




RED SOCIAL – NODE JS

SDI – CURSO 17/18

Martín Peláez Díaz – UO236974

Gemma González Gil – UO236976



Contenido

Introducción	3
Aplicación Web.....	3
Casos de uso	3
1 Público: registrarse como usuario	3
2 Público: iniciar sesión	3
3 Usuario registrado: listar todos los usuarios de la aplicación	3
4 Usuario registrado: buscar entre todos los usuarios de la aplicación	3
5 Usuario registrado: enviar una invitación de amistad a un usuario	3
6 Usuario registrado: listar las invitaciones de amistad recibidas	4
7 Usuario registrado: aceptar una invitación recibida	4
8 Usuario registrado: listar los usuarios amigos	4
Servicio Web – Implementación de la API de Servicios Web REST	4
Casos de uso	4
S.1 Identificarse con usuario – token.....	4
S.2 Usuario identificado: listar todos los amigos	4
S.3 Usuario identificado: crear un mensaje.....	4
S.4 Obtener mis mensajes de una “conversación”	4
*S.5 Usuario identificado: marcar mensaje como leído	4
Aplicación jQuery	4
Casos de uso	4
C.1 Autenticación del usuario	4
C.2 Mostrar la lista de amigos.....	5
C.3 Mostrar los mensajes	5
C.4 Crear mensaje	5
*C.5 Marcar mensajes como leídos de forma automática.....	5
*C.6 Mostrar el número de mensajes sin leer	5
Pruebas.....	5
1.1 [RegVal] Registro de Usuario con datos válidos.....	5
1.2 [RegInval] Registro de Usuario con datos inválidos (repetición de contraseña inválida).....	5
3.2 [LisUsrInVal] Intento de acceso con URL desde un usuario no identificado al listado de usuarios desde un usuario en sesión	6
4.2 [BusUsrInVal] Intento de acceso con URL a la búsqueda de usuarios desde un usuario no identificado.....	6
C1.1 [CInVal] Inicio de sesión con datos válidos	7
C1.2 [CInInVal] Inicio de sesión con datos inválidos (usuario no existente en la aplicación)	7

C.2.1 [CListAmiVal] Acceder a la lista de amigos de un usuario, que al menos tenga tres amigos	7
C.2.2 [CListAmiFil] Acceder a la lista de amigos de un usuario, y realizar un filtrado para encontrar a un amigo concreto, el nombre a buscar debe coincidir con el de un amigo	7
C3.1 [CListMenVal] Acceder a la lista de mensajes de un amigo “chat”, la lista debe contener al menos tres mensajes	7
C4.1 [CCrearMenVal] Acceder a la lista de mensajes de un amigo “chat” y crear un nuevo mensaje, validar que el mensaje aparece en la lista de mensajes	8
C5.1 [CMenLeidoVal] Identificarse en la aplicación y enviar un mensaje a un amigo, validar que el mensaje enviado aparece en el chat. Identificarse después con el usuario que recibido el mensaje y validar que tiene un mensaje sin leer, entrar en el chat y comprobar que el mensaje pasa a tener el estado leído	8
C6.1 [CListaMenNoLeidoVal] Identificarse en la aplicación y enviar tres mensajes a un amigo, validar que los mensajes enviados aparecen en el chat. Identificarse después con el usuario que recibido el mensaje y validar que el número de mensajes sin leer aparece en la propia lista de amigos	8
Información adicional.....	8

Introducción

A continuación, se explicarán los casos de uso de una la aplicación web, de un servicio web y de un cliente jQuery, así como las pruebas automatizadas que se han realizado con Selenium.

La aplicación consta de un Logger que registra la actividad de los usuarios (*miLogger.log*).

Aplicación Web

Casos de uso

1 Público: registrarse como usuario

Cualquier persona que acceda a la a aplicación podrá registrarse como usuario accediendo a la URL `"/registrarse"` o haciendo click en el botón "Regístrate" de la barra de navegación. Deberá introducir el email, el nombre, la contraseña, repetir la contraseña y darle a el botón "Registrar", de esa forma el usuario ya estará añadido en la base de datos, para posteriormente poder realizar el login.

Para que sean válidos los datos introducidos, han de cumplir las siguientes características:

- El correo electrónico debe contener el carácter '@' y texto detrás de él.
- La contraseña y la repetición de la contraseña deben coincidir.

Si el registro finaliza correctamente, el usuario recién registrado será autenticado automáticamente en la aplicación y se le mostrará la vista en la que aparece el listado de todos los usuarios de la aplicación.

2 Público: iniciar sesión

Todos los usuarios que se han registrado previamente pueden autenticarse en la aplicación mediante un formulario de login (en la URL `/identificarse` o haciendo click en el botón "Identifícate" de la barra de navegación) introduciendo su email y contraseña y dándole al botón de "Aceptar".

En caso de que los datos introducidos sean incorrectos, se notificará al usuario ("Email o contraseña incorrecto"), si los datos son correctos, se redireccionará a la URL `"/listUsers"`.

3 Usuario registrado: listar todos los usuarios de la aplicación

Los usuarios autenticados en la aplicación podrán ver un listado paginado (5 usuarios por página) con todos los usuarios de la aplicación (menos a él mismo). En ese listado aparecerá, para cada usuario, el nombre, el email y un botón para agregarlo como amigo.

4 Usuario registrado: buscar entre todos los usuarios de la aplicación

En listado anterior, habrá un cuadro de búsqueda, para buscar en ese listado por nombre y correo electrónico. La búsqueda se realizará al hacer click en el botón buscar, de forma que la página se recarga y en la lista se muestran solo los usuarios que coinciden con la búsqueda.

5 Usuario registrado: enviar una invitación de amistad a un usuario

Desde el listado de usuarios, el usuario autenticado podrá agregar a otro como amigo haciendo click en el botón "Agregar amigo". De esta forma se creará una petición de amistad entre el usuario autenticado y el seleccionado en la lista. Se notificará al usuario con un mensaje, que ha realizado está acción.

Si la petición ya había sido enviada previamente o si ya son amigos, y se vuelve a hacer click en el botón, se mostrará un mensaje para notificárselo a usuario.

6 Usuario registrado: listar las invitaciones de amistad recibidas

El usuario autenticado tendrá acceso desde la barra de navegación “Lista Peticiones Amistad” a un listado paginado (5 usuarios máximo por página) con las solicitudes de amistad recibidas (URL “`peticionAmistad/listar`”). En él aparecerá el nombre y el email de todos los usuarios que le hayan enviado una solicitud de amistad al usuario autenticado, y al lado de cada uno un botón para aceptar dicha solicitud.

7 Usuario registrado: aceptar una invitación recibida

El en listado mencionado anteriormente, se podrán aceptar las solicitudes de amistad recibidas, de forma que, al aceptar la solicitud de dejará de mostrar automáticamente en esa lista.

8 Usuario registrado: listar los usuarios amigos

Una vez aceptada la solicitud de amistad, los usuarios pasarán a ser amigos. Los usuarios amigos del usuario autenticado, serán mostrados en una lista (URL “`amistad/listar`”) paginada (5 usuarios máximo por página) accesible desde la barra de navegación “Lista Amigos”. Para cada usuario se mostrará el nombre y el correo electrónico.

Servicio Web – Implementación de la API de Servicios Web REST

Casos de uso

S.1 Identificarse con usuario – token

Mediante la URL “`/api/autenticar`”, se hace una petición POST con los valores en el cuerpo “email” y “password” de un usuario previamente registrado, de esta forma se le asignará a dicho usuario un token de autenticación para que pueda acceder a la aplicación.

S.2 Usuario identificado: listar todos los amigos

Haciendo una petición GET a la URL “`/api/amigos`”, se obtiene en formato JSON el listado de todos los amigos del usuario que está en sesión.

S.3 Usuario identificado: crear un mensaje

Haciendo una petición POST a la URL “`/api/mensajes`”, con los valores en el cuerpo “destino” y “texto”, se crea un mensaje.

S.4 Obtener mis mensajes de una “conversación”

Haciendo una petición GET a la URL “`/api/mensajes/:id`” (donde :id es el id de un usuario amigo del autenticado), se obtienen todos los mensajes intercambiados entre ambos usuarios.

**S.5 Usuario identificado: marcar mensaje como leído*

Haciendo una petición PUT a la URL “`/api/mensajes/:id`” (donde :id es el id de un mensaje), se modifica el valor leído a true de dicho mensaje.

Aplicación jQuery

Casos de uso

C.1 Autenticación del usuario

Todos los usuarios que hayan sido registrados previamente pueden autenticarse en la aplicación mediante un formulario de login introduciendo su email y contraseña y dándole al botón de “Aceptar”.

En caso de que los datos introducidos sean incorrectos, se notificará al usuario (“Usuario no encontrado”), si los datos son correctos, se mostrará una lista con los amigos del usuario.

C.2 Mostrar la lista de amigos

Una vez autenticado en la aplicación, el usuario tendrá acceso a una lista donde podrá ver el nombre y el email de todos sus amigos. En dicha lista podrá filtrar por nombre mediante un cuadro de búsqueda.

C.3 Mostrar los mensajes

El usuario autenticado deberá hacer click encima del nombre de cualquiera de sus amigos y se mostrará una lista con todos los mensajes intercambiado entre ambos usuarios.

C.4 Crear mensaje

Para crear un mensaje, el usuario autenticado deberá hacer click encima del nombre del amigo al que le quiere enviar el mensaje. Se mostrará un chat con todos los mensajes intercambiados entre ambos usuarios y habrá un cuadro de texto donde se podrá escribir el mensaje y un botón “Enviar” para crearlo, una vez enviado se mostrará en la lista de mensajes.

**C.5 Marcar mensajes como leídos de forma automática*

Cuando el usuario autenticado haga click en el nombre de alguno de sus amigos para entrar en el chat, se marcarán de forma automática como leídos los mensajes recibidos de dicho amigo.

**C.6 Mostrar el número de mensajes sin leer*

En la lista de amigos, se mostrará al lado del nombre de cada amigo el número de mensajes que el usuario autenticado tenga sin leer de dicho amigo.

Pruebas

1.1 [RegVal] Registro de Usuario con datos válidos

Se hace clic en el botón de la barra de navegación (Regístrate) y se accede al formulario de registro. Se completa con los siguientes campos: “email: *numeroAleatorio@example.com* nombre: Pepe password: 1234 repita el password: 1234” y se hace clic en el botón “Registrar”. Una vez realizado esto, se comprueba que se ha hecho autologin redirigiendo al usuario recién registrado a la vista de la lista de los usuarios del sistema.

1.2 [RegInval] Registro de Usuario con datos inválidos (repetición de contraseña invalida)

Se hace clic en el botón de la barra de navegación (Regístrate) y se accede al formulario de registro. Se completa con los siguientes datos: “email: *example@example.com* nombre: Josefo password: 1234 repita el password: 12345” y se hace clic en el botón “Registrar”. Una vez realizado esto, se comprueba que se muestra el mensaje de error.

2.1 [InVal] Inicio de sesión con datos válidos

Se rellena el formulario de login con los siguientes datos: “email: *gemma@example.com* password: 1234” que son los de un usuario que existe en la base de datos y se comprueba que se accede a la lista de todos los usuarios del sistema.

2.2 [InInVal] Inicio de sesión con datos inválidos (usuario no existente en la aplicación)

Se rellena el formulario de login con los siguientes datos: “email: *noExiste@example.com* password: 1234” que son los de un usuario que no existe en la base de datos y se comprueba que se muestra un mensaje de error.

3.1 [LisUsrVal] Acceso al listado de usuarios desde un usuario en sesión

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos y se comprueba que se accede a la lista de todos los usuarios del sistema.

3.2 [LisUsrInVal] Intento de acceso con URL desde un usuario no identificado al listado de usuarios desde un usuario en sesión

Se accede a la url: <http://localhost:8081/listUsers> y se comprueba que no permite acceder, redirigiendo automáticamente a la página de login.

4.1 [BusUsrVal] Realizar una búsqueda valida en el listado de usuarios desde un usuario en sesión

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la vista de la lista de todos los usuarios, que se muestra automáticamente al acceder, se escribe en el cuadro de búsqueda “gmail”, y se comprueba que encuentra un usuario que tienen en su nombre o email dicha cadena.

4.2 [BusUsrInVal] Intento de acceso con URL a la búsqueda de usuarios desde un usuario no identificado

Se accede a la url: <http://localhost:8090/listUsers?busqueda=gmail> y se comprueba que no permite acceder, redirigiendo automáticamente a la página de login.

5.1 [InvVal] Enviar una invitación de amistad a un usuario de forma valida

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la vista de la lista de todos los usuarios, que se muestra automáticamente al acceder, se hace clic en el botón de “Agregar amigo” y se comprueba que se muestra el mensaje de que se ha enviado la petición.

5.2 [InvInVal] Enviar una invitación de amistad a un usuario al que ya le habíamos invitado la invitación previamente

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la vista de la lista de todos los usuarios, que se muestra automáticamente al acceder, se hace clic en el botón de “Agregar amigo” dos veces, para el usuario con email “prueba@prueba.com”. La segunda vez se comprueba que se muestra el mensaje de error.

6.1 [LisInvVal] Listar las invitaciones recibidas por un usuario, realizar la comprobación con una lista que al menos tenga una invitación recibida

Se rellena el formulario de login con los siguientes datos: “email: nacho@gmail.com contraseña: 123456” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la vista de la lista de todos los usuarios, que se muestra automáticamente al acceder, se hace clic en el botón de “Agregar amigo” para el usuario “martin@gmail.com”. Una vez hecho esto, hacemos logout y nos autenticamos ahora con el usuario al que le hemos enviado la petición (“email: martin@gmail.com password: 123456”). Desde la barra de navegación hacemos clic en “Lista Peticiones Amistad” y comprobamos que aparece el nombre del usuario que nos ha enviado la petición.

7.1 [AcepInvVal] Aceptar una invitación recibida

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la vista de la lista de todos los usuarios, que se muestra automáticamente al acceder, se hace clic en el botón de “Agregar amigo” para los usuarios “nacho@gmail.com”, “martin@gmail.com” y “raul@gmail.com”. Una vez hecho esto, hacemos logout y nos autenticamos ahora con el usuario: “email: nacho@gmail.com password: 123456”, desde la barra de navegación hacemos clic en “List Peticiones Amistad” y hacemos clic en el botón “Aceptar” para aceptar la petición de amistad, y comprobamos que ya no aparece en la lista el email del usuario. Repetimos el proceso para el usuario: “email: martin@gmail.com password: 123456” y para el usuario “email: raul@gmail.com password: 123456”.

8.1 [ListAmiVal] Listar los amigos de un usuario.

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la vista de la lista de todos los usuarios, que se muestra automáticamente al acceder, comprobamos que aparecen en esa lista los usuarios “Raul”, “Nacho” y “Martin”.

C1.1 [CInVal] Inicio de sesión con datos válidos

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos y se comprueba que se accede a la lista de todos los usuarios del sistema.

C1.2 [CInnVal] Inicio de sesión con datos inválidos (usuario no existente en la aplicación)

Se rellena el formulario de login con los siguientes datos: “email: gemm@example.com password: 1234” que son los de un usuario que no existe en la base de datos y se comprueba que se muestra un mensaje de error.

C.2.1 [CListAmiVal] Acceder a la lista de amigos de un usuario, que al menos tenga tres amigos

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos y se comprueba que se accede a la lista de todos los usuarios del sistema.

C.2.2 [CListAmiFil] Acceder a la lista de amigos de un usuario, y realizar un filtrado para encontrar a un amigo concreto, el nombre a buscar debe coincidir con el de un amigo

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, en el cuadro de búsqueda que hay en la lista de amigos, se escribe “Nacho”, y se comprueba que solo aparece el usuario cuyo nombre coincide con el buscado.

C3.1 [CListMenVal] Acceder a la lista de mensajes de un amigo “chat”, la lista debe contener al menos tres mensajes

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la lista de amigos, se hace click sobre el nombre “Nacho”, para acceder al chat, y se comprueba que hay tres mensajes.

C4.1 [CCrearMenVal] Acceder a la lista de mensajes de un amigo “chat” y crear un nuevo mensaje, validar que el mensaje aparece en la lista de mensajes

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la lista de amigos, se hace click sobre el nombre “Nacho”, para acceder al chat. Desde el cuadro de texto se escribe un nuevo mensaje y se hace click en el botón “Enviar”, una vez enviado se comprueba que aparece en la lista el mensaje que se acaba de enviar. Se crean de la misma forma otros dos mensajes más.

C5.1 [CMenLeidoVal] Identificarse en la aplicación y enviar un mensaje a un amigo, validar que el mensaje enviado aparece en el chat. Identificarse después con el usuario que recibido el mensaje y validar que tiene un mensaje sin leer, entrar en el chat y comprobar que el mensaje pasa a tener el estado leído

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la lista de amigos, se hace click sobre el nombre “Raul”, para acceder al chat. Desde el cuadro de texto se escribe un nuevo mensaje y se hace click en el botón “Enviar”, una vez enviado se comprueba que aparece en la lista el mensaje que se acaba de enviar. A continuación, se vuelve al formulario de login y se rellena con los siguientes datos: “email: raul@gmail.com password:123456” que pertenecen al usuario al que se le acaba de enviar un mensaje. Una vez autenticado se comprueba que se muestra que tiene un mensaje sin leer, después se hace click sobre el nombre “Gemma” para ver el mensaje recibido y se comprueba que leído es “true”.

C6.1 [CListaMenNoLeidoVal] Identificarse en la aplicación y enviar tres mensajes a un amigo, validar que los mensajes enviados aparecen en el chat. Identificarse después con el usuario que recibido el mensaje y validar que el número de mensajes sin leer aparece en la propia lista de amigos

Se rellena el formulario de login con los siguientes datos: “email: gemma@example.com password: 1234” que son los de un usuario que existe en la base de datos. Una vez autenticado correctamente, desde la lista de amigos, se hace click sobre el nombre “Martin”, para acceder al chat. Desde el cuadro de texto se escribe un nuevo mensaje y se hace click en el botón “Enviar”, una vez enviado se comprueba que aparece en la lista el mensaje que se acaba de enviar. Se hace lo mismo con dos mensajes más. A continuación, se vuelve al formulario de login y se rellena con los siguientes datos: “email: martin@gmail.com password:123456” que pertenecen al usuario al que se le acaban de enviar tres mensajes. Una vez autenticado se comprueba que se muestra que tiene tres mensajes sin leer.

Información adicional

- Para ejecutar los test hay que cambiar la ruta de Firefox.
- Al inicio de los test se borra toda la base de datos.