



**UNIVERSIDAD DE CASTILLA-LA MANCHA**  
**ESCUELA SUPERIOR DE INFORMÁTICA**

**Memoria JMeter**

Alberto López Hurtado  
Rubén Montero Martín  
Jorge Muñoz Martínez  
Miguel Rodríguez Lorente

**Titulación:** Grado en Ingeniería Informática

**Curso:** 4º TI

**Fecha:** 05/02/2019

## Introducción

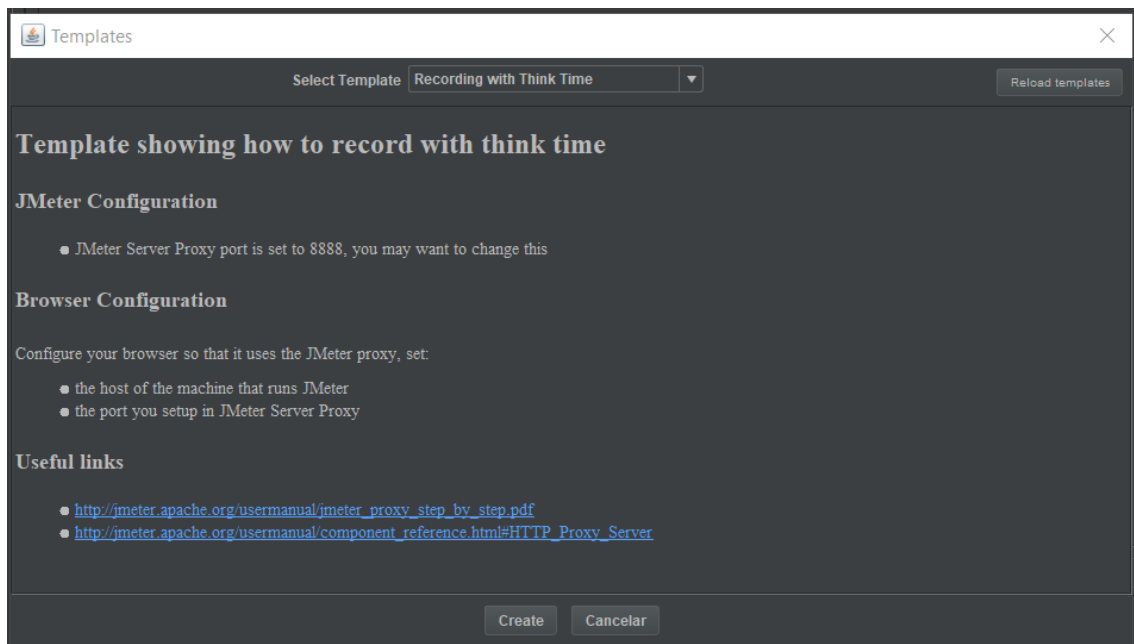
En esta prueba de jMeter se realiza sobre la función **SingUp** y **Login**, usando un archivo CSV con 600 usuarios compuestos por:

- Nombre de usuario.
- Email.
- Contraseña 1.
- Contraseña 2.
- Tipo.

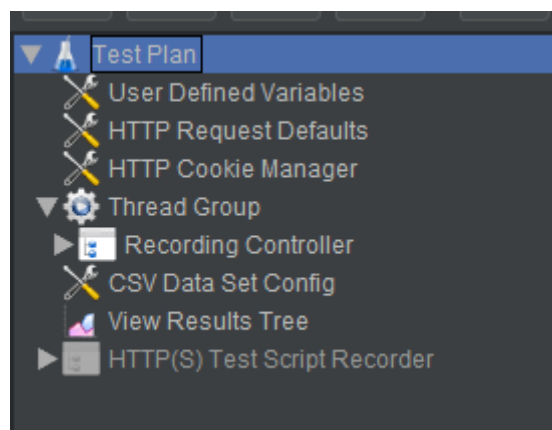
En esta prueba primero se realiza la función de **SingUp** y con los usuarios creados se realiza del **Login** de éstos, probando así el rendimiento de la aplicación cuando se enfrenta a un gran número de peticiones.

## Manual

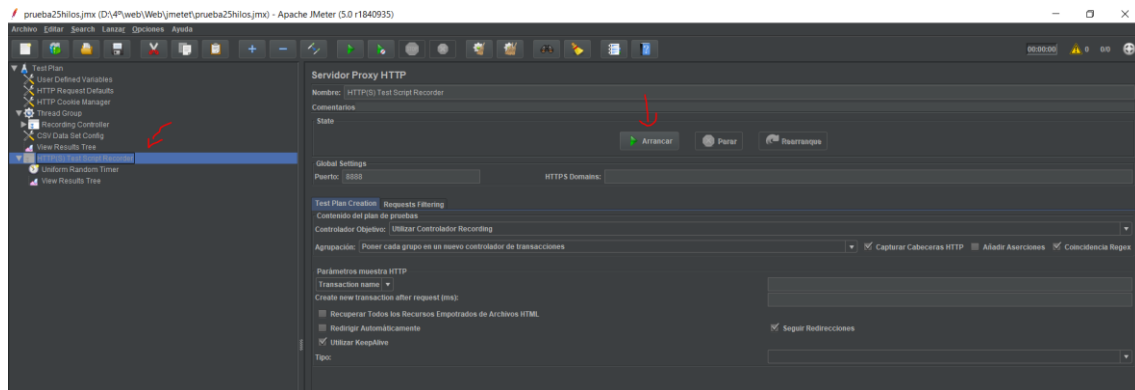
La prueba se ha creado mediante el uso de una plantilla.



Lo que nos genera es un test plan como este:

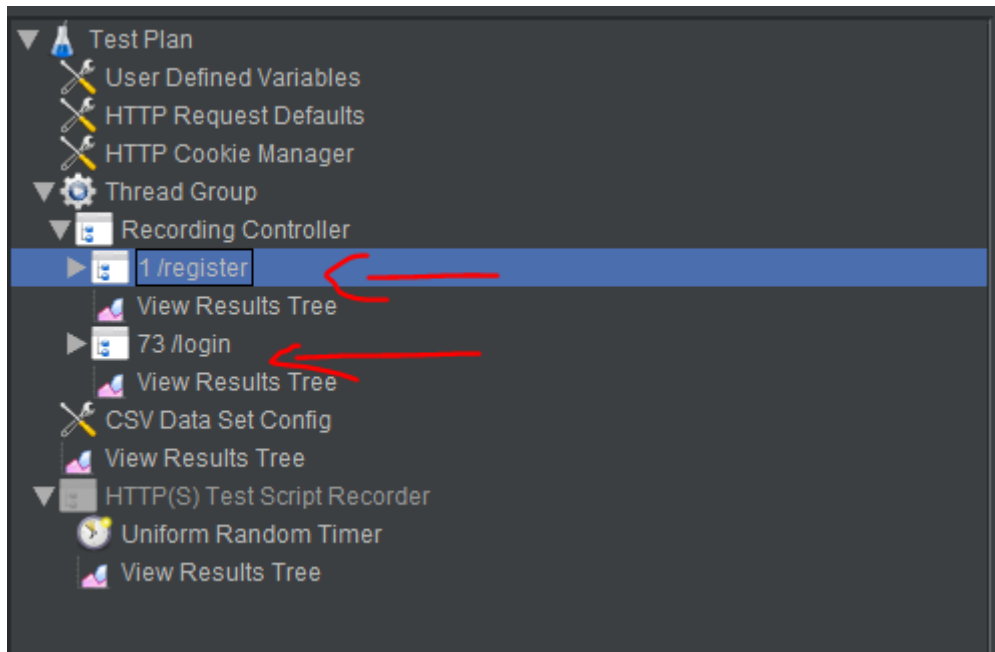


Para grabar el caso base debemos:

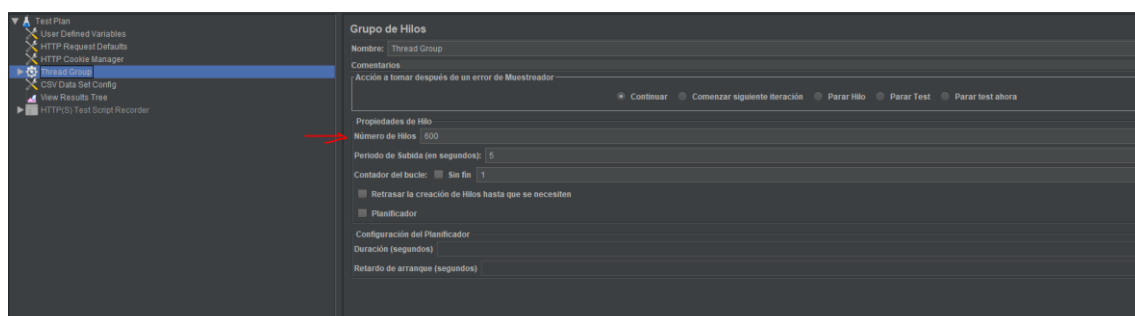


Y ejecutar el programa normalmente.

Una vez esté la grabación realizada, se nos generará en el “**Recording controller**” unos controladores de transición como éstos.



Ahora debemos parametrizar las pruebas para un gran número de usuarios, para ello en “**Thread Group**” podremos en número de usuario (hilos) que queremos ejecutar.

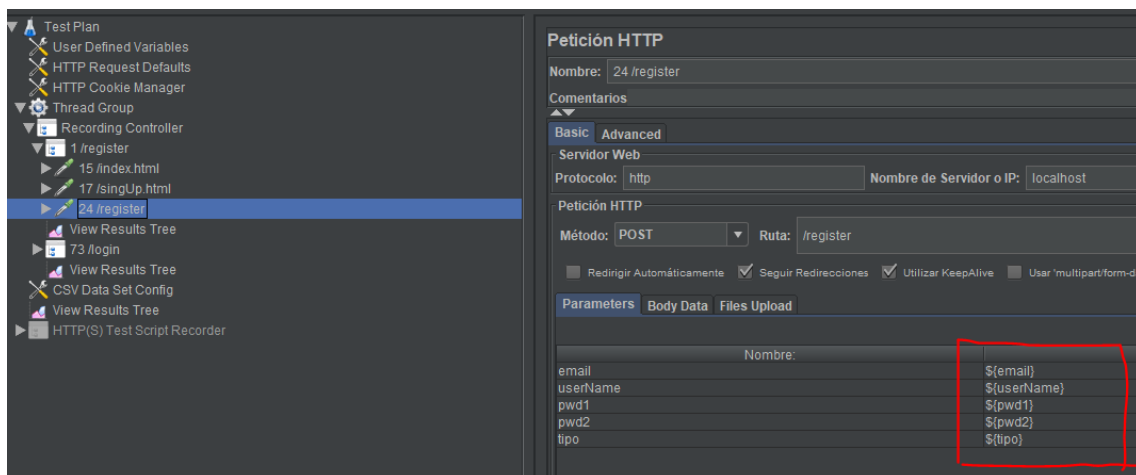


Lo siguiente que debemos hacer es parametrizar las variables que tendrán los objetos, para ellos podemos o crear un CSV e importarlo o meterlos uno a uno, la opción óptima es la creación de un CSV.

Con el CSV creado debemos meter un “**CSV Data Set Config**”, con este elemento nos permite leer variables de un CSV dado una ruta.

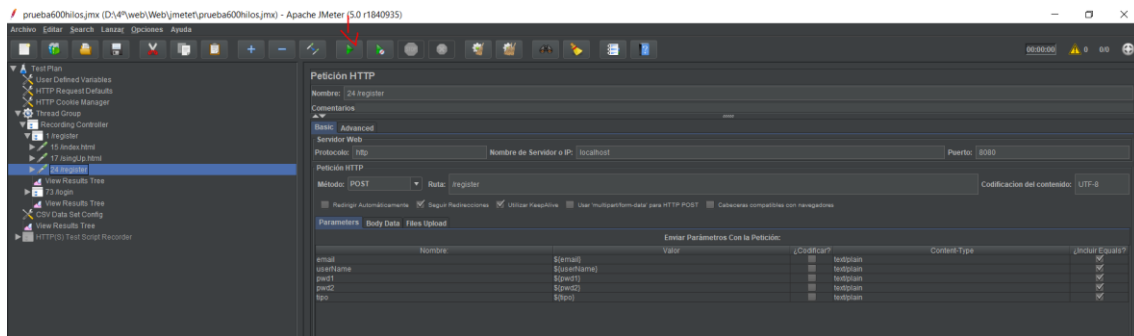


Necesitaremos que los usuarios creados obtengan las variables del CSV, para ello debemos modificar en la petición HTTP que lleva las variables del caso base para que la coja del CSV, para ello debemos realizar.

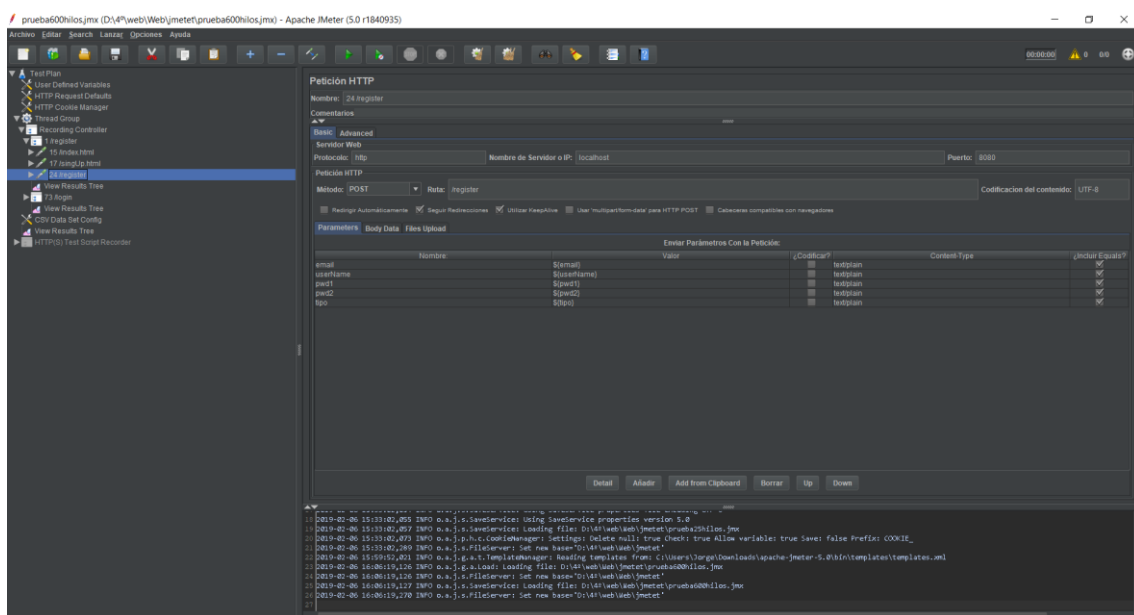


Siendo **\${Nombre de la variable en el CSV}**.

Con todo lo anteriormente dicho ya tendríamos parametrizadas las pruebas de la aplicación, ahora sólo quedaría ejecutarlas.



Seleccionado la función de **“Log Viewer”** podemos ver que está ocurriendo en tiempo real y si sucede algún fallo en la aplicación.



Si queremos tener una visión mas específica de la ejecución del programa podemos insertar un **“View Result Tree”** que nos ofrece una vista de resultado de la ejecución de las operaciones.

