**TEST CASES PETS STORE**

|  |  |
| --- | --- |
| Identificador | TC-P01 |
| Escenario | Agregar una mascota |
| Descripción | Como administrador de la tienda quiero agregar una nueva mascota para que se publique en la tienda y que funcione acorde a las especificaciones |
| Precondiciones | - Tener postman instalado  - API disponible   * - Tener mascotas con estado sold |
| Pasos | 1. Ingresar a postman 2. Ingresar metodo POST 3. ingresar POST /user/createWithList 4. body agregar esta info   {  "name": "PetTest",  "photoUrls": [  "in sed consequat",  "dolore aliqua elit in"  ],  "id": 11,  "category": {  "id": 89739517,  "name": "PetTest "  },  "tags": [  {  "id": -3869620,  "name": "PetTest"  },  {  "id": 64835211,  "name": "PetTest"  }  ],  "status": "sold"  }   1. enviar request |
| Resultado Esperado | - Al crear una nueva mascota debe arrojar status code 201 con la info de la mascota creada |
| Probador | Daniel |
| Estado | Exitoso |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P02 |
| Escenario | Creates list of users with given input array |
| Descripción | Creación de usuarios mediante la utilización del recurso  POST /user/createWithList |
| Precondiciones | * el API PetShop debe estar desplegado y accesible desde internet * tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso POST/ user/createWithList enviando información correcta en el body de la petición  [  {  "id": 1,  "username": "UserTest",  "firstName": "UserTest",  "lastName": "officia",  "email": "aliqua ex cupidatat",  "password": "pass",  "phone": "4",  "userStatus": 49682779  },  {  "id": 20253511,  "username": "IDTESTdos",  "firstName": "ea deserunt",  "lastName": "Lorem pariatur ut ut",  "email": "aliquip ullamco cupidatat nulla",  "password": "nostrud sunt ex reprehenderit proident",  "phone": "culpa",  "userStatus": -35819709  }  ] |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Exitoso |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** correct information is added in the database for the creation of users  **Given** {{baseUrl}}/user/createWithList  **And** Method POST  **When** the parameter [  {  "id": 1,  "username": "UserTest",  "firstName": "UserTest",  "lastName": "officia",  "email": "aliqua ex cupidatat",  "password": "pass",  "phone": "4",  "userStatus": 49682779  },  {  "id": 20253511,  "username": "IDTESTdos",  "firstName": "ea deserunt",  "lastName": "Lorem pariatur ut ut",  "email": "aliquip ullamco cupidatat nulla",  "password": "nostrud sunt ex reprehenderit proident",  "phone": "culpa",  "userStatus": -35819709  }  ] Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P03 |
| Escenario | Creates list of users with given input array |
| Descripción | Creación de usuarios mediante la utilización del recurso  POST /user/createWithArray |
| Precondiciones | * el API PetShop debe estar desplegado y accesible desde internet * tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso POST /user/createWithArray enviando información valida en el body de la petición  [  {  "id": -47152823,  "username": "TESTDOS",  "firstName": "UserTest",  "lastName": "cupidatat nisi Duis",  "email": "id esse ipsum mollit",  "password": "pari",  "phone": "ullamco dolor tempor",  "userStatus": -50665513  },  {  "id": -85528683,  "username": "elit e",  "firstName": "qui Lorem proident ullamco",  "lastName": "ad qui",  "email": "in elit proident",  "password": "Excepteur Ut",  "phone": "labore",  "userStatus": 5717613  }  ] |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Exitoso |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** correct information is added in the database for the creation of users  **Given** {{baseUrl}}/user/createWithList  **And** Method POST  **When** the parameter [  {  "id": -47152823,  "username": "TESTDOS",  "firstName": "UserTest",  "lastName": "cupidatat nisi Duis",  "email": "id esse ipsum mollit",  "password": "pari",  "phone": "ullamco dolor tempor",  "userStatus": -50665513  },  {  "id": -85528683,  "username": "elit e",  "firstName": "qui Lorem proident ullamco",  "lastName": "ad qui",  "email": "in elit proident",  "password": "Excepteur Ut",  "phone": "labore",  "userStatus": 5717613  }  ]  **Then** status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P04 |
| Escenario | Creación de lista de usuarios con matriz |
| Descripción | Creación de usuarios mediante la utilización del recurso  POST /user |
| Precondiciones | * el API PetShop debe estar desplegado y accesible desde internet * tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso POST /user enviando información valida en el body de la petición |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Exitoso |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Creates list of users with given input array  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P05 |
| Escenario | Registra al usuario en el sistema |
| Descripción | Creación de usuarios mediante la utilización del recurso  GET /user/login?username=IDTEST&password=pass |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso GET /user/login?username=IDTEST&password=pass enviando información correcta con el usuario y el password creados en los campos de Params |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Exitoso |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P06 |
| Escenario | Cierra la sesión actual del usuario conectado |
| Descripción | Cerrar Sesión mediante la utilización del recurso  GET /user/logout |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso GET /user/logout sin parametros |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Exitoso |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P07 |
| Escenario | Obtener usuario por nombre de usuario |
| Descripción | Buscar usuarios por nombre de usuario mediante la utilización del recurso  GET /user/:username |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso GET /user/:username con nombre de usuario creado y valido en Params |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Exitoso |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P08 |
| Escenario | Usuario actualizado |
| Descripción | Buscar usuarios por nombre de usuario mediante la utilización del recurso  PUT /user/:username |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso PUT /user/:username con información valida a actualizar |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Positiva |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P09 |
| Escenario | Borrar Usuario |
| Descripción | Borrar usuarios por nombre de usuario mediante la utilización del recurso  DELETE /user/:username |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso DELETE /user/:username con usuario valido a borrar |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Positiva |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P10 |
| Escenario | Agregar una nueva mascota a la tienda |
| Descripción | Agregar nueva mascota mediante la utilización del recurso  POST /pet |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso POST /pet con información valida en el body |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Exitosa |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P11 |
| Escenario | Actualizar una mascota existente |
| Descripción | Actualizar una mascota existente mediante la utilización del recurso  PUT /pet |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso PUT /pet con datos validos en el body |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Positiva |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P12 |
| Escenario | Encontrar mascotas por estado |
| Descripción | Encontrar mascotas por estado existente mediante la utilización del recurso  GET /pet/findByStatus?status=available&status=available |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso GET /pet/findByStatus?status=available&status=available con datos validos en el params |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Positivo |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P13 |
| Escenario | Encontrar mascotas por etiquetas |
| Descripción | Encontrar mascotas por etiquetas existente mediante la utilización del recurso  GET /pet/findByTags?tags=PetTest |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso GET /pet/findByStatus?status=available&status=available con datos validos en el params |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Positiva |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |

|  |  |
| --- | --- |
| Identificador | TC-P14 |
| Escenario | Encontrar mascotas por ID |
| Descripción | Encontrar mascotas por ID existente mediante la utilización del recurso  GET /pet/:petId |
| Precondiciones | * El API PetShop debe estar desplegado y accesible desde internet * Tener instalado un cliente REST como Postman |
| Pasos | Consumir el recurso GET /pet/:petId con ID valido en el params |
| Resultado Esperado | 200 ok |
| Probador | Daniel |
| Estado | Positivo |
| GHERKIN | **Background:**  \* url = https://petstore.swagger.io/v2  **Scenario:** Updating the informtion of a pet with data given  **Given** the path /pet/petID  **And** Method POST  **When** the parameter is <id>  Then status code 201. |