MODUL 9 VIEW MANGGUNAKAN TEMPLATE BLADE MENGGUNAKAN FORM



CAPAIAN PEMBELAJARAN

- 1. Mampu membuat view menggunakan Blade template
- 2. Mengimplementasikan komponen Form



KEBUTUHAN ALAT/BAHAN/SOFTWARE

- 1. Sistem Operasi Linux/Window 10
- 2. PHP 5.x, diutamakan versi 7.x keatas
- 3. Composer
- 4. XAMPP (PHP, Apache, MySQL)
- 5. Text Editor Notepad++ atau Visual Stodio Code



DASAR TEORI

1. Kelas Form

Sebelum menggunakan kelas Form, harus menambahkan dengan cara menginstal paket ini melalui composer. Edit file composer.json, sehingga akan menambahkan paket laravelcollective/html.

2. Maka anda harus menambahkan kelas Form dan Html, seperti pada perintah berikut:

composer require "laravelcollective/html":"^5.8.0"

di dalam file composer.json, bertambah, seperti berikut:

```
"require": {
    "php": "^7.1.3",
    "fideloper/proxy": "^4.0",
    "laravel/framework": "5.8.*",
    "laravel/tinker": "^1.0",
    taravelcollective/html": "^5.8"
},
```

3. Untuk mengaktifkan ubahlah file config/app.php, seperti pada kode program berikut:

```
<?php
 return [
providers' => [
  // ...
   Collective\Html\HtmlServiceProvider::class,
   // ...
  ],
'aliases' => [
  // ...
   'Form' => Collective\Html\FormFacade::class,
   'Html' => Collective\Html\HtmlFacade::class,
  'Input' => Illuminate\Support\Facades\Input::class,
  // ...
 ],
 ];
?>
```

- 4. Baberapa method Foram antara lain:
- Method open() dan close()

Untuk memulai penggunaan kelas Form diawali dengan method <code>open()</code>, <code>danclose()</code> untuk menutup semua penggunaan kelas Form, penulisan method <code>open()</code> seperti pada skrip berikut:

```
{!! Form::open(['route' => 'buku.store','method'=>'POST']) !!}
//.........
{!! Form::close() !!}
```

Parameter 'route'=>'buku.store' pada saat proses submit, proses dilanjutkan ke method store, dengan menggunakan method POST untuk mengirim semua masukkan ke Server.

Menggunakan action di Controller, contoh:

```
{!! Form::open(['route' => 'route.name']) !!}
{!! Form::open(['action' => 'Controller@method'])!!}

Menggunakan parameter seperti contoh berikut:
{!! Form::open(['route' =>['route.name', $user->id]]) !!}
{!! Form::open(['action' =>['Controller@method',$user->id]]) !!}

Jika Form yang akan digunakan untuk meunggah file, tambahkan opsi file ke array seperti berikut:
{!! Form::open(['url' => 'buku/store', 'files' => true]) !!}

atau
{!! Form::open(['route' => 'buku.store', 'method'=>'POST',
```

• Menambahkan Token CSRF ke Form

Laravel menyediakan method untuk melindungi aplikasi dari pemalsuan permintaan lintas situs. Pertama, token acak ditempatkan di sesi *user*. Jika menggunakan method Form::open dengan POST, PUT, atau DELETE, token CSRF akan ditambahkan ke formulir sebagai *field* tersembunyi secara otomatis. Atau, jika ingin menghasilkan HTML untuk *field* CSRF tersembunyi, bisa menggunakan method token(), contoh:

'enctype'=>'multipart/form-data']]) !!}

```
{!! Form::token() !}}
```

Menyertakan Filter CSRF ke Sebuah Rute

Cara menggunakan filter CSRF ke dalam route, seperti pada contoh berikut:

```
Route::post('profile', array('before' => 'csrf', function()
{
    //
}));
```

Tampilan Label

Method label (...) digunakan untuk penulisan Label, cara penulisan method label seperti pada skrip berikut:

```
{!! Form::label('id_penerbit', 'Penerbit:',
    ['class' => 'control-label']) !!}
```

• Masukkan file

Method file(), digunakan untuk mengupload file, cara penulisan method file seperti pada skrip berikut:

```
echo Form::file($name, $attributes = array());
echo Form::file('file_foto');
$name nama field, $attributes array seting tampilan html
```

Tombol Checkbox

Method checkbox(), untuk isian dalam bentuk pilihan dengan cara memberi tanda centang, penulisannya seperti pada skrip berikut:

```
{!! Form::checkbox('hobby1','1',true)!!} Music
{!! Form::checkbox('hobby2','1',false)!!} Olah Raga
```

• Pilihan Tombol Radio

Untuk pilihan tombol Radio, menggunakan method radio (...), penulisan method radio seperti pada skrip berikut:

```
{!! Form::radio('jenis_kel','L',true) !!} Laki-laki
{!! Form::radio('jenis_kel','P',false) !!} Perempuan
```

Jenis_kel adalah nama field, **P**:ketika Tombol **Perempuan** dipilih, false kondisi tidak dipilih

• Pilihan Drop-Down List

Pilihan derop-down-list menggunakan method select(), penulisan method select() seperti pada, skrip berikut:

```
echo Form::select('jenis_kel', array('L' => 'Laki-laki',
```

```
'P' => 'Perempuan'))
```

Contoh penggunaan method select() dengan nilai default

```
echo Form::select('size', array('L' => 'Large', 'S' =>
'Small'), 'S');
```

Masukan date

Method date(), digunakan untuk masukkan tanggal, tampilan ketika di klik akan menampilkan kalender, untuk menghindari kesalahan, user tinggal memilih tanggal. Cara penulisannya seperti pada skrip berikut:

```
{!! Form::date('tanggal', \Carbon\Carbon::now()) !!}
```

• Tombol Submit

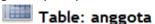
Method button() digunakan untuk menyajikan tombol, cara penulisan method button seperti pada skrip berikut:

```
{!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
```



PRAKTIK

1. Buatlah tabel anggota seperti pada struktur tabel berikut:



Column Information

	Field	Туре	Comment
7	id	varchar(5)	
	nama	varchar(100)	
	jenis_kel	char(1)	
	telepon	varchar(20)	
	e_mail	varchar(20)	
	foto	varchar(20)	
	user_id	int(11)	
	created_at	timestamp	
	updated_at	timestamp	

2. Buatlah controller menggunkan perntah artisan seperti berikut:

```
php artisan make:controller AnggotaController -resource
php artisan make:model models/Anggota
```

3. Tambahkan model Anggota.php seperti kode berikut:

4. Tambahkan AnggotaController.php seperti berikut:

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\View;
use Illuminate\Support\Facades\Validator;
use Illuminate\Support\Facades\Input;
use Illuminate\Support\Facades\Redirect;
use App\models\Anggota;
class AnggotaController extends Controller
{
    /**
     * Display a listing of the resource.
     * @return \Illuminate\Http\Response
     */
    public function index()
        $anggota = Anggota::orderBy('id', 'DESC')
                ->paginate(5);
        return view('anggota.index', compact('anggota'));
    }
     * Show the form for creating a new resource.
```

```
* @return \Illuminate\Http\Response
public function create()
    return view('anggota.create');
}
* Store a newly created resource in storage.
* @param \Illuminate\Http\Request $request
 * @return \Illuminate\Http\Response
public function store(Request $request)
    //validasi
    $this->validate($request, [
        'id'
                    => 'required|max:5',
        'nama'=> 'required ',
        'jenis_kel' => 'required ',
        'e mail'
                  => 'required|email',
        'telepon'
                    => 'required',
        'foto'=>'required',
      ]);
    $foto = $request->file('foto');
    //membaca extensi file gambar
    $ext = $request->file('foto')->getClientOriginalExtension();
    //memberi file menggunakan no anggota
    $namaFile = Input::get('id').'.'.$ext;
    //menyimpan ke folder public/foto/...
    $request->file('foto')->move('foto', $namaFile);
    $anggota = new Anggota($request->all());
    $anggota->foto = $namaFile;
    $anggota->save();
    return redirect()->route('anggota.index')
            ->with('success',
            'Menambah Anggota telah berhasil disimpan');
}
```

5. View index.blade.php, seperti pada kode program berikut:

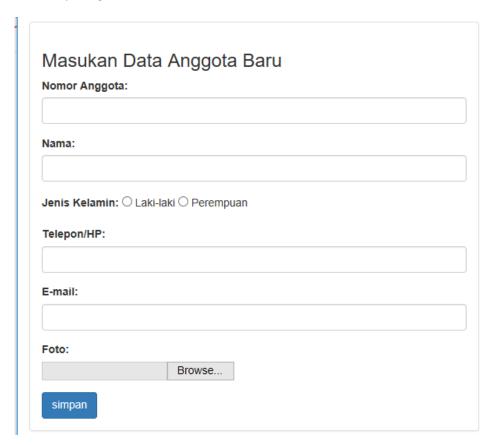
```
<div class="row">
   <div class="col-lg-12 margin-tb">
   <div class="pull-left">
   <h2 class="fa fa-check-square-o" style="font-size:28px;">
        Tabel Rekaman Anggota</h2>
   </div>
</div>
<a href="{{ url('anggota/create') }}"</pre>
   class="btn btn-primary btn-sm">
<span class="glyphicon glyphicon-plus"></span> Tambah Anggota</a>
No
         No Anggota
         Nama
         telepon
         e-mail
         Aksi
      </thead>
   @php $i=0; @endphp
   @foreach($anggota as $key => $value)
      {{ ++$i }}
         {{ $value->id }}
         {{ $value->nama }}
         {{ $value->telepon }}
         {{ $value->e_mail }}
         <a class="btn btn-info " href="{{</pre>
             route('anggota.show',$value->id) }}">Lihat</a>
         <a class="btn btn-primary" href="{{</pre>
             route('anggota.edit',$value->id) }}">Ubah</a>
         {!! Form::open(['method' => 'DELETE', 'route' =>
             ['anggota.destroy', $value->id],
              'style'=>'display:inline']) !!}
         {!! Form::submit('Hapus', ['class' =>
             'btn btn-danger confirm','data-confirm' =>
             'Are you sure you want to delete?'])!!}
         {!! Form::close() !!}
         @endforeach
   {!! $anggota->render() !!}
@endsection
```



View create.blade.php

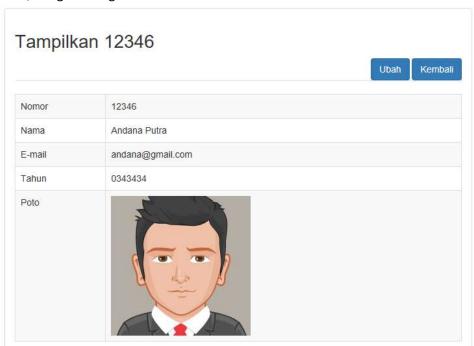
```
<!-- app/views/pegawai/create.blade.php -->
@extends('layouts.app')
@section('content')
<h3>Masukan Data Anggota Baru</h3>
{!! Form::open(['route' => 'anggota.store','method'=>'POST',
                'enctype'=>'multipart/form-data']) !!}
<div class="form-group">
    {!! Form::label('id', 'Nomor Anggota:',
          ['class' => 'control-label']) !!}
    {!! Form::text('id', null, ['class' => 'form-control']) !!}
</div>
<div class="form-group">
    {!! Form::label('nama', 'Nama:',['class' =>'control-label']) !!}
    {!! Form::text('nama', null, ['class' => 'form-control']) !!}
</div>
<div class="form-group">
    {!! Form::label('alamat', 'Alamat:',
        ['class' => 'control-label']) !!}
    {!! Form::text('alamat', null, ['class' => 'form-control']) !!}
</div>
<div class="form-group">
    {!! Form::label('jenis_kel', 'Jenis Kelamin:',
               ['class' => 'control-label']) !!}
    {!! Form::radio('jenis_kel','L',false) !!} Laki-laki
{!! Form::radio('jenis_kel','P',false) !!} Perempuan
</div>
<div class="form-group">
    {!! Form::label('tanggal_lhr', 'Tanggal Lahir',
           ['class' => 'control-label']) !!}
    {!! Form::date('tanggal_lhr',\Carbon\Carbon::now(),
           ['class' => 'form-control']) !!}
 </div>
<div class="form-group">
    {!! Form::label('telepon', 'Telepon/HP:']
                   ['class' => 'control-label']) !!}
    {!! Form::text('telepon', null, ['class' => 'form-control']) !!}
```

Hasil Form seperti gambar berikut:





1. Buatlah views show.blade.php untuk maanpilkan anggota seperti pada gambar berikut, dengan mengklik tombol Lihat.



2. Buatlah form mengubah rekaman, untuk oparasi tombol ubah tombol

```
public function edit($id)
{
    //........?
}
```

3. Buatlah method update(), untuk mengubah rekaman anggota.

Ubah



TUGAS

- 1. Ubah/tambahkan tabel anggota field alamat, tanggal_lahir,
- 2. Buat/ubah formulir create, dan edit
- 3. Ubah dan uji Aplikasi CRUD anggota.



REFERENSI

- 1. Forms & HTML, https://laravel.com/docs/5.8/blade#forms
- 2. Cheers, KodeBlog Team, Laravel 5 Views, https://kode-blog.io/laravel-5-views