

Documentação do Ambiente de React

2020

Sumário

- 1. Arquivos necessários, p3**
- 2. Instalação dos programas, p4**
 - 2.1. Node, p4**
 - 2.2. Python, p4**
 - 2.3. JDK, p5**
 - 2.4. Android Studio, p5**
 - 2.5. Genymotion, p9**
- 3. Integração do Android Studio com o Genymotion, p11**
- 4. Instalação do React Native, p16**
- 5. Configurando as variáveis de ambiente, p17**

1. Arquivos necessários

Para configurarmos o ambiente, primeiro precisaremos instalar os seguintes programas:

- **Android Studio:** (<https://developer.android.com/studio>)
- **JDK:**
(<https://www.oracle.com/br/java/technologies/javase/javase-jdk8-downloads.html>)
- **Python:** (<https://www.python.org/downloads/>)
- **Node:** (<https://nodejs.org/en/>)
- **GenyMotion:** (<https://www.genymotion.com/download/>) (cria uma conta pra uso pessoal)

Alguns dos programas podem gerar incompatibilidade com o projeto, por isso estamos utilizando as versões a seguir, procure por elas na versão do seu sistema operacional:

- **React-native-cli:** 2.0.1
- **React-native:** 0.60.5
- **Node:** v12.13.1
- **Python:** 3.8.0
- **JDK :** V:8 atualização 241 (build 1.8.0_241-b07)

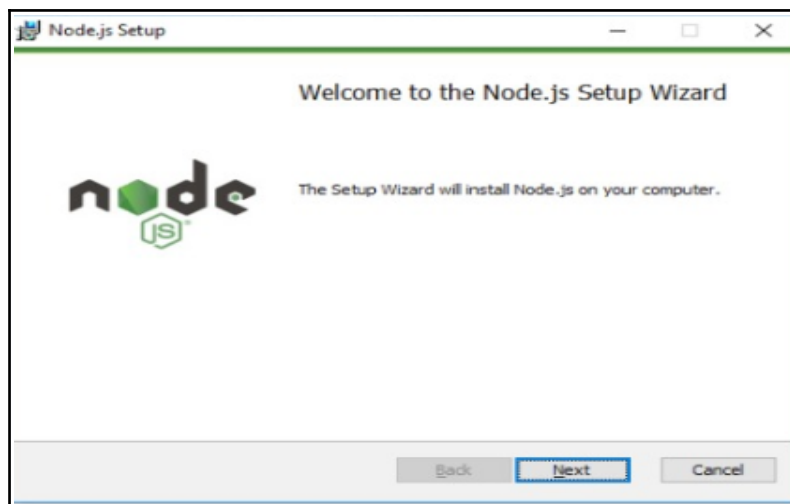
2. Instalação dos programas

Após baixar os arquivos no item anterior, realizaremos a instalação de cada um deles. Seguiremos a seguinte ordem:

1. Node
2. Python
3. JDK
4. Android Studio
5. Genymotion

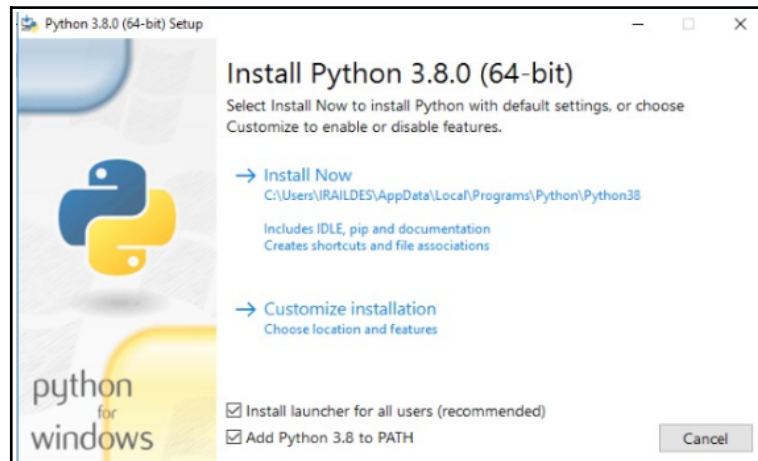
2.1. Node

A instalação do Node é um processo bem simples: abra o arquivo, aceite os termos, selecione a pasta onde deseja instalar e siga os passos.



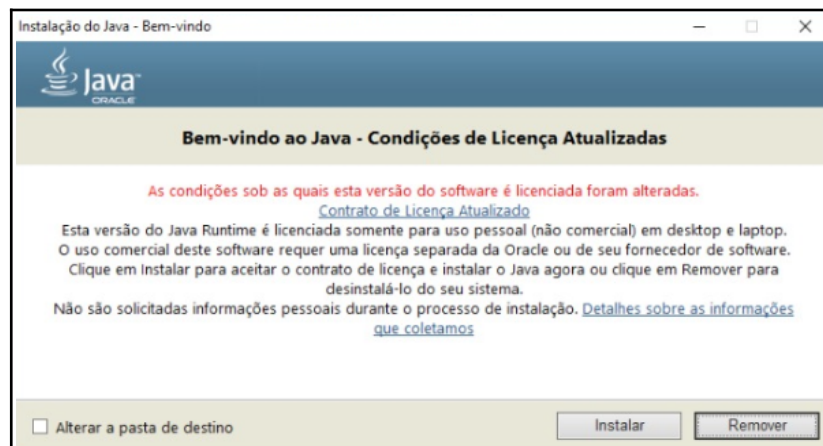
2.2. Python

O processo para o Python é bem parecido. Atente-se para a primeira aba para selecionar a caixa "Add Python 3.8 to PATH" e clique em "Install Now". Agora é só aguardar que será instalado.



2.3. JDK

Para o JDK, seguimos o mesmo padrão. Caso queira decidir um local específico para a instalação, clique na caixa “Alterar a pasta de destino” no canto inferior esquerdo. Senão, clique em instalar e aguarde.

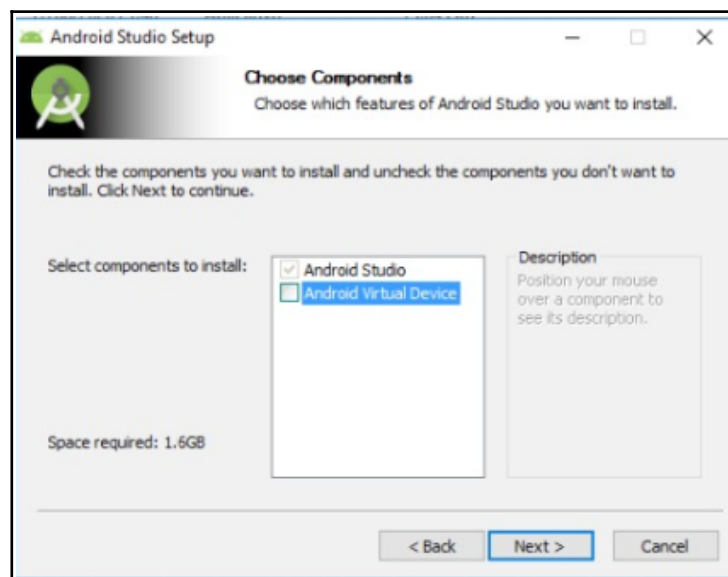


2.4. Android Studio

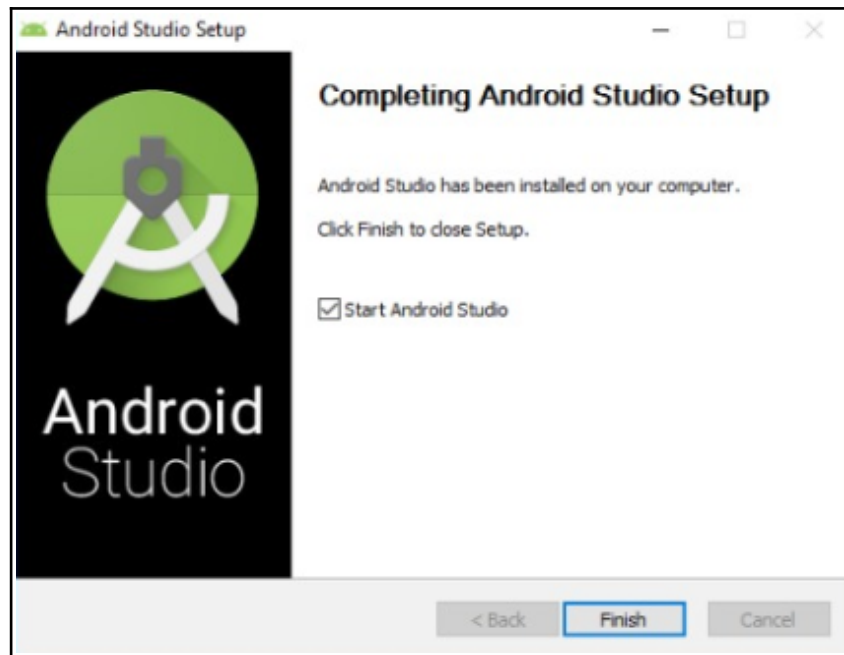
A instalação e configuração do Android Studio irá consumir mais tempo. Execute o arquivo e clique em “Next”.



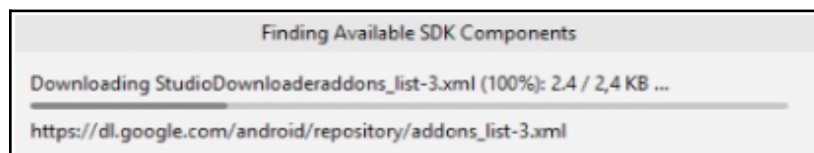
Na segunda aba, desmarque a caixa “Android Virtual Device”, estamos utilizando o Genymotion para emular, num momento posterior faremos a integração dele com o Android Studio.



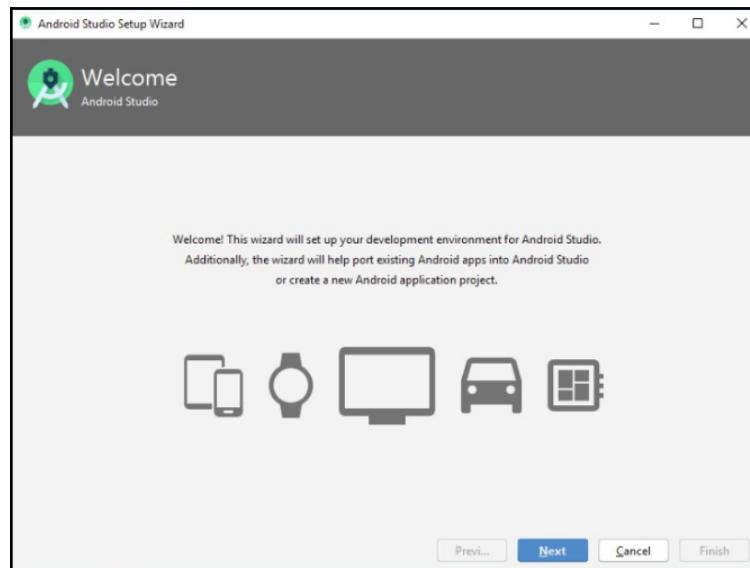
Selecione a pasta que deseja instalar os arquivos e clique em “Next”. Agora é só esperar a instalação. Quando finalizar, deixe a aba “Start Android Studio” para fazermos algumas configurações e clique em “Finish”.



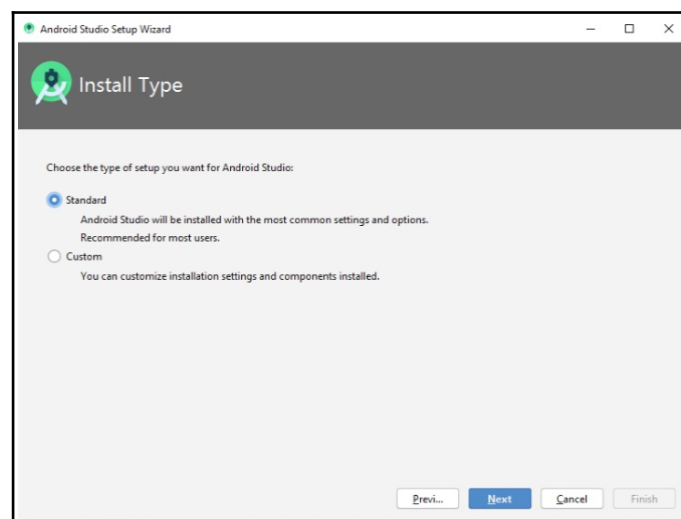
Após isso, é normal começar o download de alguns arquivos necessários, aguarde o término.



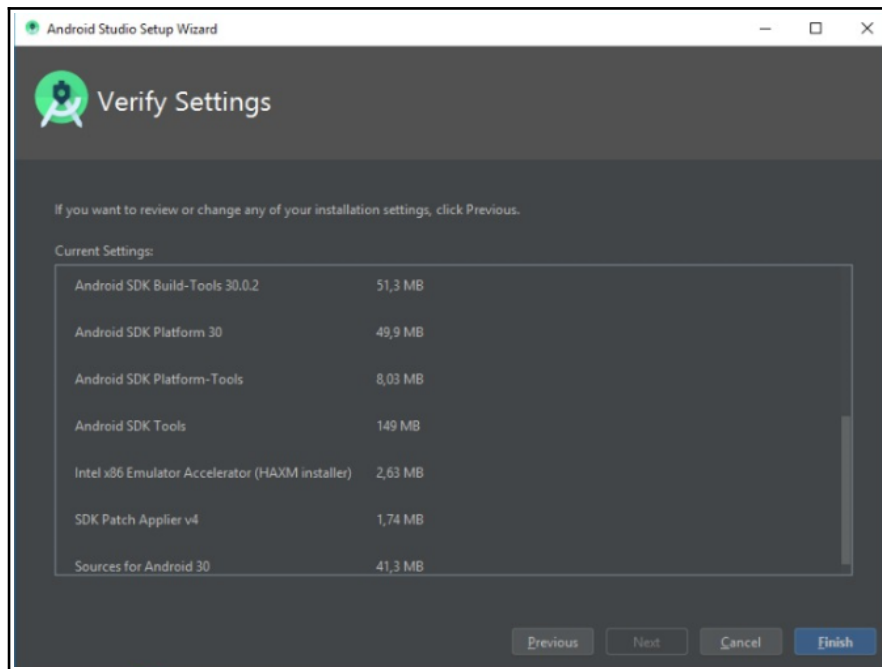
Quando acabar, o AS irá iniciar na tela a seguir, clique em “Next”.



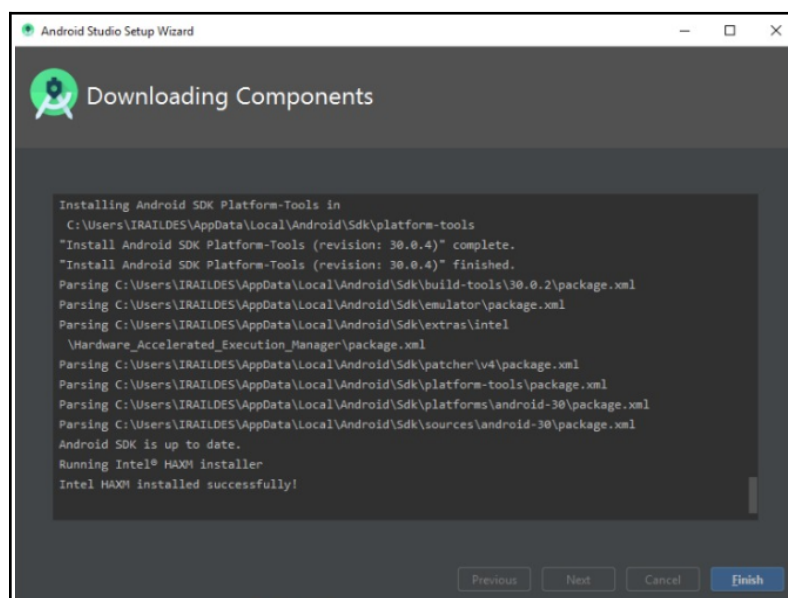
Selecione a instalação “Standard” e clique em “Next”.



A aba a seguir apenas nos mostra as configurações feitas, caso queira conferir leia e depois clique em “Finish”.

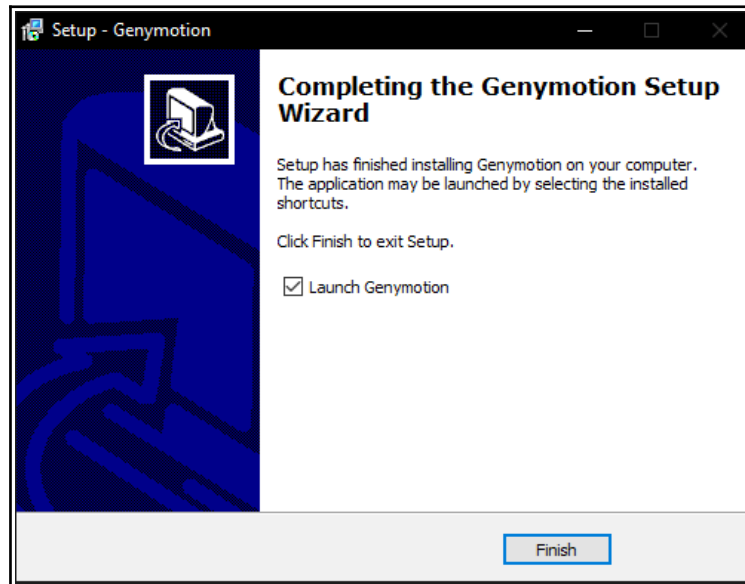


Agora o AS começará a fazer o download dos seus componentes, esse processo pode ser demorado.



2.5. Genymotion

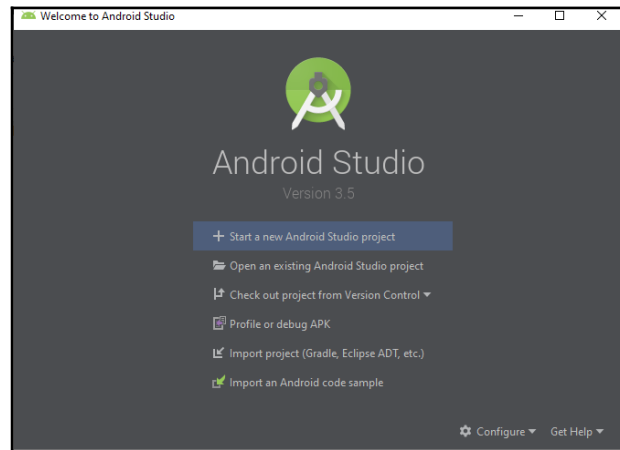
A instalação do Genymotion também segue um padrão simples. Faça as configurações básicas, peça para instalar e aguarde.



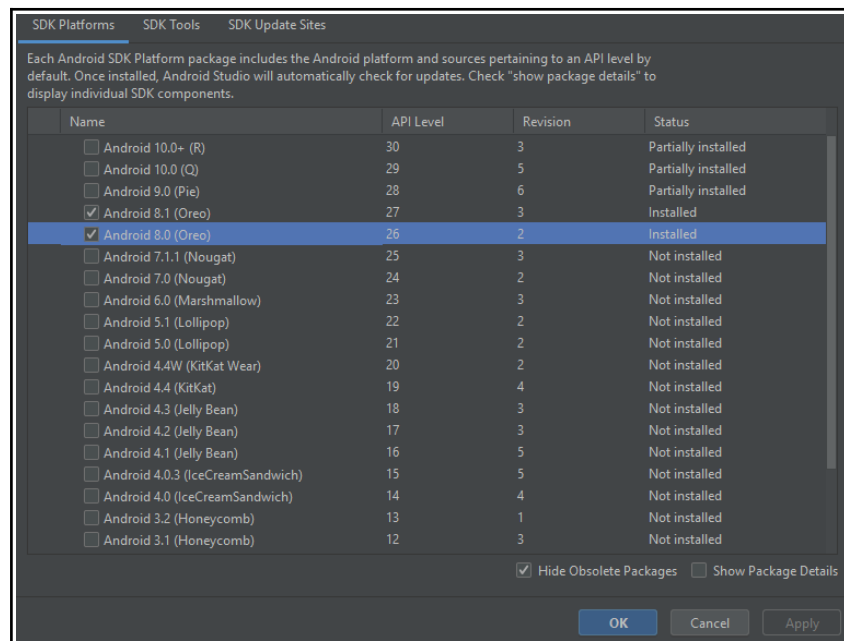
Durante a instalação, caso você não tenha em sua máquina, ele pedirá para instalar o VirtualBox, instale.

3. Integração do Android Studio com o Genymotion

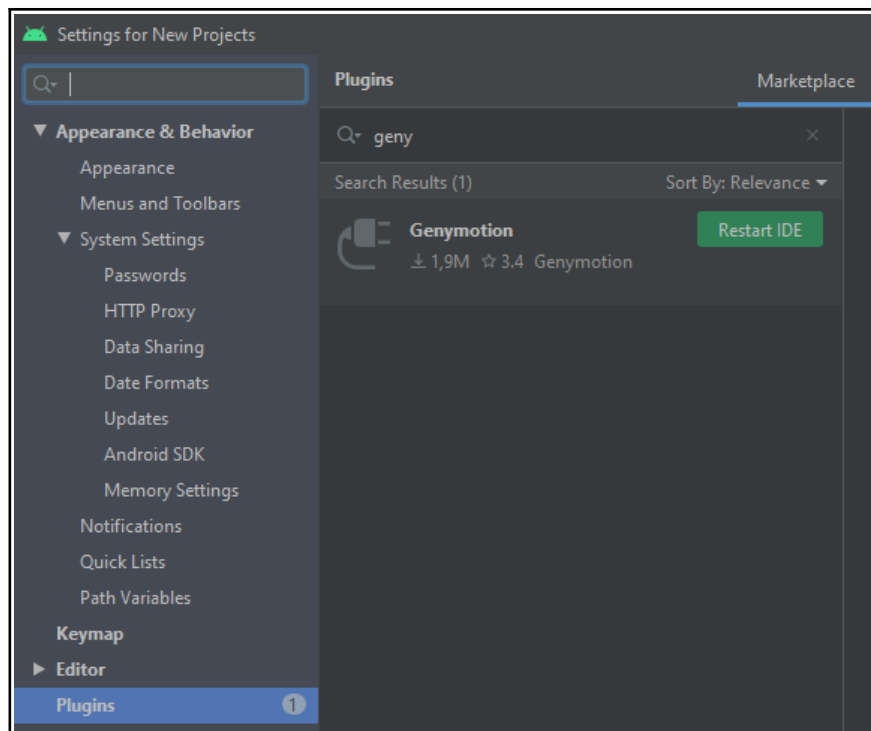
Abra o Android Studio e, na tela abaixo, clique em “Configure” e clique em “SDK Manager”.



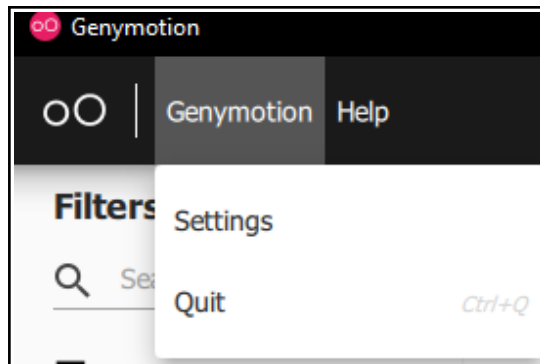
Na aba SDK Platform, selecione Android 8.0 e 8.1 e confirme.



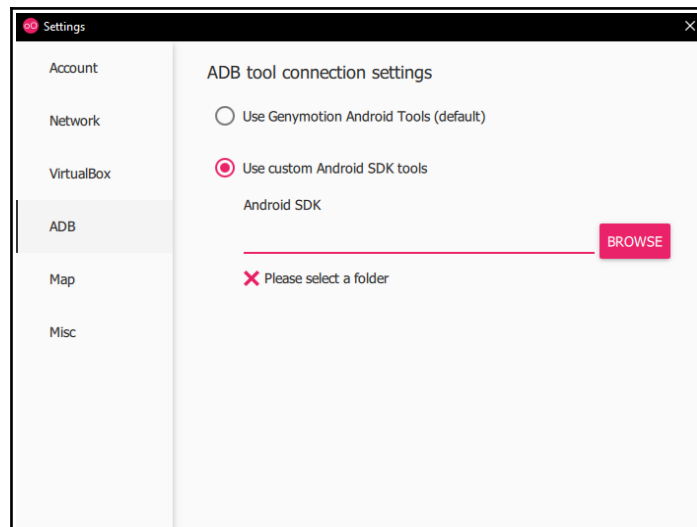
Nessa mesma aba, no menu lateral a esquerda, clique na opção “Plugins”, pesquise por “Genymotion” na lupa e instale a extensão. Após instalar, será pedido para reiniciar a IDE, clique e aguarde.



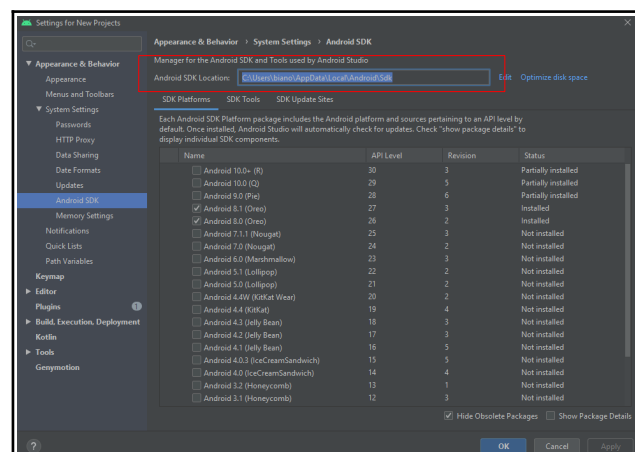
Após o AS reiniciar, minimize-o e abra o Genymotion. Com ele aberto, vá na aba superior e clique em Genymotion>Settings.



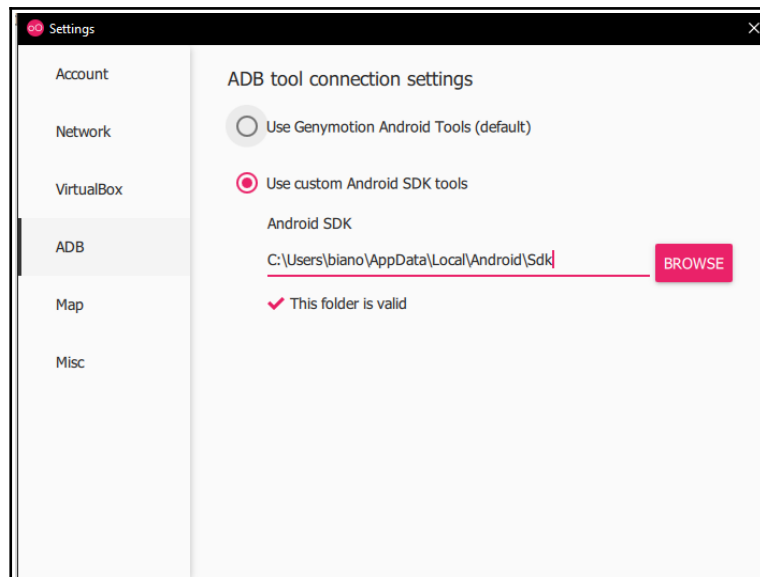
Na aba que irá abrir, clique em ADB, selecione a opção “Use custom Android SDK Tools”



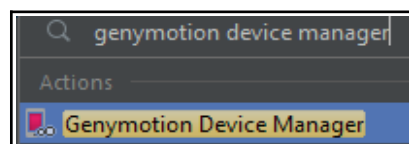
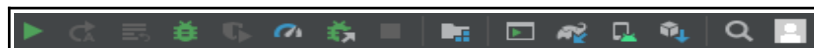
Agora volte no AS, abra o SDK Manager e copie o caminho que está escrito em “Android SDK Location”.



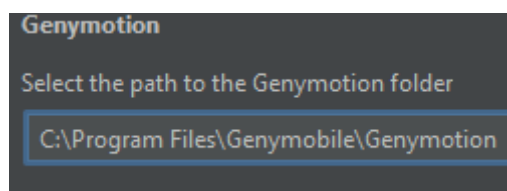
Escreva esse caminho no campo do Genymotion e feche essa aba.



Volte ao AS e, no menu superior do lado direito, você encontrará uma lupa, clique nela e pesquise por “Genymotion Device Manager”.

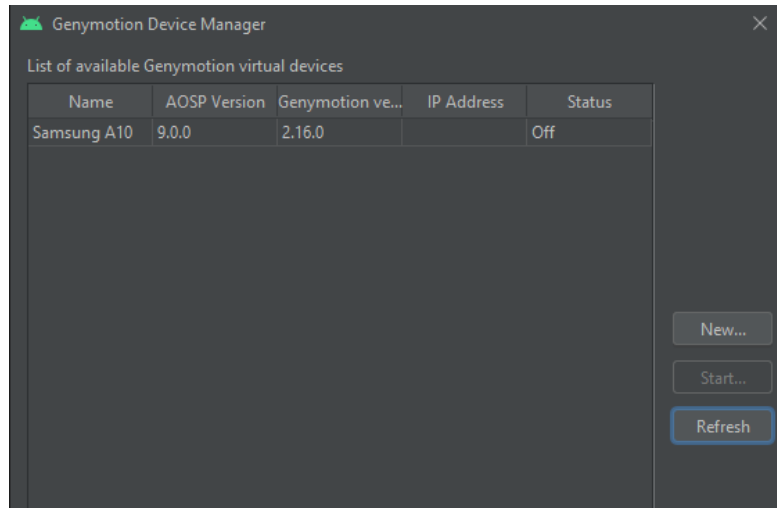


Abra, insira o caminho em que foi instalado o Genymotion, clique em Aplly e Ok.






Agora repita o mesmo processo para pesquisar o Genymotion Device

Manager e ele já abrirá com as opções de simuladores que você tem instalado no Genymotion como mostra a imagem a seguir.



Caso ainda não tenha configurado nenhum simulador, vá no Genymotion, clique no sinal de “mais”, escolha um celular e clique em instalar. Após a instalação ele irá aparecer na lista de dispositivos.

Type	Device ↕	Android API	Size	Density	Source	Status	
	Google Nexus 9	7.1 - API 25	1536 x 2048	320 - XHDPI	Genymotion	Off	⋮
	Samsung A10	9.0 - API 28	720 x 1520	260	Genymotion	Off	⋮
	Samsung Galaxy S9	8.0 - API 26	1440 x 2960	560	Genymotion	Off	⋮

4. Instalação do React Native

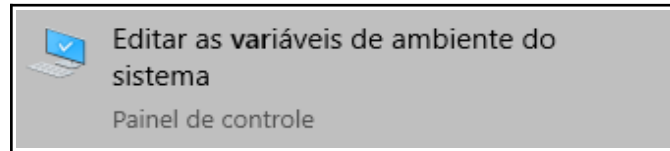
Após a instalação de todos os itens anteriores, abra o Prompt de Comando e execute os seguintes comandos um de cada vez:

- `npm install`
- `npm install -gl react-native-cli`

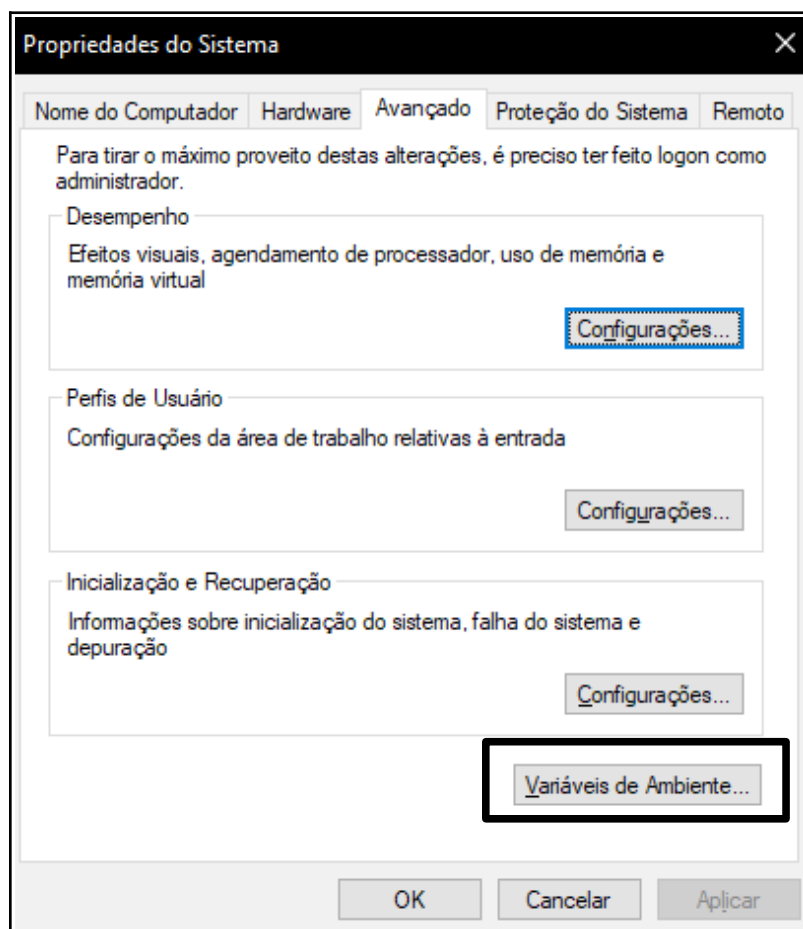
```
C:\Users\biano>npm install -gl react-native-cli
C:\Users\biano\AppData\Roaming\npm\react-native -> C:\Users\biano\AppData\Roaming\npm\node_modules\react-native-cli\index.js
+ react-native-cli@2.0.1
added 81 packages from 28 contributors and updated 1 package in 8.982s
```


5. Configurando as variáveis de ambiente

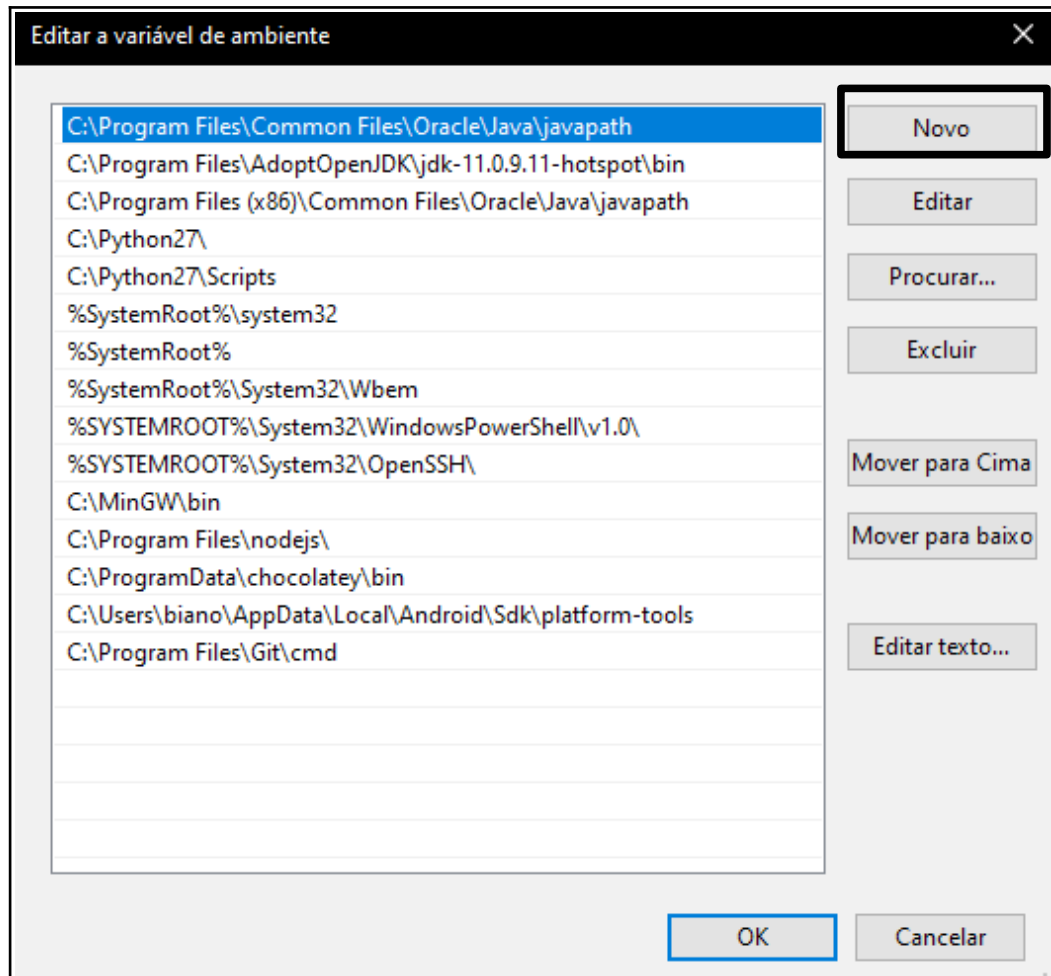
Abra o menu iniciar e pesquise por “variável de ambiente”.



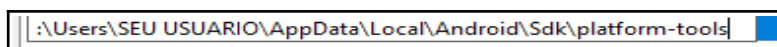
Clique em Variáveis de Ambiente> Variável de Sistema>Path>Novo



Variáveis do sistema	
Variável	Valor
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;C:\Program ...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 142 Stepping 9, GenuineIntel
PROCESSOR_LEVEL	6



Agora procure no seu computador o seguinte caminho: **C:\Users\SEU USUARIO\AppData\Local\Android\Sdk\platform-tools**, cole esse caminho no path da variável e clique em “Ok”.



Para testar, abra o Prompt de Comando novamente e digite “ABD”, se aparecer algo como na imagem abaixo está tudo certo.

```

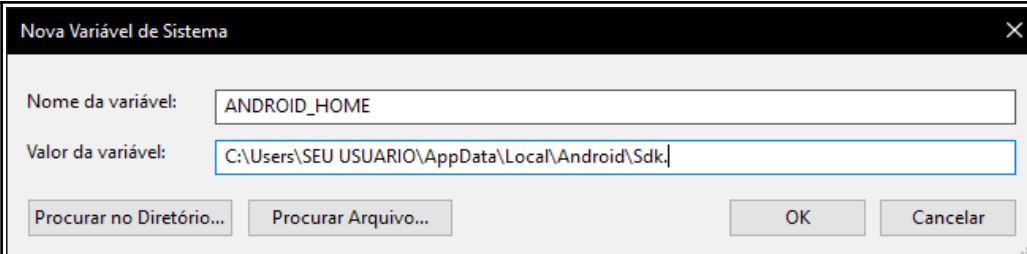
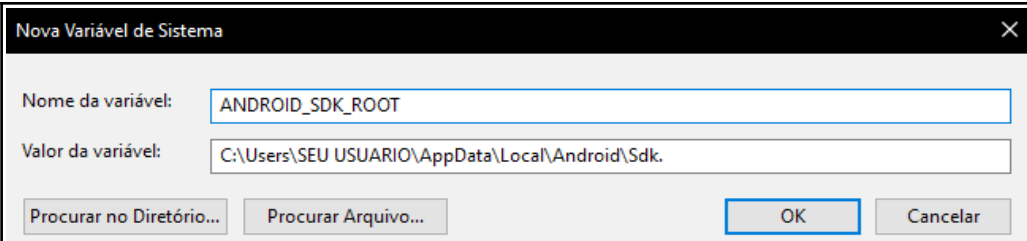
will automatically reboot the device.
reboot [bootloader|recovery|sideload|sideload-auto-reboot]
  reboot the device; defaults to booting system image but
  supports bootloader and recovery too. sideload reboots
  into recovery and automatically starts sideload mode,
  sideload-auto-reboot is the same but reboots after sideloading.
sideload OTAPACKAGE      sideload the given full OTA package
root                    restart adbd with root permissions
unroot                  restart adbd without root permissions
usb                     restart adbd listening on USB
tcpip PORT              restart adbd listening on TCP on PORT

internal debugging:
start-server            ensure that there is a server running
kill-server             kill the server if it is running
reconnect               kick connection from host side to force reconnect
reconnect device        kick connection from device side to force reconnect
reconnect offline       reset offline/unauthorized devices to force reconnect

environment variables:
$ADB_TRACE              comma-separated list of debug info to log:
                        all,adb,sockets,packets,rwx,usb,sync,sysdeps,transport,jdwp
$ADB_VENDOR_KEYS        colon-separated list of keys (files or directories)
$ANDROID_SERIAL         serial number to connect to (see -s)
$ANDROID_LOG_TAGS        tags to be used by logcat (see logcat --help)
$ADB_LOCAL_TRANSPORT_MAX_PORT max emulator scan port (default 5585, 16 emus)
$ADB_MDNS_AUTO_CONNECT  comma-separated list of mdns services to allow auto-connect (default adb-tls-connect)

```

Também precisaremos criar as variáveis **ANDROID_HOME** e **ANDROID_SDK_ROOT**. Para isso, vá novamente em Variável de Ambiente>Novo, informe o nome da variável e, no valor das duas, informe esse caminho na sua máquina **C:\Users\SEU USUARIO\AppData\Local\Android\Sdk**.

Agora já estamos prontos para utilizar o ambiente!