



Nama : Dea Putri Nastiti

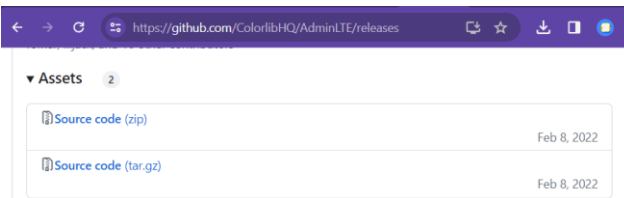
NIM : 2241720117

Kelas : TI 2H

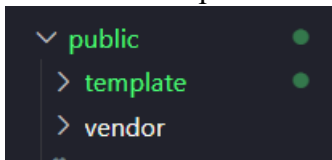
## JOBSHEET 6

### A. Template Form (AdminLTE)

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



```
welcome.blade.php > html > body.hold-transition.dark-mode.sidebar-mini.layout-fixed.layout-navbar-fixed.layout-footer-fixed > div.wrapper > div.preloader.flex-column.justify-content-center.align-items-center
<html lang="en">
<head>
<!-- AdminLTE 3 | Dashboard 2 </-->
<!-- 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('template/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

```
resources > views > welcome.blade.php > html > body.hold-transition.dark-mode.sidebar-mini.layout-fixed.layout-navbar-fixed.layout-footer-fixed > script
2 <html lang="en">
17 <body class="hold-transition dark-mode sidebar-mini layout-fixed layout-navbar-fixed layout-footer-fixed">
1713 <!-- REQUIRED SCRIPTS -->
1714 <!-- jQuery -->
1715 <script src="{{ asset('template/plugins/jquery/jquery.min.js') }}"></script>
1716 <!-- Bootstrap -->
1717 <script src="{{ asset('template/plugins/bootstrap/js/bootstrap.bundle.min.js') }}"></script>
1718 <!-- overlayScrollbars -->
1719 <script src="{{ asset('template/plugins/overlayScrollbars/js/jquery.overlayScrollbars.min.js') }}"></script>
1720 <!-- AdminLTE App -->
1721 <script src="{{ asset('template/dist/js/adminlte.js') }}"></script>
1722
1723 <!-- PAGE PLUGINS -->
1724 <!-- jQuery Mapael -->
1725 <script src="{{ asset('template/plugins/jquery-mousewheel/jquery.mousewheel.js') }}"></script>
1726 <script src="{{ asset('template/plugins/raphael/raphael.min.js') }}"></script>
1727 <script src="{{ asset('template/plugins/jquery-mapael/jquery.mapael.min.js') }}"></script>
1728 <script src="{{ asset('template/plugins/jquery-mapael/maps/usa_states.min.js') }}"></script>
1729 <!-- ChartJS -->
1730 <script src="{{ asset('template/plugins/chart.js/Chart.min.js') }}"></script>
1731
1732 <!-- AdminLTE for demo purposes -->
1733 <script src="{{ asset('template/dist/js/demo.js') }}"></script>
1734 <!-- AdminLTE dashboard demo (This is only for demo purposes) -->
1735 <script src="{{ asset('template/dist/js/pages/dashboard2.js') }}"></script>
1736 </body>
1737 </html>
<div class="preloader flex-column justify-content-center align-items-center">

8    <div class="card-header">
9      <h3 class="card-title">General Elements</h3>
10    </div>
11    <!-- /.card-header -->
12    <div class="card-body">
13      <form>
14        <div class="row">
15          <div class="col-sm-6">
16            <!-- text input -->
17            <div class="form-group">
18              <label>Text</label>
19              <input type="text" class="form-control" placeholder="Enter ...">
20            </div>
21          </div>
22          <div class="col-sm-6">
23            <div class="form-group">
24              <label>Text Disabled</label>
25              <input type="text" class="form-control" placeholder="Enter ..." disabled>
26            </div>
27          </div>
28        </div>
29        <div class="row">
30          <div class="col-sm-6">
31            <!-- textarea -->
32            <div class="form-group">
33              <label>Textarea</label>
34              <textarea class="form-control" rows="3" placeholder="Enter ..."></textarea>
35            </div>
36          </div>
37          <div class="col-sm-6">
38            <div class="form-group">
39              <label>Textarea Disabled</label>
40              <textarea class="form-control" rows="3" placeholder="Enter ..." disabled></textarea>
41            </div>
42          </div>
43        </div>
44      </form>
45      <!-- input states -->
46      <div class="form-group">
47        <label class="col-form-label" for="inputSuccess"><i class="fas fa-check"></i> Input with
48        success</label>
49        <input type="text" class="form-control is-valid" id="inputSuccess" placeholder="Enter ...">
50      </div>
51      <div class="form-group">
52        <label class="col-form-label" for="inputWarning"><i class="far fa-bell"></i> Input with
53        warning</label>
54        <input type="text" class="form-control is-warning" id="inputWarning" placeholder="Enter ...">
55      </div>
56      <div class="form-group">
57        <label class="col-form-label" for="inputError"><i class="far fa-times-circle"></i> Input with
58        error</label>
59        <input type="text" class="form-control is-invalid" id="inputError" placeholder="Enter ...">
60      </div>
61
62      <div class="row">
63        <div class="col-sm-6">
64          <!-- checkbox -->
65          <div class="form-group">
66            <div class="form-check">
67              <input class="form-check-input" type="checkbox">
68              <label class="form-check-label">Checkbox</label>
69            </div>
70            <div class="form-check">
71              <input class="form-check-input" type="checkbox" checked>
72              <label class="form-check-label">Checkbox checked</label>
73            </div>
74            <div class="form-check">
75              <input class="form-check-input" type="checkbox" disabled>
76              <label class="form-check-label">Checkbox disabled</label>
77            </div>
78          </div>
79        </div>
80        <div class="col-sm-6">
81          <!-- radio -->
82          <div class="form-group">
83            <div class="form-check">
84              <input class="form-check-input" type="radio" name="radio1">
85              <label class="form-check-label">Radio</label>
86            </div>
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
87 <div class="form-check">
88 <input class="form-check-input" type="radio" name="radio1" checked>
89 <label class="form-check-label">Radio checked</label>
90 </div>
91 <div class="form-check">
92 <input class="form-check-input" type="radio" disabled>
93 <label class="form-check-label">Radio disabled</label>
94 </div>
95 </div>
96 </div>
97 </div>
98
99 <div class="row">
100 <div class="col-sm-6">
101 <!-- select -->
102 <div class="form-group">
103 <label>Select</label>
104 <select class="form-control">
105 <option>option 1</option>
106 <option>option 2</option>
107 <option>option 3</option>
108 <option>option 4</option>
109 <option>option 5</option>
110 </select>
111 </div>
112 </div>
113 <div class="col-sm-6">
114 <div class="form-group">
115 <label>Select Disabled</label>
116 <select class="form-control" disabled>
117 <option>option 1</option>
118 <option>option 2</option>
119 <option>option 3</option>
120 <option>option 4</option>
121 <option>option 5</option>
122 </select>
123 </div>
124 </div>
125 </div>
126
127 <div class="row">
128 <div class="col-sm-6">
129 <!-- Select multiple -->
130 <div class="form-group">
131 <label>Select Multiple</label>
132 <select multiple class="form-control">
133 <option>option 1</option>
134 <option>option 2</option>
135 <option>option 3</option>
136 <option>option 4</option>
137 <option>option 5</option>
138 </select>
139 </div>
140 </div>
141 <div class="col-sm-6">
142 <div class="form-group">
143 <label>Select Multiple Disabled</label>
144 <select multiple class="form-control" disabled>
145 <option>option 1</option>
146 <option>option 2</option>
147 <option>option 3</option>
148 <option>option 4</option>
149 <option>option 5</option>
150 </select>
151 </div>
152 </div>
153 </div>
154 </form>
155 </div>
156 <!-- /.card-body -->
157 </div>
158
159 <div class="card-body">
160 <form>
161 <div class="row">
162 <div class="col-sm-6">
163 <!-- text input -->
164 <div class="form-group">
165 <label>Level id</label><input type="text" class="form-control" placeholder="id">
166 <div>
167 </div>
168 <button type="submit" class="btn btn-info">Submit </button>
169 </div>
170 @stop
171
172 @section('css')
173 {{{-- Add here extra stylesheets --}}
174 {{{-- <link rel="stylesheet" href="/css/admin_custom.css" --}} @stop
175
176 @section('js')
177 <script> console.log("Hi, I'm using the Laravel-AdminLTE package!"); </script> @stop<script> console.log("Hi, I'm using the Laravel-AdminLTE package!"); </script>
178 @stop
179
```



## 8. Apa yang tampil?

The screenshot shows the AdminLTE Dashboard. The left sidebar contains navigation links: Pages, Kategori, ACCOUNT SETTINGS (Profile, Change Password, Multi Level), and LABELS (Important, Warning, Information). The main content area is titled 'Dashboard' and 'General Elements'. It displays various form elements: Text input, Text Disabled, Textarea, Textarea Disabled, Input with success (green border), Input with warning (yellow border), Input with error (red border), and checkboxes/radio buttons. The bottom section shows a list of form elements: Checkbox, Radio, Input with success, Input with warning, Input with error, and Radio checked.

## 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

### a. User

#### UserDataTables

```
INFO DataTable [C:\laragon\www\PWL_POS\app\DataTables\UserDataTable.php] created successfully.

app > DataTables > UserDataTable.php > UserDataTable > dataTable > Closure
1 <?php
2
3 namespace App\DataTables;
4
5 use App\Models\User;
6 use App\Models\UserModel;
7 use Illuminate\Database\Eloquent\Builder as QueryBuilder;
8 use Yajra\DataTables\EloquentDataTable;
9 use Yajra\DataTables\Html\Builder as HtmlBuilder;
10 use Yajra\DataTables\Html\Column;
11 use Yajra\DataTables\Html\Editor\Editor;
12 use Yajra\DataTables\Html\Editor\Fields;
13 use Yajra\DataTables\Services\DataTable;
14
15 class UserDataTable extends DataTable
16 {
17     /**
18      * Build the DataTable class.
19      *
20      * @param QueryBuilder $query Results from query() method.
21      */
22     public function dataTable(QueryBuilder $query): EloquentDataTable
23     {
24         return (new EloquentDataTable($query))
25             ->addColumn('action', function($row){
26                 return '<a href="user/edit/" . $row->user_id . '
27                     " class="btn btn-primary">Edit</a>
28                     <a href="user/hapus/" . $row->user_id . '
29                     " class="delete btn btn-danger btn-sm">Hapus</a>';
30             })
31             ->setRowId('id');
32     }
33 }
34
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
39 public function query(UserModel $model): QueryBuilder
40 {
41     return $model->newQuery();
42 }
43
44 /**
45  * Optional method if you want to use the html builder.
46  */
47 public function html(): HtmlBuilder
48 {
49     return $this->builder()
50         ->setTableId('user-table')
51         ->columns($this->getColumns())
52         ->minifiedAjax()
53         // ->dom('BfRtip')
54         ->orderBy(1)
55         ->selectStyleSingle()
56         ->buttons([
57             Button::make('excel'),
58             Button::make('csv'),
59             Button::make('pdf'),
60             Button::make('print'),
61             Button::make('reset'),
62             Button::make('reload'),
63             Button::make('add')
64         ]);
65 }
66
67 /**
68  * Get the dataTable columns definition.
69  */
70 public function getColumns(): array
71 {
72     return [
73         Column::make('user_id'),
74         Column::make('level_id'),
75         Column::make('username'),
76         Column::make('nama'),
77         Column::make('created_at'),
78         Column::make('updated_at'),
79         Column::computed('action')
80             ->exportable(false)
81             ->printable(false)
82             ->width(160)
83             ->addClass('text-center'),
84     ];
85 }
86
87 /**
88  * Get the filename for export.
89  */
90 protected function filename(): string
91 {
92     return 'User_' . date('YmdHis');
93 }
94 }
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

## UserController.php

```
app > Http > Controllers > UserController.php > edit
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\DataTables\UserDataTable;
6  use Illuminate\Http\Request;
7  use App\Models\UserModel;
8  use Illuminate\Support\Facades\Hash;
9  use Illuminate\Database\Eloquent\ModelNotFoundException;
10 class UserController extends Controller
11 {
12
13     public function index(UserDataTable $dataTable){
14         return $dataTable->render('user.index');
15     }
16
17     public function create(){
18         return view('user.create');
19     }
20
21     public function store(Request $request){
22         UserModel::create([
23             'username' => $request->username,
24             'nama' => $request->nama,
25             'password' => Hash::make($request->password),
26             'level_id' => $request->level_id
27         ]);
28         return redirect('/user');
29     }
30
31
32     public function edit($id){
33         $user = UserModel::find($id);
34         return view('user.edit', ['data' => $user]);
35     }
36
37     public function storeEdit($id, Request $request){
38         $user = UserModel::find($id);
39
40         $user->username = $request->username;
41         $user->nama = $request->nama;
42         $user->password = Hash::make($request->password);
43         $user->level_id = $request->level_id;
44
45         $user->save();
46
47         return redirect('/user');
48     }
49
50     public function hapus($id){
51         $user = UserModel::find($id);
52         $user->delete();
53
54         return redirect('/user');
55     }
56 }
```

## VerivyCsrfToken.php

```
app > Http > Middleware > VerivyCsrfToken.php > ...
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Illuminate\Foundation\Http\Middleware\VerifyCsrfToken as Middleware;
6
7  class VerivyCsrfToken extends Middleware
8  {
9
10     /**
11      * The URIs that should be excluded from CSRF verification.
12      *
13      * @var array<int, string>
14      */
15     protected $except = [
16         '/kategori',
17         '/user',
18         '/level'
19     ];
20 }
```





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**  
**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

## UserModel.php

```
app > Models > UserModel.php > level
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Database\Eloquent\Relations\BelongsTo;
8
9 class UserModel extends Model
10 {
11     use HasFactory;
12
13     protected $table = 'm_user'; //Mendefinisikan nama tabel yang digunakan oleh model ini
14     protected $primaryKey = 'user_id'; //mendefinisikan primary key dari tabel yang digunakan
15     //protected $fillable = ['level_id', 'username', 'nama', 'password'];
16     protected $fillable = ['level_id', 'username', 'nama', 'password'];
17
18     public function level():BelongsTo{
19         return $this->belongsTo(LevelModel::class, 'level_id', 'level_id');
20     }
21 }
```

## adminlte.php

```
config > adminlte.php
3 return [
296     'menu' => [
332         [
333             'text' => 'User',
334             'url' => 'user',
335             'icon' => 'fas fa-fw list-alt',
336             'label_color' => 'success',
337         ],
338     ],
339 }
```

## index.blade.php

```
resources > views > user > index.blade.php > ...
1 @extends('layouts.app')
2
3 {{-- Customize layout sections --}}
4
5 @section('subtitle', 'User')
6 @section('content_header_title', 'Home')
7 @section('content_header_subtitle', 'User')
8
9 @section('content')
10     <div class="container">
11         <div class="card">
12             <div class="card-header">Manage User
13             </div>
14             <div class="card-body">
15                 {{ $dataTable->table() }}
16             </div>
17         </div>
18     </div>
19 @endsection
20
21 @push('scripts')
22     {{ $dataTable->scripts() }}
23 @endpush
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

## POLITEKNIK NEGERI MALANG JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

### create.blade.php

```
resources > views > user > create.blade.php > @extends('layouts.app')
1 @extends('layouts.app')
2 {{-- Customize layout sections --}}
3 @section('subtitle', 'User')
4 @section('content_header_title', 'User')
5 @section('content_header_subtitle', 'Create')
6 {{-- Content body: main page content --}}
7 @section('content')
8     <div class="container">
9         <div class="card card-primary">
10             <div class="card-header">
11                 <h3 class="card-title">Buat User baru</h3>
12             </div>
13
14             <form method="post" action="{{ route('kategori') }}">
15                 <div class="card-body">
16                     <div class="form-group">
17                         <label for="username">Username</label>
18                         <input type="text" class="form-control" id="username" name="username" placeholder="Masukkan Username">
19                     </div>
20                     <div class="form-group">
21                         <label for="nama">Nama</label>
22                         <input type="text" class="form-control" id="nama" name="nama" placeholder="Masukkan Nama">
23                     </div>
24                     <div class="form-group">
25                         <label for="password">Password</label>
26                         <input type="password" class="form-control" id="password" name="password" placeholder="Masukkan Password">
27                     </div>
28                     <div class="form-group">
29                         <label for="level">ID Level</label>
30                         <input type="text" class="form-control" id="level_id" name="level_id" placeholder="Masukkan ID Level">
31                     </div>
32                 </div>
33
34                 <div class="card-footer">
35                     <button type="submit" class="btn btn-primary">Submit</button>
36                 </div>
37             </form>
38         </div>
39     </div>
40 @endsection
```

### edit.blade.php

```
resources > views > user > edit.blade.php > @extends('layouts.app')
1 @extends('layouts.app')
2 {{-- Customize layout sections --}}
3 @section('subtitle', 'User')
4 @section('content_header_title', 'User')
5 @section('content_header_subtitle', 'Edit')
6 {{-- Content body: main page content --}}
7 @section('content')
8     <div class="container">
9         <div class="card card-primary">
10             <div class="card-header">
11                 <h3 class="card-title">Edit User</h3>
12             </div>
13
14             <form method="post" action="{{ route('kategori') }}">
15                 @csrf
16                 @method('PUT')
17
18                 <div class="card-body">
19                     <div class="form-group">
20                         <label for="username">Username</label>
21                         <input type="text" class="form-control" id="username" name="username" value="{{ $data->username }}">
22                     </div>
23                     <div class="form-group">
24                         <label for="nama">Nama</label>
25                         <input type="text" class="form-control" id="nama" name="nama" value="{{ $data->nama }}">
26                     </div>
27                     <div class="form-group">
28                         <label for="password">Password</label>
29                         <input type="password" class="form-control" id="password" name="password" value="{{ $data->password }}">
30                     </div>
31                     <div class="form-group">
32                         <label for="level_id">ID Level</label>
33                         <input type="text" class="form-control" id="level_id" name="level_id" value="{{ $data->level_id }}">
34                     </div>
35                 </div>
36
37                 <div class="card-footer">
38                     <button type="submit" class="btn btn-primary">Submit</button>
39                 </div>
40             </form>
41         </div>
42     </div>
43 @endsection
```



## web.php

```
39 //Manage User
40 Route::get('/user/create', [UserController::class, 'create']);
41 Route::post('/user', [UserController::class, 'store']);
42 Route::get('/user', [UserController::class, 'index']);
43 Route::get('/user/edit/{id}', [UserController::class, 'edit']);
44 Route::put('/user/{id}', [UserController::class, 'storeEdit']);
45 Route::get('/user/hapus/{id}', [UserController::class, 'hapus']);
```

## b. Level

### LevelDataTables

```
PS C:\laragon\www\PWL_POS> php artisan datatables:make Level
```

```
INFO DataTable [C:\laragon\www\PWL_POS\app\DataTables\LevelDataTable.php] created successfully.
```

```
app > DataTables > LevelDataTable.php > LevelDataTable > getColumns
1 <?php
2
3 namespace App\DataTables;
4
5 use App\Models\Level;
6 use App\Models\LevelModel;
7 use Illuminate\Database\Eloquent\Builder as QueryBuilder;
8 use Yajra\DataTables\EloquentDataTable;
9 use Yajra\DataTables\Html\Builder as HtmlBuilder;
10 use Yajra\DataTables\Html\Column;
11 use Yajra\DataTables\Html\Editor\Editor;
12 use Yajra\DataTables\Html\Editor\Fields;
13 use Yajra\DataTables\Services\DataTable;
14 use Yajra\DataTables\Html\Button;
15
16 class LevelDataTable extends DataTable
17 {
18     /**
19      * Build the DataTable class.
20      *
21      * @param QueryBuilder $query Results from query() method.
22      */
23     public function dataTable(QueryBuilder $query): EloquentDataTable
24     {
25         return (new EloquentDataTable($query))
26             ->addColumn('action', function($row){
27                 return '<a href="level/edit/' . $row->level_id . ' '
28                     . ' class="btn btn-primary">Edit</a>
29                 <a href="level/hapus/' . $row->level_id . ' '
30                     . ' class="delete btn btn-danger btn-sm">Hapus</a>';
31             })
32             ->setRowId('id');
33     }
34 }
```

```
37
38     * Get the query source of dataTable.
39     */
40     public function query(LevelModel $model): QueryBuilder
41     {
42         return $model->newQuery();
43     }
44
45     /**
46      * Optional method if you want to use the html builder.
47      */
48     public function html(): HtmlBuilder
49     {
50         return $this->builder()
51             ->setTableId('level-table')
52             ->columns($this->getColumns())
53             ->minifiedAjax()
54             //->dom('Bfrtip');
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
54         ->orderBy(1)
55         ->selectStyleSingle()
56         ->buttons([
57             Button::make('excel'),
58             Button::make('csv'),
59             Button::make('pdf'),
60             Button::make('print'),
61             Button::make('reset'),
62             Button::make('reload'),
63             Button::make('add')
64         ]);
65     }
66
67     /**
68      * Get the dataTable columns definition.
69      */
70     public function getColumns(): array
71     {
72         return [
73             Column::make('level_id'),
74             Column::make('level_kode'),
75             Column::make('level_nama'),
76             Column::make('created_at'),
77             Column::make('updated_at'),
78             Column::computed('action')
79                 ->exportable(false)
80                 ->printable(false)
81                 ->width(160)
82                 ->addClass('text-center'),
83         ];
84     }
85
86     /**
87      * Get the filename for export.
88      */
89     protected function filename(): string
90     {
91         return 'Level_' . date('YmdHis');
```

## LevelController.php

```
app > Http > Controllers > LevelController.php > LevelController > hapus
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\DB;
7  use App\DataTables\LevelDataTable;
8  use App\Models\LevelModel;
9
10 class LevelController extends Controller
11 {
12     public function index(LevelDataTable $dataTable){
13         return $dataTable->render('level.index');
14     }
15
16     public function create(){
17         return view('level.create');
18     }
19
20     public function store(Request $request){
21         LevelModel::create([
22             'level_kode' => $request->kodelevel,
23             'level_nama' => $request->namalevel,
24         ]);
25         return redirect('/level');
26     }
27 }
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**  
**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
28     public function edit($id){
29         $level = LevelModel::find($id);
30         return view('level.edit', ['data' => $level]);
31     }
32
33     public function storeEdit(Request $request, $id){
34         $level = LevelModel::find($id);
35
36         $level->level_kode = $request->kodelevel;
37         $level->level_nama = $request->namalevel;
38         $level->save();
39
40         return redirect('/level');
41     }
42
43     public function hapus($id){
44         $level = LevelModel::find($id);
45         $level->delete($id);
46
47         return redirect('/level');
48     }
49 }
```

### VerivyCsrfToken.php

```
app > Http > Middleware > VerifyCsrfToken.php > ...
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Illuminate\Foundation\Http\Middleware\VerifyCsrfToken as Middleware;
6
7  class VerifyCsrfToken extends Middleware
8  {
9      /**
10       * The URIs that should be excluded from CSRF verification.
11       *
12       * @var array<int, string>
13       */
14       protected $except = [
15           '/kategori',
16           '/user',
17           '/level'
18       ];
19 }
```

### LevelModel.php

```
app > Models > LevelModel.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Database\Eloquent\Relations\HasMany;
8
9  class LevelModel extends Model
10 {
11     use HasFactory;
12     protected $table = 'm_level'; //mendefinisikan nama tabel yang digunakan oleh model ini
13     protected $primaryKey = 'level_id'; //mendefinisikan primary key dari tabel yang dignakan
14
15     protected $fillable = ['level_id', 'level_kode', 'level_nama'];
16
17     public function user():HasMany{
18         return $this->hasMany(UserModel::class, 'level_id', 'level_id');
19     }
20 }
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

## adminlte.php

```
config > adminlte.php
3   return [
296   'menu' => [
338       [
339           'text' => 'Level',
340           'url' => 'level',
341           'icon' => 'fas fa-fw list-alt',
342           'label_color' => 'success',
343       ],

```

## index.blade.php

```
resources > views > level > index.blade.php > ...
1   @extends('layouts.app')
2
3   {{-- Customize layout sections --}}
4
5   @section('subtitle', 'Level')
6   @section('content_header_title', 'Home')
7   @section('content_header_subtitle', 'Level')
8
9   @section('content')
10      <div class="container">
11          <div class="card">
12              <div class="card-header">Manage Level
13              </div>
14              <div class="card-body">
15                  {{ $dataTable->table() }}
16              </div>
17          </div>
18      </div>
19  @endsection
20
21  @push('scripts')
22      {{ $dataTable->scripts() }}
23  @endpush

```

## create.blade.php

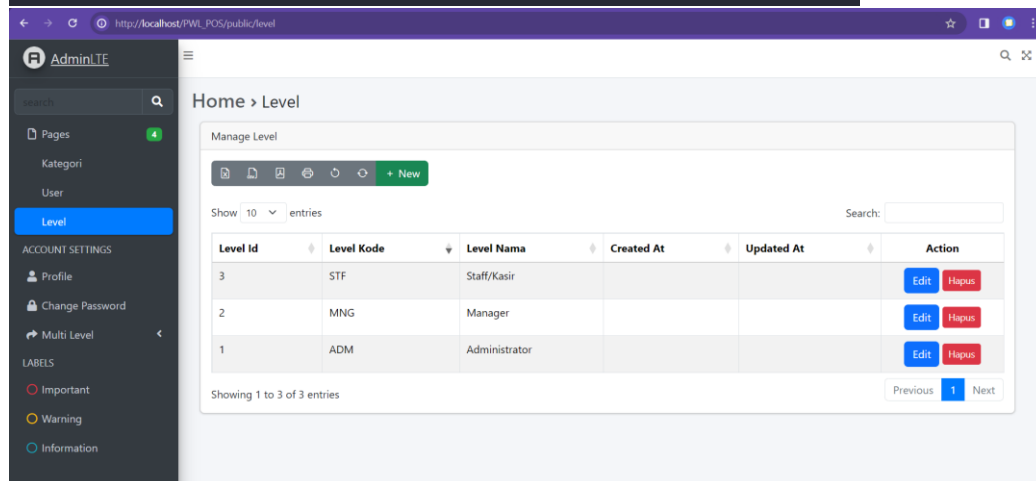
```
resources > views > level > create.blade.php > ...
1   @extends('layouts.app')
2   {{-- Customize layout sections --}}
3   @section('subtitle', 'Level')
4   @section('content_header_title', 'Level')
5   @section('content_header_subtitle', 'Create')
6   {{-- Content body, main page content --}}
7   @section('content')
8       <div class="container">
9           <div class="card card-primary">
10              <div class="card-header">
11                  <h3 class="card-title">Buat Level baru</h3>
12              </div>
13
14              <form method="post" action=".../level">
15                  <div class="card-body">
16                      <div class="form-group">
17                          <label for="kodeLevel">Kode Level</label>
18                          <input type="text" class="form-control" id="kodeLevel" name="kodeLevel" placeholder="Masukkan Kode Level">
19                      </div>
20                      <div class="form-group">
21                          <label for="namaLevel">Nama Level</label>
22                          <input type="text" class="form-control" id="namaLevel" name="namaLevel" placeholder="Masukkan Nama Level">
23                      </div>
24                  </div>
25
26                  <div class="card-footer">
27                      <button type="submit" class="btn btn-primary">Submit</button>
28                  </div>
29              </form>
30          </div>
31      </div>
32  @endsection

```



## edit.blade.php

```
resources > views > level > edit.blade.php > div.container > div.card.card-primary > form
1 @extends('layouts.app')
2 {{-- Customize layout sections --}}
3 @section('subtitle', 'Level')
4 @section('content_header_title', 'Level')
5 @section('content_header_subtitle', 'Edit')
6 {{-- Content body: main page content --}}
7 @section('content')
8     <div class="container">
9         <div class="card card-primary">
10             <div class="card-header">
11                 <h3 class="card-title">Edit Level</h3>
12             </div>
13             <form method="post" action="{{{ $data->level_id }}}">
14
15
16                 @csrf
17                 @method('PUT')
18
19                 <div class="card-body">
20                     <div class="form-group">
21                         <label for="kodeLevel">Kode Level</label>
22                         <input type="text" class="form-control" id="kodeLevel" name="kodeLevel" value="{{{ $data->level_kode }}}">
23                     </div>
24                     <div class="form-group">
25                         <label for="namaLevel">Nama Level</label>
26                         <input type="text" class="form-control" id="namaLevel" name="namaLevel" value="{{{ $data->level_nama }}}">
27                     </div>
28                 </div>
29                 <div class="card-footer">
30                     <button type="submit" class="btn btn-primary">Submit</button>
31                 </div>
32             </form>
33         </div>
34     </div>
35 @endsection
```



## 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 :

```
34 Route::get('/kategori/create', [KategoriController::class, 'create']);
35 Route::post('/kategori', [KategoriController::class, 'store']);
```

3. Edit pada KategoriKontroller dengan kode:



```
app > Http > Controllers > KategoriController.php > KategoriController > store
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB;
7 use App\DataTables\KategoriDataTable;
8 use App\Models\KategoriModel;
9 use Illuminate\Http\RedirectResponse;
10 use Illuminate\Support\Facades\Redirect;
11
12 class KategoriController extends Controller
13 {
14     public function index(KategoriDataTable $dataTable){
15         return $dataTable->render('kategori.index');
16     }
17
18     public function create(){
19         return view('kategori.create');
20     }
21
22     public function store(Request $request):RedirectResponse{
23         $validated = $request->validate([
24             'kategori_kode' => $request->kodeKategori,
25             'kategori_nama' => $request->namaKategori,
26         ]);
27         //The post is valid
28         return redirect('/kategori');
29     }
}
```

4. Berikut ini sebagai alternatif contoh penulisan validasi dan bisa Anda sesuaikan katakata dengan apa yang sedang Anda kerjakan,

Alternatively, validation rules may be specified as arrays of rules instead of a single | delimited string:

```
$validatedData = $request->validate([
    'title' => ['required', 'unique:posts', 'max:255'],
    'body' => ['required'],
]);
```

In addition, you may use the `validateWithBag` method to validate a request and store any error messages within a named error bag:

```
$validatedData = $request->validateWithBag('post', [
    'title' => ['required', 'unique:posts', 'max:255'],
    'body' => ['required'],
]);
```

5. Tulis perbedaan penggunaan `validate` dengan `validateWithBag`!  
Kedua metode ini memiliki tujuan yang sama, yaitu untuk memastikan bahwa data yang masuk ke dalam aplikasi sesuai dengan aturan validasi yang telah ditetapkan





1. validate:

- Metode validate digunakan untuk melakukan validasi input dan secara otomatis akan meredirect pengguna kembali ke form input jika terdapat kesalahan validasi.
- Jika validasi gagal, pesan error akan disimpan dalam sebuah session error default, yang biasanya dapat diakses dengan menggunakan \$errors variable di blade template.
- Cocok apabila hanya memiliki 1 form per halaman

2. validateWithBag:

- Sama seperti validate namun, perbedaannya terletak pada kemampuan untuk menentukan "bag" kustom untuk pesan error validasi.
- Dengan menggunakan validateWithBag, bisa menentukan nama session error bag yang berbeda untuk setiap form validasi.
- Cocok untuk banyak form per halaman dan ingin mengelompokkan pesan error validasi secara terpisah untuk masing-masing form.

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?

```
$request->validate([
    'title' => 'bail|required|unique:posts|max:255',
    'body' => 'required',
]);
```

```
app > Http > Controllers > KategoriController.php > KategoriController > store
12 class KategoriController extends Controller
13 {
14     public function create(){
15         return view('kategori.create');
16     }
17
18     public function store(Request $request):RedirectResponse{
19         $validated = $request->validate([
20             'kategori_kode' => 'bail|required|unique:post|max:255',
21             'kategori_nama' => 'bail|required|unique:post|max:255',
22         ]);
23         KategoriModel::create([
24             'kategori_kode' => $request->kodeKategori,
25             'namaKategori' => $request->namaKategori,
26         ]);
27         //The post is valid
28         return redirect('/kategori');
29     }
30 }
```



Saat kita mengisi data tapi tidak sesuai dengan peraturan atau ada data kosong yang belum diisi maka saat di submit akan kembali ke halaman create kategori dan data tidak akan disimpan

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 > ...
8      <div class="container">
9      <div class="card card-primary">
14     <form method="post" action="../kategori">
15     <div class="card-body">
24         @if ($errors->any())
25             <div class="alert alert-danger">
26                 <ul>
27                     @foreach ($errors->all() as $errors)
28                         <li>{{ $errors }}</li>
29                     @endforeach
30                 </ul>
31             </div>
32         @endif
33     </div>
```

8. Screenshot hasilnya.



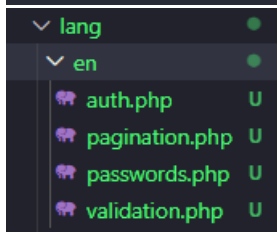
Maka akan ada pesan peringatan untuk mengisi kolom kode kategori dan nama kategori

9. Catatan : Menyesuaikan Pesan Kesalahan

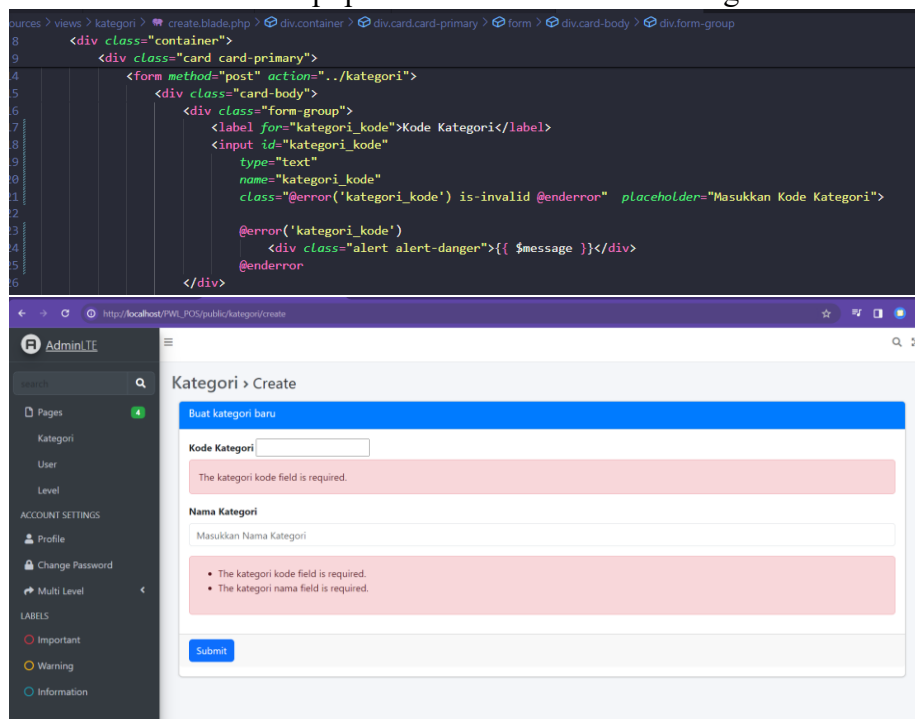
Aturan validasi bawaan Laravel masing-masing memiliki pesan kesalahan yang terletak di `lang/en/validation.php` file aplikasi Anda. Jika aplikasi Anda tidak memiliki `lang` direktori, Anda dapat menginstruksikan Laravel untuk membuatnya menggunakan `lang:publish` perintah Artisan. Di dalam `lang/en/validation.php` file tersebut, Anda akan menemukan entri terjemahan untuk setiap aturan validasi. Anda bebas mengubah atau memodifikasi pesan-pesan ini berdasarkan kebutuhan aplikasi Anda.

```
PS C:\laragon\www\PWL_POS> php artisan lang:publish
```

```
INFO Language files published successfully.
```



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





Pesan error apabila terdapat kesalahan akan berada di bawah kolom form yang error langsung, tidak seperti sebelumnya yang berada di bagian paling bawah form

### C. FORM REQUEST VALIDATION

- 1) Membuat permintaan form dengan menuliskan pada terminal

```
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.
```

kelas permintaan formulir yang dihasilkan akan ditempatkan di app/Http/Requests/direktori. Jika direktori ini tidak ada, maka akan dibuat saat Anda menjalankan make:request perintah. Setiap permintaan formulir yang dihasilkan oleh Laravel memiliki dua metode: authorize dan rules.

- 2) Ketik kode berikut pada Http/request/StorePostRequest

```
p > Http > Requests > StorePostRequest.php > StorePostRequest > store
8  class StorePostRequest extends FormRequest
23 {
24     public function rules(): array
25     {
26         return [
27             'kategori_kode' => 'required',
28             'kategori_nama' => 'required',
29         ];
30     }
31
32     public function store(StorePostRequest $request): RedirectResponse{
33         //the incoming request is valid...
34
35         //retrieve the validation input data...
36         $validated = $request->validated();
37
38         //retrieve a portion of the validated input data...
39         $validated = $request->safe()->only(['kategori_kode', 'kategori_nama']);
40         $validated = $request->safe()->except(['kategori_kode', 'kategori_nama']);
41
42         //store the post
43         return redirect('/kategori');
44     }
45 }
```

Note : Jika validasi gagal, respons pengalihan akan dihasilkan untuk mengirim pengguna kembali ke lokasi sebelumnya. Kesalahan juga akan ditampilkan ke sesi sehingga tersedia untuk ditampilkan. Jika permintaannya adalah permintaan XHR, respons HTTP dengan kode status 422 akan dikembalikan ke pengguna.

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



## StorePostRequest.php

```
app > Http > Requests > StorePostRequest.php > StorePostRequest > store
8 class StorePostRequest extends FormRequest
23 public function rules(): array
24 {
25     return [
26         'kategori_kode' => 'required',
27         'kategori_nama' => 'required',
28         'username' => 'required',
29         'nama' => 'required',
30         'password' => 'required',
31         'level_id' => 'required',
32     ];
33 }

app > Http > Requests > StorePostRequest.php > StorePostRequest > store
8 class StorePostRequest extends FormRequest
36 {
37     public function store(StorePostRequest $request): RedirectResponse{
38         //the incoming request is valid...
39
40         //retrieve the validation input data...
41         $validated = $request->validated();
42
43         //retrieve a portion of the validated input data...
44         $validated = $request->safe()->only(['kategori_kode', 'kategori_nama']);
45         $validated = $request->safe()->except(['kategori_kode', 'kategori_nama']);
46         $validated = $request->safe()->only(['username', 'nama', 'password', 'level_id']);
47         $validated = $request->safe()->except(['username', 'nama', 'password', 'level_id']);
48
49         //store the post
50         return redirect('/kategori');
51         return redirect('/user');
```

## UserController.php

```
app > Http > Controllers > UserController.php > UserController > index
12 class UserController extends Controller
19 public function create(){
20     return view('user.create');
21 }
22
23 public function store(Request $request):RedirectResponse{
24     $validated = $request->validate([
25         'username' => 'bail|required|unique:post|max:255',
26         'nama' => 'bail|required|unique:post|max:255',
27         'password' => 'bail|required|unique:post|max:255',
28         'level_id' => 'bail|required|unique:post|max:255',
29     ]);
30     UserModel::create([
31         'username' => $request->username,
32         'nama' => $request->nama,
33         'password' => Hash::make($request->password),
34         'level_id' => $request->level_id
35     ]);
36
37     return redirect('/user');
38 }
39
```



## Create.blade.php

```
resources > views > user > create.blade.php > div.container > div.card.card-primary > form
8      <div class="container">
9      <div class="card card-primary">
13
14      <form method="post" action="..../user">
15      <div class="card-body">
16      <div class="form-group">
17      <label for="usernameUser">Username</label>
18      <input id="username"
19      type="text"
20      name="username"
21      class="@error('username') is-invalid @enderror" placeholder="Masukkan Username">
22
23      @error('username')
24      <div class="alert alert-danger">{{ $message }}</div>
25      @enderror
26
27      </div>
28      <div class="form-group">
29      <label for="namaUser">Nama</label>
30      <input id="nama"
31      type="text"
32      name="nama"
33      class="@error('nama') is-invalid @enderror" placeholder="Masukkan Nama">
34
35      @error('nama')
36      <div class="alert alert-danger">{{ $message }}</div>
37      @enderror
38
39      </div>
40      <div class="form-group">
41      <label for="password">Password</label>
42      <input id="password"
43      type="password"
44      name="password"
45      class="@error('password') is-invalid @enderror" placeholder="Masukkan Password">
46
47      @error('password')
48      <div class="alert alert-danger">{{ $message }}</div>
49      @enderror
50
51      </div>
52      <div class="form-group">
53      <label for="idLevel">ID Level</label>
54      <input id="level_id"
55      type="text"
56      name="level_id"
57      class="@error('level_id') is-invalid @enderror" placeholder="Masukkan Level ID">
58
59      @error('level_id')
60      <div class="alert alert-danger">{{ $message }}</div>
61      @enderror
62
63      </div>
64      </div>
65      </form>
66      </div>
67      </div>
68      </div>
```



## - Level

### StorePostRequest.php

```
app > Http > Requests > StorePostRequest.php > StorePostRequest > store
8 class StorePostRequest extends FormRequest
23 public function rules(): array
24 {
25     return [
26         'kategori_kode' => 'required',
27         'kategori_nama' => 'required',
28         'username' => 'required',
29         'nama' => 'required',
30         'password' => 'required',
31         'level_id' => 'required',
32         'level_kode' => 'required',
33         'level_nama' => 'required',
34     ];
35 }
36
37 public function store(StorePostRequest $request): RedirectResponse{
38     //the incoming request is valid...
39
40     //retrieve the validation input data...
41     $validated = $request->validated();
42
43     //retrieve a portion of the validated input data...
44     $validated = $request->safe()->only(['kategori_kode', 'kategori_nama']);
45     $validated = $request->safe()->except(['kategori_kode', 'kategori_nama']);
46     $validated = $request->safe()->only(['username', 'nama', 'password', 'level_id']);
47     $validated = $request->safe()->except(['username', 'nama', 'password', 'level_id']);
48     $validated = $request->safe()->only(['level_kode', 'level_nama']);
49     $validated = $request->safe()->except(['level_kode', 'level_nama']);
50
51     //store the post
52     return redirect('/kategori');
53     return redirect('/user');
54     return redirect('/level');
55 }
```

### LevelController.php

```
app > Http > Controllers > LevelController.php > LevelController > store
12 class LevelController extends Controller
18 public function create(){
19     return view('level.create');
20 }
21
22 public function store(Request $request):RedirectResponse{
23     $validated = $request->validate([
24         'level_kode' => 'bail|required|unique:post|max:255',
25         'level_nama' => 'bail|required|unique:post|max:255',
26     ]);
27     LevelModel::create([
28         'level_kode' => $request->kodelevel,
29         'level_nama' => $request->namalevel,
30     ]);
31     return redirect('/level');
32 }
```

### Create.blade.php

```
resources > views > level > create.blade.php > div.container > div.card.card-primary > form > div.card-body > div.form-group > input#level_nama@error
8 <div class="container">
9     <div class="card card-primary">
10
11         <form method="post" action="../../../level">
12             <div class="card-body">
13                 <div class="form-group">
14                     <label for="kodelevel">Kode Level</label>
15                     <input id="level_kode"
16                         type="text"
17                         name="level_kode"
18                         class="@error('level_kode') is-invalid @enderror" placeholder="Masukkan Kode Level">
19
20                     @error('level_kode')
21                         <div class="alert alert-danger">{{ $message }}</div>
22                     @enderror
23                 </div>
24                 <div class="form-group">
25                     <label for="namalevel">Nama Level</label>
26                     <input id="level_nama"
27                         type="text"
28                         name="level_nama"
29                         class="@error('level_nama') is-invalid @enderror" placeholder="Masukkan Nama Level">
30
31                     @error('level_nama')
32                         <div class="alert alert-danger">{{ $message }}</div>
33                     @enderror
34                 </div>
35             </div>
36         </form>
37     </div>
38 </div>
```



#### 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 repet-repot membuat masing-masing route seperti post, get, delete dan update.

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

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

```
routes > web.php > ...
55 | Route::resource('m_user', POSController::class);
```

3. Atur pada Models/m\_user

```
PS C:\laragon\www\PWL_POS> php artisan make:model m_user
INFO Model [C:\laragon\www\PWL_POS\app\Models\m_user.php] created successfully.
```





```
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

```
database > migrations > 2024_03_06_085249_create_m_user_table.php > ...
1  <?php
2
3  use App\Models\m_user;
4  use Illuminate\Database\Migrations\Migration;
5  use Illuminate\Database\Schema\Blueprint;
6  use Illuminate\Support\Facades\Schema;
7
8  return new class extends Migration
9  {
10     /**
11      * Run the migrations.
12      */
13     public function up(): void
14     {
15         Schema::create('useri', function (Blueprint $table) {
16             $table->id('user_id');
17             $table->unsignedBigInteger('level_id')->index(); //indexing untuk Foreign Key
18             $table->string('username', 20)->unique(); //unique untuk memastikan tidak ada duplikat
19             $table->string('nama', 100);
20             $table->string('password');
21             $table->timestamps();
22
23             //Mendefinisikan Foreign Key pada kolom level_id di tabel m_level
24             $table->foreign('level_id')->references('level_id')->on('m_level');
25             //
26         });
27     }
28
29     /**
30      * Reverse the migrations.
31      */
32     public function down(): void
33     {
34         Schema::dropIfExists('useri');
35     }
36 };
```



5. Pada file Controller yang kita dapat secara otomatis akan terdapat 7 fungsi berikut yang dapat kita gunakan untuk membuat operasi CRUD.

- index()
- create()
- store()
- show()
- edit()
- update()
- destroy()

6. Selanjutnya bukalah app/Http/Controllers/POSController.php kemudian ketikkan, kodenya seperti berikut ini:

```
app > Http > Controllers > POSController.php > POSController > destroy
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\m_user;
6  use Illuminate\Http\Request;
7
8  class POSController extends Controller
9  {
10     /**
11      * Display a listing of the resource.
12      */
13     public function index()
14     {
15         //fungsi eloquent menampilkan data menggunakan pagination
16         $useri = m_user::all(); //Mengambil semua isi tabel
17         return view('m_user.index', compact('useri'))->with('i');
18     }
19
20     /**
21      * Show the form for creating a new resource.
22      */
23     public function create()
24     {
25         return view('m_user.create');
26     }
27
28     /**
29      * Store a newly created resource in storage.
30      */
31     public function store(Request $request)
32     {
33         //Melakukan validasi data
34         $request->validate([
35             'user_id' => 'max:20',
36             'username' => 'required',
37             'nama' => 'required',
38         ]);
39
40         //fungsi eloquent untuk menambah data
41         m_user::create($request->all());
42
43         return redirect()->route('m_user.index')
44             ->with('success', 'user Berhasil ditambahkan');
45     }
46
47     /**
48      * Display the specified resource.
49      */
50     public function show(string $id, m_user $useri)
51     {
52         $useri = m_user::findOrFail($id);
53         return view('m_user.show', compact('useri'));
54     }
55 }
```



```
55  /**
56   * Show the form for editing the specified resource.
57   */
58   public function edit(string $id)
59   {
60       $useri = m_user::find($id);
61       return view('m_user.edit', compact('useri'));
62   }
63
64   /**
65   * Update the specified resource in storage.
66   */
67   public function update(Request $request, string $id)
68   {
69       $request->validate([
70           'username' => 'required',
71           'nama' => 'required',
72           'password' => 'required',
73       ]);
74       //fungsi eloquent untuk mengupdate data inputan kita
75       m_user::find($id)->update($request->all());
76       //jika data berhasil diupdate, akan kembali ke halaman utama
77       return redirect()->route('m_user.index')
78         ->with('success', 'Data Berhasil Diupdate');
79   }
80
81   /**
82   * Remove the specified resource from storage.
83   */
84   public function destroy(string $id)
85   {
86       $user = m_user::findOrFail($id)->delete();
87       return redirect()->route('m_user.index')
88         ->with('success', 'Data Berhasil Dihapus');
89   }
90 }
```

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 > body
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

```
resources > views > m_user > index.blade.php > table.table-bordered
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
13          @if ($message = Session::get('success'))
14              <div class="alert alert-success">
15                  <p>{{ $message }}</p>
16              </div>
17          @endif
18          <table class="table table-bordered">
19              <tr>
20                  <th width="20px" class="text-center">User id</th>
21                  <th width="150px" class="text-center">Level id</th>
22                  <th width="200px" class="text-center">username</th>
23                  <th width="200px" class="text-center">nama</th>
24                  <th width="150px" class="text-center">password</th>
25              </tr>
26              @foreach ($useri as $m_user)
27                  <tr>
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
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
46
```

#### c. create.blade.php

```
resources > views > m_user > create.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>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
42 <div class="col-xs-12 col-sm-12 col-md-12">
43 <div class="form-group">
44 <strong>Password:</strong>
45 <input type="password" name="password" class="form-control" placeholder="Masukkan password">
46 </div>
47 </div>
48
49 <div class="col-xs-12 col-sm-12 col-md-12 text-center">
50 <button type="submit" class="btn btn-primary">Submit</button>
51 </div>
52 </form>
53 @endsection
54
```

#### d. show.blade.php

```
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> {{ $user->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> {{ $user->level_id }}
24 </div>
25 </div>
26
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
26 <div class="col-xs-12 col-sm-12 col-md-12">
27   <div class="form-group">
28     <strong>Username:</strong> {{ $user->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> {{ $user->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> {{ $user->password }}
39   </div>
40 </div>
41 </div>
42 @endsection
43
```

e. 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         @editUser/</div>
8       <div class="float-right">
9         <a class="btn btn-secondary href="{{ route('m_user.index') }}">Kembali</a>
10      </div>
11    </div>
12  </div>
13
14  @if ($errors->any())
15    <div class="alert alert-danger">
16      <strong>Ops!</strong> Error <div>
17        <ul>
18          @foreach ($errors->all() as $error)
19            <li>{{ $error }}</li>
20          @endforeach
21        </ul>
22      </div>
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>
43    <div class="col-xs-12 col-sm-12 col-md-12">
44      <div class="form-group">
45        <strong>Username:</strong>
46        <input type="text" value="{{ $user->username }}" class="form-control" name="username" placeholder="Masukkan Nomor username">
47      </div>
48    </div>
49    <div class="col-xs-12 col-sm-12 col-md-12">
50      <div class="form-group">
51        <strong>Nama:</strong>
52        <input type="text" value="{{ $user->nama }}" name="nama" class="form-control" placeholder="Masukkan nama">
53      </div>
54    </div>
55    <div class="col-xs-12 col-sm-12 col-md-12">
56      <div class="form-group">
57        <strong>Password:</strong>
58        <input type="password" value="{{ $user->password }}" name="password" class="form-control" placeholder="Masukkan password">
59      </div>
60    </div>
61    <div class="col-xs-12 col-sm-12 col-md-12 text-center">
62      <button type="submit" class="btn btn-primary">Update</button>
63    </div>
64  </div>
65 @endform
66 @endsection
```



8. Silahkan akses localhost/m\_user

CRUD user Input User

| User id | Level id | username     | nama          | password  |   |
|---------|----------|--------------|---------------|---|---|
| 1       | 1        | admin        | Administrator | \$2y\$12\$uRLgTycCq1YK4.kLTs3TteeihTsOPjEeOp0mUtuDWiuWAGgHJmx46 | <a>Show</a><br><a>Edit</a><br><a>Delete</a> |
| 2       | 2        | manager      | Manager       | \$2y\$12\$4MSbl/4wLM/oNALC8dzZBO9wISXVge0gNyGnaQNnkW7wKUvV/i9iO | <a>Show</a><br><a>Edit</a><br><a>Delete</a> |
| 3       | 3        | staff        | Staff/Kasir   | \$2y\$12\$FFtILtLca9k3XoTlwCda/ui5XLfo9nRBGLdPsWMny2Hcs.qnubx2q | <a>Show</a><br><a>Edit</a><br><a>Delete</a> |
| 6       | 2        | manager_tiga | Manager ke-3  | \$2y\$12\$vQFM1QPu0EM/hIPZYjKPauRjju2uukmLpn3WYq4k4bjWR8fbSoonO | <a>Show</a><br><a>Edit</a><br><a>Delete</a> |
| 7       | 2        | manager_dua  | Manager 2     | \$2y\$12\$1hYImryn2hPYyD9HBFr03usJZCJe749b7MtQL6Ja1OHDP9yLaA.C  | <a>Show</a><br><a>Edit</a><br><a>Delete</a> |

Input user

Membuat Daftar User Kembali

**Username:**

**Nama:**

**Password:**



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
**POLITEKNIK NEGERI MALANG**  
**JURUSAN TEKNOLOGI INFORMASI**  
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141  
Telp. (0341) 404424 – 404425, Fax (0341) 404420  
<http://www.polinema.ac.id>

## Show user



### Show User

[Kembali](#)

User\_id: 1

Level\_id: 1

Username: admin

Nama: Administrator

Password: \$2y\$12\$uRLgTycCq1YK4.kLTs3TteeihTsOPjEeOp0mUtuDWiuWAGgHJmx46

## Edit user



### Edit User

[Kembali](#)

User\_id:

Level\_id:

Username:

Nama:

Password:

[Update](#)



## Delete user

localhost says  
Apakah Anda yakin ingin menghapus data ini?

OK Cancel

Input User

CRUD user

| User id | Level id | username | nama          | password  |                        |
|---------|----------|----------|---------------|---|------------------------|
| 1       | 1        | admin    | Administrator | \$2y\$12\$uRLgTycCq1YK4.kLTs3TeeihTsOPJeOp0mUtuDWiuWAGgHJmx46   | Show<br>Edit<br>Delete |
| 2       | 2        | manager  | Manager       | \$2y\$12\$4MSbl/4wLM/oNALc8dzZBO9wiSXVge0gNyGnaQNnkW7wKUpv/i9iO | Show                   |

## Tugas:

1. Coba tampilkan level\_id pada halaman web tersebut dimana field ini merupakan foreign key

### - POSController

```
app > Http > Controllers > POSController.php > POSController > store
9 class POSController extends Controller
32 public function store(Request $request)
33 {
34     //Melakukan validasi data
35     $request->validate([
36         'user_id' => 'max 20',
37         'username' => 'required',
38         'nama' => 'required',
39         'password' => 'required',
40         'level_id' => 'required',
41     ]);
42     //fungsi eloquent untuk menambah data
43     //m_user::create($request->all());
44     m_user::create([
45         'username' => $request->username,
46         'nama' => $request->nama,
47         'password' => Hash::make($request->password),
48         'level_id' => $request->level_id,
49     ]);
50
51     return redirect()->route('m_user.index')
52     ->with('success', 'user Berhasil ditambahkan');
53 }
```

### - views/m\_user → index.blade.php

```
resources > views > m_user > index.blade.php > table.table-bordered > tr > th
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         <th width="90000" class="text-center">Actions</th>
26     </tr>
```





- views/m\_user → create.blade.php

```
resources > views > m_user > create.blade.php > ...
26 <form action="{{ route('m_user.store') }}" method="POST">
47 <div class="col-xs-12 col-sm-12 col-md-12">
48 <div class="form-group">
49 <strong>Level ID:</strong>
50 <input type="text" name="level_id" class="form-control" placeholder="Masukkan level id">
51 </div>
52 </div>
```

CRUD user

Input User

| User id | Level id | username     | nama          | password   | Actions          |
|---------|----------|--------------|---------------|--|------------------|
| 1       | 1        | admin        | Administrator | \$2y\$12\$uRlgTycCq1YK4.kLts3TteihTsOPJeOpOmUtuDWiuWAGGjHmu46  | Show Edit Delete |
| 2       | 2        | manager      | Manager       | \$2y\$12\$4MSbl/4wLM/oNALc8dzB09wlSXVge0ghyGnaQnkw7wKUpV/9iO   | Show Edit Delete |
| 3       | 3        | staff        | Staff/Kasir   | \$2y\$12\$FFTLTlca9k3XoThwCda/u5XLfo9nRBGLdPwWmny2Hcs.qnubx2q  | Show Edit Delete |
| 6       | 2        | manager_tiga | Manager ke-3  | \$2y\$12\$yQFM1QPu0EM/hIPZYjKPaUjjuZuukmlpn3WYq4k4bjWR8ft5oonO | Show Edit Delete |
| 7       | 2        | manager_dua  | Manager 2     | \$2y\$12\$1hYImryn2hPyD9HBFr0JusIZCie749b7MnQL6Ja1OHDP9yLaa.C  | Show Edit Delete |

Membuat Daftar User

Kembali

Username:  
manager123

Nama:  
Manajer satu dua tiga

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

Level ID:  
2

Submit

|    |   |            |                       |   |                  |
|----|---|------------|-----------------------|---|------------------|
| 16 | 2 | manager123 | Manajer satu dua tiga | \$2y\$12\$MX3FpvZ9zN93k94U/jY0SOTLcC1sV4NLTZR7Eil2iWZu2Eqthjwqu | Show Edit Delete |
|----|---|------------|-----------------------|---|------------------|



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**  
**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

## 2. Modifikasi dengan tema/ template kesukaan Anda

### - index.blade.php

```
resources > views > m_user > index.blade.php > div.card.card-secondary > div.card-body > table.table-bordered > tr > th.text-center
```

```
1 @extends('layouts.app')
2
3 @section('subtitle', 'M_User')
4 @section('content_header_title', 'Home')
5 @section('content_header_subtitle', 'CRUD User')
6
7 @section('content')
8 <div class="card card-secondary">
9   <div class="card-header bg-light">
10     <div class="row">
11       <div class="col-lg-12 margin-tb">
12         <div class="float-left">
13           <h1 style="font-size: 1.5rem;">CRUD User</h1>
14         </div>
15         <div class="float-right">
16           <a class="btn btn-primary" href="{{ route('m_user.create') }}">Input User</a>
17         </div>
18       </div>
19     </div>
20   </div>
21   <div class="card-body">
22
23     @if ($message = Session::get('success'))
24       <div class="alert alert-success">
25         <p>{{ $message }}</p>
26       </div>
27     @endif
28
29     <table class="table table-bordered">
30       <tr style="background-color: #f2f2f2; color: black;">
31         <th style="background-color: #A080E6; color: white; width: 20px;" class="text-center">User id</th>
32         <th style="background-color: #A080E6; color: white; width: 150px;" class="text-center">level id</th>
33         <th style="background-color: #A080E6; color: white; width: 200px;" class="text-center">username</th>
34         <th style="background-color: #A080E6; color: white; width: 200px;" class="text-center">nama</th>
35         <th style="background-color: #A080E6; color: white; width: 150px;" class="text-center">password</th>
36         <th style="background-color: #A080E6; color: white; width: 100px;" class="text-center">Actions</th>
37       </tr>
38       @foreach ($user as $m_user)
39         <tr>
40           <td>{{ $m_user->user_id }}</td>
41           <td>{{ $m_user->level_id }}</td>
42           <td>{{ $m_user->username }}</td>
43           <td>{{ $m_user->nama }}</td>
44           <td>{{ $m_user->password }}</td>
45           <td class="text-center">
46             <form action="{{ route('m_user.destroy', $m_user->user_id) }}" method="POST">
47               <a class="btn btn-info btn-sm" href="{{ route('m_user.show', $m_user->user_id) }}">Show</a>
48               <a class="btn btn-primary btn-sm" href="{{ route('m_user.edit', $m_user->user_id) }}">Edit</a>
49             @csrf
50             <method("DELETE")>
51               <button type="submit" class="btn btn-danger btn-sm" onclick="return confirm('Apakah Anda yakin ingin menghapus data ini?')>Delete</button>
52             </form>
53           </td>
54         </tr>
55       @endforeach
56     </table>
57   </div>
58 </div>
59 @endsection
```

AdminLTE

Home > CRUD User

CRUD User

Input User

| User id | Level id | username      | nama              | password   | Actions          |
|---------|----------|---------------|-------------------|--|------------------|
| 1       | 1        | admin         | Administrator     | \$2y\$12\$uRlgTycCq1YK4kLT3TeehtsOPJeOpOmUtuDWiwWAGgHmx46      | Show Edit Delete |
| 2       | 2        | manager       | Manager           | \$2y\$12\$4MSb/4wLM/oNALcBdzBZO9w5XVig0gNyGnaQnW7wKlupV/9IO    | Show Edit Delete |
| 3       | 3        | staff         | Staff/Kasir       | \$2y\$12\$FTTLca9K3XoTiwCda/ui5Xlfo9nRBGLdPpWMMny2Hcs.qnubx2q  | Show Edit Delete |
| 6       | 2        | manager_tiga  | Manager ke-3      | \$2y\$12\$+QFM1QPu0EM/hIPZYjKPaUgju2uukmlpn3Wlq4k4bJWR8ft5oonO | Show Edit Delete |
| 7       | 2        | manager_dua   | Manager 2         | \$2y\$12\$1Nymyn2hPy0YHBFr03uJ2CJe749b7MnQ6Ja1OHDp9yLaaC       | Show Edit Delete |
| 8       | 2        | manager_empat | Manager 3         | \$2y\$12\$PwNb5MhD63p.ZJOMavShuq4DpcYKxAK0TA0mtECUX3gBNBV.1m   | Show Edit Delete |
| 9       | 2        | manager22     | Manager Dua Dua   | \$2y\$12\$1bNPrs.EqRq0p4wsvLoYU2cg5oPIA65pnIdTHGZKSe.dHtZba    | Show Edit Delete |
| 10      | 2        | manager33     | Manager Tiga Tiga | \$2y\$12\$9V93c0b0yJhPiPHMtuqTAHTUOLHpOFMCpx/2mgaGdq1FpxYke    | Show Edit Delete |



- show.blade.php

```
1 @extends('n_user/template')
2
3 @section('content')
4     <div class="row mt-5 mb-5">
5         <div class="float-right">
6             <a class="btn btn-secondary" href="{ route('n_user.index') }" style="background-color: #FF6900; color: white; text-decoration: none;">Kembali</a>
7         </div>
8     </div>
9
10    <div class="card card-primary">
11        <div class="card-header" style="background-color: #6A0DAD; color: white;">
12            <h3 class="card-title">Show User</h3>
13        </div>
14        <div class="card-body">
15            <div class="text-muted">
16                <strong><i class="fas fa-book mr-1"></i> User ID</strong>
17                <p class="text-muted">{{ $user->id }}</p>
18            </div>
19            <div class="text-muted">
20                <strong><i class="fas fa-map-marker-alt mr-1"></i> Level ID</strong>
21                <p class="text-muted">{{ $user->level_id }}</p>
22            </div>
23            <div class="text-muted">
24                <strong><i class="fas fa-pencil-alt mr-1"></i> Username</strong>
25                <p class="text-muted">{{ $user->username }}</p>
26            </div>
27            <div class="text-muted">
28                <strong><i class="fas fa-file-alt mr-1"></i> Nama</strong>
29                <p class="text-muted">{{ $user->nama }}</p>
30            </div>
31            <div class="text-muted">
32                <strong><i class="fas fa-file-alt mr-1"></i> Password</strong>
33                <p class="text-muted">{{ $user->password }}</p>
34            </div>
35        </div>
36    </div>
37
38    <div class="card card-primary">
39        <div class="card-header" style="background-color: #6A0DAD; color: white;">
40            <h3 class="card-title">Show User</h3>
41        </div>
42        <div class="card-body">
43            <div class="text-muted">
44                <strong><i class="fas fa-book mr-1"></i> User ID</strong>
45                <p class="text-muted">{{ $user->id }}</p>
46            </div>
47            <div class="text-muted">
48                <strong><i class="fas fa-map-marker-alt mr-1"></i> Level ID</strong>
49                <p class="text-muted">{{ $user->level_id }}</p>
50            </div>
51            <div class="text-muted">
52                <strong><i class="fas fa-pencil-alt mr-1"></i> Username</strong>
53                <p class="text-muted">{{ $user->username }}</p>
54            </div>
55            <div class="text-muted">
56                <strong><i class="fas fa-file-alt mr-1"></i> Nama</strong>
57                <p class="text-muted">{{ $user->nama }}</p>
58            </div>
59            <div class="text-muted">
60                <strong><i class="fas fa-file-alt mr-1"></i> Password</strong>
61                <p class="text-muted">{{ $user->password }}</p>
62            </div>
63        </div>
64    </div>
65
66    @endsection
```

Kembali

| Show User |   |
|-----------|---|
| User ID   | 1   |
| Level ID  | 1   |
| Username  | admin   |
| Nama      | Administrator   |
| Password  | \$2y\$12\$uRlgTycCq1YK4kLts3TteehTsOPJfEaOp0mJtuDWuWAGgHjmw46 |

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

- \$request->validate : Digunakan untuk melakukan validasi data yang dikirim melalui form. Validate memastikan data yang masuk sesuai dengan kriteria yang telah ditentukan. Jika data tidak sesuai dengan kriteria, program akan kembali ke halaman form dan menampilkan pesan error



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

---

- **Errors** : Apabila data yang dimasukkan ke dalam form tidak sesuai dengan kriteria yang kita buat maka akan memunculkan pesan errors yang berisi pesan kesalahan validasi saat validasi gagal. Sehingga user dapat melihat dan memperbaiki kesalahan sesuai dengan pesan eror
- **Alert** : Digunakan untuk menampilkan feedback kepada user tentang hasil CRUD, seperti succes (menunjukkan operasi berhasil) atau gagal (pesan error jika terjadi kegagalan).