

UNIVERSIDAD DE EXTREMADURA



PI- Programación en Internet

ENTREGA 1

Web App using HTML, CSS, Servlets & JSP

RedNotes

AUTOR:

Francisco Javier Rojo Martín

(DNI:76042262-F)

ÍNDICE

1- Introducción	2
2- Requisitos obligatorios	3
3- Extras	5
4- Anexo I: Contraseñas de usuarios	8

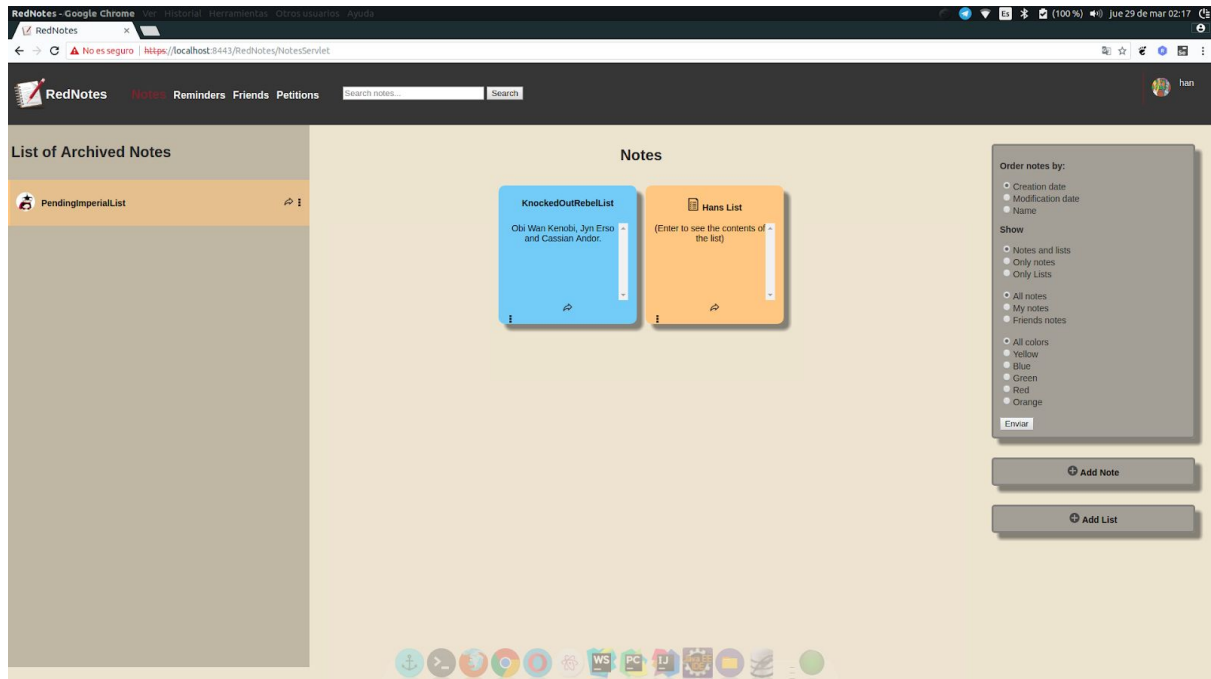
1- Introducción

El proyecto presentado consiste en una aplicación web de notas online (similar a la aplicación *Google Keep*), a la que se le ha denominado **RedNotes**.

Dicha aplicación, ofrece la posibilidad de tener una gestión de notas y listas, incluyendo funciones como recordatorios, gestión de amigos,... Todo ello, con el requisito de haberse registrado previamente en la aplicación.

Además, el usuario podrá proporcionar información como su país, nombre o ciudad, que ayudará a los usuarios a buscar nuevos amigos. También podrá poner su número de teléfono, de tal forma que sus amigos se puedan comunicar con él fuera de la aplicación. Incluso podrá personalizar su icono con una serie de imágenes predefinidas en la aplicación.

Junto a la aplicación, se entrega una base de datos: *RedNotes.sqlite*. Esta base de datos está adaptada para la aplicación Web en cuestión.



2- Requisitos obligatorios

En la lista siguiente, encontramos como se han conseguido cada uno de los requisitos de la aplicación, marcados en la entrega (la enumeración coincide con la marcada en la entrega).

- **Requisito 1: HTML & CSS are valid.**

Este requisito se cumple mediante el uso adecuado de las etiquetas HTML y CSS. Existen Warnings relacionados con las variables de color, en el CSS.

- **Requisito 2: Security.**

Se implementa la seguridad marcada mediante:

- Control, en cada GET, de que el usuario esta logueado.
- Control, cada vez que se coge una nota concreta, bien para mostrarla, bien para editarla, de que la nota que se obtiene es distinta de NULL (existe en la base de datos para el usuario).
- En el archivo “web.xml”, se ha añadido una parte para el uso de HTTPS, mediante el uso de TomCat configurado para SSL (siguiendo el guión de la práctica 6).

- **Requisito 3: Good practices.**

Se han seguido las buenas prácticas estudiadas en la asignatura en todos los Servlets. Por ejemplo, tras un POST, se hace un redirect al GET adecuado, en lugar de mostrar directamente el JSP desde el POST (siempre que se modifican los datos de la base de datos).

- **Requisito 4: Any person can register as user of the web app.**

Se ha añadido, en la esquina superior derecha, un menú para las opciones del usuario logueado. Desde ahí, se podrá consultar la información de la cuenta y modificarla, así como borrar la cuenta.

- **Requisito 5: Any user can create, see and edit notes.**

Desde la página principal o página de Notas, se incorpora un botón para añadir nuevas notas. También se puede abrir cada nota, así como editarla y eliminarla desde dentro de la página de la propia nota.

- **Requisito 6: Any user can share any of her note.**

En la página de la Nota, así como desde el menú de Notas, se pueden compartir las notas con los amigos. Para cada nota, se muestra la lista de amigos del usuario, excepto aquellos con los que ya está compartida la nota.

- **Requisito 7: Any user can edit notes that were shared with them.**

Para todas las notas, sean propias o no, se pueden editar tal y como se explicó en el requisito 5.

- **Requisito 8: Any user can “delete” her notes or notes that were shared with her.**

Para eliminar las notas, se comprueba quien es el dueño de la nota. Si la nota es del dueño, además de eliminarse la entrada en UsersNotes para el usuario, se eliminan todas las entradas de esta nota en UsersNotes (para todos los usuarios), así como se elimina definitivamente la nota.

- **Requisito 9: Pinned notes.**

Se ha implementado un Servlet, con un POST, que nos permite fijar una nota desde la página de la nota. Se encuentran dos botones: pinned y unpinned. Cuando una nota está fijada, se muestra en primer lugar, junto con las notas fijadas, en la página de Notas.

- **Requisito 10: Archived notes.**

Se ha implementado un Servlet, con un POST, que nos permite archivar una nota desde la página de la nota, o desde el propio menú de notas. Se encuentran dos botones: archived y unarchived, en la página Nota. Desde el menú de Notas, se puede archivar una nota gracias al menú desplegable de cada nota. Cuando una nota está archivada, se muestra en la lista lateral izquierda de la página de notas.

- **Requisito 11: Search.**

Se pueden buscar notas, siempre que ponga “Search note...” en la barra de búsqueda superior (pues, en otros casos, puede aparecer “Search friend...”, al ser una página relacionada con la ampliación de amigos). La búsqueda se realiza, tanto por título como por contenido. Si el texto buscado se encuentra en el título o en el contenido de una nota, se muestra esta nota, en la lista de notas buscadas.

3- Extras

En la lista siguiente, se pueden ver la lista de extras que se han incluido a la aplicación web. Estos extras, no se encuentran completamente en la lista de la entrega. Igualmente, hay otros extras de la lista que se han implementado de una manera ligeramente distinta de la que se decía en la entrega

- **Extra 1: Filtrado y ordenación de notas.**

Se añade un menú en la página principal de Notas, a la derecha. Desde este menú, se puede decidir que tipo de notas mostrar (listas o notas), así como que colores mostrar, y en qué orden.

Cuando iniciamos sesión, se ponen unas opciones de filtrada y ordenación por defecto. Sin embargo, mientras tengamos iniciada la sesión, los cambios que se produzcan en estas opciones se guardarán durante toda la sesión (mediante el uso de atributos en la sesión), hasta que las cambiemos o cerremos la sesión.

- **Extra 2: Listas.**

Se ha añadido la posibilidad de crear listas, además de notas, en la aplicación. Dichas listas, se crean desde el menú de Notas, como las notas normales. Se diferencian de las notas normales, en el hecho de que el contenido es una lista de elementos, en lugar de un texto plano. Sobre una lista creada, podremos modificar su estado, añadiendo (uno a uno) y eliminando elementos (tantos como se quieran de golpe), así como cambiar el estado de cada elemento (mediante un checkbox asociado a dicho elemento). Aparte de esto, las listas permiten una gestión normal de cualquier nota: modificar título, color, compartir,...

- **Extra 3: Imágenes de los usuarios.**

Se ha añadido información a los usuarios, de tal forma que cada usuario tenga una imagen de perfil. Por defecto, se asigna una determinada. El usuario puede modificar su imagen de sesión, cogiendo una de las que se le ofrece desde un menú de imágenes. Este menú, es accesible desde el Perfil del usuario (no desde el Menú de Edición del Perfil). Todas las modificaciones realizadas, se guardan en la base de datos, para posteriores veces que el usuario inicie sesión.

Estas imágenes, resultan muy útiles para mostrar junto con el nombre de usuario en las notas (para mostrar el usuario dueño de la nota), así como para mostrar en las distintas versiones de una nota (dejándonos ver quien ha hecho cada edición en una nota).

Dichas imágenes, se guardan en una tabla diccionario, en forma de URLs.

- **Extra 4: Recordatorios.**

Esta ampliación, nos permite que, un usuario, pueda asociar múltiples recordatorios a una nota (podrá asociar varios recordatorios a una misma nota, siempre y cuando la fecha del recordatorio no coincida en varias).

En el recordatorio, se añade la nota, la fecha y hora de recordatorio, y una breve descripción (para adjuntar información relevante). También, podemos eliminar los recordatorios que queramos, antes de que llegue su fecha.

Una vez la fecha del recordatorio es más vieja que la fecha actual, el recordatorio desaparece del menú de recordatorios para siempre.

Cada usuario tendrá sus propios recordatorios sobre sus notas (mostrándoles estás todas cuando vayan a crear el recordatorio, siendo mucho más fácil crearlo).

- **Extra 5: Gestión de amigos.**

La gestión de amigos, se complementa con el siguiente Extra. Nos permite tener amigos en la aplicación. Estos amigos, serán con los que podamos compartir las notas (mediante un menú donde aparecen los amigos a los que aún no hemos compartido la nota). El usuario, tendrá un menú para ver sus amigos, así como puede buscarlos con la barra de búsqueda u ordenarlos con las opciones del lateral izquierdo.

En la página de amigos, se muestra una barra, a la izquierda, donde vemos todos los amigos o los coincidentes con la búsqueda realizada. En la parte derecha, la información del amigo, así como un apartado para acceder a las notas que nos ha compartido ese amigo. También hay un botón para añadir amigos.

A la hora de añadir amigos, nos encontraremos con una página similar a la anterior, donde tenemos una lista de todos los usuarios de la base de datos que aún no son amigos, así como una barra de búsqueda avanzada (se explica en el siguiente Extra). Cuando pulsamos sobre un usuario, nos muestra una pequeña cantidad de información y nos da la opción de añadirlo. Si aceptamos, se le envía una petición de amistad, que el otro usuario podrá aceptar o rechazar. Igualmente, si nos arrepentimos de haberla mandado, podemos borrar la petición nosotros mismos. Todo ello desde la página Peticiones de nuestro sitio web.

Se recomienda encarecidamente probar esta parte de la aplicación, pues ha sido muy laboriosa, y resulta complejo resumirla aquí, sin una muestra.

- **Extra 6: Búsqueda avanzada de nuevos amigos.**

Se añade un menú en la página de búsqueda de amigos, que nos permite filtrar la búsqueda de nuevos amigos. Podremos buscar amigos con la combinación

de información de los campos: Name, Username, Country y City. Resulta interesante decir que se forma el WHERE de la consulta SQL para obtener los resultados mediante funciones del tipo String.

- **Extra 7: Colores.**

Se añade la opción de cambiar el color de todos los tipos de notas (notas y listas), permitiéndonos poner estas de 5 colores distintos. Además, a la hora de mostrar los colores de las notas, se comprueba si la nota está archivada o no, cambiando ligeramente el tono del color según la situación. Además, este color se guarda en la base de datos para cada usuario. Es decir, dos usuarios distintos, pueden tener, para una misma nota, dos colores distintos. Por defecto, el color inicial de una nota es el Amarillo.

- **Extra 8: Control de versiones.**

Se añade un control de versiones de las notas. Se permite, por cada nota:

- Ver versiones viejas de la nota.
- Establecer una de las versiones viejas como la actual (y eliminar las posteriores)
- Eliminar una versión determinada de la nota, mientras no sea la única que queda (en cuyo caso, se deberá eliminar directamente la nota). Si la versión a eliminar es la actual, el efecto será el mismo a volver a la versión anterior de la nota.

Para ayudar a la navegación entre versiones, se han añadido aspectos como colorear de un color más fuerte, la versión en la que nos encontramos dentro de una nota.

Al crear la nota o lista, se crea la versión inicial de la nota.

Cuando modificamos el título o el contenido de la nota o lista (en esta última, añadiendo, eliminando o modificando el estado de los elementos), se modifica la fecha de modificación de la nota, y se añade una versión nueva a la lista de versiones de la nota.

Se podrá ver quien fue el usuario que editó esa versión de la nota (usuario que provocó que se crease esa nueva versión), mediante su imagen de perfil y su username (cómo “*label*” de la imagen)

- **Extra 9: Inicio de sesión con usuario y email.**

Se permite iniciar sesión mediante el email o username. Así, el usuario podrá optar por cualquiera de estos dos campos, acompañados de la contraseña, para iniciar sesión en la aplicación web. Para esto, nos ayudamos del hecho de que los *usernames* no permiten el carácter “@”, mientras que el email obliga a tenerlo.

- **Extra 10: Encriptado de contraseñas en Base de Datos.**

Se añade ha añadido un algoritmo de encriptado al proyecto, permitiéndonos encriptar las contraseñas de los usuarios al almacenarlas en la Base de Datos (tanto al crear la cuenta como al editar la contraseña). Se ha considerado una buena ampliación, al no controlarse los SQL Injection. Gracias a esto, aunque se hiciese un “SELECT” de la tabla User, no se obtendrían la contraseñas de forma sencilla.

Téngase en cuenta que, para estas ampliaciones, se han creado varias tablas dependientes (de Note o UsersNote) en la base de datos. En todo momento, se gestiona que las tuplas dependientes se eliminan de estas nuevas tablas, al eliminar las notas o cuentas de usuarios a las que están asociadas. Podríamos dejar este trabajo a la base de datos, mediante el “ON DELETE CASCADE”, pero se ha optado por controlarlo nosotros en la aplicación, dándonos mayor seguridad de que todo será consistente.

4- Anexo I: Contraseñas de usuarios

Se añade este anexo a la documentación, con el fin de poder saber la contraseña de cada usuario, pues, al estar encriptadas en la Base de Datos, no se pueden saber al abrir la tabla simplemente.

idu	username	password	email
0	darth	vadER99	darth.vader@darksideoftheforce.org
1	emperor	palpatiNE99	emperor.palpatine@darksideoftheforce.org
2	leia	orgaNA99	leia.organa@lightsideoftheforce.org
3	luke	skywalkER99	luke.skywalker@lightsideoftheforce.org
4	han	soloEL54	han.solo@lightsideoftheforce.org
9	lopez	pruebaAA11	esto@es.unaprueba
10	userPrueba	pruebaAA12	prueba@prueba.com
11	pepe	holaM9	pepe@pep.e
12	javi97	Prueba11	javier@rojo.com
14	javiER	rojoBB99	javier.rojo@unex.es

Se añade esta imagen entre los archivos entregados también.