

Providing data to the forms using CSV files in jMeter

Antes de empezar, es importante remarcar que para una explicación más clara se tomará como ejemplo un requisito funcional en concreto: *An actor who is not authenticated must be able to register to the system as a user*. Además debemos tener creado y listo para ejecutar el caso de prueba que queramos, haciéndolo de la forma habitual.

Una vez tengamos creado el caso de prueba, lo abrimos e identificamos la acción que rellena los datos del formulario a testear:

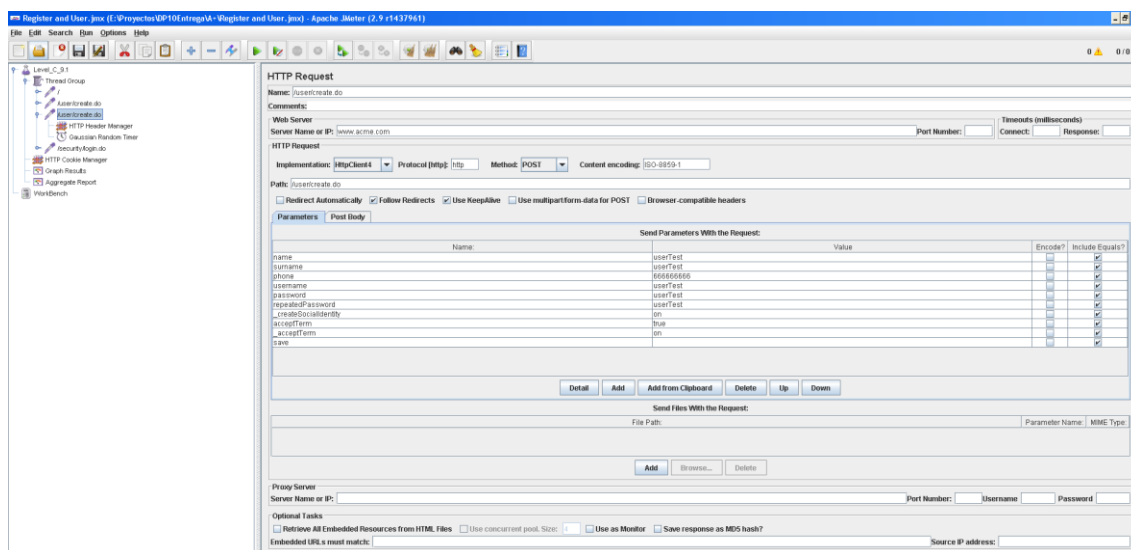


Figura 1

Como vemos, en el despliegue de la ventana “HTTP Request”, en la pestaña “Parameters” se muestran los datos que introdujimos al realizar las acciones manualmente capturadas por el proxy. El siguiente paso es crear nuestro archivo de configuración para datos CSV. Hacemos clic derecho en la acción que identificamos previamente y navegamos por “Add” > “Config Element” > “CSV Data Set Config”:

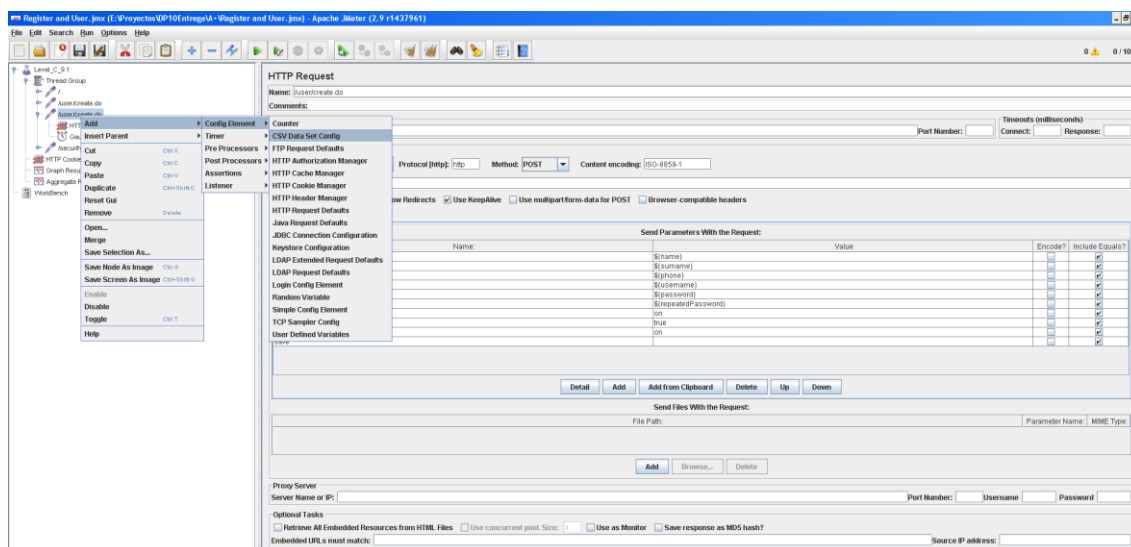


Figura 2

Una vez creado podemos modificar el campo “Name:” para darle un nombre acorde con lo que representarán dichos datos en la aplicación:

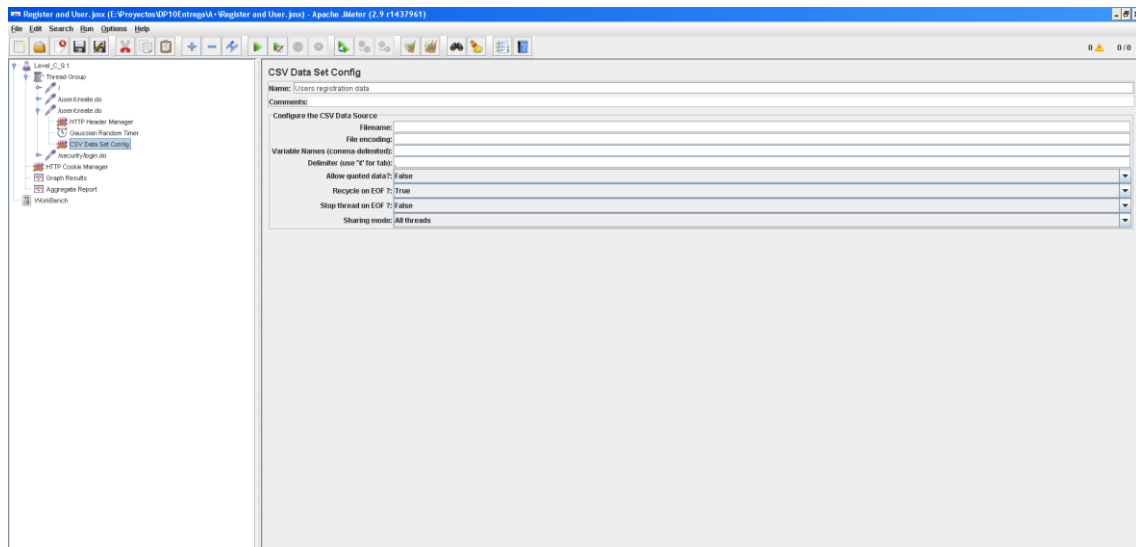


Figura 3

A continuación hacemos lo mismo con el campo “Filename:”, cuyo nombre debe acabar en “.csv” (este será el archivo donde cargaremos los datos):

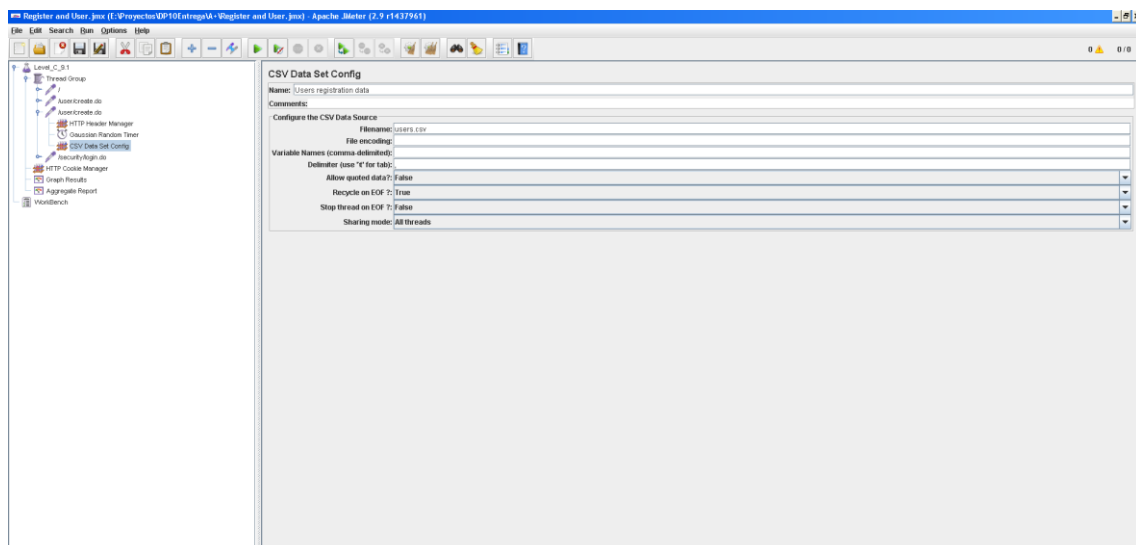


Figura 4

Ahora procedemos a crear dicho archivo. Para ello, en la misma carpeta donde tengamos guardado el caso de prueba que tenemos abierto, creamos un archivo de texto plano (En Windows: Clic derecho > “Nuevo” > “Documento de texto”) y lo renombramos con el nombre exacto que pusimos en el campo “Filename:”, con la extensión “.csv” incluida.

Ahora debemos rellenar el archivo con todos los datos distintos que queremos que se introduzcan en cada iteración del caso de prueba. Para automatizar este proceso, ya que habitualmente requeriremos de miles de datos distintos, haremos uso de la herramienta Excel de Microsoft. Abrimos el archivo con dicha herramienta y rellenamos 2 filas en las que cada columna represente un parámetro de nuestro formulario (por ejemplo, el nombre). Debemos rellenarlo de tal manera que cada fila con la anterior tenga cierta relación, ya que luego

haremos que Excel nos “automatice” un proceso para que se auto-rellene el número de filas que necesitemos y que reconocerá esta relación entre filas para realizar este autocompletado:

F2

:

user0101

| | A | B | C | D | E | F | G | H |
|---|----------|-------------|------------|----------|----------|----------|---|---|
| 1 | name0100 | surname0100 | 1234560100 | user0100 | user0100 | user0100 | | |
| 2 | name0101 | surname0101 | 1234560101 | user0101 | user0101 | user0101 | | |
| 3 | | | | | | | | |
| 4 | | | | | | | | |
| 5 | | | | | | | | |
| 6 | | | | | | | | |
| 7 | | | | | | | | |
| 8 | | | | | | | | |
| 9 | | | | | | | | |

Figura 5

NOTA: Sólo tenemos que tener en cuenta los datos que queremos que varíen o deben variar entre cada iteración, si hay parámetros constantes no tenemos que incluirlos, pues se mantendrán tal y como se hacen de manera habitual.

El siguiente paso es seleccionar los datos que hemos introducido, hacer clic sobre la esquina inferior derecha y, manteniendo pulsado el clic, arrastrar hasta un número de filas igual al número de iteraciones que necesitamos en nuestro caso de prueba. En nuestro caso, como introducimos 10 usuarios, a 100 iteraciones por usuario, necesitaremos 1000 filas:

| | A | B | C | D | E | F | G |
|------|----------|-------------|------------|----------|----------|----------|---|
| 974 | name1073 | surname1073 | 1234561073 | user1073 | user1073 | user1073 | |
| 975 | name1074 | surname1074 | 1234561074 | user1074 | user1074 | user1074 | |
| 976 | name1075 | surname1075 | 1234561075 | user1075 | user1075 | user1075 | |
| 977 | name1076 | surname1076 | 1234561076 | user1076 | user1076 | user1076 | |
| 978 | name1077 | surname1077 | 1234561077 | user1077 | user1077 | user1077 | |
| 979 | name1078 | surname1078 | 1234561078 | user1078 | user1078 | user1078 | |
| 980 | name1079 | surname1079 | 1234561079 | user1079 | user1079 | user1079 | |
| 981 | name1080 | surname1080 | 1234561080 | user1080 | user1080 | user1080 | |
| 982 | name1081 | surname1081 | 1234561081 | user1081 | user1081 | user1081 | |
| 983 | name1082 | surname1082 | 1234561082 | user1082 | user1082 | user1082 | |
| 984 | name1083 | surname1083 | 1234561083 | user1083 | user1083 | user1083 | |
| 985 | name1084 | surname1084 | 1234561084 | user1084 | user1084 | user1084 | |
| 986 | name1085 | surname1085 | 1234561085 | user1085 | user1085 | user1085 | |
| 987 | name1086 | surname1086 | 1234561086 | user1086 | user1086 | user1086 | |
| 988 | name1087 | surname1087 | 1234561087 | user1087 | user1087 | user1087 | |
| 989 | name1088 | surname1088 | 1234561088 | user1088 | user1088 | user1088 | |
| 990 | name1089 | surname1089 | 1234561089 | user1089 | user1089 | user1089 | |
| 991 | name1090 | surname1090 | 1234561090 | user1090 | user1090 | user1090 | |
| 992 | name1091 | surname1091 | 1234561091 | user1091 | user1091 | user1091 | |
| 993 | name1092 | surname1092 | 1234561092 | user1092 | user1092 | user1092 | |
| 994 | name1093 | surname1093 | 1234561093 | user1093 | user1093 | user1093 | |
| 995 | name1094 | surname1094 | 1234561094 | user1094 | user1094 | user1094 | |
| 996 | name1095 | surname1095 | 1234561095 | user1095 | user1095 | user1095 | |
| 997 | name1096 | surname1096 | 1234561096 | user1096 | user1096 | user1096 | |
| 998 | name1097 | surname1097 | 1234561097 | user1097 | user1097 | user1097 | |
| 999 | name1098 | surname1098 | 1234561098 | user1098 | user1098 | user1098 | |
| 1000 | name1099 | surname1099 | 1234561099 | user1099 | user1099 | user1099 | |
| 1001 | | | | | | | |
| 1002 | | | | | | | |
| 1003 | | | | | | | |

Figura 6

Una vez introducidos los datos pasamos a guardarlo. Para ello vamos a “Archivo” > “Guardar como”:



Figura 7

Dejamos el mismo nombre y extensión previamente configurados, pero nos preocupamos especialmente por escoger en el campo “Tipo:” la opción “CSV (delimitado por comas) (*.csv)”:

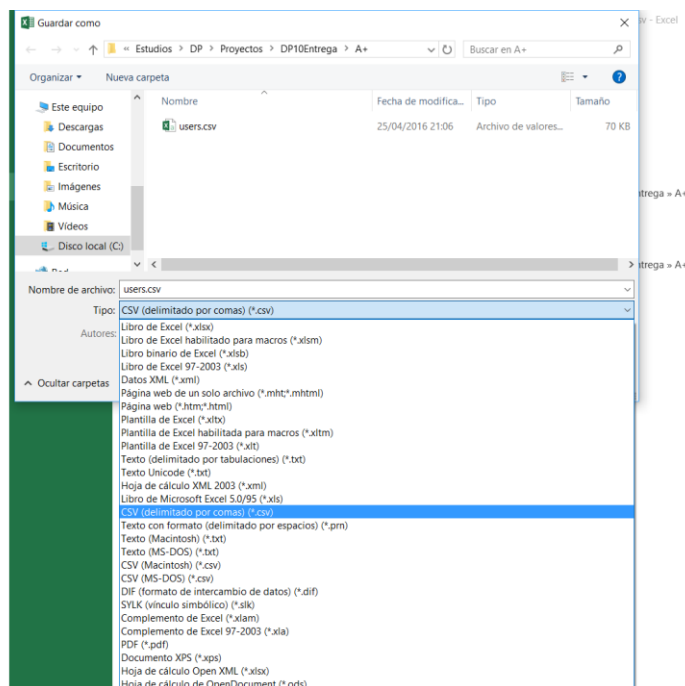


Figura 8

Volvemos a nuestro archivo de configuración CSV en jMeter y en la ventana “CSV Data Set Config” que aparece a la derecha, rellenamos el campo “Variable Names (comma-delimited):” con el nombre de los datos que necesitamos automatizar en el formulario, separado por comas:

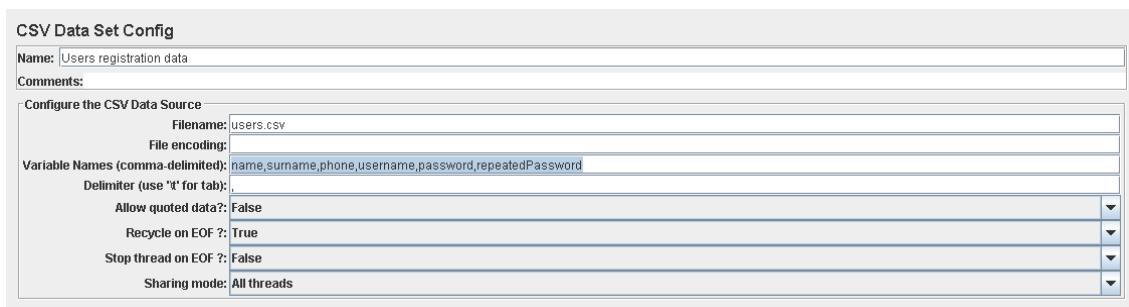


Figura 9

NOTA: No hay que separarlos por “coma + espacio” como si introdujésemos una lista de forma natural, los parámetros deben estar separados sólo por comas.

Ahora modificamos el siguiente campo “Delimiter (use ‘\t’ for tab):”, cambiando la “,” por “;”. Esto se debe a que, aunque especificamos a Excel que el formato estuviese delimitado por comas, el archivo generado resultante se genera con los parámetros separados con punto y coma. Podemos comprobarlo si abrimos el archivo con cualquier editor de texto plano como el “Bloc de notas” de Windows:

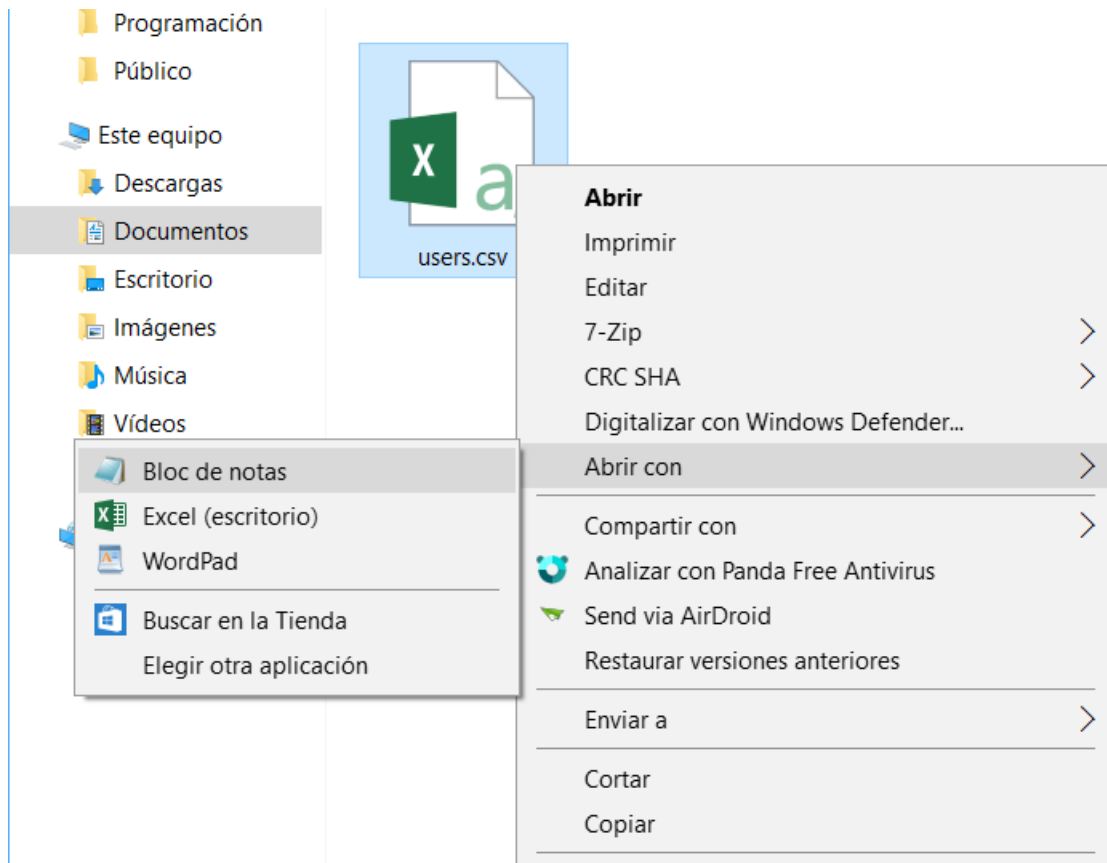


Figura 10

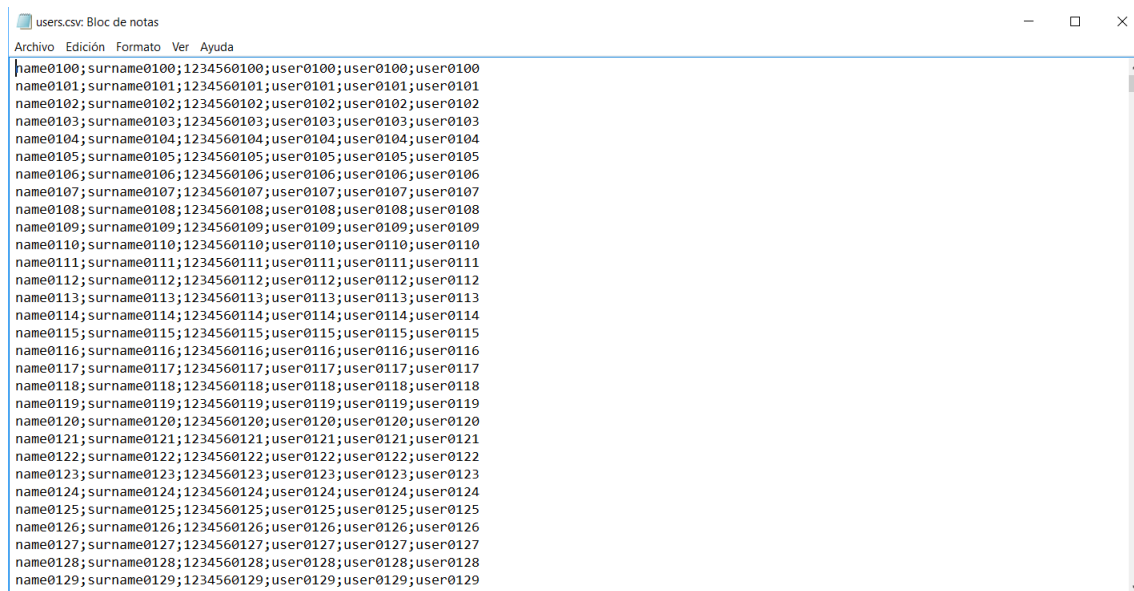


Figura 11

| CSV Data Set Config | |
|-----------------------------------|---|
| Name: | Users registration data |
| Comments: | |
| Configure the CSV Data Source | |
| Filename: | users.csv |
| File encoding: | |
| Variable Names (comma-delimited): | name,surname,phone,username,password,repeatedPassword |
| Delimiter (use '\t' for tab): | |
| Allow quoted data?: | False |
| Recycle on EOF?: | True |
| Stop thread on EOF?: | False |
| Sharing mode: | All threads |

Figura 12

El siguiente paso es posicionar nuestro archivo de configuración en el sitio adecuado para que se ejecute antes de cada iteración. Simplemente arrastramos el archivo hasta la primera acción del caso de prueba y se nos desplegará un menú en el que seleccionaremos “Insert Before”:

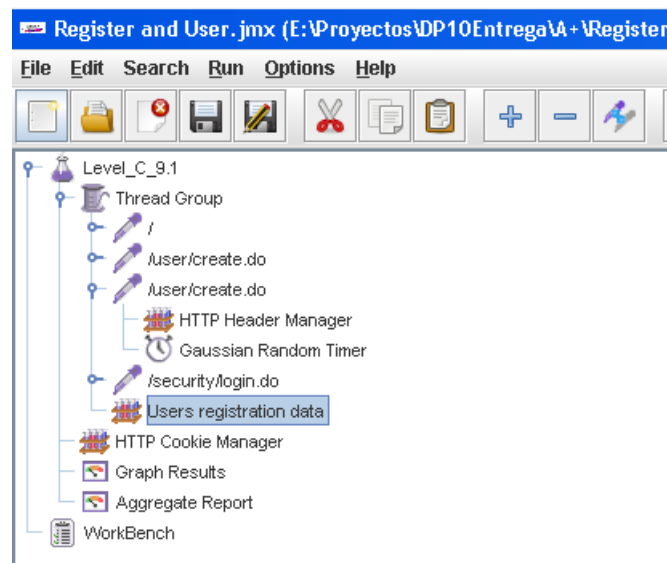


Figura 13

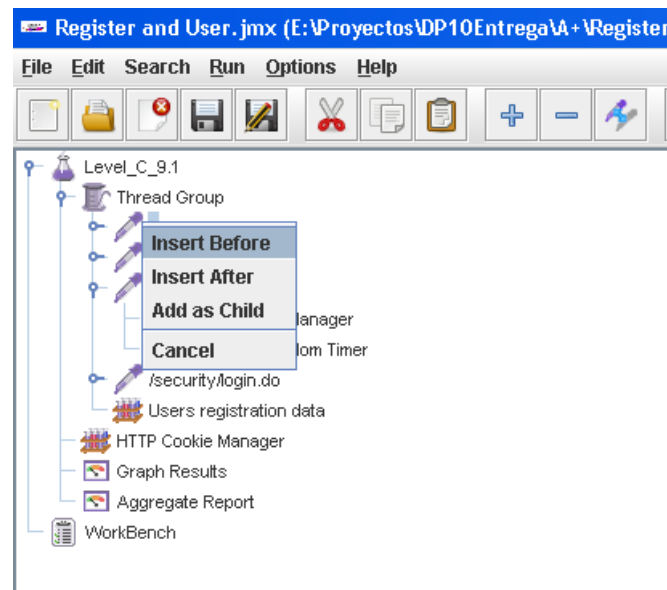


Figura 14

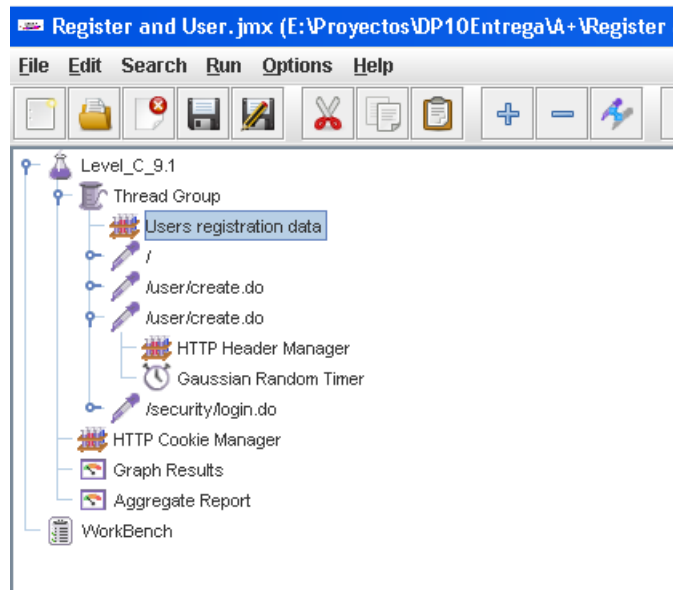


Figura 15

Volvemos a la acción protagonista y modificamos la columna “Value” de los parámetros que identificamos previamente con el siguiente formato: $\${nombreDelParámetro}$

| Name | Value | Encode? | Include Equals? |
|----------------------|------------|-------------------------------------|-------------------------------------|
| name | username | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| username | username | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| phone | 0666666666 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| password | password | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| repeatPassword | password | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| createSocialIdentity | on | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| acceptTerms | True | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| save | on | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

Figura 16

NOTA: Ese *nombreDelParámetro* debe coincidir exactamente con los introducidos en la [figura 9](#).



Una vez completados todos los pasos, ya podemos ejecutar nuestro caso de prueba de forma habitual.