

## Colocando em prática

Nosso cadastro é funcional, mas ainda precisamos cadastrar o grupo a qual cada foto pertence. Nosso back-end possui a rota `v1/grupos`, que retorna uma lista de grupos disponíveis. Integração do front-end com o back-end não é novidade para nós, porém a novidade neste exercício é representar os grupos retornados do servidor como uma combobox em `foto.html`, inclusive garantindo que as opções aparecerão sempre em caixa alta. Para completar, criaremos no final uma nova diretiva que substituirá o botão `Remove` em `principal.html`. Preparado? Sem perder o hábito, seguem os passos fundamentais que devem ser seguidos, mas sem fazer *spoiler* da solução:

1) O novo campo, isto é, nossa combobox ficará imediatamente após o campo descrição. Para ajudá-lo, segue a "cola" da marcação com o código de validação que já aprendemos, mas sem as novidades deste capítulo:

```
<!-- novidade aqui! -->
```

```
<div class="form-group">
  <label>Grupo</label>
  <select name="grupo" class="form-control" required>
    <option value="">Escolha um grupo</option>
  </select>
  <span ng-show="formulario.$submitted && formulario.grupo.$error.required" class="foi
    Grupo obrigatório
  </span>
</div>
```

2) Um controller será necessário para fornecer os dados que popularão a combobox! Chame-o de `GruposController`. Acesse a URL (endpoint) `v1/grupos` através de `$http.get`. Não há novidade, "recordar é viver".

3) Angular possui uma diretiva especial para criar os `option` de um `select`. Difícil lembrar?

4) Não basta popularmos a lista, precisamos de alguma maneira capturar a opção escolhida. Você precisará da diretiva `ng-model` para guardar o grupo selecionado diretamente na foto que estamos cadastrando.

5) Mais uma novidade! Bootstrap é fantástico, mas termos que decorar zilhões de classes não é uma tarefa trivial. Já isolamos a complexidade da marcação do `panel-default` em uma diretiva, agora faremos a mesma coisa, criando a diretiva `meuBotaoPerigo`. Quando usarmos essa diretiva, sequer precisaremos adicionar as classes do Bootstrap, justamente por estarem encapsuladas em sua definição. Sua interface de uso será `nome e acao`. O 'nome' usa o já conhecido modificador `@` no escopo isolado da diretiva, porém, `acao` usará um novo. Lembra dele? Ele é usado quando queremos avaliar uma expressão dentro da diretiva. Essa expressão avaliada geralmente roda no contexto de um controller e a diretiva nada sabe do que acontece.

6) Com a diretiva criada, faça a troca do botão `Remove` por ela, não esquecendo de passar todos os dados que ela precisa através de uma interface.

Mãos à obra e bom exercício!

