## Exercise 2.1: Save Recipe (Basic Layout)

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

* Set the name to *SaveRecipeActivity*
* Select the option *Launcher Activity*

This activity will allow you to add new recipes to your recipe book.

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. To switch to the new activity, try to edit AndroidManifest.xml*,* such that the RecipeDetailsActivity is deactivated, and you can instead see the recipe form. We will fix this problem in an upcoming exercise.

The layout consists of the following components:

A constraint layout as base layout of the activity with the following rows (see example on the right). You can wrap the constraint layout with a scroll view, as most likely the entire activity won’t fit on the screen:

* EditText (*inputType=“text“)* to input the recipe name
* Spinner with selection of country of origin
* EditText (*inputType=“ textMultiLine* *“*) to input the ingredients
* EditText (*inputType=“ textMultiLine* *“*) to input the instructions
* A save button that triggers data validation (check if the EditText fields are empty and set error text if so) and adds the new recipe to the instance of RecipeCollection

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 layout design.

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

Use the class *CountryCollection* from the Materials folder to populate the country spinner:

* Create an appropriate ArrayAdapter[[1]](#footnote-1)
* Set it for the Spinner-Views (see lecture) using the *getCountries()* method

For sake of simplicity, we are going to use only European countries. Don’t worry about the rest of the code inside the class. It will be explained in a later.

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

## Exercise 2.3: Save Recipe (Functionality)

Add a getter method for the List<Recipe> inside the RecipeCollection class, which will allow you to access the object and add new recipes to it. Register an *onClick()* handler for the save button. Check if any of the EditTexts fields are empty and if so, set an appropriate error text message to the corresponding field. If everything is valid, take the values from the text fields and the spinner, and use them to create a new recipe object and add it to the recipe list. For now, it won’t be possible to see the objects inside the list. Log a message to confirm that the recipe was created successfully.

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 message can be set using the method *setError().*

## Exercise 2.4: Recipe List (New Activity)

Create an additional activity with a list view to show all the recipes. Name it *RecipeListActivity*. Define a fitting layout for the list view and its entries. Set it as the launcher activity*,* like in Exercise 2.1

Create an array adapter that supplies the view with its entries. Extract the recipe names from the collection to a List andset it to the adapter.

1. <https://developer.android.com/reference/android/widget/ArrayAdapter.html> [↑](#footnote-ref-1)