



## Angular CLI, instalação e criação do projeto

### Transcrição

Após todos os requisitos em relação à infraestrutura, instalação do **Node.js** e do editor **Visual Studio Code**, podemos dar início à criação do nosso projeto. Há um detalhe, porém: o Angular é um conjunto de diversos *frameworks*, *libs*, bibliotecas, e por aí vai. A configuração desta infraestrutura é essencial para construirmos nosso projeto, acessarmos ele em nosso navegador, e empacotarmos e gerarmos o arquivo para a produção. Gastaremos um tempo considerável antes de começarmos a programar efetivamente.

Então, vamos acessar o site do [Angular CLI \(https://cli.angular.io/\)](https://cli.angular.io/) (*Angular Command Line Interface*), que é cliente de linha de comando do Angular, atualmente patrocinado pela sua própria equipe. Trata-se de um projeto que nos auxilia na construção de projetos, a partir do zero, sem necessidade de nos preocuparmos com toda essa infraestrutura, bastando a execução de um único comando. Isso poupa bastante tempo, e é este o fluxo que vamos seguir no curso. Aliás, é o fluxo recomendado pela própria equipe do Angular.

Para instalarmos o Angular CLI, precisamos do **npm**, um gerenciador de pacotes do Node.js — sendo assim, o npm só estará disponível caso o Node esteja instalado na máquina. É uma ferramenta que baixa módulos, bibliotecas e *frameworks* da internet, para que possamos usá-los.

Vamos abrir o Prompt de Comando, sem nos preocuparmos com qual pasta estamos. Verificaremos se o npm está instalado com `npm --version`. Será retornado `6.1.0`, que é a sua versão. Para solicitar a instalação do Angular CLI globalmente, ou seja, para que esta ferramenta esteja disponível em qualquer

diretório da aplicação, utilizaremos o comando `npm install -g @angular/cli@6.0.7`.

Acrescentamos `@6.0.7` para indicar que usaremos esta versão específica, já que na data de gravação deste curso, esta é a versão mais recente. Deste modo garantimos paridade entre o que faremos aqui e o que poderá ser feito posteriormente. Se deixarmos o comando sem este final, a versão mais recente será instalada automaticamente, e não é isto que queremos.

Outro detalhe: para que o `-g` funcione, é necessário privilégio de administrador no Prompt de Comando em uso, ou no Terminal, no caso do Linux. Este comando é executado apenas uma vez por máquina de desenvolvimento, e ele irá acessar a internet e baixar o Angular CLI. O processo é rápido e, para confirmarmos sua instalação, usaremos o `ng`, que é o módulo do Angular CLI, colocado globalmente para uso no terminal: `ng --version`. Serão exibidas as versões instaladas de todos os módulos, inclusive do TypeScript do RxJS.

Para começar o nosso trabalho, de que forma criaremos um projeto?

Com o comando `cd Desktop`, acessaremos a área de trabalho, e solicitaremos ao `ng` o `alurapic`, que é o projeto a ser desenvolvido, um sistema de gerenciamento de fotos. O comando será `ng new alurapic`, e desta vez o processo costuma demorar um pouco, pois quando se cria um projeto com Angular CLI, ele vem com pré-configurações de todos os módulos que a nossa aplicação necessitará, isto é, vários downloads são feitos por meio do npm.

Após a finalização do processo, aparecerá uma mensagem que pode parecer preocupante, mas não é: "found 13 vulnerabilities <9 low, 4 high>", isto é, foram encontradas 13 vulnerabilidades, 9 baixas e 4 altas neste projeto. A partir das versões mais novas, o npm começou a ter um sistema de análise da consistência e da segurança dos módulos utilizados pela aplicação.

Isso quer dizer que os módulos baixados pelo Angular CLI possuem algumas vulnerabilidades. No entanto, o responsável pela correção delas não é a equipe do Angular CLI, e sim de quem criou estes módulos, que são numerosos, e sobre os quais não temos controle. O mais importante é que isso não nos afeta em nada, pois o que baixamos são dependências de desenvolvimento, os quais não entrarão em produção.

Se quiséssemos um servidor em Node, por exemplo, e algum módulo tivesse vulnerabilidades, pode ser que fosse algo problemático mas, no nosso caso, não há problema algum.

Criado o nosso projeto, se acessarmos a área de trabalho, encontraremos a pasta "alurapic". Então, poderemos utilizar os comandos `cd alurapic`, e em seguida `ng serve --open`. Isso subirá um servidor local configurado pelo Angular CLI usando-se boas práticas de desenvolvimento, e automaticamente abrirá o navegador em uma página padrão. Este projeto será simples porque iremos personalizá-lo utilizando a estrutura criada para o desenvolvimento do nosso projeto.