



Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia informática
Mestrado Integrado em Engenharia Informática

Unidade Curricular de Laboratórios de Informática IV

Ano Letivo de 2023/2024

BetterFinds

Flávia Alexandra da Silva Araújo (a96587)

Miguel Torres Carvalho (a95485)

Janeiro, 2024

LI4

Data de Recepção	
Responsável	
Avaliação	
Observações	

BetterFinds

Flávia Alexandra da Silva Araújo (a96587)

Miguel Torres Carvalho (a95485)

Janeiro, 2024

Resumo

No seguimento dos métodos de avaliação da UC Laboratórios de Informática IV, lecionada pelo regente, Professor Orlando Belo, visamos a realização de um projeto que consiste na implementação de um *software de Leilões Online*.

Respetivo ao tema dos leilões, faltava-nos ainda decidir o subtema respetivo ao projeto. Deste modo, como equipa de trabalho decidimos abordar, por uma questão de gosto, os leilões de antiguidades.

Agora, relativamente à execução e desenvolvimento desta aplicação, iremos dividir o projeto em três partes que consistem em: definição e fundamentação, isto é, definir o contexto da aplicação, na especificação e na construção.

Dado que a última fase envolve a apresentação da arquitetura final, iremos recorrer a plataformas como *Microsoft Visual Studio, Microsoft C#, Microsoft .NET, Microsoft SQL Server e Visual Paradigm*.

Tal como revelamos mais adiante, todas estas fases foram sujeitas a métodos de validação das mesmas. De modo a concretizar este trabalho iremos seguir a metodologia abordada durante os seminários e aulas práticas.

Área de Aplicação: Leilões Online

Palavras-Chave: Engenharia de Software, Bases de Dados, UML, Fundamentação, Especificação, Construção, Levantamento de Requisitos, Validação, Visual Paradigm.

Índice

1.	Introdução	1
1.1.	Contextualização	1
1.2.	Apresentação do Caso de Estudo	1
1.3.	Fundamentação	1
1.4.	Motivação e Objectivos	2
1.5.	Estrutura do Relatório	3
2.	Viabilidade	4
3.	Plano de Desenvolvimento	5
3.1.	Plano de Execução de Trabalho	5
3.2.	Equipa de Trabalho	6
3.3.	Recursos a Utilizar	6
4.	Levantamento de Requisitos	9
4.1.	Apresentação da Estratégia e Método	9
4.2.	Descrição Geral dos Requisitos Levantados	10
4.2.1.	Requisitos Funcionais	10
4.2.1.1.	Gestão de Utilizador	10
4.2.1.2.	Procurar Produto	10
4.2.1.3.	Criar Leilão	11
4.2.1.4.	Finalizar Leilão	11
4.2.1.5.	Participar num Leilão	11
4.2.1.6.	Finalizar Compra	12
4.2.2.	Requisitos Não Funcionais	13
5.	Especificação e Modelação do Software	14
5.1.	Apresentação Geral Da Especificação	14
5.2.	Modelo de Domínio	14
5.3.	Diagrama de <i>Use Cases</i>	16
5.4.	Especificação de <i>Use Cases</i>	17
5.4.1.	Registrar	17
5.4.2.	Efetuar <i>Login</i>	18

5.4.4. Pesquisar Produto	19
5.4.5. Aceder à <i>Homepage</i>	19
5.4.6. Editar Perfil	20
5.4.7. Ver Notificações	20
5.4.8. Criar Leilão	21
5.4.9. Participar num Leilão	21
5.4.10. Finalizar Compra	22
5.4.11. Finalizar Leilão	23
5.4.12. Visualizar A Minha Conta	23
5.4.13. Visualizar As Minhas Licitações	24
5.4.14. Visualizar Os Meus Leilões	24
5.4.15. Terminar Sessão	25
5.5. Aspetos Comportamentais	26
6. Conceção do Sistema de Dados	30
6.1. Apresentação Geral da Estrutura do Sistema de Dados	30
6.2. Descrição Detalhada dos Elementos de Dados e Relacionamentos	31
6.2.1. Entidade "Auction"	31
6.2.2. Entidade "Client"	31
6.2.3. Entidade "Notification"	32
6.2.4. Entidade "Product"	32
6.2.5. Entidade "Bid"	32
6.2.6. Relacionamentos	33
7. Esboço das Interfaces do Sistema	35
7.1. Estrutura Geral das Interfaces do Sistema	35
7.2. Proposta de Interfaces	36
8. Metodologia e Ferramentas de Implementação	45
8.1. <i>Backend</i>	45
8.2. <i>Frontend</i>	46
8.3. Servidor	46
8.4. Testagem e Segurança	46
9. Conexão da Base de Dados	47
10. Desenvolvimento	49
10.1. Gestão de <i>Logins</i>	49
10.2. Diretoria <i>Utils</i>	49
10.2.1. Leilões (<i>Auctions.cs</i>)	49
10.2.2. Licitações (<i>Bids.cs</i>)	50
10.2.3. Cliente (<i>Client.cs</i>)	50
10.2.4. Notificações (<i>Notification.cs</i>)	50

10.2.5. Imagens (<i>Images.cs</i>)	50
10.2.5. Valores Monetários (<i>Currency.cs</i>)	50
11. Segurança e Manutenção	51
12. Testes	54
13. Arquitetura Final do Projeto	56
14. Produto Final	58
14.1. Registar	58
14.2. <i>Login</i>	59
14.3. <i>Logout</i>	60
14.4. <i>Homepage</i>	61
14.5. <i>My Account</i>	62
14.6. Perfil Público	64
14.7. Pesquisa	65
14.8. <i>My Bids</i>	66
14.9. <i>My Auctions</i>	67
14.10. <i>Create (Auction)</i>	68
14.11. Página de Leilão (<i>Auction</i>)	69
14.12. Página de Notificações	74
14.13. Exemplos de Produto Final em <i>Mobile</i>	77
15. Conclusões e Trabalho Futuro	78
15.1. Trabalho Futuro	78
15.2. Conclusão	78
Referências	79
Lista de Siglas e Acrónimos	80

Índice de Figuras

Figura 1: Diagrama de GANTT.....	5
Figura 2: Miguel Torres Carvalho (A95485).....	6
Figura 3: Flávia Alexandra Silva Araújo (A96587).....	6
Figura 4: Diagrama de Classes.....	15
Figura 5: Diagrama de Use Cases.....	16
Figura 6: Diagrama de Atividade do Vendedor.....	26
Figura 7: Diagrama de Atividade do Comprador.....	28
Figura 8: Modelo Lógico do Sistema.....	30
Figura 9: Planeamento da Estrutura das Interfaces do Sistema.....	35
Figura 10: Mockup - Página de Login.....	36
Figura 11: Mockup - Página de Pesquisa.....	37
Figura 12: Mockup - Página de Registo do Utilizador.....	38
Figura 13: Mockup - Página Pessoal do Utilizador.....	39
Figura 14: Mockup - Centro de Notificações.....	40
Figura 15: Mockup - Página Perfil Público.....	41
Figura 16: Mockup - Página de Leilão Ativo.....	42
Figura 17: Mockup - Página de finalização de compra.....	43
Figura 18: Mockup - Página de Criação de Leilão.....	44
Figura 19: Tabela Entidade "Auction".....	47
Figura 20: Tabela Entidade "Bid".....	47
Figura 21: Tabela Entidade "Client"	48
Figura 22: Tabela Entidade "Notification".....	48
Figura 23: Tabela Entidade "Product".....	48
Figura 24: Análise do código no Snyk.....	52
Figura 25: Análise do código no DeepSource.....	52
Figura 26: Arquitetura Final do Projeto.....	56
Figura 27: Página de Registo.....	58
Figura 28: Página de Login.....	59
Figura 29: Sidebar com botão Logout.....	60

Figura 30: Redirecionamento após <i>Logout</i>	61
Figura 31: <i>Homepage</i>	62
Figura 32: <i>My Account</i>	63
Figura 33: Editar Conta.....	63
Figura 34: Perfil Público.....	64
Figura 35: Página de Pesquisa.....	65
Figura 36: <i>My Bids</i>	66
Figura 37: <i>My Auctions</i>	67
Figura 38: <i>Create (Auction)</i>	68
Figura 39: Página de Leilão (Perspetiva do Comprador).....	70
Figura 40: Página de Leilão (Perspetiva do Vendedor).....	72
Figura 41: Página de Leilão - Opção de Pagamento.....	73
Figura 42: Página de Notificações.....	74
Figura 43: "Show All" Notifications.....	75
Figura 44: "You have no unread notifications".....	76
Figura 45: Página de Leilão Mobile.....	77
Figura 46: <i>Homepage Mobile</i>	77
Figura 47: <i>Login Mobile</i>	77
Figura 48: <i>My Account Mobile</i>	77

Índice de Tabelas

Tabela 1: Use Cases - Registar.....	16
Tabela 2: Use Cases - Login.....	17
Tabela 3: Use Cases - Ver Perfil Público.....	17
Tabela 4: Use Cases - Pesquisar Produto.....	18
Tabela 5: Use Cases - Aceder à Homepage.....	18
Tabela 6: Use Cases - Editar Perfil.....	19
Tabela 7: Use Cases - Ver Notificações.....	19
Tabela 8: Use Cases - Criar Leilão.....	20
Tabela 9: Use Cases - Participar num Leilão.....	21
Tabela 10: Use Cases - Finalizar Compra.....	22
Tabela 11: Use Cases - Finalizar Leilão.....	22
Tabela 12: Use Cases - Visualizar A Minha Conta.....	23
Tabela 13: Use Cases - Visualizar As Minhas Licitações.....	23
Tabela 14: Use Cases - Visualizar Os Meus Leilões.....	24
Tabela 15: Use Cases - Terminar Sessão.....	24

1. Introdução

1.1. Contextualização

Em meados dos anos 2000, houve um desenvolvimento exponencial da *internet*, dando asas aos seus utilizadores para desenvolver sites dos mais variados assuntos. Vendo esta liberdade criativa como uma oportunidade de negócio, Alma Noronha decidiu que podia reinventar o negócio familiar – uma casa de leilão de antiguidades, fundada pelo seu avô após a guerra – e expandi-lo além-fronteiras, na sua própria plataforma *online*.

Alma expôs o seu plano a alguns dos seus amigos na universidade, e formaram a equipa, dividindo entre si as tarefas que tinham pela frente. Agora que o negócio almejava um público-alvo maior, tinham de criar um sistema de organização – os registos físicos utilizados pelos seus pais e avós não eram a solução ótima, pelo que seriam substituídos por um sistema de base de dados. Além disso, seria necessário criar o *website*, com uma interface e funcionamento fácil de entender para novos e antigos clientes, para, deste modo, não perder os clientes leais ao negócio.

1.2. Apresentação do Caso de Estudo

No nosso caso, as necessidades de Alma e da sua família circundavam a necessidade da criação de um sistema de base de dados para poder organizar de maneira mais eficiente e segura todos os dados dos seus clientes e leilões, de maneira a que o salto do mundo físico para o digital fosse possível sem perdas de informação, clientes ou segurança.

1.3. Fundamentação

Após observação de alguns websites de leilões – *Ebay*, *Catawiki*, etc. – ficámos com uma ideia das informações cruciais que devemos ter em conta no processo de criação da base de dados para o negócio de leilões. No nosso caso, o sistema criado pelo avô de Alma não ia ser adequado para um negócio digital, não só pelo facto de ser pouco eficiente ter um registo em papel dos artigos em leilão e dos seus detalhes, as ofertas sugeridas bem como as vencedoras e as informações dos clientes, como, agora que

planeavam alcançar um público exponencialmente maior, iam ocorrer imensos problemas de organização de tanta informação, podendo causar erros e confusões que causariam o declínio do negócio.

1.4. Motivação e Objectivos

Tal como referido no ponto anterior, a transição de um negócio de leilões tradicional para o mundo digital pode resolver imensos problemas e trazer diversas vantagens. Foi feita uma análise dos contratempos que o antigo sistema causava, e chegámos à conclusão de que a implementação de um novo sistema de base de dados desempenha um papel fundamental nesta transição, permitindo o armazenamento, gerenciamento e análise eficiente de dados. Deste modo, podem ser resolvidos os seguintes problemas:

1 Escalabilidade: À medida que um negócio de leilões *online* cresce, a capacidade de escala é essencial. Um sistema de base de dados digital pode ser dimensionado facilmente para acomodar um grande número de leilões, itens e participantes.

2 Acessibilidade: Um sistema digital permite que os utilizadores participem em leilões a partir de qualquer lugar do mundo, eliminando barreiras geográficas e aumentando a base de clientes em potencial.

3 Organização de itens: O sistema de base de dados pode organizar detalhes completos de cada item em leilão, incluindo descrições, imagens, histórico de ofertas e informações sobre o vendedor, sendo mais fácil para procurar algum destes tópicos em específico.

4 Segurança: A segurança dos dados é fundamental num negócio digital. Um sistema de base de dados pode implementar medidas robustas de segurança para proteger informações sensíveis, como informações de pagamento e dados pessoais dos utilizadores.

5 Análise de dados: Com um sistema de base de dados digital, é possível coletar e analisar dados sobre o comportamento dos participantes, histórico de ofertas, tendências de mercado, etc., o que pode ajudar a tomar decisões estratégicas e melhorar a experiência do utilizador.

6 Controlo do inventário: Um sistema de base de dados pode ajudar a manter um controlo preciso do inventário de itens em leilão, evitando a sobreposição e a duplicação de listagens.

1.5. Estrutura do Relatório

Este relatório é composto por 15 capítulos.

Primeiro começamos por apresentar e **contextualizar** o projeto em si no mundo real, apresentando o caso de estudo e a **fundamentação** por detrás da criação desta aplicação.

No segundo capítulo, apresentamos a **viabilidade** deste projeto no mundo real.

No terceiro capítulo, começamos o **planeamento** do projeto, fazendo uma análise das diversas etapas do projeto ao longo do tempo, bem como os recursos necessários e a equipa por detrás do projeto.

No quarto capítulo, descrevemos o **levantamento de requisitos** efetuado com a explicação dos requisitos funcionais e não funcionais de forma detalhada.

No quinto capítulo aprofundamos a especificação e modulação do **software**, através do **modelo de domínio** e da apresentação e descrição dos **Use Cases**.

A partir do sexto capítulo, inclusive, quando apresentamos a **estrutura da base de dados**, começamos a descrever o desenvolvimento do projeto propriamente dito.

No capítulo sete apresentamos os **mockups** da interface do **website**, realizados no **Procreate**.

No capítulo oito, introduzimos as **ferramentas** utilizadas ao longo da implementação do projeto, tanto no **backend** e **frontend**, como a nível de servidor, testagem e segurança.

No capítulo nove descrevemos a **conexão à base de dados**, bem como as tabelas existentes no nosso projeto.

No capítulo dez, aprofundamos a fase de **desenvolvimento**, e entramos em detalhe sobre métodos de gestão de informação e uso de classes.

No capítulo onze, explicamos em maior detalhe como foram efetuados os testes à **segurança** do nosso projeto, bem como as medidas que tomámos para que fosse considerado um **website** seguro.

No capítulo doze, explicamos de maneira detalhada as técnicas de **testagem**, tanto manual como automatizada.

No capítulo treze apresentamos a **arquitetura final do projeto** e a sua relação com as tecnologias usadas.

Por fim, no capítulo catorze apresentamos o **produto final** com exemplos de utilização deste.

2. Viabilidade

A viabilidade de qualquer projeto de desenvolvimento de software baseia-se principalmente na capacidade de identificar e atender às necessidades do mercado e do público. Para tal, é necessário recorrer a um planeamento metódico e eficaz de modo a proporcionar aos utilizadores um produto confiável e de qualidade.

Considerando agora a viabilidade do nosso site de leilões de antiguidades, *BetterFinds*, acreditamos que sua implementação é totalmente viável e fazível na prática, dado que este será um sistema que permite a realização de várias operações que servem a comunidade de forma simples, rápida e com transações seguras, e que, ao ser um exemplo de vanguarda para os negócios de leilões físicos transitarem com facilidade para o mundo digital, pode acabar por contribuir futuramente para a sustentabilidade.

Acreditamos que, na realidade que vivemos, estes são dois tópicos imensamente relevantes e de grande importância para o crescimento deste tipo de negócios.

3. Plano de Desenvolvimento

3.1. Plano de Execução de Trabalho

O projeto foi planeado para ser desenvolvido ao longo de 15 semanas (desde 10 de outubro até 29 de janeiro). Por conseguinte, dividimos o mesmo em duas fases distintas: planeamento do projeto e desenvolvimento. Em paralelo foi planeado ser escrito o presente relatório. De modo a elaborar um plano de execução de trabalho eficaz, decidimos recorrer à realização de um Diagrama de GANTT, pois este permite-nos idealizar um período adequado à realização deste trabalho, bem como organizar objetivos e metas eficientes.

Para além disso, este torna-se igualmente benéfico quanto ao planeamento da divisão de tarefas, garantindo assim uma boa distribuição e planificação do trabalho a realizar.

Nome da Tarefa	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Estudo de Mercado															
Definição de Objetivos															
Definição de Use Cases															
Modelação do Projeto															
Planeamento da Interface															
Desenvolvimento e Testagem															
Escrita do Relatório															

Figura 1: Diagrama de GANTT

3.2. Equipa de Trabalho



*Figura 3: Flávia Alexandra
Silva Araújo (A96587)*



*Figura 2: Miguel Torres
Carvalho (A95485)*

3.3. Recursos a Utilizar

No que toca aos recursos que vamos utilizar, podemos dizer que vamos recorrer a três tipos de recursos, sendo estes humanos, digitais e físicos.

Os recursos humanos são importantes no sentido em que estes são os principais motores da implementação da aplicação, são o motivo que leva ao desenvolvimento da plataforma.

Recursos Humanos:

- Vendedores;
- Compradores;
- Equipa de Trabalho.

Os recursos digitais são igualmente importantes, pois estes auxiliam no desenvolvimento da aplicação, permitindo, deste modo, atingir a finalidade proposta pela equipa de trabalho.

Recursos Digitais:

- Microsoft Visual Studio (Versão 17.8.5);
- Microsoft .NET (Versão 8.0.101);
- Microsoft Office;
- Microsoft #C;
- LibreOffice;
- DeepSource;
- Snyk;
- Microsoft SQL Server 2022 (Versão 16.0.4105.2) (VS Server Explorer);
- GitHub Copilot: Code Completion;
- Nginx;
- Cloudflare;
- DigitalOcean;
- LambdaTest;
- amen.pt (domínio: betterfinds.pt);
- CertBot;
- freedns.afraid.org (domínio: betterfinds.us.to);
- Visual Paradigm;
- Procreate;
- Adobe Illustrator;
- LucidChart;
- draw.io;
- Sistemas Operativos: Windows 11 e Linux (Ubuntu 22.04.3 LTS).

Extra (Bibliotecas .NET):

- Microsoft.AspNetCore.Identity.EntityFrameworkCore (versão 8.0.1);
- Microsoft.AspNetCore.SignalR.Common (versão 8.0.1);
- Microsoft.Data.SqlClient (versão 5.1.4);
- Microsoft.EntityFrameworkCore (versão 8.0.1);
- Microsoft.EntityFrameworkCore.Design (versão 8.0.1);
- Microsoft.EntityFrameworkCore.SqlServer (versão 8.0.1);
- Microsoft.EntityFrameworkCore.Tools (versão 8.0.1);
- Microsoft.Extensions.Configuration (versão 8.0.0);
- Microsoft.Extensions.Identity.Core (versão 8.0.1);
- System.IdentityModel.Tokens.Jwt (versão 7.2.0).
- **Bibliotecas .NET adicionais relativas ao projeto de testagem:**
 - Microsoft.NET.Test.Sdk (versão 17.8.0);

- *NUnit* (versão 4.0.1);
- *NUnit.Analyzers* (versão 3.10.0);
- *NUnit3TestAdapter* (versão 4.5.0);
- *coverlet.collector* (versão 6.0.0);
- *Selenium.Support* (versão 4.16.2);
- *Selenium.WebDriver* (versão 4.16.2);
- *Selenium.WebDriver.WaitExtensions* (versão 2.0.0);
- *SeleniumExtras.WaitHelpers* (versão 1.0.2);
- *DotNetSeleniumExtras.WaitHelpers* (versão 3.11.0).

Quanto aos recursos físicos, estes são especialmente essenciais para garantir a operação contínua de nosso projeto, bem como um meio indispensável para a concretização do mesmo.

Recursos Físicos:

- Computadores;
- *Tablet* (para desenho de *mockups*);
- Telemóvel (para teste do *layout mobile*);
- Conexão à *internet*;
- Servidores.

4. Levantamento de Requisitos

4.1. Apresentação da Estratégia e Método

De modo a compreender o funcionamento e logística do gerenciamento de uma plataforma de leilões, recorremos a diversos meios de informação de forma a tornar o nosso caso o mais próximo possível de uma situação e problema real.

Quanto à estratégia e método adotados pela equipa de trabalho, é possível afirmar que, de um modo geral, após a verificação de vários sites de leilões, decidimos seguir as suas ideologias e adotar as principais funcionalidades no nosso projeto.

Dado isso, e tendo em consideração o tema do projeto, os leilões *online*, o nosso levantamento de requisitos seguiu na base do nosso conhecimento sobre o tema, mas fundamentou-se principalmente após algumas pesquisas e consultas em sites como o eBay, por exemplo, nos quais se pode fazer leilões de vários produtos.

Para além do conhecimento geral e da consulta de websites, seguimos paralelamente uma avaliação de cenários.

A avaliação de cenários permite-nos obter os requisitos funcionais, uma vez que, através desse processo, é-nos possível prever como o sistema será utilizado em diferentes situações e contextos, identificando as funcionalidades e comportamentos necessários para atender às necessidades dos utilizadores e garantir que o sistema atenda aos objetivos estabelecidos.

4.2. Descrição Geral dos Requisitos Levantados

4.2.1. Requisitos Funcionais

4.2.1.1. Gestão de Utilizador

Requisitos de Utilizador:

RF1: A aplicação permite que os utilizadores se registem no site.

RF2: O utilizador deve conseguir editar o seu nome completo, *email*, *password*, *username*, foto de perfil, e se pretende receber a *newsletter*.

Requisitos de Sistema:

RF3: O sistema regista as informações pessoais do utilizador - nome completo, nome de utilizador, foto de perfil, *email* e *password*.

RF4: O sistema não deve permitir registo de utilizadores com o mesmo *email* e/ou nome de utilizador.

RF5: O sistema apenas permite registar nomes completos entre 3 a 64 caracteres (e apenas caracteres alfabéticos e espaços), nomes de utilizador entre 3 a 32 caracteres (e apenas caracteres alfanuméricos), *passwords* entre 8 a 64 caracteres, e *emails* entre 5 a 320 caracteres.

RF7: O sistema regista se o utilizador quer receber a *newsletter*.

RF8: O sistema deve armazenar os dados do utilizador.

RF9: O sistema deve mostrar os dados do utilizador aquando a edição dos mesmos.

4.2.1.2. Procurar Produto

Requisitos de Utilizador:

RF10: O utilizador é capaz de pesquisar pelo produto pretendido através da caixa de pesquisa.

Requisitos de Sistema:

RF11: O sistema deverá permitir o utilizador de pesquisar produtos através de palavras na barra de pesquisa.

RF12: O sistema deve apresentar todos os produtos correspondentes ou associados à pesquisa efetuada pelo utilizador.

RF13: O sistema deve apresentar, para além da barra de pesquisa, a opção de ordenar os resultados ou leilões por data final, preço e nome.

4.2.1.3. Criar Leilão

Requisitos de Utilizador:

RF14: O utilizador pode criar leilões fornecendo detalhes sobre o produto a ser leiloado - nome do produto, data de término do leilão, descrição, imagens (opcional), licitação mínima (opcional), preço.

RF15: O utilizador é capaz de editar ou alterar o produto.

Requisitos de Sistema:

RF16: O sistema regista as informações dos leilões - nome do produto, data de término do leilão, descrição, imagens (opcional), licitação mínima (opcional), preço.

RF17: O sistema permite que o utilizador edite o seu leilão.

RF18: O sistema deverá restringir a edição de leilões aos quais o utilizador não tem acesso.

RF19: O sistema atualiza a sua base de dados conforme os produtos são carregados na plataforma.

4.2.1.4. Finalizar Leilão

Requisitos de Utilizador:

RF20: O utilizador é capaz de encerrar o leilão antes da data de término prevista, sendo a licitação vencedora a maior até esse instante.

RF21: O utilizador é capaz de eliminar um leilão caso não tenha mais intenções em leiloar o produto.

Requisitos de Sistema:

RF22: O sistema finaliza o leilão automaticamente assim que a data limite é alcançada, sendo a licitação vencedora a maior até esse instante.

RF23: O sistema é capaz de indicar que o leilão não se encontra mais ativo.

RF24: O sistema atualiza a sua base de dados sobre os leilões que se encontram ativos na plataforma.

4.2.1.5. Participar num Leilão

Requisitos de Utilizador:

RF25: O utilizador participa num leilão selecionando essa opção.

RF26: O utilizador indica a sua licitação no leilão.

Requisitos de Sistema:

RF27: O sistema deve impedir o vendedor de licitar no seu próprio leilão.

RF28: O sistema deve impedir o utilizador de licitar se a sua licitação já for a mais alta no leilão.

RF29: O sistema permite que o utilizador participe no leilão exibindo essa opção.

RF30: O sistema exibe ao utilizador um local para indicar a sua licitação.

RF31: O sistema informa o utilizador se a sua licitação foi ou não aceite.

RF32: O sistema exibe o valor da maior licitação atual de modo a permitir que o utilizador acompanhe a atividade do leilão.

RF33: O sistema notifica o utilizador de alterações no estado do leilão – se a sua licitação foi ultrapassada, se o leilão terminou, e/ou se foi o vencedor, podendo, deste modo, proceder ao pagamento.

4.2.1.6. Finalizar Compra

Requisitos de Utilizador:

RF34: O utilizador que venceu o leilão é capaz de efetuar a compra do produto.

RF35: O utilizador é capaz de selecionar o meio de pagamento.

Requisitos de Sistema:

RF36: O sistema notifica o utilizador da opção de pagamento após vencer o leilão.

RF37: O sistema calcula e exibe o valor do produto.

RF38: O sistema exibe a opção de pagamento ao licitante vencedor.

RF39: O sistema exibe os métodos de pagamento existentes.

RF40: O sistema atualiza a sua base de dados conforme a compra é finalizada e a transação encerrada.

4.2.2. Requisitos Não Funcionais

RNF1: A aplicação deve ser compatível com diversos dispositivos, como computadores, *tablets* e *smartphones*, bem como *browsers*, garantindo, assim, uma experiência completa e consistente ao utilizador.

RNF2: A aplicação deve estar em conformidade com as regulamentações estabelecidas no Código Penal de Portugal.

RNF3: O tempo de resposta do sistema para a atualização de valores e notificações deve ser inferior a 2 segundos, mesmo em momentos de alta atividade.

RNF4: A segurança dos dados do utilizador e informações financeiras deve atender aos padrões de segurança estabelecidos.

RNF5: O sistema deve ser escalável (permite que o sistema atenda às necessidades à medida que estas crescem), permitindo a adição de novos servidores ou recursos para lidar com o crescimento do tráfego.

RNF6: A aplicação implementa uma política de *backup* de dados regular e confiável para evitar a perda de informações críticas.

RNF7: A aplicação deverá estar disponível 24 horas por dia, 7 dias por semana.

RNF8: O sistema está configurado por pré-definição no fuso horário UTC (*Coordinated Universal Time*) para garantir uniformidade entre os horários dos leilões para os utilizadores.

5. Especificação e Modelação do Software

5.1. Apresentação Geral Da Especificação

Após a definição e validação dos requisitos, entramos na fase de elaborar a estrutura do nosso sistema e estabelecer as orientações para o seu funcionamento.

Neste ponto, concentrarmo-nos demonstrar como o sistema será organizado e quais serão suas operações e comportamentos essenciais.

5.2. Modelo de Domínio

À medida que adentramos no mundo do desenvolvimento de um sistema de leilão de antiguidades, a construção de um alicerce sólido e claro é fundamental. O modelo de domínio, representado por meio de um Diagrama de Classes, emerge como a ferramenta central para mapear o universo intrincado de antiguidades, licitações, utilizadores e interações que caracterizam nosso sistema.

Imagine o modelo de domínio como o catálogo meticoloso de um leiloeiro, identificando cada peça de valor dentro do nosso contexto de leilão de antiguidades.

Neste contexto, exploraremos a relevância e a importância do modelo de domínio específico para leilões de antiguidades. Veremos como ele desvenda a estrutura essencial do sistema, identifica os principais participantes, como utilizadores e licitadores, e estabelece as conexões que nos ajudam a realizar o nosso sistema.

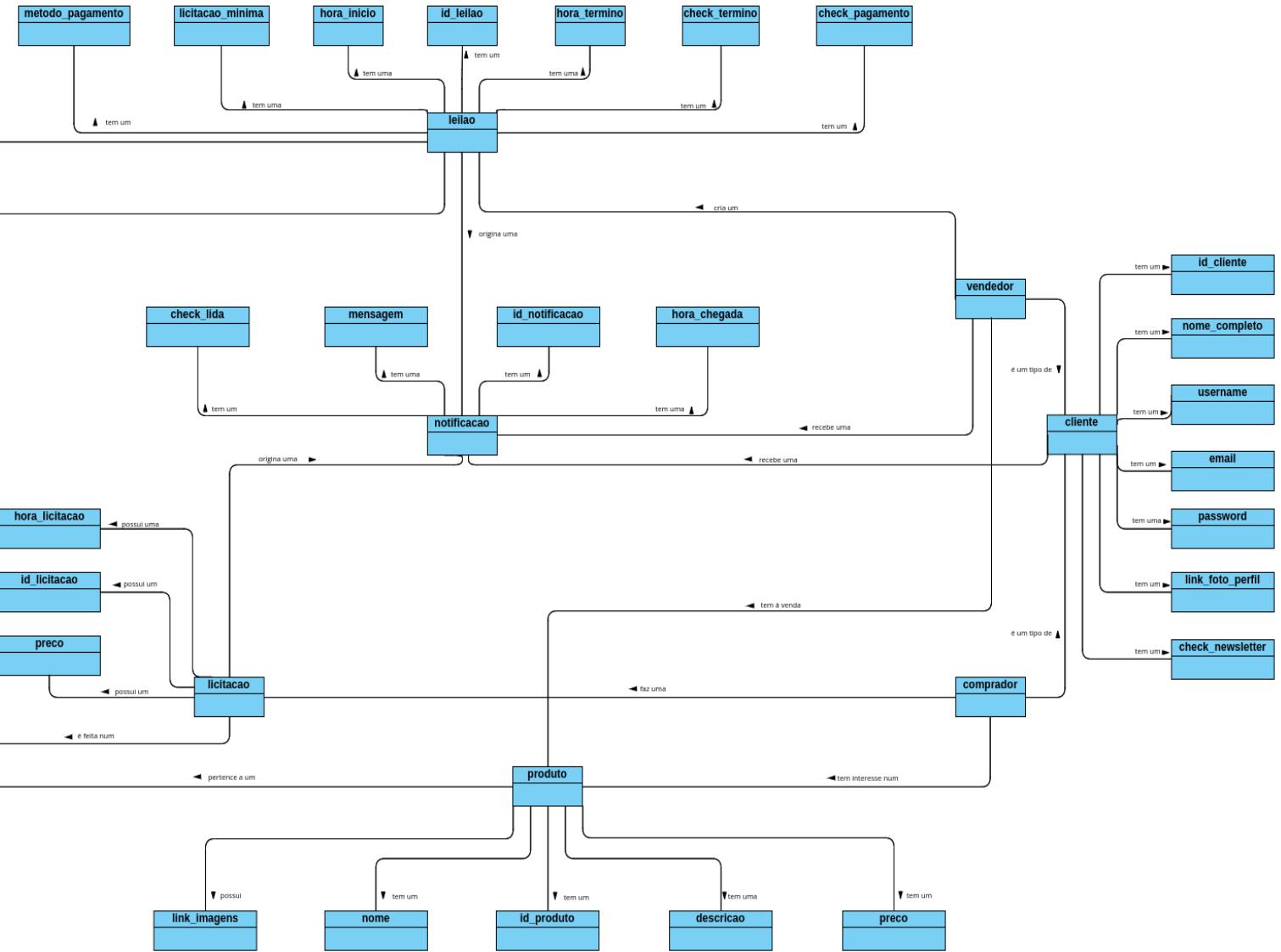


Figura 4: Diagrama de Classes

5.3. Diagrama de Use Cases

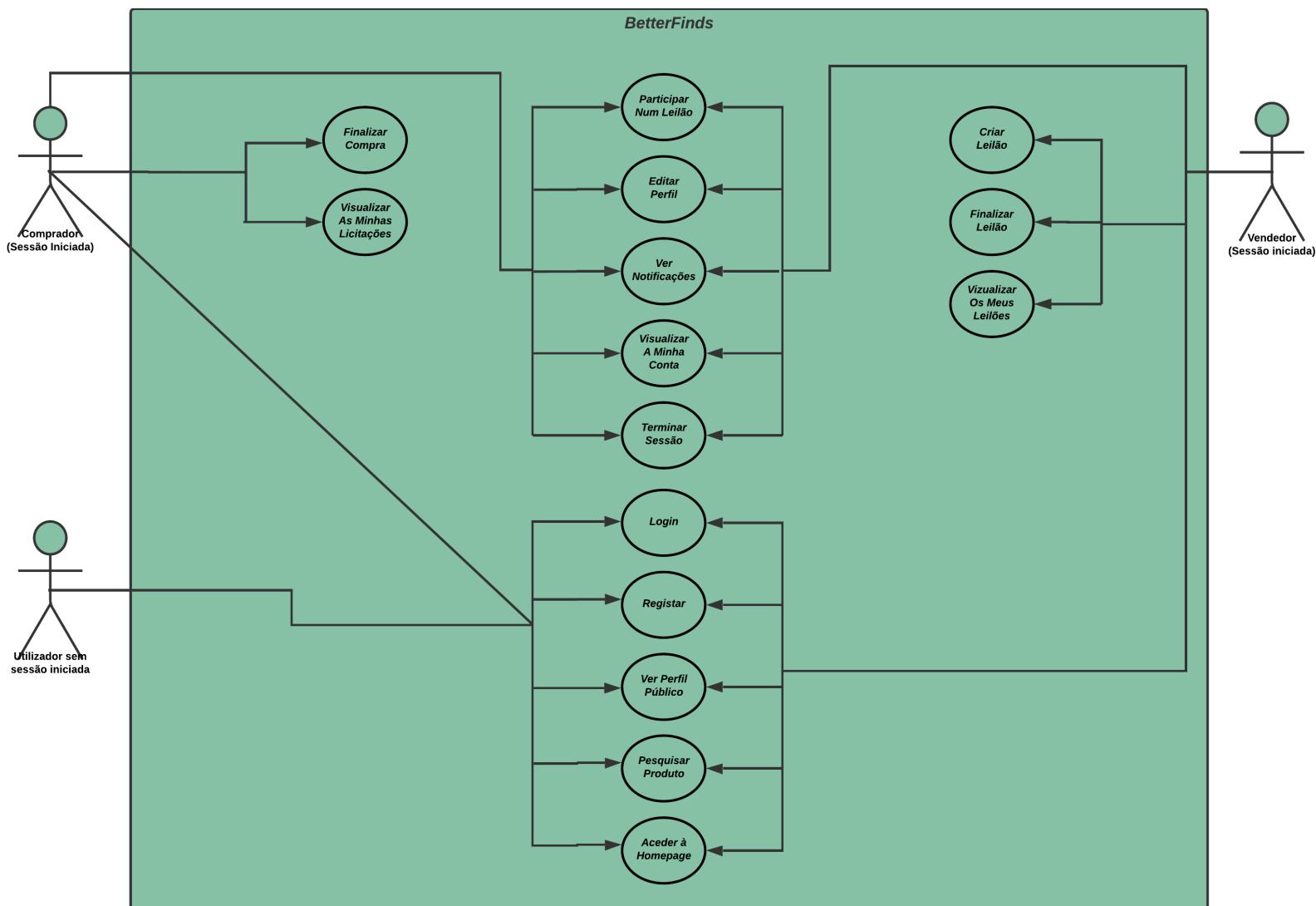


Figura 5: Diagrama de Use Cases

Uma das vantagens de fazer a transição do negócio de leilões de antiguidades do mundo físico para o digital, e um dos porquês que instigou a curiosidade da nossa equipa de trabalho, é que, neste caso, qualquer um dos nossos utilizadores pode ser Vendedor e/ou Comprador. A partir do momento em que se regista na nossa plataforma, consegue facilmente intercalar entre vender artigos e participar nos leilões que sempre quis, em qualquer lugar, a partir do website.

5.4. Especificação de Use Cases

5.4.1. Registar

Descrição: O utilizador regista-se na plataforma de Leilões de Antiguidades.		
Use Case:	Registrar	
Autor:	Utilizador	
Pré-Condição:	O visitante tem meio de acesso à plataforma.	
Pós-Condição:	O utilizador fica registado na plataforma de leilões.	
	Autor	Sistema
Fluxo Normal	1. O visitante regista-se no site.	
	2. Durante o processo de registo, este insere as suas informações pessoais (nome, email, etc).	
		3. O sistema aceita e valida as informações.
		4. O sistema cria a conta do utilizador.
Fluxo Alternativo [O email e/ou nome de utilizador já existem, ou não cumpre os requisitos de caracteres no nome/email/password/nome de utilizador] (voltar ao passo 2)		1. O sistema rejeita os dados inseridos pelo visitante pedindo novos dados que cumpram os requisitos especificados.

Tabela 1: Use Cases - Registar

5.4.2. Efetuar Login

Descrição: O utilizador efetua login na plataforma.		
Use Case:	Login	
Autor:	Comprador e Vendedor	
Pré-Condição:	As credenciais têm de ser válidas.	
Pós-Condição:	O utilizador tem acesso às funcionalidades restritas do sistema.	
	Ator	Sistema
Fluxo Normal	1. O utilizador acede ao site.	
	2. O utilizador insere as suas credenciais e opção de lembrar sessão.	
		3. O sistema verifica e confirma as credenciais inseridas.
	4. O utilizador efetua login com sucesso.	
Fluxo Alternativo [O nome de utilizador e/ou password introduzidos pelo utilizador são inválidos] (voltar ao passo 2)		1. O sistema rejeita as credenciais inseridas pelo utilizador e dá refresh à página para reintroduzi-las.

Tabela 2: Use Cases – Login

5.4.3. Ver Perfil Público

Descrição: O utilizador vizualiza o perfil público.		
Use Case:	Ver Perfil Público	
Autor:	Utilizador	
Pré-Condição:	O utilizador tem acesso à plataforma.	
Pós-Condição:	O utilizador vizualiza o perfil.	
	Ator	Sistema
Fluxo Normal	1. O utilizador clica num nome de utilizador no site.	
		2. O sistema apresenta o perfil ao utilizador.

Tabela 3: Use Cases - Ver Perfil Público

5.4.4. Pesquisar Produto

Descrição: O utilizador procura na plataforma os produtos sobre os quais tem interesse.		
Use Case:	Pesquisar Produto	
Ator:	Utilizador	
Pré-Condição:	O utilizador tem acesso à plataforma.	
Pós-Condição:	O utilizador visualiza os produtos que procura.	
	Autor	Sistema
Fluxo Normal	1. O utilizador acede ao site.	
	2. O utilizador pesquisa na barra de pesquisa os produtos pelos quais se interessa.	
		3. O sistema exibe os resultados da pesquisa.
	4. O utilizador pode ajustar a sua pesquisa, a ordem e a ocorrência dos leilões.	

Tabela 4: Use Cases - Pesquisar Produto

5.4.5. Aceder à Homepage

Descrição: O utilizador accede à Homepage, onde pode ver as novidades de leilões.		
Use Case:	Aceder à Homepage.	
Ator:	Utilizador	
Pré-Condição:	O utilizador tem acesso à plataforma.	
Pós-Condição:	O utilizador visualiza a Homepage.	
	Autor	Sistema
Fluxo Normal	1. O utilizador efetua login.	
		2. O sistema redireciona o utilizador para a sua Homepage.
Fluxo Alternativo [O utilizador quer apenas visitar o site]		1. O sistema redireciona o utilizador para a sua Homepage de convidado.

Tabela 5: Use Cases – Aceder à Homepage

5.4.6. Editar Perfil

Descrição: O utilizador altera as suas informações pessoais de perfil.		
Use Case:	Editar Perfil	
Autor:	Utilizador	
Pré-Condição:	O utilizador tem sessão iniciada.	
Pós-Condição:	Alguma informação pessoal é alterada.	
	Ator	Sistema
Fluxo Normal	1. O utilizador acede à página "My Account".	
		2. O sistema apresenta os campos editáveis do perfil.
	3. O utilizador seleciona qual campo pretende editar.	
	4. O utilizador grava as alterações.	
Fluxo Alternativo [As novas informações não cumprem os requisitos de caracteres no email/password/nome de utilizador] (voltar ao passo 3)		1. O sistema rejeita os dados inseridos pelo utilizador pedindo novos dados que cumpram os requisitos especificados.

Tabela 6: Use Cases - Editar Perfil

5.4.7. Ver Notificações

Descrição: O utilizador acede o seu painel de notificações.		
Use Case:	Ver Notificações	
Autor:	Utilizador	
Pré-Condição:	O utilizador tem sessão iniciada.	
Pós-Condição:	O utilizador visualiza as suas notificações.	
	Ator	Sistema
Fluxo Normal	1. O utilizador acede ao site.	
		2. O sistema apresenta todas as notificações do utilizador na sua página de notificações.

Tabela 7: Use Cases - Ver Notificações

5.4.8. Criar Leilão

Descrição: Um utilizador insere um produto na plataforma e especifica os seus detalhes.		
Use Case:	Criar Leilão	
Ator:	Vendedor	
Pré-Condição:	O vendedor está registado e tem o <i>login</i> efetuado na plataforma.	
Pós-Condição:	O vendedor cria o leilão e o produto fica disponível na plataforma.	
	Ator	Sistema
Fluxo Normal	1. O utilizador acede à opção para criar um leilão.	
	2. O vendedor fornece informações válidas sobre o produto, incluindo descrição, fotos (opcional), preço base, licitação mínima e data de término do leilão.	
		3. O sistema cria o leilão, apresentando o produto na plataforma.
Fluxo Alternativo [As informações sobre o produto não são válidas ou suficientes] (voltar ao passo 2)		1. O sistema apresenta um aviso que as mesmas devem ser corrigidas.

Tabela 8: Use Cases - Criar Leilão

5.4.9. Participar num Leilão

Descrição: O utilizador regista-se num leilão.		
Use Case:	Participar num Leilão	
Ator:	Utilizador	
Pré-Condição:	O utilizador precisa de estar registado na plataforma e de ter efetuado <i>login</i> , bem como o leilão que este tem interesse tem de estar ativo.	
Pós-Condição:	O utilizador fica registado na participação de um leilão.	
	Ator	Sistema
Fluxo Normal	1. O utilizador encontra na plataforma um produto do seu interesse.	
	2. O utilizador verifica os detalhes do produto.	
	3. O utilizador decide participar no leilão e insere o valor que dá pelo artigo.	
		4. O sistema verifica e confirma a validade da participação.
		5. O sistema verifica e confirma a validade do valor introduzido.
		6. O valor é aceite e o sistema atualiza o valor do produto licitado.
		7. O sistema notifica todos os utilizadores participantes da alteração no valor.
Fluxo Alternativo [A licitação não é aceite] (voltar ao passo 3)		1. O sistema rejeita o valor introduzido.
		2. O sistema permite que o utilizador ajuste o valor de acordo com os parâmetros especificados.
Fluxo Alternativo [O licitador é um visitante no site]		1. O sistema rejeita a participação no leilão.
		2. O sistema indica que o utilizador da plataforma se registe ou faça <i>login</i> de modo que a participação seja válida.
Fluxo Alternativo [O licitador é o próprio vendedor]		1. O sistema rejeita a participação no leilão.

Tabela 9: Use Cases - Participar num Leilão

5.4.10. Finalizar Compra

Descrição: O utilizador é o vencedor de um leilão e tem de finalizar a compra do produto através do pagamento.		
Use Case:	Finalizar Compra	
Ator:	Comprador	
Pré-Condição:	O comprador está registado e tem o <i>login</i> efetuado na plataforma.	
Pós-Condição:	A compra é efetuada com sucesso e o produto é enviado.	
	Ator	Sistema
Fluxo Normal	1. Um utilizador é o vencedor do leilão.	
	2. O comprador procede ao pagamento do produto.	
	3. O comprador define o método de pagamento.	
	4. O sistema confirma a compra.	

Tabela 10: Use Cases - Finalizar Compra

5.4.11. Finalizar Leilão

Descrição: O leilão finaliza aquando a data de término.		
Use Case:	Finalizar Leilão	
Ator:	Vendedor	
Pré-Condição:	O vendedor está registado, tem o <i>login</i> efetuado e tem pelo menos um leilão ativo.	
Pós-Condição:	O leilão é finalizado com sucesso.	
	Ator	Sistema
Fluxo Normal		1. O sistema encerra o leilão assim que alcança a data limite, notificando os participantes e o vendedor.
Fluxo Alternativo [O vendedor decide finalizar antes do tempo] (voltar ao passo 1)	1. O vendedor acede aos seus leilões.	
	2. O vendedor finaliza o leilão.	
Fluxo Alternativo [O vendedor cancela o leilão]	1. O vendedor acende aos seus leilões.	
	2. O vendedor cancela o leilão.	

Tabela 11: Use Cases - Finalizar Leilão

5.4.12. Visualizar A Minha Conta

Use Case: O utilizador vizualiza a sua página "My Account"		
Use Case:	Visualizar A Minha Conta	
Ator:	Utilizador	
Pré-Condição:	O utilizador tem o <i>login</i> efetuado..	
Pós-Condição:	O utilizador vizualiza a sua página de conta.	
	Ator	Sistema
Fluxo Normal	1. O utilizador clica na página "My Account".	
		2. O sistema exibe as informações da sua página privada de conta e as suas estatísticas.

Tabela 12: Use Cases – Visualizar A Minha Conta

5.4.13. Visualizar As Minhas Licitações

Use Case: O utilizador vizualiza a sua página "My Bids"		
Use Case:	Visualizar As Minhas Licitações	
Ator:	Utilizador	
Pré-Condição:	O utilizador tem o <i>login</i> efetuado..	
Pós-Condição:	O utilizador vizualiza a sua página de licitações.	
	Ator	Sistema
Fluxo Normal	1. O utilizador clica na página "My Bids".	
		2. O sistema exibe as informações da sua página privada de licitações.

Tabela 13: Use Cases – Visualizar As Minhas Licitações

5.4.14. Visualizar Os Meus Leilões

Use Case: O utilizador vizualiza a sua página "My Auctions"		
Use Case:	Visualizar Os Meus Leilões	
Ator:	Utilizador	
Pré-Condição:	O utilizador tem o <i>login</i> efetuado..	
Pós-Condição:	O utilizador vizualiza a sua página de leilões..	
	Ator	Sistema
Fluxo Normal	1. O utilizador clica na página "My Auctions".	
		2. O sistema exibe as informações da sua página privada de leilões.

Tabela 14: Use Cases – Visualizar Os Meus Leilões.

5.4.15. Terminar Sessão

Use Case: O utilizador termina a sessão na plataforma.		
Use Case:	Terminar Sessão	
Autor:	Utilizador	
Pré-Condição:	O utilizador tem o <i>login</i> efetuado..	
Pós-Condição:	O utilizador termina sessão com sucesso.	
	Ator	Sistema
Fluxo Normal	1. O utilizador clica em "Logout".	
		2. O sistema processa o pedido.
		3. O sistema redireciona o utilizador não autenticado para a página de <i>login</i> .

Tabela 15: Use Cases – Terminar Sessão

5.5. Aspectos Comportamentais

Tendo em conta que o projeto a desenvolver se trata de uma plataforma de leilões, é fulcrual retratar o comportamento dos utilizadores. Assim, utilizando diagramas de atividade, conseguimos ilustrar de modo gráfico todos os comportamentos e ações possíveis que um utilizador pode realizar na aplicação, sendo ele Vendedor ou Comprador.

A figura seguinte demonstra o fluxo geral de atividades envolvendo **Vendedores** no contexto da nossa plataforma. Ele inclui autenticação, criação e gerenciamento de leilões, visualização de notificações, interações com compradores e a capacidade de tomar decisões sobre produtos e leilões.

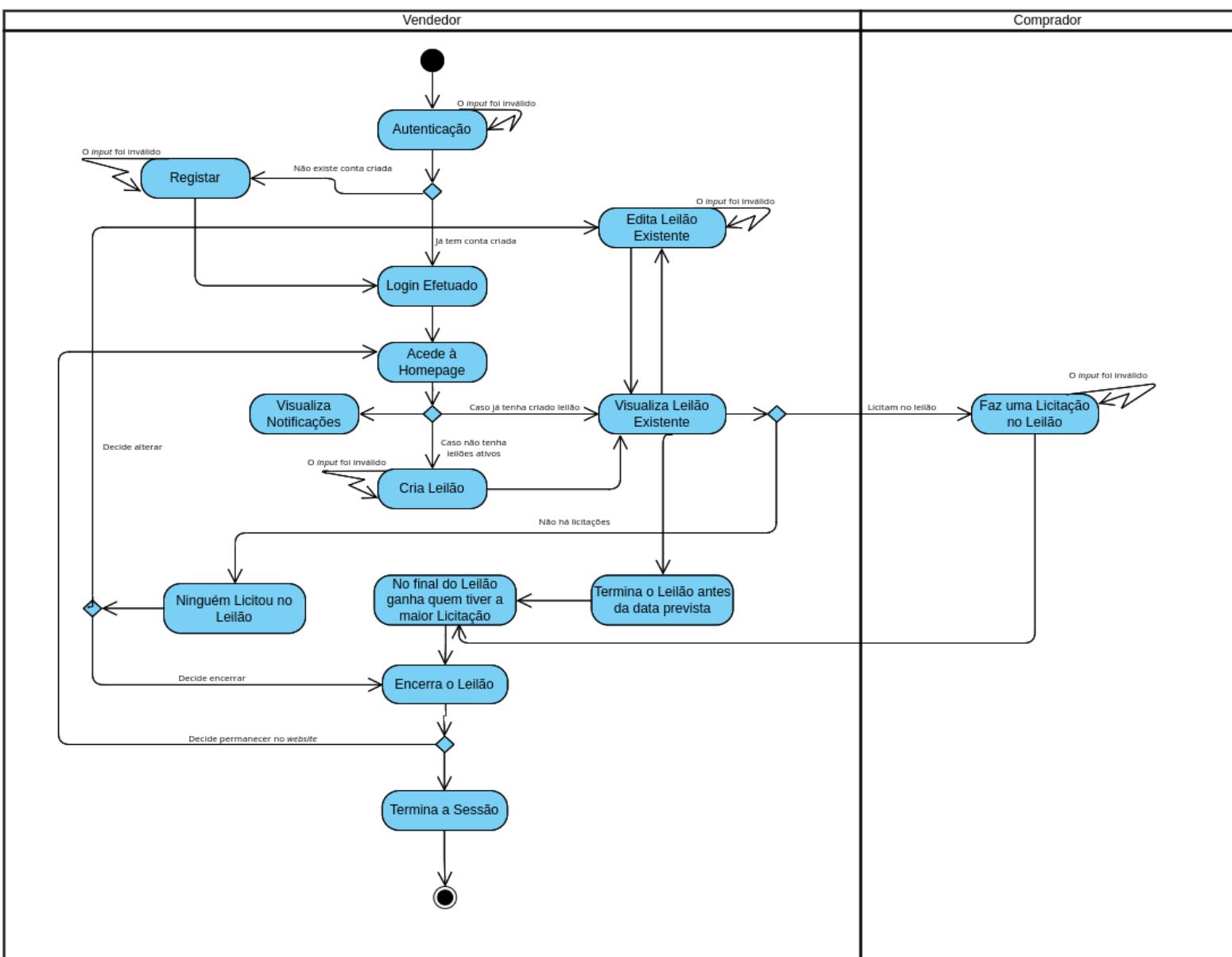


Figura 6: Diagrama de Atividade do Vendedor

Aspectos Comportamentais do Vendedor:

- O Vendedor inicia o processo efetuando *login* na plataforma:
 - Se o Vendedor não tiver uma conta, ele deve criar uma;
 - Se os dados de registo/*login* forem inválidos, ele deve tentar novamente.
- Após o *login* bem-sucedido, o Vendedor é redirecionado à *Homepage*, onde pode:
 - Acessar o seu painel de notificações;
 - Acessar um leilão já existente, se ele já tiver criado um;
 - Editar um dos seus leilões criados, se ele já tiver criado um;
 - Se não tiver um leilão existente, o Vendedor pode criar um novo leilão;
 - Se os dados do leilão forem inválidos, ele deve tentar novamente criar/editar o seu leilão.
- Ao visualizar os seus leilões ativos, o Vendedor pode:
 - Dar o leilão por finalizado antes da data prevista, sendo o vencedor o que tem a maior licitação até ao momento;
 - Esperar que termine na data de término.
- Quando o leilão está ativo, os Compradores podem fazer licitações nos diferentes leilões.
- No final do leilão, o Comprador com a licitação mais alta ganha, e o leilão termina.
- Se ninguém fez licitações no leilão, o Vendedor pode editar o seu leilão ou encerrá-lo.
- Após as suas atividades, o Vendedor pode escolher:
 - Permanecer no *website*, criar novos leilões ou editar os existentes, etc;
 - Terminar sessão.

A figura seguinte corresponde, por outro lado, ao fluxo geral de atividades para **Compradores**. Ele inclui a criação e/ou autenticação de conta, procura por leilões e perfis, acesso e edição do perfil do utilizador, gestão de notificações, decisões de licitação e, no caso de ganhar um leilão, o processo de pagamento antes de continuar a navegar por outros leilões ou encerrar a sessão na plataforma.

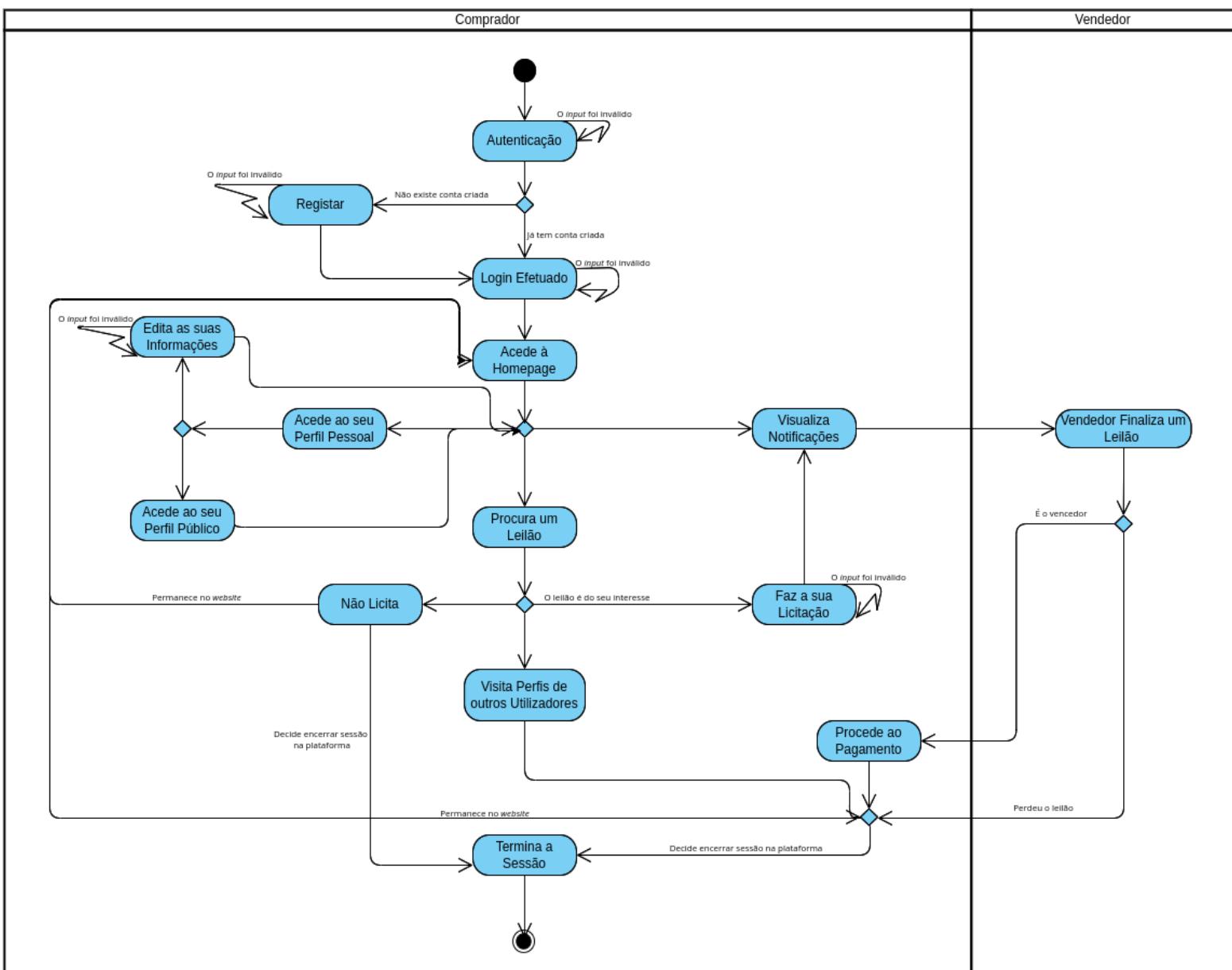


Figura 7: Diagrama de Atividade do Comprador

Aspectos Comportamentais do Comprador:

- O Comprador inicia o processo efetuando *login* na plataforma:
 - Se o Comprador não tiver uma conta, ele deve criar uma;
 - Se os dados de registo/*login* forem inválidos, ele deve tentar novamente.
- Após o *login* bem-sucedido, o Comprador é redirecionado à *Homepage*, onde pode:
 - Aceder ao seu Perfil Pessoal, onde pode:
 - Aceder ao seu Perfil Público;
 - Editar as suas informações pessoais (se o *input* na edição for inválido, ele deve tentar novamente);
 - Procurar um Leilão do seu interesse, onde pode:
 - Ele pode visitar perfis de outros utilizadores no histórico de licitações;
 - Se se interessar pelo produto, pode fazer a sua licitação, e aguardar pela notificação de término do leilão (se o *input* for inválido, deve tentar novamente);
 - Pode não licitar.
 - Aceder ao seu painel de notificações, e, se tiver notificações de leilões finalizados, pode:
 - Se tiver notificações de leilões finalizados e ele for o vencedor, deve proceder ao pagamento;
 - Se tiver notificações de leilões finalizados e ele for não for vencedor, continua com as atividades normais;
 - Pode ter notificações de leilões nos quais participou em que fizeram novas licitações.
- Após as suas atividades, o Comprador pode optar por:
 - Permanecer no *website*, continuar a procurar por leilões disponíveis, etc.;
 - Terminar sessão.

6. Conceção do Sistema de Dados

6.1. Apresentação Geral da Estrutura do Sistema de Dados

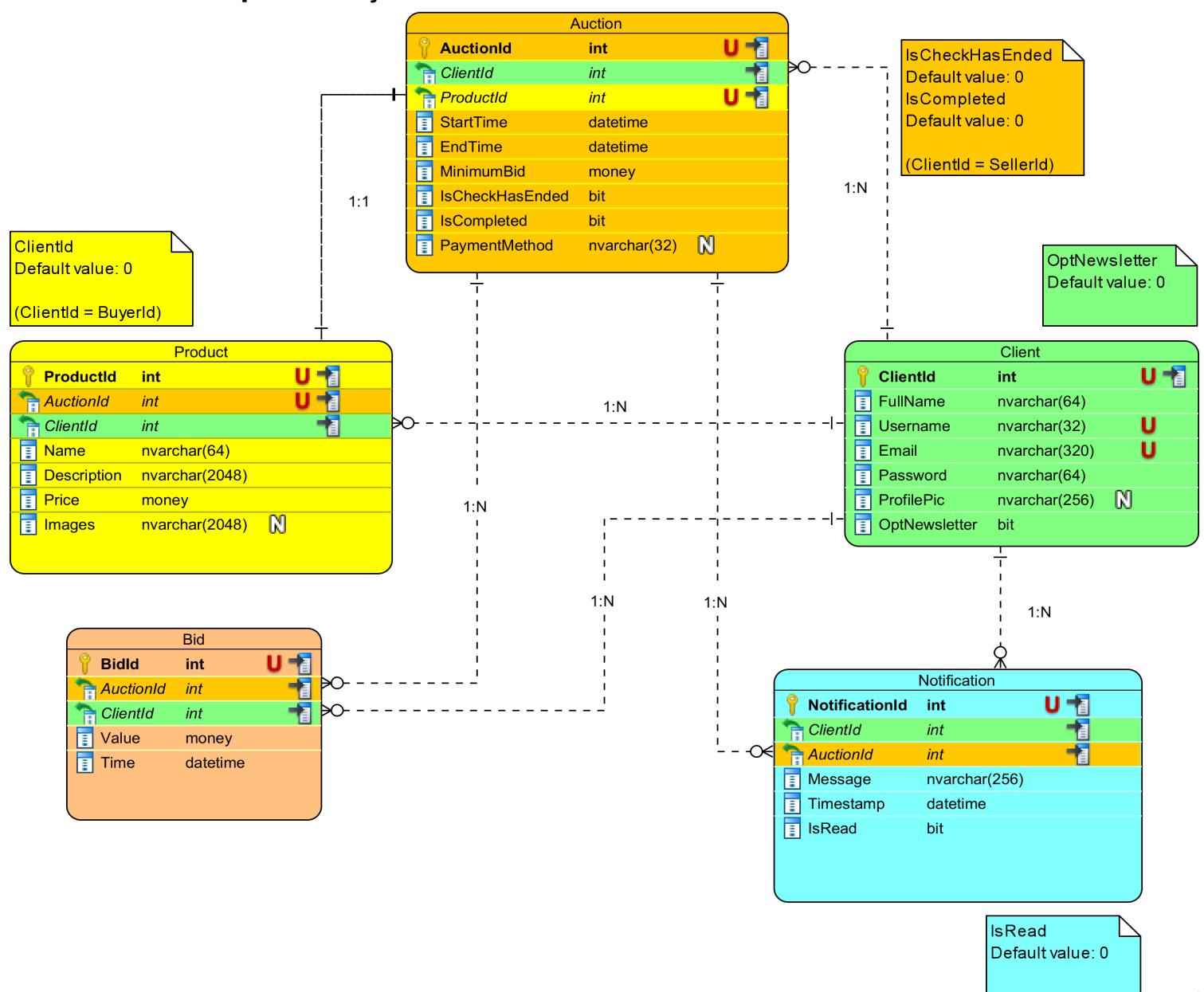


Figura 8: Modelo Lógico do Sistema

6.2. Descrição Detalhada dos Elementos de Dados e Relacionamentos

A base de dados é um aspecto fulcral para a ferramenta a ser desenvolvida, possuindo toda a informação importante para o uso regular da aplicação.

No nosso modelo lógico para um sistema de leilão de antiguidades, cada entidade e atributo desempenham papéis cruciais na representação e funcionalidade do sistema. Esta etapa é fundamental no processo de modelação do sistema, exigindo alinhamento com as metas pretendidas para a implementação posterior. Nesse contexto, a acessibilidade aos dados foi cuidadosamente considerada durante o desenvolvimento.

Além disso, ao analisar os diagramas modelados, torna-se evidente a abordagem de implementação do projeto, juntamente com as decisões tomadas. Este modelo, criado de acordo com práticas metodológicas recomendadas, foi concebido desde o início com a ênfase na produção de um resultado normalizado.

Posto isto, vamos analisar os detalhes de cada entidade e seus relacionamentos:

6.2.1. Entidade “Auction”

AuctionId: Um identificador exclusivo para cada leilão no sistema.

ClientId (Chave Estrangeira): Uma referência ao utilizador que criou o leilão.

ProductId (Chave Estrangeira): Uma referência ao produto que está a ser leiloado.

StartTime: A hora de início do leilão.

EndTime: A hora de término do leilão.

MinimumBid: A licitação mínima obrigatória no leilão.

IsCheckHasEnded: O sistema verifica em tempo real, ciclicamente, se o leilão terminou.

IsCompleted: Após o pagamento ser efetuado, o sistema define o leilão como completo.

PaymentMethod: O método de pagamento escolhido pelo comprador.

6.2.2. Entidade “Client”

ClientId: Um identificador exclusivo para cada utilizador no sistema.

FullName: O nome completo do utilizador.

Username: O username do utilizador, utilizado para comunicação e autenticação.

Email: O email do utilizador.

Password: A palavra-passe associada à conta do utilizador, que garante a segurança dos dados deste.

ProfilePic: A foto de perfil do utilizador.

OptNewsletter: Se o utilizador optou (ou não) por receber a newsletter.

6.2.3. Entidade “Notification”

NotificationId: Um identificador exclusivo para cada notificação no sistema.

ClientId (Chave Estrangeira): Uma referência ao utilizador para quem a notificação é direcionada.

AuctionId (Chave Estrangeira): Uma referência ao leilão para quem a notificação se refere.

Message: A mensagem contida na notificação.

Timestamp: Dia/Hora em que o utilizador recebeu a notificação.

IsRead: Se o utilizador marcou a notificação como lida (ou não).

6.2.4. Entidade “Product”

ProductId: Um identificador exclusivo para cada produto no sistema.

AuctionId (Chave Estrangeira): Uma referência ao leilão ao qual o produto se encontra disponível à venda.

ClientId (Chave Estrangeira): Uma referência ao utilizador que está a vender o produto.

Name: Nome do produto.

Description: Descrição do produto.

Price: Preço do produto.

Images: Fotos do produto.

6.2.5. Entidade “Bid”

BidId: Um identificador exclusivo para cada licitação no sistema.

AuctionId (Chave Estrangeira): Uma referência ao leilão ao qual a licitação se refere.

ClientId (Chave Estrangeira): Uma referência ao cliente que fez a licitação.

Value: Valor da licitação.

Time: Hora em que o utilizador fez a licitação.

Esses elementos de dados e seus relacionamentos definem a estrutura essencial do nosso sistema de leilões de antiguidades. Os utilizadores fazem licitações em leilões, produtos são oferecidos em leilões específicos e organizados por utilizadores, e cada produto pertence a um vendedor específico, sendo todas as atualizações destas interações organizadas através das notificações.. Essas relações e atributos são fundamentais para o funcionamento do sistema, permitindo que as interações entre as entidades aconteçam de maneira organizada e eficiente.

6.2.6. Relacionamentos

Dentro do nosso modelo lógico para o sistema do *BetterFinds*, várias relações desempenham um papel fundamental na conectividade e funcionalidade do sistema. Estas relações representam as interações complexas entre as entidades envolvidas, tornando o sistema eficaz e dinâmico. Aqui estão as principais relações:

- **Relação "1 Usuário para Muitos Leilões (1:N)":** Esta relação destaca que cada utilizador têm a capacidade de criar e gerenciar vários leilões ao mesmo tempo. Isso é valioso para vendedores ou utilizadores que desejam organizar e supervisionar vários leilões de antiguidades.
- **Relação "1 Leilão para 1 Produto (1:1)":** Esta relação ilustra que cada leilão apenas inclui um produto à venda.
- **Relação "1 Cliente para Muitas Notificações (1:N)":** Esta relação destaca que cada utilizador tem a capacidade de receber várias notificações. Isso é valioso para manter os utilizadores informados sobre atualizações em leilões específicos em tempo real, o que proporciona uma experiência mais completa, e informada.
- **Relação "1 Usuário para Muitas Licitações (1:N)":** Esta relação enfatiza que cada usuário tem a capacidade de fazer várias licitações no mesmo leilão (ou em vários, até em simultâneo). Isso é valioso para participantes ativos nos leilões, permitindo-lhes atualizar a sua licitação sempre que querem.
- **Relação "1 Leilão para Muitas Licitações (1:N)":** Esta relação destaca que cada leilão pode atrair várias licitações de diferentes utilizadores. Isso é crucial para entender a dinâmica do leilão, onde vários participantes podem competir para obter o produto desejado, resultando em múltiplas licitações para um único leilão.
- **Relação "1 Usuário para Muitos Produtos (1:N)":** Nesta relação, destaca-se que cada utilizador tem a capacidade de colocar vários produtos para leilão no seu perfil. Isso é útil para vendedores que desejam oferecer diversos itens simultaneamente, proporcionando uma abordagem flexível na gestão de múltiplos produtos.
- **Relação "1 Leilão para Muitas Notificações (1:N)":** Esta relação ilustra que cada leilão pode gerar várias notificações para os participantes envolvidos. Isso é valioso para manter os utilizadores informados sobre desenvolvimentos importantes durante o leilão, como novas licitações, alterações nos prazos, ou outras informações relevantes. Essas notificações contribuem para uma experiência mais interativa e envolvente.

Podemos dizer que estes relacionamentos modelam a versatilidade do sistema de leilões de antiguidades, permitindo aos utilizadores uma participação ativa em vários leilões, bem como a gestão eficiente dos seus próprios, a listagem de diversos produtos e a realização de licitações em diferentes produtos, tudo isto em simultâneo. Essa abordagem não apenas conecta entidades, mas também cria

uma experiência dinâmica e completa para todos os participantes, promovendo uma interação rica e envolvente no contexto do leilão.

7. Esboço das Interfaces do Sistema

Uma vez que todas as especificações relacionadas à lógica dos leilões foram estabelecidas, o próximo passo envolve a criação de um esquema da interface do utilizador.

7.1. Estrutura Geral das Interfaces do Sistema

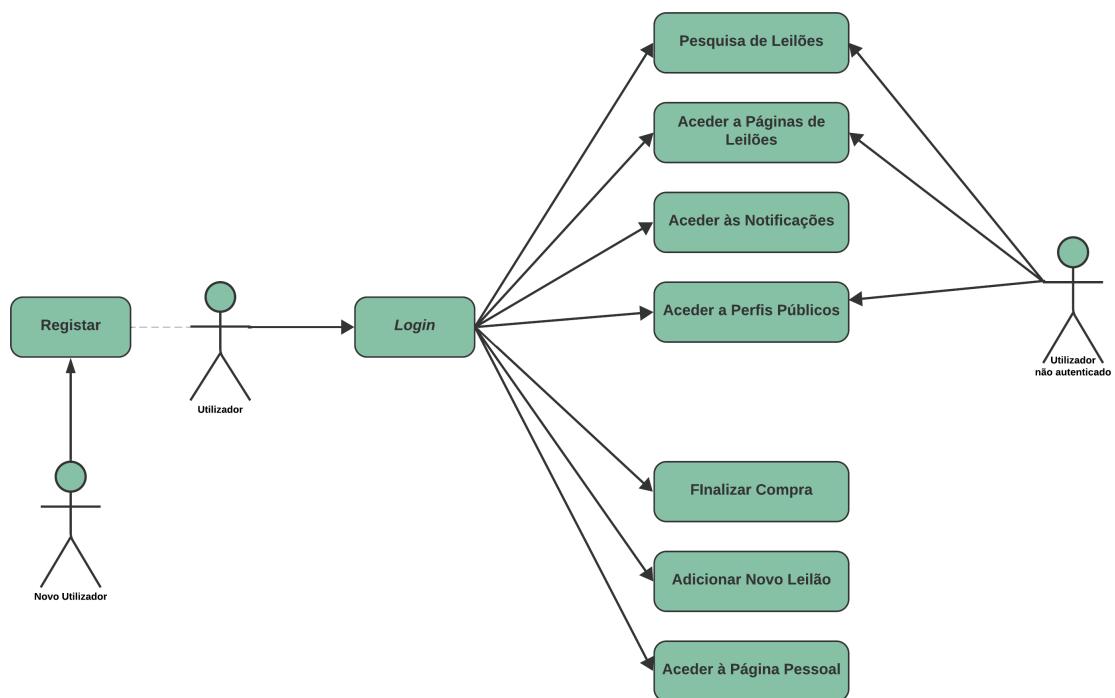


Figura 9: Planeamento da Estrutura das Interfaces do Sistema

7.2. Proposta de Interfaces

Tendo em conta os requisitos funcionais, utilizando o *Procreate*, elaborámos diversos *mockups* das páginas que o nosso site irá suportar e que são necessárias à realização deste projeto, de modo a possuirmos uma base gráfica. Esta base será útil futuramente, aquando a implementação deste projeto.

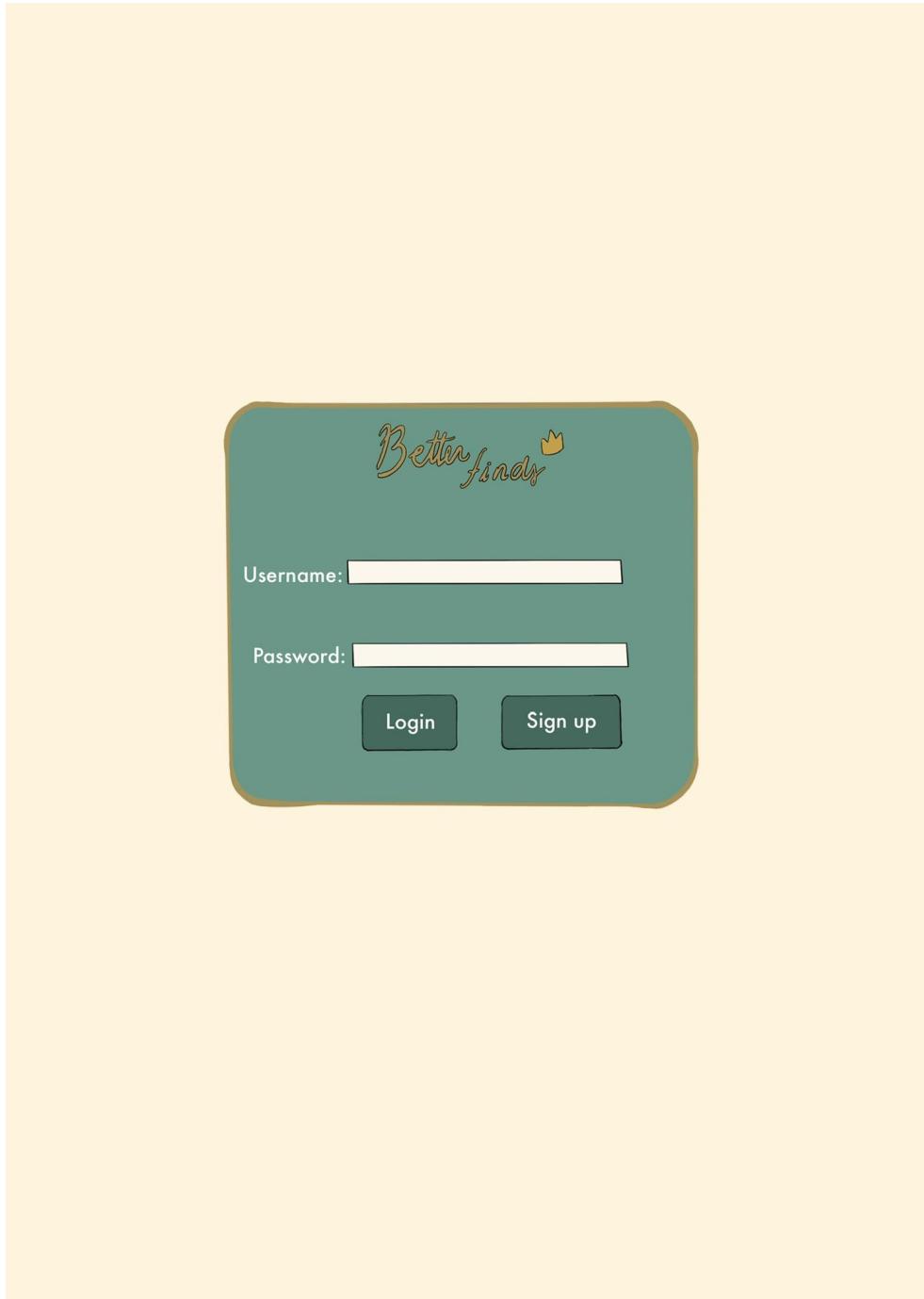


Figura 10: Mockup - Página de Login

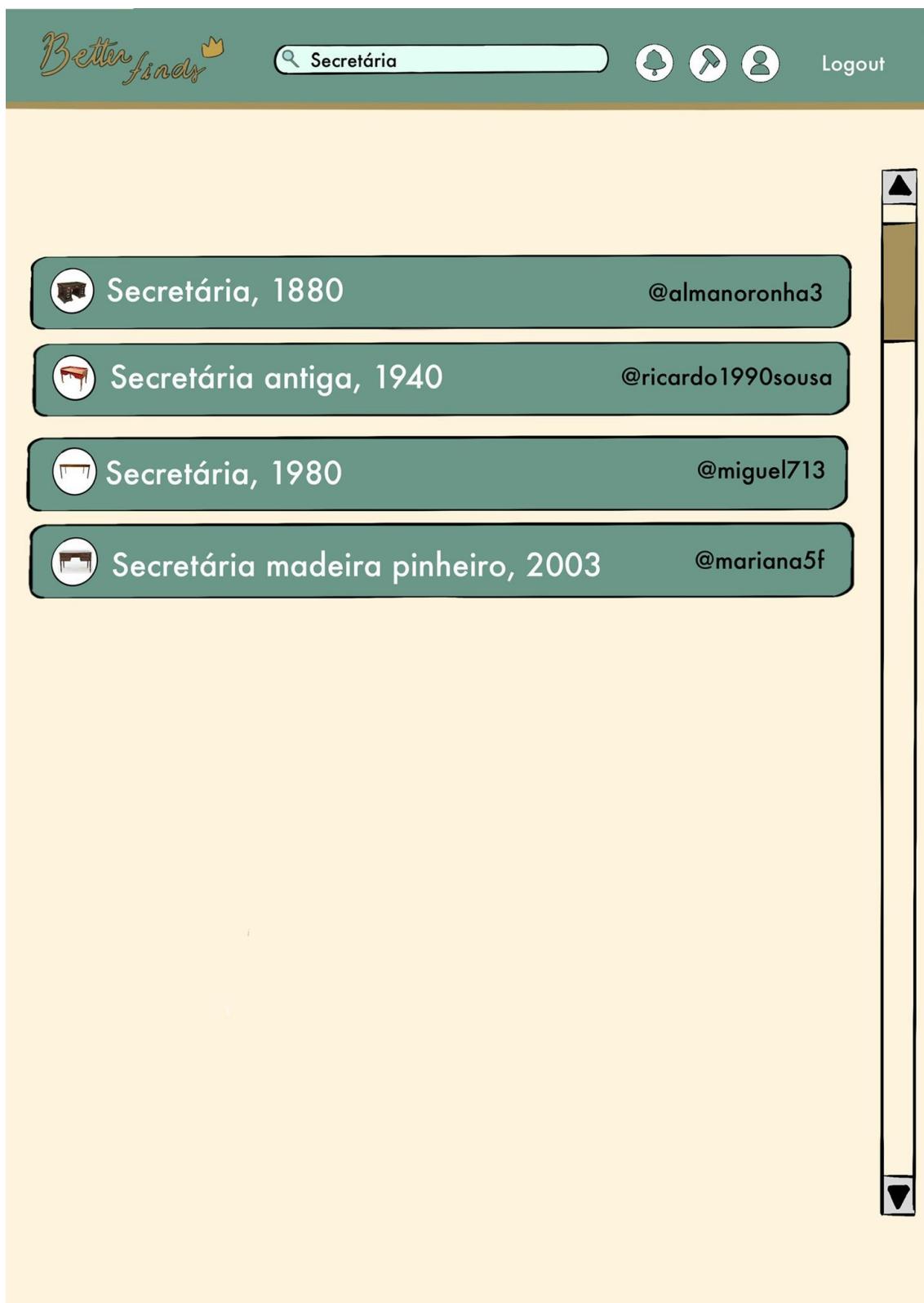


Figura 11: Mockup - Página de Pesquisa

The mockup shows the user registration page for the Betterfinals platform. At the top, there is a header bar with the logo 'Betterfinals' and a search bar. To the right of the search bar are icons for notifications and messages, followed by 'Login | Sign up'. The main title 'Registo de Utilizador' is centered at the top of the form.

Nome completo:
Digite o seu nome completo

Username:
Digite o username

Email:
Digite o email

Telemóvel:
Digite o número de telemóvel

Password:
Digite a password

A placeholder icon of a person is displayed next to the profile photo upload area, which includes a 'Upload foto' button and a 'Escolher ficheiro...' link.

Below the input fields, there is a question: 'Quer receber emails sobre os seus leilões e/ ou licitações?' with two radio buttons: 'Sim' (checked) and 'Não'.

A large green button at the bottom center contains the text 'Criar conta'. Below it, a smaller link says 'Voltar'.

Figura 12: Mockup – Página de Registo do Utilizador

The mockup shows a user profile page for 'Alma Noronha' on the 'Betterfinals' platform. The top navigation bar includes the logo, a search bar, and icons for notifications, bids, and account management, along with a 'Logout' button. The main content area features a circular profile picture of Alma Noronha, her name, and her username 'almanoronha3'. Below this, there are four green buttons: 'Dados Pessoais', 'Os meus leilões', 'As minhas encomendas', and 'As minhas licitações'. The 'As minhas licitações' button is currently selected. The right panel displays 'Dados Pessoais' (Personal Data) with fields for Name (Alma Marques Noronha), Email (almanoronha3@gmail.com), and Phone (+351 911872443). Below this is a section titled 'Os meus leilões' (My Auctions) listing four items: 'Secretária, 1880' (Status: A decorrer), 'Arcade Pinball, 1992' (Status: A decorrer), 'Mona Lisa, 1503' (Status: A decorrer), and 'Relógio de cuco, 2002' (Status: Terminado). The bottom section is titled 'As minhas encomendas' (My Orders) and lists two items: 'Anel YSL, 2009' (Status: Enviado) and 'Corset Vivienne Westwood, 1997' (Status: Entregue).

Dados Pessoais	
Nome completo:	Alma Marques Noronha
Email:	almanoronha3@gmail.com
Telemóvel:	+351 911872443

Os meus leilões	
Secretária, 1880	A decorrer
Arcade Pinball, 1992	A decorrer
Mona Lisa, 1503	A decorrer
Relógio de cuco, 2002	Terminado

As minhas encomendas	
Anel YSL, 2009	Enviado
Corset Vivienne Westwood, 1997	Entregue

Figura 13: Mockup - Página Pessoal do Utilizador

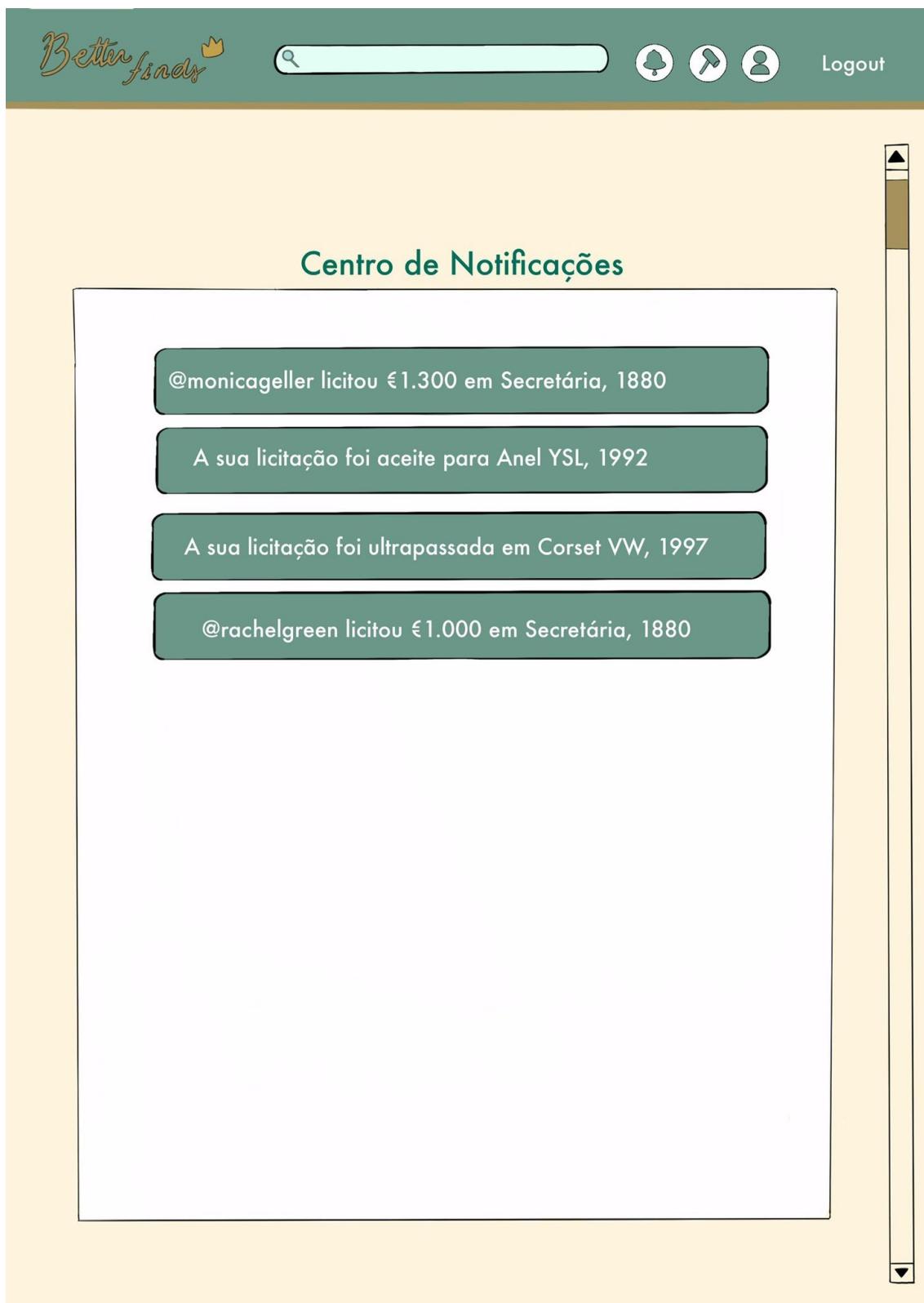


Figura 14: Mockup - Centro de Notificações

The mockup shows a user profile for "Alma Noronha" (almanoronha3). The profile picture is a circular icon featuring a woman's face with dark curly hair and a yellow patterned collar. Below the profile picture, the name "Alma Noronha" and the handle "almanoronha3" are displayed. To the right of the profile picture, there is a section titled "Leilões" (Auctions) containing four items:

- Mona Lisa, 1503** (Icon: Mona Lisa painting) - Status: A decorrer
- Arcade Pinball, 1992** (Icon: Pinball machine) - Status: A decorrer
- Secretária, 1880** (Icon: Secretary desk) - Status: A decorrer
- Relógio de Cucô, 2002** (Icon: Clock) - Status: Terminado

The top navigation bar includes the logo "BetterFinals", a search bar, and icons for notifications, messages, and user profile, along with a "Logout" button.

Figura 15: Mockup - Página Perfil Público

The mockup displays an active auction for a "Secretária" (desk) from 1880. The item is shown in a circular inset. The main title is "Secretária" with a minimum bid of €500. The auction ends in 6 days, 22 hours, 55 minutes, and 10 seconds. The current bid is €1.250, with a button to bid "€1.260 ou superior". A history of bids is listed:

Bid	Licitador	Há
€1.250	5470	9 horas
€1.200	5914	13 horas
€1000	3296	14 horas
€850	2500	18 horas
€500	4521	20 horas
	2700	

Figura 16: Mockup - Página de Leilão Ativo

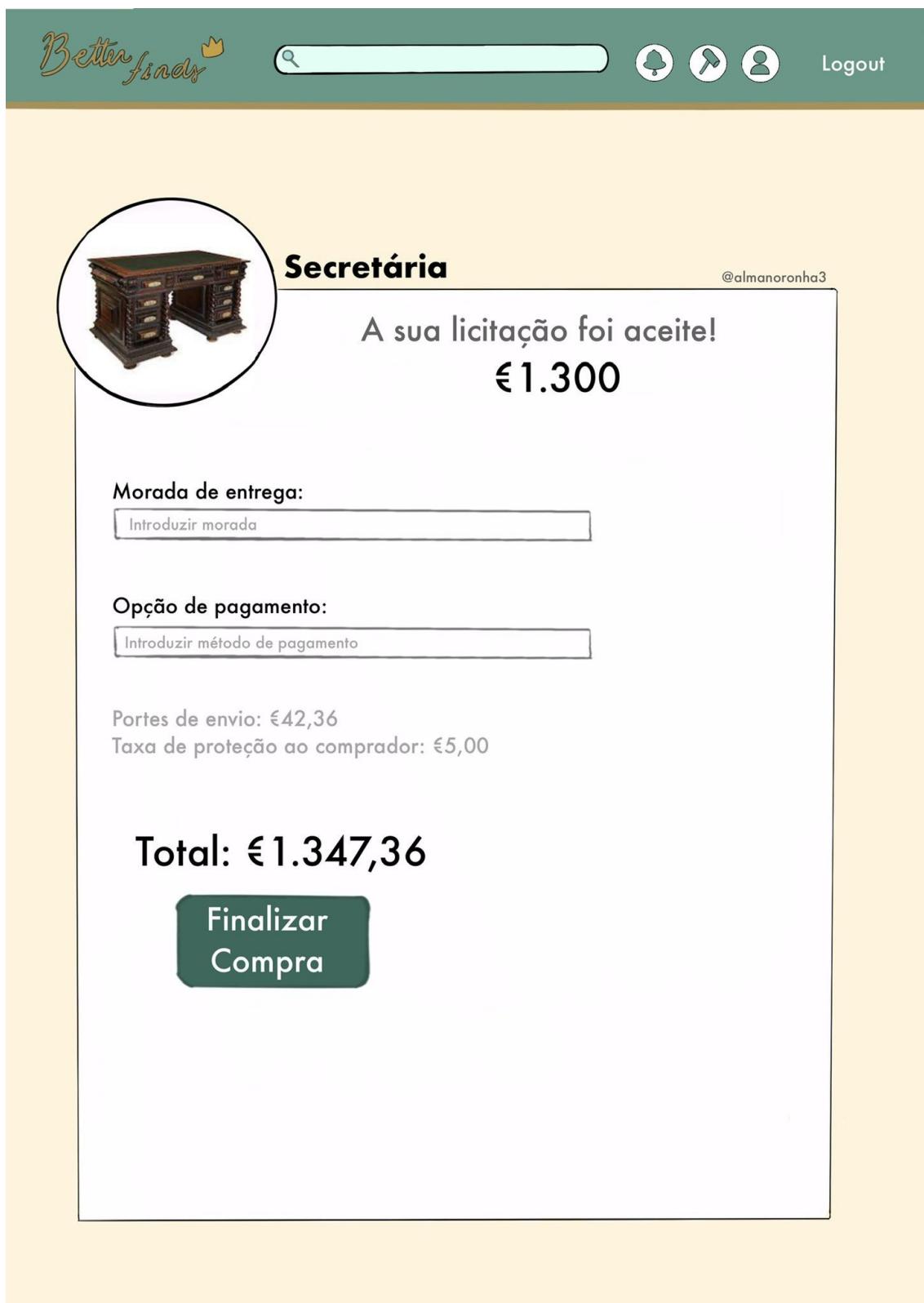


Figura 17: Mockup - Página de finalização de compra



Figura 18: Mockup - Página de Criação de Leilão

8. Metodologia e Ferramentas de Implementação

8.1. Backend

A principal ferramenta que escolhemos utilizar para desenvolver a nossa aplicação foi a *framework* de desenvolvimento **ASP.NET Core Web App (Razor Pages)**. Esta escolha foi tomada devido à sua longa lista de vantagens:

- Sintaxe simples e de fácil compreensão;
- Organização de código modular, isto é, cada ficheiro contém o elemento de *UI* (*.cshtml*), bem como o *backend* na linguagem escolhida para o projeto, C# (*.cshtml.cs*), tornando a sua compreensão e manutenção mais fácil;
- É fortemente integrado com *ASP.NET Core*, o que nos providencia funcionalidades como *dependency injection* e *middleware*;
- Providencia um “elo” de ligação entre o *input* do utilizador no *frontend* com o *backend*, facilitando o processo de validação e processamento de dados;
- Performance e testabilidade.

Apesar de não ser tão flexível e ter um ecossistema tão abrangente como *MVC (Model-View-Controller)*, consideramos que *Razor* foi uma escolha adequada para o desenvolvimento deste projeto, uma vez que tivemos uma boa experiência com a utilização deste *framework*.

Quanto à criação e gestão da nossa base de dados, a utilização do **Microsoft SQL Server** foi a escolha mais adequada devido às razões que se apresentam:

- Alta escabilidade, o que permite expandir a nossa base de dados vertical ou horizontalmente para lidar com cargas de trabalho e volumes de dados crescentes;
- Otimização de Desempenho: Inclui funcionalidades como otimização de *queries* e planos de execução, contribuindo para a otimização do desempenho das consultas à base de dados, garantindo recuperação eficiente de dados;
- Segurança: Possui robustas funcionalidades de segurança, como encriptação e mecanismos de autenticação, para salvaguardar a integridade e confidencialidade dos dados;

- **Integração com os outros softwares:** O Microsoft SQL Server permite exportação a partir da aplicação *Visual Paradigm*, onde efetuámos o Modelo Lógico, e tem uma integração perfeita com o *framework .NET*, o que é extremamente útil no caso do nosso projeto;
- A linguagem *T-SQL* é muito abrangente, rica em funcionalidades e fornece uma ampla gama de ferramentas para manipulação de dados, recuperação e programação procedural.

8.2. *Frontend*

Para a criação da interface do produto final, não utilizamos nenhum *framework*, apenas **CSS**, **HTML** e **JavaScript**.

8.3. *Servidor*

Utilizámos a ferramenta **Nginx** como *Reverse Proxy* para redirecionar o projeto para as seguintes portas : 80 (*http*) e 443 (*https*).

Recorremos ao **amen.pt** para registar o nosso domínio (*betterfinds.pt*), e à **Cloudflare** para otimizar o desempenho e fornecer segurança adicional.

Utilizámos o **CertBot** para a criação e gestão do certificado SSL utilizado para o domínio.

8.4. *Testagem e Segurança*

Utilizámos diversos *browsers* (*Chrome*, *Firefox*, *Safari*) numa multíitude de plataformas, para garantir que o website mantinha uma boa aparência independentemente de onde fosse acessada pelo utilizador.

Recorremos à **LambdaTest**, uma plataforma de testes de compatibilidade em vários navegadores e dispositivos, para automatizar a testagem do nosso programa e garantir a inexistência de erros.

A nível de segurança, utilizámos **DeepSource**, uma ferramenta para analisar e melhorar a qualidade do código, e **Snyk**, que é utilizado para verificar e corrigir vulnerabilidades de segurança no código.

A testagem e segurança serão exploradas com mais detalhe em capítulos posteriores.

9. Conexão da Base de Dados

Utilizando o esquema desenvolvido na fase de especificação, o grupo foi capaz de gerar uma BD no Microsoft SQL Server. Assim, tendo já a base criada, apenas foi necessário efetuar o seu povoamento. Este contou com o registo de um utilizador *test*, que seria usado para criar leilões teste.

A nossa equipa decidiu não adicionar mais entidades pois, como fomos desenvolvendo a base de dados em paralelo com o website, preferimos registrar novos utilizadores através deste, bem como deixá-lo povoar a base de dados ao longo do tempo. Além disso, esta decisão garantir-nos-ia uma versão mais fidedigna de como seria futuramente a interação de um utilizador a criar um leilão, a introduzir uma licitação, etc.

Posto isto, as figuras seguintes incluem as tabelas referentes à nossa base de dados.

File: dbo.Auction.sql	
1	<code>CREATE TABLE [dbo].[Auction] (</code>
2	<code> [AuctionId] INT NOT NULL,</code>
3	<code> [ClientId] INT NOT NULL,</code>
4	<code> [ProductId] INT NOT NULL,</code>
5	<code> [StartTime] DATETIME NOT NULL,</code>
6	<code> [EndTime] DATETIME NOT NULL,</code>
7	<code> [MinimumBid] MONEY NOT NULL,</code>
8	<code> [IsCheckHasEnded] BIT DEFAULT ((0)) NOT NULL,</code>
9	<code> [IsCompleted] BIT DEFAULT ((0)) NOT NULL,</code>
10	<code> [PaymentMethod] NVARCHAR (32) NULL,</code>
11	<code> PRIMARY KEY CLUSTERED ([AuctionId] ASC)</code>
12	<code>);</code>

Figura 19: Tabela Entidade "Auction"

File: dbo.Bid.sql	
1	<code>CREATE TABLE [dbo].[Bid] (</code>
2	<code> [BidId] INT NOT NULL,</code>
3	<code> [AuctionId] INT NOT NULL,</code>
4	<code> [ClientId] INT NOT NULL,</code>
5	<code> [Value] MONEY NOT NULL,</code>
6	<code> [Time] DATETIME NOT NULL,</code>
7	<code> PRIMARY KEY CLUSTERED ([BidId] ASC)</code>
8	<code>);</code>

Figura 20: Tabela Entidade "Bid"

	File: dbo.Client.sql
1 2 3 4 5 6 7 8 9 10 11	<pre> CREATE TABLE [dbo].[Client] ([ClientId] INT NOT NULL, [FullName] NVARCHAR (64) NOT NULL, [Username] NVARCHAR (32) NOT NULL, [Email] NVARCHAR (320) NOT NULL, [Password] NVARCHAR (64) NOT NULL, [ProfilePic] NVARCHAR (256) NULL, [OptNewsletter] BIT DEFAULT ((0)) NOT NULL, PRIMARY KEY CLUSTERED ([ClientId] ASC), UNIQUE NONCLUSTERED ([Username] ASC, [Email] ASC)); </pre>

Figura 21: Tabela Entidade "Client"

	File: dbo.Notification.sql
1 2 3 4 5 6 7 8 9	<pre> CREATE TABLE [dbo].[Notification] ([NotificationId] INT NOT NULL, [ClientId] INT NOT NULL, [AuctionId] INT NOT NULL, [Message] NVARCHAR (256) NOT NULL, [Timestamp] DATETIME NOT NULL, [IsRead] BIT DEFAULT ((0)) NOT NULL, PRIMARY KEY CLUSTERED ([NotificationId] ASC)); </pre>

Figura 22: Tabela Entidade "Notification"

	File: dbo.Product.sql
1 2 3 4 5 6 7 8 9 10	<pre> CREATE TABLE [dbo].[Product] ([ProductId] INT NOT NULL, [AuctionId] INT NOT NULL, [ClientId] INT DEFAULT ((0)) NOT NULL, [Name] NVARCHAR (64) NOT NULL, [Description] NVARCHAR (2048) NOT NULL, [Price] MONEY NOT NULL, [Images] NVARCHAR (2048) NULL, PRIMARY KEY CLUSTERED ([ProductId] ASC)); </pre>

Figura 23: Tabela Entidade "Product"

10. Desenvolvimento

10.1. Gestão de Logins

Com o intuito de facilitar a gestão de um utilizador, foi implementado um sistema de autenticação baseado em *cookies* na aplicação web ASP.NET Core. Após um *login* bem-sucedido, são geradas reivindicações (*claims*) que contêm informações do utilizador, como o seu nome de utilizador e identificador único (*client ID*), e, de seguida, estas reivindicações são organizadas numa identidade.

Adicionalmente, configuraram-se propriedades de autenticação, permitindo a atualização da *cookie* e a opção de persistência entre sessões, indicada pelo parâmetro "*RememberMe*", que permanece válido por 7 dias.

Este método de gestão de logins visa assegurar a segurança e proporcionar uma experiência contínua ao utilizador, prevenindo acessos não autorizados às páginas privadas.

10.2. Diretoria Utils

Com objetivo de agilizar a reutilização de certas partes do código foi criada uma diretoria nomeada "Utils" que garante métodos criados para serem utilizados globalmente, nos contextos apresentados de seguida.

10.2.1. Leilões (Auctions.cs)

Esta classe proporciona um método *GetAuctions* que recebe parâmetros que afetam a forma como a lista de leilões é retornada, variando em ordem (por data de encerramento, nome e preço do produto), se um leilão se encontra a decorrer e, opcionalmente, pelo vendedor de leilão.

Este método provou a sua utilidade através de uma fácil obtenção de leilões, bem como na reorganização dos mesmos na interface de utilizador.

Esta classe de utilidades relacionadas com leilões também inclui métodos relacionados a um serviço assíncrono que é inicializado antes do começo da aplicação web, e que visa monitorizar leilões que

tenham finalizado, reagindo através de notificações encaminhadas a utilizadores relacionados ao leilão através no sistema de notificações.

10.2.2. Licitações (*Bids.cs*)

Esta classe agiliza o processo de envio de notificações, contendo métodos para gestão de grupos de licitadores. Deste modo, torna-se mais fácil o processo de envio de notificações a grupos de utilizadores relevantes a um determinado leilão.

10.2.3. Cliente (*Client.cs*)

A classe Cliente providencia um método para obter o *clientId* através da sua sessão caso seja possível; Se não o for, obtém-no através da pesquisa na base de dados pelo seu nome de utilizador, uma chave da tabela *sql Client*, este também guardado na sessão. Adicionalmente, contém um método para obter uma lista em formato dicionário de clientes para poupar múltiplas travessias à base de dados.

10.2.4. Notificações (*Notification.cs*)

Engloba métodos relativos a gestão, organização e criação de notificações, tais como: criar uma notificação na base de dados, marcar uma ou todas as notificações como lidas, obter todas as notificações ou apenas as que estão por ler e contagem de notificações não lidas.

10.2.5. Imagens (*Images.cs*)

Inclui métodos relativos a validação, obtenção e travessia/transformação de imagens guardadas em formato de *link*, tanto para a foto de perfil do utilizador, como para imagens de leilões.

10.2.5. Valores Monetários (*Currency.cs*)

Utilidades para a apresentação de valores decimais monetários apresentados no UI.

11. Segurança e Manutenção

A segurança na nossa aplicação é de vital importância. Proteger os dados dos utilizadores e garantir a integridade do sistema são prioridades essenciais para mitigar riscos de cibersegurança, assegurando uma experiência de utilização confiável e estável.

Em geral, o *framework* ASP.NET Core forneceu uma excelente base para a prevenção de vulnerabilidades por definição.

Entrando em nível mais aprofundado sobre algumas das prevenções tomadas, começámos pelas *SQL Injections* - uma vulnerabilidade onde um atacante é capaz de executar *SQL queries* através da aplicação, comprometendo os dados e integridade da base de dados.

A nível de ataques de *Cross-Site Scripting (XSS)*, por definição o ASP.NET Core é blindado a ataques através da técnica de escapamento de caracteres especiais, não obstante, é necessário tomar precauções adicionais, pois, por exemplo, o uso do método *Html.Raw* pode causar a ocorrência desta vulnerabilidade quando aplicado a dados introduzidos pelo usuário que são dispostos na aplicação web.

Exemplo de cross-site scripting (XSS) não persistente (ou refletido):

Um exemplo deste ataque na nossa aplicação seria na página de pesquisa – um atacante poderia acionar uma pesquisa que, caso esta fosse apresentada na página de resultados de uma forma insegura, causaria o código malicioso a ser executado na página de resultados.

Exemplo de cross-site scripting (XSS) persistente (ou armazenado):

Uma maneira de um ator malicioso utilizar esta vulnerabilidade na nossa aplicação seria através da página de notificações, onde utilizámos o método *Html.Raw* descrito anteriormente para apresentar as mensagens das notificações. Ele poderia utilizar esta vulnerabilidade através da criação de um nome de utilizador com um XSS payload e da participação em um leilão que causasse a este e outros utilizadores uma notificação que inclua o seu nome, neste modo acionando esta vulnerabilidade de forma persistente na aplicação web.

Também tiramos proveito do uso de um certificado SSL com o nosso domínio `betterfinds.pt` (e subdomínio `www.betterfinds.pt`), o que permite encriptação de dados de ponta-a-ponta, evitando assim ataques *Man-in-the-middle* (MITM).

Para uma extra garantia na segurança da aplicação recorremos a ferramentas de terceiros, tais como `Snyk` e `DeepSource`.

O `Snyk` é uma plataforma de segurança que identifica e corrige vulnerabilidades em dependências de código aberto, fortalecendo a segurança das aplicações. A plataforma permitiu encontrar vulnerabilidades em duas bibliotecas .NET que se encontravam deprecadas, no entanto não eram conhecidos *exploits* até a data. De qualquer maneira, estas foram corrigidas ao utilizar as últimas versões estáveis através do gerenciador de pacotes `NuGet`.

Project	Imported	Tested	Issues
<code>Code analysis</code>	3 days ago	5 hours ago	0 C 0 H 0 M 0 L
<code>migueltc13/project-li4:BetterFinds/BetterFinds.csproj</code> net8.0	3 days ago	6 hours ago	0 C 0 H 0 M 0 L

Figura 24: Análise do código no Snyk

O `DeepSource` é uma plataforma que oferece análises estáticas de código, identifica problemas de qualidade e segurança, facilitando a criação de software robusto e sustentável, o que nos permitiu agilização em certos elementos do código fonte.

Em ambas as plataformas **não** foram encontradas vulnerabilidades através da análise do código.

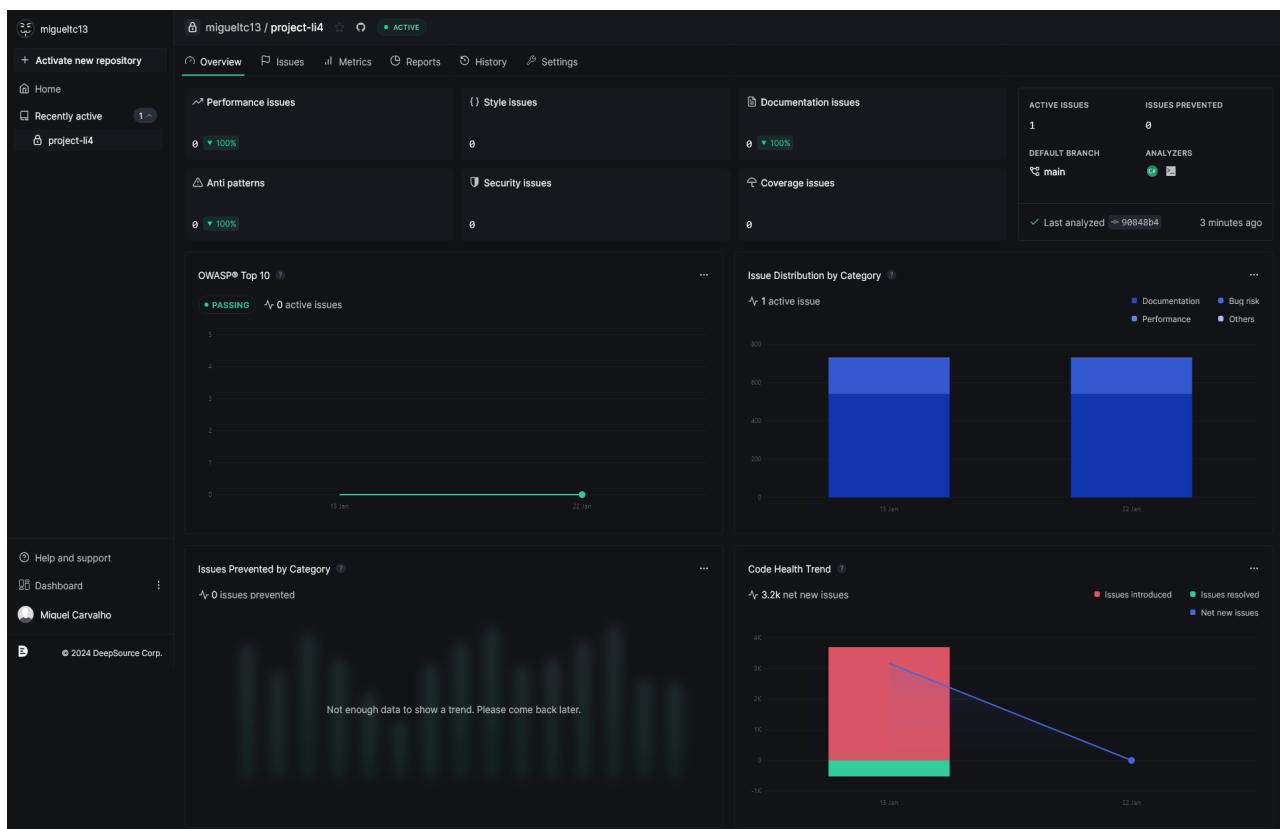


Figura 25: Análise do código no DeepSource

Em suma, embora o *framework* ofereça segurança robusta, reconhecemos a importância de precauções adicionais, especialmente em casos específicos, como na manipulação de dados de *input* do utilizador. A utilização de certificado SSL e ferramentas de terceiros, como *Snyk* e *DeepSource*, complementou a nossa abordagem, identificando e corrigindo potenciais falhas. Outra segurança que não foi implementada, porém consideramos indispensável num cenário de produção, é a conversão de palavras passe, com, por exemplo, *Secure Hashing Algorithms (SHA)*, em todo o caminho digital percorrido pelas mesmas, desde o *browser* do cliente até à base de dados.

12. Testes

No decorrer do desenvolvimento deste projeto, a fase de testagem foi integrada de forma crucial. Além dos testes manuais contínuos, foi implementado um projeto adicional de testagem automática para as funcionalidades da aplicação web principal.

Para tal, foram utilizadas as bibliotecas **NUnit** e **Selenium** na linguagem C#. O *Nunit* é um *framework* de testes unitários que simplifica a criação e execução de testes, enquanto o *Selenium* é uma ferramenta de automação de testes para aplicações web, possibilitando interações automatizadas com a interface do utilizador.

Foi também disponibilizada a integração com aplicação web **LambdaTest** para permitir testes paralelos, criação de túneis para testes locais e monitorização completa através de *logs*, vídeos e *screenshots*.

Foram conduzidos testes unitários abrangendo as seguintes funcionalidades:

- Registo;
- Login;
- Criar leilão;
- Editar leilão;
- Pesquisar leilões;
- Litar num leilão;
- Terminar um leilão de diversas formas:
 - “Naturalmente” (ao aguardar pelo seu fim);
 - Atempadamente com comprador;
 - Atempadamente sem comprador.
- Notificações:
 - Nova licitação (perspetiva de um licitador e do vendedor);
 - Leilão terminado “naturalmente” (perspetiva do comprador, de um outro licitador e do vendedor);
 - Leilão terminado atempadamente com comprador (perspetiva do comprador, de um outro licitador e do vendedor);
 - Leilão terminado atempadamente sem comprador (perspetiva do vendedor).
- Pagamento efetuado pelo vencedor do leilão.

Foi ainda criado um teste principal que executa todas as funcionalidades descritas acima de forma sequencial.

A escolha de utilizar a mesma linguagem que o projeto principal trouxe benefícios, como consistência no código, verificação direta de operações na base de dados e facilidade de colaboração entre membros da equipa.

Em conclusão, a abordagem robusta de testagem desempenha um papel crucial na garantia de qualidade do *software*, assegurando que todas as funcionalidades operam como esperado e contribuindo para a confiabilidade geral do sistema.

13. Arquitetura Final do Projeto

Após a implementação da aplicação, a arquitetura final tem o seguinte aspeto:

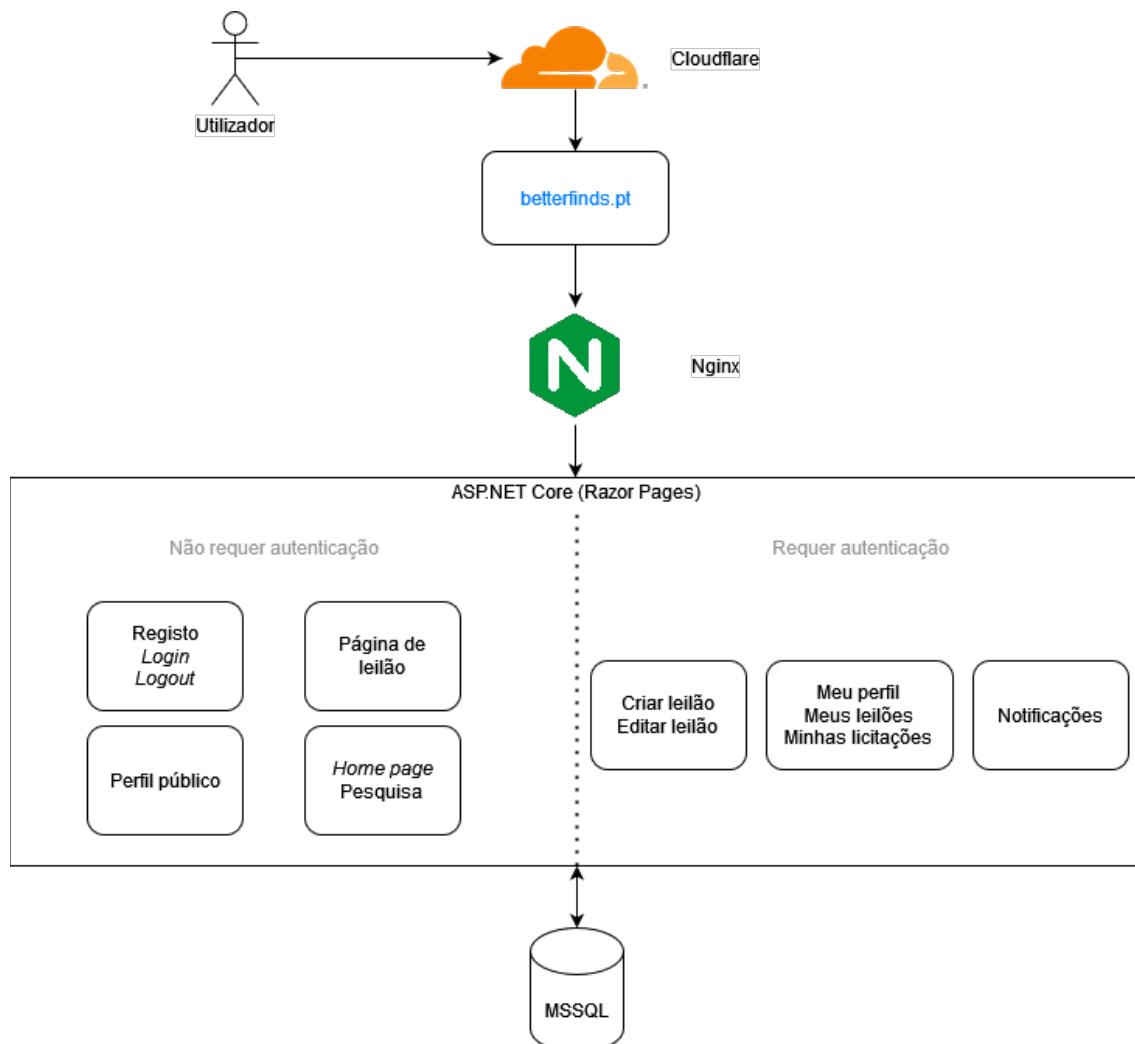


Figura 26: Arquitetura Final do Projeto

A arquitetura da aplicação *BetterFinds* foi projetada para oferecer uma experiência intuitiva e segura aos utilizadores. Por conseguinte, o utilizador inicia a interação ao acessar o domínio *betterfinds.pt* através de um *browser*. A requisição do utilizador é encaminhada através do serviço *Cloudflare*, que atua como um serviço de CDN e oferece otimizações de segurança, como proteção contra ataques DDoS e SSL/TLS,

e redireciona o tráfego para os servidores da aplicação. Para além disso, utilizamos o *Cloudflare* para a criação de uma regra *CNAME* para o subdomínio: www.betterfinds.pt.

O *Nginx* atua como um *reverse proxy*, encaminhando essas requisições para o *backoffice* da aplicação, garantindo, deste modo, uma distribuição eficiente do tráfego.

No *backoffice*, a lógica da aplicação é desenvolvida em *ASP.NET Core*, utilizando o modelo *Razor Pages*, cujas vantagens explorámos em capítulos anteriores.

Caso sejam pedidos que necessitem de autenticação, antes de serem encaminhados para o serviço correto, o utilizador é encaminhado para a página de *login* para iniciar o processo de autenticação.

Por fim, os dados da aplicação são armazenados num banco de dados *MSSQL*, integrado à infraestrutura da aplicação. Este sistema de gerenciamento de banco de dados relacional permite armazenar, recuperar e gerenciar os dados de forma eficiente e segura.

14. Produto Final

Em comparação com os *mockups*, sentimos que conseguimos desenvolver o que havia sido apresentado previamente. Por outro lado, decidimos evoluir o *design* para algo mais sofisticado, pois sentimos que se encaixava melhor com o tema de leilões de antiguidades.

14.1. Registar

O registo no sistema é efetuado a partir da página *register*, onde o utilizador vai introduzir o nome completo (entre 3 a 64 caracteres alfabéticos e espaços), o *username* (entre 3 a 32 caracteres alfanuméricos), o *email* (entre 5 a 320 caracteres), *password* e *confirm password* (ambas entre 8 a 64 caracteres), foto de perfil (campo opcional entre 0 a 256 caracteres), e subscrever a *newsletter* (*checkbox*).

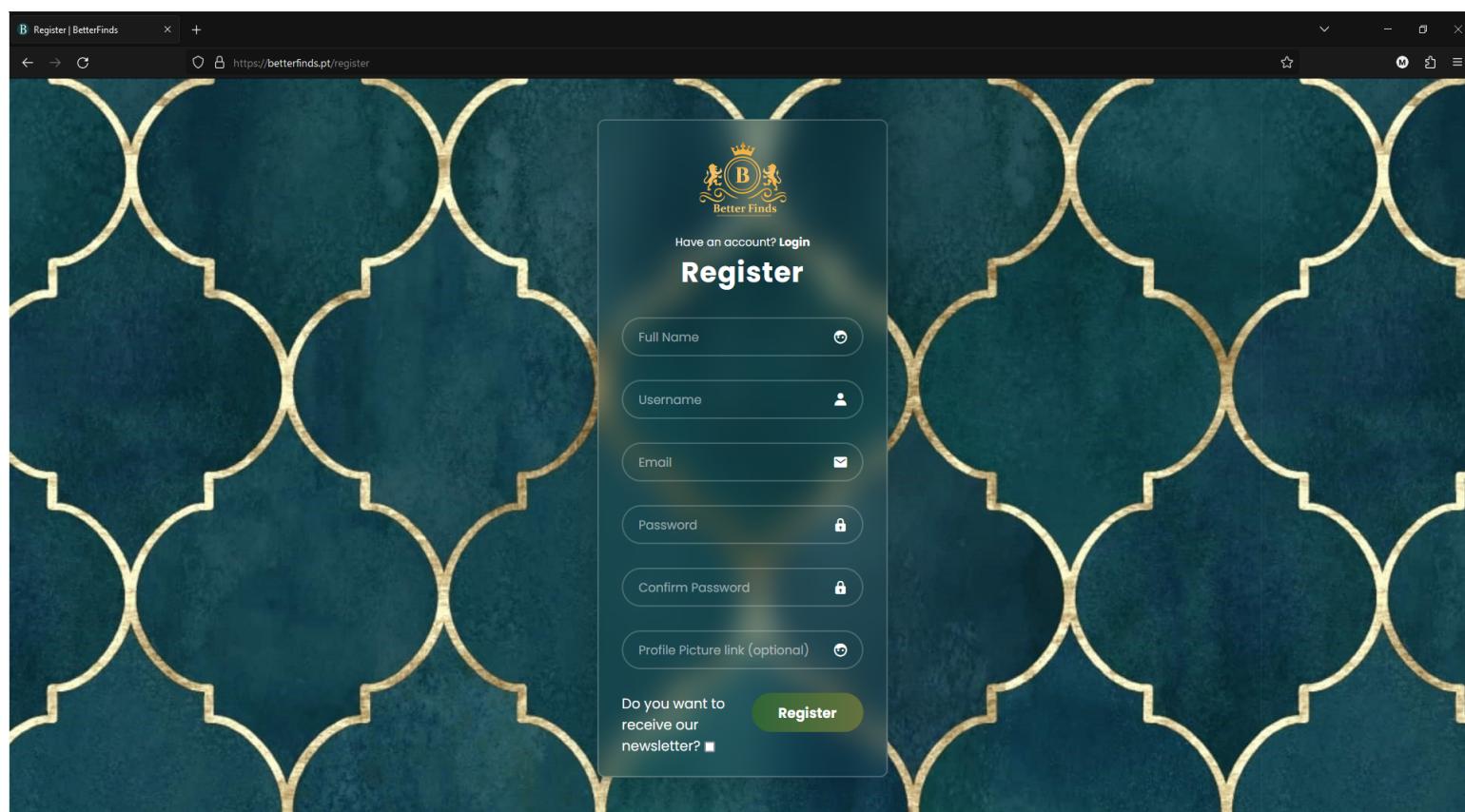


Figura 27: Página de Registo

Tanto os campos de *email* e de *username* têm de ser únicos em relação aos respetivos dados guardados posteriormente na base de dados, esta verificação é definida tanto da base de dados como na lógica do *backend*. Se o utilizador falhar ao cumprir alguma destas restrições, será apresentada na página de registo uma mensagem de erro com a restrição em questão. Após um registo com sucesso, o utilizador é redirecionado para a página de *login* onde é apresentada uma mensagem informativa que reafirma a criação da conta.

14.2. Login

O *login* no sistema é efetuado a partir da página *login*, onde o utilizador tem que introduzir o seu *username* e *password*, e têm que ser iguais aos da base de dados. Se o utilizador falhar esta restrição, será apresentada uma mensagem de erro na página: "*Invalid username or password.*"

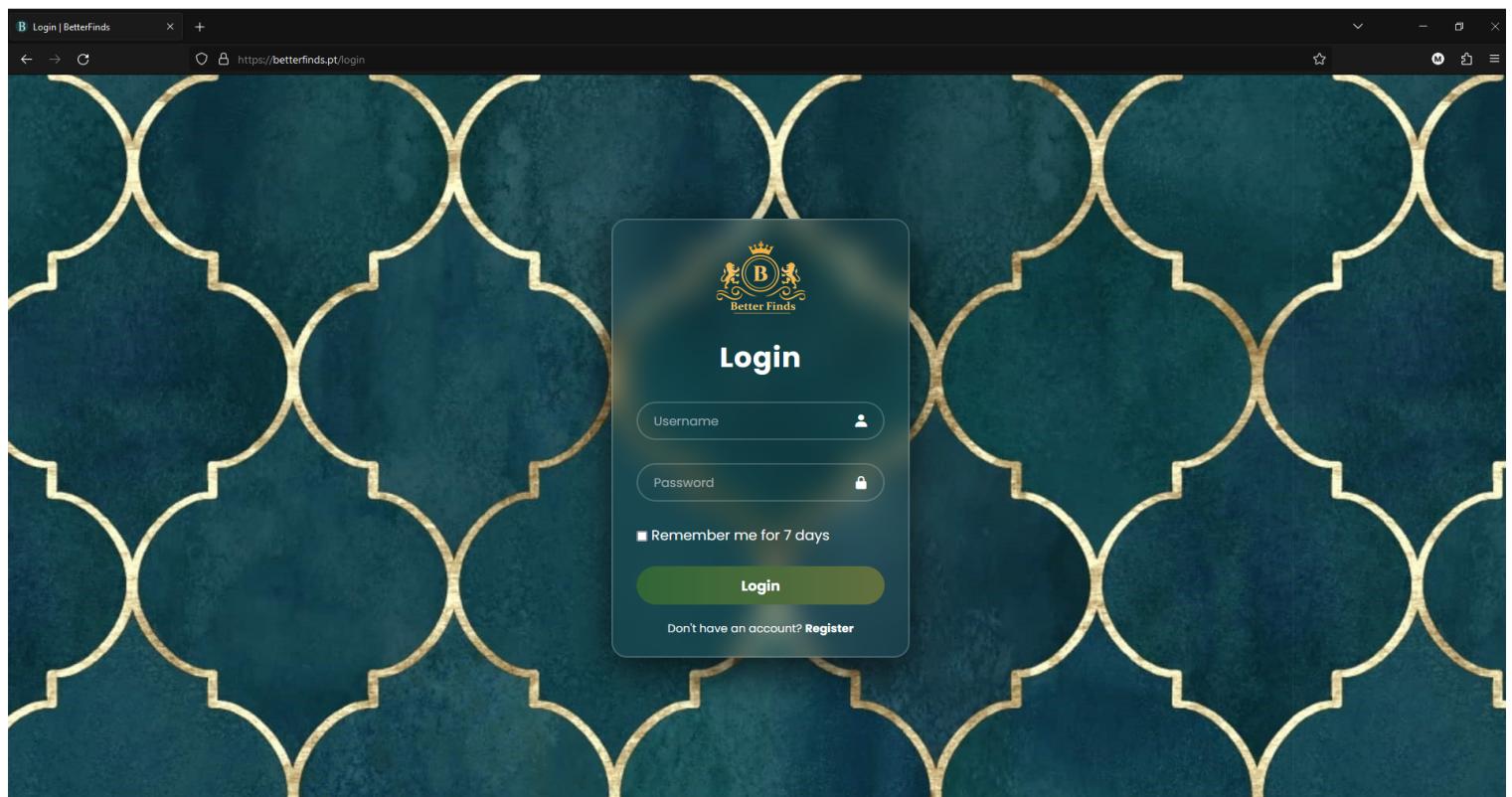


Figura 28: Página de Login

Após o *login* ser efetuado com sucesso, será guardada de forma persistente uma *cookie* que guarda a sessão do utilizador válida por 7 dias, isto caso o utilizador selecionar a opção "Remember me for 7 days". Caso o utilizador não selecionar a mesma, a sessão é valida por 30 minutos ou até fechar a sessão (o que vier primeiro).

Após a criação da sessão anterior o utilizador é redireccionado para a *Homepage* onde é apresentada uma mensagem "Welcome @[username]", que garante ao utilizador o sucesso do *login*. São guardados

o id do cliente e o seu *username* na cookie criada no intuito de facilitar operações futuras que envolvam estes dados.

14.3. Logout

O procedimento de *logout* pode ser ativado pelo utilizador ao clicar no botão "Logout" localizado na sidebar presente em todas as páginas, ou por se dirigir ao link: *http(s)://[url]/logout*.

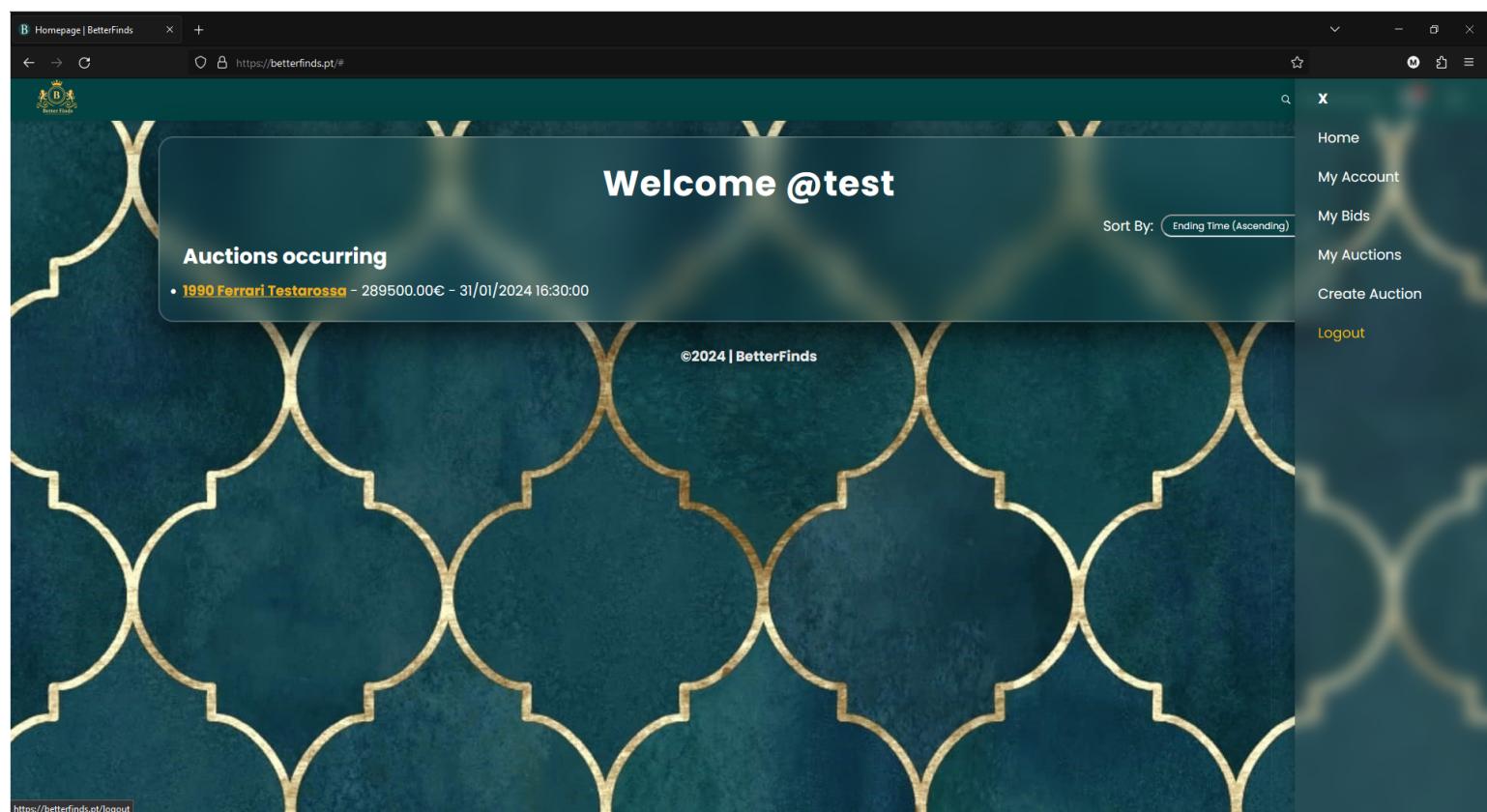


Figura 29: Sidebar com botão Logout

Uma vez acionado, o utilizador será redirecionado para a página de *login* onde é apresentada uma mensagem informativa de que efetuou *logout* com sucesso, como se pode ver na figura seguinte.

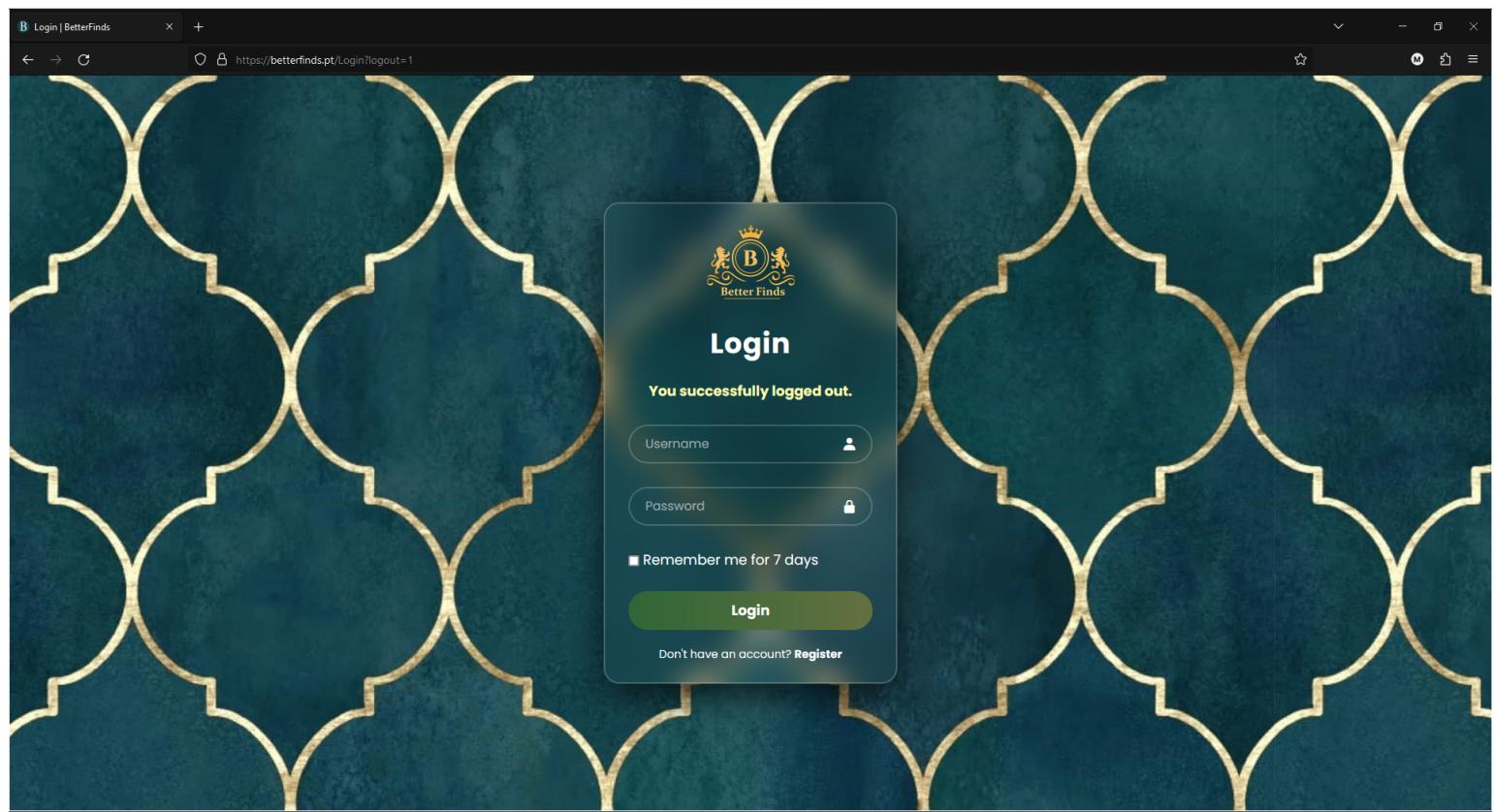


Figura 30: Redirecionamento após Logout

14.4. Homepage

Na página *Homepage/Index*, à qual o utilizador é redirecionado assim que efetua *login*, são apresentados os leilões que se encontram a decorrer.

O utilizador tem as seguintes opções:

- Organizar os leilões presentes na página (por preço, data de término e nome de produto);
- Pesquisar leilões (por nome de produto ou parte da descrição, estando a ferramenta de *sort by* disponível também);
- O menu de sidebar (*Home*, *My Account*, *My Bids*, *My Auctions*, *Create Auction* e *Logout*).

Como foi mencionado anteriormente, cada *Homepage* é personalizada com a mensagem "*Welcome @username*", sendo a página de convidado (utilizadores não registados) "*Welcome guest*".

Sempre que existe uma atualização ou a criação de um leilão, esta página é automaticamente recarregada.

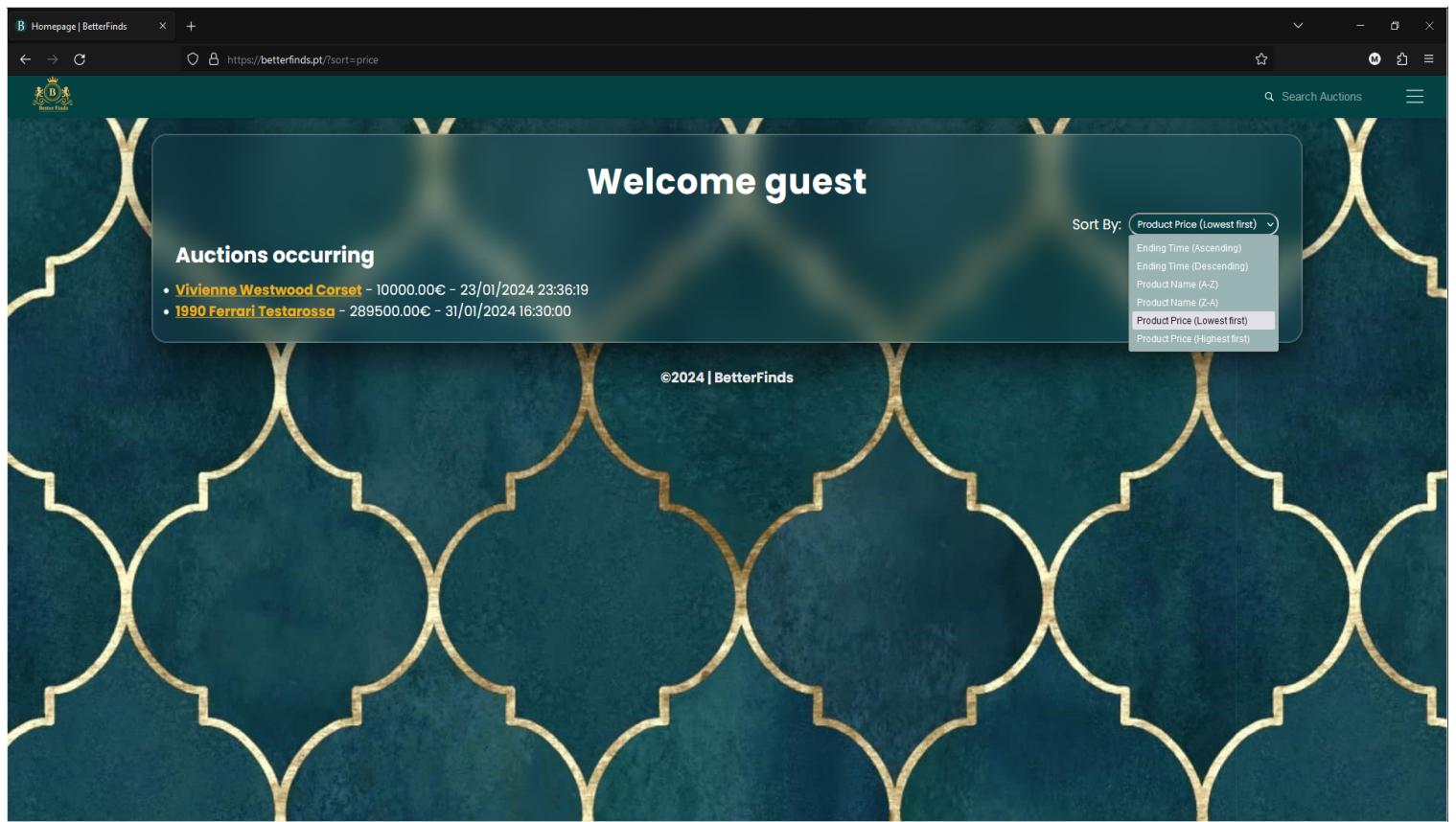


Figura 31: Homepage

14.5. My Account

Na página *My Account* – à qual o utilizador tem acesso ao clicar no botão "My Account" localizado na sidebar presente em todas as páginas –, ele tem disponibilizados os dados pessoais que introduziu quando se registou no nosso website, e tem a opção de alterá-los, tendo em conta as mesmas restrições descritas na página *register*. Além disso, o utilizador ainda tem acesso às suas estatísticas, isto é, aos seus leilões criados e licitações feitas.

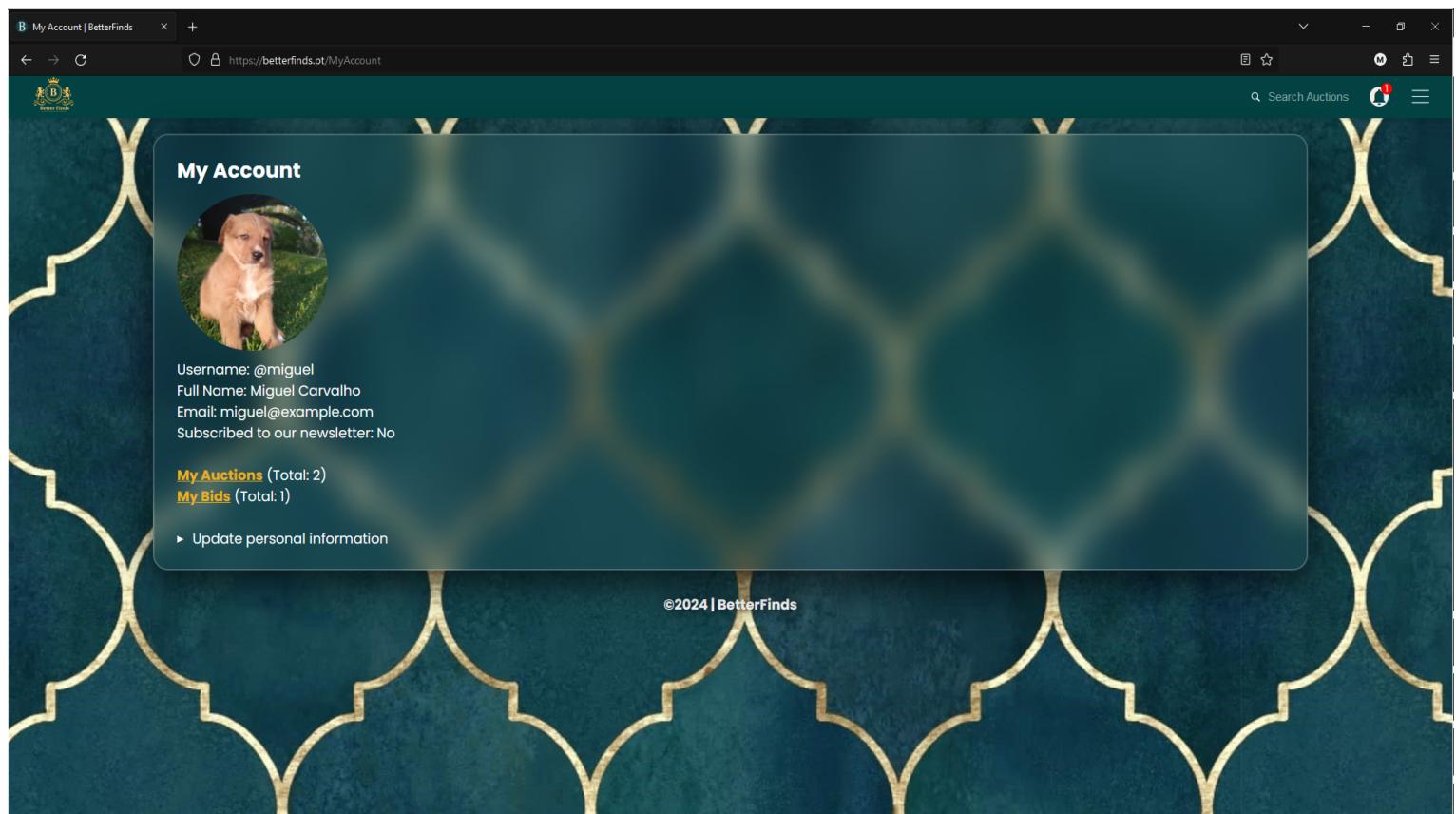


Figura 32: My Account

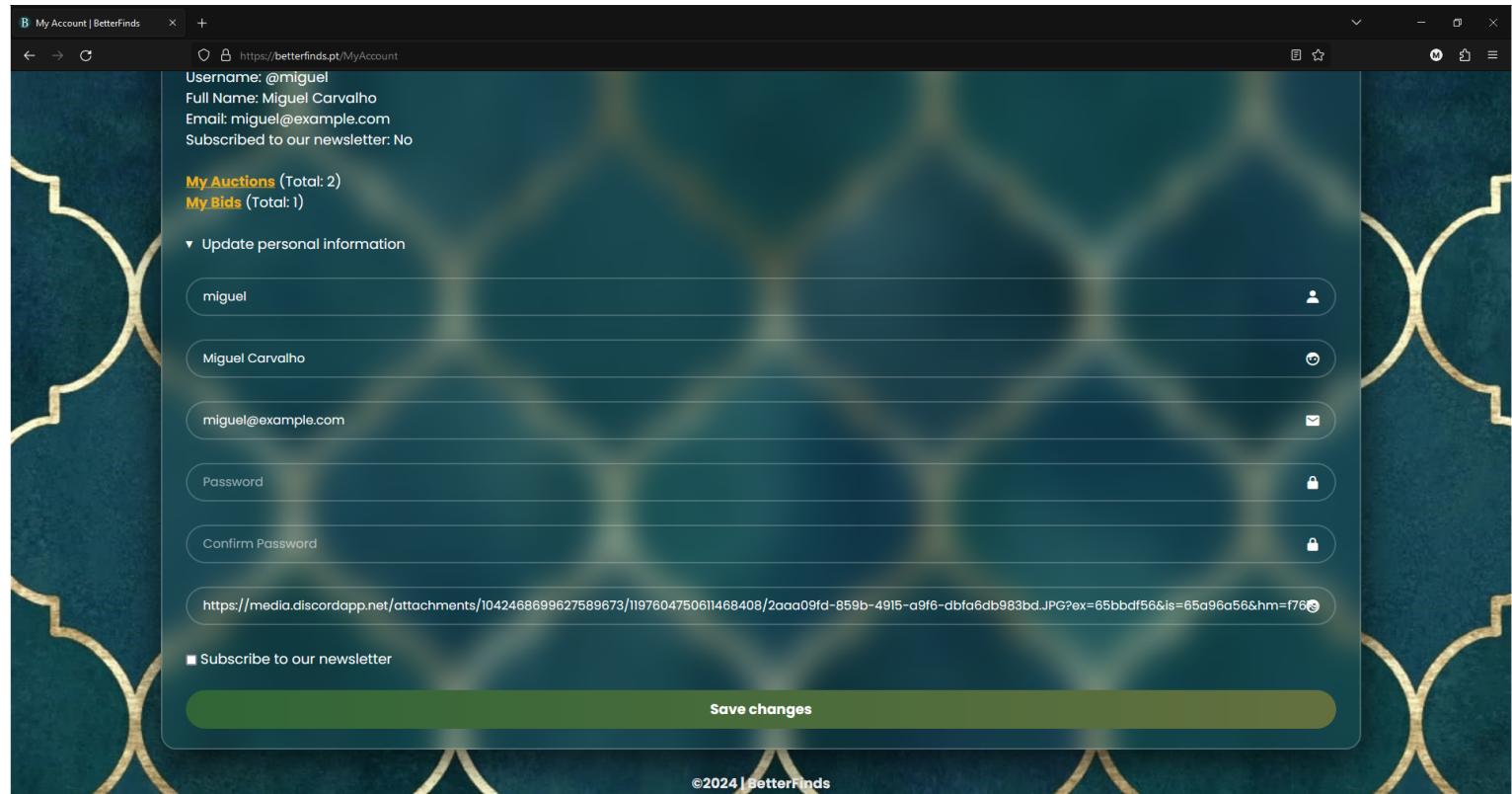


Figura 33: Editar Conta

14.6. Perfil Público

Na página *Public Profile*, à qual o utilizador consegue aceder a partir do histórico de licitações num leilão, ou, no caso do seu próprio perfil, na página *My Account*, ele tem acesso a dados e estatísticas públicas de um utilizador. Caso o utilizador estiver a ver o seu perfil público, é mostrada a opção de editar o mesmo na página *My Account*.

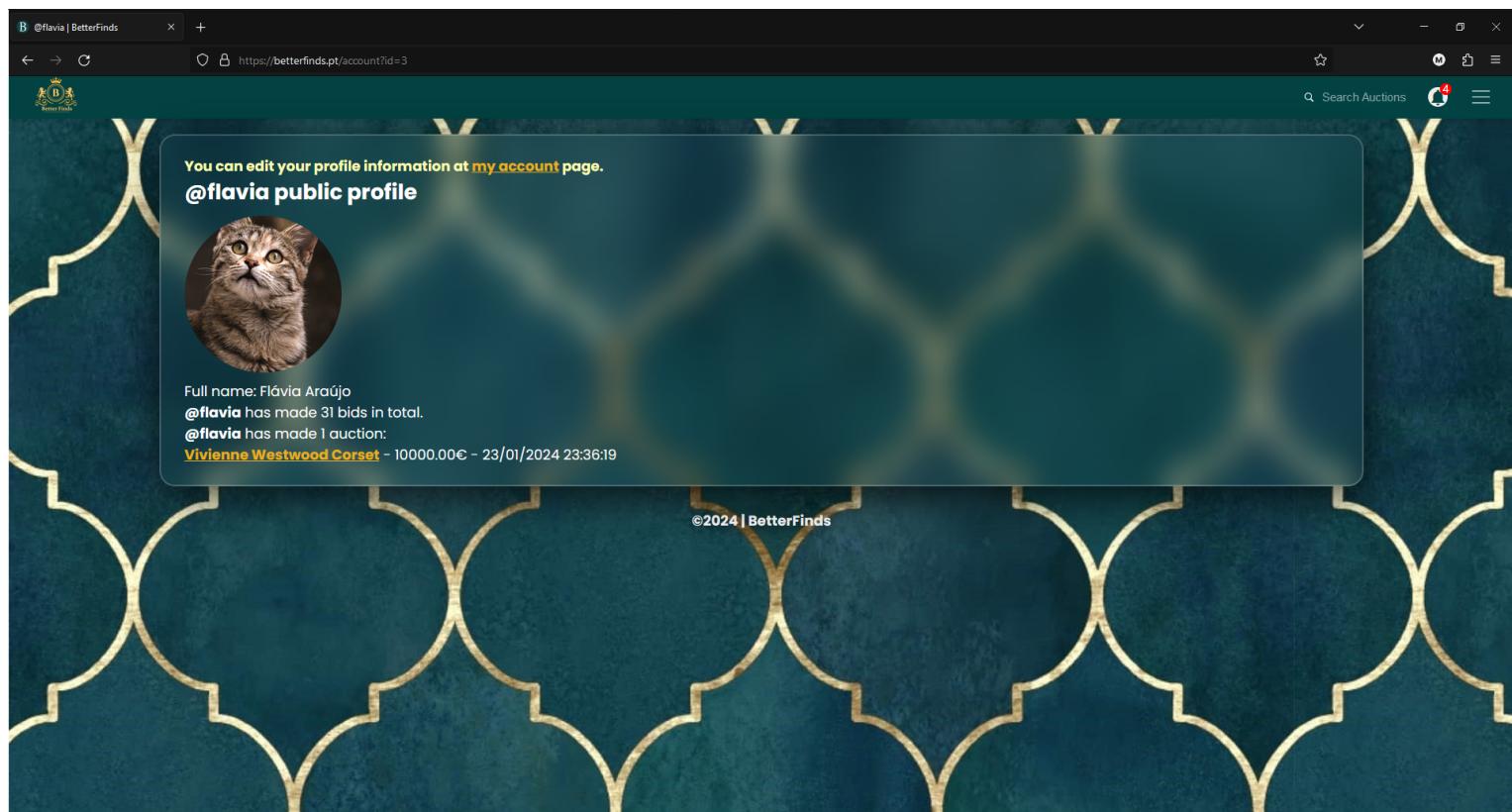


Figura 34: Perfil Público

14.7. Pesquisa

Na página de pesquisa, para além da opção de organizar os leilões pesquisados por preço, data de término e nome de produto, também é apresentada a opção de mostrar leilões que já foram terminados. Assim como ocorre na *Homepage*, sempre que existe uma atualização ou a criação de um leilão, esta página é automaticamente recarregada.

Nota: A pesquisa verifica se a *query* do utilizador está presente tanto no título do leilão/produto bem na sua descrição, sem distinção entre maiúsculas e minúsculas.

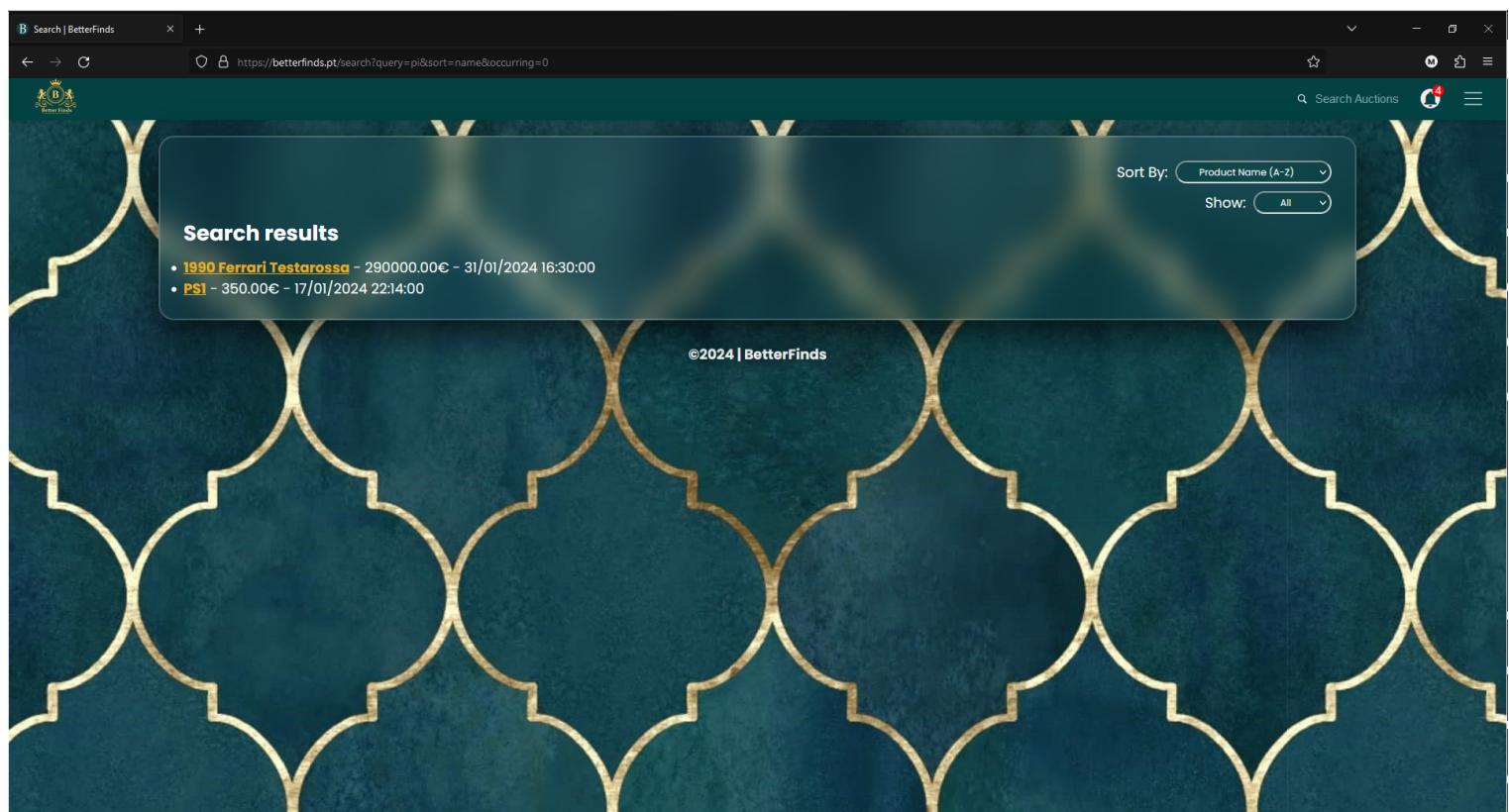


Figura 35: Página de Pesquisa

14.8. My Bids

Na página *My Bids* – à qual o utilizador tem acesso ao clicar no botão "My Bids" localizado na *sidebar* presente em todas as páginas –, o utilizador tem acesso ao seu histórico de licitações organizadas por leilão em que participou. Para poder ter acesso a esta página, o utilizador precisa ter efetuado *login*.

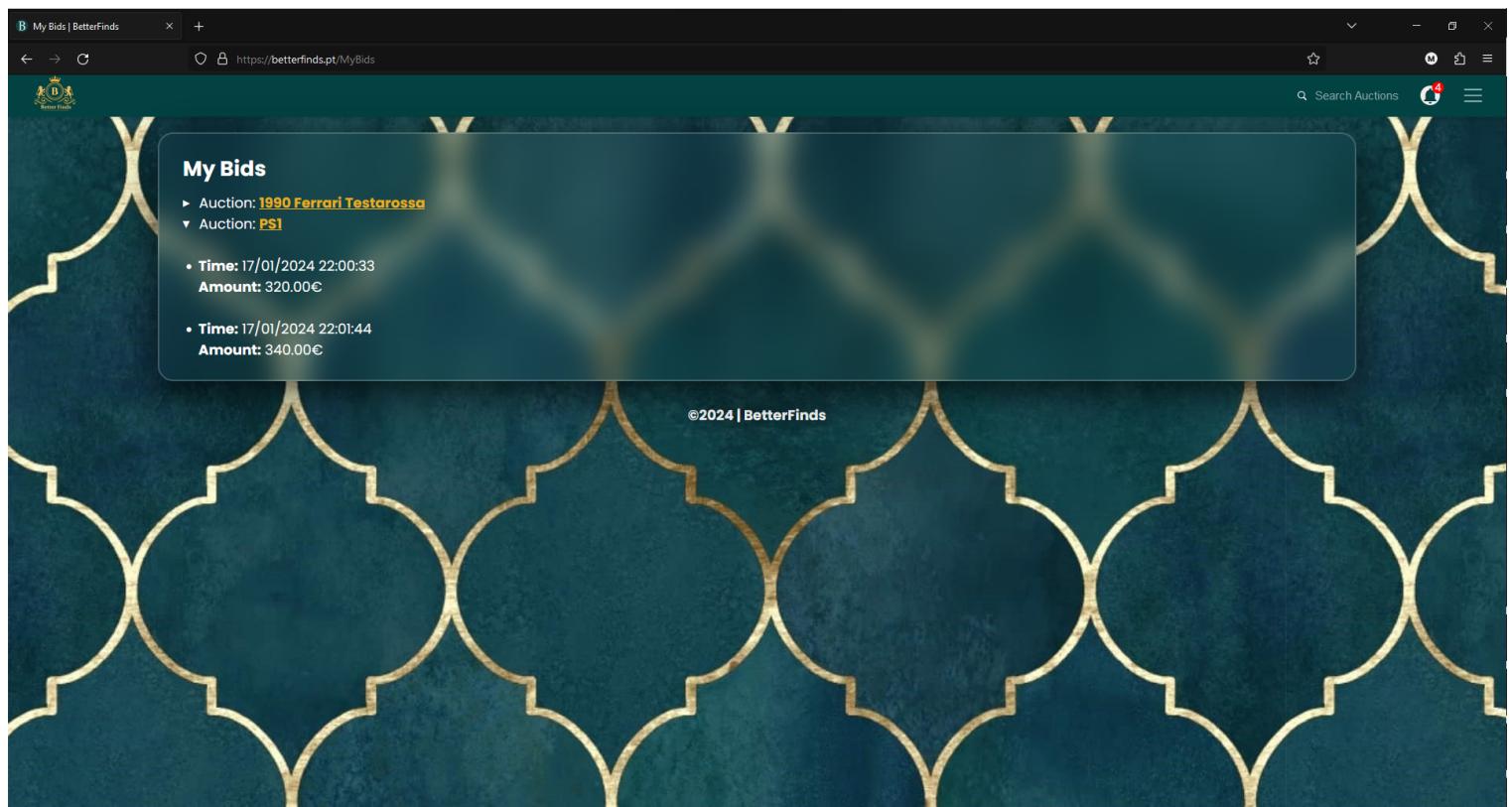


Figura 36: My Bids

14.9. My Auctions

Similar à My Bids, na página My Auctions, – à qual o utilizador tem acesso ao clicar no botão "My Auctions" localizado na sidebar presente em todas as páginas –, o utilizador tem acesso ao histórico de leilões que criou ao longo do tempo. Além disso, possui ferramentas de organização dos mesmos: por preço, data de término e nome de produto, além de ser apresentada a opção de mostar leilões que já foram terminados ou apenas os ativos atualmente.

Similarmente às páginas Search e Homepage, esta página é recarregada sempre que existem *updates*, neste caso, em leilões relevantes para o utilizador.

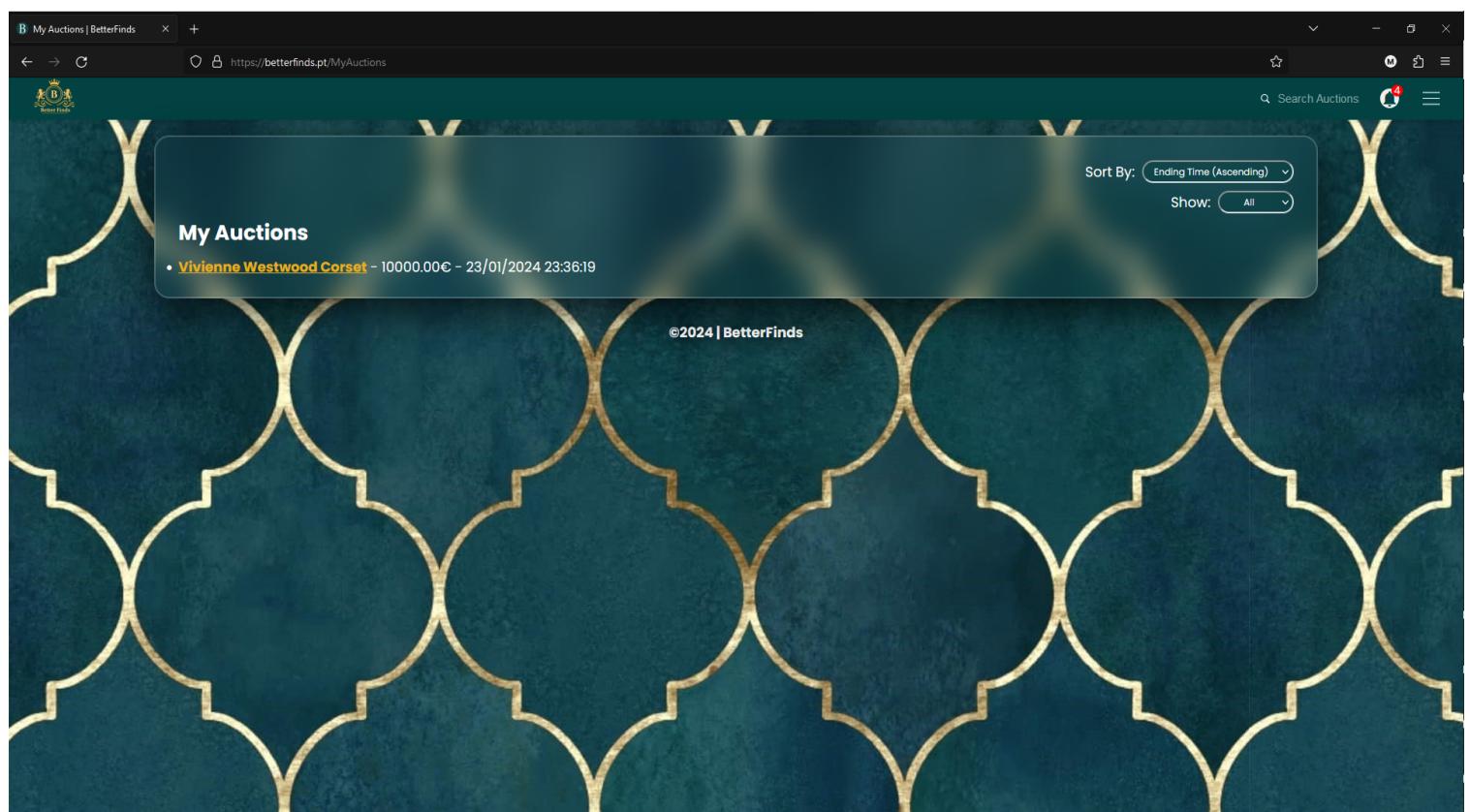


Figura 37: My Auctions

14.10. Create (Auction)

A página *Create*, – à qual o utilizador tem acesso ao clicar no botão "Create Auction" localizado na *sidebar* presente em todas as páginas –, permite ao utilizador criar um novo leilão, desde que esteja autenticado.

Nesta página existem algumas restrições, nomeadamente:

- *Title*: entre 1 a 64 caracteres;
- *Description*: entre 1 a 2048 caracteres;
- *Initial Price*: maior que zero;
- *Minimum bid increment*: maior ou igual a zero;
- *Images*: campo opcional, máximo 10 *links* de imagens separadas por vírgulas, entre 0 a 2048 caracteres.

Se o cliente falhar alguma das restrições acima será apresentada uma mensagem de erro na página com a restrição falhada, similarmente à página de registo.

Assim que o leilão é criado com sucesso, o utilizador é redirecionado para a respetiva página de leilão criada, e este leilão é adicionado a um serviço que corre no *background* onde é verificado a validade do leilão em termos temporais.

The screenshot shows a web browser window with the title 'Create Auction | BetterFinds'. The URL in the address bar is 'https://betterfinds.pt/create'. The main content is a form titled 'Create an auction' with the following fields:

- Title:** A text input field containing 'Title'.
- Description:** A text input field containing 'Description'.
- Initial price:** A text input field containing 'Initial price'.
- Minimum bid increment:** A text input field containing 'Minimum bid increment'.
- Ending date:** A date input field with the placeholder 'dd/mm/yyyy, --:--'.
- Images links (maximum 10):** A text input field containing 'image_link_1,image_link_2,...,image_link_10'.

At the bottom of the form is a large green 'Create' button. The background of the page features a dark teal color with gold-colored geometric patterns.

Figura 38: Create (Auction)

14.11. Página de Leilão (Auction)

A página de um leilão é acedida pelo utilizador através da *Homepage*, da barra de pesquisa ou do seu histórico de licitações, caso ele já tenha licitado anteriormente nesse leilão. Esta página apresenta os dados relativos ao produto a ser leiloado, bem como os detalhes do leilão:

- Título do leilão/Nome do produto;
- Preço atual;
- Temporizador para o fim do leilão;
- Até 10 imagens (se existirem), com opção de iterar entre elas com cliques do rato (ou toques, no caso de dispositivos como *smartphones*);
- Descrição;
- Horas de início e de término;
- *Seller info*: Username com opção de redireccionamento para o perfil público do vendedor;
- *Buyer info*: Similar ao *Seller info*, contudo, se não houver licitações é apresentada a mensagem "*No bids were made*".
- *Bid history* - Opção colapsável de mostrar as licitações já feitas no respetivo leilão.

Existem restrições aquando à uma nova licitação de um utilizador: O leilão tem que encontrar-se a decorrer (*end time* é maior que o tempo atual), o valor da licitação tem de ser válido (valor é maior ou igual à soma do preço atual com a *minimum bid*), o cliente não pode ser o utilizador com a licitação mais alta no momento, e o cliente não é o vencedor.

Live updates:

Quando um utilizador licita no leilão com sucesso, outros utilizadores presentes na página usufruem de *live updates* dos seguintes campos:

- *Price*;
- *Buyer info*;
- *Bid amount placeholder* (apenas se o valor atual na caixa de *input* for menor do que o novo *placeholder*, calculado através da soma do preço do leilão com a *minimum bid*).

Esta funcionalidade proporciona uma experiência natural aos utilizadores enquanto interagem com a página de leilão.

B 1990 Ferrari Testarossa | BetterFinds

https://betterfinds.pt/auction?id=2

 Better Finds

Search Auctions  4 

1990 Ferrari Testarossa

290000.00€
Minimum bid: 500.00€
Ends in: 10d 0h 31m 42s



Image: 1/3

Biarritz Sports And Classic Cars (BSCC) is delighted to present this Ferrari Testarossa. Developed on the basis of the 512 BB, the Testarossa is, along with the F40, the last Ferrari imagined by the Commendatore. A true symbol of what some consider to be the golden age of the motor industry, its instantly recognisable lines, typical of the 1980s, retain their timeless charm. The 12-cylinder flat engine, borrowed directly from Scuderia's Formula 1 engineers, develops 428 bhp.

Start Time: 2023-12-22 16:57:27
End Time: 2024-01-31 16:30:00

Seller info: [@miguel](#) - Miguel Carvalho
Buyer info: [@flavia](#) - Flávia Araújo

▼ Bid history

- [@flavia](#) - 290000.00€ - 2024-01-21 15:05:23
- [@test](#) - 289500.00€ - 2024-01-20 15:07:05
- [@flavia](#) - 289000.00€ - 2024-01-20 13:44:32
- [@test](#) - 288500.00€ - 2024-01-20 13:39:22

Bid amount:



©2024 | BetterFinds

Figura 39: Página de Leilão (Perspetiva do Comprador)

Na página de leilão, assim como se pode ver na Figura 36, é apresentada ao vendedor a funcionalidade de vender o leilão antecipadamente/terminar o leilão, dependendo esta opção de haver ou não licitações, respetivamente. Também é apresentada ao vendedor uma mensagem informativa com uma ligação à página de editar o leilão.

No caso do comprador, como se pode ver na Figura 37, assim que o leilão terminar e ele for o vencedor, será demonstrada na página a opção de escolher a forma de pagamento e respetivo botão para o realizar. O leilão é marcado como completo na base de dados após este pagamento teórico e o vendedor é notificado do mesmo.

B 1990 Ferrari Testarossa | BetterFinds +

https://betterfinds.pt/auction?id=2

You cannot bid on your own auction.
You can edit your auction at [edit auction](#) page.

1990 Ferrari Testarossa

291500.00€
Minimum bid: 500.00€
Ends in: 9d 21h 34m 7s



Image: 1/3

Biarritz Sports And Classic Cars (BSCC) is delighted to present this Ferrari Testarossa. Developed on the basis of the 512 BB, the Testarossa is, along with the F40, the last Ferrari imagined by the Commendatatore. A true symbol of what some consider to be the golden age of the motor industry, its instantly recognisable lines, typical of the 1980s, retain their timeless charm. The 12-cylinder flat engine, borrowed directly from Scuderia's Formula 1 engineers, develops 428 bhp.

Start Time: 2023-12-22 16:57:27
End Time: 2024-01-31 16:30:00

Seller info: [@miguel](#) - Miguel Carvalho
Buyer info: [@test](#) - Full Test Name

► Bid history

Bid amount:
292000.00

Bid

Sell now

©2024 | BetterFinds

Figura 40: Página de Leilão (Perspetiva do Vendedor)

B 1990 Ferrari Testarossa | BetterFinds X +

← → C ⌂ ⌂ https://betterfinds.pt/auction?id=2

Better Finds Search Auctions 5

1990 Ferrari Testarossa

291500.00€
Minimum bid: 500.00€
Auction has ended!



Image: 1/3

Biarritz Sports And Classic Cars (BSCC) is delighted to present this Ferrari Testarossa. Developed on the basis of the 512 BB, the Testarossa is, along with the F40, the last Ferrari imagined by the Commendatore. A true symbol of what some consider to be the golden age of the motor industry, its instantly recognisable lines, typical of the 1980s, retain their timeless charm. The 12-cylinder flat engine, borrowed directly from Scuderia's Formula 1 engineers, develops 428 bhp.

Start Time: 2023-12-22 16:57:27
End Time: 2024-01-20 16:30:00

Seller info: [@miguel](#) – Miguel Carvalho
Buyer info: [@test](#) – Full Test Name

► Bid history

You won the auction!

Payment method:

Credit Card ▾

- Credit Card
- PayPal
- Apple Pay
- Crypto Currency

Pay

©2024 | BetterFinds

Figura 41: Página de Leilão - Opção de Pagamento

14.12. Página de Notificações

A página de notificações é acedida através do botão de notificações na barra de navegação, e o utilizador precisa estar autenticado para as aceder. Esta página é recarregada assim que existem novas notificações.

Quando um utilizador licita no leilão com sucesso, os outros licitadores são notificados através do sistema de notificações, incrementando o número de notificações por ler no botão de notificações.

Esta funcionalidade foi implementada com a biblioteca *SignalR* desenvolvida pela Microsoft. De igual modo, o vendedor também é notificado desta maneira que recebeu uma nova licitação no seu leilão.

Quando leilão termina, o comprador é notificado para o informar que necessita de efetuar o pagamento na página de leilão, e os restantes licitadores são informados que o leilão que estavam a participar terminou. Por fim, o vendedor também é notificado que o seu leilão terminou e que terá de aguardar pelo pagamento do vencedor.

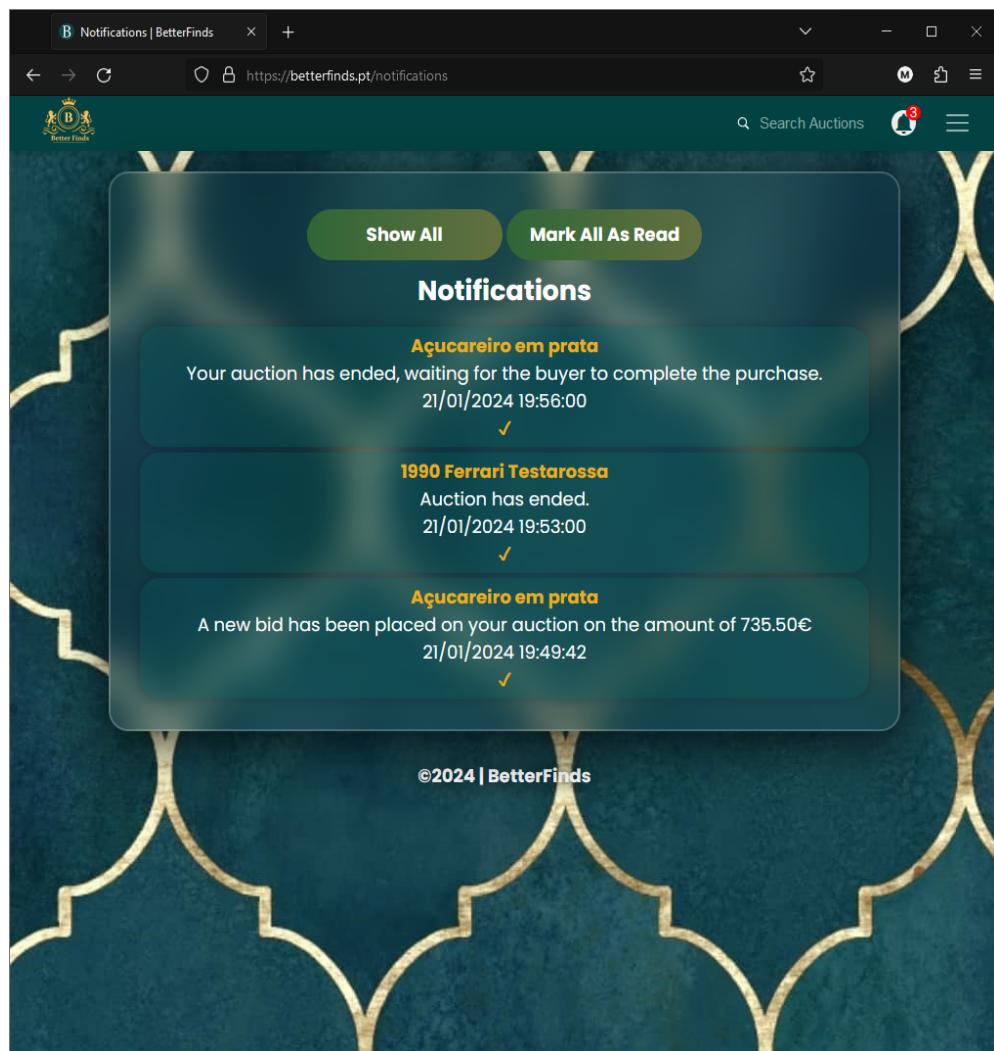


Figura 42: Página de Notificações

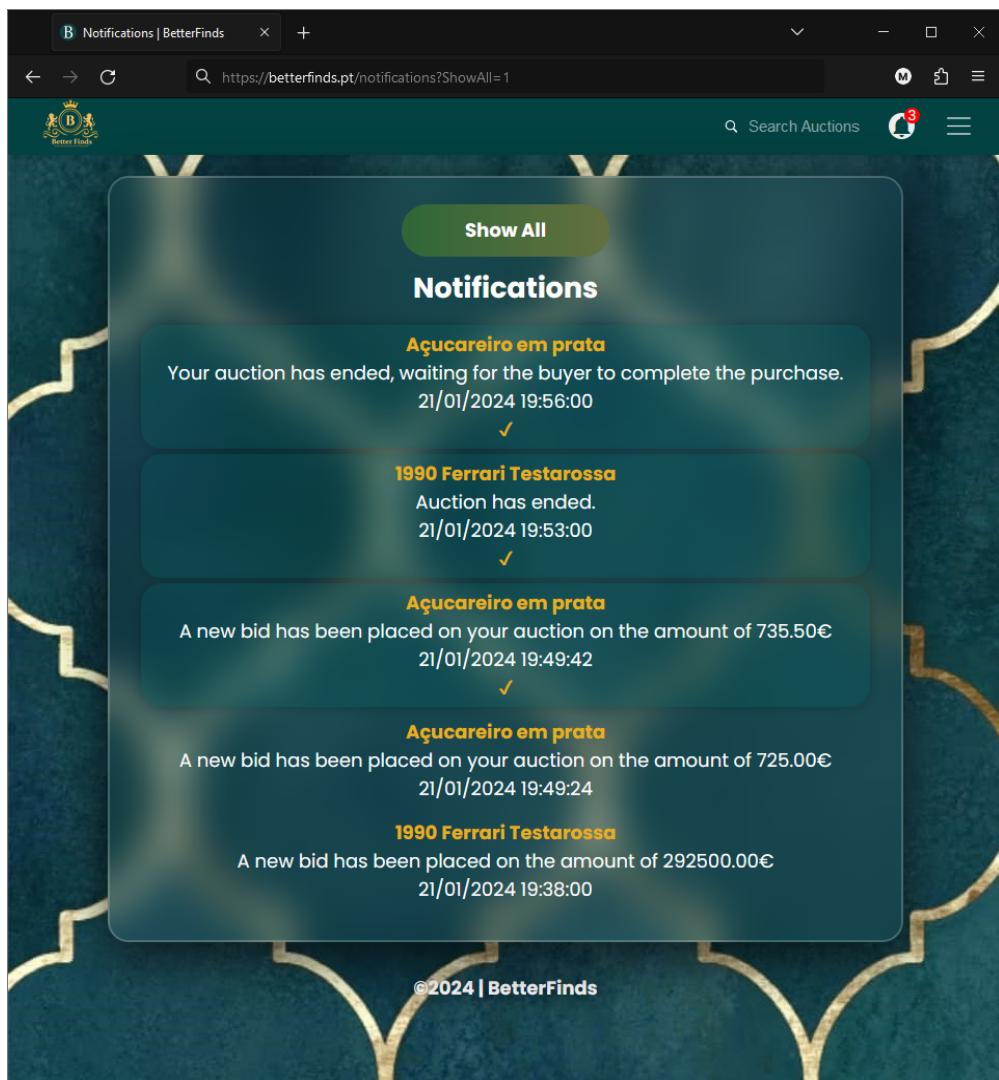


Figura 43: "Show All" Notifications

Por definição são mostradas as notificações não lidas, no entanto existe a opção de mostrar todas as notificações, assim como marcar uma notificação em específico como lida ou todas, através do uso do botão “Mark All As Read”.

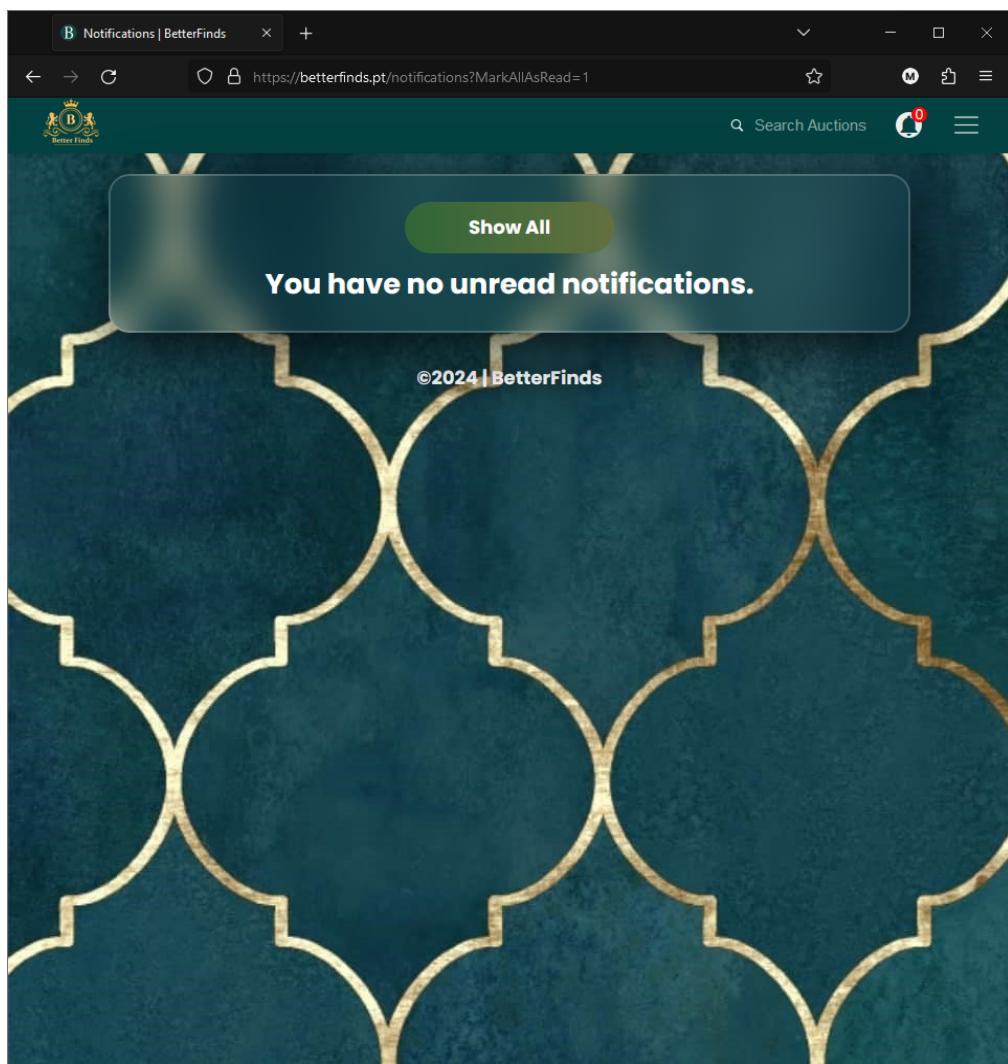


Figura 44: "You have no unread notifications"

14.13. Exemplos de Produto Final em Mobile

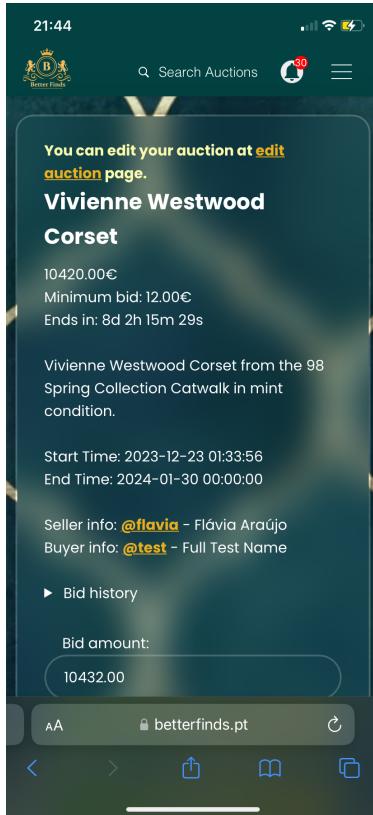


Figura 45: Página de Leilão Mobile

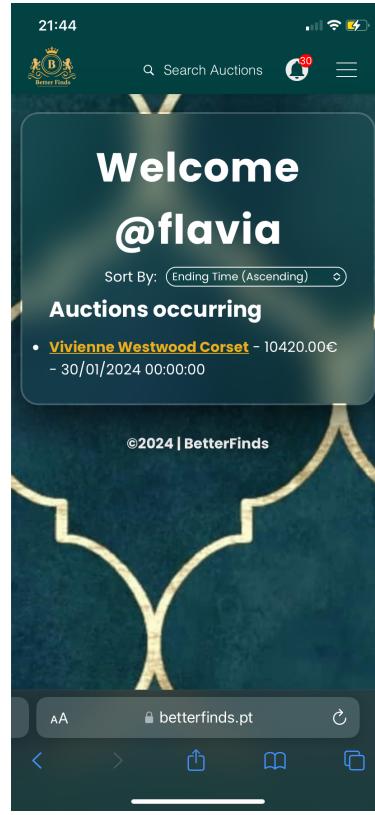


Figura 46: Homepage Mobile

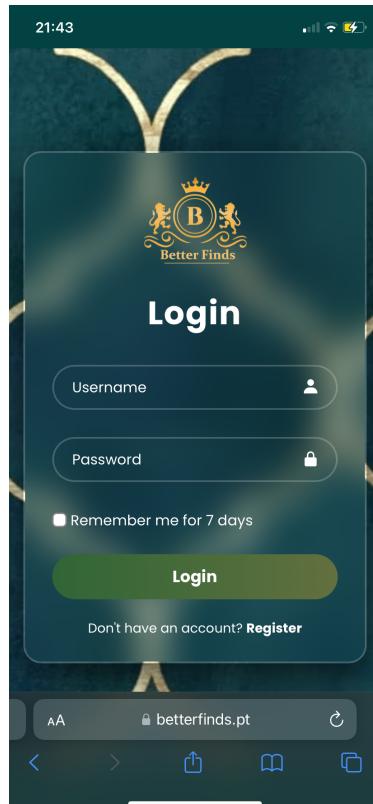


Figura 47: Login Mobile

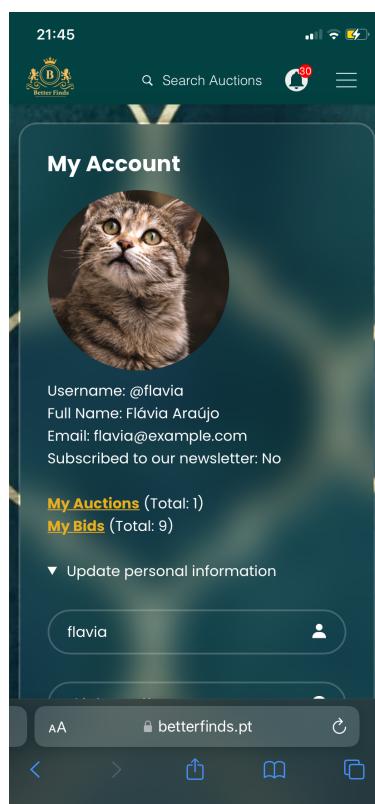


Figura 48: My Account Mobile

15. Conclusões e Trabalho Futuro

15.1. Trabalho Futuro

A nível da base de dados, gostaríamos de fazer algumas otimizações futuramente, como a remoção do *productId* da tabela *Auction* e a remoção do *auctionId* da tabela *Product*. Também queríamos introduzir o uso de chaves estrangeiras como *constraints* no nosso projeto (foi algo que tentámos fazer, porém, quando pensámos em implementá-las, no código já estava fortemente implícito a falta de uso das mesmas, e não conseguimos implementá-las como queríamos).

Futuramente, queremos implementar *transactions* em SQL para, por exemplo, criar novos leilões, pois é mais seguro em casos de saturação da aplicação. De igual modo, também queremos criar um novo *schema* para as tabelas, assim aumentaremos a *performance* e organização da base de dados.

Relativamente ao *backend*, gostaríamos de implementar o uso de várias unidades monetárias e a opção de paginação em conteúdos extensos.

Por fim, no âmbito na internacionalização, gostaríamos de implementar múltiplas linguagens através de uma extensão de terceiros.

15.2. Conclusão

Em suma, atingimos todos os objetivos definidos, cumprindo todas as tarefas propostas e prazos estabelecidos pela equipa de trabalho no diagrama de GANTT.

Para além de um melhor conhecimento das novas tecnologias utilizadas na produção deste projeto, o nosso grupo de trabalho ganhou igualmente um melhor domínio sobre metodologias de desenvolvimento e organização de tarefas em equipa, bem como métodos de trabalho no processo de desenvolvimento de *software* e a sua gestão.

Referências

- eBay. Disponível em: <https://pt.ebay.com/> [Acedido em: outubro 2023]
- CataWiki. Disponível em: <https://www.catawiki.com/en/> [Acedido em: outubro 2023]
- Sotheby's Institute of Art. Disponível em: <https://www.sothbysinstitute.com/news-and-events/news/auction-terminology> [Acedido em: novembro 2023]
- .NET. Disponível em: <https://dotnet.microsoft.com/en-us/> [Acedido em: dezembro 2023]
- Nginx. Disponível em: <https://www.nginx.com/> [Acedido em: dezembro 2023]
- SignalR. Disponível em: <https://dotnet.microsoft.com/en-us/apps/aspnet/signalr> [Acedido em: janeiro 2024]
- Cloudflare. Disponível em: <https://www.cloudflare.com/en-gb/> [Acedido em: janeiro 2024]
- DigitalOcean. Disponível em: <https://www.digitalocean.com/> [Acedido em: janeiro 2024]
- Amen. Disponível em: <https://www.amen.pt/> [Acedido em: janeiro 2024]
- CertBot. Disponível em: <https://certbot.eff.org/> [Acedido em: janeiro 2024]
- NUnit. Disponível em: <https://nunit.org/> [Acedido em: janeiro 2024]
- Selenium. Disponível em: <https://www.selenium.dev/> [Acedido em: janeiro 2024]
- NuGet. Disponível em: <https://www.nuget.org/> [Acedido em: janeiro 2024]
- LambdaTest. Disponível em: <https://www.lambdatest.com/> [Acedido em: janeiro 2024]
- DeepSource. Disponível em: <https://deepsource.com/> [Acedido em: janeiro 2024]
- Snyk. Disponível em: <https://snyk.io/> [Acedido em: janeiro 2024]
- Wikipedia: Cross-Site Scripting. Disponível em: https://pt.wikipedia.org/wiki/Cross-site_scripting [Acedido em: janeiro 2024]

Lista de Siglas e Acrónimos

BD	Base de Dados
UI	<i>User Interface</i>
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
http	<i>Hypertext Transfer Protocol</i>
https	<i>Hypertext Transfer Protocol Secure</i>
MVC	<i>Model View Controller</i>
T-SQL	<i>Transact-SQL</i>
UTC	<i>Coordinated Universal Time</i>
CDN	<i>Content Delivery Network</i>
DDoS	Distributed Denial-of-Service
SSL	<i>Secure Sockets Layer</i>
TLS	<i>Transport Layer Security</i>
MSSQL	<i>Microsoft SQL Server</i>
SQL	<i>Structured Query Language</i>
MITM	<i>Man-In-The-Middle</i>
XSS	<i>Cross-Site Scripting</i>