



Introdução

## ANDROID & IOS (MANAGED)

Windows

Linux

macOS

## ANDROID (BARE & RN CLI)

Windows

Linux

macOS

## IOS (BARE & RN CLI)

macOS

## DISPOSITIVOS VIRTUAIS

Emulador Android

Simulator iOS

# Windows

## ON THIS PAGE

[Intro](#)

[Chocolatey](#)

[JDK 11 \(LTS\).](#)

[Android Studio](#)

[Preparativos](#)

[Instalação](#)

[Configuração](#)

[Windows Hypervisor Platform - WHPX \(Extra\)](#)

[Emulador](#)

[Dispositivo Físico](#)

## Android

Configurando o ambiente  
de desenvolvimento Android

## [Intro](#)

Nesse guia iremos assumir que você já possui o ambiente básico do React Native (Expo Managed Workflow). As dependências desse ambiente são:

- Node.js (LTS);
- npm (já vem instalado com o Node);
- git
- expo-cli local
- Expo GO (app a ser instalado no dispositivo Android e/ou iOS)

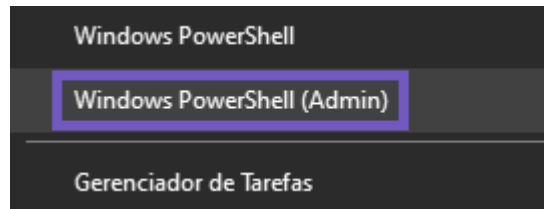
Caso você não possua esse ambiente,  **siga esses passos**

Seguindo para configuração do ambiente Android no Windows utilizando Expo Bare Workflow ou React Native CLI, iremos realizar 3 instalações principais:

- Chocolatey;
- JDK 11 (LTS);
- Android Studio e dependências.

## Chocolatey

Para instalar a JDK no Windows, vamos utilizar um gerenciador de pacotes do Windows chamado Chocolatey. Esse gerenciador nos possibilita instalar dependências e ferramentas no sistema com poucos comandos e tudo pelo terminal. Execute o Powershell como administrador utilizando a tecla Windows + X ou clicando com o botão direito sobre o botão "Iniciar":



Execute o comando abaixo para verificar se você possui permissões para instalar dependências com o terminal:

```
BASH
```

[Copy](#)

```
Get-ExecutionPolicy
```

Caso o retorno desse comando seja **diferente** de **"RemoteSigned"** ou **"Unrestricted"**, execute o seguinte comando em seu terminal:

```
BASH
```

[Copy](#)

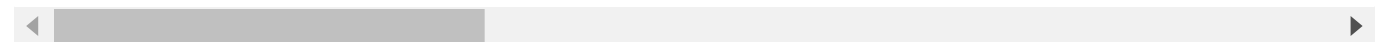
```
Set-ExecutionPolicy RemoteSigned
```

Agora, execute o seguinte comando para instalar o Chocolatey:

```
BASH
```

[Copy](#)

```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::Se
```



Agora, teste se a instalação ocorreu corretamente executando o seguinte comando no seu terminal (nada irá acontecer, mas não deve retornar erros):

Nesse passo pode ser necessário reiniciar seu terminal

```
BASH
```

```
choco
```

Copy

## JDK 11 (LTS).

Caso esteja com o JDK instalado em sua máquina, certifique-se que sua versão seja exatamente a versão 11.

```
BASH
```

```
choco install -y openjdk11
```

Copy

Agora reinicie o Powershell e execute o seguinte comando no seu terminal:

```
BASH
```

```
java -version
```

Copy

Se apresentar a versão do Java, sua instalação foi um sucesso.

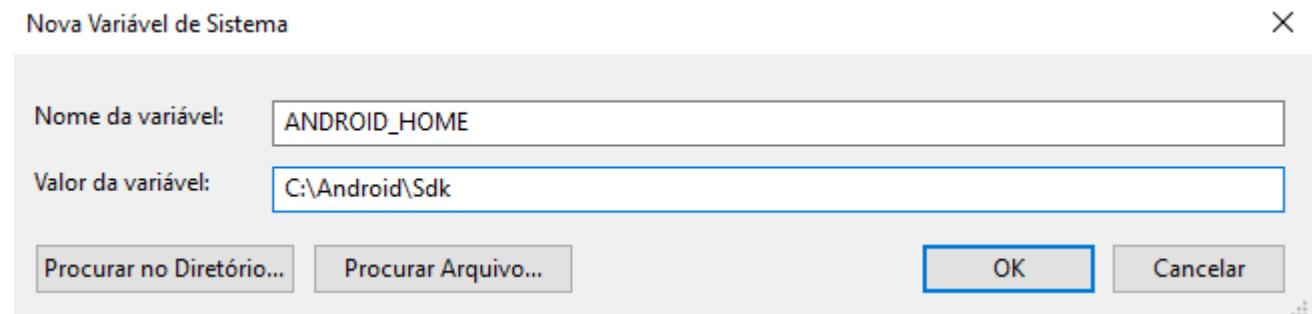
## Android Studio

### Preparativos

Crie uma pasta em um local desejado para instalação da SDK (Ex.: `C:\Android\Sdk`). É  **muito** importante que esse caminho não possua **espaços ou caracteres especiais** pois irá causar erros.

### Anote esse caminho para ser utilizado posteriormente

Agora, no Painel de Controle do Windows, abra o item "Sistema e Segurança" ou "Sistema", clique em "Configurações avançadas do sistema", selecione "Variáveis de ambiente" e clique no botão "Nova variável de ambiente", indique o nome da variável como `ANDROID_HOME`, adicione o caminho utilizado acima (Ex.: `C:\Android\Sdk`) como segundo parâmetro e clique em OK.



Faça o mesmo processo para criar a variável **JAVA\_HOME**. Mas primeiramente temos que obter o caminho de instalação.

A instalação `openjdk11` com o Chocolatey pode acabar criando essa variável automaticamente para você. Caso isso tenha ocorrido, confira se o caminho está certo e, caso sim, pode pular para a configuração da variável `PATH` (algumas linhas abaixo)

Para obter o caminho, execute o comando abaixo no Powershell:

BASH

Copy

```
Get-Command java
```

Ele deve retornar o caminho da sua instalação. Informe esse valor **excluindo** o final `bin/java.exe` no momento da criação da **JAVA\_HOME**. Por exemplo, se o comando retornar o caminho `C:\Program Files\Eclipse Adoptium\jdk-11.0.17.8-hotspot\bin\java.exe`, crie a variável **JAVA\_HOME** com o valor `C:\Program Files\Eclipse Adoptium\jdk-11.0.17.8-hotspot`.

Na mesma janela de "Variáveis de ambiente" no Windows, clique na variável **PATH** e então em "Editar". Haverá uma lista de caminhos e você deve adicionar esses quatro novos caminhos no fim da lista:

BASH

Copy

```
%ANDROID_HOME%\emulator  
%ANDROID_HOME%\tools  
%ANDROID_HOME%\tools\bin  
%ANDROID_HOME%\platform-tools
```

Se sua versão do Windows for menor que a 10 **sempre** coloque ponto-e-virgula ";" no final dos caminhos para eles não se juntarem

## Instalação



Acesse a página do **Android Studio** e clique no botão **Download Android Studio**.

Caso seu Windows seja a versão **32 bits**, será baixada a opção **.zip** que não vem com o instalador **.exe**. Basta extrair a pasta **android-studio** em um local de preferência (Ex.: `C:\`) e executar o arquivo **studio.exe** dentro da pasta **bin**. Feito isso, você pode pular até a próxima **seção**

Vá até a pasta onde o arquivo foi salvo e execute o instalador.

A primeira janela que deve aparecer é para escolher o que vai ser instalado. Por padrão, a opção **Android Studio** já vem selecionada. Selecione também a opção **Android Virtual Device** e clique em **Next**.

Na sequência, será perguntado sobre o local de instalação do Android Studio. Pode deixar o caminho padrão e clicar em **Next**.

Em seguida, será perguntado sobre a pasta no menu Iniciar. Deixe o padrão e clique em **Install**.

Nessa etapa será realizada instalação. Quando terminar, clique em **Next**.

Por fim, será apresentada a janela de fim da instalação. Deixe a opção **Start Android Studio** marcada e clique em **Finish**.

## Configuração

Com o Android Studio instalado, chegou a hora de fazer a configuração inicial do programa.

A primeira janela a ser apresentada deve ser perguntando sobre a importação de configurações de outro Android Studio. Selecione a opção **Do not import settings** e clique em **OK**.

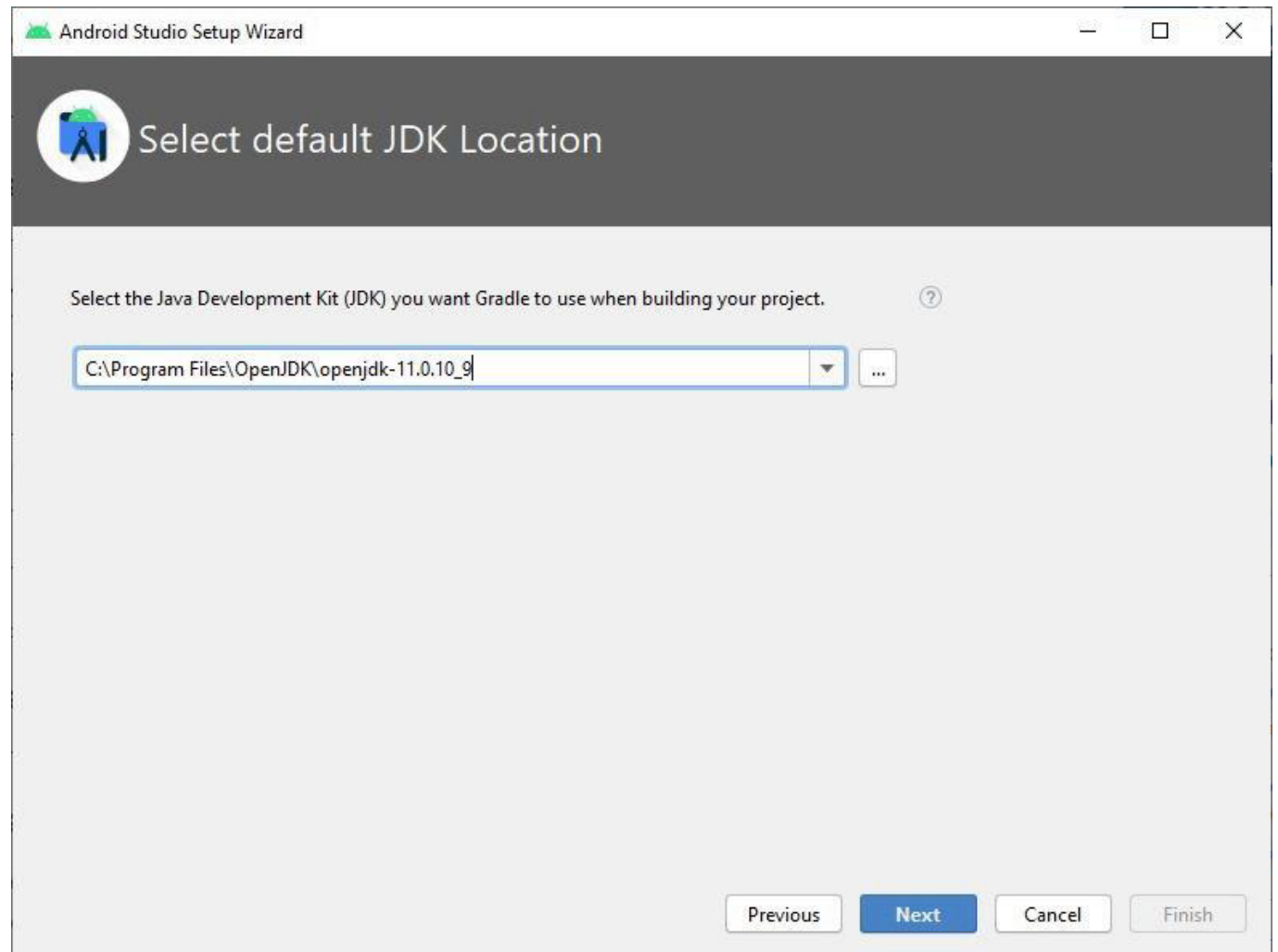
Em seguida, o Android Studio começará a carregar. Em algum ponto do carregamento, será apresentada uma janela sobre compartilhamento de dados anônimos com a Google. Essa opção é pessoal, escolha o que preferir.

Após o carregamento terminar, deve aparecer uma página de Welcome. Clique em **Next**.

Na sequência, será pedido o tipo de instalação. Escolha a opção **Custom** e clique em **Next**.

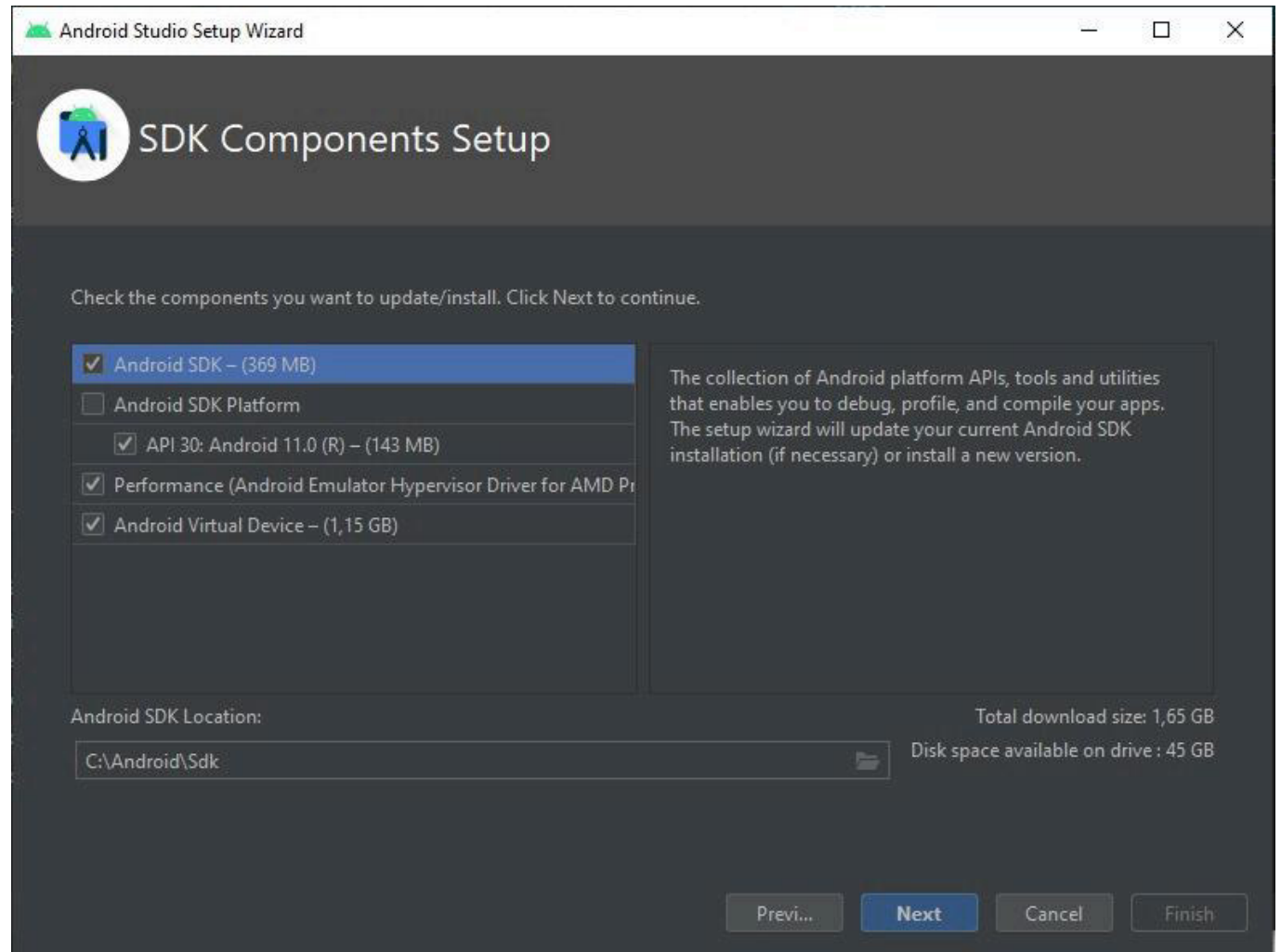
Nesse momento, será pedido para escolher a localização do pacote JDK instalado. Clique em ↓ e verifique se a opção **JAVA\_HOME** está apontando para a JDK 11. Se sim, escolha e Clique em **Next**. Caso contrário, clique no botão ... e escolha a JDK 11 (você pode inclusive utilizar o caminho anotado no **passo anterior** para te ajudar).





Em seguida, será perguntado sobre qual tema será utilizado. Escolha o que preferir e clique em **Next**

Chegamos na etapa mais importante do processo, a instalação da SDK. A janela apresentará algumas opções, marque todas.



Caso seu processador seja ADM, a opção Intel HAXM não irá aparecer. Caso seu processador seja Intel, a opção Performance (Android Emulator Hypervisor Driver for AMD) não irá aparecer.

- A **SDK** é o pacote que vai possibilitar que sua aplicação React Native faça o build. Por padrão, ele instala a última SDK estável;
- O **Intel HAXM** é para melhorar a performance da emulação de processadores Intel (Se você for utilizar o Hyper-V do Windows, **não marque essa opção**. Siga esse **guia** ao final da instalação do Android Studio);
- O **Perfomance (Android Emulator Hypervisor Driver for AMD)** é para melhorar a performance da emulação de processadores AMD;
- O **Android Virtual Device** vai criar um emulador padrão pronto para execução.

Um fator essencial nessa etapa é o caminho de instalação da SDK. Utilize a pasta que você criou na seção **Preparativos para o Android Studio** (Ex.: `C:\Android\Sdk`). Não utilize **espaços ou caracteres especiais** pois causará erros mais para frente e o próprio Android Studio alerta se o seu caminho não está bom.

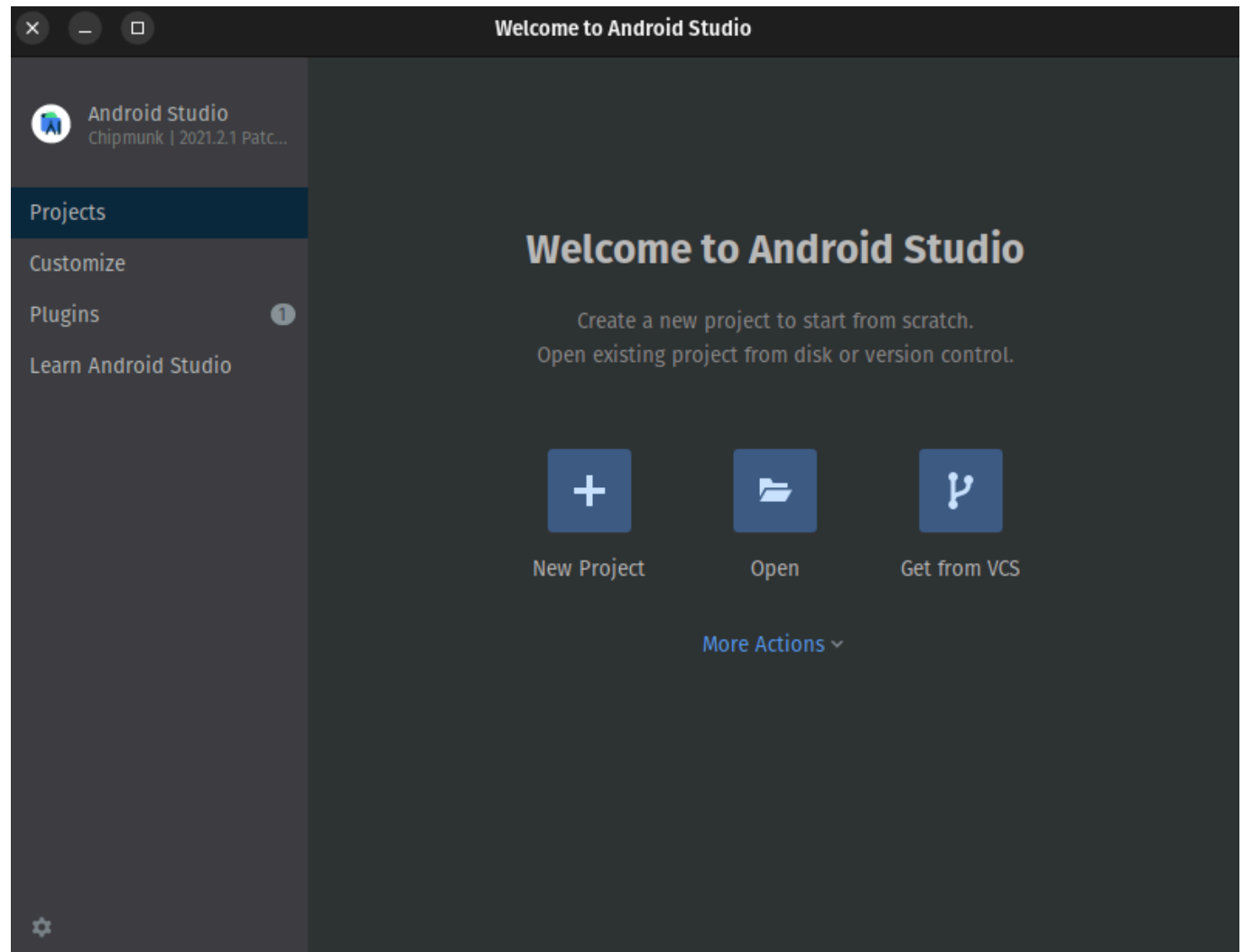
Se tudo estiver correto, clique em **Next**.

Na sequência, caso seu processador seja Intel, temos uma janela perguntando sobre a quantidade de RAM que será disponibilizada para que o HAXM utilize. Essa etapa não irá aparecer para todos pois nem todo computador é compatível com esse recurso. Deixe o recomendado pelo programa e clique em **Next**.

Em seguida, será apresentada uma janela com um resumo de todas as opções escolhidas até aqui. Verifique se está tudo certo, principalmente os caminhos da SDK e do JDK. Clique em **Finish**.

Por fim, será realizada a instalação das configurações selecionadas. Quando o programa terminar, clique em **Finish**.

Caso a instalação e configuração do Android Studio ocorra com sucesso, você deve se deparar com uma tela semelhante a essa:

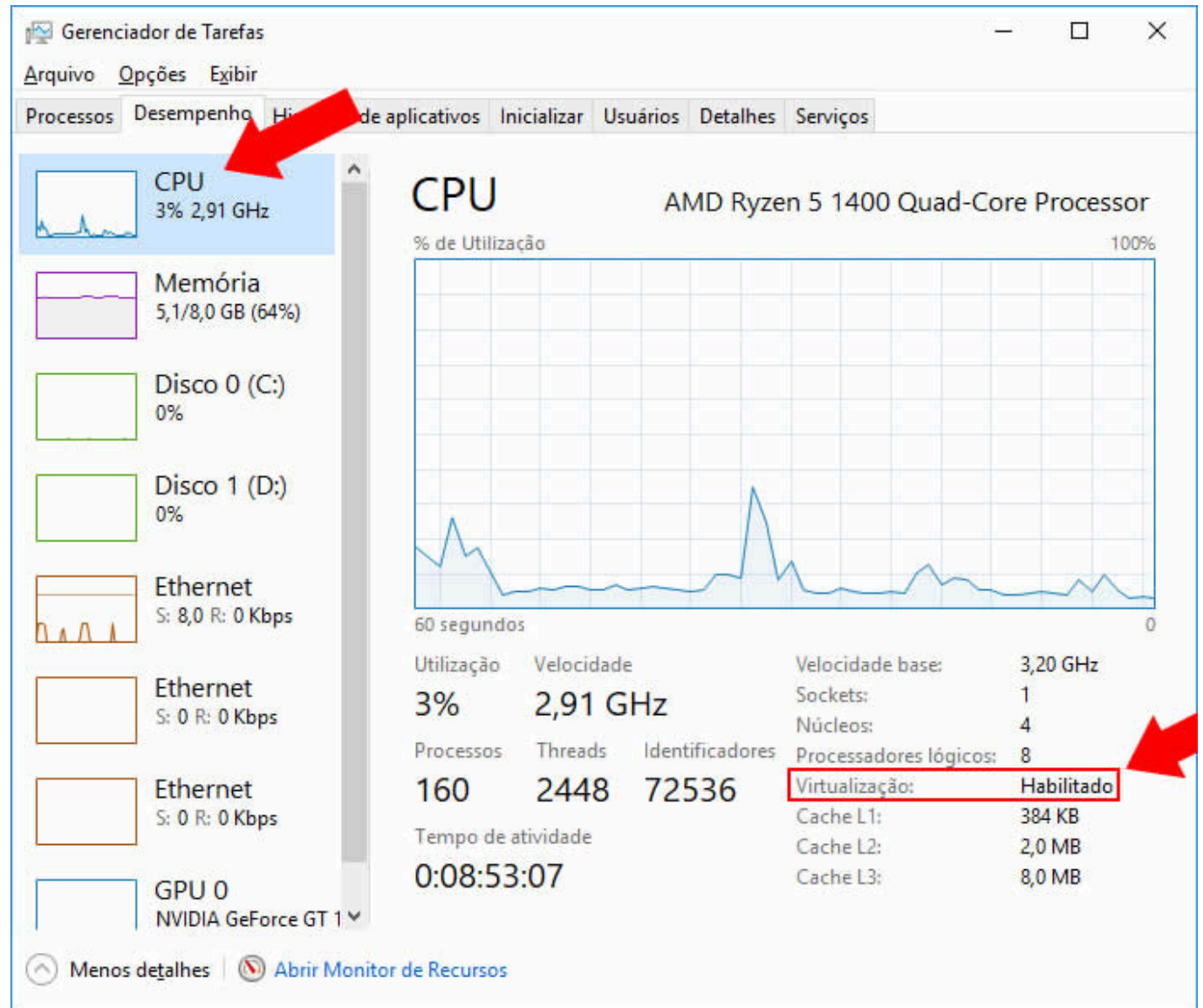


## Windows Hypervisor Platform - WHPX (Extra)

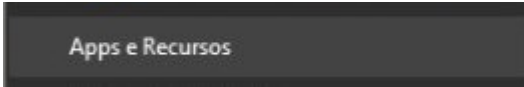
Caso você tenha um processador AMD ou queira utilizar o Hyper-V, a melhor opção é utilizar o WHPX. Para isso, é preciso verificar primeiramente se você atende aos requisitos mínimos:

- Processador Intel: compatibilidade com os recursos de Tecnologia de virtualização (VT-x), Tabelas de páginas estendidas (EPT, na sigla em inglês) e Unrestricted Guest (UG). A VT-x precisa estar ativada nas configurações do BIOS de computador.
- Processador AMD: o processador AMD Ryzen é recomendado. A virtualização ou o SVM precisa estar ativado nas configurações do BIOS do computador.
- Android Studio 3.2+.
- Android Emulator 27.3.8+.
- Windows 10 com atualização de abril de 2018 ou posterior.

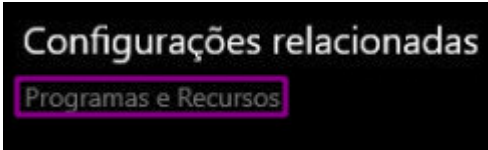
Caso não saiba se a virtualização está ativa na BIOS, basta acessar o Gerenciador de Tarefas do Windows (Ctrl + Shift + ESC) acessar a aba **Desempenho** e verificar se, a escolher a opção **CPU**, na parte inferior direita aparece a configuração **Virtualização: Habilitada**



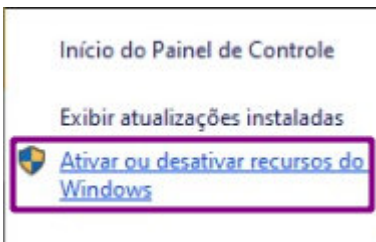
Atendidos todos os requisitos, clique com o botão direito no símbolo do Windows na barra de tarefas e escolha a opção **Apps e Recursos**.

A dark-themed sidebar from the Windows Settings application. The text "Apps e Recursos" is displayed in a light gray font.

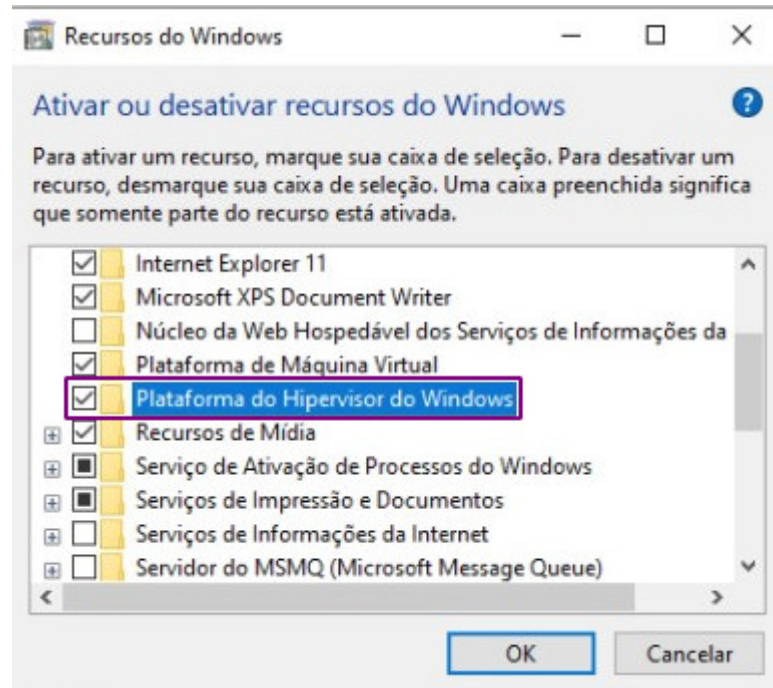
Na janela que abrir, role até o final da página e escolha a opção **Programas e Recursos** na seção Configurações Relacionadas.

A dark-themed section titled "Configurações relacionadas" (Related settings). Below the title, the text "Programas e Recursos" is highlighted with a red rectangular box.

Será apresentado o Painel de Controle. Escolha a opção **Ativar ou desativar recursos do Windows**.



Novamente será aberta uma nova janela. Nela, basta selecionar a opção **Plataforma do Hipervisor do Windows**, clicar em OK e reiniciar a máquina.



## Emulador

Com o Android Studio, é possível configurar um emulador Android e executar a sua aplicação nele.

Porém, esses emuladores consomem bastante recursos do seu computador. Por isso, se você possui um dispositivo físico Android e sua máquina tenha configurações modestas (ex.: ↓ i3, ↓ 4gb RAM), é recomendado executar a aplicação **diretamente no dispositivo físico**.

Para aprender a configurar o seu emulador no Android Studio e executar a sua aplicação React Native nele, clique **aqui**

## Dispositivo Físico



Caso você tenha um dispositivo físico Android, também é possível executar a aplicação diretamente nele. Caso tenha interesse em configurar essa etapa, clique **aqui**

 [Edit this page on GitHub](#)

 [PREV](#)  
**macOS**

[NEXT](#)   
**Linux**