**Steps for embed Sweepr app (android)**

# Prerequisites

- Install node

- npm install -g react-native-cli

Or Follow react native getting started macOS android or windows android

https://facebook.github.io/react-native/docs/getting-started


-AWS CLI

https://docs.aws.amazon.com/es_es/cli/latest/userguide/install-macos.html

- AWS configure with the Sweepr id and secret

https://docs.aws.amazon.com/es_es/cli/latest/userguide/cli-chap-configure.html

# Integrate Sweepr into a existing app

Our app support androidX only for make your app androidX in android studio go to refactor -> migrate to androidX will do the job.

1. Make a new directory. Make an android folder inner root directory where the existing app will live(Tmobile in this case) and in the root directory paste Sweepr code. You can merge our .gitignore with yours.

```
Root Folder
   |
   +-- android(Tmobile project)
   |
   +-- Sweepr Javascript Code
```

Files     Folder
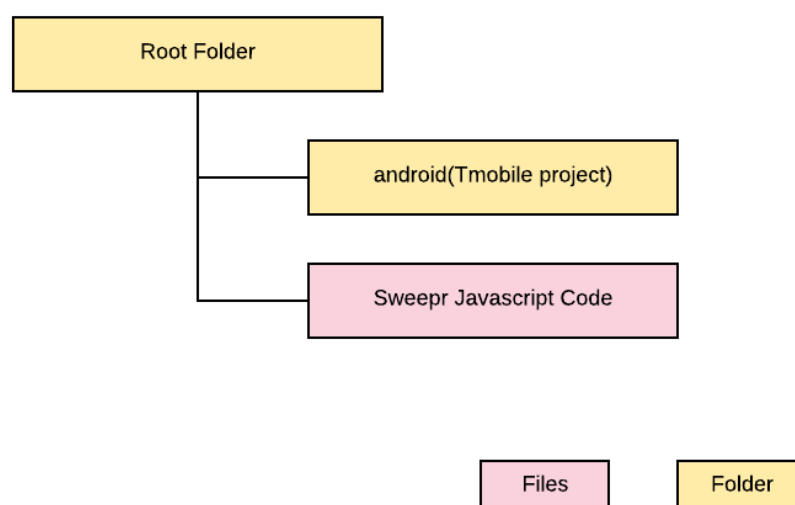
2 AWS dependencies

The scripts below use com.amazonaws.auth.profile.ProfileCredentialsProvider to grab the access key and secret key values from our local ~/.aws/credentialsfile. At time of writing Gradle isn't able to detect those credentials out of the box, so this is an easy workaround to avoid hardcoding them in the build script or duplicating them to another properties file. The format is

aws_access_key_id = AKIA3XVlGXUZTLM746OR

aws_secret_access_key = VpXER/bfHI1Rq8K9i14J5YeNMxDPaHdZQNE8VJAM


2. In the android project add to build.gradle(app) inner dependencies block

```
dependencies {

    implementation fileTree(dir: 'libs', include: ['*.jar'])

    implementation 'com.google.android.material:material:1.0.0'

    implementation "com.facebook.react:react-native:0.59.3"

    implementation 'com.google.code.gson:gson:2.8.5'

    implementation 'com.github.andriydruk:dnssd:0.9.13'

    implementation 'io.swagger:swagger-annotations:1.5.17'

    implementation 'com.squareup.okhttp:okhttp:2.7.5'

    implementation 'com.squareup.okhttp:logging-interceptor:2.7.5'

    implementation 'io.gsonfire:gson-fire:1.8.0'

    implementation 'org.threeten:threetenbp:1.3.5'

    implementation 'com.github.rosebud-iot:upnpdiscovery:1.0.5'

    implementation 'com.github.andriydruk:dnssd:0.9.13'

    implementation 'com.github.stealthcopter:AndroidNetworkTools:0.4.3'

    implementation 'org.eclipse.paho:org.eclipse.paho.client.mqttv3:1.1.0'

    implementation 'org.eclipse.paho:org.eclipse.paho.android.service:1.1.1'

    implementation 'jcifs:jcifs:1.3.17'

  implementation 'com.github.rosebud-iot:sweepr-android-mobile-framework:0.1.1'

    implementation 'com.github.rosebud-iot:sweepr-backendapi:0.1.2'


    testImplementation 'junit:junit:4.12'

    androidTestImplementation 'androidx.test:runner:1.1.0'
```

```
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.1.0'

}
```

- Inner android block

```
android {

    compileOptions {

        sourceCompatibility JavaVersion.VERSION_1_8

        targetCompatibility JavaVersion.VERSION_1_8

    }

  …

}
```
- In gradle.properties(project)

```
repoBucketName=sweepr-backendapi

mobileFrameworkBucketName=sweepr-android-mobile-framework
```

- In build.gradle(Android)

```
import com.amazonaws.auth.profile.ProfileCredentialsProvider

buildscript {

    repositories {

        google()

        jcenter()


    }

    dependencies {

        classpath 'com.android.tools.build:gradle:3.4.0'

        classpath "io.realm:realm-gradle-plugin:5.10.0"

        classpath "jp.classmethod.aws:gradle-aws-plugin:0.31"


        // NOTE: Do not place your application dependencies here; they belong
```

```
        // in the individual module build.gradle files

    }

}


ext {

    awsCredentials = new ProfileCredentialsProvider().credentials

}
```

- In the build.gradle(Android) inner all projects block

```
allprojects {
    repositories {

      maven {

      url "s3://${repoBucketName}/releases"

      credentials(AwsCredentials) {

            accessKey awsCredentials.AWSAccessKeyId

            secretKey awsCredentials.AWSSecretKey

      }

      }

      maven {

      url "s3://${mobileFrameworkBucketName}/releases"

      credentials(AwsCredentials) {

            accessKey awsCredentials.AWSAccessKeyId

            secretKey awsCredentials.AWSSecretKey

      }

      }

        google()

        jcenter()

        maven {
```

```
        url 'https://maven.google.com/'

        name 'Google'

    }

    maven{

        url "https://jitpack.io"

    }

    maven {

        url "https://repo.eclipse.org/content/repositories/paho-
snapshots/"

    }

    maven {

        url "$rootDir/../node_modules/react-native/android"

    }
    ...
  }
  ...
}
```

- In the top of android block paste

apply plugin: 'realm-android'

apply from: project(':react-native-config').projectDir.getPath() + "/
dotenv.gradle"

3.  Make sure that you have this changes in AndroidManifest.xml

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

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

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

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

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

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

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

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

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

```xml
<uses-permission android:name="android.permission.WAKE_LOCK" />
```

- Inner application tag

```xml
<application

    …

        tools:replace="android:appComponentFactory"

        android:appComponentFactory="whateverString"

        android:usesCleartextTraffic="true" tools:targetApi="28">

    <activity android:name="com.facebook.react.devsupport.DevSettingsActivity"
/>

    <activity

        android:name=".ReactIntegrationActivity"

    android:label="@string/app_name"

    android:theme="@style/Theme.AppCompat.Light.NoActionBar"></activity>

…

</application>
```

4.  As a test, in your android project create a ReactIntegrationActivity and paste this content

```java
public class ReactIntegrationActivity extends AppCompatActivity implements
DefaultHardwareBackBtnHandler {

    private ReactRootView mReactRootView;

    private final int OVERLAY_PERMISSION_REQ_CODE = 1;

    public ReactInstanceManager mReactInstanceManager;


    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);


        mReactRootView = new ReactRootView(this);

        mReactInstanceManager = ReactInstanceManager.builder()

                .setApplication(getApplication())
```

```java
                .setCurrentActivity(this)

                .setBundleAssetName("index.android.bundle")

                .setJSMainModulePath("index")

                .addPackages(this.getPackages())

                .setUseDeveloperSupport(BuildConfig.DEBUG)

                .setInitialLifecycleState(LifecycleState.RESUMED)

                .build();
        // The string here (e.g. "MyReactNativeApp") has to match

        // the string in AppRegistry.registerComponent() in index.js

        mReactRootView.startReactApplication(mReactInstanceManager, "App",
null);


        setContentView(mReactRootView);

    }


    @Override

    protected void onActivityResult(int requestCode, int resultCode, Intent
data) {

        if (requestCode == OVERLAY_PERMISSION_REQ_CODE) {

            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

                if (!Settings.canDrawOverlays(this)) {

                    // SYSTEM_ALERT_WINDOW permission not granted

                    Toast.

                        makeText(this, "You cannot open the React Native
app as you have denied the permission", Toast.LENGTH_SHORT).show();

                }

            }

        }

        mReactInstanceManager.onActivityResult( this, requestCode, resultCode,
data );

    }
```

```java
@Override

public void invokeDefaultOnBackPressed() {

    super.onBackPressed();

}


@Override

protected void onPause() {

    super.onPause();


    if (mReactInstanceManager != null) {

        mReactInstanceManager.onHostPause(this);

    }

}


@Override

protected void onResume() {

    super.onResume();


    if (mReactInstanceManager != null) {

        mReactInstanceManager.onHostResume(this, this);

    }

}


@Override

protected void onDestroy() {

    super.onDestroy();


    if (mReactInstanceManager != null) {
```

```
            mReactInstanceManager.onHostDestroy(this);

        }

        if (mReactRootView != null) {

            mReactRootView.unmountReactApplication();

        }

    }



    @Override

    public void onBackPressed() {

        if (mReactInstanceManager != null) {

            mReactInstanceManager.onBackPressed();

        } else {

            super.onBackPressed();

        }

    }



    @Override

    public boolean onKeyUp(int keyCode, KeyEvent event) {

        if (keyCode == KeyEvent.KEYCODE_MENU && mReactInstanceManager != null)
{

            mReactInstanceManager.showDevOptionsDialog();

            return true;

        }

        return super.onKeyUp(keyCode, event);

    }



    protected List<ReactPackage> getPackages() {

        return Arrays.<ReactPackage>asList(

                new MainReactPackage(),
```

```java
            new AsyncStoragePackage(),

            new NetInfoPackage(),

            new ReactNativeConfigPackage(),

            new RNGestureHandlerPackage(),

            new CallDetectionManager(getApplication()),

            new ReactNativePushNotificationPackage(),

            new VoicePackage(),

            new SvgPackage(),

            new VectorIconsPackage(),

            new SweeprCloudPackage(),

            new SweeprScanPackage(),

            new SweeprResolutionPackage(),

            new PingTestPackage()
        );
    }
}
```

5. In root directory execute npm run complete-install

6. Where you want to integrate our app you should

```java
Intent intent = new Intent(MainActivity.this, ReactIntegrationActivity.class);
startActivity(intent);
```

Troubleshotting

If you got errors in android studio try to delete .idea from the project and sync and build again.