

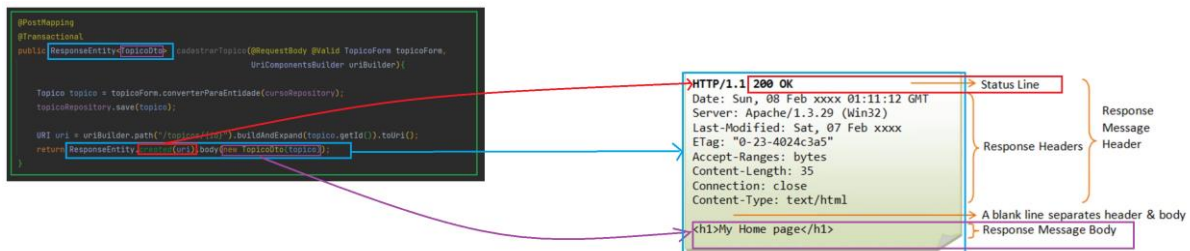


Response com ResponseEntity<T> - Spring Web

Uma API REST precisa devolver respostas de acordo com o padrão Rest. Então nossa resposta não se limita só ao **conteúdo respondido como Json**, mas também precisa devolver **status correspondentes** ao resultado do request.

O **Spring Web** já fornece algumas **respostas padrões com status padrões**, mas podemos ter ainda mais controle usando o **ResponseEntity<T>**.

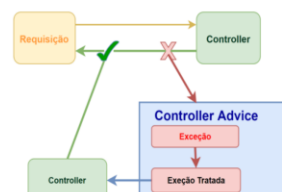
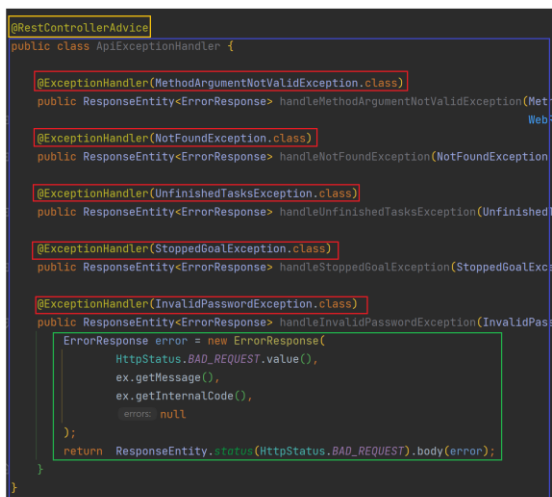
O **ResponseEntity<T>** é um wrapper que representa justamente a resposta de uma requisição rest, ele pode devolver a **resposta altamente personalizada através do padrão Builder**, construindo **status específicos**, **headers específicos**, tudo que envolve uma **resposta REST**:



Erros com RestControllerAdvice - Spring Web

O **ResponseEntity<T>** também é capaz de devolver status de erro e **poderia fazer isso diretamente no controller**, isso faria com que existisse mais linhas de código, não é exatamente errado, mas é **possível seguir por uma abordagem mais limpa, que é usando um Interceptador de exceções**.

Esse **interceptor** vai ser um **Bean do Spring**, anotada com **@RestControllerAdvice**, dentro dessa **classe** eu vou avisar **qual exceção ela vai interceptar** (podem ser diversos tipos de exceção e ela se aplica a todos os controllers), quando essa exceção for lançada, a **classe** vai captura-la e **tratar** para devolver uma resposta melhor ao cliente:



Uma abordagem seguida para as exceptions é lança-las como **RuntimeException**, então em qualquer ponto elas podem ser lançadas que serão interceptadas. Também é uma prática comum usar um **DTO de resposta padrão de erro**.