

Reporte generado Pruebas de Herramientas de automatización Serenity VS Karate(Services Rest)

Responsable: Yolima Alejandra Guadir

Fecha: 29/03/2023

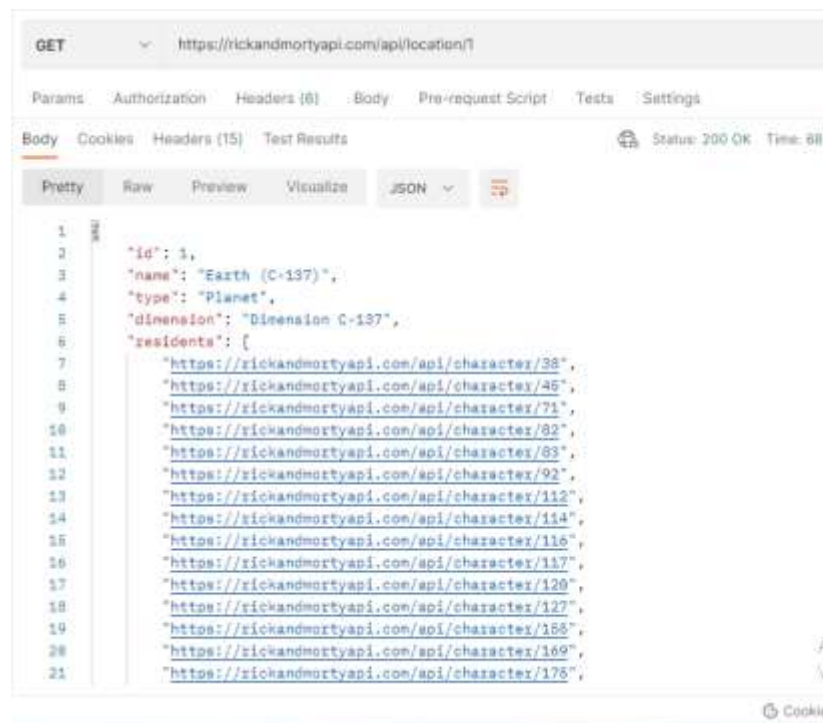
1. Resultados de pruebas Serenity



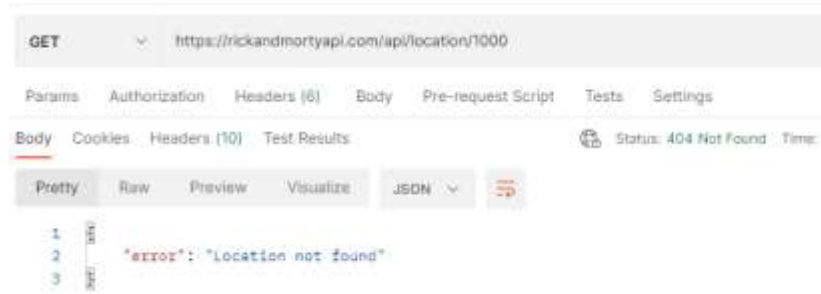
Figura 1: resultados de test de casos de prueba para servicios GET y DELETE.

GET

<https://rickandmortyapi.com/api/location/1>



GET
<https://rickandmortyapi.com/api/location/1000>



GET
<https://rickandmortyapi.com/api/location/>



DELETE
<https://jsonplaceholder.typicode.com/albums/1>



1. Resultados Karate(Services Rest)



2

0

Features

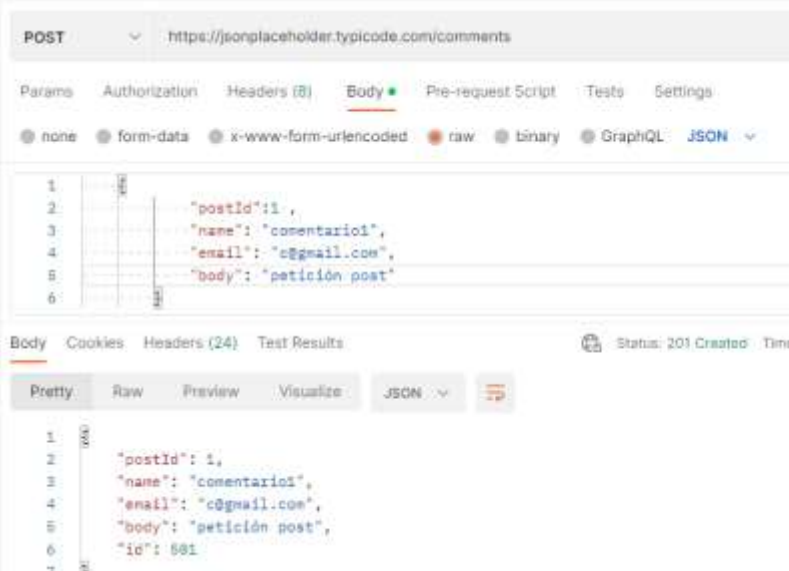
2023-03-29 12:38:33 p. m.

Tags Timeline					
Feature	Title	Passed	Failed	Scenarios	Time (ms)
src/test/java/features/comments.feature	crear comentarios en JSONPlaceholder	1	0	1	1328
src/test/java/features/user.feature	Actualizar datos de usuario en reqres	2	0	2	1123

Figura 2 resultados de test de casos de prueba para servicios POST Y PATCH

POST

<https://jsonplaceholder.typicode.com/comments>



POST <https://jsonplaceholder.typicode.com/comments>

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

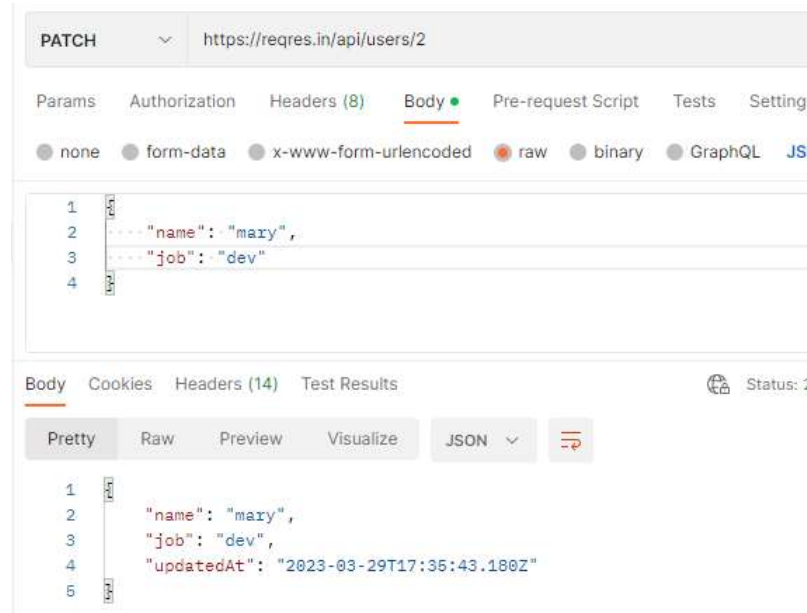
```
1  "postId": 1,
2  "name": "comentario1",
3  "email": "c@gmail.com",
4  "body": "petición post"
```

Body Cookies Headers (24) Test Results Status: 201 Created Time

Pretty Raw Preview Visualize JSON

```
1  {
2    "postId": 1,
3    "name": "comentario1",
4    "email": "c@gmail.com",
5    "body": "petición post",
6    "id": 581
7  }
```

PATCH
https://reqres.in/api/users/{id}



Conclusiones:

El escenario para obtener locations de la Api de rock and se implemento la petición GET para enviar una solicitud para obtener locations, el escenario verifica que la respuesta tenga un estado de código HTTP y que no sea nula la respuesta, si el status es 200 verifica que los campos name y type, en el caso de ser diferente retorna un mensaje con respecto al mensaje generado al realizar la petición.

El escenario para eliminar álbum, se implementó la petición DELETE para enviar una petición de eliminar álbum por Id, el escenario verifica que la respuesta tenga un estado de código que coincide con el código proporcionada en las tablas de ejemplos.

El escenario para actualizar crear comentarios se implementó la petición POST para enviar una solicitud para crear un comentario en la URL proporcionada, el escenario verifica que la respuesta tenga un estado de código HTTP que coincide con el código proporcionado en la tabla de ejemplos, también verifica que los campos "name" y "body" de la respuesta no sean nulos.

El escenario para actualizar datos utiliza la petición PATCH para enviar una solicitud para actualizar los datos de usuario en la URL proporcionada, la solicitud incluye un objeto JSON que contiene el nombre y el trabajo del usuario. El escenario verifica que la respuesta tenga un estado de código HTTP que coincida con el código proporcionado en la tabla de ejemplos, también verifica que los campos "name" y "job" de la respuesta no sean nulos y valida que el campo "name" sea diferente del campo "job".