



Instituto Politécnico de Tomar

Escola Superior de Tecnologia de Tomar

Web Application BackOffice

Relatório de Estágio

Samuel Leal Gomes Luís

Mestrado em Analítica e Inteligência Organizacional

Lisboa | Novembro | 2023



Instituto Politécnico de Tomar

Escola Superior de Tecnologia de Tomar

Web application back-office

Samuel Leal Gomes Luís

Mestrado em Analítica e Inteligência Organizacional

Orientado por:

Professora Doutora Sandra Maria Gonçalves Vilas Boas Jardim – Instituto Politécnico de Tomar

Carlos Freire – Red IT

Júri:

Professor Adjunto Hélder da Côrte Pestana – Instituto Politécnico de Tomar

Professora Adjunta Sandra Vilas Boas Jardim – Instituto Politécnico de Tomar

Professor Adjunto Vasco Renato Gestosa da Silva – Instituto Politécnico de Tomar

Relatório de Estágio apresentado ao Instituto Politécnico de Tomar para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Analítica e Inteligência Organizacional

RESUMO

O presente relatório tem como objetivo sintetizar e expor as atividades, experiências e o conhecimento adquirido ao longo do estágio curricular. Pondo em prática todos os conhecimentos obtidos, no âmbito da unidade curricular de Estágio, pertencente ao Mestrado em Analítica e Inteligência Organizacional – Especialização em pertencente ao Instituto Politécnico de Tomar. O estágio descrito teve a duração de 780 horas, tendo início no mês de fevereiro de 2023 e duração de aproximadamente 5 meses.

Durante este período participei no desenvolvimento do projeto descrito neste relatório. O estágio foi realizado na empresa tecnológica Red IT, sediada em Alvalade na cidade de Lisboa. Esta empresa tem por missão o desenvolvimento de serviços e soluções inovadoras no setor das telecomunicações, o presente estágio decorreu em concreto no departamento da academia Red IT.

O estágio permitiu um contacto direto com a realidade de desenvolvimento tecnológico a nível empresarial, aquisição de novos conhecimentos e novas competências técnicas. No presente relatório expõe-se os conhecimentos adquiridos, as tarefas e os trabalhos desenvolvidos ao longo deste estágio.

Palavras-chave: Red It, back-end, web application, arquitetura em 3 camadas.

ABSTRACT

The purpose of this report is to synthesize and present the activities, experiences, and knowledge gained during the internship. Putting into practice all the knowledge acquired within the scope of the Internship course, part of the master's program in Analytics and Organizational Intelligence - Specialization at the Polytechnic Institute of Tomar. The described internship lasted for 780 hours, starting in February 2023, and lasting approximately 5 months.

During this period, I participated in the development in the project described in this report. The internship took place at the technology company Red IT, located in Alvalade in the city of Lisbon. This company's mission is to develop innovative services and solutions in the telecommunications sector, and the internship specifically occurred within the Red IT Academy department.

The internship provided a direct insight into the reality of technological development in the business context, as well as the acquisition of new knowledge and technical skills. This report presents the knowledge acquired, tasks performed, and work carried out during this internship.

Keywords: Red It, back-end, web application, 3-tier architecture.

AGRADECIMENTOS

A concretização de um projeto desta natureza não se deve apenas do seu autor, mas também a todos aqueles que, de forma direta ou indireta, cruzaram este caminho. Foi enorme e constante a partilha de dúvidas, incertezas, conquistas e muitas, muitas aprendizagens.

Agradeço, em primeiro lugar, à minha família pela força indispensável que me foi transmitida, pela motivação que nos foi passada para a concretização do projeto desde o primeiro dia. As suas presenças foram pertinentes e imperativas.

Um especial agradecimento ao orientador do projeto da empresa Red IT, Carlos Freire, pelo grande apoio, pela disponibilidade, pela dedicação, pela orientação, que me levou a construir o caminho certo para percorrer desde a idealização até à concretização deste projeto.

Quero também agradecer ao orientador de projeto da Instituto Politécnico de Tomar, Sandra Jardim, pela ajuda e pelo apoio durante o desenvolvimento do projeto, pois, sem ela, o sucesso não teria sido atingido.

Quero também agradecer a todos os docentes pelos conhecimentos que nos transmitiram ao longo destes anos e as ajudas constantes que foram importantes para o desenrolar do projeto e para a minha evolução pessoal e profissional.

Por fim, agradecemos a todos os meus amigos e colegas de curso que sempre estiveram dispostos a ajudar e a colaborar durante percurso académico.

Obrigado!

GLOSSÁRIO

Back-End

Back-end é a parte do software que lida com o armazenamento e o processamento de dados, sem que o usuário veja. Garante que tudo funcione corretamente nos bastidores, como um suporte invisível para o que está visível no ecrã.

Web application

Web application é um tipo de software a que se pode aceder através de um navegador da web, como o Google Chrome ou o Mozilla Firefox, sem a necessidade de descarregar no dispositivo do utilizador. Oferece funcionalidades semelhantes às de uma aplicação, mas é acedida diretamente através da internet.

Front-end

Front-end é o lado do cliente, refere-se à interface que o cliente vê.

TABELA DE ACRÓNIMOS

ER: Entidade e Relacionamento

SSMS: *SQL Server Management Studio*

MVVC: *Model–view–viewmodel*

CSS: *Cascade Style Sheet*

HTML: *Hyper Text Markup Language*

HTTP: *Hypertext Transfer Protocol*

IPT: Instituto Politécnico de Tomar

JSON: *JavaScript Object Notation*

MVC: *Model View Controller*

SQL: *Structured Query Language*

OTRS: *Open-Source Ticket Request System*

IT: *Tecnologias da Informação*

API: *Application Programming Interface*

Índice

1 Introdução	13
1.1. Motivação	13
1.2. Solução	13
1.3. Descrição da empresa Red It	14
1.4. Localização.....	14
1.5. Estrutura do documento.....	15
2.Definição do Problema	16
2.1 Definição do problema	16
2.2 Objetivos previstos	17
2.3 Metodologia Utilizada.....	17
3 Análise de Requisitos e Conceção da Aplicação.....	19
3.1 Diagrama de Contexto	19
3.2 Diagrama de Casos de Uso.....	19
3.3 Descrição dos Casos de Uso.....	22
3.4 Diagramas de Sequências	31
3.5 Diagrama de Atividades	32
3.6 Diagrama de Classes.....	35
3.7 Arquitetura da Solução	37
4 Tecnologias e Software utilizado	37
4.1 Tecnologias.....	37
4.1.1 Microsoft Teams.....	37
4.1.2 Visual Studio 2022	38
4.1.3 Jira Software	38
4.1.4 Git.....	38
4.1.5 Sql Server Manager	38
4.1.6 Net 6	39
4.2 Linguagens	39
4.2.1 Back-end.....	39
4.2.2 Front-end	39
4.3 Logging.....	40
4.4 Validação	40
4.5 Arquitetura e Padrões	40
5 Implementação	41
5.1 C# em 3 camadas.....	41
5.2 Base de Dados da Solução.....	41
5.3 Descrição da aplicação	42
5.3.1 Janela de login	42
5.3.2 Janela de área Inicial.....	43

5.3.3 Janela das listas.....	44
5.3.4 Janela das listas produtos.....	44
5.3.5 Janela das listas fornecedores	45
5.3.6 Janela das listas categorias	46
5.3.7 Janela das listas clientes	46
5.3.8 Janela das listas ordens.....	47
5.3.9 Janela das listas utilizadores	47
5.3.10 Janela inserir produto	48
5.3.11 Janela editar produto.....	49
5.3.12 Janela eliminar produto	50
5.3.13 Janela detalhes produto.....	51
5.3.14 Janela detalhes fornecedores	51
5.3.15 Janela editar fornecedor.....	52
5.3.16 Janela detalhes colaborador	53
5.3.17 Janela detalhes encomenda	53
5.3.18 Janela editar encomenda.....	54
5.3.19 Janela criar utilizador.....	55
5.3.20 Janela editar utilizador.....	55
5.3.21 Janela gerir claims do utilizador	56
5.3.22 Janela eliminar utilizador.....	57
6 Conclusão	58
6.1 Trabalho desenvolvido	58
6.2 Principais Obstáculos	58
6.3 Trabalho Futuro	59
6.4 Considerações Finais	59
Referências	60

Índice de Tabelas

Tabela 1 Tabela de casos de uso.....	20
Tabela 2 Tabela de log in	24
Tabela 3 tabela de eliminação de produto	25
Tabela 4 Edição do produto.....	26
Tabela 5 Visualização dos produtos	26
Tabela 6 Eliminar fornecedores.....	27
Tabela 7 Edição dos fornecedores	28
Tabela 8 Edição do endereço do fornecedor	29
Tabela 9 Edição dos utilizadores	30
Tabela 10 Edição dos user claims.....	31

Índice de figuras

Figura 1 Funcionamento da metodologia agile	18
Figura 2 Diagrama de contexto	19
Figura 3 Diagrama de caso de uso.....	21
Figura 4 Diagrama de caso de uso.....	21
Figura 5 Diagrama de seqüências.....	32
Figura 6 Diagrama de atividades.....	33
Figura 7 Diagrama de atividades.....	34
Figura 8 Diagrama de classes.....	36
Figura 9 Diagrama da Solução	37
Figura 10 Arquitetura em camadas.....	41
Figura 11 Base de dados.....	42
Figura 12 Página Inicial.....	42
Figura 13 Autenticação.....	43
Figura 14 Dashboard	44
Figura 15 Lista de produtos.....	45
Figura 16 Lista de fornecedores	45
Figura 17 Lista de categorias.....	46
Figura 18 Lista de colaboradores	46
Figura 19 Lista de pedidos	47
Figura 20 Lista de utilizadores	48
Figura 21 Criação de produto	49
Figura 22 Edição produto	50
Figura 23 Eliminação do produto	51
Figura 24 Detalhes de produto	51
Figura 25 Detalhes do fornecedor	52
Figura 26 Edição de do fornecedor.....	53
Figura 27 Detalhes do colaborador.....	53
Figura 28 Detalhes do pedido.....	54
Figura 29 Edição do pedido.....	55
Figura 30 Criação do utilizador.....	55
Figura 31 Edição do utilizador	56
Figura 32 Edição do utilizador	57
Figura 33 Eliminação do utilizador	57

1 Introdução

Neste primeiro capítulo é feito um enquadramento do projeto desenvolvido ao longo do estágio, assim como o seu âmbito. Seguidamente, é feita uma breve descrição da empresa referindo alguns aspetos importantes a nível tecnológico, terminando o capítulo com uma breve descrição da estrutura do documento apresentado.

1.1 Motivação

No âmbito da unidade curricular de Estágio, do 2º ano do curso de Mestrado em Analítica e Inteligência Organizacional foi realizado um estágio curricular, com duração de 780 horas, correspondente a 5 meses, na empresa Red IT.

O principal objetivo deste projeto é integrar uma aplicação *Back-Office* com uma já existente de *Front-Office* utilizando a mesma base de dados.

Este projeto é uma *web app* de uma loja online que fornece aos utilizadores uma eficiência na gestão interna dos produtos, dos clientes e dos utilizadores da aplicação.

Outros objetivos estabelecidos para a realização deste estágio, centram-se na aquisição de conhecimentos, ganho de experiência e no desenvolvimento de competências na área de estudo.

1.2 Solução

A solução encontrada para a proposta foi feita de acordo com os requisitos pedidos e pretendidos, foi feito a criação de aplicação em C# e .net6 de gestão de uma loja online para a empresa Red IT.

Esta aplicação foi desenvolvida na plataforma Visual Studio 2022, o SSMS foi usado para fazer a gestão da base de dados existente e o GitHub foi usado como controlador de versões.

Com esta aplicação os utilizadores podem executar as seguintes funcionalidades:

- Manipular (inserir, alterar, apagar, consultar) todos os dados relativamente á loja online (produtos, fornecedores, categorias, clientes).
- Gestão de utilizadores na aplicação.
- Acesso aos dados no *dashboard*.

1.3 Descrição da empresa Red It

A empresa Red IT1 foi fundada em 2016 em Lisboa, onde cresceu de forma orgânica para mais de 150 colaboradores. Uma empresa conhecida pelos seus especialistas em IT que trazem um profundo conhecimento em vários setores e uma variedade de perspetivas onde questionam o status quo e promovem a mudança.

A Red IT é uma empresa de consultoria especializada em Sistemas e Tecnologia da Informação, focada em ajudar os *change makers* a mudar o futuro. As equipas da Red IT têm autonomia para agir com rapidez e colocam sempre as pessoas no centro de tudo.

Com transformações digitais bem-sucedidas em vários setores, apoiadas por uma cultura de colaboração e parcerias com os principais fornecedores de tecnologia do mundo, a Red IT preocupa-se em oferecer valiosas experiências de transformação de negócios.

1.4 Localização

A empresa possui 2 escritórios em Portugal localizados em Lisboa e no Porto, tendo clientes a nível mundial.²

O presente estágio decorreu maioritariamente em regime remoto, contudo mensalmente foram efetuadas reuniões presenciais realizadas nos escritórios de Lisboa.

¹Disponível em <https://www.redit.pt/en/about-us>

² Imagem obtida em <https://www.linkedin.com/company/redit/about/>

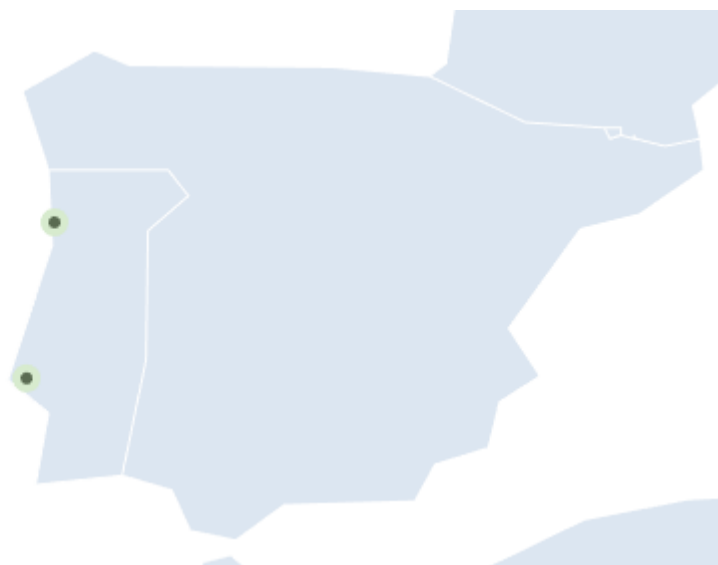


Figura 1 Localização dos escritórios

1.5 Estrutura do documento

Este documento está dividido em 6 capítulos:

- No segundo capítulo, será descrito a definição do problema e os objetivos pedidos pela empresa no início do desenvolvimento deste projeto. Será também descrita a metodologia utilizada no desenvolvimento deste projeto e os resultados esperados.
- No terceiro capítulo, é apresentado toda matéria teórica que sustenta a aplicação tais como casos de uso, diagrama de classes e entre outros.
- No quarto capítulo, é apresentada uma breve descrição das ferramentas e tecnologias utilizadas no desenvolvimento desta aplicação.
- No quinto capítulo, é descrita a implementação da solução deste projeto, ou seja, é feita uma descrição detalhada do trabalho realizado ao longo de todo o projeto.
- No sexto capítulo, está descrito a conclusão sobre este projeto e também o trabalho futuro que poderá ser implementado.

2 Definição do Problema

O projeto teve como objetivo solucionar desafios como a escalabilidade, segurança, eficiência, manutenção e a experiência do utilizador. Ao utilizar tecnologias modernas, bibliotecas relevantes e padrões adequados, a aplicação foi capaz de fornecer um suporte aprimorado para o *front-end* existente, oferecendo uma experiência do utilizador satisfatória.

2.1 Definição do problema

Escalabilidade e Eficiência:

- Antes de iniciar o desenvolvimento da plataforma, foi fundamental implementar uma arquitetura escalável que caso seja necessário possa crescer de forma flexível com o aumento do tráfego da base de dados dos clientes.
- É necessário garantir que o sistema seja eficiente, ou seja durante picos de demanda, é necessário evitar tempos de resposta lentos e instabilidade.

Segurança e Integridade dos Dados:

- A segurança dos dados é uma prioridade, devem ser sempre implementadas medidas de segurança para proteger informações confidenciais dos clientes, como por exemplo dados pessoais, dados financeiros e etc.
- A autenticação e a autorização adequadas, bem como a criptografia de dados, são essenciais para evitar vulnerabilidades.
- É importante conduzir testes regulares de segurança e manter um plano de resposta a incidentes.

Manutenção e Extensibilidade:

- O design da plataforma deve ser modular e facilmente extensível para facilitar a adição de novos recursos e futuras implementações.
- A utilização de boas práticas de desenvolvimento de software, como a adoção de padrões de projeto, pode simplificar a manutenção e o desenvolvimento contínuo.

Experiência do Utilizador:

- A experiência do utilizador deve ser priorizada desde o início do projeto.
- É necessário ter design intuitivo, uma interface de utilizador atraente e uma navegação simplificada para garantir uma experiência positiva.

2.2 Objetivos previstos

Os objetivos previstos para este projeto incluem:

- Construir uma aplicação de uma loja online eficiente, aproveitando as vantagens da plataforma .NET e da linguagem C#.
- Aprimorar a experiência do utilizador, tornando a plataforma amigável e fácil de navegar.
- Implementar medidas de segurança rigorosas para proteger os dados e garantir a confiabilidade da aplicação.
- Preparar a aplicação para um possível crescimento futuro, garantindo que ela possa acomodar o aumento da demanda.
- Antecipar e lidar com riscos potenciais para o sucesso do projeto.

2.3 Metodologia Utilizada

Neste projeto foi utilizado uma metodologia Agile³ que é baseada em uma abordagem iterativa e incremental, onde o trabalho é dividido em pequenos incrementos chamados de *sprints*. A equipa realiza reuniões diárias, conhecidas como *Daily Stand-ups*, para discutir o progresso, identificar impedimentos e definir as tarefas a serem executadas durante o dia. Esta metodologia enfatiza a colaboração, a comunicação frequente e a adaptação contínua ao longo do projeto.

No projeto em questão, foi implementado o *Kanban Board* utilizando uma ferramenta online específica, o *Jira software*. Foram definidas colunas de acordo com o fluxo de trabalho do projeto, adaptando-as às necessidades que foram surgindo. As tarefas foram representadas por cartões, contendo informações relevantes, como descrição, responsável pela tarefa e prazo estimado.

Durante as *Daily Stand-ups*, esteve-se no *Kanban Board* a rever as tarefas atribuídas. Os cartões eram movidos entre as colunas à medida que as tarefas progrediam, permitindo que houvesse uma visão clara do progresso do trabalho.

³ <https://www.jasminsoftware.pt/blog/metodologia-agile/>



4

Figura 2 Funcionamento da metodologia Agile

⁴ <https://kentechbr.wordpress.com/2019/03/15/metodologia-agile-no-faturamento-hospitalar/>

3 Análise de Requisitos e Conceção da Aplicação

3.1 Diagrama de Contexto

O diagrama de contexto como se pode ver na figura 3, é composto por fluxos de dados que mostram as interfaces entre o sistema e as entidades externas. Este permite identificar os limites dos processos, as áreas envolvidas com o processo e os relacionamentos com outros processos e elementos externos.

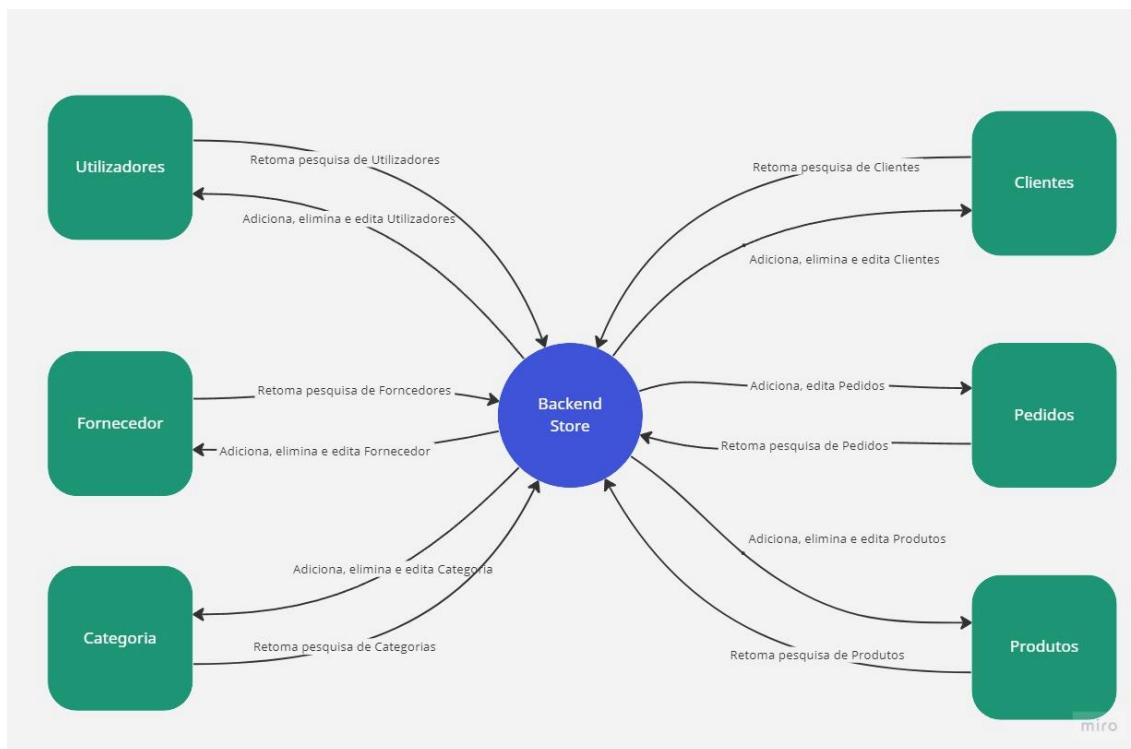


Figura 3 Diagrama de contexto

Como é apresentado no diagrama de contexto, o utilizador insere produtos, categorias, pedidos, fornecedores, clientes e utilizadores onde depois são guardados no sistema onde posteriormente poderão ser editados ou eliminados.

3.2 Diagrama de Casos de Uso

Um diagrama de casos de uso é composto por atores e por casos de uso. Os atores são entidades externas que interagem com a aplicação, os casos de uso representam as funcionalidades/serviços. Na tabela 1 apresenta-se a lista de casos de uso onde são definidos os objetivos dos mesmos.

Actor	Casos de uso	Objetivos
Admin	Gerir Utilizadores	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Utilizadores existentes.
	Gerir <i>Claims</i> dos Utilizadores	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar as <i>Claims</i> pertencentes aos Utilizadores existentes.
	Gerir Produtos	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Produtos existentes.
	Gerir Fornecedores	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Fornecedores existentes.
	Gerir Endereço dos Fornecedores	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Endereços pertencentes aos Fornecedores existentes.
	Gerir Colaboradores	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Colaboradores existentes.
	Gerir Endereço dos Colaboradores	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Endereços pertencentes aos Colaboradores existentes.
	Gerir Categorias	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar as Categorias existentes.
	<i>Log in/Log off</i>	O objetivo é o <i>Admin</i> poder aceder e sair da Aplicação.
	Aceder ao dashboard	O objetivo é o <i>Admin</i> poder visualizar a informação desposta no <i>dashboard</i> .
	Gerir os Items dos Pedidos	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Items dos Pedidos existentes.
	Gerir Pedidos	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Pedidos existentes.
	Gerir Endereço dos Pedidos	O objetivo é o <i>Admin</i> poder Criar, Editar, Eliminar e Pesquisar os Endereço pertencentes aos Pedidos existentes.

Tabela 1 de casos de uso

O diagrama de Casos de Uso, como mostra a Figura 4, descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do utilizador *Admin*. Permite ver de forma simples e rápida todas as interações que as funcionalidades do sistema têm entre si assim como todas as interações com os atores.

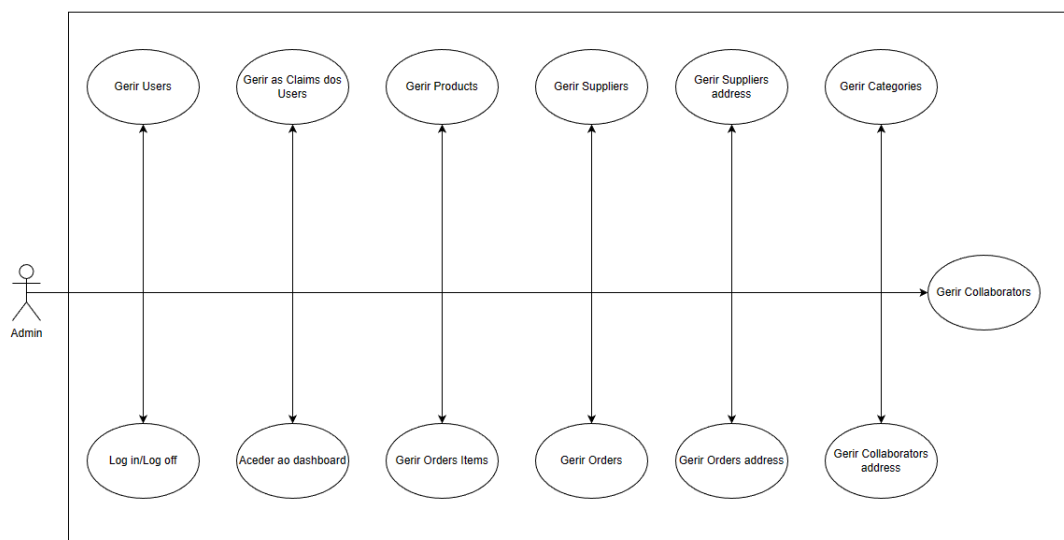


Figura 4 Diagrama de caso de uso

Como mostra a descrição dos casos de uso Figura 4, o ator *admin* gere os utilizadores, as *claims* dos utilizadores, os produtos, fornecedores, categorias, clientes, etc.

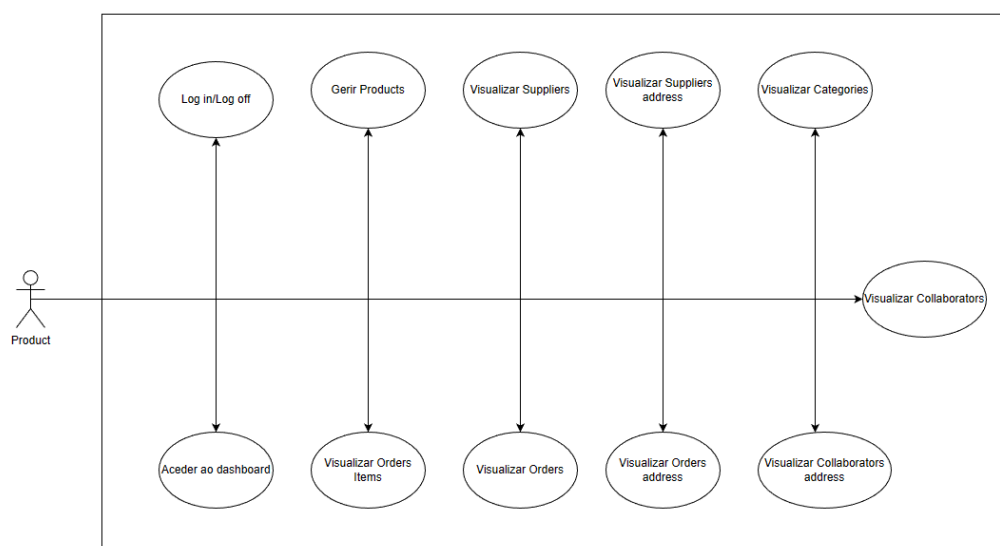


Figura 5 Diagrama de caso de uso

O diagrama de Casos de Uso, como mostra a Figura 5, descreve um cenário que mostra as

funcionalidades do sistema do ponto de vista do ator *Product*.

O ator *Product* gere os produtos, mas só tem acesso a visualização das categorias, fornecedores, categorias, clientes, etc.

3.3 Descrição dos Casos de Uso

As tabelas de descrição dos casos de uso são compostas por nome, descrição, atores envolvidos, pré-condições, fluxo principal, fluxos alternativos, suplementos e pós condições. O nome indica a designação do caso de uso que se está a definir. A descrição mostra para que serve o caso de uso que está a se desenvolver. Os atores envolvidos são todos os que interagem com o caso de uso. Pré-condições são as condições necessárias para que se possa executar o caso de uso. O fluxo principal é o que o sistema e o ator principal fazem neste caso de uso. Os fluxos alternativos são os fluxos que fazem parte do caso de uso, mas só são executados em certas circunstâncias.

Os casos de teste são as validações que são necessárias para se poder executar o caso de uso pretendido. Nas tabelas que se seguem estão alguns dos casos de uso que foram utilizados para a realização deste projeto.

Na tabela 3 está descrito em detalhe o caso de uso de fazer *Log in*.

Nome:	Log in
Descrição:	Este caso de uso tem como principal objetivo verificar se as credencias de login se encontram corretas.
Atores Envolvidos:	<i>Admin;</i> <i>Supplier;</i> <i>Product;</i>
Pré-Condições:	O ator tem de ter conta criada.
Fluxo Principal:	1º O sistema mostra os dados a preencher. 2º O ator preenche os dados que lhe são mostrados pelo sistema. 3º O sistema verifica se as credencias são as corretas. 4º O ator é redirecionado para a área do <i>dashboard</i> .

Fluxos Alternativos:	1º As credencias introduzidas são incorretas. 2º O username de login não existe.
Casos de Teste:	Verificar se o <i>username</i> existe.

Tabela 1Tabela de log in

Nome:	Inserir Produto
Descrição:	Este caso de uso tem como objetivo principal introduzir novos produtos e atribuir-lhe uma categoria e um fornecedor.
Atores Envolvidos:	<i>Admin</i> ; <i>Product</i> ;
Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i> do tipo <i>Product</i> e a <i>claim value</i> do tipo <i>Add</i> .
Fluxo Principal:	1.O ator escolhe a opção inserir novo produto. 2.O sistema disponibiliza os campos a serem preenchidos para inserir um novo produto. 3.O ator preenche os campos e seleciona o fornecedor e a categoria do produto. 4.O sistema disponibiliza os campos de acordo com o que o ator preencheu no passo anterior. 5.O sistema valida os dados introduzidos pelo ator. 6. O sistema adiciona os dados introduzidos pelo ator. 7. O sistema guarda os dados

	introduzidos pelo ator. 8. O sistema redireciona o ator para a página de visualização de produtos onde aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Pede para preencher os dados novamente.
Casos de Teste:	1.Verifica se os atributos foram introduzidos corretamente. 2.Verifica se falta preencher algum campo obrigatório.

Tabela 2 Tabela de inserir produto

Nome:	Eliminar Produto
Descrição:	Este caso de uso tem como objetivo principal eliminar um produto.
Atores Envolvidos:	<i>Admin;</i> <i>Product;</i>
Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i> do tipo <i>Product</i> e a <i>claim value</i> do tipo <i>delete</i> .
Fluxo Principal:	1.O ator escolhe a opção eliminar um produto existente. 2.O sistema disponibiliza a informação dos atributos do produto escolhido. 3.O sistema elimina o produto escolhido pelo ator. 4.O sistema verifica se a operação foi válida. 5.O sistema redireciona o ator para a página de visualização de produtos onde

	aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Não tem.
Casos de Teste:	Verifica se o produto existe

Tabela 3 tabela de eliminação de produto

Nome:	Editar Produto
Descrição:	Este caso de uso tem como principal objetivo alterar os atributos de um produto existente.
Atores Envolvidos:	<i>Admin;</i> <i>Product;</i>
Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i> do tipo Product e a claim value do tipo <i>Edit</i> .
Fluxo Principal:	1.O ator escolhe a opção editar produto. 2. O sistema disponibiliza o formulário de editar o produto. 3. O ator edita os dados do produto selecionado. 4.O sistema valida os dados introduzidos pelo o ator. 5.O sistema atualiza os dados introduzidos pelo ator. 6. O sistema guarda os dados introduzidos pelo ator. 7. O sistema redireciona o ator para a página de visualização de produtos onde aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Pede para preencher os dados novamente.

Casos de Teste:	Verifica se os atributos foram introduzidos corretamente. Verifica se falta preencher algum campo obrigatório.
------------------------	--

Tabela 4 Edição do produto

Nome:	Visualizar Produtos
Descrição:	O objetivo principal é o ator poder pesquisar um produto pelo nome, descrição, categoria e fornecedor.
Atores Envolvidos:	<i>Admin;</i> <i>Product;</i>
Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i> do tipo <i>Product</i> e a <i>claim value</i> do tipo <i>List</i> .
Fluxo Principal:	1.O sistema disponibiliza todos os produtos existentes. 2.O ator pesquisa o produto que deseja nome, descrição, categoria e fornecedor. 3.O sistema mostra todos os produtos de acordo com a pesquisa.
Fluxos Alternativos:	Não tem.
Casos de Teste:	Não tem.

Tabela 5 Visualização dos produtos

Nome:	Eliminar Fornecedores
Descrição:	Este caso de uso tem como objetivo principal eliminar um fornecedor.
Atores Envolvidos:	<i>Admin;</i> <i>Supplier;</i>

Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i> do tipo <i>Supplier</i> e a <i>claim value</i> do tipo <i>delete</i> .
Fluxo Principal:	1.O ator escolhe a opção eliminar um fornecedor existente. 2.O sistema disponibiliza a informação dos atributos do fornecedor escolhido. 3.O sistema verifica se existem produtos associados ao fornecedor. 4.O sistema elimina o fornecedor escolhido pelo ator. 5.O sistema verifica se a operação foi válida. 6.O sistema redireciona o ator para a página de visualização de fornecedores onde aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Não tem.
Casos de Teste:	Verifica se o fornecedor existe. Verifica se o fornecedor tem algum produto associado.

Tabela 6 Eliminar fornecedores

Nome:	Editar Fornecedores
Descrição:	Este caso de uso tem como principal objetivo alterar os atributos de um fornecedor existente.
Atores Envolvidos:	<i>Admin;</i> <i>Supplier;</i>
Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último

	é necessário que o ator tenha a <i>claim type</i> do tipo <i>Supplier</i> e a <i>claim value</i> do tipo <i>Edit</i> .
Fluxo Principal:	1.O ator escolhe a opção editar fornecedor. 2. O sistema disponibiliza o formulário de editar o fornecedor. 3. O ator edita os dados do fornecedor selecionado. 4.O sistema valida os dados introduzidos pelo o ator. 5.O sistema atualiza os dados introduzidos pelo ator. 6. O sistema guarda os dados introduzidos pelo ator. 7. O sistema redireciona o ator para a página de visualização de fornecedores onde aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Pede para preencher os dados novamente.
Casos de Teste:	Verifica se o fornecedor existe. Verifica se o atributo <i>document</i> se repete noutro fornecedor.

Tabela 7 Edição dos fornecedores

Nome:	Editar Endereço do Fornecedor
Descrição:	Este caso de uso tem como principal objetivo alterar a morada do fornecedor.
Atores Envolvidos:	<i>Admin;</i> <i>Supplier;</i>
Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i>

	do tipo <i>Supplier</i> e a <i>claim value</i> do tipo <i>Edit</i> .
Fluxo Principal:	1.O ator escolhe a opção editar o endereço fornecedor. 2. O sistema disponibiliza a informação existente do endereço do fornecedor. 3. O ator clica no botão de editar o endereço do fornecedor. 4.O sistema apresenta uma modal com os dados do endereço do fornecedor. 5.O ator altera os dados pretendios 6.O sistema valida os dados introduzidos pelo o ator. 7.O sistema atualiza os dados introduzidos pelo ator. 8. O sistema guarda os dados introduzidos pelo ator. 9. O sistema redireciona o ator para a página de visualização de endereços dos fornecedores onde aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Pede para preencher os dados novamente.
Casos de Teste:	Verifica se o endereço já existe. Verifica se o atributo <i>document</i> se repete noutro fornecedor.

Tabela 8 Edição do endereço do fornecedor

Nome:	Editar Utilizadores
Descrição:	Este caso de uso tem como objetivo principal alterar os dados de um utilizador.
Atores Envolvidos:	<i>Admin</i> ;
Pré-Condições:	O ator tem de ter conta criada, também é

	necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i> do tipo <i>User</i> e a <i>claim value</i> do tipo <i>Edit</i> .
Fluxo Principal:	1.O ator escolhe a opção editar utilizador. 2. O sistema disponibiliza o formulário de editar do utilizador. 3. O ator edita os dados do utilizador selecionado. 4.O sistema valida os dados introduzidos pelo o ator. 5.O sistema atualiza os dados introduzidos pelo ator. 6. O sistema guarda os dados introduzidos pelo ator. 7. O sistema redireciona o ator para a página de visualização de utilizadores onde aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Pede para preencher os dados novamente.
Casos de Teste:	Verifica se o modelo é válido. Verifica se o utilizador existe.

Tabela 9 Edição dos utilizadores

Nome:	Editar <i>Users Claims</i>
Descrição:	Este caso de uso tem como principal objetivo alterar as <i>claims</i> do utilizador.
Atores Envolvidos:	<i>Admin</i> ;
Pré-Condições:	O ator tem de ter conta criada, também é necessário estar autenticado e por último é necessário que o ator tenha a <i>claim type</i> do tipo <i>User</i> e a <i>claim value</i> do tipo <i>Edit</i> .
Fluxo Principal:	1.O ator escolhe a opção <i>claims</i> do utilizador.

	2. O sistema disponibiliza a informação existente do utilizador. 3. O ator clica no botão de <i>manage claims</i>. 4.O sistema apresenta uma modal com os dados do endereço do fornecedor. 5.O ator altera os dados pretendidos 6.O sistema valida os dados introduzidos pelo o ator. 7.O sistema atualiza os dados introduzidos pelo ator. 8. O sistema guarda os dados introduzidos pelo ator. 9. O sistema redireciona o ator para a página de visualização de produtos onde aparece uma notificação temporária de sucesso.
Fluxos Alternativos:	Pede para preencher os dados novamente.
Casos de Teste:	Verifica se o modelo é válido. Verifica se o utilizador existe.

Tabela 10 Edição dos user claims

3.4 Diagramas de Sequências

Os diagramas de sequência mostram as trocas de mensagens entre os vários objetos numa sequência temporal. São utilizados para descrever o fluxo de execução dos casos de uso e facilita a visualização do sistema, pois mostra que classes são chamadas e em que situações. Nos diagramas de sequências está representado o ator bem como os objetos e métodos ordenados pelo tempo.

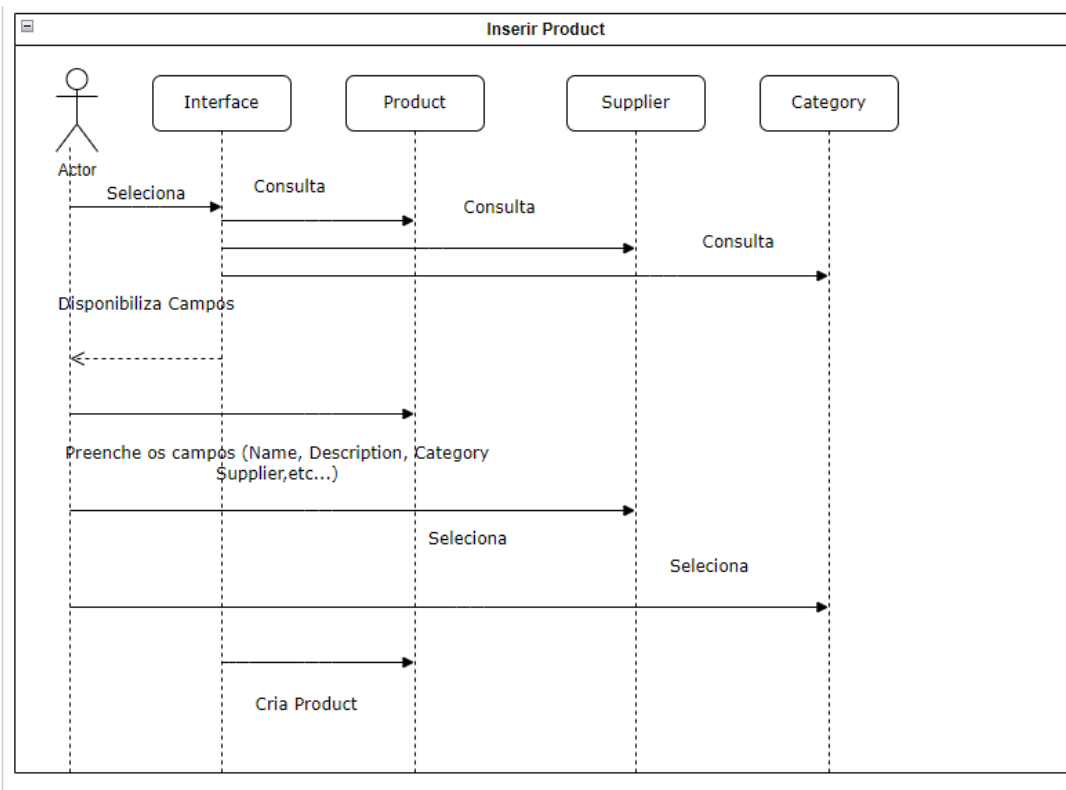


Figura 6 Diagrama de sequências

No diagrama de sequência, figura 6, o utilizador selecciona a interface criar produto, depois é feito uma consulta de todos os fornecedores e categorias existentes, por sua vez o sistema devolve o formulário a preencher. Depois de se introduzirem os dados do produto e de ter seleccionado a categoria e o fornecedor o utilizador prime o botão cria produto o sistema cria um produto e guarda os dados na base de dados.

3.5 Diagrama de Atividades

Um diagrama de atividades é basicamente um gráfico de fluxo que controla a passagem de uma atividade para outra e é usado para fazer a modelação dos aspetos dinâmicos do sistema.

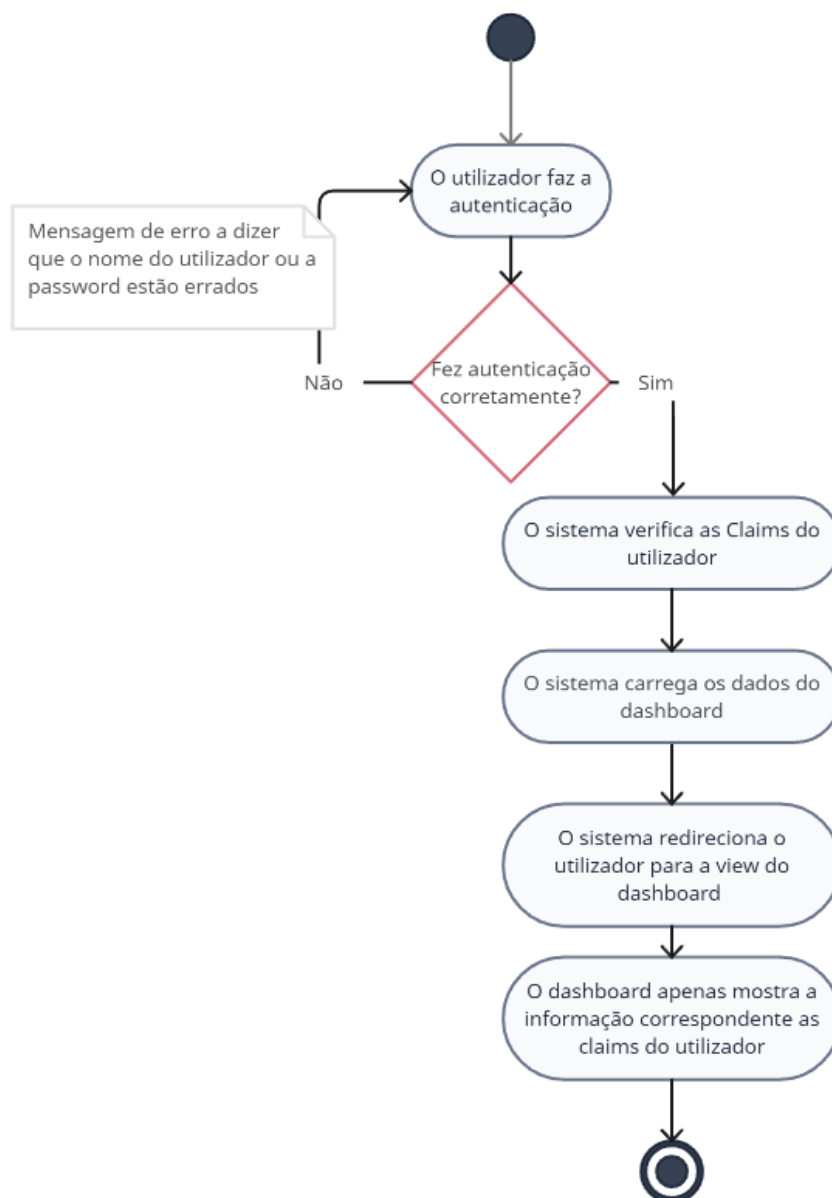


Figura 7 Diagrama de atividades

O processo começa quando o utilizador fornece as credenciais necessárias para acessar o sistema. O sistema, então, verifica a validade das credenciais fornecidas e confirma se o utilizador foi autenticado com sucesso.

Uma vez autenticado, o sistema procede à verificação das *claims* do utilizador.

Essas *claims* incluem permissões específicas para o acesso ao dashboard.

Após a verificação das *claims*, o sistema carrega os dados específicos do *dashboard*.

O sistema, então, filtra os dados do *dashboard* com base nas *claims* verificadas, garantindo que apenas os dados relevantes e autorizados sejam exibidos.

Finalmente, o *dashboard* mostra os dados filtrados de acordo com as *claims* verificadas,

proporcionando ao utilizador uma visualização alinhada com as permissões e privilégios atribuídos ao seu perfil de utilizador.

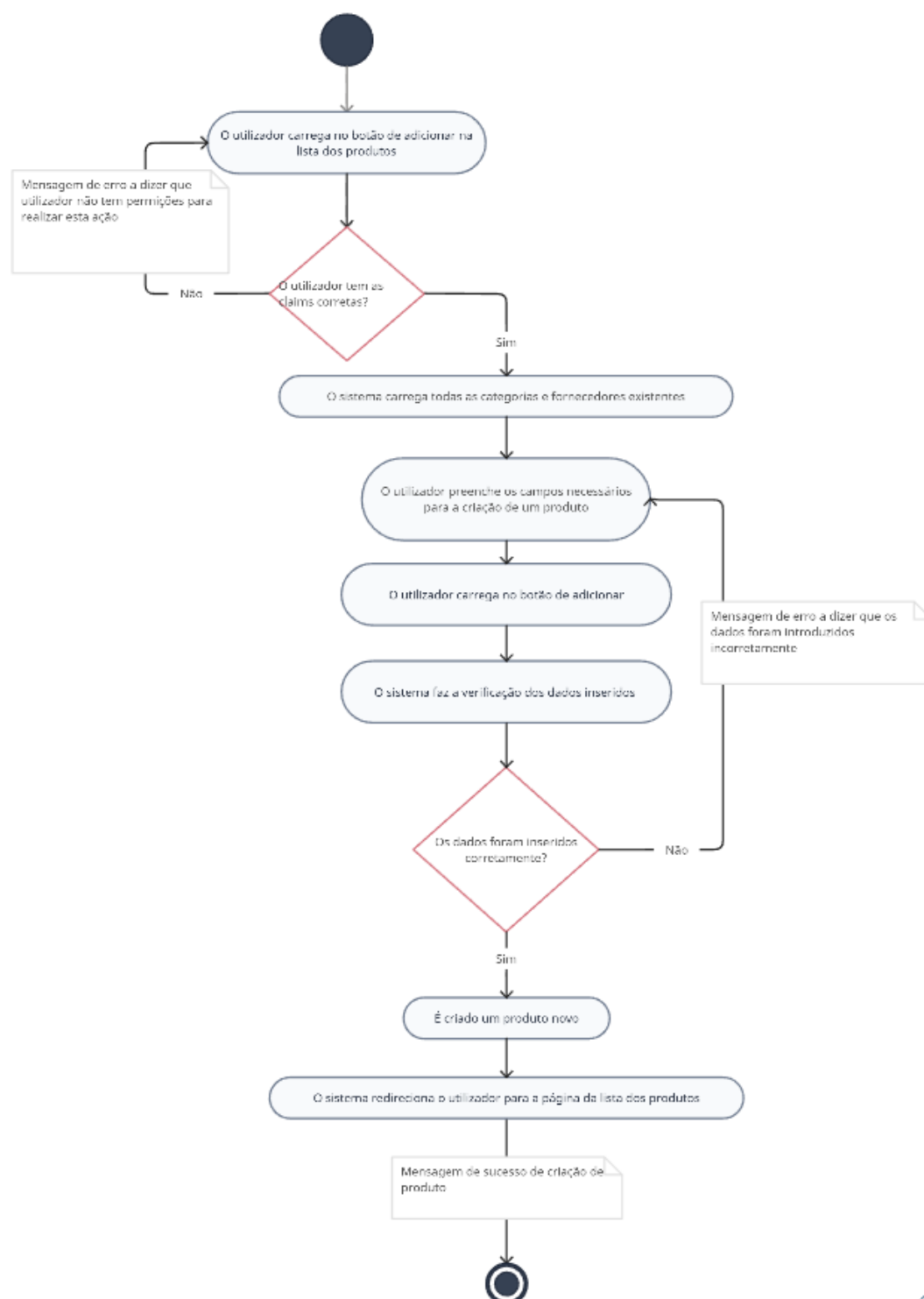


Figura 8 Diagrama de atividades

O utilizador inicia o processo ao carregar no botão de adicionar na lista de produtos, expressando a

intenção de adicionar um novo produto. O sistema, então, verifica se o utilizador possui as permissões apropriadas para adicionar produtos ao sistema.

Após a verificação das *claims*, o sistema carrega todas as categorias e fornecedores existentes, garantindo que o utilizador tenha acesso a todas as opções disponíveis ao adicionar um novo produto. O utilizador preenche os campos necessários para a criação de um produto, inserindo informações relevantes como nome, descrição, categoria e fornecedor.

Ao concluir o preenchimento, o utilizador carrega no botão de adicionar, sinalizando ao sistema que todos os campos foram devidamente preenchidos e que o utilizador deseja criar o produto.

O sistema realiza uma verificação dos dados inseridos, assegurando a integridade e a consistência das informações. Isso inclui a validação de campos obrigatórios, a verificação de formatos corretos de dados e a deteção de quaisquer erros ou omissões.

Após a verificação bem-sucedida dos dados, o sistema cria um produto no sistema, utilizando as informações fornecidas pelo utilizador durante o processo de adição.

Por fim, o sistema redireciona o utilizador para a página da lista de produtos, confirmando a criação bem-sucedida do novo produto, juntamente com uma mensagem de sucesso que notifica o utilizador sobre a conclusão bem-sucedida do processo de criação do produto.

3.6 Diagrama de Classes

Um diagrama de classes é uma representação visual da estrutura e das relações entre classes em um sistema de software. No contexto deste sistema, várias classes foram identificadas, cada uma desempenhando um papel essencial na representação dos elementos centrais. As classes incluídas neste diagrama são as seguintes:

- **Endereço:** A classe "Endereço" é responsável por representar as informações referentes a um endereço físico ou de correspondência, incluindo detalhes como rua, cidade, estado.
- **Fornecedor:** A classe "Fornecedor" representa entidades que fornecem os produtos. Contem informações relevantes, como nome, informações de contato e detalhes de produtos fornecidos.
- **Produto:** A classe "Produto" descreve os atributos e comportamentos associados a produtos específicos. Inclui detalhes como nome, descrição, preço e outras propriedades distintivas.
- **Categoria:** A classe "Categoria" representa as categorias às quais os produtos podem pertencer. Ela contém atributos que descrevem o tipo ou a classificação dos produtos associados.
- **Paginação:** A classe "Paginação" pode ser responsável por gerir a exibição de informações de

forma paginada em uma interface de usuário, permitindo a navegação entre diferentes páginas de resultados.

- **Venda:** A classe "Venda" representa transações de venda que podem ocorrer no sistema. Inclui informações sobre os produtos vendidos, o cliente envolvido, o valor total da venda e detalhes de pagamento.
- **Item Encomenda:** A classe "Item Encomenda" descreve os itens específicos incluídos em uma encomenda ou transação.
- **Encomenda:** A classe "Encomenda" representa um pedido de compra ou transação de produtos. Ela tem informações sobre os itens encomendados, detalhes de envio, status da entrega e outras informações pertinentes.
- **Colaborador:** A classe "Colaborador" representa os membros da equipe ou do pessoal envolvidos no sistema. Inclui detalhes como nome, cargo, informações de contacto e outras informações relacionadas à gestão de pessoal.
- **Utilizador:** A classe "Utilizador" representa os utilizadores do sistema. Ela abrange informações como nome de utilizador, palavra-passe, permissões, e outras informações de perfil.

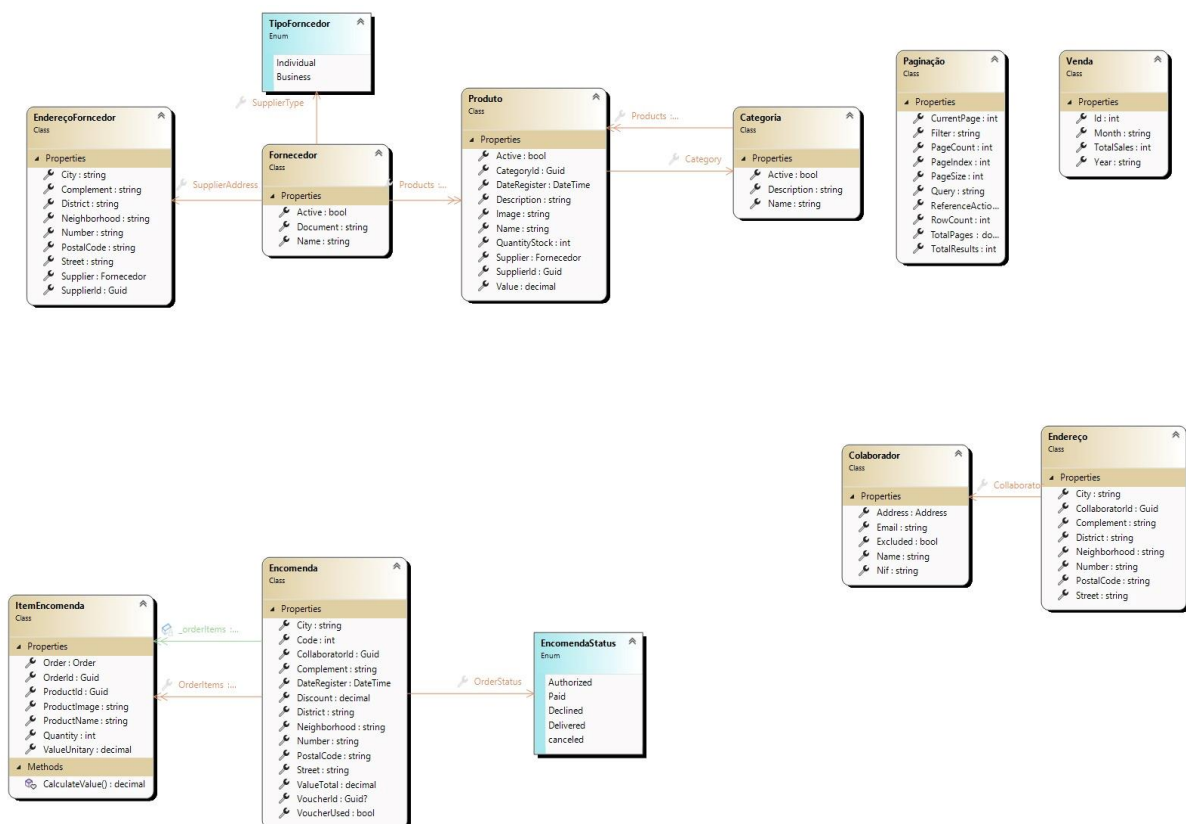


Figura 9 Diagrama de classes

Este diagrama de classes oferece uma visão abrangente das entidades principais e das relações entre elas, proporcionando uma base sólida para o desenvolvimento e a compreensão do sistema de *software* em questão.

3.7 Arquitetura da Solução

A solução foi desenvolvida em C#, usando a base de dados MySQL com recurso a 1 servidor que é o servidor MySQL.

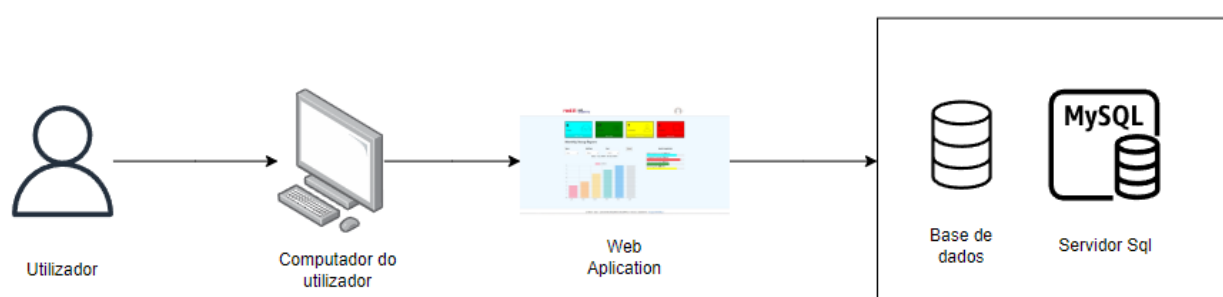


Figura 10 Diagrama da Solução

Como é visível no diagrama apresentado na figura 10, o utilizador através de um computador acede à aplicação onde esta se encontra. Toda a informação que o utilizador inserir deverá ser guardada na base de dados *localhost* que é a base de dados que o utilizador terá no seu computador.

4 Tecnologias e Software utilizado

Neste capítulo, é apresentado as tecnologias e software adotados para este projeto. Cada escolha foi feita com base nos requisitos do projeto, visando à eficiência do desenvolvimento.

É explorado as tecnologias-chave, incluindo linguagens de programação, bibliotecas e ferramentas. Segue-se uma análise sucinta das tecnologias e do software que foram essenciais para o desenvolvimento deste projeto.

4.1 Tecnologias

4.1.1 Microsoft Teams

O Microsoft Teams⁵ foi a aplicação de comunicação usada durante o estágio.

Esta aplicação permitiu á nossa equipa efetuar chamadas e realizar videochamadas. Desta forma, melhorou a comunicação e organização contexto de trabalho híbrido.

4.1.2 Visual Studio 2022

Microsoft Visual Studio⁶ é um ambiente de IDE da Microsoft para desenvolvimento de software especialmente dedicado ao .NET Framework e às linguagens Visual Basic, C, C++, C# e F#.

Também é um produto de desenvolvimento na área web, que usa a plataforma do ASP.NET, como websites, aplicativos web, serviços web e aplicativos móveis.

4.1.3 Jira Software

O Jira Software⁷ é uma ferramenta útil para equipas de desenvolvimento de software. Ajuda as equipas a acompanhar problemas, organizar tarefas e colaborar de maneira mais eficiente. O Jira Software é flexível, o que significa que pode ser adaptado para diferentes métodos de trabalho. Além disso, pode ser conectado a outras ferramentas de desenvolvimento, o que facilita o trabalho em equipa. No geral, o Jira Software ajuda as equipas a trabalhar melhor juntas e a concluir projetos de forma mais eficaz.

4.1.4 Git

O Git⁸ é um sistema de controlo de versão, *open source*, desenvolvido pelo criador do kernel Linux¹², na qual permitiu a monitorização do desenvolvimento do projeto e no controlo de versões.

Esta ferramenta permite ver o código de outros colaboradores, as modificações feitas e os históricos. Possibilitando assim um maior controlo sobre os projetos e os colaboradores, diminuindo assim possíveis falhas.

4.1.5 Sql Server Manager

Durante o projeto foi utilizado o SSMS⁹. O SSMS é um ambiente integrado que desempenhou um

⁵ Disponível em <https://www.microsoft.com/pt-pt/>

⁶ Disponível em <https://visualstudio.microsoft.com/>

⁷ Disponível em <https://www.atlassian.com/software/jira>

⁸ Disponível em <https://github.com/>

⁹ Disponível em <https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>

papel crucial na gestão da infraestrutura SQL ao longo do projeto.

O SSMS oferece um conjunto abrangente de ferramentas que nos permitiram configurar, monitorar e administrar tanto instâncias do SQL Server de bases de dados. Esta versatilidade foi fundamental para manter um controle preciso e eficaz sobre as operações relacionadas aos dados.

Além disso, o SSMS possibilitou a implantação, monitorização e atualização dos componentes da camada de dados usados pela nossa aplicação. Isso revelou-se vital para garantir a consistência e a confiabilidade dos dados em todo o ciclo de vida do projeto.

4.1.6 Net 6

No contexto deste projeto, a plataforma .NET¹⁰ desempenhou um papel fundamental no desenvolvimento da aplicação. O .NET oferece um ambiente de desenvolvimento robusto e versátil que abrange uma variedade de tecnologias e ferramentas. Aqui estão algumas maneiras pelas quais o .NET foi utilizado:

- Desenvolvimento *Back-end*.
- ASP.NET.
- Bibliotecas e *Frameworks*.
- Integração com Outras Tecnologias Microsoft.
- Segurança.
- Testabilidade.

4.2 Linguagens

4.2.1 Back-end

- C# foi escolhido como a principal linguagem de programação para o *back-end*, a fim de construir lógica do servidor robusta e escalável.
- SQL foi utilizado para interagir com o banco de dados, gerenciar dados e executar consultas.

4.2.2 Front-end

- HTML, CSS e JavaScript estas tecnologias foram utilizadas para criar a interface do utilizador e aprimorar a sua experiência.

¹⁰ Disponível em <https://dotnet.microsoft.com/en-us/>

- O *framework Bootstrap* foi empregue para implementar o design responsivo e a estilização consistente em diferentes dispositivos e tamanhos de tela.

4.3 Logging

A biblioteca NLog foi utilizada para fazer o registo e rastreamento das atividades da aplicação. Os registos foram categorizados com base no controlador, nível de log e data de criação para facilitar a análise e depuração.

4.4 Validação

O *FluentValidation* foi escolhido para lidar com a validação do lado do servidor, garantindo que a aplicação mantenha a integridade da validação mesmo quando o JavaScript está desativado no navegador. Esta biblioteca oferece uma abordagem de validação mais robusta e personalizável em comparação com *DataAnnotations*.

4.5 Arquitetura e Padrões

O padrão arquitetural MVVC foi adotado para criar uma clara separação de responsabilidades entre a interface do utilizador (*View*), lógica de negócio e dados (*Model*) e o conversor intermediário de dados (*ViewModel*).

O *AutoMapper* foi utilizado para agilizar o processo de conversão entre os *models* e *view models*. Ele mapeia automaticamente as propriedades entre objetos correspondentes, reduzindo a necessidade de mapeamento manual e repetitivo.

O padrão *Repository* foi empregue para abstrair a camada de acesso a dados do restante da aplicação. Essa separação permitiu uma manutenção, testes e escalabilidade mais fáceis.

5 Implementação

5.1 C# em 3 camadas

Neste ponto do relatório está descrita a arquitetura típica de 3 camadas no qual este projeto foi feito.¹¹

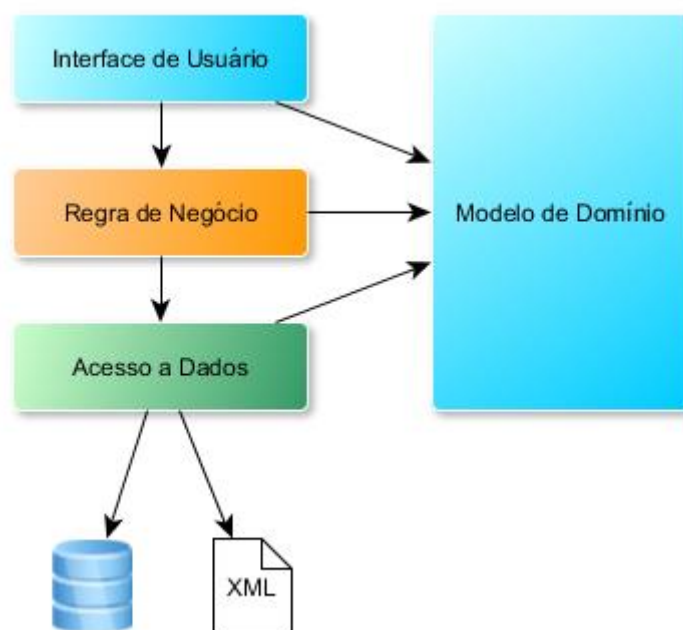


Figura 11 Arquitetura em camadas

O projeto utilizou uma arquitetura típica de três camadas, com a separação das camadas de apresentação, lógica de negócio e acesso a dados. Essa estrutura organizada permitiu uma melhor manutenção e escalabilidade, facilitando o desenvolvimento e a implementação de novas funcionalidades.

A arquitetura de três camadas contribuiu para a organização eficiente do código e a facilidade de manutenção, garantindo a escalabilidade necessária para atender a um crescimento futuro deste projeto.

5.2 Base de Dados da Solução

Na figura 12 encontram-se o diagrama ER implementado na base de dados.

A estrutura da base de dados é otimizada para garantir a consistência, a segurança e o desempenho ideal do sistema. A escolha de um sistema de gestão de base de dados adequado é crucial para garantir

¹¹ <https://felipelucioquirino.wordpress.com/2012/07/20/arquitetura-de-software-arquitetura-em-camadas/>

a escalabilidade e a eficiência do armazenamento de dados, proporcionando uma base robusta para o funcionamento ideal do sistema.

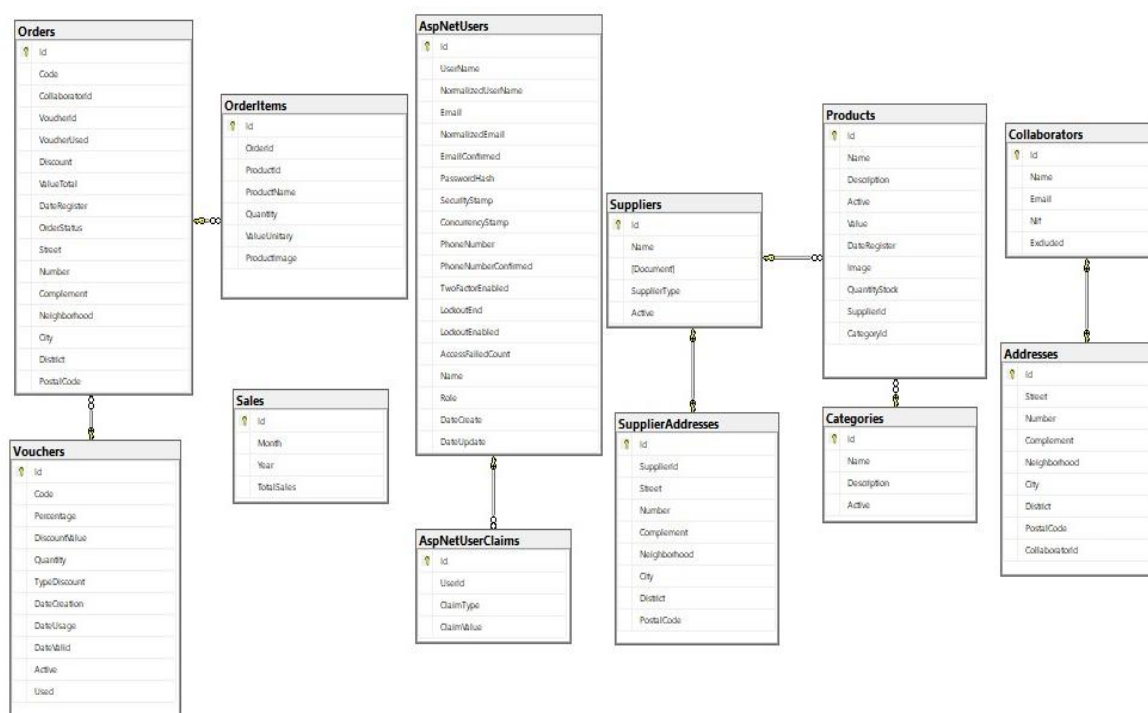
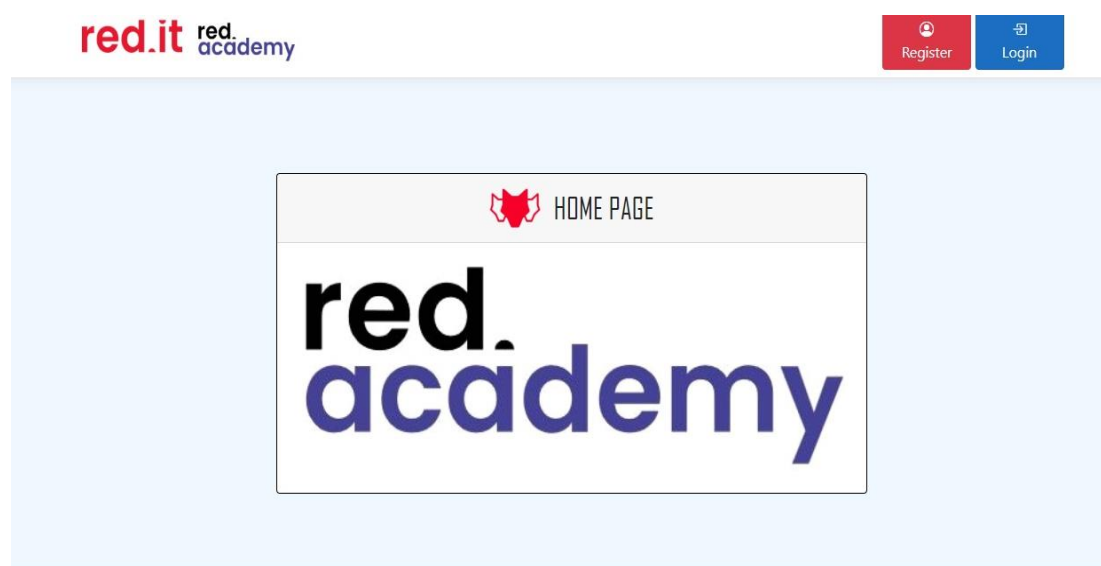


Figura 12 Base de dados

5.3 Descrição da aplicação

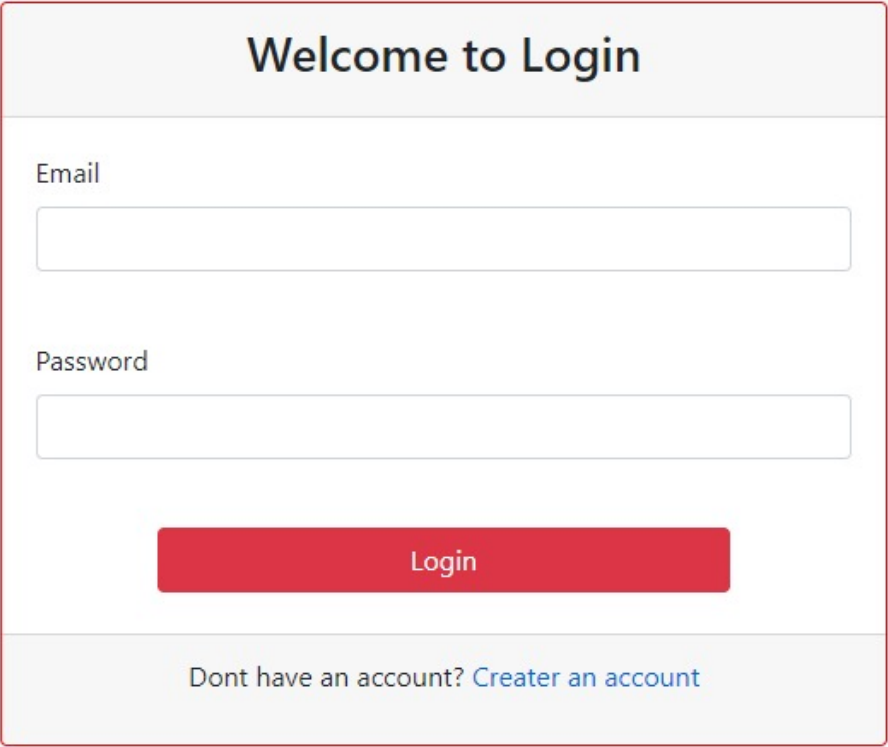


© REDIT - 2023 - License: RED ACADEMY | BACKOFFICE - Version: 1.20230319.1 - itsupport@redit.pt

Figura 13 Página Inicial

5.3.1 Janela de login

Nesta janela o utilizador poderá fazer o login para poder começar a usufruir da aplicação, para isso precisa de preencher o *username* e a *password*, como mostra na figura 14.



The image shows a login form titled "Welcome to Login". It contains two input fields: "Email" and "Password". Below the "Password" field is a red "Login" button. At the bottom of the form, there is a link that says "Dont have an account? [Creater an account](#)".

Figura 14 Autenticação

Para que o utilizador posso fazer *login* é necessário que o *username* exista na base de dados, e no campo *password* será recebida a *password* encriptada, que terá de ser posteriormente desencriptada.

5.3.2 Janela de área Inicial

Nesta janela apresenta ao utilizador os menus por onde poderá a navegar pelas várias vertentes da aplicação, como se verifica na figura 15.

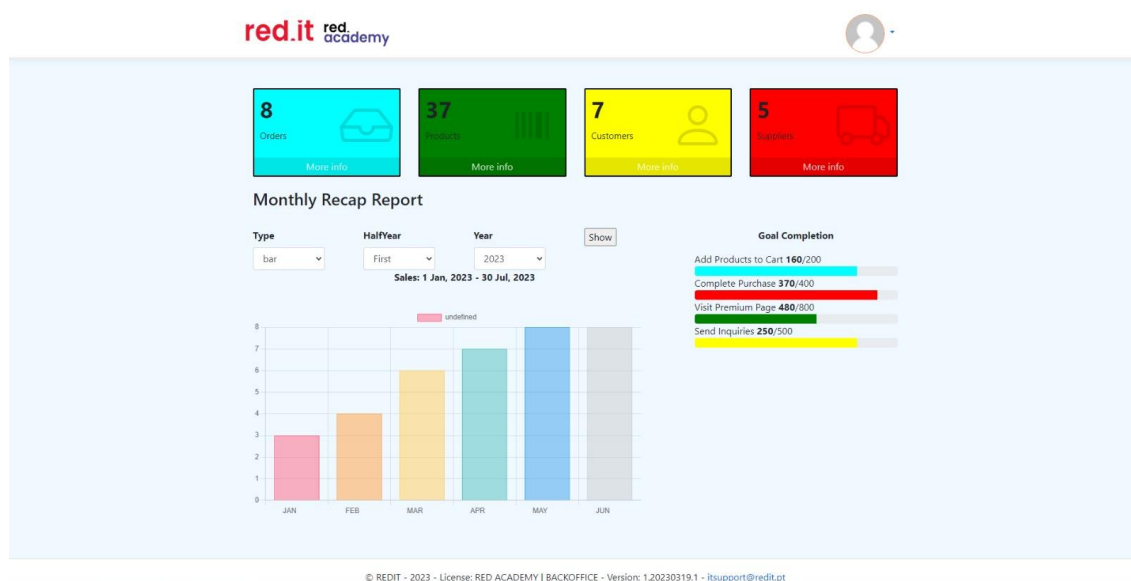


Figura 15 Dashboard

Nos botões apresentados do ecrã do *dashboard* é possível ver o número de pedidos, produtos, clientes e fornecedores respetivamente.

É possível também navegar para as respetivas listas através destes botões.

5.3.3 Janela das listas

Nesta janela o utilizador terá uma informação superficial das listas, sendo elas produtos, categorias, fornecedores, ordens, clientes e utilizadores.

O utilizador só terá acesso se tiver as respetivas *claims*.

5.3.4 Janela das listas produtos

Nesta janela o utilizador ver todos os produtos existentes. Para ajudar a pesquisa do produto em questão ele pode filtrar os produtos pelo nome, categoria ou fornecedor.

Nesta janela pode também adicionar, editar, eliminar e ver os detalhes, se tiver as permissões necessárias.

LIST OF PRODUCT

Add Name Search...

Image	Name	Category	Supplier	Value	Stock	Active	Action
	Mug Batman Preta	Clothes	Microsoft Xfsd	50.00 XDR	8 unit(s)	☒	
	Mug Big Bang Theory	Clothes	Microsoft Xfsd	50.00 XDR	0 unit(s)	☒	
	Mug Cogumelo	Clothes	Microsoft Xfsd	50.00 XDR	10 unit(s)	☒	
	Mug Geeks	Clothes	Microsoft Xfsd	50.00 XDR	10 unit(s)	☒	
	Mug Ironman	Clothes	Microsoft Xfsd	50.00 XDR	10 unit(s)	☒	

1 2 3 4 5 6 7 8

Figura 16 Lista de produtos

5.3.5 Janela das listas fornecedores

Nesta janela o utilizador ver todos os fornecedores existentes. Para ajudar a pesquisa do fornecedor em questão ele pode filtrar pelo nome e número do documento.

Nesta janela pode também adicionar, editar, eliminar e ver os detalhes, se tiver as permissões necessárias.

LIST OF SUPPLIER

Add Name Search...

Name	Document	Supplier Type	Active	Action
ABC Company	654321987	Business	☐	
ACME Corporation	123456789	Business	☒	
AWS X	222222222	Business	☒	
John Doe	987654321	Business	☒	
Microsoft	11111111111	Individual	☒	

1 2

Figura 17 Lista de fornecedores

5.3.6 Janela das listas categorias

Nesta janela o utilizador ver todas as categorias existentes. Para ajudar a pesquisa da categoria em questão ele pode filtrar pelo nome e descrição.

Nesta janela pode também adicionar, editar, eliminar e ver os detalhes, se tiver as permissões necessárias.

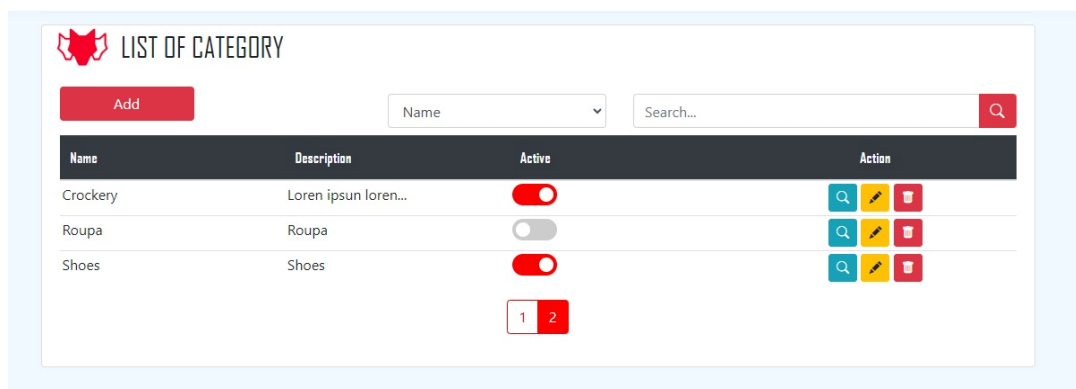


Figura 18 Lista de categorias

5.3.7 Janela das listas clientes

Nesta janela o utilizador ver todos os clientes existentes. Para ajudar a pesquisa do cliente em questão ele pode filtrar os clientes pelo nome, email ou NIF.

Nesta janela pode também adicionar, editar, eliminar e ver os detalhes, se tiver as permissões necessárias.

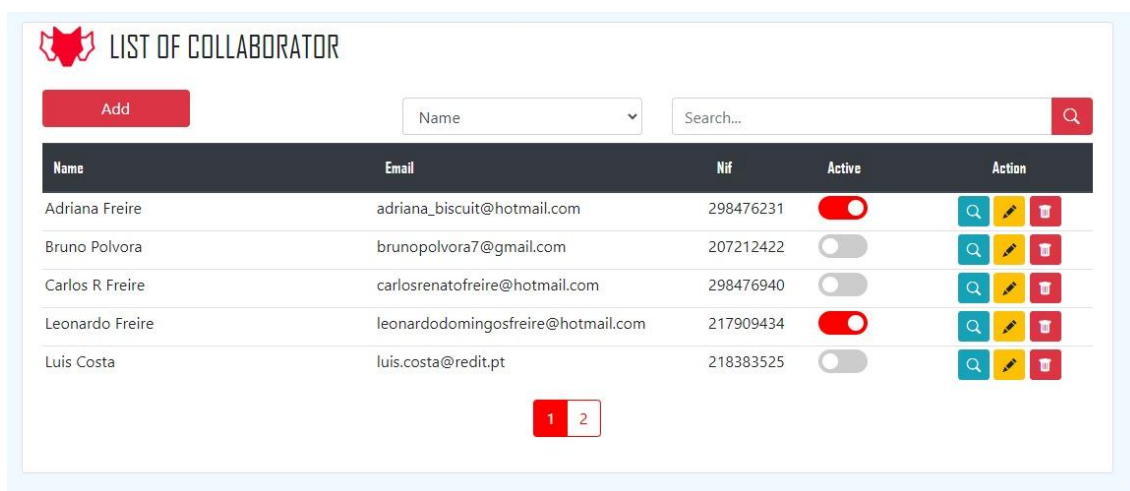
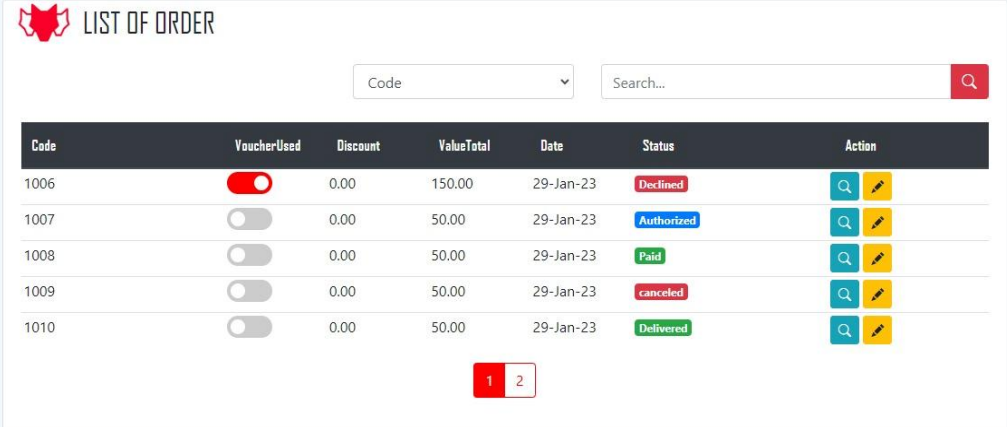


Figura 19 Lista de colaboradores

5.3.8 Janela das listas ordens

Nesta janela o utilizador ver todos os pedidos existentes. Para ajudar a pesquisa dos pedidos em questão ele pode filtrar os produtos pelo código ou o valor total.

Nesta janela pode também editar e ver os detalhes, se tiver as permissões necessárias.



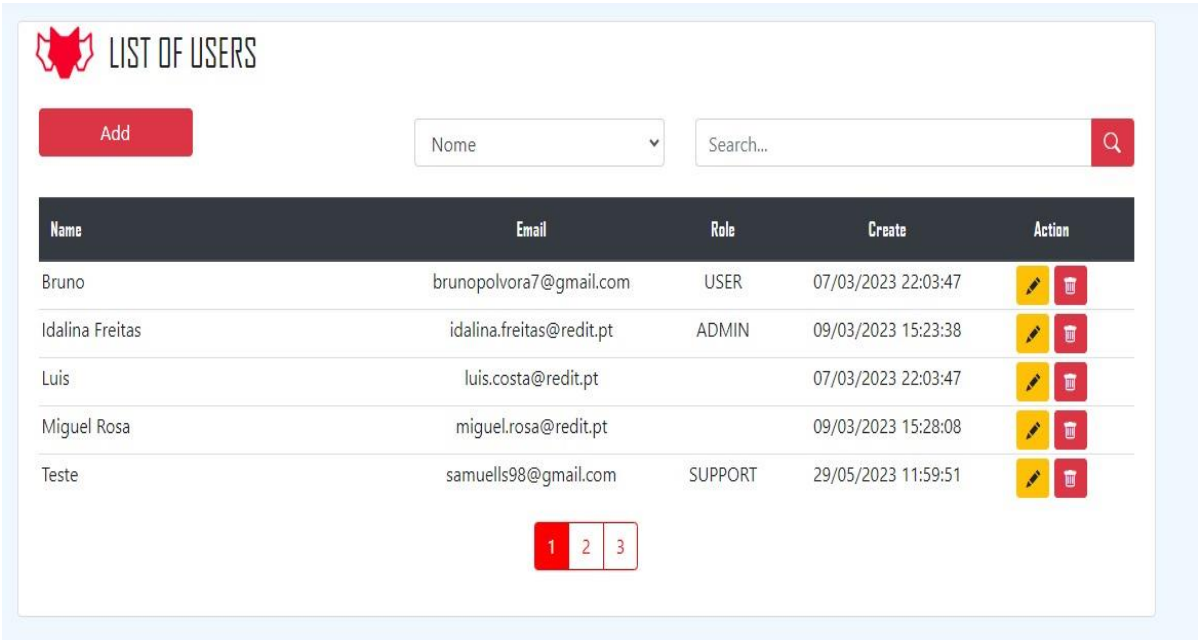
Code	VoucherUsed	Discount	ValueTotal	Date	Status	Action
1006	☒	0.00	150.00	29-Jan-23	Declined	 
1007	☐	0.00	50.00	29-Jan-23	Authorized	 
1008	☐	0.00	50.00	29-Jan-23	Paid	 
1009	☐	0.00	50.00	29-Jan-23	canceled	 
1010	☐	0.00	50.00	29-Jan-23	Delivered	 

Figura 20 Lista de pedidos

5.3.9 Janela das listas utilizadores

Nesta janela o utilizador ver todos os utilizadores existentes. Para ajudar a pesquisa do utilizador em questão ele pode filtrar os utilizadores pelo nome ou email.

Nesta janela pode também adicionar, editar e eliminar, se tiver as permissões necessárias.



Name	Email	Role	Create	Action
Bruno	brunopolvora7@gmail.com	USER	07/03/2023 22:03:47	 
Idalina Freitas	idalina.freitas@redit.pt	ADMIN	09/03/2023 15:23:38	 
Luis	luis.costa@redit.pt		07/03/2023 22:03:47	 
Miguel Rosa	miguel.rosa@redit.pt		09/03/2023 15:28:08	 
Teste	samuells98@gmail.com	SUPPORT	29/05/2023 11:59:51	 

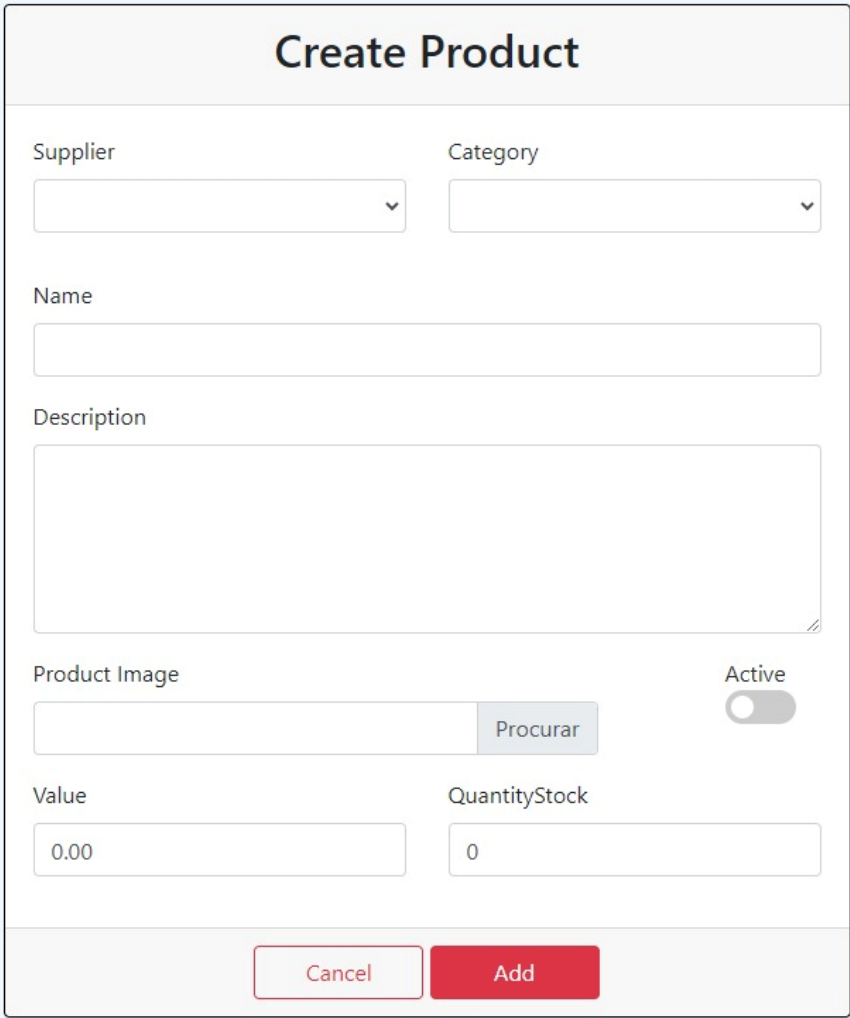
1 2 3

Figura 21 Lista de utilizadores

5.3.10 Janela inserir produto

Esta janela é acedida através do botão de adicionar da listagem dos produtos. Aqui o utilizador poderá inserir o seu novo produto, sendo obrigado a preencher os campos obrigatórios que terão de ser aceites de acordo com as validações.

Na figura 25 é visível como é a janela e os dados que devem ser preenchidos.



The image shows a web form titled "Create Product". It contains several input fields: "Supplier" and "Category" are dropdown menus; "Name" is a text input; "Description" is a large text area; "Product Image" is a text input with a "Procurar" (Search) button next to it; "Value" is a text input with "0.00" pre-filled; and "QuantityStock" is a text input with "0" pre-filled. There is also an "Active" toggle switch. At the bottom, there are "Cancel" and "Add" buttons.

Create Product	
Supplier	Category
Name	
Description	
Product Image	Active
Procurar	☐
Value	QuantityStock
0.00	0
Cancel Add	

Figura 22 Criação de produto

5.3.11 Janela editar produto

Esta janela é acedida através do botão de editar da lista dos produtos.

Aqui o utilizador pode editar os dados do produto sendo obrigado a preencher os campos obrigatórios que terão de ser aceites de acordo com as validações. Nesta janela serão carregados os dados que o utilizador preencheu ao criar um produto que se encontra na figura 23.

Edit Product

Supplier

Microsoft Xfsd

Category

Clothes

Name

Mug Batman Preta

Active

☒

Description

Porcelain mug with high resistance thermal printing.

Current Image



Product Image

Search

Value

50.00

Quantity/Stock

8

Cancel

Edit

Figura 23 Edição produto

5.3.12 Janela eliminar produto

Esta janela é acedida através do botão de eliminar da lista dos produtos.

Aqui o utilizador pode eliminar os dados do produto. Nesta janela serão carregados os dados que o utilizador preencheu ao criar um produto que se encontra na figura 24.

DELETE PRODUCT



MINI

135ml

Are you sure you want to delete?

Supplier
Microsoft Xfsd

Name
Mug Batman Preta

Description
Porcelain mug with high resistance thermal printing.

Value
50.00 XDR

DateRegister
19/07/2019 00:00:00

Active

☒

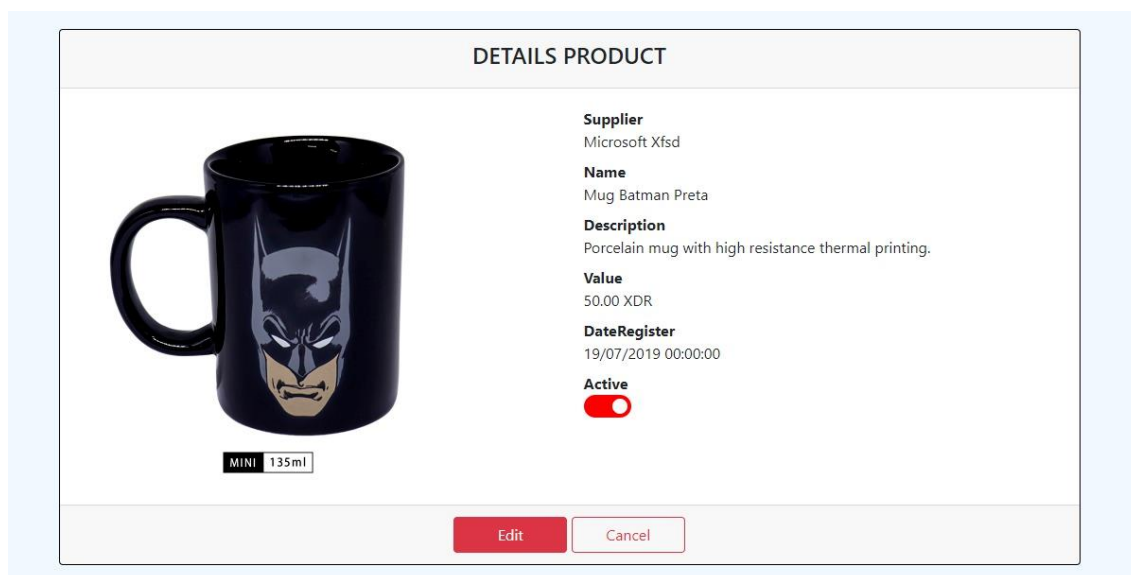
Delete

Back

Figura 24 Eliminação do produto

5.3.13 Janela detalhes produto

Esta janela é acedida através do botão de detalhes da lista dos produtos, se o utilizador quiser pode editar o produto se carregar no botão de editar.

*Figura 25 Detalhes de produto*

5.3.14 Janela detalhes fornecedores

Esta janela é acedida através do botão de detalhes da lista dos fornecedores.

Aqui o utilizador pode ver os dados dos fornecedores, é possível também consultar dos dados do endereço desse mesmo fornecedor. Se utilizador desejar também pode editar os dados do seu endereço.

Details Supplier

Name: ABC Company
Supplier Type: Business
Active: ☐

Document: 654321987

ADDRESS

Street	Number	Complement	Neighborhood	PostalCode	City	District
adress	11	Complement	Azoia	11111111	Lisboa	Lisboa

Edit
Cancel

Figura 26 Detalhes do fornecedor

5.3.15 Janela editar fornecedor

Esta janela é acedida através do botão de editar da lista dos fornecedores.

Aqui o utilizador pode editar os dados dos fornecedores, é possível também consultar dos dados do endereço desse mesmo fornecedor. Se utilizador desejar também pode editar os dados do seu endereço. Nesta janela o utilizador também acesso aos produtos que estão associados ao fornecedor selecionado, onde pode aceder, editar ou até mesmo eliminar esses produtos.

Edit supplier

Name

Document

SupplierType

Individual
▼

Active ☒

Address details

Street	Number	Complement	Neighborhood	PostalCode	City	District
Azoia	2	2 esq	Lisboa	2625-236	Povoa de Santa Iria	Lisboa

Products

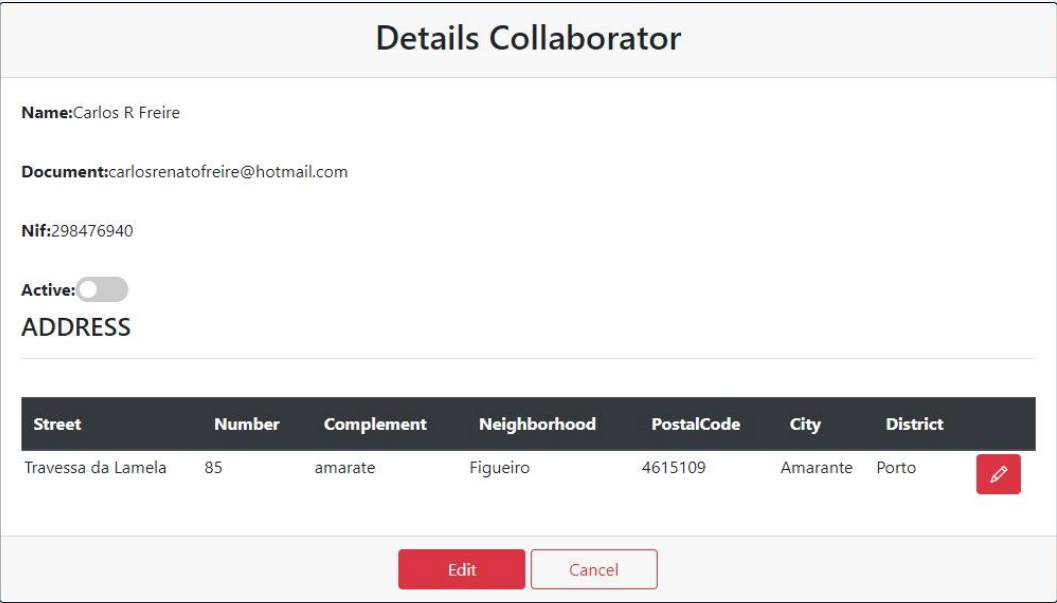
Image	Name	Category	Supplier	Value	Stock	Active	Action
	T-shirt 4 Head	Clothes	Microsoft	50.00 XDR	5 unit(s)	☒	🔍 ✎ 🗑
	T-shirt 4 Head White	Clothes	Microsoft	50.00 XDR	5 unit(s)	☒	🔍 ✎ 🗑
	T-shirt Tiltado	Clothes	Microsoft	50.00 XDR	10 unit(s)	☒	🔍 ✎ 🗑

Figura 27 Edição do fornecedor

5.3.16 Janela detalhes colaborador

Esta janela é acedida através do botão de detalhes da lista dos colaboradores.

Aqui o utilizador pode ver os dados dos colaboradores, é possível também consultar dos dados do endereço desse mesmo fornecedor. Se utilizador desejar também pode editar os dados do seu endereço.



The screenshot shows a web application window titled "Details Collaborator". It contains the following fields and controls:

- Name:** Carlos R. Freire
- Document:** carlosrenatofreire@hotmail.com
- Nif:** 298476940
- Active:** A toggle switch is currently turned off.
- ADDRESS:** A section containing a table with address details.

Street	Number	Complement	Neighborhood	PostalCode	City	District
Travessa da Lamela	85	amarate	Figueiro	4615109	Amarante	Porto

At the bottom of the window, there are two buttons: "Edit" (highlighted in red) and "Cancel". A small red edit icon is also visible next to the address table.

Figura 28 Detalhes do colaborador

5.3.17 Janela detalhes encomenda

Esta janela é acedida através do botão de detalhes da lista das encomendas.

Aqui o utilizador pode visualizar os dados das encomendas, é possível também consultar dos dados do endereço da encomenda selecionada. Se utilizador desejar também pode editar os dados do seu endereço.

Nesta janela o utilizador também tem acesso aos produtos que pertencem a encomenda, onde pode aceder, editar ou até mesmo eliminar esses produtos.

Details Order

Code:1006

OrderStatus: Declined

Value Total:150.00

Discount:0.00

VoucherUsed: ☐

Address

Street	Number	Neighborhood	PostalCode	City	District
Travessa da Lamela	85	Figueiro	4615109	Amarante	Porto

Order Items

ProductImage	ProductName	Quantity	Value	Action
	Mug Batman Preta	2 unit(s)	50,00 XDR	  
	Mug Batman	1 unit(s)	50,00 XDR	  

Edit

Cancel

Figura 29 Detalhes do pedido

5.3.18 Janela editar encomenda

Esta janela é acedida através do botão de editar da lista das encomendas.

Aqui o utilizador pode editar os dados das encomendas, é possível também editar dos dados do endereço da encomenda selecionada. Se utilizador desejar também pode editar os dados do seu endereço.

Nesta janela os utilizadores também têm acesso aos produtos que pertencem a encomenda, onde pode aceder, editar ou até mesmo eliminar esses produtos.

Edit Order

Code: OrderStatus: ☐ Authorized ☐ Paid ☒ Declined ☐ Delivered ☐ Canceled

ValueTotal: Discount: Voucher Used: ☒

Address

Street: Number: Complement: Neighborhood: City: District: PostalCode:

Order Items

ProductImage	ProductName	Quantity	Value	Action
	Mug Batman Preta	2 unit(s)	50.00 XDR	  
	Mug Batman	1 unit(s)	50.00 XDR	  

Figura 30 Edição do pedido

5.3.19 Janela criar utilizador

Esta janela é acedida através do botão de adicionar da listagem dos utilizadores. Aqui o utilizador poderá inserir o seu novo utilizador, sendo obrigado a preencher os campos obrigatórios que terão de ser aceites de acordo com as validações.

Na figura 31 é visível como é a janela e os dados que devem ser preenchidos.

CREATE USER

Name: Email:

Role: Password: Confirm the Password:

Figura 31 Criação do utilizador

5.3.20 Janela editar utilizador

Esta janela é acedida através do botão de editar da listagem dos utilizadores. Aqui o utilizador poderá

editar o utilizador seleccionado, sendo obrigado a editar os campos obrigatórios que terão de ser aceites de acordo com as validações.

Na figura abaixo é possível observar uma *partial view* com os dados das *claims* e permissões.

EDIT USER

Name: UserName: Email:

Role: Password: PasswordConfirmation:

Assign Permissions by Claims:

Modules	Permissions
Category	List,View,Add,Edit,Delete
Collaborator	List,View,Add,Edit,Delete
Dashboard	List,
Order	List,View,Add,Edit,Delete
Product	List,View,Add,Edit,Delete
Supplier	List,View,Add,Edit,Delete
User	List,View,Add,Edit,Delete

[Manage Claims](#)

[Save](#) [Back](#)

Figura 32 Edição do utilizador

5.3.21 Janela gerir claims do utilizador

Após ter sido permitido o botão de editar as *claims* aparece uma modal onde através de várias *checkboxes* é possível seleccionar as *claims* e as permissões desejadas.

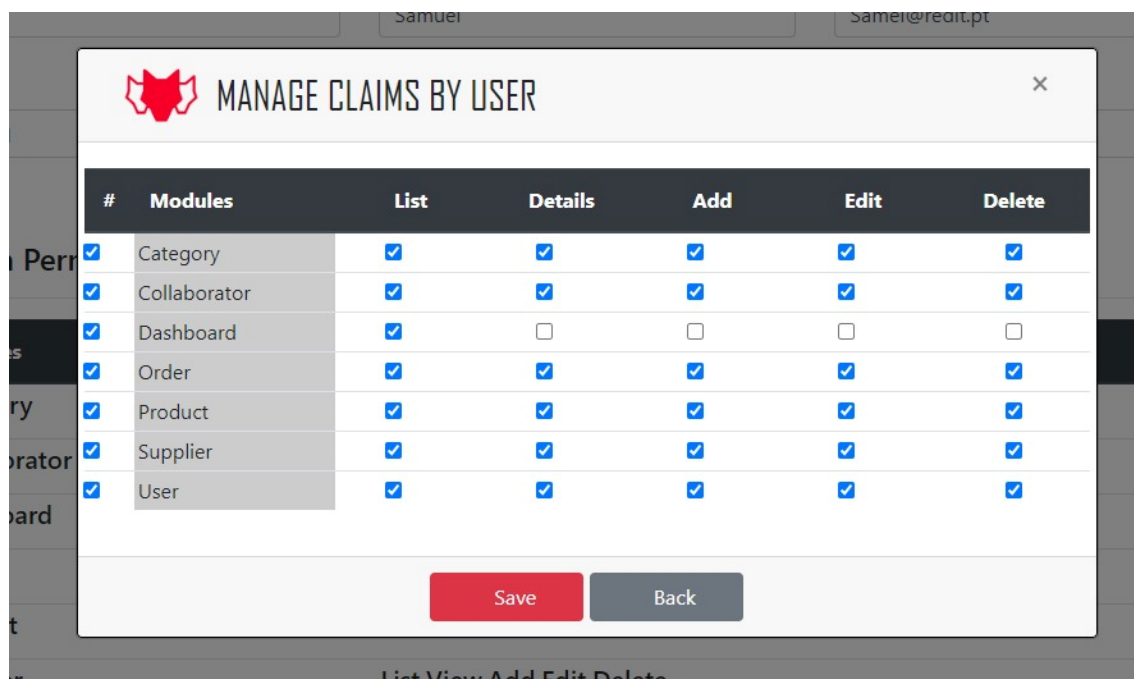


Figura 33 Edição do utilizador

5.3.22 Janela eliminar utilizador

Após o utilizador carregar no botão de eliminar será apresentada uma modal onde o utilizador toma a decisão se pretende eliminar o utilizador.

Na *modal* é apresentada o email e imagem do utilizador.

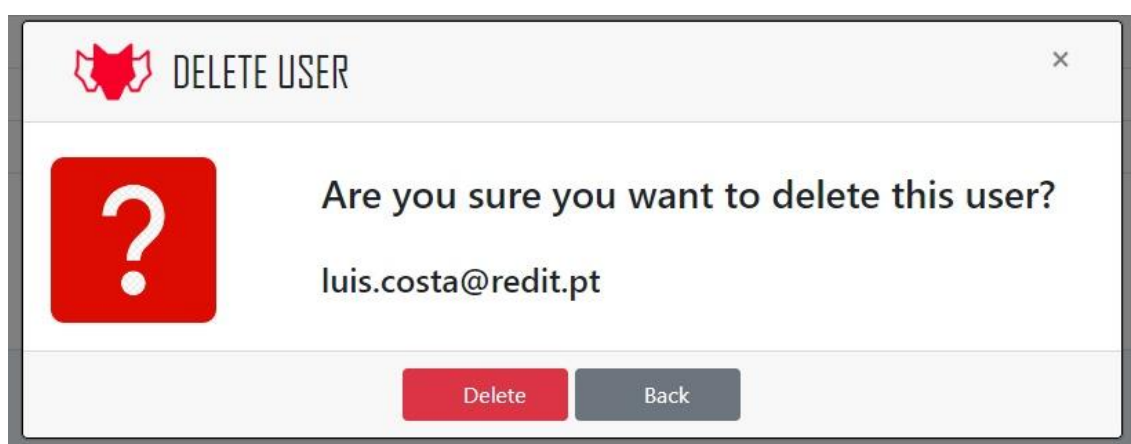


Figura 34 Eliminação do utilizador

6 Conclusão

Neste capítulo, abordarei as diversas motivações e elementos que me impulsionaram que me levaram a realizar este projeto, os desafios que enfrentei durante sua execução, bem como uma visão dos trabalhos futuros e potenciais ajustes a serem considerados.

Por fim, apresentarei minhas conclusões finais em relação ao estágio curricular.

6.1 Trabalho desenvolvido

A conclusão bem-sucedida do projeto desenvolvido demonstrou a eficaz utilização de várias tecnologias, bibliotecas e padrões.

Neste relatório, apresento uma visão abrangente do trabalho que realizei até o momento. O projeto em questão envolveu o desenvolvimento de uma aplicação *web back-office* para fornecer suporte a um *front-end* de uma loja online.

Utilizei uma variedade de tecnologias, incluindo C# para o *back-end*, MySQL para a base de dados e HTML, CSS e JavaScript para o *front-end*. Implementei bibliotecas como *Bootstrap* para aprimorar o design da interface do utilizador, *NLog* para registos detalhados e *FluentValidation* para garantir a validação robusta dos dados.

Além disso, foi adotada uma arquitetura em três camadas para garantir a organização eficaz do código e a escalabilidade da aplicação. O uso de padrões como MVVC com *AutoMapper* e o padrão *Repository* contribuiu para uma estrutura de código sólida e modular.

6.2 Principais Obstáculos

A incorporação de tecnologias inovadoras, especialmente no campo de *back-end*, resultou numa fase inicial mais demorada, dado que se tratava de uma área onde eu tinha pouca experiência.

Entretanto, esse obstáculo inicial foi superado por meio de pesquisa e realização de cursos online.

Durante a realização deste projeto foram aparecendo algumas dificuldades tais como trabalhar em 3 camadas, as validações dos campos, ligação a base de dados, utilização de bibliotecas novas, procedimentos em MySQL, carregamento e inserção de imagens na base de dados. Contudo no decorrer da realização do projeto foi se tornando cada vez mais fácil a implementação do código da aplicação.

A segurança dos dados também foi uma preocupação, dada a sua natureza sensível.

Foram implementadas medidas rigorosas para proteger os dados e garantir a privacidade e a integridade das informações.

6.3 Trabalho Futuro

Olhando para o futuro, vejo várias oportunidades para melhorias e expansões. Um aspecto que merece ser referido é a parte visual do projeto. Reconheço a importância de uma interface de utilizador atraente e intuitiva. É necessário realizar atualizações no design e na experiência do usuário para garantir que a aplicação permaneça moderna e agradável de usar. Essas melhorias visuais não apenas aprimorarão a estética, mas também aumentarão a usabilidade e a satisfação geral dos usuários.

6.4 Considerações Finais

Para alcançar o êxito final, foi imperativo dominar e aplicar uma diversidade de conceitos assimilados ao longo do curso, conhecimentos em programação `c#`, medidas de segurança, medidas de Escalabilidade e Eficiência e outras abordagens. Além disso, esta jornada possibilitou o primeiro contato prático com metodologias e técnicas que, até então, haviam sido apenas estudadas na teoria. Os objetivos que foram estabelecidos proporcionaram uma valiosa experiência no desenvolvimento de projetos dentro de um contexto empresarial. Ao mesmo tempo, esta jornada resultou na aquisição de um repertório sólido de conhecimentos e na incorporação de boas práticas de programação. Em última análise, posso afirmar com confiança que os propósitos do estágio foram alcançados. Toda a experiência adquirida durante esse período contribuiu de forma significativa para enriquecer meu conhecimento e expandir minha experiência profissional.

Referências

ASP. NET Core MVC. (14 de 10 de 2023). Obtido de Microsoft Docs: <https://learn.microsoft.com/en-us/aspnet/core/mvc/overview?view=aspnetcore-6.0>

Automapper. (18 de 09 de 2023). Obtido de Pro Code Guide: <https://procodeguide.com/programming/automapper-in-aspnet-core/>

Bootstrap. (16 de 9 de 2023). Obtido de Bootstrap Docs: <https://getbootstrap.com/docs/5.1/getting-started/introduction/>

C# documentation. (2 de 9 de 2023). Obtido de Microsoft: <https://learn.microsoft.com/en-us/dotnet/csharp/>

Diagrams .net. (19 de 9 de 2023). Obtido de draw.io: <https://app.diagrams.net/>

FluentValidation. (24 de 8 de 2023). Obtido de FluentValidation: <https://docs.fluentvalidation.net/en/latest/>

Microsoft Docs. (14 de 11 de 2023). Obtido de ASP. NET Core : <https://learn.microsoft.com/en-us/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-6.0>

MySQL. (6 de 9 de 2023). Obtido de MySQL Documentation: <https://dev.mysql.com/doc/>

Red It. (05 de 08 de 2023). Obtido de Red IT.

Red It. (5 de 8 de 2023). Obtido de Red It: <https://www.redit.pt/en/about-us>

Three Layer Architecture in C# .NET. (5 de 9 de 2023). Obtido de Code Project: <http://www.codeproject.com/Articles/36847/Three-Layer-Architecture-in-C-NET>

What Is Agile Methodology? (7 de 9 de 2023). Obtido de write: <https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/>