



UNIVERSIDAD DE GRANADA

SWAP

Práctica 5

Trabajo individual

Nuria Manzano Mata

Profesor: José Manuel Soto Hidalgo



Tiempo dedicado: 7 horas

ÍNDICE

1. Introducción.....	2
2. Desarrollo tareas básicas.....	2
2.1. B1. Configuración del entorno de benchmarking.....	2
2.2. B2. Implementación con Apache Benchmark.....	3
2.3. B3. Implementación con Locust.....	5
2.4. B4. Ejecución de pruebas de carga.....	7
2.5. B5. Análisis de resultados.....	10
2.6. Análisis propuesta IA.....	14
3. Desarrollo tareas avanzadas.....	15
3.1. A1. Desarrollar tareas avanzadas en Locustfile.py.....	16
3.2. A2. Crear escenario multicontenedor con algún CMS.....	18
3.3. A3. Ejecución y Análisis de cargas de prueba avanzadas sobre CMS.....	21
3.4. Análisis propuesta IA.....	29
Bibliografía:.....	30

1. Introducción

Esta práctica tiene como objetivo principal evaluar el rendimiento de nuestra infraestructura web utilizando contenedores Docker y aplicando diferentes test de carga a través de herramientas de Benchmarking. El **desarrollo de este documento se estructura de manera progresiva**, abordando cada tarea con un **doble propósito**: explicar detalladamente los procedimientos y soluciones implementadas, y justificar el razonamiento detrás de cada decisión. Para facilitar la comprensión, se incluyen capturas de pantalla que muestran el proceso. Debido a la extensión del documento, algunos apartados se presentan de manera concisa, mostrando los resultados más significativos o los aspectos más importantes.

El .zip entregado contiene dos carpetas, una llamada “P5” donde está todo el desarrollo de las tareas básicas y otra carpeta llamada “P5-ADVANCED” donde está todo el desarrollo de las tareas avanzadas.

2. Desarrollo tareas básicas

2.1. B1. Configuración del entorno de benchmarking

- Preparación de un entorno de trabajo específico para realizar tests de carga.
- Creación de directorios y archivos necesarios para los tests con Apache Benchmark (AB) y Locust.

Para la preparación del entorno de trabajo, se han tenido que crear una serie de directorios donde en el primer directorio “P5/P5-granjaweb” se han incorporado diferentes archivos realizados en la práctica anterior. Para el resto de directorios creados, he añadido el archivo “docker-compose.yml” que más adelante se necesitará.

```
2025-05-29 13:52:43 🕒 Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5
[○ → ls
P5-ab          P5-granjaweb P5-locust

2025-05-29 13:54:05 🕒 Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5
[○ → ls P5-granjaweb/
P5-mmnnuria-apache      P5-mmnnuria-nginx      web_mmnnuria
P5-mmnnuria-certificados docker-compose.yml

2025-05-29 13:53:58 🕒 Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5
[○ → ls P5-ab/
docker-compose.yml
```

```
2025-05-29 13:54:40 Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUAT
RI/SWAP/ManzanoMataNuriaP5
[○ → ls P5-locust/
docker-compose.yml
```

2.2. B2. Implementación con Apache Benchmark

- Desarrollo de un Dockerfile (DockerFileAB) que configure un contenedor para ejecutar AB.

El desarrollo de este apartado se ha realizado en la carpeta “P5/P5-ab” donde he optado por crear un Dockerfile bastante simple:

```
2025-05-30 12:13:03 Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUAT
RI/SWAP/ManzanoMataNuriaP5/P5/P5-ab
[○ → cat DockerFileAb
FROM debian:latest
RUN apt-get update && apt-get install -y apache2-utils
```

- Ejecución de AB desde un contenedor, lanzando peticiones HTTP y HTTPS al balanceador en la dirección 192.168.10.50.

Este apartado se ha conseguido creando un fichero docker-compose.yml donde se crea un contenedor que lanza peticiones al balanceador de carga que tenemos implementado previamente en “P5/P5-granjaweb/docker-compose.yml” (se realizó en la práctica anterior). Muestro información del docker-compose.yml nuevo ya que, el que contiene la granja web es igual que la práctica 4.

```
2025-05-30 12:14:58 Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUAT
RI/SWAP/ManzanoMataNuriaP5/P5/P5-ab
[○ → cat docker-compose.yml

services:
  apache-benchmark-P5:
    build:
      context: .
      dockerfile: DockerFileAb
    image: mmnuria-ab-image:p5
    container_name: apache_benchmark-P5
    command: [ "ab", "-n", "10000", "-c", "100", "https://192.168.10.50:443/" ]
    networks:
      p5-granjaweb_red_web:
        ipv4_address: 192.168.10.60

networks:
  p5-granjaweb_red_web:
    external: true
```

- Asegurar que el contenedor AB esté en la misma red (red_web) que el balanceador.

Se puede ver cómo se ha añadido el contenedor AB a la misma red en la que está el balanceador, gracias a la siguiente línea:

```
networks:
  p5-granjaweb_red_web:
    external: true
```

Muestro la siguiente captura de las redes creadas en mi docker:

```
2025-05-30 12:21:19 ✘ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuri
ap5/P5-ADVANCED/P5-locust
● o → docker network ls
NETWORK ID      NAME          DRIVER      SCOPE
9303cb8da8dc   bridge        bridge      local
8e326673211a   dai-p7_default bridge      local
b56103d9fee0   host          host       local
40f685edd134   none          null       local
4c639b75017d   p5-granjaweb_red_servicios bridge      local
7e8870c1fbff   p5-granjaweb_red_web   bridge      local
1e3ba203e3d1   practical_default bridge      local
3ff0297b56ba   proyecto-dai_default bridge      local
```

El balanceador está en la red:

```

2025-05-30 12:26:25 ✘ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuri
aP5/P5/P5-ab
● o → docker inspect balanceador-nginx-ssl | grep -A 20 "Networks"
"Networks": {
    "p5-granjaweb_red_web": {
        "IPAMConfig": {
            "IPv4Address": "192.168.10.50"
        },
        "Links": null,
        "Aliases": [
            "balanceador-nginx-ssl",
            "nginx-ssl"
        ],
        "MacAddress": "66:11:45:67:a7:63",
        "DriverOpts": null,
        "GwPriority": 0,
        "NetworkID": "7e8870c1bfbfd6cba947fcf59285466e0cb19f33e3657154afb66fcc01f3f575",
        "EndpointID": "04a6dbb52ed210e81e4986c1e05f349355dc8e5ca21f0ce1ff791f3da133f84f",
        "Gateway": "192.168.10.1",
        "IPAddress": "192.168.10.50",
        "IPPrefixLen": 24,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
    }
}

```

Y el contenedor AB:

```

2025-05-30 12:26:46 ✘ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuri
aP5/P5/P5-ab
● o → docker inspect apache_benchmark-P5 | grep -A 20 "Networks"
"Networks": {
    "p5-granjaweb_red_web": {
        "IPAMConfig": {
            "IPv4Address": "192.168.10.60"
        },
        "Links": null,
        "Aliases": [
            "apache_benchmark-P5",
            "apache-benchmark-P5"
        ],
        "MacAddress": "",
        "DriverOpts": null,
        "GwPriority": 0,
        "NetworkID": "7e8870c1bfbfd6cba947fcf59285466e0cb19f33e3657154afb66fcc01f3f575",
        "EndpointID": "",
        "Gateway": "",
        "IPAddress": "",
        "IPPrefixLen": 0,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
    }
}

```

Por lo que, nos aseguramos doblemente de que están correctamente configurados para acceder a la misma red web.

2.3. B3. Implementación con Locust

- Configuración de un docker-compose.yml para ejecutar Locust con un nodo master y múltiples workers (ejemplo: 6).

La configuración de este apartado se ha realizado en la ruta “P5/P5-locust”, en ese directorio tenemos el archivo necesario para la resolución de este apartado, docker-compose.yml que ejecuta un nodo master y 6 workers:

```

2025-05-30 12:31:01 ✉ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/S
WAP/ManzanoMataNuriaP5/P5/P5-locust
[○ → cat docker-compose.yml
services:
  master-mmnuria:
    image: locustio/locust
    ports:
      - "8098:8089"
    volumes:
      - ./mnt/locust
    command: -f /mnt/locust/locustfile.py --master -H https://192.168.10.50:443/
    networks:
      p5-granjaweb_red_web:
        ipv4_address: 192.168.10.70

  worker-mmnuria:
    image: locustio/locust
    volumes:
      - ./mnt/locust
    command: -f /mnt/locust/locustfile.py --worker --master-host master-mmnuria
    depends_on:
      - master-mmnuria
    deploy:
      replicas: 6
    networks:
      - p5-granjaweb_red_web

networks:
  p5-granjaweb_red_web:
    external: true

```

- Preparación del archivo locustfile.py con tareas definidas para realizar peticiones HTTP y HTTPS a la granja web y tiempo aleatorio entre peticiones.

El archivo locustfile.py se encuentra en la misma ruta “P5/P5-locust” donde se definen las peticiones correspondientes a la granja web con un tiempo aleatorio:

```

2025-05-30 12:31:04 ✉ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/S
WAP/ManzanoMataNuriaP5/P5/P5-locust
[○ → cat locustfile.py

from locust import HttpUser, TaskSet, task, between
class P5_mmnuria(TaskSet):
    @task
    def load_index(self):
        self.client.get("/index.php", verify=False)
class P5_usuarios(HttpUser):
    tasks = [P5_mmnuria]
    wait_time = between(1, 5)

```

- Establecimiento de la configuración necesaria para permitir peticiones HTTPS a un sitio con certificado autofirmado (uso de verify=False).

Este apartado se ha realizado en conjunto al anterior, añadiendo la línea `verify = False` como se ha sugerido en la práctica:

```
2025-05-30 12:31:04 ✉ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5/P5/P5-locust
[○ → cat locustfile.py

from locust import HttpUser, TaskSet, task, between
class P5_mmnuria(TaskSet):
    @task
    def load_index(self):
        self.client.get("/index.php", verify=False)
class P5_usuarios(HttpUser):
    tasks = [P5_mmnuria]
    wait_time = between(1, 5)
```

2.4. B4. Ejecución de pruebas de carga

- Despliegue y ejecución de los escenarios con Apache Benchmark y Locust, monitorizando el comportamiento y el rendimiento de la granja web bajo carga.

A continuación, muestro de forma ordenada los pasos que se deben seguir para el despliegue y ejecución de todas las configuraciones anteriores para asegurar la correcta resolución de las tareas básicas:

1. Lanzamos la granja web y comprobamos su correcto despliegue:

```
2025-05-30 12:22:45 ✉ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5/P5/P5-granjaweb
● ○ → docker compose up -d
[+] Running 9/9
✓ Container web6           Running      0.0s
✓ Container web8           Running      0.0s
✓ Container web7           Running      0.0s
✓ Container web3           Running      0.0s
✓ Container web5           Running      0.0s
✓ Container web4           Running      0.0s
✓ Container web1           Running      0.0s
✓ Container web2           Running      0.0s
✓ Container balanceador-nginx-ssl Running      0.0s
```

<input type="checkbox"/>	<input checked="" type="checkbox"/>	p5-granjaweb	-	-	-	0.07%	57 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web6	c7af70e66b1e	mmnuria-apache-im:	8091:443 ↗ Show all ports (2)	0.01%	59 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web7	ce65989ef3f	mmnuria-apache-im:	8093:443 ↗ Show all ports (2)	0.01%	59 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web1	4ba3ac3516bc	mmnuria-apache-im:	8081:443 ↗ Show all ports (2)	0%	59 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web3	ab8e136cce39	mmnuria-apache-im:	8085:443 ↗ Show all ports (2)	0.01%	59 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web2	0458ebd0aca1	mmnuria-apache-im:	8083:443 ↗ Show all ports (2)	0.01%	59 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web8	5b0e2840a88f	mmnuria-apache-im:	8095:443 ↗ Show all ports (2)	0.01%	59 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web4	d594e050c1f0	mmnuria-apache-im:	8087:443 ↗ Show all ports (2)	0.01%	59 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	balanceador-nginx	b28a85e0a6b4	mmnuria-nginx-img:	8097:443 ↗ Show all ports (2)	0%	57 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web5	23c7cf641062	mmnuria-apache-im:	8089:443 ↗ Show all ports (2)	0.01%	59 seconds ago	⋮	trash

2. Lanzamos contenedor Apache Benchmark y comprobamos su correcto despliegue:

2025-05-30 12:24:22 ☑ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUA TRI/SWAP/ManzanoMataNuriaP5/P5/P5-ab									
● o → docker compose up -d									
[+] Running 1/1									
✓ Container apache_benchmark-P5 Star...									
0.4s									
<input type="checkbox"/>	<input checked="" type="checkbox"/>	p5-ab	-	-	-	75.91%	21 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	apache_benchmar	d0f583e536f9	mmnuria-ab-image:p	-	75.91%	21 seconds ago	⋮	trash

3. Lanzamos contenedor Locust y comprobamos su correcto despliegue:

[+] Running 7/7-locust-worker-mmnuria-6 Starting 2.4s									
✓ Container p5-locust-master-mmnuria-1 Started 0.7s									
✓ Container p5-locust-worker-mmnuria-6 Started 2.6s									
✓ Container p5-locust-worker-mmnuria-2 Started 2.1s									
✓ Container p5-locust-worker-mmnuria-4 Started 1.7s									
✓ Container p5-locust-worker-mmnuria-3 Started 1.0s									
✓ Container p5-locust-worker-mmnuria-5 Started 3.0s									
✓ Container p5-locust-worker-mmnuria-1 Started 1.3s									
<input type="checkbox"/>	<input checked="" type="checkbox"/>	p5-locust	-	-	-	1.69%	20 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	master-mmnuria-1	58383f10626c	locustio/locust	8098:8089 ↗	0.85%	22 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-4	86a2cbef806f	locustio/locust	-	0.1%	21 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-1	3addd9298d29	locustio/locust	-	0.08%	22 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-5	ae2400e8c09c	locustio/locust	-	0.12%	20 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-2	a334772e25ae	locustio/locust	-	0.14%	21 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-6	e953ff33e2af	locustio/locust	-	0.19%	21 seconds ago	⋮	trash
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-3	b4145dbd76f0	locustio/locust	-	0.21%	22 seconds ago	⋮	trash

4. Finalmente, comprobamos con docker ps que efectivamente está todo funcionando sin problema:

2025-05-30 12:36:30 ✎ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuri aP5/P5/P5-locust					
● o → docker ps	CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
		locustio/locust	"locust -f /mnt/locu..."	About a minute ago	Up About a
	b4145dbd76f0	locustio/locust	"p5-locust-worker-mmnuria-3		
	minute	5557/tcp, 8089/tcp	"locust -f /mnt/locu..."	About a minute ago	Up About a
	3add9298d29	locustio/locust	"p5-locust-worker-mmnuria-1		
	minute	5557/tcp, 8089/tcp	"locust -f /mnt/locu..."	About a minute ago	Up About a
	86a2cbcf806f	locustio/locust	"p5-locust-worker-mmnuria-4		
	minute	5557/tcp, 8089/tcp	"locust -f /mnt/locu..."	About a minute ago	Up About a
	a334772e25ae	locustio/locust	"p5-locust-worker-mmnuria-2		
	minute	5557/tcp, 8089/tcp	"locust -f /mnt/locu..."	About a minute ago	Up About a
	e953ff33e2af	locustio/locust	"p5-locust-worker-mmnuria-6		
	minute	5557/tcp, 8089/tcp	"locust -f /mnt/locu..."	About a minute ago	Up About a
	ae2400e8c09c	locustio/locust	"p5-locust-worker-mmnuria-5		
	minute	5557/tcp, 8089/tcp	"locust -f /mnt/locu..."	About a minute ago	Up About a
	58383f10626c	locustio/locust	"p5-locust-master-mmnuria-1		
	minute	5557/tcp, 0.0.0.0:8098->8089/tcp	"ab -n 10000 -c 100 ..."	13 minutes ago	Up 4 secon
	d0f583e536f9	mmnuria-ab-image:p5	"apache_benchmark-P5		
ds					
	b28a85e0a6b4	mmnuria-nginx-image:p5	"/docker-entrypoint..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8096->80/tcp, 0.0.0.0:8097->443/tcp	balanceador-nginx-ssl		
	ab8e136cce39	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8084->80/tcp, 0.0.0.0:8085->443/tcp	web3		
	ce65989eef3f	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8092->80/tcp, 0.0.0.0:8093->443/tcp	web7		
	4ba3ac3516bc	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8080->80/tcp, 0.0.0.0:8081->443/tcp	web1		
	23c7cf641062	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8088->80/tcp, 0.0.0.0:8089->443/tcp	web5		
	0458ebd0aca1	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8082->80/tcp, 0.0.0.0:8083->443/tcp	web2		
	5b0e2840a88f	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8094->80/tcp, 0.0.0.0:8095->443/tcp	web8		
	d594e050c1f0	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8086->80/tcp, 0.0.0.0:8087->443/tcp	web4		
	c7af70e66b1e	mmnuria-apache-image:p5	"/entrypoint.sh apac..."	15 minutes ago	Up 15 minu
tes		0.0.0.0:8090->80/tcp, 0.0.0.0:8091->443/tcp	web6		
s	00a11bc1aee9	moby/buildkit:buildx-stable-1	"buildkitd"	2 weeks ago	Up 22 hour
			buildx_buildkit_loving_hodgkin0		

- Pruebas realizadas con al menos 10000 peticiones y 100 usuarios en concurrencia así como peticiones https y http.

Comenzaremos las pruebas de peticiones, la parte de Apache Benchmark se realiza directamente al ejecutar el contenedor, por lo que mostraré los resultados en el siguiente apartado y para las pruebas de Locust se realizan configurando primeramente desde la interfaz web:

The screenshot shows the Locust web interface at `localhost:8098`. At the top, there's a header with the Locust logo, the URL, and status information: Host `https://192.168.10.50:443/`, Status READY, Workers 6, RPS 0, Failures 0%. Below the header, there's a section titled "Start new load test" with fields for "Number of users (peak concurrency)*" (set to 10000), "Ramp up (users started/second)*" (set to 100), and "Host" (set to `https://192.168.10.50:443/`). A "Advanced options" section contains a "Run time (e.g. 20, 20s, 3m, 2h, 1h20m, 3h30m10s, etc.)" field set to 5m, and a "Profile" field. At the bottom is a large green "START" button.

2.5. B5. Análisis de resultados

- Revisión y análisis de los datos obtenidos de las pruebas de carga ejecutadas con AB y Locust.

Comenzaré revisando y analizando los resultados de las pruebas por parte de **Apache Benchmark**:

[Containers](#) / apache_benchmark-P5

apache_benchmark-P5

 9f9df30e28ad [Logs](#) [mnnuria-ab-image:p5](#)

STATUS
Exited (0) (47 seconds ago)

[Logs](#) [Inspect](#) [Bind mounts](#) [Exec](#) [Files](#) [Stats](#)

```
This is ApacheBench, Version 2.3 <$Revision: 1913912 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 192.168.10.50 (be patient)
Completed 1000 requests
Completed 2000 requests
Completed 3000 requests
Completed 4000 requests
Completed 5000 requests
Completed 6000 requests
Completed 7000 requests
Completed 8000 requests
Completed 9000 requests
Completed 10000 requests
Finished 10000 requests

Server Software:        nginx/1.27.5
Server Hostname:       192.168.10.50
Server Port:           443
SSL/TLS Protocol:     TLSv1.3,TLS_AES_256_GCM_SHA384,2048,256
Server Temp Key:       X25519 253 bits

Document Path:          /
Document Length:        225 bytes

Concurrency Level:      100
Time taken for tests:   38.247 seconds
Complete requests:      10000
Failed requests:         0
Total transferred:      4060000 bytes
```

```

HTML transferred:          2250000 bytes
Requests per second:    261.46 [#/sec] (mean)
Time per request:        382.468 [ms] (mean)
Time per request:        3.825 [ms] (mean, across all concurrent requests)
Transfer rate:           103.66 [Kbytes/sec] received

Connection Times (ms)
                         min   mean[+/-sd] median   max
Connect:            3   19  26.4      7   209
Processing:        10  361 266.3    296   2269
Waiting:           9  361 266.3    296   2269
Total:             14  380 267.3    318   2302

Percentage of the requests served within a certain time (ms)
 50%    318
 66%    422
 75%    502
 80%    558
 90%    734
 95%    901
 98%   1130
 99%   1286
100%   2302 (longest request)

```

Ahora mostraré los resultados por parte de **Locust**:

- En esta imagen se muestra el resumen de las métricas:

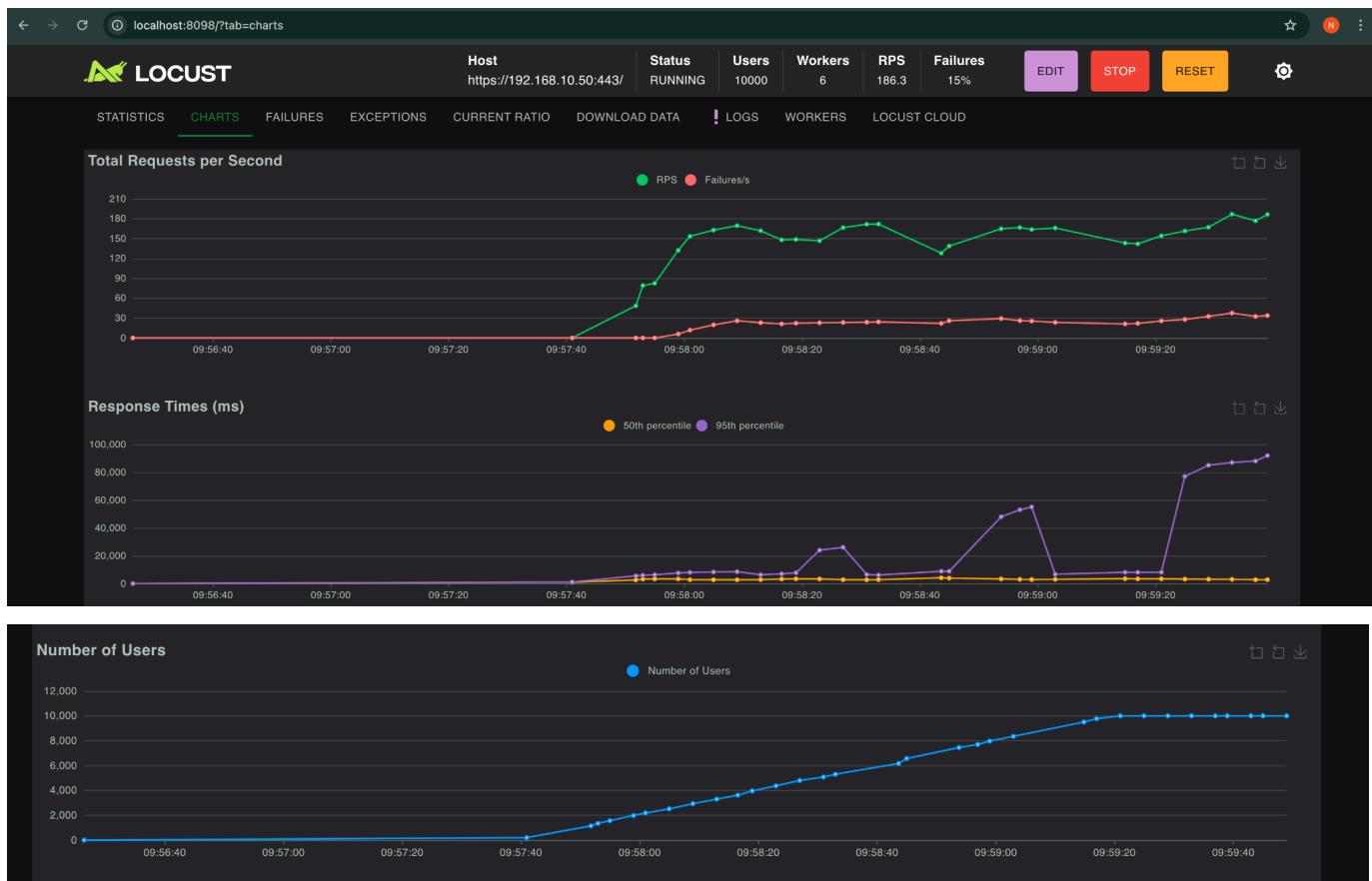
Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	//index.php	11384	1581	3200	8400	51000	4927.34	849	62129	193.75	163.7	25.5
	Aggregated	11384	1581	3200	8400	51000	4927.34	849	62129	193.75	163.7	25.5

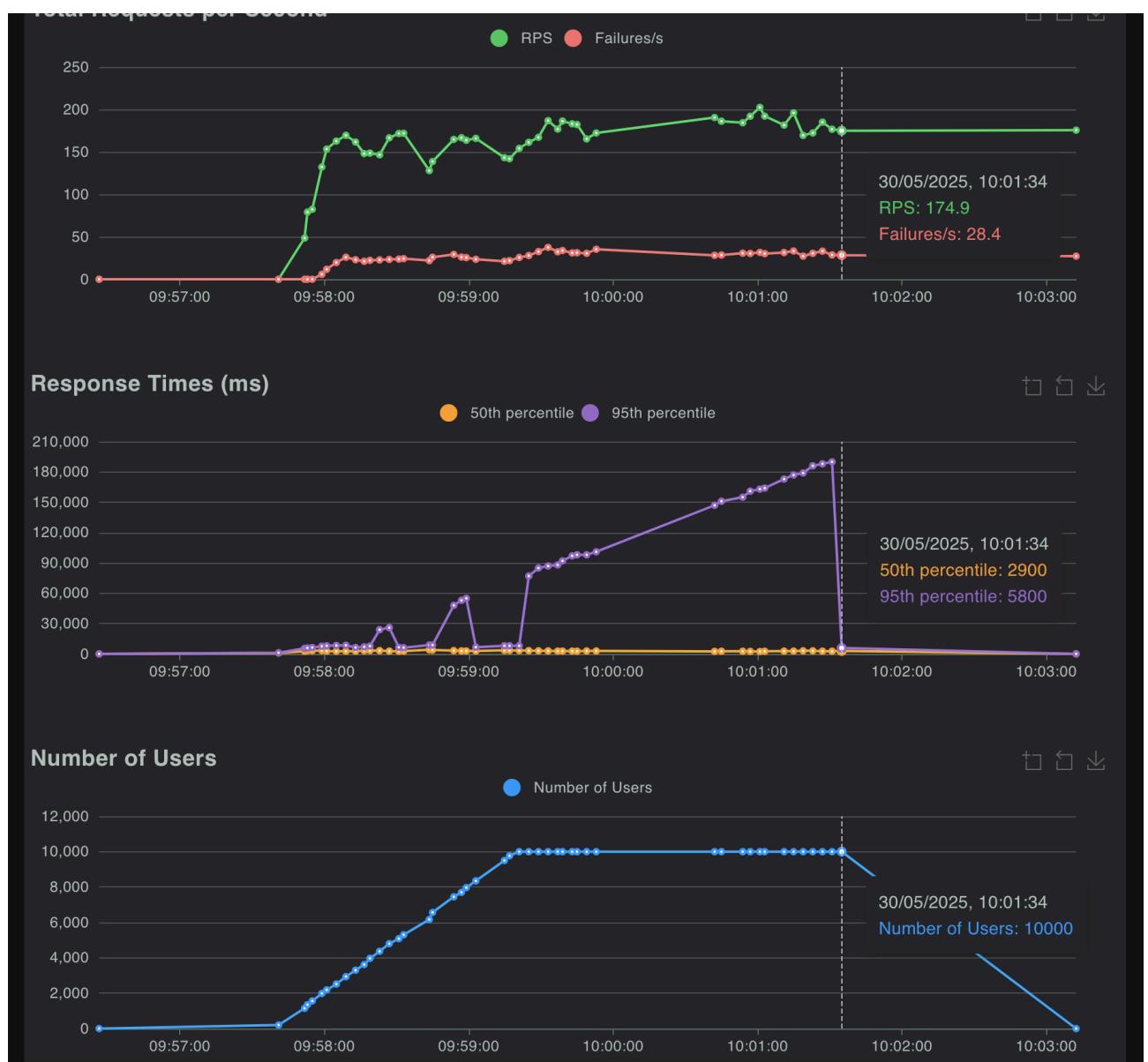
- En esta imagen se muestra el resumen de los workers:

Worker		State	# users	CPU usage	Memory usage
21eb7cf1d79d_22183370edce4af7a57678f0785b1a81		running	800	100.4	226.02 MB
aee5fb27c014_9754e7b2146943c69dba992812542fca		running	800	100.2	244.83 MB
becef228b278_849d8b28b6704d3aa54c9bf9cb57c217		running	800	99.8	233.89 MB
c0a539f0fd64_ac76c767b724418f9ef3ff2220eca2d2		running	800	100.1	241.73 MB
dd3919b4b8c1_57295df856d14e8285e0ca481b133f9f		running	800	99.9	237.11 MB
fc3e8618adcc_d8e0cc3cef314d42adca32d41bd5db18		running	800	100.2	227.3 MB

- Documentación de los resultados, incluyendo métricas como solicitudes por segundo, tiempos de respuesta y errores.

Locust nos ofrece un apartado donde podemos visualizar las métricas de las solicitudes, de forma que adjunto capturas de pantalla con ellas:





2.6. Análisis propuesta IA

Con este apartado voy a tratar de **analizar los resultados propuestos por la IA** para este primer apartado de tareas básicas. He preguntado a la IA escribiendo directamente el enunciado de los diferentes apartados para que me mostrara paso a paso lo que debe de incluir cada uno, [pulsa aquí para acceder al chat](#)

De forma resumida, la IA me ha proporcionado soluciones algo útiles:

- Tarea B1: No he necesitado recurrir a la IA para este apartado, aunque sí quiero destacar que, en ocasiones, las sugerencias por su parte, suelen ser complicando tareas simples con soluciones que incluyen elementos innecesarios. En este caso,

por ejemplo, propuso archivos de configuración o scripts que no eran requeridos en ningún momento de la práctica.

- Tarea B2: Las sugerencias de la IA no me han sido útiles en este apartado. Como comenté anteriormente, se inventó la necesidad de crear un script para ejecutar el contenedor de Apache Benchmark. Para completar esta tarea, me he guiado únicamente por las instrucciones proporcionadas por el profesor.
- Tarea B3: Aquí la IA estuvo un poco más acertada, aunque con algunos errores. He seguido principalmente lo que el profesor explicó en la práctica, pero también incorporé ciertas sugerencias de la IA, haciendo las modificaciones necesarias. Por ejemplo, el puerto que he usado es el 8098, en lugar del 8089 que ya tenía ocupado, y el nombre de la red no es “red_web” como indicaba la IA, sino “p5-granjaweb_red_web” en mi caso. Para la creación del `locustfile.py`, opté por seguir las indicaciones del profesor, ya que, de nuevo, la IA proponía cosas innecesarias. En general, puedo decir que en este apartado los consejos de la IA fueron algo más útiles, pero mi base principal han sido las explicaciones del profesor.
- Tareas B4 y B5: En estas tareas no he necesitado la ayuda de la IA, ya que se trataba simplemente de desplegar los contenedores y verificar su funcionamiento con las pruebas correspondientes. Si bien algunas de las indicaciones de la IA eran correctas, las explicaciones del profesor me han resultado mucho más claras y útiles. En varias ocasiones, la IA tiende a enredar procesos que, en realidad, son bastante más sencillos.

3. Desarrollo tareas avanzadas

Para resolver este apartado, he creado una carpeta nueva llamada "P5-ADVANCED", en la que **cada sección tiene su propia subcarpeta/archivo** para facilitar la verificación del funcionamiento y mantener todo organizado. En cada apartado se especifica el nombre de la subcarpeta/archivo creado y el paso a paso seguido para su resolución.

IMPORTANTE: En este caso, he optado por levantar nuevos contenedores tanto de la granja web y locust, ya que, se tienen que realizar algunas modificaciones. Para su correcta ejecución y comprobación, los anteriores contenedores deben estar borrados ya que, no se han cambiado los puertos, ni direcciones ips, ni los nombres de los contenedores e instancias. Esta decisión la he tomado para agilizar la realización de esta parte, ya que, no perjudica para la correcta consecución en el desarrollo de las tareas avanzadas.

3.1. A1. Desarrollar tareas avanzadas en Locustfile.py

- Definir tareas que incluyan la navegación por páginas, la creación o interacción con contenido, y las interacciones con la base de datos como insertar comentarios o publicaciones.
- Incorporar tareas que reflejen operaciones típicas del CMS, como la autenticación de usuarios, la carga de múltiples tipos de contenido, y la ejecución de consultas de búsqueda.

Ambos apartados se han realizado en conjunto, es decir, se ha modificado el archivo “P5-ADVANCED/P5-locust/locustfile.py” de forma que incorpora nuevas tareas como navegación por páginas, autenticación de usuarios, interacción con contenido, consultas de búsqueda e interacciones con la base de datos.

```

oMataNuriaP5/P5-ADVANCED/P5-locust
o ➔ cat locustfile.py
from locust import HttpUser, task, between
from bs4 import BeautifulSoup

class WordPressUser(HttpUser):
    wait_time = between(1, 3)

    def on_start(self):
        self.login()

    def login(self):
        # Simula inicio de sesión de un usuario
        response = self.client.post(
            "/login",
            data={"username": "admin", "password": "admin123"},
            verify=False,
            allow_redirects=True,
        )
        if response.status_code != 200:
            print("Error al iniciar sesión")

    @task(2)
    def visitar_pagina_principal(self):
        self.client.get("/", verify=False)

    @task(2)
    def navegar_articulos(self):
        articulo_id = 20
        self.client.get(f"/articulo/{articulo_id}", verify=False)

    @task(1)
    def crear_comentario(self):
        articulo_id = 2
        comentario = f"Comentario de prueba 2"
        self.client.post(
            f"/articulo/{articulo_id}/comentario",
            data={"comentario": comentario},
            verify=False
        )

    @task(1)
    def publicar_noticia(self):
        titulo = f"Noticia de prueba 2"
        contenido = "Contenido generado automáticamente para pruebas de carga."
        self.client.post(
            "/admin/publicar",
            data={"titulo": titulo, "contenido": contenido},
            verify=False
        )

    @task(1)
    def realizar_busqueda(self):
        consulta = "tecnología"
        self.client.get(f"/buscar?q={consulta}", verify=False)

    @task(1)
    def cargar_galeria(self):
        self.client.get("/galeria", verify=False)

    @task(1)
    def ver_perfil_usuario(self):
        user_id = 3
        self.client.get(f"/usuario/{user_id}/perfil", verify=False)

```

También he tenido que realizar un DockerFileLocust para que se instalen algunas dependencias y he modificado el docker-compose.yml para que use la construcción de la imagen en base a ese DockerFile:

```
2025-05-30 13:46:42 ✘ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5/P5-ADVANCED/P5-locust
└─o → cat DockerFileLocust
FROM locustio/locust

RUN pip install beautifulsoup4

2025-05-30 13:46:49 ✘ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5/P5-ADVANCED/P5-locust
└─o → cat docker-compose.yml
services:
  master-mmnuria:
    build:
      context: .
      dockerfile: DockerFileLocust
    image: mmnuria-locust:p5
    ports:
      - "8098:8089"
    volumes:
      - ./mnt/locust
    command: -f /mnt/locust/locustfile.py --master -H https://192.168.10.50:443/
    networks:
      p5-granjaweb_red_web:
        ipv4_address: 192.168.10.70

  worker-mmnuria:
    image: mmnuria-locust:p5
    volumes:
      - ./mnt/locust
    command: -f /mnt/locust/locustfile.py --worker --master-host master-mmnuria
    depends_on:
      - master-mmnuria
    deploy:
      replicas: 6
    networks:
      - p5-granjaweb_red_web

networks:
  p5-granjaweb_red_web:
    external: true
```

3.2. A2. Crear escenario multicontenedor con algún CMS

- Configurar contenedores Docker adicionales para cada instancia de un CMS seleccionado, como Drupal, WordPress, o Moodle, integrados con la granja web existente.

Para la realización de este apartado, he añadido un nuevo directorio y archivos “P5-ADVANCED/P5-wordpress/docker-compose.yml” donde se puede ver la correcta configuración de un contenedor WordPress y una base de datos, ambos integrados con la granja web:

```

2025-05-30 13:25:21 🏃 Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/Man
zanoMataNuriaP5/P5-ADVANCED/P5-wordpress
[O → cat docker-compose.yml

services:
  db:
    image: mysql:5.7
    container_name: wpdb-p5
    restart: always
    environment:
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wpuser
      MYSQL_PASSWORD: wppass
      MYSQL_ROOT_PASSWORD: rootpass
    networks:
      p5-granjaweb_red_servicios:
        ipv4_address: 192.168.20.101

  wordpress:
    image: wordpress:latest
    container_name: wordpress-p5
    restart: always
    environment:
      WORDPRESS_DB_HOST: db
      WORDPRESS_DB_NAME: wordpress
      WORDPRESS_DB_USER: wpuser
      WORDPRESS_DB_PASSWORD: wppass
    depends_on:
      - db
    ports:
      - "8099:80"
    networks:
      p5-granjaweb_red_servicios:
        ipv4_address: 192.168.20.100

networks:
  p5-granjaweb_red_servicios:
    external: true

```

- Asegurar la configuración adecuada del balanceador de carga para dirigir el tráfico hacia las instancias del CMS.

Para asegurar la correcta configuración del balanceador de carga he tenido que añadir en su configuración

“P5-ADVANCED/P5-granjaweb/P5-mmnuria-nginx/mmnuria-nginx-ssl.conf”

lo siguiente:

```

http {
    include mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
                    '$status $body_bytes_sent "$http_referer" '
                    '"$http_user_agent" "$http_x_forwarded_for"';

    # Grupo de servidores backend
    upstream backend_mmnuria {
        server 192.168.10.2;
        server 192.168.10.3;
        server 192.168.10.4;
        server 192.168.10.5;
        server 192.168.10.6;
        server 192.168.10.7;
        server 192.168.10.8;
        server 192.168.10.9;
        server 192.168.20.100;
    }
}

```

De esta forma el balanceador de carga puede dirigir el tráfico hacia las instancias del CMS.

- Establecer conexiones a bases de datos adecuadas y asegurarse de que todas las instancias del CMS puedan realizar operaciones de lectura y escritura de manera eficiente.

Este apartado se ha realizado en conjunto al anterior, ya que, toda la configuración para establecer las conexiones correctamente y las operaciones de lectura y escritura eficientemente, se han incluido en el mismo archivo “P5-ADVANCED/P5-wordpress/docker-compose.yml”

- Conectar base de datos a red_servicios de la granja.

Se puede comprobar la correcta configuración en la siguiente línea del archivo docker-compose.yml:

```

services:
  db:
    image: mysql:5.7
    container_name: wpdb-p5
    restart: always
    environment:
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wpuser
      MYSQL_PASSWORD: wppass
      MYSQL_ROOT_PASSWORD: rootpass
  networks:
    p5-granjaweb_red_servicios:
      ipv4_address: 192.168.20.101

```

3.3. A3. Ejecución y Análisis de cargas de prueba avanzadas sobre CMS

- Desplegar el escenario con el balanceador de carga y las instancias del CMS configuradas para las pruebas.

Para el despliegue muestro diferentes capturas de pantalla en el orden realizado:

1. Despliegue de la granja web con la modificación del archivo de configuración del balanceador de carga:

[+] Running 10/10		Started	2.5s
✓ Network p5-granjaweb_red_servicios	Created	0.1s	
✓ Container web3	Started	1.2s	
✓ Container web8	Started	1.1s	
✓ Container web5	Started	2.5s	
✓ Container web6	Started	1.7s	
✓ Container web1	Started	2.1s	
✓ Container web4	Started	1.8s	
✓ Container web7	Started	1.4s	
✓ Container web2	Started	2.0s	
✓ Container balanceador-nginx-ssl	Started	2.7s	

<input type="checkbox"/>	<input checked="" type="checkbox"/>	p5-granjaweb	-	-	-	0.06%	50 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web2	bee430dcbea	mmnuria-apache-im:	8083:443 ↗ Show all ports (2)	0%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web4	0ade4882af21	mmnuria-apache-im:	8087:443 ↗ Show all ports (2)	0%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	balanceador-nginx	9cae59a0646d	mmnuria-nginx-img	8097:443 ↗ Show all ports (2)	0%	50 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web8	76dbc97f003e	mmnuria-apache-im:	8095:443 ↗ Show all ports (2)	0.01%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web3	95adaba2a81b	mmnuria-apache-im:	8085:443 ↗ Show all ports (2)	0.01%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web7	35faf13b046e	mmnuria-apache-im:	8093:443 ↗ Show all ports (2)	0.01%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web6	4169e7ef56b9	mmnuria-apache-im:	8091:443 ↗ Show all ports (2)	0.01%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web1	a6b12c8aac56	mmnuria-apache-im:	8081:443 ↗ Show all ports (2)	0.01%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	web5	42957b48a360	mmnuria-apache-im:	8089:443 ↗ Show all ports (2)	0.01%	52 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>

2. Despliegue del contenedor WordPress y la base de datos creada:

2025-05-30 13:36:58 ☰ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5/P5-ADVANCED/P5-wordpress										
● o → docker compose up -d										
[+] Running 2/2										
✓ Container wpdb-p5 Starte... 0.5s										
✓ Container wordpress-p5 S... 0.7s										
<input type="checkbox"/>	<input checked="" type="checkbox"/>	p5-wordpress	-	-	-	0.2%	19 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	wpdb-p5	60578eaf9a46	mysql:5.7	-	0.19%	19 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	wordpress-p5	07b6f6c93e6d	wordpress:latest	8099:80 ↗	0.01%	19 seconds ago	<input type="button"/>	<input type="button"/>	<input type="button"/>

3. Despliegue del contenedor Locust que ejecuta las nuevas tareas incorporadas en el archivo locustfile.py:

2025-05-30 13:38:22 ☰ Nurias-MacBook-Pro in ~/Desktop/UNI/QUINTO/SEGUNDO CUATRI/SWAP/ManzanoMataNuriaP5/P5-ADVANCED/P5-locust										
● o → docker compose up -d										
[+] Running 7/7										
✓ Container p5-locust-master-mmnuria-1 Started 0.7s										
✓ Container p5-locust-worker-mmnuria-6 Started 1.6s										
✓ Container p5-locust-worker-mmnuria-1 Started 1.3s										
✓ Container p5-locust-worker-mmnuria-3 Started 2.3s										
✓ Container p5-locust-worker-mmnuria-4 Started 2.0s										
✓ Container p5-locust-worker-mmnuria-5 Started 2.9s										
✓ Container p5-locust-worker-mmnuria-2 Started 1.0s										

<input type="checkbox"/>	<input checked="" type="checkbox"/>	p5-locust	-	-	-	1.6%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	master-mmnuria-1 b33e794663ca	mmnuria-locust:p5	8098:8089	tcp	0.86%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-5 8cac006b7699	mmnuria-locust:p5			0.2%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-3 46bf0e31c0f9	mmnuria-locust:p5			0.15%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-6 5bf2fb12e8f1	mmnuria-locust:p5			0.07%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-1 f0e545b33805	mmnuria-locust:p5			0.11%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-2 a22123ac340b	mmnuria-locust:p5			0.1%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	worker-mmnuria-4 87d341b80f05	mmnuria-locust:p5			0.11%	2 minutes ago	<input type="button" value="..."/>	<input type="button" value="trash"/>

4. Accedo al contenedor WordPress para crear sesión como administrador y la inicio correctamente:

The screenshot shows a web browser window with the URL `localhost:8099/wp-admin/install.php?step=1`. The page is titled "Hola". It contains a form for installing WordPress. The fields are as follows:

- Título del sitio:** CMS Granja Web P5
- Nombre de usuario:** admin
- Contraseña:** EwoYgbZhrCo\$C^0Bdc (marked as Strong)
- Tu correo electrónico:** e.mmnuria@go.ugr.es
- Visibilidad en los motores de búsqueda:** Pedir a los motores de búsqueda que no indexen este sitio (unchecked)

At the bottom is a blue "Instalar WordPress" button.

- Para comprobar que está correctamente configurado y accesible al público, hago una petición al servidor y éste me devuelve la página inicial correctamente:



Blog

¡Hola, mundo!

Te damos la bienvenida a WordPress. Esta es tu primera entrada. Editala o bórrala, ¡luego empieza a escribir!

30 de mayo de 2025



- Lanzar pruebas de carga desde el contenedor master de Locust y coordinar a los workers para generar tráfico significativo hacia el CMS, evaluando la capacidad del sistema para manejar varias solicitudes simultáneas y operaciones de base de datos.

En este apartado muestro una serie de capturas de pantalla para que se pueda ver como se han lanzado las pruebas de carga desde el contenedor master. En los siguientes apartados se detallaran los resultados obtenidos:

localhost:8098

LOCUST

Host	https://192.168.10.50:443/	Status	READY	Workers	6	RPS	0	Failures	0%
------	----------------------------	--------	-------	---------	---	-----	---	----------	----

Start new load test

Number of users (peak concurrency) *

Ramp up (users started/second) *

Host

Advanced options

Run time (e.g. 20, 20s, 3m, 2h, 1h20m, 3h30m10s, etc.)

Profile

START

localhost:8098/?tab=stats

LOCUST

Host	https://192.168.10.50:443/	Status	SPAWNING	Users	7367	Workers	6	RPS	64.4	Failures	63%	EDIT	STOP	RESET
------	----------------------------	--------	----------	-------	------	---------	---	-----	------	----------	-----	-------------	-------------	--------------

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS WORKERS LOCUST CLOUD

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	//	2274	463	5200	7200	62000	5268.12	713	63370	179.19	34.5	9.6
GET	/galeria	1180	1180	5100	7200	8400	5143.13	826	63565	221.17	18.1	18.1
POST	/login	1481	1481	5800	59000	64000	12609.82	817	65839	264.42	11.8	11.8
Aggregated		4935	3124	5200	31000	62000	7441.49	713	65839	214.8	64.4	39.5

localhost:8098/?tab=workers

LOCUST

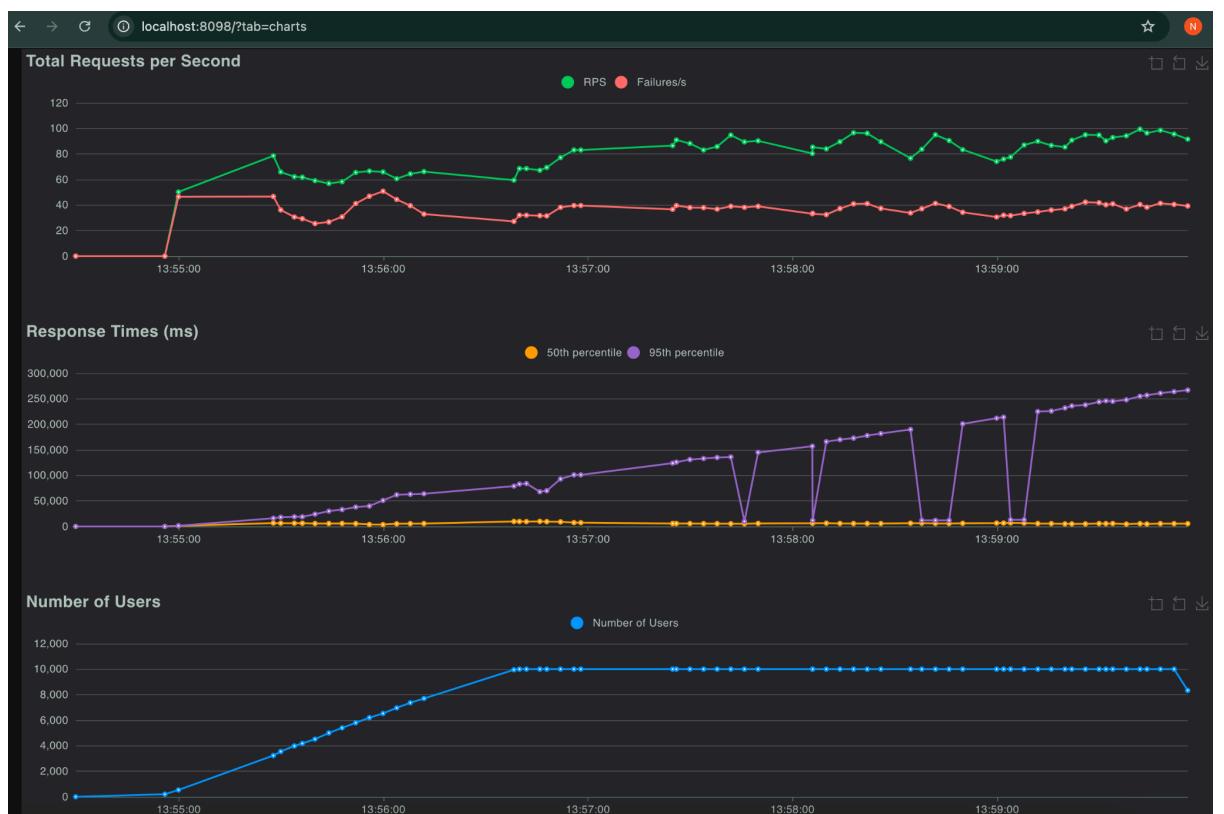
Host	https://192.168.10.50:443/	Status	RUNNING	Users	10000	Workers	6	RPS	95	Failures	49%	EDIT	STOP	RESET
------	----------------------------	--------	---------	-------	-------	---------	---	-----	----	----------	-----	-------------	-------------	--------------

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS WORKERS LOCUST CLOUD

Worker	State	# users	CPU usage	Memory usage
46bf0e31c0f9_9b54ade239a44e6f85db82ee79e5b71c	running	1667	100.4	324.36 MB
5bf2fb12e8f1_f624d4f1b061452fa94bec8c6ecd1de2	running	1667	100.7	318.52 MB
87d341b80f05_b3cee5cc2a0f45828acb61e93c551b7d	running	1667	100.3	326.7 MB
8cac006b7699_548f190f5b1e4e7a9677742b69a62d72	running	1667	100.3	331.14 MB
a22123ac340b_92c64c4783bc45c69984717826efcc6c	running	1666	100	330.09 MB
f0e545b33805_e7ee02c99755457690f0d46e9ae39ed7	running	1666	100.3	321.99 MB

- Analizar los resultados proporcionados por Locust, incluyendo el número de usuarios simultáneos que el sistema puede soportar, los tiempos de respuesta y la tasa de errores.
- Determinar los cuellos de botella y las limitaciones de rendimiento, y proporcionar recomendaciones para optimizar la configuración del CMS y la infraestructura de la granja web.

Para el análisis y la explicación detallada de estos dos apartados, los haré de forma conjunta, ya que, tienen bastante relación. Primero mostraré capturas de pantalla con información obtenida de los resultados proporcionados por Locust y al final haré un análisis global, detallando cada uno de los resultados.



localhost:8098/?tab=failures								
LOCUST		Host https://192.168.10.50:443/	Status RUNNING	Users 10000	Workers 6	RPS 94.8	Failures 53%	EDIT STOP RESET
STATISTICS		CHARTS	FAILURES	EXCEPTIONS	CURRENT RATIO	DOWNLOAD DATA	LOGS WORKERS LOCUST CLOUD	
# Failures Method Name Message								
475	GET	//	RemoteDisconnected('Remote end closed connection without response')					
334	GET	//	SSLError(SSLEOFError(8, '[SSL: UNEXPECTED_EOF_WHILE_READING] EOF occurred in violation of protocol (_ssl.c:1010)'))					
2841	GET	/galeria	HTTPError('404 Client Error: Not Found for url: /galeria')					
240	GET	/galeria	RemoteDisconnected('Remote end closed connection without response')					
179	GET	/galeria	SSLError(SSLEOFError(8, '[SSL: UNEXPECTED_EOF_WHILE_READING] EOF occurred in violation of protocol (_ssl.c:1010)'))					
2128	POST	/login	HTTPError('404 Client Error: Not Found for url: /login')					
57	POST	/login	SSLError(SSLEOFError(8, '[SSL: UNEXPECTED_EOF_WHILE_READING] EOF occurred in violation of protocol (_ssl.c:1010)'))					

localhost:8098/?tab=ratios								
LOCUST		Host https://192.168.10.50:443/	Status RUNNING	Users 10000	Workers 6	RPS 96.6	Failures 50%	EDIT STOP RESET
STATISTICS		CHARTS	FAILURES	EXCEPTIONS	CURRENT RATIO	DOWNLOAD DATA	LOGS WORKERS LOCUST CLOUD	
Ratio Per Class								
<ul style="list-style-type: none"> • 100.0% WordPressUser <ul style="list-style-type: none"> ◦ 11.1% cargarGaleria ◦ 11.1% crearComentario ◦ 22.2% navegarArticulos ◦ 11.1% publicarNoticia ◦ 11.1% realizarBusqueda ◦ 11.1% verPerfilUsuario ◦ 22.2% visitarPaginaPrincipal 								
Total Ratio								
<ul style="list-style-type: none"> • 100.0% WordPressUser <ul style="list-style-type: none"> ◦ 11.1% cargarGaleria ◦ 11.1% crearComentario ◦ 22.2% navegarArticulos ◦ 11.1% publicarNoticia ◦ 11.1% realizarBusqueda ◦ 11.1% verPerfilUsuario ◦ 22.2% visitarPaginaPrincipal 								

Como se puede observar en los resultados, se consigue mantener estable el volumen de usuarios simultáneos configurado en la prueba, que en este caso fue de 10.000. Esto indica que el sistema no colapsa por completo ante una carga elevada. No obstante, también se puede apreciar una tasa de errores considerablemente alta y persistente. Tal como se muestra en las capturas de pantalla, la mayoría de estos errores son 404, junto con alguno más, lo que probablemente se debe al planteamiento de las tareas definidas en el locustfile.py, ya que varias de las rutas utilizadas no existen en el contenedor de WordPress. Como mejora, sería recomendable ajustar dichas tareas para que trabajen con rutas reales, o bien añadir los endpoints necesarios a WordPress para que no generen estos errores.

En cuanto a los tiempos de respuesta, se observa que el percentil 95 supera los 300.000 ms, lo que significa que el 5% de las peticiones más lentas tienen una latencia extremadamente alta. Además, el percentil 50 comienza a elevarse significativamente a partir de los 6.000 usuarios, alcanzando valores aún mayores al llegar a los 8.000. Este comportamiento refleja claramente un cuello de botella en la infraestructura web, lo cual puede estar relacionado con una configuración ineficiente del balanceador de carga, que podría no estar escalando correctamente ante el incremento de usuarios.

Como recomendaciones a nivel del CMS, se sugiere revisar las tareas simuladas para eliminar o corregir aquellas que acceden a rutas inexistentes, así como considerar la configuración de plugins que ayuden a gestionar y redirigir correctamente dichas rutas. A nivel de infraestructura, una posible solución sería aumentar los recursos asignados a los contenedores de la granja web o implementar un sistema de escalado horizontal, añadiendo más instancias para distribuir la carga. Finalmente, en el archivo locustfile.py también se podrían implementar validaciones que eviten realizar pruebas sobre endpoints no válidos, además de establecer una estrategia de aumento progresivo de usuarios para prevenir saturaciones bruscas.

3.4. Análisis propuesta IA

En este apartado trato de analizar los resultados propuestos por la IA de este segundo apartado para las tareas avanzadas. He preguntado a la IA escribiendo directamente el enunciado de las diferentes tareas para que me mostrara paso a paso lo que debe de incluir cada una, [pulsa aquí para acceder al chat](#)

De forma resumida, la IA me ha proporcionado soluciones bastante útiles:

- Tarea A1: Las sugerencias de la IA en este apartado me han resultado bastante útiles. Me sirvieron como base y me ayudaron a entender mejor cómo se construyen las tareas en Locust. Aun así, mi archivo final no es exactamente igual, ya que he añadido tareas nuevas y eliminado otras que no me parecían tan relevantes o necesarias.
- Tarea A2: En este caso me resultó un poco más confusa la solución que propuso la IA, así que decidí no seguirla. Preferí trabajar directamente con los contenedores ya creados de la granja web, haciendo algunas modificaciones como ajustar el balanceador de carga para que tenga en cuenta el nuevo contenedor de WordPress que añadí previamente, el cual está conectado a una base de datos. Considero que no era necesario volver a crear contenedores de Locust, como sugería la IA, porque ya los habíamos configurado antes y así se puede comprobar bien que todo esté funcionando y conectado correctamente.

- Tarea A3: Esta parte solo consiste en comprobar y analizar la ejecución de los apartados anteriores, así que no he recurrido a la IA, ya que es algo bastante sencillo.

Bibliografía:

- Apache Software Foundation. (s.f.). *Apache HTTP server benchmarking tool (ab)*. <https://httpd.apache.org/docs/2.4/programs/ab.html>
- Locust.io. (s.f.). *Locust - Scalable user load testing tool*. <https://locust.io>
- Python Software Foundation. (s.f.). *Python requests library documentation*. <https://docs.python-requests.org/en/latest/>
- Soto Hidalgo, José Manuel. (2025). *Práctica 5: Benchmarking y Rendimiento de la Granja Web*. (Material didáctico proporcionado).