

Bem meus amigos, hoje resolvi voltar um pouco as raízes e com isso fazer algo um pouco diferente.

E o que seria diferente?

Montar a minha primeira app em React Native e de cara já integrá-lo ao Dynatrace para acompanhar todas as métricas do usuário ☺

Incrível não é mesmo

Mãos a massa então Sayajin

Primeiro como vou focar na questão de integração então parto do pressuposto que você já tenha um aplicativo em React Native criado, caso não tenha tem um vídeo legal do Canal [Programador BR](#) no YouTube, então segue o pessoal e monta a sua App beleza!

Inicialmente vamos precisar instalar o plugin do Dynatrace na aplicação.

No terminal na pasta raiz do app (no meu caso estou usando o terminal a partir do vscode).

Digite: npm i @dynatrace/react-native-plugin

```
PS D:\projetos\react-native\AppDyna> npm i @dynatrace/react-native-plugin
```

Aguarde até terminar o download do plugin.

```
+ @dynatrace/react-native-plugin@1.192.2
added 191 packages from 98 contributors, removed 70 packages, updated 994 packages and audited 334975 packages in 182.933s

42 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Exemplo da saída

Versões

Para versão: 0.60.0 ou superior: npm install @dynatrace/react-native-plugin

Versão inferior a 0.60.0: react-native install @dynatrace/react-native-plugin

No meu exemplo estou na 0.62

Na sequência rode o comando: npm install @dynatrace/react-native-plugin

```
+ @dynatrace/react-native-plugin@1.192.2
removed 2 packages, updated 8 packages and audited 334975 packages in 34.752s

39 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Exemplo da saída

Registrar o Dynatrace no transformer

No diretório raiz do seu projeto, crie ou estenda (\geq RN 0.59) metro.config.js (\geq RN 0.57 – 0.58) rn-cli.config.js para que ele contenha a propriedade transformer.babelTransformerPath.

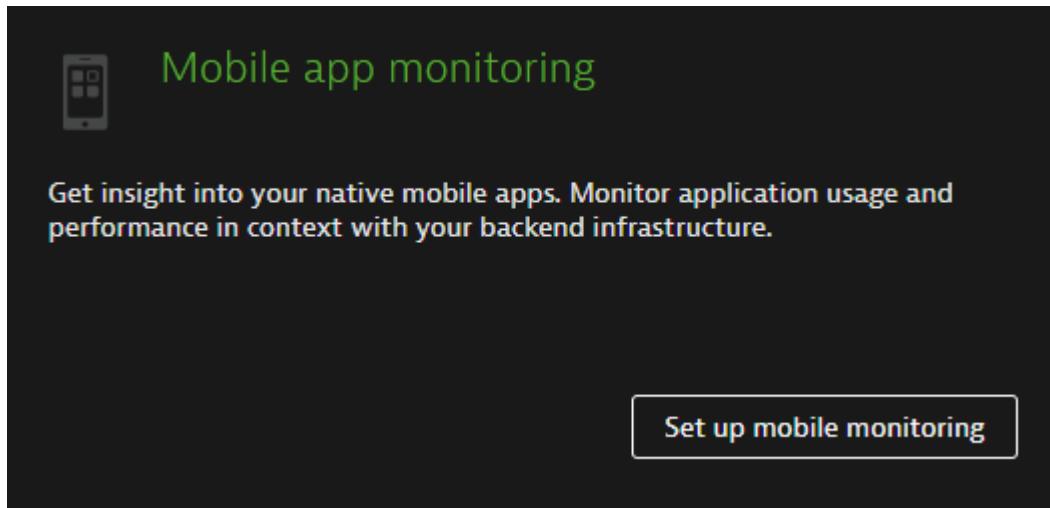
Abra o metro.config.js e adicione o Dynatrace conforme indicado abaixo:



```
JS index.js
JS metro.config.js
npm package-lock.json
node_modules
babelTransformerPath: require.resolve('@dynatrace/react-native-plugin/lib/dynatrace-transformer')
reporter: require("@dynatrace/react-native-plugin/lib/dynatrace-reporter")
module.exports = {
  transformer: {
    babelTransformerPath: require.resolve(
      '@dynatrace/react-native-plugin/lib/dynatrace-transformer',
    ),
    getTransformOptions: async () => ({
      transform: {
        experimentalImportSupport: false,
        inlineRequires: false,
      },
    }),
  },
  reporter: require('@dynatrace/react-native-plugin/lib/dynatrace-reporter'),
};
```

Por fim agora precisamos obter o arquivo com as definições da própria instalação do seu Dynatrace.

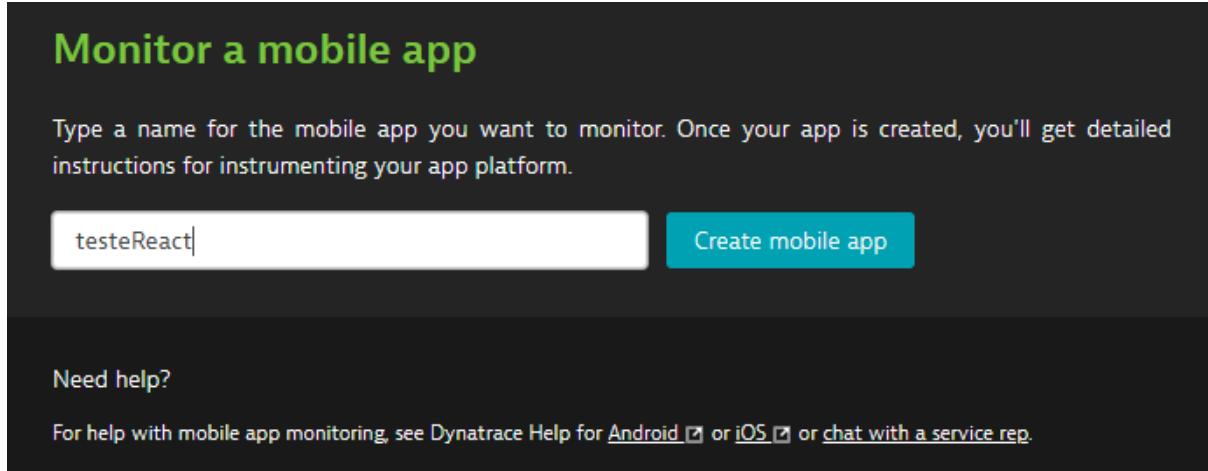
No dynatrace, vá até Deploy e em Mobile app monitoring



The image shows the mobile app monitoring landing page. It features a smartphone icon on the left, followed by the title "Mobile app monitoring". Below the title is a subtitle: "Get insight into your native mobile apps. Monitor application usage and performance in context with your backend infrastructure." At the bottom right of the main content area is a button labeled "Set up mobile monitoring".

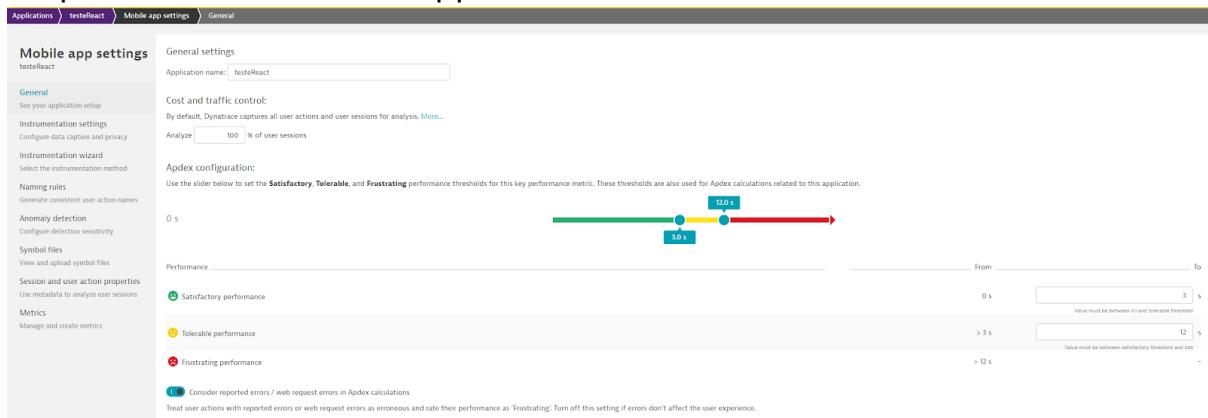
Clique em Set up mobile monitoring

Coloque um nome para o aplicativo no meu exemplo deixei testeReact



The image shows the "Monitor a mobile app" screen. It has a text input field containing "testeReact" and a blue "Create mobile app" button. Below the input field, there's a "Need help?" link and a note about getting help from Dynatrace Help for Android or iOS or chat with a service rep.

Clique em Create mobile app



The image shows the "Mobile app settings" screen for the "testeReact" application. On the left, there's a sidebar with options like General, Instrumentation settings, Instrumentation wizard, Naming rules, Anomaly detection, Symbol files, Session and user action properties, and Metrics. The main panel shows "General settings" with an "Application name" field set to "testeReact". It also includes sections for "Cost and traffic control" (with a note about Dynatrace capturing all user actions and sessions), "Apdex configuration" (with a slider set at 0.9), and "Performance" (with sliders for "Satisfactory performance" at 0 s and "Tolerable performance" at 3 s). There are also checkboxes for "Consider reported errors / web request errors in Apdex calculations" and "Treat user actions with reported errors or web request errors as erroneous and rate their performance as 'Frustrating'".

Essa é a tela inicial, aqui é possível definir *thresholds* da aplicação e quantidade de dados capturados por Default vem definido em 100% aqui recomendo ajustar

conforme a necessidade do negócio, atente se que quanto maior a captura maior o custo.

Em seguida clique em Instrumentation wizard

The screenshot shows the 'Mobile app settings' screen for a project named 'testeReact'. On the left, there's a sidebar with options: General (with 'See your application setup'), Instrumentation settings (with 'Configure data capture and privacy'), and Instrumentation wizard (which is selected, indicated by a grey background). Under 'Instrumentation wizard', it says 'Select the instrumentation method'. On the right, the title 'Instrumentation wizard' is displayed with the sub-instruction 'Please select the platform you want to use'. Below this are five icons representing different platforms: Android (an Android robot icon), iOS (an iPhone icon with 'iOS' text), tvOS (a television icon with 'tvOS' text), Cordova (a blue hexagon icon with a white phone-like shape), and React Native (a blue atom-like icon with 'React Native' text).

Em seguida em React Native

Em seguida o que interessa para nós é o terceiro passo

3 Setup dynatrace.config.js

You need to place the config file into the root directory of your project next to app.json. If the file is not available, the auto-instrumentation will not work.

[Download dynatrace.config.js](#)

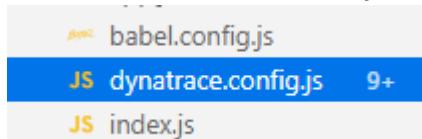
Clique em Download dynatrace.config.js para fazer o Download do Arquivo.

Importe!

O arquivo já vem configurado para o seu enviroment do Dynatrace na qual já está definido o nome da aplicação e informações da instância da ferramenta.

Copie o arquivo para a raiz do projeto.

NOTA: Caso o mesmo já exista devido a instalação do plugin apenas substitua.



Ok, vamos o build e rodar o app

Digite no terminal: npm run instrumentDynatrace

Em seguida: npx react-native run-android

Aguarde até terminar

BUILD SUCCESSFUL in 1m 40s

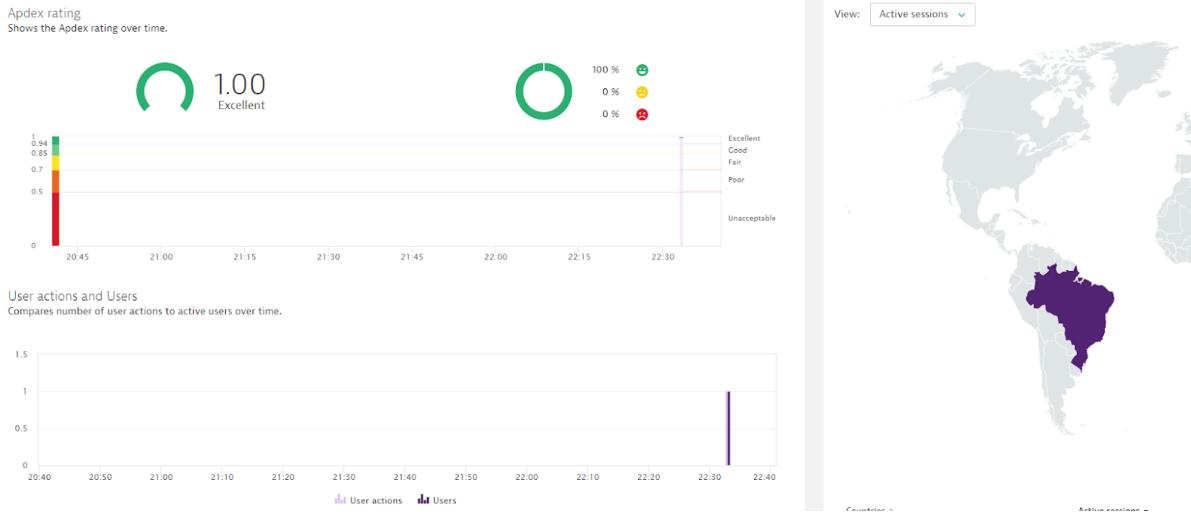
Finalizado, agora o seu aplicativo já está configurado.

Vamos ver os resultados.

Em aplicações na console Dynatrace localize a App que criou

Name	Apdex	Actions
testeReact	Excellent	<0.01 /min

Acesse e veja os dados aparecendo.



Agora é só aguardar.

Porém ainda é possível melhorar o Dynatrace permite adicionar ações personalizada para o aplicativo e ir muito além em suas análises.

Porém esse ponto vou deixar para outro tutorial futuro.

Espero que tenha contribuído mais um pouco com a comunidade e qualquer dúvida podem perguntar

Abraços e sucesso pessoal.