



Universidade do Minho  
Escola de Engenharia  
Licenciatura em Engenharia Informática

## Unidade Curricular de Laboratórios de Informática IV

Ano Letivo de 2023/2024

### Leilões online

Luís Borges (a96357)  
José Vasconcelos (a100763)  
Flávio Alexandre Silva (a97352)  
Bento João Concieiro Guimarães (a96296)  
Tiago Miguel Marques Pereira (a96429)

29 de janeiro de 2024

LI4

Data de Receção	
Responsável	
Avaliação	
Observações	

## Leilões online

Luís Borges (a96357)

José Vasconcelos (a100763)

Flávio Alexandre Silva (a97352)

Bento João Concieiro Guimarães (a96296)

Tiago Miguel Marques Pereira (a96429)

29 de janeiro de 2024

# Resumo

O seguinte projeto foi realizado no âmbito da unidade curricular Laboratórios de Informática IV, da licenciatura em Engenharia Informática da Universidade do Minho.

A primeira fase passou pela definição do sistema, que consiste numa apresentação geral do projeto. Seguidamente, foi feita a definição de requisitos, seguida da especificação e modelação de software e da conceção do sistema de dados. Posteriormente, desenvolveram-se os esboços das interfaces do sistema. Por fim, foi realizada a implementação, tendo como produto final uma plataforma de leilões completamente funcional.

**Área de Aplicação:** Sistemas de Bases de Dados, Engenharia Web.

**Palavras-Chave:** *SQL, SQL Server, C#, .NET Core, Leilões, Visual Studio, Requisitos, Engenharia de Software, UML, Use Cases, ComprasOnline.*

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Contextualização . . . . .	1
1.2	Fundamentação . . . . .	1
1.3	Objetivos . . . . .	1
1.4	Viabilidade . . . . .	2
1.5	Recursos a utilizar . . . . .	2
1.5.1	Recursos humanos . . . . .	2
1.5.2	Recursos não humanos . . . . .	2
1.6	Equipa de Trabalho . . . . .	3
1.7	Plano de Execução de Trabalho . . . . .	4
<b>2</b>	<b>Definição de Requisitos</b>	<b>5</b>
2.1	Apresentação da estratégia e método . . . . .	5
2.2	Descrição geral dos requisitos (funcionais e não funcionais) levantados . . . . .	6
2.2.1	Requisitos funcionais . . . . .	6
RF01	- Registo . . . . .	6
RF02	- Autenticação . . . . .	6
RF03	- Criação de um leilão . . . . .	7
RF04	- Aprovação de um leilão . . . . .	7
RF05	- Cancelamento de um leilão . . . . .	7
RF06	- Consulta de perfil . . . . .	8
RF07	- Consulta da página inicial . . . . .	8
RF08	- Licitação de um item . . . . .	8
RF09	- Desconexão . . . . .	9
RF10	- Consulta de um leilão . . . . .	9
RF11	- Consulta de estatísticas . . . . .	9
RF12	- Adicionar saldo à conta . . . . .	10
2.2.1	Requisitos Não funcionais . . . . .	10
RNF01	- Ambiente de execução do programa . . . . .	10
RNF02	- Ferramentas utilizadas para o Desenvolvimento do programa . . . . .	10
RNF03	- Tempo Operacional do Servidor . . . . .	11
RNF04	- Tempo de Execução das funcionalidades interativas . . . . .	11
2.3	Validação dos requisitos estabelecidos . . . . .	11
<b>3</b>	<b>Especificação e Modelação de Software</b>	<b>12</b>
3.1	Apresentação geral da especificação . . . . .	12

3.2	Aspetos estruturais . . . . .	13
3.3	Aspetos comportamentais . . . . .	14
3.3.1	Use cases de utilização geral . . . . .	15
3.3.2	Use cases do utilizador regular . . . . .	15
3.3.3	Use cases do administrador . . . . .	19
3.3.4	Diagramas de atividades . . . . .	20
<b>4</b>	<b>Conceção do Sistema de Dados</b>	<b>23</b>
4.1	Apresentação geral da estrutura do sistema de dados . . . . .	23
4.2	Descrição detalhada dos vários elementos de dados e seus relacionamentos . . . . .	24
4.2.1	Utilizador . . . . .	24
4.2.2	Produto . . . . .	25
4.2.3	Licitação . . . . .	26
4.2.4	Leilão . . . . .	27
4.2.5	Relação 1:1 <i>Leilão_has_Produto</i> . . . . .	28
4.2.6	Relação 1:N <i>User_creates/deletes_Leilão</i> . . . . .	28
<b>5</b>	<b>Esboço das Interfaces do Sistema</b>	<b>29</b>
5.1	Estrutura Geral das Interfaces do Sistema . . . . .	29
5.2	Caraterização das Interfaces . . . . .	30
5.2.1	Página Inicial . . . . .	30
5.2.2	Página de Criar Leilão . . . . .	31
5.2.3	Página de Perfil . . . . .	32
5.2.4	Página de Leilão . . . . .	33
5.2.5	Página de Estatísticas . . . . .	34
5.2.6	Página de Autenticação . . . . .	35
<b>6</b>	<b>Implementação</b>	<b>36</b>
6.1	Base de dados . . . . .	36
6.2	Back-end . . . . .	36
6.2.1	<i>HomeController</i> . . . . .	37
6.2.2	<i>LeilaoController</i> . . . . .	37
6.2.3	<i>LicitacaoController</i> . . . . .	37
6.2.4	<i>ProdutoController</i> . . . . .	37
6.2.5	<i>StatsController</i> . . . . .	37
6.2.6	<i>UtilizadorController</i> . . . . .	37
6.2.7	Outros ficheiros relevantes . . . . .	38
6.3	Front-end . . . . .	38
<b>7</b>	<b>Conclusões e Trabalho Futuro</b>	<b>39</b>
	<b>Anexos</b>	<b>41</b>

# Lista de Figuras

1.1	Diagrama de Gantt . . . . .	4
3.1	Modelo de domínio . . . . .	13
3.2	Diagrama de <i>use cases</i> . . . . .	14
3.3	Diagrama de atividades do utilizador regular . . . . .	21
3.4	Diagrama de atividades do administrador . . . . .	22
4.1	Modelo lógico . . . . .	23
5.1	Estrutura das Interfaces do Sistema . . . . .	29
5.2	Página Inicial . . . . .	30
5.3	Página de Criar Leilão . . . . .	31
5.4	Página de Perfil . . . . .	32
5.5	Página de Leilão . . . . .	33
5.6	Página de Estatísticas . . . . .	34
5.7	Página de Autenticação . . . . .	35

# Lista de Tabelas

4.1	Entidade Utilizador . . . . .	24
4.2	Entidade Produto . . . . .	25
4.3	Entidade Licitação . . . . .	26
4.4	Entidade Leilão . . . . .	27
4.5	Relação <i>Leilão_has_Produto</i> . . . . .	28
4.6	Relação <i>User_creates/deletes_Leilão</i> . . . . .	28



# 1 Introdução

## 1.1 Contextualização

O leilão está presente na história da humanidade há mais de 2500 anos. Desde os tempos do império grego até à atualidade, tem sido um dos mais importantes meios de compra e venda de mercadorias e serviços. Devido à popularidade da prática, a empresa portuguesa ComprasOnline, que já tem plataformas digitais de compra e venda de produtos, decidiu investir numa plataforma de leilões.

## 1.2 Fundamentação

Tendo em conta a recente popularização da implementação em meios digitais de conceitos tradicionalmente físicos, tal como a troca e venda de produtos via sites como eBay ou OLX, acreditamos que haja uma janela de mercado ainda não explorada no campo dos leilões. Para além disso, o nosso projeto resolve todos os problemas relacionados com limites físicos dos leilões.

## 1.3 Objetivos

Com esta plataforma pretendemos criar um ambiente próspero para a realização de leilões, modernizando o meio e permitindo:

- Aproveitar a ausência de limitações físicas para aumentar o alcance dos leilões.
- Tornar o processo do leilão simples e providenciar ao utilizador uma experiência sem problemas.
- Expandir o tipo de produtos vendidos para além do que, tradicionalmente, é exposto em leilões.
- Capitalizar nesta oportunidade de mercado e gerar lucros para a empresa ComprasOnline.

## 1.4 Viabilidade

É sabido que a Internet é cada vez mais usada por todos os ramos da população e que existe, em Portugal, um mercado para a compra e venda de produtos, com empresas como o OLX, tal como existe mercado *online* para a compra de produtos em grandes superfícies comerciais, seja roupa, comida, mobília, etc. No entanto, em território português, não há nenhuma empresa estabelecida na área dos leilões que se compare, em termos de utilizadores mensais, às plataformas mencionadas. Assim, concluiu-se que áreas "vizinhas" à dos leilões têm bastante sucesso, mas os leilões em si não têm um projeto estabelecido, pelo que acreditamos estar perante uma lacuna no mercado que pretendemos aproveitar.

## 1.5 Recursos a utilizar

Para este trabalho vão ser necessários vários tipos de recursos diferentes. Compradores, vendedores, objetos para leilão e o preço.

Neste projeto específico, cinco pessoas serão responsáveis pela construção de um software que gere os leilões de maneira eficiente.

Dividimos os recursos entre humanos e não humanos:

### 1.5.1 Recursos humanos

- Os 5 elementos do grupo responsável pelo projeto.
- Funcionários da ComprasOnline.
- 5 especialistas em leilões.

### 1.5.2 Recursos não humanos

- Microsoft Visual Studio.
- Microsoft .NET C#.
- Microsoft SQL Server.
- Microsoft Office.
- Sistema de gestão de base de dados.

## 1.6 Equipa de Trabalho

A equipa responsável pelo projeto é constituída por várias pessoas que desempenham funções de diferentes áreas, onde a colaboração de todos os envolvidos é importante para o sucesso deste projeto.

Nessa equipa estão envolvidos:

*Pessoal da ComprasOnline:* Funcionários da ComprasOnline serão responsáveis por fornecerem a informação necessária para o desenvolvimento do projeto.

*Grupo contratado:* Um grupo de 5 alunos de LEI da Universidade do Minho.

- Luís Tiago Pereira Borges (a96357)
- José Pedro Torres Vasconcelos (a100763)
- Flávio Alexandre Silva (a97352)
- Bento João Concieiro Guimarães (a96296)
- Tiago Miguel Marques Pereira (a96429)

*Outros:*

- Clientes e vendedores experientes na área dos leilões, que serão de grande importância no esclarecimento de dúvidas.

# 1.7 Plano de Execução de Trabalho

Ao planejar o desenvolvimento, tomou-se a iniciativa de dividir tarefas pela equipa de trabalho externa. Para tal, recorreu-se ao diagrama de GANTT, ferramenta esta que se mostrou bastante útil não só para a distribuição de tarefas, como também para a estipulação de prazos.

LI4

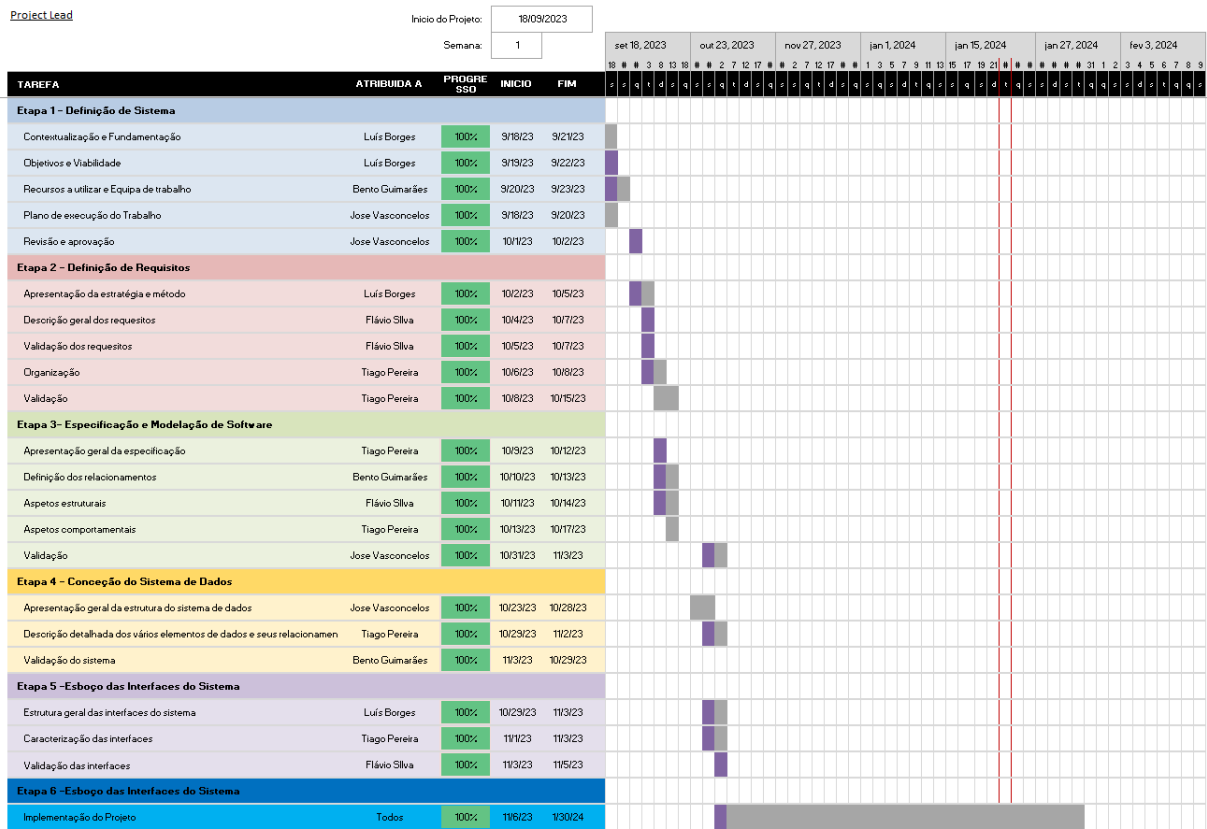


Figura 1.1: Diagrama de Gantt

## 2 Definição de Requisitos

### 2.1 Apresentação da estratégia e método

De forma a definirmos os requisitos da melhor forma possível, adotamos vários métodos. Primeiramente, a equipa reuniu-se e foi participar num leilão presencial, de forma a perceber conceito original.

No dia seguinte, a equipa distribuiu uma pequena parte do orçamento pelos 5 membros e participaram todos em leilões *online*, nas várias plataformas existentes, tanto nacionais como internacionais, de forma a explorá-las e compreendê-las

Por fim, a equipa fez entrevistas com 5 especialistas portugueses em leilões, incluindo alguns dos maiores clientes do país, onde procurou saber quais as funcionalidades que eles gostariam de ver implementadas.

## **2.2 Descrição geral dos requisitos (funcionais e não funcionais) levantados**

Segue-se uma breve descrição dos requisitos obtidos pela equipa, separados em funcionais e não funcionais.

### **2.2.1 Requisitos funcionais**

#### **RF01 - Registo**

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

1. O utilizador deve poder criar uma conta.

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

1. O sistema deve pedir ao utilizador o nome completo, nome de utilizador, palavra-passe, endereço de correio eletrónico e NIF.
2. O nome de utilizador, endereço de correio eletrónico e NIF devem ser valores únicos. Caso contrário, o utilizador deve ser notificado. O registo pode apenas ser concluído quando o utilizador mudar esses valores para valores únicos.
3. O sistema deve registar a conta.

#### **RF02 - Autenticação**

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

1. O utilizador/administrador deve poder aceder à sua conta.

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

1. O sistema deve pedir ao utilizador o seu nome de utilizador e a sua password.
2. O sistema deve confirmar se os dados estão corretos.
3. Em caso positivo, o sistema deve dar ao utilizador acesso ao site.
4. Em caso negativo, o sistema deve apresentar uma mensagem de erro.

## **RF03 - Criação de um leilão**

### **Requisitos do utilizador**

1. O utilizador deve poder criar um leilão.

### **Requisitos do sistema**

1. O sistema deve pedir ao utilizador o nome do item a leiloar, uma imagem do item, uma descrição do item, o número de donos anteriores, o preço de licitação mínimo, a data de início do leilão e a data final.
2. O sistema deve atribuir um número de identificação único para o leilão.
3. O sistema deve registar o leilão, mantendo-o em estado pendente (até receber aprovação do administrador).

## **RF04 - Aprovação de um leilão**

### **Requisitos do utilizador**

1. O administrador deve poder aprovar a criação de um leilão.

### **Requisitos do sistema**

1. O sistema deve dar ao administrador a opção de aprovar ou não aprovar um leilão.
2. Caso seja aprovado, deve ser registado no sistema.
3. Caso não seja aprovado, os dados devem ser eliminados.

## **RF05 - Cancelamento de um leilão**

### **Requisitos do utilizador**

1. O utilizador deve poder cancelar um leilão que criou anteriormente, desde que o faça com uma hora de antecedência do início do leilão em questão.

### **Requisitos do sistema**

1. O sistema deve verificar que falta mais de uma hora para o início do leilão.
2. Caso a condição acima se verifique, o leilão deve ser eliminado do sistema.

3. Caso contrário, o sistema deve apresentar uma mensagem de erro.

## **RF06 - Consulta de perfil**

### **Requisitos do utilizador**

1. O utilizador deve poder consultar o seu perfil, mas não o dos outros.

### **Requisitos do sistema**

1. O sistema deve mostrar ao utilizador o seu perfil, onde se encontram as suas informações pessoais e a sua lista de leilões.

## **RF07 - Consulta da página inicial**

### **Requisitos do utilizador**

1. O utilizador deve poder consultar a página inicial

### **Requisitos do sistema**

1. O sistema deve mostrar uma lista de leilões em que, para cada leilão, deve mostrar o nome do item e o valor da licitação atual. Deve ser disponibilizada, também, para cada leilão na lista, uma opção para aceder à página desse leilão.

## **RF08 - Licitação de um item**

### **Requisitos do utilizador**

1. O utilizador deve poder licitar um item.

### **Requisitos do sistema**

1. O sistema deve pedir um valor para licitar.
2. O sistema deve confirmar que o valor é válido. Caso contrário, deve apresentar um erro.
3. O sistema deve confirmar que o valor está acima do valor mínimo. Caso contrário, deve apresentar um erro.
4. O sistema deve confirmar que o valor está acima do maior valor licitado até agora. Caso



contrário, deve apresentar um erro.

5. O sistema deve verificar que o utilizador a licitar não é o criador do leilão. Caso contrário, deve apresentar um erro.
6. O sistema deve registar a licitação, associando-a ao utilizador que a fez.
7. O sistema deve retirar ao saldo do utilizador o valor da licitação.

## **RF09 - Desconexão**

### **Requisitos do utilizador**

1. O utilizador/administrador deve poder sair da sua conta.

### **Requisitos do sistema**

1. O sistema deve permitir ao utilizador sair da conta.
2. Esta opção deve estar localizada no perfil do utilizador.

## **RF10 - Consulta de um leilão**

### **Requisitos do utilizador**

1. O utilizador deve poder consultar os leilões da página inicial.

### **Requisitos do sistema**

1. O sistema deve apresentar ao utilizador a informação relativa ao leilão, nomeadamente, o nome do item, uma descrição, uma imagem, o número de donos anteriores, o valor inicial de licitação, licitação atual, data de início e data final.

## **RF11 - Consulta de estatísticas**

### **Requisitos do utilizador**

1. O utilizador deve poder aceder a uma página com as suas estatísticas.

### **Requisitos do sistema**

1. O sistema deve apresentar uma lista com os 10 últimos itens vendidos e uma lista com as 10 vendas mais caras.
2. Deve também ser apresentado o número total de vendas e o total de dinheiro, em vendas.
3. Para cada item em cada uma das listas, deve ser apresentado o nome do item e o valor de venda.

## **RF12 - Adicionar saldo à conta**

### **Requisitos do utilizador**

1. O utilizador deve poder adicionar saldo à sua conta.

### **Requisitos do sistema**

1. O sistema deve pedir a quantia a adicionar.
2. O sistema deve adicionar a quantia introduzida ao saldo.

## **2.2.1 Requisitos Não funcionais**

### **RNF01 - Ambiente de execução do programa**

- A parte do sistema com que os utilizadores interagem deverá executar a partir de um browser de internet, por enquanto a parte do banco/armazenamento de dados e da lógica de negócio deverá correr em ambiente Microsoft Windows.

### **RNF02 - Ferramentas utilizadas para o Desenvolvimento do programa**

- Para o desenvolvimento do sistema, utilizamos obrigatoriamente a framework NET e a linguagem de programação C#. Para a gestão de dados necessária para o funcionamento do sistema, deverá ser usado o sistema Microsoft SQL Server.

## **RNF03 - Tempo Operacional do Servidor**

- O servidor deverá estar operacional 24h por 7 dias da semana.

## **RNF04- Tempo de Execução das funcionalidades interativas**

- O tempo de execução das várias funcionalidades não deverá ser superior 10 segundos.

## **2.3 Validação dos requisitos estabelecidos**

Juntamos todas as ideias registadas e reunimos com o cliente para obter uma avaliação e obter sugestões para este projeto de modo a validar os requisitos. Após várias entrevistas e uma avaliação ótima dos requisitos finais podemos confortavelmente dizer que estão válidos.

## 3 Especificação e Modelação de Software

### 3.1 Apresentação geral da especificação

Após a definição dos requisitos funcionais e não funcionais, passamos a compreender não só como a aplicação de leilões deve ser construída, mas também como esta se deve comportar. Com o objetivo de criar diagramas que suportem a especificação, recorreremos à notação **UML**.

Inicialmente, construímos o modelo de domínio, no qual definimos os aspetos estruturais do nosso software. Em seguida definimos os *Use cases* de utilização geral, que servem para definir como a nossa aplicação se deve comportar, seguindo, por fim, para a elaboração do diagrama de *Use cases*.

## 3.2 Aspetos estruturais

O modelo de domínio, apresentado de seguida, permite uma compreensão visual das entidades do sistema e da sua interação, pelo que constitui uma parte essencial de qualquer projeto.

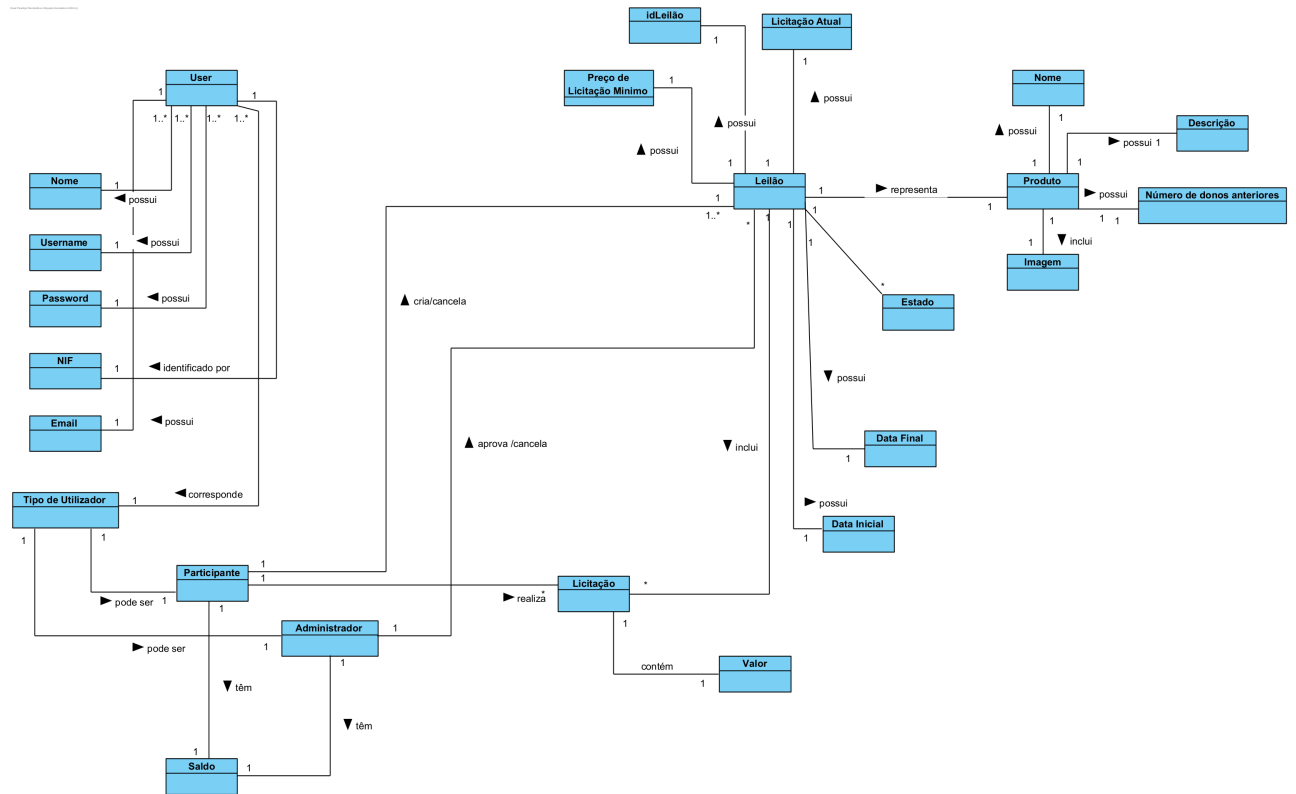


Figura 3.1: Modelo de domínio

### 3.3 Aspetos comportamentais

A nossa aplicação é apenas dividida entre utilizadores e administradores, sendo que o utilizador pode desempenhar tanto o papel de comprador como de vendedor (com as devidas restrições, evidentemente). Já o administrador tem um papel que se resume há aprovação de leilões. Esta dinâmica é exposta de forma mais clara no seguinte diagrama de *use cases*:

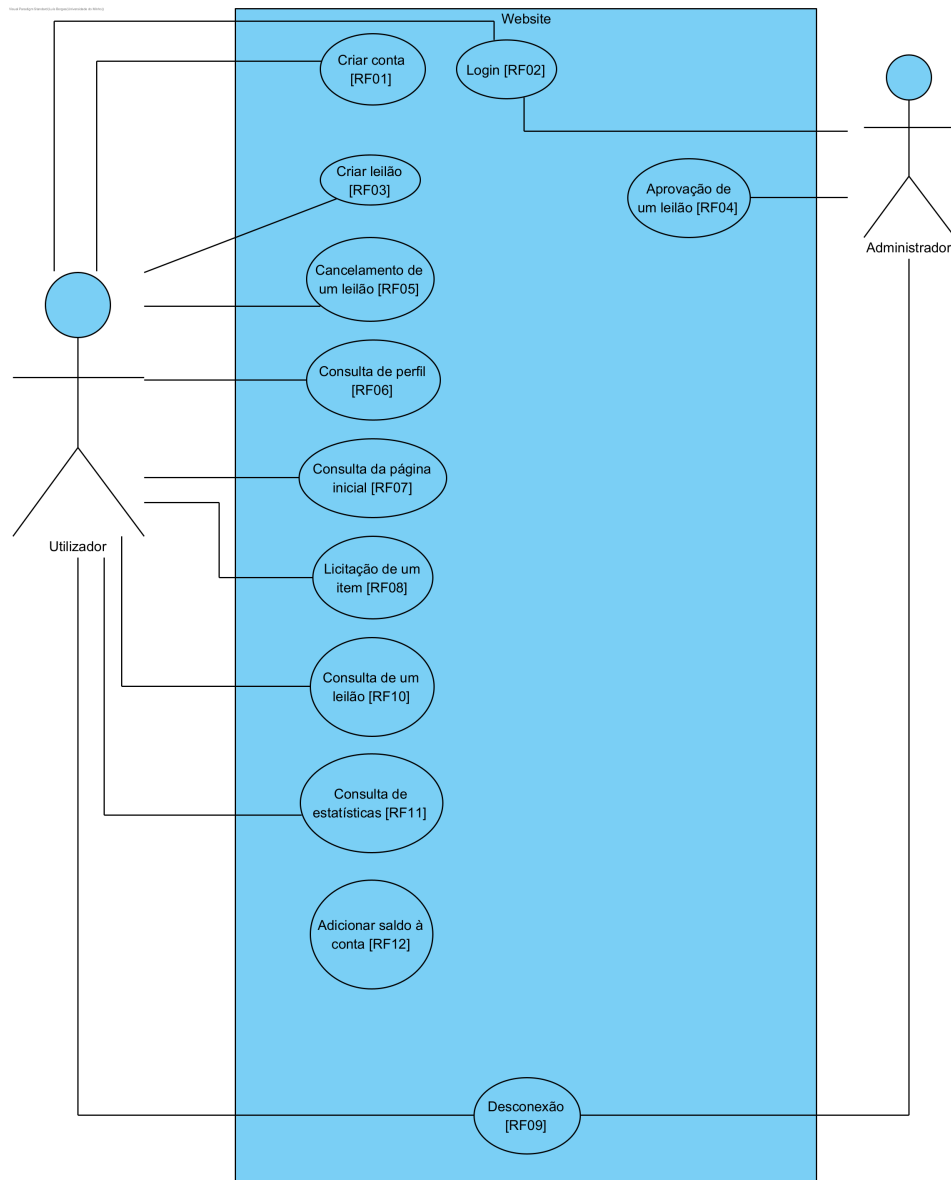


Figura 3.2: Diagrama de *use cases*

### 3.3.1 Use cases de utilização geral

#### Login (RF02)

Um utilizador que já tenha criado conta, pode entrar na mesma realizando *login*. Para isso, deve fornecer o *username* e *password*.

O sistema começa por verificar a existência da conta através do *username*. Se o *username* não estiver a ser utilizado, o cliente recebe uma mensagem de erro a informar que a conta não existe.

Se a conta existir, o sistema verifica se a *password* é a que está associada ao *username*. Se a *password* estiver incorreta, o cliente recebe uma mensagem de erro. Se estiver correta, o sistema informa o cliente que este foi autenticado corretamente, permitindo assim a utilização das várias funcionalidades da aplicação.

#### Desconexão (RF09)

O utilizador e o administrador devem poder sair da sua conta. Esta opção deve estar localizada no perfil do utilizador e não precisam de ser feitas quaisquer verificações pelo sistema para desconectar.

### 3.3.2 Use cases do utilizador regular

#### Criar conta (RF01)

É necessário criar conta para utilizar o website, quer seja vendedor ou não. Os dados seguintes são obrigatórios para a criação de uma conta:

1. Nome completo;
2. Username;
3. Email;
4. Password;
5. NIF.

O nome de utilizador, NIF e email devem ser únicos, ou seja, não devem existir dois utilizadores que partilhem um ou mais destes dados. Desta forma, antes da criação de uma conta, o sistema vai verificar se já existe algum utilizador algum dos dados. Se for o caso, o sistema informa dos dados que não podem ser utilizados.

### **Criar leilão (RF03)**

Qualquer utilizador pode criar um leilão, ou seja, o sistema não diferencia entre vendedor e não vendedor. Para a criação de um leilão devem ser fornecidos os seguintes dados:

1. Nome do item a ser leiloado;
2. Descrição do item;
3. Número de donos anteriores (opcional);
4. Preço de licitação mínimo;
5. Data de início;
6. Data final;
7. Imagem do item (preferencialmente com proporções 1:1);

Antes do sistema adicionar um leilão, este verifica se todos os campos anteriores estão preenchidos, exceto o número de donos anteriores, que é opcional. Se algum campo estiver em falta, o sistema avisa do(s) campo(s) em falta e não termina a publicação do leilão.

Se todos os campos estiveram preenchidos, o sistema verifica se a data final é superior à data de início, já que não é possível um leilão terminar antes da data inicial. Da mesma forma, se a data final for inferior à data inicial, o sistema mostra uma mensagem de erro.

Depois destas verificações, o sistema atribui um identificador único ao leilão e guarda-o em estado pendente.

### **Cancelamento de um leilão (RF05)**

O criador de um leilão pode cancelá-lo, desde que o faça com o mínimo de uma hora de antecedência do início do leilão em questão.

O sistema começa por verificar o tempo que falta para o início do leilão. Se faltar mais de uma hora, todos os dados referentes ao leilão são imediatamente eliminados. Caso contrário, o leilão deve prosseguir e o utilizador deve receber uma mensagem de erro.



### **Consulta de perfil (RF06)**

O utilizador pode consultar o seu perfil, constituído por:

1. Informações pessoais (nome completo, username, email, password e NIF);
2. Leilões que tenha criado e que estejam a decorrer;
3. Leilões que tenha criado, mas que já não estejam a decorrer;

Cada utilizador apenas pode verificar o seu próprio perfil.

### **Consulta da página inicial (RF07)**

O utilizador deve poder consultar a página inicial, na qual o sistema deve mostrar uma lista de leilões. Cada leilão deve mostrar o nome do item e o valor da licitação atual.

Para além disso, deve ser possível o utilizador aceder a um leilão em específico de modo a obter mais informações sobre o mesmo.

### **Licitação de um item (RF08)**

O utilizador deve poder licitar um item. Para isso, o sistema pede um valor e depois realiza várias verificações:

1. O valor deve ser válido.
2. O valor deve ser superior ao valor mínimo.
3. O valor deve ser superior ao maior valor licitado até à altura.
4. O utilizador a licitar não é o criador do leilão.

Se algum destes critérios não for cumprido, o sistema deve interromper a licitação e apresentar um mensagem de erro ao utilizador.

Caso contrário, o sistema verifica se o utilizador tem saldo suficiente e retira-o.

Finalmente, a licitação é registada no sistema e deve estar associada ao utilizador que a realizou. Quando a licitação for registada com sucesso, esta pode aparecer publicamente para os restantes utilizadores.

### **Consulta de um leilão (RF10)**

Na página inicial são apresentados os leilões apenas com o nome do item e o valor da licitação atual. Desta forma, é fornecida uma forma de redirecionar o utilizador para uma página com mais informação sobre um determinado leilão. A informação que deve ser possível consultar é a seguinte:

1. Nome do item;
2. Descrição do item;
3. Imagem do item;
4. Número de donos anteriores;
5. Valor inicial da licitação;
6. Licitação atual;
7. Data de início;
8. Data final;

### **Consulta de estatísticas (RF11)**

O utilizador deve poder consultar algumas estatísticas sobre as suas vendas:

1. As 10 últimas vendas que realizou;
2. As 10 vendas mais caras que realizou;
3. Número total de vendas;
4. Total de dinheiro em vendas;

As vendas mais caras que realizou devem ser apresentadas numa lista ordenada por preço de venda e para cada item deve ser apresentado o seu nome e preço pelo qual foi vendido.

### **Adicionar saldo à conta (RF12)**

O utilizador deve poder adicionar saldo à sua conta, de forma a poder utilizá-lo mais tarde em leilões.

Para isso, o sistema começa por pedir a quantia a adicionar. Depois de verificar se os dados estão corretos, o sistema espera que o dinheiro seja corretamente transferido, para apenas depois adicionar ao saldo.

Quando o dinheiro é adicionado ao saldo, este torna-se imediatamente utilizável.

Se os dados não estiverem corretos ou se houver algum problema com a transferência, o sistema deve avisar.

## **3.3.3 Use cases do administrador**

### **Aprovação de um leilão (RF04)**

Antes de um leilão ser visível para todos os utilizadores, o administrador tem de o aprovar.

Caso seja aprovado, o sistema deve registar a aprovação e mostrar o leilão para os utilizadores.

Caso não seja aprovado, o sistema deve eliminar todos os dados referentes ao leilão e enviar uma mensagem ao criador a dizer que o leilão foi rejeitado.

### **3.3.4 Diagramas de atividades**

Os diagramas de atividades são uma ferramenta essencial na compreensão do funcionamento da aplicação, uma vez que fornecem uma perspectiva visual sobre o que um certo utilizador pode e não pode fazer.

No caso do nosso projeto, necessitamos apenas de um diagrama de atividades do utilizador regular e de um diagrama dedicado ao administrador, que apresentamos de seguida:

## Diagrama de atividades do utilizador regular

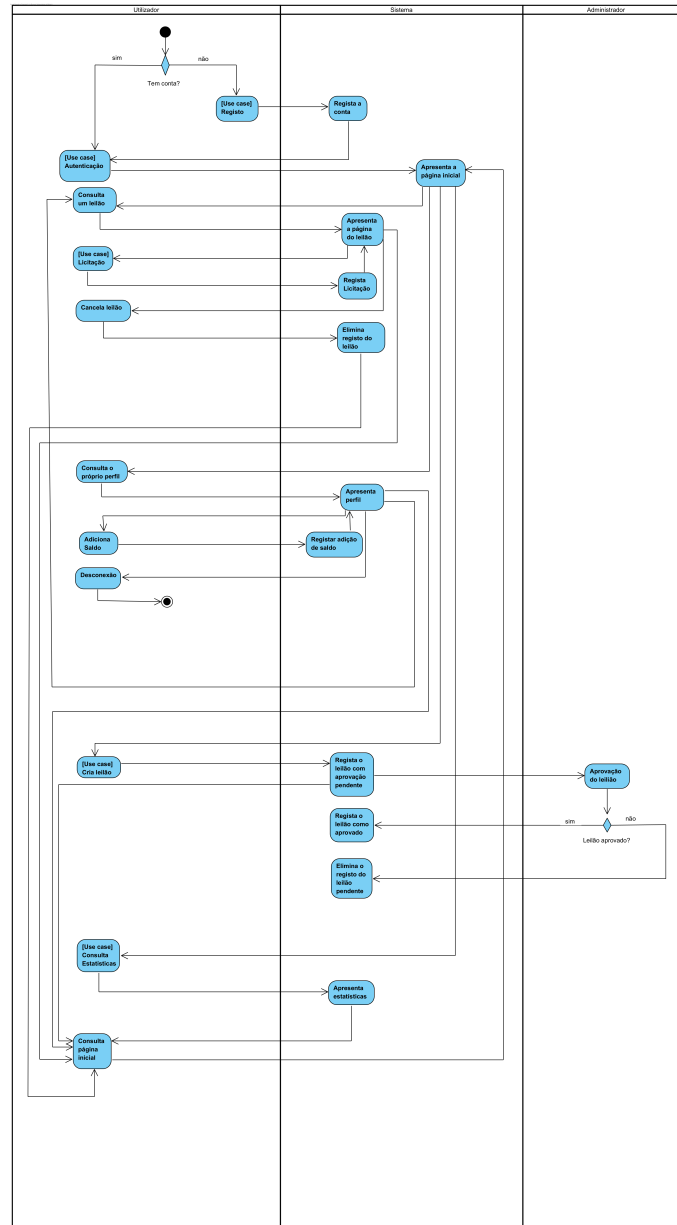


Figura 3.3: Diagrama de atividades do utilizador regular

Este diagrama representa a utilização da aplicação para um utilizador regular. Quando entra na aplicação, escolhe a opção de autenticação ou registo, conforme tenha ou não conta, e, quando concluir essa parte, será direcionado à página inicial.

Como será abordado posteriormente, uma parte fundamental da aplicação passa por ter a página inicial a funcionar como um *hub*, ou seja, a partir desta página, o utilizador pode escolher as várias funcionalidades que quer executar (consultar um leilão, visitar o seu perfil, etc) e, acedendo a essas funcionalidades, terá sempre a opção de se dirigir à página inicial.

## Diagrama de atividades do administrador

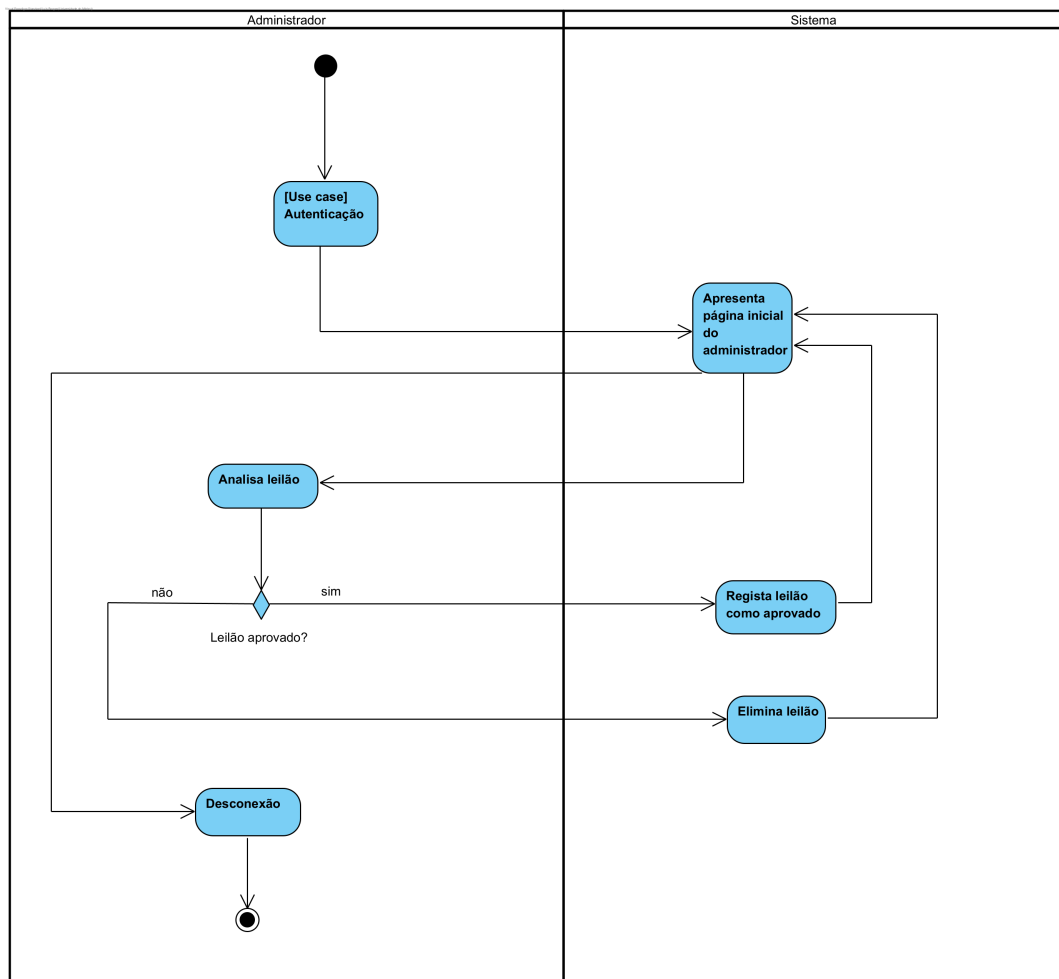


Figura 3.4: Diagrama de atividades do administrador

O diagrama de atividades do administrador é extremamente simples, uma vez que a sua ação na aplicação é relativamente reduzida.

O administrador passa pela autenticação e a sua única função consiste em analisar os leilões que se encontram em estado pendente e decidir se deve aprová-los ou não.

## 4 Conceção do Sistema de Dados

### 4.1 Apresentação geral da estrutura do sistema de dados

Após a reunião dos requisitos e a sua organização em use cases, o passo seguinte passa pela criação de um modelo lógico, para auxiliar a compreensão das entidades e relacionamentos.

Segue-se o modelo lógico do nosso projeto:

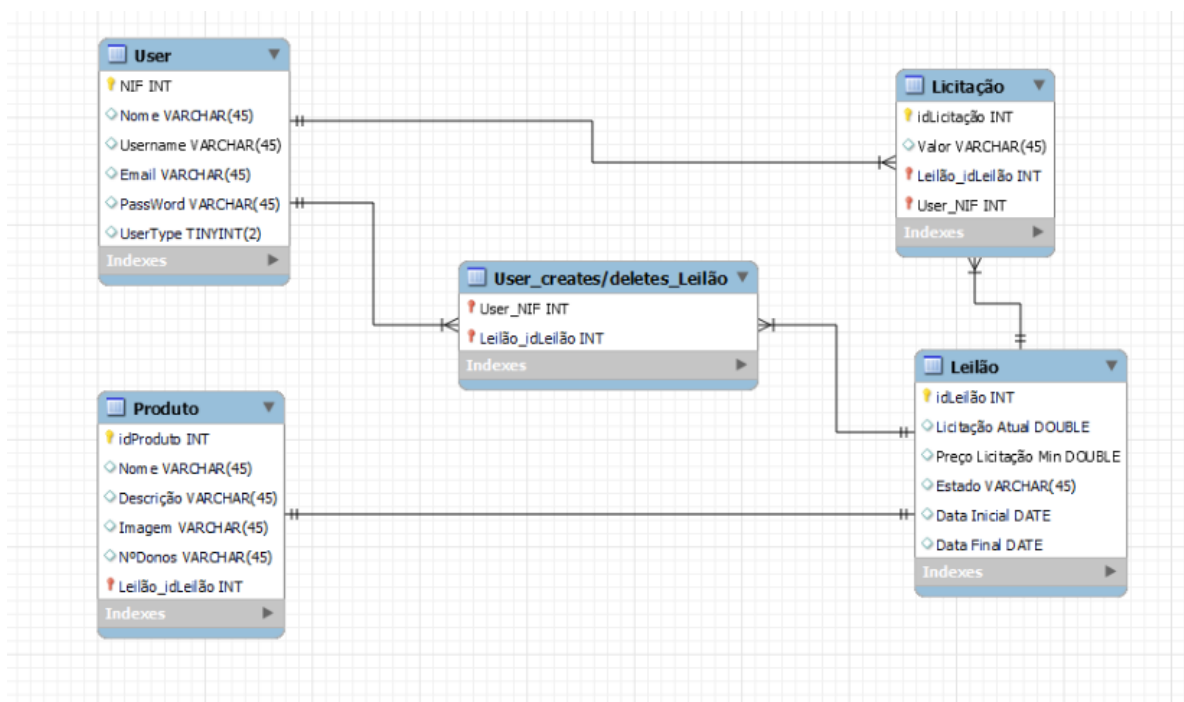


Figura 4.1: Modelo lógico

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

### 4.2.1 Utilizador

Um utilizador é identificado pelo seu NIF.

Os restantes campos são atributos simples do utilizador.

Atributo	Tipo de dados	Descrição	Exemplo
NIF	Int	NIF do utilizador	123456789
Nome	Varchar(45)	Nome do utilizador	João Mário
Username	Varchar(45)	Username para login do utilizador	joao1234mario
Email	Varchar(45)	Email do utilizador	joaom@gmail.com
Password	Varchar(45)	Password do utilizador	joao1221
UserType	TinyInt	Tipo de utilizador (participante ou administrador)	participante

Tabela 4.1: Entidade Utilizador



## 4.2.2 Produto

Um produto é identificado por um id que a base de dados gera automaticamente. A imagem é um string com um caminho que aponta para o local onde a imagem está guardada.

Os restantes campos são atributos simples do produto.

Atributo	Tipo de dados	Descrição	Exemplo
idProduto	Int	Identificador único do produto	1
Nome	Varchar(45)	Nome do produto	Vestido
Descrição	Varchar(45)	Descrição do produto	Vestido azul muito pouco usado
Imagem	Varchar(45)	Imagem do produto	joaom@gmail.com
Nºdonos	Varchar(45)	Número de donos anteriores	1

Tabela 4.2: Entidade Produto

### 4.2.3 Licitação

Uma licitação é identificada por um id que a base de dados gera automaticamente. Para além disso, cada licitação tem um atributo "Leilão\_idLeilão" que guarda o id do leilão onde a licitação foi realizada, e um atributo "User\_NIF" para saber qual foi o utilizador que fez a licitação.

Os restantes campos são atributos simples da licitação.

Atributo	Tipo de dados	Descrição	Exemplo
idLicitação	Int	Identificador único da licitação	1
Valor	Varchar(45)	Valor da licitação	32,99
Leilão_idLeilão	Int	Identificador único do leilão onde foi feita a licitação	4
User_NIF	Int	NIF do utilizador que efetuou a licitação	123456789

Tabela 4.3: Entidade Licitação

#### 4.2.4 Leilão

Um leilão é identificado por um id que a base de dados gera automaticamente.

Os restantes campos são atributos simples do leilão.

Atributo	Tipo de dados	Descrição	Exemplo
idLeilão	Int	Identificador único do leilão	4
Licitação Atual	Double	Valor da licitação mais alta	32,99
Preço Licitação Min	Double	Valor mínimo a licitar	10,05
Estado	Varchar(45)	Estado do leilão (ativo, pendente ou terminado)	Ativo
Data Inicial	Date	Data em que o leilão começou ou vai começar	22/09/2023
Data Final	Date	Data em que o leilão acabou ou vai acabar	29/09/2023

Tabela 4.4: Entidade Leilão

#### 4.2.5 Relação 1:1 *Leilão\_has\_Produto*

Um produto está associado a um único leilão.

Um leilão tem exatamente um produto associado.

Entidade	Atributo	Tipo de dados	Descrição	Exemplo
Leilão_has_Produto	Leilão_idLeilão	INT	Id do leilão	1
	Produto_idProduto	INT	Id do produto	7

Tabela 4.5: Relação *Leilão\_has\_Produto*

#### 4.2.6 Relação 1:N *User\_creates/deletes\_Leilão*

Um utilizador pode criar/cancelar vários leilões.

Um leilão pode ser criado ou cancelado apenas por um utilizador.

Entidade	Atributo	Tipo de dados	Descrição	Exemplo
User_creates/deletes_Leilão	Leilão_idLeilão	INT	Id do leilão	1
	User_NIF	INT	Id do <i>user</i>	123 456 789

Tabela 4.6: Relação *User\_creates/deletes\_Leilão*

# 5 Esboço das Interfaces do Sistema

## 5.1 Estrutura Geral das Interfaces do Sistema

O esquema apresentado abaixo cumpre a função de auxiliar a compreensão do funcionamento da aplicação. É relevante referir que a estratégia passar por ter a página inicial a funcionar quase como um *hub*, ou seja, dar a possibilidade ao utilizador de, a qualquer momento, aceder à página inicial.

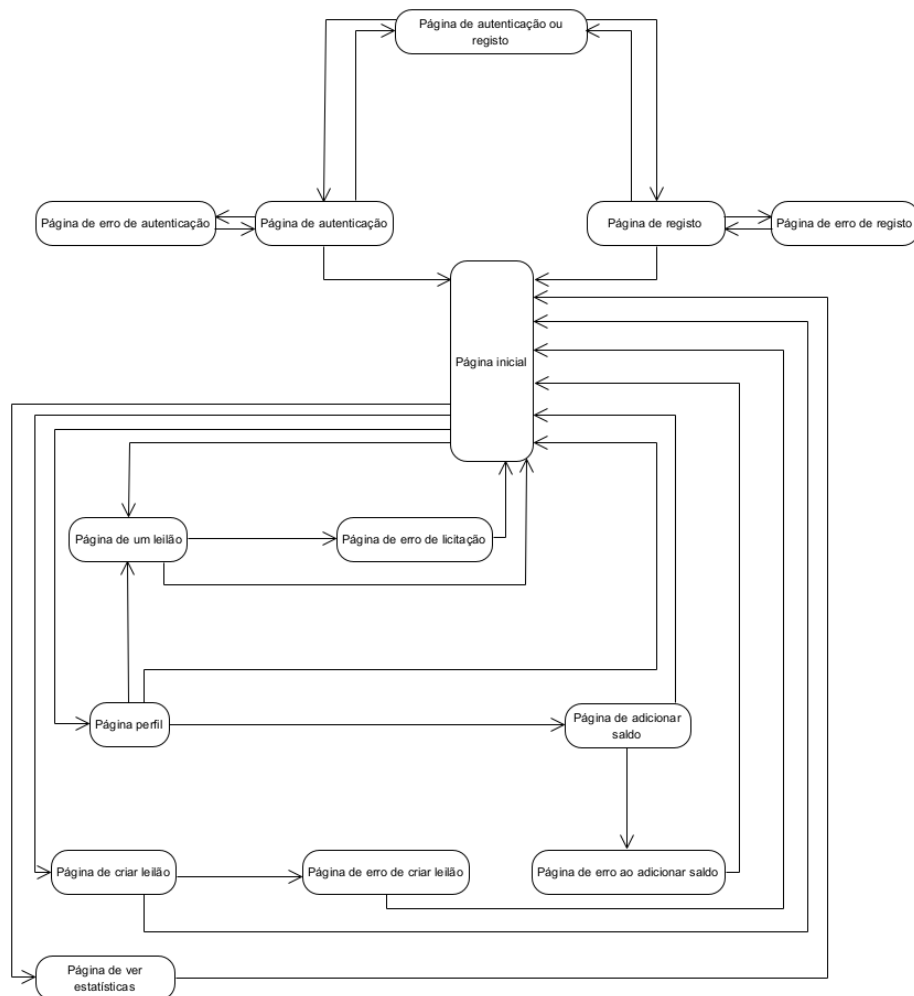


Figura 5.1: Estrutura das Interfaces do Sistema

## 5.2 Caraterização das Interfaces

Seguem-se os *mockups* das interfaces que consideramos relevantes.

### 5.2.1 Página Inicial

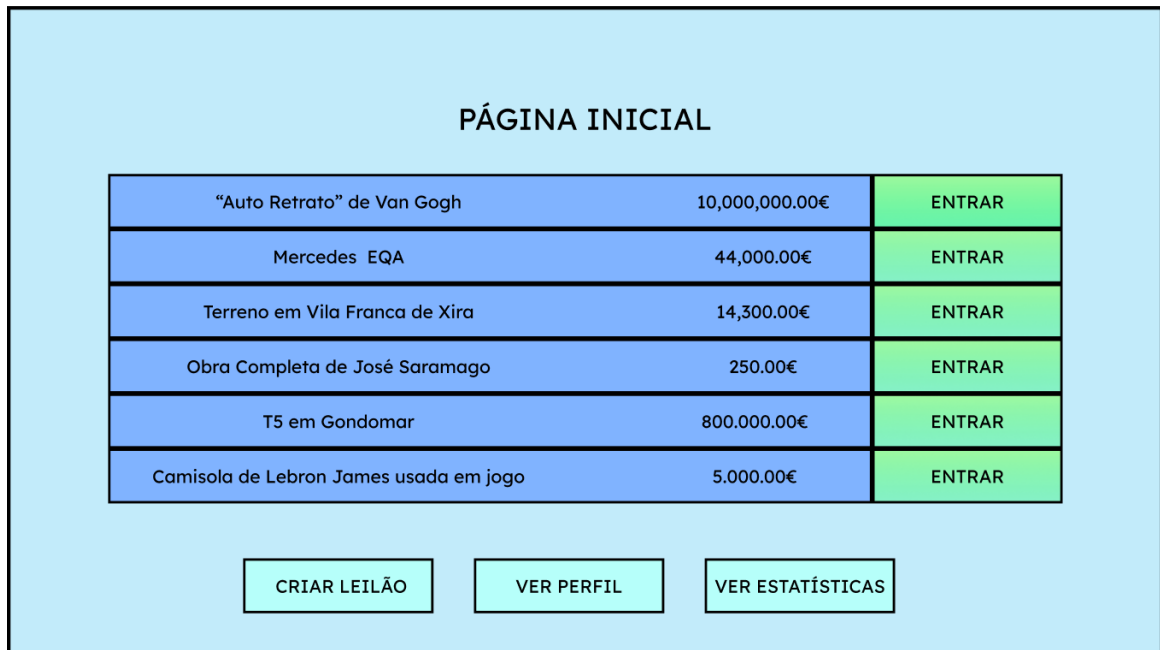


Figura 5.2: Página Inicial

A página inicial serve como *hub* para toda a aplicação. Na sua construção, tentamos priorizar a simplicidade.

Nesta página, o utilizador vê uma lista de leilões e três opções adicionais. O utilizador pode entrar num dos leilões, criar um leilão, entrar no próprio perfil ou aceder à página de estatísticas.

## 5.2.2 Página de Criar Leilão





### CRIAR LEILÃO

NOME DO OBJETO:

DESCRIÇÃO:

NR DE DONOS ANTERIORES:

LICITAÇÃO MÍNIMA:

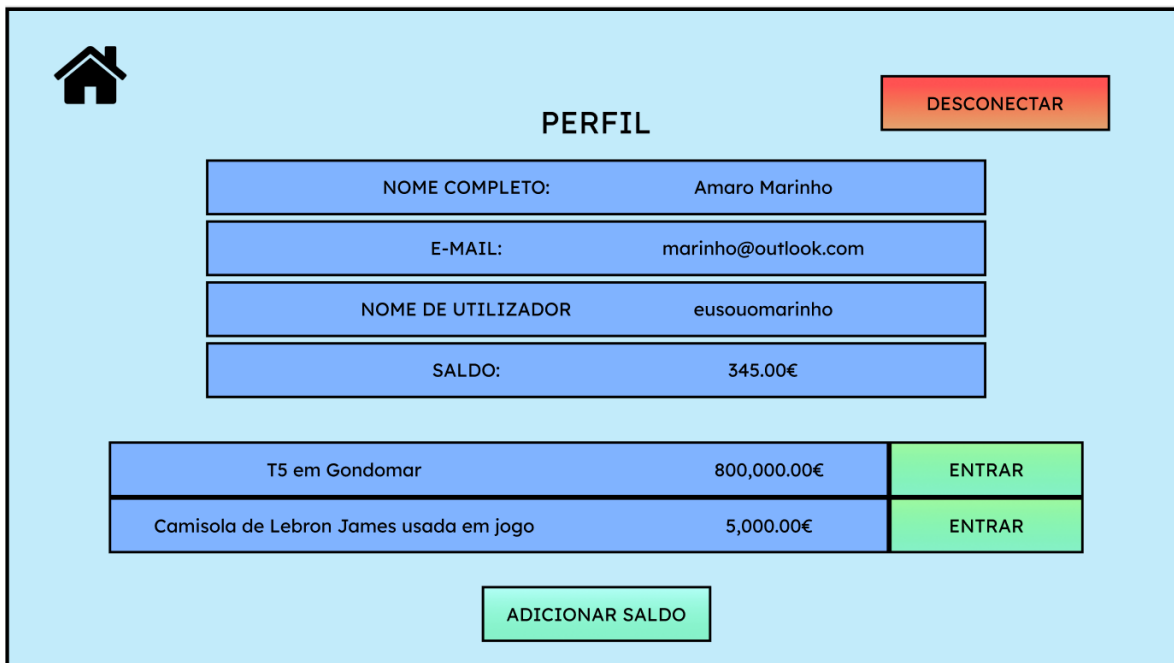
DATA INICIAL / DATA FINAL

IMAGEM

Figura 5.3: Página de Criar Leilão

Na página dedicada à criação de um leilão, que pode ser acedida através da página inicial, existem apenas os vários campos que devem ser preenchidos e o botão para fazer *upload* da imagem correspondente ao item. Por fim, existe também o botão para proceder à criação do leilão, quando todos os campos estiverem preenchidos. No canto superior esquerdo está uma imagem de uma casa que permite ir diretamente para a página inicial.

### 5.2.3 Página de Perfil



The image shows a user profile page with a light blue background. In the top left corner is a black house icon. In the top right corner is a red button labeled 'DESCONECTAR'. The title 'PERFIL' is centered at the top. Below the title are four blue boxes, each containing a label and a value: 'NOME COMPLETO: Amaro Marinho', 'E-MAIL: marinho@outlook.com', 'NOME DE UTILIZADOR eusouomarinho', and 'SALDO: 345.00€'. Below these are two rows of items, each with a description, a value, and a green 'ENTRAR' button. The first row is 'T5 em Gondomar' with a value of '800,000.00€'. The second row is 'Camisola de Lebron James usada em jogo' with a value of '5,000.00€'. At the bottom center is a green button labeled 'ADICIONAR SALDO'.

PERFIL		
NOME COMPLETO:	Amaro Marinho	
E-MAIL:	marinho@outlook.com	
NOME DE UTILIZADOR	eusouomarinho	
SALDO:	345.00€	
T5 em Gondomar	800,000.00€	ENTRAR
Camisola de Lebron James usada em jogo	5,000.00€	ENTRAR
ADICIONAR SALDO		

Figura 5.4: Página de Perfil


A página de perfil é quase apenas ilustrativa. Nela, estão expostas as informações relevantes do utilizador. Para além disso, é possível entrar nas páginas dos próprios leilões, estando eles a decorrer ou não.

Por fim, o utilizador pode também adicionar saldo à sua conta.

Nesta página encontram-se também a ligação à página inicial através do símbolo da casa, no canto superior esquerdo, e a opção de desconexão, no canto superior direito




## 5.2.4 Página de Leilão



### LEILÃO

ITEM	T5 em Gondomar
DESCRIÇÃO	T5 novo em Baguim do Monte, Gondomar, com 3 casas de banho e cozinha mobilada
NR DE UTILIZADORES ANTERIORES	0
LICITAÇÃO INICIAL	750,000.00€
VALOR ATUAL	800,000.00€
DATA DE INÍCIO	30/10/2023
DATA FINAL	29/01/2024



\*VALOR\*

LICITAR

Figura 5.5: Página de Leilão

Na página de cada leilão, podem ser vistas as informações sobre o próprio leilão. Para além disso, é dada ao utilizador a possibilidade de licitar, mediante a introdução de um valor.

## 5.2.5 Página de Estatísticas



Figura 5.6: Página de Estatísticas

A página da estatísticas permite ao utilizador ver algumas estatísticas relevantes sobre a aplicação.

### 5.2.6 Página de Autenticação

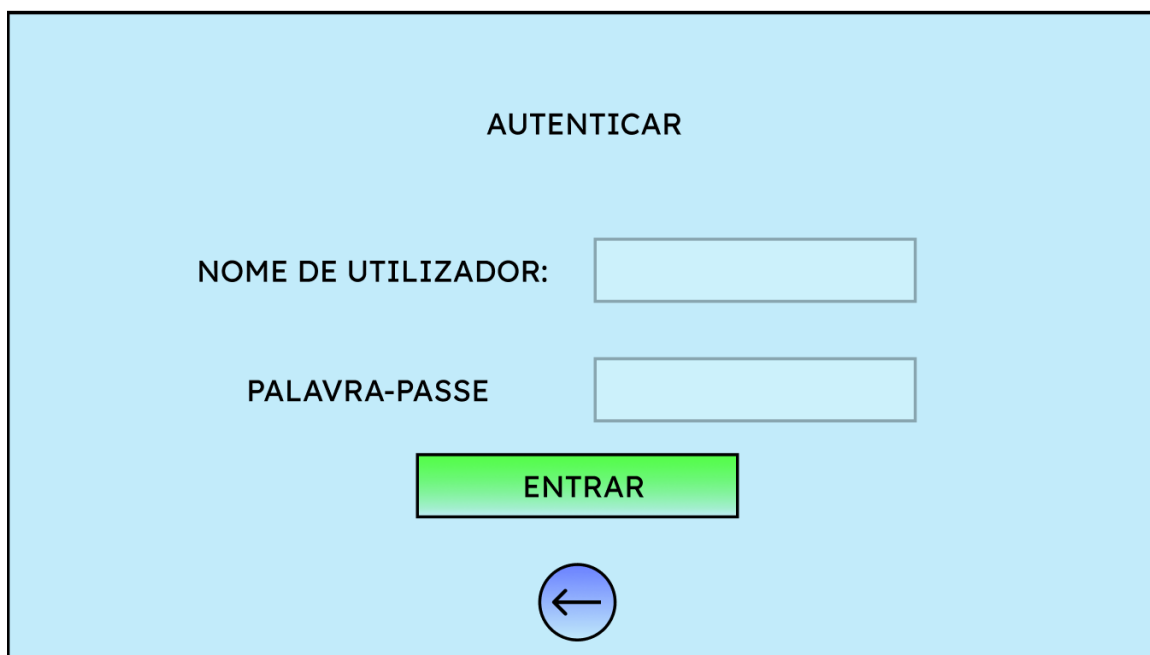


Diagrama da página de autenticação. O fundo é azul claro. No topo, centralizado, está o título "AUTENTICAR". Abaixo dele, há dois campos de entrada: "NOME DE UTILIZADOR:" seguido de um retângulo branco, e "PALAVRA-PASSE" seguido de um retângulo branco. Abaixo dos campos, há um botão verde com o texto "ENTRAR". No centro da página, abaixo do botão, há um círculo azul com uma seta branca apontando para a esquerda.

Figura 5.7: Página de Autenticação

Finalmente, a página de autenticação permite a um utilizador que possui uma conta aceder à aplicação.

## 6 Implementação

No desenvolvimento da nossa aplicação, começamos pela implementação da base de dados, permitindo que os dados fossem guardados corretamente, garantindo, assim, que o *back-end* tivesse acesso aos mesmos, através da utilização da API criada.

Utilizamos o *framework ASP .NET MVC* para o *back-end*, onde criamos *Controllers* e *Models* para manusear as entidades.

Para o *front-end*, utilizamos o *framework Blazor Server*, onde construímos páginas e *view models*.

Utilizamos também uma base de dados em *SQL* para guardar a informação necessária.

### 6.1 Base de dados

O código de implementação da base de dados é simples. Começamos por criar a própria base de dados e, de seguida, cada uma das quatro tabelas. Usamos a primitiva *INSERT* para as preencher e, por fim, usamos a primitiva *SELECT* para verificar os resultados.

### 6.2 Back-end

Para fazer o *back-end*, começamos por criar classes correspondentes a cada entidade, que se encontram em *leiloes/models*. Nesta pasta, existem também *ViewModels* necessários para a implementação de alguns métodos.

No entanto, a parte mais importante está concentrada em *leiloes/controllers*. Nesta pasta, foi criado um *controller* para cada entidade, cuja função passa por analisar as bases de dados e enviar informação para o *front-end*.

Para além dos *controllers* dedicados às entidades, existe também um dedicado às estatísticas e um dedicado à página inicial. Segue-se uma breve descrição de cada um.

### **6.2.1 *HomeController***

*HomeController* é o módulo que serve a página inicial. Tem um método que devolve ao *front-end*, para cada leilão ativo, o seu identificador, o nome do item a ser licitado e o valor da última licitação, usado na página inicial dos utilizadores regulares, e outro cuja função é a mesma, porém, para leilões em estado pendente, sendo este último utilizado na página inicial dos administradores.

### **6.2.2 *LeilaoController***

*LeilaoController* coordena as atividades relacionadas com os leilões. Permite obter uma lista dos leilões, obter um leilão a partir do seu Id e criar, aprovar e eliminar um leilão da base de dados. Permite, também, obter todos os leilões criados por um certo utilizador, através do seu NIF.

### **6.2.3 *LicitacaoController***

*LicitacoesController* permite manusear dados relativos a licitações. Mais concretamente, estão escritos métodos para criar uma licitação e consultar uma lista com todas as licitações existentes.

### **6.2.4 *ProdutoController***

*ProdutoController* torna possível criar ou eliminar um produto, obter uma lista com todos os produtos ou obter um produto através do seu identificador.

### **6.2.5 *StatsController***

Por fim, o *StatsController* possui apenas um método que devolve, para um utilizador indentificado a partir do seu nif, as suas estatísticas.

### **6.2.6 *UtilizadorController***

*UtilizadorController* é, destes, certamente o mais complexo. Permite obter uma lista com todos os utilizadores existentes, obter os dados de um utilizador através do seu nif e fazer o registo ou autenticação de uma conta. Tem também métodos que facilitam a consulta do perfil do utilizador ou a adição de saldo à conta.

É, aqui, relevante mencionar que, no processo de autenticação, é gerada uma *JWT Token* que permite, a qualquer ponto, saber quem está a utilizar a aplicação. Na *token*, é guardado o nif, o email, o tipo de utilizador e o nome de utilizador. Estes atributos são essenciais ao longo do processo uma vez que permitem que, por exemplo, quando o utilizador escolhe a opção de consultar o seu perfil, o programa saiba exatamente que perfil mostrar.

### 6.2.7 Outros ficheiros relevantes

Para além dos ficheiros já mencionados, é também importante mencionar o ficheiro *Program.cs*, onde estão codificadas partes essenciais do programa. Juntamente com todas as configurações associadas à *JWT Token*, são relevantes os métodos *CheckLeiloes* e *VerificarETerminarLeiloes* que, em conjunto, permitem que os leilões terminem no *background*. Para qualquer leilão ativo cuja data limite tenha sido ultrapassada, verifica-se se não existiu nenhuma licitação nos últimos 5 minutos e, em caso positivo, o leilão é dado como terminado. Este processo corre no *background* de segundo em segundo, garantindo assim que os leilões terminam corretamente.

## 6.3 Front-end

O *back-end* da aplicação consiste num projeto *Blazor Server* que tira partido dos ficheiros na pasta *Pages* para gerar uma interface para os utilizadores. Os ficheiros desta página utilizam os *controllers* do *back-end* para obter ou processar informação, apresentando a necessária numa estrutura visualmente agradável, graças à utilização de HTML e CSS. Em geral, foi criado um ficheiro *.razor* para cada página existente na estrutura das interfaces do sistema (Figura 5.1), mas esta ideia não é cumprida à risca.

## 7 Conclusões e Trabalho Futuro

Para o futuro, de forma a manter a plataforma interessante e abranger novos públicos, podem ser implementadas novas funcionalidades, tais como:

- Recomendações personalizadas de leilões de acordo com o histórico e perfil do utilizador.
- Novos métodos de pagamento.
- Adaptação para diferentes idiomas.

Para além disso, se a aplicação tiver um aumento no número de utilizadores será necessário garantir que a base de dados se mantém segura e eficiente, podendo ser mais viável explorar outras estratégias como a utilização de serviços na nuvem, que podem oferecer mais flexibilidade e recursos.

# Referências

EconPort: History of Auctions (s.d.). Disponível em: <https://www.econport.org/content/handbook/auctions/historyofauctions.html> [Visitado a 25/09/2023].

Internet World Stats: Internet Growth Statistics (s.d.). Disponível em: <https://www.internetworldstats.com/emarketing.htm> [Visitado a 02/10/2023].

Semrush: Visão do domínio olx.pt (s.d.) Disponível em: <https://pt.semrush.com/analytics/overview/?q=olx.pt&protocol=https&searchType=domain&db=pt> [Visitado a 02/10/2023]

Semrush: Visão do domínio continente.pt (s.d.) Disponível em: <https://pt.semrush.com/analytics/overview/?q=continente.pt&protocol=https&searchType=domain&db=pt> [Visitado a 02/10/2023]

Semrush: Visão do domínio e-leiloes.pt (s.d.) Disponível em: <https://pt.semrush.com/analytics/overview/?q=e-leiloes.pt&protocol=https&searchType=domain> [Visitado a 02/10/2023]



# Anexos

## Instruções *SQL* para a criação das tabelas da base de dados

```
USE leiloes ;
```

```
GO
```

```
CREATE TABLE Produto (  
    idProduto INT PRIMARY KEY IDENTITY(1,1),  
    nome VARCHAR(45) NOT NULL,  
    descricao VARCHAR(255) NOT NULL,  
    imagem VARCHAR(255),  
    numDonosAnt INT NOT NULL DEFAULT 0
```

```
);
```

```
GO
```

```
CREATE TABLE Utilizador (  
    nif VARCHAR(9) PRIMARY KEY,  
    nome VARCHAR(45) NOT NULL,  
    username VARCHAR(45) NOT NULL,  
    email VARCHAR(45) NOT NULL,  
    password VARCHAR(45) NOT NULL,  
    userType INT NOT NULL,  
    saldo DECIMAL(10, 2) NOT NULL
```

```
);
```

```
GO
```

— Criar a tabela Leilao

```
CREATE TABLE Leilao (  
    idLeilao INT PRIMARY KEY IDENTITY(1,1),  
    licitacaoAtual DECIMAL(10, 2) NOT NULL,  
    precoMinLicitacao DECIMAL(10, 2) NOT NULL,  
    estado NVARCHAR(50) NOT NULL,  
    dataInicial DATETIME NOT NULL,  
    dataFinal DATETIME NOT NULL,  
    criadorId VARCHAR(9),  
    produtoId INT,  
    FOREIGN KEY (criadorId) REFERENCES Utilizador(nif),  
    FOREIGN KEY (produtoId) REFERENCES Produto(idProduto)
```

```
);  
GO
```

```
CREATE TABLE Licitacao (  
    idLicitacao INT PRIMARY KEY IDENTITY(1,1),  
    valor DECIMAL(10, 2) NOT NULL,  
    leilao_idLeilao INT NOT NULL,  
    user_NIF VARCHAR(9) ,  
    dataLicitacao DATETIME NOT NULL,  
    FOREIGN KEY (leilao_idLeilao) REFERENCES Leilao(idLeilao),  
    FOREIGN KEY (user_NIF) REFERENCES Utilizador(nif)  
);
```