ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

ECYS | DTT
DESARROLLO DE TRANSFERENCIA TECNOLÓGICA

Día, Fecha:	Miércoles, 18/09/2024
Hora de inicio:	15:40

Análisis y Diseño de Sistemas 2 [B]

Luis Angel Barrera Velásquez

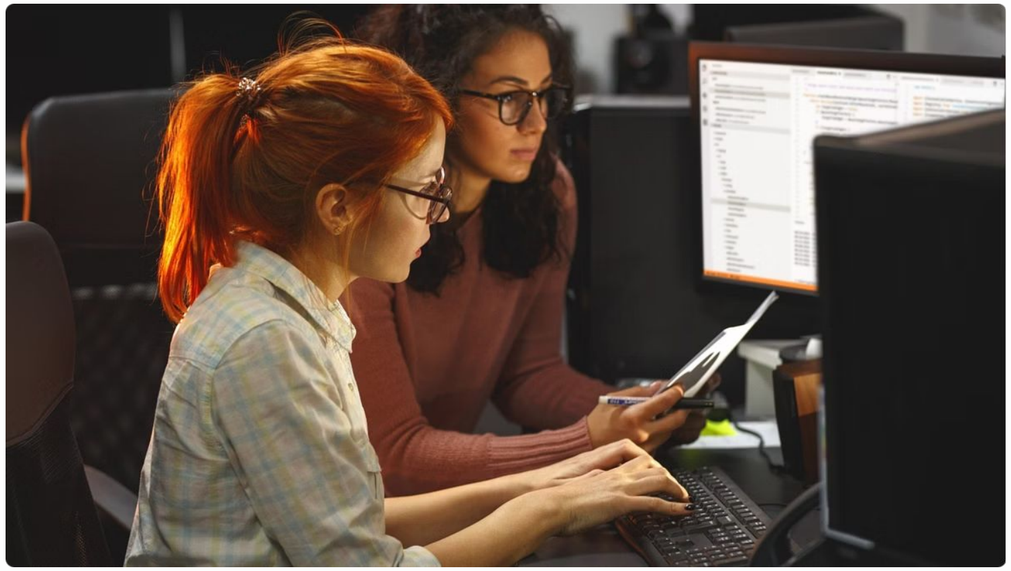
AGENDA

1	Avisos
2	Dudas de fase 2
3	Contenido Teórico Pruebas no Funcionales
4	Kahoot

Pruebas de software

Introducción a las pruebas de software

Las pruebas de software son un proceso fundamental para garantizar la calidad y el correcto funcionamiento de los sistemas informáticos. Permiten identificar y corregir errores antes de que el software se ponga en producción, lo que ahorra tiempo y recursos.



Importancia de las pruebas de software

1

Calidad

Las pruebas ayudan a asegurar que el software cumpla con los requisitos y funcione sin errores.

2

Satisfacción del usuario

Un software bien probado brinda una mejor experiencia de usuario y genera más confianza.

3

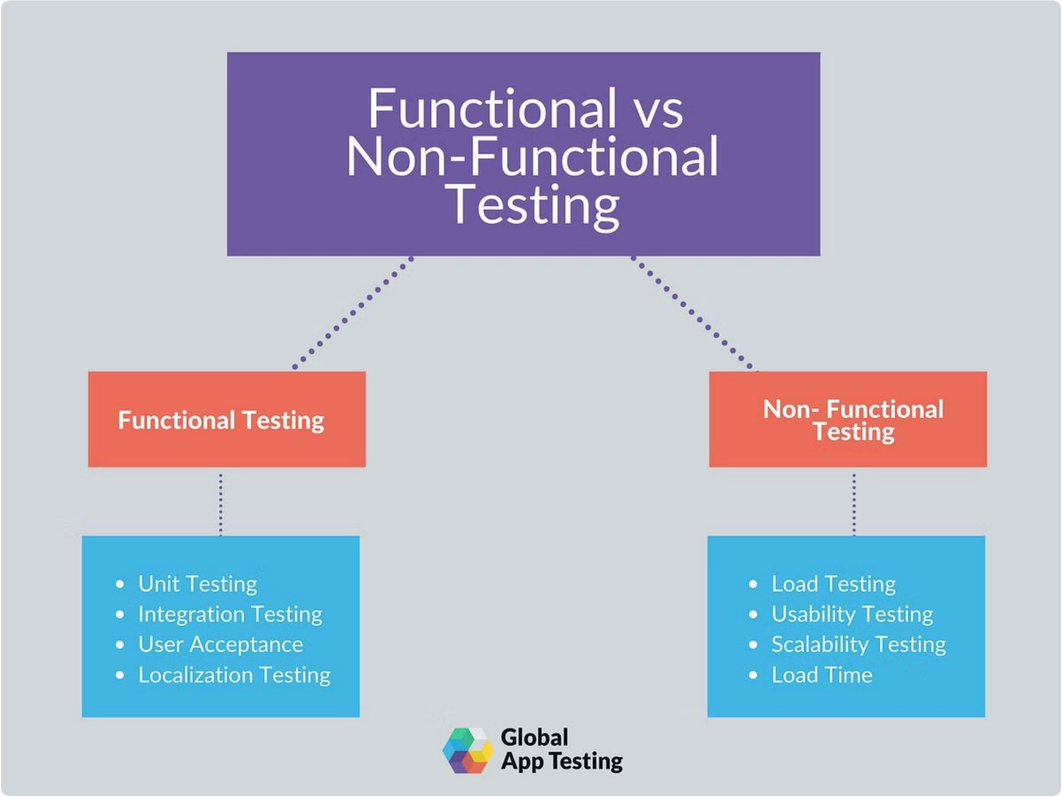
Detección temprana de errores

Identificar y corregir problemas en las primeras etapas reduce costos y esfuerzo..

Tipos de pruebas de software

Pruebas Funcionales

Validan que el software cumpla con los requisitos especificados.



Pruebas No Funcionales

Evalúan aspectos como rendimiento, seguridad, usabilidad y compatibilidad.

Pruebas no funcionales

Las pruebas no funcionales se centran en evaluar las características del sistema que no están relacionadas directamente con las funciones o características específicas que realiza, sino con la calidad del sistema en cuanto a su rendimiento, capacidad de respuesta, escalabilidad, entre otras.

Pruebas no funcionales

Objetivo: Evaluar cómo se comporta el sistema en términos de rendimiento, eficiencia, usabilidad, fiabilidad, y otros aspectos no funcionales bajo diferentes condiciones operativas.

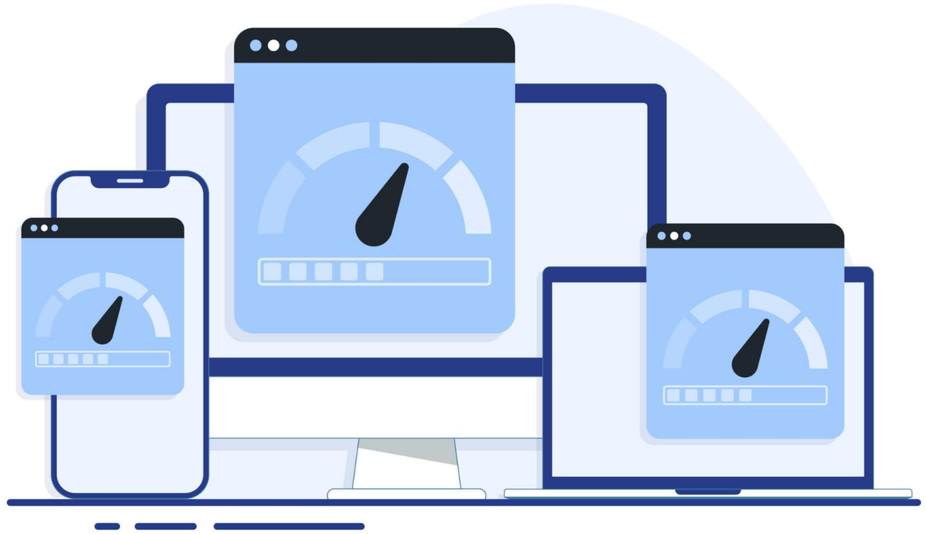
Tipos comunes de pruebas no funcionales:

- Pruebas de carga
- Pruebas de estrés
- Pruebas de escalabilidad
- Pruebas de portabilidad

Tipos de pruebas no funcionales

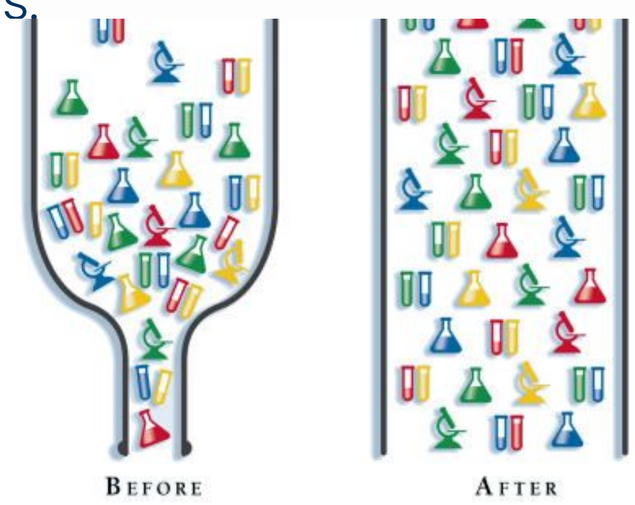
Pruebas de carga

Las pruebas de carga se realizan para verificar el comportamiento del sistema bajo una carga esperada de usuarios, transacciones o datos. Se miden aspectos como el tiempo de respuesta, el uso de recursos y la estabilidad bajo condiciones normales de uso.



Objetivo de las pruebas de carga

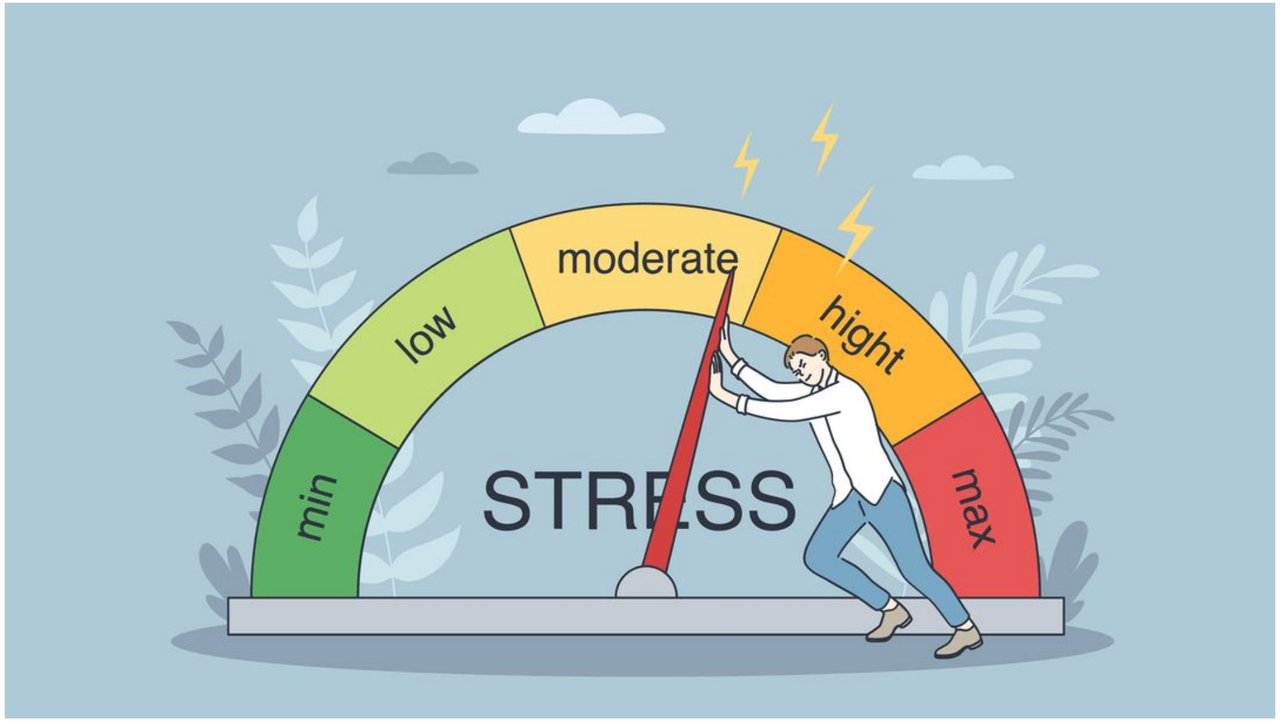
- Determinar cómo el sistema maneja múltiples usuarios o transacciones de manera simultánea.
- Identificar cuellos de botella o problemas de rendimiento cuando el sistema está bajo su carga máxima esperada.
- Garantizar que el tiempo de respuesta sea adecuado bajo las cargas de trabajo normales.



Simular 1,000 usuarios concurrentes accediendo a una aplicación web, midiendo cómo el servidor maneja la carga y los tiempos de respuesta.

Pruebas de estrés

Las pruebas de estrés evalúan el comportamiento del sistema bajo condiciones extremas, mucho más allá de su carga esperada. El objetivo es determinar en qué momento el sistema falla, cómo responde en situaciones de sobrecarga y cómo se recupera después de una falla.



Objetivo de las pruebas de estrés

- Identificar el límite máximo de carga que el sistema puede soportar antes de fallar.
- Evaluar si el sistema puede recuperarse de un fallo, y si lo hace, en cuánto tiempo.
- Asegurar que, aunque el sistema falle, lo haga de manera controlada, sin pérdida de datos o funcionalidad crítica.

Someter a una tienda en línea a un tráfico 10 veces superior al máximo esperado durante un evento promocional para ver si el sitio colapsa o cómo responde a la sobrecarga.

Locust

es una herramienta de código abierto diseñada para realizar pruebas de carga y pruebas de estrés en aplicaciones web y servicios. Es especialmente popular por su facilidad de uso, flexibilidad y la posibilidad de escribir scripts de prueba en Python. Aquí te doy más información sobre sus características y usos.

Tipos de pruebas con Locust

- **Pruebas de carga:** Simula un número determinado de usuarios concurrentes para medir cómo se comporta el sistema bajo condiciones de carga esperada.
- **Pruebas de estrés:** Aumenta progresivamente la carga hasta llevar el sistema a su límite, con el objetivo de observar el punto de fallo.
- **Pruebas de rendimiento:** Evalúa tiempos de respuesta y capacidad de manejo de solicitudes bajo diferentes escenarios.

LOCUST

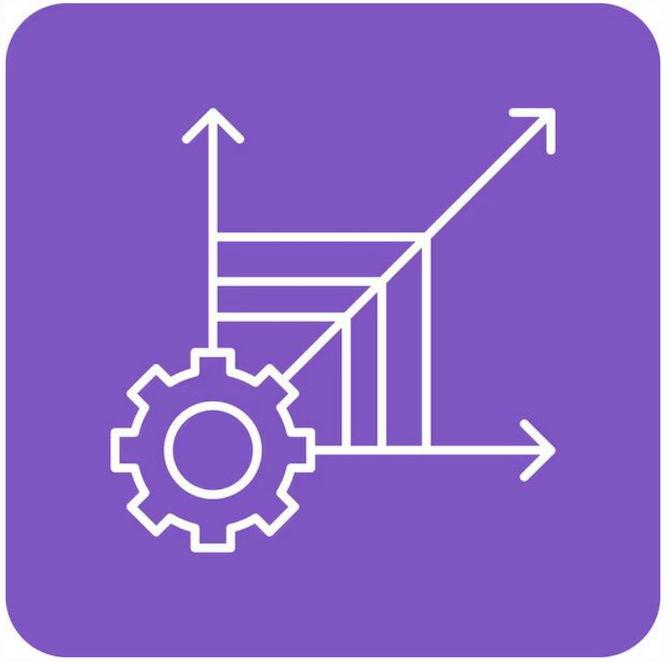
HOST	STATUS	USERS	WORKERS	RPS	FAILURES			
http://api.initech.com	RUNNING	21400	6	233.1	0%	EDIT	STOP	RESET

STATISTICS | CHARTS | FAILURES | EXCEPTIONS | CURRENT RATIO | DOWNLOAD DATA | LOGS | WORKERS

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
GET	/	4137	0	21	37	38	21.22	4	38	20117.16	40	0
GET	/blog	1314	0	26	47	49	25.69	3	49	19749.39	14.3	0
GET	/blog/[post-slug]	1420	0	14	26	27	14.12	2	27	20116	13.1	0
POST	/groups/create	149	0	58	100	110	59.1	5	109	3273.26	1.7	0
GET	/signin	8199	0	26	47	49	26.01	3	49	20070.84	68	0
POST	/signin	8199	0	83	120	120	82.56	45	120	20082.7	68	0
GET	/users/[username]	1369	0	31	53	55	30.47	6	55	19932.43	13.3	0
POST	/users/[username]	132	0	68	120	120	68.06	12	119	11175.93	2	0
GET	/v1/users/	1324	0	26	47	49	26.13	3	49	19932.11	12.7	0

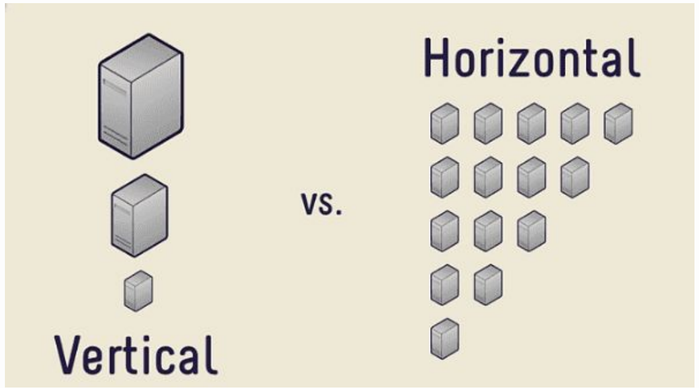
Pruebas de Escalabilidad

Las pruebas de escalabilidad evalúan la capacidad de un sistema para manejar un aumento progresivo en la carga (usuarios, datos, transacciones) mientras mantiene el rendimiento adecuado. Estas pruebas verifican si el sistema puede expandirse o reducirse adecuadamente en función de las necesidades sin afectar su funcionalidad.



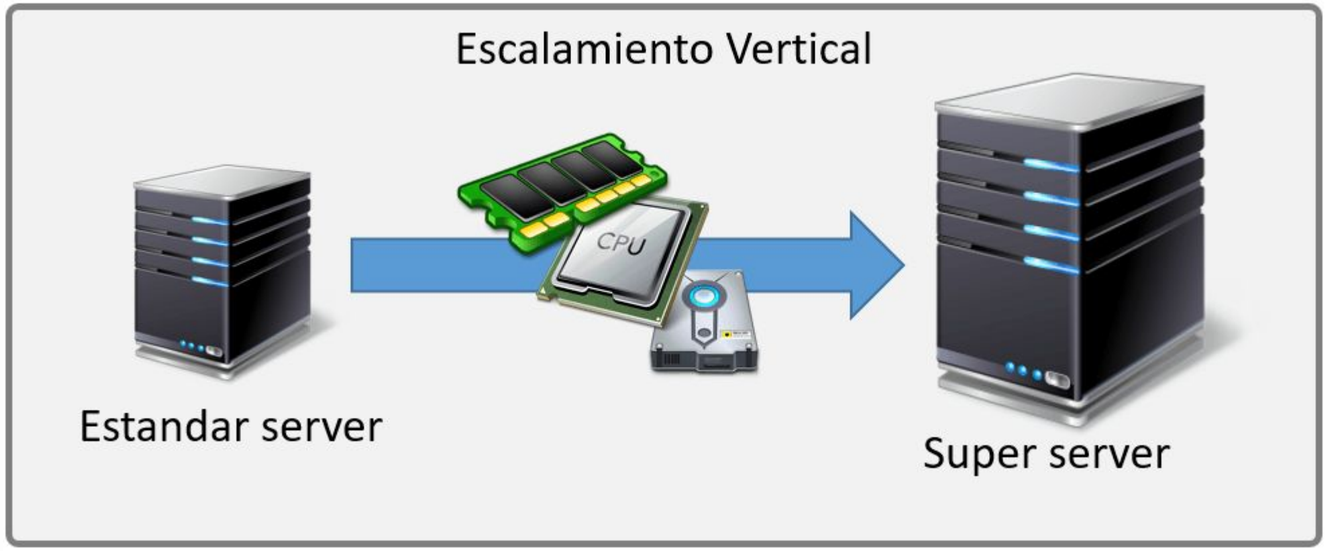
¿Qué es la escalabilidad?

La escalabilidad es una de las características más deseables en las aplicaciones y una de las principales preocupaciones para equipos de desarrollo y administración de servidores. Básicamente, se refiere a la capacidad de crecimiento de la aplicación para atender a un número cada vez mayor de solicitudes y usuarios con total normalidad y sin degradaciones de servicio. De cara a conseguir esta deseada cualidad, existen diferentes vías que no son incompatibles: algunas más centradas en el desarrollo del software y otras en los servidores que harán de plataforma de ejecución.



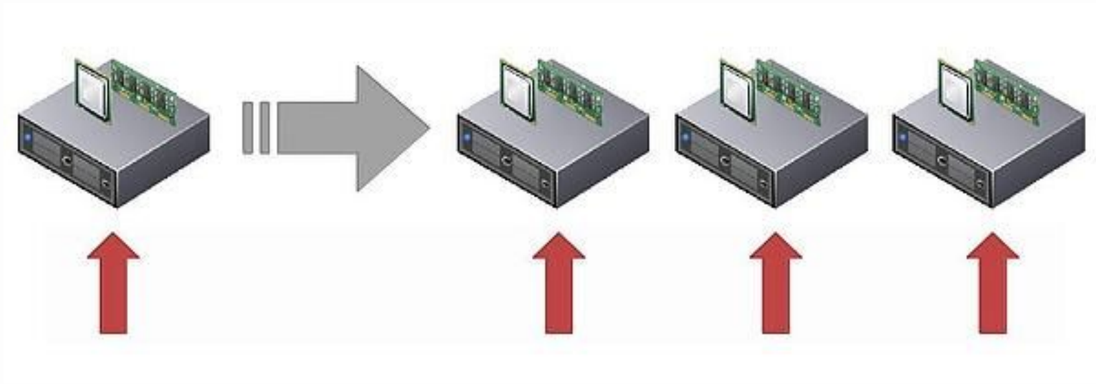
¿Qué es la escalabilidad vertical?

El escalado vertical tiene mucho que ver con el hardware del servidor de la aplicación. Se consigue de una manera muy sencilla: aumentando los recursos del servidor. Principalmente, en lo que respecta a la capacidad de procesamiento, memoria y almacenamiento.



¿Qué es la escalabilidad horizontal?

Por su parte, la escalabilidad horizontal se consigue aumentando el número de servidores que atienden una aplicación. Para ello, un grupo de distintos servidores se configura para atender las peticiones de manera conjunta (es lo que se denomina cluster) y la carga de trabajo se distribuye entre ellos a través de un balanceador. Cada uno de esos servidores se conoce como nodo y el escalado se realiza simplemente agregando un nuevo nodo al cluster.



Pruebas de Portabilidad

Las pruebas de portabilidad evalúan la capacidad del sistema para operar en diferentes entornos (hardware, software, plataformas). Estas pruebas verifican la adaptabilidad del sistema y si es capaz de funcionar correctamente en distintos sistemas operativos, navegadores o dispositivos.

Objetivo de las Pruebas de Portabilidad

- Garantizar que el sistema funcione de manera consistente y sin errores al trasladarse de un entorno a otro (por ejemplo, de Windows a Linux).
- Verificar que los usuarios puedan ejecutar el sistema en una variedad de configuraciones sin problemas.


CLASIFICACION Y TIPOS
DE PRUEBAS