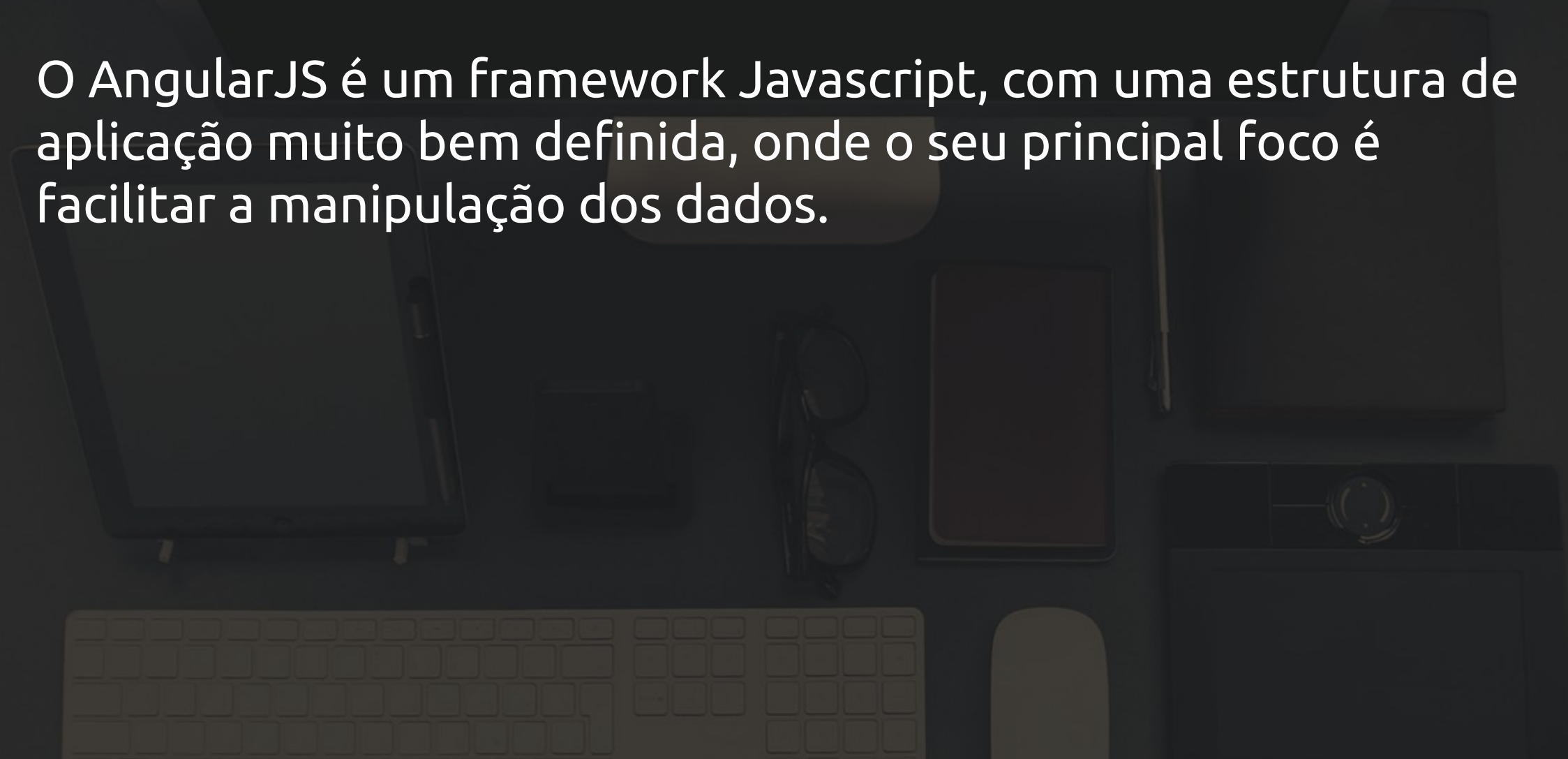




Oelton Kamchen

Sobre

O AngularJS é um framework Javascript, com uma estrutura de aplicação muito bem definida, onde o seu principal foco é facilitar a manipulação dos dados.



Ponto Forte

- Facilidade em trabalhar com o Javascript;
- Utiliza a própria linguagem HTML como marcação e template;
- Permite a criação de seus próprios componentes;
- Grande potencial para a redução de código;
- Infraestrutura pronta para integrar com back-end, de modo “ajax”;
- Facilidade em automação de testes;
- Documentação e comunidade;

Origem

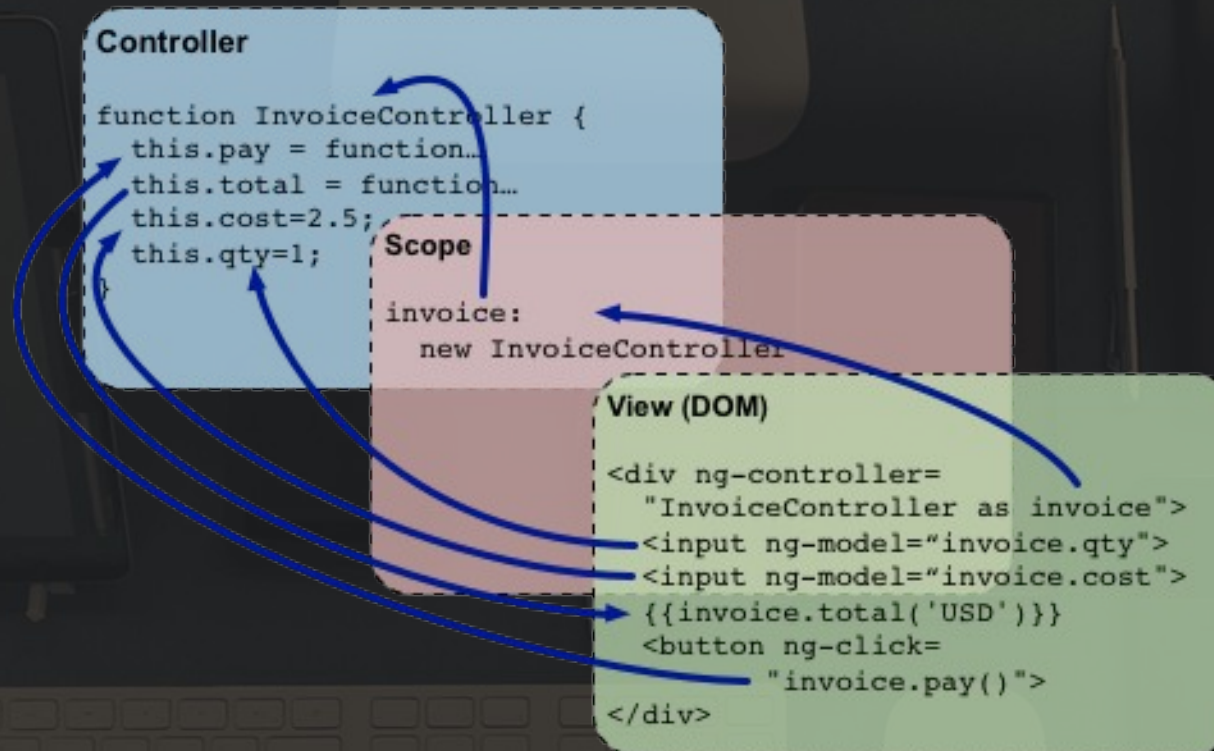
O AngularJS começou como um projeto pessoal do Miško Hevery em 2009.

Logo após, Hevery foi trabalhar no projeto Google Feedback, onde aplicando o AngularJS, conseguiu reduzir o número de linhas de código de 17 mil linhas, para 1500, em apenas 3 semanas de trabalho.

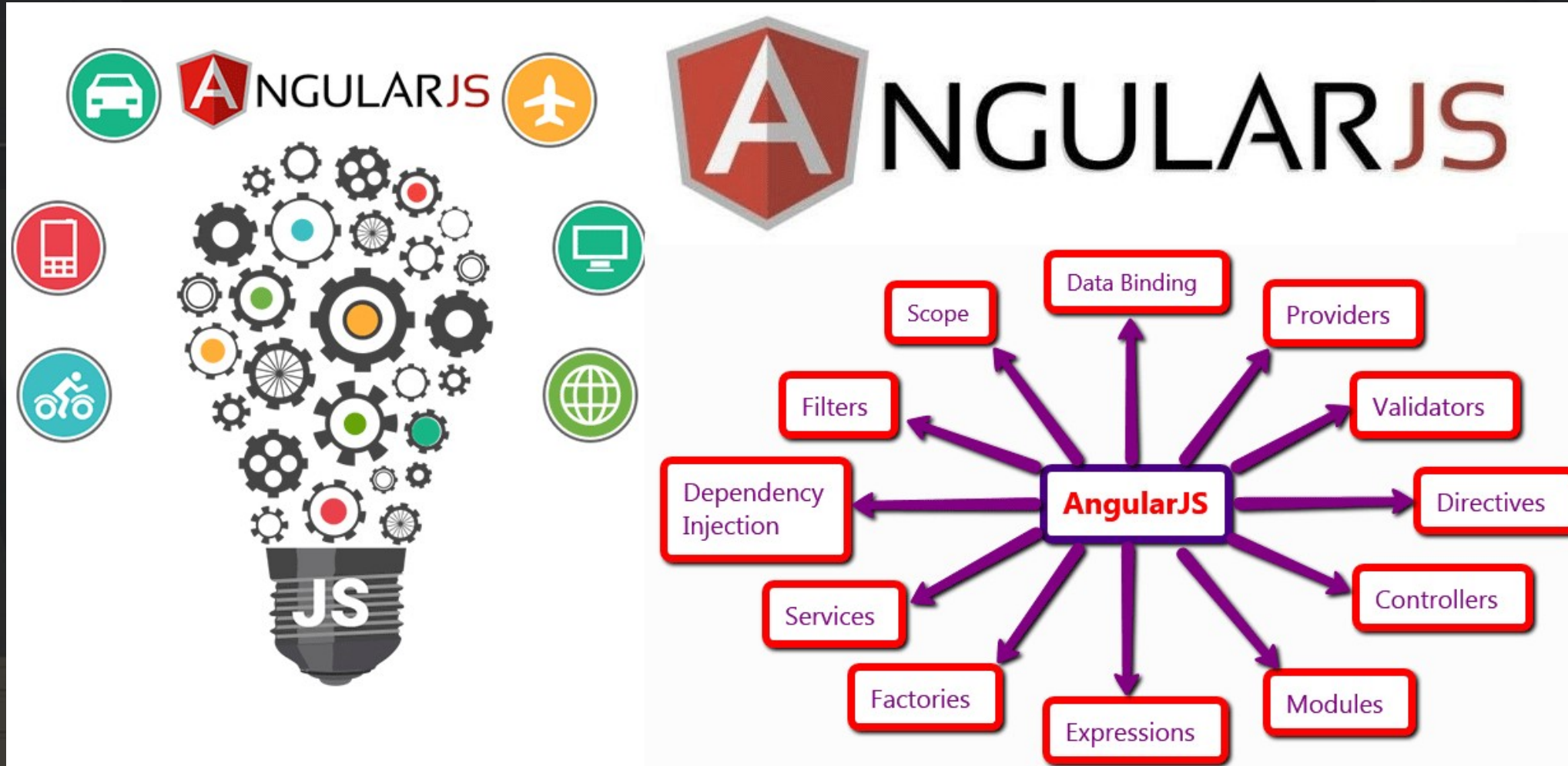
Conceitos básicos... MV*

- Controller: Responsável pela manipulação e transformação dos dados;
- Scope: responsável pelo transporte dos dados entre Controllers e Views;
- View(DOM): responsável pela exibição e alteração dos dados;

Conceito básico...



Blz, vamos continuar...



Views

Utilizando os documentos HTML como template, as views poderão ser divididas em partes, deixando assim, cada uma com sua responsabilidade. Com isso, nos permite uma reutilização de código e diminuição da complexidade da aplicação.

DataBind

Usado para ligar uma variável do JavaScript (ou um objeto) a algum elemento do documento HTML. E para que este elemento seja ligado ao nosso template, é necessário a utilização da variável “\$scope”, sendo ele o escopo da aplicação que pode ser utilizada no template da controller criada.

Controllers

Contém funcionalidades pertinentes à alguma parte do documento HTML. Não existe uma regra para o controller, como por exemplo ter um controller por arquivo HTML, mas sim uma forma de sintetizar as regras de negócio (funções javascript) em um lugar separado ao documento HTML.

Filters

Filters são a manipulação dos dados em tela, para a transformação para obter os dados ou exibir de outra forma os dados, transformando os dados de entrada, em dados de saída pretendidos.

Services e Resource

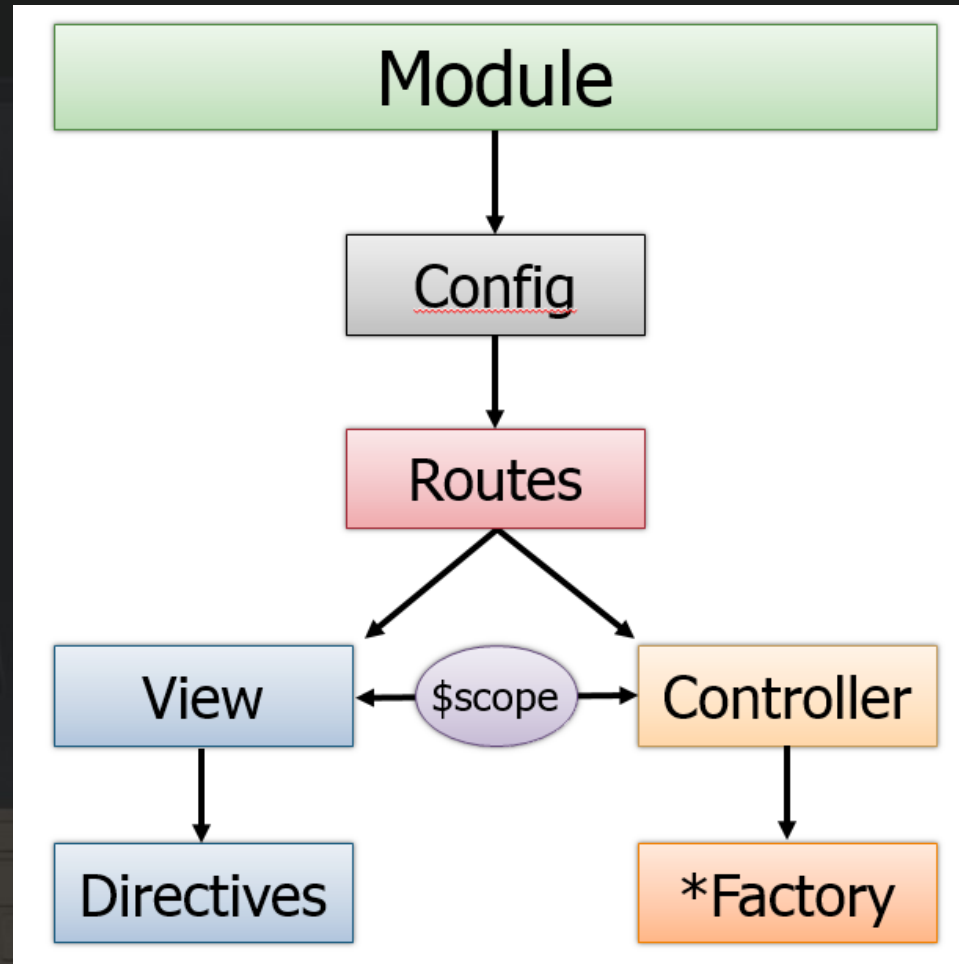
O service será o responsável pelo acesso aos serviços disponibilizados pela API, e com a injeção da dependência de “\$resource” irá tornar esse acesso mais abstrato. Desta forma, nosso front-end vai realizar chamadas para URL's determinadas na API, cada qual com sua responsabilidade, sendo ela consultar, inserir, alterar ou deletar.

Diretivas

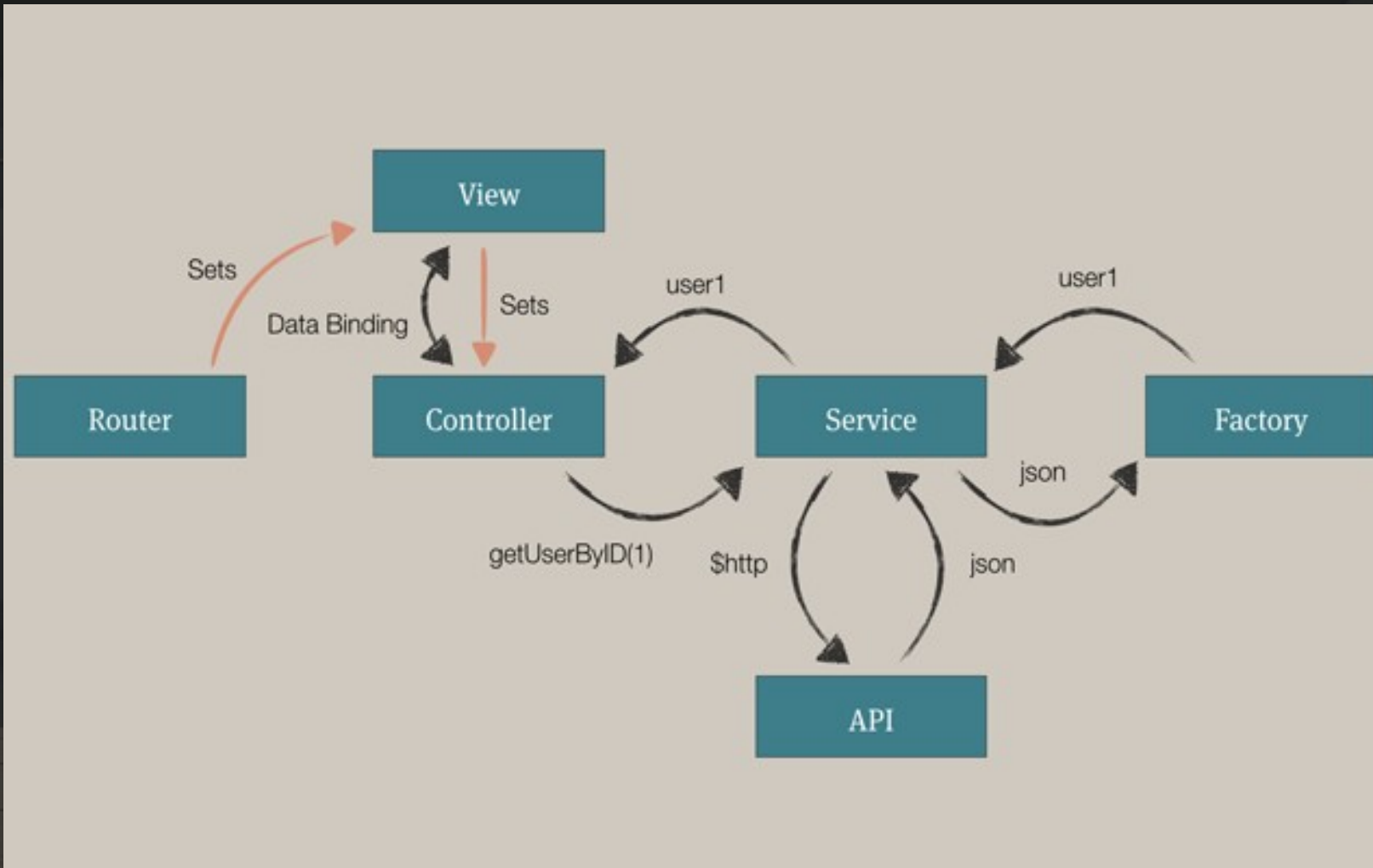
A diretiva é um atributo especial que pode adicionado a qualquer elemento do html. Exemplos de diretivas são “ng-app” , “ng-model” entre outras.

Além das diretivas que o AngularJS possui, podemos criar diretivas personalizadas de acordo com as necessidades do nosso projeto. A ideia aqui é que possamos criar funcionalidades extras que serão reutilizadas em qualquer parte do nosso código.

Estrutura Final



Conversação entre camadas



Referência

<https://angularjs.org/>

<http://blog.caelum.com.br/como-anda-o-angular-js-devo-embarcar-nessa/>

https://www.youtube.com/watch?v=_y7rKxqPoyg&list=PLQCMsnNFVYnTD5p2fR4EXmtlR6jQJMbPb

Git

A dark, top-down view of a desk with various items: a large computer monitor at the top, a tablet on the left, a smartphone in the center, a pair of glasses below the phone, a pen to the right of the phone, a keyboard at the bottom left, and a mouse at the bottom right. The background is a dark, textured surface.

<https://github.com/SIFeevale/Web-design-2/tree/oelton/apresentacao-angularjs>