

1,2,3 probando probando



Ane Undagoitia // Responsable Funcional Manual
ane.undagoitia@globetesting.com



ARABATIC

EL SOFTWARE ESTÁ EN TODOS LADOS



INTELIGENCIA ARTIFICIAL



INTERNET DE LA INTELIGENCIA



BMW llama a revisión a 12 mil unidades Serie 7

Publicado por [José Antonio Martínez](#) el 28 de febrero de 2018 a las 19:30.

f Comparte

t Twitea

g+

p

e



El accidente de un Tesla Model X provocó la muerte del conductor y una nueva investigación sobre la conducción autónoma

f 1330

t

e

¿Y si los daños cuestan millones o son daños personales?



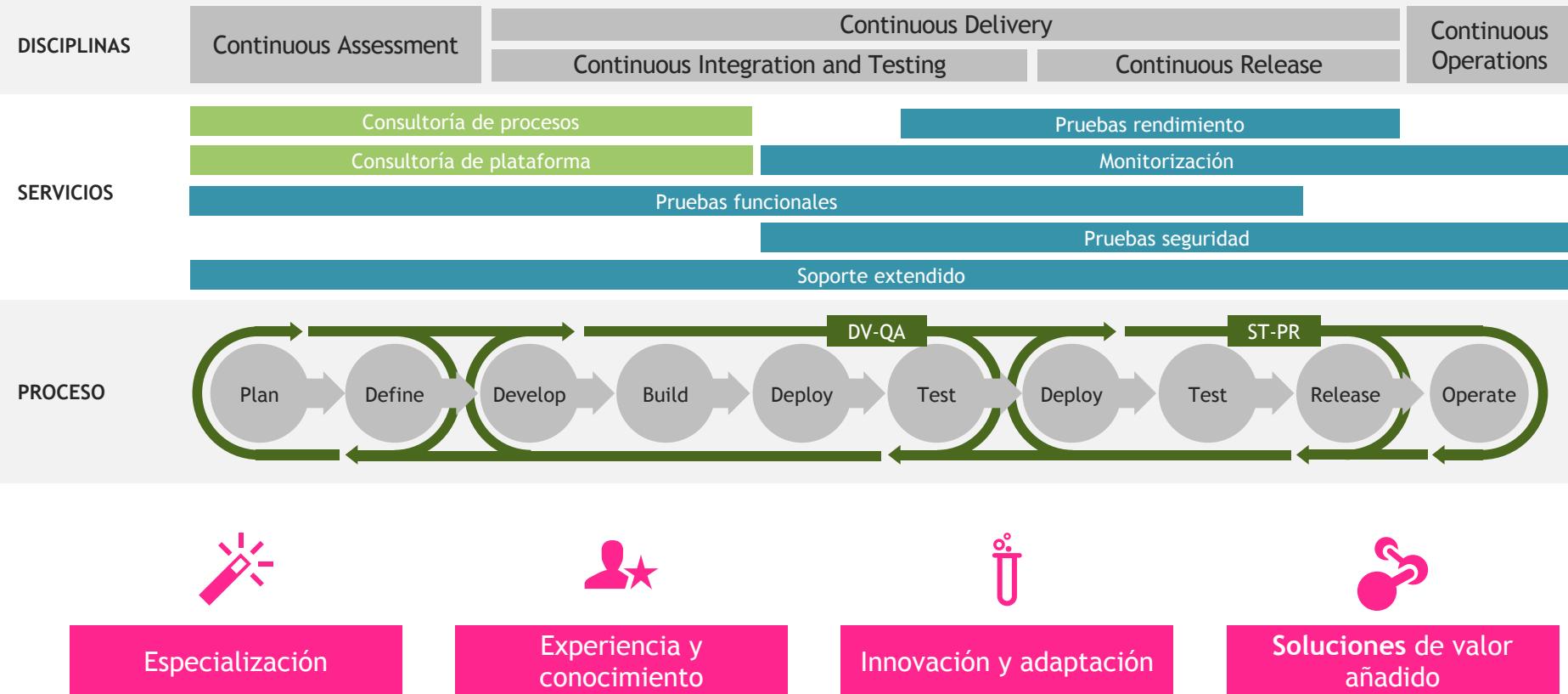
Software testing DONE RIGHT



A vibrant comic book illustration of the Justice League. In the foreground, Wonder Woman stands on the left, her lasso of truth held high. Next to her is Aquaman, his golden scales catching the light. In the center, Superman stands prominently, his iconic red and blue suit and 'S' emblem clearly visible. To his right is Cyborg, his metallic body and mechanical arm a stark contrast to his teammates. Batman stands behind him, his dark cowl and suit partially obscured by shadows. On the far right, The Flash runs towards the viewer, his red suit and lightning bolt emblem glowing. The background is a deep blue space filled with stars, creating a sense of cosmic scale.

TRABAJO EN EQUIPO

Ciclo de vida de un Software





PRUEBAS
FUNCIONALES

PRUEBAS DE
RENDIMIENTO

PRUEBAS DE
SEGURIDAD

MONITORIZACIÓN

Pruebas funcionales

By C. Michael Jones, Ph.D.

Pruebas Funcionales

¿Qué significan las pruebas funcionales?



Pruebas que se realizan a la hora de dar el pase a producción a cualquier tipo de software. Caja negra



Se hacen mediante el diseño de pruebas que buscan evaluar cada una de las opciones con las que cuenta el paquete informático



Pruebas específicas, concretas y exhaustivas para probar y validar que el software hace lo que debe y sobre todo lo que se ha especificado



Permite la corrección de los mismos a costos reducidos al ser encontrados en etapas tempranas

Pruebas Funcionales

¿Qué conseguimos con la realización de las pruebas funcionales?



Asegurar requisitos funcionales

Identificar inconsistentes

Aumentar satisfacción del cliente

Mejorar productividad

Reducir Costo

Ayuda a reforzarlo y a mejorar la calidad del SW



Pruebas Funcionales

¿Cómo se pueden realizar?



Pruebas Manuales



Pruebas Automatizadas

- Para ahorrar tiempo en procedimientos repetitivos.
- Aumentar la fiabilidad de las mismas eliminando el error humano intrínseco de las pruebas funcionales.

Pruebas Funcionales Manuales

¿Qué son las pruebas funcionales manuales?



Prueba realizada por un ser humano



Ejecuta un test como si fuese un usuario pero siguiendo una serie de pasos



El tester realizará las acciones indicadas en cada paso



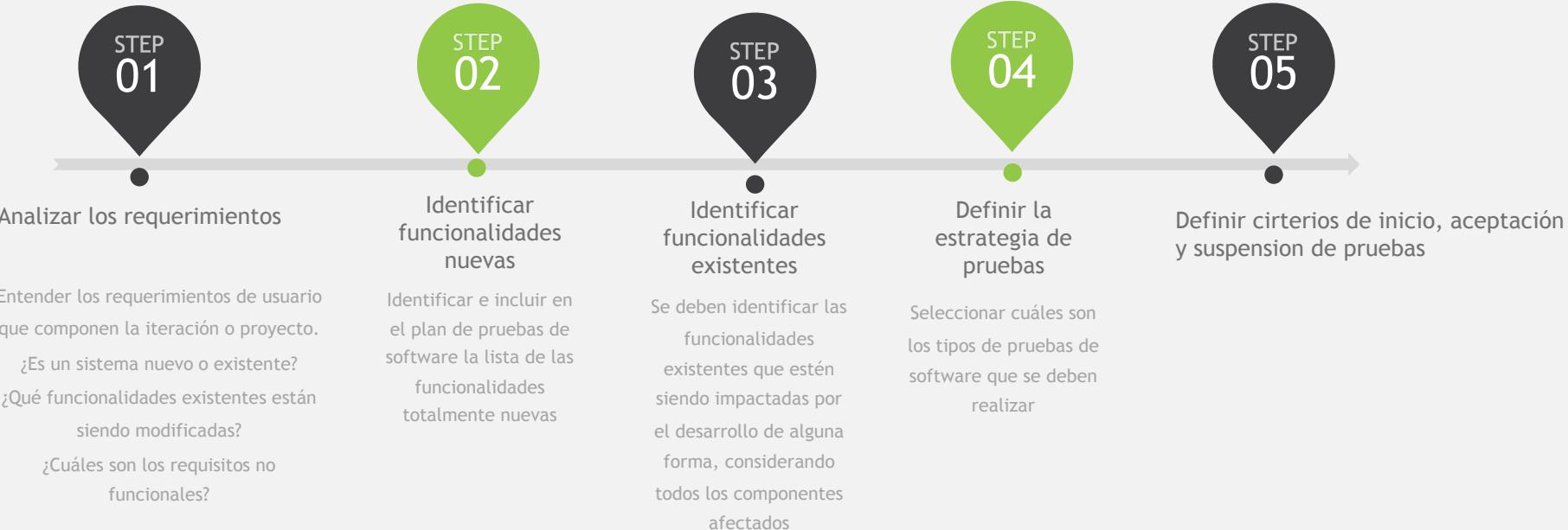
Si el resultado es distinto al esperado, se reportará un defecto con todo detalle: descripción, datos utilizados, capturas de pantalla



Pruebas funcionales manuales



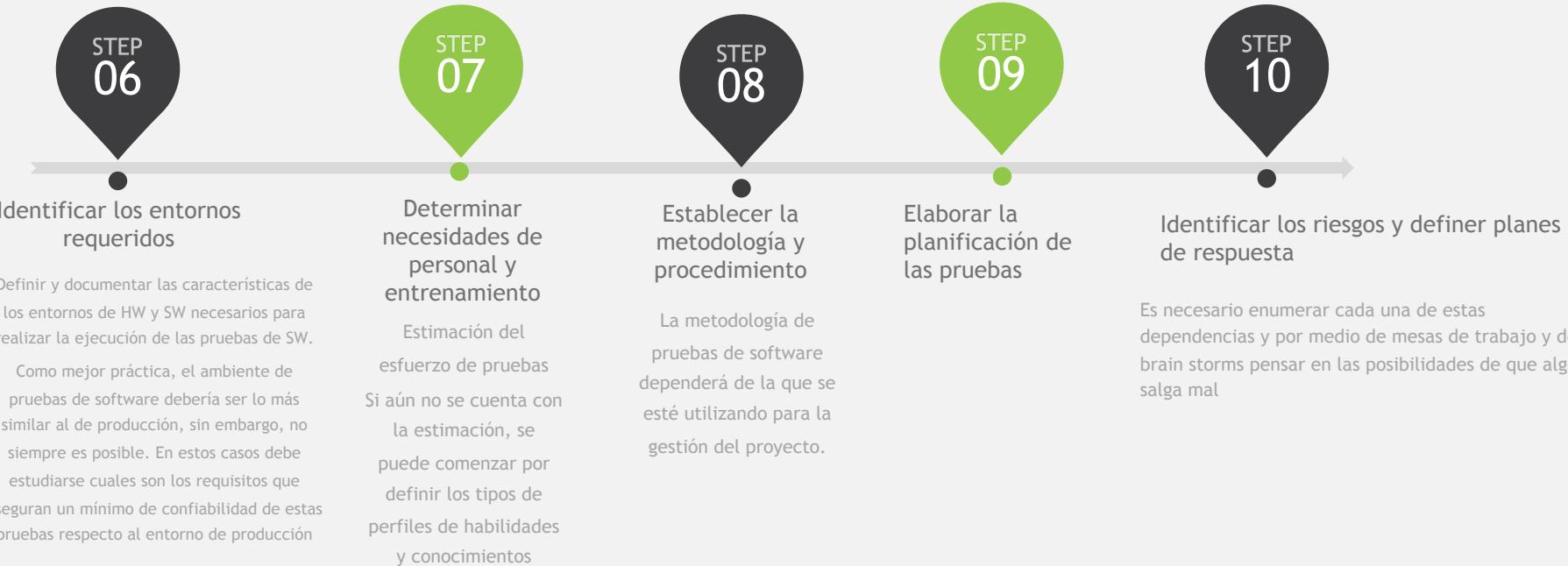
Diseño del plan de pruebas → 10 pasos para elaborar un Plan de pruebas



Pruebas funcionales manuales



Diseño del plan de pruebas → 10 pasos para elaborar un Plan de pruebas



Pruebas Funcionales Manuales

Según tipo de prueba → Pruebas Exploratorias



- Pruebas en las que se obtiene un aprendizaje y conocimiento de la aplicación a probar.
- Ayudan a la integración de la fase de pruebas de una forma mucho más rápida.
- Útiles a la hora de probar aplicaciones ya desarrolladas, es decir, aquellas pruebas de software que no comienzan a la vez que el desarrollo.
- Para realizar las pruebas funcionales exploratorias se identificarán los distintos procesos de negocio o módulos de la aplicación y se le dará al tester libertad, poniéndose en la piel de un usuario, para probarlos.
- Estas pruebas exploratorias deberán ejecutarse sobre la última versión cerrada disponible de la aplicación.



Pruebas Funcionales Manuales

Según tipo de prueba → Pruebas de regresión

- Comprobar qué cambios realizados en el software no introducen un comportamiento no deseado o errores adicionales en otros módulos o partes no modificados.
- Deben llevarse a cabo cada vez que se hace un cambio en el sistema, tanto para corregir un error como para realizar una mejora.
- No es suficiente probar sólo los componentes modificados o añadidos, o las funciones que en ellos se realizan, sino que también es necesario controlar que las modificaciones no produzcan efectos negativos sobre el mismo u otros componentes.
- Garantizar que tras un cambio en el software, al menos la funcionalidad más importante sigue funcionando



Pruebas Funcionales Manuales

Según tipo de prueba → Pruebas de compatibilidad



- Pruebas funcionales realizadas en diferentes entornos como en cada navegador de internet, sistema operativo o dispositivo, para garantizar el correcto funcionamiento del aplicativo en todos los medios.
- El mismo software puede presentar errores dependiendo de dónde se ejecute: funcionales (botones y enlaces pueden dejar de funcionar, producen errores de sistema o simplemente no realizan la funcionalidad esperada), estéticos (pueden descuadrarse frames de la aplicación, no cargarse imágenes, desaparecer enlaces o botones y textos).



Pruebas Funcionales Manuales

Según tipo de prueba → Pruebas de integración



- Pruebas funcionales entre dos o más sistemas.
- El objetivo de las pruebas de integración es verificar el correcto ensamblaje entre los distintos componentes una vez que han sido probados unitariamente con el fin de comprobar que interactúan correctamente a través de sus interfaces, cubren la funcionalidad establecida y se ajustan a los requisitos.



Pruebas Funcionales Manuales

Según tipo de prueba → Pruebas de aceptación

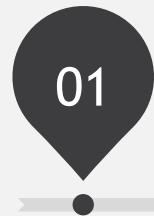


- El objetivo de las pruebas de aceptación es validar que un sistema cumple con el funcionamiento esperado y permitir al usuario de dicho sistema que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento.
- En las pruebas de aceptación, la ejecución y aprobación final corresponden al usuario o cliente, que es el que valida y verifica que el alcance es el correcto.

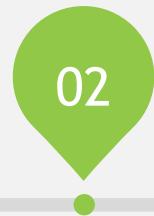


Beneficios de la automatización

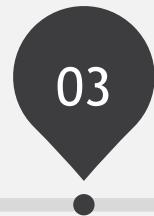
¿Qué nos aportan las pruebas automatizadas?



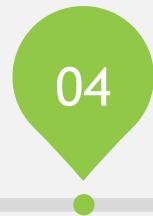
01



02



03



04



05

Rapidez

Las herramientas de testing automatizado corren las pruebas significativamente más rápido que los testers humanos.

Fiabilidad

Las pruebas ejecutan precisamente las mismas operaciones cada vez que se ejecutan, eliminando el error humano

Repetición

Se puede testear como reacciona el software bajo repetidas ejecuciones de las mismas operaciones

Programable

Se pueden programar pruebas sofisticadas y complejas que muestren información oculta de la aplicación

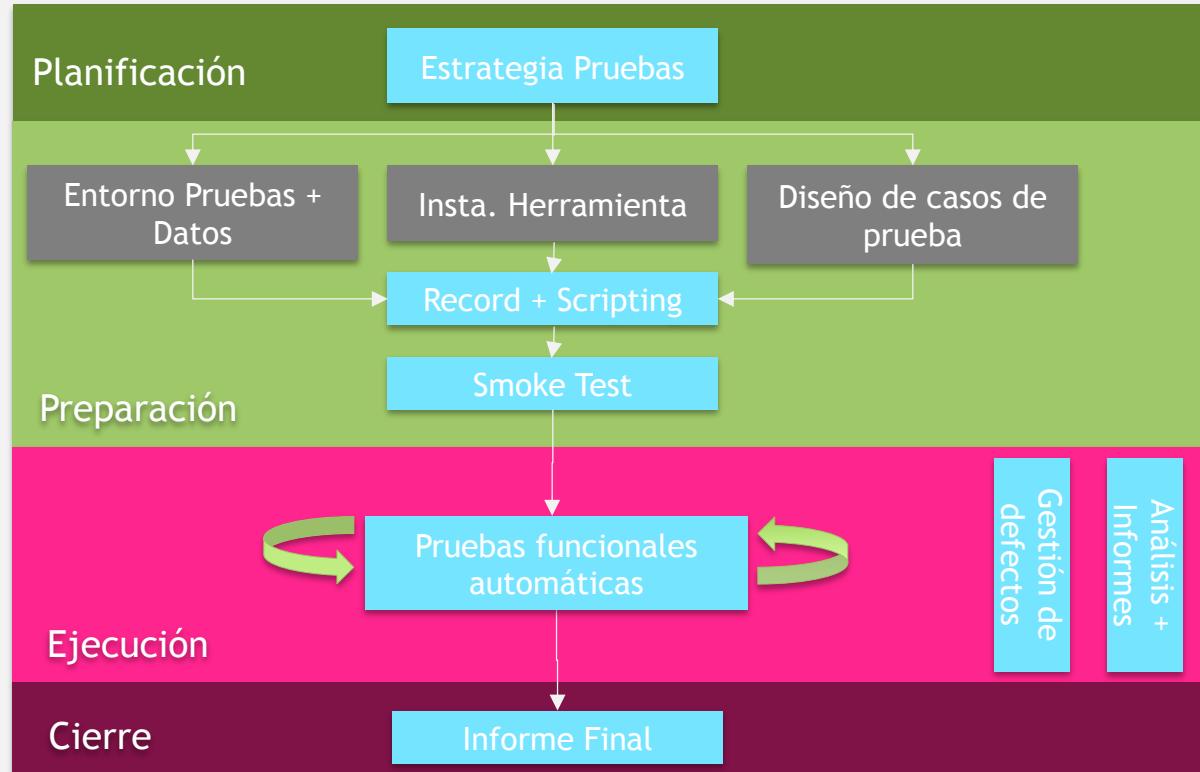
Reusabilidad

Se pueden rehusar los scripts con pruebas automatizadas, las funciones, etc

Automation

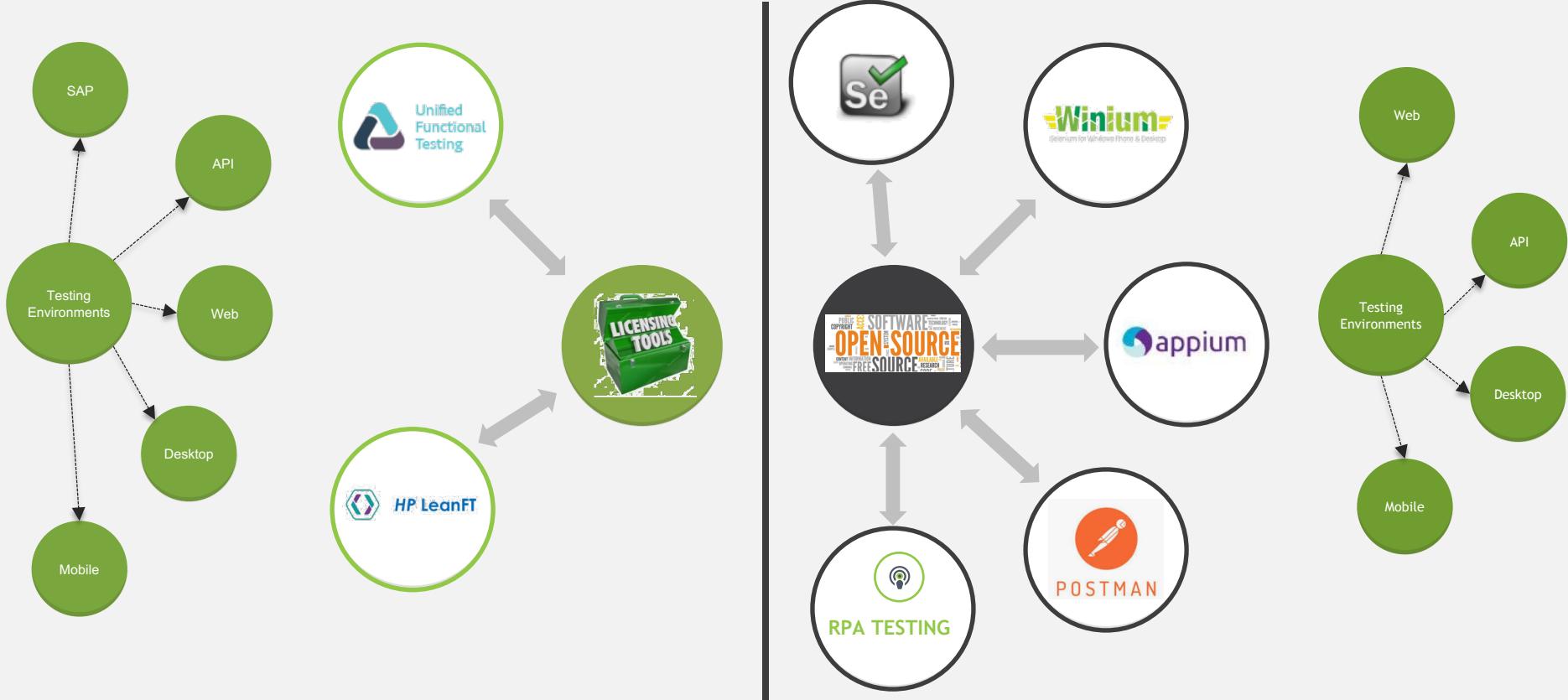
Pruebas funcionales

Proceso

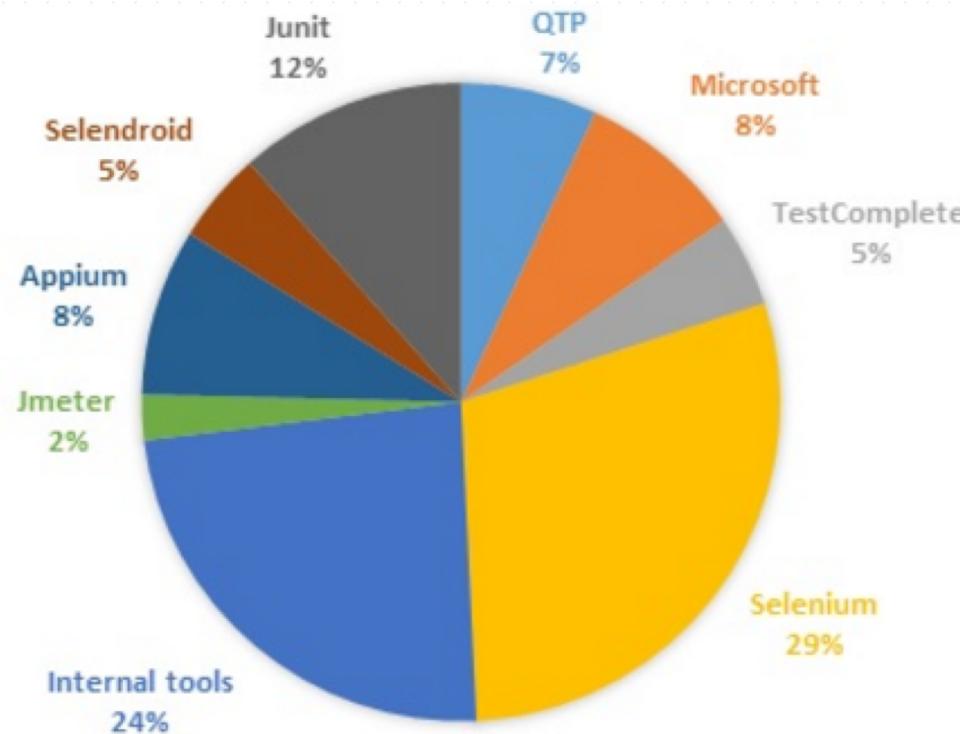


Herramientas automatización

¿Con qué herramientas podríamos desarrollar los scripts?

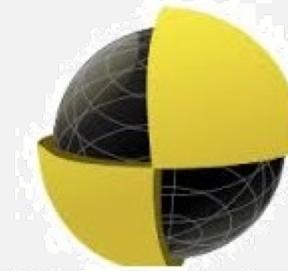


Análisis de la automatización en empresas



Pruebas Funcionales

Herramientas para el ciclo de vida del software



TestLink

TestRail

An Idera, Inc. Company

Pruebas de rendimiento

By C. Michael Jones, Ph.D.

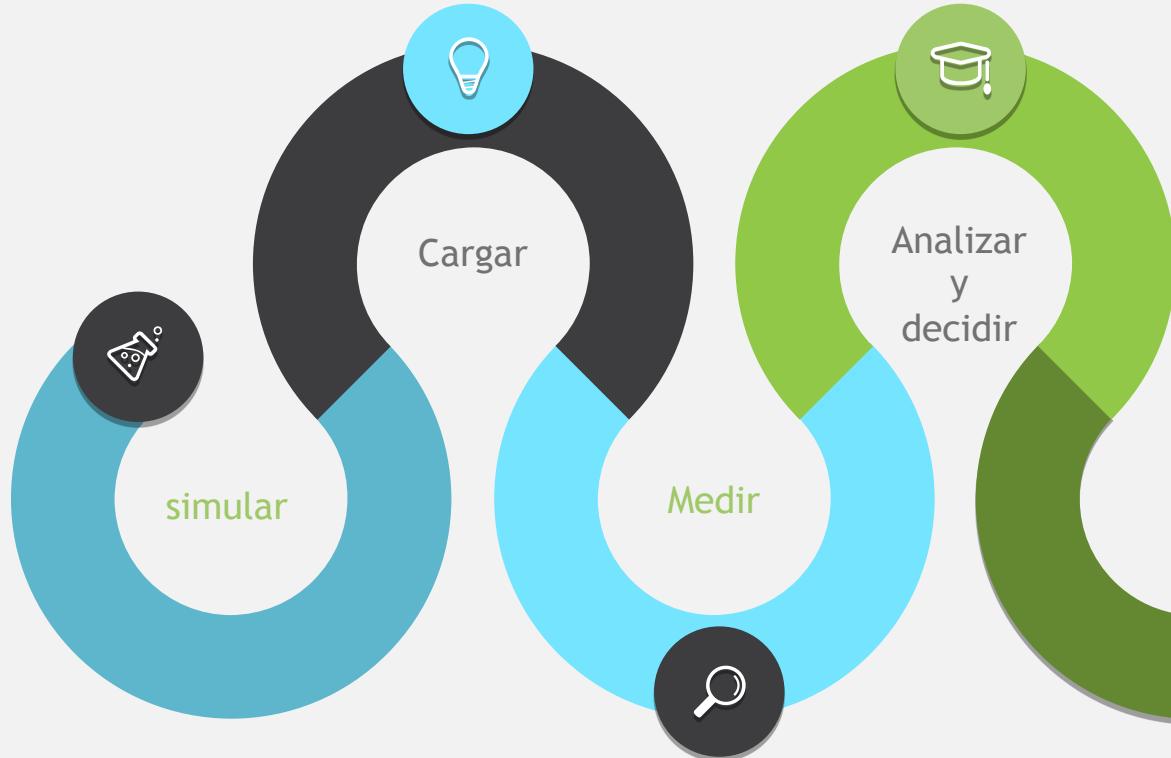


Por qué probar el rendimiento

- Reducir costes mediante la anticipación.
- Entender el comportamiento de la aplicación bajo carga.
- Definir la evolución de la aplicación e infraestructura.

Factores clave

- Definir los criterios de aceptación.
- Diseñar el comportamiento de los usuarios simulando los procesos de negocio.
- Diseñar y entender los escenarios de concurrencia.
- Monitorizar durante la ejecución.

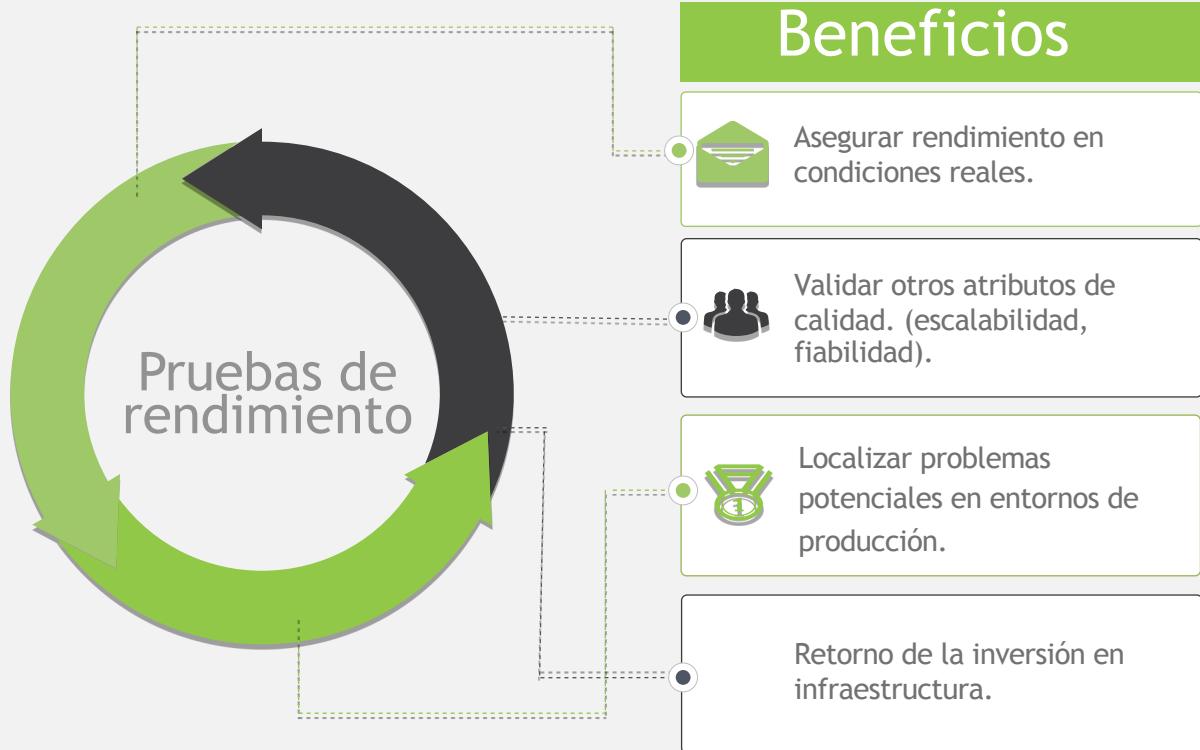




Objetivos

Determinar el comportamiento del sistema bajo ciertas condiciones de carga.

Demostrar que el sistema cumple con los criterios de rendimiento.



Beneficios

Asegurar rendimiento en condiciones reales.

Validar otros atributos de calidad. (escalabilidad, fiabilidad).

Localizar problemas potenciales en entornos de producción.

Retorno de la inversión en infraestructura.

Procedimiento de pruebas





Esfuerzo

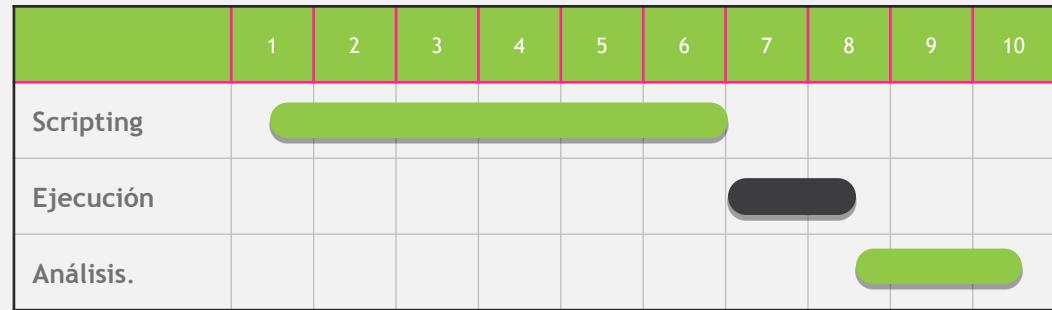
65% Scripting

Factores clave:

- Complejidad de la aplicación.
- Protocolo.
- Estabilidad del entorno.
- Datos de prueba.

15%
Ejecución

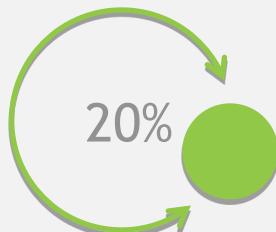
20%
Análisis



Recomendaciones:

- Trabajar conjuntamente con los responsables funcionales de la aplicación.
- Realizar un análisis previo de riesgos y aspectos a tener en cuenta. (juegos de datos, disponibilidad del entorno, limitaciones de red, etc.).
- Definición de los objetivos de la prueba.
- Planificar con tiempo las pruebas antes del pase a producción.
- Apoyo del departamento de sistemas (monitorización y soporte).

Tipos de pruebas de rendimiento

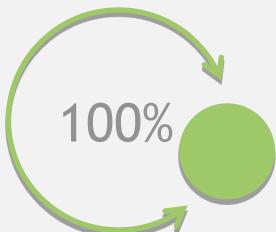


Baseline

Carga: 20%

Tiempo estable: 30min

Objetivos:
Comprobar funcionamiento en concurrencia.
Validar juegos de datos.
Obtener primeros tiempos con baja carga.

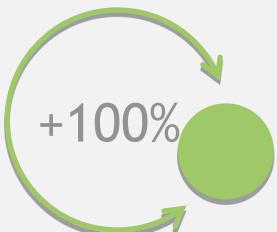


Carga

Carga: 100%

Tiempo estable: 2h

Objetivos:
Cargar el sistema con el 100% de la carga esperada.
Indicadores de calidad.
Identificar cuellos de botella.



Stress

Carga: +100%

Tiempo estable: 3h

Objetivos:
Medir escalabilidad ante el aumento de carga.
Punto de ruptura.
Identificar cuellos de botella bajo mucha carga.



Reliability

Carga: 100%

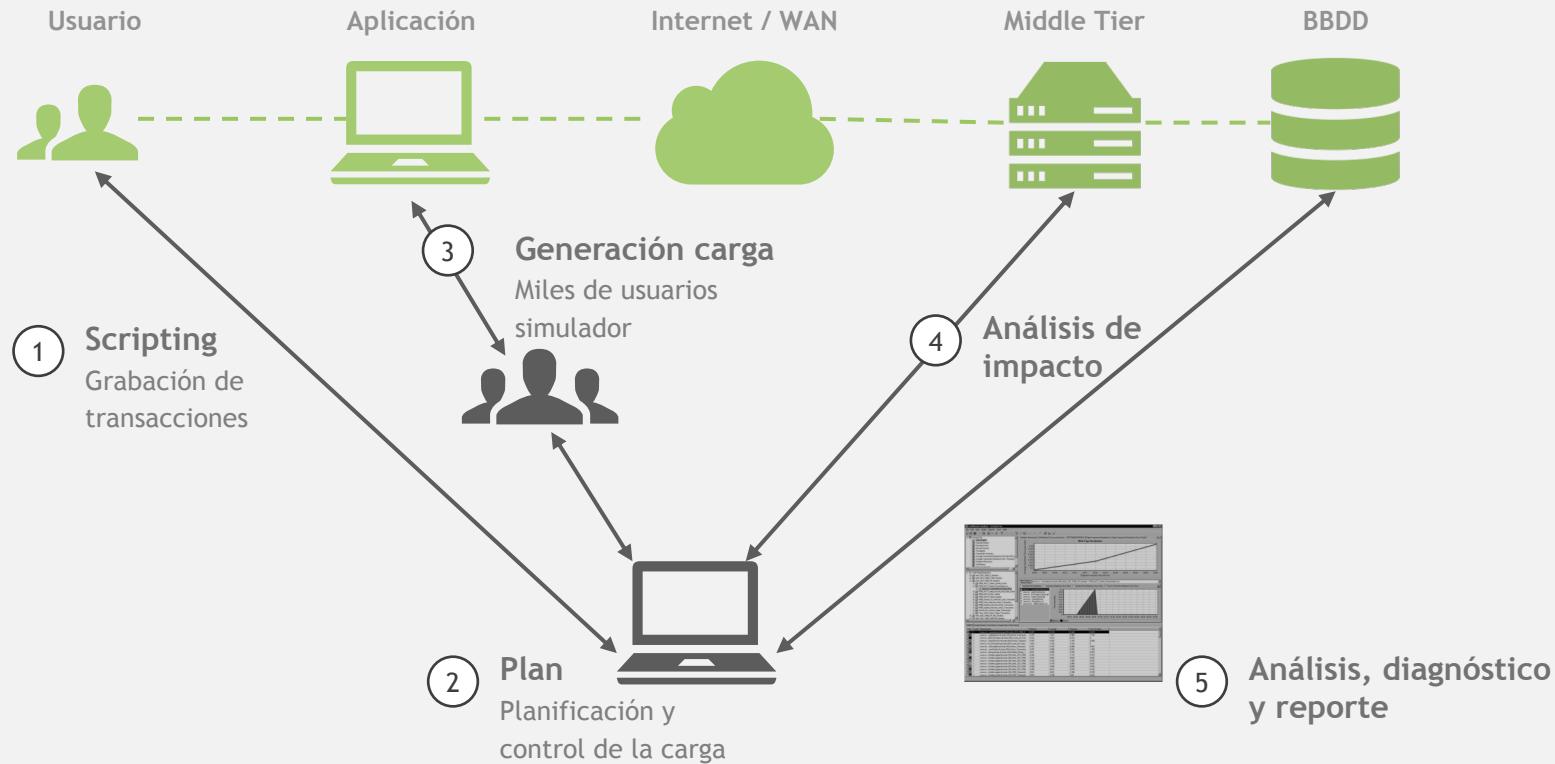
Tiempo estable: 12h

Objetivos:
Comprobar la fiabilidad del sistema en el tiempo.
Identificar degradación en la arquitectura.

Pruebas de rendimiento



Proceso



Pruebas de rendimiento

Microfocus LoadRunner



La solución líder en el mercado para proyectos de rendimiento

Permite de forma consistente y repetible:

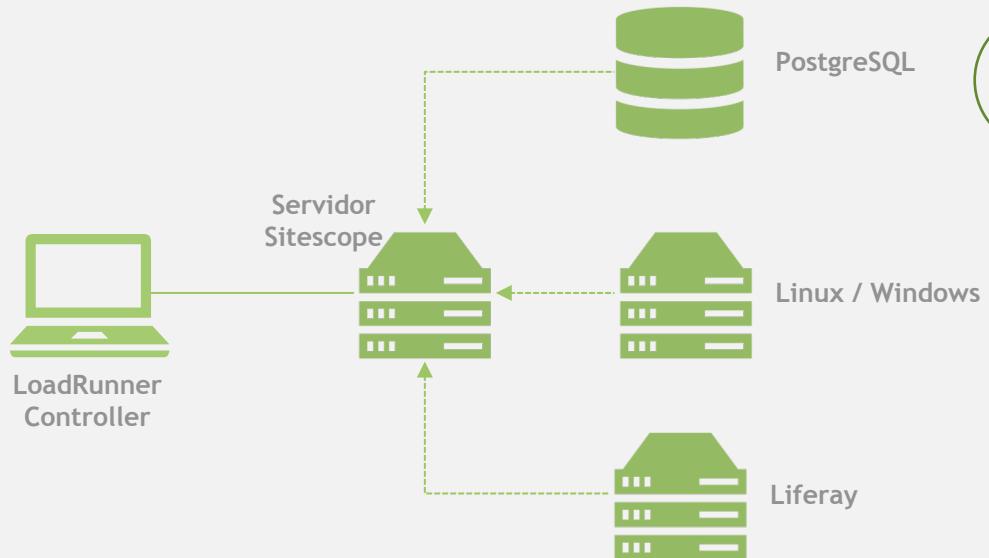
- Emular la carga de producción
- Probar multitud de aplicativos
- Identificar cuellos de botella
- Diagnosticar causa-raíz
- Optimizar rendimiento antes del despliegue



Reduce Riesgo, Coste y Tiempo

Monitorización

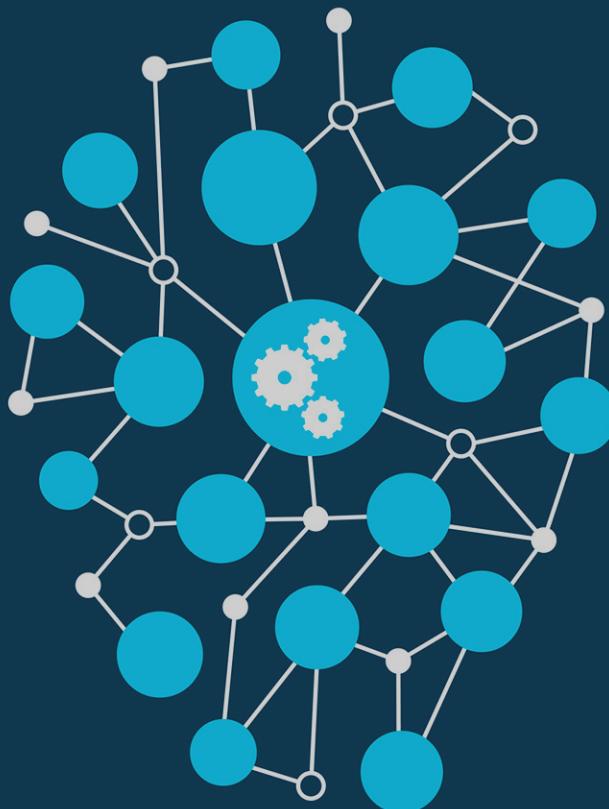
Microfocus SiteScope



Sistema de Monitorización

- Monitorización de sistemas mientras se realiza la prueba de carga
- Integrado con Microfocus LoadRunner
- Monitorización sin agentes
- Gestión web configurable
- Métricas y alertas configurables
- Configuración del monitor desde Microfocus LoadRunner.

MACHINE LEARNING





**TOD@S DEBEMOS
CRECER**
con la tecnología