

José Carlos Sequeira Alvarado SA211247

DPS07L

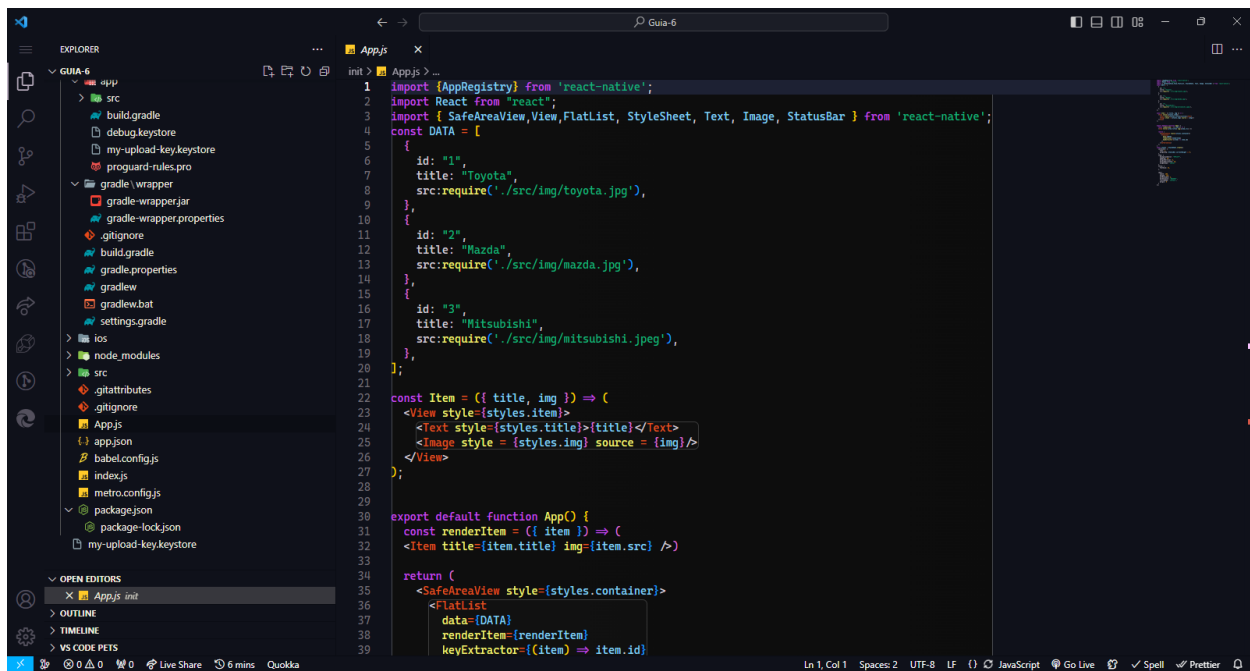
Guia 4

Aplicación que muestra 3 diferentes marcas de automóviles

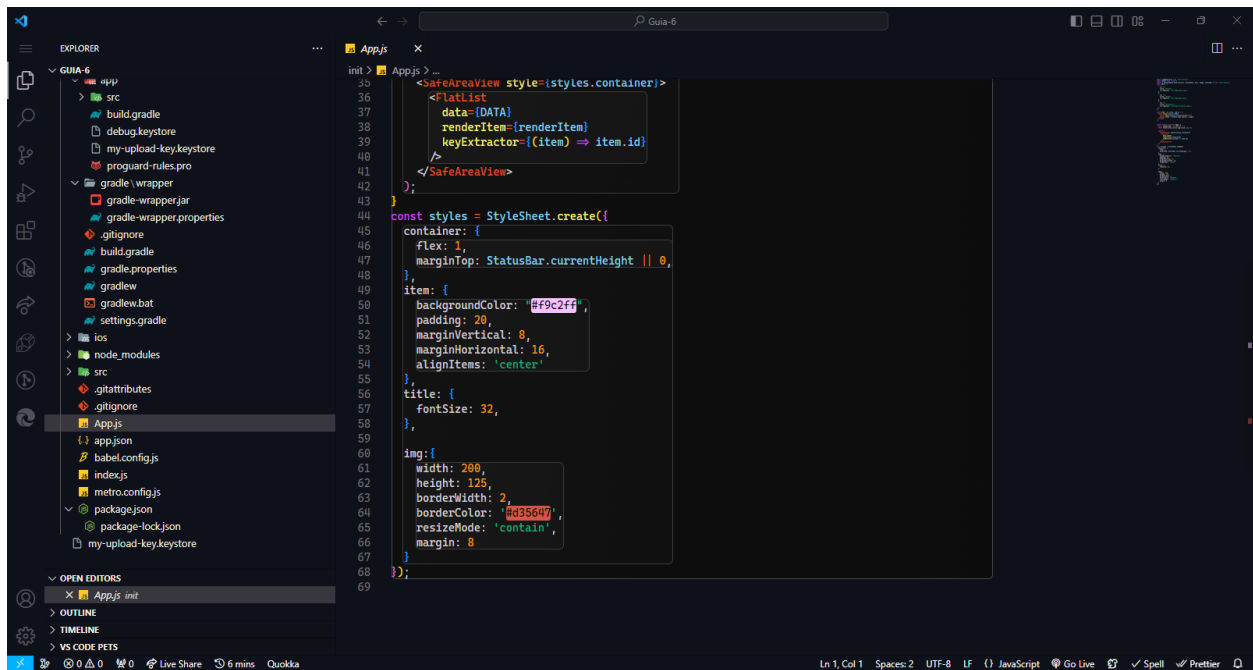
<https://github.com/Sequeiraa/DPS07L/tree/master/Semana%206/Guia-4>

Capturas de código:

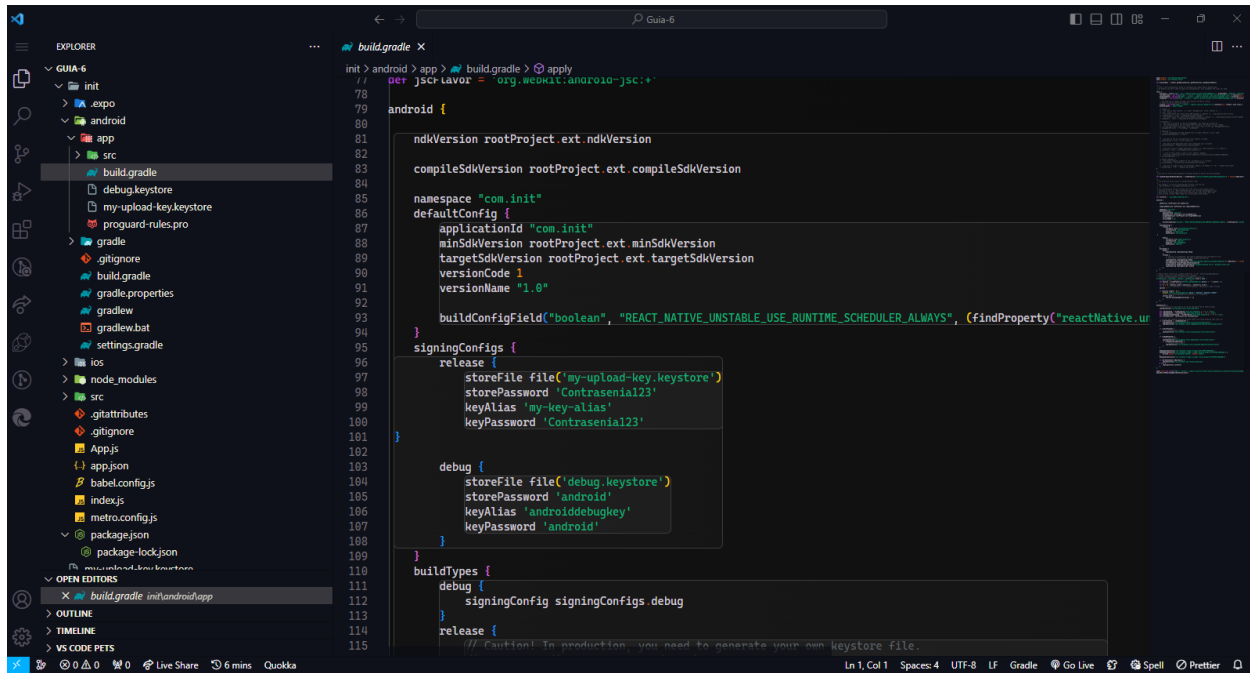
App.js:



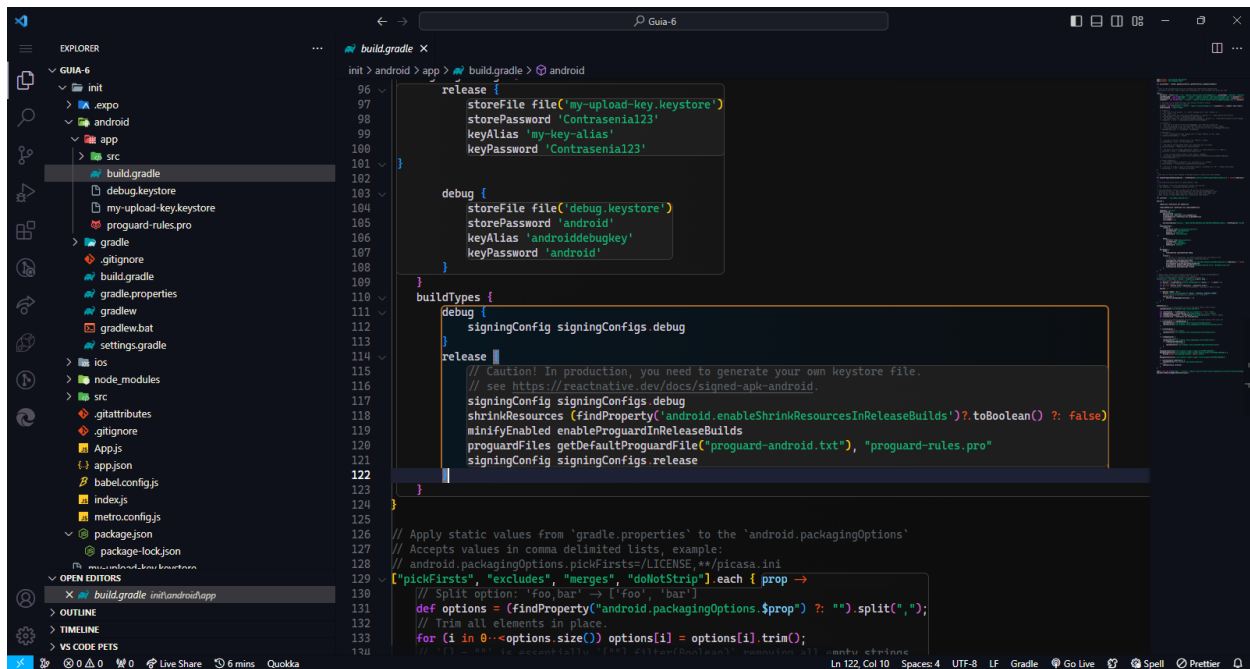
```
1 import {AppRegistry} from 'react-native';
2 import React from 'react';
3 import { SafeAreaView, View, FlatList, StyleSheet, Text, Image, StatusBar } from 'react-native';
4 const DATA = [
5   {
6     id: "1",
7     title: "Toyota",
8     src: require('./src/img/toyota.jpg'),
9   },
10  {
11    id: "2",
12    title: "Mazda",
13    src: require('./src/img/mazda.jpg'),
14  },
15  {
16    id: "3",
17    title: "Mitsubishi",
18    src: require('./src/img/mitsubishi.jpeg'),
19  },
20 ];
21
22 const Item = ({ title, img }) => (
23   <View style={styles.item}>
24     <Text style={styles.title}>{title}</Text>
25     <Image style={styles.img} source = {img}/>
26   </View>
27 );
28
29 export default function App() {
30   const renderItem = ({ item }) => (
31     <Item title={item.title} img={item.src} />
32   );
33
34   return (
35     <SafeAreaView style={styles.container}>
36       <FlatList
37         data={DATA}
38         renderItem={renderItem}
39         keyExtractor={item => item.id}
40       />
41     </SafeAreaView>
42   );
43 }
```



Carpeta: app – src – build.gradle:



```
init > android > app > build.gradle > apply
// jsc: flavor = 'org.webkit.android-jsc:'
android {
    ndkVersion rootProject.ext.ndkVersion
    compileSdkVersion rootProject.ext.compileSdkVersion
    namespace "com.init"
    defaultConfig {
        applicationId "com.init"
        minSdkVersion rootProject.ext.minSdkVersion
        targetSdkVersion rootProject.ext.targetSdkVersion
        versionCode 1
        versionName "1.0"
        buildConfigField("boolean", "REACT_NATIVE_UNSTABLE_USE_RUNTIME_SCHEDULER_ALWAYS", (findProperty("reactNative.unstableUseRuntimeSchedulerAlways") ?: true).toString())
    }
    signingConfigs {
        release {
            storeFile file("my-upload-key.keystore")
            storePassword 'Contrasenial23'
            keyAlias 'my-key-alias'
            keyPassword 'Contrasenial23'
        }
        debug {
            storeFile file("debug.keystore")
            storePassword 'android'
            keyAlias 'androiddebugkey'
            keyPassword 'android'
        }
    }
    buildTypes {
        debug {
            signingConfig signingConfigs.debug
        }
        release {
            // Caution! In production, you need to generate your own keystore file.
            // See https://developer.android.com/studio/run/debug.html for more details.
        }
    }
}
```



```
init > android > app > build.gradle > android
96 release {
97     storeFile file("my-upload-key.keystore")
98     storePassword 'Contrasenial23'
99     keyAlias 'my-key-alias'
100     keyPassword 'Contrasenial23'
101 }
102
103 debug {
104     storeFile file("debug.keystore")
105     storePassword 'android'
106     keyAlias 'androiddebugkey'
107     keyPassword 'android'
108 }
109
110 buildTypes {
111     debug {
112         signingConfig signingConfigs.debug
113     }
114     release {
115         // Caution! In production, you need to generate your own keystore file.
116         // See https://reactnative.dev/docs/signed-apk-android.
117         signingConfig signingConfigs.debug
118         shrinkResources (findProperty("android.enableShrinkResourcesInReleaseBuilds")?.toBoolean() ?: false)
119         minifyEnabled enableProguardInReleaseBuilds
120         proguardFiles getDefaultProguardFile("proguard-android.txt"), "proguard-rules.pro"
121         signingConfig signingConfigs.release
122     }
123 }
124
125 // Apply static values from 'gradle.properties' to the 'android.packagingOptions'
126 // Accepts values in comma delimited lists, example:
127 // android.packagingOptions.pickFirstsWith=LICENSE,**/picasa.ini
128 // ["pickFirstsWith", "excludes", "merges", "doNotStrip"].each { prop ->
129 //     // Split option: 'foo,bar' -> ['foo', 'bar']
130     def options = (findProperty("android.packagingOptions.$prop") ?: "").split(",");
131     // Trim all elements in place.
132     for (i in 0..options.size()-1) options[i] = options[i].trim();
133     // (i) = "" is essentially (i) == false, so we can use it to remove all empty strings
134 }
```

Aplicación en funcionamiento:

