



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

<b>Nama</b>	:	JIHAN KARUNIA PUTRI
<b>NIM</b>	:	2241720031
<b>Kelas</b>	:	TI - 2H

## JOBSHEET 6

### A. Template Form (Admin LTE)

1. Penggunaan template AdminLTE Bootstrap Admin Dashboard Template dengan langkah pertama akses <https://adminlte.io/> , lalu klik download pada source code (zip)



2. Lakukan extract All dan ubah nama folder AdminLTE-3.2.0 menjadi template, folder template pindahkan ke laragon/www/PWL\_2024/public. Buka folder public pada VSCode maka akan tampil folder baru bernama template



3. Copy isi public/template/index2.html pada welcome.blade.php lakukan edit agar dapat diload, edit href="{{ asset('template/.....') }}" dan src="{{ asset('template/.....') }}" sesuai gambar berikut

```
<!-- Google Font: Source Sans Pro -->
<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
<!-- Font Awesome Icons -->
<link rel="stylesheet" href="{{ asset('tempalte/plugins/fontawesome-free/css/all.min.css') }}">
<!-- overlayScrollbars -->
<link rel="stylesheet" href="{{ asset('template/plugins/overlayScrollbars/css/OverlayScrollbars.min.css') }}">
<!-- Theme style -->
<link rel="stylesheet" href="{{ asset('template/dist/css/adminlte.min.css') }}">
</head>
<body class="hold-transition dark-mode sidebar-mini layout-fixed layout-navbar-fixed layout-footer-fixed">
<div class="wrapper">

<!-- Preloader -->
<div class="preloader flex-column justify-content-center align-items-center">

</div>
```

4. Scroll ke bagian paling bawah dan sesuaikan dengan kode berikut



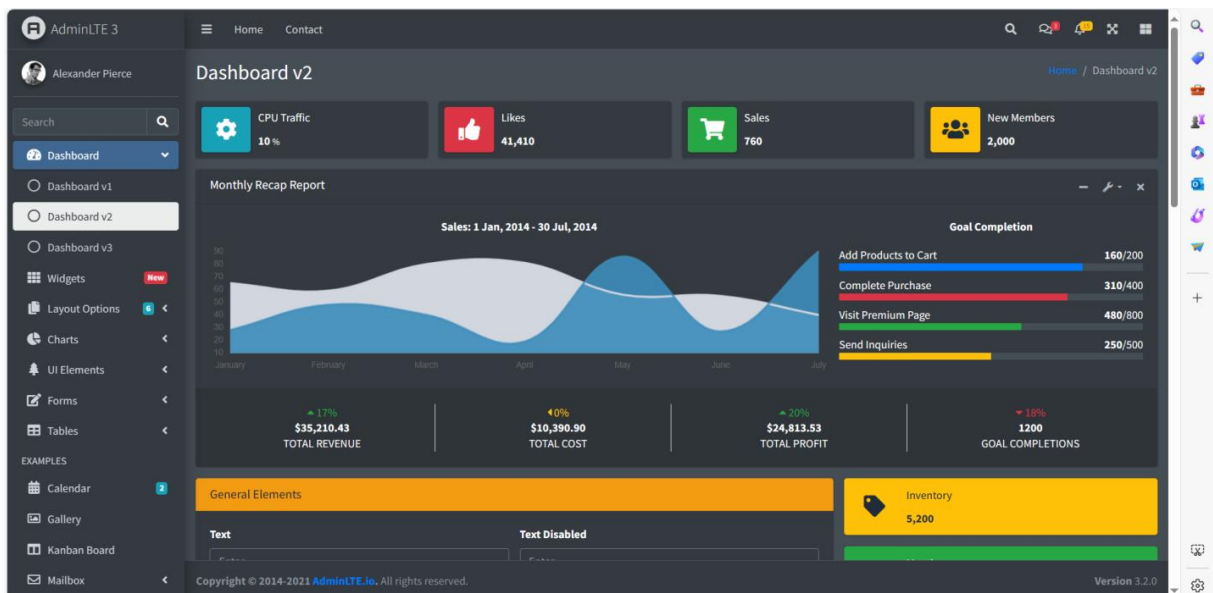
Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

```
<!-- REQUIRED SCRIPTS -->
<!-- jQuery -->
<script src="{{ asset('template/plugins/jquery/jquery.min.js') }}"></script>
<!-- Bootstrap -->
<script src="{{ asset('template/plugins/bootstrap/js/bootstrap.bundle.min.js') }}"></script>
<!-- overlayScrollbars -->
<script src="{{ asset('template/plugins/overlayScrollbars/js/jquery.overlayScrollbars.min.js') }}"></script>
<!-- AdminLTE App -->
<script src="{{ asset('template/dist/js/adminlte.js') }}"></script>

<!-- PAGE PLUGINS -->
<!-- jQuery Mapael -->
<script src="{{ asset('template/plugins/jquery-mousewheel/jquery.mousewheel.js') }}"></script>
<script src="{{ asset('template/plugins/raphael/raphael.min.js') }}"></script>
<script src="{{ asset('template/plugins/jquery-mapael/jquery.mapael.min.js') }}"></script>
<script src="{{ asset('template/plugins/jquery-mapael/maps/usa_states.min.js') }}"></script>
<!-- ChartJS -->
<script src="{{ asset('template/plugins/chart.js/Chart.min.js') }}"></script>

<!-- AdminLTE for demo purposes -->
<script src="{{ asset('template/dist/js/demo.js') }}"></script>
<!-- AdminLTE dashboard demo (This is only for demo purposes) -->
<script src="{{ asset('template/dist/js/pages/dashboard2.js') }}"></script>
</body>
</html>
```

5. Jalankan pada browser



6. Pada bagian menu terdapat pilihan Form kita khususnya dulu pada Form yang disediakan oleh AdminLTE



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

- General Elements
- Advanced Elements
- Editor
- Validation

Cara menggunakan template bisa dilihat nama formnya contoh judulnya 'General Elements', agar dapat digunakan dapat melakukan akses pada folder yang terdapat template adminLTE contoh : C:\laragon\www\PWL\_2024\public\template\pages\forms

7. Buka file pada form> general cari keyword dengan 'general elements' dan coba modifikasi pada welcome.blade.php

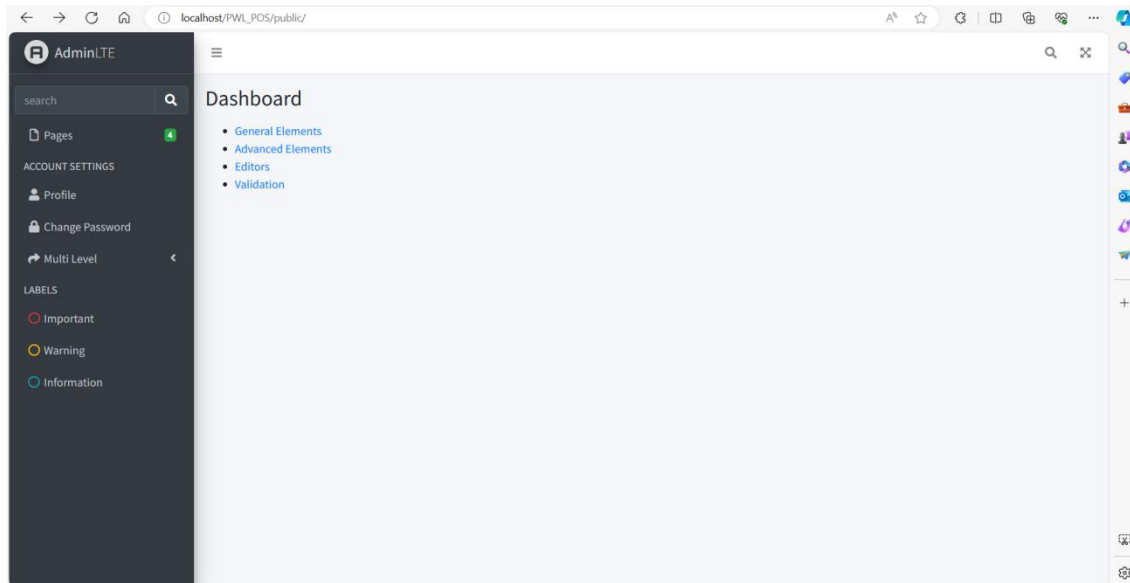
```
resources > views > welcome.blade.php > ...
1  @extends('adminlte::page')
2  @section('title', 'Dashboard')
3  @section('content_header')
4      <h1>Dashboard</h1>
5  @stop
6  @section('content')
7      <ul>
8          <li><a href="{{ url('/forms/general') }}">General Elements</a></li>
9          <li><a href="{{ url('/forms/advanced') }}">Advanced Elements</a></li>
10         <li><a href="{{ url('/forms/editors') }}">Editors</a></li>
11         <li><a href="{{ url('/forms/validation') }}">Validation</a></li>
12     </ul>
13 @stop
14 @section('css')
15     {{-- Add here extra stylesheets --}}
16     {{-- <link rel="stylesheet" href="/css/admin_custom.css"> --}}
17 @stop
18 @section('js')
19     <script>
20         console.log("Hi, I'm using the Laravel-AdminLTE package!");
21     </script>
22 @stop
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

8. Apa yang tampil?



9. Eksplorasi jenis form pada adminLTE dan coba terapkan yang sesuai untuk studi kasus POS PWL, buatlah form untuk tabel m\_user dan m\_level

● m\_user

```
resources > views > user_tambah.blade.php > ...
28 @extends('layouts.app')
29 {{-- Customize layout sections --}}
30 @section('subtitle', 'User')
31 @section('content_header_title', 'User')
32 @section('content_header_subtitle', 'Add Data')
33 @section('content')
34 <!-- general form elements disabled -->
35 <div class="card card-primary">
36   <div class="card-header">
37     <h3 class="card-title">Form Tambah Data User</h3>
38   </div>
39   <!-- /.card-header -->
40   <div class="card-body">
41     <form method="post" action="tambah_simpan">
42       {{ csrf_field() }}
43       <div class="row">
44         <div class="col-sm-12">
45           <!-- text input -->
46           <div class="form-group">
47             <label for="username">Username</label>
48             <input type="text" class="form-control" id="username" name="username" placeholder="Masukkan Username" required>
49           </div>
50           <div class="form-group">
51             <label for="nama">Nama</label>
52             <input type="text" class="form-control" id="nama" name="nama" placeholder="Masukkan Nama" required>
53           </div>
54           <div class="form-group">
55             <label for="password">Password</label>
56             <input type="password" class="form-control" id="password" name="password" placeholder="Masukkan Password" required>
57           </div>
58         </div>
59       </div>
60     </form>
61   </div>
62 </div>
```





## Jurusan Teknologi Informasi Politeknik Negeri Malang

### Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD

Mata Kuliah : Pemrograman Web Lanjut

```
58 <div class="form-group">
59 <label for="level_id">Level ID</label>
60 <input type="number" class="form-control" id="level_id" name="level_id" placeholder="Masukkan ID Level" required>
61 </div>
62 <div class="card-footer">
63 <a href=".." user" class="btn btn-secondary">Kembali</a>
64 <button type="submit" class="btn btn-primary">Add Data</button>
65 </div>
66 </div>
67
68 </form>
69 </div>
70 <!-- /.card-body -->
71 </div>
72 @stop
```

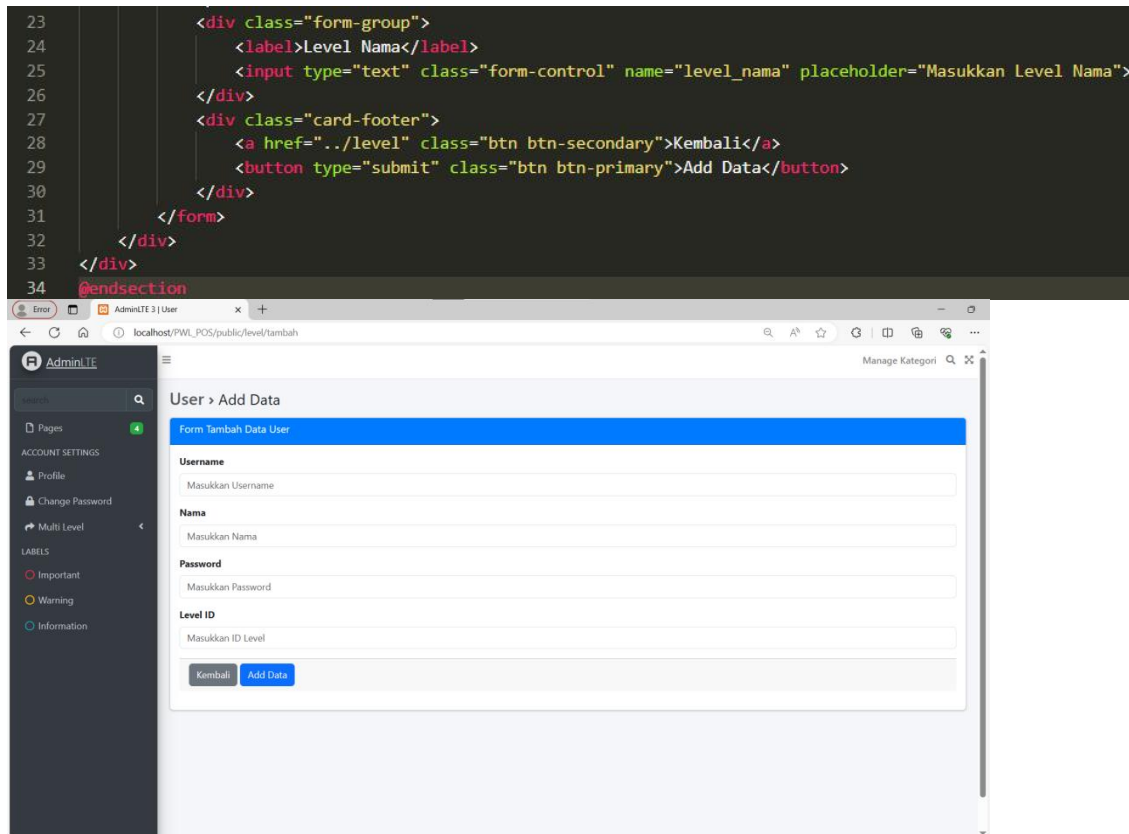
- m level

```
resources > views > level_tambah.blade.php > ...
1 @extends('layout.app')
2
3 {{-- Customize layout sections --}}
4 @section('subtitle', 'Level')
5 @section('content_header_title', 'Level')
6 @section('content_header_subtitle', 'Add Level')
7
8 {{-- Content body: main page content --}}
9 @section('content')
10 <div class="card card-primary">
11 <div class="card-header">
12 <h3 class="card-title">Form Tambah Level</h3>
13 </div>
14 <!-- /.card-header -->
15 <div class="card-body">
16 <form method="post" action="tambah_simpan">
17 <{{ csrf_field() }}>
18 <!-- text input -->
19 <div class="form-group">
20 <label>Level Kode</label>
21 <input type="text" class="form-control" name="level_kode" placeholder="Masukkan Level Kode">
22 </div>
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---



## B. VALIDASI PADA SERVER

1. Buka kembali file tugas di jobsheet 05, klik dan baca documentation ini [Validation -Laravel 10.x - The PHP Framework For Web Artisans](#)
2. Pastikan sudah terdefiniskan route pada web.php :

```
Route::get('/kategori/create', [KategoriController::class, 'create'])->name('kategori.create');  
Route::get('/kategori', [KategoriController::class, 'index'])->name('kategori.index');
```

3. Edit pada KategoriKontroller



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
app > Http > Controllers > KategoriController.php > KategoriController > edit
9  class KategoriController extends Controller
15
16      public function create()
17      {
18          return view('kategori.create');
19      }
20
21      public function store(Request $request): RedirectResponse
22      {
23          $validated = $request->validate([
24              'kategori_kode' => 'required',
25              'kategori_nama' => 'required',
26          ]);
27          return redirect('/kategori');
28      }
```

4. Berikut ini sebagai alternatif contoh penulisan validasi dan bisa Anda sesuaikan katakata dengan apa yang sedang Anda kerjakan,
5. Tulis perbedaan penggunaan validate dengan validateWithBag!
  - validate(): Metode validate() digunakan untuk mengembalikan respons langsung jika validasi gagal. Ini adalah pendekatan yang umum digunakan ketika Anda ingin menangani kesalahan validasi dengan mengembalikan respons langsung kepada pengguna, seperti menampilkan pesan kesalahan di halaman web.
  - validateWithBag(): Metode validateWithBag() digunakan untuk menyimpan pesan kesalahan validasi dalam "message bag" untuk digunakan kemudian dalam logika kontroler. Ini berguna jika Anda ingin menangani pesan kesalahan validasi secara terpisah, seperti menyimpannya dalam log atau mengirimkan email pemberitahuan.
  - Jadi, perbedaan utama adalah pada bagaimana hasil validasi ditangani: validate() secara langsung mengembalikan respons, sedangkan validateWithBag() memungkinkan Anda menyimpan pesan kesalahan untuk penggunaan selanjutnya.
6. Terkadang Anda mungkin ingin berhenti menjalankan aturan validasi pada suatu atribut setelah kegagalan validasi pertama. Untuk melakukannya, tetapkan bail aturan ke atribut: coba sesuaikan dengan field pada m\_kategori, yuk bisa.. apa yang terjadi?
  - Jika menetapkan aturan bail pada suatu atribut dalam validasi pada model m\_kategori pada framework Laravel, maka validasi akan berhenti di atribut tersebut setelah terjadi kegagalan validasi pertama. Artinya, jika validasi gagal untuk atribut tersebut, validasi tidak akan melanjutkan ke aturan validasi lainnya yang mungkin ada untuk atribut tersebut.



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

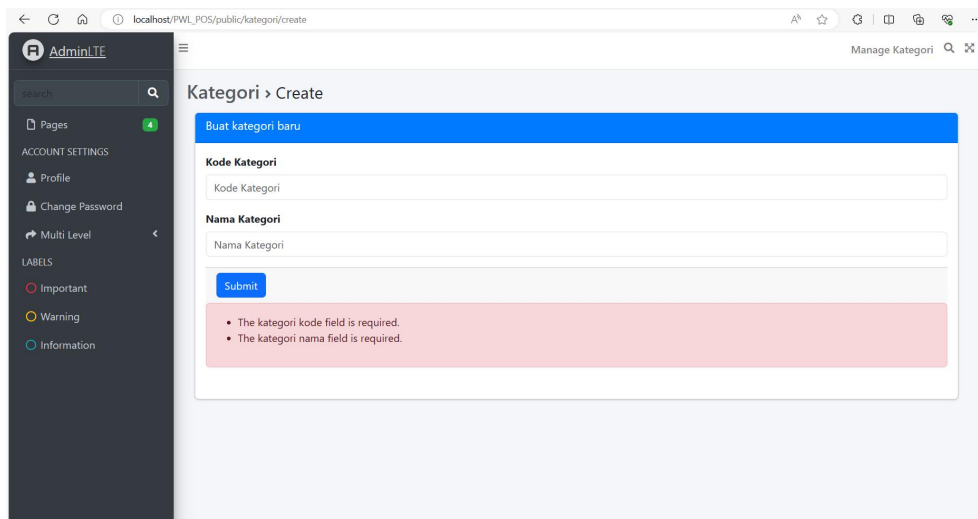
---

```
$validated = $request->validate([  
    'kategori_kode' => 'bail|required|unique:post|max:255',  
    'kategori_nama' => 'bail|required|unique:post|max:255',  
]);
```

7. Pada view/create.blade.php tambahkan code berikut agar ketika validasi gagal, kita dapat menampilkan pesan kesalahan dalam tampilan:

```
resources > views > kategori > create.blade.php > div.container > d  
/ @section( 'content' )  
8  
9 @if ( $errors->any() )  
10 <div class="alert alert-danger">  
11 <ul>  
12 @foreach ( $errors->all() as $error )  
13 <li>{{ $error }}</li>  
14 @endforeach  
15 </ul>  
16 </div>  
@endif
```

8. Hasil



- Ketika form tidak diisi data lalu diklik submit, maka akan muncul pesan kesalahan.
9. Catatan : Menyesuaikan Pesan Kesalahan Aturan validasi bawaan Laravel masing-masing memiliki pesan kesalahan yang terletak di lang/en/validation.php file aplikasi Anda. Jika aplikasi Anda tidak memiliki lang/direktori, Anda dapat menginstruksikan Laravel untuk membuatnya menggunakan lang:publish perintah Artisan. Di dalam lang/en/validation.php file tersebut, Anda akan menemukan entri terjemahan untuk setiap aturan validasi. Anda bebas mengubah atau memodifikasi pesan-pesan ini berdasarkan kebutuhan aplikasi Anda
10. Pada view/create.blade.php tambahkan dan coba running code berikut :





## Jurusan Teknologi Informasi Politeknik Negeri Malang

### Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD

Mata Kuliah : Pemrograman Web Lanjut

```
<input type="text" class="form-control @error('kategori_kode') is-invalid @enderror" id="kodeKategori" name="kodeKategori" placeholder="Kode Kategori">
@error('kategori_kode')
<div class="alert alert-danger">{{ $message }}</div>
@enderror
```

### C. FORM REQUEST VALIDATION

1. Pada view/create.blade.php tambahkan dan coba running code berikut :

```
PS C:\laragon\www\PWL_POS> php artisan make:request StorePostRequest
INFO Request [C:\laragon\www\PWL_POS\app\Http\Requests\StorePostRequest.php] created successfully.
```

2. Ketik kode berikut pada Http/request/StorePostRequest

```
app > Http > Requests > StorePostRequest.php > ...
7 class StorePostRequest extends FormRequest
21 {
22     public function rules(): array
23     {
24         return [
25             'kategori_kode' => 'required',
26             'kategori_nama' => 'required',
27         ];
28     }
}
```



## Jurusan Teknologi Informasi Politeknik Negeri Malang

### Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD

Mata Kuliah : Pemrograman Web Lanjut

```
app > Http > Controllers > KategoriController.php > KategoriController > store
9 class KategoriController extends Controller
31 public function store(StorePostRequest $request):RedirectResponse
32 {
33     // $validate = $request->validate(
34     //     [
35     //         'kodeKategori' => 'bail|required|max:5|unique:m_kategori,kategori
36     //         'namaKategori' => 'required'
37     //     ]
38     // );
39     $validated = $request->validated();
40
41     $validated = $request->safe()->only(['kategori_kode', 'kategori_nama']);
42     $validated = $request->safe()->except(['kategori_kode', 'kategori_nama']);
43
44     return redirect('/kategori');
45 }
```

3. Terapkan validasi juga pada tabel m\_user dan m\_level.

- user

```
app > Http > Controllers > UserController.php > UserController
10 class UserController extends Controller
23 public function tambah_simpan(StorePostRequest $request)
24 {
25     UserModel::create([
26         'username' => $request->username,
27         'nama' => $request->nama,
28         'password' => Hash::make($request->password),
29         'level_id' => $request->level_id,
30     ]);
31
32     $validated = $request->validate();
33     $validated = $request->safe()->only(['level_id', 'username', 'nama', 'password']);
34     $validated = $request->safe()->except(['level_id', 'username', 'nama', 'password']);
35
36     return redirect('/user');
37 }
```

- level



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
app > Http > Controllers > LevelController.php > levelController
8  class levelController extends Controller
29 public function tambah_simpan(StorePostRequest $request)
30 {
31     DB::insert('insert into m_level (level_kode, level_nama, created_at) values (?, ?, ?)', [$request->kodeLevel, $request->namaLevel, now()]);
32     return redirect('/level');
33 }
34 $validated = $request->validate();
35 $validated = $request->safe()->only(['level_kode', 'level_nama']);
36 $validated = $request->safe()->except(['level_kode', 'level_nama']);
37 return redirect('/level');
38 }
```

#### D. CRUD(create, read, update, delete)

1. Buat POSController lengkap dengan resourcenya, Membuat Resource Controller dan Route yang berfungsi untuk route CRUD sehingga tidak perlu repot-repot membuat masing-masing route seperti post, get, delete dan update

```
PS C:\laragon\www\PWL_POS> php artisan make:controller POSController--resource
INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\POSController--resource.php] created successfully.
```

2. Berikut perintahnya: sekarang tambahkan kode pada route/web.php

```
Route::resource('m_user', POSController::class);
```

3. Atur pada Models/m\_user

```
app > Models > m_user.php > m_user
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class m_user extends Model
9  {
10     protected $table = "m_user";
11     public $timestamps = false;
12     protected $primaryKey = 'user_id';
13
14     protected $fillable = [
15         'user_id',
16         'level_id',
17         'username',
18         'nama',
19         'password',
20     ];
21 }
```

4. Atur pada Migration/m\_user\_table



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
database > migrations > 2024_03_06_084503_create_m_user_table.php > class > down
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6  use App\Models\m_user;
7
8  return new class extends Migration
9  {
10
11      /**
12       * Run the migrations.
13       */
14      public function up(): void
15      {
16          Schema::create('useri', function (Blueprint $table) {
17              $table->id('user_id');
18              $table->unsignedBigInteger('level_id')->index(); //indexing untuk For
19              $table->string('username', 20)->unique(); //unique untuk memastikan t
20              $table->string('nama', 100);
21              $table->string('password');
22              $table->timestamps();
23
24              //mendefinisikan Foreign key pada kolom level_id mengacu pada kolom l
25              $table->foreign('level_id')->references('level_id')->on('m_level');
26          });
27
28      /**
29       * Reverse the migrations.
30       */
31      public function down(): void
32      {
33          Schema::dropIfExists('useri');
34      }
35  };
36
```

5. Pada file Controller yang kita dapat secara otomatis akan terdapat 7 fungsi berikut yang dapat kita gunakan untuk membuat operasi CRUD
6. Selanjutnya bukalah app/Http/Controllers/POSController.php kemudian ketikkan, kodenya seperti berikut ini:

```
<?php
namespace App\Http\Controllers;
use App\Models\m_user;
use Illuminate\Http\Request;

class POSController extends Controller
{
    /**
     * Display a listing of the resource.
     */
    public function index()
    {
        //fungsi eloquent menampilkan data menggunakan pagination
        $useri = m_user::all(); //Mengambil semua isi tabel
    }
}
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server  
Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
        return view('m_user.index', compact('useri'))->with('i');
    }

    /**
     * Show the form for creating a new resource.
     */
    public function create()
    {
        return view('m_user.create');
    }

    /**
     * Store a newly created resource in storage.
     */
    public function store(Request $request)
    {
        //Melakukan validasi data
        $request->validate([
            'user_id' => 'max 20',
            'username' => 'required',
            'nama' => 'required',
        ]);
        //fungsi eloquent untuk menambah data
        m_user::create($request->all());

        return redirect()->route('m_user.index')
            ->with('success', 'user Berhasil ditambahkan');
    }

    /**
     * Display the specified resource.
     */
    public function show(string $id, m_user $useri)
    {
        $useri = m_user::findOrFail($id);
        return view('m_user.show', compact('useri'));
    }

    /**
     * Show the form for editing the specified resource.
     */
    public function edit(string $id)
    {
        $useri = m_user::find($id);
        return view('m_user.edit', compact('useri'));
    }

    /**
```





Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
* Update the specified resource in storage.
*/
public function update(Request $request, string $id)
{
    $request->validate([
        'username' => 'required',
        'nama' => 'required',
        'password' => 'required',
    ]);
    //fungsi eloquent untuk mengupdate data inputan kita
    m_user::find($id)->update($request->all());
    //jika data berhasil diupdate, akan kembali ke halaman utama
    return redirect()->route('m_user.index')
    ->with('success', 'Data Berhasil Diupdate');
}

/**
 * Remove the specified resource from storage.
 */
public function destroy(string $id)
{
    $user = m_user::findOrFail($id)->delete();
    return \redirect()->route('m_user.index')
    ->with('success', 'Data Berhasil Dihapus');
}
}
```

7. Buatlah folder di Resources/Views/m\_user dengan beberapa blade dan isian kode berikut:

a) template.blade.php

```
resources > views > m_user > template.blade.php > html
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>CRUD Laravel</title>
5          <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
6      </head>
7      <body>
8          <div class="container">
9              @yield('content')
10         </div>
11     </body>
12 </html>
```

b) index.blade.php



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server  
Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
resources > views > m_user > index.blade.php > ...
1  @extends('m_user/template')
2  @section('content')
3      <div class="row mt-5 mb-5">
4          <div class="col-lg-12 margin-tb">
5              <div class="float-left">
6                  <h2>CRUD user</h2>
7              </div>
8              <div class="float-right">
9                  <a class="btn btn-success" href="{{ route('m_user.create') }}"> Input User</a>
10             </div>
11         </div>
12     </div>
13     @if ($message = Session::get('success'))
14         <div class="alert alert-success">
15             <p>{{ $message }}</p>
16         </div>
17     @endif
18     <table class="table table-bordered">
19         <tr>
20             <th width="20px" class="text-center">User id</th>
21             <th width="150px" class="text-center">Level id</th>
22             <th width="200px" class="text-center">username</th>
23             <th width="200px" class="text-center">nama</th>
24             <th width="150px" class="text-center">password</th>
25         </tr>
26         @foreach ($user as $m_user)
27             <tr>
28                 <td>{{ $m_user->user_id }}</td>
29                 <td>{{ $m_user->level_id }}</td>
30                 <td>{{ $m_user->username }}</td>
31                 <td>{{ $m_user->nama }}</td>
32                 <td>{{ $m_user->password }}</td>
33                 <td class="text-center">
34                     <form action="{{ route('m_user.destroy', $m_user->user_id) }}" method="POST">
35                         <a class="btn btn-info btn-sm" href="{{ route('m_user.show', $m_user->user_id) }}">Show</a>
36                         <a class="btn btn-primary btn-sm" href="{{ route('m_user.edit', $m_user->user_id) }}">Edit</a>
37                     @csrf
38                     @method('DELETE')
39                     <button type="submit" class="btn btn-danger btn-sm" onclick="return confirm('Apakah Anda yakin ingin menghapus data ini?')>Delete</button>
40                 </form>
41                 </td>
42             </tr>
43         @endforeach
44     </table>
45 @endsection
```

c) create.blade.php



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
resources > views > m_user > create.blade.php > div.alert.alert-danger > ul
1  @extends('m_user/template')
2
3  @section('content')
4  <div class="row mt-5 mb-5">
5      <div class="col-lg-12 margin-tb">
6          <div class="float-left">
7              <h2>Membuat Daftar User</h2>
8          </div>
9          <div class="float-right">
10             <a class="btn btn-secondary" href="{{ route('m_user.index') }}"> Kembali</a>
11          </div>
12      </div>
13  </div>
14
15  @if ($errors->any())
16  <div class="alert alert-danger">
17      <strong>Ops</strong> Input gagal<br><br>
18      <ul>
19          @foreach ($errors->all() as $error)
20              <li>{{ $error }}</li>
21          @endforeach
22      </ul>
23  </div>
24  @endif
25
26  <form action="{{ route('m_user.store') }}" method="POST">
27      @csrf
28      <div class="col-xs-12 col-sm-12 col-md-12">
29          <div class="form-group">
30              <strong>Username:</strong>
31              <input type="text" name="username" class="form-control" placeholder="Masukkan username">
32          </div>
33      </div>
34
35      <div class="col-xs-12 col-sm-12 col-md-12">
36          <div class="form-group">
37              <strong>Nama:</strong>
38              <input type="text" name="nama" class="form-control" placeholder="Masukkan nama">
39          </div>
40      </div>
41      <div class="col-xs-12 col-sm-12 col-md-12">
42          <div class="form-group">
43              <strong>Password:</strong>
44              <input type="password" name="password" class="form-control" placeholder="Masukkan password">
45          </div>
46      </div>
47
48      <div class="col-xs-12 col-sm-12 col-md-12 text-center">
49          <button type="submit" class="btn btn-primary">Submit</button>
50      </div>
51  </form>
52  @endsection
```

d) show.blade.php



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server  
Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

```
resources > views > m_user > show.blade.php > ...
1  @extends('m_user/template')
2
3  @section('content')
4  <div class="row mt-5 mb-5">
5      <div class="col-lg-12 margin-tb">
6          <div class="float-left">
7              <h2>Show User</h2>
8          </div>
9          <div class="float-right">
10             <a class="btn btn-secondary" href="{{ route('m_user.index') }}">Kembali</a>
11          </div>
12      </div>
13  </div>
14
15  <div class="row">
16      <div class="col-xs-12 col-sm-12 col-md-12">
17          <div class="form-group">
18              <strong>User_id:</strong> {{ $useri->user_id }}
19          </div>
20      </div>
21      <div class="col-xs-12 col-sm-12 col-md-12">
22          <div class="form-group">
23              <strong>Level_id:</strong> {{ $useri->level_id }}
24          </div>
25      </div>
26      <div class="col-xs-12 col-sm-12 col-md-12">
27          <div class="form-group">
28              <strong>Username:</strong> {{ $useri->username }}
29          </div>
30      </div>
31      <div class="col-xs-12 col-sm-12 col-md-12">
32          <div class="form-group">
33              <strong>Nama:</strong> {{ $useri->nama }}
34          </div>
35      </div>
36      <div class="col-xs-12 col-sm-12 col-md-12">
37          <div class="form-group">
38              <strong>Password:</strong> {{ $useri->password }}
39          </div>
40      </div>
41  </div>
42  @endsection
```

e) edit.blade.php



## Jurusan Teknologi Informasi Politeknik Negeri Malang

### Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD

Mata Kuliah : Pemrograman Web Lanjut

---

```
resources > views > m_user > edit.blade.php > ...
1  @extends('m_user/template')
2
3  @section('content')
4  <div class="row mt-5 mb-5">
5      <div class="col-lg-12 margin-tb">
6          <div class="float-left">
7              <h2>Edit User</h2>
8          </div>
9          <div class="float-right">
10             <a class="btn btn-secondary" href="{{ route('m_user.index') }}">Kembali</a>
11          </div>
12      </div>
13  </div>
14
15  @if ($errors->any())
16  <div class="alert alert-danger">
17      <strong>Ops!</strong> Error <br><br>
18      <ul>
19          @foreach ($errors->all() as $error)
20              <li>{{ $error }}</li>
21          @endforeach
22      </ul>
23  </div>
24  @endif
25
26  <form action="{{ route('m_user.update', $user->user_id) }}" method="POST">
27      @csrf
28      @method('PUT')
29      <div class="row">
30          <div class="col-xs-12 col-sm-12 col-md-12">
31              <div class="form-group">
32                  <strong>User_id:</strong>
33                  <input type="text" name="userid" value="{{ $user->user_id }}" class="form-control" placeholder="Masukkan user id">
34              </div>
35          </div>
36          <div class="col-xs-12 col-sm-12 col-md-12">
37              <div class="form-group">
38                  <strong>Level_id:</strong>
39                  <input type="text" name="levelid" value="{{ $user->level_id }}" class="form-control" placeholder="Masukkan level">
40              </div>
41          </div>
42          <div class="col-xs-12 col-sm-12 col-md-12">
43              <div class="form-group">
44                  <strong>Username:</strong>
45                  <input type="text" value="{{ $user->username }}" class="form-control" name="username" placeholder="Masukkan Nomor username">
46              </div>
47          </div>
48          <div class="col-xs-12 col-sm-12 col-md-12">
49              <div class="form-group">
50                  <strong>Nama:</strong>
51                  <input type="text" value="{{ $user->nama }}" name="nama" class="form-control" placeholder="Masukkan nama">
52              </div>
53          </div>
54          <div class="col-xs-12 col-sm-12 col-md-12">
55              <div class="form-group">
56                  <strong>Password:</strong>
57                  <input type="password" value="{{ $user->password }}" name="password" class="form-control" placeholder="Masukkan password">
58              </div>
59          </div>
60          <div class="col-xs-12 col-sm-12 col-md-12 text-center">
61              <button type="submit" class="btn btn-primary">Update</button>
62          </div>
63      </div>
64  </form>
65  @endsection
```

8. Silahkan akses localhost/m\_user





## Jurusan Teknologi Informasi Politeknik Negeri Malang

### Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD

Mata Kuliah : Pemrograman Web Lanjut

CRUD user

Input User

User id	Level id	username	nama	password	
1	1	admin	Administrator	\$2y\$12\$VqQaeT1AjhR..oiS4UT5.k97fIKTR5BVAsq4Ew4JFPvdGaVWTigu	Show Edit Delete
2	2	manager	Manager	\$2y\$12\$6i4esZmzpXcQdUwSziA7zOJ870d26laenhZfwpRL2Wde7MQdF.f/O	Show Edit Delete
3	3	staff	Staff/Kasir	\$2y\$12\$IHT.GX2pOR5awNDxONg6/OvnW8f0yLdCoibX5AqcFCXC6LZB2MyQm	Show Edit Delete
7	4	customer-1	Pelanggan Pertama	\$2y\$12\$oZ2Q.HWLC0ew2pTTOe/lye3eAKHJ8t0XjylEksGQLQ1CMh/hDVi	Show Edit Delete
8	2	manager_dua	Manager 2	\$2y\$12\$QaHv8GCArC49UzsqDlUykOHpjYtHqrKsunJnQr2jM3fBSfU9lkrC	Show Edit

### TUGAS

1. Coba tampilkan level\_id pada halaman web tersebut dimana field ini merupakan foreign key
2. Modifikasi dengan tema/ template kesukaan Anda

Show User

Kembali

User ID: 1

Level ID: 1

Username: admin

Nama: Administrator

Password: \$2y\$12\$VqQaeT1AjhR..oiS4UT5.k97fIKTR5BVAsq4Ew4JFPvdGaVWTigu

Membuat Daftar User

Kembali

Username:  
Masukkan username

Nama:  
Masukkan nama

Password:  
Masukkan password

Level ID:  
Masukkan level id

Submit



## Jurusan Teknologi Informasi Politeknik Negeri Malang

### Jobsheet-6: Template Form (AdminLTE), Server Validation, Client Validation, CRUD

Mata Kuliah : Pemrograman Web Lanjut

localhost/PWL\_POS/public/m\_user/1/edit

### Edit User

Kembali

User\_id:  
1

Level\_id:  
1

Username:  
admin

Nama:  
Administrator

Password:  
\*\*\*\*\*

Update

---

localhost/PWL\_POS/public/m\_user

### AdminLTE

Home > CRUD User

CRUD User

Input User

User id	Level id	Username	Name	Password	Actions
1	1	admin	Administrator	\$2y\$12\$VqQaeT1AjhR_oIS4UT5.k97fJKTR5BVA4Ew4fFPvdGaVWTigu	Show Edit Delete
2	2	manager	Manager	\$2y\$12\$6i4esZmzpXcQdUwSziA7zOJ870d26laenhzFwpRL2Wde7MQdF.f/O	Show Edit Delete
3	3	staff	Staff/Kasir	\$2y\$12\$SIHT.GX2pOR5awNDxONg6/OvnW8f0yLdColb5AqCFCX6LZB2MyQm	Show Edit Delete
7	4	customer-1	Pelanggan Pertama	\$2y\$12\$ozZQ.HWLC0ew2pTTOe/Ye3eAKHJ8t0XjyJlEksGQLQ1CMe/hDVi	Show Edit Delete
8	2	manager_dua	Manager 2	\$2y\$12\$Qahv8GCArO49UzsqDIUyKOHpjYHqKsUnJnQr2jM3fBSfU9JkrC	Show Edit Delete
9	2	manager22	Manager Dua Dua	\$2y\$12\$A/bBVsi8B5lgGM8vawcgBu8257OnbV4w2/vdRW5u1qVU668Tj4TW	Show Edit Delete
10	2	manager33	Manager Tiga Tiga	\$2y\$12\$WZ0Gy5jMAB07i3KHJLAM.ZEu7XgMdqP6r31MfJFNHo3s3.Hiw6VO	Show Edit Delete
11	2	manager56	Manager55	\$2y\$12\$HhCKpHNSt1kFPegdNK95..tKwb5dMKJcdlnVgVpw5kUF6FQ.eOKK	Show Edit Delete

3. Apa fungsi \$request->validate, \$error dan alert yang ada pada halaman CRUD tersebut

- Di halaman CRUD tersebut, \$request->validate, \$error, dan alert digunakan untuk validasi input dan menampilkan pesan kesalahan kepada pengguna jika ada kesalahan pada saat memasukkan atau memperbarui data.
- \$request->validate: Ini adalah metode untuk melakukan validasi input yang dikirimkan oleh pengguna. Ini memeriksa apakah data yang dimasukkan oleh pengguna sesuai dengan aturan validasi yang ditentukan sebelum data tersebut disimpan atau diperbarui di database. Jika validasi gagal, maka proses penyimpanan atau pembaruan data akan dihentikan dan pesan kesalahan akan ditampilkan kepada pengguna.
- \$error: Ini adalah variabel yang digunakan untuk menyimpan pesan-pesan kesalahan validasi yang dihasilkan oleh \$request->validate. Pesan-pesan kesalahan ini kemudian dapat



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Jobsheet-6: Template Form (AdminLTE), Server  
Validation, Client Validation, CRUD**  
Mata Kuliah : Pemrograman Web Lanjut

---

ditampilkan kepada pengguna untuk memberi tahu mereka tentang kesalahan yang terjadi pada saat validasi.

- alert: Ini adalah elemen HTML atau kode JavaScript yang digunakan untuk menampilkan pesan kesalahan kepada pengguna dalam bentuk pop-up atau pemberitahuan. Pesan-pesan kesalahan yang disimpan dalam variabel \$error dapat ditampilkan dalam elemen alert untuk memberi tahu pengguna tentang kesalahan yang terjadi dan membantu mereka untuk memperbaikinya. Biasanya, ini akan menjadi pesan-pesan kesalahan yang dihasilkan oleh validasi form sebelumnya.