



Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia Informática

Unidade Curricular de Laboratórios de Informática IV

Ano Letivo de 2024/2025

<< **ComplicadaMente** >>

Tiago Matos Guedes(A97369)

28 de julho de 2025

LI4

Data de Receção	
Responsável	
Avaliação	
Observações	

<<**ComplicadaMente**>>

Tiago Matos Guedes(A97369)

28 de julho de 2025

Resumo

Este projeto surgiu inspirado pela paixão de José Pinto pelos quebra-cabeças, algo que começou na sua infância quando foi desafiado pela primeira vez pela "Estrela de Galileu". Essa paixão culminou a abertura da sua loja de quebra-cabeças artesanais, a "ComplicadaMente", em Lomar, Braga. A loja numa fase inicial apenas atraía público mais jovem, mas rapidamente o interesse pelos puzzles cresceu, abrangendo também adultos que viam os quebra-cabeças como forma de desenvolver as suas capacidades mentais. Com o aumento do negócio, a produção dos puzzles expandiu e chegou aos filhos de José e á criação de várias lojas pelo país. O desenvolvimento do projeto focou-se na criação de uma aplicação que visa preservar o legado de José Pinto e a evolução do seu negócio. Durante a fase inicial, foram definidos os principais requisitos da aplicação para posterior desenvolvimento que consistirá em que os utilizadores consigam navegar por uma galeria de puzzles/quebra-cabeças, consigam fazer as suas encomendas ou até mesmo encomendar possíveis peças caso por ventura se percam ou partam. Para garantir a robustez foi criado um modelo de domínio do projeto, bem como os seus use-cases e o próprio modelo, facilitando assim o processo de desenvolvimento do projeto. Atualmente o projeto encontra-se em fase final, com todas as funcionalidades cruciais implementadas e um protótipo funcional. O próximo passo seria melhorar com base no feedback que vai chegando por parte dos utilizadores, para finalmente ser lançada oficialmente a aplicação.

Área de Aplicação: Desenho e arquiteturas de sistemas informáticos.

Palavras-Chave: Sistemas de Gestão de Linha de Montagem, Desenvolvimento de Sistemas de Software, Visual Paradigm, Gestão de Base de Dados, UML.

Índice

1	Introdução	1
1.1	Contextualização	1
1.2	Motivação e Objectivos	1
1.3	Justificação, utilidade e viabilidade do sistema	2
1.4	Estabelecimento da identidade do projeto	3
1.5	Identificação dos recursos necessários	4
1.5.1	Recursos Humanos	4
1.5.2	Recursos Tecnológicos	4
1.5.3	Recursos Físicos	5
1.5.4	Recursos Financeiros	5
1.6	Maquete do sistema	5
1.7	Definição de um conjunto de medidas de sucesso	6
1.8	Plano de desenvolvimento (Diagrama de gantt)	7
2	Levantamento e Análise de Requisitos	8
2.1	Apresentação da estratégia e método	8
2.2	Descrição geral dos Requisitos levantados	9
2.2.1	Requisitos Funcionais	9
2.2.2	Requisitos Não Funcionais	11
2.3	Validação dos requisitos estabelecidos	11
3	Especificação e Modelação do Software	13
3.1	Apresentação geral da especificação	13
3.2	Aspetos Estruturais	14
3.3	Aspetos Comportamentais	15
4	Conceção do Sistema de Dados	29
4.1	Apresentação geral da estrutura do sistema de dados	30
4.2	Descrição detalhada dos vários elementos de dados e seus relacionamentos	31
4.2.1	Utilizador	32
4.2.2	Funcionário	33
4.2.3	Encomenda	34
4.2.4	Peça	35
4.2.5	Quebra-Cabeça	36
4.2.6	Quebra-Cabeça de Encomenda	37
4.2.7	Peça de Encomenda	38

5	Esboço dos Interfaces do Sistema	39
5.1	Caracterização das interfaces	40
5.1.1	Página de Login e Registo do Cliente	40
5.1.2	Página principal	41
5.1.3	Página de resultados	42
5.1.4	Scroll da página de resultados	43
5.1.5	Botão menu	44
5.1.6	Botão user	45
5.1.7	Selecionar produto	46
5.1.8	Carrinho	47
5.1.9	Finalizar Compra - Confirmar produtos	48
5.1.10	Finalizar Compra - Confirmar dados pessoais	49
5.1.11	Finalizar Compra - Selecionar Morada	50
5.1.12	Finalizar Compra - Selecionar Método de pagamento	51
5.1.13	Finalizar Compra - Dados da encomenda	52
5.1.14	Consultar estado de encomenda	53
5.1.15	Consultar produto encomendado	54
5.1.16	Selecionar tipo de funcionário	55
5.1.17	Login do funcionário	56
5.1.18	Página principal do funcionário	57
5.1.19	Passos da produção/montagem	58
5.1.20	Informações adicionais da peça/puzzle	59
5.1.21	Finalização da peça/montagem	60
5.1.22	Página principal do administrador	61
5.1.23	Scroll da página principal do administrador	62
5.1.24	Consultas e alterações de dados	63
6	Implementação da Aplicação	64
6.1	Apresentação e descrição do processo de implementação realizado.	64
6.1.1	Preparação da Implementação	64
6.1.2	Processo de Desenvolvimento	64
6.2	Apresentação da aplicação e explicação dos serviços implementados	65
6.2.1	Criação da Base de Dados	65
6.2.2	Desenvolvimento do back-end e do front-end	65
6.2.3	Login	66
6.2.4	Loja — Visão Utilizador	67
6.2.5	Populares — Visão Utilizador	68
6.2.6	Encomendas do Utilizador — Visão Utilizador	69
6.2.7	Carrinho de Compras — Visão Utilizador	71
6.2.8	Produtos — Visão Administrador	72
6.2.9	Encomendas — Visão Administrador	73
6.2.10	Manuais — Visão Administrador	74
6.2.11	Instruções dos Manuais — Visão Administrador	75
6.3	Análise e avaliação da aplicação desenvolvida	75

7	Conclusões e Trabalho Futuro	77
7.1	Conclusões	77
7.2	Trabalho Futuro	77

Lista de Figuras

1.1	Logótipo ComplicadaMente	3
1.2	Palete de cores ComplicadaMente	4
1.3	Maquete do sistema	6
1.4	Diagrama de Gantt	7
3.1	Modelo de domínio da ComplicadaMente	14
3.2	Modelo de Use Cases da ComplicadaMente	15
3.3	Use Case: Registrar Conta	16
3.4	Use Case: Alterar dados da conta	17
3.5	Use Case: Adicionar quebra-cabeças/peça á encomenda	17
3.6	Use Case: Alterar Encomenda	18
3.7	Use Case: Registrar Encomenda	18
3.8	Use Case: Criar Conta de Funcionário	19
3.9	Use Case: Eliminar conta de Funcionário	20
3.10	Use Case: Gera Lucros	20
3.11	Use Case: Montar quebra-cabeças.	21
3.12	Use Case: Alterar Estado da Encomenda.	22
3.13	Use Case: Criar Peça.	22
3.14	Use Case: Preparar Encomenda	23
3.15	Use Case: Fazer Login	24
3.16	Use Case: Fazer Logout	24
3.17	Diagrama de Atividade do Administrador	25
3.18	Diagrama de Atividade do Utilizador	26
3.19	Diagrama de Atividade do Funcionário em relação á Encomenda	27
3.20	Diagrama de Atividade do Funcionário em relação á Peça.	28
4.1	Modelo Lógico do Sistema	30
5.1	Mockup página de login/registo	40
5.2	Mockup página principal	41
5.3	Mockup resultados de pesquisa	42
5.4	Mockup scroll de resultados de pesquisa	43
5.5	Mockup botão menu	44
5.6	Mockup botão user	45
5.7	Mockup vista produto	46
5.8	Mockup carrinho	47
5.9	Mockup confirmar produtos	48

5.10	Mockup confirmar dados pessoais	49
5.11	Mockup selecionar morada	50
5.12	Mockup selecionar método de pagamento	51
5.13	Mockup dados da encomenda	52
5.14	Mockup consultar encomendas	53
5.15	Mockup consultar produto encomendado	54
5.16	Mockup selecionar tipo de funcionário	55
5.17	Mockup login do funcionário	56
5.18	Mockup página principal do funcionário	57
5.19	Mockup passos da produção/montagem	58
5.20	Mockup informações adicionais da peça	59
5.21	Mockup finalização da peça	60
5.22	Mockup página principal do administrador	61
5.23	Mockup scroll da página principal do administrador	62
5.24	Mockup consultas e alterações de dados	63
6.1	Implementação da Autenticação	66
6.2	Implementação da Loja	67
6.3	Erro de Stock na forma de Modal	68
6.4	Implementação da Loja de Produtos Populares	68
6.5	Implementação das Encomendas do Utilizador	69
6.6	Detalhes das Encomendas do Utilizador	70
6.7	Implementação do Carrinho de Compras do Utilizador	71
6.8	Confirmação de Encomenda	72
6.9	Implementação da página dos Produtos	72
6.10	Implementação da página de Encomendas	73
6.11	Implementação da página de Manuais	74
6.12	Implementação da página de Instruções dos respetivos Manuais	75

Lista de Tabelas

4.1	Tabela da entidade Utilizador.	32
4.2	Tabela da entidade Funcionário.	33
4.3	Tabela da entidade Encomenda.	34
4.4	Tabela da entidade Peça.	35
4.5	Tabela da entidade Quebra-Cabeça.	36
4.6	Tabela da entidade Quebra-Cabeça de Encomenda.	37
4.7	Tabela da entidade Peça de Encomenda.	38

1 Introdução

1.1 Contextualização

Em 1945, quando o senhor José Pinto ainda só tinha 10 anos, foi introduzido pela primeira vez a um quebra-cabeças... a estrela de Galileu. Ao longo dos anos, desenvolveu uma paixão por estes puzzles e começou a tentar fazê-los à mão em casa. Aos 22 anos, em 1967 abriu a primeira loja de quebra-cabeças feitos à mão por ele, a “ComplicadaMente”, situada em Lomar, Braga. Durante os primeiros meses, apenas pais ou algumas crianças com dinheiro que ganhavam nos afazeres é que passavam lá para comprar um puzzle ou outro, mas, ao levar os puzzles para casa, os adultos das famílias perceberam que aquilo era mais do que um brinquedo, era um desafio mental. Isto levou à expansão do público-alvo e, em 1984, os filhos de José já ajudavam na produção dos quebra-cabeças que, devido à grande procura, já eram encomendados para serem revendidos em várias lojas do país.

1.2 Motivação e Objectivos

Com o constante crescimento da “ComplicadaMente”, o senhor José, mesmo com a ajuda dos filhos, já não dava conta do recado para tantas encomendas. Assim, a “ComplicadaMente” passou de ser uma loja a ser uma fábrica. Vários funcionários foram contratados para a produção dos quebra-cabeças e respetiva montagem. Porém, com esta expansão, a gestão de encomendas, peças a produzir, montagem do puzzle e colocação nas caixas para serem vendidos, também complicou bastante. O senhor José, com agora 89 anos, já não consegue dar conta do recado e pediu ajuda ao seu braço direito, o filho mais velho. Após alguma ponderação, ficou claro que fazia falta um sistema software para ajudar a lidar com esta gestão. Ficaria mais fácil o processamento das encomendas, a notificação dos clientes sobre o estado das encomendas, a atribuição de peças a fazer consoante a prioridade dos pedidos e também da atribuição dos quebra-cabeças a montar para serem enviados. Para além disso, os manuais de instruções nas linhas de montagem que estavam sempre a ser perdidos ou danificados, passariam a estar disponíveis no programa em formato digital. Para tal, o senhor Arlindo, filho de José, contratou a “SimpleSistems” para tratar do desenvolvimento do sistema de Software.

O objetivo geral deste sistema de Software é então facilitar a gestão da linha de montagem dos quebra-cabeças, através do design, desenvolvimento e implementação de uma plataforma digital. Juntamente com o senhor José, a equipa da SimpleSistems definiu um conjunto de

objetivos que pretendem alcançar com a implementação do sistema software:

- O desenvolvimento de uma plataforma de utilização intuitiva e apelativa aos clientes.
- Que o sistema trate da gestão das encomendas registadas pelos clientes, fazendo a análise da quantidade de peças e puzzles existentes e a produzir.
- Que seja feita a distribuição automática das peças a serem produzidas e dos puzzles a serem montados pelos funcionários e que estes tenham acesso digital aos respetivos manuais.
- Que o sistema registe sempre que uma peça ou puzzle é finalizado, tendo em conta se este está em condições.
- A atualização em tempo real das encomendas.
- A notificação automática do administrador das encomendas que estão prontas para serem enviadas.
- O acesso do administrador a estatísticas de produção, lucro, peças/puzzles produzidos e descartados.

1.3 Justificação, utilidade e viabilidade do sistema

A viabilidade do projeto foi cuidadosamente avaliada através de uma abordagem metodológica e fundamentada. Para tal, foi feito um estudo para saber qual seria a média da aderência de utilizadores de tecnologias desenvolvidas do género.

Segundo as estatísticas do site AppBrain, a categoria Business aparece no segundo lugar na loja de aplicações Google Play. É importante referir que existem cerca de 49 categorias diferentes de aplicações. Devido à comunidade de amantes de quebra-cabeças ter crescido exponencialmente ao longo dos anos, o que levou a que o senhor José Pinto tivesse que expandir a própria loja, o facto de o próprio negócio do mesmo já ter obtido sucesso anteriormente e também a particularidade de ser um negócio incomum garantem a viabilidade financeira por possuir o monopólio completo desse ramo.

Quanto à viabilidade técnica e financeira, já havia inicialmente uma noção concreta, pois tinha sido definido numa primeira reunião com o Sr. Arlindo, bem como os nossos recursos a nível de máquinas e mão de obra, permitindo então fazer uma alocação cuidadosa de recursos para todas as fases do projeto.

A principal utilidade do sistema reside na capacidade de apoiar o crescimento e a consolidação do negócio de José Pinto, que já possui um monopólio no nicho de quebra-cabeças. O sistema pode ser usado para otimizar processos operacionais, como a gestão de inventário e logística, crucial para acompanhar a expansão da loja. Com a crescente demanda de aplicações de

negócios, um sistema eficiente pode fortalecer a estratégia de marketing e garantindo o sucesso contínuo do negócio. A tecnologia permite que o sistema registre, priorize e acompanhe encomendas em tempo real, o que aumenta a produtividade, pois a distribuição de tarefas é efetuada de forma automática. Com uma análise de dados, o sistema pode ainda fornecer relatórios de estatísticas sobre o desempenho da linha de montagem, percentagem de peças descartadas, lucros, etc..., facilitando decisões necessárias baseadas em objetivos.

Em suma, a tecnologia aplicada neste sistema não só melhora a eficiência como também oferece uma margem de crescimento para o futuro da empresa, reduzindo a margem do erro humano, concluindo que o projeto é viável e com tendência para o sucesso.

1.4 Estabelecimento da identidade do projeto

A identidade da empresa "ComplicadaMente" já foi bastante definida, faltando o logótipo:

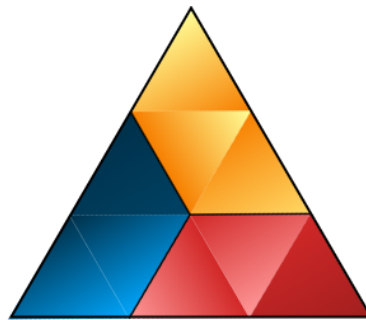


Figura 1.1: Logótipo ComplicadaMente

A paleta de cores:

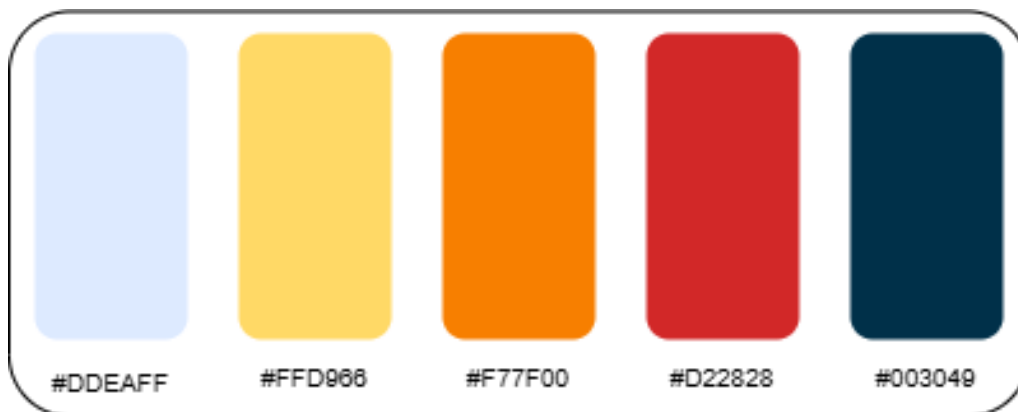


Figura 1.2: Paleta de cores ComplicadaMente

Em relação à SimpleSistemas, é uma empresa de desenvolvimento de sistemas de software, criada no início de 2024, composta por uma equipa de 5 engenheiros de software, entre os quais 3 são especialistas de Backend e bases de dados e 2 são especialistas de Frontend. Não achamos relevante para o projeto apresentar o logótipo e a paleta de cores da SimpleSistemas.

1.5 Identificação dos recursos necessários

Para o projeto da ComplicadaMente foi necessário alocar os seguintes recursos:

1.5.1 Recursos Humanos

A equipa de desenvolvimento da "SimpleSistemas", incluindo um gestor do projeto para coordenar atividades e o trabalho em equipa. Além disso serão necessários engenheiros de software para desenvolver a estrutura operacional que garanta todas as funcionalidades pretendidas, o chamado back-end. Posteriormente, serão necessários especialistas em front-end para realizar a interface intuitiva e também uma equipa para efetuar testes para garantir o bom funcionamento da aplicação.

Adicionalmente são necessários operadores de linha bem como administradores e supervisores. Será necessário também uma formação a todos os utilizadores do software para que se sintam familiarizados quando forem efetivamente utilizá-lo.

1.5.2 Recursos Tecnológicos

A fábrica necessitará de servidores e infraestrutura de rede para hospedar o sistema e garantir o acesso ao sistema em várias máquinas. É crucial que todas as estações de trabalho estejam

equipadas com computadores ou tablets para que os funcionários possam utilizar o software. Além disso é necessário também um sistema de base de dados para armazenar as informações de inventário, encomendas, clientes, bem como monitorizar o progresso das encomendas. É intrínseco também um sistema de backup e segurança para garantir a integridade dos dados armazenados.

1.5.3 Recursos Físicos

A fábrica precisará de um espaço adequado para a instalação dos equipamentos tecnológicos bem como as máquinas de suporte técnico, assegurando que o sistema funcione como um todo naturalmente.

1.5.4 Recursos Financeiros

Como foi mencionado no capítulo 1.3, os recursos financeiros foram definidos numa reunião inicial com o Sr. Arlindo. Haverá inicialmente um investimento significativo para o desenvolvimento e implementação do software, que tem em conta o pagamento á "SimpleSistems" e os custos de manutenção. Além disso será também necessário investir nos equipamentos tanto tecnológicos como técnicos (computadores, tablets, máquinas de automatização) bem como em formações para os operadores.

1.6 Maquete do sistema

O sistema a desenvolver pode ser separado em duas componentes: Front-end e Back-end:

- O website permite ao utilizador ter um conjunto de funcionalidades, entre as quais acompanhar as suas encomendas em relação ao seu estado, poder realizar as próprias e também por ventura de peças em falta para algum tipo de puzzle.
- A lógica do negócio corresponde a gerir os utilizadores e as suas encomendas, bem como permitir que se registem, efetuem login e usufruam das funcionalidades.
- Já o back-end é responsável por ser o "suporte" do website, alocando todos os dados de todas as entidades(utilizadores, encomendas, puzzles,...), gerir essa mesma informação e manter o sistema estável internamente.

A seguir encontra-se uma imagem de como pensámos ser a maquete do nosso sistema.

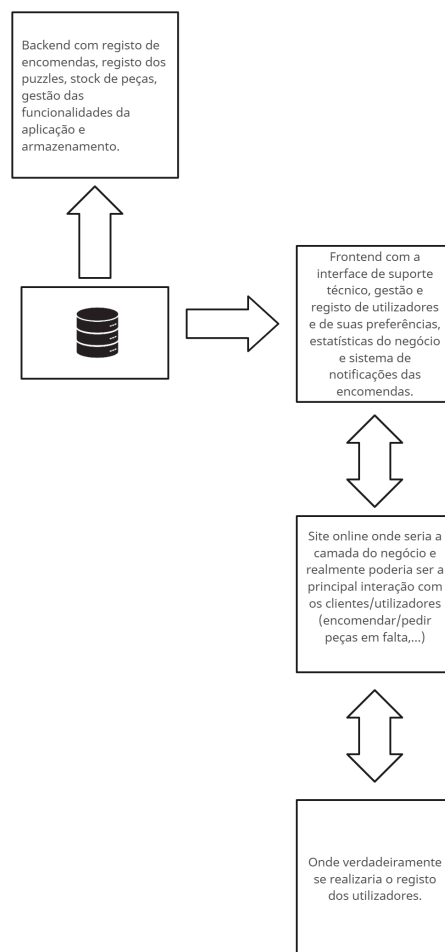


Figura 1.3: Maquete do sistema

1.7 Definição de um conjunto de medidas de sucesso

Para o sucesso da gestão da linha de montagem da ComplicadaMente existe um número de medidas de sucesso intrinsecamente importantes de atingir para o crescimento exponencial do projeto, tais como:

- Redução máxima do tempo de processamento de cada encomenda em cerca 25%, fazendo com que seja mais rápido o processo de uma encomenda e posterior entrega ao cliente, tentando captar mais clientes com o serviço rápido e com qualidade.
- Intransigência no controlo de stock, evitando ao limite erros no stock fazendo com que não existam inconsistências ou o menor número possível delas para maximizar a precisão em posteriores encomendas de novo material, evitando custos desnecessários.

- Aumentar a eficiência na produção comparando com o pré software em cerca de 20%, elevando assim o sucesso da empresa fabricando a maior quantidade de produto num menor espaço de tempo.
- Satisfação dos clientes, sendo uma das medidas mais importantes pois efetuando as encomendas de uma forma mais eficiente e com melhor qualidade vai fazer com que o público consumidor fique satisfeito com o serviço, sendo importante também para a percentagem de produtos que são devolvidos diminuir.
- Cumprimento dos prazos de entrega é crucial para que o sistema de gestão funcione, tanto para o cliente como para a própria empresa pois o não cumprimento afeta de certa forma a gestão do stock desequilibrando o inventário, por isso seria importante executar com exatidão os prazos estipulados.

1.8 Plano de desenvolvimento (Diagrama de gantt)

Para otimizar a gestão do nosso tempo de forma eficaz, criamos um diagrama de Gantt. Neste diagrama, dividimos o projeto em 5 fases principais, alocando o tempo necessário às fases mais complexas.

TAREFA	Atribuída a	Progresso	Início	FIM
Definição do Sistema				
Contextualização e Fundamentação	Sofia	100%	3/16/24	3/19/24
Objetivos e Viabilidade	Tiago	100%	3/19/24	3/22/24
Recursos a utilizar e equipa de trabalho	Rafael	100%	3/22/24	3/25/24
Plano de Execução	Manuel	100%	3/25/24	3/28/24
Revisão e Aprovação	Rodrigo	100%	3/28/24	3/29/24
Definição de Requisitos				
Apresentação da estratégia e do método	Tiago	100%	3/29/24	10/2/24
Levantamento de Requisitos	Sofia e Manuel	100%	10/2/24	10/7/24
Validação dos requisitos estabelecidos	Rodrigo e Rafael	100%	10/7/24	10/10/24
Especificação e Modelação do Software				
Apresentação geral da especificação	Tiago	100%	10/10/24	10/13/24
Aspetos estruturais - Modelo de domínio	Rafael e Sofia	100%	10/13/24	10/17/24
Aspetos comportamentais- Diagrama de Use cases	Rodrigo e Manuel	100%	10/17/24	10/21/24
Aspetos comportamentais- Especificação de Use Cases	Manuel, Rafael, Rodrigo, Sofia e Tiago	100%	10/21/24	10/31/24
Conceção do Sistema de dados				
Apresentação geral da estrutura (esquema) do sistema de dados	Rafael e Manuel	100%	10/31/24	11/3/24
Descrição detalhada dos vários elementos de dados e seus relacionamentos	Rodrigo, Sofia e Tiago	100%	11/3/24	11/11/24
Esboço das Interfaces do Sistema				
Estrutura geral das interfaces do sistema	Manuel e Sofia	100%	11/11/24	11/14/24
Caracterização das interfaces	Rafael, Rodrigo e Tiago	100%	11/14/24	11/19/24
Implementação da plataforma de gestão da linha de montagem				
Criação da Base de Dados	Rafael	0%	11/19/24	12/19/24
Front-end	Rodrigo e Tiago	0%	11/19/24	12/19/24
Back-end	Sofia e Manuel	0%	11/19/24	12/19/24

Figura 1.4: Diagrama de Gantt

2 Levantamento e Análise de Requisitos

2.1 Apresentação da estratégia e método

Para concretizar o levantamento dos requisitos necessários ao desenvolvimento do sistema da "ComplicadaMente", optamos por uma abordagem multifacetada, incluindo diversas técnicas e métodos de recolha de informação.

Inicialmente, foram realizadas reuniões com os principais stakeholders da fábrica, incluindo o Sr. Arlindo e os responsáveis pelas diferentes áreas de produção e montagem de quebra-cabeças. O objetivo dessas reuniões foi identificar as necessidades dos utilizadores em relação ao sistema, recolhendo assim os requisitos necessários (funcionais e não funcionais).

Outra estratégia utilizada foi a realização de sessões de brainstorming com a equipa de desenvolvimento da "SimpleSistemas" e os utilizadores em concreto (operários das linhas de montagem, supervisores e responsáveis pelo armazenamento do produto) para explorar possíveis ideias, requisitos demasiado específicos que apenas os trabalhadores mais interinos conseguiam fornecer e validar funcionalidades chave para a gestão de peças e controlo de stock.

Adicionalmente, recorremos á análise de documentos e relatórios, incluindo inventários de peças, planos de produção e ordens de encomenda para entender melhor os processos internos da empresa. Esta análise permitiu-nos identificar requisitos técnicos adicionais, como a necessidade de um sistema de notificação para o estado das encomendas e disponibilidade de peças.

Durante este processo foi gerada documentação que inclui atas de reuniões, notas de entrevista com os utilizadores e modelos de casos de uso, que foram incorporados no desenvolvimento do sistema.

Este método de levantamento e análise de requisitos possibilitou a compreensão detalhada das necessidades da empresa "ComplicadaMente" e das funcionalidades intrínsecas para que o sistema final seja eficaz e alinhado com os objetivos de melhoria da gestão da linha de montagem.

Ata-resumo da reunião de levantamento de requisitos Data: 14 de outubro de 2024 Participantes:

- Equipa de desenvolvimento da "SimpleSistemas"
- Sr. Arlindo
- Operários da linha de montagem
- Responsáveis pelo armazenamento
- Supervisores da produção e montagem

Nesta reunião foram discutidos as principais adversidades enfrentadas pela linha de montagem, nomeadamente a dificuldade da gestão manual das encomendas, a desorganização e o acompanhamento das mesmas. Os supervisores destacaram a importância de um sistema que priorize encomendas em que o prazo limite de entrega seja mais urgente.

A equipa de produção da linha de montagem enfatizou a necessidade de um sistema que mostre os manuais de montagem digital, eliminando a dependência constante de manuais físicos que facilmente se perdem ou se danificam. Foi sugerida também pelo Sr. Arlindo um sistema para acompanhamento do envio das encomendas e o próprio estado das mesmas pois sabe a frustração que os clientes sentem quando por vezes o processo é demorado, diminuindo assim a clientela que por ventura poderá evitar a empresa sabendo esse motivo.

Cada um destes pontos foi discutido e os requisitos necessários para o desenvolvimento foram documentados para garantir a sua implementação.

Fim da reunião.

2.2 Descrição geral dos Requisitos levantados

2.2.1 Requisitos Funcionais

1. O sistema deve permitir que um utilizador se autentique e posteriormente possa efetuar login e logout da interface do sistema através de um username e de uma password.
2. O sistema deve permitir ao administrador (admin) efetuar a criação das contas dos funcionários, bem como a gestão dos mesmos e assegurar cargos mediante cada função na linha de montagem.
3. O sistema deve permitir que o utilizador registre encomendas, associando o nome de cliente, tipo de quebra-cabeças, número de unidades e a data de entrega prevista.
4. O sistema deve verificar automaticamente o stock de peças existentes no armazém,

enviando uma notificação ao utilizador caso exista abaixo de uma certa quantidade estabelecida para um produto.

5. O sistema deve permitir a criação de perfis para diferentes utilizadores com permissões específicas mediante o seu posto de trabalho (operários, supervisores ou administradores).
6. Os utilizadores dos postos de trabalho de criação de peças (tipo 1) ou montagem de quebra cabeças (tipo 2) devem poder aceder aos manuais de instruções digitais disponíveis no sistema.
7. O sistema deve atribuir automaticamente as tarefas específicas a cada funcionário mediante a sua função e cargo.
8. O sistema deve permitir que o utilizador organize as encomendas por prioridade, com base nas datas de entrega e/ou em condições referidas pelo cliente.
9. O sistema deve manter um histórico das encomendas já realizadas com a possibilidade de as consultar na eventualidade de uma análise ou estatística futura.
10. O sistema deve permitir ao utilizador verificar, a qualquer momento, o estado de uma encomenda específica, verificando se está ainda em produção, em fase de montagem ou já se encontra pronta a ser enviada.
11. O sistema deve notificar os responsáveis pelas encomendas cada vez que as mesmas estiverem prontas para entrega.
12. O sistema deve permitir ao utilizador visualizar de forma detalhada o processo de montagem de uma peça mostrando instruções passo a passo e uma interface interativa que mostre a dada peça a ser montada corretamente.
13. O sistema deve permitir ser gerado um relatório que inclua a quantidade de peças produzidas, quebra-cabeças montados, peças descartas por defeito/erro e lucro gerado por o espaço de tempo que for pretendido, facilitando a análise de desempenho da linha.
14. O sistema deve ter a informação guardada de eficiência individual por utilizador, permitindo que os gestores consigam identificar os trabalhadores que são mais produtivos efetivamente.
15. O sistema deve enviar notificações automáticas aos utilizadores responsáveis pela montagem de existir alguma atualização ou modificação nos manuais de montagem.
16. O sistema deve permitir a reatribuição automática de tarefas de montagem de quebra-cabeças ou de produção de peças aos funcionários numa eventualidade de falha de equipamentos ou indisponibilidade/falta de algum funcionário, para evitar assim atrasos e garantir a continuidade da produção.

2.2.2 Requisitos Não Funcionais

1. A plataforma apenas necessitará de acesso á internet para o funcionamento, bem como possivelmente em alguns dispositivos mais antigos ser necessária a aquisição de um plugin para a visualização da montagem.
2. O sistema deve ser escalável fazendo com que suporte o aumento de utilizadores sem comprometer o desempenho da plataforma.
3. A plataforma deverá garantir total segurança dos dados tanto dos utilizadores como das próprias encomendas, que sejam protegidos por autenticação e por permissão, fazendo com que cada cliente apenas possa aceder aos dados e acompanhamento da sua encomenda.
4. A plataforma deverá estar funcional 24 horas por dia, 7 dias por semana.
5. A plataforma deve ter um design minimalista e de simples visualização.
6. O sistema deve ser modular e de fácil manutenção, garantindo que na existência de algum erro, seja fácil a identificação e correção do mesmo.
7. A plataforma deverá ter uma boa performance, tanto em termos de resposta como em processamento (ficando dependente também da conexão internet a que o dispositivo que a executa está ligado).

2.3 Validação dos requisitos estabelecidos

O processo de validação dos requisitos (funcionais e não funcionais) foi repartida em três fases, abordadas em reuniões colaborativas entre a equipa da SimpleSystems e os stakeholders da ComplicadaMente:

1. Fase de Descrição dos Requisitos: Numa fase inicial, o objetivo foi identificar e especificar as funcionalidades mais cruciais que o sistema deveria ter para oferecer ao utilizador. Os pontos que foram abordados incluíram o sistema de autenticação, a gestão de contas (perfis), controlo de stock e a gestão de encomendas. Foi também necessitada a visualização da montagem das peças como crucial e a capacidade de atribuir tarefas de forma automática.
2. Fase de Exploração dos Requisitos: De seguida, a atenção foi concentrada em validar e aprofundar requisitos funcionais adicionais que contribuiriam para aumentar a eficiência da linha de montagem, como por exemplo a distribuição automática de tarefas e as notificações automáticas aos utilizadores em caso de alguma atualização. Nesta fase o objetivo principal foi identificar todas as informações que o sistema deveria guardar e fornecer ao utilizador, como o histórico de encomendas, garantindo assim que cada

elemento tivesse o devido suporte para a continuidade operacional da empresa.

3. Fase de Administração e Restrições: No final foram analisadas as restrições que garantiriam a segurança, estabilidade e escalabilidade do sistema, como o controlo de acessos simultâneos, a respetiva segurança de dados dos utilizadores e das encomendas, a necessidade de um design simples mas ao mesmo tempo atrativo.

Ata-resumo da Reunião de Validação de Requisitos Data: 18 de outubro de 2024

Participantes:

- Equipa de desenvolvimento da "SimpleSystems"
- Sr. Arlindo
- Operários da linha de montagem
- Responsáveis pelo armazenamento
- Supervisores da produção e montagem

Durante a discussão geral realizou-se uma revisão de todos os requisitos funcionais e não funcionais do sistema, com o objetivo de garantir que todos estavam bem definidos e que correspondiam a todas as necessidades e expectativas do projeto em si:

Pontos críticos abordados:

- Foi validado que as funcionalidades prioritárias, relacionadas com a gestão de encomendas, monitorização do progresso da produção e mesmo a visualização da montagem da peça, estavam todas alinhadas com o pretendido e respeitavam os objetivos da empresa.
- Analisaram-se também potenciais melhorias em relação á segurança, especialmente de dados respetivos ao cliente, como moradas, dados de encomendas, históricos de compras, informação confidencial que tem de ser restrita e estar protegida de forma segura.
- Foram também discutidos os critérios de escalabilidade e de manutenção do sistema, de modo a garantir que este possa acompanhar o crescimento da empresa num eventual salto populacional e possíveis expansões sem comprometer obviamente a performance e a estabilidade do sistema.

Conclusão: No final da reunião chegou-se a um consenso geral sobre todos os requisitos definidos, considerando-os adequados para cumprir os objetivos do projeto tanto em termos operacionais como em termos estratégicos. A equipa de desenvolvimento validou também os requisitos juntamente com a ComplicadaMente, em que ambas concordaram com o conjunto de funcionalidades estabelecidas para o sistema.

3 Especificação e Modelação do Software

3.1 Apresentação geral da especificação

Este modelo de domínio representa as relações entre as várias entidades do sistema da ComplicadaMente, nomeadamente a gestão de stock, encomendas e da própria linha de montagem, controlo de produção e também da visualização da montagem do produto. O nosso modelo de domínio foi efetuado em UML de forma a garantir que todos os aspetos críticos do sistema estão bem representados e alinhados com o objetivo do projeto, de forma a fazer com que o desenvolvimento do sistema em si seja mais estruturado e eficiente.

3.2 Aspetos Estruturais

- Os clientes estão habilitados para fazerem encomendas e estas podem possuir um estado e também um tipo de quebra-cabeças encomendado (tipo e quantidade).
- Os administradores supervisionam os funcionários/operários de linha de montagem, verifica o stock e controla as informações e lucro gerado.
- Os funcionários montam o quebra-cabeças que por sua vez é feito de peças criadas por eles próprios.
- Cada utilizador (cliente/funçãoário/administrador) autentica-se com o seu ID respetivo e a sua password.
- O stock de peças contém uma quantidade específica que é verificada regularmente pelo administrador de forma a garantir que a produção ocorra eficientemente e sem interrupções.
- O lucro é gerado baseado nas encomendas realizadas, e o administrador acompanha essas métricas de forma a avaliar o desempenho geral da linha de montagem.

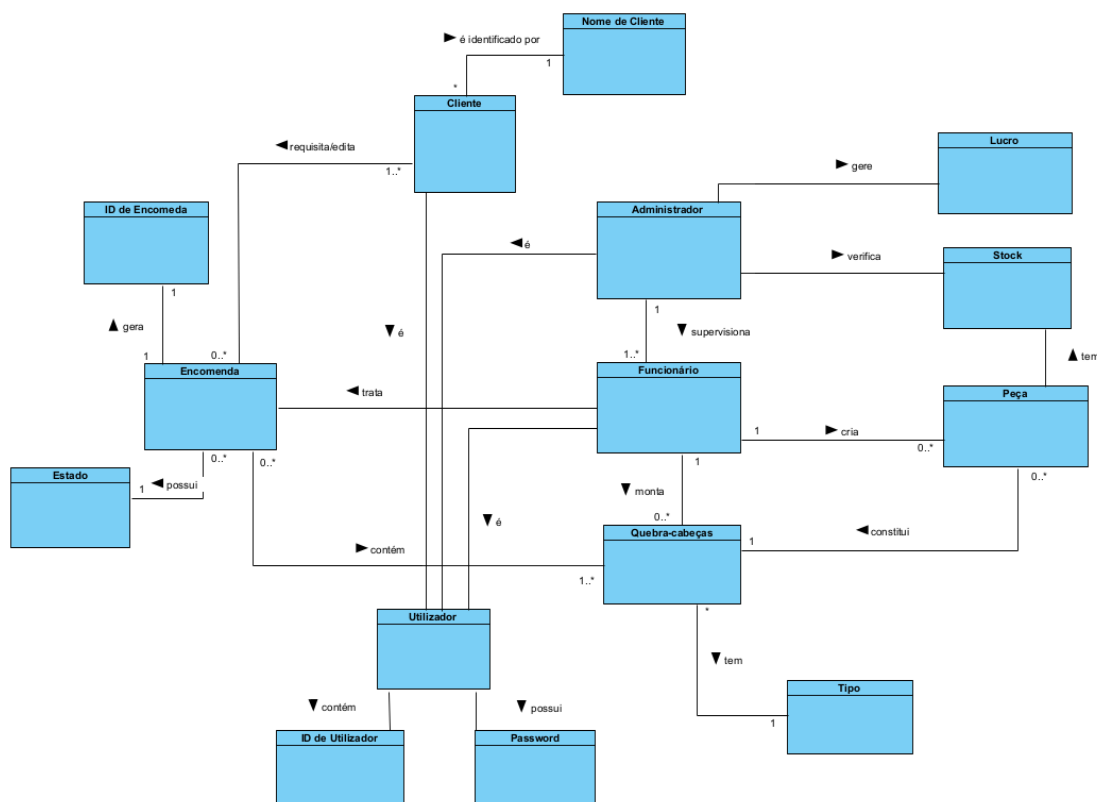


Figura 3.1: Modelo de domínio da ComplicadaMente

3.3 Aspetos Comportamentais

Nesta secção fazemos referência aos aspetos comportamentais da plataforma, que desempenham um papel intrínseco no que toca á interação do sistema com todos os utilizadores do mesmo. Para demonstrar melhor a interação entre os atores (administradores, funcionários e utilizadores) foi criado o diagrama de use cases do sistema que contém os casos de uso que considerámos essenciais para o projeto.

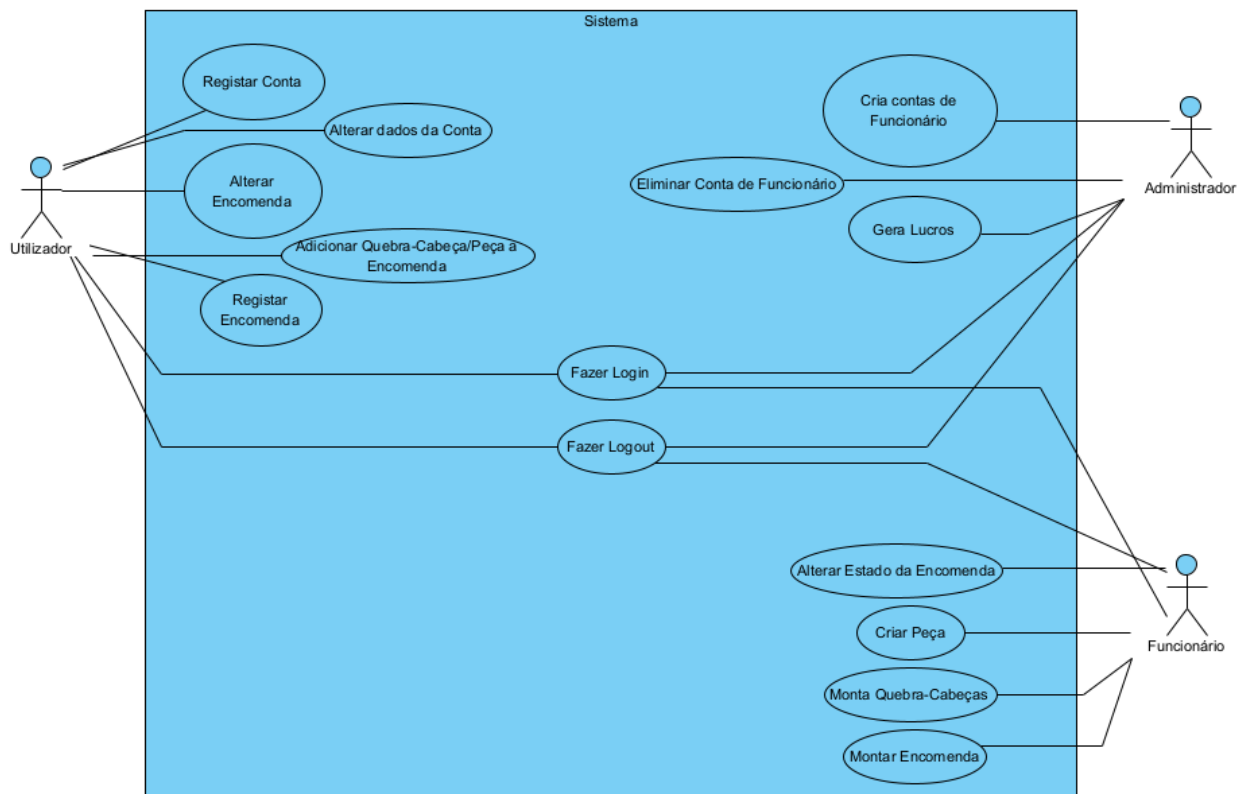


Figura 3.2: Modelo de Use Cases da ComplicadaMente

▪ Use Cases Utilizador:

1. Registar Conta

Um utilizador pode criar uma conta na aplicação para aceder às funcionalidades disponíveis. Para isso, ele deve fornecer informações como nome, e-mail e senha. Uma vez registados os dados, o sistema verifica se o e-mail já está associado a outra conta. Se não estiver, a conta é criada, e o utilizador é notificado. Caso contrário, a aplicação informa que o e-mail já está em uso.

USE CASE:	Registar Conta		
DESCRIÇÃO:	Permitir que o utilizador registe uma nova conta.		
CENÁRIOS:	Criar conta nova		
PRÉ-CONDIÇÃO:	Sistema está acessível		
PÓS-CONDIÇÃO:	A conta do utilizador é criada com sucesso.		
	Ator		Sistema
FLUXO NORMAL:	1. Utilizador preenche as informações necessárias para o registo.	2.	O sistema valida as informações fornecidas.
		3.	Sistema cria a conta e notifica o utilizador que a conta foi criada com sucesso.
FLUXO DE EXCEÇÃO		(1)	Informações fornecidas são inválidas ou já existem. (Passo 2)
		2.1.	O sistema informa que as informações são inválidas ou o ID já está em uso.

Figura 3.3: Use Case: Registar Conta

2. Alterar dados da Conta

Este caso de uso permite que um utilizador autenticado no sistema atualize informações pessoais associadas à sua conta. Esta funcionalidade é fundamental para garantir que os dados da conta estejam sempre atualizados e corretos. Para realizar a atualização, o utilizador navega até a página de edição do perfil, onde pode modificar informações como nome, e-mail, password ou outros dados relevantes. O sistema verifica as alterações realizadas, validando os dados inseridos de acordo com os critérios estabelecidos, como unicidade ou formato correto. Se os dados forem válidos, o sistema atualiza as informações na base de dados e informa ao utilizador que as alterações foram realizadas com sucesso. No entanto, caso os dados inseridos sejam duplicados (por exemplo, um e-mail já utilizado por outra conta), o sistema alerta o utilizador sobre o conflito e solicita que insira informações únicas.

USE CASE:	Alterar Dados da Conta		
DESCRIÇÃO:	Permitir que o utilizador atualize informações da sua conta.		
CENÁRIOS:	Atualizar informações pessoais.		
PRÉ-CONDIÇÃO:	O utilizador autenticado no sistema.		
PÓS-CONDIÇÃO:	As informações da conta do utilizador são atualizadas no sistema.		
	Ator		Sistema
FLUXO NORMAL:	1. O utilizador navega até a página de edição do perfil. 2. O utilizador insere as novas informações desejadas.	3. O sistema valida os dados inseridos. 4. O sistema guarda as alterações na conta do utilizador. 5. O sistema informa o utilizador que as alterações foram realizadas com sucesso.	
FLUXO DE EXCEÇÃO		(1) O utilizador insere dados duplicados (Passo 3) 1.1. O sistema informa que os dados fornecidos já são existentes no sistema. 1.2. O sistema solicita que o utilizador insira dados únicos.	

Figura 3.4: Use Case: Alterar dados da conta

3. Adicionar quebra-cabeças/peça á encomenda

Um utilizador autenticado no sistema pode adicionar um quebra-cabeças ou uma peça específica a uma encomenda existente. Para iniciar, o utilizador navega até a secção de itens no sistema, onde é apresentada uma lista completa de quebra-cabeças e peças disponíveis. Após visualizar a lista, o utilizador seleciona o item desejado. O sistema exibe os detalhes do item selecionado, como descrição, preço e disponibilidade em stock. O utilizador, então, opta por adicionar o item à encomenda. Antes de confirmar, o sistema verifica se há stock suficiente do item escolhido. Caso o stock seja adequado, o sistema adiciona o item à encomenda. Por fim, o sistema exibe uma mensagem de confirmação que o item foi adicionado com sucesso à encomenda. Se o item não estiver disponível em stock, o sistema informa o utilizador da indisponibilidade.

USE CASE:	Adicionar um quebra-cabeças/peça à encomenda		
DESCRIÇÃO:	Este caso de uso descreve como o utilizador pode adicionar um quebra-cabeças ou uma peça específica a uma encomenda existente.		
CENÁRIOS:	Adicionar um quebra-cabeças/peça a uma encomenda existente.		
PRÉ-CONDIÇÃO:	O utilizador deve estar autenticado no sistema.		
PÓS-CONDIÇÃO:	A encomenda do utilizador é atualizada com o novo item.		
	Ator		Sistema
FLUXO NORMAL:	1. O utilizador navega até a secção de itens. 3. O utilizador seleciona um quebra-cabeças/peça específico. 5. O utilizador clica adiciona à encomenda o item. 8. O utilizador recebe a confirmação que o item foi adicionado.	2. O sistema apresenta a lista de itens. 4. O sistema exibe os detalhes do item. 6. O sistema verifica se existe stock do item escolhido. 7. O sistema adiciona o quebra-cabeças/peça à encomenda selecionada pelo utilizador.	
FLUXO DE EXCEÇÃO		(1) Stock Insuficiente (Passo 6) 1.1. O sistema informa que o item encontra-se fora de stock.	

Figura 3.5: Use Case: Adicionar quebra-cabeças/peça á encomenda

4. Alterar Encomenda

Um utilizador autenticado pode alterar uma encomenda existente na aplicação, caso ainda esteja dentro do prazo de alteração definido pela empresa. Para isso, ele deve selecionar a encomenda desejada e especificar as alterações, como o tipo do quebra-cabeças e a quantidade desejada. O sistema verifica se a encomenda está elegível para edição e, caso esteja, aplica as alterações definidas e atualiza as informações na base de dados. Se a encomenda não puder ser alterada, o sistema informa o utilizador do porquê de não poder ser alterada.

Este use case é validado pelo requisito funcional 10.

USE CASE:	Alterar Encomenda		
DESCRIÇÃO:	Permitir que o utilizador altere detalhes de uma encomenda existente.		
CENÁRIOS:	Atualizar encomenda		
PRÉ-CONDIÇÃO:	O utilizador tem uma encomenda existente e o sistema está acessível.		
PÓS-CONDIÇÃO:	A encomenda é alterada com sucesso.		
	Ator	Sistema	
FLUXO NORMAL:	1. Utilizador seleciona a encomenda que deseja modificar. 2. Utilizador altera as informações necessárias.	3. O sistema valida e guarda as novas informações. 4. Sistema informa o utilizador que a encomenda foi atualizada com sucesso.	
FLUXO DE EXCEÇÃO		(1) (1) Encomenda não pode ser alterada. (e.g., já está em envio) (Passo 2) 2.1. O sistema informa que a encomenda já foi processada e não pode ser alterada.	

Figura 3.6: Use Case: Alterar Encomenda

5. Registrar Encomenda

Um utilizador autenticado pode aceder ao sistema para realizar o registo de uma nova encomenda, escolhendo os produtos desejados. O processo inicia-se quando o utilizador seleciona os produtos e os adiciona ao carrinho. Em seguida o sistema valida as informações fornecidas, verificando a disponibilidade dos produtos selecionados. Caso todos os critérios sejam atendidos, a encomenda é registada com sucesso no sistema, e o utilizador é notificado. Se um ou mais produtos estiverem fora de stock, o sistema informa o utilizador sobre a indisponibilidade, permitindo que ele remova os itens que de momento estão indisponíveis e prossiga com o resto da encomenda. Este fluxo assegura que o utilizador possa concluir a compra sem interrupções desnecessárias, mesmo em casos de indisponibilidade parcial dos produtos selecionados.

Este use case é validado pelo requisito funcional 3.

USE CASE:	Registrar Encomenda		
DESCRIÇÃO:	Permitir que o utilizador faça uma nova encomenda.		
CENÁRIOS:	Fazer nova encomenda.		
PRÉ-CONDIÇÃO:	O utilizador possui uma conta ativa.		
PÓS-CONDIÇÃO:	A encomenda é registada no sistema com sucesso.		
	Ator	Sistema	
FLUXO NORMAL:	1. O utilizador seleciona os produtos e adiciona-os ao carrinho. 2. O utilizador insere os detalhes de entrega e pagamento.	3. O sistema valida as informações de pagamento e confirma a encomenda 4. O sistema informa o utilizador que a encomenda foi registada com sucesso.	
FLUXO DE EXCEÇÃO		(1) Um ou mais produtos estão fora de stock. (Passo 1) 2.1. O sistema informa o utilizador que os produtos selecionados não estão disponíveis no momento. 2.2. O utilizador pode remover os produtos indisponíveis do carrinho e continuar com a encomenda.	

Figura 3.7: Use Case: Registrar Encomenda

▪ Use Cases Administrador:

1. Criar Conta de Funcionário

Um administrador autenticado pode aceder ao sistema para criar uma nova conta de funcionário, permitindo a gestão e atribuição de cargos. Para iniciar o processo, o administrador insere as informações necessárias do funcionário, como nome, cargo e permissões associadas. O sistema verifica os dados fornecidos. Caso todas as informações estejam corretas, o sistema regista a nova conta na base de dados e notifica o administrador sobre a criação. Se as informações fornecidas pelo administrador forem inválidas, como campos obrigatórios não preenchidos ou valores incorretos, o sistema exibe uma mensagem de erro detalhada e solicita que o administrador corrija os dados antes de prosseguir.

Este use case é validado pelos requisitos funcionais 2 e 5.

USE CASE:	Criar Conta de Funcionário		
DESCRIÇÃO:	Permitir que um administrador crie uma conta de funcionário no sistema.		
CENÁRIOS:	Adicionar novo funcionário		
PRÉ-CONDIÇÃO:	Administrador está autenticado no sistema.		
PÓS-CONDIÇÃO:	conta do funcionário é criada com sucesso.		
	Ator		Sistema
FLUXO NORMAL:	1. O administrador insere as informações do funcionário.	2. O sistema valida as informações fornecidas.	
		3. O sistema cria a conta de funcionário e notifica o administrador do sucesso.	
FLUXO DE EXCEÇÃO		(1) Informações do funcionário são inválidas (Passo 2)	
		2.1. O sistema informa que as informações são inválidas e solicita correção.	

Figura 3.8: Use Case: Criar Conta de Funcionário

2. Eliminar Conta de Funcionário

Um administrador autenticado pode aceder ao sistema para eliminar permanentemente a conta de um funcionário. Este processo é utilizado para remover do sistema os dados de acesso de funcionários que não estão de momento a trabalhar na empresa ou que não necessitam de acesso à plataforma. Para realizar esta ação, o administrador visualiza uma lista com todas as contas de funcionários registadas no sistema. Em seguida, seleciona a conta que deseja eliminar. O sistema solicita uma confirmação da ação para evitar a remoção acidental de uma conta. Após o administrador confirmar a remoção, o sistema elimina permanentemente os dados da conta do funcionário. Por fim, o sistema exibe uma mensagem de sucesso. Se o administrador tentar eliminar uma conta inexistente ou que já tenha sido removida, o sistema informa que a conta selecionada não foi encontrada, não podendo prosseguir o processo.

USE CASE:	Eliminar Conta de Funcionário		
DESCRIÇÃO:	Permitir que o administrador remova permanentemente uma conta de um funcionário.		
CENÁRIOS:	Eliminar uma conta de um funcionário.		
PRÉ-CONDIÇÃO:	Administrador autenticado no sistema.		
PÓS-CONDIÇÃO:	A conta do funcionário é removida do sistema.		
	Ator		Sistema
FLUXO NORMAL:		1.	O sistema exibe a lista de contas de funcionários.
	2. O administrador seleciona a conta que deseja remover.	3.	O sistema solicita a confirmação da ação.
	4. O administrador confirma a remoção.	5.	O sistema remove a conta do funcionário e exibe uma mensagem de sucesso.
FLUXO DE EXCEÇÃO		(1)	Administrador tenta remover uma conta que não existe (Passo 4)
		1.1.	O sistema informa que a conta selecionada não foi encontrada.

Figura 3.9: Use Case: Eliminar conta de Funcionário

3. Gera Lucros

Um administrador autenticado pode aceder ao sistema para gerar os lucros da empresa com base num período de tempo especificado. Este processo permite ao administrador obter uma visão detalhada do desempenho financeiro da empresa em intervalos como semanal, mensal ou anual. Para iniciar, o administrador seleciona o período de tempo desejado para o cálculo dos lucros. O sistema valida a entrada fornecida, verifica a disponibilidade de dados no intervalo especificado e calcula os lucros correspondentes. Assim que o cálculo é concluído, o sistema gera os lucros e apresenta-o ao administrador. Se não houver dados disponíveis para o período selecionado, o sistema informa o administrador sobre a indisponibilidade. Além disso, caso o administrador insira um período de tempo inválido, o sistema exibe uma mensagem de erro que sugere ao administrador a inserção de uma data válida.

Este use case é validado pelo requisito funcional 13.

USE CASE:	Gera Lucros		
DESCRIÇÃO:	Permitir que o administrador gere os lucros da empresa.		
CENÁRIOS:	Gerar lucros (semanal, mensal, anual).		
PRÉ-CONDIÇÃO:	O administrador está autenticado.		
PÓS-CONDIÇÃO:	Os lucros são gerados e disponibilizados para visualização.		
	Ator		Sistema
FLUXO NORMAL:	1. O administrador seleciona o período de tempo para o o lucro.	2.	O sistema calcula o lucro no período de tempo selecionado.
		3.	O sistema gera o lucro e mostra-o através de uma mensagem ao administrador.
FLUXO DE EXCEÇÃO		(2)	Não há dados disponíveis para o período selecionado (Passo 2)
		2.1.	O sistema informa que não há dados para o período especificado.
		(1)	Período de tempo inválido (Passo 1)
		1.1.	O sistema verifica que o período de tempo inserido pelo administrador é inválido.
		1.2.	O sistema informa o administrador e solicita que introduza um período que seja válido.

Figura 3.10: Use Case: Gera Lucros

▪ Use Cases Funcionário:

1. Montar Quebra-Cabeças

Um funcionário com acesso autorizado a uma encomenda pode utilizar o sistema para montar um quebra-cabeças associado a essa encomenda. Este processo assegura que as encomendas dos clientes sejam preparadas corretamente. Para iniciar, o funcionário acessa a encomenda e seleciona a opção de montagem do quebra-cabeças. O sistema fornece o manual necessário para auxiliar o funcionário no processo de montagem. Após a conclusão, o sistema registra que o quebra-cabeças foi montado com sucesso. Se houver peças em falta durante a montagem, o sistema informa o funcionário sobre a impossibilidade de concluir o processo devido à ausência de peças correspondentes. Além disso, caso a montagem seja interrompida por qualquer motivo referente ao funcionário, o mesmo pode optar por guardar o progresso como uma montagem parcial. Nesse caso, o sistema registra o estado atual como "montagem parcial", podendo o funcionário retomar a montagem quando entender.

Este use case é validado pelos requisitos funcionais 7 e 12.

USE CASE:	Montar Quebra Cabeças		
DESCRIÇÃO:	Permitir que um funcionário monte um quebra-cabeças para uma encomenda.		
CENÁRIOS:	Montar quebra-cabeças de cliente.		
PRÉ-CONDIÇÃO:	O funcionário está autenticado e tem acesso à encomenda.		
PÓS-CONDIÇÃO:	O quebra-cabeça é montado e registrado na encomenda.		
	Ator		Sistema
FLUXO NORMAL:	1. O funcionário acessa a encomenda e seleciona a opção para montar o quebra-cabeças.		
	2. O funcionário segue as instruções para montar o quebra-cabeças.		
		3.	O sistema registra que o quebra-cabeças foi montado com sucesso.
FLUXO DE EXCEÇÃO		(1)	Existem peças em falta em stock para a montagem. (Passo 2)
		2.1.	O sistema informa que o quebra-cabeça não pode ser concluído devido a falta de peças correspondentes.
		(2)	Interrupção na montagem. (Passo 2)
		2.1.	O funcionário reporta que não terminou a montagem de momento e guarda o estado como montagem parcial.
		2.1.	O sistema registra o estado atual da montagem como montagem parcial.

Figura 3.11: Use Case: Montar quebra-cabeças.

2. Alterar Estado da Encomenda

Um funcionário autorizado pode utilizar o sistema para alterar o estado de uma encomenda. Este processo garante que o estado das encomendas seja atualizado de forma precisa. Para iniciar, o funcionário seleciona a encomenda a ser atualizada e escolhe um novo estado entre as opções disponíveis: "não montada", "parcialmente montada", "montada" ou "enviada". O sistema valida a escolha do estado e registra a alteração. Após o registro, o sistema confirma ao funcionário que a alteração foi realizada com sucesso. Se o funcionário tentar selecionar um estado inválido ou não permitido para a encomenda, o sistema deteta a inconsistência e informa que a escolha não é válida. Nessa situação, o sistema orienta o funcionário a escolher um estado correto e válido para continuar o processo.

Este use case é validado pelo requisito funcional 11.

USE CASE:	Alterar Estado Encomenda		
DESCRIÇÃO:	Permitir que um funcionário altere o estado de uma encomenda.		
CENÁRIOS:	Atualizar estado da encomenda.		
PRÉ-CONDIÇÃO:	A encomenda existe e encontra-se num estado possível de editar.		
PÓS-CONDIÇÃO:	O estado da encomenda é atualizado com sucesso.		
	Ator	Sistema	
FLUXO NORMAL:	<ol style="list-style-type: none"> 1. O funcionário seleciona a encomenda a ser atualizada. 2. O funcionário escolhe o novo estado para a encomenda. (não montada, parcialmente montada, montada, enviada) 	<ol style="list-style-type: none"> 3. O sistema regista a alteração de estado. 4. O sistema confirma que o estado foi atualizado com sucesso. 	
FLUXO DE EXCEÇÃO		<ol style="list-style-type: none"> (1) O funcionário tentou alterar para um estado inválido. (Passo 2) 2.1. O sistema informa que o estado não é válido para esta encomenda. 2.2. O sistema solicita ao funcionário um estado válido para a mesma. 	

Figura 3.12: Use Case: Alterar Estado da Encomenda.

3. Criar Peça

Um funcionário autenticado, designado para a fabricação de peças, pode aceder ao sistema para criar uma nova peça a ser utilizada na montagem de um quebra-cabeças. Para iniciar o processo, o funcionário insere no sistema as especificações da peça, como o tipo, dimensões, material, e quantidade a ser produzida. O sistema verifica se todos os campos obrigatórios foram preenchidos corretamente e se as especificações estão de acordo com os padrões de qualidade definidos pela empresa. Caso todos os dados sejam válidos, o sistema regista a criação da peça na base de dados e atualiza o stock, adicionando a nova quantidade de peças disponíveis para utilização. Após o registo, o sistema informa o funcionário de que a peça foi criada com sucesso e está pronta a ser utilizada na montagem do quebra-cabeças cuja essa mesma faz parte. Se algum dado estiver incorreto ou faltar alguma especificação essencial, o sistema exibe uma notificação de erro, informando o funcionário sobre o problema e orientando-o a corrigir as informações para prosseguir com a criação da peça.

Este use case é validado pelo requisito funcional 7.

USE CASE:	Criar Peça		
DESCRIÇÃO:	Permitir que um funcionário autenticado crie uma nova peça para ser utilizada na montagem de um quebra-cabeças.		
CENÁRIOS:	Criar uma nova peça.		
PRÉ-CONDIÇÃO:	O funcionário está autenticado e possui autorização para criar peças.		
PÓS-CONDIÇÃO:	A peça é criada, registada na base de dados e criada uma secção no stock para a mesma.		
	Ator	Sistema	
FLUXO NORMAL:	<ol style="list-style-type: none"> 1. O funcionário acede à funcionalidade de criação de peças. 2. O funcionário insere as especificações da peça. (tipo, tamanho, material, quantidade) 	<ol style="list-style-type: none"> 3. O sistema valida as informações fornecidas. 4. O sistema regista a peça na base dados e no stock. 5. O sistema informa o funcionário com uma mensagem de sucesso. 	
FLUXO DE EXCEÇÃO		<ol style="list-style-type: none"> (1) O funcionário insere dados inválidos ou incompletos. (Passo 2) 2.1. O sistema informa o funcionário que os dados fornecidos são inválidos ou incompletos nos campos. 2.2. O sistema solicita a correção das respetivas especificações para continuar o processo. 	

Figura 3.13: Use Case: Criar Peça.

4. Preparar Encomenda

Um funcionário autenticado, designado para a preparação de encomendas, pode aceder ao sistema para realizar o processo de preparação de uma encomenda específica. Este processo envolve a montagem de quebra-cabeças associados à encomenda, garantindo que os produtos finais atendem aos padrões de qualidade estabelecidos pela empresa. Para iniciar, o funcionário seleciona a encomenda que deseja preparar. O sistema apresenta a lista de quebra-cabeças incluídos nessa encomenda, juntamente com os manuais de preparação detalhados para cada quebra-cabeças. O funcionário segue as instruções fornecidas no manual para preparar cada quebra-cabeças. Após finalizar a preparação de todos os quebra-cabeças, o sistema regista automaticamente o estado da encomenda como "Preparada" e informa o funcionário de que a mesma está pronta para a próxima etapa, que geralmente será a entrega. Caso o funcionário opte por preparar apenas parte dos quebra-cabeças da encomenda, o sistema atualiza o estado para "Parcialmente Preparada", permitindo que a preparação seja concluída posteriormente. Se a encomenda já estiver preparada ou num estado que não permita alterações, o sistema informa o funcionário de que não é possível iniciar a preparação.

Este use case é validado pelos requisitos funcionais 4 e 11.

USE CASE:	Preparar Encomenda		
DESCRIÇÃO:	Permitir que um funcionário prepare uma encomenda.		
CENÁRIOS:	Iniciar preparação da encomenda.		
PRÉ-CONDIÇÃO:	A encomenda está num estado editável.		
PÓS-CONDIÇÃO:	A encomenda é marcada como montada.		
	Ator	Sistema	
FLUXO NORMAL:	1. O funcionário seleciona a encomenda para iniciar a montagem.	2. O sistema exibe a lista de quebra-cabeças associados a essa encomenda e o manual de montagem respetivo de cada um.	
	3. O funcionário monta os quebra-cabeças da encomenda.	4. O sistema regista o estado da encomenda como preparada.	
		5. O sistema informa o funcionário que a encomenda foi preparada e está pronta para entrega.	
FLUXO DE EXCEÇÃO		(1) Encomenda já foi preparada (Passo 1)	
		2.1. O sistema informa que a encomenda já está em um estado que não permite mais alterações.	
FLUXO ALTERNATIVO		(3) O funcionário prepara apenas uma parte dos quebra-cabeças da encomenda.	
		3.1 O sistema atualiza o estado da encomenda para parcialmente preparada.	
		3.2 O sistema informa o funcionário que a preparação parcial foi registada.	
	3.3 Quando o funcionário estiver pronto para concluir a preparação restante, o mesmo pode retornar à encomenda e repetir o fluxo normal ou o fluxo alternativo até que o estado da encomenda seja preparada.		

Figura 3.14: Use Case: Preparar Encomenda

▪ Use Cases Gerais:

1. Fazer Login

Qualquer utilizador pode aceder ao sistema para se autenticar. Este processo garante que apenas utilizadores autorizados conseguem aceder ao sistema e às funcionalidades correspondentes ao seu perfil. Para iniciar, o utilizador insere as suas credenciais, incluindo o nome de utilizador e a palavra-passe. O sistema valida os dados, verifica a sua correspondência e caso correspondam o utilizador autentica-se no sistema. No entanto, se os dados forem inválidos, o sistema informa o utilizador de que os dados inseridos não são válidos, imprimindo uma mensagem a indicar ao utilizador para tentar novamente.

Este use case é validado pelo requisito funcional 1.

USE CASE:	Fazer Login		
DESCRIÇÃO:	Permite que qualquer tipo de utilizador registe uma conta.		
CENÁRIOS:	Autenticação de um utilizador. (qualquer tipo)		
PRÉ-CONDIÇÃO:	True		
PÓS-CONDIÇÃO:	Sistema autentica o utilizador com sucesso.		
	Ator		Sistema
FLUXO NORMAL:	1. O utilizador insere as credencias correspondentes.	2. O sistema valida as credenciais.	
		3. O sistema informa o utilizador que foi autenticado com sucesso.	
FLUXO DE EXCEÇÃO		(1) As informações são inválidas. (passo 2)	
		2.1. O sistema informa que os dados não são válidos.	

Figura 3.15: Use Case: Fazer Login

2. Fazer Logout

Um utilizador autenticado pode encerrar a sua sessão no sistema ao utilizar esta funcionalidade. Este processo garante que a sessão do utilizador seja finalizada. Para iniciar, o utilizador seleciona a opção de logout no sistema. Em seguida, o sistema encerra a sessão ativa e remove o acesso do utilizador, garantindo que não seja possível continuar a interagir com o sistema sem nova autenticação. Após concluir o processo, o sistema exibe uma mensagem de confirmação a indicar que o utilizador encerrou com sucesso a sessão.

Este use case é validado pelo requisito funcional 1.

USE CASE:	Fazer Logout		
DESCRIÇÃO:	Permitir que o utilizador saia do sistema.		
CENÁRIOS:	Encerrar sessão.		
PRÉ-CONDIÇÃO:	O utilizador estar autenticado no sistema.		
PÓS-CONDIÇÃO:	A sessão do utilizador é encerrada.		
	Ator		Sistema
FLUXO NORMAL:	1. O utilizador seleciona a opção de logout.	2. O sistema encerra a sessão do utilizador.	
		3. O sistema retira o utilizador do sistema.	
		4. O sistema informa o utilizador que a sessão foi encerrada com sucesso.	

Figura 3.16: Use Case: Fazer Logout

Vão ser apresentados também a seguir os respetivos diagramas de atividade de cada entidade (e também dos relacionamentos) para facilitarem o processo de explicação do que faz verdadeiramente o sistema em questão na prática.

- Este diagrama de atividades representa as ações de um administrador no sistema, incluindo o login, criação de contas para funcionários e gestão de lucros. Após o login, o administrador pode optar por gerir lucros (escolhendo um período de tempo válido) ou criar uma nova conta de funcionário. O sistema calcula os lucros para o período selecionado e gera relatórios detalhados.

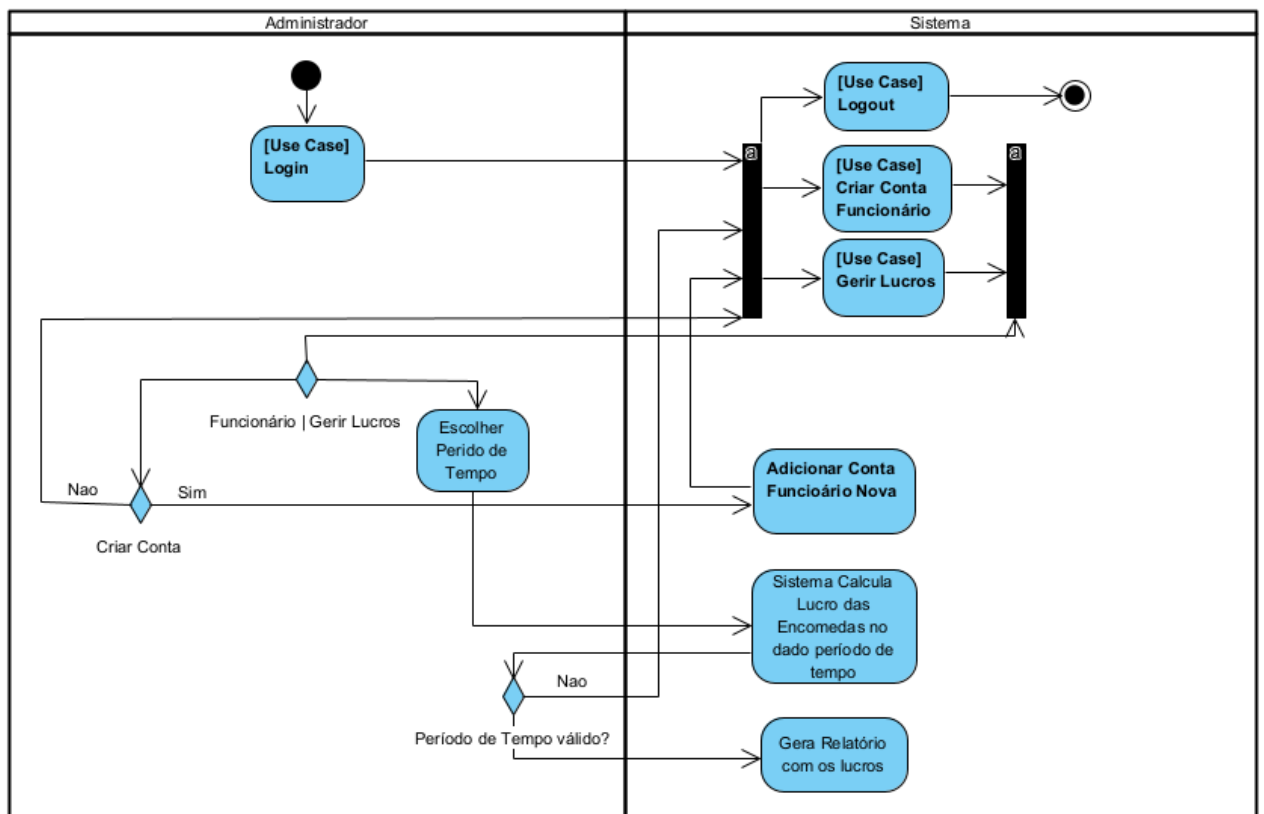


Figura 3.17: Diagrama de Atividade do Administrador

- Este diagrama de atividades descreve as ações de um utilizador no sistema. O utilizador pode realizar login ou registar uma conta. Após acessar, pode visualizar encomendas, produtos (quebra-cabeças/peças) ou editar uma encomenda existente (e possível de editar). Caso escolha adicionar um produto, o sistema apresenta os detalhes e finaliza o processo ao confirmar a adição do item à encomenda.

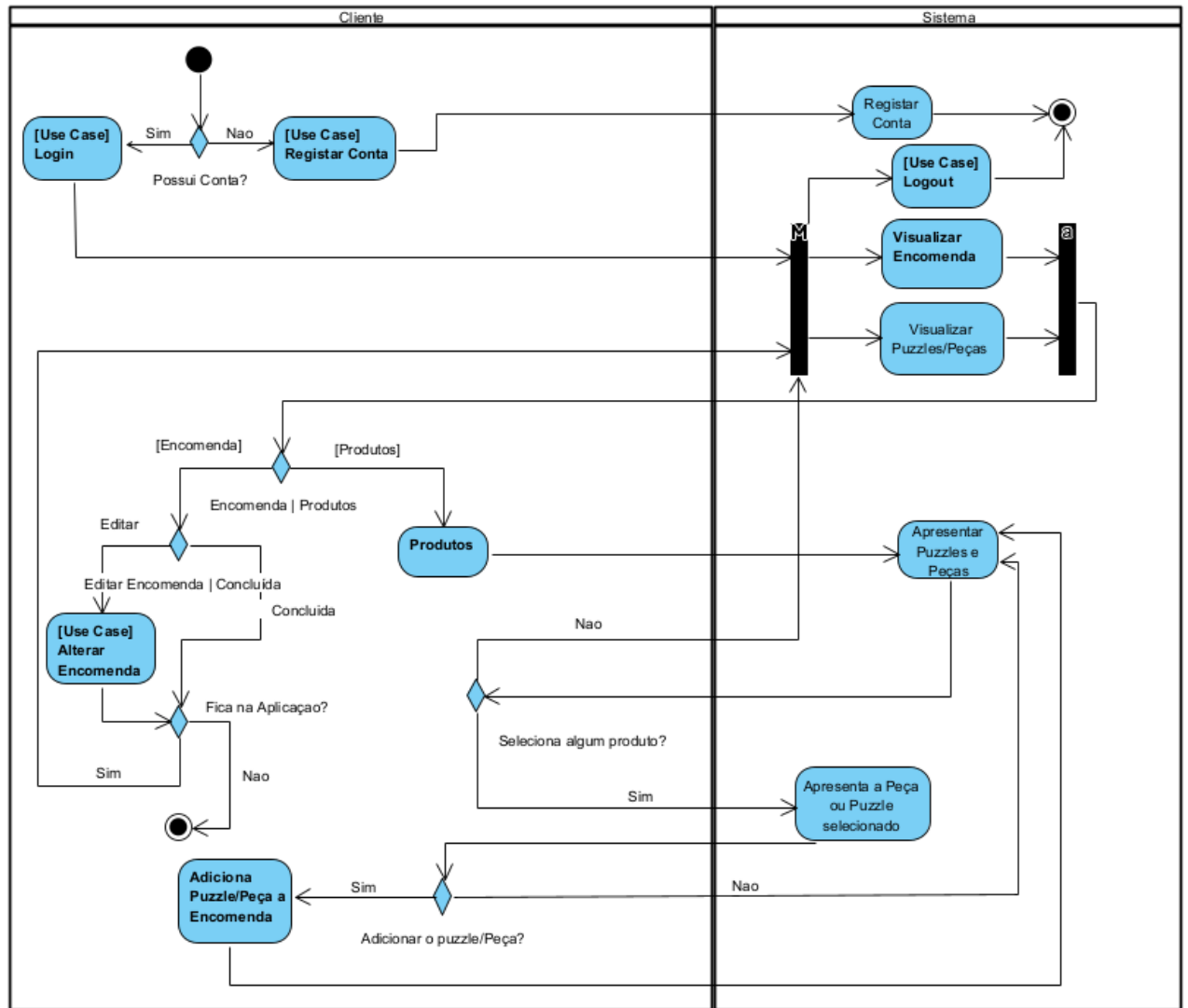


Figura 3.18: Diagrama de Atividade do Utilizador

- Este diagrama do funcionário associado à encomenda envolve ações como login, visualização e seleção de encomendas, montagem ou alteração de seu estado, criação de peças (se necessário e com materiais disponíveis), atualização do stock e validação das condições para concluir a encomenda. Caso tudo esteja correto, o funcionário pode finalizar a montagem do quebra-cabeças relacionado à encomenda.

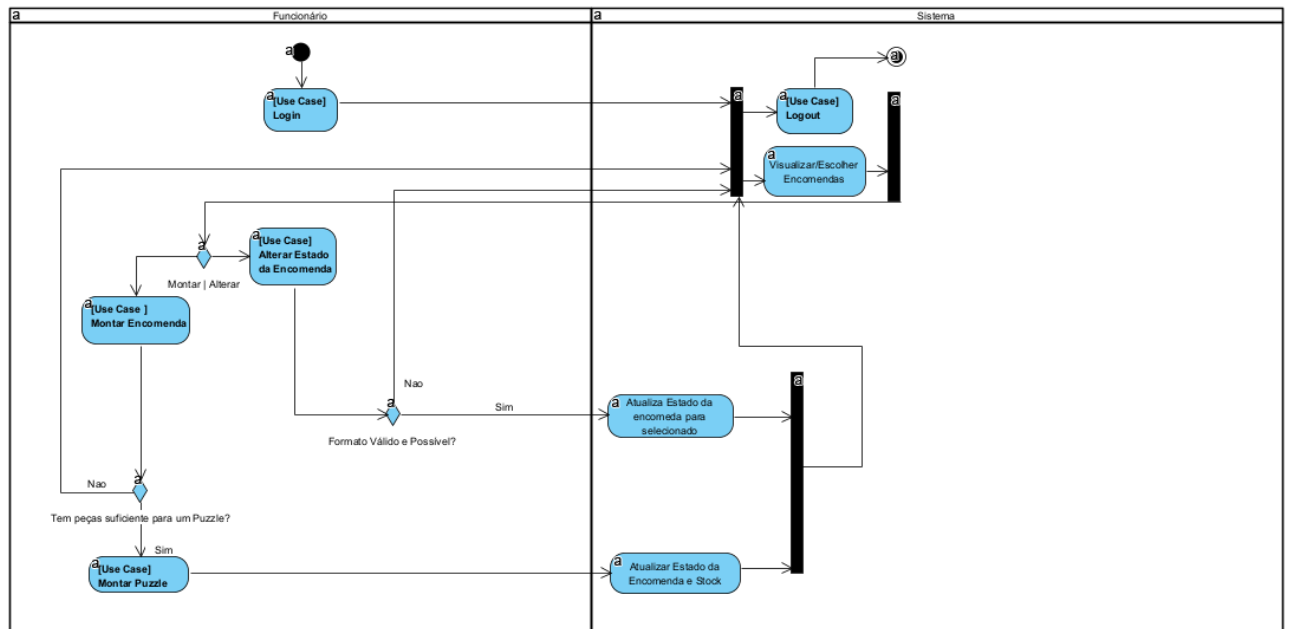


Figura 3.19: Diagrama de Atividade do Funcionário em relação á Encomenda

- Este diagrama do funcionário associado à peça inclui ações como login, verificação de peças disponíveis, criação de novas peças caso necessário (tendo sempre em conta os materiais disponíveis), atualização do stock e validação de peças para utilização em encomendas ou montagem de quebra-cabeças.

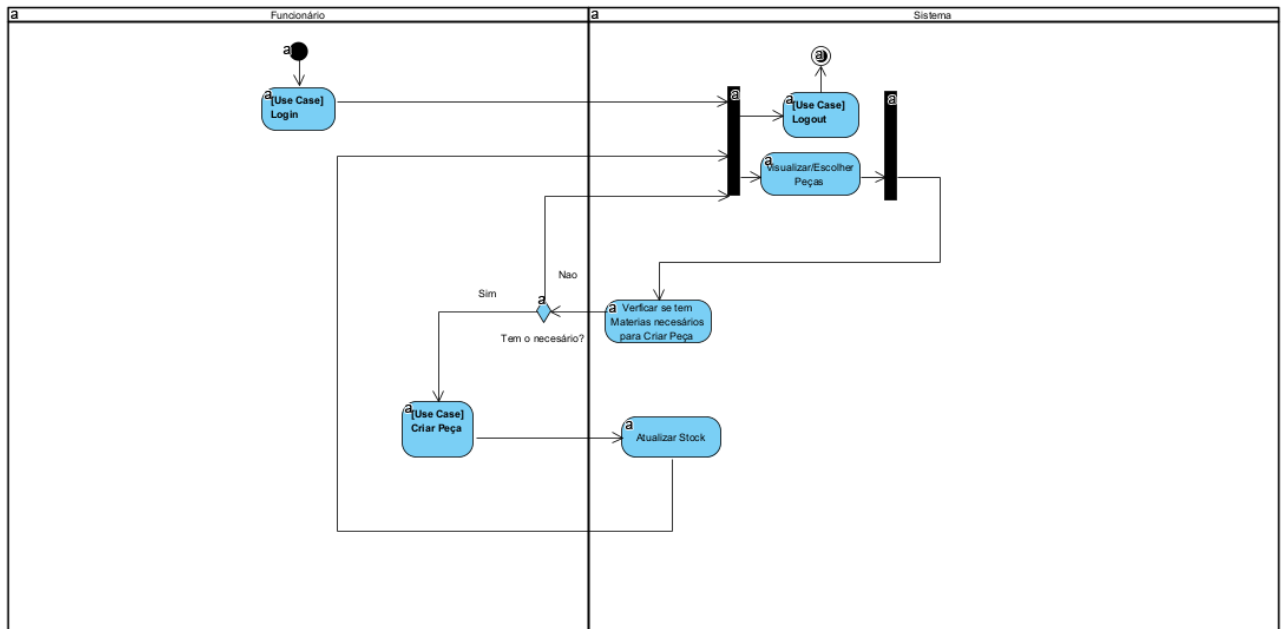


Figura 3.20: Diagrama de Atividade do Funcionário em relação à Peça.

4 Conceção do Sistema de Dados

Este capítulo apresenta uma visão geral da estrutura de dados do sistema, sendo intrínseco no âmbito do projeto.

A estrutura de dados é essencial para armazenar, organizar e gerir informações associadas a utilizadores, encomendas, produtos, configurações e outros elementos fundamentais. Desempenha um papel vital ao garantir uma experiência segura, eficiente e confiável para todos os utilizadores da plataforma.

Ao longo deste capítulo, será detalhada a arquitetura de dados do sistema, o design da base de dados, a modelação das entidades, os relacionamentos entre tabelas e as estratégias adotadas para armazenamento de dados.

4.1 Apresentação geral da estrutura do sistema de dados

No modelo a seguir representado são apresentadas as várias entidades essenciais para gerir o sistema da melhor forma possível.

As entidades representadas são **Encomenda**, **Funcionário**, **Utilizador**, **Peça** e **Quebra-Cabeças**.

Devido aos relacionamentos entre **Encomenda** e **Peça**, surge a tabela **Peças da Encomenda**, e através do relacionamento entre **Encomenda** e **Quebra-Cabeças** surge a tabela **Quebra-Cabeças da Encomenda**.

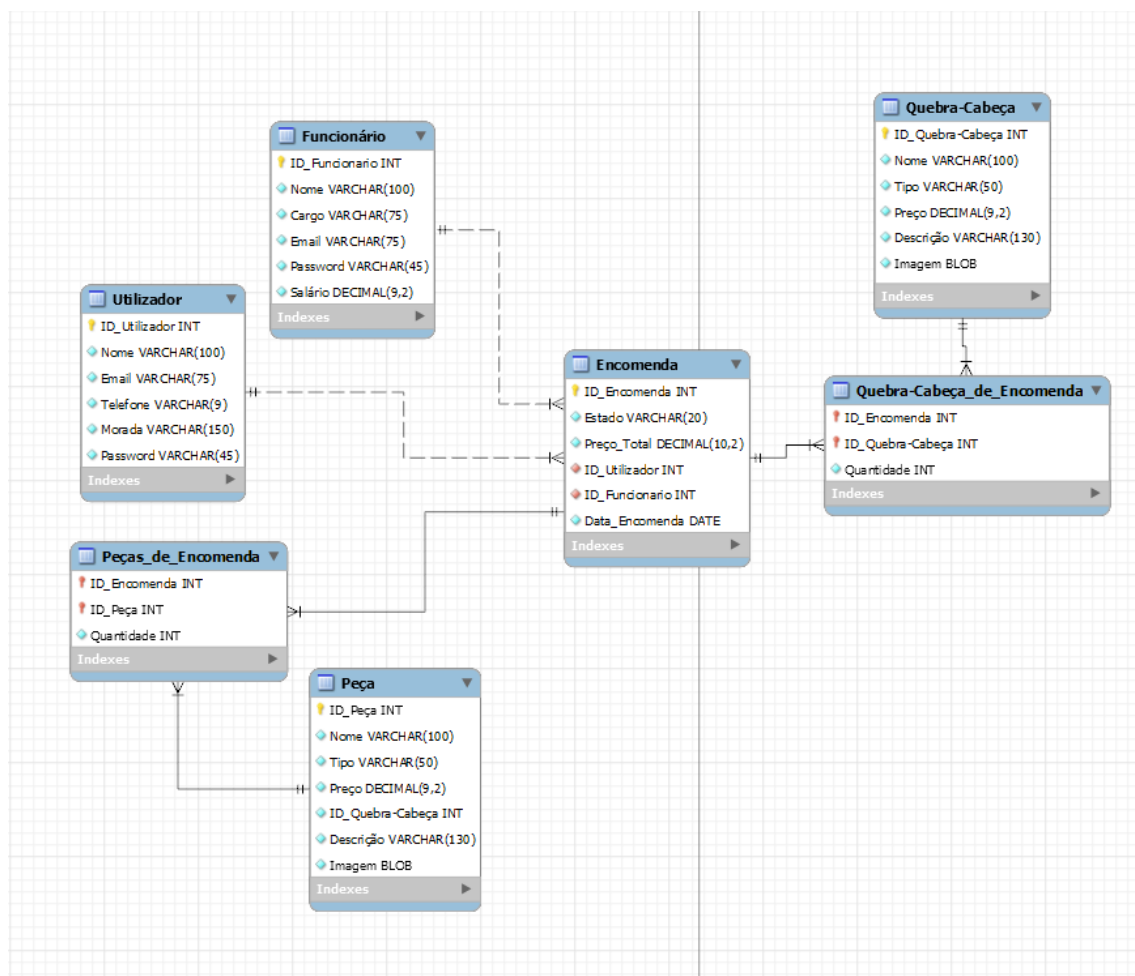


Figura 4.1: Modelo Lógico do Sistema

4.2 Descrição detalhada dos vários elementos de dados e seus relacionamentos

Para melhor compreensão das nossas entidades e relacionamentos elaboraram-se tabelas para que se explique ao detalhe o que consiste cada um.

4.2.1 Utilizador

A entidade **Utilizador** e os seus atributos servem para caracterizar os clientes em si, quem vai realmente comprar os itens/produtos e é identificado pela chave primária ID.

Entidade	Atributo	Tipo de Dados	Bytes	Descrição	Exemplo
Utilizador	ID	INT	4	ID de um único utilizador	10
	Nome	VAR-CHAR(100)	102	Designação do utilizador	Joaquim
	Email	VAR-CHAR(75)	77	Email de acesso do utilizador	quim@gmail.com
	Telefone	VAR-CHAR(9)	11	Número de telefone do utilizador	912345678
	Morada	VAR-CHAR(150)	152	Morada do utilizador para encomendas	Rua de Monsanto
	Password	VAR-CHAR(45)	47	Credencial secreta do utilizador	joa-quim123!

Tabela 4.1: Tabela da entidade Utilizador.

4.2.2 Funcionário

A entidade **Funcionário** e os seus atributos servem para caracterizar os operários, quem vai realmente fabricar os itens/produtos e é identificado pela chave primária ID.

Entidade	Atributo	Tipo de Dados	Bytes	Descrição	Exemplo
Funcionário	ID	INT	4	ID de um único funcionário	12
	Nome	VAR-CHAR(100)	102	Designação do funcionário	Eduardo
	Cargo	VAR-CHAR(75)	77	Cargo do funcionário	Operário de linha
	Email	VAR-CHAR(75)	77	Email de acesso do funcionário	edu@gmail.com
	Password	VAR-CHAR(45)	47	Credencial secreta do funcionário	edu123789!
	Salário	DECIMAL(9,2)	5	Ordenado do funcionário	875,25

Tabela 4.2: Tabela da entidade Funcionário.

4.2.3 Encomenda

A entidade **Encomenda** e os seus atributos servem para caracterizar a encomenda em si, o conteúdo, as suas características e é identificado pela chave primária ID.

Entidade	Atributo	Tipo de Dados	Bytes	Descrição	Exemplo
Encomenda	ID	INT	4	ID de uma única encomenda	87
	Estado	VAR-CHAR(20)	22	Estado da encomenda	Preparada
	Preço Total	DECIMAL(10,2)	6	Preço total da encomenda	30,99
	ID Utilizador	INT	4	ID do utilizador associado á encomenda	20
	ID Funcionário	INT	4	ID do funcionário associado á encomenda	29
	Data Encomenda	DATE	5	Data da encomenda	06/11/2028

Tabela 4.3: Tabela da entidade Encomenda.

4.2.4 Peça

A entidade **Peça** e os seus atributos servem para caracterizar a peça em si, as suas características e é identificado pela chave primária ID.

Entidade	Atributo	Tipo de Dados	Bytes	Descrição	Exemplo
Peça	ID	INT	4	ID de uma única peça	99
	Nome	VAR-CHAR(100)	102	Nome da peça	Bloco
	Tipo	VAR-CHAR(50)	52	Tipo da peça	Metálica
	Preço	DECIMAL(9,2)	5	Preço associado á peça	2,99
	ID Quebra-Cabeça	INT	4	ID do quebra-cabeça associado á peça	13
	Descrição	VAR-CHAR(130)	132	Descrição da peça em questão	Peça metálica em forma de bloco
	Imagem	Blob	???	Imagem da peça	???

Tabela 4.4: Tabela da entidade Peça.

4.2.5 Quebra-Cabeça

A entidade **Quebra-Cabeça** e os seus atributos servem para caracterizar a o puzzle em si, as suas características e é identificado pela chave primária ID.

Entidade	Atributo	Tipo de Dados	Bytes	Descrição	Exemplo
Quebra-Cabeça	ID	INT	4	ID de um único quebra-cabeça	23
	Nome	VAR-CHAR(100)	102	Nome do quebra-cabeça	Pokemon
	Tipo	VAR-CHAR(50)	52	Tipo do quebra-cabeça	Puzzle
	Preço	DECIMAL(9,2)	5	Preço associado ao quebra-cabeça	25,99
	Descrição	VAR-CHAR(130)	132	Descrição do quebra-cabeça em questão	Puzzle do anime Pokemon
	Imagem	Blob	???	Imagem do quebra-cabeça	???

Tabela 4.5: Tabela da entidade Quebra-Cabeça.

4.2.6 Quebra-Cabeça de Encomenda

A entidade **Quebra-Cabeça de Encomenda** e os seus atributos servem para caracterizar o puzzle que pertence a uma encomenda, as suas características e é identificado pela chave primária composta ID Encomenda e ID Quebra-Cabeça.

Entidade	Atributo	Tipo de Dados	Bytes	Descrição	Exemplo
Quebra-Cabeça de Encomenda	ID Encomenda	INT	4	ID de uma encomenda	45
	ID Quebra-Cabeça	INT	4	ID de um quebra-cabeça	56
	Quantidade	INT	4	Quantidade de quebra-cabeças na encomenda	3

Tabela 4.6: Tabela da entidade Quebra-Cabeça de Encomenda.

4.2.7 Peça de Encomenda

A entidade **Peça de Encomenda** e os seus atributos servem para caracterizar a peça que pertence a uma encomenda, as suas características e é identificado pela chave primária composta ID Peça e ID Encomenda.

Entidade	Atributo	Tipo de Dados	Bytes	Descrição	Exemplo
Peça de Encomenda	ID Peça	INT	4	ID de uma peça	92
	ID Encomenda	INT	4	ID de uma encomenda	76
	Quantidade	INT	4	Quantidade de peças na encomenda	7

Tabela 4.7: Tabela da entidade Peça de Encomenda.

5 Esboço dos Interfaces do Sistema

A interface do utilizador desempenha um papel fundamental nos sistemas de software, podendo até definir o sucesso do projeto. Um sistema de software que tenha uma boa estrutura, mas que fique a perder na interface do utilizador pode comprometer completamente o projeto. Tendo isto em consideração, pretende-se que a interface do sistema a ser desenvolvido seja intuitiva para facilitar o uso dos clientes e dos trabalhadores e, para além disso, que tenha um design apelativo aos clientes. Para tal, a equipa da SimpleSystems preocupou-se em desenvolver mockups iniciais para representar o design da interface que se pretende.

5.1 Caracterização das interfaces

Em seguida, são apresentados os mockups iniciais, desenvolvido com o auxílio do site Draw.io, das interfaces que representam as principais funcionalidades do sistema. Decidimos fazer uma interface para o cliente, mais apelativa e outra para o funcionário, mais simplificada para facilitar a aprendizagem de uso do sistema e a própria utilização.

5.1.1 Página de Login e Registo do Cliente

Ao aceder à plataforma, esta é a primeira vista do cliente. A simplicidade deste modelo torna o login intuitivo. Para além disso, do lado esquerdo temos uma pequena apresentação da empresa ComplicadaMente, incluindo o logótipo, o slogan e uma imagem do criador da mesma com a estrela de Galileu na mão.

Esta interface é validada pelo use case de utilizador "Fazer Login" e "Registar Conta".



Figura 5.1: Mockup página de login/registo

5.1.2 Página principal

Após o login do cliente, é apresentada a página principal. Aqui são apresentados os itens em destaque, uma caixa de pesquisa para o cliente poder procurar os produtos que quiser e um botão para aceder ao carrinho que ainda está por ser definido. Para além disso, na barra superior temos no canto esquerdo um botão de menu e no canto direito um botão com o símbolo de utilizador. As funcionalidades dos mesmos estão apresentadas mais abaixo.



Figura 5.2: Mockup página principal

5.1.3 Página de resultados

Caso o cliente faça uma pesquisa, é apresentada uma página de com os vários produtos que estão relacionados com o que o cliente escreveu na barra de pesquisa. É possível fazer scroll nesta página para visualizar todos os itens relacionados. O exemplo deste scroll está seguidamente apresentado.

Esta interface é validada pela ação presente no diagrama de atividades do utilizador "Apresentar puzzles e peças".



Figura 5.3: Mockup resultados de pesquisa

5.1.4 Scroll da página de resultados

Este mockup é a continuação do anterior. Os clientes podem fazer scroll para ver todos os produtos relacionados com a pesquisa.



Figura 5.4: Mockup scroll de resultados de pesquisa

5.1.5 Botão menu

Ao carregar no botão de menu no canto superior esquerdo é aberta uma aba com várias opções para facilitar a pesquisa de produtos.



Figura 5.5: Mockup botão menu

5.1.6 Botão user

Ao carregar no botão de user é aberta uma página que mostra o perfil do utilizador com uma aba do lado esquerdo com as opções de ver os dados pessoais, as encomendas feitas e o carrinho. Para além disso, na mesma aba há ainda a opção de terminar sessão ou de eliminar a conta. Daqui o cliente pode depois ter acesso a alterar os dados da conta, ver o estado das encomendas feitas e aceder ao carrinho. Nesta página há ainda um botão em forma de X que leva à página principal.

Esta interface é validada pelo use case de utilizador "Alterar Dados da Conta".



Figura 5.6: Mockup botão user

5.1.7 Selecionar produto

Quando um cliente seleciona um produto, é aberta uma página que apresenta os detalhes do mesmo e permite que este seja adicionado ao carrinho por unidade e/ou por lote. Existe ainda um botão que permite que o cliente veja mais produtos daquele tipo. Ao carregar nesse botão é aberta a página de pesquisa apresentada anteriormente, mas em vez da pesquisa ser feita com base no que o cliente escreveu, é feita com base nos dados em comum que o produto tem com outros.

Esta interface é validada pela ação presente no diagrama de atividades do utilizador "Selecionar Produto" e "Apresenta a peça ou puzzle selecionado".



Figura 5.7: Mockup vista produto

5.1.8 Carrinho

Quando um cliente seleciona a opção "o meu carrinho", quer seja através do menu user ou do botão na página principal, é aberta uma página que mostra todos os itens que foram adicionados ao carrinho, com o respetivo ID, quantidade adicionada, preço unitário e o preço total para a quantidade do item. É possível adicionar ou retirar unidades de um produto. Nesta página é ainda possível voltar à pagina principal através de um botão em forma de seta no canto superior esquerdo ou prosseguir para a finalização da compra através de um botão no canto inferior direito. Para que o cliente consiga ver todos os produtos do carrinho no caso de serem vários está disponível o scroll da página.

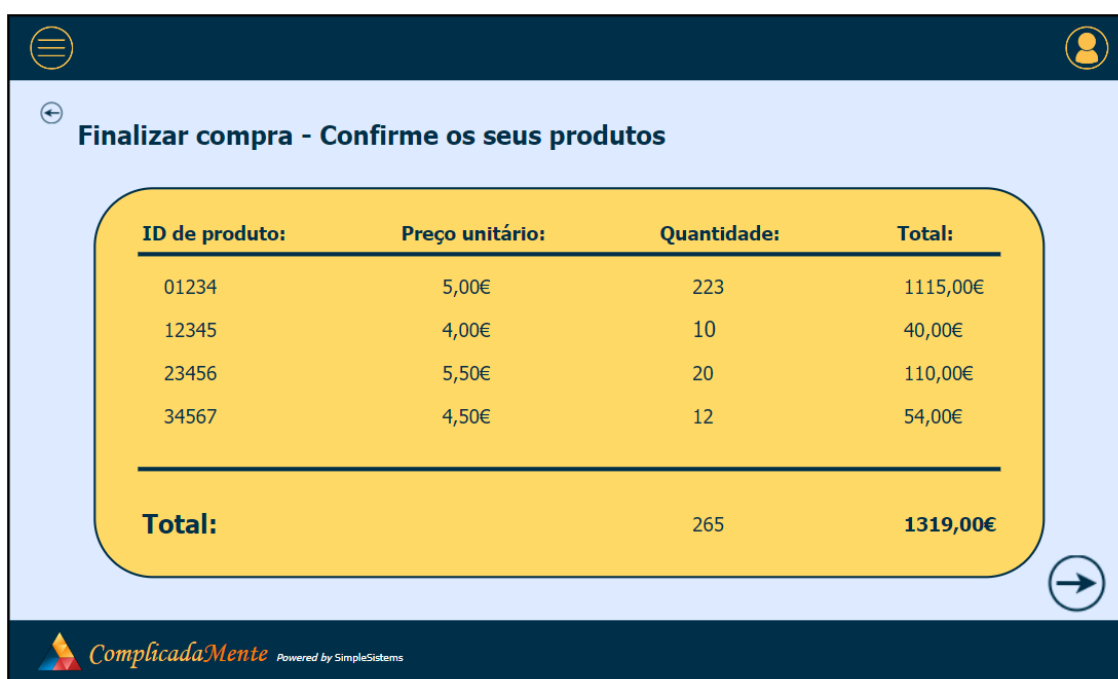


Figura 5.8: Mockup carrinho

5.1.9 Finalizar Compra - Confirmar produtos

Continuando o seguimento do mockup anterior, quando um cliente escolhe finalizar a sua compra começam uma série de passos para tal. O primeiro, representado neste mockup, é confirmar os produtos. Aparece um lista com os ID's dos produtos, o preço unitário, a quantidade e o total monetário da quantidade selecionada de cada produto. No final da lista, fica apresentado o total de produtos no carrinho e o total monetário de todos os produtos selecionado tendo em conta as respetivas quantidades. O cliente pode continuar para o próximo passo clicando na seta no canto inferior direito, ou voltar para o carrinho clicando na seta no canto superior esquerdo.

Esta interface é validada pelo ação no diagrama de atividades do utilizador "Encomenda Concluída".



ID de produto:	Preço unitário:	Quantidade:	Total:
01234	5,00€	223	1115,00€
12345	4,00€	10	40,00€
23456	5,50€	20	110,00€
34567	4,50€	12	54,00€
Total:		265	1319,00€

Figura 5.9: Mockup confirmar produtos

5.1.10 Finalizar Compra - Confirmar dados pessoais

O segundo passo para a finalização da encomenda é a confirmação dos dados pessoais. Aqui o cliente pode seleccionar se quer fatura com número de contribuinte e ver os dados pessoais para serem confirmados. Assim, caso alguma encomenda seja feita com os dados desatualizados não será culpa da ComplicadaMente nem da SimpleSystems, pois o cliente é avisado do estado dos dados antes da finalização da encomenda.

Finalizar compra - Dados Pessoais

Nome do Cliente: Orlando Belo

NIF: 012345678

Quer fatura com nº de contribuinte? Sim ☒ Não ☐

Telemóvel: 912 345 678

Email: obelodiuminho@gmail.com

ComplicadaMente Powered by SimpleSystems

Figura 5.10: Mockup confirmar dados pessoais

5.1.11 Finalizar Compra - Seleccionar Morada

Para que as encomenda sejam entregues, é fundamental que haja uma morada de entrega. Por isso, ao prosseguir com a finalização da encomenda, o seguinte passo é seleccionar a morada. O cliente pode seleccionar a morada guardada no sistema ou escolher usar outra morada. Ao escolher a segunda opção, os dados da morada têm de ser todos obrigatoriamente preenchidos antes de prosseguir.

Finalizar compra - Morada

Usar a minha morada

Rua: Av. da Liberdade N° da Porta: n21, 5º dt
Cidade: Braga Distrito: Braga
Código Postal: 4715-085 País: Portugal

Usar outra morada

Rua: «Rua diferente» N° da Porta: «Nº diferente»
Cidade: «Cidade diferente» Distrito: «Distrito diferente»

ComplicadaMente Powered by SimpleSystems

Figura 5.11: Mockup seleccionar morada

5.1.12 Finalizar Compra - Selecionar Método de pagamento

Outro passo fundamental é o método de pagamento, pois não se pode começar a produção de uma encomenda sem o pagamento ter sido realizado. Assim como a morada, é possível seleccionar o método de pagamento guardado ou escolher usar outro método de pagamento preenchendo os dados necessários. Após carregar no botão finalizar compra a encomenda é finalizada e já não é possível voltar atrás.

Finalizar compra - Método de pagamento

☒ Usar método de pagamento guardado

MBway: 912345678 Total: 1319,00€

Finalizar compra

☐ Usar outro método de pagamento

☐ Cartão: Nº: «Nº do cartão» CVV: «Nº CVV»

☐ Mbway: Nº: «Nº MBway»

ComplicadaMente Powered by SimpleSystems

Figura 5.12: Mockup seleccionar método de pagamento

5.1.13 Finalizar Compra - Dados da encomenda

Após a finalização da encomenda, é apresentada uma página com os dados da mesma. Para ver todos os dados da encomenda está disponibilizado o scroll da página.

Obrigado pela sua compra! - Dados da sua encomenda:

Nome do Cliente: Orlando Belo

NIF: 012345678

Quer fatura com nº de contribuinte? Sim ☒ Não ☐

Telemóvel: 912 345 678

Email: obelodiuminho@gmail.com

Nº da encomenda: 12567803

Morada de envio

ComplicadaMente Powered by SimpleSystems

Figura 5.13: Mockup dados da encomenda

5.1.14 Consultar estado de encomenda

O cliente pode consultar o estado das suas encomendas. Ao selecionar a opção "As minhas encomendas" no menu de user, aparecem as encomendas do cliente com as respetivas informações gerais por ordem de data em que foram efetuadas. Uma barra de progresso indica em percentagem quanto da encomenda é que já foi produzido e está pronto para envio. Quando a encomenda fica pronta, em vez da barra de progresso, aparece uma mensagem de como a encomenda foi enviada. Ao selecionar uma das encomendas, aparecem os dados mais pormenorizados.

Esta interface é validada pelo use case de utilizador "Alterar Encomenda" e também presente no diagrama de atividades do utilizador.

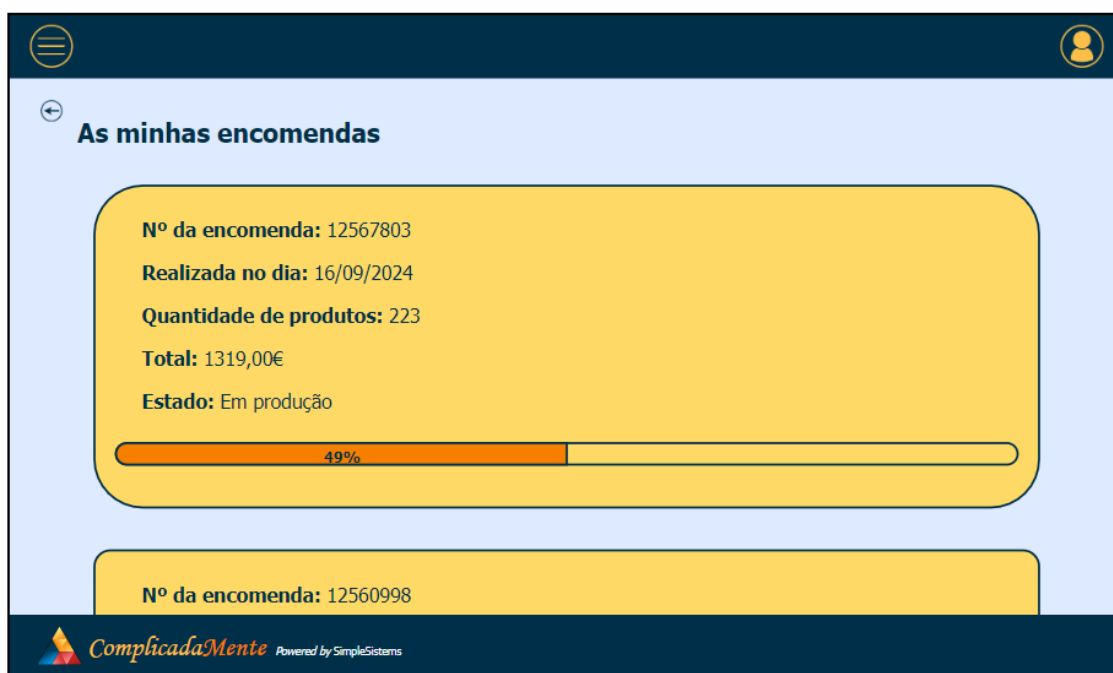


Figura 5.14: Mockup consultar encomendas

5.1.15 Consultar produto encomendado

Ao consultar os dados pormenorizados da encomenda, o utilizador pode escolher um dos produtos encomendados para ver as suas informações gerais, a quantidade que foi encomendada e a quantidade que já foi produzida. Para além disso, do lado direito da página, o cliente pode ver como é feita a montagem ou produção do produto. Para tal, os passos da montagem (ou produção no caso das peças) são apresentados em fotografias. Para poder passar os passos de um para o outro, o cliente pode fazer uso das setas à beira da fotografia, usando a da esquerda para ver o passo anterior e a da direita para ver o passo seguinte.



Figura 5.15: Mockup consultar produto encomendado

5.1.16 Selecionar tipo de funcionário

Como foi dito no início deste capítulo, foram feitos dois tipos de interfaces. A do cliente e a do funcionário. Para não haver confusões com os postos de trabalho, antes de fazer login no programa, o funcionário seleciona o seu tipo. Pode ser um funcionário de Montagem, produção de Peças ou o administrador.

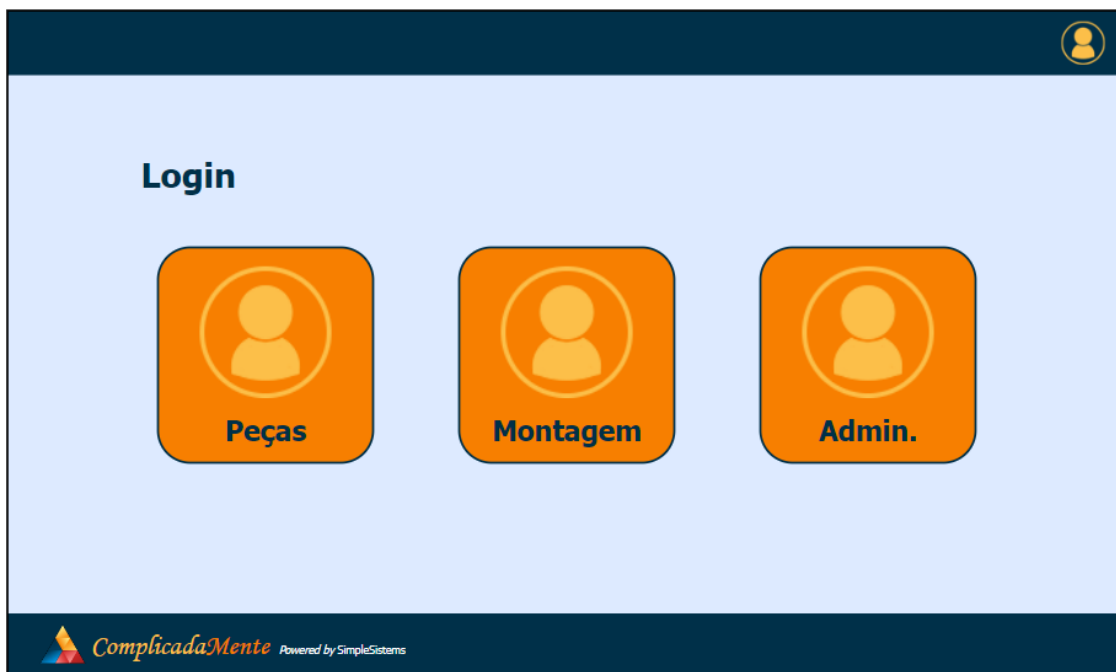


Figura 5.16: Mockup selecionar tipo de funcionário

5.1.17 Login do funcionário

Após escolher o tipo de funcionário, aparece a página de login. O funcionário tem de preencher o ID e a password com os seus dados com o teclado do visor e carregar no botão de "LOGIN" para entrar. Caso tenha escolhido o tipo de funcionário errado, pode carregar no botão de user para voltar à página de seleccionar tipo. Caso já tenha o login feito e queira dar logout também deve usar este botão.

Esta interface é validada pelo use case geral "Fazer login".

Peças

Nº de utilizador:

Password:

LOGIN

1 2 3
4 5 6
7 8 9
0 ←


 **ComplicadaMente** Powered by SimpleSystems

Figura 5.17: Mockup login do funcionário

5.1.18 Página principal do funcionário

Com o login feito, o funcionário entra na página principal tendo apenas duas possibilidades. Pode dar logout no botão de user ou pode carregar no botão de próxima peça (ou puzzle no caso da montagem). Escolhendo a segunda opção o sistema apresentará os passos de produção da peça(ou puzzle no caso da montagem).

Esta interface é validada pelo use case de funcionário "Criar peça" e "Montar quebra-cabeça".

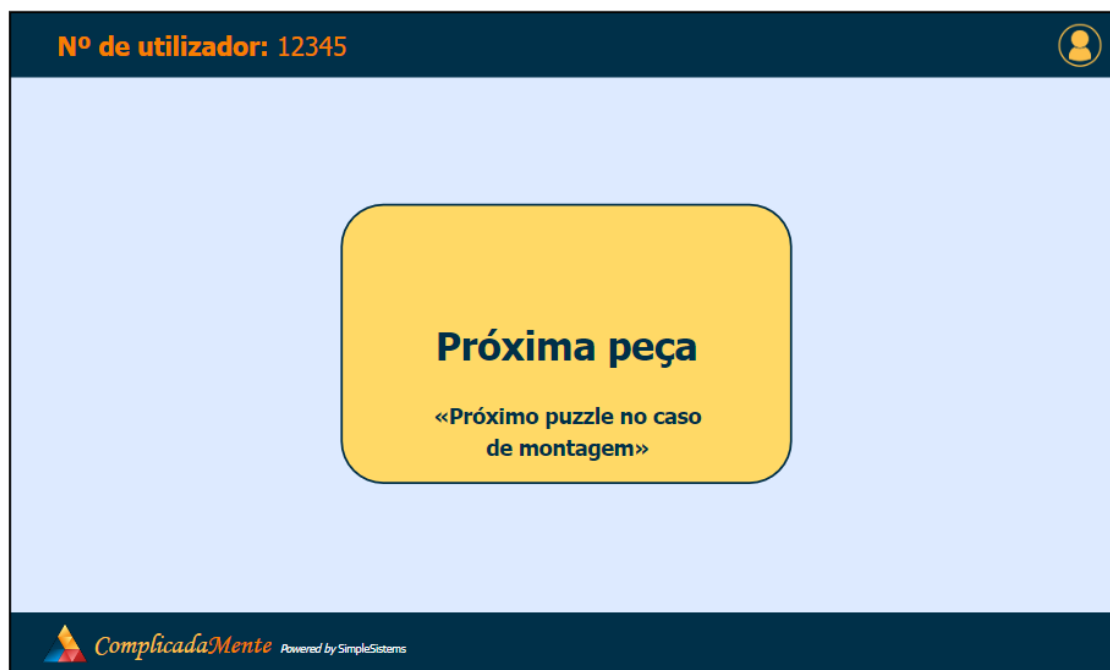


Figura 5.18: Mockup página principal do funcionário

5.1.19 Passos da produção/montagem

Após carregar no botão de próxima peça, o programa apresentará os passos da produção da peça com uma descrição do passo e entre 1 a 4 fotos a exemplificar o passo. O utilizador pode percorrer os passos através das setas laterais.

Esta interface é validada pelo use case de funcionário "Criar peça" e "Montar quebra-cabeça".

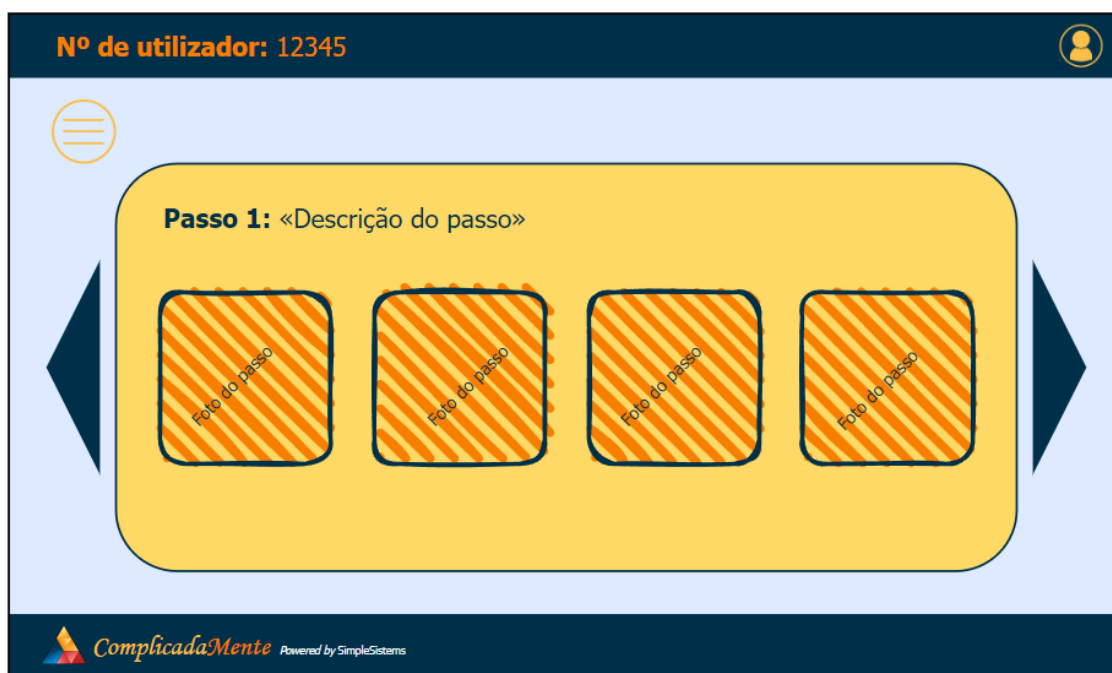


Figura 5.19: Mockup passos da produção/montagem

5.1.20 Informações adicionais da peça/puzzle

No mockup anterior, existe ainda um botão de menu que o funcionário pode carregar para ver as informações adicionais da peça (ou puzzle no caso da montagem).

Esta interface é validada pela ação no diagrama de atividades de funcionário em relação á peça "Visualizar/escolher peças".

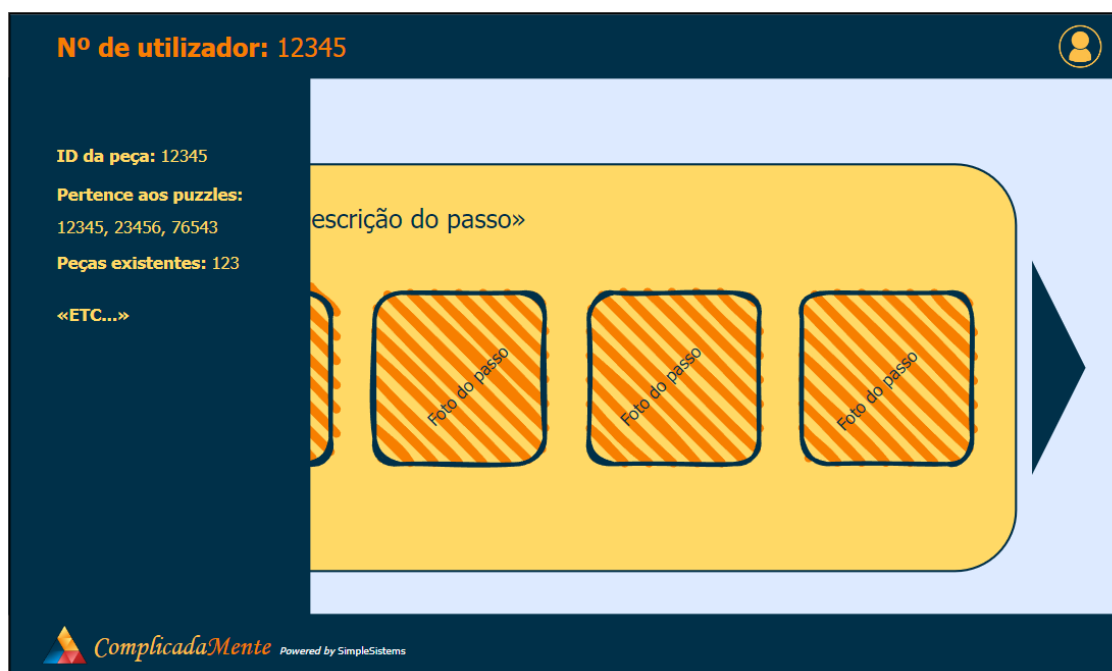


Figura 5.20: Mockup informações adicionais da peça

5.1.21 Finalização da peça/montagem

Quando o funcionário termina a peça/montagem tem de dizer se ficou em condições antes de prosseguir. Isto para evitar o programa registar a produção de uma peça ou de uma montagem que possam não estar em condições de serem enviadas.

Esta interface é validada pelo use case de funcionário "Criar peça".

The mockup shows a mobile application interface. At the top, a dark blue header bar contains the text "Nº de utilizador: 12345" in orange on the left and a yellow user icon on the right. Below the header, the main area has a light blue background. In the top-left corner of this area is a circular button with a left-pointing arrow. Centered below this is the question "Peça produzida sem defeitos?" in dark blue. Below the question are two large, rounded square buttons: a yellow one on the left labeled "SIM" in dark blue, and an orange one on the right labeled "NÃO" in dark blue. At the bottom, a dark blue footer bar contains the logo for "ComplicadaMente" (a stylized triangle) and the text "ComplicadaMente Powered by SimpleSystems" in white.

Figura 5.21: Mockup finalização da peça

5.1.22 Página principal do administrador

Se o login tiver sido feito como administrador, este obviamente terá mais funcionalidades do que o funcionário de produção. Por isso, a página principal do administrador tem várias opções.

Esta interface é validada pelos use cases de funcionário "Criar conta de funcionário" e "Gerir lucros".

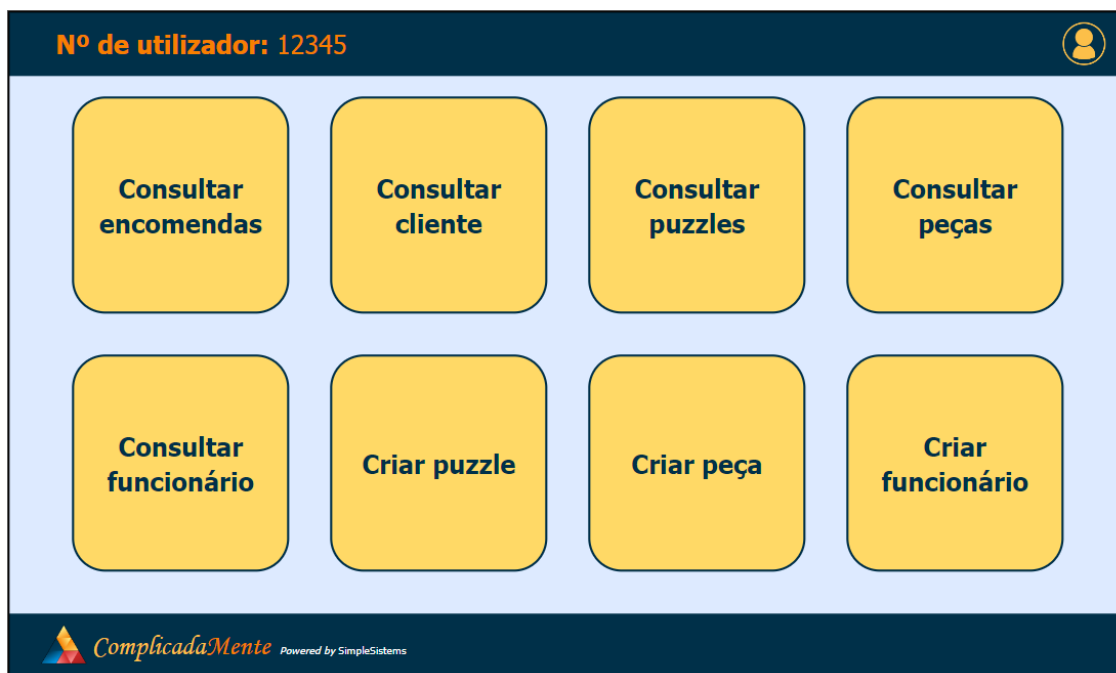


Figura 5.22: Mockup página principal do administrador

5.1.23 Scroll da página principal do administrador

Como tem várias funcionalidades, o administrador pode dar scroll da sua página principal.



Figura 5.23: Mockup scroll da página principal do administrador

5.1.24 Consultas e alterações de dados

Caso seja preciso fazer mudanças manualmente, seja no estado das encomendas, bem como nos dados dos puzzle ou peças ou até mesmo dos funcionários, o administrador pode consultar, seleccionar e em seguida alterar os dados.

Esta interface é validada pelo use case de funcionário "Alterar estado da encomenda".

The mockup shows a user interface for managing orders. At the top, a dark blue header displays the user ID "Nº de utilizador: 12345" and a user icon. Below this, a light blue section contains a back arrow and the order number "Nº da encomenda: 12567803". The main content area has a yellow background and contains the following information:

- Nome do Cliente:** Orlando Belo
- NIF:** 012345678
- Quer fatura com nº de contribuinte?** Sim ☒ Não ☐
- Telemóvel:** 912 345 678
- Email:** obelodiuminho@gmail.com
- Nº da encomenda:** 12567803

Below this, another yellow section is titled "Morada de envio". It contains the address "Rua: Av. da Liberdade" and "Nº da Porta: n21, 5º dt". A red dashed box highlights a button labeled "«Botão alterar estado da encomenda»" in the bottom right corner of this section.

At the bottom of the interface, a dark blue footer displays the logo for "ComplicadaMente" and the text "Powered by SimpleSistems".

Figura 5.24: Mockup consultas e alterações de dados

6 Implementação da Aplicação

6.1 Apresentação e descrição do processo de implementação realizado.

6.1.1 Preparação da Implementação

A fase de preparação da implementação é fundamental para garantir que todos os recursos e condições estejam devidamente alinhados antes do arranque do desenvolvimento do sistema. Durante este momento inicial, foram organizados os elementos necessários, como o ambiente de desenvolvimento, as tecnologias a utilizar, as ferramentas de apoio e o planeamento geral do projeto.

Tendo em conta que se trata de um projeto orientado a objetos, a SimpleSistemas recomendou a utilização da plataforma.NET, aliada ao framework Blazor. Após uma análise técnica por parte da equipa, concluiu-se que esta combinação era a mais adequada para responder aos requisitos definidos. Esta escolha oferece um ambiente de desenvolvimento robusto, uma experiência de utilizador eficiente e uma infraestrutura tecnológica com capacidade de evolução. A decisão foi formalizada e refletida na configuração do ambiente de trabalho.

Com a tecnologia definida, o próximo passo foi a preparação do ambiente de desenvolvimento, que incluiu a escolha do Visual Studio como IDE principal, devido às suas funcionalidades avançadas e compatibilidade com o tipo de aplicação a desenvolver. Para controlo de versões e colaboração entre os membros da equipa, foi configurado o GitHub, garantindo o registo das alterações realizadas ao longo do projeto.

Este planeamento assegurou uma transição organizada para a fase de desenvolvimento, promovendo o cumprimento dos prazos estabelecidos e garantindo que o projeto avançasse com uma base sólida e bem estruturada.

6.1.2 Processo de Desenvolvimento

Para garantir uma implementação eficaz e alinhada com os objetivos do projeto, a equipa da SimpleSistemas iniciou o trabalho com uma análise cuidada das etapas já realizadas. Com base nessa análise, foram definidas três áreas principais de atuação: criação da base de dados,

desenvolvimento do back-end e desenvolvimento do front-end.

Foi adotada uma abordagem iterativa e estruturada, permitindo validar cada componente ao longo do processo e minimizar eventuais riscos de falha.

A base de dados foi desenvolvida dentro do prazo estipulado, seguindo o modelo lógico previamente definido e utilizando SQL Server. Entre os dias 19 de novembro e 19 de dezembro, decorreu o desenvolvimento do back-end, implementado em C# com o apoio do Blazor, tecnologia escolhida também para suportar as interfaces da aplicação.

Paralelamente, teve lugar o desenvolvimento do front-end, dentro do mesmo período, permitindo testar a comunicação entre as diferentes camadas da aplicação através de interfaces funcionais simplificadas.

Por fim, foi reservado um período para testes e validação em colaboração com o cliente, com o objetivo de garantir que a solução final corresponde às expectativas, tanto ao nível funcional como da experiência de utilização.

6.2 Apresentação da aplicação e explicação dos serviços implementados

6.2.1 Criação da Base de Dados

Para concretizar a definição do sistema de dados estabelecida pela equipa, foi utilizada uma solução adaptada que combina o uso do SQLServer com Azure. Esta abordagem permitiu manter a compatibilidade com o modelo de dados previsto, garantindo simultaneamente um ambiente leve e funcional para o desenvolvimento.

A equipa procedeu à configuração local do ambiente com SQLServer e Azure Studio, tirando partido da sua integração simples e rápida, adequada ao sistema operativo em uso. Com base no esquema lógico previamente validado, foram desenvolvidos os scripts necessários para a criação da instância, definição das tabelas e inserção de dados de teste. Esta estrutura garantiu a consistência dos dados e permitiu validar o funcionamento da aplicação de forma eficaz durante as fases iniciais do desenvolvimento.

6.2.2 Desenvolvimento do back-end e do front-end

Para o desenvolvimento do back-end e do front-end, foi utilizada a linguagem C# com a plataforma .NET, recorrendo especificamente ao framework Blazor, como referido anteriormente. Esta escolha permitiu desenvolver a aplicação de forma integrada, tirando partido de uma base tecnológica unificada e moderna.

Foram implementados os serviços de utilizador e de carrinho, permitindo manter um estado persistente ao longo da utilização da aplicação. Desta forma, os dados do utilizador são mantidos durante a sessão, tal como o carrinho de compras, que permanece associado ao utilizador autenticado.

Os modelos da aplicação foram desenvolvidos com base nas entidades definidas e alinhados com a estrutura da base de dados. Durante esta fase, foi também introduzida uma alteração ao modelo da base de dados, com a inclusão de manuais associados a cada quebra-cabeças, reforçando a funcionalidade e a riqueza da informação disponível.

As páginas da aplicação foram desenvolvidas com base nos requisitos identificados e nos use cases definidos para os utilizadores. Este processo garantiu que a aplicação cumprisse com o seu propósito e com os compromissos assumidos pela equipa da SimpleSystems.

6.2.3 Login

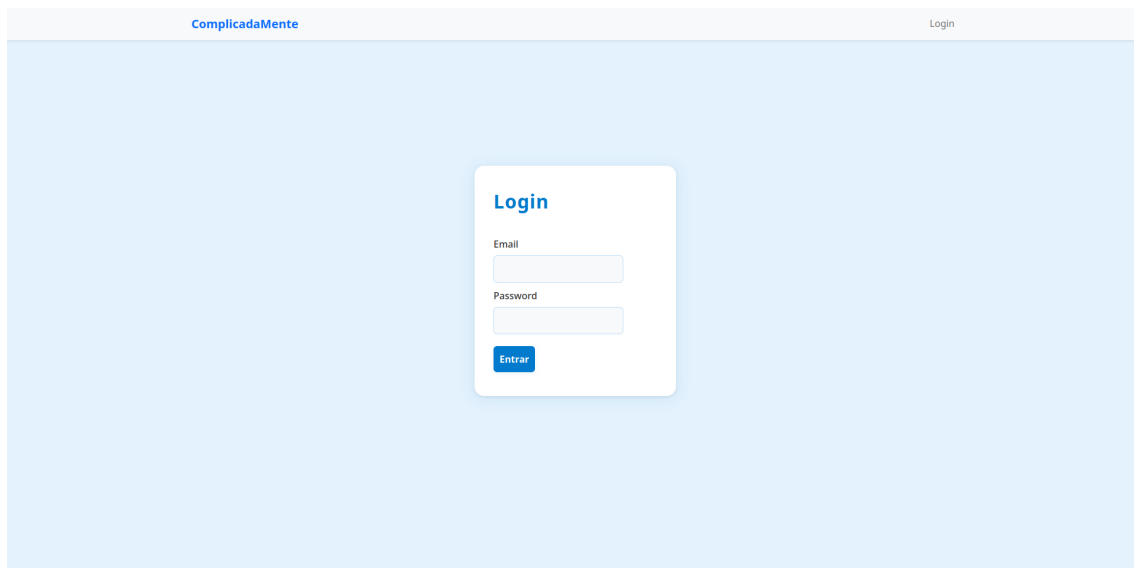


Figura 6.1: Implementação da Autenticação

Com o objetivo de garantir um controlo de acesso adequado às diferentes áreas da aplicação, a equipa da SimpleSystems priorizou a implementação de um sistema de autenticação baseado em funções. A lógica de autenticação foi desenvolvida manualmente e integra dois tipos distintos de utilizadores: Funcionários (admin) e Utilizadores (clientes).

O processo de login foi estruturado para validar as credenciais de ambos os perfis. Inicialmente, ao submeter o formulário de autenticação, a aplicação verifica se o e-mail inserido pertence a um Funcionário. Caso seja encontrado, é feita a verificação da password, usando BCrypt para garantir a segurança na validação dos dados. Se a autenticação for bem-sucedida, a aplica-

ção armazena os dados do utilizador autenticado através de um serviço de gestão de estado (UserState) mencionado anteriormente, que mantém as informações essenciais do utilizador ao longo da sessão, como o ID, nome, e-mail e função.

Se o e-mail não corresponder a nenhum Funcionário, o sistema prossegue com a mesma verificação no grupo de Utilizadores. O fluxo de autenticação mantém-se idêntico, adaptando apenas a função atribuída no estado persistente (neste caso, "client").

Dependendo da função do utilizador autenticado, o sistema redireciona automaticamente para a área correspondente: página de Produtos no caso dos Funcionários, ou a página de loja no caso dos Utilizadores.

Além disso, foram previstas mensagens de erro claras para situações como credenciais inválidas, ausência de password configurada ou e-mail não registado. Todo este processo foi encapsulado numa interface de login simples e funcional, garantindo uma boa experiência de utilização e preparando o sistema para validações e autorizações futuras com base nas funções atribuídas.

6.2.4 Loja — Visão Utilizador

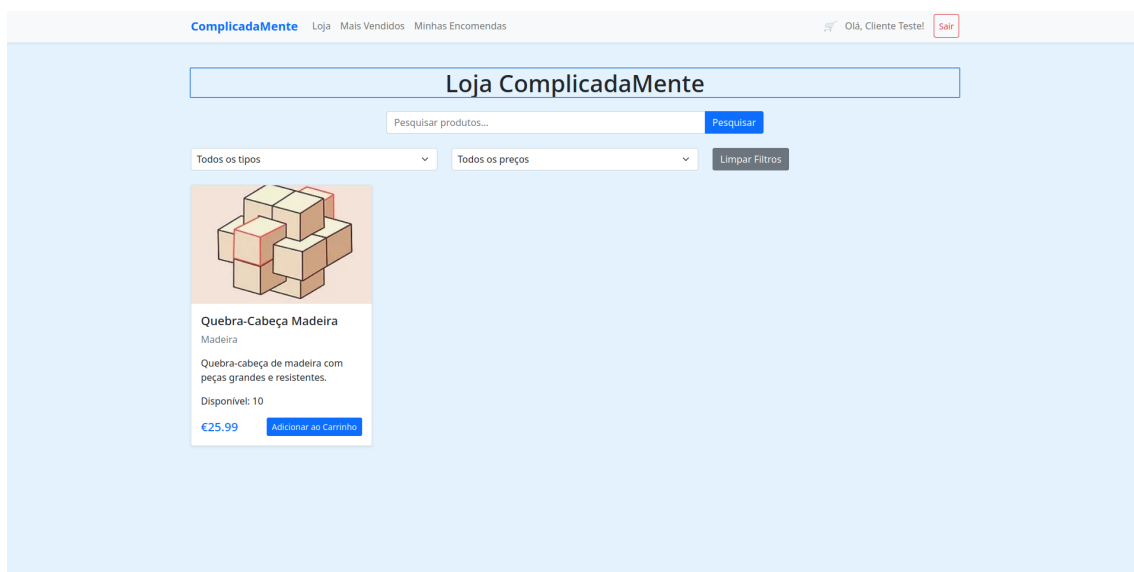


Figura 6.2: Implementação da Loja

A loja online da aplicação apresenta uma listagem dinâmica de produtos, composta por quebra-cabeças completos e peças individuais. A interface permite ao utilizador pesquisar, filtrar por tipo e filtrar por faixa de preço, tornando a navegação mais intuitiva e personalizada.

Todos os produtos são carregados a partir da base de dados, e os filtros são aplicados localmente, permitindo respostas rápidas sem recarregar a página. A aplicação garante também

uma gestão rigorosa de stock: antes de adicionar qualquer item ao carrinho, é verificada a quantidade disponível, tendo em conta tanto o stock atual como o número de unidades já reservadas em encomendas pendentes ou em montagem.

Caso o utilizador tente adicionar mais produtos do que o disponível, a loja exibe uma mensagem de erro clara, impedindo a adição e garantindo a integridade do processo de compra.

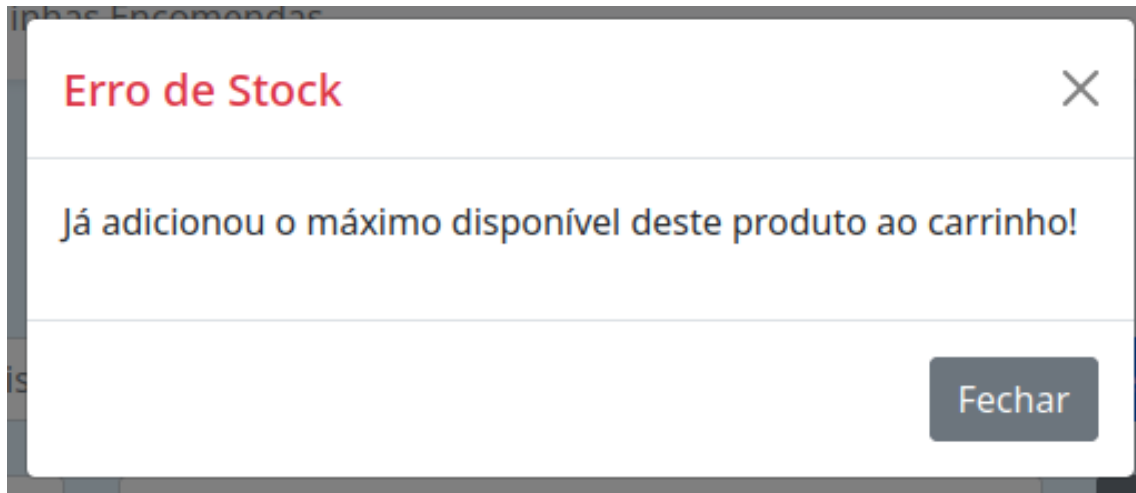


Figura 6.3: Erro de Stock na forma de Modal

6.2.5 Populares — Visão Utilizador



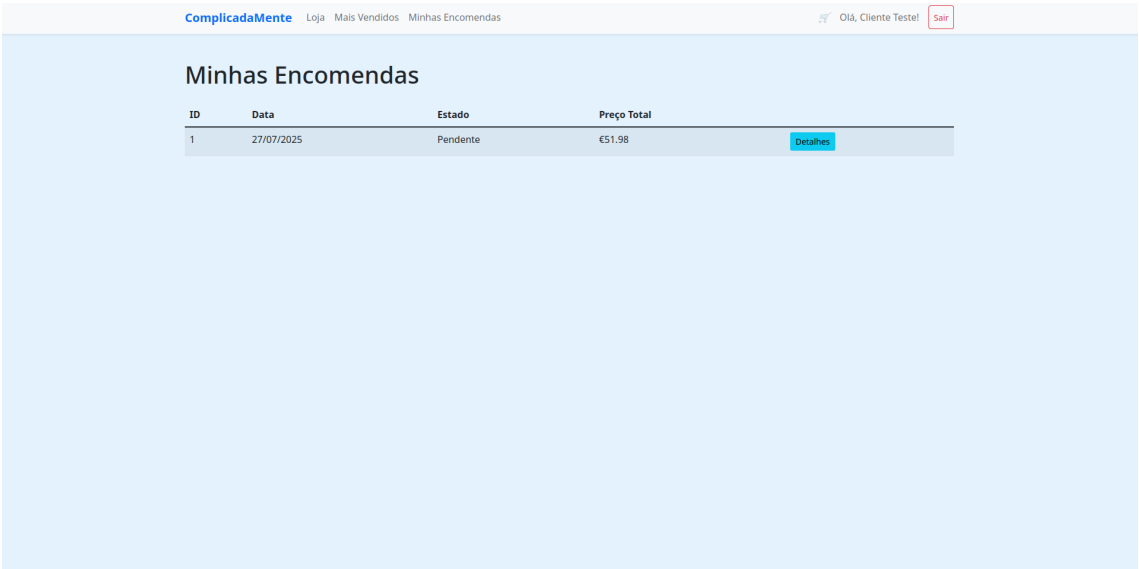
Figura 6.4: Implementação da Loja de Produtos Populares

A secção de produtos populares foi desenvolvida com o objetivo de destacar os artigos mais vendidos da loja. Para isso, a aplicação analisa os dados de vendas, agrupando e somando as quantidades encomendadas de cada quebra-cabeça e peça, e ordena os resultados por número de unidades vendidas.

Com base nesta análise, são apresentados até 12 produtos mais vendidos, misturando peças e quebra-cabeças. Caso não existam dados suficientes de vendas, o sistema mostra produtos aleatórios, garantindo sempre uma experiência de navegação atrativa.

Tal como na loja, é possível adicionar os produtos diretamente ao carrinho, com validações de stock disponível para evitar a adição de itens em excesso.

6.2.6 Encomendas do Utilizador — Visão Utilizador



The screenshot displays the 'Minhas Encomendas' (My Orders) section of the 'ComplicadaMente' application. The header includes the application name and navigation links for 'Loja', 'Mais Vendidos', and 'Minhas Encomendas'. A user greeting 'Olá, Cliente Teste!' and a 'Sair' (Logout) button are visible in the top right. The main content area features a table with the following data:

ID	Data	Estado	Preço Total	
1	27/07/2025	Pendente	€51.98	Detalhes

Figura 6.5: Implementação das Encomendas do Utilizador

A secção de encomendas permite aos utilizadores autenticados consultar o histórico das suas encomendas. Assim que o utilizador entra na página, são carregadas todas as encomendas associadas à sua conta, apresentadas por ordem decrescente de data.

Cada encomenda inclui o detalhe dos produtos adquiridos, sejam quebra-cabeças ou peças, com possibilidade de visualizar os detalhes de forma expandida. Esta funcionalidade garante ao utilizador um acesso claro e organizado ao seu registo de encomendas.



Figura 6.6: Detalhes das Encomendas do Utilizador

6.2.7 Carrinho de Compras — Visão Utilizador

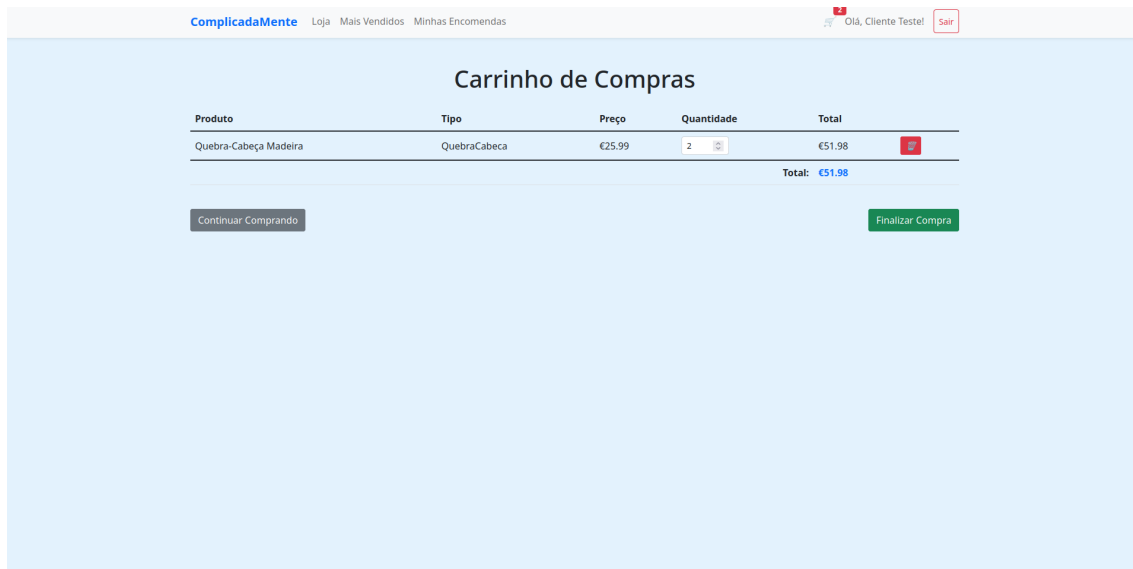


Figura 6.7: Implementação do Carrinho de Compras do Utilizador

O carrinho de compras permite aos utilizadores visualizar, atualizar ou remover os produtos que pretendem adquirir, sejam quebra-cabeças ou peças. É possível ajustar a quantidade de cada item ou eliminá-lo por completo.

Quando o utilizador está pronto para finalizar a compra, pode iniciar o processo de checkout através de um modal que confirma os dados. Se autenticado, o sistema regista a encomenda na base de dados, associa os produtos ao utilizador e limpa automaticamente o carrinho após a conclusão, garantindo uma experiência de compra simples e eficaz.

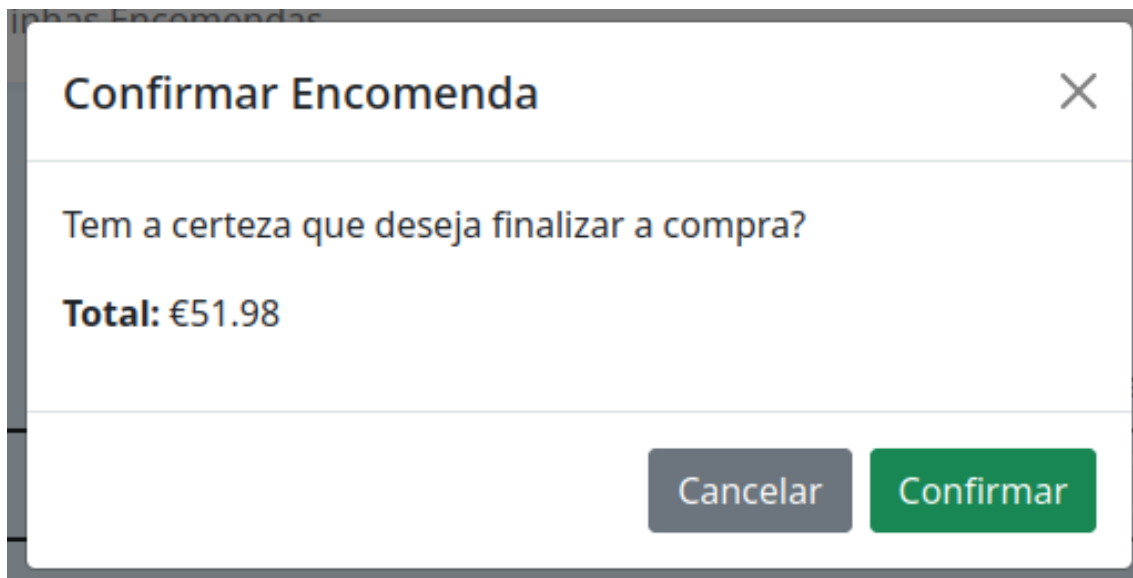


Figura 6.8: Confirmação de Encomenda

6.2.8 Produtos — Visão Administrador

ComplicadaMente

Produtos

Encomendas

Manuais

Olá, Admin Sistema!

Sair

Produtos Disponíveis

Puzzles (Quebra-Cabeças)

Nome	Tipo	Preço	Quantidade
Quebra-Cabeça Madeira	Madeira	€25.99	10

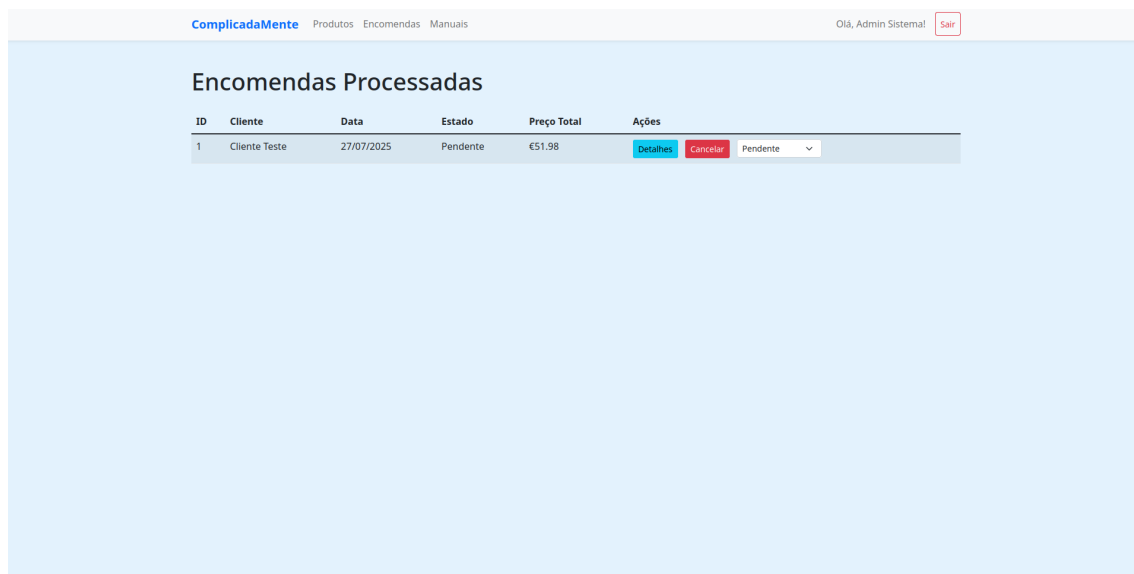
Peças

Nome	Tipo	Puzzle	Preço	Quantidade
------	------	--------	-------	------------

Figura 6.9: Implementação da página dos Produtos

A página de produtos do administrador apresenta uma listagem completa de todos os quebra-cabeças e peças existentes na base de dados. Esta página serve de painel de gestão para o administrador, permitindo-lhe aceder rapidamente aos produtos disponíveis — como, por exemplo, a quantidade existente em stock de cada produto, por forma a saber quais os produtos em falta.

6.2.9 Encomendas — Visão Administrador



ID	Cliente	Data	Estado	Preço Total	Ações
1	Cliente Teste	27/07/2025	Pendente	€51.98	Detalhes Cancelar Pendente ▾

Figura 6.10: Implementação da página de Encomendas

A página de encomendas do administrador permite visualizar e gerir todas as encomendas realizadas pelos utilizadores. Cada encomenda pode ser detalhada individualmente, incluindo os produtos encomendados e os dados do cliente (semelhante ao caso do cliente quando clica para ver os detalhes da sua encomenda).

O administrador pode alterar o estado da encomenda (por exemplo, para "Enviada" ou "Cancelada") e, no caso de envio, o sistema verifica automaticamente se existe stock suficiente. Se não houver, é apresentado um aviso ao administrador (semelhante ao que é apresentado ao cliente no caso de não existir stock para encomendar). Caso o stock seja suficiente, o inventário é atualizado automaticamente para refletir a saída dos produtos.

Esta página garante uma gestão eficaz e segura do processo de encomendas, contribuindo para o bom funcionamento da linha de montagem.

6.2.10 Manuais — Visão Administrador

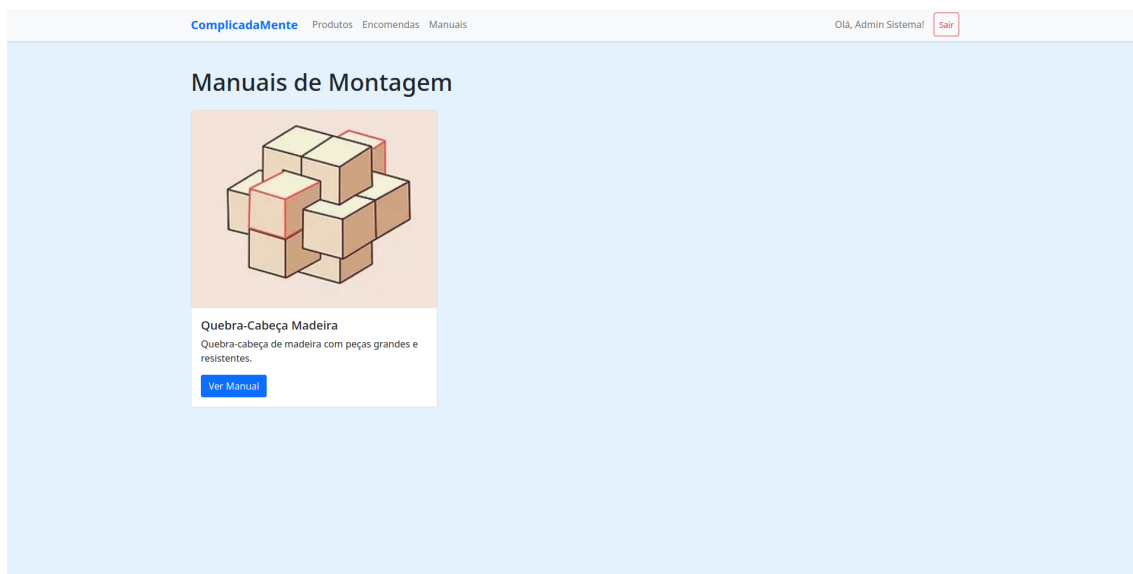


Figura 6.11: Implementação da página de Manuais

A página de manuais permite visualizar a lista de manuais associados aos respectivos quebra-cabeças existentes no sistema, proporcionando ao administrador uma visão clara sobre o processo de montagem de um determinado quebra-cabeças que esteja a ser encomendado.

6.2.11 Instruções dos Manuais — Visão Administrador

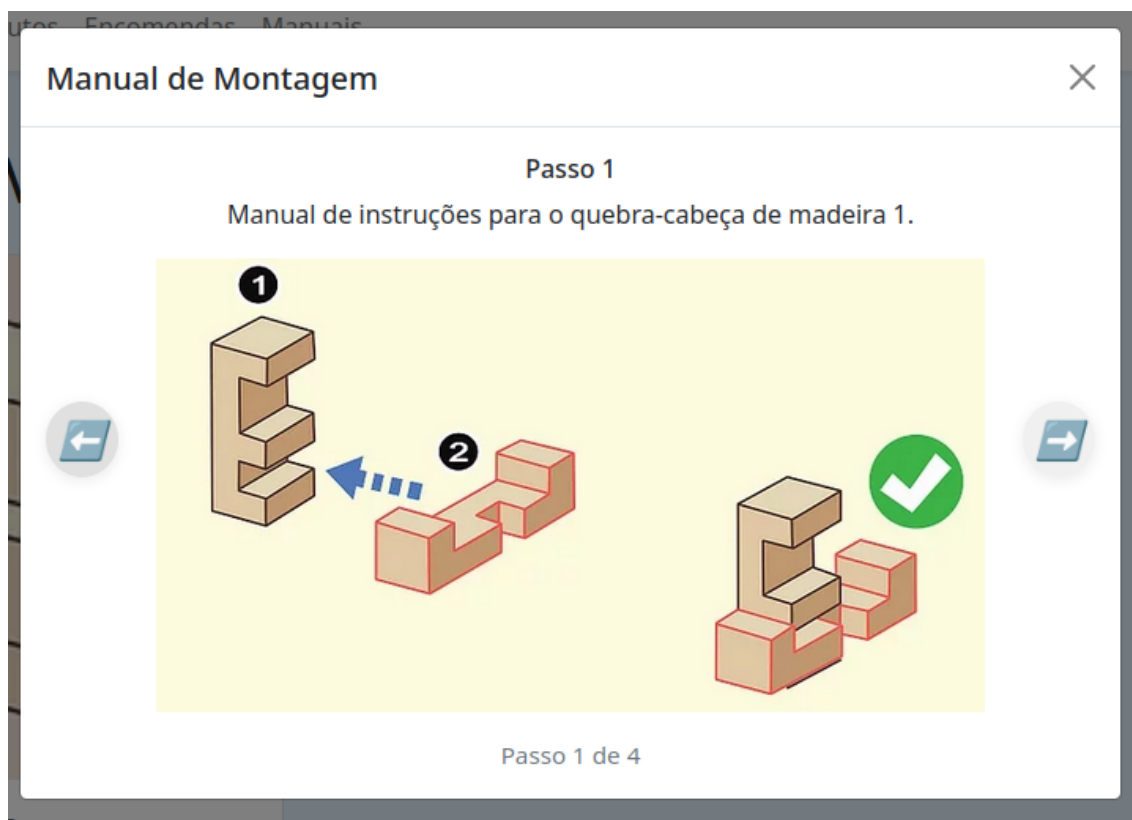


Figura 6.12: Implementação da página de Instruções dos respetivos Manuais

A página de manuais do administrador permite visualizar os manuais passo a passo associados a cada quebra-cabeças. O administrador pode selecionar um quebra-cabeças e consultar os seus respetivos passos organizados por ordem. A navegação entre os passos é feita através de botões para avançar ou retroceder, oferecendo uma visualização clara e sequencial das instruções.

Esta funcionalidade facilita o controlo e a validação dos conteúdos pedagógicos ou de montagem associados a cada produto.

6.3 Análise e avaliação da aplicação desenvolvida

Após o desenvolvimento da aplicação, foi realizada uma reunião de validação com os a equipa da ComplicadaMente. A aplicação foi considerada funcional, com uma base de dados sólida que responde bem às necessidades de gestão de utilizadores, produtos e encomendas.

As interfaces foram bem recebidas, destacando-se a separação clara por tipo de utilizador (administrador e cliente) e a facilidade de navegação.

Os serviços desenvolvidos, como a gestão de encomendas, autenticação e manuais interativos, demonstraram-se eficazes. Para garantir a escalabilidade futura da aplicação, foi sugerida a implementação de mecanismos como caching, balanceamento de carga e análise contínua de logs.

7 Conclusões e Trabalho Futuro

7.1 Conclusões

Nesta fase inicial do desenvolvimento do projeto, focámo-nos em definir uma ideia sólida para o sistema, os seus objetivos principais, funcionalidades essenciais e os recursos necessários para a sua viabilidade. Esta preparação permitiu estabelecer uma base clara e bem fundamentada para o desenvolvimento posterior.

Procedemos à especificação dos requisitos, identificando as necessidades dos utilizadores e os objetivos do sistema, garantindo o alinhamento da solução com as expectativas do projeto. A criação dos modelos e a análise dos use cases ajudaram a visualizar os diferentes cenários de utilização e a conceber uma interface intuitiva e funcional.

Foram, no entanto, identificados alguns pontos a melhorar, como a simplificação excessiva do modelo de domínio — por exemplo, o stock não foi tratado como uma entidade independente, o que dificultou a fase de implementação. Outro aspeto foi a ausência de um use case que permitisse ao administrador editar o cargo de um funcionário, impedindo a sua transição entre secções, o que será corrigido futuramente.

Na implementação, a aplicação demonstrou ser simples, funcional e bastante user-friendly, com uma utilização intuitiva tanto para clientes como para administradores. As funcionalidades essenciais estão operacionais, com especial cuidado na segurança e separação de permissões. No entanto, a lógica de construção de quebra-cabeças com base nas peças ainda não foi implementada, e a interface de administração pode beneficiar de mais opções de interação e uma melhoria visual.

7.2 Trabalho Futuro

Como orientações para o futuro, propõe-se:

1. Implementar a funcionalidade de montagem automática de quebra-cabeças com base nas peças existentes;
2. Aumentar a interatividade para administradores, com mais ferramentas de gestão;

3. Melhorar a interface gráfica, tornando-a mais apelativa e responsiva;
4. Realizar testes com utilizadores finais para otimizar a experiência.

Estes melhoramentos permitirão evoluir a aplicação para um produto mais completo, eficiente e alinhado com as necessidades reais dos utilizadores.