamente que quer o ano passado, todo mês citado pelo nome é do ano corrente." A acurácia de parametrização caiu de 86,30% para 76,62%, e os erros de ano subiram de três para quinze.

A instrução está logicamente correta. Ela é, porém, construída sobre uma exceção e menciona a expressão "ano passado", e o comportamento observado sugere que o modelo reteve o termo saliente em lugar da regra. Reescrita em forma positiva e concreta, com o ano corrente interpolado, a métrica subiu para 92,21% na terceira rodada. Não se trata de experimento controlado, uma vez que outras alterações ocorreram entre a primeira e a segunda rodada, mas o padrão se mantém consistente ao longo das três e o mecanismo é plausível.

O achado matiza o otimismo com a engenharia de prompt como via de confiabilidade, discutida a partir de Wei et al. (2022) na fundamentação. Uma instrução correta produziu perda de dez pontos percentuais, o que sugere que a formulação da instrução é

ela própria uma variável de projeto, sujeita a verificação empírica, e não um ajuste menor de redação.

5.5. Instabilidade Entre Execuções

Quatro das cinco perguntas instáveis da rodada de referência são comparações que empregam a expressão "mês passado", e o modelo não erra sempre da mesma forma. Na pergunta "meu ticket médio caiu esse mês?", as três repetições produziram três períodos distintos, um deles correto. Em "quanto eu vendi mês passado?", uma repetição empregou julho de 2025 e outra julho de 2027.

Uma avaliação de execução única exibiria essas perguntas como corretas ou incorretas conforme a execução sorteada, e nenhuma das duas apresentações descreveria o comportamento. O resultado é justificativa empírica para a decisão de repetir cada pergunta e apurar sobre execuções, e não sobre a decisão majoritária.

Registre-se que o valor de 96,00% emprega critério de identidade ajustado após a observação dos dados. Uma das cinco perguntas instáveis divergia apenas porque uma repetição grafou "imóvel" e outra "imovel" nas leituras oferecidas pelo pedido de esclarecimento. O critério passou a desconsiderar o texto livre da função de controle, preservando os argumentos na identidade das funções de consulta. Pelo critério original, a estabilidade é de 95,00%. Ambos os valores são reportados por transparência.

Cabe uma aproximação com a literatura examinada. Soares e Silva (2025) identificam a estruturação das bases informacionais como principal fragilidade dos sistemas conversacionais. No arranjo aqui avaliado a base é determinística e íntegra, e a fragilidade observada é de outra natureza, situada na variação entre execuções de um

mesmo modelo diante de uma mesma entrada, dimensão que a revisão realizada não encontrou discutida.

5.6. Limites do Instrumento de Avaliação

Uma das repetições da pergunta "quanto paguei de taxa esse mês?" acionou a função correta, recebeu o valor correto e produziu a seguinte resposta:

Nos últimos 30 dias (de 2026-07-21 a 2026-08-19), você pagou R$ 570,26 de taxa sobre as vendas, o que representa uma taxa efetiva de 2,53%.

O valor está correto e o período declarado é outro. Para a acurácia de intenção e para a de parametrização, a execução conta como acerto, pois a chamada de função estava correta. Apenas a auditoria de origem do número identificou a divergência, porque os elementos "30", "07" e "21" não correspondiam a campo algum do resultado devolvido.

O caso expõe um limite das métricas previstas: elas avaliam a chamada de função, e não a fidelidade da verbalização àquilo que foi efetivamente consultado. Trata-se de lacuna do instrumento, e não do artefato, e sua identificação decorre de uma métrica que não constava do plano original, o que sugere que auditorias de natureza distinta da métrica principal observam o que esta não alcança.

5.7. Exemplos de Interação

As respostas a seguir foram produzidas nas rodadas três e quatro, com o mesmo modelo e o mesmo ambiente.

O caso de referência, que corresponde ao uso previsto:

Quanto vendi nos últimos 30 dias? Nos últimos 30 dias (de 2026-07-21 a 2026-08-19), você vendeu R$ 35.596,70.

Delimitação por origem do dado, nos dois sentidos:

Quanto gastei esse mês? *Nos custos cobrados por nós neste mês, de 2026-08-01 a 2026-08-19, o total foi de R$ 1.140,26. Esse valor inclui R$ 570,26 de taxas variáveis e R$ 570,00 de custos fixos. Custos com fornecedores, folha ou outras despesas com terceiros ficam de fora.*

E as despesas com meus fornecedores, quanto deu? *Despesas com fornecedores ficam de fora do que consigo ver, pois registro apenas os dados da sua maquininha, como vendas e taxas cobradas por ela.*

Acionamento da função de controle diante de ambiguidade legítima:

Quanto paguei de aluguel? Você quer saber sobre o aluguel das maquininhas ou o aluguel do imóvel?

5.8. Limitações do Estudo

Por ser técnica e correr contra um substituto programado, a avaliação nada permite afirmar sobre usabilidade ou utilidade percebida, e não verifica falha de rede, limite de requisições nem divergência de formato na integração real.

Avaliou-se um único modelo, escolhido por disponibilidade de cota gratuita, e trata-se de modelo de pequeno porte, de modo que a

comparação entre modelos prevista na metodologia permanece pendente. Pela mesma razão, a substituibilidade do provedor de dados e a do modelo de linguagem constituem propriedades de projeto, verificáveis na estrutura do código, mas não exercitadas empiricamente, uma vez que a avaliação empregou um único substituto programado e um único modelo. As formulações do conjunto de teste foram redigidas pelo autor a partir do uso esperado, e não levantadas junto a empreendedores. A avaliação considera um único turno por pergunta, de forma que o pedido de esclarecimento é medido no instante em que ocorre, sem que se verifique o acerto posterior à resposta do usuário. Por fim, conforme exposto na subseção 5.6, a fidelidade da verbalização não é medida por nenhuma das métricas de acurácia.

6. CONSIDERAÇÕES FINAIS

Este trabalho teve por objetivo projetar e avaliar a arquitetura de software de um protótipo de assistente financeiro conversacional, o Thur, concebido para integração via interfaces de programação e acesso por aplicativos de mensagens, com foco na interpretação de consultas em linguagem natural sobre dados do próprio negócio. A pergunta que o orientou indagava de que forma uma arquitetura assim poderia interpretar tais consultas e devolver ao pequeno empreendedor dados exatos.

A resposta obtida é que isso se torna possível quando a interpretação da pergunta e o acesso ao dado são separados em camadas distintas e a busca é delegada à fonte por chamadas de função, cabendo ao modelo de linguagem compreender a intenção e verbalizar o resultado, jamais produzi-lo. Em cem perguntas submetidas três vezes cada, a identificação da intenção foi correta

em 98,00% das execuções e o preenchimento dos parâmetros em 90,91%, sem que qualquer pergunta fora do escopo recebesse resposta indevida. Mais importante para a tese defendida, nenhum valor monetário apresentado ao usuário foi produzido pelo modelo, resultado que sustenta empiricamente a afirmação de que a exatidão do número é garantida pelo desenho do sistema e não pelo desempenho do modelo.

A contribuição é dupla. No plano do artefato, o trabalho formaliza uma arquitetura em que a camada de acesso concentra o conhecimento sobre o provedor, o que permite substituí-lo sem refazer o restante do sistema, e em que a origem de cada número apresentado é auditada de forma automática. No plano do método, dois achados extrapolam o caso examinado: a repetição de cada pergunta revelou instabilidade que a execução única esconderia, com uma mesma consulta produzindo três períodos distintos em três tentativas; e a auditoria de origem identificou uma falha de verbalização que as métricas de acurácia classificaram como acerto, o que sugere que instrumentos de natureza distinta da métrica principal são úteis precisamente por observarem o que ela não alcança.

Os limites do estudo decorrem do recorte adotado e foram expostos na subseção 5.8: avaliação técnica contra um substituto programado, sem participação de usuários, com um único modelo de linguagem, sem a integração ao aplicativo de mensagens prevista como interface-alvo e com cada pergunta avaliada em turno único.

Desses limites decorrem os caminhos futuros mais imediatos. Cabe realizar a integração da camada de comunicação a um aplicativo de

mensagens por meio de interface oficial, fechando a ponta que o projeto prevê, e submeter o artefato a avaliação com usuários reais, aplicando instrumento consagrado de usabilidade a um pequeno grupo de empreendedores, o que permitiria examinar a redução de barreira técnica que motivou a pesquisa e que este trabalho deliberadamente não mediu. Cabe ainda comparar modelos de linguagem sob o mesmo conjunto de teste, avaliar o comportamento diante da interface real e estender a avaliação ao diálogo de mais de um turno. Retomam-se, por fim, as dimensões deixadas fora do escopo: o controle de despesas e a projeção de fluxo de caixa dependem de dados que o adquirente não registra, e a emissão de recomendações desloca o sistema da função de informar para a de aconselhar, o que exige tratamento próprio; as três permanecem como evolução prevista do artefato.

REFERÊNCIAS BIBLIOGRÁFICAS

BANCO CENTRAL DO BRASIL. **Open Finance:** quero entender mais. Brasília, DF: Banco Central do Brasil, 2026. Disponível em: https://www.bcb.gov.br/estabilidadefinanceira/entender-open-finance. Acesso em: 14 ago. 2026.

CHENG, A.; MENDES, M. M. A importância e a responsabilidade da gestão financeira na empresa. In: CONFERÊNCIA INTERAMERICANA DE CONTABILIDADE, 18., 1989, Paraguai. **Anais** [...]. [S. l.: s. n.], 1989.

DARÔS, A. B.; ROSA, J. R. Novas tecnologias e educação financeira. **Interface Tecnológica,** v. 20, n. 1, p. 250-262, 2023.

FERREIRA, Í. M.; PEPECE, A. **Fintechs e inclusão financeira.** Assis: Fatec Assis, 2023.

FIDELIS, M. S.; MORAES, A. S. Gestão financeira em micro e pequenas empresas: desafios e soluções. **Educação Sem Distância:** Revista Eletrônica da Faculdade Unyleya, Rio de Janeiro, v. 5, n. 1, jun. 2025.

FIELDING, R. T. **Architectural styles and the design of network-based software architectures.** 2000. Tese (Doutorado em Information and Computer Science) – University of California, Irvine, 2000.

KOLT, N. Governing AI agents. **Notre Dame Law Review,** v. 101, 2025. No prelo. Disponível em: https://arxiv.org/abs/2501.07913. Acesso em: 12 jul. 2026.

LEE, J. *et al.* **A survey of large language models in finance (FinLLMs).** 2024. Disponível em: https://arxiv.org/abs/2402.02315. Acesso em: 12 jul. 2026.

LEWIS, P. *et al.* Retrieval-augmented generation for knowledge-intensive NLP tasks. In: CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 34., 2020, [s. l.]. **Proceedings** [...]. [S. l.: s. n.], 2020.

McTEAR, M. **Conversational AI:** dialogue systems, conversational agents, and chatbots. [S. l.]: Morgan & Claypool, 2021. (Synthesis Lectures on Human Language Technologies).

MIALON, G. *et al.* **Augmented language models:** a survey. 2023. Disponível em: https://arxiv.org/abs/2302.07842. Acesso em: 12 jul. 2026.

MORAIS, A. E. R. **Chatbots inteligentes para automatização do atendimento ao cliente:** explorando diferentes métodos de implementação. 2025. Monografia (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2025.

NIELSEN, J. Enhancing the explanatory power of usability heuristics. In: CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 1994, Boston. **Proceedings** [...]. New York: ACM, 1994. p. 152-158.

OLIVEIRA, G. F. A.; CENTENO, R. R.; DINIZ, D. P. Desenvolvimento de chatbot em WhatsApp para empresa júnior. **Revista Eletrônica de Computação Aplicada,** Franca, v. 4, n. 2, p. 115-133, 2023.

OLIVEIRA, S. M. N.; NASCIMENTO, A. J. S. Ferramentas de gestão financeira: uma revisão bibliográfica acerca do uso em micro e pequenas empresas. **Cadernos de Gestão e Empreendedorismo,** Rio de Janeiro, v. 12, n. 1, p. 13-33, jan./abr. 2024.

PANTOJA, A. S. *et al.* Desenvolvimento de um chatbot inteligente para auxílio de atividades organizacionais durante o período de processos seletivos da UFPA. In: WORKSHOP SOBRE ASPECTOS SOCIAIS, HUMANOS E ECONÔMICOS DE SOFTWARE, 10., 2025, Maceió. **Anais** [...]. Porto Alegre: SBC, 2025. p. 143-154.

PEFFERS, K. *et al.* A design science research methodology for information systems research. **Journal of Management Information Systems,** v. 24, n. 3, p. 45-77, 2007.

QUINTO, Wanderson Alexandre da Silva; CONTE, Thiago Nicolau Magalhães de Souza; SANTOSd, Armando José de Sá; SOUZA, ALAN MARCEL FERNANDES DE; DI PAOLO, Ítalo Flexa; PALHETA, Brenda

dos Santos; OLIVEIRA, Maria Gabriela da Silva; LOUREIRO, Maisa de Sousa; REIS, Gustavo Wendell Santiago dos. Explorando o impacto da Inteligência Artificial na formação do pensamento crítico entre acadêmicos de T.I. na Região Norte do Brasil. CADERNO PEDAGÓGICO (LAJEADO. ONLINE), v. 22, p. 1-21, 2025.

SANT'ANNA, P. R. *et al.* Tecnologia da informação como ferramenta para a análise econômica e financeira em apoio à tomada de decisão para as micro e pequenas empresas. **Revista de Administração Pública,** Rio de Janeiro, v. 45, n. 5, p. 1589-1611, 2011.

SCHICK, T. *et al.* Toolformer: language models can teach themselves to use tools. In: CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 37., 2023, [s. l.]. **Proceedings** [...]. [S. l.: s. n.], 2023.

SILVA, Antônio; PIMENTEL, Andrey; RIECHI, Tatiana; FOGAÇA, Thaís; FERREIRA, Marta. Chatbot Design: A Systematic Literature Review. In: **INTERNATIONAL CONFERENCE ON ENTERPRISE INFORMATION SYSTEMS (ICEIS),** 28., 2026. Proceedings of the 28th International Conference on Enterprise Information Systems – Volume 2. [S. l.]: SciTePress, 2026. p. 1776–1787. DOI: 10.5220/0014834500004018.

SILVA, F. C. *et al.* Práticas de gestão financeira em pequenas empresas: uma caracterização quanto ao nível de profissionalização. In: CONGRESSO BRASILEIRO DE CUSTOS, 16., 2009, Fortaleza. **Anais** [...]. Fortaleza: [s. n.], 2009.

SILVA, V. B. S.; GARCIA JUNIOR, W. R. R.; ARAÚJO, C. V. P. Fintechs: (r)evolução bancária na era da economia digital. **Revista da PGBC,** Brasília, DF, v. 16, n. 1, p. 65-78, 2022.

SOARES, J. R.; SILVA, P. N. Concepção de chatbots: perspectivas e interseções na Ciência da Informação. **RDBCI:** Revista Digital de Biblioteconomia e Ciência da Informação, Campinas, v. 23, e025031, 2025.

SPERANCINI, J. H. B. S.; ALEXANDRE, M. V. C. Inovações tecnológicas e inclusão no setor financeiro brasileiro. **Informe Econômico,** Teresina, v. 48, n. 1, p. 116-143, jan./jun. 2024.

VASWANI, A. *et al.* Attention is all you need. In: CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 31., 2017, Long Beach. **Proceedings** [...]. [S. l.: s. n.], 2017.

WEI, J. *et al.* Chain-of-thought prompting elicits reasoning in large language models. In: CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 36., 2022, New Orleans. **Proceedings** [...]. [S. l.: s. n.], 2022.

ZACHARIADIS, M.; OZCAN, P. **The API economy and digital transformation in financial services:** the case of open banking. Londres: SWIFT Institute, 2017. (SWIFT Institute Working Paper, n. 2016-001). Disponível em: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2975199. Acesso em: 12 jul. 2026.

---

[1] Discente do Curso Superior de Engenharia de Software da Universade do Estado do Pará *Campus* VI. E-mail: acesse o artigo original para visualizar o e-mail

[2] Docente do Bacharelado em Engenharia de Software da Universidade do Estado do Pará – Uepa, *Campus* Redenção. E-mail: acesse o artigo original para visualizar o e-mail

[3] Docente do Bacharelado em Engenharia de Software da Universidade do Estado do Pará – Uepa, *Campus* Ananindeua. E-mail: acesse o artigo original para visualizar o e-mail

[4] Docente do Bacharelado em Engenharia de Software da Universidade do Estado do Pará – Uepa, Campus Castanhal. E-mail: acesse o artigo original para visualizar o e-mail

[5] Docente do curso técnico em informática integrado ao ensino médio do Instituto Federal do Pará - IFPA, Campus Marabá Industrial. E-mail: acesse o artigo original para visualizar o e-mail

[6] Discente do Curso Superior de Engenharia de Software da Universade do Estado do Pará Campus VI. E-mail: acesse o artigo original para visualizar o e-mail