## Graphical user interface Description automatically generated with low confidenceExercise 2.1: Create Recipe (Basic Layout)

You will create a new activity that allows you to add new recipe to your recipe book. To switch to the new activity, try to edit *AndroidManifest.xml,* such that the main *Activity* is deactivated, and you can instead see the create recipe form. We will fix this problem in an upcoming exercise.

The user interface might look like shown in the picture on the right. The layout consists of the following components:

ConstraintLayout as base layout of the *Activity* with the following rows (see example on the right). You can wrap the ConstraintLayout with a ScrollView, as most likely the entire Activity won’t fit on the screen:

* EditText (inputType=“text“) to input the dish name
* Spinner with selection of country of origin
* EditText (inputType=“ textMultiLine “) to input the ingredients
* EditText (inputType=“ textMultiLine “) to input the instructions
* Button „Save“ triggering data validation (check if the EditText fields are empty and set error text if so) and adding the new recipe to the repository

Create a new project with one *Activity* and design its layout!

Change between the Design- and XML-view and look up the documentation[[1]](#footnote-1) when unsure. Especially take care about the layout properties of ConstraintLayout.

For this exercise the layout should only be shown. The selection box (*Spinner*) for country of origin can remain empty. The “Save” button is inoperable.

Feel free to experiment with the design or create something looking completely different!

## Exercise 2.2: Create Recipe (List of Countries)

Use the list of countries from the Materials folder to populate the country Spinner:

* Create an appropriate ArrayAdapter[[2]](#footnote-2)
* Set it for the Spinner-Views (see lecture)

Your app should now be able to select a country of origin for your recipe.

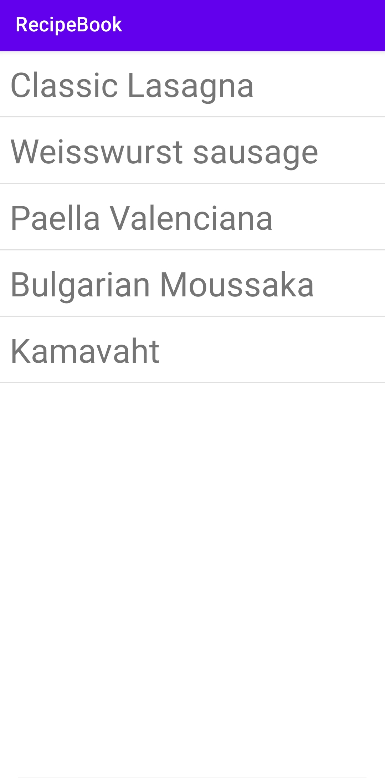
## Exercise 2.3: Create Recipe (Functionality)

Create a method for the RecipeRepository that allows you to add a new recipe. Register an onClick handler with the button. Check if the EditTexts are empty and if so, set the error text to the corresponding field. If everything is valid, take the values from the text fields and the spinner, use them to create a new recipe object and add it to the repository. For now, it won’t be possible to see if the new object was added successfully, but you can clear the EditTexts and reset the Spinner to the default value for confirmation.

Remarks:

* Single layout views can be found using findViewById(R.id.id\_of\_view).
* The selected item of a spinner can be obtained with getSelectedItem().
* The error text can be set using the method setError(yourText).

## Exercise 2.4: Recipe List (New Activity)

Create an additional *Activity* with a list of all recipes (example on the right).

Create a new empty *Activity* for your project (File 🡪 New... 🡪 Activity 🡪 Empty Activity):

* Enter a good name for the *Activity* (f.e. *„RecipeListActivity“)*
* Select the option *„Launcher Activity“*

Define a fitting layout for the ListView and its entries.

Create an ArrayAdapter that supplies the view with its entries.

When running the app the launcher might show two icons, depending on what you selected when creating the *Activity*. However, each icon will start the original *Activity*.

1. f.e. <https://developer.android.com/guide/topics/ui/declaring-layout.html> and

   <https://developer.android.com/guide/topics/ui/controls/spinner.html> [↑](#footnote-ref-1)
2. <https://developer.android.com/reference/android/widget/ArrayAdapter.html> [↑](#footnote-ref-2)