

Universidad de Ingenieria y Tecnologia
Departamento de Ciencias de la Computación

Ingeniería de Software I

Lab VIII: Adding Basic Telemetry
2024 - II

TEAM	SegmentationFault
INTEGRANTES	Christopher Najarro Mancco Mauricio Salazar Hillenbrand Leonardo Matias Candio Ormeño Manyory Cueva Mendoza Angel Ulises Tito Berrocal
Puntos EC	2 ptos

Background:

JMeter

Performance Testing

CASO DE ESTUDIO (level: intermediate):

1. JMeter (Performance Testing):

- Elija dos endpoints del backend de su proyecto.

POST	/auth/register	Register User	▼
POST	/content/definir_campana	Definir Campana	▼
POST	/content/definir_publico_ubicaciones	Definir Publico Ubicaciones	▼
POST	/content/elegir_formato_cta	Elegir Formato Cta	▼
POST	/content/crear_contenido_creativo	Crear Contenido Creativo	▼
POST	/content/create_heading	Create Heading	▼

- Correr JMeter y obtener el response time para cada uno de los endpoints seleccionados

Type	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s	95%ile (ms)	99%ile (ms)
POST	/content/crear_contenido_creativo	31	0	4535.57	2081	11693	693.77	0.06	0	6500	9900
POST	/content/create_heading	26	0	3601.31	899	11789	195.85	0.05	0	10000	12000
POST	/content/definir_campana	23	0	5211.31	1994	12249	487.65	0.05	0	10000	11000
POST	/content/definir_publico_ubicaciones	20	0	4695.54	2981	12083	769.95	0.04	0	9500	12000
POST	/content/elegir_formato_cta	22	0	4689.17	1674	11435	495.73	0.04	0	8700	11000

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
POST	/auth/register	30	0	4600	5200	5400	4179.17	1642	5439	39	0.9	0
	Aggregated	30	0	4600	5200	5400	4179.17	1642	5439	39	0.9	0

2. Telemetry

- Crear la tabla **Tracking** en su base de datos.

Tabla: Tracking

Tracking

TraceId

UserId

Route (nombre del endpoint que está accediendo)

HttpStatusCode (Status que devuelve la API: 400, 500)

StartDate - tiempo que inició el request

EndDate - tiempo que finalizó el request

Latency (EndDate - StartDate)

- Crear una clase llamada **TrackingHelper** que tendra un metodo llamado CreateMetric y que se invocará al finalizar la llamada a su endpoint. **Esta clase no debe bloquear la ejecucion o response time de su endpoint.** Revise alternativas como mandar a una cola o que sea asincrono.
- Ejecute el jmeter nuevamente y valide que la tabla se esta llenando con data y con los StatusCode correctos.

alembic_version usuarios console_1 tracking									
105 rows									
WHERE ORDER BY									
TraceId	UserId	Route	HttpStatusCode	StartDate	EndDate	Latency			
1	32	756 /auth/register	200	2024-11-06 00:17:13.937513	2024-11-06 00:17:14.819726	0.882213			
2	33	757 /auth/register	200	2024-11-06 00:17:15.039050	2024-11-06 00:17:15.922808	0.883758			
3	34	758 /auth/register	200	2024-11-06 00:17:16.141441	2024-11-06 00:17:17.026989	0.885548			
4	35	759 /auth/register	200	2024-11-06 00:17:17.245736	2024-11-06 00:17:18.124105	0.878369			
5	36	760 /auth/register	200	2024-11-06 00:17:18.342255	2024-11-06 00:17:19.232025	0.889777			
6	37	761 /auth/register	200	2024-11-06 00:17:19.454810	2024-11-06 00:17:20.343079	0.888269			
7	38	762 /auth/register	200	2024-11-06 00:17:20.560864	2024-11-06 00:17:21.436525	0.875661			
8	39	763 /auth/register	200	2024-11-06 00:17:21.653816	2024-11-06 00:17:22.540845	0.887029			
9	40	764 /auth/register	200	2024-11-06 00:17:22.764766	2024-11-06 00:17:23.646113	0.881347			
10	41	765 /auth/register	200	2024-11-06 00:17:23.866916	2024-11-06 00:17:24.744127	0.877211			
11	42	766 /auth/register	200	2024-11-06 00:17:24.964097	2024-11-06 00:17:25.839752	0.875655			
12	43	767 /auth/register	200	2024-11-06 00:17:26.059898	2024-11-06 00:17:26.942817	0.882919			
13	44	768 /auth/register	200	2024-11-06 00:17:27.166809	2024-11-06 00:17:28.053380	0.886571			
14	45	769 /auth/register	200	2024-11-06 00:17:28.518533	2024-11-06 00:17:29.400214	0.881681			
15	46	770 /auth/register	200	2024-11-06 00:17:29.619159	2024-11-06 00:17:30.503939	0.88478			
16	47	771 /auth/register	200	2024-11-06 00:17:30.722290	2024-11-06 00:17:31.605535	0.883245			
17	48	772 /auth/register	200	2024-11-06 00:17:31.825361	2024-11-06 00:17:32.703305	0.877944			
18	49	773 /auth/register	200	2024-11-06 00:17:32.921295	2024-11-06 00:17:33.801737	0.880442			
19	50	774 /auth/register	200	2024-11-06 00:17:34.024872	2024-11-06 00:17:34.903848	0.878976			
20	51	775 /auth/register	200	2024-11-06 00:17:35.157579	2024-11-06 00:17:36.043234	0.885655			
21	52	776 /auth/register	200	2024-11-06 00:17:36.262465	2024-11-06 00:17:37.140315	0.87785			
22	53	777 /auth/register	200	2024-11-06 00:17:37.358033	2024-11-06 00:17:38.238558	0.880525			
23	54	778 /auth/register	200	2024-11-06 00:17:38.460680	2024-11-06 00:17:39.341606	0.880926			
24	55	779 /auth/register	200	2024-11-06 00:17:39.559256	2024-11-06 00:17:40.435832	0.876576			
25	56	780 /auth/register	200	2024-11-06 00:17:40.653664	2024-11-06 00:17:41.535104	0.88144			
26	57	781 /auth/register	200	2024-11-06 00:17:41.753467	2024-11-06 00:17:42.631064	0.877597			
27	58	782 /auth/register	200	2024-11-06 00:17:42.856177	2024-11-06 00:17:43.737715	0.874096			

- Calcule el availability y reliability de sus endpoints.

Todas las solicitudes fueron exitosas (código 2xx), por lo que no hay errores 4xx ni 5xx
 Esto indica que el endpoint fue totalmente confiable y estuvo disponible durante todas las solicitudes en esta prueba.

Tienen un availability y un reliability del 100%.