

## Implementando spinner

### Transcrição

Um outro recurso interessante para nosso usuário é mostrar que estamos executando a requisição AJAX através de uma barra de progresso e outro recurso. Assim o usuário fica sabendo que a aplicação está trabalhando por baixo dos panos que é útil quando a rede ou servidor demora.

É muito comum, quando se usa AJAX, mostrar o componente spinner para indicar a execução como apresentado na imagem abaixo:



### Aplicando o spinner

Na nossa aplicação, dentro da pasta `public/img` já existe um arquivo `spinner.gif` para esse propósito. Agora só falta mostrar essa imagem enquanto a requisição AJAX está sendo executada.

Vamos adicionar essa imagem dentro da `div` da mensagem de erro, mas antes do parágrafo do erro:

```
<div class="center">
  
  <p id="erro">Ocorreu um erro, por favor tente novamente</p>
</div>
```

Ao atualizar a página já aparece a imagem. Agora só falta ativar e desativar programaticamente

### Toggle do spinner

O spinner deve começar invisível, e ao executar a requisição AJAX, devemos mostrar e depois esconder.

Para começar invisível usamos o CSS, dentro do arquivo `estilos.css` :

```
#spinner{
  display: none;
}
```

Para mostrá-lo quando a requisição AJAX está sendo executada é preciso alterar o arquivo `frases.js` . O spinner deve aparecer ao clicar no botão ( `#botão-frase` ). No início da função `fraseAleatoria` colocaremos:

```
$("#spinner").toggle(); //mostrando o spinner
```

E depois da requisição AJAX devemos esconder o spinner. Importante é que **sempre** devemos esconder o spinner, tanto faz se a requisição concluiu com sucesso ou falhou.

Para tal o jQuery oferece uma nova função `always` que podemos chamar depois do `fail`. Segue o código completo da função `fraseAleatoria`:

```
function fraseAleatoria() {  
  
    $("#spinner").toggle();  
  
    $.get("http://localhost:3000/frases", trocaFraseAleatoria)  
    .fail(function(){  
        $("#erro").toggle();  
        setTimeout(function(){  
            $("#erro").toggle();  
        }, 1500);  
    })  
    .always(function(){ //sempre escondendo o spinner  
        $("#spinner").toggle();  
    });  
}
```

Ao testar podemos ver que aparece e desaparece o spinner, indicando ao usuário que a requisição AJAX está sendo executada. Repare também a nova função `always` para invocar um código *sempre* (erro ou sucesso) após da requisição AJAX.

## O que aprendemos:

- Melhorando a experiência do usuário
- Criando um spinner na aplicação
- A função `.always()` do jQuery