

**CENTRO UNIVERSITÁRIO ALVES FARIA
(UNIALFA) CURSO DE ENGENHARIA DE
SOFTWARE**

**Kaue dos Anjos Silva
Luis Gustavo dos Passos Ribeiro
Marcos Paulo Ferraz da Silva
Weder Oliveira Rocha**

SISTEMA PARA GESTÃO FINANCEIRA - Finansec

Projeto de Extensão desenvolvido como requisito para aprovação nas disciplinas de Fábrica de Software I e Administração e organização de banco de dados, do curso de Engenharia de Software do Centro Universitário Alves Faria (UNIALFA)

GOIÂNIA

2025

**Kaue dos Anjos Silva
Luis Gustavo dos Passos Ribeiro
Marcos Paulo Ferraz da Silva
Weder Oliveira Rocha**

SISTEMA PARA GESTÃO FINANCEIRA - Finansec

Professor: Edivan Carvalho - Fábrica de Software I e Yhury Rezende -
Administração e organização de banco de dados

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema financeiro desktop, criado por quatro estudantes do curso de Engenharia de Software de uma universidade particular. O sistema foi desenvolvido em C# utilizando a biblioteca Windows Forms, com banco de dados SQL Server hospedado na plataforma Azure. O ambiente de desenvolvimento adotado foi o Visual Studio, e o gerenciamento do banco foi feito via SQL Server Management Studio. O projeto segue boas práticas de versionamento e organização utilizando o GitHub.

A aplicação possui cinco seções: menu, dashboard, financeiro, cadastros e conta. O menu exibe notificações sobre alterações no banco. O dashboard apresenta dados de entradas, saídas,

saldo, calendário com filtro de período e gráfico de projeção. A seção financeira organiza contas a pagar e receber, além do registro de receitas e despesas. A parte de cadastros permite criar, editar e consultar registros de pessoas físicas e jurídicas, com diversos filtros. A aba de conta permite editar informações do usuário, alterar senha, excluir conta e sair do sistema.

O sistema está sendo ampliado com uma API REST para comunicação com o banco de dados, o que facilitará futuras integrações com aplicações web e mobile. Seu uso é aplicável a pequenas e médias empresas que necessitam de controle financeiro mais preciso e centralizado. Entre os principais benefícios estão a automação de rotinas, a confiabilidade dos dados, o suporte à tomada de decisão e a escalabilidade em ambientes corporativos.

Palavras-chave: sistema financeiro. C#. SQL Server. WinForms. Azure. engenharia de software. gestão financeira.

Sumário

Sumário.....	3
Lista de figuras.....	4
1.1 Objetivos	6
1.1.1 Objetivos Geral	6
1.1.2 Objetivos Específicos	6
1.2 Justificativa	8
2 METODOLOGIA.....	9
2.1 Caracterização.....	9
2.2 Versionamento.....	11
2.2 variáveis implementadas.....	13
2.2.1 variáveis de Front-End.....	13
3. tecnologias e ferramentas utilizadas no desenvolvimento	15
3.1 Tecnologias e Ferramentas Utilizadas no Desenvolvimento	15

4. CONSIDERAÇÕES FINAIS	16
5. Referências	17

Lista de figuras

Diagrama de Caso de Uso.....	11
Diagrama Estrutura do Banco de Dado.....	11

1 INTRODUÇÃO

A gestão financeira é um dos elementos centrais para o funcionamento e a sustentabilidade de qualquer organização. No atual cenário empresarial, marcado pela digitalização de processos e pela necessidade de tomada de decisões rápidas e precisas, o uso de sistemas informatizados tornou-se essencial para garantir eficiência, controle e segurança na administração das finanças. Pequenas e médias empresas, em especial, enfrentam desafios significativos nesse campo, seja pela ausência de ferramentas apropriadas, seja pela utilização de sistemas obsoletos, limitados ou mal adaptados às suas realidades operacionais.

A linguagem C# é a linguagem mais popular para a plataforma .NET, um ambiente de desenvolvimento gratuito, multiplataforma e de código aberto. Programas em C# podem ser executados em diversos dispositivos, desde dispositivos da Internet das Coisas (IoT) até a nuvem e em qualquer lugar entre eles. Você pode criar aplicativos para celulares, desktops, laptops e servidores. (Microsoft, A tour of the c# language, 2025).

Neste contexto, desenvolveu-se o projeto de um sistema financeiro com o objetivo de centralizar informações essenciais, automatizar processos, auxiliar nos cálculos e oferecer suporte à análise de dados financeiros. A proposta visa atender empresas que ainda não utilizam soluções informatizadas ou que dependem de softwares ultrapassados, oferecendo

uma alternativa eficiente, moderna e acessível. A ideia central do sistema é promover a organização financeira, reduzir o retrabalho, facilitar a gestão e ampliar a compreensão dos fluxos de receitas e despesas por meio de recursos gráficos e relatórios dinâmicos.

A aplicação foi desenvolvida como um sistema desktop utilizando a linguagem C# e a biblioteca Windows Forms, ambas pertencentes ao ecossistema Microsoft. A escolha do C# justifica-se por sua alta eficiência na criação de aplicações para o sistema operacional Windows, que, em abril de 2025, detinha aproximadamente 71% do mercado mundial segundo o ranking do StatCounter Global Stats. Essa escolha estratégica também se deve à compatibilidade nativa entre o C# e o banco de dados relacional SQL Server, adotado no projeto por suas vantagens em desempenho otimizado, alta disponibilidade, segurança robusta, escalabilidade e familiaridade técnica por parte da equipe de desenvolvimento.

O sistema possui uma interface simples e intuitiva, organizada em módulos principais: Dashboard, Cadastros, Financeiro, Menu e Conta. A aba de dashboard apresenta os totais de entradas, saídas e saldo, além de um gráfico de projeção e um calendário para seleção de período.

O módulo financeiro é subdividido em quatro abas: contas a pagar/receber, contas pagas/recebidas, inclusão de entradas e inclusão de saídas. Já o módulo de cadastros permite o gerenciamento de pessoas físicas e jurídicas com campos específicos como CNPJ, razão social, cidade, telefone e e-mail, além de filtros para consulta. A aba de conta oferece ao usuário a possibilidade de editar seu perfil, alterar a senha, excluir a conta ou encerrar a sessão.

A metodologia empregada baseou-se em práticas ágeis e desenvolvimento incremental. O gerenciamento do projeto foi realizado por meio da plataforma GitHub, com versionamento contínuo, controle de tarefas e documentação colaborativa. O banco de dados foi modelado no SQL Server Management Studio (SSMS) e hospedado na nuvem utilizando os serviços do Microsoft Azure, garantindo acesso remoto, estabilidade e segurança. Uma API RESTful também está em desenvolvimento, com o intuito de permitir a futura integração da aplicação desktop com plataformas web e mobile, promovendo maior escalabilidade.

O problema de pesquisa está centrado na lacuna existente entre a demanda por soluções financeiras acessíveis e a limitada oferta de sistemas eficazes voltados a pequenas e médias empresas. O sistema proposto busca preencher essa lacuna ao oferecer uma solução que alia tecnologia moderna, facilidade de uso, segurança da informação e potencial de expansão, atendendo às reais necessidades do público-alvo de maneira prática e eficaz.

1.1 Objetivos

1.1.1 Objetivos Geral

O presente trabalho tem como objetivo geral desenvolver um sistema completo por meio de uma aplicação desktop de gestão financeira, utilizando a linguagem C# integrada ao banco de dados SQL Server no ambiente de desenvolvimento Visual Studio. A proposta visa fornecer uma solução prática e eficaz, que contemple o registro de entradas e saídas financeiras, o cadastro de entidades jurídicas e físicas, e a análise de dados por meio de uma interface gráfica com dashboard, com o intuito de aprimorar o controle das finanças empresariais, especialmente em pequenas e médias empresas.

A partir desse objetivo central, busca-se implementar funcionalidades que possibilitem a centralização de receitas, despesas, contas a pagar e a receber em um único ambiente digital. Além disso, pretende-se reduzir erros manuais por meio de automatizações estratégicas no processo de lançamento e controle financeiro. O sistema deverá também oferecer mecanismos que contribuam para a conciliação bancária com maior precisão e eficiência, bem como permitir a projeção de cenários futuros por meio de gráficos e indicadores visuais baseados em dados históricos. Outro objetivo relevante consiste em registrar e manter um histórico completo das transações realizadas, promovendo transparência, organização e rastreabilidade das informações financeiras.

Tecnologias digitais e avançadas, como a nuvem, estão “mudando rápida e fundamentalmente o que é possível para as empresas e como elas competem”, explica um artigo recente da Deloitte Global. “Inovações — frequentemente impulsionadas pelas grandes plataformas de nuvem —

Por meio desses objetivos, o projeto visa entregar uma ferramenta funcional, acessível e escalável, que possa atender às necessidades operacionais das empresas em suas rotinas financeiras, promovendo não apenas agilidade e segurança, mas também suporte à tomada de decisões estratégicas fundamentadas em dados confiáveis em nuvem.

1.1.2 Objetivos Específicos

- Criar uma interface com abas funcionais: Menu, Dashboard, Financeiro, Cadastros e Conta, proporcionando uma navegação intuitiva para o usuário.
- Permitir o cadastro, edição e exclusão de registros de entidades, incluindo clientes e fornecedores, com distinção entre pessoa física e jurídica.

- Implementar o controle de movimentações financeiras por meio do registro de contas a pagar, contas a receber, contas pagas e recebidas.
- Automatizar o cálculo de saldos totais, bem como projeções financeiras com base nas transações registradas.
- Disponibilizar um dashboard com gráficos e indicadores visuais que representem entradas, saídas, saldo e projeções financeiras.
- Armazenar todas as informações em um banco de dados relacional seguro, estruturado e hospedado em ambiente em nuvem.
- Implementar um sistema de notificações que informe o usuário sobre alterações e eventos importantes no banco de dados.
- Fornecer opções para alteração de perfil do usuário, como troca de senha, edição de dados da conta e encerramento de sessão.
- Reduzir erros manuais por meio de validações e processos automatizados no registro e edição de dados financeiros.
- Registrar o histórico completo das transações, permitindo consultas e análises retrospectivas para fins de auditoria.
- Desenvolver uma API RESTful para integração futura com sistemas web ou mobile, ampliando a escalabilidade do projeto.

1.2 Justificativa

A gestão financeira eficaz é um dos fatores determinantes para a sobrevivência e o crescimento de qualquer organização. Contudo, observa-se que grande parte das pequenas e médias empresas ainda opera sem sistemas adequados para controle de receitas, despesas e cadastros de clientes e fornecedores, o que compromete sua eficiência operacional e aumenta a exposição a falhas humanas. Com frequência, essas empresas recorrem ao uso de planilhas eletrônicas ou métodos manuais para gerenciamento financeiro, que embora úteis em estágios iniciais, tornam-se rapidamente limitados frente ao aumento da complexidade das operações.

De acordo com Akinola S.A - Vencedor do Prêmio de Realização Excepcional em Pesquisa e Cidadania Acadêmica, 2023 na UJ - “A gestão financeira é um fator crucial para o sucesso das PMEs. Ao enfrentar desafios e implementar práticas de excelência, as PMEs podem otimizar seu desempenho financeiro, tomar decisões informadas e se preparar para o crescimento e a prosperidade a longo prazo”.

Diante desse cenário, o desenvolvimento de um sistema financeiro específico para este perfil empresarial mostra-se altamente relevante. A proposta de construção de uma aplicação desktop utilizando C# e SQL Server responde diretamente a essa demanda, oferecendo uma solução moderna, segura e personalizável. A linguagem C# é amplamente utilizada em ambientes corporativos e proporciona robustez, desempenho e integração eficiente com o sistema operacional Windows, que atualmente representa a maior fatia do mercado de desktops. Aliada a isso, a escolha do banco de dados SQL Server permite o armazenamento seguro e estruturado das informações, além de facilitar a manutenção, a escalabilidade e o acesso remoto quando hospedado em plataformas como o Microsoft Azure.

O sistema em desenvolvimento busca não apenas informatizar processos, mas também automatizar tarefas repetitivas, reduzir o retrabalho e oferecer suporte à análise estratégica por meio de recursos como dashboards interativos e relatórios financeiros. Trata-se, portanto, de uma solução que atende às necessidades práticas de empresas de pequeno e médio porte, contribuindo para a profissionalização da gestão, a padronização das rotinas financeiras e a tomada de decisões mais assertivas. Sua adoção pode representar um avanço significativo na organização interna, no controle de fluxo de caixa e na projeção de cenários futuros, gerando valor direto ao negócio.

2 METODOLOGIA

2.1 Caracterização

A metodologia adotada para o desenvolvimento do sistema financeiro seguiu uma abordagem incremental e iterativa, com base em práticas da engenharia de software e elementos de metodologias ágeis, principalmente no que diz respeito à divisão do projeto em ciclos de entrega contínua. Essa escolha visou garantir maior controle sobre o progresso das funcionalidades, facilitando ajustes e validações constantes com os objetivos previamente definidos.

O processo iniciou-se com a análise dos requisitos funcionais e não funcionais, baseando-se nas principais demandas enfrentadas por pequenas e médias empresas no que se refere à gestão financeira. A modelagem conceitual e lógica do banco de dados foi elaborada por meio do SQL Server Management Studio (SSMS), utilizando a linguagem SQL para estruturação de tabelas, chaves primárias e estrangeiras, índices e procedimentos armazenados. O banco de dados foi posteriormente hospedado na plataforma Microsoft Azure, permitindo acesso remoto seguro e escalável.

A aplicação foi desenvolvida na linguagem C#, utilizando a biblioteca gráfica Windows Forms dentro do ambiente de desenvolvimento Visual Studio. Esta combinação foi escolhida pela sua compatibilidade com o sistema operacional Windows, sua ampla documentação e integração nativa com o SQL Server. A interface do sistema foi projetada com foco em usabilidade, utilizando elementos gráficos claros e organizados por meio de um menu de navegação por abas: Menu, Dashboard, Financeiro, Cadastros e Conta.

Para a Microsoft Azure — *Azure SQL Database benefits*. O Banco de Dados SQL do Azure fornece um serviço de banco de dados relacional totalmente gerenciado, escalável e seguro de que ajuda as organizações a reduzir o custo e a complexidade do gerenciamento de bancos de dados, garantindo alta disponibilidade e conformidade. Conforme o desenvolvimento em ambiente Microsoft, foi comprovado tais características.

Durante o desenvolvimento, o controle de versão foi gerenciado por meio da plataforma GitHub, onde foram realizados commits frequentes, gerenciamento de tarefas, registros de issues e integração colaborativa entre os membros da equipe. As funcionalidades foram implementadas gradualmente, começando pelas operações de cadastro de entidades, em seguida o controle de movimentações financeiras, a geração de relatórios e, por fim, os módulos de segurança e configurações do usuário.

A validação das funcionalidades foi realizada por meio de testes manuais, simulando casos reais de uso, como cadastros, lançamentos financeiros e geração de relatórios. Além disso, foram aplicados testes de consistência no banco de dados, para garantir a integridade das informações. O sistema foi projetado para permitir expansões futuras, com foco na modularidade do código e separação clara entre camadas de apresentação, lógica de negócio e acesso a dados.

Imagem 1 – Diagrama de Caso de Uso

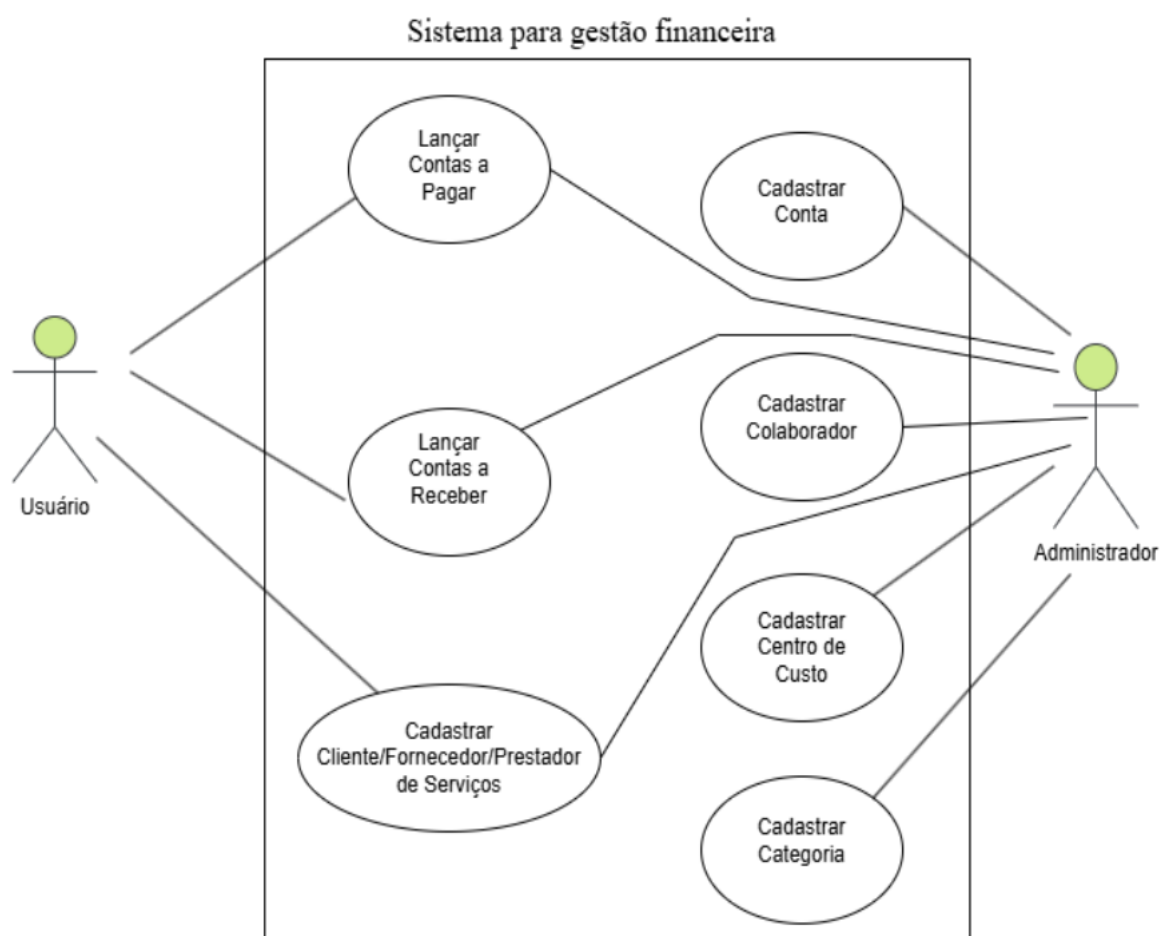
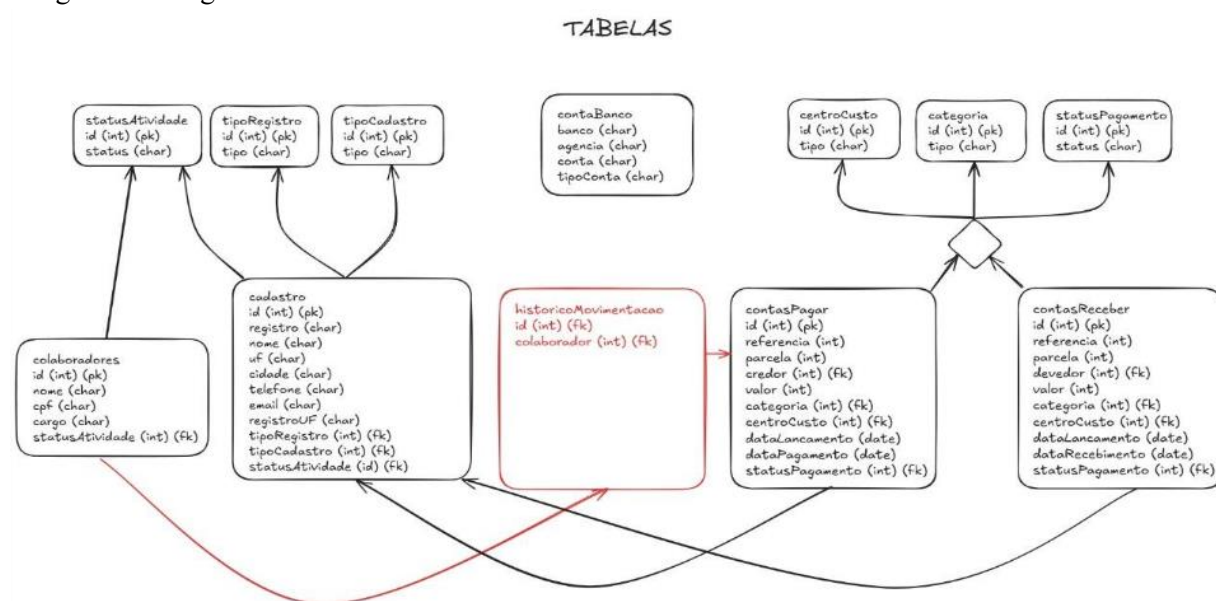


Imagem 2 – Diagrama Estrutura do Banco de Dados



2.2 Versionamento

Versão 1.0 (17/03/2025): Estrutura inicial do projeto foi configurada, incluindo o repositório e ambiente de desenvolvimento.

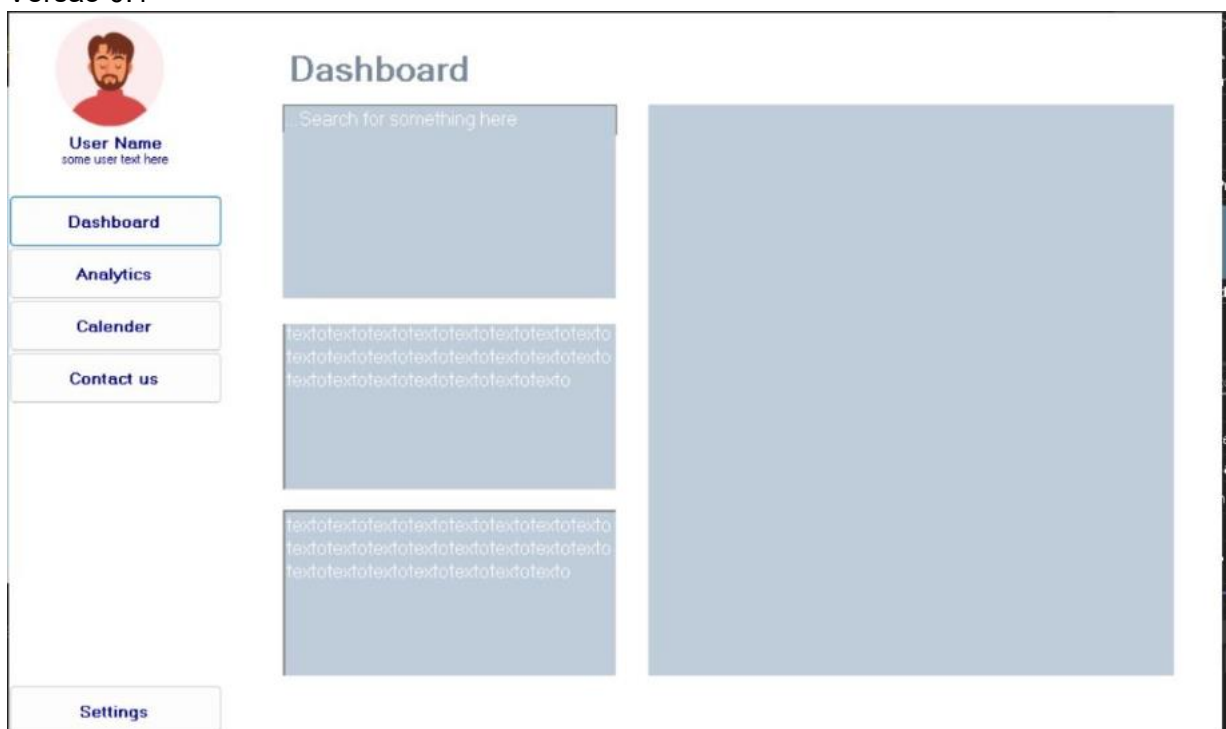
Versão 0.1 (20/03/2025): Protótipo inicial com formulários e tela inicial criados utilizando WinForms.

Versão 0.2 (28/03/2025): Integração da dependência FontAwesome para padronizar o estilo e melhorar a interface gráfica.

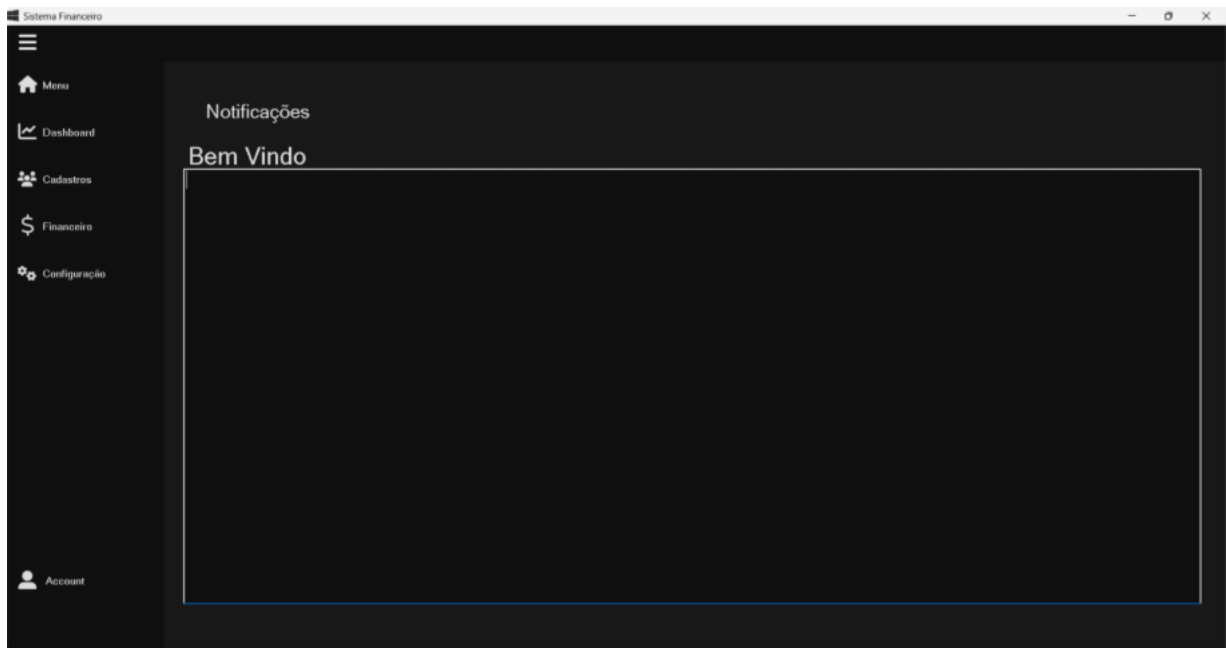
Versão 0.3 (03/04/2025): Desenvolvimento das abas principais com funcionalidade de redirecionamento entre elas e os primeiros testes de integração com o banco de dados.

Versão 1.0 (22/05/2025): Banco de dados totalmente configurado e integrado ao aplicativo. A dependência LiveCharts 0.9.7 foi adicionada para permitir a visualização de gráficos financeiros no dashboard. Também foi realizada uma atualização no menu retrátil do frontend, melhorando a navegação do usuário.

Versão 0.1



Versão 1



2.2 variáveis implementadas

Usamos diversas variáveis no código referentes tanto a parte de interface/front-end quanto a interação com o banco de dados/back-end

2.2.1 variáveis de Front-End

Esses manipuladores de eventos permitem que a interface do usuário se torne interativa. Ao clicar em diferentes botões no menu, o usuário pode navegar entre diferentes seções do aplicativo, como Dashboard, Cadastros, Financeiro, Configurações e Logout. O estilo visual do botão que foi clicado também é alterado para dar um feedback visual ao usuário, indicando qual seção está ativa.

```

102 private void BtnDashboard_Click(object sender, EventArgs e)
103 {
104     ActivateButton(sender, RGBColors.highlight);
105     openChildForm(new DashboardForm());
106 }
107 private void BtnCadastrados_Click(object sender, EventArgs e)
108 {
109     ActivateButton(sender, RGBColors.highlight);
110     openChildForm(new CadastrosForm());
111 }
112 private void BtnFinanceiro_Click(object sender, EventArgs e)
113 {
114     ActivateButton(sender, RGBColors.highlight);
115     openChildForm(new FinanceiroForm());
116 }
117 private void BtnConfiguracoes_Click(object sender, EventArgs e)
118 {
119     ActivateButton(sender, RGBColors.highlight);
120     openChildForm(new ConfiguracoesForm());
121 }
122 private void BtnLogout_Click(object sender, EventArgs e)
123 {
124     ActivateButton(sender, RGBColors.highlight);
125     openChildForm(new accountForm());
126 }
127

```

2.2.2 variáveis de Back-End

A variável `sqlQuery` é uma **string** que contém um comando **SQL** de inserção de dados em um banco de dados **SQL Server**. Ela é construída dinamicamente usando interpolação de string (com o prefixo `$`), onde os valores de variáveis são inseridos diretamente dentro da consulta. Vamos analisar o que ocorre passo a passo.

```

225 var sqlQuery = $"INSERT INTO {tabelaDestino} " +
226     "(registro, nome, uf, cidade, telefone, email, registroUF, tipoRegistro, tipoCadastro, statusAtividade) " +
227     $"VALUES ('{registro}', '{nome}', '{uf}', '{cidade}', '{telefone}', '{email}', '{registroUF}', '{tipoRegistro}', " +
228     $"{tipoCadastro}', " + $"'{statusAtividade}');"
229
230 SqlCommand cmd = new SqlCommand(sqlQuery, cn);
231 int resultado = cmd.ExecuteNonQuery();
232
233 if (resultado > 0)
234 {
235     MessageBox.Show("Cadastro realizado com sucesso!", "Sucesso!", MessageBoxButtons.OK, MessageBoxIcon.Information);
236 }
237 else
238 {
239     MessageBox.Show("Nenhuma linha foi inserida.");
240 }
241 cn.Close(); // boa prática: fechar mesmo que o GC vá cuidar disso depois
242
243 catch (Exception ex)
244 {
245     MessageBox.Show("Falha ao tentar conectar ou salvar\n\n" + ex.Message);
246 }
247

```

3. tecnologias e ferramentas utilizadas no desenvolvimento

Pretendemos criar um software que se encaixe na realidade das empresas faremos testes de implementação em diversos ambientes e analisaremos as respostas.

3.1 Tecnologias e Ferramentas Utilizadas no Desenvolvimento

Para a implementação do sistema, adotou-se a linguagem de programação C#, amplamente reconhecida pela sua robustez e compatibilidade com o ambiente Windows. O desenvolvimento foi realizado no Visual Studio, ambiente integrado que proporciona recursos avançados para codificação, depuração e testes.

A aplicação foi concebida com um design responsivo, assegurando uma experiência fluida e adaptável a diferentes tamanhos de tela. Essa estratégia visa garantir acessibilidade tanto em desktops quanto em dispositivos móveis, ampliando o alcance e a flexibilidade de uso do sistema.

O banco de dados relacional utilizado foi o SQL Server, hospedado na plataforma Microsoft Azure, o que proporciona alta disponibilidade, segurança e escalabilidade. Para facilitar a comunicação entre a aplicação e o banco, está sendo desenvolvida uma API RESTful, que segue os padrões modernos de integração entre sistemas

Abaixo, no Quadro 1, apresentam-se as principais ferramentas e tecnologias selecionadas para o desenvolvimento, detalhando suas funções e justificativas para a escolha.

Quadro 1 – Resumo das ferramentas e tecnologias utilizadas no desenvolvimento

FERRAMENTAS / TECNOLOGIAS	DESCRIÇÃO	JUSTIFICATIVA PARA USO
C#	Linguagem de desenvolvimento para aplicações de desktop. O design será responsivo, adaptando-se a diferentes tamanhos de tela.	Essencial para garantir uma experiência de usuário fluida e responsiva, acessível tanto em desktops quanto em dispositivos móveis.

GitHub	Controle de versões, gestão de tarefas e colaboração	Organização, divisão de responsabilidades e rastreabilidade
SQL Server	Banco de dados robusto e altamente compatível com C#.	Banco de dados da Microsoft e mais compatível com o conhecimento geral do grupo.
Visual Studio	Principal ferramenta para edição de código em winforms	É utilizado para fazer trabalhos em c# por permitir a visualização, edição e depuração do código-fonte
Microsoft Azure	Hospedagem do banco de dados na nuvem	Disponibilidade, segurança e escalabilidade
Windows Forms	Biblioteca para criação de interfaces gráficas	Facilidade na construção de interfaces

Fonte: Elaborado pelos componentes do grupo.

4. CONSIDERAÇÕES FINAIS

Este trabalho apresentou o desenvolvimento de um sistema financeiro voltado para pequenas e médias empresas, com foco na automatização do controle de entradas, saídas e cadastros, por meio de uma aplicação desktop desenvolvida em C# e com banco de dados SQL Server hospedado na plataforma Microsoft Azure. A proposta atendeu à necessidade identificada de muitas organizações que ainda utilizam métodos manuais ou sistemas pouco eficientes para gestão financeira, oferecendo uma solução integrada, segura e escalável.

Durante o processo de desenvolvimento, foi possível aplicar conceitos fundamentais da engenharia de software, como modelagem de dados, controle de versões e práticas ágeis, além de explorar a integração entre diferentes tecnologias. A implementação de funcionalidades como dashboards dinâmicos, registro completo de transações e sistema de notificações proporcionou uma experiência mais completa e eficiente para o usuário final.

Entretanto, o projeto ainda apresenta limitações, principalmente no que tange à integração total com plataformas móveis e web, que está sendo endereçada pelo desenvolvimento da API RESTful. Ademais, a validação do sistema em ambientes reais de negócio ainda é necessária para aprimorar a interface, a usabilidade e a adequação das funcionalidades às demandas específicas de diferentes tipos de empresas.

Como trabalhos futuros, destacam-se a ampliação das funcionalidades da API para suportar múltiplas plataformas, a implementação de módulos avançados de análise financeira e inteligência de negócios, e a realização de estudos mais aprofundados de usabilidade com usuários reais. Dessa forma, espera-se que o sistema evolua para uma ferramenta ainda mais robusta, que contribua significativamente para a profissionalização da gestão financeira em organizações de pequeno e médio porte.

5. Referências

1. Desktop Operating System Market Share Worldwide – may 2025.
<https://gs.statcounter.com/os-market-share/desktop/worldwide/>

Acesso em Maio/2025.

2. Um tour pela linguagem C# - 09/05/2024 <https://learn.microsoft.com/pt-br/dotnet/csharp/tour-of-csharp/overview>

Acesso em 12/04/2025 às 10:44

3. Github Flow <https://docs.github.com/en/get-started/quickstart/github-flow/>

Acesso em 28/04/2025.

4. <https://azure.microsoft.com/en-us/services/sql-database/>

Acesso em 15/05/2025 às 19:21

5. Akinola S.A Recipient of the Award for Outstanding Achievement in Research and Academic Citizenship,

2023 In UJ

https://www.researchgate.net/publication/375905844_The_importance_of_financial_management_in_small_and_mediumsized_enterprises_SMEs_an_analysis_of_challenges_and_best_practices

Acesso em 21/04/2025 às 14:17

6. <https://www2.deloitte.com/us/en/insights/topics/leadership/digital-transformation-topics-for-corporate-technology-leadership.html>

Acesso em 21/04/2025 às 16:20

7. Chat Gpt

Acessado constantemente durante o desenvolvimento do projeto.