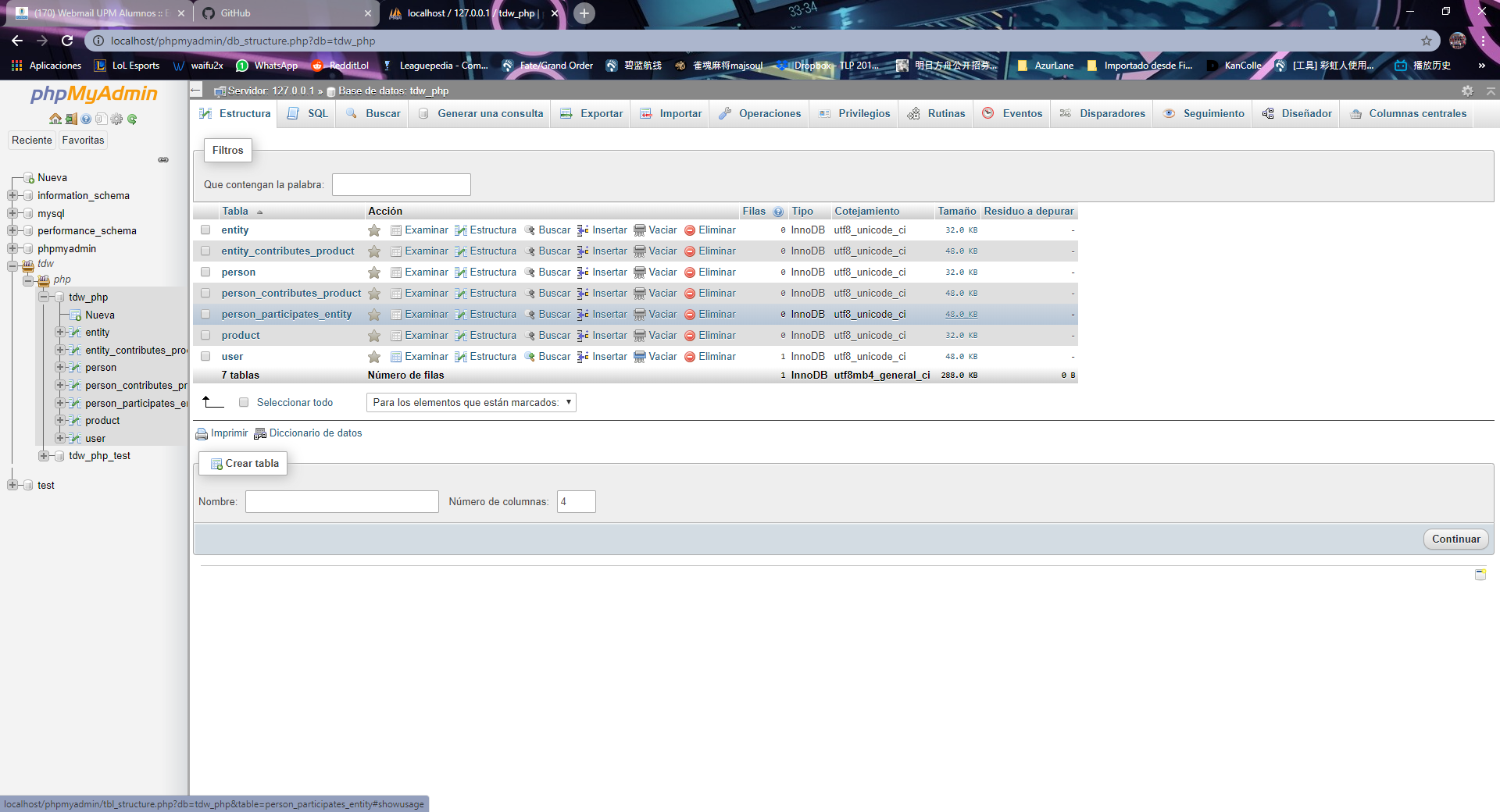
|  |
| --- |
|  |
| Practica PHP TDW |
|  |

|  |
| --- |
| Xin Chen Xu – bn0059 |

# Configuración del entorno de la práctica.

Como primer paso, se ha usado el comando de *composer install* para instalar los componentes necesarios y tras eso, se ejecuta el comando por segunda vez para asegurar que se han metido los datos en la bbdd creada antes en phpmyadmin.



Aparte de ello, en phpmyadmin se ha creado un usuario adminUser que coincide con el usuario administrador que esta tanto en env.local y phpunit.xml. Probamos el conjunto de tests del proyecto y vemos que falla en entityController, entityRelationsController, personController y PersonRelationsController.

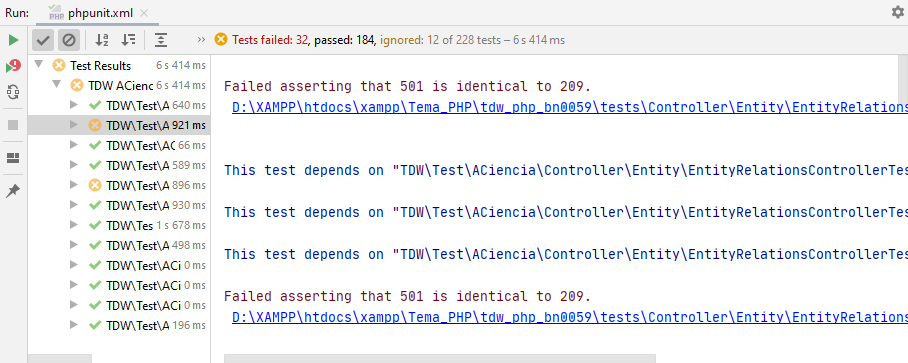
# EntityController

Se completa la parte de código de entityController, en concreto las funciones cget, get, getEntityname, delete, post y put tomando como ejemplo las de productController y se ejecuta una vez phpunit.xml para comprobar que esta correcto esa parte. Tras la comprobación se procede a hacer un push al repositorio github creado.



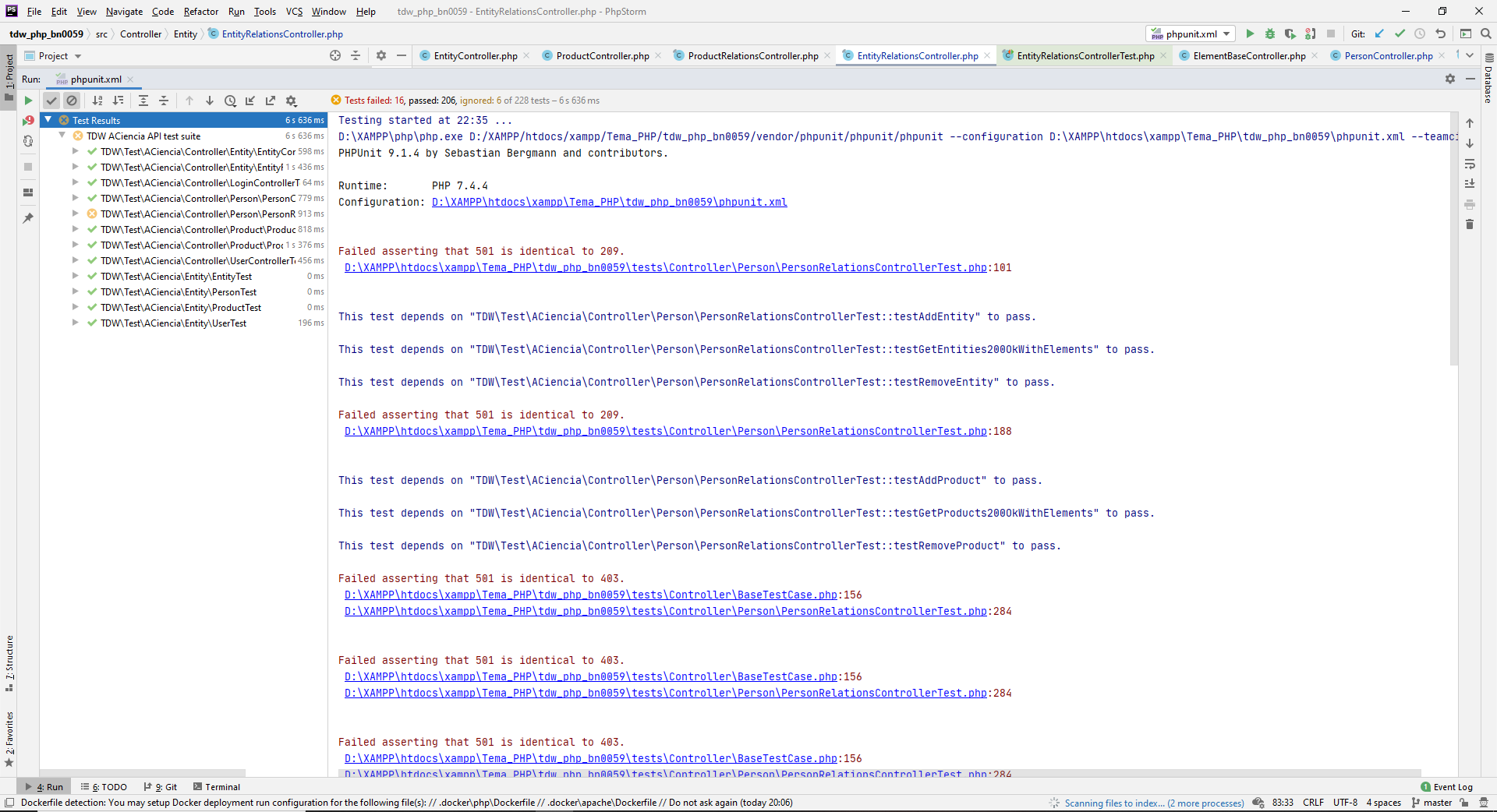
# PersonController

Se completa la parte de código de personController, en concreto las funciones cget, get, getEntityname, delete, post y put tomando como ejemplo las de productController y se ejecuta una vez phpunit.xml para comprobar que esta correcto esa parte. Tras la comprobación se procede a hacer un push al repositorio github creado.



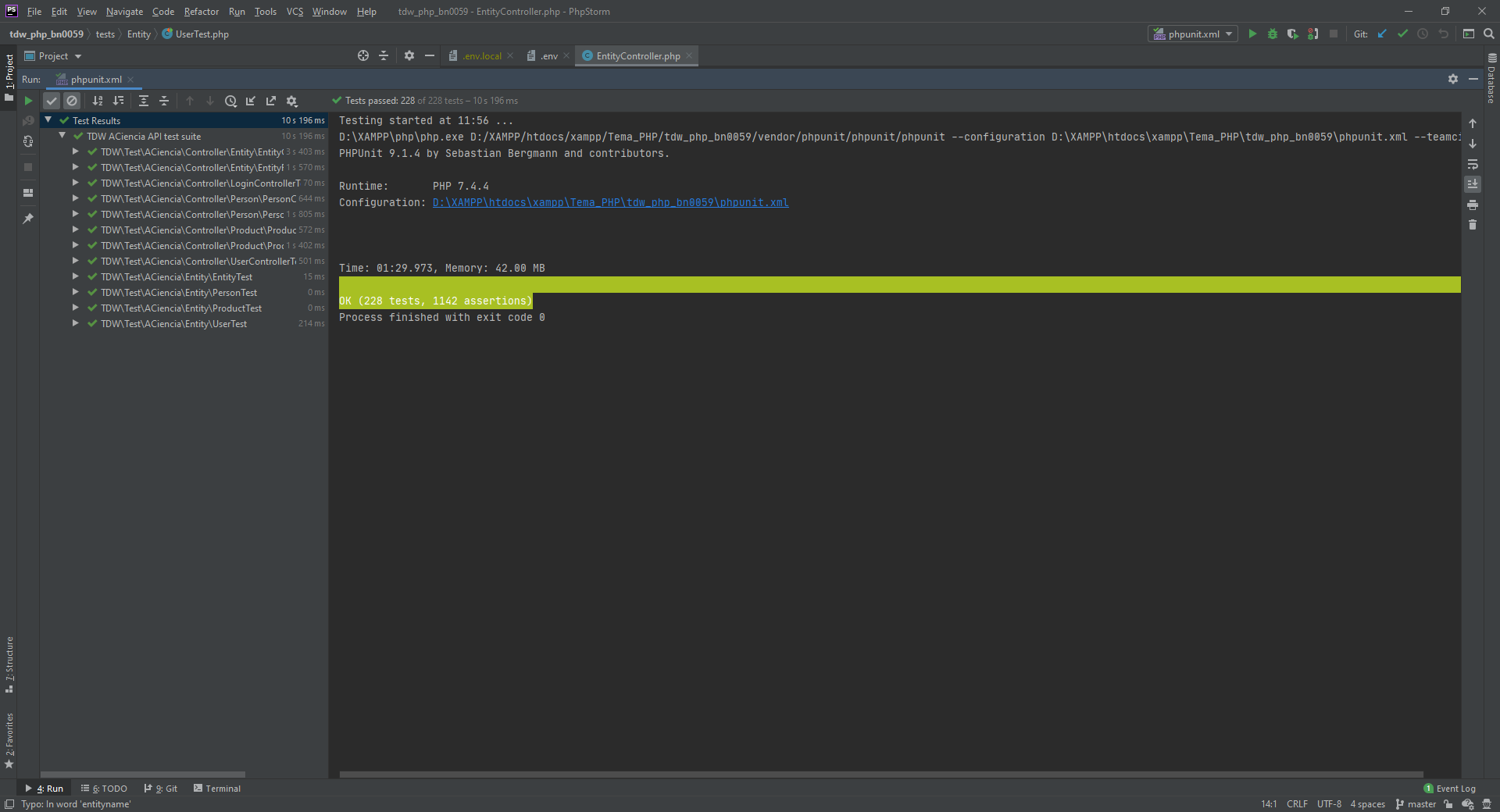
# EntityRelationsController

Se completa la parte de código de entityRelationsController, en concreto las funciones getPersons, operationPerson, getProducts y operationProduct tomando como ejemplo las de productRelationsController y se ejecuta una vez phpunit.xml para comprobar que esta correcto esa parte. Tras la comprobación se procede a hacer un push al repositorio github creado.



# PersonRelationsController

Se completa la parte de código de personRelationsController, en concreto las funciones getEntities, operationEntity, getProducts y operationProduct tomando como ejemplo las de productRelationsController y se ejecuta una vez phpunit.xml para comprobar que esta correcto esa parte y vemos que el proyecto ha pasado todos los tests.



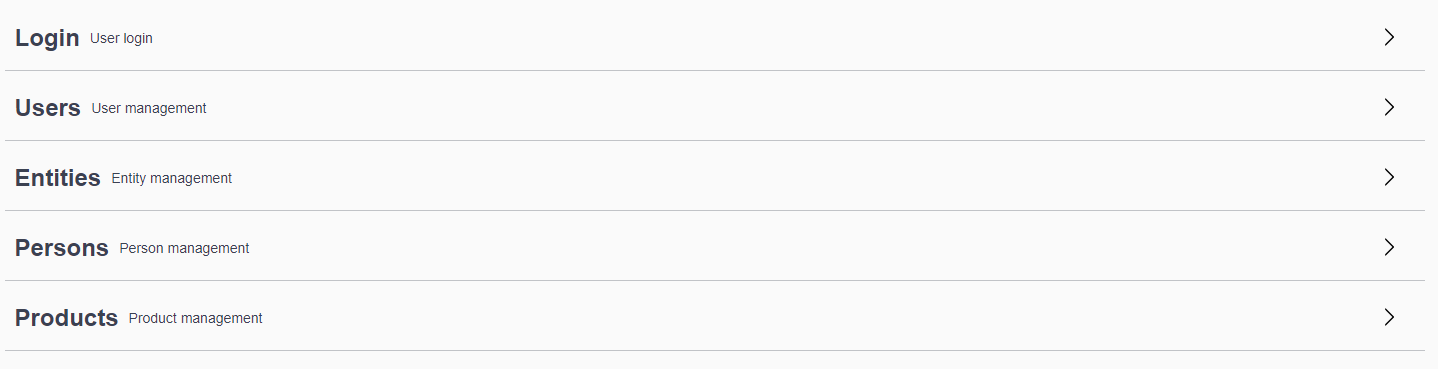
Tras ello, se procede a realizar un push al repositorio github y estaría completa la parte de los controladores.

# Modificación del api de la practica

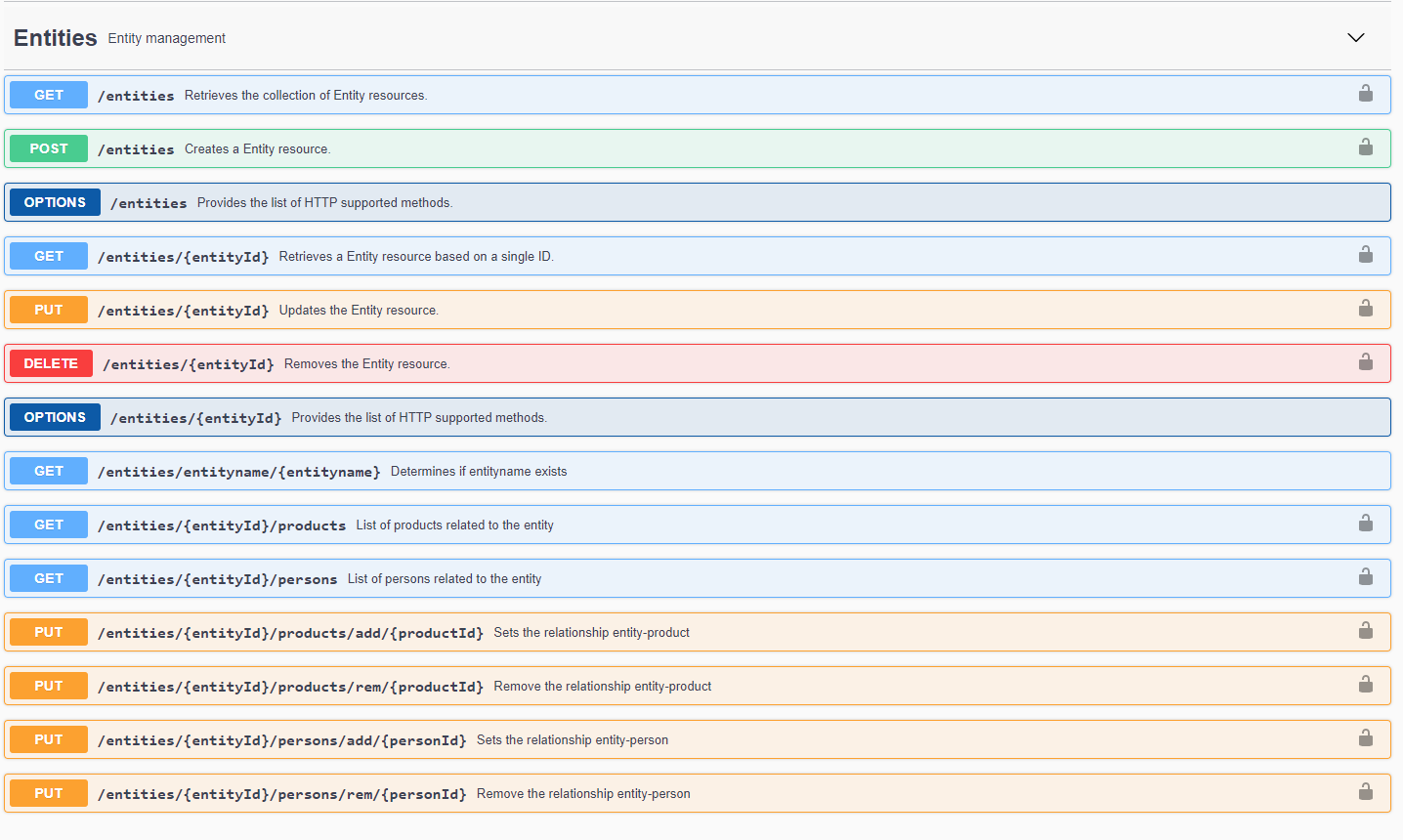
En el archivo openapi.yami, añadimos los tags correspondientes a Entities y Persons para que aparezcan al ejecutarlos y creamos todas las funciones necesarias para esas 2, en este caso también es necesario la creación de los parámetros entityname y productname que necesitan alguna de las funciones. Tras ello, se ejecuta la api y se procede a realizar la comprobación del correcto funcionamiento de todas las opciones que nos da la api.

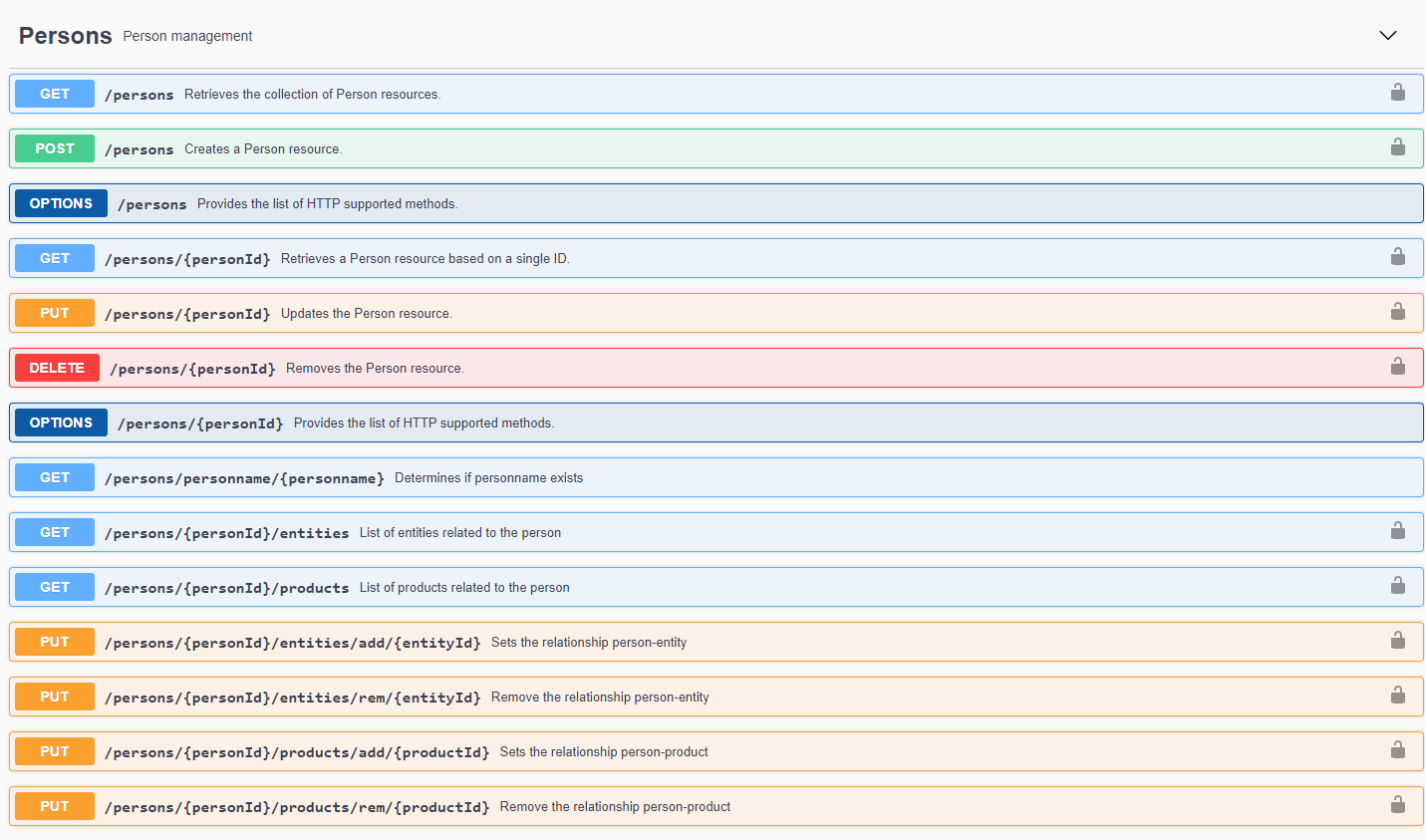
Se utiliza Usuario: adminUser y Password: \*adminUser\*

Api:



Entities:



Persons: 

Github: <https://github.com/XinChen97/tdw_php_bn0059>