

# Arquitectura de Software en la Práctica

## jMeter

---

### Objetivo

Definir y ejecutar un plan de pruebas con JMeter

### Procedimiento

Apache JMeter se puede utilizar para probar la performance de aplicaciones web.

Se puede utilizar para simular carga en un servidor, grupo de servidores, red, etc. y analizar el rendimiento general bajo diferentes tipos de carga.

Podemos ejecutar las pruebas en la *GUI de jMeter* o por *consola*.

### Construyendo un test plan

Un test plan describe una serie de pasos que ejecutará jMeter al realizar la prueba. Un plan de prueba completo constará de uno o más *thread groups*, *logic controllers*, *sample generating controllers*, *listeners*, *timers*, *assertions*, y *configuration elements*.

### Elementos de un test plan

Esta sección describe las diferentes partes de un plan de prueba.

Una prueba mínima consistirá en un *test plan*, un *thread group* y uno o mas *samplers*.

- **Test plan:** Es el plan de ejecución de prueba en sí
- **Thread group:** es el punto de inicio de cualquier *test plan*. Todos los *controllers* y *samplers* tienen que estar bajo un *thread group*. Como el nombre indica los *thread group* controlan la cantidad de threads que jMeter va a utilizar para ejecutar la prueba.

Podemos configurar:

- Cantidad de threads en el *thread group* (cada thread lo consideramos como un usuario)
- Periodo de ramp-up
- La cantidad de veces que vamos a ejecutar la prueba

*Nota:* Cada thread ejecutará el test plan en su totalidad y de forma completamente independiente de otros threads de prueba. Se utilizan varios subprocesos para simular conexiones simultáneas a la aplicación.

- **Controllers:**
  - **Samplers:** los *samplers* le dicen a jMeter que mande solicitudes a un servidor, por ejemplo, podemos realizar solicitudes HTTP utilizando un *HTTP sampler*
  - **Logical controllers:** permiten customizar la lógica en que enviamos las peticiones al servidor. Por ejemplo, podemos usar un *interleave logic controller* para alternar las llamadas entre dos *http samplers*
- **Listeners:** los *listeners* proveen acceso a la información mientras jMeter obtiene información sobre los test cases al ejecutar las pruebas.  
Algunos de ellos son:
  - Summary report
  - Graph results
  - Aggregate result
  - Response time graph
  - View tree results

## Servicio de prueba

Para realizar este ejercicio se realizó un API de “Todos” publicada en el siguiente endpoint:

<http://backend-app-alb-266270533.us-east-1.elb.amazonaws.com/>

El endpoint de pruebas a utilizar es:

- **GET /todos:** Donde retorna una lista de tareas con el siguiente formato JSON:

```
{
  "id": "3555c919-e890-4c71-8b6e-d04a2f80694b",
  "title": "Todo title",
  "order": 0,
  "completed": false,
  "url": "https://backend-app-alb-266270533.us-east-1.elb.amazonaws.com/todos"
}
```

- **GET /todos?time=2000:** Adicionalmente podemos indicar en el *query string* el parámetro *time* para simular demoras en el servidor, este valor se encuentra en milisegundos

## Test plan de ejemplo

1. Crear un *thread group* de 2 threads y agregar *samplers* hacia */todos*. Ver resultados en un *listener view results tree*

2. Agregar los *listeners summary report*, *aggregate report* y *graph report*. Comentar diferencias y prestaciones
3. Agregar otro *sampler* hacia el endpoint que simula demoras. Ver resultados en un *listener response time graph*
4. Aumentar cantidad de *threads* en 50, 100 y 200. Comparar resultados y comentar.
5. Lanzar prueba con 200 *threads*, luego de 2 minutos escalar el servicio y comentar resultados.