

Tiago Porto

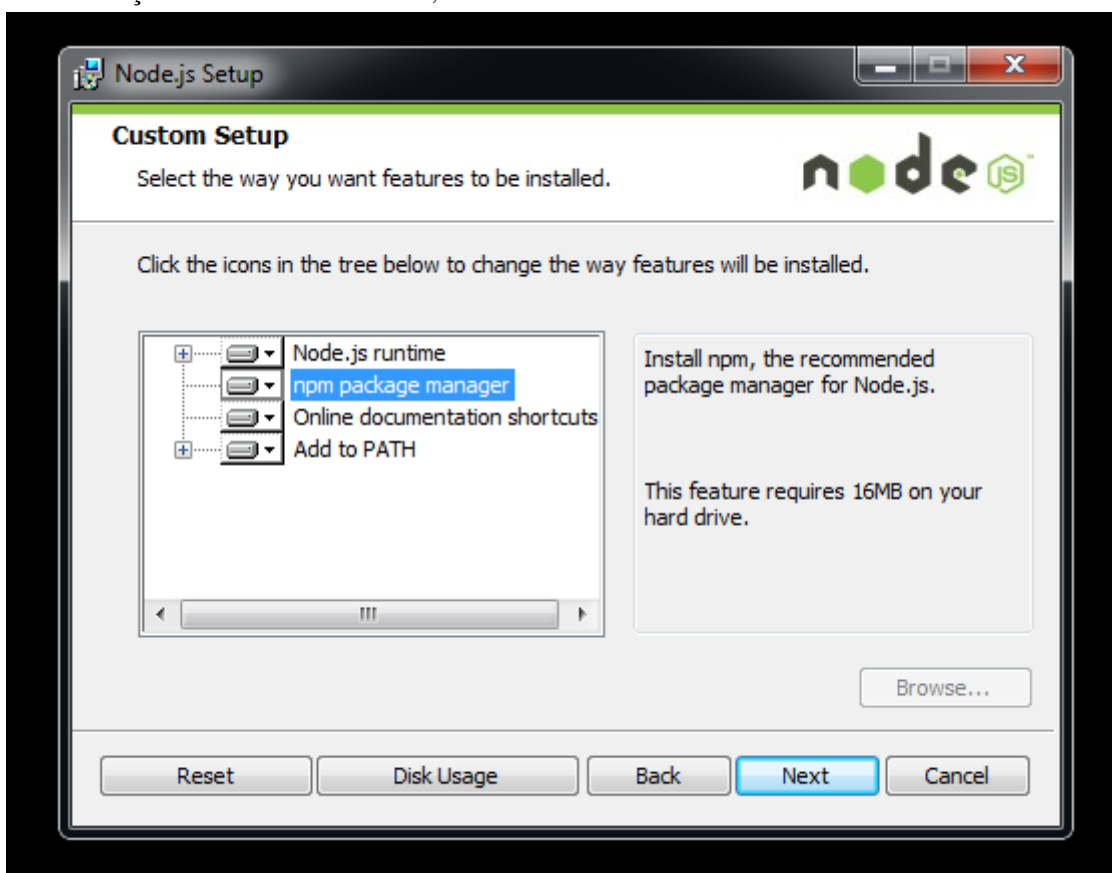
Designer de Interface & Desenvolvedor Front-End

Tutorial de configuração do ambiente Ionic + Android no Windows

Esse post vai em resposta à um pedido feito por um dev que acessou minha palestra Aplicativos híbridos com Ionic. Você também pode começar a desenvolver agora!. Ele ficou em dúvida de como montar o ambiente para desenvolver em Android.

Então, vamos direto ao que interessa.

1. Logicamente precisamos do Ionic e o Apache Cordova, se ainda não os instalou, vamos lá.
 - Ambos são pacotes do Node e para isso precisamos do Node instalado. Faça o download em <https://nodejs.org/en/download/>.
 - Na instalação é necessário o NPM, não o remova.

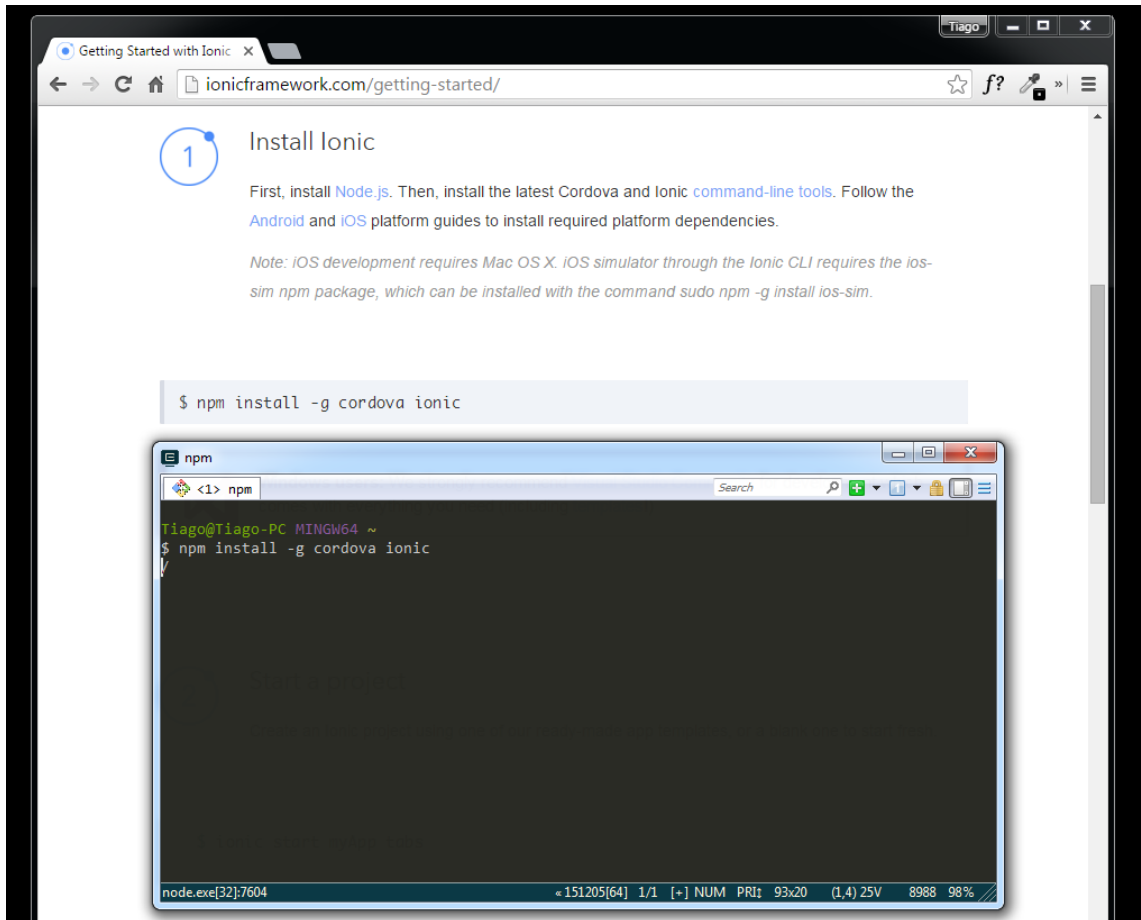


**Não sabe o que é Node e NPM, acesse <http://nodebr.com/o-que-e-a-npm-do-nodejs/>*

2. Com o Node instalado, agora vamos abrir o terminal e instalar o Ionic e o Cordova.

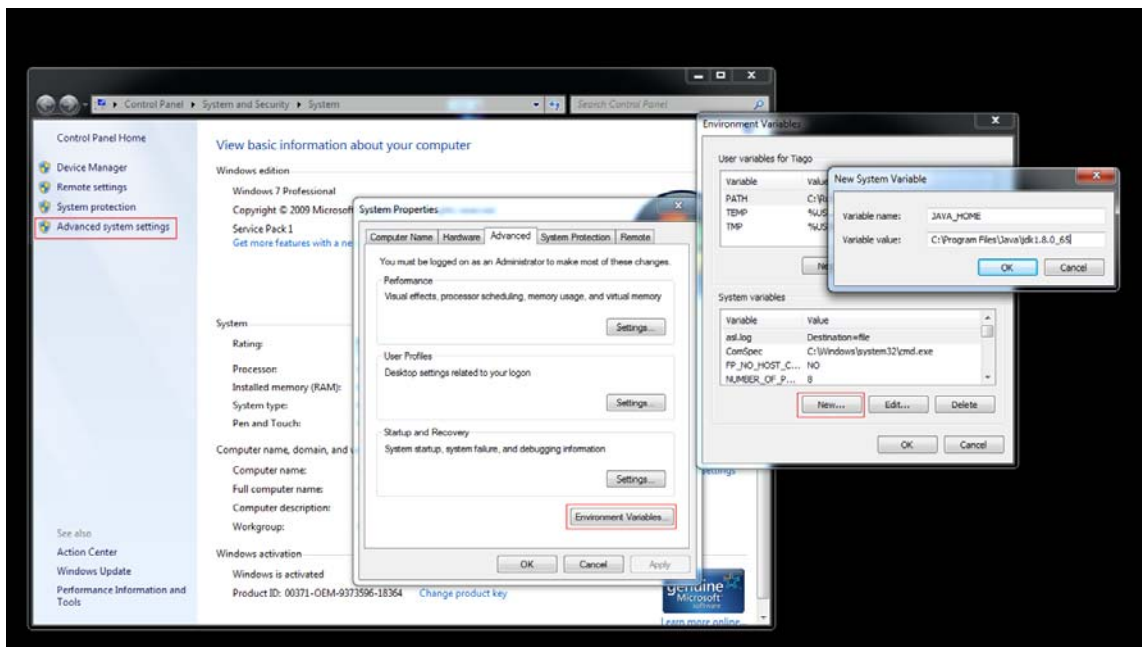
- Execute o comando

```
npm install -g cordova ionic.
```

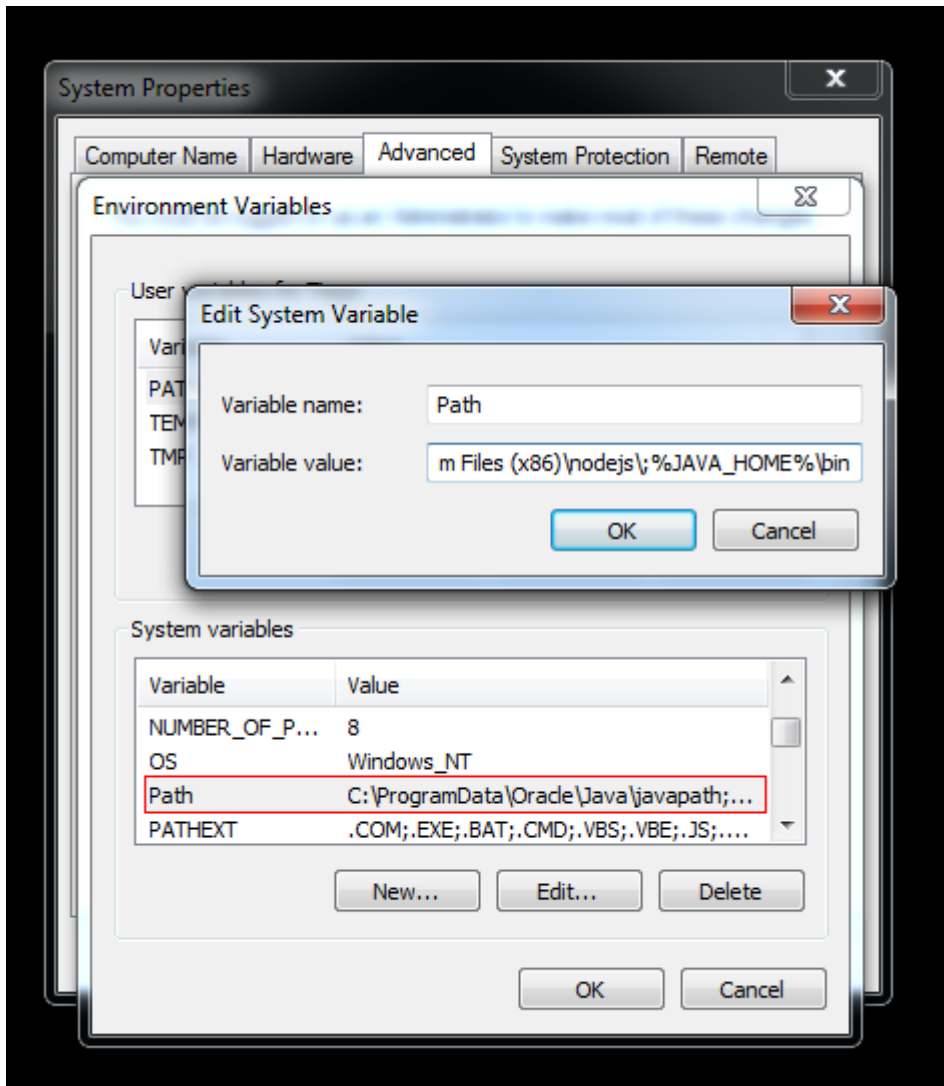


3. Iniciaremos a configuração do ambiente para Android. Vamos instalar o Java JDK.

- Acesse o link <http://www.oracle.com/technetwork/java/javase/downloads/index.html>, selecione JDK download, faça o download da versão compatível com seu SO (32 ou 64 bits). *Veja o path de onde você instalou, será necessário no próximo passo.*
- Agora vamos criar a variável **JAVA_HOME** no ambiente do windows, abra a tela de sistema com as teclas “windows + pause/break” ou vá em “painel de Controle / Sistema e Segurança / Sistema”.
- Clique em “Configurações avançadas do sistema”.
- Selecione Variáveis de ambiente.
- Selecione Nova variável de ambiente.
- Na nova janela, em nome da variável coloque **JAVA_HOME**, e no valor coloque o caminho onde você instalou o JDK. Por exemplo, **C:\Program Files\Java\jdk1.8.0_65**. Dê OK.

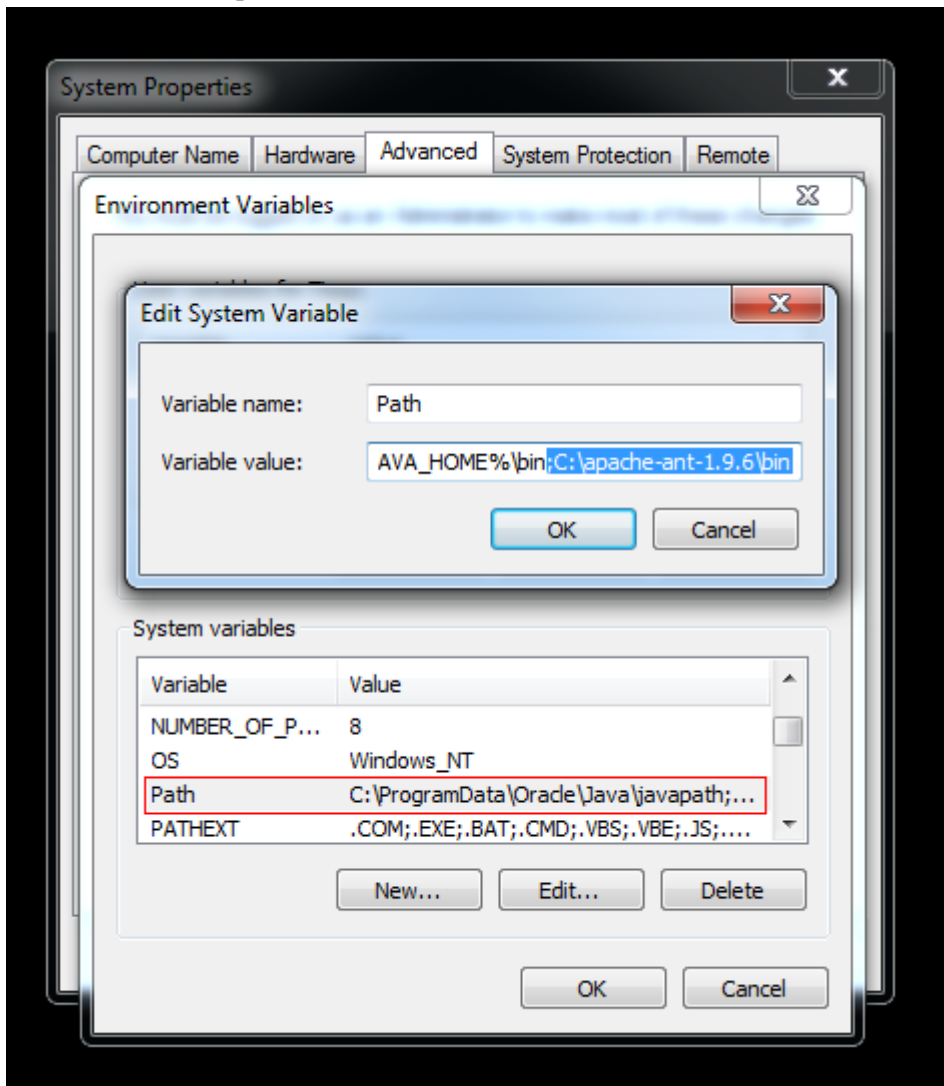


- Selecione a variável **path**, e clique em editar. Novamente irá abrir uma janela com os dados da variável, no campo valores da variável, vá até o final da *string* e adicione **;%JAVA_HOME%\bin**. Não se esqueça do “;”.

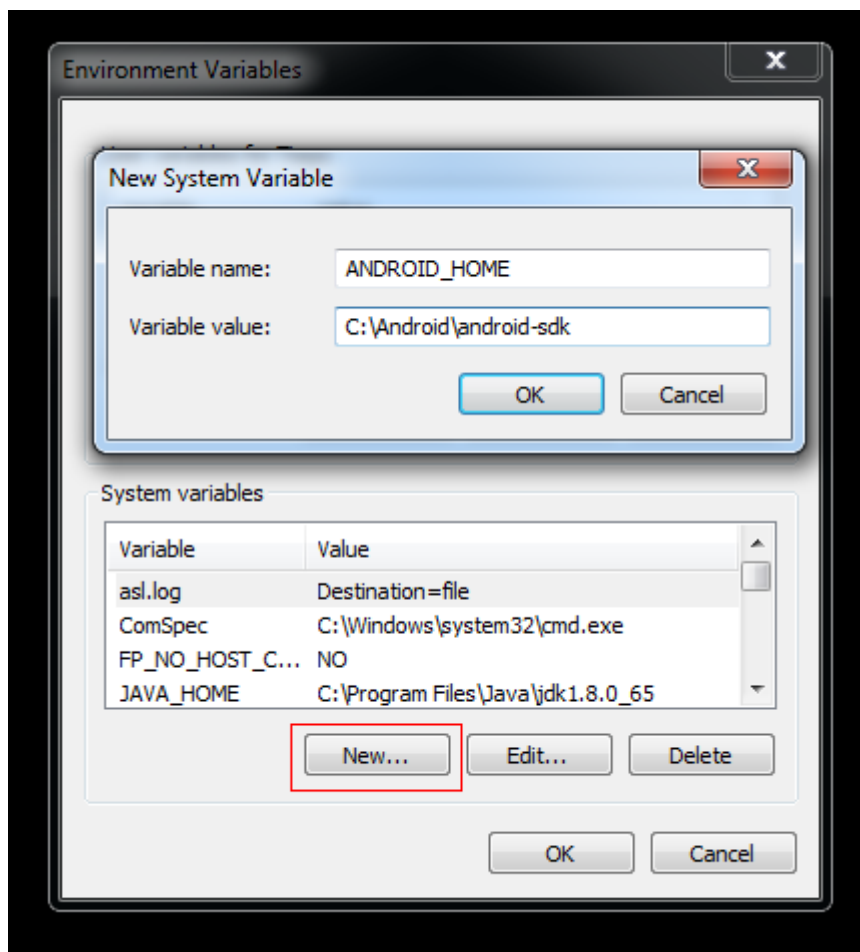


4. Com o Java configurado, agora precisamos do Apache Ant, o responsável por fazer o *build* de nosso aplicativo. Faça o download no link <http://ant.apache.org/bindownload.cgi>.
 - Após concluído o download, extraia o diretório para a raiz de C:.

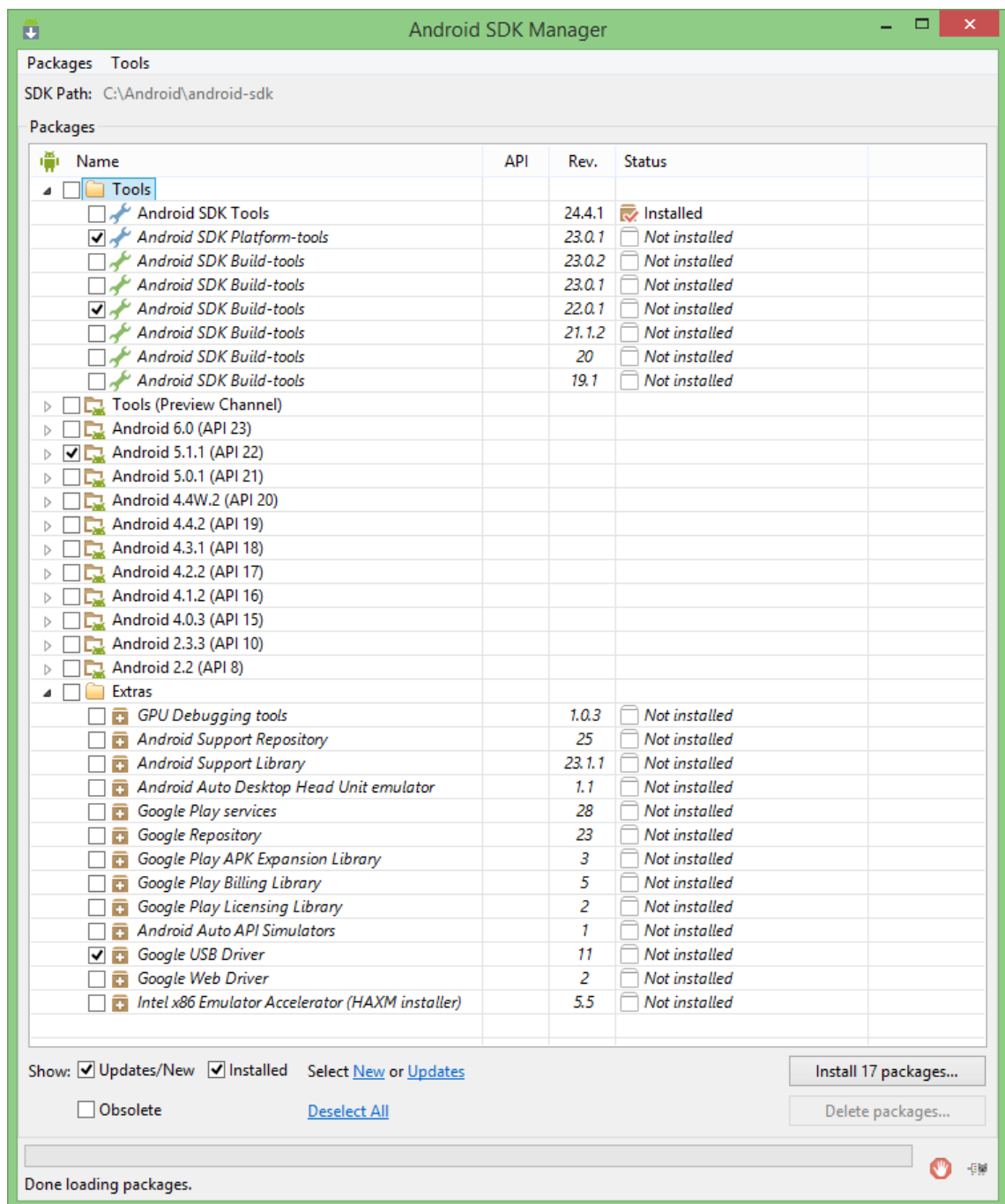
- Mais uma vez precisamos atualizar a variável **path** para acrescentar o Ant. Abra as variáveis de ambiente do windows novamente.
- Selecione Variáveis de ambiente.
- Selecione a variável **path** e clique em editar.
- Acrescente o caminho até a pasta bin do Ant, por exemplo **;C:\apache-ant-1.9.6\bin**. *Mais uma vez não se esqueça do “;”*.



5. O próximo passo é a instalação do Android SDK, a API que fornece as bibliotecas e ferramentas necessárias para *build*, teste e *debug* para Android.
 - Faça o download em <http://developer.android.com/sdk/index.html#Other>, selecione o **.exe** (Recommended).
 - Após instalado, precisamos criar a variável **ANDROID_HOME**, necessária para o Cordova. Mais uma vez abra as variáveis de ambiente do windows.
 - Selecione Variáveis de ambiente.
 - Clique em Nova.
 - Na nova janela em nome da variável coloque **ANDROID_HOME**, e no valor da variável coloque o caminho até a pasta **android-sdk** que você acabou de instalar, por exemplo: **C:\Android\android-sdk**. Dê OK.



- Voltando às variáveis de ambiente, selecione a variável **path**, e clique em editar. Na nova janela no campo valor da variável, vá até o final da *string* e acrescente
;%ANDROID_HOME%\tools;%ANDROID_HOME%\platform-tools.
- Estamos quase lá, agora precisamos instalar os pacotes do android. Abra o arquivo **SDK Manager.exe**, localizado dentro da pasta **android-sdk**.
- Por padrão o **Tools/Android SDK Tools** já está instalado. Se por algum motivo ele não estiver marcado como instalado, selecione-o. Selecione também **Tools/Android SDK Platform-tool**, **Tools/SDK build-tools** (uma versão acima da 22), e na **API** selecione a mesma versão que você selecionou no **Build Tools**. Marque também **Extras/Google USB Driver** (necessário para *debug* direto no dispositivo), e clique em instalar.

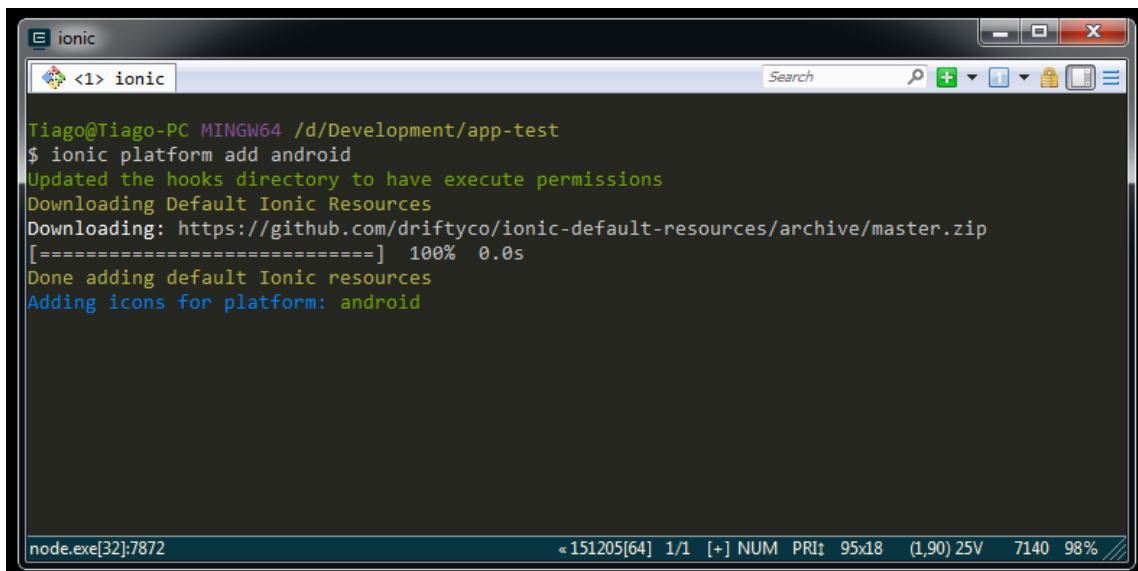


- Ao final das instalações nosso ambiente está configurado.

6. Vamos fazer o teste. Voltando ao nosso aplicativo, abra o diretório do aplicativo pelo terminal e adicione a plataforma android.

- Execute

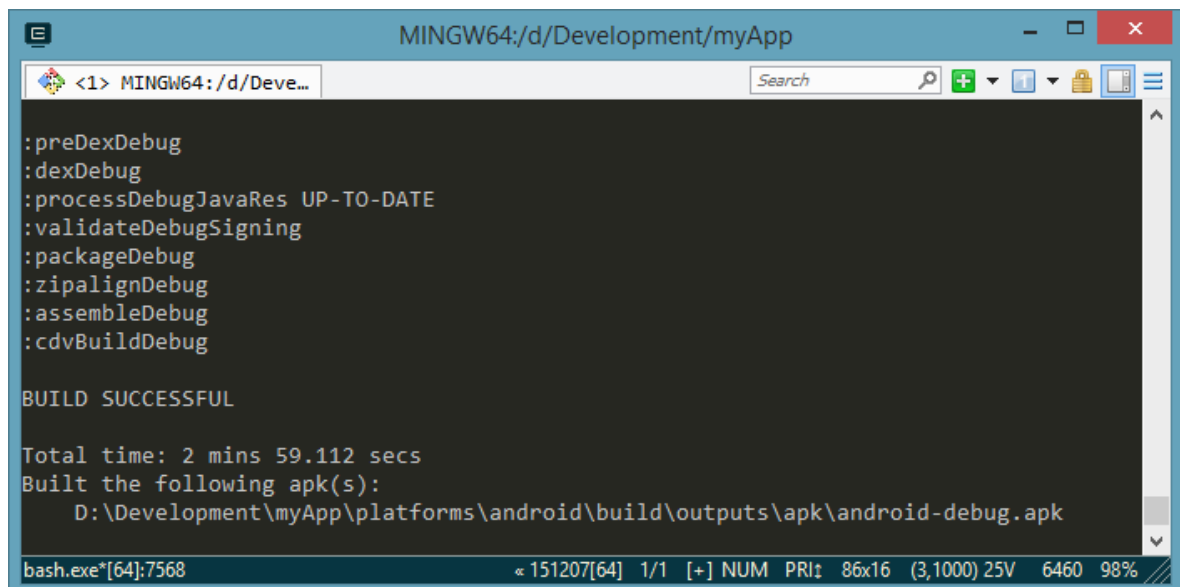
```
ionic platform add android
```



```
ionic
<1> ionic
Tiago@Tiago-PC MINGW64 /d/Development/app-test
$ ionic platform add android
Updated the hooks directory to have execute permissions
Downloading Default Ionic Resources
Downloading: https://github.com/driftyco/ionic-default-resources/archive/master.zip
[=====] 100% 0.0s
Done adding default Ionic resources
Adding icons for platform: android
node.exe[32]:7872
```

7. Vamos dar o *build* no nosso .apk. Execute

```
ionic build android
```

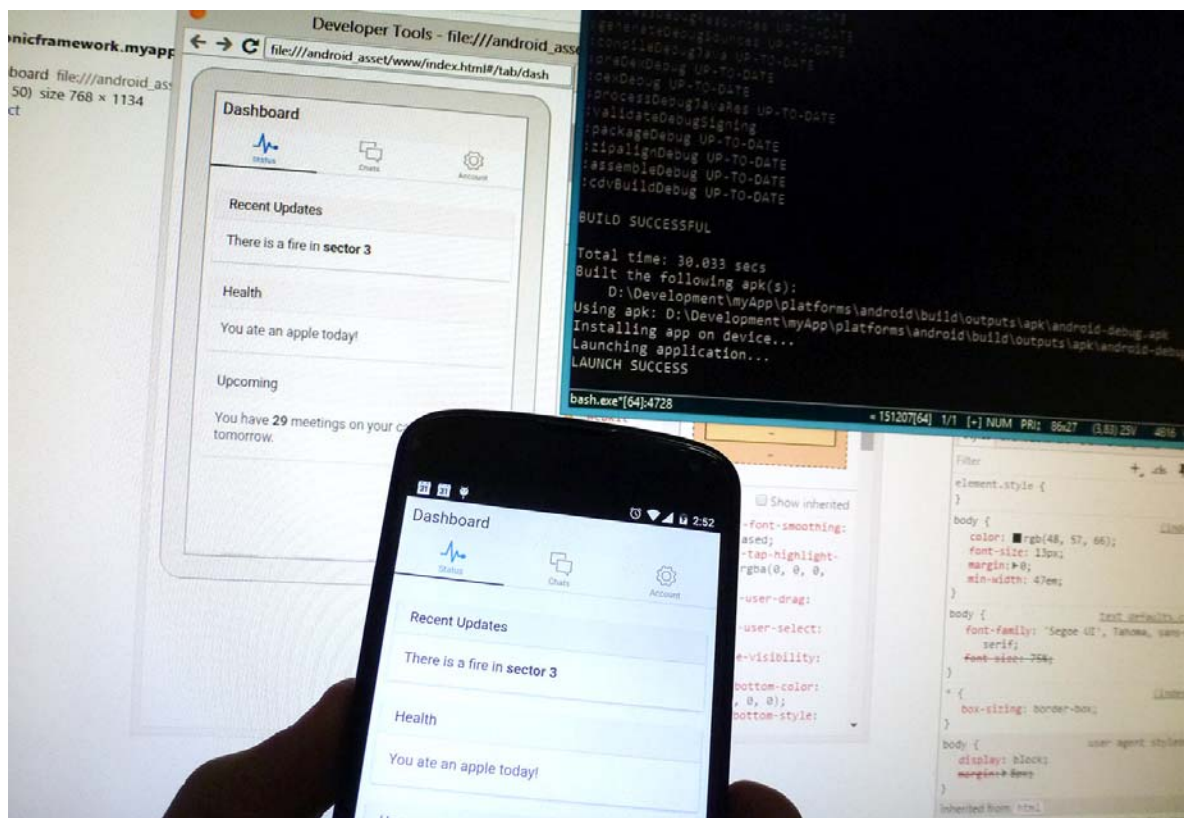


```
MINGW64:/d/Development/myApp
<1> MINGW64:/d/Deve...
:preDexDebug
:dexDebug
:processDebugJavaRes UP-TO-DATE
:validateDebugSigning
:packageDebug
:zipalignDebug
:assembleDebug
:cdvBuildDebug
BUILD SUCCESSFUL
Total time: 2 mins 59.112 secs
Built the following apk(s):
  D:\Development\myApp\platforms\android\build\outputs\apk\android-debug.apk
bash.exe*[64]:7568
```

- Após finalizado o processo, o caminho do .apk gerado é exibido no terminal.

8. Podemos também *debugar* o aplicativo direto no dispositivo. Plugue o dispositivo via cabo USB, certifique de ter habilitado o **USB debugging**, e execute

```
ionic run android
```



É isso aí, boa sorte e bom desenvolvimento.

📅 27 dezembro 2015 👤 Tiago Porto 📁 Tutorial 🔑 android, ionic, montando ambiente, tutorial, windows

Um comentário sobre “Tutorial de configuração do ambiente Ionic + Android no Windows”



Matheus

21 setembro 2016 às 1:00 PM

Muito obrigado, vai me ajudar a começar.

Orgulhosamente mantido com WordPress