


Norma - Pruebas de rendimiento automáticas


Oficina de Calidad

Versión: 1.0.0

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD
		SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD

Control de versiones

Versión	Fecha	Autor	Descripción / Comentarios
V00_v01	28-09-2018	Oficina de Calidad	Versión Inicial
V00_v02	22-11-2018	Oficina de Calidad	A revisión
V00_v03	27-11-2018	Oficina de Calidad	Revisado
V00_v03	29-11-2018	Oficina de Calidad	Publicación. Se añade campos para los requisitos de rendimiento. Cambio de título.
V00_v04	18/03/2020	Oficina de Calidad	Se incluye la referencia a la normativa Maven bajo el proyecto unificación metodología
0.5.0	15/02/2021	Oficina de Calidad	Se actualizan las referencias del apartado "DOCUMENTOS RELACIONADOS"
1.0.0	26/02/2021	Oficina de Calidad	Unificación documento


 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD
		SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD

Índice

1.	INTRODUCCIÓN	4
1.1.	OBJETO	4
1.2.	ALCANCE	4
1.3.	RESOLUCIÓN DE DUDAS	4
1.4.	GLOSARIO	4
1.5.	DOCUMENTOS RELACIONADOS	4
2.	DESCRIPCIÓN GENERAL DE LA PLATAFORMA DE PRUEBAS DEL MIR	5
3.	JMETER	6
4.	PROCESO DE AUTOMATIZACIÓN	6
5.	PROCEDIMIENTO DE PRUEBAS NO FUNCIONALES AUTOMÁTICAS	7
5.1.	ANÁLISIS DE PRUEBAS	7
5.1.1.	Criterios de certificación de pruebas de rendimiento	7
5.1.2.	Requisitos de rendimiento	7
5.2.	PRUEBAS DE CARGA	8
5.3.	CONSTRUCCIÓN DEL PLAN DE PRUEBAS AUTOMÁTICAS	8
5.4.	EJECUCIÓN DE PRUEBAS AUTOMÁTICAS	10
5.5.	RESULTADO DE LAS PRUEBAS AUTOMÁTICAS	11
6.	FASE DE VALIDACIÓN DE LAS PRUEBAS	11
7.	HERRAMIENTAS	11
7.1.	JMETER	11

ÍNDICE DE ILUSTRACIONES

ILUSTRACIÓN 1. ARQUITECTURA ENTORNO DE PRUEBAS NO FUNCIONALES	5
ILUSTRACIÓN 2. ESTRUCTURA JMETER	6
ILUSTRACIÓN 6. BUILD.GRADLE DE SIAVT	9

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD
--	---	---

1. INTRODUCCIÓN

1.1. OBJETO

La normativa detallada en el presente documento es de obligado cumplimiento para todas aquellas personas, equipos y empresas responsables de la definición, implementación, ejecución y evaluación de resultados de los Planes de Prueba requeridos dentro del marco de aseguramiento de la calidad de los sistemas de información desarrollados para la SGSICS

El objetivo del documento es definir unas directrices a seguir por los distintos equipos (desarrolladores, pruebas, jefes de proyecto) para el mejor funcionamiento del proceso de desarrollo del software y pruebas automatizadas de los diferentes tipos de pruebas (funcionales, no funcionales).

En este documento se definirá una normativa en relación a la creación de los scripts de las pruebas no funcionales, en concreto con la herramienta jMeter.

1.2. ALCANCE

En este documento define la normativa **de obligado cumplimiento** para todos los distintos equipos (desarrolladores, pruebas, jefes de proyecto) que vayan a realizar un proceso de pruebas de rendimiento sobre aplicaciones en la MIR (Ministerio del Interior).

1.3. RESOLUCIÓN DE DUDAS


Para cualquier duda se debe realizar una consulta a la oficina de Calidad del MIR a través del correo electrónico sgsics.calidadsw@interior.es con la etiqueta "[JMETER]".

1.4. GLOSARIO

Término	Descripción
SGSICS	Subdirección General de Sistemas de Información y Comunicaciones para la Seguridad
MIR	Ministerio del Interior
SUT	System Under Test (Sistema bajo prueba)

1.5. DOCUMENTOS RELACIONADOS

/Ubicación/Documento	Descripción
PublicacionIntranet/22-PRUEBAS/MIR-INT-CALIDAD-PLAN_PRUEBAS.docx	Plan de pruebas, definición de la metodología a seguir para la definición de los Planes de Pruebas de los Sistemas de Información desarrollados para la SGSICS en el marco de la Gestión de la Calidad del software.
PublicacionIntranet/plantillas/doc/01.A	Entregable de Especificaciones

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD
		SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD

SI/MIR-PROY-ASI- ESPECIFICACIONES.docx	
PublicacionIntranet/plantillas/doc/04.IA S/MIR-APLIC-IAS- INFORME_PRUEBAS.docx	Entregable del informe de pruebas

2. DESCRIPCIÓN GENERAL DE LA PLATAFORMA DE PRUEBAS DEL MIR

En el siguiente diagrama se muestra la arquitectura de la plataforma de Integración Continua sobre la que se añadido los componentes necesarios para la realización de pruebas no funcionales (Rendimiento).

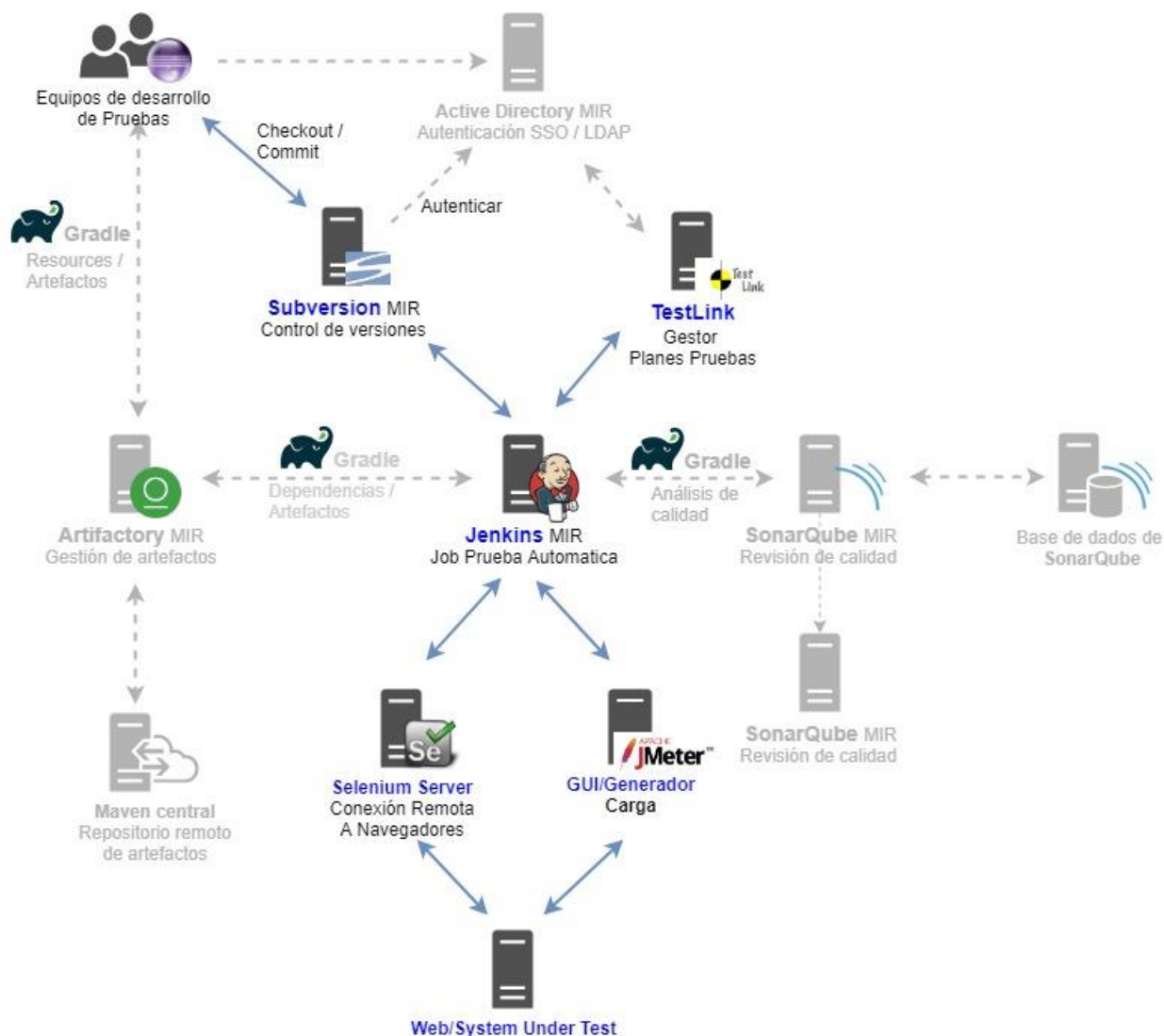



Ilustración 1. Arquitectura entorno de Pruebas No Funcionales

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD
--	---	---

Actualmente el sistema de pruebas no funcionales (rendimiento) consta de los siguientes elementos:

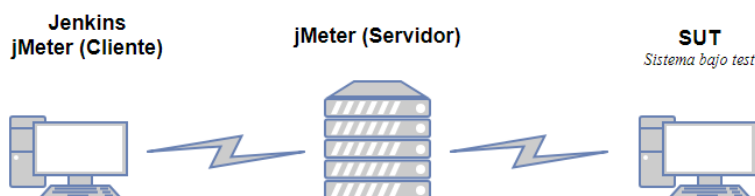


Ilustración 2. Estructura jMeter

3. JMETER

JMeter es un proyecto apache es un software open source (Java), diseñado para ser usado como una herramienta de testing, para la realización de pruebas de carga (load test) para analizar y medir el comportamiento de una serie de servicios, con especial atención a los servicios web.

JMeter es la herramienta de código abierto elegida por el MIR que permite el diseño y ejecución de las pruebas de carga de comportamiento funcional y medición de rendimiento

Aparte de su función de generación de carga, incorpora otras funciones como listeners (ver los distintos resultados), aserciones, tipos de peticiones (http, ftp, LDAP, Bases de datos, etc). Por cada petición que se usa, recopilan una serie de datos.

La herramienta tiene tres modos de empleo, una con el interfaz gráfico (GUI) y a través de la línea de comando.

- Interfaz gráfico (GUI) se usa para diseñar los Testplan, de la cual se pueden añadir los diversos elementos gráficos que tiene disponible la herramienta, una vez diseñado/creado el tesplan se podrá ejecutar y ver los resultados obtenidos.
- Línea de comando, una vez diseñado el tesplan, este se puede almacenar en un fichero .jmx, del cual se puede lanzar sin usar el interfaz gráfico, evitando así tener que arrancar el modo gráfico. Este es el modo que se debe usar la herramienta, y sobre todo para pruebas automáticas.

Servidor, se usa para pruebas distribuidas. Estas pruebas funcionan como modelo cliente-servidor. Jmeter funciona en modo servidor en un servidor y en modo cliente en otro computador cliente, usando el interfaz gráfico (GUI), por ejemplo.

La herramienta proporciona los elementos para:


- Diseñar un testplan (interfaz gráfico), y salvarlo a un fichero .jmx.
- Ejecutar un testplan (interfaz gráfico o línea de comando).
- Visualizar los diferentes resultados obtenidos con la ejecución de la prueba (listeners).

4. PROCESO DE AUTOMATIZACIÓN

Para el proceso de automatización de las pruebas No funcionales, en nuestro caso, pruebas de rendimiento usaremos la aplicación JMeter.

Como se ha comentado anteriormente esta herramienta nos crea un script que será usado para la realización de las pruebas de rendimiento.

La [plataforma de pruebas automáticas](#) del MIR facilita la ejecución de las pruebas No funcionales en la cual se puede realizar:

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD
--	---	---

- La ejecución de las pruebas de forma desatendida, se encarga la herramienta Jenkins, la cual puede lanzar las pruebas a cualquier hora, en días no laborables y por la noche para al día siguiente tener los resultados de la prueba, por ejemplo.
- Ejecutar las pruebas un número mayor de pruebas, si se ha automatizado un Caso de Uso o un Caso de Prueba, éste se puede ejecutar varias veces con distintos datos de prueba o parámetros (rampup, nº de usuarios, etc).
- Pruebas de tendencia de rendimiento son pruebas de rendimiento sobre una aplicación que se está realizando modificaciones, evolutivos, corrigiendo por algún defecto, refactorizando, etc, y que su finalidad es comprobar que la aplicación no se está degradando comparando con pruebas de rendimiento anteriores.

5. **PROCEDIMIENTO DE PRUEBAS NO FUNCIONALES AUTOMÁTICAS**

5.1. **ANÁLISIS DE PRUEBAS**

Esta actividad consiste en realizar un análisis de la fase de pruebas no funcionales automáticas de una aplicación con el fin de conocer los diferentes escenarios de carga que se deberán simular y cuyo propósito es verificar el comportamiento del sistema bajo diferentes escenarios de carga de trabajo.

Las pruebas rendimiento igual que las pruebas funcionales, son obligatorias, en la cual tendrá más o menos número de acciones que serán valoradas por el responsable funcional.

5.1.1. **Criterios de certificación de pruebas de rendimiento**

Las pruebas de rendimiento se harán necesarias con el criterio del responsable funcional, siempre que el sistema presente alguna de las siguientes características:

- Valoración subjetiva: considera adecuada la ejecución de pruebas de rendimiento
- El volumen de usuarios sea elevado
- El sistema implemente funcionalidades de alta demanda.
- El sistema implemente funcionalidades de alta demanda de información
- El sistema implemente funcionalidades costosas
- El sistema tenga de proporcionar información en un tiempo de respuesta reducido.
- El sistema tenga que almacenar una cantidad significativa de información en un tiempo reducido.
- El sistema implemente servicios que han de ser consumidos por otros sistemas


5.1.2. **Requisitos de rendimiento**

Los requisitos de rendimiento son especificaciones relacionadas con la carga que se espera tenga que soportar el sistema en términos medibles.

Por ejemplo, el número esperado de usuarios simultáneamente conectados, número de transacciones por segundo que deberá soportar el sistema, el 95 % de las transacciones deben realizarse en menos de 1 segundo, el uso de disco duro, CPU, memoria, etc. Además, se especificarán los requisitos de datos, si afectan a la información insertada en la base de datos, si fuera necesario, como frecuencia de uso, cantidad de registros que se espera almacenar, etc.

Los requisitos de rendimiento deberán ser identificados a través de los siguientes atributos:

- Nombre
- Descripción

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD
--	---	---

- Versión de la aplicación
- Tiempo medio de respuesta
- Tiempo máximo de respuesta
- Número de usuarios
- Tecnología. Web o web services
- Ritmo transaccional
- Trazabilidad con requisitos funcionales

5.2. PRUEBAS DE CARGA

El objetivo de las **pruebas de carga** es verificar que el sistema es capaz de soportar el volumen de peticiones (carga) esperado (requerido) sin presentar degradación en el rendimiento del mismo.

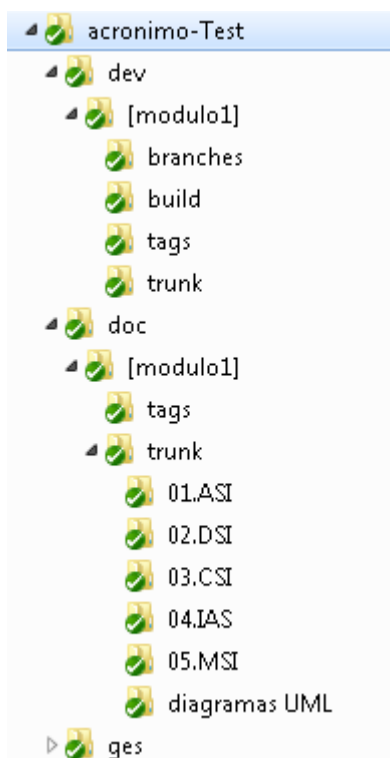
Esta carga puede ser el número esperado de usuarios concurrentes que utilizarán la aplicación, realizando un número específico de transacciones durante un período de tiempo establecido.

5.3. CONSTRUCCIÓN DEL PLAN DE PRUEBAS AUTOMÁTICAS


Después de explicar en los apartados anteriores la línea que debe seguir, en esta fase, el desarrollador de las pruebas automáticas debe comenzar a desarrollar los casos de prueba, teniendo en cuenta lo siguiente:

En la herramienta de control de versiones, en este caso subversión, se usará la siguiente estructura en la cual se almacenará el código de la prueba. Si no existe la estructura se tendrá que solicitar a Calidad de Software (sgsics.calidadsw@interior.es).

La estructura será la siguiente:



El código de la prueba debe ir en:

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD
--	---	---

<acrónimo>-Test/dev/[modulo1]/trunk/src/jmeter

Para la documentación del código, si es necesaria se deberá ir en:

<acrónimo>-Test/doc/[modulo1]/trunk

Acrónimo corresponde al valor del “acrónimo” que existe en el fichero build.gradle de la aplicación a probar.

Por ejemplo, el acrónimo de la aplicación de SIAVT en el build.gradle.

```

plugins {
    id 'project-report'
    id 'org.sonarqube' version '2.5'
    id "jacoco"
}

defaultTasks 'clean', 'build'

//Información global de proyecto
ext {
    projectInfo = [
        description : ""Sistema integral de ayudas a victimas del terrorismo"",
        organization : 'Ministerio del Interior',
        developmentVendor : 'RICOH',
        acronimo : 'siavt',
        group : "es.mir.${rootProject.name}",
        name : "${rootProject.name}",
        version : '1.32.8',
        encoding : 'UTF-8',
        sourceJdkVersion : '1.7',
        targetJdkVersion : '1.7'
    ]
}

apply from: "gradle-env-conf/environmentBuildConfig.gradle"

```

Ilustración 3. Build.gradle de SIAVT


Cuando se realice una entrega de las pruebas automáticas, esta se realizará a través de la generación de una etiqueta <tag>. El nombre de la etiqueta corresponde a la versión del código, explicado más abajo.

La versión del código debe seguir lo que se indica en el apartado de “Versionado Semántico” de la normativa Gradle / Maven y será igual que para el código de la aplicación.

Teniendo en cuenta, que si la aplicación a probar corresponde siavt-3.1.0.

La versión del código de prueba correspondería a “rendimiento-siavt-3.yy.zz”

- Nombre del proyecto. Corresponde al <rootProject.name>, en el ejemplo “siavt”.
- Dada una versión MAJOR.MINOR.PATCH
 - MAJOR. Debe coincidir con la versión del software a probar, en el ejemplo “3”.
 - MINOR. “yy”, como se indica en la normativa Gradle / Maven (Versionado Semántico).
 - PATCH. “zz”, como se indica en la normativa Gradle / Maven (Versionado Semántico).
- Prefijos: Tipos de prueba, nivel global “rendimiento”, “carga”, “estabilidad”.

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD
--	---	---

En la construcción se ultimará el plan de pruebas de rendimiento completando la información a los casos de pruebas (ficheros de datos de entrada, secuencia de pasos a seguir, definición de escenarios de prueba etc.)

A continuación, se detallan los productos que deberá proporcionar el desarrollador como resultado de la construcción de las pruebas de rendimiento:

- TestPlan de JMeter (fichero .jmx) que contenga los casos de prueba definidos en base a los requisitos del sistema y niveles de criticidad.
- Especificación de las propiedades de configuración del TestPlan. Los TestPlans han de ser configurables para asegurar que estos puedan ejecutarse en varios entornos de pruebas modificando únicamente la configuración. Algunos valores a parametrizarán serán:
 - El hostname/dirección IP del servidor contra el que se ejecuta el testplan
 - Credenciales de los usuarios
 - Strings / expresiones regulares que se comprueban en las aserciones
 - Datos de configuración de los grupos de hilos (thread groups). Esto es, número de hilos (threads), período de inicialización o lanzamiento (ramp-up period), número de iteraciones o tiempo de la ejecución, etc.
- No es necesario implementar el fichero de propiedades (.properties), pero si fuera necesario modificar algún parámetro se debe comunicar al equipo de calidad, y este proporcionará el fichero que está actualmente configurado, para que se realice las modificaciones oportunas por parte del equipo de pruebas y se pasará a JMeter en tiempo de ejecución mediante el modificador -q. Estos ficheros de propiedades deben auto contener información de versión y fechas de última modificación, así como comentarios que proporcionen información relativa al significado de cada propiedad y posibles valores de la misma.
- Ficheros de datos utilizados para la ejecución de las pruebas
 - Ficheros CSV con los registros necesarios para la carga dinámica de datos. El volumen y la variedad de datos deberá ser el adecuado para que la ejecución de pruebas se asemeje lo máximo posible a la realidad.

Estos elementos serán almacenados en el repositorio de versiones junto con el resto de código fuente.

El ejemplo:

```


<acrónimo>-Test/dev/[modulo1]/trunk/src/jmeter/rendimiento-siavt-3.1.0.jmx
<acrónimo>-Test/dev/[modulo1]/trunk/src/jmeter/usuariosPassword.csv
<acrónimo>-Test/dev/[modulo1]/trunk/src/jmeter/rendimiento-siavt-3.1.0.properties

```

5.4. EJECUCIÓN DE PRUEBAS AUTOMÁTICAS

En esta fase se realiza la ejecución de las pruebas automáticas, aunque no excluye que se realicen pruebas de rendimiento manuales, pero no es el documento adecuado.

Importante: Las pruebas de rendimiento se realizan una vez que han pasado las pruebas funcionales, cuyo resultado sea correcto y la validación del responsable funcional.

 MINISTERIO DEL INTERIOR	Norma - Pruebas de rendimiento automáticas	SECRETARÍA DE ESTADO DE SEGURIDAD SUBDIRECCIÓN GENERAL DE SISTEMAS DE INFORMACIÓN Y COMUNICACIONES PARA LA SEGURIDAD
--	---	---

Una vez que se ha creado el script (.jmx) y los ficheros auxiliares que usa el script, se solicita al equipo de calidad la creación del job en Jenkins.

5.5. RESULTADO DE LAS PRUEBAS AUTOMÁTICAS

El resultado de las pruebas se debe registrarse en TestLink. El registro de dichos ítems podrá llevarse a cabo de forma manual.

6. FASE DE VALIDACIÓN DE LAS PRUEBAS

Una vez finalizada la prueba y reflejados los resultados en la herramienta de TestLink, el equipo de pruebas de rendimiento con los requisitos o/y el responsable funcional deben realizar la validación de las pruebas.

Con los siguientes criterios:

Los tiempos de respuesta serán:

- Para peticiones web
 - Tiempo Máximo de Respuesta: 5 seg.
 - Tiempo Medio de Respuesta: 3 seg.
- Para respuesta de web services
 - Tiempo Máximo de Respuesta: 4 seg.
 - Tiempo Medio de Respuesta: 2 seg.

En caso que exista alguna justificación técnica por la que la aplicación deba exceder ese tiempo, será el responsable funcional el que tome la decisión de pasar o no a producción. Además, el responsable funcional indicará si se deben repetir las pruebas o realizar más pruebas de rendimiento para determinar la causa del exceso de ese tiempo.

7. HERRAMIENTAS

7.1. JMeter

JMeter versión: 4.0