

## AUTOMATION CHALLENGE - POSTMAN

### Método: GET de usuarios

Se puede hacer dos tipos de peticiones para obtener usuarios:

- <https://gorest.co.in/public/v2/users>
- <https://gorest.co.in/public/v2/users?page=128>

goRest\_Tests / users

GET

<https://gorest.co.in/public/v2/users?page=116>

Se obtiene con este request una lista con objetos json, donde cada uno hace referencia a un usuario:

Body Cookies Headers (26) Test Results 200 OK 305 ms 1.58 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "id": 27,
4     "name": "Brahmdev Asan",
5     "email": "asan_brahmdev@skiles-kuphal.co",
6     "gender": "female",
7     "status": "active"
8   },
9   {
10    "id": 23,
11    "name": "Dr. Bhupati Achari",
12    "email": "achari_bhupati_dr@auer.org",
13    "gender": "female",
14    "status": "active"
15  },
16  {
```

Además, para este request, se obtiene un status code de 200

200 OK 305 ms 1.58 KB Save Response

El mismo comportamiento se tiene si se quiere hacer un request de todos los posts generados: <https://gorest.co.in/public/v2/posts> o de los comments realizados: <https://gorest.co.in/public/v2/comments>

## Método: DEL de usuario

Es necesario especificar al final de la URL el id del usuario que se desea eliminar. Por ejemplo, si se quiere eliminar el usuario con id 27, el request sería:

goRest\_Tests / user

DELETE



https://gorest.co.in/public/v2/users/27

Se debe tener en cuenta que para esta petición es necesario contar con autorización y esta información se especifica en el tab de **“Authorization”**. Se usa un tipo de **“Bearer token”** authorization y se indica un token que la API provee para realizar las pruebas:

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Type Bearer ...

The authorization header will be automatically generated when you send the request.  
[Learn more about authorization](#)

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. [Learn more about variables](#)

Token a9648006f605cb427d8214f2daec2d4b166 ...

Al hacer el request, no se devuelve un body y el status code es un 204:

Body Cookies Headers (17) Test Results 204 No Content 1011 ms 553 B Save Response

Pretty Raw Preview Visualize Text

1

En el caso de que el usuario a eliminar no se encuentre en el sistema, se recibe un body indicando que no se encontró el recurso y un status code de 404.

Body Cookies Headers (21) Test Results 404 Not Found 993 ms 706 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Resource not found"
3 }
```

Los otros tipos de peticiones que se pueden realizar son:

- **PUT** → <https://gorest.co.in/public/v2/users/51> Indicando en este caso ID del usuario a actualizar (Necesita token de autorización y body con la información a actualizar). Devuelve status code de **200**.
- **POST** → <https://gorest.co.in/public/v2/users> Necesita autorización y body con la información del nuevo usuario. Devuelve un **201** como status code.
- **PATCH** → <https://gorest.co.in/public/v2/users/49> Necesita autorización y body con campos de usuario a cambiar. Da un **200** como status code.