

FORM, MODEL DAN PENYIMPANAN DATA

Pada materi sebelumnya, kita telah mempelajari dan memahami Blade serta prinsip template dalam Laravel. Blade adalah engine template bawaan Laravel yang mempermudah pengembangan aplikasi dengan cara menyediakan sintaks yang lebih bersih dan rapi dibandingkan dengan PHP murni. Konsep template ini memungkinkan pengembang untuk membangun komponen antarmuka yang modular dan dapat digunakan kembali, sehingga meningkatkan efisiensi dan menjaga konsistensi tampilan aplikasi. Pada materi kali ini, kita akan melanjutkan dengan membuat form input barang dan menyimpannya ke dalam *database*. Form ini berfungsi sebagai antarmuka bagi pengguna untuk memasukkan data barang baru. Dalam proses ini, kita akan menggunakan Tailwind CSS, sebuah library framework CSS yang diintegrasikan secara default dalam Laravel 11. Tailwind CSS menawarkan fleksibilitas tinggi dengan pendekatan utility-first, memungkinkan kita untuk dengan cepat membuat desain yang responsif dan modern tanpa perlu menulis banyak kode CSS secara manual.

Langkah pertama adalah membuat form menggunakan sintaks Blade, dengan mengatur elemen-elemen seperti input teks, dropdown, dan tombol submit. Kemudian, kita akan mengatur rute (routes) di Laravel untuk menangani permintaan (request) form yang dikirim oleh pengguna. Data yang dikirimkan akan divalidasi terlebih dahulu untuk memastikan keabsahan input. Setelah itu, data valid akan disimpan ke dalam database menggunakan fitur Eloquent ORM yang mendukung arsitektur Laravel. Dengan kombinasi Blade, Tailwind CSS, dan Eloquent ORM, proses pembuatan form barang ini akan lebih cepat, efisien, dan memberikan pengalaman pengguna yang baik. Melalui materi ini, kita diharapkan dapat memahami alur lengkap dari pembuatan form, penanganan data, hingga penyimpanan ke dalam database, serta meningkatkan keterampilan dalam menggunakan fitur-fitur Laravel yang modern dan relevan.

Percobaan keempat membuat form barang.

Kita perlu membuat menu Master Product terlebih dahulu pada template yang sudah dibuat sebelumnya, jika perlu memodifikasi file navigasi yang berada di `resources\views\layouts\navigation.blade.php` dengan kode seperti dibawah ini

```
<!-- Data master Dropdown -->
<div class="hidden sm:flex sm:items-center sm:ms-6">
```

```

        <x-dropdown align="right" width="48">
            <x-slot name="trigger">
                <button class="inline-flex items-center px-3
py-2 text-sm font-medium leading-4 text-gray-500 transition duration-150
ease-in-out bg-white border border-transparent rounded-md
hover:text-gray-700 focus:outline-none">
                    <div>Data Master</div>

                    <div class="ms-1">
                        <svg class="w-4 h-4 fill-current"
xmlns="http://www.w3.org/2000/svg" viewBox="0 0 20 20">
                            <path fill-rule="evenodd" d="M5.293
7.293a1 1 0 011.414 0L10 10.586l3.293-3.293a1 1 0 111.414 1.414l-4 4a1 1
0
                                01-1.414 0l-4-4a1 1 0 010-1.414z"
clip-rule="evenodd" />
                        </svg>
                    </div>
                </button>
            </x-slot>

            <x-slot name="content">
                <x-dropdown-link
:href="route('product-create') ">
                    Product Master
                </x-dropdown-link>

            </x-slot>
        </x-dropdown>
    </div>

```

Setelah membuat menu Product master, selanjutnya kita perlu membuat form barang dengan nama file `create-product.blade.php` yang berada di direktori `resources\views\master-data\product-master\create-product.blade.php` jika di project anda tidak ditemukan folder-folder seperti diatas, silahkan membuat folder baru dengan nama tersebut, setelah berhasil membuat file `create-product.blade.php` selanjutnya kita mengisi file tersebut dengan kode dibawah ini

```

<x-app-layout>
  <x-slot name="header">
    <h2 class="text-xl font-semibold leading-tight text-gray-800">
      {{ __( 'Dashboard' ) }}
    </h2>
  </x-slot>

  <div class="py-12">

    <div class="mx-auto max-w-7xl sm:px-6 lg:px-8">
      <div class="overflow-hidden bg-white shadow-sm sm:rounded-lg">
        <div class="p-6 text-gray-900">
          <div class="container mx-auto mt-5">
            <h2 class="mb-5 text-2xl font-bold">Create New
Product</h2>

            <x-auth-session-status class="mb-4"
:status="session('success') " />

            <form action="{{ route('product-store') }}"
method="POST" class="space-y-4">
              @csrf <!-- Laravel CSRF protection -->

              <div class="form-group">
                <label for="product_name" class="block text-sm font-medium text-gray-700">Product Name</label>
                <input type="text" id="product_name"
name="product_name" class="block w-full p-2 mt-1 border border-gray-300 rounded-md shadow-sm focus:outline-none focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm" required>
              </div>

              <div class="form-group">
                <label for="unit" class="block text-sm font-medium text-gray-700">Unit</label>
                <select id="unit" name="unit"
class="block w-full p-2 mt-1 border border-gray-300 rounded-md shadow-sm focus:outline-none focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm" required>

```

```

                                <option value="" disabled
selected>Select a unit</option>
                                <option value="kg">Kilogram
(kg)</option>
                                <option value="ltr">Liter
(ltr)</option>
                                <option value="pcs">Pieces
(pcs)</option>
                                <option value="box">Box</option>
                                </select>
                            </div>

                            <div class="form-group">
                                <label for="type" class="block text-sm
font-medium text-gray-700">Type</label>
                                <input type="text" id="type" name="type"
class="block w-full p-2 mt-1 border border-gray-300 rounded-md shadow-sm
focus:outline-none focus:ring-indigo-500 focus:border-indigo-500
sm:text-sm" required>
                            </div>

                            <div class="form-group">
                                <label for="information" class="block
text-sm font-medium text-gray-700">Information</label>
                                <textarea id="information"
name="information" rows="3" class="block w-full p-2 mt-1 border
border-gray-300 rounded-md shadow-sm focus:outline-none
focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm"></textarea>
                            </div>

                            <div class="form-group">
                                <label for="qty" class="block text-sm
font-medium text-gray-700">Quantity</label>
                                <input type="number" id="qty" name="qty"
class="block w-full p-2 mt-1 border border-gray-300 rounded-md shadow-sm
focus:outline-none focus:ring-indigo-500 focus:border-indigo-500
sm:text-sm" required>
                            </div>

```

```
        <div class="form-group">
            <label for="producer" class="block
text-sm font-medium text-gray-700">Producer</label>
            <input type="text" id="producer"
name="producer" class="block w-full p-2 mt-1 border border-gray-300
rounded-md shadow-sm focus:outline-none focus:ring-indigo-500
focus:border-indigo-500 sm:text-sm" required>
        </div>

        <button type="submit" class="inline-flex
justify-center px-4 py-2 text-sm font-medium text-white bg-indigo-600
border border-transparent rounded-md shadow-sm hover:bg-indigo-700
focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500">Submit</button>
    </form>
</div>

    @vite('resources/js/app.js') <!-- Include Vite's JS
assets -->
    </div>
</div>
</div>
</div>
</x-app-layout>
```

Setelah membuat form Master Product selanjutnya kita perlu memberikan nama pada routes product yang sebelumnya kita buat menjadi seperti berikut

```
9 Route::get('/product/create', [ProductController::class, 'create'])->name("product-create");
10 Route::post('/product', [ProductController::class, 'store'])->name("product-store");
```

Jika sudah sesuai, maka silahkan akses form Master Product yang telah dibuat.

Membuat Model Laravel

Untuk membuat model di Laravel, Anda dapat menggunakan perintah artisan yang disediakan oleh Laravel. Model adalah komponen penting dalam arsitektur MVC

Laravel dan digunakan untuk berinteraksi dengan tabel di database. Di Laravel, model berada di direktori `app/Models`. Berikut ini adalah langkah-langkah rinci untuk membuat model di Laravel, Jalankan perintah berikut untuk membuat model

```
php artisan make:model Product
```

Penjelasan

- Product Nama model yang akan Anda buat. Gunakan bentuk tunggal dan kapital pada huruf pertama sesuai konvensi Laravel.
- Laravel akan membuat file model `Product.php` di direktori `app/Models`.
- Jika direktori `app/Models` belum ada, Laravel akan membuatnya secara otomatis.

Mendefinisikan Atribut Model

Setelah model berhasil dibuat, buka file model di `app/Models/Product.php` dan tambahkan properti `$fillable` untuk menentukan field yang dapat diisi secara massal, silahkan modifikasi seperti berikut

```
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Product extends Model
9  {
10     use HasFactory;
11
12     protected $fillable = [
13         'product_name',
14         'unit',
15         'type',
16         'information',
17         'qty',
18         'producer',
19     ];
20 }
21
```

Percobaan kelima memodifikasi file ProductController.php

Anda perlu memodifikasi File `ProductController.php` pada bagian method create menjadi seperti berikut

```
21     public function create()
22     {
23         return view("master-data.product-master.create-product");
24     }
```

Penjelasan kode pada baris 23 adalah ketika method `create()` dipanggil pada Routes maka akan memanggil view yang berada di folder `master-data/product-master/create-product.blade.php`

Kemudian Anda perlu memodifikasi method `store()` untuk memproses penyimpanan data sebagai berikut

```
29 public function store(Request $request)
30 {
31     // validasi input data
32     $validasi_data = $request->validate([
33         'product_name' => 'required|string|max:255',
34         'unit' => 'required|string|max:50',
35         'type' => 'required|string|max:50',
36         'information' => 'nullable|string',
37         'qty' => 'required|integer',
38         'producer' => 'required|string|max:255',
39     ]);
40
41     // Proses simpan data kedalam database
42     Product::create($validasi_data);
43
44     return redirect()->back()->with('success', 'Product created successfully!');
45 }
```

Karena pada baris 42 diatas kita memanggil model Product, maka kita perlu mengimport model Product pada `ProductController.php` seperti berikut

```
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Product;
```

Selanjutnya silahkan anda coba menambahkan data barang, jika berhasil akan menambah data pada tabel Product

 Dashboard
 Data Master ▾
Test User ▾

Dashboard

Create New Product

Product Name

Unit
 Select a unit ▾

Type

Information

Quantity

Producer

Submit

Untuk mencoba apakah form sudah dapat menyimpan data kedalam *database* silahkan isi form dan kemudian submit.

✓ Menampilkan baris 0 - 6 (total 7, Pencarian dilakukan dalam 0,0008 detik.)

SELECT * FROM `products`

☐ Profil [\[Edit dikotak \]](#) [\[Ubah \]](#) [\[Jelaskan SQL \]](#) [\[Buat kode PHP \]](#) [\[Segarkan \]](#)

☐ Tampilkan semua | Jumlah baris: 25 ▾ | Saring baris: Cari di tabel ini | Sort by key: Tidak ada ▾

Extra options

					id	product_name	unit	type	information	qty	created_at	updated_at	producer
☐					2	tes	ltr	botol	tes	3	2024-10-04 07:46:21	2024-10-04 07:46:21	indofood
☐					3	test	kg	tes	tes	2	2024-10-04 07:47:42	2024-10-04 07:47:42	indofood
☐					4	test	pcs	botol	tes	2	2024-10-04 07:48:31	2024-10-04 07:48:31	indofood
☐					5	test	ltr	tes	tes	2	2024-10-04 07:49:38	2024-10-04 07:49:38	a
☐					6	test	kg	a	a	1	2024-10-04 07:50:08	2024-10-04 07:50:08	a
☐					7	test	pcs	a	a	1	2024-10-04 07:50:41	2024-10-04 07:50:41	2
☐					8	test	ltr	tes	a	1	2024-10-04 07:51:09	2024-10-04 07:51:09	a

☐ Pilih Semua
 Dengan pilihan:
  Ubah
  Salin
  Hapus
  Ekspor

