**PRUEBA DE RENDIMIENTO HU-16**

**Pasos que sigue:**

Caso positivo: en primer lugar va a home, se loguea como reader y va a la lista de recomendaciones y ve una página con recomendaciones.

Caso negativo: va a home, se loguea como admin y va a la lista de recomendaciones y ve una página con un mensaje de que la pagina no tiene libros para recomendarle.

**Stress test:** El número minimo de usuarios concurrentes son 105.000 en 10 segundos, termina el script sin romper el sistema, con la CPU al 100% la mayoría del tiempo, por lo que la CPU es el cuello de botella, y la memoria también se usa demasiado.

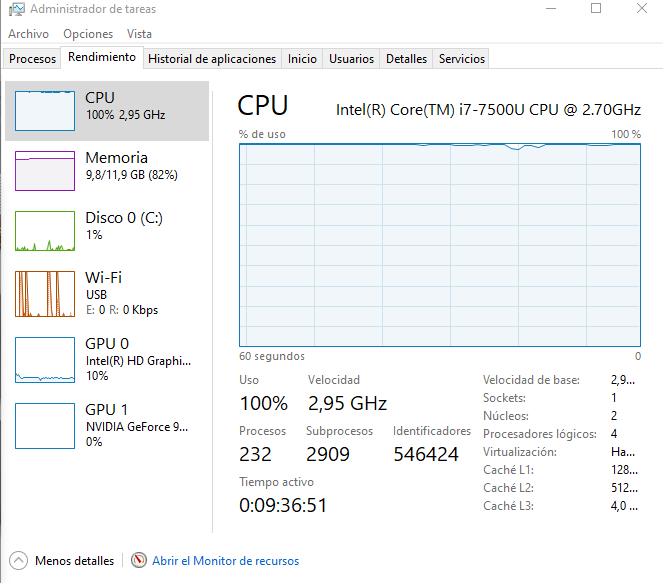
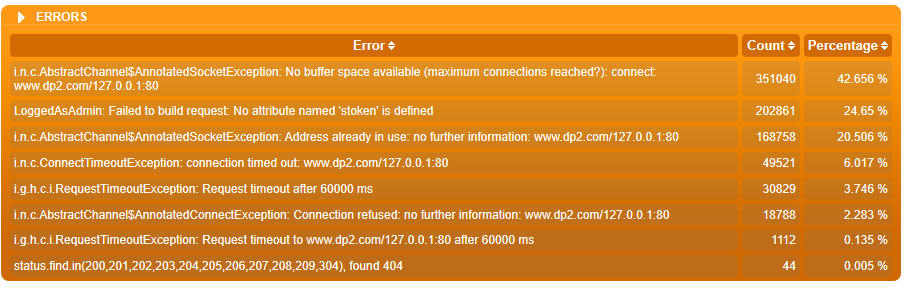
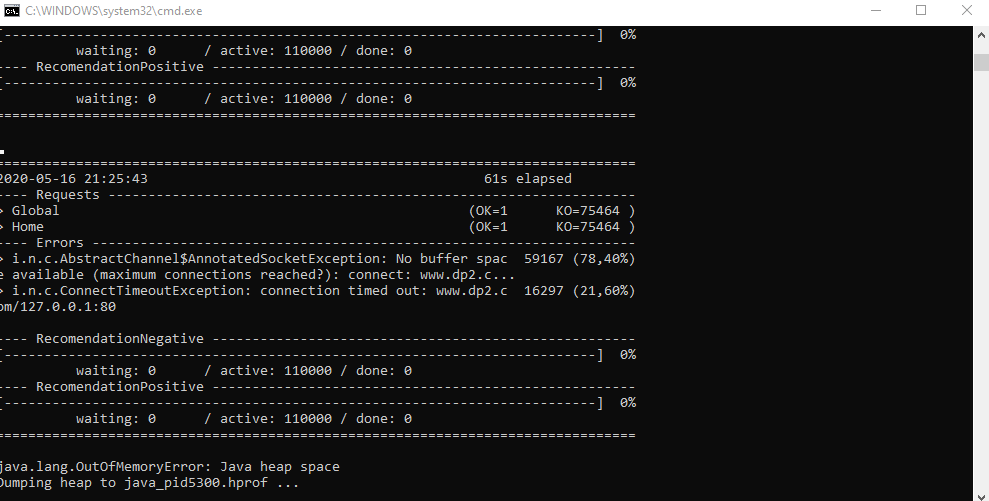


Ilustración X: cuello de botella HU-16

Y aquí podemos ver una lista de errores



Como vemos con 110.00 usuarios la aplicación se rompe:



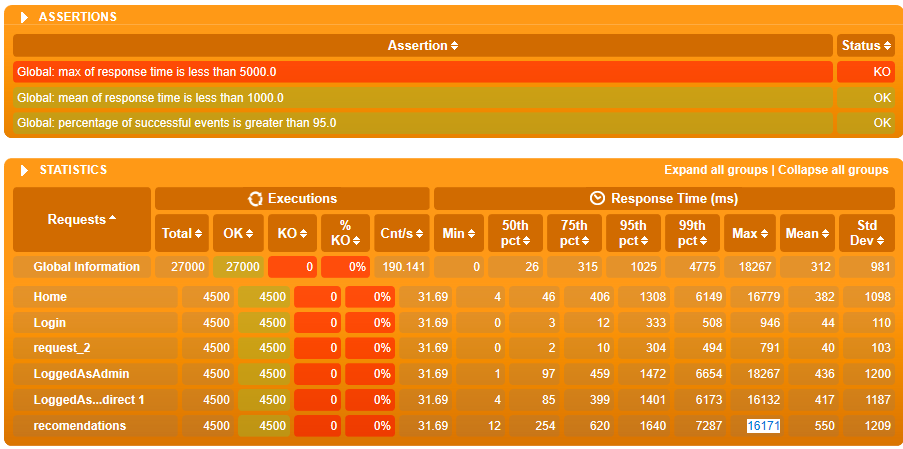
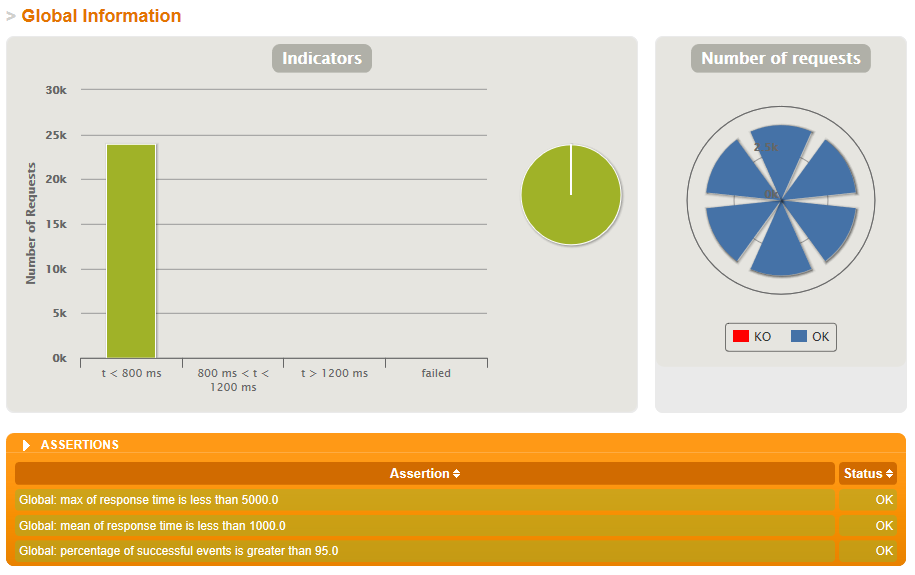
**Load Test:** con 2250 usuario por escenario repartidos en 100 segundos, vemos que tiene una conducta bastante aceptable pero hay varias peticiones que superan los 5 segundos siendo 18 segundos el máximo.

Ilustración X: test de carga HU-16

Con 2000 usuario concurrentes por escenario se cumplen ya todas las restricciones.

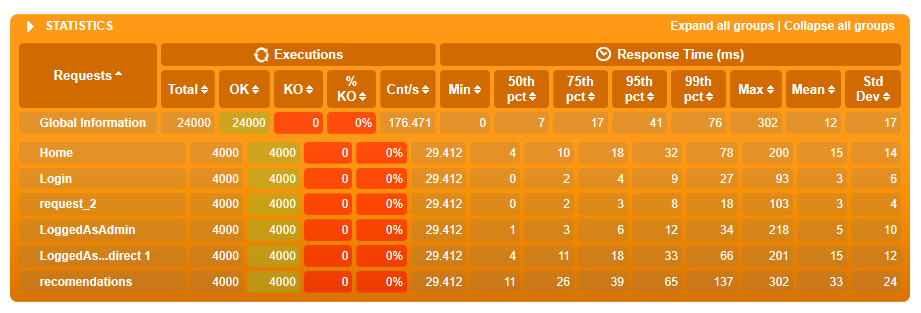


Ilustración X: test de carga HU-16

Podemos ver que el máximo numero de usuarios activos es sobre los 1450.

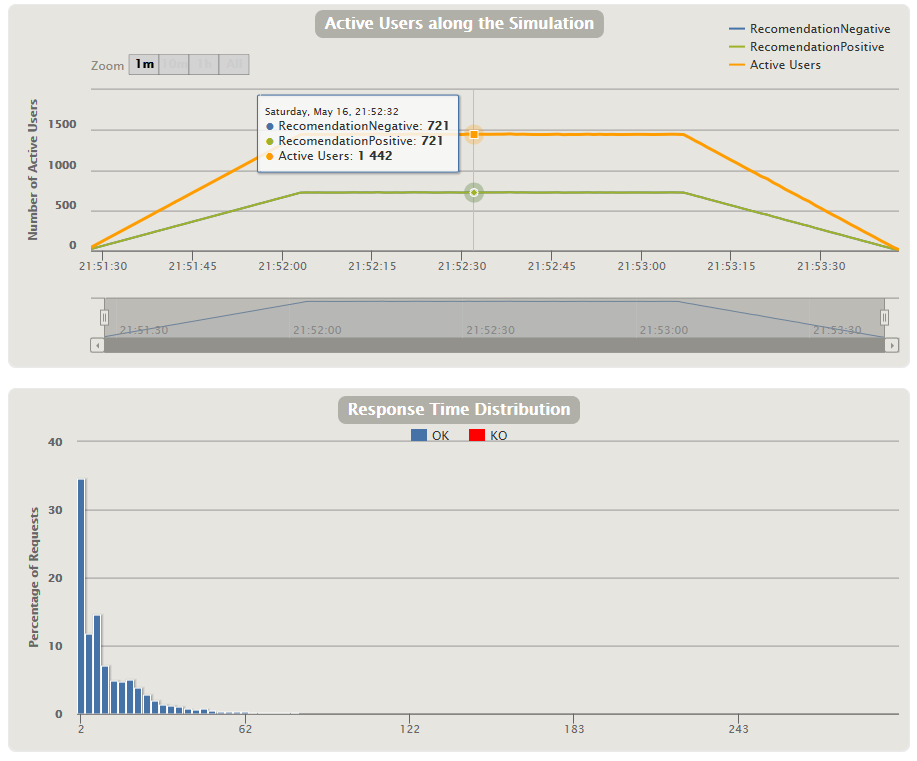
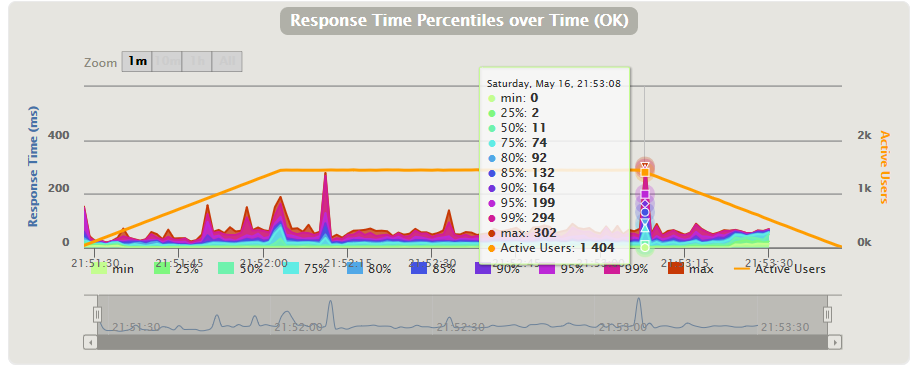


Ilustración X: test de carga HU-16



Aquí vemos el número de peticiones por segundo es practicamente igual al número de respuestas por segundo

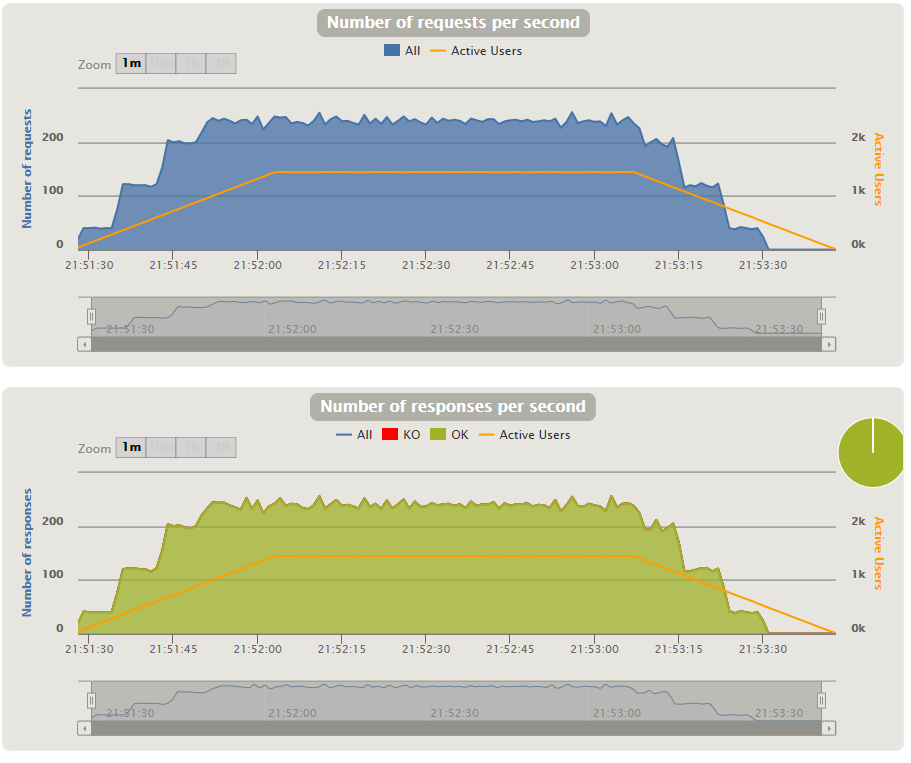
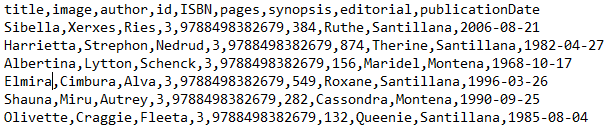


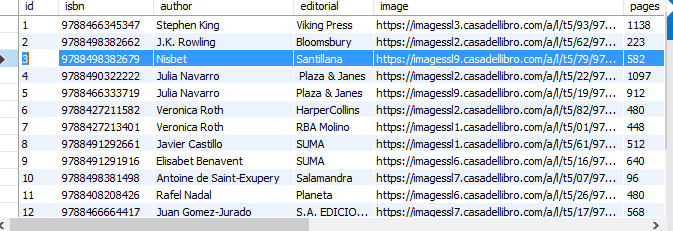
Ilustración X: test de carga HU-16

**PRUEBA DE RENDIMIENTO HU-02**

**Csv:** Se ha usado dos csv para editar un mismo libro con diferentes datos. Uno por excenario(ejemplo de primeros casos).



Y podemos ver los cambios en la base de datos



**Pasos que sigue:**

Caso positivo: en primer lugar va a home, se loguea como admin y va al buscador de libros donde se puede ver un poema, obtiene la lista de libros, se mete en un libro y pulsa editar, hace los cambios y le da a aceptar.

Caso negativo: es igual que el caso positivo pero en este caso mete errores en ciertos campos.

**Stress Test:**  90.000 usuarios por escenario es el número máximo que soporta la aplicación sin romperse.

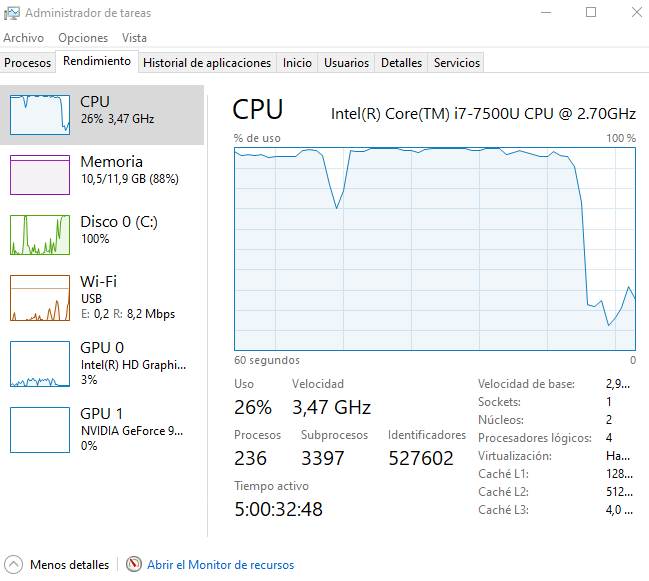
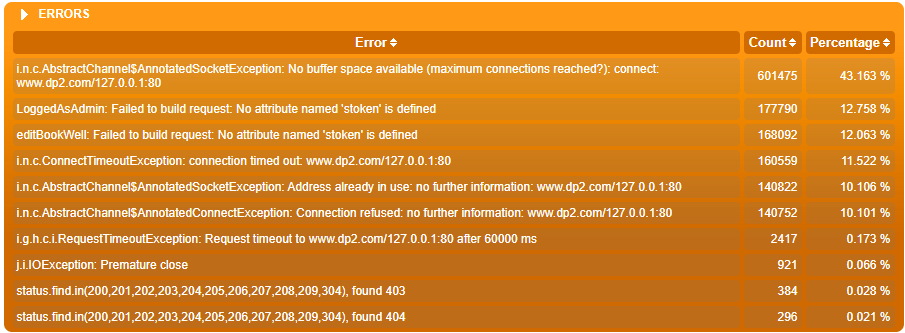


Ilustración X: cuello de botella HU-16

Aquí podemos ver una lista de errores

**Load Tests:** Debido a que esta HU también pasa por la API ocurre el problema ya comentado al principio, por lo que el rendimiento también es bastante pobre.

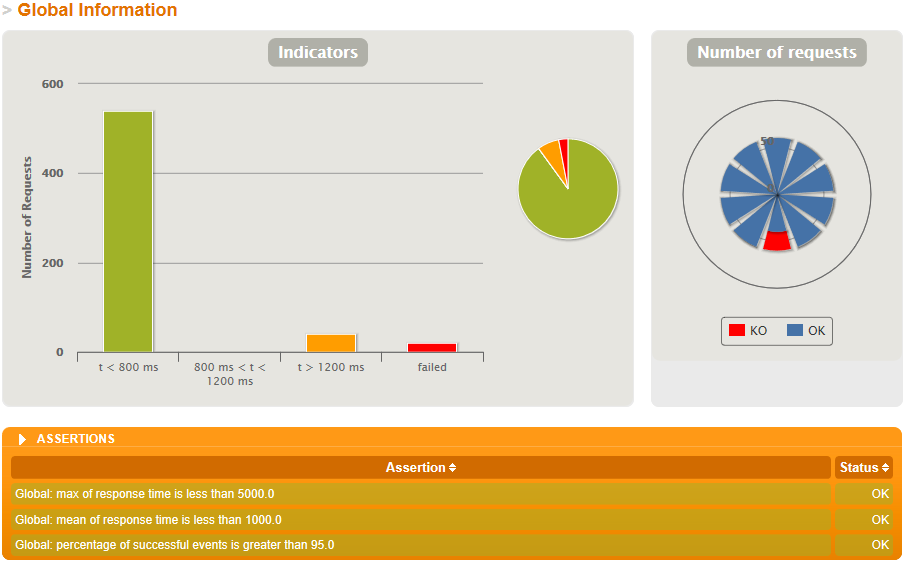
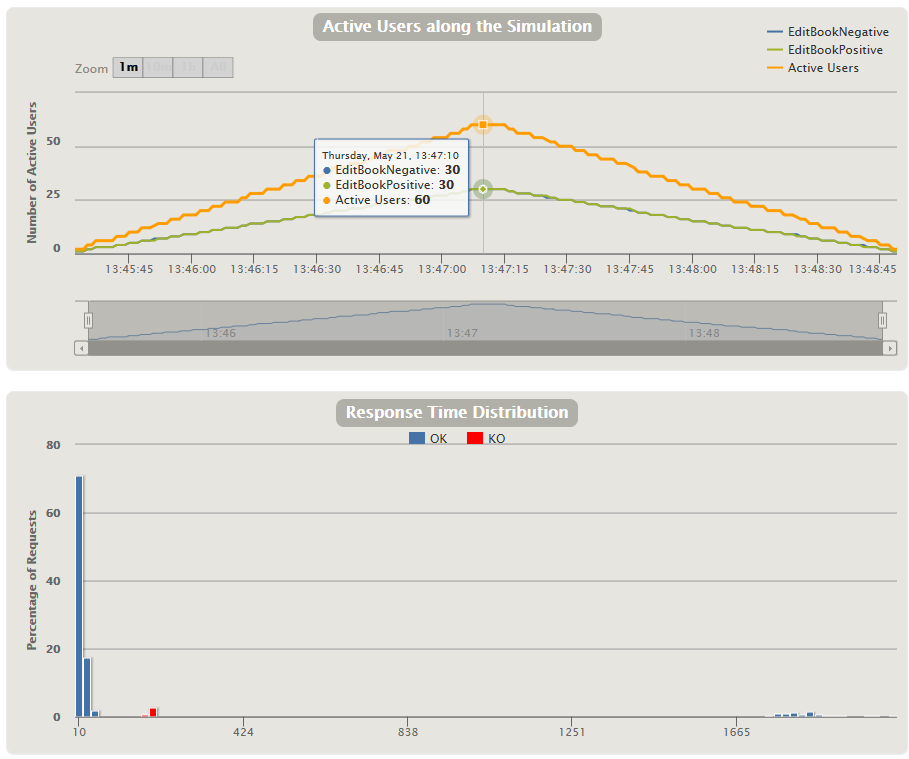
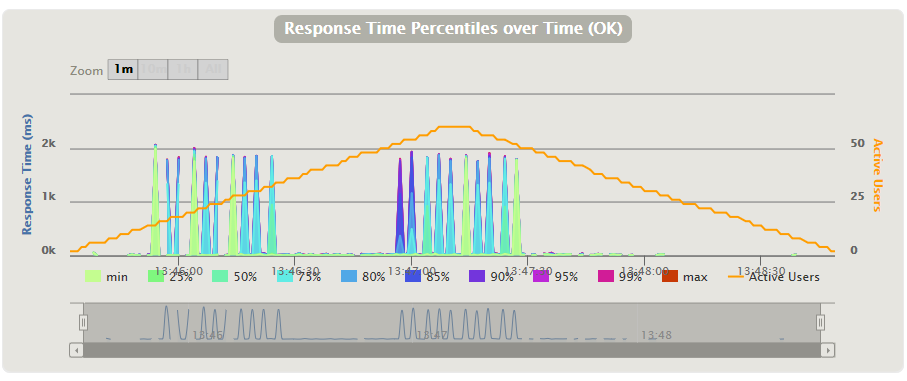


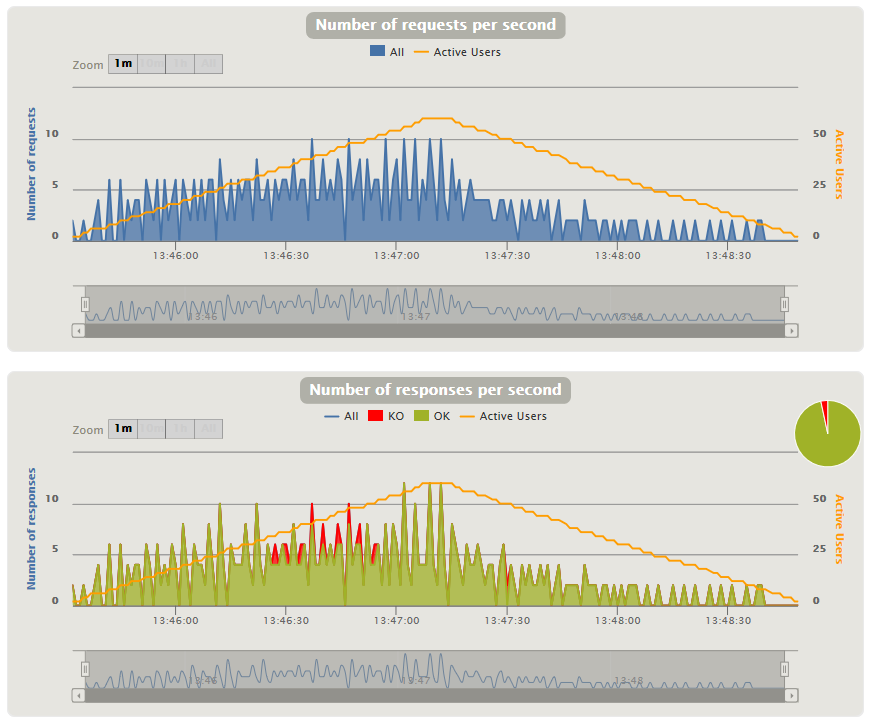


Ilustración X: test de carga HU-02

Aquí vemos que el pico de usuarios activos es 30 por escenario. Y que hay algunas respuestas KO, debido a la API.

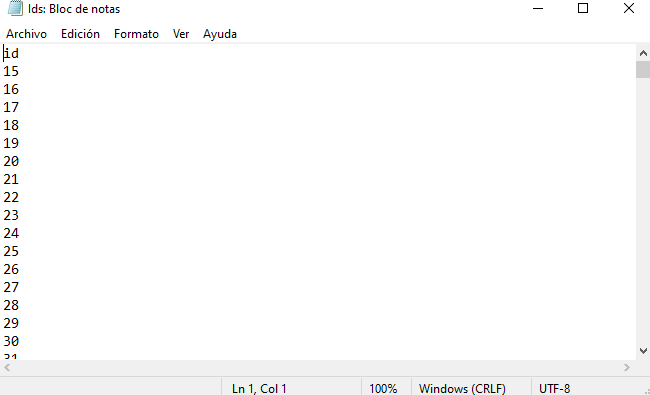






**PRUEBA DE RENDIMIENTO HU-13**

**Csv:** Se usa el mismo csv que para HU-01 pero adicionalmente 3 csv con Ids para ir añadiendo el libro que se va creando a leidos con la siguiente forma.



**Pasos que sigue:**

Caso positivo: va a home, se loguea, va al buscador de libros donde hace la llamada a la API, obtenemos lista de libros creamos un nuevo libro y añadimos ese a la lista de libros leidos.

Caso negativo: sigue los mismos pasos pero cuando obtiene la lista de libros se mete en uno que ya esta en su lista y lo intenta añadir a libros leido.

**Stress test:** Determinamos el número máximo de usuarios concurrentes en 95.000 en escenario, aquí podemos ver la CPU al límite, la memoria muy cerca del límite y el disco también tiene picos.

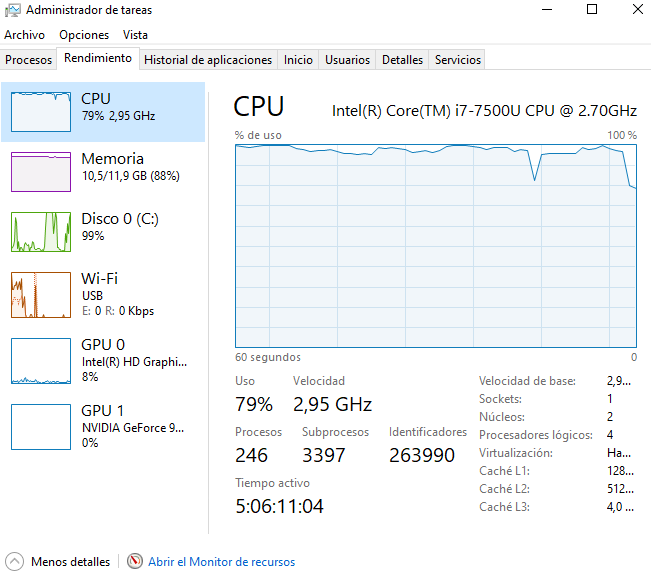
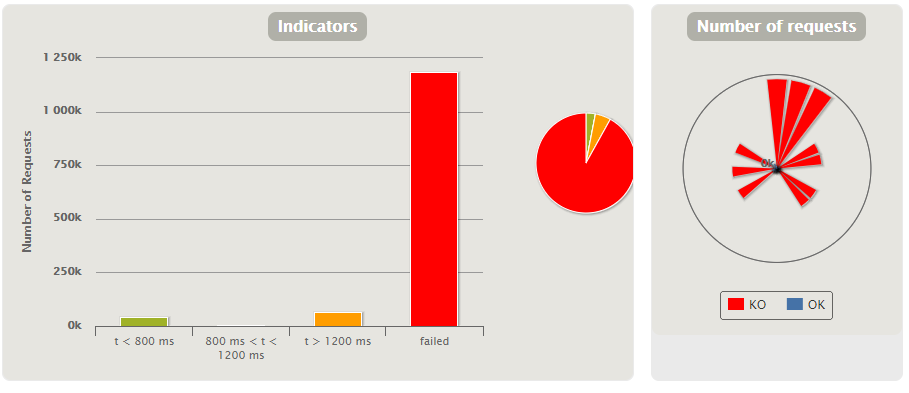


Ilustración : cuello de botella HU-13



**Load Test:** Este caso vuelve a pasar por la API por lo que el rendimiento vuelve a ser muy pobre con 40 usuarios por escenario.

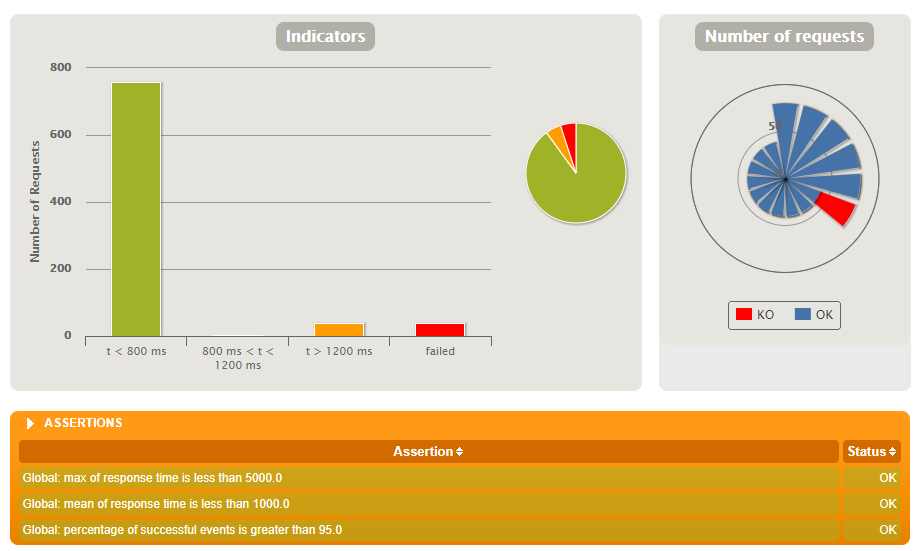
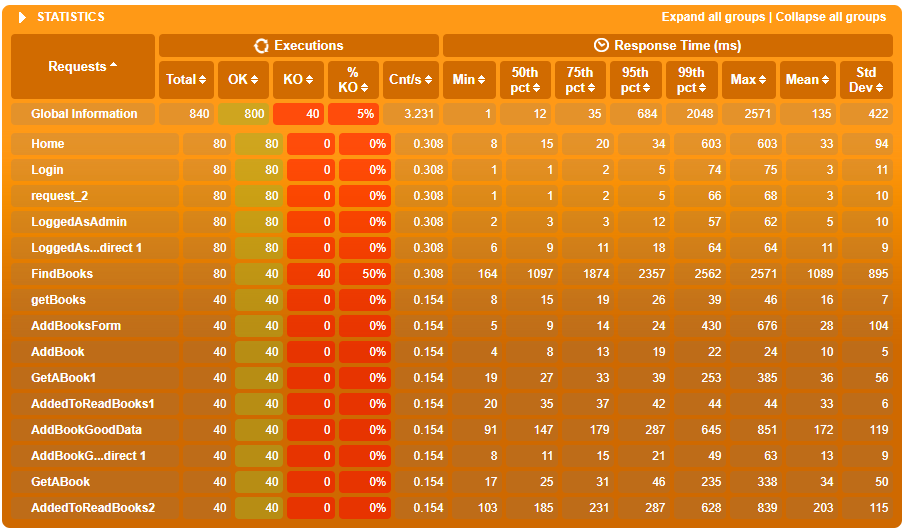


Ilustración : test de carga HU-13



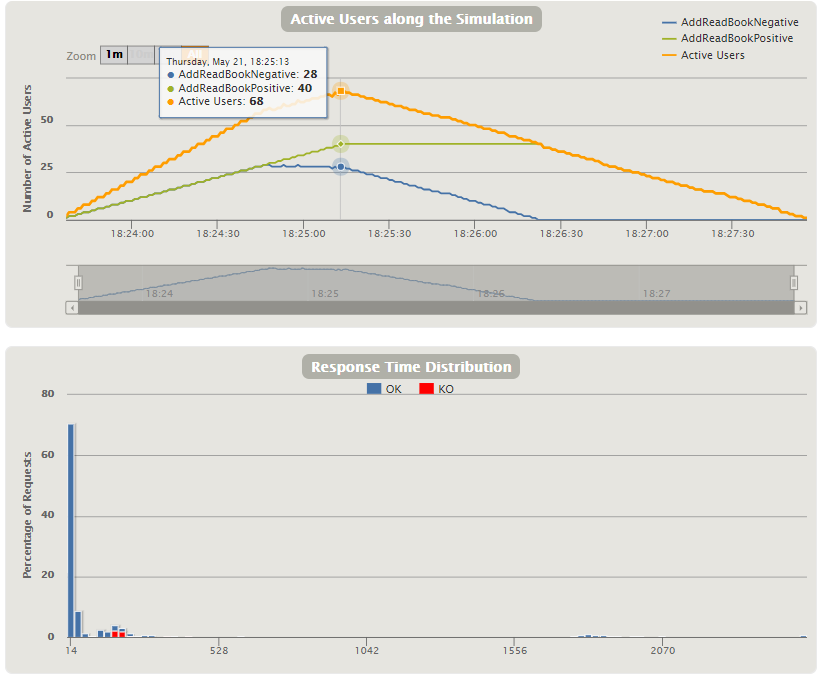
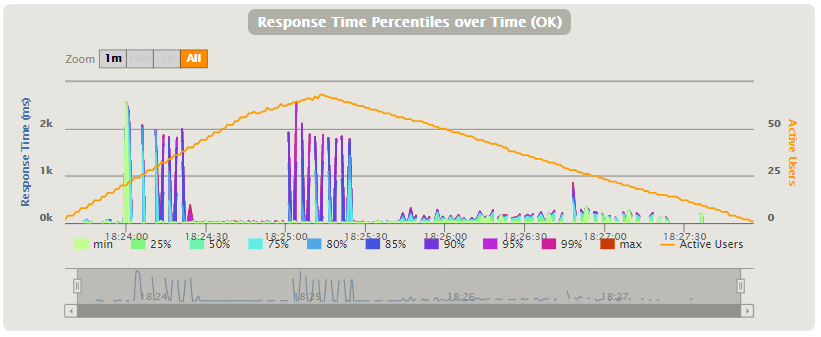
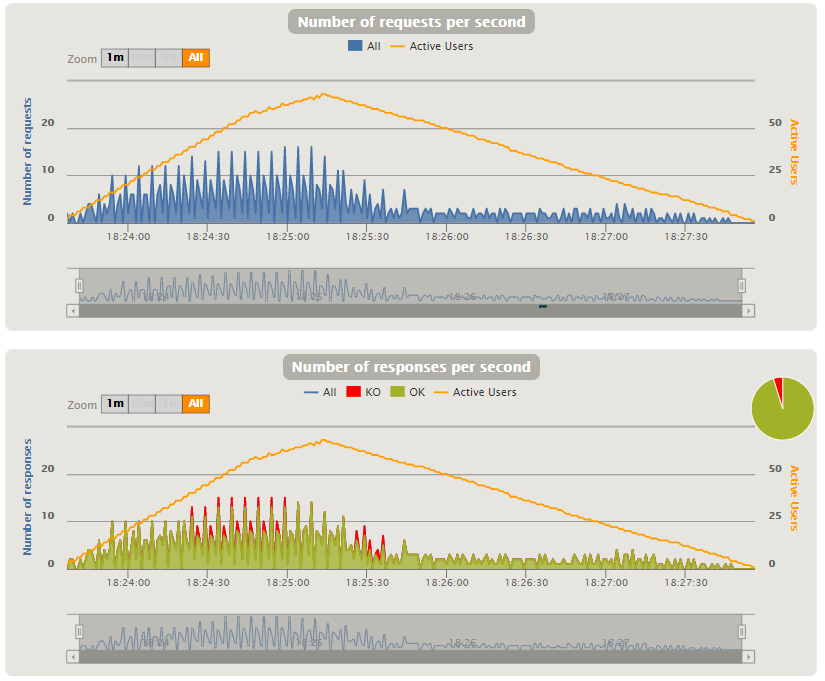


Ilustración X: test de carga HU-13





**PRUEBA DE RENDIMIENTO HU-09 y HU-10**

Dado que estas HU están muy ligadas la una a la otra hemos tomado la decisión de probarlas juntas con 3 escenarios.

**Pasos que sigue:**

Caso positivo HU-09 y 10: va a home, se loguea, va al buscador de meetings donde se encuentra una api de frases, se obtiene una lista de meetings, se inscribe a una meeting, vuelve a obtener esa meeting y se desuscribe.

Caso negativo HU 10: va a home se loguea, busca las meetings y se intenta descuscribir de una meeting a la que no está suscrito.

Caso negativo HU-09: hace igual que en la anterior pero intenta inscribirse a una a la que ya está suscrito.

**Stress test:** La máxima concurrencia de usuarios son 65.000 por escenario.

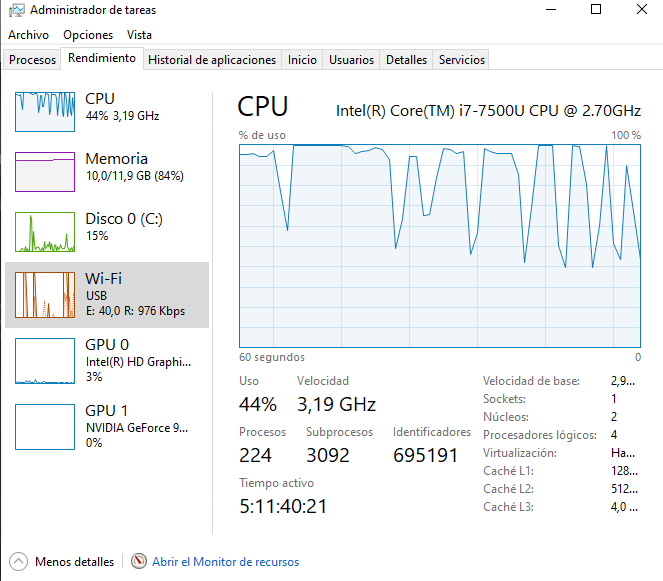


Ilustración X: cuello de botella HU-06

