

37. Personalizando o retorno da validação

Quando tentamos inserir uma categoria com nome vazio, menos de 5 ou mais de 80 caracteres o endpoint /categorias dispara uma exceção que retorna uma lista de erros com algumas informações não necessárias, por isso ao capturar essa exceção vamos formatá-las e retirar o que não é preciso.

resources/exceptions/ResourceExceptionHandler é responsável por interceptar as exceções.

resources/exceptions/FieldMessage é uma classe auxiliar que armazena o nome do campo e a mensagem de erro gerada.

resources/exceptions/ValidationError é uma classe especializada de StandardError. ValidationError contém uma lista chamada errors do tipo FieldMessage.

resources/exceptions/ResourceExceptionHandler.java

- Intercepta exceções
- Tratar MethodArgumentNotValidException gerada pelo endpoint /categorias como citado no primeiro parágrafo.
- Usar o esqueleto de outro método, alterações são necessárias:
 - Nome do método.
 - Parâmetro MethodArgumentNotValidException.
 - Instância de ValidationError no corpo do método.
 - ForEach no corpo do método.

e.getBindingResult().getFieldErrors() retorna a lista de erros associados a um campo.

<https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/validation/Errors.html#getFieldErrors-->

<https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/validation/FieldError.html>

O nome do campo e mensagem padrão são passados como parâmetro para o método addError(), este método adiciona um novo FieldMessage na lista errors da instancia ValidationError da classe ValidationError.

```
@ExceptionHandler(MethodArgumentNotValidException.class)
public ResponseEntity<StandardError> validation(MethodArgumentNotValidException e, HttpServletRequest request) {
    ValidationError validationError = new ValidationError(HttpStatus.BAD_REQUEST.value(), msg: "Erro de validação.", System.currentTimeMillis());
    e.getBindingResult().getFieldErrors().forEach(error -> {
        validationError.addError(error.getField(), error.getDefaultMessage());
    });
    return ResponseEntity.status(HttpStatus.BAD_REQUEST).body(validationError);
}
```

resources/exceptions/FieldMessage.java

- FieldMessage é uma classe auxiliar para formatar os dados da exceção.
- São implementados dois construtores, um deles sem parâmetros e o outro recebendo duas strings referentes aos atributos fieldName e message.
- Os getters e setters implementados de maneira comum.

```
import java.io.Serializable;

public class FieldMessage implements Serializable {
    private static final long serialVersionUID = 1L;

    private String fieldName;
    private String message;
}
```

resources/exceptions/ValidationError.java

- Subclasse de StandardError.
- Atributo lista do tipo FieldMessage.
- Método addError insere um objeto do tipo FieldMessage na lista de erros.
- Método get segue o padrão.

```
public class ValidationError extends StandardError {
    private static final long serialVersionUID = 1L;

    private List<FieldMessage> errors = new ArrayList<>();

    public ValidationError(Integer id, String msg, Long timeStamp) { super(id, msg, timeStamp); }

    public List<FieldMessage> getErrors() { return errors; }

    public void addError(String fieldName, String error) {
        this.errors.add(new FieldMessage(fieldName, error));
    }
}
```