

## Pruebas Funcionales Automatizadas

---

### Caso de Prueba 1: Obtener un usuario específico

- **ID del Caso de Prueba:** TC\_01
- **Título:** Obtener un usuario específico y validar la respuesta
- **Objetivo:** Verificar que se pueda obtener correctamente la información de un usuario específico mediante una solicitud GET a la API.

#### Paso a Paso:

1. El desarrollador realiza una solicitud GET para obtener el usuario con ID 2.
2. Verificar que el código de estado de la respuesta sea 200.
3. Validar que la respuesta del servicio contenga el valor "fuchsia rose" en la ruta data.name.

### Caso de Prueba 2: Registrar un nuevo usuario con nombre y trabajo

- **ID del Caso de Prueba:** TC\_02
- **Título:** Registrar un nuevo usuario con nombre y trabajo y validar la respuesta
- **Objetivo:** Confirmar que se pueda registrar un nuevo usuario con su nombre y trabajo utilizando una solicitud POST a la API.

#### Paso a Paso:

1. El desarrollador realiza una solicitud POST para crear un nuevo usuario con los siguientes datos:
  - name: "Sebastian"
  - job: "Automation Engineer"
2. Verificar que el código de estado de la respuesta sea 201.
3. Validar que la respuesta del servicio contenga el valor "Sebastian" en la ruta name.
4. Validar que la respuesta del servicio contenga el valor "Automation Engineer" en la ruta job.

### Caso de Prueba 3: Actualizar información de un usuario existente

- **ID del Caso de Prueba:** TC\_03
- **Título:** Actualizar información de un usuario y validar la respuesta
- **Objetivo:** Asegurar que se pueda actualizar la información de un usuario existente mediante una solicitud PUT a la API.

**Paso a Paso:**

1. El desarrollador realiza una solicitud PUT para actualizar el usuario con ID 3 con los siguientes datos:
  - name: "Sebas put"
  - job: "Automation put"
2. Verificar que el código de estado de la respuesta sea 200.
3. Validar que la respuesta del servicio contenga el valor "Sebas put" en la ruta name.
4. Validar que la respuesta del servicio contenga el valor "Automation put" en la ruta job.

**Caso de Prueba 4: Eliminar un usuario existente**

- **ID del Caso de Prueba:** TC\_04
- **Título:** Eliminar un usuario existente y validar la respuesta
- **Objetivo:** Confirmar que se pueda eliminar correctamente un usuario existente utilizando una solicitud DELETE a la API.

**Paso a Paso:**

1. El desarrollador realiza una solicitud DELETE para eliminar el usuario creado con ID 5.
2. Verificar que el código de estado de la respuesta sea 204 (No Content).

**Caso de Prueba 5: Registrar un nuevo usuario con correo electrónico y contraseña**

- **ID del Caso de Prueba:** TC\_05
- **Título:** Registrar un nuevo usuario con correo electrónico y contraseña y validar la respuesta
- **Objetivo:** Asegurar que se pueda registrar un nuevo usuario utilizando su correo electrónico y contraseña mediante una solicitud POST a la API.

**Paso a Paso:**

1. El desarrollador realiza una solicitud POST para crear un nuevo registro con los siguientes datos:
  - email: "eve.holt@reqres.in"
  - password: "pistol"
2. Verificar que el código de estado de la respuesta sea 200.

### Estrategia General

La configuración presentada parece estar diseñada para realizar una prueba de carga escalonada, donde la carga en el sistema se incrementa en etapas, cada una con un aumento significativo en el número de usuarios virtuales (hilos). Esto permite observar cómo el sistema responde a diferentes niveles de carga, desde una carga baja hasta una carga muy alta.

- **Carga Escalonada:** La prueba aumenta gradualmente la carga desde 10 usuarios hasta 500 usuarios en tres etapas distintas.
- **Inicio Inmediato:** Todas las etapas tienen un Ramp-Up Period de 1 segundo, asegurando que la carga se aplique rápidamente para evaluar el impacto inmediato en el sistema.
- **Duración Variable:** Cada etapa tiene una duración mayor que la anterior, proporcionando un tiempo suficiente para observar el comportamiento del sistema bajo diferentes niveles de carga.

### Configuración de Pruebas de Carga

1. Thread Group 1
  - Number of Threads (Users): 10
  - Ramp-Up Period (seconds): 1
  - Duration (seconds): 10
2. Thread Group 2
  - Number of Threads (Users): 100
  - Ramp-Up Period (seconds): 1
  - Duration (seconds): 20
3. Thread Group 3
  - Number of Threads (Users): 500
  - Ramp-Up Period (seconds): 1
  - Duration (seconds): 30