



VibraGaming

Challenge Plataforma v0.4a

Requisitos

Desarrollar una aplicación con un API que tenga un método GET para búsqueda en un archivo CSV* por querystring de los campos (opcionales) y un parámetro adicional para límite de resultados.

Esta búsqueda se debería realizar asincrónicamente, en algún servicio que sea parte del sistema, se pueden usar colas de mensajería para procesar estas peticiones. La idea es que si hay un reinicio de los servicios el sistema asegure que la petición se procese.

La respuesta de la búsqueda debería ser asincrónica con alguna tecnología para responder esa llamada. Opciones

- un webhook como argumento de la llamada
- un id de transacción que permita hacer un polling al sistema
- Un canal de websocket donde haya un método de búsqueda y la respuesta llegue cuando se finalizó.
- SSE o Push notifications

Es requisito armar en docker (con docker compose) los servicios necesarios y describir su funcionamiento.

*El archivo CSV se puede pre-cargar en el setup del proyecto en alguna base de datos, esto es opcional

Ejemplo de query:

http://<host>:<port>/search?name=<string>&city=<string>&quantity=<int>

Se puede utilizar la tecnología que resulte más familiar (Flask, Django Rest API, Express with node Js, etc).



VibraGaming

Recomendamos partir de un fork de algún **skeleton** según la tecnología escogida.

Recursos utiles

- [Python flask with docker skeleton](#) (proyecto base, puede ser otro fork según tecnología o preferencia)
- [Csv a utilizar](#)

Se valorará también

- Uso de repositorio GIT, con commits describiendo la historia del desarrollo.
- Test unitarios de caso de uso.
- Diseño y buenas prácticas.
- Uso de Docker para desarrollo local. Con instrucciones de ejecución.



VibraGaming

Otras valoraciones extra de nuestro lado

(No advertir a los participantes)

Preguntas que nos hagan sobre arquitectura del desarrollo.

Domain Driven Development (DDD)

Escalabilidad