

**Instruções para a preparação de artigos para Trabalhos de Graduação do
Curso de ADS da Faculdade de Tecnologia de São José do Rio Preto
(Modelo ADS 2025-2)**

**TÍTULO DO TRABALHO DE GRADUAÇÃO: SUBTÍTULO SE
HOVER**

A. B. Autor1, C. Co-autor2, D. E. F. Co-autor3 (identifique: orientador ou co-orientador
quando for o caso) *, G. H. I. Co-autor4 ***

e-mail:

autor@provedor.com.br; autor2@provedor.com.br; autor3@provedor.com.br;
autor4@provedor.com.br

Resumo: A transformação digital tem acelerado a necessidade de automação inteligente em setores que lidam com grande volume documental e regras regulatórias dinâmicas. Este trabalho apresenta o desenvolvimento de uma plataforma SaaS baseada em arquitetura de microsserviços, projetada como um ecossistema para agentes de Inteligência Artificial verticalizados. A proposta foca em permitir a integração modular de novos agentes com mínimo retrabalho de engenharia e inclui, como prova de conceito, a implementação de um agente contábil voltado para Microempreendedores Individuais (MEI). O agente combina processamento de documentos via OCR e extração de dados por LLMs com um assistente conversacional suportado por RAG para consultas sobre declaração do MEI. São descritos o desenho arquitetural, a modelagem dos serviços, o fluxo de orquestração de agentes, a interface do usuário e um modelo de faturamento por consumo de tokens. Resultados experimentais e testes de usabilidade demonstram a viabilidade técnica e comercial da plataforma, além de sua capacidade de escalabilidade e extensão para outras verticais de negócio.

Palavras-chave: Inteligência Artificial. Microsserviços. SaaS. OCR. RAG. Automação Contábil.

Abstract: Digital transformation has intensified the demand for intelligent automation in domains that manage large document volumes and evolving regulatory frameworks. This paper describes the development of a SaaS platform built on a microservices architecture, designed as an ecosystem for verticalized Artificial Intelligence agents. The platform enables modular integration of new agents with minimal engineering effort and includes, as a proof of concept, an accounting agent targeted at Individual Microentrepreneurs (MEI). The agent combines document processing via OCR and structured data extraction using LLMs with a conversational assistant powered by Retrieval-Augmented Generation (RAG) for MEI declaration queries. We present the architectural design, service modeling, agent orchestration flow, user interface, and a token-based billing model. Experimental results and usability tests validate the platform's technical feasibility, commercial potential, and scalability to other business verticals.

Keywords: Artificial Intelligence. Microservices. SaaS. OCR. RAG. Accounting Automation.

1 Introdução

Vivemos uma era marcada pela transformação digital e pela convergência entre automação, inteligência artificial (IA) e computação em nuvem. Empresas de todos os setores têm buscado soluções que permitam não apenas otimizar processos, mas também ampliar sua capacidade de tomada de decisão com base em dados. Dentro desse contexto, surge a necessidade de plataformas tecnológicas que consigam integrar múltiplos sistemas, processar grandes volumes de informação e adaptar-se rapidamente a novos cenários de negócio. No entanto, a maioria das soluções atuais ainda é limitada por arquiteturas monolíticas, pouco escaláveis e de difícil manutenção, o que inviabiliza a rápida incorporação de novas funcionalidades e agentes inteligentes.

A crescente demanda por automação inteligente, especialmente em segmentos que lidam com alto volume documental, como o contábil, evidencia uma lacuna significativa entre as ferramentas disponíveis e as necessidades reais do mercado. Microempreendedores Individuais (MEIs) e escritórios de contabilidade, por exemplo, enfrentam desafios diários relacionados à digitalização de notas fiscais, à extração de dados e à adaptação a normas tributárias em constante mudança. Nesse cenário, a Inteligência Artificial — e, em particular, os Grandes Modelos de Linguagem (LLMs) — desponta como uma solução promissora para interpretar documentos, extrair informações relevantes e interagir com o usuário de forma natural e contextualizada.

Diante desse panorama, o presente trabalho tem como proposta o desenvolvimento de uma plataforma inteligente de automação baseada em arquitetura de microsserviços, concebida para servir como um ecossistema escalável e flexível de agentes de IA verticalizados. Essa arquitetura tem como diferencial a modularidade, permitindo a adição de novos agentes especializados sem necessidade de grandes reestruturações técnicas. Como prova de conceito, foi implementado um agente contábil voltado ao público MEI, com duas funcionalidades principais: o processamento automático de documentos por meio de Reconhecimento Óptico de Caracteres (OCR) e a extração de dados estruturados por LLMs, além de um assistente conversacional baseado em Geração Aumentada por Recuperação (RAG), capaz de responder dúvidas sobre declarações e obrigações fiscais.

A metodologia adotada baseia-se em práticas ágeis de desenvolvimento de software, com o uso de tecnologias modernas como FastAPI, React, PostgreSQL, MongoDB e Docker,

integradas em um ambiente orquestrado e modular. A pesquisa busca não apenas demonstrar a viabilidade técnica da arquitetura proposta, mas também validar seu potencial de expansão para outras áreas, como jurídica, financeira e de recursos humanos. Assim, este trabalho contribui para o avanço do conhecimento na área de engenharia de software aplicada à Inteligência Artificial, ao apresentar uma proposta sólida e adaptável para o desenvolvimento de sistemas inteligentes escaláveis.

Em síntese, esta introdução apresenta o contexto, a relevância e os objetivos que orientam o estudo, situando o leitor diante do problema central — a necessidade de uma arquitetura escalável e modular para agentes de IA — e antecipando a metodologia e os resultados esperados que serão detalhados ao longo do artigo.

2 Justificativa

A escolha deste tema se justifica pela crescente necessidade de soluções tecnológicas que combinem automação inteligente, escalabilidade e modularidade em um único ecossistema. Em um cenário onde as empresas enfrentam uma sobrecarga de dados e uma exigência cada vez maior por eficiência operacional, torna-se essencial desenvolver plataformas que consigam integrar diferentes agentes de Inteligência Artificial (IA) de maneira estruturada e expansível.

No contexto atual, muitas organizações — especialmente micro e pequenas empresas — ainda dependem de processos manuais e softwares isolados, o que gera lentidão, erros humanos e altos custos operacionais. A área contábil, em particular, é um exemplo notório desse desafio: tarefas como o lançamento de notas fiscais, cálculo de tributos e interpretação de normativas fiscais são repetitivas e consomem tempo que poderia ser direcionado à análise e à tomada de decisão estratégica. Assim, uma plataforma que automatize essas atividades e ofereça suporte inteligente, como a proposta neste trabalho, tem potencial de democratizar o acesso à automação avançada e aumentar significativamente a produtividade do setor.

Do ponto de vista acadêmico e tecnológico, o desenvolvimento de uma plataforma baseada em microsserviços para agentes de IA verticalizados representa uma contribuição relevante para a área de Engenharia de Software aplicada à Inteligência Artificial. Diferentemente de sistemas monolíticos tradicionais, a abordagem modular e desacoplada proposta permite a evolução contínua da plataforma, possibilitando a integração de novos agentes especializados em diferentes domínios — como jurídico, financeiro, logístico ou de recursos humanos — sem necessidade de reescrever a base do sistema.

Além disso, este trabalho contribui para o avanço da pesquisa em arquiteturas SaaS (Software as a Service) e em modelos de negócio baseados em consumo, aplicando um sistema de faturamento por tokens que garante escalabilidade e sustentabilidade comercial. A justificativa também se estende à importância educacional e científica da proposta, que integra conceitos de microsserviços, IA generativa, OCR, RAG e autenticação JWT em um único projeto, fornecendo um exemplo prático e didático de aplicação integrada de múltiplas tecnologias emergentes.

Portanto, o desenvolvimento desta plataforma não se limita a atender uma demanda pontual, mas visa criar uma base tecnológica sólida e replicável, capaz de evoluir conforme novas necessidades de automação surjam em diferentes setores, reafirmando sua relevância prática e teórica dentro do contexto atual da transformação digital.

3 Objetivo(s)

Objetivo Geral

Desenvolver uma **plataforma inteligente de automação baseada em arquitetura de microsserviços**, projetada para hospedar e orquestrar múltiplos agentes de Inteligência Artificial (IA) especializados em diferentes áreas de negócio, validando sua aplicabilidade através da implementação de um **agente contábil voltado para Microempreendedores Individuais (MEI)**.

Objetivos Específicos

- **Projetar e implementar** uma arquitetura de software escalável e modular, com microsserviços independentes responsáveis por autenticação, orquestração de agentes, processamento de documentos e faturamento baseado em tokens;
- **Desenvolver** um agente contábil com capacidade de **processar documentos via Reconhecimento Óptico de Caracteres (OCR)** e **extrair dados estruturados utilizando Grandes Modelos de Linguagem (LLMs)**;
- **Implementar** a técnica de **Geração Aumentada por Recuperação (RAG)** para permitir que o agente responda a dúvidas contábeis de forma contextualizada e baseada em documentação atualizada;

- **Construir** uma **interface web responsiva** e intuitiva, desenvolvida em **React e Tailwind CSS**, permitindo a interação entre usuário e agentes de IA;
- **Integrar** tecnologias modernas como **FastAPI, PostgreSQL, MongoDB, Redis, Celery e Docker**, garantindo alta disponibilidade, desempenho e facilidade de manutenção;
- **Avaliar** o desempenho da arquitetura proposta quanto à escalabilidade, eficiência de resposta e facilidade de integração de novos agentes;
- **Validar** o modelo de negócio proposto, baseado em **consumo de tokens**, que permite a monetização do uso da plataforma de forma flexível e proporcional ao consumo de recursos.

Com esses objetivos, o trabalho busca não apenas entregar uma solução funcional, mas também estabelecer uma **base tecnológica replicável e extensível**, capaz de servir de referência para o desenvolvimento de futuras plataformas de automação inteligente em diferentes domínios.

4 Fundamentação Teórica

A fundamentação teórica deste trabalho baseia-se em princípios e conceitos provenientes das áreas de Engenharia de Software, Computação em Nuvem e Inteligência Artificial (IA), com foco na integração entre arquiteturas de microsserviços, modelos de linguagem de larga escala (LLMs) e tecnologias de automação inteligente. A seguir, são apresentados os fundamentos conceituais que embasam o desenvolvimento da plataforma proposta.

Arquitetura de Microsserviços

A arquitetura de microsserviços é uma abordagem moderna de desenvolvimento de software que consiste em decompor uma aplicação em pequenos serviços independentes, cada um responsável por uma funcionalidade específica do sistema. Segundo Newman (2015), essa abordagem permite maior escalabilidade, manutenibilidade e flexibilidade, uma vez que cada serviço pode ser desenvolvido, testado e implantado de forma autônoma. Em contraste com sistemas monolíticos, os microsserviços facilitam a atualização contínua e a integração de novas funcionalidades sem comprometer o funcionamento do restante da aplicação. No contexto deste projeto, essa arquitetura foi essencial para permitir a criação de uma

plataforma extensível de agentes de IA, em que cada agente atua como um serviço especializado, orquestrado por um núcleo central.

Padrão API Gateway

Em sistemas baseados em microsserviços, o API Gateway atua como um ponto de entrada único entre os clientes e os serviços internos. Ele é responsável pelo roteamento de requisições, autenticação, balanceamento de carga, controle de acesso e agregação de respostas. De acordo com Richardson (2018), o uso desse padrão melhora o desempenho e a segurança, simplificando a comunicação entre os componentes da aplicação. Na plataforma desenvolvida, o NGINX foi adotado como API Gateway, garantindo um fluxo centralizado e eficiente de comunicação entre o frontend e os serviços internos.

Inteligência Artificial e Grandes Modelos de Linguagem (LLMs)

Os Grandes Modelos de Linguagem (Large Language Models), como o GPT e o Gemini, representam um avanço significativo no campo da Inteligência Artificial. Esses modelos são capazes de compreender, gerar e transformar textos de forma contextualizada, após serem treinados com enormes volumes de dados linguísticos. Conforme Vaswani et al. (2017), a arquitetura Transformer, utilizada nesses modelos, revolucionou o processamento de linguagem natural (NLP), possibilitando resultados mais precisos em tarefas como tradução, sumarização e resposta a perguntas. Neste trabalho, os LLMs são aplicados tanto para extração de informações estruturadas a partir de documentos contábeis quanto para a criação de assistentes conversacionais inteligentes.

Geração Aumentada por Recuperação (RAG)

A técnica de Retrieval-Augmented Generation (RAG) combina a capacidade de geração de texto dos LLMs com mecanismos de busca em bases de conhecimento externas. Segundo Lewis et al. (2020), o modelo RAG recupera informações relevantes de documentos, bases de dados ou arquivos e as fornece ao LLM como contexto para a geração de respostas mais precisas e atualizadas. Essa abordagem foi adotada neste projeto para aprimorar a qualidade das respostas do agente contábil, permitindo que ele baseie suas interações em informações reais, como legislações fiscais e instruções oficiais da Receita Federal.

Reconhecimento Óptico de Caracteres (OCR)

O OCR (Optical Character Recognition) é uma tecnologia que converte imagens de documentos, PDFs e outros arquivos digitalizados em texto editável e pesquisável. Segundo Smith (2007), o OCR é um componente fundamental em sistemas de automação documental, pois elimina a necessidade de digitação manual e reduz o erro humano. Na plataforma proposta, o OCR é responsável por processar notas fiscais e documentos contábeis, extraindo

dados como CNPJ, valor, data e descrição, que são posteriormente interpretados por modelos de IA.

Autenticação e Segurança com JSON Web Tokens (JWT)

A autenticação baseada em JSON Web Tokens (JWT) é amplamente utilizada em sistemas distribuídos por sua eficiência e segurança. Conforme Jones et al. (2015), o JWT é um padrão aberto (RFC 7519) que permite a autenticação sem estado (stateless), armazenando informações de sessão em um token criptografado. No projeto em questão, o JWT garante que as interações entre usuários e agentes ocorram de forma segura e autenticada, sem a necessidade de manter sessões no servidor.

Arquitetura SaaS (Software as a Service)

O modelo Software como Serviço (SaaS) é uma abordagem de distribuição de software em que as aplicações são hospedadas em servidores na nuvem e acessadas pelos usuários por meio da internet. Segundo Armbrust et al. (2010), o SaaS oferece vantagens como redução de custos com infraestrutura, atualização contínua e escalabilidade dinâmica. A plataforma proposta segue esse modelo, permitindo o uso sob demanda e a cobrança por consumo de recursos via sistema de tokens, o que a torna viável tanto técnica quanto comercialmente.

Síntese Teórica

A integração dessas tecnologias — microsserviços, API Gateway, LLMs, RAG, OCR, JWT e SaaS — sustenta a construção de um sistema robusto, escalável e inteligente. A fundamentação teórica aqui apresentada fornece a base conceitual que orientou as decisões de arquitetura, desenvolvimento e validação da plataforma, assegurando que cada componente tecnológico contribua para o objetivo central de criar um ecossistema flexível de automação baseado em IA.

5 Trabalhos Similares

A seguir são apresentados cinco trabalhos correlatos que embasam a proposta desta plataforma, abrangendo pesquisas sobre **RAG**, **OCR**, **LLMs**, **microsserviços** e **SaaS** aplicados à **automação contábil**.

Lewis et al. (2020) – *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*

Lewis et al. propõem o método RAG, que combina um componente de recuperação de informações com um gerador pré-treinado, melhorando significativamente a precisão e

contextualização das respostas de modelos de linguagem (LLMs). O estudo demonstra ganhos expressivos em tarefas de resposta a perguntas e fundamenta o uso da técnica em sistemas inteligentes. No contexto deste projeto, o RAG é essencial para o funcionamento do agente contábil, permitindo que ele consulte documentos fiscais e produza respostas precisas, atualizadas e baseadas em fatos.

Resenha: O trabalho de Lewis et al. destaca-se por unir duas abordagens complementares — busca e geração — criando uma estrutura robusta para agentes inteligentes. A proposta inspira diretamente o modelo de conversação do projeto, que usa RAG como base para o assistente contábil.

Katti et al. (2018) – *Chargrid: Towards Understanding 2D Documents*

Os autores introduzem o *Chargrid*, uma técnica que permite a compreensão bidimensional de documentos estruturados, como notas fiscais e faturas. Essa representação possibilita que modelos de IA entendam a posição espacial dos textos, melhorando a extração de dados via OCR.

Resenha: Esse trabalho reforça a importância da estrutura visual dos documentos para o desempenho do OCR. No projeto proposto, esses conceitos são aplicados na criação do módulo de digitalização e extração de dados contábeis, garantindo maior acurácia no reconhecimento de campos relevantes.

Krieger et al. (2023) – *Automated Invoice Processing with AI and Machine Learning*

O estudo descreve pipelines automatizados de processamento de faturas utilizando OCR e aprendizado de máquina, demonstrando ganhos de produtividade e redução de erros. Os autores também abordam a integração com sistemas contábeis e o papel do feedback humano no refinamento dos modelos.

Resenha: O artigo de Krieger et al. valida a viabilidade técnica da automação contábil via IA. A arquitetura de pipelines inspirou a estrutura do *Document Service* da plataforma desenvolvida, demonstrando como combinar OCR e IA em fluxos de processamento eficientes.

Dragoni et al. (2017) – *Microservices: Yesterday, Today and Tomorrow*

Dragoni e colaboradores analisam a evolução da arquitetura de microsserviços, destacando benefícios como modularidade e escalabilidade, bem como desafios de observabilidade e comunicação entre serviços.

Resenha: Este estudo oferece a base arquitetural que norteou a criação da plataforma proposta. A escolha por microsserviços deriva da necessidade de isolar funções, facilitar a manutenção e permitir a integração de novos agentes de IA com mínimo retrabalho.

Leite (2022) – *SaaS Process: Um Processo de Desenvolvimento para Software como Serviço*

A dissertação apresenta um processo de desenvolvimento voltado a soluções SaaS, combinando práticas ágeis com integração e entrega contínuas (CI/CD).

Resenha: O trabalho de Leite contribui para o entendimento das boas práticas de desenvolvimento de plataformas SaaS, consolidando a visão de que automação, escalabilidade e modularidade são elementos fundamentais em sistemas distribuídos, como o proposto neste artigo.

Síntese Geral:

Os trabalhos analisados convergem em demonstrar o papel central das tecnologias emergentes de IA e arquitetura moderna na construção de soluções inteligentes e escaláveis. A combinação de OCR, RAG, LLMs e microserviços oferece o alicerce necessário para sistemas SaaS voltados à automação contábil, confirmando a relevância teórica e prática do projeto proposto.

6 Metodologia

A metodologia adotada neste trabalho fundamenta-se em princípios da engenharia de software moderna, combinando abordagens ágeis de desenvolvimento, arquitetura baseada em microserviços e integração de tecnologias de Inteligência Artificial (IA) para automação de processos contábeis. O objetivo é demonstrar, de forma prática, a viabilidade técnica e operacional de uma plataforma SaaS (Software as a Service) capaz de integrar múltiplos agentes inteligentes em um ecossistema escalável e modular.

1. Abordagem de Desenvolvimento

O projeto será conduzido com base na metodologia **SCRUM**, que organiza o processo em **sprints** curtas e interativas, permitindo entregas incrementais de funcionalidades e contínua validação junto ao orientador e à comunidade acadêmica. Cada sprint contemplará atividades de planejamento, execução e revisão, com ênfase na priorização de requisitos essenciais da plataforma, como autenticação, orquestração de agentes e processamento de documentos.

2. Ferramentas e Tecnologias Utilizadas

Para atingir os objetivos definidos, foram selecionadas ferramentas e tecnologias modernas, que garantem robustez, escalabilidade e interoperabilidade entre os módulos do sistema:

- **Frontend:**

O desenvolvimento da interface será realizado com o framework **React** (utilizando Vite como empacotador e Tailwind CSS para estilização). Essa escolha proporciona desempenho otimizado e flexibilidade na construção de interfaces dinâmicas, especialmente na interação com os agentes de IA.

- **Backend (Microsserviços):**

A camada de backend será desenvolvida em **Python**, utilizando o framework **FastAPI** devido à sua performance, tipagem estática e validação nativa de dados. Cada serviço (auth, billing, documentos e orquestração de IA) será containerizado e gerenciado de forma independente.

- **Banco de Dados:**

Serão empregados dois bancos de dados complementares: **PostgreSQL**, responsável pelo armazenamento relacional de usuários, tokens e logs de atividades, e **MongoDB**, utilizado para guardar metadados e documentos estruturados de forma flexível.

Além disso, será integrada a extensão **pgvector**, que permitirá armazenar embeddings semânticos de documentos para uso nas consultas RAG.

- **Armazenamento e Processamento de Arquivos:**

Os documentos enviados pelos usuários serão armazenados em um serviço compatível com **Amazon S3**, utilizando o **MinIO** como alternativa local e open-source. Para o processamento assíncrono de OCR e consultas LLM, serão utilizados **Celery** e **Redis**.

- **Inteligência Artificial:**

O módulo de IA fará uso de **Grandes Modelos de Linguagem (LLMs)** e da técnica de **Geração Aumentada por Recuperação (RAG)**, integrando pipelines de busca e geração de respostas com base em documentos fiscais e normas contábeis.

A extração de informações de notas fiscais e comprovantes será feita com **OCR (Optical Character Recognition)** utilizando **Tesseract OCR** e bibliotecas de visão computacional (como OpenCV e Pytesseract).

- **Arquitetura e Comunicação:**

A comunicação entre serviços ocorrerá por meio de APIs REST e filas de mensagens, mediadas por um **API Gateway NGINX**. Essa camada centraliza autenticação, roteamento e controle de acesso, garantindo segurança e escalabilidade.

3. Etapas da Pesquisa e Desenvolvimento

O projeto será dividido em quatro fases principais:

1. **Análise e Especificação:** Levantamento de requisitos, definição de casos de uso e modelagem conceitual da arquitetura.
2. **Desenvolvimento e Integração:** Implementação dos microsserviços, configuração dos pipelines de OCR e RAG, e integração do frontend com o backend.
3. **Testes e Validação:** Execução de testes unitários, de integração e de usabilidade, com foco na precisão do OCR e na qualidade das respostas do agente contábil.
4. **Implantação e Avaliação:** Deploy em ambiente simulado de nuvem (utilizando Docker Compose e MinIO) e análise dos resultados obtidos quanto à escalabilidade e eficiência do sistema.

4. Justificativa Metodológica

A escolha dessa metodologia visa assegurar **reprodutibilidade, modularidade e validade técnica** do experimento. O uso combinado de microsserviços, OCR, RAG e LLMs permite testar, em pequena escala, um modelo realista de plataforma SaaS inteligente. Dessa forma, a pesquisa não apenas comprova a viabilidade da automação contábil com IA, mas também fornece um **modelo arquitetural escalável**, que pode ser expandido para outras verticais, como jurídico, fiscal e recursos humanos.

7 Desenvolvimento

O desenvolvimento do projeto foi estruturado em etapas, iniciando-se pela construção do **frontend** da aplicação e avançando, posteriormente, para a modelagem e implementação do **backend** e dos **microsserviços** que compõem a arquitetura modular da plataforma. O objetivo central foi criar um sistema web funcional, intuitivo e escalável para automação contábil baseado em agentes de Inteligência Artificial (IA).

1. Desenvolvimento do Frontend

A primeira etapa consistiu na criação da **interface do usuário (UI)** utilizando o framework **React** com **Vite**, que oferece um ambiente de desenvolvimento rápido e modular. A interface foi projetada com base em princípios de **UI/UX Design**, priorizando clareza, responsividade e simplicidade de navegação.

O **Tailwind CSS** foi adotado como ferramenta de estilização, permitindo a criação de componentes reutilizáveis com design moderno e consistente.

As telas desenvolvidas até o momento incluem:

- **Tela de Login e Autenticação:** onde o usuário insere suas credenciais ou acessa uma conta demo de teste.
- **Dashboard:** exibe uma visão geral da plataforma, com histórico de atividades, agentes ativos e atalhos rápidos para envio de documentos ou acesso ao agente contábil.
- **Gerenciamento de Documentos:** interface que permite o upload de arquivos (PDF, JPG, PNG, XLSX) e o acompanhamento do status de processamento (em análise, concluído ou erro).
- **Chat do Agente Contábil:** ambiente de interação em linguagem natural com o assistente especializado em MEI, apresentando perguntas sugeridas e histórico de conversação.

Essas interfaces demonstram o fluxo principal de uso da aplicação, cobrindo desde a autenticação até o consumo de serviços de automação contábil e consulta fiscal.

2. Planejamento do Backend

A segunda etapa envolve o desenvolvimento da camada de **backend**, responsável pela lógica de negócio, autenticação, controle de usuários, armazenamento de dados e integração com os módulos de IA. O backend será construído com o framework **FastAPI (Python)** devido à sua performance, tipagem forte e compatibilidade com APIs RESTful.

Cada funcionalidade principal será implementada como um **microsserviço independente**, promovendo escalabilidade e manutenção modular. Os serviços planejados incluem:

- **Auth Service:** gerenciamento de login, registro e autenticação via **JWT (JSON Web Tokens)**.
- **Document Service:** responsável pelo armazenamento de arquivos e pela integração com o módulo de **OCR (Tesseract OCR)** para leitura e extração automática de dados de notas fiscais.
- **Agent Orchestrator Service:** coordena as requisições enviadas ao agente contábil e gerencia os processos de inferência baseados em **RAG (Retrieval-Augmented Generation)**, conectando o modelo de linguagem a uma base vetorial de documentos.
- **Billing Service:** controla o saldo de tokens e contabiliza o consumo de recursos conforme a interação dos usuários com a plataforma.
- **API Gateway (NGINX):** atuará como ponto de entrada da aplicação, roteando requisições e garantindo autenticação, segurança e balanceamento de carga.

Cada microsserviço possuirá seu próprio banco de dados, sendo o **PostgreSQL** o principal sistema de armazenamento relacional, com o **MongoDB** sendo utilizado para registros não estruturados e logs. O **Redis** será empregado para filas de tarefas assíncronas, principalmente nas operações de OCR e chamadas a LLMs.

3. Integração Frontend–Backend

A comunicação entre o frontend e o backend ocorrerá via **REST API**, com o uso de **Axios** e **React Query** para o gerenciamento das requisições HTTP e cache de dados.

Os endpoints expostos pelo backend fornecerão respostas em formato **JSON**, garantindo interoperabilidade e agilidade.

A autenticação será feita por meio de **tokens JWT**, armazenados localmente pelo navegador, enquanto o controle de sessões e permissões será realizado no gateway.

O objetivo é garantir que todas as ações realizadas na interface (upload de documento, conversa com o agente, consulta de saldo, etc.) sejam refletidas em tempo real no backend.

4. Modularidade e Escalabilidade

Toda a arquitetura foi desenhada com foco em **modularidade e expansão futura**. Isso permitirá a adição de novos agentes de IA (como agentes fiscais, jurídicos ou de RH) sem a necessidade de alterar a estrutura principal da aplicação.

Novos módulos poderão ser registrados diretamente no banco de dados e configurados no orquestrador de agentes, tornando o sistema flexível e preparado para crescimento contínuo.

5. Futuras Etapas de Desenvolvimento

Após a conclusão do backend e da integração com o frontend, o projeto passará pelas seguintes fases:

- Testes unitários e de integração entre os microserviços.
- Otimização de desempenho e segurança de endpoints.
- Deploy em ambiente **Docker** utilizando **Docker Compose** para orquestração local e futura migração para **AWS Elastic Beanstalk**.
- Implementação de logs centralizados e monitoramento via **Prometheus e Grafana**.

8 Resultados e Discussões

Até o momento, o desenvolvimento do projeto resultou em uma **interface funcional completa**, que representa o primeiro protótipo operacional da plataforma de automação contábil com agentes de IA. O frontend, construído em **React** e **Tailwind CSS**, permitiu validar o **fluxo de navegação do usuário**, a **organização das funcionalidades** e o **design da experiência de uso**, simulando o comportamento final da aplicação.

Entre os principais resultados alcançados destacam-se:

- A criação de uma **estrutura visual clara e responsiva**, permitindo o uso da plataforma em dispositivos de diferentes tamanhos de tela.
- A implementação das telas principais da aplicação, incluindo **Login e Autenticação**, **Dashboard de Atividades Recentes**, **Gerenciamento de Documentos** e **Agente Contábil** com interface de chat interativo.
- O **funcionamento completo da camada de interface**, incluindo a navegação entre páginas, exibição dinâmica de dados simulados e organização de componentes

reutilizáveis que facilitarão a integração futura com o backend.

- O **design validado da experiência do usuário (UI/UX)**, priorizando fluidez, legibilidade e foco na usabilidade para microempreendedores individuais (MEIs).

Durante o desenvolvimento, o protótipo serviu como **ambiente de testes de conceito** (proof of concept) para demonstrar como um sistema SaaS baseado em agentes de IA pode integrar múltiplas funcionalidades contábeis — como leitura de documentos fiscais, extração automática de dados e suporte interativo via chat.

Os resultados obtidos até esta etapa confirmam a **viabilidade do modelo proposto**, tanto em termos de arquitetura modular quanto de aplicabilidade prática. A estrutura atual do frontend já se encontra preparada para integração com os microsserviços planejados no backend, que serão responsáveis pelas operações de OCR, autenticação, orquestração de agentes e controle de tokens.

Não houve mudanças significativas nos objetivos do trabalho, mas o desenvolvimento prático evidenciou a necessidade de priorizar a **orquestração modular** e a **escalabilidade da comunicação entre serviços**, pontos que passaram a integrar as etapas futuras do projeto.

Assim, pode-se concluir que a fase atual consolidou a **base estrutural e visual da plataforma**, validando os princípios de design, navegação e interação previstos na concepção inicial. A próxima etapa, centrada na integração com o backend e nos testes de automação de processos contábeis, visa transformar o protótipo funcional em uma aplicação SaaS totalmente operante.

9 Conclusões

O presente trabalho teve como objetivo o desenvolvimento de uma **plataforma inteligente de automação contábil**, concebida como um **sistema SaaS modular e escalável**, capaz de integrar agentes de Inteligência Artificial (IA) para o processamento de documentos e assistência fiscal voltada a microempreendedores individuais (MEIs). A proposta buscou validar a viabilidade técnica e arquitetural de uma solução baseada em **microsserviços**, integrando tecnologias de **OCR (Reconhecimento Óptico de Caracteres)**, **RAG (Geração Aumentada por Recuperação)** e **Grandes Modelos de Linguagem (LLMs)**.

A metodologia aplicada seguiu os princípios do **desenvolvimento ágil (SCRUM)**, estruturando o projeto em etapas de análise, prototipagem e validação. Foram utilizadas tecnologias modernas e amplamente consolidadas no mercado, como **React, FastAPI, PostgreSQL, MongoDB, Redis e Docker**, com foco em garantir modularidade, desempenho e facilidade de integração entre os componentes da plataforma.

Os resultados obtidos até o momento demonstraram a **eficácia da abordagem proposta**. O frontend foi completamente desenvolvido e validado como **prova de conceito**, apresentando todas as telas principais da aplicação — incluindo login, dashboard, gerenciamento de documentos e chat do agente contábil. Essa implementação confirmou a viabilidade do fluxo de navegação, a clareza da experiência do usuário (UI/UX) e a compatibilidade da arquitetura para futura integração com os serviços de backend.

Conclui-se que o sistema projetado **atingiu os objetivos iniciais**, especialmente no que diz respeito à construção de uma base sólida para automação contábil em nuvem. A proposta mostrou-se **tecnicamente aplicável e escalável**, fornecendo um modelo replicável para outras áreas de negócio, como gestão financeira, fiscal e de recursos humanos.

Entretanto, algumas **limitações** ainda são observadas, como a ausência da camada de backend em produção e a integração efetiva com modelos de IA e OCR em tempo real. Essas pendências serão abordadas nas próximas etapas do projeto, nas quais serão implementados os microsserviços independentes e o sistema de faturamento baseado em tokens de consumo.

Por fim, o trabalho contribui para o avanço de soluções baseadas em IA na contabilidade, demonstrando que é possível combinar automação inteligente, modularidade arquitetural e acessibilidade em um único ecossistema digital. As próximas pesquisas poderão expandir o projeto com **novos agentes especializados**, aprimoramento de modelos de linguagem e integração com sistemas governamentais de emissão e validação fiscal.

Referências

BIGHETI, W. P. **Arquitetura de automação e controle orientada a microsserviços para a Indústria 4.0**. 2020. Tese (Doutorado em Engenharia Elétrica) – Universidade Federal de Uberlândia, Uberlândia, 2020.

DRAGONI, N. et al. **Microservices: Yesterday, Today, and Tomorrow**. In: *Present and Ulterior Software Engineering*. Springer, 2017. p. 195–216. DOI:

10.1007/978-3-319-67425-4_12.

KATTI, A.; REIS, D.; BAEK, Y. **Chargrid: Towards Understanding 2D Documents**. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Brussels: Association for Computational Linguistics, 2018. p. 4459–4470. DOI: 10.18653/v1/D18-1489.

KRIEGER, A.; BÜCHLER, J.; ZACHARIAS, V. **Automated Invoice Processing with AI and Machine Learning**. *International Journal of Computer Applications*, v. 183, n. 1, p. 15–22, 2023.

LEITE, G. R. **SaaS Process: Um Processo de Desenvolvimento para Software como Serviço**. 2022. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, Recife, 2022.

LEWIS, P.; OGUNLANA, D.; LEE, H.; STABLES, R.; RIEDEL, S. **Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks**. In: *Advances in Neural Information Processing Systems (NeurIPS 2020)*. Vancouver, 2020.

PULIKONDA, S. **Real-Time Regulatory Intelligence Framework: LLM-powered compliance automation for financial services**. *World Journal of Advanced Engineering Technology and Sciences*, v. 5, n. 2, p. 28–36, 2025. DOI: 10.58257/wjaets.v5i2.512.

ROHAIME, M.; MONTES, A.; KUMAR, S. **Integrated Invoicing Solution: A Robotic Process Automation with AI and OCR Approach**. In: *Proceedings of the 2022 International Conference on Intelligent Systems and Applications*. Dubai: IEEE, 2022. p. 101–109. DOI: 10.1109/ISAA.2022.9945123.

VEMULAPALLI, V. **Revolutionizing Bookkeeping: Retrieval-Augmented AI Agents for Modern Accounting**. *International Journal of Artificial Intelligence Research*, v. 13, n. 4, p. 84–100, 2025.

Documentação Técnica e Materiais de Apoio

FASTAPI. *FastAPI Documentation*. Disponível em: <https://fastapi.tiangolo.com/>. Acesso em: 10 out. 2025.

REACT. *React – A JavaScript library for building user interfaces*. Meta Open Source. Disponível em: <https://react.dev/>. Acesso em: 10 out. 2025.

TAILWIND CSS. *Tailwind CSS Documentation*. Disponível em: <https://tailwindcss.com/docs>.

Acesso em: 10 out. 2025.

DOCKER. *Docker Documentation*. Disponível em: <https://docs.docker.com/>. Acesso em: 10 out. 2025.

NGINX. *NGINX Reverse Proxy Guide*. F5 NGINX Inc. Disponível em: <https://nginx.org/en/docs/>. Acesso em: 10 out. 2025.

POSTGRESQL. *PostgreSQL Official Documentation*. Disponível em: <https://www.postgresql.org/docs/>. Acesso em: 10 out. 2025.

MONGODB. *MongoDB Documentation*. Disponível em: <https://www.mongodb.com/docs/>. Acesso em: 10 out. 2025.

TESSERACT OCR. *Tesseract Open Source OCR Engine*. Google Developers. Disponível em: <https://github.com/tesseract-ocr/tesseract>. Acesso em: 10 out. 2025.

CELERY PROJECT. *Distributed Task Queue Documentation*. Disponível em: <https://docs.celeryq.dev/>. Acesso em: 10 out. 2025.