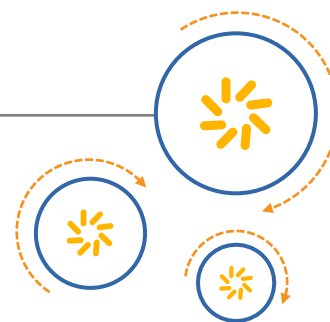




Qualcomm Technologies, Inc.



Device Characterization Platform

Developer Guide

DCP-Developer-Guide-Android-App

September 22, 2019

Copyright © 2019 Qualcomm Technologies, Inc.

This work is licensed under a [Creative Commons Attribution-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nd/4.0/).

Revision history

Revision	Date	Description
A	September 2019	Initial release

Contents

1 Introduction	4
1.1 Purpose & Scope	4
1.2 Audience & Technical Stakeholder	4
1.3 Definitions, Acronyms & Abbreviations	4
1.4 References	4
1.5 Getting started	4
2 Installation	5
2.1 System Requirements.....	5
2.1.1 Software Requirements	5
2.1.2 Hardware Requirements.....	5
2.2 Operating System	5
2.3 Manual Installation	5
3 Configuration	6
3.1 App Configuration	6
4 Architecture.....	6
4.1 Dependencies.....	6
4.2 File Structure	7
5 Testing.....	8
6 Publish App.....	8

1 Introduction

1.1 Purpose & Scope

This document provides:

- Installation instructions for the Device Characterization Platform Mobile app for
- Instructions for running test commands/steps to verify DCP app installation

1.2 Audience & Technical Stakeholder

This Installation and Deployment Guide is intended to be used by technical stakeholders of the Device Characterization Platform Mobile app, who will be responsible for planning, performing, or maintaining the installation or deployment, such as the System Administrators and Developers.

1.3 Definitions, Acronyms & Abbreviations

Table 1- Definitions, Acronyms & Abbreviations

Term	Explanation
DCP	Device Characterization Platform
OS	Operating System

1.4 References

N.A.

1.5 Getting started

The instructions provided in this document assume that the recommended equipment has been installed and configured with Ubuntu/Windows/Mac OS.

2 Installation

NOTE: The reader acknowledges and agrees that it is entirely and solely responsible for the selection and use of all third-party software modules downloaded and installed by this installation method, including securing all appropriate and proper rights of use to any of such third-party software modules and to comply fully with any terms of use that may apply to or accompany any such third-party software modules.

Qualcomm Technologies, Inc. does not undertake any obligations, duties, or other responsibilities in connection with the selection or use by the reader of any of such third-party software modules.

2.1 System Requirements

2.1.1 Software Requirements

Below mentioned are the software requirements.

- JDK 1.8 or more
- Android Studio 3.5
- Target Android SDK v29
- Minimum Android version 14
- Gradle 5.4.1
- Android Plugin version 3.5.0

2.1.2 Hardware Requirements

Below are the hardware requirements to run DCP.

- At least 512 MB of RAM
- At least 1G of disk space
- Camera
- Android Device
- Active Internet Connection

2.2 Operating System

This system can be installed and configured with Ubuntu/Windows/ Mac OS.

2.3 Manual Installation

Import-dcp-android-1.0.0 in Android Studio.

3 Configuration

3.1 App Configuration

- To change the logo of app go to app/res/drawable folder and paste logo file but make sure the file name should be dcp_logo.png
- To change the colors of the app go to app/res/values/colors.xml file and mention hex color code of your required color
- To change app icon update all PNG files named “ic_launcher” in folders starting with “mipmap”
- To change DCP backend API base URL open root_directory/app/build.gradle and update value of BASE_URL in buildTypes.

4 Architecture

DCP Android application is developed using MVP(Model View Presenter) architecture.

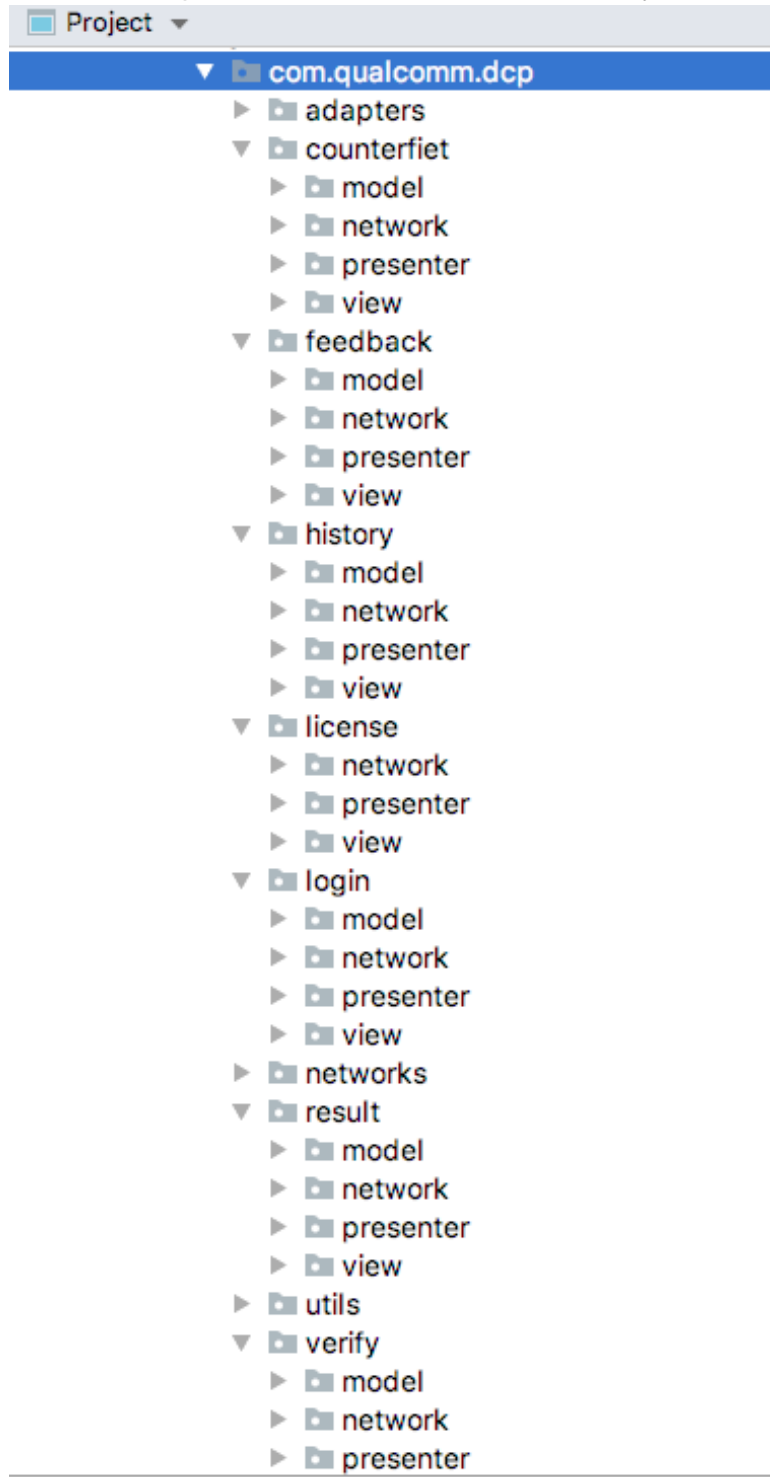
4.1 Dependencies

DCP app utilizes some open source libraries to meet its functional requirements. The libraries used and the purpose of their usage are

- AndroidX libraries for designing user interface
- Retrofit and OkHTTP3 for making network calls ([View on Github](#))
- Barcode Scanner for scanning barcode of IMEI ([View on Github](#))
- RxJava and RxAndroid for reactive programming ([RxJava on Github](#), [RxAndroid on Github](#))
- ButterKnife for binding views and callbacks to fields and methods ([View on Github](#))
- Espresso for unit testing ([View Online](#))

4.2 File Structure

In DCP source code files are structured in a way that files related to a screen are placed in a folder having sub folders for model, view, presenter and network calls. There are some other folders for adapters, network related files and utility files.



5 Testing

To test the successful installation of app, run the app either on any physical device or on Android Emulator. For testing the app on physical device make sure that USB Debugging is enable. If you don't know how to enable USB Debugging on your device visit this [link](#).

6 Publish App

To publish app on Google play store follow instructions given on this [link](#).