



SDI – Sistemas Distribuidos e Internet

ENUNCIADO PRÁCTICA 2 – NODE.Js & Servicios Web Extraordinaria adelantada

INFORME

Nombre:	Samuel
Apellidos:	Moreno Vincent
Email:	UO266321@uniovi.es
Cód. ID GIT	613
Repositorio Github	https://github.com/samuelmorenov/sdi2021-entrega2-613-ext-adel



Índice

INTRODUCCIÓN.....	3
MAPA DE NAVEGACIÓN	3
1. APLICACIÓN WEB	3
2. SERVICIOS WEB	4
ASPECTOS TÉCNICOS Y DE DISEÑO RELEVANTES	5
INFORMACIÓN NECESARIA PARA EL DESPLIEGUE Y EJECUCIÓN.....	5
CAMBIOS INCLUIDOS	5
CONCLUSIÓN.....	5



Introducción

Para la realización de este proyecto se ha utilizado como base el proyecto de NodeJs realizado en el desarrollo de las clases prácticas de la asignatura, siguiendo los guiones de las sesiones de la 7 a la 10. Así pues, se ha mantenido toda la apariencia visual que tenía dichas prácticas. Además, y como se detalla en los cambios incluidos, este proyecto ha sido realizado sin una entrega anterior. En algunas partes del proyecto se han tomado partes del código de mi proyecto de Spring Boot, como las vistas, para que fuesen lo mas similares posibles y agilizar las pruebas. Las pruebas fueron importadas desde el proyecto de Spring Boot y modificadas para adaptarse a las distintas funcionalidades que ofrece este proyecto.

Mapa de navegación

1. Aplicación Web

En primer lugar, cuando accedemos al enlace <https://localhost:8081/> tenemos en la parte superior la barra de navegación, donde tenemos las opciones de identificarse y registrarse.



Si pinchamos en la opción de *Registrate* se nos abrirá un formulario que nos pide nuestros datos. Una vez rellenados, y si todo esta correcto, nos redireccionará a la pagina de login.

En caso de pinchar en la opción *Identificate* el formulario será mas pequeño y nos pedirá unicamente el Email y la contraseña.

Una vez estamos logueados la barra de navegación cambiará dandonos nuevas opciones.



El boton de *Inicio* nos lleva a la pagina de bienvenida de la aplicación.

El boton *Usuarios* nos abre un submenu con la opción de ver la lista de usuarios que hay en el sistema.

En la pantalla de la lista de usuarios se muestran todos los usuarios con su email, nombre y apellidos y tienes la opción de enviar una petición de amistad. Además puedes usar el buscador en la parte superior para filtra los usuarios del sistema.


El boton *Amigos* es un submenu donde tenemos la opción de ver la lista de invitaciones de amistad, donde podremos ver las invitaciones que nos han hecho y aceptarlas. Y la lista de amigos, donde podremos ver los amigos que tenemos.

Por ultimo los botones de *Registrate* e *Identificate* ya no estan y esta un boton para *Desconectarse*.



2. Servicios Web

En primer lugar, cuando accedemos al enlace <https://localhost:8081/cliente.html> nos carga una vista de login, donde podremos ingresar los datos de usuario para loguearse.

 Amigos


Identificación de usuario

Email:

Password:




Aceptar

Una vez estamos logueados la vista cambiará mostrandonos los amigos que tenemos.

 Amigos

Lista de amigos


Buscar

Nombre	Apellidos	Email	Mensajes
Pedro	Díaz	pedro@email.com	
Lucas	Núñez	lucas@email.com	
Marta	Almonte	marta@email.com	

En esta vista podemos ver la lista de amigos, con su email, nombre y apellidos, que podremos filtrar usando el buscador de la parte superior.

Si hacemos clic en el icono de los mensajes de un amigo nos cargará la vista de mensajes, donde podremos ver los mensajes que nos hemos enviado y enviar mensajes nuevos.

Para regresar a la lista de amigos solo tenemos que pinchar en el panel de navegador donde pide Amigos.

 Amigos

Chat con Pedro Díaz

Hola Pedro, ¿qué tal?

Hola Pelayo, yo muy bien, y tu ¿qué tal?

Bien también, a ver si nos vemos

Pues sí, un día de estos

¿Estás libre mañana?

Enviar



Aspectos técnicos y de diseño relevantes

Toda la arquitectura y diseño de la aplicación está basada en la aplicación realizada durante las practicas. En el único sitio donde hay cambios significativos es en el diseño de las pruebas. En lugar de estar todos apilados en una única clase se han separado en 14 clases distintas correspondientes a cada uno de los ejercicios. Todos ellos se ejecutan de forma consecutiva desde la clase *"MySocialNetwork_Tests.java"*. Esto conlleva el inconveniente de no poder ejecutar un test de forma individual a priori, ya que no ejecutaría el método *"BeforeClass"*. Para solucionar este problema, y además hacer la programación de los test más fluida se ha implementado un patrón Singleton para el driver *"WebDriver"* de selenium, de manera que ahora todos los test llaman a la misma instancia del driver y se inicializa una única vez. Este patrón se puede ver en la clase *DriverSingleton.java*

Información necesaria para el despliegue y ejecución

Antes de nada, para que la aplicación arranque sin problemas debemos tener un fichero en el directorio *raid* llamado *pass.txt* con la contraseña del administrador de la base de datos de mongo. Esto es así porque me pareció una vulnerabilidad enorme tener la contraseña hardcodeada, y este archivo no se incluye en repositorio de GitHub.

Para arrancar la aplicación únicamente tendremos que ejecutar el archivo *run.bat*. Si es la primera vez que lo ejecutamos, necesitaremos hacer antes en *npm install* en la consola.

Para ejecutar las pruebas debe estar desplegada la aplicación. Se ejecutará la clase *MySocialNetwork_Tests.java* como *JUnit Test*

Cambios incluidos

Para la realización del proyecto se ha optado por empezar a partir del proyecto resultado de las sesiones de laboratorio como se indica en la introducción. Se ha utilizado también el proyecto de Spring Boot a modo de guía para agilizar la construcción del proyecto, utilizando parte de las vistas html para que fuesen lo más parecidas posibles.

No existe una entrega previa, ya que en el curso pasado no llegamos a empezar este proyecto.

En este proyecto cometí el error de usar el mismo repositorio que el que había usado para realizar las practicas, por lo que los primeros commits corresponden a estas prácticas. Además, al final resultó poco útil ya que la mayoría de las cosas que habían de esa practica no eran útiles para la red social.

Conclusión

Aunque no he podido realizar las partes opcionales del proyecto por falta de tiempo, he aprendido mucho acerca de Node Js realizando estas practicas y me parece un complemento estupendo junto a Spring Boot. Además, la parte de servicios Web es realmente útil, habría querido hacer una parte así también en Spring Boot ya que es lo que estoy haciendo en las practicas de empresa.