**PRACTICA 6: CREACIÓN APLICACIÓN MIS LUGARES**

**Parte I: Creación de la aplicación Mis Lugares con barra extendida**

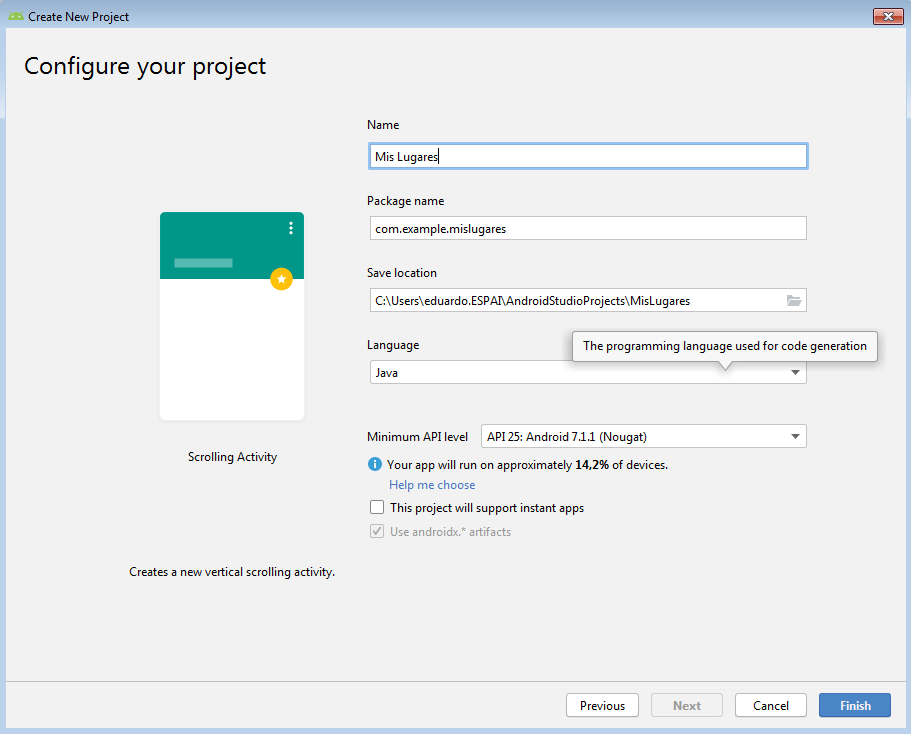
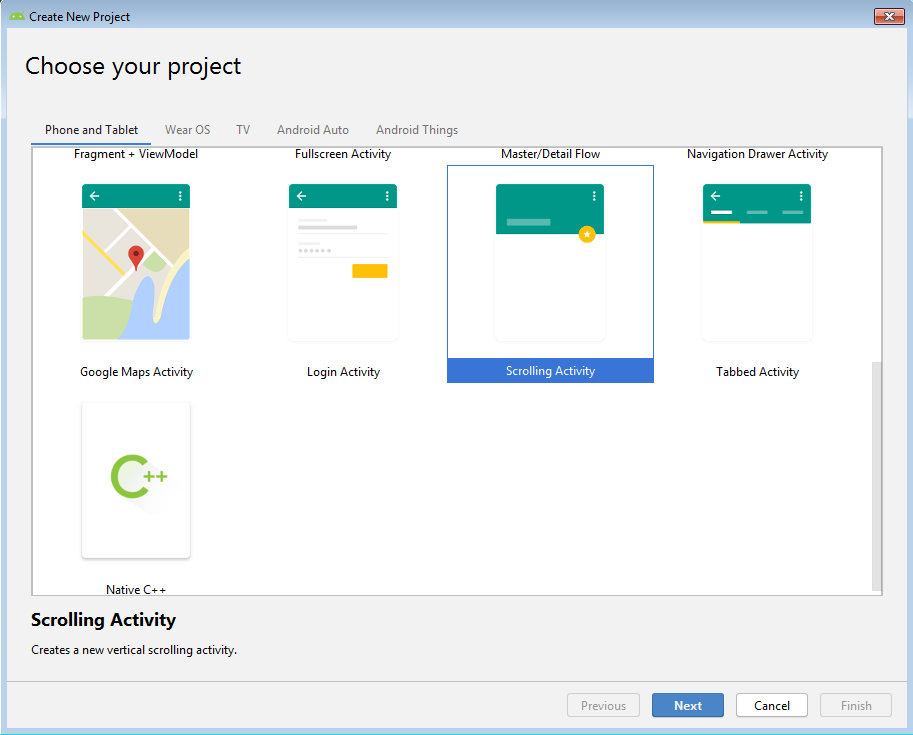
En este curso vamos a crear una aplicación de ejemplo. Tendrá por nombre Mis Lugares y permitirá que los usuarios guarden información relevante sobre los sitios que suelen visitar (restaurantes, tiendas, etc.). En este apartado comenzaremos creando la aplicación Android que solo contendrá una actividad con una barra de acciones extendida.

La barra de acciones extendida es un patrón de diseño propuesto en Material Design que habrás observado en algunas aplicaciones actuales. Al entrar en la actividad la barra de acciones se muestra en modo extendido. Al desplazar el contenido la barra de acciones pasará al tamaño habitual.

Para conseguir este efecto has de utilizar el layout CollapsingToolbarLayout. En este ejercicio se describe cómo hacerlo:

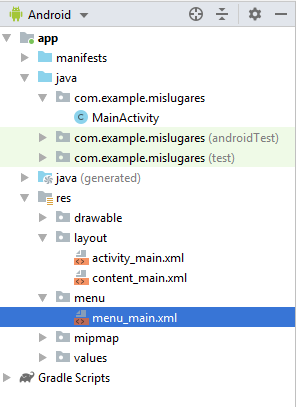
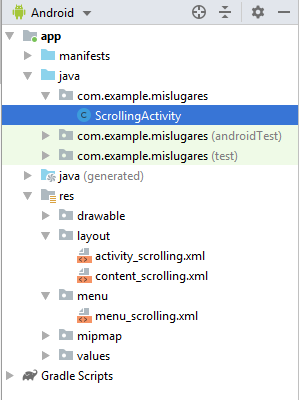
**Paso 1.** Crea un nuevo proyecto con los siguientes datos:

* *Phone and Tablet / Scrolling Activity*
* *Name: Mis Lugares*

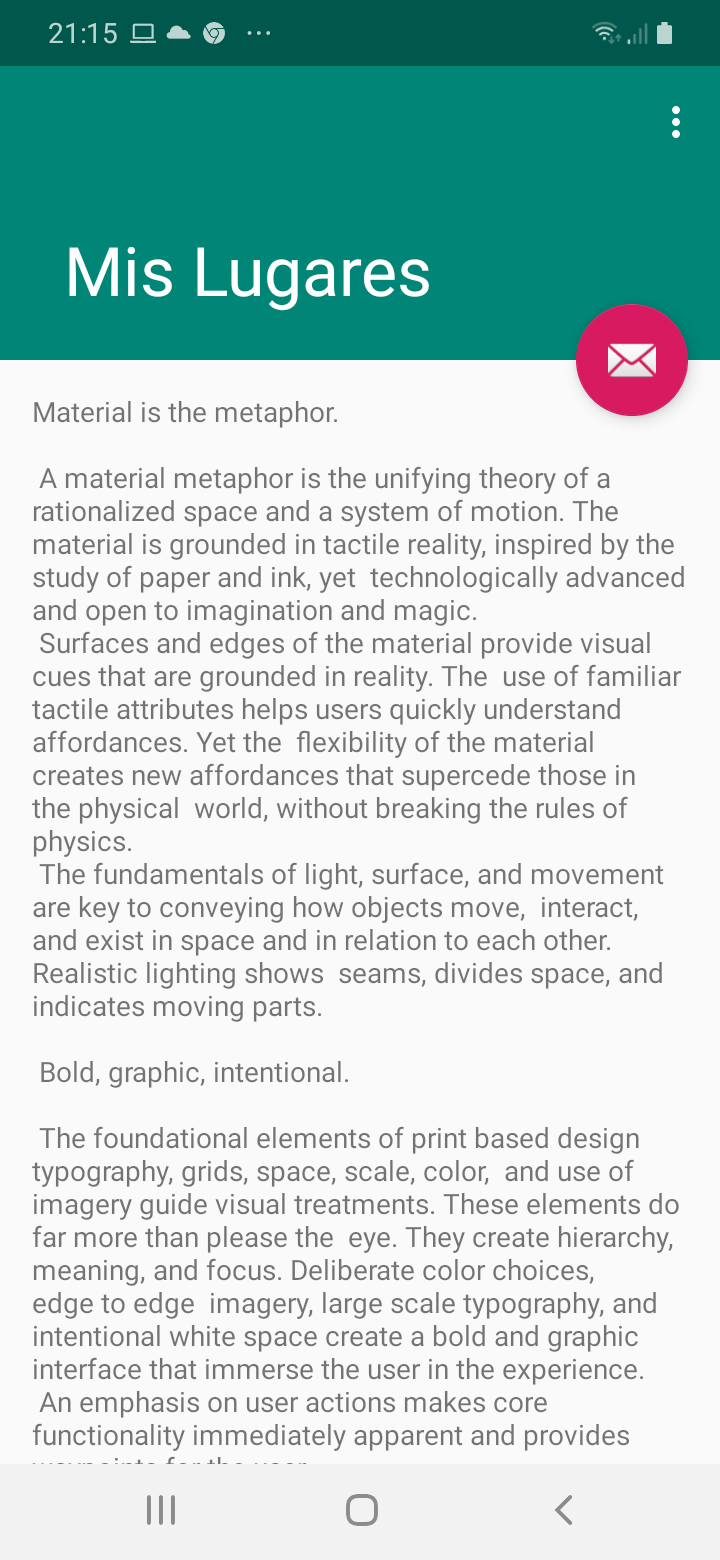


De esta forma se creará una actividad similar a la que hemos creado en el ejercicio anterior. Igualmente dispondrá de una barra de acciones y un botón flotante. Pero ahora el contenido principal podrá desplazarse, a la vez que la barra de acciones reduce su tamaño.

**Paso 2.** En el navegador de proyecto pulsa con el botón derecho sobre la clase ScrollingActivity y selecciona Refactor > Rename. Introduce como nuevo nombre MainActivity. En la carpeta res/layout renombra el fichero activity\_scrolling.xml por activity\_main.xml. En la misma carpeta renombra content\_scrolling.xml por content\_main.xml. Finalmente en la carpeta res/menu renombra menu\_scrolling.xml por menu\_main.xml.



**Paso 3.** Ejecuta el proyecto y verifica su comportamiento:

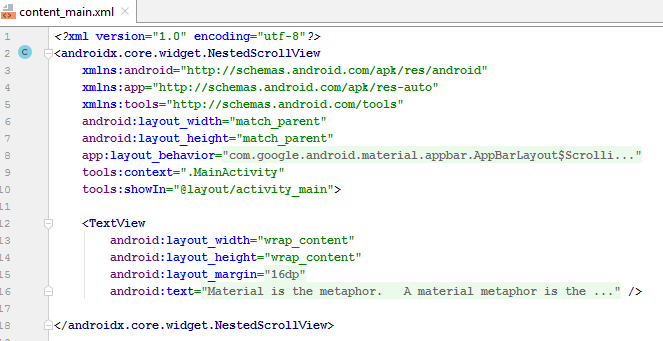


**Paso 4.** Compara los layouts creados para activity\_main.xml de este proyecto y el anterior. En la siguiente figura se muestra un resumen con las nuevas etiquetas y atributos XML que han sido añadidos. No ha de preocuparte si conoces el significado exacto de cada atributo.

|  |  |
| --- | --- |
| Blank Activity  <CoordinatorLayout…>  <AppBarLayout …>  <Toolbar…/>  </AppBarLayout>  <include layout=  "@layout/content\_main"/>  <FloatingActionButton…/>  </CoordinatorLayout> | Scrolling Activity  <CoordinatorLayout…>  <AppBarLayout …  layout\_height="@dimen/app\_bar\_height"  fitsSystemWindows="true">  <CollapsingToolbarLayout…/>  <Toolbar…/>  </CollapsingToolbarLayout>  </AppBarLayout>  <include layout=  @layout/content\_main"/>  <FloatingActionButton  layout\_anchor="@id/app\_bar"/>  </CoordinatorLayout> |

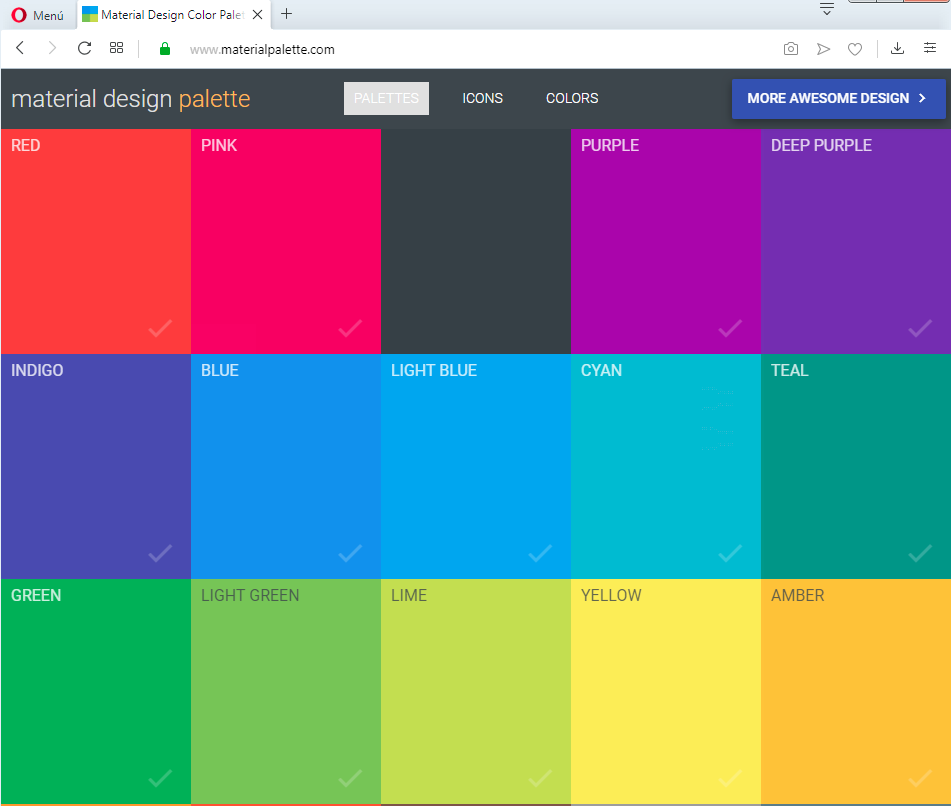
**Nota:** *De momento quédate con los conceptos principales. Si en un futuro quieres modificar este comportamiento básico, puedes consultar la información necesaria.*

**Paso 5.** Compara los layouts creados para content\_main.xml. Observa como ahora se utiliza el layout <NestedScrollView> para conseguir que el contenido sea desplazable. Además, el campo de texto es muy largo para conseguir que no quepa completo en pantalla.

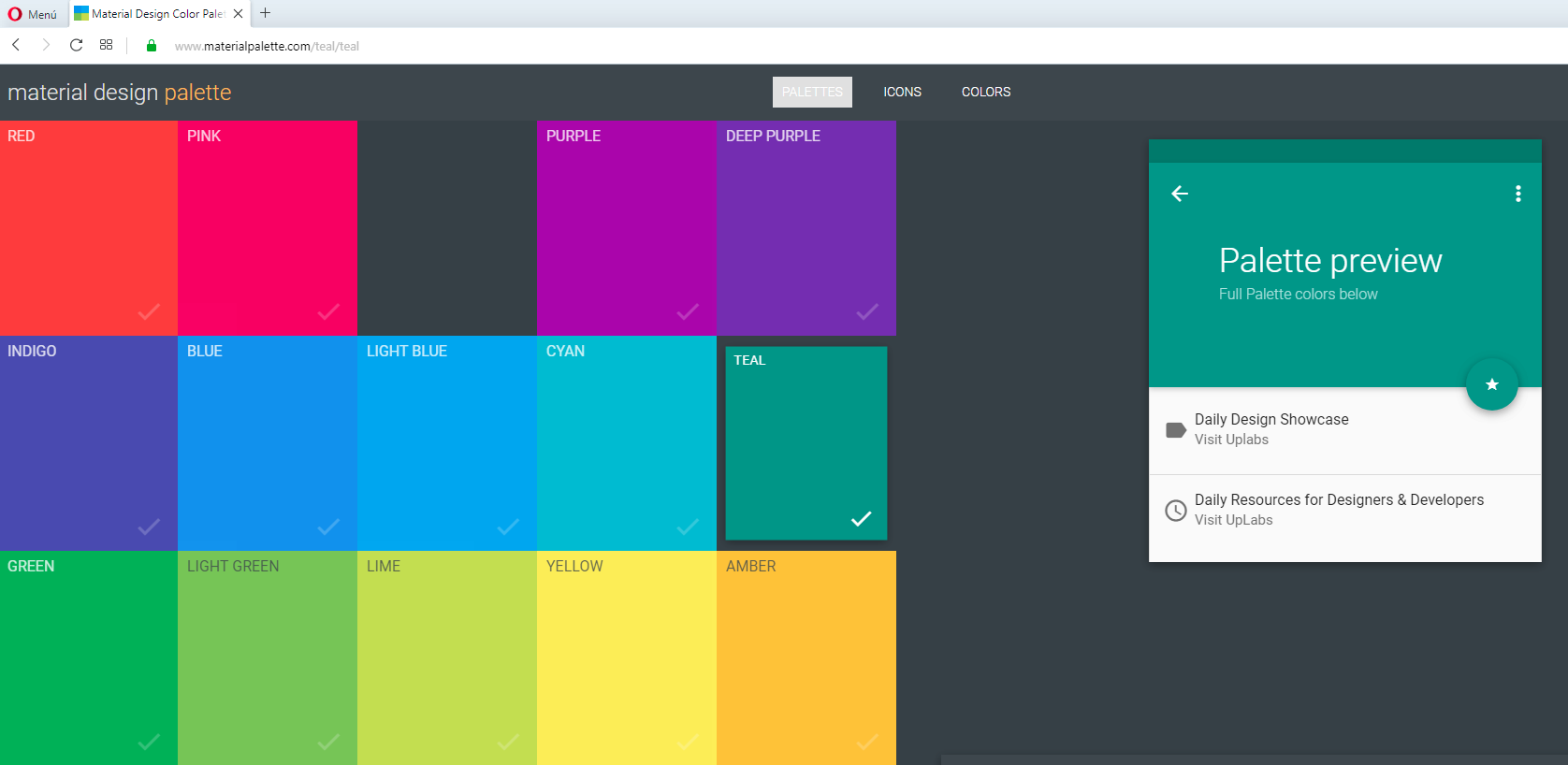


**Parte II: Definir la paleta de colores de la aplicación**

**Paso 1.** Para definir los colores de nuestra aplicación vamos a utilizar la herramienta que encontrarás en: <https://www.materialpalette.com/>



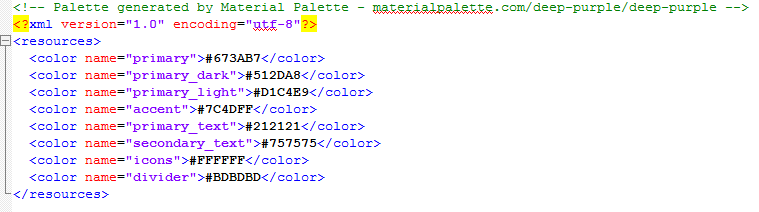
**Paso 2.** Es muy sencilla de manejar, escoge primero el color primario para tu aplicación y a continuación el color de resalte o acentuación. Observa como a la derecha aparece una previsualización de una aplicación que utilizara estos colores. Aunque esta aplicación solo nos permite seleccionar el color primario (primary color) y el de resalte (accent color), realmente puedes configurar hasta 8 colores en la paleta de tu aplicación. Dark primary color y light primary color son colores derivados del primario. El resto se utilizan para textos e iconos y no es recomendable modificar.



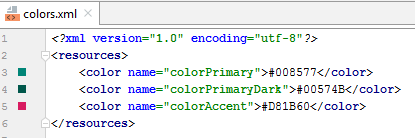
**Paso 3.** Puedes realizar varias pruebas hasta obtener unos colores de tu gusto.

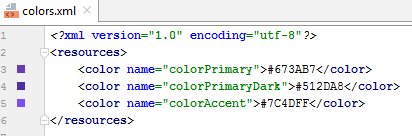
**Paso 4.** Cuando los tengas, puedes pulsar en DOWNLOAD y selecciona XML, para descargarte un fichero de recursos Android donde se definen estos colores.





**Paso 5.** Como solo vamos a cambiar tres valores, va a ser más sencillo que abras el fichero res / values / colors.xml de la aplicación y reemplaces los tres valores definidos, con los valores en hexadecimal que has seleccionado. Estos valores pueden ser diferentes a los que se muestran a continuación:





**Paso 6.** Si abres el fichero res / values / styles.xml podrás observar como estos tres colores son utilizados para configurar los colores del tema aplicado por defecto a tu aplicación:



El resto de colores de la paleta no son definidos dado que se utilizarán los colores por defecto.

**Paso 7.** Si abres el fichero AndroidManifest.xml podrás observar como este tema es asignado a la aplicación.

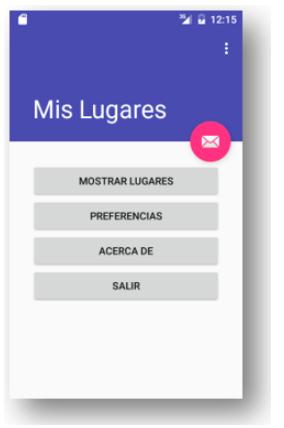


**Paso 8.** Todavía no podremos ejecutar la aplicación. Has de esperar a terminar el próximo apartado.

**Parte III: Creación de una primera actividad en Mis Lugares**

En esta práctica crearemos una primera actividad que contendrá simplemente cuatro botones.

**Paso 1.** Edita el fichero res > layout > content\_main.xmly trata de crear una vista similar a la que ves a continuación. Has de dejar el NestedScrollView que ya tenías y reemplazar el TextView por un ConstraintLayout o LinearLayout que contenga cuatro Button. Un NestedScrollView solo puede contener dentro un elemento, por lo que no puedes introducir directamente los cuatro botones. Usando un layout que los contenga se resuelve el problema.

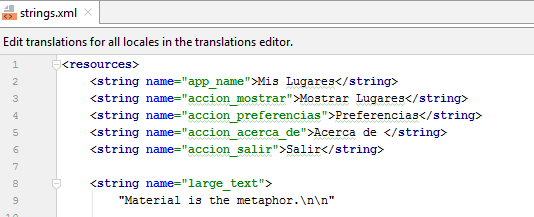


El fichero content\_main.xml ha de ser similar al siguiente (primero con LinerLayout y luego con ConstraintLayout):





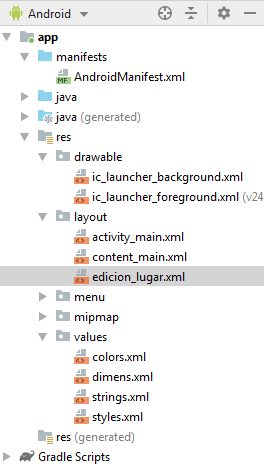
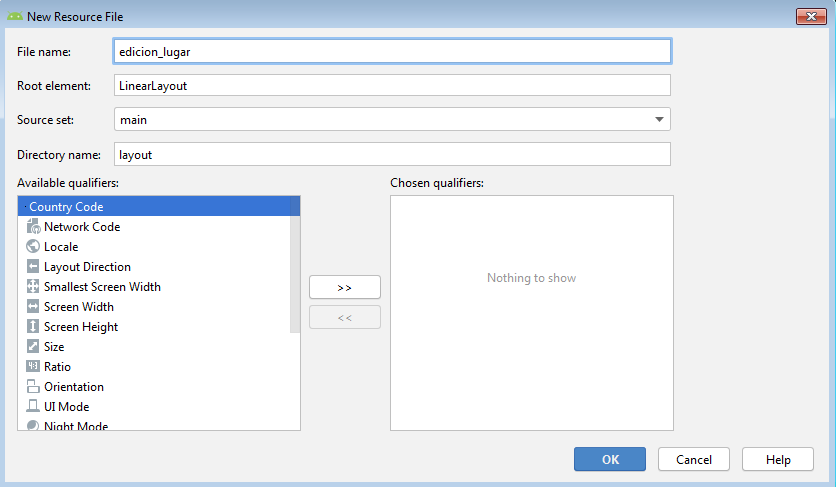
**Paso 2.** El fichero res > values > strings.xml ha de tener el siguiente contenido:



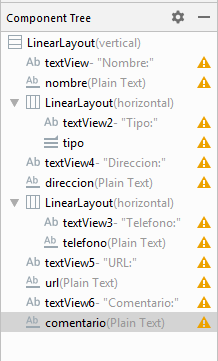
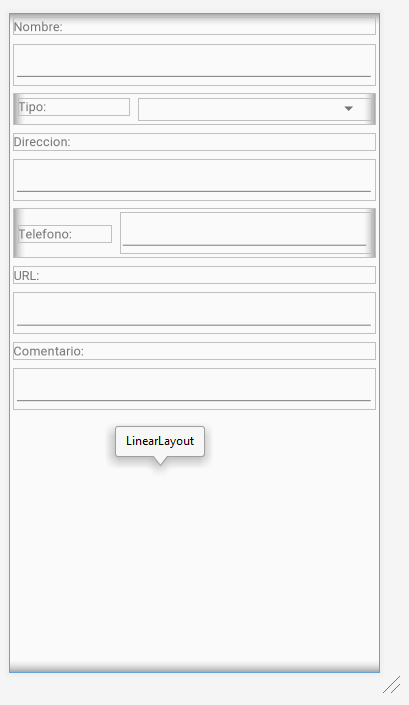
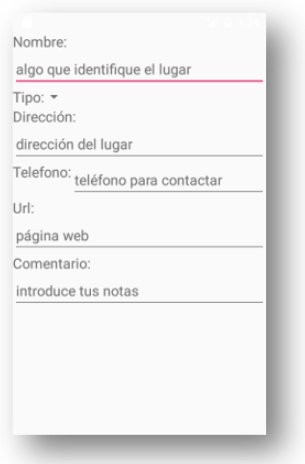
**Parte IV: Un formulario para introducir nuevos lugares**

El objetivo de esta práctica es crear un layout que permita introducir y editar lugares en la aplicación Mis Lugares.

**Paso 1.** Crea un nuevo layout con nombre edicion\_lugar.xml.



**Paso 2.** Ha de parecerse al siguiente formulario. Puedes basarte en un LinearLayout o ConstraintLayout para distribuir los elementos. Pero es importante que este layout, se encuentre dentro de un NestedScrollView para que cuando el formulario no quepa en pantalla se pueda desplazar verticalmente.

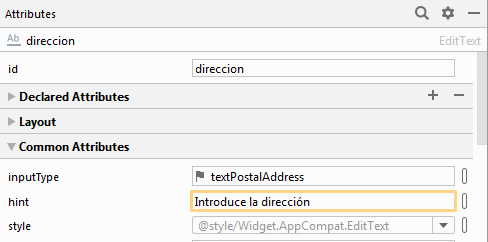
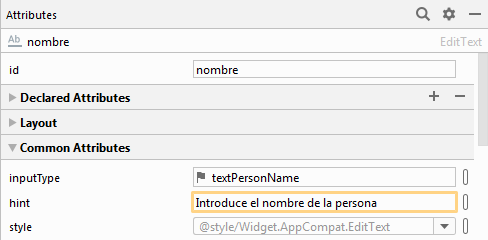


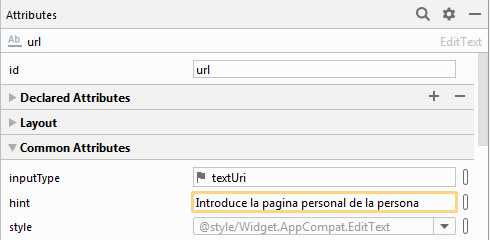
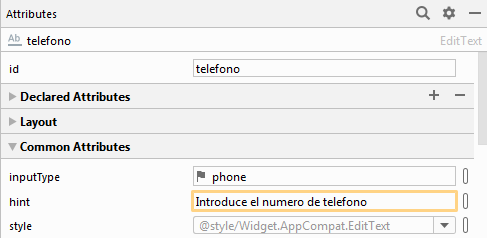
**Paso 3.** Introduce a la derecha del TextView con texto “Tipo:” un Spinner con id tipo. Más adelante configuraremos esta vista para que muestre un desplegable con los tipos de lugares.

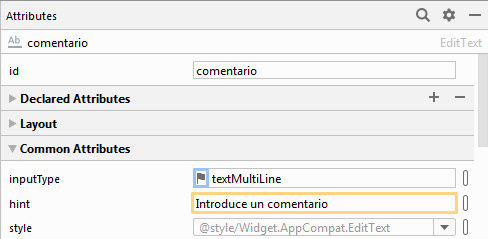
**Paso 4.** Las vistas EditText han de definir el atributo id con los valores: nombre, direccion, telefono, url y comentario. Utiliza también el atributo hint para dar indicaciones sobre el valor a introducir. Utiliza el atributo inputType para indicar qué tipo de entrada esperamos. De esta manera se mostrará un teclado adecuado (por ejemplo si introducimos un correo electrónico aparecerá la tecla @).

Nota: El atributo inputType admite los siguientes valores (en negrita los que has de utilizar en este ejercicio): none, text, textCapCharacters, textCapWords, textCapSentences, textAutoCorrect, textAutoComplete, **textMultiLine**, textImeMultiLine, textNoSuggestions, **textUri**, textEmailAddress, textEmailSubject, textShortMessage, textLongMessage, textPersonName, **textPostalAddress**, textPassword, textVisiblePassword, textWebEditText, textFilter, textPhonetic, textWebEmailAddress, textWebPassword, number, numberSigned, numberDecimal, numberPassword, **phone**, datetime, date y time.

* Nombre: inputType= textPersonName
* Direccion: inputType= textPostalAddress
* Telefono: inputType= phone
* URL: inputType= textUri
* Comentario: inputType= textMultiLine



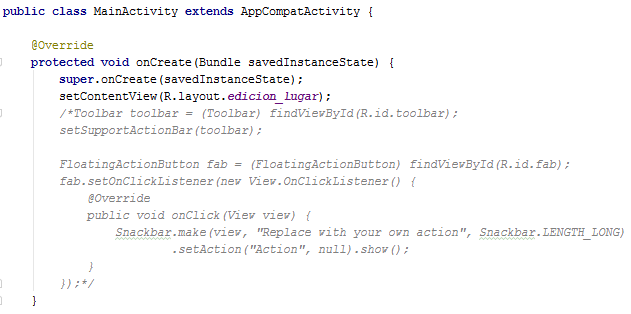




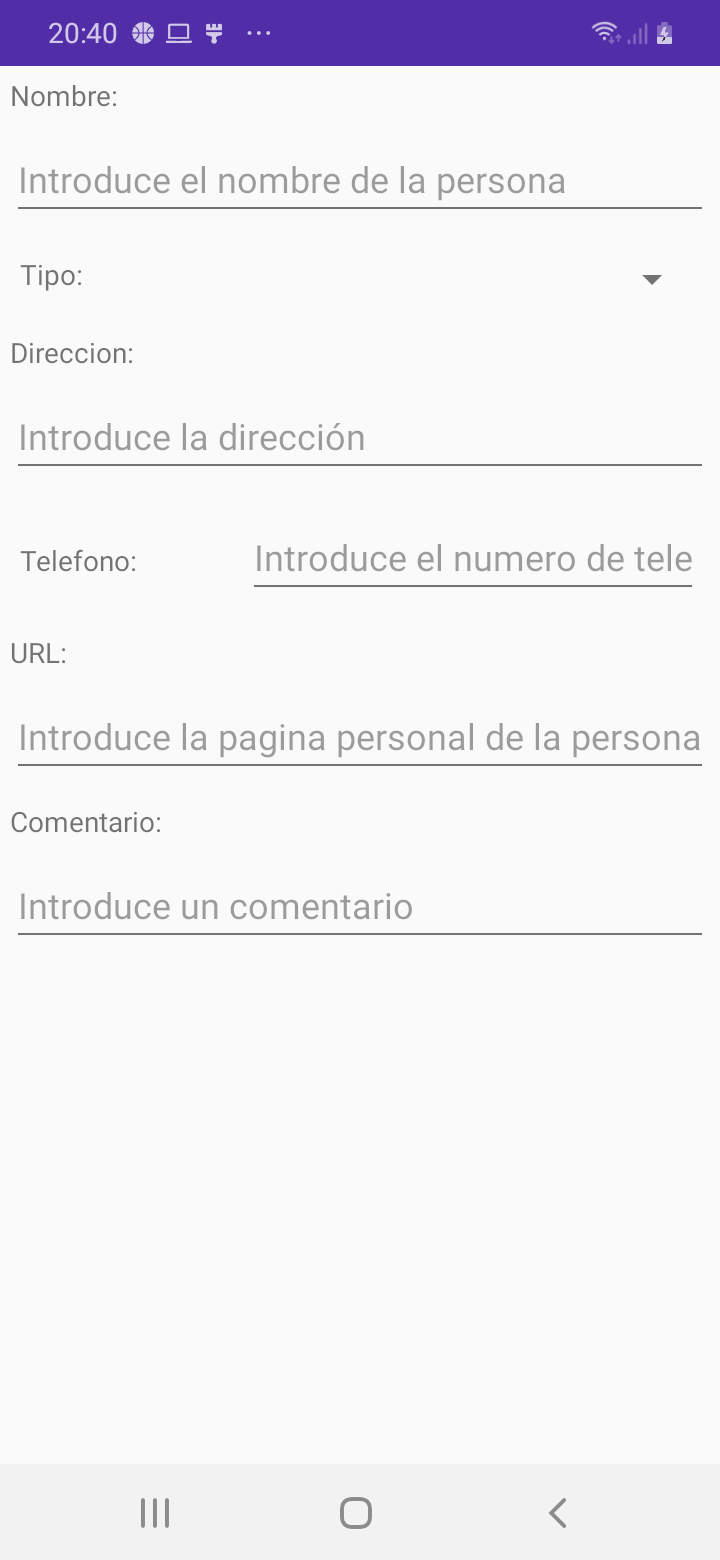
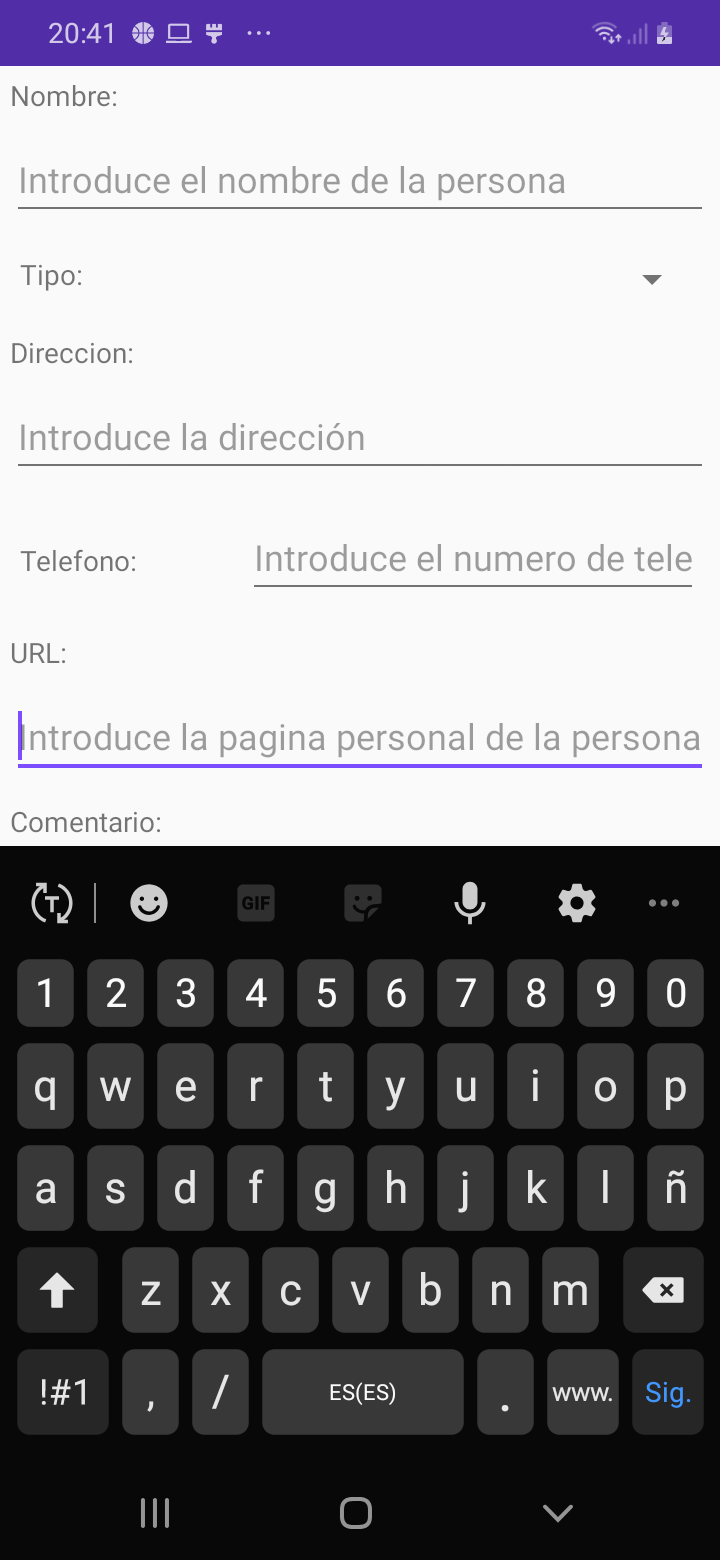
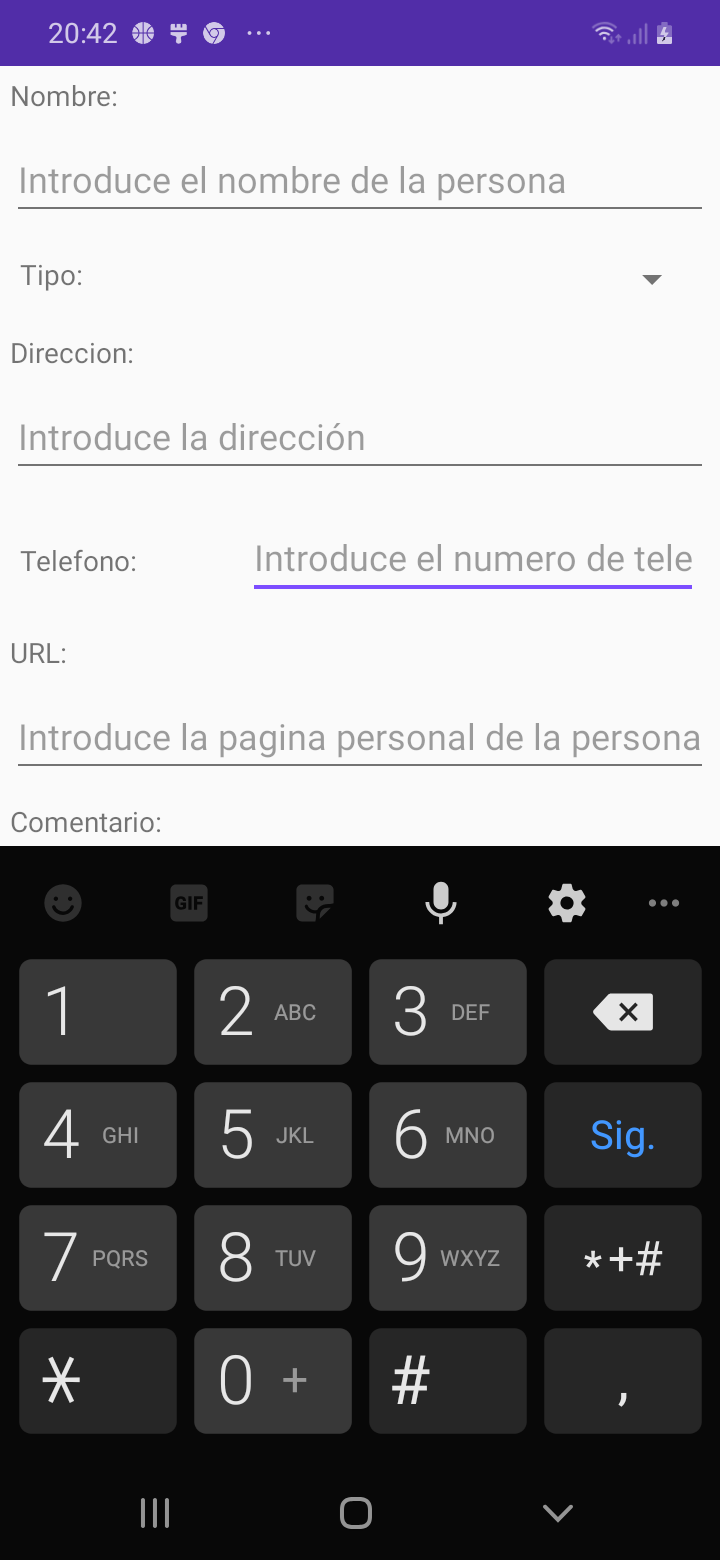
**Paso 5.** Abre la clase MainActivity y en el método onCreate() reemplaza:

setContentView(R.layout.edicion\_lugar);

**Paso 6.** Comenta todas la líneas de este método que hay debajo usando /\*... \*/. Como ya no se crea el layout activity\_main los id de vista a los que se accede ya no existen.



**Paso 7.** Ejecuta la aplicación y verifica como cambia el tipo de teclado en cada EditText

**Paso 8.** Deshaz el cambio realizado en el punto 5 y 6.

