

LEI_PAW_TP_AF

RELATÓRIO

Versão <1.0>

<20/06/2022>

Índice

| | |
|---------------------------------------|-----------|
| Introdução. | 3 |
| ESPECIFICAÇÃO DO PROBLEMA..... | 4 |
| DESCRIÇÃO TÉCNICA..... | 5 |
| FALHAS/DEFECIÊNCIAS..... | 11 |
| CONCLUSÃO..... | 12 |

INTRODUÇÃO

Este trabalho foi realizado no âmbito da unidade curricular de Programação em Ambiente Web (PAW). O mesmo consistia em desenvolver uma aplicação web responsável por suportar uma agência imobiliária, desde a publicação de propriedades por parte dos agentes imobiliários, editá-los etc, aos clientes, em que podiam após estarem registados no website consultar informação detalhada de todos os imóveis listados, assim como adicioná-los a uma lista de favoritos em que podiam seguir o imóvel a qualquer altura, em modo anónimo os clientes apenas podiam consultar informação básica dos imóveis.

A nível de tecnologia utilizada, foi-nos recomendado a utilização da framework Angular para o frontend, ou seja tudo o que seja da parte do cliente, Node e Express para o backend e MongoDB para armazenar todos os dados que suportam esta aplicação, assim como a arquitetura REST para a API, e por fim o git/gitlab para o controlo de versões, tendo sido utilizado o gitlab da Estg no repositório

https://gitlab.estg.ipp.pt/8170464/8170464_paw_tp_af.

ESPECIFICAÇÃO DO PROBLEMA

No início, foram utilizados diagramas UML para o melhor entendimento do problema e da solução a desenvolver, em concretos diagramas de Use Case, em que foram definidas todas as funções/permisões atribuídas a todos os participantes da aplicação, desde o agente imobiliário, cliente ao cliente não autenticado, assim como podemos ver na Figura 1. apresentada de seguida

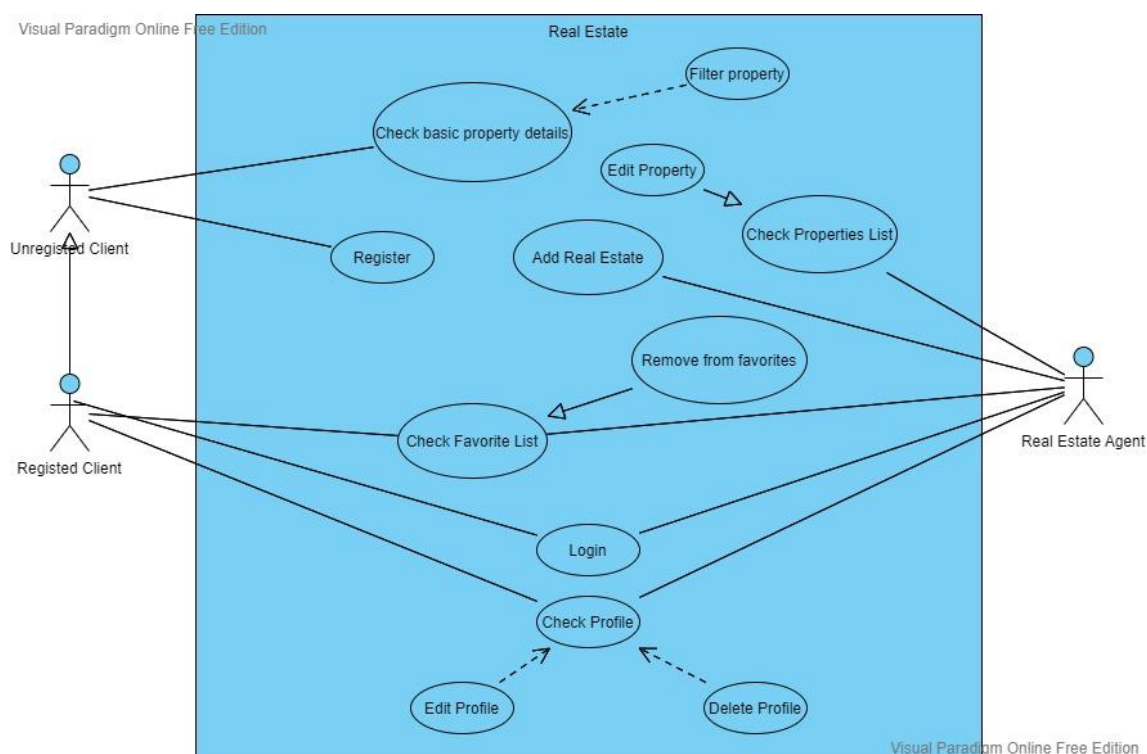


Figura 1 Diagrama Use Cases do sistema

Como é possível observar no diagrama, aquilo que um cliente não registado pode fazer é muito limitado, apenas pode verificar informação básica do imóvel, desde a tipologia, metros quadrados, a morada e o contacto do agente responsável pelo mesmo, após estar autenticado pode verificar toda a informação do mesmo, assim como imagens e o preço. Verificar o seu perfil assim como editá-lo ou até mesmo o

eliminar caso não pretenda continuar a utilizar a aplicação, também pode seguir o imóvel para mais informação.

Relativamente ao agente imobiliário o mesmo pode fazer a mesmas ações que um cliente autenticado e ainda adicionar propriedades, assim como verificar a lista de todas as propriedades e editá-las.

DESCRIÇÃO TÉCNICA

Ao iniciar a aplicação, no endereço “/”, é possível ver a lista dos imóveis publicados, mas uma vez que não estamos autenticados não é possível ver toda a informação dos imóveis, também é possível filtrar os imóveis por tipologia ex: T1;T3, a cidade e os metros quadrados do mesmo, também é possível ver o “header”, que é um componente separado que foi injetado no principal, em que existe um “about” que abre um “matDialog” com alguma informação sobre a empresa, o botão “home” que redireciona para o próprio componente e o login/register que é substituído por outros botões no momento que efetuamos o login/registo.

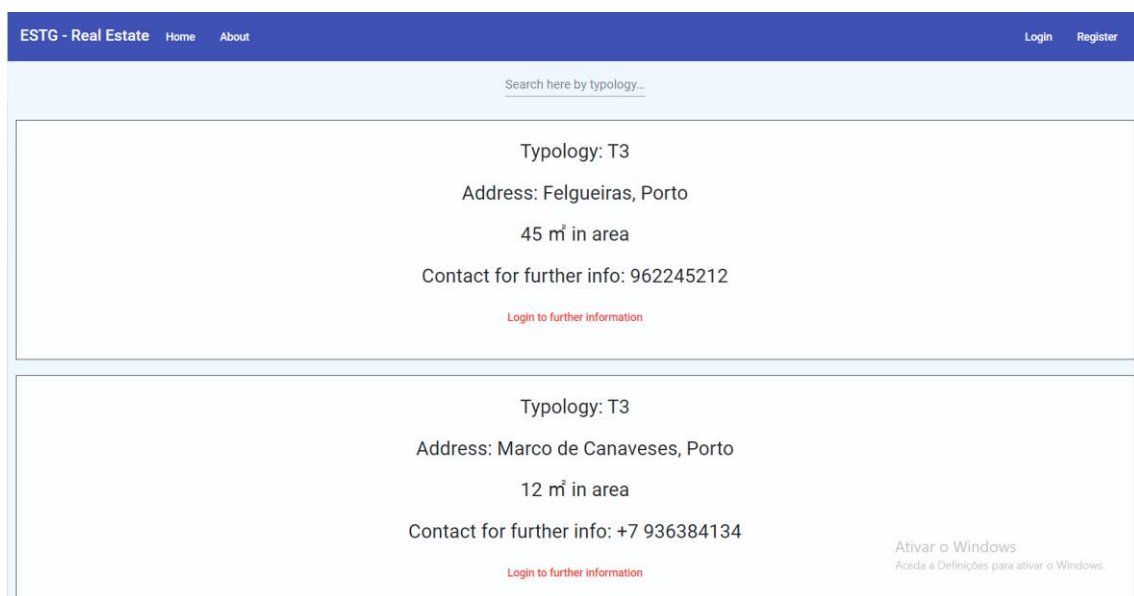


Figura 2 Home component

Como é possível observar na Figura 3, após efetuar o login, o “header” é alterado assim como a interface dos imóveis, é possível aceder ao perfil, aceder a lista de favoritos e dar logout, também é possível aceder a lista completa de imóveis e adicionar uma propriedade, mas essas rotas estão protegidas e apenas podem ser acedidas por agentes imobiliários, para executar essa verificação, ao dar login/register é atribuída ao utilizador um token que fica guardado no localStorage, e sempre que é necessário verificar a “role” de um utilizador, esse mesmo token é enviado no header do pedido http através de um interceptor, e esse mesmo é validado através de um middleware no backend, caso o token esteja válido e com as permissões adequadas, o middleware dá “next” e é acedida a rota. Caso contrário é retornado um código http e somos redirecionados para a página principal.

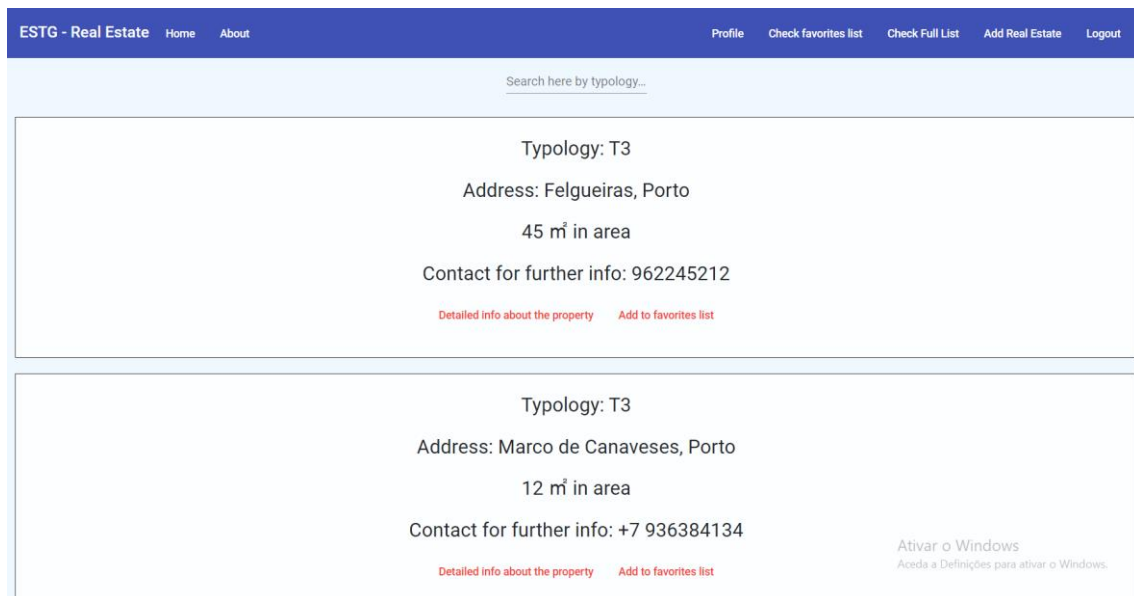


Figura 3 Home component após login/register

Antes de ter efetuado a autenticação, não era possível aceder a informações detalhadas, agora aparecem dois botões, um para abrir os detalhes do imóvel, o outro é para adicionar o próprio imóvel à lista de favoritos.

Detailed info about the property

Typology : T0

Square Feet : 2 m²

Full Address: Porto, Porto Rua Dona Fernanda 4500-458

Agent Cellphone: 903214568

Neighborhood Description: Very nice

Price: 120000€



Figura 4 Detalhes do imóvel

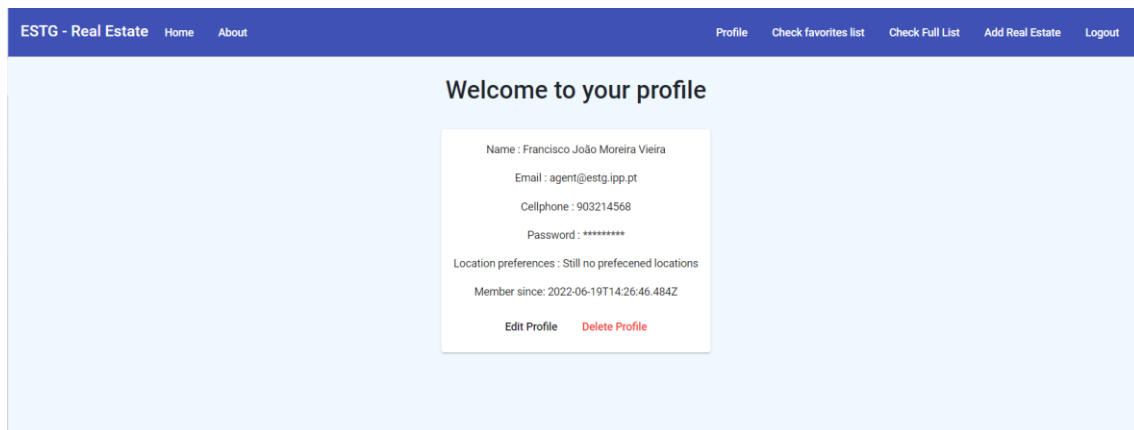


Figura 5 Informação do perfil

Como é possível ver na figura 5, é apresentado toda a informação sobre o utilizador, desde o nome á data em que o cliente foi registado, a informação é obtida através de um “service” que faz um GET Request à API e que a mesma devolve um objecto do tipo any que contém toda a informação do cliente, também é possível editar o perfil e eliminar o mesmo, que utiliza o mesmo serviço, porém métodos diferentes, DELETE para eliminar e PUT para editar.

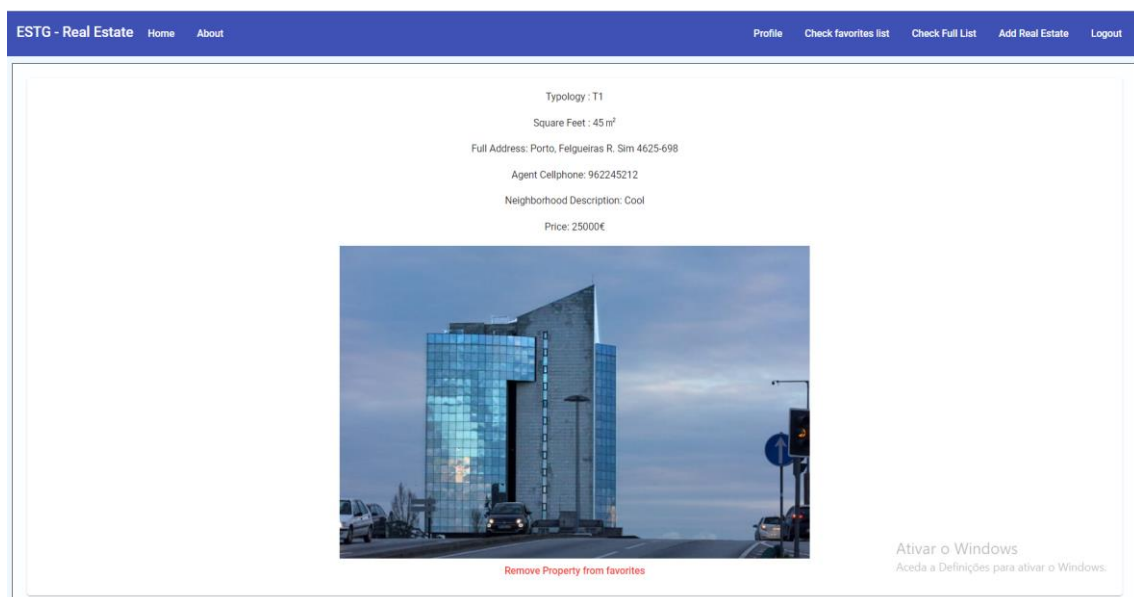


Figura 6 Lista de Favoritos

Como é possível observar na figura 6, ao adicionar um imóvel aos favoritos, quando acedemos a essa lista, podemos ver toda a informação do imóvel assim como a sua fotografia, uma vez que quando acionamos o imóvel, o mesmo é adicionado ao array de imóveis no perfil do utilizador, que será possível ver já de seguida.


```

    _id: ObjectId("62af3226526080f874f3a499")
    name: "Francisco João Moreira Vieira"
    email: "agent@estg.ipp.pt"
    password: "$2a$08$ip1dQtNKsB6G18YIHveWmelaLUcoa0.HVgZ1ZprUB007LFkjFBvRS"
    cellphone: "903214568"
    role: "agent"
  ✓ locationPreferen... : Array
    0: "Felgueiras"
    1: "Porto"
    2: "Lisboa"
    createdAt: 2022-06-19T14:26:46.484+00:00
  ✓ favoriteList: Array
    > 0: Object
    > 1: Object
    > 2: Object
    __v: 0

```

Figura 7 Model de um Utilizador

Este modelo de dados é definido no mongoose, sendo que os dados name,email,password,cellphone e role são obrigatórios, ou seja no momento da criação do perfil, esses dados têm de ser conhecidos, os restantes podem ser adicionados mais tarde, como favoriteList e locationPreference, o campo createdAt, é criado automaticamente no momento que o utilizador é registado, o modelo de dados de uma propriedade também foi criado e será apresentado de seguida.

```

    _id: ObjectId("62af4b701802e6a7a61bd916")
  ✓ description: Object
    typology: "T3"
    sqFeet: 45
    _id: ObjectId("62b1a21ca0d41f0ee7425947")
  ✓ location: Object
    district: "Porto"
    city: "Felgueiras"
    street: "R. Sim"
    cp: "4625-698"
    _id: ObjectId("62b1a21ca0d41f0ee7425948")
    neighborhoodDescrip... : "Cool"
    price: 0
    agentCellphone: "962245212"
    image_url: "https://portoalities.com/wp-content/uploads/2018/04/torre-antas-edific..."
    forSale: "for sale"
    publishState: "published"
    __v: 0

```

Figura 8 Model de uma propriedade

Na figura 8, podemos ver o modelo de dados de uma propriedade, que consiste na morada, descrição do tipo de casa e metros quadrados, ainda o tipo de bairro, o preço, o contacto do agente responsável pelo mesmo, um endereço da imagem do imóvel, se ainda está para venda e o estado de publicação do imóvel. A informação relativamente a descrição e a morada foram estruturadas num Schema separado para uma melhor definição da propriedade.

FALHAS/DEFECIENCIAS

Relativamente aos pontos fracos do projecto, a grande falha do mesmo é as imagens do imóvel, uma vez que a mesmo não foi adicionada dinamicamente, mas sim por manipulação da base de dados.

De resto, penso que o projeto esteja bastante alinhado com aquilo que nos foi proposto e aquilo que foi lecionado na unidade curricular.

CONCLUSÃO

Posso concluir que este projeto foi importante na obtenção de conhecimentos na forma em que a web funciona, mais em concreto na arquitetura cliente/servidor.

No potencial da framework Angular, uma vez que permite executar tudo aquilo envolvendo o cliente, e ainda através de interceptores ajudar a comunicação com a Base de dados através da REST API, foi também importante no ponto de vista do backend, para realmente entender a lógica do servidor no momento em que tem que lidar com pedidos e a maneira que trata deles.

Em suma, exceto a falha já referida anteriormente relativamente à maneira em que as imagens são tratadas pela aplicação, foi um trabalho muito relevante para a aquisição de conhecimentos na área do web.