EXPERIMENT- 7

Aim: To test and deploy production ready Flutter App on Android platform

Lab Outcome: Analyze and Build production ready Flutter App by incorporating backend services and deploying on Android / iOS.

Theory:

1. What is widget testing?

Widget testing is a type of software testing used in the development of user interface (UI) components for mobile and web applications. It involves testing individual UI components, or widgets, to ensure that they function correctly and meet the desired design specifications.

In the context of mobile app development using frameworks like Flutter, widget testing involves writing automated test scripts that simulate user interactions with the app's widgets, such as buttons, text fields, and menus. The tests verify that the widgets behave correctly in response to user inputs and that they render as expected on different screen sizes and resolutions.

Widget testing helps developers to catch UI issues early in the development cycle, enabling them to make corrections before the app is released to users. It can also help to identify regressions caused by code changes or updates to the framework or dependencies. By running tests frequently during the development process, developers can ensure that the app's UI is consistent, responsive, and user-friendly.

Widget testing is performed within a Dart test file, where developers can use the Flutter testing framework to create and run tests. The framework provides APIs for creating widget instances, injecting dependencies, and interacting with widgets.

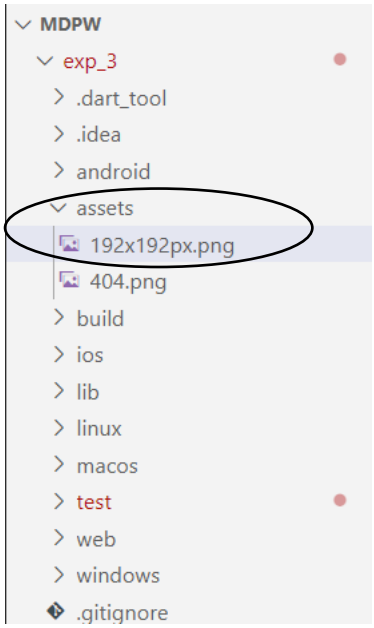
Widget testing in Flutter provides a fast and efficient way to catch UI bugs early in the development process, helping to ensure that the application's UI is functioning as intended. It can also help developers to make changes to the UI with confidence, knowing that their tests will catch any regressions that may be introduced.

2. ***Steps to set icon for flutter application.***

Practical & Outcome:

Step 1: Create an icon of size 192 x 192 px in png format.

Step 2: Add the icon in the assets folder



Step 3 : Add dependencies for the icon by visitng the pub dev site

```
dev_dependencies:
  flutter_launcher_icons: "^0.13.1"

flutter_launcher_icons:
  android: "true "
  ios: true
  image_path: "assets/icon/icon.png"
  min_sdk_android: 21 # android min sdk min:16, default 21
```

Step 4: run the ‘flutter pub get’ command
Then run ‘flutter pub run flutter_launcher_icons:main’

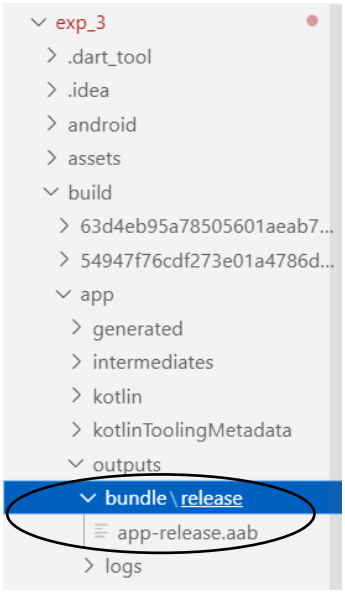
```
PS C:\Users\husna\OneDrive\Desktop\MDPW\exp_3> flutter pub get
Running "flutter pub get" in exp_3...
Resolving dependencies... (1.0s)
  async 2.10.0 (2.11.0 available)
  characters 1.2.1 (1.3.0 available)
  collection 1.17.0 (1.17.1 available)
  js 0.6.5 (0.6.7 available)
  matcher 0.12.13 (0.12.15 available)
  material_color_utilities 0.2.0 (0.3.0 available)
  meta 1.8.0 (1.9.1 available)
  path 1.8.2 (1.8.3 available)
  petitparser 5.1.0 (5.3.0 available)
  source_span 1.9.1 (1.10.0 available)
  test_api 0.4.16 (0.5.2 available)
Got dependencies!
PS C:\Users\husna\OneDrive\Desktop\MDPW\exp_3> flutter pub run flutter_launcher_icons:main
This command is deprecated and replaced with "flutter pub run flutter_launcher_icons"
  ═══════════════════════════════════════
     FLUTTER LAUNCHER ICONS (v0.13.1)
  ═══════════════════════════════════════

• Creating default icons Android
• Overwriting the default Android launcher icon with a new icon

WARNING: Icons with alpha channel are not allowed in the Apple App Store.
Set "remove_alpha_ios: true" to remove it.

• Overwriting default iOS launcher icon with new icon
No platform provided

✓ Successfully generated launcher icons
```

Step 5: After saving the file run the following command for app bundle:

flutter build appbbundle

```
PS C:\Users\husna\OneDrive\Desktop\MDPW\exp_3> flutter build appbundle

 Building with sound null safety

Checking the license for package Android SDK Tools in C:\Users\husna\AppData\Local\Android\sdk\licenses
License for package Android SDK Tools accepted.
Preparing "Install Android SDK Tools (revision: 26.1.1)".
"Install Android SDK Tools (revision: 26.1.1)" ready.
Installing Android SDK Tools in C:\Users\husna\AppData\Local\Android\sdk\tools
"Install Android SDK Tools (revision: 26.1.1)" complete.
"Install Android SDK Tools (revision: 26.1.1)" finished.
Checking the license for package Android SDK Build-Tools 30.0.3 in C:\Users\husna\AppData\Local\Android\sdk\licenses
License for package Android SDK Build-Tools 30.0.3 accepted.
Preparing "Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)".
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" ready.
Installing Android SDK Build-Tools 30.0.3 in C:\Users\husna\AppData\Local\Android\sdk\build-tools\30.0.3
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" complete.
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" finished.
Checking the license for package Android SDK Platform 33 in C:\Users\husna\AppData\Local\Android\sdk\licenses
License for package Android SDK Platform 33 accepted.
Preparing "Install Android SDK Platform 33 (revision: 2)".
"Install Android SDK Platform 33 (revision: 2)" ready.
Installing Android SDK Platform 33 in C:\Users\husna\AppData\Local\Android\sdk\platforms\android-33
"Install Android SDK Platform 33 (revision: 2)" complete.
"Install Android SDK Platform 33 (revision: 2)" finished.
Running Gradle task 'bundleRelease'...                             3198.8s
√ Built build\app\outputs\bundle\release\app-release.aab (17.6MB).
PS C:\Users\husna\OneDrive\Desktop\MDPW\exp_3>
```

Once you run this command, you will find the generated .aab file in the **build/app/outputs/bundle/release/** directory of your Flutter project.

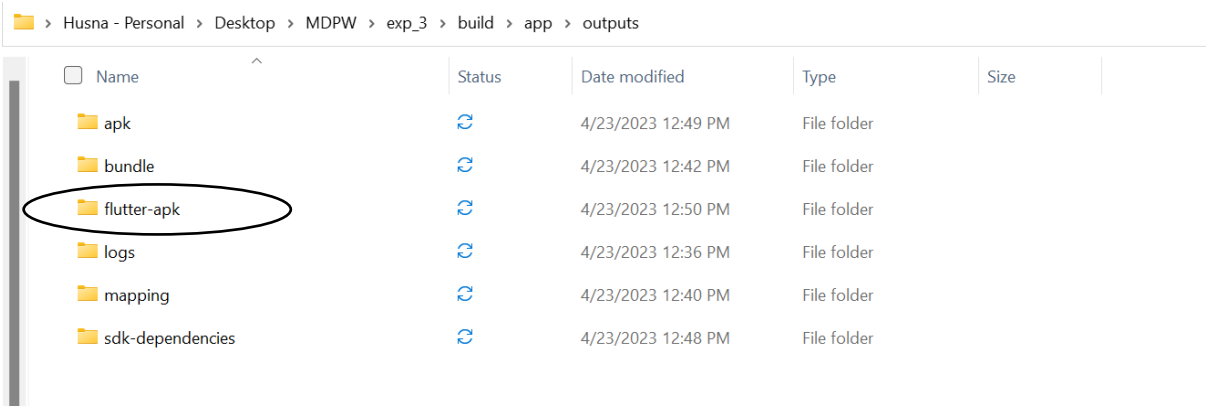


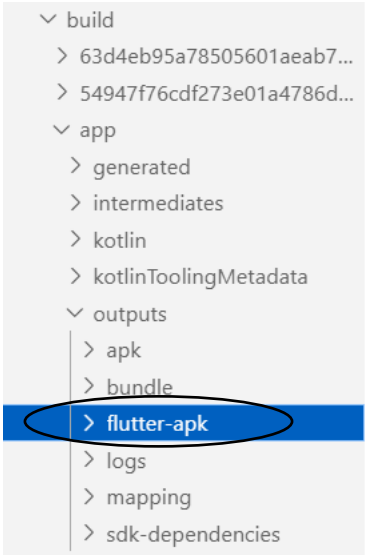
Step 6: After that run the following command for apk bundle

```
flutter build apk
```

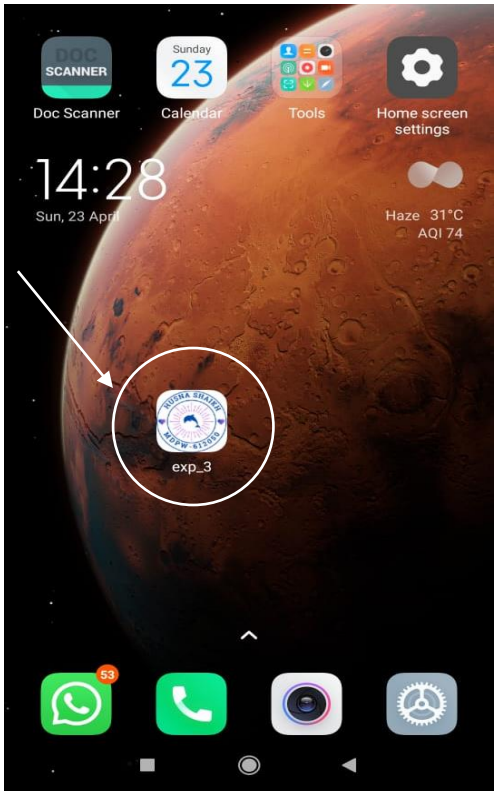
```
PS C:\Users\husna\OneDrive\Desktop\MDPW\exp_3> flutter build apk

 Building with sound null safety

w: Runtime JAR files in the classpath should have the same version. These files were found in the classpath:
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-jdk8/1.5.30/5fd47535cc85f9e24996f939c2de6583991481b0/kotlin-stdlib-jdk8-1.5.30.ja
r (version 1.5)
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-jdk7/1.6.10/e1c380673654a089c4f0c9f83d0ddfdc1efdb498/kotlin-stdlib-jdk7-1.6.10.ja
r (version 1.6)
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib/1.6.10/b8af3fe6f1ca88526914929add63cf5e7c5049af/kotlin-stdlib-1.6.10.jar (version
 1.6)
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-common/1.6.10/c118700e3a33c8a0d9adc920e9dec0831171925/kotlin-stdlib-common-1.6.10
.jar (version 1.6)
w: Some runtime JAR files in the classpath have an incompatible version. Consider removing them from the classpath
w: Runtime JAR files in the classpath should have the same version. These files were found in the classpath:
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-jdk8/1.5.30/5fd47535cc85f9e24996f939c2de6583991481b0/kotlin-stdlib-jdk8-1.5.30.ja
r (version 1.5)
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-jdk7/1.6.10/e1c380673654a089c4f0c9f83d0ddfdc1efdb498/kotlin-stdlib-jdk7-1.6.10.ja
r (version 1.6)
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib/1.6.10/b8af3fe6f1ca88526914929add63cf5e7c5049af/kotlin-stdlib-1.6.10.jar (version
 1.6)
    C:/Users/husna/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-common/1.6.10/c118700e3a33c8a0d9adc920e9dec0831171925/kotlin-stdlib-common-1.6.10
.jar (version 1.6)
w: Some runtime JAR files in the classpath have an incompatible version. Consider removing them from the classpath
Running Gradle task 'assembleRelease'...                        405.6s
√  Built build\app\outputs\flutter-apk\app-release.apk (17.5MB).
PS C:\Users\husna\OneDrive\Desktop\MDPW\exp_3>
```

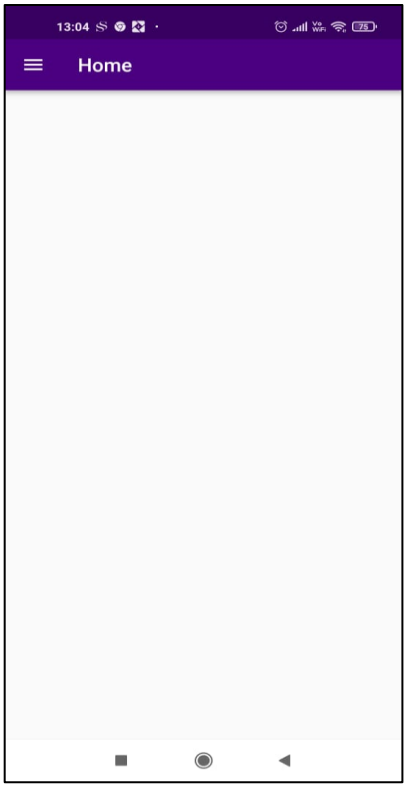
Once you run this command, you will find the generated APK file in the **build/app/outputs/flutter-apk/** directory of your Flutter project.

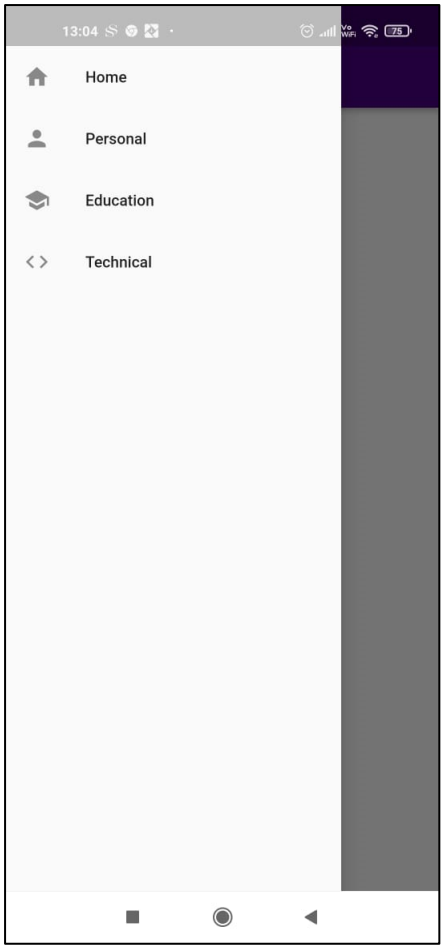


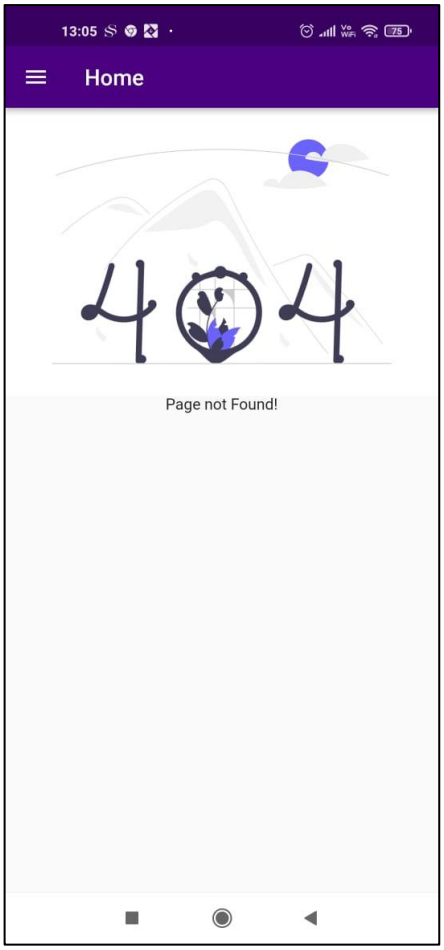


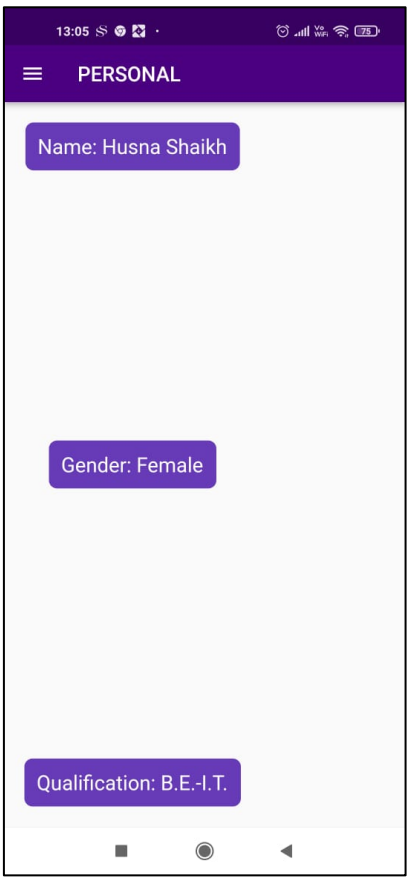
Download the apk file on your personal smart phone and install it.

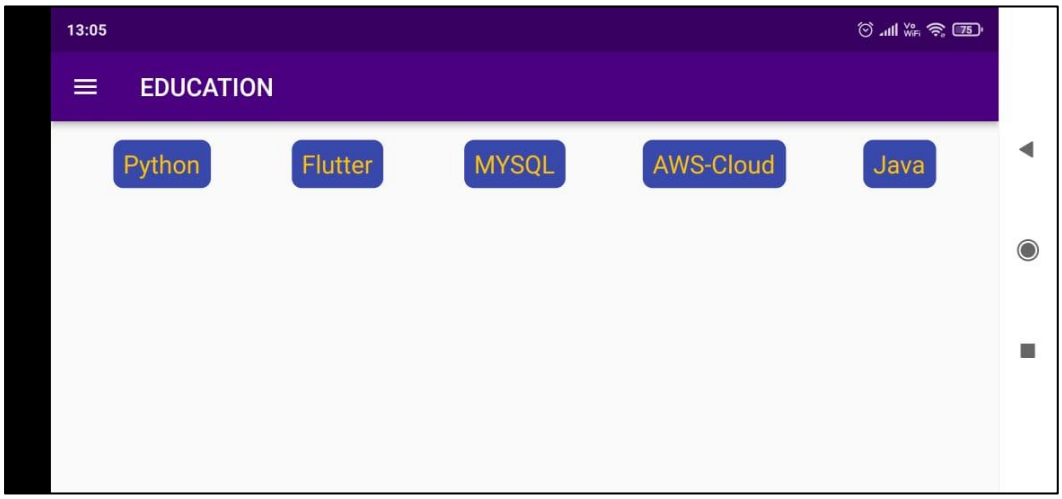





13:04
Home
Personal
Education
Technical


13:05
Home
Page not Found!


13:05
PERSONAL
Name: Husna Shaikh
Gender: Female
Qualification: B.E.-I.T.



Tools used: VS Code, Flutter

Conclusion:

From this experiment we have learned how to change the icon of a flutter app and then download and install it onto our personal device.