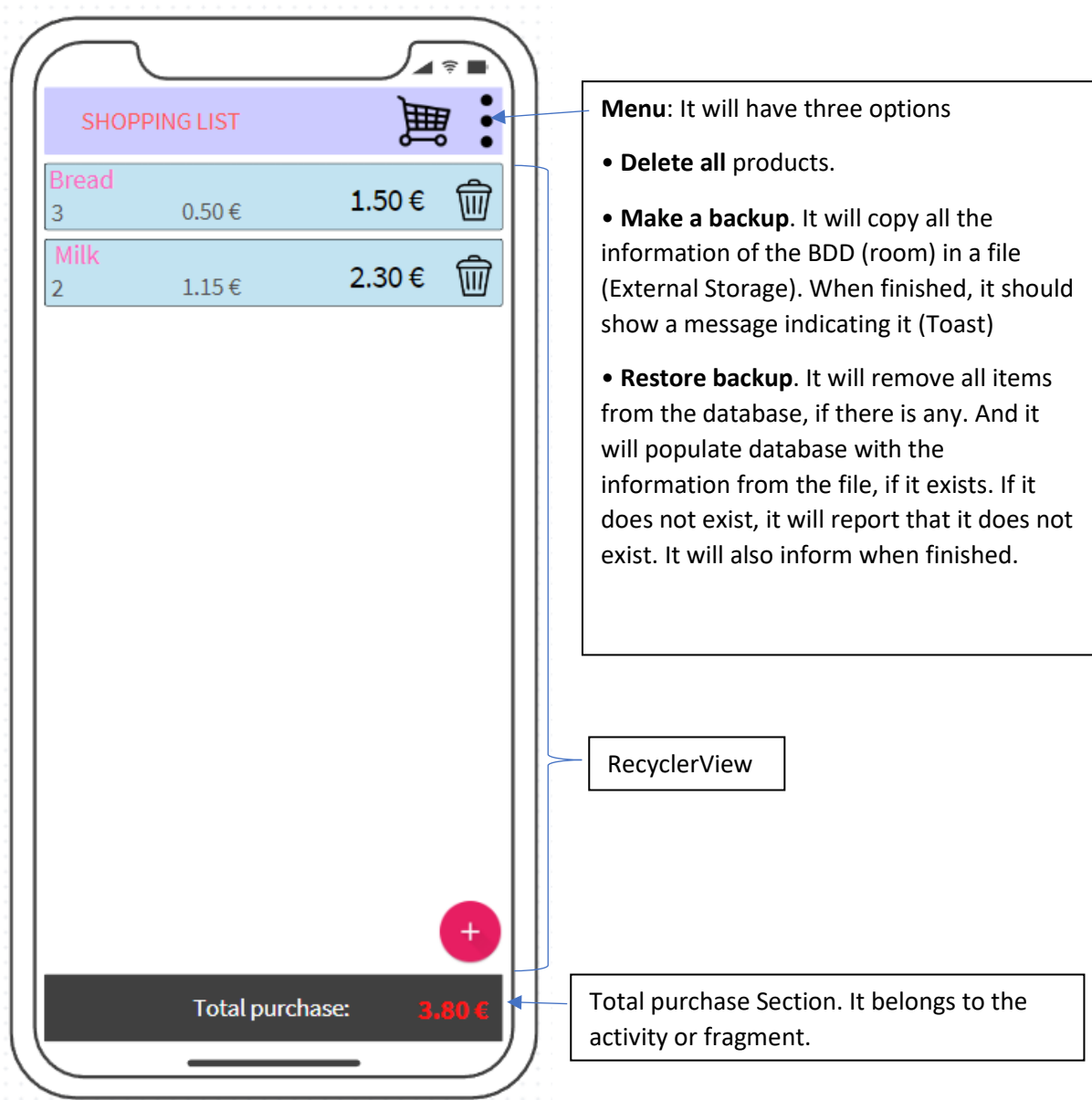# PRACTICE 5: SHOPPING LIST

We are going to create an application that serves to make the shopping list. This app will look similar to this (colors can be customized).

Menu: It will have three options

• **Delete all** products.

• **Make a backup**. It will copy all the information of the BDD (room) in a file (External Storage). When finished, it should show a message indicating it (Toast)

• **Restore backup**. It will remove all items from the database, if there is any. And it will populate database with the information from the file, if it exists. If it does not exist, it will report that it does not exist. It will also inform when finished.

RecyclerView

Total purchase Section. It belongs to the activity or fragment.

It must be implemented:

- A **RecyclerView** where the purchase items will be added.
- A **Total purchase section** where you will see the total amount of the purchase. (Add the value of each item multiplied by its quantity). This is part of the activity or fragment.

- **FloatingAction button**. Pressing it will take us to another Activity, fragment, dialogue where we can create a new article, this will be saved in a database with Room and of course the list on the screen will be updated.
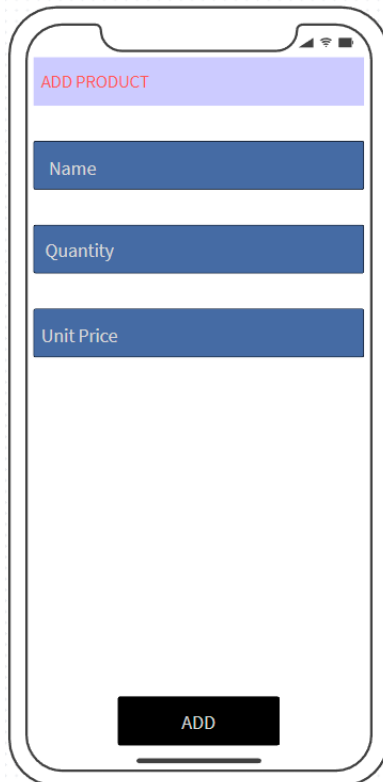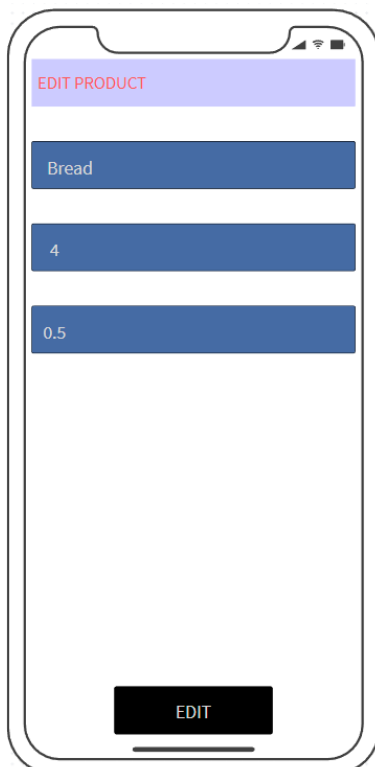
ADD PRODUCT Name Quantity Unit Price ADD	This could be a dialogue, an activity or a fragment **TextEdits**: Product, will be text type, maximum 10 characters • Quantity: it Will be a possitive Int • Unit Price: it will be a decimal number with 2 decimal digits. • The help text (hint) should appear

• When **clicking on the trash**, the quantity will decrease by one, if it reaches zero the article will be removed.

• When **clicking on any part** of the item in the list (except the trash), another Activity, fragment, dialog will be displayed where we can edit the item.

EDIT PRODUCT Bread 4 0.5 EDIT	This could be a dialogue, an activity or a fragment **TextEdits**: Product, will be text type, maximum 10 characters • Quantity: it Will be a positive Int • Unit Price: it will be a decimal number with 2 decimal digits. • The help text (hint) should appear
	You must store the data in a database using room and also, you must implement the DAO pattern and ViewModel pattern using LiveData and of course, using coroutines. When working with files try use Flows. Optative.