

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 741 895**

51 Int. Cl.:

G06F 9/44

(2008.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **12.09.2006 PCT/EP2006/008875**

87 Fecha y número de publicación internacional: **12.04.2007 WO07039048**

96 Fecha de presentación y número de la solicitud europea: **12.09.2006 E 06791997 (7)**

97 Fecha y número de publicación de la concesión europea: **15.05.2019 EP 1934717**

54 Título: **Método para controlar una ventana de navegador**

30 Prioridad:

26.09.2005 US 720607 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

12.02.2020

73 Titular/es:

**TELEFÓNICA GERMANY GMBH & CO. OHG
(100.0%)**

**Georg-Brauchle-Ring 23-25
80992 München, DE**

72 Inventor/es:

THIERBACH, HEIKO

74 Agente/Representante:

CARVAJAL Y URQUIJO, Isabel

ES 2 741 895 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Método para controlar una ventana de navegador

Campo de la invención

5 La presente invención se refiere a un método para controlar una ventana de navegador. La invención se refiere además a un dispositivo servidor que se comunica con un dispositivo cliente y está dispuesto para posibilitar el control de una ventana de navegador en un dispositivo cliente. La presente invención se refiere además a un programa de ordenador que posibilita el control de una ventana de navegador.

Técnica anterior

10 Hoy en día, un navegador web ofrece la posibilidad de ejecutar aplicaciones o presentar datos dentro de más de una ventana de navegador. Por ejemplo, en la primera ventana de navegador se presenta la página de inicio de una aplicación de portal, mientras que una segunda ventana de navegador se abre por otra aplicación, por una actividad de usuario o automáticamente. De la segunda ventana puede abrirse y utilizarse una tercera ventana, y así sucesivamente. Esto lleva a la situación de que pueden abrirse un montón de ventanas de navegador y usarse simultáneamente en el ordenador cliente.

15 Las aplicaciones de portal pueden ofrecer la posibilidad de identificar a un usuario. Mediante un mecanismo de inicio de sesión, por ejemplo, se identifica al usuario y comienza una sesión personalizada. Una sesión personalizada es el periodo de tiempo entre el inicio de sesión de un usuario y el cierre de sesión de un usuario, o más general, el periodo de tiempo mientras un usuario es conocido, en el cual un usuario puede realizar acciones privadas en el ordenador cliente. La identidad del usuario obtenida puede transferirse a otras aplicaciones, por ejemplo, enviando los datos de
20 identidad del usuario a la otra aplicación mediante TCP/IP, de modo que el usuario no necesita autenticarse por sí mismo de nuevo en otra aplicación relacionada con la misma sesión personalizada. En este caso, el usuario ya es conocido y puede acceder a datos personales dentro de diferentes aplicaciones, por ejemplo, emails, facturas, control de gastos. De manera similar, en dicha sesión personalizada, el usuario está habilitado para realizar pagos, compras, ventas u otras acciones privadas sin autenticarse a sí mismo en cada una de dichas aplicaciones tras comenzar una
25 sesión personalizada.

Al final de una sesión personalizada, se recomienda que el usuario abandone el estado "en sesión" para pasar a "fuera de sesión" de todas las aplicaciones de portal que haya usado antes. En práctica, se ofrece una función de cierre de sesión dentro de cada aplicación, lo que informa a las aplicaciones y a un caso de autenticación centralizada, de que la sesión personalizada se ha vuelto no válida a petición de un usuario. Por ejemplo, tras iniciar sesión en un sitio de
30 subastas como EBAY, un usuario puede abrir múltiples ventanas y realizar acciones sin autenticación. Sin embargo, si el usuario cierra sesión en una de las ventanas, las otras ventanas permanecen abiertas y asumen que la sesión personalizada aún está activa. Para realizar una acción privada en una de las ventanas tal como hacer una puja para un objeto, primero tiene que iniciar sesión otra vez.

35 Hoy en día existen diferentes implementaciones de portales. Implementaciones de portales conocidas ofrecen la posibilidad de ejecutar todas las aplicaciones dentro de solo una ventana de navegador (por ejemplo, utilizando la tecnología marco) o abriendo múltiples ventanas que no se cierran simultáneamente después de que un usuario solicite cerrar sesión.

40 El artículo de Internet "How do I cookie a popup?" (ANÓNIMO, 16 de enero 2003, págs. 1-8, XP002409242, recuperado de Internet por EPO el 27-11-2006 de la URL: <http://lamasery.com/forums/archive/index.php/t-1221.html>) describe un método basado en cookies para controlar el número de veces que se muestra una ventana pop-up a un usuario de un navegador y cierra la ventana pop-up cuando el usuario ha hecho click en el pop-up. El documento WO 03/069465 se refiere a un método para efectuar el cierre de una ventana de aplicación en respuesta a un evento asociado con su ventana padre. El método comprende registrar un indicador asociado con la ventana padre de aplicación y en la
45 ausencia de dicho indicador acceder a una función proporcionada por un sistema provocando el cierre de la ventana de aplicación.

Sumario de la invención

La presente invención busca proporcionar un método mejorado para controlar una ventana de navegador según las reivindicaciones 1 y 5. Se definen características opcionales en las reivindicaciones dependientes 2-4.

50 Según un aspecto de la presente invención, el método para controlar una ventana de navegador comprende las etapas de asociar la ventana de navegador a un testigo indicativo para una sesión personalizada; verificar dicho testigo iterativamente; y cerrar la ventana de navegador con dependencia de dicho testigo. La utilización de un testigo indicativo para una sesión personalizada permite indicar para múltiples aplicaciones en múltiples ventanas si el usuario aún está conectado en dicha sesión personalizada. Cada una de las ventanas de navegador que pertenecen a dicha sesión personalizada pueden monitorizar al testigo. Un cambio en el estado de la sesión personalizada realizado en cualquiera
55 de las aplicaciones puede aplicarse al testigo. Tan pronto como el testigo indica que el usuario ha cerrado sesión, una

ventana de navegador que pertenece a dicha sesión personalizada puede detectar esto y puede realizar la acción correspondiente como cerrar la ventana de navegador.

5 En una realización adicional de la invención el método comprende además la etapa de eliminar a dicho testigo iniciado mediante cierre de sesión de dicha sesión personalizada. Eliminar el testigo puede ser un mecanismo simple y seguro para indicar a todas las ventanas asociadas con la sesión personalizada que el usuario ha cerrado sesión y tiene que cerrarse la ventana. Incluso cuando el testigo se eliminó accidentalmente por cualquier aplicación no asociada con la sesión personalizada, pueden cerrarse todas las ventanas para proteger al usuario ante el uso no autorizado de cualquiera de las ventanas de navegador asociadas con la sesión personalizada.

10 En una realización adicional de la invención el método comprende además la etapa de generar un marco para realizar la verificación de dicho testigo iterativamente; y cerrar la ventana de navegador con dependencia de dicho testigo. La utilización de un marco específico en la ventana de navegador para realizar dichas acciones permite la programación fácil de los otros marcos. Un ingeniero de software no tiene en cuenta que este software en un marco de dichas ventanas realiza la acción correspondiente. En una realización de la invención dicho marco para realizar la verificación de dicho testigo iterativamente; y cerrar la ventana de navegador con dependencia de dicho testigo, es un marco invisible. La utilización de un marco invisible (u oculto) permite controlar la ventana de navegador sin molestar al usuario con la existencia de dicho marco.

15 En una implementación a modo de ejemplo adicional de la invención, el método comprende además generar dicho testigo en inicio de dicha sesión personalizada. Mediante la generación del testigo al inicio de una sesión personalizada y la asociación de la ventana de navegador a dicho testigo, se ha descubierto una manera uniforme y no ambigua de controlar las ventanas de navegador asociadas con la sesión personalizada.

20 En una implementación a modo de ejemplo adicional de la invención, se genera el testigo en una ventana de navegador adicional, y en generación de la ventana de navegador a partir de la ventana de navegador adicional, la ventana de navegador se asocia con el testigo generado en la ventana de navegador adicional. La utilización de esta implementación a modo de ejemplo garantiza que después de que se haya generado un testigo en una ventana de navegador, todas las ventanas de navegador subsecuentes que se abrieran por una acción de usuario dicha ventana de navegador pueden asociarse con dicho testigo. De esta forma una nueva ventana de navegador iniciada de una ventana de navegador asociada con una sesión personalizada puede asociarse automáticamente con dicha sesión personalizada y por consiguiente, cerrarse tan pronto como en cualquiera de las ventanas de navegador el usuario que ha cerrado sesión de dicha sesión personalizada.

30 **Breve descripción de los dibujos**

A continuación, la presente invención se explicará más detalladamente, utilizando un número de realizaciones a modo de ejemplo, con referencia a los dibujos adjuntos, en los cuales

la figura 1 muestra esquemáticamente una pantalla de ordenador con un número de ventanas de navegador.

La figura 2 es un diagrama de flujo para un primer método para controlar una ventana de navegador.

35 La figura 3 es un diagrama de flujo para un segundo método para controlar una ventana de navegador.

La figura 4 muestra esquemáticamente un sistema de ordenador dispuesto para controlar una ventana de navegador en un dispositivo cliente.

Descripción detallada de realizaciones a modo de ejemplo

40 Para el propósito de enseñanza de la invención, se describen a continuación realizaciones preferidas del método y sistema de la invención. Para el experto en la técnica resultará evidente que pueden concebirse y reducirse a la práctica otras realizaciones de la invención alternativas y equivalentes sin apartarse del verdadero espíritu de la invención, estando limitado solamente el alcance de la invención por las reivindicaciones finalmente concedidas.

45 La figura 1 muestra esquemáticamente una pantalla de ordenador con un número de ventanas de navegador. La pantalla de ordenador mostrada es un ejemplo de una pantalla de ordenador que podría ser visible hoy en día. En el ejemplo una ventana 2 de navegador en el fondo de la pantalla es la ventana de navegador para iniciar sesión en una página web portal de internet con el fin de iniciar una sesión personalizada. Se prevé que un portal de sitio web de internet sea la primera página web que encontrará o utilizará un usuario cuando busque información o un servicio de una organización específica. Proporciona asesoramiento a los usuarios para encontrar información, utilizando servicios electrónicos, y enviando preguntas y comentarios. Los portales proporcionan un amplio grupo de recursos y servicios, incluyendo un directorio de otros sitios web, una facilidad para buscar otros sitios, noticias, e-mail, información telefónica y mapas, y a veces un foro de comunidad. Una sesión personalizada es todo el intercambio de datos entre el ordenador cliente y un proveedor del servicio, comenzando cuando se establece la conexión iniciando sesión mediante autenticación de un usuario y finalizando cuando termina la conexión mediante cierre de sesión.

Tras iniciar sesión en el portal del sitio web el usuario puede iniciar otra aplicación, o bien del portal del sitio web o bien de cualquier otra fuente de aplicación. Cada aplicación nueva se inicia en una nueva ventana de navegador. Como el usuario inicia sesión en el portal del sitio web, el usuario no tiene que autenticarse a sí mismo para iniciar cualquier aplicación nueva para acceder a datos personales. En la ventana 4 de navegador, por ejemplo, se inicia una aplicación de cuentas bancarias, en la ventana 6 de navegador una aplicación de correo electrónico y en la ventana 8 una aplicación "i-mode". "I-mode" es un servicio de información basada en paquete del propietario que envía información, tales como banca móvil y dibujos animados a teléfonos móviles, y posibilita el intercambio de correos electrónicos de teléfonos. Cada una de las ventanas 2, 4, 6, 8 de navegador, que pertenecen todas a la misma sesión personalizada, tienen una posibilidad 10, 12 de cerrar sesión en el portal de la página web. Tan pronto como un usuario decide cerrar sesión, se cerrarán todas las ventanas de navegador que pertenecen a la sesión personalizada. Por medio de los diagramas de flujo en la figura 2 y la figura 3 se explicará más detalladamente una realización del presente método de controlar dichas ventanas de navegador.

La figura 2 es un diagrama de flujo de una primera realización del método para controlar una ventana de navegador. Controlar una ventana de navegador es, por ejemplo, la apertura y/o cierre de la ventana de navegador. Otro ejemplo de controlar una ventana de navegador es situar la ventana de navegador en una ubicación x-y específica. Aún otro ejemplo de controlar una ventana de navegador es ajustando o cambiando el tamaño de la ventana de navegador. Se asume que el usuario ya abrió una ventana de navegador. En el estado 200 el usuario entró al portal de la página web. Esto podría realizarse, por ejemplo, entrando o seleccionando la URL correspondiente en la ventana de navegador. Normalmente, en la ventana de navegador el usuario podría introducir su nombre de usuario y contraseña para autenticar al usuario en el sitio web. La autenticación se realiza en el estado 202 y la sesión personalizada se inicia. Posteriormente, se genera un testigo válido en el estado 204. El testigo podría ser parte de un conjunto de datos común, que está disponible dentro del entorno del navegador. El testigo se crea por una aplicación de portal tras autorización con éxito de un usuario. Podría usarse una cookie como un depósito de testigo en el ordenador cliente. El testigo también podría ser datos almacenados en un servidor del sitio web como una página web. En este caso, el testigo podría ser accesible mediante una URL. El testigo podría ser, por ejemplo, un archivo con un número de archivo específico o datos específicos en un archivo específico. Se considera válido un testigo si el testigo se corresponde con una definición predefinida. El testigo podría ser válido a través de su existencia o similitud con un valor o secuencia de caracteres predefinido.

Después de que se crea el testigo, se crea un primer marco en el estado 206 y se crea un segundo marco en el estado 208. Un marco es un formato para documentos web que dividen la pantalla en segmentos, cada uno con una barra de desplazamiento como si fuera una "ventana" dentro de la ventana de navegador. Preferiblemente, el primer marco es invisible o imperceptible por el usuario ('marco oculto'). En el primer marco, se incluye un trozo de software que espera iterativamente un periodo de tiempo 210 y verifica la validez del testigo 212 hasta que el testigo se vuelve no válido. Cuando se detecta que el testigo no es válido, se cierra la ventana de navegador en el estado 214. En una realización a modo de ejemplo del método el periodo de tiempo entre dos verificaciones subsecuentes es 2 segundos.

El siguiente código fuente resumido muestra una implementación de referencia:

```
<script language="JavaScript">
  function wait() {
    // call the function token after 1 second
    windows.setTimeout("token() ",1000);
  }
  function token() {
    // check, if the token is still present
    if (document.cookie.indexOf("anysting") != -1) {
      // do nothing
    } else {
      // close the window
      self.close();
    }
    // call the function wait after 1 second
    windows.setTimeout("wait() ",1000);
  }
</script>
```

En este ejemplo, el testigo está conectado a una cookie HTTP (Protocolo de Transferencia de Hipertexto). La secuencia de órdenes anterior debe implementarse dentro del documento marco como sigue:

```

<html>
  ...
<body onload=wait()>
  ...
</html>

```

5 En el segundo marco se inicia una aplicación de portal con las correspondientes acciones 216 de usuario. Normalmente, una de las acciones que puede realizar un usuario en una aplicación de portal es cerrar sesión de la aplicación de portal. Esto se corresponde con el fin de la sesión personalizada 218. Si el usuario decide finalizar la sesión personalizada, el testigo se invalidará 220. Cómo debe invalidarse el testigo depende de la implementación. El testigo puede invalidarse por eliminación del registro de página web o cambiando el valor o secuencia de caracteres en un registro o página web.

Debe observarse que no siempre tiene que crearse el segundo marco. La aplicación de portal podría iniciarse en un marco que ya existe en la ventana de navegador.

10 Una de las acciones del usuario que puede realizarse en el estado 216 es iniciar otra aplicación de portal en una nueva ventana. La figura 3 es un diagrama de flujo para una realización adicional del presente método para controlar dicha nueva ventana de navegador. En el estado 300 la ventana de navegador se crea en la pantalla. Como la ventana de navegador se crea por una acción de una sesión personalizada, la ventana de navegador se asocia con el testigo 302. Dicho testigo corresponde al testigo de la ventana de navegador en la que se realiza la acción de abrir la nueva ventana de navegador. Posteriormente en la ventana de navegador se crean un primer marco y un segundo marco en la ventana de navegador, correspondiéndose a los estados 306 y 308 respectivamente. El primer marco podría ser invisible o imperceptible para el usuario.

20 En el primer marco se incluye un trozo de software que espera iterativamente un periodo 310 de tiempo y verifica la validez del testigo 312 hasta que el testigo se vuelve no válido. Cuando se detecta que el testigo no es válido, la ventana de navegador se cierra en el estado 314. En una realización a modo de ejemplo del método el periodo de tiempo entre dos verificaciones subsecuentes es 2 segundos.

25 En el segundo marco la aplicación de portal solicitada se inicia en el estado 304. El usuario puede realizar las acciones 316 de usuario deseadas. Normalmente, una de las acciones que puede realizar un usuario en una aplicación de portal es el cierre de sesión de la aplicación de portal. Esto corresponde con la finalización de la sesión personalizada 318. Si el usuario decide finalizar la sesión personalizada, el testigo se invalidará 320. Cómo debe invalidarse el testigo depende de la implementación. El usuario también podría realizar la acción de usuario para iniciar otra aplicación de portal en una nueva ventana de navegador. Para esta ventana puede usarse el método según el diagrama de flujo en la figura 2.

30 La figura 2 difiere de la figura 3 en que la creación del testigo se muestra en la realización de la figura 2 y la asociación con un testigo ya presente se muestra en la realización de la figura 3. Después de que se determine el testigo que va a verificarse en la ventana de navegador, ambos diagramas de flujo utilizan el mismo software para controlar la ventana de navegador. Dado que todas las ventanas de navegador que pertenecen a la misma sesión personalizada verifican el mismo testigo, todas las ventanas de navegador se cerrarán tan pronto como se invalide el testigo. De este modo se presenta un método muy simple pero efectivo para controlar ventanas de navegador que pertenecen a la misma sesión personalizada.

40 La figura 4 muestra esquemáticamente un sistema dispuesto para controlar una ventana de navegador. El sistema comprende un dispositivo (404) servidor y un dispositivo (402) cliente. El dispositivo (404) servidor es un sistema de ordenador de un proveedor del servicio que posibilita mediante aplicaciones de portal la búsqueda selectiva y mantenimiento de datos personales. Los datos personales son solo accesibles preferiblemente en una sesión personalizada. El dispositivo servidor comprende una unidad (410) de almacenamiento que porta un primer programa que posibilita iniciar una sesión personalizada. La unidad (410) de almacenamiento puede ser cualquier medio de almacenamiento masivo, tal como un disco duro o una agrupación de discos duros. En caso de múltiples discos duros, los discos pueden situarse en ubicaciones diferentes y se pretende que el dispositivo (404) de servicio según la invención abarque todos los dichos discos duros. La unidad (410) de almacenamiento porta programas adicionales para que se ejecuten en una ventana de navegador en el dispositivo (402) cliente. Los programas adicionales podrían estar en cualquier lenguaje de ordenador que es adecuado para ejecutarse en una ventana de navegador. Por medio de los programas adicionales un usuario puede iniciar una diversidad de aplicaciones de portal, tales como correo electrónico, facturas, control de gastos, reservas, navegación web, etc.

50 El dispositivo (402) cliente está dispuesto para recuperar un programa adicional del dispositivo servidor y comprende una unidad (408) de memoria dispuesta para almacenar dicho programa adicional. Ejemplos de un dispositivo cliente son PC, ordenadores portátiles, notebook, PDA, teléfono móvil, etcétera. La unidad (408) de memoria podría ser una memoria interna o un disco duro o cualquier otro dispositivo de almacenamiento de datos. El dispositivo (402) cliente comprende además una unidad (406) de procesamiento dispuesta para ejecutar dicho programa adicional.

5 Por medio del primer programa un usuario inicia una sesión personalizada y genera un testigo indicativo de dicha sesión personalizada. Por medio de un programa adicional el usuario puede utilizar una aplicación de portal. El programa adicional asocia la ventana de navegador en la que se inicia la aplicación de portal, con el testigo de la sesión personalizada correspondiente. El programa adicional posibilita además verificar iterativamente la validez del testigo y cierra la ventana de navegador con dependencia de la validez de dicho testigo. El testigo podría ser una cookie almacenada en la unidad (408) de memoria del dispositivo cliente. Por consiguiente, el primer programa está dispuesto para almacenar el testigo en la unidad (408) de memoria del dispositivo (402) cliente y el segundo programa está dispuesto para verificar la validez de dicho testigo almacenado en la unidad (408) de memoria. El testigo también podría ser una página web almacenada en la unidad (410) de almacenamiento. Por consiguiente, el primer programa está dispuesto para almacenar el testigo en la unidad (410) de almacenamiento del dispositivo (404) servidor y el segundo programa está dispuesto para verificar la validez de dicho testigo almacenado en la unidad (410) de almacenamiento.

15 Anteriormente se han descrito varias realizaciones de la invención por medio de realizaciones a modo de ejemplo. Pueden realizarse diversas modificaciones y variaciones para los elementos descritos con respecto a estas realizaciones por el experto en la técnica sin alejarse del alcance de la presente invención, que se define por medio de las reivindicaciones adjuntas. Por ejemplo, el término ventana de navegador no se pretende que se limite a un programa para vista de documentos HTML, pero podría ser cualquier software para tener múltiples ventanas en una pantalla, en el que en una ventana pueden ejecutarse múltiples tareas simultáneamente. En este caso no hay necesidad de generar el primer y segundo marco en una ventana de navegador.

REIVINDICACIONES

- 5 1. Método para controlar ventanas de navegadores múltiples que pertenecen a la misma sesión personalizada de un portal web en el que una sesión personalizada se refiere a un intercambio de datos entre un ordenador cliente y un proveedor del servicio, comenzando cuando se establece la conexión iniciando sesión mediante autenticación de un usuario en el portal web y finalizando cuando termina la conexión cerrando sesión en el portal web y en el que se genera un testigo tras autenticación con éxito del usuario, las siguientes etapas se realizan para cada una de las ventanas de navegadores múltiples:
- asociar la ventana de navegador al testigo indicativo de la sesión personalizada;
 - esperar iterativamente un periodo de tiempo y verificar la validez de dicho testigo; y
 - 10 - cerrar la ventana de navegador cuando se detecta que el testigo no es válido
- en el que el método se caracteriza porque cada una de las ventanas de navegador abiertas verifica el mismo testigo y el testigo se vuelve no válido si el usuario ha cerrado sesión de la sesión personalizada, en el que cada una de las ventanas de navegador abiertas proporciona una posibilidad de cerrar sesión de la sesión personalizada del portal web y en el que se usa una cookie como un depósito de testigo.
- 15 2. Método según la reivindicación 1, que comprende además la etapa de: al eliminar dicho testigo se desconecta cerrando sesión de dicha sesión personalizada.
3. Método según la reivindicación 1, que comprende además la etapa de generar un marco para realizar una verificación de la validez de dicho testigo iterativamente.
4. Método según la reivindicación 3, en el que dicho marco es un marco invisible.
- 20 5. Hardware de almacenamiento legible por ordenador que almacena instrucciones ejecutables por ordenador, instrucciones que cuando se ejecutan en un sistema de ordenador realizan el método para controlar ventanas de navegadores múltiples según una cualquiera de las reivindicaciones 1 a 4.

Fig 1

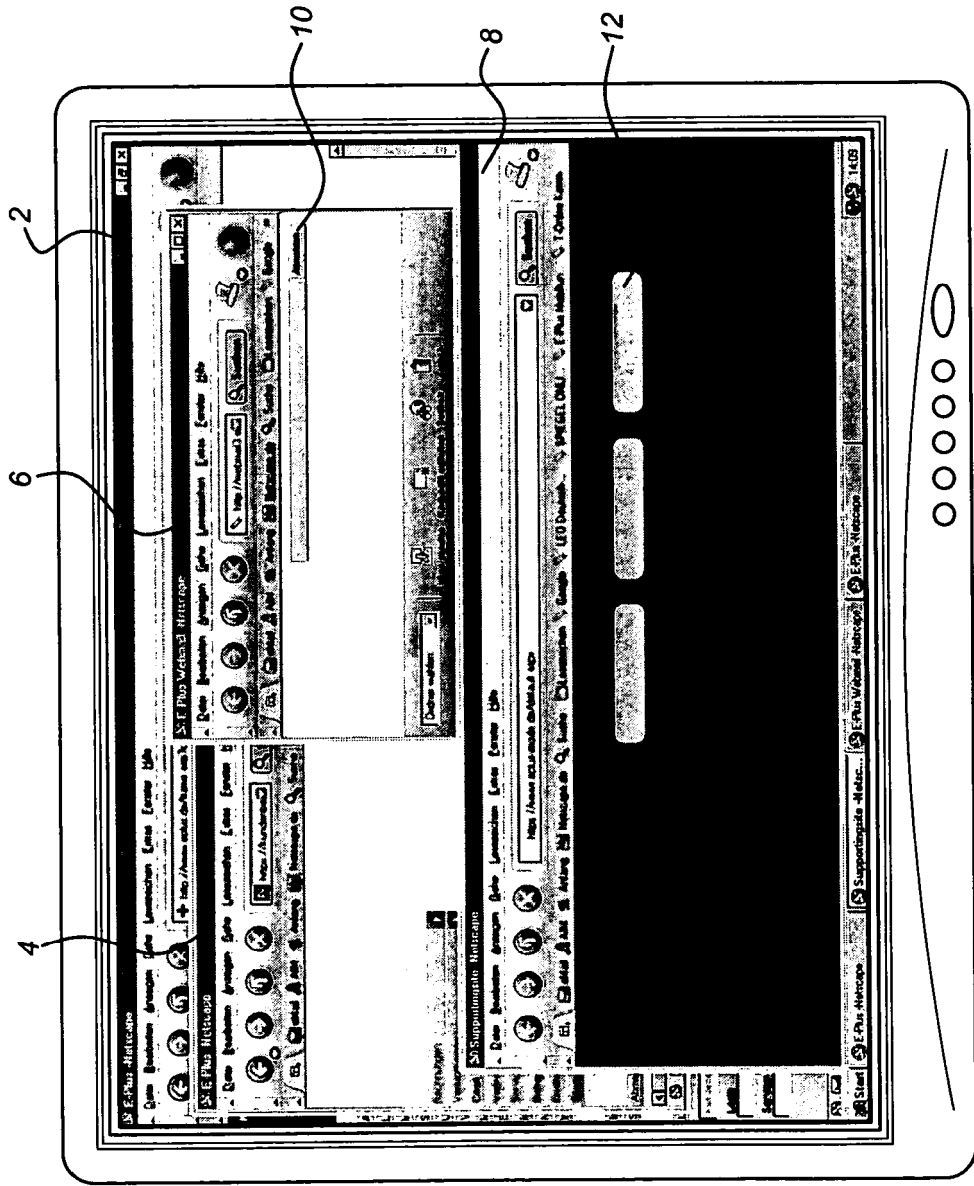


Fig 2

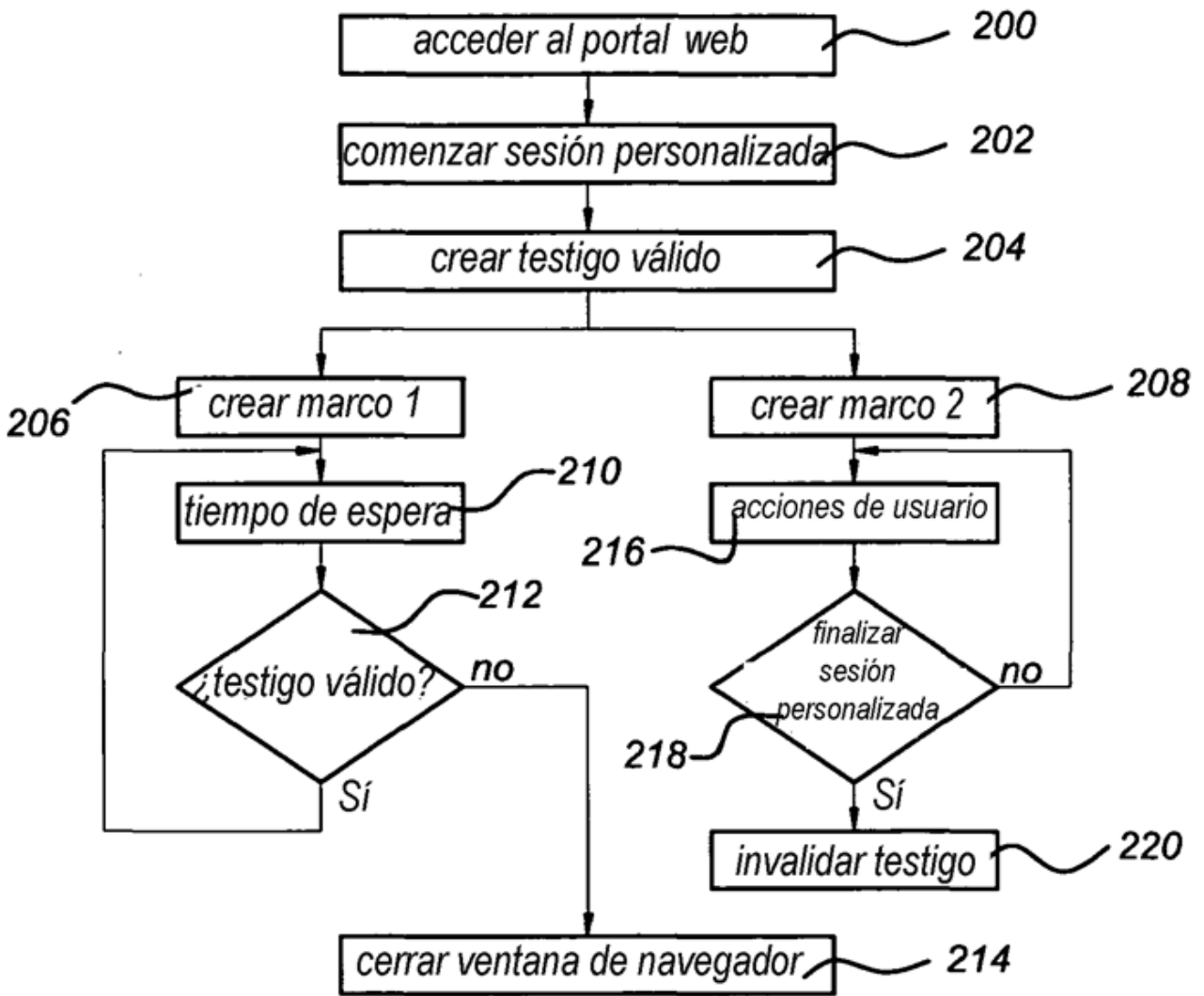


Fig 3

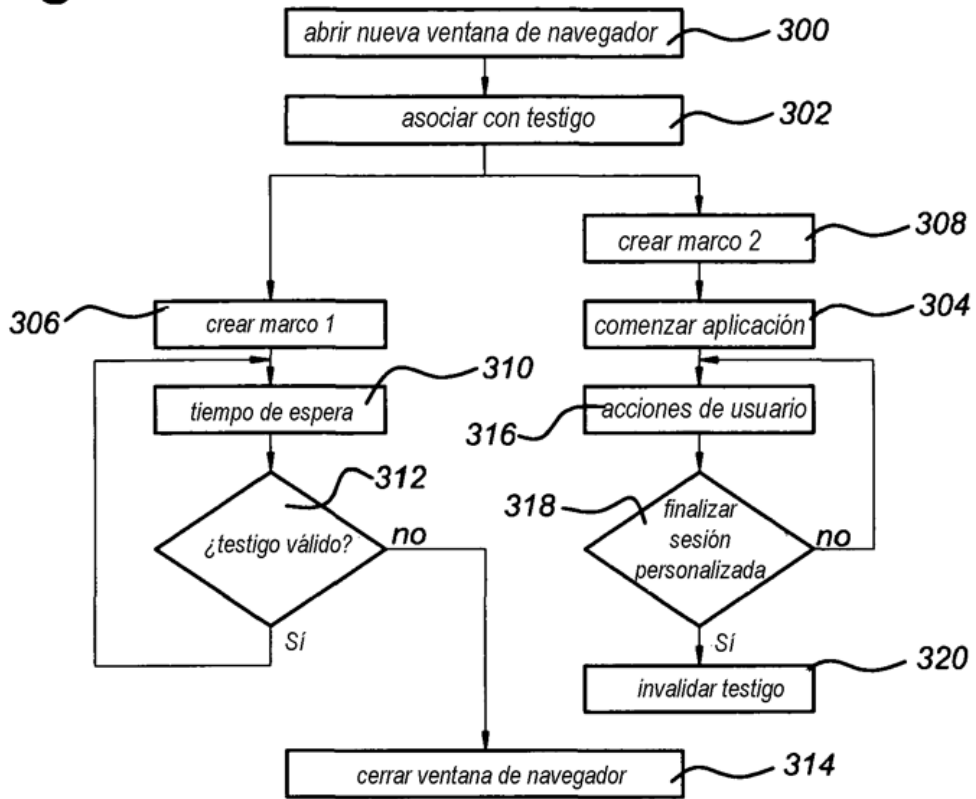


Fig 4

