



SI657 – FUNDAMENTOS DE ARQUITECTURA DE SOFTWARE

2024-02

PRACTICA CALIFICADA 2

Sección: SW72
Profesores : JORGE DELGADO

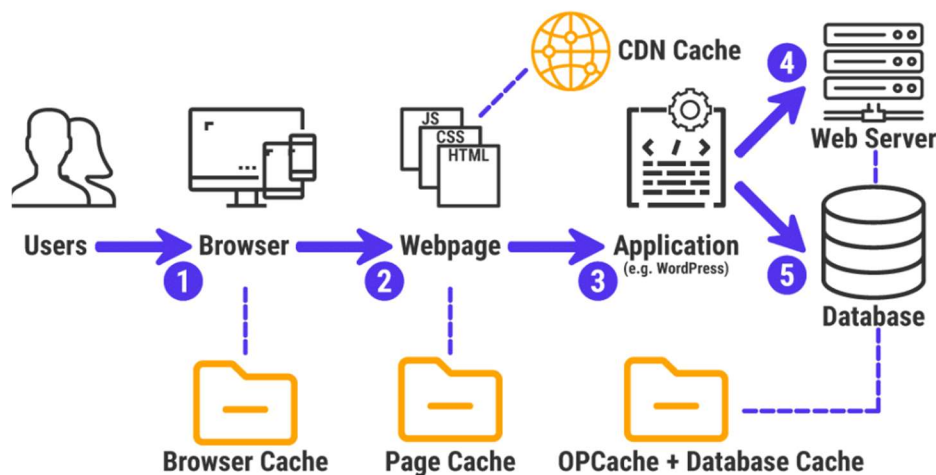
Duración : 110 minutos
Indicaciones :

Business Case:






Uso de Cache para aplicaciones distribuidas

Redis cache es una de las herramientas de almacenamiento en caché más populares utilizadas para acelerar los sitios web, de forma rápida, altamente disponible, resistente y escalable que se extiende a través de la nube.

Con el almacenamiento en caché, los datos almacenados en bases de datos más lentas pueden lograr un rendimiento inferior al milisegundo. Eso ayuda a las empresas a responder a la necesidad de aplicaciones en tiempo real. Redis está diseñado para el almacenamiento en caché a gran escala. Su funcionalidad garantiza que las aplicaciones críticas se ejecuten de manera fiable y súper rápida, al tiempo que brinda integraciones para simplificar el almacenamiento en caché y ahorrar tiempo y dinero.



A fin de proporcionar información necesaria a gran velocidad para un servicio, se ha generado aplicaciones en Python, Java y Node para cargar información en un cache Redis.

Nombre	
 appRedis	
 dependencias	
 Faker	
 Redis2Java.jar	
 Redis2Java.java	

Aplicaciones principales

Faker.zip: Proyecto en eclipse que genera registros json usando librería faker.

appRedis: Aplicación Python que graba información en Redis desde Python, Redis debe estar levantado y ejecutando en el puerto 6379.

RedisPcJava: Aplicación en java que graba o lee información en Redis.

Implementar el siguiente escenario evidenciando los pasos necesarios para realizarlo:

- Levantar en un contenedor de Docker Desktop la imagen “redis” en el puerto 6379
- Usando la aplicación Faker generar 100 registros JSON de trabajadores.
- Usando appRedis.py cargar los 100 registros JSON al cache redis en el puerto 6379.
- Usando NodePc2 0 Redis2Java.java leer los 100 registros de Redis en el puerto 6379.

Posteriormente la pregunta 3 pide generar un servicio REST que lea la información de **Redis**

Módulo de publicación de la información

Implemente un servicio en Node, Python o Java que tenga un endpoint: **/worker** que muestre los 100 registros leídos desde redis

Para el caso mencionado se pide:

Pregunta 1 (Elaborar el entorno de ejecución solicitado 8)

Configure e implemente el entorno de ejecución solicitado evidenciando la ejecución de todos los componentes mencionados

Pregunta 2 (Diagrama de contenedores 7 p.)

Elabore un diagrama de contenedores C4 para el caso planteado

Pregunta 3 (Código de un servicio 5 ptos.)

Implemente el código necesario para el **servicio** que lea el cache Redis y muestre un endpoint **/worker** con los 100 registros solicitados.

**DOCUMENTE ADECUADAMENTE LOS PASOS REALIZADOS, LEVANTAR Y CONSUMIR
SERVICIOS, PROBAR SERVICIOS, ETC.**

8/11/2024