

# TPW\_Trabalho2 - Football Nerds.

João Soares Nmec 93078

Hugo Ferreira Nmec 93093

## Como executar:

- Verificar os requirements de DjangoServer (Rest Api) e AngularClient (Aplicação Angular)
- A aplicação angular está pré configurada para aceder ao Django Server via localhost.
- Caso pretenda mudar para interagir com a versão deployed apenas é necessário aceder ao ficheiro *AngularClient/src/app/\_classes/globalconstants.ts* e mudar os url para:

```
1 export class GlobalConstants {  
2     public static apiUrl: string = "http://joaots.pythonanywhere.com/ws/"  
3     public static baseUrl: string = "http://joaots.pythonanywhere.com"  
4 }
```

## Deployment

Link DjangoServer:

<http://joaots.pythonanywhere.com>

Link Aplicação Angular:

<https://footballnerds.herokuapp.com>

## Contas Disponíveis

- **Admin Account:**
  - username: admin
  - password: admin1234
- **NormalUser Account:**
  - username: user1
  - password: 3=eX6xXK

Nota:

- Os Dados locais podem ter ligeiras diferenças da versão online.
- Players como o "Palhinha" demonstram algumas das funcionalidades.
- Notar que a versão online tem um ligeiro "delay"

## Pequena Contextualização

Football Nerds. é um website com dados relacionados com o mundo do Futebol. Contém dados sobre Equipas, Jogadores, Staff, Competições e seus jogos, permitindo uma gestão dos diferentes dados.

## Funcionalidades e Autorizações

*NoLogin -> User sem login realizado*

### Funcionalidade Sign Up/Login:

- **NoLogin:**
  - Poderá realizar Sign Up apenas de NormalUser
  - É pedido username, email address, Password
  - Caso insira dados incorretos serão apresentadas mensagens de erro
- **NormalUser:**
  - Pode realizar o Login
- **Admin:**
  - Pode realizar o Login

### Funcionalidade Adicionar Comentários:

- **NoLogin:**
  - Não pode adicionar comentários na Página\_Detalhada de Competitions/Teams/Players/Staff.
- **NormalUser:**
  - Pode adicionar comentários na Página\_Detalhada de Competitions/Teams/Players/Staff.
- **Admin:**
  - Não pode adicionar comentários na Página\_Detalhada de Competitions/Teams/Players/Staff.

### Funcionalidade Adicionar/Remover Favoritos:

- **NoLogin:**
  - Não pode adicionar/remover Favoritos.
- **NormalUser**
  - Pode adicionar e posteriormente remover *Competitions/Teams/Players* como favoritos e visualizar a lista dos mesmos no seu profile.
- **Admin:**
  - Não pode adicionar/remover Favoritos.

### Funcionalidade Pesquisa por Nome:

**NormalUser/Admin/NoLogin** podem pesquisar *Teams/Competitions/Players/Staff* pelo nome.

Para iniciar a pesquisa apenas começar a escrever na barra de pesquisa.

Para reiniciar a pesquisa simplesmente apagar os conteúdos da barra de pesquisa.

### Funcionalidade Visualizar Dados em Detalhe:

**NoLogin/NormalUser/Admin** podem visualizar:

- **Competition:** Pode Visualizar Comentários, Dados sobre a Competition e por época as Teams que participam e Matches da Competition que permitem gerar a Tabela Classificativa.
- **Team:** Pode Visualizar Comentários, Dados sobre a Team e por época os Players da Team, o Staff da Team e as Competitions que participam.
- **Player:** Pode Visualizar Comentários, Dados sobre o Player e as Teams que o Player jogou em determinada época.
- **Staff:** Pode Visualizar Comentários, Dados sobre o Staff e as Teams que o Staff "participou" em determinada época.

Por Default sem selecionar uma época os dados expostos são sobre a atual época 2020-2021

### Funcionalidade Inserir Dados:

- **NoLogin:**
  - Não pode Inserir Dados
- **NormalUser:**
  - Pode apenas Inserir *Team/Staff/Player/Competition*
- **Admin:**
  - Pode inserir *Team/Staff/Player/Competition*
  - Pode inserir relações entre as entidades *Team/Staff/Player/Competition*:
    - Team joga numa Competition em determinada época (*ClubPlaysIn*)
    - Player joga numa Team em determinada época (*StaffManages*)
    - Staff trabalha numa Team em determinada época (*PlayerPlaysFor*)
  - Pode Inserir as Matches de uma Competition:
    - (*Match*) e (*CompetitionsMatches*) realizada de forma simultânea

### Funcionalidade Editar Dados:

- **NoLogin:**
  - Não pode Editar Dados
- **NormalUser:**
  - Não pode Editar Dados
- **Admin:**
  - Pode Editar *Team/Staff/Player/Competition*

### Funcionalidade Eliminar Dados:

- **NoLogin:**
  - Não pode Eliminar Dados
- **NormalUser:**
  - Não pode Eliminar Dados
- **Admin:**
  - Pode Eliminar *Team/Staff/Player/Competition* e consequentemente as suas Relações.

## Detalhes de Implementação

### Autenticação:

Para realizar a autenticação de utilizadores foi utilizado JWT Authentication. Desta forma geríamos as permissões para certas funcionalidades entre diferentes tipos de utilizador da aplicação.

Assim na aplicação certo método contém `canActivate` para verificar se utilizadores válidos (se não forem são redirecionados para Login):

```
{path: 'insertcompetition', component: InsertCompetitionComponent, canActivate: [AuthGuard]],
```

No Django Server também certos métodos tinham uma verificação da autenticação:

```
@permission_classes((IsAuthenticated,))  
def insert_competition(request):
```

### Editar e Inserir Dados:

Foi utilizado a class `FormGroup` para guardar dados dos formulários à medida que estes são alterados e para enviar a informação por POST utilizamos a class `FormData` o que ajudou na tarefa de enviar imagens para o DjangoServer (que caso não seja adicionado no inserir este utiliza uma imagem defaultl).

Ao longo da aplicação são utilizadas diferentes funcionalidades do angular para realizar pesquisa por nome e filtrar por época.

## Repositórios

*GitHub Repository Principal:*

[https://github.com/JoaoTS20/TPW\\_Trabalho2](https://github.com/JoaoTS20/TPW_Trabalho2)

[https://github.com/JoaoTS20/TPW\\_FootballNerds\\_Angular](https://github.com/JoaoTS20/TPW_FootballNerds_Angular) (repositório auxiliar para as configurações de deployment da aplicação Angular)

## AngularClient Requirements:

- npm install -s moment
- npm install -s jwt-decode
- npm install -s @types/jwt-decode
- npm install font-awesome --save

## DjangoServer Requirements:

- Python 3.8 / 3.9
- Django 3.2
- Pillow 2.8.0
- Pip 20.2.3
- django-allauth 0.44.0
- django-cors-headers 3.7.0
- django-cors-middleware 1.5.0
- django-filter 2.4.0
- django-rest-auth 0.9.5
- djangorestframework 3.12.4
- djangorestframework-jwt 1.11.0