ASSIGNMENT 1
DUE ON: Saturday June 24, 2023, 11:59 PM
Delicia Beverage
Total Mark: 10

1. Overview: This document specifies the functional, non-functional, and deliverable requirements for the **Delicia Beverage** Android application development assignment.

2. Scope: This assignment is designed to allow you to master the implementation of the standard GUI elements that you will encounter on the current generation of mobile phones. You are tasked to implement a user interface that uses the UI elements outlined in this document, for developing an Android application for a shop dealing with beverage sales.

3. Functional Requirements: The following functional requirements must be implemented to complete the assignment. Create a new Android project with an empty views activity. Application should be named in the format 'YournameAssignment1' (Example: PeterAssignment1).

 3.1 App bar: Design the app bar using material tool bar. Implement a logo of the shop and the title 'Delicia Beverage' on the app bar. Check http://www.iconarchive.com to get your free logo image.

 3.2 Customer Name: The TextInputLayout material design component should be implemented to enter the customer name.

 3.3 E-mail id: The TextInputLayout material design component should be implemented to enter the customer e-mail id.

 3.4 Customer Phone Number: The TextInputLayout material design component should be implemented to enter the customer phone number.

 3.5 Type of Beverage (Tea/Coffee): Use radio buttons to provide the two choices for the type of beverage '**Coffee**' or '**Tea**'. The option 'Coffee' should be default selected when the app launches.

 3.6 Additionals: The user must be able to select additionals '**Milk**' and/or '**Sugar**'. Use checkboxes to implement this functionality.

 3.7 Size of Beverage: Use radio buttons to select the size of beverage '**Small**', '**Medium**' and '**Large**'. All radio buttons should be **unchecked** when the app launches.

 3.8 Added flavourings: Use a spinner dropdown to select any added flavorings.
 - For coffee this could be 'None', 'Pumpkin Spice' or 'Chocolate'
 - For tea this could be 'None', 'Lemon' or 'Ginger'

 Note: There should be only one spinner implemented for the added flavourings. The choices populated in this spinner should dynamically change as per the selection of the type of beverage.

 3.9 Select the region: Use an AutoCompleteTextView with a TextInputLayout to select a region out of the four regions – 'Waterloo', 'London', 'Milton' and 'Mississauga'. The user must be able to select **any of these four regions only** by typing the first two letters of the region.

 4.0 Select the store: Use a spinner to dynamically populate the address of stores as per the selection of the region. The list of stores for different regions should be as given in the below table:

Region	List of Stores
Waterloo	1. 65 University Ave E 2. 415 King St 3. 585 Weber St
London	1. 616 Wharncliffe Rd 2. 1885 Huron St 3. 670 Wonderland Road 4. 1181 Highbury Ave
Milton	1. 900 Steeles Ave 2. 80 Market Dr 3. 820 Main St
Mississauga	1. 144 Dundas St 2. 30 Eglinton Ave 3. 6075 Creditview Rd

 3.1 Sales Date: User must be able to select the sales date using datepicker.

3.2 Print Bill: Implement a button to calculate the total invoice amount based on the price chart given below. A snackbar should be implemented to display the details such as customer name, phone number, type of beverage, additionals, flavourings, size of beverage, region, store, and total bill amount. The total bill amount should be computed with additional taxes; taxes are 13%. (Note: make sure that the amount of money billed is displayed with a dollar sign and two decimal places.)

Price Chart				
Tea		**Coffee**		
Size:		**Size:**		
Small	$1.50	Small	$1.75	
Medium	$2.50	Medium	$2.75	
Large	$3.25	Large	$3.75	
Additionals:		**Additionals:**		
Milk	$1.25	Milk	$1.25	
Sugar	$1.00	Sugar	$1.00	
Flavorings:		**Flavorings:**		
None	0.00	None	0.00	
Lemon	$0.25	Pumpkin Spice	$0.50	
Ginger	$0.75	Chocolate	$0.75	

5. Implement the best practices and validations. The following validations must be implemented:
a) Data should be entered in the fields customer name, email-id, customer phone number, region, and sales date.
b) Email id entered should be in a valid email format. (For example, tmathew@gmail.com)
c) One of the beverage sizes should be selected.
d) Sales date entered must be less than or equal to the current date.

6. Material Design Guideline Principles should be followed to design a good-looking UI.

7. Implement Object Oriented Programming principles with at least one Java class in which encapsulation is applied. You are free to choose a Java class that is relevant in the application.

8. Non-Functional Requirements: The following non-functional requirements must be implemented to complete the assignment.

8.1 Mobile Application: The 'Delicia Beverage' mobile application must be implemented for a mobile device of your choosing.

9. Deliverable Requirements

9.1 Comment the Code: You must provide meaningful comments to every class, method, and every and any significant code segment.

9.2 Source Code: You must submit YOUR OWN WORK via DropBox.

9.3 Screen Shots: Include all relevant screenshots of your running application and commented Java code in a Word document. Do not zip this document. **Commented code should be pasted as text.** Submission will not be considered for evaluation if commented code is pasted as an image and a grade of 0 (zero) will be awarded. Name this file as 'YournameAssignment1.docx' (Example: PeterAssignment1.docx)

9.4 Export your Android application as a zip file. (Choose File -> Export -> Export to Zip File)

9.5 Upload the Word document and the zip file of the application package on e-Conestoga into the folder 'Assignment 1 – Delicia Beverage'.

9.6 This is an individual assignment.

9.7 For late submission: marks will be deducted per day as specified below.

9.8 Please check the marking sheet in the next page for the distribution and deduction of marks.

9.9 Due Date: **Saturday June 24, 2023, 11:59 PM**.

Marking Sheet

Description	Marks Allocated	Marks Achieved
Page designed as specified	2	
All functionalities implemented as per requirements	6	
Implementation of OOP principles with one Java class having encapsulation applied	2	
Deduction		
Non submission of Word document file with all relevant screenshots and commented Java code pasted as text	8	
Non implementation of ViewBinding	4	
Runtime errors	1.5 x --------	
Hardcoded text found in the UI	3	
Hardcoded color found in the UI	3	
Validations not implemented	1 x --------	
Material design guidelines not followed in the UI design	4	
Assignment Standard (proper project name, submission docs name etc)	1 x --------	
Programming Standards	1 x --------	
Late Submission (softcopy)	-------- days	
Bugs (including missing requirements mentioned in this specification)	1-5 based on severity	
Total		

Late Penalty (Softcopy submission)

Days Late	Penalty %
1	5
2	10
3	20
4	40
5	80
6	100

Please note: How "Days Late" is calculated: Your assignment is due on Sunday 11:59pm. You are considered 1 day late if you submit on anytime on Monday (until 11:59PM). If you submit anytime on Tuesday (until 11:59PM). you are considered 2 days late.