



**Universidade do Minho**  
Escola de Engenharia  
Licenciatura em Engenharia Informática

## **Unidade Curricular de Laboratórios de Informática IV**

Ano Letivo de 2022/2023

### **FAIRTRADE**

João Cunha nº96386  
Matilde Fernandes nº95319  
Afonso Amorim nº97569  
Francisco Lameirão nº97504  
Artur Nóbrega nº95414

21 de fevereiro de 2023

# **LI4**

Data de Receção	
Responsável	
Avaliação	
Observações	

**FAIRTRADE**

João Cunha nº96386  
Matilde Fernandes nº95319  
Afonso Amorim nº97569  
Francisco Lameirão nº97504  
Artur Nóbrega nº95414

21 de fevereiro de 2023

# Resumo

O projeto baseia-se na construção de uma aplicação que permita a transição do comércio das feiras populares para a sua comercialização online. A ideia surgiu após a pandemia. A COVID-19 teve um enorme impacto em aspetos sociais e económicas, como consequência, em grande parte, das medidas adotadas para controlar a dispersão do vírus.

A FairTrade, nome da aplicação, tem como propósito ajudar comércio locais que possam ter sido afetados pela modernização do mercado e, principalmente, pela pandemia, acolhendo diversas feiras de numerosos tipos. Pretendemos estabelecer parcerias com as câmaras para que o processo de arranjar e comunicar com comerciantes, que queiram participar nas feiras e adicionar os seus produtos, seja facilitado.

Os utilizadores da FairTrade podem decidir, entre as feiras que estejam a decorrer, quais querem "visitar" e posteriormente escolher os produtos que querem adquirir. Além disso podem comunicar com o comerciante para tirar qualquer dúvida ou, até mesmo, negociar o preço dos seus produtos em que tenham interesse.

A aplicação fornece uma experiência agradável aos seus utilizadores, devido à sua apresentação simples e atrativa.

A etapa inicial do projeto passa por uma definição e fundamentação do mesmo, o que nos ajuda a ter uma visão mais profunda e clara da aplicação. É necessário a criação de um bom plano para o desenvolvimento do projeto e para efetuar uma boa repartição de responsabilidades entre os elementos da equipa.

A partir daí, passaremos para análise e especificação dos requisitos operacionais e funcionais e, finalmente, a construção da aplicação.

**Área de Aplicação:** Desenvolvimento de uma Aplicação para Feiras Online

**Palavras-Chave:** FairTrade, Feiras, Comércio, Negócio, Aplicação Web, Visual Paradigm, Microsoft SQL, C#.

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Contextualização . . . . .	1
1.2	Fundamentação . . . . .	2
1.3	Motivação e Objetivos . . . . .	2
1.4	Viabilidade . . . . .	3
1.5	Recursos a utilizar . . . . .	3
1.6	Equipa de trabalho . . . . .	4
1.7	Plano de Execução do Trabalho . . . . .	5
1.8	Estrutura do Relatório . . . . .	6
<b>2</b>	<b>Levantamento e Análise de Requisitos</b>	<b>7</b>
2.1	Apresentação da estratégia e método . . . . .	7
2.2	Descrição geral dos requisitos levantados . . . . .	8
2.2.1	Requisitos funcionais . . . . .	8
<b>3</b>	<b>Especificação e Modelação do Software</b>	<b>15</b>
3.1	Modelação do domínio . . . . .	15
3.2	Modelo de use cases . . . . .	16
3.2.1	Diagramas . . . . .	16
3.2.2	Use cases . . . . .	19
<b>4</b>	<b>Conceção do Sistema de Dados</b>	<b>33</b>
4.1	Base de dados . . . . .	33
4.1.1	Modelo lógico . . . . .	33
<b>5</b>	<b>Interface do Sistema</b>	<b>38</b>
5.1	Estrutura Geral das Interfaces . . . . .	38
5.2	Caracterização das Interfaces . . . . .	41
<b>6</b>	<b>Implementação da Aplicação</b>	<b>42</b>
6.1	Processo de Implementação . . . . .	42
6.2	Análise e Avaliação da aplicação desenvolvida . . . . .	43
6.3	Arquitetura da aplicação e ferramentas utilizadas . . . . .	43
6.4	Serviços implementados e estrutura final da aplicação . . . . .	44
6.4.1	Registar utilizador . . . . .	44
6.4.2	Fazer "login" . . . . .	45
6.4.3	Página inicial depois de dar "login" . . . . .	46

6.4.4	Página da lista de feiras . . . . .	47
6.4.5	Página de uma feira . . . . .	48
6.4.6	Página de um produto . . . . .	49
6.4.7	Página da lista de Vendedores . . . . .	50
6.4.8	Perfil do vendedor visto pelo cliente . . . . .	51
<b>7</b>	<b>Conclusões e Trabalho Futuro</b>	<b>52</b>
7.1	Conclusão . . . . .	52
7.2	Trabalho Futuro . . . . .	52

# Lista de Figuras

1.1	Diagrama de Gantt - Etapa 1 . . . . .	5
1.2	Diagrama de Gantt - Etapa 2 . . . . .	5
1.3	Diagrama de Gantt - Etapa 3 (Final) . . . . .	6
3.1	Modelo do domínio . . . . .	15
3.2	Diagrama de use cases - geral . . . . .	16
3.3	Subdiagrama - Cliente efetua compra . . . . .	17
3.4	Subdiagrama - Gestão do cliente . . . . .	17
3.5	Subdiagrama - Gestão de contas de vendedores . . . . .	17
3.6	Subdiagrama - Gestão de produtos . . . . .	18
3.7	Subdiagrama - Pesquisa . . . . .	18
3.8	Subdiagrama - Gestão do vendedor . . . . .	18
4.1	Modelo lógico . . . . .	33
5.1	"login" . . . . .	38
5.2	Página Inicial . . . . .	39
5.3	Perfil do Cliente . . . . .	39
5.4	Perfil do Vendedor . . . . .	40
5.5	Feira . . . . .	40
5.6	Produto . . . . .	41
6.1	"Layout" para registrar uma conta no sistema . . . . .	44
6.2	"Layout" para fazer "login" . . . . .	45
6.3	"Layout" da página principal após o início de sessão . . . . .	46
6.4	"Layout" da página de lista de feiras . . . . .	47
6.5	"Layout" da página de uma feira . . . . .	48
6.6	"Layout" da página de um produto . . . . .	49
6.7	"Layout" da página de lista vendedores . . . . .	50
6.8	"Layout" do perfil de um vendedor visto por um cliente . . . . .	51

## **Lista de Tabelas**

# 1 Introdução

## 1.1 Contextualização

As Feiras são eventos realizados em espaços públicos onde as pessoas expõem todo o tipo de produtos para venda, seja roupa, mobília, dispositivos eletrónicos, entre outros.

Existem fontes que permitem afirmar que, em 500 a.C., já se realizava esta atividade no Médio Oriente. As referências a feiras na Antiguidade e na idade Média estão geralmente associadas a datas e festividades religiosas.

Em Portugal, o crescimento económico e demográfico dos séculos XII e XIII permitiu a criação de excedentes, que eram objeto de venda nos mercados e feiras. Com o crescimento populacional dos centros urbanos, o consumo aumentou. As feiras foram muito importantes no período medieval em Portugal e eram um espaço de encontro de produtores, consumidores e distribuidores de diversas regiões. A sua importância económica testemunhou a proteção dispensada às mesmas pelos monarcas, que concediam privilégios aos mercadores que nelas participassem.

Atualmente, as feiras continuam a ter alguma participação na economia do país, no entanto, devido à concorrência e modernização do comércio, o futuro destas está em risco.

O mercado das feiras tem como principal objetivo promover o comércio local e pequenos artesãos, dando oportunidade a todo o tipo de comerciantes de expor e vender os seus produtos, assim quantas mais pessoas forem às feiras, mais visibilidade os comerciantes vão ganhar, gerando mais oportunidades de negócio.

Uma das características mais importantes das feiras é a oportunidade que os clientes têm de negociar preços de artigos com o comerciante de maneira a que ambos façam um bom negócio e fiquem satisfeitos com o mesmo. A arte de regatear é um dos maiores trunfos deste tipo de comércio.

As feiras podem ser entendidas como um modelo específico de evento, mas cada feira possui as suas particularidades dependendo do objetivo da mesma.



## 1.2 Fundamentação

Como referido anteriormente, há uma grande incerteza no futuro das feiras. Cada vez mais, a maioria das pessoas não considera a opção de ir a uma feira para fazer compras. Existem inúmeras razões para explicar o decaimento de vendas nas feiras.

Devido à pandemia de Covid-19, a realização de feiras foi suspensa durante tempo indefinido. Como consequência, o mercado local sofreu ainda mais enquanto as restrições não eram levantadas. Isto traduziu-se no agravamento dos problemas que os feirantes já tinham mesmo antes da pandemia e do isolamento.

A feira tornou-se um conceito "antiquado" no quotidiano da sociedade contemporânea, tendo caído em especial desuso com a pandemia, dado que a opção online tornou-se cada vez mais presente nos hábitos dos consumidores.

Partindo do pressuposto de não existir, à data, um aplicativo que se traduzisse na solução deste problema, surge a proposta de apresentar ao mercado a FairTrade. Uma aplicação que permite a existência de feiras online.

## 1.3 Motivação e Objetivos

A FairTrade é uma aplicação que proporciona a venda de qualquer tipo de artigos, englobando inúmeros comércios, que torna possível, de uma forma intuitiva, aos comerciantes escolherem em que feira querem vender os seus artigos e, da mesma forma, os clientes também podem escolher que feira querem visitar consoante o que querem comprar.

Para nos aproximarmos daquilo que são os negócios de feiras, é possível o vendedor colocar artigos em modo leilão, no qual o cliente que der mais dinheiro compra o produto, cada vez que o preço oferecido por um cliente for excedido ele receberá uma notificação.

Devido a todas as razões que nos motivaram a pensar na criação e desenvolvimento desta aplicação, o nome "FairTrade" surgiu facilmente e pareceu ao grupo a melhor opção para o nosso projeto, por ser um nome apelativo, marcante e que representa perfeitamente os nossos objetivos, um comércio justo, honesto e que simule as feiras que já foram e podem vir a continuar a ser tão importantes para as pessoas e país.

Pretendemos apoiar os feirantes e tentar solucionar alguns dos problemas já mencionados.

Assim, os principais objetivos que pretendemos atingir com a criação desta aplicação são:

- Criar um sistema de compra e venda que consiga funcionar sem ser necessário contacto pessoal entre o vendedor e o cliente, evitando que a realização de feiras em contextos como, por exemplo, de pandemia tenha que ficar suspensa.

- Promover o comércio tradicional e local e contribuir para a sua expansão a nível nacional e internacional.
- Venda segura e com restrições de autenticidade para prevenir cópias/falsificações.
- Aumentar o público-alvo, já que é uma aplicação online, qualquer pessoa consegue facilmente aceder à mesma.

## 1.4 Viabilidade

Com o desenvolvimento contínuo da tecnologia e excluindo o contexto de pandemia, há um aumento das compras e vendas via online. Deste modo, o nosso projeto traz uma potencial solução de forma a que o contexto e significado de feira não se torne algo do passado.

Mesmo encontrando algumas aplicações que cumprissem alguns requisitos e funcionalidades procuradas ou que achássemos que fossem essenciais, decidimos tomar ação e partir para o desenvolvimento de uma aplicação nós mesmo. Sendo assim, o nosso sistema tem o foco nos interesses dos nossos utilizadores, não só disponibiliza uma grande diversidade de produtos para os nossos utilizadores poderem comprar mas também abre portas a novos vendedores que estão a entrar no mercado e querem um lugar onde podem promover os seus produtos, possuímos as ferramentas capazes de os guiar no caminho certo para isso.

Um ponto a destacar é que a nossa aplicação varia conforme os gostos de cada utilizador. A nossa aplicação, com base em produtos visitados anteriormente, vai guardar as preferências de cada utilizador e adaptar a frequência com que esse tipo de produtos aparece, para que consigam encontrar certos produtos com maior facilidade.

## 1.5 Recursos a utilizar

Para a construção da aplicação FairTrade, será necessário um grupo de 5 pessoas, com todas as competências para realizar a estruturação, desenvolvimento e implementação da mesma. Para isto, será imperativo todos os membros do grupo adquirirem conhecimento das várias áreas que a aplicação engloba, tanto na área da gestão e projeção do projeto como do próprio mercado de feiras.

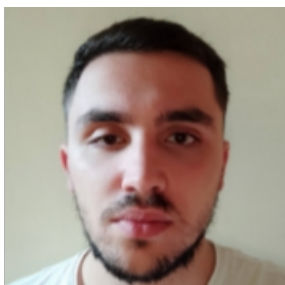
Para desenvolver a modelação do projeto em UML, será utilizado o programa Visual Paradigm.

Para criar e gerir a base de dados iremos utilizar Microsoft SQL, mais especificamente a versão Express.

Para a aplicação em si, será utilizado ASP.NET 7.0 para conseguirmos facilmente criar uma aplicação que envolva ligações entre páginas (em HTML) e código (em C#).

## 1.6 Equipa de trabalho

A equipa de trabalho que criou e desenvolveu a aplicação é constituída por 5 pessoas.



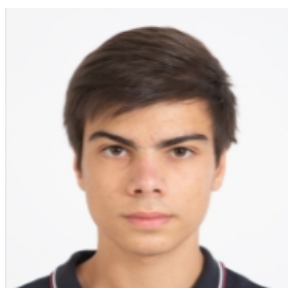
Francisco, A97504



Matilde, A95319



Artur, A95414



Afonso, A97569



João, A96386

## 1.7 Plano de Execução do Trabalho

No que diz respeito à planificação do desenvolvimento e execução do trabalho, podemos dizer que está dividida em 3 etapas: identificação e Caracterização geral da aplicação, em seguida, a Recolha e Análise dos requisitos da aplicação e, por último, Construção e Implementação da aplicação.

Estão representados de seguida, os diagramas de Gantt, que demonstram as tarefas e a distribuição feita das mesmas entre os elementos do grupo, referentes a cada etapa de execução do projeto.

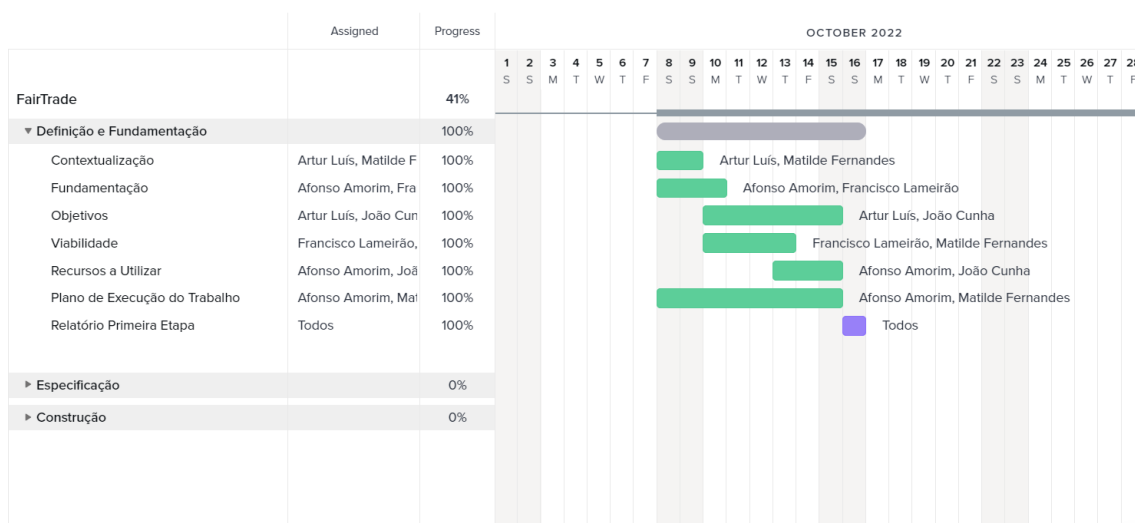


Figura 1.1: Diagrama de Gantt - Etapa 1

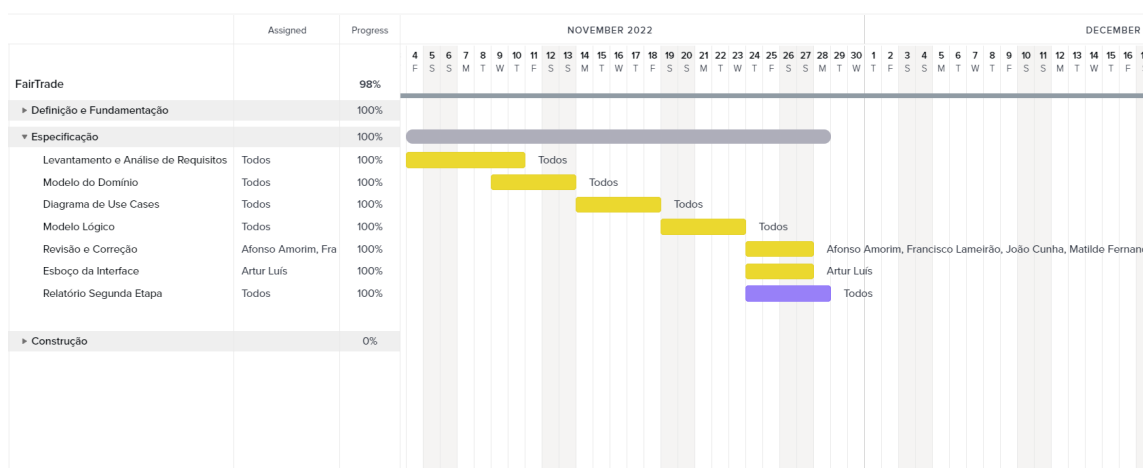


Figura 1.2: Diagrama de Gantt - Etapa 2

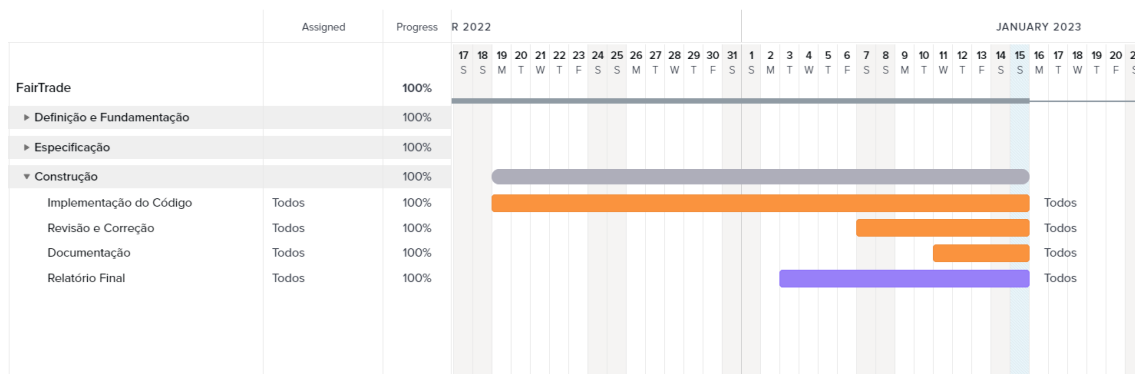


Figura 1.3: Diagrama de Gantt - Etapa 3 (Final)

## 1.8 Estrutura do Relatório

Após esta breve introdução o nosso relatório estará dividido pelos seguintes capítulos:

**Levantamento e Análise de Requisitos:** Neste primeiro capítulo vamos apresentar todos os requisitos funcionais e não funcionais que tivemos em consideração para o funcionamento correto da nossa aplicação e que fosse de acordo ao que nós pretendíamos.

**Especificação e Modelação do Software:** É possível encontrar, nesta parte do relatório, os modelos do domínio e de use cases e respetivos diagramas. Apresentamos modelos abstratos que traduzem as nossas ideias para as funcionalidades da aplicação e relação entre as entidades que dela fazem parte.

**Conceção do Sistema de Dados:** Aqui vamos fazer uma breve explicação no que constitui a nossa base dados, apresentando assim o nosso modelo lógico e explicamos também o porquê dessas tabelas existirem e o mesmo se aplica aos relacionamentos presentes.

**Interface do Sistema:** Nesta parte do relatório mostramos um bocado o que pretendemos alcançar com este projeto no que toca a uma perspetiva mais estética e apresentamos a estrutura da interface e uma descrição geral dos requisitos levantados.

**Implementação da Aplicação:** Aqui iremos abordar o processo que levou à implementação dos requisitos e das funcionalidades que nós queremos que a nossa aplicação tenha. Iremos falar da maneira de como implementamos o código e as ferramentas que usamos para o fazer.

**Conclusões e Trabalho Futuro:** Por fim, incluímos um pequeno comentário que resume a opinião do grupo sobre a execução do trabalho, expondo as nossas maiores dificuldades ao longo da mesma e o nosso grau de satisfação com o resultado.

## 2 Levantamento e Análise de Requisitos

### 2.1 Apresentação da estratégia e método

O objetivo do nosso projeto é então construir um website que simule aquilo que é uma feira, a nível virtual. Para isso, teríamos de saber ao certo como funciona uma feira em regime físico/presencial. Assim sendo, falamos com vários feirantes, uns mais experientes e uns mais jovens que, responsáveis pelo futuro desta tradição, necessitam de criatividade e ambição de modo a que esta tradição não se perca, independentemente das próximas gerações.

Com todas estes encontros, criamos um conjunto de entidades e funcionalidades para o funcionamento e a boa implementação do nosso projeto.

-Cliente

-Perfil Cliente

-Conta

-Perfil Funcionario

-Feira

-Produto

- "chatbox"

-Review

-Camara

-Informação

## **2.2 Descrição geral dos requisitos levantados**

### **2.2.1 Requisitos funcionais**

#### **CLIENTE**

##### **Criar uma conta:**

##### **Requisitos do utilizador**

O utilizador só pode usar o serviço estando registado no sistema.

##### **Requisitos do sistema**

1. O sistema pede nome de utilizador ("username"), "email", palavra-passe, método de pagamento e morada.
2. O sistema guarda os dados solicitados na base de dados.
3. O sistema não permite que seja feito um registo com um "username" já existente.

##### **Fazer "login":**

##### **Requisitos do utilizador**

O utilizador precisa de fazer "login" para utilizar o serviço.

##### **Requisitos do sistema**

1. O sistema pede o nome de utilizador e a palavra-passe.
2. O sistema tem de verificar se as credenciais estão registadas na base de dados, recusando as mesmas caso contrário.

##### **Editar conta:**

##### **Requisitos do utilizador**

O utilizador acede ao seu perfil para editar os dados desejados.

##### **Requisitos do sistema**

1. O nome de utilizador e o e-mail associado à conta não podem ser editados.

2. Todas as alterações são imediatamente guardadas na base de dados

#### **Pesquisar feira:**

##### **Requisitos do utilizador**

O utilizador pode pesquisar uma feira a partir do menu inicial.

##### **Requisitos do sistema**

1. O sistema dispõe uma possível feira ou lista de feiras segundo o nome indicado pelo utilizador.

#### **Entrar numa feira:**

##### **Requisitos do utilizador**

O utilizador pode entrar numa feira e aceder à lista de produtos em exposição na mesma.

##### **Requisitos do sistema**

1. O sistema apresenta a lista de produtos com os seus respectivos preços que estão a ser vendidos na feira seleccionada.
2. Seleccionando um dos produtos, o sistema deverá apresentar a página com todos os detalhes do produto seleccionado.

#### **Consulta de um produto:**

##### **Requisitos do utilizador**

1. O utilizador pode entrar na página disponibilizada para a venda e visualização específica de um produto.
2. O utilizador pode consultar o perfil do vendedor que está a vender o produto.

##### **Requisitos do sistema**

1. O sistema apresenta imagens do produto disponibilizadas pelo vendedor, assim como o preço e as reviews dadas anteriormente ao produto por outros utilizadores.



**Consulta do perfil de um vendedor:****Requisitos do utilizador**

O utilizador pode consultar o perfil de um vendedor específico.

**Requisitos do sistema**

1. O sistema deve apresentar os dados públicos do perfil do vendedor, uma lista com os produtos que tem disponíveis e a avaliação média dos produtos que vendeu.

**Colocar produto no carrinho:****Requisitos do utilizador**

O utilizador pode adicionar um produto ao carrinho.

**Requisitos do sistema**

1. O sistema deve registar a escolha do produto, adicionando-o ao carrinho do utilizador, somando, simultaneamente, o seu valor ao valor total do carrinho.

**Efetuar pagamento:****Requisitos do utilizador**

O utilizador pode escolher um método de pagamento que tenha guardado na sua conta.

**Requisitos do sistema**

1. O sistema efetua o pagamento de acordo com o método que o utilizador selecionou.
2. O sistema avisa o utilizador se o pagamento foi efetuado com sucesso.

**Fazer review de um produto:****Requisitos do utilizador**

O utilizador pode dar uma review ao produto na página disponibilizada para a venda do mesmo.

**Requisitos do sistema**

1. O sistema apresenta uma escala de 1 a 5 estrelas onde o utilizador vai escolher o "rating"

que quer dar e pode decidir deixar um comentário na sua avaliação.

2. O sistema anota a review dada pelo o utilizador e calcula a avaliação media com base nos "rating"s dados por todos os utilizadores.

#### **Ver histórico de compras:**

##### **Requisitos do utilizador**

O utilizador pode aceder ao seu histórico de compras.

##### **Requisitos do sistema**

1. O sistema disponibiliza uma página onde o utilizador pode ver todas as compras que já fez na aplicação.

## **VENDEDOR**

#### **Faz "login":**

##### **Requisitos do utilizador**

O vendedor só consegue utilizar o sistema se tiver logado no sistema.

##### **Requisitos do sistema**

1. O sistema pede o "username" e a palavra passe.
2. O sistema tem de verificar se as credenciais estão registadas na base de dados, recusando a autenticação caso contrário.

#### **Editar conta:**

##### **Requisitos do utilizador**

O vendedor pode aceder ao seu perfil e editar os dados.

##### **Requisitos do sistema**

1. O sistema não permite a alteração do "username" e do e-mail associado à conta.

2. O sistema guarda automaticamente todas as alterações efetuadas na base de dados.

#### **Vendedor lista produto:**

##### **Requisitos do utilizador**

O vendedor pode colocar um produto à venda.

##### **Requisitos do sistema**

1. O sistema pede o nome, preço, stock, categoria e descrição do produto.
2. O sistema guarda todas as informações do produto na base de dados e disponibiliza-o na feira.

#### **Notificação ao vendedor:**

##### **Requisitos do utilizador**

O Vendedor recebe uma notificação da solicitação de um produto.

##### **Requisitos do sistema**

1. Caso a compra de um produto tenha sido concluída, o vendedor é notificado pelo sistema.

## **FUNCIONÁRIO**

#### **Registo como funcionário:**

##### **Requisitos do utilizador**

O funcionário associado à empresa cria uma conta com permissões para gerir o sistema.

##### **Requisitos do sistema**

1. O sistema solicita o id do funcionário, o e-mail e a palavra-passe.
2. O sistema guarda os dados recebidos pelo funcionário na base de dados.

### **"login" como funcionário:**

#### **Requisitos do utilizador**

Para o funcionário poder exercer as suas funções na aplicação tem de ter o "login" efetuado.

#### **Requisitos do sistema**

1. O sistema solicita o id do funcionário e a palavra-passe.
2. O sistema tem de verificar se as credenciais estão registadas na base de dados, recusando a autenticação caso contrário.

### **Cria conta do vendedor:**

#### **Requisitos do utilizador**

O funcionário é responsável por receber as informações relativas ao vendedor e criar a conta do mesmo.

#### **Requisitos do sistema**

1. O sistema pede "username", "email", palavra-passe, método de pagamento e morada.
2. O sistema guarda os dados solicitados na base de dados.
3. O sistema não pode permitir que seja feito um registo com um "username" já em uso.

### **Apagar conta de um vendedor:**

#### **Requisitos do utilizador**

O funcionário pode eliminar a conta de qualquer vendedor.

#### **Requisitos do sistema**

1. O Sistema apresenta uma opção para eliminar a conta de um vendedor.

### **Gestão de uma feira:**

#### **Requisitos do utilizador**

O funcionário é responsável por gerir uma feira (abrir e encerrar feira).

### **Requisitos do sistema**

1. O Sistema apresenta uma opção para abrir uma nova feira e, após o decorrer da mesma, uma opção para encerrá-la.

## 3 Especificação e Modelação do Software

### 3.1 Modelação do domínio

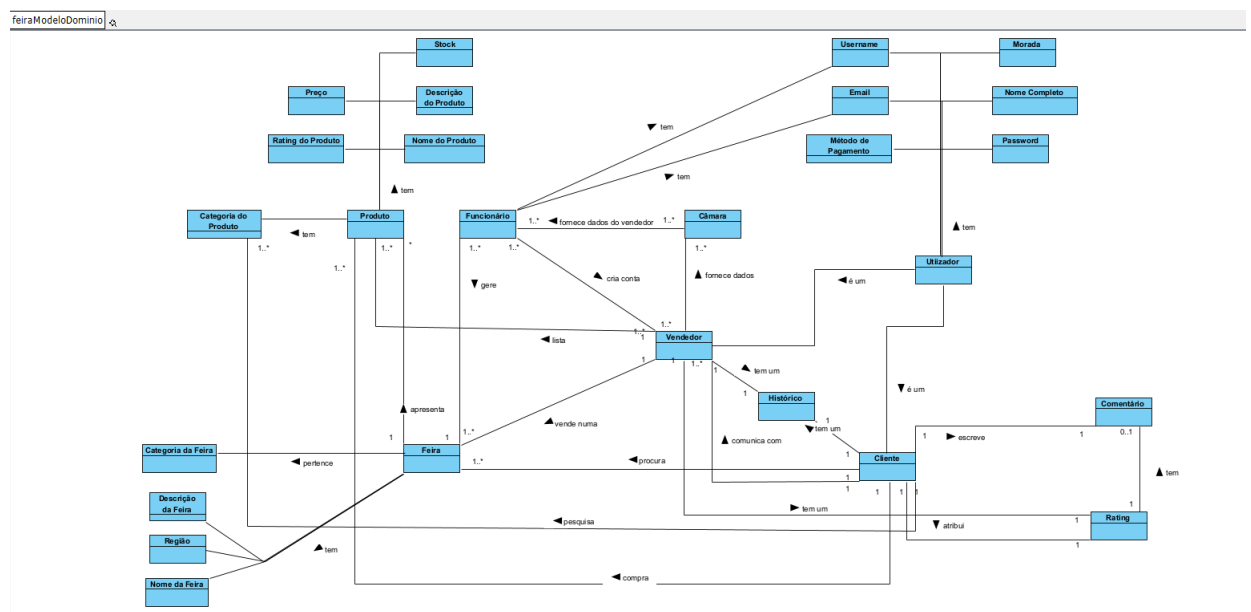


Figura 3.1: Modelo do domínio

Os utilizadores podem estar registados no sistema como vendedores ou clientes. Os funcionários são caracterizados por um "username" e um "email" e recebem dados sobre os vendedores e sobre as feiras. Esses dados são fornecidos pelas Câmaras, e com eles são criadas as contas dos vendedores e as feiras..

Um utilizador, quando registado no sistema como vendedor, possui várias entidades como: perfil, histórico e um "rating"; e também possui vários atributos como: "username", morada, "email", nome completo, método de pagamento e "password". Todos estes servem para caracterizar um vendedor. Neste sistema, o vendedor pode listar os seus produtos, que têm como atributos um stock, preço, nome do produto, descrição do produto, "rating" do produto e categoria do produto em várias feiras. Por sua vez, as feiras são caracterizadas tendo em conta a categoria da feira, nome da feira, descrição da feira e região.

Caso o utilizador esteja registado como cliente vai possuir como entidades um perfil e um histórico e como atributos que o caracterizam vai dispor dos mesmos que o vendedor. No sistema o cliente poderá pesquisar por um nome de uma feira, de um produto ou categoria de produto, consegue também seleccionar um produto e ver os atributos que caracterizam o mesmo, quando seleccionado um produto o cliente irá ter a opção de o comprar e de deixar um "rating" no produto e no vendedor que o vende.

## 3.2 Modelo de use cases

### 3.2.1 Diagramas

O nosso diagrama de use cases encontra-se dividido em 7 subdiagramas. O primeiro mais geral apresenta as relações dos 3 atores (cliente, vendedor e funcionário) com os diferentes grupos de use cases.

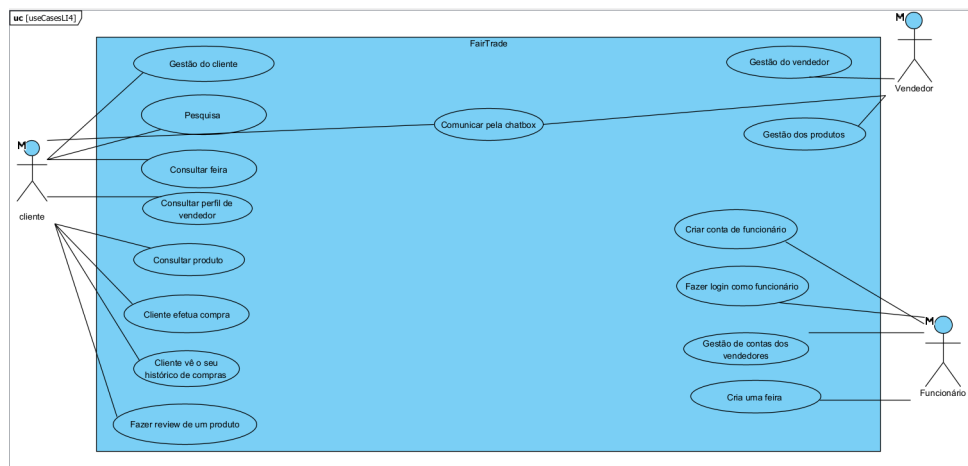


Figura 3.2: Diagrama de use cases - geral

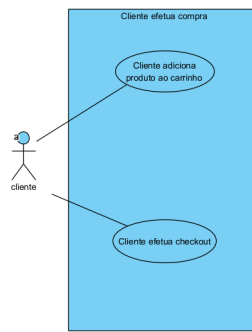


Figura 3.3: Subdiagrama - Cliente efetua compra

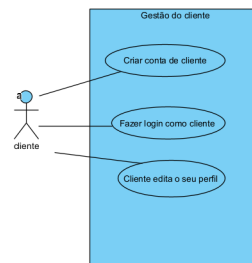


Figura 3.4: Subdiagrama - Gestão do cliente

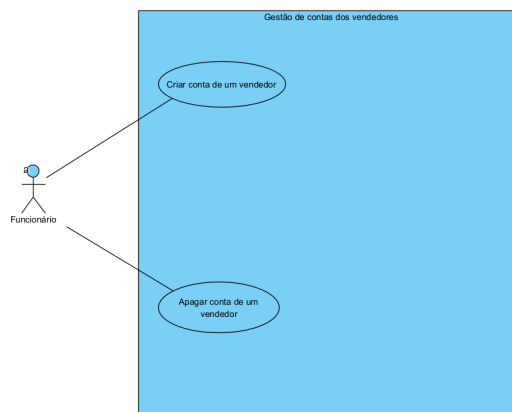


Figura 3.5: Subdiagrama - Gestão de contas de vendedores



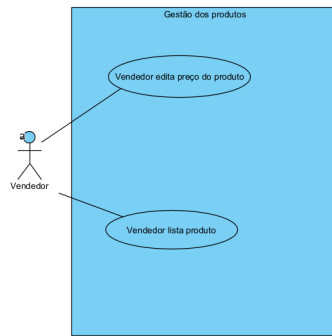


Figura 3.6: Subdiagrama - Gestão de produtos

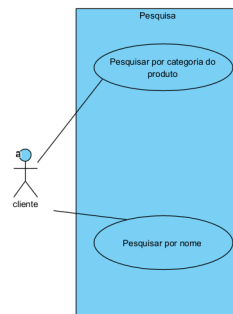


Figura 3.7: Subdiagrama - Pesquisa

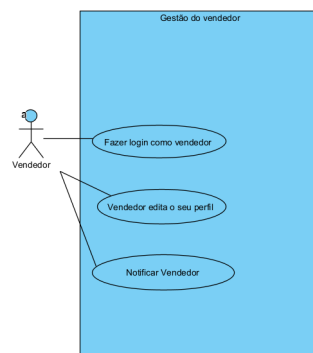


Figura 3.8: Subdiagrama - Gestão do vendedor

### 3.2.2 Use cases

#### Use Case: Criar conta de cliente

**Descrição:** Cria uma conta referente a um novo cliente

**Pré-condição:** True

**Pós-condição:** O cliente fica com a conta criada e registada no sistema

**Fluxo Normal:**

1. O sistema pede para o cliente introduzir o seu "email", o seu nome, um "username" e uma "password"
2. O cliente escreve o "email", o seu nome, o "username" e a "password"
3. O sistema verifica que o "username" e o "email" não estão em uso
4. O sistema pede ao cliente para introduzir um método de pagamento e uma morada
5. O cliente indica um método de pagamento e uma morada
6. O sistema verifica a validade da morada e do método de pagamento e regista os dados indicados pelo cliente na base de dados
7. O cliente fica registado no sistema

**Fluxo Alternativo 1:**["username" escolhido já existe] (passo 3)

- 3.1 O sistema avisa que o "username" escolhido já existe
- 3.2 Regressa ao passo 1

**Fluxo Alternativo 2:**["email" em uso] (passo 3)

- 3.1 O sistema avisa que o "email" indicado pelo cliente já está em uso
- 3.2 Regressa ao passo 1

**Fluxo Alternativo 3:**[Método de pagamento inválido] (passo 6)

- 6.1 O sistema avisa que o método de pagamento inserido pelo cliente é inválido
- 6.2 Regressa ao passo 4

**Fluxo Alternativo 4:** [Morada inválida] (passo 6)

- 6.1 O sistema avisa que a morada indicada pelo cliente é inválida

## 6.2 Regressa ao passo 4

### Use Case: Fazer "login" como cliente

**Descrição:** Entra no sistema como cliente

**Pré-condição:** O cliente tem de ter conta já criada

**Pós-condição:** O cliente fica ligado ao sistema e pode comprar produtos

**Fluxo Normal:**

1. O cliente fornece o seu "username" e "password"
2. Sistema verifica que o "username" e "password" foram corretamente inseridos
3. O cliente fica ligado ao sistema e pode comprar produtos

**Fluxo Alternativo 1:**[Nome ou "password" mal inseridos] (passo 2)

2.1 O sistema verifica que ou o "username" já existe e a "password" está mal inserida ou o contrário.

2.2 Sistema pede para reinserir o "username" e "password"

2.3 Regressa ao passo 1

### Use Case: Cliente edita o seu perfil

**Descrição:** O cliente faz alterações nos dados do seu perfil

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** As mudanças feitas pelo cliente ficam guardadas no sistema

**Fluxo Normal:**

1. O Sistema apresenta uma lista com os dados do cliente que podem ser alterados ("username", nome, morada, "password", e-mail, mudar método de pagamento)
2. O cliente escolhe que dados quer alterar e altera-os

3. As alterações feitas ficam guardadas no sistema

**Fluxo de exceção (1):**[Alterações inválidas] (passo 3)

3.1 O sistema avisa que alguma(s) das alterações feitas pelo cliente são inválidas e aborta a tentativa de edição

## **Use Case: Pesquisa por nome**

**Descrição:** O cliente pesquisa por algo usando um termo de pesquisa introduzido pelo cliente

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O que o cliente procura fica pronto para ser visualizado

**Fluxo Normal:**

1. O sistema apresenta a hipótese de procura uma feira, um produto ou um vendedor
2. O cliente escolhe procurar por uma feira
3. O sistema pede o nome da feira
4. O cliente escreve o nome da feira que quer procurar
5. O sistema devolve uma lista de feiras relevantes ao termo de pesquisa inserido

**Fluxo de exceção (1):** [Nenhum resultado com relevância] (passo 5)

5.1 O sistema informa o cliente que não encontrou resultados relevantes ao termo de pesquisa inserido

5.2 O sistema aborta a pesquisa

**Fluxo Alternativo 2:**[O cliente escolhe procurar por um produto] (passo 2)

- 2.1 O sistema pede o nome do produto
- 2.2 O cliente escreve o nome do produto que quer procurar
- 2.3 O sistema devolve uma lista de produtos relevantes ao termo de pesquisa inserido

**Fluxo Alternativo 3:**[O cliente escolhe procurar por um vendedor] (passo 2)

- 2.1 O sistema pede o nome do vendedor

2.2 O cliente escreve o nome do vendedor que quer procurar

2.3 O sistema devolve uma lista de vendedores relevantes ao termo de pesquisa inserido

### **Use Case: Pesquisar por categoria**

**Descrição:** O cliente pesquisa por algo selecionando a categoria que está interessado

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O que o cliente procura fica pronto para ser visualizado

#### **Fluxo Normal:**

1. O sistema apresenta a hipótese de procura uma feira ou um produto
2. O cliente escolhe procurar por uma feira
3. O sistema apresenta a hipótese de escolher entre regiões e categorias.
4. O cliente escolhe procurar por categoria
5. O sistema apresenta as categorias disponíveis para feiras
6. O cliente escolhe uma categoria
7. O sistema devolve uma lista de feiras associadas à categoria escolhida pelo cliente

#### **Fluxo de exceção (1):** [Nenhum resultado com relevância] (passo 7)

7.1 O sistema informa o cliente que não encontrou nenhum resultado com a seleção feita pelo cliente

7.2 O sistema aborta a pesquisa

#### **Fluxo Alternativo 2:**[ Procurar por um produto] (passo 2)

- 2.1 O cliente escolhe procurar por um produto
- 2.2 O sistema apresenta as categorias disponíveis para um produto
- 2.3 O cliente escolhe a categoria que pretende procurar
- 2.4 O sistema devolve uma lista de produtos associados à categoria escolhida

#### **Fluxo Alternativo 3:**[O cliente escolhe procurar uma feira por região] (passo 4)

- 4.1 O cliente escolhe procurar por região
- 4.2 O sistema apresenta as regiões disponíveis para procura
- 4.3 O cliente escolhe a região que pretende procurar
- 4.4 O sistema devolve uma lista de feiras associadas à região escolhida

### **Use Case: Consultar feira**

**Descrição:** O cliente seleciona a feira que quer visitar

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O cliente entra na feira e consegue ver a lista de produtos da mesma

**Fluxo Normal:**

1. O sistema apresenta uma lista de feiras
2. O cliente seleciona uma feira
3. O sistema abre uma página feira e apresenta a lista de produtos da mesma

### **Use Case: Consultar produto**

**Descrição:** O cliente acede à página de um produto específico, onde pode ver os detalhes do produto, reviews feitas por outros clientes e adicioná-lo ao carrinho de compra

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O cliente consegue ver detalhes e reviews de um produto e consegue adicioná-lo ao seu carrinho de compras

**Fluxo Normal:**

1. O sistema apresenta uma lista de produtos
2. O cliente seleciona um produto
3. O sistema abre uma página onde vão ser disponibilizados todas as informações sobre o produto selecionado assim como o "rating" do mesmo, a opção de o colocar no carrinho e

a de consultar o perfil do vendedor

### **Use Case: Consultar perfil de vendedor**

**Descrição:** O cliente consegue ver a informação sobre o vendedor e todos os produtos que ele vende

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O cliente vê a informação sobre o vendedor desejado e todos os produtos que ele vende

#### **Fluxo Normal:**

1. O sistema apresenta uma lista de vendedores
2. O cliente seleciona um vendedor
3. O sistema abre uma página onde vão ser disponibilizados todas as informações sobre o vendedor selecionado assim como os produtos que o mesmo vende

### **Use Case: Cliente adiciona produto ao carrinho**

**Descrição:** O cliente usa o botão "Adicionar ao carrinho" de modo acrescentar os produtos ao seu carrinho de compras para, em seguida, conseguir fazer a encomenda dos mesmos.

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O cliente tem o produto no carrinho

#### **Fluxo Normal:**

1. O cliente indica a quantidade do produto que pretende adicionar ao carrinho
2. O sistema adiciona o produto com a quantidade indicada ao carrinho
3. O sistema adiciona o preço subtotal do produto adicionado ao preço total do carrinho

## **Use Case: Cliente efetua checkout**

**Descrição:** O cliente compra todos os produtos que tem no seu carrinho de compras

**Pré-condição:** O cliente já fez "login" no sistema e tem produtos no carrinho

**Pós-condição:** O cliente efetua o pagamento dos produtos e estes são enviados para a sua morada

### **Fluxo Normal:**

1. O cliente indica que quer finalizar a compra
2. O sistema calcula a soma do total dos produtos e as taxas de entrega (calculadas a partir da morada do cliente) e apresenta-a ao cliente.
3. O cliente confirma a compra final
4. O sistema processa o pagamento através do método de pagamento guardado na conta do cliente
5. O sistema notifica o cliente que a sua encomenda foi realizada com sucesso e guarda a transação na base de dados

### **Fluxo de exceção (1): [Erro no processo de pagamento] (passo 4)**

- 4.1 O sistema é notificado que o processo de pagamento não teve sucesso
- 4.2 O sistema notifica o cliente que a sua encomenda não foi feita e aborta o processo de checkout

## **Use Case: Fazer review de um produto**

**Descrição:** O cliente deixa o seu "rating" e comentário a um produto

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O "rating" e o comentário ficam registados no sistema

### **Fluxo Normal:**

1. O sistema apresenta uma escala de 1 a 5 para o "rating"
2. O cliente escolhe o "rating" que quer dar



3. O sistema apresenta uma caixa de texto para o comentário
4. O cliente escreve um comentário sobre o produto
5. O "rating" e o comentário ficam registados no sistema

**Fluxo de exceção (1):**[O cliente quer deixar comentário] (passo 4)

- 4.1 Apenas o "rating" fica registado no sistema

### **Use Case: Cliente vê o seu histórico de compras**

**Descrição:** O cliente consegue ver todos os produtos que comprou na aplicação

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O cliente vê o seu histórico de compras

**Fluxo Normal:**

1. O sistema apresenta a opção de ver o histórico
2. O cliente seleciona a opção de ver o histórico
3. O sistema abre uma página onde o cliente pode ver o seu histórico

### **Use Case: Cliente efetua checkout**

**Descrição:** O cliente compra todos os produtos que tem no seu carrinho de compras

**Pré-condição:** O cliente já fez "login" no sistema e tem produtos no carrinho

**Pós-condição:** O cliente efetua o pagamento dos produtos e estes são enviados para a sua morada

**Fluxo Normal:**

1. O cliente indica que quer finalizar a compra
2. O sistema calcula a soma do total dos produtos e as taxas de entrega (calculadas a partir da morada do cliente) e apresenta-a ao cliente.

3. O cliente confirma a compra final
4. O sistema processa o pagamento através do método de pagamento guardado na conta do cliente
5. O sistema notifica o cliente que a sua encomenda foi realizada com sucesso e guarda a transação na base de dados

**Fluxo de exceção (1):**[Erro no processo de pagamento] (passo 4)

- 4.1 O sistema é notificado que o processo de pagamento não teve sucesso
- 4.2 O sistema notifica o cliente que a sua encomenda não foi feita e aborta o processo de checkout

### **Use Case: Fazer review de um produto**

**Descrição:** O cliente deixa o seu "rating" e comentário a um produto

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O "rating" e o comentário ficam registados no sistema

**Fluxo Normal:**

1. O sistema apresenta uma escala de 1 a 5 para o "rating"
2. O cliente escolhe o "rating" que quer dar
3. O sistema apresenta uma caixa de texto para o comentário
4. O cliente escreve um comentário sobre o produto
5. O "rating" e o comentário ficam registados no sistema

**Fluxo de exceção (1):**[O cliente quer deixar comentário] (passo 4)

- 4.1 Apenas o "rating" fica registado no sistema

### **Use Case: Cliente vê o seu histórico de compras**

**Descrição:** O cliente consegue ver todos os produtos que comprou na aplicação

**Pré-condição:** O cliente já fez "login" no sistema

**Pós-condição:** O cliente vê o seu histórico de compras

**Fluxo Normal:**

1. O sistema apresenta a opção de ver o histórico
2. O cliente seleciona a opção de ver o histórico
3. O sistema abre uma página onde o cliente pode ver o seu histórico

### **Use Case: Fazer "login" como vendedor**

**Descrição:** Entra no sistema como vendedor

**Pré-condição:** True

**Pós-condição:** O vendedor fica ligado ao sistema e pode vender produtos

**Fluxo Normal:**

1. O vendedor fornece o seu "username" e "password"
2. Sistema verifica que o "username" e "password" foram corretamente inseridos
3. O vendedor fica ligado ao sistema e pode vender produtos

**Fluxo alternativo 1:**[Nome ou "password" mal inseridos] (passo 2)

2.1 O sistema verifica que ou o "username" já existe e a "password" está mal inserida ou o contrário.

2.2 Sistema pede para reinserir o "username" e "password"

2.3 Regressa ao passo 1

### **Use Case: Vendedor edita o seu perfil**

**Descrição:** O vendedor faz alterações nos dados do seu perfil

**Pré-condição:** O vendedor já fez "login" no sistema

**Pós-condição:** As mudanças feitas pelo vendedor ficam guardadas no sistema

**Fluxo Normal:**

1. O sistema apresenta uma lista com os dados do vendedor que podem ser alterados
2. O vendedor escolhe que dados quer alterar e altera-os
3. As alterações feitas ficam guardadas no sistema

**Fluxo de exceção (1):**[Alterações inválidas] (passo 3)

3.1 O sistema avisa que alguma(s) das alterações feitas pelo vendedor são inválidas e aborta a tentativa de edição

**Use Case: Vendedor lista produto**

**Descrição:** O vendedor coloca um produto que pretende vender na feira em que está inscrito

**Pré-condição:** O vendedor fez "login" no sistema

**Pós-condição:** O produto é colocado em venda numa feira

**Fluxo Normal:**

1. O vendedor indica que quer colocar um novo produto à venda
2. O sistema pede o nome, preço, stock, categoria e descrição do produto
3. O vendedor indica o nome, preço, stock, categoria e descrição do produto
4. O sistema guarda o produto na base de dados e disponibiliza-o na feira

**Use Case: Notificar Vendedor**

**Descrição:** O Vendedor recebe uma notificação da solicitação de um produto.

**Pré-condição:** O vendedor já fez "login" no sistema

**Pós-condição:** O vendedor recebe uma notificação

**Fluxo Normal:**

1. Um cliente comprou um produto
2. Fica guardado no sistema que o produto foi vendido
3. O sistema envia uma notificação ao vendedor de que um produto foi vendido
4. O vendedor recebe a notificação

**Use Case: Criar conta de funcionário**

**Descrição:** Cria uma conta funcionário com os respetivos privilégios

**Pré-condição:** Ser um empregado da FairTrade

**Pós-condição:** O funcionário fica registado no sistema

**Fluxo Normal:**

1. O sistema pede o "email", "username" e a "password"
2. O funcionário dá o "email", o "username" e a "password"
3. O sistema verifica que o "username" não está já a ser utilizado
4. O funcionário fica registado no sistema

**Fluxo alternativo 1:**["username" escolhido já existe] (passo 3)

- 3.1 O sistema avisa que o "username" escolhido já existe
- 3.2 Regressa ao passo 1

**Use Case: Fazer "login" como funcionário**

**Descrição:** Entra no sistema como funcionário

**Pré-condição:** O funcionário tem conta já criada

**Pós-condição:** O funcionário fica ligado ao sistema

**Fluxo Normal:**

1. O cliente fornece o seu "username" e "password"
2. Sistema verifica que o "username" e "password" foram corretamente inseridos
3. O funcionário fica ligado ao sistema

**Fluxo alternativo 1:**[Nome ou "password" mal inseridos] (passo 2)

2.1 O sistema verifica que ou o "username" já existe e a "password" está mal inserida ou o contrário.

2.2 Sistema pede para reinserir o "username" e "password"

2.3 Regressa ao passo 1

**Use Case: Criar conta de vendedor**

**Descrição:** Um funcionário cria a conta de um vendedor

**Pré-condição:** O funcionário já fez "login" no sistema

**Pós-condição:** O vendedor fica registado no sistema

**Fluxo Normal:**

1. O sistema apresenta a hipótese de criar um novo vendedor
2. O funcionário seleciona a hipótese de criar um novo vendedor
3. O sistema pede o "email", "username" e a "password"
4. O funcionário dá o "email", o "username" e a "password"
5. O sistema verifica que o "username" não está já a ser utilizado
6. O funcionário fica registado no sistema

**Fluxo alternativo 1:**["username" escolhido já existe] (passo 5)

5.1 O sistema avisa que o "username" escolhido já existe

5.2 Regressa ao passo 4

### **Use Case: Apagar a conta de um vendedor**

**Descrição:** Um funcionário apaga a conta de um vendedor

**Pré-condição:** O funcionário já fez "login" no sistema

**Pós-condição:** A conta do vendedor deixa de existir no sistema

**Fluxo Normal:**

1. O sistema apresenta a hipótese de apagar a conta de um vendedor
2. O funcionário seleciona a hipótese de apagar a conta de um vendedor
3. A conta do vendedor é apagada do sistema

### **Use Case: Criar uma feira**

**Descrição:** O Funcionário cria uma feira no sistema

**Pré-condição:** O funcionário já fez "login" no sistema

**Pós-condição:** A feira criada fica registada no sistema

**Fluxo Normal:**

1. O sistema apresenta a hipótese de criar uma feira
2. O funcionário seleciona a hipótese de criar uma feira
3. O sistema pede o nome, a data de início, a duração, a região, a descrição e a categoria da feira
4. O funcionário escreve o nome, a data de início, a duração, a região, a descrição e a categoria da feira
5. A feira fica registada no sistema

## 4 Conceção do Sistema de Dados

### 4.1 Base de dados

Para o desenvolvimento da Base de Dados do nosso projeto, eis o que foi pensado enquanto grupo no que toca ao desenvolvimento da Modelação Lógica da Base de dados.

#### 4.1.1 Modelo lógico

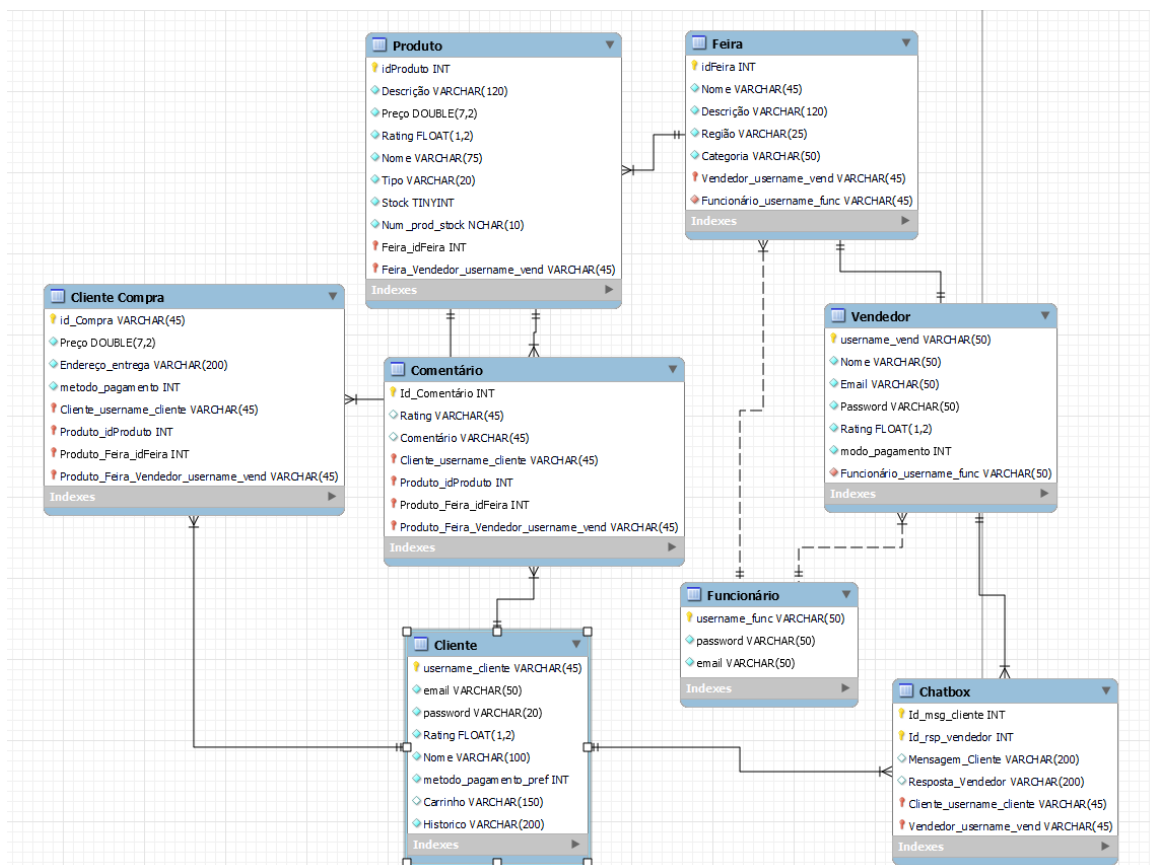


Figura 4.1: Modelo lógico



## Descrição detalhada dos vários elementos de dados e seus relacionamentos

Inicialmente iremos explicar o nosso raciocínio em relação à escolha de entidades e o que nos levou à criação do nosso modelo lógico :

Começemos por entender como foi o nosso raciocínio quando pensamos em que entidades nos focar e o porque de existirem as tabelas que existem na figura apresentada mais acima:

### **Feira:**

A Feira vai ser onde vão ser postos os Produtos à venda disponibilizados pelo Vendedor. Cada feira tem um id de forma a identifica-la no sistema, atuando assim como chave primária. Cada atributo excluindo a chave primária e as chaves estrangeiras atuam apenas como atributos que dão a conhecer a feira aos futuros clientes que quiserem fazer as suas compras lá

**Chave Primária:** idFeira.

**Chave estrangeira Vendedor-"username"-vend:** Esta chave estrangeira identifica quem é o dono da Feira, neste caso o "username" do Vendedor.

**Chave estrangeira Funcionário-"username"-func:** Esta chave estrangeira, não identifica quem foi o funcionário que criou a feira, no entanto, está associado à mesma, apesar de não ser identificado o "username".

### **Funcionário:**

O Funcionário atua como criador, gerente de Feiras e do Perfil dos Vendedores. Desta forma, após ser aprovado pela Câmara a venda de produtos de um Vendedor na FairTrade, o Funcionário vai criar um perfil para o Vendedor, que difere dos Clientes, e vai também criar a página da Feira do Vendedor respetivo, ficando ao critério do Vendedor de como a quer personalizar, etc. A sua chave primária é o seu "username", para apenas saber como identificar os Funcionários da Fairtrade. Os seus atributos são apenas a "password" e o "email" do Funcionário para este dar "login" na sua conta.

### **Comentário:**

A tabela do Comentário está presente de forma a registar um comentário/uma avaliação de um produto caso o Cliente decida fazer. Desta forma os únicos atributos presentes são o "rating" dado ao produto e o devido Comentário. O Cliente após a compra de um produto pode escolher fazer um comentário e/ou dar um "rating", daí estes atributos poderem ser ambos "NULL". Cada comentário tem um ID de forma a ser facilmente associado e encontrado pelo sistema, sendo este gerado aleatoriamente. As únicas chaves estrangeiras relevantes a descrever são:

**Cliente-"username"-cliente:** Esta chave estrangeira identifica o Cliente que fez o comentário, não havendo anonimato nestas situações.

**Produto-idProduto:** Esta chave refere-se ao ID do produto que está a ser avaliado.

**Produto-Feira-idFeira:** Esta chave é importante porque a avaliação dos produtos de uma dada feira pode afetar a sua popularidade.

### ***Cliente Compra:***

Esta tabela tem como propósito anotar na base de dados quando o Cliente efetua a compra de um produto. Como dá para ver na tabela, esta mesma possui atributos como o preço do produto, o endereço onde o produto vai ser entregue e o método de pagamento utilizado para efetuar a compra. Como chave primária temos o ID da compra (id-Compra) para ser facilmente identificada pelo sistema. Em relação às chaves estrangeiras, eis aquelas relevantes referir a sua utilidade:

**Cliente-"username"-cliente:** esta chave estrangeira é usada para identificar o cliente que efetuou a compra do produto.

**Produto-idProduto:** esta chave refere-se ao produto que foi comprado.

**Produto-Feira-idFeira:** Por fim, esta chave refere-se à feira onde este produto foi comprado.

### ***"chatbox":***

A "chatbox" tem como propósito resolver uma das maiores diferenças entre o mercado das feiras e uma loja online normal, a comunicação entre Vendedores e Clientes. Desta forma é possível para um cliente e um vendedor negociarem entre si ou esclarecer dúvidas por parte de um cliente em relação a um dos produtos disponibilizados na feira do respetivo vendedor. Sendo assim, só existem 2 atributos sendo eles a mensagem de um cliente e a resposta do vendedor. Como nem sempre há uma comunicação estabelecida entre estas duas entidades, ambos atributos podem ser "NULL". Em comparação com as outras tabelas deste modelo, a "chatbox" é a única tabela que possui duas chaves primárias, sendo elas o ID da mensagem do cliente e o ID da mensagem vendedor que estarão a comunicar entre si, caso situação surja. Da mesma forma as chaves estrangeiras são as respetivas chaves primárias que identificam o respetivo cliente e o respetivo vendedor.

### ***Cliente:***

A tabela do cliente aborda atributos como "email" e "password", que servem para o cliente poder entrar na sua conta após se ter registado, este depois pode personalizar o seu nome, método de pagamento, etc. No entanto este possui atributos que podem afetar a sua pesquisa ou até mesmo compra de produtos, nomeadamente o seu "rating" e histórico, que vão influenciar nos tipos de produtos que compra. Por fim, tem o carrinho com produtos escolhidos pelo cliente, este pode ou não estar vazio. A sua chave primária que vai identificar cada cliente no sistema é o seu "username".

### **Vendedor:**

A entidade de vendedor atua como dono da feira, o perfil de um vendedor é criado por um funcionário qualquer. Este mesmo possui atributos como "email" e "password" que são necessários para este dar "login" na sua conta. Para além do nome, o vendedor tem um "rating", que é alterado com base nas avaliações feitas pelos clientes e um método de pagamento. Esta entidade apenas tem uma chave estrangeira que é: **Funcionário-"username"-func**: esta chave refere-se ao funcionário que criou o perfil do respetivo vendedor, isto sem o identificar.

### **Produto:**

A tabela do produto refere-se às características do produto que está a ser vendido. Neste caso é abordada a descrição do produto, o seu preço, tipo, nome, etc. Um produto também pode ter um "rating" que é afetado por uma avaliação que lhe é feito, para além disso, também podemos ver se este produto está em stock ou não e se sim, quantos produtos este tem em stock. Como chaves estrangeiras a referir temos:

-> **Feira-idFeira**: esta chave refere-se a que feira este produto pertence.

-> **Feira-Vendedor-"username"-vend**: esta chave refere-se ao vendedor que está a vender o produto apresentado.

Agora no que toca a parte dos Relacionamentos entre as entidades, eis uma breve explicação de como foi o nosso raciocínio durante o seu desenvolvimento:

**Cliente - Produto (Cliente Compra):  $N:M$**  -> Pode surgir uma situação em que o cliente decide fazer uma compra, visto que vários clientes podem comprar vários produtos, surge aí a tabela Cliente Compra, onde 0 ou mais clientes podem comprar 0 ou mais produtos, daí existir este relacionamento.

**Cliente - Produto (Comentário):  $N:M$**  -> Da mesma maneira que 0 ou muitos clientes podem comprar um produto, 0 ou muitos clientes podem fazer comentários em relação à sua satisfação com o seu produto, daí o motivo deste relacionamento existir.

**Feira - Produto:  $1:N$**  -> Uma feira é capaz de disponibilizar vários produtos que estejam à venda e associados a essa feira, esses produtos estarão associados a essa feira.

**Vendedor - Feira:  $1:1$**  -> Um vendedor apenas pode estar associado a uma feira sendo identificado a que feira está associado.

**Cliente - Vendedor ("chatbox"):**  $N:M$  -> Vários clientes podem negociar com muitos vendedores com o intuito de chegar a um melhor negocio, desta forma existe a "chatbox" que leva a que 0 ou vários clientes negoceiem com 0 ou vários vendedores, sendo ambos identificados pelo sistema.

**Funcionário - Vendedor:**  $1:N$  -> Um funcionário cria o perfil de vários vendedores, no entanto, ao contrário dos outros relacionamentos não é identificado o funcionário que cria estes perfis.

**Funcionário - Feira:**  $1:N$  -> Da mesma forma que um funcionário cria o perfil de um vendedor, este também cria a feira a qual o vendedor está associado, não identificando o funcionário que criou a pagina da feira.

# 5 Interface do Sistema

## 5.1 Estrutura Geral das Interfaces

Apresentamos, de seguida, os esboços das seguintes funcionalidades: efetuar "login", visitar a Página Inicial, Perfil do Cliente, Perfil do Vendedor, visitar uma Feira e, por último, selecionar um Produto.

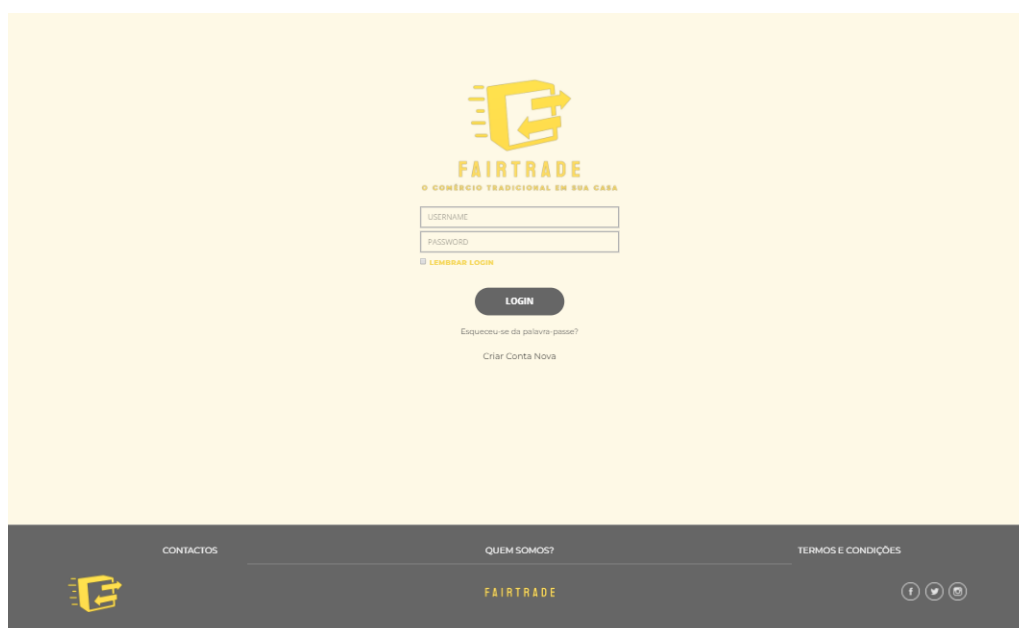


Figura 5.1: "login"



Figura 5.2: Página Inicial

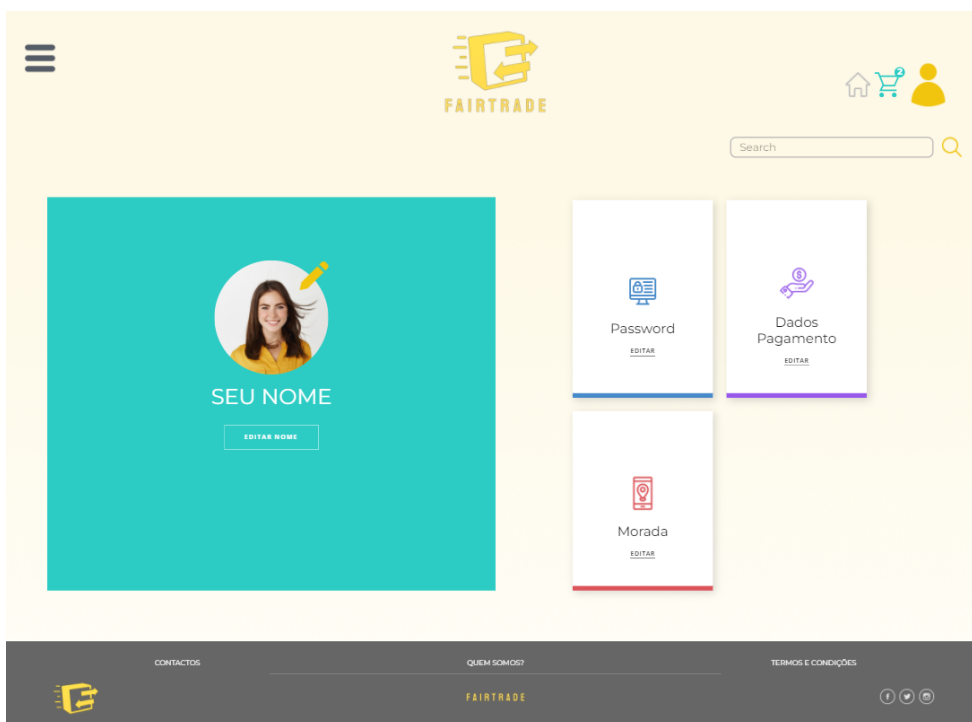


Figura 5.3: Perfil do Cliente

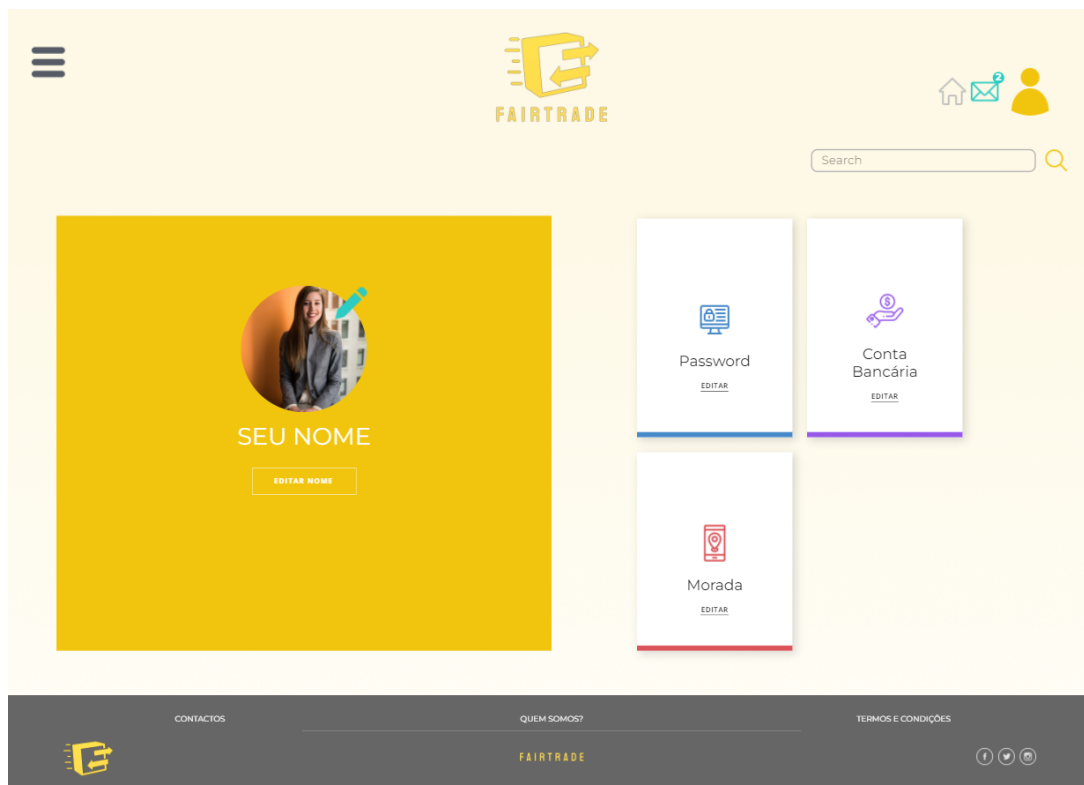


Figura 5.4: Perfil do Vendedor

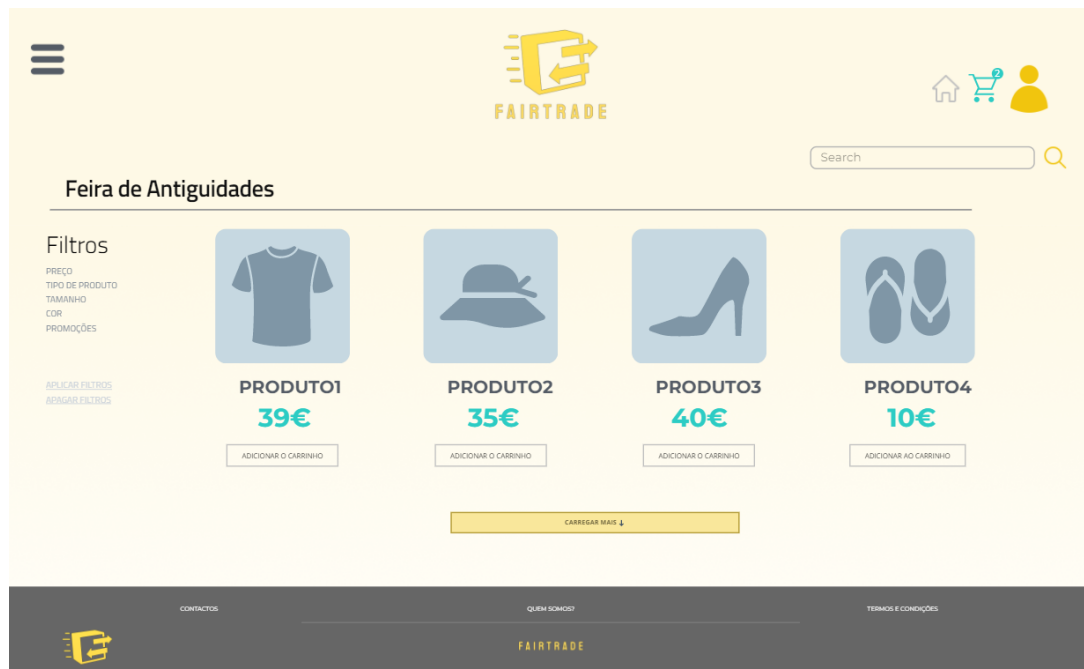


Figura 5.5: Feira



Figura 5.6: Produto

## 5.2 Caracterização das Interfaces

No que se refere ao aspeto da interface da aplicação, a equipa refletiu, primeiramente, sobre quais seriam os objetivos a alcançar com a escolha de cores, formas, posição das diferentes funcionalidades do sistema, entre outras.

O grupo pretende que os utilizadores da aplicação considerem a utilização da mesma intuitiva e atraente, tendo assim utilizado um formato simples, claro e harmonioso.



## 6 Implementação da Aplicação

Neste capítulo vamos abordar a terceira fase do projeto prático onde procedemos à implementação dos requisitos e das funcionalidades acima descritos. Nesta fase, foi essencial fazer uma pesquisa sobre as ferramentas e os recursos que iríamos utilizar para uma correta utilização dos mesmos. Mesmo assim, tivemos várias dificuldades ao longo da implementação que resultaram numa aplicação final com muitas das funcionalidades que pretendíamos implementar não funcionais.

### 6.1 Processo de Implementação

Quando começamos a implementação, decidimos criar primeiro as páginas que os funcionários iriam utilizar para criar as feiras e as contas dos vendedores. Isto também implicou criar as definições das tabelas para a base de dados, que foram alteradas ao longo do projeto para incluir variáveis que não tínhamos no modelo lógico inicial da base de dados. Depois da implementação destas páginas estar feita, tentamos usar o API "identity" do "ASP.NET" para criar roles que definiam que contas podiam aceder a que páginas. Deparamo-nos com vários problemas que resultaram em usarmos o API simplesmente para a conta do Cliente. Esta decisão foi tomada não só pelas dificuldades que tivemos, mas também pela falta de tempo que tínhamos para elaborar uma aplicação que podia ser "acedida" por 3 tipos de utilizador diferente.

Assim, decidimos reutilizar as páginas que tínhamos criado para o funcionário para a implementação do "lado" do Cliente, pois estas páginas tinham as listas que pretendíamos apresentar ao Cliente. Com o tempo que nos sobrava, decidimos implementar e trabalhar na apresentação das várias páginas de produtos, feiras e vendedores que o Cliente podia aceder, só ficando no projeto final a implementação em relação ao cliente. Finalmente, criamos as páginas para apresentar os detalhes de cada item (produtos, vendedores e feiras).

## **6.2 Análise e Avaliação da aplicação desenvolvida**

Mesmo assim, ficaram várias funcionalidades por implementar do "lado" do Cliente. Na versão final da nossa aplicação, o Cliente só pode aceder e ler os dados das várias feiras, produtos e vendedores que estão na base de dados (como não conseguimos implementar roles ou equivalente, retiramos as funcionalidades de criar, editar e eliminar os três elementos da aplicação). Não conseguimos implementar o carrinho de compras nem qualquer forma de comprar um produto e, consequentemente, não conseguimos implementar um sistema de críticas feitas por Clientes. Também está em falta as várias formas de pesquisa da aplicação, pois não estávamos a conseguir implementá-las de uma forma que fossem úteis para um Cliente. Por fim, também não implementamos a "chatbox", pois como só fizemos o "lado" do Cliente não iria haver forma de ele comunicar com um Vendedor.

Em suma, a parte da implementação do nosso projeto não correu muito bem, principalmente devido à falta de conhecimento e agilidade de todos os membros do grupo nas linguagens e ferramentas utilizadas no projeto. Esta inexperiência resultou em várias más decisões tomadas por nós no início da nossa implementação, pois não tínhamos suficiente conhecimento da plataforma para perceber o que iria ajudar na elaboração da aplicação e o que iria dificultar o processo.

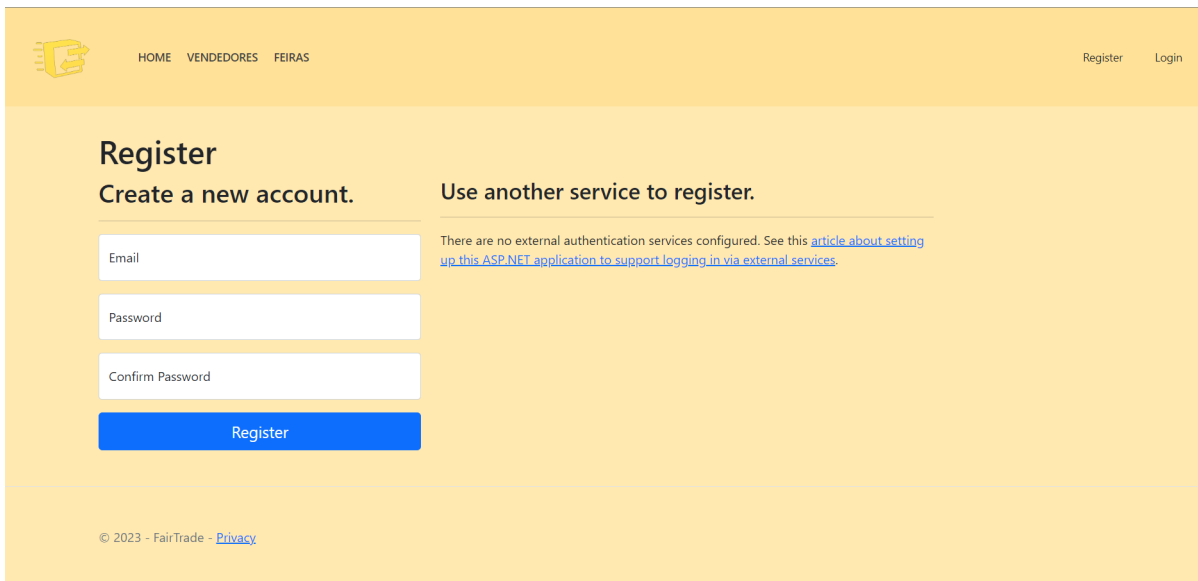
## **6.3 Arquitetura da aplicação e ferramentas utilizadas**

Na elaboração da nossa aplicação, decidimos utilizar a opção de "ASP.NET CORE Webapp", pois pareceu a opção que mais útil para o nosso projeto. Como referido anteriormente, usamos o "API" de autenticação "identity" disponível para uso no "Visual Studio". Para a base de dados, usamos o Microsoft SQL "Server", mais especificamente a versão "Express".

## 6.4 Serviços implementados e estrutura final da aplicação

### 6.4.1 Registrar utilizador

Um Utilizador consegue registar-se na nossa aplicação, tornando-se, desta forma, um utilizador que se pode autenticar e aceder às funcionalidades reservadas apenas a clientes autenticados.

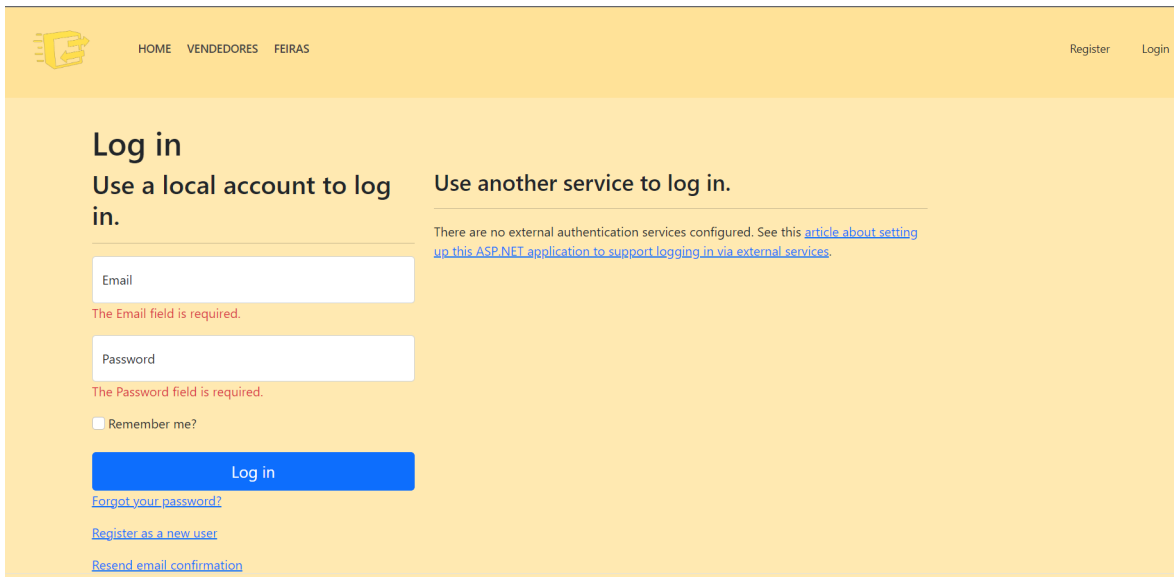


The screenshot displays a web application's registration interface. At the top, a yellow header bar contains a logo on the left, navigation links 'HOME', 'VENDEDORES', and 'FEIRAS' in the center, and 'Register' and 'Login' links on the right. The main content area is divided into two columns. The left column, titled 'Register' and 'Create a new account.', features three input fields labeled 'Email', 'Password', and 'Confirm Password', followed by a blue 'Register' button. The right column, titled 'Use another service to register.', contains a message stating that no external authentication services are configured, with a link to an article about setting up ASP.NET applications for external services. The footer of the page shows the copyright notice '© 2023 - FairTrade - Privacy'.

Figura 6.1: "Layout" para registar uma conta no sistema

## 6.4.2 Fazer "login"

Estando previamente registrado, um cliente consegue iniciar sessão na nossa aplicação, obtendo, deste modo, o acesso às funcionalidades do sistema definidas para um utilizador.



The screenshot displays a web application's login interface. At the top, a navigation bar includes a logo on the left and links for 'HOME', 'VENDEDORES', 'FEIRAS', 'Register', and 'Login' on the right. The main content area is divided into two sections. The left section, titled 'Log in', prompts the user to 'Use a local account to log in.' and contains two input fields: 'Email' and 'Password'. Below the 'Email' field is a red error message: 'The Email field is required.' Below the 'Password' field is a red error message: 'The Password field is required.' There is also a 'Remember me?' checkbox. A blue 'Log in' button is positioned below these fields. Underneath the button are three links: 'Forgot your password?', 'Register as a new user', and 'Resend email confirmation'. The right section, titled 'Use another service to log in.', includes a message stating 'There are no external authentication services configured. See this [article about setting up this ASP.NET application to support logging in via external services.](#)'

Figura 6.2: "Layout" para fazer "login"

### 6.4.3 Página inicial depois de dar "login"



Figura 6.3: "Layout" da página principal após o início de sessão

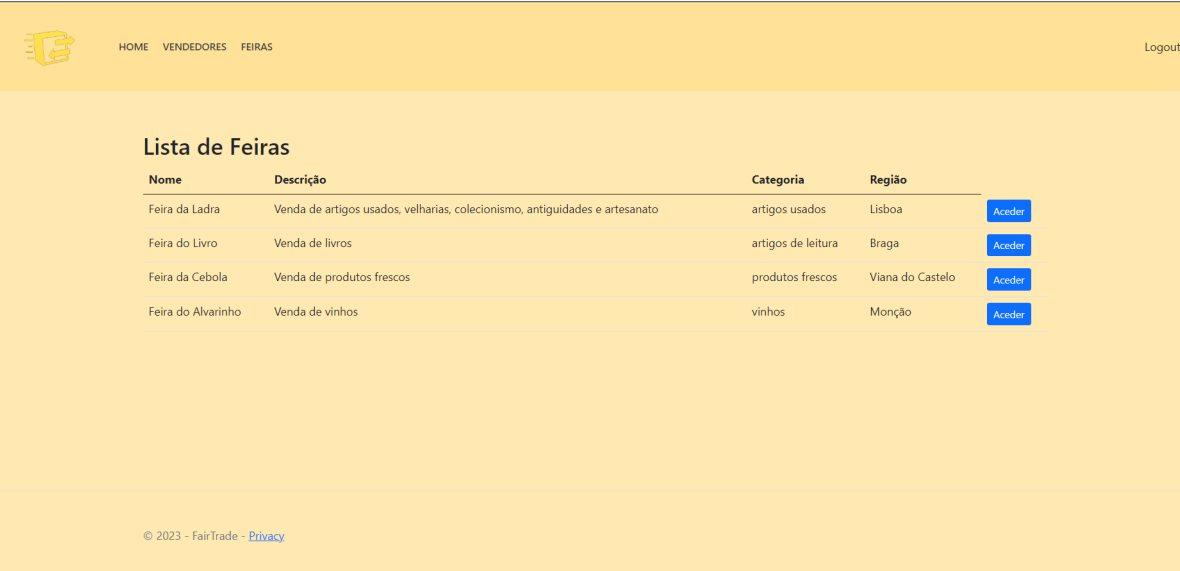
1. O objetivo foi fazer um "slide"
2. Quando clicado no nome da feira (centro da imagem) somos redirecionados para a página da feira em questão.
3. Quando clicado nos botões presentes na parte inferior da imagem, apresenta outra feira com outro texto capaz de redirecionar o cliente para a página da mesma.

Dificuldades na implementação do "slider":

1. Num estado inicial, as imagens ficavam mal formatadas devido ao "aspect ratio".
2. Após a criação de um elemento `<a>` que redirecionava o utilizador para a página da feira em questão, todos os textos ficavam sobrepostos.
3. Após a correção do último erro através de um "script", sempre que clicamos num botão, o texto altera, mesmo não passando para outro "slide".

## 6.4.4 Página da lista de feiras

Um cliente ao selecionar o botão *Feiras* irá ter acesso à lista de feiras disponíveis, nesta página para cada feira existe um botão *Aceder* que se selecionado irá levar o cliente para a página da feira correspondente.



Nome	Descrição	Categoria	Região	
Feira da Ladra	Venda de artigos usados, velharias, colecionismo, antiguidades e artesanato	artigos usados	Lisboa	<a href="#">Aceder</a>
Feira do Livro	Venda de livros	artigos de leitura	Braga	<a href="#">Aceder</a>
Feira da Cebola	Venda de produtos frescos	produtos frescos	Viana do Castelo	<a href="#">Aceder</a>
Feira do Alvarinho	Venda de vinhos	vinhos	Monção	<a href="#">Aceder</a>

© 2023 - FairTrade - [Privacy](#)

Figura 6.4: "Layout" da página de lista de feiras

## 6.4.5 Página de uma feira

Um cliente irá conseguir aceder a uma feira e ver todos os produtos que nela são vendidos, irá poder também seleccionar um produto ao clicar em *compra-me!* e aceder à página do mesmo.

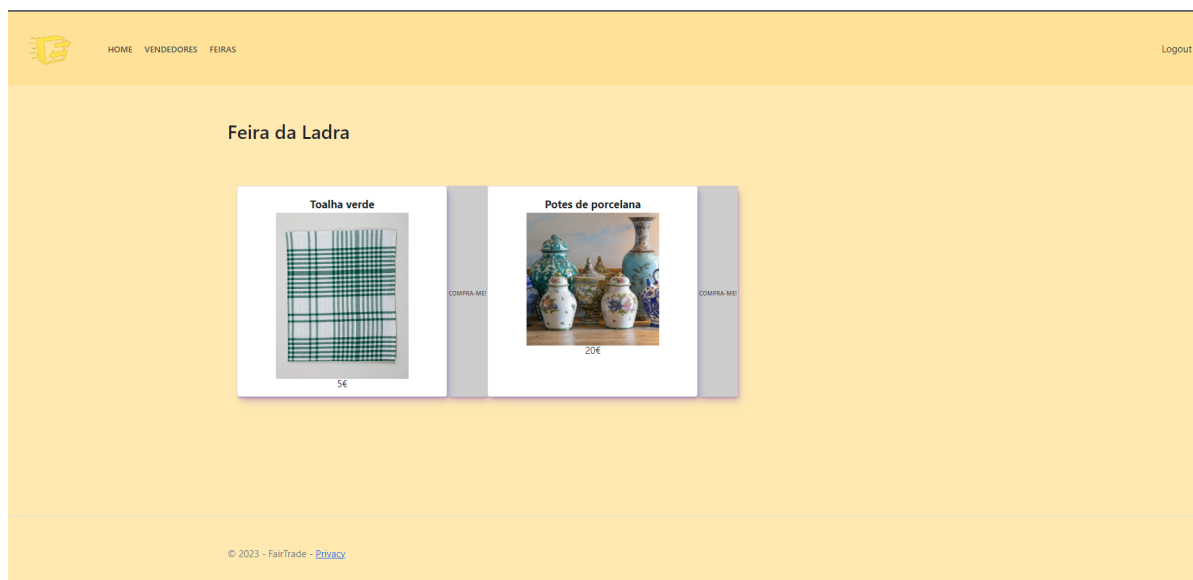


Figura 6.5: "Layout" da página de uma feira

## 6.4.6 Página de um produto

Ao seleccionar um produto o cliente vai ter acesso à página do produto onde vai poder ver várias características do produto seleccionado.

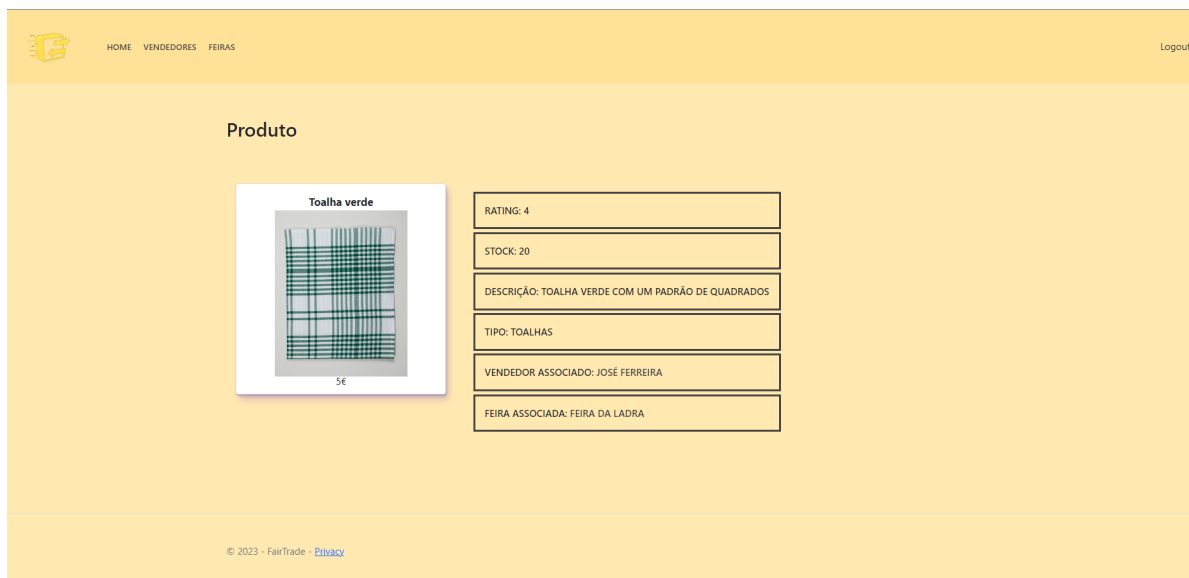
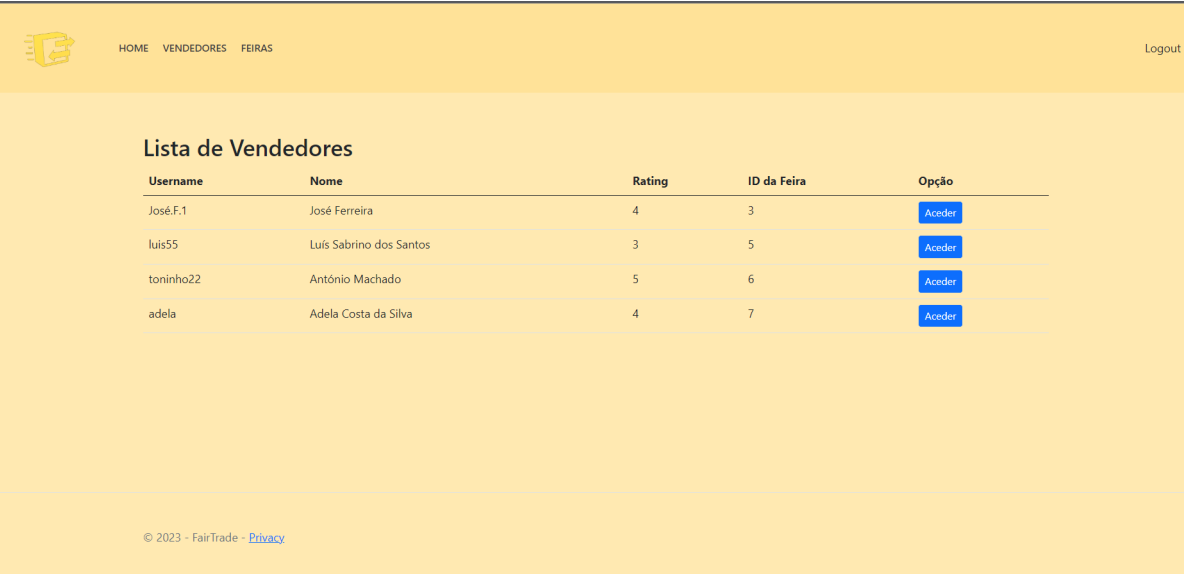


Figura 6.6: "Layout" da página de um produto



## 6.4.7 Página da lista de Vendedores

Um cliente ao selecionar Vendedores irá ter acesso à lista de Vendedores registados no sistema, nesta página para cada Vendedor existe um botão *Aceder* que se selecionado irá levar o cliente para a página de perfil do vendedor correspondente.



Username	Nome	Rating	ID da Feira	Opção
José.F.1	José Ferreira	4	3	<a href="#">Aceder</a>
luis55	Luís Sabrino dos Santos	3	5	<a href="#">Aceder</a>
toninho22	António Machado	5	6	<a href="#">Aceder</a>
adela	Adela Costa da Silva	4	7	<a href="#">Aceder</a>

© 2023 - FairTrade - [Privacy](#)

Figura 6.7: "Layout" da página de lista vendedores

## 6.4.8 Perfil do vendedor visto pelo cliente

Um cliente irá conseguir aceder ao perfil de um dado vendedor e ver algumas informações sobre o mesmo.

HOME VENDEDORES FEIRAS Logout

### Perfil de vendedor!

**José Ferreira**

RATING: 4

EMAIL: Zéf@GMAIL.COM

NOME: JOSÉ FERREIRA

FEIRA ASSOCIADA: FEIRA DA LADRA

### Lista de Produtos

Nome	Descrição	Preço	Rating	Tipo	Stock	
Toalha verde	Toalha verde com um padrão de quadrados	5	4	toalhas	20	<a href="#">Adquirir</a>
Potes de porcelana	Conjunto de porcelanas	20	5	velharias	4	<a href="#">Adquirir</a>

© 2023 - FairTrade - [Privacy](#)

Figura 6.8: "Layout" do perfil de um vendedor visto por um cliente

# 7 Conclusões e Trabalho Futuro

## 7.1 Conclusão

Em geral, achamos que o trabalho podia ter tido um resultado final melhor se tivéssemos sido mais realistas em relação as funcionalidades que pretendíamos implementar e tivéssemos gastado mais tempo a tentar perceber as especificidades das plataformas que utilizamos para desenvolver a aplicação. Foi claramente notável as dificuldades que surgem por causa de falta de planeamento, pois tivemos de adicionar e retirar vários elementos do projeto por não termos feito a preparação devida.

O resultado final foi uma aplicação que, apesar de funcional, tem em falta muitas das funcionalidades que pretendíamos implementar inicialmente. Apesar disto, achamos que a maioria dos problemas do trabalho surgiram devido à nossa falta de perícia com as ferramentas que pretendíamos utilizar e não de um mau trabalho feito nas fases anteriores do projeto, que foram essenciais para a elaboração do trabalho que fizemos.

## 7.2 Trabalho Futuro

Para um trabalho futuro na mesma plataforma, achamos que, relacionando melhor as etapas iniciais do projeto com o que pretendemos fazer e tendo um maior e melhor conhecimento das ferramentas que vamos utilizar irá resultar num projeto com objetivos mais realistas e com uma implementação de tudo o que se pretendia.

# Referências

- Wikipedia contributors. (2005, 4 de novembro). Feira. Wikipedia, The Free Encyclopedia. <https://pt.wikipedia.org/wiki/Feira> (Consultado em: 2022, novembro 20).
- Asp.net documentation. ([s.d.]). Microsoft.com. <https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-7.0> (Consultado em: 2023, janeiro 7).
- Graça, A. R. (2020, maio 27). Ser feirante: um trabalho que perdura na incerteza. JPN - JornalismoPortoNet. <https://www.jpn.up.pt/2020/05/27/ser-feirante-um-trabalho-que-perdura-na-incerteza/> (Consultado em: 2022, janeiro 10).