

CRIANDO PROJETO ANGULAR

PRÉ-REQUISITOS:

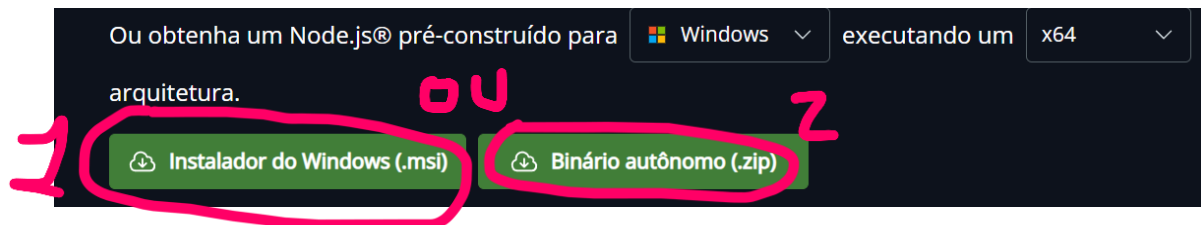
1 - Ter **node.js** instalado na máquina.

Para verificar se python está instalado **Abra o terminal** e digite “**node -version**”:

```
PS C:\Users\jonatas.lopes> node --version
v20.17.0
```

Se mostra a versão, está ok.

Senão, instale o node pelo link = [https://nodejs-
org.translate.goog/en/download/? x tr sl=en& x tr tl=pt& x tr hl=pt& x tr pto=tc& x tr hist
=true](https://nodejs.org.translate.goog/en/download/? x tr sl=en& x tr tl=pt& x tr hl=pt& x tr pto=tc& x tr hist=true)



2 – Ter **Angular CLI** instalado.

- **No mesmo terminal** digite “**npm install -g @angular/cli**”:

```
PS C:\Users\jonatas.lopes> npm install -g @angular/cli
```

*Pra testar se funciona, digite **ng version***

CRIANDO O PROJETO ANGULAR PELO TERMINAL:

(pule essa seção se quiser instalar direto pelo webstorm. Vá para pág.5)

1 – Vamos **criar um novo projeto Angular** chamado “projeto-angular”

A – QUAL TIPO DE CSS?

```
PS C:\Users\jonatas.lopes> ng new projeto-angular
? Which stylesheet format would you like to use? (Use arrow keys)
> CSS [ https://developer.mozilla.org/docs/Web/CSS ]
  Sass (SCSS) [ https://sass-lang.com/documentation/syntax#scss ]
  Sass (Indented) [ https://sass-lang.com/documentation/syntax#the-indented-syntax ]
  Less [ http://lesscss.org ]
```

B – SSR e Prerendering

```
? Do you want to enable Server-Side Rendering (SSR) and Static Site Generation (SSG/Prerendering)? (y/N) |
```

SSR = Renderização no lado do servidor

- O Angular **monta o HTML completo no servidor**, antes de enviar para o navegador.
- O usuário **recebe a página já pronta**, com conteúdo visível de cara.

Benefícios:

- Páginas carregam **mais rápido inicialmente**.
- Funciona melhor para **SEO (otimização para buscadores)**.
- Melhor experiência em conexões lentas.

SSG = Geração estática de páginas no momento da build

- Em vez de montar a página dinamicamente no servidor (como o SSR), o Angular **gera arquivos HTML prontos durante o build**.
- Esses arquivos são servidos **como páginas estáticas** (super rápidas).

Benefícios:

- Rápido como o SSR, **sem precisar de servidor Node rodando**.
- Ideal para páginas que **não mudam com frequência** (ex: blog, página institucional).
- **CSR (padrão)**: Quando você está fazendo um sistema web (ex: painel de controle, dashboard).
- **SSR**: Quando precisa de boa performance inicial e SEO (ex: e-commerce, notícias).
- **SSG/Prerender**: Quando o conteúdo muda pouco e precisa carregar rápido (ex: landing pages, blogs).

C – SERVER ROUTING, APP ENGINE APIS (DEVELOPER PREVIEW)

```
Do you want to enable Server-Side Rendering (SSR) and Static Site Generation (SSG/Prerendering)? Yes  
? Would you like to use the Server Routing and App Engine APIs (Developer Preview) for this server application? (y/N) |
```

Server Routing é a capacidade de o servidor **responder diretamente a URLs específicas**, em vez de deixar tudo para o Angular no navegador (como acontece no padrão SPA).

- Reconhece URLs como `/produto/123`
- Decide o que fazer com elas **antes mesmo de chegar ao Angular no navegador**
- Pode **gerar HTML pronto (SSR)**, fazer redirecionamentos, autenticação, etc.

E o que são "App Engine APIs"?

Esse recurso traz uma **forma leve de criar APIs** direto no Angular Universal (lado servidor), sem precisar configurar um servidor Node.js separado com Express, por exemplo.

Esse recurso ainda está **em fase de testes, não é estável para produção**.

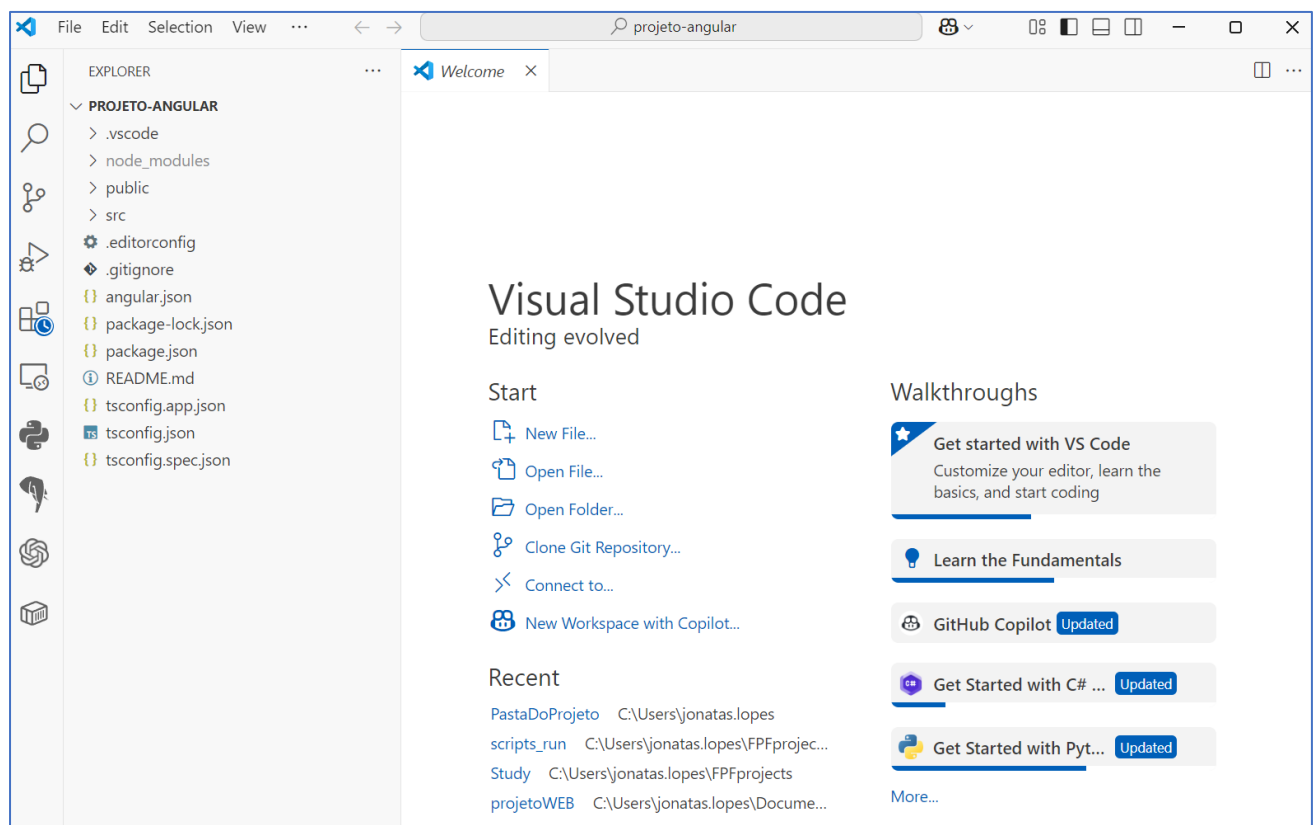
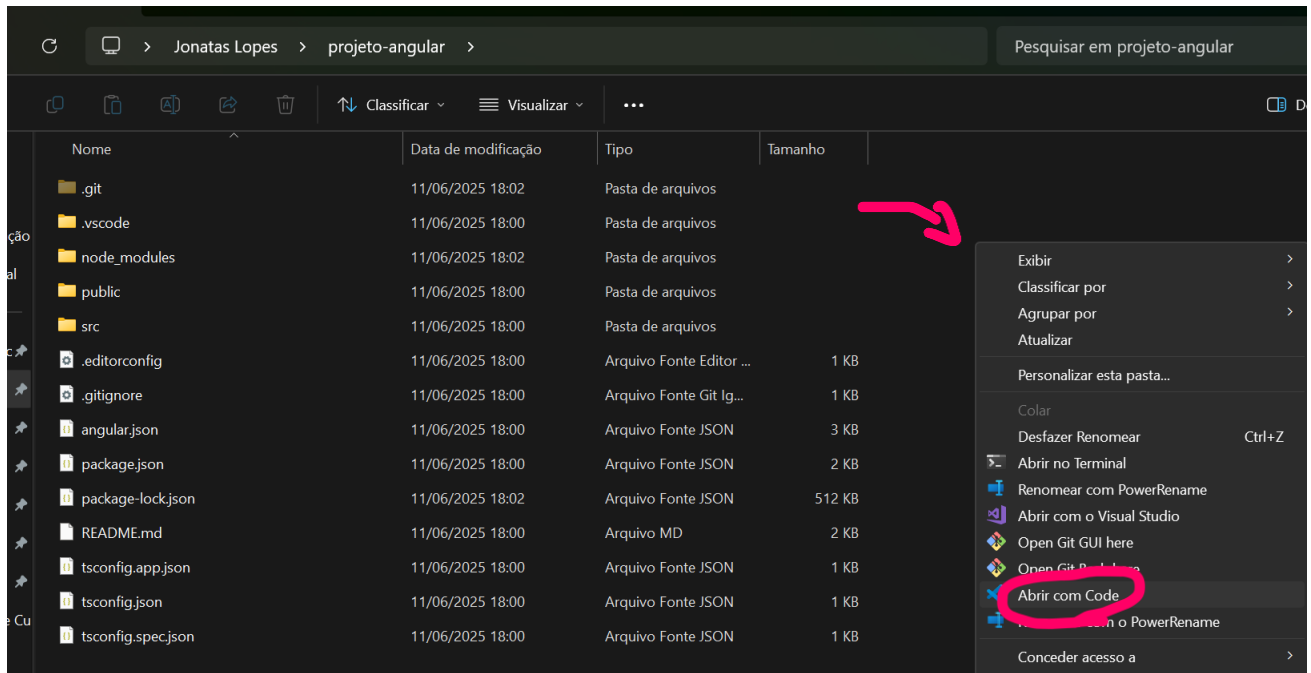
Ou seja:

- Está disponível para experimentar.
- A API pode mudar.
- Pode ter limitações ou bugs.

```
CREATE projeto-angular/public/favicon.ico (15086 bytes)
✓Packages installed successfully.
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
Successfully initialized git.
PS C:\Users\luciano\Documents>
```

VSCODE:

1 – Abrir a pasta criada do Angular > Clique botão direito > “Abrir com Code”



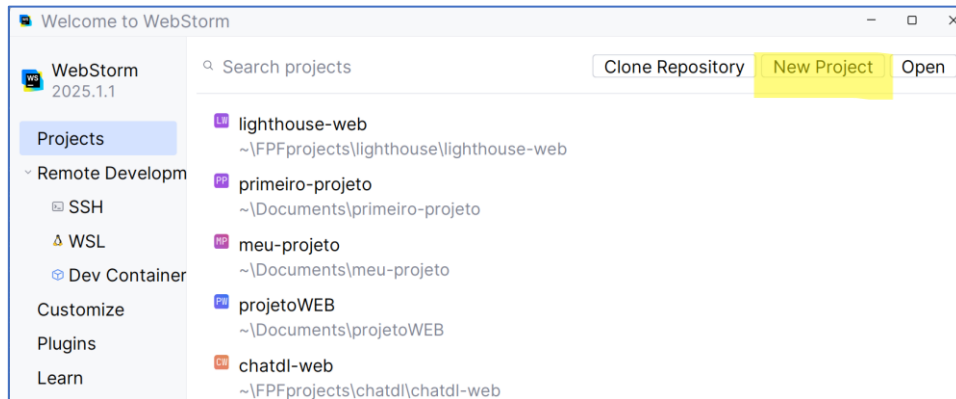
2 – Abrir o terminal do projeto “Ctrl + ‘ (apóstrofo) ”

- Fazer códigos do ng

WEBSTORM:

1 – Abra Webstorm, se tiver projeto aberto, Vá em **Settings** > “**Close Project**”

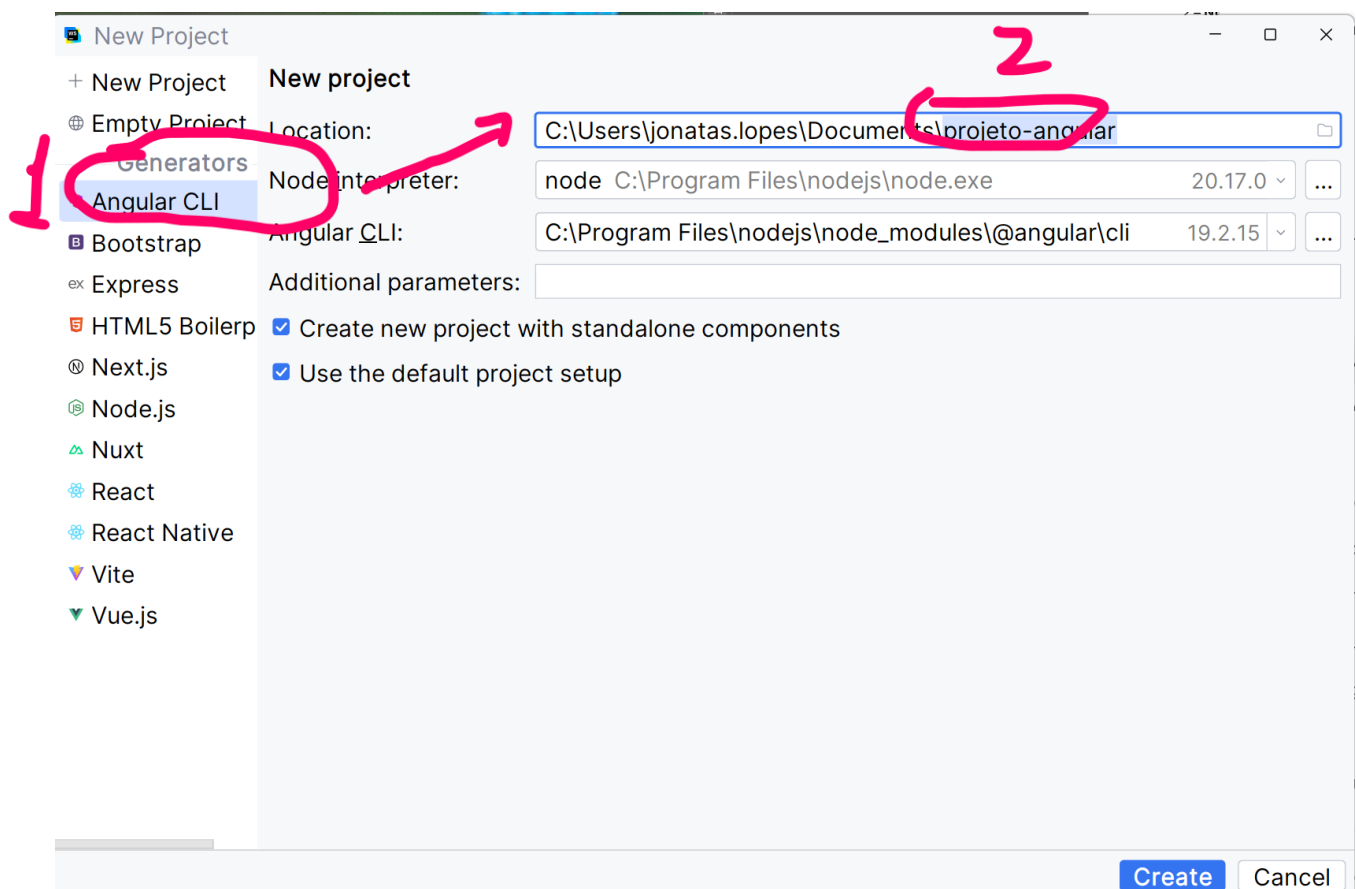
New Project



2 – Na janela de Novo Projeto, selecione:

1 – **Tipo do Projeto** = Angular CLI

2 – **Nome do projeto** = projeto-angular



3 – Já está rodando no Webstorm



PEGANDO O PROJETO DE UM GIT:

LEMBRE-SE: FAZER NPM INSTALL

DICA: organização de pastas, crie pastas:

- components

- interfaces

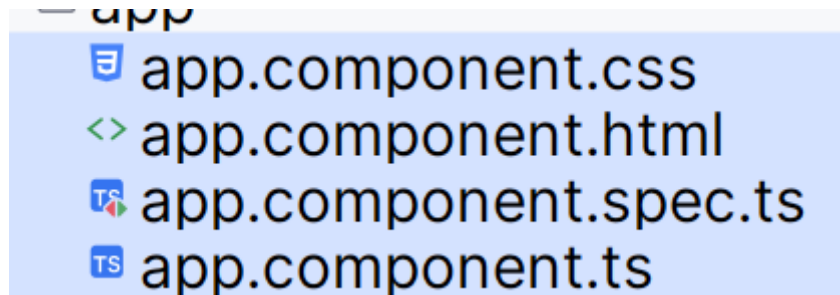
- services

DICAS ÚTEIS DE ANGULAR (NG):

1 – Criando novo componente chamado “exemplo” dentro da pasta components

```
documents\projeto-angular> ng g c components/exemplo
```

ATENÇÃO: Ao criar pasta componentes, crie uma pasta AppComponent e coloque esses arquivos lá dentro (SOMENTE ESSES 4 arquivos)



2 – Criando nova interface chamada “exemplo” dentro da pasta interfaces

```
documents\projeto-angular> ng g i interfaces/exemplo
```

3 – Criando novo serviço chamado “exemplo” dentro da pasta serviços

```
documents\projeto-angular> ng g s services/exemplo
```

Pra rodar

```
documents\projeto-angular> ng serve
```

Pra PARAR, no terminal (que está rodando) digite (Ctrl + C)