



NETIC

Tu aplicación de citas favorita

Teresa González García

Alejandro Chávez Macías

3º GITT + ADE

Netic

1. INTRODUCCIÓN	3
2. ESTRUCTURA DE LA BBDD	3
3. ESTRUCTURA CONTROLADOR USUARIO	4
4. ESTRUCTURA CONTROLADOR ADMINISTRADOR	4
5. ESTRUCTURA VISTA COMÚN AL USUARIO Y ADMINISTRADOR	5
6. ESTRUCTURA VISTA CUENTA USUARIO	9
7. ESTRUCTURA VISTA ADMINISTRADOR	16
8. ESTRUCTURA PATRÓN DAO	19
9. OTROS ASPECTOS	20

1. INTRODUCCIÓN

Este proyecto consiste en el diseño de una aplicación de citas sencilla, con una interfaz de usuario fácil de utilizar para cualquier usuario. Dentro de la misma, se distinguen cuentas de tipo usuario y tipo administrador, con diferentes funcionalidades cada uno.

En el lado del usuario, el mismo podrá una vez registrado acceder a su perfil, el cual podrá cambiar en todo momento, ver perfiles de otros usuarios e interactuar con ellos de forma básica. Además, antes de cerrar sesión podrá ver un resumen de su actividad en la página.

En el lado del administrador, este podrá gestionar toda la información recogida en las BBDD de los distintos usuarios, además de editar sus perfiles, darles de baja o concederles permisos cuando sea conveniente, por lo que la aplicación admite varios administradores.

En cuanto al diseño (vista) de la aplicación, se pretende con él en todo momento captar la atención del usuario con sus vivos colores y utilizando muchas imágenes y testimonios de otros usuarios. Además, se aporta mucha información sobre el funcionamiento de la aplicación en la página principal, con el objetivo de aclarar dudas a aquellos futuros clientes, como la gestión de datos o utilización de cookies. Otras aplicaciones existentes, como Meetic o Tinder, han servido como inspiración para diseñar esta aplicación, ya que debido a su interfaz tan accesible, son las aplicaciones que predominan en el mercado.

El funcionamiento de nuestra aplicación se basa en la capacidad de encontrar la persona más afín al usuario mediante filtros de búsqueda exhaustivos, en los cuales el usuario puede seleccionar aquellos criterios que le interese, y la lógica interna de la aplicación le mostrará los perfiles que coincidan con dichos criterios, pudiendo realizarse ilimitadas búsquedas e iteraciones con otros usuarios por sesión. Para ello, utilizamos varias BBDD con muchos campos que recogen el perfil detallado de cada uno de los usuarios, por lo que la parte de conexión con la BBDD y gestión de los datos es el motor de nuestra aplicación.

2. ESTRUCTURA DE LA BBDD

Nuestra BBDD consiste en 3 tablas con distintas finalidades, todas ellas relacionadas mediante una foreign key: user, el nombre de usuario (el cual es único para cada cuenta).

1. **Tabla usuario:** recoge los campos básicos y obligatorios del registro de un nuevo usuario; la información necesaria para poder darse de alta.
2. **Tabla usr_extra:** recoge campos extras de información del usuario para completar su perfil y poder realizar búsquedas más exhaustivas. Esta información es opcional.
3. **Tabla usr_info:** recoge la actividad más importante del usuario durante la sesión una vez se desconecta de la misma. Esta información será utilizada por el administrador para gestionar resultados.

3. ESTRUCTURA CONTROLADOR USUARIO

- **ServletAcceso:** gestiona el acceso a la página, si el usuario está ya registrado le dirige a la página principal creando una nueva sesión, pero si no lo está la redirige a una página de error.
- **ServletRegistro:** gestiona dar de alta un nuevo usuario, si ha ocurrido un problema en el registro como que el nombre de user ya este utilizado, redirige a una página de error, y si ha ido todo bien le dirige a la página principal, creando una nueva sesión.
- **ServletController:** es el servlet principal al que vuelcan la información muchos servlets como el Servlet Búsqueda , que hace múltiples funciones como redirigir a la página principal, mostrar usuarios en inicio y en búsqueda o redirigir al perfil del usuario.
- **ServletBúsqueda:** gestiona los criterios de búsqueda y el acceso a la BBDD según los mismos, además recoge los resultados de las búsquedas.
- **ServletUpdateUsuario:** gestiona la actualización de la información del perfil del usuario a la BBDD.
- **ServletImagen:** gestiona las actualizaciones de las imágenes aparte.
- **ServletCarrito:** es un servlet también con muchas funciones, ya que redirige a perfiles públicos y gestiona la actividad de los 4 tipos de carritos mostrando la información en la página de resumen de actividad.
- **ServletAjax:** se encarga de las peticiones asíncronas de likes, mensajes y de la gestión de sus carritos, además de borrar registros de actividad.
- **ServletDesconexión:** es un servlet que es muy importante, ya que vuelca la actividad de cada usuario en el momento de desconexión en la BBDD, además de invalidar la sesión y redirigir a index.

4. ESTRUCTURA CONTROLADOR ADMINISTRADOR

- **ServletAcceso:** gestiona el acceso a la página, si el usuario está ya registrado le dirige a la página principal creando una nueva sesión, pero si no lo está la redirige a una página de error.

- **ServletController:** es el servlet principal que distingue si la cuenta es tipo usuario o tipo administrador. Si es del último tipo, redirigirá a unas páginas de gestión no accesibles para el usuario.
- **ServletUpdateUsuario:** gestiona la modificación del perfil de algún usuario a la BBDD por parte del administrador.
- **ServletImagen:** gestiona las actualizaciones de las imágenes aparte.
- **ServletAdministrarPerfil:** se encarga de gestionar las peticiones del administrador en cuanto a ver perfiles públicos o editarlos.
- **ServlePermisos:** se encarga de gestionar los permisos dados por el administrador logueado en ese momento a cualquier perfil (ya que la app admite multiadministradores), cambiando el tipo de dicho perfil en la BBDD.
- **ServletEstadísticas:** se encarga de recoger los registros de actividad de los usuarios y visualizarla y gestionar los mismos.
- **ServletAjax:** se encarga de borrar los registros de actividad de los usuarios en la BBDD (estadísticas).
- **ServletDesconexión:** Invalida la sesión y redirige a index. Ya que la cuenta es de tipo administrador, no se vuelca ningún tipo de información en la BBDD.

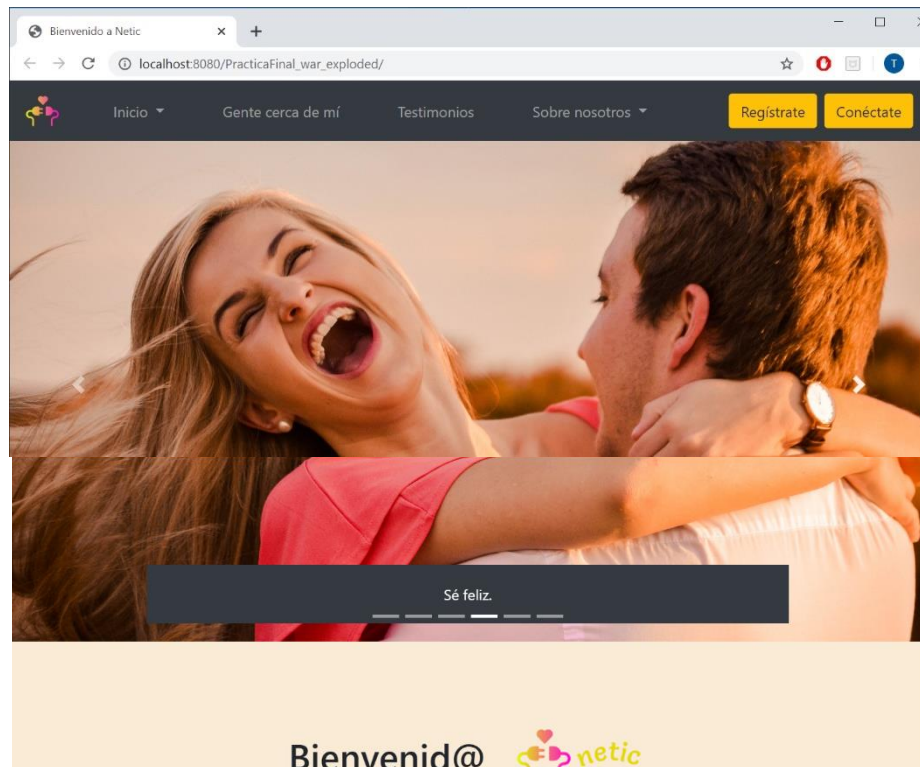
5. ESTRUCTURA VISTA COMÚN AL USUARIO Y ADMINISTRADOR

1. PÁGINA PRINCIPAL (index.html)

URL: http://localhost:8080/PracticaFinal_war_exploded/

El diseño de la vista está fundamentado en Bootstrap, por lo que es responsive para cualquier dispositivo . Se trata de una página muy accesible ya que está basada en la fácil navegación por la misma mediante muchos enlaces, y se permite acceder o registrarse desde diferentes puntos de la página. Además, tiene un carrusel principal de fotos en pareja que llama la atención al usuario.

En ella se muestra información detallada sobre el funcionamiento de la misma , se aportan consejos en cuanto al registro y se aportan datos de contacto y políticas de privacidad además de testimonios.



¿Por qué Netic?

Encuentra la persona adecuada para ti

Nuestros usuarios buscan encontrar la pareja ideal con la que escribir sus historias. Nosotros nos encargamos de que así sea. Solo tienes que registrarte.

Regístrate de forma rápida y gratuita

[Regístrate ya!](#)

[Ya estás registrado? Accede!](#)

Casi 2 millones de parejas en toda España se han conocido a través de Netic

[Conoce sus historias](#)

¿Cómo funciona la búsqueda de pareja?

Si hay el perfil perfecto para ti, lo encontrarás aquí.

Pasos a seguir

1. ¡Cuenta quién eres!

Habla sobre ti. Unas cuantas líneas escritas con naturalidad y espontaneidad son suficientes para causar una buena impresión. ¿Por qué no añadir también un toque de humor?

Precisión: cuantos más criterios completes, más oportunidades de que te contacten personas con tus mismos intereses y estilo de vida.

Comparte tus fotos. Crea un álbum que refleje tu personalidad.

2. ¡Encuentra a la persona que estás buscando!

Encuentra a la persona realmente adecuada para ti gracias a la búsqueda

Si tus intereses cambian siempre podrás modificar tu perfil. No te olvides de darle like a aquellos perfiles que te gusten!

3. Contacta

¿No sabes cómo empezar una conversación? Una buena forma de romper el hielo es hablar sobre los detalles que te atrajeron de su perfil o de cosas que tengáis en común. O compartir un enlace de una canción o un GIF. ¡Desmárcate y déjate llevar!



Q. Ponte a buscar en 3, 2, 1 ...¡ya!

En Netic, los perfiles son muy detallados.

Encuentra a quien encaja contigo usando la búsqueda por criterios o déjate guiar por nuestras sugerencias de perfiles.



Echa un vistazo a los perfiles que más te interesen. Los filtros de Netic harán que te encuentres con el perfil que andabas buscando.



Claudia Vázquez

Hola! Me llamo Claudia, y tengo 24 años. Soy arquitecta y vivo en Madrid. Soy deportista y me gustan mucho los animales.

[See Profile](#)



Conoce a personas solteras en tu zona

En **Barcelona, Madrid, Bilbao, Sevilla...** ¡Dondequiera que estés, hay gente esperando conocerte! **Si la proximidad es un requisito importante para ti, añade tu localización en tu criterio de búsqueda.**

¿Todavía no te hemos convencido? ¡Regístrate ahora y compruébalo!
El amor se encuentra en todas partes, en donde menos te lo esperas.

[Regístrate ya!](#)



ME ENCANTA ESTE PERFIL! ♥

♥ Nuestras historias de éxito.

Cada mes, más de 2000 historias reales comienzan gracias a Netic.



Carlos y Lucía Publicado el 18 de Febrero, 2019

Nos conocimos el 17 de junio de 2017, tras estar dos intensas semanas intercambiando mensajes en Netic y posteriormente con largas llamadas telefónicas.

Conectamos desde el primer minuto, ya que teníamos muchísimas cosas en común, y siempre había tema de conversación.

Lucía: me atrajo de Carlos su maravillosa sonrisa y su mirada porque me transmitió mucha sinceridad.

Carlos: me atrajo de Lucía que tras leer su perfil sabía que detrás había una persona especial y diferente.



Juan y Laura Publicado el 5 de Junio, 2019

Nos conocimos el 6 de enero de 2018. Nos atrajo la descripción, los valores y el estilo de vida que queríamos llevar los dos, ya que ambos estábamos buscando formar una familia.

Todos los días tenemos algún detalle, ya sea un mensaje, una llamada o una foto. El secreto de que nuestra relación funcione tan bien es la paciencia.

Y muchas más! ¿Preparado para comenzar la tuya?

@ Sobre nosotros

! Ayuda

Si no consigues darte de alta, pruebe con otro nombre de usuario y contraseña. Por otro lado, si se ha equivocado al rellenar su perfil o quiere cambiarlo en algún momento siempre tendrá la posibilidad de hacerlo.

Por otro lado, cada vez que inicie sesión, no olvide de cerrarla o la información persistirá en el navegador. Además, si quiere borrar su cuenta siempre podrá hacerlo, y volver a registrarse cuando quiera.

! Privacidad

Nuestro compromiso es la seguridad de nuestros usuarios. Los perfiles se revisan y tus datos personales están protegidos.

Usted elige darnos información cuando se da de alta, y completa su perfil. Además, procesamos sus conversaciones. Por otro lado, utilizamos una política de cookies para reconocerle a usted y a sus dispositivos.

Cualquier duda sobre el tratamiento de sus datos, no dude en contactar con nosotros.

! Contacto

C/ Sol, 2
Madrid, España
28013
Teléfono: 912345678
neticempresa@gmail.com

2. ACCESO

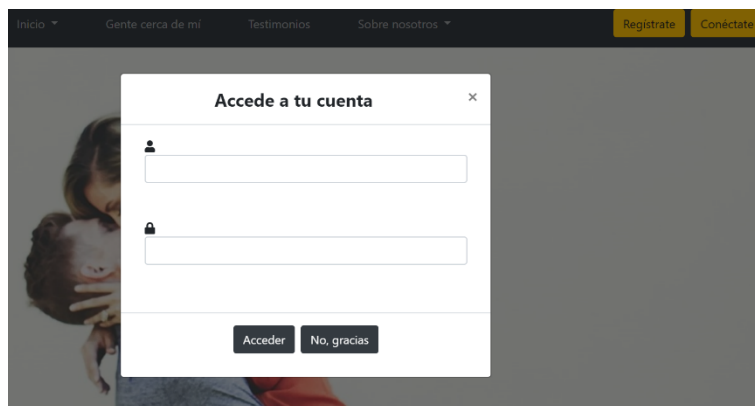
Y

REGISTRO

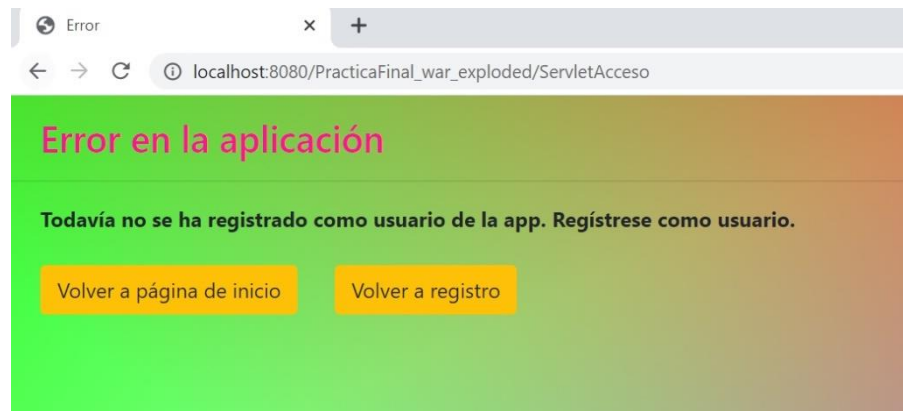
(registro.jsp)

URL: http://localhost:8080/PracticaFinal_war_exploded/registro.jsp

El acceso a la página como usuario o administrador indistintamente es muy sencillo y se puede hacer desde la página principal. Según el tipo de cuenta que sea ya la lógica (ServletController) redirigirá a una página u a otra, diseñadas con combinación de Bootstrap + CSS.



La lógica (ServletAcceso y ServletRegistro respectivamente) gestiona posibles errores como que el usuario intente acceder sin haberse registrado previamente o que se registre con un usuario no permitido. Por ejemplo:



En cuanto al registro, inicialmente se muestra un formulario cuyos campos son todos obligatorios ya que son la información que se guardará en la tabla usuario.

A screenshot of a web browser showing a registration form. The browser's address bar displays 'localhost:8080/PracticaFinal_war_exploded/registro.jsp'. The page has a green-to-blue gradient background. At the top, it says 'Regístrate ya!' in pink, followed by a logo. Below this, there are four sections: 'Información general' with fields for 'Nombre', 'Apellidos', 'Sexo' (radio buttons for 'Hombre' and 'Mujer'), and 'Fecha de nacimiento' (a date picker showing 'dd/mm/aaaa'); 'Datos de acceso' with fields for 'Usuario', 'Contraseña', 'Confirmar contraseña', and 'Correo electrónico' (pre-filled with 'emailusuario@gmail.com'); 'Intereses generales' with a 'Busco...' section (radio buttons for 'Hombre', 'Mujer', and 'Ambos') and a 'Cuéntanos algo de ti' section (a large text area); and a bottom section with a '0/200' character count and 'Registrar' and 'Borrar' buttons.

Si la información insertada es válida (puede repetirlo tantas veces como quiera), se desplegará un registro extra sobre la información opcional guardada en la tabla `usr_extra`, donde se pregunta al usuario por diferentes aspectos: estado civil, condición física, hijos, ciudad... El usuario puede dejar sin responder cualquiera de los campos, siendo posible rellenarlos cuando le plazca más adelante.

Cabe destacar que por defecto se crea una cuenta de usuario. Ya que la aplicación soporta varios administradores, será un administrador el que tenga que darle permisos a la cuenta de usuario en cuestión.

En ambas páginas hay validación tanto por parte del cliente (javascript) como por parte del servidor (servlet), las cuales se explicarán más adelante.

6. ESTRUCTURA VISTA CUENTA USUARIO

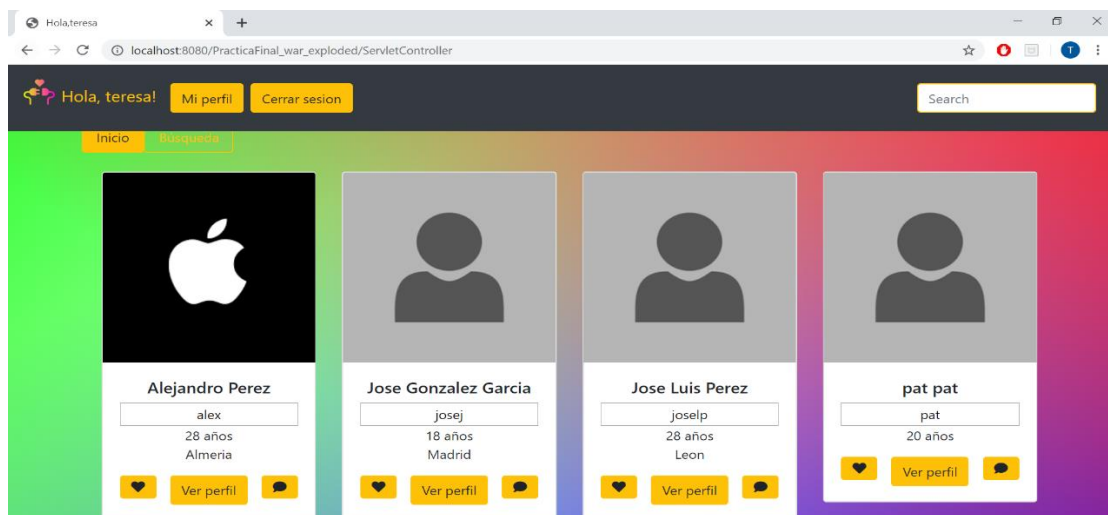
1. PÁGINA PRINCIPAL USUARIO (principal.jsp)

URL : http://localhost:8080/PracticaFinal_war_exploded/ServletController

Esta página es el motor de actividad del usuario, la cual se divide en dos subpáginas, diseñadas mediante Bootstrap + CSS. . Ambas páginas tienen validación de lado del cliente y servidor.

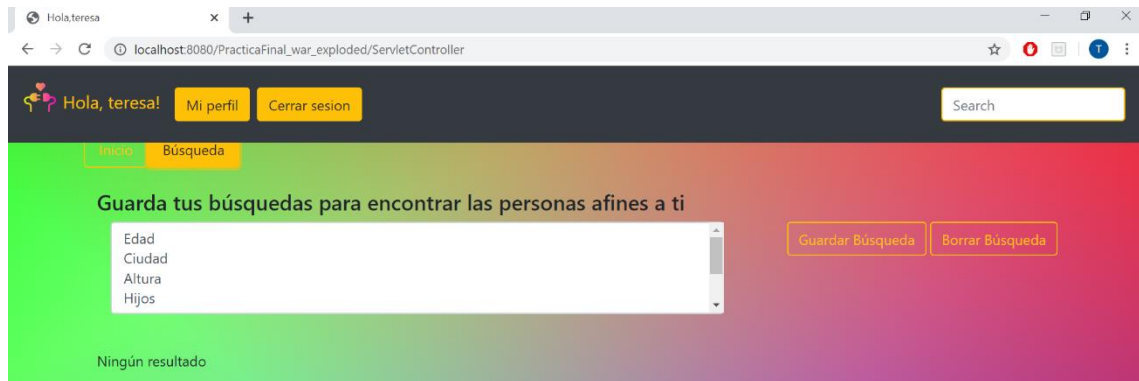
- **INICIO**

En esta página se muestran al usuario todos los perfiles dados de alta acordes a su preferencia (la cual podrá cambiar en cualquier momento) y que además, tengan gustos coincidentes con el sexo del propio usuario. Es decir, garantizamos que la interacción por ambas partes resulte satisfactoria ya que ambos usuarios buscan lo mismo.



- **BÚSQUEDA**

Como se ha explicado en la introducción, la potencia de la aplicación reside en el buscador de perfiles, que se basa en 7 criterios combinables entre ellos, por lo tanto el usuario puede realizar hasta 127 búsquedas diferentes. Los criterios son: **edad (entre 2 edades determinadas), ciudad, altura (entre 2 alturas determinadas), hijos, estado civil, condición física y aficiones**. Mediante código Javascript+ JQuery, cada vez que se seleccione un criterio de la búsqueda (selección múltiple), aparecerá una sección con las opciones de dicho criterio, desapareciendo al quitar la selección, combinándose 127 veces.



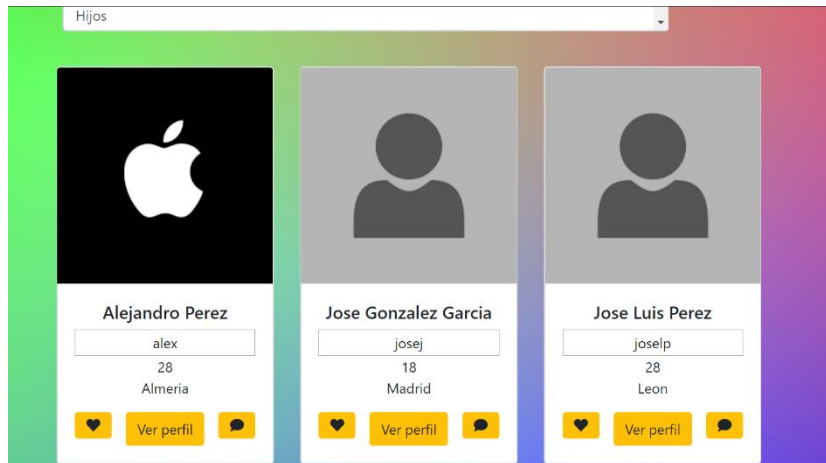
El número de búsquedas es ilimitado, y se irán guardando al pulsar “Guardar Búsqueda”. Se podrá repetir la búsqueda si no se considera válida por el lado del cliente pulsando “Borrar Búsqueda”.

Esta búsqueda personalizada es controlada por ServletBúsqueda, cuya lógica accede a la BBDD y muestra los perfiles que coincidan de forma exacta con los criterios si existen, si no mostrará “Ningún resultado”. Como se puede ver, hay combinación de criterios obligatorios y opcionales, por lo tanto puede ocurrir que cierto usuario no haya rellenado los criterios opcionales por lo que no aparezca como resultado de búsqueda. Sin embargo, como podrá editar su perfil en cualquier momento, podrá aparecer en futuras búsquedas.

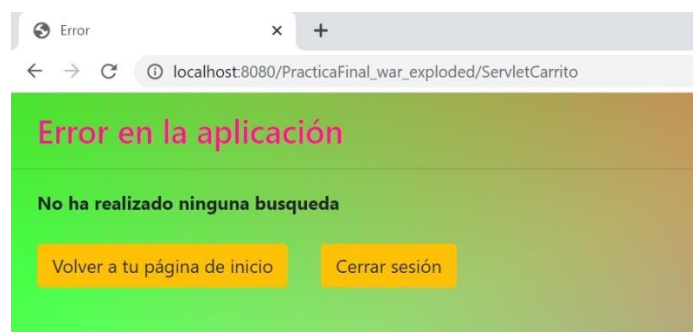
Como en la página inicio, solo se mostrarán perfiles que coincidan con la preferencia del usuario y cuya preferencia sea el sexo del mismo usuario.

RESULTADO DE
BÚSQUEDA:

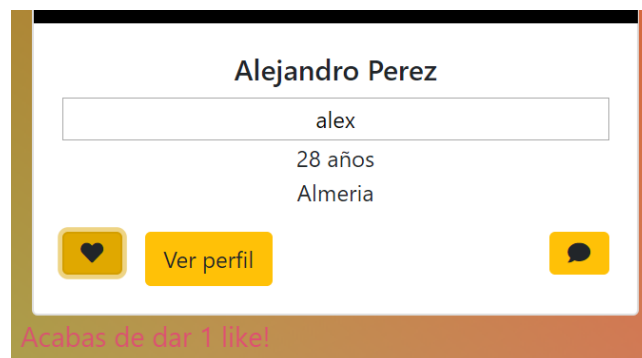
Hijos = No



Si ocurre algún error en la página, la validación del servidor redirigirá a una página de error:



En el listado de usuarios de ambas páginas se puede ver una descripción breve del perfil (nombre y apellidos, edad, ciudad) y su foto si la tiene. La descripción completa (información básica+ extra) se podrá ver si el usuario accede al perfil público (botón ver perfil). Además, se puede también interactuar de forma sencilla con el/los perfil/perfiles que se desee, dándole like o enviando un mensaje. Estas acciones son controladas por Ajax (ServletAjax):



Por otro lado, toda actividad que realice el usuario en esta página y en las siguientes (visitas, búsquedas, likes, mensajes) será gestionada por el ServletCarrito y será accesible para el usuario, lo que se explicará más adelante.

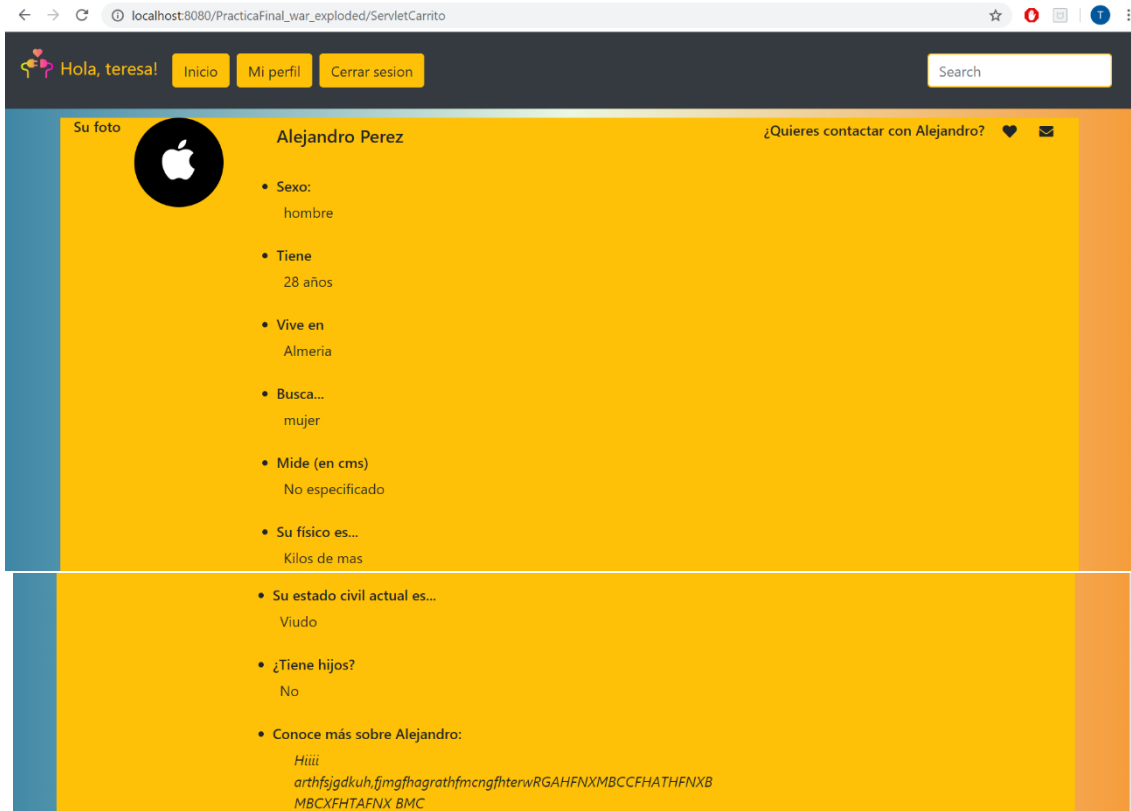
Finalmente, en todo momento el usuario puede buscar cualquier palabra o elemento que quiera en la barra de búsquedas, además de acceder a su perfil o cerrar sesión. La mayoría de las funcionalidades de esta página debido a su mayor complejidad han necesitado de jQuery.

2. VER PERFIL PÚBLICO (verperfil.jsp)

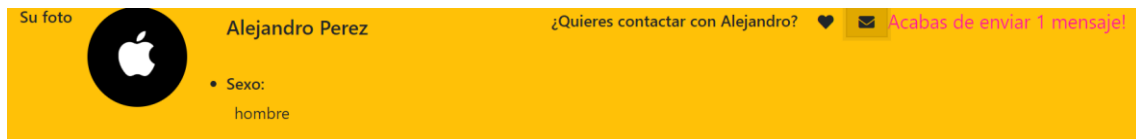
URL: http://localhost:8080/PracticaFinal_war_exploded/ServletCarrito

Esta página muestra la información al completo (obligatoria + extra) disponible del perfil en cuestión y es gestionada por ServletCarrito cada vez que desde la página principal (tanto en inicio como búsquedas) se pulsa al botón “ver perfil” .

En ese momento se añadirá una visita y el perfil visitado a la actividad del usuario. Dentro de la misma página, también se puede dar like o enviar un mensaje (gestionado por ServletAjax), añadiendo estas iteraciones a su actividad.



Actividad controlada por ServletAjax con peticiones asíncronas:



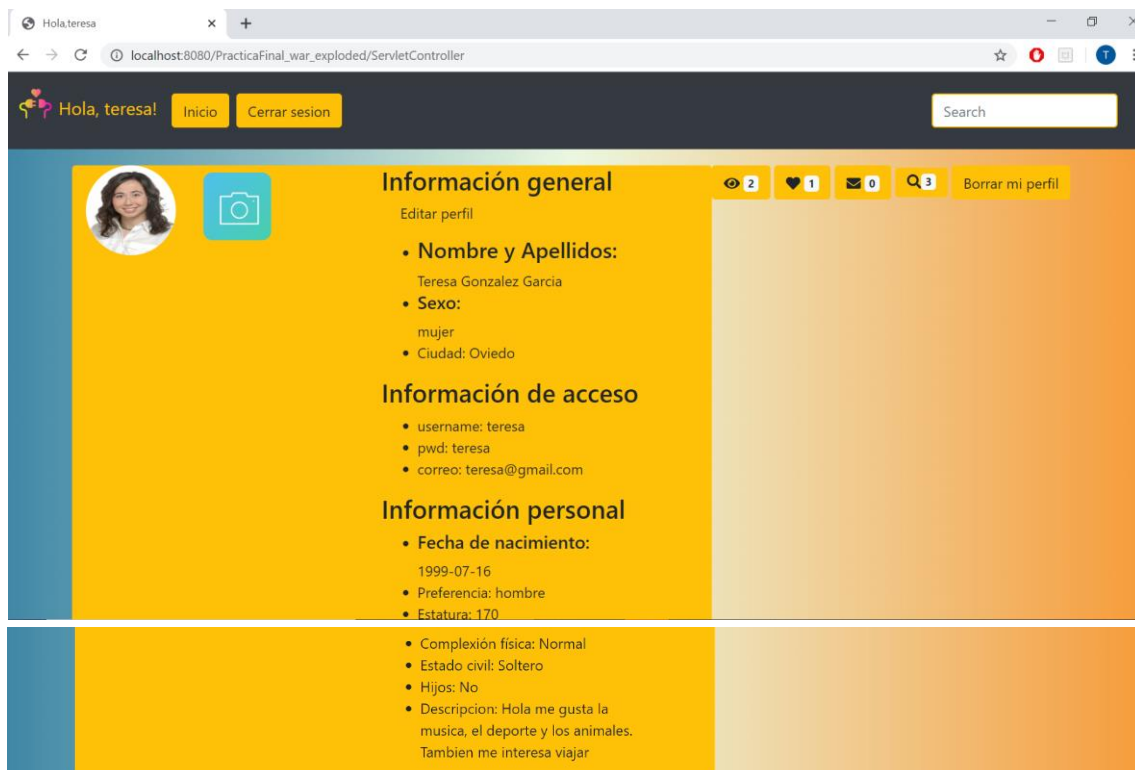
De nuevo, se podrá acceder al perfil de usuario, a la página principal o cerrar sesión, además de la barra de búsqueda.

3. PERFIL USUARIO (perfil.jsp)

URL: http://localhost:8080/PracticaFinal_war_exploded/ServletController



En esta página se mostrará toda la información disponible (obligatoria + extra) del usuario. Además, puede acceder a los registros de su actividad pulsando los distintos botones, ya que como hemos dicho son accesibles por el usuario. Se puede ver un número al lado de los iconos, que corresponde al número de iteraciones de cada tipo que ha hecho hasta el momento, y se va actualizando automáticamente. Estos registros son carritos de likes, mensajes, visitas y búsquedas gestionados por ServletCarrito. Por otro lado, puede borrar su perfil, lo que hará que se salga de la aplicación y se muestre index.html.

El usuario puede en todo momento filtrar palabras con la barra de búsqueda, a la vez que volver a la pagina principal (ServletController) o cerrar sesión (ServletDesconexión).



Por otro lado, el usuario como hemos dicho puede editar su propio perfil (no otros perfiles), mediante el botón “Editar Perfil”. Esta funcionalidad está gestionada por ServletUpdateUsuario al pulsar “Guardar cambios”, y se editará la información correspondiente de la BBDD. Como se muestra en las imágenes, no toda la información se puede cambiar (user, fecha, sexo). Esta información es validada por el cliente y servidor.

Además, el usuario puede actualizar su foto dándole al botón azul y a “Actualizar Foto”; esta función la controla ServletImagen, cambiando la BBDD. Se previsualiza su foto actual en todo momento.

Información general

[Editar perfil](#)

- Nombre y Apellidos:**
 Teresa Gonzalez Garcia
 Nombre:
 Apellidos:
- Sexo:**
 mujer
- Ciudad:** Oviedo

Información de acceso

- username: teresa
- pwd: teresa
- correo: teresa@gmail.com

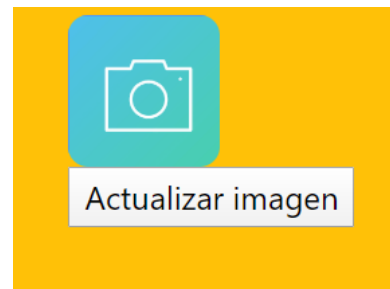
Información personal

- Fecha de nacimiento:**
 1999-07-16
- Preferencia:** hombre
☐ Hombre
☐ Mujer
☐ Ambos
- Estatura:** 170
 140 210
- Complexión física:** Normal

- ☐ Delgada
- ☐ Atlético
- ☐ Normal
- ☐ Algunos kilos de más
- ☐ Corpulenta
- Estado civil:** Soltero
☐ Soltero
☐ Separado
☐ Divorciado
☐ Viudo
- Hijos:** No
☐ Sí
☐ No
- Descripción:** Hola me gusta la musica, el deporte y los animales. Tambien me interesa viajar
Cuéntanos algo de ti

0/200

[Guardar cambios](#)



4. ACTIVIDAD DE SESIÓN DEL USUARIO (listadoCarrito.jsp)

URL: http://localhost:8080/PracticaFinal_war_exploded/ServletCarrito

Como se ha explicado anteriormente, cada iteración que haga el usuario en su sesión se irá guardando progresivamente en cuatro tipos de carritos : visitas, likes, mensajes y búsquedas. Esta información es accesible al usuario y puede verla e incluso borrarla antes de cerrar sesión si le da a ciertos botones en su perfil.

Cada vez que se pulse uno de ellos (likes, mensajes, visitas, búsquedas), ServletCarrito redirigirá al usuario a esta nueva página e irá mostrando el registro actualizado correspondiente, hasta que se ve la información completa en la página. Si algún carrito está vacío, se mostrará este mensaje:



La información se guarda cada vez que se visita un perfil (ServletCarrito, botón “ver perfil”), cada vez que se realiza una búsqueda (ServletBúsqueda, “botón guardar búsqueda) o mediante peticiones asíncronas (ServletAjax: like, mensaje).

En todo momento el usuario puede volver a la página principal para seguir su actividad (ServletController), a su perfil (ServletController) o cerrar sesión (ServletDesconexión) y también usar la barra de búsquedas.

The screenshot displays a web application interface for a user named 'teresa'. The top navigation bar includes links for 'Inicio', 'Mi perfil', and 'Cerrar sesion', along with a search bar. The main content area is divided into several sections:

- Resumen de tu actividad, teresa**: A summary of the user's activity.
- Tu registro de visitas**: A table showing the number of visits to different profiles.
- Resumen de visitas**: A summary of visits, stating that 2 profiles were visited during the session, with a total of 4 visits.
- Tu registro de búsquedas**: A table showing search results for various criteria.
- Resumen de búsquedas**: A summary of searches, stating that 4 searches were performed.
- Tu registro de likes**: A table showing the profiles of users who liked the current user.
- Tu registro de mensajes**: A table showing the number of messages sent to different profiles.
- Resumen de mensajes**: A summary of messages, stating that 4 profiles were contacted during the session, with a total of 5 messages.

Tu registro de visitas

Perfil	Número de visitas
alex	3
josej	1

Resumen de visitas

Has visitado 2 perfiles distintos durante la sesión, siendo 4 el número de visitas totales.
El perfil más visitado: alex

Tu registro de búsquedas

Búsqueda	Usuarios compatibles
Edad entre 20 y 39 Ciudad: Madrid	
Hijos: No	alex josej joselp
Estado civil: Soltero Aficiones: Cocina	josej joselp
Hijos: No Estado civil: Soltero Condición física: Normal Aficiones: Cocina	joselp

Resumen de búsquedas

Has realizado 4 búsquedas! Sigue buscando para conseguir tu pareja ideal.

Tu registro de likes

Perfil
pat
alex

Tu registro de mensajes

Perfil	Número de mensajes
pat	1
alex	2
joselp	1
josej	1

Resumen de mensajes

Has contactado con 4 perfiles distintos durante la sesión, siendo 5 el número de mensajes totales.
El perfil más contactado: alex

Por otro lado, se ve como en cada registro hay un botón X, el cual si se pulsa se borrará el registro correspondiente, dejando el carrito de dicha iteración vacío. Todo ello lo controla ServletAjax con peticiones asíncronas. Una vez borrado el carrito puede volver a llenarse de nuevas iteraciones de forma ilimitada. La información borrada nunca se guardará en la BBDD.

Ejemplo Registro de visitas borrado:

Resumen de tu actividad,teresa

Registro borrado con éxito!

Tu registro de búsquedas

Búsqueda	Usuarios compatibles
Edad entre 20 y 39	

0 3 5 4 Borrar mi perfil

Esta información que se muestra en las tablas persistirá en el momento en el que se cierre sesión (ServletDesconexión), volcándose en la BBDD usr_info, que será recogida por el administrador. Si hubiese algún error en el proceso se redirige a una página de error.

7. ESTRUCTURA VISTA ADMINISTRADOR

En nuestra página hemos diferenciado dos tipos de usuario: El tipo usuario, con capacidad de tener un perfil público, hacer búsquedas de perfiles con los que interactuar e, interactuar con estos enviando likes, mensajes o visitando su perfil.

Por otra parte, el tipo administrador se encargará de mantener en correcto orden el funcionamiento de la plataforma. El administrador, entre sus funciones, podrá ver la última actividad de los usuarios en la pestaña estadísticas, donde se incluyen los mensajes realizados, visitas a perfiles y likes enviados, junto a la fecha en la que se realizaron.

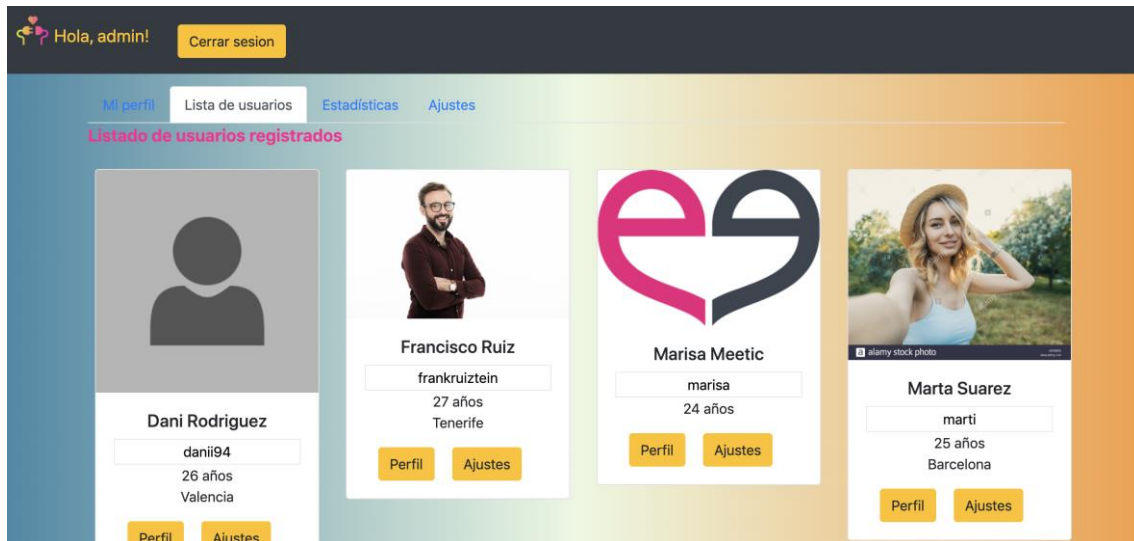
Hola, admin! Cerrar sesion

Mi perfil Lista de usuarios Estadísticas Ajustes

Última actividad de los usuarios

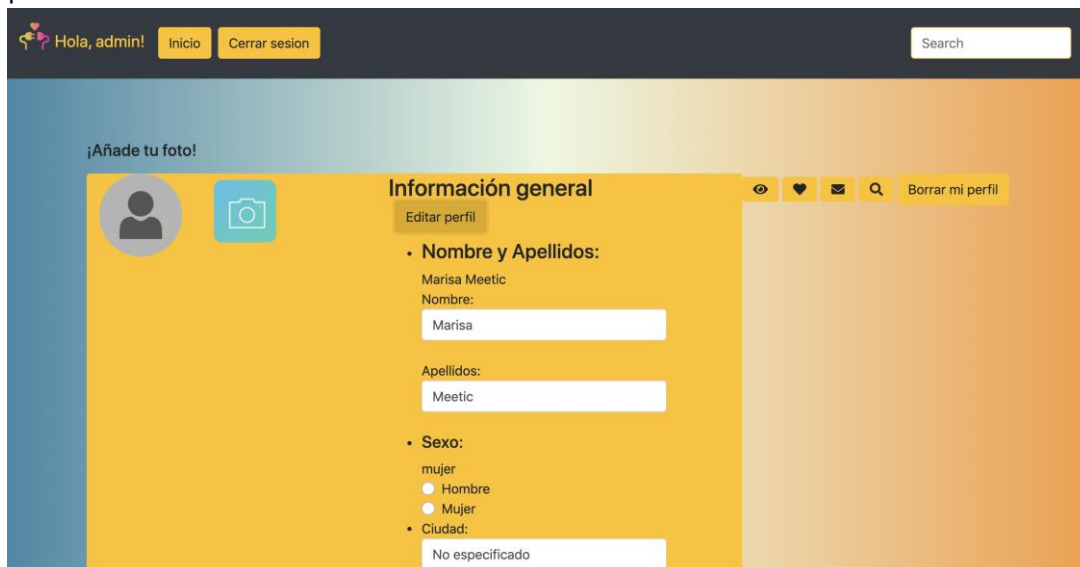
Fecha	Usuario	Visitas	Búsquedas	Likes	Mensajes	Likeados	+ Contactados	+ Visitados
2020-05-08	sara97	1	0	1	1	[frankruiztein]	frankruiztein	frankruiztein
2020-05-08	marti	2	0	2	2	[Toni, frankruiztein]	frankruiztein	frankruiztein
2020-05-08	dani94	1	0	3	1	[sara97, frankruiztein, marti]	marti	marti
2020-05-08	marti	0	0	1	0	[Victorioso]		
2020-05-08	marti	0	0	0	1	[]	Victorioso	
2020-05-08	marti	1	0	0	0	[]		Victorioso

El administrador puede ver la última actividad de los usuarios



Listado de usuarios desde la vista del administrador

Otra función exclusiva de los administradores es que, si lo vieran necesario, podrán editar los perfiles de los usuarios.



Desde una cuenta admin es posible entrar en el perfil privado de un usuario.

Hola, admin! Inicio Cerrar sesion Search

☐ Algunos kilos de más
☐ Corpulenta
 • Estado civil: No especificado
☐ Soltero
☐ Separado
☐ Divorciado
☐ Viudo
 • Hijos: No especificado
☐ Sí
☐ No
 • Descripción: Pues... neetic no estÃ¡ mal... Â¡Pero meetic es otro rollo! RegÃ¡strate ya en meetic.com para encontrar pareja ya mismo.
Cuéntanos algo de ti
 Pues... neetic no estÃ¡ mal... Â¡Pero meetic es otro rollo! RegÃ¡strate ya en meetic.com para encontrar pareja ya mismo.
 0/200
 Guardar cambios

Usar una cuenta para hacer spam podría ser un motivo para que el administrador tome medidas.

Además, si un administrador detectara un comportamiento no adecuado en algún usuario, podrá eliminar su perfil en la pestaña Ajustes. En esta misma pestaña también podrá dar permisos a otros perfiles de tipo usuario.

Mi perfil Lista de usuarios Estadísticas Ajustes

Listado de usuarios registrados

Usuario	Tipo	Estado	Fecha de alta	Gestionar permisos	Eliminar
Francisco	usuario	Activo	2020-05-07	Gestionar permisos	Eliminar
Marisa	usuario	Activo	2020-05-08	Gestionar permisos	Eliminar
Marta	usuario	Activo	2020-05-07	Gestionar permisos	Eliminar
Sara	usuario	Activo	2020-05-07	Gestionar permisos	Eliminar
Antonio	usuario	Activo	2020-05-07	Gestionar permisos	Eliminar
Victor	usuario	Activo	2020-05-07	Gestionar permisos	Eliminar

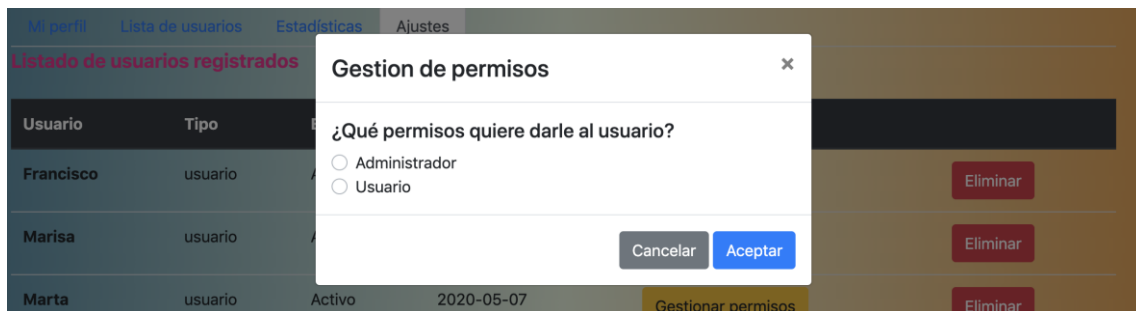
Advertencia

¿Está seguro de que desea eliminar el perfil?

Esta operación no se puede deshacer

Cancelar Eliminar

Ventana emergente al pulsar el botón "Eliminar"



Ventana de gestión de permisos

8. ESTRUCTURA PATRÓN DAO

En esta práctica se utiliza la conexión a la BBDD mysql mediante el **driver tipo 4**.

➤ DOMINIO

Se encuentran en este paquete las clases java necesarias para instanciar diferentes elementos, que se utilizarán en los métodos DAO: Usuario, Búsqueda, AdminInfo(información actividad) y los distintos carritos, los cuales están gestionados por CarritoServicio (paquete servicio).

➤ DAO

- **Conexión:** clase que se instanciará para poder conectarse a la BBDD de mysql mediante el driver tipo 4.
- **UsuarioDAO:** clase que contiene todos los métodos necesarios para obtener la información completa o parcial de un perfil, además de dar de alta o de baja un perfil y saber si ya esta o no registrado. También se encuentran los métodos de actualización de imágenes e información.
- **BusquedaDAO:** clase que contiene métodos que devuelven collections de los usuarios de inicio y de búsqueda mediante funciones eficientes que permiten ir construyendo las queries de forma progresiva.
- **AdminDAO:** clase que contiene los métodos que utiliza el administrador para listar todos los perfiles, borrarlos, mostrar su actividad.

Los servlets correspondientes utilizan los métodos DAO para realizar diferentes funciones. Ya que la gestión de los perfiles de la aplicación y la búsqueda es la base de la aplicación, se realiza un número considerable de consultas a la BBDD, por lo que los DAO tienen un método para cada caso, que será utilizado por el servlet. Pese a tener que hacer muchas consultas, se ha optimizado el número de queries, como se podrá ver en el código. Por ejemplo, aunque tengamos 127 tipos de búsquedas diferentes, se implementa una función que va construyendo una query única según los criterios que pida el usuario. Además, tanto administrador como usuario comparten métodos.

9. OTROS ASPECTOS

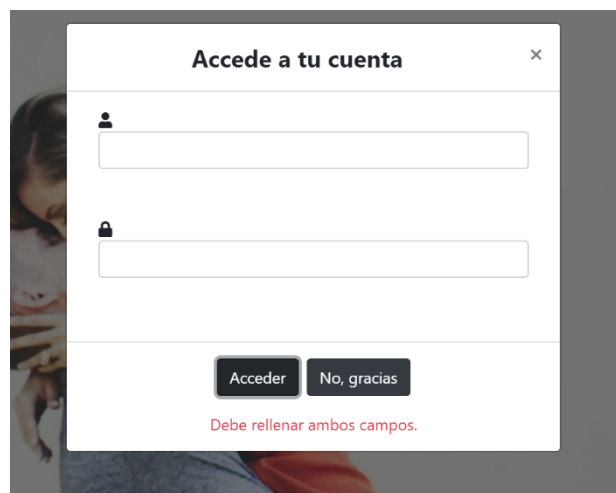
Validaciones

Validaciones por el lado de cliente (Javascript):

Este aspecto está muy reforzado en nuestra página web, para que la experiencia del usuario sea lo más satisfactoria posible. Por ello, se garantiza que los perfiles registrados contengan campos de información coherentes que se ajusten al tipo de dato que se requiere, por ejemplo a una fecha válida. Los mensajes de error aparecen en un tamaño y color llamativo para que sea muy accesible la detección de error para el usuario. Dicho mensaje desaparece en el momento que se borran los campos del formulario correspondiente. Las validaciones más importantes son las siguientes:

- **Acceso**

En esta página, es evidente que para poder acceder hay que rellenar su nombre de usuario y clave para que se pueda buscar si ese usuario está registrado ya en la BBDD (ServletAcceso también hace su validación interna).

Una captura de pantalla de un modal de acceso a la cuenta. El modal tiene el título "Accede a tu cuenta" y un botón de cerrar (X). Contiene dos campos de entrada: uno para el nombre de usuario (con un ícono de persona) y otro para la contraseña (con un ícono de candado). Debajo de los campos hay dos botones: "Acceder" y "No, gracias". En la parte inferior del modal, hay un mensaje de error en rojo que dice "Debe rellenar ambos campos.".

- **Registro**

En esta página como se ha explicado, la validación es muy exigente para asegurarnos por ejemplo que el usuario es mayor de edad o que el usuario confirme su contraseña.

Una captura de pantalla de un formulario de registro con tres mensajes de error en rojo. El primer mensaje dice "Debe rellenar todos los campos para poder registrarse." y aparece cuando los campos "Registrarse" y "Borrar" están vacíos. El segundo mensaje dice "Lo sentimos, debe ser mayor de edad para poder usar netic" y aparece cuando el campo "Registrarse" contiene un valor no válido. El tercer mensaje dice "Las contraseñas no coinciden" y aparece cuando los campos "Registrarse" y "Borrar" contienen valores que no coinciden.

- **Búsqueda**

En esta página también la validación es muy importante para no realizar en ningún momento una búsqueda vacía, lo que provocaría un fallo en la lógica de la app. Se realizan en concreto 127 validaciones. Además, cabe destacar que se ha limitado el número de aficiones a 2.

The image displays four sequential screenshots of a web application's search interface, illustrating various validation rules:

- First Screenshot:** The 'Búsqueda' tab is active. A dropdown menu is open, showing options: 'Edad', 'Ciudad', 'Altura', and 'Hijos'. Below the menu, a message states 'Ningún resultado' (No results). A red error message box says: 'Debe seleccionar algún criterio de búsqueda' (You must select a search criterion).
- Second Screenshot:** The 'Edad' (Age) criterion is selected. The input shows 'Entre 20 y 100 años.' (Between 20 and 100 years). A red error message box indicates: 'El valor debe ser inferior o igual a 99' (The value must be less than or equal to 99).
- Third Screenshot:** The search criteria are fully populated: 'Edad' (18-99), 'Ciudad' (150-200 cms.), and '¿Tiene hijos?' (Si/No). The 'Guardar Búsqueda' (Save Search) and 'Borrar Búsqueda' (Delete Search) buttons are visible. A red message says: 'Debe rellenar todos los criterios seleccionados.' (You must fill in all selected criteria).
- Fourth Screenshot:** The 'Búsqueda' tab is active. A dropdown menu is open, showing options: 'Hijos', 'Estado civil', 'Físico', and 'Aficiones'. Below the menu, a message states 'Ningún resultado' (No results). A red error message box says: 'Solo puede seleccionar 2 aficiones!' (You can only select 2 hobbies!).

- Actualización del perfil

En esta página se garantiza que no se actualicen las BBDD con determinados campos nulos, ya que como se ha explicado hay información necesaria obligatoria.

- ☐ Separado
- ☐ Divorciado
- ☐ Viudo
- ☒ Hijos: No
 - ☐ Sí
 - ☐ No
- Descripción: Hola me gusta la musica, el deporte y los animales. Tambien me interesa viajar.

Cuéntanos algo de ti

Hola me gusta la musica, el deporte y los animales. Tambien me interesa viajar

0/200

Los campos obligatorios no pueden quedar vacíos

Guardar cambios

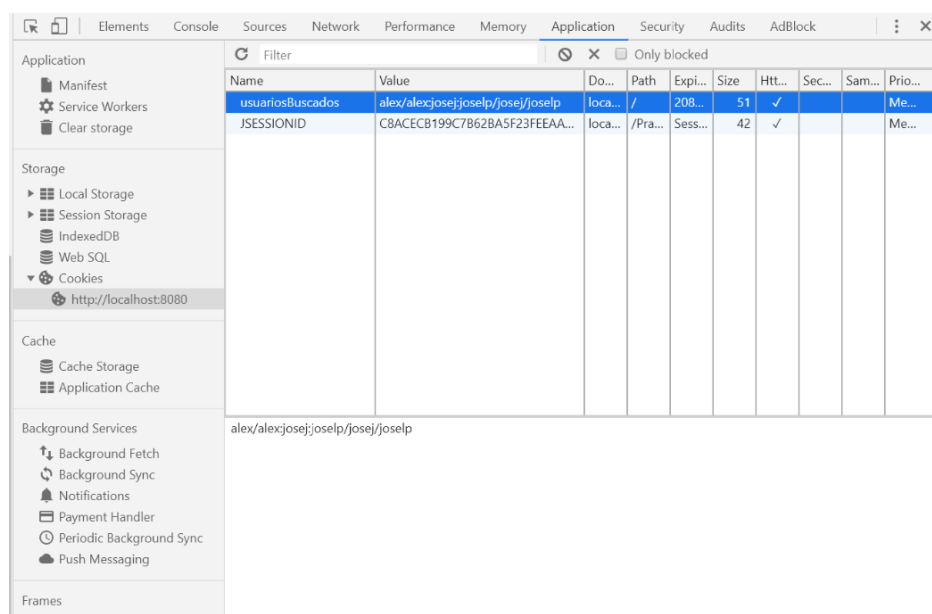
Validaciones por el lado del servidor:

Se comprueba en todo momento la existencia de una sesión, la cual recordamos que expira en 30 minutos si no se detecta ninguna actividad. Por lo que si alguien que no ha accedido a la página intenta entrar mediante la URL, la app le mostrará este mensaje:



Gestión de cookies

Como se puede ver en la imagen, la app guarda (hasta tiempo máximo permitido) los resultados de cada búsqueda realizada por el usuario, siendo en cada una resultado único o múltiple (los términos separados por :). Además, también se ve la sesión iniciada.



Seguridad

Para prevenir posibles ataques de hackers a nuestra página web, tuvimos en cuenta diferentes patrones de diseño.

Prevención ante inyecciones de SQL:

Toda consulta o inserción de datos en la base de datos se realiza mediante PreparedStatement, de tal forma que un hacker no podrá acceder a información sensible a partir de inyecciones de SQL. Con PreparedStatement, parametrizamos las consultas a la base de datos de forma que, si el hacker intenta acceder a la base de datos introduciendo caracteres como comillas para modificar el resto de la query, no podrá, ya que el parámetro no se añade a la query sumando Strings.

Cross-Site Scripting

Para prevenir este tipo de ataques hicimos lo siguiente:

- Por una parte, escapamos el HTML, de forma que no podrá ejecutarse código externo en nuestra página
- Definimos las cookies como HttpOnly en nuestro web.xml

Otras medidas de seguridad que se tomaron:

- No persiste información sensible en las cookies. Por ejemplo, la información del usuario logueado en neetic persiste en la sesión, y esta se cierra al desconectarse o al pasar un tiempo establecido de 30 minutos. Así evitamos el Form Tampering.
- Se valida al usuario cada vez que quiera hacer una consulta en la base de datos.
- Todas las peticiones de formularios con información sensible a servlets se realizan con el método POST.
- Si hubiera un error al consultar o añadir datos a la base de datos, se intenta ser lo más discreto posible en cuanto a mostrar información sobre el origen del error. En caso de que este error ocurra, el usuario es redirigido a una página error.jsp donde se explica que ha habido un error y que pruebe otra vez a realizar su consulta.

Accesibilidad

Por último, destacar que nuestra aplicación tiene varios elementos que se han ido explicando en este documento para hacer la accesibilidad más fácil al usuario, desde una página de inicio en la cual es muy fácil de navegar, hasta utilizar elementos strong en todas las páginas, colores llamativos, muchos encabezados, elementos tipo alt, parámetro lang. Además, como hemos explicado, el hecho de tener una validación por parte del cliente muy reforzada también es una ayuda para personas que tengan dificultades a la hora de registrarse o buscar usuarios. Por último, en las páginas siempre hay un encabezado muy visible en el cual se puede navegar por las diferentes páginas.

