

B.1 Manual de Usuario

La aplicación esta desplegada en Heroku para que el tribunal pueda hacer uso de esta sin tener que hacer un Set Up local.

Al tener un servidor gratuito en Heroku, los servicios se apagan tras unos minutos de inactividad, con lo que la primera llamada tardará más de lo normal dado que tiene que levantar dos servidores.

<https://ulpgc-course.herokuapp.com/>

usuarios: admin@tfg.com profesor@tfg.com alumno@tfg.com
contraseña: secret

B.1.1.1 Set Up Laravel (API)

Para la instalación de Laravel es necesario tener instalado los siguientes servicios:

- Servidor web Apache ≥ 2.4 o Nginx > 1.2 .
- Base de Datos MySQL ≥ 5.6 o PostgreSQL ≥ 9 .
- Lenguaje PHP $\geq 7.0.0$.
 - OpenSSL PHP Extension.
 - PDO PHP Extension.
 - Mbstring PHP Extension.
 - Tokenizer PHP Extension.
 - XML PHP Extension.
- Manejador de Dependencias **Composer** $\geq 1.2.0$.

También se puede hacer uso de software ya empaquetado y configurados como pueden ser:

- XAMPP
- WAMP
- MAMP

Además, tenemos que tener una cuenta de AWS S3 para la subida de ficheros en la nube. Después de tener estos servicios instalados, creamos una base de datos desde phpadmin o desde el terminal y ejecutamos lo siguiente:

1. `cd [ubicación_del_proyecto]`
2. `composer install`

Una vez instaladas todas las dependencias de Laravel procedemos a cambiar las variables de entorno ubicadas en el fichero `.env`, las variables que debemos actualizar son:

- `DB_CONNECTION=[mysql|pgsql]`
- `DB_HOST=127.0.0.1`
- `DB_PORT=XXXX`

- DB_DATABASE=[nombre_de_la_base_de_datos]
- DB_USERNAME=[nombre_de_usuario]
- DB_PASSWORD=XXX
- FRONT_END_END_POINT=XXX

Y para finalizar ejecutamos “php artisan migrate –seed” para insertar las tablas en la base de datos y poblarlas de datos.

B.1.1.2 Postman – Test API

Rutas de la API

Domain	Method	URL	Name	Action	Middleware
	GET HEAD	/		Closure	web
	POST	api/v1/auth/login		App\Http\Controllers\AuthController@login	api
	POST	api/v1/auth/logout		App\Http\Controllers\AuthController@logout	api, auth:api
	POST	api/v1/auth/me		App\Http\Controllers\AuthController@me	api, auth:api
	POST	api/v1/auth/refresh		App\Http\Controllers\AuthController@refresh	api, auth:api
	POST	api/v1/auth/register		App\Http\Controllers\AuthController@register	api
	POST	api/v1/cart		App\Http\Controllers\CartController@pay	api, auth:api
	POST	api/v1/courses		App\Http\Controllers\CourseController@store	api, auth:api, role:admin teacher
	GET HEAD	api/v1/courses		App\Http\Controllers\CourseController@index	api
	GET HEAD	api/v1/courses/myCourse		App\Http\Controllers\CourseController@myCourse	api, auth:api
	POST	api/v1/courses/photo/{course}		App\Http\Controllers\CourseController@photo	api, auth:api, role:admin teacher
	POST	api/v1/courses/section		App\Http\Controllers\CourseSectionController@store	api, auth:api, role:admin teacher
	POST	api/v1/courses/section/resource		App\Http\Controllers\CourseResourceController@store	api, auth:api, role:admin teacher
	DELETE	api/v1/courses/section/resource/{courseResource}		App\Http\Controllers\CourseResourceController@destroy	api, auth:api, role:admin teacher
	PUT	api/v1/courses/section/resource/{courseResource}		App\Http\Controllers\CourseResourceController@update	api, auth:api, role:admin teacher
	PUT	api/v1/courses/section/{courseSection}		App\Http\Controllers\CourseSectionController@update	api, auth:api, role:admin teacher
	DELETE	api/v1/courses/section/{courseSection}		App\Http\Controllers\CourseSectionController@destroy	api, auth:api, role:admin teacher
	GET HEAD	api/v1/courses/{course}		App\Http\Controllers\CourseController@show	api, auth:api
	PUT	api/v1/courses/{course}		App\Http\Controllers\CourseController@update	api, auth:api, role:admin teacher
	DELETE	api/v1/courses/{course}		App\Http\Controllers\CourseController@destroy	api, auth:api, role:admin teacher
	PUT	api/v1/courses/{course}/vote		App\Http\Controllers\CourseController@vote	api, auth:api
	POST	api/v1/password/email		App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail	api, guest
	POST	api/v1/password/reset		App\Http\Controllers\Auth\ResetPasswordController@reset	api, guest
	GET HEAD	api/v1/teachers		App\Http\Controllers\UserController@teachers	api
	GET HEAD	api/v1/users		App\Http\Controllers\UserController@index	api, auth:api, role:admin
	GET HEAD	api/v1/users/payment		App\Http\Controllers\UserController@payment	api, auth:api
	DELETE	api/v1/users/{user}		App\Http\Controllers\UserController@destroy	api, auth:api
	PUT	api/v1/users/{user}		App\Http\Controllers\UserController@update	api, auth:api
	GET HEAD	api/v1/users/{user}		App\Http\Controllers\UserController@show	api, auth:api, role:admin

Figura B.1.1.2: Rutas de la API

Registro

POST

{{url}}/api/v1/auth/register

Params

Send

Save

Authorization

Headers (2)

Body

Pre-request Script

Tests

Cookies

Code

form-data

x-www-form-urlencoded

raw

binary

JSON (application/json)

```

1 {
2   "name": "Postman",
3   "last_name": "Postman",
4   "email": "postman@gmail.com",
5   "password": "secret"
6 }

```

Body

Cookies (2)

Headers (10)

Test Results

Status: 201 Created

Time: 2180 ms

Size: 476 B

Pretty

Raw

Preview

JSON

Save Response

```

1 {
2   "user": {
3     "name": "Postman",
4     "last_name": "Postman",
5     "email": "postman@gmail.com",
6     "updated_at": "2018-07-13 17:20:24",
7     "created_at": "2018-07-13 17:20:24",
8     "id": 52
9   }
10 }

```

Figura B.1.1.2: Postman Registro

Login

POST `{{url}}/api/v1/auth/login` Params **Send** Save

Authorization Headers (1) **Body** Pre-request Script Tests Cookies Code

☒ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary

Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/> email	admin@tfg.com			
<input checked="" type="checkbox"/> password	secret			
New key	Text ▾ Value	Description		

Body Cookies (2) Headers (10) Test Results Status: 200 OK Time: 278 ms Size: 955 B

Pretty Raw Preview JSON ▾

```

1 {
2   "user": {
3     "id": 1,
4     "name": "Mario",
5     "last_name": "Oberbrunner",
6     "email": "admin@tfg.com",
7     "photo": null,
8     "created_at": "2018-07-03 23:50:26",
9     "updated_at": "2018-07-07 23:01:30",
10    "deleted_at": null,
11    "stripe_id": null,
12    "card_brand": null,
13    "card_last_four": null,
14    "trial_ends_at": null
15  },
16  "rol": "admin",
17  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
    .eyJpc3MiOiJodHRwOlwvXC90ZmcudGVzdFwvYXBPXC92MVwwYXV0aFwvbG9naW4iLCJpYXQoIjE1MDE0MTUzMzAzOTgyNSwibmJmIjoxtNTMxNDAzODI1LlCjZqdGkiOiJKUjgwdWVFWVd0VmVmaVJ2Iiwic3ViIjoxtLjcwcnYiOiI0NDUyYWYxZWY5ZmQxNTgxNmMzZWMSNzE1M2xNGUwYjA0bzU0NmFiIn0.LSENi1yWcJQPNCJ4yGR_WvKz1rMDjW0jHhKKhs8k8",

```

Figura B.1.1.2: Postman Login

Refresh Token

POST{{url}}/api/v1/auth/refreshParamsSendSave

AuthorizationHeaders (2)BodyPre-request ScriptTestsCookiesCode

Key	Value	Description	...	Bulk Edit	Presets
Accept	application/json				
Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ...				
New key	Value	Description			

BodyCookies (2)Headers (11)Test ResultsStatus: 200 OKTime: 156 msSize: 979 B

PrettyRawPreviewJSONSave Response

```
1 {
2   "user": {
3     "id": 1,
4     "name": "Mario",
5     "last_name": "Oberbrunner",
6     "email": "admin@tfg.com",
7     "photo": null,
8     "created_at": "2018-07-03 23:50:26",
9     "updated_at": "2018-07-07 23:01:30",
10    "deleted_at": null,
11    "stripe_id": null,
12    "card_brand": null,
13    "card_last_four": null,
14    "trial_ends_at": null
15  },
16  "rol": "admin",
17  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
    .eyJpc3MiOiJodHRwOlwvXC90ZmcudGVzdFwvYXBpXC92MVwvYXV0aFwvcmVmcVzaCisImldhdiG6MTUzMTUwMzgyNSwiZXhwIjoxNTYzMDQxMjY
    0LCJyYmYiOiJlMzE1MDUyNjQsImp0aSI6ImUxejFyRkQwc1FDeHh4dWIiLCJzdWIiOiJEsInBydiI6Ijg3ZTBhZjF1ZjlmZDE1ODUyZmRlYzI3MTU
    zYTE0ZTBiMDQ3NTQyWEIfQ.eoSpmoa_Ati4ouv0Art-BGKMu3mZU_XCf_1rb0bBBKY",
18  "expires_in": 31536000
19 }
```

Figura B.1.1.2: Postman Refresh Token

Logout

POST{{url}}/api/v1/auth/logoutParamsSendSave

AuthorizationHeaders (2)BodyPre-request ScriptTestsCookiesCode

Key	Value	Description	...	Bulk Edit	Presets
Accept	application/json				
Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ...				
New key	Value	Description			

BodyCookies (2)Headers (11)Test ResultsStatus: 200 OKTime: 97 msSize: 373 B

PrettyRawPreviewJSONSave Response

```
1 {
2   "message": "Successfully logged out"
3 }
```

Figura B.1.1.2: Postman Logout

Me

POST{{url)}/api/v1/auth/meParamsSendSave

AuthorizationHeaders (2)BodyPre-request ScriptTestsCookiesCode

	Key	Value	Description	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	Accept	application/json				
<input checked="" type="checkbox"/>	Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ...				
	New key	Value	Description			

BodyCookies (2)Headers (11)Test ResultsStatus: 200 OKTime: 102 msSize: 972 B

PrettyRawPreviewJSONSave Response

```
1 {
2   "user": {
3     "id": 1,
4     "name": "Mario",
5     "last_name": "Oberbrunner",
6     "email": "admin@tfg.com",
7     "photo": null,
8     "created_at": "2018-07-03 23:50:26",
9     "updated_at": "2018-07-03 23:01:30",
10    "deleted_at": null,
11    "stripe_id": null,
12    "card_brand": null,
13    "card_last_four": null,
14    "trial_ends_at": null
15  },
16  "rol": "admin",
```

Figura B.1.1.2: Postman Me

Usuarios

GET{{url)}/api/v1/usersParamsSendSave

AuthorizationHeaders (2)BodyPre-request ScriptTestsCookiesCode

	Key	Value	Description	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	Accept	application/json				
<input checked="" type="checkbox"/>	Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ...				
	New key	Value	Description			

BodyCookies (2)Headers (11)Test ResultsStatus: 200 OKTime: 161 msSize: 3.42 KB

PrettyRawPreviewJSONSave Response

```
1 {
2   "users": {
3     "current_page": 1,
4     "data": [
5       {
6         "id": 2,
7         "name": "Mrs. Etha Schmeler",
8         "last_name": "Vandervort",
9         "email": "turner10@hotmail.com",
10        "photo": null,
11        "created_at": "2018-07-03 23:50:29",
12        "updated_at": "2018-07-03 23:50:29",
13        "deleted_at": null,
14        "stripe_id": null,
15        "card_brand": null,
16        "card_last_four": null,
17        "trial_ends_at": null,
18        "rol": "teacher"
19      },
```

Figura B.1.1.2: Postman Usuarios

Para no subir una captura de todas las rutas adjunto el fichero de Postman donde están todas las llamadas a la API.

https://drive.google.com/open?id=1FvWXUWrCv_cHdJoI7G1fGUC4YOMXxtxx

B.1.2.1 Set Up React JS

1. Instalar [Node JS](#)
2. Abrir el terminal.
3. cd [ubicación_del_proyecto]
4. Modificamos el valor de REACT_APP_API ubicado en el archivo .env .
5. Ejecutar en la consola “npm i”
6. npm start

B.1.3 Enlaces al código fuente

<https://github.com/ckwanted/tfg-api>

<https://github.com/ckwanted/tfg-web>

B.1.4 Videos sobre el uso de la aplicación

Login

<https://drive.google.com/open?id=1R7emD3PTWcfogT7t1VOXmwL-09f740I1>

Registro

https://drive.google.com/open?id=1XFSyvbrn8jP3NsEDC_s0Qf5LYeHi3kqS

Comprar Curso (Alumno)

<https://drive.google.com/open?id=1LIc4k44DHtlhueFzJGQ43Axv5EitsMMb>

Editar Curso (Profesor o Administrador)

<https://drive.google.com/open?id=1muZVe67b4QMUKmg0Jogp2QDkWllabEIS>

Editar usuarios (Administrador)

<https://drive.google.com/open?id=1gQe9FPYpzpyBVPRWDTnzToLL95tVe4YH>