

ESPECIFICACIÓN DE REQUISITOS "UCM RAILS"

GRUPO 8
TRABAJO IS2



PROFESORA: MARTA LÓPEZ FERNÁNDEZ

Índice

DESCRIPCIÓN DEL PROYECTO.	1
Tema.	1
Introducción.	1
Usuarios.	1
Gestión de viajes.	1
Compra de billetes.	1
CASOS DE USO.	2

DESCRIPCIÓN DEL PROYECTO.

Tema.

Aplicación destinada para la gestión de la compraventa de billetes y rutas de una compañía de transporte ferroviario.

Introducción.

Se llevará a cabo el diseño e implementación de una aplicación con el fin de gestionar el transporte ferroviario, además de ofrecer de forma óptima sus servicios. Esta aplicación está destinada para dos tipos de usuarios: los administradores de la compañía ferroviaria y los viajeros. La funcionalidad de esta se basará en la búsqueda y compraventa de billetes, así como en la gestión eficiente de rutas de la compañía.

Usuarios.

Cada cliente dado de alta en dicha plataforma puede comprar un billete asociado a un asiento de un tren, dependiendo el precio del tipo de tren en el que se viaje y de los servicios extra seleccionados (uso de cafetería o necesidad de servicios especiales). Los administradores, por su parte, gestionan las rutas, añadiendo o quitando estaciones cuando sea necesario.

Gestión de viajes.

Los usuarios con rol de administrador pueden crear nuevas rutas. Para ello, seleccionarán un origen y un destino e indicarán la fecha, las horas de parada en cada estación, el tipo de tren que realizará el viaje y un nombre único asignado a la ruta. Además, pueden añadir o quitar estaciones en la base de datos.

Compra de billetes.

El usuario elige el trayecto deseado, obtenido como resultado de la búsqueda de viajes y se le informará de la disponibilidad del mismo para proceder a su pago o a una nueva búsqueda. Además, el cliente puede anular un viaje pendiente, valorar uno realizado y consultar su historial de viajes.

CASOS DE USO.

Aclaraciones

1. Los viajes son EXCLUSIVAMENTE de origen a destino
2. Una estación tiene como clave primaria sus dos únicos atributos. (Ej: Dos estaciones con el mismo nombre una en Madrid y otra en Barcelona, y dos estaciones en Madrid con distinto nombre)
3. La gestión del pago corre a cuenta de otra aplicación gestora de transacciones ya que una aplicación de viajes no debería tener acceso a la cuenta bancaria del usuario.
4. La clase tren es similar a lo que sería un viaje (ej: tren/viaje a las 13:05 de la ruta Barcelona-Madrid). Por ello si elimino un tren se eliminan los respectivos billetes asociados.No se trata en del tren físico.

CASO DE USO #1	Registro del usuario
Objetivo en contexto	El usuario se registra en la aplicación, generando unas nuevas credenciales en la base de datos con la información introducida.
Entradas	
Precondiciones	No se ha accedido con ninguna credencial a la aplicación (Se ha iniciado la aplicación y no se ha identificado ningún usuario).
Salidas	
Postcondición si éxito	Los datos del usuario quedan registrados y asociados a una cuenta personal en la base de datos.
Postcondición si fallo	El registro no se lleva a cabo en la base de datos.
Actores	El usuario y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none">1. Preguntar al usuario si está intentando crear una cuenta de administrador.2. El usuario introduce los datos (nombre, apellidos, DNI y contraseña) en los campos correspondientes.Si ha elegido ser administrador se le pedirá una contraseña extra de validación de administrador. si erro, S-4.3. Lectura de campos. Si error S-1.4. Verificación de datos. Si error S-2.5. Comprobar que el DNI no está asociado a otra cuenta. Si error S-3..6. Inserción de las credenciales en la base de datos.

Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. Alguno de los campos está vacío.</p> <ol style="list-style-type: none"> 1. Indicar cuál de los campos está vacío. 2. Volver al paso 1 de la secuencia normal. <p>S-2. Alguno de los campos no es válido.</p> <ol style="list-style-type: none"> 1. Indicar cuál de los datos no es válido (se ha introducido algo distinto a un entero para la <u>edad</u>; la <u>contraseña</u> debe tener al menos un número, una letra, un carácter especial y tener una longitud mínima de 8 caracteres; el <u>DNI</u> debe de tener el formato correspondiente [8 números seguidos de una letra mayúscula]). 2. Indicar cuál es el problema de formato que lo invalida con un mensaje . 3. Volver al paso 1 de la secuencia normal. <p>S-3. Documento de identidad asociado a otra cuenta.</p> <ol style="list-style-type: none"> 1. Indicar que existe una cuenta ya asociada al dni proporcionado. 2. Volver al paso 1 de la secuencia normal. <p>S-4. Si el campo de clave está vacío o no es la clave válida “1234” (Previamente proporcionada a la empresa para la creación de perfiles de administradores)</p> <ol style="list-style-type: none"> 1. Indicar que la clave de administrador no es válida. 2. Volver al paso 5.b.
-------------------------	---

CASO DE USO #2	Inicio de sesión
Objetivo en contexto	El usuario accede a su cuenta en la aplicación.
Entradas	
Precondiciones	No se ha accedido con ninguna credencial a la aplicación (se ha iniciado la aplicación y no se ha identificado ningún usuario).
Salidas	
Postcondición si éxito	El usuario accede correctamente a la aplicación, teniendo así acceso a sus funcionalidades.
Postcondición si fallo	El usuario no es capaz de acceder a las funcionalidades de la aplicación.
Actores	El usuario y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none"> 1. El usuario introduce el DNI y la contraseña en los campos correspondientes.

	<ol style="list-style-type: none"> 2. Lectura de campos. Si error S-1. 3. Verificación de campos. Si error S-2. 4. Comprobar que la cuenta existe. Si error S-3. 5. Comprobar que la contraseña es la asociada a la cuenta del usuario con el DNI proporcionado. Si error S-4. 6. Ejecución de la pantalla de la aplicación principal.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. Alguno de los campos está vacío</p> <ol style="list-style-type: none"> 1. Indicar cuál de los campos está vacío. 2. Volver al paso 1 de la secuencia normal. <p>S-2. El DNI o la contraseña no tienen un formato válido</p> <ol style="list-style-type: none"> 1. Indicar cuál de los campos no es válido. 2. Indicar cuál es el problema que lo invalida. 3. Volver al paso 1 de la secuencia normal. <p>S-3. No existe una cuenta asociada al DNI proporcionado</p> <ol style="list-style-type: none"> 1. Indicar al usuario que para acceder debe crear antes una cuenta o debe revisar bien los datos proporcionados. 2. Volver al paso 1 de la secuencia normal <p>S-4. La contraseña no coincide con la cuenta</p> <ol style="list-style-type: none"> 1. Mostrar error: "Error: Contraseña incorrecta. ". 2. Volver al paso 1.

CASO DE USO #3	Búsqueda de viajes
Objetivo en contexto	Ofrecer una lista de los distintos viajes según los filtros introducidos por el usuario en el buscador.
Entradas	
Precondiciones	El usuario ha iniciado sesión previamente en la aplicación y es cliente.
Salidas	Lista del conjunto de viajes de acuerdo a las condiciones introducidas por el usuario.
Postcondición si éxito	El sistema obtiene los viajes que concuerdan con los parámetros seleccionados.
Postcondición si fallo	La búsqueda no se ha llevado a cabo.
Actores	El usuario (cliente) y la base de datos.
Secuencia normal	Paso. Acción

	<ol style="list-style-type: none"> 1. Esperar a que el usuario introduzca localidad de destino y de origen, día y el tipo de tren en el que quiere viajar en los campos correspondientes. 2. Lectura de campos. Si error S-1. 3. Verificación de campos (Comprobar que las localizaciones están asociadas a estaciones contenidas en el repertorio de la compañía; <u>día</u> posterior a la fecha actual). Si error S-2. 4. Comprobar que existen en la base de datos de viajes que concuerden con la fecha, las localizaciones y (si se ha introducido) el tipo de tren. Si error S-3. 5. Muestra el conjunto de viajes resultantes de la búsqueda en la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. Alguno de los campos que no puede ser nulo está vacío (localización de origen y de destino, día)</p> <ol style="list-style-type: none"> 1. Indicar cuál de los campos está vacío. 2. Volver al paso 1 de la secuencia normal. <p>S-2. Alguno de los campos no es válido</p> <ol style="list-style-type: none"> 1. Indicar cuál de los campos no es válido (Ej: No se han encontrado estaciones en las localizaciones aportadas o la fecha no se ha puesto en el formato indicado por pantalla). 2. Indicar cuál es el problema que lo invalida. 3. Volver al paso 1 de la secuencia normal. <p>S-3. No existen viajes registrados para los parámetros seleccionados.</p> <ol style="list-style-type: none"> 1. Indicar que no hay ningún viaje registrado que cumpla las condiciones requeridas. 2. Volver al paso 1 de la secuencia normal.

CASO DE USO #4	Elección de billete a comprar
Objetivo en contexto	El usuario elige el viaje entre los disponibles en la lista y los servicios que quiere contratar en su viaje.
Entradas	Lista del conjunto de viajes obtenidos del caso de uso: “búsqueda de viajes” .
Precondiciones	Se ha iniciado sesión previamente en la aplicación como cliente.

Salidas	Billete a comprar (Datos necesarios para su registro en la base de datos)
Postcondición si éxito	Se procede al caso de uso de pago de billetes.
Postcondición si fallo	No se genera ningún billete a comprar.
Actores	El usuario (cliente) y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none"> 1. Seleccionar viaje de la lista de entrada proporcionada. 2. Comprobar que en el viaje aún quedan billetes disponibles. Si error S-1. 3. Preguntar al usuario si desea contratar algún servicio. <ol style="list-style-type: none"> 1. Si responde que no: Ir al paso 4. 2. Si responde que sí: Se muestran los distintos servicios ofrecidos por la compañía para que se seleccionen. 3. Se añaden los precios de los servicios seleccionados al precio del billete. 4. Proceder con el caso de uso: “Pago de billete”.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. No quedan asientos libres en el viaje.</p> <ol style="list-style-type: none"> 1. Informar del error. 2. Volver al paso 1 de la secuencia normal.

CASO DE USO #5	Añadir servicio
Objetivo en contexto	Agregar un nuevo servicio a los ofrecidos por la compañía.
Entradas	
Precondiciones	Se ha iniciado sesión previamente en la aplicación como administrador.
Salidas	
Postcondición si éxito	Un nuevo servicio se ha agregado.
Postcondición si fallo	No se puede realizar la acción y no se registra en la base de datos.
Actores	El usuario (administrador) y la base de datos.

Secuencia normal	Paso. Acción 1. Introducir el nombre del servicio y el precio del servicio. 2. Validar y comprobar los datos. Si error S-1
Secuencias alternativas	Paso. Acción S-1. Ya existe un servicio con dicho nombre. 1. El sistema muestra un mensaje <i>“Error: el servicio ya existe.”</i> 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #6	Modificar servicio
Objetivo en contexto	Modificar un servicio de los ofrecidos por la compañía.
Entradas	
Precondiciones	Se ha iniciado sesión previamente en la aplicación como administrador.
Salidas	
Postcondición si éxito	Se modifica correctamente el servicio deseado.
Postcondición si fallo	No se puede llevar a cabo la acción y no se registra en la base de datos los cambios realizados.
Actores	El usuario (administrador) y la base de datos.
Secuencia normal	Paso. Acción 1. Introducir el nombre del servicio y comprobar que existe en la base de datos. Si error S-1. 2. Se introduce el nuevo precio del servicio, modificándose en la base de datos.
Secuencias alternativas	Paso. Acción S-1. No existe un servicio con dicho nombre. 1. El sistema muestra un mensaje <i>“Error: el servicio no existe.”</i> 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #7	Eliminar servicio
-----------------------	--------------------------

Objetivo en contexto	Eliminar un servicio de los ofrecidos por la compañía
Entradas	
Precondiciones	Se ha iniciado sesión previamente en la aplicación como administrador.
Salidas	
Postcondición si éxito	Se ha eliminado un servicio.
Postcondición si fallo	No se puede realizar la acción y no se registra en la base de datos el cambio deseado.
Actores	El usuario (administrador) y la base de datos.
Secuencia normal	Paso. Acción <ol style="list-style-type: none"> 1. Introducir el nombre del servicio y comprobar que existe en la base de datos. Si error S-1. 2. Se borra de la base de datos.
Secuencias alternativas	Paso. Acción S-1. No existe un servicio con dicho nombre. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje <i>“Error: el servicio no existe.”</i> 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #8	Pago de billete
Objetivo en contexto	El usuario realiza la compra de un billete para el viaje especificado.
Entradas	Billete procedente de caso de uso: “Elección de billete a comprar” .
Precondiciones	Se ha realizado la selección del billete a comprar.
Salidas	
Postcondición si éxito	El billete deja de estar disponible para su compra y se realiza la transacción de dinero desde el saldo del usuario.
Postcondición si fallo	No se realiza la transacción.
Actores	El usuario(cliente) y la base de datos.
Secuencia normal	Paso. Acción

	<ol style="list-style-type: none"> 1. Mostrar los datos del billete a comprar. 2. Comprobar que el usuario tiene saldo suficiente en su cuenta para pagar el precio del billete. Si error S-1. 3. El billete pasa a no estar disponible para su compra.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. El usuario no tiene saldo suficiente en la cuenta para pagar.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje “Error: No hay saldo suficiente para pagar el billete.” 2. Se sale del caso de uso.

CASO DE USO #9	Creación de rutas
Objetivo en contexto	El administrador inserta una nueva ruta en la base de datos.
Entradas	
Precondiciones	El usuario ha iniciado sesión previamente y es administrador.
Salidas	
Postcondición si éxito	La ruta se registra en la base de datos.
Postcondición si fallo	No se puede realizar la acción y no se registra en la base de datos.
Actores	El administrador y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none"> 1. Introducir el nombre de la ruta, la estación de inicio, de destino, horarios de salida, llegada y el precio de la ruta. 2. Validación de ruta. Si error S-1. 3. Validación de estaciones. Si error S-2. 4. Validación de horas. Si error S-3. 5. Se introduce en la base de datos. 6. AÑADIR TREN(Caso de uso 11).
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. Nombre repetido.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje “Error: la ruta ya existe.”

	<p>2. Vuelve al paso 1 de la secuencia normal.</p> <p>S-2. alguna de las estaciones introducidas no existe.</p> <ol style="list-style-type: none"> 1. Se informa del error. 2. Vuelve al paso 2 de la secuencia normal. <p>S-3. Horas incorrectas u hora de destino menor que hora de llegada.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje: "Error al introducir las horas." 2. Vuelve al paso 3 de la secuencia normal.
--	--

CASO DE USO #10	Eliminación de rutas
Objetivo en contexto	El administrador elimina una ruta de la base de datos.
Entradas	
Precondiciones	El usuario ha iniciado sesión previamente y es administrador.
Salidas	
Postcondición si éxito	La ruta se elimina de la base de datos.
Postcondición si fallo	No se puede realizar la acción y no se elimina la ruta de la base de datos.
Actores	El usuario(administrador) y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none"> 1. Introducir nombre de la ruta y se comprueba que existe en la base de datos. Si error S-1. 2. Se eliminan los trenes asociados (Caso de uso: "Eliminar tren") y la ruta de la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. No existe en la base de datos una ruta con el nombre proporcionado.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje "Error: la ruta no existe." 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #11	Añadir tren
------------------------	--------------------

Objetivo en contexto	Asociar un tren a una ruta
Entradas	
Precondiciones	Se ha iniciado sesión previamente en la aplicación como administrador.
Salidas	
Postcondición si éxito	Un tren se ha añadido a cierta ruta.
Postcondición si fallo	No se puede realizar la acción y no se registra en la base de datos el tren a la ruta deseada.
Actores	El usuario (administrador) y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none"> 1. Introducir el identificador del tren, el tipo de tren, la hora de salida y de llegada. 2. Comprobar que es único en la base de datos. Si error S-1. 3. Comprobar que el resto de datos sean correctos. Si error S-2. 4. Se registra finalmente en la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. Ya existe un tren con dicho identificador.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje <i>"Error: el tren ya existe."</i> 2. Vuelve al paso 1 de la secuencia normal. <p>S-2. El dato no es válido</p> <ol style="list-style-type: none"> 1. Muestra el dato inválido. 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #12	Modificar tren
Objetivo en contexto	Modificar un tren asociado a una ruta.
Entradas	
Precondiciones	Se ha iniciado sesión previamente en la aplicación como administrador.
Salidas	

Postcondición si éxito	Se modifica un tren.
Postcondición si fallo	No se puede realizar la acción y no se registra en la base de datos los cambios realizados.
Actores	El usuario (administrador) y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none"> 1. Introducir el identificador del tren , el tipo de tren, el horario y el nombre de la ruta. 2. Comprobar en la base de datos. Si error S-1. 3. Comprobar el resto de los elementos. Si error S-2 4. Busca en la base de datos la ruta. Si error S-3. 5. Modifica en la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. No existe un tren con dicho identificador.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje <i>"Error: el tren no existe."</i> 2. Vuelve al paso 1 de la secuencia normal. <p>S-2. El dato no es válido</p> <ol style="list-style-type: none"> 1. Muestra el dato inválido. 2. Vuelve al paso 1 de la secuencia normal. <p>S-3. No existe una ruta con dicho identificador.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje <i>"Error: la ruta no existe"</i>. 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #13	Eliminar tren
Objetivo en contexto	Eliminar un tren asociado a una ruta.
Entradas	
Precondiciones	Se ha iniciado sesión previamente en la aplicación como administrador.
Salidas	
Postcondición si éxito	Se elimina de la base de datos el tren deseado.
Postcondición si fallo	No se puede realizar la acción, por lo tanto no se elimina el tren deseado de la base de datos.
Actores	El usuario (administrador) y la base de datos.
Secuencia normal	Paso. Acción

	<ol style="list-style-type: none"> 1. Introducir el identificador del tren y comprobar que existe en la base de datos. Si error S-1. 2. Eliminar el tren de la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. No existe un tren con dicho identificador.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje <i>“Error: el tren no existe.”</i> 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #14	Anular billete
Objetivo en contexto	El usuario cancela un viaje y se procede a su reembolso.
Entradas	
Precondiciones	El usuario ha comprado un billete.
Salidas	
Postcondición si éxito	El billete asociado queda disponible y se devuelve el importe.
Postcondición si fallo	El billete no se anula.
Actores	El usuario y la base de datos.
Secuencia normal	<p>Paso. Acción</p> <ol style="list-style-type: none"> 1. El cliente selecciona en la interfaz la opción de anulación de viajes. 2. Se muestran los billetes asociados al usuario. 3. El usuario selecciona el billete que quiere anular. Si error S-1. 4. Se reembolsa al saldo del cliente el importe del billete. 5. El billete queda disponible para su compra por otros viajeros.
Secuencias alternativas	<p>S-1: Si la fecha del viaje es anterior a la fecha del sistema.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje: <i>“Error: el billete caducó.”</i> 2. Vuelve al paso 1 de la secuencia normal.

CASO DE USO #15	Consulta viajes realizados
------------------------	-----------------------------------

Objetivo en contexto	El cliente visualiza su historial de viajes realizados.
Entradas	
Precondiciones	El usuario ha iniciado sesión, es cliente y ya ha realizado algún viaje.
Salidas	Lista de viajes realizados.
Postcondición si éxito	El sistema muestra todos los viajes realizados por el cliente.
Postcondición si fallo	No se muestra el registro de viajes.
Actores	Usuario (cliente) y la base de datos.
Secuencia normal	Paso. Acción <ol style="list-style-type: none"> 1. El cliente selecciona la opción de ver la lista con los viajes que ha realizado. 2. Se muestran los viajes asociados al cliente cuya fecha es anterior a la fecha del sistema.
Secuencias alternativas	Paso. Acción

CASO DE USO #16	Valoración del viaje
Objetivo en contexto	El cliente valora el viaje.
Entradas	Lista procedente de caso de uso: "Consulta viajes realizados".
Precondiciones	El usuario tiene una cuenta y al menos un viaje registrado con la compañía.
Salidas	
Postcondición si éxito	Se registra en la base de datos la nota asignada por el cliente al viaje en el billete que le corresponde.
Postcondición si fallo	No se registra en la base de datos una valoración nueva.
Actores	Usuario (cliente) y la base de datos.
Secuencia normal	Paso. Acción <ol style="list-style-type: none"> 1. El cliente selecciona la opción de valorar un viaje realizado.

	<ol style="list-style-type: none"> 2. El cliente valora el servicio ofrecido por la compañía de 0 a 5 estrellas. Si error S-1. 3. El cliente manda la valoración y el sistema la registra en la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1: Valoración no válida.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje “Error: valoración no válida.” 2. Vuelve al paso 2 de la secuencia normal.

CASO DE USO #17	Añadir Estación
Objetivo en contexto	El administrador registra una estación nueva en la base de datos
Entradas	
Precondiciones	El usuario ha iniciado sesión previamente y es administrador.
Salidas	
Postcondición si éxito	Se registra una nueva estación en la base de datos
Postcondición si fallo	No se registra en la base de datos la nueva estación.
Actores	Usuario (administrador) y la base de datos.
Secuencia normal	<p>Paso Acción</p> <ol style="list-style-type: none"> 1. El administrador selecciona la opción de añadir estación de la interfaz. 2. El usuario introduce el nombre de la estación nueva y su localidad y se comprueba que no existe otra con el mismo nombre en la misma localización en la base de datos. Si error S-1. 3. Los datos introducidos por el usuario sobre la estación, quedan registrados en la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. Estación repetida.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje “Error: la estación ya existe.” 2. Vuelve al paso 1 de la secuencia normal

CASO DE USO #18	Quitar Estación
Objetivo en contexto	El administrador elimina una estación de la base de datos
Entradas	
Precondiciones	El usuario ha iniciado sesión previamente y es administrador.
Salidas	
Postcondición si éxito	Se elimina una estación de la base de datos
Postcondición si fallo	No se elimina de la base de datos la estación.
Actores	Usuario (administrador) y la base de datos.
Secuencia normal	<p>Paso Acción</p> <ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar estación de la interfaz 2. El usuario introduce el nombre de la estación a eliminar y su localidad y se comprueba que existe en la misma localización una estación con ese nombre en la base de datos. Si error S-1. 3. Los datos sobre la estación se eliminan de la base de datos.
Secuencias alternativas	<p>Paso. Acción</p> <p>S-1. No existe en la base de datos una estación con los datos proporcionados.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje <i>“Error: la estación no existe.”</i> 2. Vuelve al paso 1 de la secuencia normal