

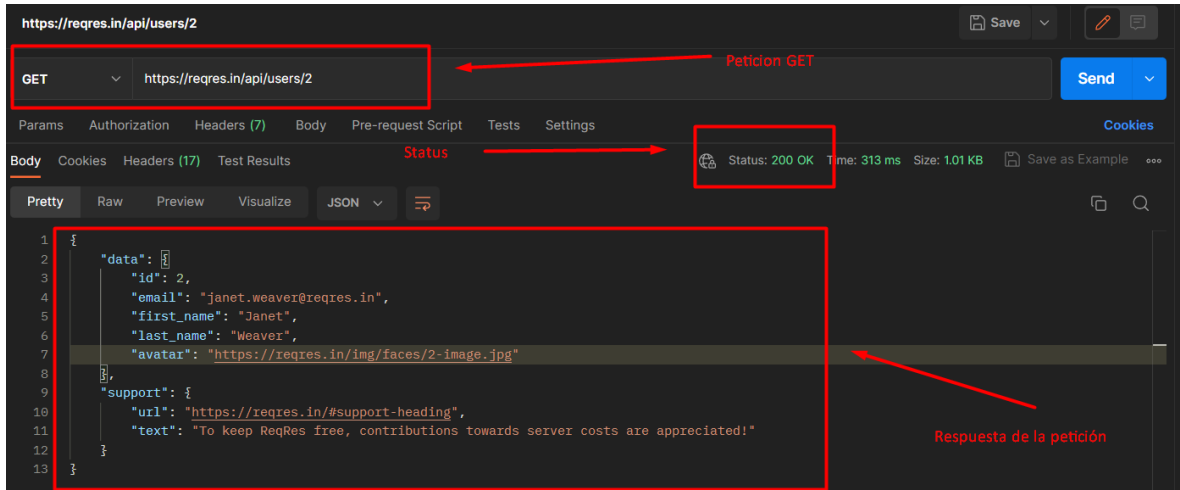
Herramientas de automatización Serenity VS Karate(Services Rest)

Pagina Seleccionada <https://reqres.in/>

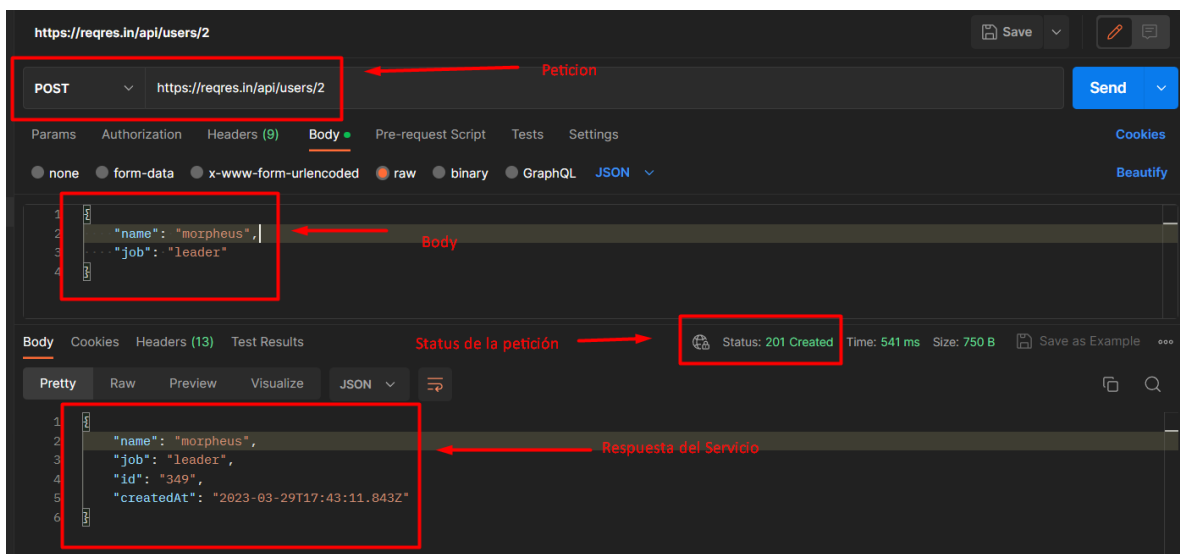
INFORME DE SERENITY BDD

Exploración del servicio en Postman

Servicio: Get Simple Usuario



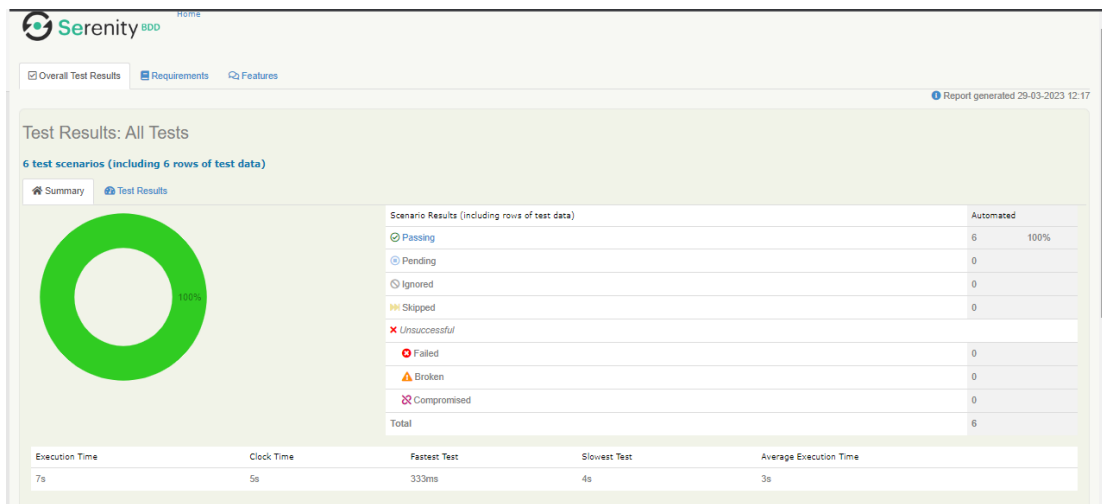
Servicio: Post Crear Usuario



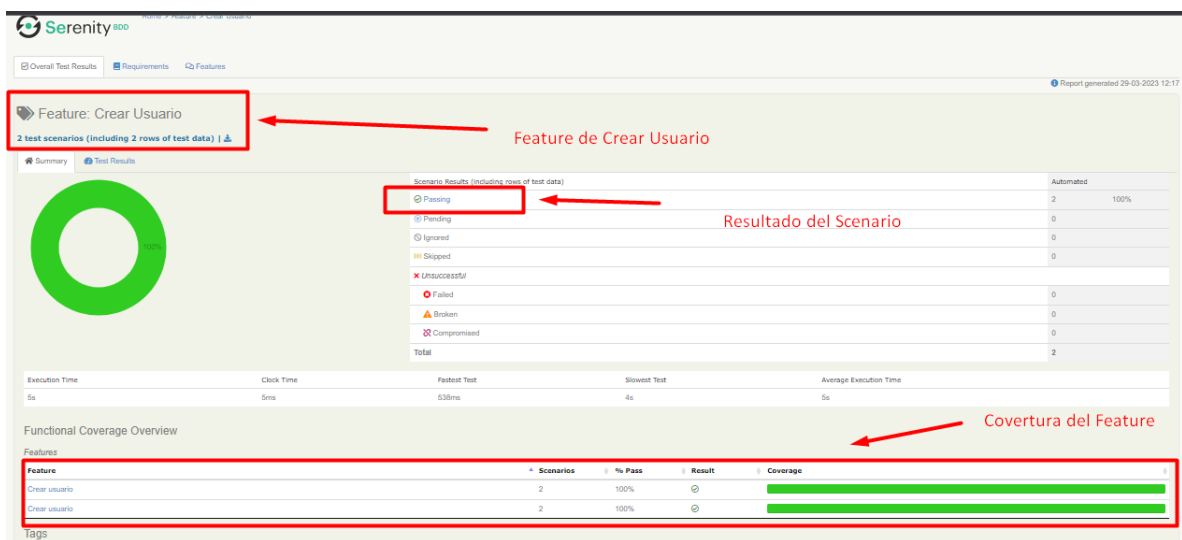
Servicios automatizados con Screenplay

- Get Single user
- Post Create

INFORME DE SERENITY BDD



Resultado individual del Feature Crear Usuario



Step del escenario

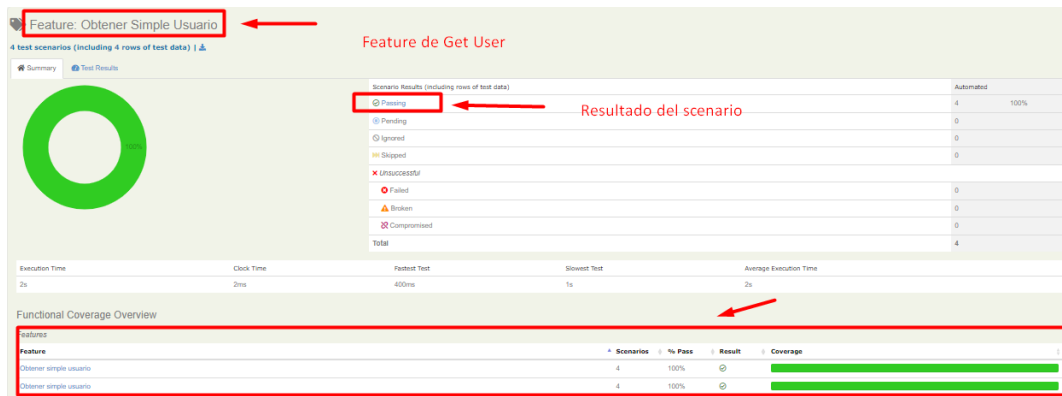
Examples:

#	Name	Job	Code
1	"morphous"	"leader"	201
2	"morphous"	"leader"	201

Steps

Steps	Outcome	Time
Example: (name="morphous", job="leader", code=201)	SUCCESS	5.86s
Given que el usuario se encuentra en la página crear usuario	SUCCESS	0.18s
When el usuario envia la petición de crea un nuevo usuario con "morphous", "leader"	SUCCESS	3.43s
Then el usuario debería ver un mensaje con información del nuevo usuario creado con un estatus 201	SUCCESS	0s
Example: (name="morphous", job="leader", code=201)	SUCCESS	0.56s
Given que el usuario se encuentra en la página crear usuario	SUCCESS	0.03s
When el usuario envia la petición de crea un nuevo usuario con "morphous", "leader"	SUCCESS	0.54s
Then el usuario debería ver un mensaje con información del nuevo usuario creado con un estatus 201	SUCCESS	0s
	SUCCESS	6.69s

Resultado individual del Feature Get User



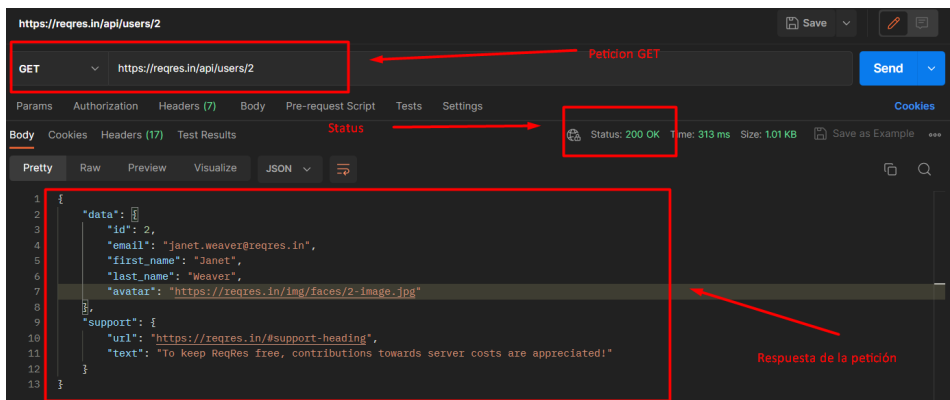
INFORME DE KARATE

Exploración del servicio en Postman

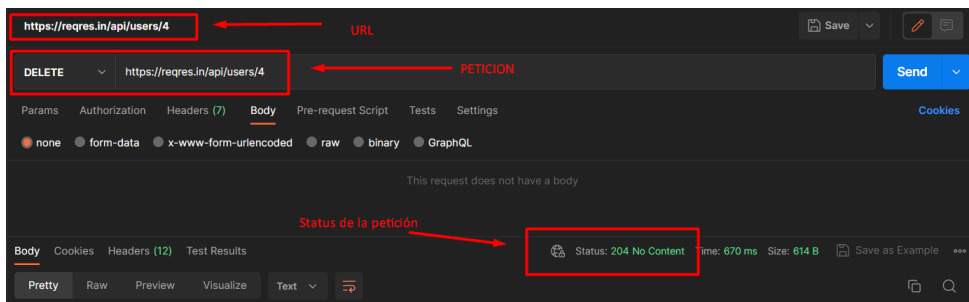
Servicios realizados

- Delete
- GetUser
- Patch
- Update
- TestOutline

Servicio: GetUser



Servicio: DeleteUser



Servicio: PutUser

The screenshot shows a REST client interface with the following details:

- URL:** `https://reqres.in/api/users/2`
- Method:** `PUT`
- Body:**

```
{  "name": "yeison osorio",  "job": "QA"}
```
- Status:** `200 OK` (Time: 332 ms, Size: 778 B)
- Response Body:**

```
{  "name": "yeison osorio",  "job": "QA",  "updatedAt": "2023-03-29T19:40:22.854Z"}
```

Ejecución de un Escenario Outline

The screenshot shows a Karate Scenario Outline for a POST request:

```
Feature: Post Outline

Scenario Outline: casos de prueba <nombreCaso2>
  Given url 'https://reqres.in/api/users'
  When request {"<name>": <Vname>,"<job>": <Vjob>}
  And method POST
  Then status 201
  Examples:
    | nombreCaso2 | name | Vname | job | Vjob |
    | name vacio  | name |       | job | "QA" |
    | name nulo   | name | null  | job | "QA" |
    | name 3 letras | name | "sol"  | job | "QA" |
    | sin campo name |      |        | job | "QA" |
    | job vacio    | name | "yeison" | job | ""   |
    | job nulo     | name | "yeison" | job | null |
    | job 3 letras | name | "yeison" | job | "QAs" |
    | sin campo job | name | "yeison" |      | ""   |
```

Reporte proporcionado por Karate


Se probaron 5 servicios con la herramienta karate

The screenshot shows a Karate test report with the following details:

- Summary:** 5 Passed, 0 Failed
- Tags:** Tags | Timeline
- Table:**

Feature	Title	Passed	Failed	Scenarios	Time (ms)
<code>src/test/java/features/delete.feature</code>	Delete User	1	0	1	3157
<code>src/test/java/features/getuser.feature</code>	Get User	1	0	1	294
<code>src/test/java/features/patchuser.feature</code>	Patch User	1	0	1	561
<code>src/test/java/features/test.feature</code>	Post Outline	8	0	8	4085
<code>src/test/java/features/updateuser.feature</code>	Update User	1	0	1	483

Vista de un Escenario donde vemos donde se ingresan los datos del escenario offline



8
0

Scenarios

2023-03-29 01:47:58 p. m.

[1.1:10] casos de prueba name vacio

[1.2:11] casos de prueba name nulo

[1.3:12] casos de prueba name 3 letras


[1.4:13] casos de prueba sin campo name

[1.5:14] casos de prueba job vacio

Summary | Tags | Feature: `src/test/java/features/test.feature` | **Post Outline** ← Scenario Outline

Scenario: [1.1:10] casos de prueba name vacio	ms: 555
4 Given url 'https://reqres.in/api/users'	1
5 When request {"name": "", "job": "QA"}	1
6 And method POST	553
7 Then status 201	0
Scenario: [1.2:11] casos de prueba name nulo	ms: 515
4 Given url 'https://reqres.in/api/users'	0
5 When request {"name": null, "job": "QA"}	0
6 And method POST	514
7 Then status 201	0
Scenario: [1.3:12] casos de prueba name 3 letras	ms: 492
4 Given url 'https://reqres.in/api/users'	0
5 When request {"name": "sol", "job": "QA"}	0
6 And method POST	491
7 Then status 201	0
Scenario: [1.4:13] casos de prueba sin campo name	ms: 512

Escenario Put de forma individual



1
0

Scenarios


2023-03-29 01:47:50 p. m.

[1:3] Put a user

Summary | Tags | Feature: `src/test/java/features/delete.feature` | Delete User

Scenario: [1:3] Put a user	ms: 3157
4 Given url "https://reqres.in" + "/api/users/" + "1"	108
5 And request { "name": "yeison", "job": "zion resident" }	271
6 When method delete	2778
7 Then status 204	0

Escenario Delete



1
0

Scenarios

2023-03-29 01:47:50 p. m.

[1:3] Put a user

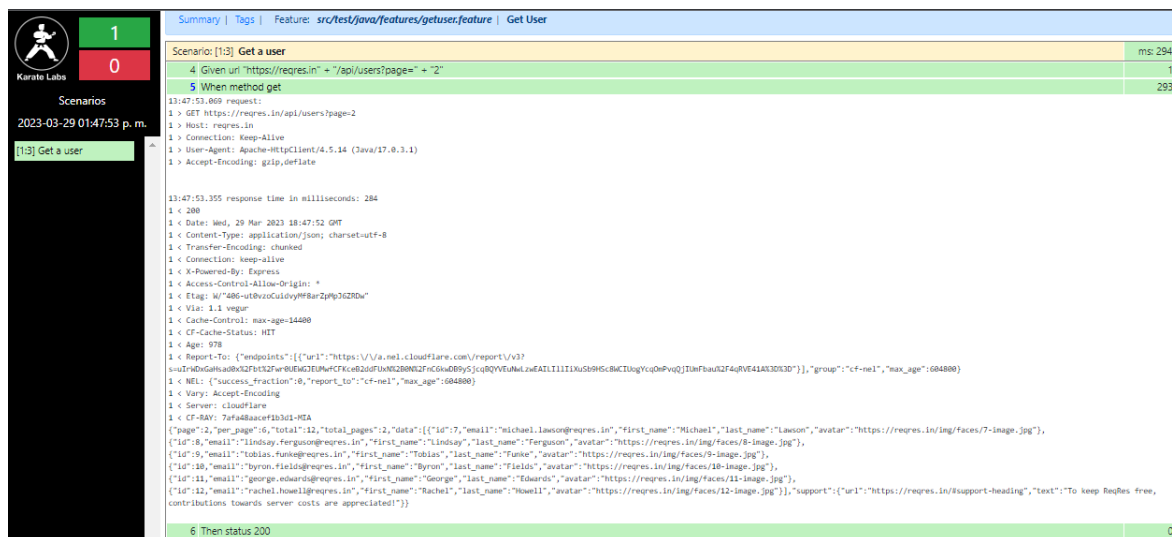
Summary | Tags | Feature: `src/test/java/features/delete.feature` | Delete User

Scenario: [1:3] Put a user	ms: 3157
4 Given url "https://reqres.in" + "/api/users/" + "1"	108
5 And request { "name": "yeison", "job": "zion resident" }	271
6 When method delete	2778
7 Then status 204	0

13:47:49.351 request:
1 > DELETE https://reqres.in/api/users/1
1 > Content-Type: application/json; charset=UTF-8
1 > Content-Length: 39
1 > Host: reqres.in
1 > Connection: Keep-Alive
1 > User-Agent: Apache-HttpClient/4.5.14 (Java/17.0.3.1)
1 > Accept-Encoding: gzip,deflate
{"name": "yeison", "job": "zion resident"}

13:47:50.838 response time in milliseconds: 1480
1 < 204
1 < Date: Wed, 29 Mar 2023 18:47:50 GMT
1 < Content-Length: 0
1 < Connection: keep-alive
1 < X-Powered-By: Express
1 < Access-Control-Allow-Origin: *
1 < Etag: W/"2-vyGpPvF04RvsfTpolwehYIC8"
1 < Via: 1.1 vegur
1 < CF-Cache-Status: DYNAMIC
1 < Report-To: [{"endpoints":[{"url":"https://l/a.ne1.cloudflare.com/report/v3? s-b0hX2d0p0wz120b5AVt29k9dVxVbV2FEDKSE6VoudR2JFis1co1W6hWfiesLd0-jPcgU5vC1EVh2BWHPlxhUQC185sXwR9125peAsqTKUId6PJ01k53k22r3VgQ30R3D"}], "group": "cf-ne1", "max_age": 604800}]
1 < NEL: [{"success_fraction": 0, "report_to": "cf-nel", "max_age": 604800}]
1 < Server: cloudflare
1 < CF-RAY: 7afa48995cbb8ddf-MIA

Escenario Get User



The screenshot shows the Karate Labs interface. On the left, there's a sidebar with 'Scenarios' and a list of scenarios. The 'Get a user' scenario is selected. The main panel displays the scenario details, including the URL, method, and response body. The response body is a JSON object containing user information.

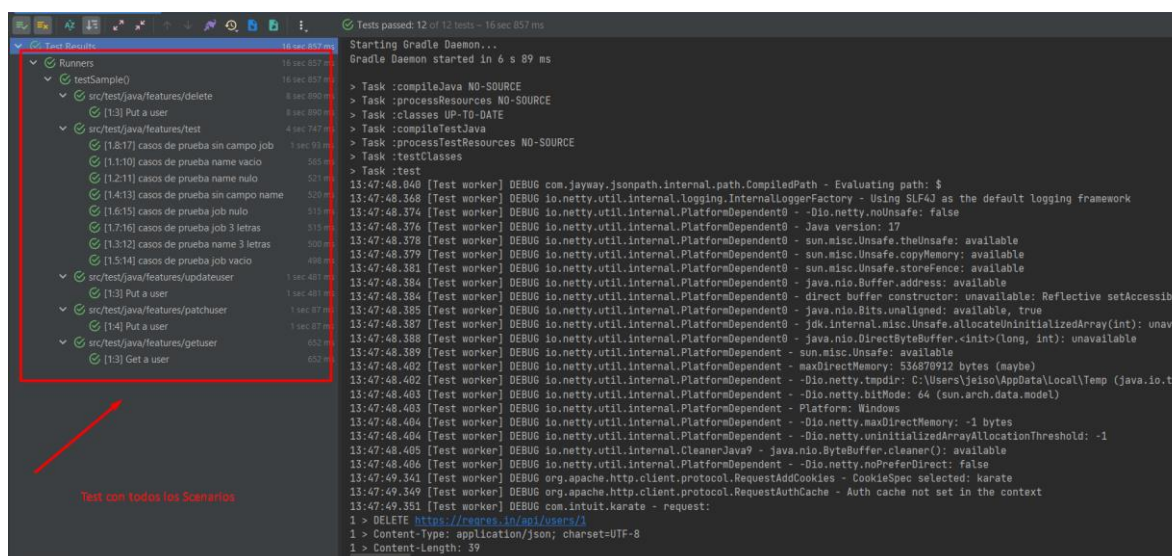
```
Scenario: [13] Get a user
1 > GET https://reqres.in/api/users?page=2
2 > Host: reqres.in
3 > Connection: Keep-Alive
4 > User-Agent: Apache-HttpClient/4.5.14 (Java/17.0.3.1)
5 > Accept-Encoding: gzip,deflate

13:47:53.069 request:
1 > GET https://reqres.in/api/users?page=2
2 > Host: reqres.in
3 > Connection: Keep-Alive
4 > User-Agent: Apache-HttpClient/4.5.14 (Java/17.0.3.1)
5 > Accept-Encoding: gzip,deflate

13:47:53.355 response time in milliseconds: 284
1 < 200
2 < Date: Wed, 29 Mar 2023 18:47:52 GMT
3 < Content-Type: application/json; charset=utf-8
4 < Transfer-Encoding: chunked
5 < Connection: keep-alive
6 < X-Powered-By: Express
7 < Access-Control-Allow-Origin: *
8 < Etag: W/"486-ut6vzCudvY#8ur2p3t2Dw"
9 < Via: 1.1 vegur
10 < Cache-Control: max-age=14400
11 < CF-Cache-Status: HIT
12 < Age: 978
13 < Report-To: {"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v3?u=rdndnawhchd32p3u0W0G2D0uCFv0u0b0u020W020c0u0b0y50u0Y0u042u0EAT1111X0u09HSc0Mc0UgY0u0p0v0J0u0f0u02F0u0E40X0D0"}],"group":"cf-nel","max_age":604800}}
14 < NEL: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
15 < Vary: Accept-Encoding
16 < Server: cloudflare
17 < CF-RAY: 79f4a8a0c1b03b7d-HBA

{"page":2,"per_page":6,"total":12,"total_pages":2,"data":[{"id":7,"email":"michael.lawson@reqres.in","first_name":"Michael","last_name":"Lawson","avatar":"https://reqres.in/img/faces/7-image.jpg"}, {"id":8,"email":"lindsay.ferguson@reqres.in","first_name":"Lindsay","last_name":"Ferguson","avatar":"https://reqres.in/img/faces/8-image.jpg"}, {"id":9,"email":"tobias.funke@reqres.in","first_name":"Tobias","last_name":"Funke","avatar":"https://reqres.in/img/faces/9-image.jpg"}, {"id":10,"email":"byron.felding@reqres.in","first_name":"Byron","last_name":"Felding","avatar":"https://reqres.in/img/faces/10-image.jpg"}, {"id":11,"email":"george.edwards@reqres.in","first_name":"George","last_name":"Edwards","avatar":"https://reqres.in/img/faces/11-image.jpg"}, {"id":12,"email":"rachel.howell@reqres.in","first_name":"Rachel","last_name":"Howell","avatar":"https://reqres.in/img/faces/12-image.jpg"}],"support":{"url":"https://reqres.in/support-heading","text":"To keep ReqRes free, contributions towards server costs are appreciated!"}]}
```

Run del id donde vemos todos los escenarios ejecutados



The screenshot shows the Karate Labs interface with the 'Runners' tab selected. The interface displays a list of runners and their execution status. A red box highlights the 'src/test/java/features/getuser' runner, and a red arrow points to it with the text 'Test con todos los Scenarios'.

```
Test con todos los Scenarios
```

Conclusiones del taller

Karate es una herramienta útil para empezar a realizar pruebas de software sin tener un conocimiento profundo de programación. Al trabajar con Karate, nos concentramos en escribir las especificaciones en lenguaje gherkin, lo que hace que sea más fácil de entender para todas las partes interesadas en el proyecto.

Karate no es la herramienta adecuada para todos los casos. A medida que las necesidades del proyecto se vuelven más complejas, puede ser necesario tener más control sobre el proceso

de pruebas. En este caso, una herramienta como Serenity puede ser más adecuada ya que ofrece recursos y capacidades más avanzadas, como dividir cada tarea para que solo realice una tarea en específico, ajustándonos al patrón de single responsibility.

Karate es una herramienta útil para empezar a realizar pruebas sin una gran experiencia en programación. Pero para proyectos más grandes y complejos, puede ser necesario utilizar herramientas más avanzadas y escalables como Serenity.