



Lidando com diferentes exceptions

Caso você precise do projeto com todas as alterações realizadas na aula passada, [você pode baixá-lo neste link \(https://github.com/alura-cursos/flutter-webapi-2/archive/aula-2.zip\)](https://github.com/alura-cursos/flutter-webapi-2/archive/aula-2.zip).

Adicione mais um `catchError()` para capturar a exception de timeout (`TimeoutException`), então, apresente um dialog de falha indicando a mensagem esperada para essa exception.

Em seguida, crie a exception genérica para problemas de HTTP (`HttpException`) e ajuste o código para que capture a `HttpException` em vez de `Exception` .

Por fim, adicione um timeout no `save` do web client para simular o problema de timeout. Para facilitar a implementação, configure o timeout direto no client.

Lembre-se de que é preciso passar um endereço válido que não tem nenhum serviço para simular o timeout.

Faça o teste no app e veja se apresenta a mensagem de timeout e se as demais mensagens de falhas ainda funcionam como esperado.