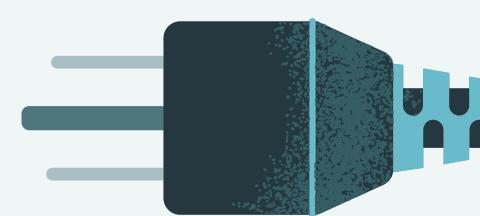
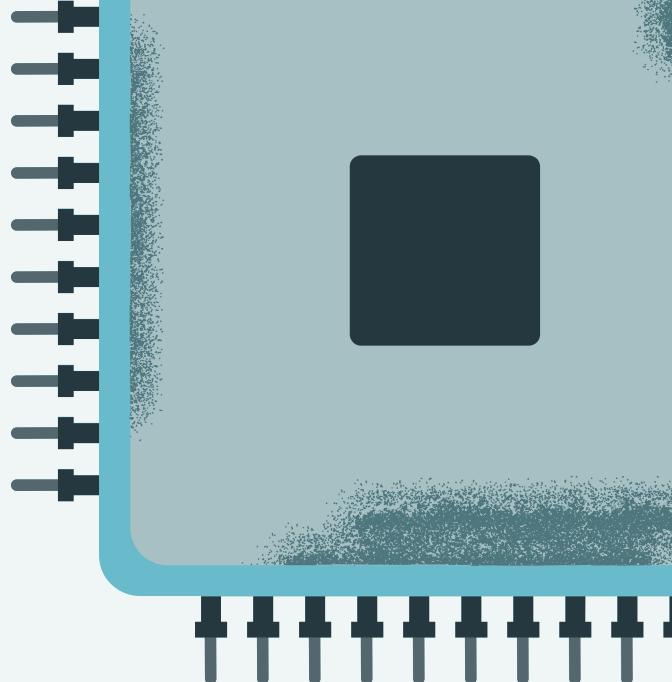
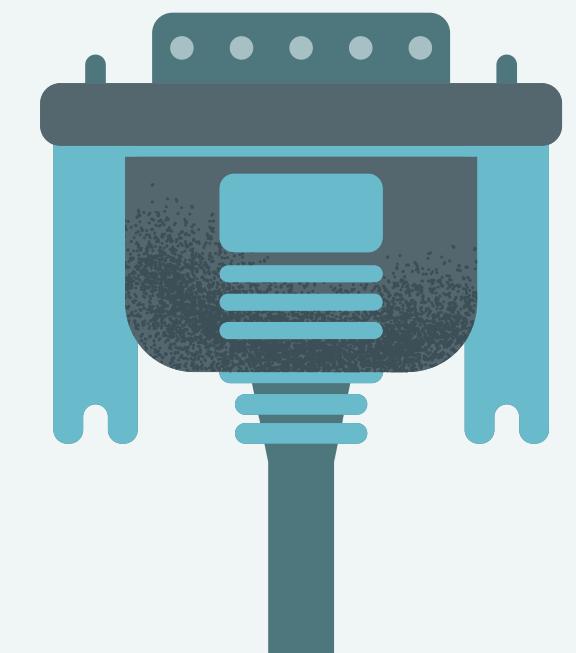
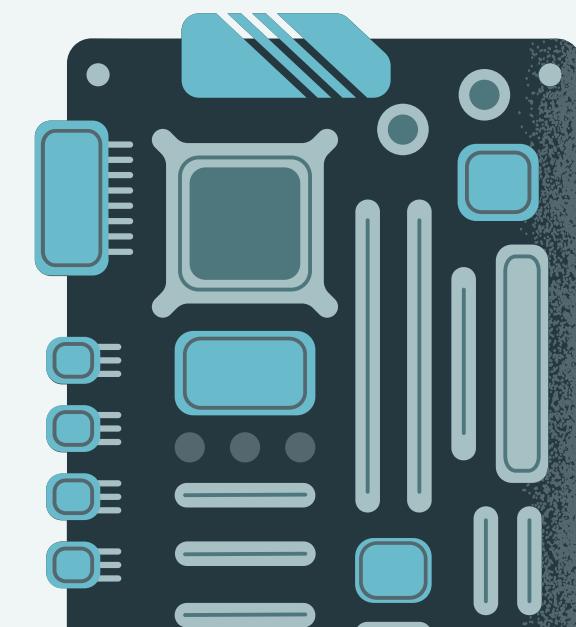
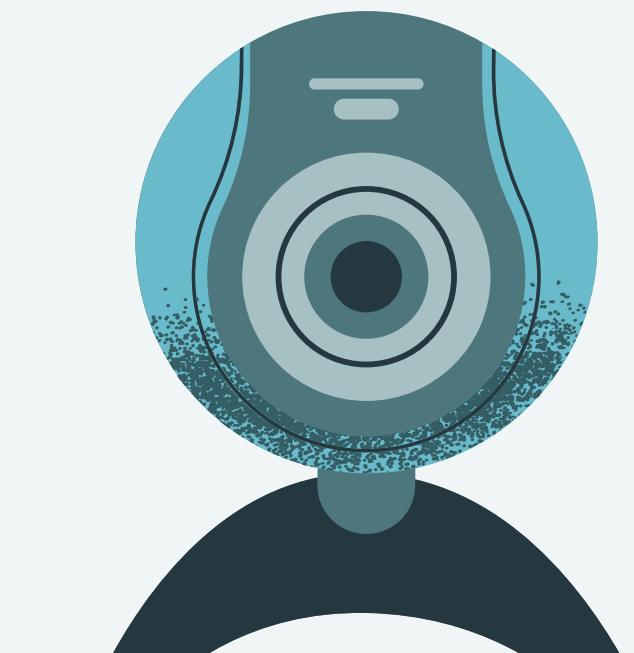
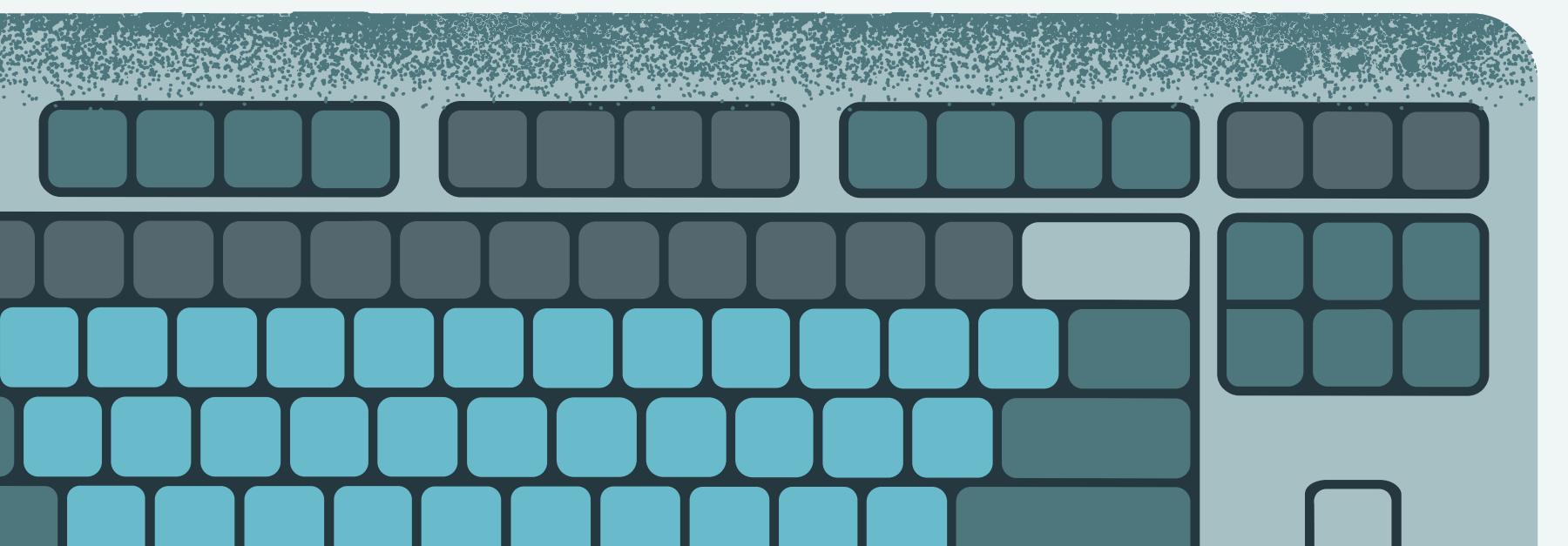
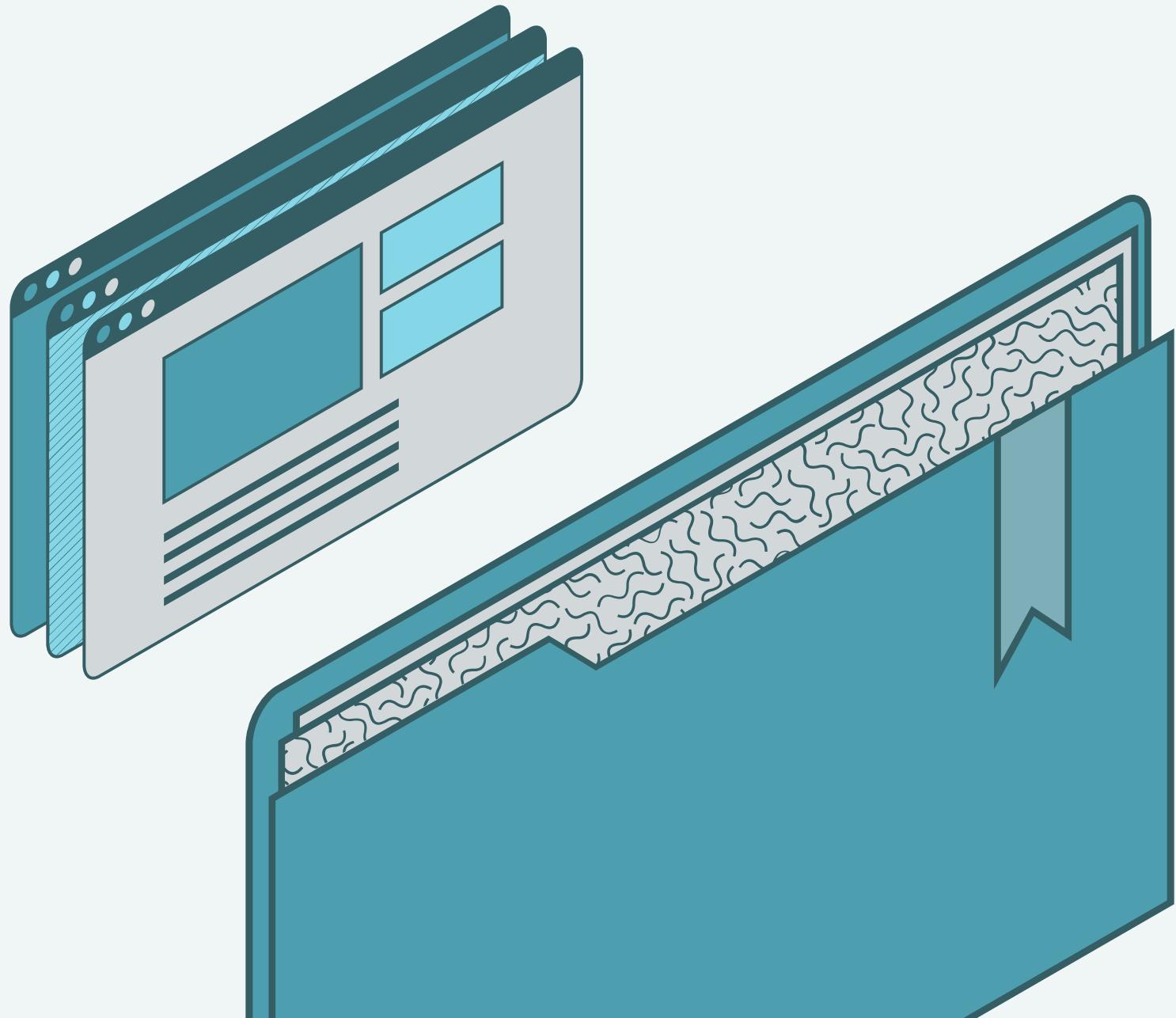


TESTING DE RENDIMIENTO Y ESTRES



ÍNDICE



-
- 01. Conceptos de rendimiento, carga y estrés**

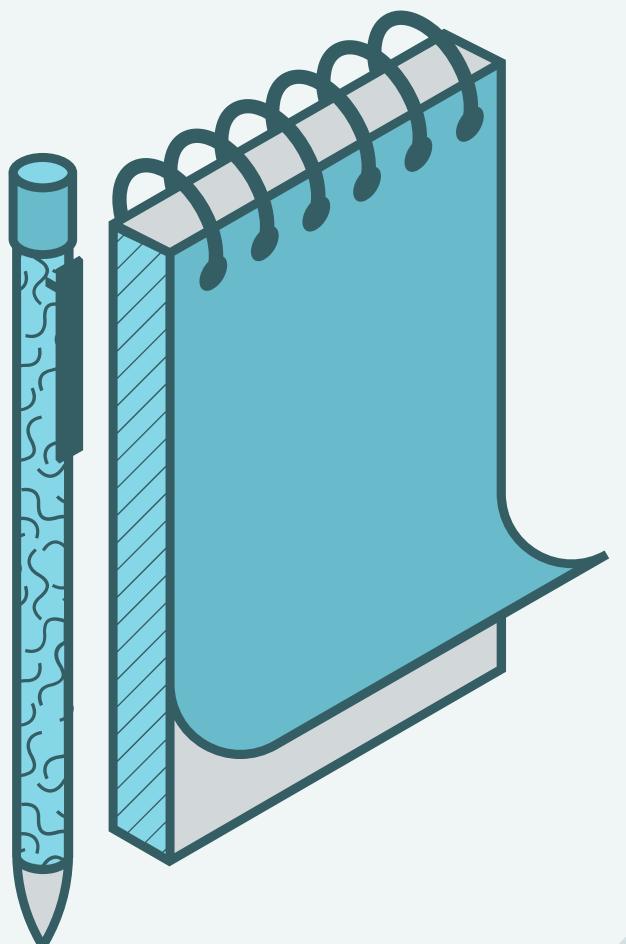
 - 02. Tipos de pruebas**

 - 03. Métricas claves**

 - 04. Herramientas populares**
-

RENDIMIENTO, CARGA Y ESTRES

- **Rendimiento:** Capacidad de una aplicación para responder eficientemente a las peticiones de los usuarios.
- **Carga:** Número de usuarios o peticiones concurrentes que una aplicación puede manejar correctamente.
- **Estrés:** Situación en la que la aplicación se somete a más carga de la que puede manejar para detectar su punto de quiebre.



TIPOS DE PRUEBAS: CARGA, ESTRÉS, ESCALABILIDAD Y DURACIÓN

- **Prueba de carga:** Verifica cómo se comporta la app bajo un volumen normal o creciente de usuarios.
- **Prueba de estrés:** Evalúa el comportamiento bajo sobrecarga, hasta que falle.
- **Prueba de escalabilidad:** Evalúa cómo responde la app al aumentar recursos o instancias.
- **Prueba de duración:** Verifica la estabilidad durante un período largo de tiempo.

EJEMPLO:

- SIMULAR 100, 500 Y 1000 USUARIOS Y OBSERVAR TIEMPOS DE RESPUESTA.
- DEJAR UNA PRUEBA CORRIENDO POR 10 MINUTOS PARA SIMULAR PRUEBA DE DURACIÓN.

MÉTRICAS CLAVES

- **Tiempo de respuesta (Response Time):** Tiempo desde que el usuario hace una solicitud hasta recibir una respuesta.
- **Throughput:** Número de peticiones procesadas por segundo (RPS - requests per second).
- **Errores:** Códigos 4xx, 5xx u otros fallos.
- **Disponibilidad:** Porcentaje del tiempo que el sistema está operativo.



MÉTRICAS CLAVES

Ejemplo de reporte generado por Artillery o JMeter:

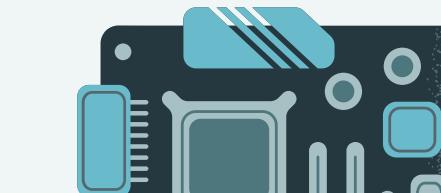
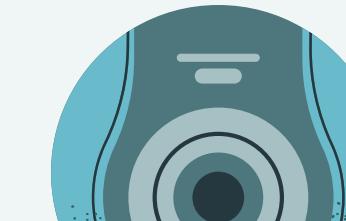
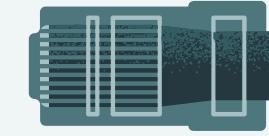
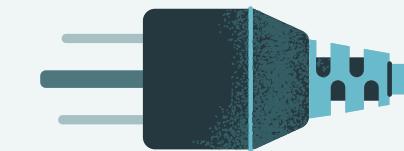
```
{  
  "latency": { "min": 10, "max": 1200, "median": 250 },  
  "requestsCompleted": 1500,  
  "rps": { "mean": 25 },  
  "errors": { "ECONNRESET": 10 }  
}
```



HERRAMIENTAS POPULARES



Artillery





Artillery

Instalación: *npm install -g artillery*

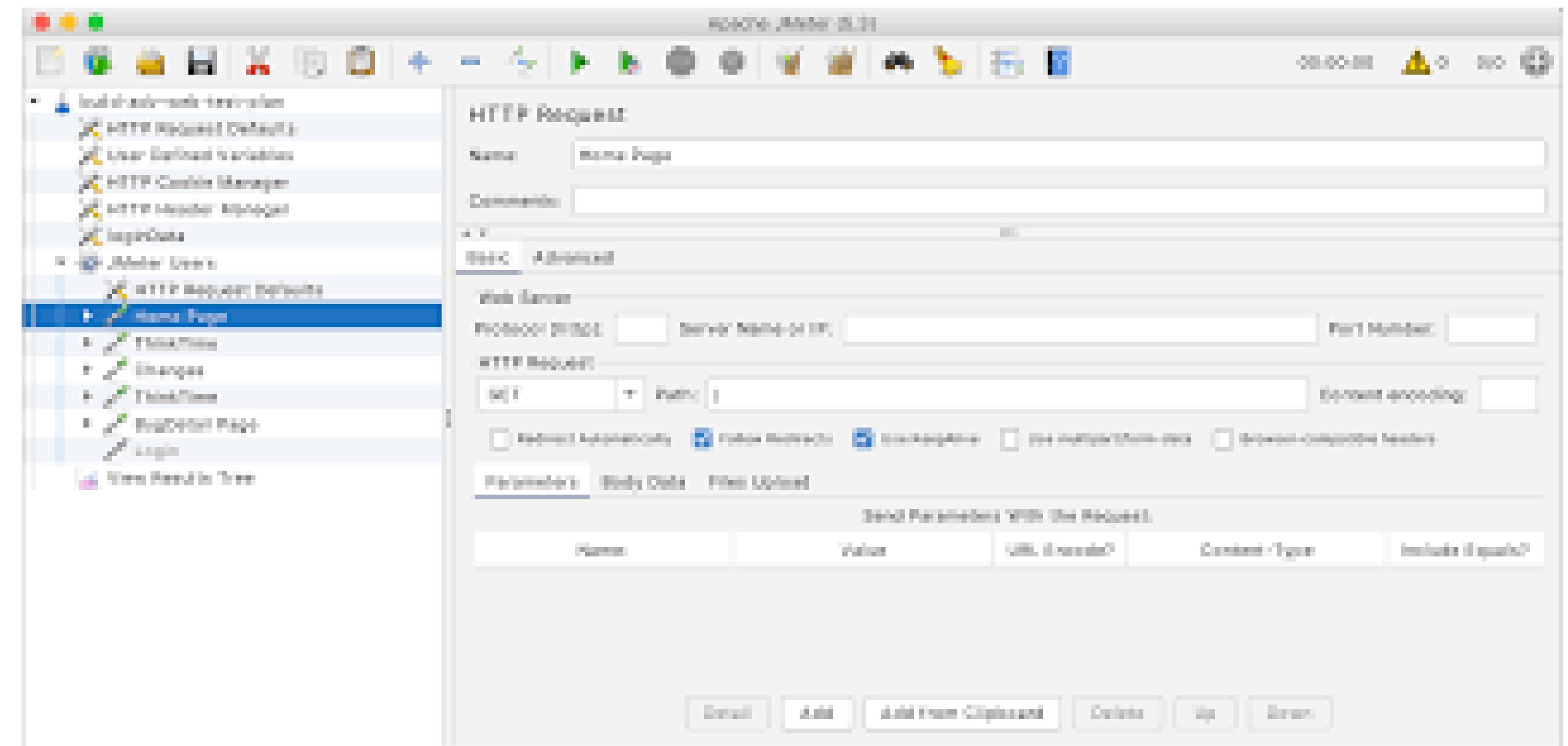
Script básico de prueba:

```
config:  
  target: "http://localhost:3000"  
  
phases:  
  - duration: 60  
    arrivalRate: 10  
  
scenarios:  
  - flow:  
    - get:  
      url: "/api/usuarios"
```

Comando ejecucion: *artillery run test.yml*



- Interfaz gráfica, ideal para principiantes.
- Requiere Java.
- Muy configurable.





- Moderno y orientado a desarrolladores.
- Scripting con JavaScript.
- Comando básico: *k6 run script.js*

```
import http from 'k6/http'

export default function(){
    let url = "https://httpbin.test.k6.io/post";
    let response = http.post(url, "primer prueba")
}
```