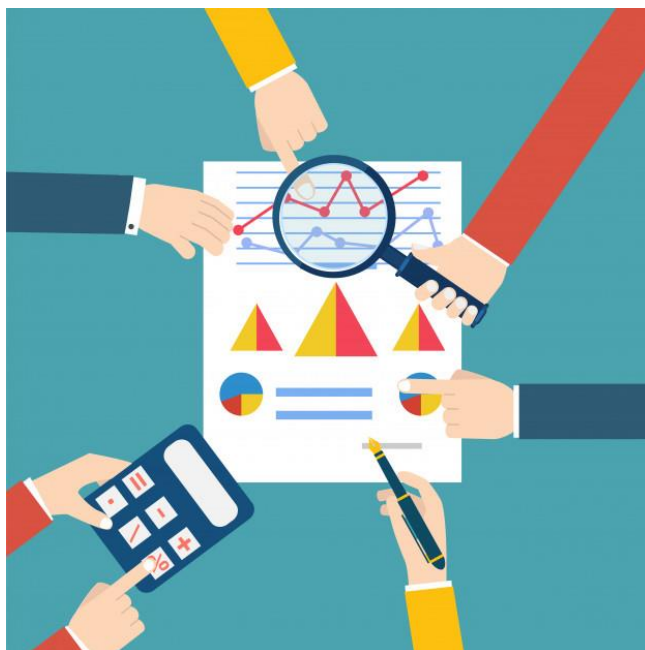


**PROJETO DE SERVIÇOS E INTEROPERABILIDADE DE SISTEMAS**

# **Relatório de Acompanhamento do Ciclo de Desenvolvimento de Software do Projeto em Serviços e Interoperabilidade de Sistemas**



<b>Turno:</b> PL1	<b>Grupo:</b> A	<b>Docente:</b> Mário Fernandes
<b>Nº 2202415</b>	Andreia Agostinho	
<b>Nº 2201126</b>	Diogo Pereira	
<b>Nº 2201131</b>	Gonçalo Ferreira	

Cofinanciado por:



Instituto Politécnico de Leiria  
Torres Vedras  
Curso Técnico Superior Profissional de Programação de Sistemas de Informação  
10 de Janeiro de 2022  
Andreia Agostinho  
Diogo Pereira  
Gonçalo Ferreira

Cofinanciado por:



# ÍNDICE

1	INTRODUÇÃO.....	7
2	ENDPOINTS.....	8
2.1	Cars	8
2.1.1	Métodos de Implementação .....	8
2.2	Companies	13
2.2.1	Métodos de Implementação .....	13
2.3	CompaniesList	16
2.3.1	Métodos de Implementação .....	16
2.4	Contributors	17
2.4.1	Métodos de Implementação .....	17
2.5	Login	20
2.5.1	Métodos de Implementação .....	20
2.6	Repairs	21
2.6.1	Métodos de Implementação .....	21
2.7	Schedules	26
2.7.1	Métodos de Implementação .....	26
2.8	Signup	30
2.8.1	Métodos de Implementação .....	30
2.9	User	31
2.9.1	Métodos de Implementação .....	31
2.10	Informações	34
3	MOSQUITTO .....	35
3.1	Introdução ao Mosquitto	35
3.2	Aplicação Prática	35
4	CONCLUSÃO.....	39

Cofinanciado por:



# ÍNDICE DE FIGURAS

Figura 1 - Cars, Método GET .....	8
Figura 2 - Cars, Método POST .....	9
Figura 3 - Cars, Método SET CUSTOM .....	9
Figura 4 - Cars, Método TOTAL CUSTOM.....	10
Figura 5 - Cars, Método TOTAL USER CUSTOM .....	10
Figura 6 - Cars, Método DELETE.....	10
Figura 7 - Cars, Método PUT .....	11
Figura 8 - Cars, Método DELETE CUSTOM .....	11
Figura 9 - Cars, Método POST CUSTOM.....	12
Figura 10 - Cars, Método PUT CUSTOM .....	12
Figura 11 - Companies, Método GET .....	13
Figura 12 - Companies, Método POST .....	13
Figura 13 - Companies, Método PUT .....	14
Figura 14 - Companies, Método DELETE.....	14
Figura 15 - Companies, Método TOTAL CUSTOM.....	15
Figura 16 - Companies, Método SET CUSTOM .....	15
Figura 17 - CompaniesList, Método GET.....	16
Figura 18 - Contributors, Método GET .....	17
Figura 19 - Contributors, Método POST .....	17
Figura 20 - Contributors, Método PUT .....	18
Figura 21 - Contributors, Método DELETE .....	18
Figura 22 - Contributors, Método SET CUSTOM.....	19
Figura 23 - Contributors, Método TOTAL CUSTOM .....	19
Figura 24 - Login, Método DO CUSTOM .....	20
Figura 25 - Repairs, Método GET .....	21
Figura 26 - Repairs, Método HISTORY CUSTOM .....	22
Figura 27 - Repairs, Método REPAIR CONTRIBUTOR CUSTOM .....	22
Figura 28 - Repairs, Método POST CUSTOM .....	23
Figura 29 - Repairs, Método PUT CUSTOM .....	23
Figura 30 - Repairs, Método DELETED CUSTOM.....	24
Figura 31 - Repairs, Método TOTAL CUSTOM .....	24
Figura 32 - Repairs, Método SET CUSTOM .....	25
Figura 33 - Schedules, Método TOTAL CUSTOM .....	26
Figura 34 - Schedules, Método SET CUSTOM .....	26
Figura 35 - Schedules, Método POST CUSTOM .....	27
Figura 36 - Schedules, Método DELETE CLIENT CUSTOM.....	27
Figura 37 - Schedules, Método DELETE CONTRIBUTOR CUSTOM .....	28
Figura 38 - Schedules, Método PUT CLIENT CUSTOM .....	28
Figura 39 - Schedules, Método PUT CONTRIBUTOR CUSTOM .....	29
Figura 40 - Signup, Método POST CUSTOM.....	30

Figura 41 - User, Método ACCOUNT CUSTOM .....	31
Figura 42 - User, Método TOTAL CUSTOM .....	31
Figura 43 - User, Método SET CUSTOM .....	32
Figura 44 - User, Método DELETED CUSTOM .....	32
Figura 45 - User, Método PUT CUSTOM .....	33
Figura 46 - Mosquitto, Função afterSave .....	35
Figura 47- Mosquitto, Função FazPublish.....	36
Figura 48 - MQTT Conexão Móvel .....	37
Figura 49 - MQTT Subscribe .....	38
Figura 50 - Exemplo de Notificação .....	38

Cofinanciado por:



# 1 INTRODUÇÃO

No âmbito da unidade curricular de Serviços e Interoperabilidade de Sistemas do 1º Semestre do 2º ano do Curso Técnico Superior Profissional de Programação de Sistemas de Informação do Instituto Politécnico de Leiria, foi criado o projeto apelidado de “CarBuddy”, que consiste no desenvolvimento de uma plataforma de gestão de todo o tipo de veículos, desde as suas reparações às suas manutenções.

A nossa plataforma consiste numa combinação de três componentes, sendo eles um Website, uma Aplicação Móvel e uma API.

Assim, para a presente unidade curricular, o objetivo é o desenvolvimento da API, sendo esta uma API REST, para a unidade curricular de Plataformas de Sistemas de Informação, o objetivo é a criação e desenvolvimento do Website e para a unidade curricular de Acesso Móvel a Sistemas de Informação, o objetivo é a criação de uma aplicação móvel.

No website, é onde ocorre a gestão de contas de utilizador, onde irão ser vistos/registados os veículos, o seu estado e as suas reparações.

Na aplicação, é onde cada cliente tem listadas todas as reparações e os respetivos veículos, sendo também possível realizar todas as ações do website na ótica do cliente.

Por fim, para que haja uma interligação de dados entre ambos os componentes referidos anteriormente, procedemos ao desenvolvimento da API REST.

O objetivo, é também que mesmo a nível internacional, se possam registar e verificar o histórico de reparações, centralizando assim todos os dados.

O projeto é comum entre ambas as disciplinas, mas as fases que traçam este projeto são divididas pelas três, conforme a sua pertinência.

Podemos referir também o facto de que o projeto tem como objetivo geral a consolidação de todo o conhecimento adquirido ao longo do curso.

## 2 ENDPOINTS

### 2.1 Cars

#### 2.1.1 Métodos de Implementação

##### Método GET

Este método serve para obter a listagem de todos os veículos registados na plataforma e apenas o administrador consegue utilizar este método.

Relativamente ao Token, este é variável consoante o utilizador.

```
PS C:\Users\diogo> curl http://127.0.0.1:8000/api/cars?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH

StatusCode      : 200
StatusDescription : OK
Content         : [{"id":1,"vin":"1FUJA6CV74DM34063","brand":"FREIGHTLINER","model":"Columbia","color":
                  : "#ff0000","carType":"TRUCK","displacement":12800,"fuelType":"Diesel","registration"
                  : "XX-55-44","modelyear":"2004","...
RawContent      : HTTP/1.1 200 OK
                  Host: 127.0.0.1:8000
                  Connection: close
                  Vary: Accept
                  X-Pagination-Total-Count: 8
                  X-Pagination-Page-Count: 1
                  X-Pagination-Current-Page: 1
                  X-Pagination-Per-Page: 20
                  Link: <http://...
Forms           : {}
Headers         : {[Host, 127.0.0.1:8000], [Connection, close], [Vary, Accept],
                  [X-Pagination-Total-Count, 8]...}
Images          : {}
InputFields     : {}
Links           : {}
ParsedHtml      : mshtml.HTMLDocumentClass
RawContentLength : 2019
```

8

Figura 1 - Cars, Método GET

URL: `curl` [http://127.0.0.1:8000/api/cars?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/cars?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

Cofinanciado por:





## Método POST

Este método serve para adicionar um novo veículo.

```
C:\Users\diogo>curl -X POST -F "vin=JYAVP18E07A005321" -F "brand=YAMAHA" -F "model=XVZ13CTS/XVZ13CTSC/Royal Star" -F "color=#ffe529b6" -F "carType=MOTORCYCLE" -F "displacement=2000" -F "fuelType=Gasoline" -F "registration=45-34-33" -F "modelyear=2010" -F "kilometers=10000" -F "state=Accepted" -F "userId=5" http://127.0.0.1:8000/api/cars?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
{"vin":"JYAVP18E07A005321","brand":"YAMAHA","model":"XVZ13CTS/XVZ13CTSC/Royal Star","color":"#ffe529b6","carType":"MOTORCYCLE","displacement":"2000","fuelType":"Gasoline","registration":"45-34-33","modelyear":"2010","kilometers":"10000","state":"Accepted","userId":"5","id":53}
```

Figura 2 - Cars, Método POST

**URL:** curl -X POST -F "vin=JYAVP18E07A005321" -F "brand=YAMAHA" -F "model=XVZ13CTS/XVZ13CTSC/Royal Star" -F "color=#ffe529b6" -F "carType=MOTORCYCLE" -F "displacement=2000" -F "fuelType=Gasoline" -F "registration=45-34-33" -F "modelyear=2010" -F "kilometers=10000" -F "state=Accepted" -F "userId=5" [http://127.0.0.1:8000/api/cars?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/cars?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método SET - CUSTOM

Este método serve para mostrar uma quantidade específica de dados, neste caso, todos os veículos até um certo limite definido pelo utilizador.

9

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/cars/set/2?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
{"limite":2,"Records":[{"id":1,"vin":"1FUJA6CV74DM34063","brand":"FREIGHTLINER","model":"Columbia","color":"#ff0000","carType":"TRUCK","displacement":12800,"fuelType":"Diesel","registration":"XX-55-44","modelyear":"2004","kilometers":100000,"state":"Pending","userId":2},{id":38,"vin":"2FTJW36M6LCA90573","brand":"FORD","model":"F-350","color":"#ec4116","carType":"TRUCK","displacement":7292.24,"fuelType":"Diesel","registration":"34-65-gh","modelyear":"1990","kilometers":100000,"state":"Accepted","userId":2}]}
```

Figura 3 - Cars, Método SET CUSTOM

**URL:** curl [http://127.0.0.1:8000/api/cars/set/2?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/cars/set/2?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

Cofinanciado por:



## Método TOTAL - CUSTOM

Este método só pode ser executado pelo administrador e serve para obter o número total de veículos registados na plataforma.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/cars/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH  
{"total":7}
```

Figura 4 - Cars, Método TOTAL CUSTOM

URL: curl [http://127.0.0.1:8000/api/cars/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/cars/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método TOTAL USER - CUSTOM

Este método serve para obter o número total de veículos registados na plataforma de um dado utilizador, através da utilização do Access Token.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/cars/totaluser?access-token=hg95zPz_0_IKUJVLNf07jwT7cgE_ad2  
{"total":1}
```

Figura 5 - Cars, Método TOTAL USER CUSTOM

URL: curl [http://127.0.0.1:8000/api/cars/totaluser?access-token=hg95zPz\\_0\\_IKUJVLNf07jwT7cgE\\_ad2](http://127.0.0.1:8000/api/cars/totaluser?access-token=hg95zPz_0_IKUJVLNf07jwT7cgE_ad2)

## Método DELETE

Este método só pode ser executado pelo administrador e serve para eliminar o veículo selecionado pelo mesmo.

```
C:\Users\diogo>curl -X DELETE http://127.0.0.1:8000/api/cars/58?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
```

Figura 6 - Cars, Método DELETE

URL: curl -X DELETE [http://127.0.0.1:8000/api/cars/58?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/cars/58?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

Cofinanciado por:



## Método PUT

Este método serve para atualizar os dados de um veículo selecionado pelo administrador.

```
C:\Users\diogo>curl -X PUT -d "model=teste" http://127.0.0.1:8000/api/cars/58?access-token=fvw_tLyV787AOWs0AfEEQEDkm8xmN9Fk
{"id":58,"vin":"JYAVP18E07A005321","brand":"YAMAHA","model":"teste","color":"#ffe529b6","carType":"MOTORCYCLE","displacement":2000,"fuelType":"Gasoline","registration":"45-34-33","modelyear":"2010","kilometers":10000,"state":"Accepted","userId":2}
```

Figura 7 - Cars, Método PUT

**URL:** `curl -X PUT -d "model=teste"` [http://127.0.0.1:8000/api/cars/58?access-token=fvw\\_tLyV787AOWs0AfEEQEDkm8xmN9Fk](http://127.0.0.1:8000/api/cars/58?access-token=fvw_tLyV787AOWs0AfEEQEDkm8xmN9Fk)

## Método DELETE - CUSTOM

Este método só pode apagar registos de veículos quando o utilizador mencionado no Access Token é o proprietário do mesmo. Este método retorna 1 quando apaga o registo com sucesso e 0 quando o pedido falha.

```
C:\Users\diogo>curl -X DELETE http://127.0.0.1:8000/api/cars/deleted/61?access-token=3YEr44AJGN0CZHIXyvqKp9vYffzw0u1J
{"SaveError":1}
```

Figura 8 - Cars, Método DELETE CUSTOM

**URL:** `curl -X DELETE` <http://127.0.0.1:8000/api/cars/deleted/61?access-token=3YEr44AJGN0CZHIXyvqKp9vYffzw0u1J>

## Método POST – CUSTOM

Este método serve para inserir um registo de um veículo na ótica do cliente.

```
C:\Users\diogo>curl -X POST -F "vin=JYAVP18E07A005321" -F "brand=YAMAHA" -F "model=XVZ13CTS/XVZ13CTSC/Royal Star" -F "color=#ffe529b6" -F "carType=MOTORCYCLE" -F "displacement=2000" -F "fuelType=Gasoline" -F "registration=45-34-33" -F "modelyear=2010" -F "kilometers=10000" -F "state=Accepted" http://127.0.0.1:8000/api/cars/post?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH {"SaveError":true}
```

Figura 9 - Cars, Método POST CUSTOM

**URL:** curl -X POST -F "vin=JYAVP18E07A005321" -F "brand=YAMAHA" -F "model=XVZ13CTS/XVZ13CTSC/Royal Star" -F "color=#ffe529b6" -F "carType=MOTORCYCLE" -F "displacement=2000" -F "fuelType=Gasoline" -F "registration=45-34-33" -F "modelyear=2010" -F "kilometers=10000" -F "state=Accepted" [http://127.0.0.1:8000/api/cars/post?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/cars/post?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método PUT – CUSTOM

Este método serve para editar os dados de um veículo selecionado pelo utilizador. O veículo só pode ser atualizado se o utilizador for o proprietário.

12

```
C:\Users\diogo>curl -X PUT -d "brand=teste" http://127.0.0.1:8000/api/cars/66?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH {"id":66,"vin":"JYAVP18E07A005321","brand":"teste","model":"XVZ13CTS/XVZ13CTSC/Royal Star","color":"#ffe529b6","carType":"MOTORCYCLE","displacement":2000,"fuelType":"Gasoline","registration":"45-34-33","modelyear":"2010","kilometers":10000,"state":"Accepted","userId":2}
```

Figura 10 - Cars, Método PUT CUSTOM

**URL:** curl -X PUT -d "brand=teste" [http://127.0.0.1:8000/api/cars/66?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/cars/66?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## 2.2 Companies

### 2.2.1 Métodos de Implementação

#### Método GET

Este método serve para obter uma lista completa de todas as empresas registadas na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/companies?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
[{"id":1,"companyname":"Midas","nif":"234565678","email":"midas@gmail.com","phonenummer":"261323332","registrationdate":"2021-11-30 17:19:56"}, {"id":2,"companyname":"maisuma","nif":"456768765","email":"maisuma@gmail.com","phonenummer":"123454656","registrationdate":"2021-11-30 17:21:40"}, {"id":3,"companyname":"goierng+","nif":"345654567","email":"egfwfew@goerbger.com","phonenummer":"345456767","registrationdate":"2021-11-30 17:24:26"}, {"id":4,"companyname":"teste","nif":"143546543","email":"teste@teste.com","phonenummer":"434565456","registrationdate":"2021-12-19 10:51:39"}, {"id":5,"companyname":"teste2","nif":"143516543","email":"teste2@teste.com","phonenummer":"434566456","registrationdate":"2021-12-19 10:52:34"}]
```

Figura 11 - Companies, Método GET

URL: `curl` [http://127.0.0.1:8000/api/companies?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/companies?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

13

#### Método POST

Este método serve para registar uma nova empresa na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl -X POST -F "companyname=SobreRodas" -F "nif=656787654" -F "email=sobrerodas@gmail.com" -F "phonenummer=261567434" http://127.0.0.1:8000/api/companies?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
{"companyname":"SobreRodas","nif":"656787654","email":"sobrerodas@gmail.com","phonenummer":"261567434","id":6}
```

Figura 12 - Companies, Método POST

URL: `curl -X POST -F "companyname=SobreRodas" -F "nif=656787654" -F "email=sobrerodas@gmail.com" -F "phonenummer=261567434"`  
[http://127.0.0.1:8000/api/companies?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/companies?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método PUT

Este método serve para editar um registo de uma empresa na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl -X PUT -d "companyname=SobreRodinhas" http://127.0.0.1:8000/api/companies/6?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH
{"id":6,"companyname":"SobreRodinhas","nif":"656787654","email":"sobrerodas@gmail.com","phonenumber":"261567434","registrationdate":"2021-12-27 16:36:50"}
```

Figura 13 - Companies, Método PUT

**URL:** curl -X PUT -d "companyname=SobreRodinhas"  
[http://127.0.0.1:8000/api/companies/6?access-token=USg4v13yHb5KgxSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/companies/6?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH)

## Método DELETE

Este método serve para eliminar um registo de uma empresa na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl -X DELETE http://127.0.0.1:8000/api/companies/6?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH
```

Figura 14 - Companies, Método DELETE

**URL:** curl -X DELETE [http://127.0.0.1:8000/api/companies/6?access-token=USg4v13yHb5KgxSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/companies/6?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH)

## Método TOTAL – CUSTOM

Este método serve para obter o número total de empresas registadas na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/companies/total?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH
{"total":6}
```

Figura 15 - Companies, Método TOTAL CUSTOM

URL: curl [http://127.0.0.1:8000/api/companies/total?access-token=USg4v13yHb5KgxSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/companies/total?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH)

## Método SET – CUSTOM

Este método serve para obter uma lista até um dado limite das empresas registadas na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/companies/set/2?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH
{"limite":"2","Records":[{"id":1,"companyname":"Midas","nif":"234565678","email":"midas@gmail.com","phonenumber":"261323332","registrationdate":"2021-11-30 17:19:56"}, {"id":2,"companyname":"maisuma","nif":"456768765","email":"maisuma@gmail.com","phonenumber":"123454656","registrationdate":"2021-11-30 17:21:40"}]}
```

Figura 16 - Companies, Método SET CUSTOM

URL: curl [http://127.0.0.1:8000/api/companies/set/2?access-token=USg4v13yHb5KgxSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/companies/set/2?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH)

## 2.3 CompaniesList

### 2.3.1 Métodos de Implementação

#### Método GET

Este método serve para obter uma lista completa de todas as empresas registadas na plataforma e pode ser executado por qualquer utilizador, incluído o guest. Não requer Access Token.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/companieslist
[{"id":1,"companyname":"Midas","nif":"234565678","email":"midas@gmail.com","phonenummer":"261323332","registrationdate":"2021-11-30 17:19:56"}, {"id":2,"companyname":"maisuma","nif":"456768765","email":"maisuma@gmail.com","phonenummer":"123454656","registrationdate":"2021-11-30 17:21:40"}, {"id":3,"companyname":"goierng+","nif":"345654567","email":"egfwfew@goerbger.com","phonenummer":"345456767","registrationdate":"2021-11-30 17:24:26"}, {"id":4,"companyname":"teste","nif":"143546543","email":"teste@teste.com","phonenummer":"434565456","registrationdate":"2021-12-19 10:51:39"}, {"id":5,"companyname":"teste2","nif":"143516543","email":"teste2@teste.com","phonenummer":"434566456","registrationdate":"2021-12-19 10:52:34"}, {"id":7,"companyname":"SobreRodas","nif":"656787654","email":"sobrerodas@gmail.com","phonenummer":"261567434","registrationdate":"2021-12-27 16:42:38"}]
```

Figura 17 - CompaniesList, Método GET

URL: curl <http://127.0.0.1:8000/api/companieslist>



## 2.4 Contributors

### 2.4.1 Métodos de Implementação

#### Método GET

Este método serve para obter uma lista completa de todos os colaboradores registados na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/contributors?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
[{"id":2,"speciality":"Mechanical","companyId":1,"userId":3},{ "id":3,"speciality":"Locksmith","companyId":2,"userId":5},{ "id":4,"speciality":"Mechanical","companyId":3,"userId":6}]
```

Figura 18 - Contributors, Método GET

URL                      curl                      [http://127.0.0.1:8000/api/contributors?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/contributors?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

17

#### Método POST

Este método serve para registar um novo colaborador na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl -X POST -F "speciality=Painter" -F "companyId=1" -F "userId=2" http://127.0.0.1:8000/api/contributors?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
{"speciality":"Painter","companyId":"1","userId":"2","id":5}
```

Figura 19 - Contributors, Método POST

URL: curl -X POST -F "speciality=Painter" -F "companyId=1" -F "userId=2" [http://127.0.0.1:8000/api/contributors?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/contributors?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

Cofinanciado por:



## Método PUT

Este método serve para editar um registo de um colaborador na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl -X PUT -d "speciality=Electrician" http://127.0.0.1:8000/api/contributors/5?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
```

Figura 20 - Contributors, Método PUT

**URL:** curl -X PUT -d "speciality=Electrician"  
[http://127.0.0.1:8000/api/contributors/5?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/contributors/5?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método DELETE

Este método serve para eliminar um registo de um colaborador na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl -X DELETE http://127.0.0.1:8000/api/contributors/5?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
```

18

Figura 21 - Contributors, Método DELETE

**URL:** curl -X DELETE [http://127.0.0.1:8000/api/contributors/5?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/contributors/5?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

Cofinanciado por:



## Método SET – CUSTOM

Este método serve para obter uma lista até um dado limite dos colaboradores registados na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/contributors/set/2?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
{"limite": "2", "Records": [{"id": 2, "speciality": "Mechanical", "companyId": 1, "userId": 3}, {"id": 3, "speciality": "Locksmith", "companyId": 2, "userId": 5}]}
```

Figura 22 - Contributors, Método SET CUSTOM

URL: `curl` [http://127.0.0.1:8000/api/contributors/set/2?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/contributors/set/2?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método TOTAL – CUSTOM

Este método serve para obter o número total de colaboradores registados na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/contributors/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
{"total": 3}
```

Figura 23 - Contributors, Método TOTAL CUSTOM

URL: `curl` [http://127.0.0.1:8000/api/contributors/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/contributors/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## 2.5 Login

### 2.5.1 Métodos de Implementação

#### Método DO – CUSTOM

Este método serve para validar a autenticação dos utilizadores e retornar os seus dados, de forma que este receba o token para ter acesso aos outros métodos. Além disso, este método também retorna a informação da ultima reparação efetuada a um veiculo desse utilizador, guardada no serviço Mosquitto, para que quando um utilizador fizer login este possa receber logo a informação de como se encontra o seu veiculo na oficina. Caso não haja essa informação no serviço Mosquitto, será enviado uma chave chamada “repair” com o valor true.

```
C:\Users\diogo>curl -X POST -F "username=diogo" -F "password=12345678" http://127.0.0.1:8000/api/login/do
{"user":{"id":1,"username":"Diogo","password_hash":"$2y$13$9xhIvL/N.YFLH6mramOBwu7arDerLmRVC2JhVctUdZphvsk.Ybzbq","verification_token":"XJe766xRVUZxXNo6ZZGNPGwSD7DyEZcC_1639000307","password_reset_token":null,"auth_key":"NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7","status":10,"updated_at":1639000307,"created_at":1639000307,"usertype":"admin","nif":"132132132","birthday":"2021-12-08","email":"diogo.rc.pereira@gmail.com","phonenumber":"911162402","registrationdate":"2021-12-08 21:51:47"},"repair":true}
```

Figura 24 - Login, Método DO CUSTOM

**URL:** `curl -X POST -F "username=diogo" -F "password=12345678" http://127.0.0.1:8000/api/login/do`

## 2.6 Repairs

### 2.6.1 Métodos de Implementação

#### Método GET

Este método serve para obter uma lista completa de todas as reparações registadas na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/repairs?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
[{"id":1,"kilometers":130,"repairdate":"2021-12-07 18:01:11","repairdescription":"teste","state":"Pending","repairtype":"Repair","carId":1,"contributorId":2},{ "id":2,"kilometers":300000,"repairdate":"2021-12-16 15:22:26","repairdescription":"maisum","state":"To Repair","repairtype":"Maintenance","carId":1,"contributorId":3},{ "id":3,"kilometers":300000,"repairdate":"2021-12-16 15:22:48","repairdescription":"maisum","state":"To Repair","repairtype":"Repair","carId":40,"contributorId":3},{ "id":4,"kilometers":300000,"repairdate":"2021-12-16 15:22:56","repairdescription":"maisum","state":"To Repair","repairtype":"Repair","carId":43,"contributorId":3},{ "id":5,"kilometers":130,"repairdate":"2021-12-17 09:55:38","repairdescription":"mais um teste","state":"Repaired","repairtype":"Repair","carId":42,"contributorId":4},{ "id":6,"kilometers":120000,"repairdate":"2021-12-17 11:35:35","repairdescription":"teste","state":"Under Repair","repairtype":"Repair","carId":1,"contributorId":4},{ "id":7,"kilometers":130,"repairdate":"2021-12-17 11:40:30","repairdescription":"Mosca","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":8,"kilometers":12000,"repairdate":"2021-12-17 11:41:13","repairdescription":"mosquinha","state":"To Repair","repairtype":"Maintenance","carId":42,"contributorId":4},{ "id":9,"kilometers":100000,"repairdate":"2021-12-17 11:45:33","repairdescription":"Claudio","state":"Under Repair","repairtype":"Maintenance","carId":43,"contributorId":4},{ "id":11,"kilometers":100,"repairdate":"2021-12-17 17:25:52","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":12,"kilometers":100,"repairdate":"2021-12-17 17:26:16","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":13,"kilometers":100,"repairdate":"2021-12-17 17:27:15","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":14,"kilometers":100,"repairdate":"2021-12-17 17:27:35","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":15,"kilometers":100,"repairdate":"2021-12-17 17:32:20","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":16,"kilometers":100,"repairdate":"2021-12-17 17:42:30","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":17,"kilometers":100,"repairdate":"2021-12-17 17:43:31","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{ "id":18,"kilometers":100,"repairdate":"2021-12-17 17:45:36","repairdescription":"teste","state":"Repaired","repairtype":"Maintenance","carId":40,"contributorId":4},{ "id":19,"kilometers":100,"repairdate":"2021-12-17 17:50:16","repairdescription":"teste","state":"Repaired","repairtype":"Maintenance","carId":40,"contributorId":4},{ "id":20,"kilometers":130,"repairdate":"2021-12-17 18:01:26","repairdescription":"pneu","state":"Under Repair","repairtype":"Repair","carId":1,"contributorId":2},{ "id":21,"kilometers":130,"repairdate":"2021-12-17 18:12:39","repairdescription":"teste","state":"Under Repair","repairtype":"Maintenance","carId":1,"contributorId":2}]
```

21

Figura 25 - Repairs, Método GET

URL: [http://127.0.0.1:8000/api/repairs?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/repairs?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

[http://127.0.0.1:8000/api/repairs?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/repairs?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

Cofinanciado por:



## Método HISTORY – CUSTOM

Este método serve para obter uma lista completa de todas as reparações de um determinado veículo registado na plataforma e apenas pode ser executado pelo proprietário do veículo.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/repairs/history/1?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH
[{"id":1,"kilometers":130,"repairdate":"2021-12-07 18:01:11","repairdescription":"teste","state":"Pending","repairtype":"Repair","carId":1,"contributorId":2},{ "id":2,"kilometers":300000,"repairdate":"2021-12-16 15:22:26","repairdescription":"maisum","state":"To Repair","repairtype":"Maintenance","carId":1,"contributorId":3},{ "id":6,"kilometers":120000,"repairdate":"2021-12-17 11:35:35","repairdescription":"teste","state":"Under Repair","repairtype":"Repair","carId":1,"contributorId":4},{ "id":20,"kilometers":130,"repairdate":"2021-12-17 18:01:26","repairdescription":"pneu","state":"Under Repair","repairtype":"Repair","carId":1,"contributorId":2},{ "id":21,"kilometers":130,"repairdate":"2021-12-17 18:12:39","repairdescription":"teste","state":"Under Repair","repairtype":"Maintenance","carId":1,"contributorId":2},{ "id":22,"kilometers":130,"repairdate":"2021-12-17 18:15:14","repairdescription":"teste","state":"Under Repair","repairtype":"Maintenance","carId":1,"contributorId":2}]
```

Figura 26 - Repairs, Método HISTORY CUSTOM

URL: curl [http://127.0.0.1:8000/api/repairs/history/1?access-token=USg4v13yHb5KgxSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/repairs/history/1?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH)

22

## Método REPAIR CONTRIBUTOR – CUSTOM

Este método serve para retornar todas as reparações de um determinado colaborador e só pode ser executado por este mesmo colaborador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/repairs/repaircontributor?access-token=hg95zPz_0_IKUJVLrnf07jwT7cgE_ad2
[{"id":2,"kilometers":300000,"repairdate":"2021-12-16 15:22:26","repairdescription":"maisum","state":"To Repair","repairtype":"Maintenance","carId":1,"contributorId":3},{ "id":3,"kilometers":300000,"repairdate":"2021-12-16 15:22:48","repairdescription":"maisum","state":"To Repair","repairtype":"Repair","carId":40,"contributorId":3},{ "id":4,"kilometers":300000,"repairdate":"2021-12-16 15:22:56","repairdescription":"maisum","state":"To Repair","repairtype":"Repair","carId":43,"contributorId":3}]
```

Figura 27 - Repairs, Método REPAIR CONTRIBUTOR CUSTOM

URL: curl [http://127.0.0.1:8000/api/repairs/repaircontributor?access-token=hg95zPz\\_0\\_IKUJVLrnf07jwT7cgE\\_ad2](http://127.0.0.1:8000/api/repairs/repaircontributor?access-token=hg95zPz_0_IKUJVLrnf07jwT7cgE_ad2)

Cofinanciado por:



## Método POST – CUSTOM

Este método serve para registar uma nova reparação na plataforma e apenas pode ser executado por um colaborador.

```
C:\Users\diogo>curl -X POST -F "kilometers=200000" -F "repairdate=2022-01-02" -F "repairdescription=Reparacao de teste" -F "state=Repaired" -F "repairtype=Maintenance" -F "carId=40" http://127.0.0.1:8000/api/repairs/post/?access-token=fvw_tLyV787AOWs0AfEEQEDkm8xmN9Fk {"Save":true}
```

Figura 28 - Repairs, Método POST CUSTOM

**URL:** `curl -X POST -F "kilometers=200000" -F "repairdate=2022-01-02" -F "repairdescription=Reparacao de teste" -F "state=Repaired" -F "repairtype=Maintenance" -F "carId=40"`  
[http://127.0.0.1:8000/api/repairs/post/?access-token=fvw\\_tLyV787AOWs0AfEEQEDkm8xmN9Fk](http://127.0.0.1:8000/api/repairs/post/?access-token=fvw_tLyV787AOWs0AfEEQEDkm8xmN9Fk)

## Método PUT – CUSTOM

Este método serve para alterar uma reparação na plataforma e apenas pode ser executado pelo colaborador que a criou.

```
C:\Users\diogo>curl -X PUT -d "repairdescription=Reparacao Alterada" http://127.0.0.1:8000/api/repairs/put/26?access-token=hg95zPz_0_IKUJVLrnF07jwT7cgE_ad2 {"Save":true}
```

Figura 29 - Repairs, Método PUT CUSTOM

**URL:** `curl -X PUT -d "repairdescription=Reparacao Alterada"`  
[http://127.0.0.1:8000/api/repairs/put/26?access-token=hg95zPz\\_0\\_IKUJVLrnF07jwT7cgE\\_ad2](http://127.0.0.1:8000/api/repairs/put/26?access-token=hg95zPz_0_IKUJVLrnF07jwT7cgE_ad2)

## Método DELETED – CUSTOM

Este método serve para apagar uma reparação na plataforma e apenas pode ser executado pelo colaborador que a criou.

```
C:\Users\diogo>curl -X DELETE http://127.0.0.1:8000/api/repairs/deleted/26?access-token=hg95zPz_0_IKUJVLrnf07jwT7cgE_ad2
{"Del":1}
```

Figura 30 - Repairs, Método DELETED CUSTOM

**URL:** curl -X DELETE [http://127.0.0.1:8000/api/repairs/deleted/26?access-token=hg95zPz\\_0\\_IKUJVLrnf07jwT7cgE\\_ad2](http://127.0.0.1:8000/api/repairs/deleted/26?access-token=hg95zPz_0_IKUJVLrnf07jwT7cgE_ad2)

## Método TOTAL – CUSTOM

Este método serve para dar a informação ao administrador de quantas reparações foram registadas na plataforma.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/repairs/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH
{"total":23}
```

24

Figura 31 - Repairs, Método TOTAL CUSTOM

**URL:** curl [http://127.0.0.1:8000/api/repairs/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/repairs/total?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

Cofinanciado por:





## Método SET – CUSTOM

Este método serve para obter uma lista até um dado limite das reparações registadas na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\diogo>curl http://127.0.0.1:8000/api/repairs/set/10?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH
{"limite":10,"Records":[{"id":1,"kilometers":130,"repairdate":"2021-12-07 18:01:11","repairdescription":"teste","state":"Pending","repairtype":"Repair","carId":1,"contributorId":2},{id":2,"kilometers":300000,"repairdate":"2021-12-16 15:22:26","repairdescription":"maisum","state":"To Repair","repairtype":"Maintenance","carId":1,"contributorId":3},{id":3,"kilometers":300000,"repairdate":"2021-12-16 15:22:48","repairdescription":"maisum","state":"To Repair","repairtype":"Repair","carId":40,"contributorId":3},{id":4,"kilometers":300000,"repairdate":"2021-12-16 15:22:56","repairdescription":"maisum","state":"To Repair","repairtype":"Repair","carId":43,"contributorId":3},{id":5,"kilometers":130,"repairdate":"2021-12-17 09:55:38","repairdescription":"mais um teste","state":"Repaired","repairtype":"Repair","carId":42,"contributorId":4},{id":6,"kilometers":120000,"repairdate":"2021-12-17 11:35:35","repairdescription":"teste","state":"Under Repair","repairtype":"Repair","carId":1,"contributorId":4},{id":7,"kilometers":130,"repairdate":"2021-12-17 11:40:30","repairdescription":"Mosca","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4},{id":8,"kilometers":12000,"repairdate":"2021-12-17 11:41:13","repairdescription":"mosquinha","state":"To Repair","repairtype":"Maintenance","carId":42,"contributorId":4},{id":9,"kilometers":100000,"repairdate":"2021-12-17 11:45:33","repairdescription":"Reparacao de teste","state":"Under Repair","repairtype":"Maintenance","carId":43,"contributorId":4},{id":11,"kilometers":100,"repairdate":"2021-12-17 17:25:52","repairdescription":"teste","state":"Repaired","repairtype":"Repair","carId":40,"contributorId":4}]}
```

Figura 32 - Repairs, Método SET CUSTOM

URL: `curl` [http://127.0.0.1:8000/api/repairs/set/10?access-token=USg4v13yHb5KgxSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/repairs/set/10?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH)

## 2.7 Schedules

### 2.7.1 Métodos de Implementação

#### Método TOTAL – CUSTOM

Este método serve para obter o número total de agendamentos registados na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\badal>curl http://127.0.0.1:8000/api/schedules/total?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7  
{"total":2}
```

Figura 33 - Schedules, Método TOTAL CUSTOM

URL: curl <http://127.0.0.1:8000/api/schedules/total?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7>

#### Método SET – CUSTOM

Este método serve para obter uma lista até a um dado limite dos agendamentos registados na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\badal>curl http://127.0.0.1:8000/api/schedules/set/1?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7  
{  
  "limite": "1",  
  "Records": [  
    {  
      "id": 1,  
      "currentdate": "2021-11-30 17:27:36",  
      "schedulingdate": "2021-11-30 17:27:00",  
      "repairdescription": "Manutenção",  
      "state": "Pending",  
      "repairtype": "Maintenance",  
      "carId": 2,  
      "companyId": 1  
    }  
  ]  
}
```

Figura 34 - Schedules, Método SET CUSTOM

URL: curl <http://127.0.0.1:8000/api/schedules/set/1?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7>

## Método POST – CUSTOM

Este método serve para adicionar um novo agendamento na plataforma e pode ser executado por todos os utilizadores à exceção do guest.

```
C:\Users\diogo>curl -X POST -F "schedulingdate=2022-01-02 15:00:00" -F "repairdescription=teste" -F "repairtype=Pneus" -F "carId=66" -F "companyId=1" http://127.0.0.1:8000/api/schedules/post?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH {"Save":true}
```

Figura 35 - Schedules, Método POST CUSTOM

**URL:** curl -X POST -F "schedulingdate=2022-01-02 15:00:00" -F "repairdescription=teste" -F "repairtype=Pneus" -F "carId=66" -F "companyId=1" [http://127.0.0.1:8000/api/schedules/post?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/schedules/post?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método DELETE CLIENT – CUSTOM

Este método serve para eliminar um registo de um agendamento na plataforma e apenas pode ser executado pelo responsável da marcação do agendamento.

```
C:\Users\diogo>curl -X DELETE http://127.0.0.1:8000/api/schedules/deleteclient/3?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH {"DEL":1}
```

Figura 36 - Schedules, Método DELETE CLIENT CUSTOM

**URL:** curl -X DELETE [http://127.0.0.1:8000/api/schedules/deleteclient/3?access-token=USg4v13yHb5KgXSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/schedules/deleteclient/3?access-token=USg4v13yHb5KgXSp7PbEbWGTg_ABVATH)

## Método DELETE CONTRIBUTOR – CUSTOM

Este método serve para eliminar um registo de um agendamento na plataforma e apenas pode ser executado pelo responsável da empresa do agendamento.

```
C:\Users\diogo>curl -X DELETE http://127.0.0.1:8000/api/schedules/deletecontributor/2?access-token=hg95zPz_0_IKUJVLNf07jwT7cgE_ad2
```

Figura 37 - Schedules, Método DELETE CONTRIBUTOR CUSTOM

**URL:** curl -X DELETE [http://127.0.0.1:8000/api/schedules/deletecontributor/2?access-token=USg4v13yHb5KgxSp7PbEbWGTg\\_ABVATH](http://127.0.0.1:8000/api/schedules/deletecontributor/2?access-token=USg4v13yHb5KgxSp7PbEbWGTg_ABVATH)

## Método PUT CLIENT – CUSTOM

Este método serve para editar um registo de um agendamento na plataforma e apenas pode ser executado pelo responsável da marcação do agendamento.

```
C:\Users\badal>curl -X PUT -d "schedulingdate=2022-10-10 14:00:00" -d "repairdescription=Maintenance" -d "repairtype=Manuntencao" -d "carId=2" http://127.0.0.1:8000/api/schedules/putclient/1?access-token=NL1pWC3C2QNQdDLZWWhFbTMuceMhjz7p7{"PUT":true}
```

Figura 38 - Schedules, Método PUT CLIENT CUSTOM

**URL:** curl -X PUT -d "schedulingdate=2022-10-10 14:00:00" -d "repairdescription=Maintenance" -d "repairtype=Manuntencao" -d "carId=2" <http://127.0.0.1:8000/api/schedules/putclient/1?access-token=NL1pWC3C2QNQdDLZWWhFbTMuceMhjz7p7>

## Método PUT CONTRIBUTOR – CUSTOM

Este método serve para editar um registo de um agendamento na plataforma e apenas pode ser executado pelo responsável da empresa do agendamento.

```
C:\Users\badal>curl -X PUT -d "schedulingdate=2022-10-10 14:00:00" -d "state=Accepted" http://127.0.0.1:8000/api/schedules/putcontributor/1?access-token=5ft7TRO-AFQVty3EjGTSRnqkAjUbONKq  
{"PUT":true}
```

*Figura 39 - Schedules, Método PUT CONTRIBUTOR CUSTOM*

**URL:** `curl -X PUT -d "schedulingdate=2022-10-10 14:00:00" -d "state=Accepted"`  
<http://127.0.0.1:8000/api/schedules/putcontributor/1?access-token=5ft7TRO-AFQVty3EjGTSRnqkAjUbONKq>

## 2.8 Signup

### 2.8.1 Métodos de Implementação

#### Método POST – CUSTOM

Este método serve para registar qualquer tipo de utilizador, não sendo obrigatório o Access Token.

```
C:\Users\badal>C:\ProgramData\chocolatey\bin\curl.exe -X POST -F "username=goncalo12345678" -F "password=12345678" -F "nif=123123000" -F "email=goncalo12345678@hotmail.com" -F "phonenumber=912123098" -F "birsthday=2001-10-04" http://127.0.0.1:8000/api/signup/post {"message":true}{"Save":true}
```

*Figura 40 - Signup, Método POST CUSTOM*

**URL:** curl -X POST -F "username=goncalo12345678" -F "password=12345678" -F "nif=123123000" -F "email=goncalo12345678@hotmail.com" -F "phonenumber=912123098" -F "birsthday=2001-10-04" <http://127.0.0.1:8000/api/signup/post>

## 2.9 User

### 2.9.1 Métodos de Implementação

#### Método ACCOUNT – CUSTOM

Este método serve para obter os dados da conta do utilizador associado ao Token utilizado no pedido. Este método requer Token e não está disponível para utilizadores guest.

```
C:\Users\badal>C:\ProgramData\chocolatey\bin\curl.exe http://127.0.0.1:8000/api/user/account?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7
{"id":1,"username":"Goncalo","password_hash":"$2y$13$9xhIvL/N.YFLH6mramOBwu7arDerLmRveC2JhVctUdZphvsk.Ybzq","verification_token":"XJe766xRVUZxXNo6ZZGNPGwSD7DyEZcC_1639000307","password_reset_token":null,"auth_key":"NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7","status":10,"updated_at":1639000307,"created_at":1639000307,"usertype":"admin","nif":"132132132","birthday":"2021-12-08","email":"badalha81@gmail.com","phonenumber":"966186487","registrationdate":"2021-12-08 21:51:47"}
```

Figura 41 - User, Método ACCOUNT CUSTOM

URL: `curl` <http://127.0.0.1:8000/api/user/account?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7>

31

#### Método TOTAL – CUSTOM

Este método serve para obter o número total de utilizadores registados na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\badal>C:\ProgramData\chocolatey\bin\curl.exe http://127.0.0.1:8000/api/user/total?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7
{"total":5}
```

Figura 42 - User, Método TOTAL CUSTOM

URL: `curl` <http://127.0.0.1:8000/api/user/total?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7>

## Método SET – CUSTOM

Este método serve para obter uma lista até a um dado limite dos utilizadores registados na plataforma e apenas pode ser executado pelo administrador.

```
C:\Users\badal>C:\ProgramData\chocolatey\bin\curl.exe http://127.0.0.1:8000/api/user/set/2?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7
{"limite":2,"Records":[{"id":1,"username":"Goncalo","password_hash":"$2y$13$9xhIvL/N.YFLH6mramOBwu7arDerLmRveC2JhVctUdZphvsk.Ybzq","verification_token":"XJe766xRVUZxXNo6ZZGNPGwSD7DyEZcC_1639000307","password_reset_token":null,"auth_key":"NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7","status":10,"updated_at":1639000307,"created_at":1639000307,"usertype":"admin","nif":"132132132","birsthday":"2021-12-08","email":"badalha81@gmail.com","phonenumber":"966186487","registrationdate":"2021-12-08 21:51:47"}, {"id":14,"username":"Goncalo22","password_hash":"$2y$13$.NN89Wo69kHH9A3vLKjCpuEwVuz7kXQkLYPSwc/Lhvz/Gy9/P7it2","verification_token":"s6I4uBFko6A7aq1ssfYgSPZTR65vzb9_1639000365","password_reset_token":null,"auth_key":"5ft7TR0-AFQVty3EjGTSRnqkAjUbONKq","status":10,"updated_at":1639000365,"created_at":1639000365,"usertype":"collaborator","nif":"345543567","birsthday":"2021-12-08","email":"badalha81@hotmail.com","phonenumber":"966186487","registrationdate":"2021-12-08 21:52:45"}]}
```

Figura 43 - User, Método SET CUSTOM

**URL:** curl <http://127.0.0.1:8000/api/user/set/2?access-token=NL1pWC3C2QNQdDLZWhFbTMuceMhjz7p7>

## Método DELETED - CUSTOM

Este método serve para eliminar um registo de um utilizador na plataforma e pode ser executado por qualquer utilizador, exceto os guest, visto que não possuem Access Token. Não requer o Id do utilizador, visto que o método utiliza o Access Token do utilizador em questão.

32

```
C:\Users\badal>C:\ProgramData\chocolatey\bin\curl.exe -X DELETE http://127.0.0.1:8000/api/user/deleted?access-token=55B09BxZPtVY5Q-UMjVA7GJ4J5F_3qvG
{"DelError":1}
```

Figura 44 - User, Método DELETED CUSTOM

**URL:** curl -X DELETE [http://127.0.0.1:8000/api/user/deleted?access-token=55B09BxZPtVY5Q-UMjVA7GJ4J5F\\_3qvG](http://127.0.0.1:8000/api/user/deleted?access-token=55B09BxZPtVY5Q-UMjVA7GJ4J5F_3qvG)



## Método PUT - CUSTOM

Este método serve para editar um registo de um utilizador na plataforma e pode ser executado por todos os utilizadores, exceto o guest.

```
C:\Users\badal>C:\ProgramData\chocolatey\bin\curl.exe -X PUT -d "email=goncalo1234@gmail.pt" -d "password=12345678" http://127.0.0.1:8000/api/user/put?access-token=60Si98cxQoBRkcUSjk4drckQDeAKPaZM
{"SaveError1":{"id":16,"username":"Goncalo1234","password_hash":"$2y$13$vsYqQs43BhRhc/Nez066LOC/Zs/B.I3j7rJk6sPWhfLK0ZCzmyIom","verification_token":"V-VqdjF0rjlsbHTRtE_LLR-UcuvrMHzt_1639669226","password_reset_token":null,"auth_key":"60Si98cxQoBRkcUSjk4drckQDeAKPaZM","status":10,"updated_at":1640794910,"created_at":1639669226,"usertype":"collaborator","nif":"123123765","birsthday":"2021-12-16","email":"goncalo1234@gmail.pt","phonenumber":"912123123","registrationdate":"2021-12-16 15:40:26"}}
```

Figura 45 - User, Método PUT CUSTOM

**URL:** `curl -X PUT -d "email=goncalo1234@gmail.pt" -d "password=12345678"`  
<http://127.0.0.1:8000/api/user/put?access-token=60Si98cxQoBRkcUSjk4drckQDeAKPaZM>

## 2.10 Informações

Alguns dos exemplos dos métodos referidos anteriormente necessitam, previamente, das credencias de autenticação, neste caso o access token do utilizador. Para esse efeito, para voltar a executar cada um desses métodos, o access token terá de ser alterado para um token de um utilizador que já se encontre registado na base de dados do projeto.

**Exemplo:** `curl -X PUT -d "email=goncalo1234@gmail.pt" -d "password=12345678"`  
<http://127.0.0.1:8000/api/user/put?access-token=60Si98cxQoBRkcUSjk4drckQDeAKPaZM>

**Explicação:** A zona a negrito do URL deve ser alterada para um novo token, salvo se, esse token estiver associado a um user já registado na base de dados.

O mesmo acontece com alguns métodos que necessitem do campo ID de um dado objeto. Por exemplo, alguns métodos de PUT ou Delete precisam do ID do objeto que queremos alterar, e mais uma vez esse ID terá que ser alterado para o ID de um objeto registado nessa tabela especifica.

**Exemplo:** `curl -X PUT -d "schedulingdate=2022-10-10 14:00:00" -d "state=Accepted"`  
<http://127.0.0.1:8000/api/schedules/putcontributor/1?access-token=5ft7TRO-AFQVty3EjGTSRngkAjUbONKq>

**Explicação:** A zona a negrito do URL deve ser alterada para um novo ID, salvo se, esse ID estiver associado a um objeto desse tipo já registado na base de dados.

## 3 MOSQUITTO

### 3.1 Introdução ao Mosquitto

O Mosquitto é um serviço que utiliza o protocolo MQTT. O MQTT (Message Queuing Telemetry Transport), é um protocolo de mensagens leve para sensores e pequenos dispositivos móveis otimizado para redes TCP/IP. O esquema de troca de mensagens é fundamentado no modelo Publish-Subscribe, sendo extremamente simples e leve.

### 3.2 Aplicação Prática

Tendo em conta o nosso projeto, o caso de uso do Mosquitto foi no modelo das reparações, intercetando a gravação do modelo das reparações utilizando a função aftersave.

Como se pode ver na figura abaixo, o objetivo desta função é publicar uma mensagem onde constem algumas das propriedades do objeto reparação, como por exemplo, a data e o estado.

35

```
//Mosquitto
public function afterSave($insert, $changedAttributes)
{
    parent::afterSave($insert, $changedAttributes);

    $message = $this->car->id.":::". $this->car->brand . " " . $this->car->model . ":::is " . $this->state;

    if ($insert) {
        $this->FazPublish( canal: "REPAIR-".$this->car->userId, $message);
    } else {
        $this->FazPublish( canal: "REPAIR-".$this->car->userId, $message);
    }
}
```

Figura 46 - Mosquitto, Função afterSave

Por fim, é chamada a função fazpublish, onde indicamos o nome do canal e a mensagem que queremos passar. O nome do canal será “REPAIR-X”, onde X representa o Id do utilizador do carro que está relacionado com esta reparação. A mensagem é constituída pelo JSON do objeto criado anteriormente, onde constam algumas das propriedades do objeto reparação, como por exemplo, a data e o estado.

Como podemos notar na figura abaixo, a função fazpublish vai publicar a mensagem descrita anteriormente, no seu respetivo canal.

```
public function FazPublish($canal, $msg)
{
    $server = "127.0.0.1";
    $port = 1883;
    $username = ""; // set your username
    $password = ""; // set your password
    $client_id = "phpMQTT-publisher"; // unique!
    $mqtt = new phpMQTT($server, $port, $client_id);
    if ($mqtt->connect( clean: true, will: NULL, $username, $password)) {
        $mqtt->publish($canal, $msg, qos: 0, retain: true);
        $mqtt->close();
    } else {
        file_put_contents( filename: 'debug.output', data: 'Time out!');
    }
}
```

Figura 47- Mosquitto, Função FazPublish

A última mensagem fica guardada para posteriormente ser consultada pelo cliente, tendo em conta que, no método PUBLISH, a propriedade retain está definida como true.

Todo este processo é realizado nos modelos da API, e é também repetido no modelo de agendamentos.

Por fim, como podemos analisar na imagem abaixo, temos a conexão na nossa aplicação móvel, em Android, ao servidor MQTT, que caso dê sucesso procede à subscrição no canal.

```
public static void connectionMQTTRepair(Context context) {

    String clientId = MqttClient.generateClientId();
    MqttAndroidClient client = new MqttAndroidClient(context, serverURI: "tcp://10.0.2.2:1883", clientId);

    try {
        IMqttToken token = client.connect();
        token.setActionCallback(new IMqttActionListener() {
            @Override
            public void onSuccess(IMqttToken asyncActionToken) {
                // We are connected
                Log.d( tag: "MQTT", msg: "onSuccess");
                subscribeMQTTRepair(client, context);
            }

            @Override
            public void onFailure(IMqttToken asyncActionToken, Throwable exception) {
                // Something went wrong e.g. connection timeout or firewall problems
                Log.d( tag: "MQTT", msg: "onFailure");
            }
        });
    } catch (MqttException e) {
        e.printStackTrace();
    }
}
```

Figura 48 - MQTT Conexão Móvel

Como podemos analisar na imagem abaixo, é neste código que fazemos a subscrição do canal anteriormente fundamentado, e onde se procede à criação de uma notificação na aplicação móvel.

```
private static void subscribeMQTTRepair(MqttAndroidClient client, Context context) {
    String topic = "REPAIR-" + LoginSingleton.getInstance(context).getLogin().getId();
    int qos = 0;
    try {
        client.subscribe(topic, qos);
        client.setCallback(new MqttCallback() {
            @Override
            public void connectionLost(Throwable cause) { Log.d("MQTT", "msg: \"Connection Lost\"); }

            @Override
            public void messageArrived(String topic, MqttMessage message) throws Exception {
                String mensagem = new String(message.getPayload());
                String[] separated = mensagem.split("::");
                NotificationCompat.Builder builder = new NotificationCompat.Builder(context, "VehicleRepair");
                builder.setContentTitle(separated[1]);
                builder.setContentText(separated[2]);
                builder.setSmallIcon(R.drawable.ic_car_repair);
                builder.setAutoCancel(true);

                NotificationManagerCompat managerCompat = NotificationManagerCompat.from(context);
                managerCompat.notify(1 + Integer.parseInt(separated[0]), builder.build());
            }

            @Override
            public void deliveryComplete(IMqttDeliveryToken token) {
            }
        });
    } catch (MqttException e) {
        e.printStackTrace();
    }
}
```

38

Figura 49 - MQTT Subscribe

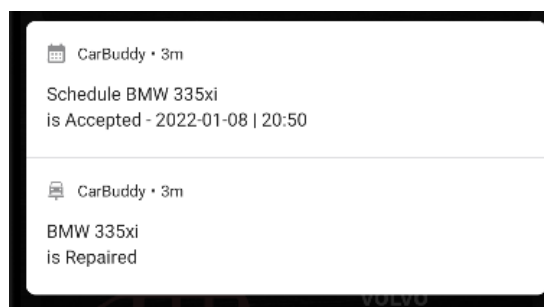


Figura 50 - Exemplo de Notificação

Todo este processo é realizado na aplicação móvel, e é feito tanto nas reparações como nos agendamentos.

Cofinanciado por:



## 4 CONCLUSÃO

Com a elaboração deste projeto, cujo tema seria a criação de uma API onde o objetivo é haver comunicação entre plataformas, conseguimos desenvolver a prática da programação orientada a objetos e a sistemas API REST.

A nível de dificuldades podemos referir, que inicialmente, não sabíamos por onde começar. Com o passar do tempo, e das aulas da Unidade Curricular de Serviços e Interoperabilidade de Sistemas, e após o desenvolvimento de algumas das fichas práticas, fomos ganhando prática no desenvolvimento de API's REST, sendo também bastante útil toda a documentação da YII Framework. Bem perto do fim do projeto, sentimos dificuldades também na implementação da tecnologia Mosquitto, pois não sabíamos bem onde a implementar.

Enquanto grupo, o trabalho foi sempre dividido e muito bem gerido entre nós, podendo assim afirmar que os três elementos do grupo trabalharam de igual modo. O grupo fica contente por conseguir concluir assim um projeto com esta dimensão pois, com isto, sentimos evolução pessoal e profissional.