

Curso Técnico Superior Profissional em: Tecnologias e Programação de Sistemas de Informação

2º Ano/1º Semestre

Unidade Curricular: Aplicações Centradas em Redes

Docente: Michael Silva / Hugo Perdigão

INTRODUÇÃO ÀS LINGUAGENS PARA O DESENVOLVIMENTO DE APLICAÇÕES CENTRADAS EM REDES

Laravel – Exercício (parte IV)

Objetivo:

- Criar mensagens de erro customizadas
- Utilizar SASS
- Criar relações em tabelas

- 1) Criar um Request para o Formulário e passar as regras de validação para o novo Request
 - a. php artisan make:request **NewCarRequest**
 - b. Alterar o tipo de dados recebido no método store para o novo Request e eliminar as regras de validação.
 - c. Colocar as regras de validação na class **NewCarRequest**
- 2) Customizar as regras de validação, na class **NewCarRequest**.

NECESSITA NODE.JS

- 3) Utilizar um ficheiro SASS (.scss) para o estilo
 - a. executar no terminal, npm install
 - b. especificar os recursos SASS no ficheiro webmix.js
 - c. criar a pasta **sass** dentro da pasta resources.
 - d. Executar o comando npm run dev para compilar os recursos.
- 4) Para criar tabelas com relações, é necessário ter pelo menos duas tabelas.

Cofinanciado por:

- a. Criar a tabela **marca**
 - b. Criar a migration correspondente
php artisan make:migration **create_marcas_table**
 - c. Criar as colunas necessárias na migration
 - d. Alterar a tabela dos **carros** para incluir uma nova coluna e criar a chave estrangeira.
 - i. Criar uma migration para adicionar a nova coluna e chave estrangeira à tabela de
php artisan make:migration add_fk_marca_carro
 - ii. Na nova migration, ir buscar a estrutura da tabela e adicionar a coluna e chave estrangeira: **maca_id**
NOTA: a coluna deve ser do tipo bigInteger e unsigned para corresponder ao tipo criado no id
 - iii. Executar a migração
- 5) Criar o modelo para a nova tabela.
php artisan make:model Marca
- 6) No formulário de criação de carros, acrescentar uma caixa de seleção para escolher a marca.
- 7) Criar algumas marcas manualmente na base de dados
 - a. Atribuir uma marca aos carros existentes
- 8) Especificar essa relação no modelo Marca (marca **hasMany** carros)
 - a. No model Marca, criar o método carros e definir a relação
- 9) Criar uma nova Route para aceder listar os carros por marca
 - a. No controller criar o método correspondente
 - b. Alterar a view para interpretar os novos dados

Bom Trabalho!

Cofinanciado por:

