Lappeenrannan teknillinen yliopisto

School of Software Engineering

Software Development Skills: Full-Stack

Naosad Hossen, 001489455

LEARNING DIARY, MOBILE 2023-24 MODULE

**LEARNING DIARY**

2.01.2024

I enrolled to “Software Development Skill: Mobile 2023-24” to learn and understand how software development works in this kind of technology. As a software project manager, it is important to understand this popular technology. So, I am excited to participate in this course and gaining practical knowledge about Mobile Development technology.

I went through “General Course Information” and understood the course goal is to improve my competitiveness in the job market by providing tools for creating unique projects and to help me find your passion as a software developer.

While learning how to develop Android Mobile Apps, I will develop a Apps call “Customer Contact Management” which will talk to the following APIs to login, add customer contact and view all customers information:

1. Login API: <http://192.168.0.140:8080/users/authenticate>
2. Get Profile API: [http://192.168.0.140:8080/users/profile](http://192.168.0.140:8080/users/profileauthenticate)
3. Add Customer Contact API: http://192.168.0.140:8080/customers/addcustomer
4. Fetch All Customers Information API: http://192.168.0.140:8080/customers/ /getallcustomers

All these APIs were developed by myself in course “Software Development Skills: Full-Stack 2023-24” and backend is installed locally.

02-01-2024

# Install ANdroid studio and environment setup

Installed Android Studio (version Hedgehog | 2023.1.1) and created a test project using Empty Views Activity.

# Watching the Videos

## Introduction [Part 1]

Followed the coding in the video, got myself familiar with the folder structure and built a sample App. Also learnt how to debug code.

## Core Elements [Part 2]

Followed the coding in the video. Here I learnt how to write code go to another page in the App (Activity) our outside of the App (such as Web Browser) and pass some value to the page.

## Lists, Layouts, and Images [Part 3]

Followed the coding in the video. Here I learnt how to use ListView to display array data and format the view. Here also practiced how to use Activities and Intent to switch between the Activities. Also learnt customer template to organize the TextView items for ListView element.

02-01-2024 and 04-01-2024

# developing customercontactmgmt apps

GitHub Repository: <https://github.com/naosadhossen/CustomerConactMgmt>

Created a Android Project call “CustomerContactMgmt” with Empty View Activity and language as JAVA.

A screenshot of a computer

Description automatically generated

Added the following in manifest file as the App will interact with API over the network.

<uses-permission android:name="android.permission.INTERNET" />

## Login Page

This is the main page with two input field (TextView) for Username and Password and a Login Button. I want to see the JWT token that I received after successful login, so added a Multiline Text box where JWT token will be displayed. Below is the layout:

A screenshot of a phone

Description automatically generated

As the login will POST authentication information to API, I imported the following in my code:

import com.google.gson.Gson**;**import okhttp3.MediaType**;**import okhttp3.RequestBody**;**import retrofit2.Call**;**import retrofit2.Callback**;**import retrofit2.Response**;**import retrofit2.Retrofit**;**import retrofit2.converter.gson.GsonConverterFactory**;**

Upon clicking on Login, it does not move forward. After debugging, the error message was

"CLEARTEXT communication to 192.168.0.140 not permitted by network security policy,". The reason was that Andriod does not allow unencrypted communication by default. My API is running on HTTP and not HTTP. The following blog provided the solution to add the following in Application section in Manifest file.

https://stackoverflow.com/questions/41650965/cleartext-communication-not-supported-on-retrofit

android:usesCleartextTraffic="true"

After adding this property, I can login now.

## Profile Page with buttons to go to add customer page and view all customer page.

After login my App will show Profile page to Welcome the user and give two buttons which will direct to Add Customer Page and View All Customer page. Login page will pass the JWT Token to profile page, I would also like to show the JWT token in Multiline Text box.

Here is the page layout.

A screenshot of a phone

Description automatically generated

When the page loads it will make a GET call to API [http://192.168.0.140:8080/users/profile](http://192.168.0.140:8080/users/profileauthenticate) to receive the profile information (Username, Name, Email) with JWT Token.

The Name will be shown in the TextView box as Welcome, <Name>

A screenshot of a phone

Description automatically generated

In order to make GET call to the API, I used Volley HTTP library and imported following in the code.

import com.android.volley.AuthFailureError**;**import com.android.volley.Request**;**import com.android.volley.RequestQueue**;**import com.android.volley.Response**;**import com.android.volley.VolleyError**;**import com.android.volley.toolbox.JsonObjectRequest**;**import com.android.volley.toolbox.StringRequest**;**import com.android.volley.toolbox.Volley**;**

## Add Customer Page

I will add a customer in the system with following information: Name, Email. Address, Phone, Country. Below is the page layout.

A screenshot of a phone

Description automatically generated

Add Customer button will make a POST API call to <http://192.168.0.140:8080/customers/addcustomer>

Here I also used Volley HTTP library.

Upon pressing the Add Customer button, it will add the user in the system and will go to View All Customer page.

## View All Customer Page

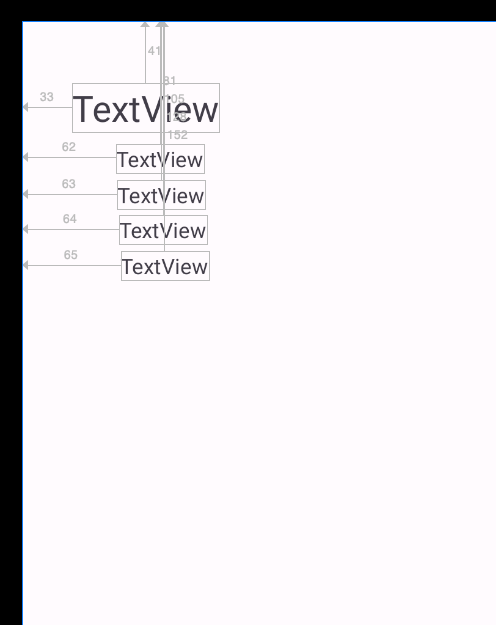
Here I used ListView to show the GET API result call to http://192.168.0.140:8080/customers/ /getallcustomers

Below is the layout.

A screenshot of a phone

Description automatically generated

For each item, I created a customer list item view as below.



A “Customer” class and “CustomerAdapter” class was created to populate this ListView with JSON Object received from API call.