Android Programming: QR-Code
Mobile Internet Lab3

Liu Hanwen (517030910294)
Shanghai Jiao Tong University IEEE Honer Class
hwLiu2017@sjtu.edu.cn

May 1, 2020

1 Environment

I use Android Studio 3.6.2 with Oracle Java 1.8.0.

2 Brief Introduction of QR-Code

A QR code uses four standardized encoding modes (numeric, alphanumeric, byte/binary, and kanji) to efficiently store data; extensions may also be used.

The QR Code system became popular outside the automotive industry due to its fast readability and greater storage capacity compared to standard UPC barcodes.
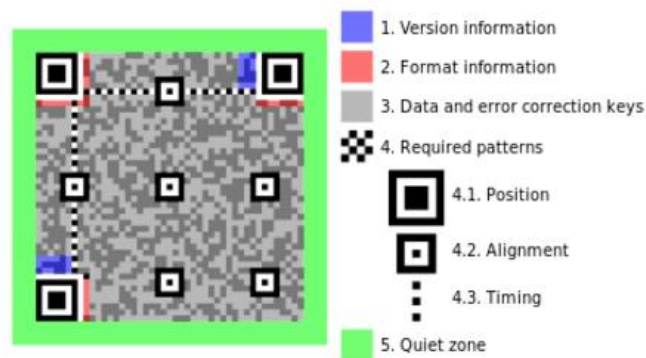
Applications include product tracking, item identification, time tracking, document management, and general marketing. A QR code consists of black modules (square dots) arranged in a square grid on a white background, which can be read by an imaging device (such as a camera, scanner, etc.) and processed using Reed–Solomon error correction until the image can be appropriately interpreted. The required data are then extracted from patterns that are present in both horizontal and vertical components of the image.

3   Project: QR-Code

3.1   Small Trouble while Running

Provided code has complete function, and it can pass building process successfully. While in the process of deploying it onto my Android Phone, it meets a small problem with the following error message:

- Unable to determine application id: com.android.tools.idea.run.ApkProvisionException:

- No outputs for the main artifact of variant

I find out that this problem is introduced by my gradle version in project build.gradle configuration is too old for a newer Android Studio version. In this case, I change the version in my gradle as the following steps:

- Change the gradle configuration into a newer one:

  Path: build.gradle(project)->buildscript->dependencies->classpath;

  I change it to *classpath 'com.android.tools.build:gradle:3.1.2'*

- Change the configuration in gradle-wrapper.properties into a newer version.

  I change it to *distributionUrl=https://services.gradle.org/distributions/gradle-4.4-all.zip*

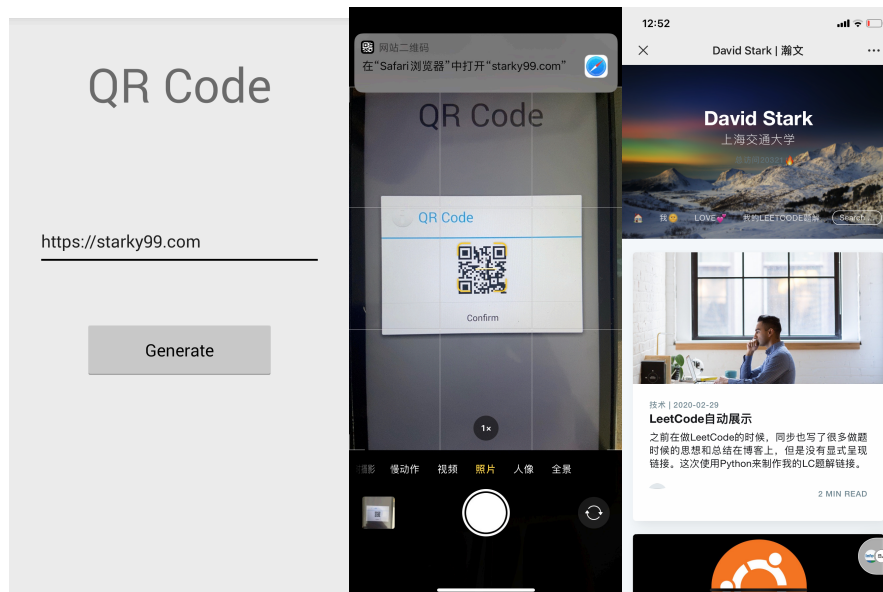And then rebuild it and it can be successfully deployed.

3.2   Performace

The performance is good.

4   Demo

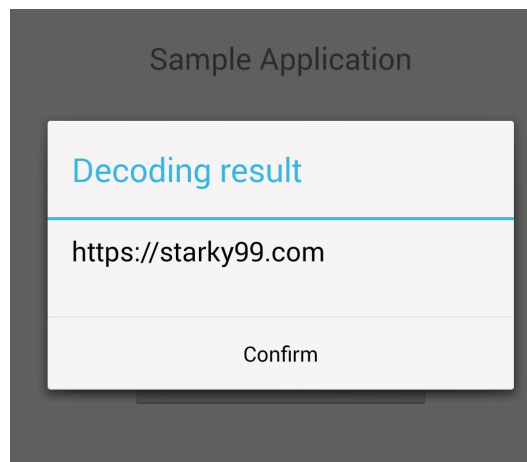The URL I use is my personal blog: https://starky99.com

4.1   Encode

I input my URL, and then the program gives me the corresponding QR-Code. Then I use my iPhone to scan this QR-Code, and I redirect to my home page as the URL indicates.

5 Decode

The program also give me correct URL from the QR-Code.

6 Conclusion

By this project, I learn about how to import other's tools and make use of camera, and then realize encode and decode the QR-Code.