

Un procedimiento que puede ayudarnos a encontrar las rutas que debemos tratar en un controlador es fijarnos en todas nuestras peticiones posibles y hacernos un mapa

¿cómo dilucidamos todas las peticiones posibles... Bueno, sabemos que siempre existe una petición de tipo GET para acceder a la raíz de la aplicación. Todas las demás vienen dadas por las vistas de nuestra aplicación. Así que habrá que hacer un “esbozo” de lo que van a ser nuestras vistas

Vista para ingresar el nick:

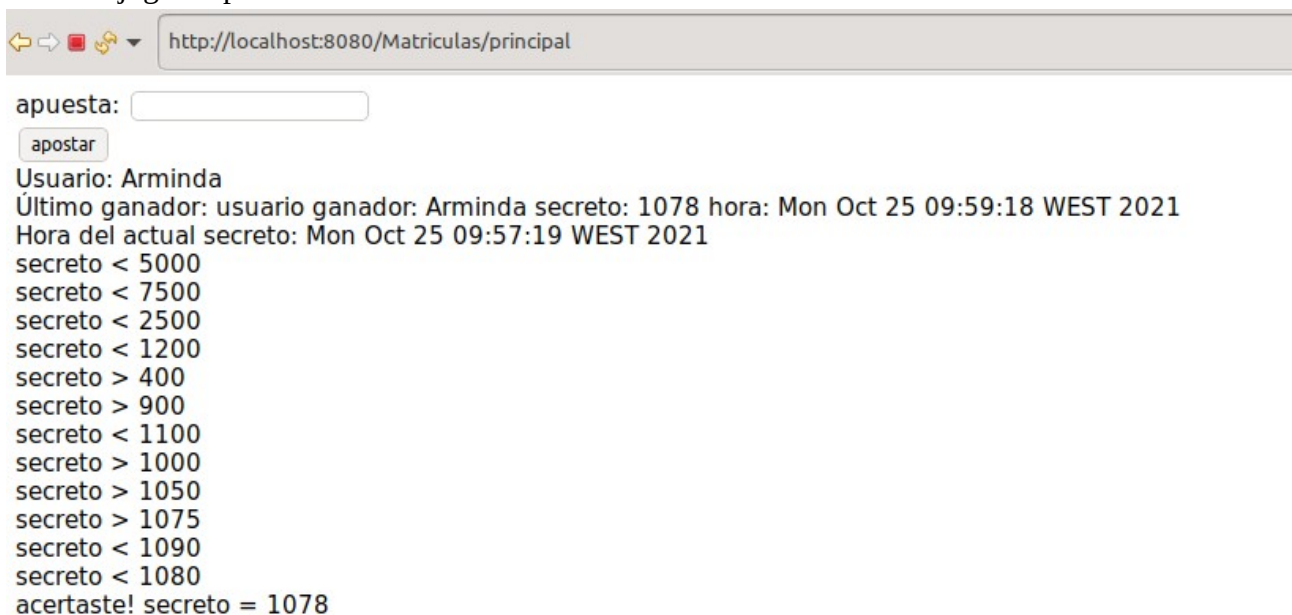


Se debe crear un nick para jugar:

Nick usuario:

Crear

Vista de jugar la partida



apuesta:

apostar

Usuario: Arminda
Último ganador: usuario ganador: Arminda secreto: 1078 hora: Mon Oct 25 09:59:18 WEST 2021
Hora del actual secreto: Mon Oct 25 09:57:19 WEST 2021
secreto < 5000
secreto < 7500
secreto < 2500
secreto < 1200
secreto > 400
secreto > 900
secreto < 1100
secreto > 1000
secreto > 1050
secreto > 1075
secreto < 1090
secreto < 1080
acertaste! secreto = 1078

Así pues tenemos:

- Petición GET a la raíz de la aplicación (siempre se da al inicio de la aplicación)

- Peticiones desde la vista que crea el nick:

* Petición GET desde esta vista: descartable. Vemos que únicamente tiene un formulario post para ingresar el nick. NO se ve que haya aporte de información del servidor de una posible petición GET desde esta página (recordar que una petición GET tiene sentido si nos pueden traer información desde el servidor. Pero en esta vista no tenemos nada en ese sentido) → **LA DESCARTAMOS**

* Petición POST desde esta vista: ¿ qué parámetros enviará el usuario ? Únicamente hay un input para el nick. Parece lógico pensar que nos están enviando un parámetro llamado: **nick**

- Peticiones desde la vista que juega la partida:

* Petición GET desde esta vista: nos debe devolver de nuevo todas las jugadas que ya estamos viendo (porque las hemos hecho nosotros) y lo único que puede aportarnos que sea “nuevo“ es la información si otro usuario ha ganado la actual partida: No la vamos a descartar entonces. ¿ Qué información de parámetros nos puede enviar ? → en este caso no es aplicable. No hay parámetros get que nos envíen desde esta vista, simplemente devolverá la misma vista actualizada

* Petición POST desde esta vista: ¿ qué parámetros envía el usuario? : **apuesta**

Bueno.. Ahora podemos construir una tabla con los parámetros para petición get y post en el Servlet, y los casos de uso:

GET :

Petición GET a la raíz. NO es relevante parámetros	----- >	Si usuario no está “autenticado”, osea no tiene nick en la sesión => enviar a vista crear nick
		Si usuario tiene ya nick guardado en la sesión => enviar a la vista jugar partida

POST

Parámetros enviado al servlet		Acción en el Servlet
NICK	APUESTA	
No enviado	No enviado	NO tiene sentido. Un envío post debe lleva datos => llevar a una vista sin hacer nada ¿ pero a qué vista ? Eso sabemos que depende de si el usuario se ha “autenticado” si no lo ha hecho => reenviar a vista crear nick si se ha autenticado => reenviar a jugar partida
No enviado	enviado	Se ha recibido parámetro apuesta y no el parámetro nick. Otra vez hay que ver los dos casos: usuario ya autenticado (tiene nick en sesión): se ha recibido una apuesta así que habrá que contrastar la apuesta contra el número secreto y reenviar a la vista de jugar partida usuario no autenticado (no tiene nick en sesión): si se recibe una apuesta, NO se ha recibido un parámetro nick y el usuario no está autenticado (no tenemos un nick guardado en la sesión) no podemos jugar la apuesta ya que no tenemos nick => reenviar a crear nick
enviado	No enviado	Hemos recibido el parámetro nick y NO el parámetro apuesta. Otra vez debemos decidir según si el usuario ya tiene un nick guardado en la sesión: usuario ya tiene nick en sesión (autenticado). En este caso ¿ de qué nos vale el parámetro nick recibido ? Ya lo tenemos almacenado en la sesión y NO permitimos que se cambie de nombre. Como no ha enviado una apuesta tampoco podemos validarla contra el número secreto. => lo enviamos a la vista de jugar partida para que nos envíe una apuesta Usuario NO tiene nick en sesión (no autenticado) En este caso sí hemos recibido un parámetro nick así que => establecemos el parámetro nick recibido como el nick de nuestra sesión y reenviamos a la pantalla de jugar partida
enviado	enviado	Parece absurdo que podamos recibir los dos parámetros post a la vez ya que ninguna de las dos vistas que tenemos lo permite. Podemos descartar esta opción si consideramos que el usuario no es malintencionado ya que no puede hacerse desde nuestras vistas creadas ¿ y si es malintencionado ha generado una petición post fuera de nuestras vistas y la está enviando ? En ese caso podría estar

		<p>tratando de jugar una apuesta en nombre de otro usuario. Una alternativa podría ser: “castigarlo” quitándole el nick que ya tenga asignado (si lo tiene) borrarle todas sus jugadas y enviarlo a la página de crear nick</p> <p>Otra alternativa es no hacerle caso, ya que no nos afecta. En ese caso simplemente se le lleva a la vista que le corresponde => vista de crear nick si no lo tiene o vista de jugar partida si tiene creado un nick en la sesión</p>
--	--	--