

Test Matrix

QA Course Project (Back End)

21/05/2022

Ligia Samara Diaz Hirashi

Version: 1.0

Created: 21/05/2022

Last Updated: 04/07/2022

Location: Monterrey, N.L., México.

Revision and Sign-off Sheet

Document History-

Version	Date	Author	Description of Change
1	20/05/2022	Samara Diaz Hirashi	Draft

Reference Documents-

Version	Date	Document Name
1.0	19/05/2022	QA Course Project (Back End)

Test Matrix	1
1. Test Cases	5
1.1. POST Pet Image (QPK-294)	5
1.1.1 Use Case POST Pet Image	5
1.1.2 Test Case POST Pet Image	5
1.2. POST Pet (QPK-295)	6
1.2.1. Use Case POST Pet	6
1.2.2. Test Case POST Pet	6
1.3. PUT Pet (QPK-296)	7
1.3.1. Use Case PUT Pet	7
1.3.2. Test Case PUT Pet	7
1.4. GET Pet by Status (QPK-330)	8
1.4.1. Use Case GET Pet by Status	8
1.4.2. Test Case GET Pet by Status	8
1.5. GET Pet by Id (QPK-331)	9
1.5.1. Use Case GET Pet By Id	9
1.5.2. Test Case GET Pet By Id	9
1.6. POST Pet by Id (QPK-233)	10
1.6.1. Use Case POST Pet by Id	10
1.6.2. Test Case POST Pet by Id	10
1.7. DELETE Pet by Id (QPK-350)	11
1.7.1. Use Case DELETE Pet by Id	11
1.7.2. Test Case DELETE Pet by Id	11
1.8. POST Order (QPK-351)	12
1.8.1. Use Case POST Order	12
1.8.2. Test Case POST Order	12
1.9. GET Order (QPK-352)	13
1.9.1. Use Case GET Order	13
1.9.2. Test Case GET Order	13
1.10. DELETE Order (QPK-353)	14
1.10.1. Use Case DELETE Order	14



1.10.2. Test Case DELETE Order	14
1.11. GET Inventory (QPK-354)	15
1.11.1. Use Case GET Inventory	15
1.11.2. Test Case GET Inventory	15
1.12. POST User by Array (QPK-355)	16
1.12.1. Use Case POST User by Array	16
1.12.2. Test Case POST User by Array	16
1.13. POST User by List (QPK-356)	17
1.13.1. Use Case POST User by List	17
1.13.2. Test Case POST User by List	17
1.14. GET User by Username (QPK-357)	18
1.14.1. Use Case GET User by Username	18
1.14.2. Test Case GET User by Username	18
1.15. PUT User (QPK-358)	19
1.15.1. Use Case PUT User	19
1.15.2. Test Case PUT User	19
1.16. DELETE User (QPK-368)	20
1.16.1. Use Case DELETE User	20
1.16.2. Test Case DELETE User	20
1.17. GET User Login (QPK-371)	21
1.17.1. Use Case GET User Login	21
1.17.2. Test Case GET User Login	21
1.18. GET User Logout (QPK-373)	22
1.18.1. Use Case GET User Logout	22
1.18.2. Test Case GET User Logout	22
1.19. POST User (QPK-376)	23
1.19.1. Use Case POST User	23
1.19.2. Test Case POST User	23
2. Test Matrix	24

1. Test Cases

1.1. POST Pet Image (QPK-294)

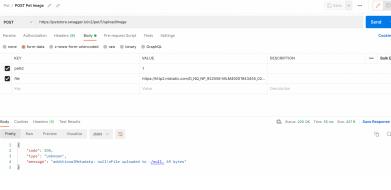
1.1.1 Use Case | POST Pet Image

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that uploads a pet image.”

1.1.2 Test Case | POST Pet Image

Description:

“Uploads an Image.”

Summary	Steps	Expected Result	Actual Result	Status
The request should upload a pet image into the server.	1.- Create a new HTTP POST Request from https://petstore.swagger.io 2.- Type the POST request “/pet/{petId}/uploadImage” along with the required parameters (id). 3.- Verify that the response uploads an image into Pet.	The HTTP POST Request should upload an image into Pets and return a “200” code response.	The request uploads an image into Pets and returns code “200”. 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/oxJKM

1.2. POST Pet (QPK-295)

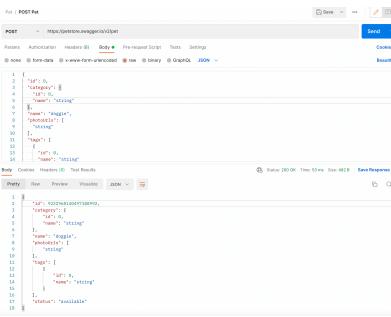
1.2.1. Use Case | POST Pet

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that adds a new pet to the store.”

1.2.2. Test Case | POST Pet

Description:

“Adds a new pet to the store.”

Summary	Steps	Expected Result	Actual Result	Status
The request should add a new pet into the store.	<ol style="list-style-type: none"> 1.- Create a new HTTP POST Request from https://petstore.swagger.io 2.- Type the POST request “/pet” along with the required parameters (body). 3.- Verify that the response adds a new pet. 	The HTTP POST Request should add a new pet to the store and return a “200” code response.	<p>The request adds a new pet into the store and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/qtuvC

1.3. PUT Pet (QPK-296)

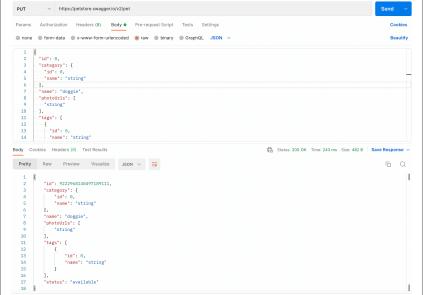
1.3.1. Use Case | PUT Pet

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that updates an existing pet.”

1.3.2. Test Case | PUT Pet

Description:

“Updates an existing Pet.”

Summary	Steps	Expected Result	Actual Result	Status
The request should update an existing pet.	1.- Create a new HTTP PUT Request from https://petstore.swagger.io 2.- Type the PUT request “/pet” along with the required parameters (body). 3.- Verify that the response updates an existing pet.	The HTTP PUT Request should update an existing pet and return a “200” code response.	The request updates an existing pet and returns code “200”. 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/guvzN

1.4. GET Pet by Status (QPK-330)

1.4.1. Use Case | GET Pet by Status

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that finds a pet by status.”

1.4.2. Test Case | GET Pet by Status

Description:

“Finds Pets by status.”

Summary	Steps	Expected Result	Actual Result	Status
The request should find a pet by its status.	<p>1.- Create a new HTTP GET Request from https://petstore.swagger.io</p> <p>2.- Type the GET request “/pet/findByStatus” along with the required parameters (status).</p> <p>3.- Verify that the response finds a pet when given the status.</p>	The HTTP GET Request should find a pet by its status and return a “200” code response.	<p>The request displays the pets by status and returns code “200”.</p> 	Pass ✓

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/rsvwL

1.5. GET Pet by Id (QPK-331)

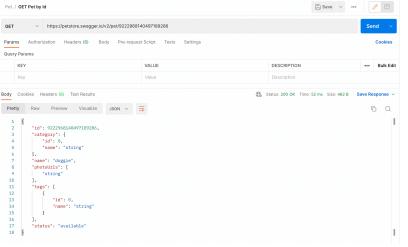
1.5.1. Use Case | GET Pet By Id

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that finds a pet by Id.”

1.5.2. Test Case | GET Pet By Id

Description:

“Finds pet by ID.”

Summary	Steps	Expected Result	Actual Result	Status
The request should find a pet by its Id.	<p>1.- Create a new HTTP GET Request from https://petstore.swagger.io</p> <p>2.- Type the GET request “/pet/{petId}” along with the required parameters (petId).</p> <p>3.- Verify that the response finds a pet when given the Id.</p>	The HTTP GET Request should find a pet by its Id and return a “200” code response.	<p>The request finds a pet by its Id and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/svzIM

1.6. POST Pet by Id (QPK-233)

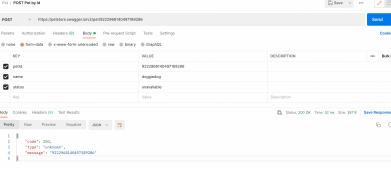
1.6.1. Use Case | POST Pet by Id

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that updates a pet in the store from a form data.”

1.6.2. Test Case | POST Pet by Id

Description:

“Updates a pet in the store with form data.”

Summary	Steps	Expected Result	Actual Result	Status
The request should update a pet in the store from data introduced into a form.	<p>1.- Create a new HTTP POST Request from https://petstore.swagger.io</p> <p>2.- Type the POST request “/pet/{petId}” along with the required parameters (petId, name and status).</p> <p>3.- Verify that the response updates a pet given the form data.</p>	The HTTP GET Request should update a pet in the store given the form data and return a “200” code response.	<p>The request updates a pet according to the input data and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/oMO78

1.7. DELETE Pet by Id (QPK-350)

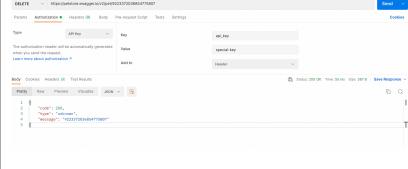
1.7.1. Use Case | DELETE Pet by Id

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that deletes a pet.”

1.7.2. Test Case | DELETE Pet by Id

Description:

“Deletes a pet.”

Summary	Steps	Expected Result	Actual Result	Status
The request should delete a pet.	<p>1.- Create a new HTTP DELETE Request from https://petstore.swagger.io</p> <p>2.- Type the DELETE request “/pet/{petId}” along with the required parameters (api_key and petId).</p> <p>3.- Verify that the response deletes a pet.</p>	The HTTP DELETE Request should delete a pet and return a “200” code response.	<p>The request deletes a pet given an Id and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/coCER

1.8. POST Order (QPK-351)

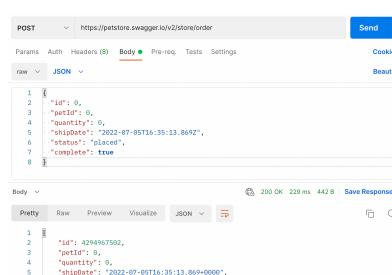
1.8.1. Use Case | POST Order

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server places an order for a pet.”

1.8.2. Test Case | POST Order

Description:

“Places an order for a pet.”

Summary	Steps	Expected Result	Actual Result	Status
The request should place an order for a pet.	<p>1.- Create a new HTTP POST Request from https://petstore.swagger.io</p> <p>2.- Type the POST request “/store/order” along with the required parameters (body).</p> <p>3.- Verify that the response places an order.</p>	The HTTP POST Request should place an order for a pet and return a “200” code response.	<p>The request places a new order and returns code “200”.</p> 	Pass ✓

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/bsE26

1.9. GET Order (QPK-352)

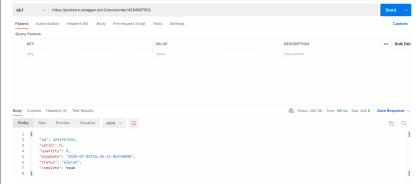
1.9.1. Use Case | GET Order

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that finds a purchased order by Id.”

1.9.2. Test Case | GET Order

Description:

“Finds purchased order by Id.”

Summary	Steps	Expected Result	Actual Result	Status
The request should find an order from its Id.	<ol style="list-style-type: none"> - Create a new HTTP GET Request from https://petstore.swagger.io - Type the GET request “/store/order/{orderId}” along with the required parameters (orderId). - Verify that the response finds the purchased order by its Id. 	The HTTP GET Request should find an order from its Id and return a “200” code response.	<p>The request finds an order from an Id and returns code “200”.</p>  <pre> { "id": 1, "qty": 1, "status": "placed", "description": "test" } </pre>	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/eryOY

1.10. DELETE Order (QPK-353)

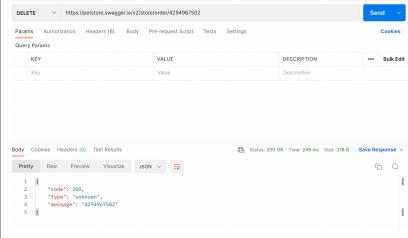
1.10.1. Use Case | DELETE Order

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that deletes an order by Id.”

1.10.2. Test Case | DELETE Order

Description:

“Deletes purchased order by Id.”

Summary	Steps	Expected Result	Actual Result	Status
The request should delete an order from its Id.	<p>1.- Create a new HTTP DELETE Request from https://petstore.swagger.io</p> <p>2.- Type the DELETE request “/store/order/{orderId}” along with the required parameters (orderId).</p> <p>3.- Verify that the response deletes the purchased order by its Id.</p>	The HTTP DELETE Request should delete an order from its Id and return a “200” code response.	<p>The request deletes an order given an Id and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/hoqyT

1.11. GET Inventory (QPK-354)

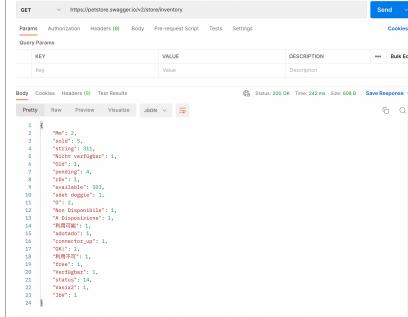
1.11.1. Use Case | GET Inventory

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that returns pet inventories by status.”

1.11.2. Test Case | GET Inventory

Description:

“Returns pet inventories by status.”

Summary	Steps	Expected Result	Actual Result	Status
The request should return pet inventories by status.	<ol style="list-style-type: none"> - Create a new HTTP GET Request from https://petstore.swagger.io - Type the GET request “/store/inventory”. - Verify that the response returns pet inventories by status. 	The HTTP GET Request should return pet inventories by status and return a “200” code response.	<p>The request returns the pet inventory but not by status and returns code “200”.</p>  <pre> [{"id": 1, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "IN_STOCK", "available": 100}, {"id": 2, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "IN_STOCK", "available": 100}, {"id": 3, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "IN_STOCK", "available": 100}, {"id": 4, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "OUT_OF_STOCK", "available": 0}, {"id": 5, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "OUT_OF_STOCK", "available": 0}, {"id": 6, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "OUT_OF_STOCK", "available": 0}, {"id": 7, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "IN_STOCK", "available": 100}, {"id": 8, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "IN_STOCK", "available": 100}, {"id": 9, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "IN_STOCK", "available": 100}, {"id": 10, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "IN_STOCK", "available": 100}, {"id": 11, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "IN_STOCK", "available": 100}, {"id": 12, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "IN_STOCK", "available": 100}, {"id": 13, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "IN_STOCK", "available": 100}, {"id": 14, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "IN_STOCK", "available": 100}, {"id": 15, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "IN_STOCK", "available": 100}, {"id": 16, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "IN_STOCK", "available": 100}, {"id": 17, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "IN_STOCK", "available": 100}, {"id": 18, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "IN_STOCK", "available": 100}, {"id": 19, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "IN_STOCK", "available": 100}, {"id": 20, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "IN_STOCK", "available": 100}, {"id": 21, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "IN_STOCK", "available": 100}, {"id": 22, "name": "dog", "category": {"id": 1, "name": "Cat"}, "status": "IN_STOCK", "available": 100}, {"id": 23, "name": "cat", "category": {"id": 2, "name": "Dog"}, "status": "IN_STOCK", "available": 100}, {"id": 24, "name": "fish", "category": {"id": 3, "name": "Fish"}, "status": "IN_STOCK", "available": 100}] </pre>	Fail 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/glqAN

1.12. POST User by Array (QPK-355)

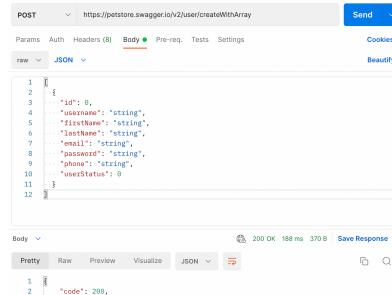
1.12.1. Use Case | POST User by Array

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that creates a list of users given an array.”

1.12.2. Test Case | POST User by Array

Description:

“Creates a list of users with given input array.”

Summary	Steps	Expected Result	Actual Result	Status
The request should create a list of users with a given array.	<ol style="list-style-type: none"> 1.- Create a new HTTP POST Request from https://petstore.swagger.io 2.- Type the POST request “/user/createWithArray” with the required parameters (body). 3.- Verify that the response creates a list of users with the given input array. 	The HTTP POST Request should create a list of users with the given input array and return a “200” code response.	<p>The request creates a list of users with the given array and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/mlS15

1.13. POST User by List (QPK-356)

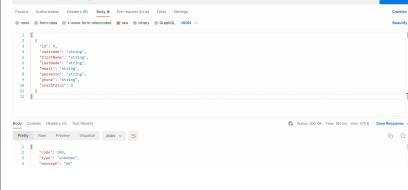
1.13.1. Use Case | POST User by List

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that creates a list of users given an array.”

1.13.2. Test Case | POST User by List

Description:

“Creates list of users with given input array.”

Summary	Steps	Expected Result	Actual Result	Status
The request should create a list of users with a given array.	<ol style="list-style-type: none"> 1.- Create a new HTTP POST Request from https://petstore.swagger.io 2.- Type the POST request “/user/createWithList” with the required parameters (body). 3.- Verify that the response creates a list of users with the given input array. 	The HTTP POST Request should create a list of users with the given input array and return a “200” code response.	<p>The request creates a list of users with the given array and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/ehnyH

1.14. GET User by Username (QPK-357)

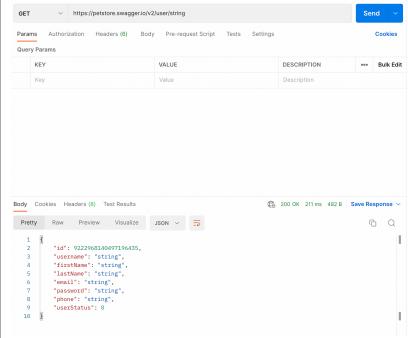
1.14.1. Use Case | GET User by Username

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that returns a user given the username.”

1.14.2. Test Case | GET User by Username

Description:

“Gets user by username.”

Summary	Steps	Expected Result	Actual Result	Status
The request should return a user given the username.	<p>1.- Create a new HTTP GET Request from https://petstore.swagger.io</p> <p>2.- Type the GET request “/user/{username}” with the required parameters (username).</p> <p>3.- Verify that the response returns a user given the username</p>	The HTTP GET Request should return a user given certain username and return a “200” code response.	<p>The request returns a user given the username and returns code “200”.</p>  <pre> 1 { 2 "id": 0, 3 "username": "string", 4 "firstName": "string", 5 "lastName": "string", 6 "email": "string", 7 "password": "string", 8 "phone": "string", 9 "userStatus": 0 10 </pre>	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/hilRV

1.15. PUT User (QPK-358)

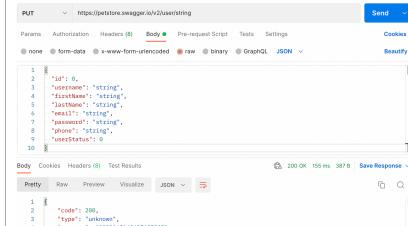
1.15.1. Use Case | PUT User

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that updates a user.”

1.15.2. Test Case | PUT User

Description:

“Update user.”

Summary	Steps	Expected Result	Actual Result	Status
The request should update a user.	<p>1.- Create a new HTTP PUT Request from https://petstore.swagger.io</p> <p>2.- Type the PUT request “/user/{username}” with the required parameters (username).</p> <p>3.- Verify that the response returns a user given the username</p>	The HTTP PUT Request should update a user given certain username and return a “200” code response.	<p>The request updates a user given a username and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/jxE01

1.16. DELETE User (QPK-368)

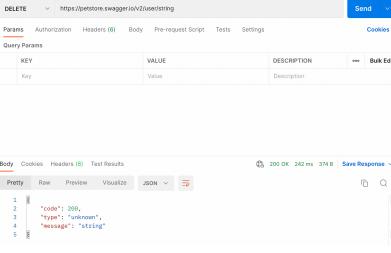
1.16.1. Use Case | DELETE User

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server that deletes a user.”

1.16.2. Test Case | DELETE User

Description:

“Delete user.”

Summary	Steps	Expected Result	Actual Result	Status
The request should delete a user.	<p>1.- Create a new HTTP DELETE Request from https://petstore.swagger.io</p> <p>2.- Type the DELETE request “/user/{username}” with the required parameters (username).</p> <p>3.- Verify that the response deletes a user given the username</p>	The HTTP DELETE Request should delete a user given certain username and return a “200” code response.		Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/atxyz

1.17. GET User Login (QPK-371)

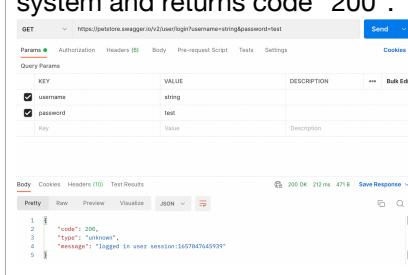
1.17.1. Use Case | GET User Login

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server logs a user into the system.”

1.17.2. Test Case | GET User Login

Description:

“Logs user into the system.”

Summary	Steps	Expected Result	Actual Result	Status
The request should log a user into the system.	<ol style="list-style-type: none"> 1.- Create a new HTTP GET Request from https://petstore.swagger.io 2.- Type the GET request “/user/login” with the required parameters (username and password). 3.- Verify that the response logs the user in. 	The HTTP GET Request should log a user into the system given a username and password and return a “200” code response.		Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/NS127

1.18. GET User Logout (QPK-373)

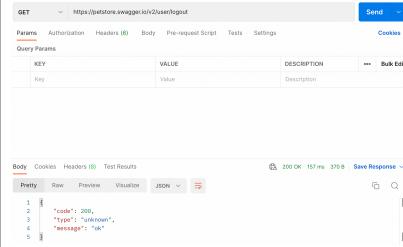
1.18.1. Use Case | GET User Logout

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server logs a user out of the system.”

1.18.2. Test Case | GET User Logout

Description:

“Logs out current logged in user session.”

Summary	Steps	Expected Result	Actual Result	Status
The request should log out current logged in user.	<ol style="list-style-type: none"> 1.- Create a new HTTP GET Request from https://petstore.swagger.io 2.- Type the GET request “/user/logout”. 3.- Verify that the response logs out the current logged in user. 	The HTTP GET Request should log out current logged in user and return a “200” code response.	<p>The request logs out the current logged in user and returns code “200”.</p> 	Pass 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/FHKYZ

1.19. POST User (QPK-376)

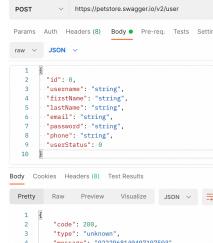
1.19.1. Use Case | POST User

“As a tester I want to make sure that I can design and implement a test on the Swagger PetStore Server creates a user.”

1.19.2. Test Case | POST User

Description:

“Creates user. This can only be done by the logged in user.”

Summary	Steps	Expected Result	Actual Result	Status
The request should create a user.	1.- Create a new HTTP POST Request from https://petstore.swagger.io 2.- Type the POST request “/user”. 3.- Verify that the response creates a user.	The HTTP POST Request should create a new user and return a “200” code response.	The request creates a user even when the user is not logged in and returns code “200”. 	Fail 

Software version: 2.0

Browser: Firefox version 100.0.1.

Jira: shorturl.at/bdMOQ

2. Test Matrix

Test Case	Mozilla Firefox 100.0.1	Google Chrome 103.0.5060.53
POST Pet Image		
POST Pet		
PUT Pet		
GET Pet by Status		
GET Pet by Id		
POST Pet by Id		
DELETE Pet by Id		
POST Order		
GET Order		
DELETE Order		
GET Inventory		
POST User by Array		
POST User by List		
GET User by Username		
PUT User		
DELETE User		
GET User Login		
GET User Logout		
POST User		

- **Failed:** 10.1%
- **Passed:** 89.9%