

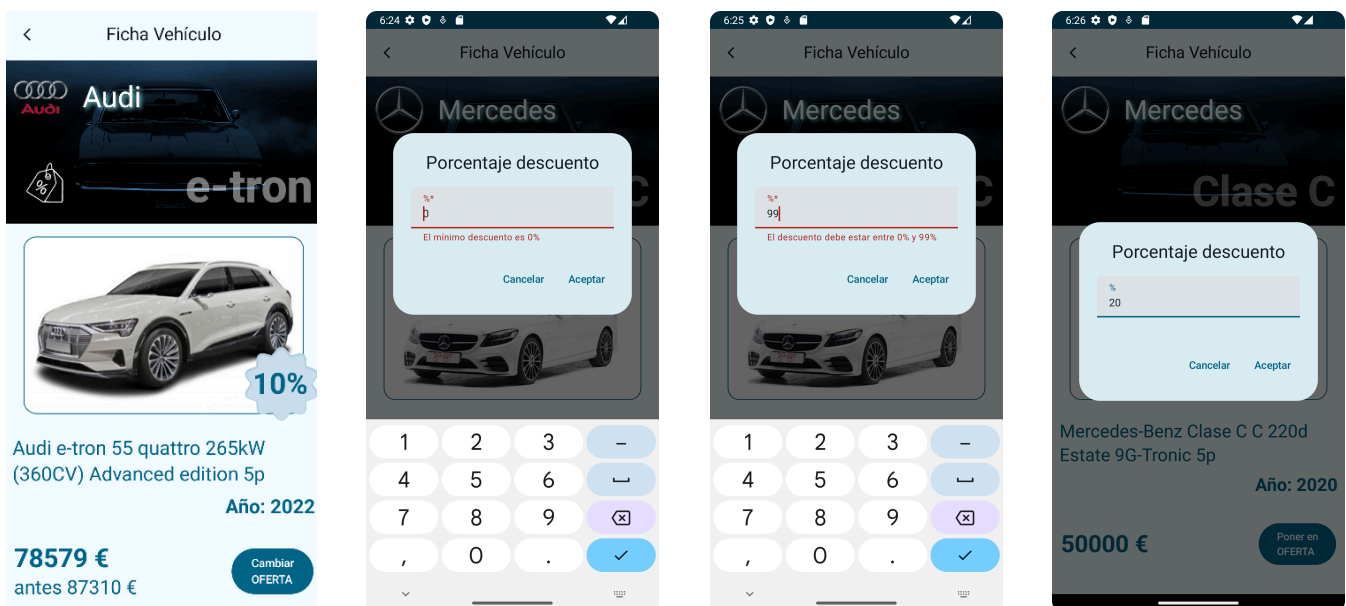
# Ejercicio 2

Descargar este enunciado [pdf](#) o [html](#)

Partiendo del [Ejercicio 1](#), se pide realizar las siguientes tareas:

Completa el código de `FichaCocheScreen.kt` para que muestre la ficha de un coche y en la parte derecha del precio muestre un botón que si el coche no tiene oferta ponga "*Poner en OFERTA*" y si el coche tiene oferta ponga "*Cambiar OFERTA*". Además, deberemos añadir un `Scaffold` con un `CenterAlignedTopAppBar` fijo que permita la navegación hacia atrás como se muestra en el ejemplo.

Al pulsar el nuevo botón, se debe mostrar un `AlertDialog` con un `TextField` para cambiar el porcentaje de descuento del coche. De tal manera que si intentamos escribir borrar todo su contenido, este se ponga a 0 y se muestre el mensaje de error "*El mínimo descuento es 0%*". Además, si intentamos escribir más de 2 dígitos, se muestre el mensaje de error "*El descuento debe estar entre 0% y 99%*". Si el valor de descuento pasa a estar a 0 el coche dejará de estar en oferta.



La gestión del estado de visualización del diálogo, así como la actualización del porcentaje de descuento en la base de datos se realizará en `FichaCocheViewModel.kt` sin necesidad de crear una clase sellada para gestionar los eventos.

Los estados del propio `AlertDialog` puedes gestionarlos en el componente de `FichaCocheScreen.kt` que emite el diálogo, pasando este a ser '*Stateful*', además puedes usar de forma opcional los edit definidos en `FieldTextCommon` y por ende las clases de utilidad de validación utilizadas durante el curso. En otras palabras, `FichaCocheViewModel.kt` solo debería gestionar como estados el `CocheUiState` que se está visualizando y si el diálogo se visualiza o no y definirá un método

`fun onAceptarDialogoDescuento(porcentajeDescuento: Int)` que se le pasará como callback a `FichaCocheScreen` para que este lo invoque cuando se acepte el diálogo.

Para que el teclado del `TextField` permita solo introducir dígitos numéricos, debes establecer la propiedad `keyboardOptions` al valor `KeyboardOptions(keyboardType = KeyboardType.Number)`



### Nota

Es interesante que uses el componente `TextFieldWithErrorState` junto con los interfaces `Validador<T>` y `Validacion`, definidos en `com.github.pmdmiesbalmis.components.validacion`, para comprobar que el porcentaje de descuento introducido es correcto.

Como se comenta en el enunciado, las concrecciones de estos interfaces serán estado definidos dentro de `FichaCocheScreen.kt` antes de visualizar el diálogo.