Mobile App Development

In-Class Assessment 3

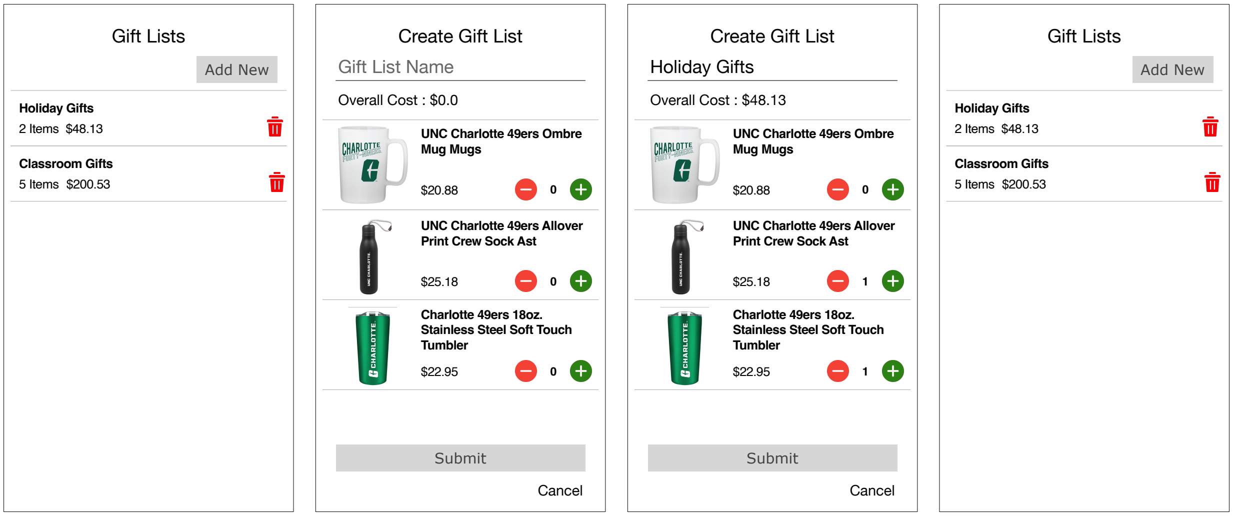
Basic Instructions:

1. This is and In Class Assessment, which will count for 10% of the total course grade.
2. This assessment is an individual effort. Each student is responsible for her/his own assessment and its submission.
3. Once you have picked up the assessment, you may not discuss it in any way with anyone until the assessment period is over.
4. During the assessment, you are allowed to use the course videos, slides, and your code from previous home works and in class assignments. You can use the internet to search for answers. You are NOT allowed to use code provided by other students or solicit help from other online persons.
5. Answer all the assessment parts, all the parts are required.
6. During the assessment the teaching assistants and Instructors will pass by each student and ask them to demonstrate their application. Your interaction with the teaching assistants and instructors will be taken into consideration when grading your assessment submission.
7. Please download the support files provided with the assessment 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. Create a zip file which includes all the project folder, any required libraries, and your presentation material. Submit the exported file using the provided canvas submission link.
10. **Do not try to use any Social Messenger apps, Emails, Or Cloud File Storage services in this exam.**
11. **Failure to follow the above instructions will result in point deductions.**
12. **Any violation of the rules regarding consultation with others will not be tolerated and will result disciplinary action and failing the course.**

<u>In-Class Assessment 3 (100 Points)</u>

In this assignment you will develop a simple gifts list application. You are provided with a Postman file that contains all the APIs for this app.

1. All the data returned by the APIs is in JSON format.
2. All the network calls should be done in a background thread.
3. All UI changes, updates and edits should be performed on the main thread.



(a) Gift Lists (b) Create Gift List (d) Create Gift List (d) Gift Lists

Figure 1, App Wireframe

Part 0, Authentication and Flow (0 Points):

You are provide with the authentication screens and the comments will indicate how you can retrieve the access token which is required to make the API calls. Please check the provide files to access the authentication token. Please follow the steps:

1. All the API calls will require an "Authorization" header with the "BEARER token" in order to make the api calls.

Part 1, Gift Lists (25 Points):

This screen displays a list of gift lists as shown in Figure 1(a). Please follow the steps:

1. Retrieve the gift lists for the currently logged user using the "/api/giftlists/lists" API:
   a. Note that this API requires the Authorization header to include the token
   b. Make the api request and then parse the JSON response.
   c. Refresh the displayed list to show the different gift lists as shown in Figure 1(a).
2. Each row item should display gift list name, total number items and total cost of the gift list
3. Clicking the trash can in a row item should delete the selected gift list by calling the delete gifts "/api/giftlists/delete" API:
   a. Note that this API requires the Authorization header to include the token
   b. Retrieve the update gift lists by calling the "/api/giftlists/lists" API, parsing the JSON and then refresh the displayed list.
4. Clicking the "Add New" button should:

    a. Transition to the the Create Gift List screen.
    b. Upon returning this screen refresh the list of Gift Lists by retrieving the shopping gift lists by calling the “/api/giftlists/lists” API.

5. Clicking on a row item should:
    a. Transition to the Gift List screen, send it the selected GiftList object.
    b. Upon returning this screen refresh the list of Gift Lists by retrieving the shopping gift lists by calling the “/api/giftlists/lists” API.

Part 2, Create Gift List Screen (35 Points):

This screen is a form that is used to create new gift list. This screen also maintains a list of products that will be added to the new gift list, please follow the steps:

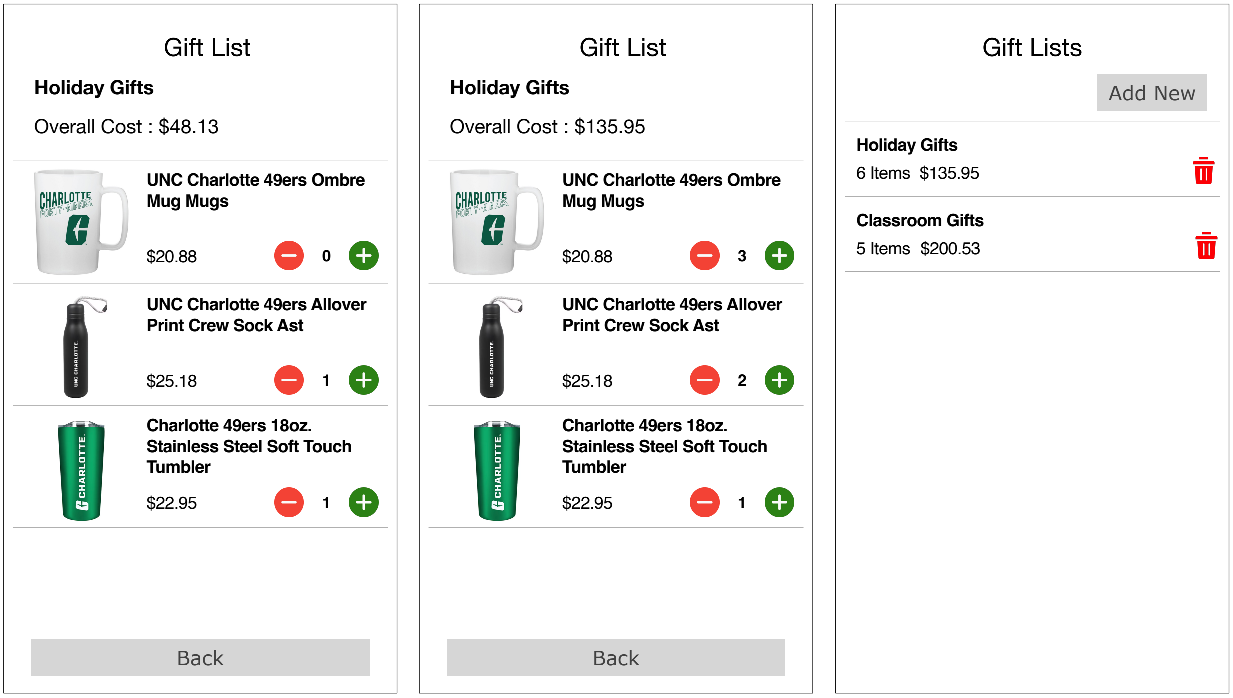
1. Retrieve the list of products using the “/api/giftlists/products” API:
    a. Note that this API requires the Authorization header to include the token
    b. Make the api request and then parse the JSON response.
    c. Refresh the displayed list to show the products retrieved and set the count for each product to 0 as shown in Figure 1(b).
2. Each row item should display the product name, price, image and the count selected for each product as shown in Figure 1(b).
3. Note that this screen also displays the “Overall Cost” which is the sum of the cost of all the products selected and displayed in the list.
4. Clicking on the “+” icon should:
    a. Increment the count for the selected product by 1, refresh the list to display the updated row.
    b. Update the “Overall Cost” display to accommodate the changed count.
5. Clicking on the “1” icon should:
    a. Decrement the count for the selected product by 1, refresh the list to display the updated row.
    b. Update the “Overall Cost” display to accommodate the changed count.
6. Clicking on the Submit button should:
    a. If the name is not entered or no products selected then show a toast message indicating the missing form input.
    b. If all the required data is entered correctly then:
        - Call the “/api/giftlists/new” API to add the gift list to the server. After the API is completed successfully then go back to the Gifts List screen.
        - Note that this API requires the Authorization header to include the token

Part 3, Gift List Screen (20 Points):

This screen display the details of specific Gift List as shown in Figure 2(a). Please follow the steps:

1. Display the products list associated with the current Gift List object as shown in Figure 2(a).
2. Note that this screen also displays the “Overall Cost” which is the sum of the cost of all the products selected and displayed in the list.
3. Clicking on the “+” icon should:
    a. Call the “/api/giftlists/add-item” API. Note that this API requires the Authorization header to include the token.

   b. After the API returns then increment the count for the selected product by 1, refresh the list to display the updated row.
   c. Update the “Overall Cost” display to accommodate the changed count.
4. Clicking on the “-” icon should:
   a. Call the “/api/giftlists/remove-item” API. Note that this API requires the Authorization header to include the token.
   b. After the API returns then decrement the count for the selected product by 1, refresh the list to display the updated row.
   c. Update the “Overall Cost” display to accommodate the changed count.
5. Clicking the “Back” button should go back to the Gift Lists Screen


Gift List
Holiday Gifts
Overall Cost : $48.13
UNC Charlotte 49ers Ombre Mug Mugs
$20.88
0
UNC Charlotte 49ers Allover Print Crew Sock Ast
$25.18
1
Charlotte 49ers 18oz. Stainless Steel Soft Touch Tumbler
$22.95
1
Back
Gift List
Holiday Gifts
Overall Cost : $135.95
UNC Charlotte 49ers Ombre Mug Mugs
$20.88
3
UNC Charlotte 49ers Allover Print Crew Sock Ast
$25.18
2
Charlotte 49ers 18oz. Stainless Steel Soft Touch Tumbler
$22.95
1
Back
Gift Lists
Add New
Holiday Gifts
6 Items $135.95
Classroom Gifts
5 Items $200.53

(a) Gift List

(b) Gift List

(c) Gift Lists

Figure 2, App Wireframe

Section:	
Student Name:	
Student ID:	

Part #	Features	Total	Grade	Comments
Part 1	Call the get gift lists API and parse the JSON + Display the list of items correctly (total items and total cost calculated correctly)	15		
Part 1	Call the delete gift list API to delete the selected item + refresh the list of items correctly	10		
Part 2	Call the get products API and parse the JSON + Display the list of items correctly	15		
Part 2	Call the + and - implemented correctly, refreshes the list upon change, and display the overall cost correctly.	10		
Part 2	Clicking “Submit” does the validation checking, calls the add new gift list correctly. Then transitions back.	10		
Part 3	Display the list and the overall cost correctly the overall cost correctly.	10		
Part 3	Call the + and - implemented correctly, call the correct api, refreshes the list upon change, and display the overall cost correctly.	30		
	Total	**100**		

Table 1: Grading Key