

Setup del proyecto con Capacitor

Si se está corriendo `ionic serve` en la consola, debe ser cancelado. Después de esto hacer un build limpio del proyecto, solucionando todos los errores que se presenten con:

```
$ ionic build
```

Luego configurar el archivo `capacitor.config.json`

```
{  
  // nombre del paquete  
  "appId": "com.unmdpfi.appclub",  
  // nombre de la app  
  "appName": "App Club",  
  // directorio web de donde tomar los assets, por defecto  
  // es www, pero ionic build crea el directorio build  
  "webDir": "build"  
}
```

Luego, crear el proyecto para Android:

```
$ ionic cap add android
```

Así, estará creada la carpeta `android` en el directorio raíz del proyecto.

A partir de este punto, cada vez que se ejecute `ionic build` que actualice el directorio web, se deben copiar los cambios en el proyecto nativo con:

```
$ ionic cap copy android
```

Deployment en Android

La app es configurada y administrada mediante Android Studio. Primero se abre el proyecto nativo Android en esta IDE ejecutando:

```
$ ionic cap open android
```

Luego, se puede conectar un dispositivo Android o usar el emulador. Se clickea en Run, se selecciona el dispositivo Android y luego OK. Así se construye, instala y lanza la app.

Si al ejecutar el comando anterior Android Studio muestra el siguiente error: *No variants found for 'app'. Check build files to ensure at least one variant exists*, hacer lo siguiente: abrir el SDK Manager (ver imagen), y activar la opción Android 10.0 (API 29). Aceptar la licencia, descargar el

paquete y reabrir el proyecto. Luego tardará un tiempo en terminar de cargar.

Reemplazar Splashes

Por defecto, la app se genera con splashes de Capacitor. Estos deben ser reemplazados por los suministrados en el repositorio, colocándolos en `android\app\src\main\res`

Modificar AndroidManifest para permitir tráfico

Crear el archivo `network_security_config.xml`, en `android\app\src\main\res\xml`:

```
<?xml version="1.0" encoding="utf-8"?>
<network-security-config>
    <base-config cleartextTrafficPermitted="true" />
    <domain-config cleartextTrafficPermitted="true">
        <domain includeSubdomains="true">
            https://couchdb-unmdp.mooo.com
        </domain>
    </domain-config>
</network-security-config>
```

Y agregar en `androidManifest.xml`:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest ...>
    <uses-permission
        android:name="android.permission.INTERNET" />
    <application
        ...
        android:networkSecurityConfig="@xml/network_security_config"
        ...>
    </application>
</manifest>
```

Compilar un APK de depuración

El APK se firma con una clave de depuración proporcionada por las herramientas del SDK y permite realizar la depuración mediante adb. Para compilar un APK de depuración, abrir una línea de comandos y navegar hasta la raíz del directorio del proyecto. También se puede hacer desde la terminal dentro del Android Studio. Ejecutar:

```
$ gradlew assembleDebug
```

De esta forma, se creará un APK con el nombre app-debug.apk en el directorio android\app\build\outputs\apk\debug. El archivo ya está firmado con la clave de depuración y alineado con zipalign, por lo que se puede instalar de inmediato en un dispositivo.

Links

- <https://ionicframework.com/docs/angular/your-first-app/6-deploying-mobile>
- <https://developer.android.com/studio/build/building-cmdline>
- <https://capacitorjs.com/docs/config>
- <https://stackoverflow.com/questions/64366729/android-studio-no-build-variant-found-error>
- <https://imstudio.medium.com/android-8-clear-text-http-traffic-not-permitted-73c1c9e3b803>
- <https://stackoverflow.com/questions/64366729/android-studio-no-build-variant-found-error>