

Relatório Final



OFIPEÇAS

Aluno: Rodrigo Ferreira

Turma: TGPSI23S

N.º Aluno: 2223233

OfiPeças -- Gestão Simplificada de Encomendas para Oficinas

Síntese do Projeto

OfiPeças é uma aplicação de desktop desenvolvida em C# com SQL Server e interface Windows Forms, concebida para agilizar e centralizar o processo de encomenda de peças automóveis em oficinas.

Destinada a gestores e mecânicos, a solução oferece um sistema de autenticação seguro, controlo de acesso por perfis (administrador e utilizador padrão), gestão de stock, um ciclo de compra completo desde o carrinho até à encomenda final, e um histórico de compras com exportação de faturas em PDF. Ao eliminar processos manuais e centralizar a informação, o OfiPeças visa reduzir erros, otimizar o fluxo de trabalho e aumentar a eficiência operacional das oficinas.

Índice

1. Introdução

- 1.1. Contextualização do Projeto
- 1.2. Objetivos e Âmbito
- 1.3. Estrutura do Relatório

2. Tecnologias e Recursos

- 2.1. Tecnologias de Software
- 2.2. Recursos de Hardware e Ferramentas

3. Implementação

- 3.1. Cronograma de Desenvolvimento
- 3.2. Aspetos Técnicos do Desenvolvimento
 - 3.2.1. Análise de Requisitos
 - 3.2.2. Modelo de Dados
 - 3.2.3. Arquitetura da Aplicação
 - 3.2.4. Interfaces da Aplicação

4. Conclusão

- 4.1. Balanço Final
- 4.2. Dificuldades Encontradas e Soluções
- 4.3. Trabalho Futuro

5. Bibliografia

1. Introdução

1.1. Contextualização do Projeto

O presente documento constitui o Relatório Técnico Final do projeto OfiPeças, desenvolvido no âmbito da Prova de Aptidão Profissional. O OfiPeças consiste numa aplicação de desktop, desenvolvida em C# com uma base de dados relacional em SQL Server, que tem como objetivo principal otimizar e automatizar o processo de encomenda de peças em oficinas automóveis.

A solução foi concebida para substituir métodos tradicionais e manuais, como o telefone ou o email, que são frequentemente morosos e suscetíveis a erros humanos. Ao centralizar todo o fluxo de trabalho numa única aplicação robusta, desde a autenticação do utilizador até à gestão de encomendas e utilizadores por parte de um administrador, o projeto visa proporcionar um aumento significativo de eficiência e controlo para o ambiente profissional.

1.2. Objetivos e Âmbito

O projeto teve como principal objetivo o desenvolvimento de uma aplicação para ambiente Windows que permitisse às oficinas encomendarem peças de forma simples, organizada e segura. Os objetivos secundários, que foram todos cumpridos, incluíram:

- **Reduzir o tempo de procura e seleção de peças** através de um sistema de filtros por categoria e pesquisa por nome.
- **Implementar um ciclo de compra completo**, com a gestão de um carrinho de compras e a finalização de encomendas com registo permanente.
- **Garantir um sistema de autenticação seguro** com dois níveis de perfis: utilizador normal e administrador, com funcionalidades distintas.

- **Fornecer ferramentas de gestão**, como um histórico de encomendas com capacidade de exportação de faturas e um painel de administração completo.

O público-alvo desta aplicação são as oficinas automóveis, sendo que a sua utilização se destina tanto a mecânicos, que podem consultar e requisitar peças, como a gestores, que podem administrar todo o catálogo, utilizadores e encomendas.

1.3. Estrutura do Relatório

Este relatório está estruturado de forma a apresentar detalhadamente todo o processo de desenvolvimento do projeto. O capítulo 2 descreve as tecnologias e recursos utilizados. O capítulo 3 detalha a implementação, apresentando a análise de requisitos, o modelo de dados, a arquitetura do software e as interfaces desenvolvidas. Finalmente, o capítulo 4 apresenta a conclusão do projeto, fazendo um balanço final, discutindo as dificuldades encontradas e propondo possíveis evoluções futuras.

2. Tecnologias e Recursos

Neste capítulo são detalhadas todas as tecnologias, ferramentas de software e recursos de hardware utilizados para o desenvolvimento do projeto OfiPeças.

2.1. Tecnologias de Software

As tecnologias base do projeto seguiram os requisitos obrigatórios, sendo complementadas por bibliotecas externas que permitiram elevar a qualidade visual e funcional da aplicação.

- **Linguagem de Programação: C#**
 - Foi utilizada a linguagem C# no ecossistema .NET Framework, devido à sua robustez, orientação a objetos e integração nativa com o ambiente Windows para a criação de aplicações de desktop.
- **Base de Dados: Microsoft SQL Server**
 - Todas as informações da aplicação, desde utilizadores a peças e encomendas, são armazenadas e geridas por uma base de dados relacional implementada em Microsoft SQL Server. Esta escolha deve-se à sua fiabilidade, performance e excelentes ferramentas de gestão.
- **Interface Gráfica: Windows Forms (WinForms)**
 - A interface com o utilizador foi totalmente desenvolvida em Windows Forms, a framework padrão da Microsoft para a criação de aplicações de desktop em ambiente Windows.
- **Bibliotecas Externas (NuGet Packages):**
 - **Guna UI2 Framework for Windows Forms:** Esta biblioteca foi fundamental para o design da aplicação. Foi escolhida para substituir os controlos visuais padrão do WinForms, permitindo a criação de uma interface moderna, esteticamente apelativa e com uma melhor experiência de utilizador, através de componentes com mais opções de personalização (bordas arredondadas, animações, etc.).

- **DotNetEnv:** Para aumentar a segurança e a portabilidade da aplicação, esta biblioteca foi utilizada para carregar as credenciais de acesso à base de dados a partir de um ficheiro de ambiente (.env), evitando que informações sensíveis ficassem escritas diretamente no código-fonte.
- **PDFsharp:** Para a funcionalidade de exportação de faturas, foi escolhida a biblioteca PDFsharp. Após uma análise inicial de outras opções, esta demonstrou ser uma solução estável e eficaz para gerar documentos PDF de forma programática.

2.2. Recursos de Hardware e Ferramentas

- **Recursos de Hardware:**
 - O desenvolvimento foi realizado num computador com especificações padrão (Processador i5/i7 ou equivalente, 8GB de RAM, Disco SSD), não havendo requisitos de hardware específicos para além dos necessários para correr o sistema operativo Windows 10 ou superior e as ferramentas de desenvolvimento.
- **Ferramentas de Desenvolvimento (Software):**
 - **Microsoft Visual Studio 2022:** Foi o Ambiente de Desenvolvimento Integrado (IDE) utilizado para escrever, compilar e depurar todo o código C# da aplicação.
 - **Microsoft SQL Server Management Studio (SSMS):** Ferramenta essencial para desenhar, implementar e gerir a base de dados SQL Server.
 - **Git e GitHub:** O sistema de controlo de versões Git foi utilizado para gerir as alterações ao código-fonte e garantir a segurança do projeto.

3. Implementação

Neste capítulo, são detalhados o cronograma de desenvolvimento do projeto e os aspetos técnicos mais relevantes da sua implementação.

3.1. Cronograma de Desenvolvimento do Projeto

Fase	Duração Estimada	Descrição das Tarefas
1. Análise e Planeamento	1 Semana	Definição do tema, objetivos, requisitos e protótipo inicial.
2. Base de Dados e Autenticação	1 Semana	Desenho e implementação da base de dados. Desenvolvimento do sistema de login, registo e recuperação.
3. Loja e Carrinho de Compras	2 Semanas	Desenvolvimento da interface da loja, carregamento de produtos e implementação do carrinho de compras.
4. Checkout e Histórico	1 Semana	Implementação do fluxo de encomenda, gestão de stock e histórico de compras com exportação para PDF.
5. Painel de Administração	1 Semana	Criação do painel de gestão para peças, categorias e utilizadores.
6. Testes e Documentação	1 Semana	Testes finais, depuração e elaboração da documentação.

3.2. Aspetos Técnicos do Desenvolvimento

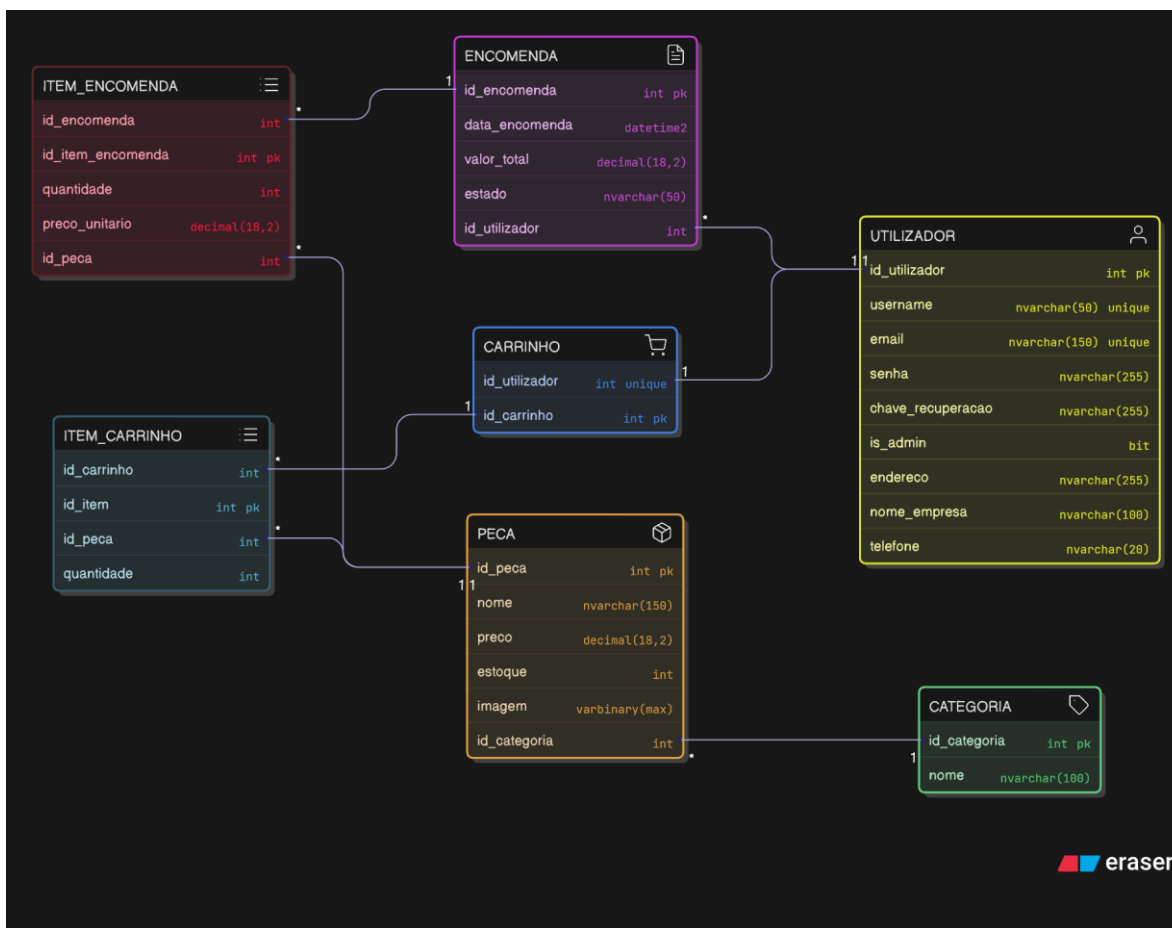
3.2.1. Análise de Requisitos

A seguir, é apresentada a lista de requisitos definidos na fase inicial e o estado de cumprimento final de cada um.

- **REQ0001 - Sistema de login com diferentes perfis:** Cumprido.
- **REQ0002 - Interface para seleção de viatura:** Não Cumprido (Decisão estratégica de simplificar o âmbito).
- **REQ0003 - Pesquisa de peças por categoria:** Cumprido.
- **REQ0004 - Adição de peças a um carrinho:** Cumprido.
- **REQ0005 - Finalização de encomenda:** Cumprido.
- **REQ0006 - CRUD para gestão de peças:** Cumprido.
- **REQ0007 - CRUD para gestão de viaturas:** Não Cumprido.
- **REQ0008 - CRUD para categorias de peças:** Cumprido.
- **REQ0009 - Área de administração para utilizadores:** Cumprido.
- **REQ0010 - Capacidade de armazenar até 200 produtos:** Cumprido.
- **REQ0011 - Tempo médio de resposta até 2s:** Cumprido.

3.2.2. Modelo de Dados

A base de dados OfiPecas foi implementada em SQL Server e é composta por 7 tabelas que modelam as entidades principais da aplicação, garantindo a integridade dos dados através de chaves primárias, estrangeiras e restrições.



3.2.3. Arquitetura da Aplicação

A aplicação segue uma arquitetura em camadas para promover a separação de responsabilidades e a manutenibilidade do código:

1. **Camada de Interface (UI):** Os formulários Windows Forms (Loja, Carrinho, etc.), responsáveis pela interação com o utilizador.
2. **Camada de Serviços:** Classes estáticas (StoreService, CartService, AdminService, etc.) que contêm a lógica de negócio.
3. **Camada de Acesso a Dados:** A classe DatabaseConnection que centraliza a conexão com a base de dados.

Esta separação foi crucial para organizar um projeto desta dimensão de forma limpa e escalável.

3.2.4. Interfaces da Aplicação

A seguir, são apresentados screenshots das principais interfaces da aplicação.

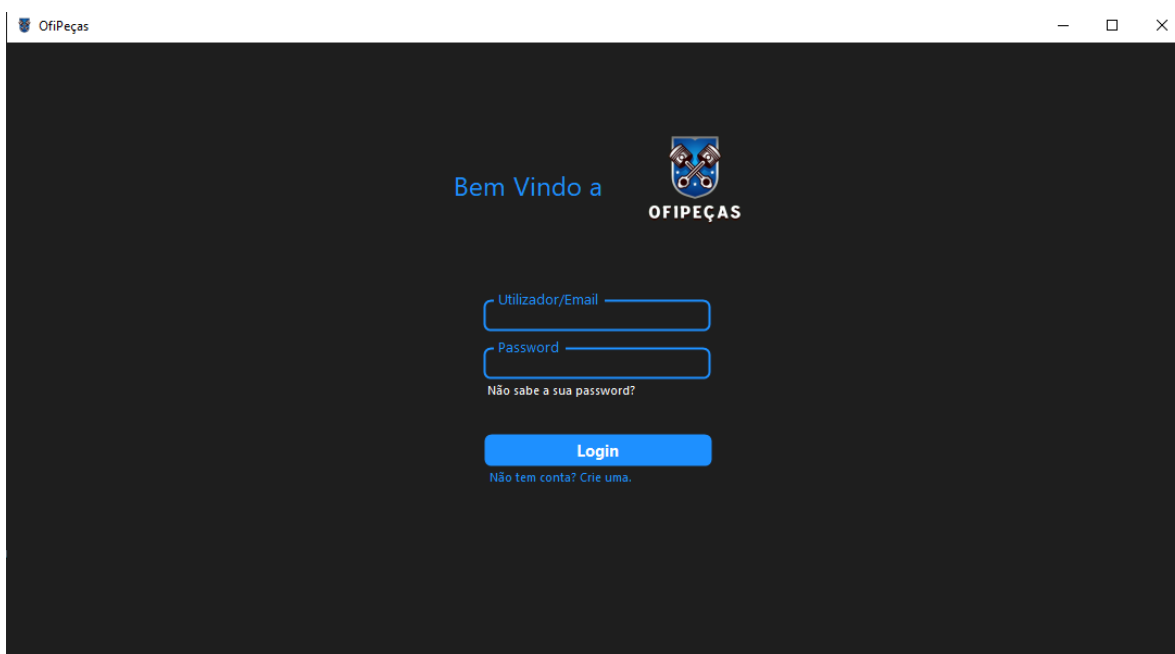


Figura 1: Ecrã de Login

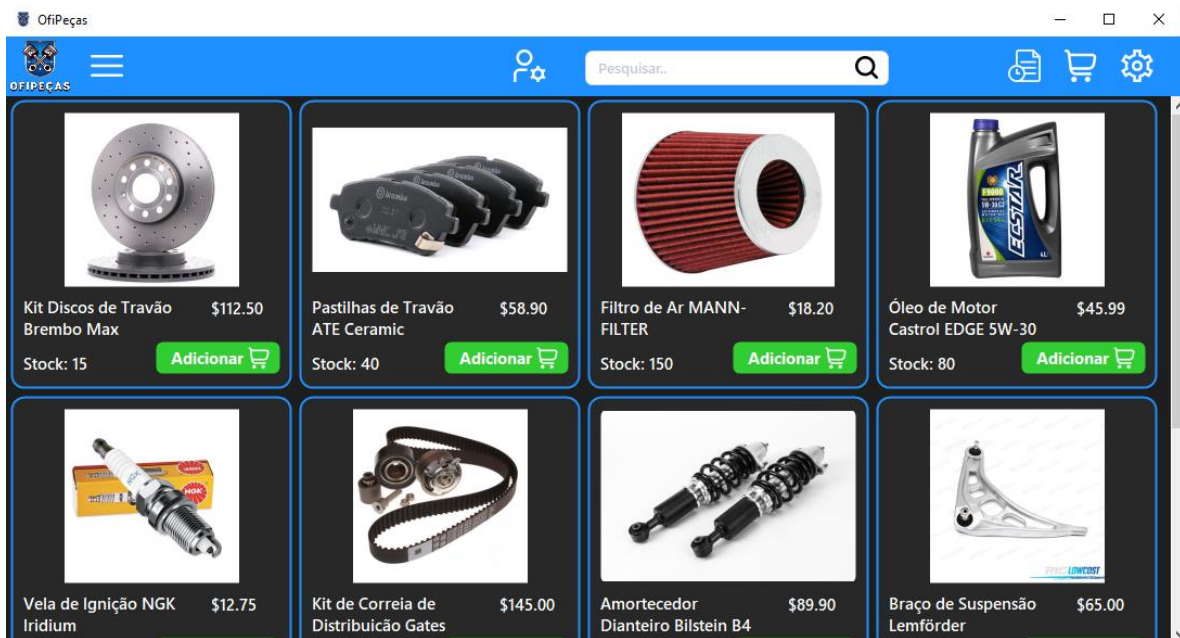


Figura 2: Ecrã Principal da Loja

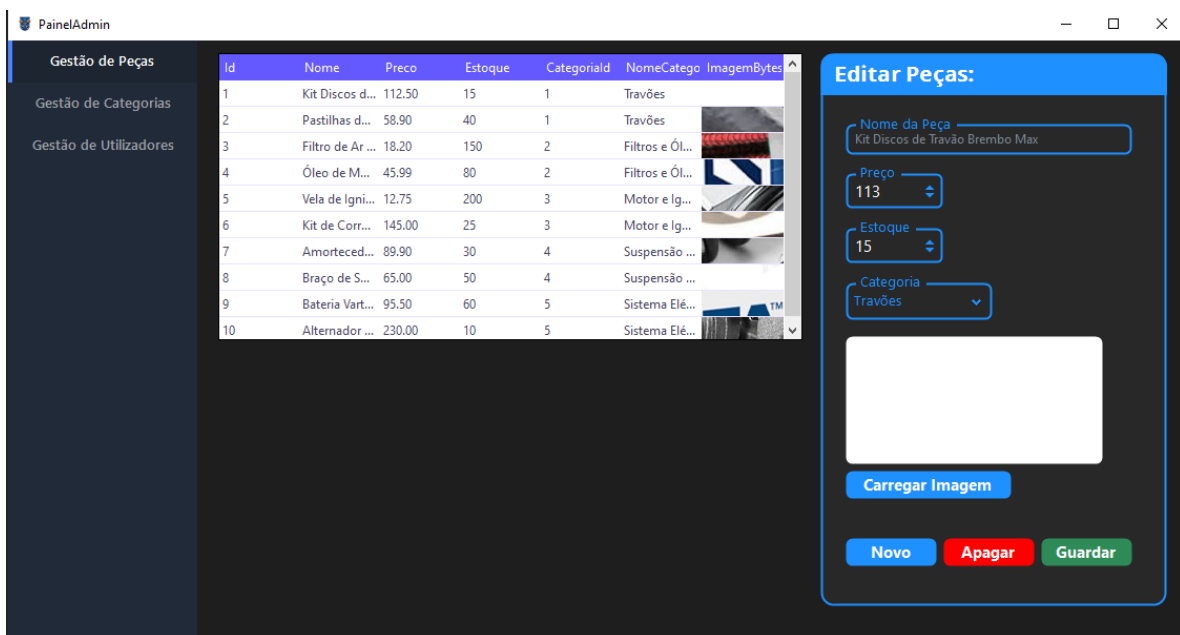


Figura 3: Painel de Administração

4. Conclusão

Neste capítulo final, é efetuado um balanço do trabalho desenvolvido, destacando os objetivos alcançados, as dificuldades encontradas e as possíveis vias para o desenvolvimento futuro do projeto.

4.1. Balanço Final

O projeto OfiPeças foi concluído com sucesso, cumprindo todos os objetivos centrais definidos. O resultado é uma aplicação de desktop robusta e funcional, com uma interface de utilizador moderna, que gere de forma eficiente todo o ciclo de encomenda de peças numa oficina. Foram implementadas com êxito funcionalidades complexas como um sistema de autenticação seguro, um catálogo de produtos dinâmico, um ciclo de compra completo com gestão de stock, um histórico de encomendas com exportação para PDF, e um painel de administração abrangente.

Considera-se que o projeto representa uma solução viável e de valor acrescentado para o contexto oficial, demonstrando a aplicação prática das competências adquiridas.

4.2. Dificuldades Encontradas e Soluções

Durante o desenvolvimento, foram encontrados alguns desafios técnicos:

- **Geração de Faturas em PDF:** A integração de bibliotecas de PDF revelou-se um desafio, exigindo a troca da biblioteca iText7 pela PDFsharp e a implementação de um FontResolver para garantir a compatibilidade com as fontes do sistema.
- **Comportamento de Eventos em Windows Forms:** Foram encontrados problemas subtis na interface do Painel de Administração, relacionados com o "disparo de eventos" em controlos ComboBox, que foram resolvidos com a implementação de lógica de controlo de eventos.

Estes desafios foram cruciais para aprofundar os conhecimentos em depuração de código e na interação com bibliotecas externas.

4.3. Trabalho Futuro

Embora completo no seu âmbito atual, o projeto tem potencial para evoluir com novas funcionalidades:

- **Filtro Avançado por Viatura:** Implementar a funcionalidade originalmente planeada de filtrar peças por Marca, Modelo e Ano.
- **Notificações por Email:** Integrar um serviço de envio de emails para notificar os utilizadores sobre o estado das suas encomendas.
- **Relatórios e Estatísticas:** Desenvolver uma secção no painel de administração para gerar relatórios de vendas.

5. Bibliografia

- **Microsoft.** ".NET Desktop Guide - Windows Forms Overview." Microsoft Learn. Disponível em: <https://learn.microsoft.com/en-us/dotnet/desktop/winforms/>
- **Microsoft.** "SQL Server Technical Documentation." Microsoft Learn. Disponível em: <https://learn.microsoft.com/en-us/sql/sql-server/>
- **Guna UI2.** "Guna UI2 Framework for Windows Forms." Disponível em: <https://gunauui.com/>
- **DotNetEnv.** "DotNetEnv on NuGet." Disponível em: <https://www.nuget.org/packages/DotNetEnv>
- **PDFsharp Team.** "PDFsharp Library." Disponível em: <http://www.pdfsharp.net/>