Test Cases

API tests for PetStore   5

C1: Add new pet to the store by POST method

Type	Priority	Estimate	References
Functional	Critical	None	None

Automation Type

None

Preconditions

Headers request:
Content-Type: application/json

Steps

1	Send POST request:https://petstore.swagger.io/v2/pet
	
	The body of the request is JSON object:
	{
	 "id": 0,
	 "category": {
	  "id": 0,
	  "name": "cat"
	 },
	 "name": "candy",
	 "photoUrls": [
	  "string"
	],
	 "tags": [
	  {
	   "id": 0,
	   "name": "string"
	  }
	],
	 "status": "available"
	}
		The request is successfully sent to the server

2	Check Status code	HTTP Status: 200 OK

3	Check the body of the server response	The response body will be in JSON format. Each value is conformed to the value from POST request:
		{
		 "id": 9223372036854175179,
		 "category": {
		  "id": 0,
		  "name": "cat"
		 },
		 "name": "candy",
		 "photoUrls": [
		  "string"
		],
		 "tags": [
		  {
		   "id": 0,
		   "name": "string"

5

```
            }
          ],
          "status": "available"
        }
```

4	Check the structure of the response	JSON schema is displayed correctly:
		The fields name and type are conformed to the requirements.
		The fields values match to the values sent with POST request

5	Check the response headers	

C2: Add new pet to the store with invalid method(e.g. GET)

Type	Priority	Estimate	References
Functional	Medium	None	None

Automation Type
None

Preconditions

Headers request:
Content-Type: application/json

Steps

1	Send GET request:https://petstore.swagger.io/v2/pet	The request is successfully sent to the server

The body of the request is JSON object:

```
{
"id": 0,
"category": {
  "id": 0,
  "name": "cat"
},
"name": "candy",
"photoUrls": [
  "string"
],
"tags": [
  {
    "id": 0,
```

5

```
      "name": "string"

    }

  ],

  "status": "available"

}
```

2	Check the status code	HTTP Status: 405 Method Not Allowed

3	Check the body of the server response	The response body in JSON format:

```
{
  apiResponse: {
    "type": "unknown"
  }
}
```

the description of the error is matched to the error type

4	Check the response headers	

C3: Add new pet to the store by POST method with JSON body, but XML content type

Type	Priority	Estimate	References
Functional	Medium	None	None

Automation Type
None

Preconditions

Headers request:
Content-Type: application/xml

Steps

1	Send POST request:https://petstore.swagger.io/v2/pet	The request is successfully sent to the server

The body of the request is JSON object:

```
{

"id": 0,

"category": {

  "id": 0,

  "name": "cat"

},

"name": "candy",

"photoUrls": [
```

5

```
    "string"

  ],

  "tags": [

    {

      "id": 0,

      "name": "string"

    }

  ],

  "status": "available"

}
```

2	Check Status code	HTTP Status: 415 Unsupported media type

| 3 | Check the body of the server response | The body of the response is in JSON format:
{
  "code": 415,
  "type": "unknown",
  "message": "javax.xml.bind.UnmarshalException\n - with linked exception:\n[org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Content is not allowed in prolog.]"
}

The description of the error match to the error type |

| 4 | Check the response headers | |

C4: Add new pet to the store by POST method, but a non-existent address

Type	Priority	Estimate	References
Functional	Medium	None	None

Automation Type
None

Preconditions

Headers request:
Content-Type: application/json

Steps

1	Send POST request:https://petstore.swagger.io/v2/pet1 The body of the request is JSON object: {	The request is successfully sent to the server

5

```
    "id": 0,

    "category": {

    "id": 0,

    "name": "cat"

    },

    "name": "candy",

    "photoUrls": [

    "string"

    ],

    "tags": [

    {

       "id": 0,

       "name": "string"

    }

    ],

    "status": "available"

    }
```

2	Check Status code	HTTP Status: 404 Not Found

3	Check the body of the server response	The response body in JSON format:
		`{`
		
		`"apiResponse": {`
		
		`    "message": "null for uri:`
		`http://petstore.swagger.io/v2/pet1",`
		
		`    "type": "unknown"`
		
		`}`
		
		`}`
		the description of the error is matched to the error type

4	Check the response headers	

C5: Add new pet to the store by POST method, but empty body(JSON format)   5

Type	Priority	Estimate	References
Functional	Critical	None	None

Automation Type

None

Preconditions

Headers request:
Content-Type: application/json

Steps

1	Send GET request:https://petstore.swagger.io/v2/pet
	
	The body of the request is JSON object:
	{
	
	}
		The request is successfully sent to the server

2	Check the status code	HTTP Status: 200 OK

3	Check the body of the server response	The response body in JSON format:
		{
		  "id": "This field is required.",
		  "name": "This field is required."
		}

4	Check the response headers	

5