

# MAPAssignment #1

## Cash Register Application



### Objectives:

- Use Xamarin Forms to create both cross platform app for cash register;
- Use Labels, Buttons and other UI components;
- Use Stack Layout, Grid and List view layouts;
- Create Handlers and manage events;
- Create models (add, delete and update) to manage application data;
- Handel incorrect values.
- Multiple page navigation.

# Cash Register Application (Part 1)

Part 1:

UI

- Grid layout
- 9 digits Buttons
- 1 List View
- 3 Labels
- 1 Buy Button

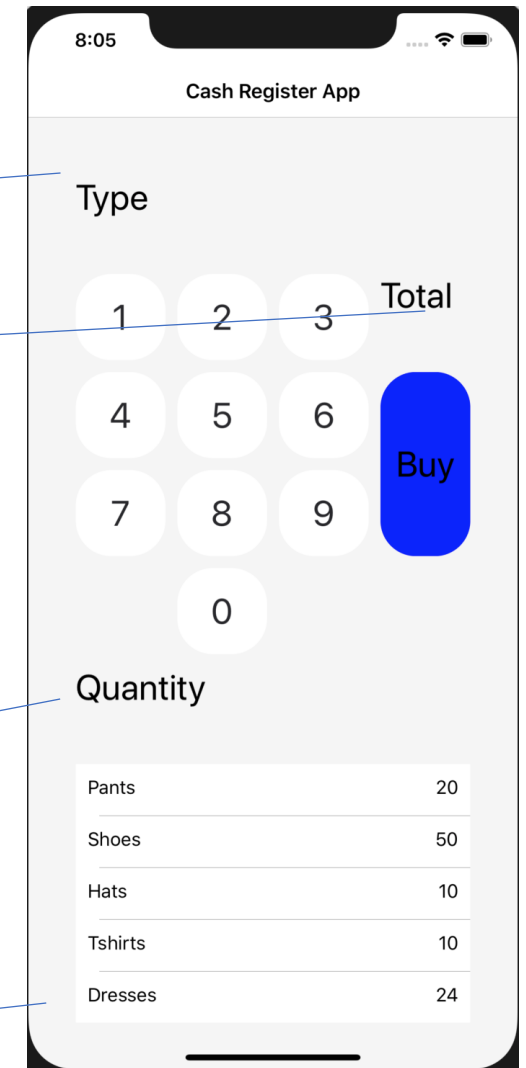
Buy the amount that the user entered

Each Product has  
name, quantity and price  
attributes.

Total amount Label

Quantity Label

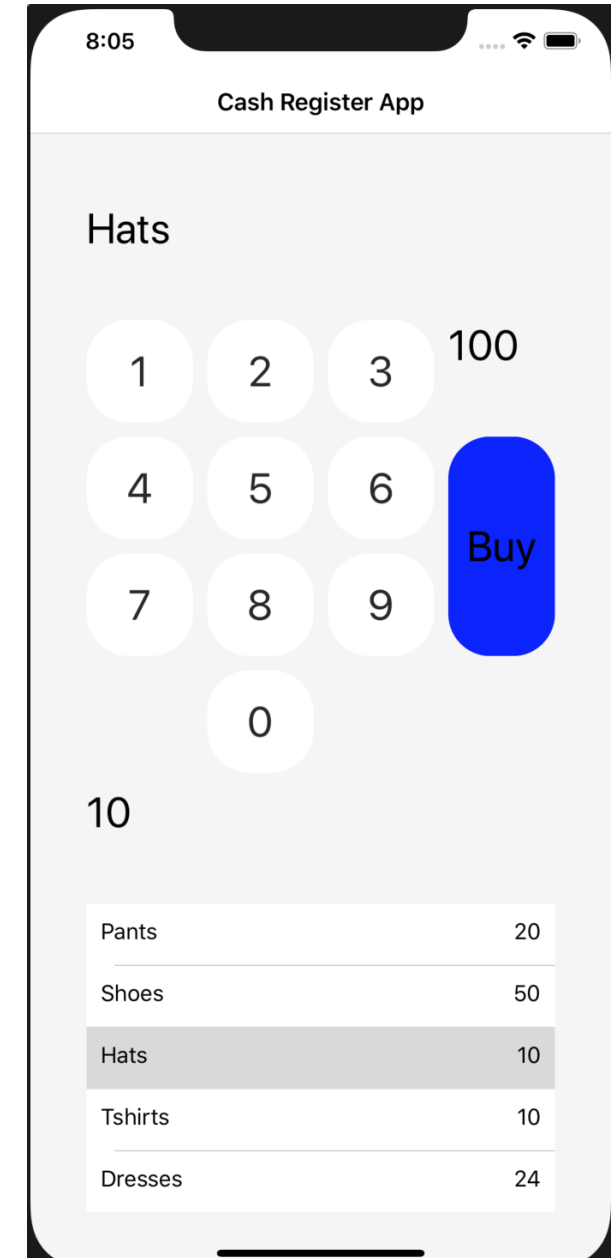
Products Names, Quantity  
and Price List View



# Cash Register Application (Part 1)

Interactive with UI:

- When the user selects one product from the list, type label must be updated to have the product name;
- When the user selects the quantity from the numbers pad, the quantity label must be updated to have the selected quantity;
- Total label should show the total price of that purchase
- $\text{Total} = \text{amount} * \text{item price}$

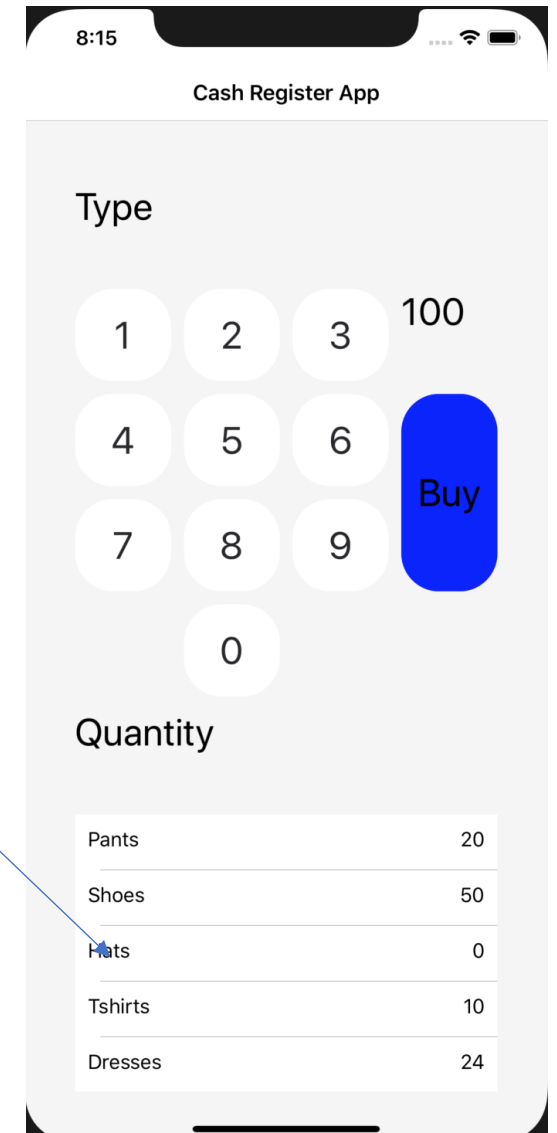


# Cash Register Application (Part 1)

When the user clicks on Buy Button the quantity of the purchased item should be updated and the user interface should be reset to accept a new purchase.

New quantity = old quantity – purchased amount

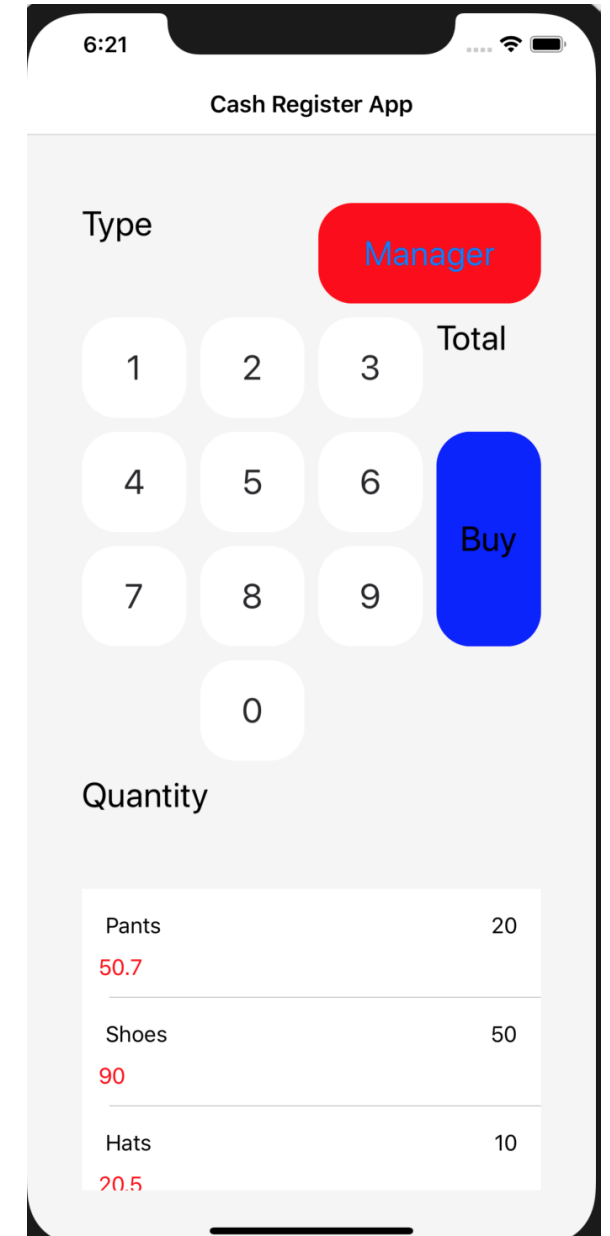
If the user selects a quantity which is more than the available quantity in the stock, an alert popped up with an error message.



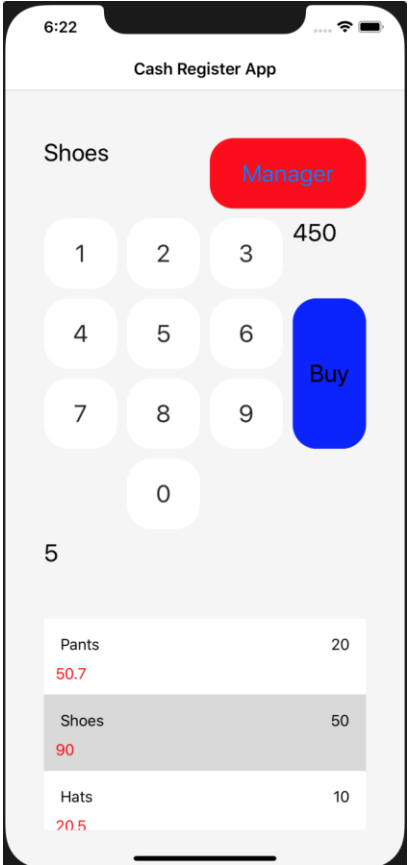
# Cash Register Application (Part 2)

- 1 Take Cash Register From Part 1
2. Rearrange as you wish so all these Components would fit
3. Add a manager Button

When manager button is clicked The next page (Manager Panel) will appear.

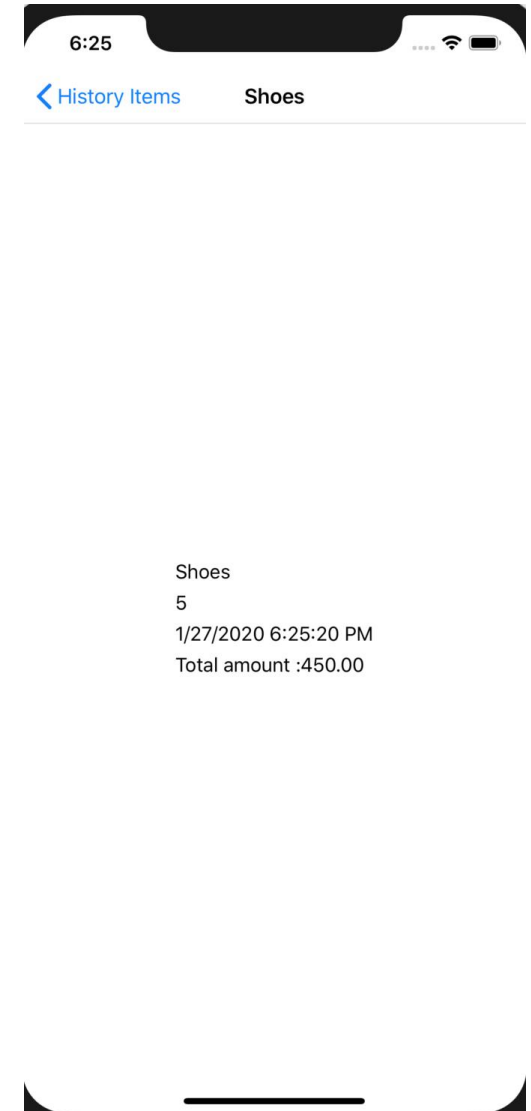


The Manager page has two buttons History and Restock



# History

- History Page list all purchased products with quantities.
- When the user selects one item the app navigates to another page with more details about that purchase.
- **Note: You will need to add the model class To be able to obtain some sort of history Object and show it here.**
- Your history class (object) should have these properties(Name, Quantity, Total price, purchase date).



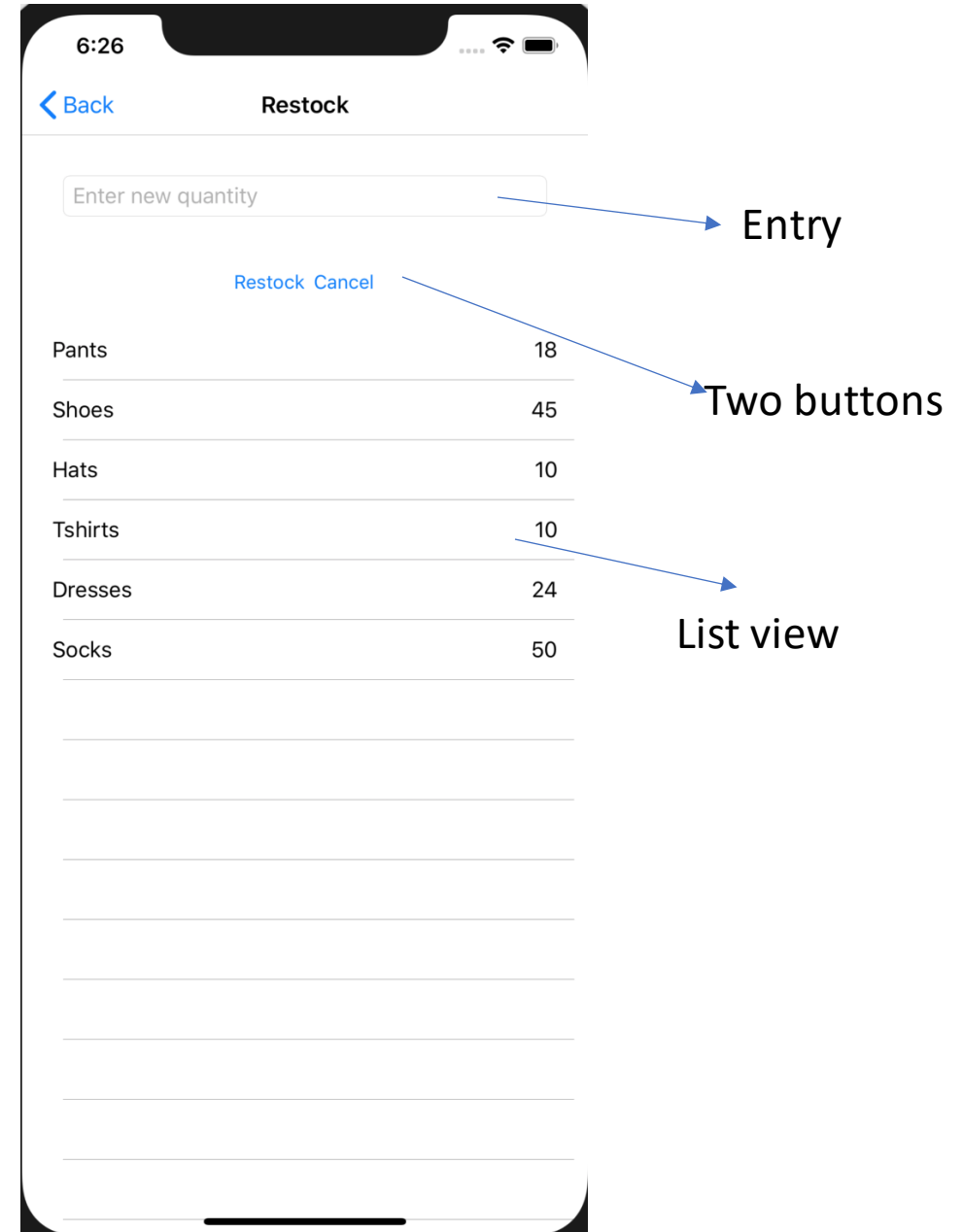
# Restock

Will show everything the Store has In the stock

When the manager selects a product and enter a new quantity in entry view, by Pressing on Restock button, the Item's Quantity will be added with the new amount.

List view will get updated instantly.

Cancel button will navigate back to manager panel.

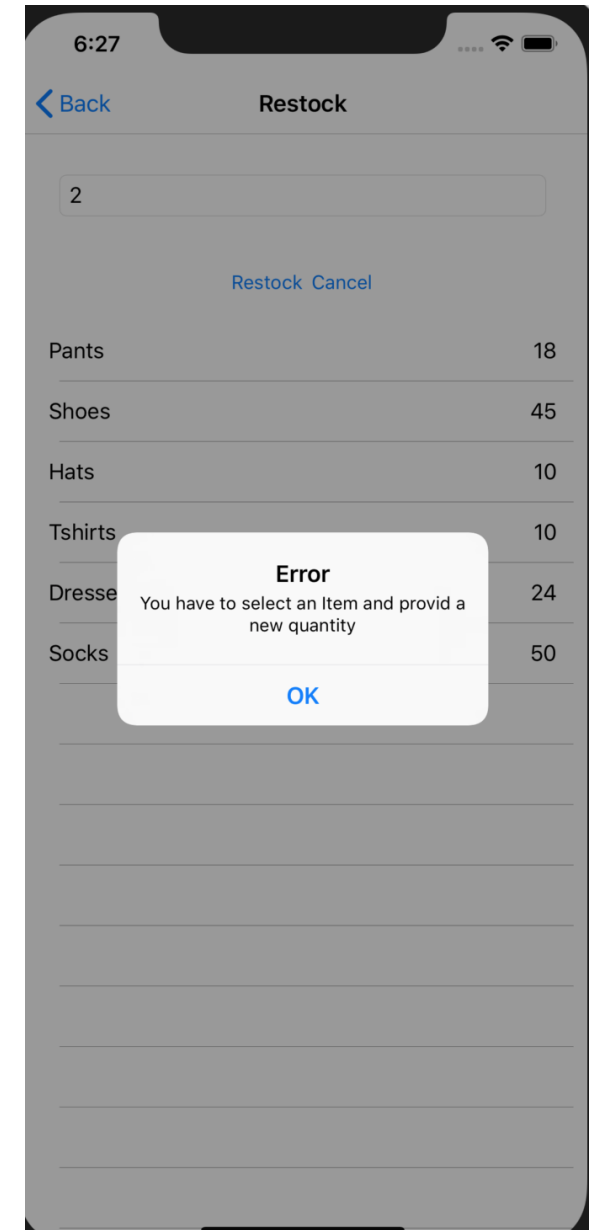
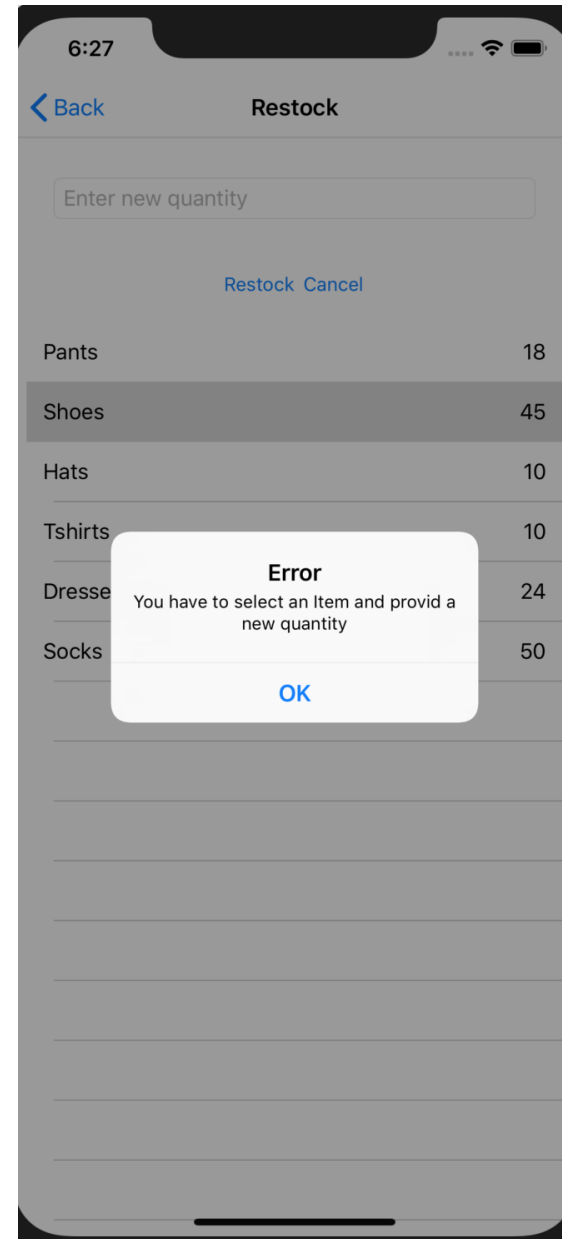






# Handel misusing

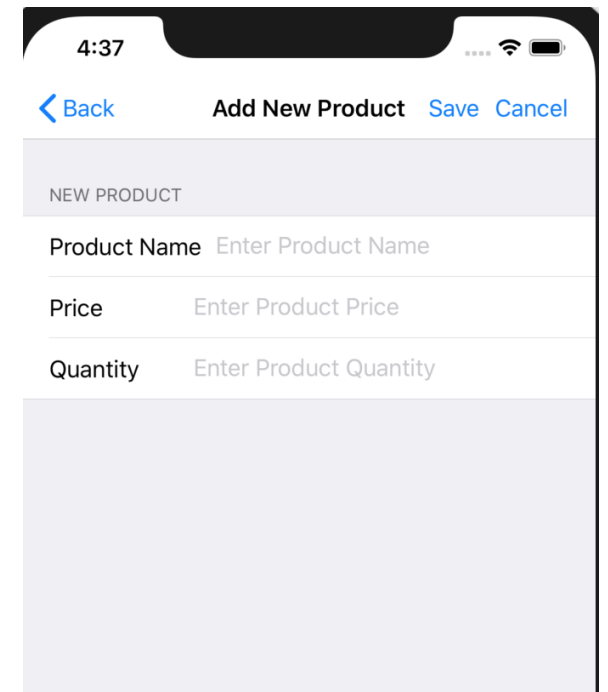
If the user selects an item from the list without entering a new quantity OR if the user enters a new quantity without selecting an item from the list, the app should display an alert.



# Additional functionality (Optional)

Add a new functionality to manager panel, where the manager can add new product.

Add New Product page contains Table View with three Entry Cells and two bar items to save or cancel.



# Additional functionality (Optional)

4:30

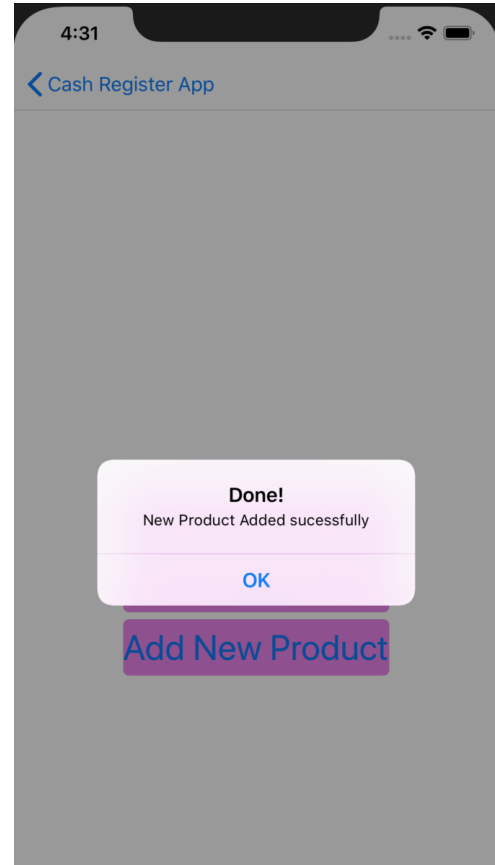
< Back Save Cancel

NEW PRODUCT

Product Name Sweater

Price 99.9

Quantity 50



4:31

Cash Register App

Type

Manager

Total

1 2 3

4 5 6

7 8 9

0

Buy

Quantity

Dresses	24
140.3	
Socks	50
10.3	
Sweater	50
99.9	

Good Luck