

## **ENSINO MÉDIO INTEGRADO - INFORMÁTICA**

### **Disciplina de Desenvolvimento Web**

Trabalho Avaliativo: CRUD / Gráficos / Relatório PDF / Autenticação

---

*Gil Eduardo de Andrade*

*Referência - Projeto Laravel Base*

<https://github.com/SisNAPNE/aulas-webii/tree/master/start-project>

*Seeder - Povoar as Tabelas com Dados Iniciais*

<https://www.gileduardo.com.br/ifpr/dw/downloads/seeders.zip>

#### **Introdução e Contextualização**

O trabalho avaliativo tem como objetivo o desenvolvimento de uma aplicação web que contemple todos os conteúdos lecionados até o momento na disciplina de Desenvolvimento Web. A aplicação em questão será um sistema de registro de veículos (carros), contendo as seguintes informações: placal, modelo/marca, estado, cor.

#### **Metodologia de Correção e Avaliação**

Após o aluno apresentar o sistema funcionando e o professor verificar todos os itens (recursos a serem desenvolvidos) elencados na continuação deste documento o conceito será definido seguindo o critério: **A** => desempenho maior ou igual a **90**; **B** => desempenho maior ou igual a **75**; **C** => desempenho maior ou igual a **60**. O professor pode, se considerar necessário, efetuar questionamentos com relação a codificação apresentada, caso o aluno não saiba responder a pergunta, o professor pode descontar pontos do seu desempenho final.

#### **Recursos que devem ser, obrigatoriamente, utilizados:**

1. Definição e utilização de Rotas;
2. Criação e utilização de classes de Controle;





3. Criação e utilização de arquivos de visualização com recursos Blade (@extends, @foreach, ect);
4. Criação e utilização de classes de modelo;
5. Criação das tabelas do banco de dados via arquivos de migração;
6. Utilização do recurso “*validate*”, validação dos formulários da aplicação;
7. Utilização do pacote “*Breeze*” para realização da autenticação;
8. Utilização do pacote “*Dompdf*” para geração de relatórios PDF;
9. Utilização da biblioteca “*Google Chart*” para geração dos gráficos estatísticos.

### Funcionalidades e Procedimentos Gerais:

*Todos os CRUDs especificados a seguir só podem ser acessados após autenticação no sistema. Caso o usuário tente acessar um dos CRUDs, antes do processo de autenticação, a aplicação deve ser redirecionada para tela de login.*

1. CRUD de cores (campos): **(5 pontos)**
  - **nome:** requerido, tamanho máximo de 30 caracteres, tamanho mínimo de 2 caracteres;



NOME	ACTIONS
AMARELO	  
AZUL	  

2. CRUD de marcas (campos): **(5 pontos)**

- **nome:** requerido, tamanho máximo de 30 caracteres, tamanho mínimo de 2 caracteres;









NOME	ACTIONS
CHEVROLET	  
FIAT	  

### 3. CRUD de modelos (campos):

**(10 pontos)**

- **nome:** requerido, tamanho máximo de 30 caracteres, tamanho mínimo de 2 caracteres;
- **marca\_id:** requerido;



NOME	MARCA	ACTIONS
208	PEUGEOT	  
ARGO	FIAT	  

### 4. CRUD de carros (campos):

**(15 pontos)**

- **placa:** requerido, tamanho máximo de 8 caracteres, tamanho mínimo de 8 caracteres;
- **modelo\_id:** requerido;
- **estado\_id:** requerido;
- **cor\_id:** requerido;

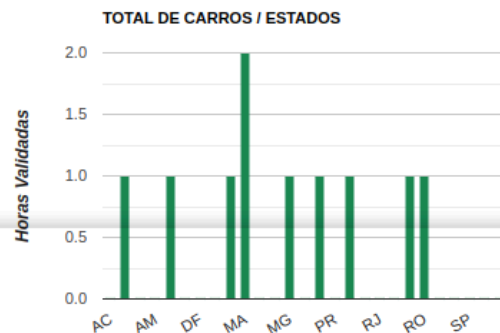
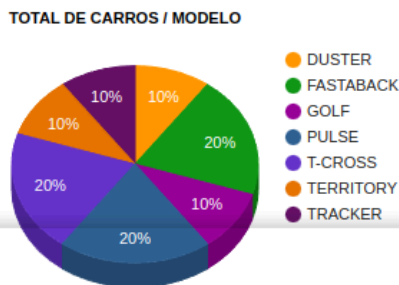
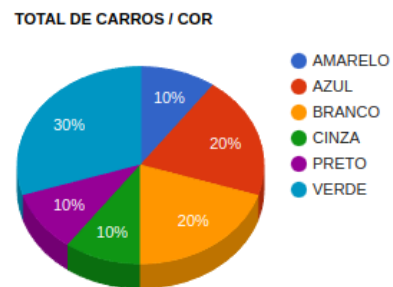
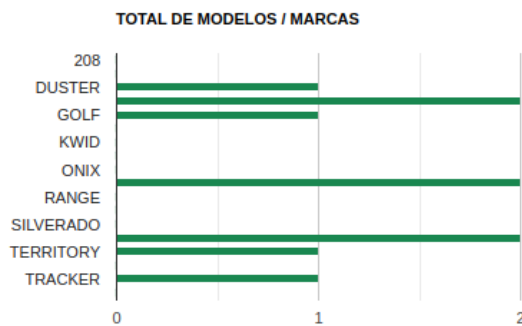


PLACA	MODELO	COR	ESTADO	ACTIONS
GXN-8166	T-CROSS	VERDE	RO	
IAD-4132	FASTABACK	BRANCO	MA	

5. Tela Inicial - Apresentação dos Gráficos:

**(25 pontos)**

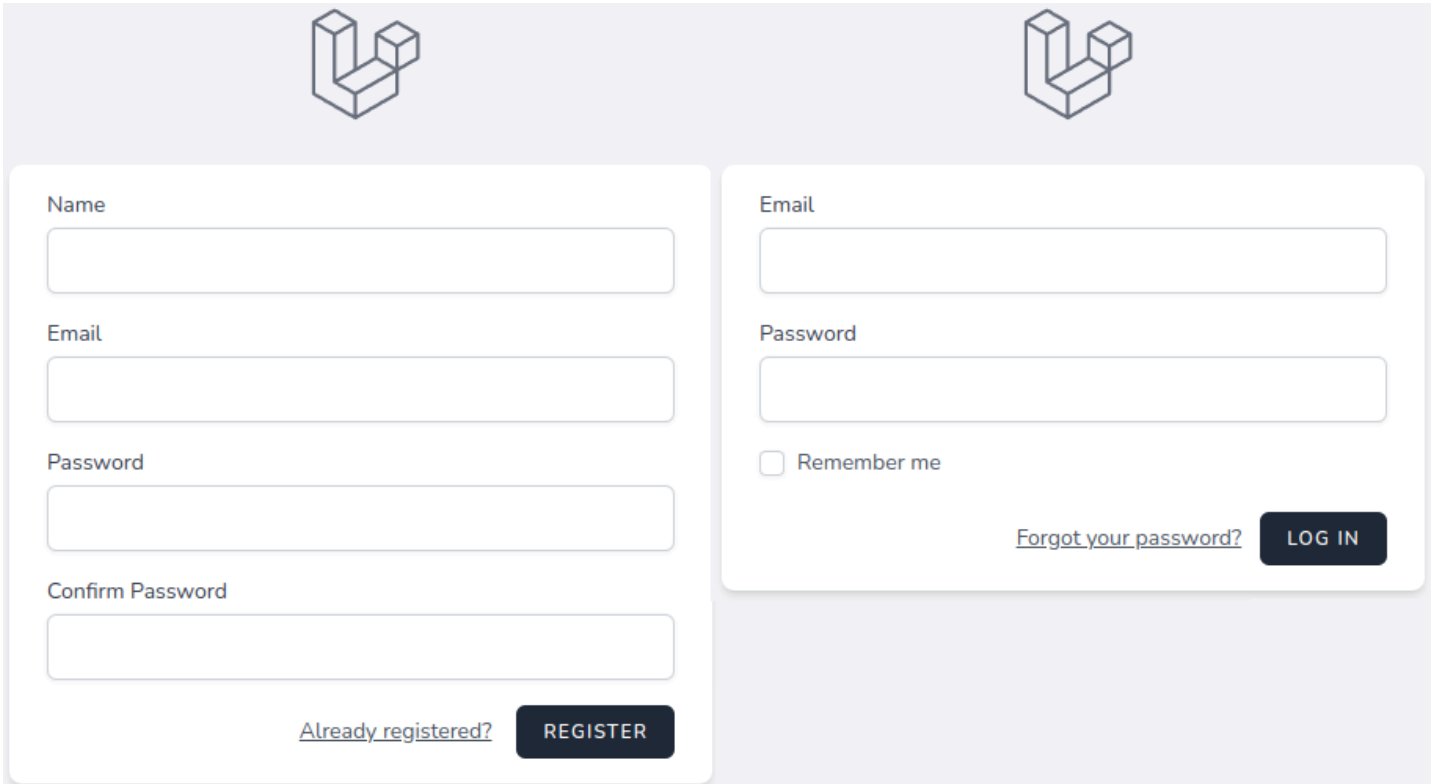
- Apresentar adequadamente o (layout) e os quatro gráficos para as seguintes informações: (a) Total de Modelos por Marcas / gráfico de barras; (b) Total de Carros por Cor / gráfico de pizza; (c) Total de Carros por Modelo / gráfico de pizza; (d) Total de Carros por Estado / gráfico de colunas;



6. Tela Inicial - Autenticação e Cadastro do Usuário:

**(15 pontos)**

- Apresentar adequadamente (layout) e disponibilizar os menus para que o usuário possa se cadastrar e efetuar autenticação no sistema;



7. Relatório PDF - Carros Registrados no Sistema

**(15 pontos)**

- Apresentar adequadamente (layout / como imagem a seguir) das informações especificadas (placa, estado, modelo e cor) para cada um dos veículos cadastrados no sistema;

---

## RELATÓRIO DE VEÍCULOS

---

### GXN-8166

- RONDÔNIA (RO)
- T-CROSS
- VERDE

---

### IAD-4132

- MARANHÃO (MA)
  - FASTABACK
  - BRANCO
- 

#### 8. Funcionamento Geral

**(10 pontos)**

- A aplicação está executando de maneira plena, sem apresentar qualquer erro de navegação, layout, etc.;

#### Dicas e Procedimentos:

- 1) Aplicação base: <https://github.com/SisNAPNE/aulas-webii/tree/master/start-project>
- 2) Configurar aplicação:
  - `composer install` (no terminal, dentro da pasta do projeto)
  - criar um cópia do arquivo `.env.example` e renomeá-la para `.env` (manual)
  - `php artisan key:generate` (no terminal, dentro da pasta do projeto)
- 3) Executar a aplicação: `php artisan serve`
- 4) Criar Controller: `php artisan make:controller NomeController -r`
  - Arquivo criado no diretório: `app/Http/Controllers`
- 5) Definição das rotas: `arquivo -> route/web.php`

6) Visualizar todas as rotas: *php artisan route:list*

7) Criar Model/Migration: *php artisan make:model Nome -m*

- Arquivo criado no diretório: *app/Models* (Modelo)
- Arquivo criado no diretório: *database/migrations* (Migração)

8) Configurar Banco de Dados: *arquivo -> .env*

- Linhas:

```
DB_DATABASE=sua_base_dados  
DB_USERNAME=seu_usuario  
DB_PASSWORD=sua_senha
```

9) Efetuar Migração: *php artisan migrate:fresh*

10) Criar View: *diretório -> resources/views*