CMPS 312 Mobile App Development
Banking App
Lab Assignment 3
Deadline - Sunday October 24, 2021

Objective

The objective of this assignment is to practice building an android app that communicates with a Web API using Ktor and coroutines. You will also practice using the navigation component, View Models, and State variables.

Overview

In the assignment, you will design and implement a **Bank App** that allows the bank employees to add, update, delete accounts. A web version of the app is available at https://employee-bank-app.herokuapp.com. Your task is to implement a mobile app version to allow listing, adding, updating and deleting accounts.

Implementation Instructions

You should write the whole app by yourself no base solution is provided. You can use the skills acquired in labs **6, 7 , 8 and 9** to help you deliver the app.

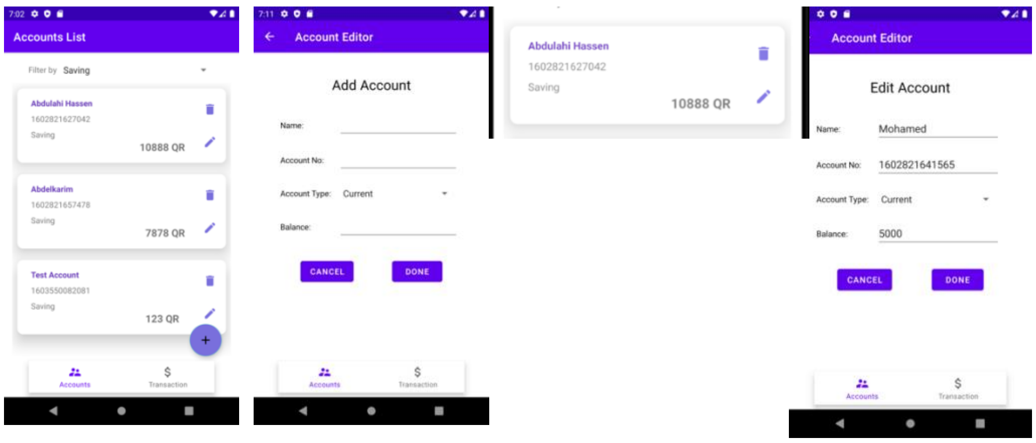
1. Create a new project and name it "**QA-Banking App**" under your repository and add all the needed dependencies.
2. Add one model class [**Account**] that can store the Account objects that you will get from the Web API. An example account object in JSON format is shown below.

```
{
    "accountNo": "1602821627042"
    "name": "Abdulahi",
    "acctType": "Saving",
    "balance": 10888,
}
```

3. Use the **Ktor** library to communicate with the Web API. First, create a Service Interface and an Implementation of the service to **get**, **add**, **update**, and **delete** accounts using the Web API shown in the table below and available at https://employee-bank-app.herokuapp.com. You can test the Web API using **Postman** to better understand what each of the following URLs returns and how to call them.

HTTP Verb	Url	Functionality
Get	/api/accounts	Get all counts
Get	/api/accounts/:accountNo	Get an account by account no
Post	/api/accounts	Add an account
Put	/api/accounts/:accountNo	Update an account
Delete	/api/accounts/:accountNo	Deletes an account by account no

4. Create the app navigation, that allows the user to access **Accounts List**, **Add account, and Update account screens** (As shown in figure 1).
5. Design and implement the screens for list, add, update and delete account.
   - Implement Account List screen as show in Figure 1.
   - Implement the delete account. When the user clicks on the delete button, you should remove the account from both the list as well as from the server by calling the Web API. Then check if the account is removed by visiting this URL [https://employee-bank-app.herokuapp.com]
   - Implement the **add** and **update** account scenarios as shown in Figure 1. You must use the same screen for Add & Update account. **Note**-> Account Type is a dropdown [saving, current].



Accounts List

Filter by Saving

Abdulahi Hassen
1602821627042
Saving
10888 QR

Abdelkarim
1602821657478
Saving
7878 QR

Test Account
1603550082081
Saving
123 QR

Account Editor

Add Account

Name:
Account No:
Account Type: Current
Balance:

CANCEL DONE

Abdulahi Hassen
1602821627042
Saving
10888 QR

Account Editor

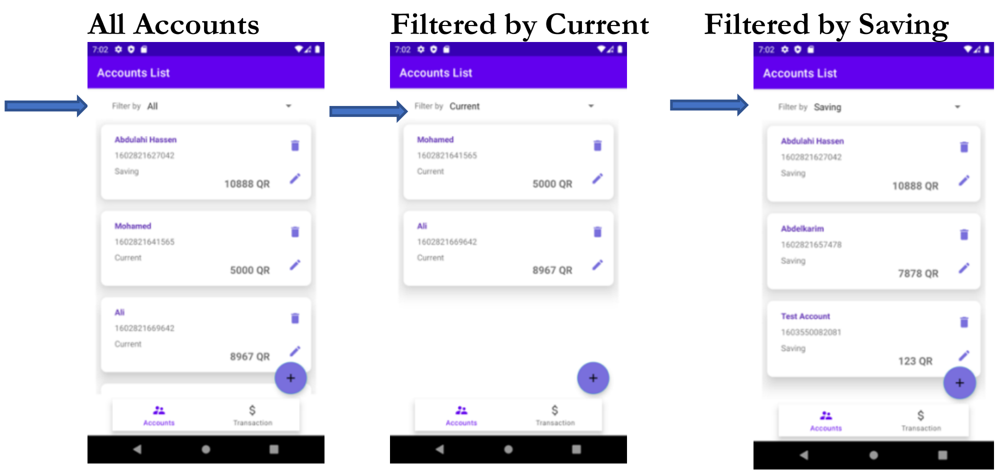
Edit Account

Name: Mohamed
Account No: 1602821641565
Account Type: Current
Balance: 5000

CANCEL DONE

Figure 1. List and Add Account

6. Implement the accounts filtering by different account types. When the user, selects the dropdown you should show the three options [all, current and saving].
   a. **All** -> **gets all the accounts** from https://employee-bank-app.herokuapp.com/api/accounts
   b. **Current** -> gets **current accounts** from https://employee-bank-app.herokuapp.com/api/accounts?acctType=Current
   c. **Saving** -> gets saving accounts from https://employee-bank-app.herokuapp.com/api/accounts?acctType=Saving



Accounts List

Filter by All

Abdulahi Hassen
1602821627042
Saving
10888 QR

Mohamed
1602821641565
Current
5000 QR

Ali
1602821669642
Current
8967 QR

Accounts List

Filter by Current

Mohamed
1602821641565
Current
5000 QR

Ali
1602821669642
Current
8967 QR

Accounts List

Filter by Saving

Abdulahi Hassen
1602821627042
Saving
10888 QR

Abdelkarim
1602821657478
Saving
7878 QR

Test Account
1603550082081
Saving
123 QR

Figure 2. Filter Account By Type

Submit the testing sheet as well as the code under "**your_repository/assignments/assignment3**"