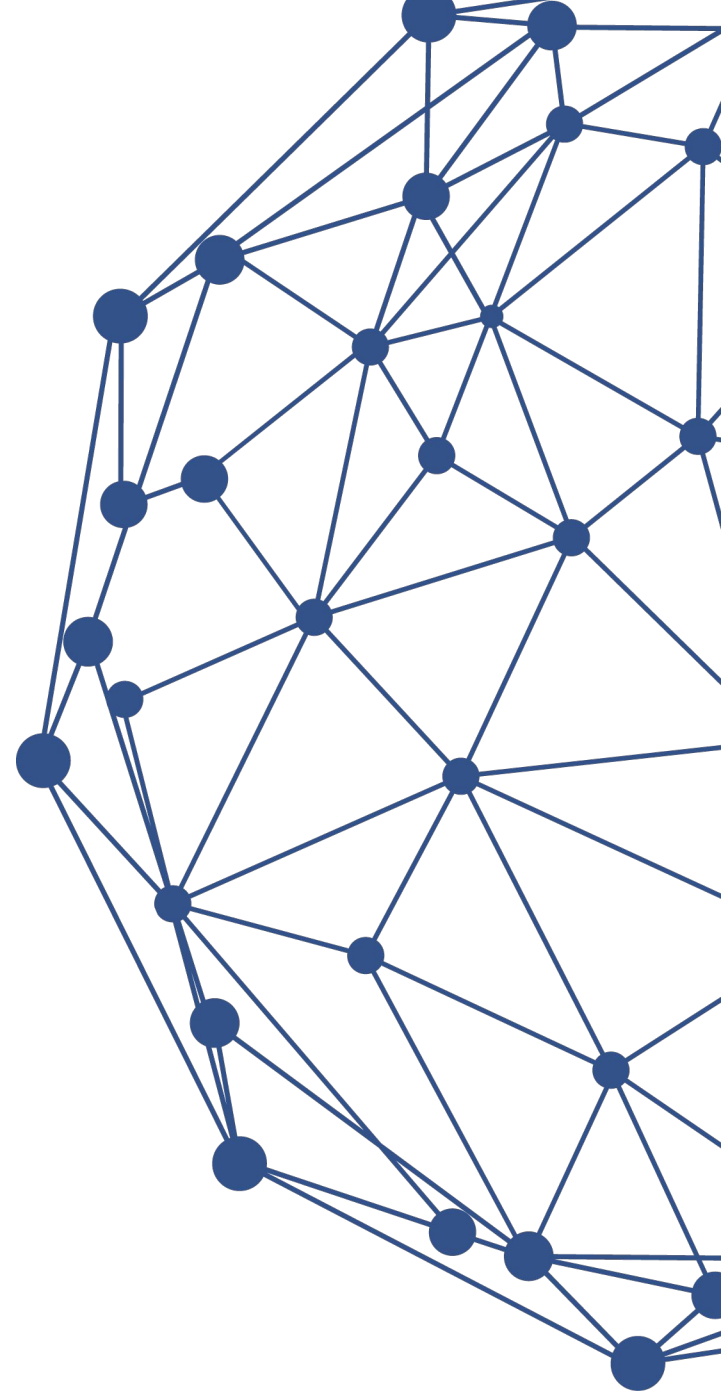




UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA
Construyendo el futuro

Diseño de Pruebas





Pruebas de carga - Python vs PHP

Introducción

Puntos claves

Importancia del Rendimiento: El rendimiento de un sitio web es crucial para la experiencia del usuario y el éxito del negocio.

Comparación entre Python y PHP: Evaluaremos el rendimiento de un sitio web desarrollado en Python (utilizando Django) y otro en PHP (utilizando Laravel) bajo cargas de trabajo simuladas.



Caso de Estudio:



Sitio Web en Python (Django):

Versión de Django: 3.2

Descripción: Desarrollado utilizando Django, un marco de trabajo de alto nivel para Python.

Funcionalidades: Implementa funcionalidades CRUD para gestionar productos.

Base de Datos: Mysql.

Sitio Web en PHP (Laravel):

Versión de Laravel: 8.2

Descripción: Desarrollado utilizando Laravel, un marco de trabajo elegante y expresivo para PHP.

Funcionalidades: También implementa funcionalidades CRUD para gestionar productos.

Base de Datos: MySQL.



Caso de Estudio:



Aspecto	PHP (Laravel)	Python (Django)
Lenguaje de Programación	PHP	Python
Framework	Laravel	Django
Versión del Framework	8.x	3.2
Base de Datos	MySQL o PostgreSQL	SQLite (Desarrollo) PostgreSQL (Producción)
Escenarios de Pruebas	CRUD de productos	CRUD de productos
Tiempo de Respuesta Promedio (ms)	(Resultados)	(Resultados)
Tasa de Errores (%)	(Resultados)	(Resultados)
Utilización del Servidor	CPU y memoria	CPU y memoria

Herramienta Utilizada:



JMeter: Una herramienta de prueba de carga de código abierto que simula cargas pesadas en aplicaciones web, bases de datos y servidores.

Versión: 5.6.3



Configuraciones de Pruebas:



Escenarios de Pruebas: Se diseñaron escenarios de pruebas para simular diferentes cargas de usuarios, incluyendo:

- Creación de productos.
- Lectura de productos.
- Actualización de productos.
- Eliminación de productos.

Configuración de Usuarios Virtuales: Se crearon rampas de usuarios virtuales para simular aumentos graduales en la carga.

Duración de Pruebas: Cada prueba se ejecutó durante un período de tiempo específico para recopilar datos significativos.



Tipos de Pruebas:



Tipos de Pruebas Realizadas:

Pruebas de Carga: Se evaluó el rendimiento de los sitios web bajo diferentes niveles de carga simulada.

Pruebas de Estrés: Se aumentó la carga gradualmente para determinar el punto de quiebre o saturación del sistema.

Pruebas de Rendimiento: Se midió el tiempo de respuesta y la tasa de errores para evaluar el rendimiento general de los sitios web.



METODOS



Python (Django):

- GET:
 - Obtener todos los productos: `GET /api/products/`
 - Obtener un producto específico: `GET /api/products/{id}`
- POST:
 - Crear un nuevo producto: `POST /api/products/`
- PUT:
 - Actualizar un producto existente: `PUT /api/products/{id}`
- DELETE:
 - Eliminar un producto existente: `DELETE /api/products/{id}`

PHP (Laravel):

- GET:
 - Obtener todos los productos: `GET /api/products`
 - Obtener un producto específico: `GET /api/products/{id}`
- POST:
 - Crear un nuevo producto: `POST /api/products`
- PUT:
 - Actualizar un producto existente: `PUT /api/products/{id}`
- DELETE:
 - Eliminar un producto existente: `DELETE /api/products/{id}`

Tipos de Pruebas:



1. Prueba de Rendimiento del CRUD:
 - Objetivo: Evaluar el rendimiento de las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en la base de datos.
 - Configuración: Simula acciones de creación, lectura, actualización y eliminación de productos en la aplicación.
 - Métricas a Medir: Tiempo de respuesta promedio para cada operación CRUD, tasa de error, utilización de recursos del servidor.
2. Prueba de Rendimiento de la API:
 - Objetivo: Evaluar el rendimiento de la API de cada aplicación.
 - Configuración: Simula solicitudes HTTP directas a las API de la aplicación para realizar operaciones CRUD.
 - Métricas a Medir: Tiempo de respuesta promedio de las solicitudes API, tasa de error, rendimiento del servidor.
3. Prueba de Escalabilidad Horizontal:
 - Objetivo: Evaluar cómo cada aplicación escala al agregar más instancias del servidor.
 - Configuración: Incrementa gradualmente el número de instancias del servidor y evalúa cómo afecta al rendimiento global.
 - Métricas a Medir: Tiempo de respuesta promedio, tasa de error, rendimiento del servidor bajo diferentes cargas y configuraciones de escalabilidad.

Tipos de Pruebas:

- **Número de hilos:** 100 (Número de usuarios que se conectan al sitio web de destino: 100)
- **Cuenta de bucle:** 10 (Número de tiempo para ejecutar la prueba)
- **Ramp-Período de subida:** 10



Caso de Estudio:



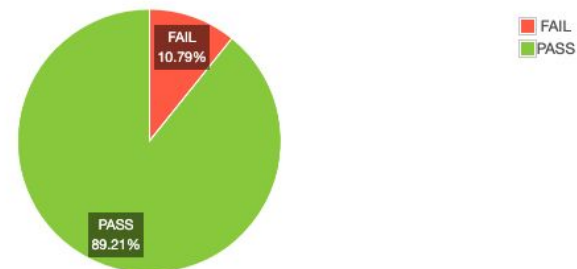
Test and Report information

Source file	"sumary.csv"
Start Time	"5/5/24, 02:26"
End Time	"5/5/24, 02:27"
Filter for display	" "

APDEX (Application Performance Index)

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
0.719	500 ms	1 sec 500 ms	Total
0.574	500 ms	1 sec 500 ms	PHP
0.792	500 ms	1 sec 500 ms	PHP-0
0.796	500 ms	1 sec 500 ms	PHP-1
0.838	500 ms	1 sec 500 ms	PYTHON

Requests Summary



Caso de Estudio:



Requests		Executions		Response Times (ms)							Throughput	Network (KB/sec)	
Label	#Samples	FAIL	Error %	Average	Min	Max	Median	90th pct	95th pct	99th pct	Transactions/s	Received	Sent
Total	14004	1511	10.79%	453.26	2	4821	206.00	957.00	1363.50	4663.95	158.58	2185.75	42.91
PHP	6000	997	16.62%	752.72	7	4821	525.00	1307.00	4289.00	4777.99	67.95	946.27	17.37
PHP-0	1002	0	0.00%	432.48	7	738	463.50	663.00	680.00	717.00	16.51	10.48	4.70
PHP-1	1002	0	0.00%	438.54	6	741	465.50	668.70	683.00	721.00	16.51	301.22	2.39
PYTHON	6000	514	8.57%	159.74	2	3849	5.00	799.80	1082.95	1613.94	68.11	1027.79	20.73

Errors

Type of error	Number of errors	% in errors	% in all samples
404/Not Found	1497	99.07%	10.69%
Non HTTP response code: java.net.SocketException/Non HTTP response message: Broken pipe	14	0.93%	0.10%

CONCLUSIÓN



En conclusión, este estudio comparativo exhaustivo tuvo como objetivo desentrañar las complejidades del rendimiento de PHP y Python, proporcionando a los tomadores de decisiones y desarrolladores una visión matizada de sus respectivas fortalezas y debilidades, resaltando aspectos clave que influyen en la selección entre PHP y Python para diversos proyectos de desarrollo de software.