

# API Sem Complicação

Aula 06





# Adam Vinicius Cezar Luchini

## ■ Carreira:

- Engenheiro de Testes de Software
- Instrutor de automação teste e desenvolvimento
- QA desde 2013





# Newman

- Executor de collections para o Postman
- Possível executar os testes do Postman via linha de comando





# Instalando o Newman

Com o Nodejs já instalado na máquina é possível realizar a instalação do Newman

Via linha de comando

Para verificar se possui o Nodejs basta abrir qualquer terminal(Git bash, cmd, Powershell)

E executar o comando `npm -version`

```
MINGW64:/c/Users/AdamVinicius  
AdamVinicius@DESKTOP-BNN0ILG MINGW64 ~  
$ npm -version  
6.14.15
```





Caso seja exibido o erro: npm não é um comando reconhecível (ou algo parecido rs)

Significa que não possui ele instalado na máquina

Na aula 01, foi disponibilizado um tutorial de instalação

Basta segui-lo que será sucesso hehe

Lembrando que sempre que realizar qualquer instalação é necessário fechar e abrir o terminal novamente.





Com o Nodejs configurado podemos instalar o Newman  
Basta utilizarmos o comando  
`npm install -g newman`

```
MINGW64:/c/Users/AdamVinicius  
AdamVinicius@DESKTOP-BNN0ILG MINGW64 ~  
$ npm install -g newman|
```





Significado do comando:

npm: É o comando do nodejs para qualquer execução

install: Para instalar qualquer biblioteca node

-g: será configurado para a máquina inteira e não somente para o projeto

newman: nome da biblioteca/aplicativo que será instalado





Após digitar o comando e pressionar enter  
A instalação será iniciada

```
MINGW64/c/Users/AdamVinicius
AdamVinicius@DESKTOP-BNN0ILG MINGW64 ~
$ npm install -g newman
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher.
Older versions may use Math.random() in certain circumstances, which is known to be problematic.
See https://v8.dev/blog/math-random for details.
C:\Users\AdamVinicius\AppData\Roaming\npm\newman -> C:\Users\AdamVinicius\AppData\Roaming\npm\node_modules\newman\bin\newman.js
+ newman@5.3.0
added 4 packages from 6 contributors, removed 31 packages, updated 25 packages and moved 2 packages in 16.981s
```



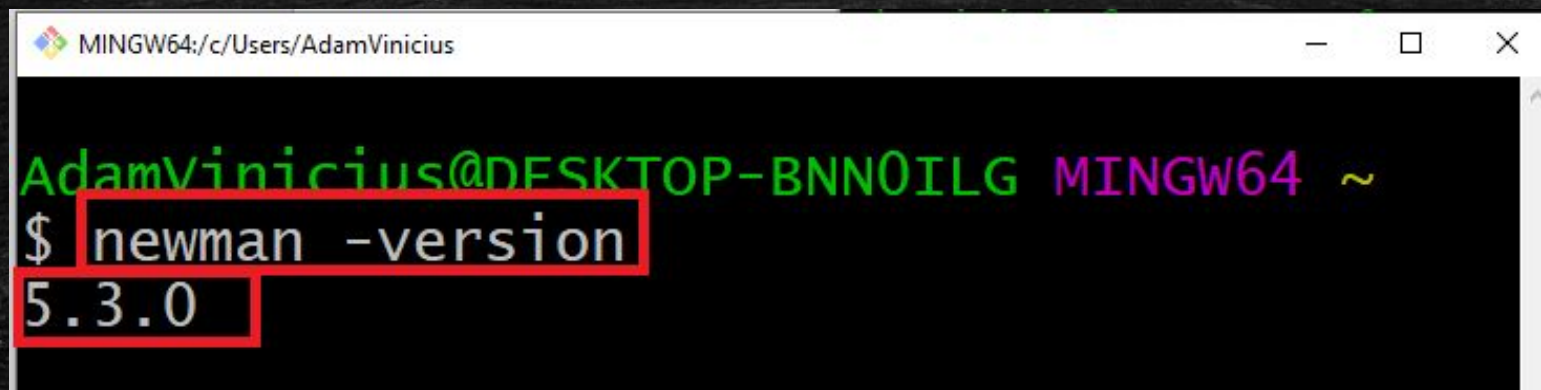


Na imagem anterior é possível identificar onde foi configurado  
o Newman  
E qual a versão instalada





Podemos também verificar a versão do Newman via linha de comando  
Utilizando Newman -version

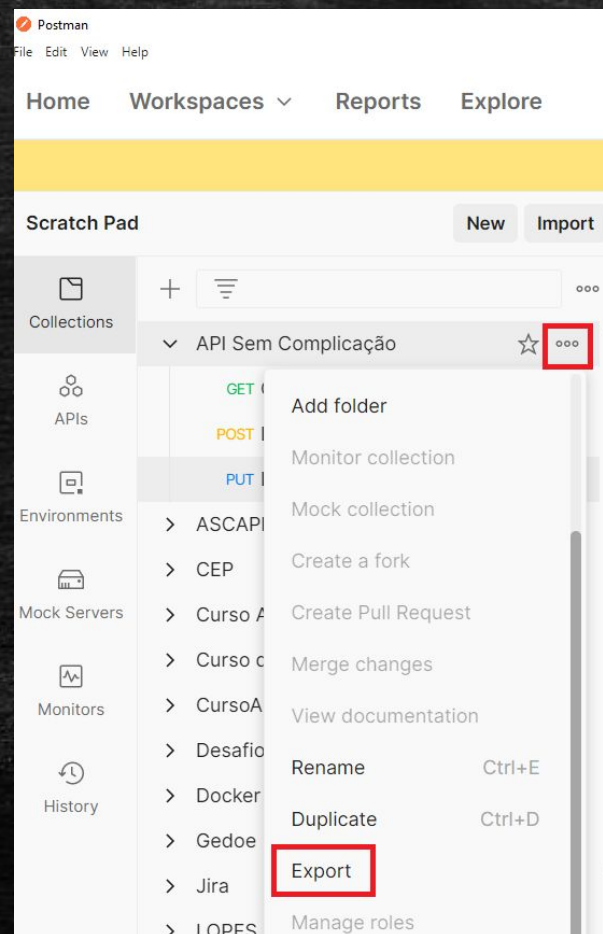


```
MINGW64:/c/Users/AdamVinicius  
AdamVinicius@DESKTOP-BNN0ILG MINGW64 ~  
$ newman -version  
5.3.0
```



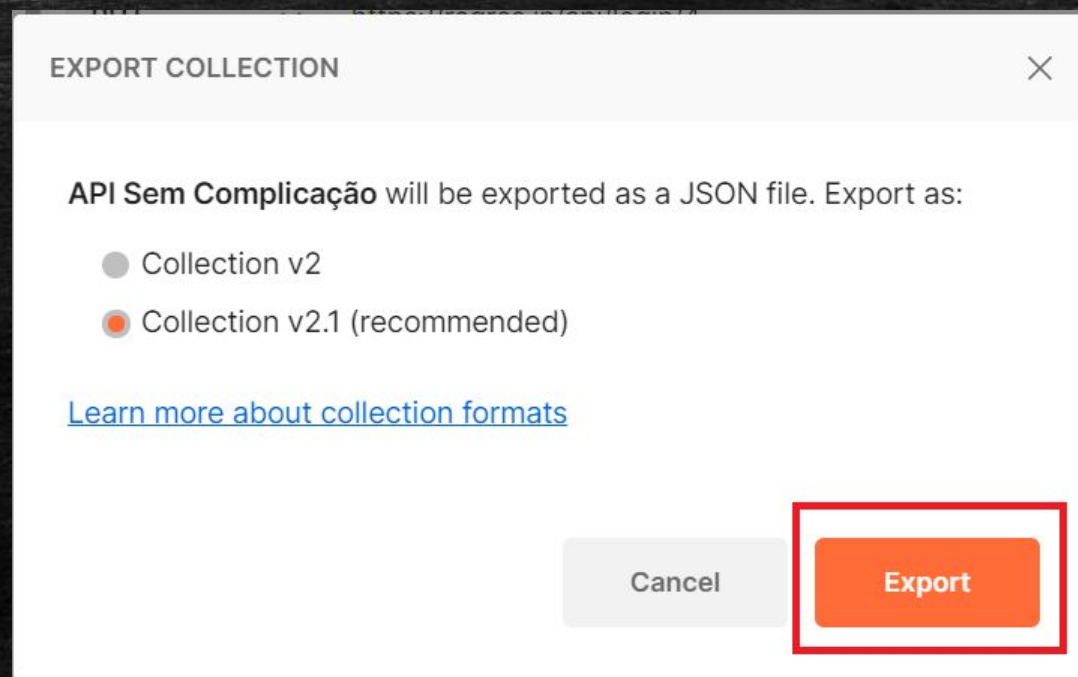


Com isso foi possível validar que a instalação realmente funcionou  
Para executar nossos testes precisamos exportar nossa collection  
Vamos para o Postman e clicar nos 3 pontos da collection



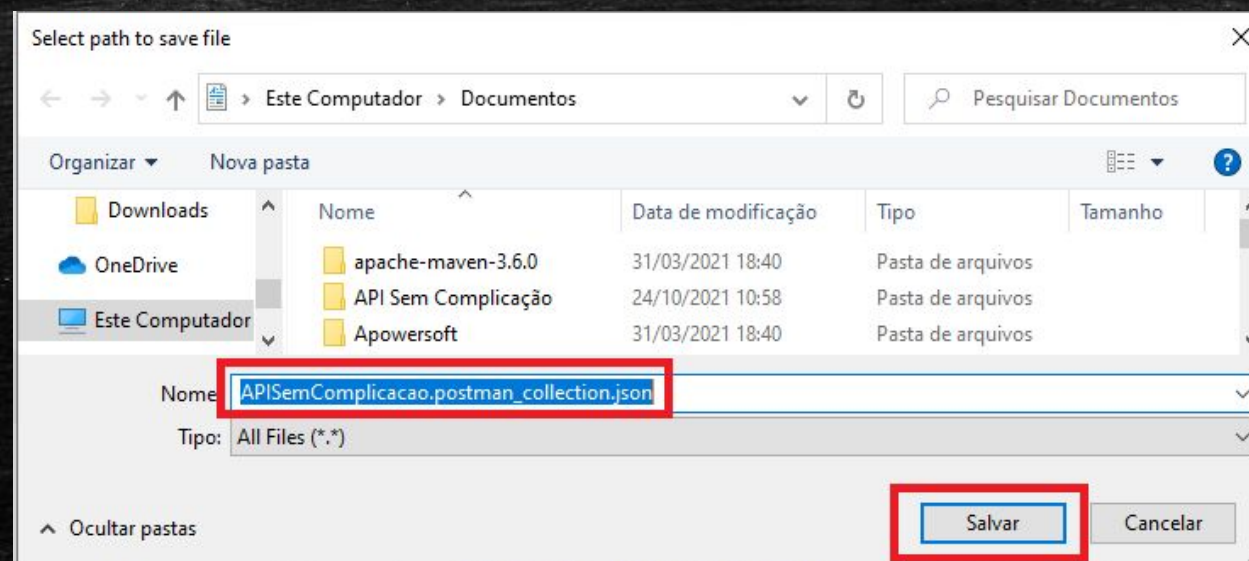


# Clicar no botão Export



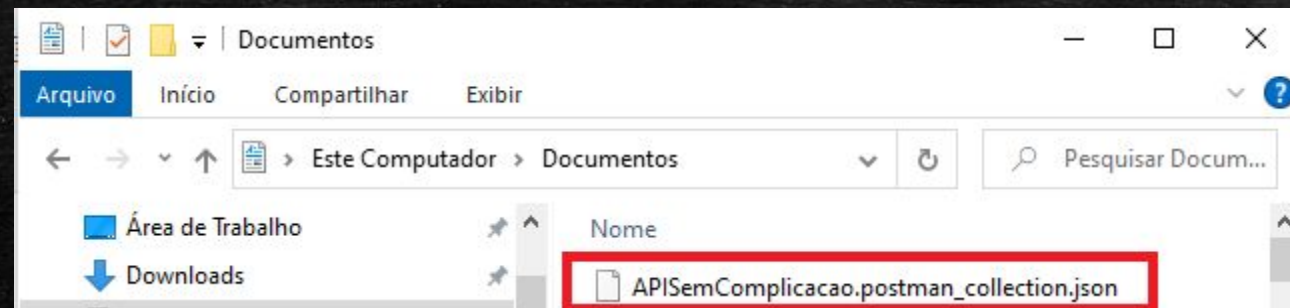


Basta escolher o nome do arquivo e o diretório  
E clicar no botão salvar



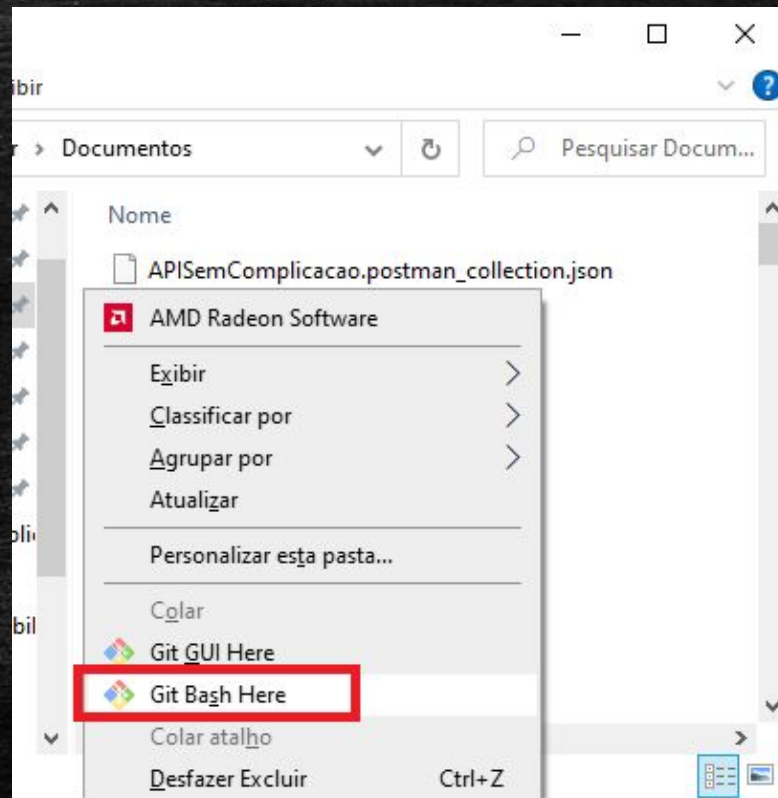


Podemos abrir o diretório e validar que o arquivo foi criado



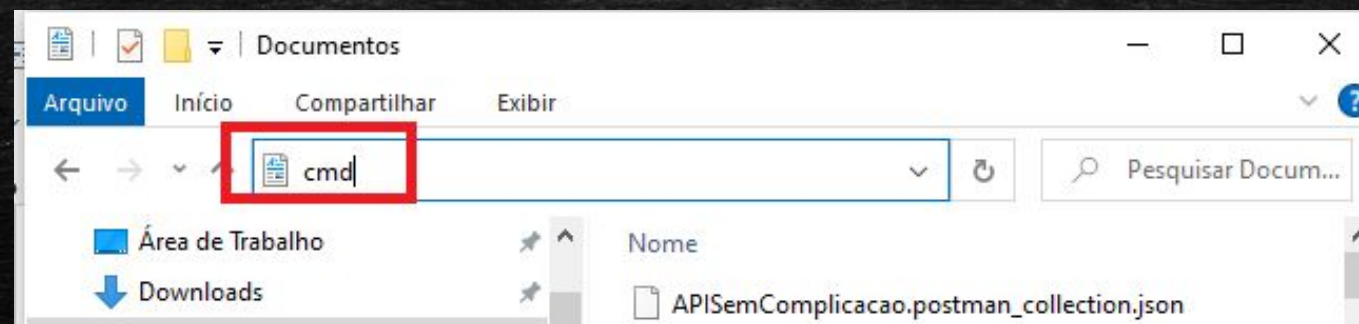


Agora que já temos o arquivo criado  
Vamos abrir a linha de comando nesse diretório que foi salvo  
Caso tenha o git bash instalado, basta clicar com  
o botão direito do mouse  
E ir git bash here





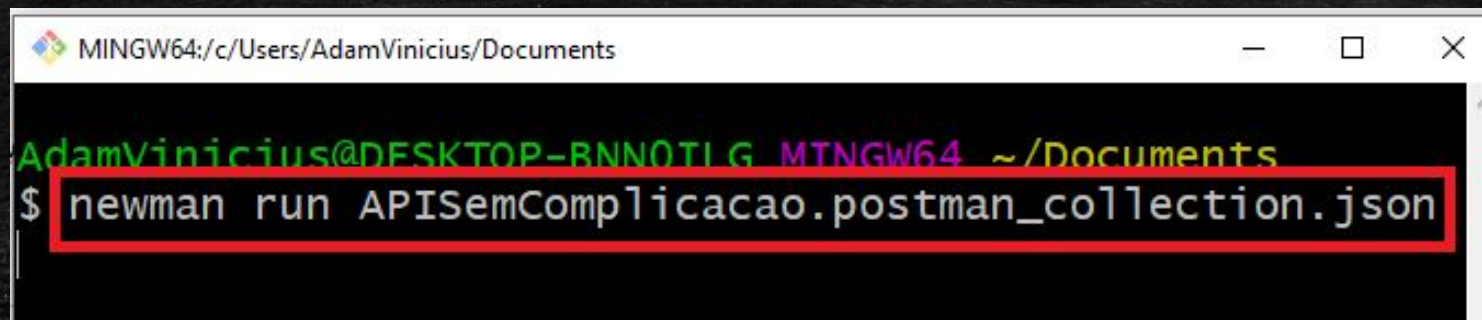
Caso for usar o cmd, basta escrever cmd no endereço do arquivo  
E pressionar Enter





Com o terminal aberto no diretório do arquivo recém criado  
Podemos executar o comando do Newman para realizar  
a execução dos testes

Sendo: `newman run nomeArquivoColection.json`



```
MINGW64:/c:/Users/AdamVinicius/Documents
AdamVinicius@DESKTOP-BNN0TIG MINGW64 ~/Documents
$ newman run APISemComplicacao.postman_collection.json
```





Basta pressionar Enter que as requisições  
E os testes serão executados

API Sem Complicação

→ CEP

GET http://viacep.com.br/ws/{{CEP}}/json/ [400 Bad Request, 737B, 373ms]

→ Login

POST https://reqres.in/api/login [200 OK, 884B, 478ms]

✓ Status code is 200

→ Login

POST https://reqres.in/api/login [200 OK, 882B, 460ms]

PUT https://reqres.in/api/login/4 [200 OK, 1.01kB, 464ms]

✓ Status code is 200

✓ Validate email

✓ Validate password





Mas temos um problema a nossa requisição do CEP  
Nos retornou 400, pois a mesma não encontrou as variáveis de ambiente

```
AdamVinicius@DESKTOP-BNN0ILG MINGW64 ~/Documents
$ newman run APISemComplicacao.postman_collection.json
newman

API Sem Complicação

→ CEP
  GET http://viacep.com.br/ws/{{CEP}}/json/ [400] Bad Request, 737B, 373ms

→ Login
  POST https://reqres.in/api/login [200 OK, 884B, 478ms]
]

✓ Status code is 200
```





Para resolver esse problema, vamos exportar o ambiente que criamos  
Voltando para Postman, vamos clicar no olho (que tudo vê rs)  
E ir em Edit

POST Login

PUT Login

CEP Adam

▼

👁

API Sem Com

CEP Adam

Edit

VARIABLE	INITIAL VALUE	CURRENT VALUE
CEP	04055041	04055041

Params

Auth





Clicar nos 3 pontos próximos do botão salvar  
E ir em Export

POST Login PUT Login CEP Adam X + ... CEP Adam

CEP Adam Fork Save Share ...

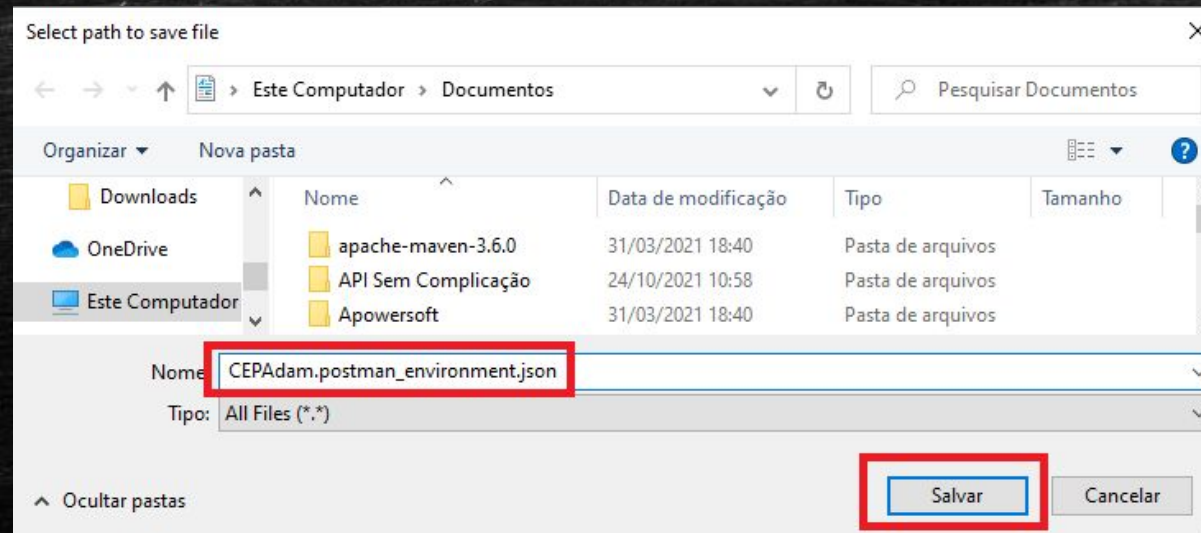
	VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ
<input checked="" type="checkbox"/>	CEP	04055041	04055041
	Add a new variable		

Set as active environment  
Export  
Duplicate Ctrl+D





Só escolher o nome do arquivo e salvar





Lembrar de salvar no mesmo diretório do arquivo anterior  
Podemos voltar para linha de comando e executar novamente o código  
Dessa vez vamos adicionar mais um ponto é o -e com o nome do arquivo  
Responsável por identificar qual ambiente que estará sendo executado

```
MINGW64; c:/Users/AdamVinicius/Documents  
AdamVinicius@DESKTOP-BNN0ILG MINGW64 ~/Documents  
$ newman run APISemComplicacao.postman_collection.json -e CEPAdam.postman_environment.json
```





Pressionamos Enter para iniciar a execução  
Nossa request foi corrigida

```
MINGW64:/c:/Users/AdamVinicius/Documents
AdamVinicius@DESKTOP-BNN0ILG MINGW64 ~/Documents
$ newman run APISemComplicacao.postman_collection.json -e CEPAdam.postman_environment.json
newman

API Sem Complicação
→ CEP
GET http://viacep.com.br/ws/04055041/json/ [200 OK, 737B, 375ms]
→ Login
POST https://reqres.in/api/login [200 OK, 886B, 507ms]
✓ Status code is 200
→ Login
POST https://reqres.in/api/login [200 OK, 886B, 419ms]
PUT https://reqres.in/api/login/4 [200 OK, 1.01kB, 445ms]
✓ Status code is 200
✓ Validate email
✓ Validate password
```





Também é possível verificar um report com as execuções

	executed	failed
iterations	1	0
requests	4	0
test-scripts	2	0
prerequest-scripts	2	0
assertions	4	0
total run duration: 2s		
total data received: 384B (approx)		
average response time: 436ms [min: 375ms, max: 507ms, s.d.: 47ms]		





Um ponto interessante de ser observado  
Executamos 4 requests e 4 testes  
Levando 2s para executar todas elas.

	executed	failed
iterations	1	0
requests	4	0
test-scripts	2	0
prerequest-scripts	2	0
assertions	4	0
total run duration: 2s		
total data received: 384B (approx)		
average response time: 436ms [min: 375ms, max: 507ms, s.d.: 47ms]		





Para refletir:

Quanto tempo se levar para realizar login em qualquer sistema web

Levando em consideração o tempo de o browser abrir e carregar a página?





Link para aprofundar mais sobre o Newman e Postman

<https://learning.postman.com/docs/running-collections/using-newman-cli/command-line-integration-with-newman/>





Muito obrigado por fazer parte dessa história da área de  
qualidade

Te espero na próxima aventura!!

Vamos juntos!?!?





# Obrigado, pessoal!



LinkedIn



YouTube



LinkedIn Pessoal

## Até a próxima!

