

Índice

api_challenge	1
login	2
POST login	2
GET logout	2
hired_employees	3
GET listar_hired_employees	3
GET leer_hired_employees	3
POST registro_hired_employees	3
PUT update_hired_employees	3
DEL delete_hired_employees	3
departments	4
GET listar_departments	4
GET leer_departments	4
POST registro_departments	4
PUT update_departments	4
DEL delete_departments	4
jobs	5
GET listar_jobs	5
GET leer_jobs	5
POST registro_jobs	5
PUT update_jobs	5
DEL delete_jobs	5
avro_actions	6
GET list_avro	6
POST create_avro	6
POST read_avro	6
POST upload_avro	6
csv_actions	7
POST upload_hired_employees	7
POST upload_departments	7
POST upload_jobs	7

api_challenge

login

POST login

request that allows log in
<http://localhost:5000/login>

Bodyraw (json)

json

```
{  
  "username": "admin",  
  "password": "#####"  
}
```

GET logout

request that allows log out
<http://localhost:5000/logout>

hired_employees

GET listar_hired_employees

List all the hired employees

http://localhost:5000/api/hired_employees

GET leer_hired_employees

search for a specific hired employee by its id

http://localhost:5000/api/hired_employees/<id>

POST registro_hired_employees

Adds a new hired employee if all the parameters are correct

http://localhost:5000/api/hired_employees

Bodyraw (json)

json

```
{
  "id": 4535,
  "name": "Marcelo Gonzalez",
  "datetime": "2021-07-27T16:02:08Z",
  "department_id": 1,
  "job_id": 2
}
```

PUT update_hired_employees

Modifies an existing hired employee

http://localhost:5000/api/hired_employees

Bodyraw (json)

json

```
{
  "id": 4535,
  "name": "Marcelo Gonzalez",
  "datetime": "2021-07-27T16:02:08Z",
  "department_id": 1,
  "job_id": 2
}
```

DEL delete_hired_employees

deletes an employee by its id

http://localhost:5000/api/hired_employees/<id>

departments

GET listar_departments

List all the departments

<http://localhost:5000/api/departments>

GET leer_departments

search for a specific department by its id

<http://localhost:5000/api/departments/<id>>

POST registro_departments

Adds a new department if all the parameters are correct

<http://localhost:5000/api/departments>

Bodyraw (json)

json

```
{
  "id": 1,
  "department": "Supply Chain"
}
```

PUT update_departments

Modifies an existing department

<http://localhost:5000/api/departments>

Bodyraw (json)

json

```
{
  "id": 1,
  "department": "Supply Chain"
}
```

DEL delete_departments

deletes a department by its id

<http://localhost:5000/api/departments/<id>>

jobs

GET listar_jobs

List all the jobs

<http://localhost:5000/api/jobs>

GET leer_jobs

search for a specific job by its id

<http://localhost:5000/api/jobs/<id>>

POST registro_jobs

Adds a new job if all the parameters are correct

<http://localhost:5000/api/jobs>

Bodyraw (json)

json

```
{  
  "id": 1,  
  "job": "Recruiter"  
}
```

PUT update_jobs

Modifies an existing job

<http://localhost:5000/api/jobs>

Bodyraw (json)

json

```
{  
  "id": 1,  
  "job": "Recruiter"  
}
```

DEL delete_jobs

deletes a job by its id

<http://localhost:5000/api/jobs/<id>>

Add request description...

avro_actions

GET list_avro

List all the avros in the system

http://localhost:5000/api/list_avro

Bodyraw (json)

```
json
{
  "name_avro": "ejemplo2"
}
```

POST create_avro

create a backup avro file according to the table

http://localhost:5000/api/create_avro

Bodyraw (json)

```
json
{
  "name_avro": "ejem34",
  "name_table": "departments"
}
```

POST read_avro

Shows the data in an avro

http://localhost:5000/api/read_avro

Bodyraw (json)

```
json
{
  "name_avro": "hired_employees_ejem.avro"
}
```

POST upload_avro

restore a certain table using the avro backup

http://localhost:5000/api/upload_avro

Bodyraw (json)

```
json
{
  "name_avro": "departments_ejem.avro"
}
```

csv_actions

POST upload_hired_employees

copy the csv data to the database, using the csv title as data to select the table

`http://localhost:5000/api/csv`

Bodyform-data

file

`/C:/Users/danil/Documents/proyectos/hired_employees.csv`

POST upload_departments

copy the csv data to the database, using the csv title as data to select the table

`http://localhost:5000/api/csv`

Bodyform-data

file

`/C:/Users/danil/Documents/proyectos/departments.csv`

POST upload_jobs

copy the csv data to the database, using the csv title as data to select the table

`http://localhost:5000/api/csv`

Bodyform-data

file

`/C:/Users/danil/Documents/proyectos/jobs.csv`