



INSTITUTO SUPERIOR TECNOLÓGICO DEL AZUAY

NOMBRE:

Xavier Yanza

CURSO:

Examen Complexivo

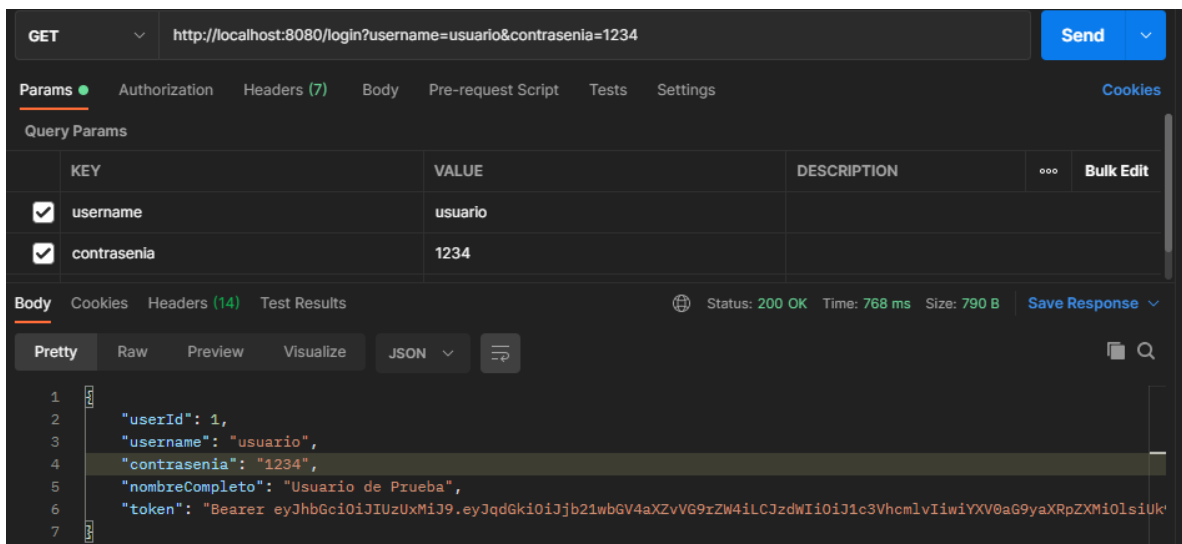
MATERIA:

Tendencias Actuales de Programación

DOCENTE:

Ing. Pedro Cornejo

Anexo 1: Crear usuario de prueba.



Anexo 3: Creación de usuarios con ayuda del token.

The screenshot shows a REST client interface with a POST request to `http://localhost:8080/crear-usuario...`. The **Headers** tab is active, showing an **Authorization** header with the value `Bearer eyJhbGciOiJIUzUxMiJ9.eyJqdGkiOiJjb21...`. The **Body** tab is also active, displaying a JSON object in the **Pretty** view:

```
1 {
2   "userId": 2,
3   "username": "jessi",
4   "contrasenia": "12345",
5   "nombreCompleto": "Jessica Alvarez",
6   "token": null
7 }
```

The response status is **200 OK**, with a time of **1451 ms** and a size of **611 B**.

Anexo 4: Prueba de creación de usuario sin token.

The screenshot shows the same REST client interface, but the **Authorization** header is now unchecked. The **Body** tab displays a JSON object in the **Pretty** view:

```
1 {
2   "timestamp": "2021-05-03T18:40:09.136+00:00",
3   "status": 403,
4   "error": "Forbidden",
5   "message": "Access Denied",
6   "path": "/crear-usuario"
7 }
```

The response status is **403 Forbidden**, with a time of **259 ms** and a size of **481 B**.

Anexo 5: Lista de usuarios con ayuda del token.

GET Send

Params Authorization **Headers (9)** Body Pre-request Script Tests Settings Cookies

Headers 8 hidden

	KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	Authorization	Bearer eyJhbGciOiJIUzUxMiJ9.eyJqdGkiOiJjb21...				
	Key	Value	Description			

Body Cookies (1) Headers (14) Test Results Status: 200 OK Time: 106 ms Size: 839 B Save Response

Pretty Raw Preview Visualize JSON ≡

```
1 {
2   {
3     "userId": 1,
4     "username": "usuario",
5     "contrasenia": "1234",
6     "nombreCompleto": "Usuario de Prueba",
7     "token": null
8   },
9   {
10    "userId": 2,
11    "username": "jessi",
12    "contrasenia": "12345".
```

Anexo 6: Actualización de Usuario con ayuda del token.

PUT Send

Params Authorization **Headers (11)** Body Pre-request Script Tests Settings Cookies

Headers 10 hidden

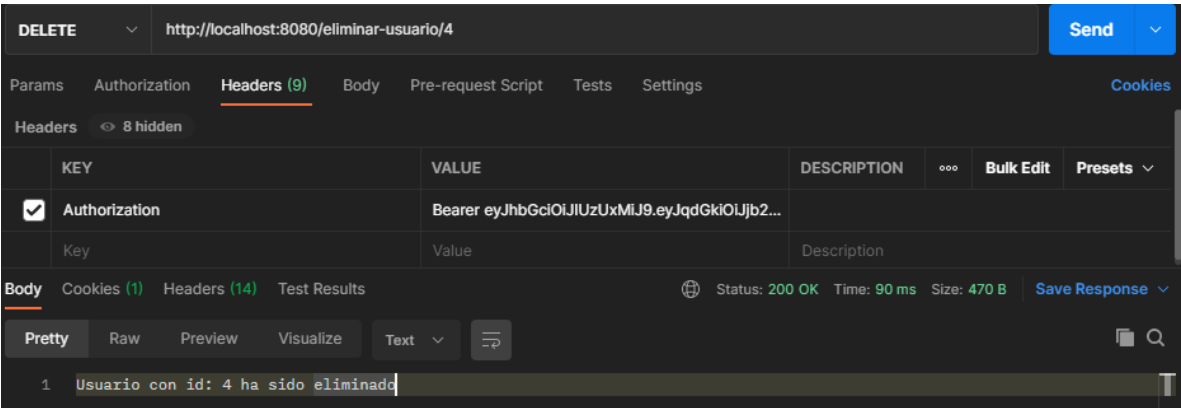
	KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	Authorization	Bearer eyJhbGciOiJIUzUxMiJ9.eyJqdGkiOiJjb21...				
	Key	Value	Description			

Body Cookies (1) Headers (15) Test Results Status: 200 OK Time: 598 ms Size: 617 B Save Response

Pretty Raw Preview Visualize JSON ≡

```
1 {
2   "userId": 1,
3   "username": "Juan Cardenas",
4   "contrasenia": "56789",
5   "nombreCompleto": "Juan Cardenas",
6   "token": null
7 }
```

Anexo 7: Eliminación de usuario con ayuda del token.



Evidencias de consumos desde Front end con Angular

Anexo 8: Login en la parte de Front end.

Login

Username

jessi

Password

.....

Login

Anexo 9: Lista de usuarios desde Front end.

Lista de Usuarios

ID	Username	Nombre Completo
1	Juan Cardenas	Juan Cardenas
2	mary	Mary Mejia
3	jessi	Jessica Alvarez
4	liss	Lisbeth Ortega
5	mauro	Mauro Samaniego
6	alvaro	Alvaro Bermeo

Conclusión:

En conclusión, se puede indicar desde mi punto de vista que el uso del token es una gran alternativa para proporcionar un grado de seguridad al proyecto a realizar, ya que de esta forma se puede tener un mejor control sobre el manejo y manipulación de información dentro del sistema. Además, se puede indicar que para el manejo del token se necesita tener conocimiento acerca de la sintaxis que este uso para de esta forma poder emplearlo de una mejor manera.