

# Angular 2

## Blocos principais



## Componentes:

## Componente



### *Encapsula:*

- Template
- Metadata: processamento das classes
- Dado a ser mostrado na tela (Data Binding)
- Comportamento da VIEW

Os **componentes** seria a VIEW é o que o usuário vai ver e dentro dele temos:

**TEMPLATE:** que é o código HTML que a gente vai mostrar para o usuário (pode ser um formulário, div, botões, data-grid).

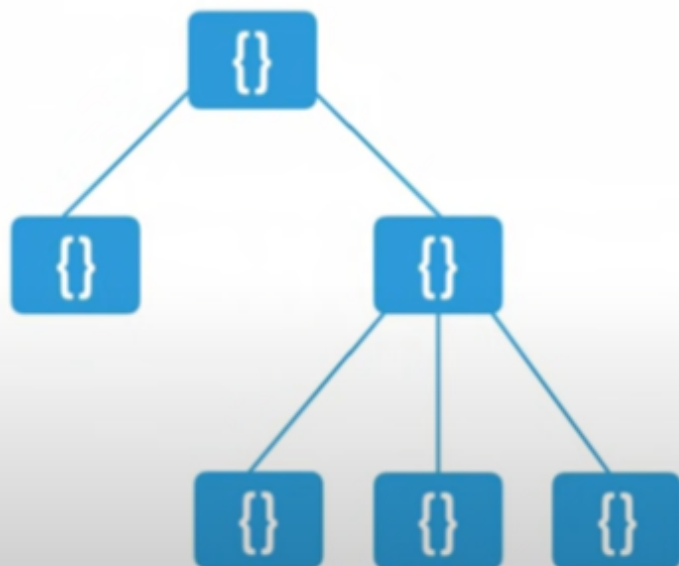
**METADATA:** Metadados que permitem o Angular 2 ler as classes e fazer o processamento das mesmas.

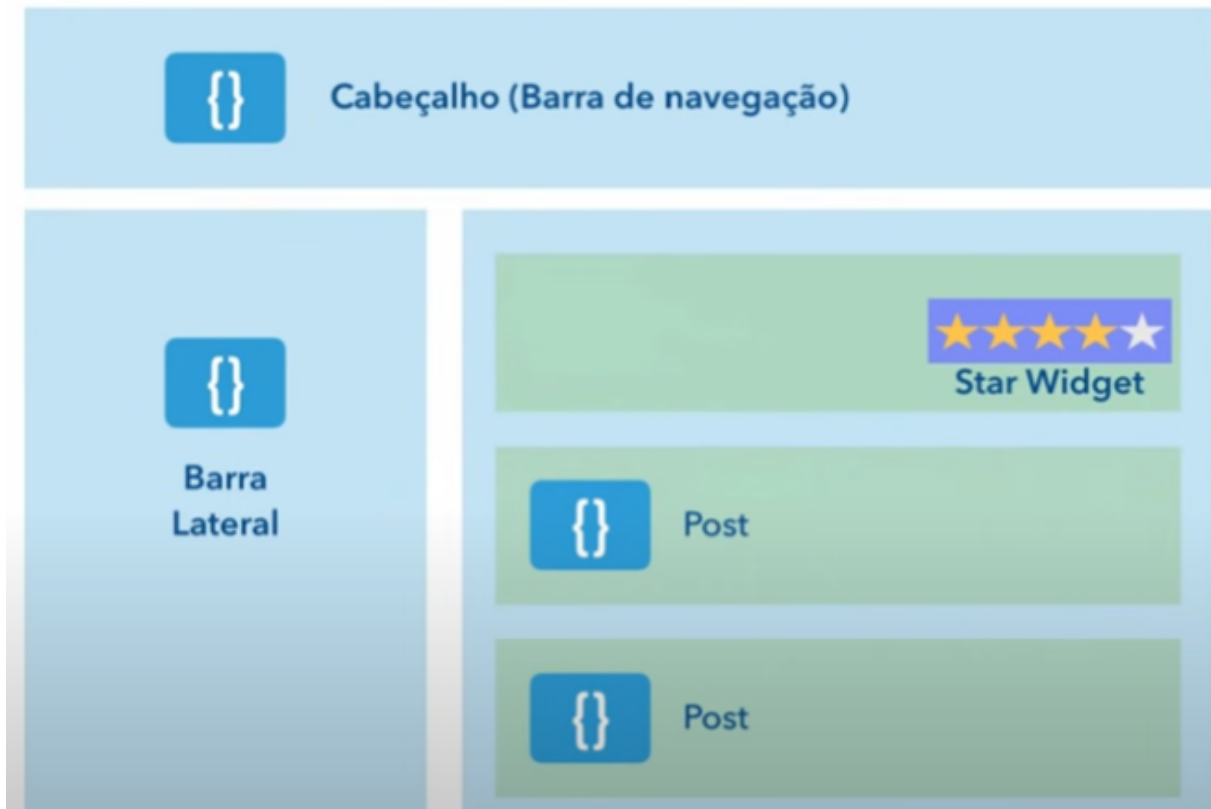
**Data Binding:** Já que usamos template nós temos o data binding que é a associação dos dados que temos no nosso componente com os componentes HTML que temos no template.

Ou seja o componente ele consegue conversar com o template mostrando dados através do data binding, além de ser responsável pelo comportamento da view.

Tudo que temos na nossa aplicação é um componente PAI ou MÃE (barra de navegação, lista, barra lateral, etc.) e cada componente pode ter outros dentro dele, isso traz o benefício de ser muito mais fácil de se testar algo separadamente e encontrar os erros de forma mais rápida.

## Componente Raiz (Root)





Como principal objetivo do componente é mostrar dados podemos fazer a integração com o Backend (servidor) e não importa a linguagem que usaremos.

E como é de bom tom e padronizado, não escrevemos códigos de lógica de negócio nos nossos componentes e sim no **SERVICE** que será responsável por ter toda lógica de negocio e se comunicar com o backend e pode ser injetado em outras classes e para isso usamos **injeção de dependência** do Angular.

## Roteamento:

Para termos mais de uma tela no Angular usamos o ROUTER que é o responsável pela navegação no Angular2 que nós utilizamos para fazer o roteamento das páginas

## Router



Responsável pela navegação

## Diretiva:

Responsável por modificar elementos DOM e/ou seu comportamento (componentes também são diretivas, pois modifica elementos DOM, seja fazendo display de um formulário ou div, etc), temos também as diretivas cujo principal função é modificar elementos DOM como por exemplo a **autoGrow** (faz o elemento crescer de tamanho ao ficar no foco)

Angular

```
<input type="text" autoGrow>
```