

CSC 372/472 Mobile Application Development for Android

Programming Assignment 3 A Simple Point of Sales App

Due Date

- Assignment due on Wednesday October 4, 2017, 11:59pm
- Submit your assignment in D2L
- Submit a single zip file that contains the contents of the project folder
- You must use a unique prefix for the project name. (I suggest you use your last name and first initial as your prefix.) **Please use the same prefix for all your assignments.**
- Do not include unused or unrelated files.
- Before you submit, build and run the project, make sure everything compiles and works. Close your project before zipping the folder.
- Here are the most common reasons assignments are marked down:
 - Project does not build.
 - Project does not build without warnings.
 - One or more items in the Requirements section were not satisfied.
 - A fundamental concept was not understood.
 - Code is sloppy and hard to read (e.g. indentation is not consistent, etc.).
 - Your solution is difficult (or impossible) for someone reading the code to understand due to lack of comments, poor variable/method names, poor solution structure, etc.
- Bonus points.
 - Bonus points may be awarded to projects with *exceptional* qualities in one or more aspects.
 - Bonus points will only be awarded after all the required elements have been satisfied.
 - Bonus points will not be awarded merely for extra amount of work (or code).
 - Bonus points are awarded at the sole discretion of the instructor.

Goals

- Create a simple app with a single activity
- Using Linear Layout to create a functional UI
- Using text input field and handle events related to text input

Requirements

Create a simple *Point of Sales* (POS) app for ordering food items in a restaurant. The app allows one to enter the orders by customers.

Functional Requirements

- The initial UI of the app should be similar to the one shown in Figure 1. The default values for the input fields are shown.
- The “New Order” button starts a new order from a customer. Items will be entered one at a time.
- The items can be from a predefined menu (with name and price for each item) or a special item that is not on the menu. The text field for entering item names should support auto-complete for items on the menu.
- If the item is from the menu, the keyboard action button for item name will fill in the unit price of the item (Figure 2). If the item is not from the menu, the unit price will remain 0.00, and it needs to be entered manually.
- The default value for quantity is 1. It can be changed to another integer value.
- Press the “Total” button after each item is confirmed. It will update the total amount for all items in this order and update the text view at the bottom of the screen to show a summary of the order (Figure 3). Figure 3 shows the screen after two items have been entered. The order summary lists each item in the order and the quantity.
- The “New Item” button will reset the default values for all text input fields, and prepare for entering another item in the order. Press “New Item” before “Total” will cancel the current item.

UI Design Requirements

- Note that the rows other than the one for item names and the order summary are evenly divided into 2

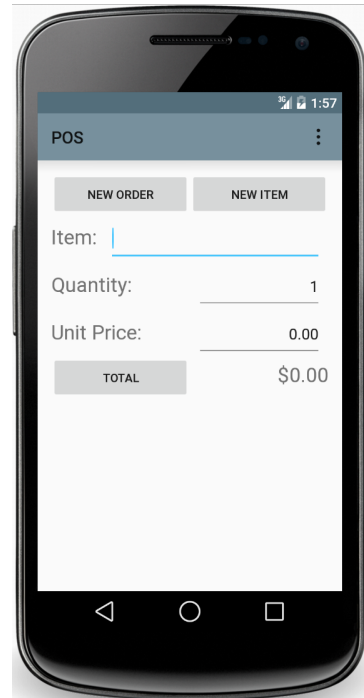


Figure 1

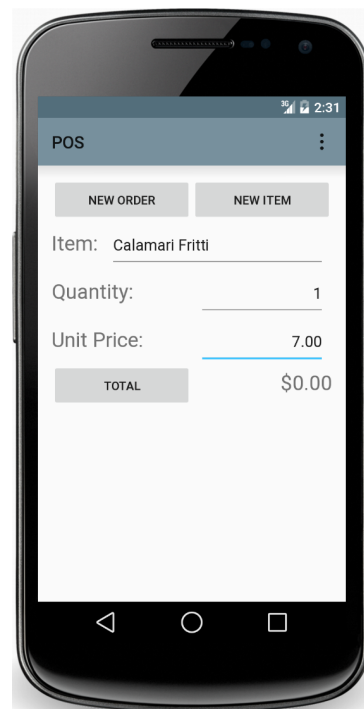


Figure 2

columns.

- Each text input field should bring up a soft keyboard that is appropriate for the values of the field.
- All text input fields should be restricted to single line.
- Create the UI using Linear Layout only.

Testing Your App

To test your app you should create a predefined menu with 5 or more items. Test at least the following cases:

- Ordering more than one items
- Ordering items both on the menu and off the menu
- Ordering items with different quantities

Make sure your app functions as described, and the order summary and total amount are accurate.

Your app is only required to handle the normal use cases. In other words, robustness is not a requirement for this assignment.

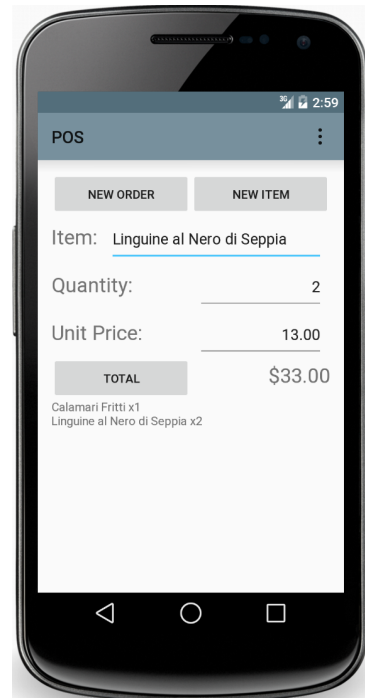


Figure 3