

Statistik dan Grafik

Sebelum mengimplementasikan grafik pada halaman dashboard, buka `resources/views/layouts/app.blade.php` kemudian tambahkan kode `@yield('scripts')` di bagian paling bawah tag script.

Nama file	resources/views/layouts/app.blade.php
Deskripsi	Layout scaffold
<pre>... <script> \$('[data-toggle="tooltip"]').tooltip(); </script> @yield('scripts') ...</pre>	

Penambahan kode ini memungkinkan kita untuk menambahkan custom javascript pada template. Terutama untuk menggunakan chartjs. Sedangkan chartjs tidak perlu dimuat di semua halaman.

Buka model Kategori dan tambahkan accessor untuk warna. Warna ini akan digunakan untuk menampilkan warna di dalam grafik donat jumlah buku dalam suatu kategori. Warna di generate berdasarkan dari nama kategori. Jika setiap nama kategori unik maka juga akan memiliki hash md5 yang unik. Dari md5 inilah dipotong 6 karakter pertama sebagai hash warna. Cara ini lebih bagus daripada mendefinisikan warna untuk setiap kategori ataupun mengacak warna.

Nama file	app/Models/Kategori.php
Deskripsi	Model kategori
<pre>... public function getColorAttribute() { return "#".substr(md5(\$this->nama), 0, 6); } ...</pre>	

Buka DashboardController, disini terdapat perubahan besar untuk menampilkan data dalam bentuk grafis. Salin kode berikut ke dalam DashboardController.

Nama file	app/Http/Controllers/DashboardController.php
Deskripsi	Controller untuk dasbor
<pre><?php namespace App\Http\Controllers; use App\Models\Kategori; use App\Models\Pembelian; use App\Models\Penjualan; use Carbon\Carbon; use Illuminate\Database\Eloquent\Builder; use Illuminate\Http\Request; use Illuminate\Support\Collection; use Illuminate\Support\Facades\DB;</pre>	

```

class DashboardController extends Controller{

    public function index(Request $request){
        return view('dasbor', [
            'penjualan' => $this->sumStat(Penjualan::query()),
            'pembelian' => $this->sumStat(Pembelian::query()),
            'kategori' => Kategori::query()
                ->withCount('buku')
                ->get()
                ->sortByDesc('buku_count')
        ]);
    }

    private function sumStat(Builder $query){
        $data = $query
            ->select(
                DB::raw('sum(stok) as sum'),
                DB::raw("DATE_FORMAT(created_at,'%c') as monthKey")
            )
            ->groupBy('monthKey')
            ->orderByDesc('created_at')
            ->whereDate('created_at', '>=', Carbon::now()->subMonths(11)-
>firstOfMonth())
            ->get();

        $stats = new Collection();
        $now = Carbon::now()->setDay(1);
        for($i=0; $i<12; $i++){
            $stats->push((object) [
                'key' => $now->month,
                'label' => $now->monthName,
                'data' => 0
            ]);

            $now->subMonth();
        }

        foreach($data as $item){
            $stats
                ->firstWhere('key', '=', $item->monthKey)
                ->data = intval($item->sum);
        }

        return $stats->reverse();
    }
}

```

Buka file view resources/views/dasbor.blade.php dan edit konten file tersebut menjadi kode seperti di bawah ini.

Nama file	resources/views/dasbor.blade.php
Deskripsi	View untuk dasbor
<pre> @extends('layouts.app') @section('title', 'Dashboard') @section('content') <div class="row"> <div class="col-md-8"> </pre>	

```

        <div class="card shadow mb-4">
            <div class="card-header py-3">
                <h6 class="m-0 font-weight-bold text-primary">Penjualan dan
Pembelian 12 Bulan Terakhir</h6>
            </div>
            <div class="card-body">
                <div class="chart-area">
                    <canvas id="stok-chart"></canvas>
                </div>
            </div>
        </div>
</div>
<div class="col-md-4">
    <div class="card shadow mb-4">
        <div class="card-header py-3">
            <h6 class="m-0 font-weight-bold text-primary">Kategori
Buku</h6>
        </div>
        <div class="card-body">
            <div class="chart-pie">
                <canvas id="kategori-chart"></canvas>
            </div>
        </div>

        <ul class="list-group list-group-flush">
            @foreach($kategori as $item)
                <li class="list-group-item py-1">
                    <span>{{ $item->nama }}</span>
                    <span class="badge badge-primary float-right">{{ $item-
>buku_count }}</span>
                </li>
            @endforeach
        </ul>
    </div>
</div>
</div>
@endsection

@section('scripts')
    <script src="{{ asset('vendor/chart.js/Chart.min.js') }}"></script>
    <script>
        $(document).ready(function(){
            window.stokChart = new Chart(document.getElementById('stok-
chart').getContext('2d'), {
                type: 'line',
                data: {
                    labels: {!! $penjualan->pluck('label')->toJson() !!},
                    datasets: [
                        {
                            label: 'Penjualan',
                            backgroundColor: 'rgb(75, 192, 192, 0.2)',
                            borderColor: 'rgb(75, 192, 192)',
                            data: {!! $penjualan->pluck('data')->toJson() !!},
                            pointHitRadius: 10,
                        },
                        {
                            label: 'Pembelian',
                            backgroundColor: 'rgb(255, 99, 132, 0.2)',
                            borderColor: 'rgb(255, 99, 132)',
                            data: {!! $pembelian->pluck('data')->toJson() !!},
                            pointHitRadius: 10,
                        }
                    ]
                }
            });
        });
    </script>

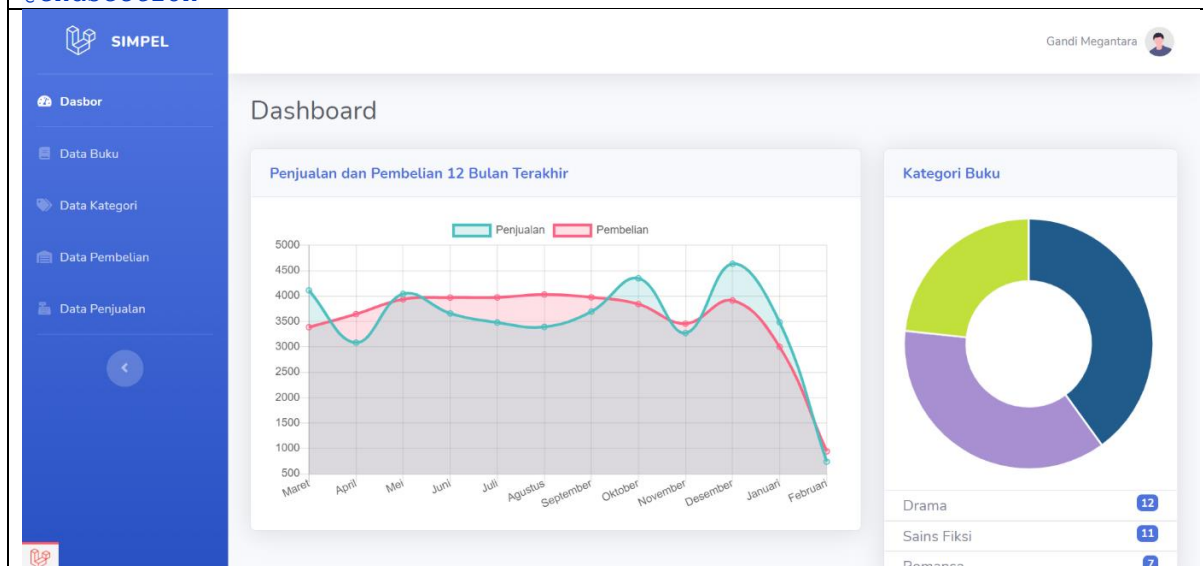
```

```

    },
  ],
},
options: {
  maintainAspectRatio: false,
}
});

window.kategoriChart = new Chart(document.getElementById('kategori-
chart').getContext('2d'), {
  type: 'doughnut',
  data: {
    labels: {!! $kategori->pluck('nama')->toJson() !!},
    datasets: [{
      data: {!! $kategori->pluck('buku_count')->toJson() !!},
      backgroundColor: {!! $kategori->pluck('color')->toJson() !!},
    ]},
  },
  options: {
    maintainAspectRatio: false,
    tooltips: {
      displayColors: false,
    },
    legend: {
      display: false
    },
    cutoutPercentage: 50,
  },
});
</script>
@endsection

```



Pada view di atas kita memuat chart js yang disediakan oleh SB Admin 2. Kemudian membuat canvas dan mendefinisikan konfigurasi dari grafik. Data yang diperlukan di sisipkan di dalam kode js menggunakan method toJson(). Banyak sekali macam-macam grafik yang bisa kamu terapkan. Baca dokumentasi lengkapnya di <https://www.chartjs.org/>.