



VENTANA RUTA

ruta.fxml

Go toViewHelp

Filter By Date


From:  To: 


Filter By Distance & Time


Filter Type

Filter By Localizador

Localizador	Origen	Destino	Distancia	Tiempo	Fecha Creación
Tabla sin contenido					







DESCRIPCIÓN

Ventana que sirve para poder modificar, crear, borrar y ver rutas.
Se podrán consultar datos clave y filtrar por distintos criterios.

CONTROL	ACCIÓN	RESPUESTA
Vista (Stage)	Inicializar (initStage)	<ul style="list-style-type: none">● Establecer el título de la ventana al valor "Ruta".● Mostrar la ventana.● La ventana no debe ser redimensionable.● Si se produce cualquier excepción en esta respuesta, mostrar un mensaje al usuario con el texto de la excepción.● El combo Filter se carga con los dos tipos diferentes de filtros: "Por distancia" y "Por tiempo".● El combo Type se carga con los diferentes tipos de criterio (">","<" y "=").● Llamar al método selectAll para recoger todas las rutas e implementarlas en la tabla● Se cargará la tabla con los datos de las rutas ya creadas y serán editables.

		<ul style="list-style-type: none"> • Las columnas localizador, distancia y tiempo no serán editables • Se podrá multi seleccionar • La distancia aproximada se calculará en la clase CalcularDistanciaTiempo en el método calcularDistancia, recibiendo como parámetro el origen y el destino • El tiempo aproximado se calculará en la clase CalcularDistanciaTiempo en el método calcularTiempo, recibiendo como parámetro la distancia • Cuando fecha creación esté en modo edición se verá un DatePicker • La fecha tendrá el formato dd/MM/yyyy. • El botón deleteRow estará deshabilitado. • Se enfocara el campo "fromDatePicker". • Si se produce cualquier excepción en esta respuesta, mostrar un mensaje al usuario con el texto de la excepción.
<p>Botón Apply (applyFilterButton)</p>	<p>Pulsar (ActionEvent)</p>	<ul style="list-style-type: none"> • Validar que al menos un campo de fecha esté informado, en caso de no estarlo lanzar la excepción con el siguiente mensaje: "Debe rellenar al menos un campo de fecha" • Validar que las fechas están en el formato correcto (dd/MM/yyyy), en caso de no ser válido lanzar una excepción con el texto "Formato de fecha incorrecto: el formato debe ser dd/MM/yyyy". • Si la fecha From y To están informadas Validar que la fecha From no sea mayor a la fecha To, en caso de no ser válido lanzar una excepción con el texto: "La fecha desde no debe de ser mayor a la fecha hasta". • Se llama al método findRoutesByDate pasándole los valores de los campos From y To. • Los datos que devuelve el método anterior se cargarán en la tabla Rutas • Si se produce cualquier excepción mostrar una alerta con el mensaje de la excepción.

<p>Botón Search (searchButton1)</p>	<p>Pulsar (ActionEvent)</p>	<ul style="list-style-type: none"> • Validar que se ha seleccionado un elemento de cada uno de los dos checkbox, en caso de no ser válido lanzar una excepción con el texto: "Debe seleccionar un filtro y un tipo de filtro". • Validar que el campo de valor esté informado, en caso de no estarlo, lanzar una excepción con el texto: "Debe introducir un valor para poder aplicar el filtro" • Si el valor introducido en el campo valor contiene algo que no sea un número, ya sean símbolos o letras, validar que el único dato introducido sea un número. En caso de no ser válido lanzar una excepción con el texto: "El valor no puede contener nada que no sea un número positivo" . • Si en la combo está seleccionado "Por distancia" Se llama al método findRoutesByDistance pasándole la elección de la combo tipo y el valor del campo de texto. • Si en la combo está seleccionado "Por tiempo" Se llama al método findRoutesByTime pasándole la elección de la combo tipo y el valor del campo de texto. • Los datos que devuelven los métodos anteriores se cargarán en la tabla Rutas • Si se produce cualquier excepción mostrar una alerta con el mensaje de la excepción.

Botón Search 2 (searchButton)	Pulsar (ActionEvent)	<ul style="list-style-type: none"> • Si el valor introducido en el campo valor contiene algo que no sea un número, ya sean símbolos o letras, validar que el único dato introducido sea un número. En caso de no ser válido lanzar una excepción con el texto: "El localizador no puede contener nada que no sea un número positivo" . • Validar que el campo del localizador esté informado, en caso de no estarlo, lanzar una excepción con el texto: "Debe introducir un localizador para poder buscarlo" • Se llama al método findRouteBylocalizer pasándole el valor del textField searchLocalizer. • Los datos que devuelven los métodos anterior se cargarán en la tabla Rutas • Si se produce cualquier excepción mostrar una alerta con el mensaje de la excepción.
TableColumn Origen	Confirmación de edición (onEditCommit)	<ul style="list-style-type: none"> • Validar que el campo de origen contenga un origen válido, en caso de que no lo sea lanzará una excepción con el texto: "El lugar elegido no es válido". • En caso de que también tenga un destino válido. Obtener el objeto Ruta, con el nuevo lugar de origen, distancia y tiempo. • En caso de que no tenga un destino válido todavía. Obtener el objeto Ruta, con el nuevo lugar de origen, distancia y tiempo. Donde la distancia y el tiempo estarán vacíos. • Se llamará al método Update pasando como parámetro el objeto Ruta recibido anteriormente. • Si se produce cualquier excepción mostrar una alerta con el mensaje de la excepción

TableColumn Destino	Confirmación de edición (onEditCommit)	<ul style="list-style-type: none"> • Validar que el campo de destino contenga un destino válido, en caso de que no lo sea lanzará una excepción con el texto: "El lugar elegido no es válido". • En caso de que también tenga un origen válido. Obtener el objeto Ruta, con el nuevo lugar de destino, distancia y tiempo. • En caso de que no tenga un origen válido todavía. Obtener el objeto Ruta, con el nuevo lugar de destino, distancia y tiempo. Donde la distancia y el tiempo estarán vacíos. • Se llamará al método Update pasando como parámetro el objeto Ruta recibido anteriormente. • Si se produce cualquier excepción mostrar una alerta con el mensaje de la excepción
TableColumn Fecha Creación	Confirmación de edición (onEditCommit)	<ul style="list-style-type: none"> • Obtener el objeto de Ruta, con la nueva fecha entrega actualizada • Llamar al método Update pasando como parámetro el objeto Ruta obtenido anteriormente • Si se produce cualquier excepción mostrar una alerta con el mensaje de la excepción
Menu Item Add Route Botón Add Route (addShipmentBtn)	Pulsar (ActionEvent)	<ul style="list-style-type: none"> • Se añadirá una fila de valores por defecto y se llamará a la función Insert, todas las columnas estarán vacías a excepción de la de localizador. • Se mostrará la nueva fila en la tabla. • Si hay algún error al hacer el Insert se lanzará una Excepción y se informará al usuario
Menu Item Delete Route Botón Delete Route (removeShipmentBtn)	Pulsar (ActionEvent)	<ul style="list-style-type: none"> • Para que este botón se habilite el cliente deberá seleccionar una o varias rutas, una vez haga click le pedirá confirmación para borrar la/s ruta/s seleccionada/s • Se mostrará una alerta pidiendo la confirmación al cliente con el mensaje:

		<p>“¿Estás seguro de que quieres borrar la/s ruta/s seleccionada/s?”.</p> <ul style="list-style-type: none"> • En caso de denegar, las líneas seleccionadas no se borrarán • En caso de confirmar se borrarán todas las líneas seleccionadas pasando sus localizadores a la función Delete • Si hay algún error al hacer el Delete se lanzará una Excepción y se informará al usuario
Botón Print Report (printReportBtn)	Pulsar (ActionEvent)	<ul style="list-style-type: none"> • Mostrará una ventana con los datos de la tabla como parametro donde podrá hacer el reporte.
Ventana (Stage)	Cerrar (onCloseRequest)	<p>Pedir confirmación al usuario para salir:</p> <ul style="list-style-type: none"> • Si el usuario confirma, saldrá de la aplicación. • Si no confirma, mantenerse en la ventana