**White Label**

White labelling is the process of creating different products using the same template. In the mobile development world, this means creating a project that has one code base but can produce many apps all with different:

* Modules
* Content
* Theme
* App Id
* App Icon

**Getting Started:**

To begin, we will create a React Native project. Let’s call it **whitelabel**

**Step 1:**

Install **react-native-config** : npm i react-native-config

**Step 2:**

Manual link (Android)

* Add that code in **Android/setting.gradle**

include ':react-native-config'

project(':react-native-config').projectDir = new File(rootProject.projectDir, '../node\_modules/react-native-config/android')

File will be like that

rootProject.name = 'whitelabel'

apply from: file("../node\_modules/@react-native-community/cli-platform-android/native\_modules.gradle"); applyNativeModulesSettingsGradle(settings)

include ':app'

include ':react-native-config'

project(':react-native-config').projectDir = new File(rootProject.projectDir, '../node\_modules/react-native-config/android')

* Add that code in **Android/app/build.gradle**

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

in first file after

apply plugin: "com.android.application"

to apply plugin

-Go to dependencies and add

implementation project(':react-native-config')

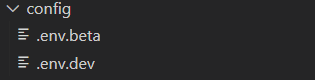
after

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

**Step 3:**

Setup for android:

* **Create two files in folder called config**



.env.beta will be:

ENV=beta

APPLICATION\_ID=com.beta

Application\_Version=1.1.0

APP\_NAME = beta

.env.dev will be:

ENV=dev

APPLICATION\_ID=com.dev

Application\_Version=1.1.1

APP\_NAME = dev

* Now we need to define envConfigFiles in **android/app/build.gradle** like that:

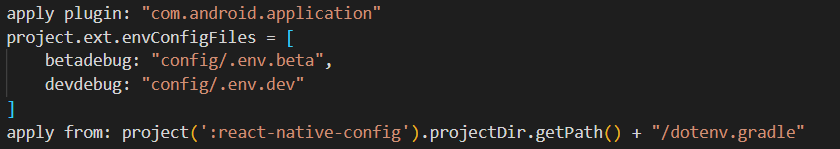
project.ext.envConfigFiles = [

    betadebug: "config/.env.beta",

    devdebug: "config/.env.dev"

]

File will be like that:



* Add

flavorDimensions "default"

before **defaultConfig**

and change defaultConfig to be like that:

    defaultConfig {

        applicationId project.env.get("APP\_ID")

        minSdkVersion rootProject.ext.minSdkVersion

        targetSdkVersion rootProject.ext.targetSdkVersion

        versionCode 1

        versionName project.env.get("Application\_Version")

    }

* Now we need adding product flavors:

productFlavors {

dev

{

minSdkVersion rootProject.ext.minSdkVersion

applicationId 'com.dev'

targetSdkVersion rootProject.ext.targetSdkVersion

resValue "string", "build\_config\_package", "com.whitelabel"

}

beta

{

minSdkVersion rootProject.ext.minSdkVersion

applicationId 'com.beta'

targetSdkVersion rootProject.ext.targetSdkVersion

resValue "string", "build\_config\_package", "com.whitelabel"

}

}

* **And now you can create 2 apps with different App Id and different version**

**Step 4:**

**To make app name dynamic**

* Go to **android/app/src/main/values/string.xml** and will be like that:

<resources>

    <string name="app\_name">@string/APP\_NAME</string>

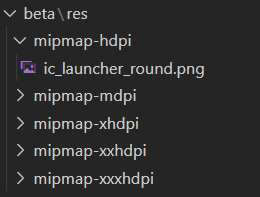
</resources>

* **And now you can create two apps with different name**

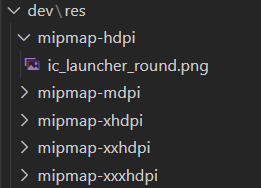
**Step 5:**

**Custom icons for each environment**

* Create folder in **android/app/src** called beta and will be like that



* Create folder in **android/app/src** called dev and will be like that



* **And now you can create two apps with different Icons**

**Step 6:**

**If you need change app theme or any content you can use app name in config to change values**

* **Command To Run:**

**Beta:**

npx react-native run-android --variant betadebug --appId com.beta

**Dev:**

npx react-native run-android --variant devdebug --appId com.dev