

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

GUSTAVO SANTOS SCHNEIDER

**DAILY CONTÁBIL - SISTEMA WEB PARA TREINAMENTO INTERNO E
GESTÃO DE ROTINA EM ESCRITÓRIOS DE CONTABILIDADE**

GUARAPUAVA

2025

GUSTAVO SANTOS SCHNEIDER

**DAILY CONTÁBIL - SISTEMA WEB PARA TREINAMENTO INTERNO E
GESTÃO DE ROTINA EM ESCRITÓRIOS DE CONTABILIDADE**

**Daily Contábil - Web System for Internal Training and Routine Management
in Accounting Offices**

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do
título de Tecnólogo em Tecnologia em Sistemas
para Internet do Curso Superior de Tecnologia
em Sistemas para Internet da Universidade
Tecnológica Federal do Paraná.

Orientadora: Dr^a Renata Luiza Stange

Coorientador: Dr. Diego Marczal

GUARAPUAVA

2025



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

GUSTAVO SANTOS SCHNEIDER

**DAILY CONTÁBIL - SISTEMA WEB PARA TREINAMENTO INTERNO E
GESTÃO DE ROTINA EM ESCRITÓRIOS DE CONTABILIDADE**

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do
título de Tecnólogo em Tecnologia em Sistemas
para Internet do Curso Superior de Tecnologia
em Sistemas para Internet da Universidade
Tecnológica Federal do Paraná.

Data de aprovação: 02/dezembro/2025

Renata Luiza Stange
Doutora
Universidade Tecnológica Federal do Paraná

Diego Marczal
Doutor
Universidade Tecnológica Federal do Paraná

Luciano Ogiboski
Doutor
Universidade Tecnológica Federal do Paraná

**GUARAPUAVA
2025**

RESUMO

A rotina dos escritórios de contabilidade é marcada por uma alta demanda de tarefas e constantes alterações na legislação, e, quando há necessidade de ingresso de novos colaboradores, há uma necessidade de um intenso treinamento interno prático, o que gera um gargalo operacional. Este processo sobrecarrega os colaboradores mais experientes quando há rotatividade de funcionários e resulta na perda de conhecimento, o qual muitas vezes levou meses ou anos para ser adquirido. Diante disso, este trabalho foi idealizado com o objetivo de contribuir com a organização desse processo, visando facilitar a capacitação de novos colaboradores. Com isso, acredita-se que, ao promover a organização desse conhecimento, é possível otimizar o tempo de toda a equipe. O Daily Contábil é um sistema web que atua como uma universidade interna virtual, centralizando o conhecimento do escritório e promovendo a sincronia da equipe, o que agiliza o ingresso de novos profissionais e libera os funcionários mais experientes para atividades estratégicas. Pelo sistema, os colaboradores experientes podem cadastrar tutoriais completos, divididos em etapas, contendo vídeo-aulas, textos, planilhas e outros recursos. Os demais usuários podem navegar pelas categorias de conteúdo (como "Fiscal" e "Contábil"), pesquisar por tutoriais e até mesmo consultar procedimentos específicos anexados a cada empresa cliente.

Palavras-chave: contabilidade; treinamento interno; perda de conhecimento; sistema web; universidade interna virtual.

ABSTRACT

The routine of accounting firms is marked by a high demand for tasks and constant changes in legislation, and, when there is a need for new employees to join, there is a need for intense practical internal training, which creates an operational bottleneck. This process overburdens the most experienced employees when there is staff turnover and results in the loss of knowledge, which often took months or years to be acquired. Given this, this work was idealized with the objective of contributing to the organization of this process, aiming to facilitate the training of new collaborators. With this, it is believed that by promoting the organization of this knowledge, it is possible to optimize the entire team's time. Daily Contábil is a web system that acts as a virtual internal university, centralizing the office's knowledge and promoting team synchronization, which speeds up the ingress of new professionals and frees up experienced employees for strategic activities. Through the system, experienced collaborators can register complete tutorials, divided into steps, containing video-lessons, texts, spreadsheets, and other resources. The other users can browse content categories (such as "Fiscal" and "Accounting"), search for tutorials, and even consult specific procedures attached to each client company.

Keywords: accounting; internal training; loss of knowledge; web system; virtual internal university.

LISTA DE FIGURAS

Figura 1 – Exemplo de quadro Kanban.	13
Figura 2 – Tela de Login	22
Figura 3 – Tela inicial	22
Figura 4 – Página de Categorias	23
Figura 5 – Página de Tutoriais	23
Figura 6 – Página Individual de um Tutorial	24
Figura 7 – Página de Empresas	24
Figura 8 – Página Individual de uma Empresa	25
Figura 9 – Página de Usuários	25
Figura 10 – Interface do sistema contendo o menu superior (<i>header</i>) e o menu lateral (<i>sidebar</i>).	30
Figura 11 – Seção hero contendo proposta de valor e CTA.	34

LISTA DE ABREVIATURAS E SIGLAS

Siglas

API	<i>Application Programming Interface</i>
AWS	<i>Amazon Web Services</i>
B2B	<i>Business to Business</i>
CD	<i>Continuous Delivery</i>
CFC	<i>Conselho Federal de Contabilidade</i>
CI	<i>Continuous Integration</i>
CRUD	<i>Create, Read, Update and Delete</i>
CSS	<i>Cascading Style Sheets</i>
CTA	<i>Call to Action</i>
HTML	<i>HyperText Markup Language</i>
UI	<i>User Interface</i>
UX	<i>User Experience</i>
WIP	<i>Work-In-Progress</i>

SUMÁRIO

1	INTRODUÇÃO	8
1.1	Objetivo Geral	9
1.1.1	Objetivos específicos	9
1.2	Justificativa	9
2	MATERIAIS E MÉTODOS	11
2.1	Materiais	11
2.2	Métodos	12
2.2.1	MoSCoW	12
2.2.2	Kanban	12
2.2.3	Etapas de Desenvolvimento	13
3	ANÁLISE E PROJETO DO SISTEMA	15
3.1	Levantamento de Requisitos	15
3.2	Análise de Requisitos	15
3.2.1	Requisitos Funcionais e Não Funcionais	16
3.2.2	Histórias de Usuário	18
3.2.3	Protótipos de Telas	21
4	DESENVOLVIMENTO DO SISTEMA	26
4.1	Escopo de desenvolvimento	26
4.1.1	Requisitos Gerais	26
4.1.2	Requisitos de Administrador	26
4.1.3	Requisitos de Usuário	27
4.2	Configuração do Ambiente	27
4.3	Integração Contínua e Entrega Contínua (<i>Continuous Integration</i> (CI)/<i>Continuous Delivery</i> (CD))	27
4.4	Infraestrutura de Produção e <i>Deploy</i>	28
4.5	Implementação das Funcionalidades	28
4.5.1	Autenticação de usuário	29
4.5.2	Menus de Navegação	29
4.5.3	Tela de Empresas	30
4.5.4	Biblioteca de Conteúdos	30

4.5.5	Tela de Tutoriais	31
4.5.6	Gerenciamento de Categorias	32
4.5.7	<i>Landing Page</i>	33
4.5.8	Filtro e Pesquisa	34
5	CONCLUSÃO	35
5.1	Considerações Finais	35
5.2	Trabalhos Futuros	36
	REFERÊNCIAS	37

1 INTRODUÇÃO

A legislação brasileira exige que as empresas mantenham registros escriturados de todas as suas atividades econômicas (BRASIL, 2002). Isso inclui desde pequenas aquisições de materiais de escritório até operações comerciais de grande porte. Cada movimentação financeira deve ser registrada com precisão e informada aos órgãos competentes, como governos, bancos, investidores, fornecedores e outros agentes reguladores (CONTADOR AGORA, 2021). Em meio a essa exigência, destaca-se o papel da contabilidade. A contabilidade é uma ciência aplicada que estuda os métodos de registro, controle e interpretação das movimentações financeiras de uma entidade (OXFORD LANGUAGES, 2021). Suas atividades são exercidas por profissionais habilitados, denominados contadores, seja de forma autônoma ou reunidos em escritórios especializados. Segundo dados do *Conselho Federal de Contabilidade* (CFC), no início de 2024, o Brasil possuía 89.969 empresas contábeis e 529.168 profissionais registrados (CONSELHO FEDERAL DE CONTABILIDADE, 2024).

Entre as principais atribuições da contabilidade estão a elaboração das demonstrações financeiras, como o balanço patrimonial e a demonstração de resultados do exercício¹ (CONTABILIZEI, 2024). Tais informações são essenciais para a tomada de decisão de empresários e gestores, podendo impactar diretamente a sustentabilidade e o sucesso das organizações (AGENDA CONTÁBIL (2024).

Apesar de sua importância estratégica, o setor contábil enfrenta desafios recorrentes, como elevado volume de trabalho, prazos curtos, escassez de mão de obra qualificada e constantes alterações na legislação (AGENDA CONTÁBIL, 2024). Superar essas dificuldades requer conhecimento técnico e experiência prática, adquiridos majoritariamente no próprio ambiente de trabalho (APARECIDO, 2023). Contudo, a carência de profissionais qualificados e as dificuldades na transmissão de conhecimento interno contribuem para a sobrecarga dos colaboradores, impactando diretamente a qualidade dos serviços prestados (CONTABEIS, 2022).

Diante desse cenário, o não cumprimento adequado das obrigações contábeis pode gerar consequências rigorosas, como aplicação de multas, insatisfação dos clientes, perda de credibilidade e até o encerramento das atividades dos escritórios (ATTEMPO, 2024).

A motivação deste projeto surge da vivência profissional do autor em um escritório de contabilidade situado em Guarapuava, Paraná. Durante essa experiência profissional, foi possível observar de forma recorrente os desafios associados à capacitação de novos colaboradores e à sobrecarga dos profissionais mais experientes. Diante disso, identificou-se a possibilidade de desenvolver uma solução tecnológica capaz de oferecer uma forma de amenizar esses prejuízos.

¹ Resultado do exercício: relatório contábil que tem a finalidade de descrever as operações realizadas pela empresa no período analisado. Ela permite confrontar receitas e despesas, apontando o resultado líquido do negócio, seja em relação ao lucro ou prejuízo.

Diante desse contexto, este trabalho propõe o desenvolvimento de um sistema de software voltado ao gerenciamento de treinamentos internos de um escritório de contabilidade. A plataforma permitirá que os escritórios de contabilidade organizem e disponibilizem materiais educativos, como vídeo-tutoriais, textos, planilhas e outros recursos, produzidos internamente ou por terceiros. A proposta visa apoiar a capacitação dos profissionais, reduzir a recorrência de erros operacionais e contribuir para a melhoria da produtividade e da qualidade dos serviços contábeis.

Assim, os resultados obtidos e as propostas apresentadas reforçam a relevância do *Daily Contábil* como uma iniciativa inovadora no contexto da contabilidade digital, contribuindo tanto para o desenvolvimento tecnológico da área quanto para a formação contínua de profissionais mais preparados e eficientes.

1.1 Objetivo Geral

Desenvolver um sistema de software que permita aos escritórios de contabilidade organizar e disponibilizar materiais de treinamento, como vídeo-tutoriais, textos, planilhas e outros recursos, a fim de apoiar a capacitação prática de seus colaboradores.

1.1.1 Objetivos específicos

- Levantar e analisar os requisitos funcionais do sistema, com foco nas necessidades dos escritórios de contabilidade para treinamentos internos.
- Desenvolver funcionalidades que permitam o cadastro, organização e gestão de diferentes formatos de materiais (vídeos, textos, planilhas, documentos, links, etc.).
- Implementar o módulo de gerenciamento de conteúdo, permitindo ao administrador o cadastro, a edição e a exclusão da estrutura hierárquica de categorias, como "Departamento Pessoal, subcategorias, como "Como contabilizar despesas" e tutoriais organizados;
- Implementar a funcionalidade de gestão de empresas, possibilitando ao administrador associar diferentes tipo de conteúdo a cada empresa cadastrada.
- Implementar mecanismos de categorização, busca e filtragem dos conteúdos, facilitando o acesso rápido aos materiais de acordo com atividades ou áreas de atuação.

1.2 Justificativa

A capacitação eficaz de novos profissionais impacta diretamente na qualidade dos serviços prestados por escritórios de contabilidade, contribuindo para a execução adequada das

atividades no devido prazo e de forma coerente, contribuindo para a execução adequada das atividades, evitando prejuízos operacionais.

Atualmente, a transmissão de conhecimento em escritórios de contabilidade ocorre, majoritariamente, de maneira direta e presencial, de forma presencial e assistida. Esse modelo apresenta algumas limitações importantes, como a sobrecarga de informações de um curto período de tempo, o que dificulta a absorção do conteúdo por parte do aprendiz. Além disso, o mesmo processo precisa ser repetido continuamente para diferentes colaboradores, exigindo a destinação de horas de trabalho que poderiam ser utilizadas em outras atividades importantes.

Por meio do trabalho proposto, o mercado de trabalho contábil pode tornar-se mais atraativo e acessível. Isso porque, ao contribuir para a redução da sobrecarga de trabalho dos profissionais mais experientes, a proposta incentiva a contratação, por parte dos escritórios, de profissionais sem conhecimento prático prévio, facilitando sua inserção no setor.

Esse contexto se torna ainda mais relevante diante da iminente reforma tributária no Brasil, prevista para o ano de 2026, a qual promete impactar significativamente a rotina diária dos escritórios (FAZENDA, 2024). Isso exigirá um estudo aprofundado por parte dos profissionais da área, para que possam se adaptar e executar suas atividades de acordo com as novas exigências estabelecidas.

Enquanto colaborador de um escritório de contabilidade, o autor vivenciou diariamente os desafios enfrentados nesse ambiente, especialmente no que diz respeito à capacitação de novos colaboradores e à sobrecarga de trabalho dos profissionais mais experientes. Essa experiência despertou o interesse em buscar uma solução prática e eficiente para enfrentar esses desafios e contribuir com a melhoria da rotina deste escritório.

Este projeto mostra-se viável devido ao conhecimento aprofundado do problema por parte do autor, somado ao conhecimento das tecnologias necessárias para o desenvolvimento do sistema, como linguagens de programação e banco de dados. Além disso, o contexto atual do setor contábil torna o momento oportuno para o desenvolvimento de uma plataforma voltada à capacitação prática de novos profissionais.

2 MATERIAIS E MÉTODOS

Este capítulo descreve as tecnologias, ferramentas e métodos que foram utilizados no desenvolvimento do Daily Contábil.

2.1 Materiais

Os materiais incluem ferramentas, tecnologias e ambientes de desenvolvimento que foram utilizados para uma maior eficiência, organização e qualidade do projeto. A seguir, estão listados e descritos os principais recursos que foram empregados:

- **Laravel:** Framework PHP para desenvolvimento de sistemas web que utilizam o padrão MVC (model, view, controller). Ele permite a escrita de um código mais simples, legível e possui suporte a recursos avançados que agilizam o processo de desenvolvimento (LARAVEL, 2025a).
- **ClickUp:** É uma plataforma tudo-em-um de gestão de projetos e produtividade que oferece uma gama completa de ferramentas para organizar tarefas, gerenciar projetos, colaborar em equipe e muito mais, tudo em um único espaço de trabalho (CLICKUP, 2025).
- **Git e GitHub:** O Git é um sistema que possibilita o controle de versões de código-fonte. Ele é responsável por guardar o histórico de alterações sempre que alguém modificar algum arquivo que está sendo monitorado por ele (GIT, 2025). GitHub é uma plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git, que disponibiliza uma interface que permite ao desenvolvedor executar ações diárias de forma ágil (GITHUB, 2025).
- **MySQL:** É um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface (MYSQL, 2025).
- **Digital Ocean:** É um provedor de hospedagem em nuvem exclusivo, que oferece serviços de computação em nuvem para entidades de negócios para que possam se escalar implantando aplicativos DigitalOcean que funcionam paralelamente em vários servidores em nuvem sem comprometer o desempenho (OCEAN, 2025).
- **Railway:** O Railway é uma plataforma baseada na nuvem que visa tornar o processo de desenvolvimento mais simples e acessível. Ela permite que desenvolvedores e equipes de desenvolvimento criem, configurem e implementem aplicações de forma rápida e eficiente, sem a necessidade de se preocupar com a infraestrutura (RAILWAY, 2025).

2.2 Métodos

Nesta seção será apresentado o plano utilizado para o desenvolvimento do projeto. Para otimização do desenvolvimento, foram aplicados princípios de metodologia ágil presentes na engenharia de software, visando garantir um resultado final consistente. Para isso, foram utilizadas duas metodologias principais: Kanban e MoSCoW.

2.2.1 MoSCoW

A metodologia MoSCoW é uma técnica de priorização de requisitos usada em gerenciamento, análise de negócios, gerenciamento de projetos e desenvolvimento de software. Tem o objetivo de chegar a um entendimento comum com as partes interessadas sobre a importância que atribuem à entrega de cada requisito. Seu nome é um acrônimo que representa quatro categorias principais de prioridades:

- *Must-have* (deve ter): requisitos essenciais do sistema, que certamente serão desenvolvidos;
- *Should-have* (deveria ter): requisitos importantes que deveriam existir, mas não são cruciais para o desenvolvimento inicial do projeto;
- *Could-have* (poderia ter): requisitos desejáveis, mas não essenciais para o desenvolvimento.
- *Won't have* (não terá): requisitos que não serão incluídos no escopo do projeto atual.

Esta metodologia foi aplicada na condução das entrevistas realizadas durante o levantamento de requisitos, permitindo a coleta de informações e a identificação de requisitos relevantes para o desenvolvimento do sistema. Esses requisitos, junto aos previamente definidos, foram organizados em uma tabela e categorizados conforme a prioridade estabelecida.

2.2.2 Kanban

Kanban é uma metodologia ágil que orquestra a progressão contínua de tarefas por meio de fluxos de trabalho visualizados (ATLASSIAN, 2025). Estabelecer um fluxo kanban estruturado é essencial para a sua implementação. Isso garante a progressão fluída de tarefas e um fluxo de trabalho otimizado. O fluxo kanban será estruturado da seguinte forma: Kanban foi uma metodologia ágil que orquestrou a progressão contínua de tarefas por meio de fluxos de trabalho visualizados (ATLASSIAN, 2025). Estabelecer um fluxo kanban estruturado foi essencial para a sua implementação. Isso garantiu a progressão fluída de tarefas e um fluxo de trabalho otimizado. O fluxo kanban foi estruturado da seguinte forma:

- Visualização do fluxo de trabalho: foram utilizados os quadros kanban, que retrataram de forma organizada cada etapa do desenvolvimento.
- Padronização do fluxo de trabalho: foi necessário definir e padronizar os estágios do fluxo de trabalho. Estágios comuns foram: "a fazer", "em progresso" e "feito".
- Limitação do Trabalho em Progresso - *Work-In-Progress* (WIP): a ideia foi definir um número máximo de tarefas que poderiam estar em cada coluna, com o objetivo de evitar a sobrecarga de trabalho e manter um fluxo de trabalho estável.
- Implementação de Ciclos de Retroalimentação (*Feedback Loops*): realizaram-se reuniões regulares com orientadores para analisar o fluxo, os problemas e as métricas.

A Figura 1 apresenta um exemplo de quadro Kanban com colunas de acompanhamento de tarefas.



Figura 1 – Exemplo de quadro Kanban.

2.2.3 Etapas de Desenvolvimento

O desenvolvimento do sistema foi dividido em etapas estruturadas da seguinte maneira:

- Levantamento de Requisitos: O levantamento de requisitos foi realizado por meio de um método qualitativo, utilizando entrevistas com profissionais da área que atuam em escritórios de contabilidade. As entrevistas tinham algumas perguntas pré-definidas, mas a busca foi por um aprofundamento nas experiências e perspectivas dos entrevistados, visando a validação do problema apresentado.
- Modelagem do Banco de Dados: Com base nos requisitos definidos, foi realizada a modelagem do banco de dados relacional, definindo o esquema de tabelas e seus respectivos relacionamentos para suportar as funcionalidades do sistema.

- Configuração do Ambiente de Desenvolvimento: O ambiente foi preparado para o desenvolvimento com o Framework Laravel, o que inclui a configuração do editor de código e a instalação de todas as bibliotecas e dependências necessárias.
- Implementação: Nesta etapa, foi realizado o desenvolvimento do back-end (regras de negócio e *Application Programming Interface* (API)) e do front-end (interface do usuário) do sistema, garantindo o pleno funcionamento dos requisitos definidos.

3 ANÁLISE E PROJETO DO SISTEMA

Este capítulo tem por objetivo descrever e analisar os requisitos do sistema, baseado na pesquisa de usuários, bem como estruturar tecnicamente o seu desenvolvimento utilizando as informações coletadas.

3.1 Levantamento de Requisitos

Foram selecionados dois escritórios de contabilidade para participar da entrevista de análise, Escritório Contábil Polly e Bine Contabilidade & Consultoria, permitindo uma análise aprofundada da relevância dos problemas no contexto real do público-alvo. O objetivo central era coletar percepções diretas sobre a viabilidade da proposta, seus pontos fortes e fracos, e obter sugestões que orientassem o desenvolvimento do Daily Contábil. As entrevistas foram gravadas em áudio e sua condução se deu em quatro etapas sequenciais:

1. **Imersão e Contextualização:** A etapa inicial consistiu em compreender a rotina, a dinâmica das equipes e os processos de trabalho de cada escritório, a fim de contextualizar os desafios enfrentados diariamente.
2. **Validação do Problema:** A seguir, a conversa foi direcionada para as dificuldades no treinamento de novos colaboradores e na gestão do conhecimento interno. Isso foi feito por meio de perguntas pré-definidas relacionadas ao método atual utilizado por esses escritórios, analisando sua eficácia para confirmar se os desafios apresentados neste trabalho correspondiam à realidade dos entrevistados.
3. **Apresentação da Proposta:** Com o problema validado, o conceito do sistema foi apresentado, seguido pela demonstração de um protótipo que apresentava as principais funcionalidades elaboradas para contribuir com a melhoria dos problemas enfrentados.
4. **Coleta de Feedback e Levantamento de Requisitos:** Por fim, coletou-se o feedback em relação à proposta apresentada. Os profissionais entrevistados foram incentivados a expressar suas opiniões, sugerir funcionalidades, modificações, avaliar pontos positivos e negativos e a viabilidade do sistema.

3.2 Análise de Requisitos

Com base na análise das entrevistas, foram definidos os requisitos do sistema. Cada requisito busca identificar uma funcionalidade do sistema. A priorização dessas funcionalidades foi realizada por meio do método MoSCoW, garantindo que o desenvolvimento se concentre nos recursos de maior impacto.

3.2.1 Requisitos Funcionais e Não Funcionais

Para facilitar a compreensão, os requisitos serão apresentados de forma global, representando funcionalidades que não atreladas a nenhum perfil de usuário específico e também de acordo com os dois perfis de usuário:

- Administrador: geralmente um colaborador experiente responsável pela curadoria e gestão do conteúdo de treinamento;
- Usuário Padrão: perfil destinado aos novos colaboradores que irão consumir os materiais para apoiar sua rotina de trabalho.

A Tabela 1 foi dividida em duas categorias: a primeira detalha os Requisitos Não Funcionais (RNFs), que estabelecem os critérios de qualidade e o comportamento esperado do sistema, ou seja, como ele deve funcionar; a segunda parte apresenta os Requisitos Funcionais Gerais, que são as funcionalidades não atreladas a um perfil de usuário específico e se aplicam ao sistema como um todo.

Tabela 1 – Requisitos Gerais

Identificação	Requisito	Descrição	MoSCoW
RNF01	Autenticação de usuário	Login, logout e controle de acesso.	Must
RNF02	Controle de acesso	Dois níveis de permissão distintos: administrador e usuário padrão.	Must
RNF03	Desempenho	Possuir tempos de resposta rápidos para carregamento de páginas e resultado de buscas.	Should
RNF04	Compatibilidade	Possuir compatibilidade com os principais navegadores de internet modernos.	Should
RF01	Sugestão de gravação	O sistema deve apresentar uma lista de sugestões de vídeos a serem gravados pelos administradores.	Could
RF02	Gerar relatório de empresas	Ambos os perfis de usuário podem gerar relatório contendo informações de todas as empresas cadastradas no sistema.	Won't

A Tabela 2 contém os requisitos do administrador. Para ela, foram idealizadas funcionalidades que podem agregar valor ao sistema, mas que não foram implementadas na primeira etapa de desenvolvimento. A implementação destes requisitos permanece como uma possibilidade para a evolução contínua do projeto após a conclusão deste trabalho.

Tabela 2 – Requisitos de Administrador

Requisitos: administrador			
Identificação	Requisito	Descrição	MoSCoW
RF03	Gerenciar categorias	Cadastro, visualização, edição e exclusão de categorias	Must
RF04	Gerenciar subcategorias	Cadastro, visualização, edição e exclusão de subcategorias dentro das categorias existentes	Must
RF05	Gerenciar tutoriais	Cadastro, visualização, edição e exclusão de tutoriais dentro das subcategorias	Must
RF06	Detalhamento de tutoriais	Os tutoriais podem incluir vídeos, imagens, textos, links e planilhas	Must
RF07	Gerenciar empresas	Cadastro, visualização, edição e exclusão de empresas atendidas pelo escritório	Must
RF08	Associar conteúdo à empresa	O administrador pode vincular vídeos, descrições personalizadas e outros materiais pertinentes às empresas.	Must
RF09	Gerenciar usuários	Cadastro, visualização, edição e exclusão de usuários e permissões	Must
RF10	Destaque de conteúdo	O administrador pode marcar um tutorial como "destaque" e ele apareceria na página inicial do usuário	Won't
RF11	Caixa de mensagens/feedback	O administrador pode receber feedbacks e relatórios de erros nos tutoriais	Won't

Para os requisitos do usuário padrão, apresentados na Tabela 3, a priorização pelo método MoSCoW foi essencial para definir o escopo da entrega inicial. Funcionalidades levantadas, como a filtragem avançada de conteúdo (RF14) e um sistema para apontar erros nos tutoriais (RF18), foram identificadas como valiosas, porém classificadas como Could-have e Won't-have, respectivamente. A implementação desses recursos não se dará na etapa atual, uma decisão estratégica que visa garantir uma entrega robusta dentro do cronograma de desenvolvimento.

Tabela 3 – Requisitos de Usuário

Requisitos: usuário padrão			
Identificação	Requisito	Descrição	MoSCoW
RF12	Visualizar tutoriais	O usuário pode navegar pelas categorias e subcategorias para visualizar os tutoriais	Must
RF13	Pesquisar tutoriais	O usuário pode utilizar a barra de pesquisa para encontrar o tutorial desejado	Must
RF14	Filtrar conteúdo	O usuário pode filtrar os tutoriais dentro de uma subcategoria para refinar a busca	Could
RF15	Consultar empresas	O usuário pode consultar a lista de empresas e visualizar os tutoriais ou anotações específicas que foram associados a cada cliente.	Must
RF16	Marcar como favorito	O usuário pode favoritar um tutorial que visualiza com frequência, criando um atalho ao mesmo	Should
RF17	Marcar como concluído	O usuário pode marcar um tutorial ou uma sequência de tutoriais como concluído	Won't
RF18	Apontar erros nos conteúdos	O usuário pode relatar erros no conteúdo de um tutorial	Won't
RF19	Seção de comentários	Os usuários podem adicionar comentários e iniciar uma discussão em um tutorial	Won't

3.2.2 Histórias de Usuário

Além dos requisitos levantados, foram elaboradas histórias de usuário com base nos itens *Must* e *Should* de cada perfil. Elas têm como objetivo representar, de forma simples e centrada no usuário, as necessidades e expectativas em relação às funcionalidades do aplicativo. As histórias foram escritas a partir das conclusões obtidas no levantamento e na priorização dos requisitos do projeto. Cada história está associada ao requisito correspondente, conforme a identificação definida na Seção 3.2.1, e sua apresentação será organizada de acordo com os tipos de usuário

As histórias de usuário definem o fluxo de interação com as funcionalidades centrais do sistema. Elas foram criadas para assegurar uma jornada de uso lógica, produtiva e sem interrupções.

Feature: Criar uma nova categoria (RF03)

Como um administrador, **quero** criar uma nova categoria com um nome específico, **para** que eu possa começar a organizar novos grupos de tutoriais.

Dado que estou na página de “Categorias”

Quando eu clicar no botão “Adicionar Categoria” e digitar “Departamento Pessoal” no campo de nome e clicar no botão “Salvar Categoria”

Então a categoria “Departamento Pessoal” deve aparecer na lista de categorias existentes.

Feature: Criar uma nova subcategoria (RF04)

Como um administrador, **quero** criar uma nova subcategoria dentro de uma categoria existente, **para** que os tutoriais sobre um mesmo assunto fiquem agrupados e fáceis de encontrar.

Dado que estou visualizando a página da categoria “Contábil”

Quando eu clicar no botão “Adicionar Subcategoria” e preencher todos os campos

Então a nova subcategoria deve ser exibida na lista de subcategorias da página.

Feature: Cadastrar um tutorial (RF05)

Como um administrador, **quero** cadastrar um novo tutorial e atribuí-lo à uma subcategoria, **para** que os usuários possam assistir o tutorial e aprender o conteúdo.

Dado que estou visualizando a página da categoria “Contábil” e selecionei a subcategoria “Como contabilizar despesas”

Quando eu clicar no botão “Cadastrar Tutorial” e preencher todos os campos

Então o novo tutorial deve ser exibido na lista de tutoriais da subcategoria selecionada.

Feature: Cadastrar uma empresa (RF07)

Como um administrador, **quero** cadastrar uma nova empresa no sistema **para** atribuir conteúdo à ela.

Dado que estou na página “Empresas”

Quando eu clicar no botão “Cadastrar Empresa” e preencher todos os campos

Então a nova empresa deve ser exibida na lista de empresas cadastradas.

Feature: Adicionar conteúdo à uma empresa (RF08)

Como um administrador, **quero** adicionar uma planilha ao conteúdo de uma empresa cadastrada **para** que os usuários possam consultar as particularidades dela.

Dado que cliquei no card da empresa “UTFPR LTDA” na página “Empresas”

Quando eu clicar no botão “Adicionar Conteúdo” e anexar uma planilha

Então a planilha deve ser exibida na página da empresa.

Feature: Cadastrar um novo usuário (RF09)

Como um administrador, **quero** cadastrar no sistema um usuário para um novo funcionário do escritório **para** que ele possa visualizar os tutoriais.

Dado que estou na página “Usuários”

Quando eu clicar no botão “Adicionar Usuário” e preencher os campos

Então o novo funcionário deve receber acesso ao sistema por meio desse usuário.

Feature: Pesquisar um tutorial (RF13)

Como um usuário, **quero** usar a barra de pesquisa **para** pesquisar um tutorial.

Dado que estou na página da categoria “Contábil”

Quando eu pesquisar “Como contabilizar despesas médicas”

Então a subcategoria ou o tutorial que atenderem à pesquisa devem ser exibidos.

Feature: Assistir um tutorial (RF12)

Como um usuário, **quero** assistir o tutorial “Como contabilizar despesas médicas” **para** aprender esse conteúdo.

Dado que selecionei a subcategoria “Contabilização de Despesas”

Quando poderei assistir ao tutorial.

Então a subcategoria ou o tutorial que atenderem à pesquisa devem ser exibidos.

Feature: Marcar um tutorial como favorito (RF16)

Como um usuário, **quero** marcar o tutorial “Como contabilizar despesas médicas” como favorito **para** assisti-lo novamente.

Dado que estou na página do tutorial e decidi favoritá-lo

Quando eu clicar no botão “marcar como favorito”

Então o tutorial será armazenado na página “Favoritos”.

Feature: Consultar uma empresa (RF16)

Como um usuário, **quero** consultar as informações de uma empresa **para** aprender os procedimentos e particularidades daquela empresa.

Dado que estou na página de empresas

Quando eu clicar na empresa “UTFPR LTDA”

Então poderei visualizar todas as informações e conteúdos cadastrados nessa empresa.

3.2.3 Protótipos de Telas

A etapa de prototipação do sistema foi realizada por meio do desenvolvimento de um protótipo de média a alta fidelidade codificado. A decisão de utilizar diretamente *HyperText Markup Language* (HTML) e *Cascading Style Sheets* (CSS) nessa fase teve como objetivo estratégico não apenas delinear a aparência visual (*User Interface* (UI)), mas também validar aspectos práticos da interação do usuário (*User Experience* (UX)) em um ambiente real de navegador.

Conforme a Figura 2 primeira página a ser acessada é a de login, onde o usuário pode preencher seus dados. O escritório de contabilidade que contratar o sistema receberá um login com permissão de administrador, por meio do qual poderá criar outros usuários, não havendo uma tela para criação de conta fora da autenticação.

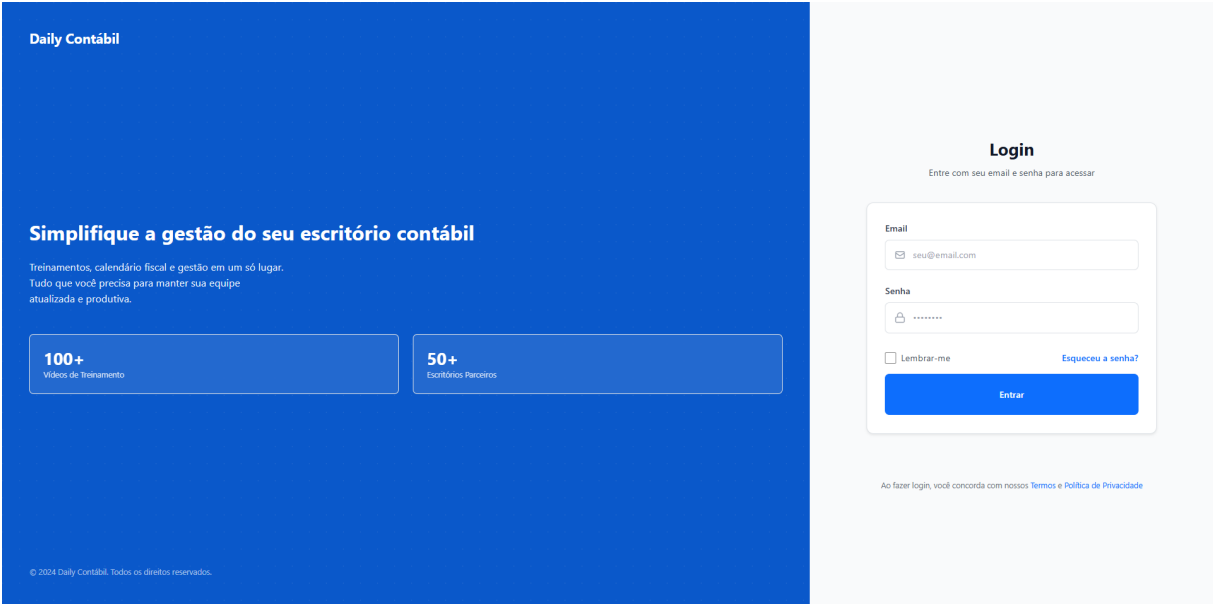


Figura 2 – Tela de Login

Ao efetuar o login, o usuário pode visualizar uma tela simples, apresentada na Figura 3 com algumas informações sobre o sistema e alguns botões de atalho.

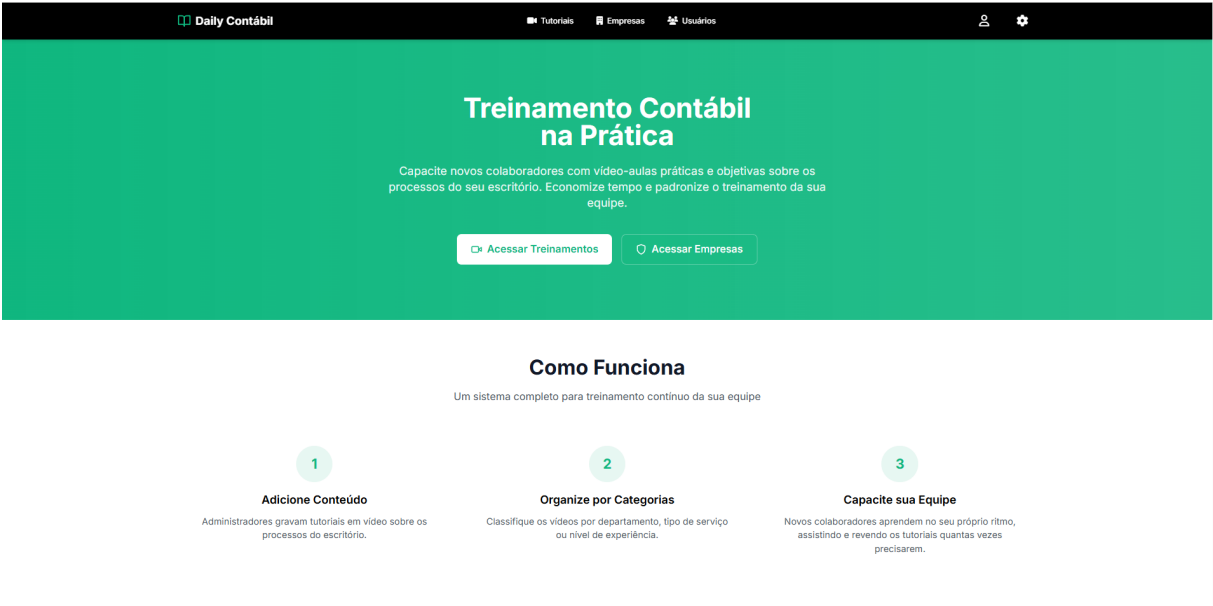


Figura 3 – Tela inicial

Para visualizar tutoriais, a primeira tela que o usuário acessará será a página de categorias, conforme a Figura 4, onde o usuário poderá escolher entre as categorias cadastradas aquela que possui os tutoriais que ele procura. Caso o usuário seja um administrador, estarão disponíveis botões para adição, edição e exclusão de categorias.

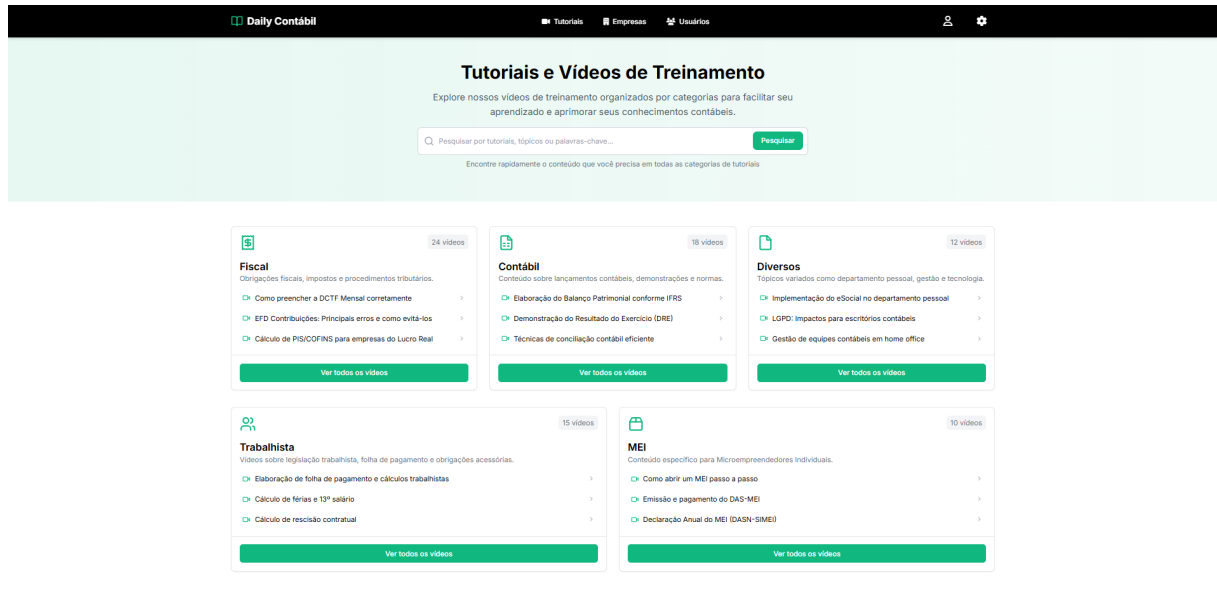


Figura 4 – Página de Categorias

Na página de uma categoria, podemos visualizar as subcategorias pertencentes à ela, além dos tutoriais que essas subcategorias possuem, conforme a Figura 5.

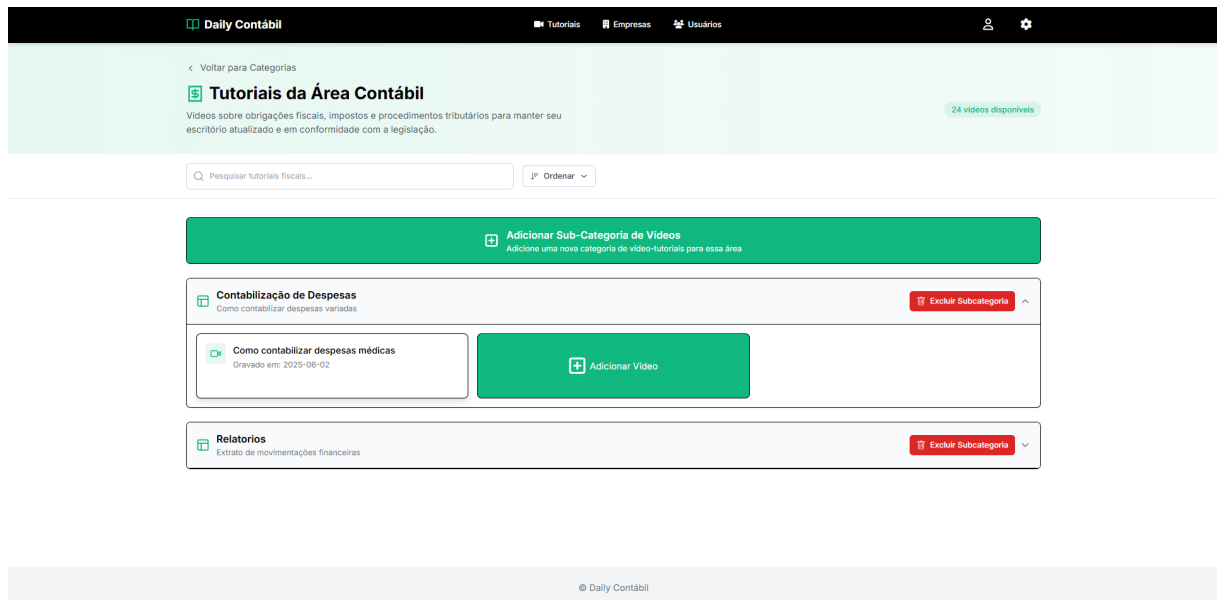


Figura 5 – Página de Tutoriais

Conforme a Figura 6, ao escolhermos um tutorial, podemos visualizar a página individual daquele tutorial, contendo todo o conteúdo atribuído ao mesmo.



Figura 6 – Página Individual de um Tutorial

Na página de empresas, podemos visualizar todas as empresas cadastradas no sistema, conforme a Figura 7. Se o usuário for um administrador, estarão disponíveis as opções de cadastrar, editar e excluir empresas.

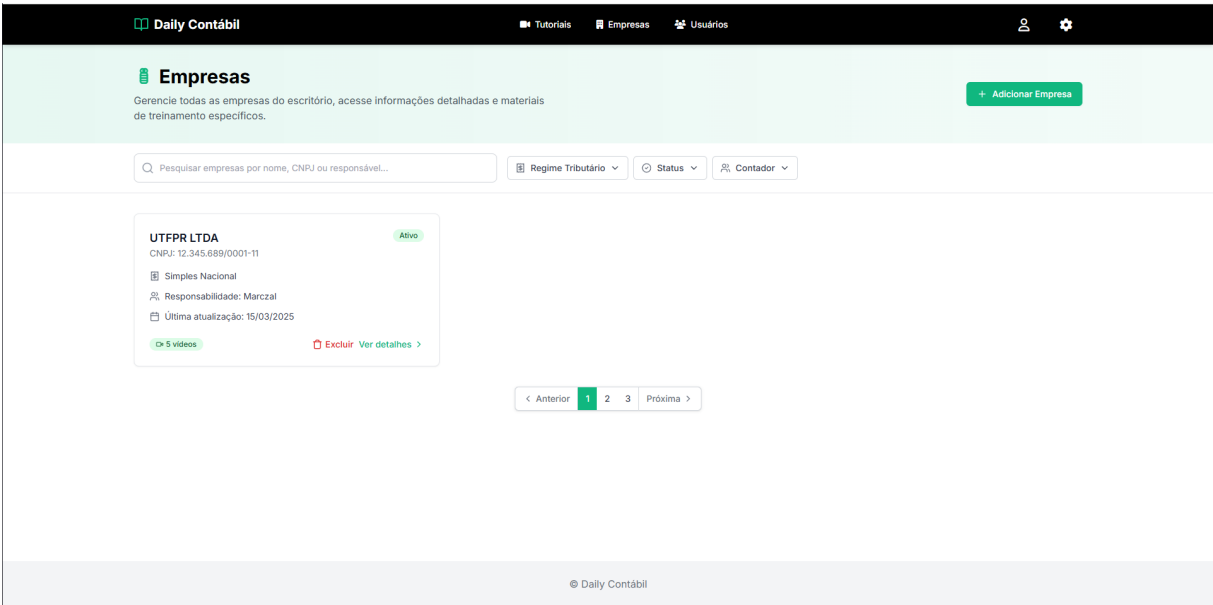


Figura 7 – Página de Empresas

Conforme a Figura 8, ao escolhermos uma empresa, podemos visualizar a página individual da mesma, contendo todo o conteúdo atribuído à ela.

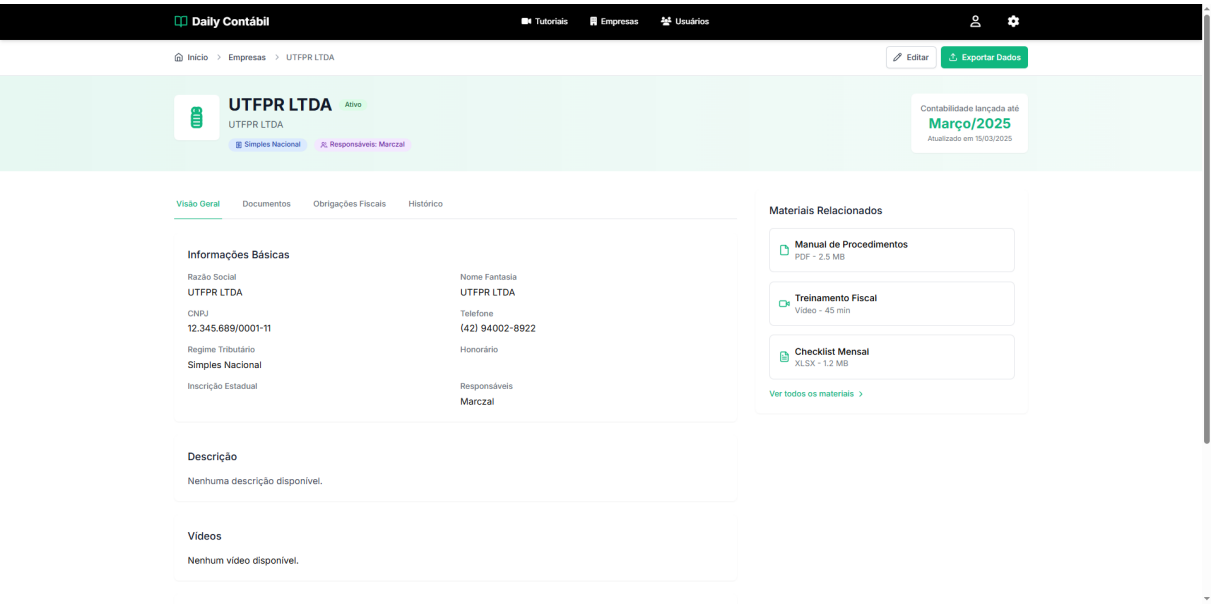


Figura 8 – Página Individual de uma Empresa

Conforme a Figura 9, a página de usuários, visível apenas ao administradores, permite à eles gerenciarem todos os usuários do sistema.

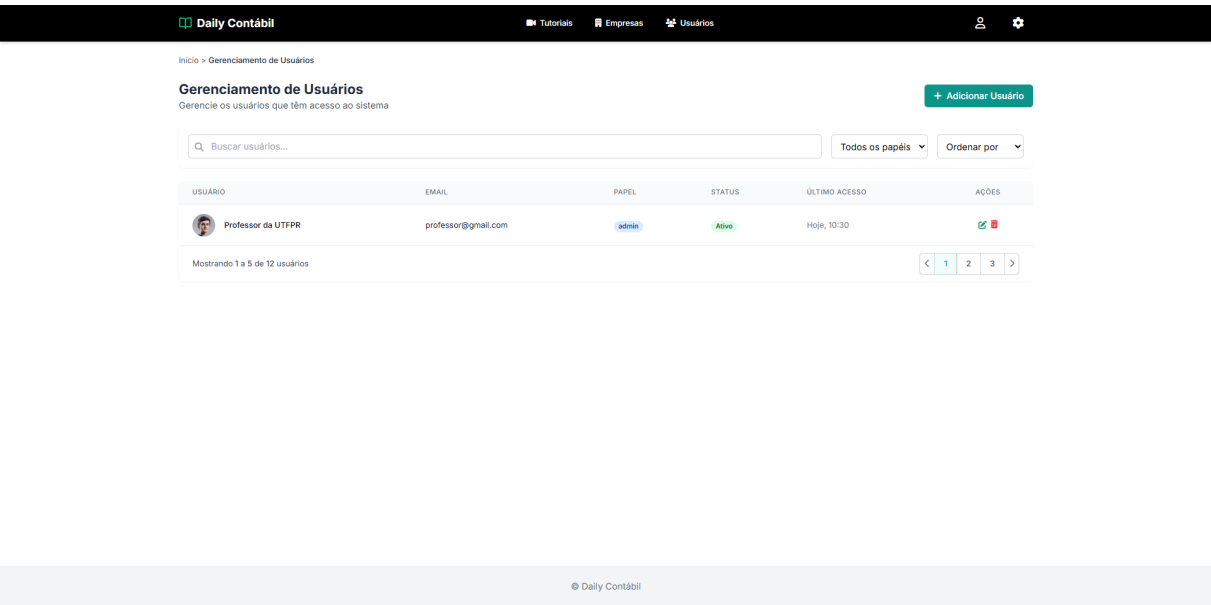


Figura 9 – Página de Usuários

4 DESENVOLVIMENTO DO SISTEMA

O processo de desenvolvimento consistiu na implementação de uma aplicação web *full-stack*, traduzindo os requisitos do projeto em uma solução técnica funcional. A estratégia de implementação foi segmentar o trabalho por *Features* (funcionalidades) completas, onde cada entrega envolvia o ciclo completo de desenvolvimento: desde a estruturação do banco de dados e a lógica de negócios no *back-end* até a construção dos componentes de interface no *front-end*. Esta abordagem permitiu que cada funcionalidade fosse entregue de forma coesa.

4.1 Escopo de desenvolvimento

Durante o desenvolvimento, houve uma busca pelo desenvolvimento completo de todos os requisitos priorizados como "Must" e, se possível, dos marcados como "Should" pelo método MoSCoW, definidos na Seção 3.2.1.

4.1.1 Requisitos Gerais

Dentre os requisitos gerais levantados, os seguintes foram desenvolvidos:

- **RNF01 - Autenticação de Usuário:** descrito na Seção 4.5.1, este requisito era essencial para controlar o acesso externo ao sistema.
- **RNF02 - Controle de Acesso:** descrito na Seção 4.5.6, este requisito era essencial para controlar o acesso interno do sistema, separado por papéis de usuário.

4.1.2 Requisitos de Administrador

Dentre os requisitos de administrador, os seguintes foram desenvolvidos:

- **RF03 - Gerenciar categorias:** descrito na Seção 4.5.6, este requisito era necessário para uma melhor separação e organização de conteúdo no sistema.
- **RF05 - Gerenciar Tutoriais:** descrito na Seção 4.5.5, este requisito era essencial para controlar o acesso interno do sistema, separado por papéis de usuário.
- **RF06 - Detalhamento de tutoriais:** descrito na Seção 4.5.4, este requisito permitiu que um tópico complexo fosse dividido em etapas ordenadas, facilitando o aprendizado progressivo.
- **RF07 - Gerenciar empresas:** descrito na Seção 4.5.3, este requisito era necessário para que os escritórios possam gerenciar informações de todas as suas empresas

dentro do sistema, atribuindo à elas conteúdos específicos e particulares que podem ser útil no ensino de novos colaboradores.

- **RF08 - Associar conteúdo à empresa:** desenvolvido durante a criação da funcionalidade descrita na Seção 4.5.4, este requisito foi essencial para permitir a personalização do ensino, vinculando procedimentos e documentos particulares a cada cliente.

4.1.3 Requisitos de Usuário

Dentre os requisitos de usuário, os seguintes foram desenvolvidos:

- **RF12 - Visualizar tutoriais:** descrito na Seção 4.5.5, este requisito era necessário para que o novo colaborador possa ter acesso ao conteúdo de tutoriais transmitido pelos administradores.
- **RF13 - Pesquisar tutoriais:** descrito na Seção 4.5.8, este requisito era necessário para que o novo colaborador possa ter acesso rápido e facilitado ao conteúdo que ele precisa aprender.
- **RF15 - Consultar empresas:** descrito na Seção 4.5.3, este requisito era necessário para que o novo colaborador possa ter acesso ao conteúdo específico de cada empresa, transmitido pelos administradores.

4.2 Configuração do Ambiente

A etapa de preparação do ambiente consistiu na instalação do Framework Laravel, juntamente com o `Inertia.js`, uma biblioteca que possibilita a integração entre Laravel e frameworks front-end modernos. Ele será utilizado em conjunto com o `Vue.js`, um framework progressivo em JavaScript. Além disso, nesta etapa foi realizada a configuração da conexão com um banco de dados local, a fim de fornecer suporte ao desenvolvimento.

4.3 Integração Contínua e Entrega Contínua (CI/CD)

Esta tarefa consistiu na estruturação de um *pipeline* de *Continuous Integration* (CI), com o objetivo de automatizar os processos de integração de código. A cada implementação de uma nova funcionalidade, o *pipeline* foi responsável por criar automaticamente um ambiente de desenvolvimento para a execução de testes, garantindo maior confiabilidade no fluxo de desenvolvimento.

Inicialmente, planejava-se também a configuração do *Continuous Delivery* (CD), que automatizaria a entrega de software em ambiente de produção. No entanto, essa etapa envolvia custos financeiros. Para viabilizar o processo, tentou-se ativar o *GitHub Student Developer*

Pack, que oferecia um crédito de US\$200 na *Digital Ocean*, mas a solicitação foi negada. Em seguida, buscou-se utilizar o crédito de US\$300 do *Google Cloud* para novos clientes; contudo, ao inserir as informações de pagamento, ocorreu o seguinte erro: “Não foi possível concluir a ação. [OR-BACR2-44]”. Diante dessas limitações, o desenvolvimento do CD foi postergado para uma próxima etapa, que será descrita a seguir.

4.4 Infraestrutura de Produção e Deploy

O objetivo desta tarefa foi estabelecer um fluxo de trabalho profissional e automatizado para os processos de teste e *deploy* da aplicação “*Daily Contábil*”, utilizando uma infraestrutura de nuvem moderna e robusta na plataforma da *Digital Ocean*.

A primeira fase consistiu na criação da infraestrutura de produção. Optou-se por uma arquitetura desacoplada, visando maior segurança e escalabilidade, com a utilização de dois serviços principais da *Digital Ocean*: um *Droplet* (servidor virtual privado) para hospedar a aplicação *Laravel* e um Banco de Dados Gerenciado (*Managed MySQL Database*) para armazenar os dados de forma segura e confiável, com backups automáticos. Adicionalmente, foi configurado o domínio *dailycontabil.com.br*, incluindo o apontamento de DNS e a instalação de um certificado SSL via *Let’s Encrypt*.

Com a infraestrutura concluída, a segunda fase concentrou-se na edição do *pipeline* de CI/CD já existente, a fim de inserir o *job* de *deploy* automatizado.

Os principais desafios dessa tarefa estiveram relacionados às permissões de arquivos no servidor e à correção de dependências no ambiente virtual.

4.5 Implementação das Funcionalidades

Com a infraestrutura de produção devidamente configurada e os processos de integração e entrega contínua estabelecidos, iniciou-se a fase de implementação das funcionalidades do sistema. Esta etapa teve como foco o desenvolvimento incremental das principais *features*, organizadas em tarefas gerenciadas por meio da ferramenta ClickUp. Cada *feature* representou uma unidade funcional do produto, planejada e desenvolvida de forma independente, permitindo o controle de progresso, a estimativa de esforço e a rastreabilidade em relação aos requisitos levantados.

O processo de desenvolvimento seguiu uma abordagem iterativa, com ciclos curtos de entrega e revisão, garantindo que cada incremento fosse devidamente testado e integrado ao ambiente principal do projeto. Para cada tarefa, foram documentadas as decisões técnicas adotadas, as dificuldades encontradas e as soluções implementadas, de modo a registrar a evolução do sistema e o aprendizado obtido ao longo do processo.

As subseções a seguir detalham as principais *features* implementadas, abordando os aspectos técnicos de sua construção, os desafios enfrentados e os resultados obtidos em cada etapa, contemplando desde a autenticação e a estrutura de navegação até a implementação de telas específicas e componentes centrais da aplicação.

4.5.1 Autenticação de usuário

Esta tarefa consistiu na estruturação de uma página de login utilizando o *Laravel Breeze*, uma implementação oficial e simplificada das funcionalidades de autenticação do framework *Laravel* (LARAVEL, 2025b). O *Breeze* fornece uma base pronta para operações essenciais, como login, registro e confirmação de e-mail, entre outras.

A adoção do Breeze possibilitou que *controllers*, *models*, rotas e toda a estrutura de *back-end* necessária fossem automaticamente configurados, demandando apenas ajustes visuais para atender às necessidades específicas do projeto.

A principal dificuldade encontrada esteve relacionada à implementação dos testes de *feature*. Por padrão, o Laravel recomenda o uso do banco de dados SQLite em memória para execução de testes automatizados, configurado por meio do arquivo `phpunit.xml` (PHPU-NIT, 2025). No entanto, devido a incompatibilidades observadas com dependências do projeto, optou-se por configurar um banco de dados local dedicado exclusivamente à execução dos testes, garantindo estabilidade e reprodutibilidade no processo de validação do código.

4.5.2 Menus de Navegação

Nesta etapa, foram desenvolvidos os principais componentes de layout da aplicação: o *header* (cabeçalho) e a *sidebar* (barra de navegação lateral).

O *header*, posicionado horizontalmente no topo da interface, foi projetado para conter os elementos globais da plataforma. Conforme ilustrado na Figura 10, foram implementados a logo da marca, um campo de busca para acesso rápido aos tutoriais e o menu de perfil do usuário, que futuramente abrigará as opções de *Configurações*.

A *sidebar* vertical, também ilustrada na Figura 10, por sua vez, funciona como o principal centro de navegação do sistema. Utilizando o componente `<Link>` do *Inertia.js* para garantir transições rápidas entre as páginas sem recarregamento, foram implementados os links para as seções já planejadas do sistema: *Dashboard*, *Tutoriais* e *Empresas*.

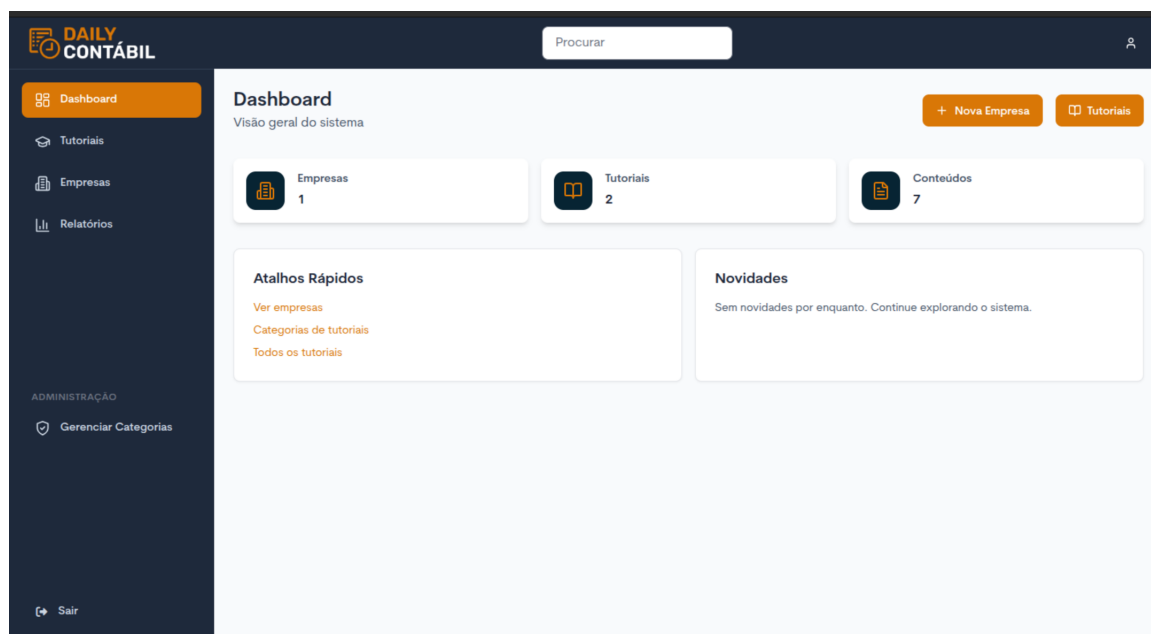


Figura 10 – Interface do sistema contendo o menu superior (*header*) e o menu lateral (*sidebar*).

Durante a fase de implementação, foi identificada uma dificuldade de alinhamento estético com a logo inicialmente proposta, que não se adaptava adequadamente ao espaço do *header*. Como solução, realizou-se um processo iterativo de design que resultou na criação de uma nova logo, agora devidamente integrada e alinhada à identidade visual do projeto.

4.5.3 Tela de Empresas

Esta tarefa consistiu na implementação da tela de listagem de empresas e visualização de uma empresa, possibilitando aos usuários efetuarem cadastros de empresas e visualizar informações relacionadas à elas. A maior dificuldade aqui se deu em relação ao encontro de um design moderno e consistente para a exibição da tabela de empresas, visto que o modelo apresentado no protótipo estava defasado. Essa tarefa se estendeu por 18 horas no total, em vista da complexidade dos componentes visuais envolvidos, como tabela dinâmica e preparação para exibição de conteúdo.

4.5.4 Biblioteca de Conteúdos

Esta foi uma das tarefas centrais e de maior complexidade do projeto, consistindo na criação de um sistema de gerenciamento de mídias (a "Biblioteca de Conteúdos") que fosse ao mesmo tempo centralizado e reutilizável. O trabalho foi dividido em três frentes principais:

1. **Arquitetura de Backend:** O maior desafio foi projetar uma arquitetura de dados que permitisse a um Content (vídeo, PDF, etc.) ser anexado a diferentes tipos de entidades (como Companies ou Tutorials). A solução foi a implementação de uma Relação

Polimórfica de Muitos-para-Muitos. Utilizando uma tabela intermediária contentables, o modelo Content pode se relacionar com qualquer outro modelo do sistema, garantindo que o mesmo conteúdo possa ser reutilizado em múltiplos contextos, sem duplicação de dados.

2. **Integração com Infraestrutura de Nuvem (*Amazon Web Services (AWS) S3*):** Para garantir a escalabilidade e a performance, o sistema de armazenamento de arquivos foi desacoplado da aplicação. Por meio do Laravel Storage, foi configurada a integração direta com um bucket S3 da Amazon Web Services. Toda a lógica de upload no ContentController foi escrita para enviar os arquivos diretamente para o S3, retornando a URL pública para ser salva no banco de dados.
3. **Interface do Usuário (UI/UX Dinâmica):** No frontend, a solução foi materializada em um componente Vue.js reutilizável, o ContentManagerModal.vue. Este modal expõe a lógica de "selecionar ou enviar" através de duas abas:
 - **"Biblioteca":** Lista os conteúdos existentes no banco de dados, com filtros e busca.
 - **"Fazer Upload":** Permite o envio de novos arquivos. A complexidade aqui foi criar uma experiência de usuário rica, com suporte a "arrastar e soltar" (drag-and-drop), feedback visual de progresso com barras de carregamento individuais, e a exibição de ícones dinâmicos de acordo com o tipo de arquivo.

O tempo total de 10 horas de desenvolvimento reflete a complexidade de se projetar a relação polimórfica, configurar a API de upload para o S3 e desenvolver o componente de modal dinâmico no Vue.js, que já foi integrado e testado na tela de gerenciamento de Empresas.

4.5.5 Tela de Tutoriais

O desenvolvimento da tela de tutoriais foi uma das etapas mais importantes, e exigiu uma revisão do que havia sido planejado inicialmente. O plano original era que o usuário escolhesse uma categoria (ex: "Fiscal") e depois uma "subcategoria" (ex: "ICMS"), para só então ver os tutoriais. Percebi que isso traria uma complexidade desnecessária, pois "categoria" e "subcategoria" podiam ser facilmente confundidos.

A solução foi simplificar o fluxo: o usuário agora acessa "Tutoriais", escolhe uma Categoria e vê a lista de Tutoriais diretamente. Para compensar, o conceito de Tutorial ficou mais robusto: em vez de ser uma única instrução, ele agora possui "Etapas". Isso permite que um assunto complexo seja dividido em um passo a passo organizado, com o usuário podendo acompanhar seu progresso em cada etapa.

Um dos desafios aqui foi como lidar com as categorias padrão. Eu queria que todo novo escritório já começasse com 5 categorias pré-cadastradas (Contábil, Fiscal, etc.), mas

que o usuário pudesse gerenciá-las. A primeira tentativa foi usar um *Seeder* (um "populador" de banco de dados). O problema é que o *Seeder* roda de forma global e não sabia a qual escritório (office_id ou *tenant*) as categorias pertenciam. Isso criava registros "órfãos" que o usuário não podia ver ou excluir, o que ia contra os requisitos.

A solução encontrada foi usar um *Observer* do Laravel. Basicamente, é um "vigia" que foi programado para "assistir" a criação de novos escritórios. Assim que um *super-admin* cadastra um novo escritório, o *Observer* entra em ação e, nesse exato momento, cria as 5 categorias padrão, já associando elas corretamente ao ID daquele novo escritório.

Por fim, esta tela integrou a "Biblioteca de Conteúdos" de duas formas:

1. Cada Etapa do tutorial foi conectada a um conteúdo principal (como um vídeo ou PDF) através de uma relação direta.
2. Os "Materiais Complementares" do tutorial (visíveis ao final da página) usaram a relação polimórfica que já havíamos criado, permitindo anexar múltiplos arquivos reutilizáveis ao tutorial como um todo.

4.5.6 Gerenciamento de Categorias

Esta tarefa consistiu na implementação do *Create, Read, Update and Delete* (CRUD) (Create, Read, Update, Delete) completo para o recurso "Categorias". O desenvolvimento foi focado em três desafios técnicos principais: controle de acesso, isolamento de dados e personalização da interface.

1. **Controle de Acesso Baseado em Papel:** Conforme o requisito de que apenas usuários administrativos pudessem gerenciar categorias, foi implementada uma nova seção na barra de navegação lateral, intitulada "Administração", como ilustrado na Figura 10. O acesso a esta seção foi restrito programaticamente. No *front-end*, o link só é renderizado se o usuário logado possuir a permissão de admin. No *back-end*, todas as rotas do *CategoryController* são protegidas por um *middleware* do Laravel, garantindo que apenas usuários autorizados possam executar ações de criação, edição ou exclusão, bloqueando qualquer tentativa de acesso direto.
2. **Interface e UX Dinâmica:** Um desafio de UX foi manter a interface visualmente atraente, mesmo permitindo que usuários criassem suas próprias categorias. A solução foi estender a tabela *categories* para incluir os campos *icon_name* e *icon_color*. Foi desenvolvido um componente *IconPicker.vue* que permite ao administrador, durante o cadastro ou edição de uma categoria, selecionar um ícone e uma cor de uma paleta pré-definida. Isso garante a consistência visual da plataforma, ao mesmo tempo que dá ao usuário o controle e a personalização desejados.

4.5.7 *Landing Page*

Esta tarefa consistiu na implementação da *landing page* (página de aterrissagem) do sistema, que atua como a principal vitrine B2B e ferramenta de captação de clientes para o "Daily Contábil".

O desafio foi projetar uma página que comunicasse o valor de um produto SaaS para um público-alvo específico (donos de escritórios contábeis). Para isso, a página foi estruturada como uma narrativa de marketing persuasiva, com o objetivo de guiar o visitante por um funil de vendas lógico, desde a identificação do problema até a conversão para uma *Call to Action* (CTA).

A estrutura da página foi dividida nas seguintes seções estratégicas:

- **Seção Hero:** Conforme ilustrado na figura 11 seção principal, que captura a atenção com uma proposta de valor clara — "Transforme o conhecimento do seu escritório em um ativo permanente"— e um CTA principal para "Agendar uma Demonstração".



Figura 11 – Seção hero contendo proposta de valor e CTA.

- **Seção de Dores:** Logo abaixo, a página se conecta com o visitante abordando os problemas centrais do público, como "Horas de sênior perdido", "Treinamento demorado" e "Conhecimento perdido", validando a necessidade da solução.
- **Apresentação da Solução:** Em seções de contraste (fundo escuro), a página apresenta as *features* do "Daily Contábil" ("Tutoriais por passos", "Biblioteca e categorias") como a solução direta para as dores apresentadas.
- **Seção de Benefícios:** Foca no retorno sobre o investimento e nos resultados práticos para o gestor, como "Onboarding mais rápido", "Execução padronizada" e "Conhecimento preservado".
- **Seção de Prova Social:** Utiliza depoimentos ("O que dizem") e logos de empresas (placeholder) para gerar confiança e autoridade, um elemento crucial para a tomada de decisão em um contexto *Business to Business* (B2B).

4.5.8 Filtro e Pesquisa

Para atender ao requisito de "acesso rápido e organizado" à informação, foi implementada uma funcionalidade de pesquisa e filtragem dinâmica nas principais telas de listagem do sistema (como "Empresas" e "Tutoriais"). O principal desafio técnico foi criar uma experiência de usuário fluida e instantânea, que não dependesse de recarregamentos completos da página.

5 CONCLUSÃO

Diante da elevada carga de trabalho dos escritórios contábeis, da dificuldade no processo de aprendizagem dos profissionais da área e das constantes alterações na legislação brasileira, torna-se essencial a presença de colaboradores devidamente capacitados para lidar com as demandas da rotina contábil. A ausência de profissionais qualificados não apenas sobrecarrega os colaboradores mais experientes, como também compromete o cumprimento dos prazos e a entrega das obrigações exigidas pelos órgãos reguladores.

5.1 Considerações Finais

Diante do cenário apresentado, o desenvolvimento do "Daily Contábil" foi proposto como uma solução direta para os desafios de treinamento e gestão de conhecimento nos escritórios de contabilidade. O objetivo central foi criar uma ferramenta que permitisse otimizar a rotina de quem ensina e de quem aprende, facilitando a transmissão do conhecimento prático e operacional.

Para atingir esse objetivo, o projeto partiu de um levantamento de requisitos com profissionais da área, resultando na implementação de uma aplicação web *full-stack* funcional. O trabalho concluído nesta etapa estabeleceu as fundações do sistema, que incluem:

- **Infraestrutura de Produção:** Configuração de um ambiente de nuvem na *Digital Ocean*, com *Droplet* para a aplicação e Banco de Dados Gerenciado.
- **Automação de Deploy (CI/CD):** Estruturação de um *pipeline* no *GitHub Actions* para testes automatizados e entrega contínua.
- **Autenticação e Navegação:** Implementação do sistema de *login* (via *Laravel Breeze*) e dos componentes centrais de navegação, como o *header* e a *sidebar*.
- **Módulos de Negócio (CRUD):** Desenvolvimento das funcionalidades de gerenciamento para Empresas, Categorias e Tutoriais.
- **Arquitetura de Conteúdo:** A implementação da "Biblioteca de Conteúdos", uma *feature* complexa que utiliza relações polimórficas para anexar arquivos (via *AWS S3*) a diferentes módulos do sistema.

Embora os objetivos principais tenham sido alcançados com êxito, o processo de desenvolvimento enfrentou desafios técnicos e de gestão de escopo que exigiram adaptações na estratégia inicial.

Um dos principais obstáculos esteve relacionado à infraestrutura e custos de nuvem. O planejamento inicial previa a implementação de um fluxo completo de *Continuous Delivery*

(CD). No entanto, dificuldades na obtenção de créditos educacionais em provedores de nuvem e limitações orçamentárias inviabilizaram a automação total do *deploy* neste ciclo. Como solução, priorizou-se a estabilidade do ambiente de produção e a automação dos testes (CI), mantendo o processo de entrega semi-automatizado, mas funcional.

Houve também desafios relacionados à definição de requisitos e UX. Durante a implementação do módulo de tutoriais, percebeu-se que a estrutura de "Categorias e Subcategorias" originalmente planejada geraria uma navegação confusa. Foi necessária uma refatoração do escopo para simplificar a hierarquia, optando-se por um modelo mais direto de "Categorias e Tutoriais com Etapas". Essa decisão, embora tenha consumido tempo de desenvolvimento, resultou em uma experiência de usuário mais fluida e intuitiva.

5.2 Trabalhos Futuros

Como perspectiva futura, pretende-se ampliar as funcionalidades do *Daily Contábil*, incorporando recursos de acompanhamento de desempenho dos usuários, gamificação do processo de aprendizagem e integração com plataformas externas de gestão contábil. Além disso, estudos adicionais poderão avaliar a eficácia da ferramenta em diferentes contextos empresariais, verificando seu impacto na produtividade e na redução de erros operacionais. Dessa forma, espera-se que o sistema evolua continuamente, consolidando-se como uma ferramenta de apoio essencial para o aprimoramento profissional e a modernização dos escritórios contábeis.

REFERÊNCIAS

AGENDA CONTÁBIL. **Desafios do Setor Contábil: como superá-los?** 2024. Disponível em: <https://www.agendacontabil.com.br/artigo.php?noticia=10158>. Acesso em: 04 mai. 2025.

APARECIDO, A. d. J. **CONTABILIDADE 4.0: OS DESAFIOS PARA PROFISSIONAIS CONTÁBEIS**. 2023. Disponível em: <https://periodicorease.pro.br/rease/article/download/7031/2739/10401>. Acesso em: 04 mai. 2025.

ATLASSIAN. **O que é Kanban?** 2025. Disponível em: <https://www.atlassian.com/agile/kanban>. Acesso em: 09 jul. 2025.

ATTEMPO. **Entenda os riscos da negligência contábil e como evitá-los**. 2024. Disponível em: <https://attempo.com.br/entenda-os-riscos-da-negligencia-contabil-e-como-evita-los/>. Acesso em: 05 mai. 2025.

BRASIL. **Lei nº 10.406, de 10 de janeiro de 2002 — Institui o Código Civil**. 2002. Art. 1.179 - Escrituração contábil obrigatória. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/2002/l10406compilada.htm. Acesso em: 10 abr. 2025.

CLICKUP. **ClickUp™ - Project Management**. 2025. Disponível em: <https://clickup.com/>. Acesso em: 03 jun. 2025.

CONSELHO FEDERAL DE CONTABILIDADE. **DIA DO EMPRESÁRIO CONTÁBIL: A IMPORTÂNCIA DAS EMPRESAS DE CONTABILIDADE PARA O CRESCIMENTO DOS NEGÓCIOS NO PAÍS**. 2024. Disponível em: <https://cfc.org.br/noticias/dia-do-empresario-contabil-a-importancia-das-empresas-de-contabilidade-para-o-crescimento-dos-negocios>. Acesso em: 04 mai. 2025.

CONTABEIS, P. **84,1 contratar profissionais qualificados**. 2022. Disponível em: <https://www.contabeis.com.br/noticias/54009/empresas-tem-mais-dificuldade-para-recrutar-do-que-antes-da-pandemia/>. Acesso em: 04 mai. 2025.

CONTABILIZEI. **Qual o objetivo da contabilidade?** 2024. Disponível em: <https://www.contabilizei.com.br/contabilidade-online/objetivo-da-contabilidade/>. Acesso em: 04 mai. 2025.

CONTADOR AGORA. **Contabilidade: o que é, para que serve e tipos**. 2021. Disponível em: <https://www.contadoragora.com/contabilidade/#:~:text=Agora%2C%20segundo%20o%20Oxford%20Languages,do%20patrim%C3%B4nio%20de%20uma%20empresa>. Acesso em: Acessado em: 4 maio 2025.

FAZENDA, M. da. **Perguntas e Respostas - Reforma Tributária**. 2024. Disponível em: https://www.gov.br/fazenda/pt-br/acesso-a-informacao/acoes-e-programas/reforma-tributaria/arquivos/perguntas-e-respostas-reforma-tributaria_.pdf. Acesso em: 05 mai. 2025.

GIT. **git –fast-version-control**. 2025. Disponível em: <https://git-scm.com/>. Acesso em: 03 jun. 2025.

GITHUB. **GitHub**. 2025. Disponível em: <https://github.com/>. Acesso em: 03 jun. 2025.

LARAVEL. **Laravel - The PHP Framework For Web Artisans**. 2025. Disponível em: <https://laravel.com/>. Acesso em: 03 jun. 2025.

LARAVEL. **Laravel Documentation**. 2025. Acesso em: 28 out. 2025. Disponível em: <https://laravel.com/docs>.

MYSQL. **MySQL**. 2025. Disponível em: <https://www.mysql.com/>. Acesso em: 03 jun. 2025.

OCEAN, D. **DigitalOcean | Cloud Infrastructure for Developers**. 2025. Disponível em: <https://www.digitalocean.com/>. Acesso em: 03 jun. 2025.

OXFORD LANGUAGES. **O que é contabilidade?** 2021. Acesso em: 10 abr. 2025. Disponível em: <https://www.contadoragora.com/contabilidade/>.

PHPUNIT. **PHPUnit Manual**. 2025. Acesso em: 28 out. 2025. Disponível em: <https://phpunit.de/documentation.html>.

RAILWAY. **Railway**. 2025. Disponível em: <https://railway.com/>. Acesso em: 03 jun. 2025.