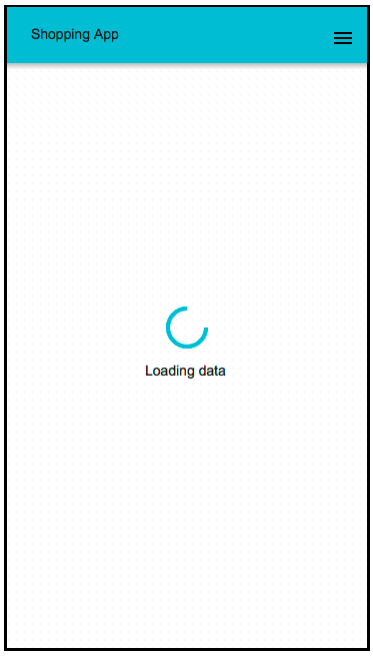
ITIS/ITCS 4180/5180 Mobile Application Development
Midterm

Basic Instructions:

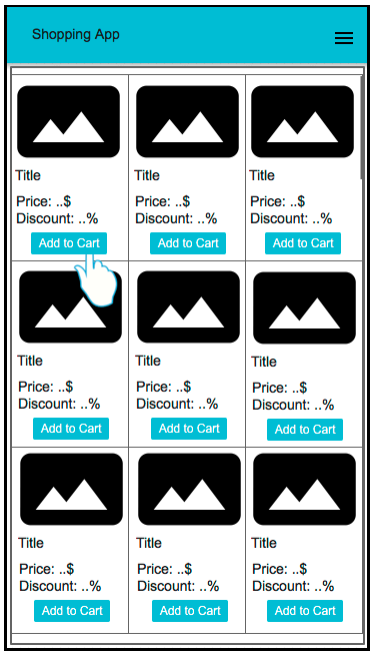
1. This is the Midterm Exam, which will count for 20% of the total course grade.
2. In every file submitted you MUST place the following comments:
   a. Your Full Name.
3. This Midterm is an individual effort. Each student is responsible for her/his own Midterm and its submission.
4. Once you have picked up the exam, you may not discuss it in any way with anyone until the exam period is over.
5. During the exam, you are allowed to use the course videos, slides, and your code from previous home works and in class assignments. You are NOT allowed to use code provided by other students or from other sources.
6. Answer all the exam parts, all the parts are required.
7. Please download the support files provided with the Midterm and use them when implementing your project.
8. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will loose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
9. Export your Android project and create a zip file which includes all the project folder and any required libraries. The file name is very important and should follow the following format: **800#_Midterm.zip.** Submit the exported file using the provided canvas submission link.
10. **Failure to do the above instructions will result in loss of points.**
11. **Any violation of the rules regarding consultation with others will not be tolerated and will result disciplinary action and failing the course.**

Midterm (100 points)

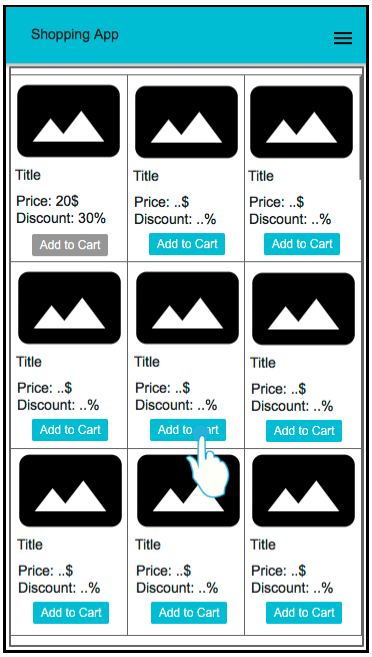
In this assignment, we will be developing a simple shopping application, which displays a list of products. The user can add any product to the cart, and checkout. The user can even see the previous orders they make. The cart and the history are stored in ArrayList and SQLite respectively.



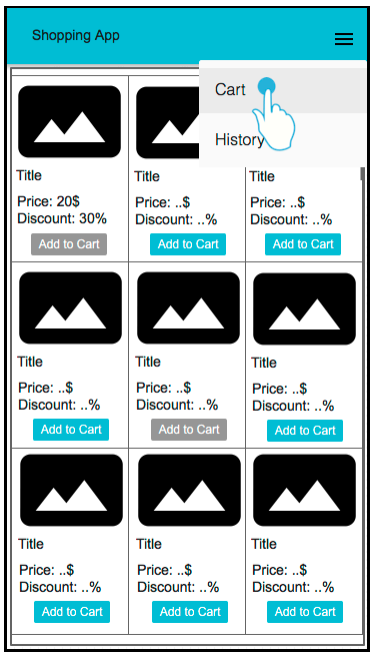
(a) Parsing Data



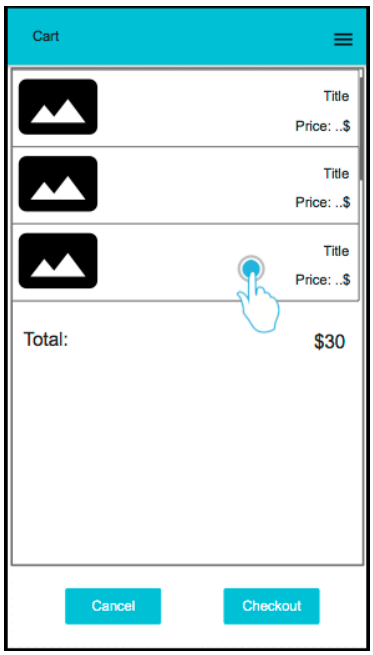
(b) Product List (Recycler View, Grid Layout)



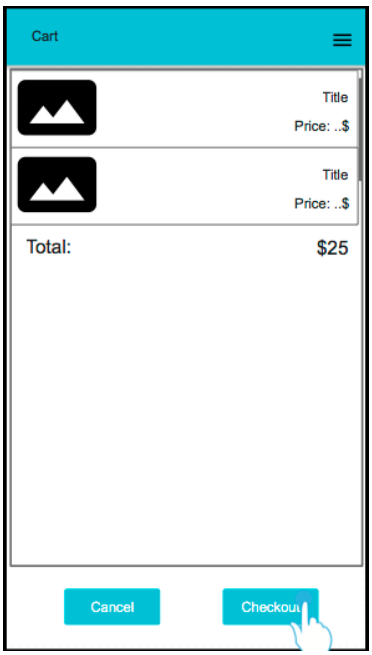
(c) Added to Cart, button disabled



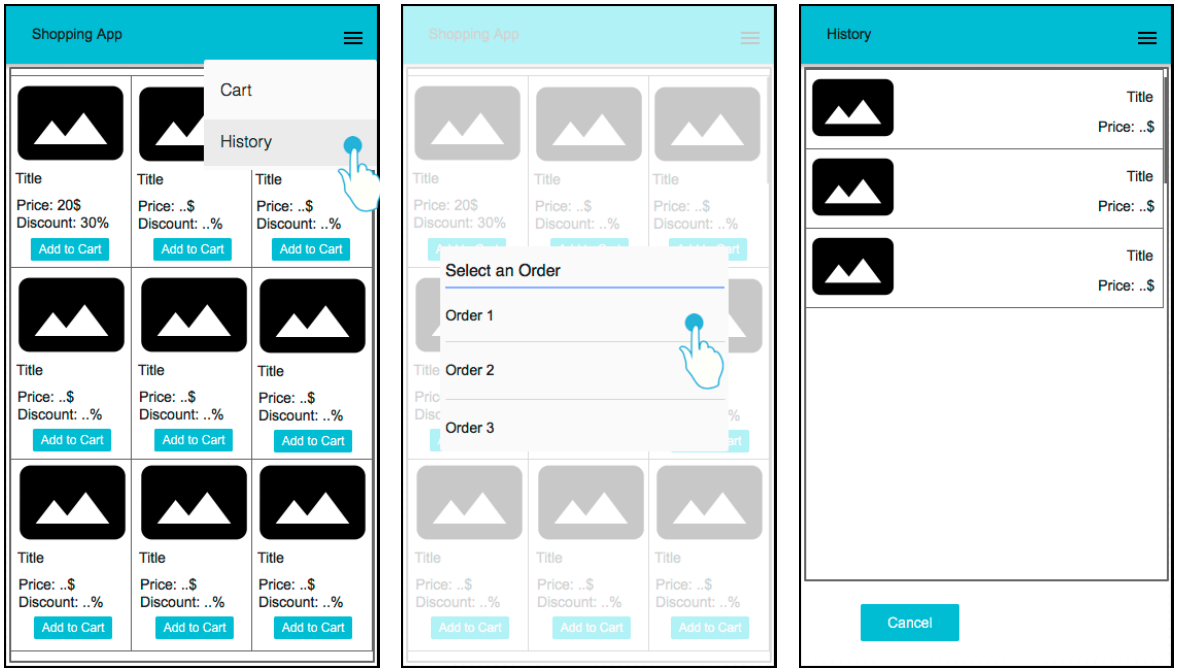
(d) Menu options, Click to view cart



(e) Cart, Long click to delete from cart



(f) Checkout current cart.



(g) Menu: History (h) Click on a Previous Order (i) Display the selected order

Figure 1: App Wireframes

.

This assignment consists of three screens:

1. Main Screen
2. Cart Screen
3. History Screen

Use either Activity or Fragment to implement the screens.

JSON API: We will be using a custom JSON API for the midterm. The API URL is: **http://52.90.79.130:8080/MidTerm/get/products** . Please read the following Table for API details:

API URL:	http://52.90.79.130:8080/MidTerm/get/products
name	Name of the Product
image_urls	Image URLs to be used. Use first two images (91x121 and 300x400)
skus	Product details. Use the first item each time. You will be needing **msrp_price** and, **sale_price**

Main screen: please follow the instructions in the following:

1. When the user starts the app, it should display a progress dialog, while it should parse the products, see figure 1(a).
2. After parsing is complete, the screen should display a list of products retrieved from the API. It should be a RecyclerView with Grid Layout. In each cell of the grid, the app should display the larger image (300x400), the name of the product, the price of

the product (**sale_price**), discount percentage and, an Add to Cart button. See figure 1(b).

3. Discount percentage can be measured by the formula:
   (msrp_price - sale_price) / msrp_price x 100
4. Clicking on Add to Cart button will add the product to the cart. You should maintain a Cart ArrayList to keep the current Cart. For the sake of simplicity, assume that the user can buy only 1 item of a certain product only. So, when the user clicks on the Add to Cart button, it should be disabled until the user checks out. See figure 1(c).
5. There should be two menu items: Cart and, History.
6. If the user clicks on the Cart menu button, it should start the Cart screen. In this screen, the app should display a ListView containing the products in the Cart ArrayList. Below the ListView, the app should calculate the Total amount of money the user needs to pay.
7. Long Click on any item on the ListView should delete the item from the cart, see figure 1(e).
8. At the bottom of the screen, there should be two buttons: Cancel and Checkout.
9. Clicking on the Cancel button should take the user to the main screen. Remember to make Add to Cart button(s) disabled for the products already in Cart.
10. Clicking on Checkout button should save the ArrayList to SQLite database. HINT: Please create unique IDs for each Cart. You can use ***UUID.randomUUID()*** function to generate unique 128-bit value.
11. After checking out, the current Cart ArrayList should be deleted, and the user should be sent back to the main screen.
12. If the user clicks on the History menu button, the app should display an alert dialog to select from the previously made orders, see figure 1(h).
13. Clicking on one of the previous orders should take the user to the History screen, identical to the Cart, except, there is no Checkout button, see figure 1(i). Clicking on the Cancel button should take the user to the Main screen.

Rubrics:

Graphical User Interface	10 points
Parsing API	10 points
SQLite management	30 points
RecyclerView with Grid Layout	20 points
Adding to Cart (Maintain ArrayList)	5 points
ListView	20 points
Alert Dialog	5 points
Total	**100 points**