



# Unimodules Config

Configuração do pacote React Native Unimodules

[Implementação em vídeo](#)

License MIT

## Instalação do pacote

O primeiro passo deve ser a instalação do pacote no seu projeto, para isso, basta rodar o comando abaixo via terminal, dentro da pasta do seu projeto:

```
yarn add react-native-unimodules
```

## IOS Config

Para prosseguir com a instalação do ios, precisamos alterar manualmente alguns arquivos, são eles:

ios/Podfile

1. No item **platform :ios**, mudar de **9.0** para **10.0**
2. Após o **require\_relative**, da linha 2, adicione uma nova linha abaixo, com o seguinte código:

```
require_relative '../node_modules/react-native-unimodules/cocoapods.rb'
```

3. Após o **use\_native\_modules!**, da linha 43, adicione o seguinte código:

```
use_unimodules!
```

ios/NOME\_DO\_SEU\_PROJETO/AppDelegate.h

1. Após o código **#import <React/RCTBridgeDelegate.h>**, da linha 9, adicione uma linha com o seguinte código:

```
#import <UMReactNativeAdapter/UMModuleRegistryAdapter.h>
```

2. Antes do `@property`, da linha 14, adicione uma linha anterior, com o seguinte código:

```
@property (nonatomic, strong) UModuleRegistryAdapter
*moduleRegistryAdapter;
```

ios/NOME\_DO\_SEU\_PROJETO/AppDelegate.m

1. Apague o código `#import <React/RCTBridge.h>`, da linha 8.
2. Após o código `#import <React/RCTRootView.h>`, da linha 11 (ou linha 12), adicione o seguinte código:

```
#import <UMCore/UMModuleRegistry.h>
#import <UMReactNativeAdapter/UMNativeModulesProxy.h>
#import <UMReactNativeAdapter/UMModuleRegistryAdapter.h>
```

3. Após a abertura das chaves do `– (BOOL)application:`, logo antes do `RCTBridge *bridge`, adicione a seguinte linha de código:

```
self.moduleRegistryAdapter = [[UModuleRegistryAdapter alloc]
initWithModuleRegistryProvider:[UModuleRegistryProvider alloc] init]];
```

4. Antes do `– (NSURL *)`, da linha 37, adicione o seguinte código:

```
– (NSArray<id<RCTBridgeModule>> *)extraModulesForBridge:(RCTBridge
*)bridge
{
    NSArray<id<RCTBridgeModule>> *extraModules = [_moduleRegistryAdapter
extraModulesForBridge:bridge];
    return extraModules;
}
```

5. Mude o `#if DEBUG`, da linha 45, para `#ifdef DEBUG`

Info.plist

1. Após o `<dict>`, da linha 4, adicione o seguinte código:

```
<key>NSCalendarsUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to access your calendar</string>
```

```
<key>NSCameraUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to use the camera</string>
<key>NSContactsUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to access your contacts</string>
<key>NSLocationAlwaysAndWhenInUseUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to use your location</string>
<key>NSLocationAlwaysUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to use your location</string>
<key>NSLocationWhenInUseUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to use your location</string>
<key>NSMicrophoneUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to access your microphone</string>
<key>NSMotionUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to access your device's
accelerometer</string>
<key>NSPhotoLibraryAddUsageDescription</key>
<string>Give $(PRODUCT_NAME) permission to save photos</string>
<key>NSPhotoLibraryUsageDescription</key>
<string>Give $(PRODUCT_NAME) permission to access your photos</string>
<key>NSRemindersUsageDescription</key>
<string>Allow $(PRODUCT_NAME) to access your reminders</string>
```

## Configuração Final e Teste no IOS

1. Entre na pasta ios, via terminal, e rode o comando `pod install`.
2. Para finalizar, volte para a pasta principal do seu projeto e rode o comando `react-native run-ios`.

**Caso o app inicie normalmente, a configuração do IOS foi um sucesso!**

## Android

Agora precisamos alterar alguns arquivos relacionados as configuração no android, são eles:

android/settings.gradle

1. Logo após a linha contendo o `apply from:`, da linha 2, adicione as 2 linhas de código abaixo:

```
apply from: "../node_modules/react-native-unimodules/gradle.groovy"
includeUnimodulesProjects()
```

android/build.gradle

1. No item **minSdkVersion**, altere o valor de `16` para `21`.

android/app/build.gradle

Não confundir esse arquivo com o arquivo **build.gradle** da pasta **android/**!

1. Na linha 85, logo após a linha contendo o `apply from:`, adicione a seguinte linha de código:

```
apply from: "../../node_modules/react-native-unimodules/gradle.groovy"
```

2. Dentro das chaves do `dependencies {`, que deve estar por volta da linha 182, adicione o seguinte código:

```
addUnimodulesDependencies()
```

## MainApplication.java

1. Logo após o `package com.uni;`, da linha 1, adicione os seguintes *imports*:

```
import com.uni.generated.BasePackageList;
import org.unimodules.adapters.react.ModuleRegistryAdapter;
import org.unimodules.adapters.react.ReactModuleRegistryProvider;
import org.unimodules.core.interfaces.SingletonModule;
import java.util.Arrays;
```

2. Logo após a abertura das chaves da `public class MainApplication extends Application implements ReactApplication {`, da linha 18, adicione o seguinte código:

```
private final ReactModuleRegistryProvider mModuleRegistryProvider = new
ReactModuleRegistryProvider(new BasePackageList().getPackageList(),
Arrays.<SingletonModule>asList());
```

3. Logo antes do `return packages;`, da linha 34, adicione o seguinte código:

```
List<ReactPackage> unimodules = Arrays.<ReactPackage>asList(
    new ModuleRegistryAdapter(mModuleRegistryProvider)
);
packages.addAll(unimodules);
```

## Teste no Android

1. Com todas as configurações feitas, vá até a pasta principal do seu projeto e rode o comando `react-native run-android`.

**Caso o app inicie normalmente, a configuração do Android foi um sucesso!**

## Autor



Carlos Levir

## Licença

This project is licensed under the MIT License - see the [LICENSE](#) page for details.