



ENUNCIADO PRÁCTICA 1– SPRING

MY WALLAPOP

1	INSTRUCCIONES.....	2
2	ENUNCIADO	2
3	REQUISITOS OBLIGATORIOS	3
3.1	PÚBLICO: REGISTRARSE COMO USUARIO DEL SISTEMA	3
3.2	USUARIO REGISTRADO: INICIAR SESIÓN	4
3.3	USUARIO REGISTRADO: CERRAR SESIÓN	4
3.4	ADMINISTRADOR: LISTADO DE USUARIOS	4
3.5	ADMINISTRADOR: BORRADO MÚLTIPLE DE USUARIOS	5
3.6	USUARIO REGISTRADO: DAR DE ALTA UNA NUEVA OFERTA	5
3.7	USUARIO REGISTRADO: LISTADO DE OFERTAS PROPIAS	6
3.8	USUARIO REGISTRADO: DAR DE BAJA UNA OFERTA	6
3.9	USUARIO REGISTRADO: BUSCAR OFERTAS	6
3.10	USUARIO REGISTRADO: COMPRAR UNA OFERTA	7
3.11	USUARIO REGISTRADO: VER EL LISTADO DE OFERTAS COMPRADAS	7
3.12	INTERNACIONALIZACIÓN DE TODAS LAS VISTAS	7
3.13	SEGURIDAD	8
4	INFORME DE ENTREGA Y PLANTILLA DE CASOS DE PRUEBA	8
4.1	INFORME DE ENTREGA	8
4.2	PLANTILLA DE CASOS DE PRUEBA	9
5	CORRECCIÓN POR PARES.....	9
6	REQUISITOS OPCIONALES	9
6.1	USUARIO REGISTRADO: ENVIAR MENSAJES A UNA OFERTA	9
6.2	USUARIO REGISTRADO: VER EL LISTADO DE CONVERSACIONES	10
6.3	USUARIO REGISTRADO: ELIMINAR UNA CONVERSACIÓN	10
6.4	USUARIO REGISTRADO: MARCAR UNA OFERTA COMO DESTACADA	10
7	PROYECTO A ENTREGAR, PRUEBAS AUTOMATIZADAS Y EL PROTOCOLO DE PRUEBA.....	11
8	ASPECTOS GENERALES.....	11
1.1	SEGURIDAD	11
8.1	ARQUITECTURA	11
8.2	ENTREGA	12
8.3	FECHA MÁXIMA DE ENTREGA	13
9	EVALUACIÓN	13



1 Instrucciones

Los requisitos del 1 - 15 son de carácter obligatorio, y para que el trabajo pueda ser evaluado esos 15 requisitos deben estar implementados, con sus correspondientes pruebas automatizadas (***no se evaluarán los casos de uso que no contengan todas sus pruebas***). La puntuación máxima a la que se opta por implementar estos 15 requisitos de forma perfecta es de 7 sobre 11.

Los casos de uso 16 – 19 son de carácter opcional. La puntuación máxima a la que se opta por implementar los N casos de forma perfecta es de 11 sobre 11.

Notas importantes:

- **Trabajo en equipo:** Esta práctica se podrá realizar en ***equipos de hasta 2 alumnos*** (pertenecientes al mismo grupo de prácticas) y su entrega es obligatoria.

Para los alumnos que trabajen en equipo, se tendrá en cuenta el porcentaje de participación de cada miembro y su ***grado de implicación*** durante el desarrollo del proyecto. Esto se hará mediante dos vías:

- En el informe de entrega se deberá indicar el porcentaje que hay realizado cada uno, así como un apartado explicativo de las tareas realizadas (*Razonamiento de los porcentajes de participación*).
 - El profesor de prácticas realizará un seguimiento de la actividad de cada alumno en el repositorio GitHub de la entrega para contrastarlo con los porcentajes que indiquen los alumnos en el informe.
- **Corrección por pares:** Además, una vez finalizada la ***entrega el 21 de marzo***, se pondrá a disposición de los alumnos a través del campus virtual, una corrección por pares que será INDIVIDUAL y OBLIGATORIA. Tal y como indica el apartado 15.
 - **Prueba de autoría:** aquellos alumnos que sean requeridos deberán realizar una defensa presencial del trabajo en el laboratorio de prácticas, consistente en la implementación de nuevos casos de uso. La nota de la práctica se verá condicionada a la correcta implementación de esos casos de uso. El listado de alumnos y fecha de dicha prueba se publicará en el CV con suficiente antelación.

2 Enunciado

Se trata de desarrollar una aplicación Web de compra-venta entre particulares (al estilo Wallapop) en la que existirán perfiles de usuario de tipo: Público (Anónimo), Usuario Registrado (Administrador y Usuario Estándar).

Los requisitos de cada uno de los perfiles serán:

REQUISITOS OBLIGATORIOS (Hasta 7.0 puntos)

Público. Las opciones públicas serán:

1. Registrarse como usuario del sistema. (0.25 puntos)

Usuario Registrado: Un usuario Administrador o Estándar podrá:

2. Iniciar sesión. (0.25 puntos)
3. Cerrar sesión. (0.25 puntos)

Usuario Administrador. El usuario Administrador tendrá la capacidad de:



4. Ver el Listado de los usuarios del sistema. (0.25 puntos)
5. Borrar múltiples usuarios del sistema. (0.5 puntos)

Usuario Estándar. Un usuario Estándar podrá realizar las siguientes acciones:

6. Dar de alta una nueva oferta. (0.5 puntos)
7. Ver el Listado de ofertas propias. (0.5 puntos)
8. Dar de baja una oferta propia. (0.5 puntos)
9. Buscar ofertas por diferentes campos de la misma. (0.5 puntos)
10. Comprar una oferta. (0.5 puntos)
11. Ver el Listado de ofertas compradas. (0.5 puntos)

Internacionalización (Hasta 1.0 puntos)

12. Internacionalización de páginas. (1.0 punto)

Seguridad. (Hasta 1.0 puntos)

13. Vulnerabilidad de recursos y acciones (1.0 puntos)

Informe obligatorio. (Hasta 0.50 puntos)

14. Escribir un informe obligatorio del proyecto. (0.50 puntos)

Revisión por pares.

15. Realizar revisión por pares

REQUISITOS OPCIONALES (Hasta 4.0 puntos)

16. Enviar un mensaje a una oferta. (1.0 punto)
17. Ver el listado de conversaciones (1.0 punto)
18. Eliminar una conversación (1.0 punto)
19. Marcar una oferta como destacada (1.0 punto)

3 Requisitos OBLIGATORIOS

3.1 Público: Registrarse como usuario del sistema

Los usuarios deben poder registrarse en la aplicación introduciendo: email, nombre, apellidos y una contraseña (que deberá repetirse dos veces y coincidir entre sí). Además, por defecto, un usuario tendrá una cuenta de dinero que se iniciará con 100 Euros (este dato se gestionará internamente sin necesidad de un campo de formulario).

Consideraciones importantes:

- El contador de dinero de cada usuario deberá mostrarse, en todas las vistas de acceso privado para el usuario, al lado de su email.
- El email del usuario no podrá estar repetido en el sistema, se debe informar al usuario de los errores en el proceso de registro.
- Es obligatorio realizar las validaciones del lado del servidor.
- Una vez registrado un usuario será autenticado automáticamente redirigiéndole a la vista de opciones de usuario registrado.



Pruebas Funcionales

[Prueba1] Registro de Usuario con datos válidos.

[Prueba2] Registro de Usuario con datos inválidos (email vacío, nombre vacío, apellidos vacíos).

[Prueba3] Registro de Usuario con datos inválidos (repetición de contraseña inválida).

[Prueba4] Registro de Usuario con datos inválidos (email existente).

3.2 Usuario Registrado: Iniciar sesión

Suministrando su email y contraseña, un usuario registrado podrá autenticarse en el sistema. Sólo los usuarios que proporcionen correctamente su email y su contraseña podrán iniciar sesión con éxito.

En caso de que el inicio de sesión fracase, será necesario mostrar un mensaje de error indicando el problema.

Caso 1: Usuario con perfil de administrador

- Sólo existirá un usuario administrador en el sistema con email “admin@email.com” y contraseña “admin”.
- En caso de que el inicio de sesión sea correcto, se debe redirigir al usuario a la vista que contenga las opciones del *perfil Administrador*.

Caso 2: Usuario Estándar

- En caso de que el inicio de sesión sea correcto, se debe redirigir al usuario a la vista que contenga las opciones del *perfil Usuario*.

Pruebas Funcionales

[Prueba5] Inicio de sesión con datos válidos (administrador).

[Prueba6] Inicio de sesión con datos válidos (usuario estándar).

[Prueba7] Inicio de sesión con datos inválidos (usuario estándar, campo email y contraseña vacíos).

[Prueba8] Inicio de sesión con datos válidos (usuario estándar, email existente, pero contraseña incorrecta).

[Prueba9] Inicio de sesión con datos inválidos (usuario estándar, email no existente en la aplicación).

3.3 Usuario Registrado: Cerrar sesión

Añadir una opción de menú que permita finalizar la sesión, enviando al usuario al formulario de Inicio de sesión. Este botón solo se mostrará si un usuario se ha autenticado previamente.

Pruebas Funcionales

[Prueba10] Hacer click en la opción de salir de sesión y comprobar que se redirige a la página de inicio de sesión (Login).

[Prueba11] Comprobar que el botón cerrar sesión no está visible si el usuario no está autenticado.

3.4 Administrador: Listado de usuarios

Un usuario identificado con perfil de administrador debe poder acceder a una lista en la que figuren todos los usuarios de la aplicación. Para cada usuario se mostrará su email, nombre y apellidos.

Consideraciones importantes:



- No es necesario incluir sistema de paginación en este listado.

Pruebas Funcionales

[Prueba12] Mostrar el listado de usuarios y comprobar que se muestran todos los que existen en el sistema.

3.5 Administrador: Borrado múltiple de usuarios

En la lista anterior donde figuran todos los usuarios de la aplicación, se debe poder seleccionar múltiples usuarios (haciendo uso de checkboxes). Mediante un botón **“Eliminar”** se confirmará el borrado de todos aquellos seleccionados. Al pulsar el botón de Eliminar se deben eliminar todos los usuarios, así como toda la información relativa a los mismos (datos, ofertas, conversaciones, etc.).

Consideraciones importantes:

- En este listado no debería aparecer el usuario administrador o en su defecto no debería poder borrarse.
- No es necesario incluir sistema de paginación en este listado.
- Si se incluye paginación hay que tenerlo en cuenta en las pruebas funcionales.

Pruebas Funcionales

[Prueba13] Ir a la lista de usuarios, borrar el primer usuario de la lista, comprobar que la lista se actualiza y que el usuario desaparece.

[Prueba14] Ir a la lista de usuarios, borrar el último usuario de la lista, comprobar que la lista se actualiza y que el usuario desaparece.

[Prueba15] Ir a la lista de usuarios, borrar 3 usuarios, comprobar que la lista se actualiza y que los usuarios desaparecen.

3.6 Usuario registrado: Dar de alta una nueva oferta

Un usuario identificado con perfil de Usuario Estándar, debe poder crear una oferta suministrando: título descriptivo de la oferta, detalle textual de la oferta, fecha de alta de la oferta (Esta fecha puede ser la del sistema) y cantidad solicitada en euros. Los tamaños y tipos de estos campos quedan a criterio del alumno.

Consideraciones importantes:

- Es obligatorio realizar las validaciones del lado del servidor.

Pruebas Funcionales

[Prueba16] Ir al formulario de alta de oferta, rellenarla con datos válidos y pulsar el botón Submit. Comprobar que la oferta sale en el listado de ofertas de dicho usuario.

[Prueba17] Ir al formulario de alta de oferta, rellenarla con datos inválidos (campo título vacío) y pulsar el botón Submit. Comprobar que se muestra el mensaje de campo obligatorio.



3.7 Usuario registrado: listado de ofertas propias

Un usuario identificado con perfil de Usuario Estándar debe poder acceder a una lista en la que figuren todas sus ofertas. Para cada oferta se mostrará: texto descriptivo de la oferta, detalle de la oferta y cantidad solicitada en euros.

Consideraciones importantes:

- No es necesario incluir sistema de paginación en este listado.

Pruebas Funcionales

[Prueba18] Mostrar el listado de ofertas para dicho usuario y comprobar que se muestran todas las que existen para este usuario.

3.8 Usuario registrado: Dar de baja una oferta

Un usuario identificado con perfil de Usuario Estándar debe poder dar de baja una oferta del listado de sus ofertas seleccionando un botón/enlace Eliminar para la oferta que desee suprimir. Al pulsar el botón Eliminar se deben eliminar tanto la oferta como toda la información relativa a la misma.

Consideraciones importantes:

- Un usuario no podrá dar de baja a una oferta de otro usuario.
- No es necesario incluir sistema de paginación en este listado.

Pruebas Funcionales

[Prueba19] Ir a la lista de ofertas, borrar la primera oferta de la lista, comprobar que la lista se actualiza y que la oferta desaparece.

[Prueba20] Ir a la lista de ofertas, borrar la última oferta de la lista, comprobar que la lista se actualiza y que la oferta desaparece.

3.9 Usuario registrado: Buscar ofertas

Incluir un sistema que permita realizar una búsqueda de ofertas por su título. El cuadro de búsqueda contendrá un único campo de texto. Por ejemplo, si escribimos la cadena “coch” deberá devolver ofertas en las que la cadena “coch” sea parte de su título. Si la cadena está vacía deberá mostrar un listado completo con todas las ofertas existentes en el sistema.

Para cada oferta se mostrará: *texto descriptivo de la oferta, detalle de la oferta y cantidad solicitada (en euros)*. A la derecha de cada oferta, un enlace o botón “**Comprar**” si la oferta está disponible para la compra o bien el texto “**Vendido**” si la oferta ya ha sido vendida.

El resultado de la búsqueda debe ser una lista que muestre las coincidencias encontradas.

Consideraciones importantes:

- Esta lista debe incluir un sistema de paginación y mostrar 5 ofertas por página.

Pruebas Funcionales

[Prueba21] Hacer una búsqueda con el campo vacío y comprobar que se muestra la página que corresponde con el listado de las ofertas existentes en el sistema



[Prueba22] Hacer una búsqueda escribiendo en el campo un texto que no exista y comprobar que se muestra la página que corresponde, con la lista de ofertas vacía.

3.10 Usuario registrado: Comprar una oferta

Sobre el listado resultante de una búsqueda de ofertas, un usuario podrá comprar una oferta haciendo click en el botón “Comprar” correspondiente. Sólo se permite la compra de una oferta si el Contador de dinero del Usuario es igual o superior al precio de la misma. Al comprar una oferta, se deberán realizar las siguientes acciones:

- Decrementar el contador de dinero del comprador en el precio que tenga la oferta.
- Marcar la oferta como comprada para evitar que vuelva a ser vendida.

En caso de que se intente comprar una oferta de mayor cantidad a la disponible se deberá indicar que el comprador no tiene saldo suficiente.

Pruebas Funcionales

[Prueba23] Sobre una búsqueda determinada (a elección del desarrollador), comprar una oferta que deja un saldo positivo en el contador del comprador. Comprobar que el contador se actualiza correctamente en la vista del comprador.

[Prueba24] Sobre una búsqueda determinada (a elección del desarrollador), comprar una oferta que deja un saldo 0 en el contador del comprador. Comprobar que el contador se actualiza correctamente en la vista del comprador.

[Prueba25] Sobre una búsqueda determinada (a elección del desarrollador), intentar comprar una oferta que esté por encima de saldo disponible del comprador. Y comprobar que se muestra el mensaje de saldo no suficiente.

3.11 Usuario registrado: Ver el Listado de ofertas compradas

Un usuario deberá disponer de una opción que le muestre el listado de ofertas que haya comprado mostrando para cada oferta los datos: *título, detalle, precio y el email del vendedor*.

Consideraciones importantes:

- No es necesario incluir sistema de paginación en este listado.

Pruebas Funcionales

[Prueba26] Ir a la opción de ofertas compradas del usuario y mostrar la lista. Comprobar que aparecen las ofertas que deben aparecer.

3.12 Internacionalización de todas las vistas

Internacionalizar todas las vistas implementadas, incluyendo los fragmentos.

Pruebas Funcionales

[Prueba27] Visualizar al menos cuatro páginas haciendo el cambio español/inglés/español (comprobando que algunas de las etiquetas cambian al idioma correspondiente). Página principal/Opciones principales de usuario/Listado de usuarios /Vista de alta de oferta.



3.13 Seguridad

Deberá diseñarse adecuadamente la política de seguridad con Spring Security para que no existan situaciones de vulnerabilidad en el acceso a recursos/acciones de usuarios registrados.

Pruebas Funcionales

[Prueba28] Intentar acceder sin estar autenticado a la opción de listado de usuarios del administrador. Se deberá volver al formulario de login.

[Prueba29] Intentar acceder sin estar autenticado a la opción de listado de ofertas propias de un usuario estándar. Se deberá volver al formulario de login.

[Prueba30] Estando autenticado como usuario estándar intentar acceder a la opción de listado de usuarios del administrador. Se deberá indicar un mensaje de acción prohibida.

4 Informe de entrega y plantilla de casos de prueba

Se deberá entregar un informe técnico en formato PDF, así como un catálogo de casos de prueba en formato XLSX (que a su vez es formulario de autoevaluación). Se suministran dos plantillas que deberán tomarse como base para la elaboración de dichos documentos:

- Una plantilla Word para el informe (sdi-entrega1-n.docx). (Se entregará en formato PDF).
- Una plantilla Excel para los casos de prueba (sdi-entrega1-n.xlsx). (Se entregar en formato Excel).

Ambos documentos deberán ser renombrados cambiando la n por el código ID-GIT de los miembros del equipo (ej. sdi-entrega1-2021-801-2021-804.pdf, sdi-entrega1-2021-801-2021-804.xlsx).

4.1 Informe de entrega

El contenido del informe deberá ser el siguiente:

- En la portada se deberá incluir:
 - Los datos personales de los autores: Nombre y apellidos, Emails de ambos, Código ID GIT de cada uno y los porcentajes de participación de cada miembro del equipo.
- Razonamiento de los porcentajes de participación.
- Diagrama de clases del modelo.
- Mapa de Navegación.
- Una descripción clara y detallada aquellos aspectos técnicos y/o diseño que considere relevantes.
 - No se trata de describir qué es Spring Boot o Thymeleaf.
 - Se trata de detallar la toma de decisiones a la hora de implementar la práctica en los aspectos técnicos y/o de diseño relevantes.
- Cualquier otra información necesaria para una descripción razonablemente detallada de lo entregado y su correcto despliegue y ejecución (Versiones de java, hsqldb, spring, etc.).



4.2 Plantilla de casos de prueba

La plantilla suministrada para cumplimentar los casos de prueba realizados presenta dos hojas:

- Una (denominada Pruebas) con una serie de tablas, de las cuales el alumno deberá rellenar aquella indicada en amarillo (Casos de Prueba) y donde deberá reflejar los casos de prueba que haya diseñado, el estado de dicho caso de prueba y una explicación/aclaración sobre el mismo en caso de fallo o no haber rellenado el caso. También en el caso de incluir casos de prueba extra deberá incluir en la columna explicación/aclaración en que consiste dicha prueba.
- Una segunda hoja (Instrucciones) con las instrucciones para cumplimentar la primera hoja.

5 Corrección por pares

Una vez finalizado el plazo de entrega el 21 de marzo, **será obligatorio completar una tarea por de corrección por pares.**

Cada alumno, deberá rellenar una plantilla de corrección que estará disponible en el Campus Virtual, tal y como se indica a continuación:

- Cada alumno, independientemente de que forme equipo con otro, deberá rellenar de **forma individual** la plantilla aplicándola al ***código que ha entregado.***
- Cada alumno, independientemente de que forme equipo con otro, deberá rellenar de **forma individual** la plantilla aplicándola al ***código de la práctica de otro compañero.*** Esta otra práctica, será proporcionada en tiempo y forma por los profesores de la asignatura mediante el Campus Virtual.

El plazo para cumplimentar esta plantilla para los dos casos (práctica propia y práctica de compañero), será de 15 días una vez proporcionado el material necesario.

6 Requisitos OPCIONALES

6.1 Usuario registrado: Enviar mensajes a una oferta

Tomando como punto de partida el listado resultante de una búsqueda de ofertas (un botón o enlace para enviar mensajes de oferta), un usuario podrá enviar mensajes (tipo chat) al usuario de una oferta (un enfoque gráfico propuesto sería un pequeño formulario donde meter el mensaje y una tabla con scroll donde se van metiendo los mensajes de uno y otro). Se abrirá una conversación por cada oferta entre cada par de usuarios. La primera vez que un usuario envíe un mensaje a una oferta se creará dicha conversación y, a partir de ahí, los mensajes enviados por ambos lados se irán incorporando a la conversación. Para cada mensaje se mostrará y almacenará: *el autor del mensaje, la fecha/hora del mensaje y el texto del mismo* (similar a WhatsApp).

Pruebas Funcionales

[Prueba31] Sobre una búsqueda determinada de ofertas (a elección de desarrollador), enviar un mensaje a una oferta concreta. Se abriría dicha conversación por primera vez. Comprobar que el mensaje aparece en el listado de mensajes.

[Prueba32] Sobre el listado de conversaciones enviar un mensaje a una conversación ya abierta. Comprobar que el mensaje aparece en la lista de mensajes.



6.2 Usuario registrado: Ver el listado de conversaciones

Un usuario podrá ver un listado de las conversaciones abiertas y podrá reanudar cada una de las conversaciones enviando un nuevo mensaje y viendo los anteriores. El listado mostrará el email del ofertante y el título de la oferta además de un botón/enlace para reanudar la conversación.

Pruebas Funcionales

[Prueba33] Mostrar el listado de conversaciones ya abiertas. Comprobar que el listado contiene las conversaciones que deben ser.

6.3 Usuario registrado: Eliminar una conversación

Sobre el listado anterior se podrá borrar una conversación incluyendo un botón/enlace “Eliminar” para cada conversación. Al pinchar sobre el botón/enlace se eliminará dicha conversación y todos los datos relacionados.

Pruebas Funcionales

[Prueba34] Sobre el listado de conversaciones ya abiertas. Pinchar el enlace Eliminar de la primera y comprobar que el listado se actualiza correctamente.

[Prueba35] Sobre el listado de conversaciones ya abiertas, pulsar el enlace Eliminar de la última y comprobar que el listado se actualiza correctamente.

6.4 Usuario registrado: Marcar una oferta como destacada

Un usuario puede marcar una oferta como Destacada pagando 20 euros (cantidad fija sin fecha de caducidad). Una oferta podrá marcarse como Destacada tanto al crearla (mediante un check en el formulario de creación de una oferta) como mediante un enlace Normal/Destacada en el listado de ofertas propias de un usuario.

Esta oferta aparecerá en una sección destacada en la vista de la página principal de opciones privadas de Usuario con el Botón/enlace de Compra/Vendido. El contador de dinero del usuario se decrementará en 20 euros.

Pruebas Funcionales

[Prueba36] Al crear una oferta marcar dicha oferta como destacada y a continuación comprobar: i) que aparece en el listado de ofertas destacadas para los usuarios y que el saldo del usuario se actualiza adecuadamente en la vista del ofertante (-20).

[Prueba37] Sobre el listado de ofertas de un usuario con menos de 20 euros de saldo, pinchar en el enlace Destacada y a continuación comprobar: que aparece en el listado de ofertas destacadas para los usuarios y que el saldo del usuario se actualiza adecuadamente en la vista del ofertante (-20).

[Prueba38] Sobre el listado de ofertas de un usuario con menos de 20 euros de saldo, pinchar en el enlace Destacada y a continuación comprobar que se muestra el mensaje de saldo no suficiente.



7 Proyecto a entregar, pruebas automatizadas y el protocolo de prueba

Se entregará un proyecto Spring Boot con la librería JUnit así como con las librerías del Framework Selenium (Se suministrará en el campus virtual un proyecto plantilla de ejemplo). Este proyecto contendrá dos carpetas de código fuente Java:

- El proyecto Spring Boot estará en la carpeta **src/main/java** y
- En la carpeta **src/test/java** se incluirán todas las clases de prueba tal cual se ha indicado en la clase sobre Selenium. Los casos de prueba mínimos (válidas e inválidas) que deberán realizarse se encuentran al final de cada requisito. Así mismo todas las pruebas deben incluir el clic en las opciones de menú correspondientes, escritura de datos... tal y como lo haría un usuario real.

De este modo este proyecto se puede ejecutar como proyecto Spring Boot (Run As Spring Boot App) o como proyecto de pruebas (Run As JUnit Test).

Para probar cada proyecto el profesor realizará los siguientes pasos:

- a) Lanzará la base de datos hsqldb suministrada por el alumno.
- b) Importará el proyecto suministrado.
- c) Se ejecutará el proyecto importado como Spring Boot (Run As Spring Boot App). El servidor Tomcat de la aplicación Spring Boot quedará a la escucha si todo fue bien.
- d) Se ejecutará el proyecto importado como proyecto Junit (Run As JUnit Test) para lanzar las pruebas.

Consideraciones importantes:

- Asegurarse que el esquema de base de datos se crea al iniciar la aplicación.

8 Aspectos Generales

1.1 Seguridad

Deberán tenerse en cuenta los siguientes aspectos de seguridad:

- Emplear la técnica de autenticación/autorización más adecuada a este contexto.
- En caso necesario comprobar el rol de cada usuario para el acceso a funcionalidades dependientes de rol.
- Registrar el acceso de los usuarios y las diferentes acciones de los usuarios en un Logger (por ejemplo `org.slf4j.Logger`).

8.1 Arquitectura

La aplicación deberá estar obligatoriamente diseñada siguiendo el patrón arquitectónico visto en clase. Usando siempre de forma correcta controladores, servicios, repositorios y vistas. La utilización incorrecta de elementos de esta arquitectura **será fuertemente penalizada** (por ejemplo, implementar parte de la lógica de negocio de la aplicación en un controlador, acceder a un repositorio desde un controlador, configurar una vista desde un servicio, etc.).



Otros aspectos que serán evaluados

- Claridad y calidad de la implementación del código Java y cumplimiento de *Java Code Conventions*.
- Calidad de la implementación de las vistas usando todas las funcionalidades vistas en clase (fragmentos, replace, etc.)
- Internacionalización de la aplicación, debe estar disponible en inglés y español.
- Validaciones en los formularios.
- Diseño de la interfaz de la aplicación.

8.2 Entrega

El número n de cada trabajo será el ID-GIT del integrante del grupo (en caso de un integrante) o la concatenación de los ID-GIT (en caso de dos ej: 2021-801-2021-804) Según ese número se deberá crear un proyecto STS Spring Boot/Junit con sus pruebas unitarias (tal cual se ha visto en clase todo en un mismo proyecto) con el nombre **sdi-entrega1-n**.

Según lo anterior la entrega consistirá en los siguientes puntos:

- 1) Subir el proyecto a un repositorio GIT con nombre **sdi-entrega1-n**. E invitar al usuario **sdigithubuniovi** como colaborador. Sobre dichos repositorios deberán realizarse actualizaciones frecuentes para que se pueda hacer un seguimiento del ritmo de trabajo.
- 2) Subir a la tarea del Campus Virtual correspondiente un archivo ZIP (usando el formato ZIP) con el nombre **sdi- entrega1-n.zip** (en minúsculas) y que deberá contener en su raíz:
 - El INFORME OBLIGATORIO en formato PDF con nombre **sdi-entrega1-n.pdf** y el archivo Excel con el catálogo de casos de prueba (**sdi- entrega1-n.xlsx**).
 - El proyecto Spring Boot/JUnit en formato carpeta (no comprimido) con el nombre. **\sdi-entrega1-n**
 - Una copia **completa** de la base de datos que utiliza la aplicación (toda la carpeta hsqldb renombrada como hsqldb-n). (Tanto si los datos se generan automáticamente mediante código como si no, es necesario incluir esta carpeta).
 - En resumen, el zip deberá contener en su raíz (MUY IMPORTANTE: el proyecto Spring Boot/Junit deberán renombrarse con la n correspondiente dentro de STS antes de exportarlos para meterlos en el zip):
 - **sdi-entrega1-n.pdf** (Archivo PDF suministrado y renombrado cambiado la n).
 - **sdi-entrega1-n.xlsx** (Archivo Excel suministrado y renombrado cambiado la n).
 - **sdi-entrega1-n** (Carpeta del proyecto Spring Boot/Junit renombrada cambiado la n)
 - **hsqldb-n** (Carpeta con la base de datos renombrada cambiado la n).

Cuando se abra el plazo para la corrección por pares, será obligatorio realizar la tarea propuesta en el apartado **15. Corrección por pares**



8.3 Fecha máxima de entrega

La fecha de entrega el día **21 de marzo a las 23:55**. Posteriormente, se establecerá el plazo de entrega para la corrección por pares.

Consideraciones importantes:

- *No se aceptará ningún trabajo fuera de plazo.*
- *Todos los trabajos deben entregarse por el campus virtual.*

9 Evaluación

La calificación máxima que se puede obtener será de 11 puntos, en base al siguiente reparto: 7 puntos pertenecen a la funcionalidad obligatoria, 4 puntos a la parte opcional.

Se penalizará:

- Que la compilación del código genere “*warnings*”.
- Que se presenten problemas durante el despliegue.