



Search or jump to...

[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)



Learn Git and GitHub without any code!

Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

[Read the guide](#)

[gaurang053](#) / [petStore](#) Private

[Unwatch](#) 1 [Star](#) 0 [Fork](#) 0

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)

[master](#) 1 branch 0 tags

[Go to file](#) [Add file](#) [Code](#)

gaurang.vadher Update README.md	19ab0df 13 seconds ago	🕒 47 commits
.settings	Removed src main resources test folder	3 days ago
ExecutionReport	TestReport Added	7 hours ago
src/test	Updated README and Rename method name	3 hours ago
.classpath	Removed src main resources test folder	3 days ago
.gitattributes	updated cucumber-serenity-restassured skeleton	3 days ago
.gitignore	removed target folder	3 days ago
.project	updated cucumber-serenity-restassured skeleton	3 days ago
ProjectStructure.png	Added Project Structure Screenshot	6 minutes ago
README.md	Update README.md	13 seconds ago
Read_Me.pdf	ReadMe PDF version	2 minutes ago
TestReport.png	Updated README file	4 hours ago
execute.bat	Updated ReadMe file	4 hours ago
pom.xml	Features completed	2 days ago
rest-assured-logo-green.png	Update ReadMe file	4 hours ago
serenity.properties	updated cucumber-serenity-restassured skeleton	3 days ago

About

serenity-restassured-cucumber

[Readme](#)

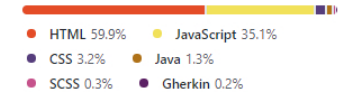
Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Languages



README.md



REST-assured

Applicatin : PetStore

Pet store is Serenity BDD framework based REST API Automation project. Imagine you are a part of the team that performs quality assurance for a Pet store application, The frontend design is under development but API has already been published. QA collaborate with developer team to make the feature more robust, by targeting to write tests for some workflows that they might break while developing business logic.

Goal is to cover:

- * To create a test automation framework skeleton
- * To test the application using available endpoint

- Reference: Swagger URL# <https://petstore.swagger.io/#/pet/updatePet>

Testing REST services in Java Technology used - Serenity, REST Assured, Cucumber and JAVA

Prerequisite

- JAVA 8 or higher
- As Integrated Development Environment - Used Eclipse Version: 2020-12 (4.18.0)
- A Build Tool - Used Maven

Getting Started

- Clone this repository - `gh repo clone gaurang053/petStore`

How to install and run this project?

- Double click on execute.bat file present at root or
- or Execute maven command - `mvn clean verify`

Story Description

Below list of story covers API capabilities of Login, UserProfile, Store and User. The story design such a way it covers all APIs present in the swagger.

- User - Operations about users

```
1_ As a End User, I can login and logout to the application using API
Description: Valid and Invalid Scenario covered
API Scenario:
* GET /user/login - Logs user into the system
* GET /user/logout - Logs out current logged in user session
```

```
2_ As a Admin User, I can create new user, view or update existing user information and delete the user
Description: Create new user, View user information, Update or Delete Existing user
API Scenario:
* POST /user Create user
* GET /user/{username} Get user by username
* PUT /user/{username} Updated user
* DELETE /user/{username} Delete User
* POST /user/createWithArray Creates list of users with given input array
* POST /user/createWithList Creates list of users with given input array
```

- Store - Access to Petstore order

```
3_ As a End User, I can control Pet Order Cart
```

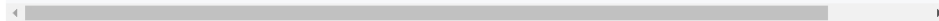
```
Descriptoin : Place new order, Find or Delete purchase order by ID
* POST /store/order Place an order for a pet
* GET /store/order/{orderId} Find purchase order by ID
* DELETE /store/order/{orderId} Delete purchahse order by ID
```

```
4_ As a Store Owner, I would like to check my inventory
Descriptoin : Check Inventory
* GET /store/inventory Return pet inventories by status
```

- Pet - Everything about Pets

```
5_ As a Store Owner, I would like add new pet profile, upload pet image and delete pet profile
Description : Create, Upload image, Search or Delete Pet profile
* POST /pet Add an new pet to the store
* GET /pet/{petId} Find pet by ID
* DELETE /pet/{petId} Delete a pet
* POST /pet/{petId}/uploadImage upload an Image
```

```
6_ As a Store Owner, I would like add update pet profile, view pet info by status or create new pet profile wi
Description : Create, Update a pet profile with body or form data and Find Pet By Status
* POST /pet Add an ew pet to the store
* GET /pet/findByStatus Finds Pets by status
* PUT /pet Add a new pet to the store
* POST /pet/{petId} Update ta pet in the store with form data
```



How to write new tests ?

- Step1. Create new feature file or Add Scenario (Outline, Given, When, Then) in a existing feature file with Valid Data

```
***** Sample Cucumber File *****

Scenario Outline: As a End User, Valid user can login and logout to the application using API
  Given I provide login credentials "<username>" and "<password>"
  When I send request to login
  Then login is successful
  Then logout is successful

Examples: Valid
  |username|password|
  |test|test@123|
```

- Step2. Add new methods with Given, When and Then Annotation in new/existing Step Defination class file.

```
***** Sample Class File *****

@RunWith(SerenityRunner.class)
public class LoginStepDefn{

    @Before
    public void setup()
    {
        RestAssured.baseURI = "https://petstore.swagger.io/v2";
    }

    @After
    public void tearDown()
    {
        RestAssured.reset();
    }

    @Given("^I provide login credentials \"(.*?)\" and \"(.*?)\"$")
```

```

    public void i_provide_login_credentials_and(String username, String password){
        //Write your code
    }

    @When("^I send request to login$")
    public void i_send_request_to_login() {
        // Write code here that turns the phrase above into concrete actions
    }

    @Then("^login is successful$")
    public void login_is_successful() {
        // Write code here that turns the phrase above into concrete actions
    }

    @Then("^logout is successful$")
    public void logout_is_successful() {
        // Write code here that turns the phrase above into concrete actions
    }
}

```

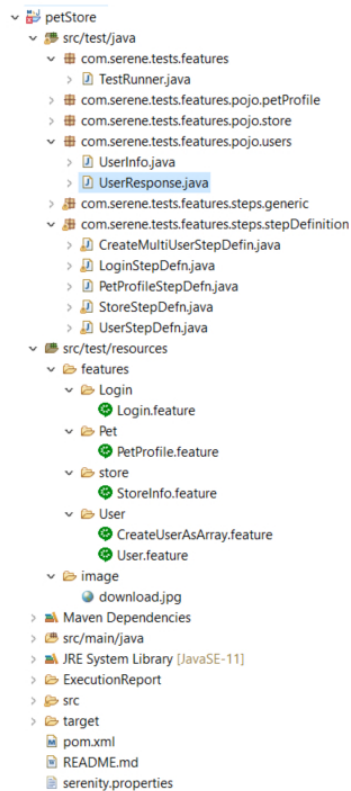
- Step3. Create a POJO class based on Request or Response JSON

* Create package like.. com.serene.tests.features.pojo.users

* Create class like.. UserInfo or UserResponse

For detail information, please refer usecase present at# <https://www.toolsqa.com/rest-assured/convert-json-to>

- Project Structure



Reference Test Report

