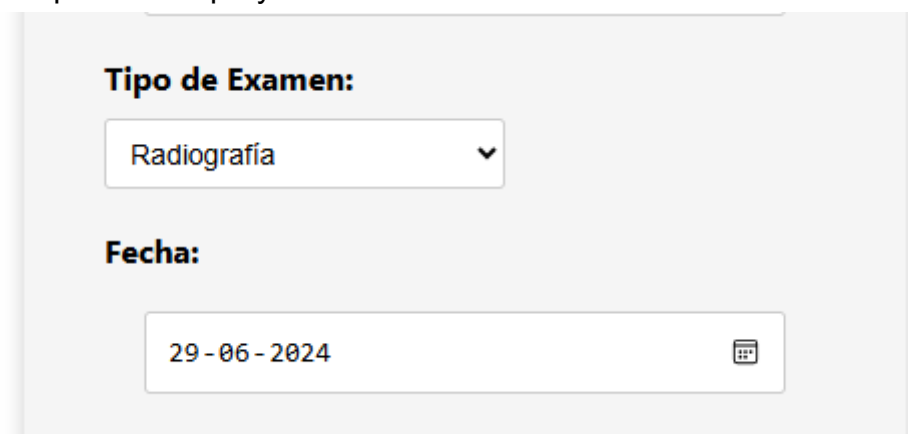


Pruebas de Performance en Apache Jmeter.

Haciendo uso de la aplicación Apache Jmeter hemos diseñado 3 pruebas para distintos endpoints del proyecto, específicamente nos enfocamos en la historia de usuario “registrar hora médica” y “Buscar hora registrada”.

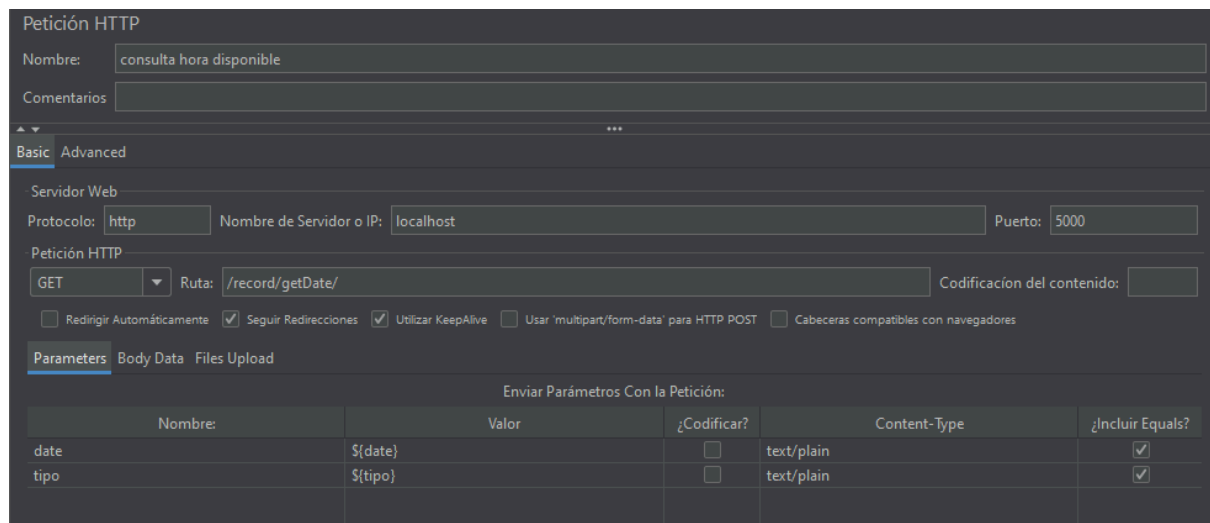
Para la primera historia de usuario creamos 2 pruebas, una para la consulta de horas disponibles según el tipo de exámen, la cual hace una consulta de tipo GET a la base de datos, por otra parte hicimos una prueba de tipo POST para el agendamiento de horas médicas.

Vista del endpoint en el proyecto:



The screenshot shows a web form with two main sections. The first section is titled "Tipo de Examen:" and contains a dropdown menu with "Radiografía" selected. The second section is titled "Fecha:" and contains a date input field with "29-06-2024" entered, accompanied by a calendar icon.

Aquí podemos ver el desarrollo de la prueba en Apache Jmeter.



The screenshot shows the Apache Jmeter configuration interface for an HTTP request. The "Nombre:" field is set to "consulta hora disponible". The "Comentarios" field is empty. The "Basic" tab is selected, showing the "Servidor Web" section with "Protocolo:" set to "http", "Nombre de Servidor o IP:" set to "localhost", and "Puerto:" set to "5000". The "Petición HTTP" section shows the "Método:" set to "GET" and the "Ruta:" set to "/record/getDate/". The "Codificación del contenido:" field is empty. Below this, there are checkboxes for "Redirigir Automáticamente", "Seguir Redirecciones", "Utilizar KeepAlive", "Usar 'multipart/form-data' para HTTP POST", and "Cabeceras compatibles con navegadores". The "Parameters" tab is selected, showing a table for "Enviar Parámetros Con la Petición:".

Nombre:	Valor	¿Codificar?	Content-Type	¿Incluir Equals?
date	\${date}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
tipo	\${tipo}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Vista del endpoint de la segunda prueba:

Registrar Hora para Examen Médico

Nombre:

Apellido:

RUT (sin guión ni puntos):

Tipo de Examen:

Radiografía

▼

Fecha:

29-06-2024

📅

Hora:

12:30:00

▼

Nombre del Médico:

Observación del Examen:

nada

Registrar Hora

Vista del desarrollo de la segunda prueba:

Petición HTTP

Nombre: HTTP Request

Comentarios

BasicAdvanced

Servidor Web

Protocolo: http

Nombre de Servidor o IP: localhost

Puerto: 5000

Petición HTTP

POST

▼

Ruta: /record/add

Codificación del contenido:

☐ Redirigir Automáticamente

☒ Seguir Redirecciones

☒ Utilizar KeepAlive

☐ Usar 'multipart/form-data' para HTTP POST

☐ Cabeceras compatibles con navegadores

ParametersBody DataFiles Upload

1 {

2 "nombre": "\${nombre}",

3 "apellido": "\${apellido}",

4 "rut": "\${rut}",

5 "tipo_examen": "\${examen}",

6 "fechahora": "\${fecha}",

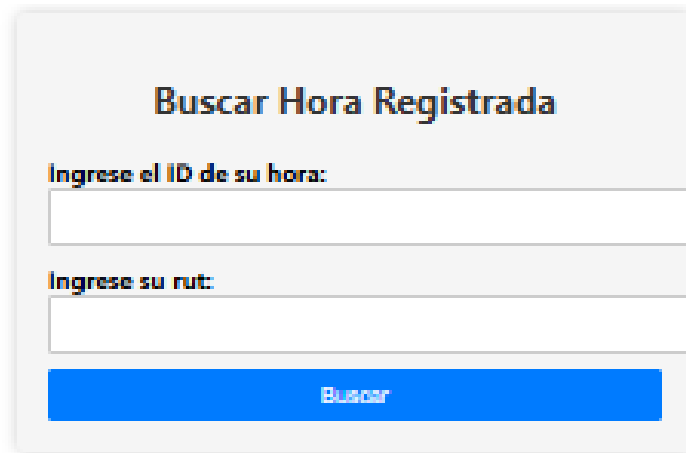
7 "medico": "\${medico}",

8 "observaciones": "\${obs}"

9 }

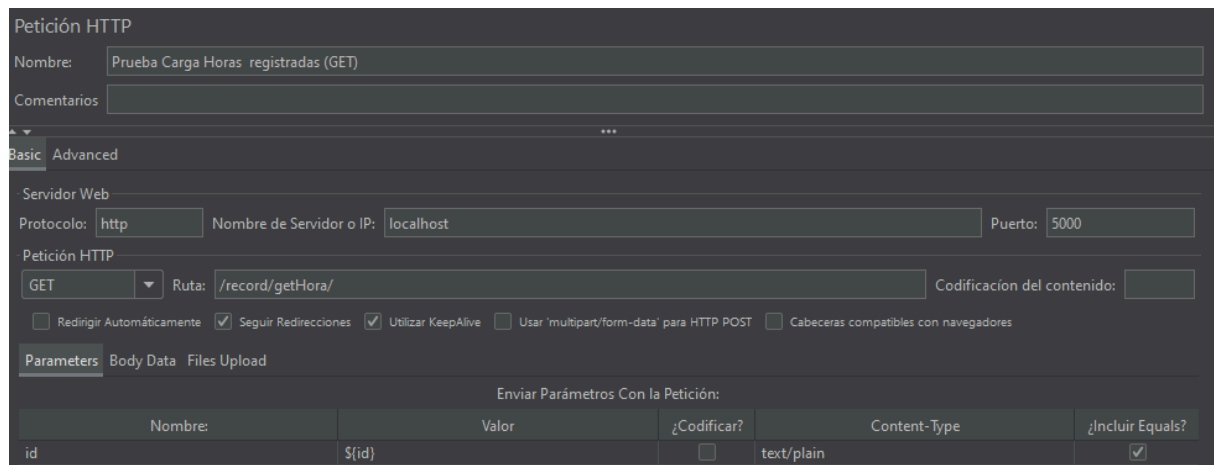
Pasando a la prueba realizada en la aplicación de la HU Buscar ahora registrada, creamos un archivo csv que contiene id's de horas médicas a modo de prueba, en el archivo agregamos id's de horas que hemos sacado de la base de datos y otras con errores para así comprobar que el caso de error también estuviera funcionando.

Vista del endpoint.



Formulario de búsqueda de hora registrada. El título es "Buscar Hora Registrada". Hay dos campos de entrada: "Ingrese el ID de su hora:" y "Ingrese su rut:". Debajo de los campos hay un botón azul que dice "Buscar".

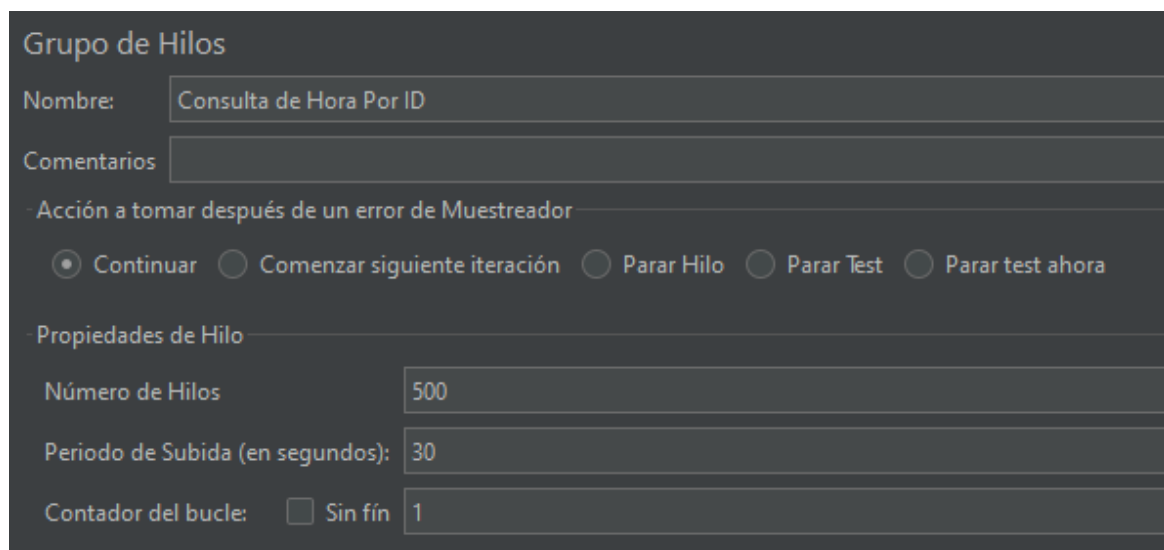
Vista del desarrollo en Apache Jmeter:



Vista del desarrollo en Apache Jmeter. Se muestra la configuración de una petición HTTP. El nombre es "Prueba Carga Horas registradas (GET)". El protocolo es "http", el nombre de servidor o IP es "localhost" y el puerto es "5000". La petición HTTP es "GET" y la ruta es "/record/getHora/". Se muestran las opciones de redirección y keep-alive. En la sección de parámetros, se define un parámetro "id" con el valor "\${id}", no se codifica, el tipo de contenido es "text/plain" y se incluye el signo igual.

Nombre:	Valor	¿Codificar?	Content-Type	¿Incluir Equals?
id	\${id}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Datos de los threads:



Grupo de Hilos. Nombre: "Consulta de Hora Por ID". Comentarios: (vacío). Acción a tomar después de un error de Muestreador: Continuar (seleccionado). Propiedades de Hilo: Número de Hilos: 500, Periodo de Subida (en segundos): 30, Contador del bucle: Sin fin (desseleccionado), 1.

Datos obtenidos con la prueba:

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo

Navegar...

Log/Mostrar sólo:

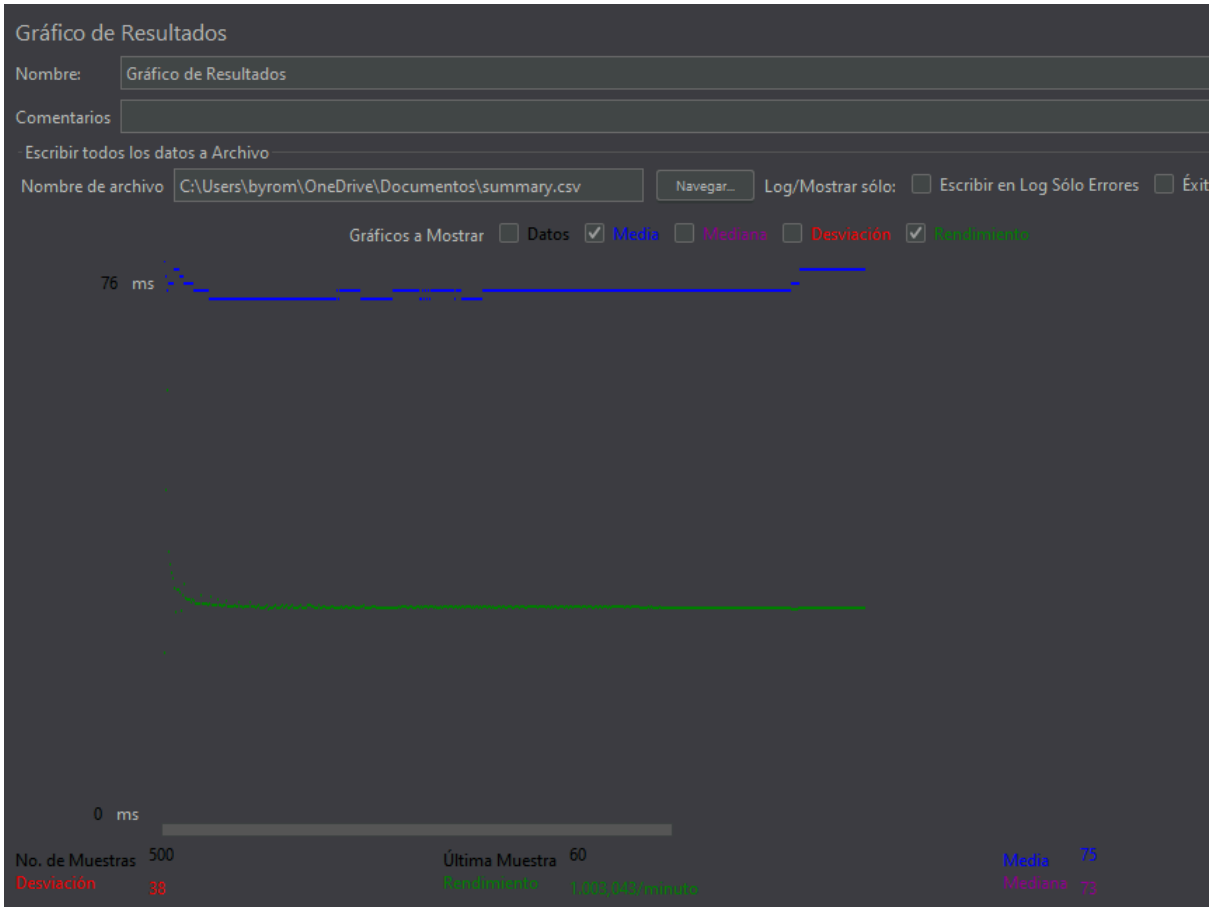
Escribir en Log Sólo Errores

Éxitos

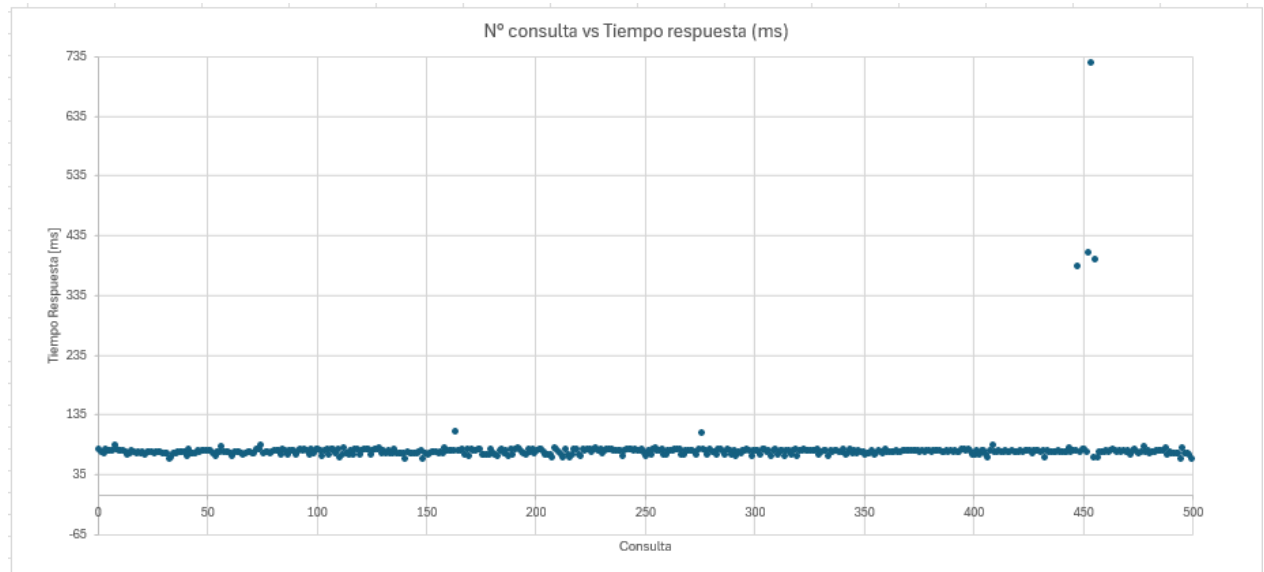
Configurar

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt...
Consulta de ...	500	75	60	722	38,35	60,00%	16,7/sec	6,16	2,53	377,2
Total	500	75	60	722	38,35	60,00%	16,7/sec	6,16	2,53	377,2

Gráfica Generada por la app:



Una vez concluida la prueba con los datos obtenidos podemos generar un gráfico que muestra el tiempo de respuesta de cada consulta:



Considerando como máximo un 1[s] el tiempo de espera promedio podemos decir que a pesar de tener un alto flujo de consultas la aplicación logró completar las 500 consultas en 30[s] con una media de 75[ms].

Creemos que se consigue tener estos tiempos debido a que la consulta es sencilla, es decir, solo es encontrar el documento en la base de datos que contenga el valor de id que se está buscando, además no tenemos un gran volumen de horas registradas en la base de datos por lo que eso ayuda a que los tiempos sean bajos, en un contexto real con un mayor volumen de datos, los tiempos deberían tender a aumentar.