

Nama : Adinda Ivanka Maysanda Putri

Kelas : SIB 2B

NIM : 2341760058

JOBSHEET 4

MODEL dan ELOQUENT ORM

A. PROPERTI \$fillable DAN \$guarded

Praktikum 1 - \$fillable:

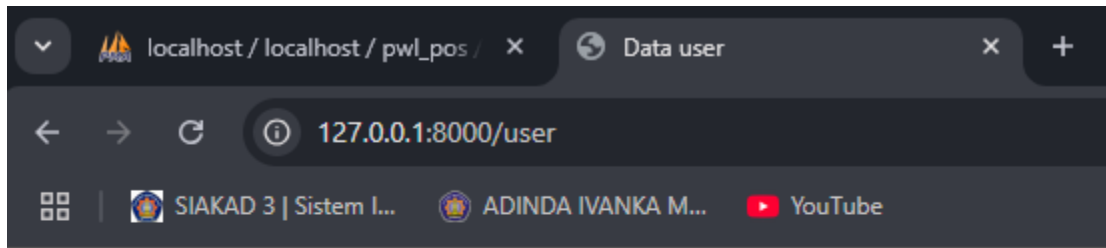
1. Buka file model dengan nama UserModel.php dan tambahkan \$fillable seperti gambar di bawah ini

```
protected $fillable = ['level_id', 'username', 'nama', 'password'];
```

2. Buka file controller dengan nama UserController.php dan ubah script untuk menambahkan data baru seperti gambar di bawah ini

```
$data = [  
    'level_id' => 2,  
    'username' => 'manager_dua',  
    'nama' => 'Manager 2',  
    'password' => Hash::make('12345')  
];  
UserModel::create($data);
```

3. Simpan kode program Langkah 1 dan 2, dan jalankan perintah web server. Kemudian jalankan link localhostPWL_POS/public/user pada browser dan amati apa yang terjadi



Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3
4	manager_dua	Manager 2	2

- Ubah file model UserModel.php seperti pada gambar di bawah ini pada bagian \$fillable

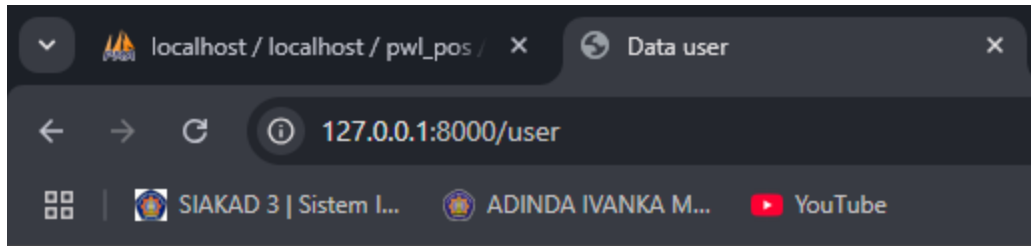
```
protected $fillable = ['level_id', 'username', 'nama'];
```

- Ubah kembali file controller UserController.php seperti pada gambar di bawah hanya bagian array pada \$data

```
public function index()
{
    $data = [
        'level_id' => 2,
        'username' => 'manager_tiga',
        'nama' => 'Manager 3',
        'password' => Hash::make('12345')
    ];
    UserModel::create($data);

    $user = UserModel::all();
    return view('user', ['data' => $user]);
}
```

- Simpan kode program Langkah 4 dan 5. Kemudian jalankan pada browser dan amati apa yang terjadi



Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3
4	manager_dua	Manager 2	2
5	manager_tiga	Manager 3	2

Sempat terjadi error, hal ini karena sebelumnya UserModel sebelumnya telah dimodifikasi yang mana menghapus kolom password, sehingga ketika ingin menambahkan data baru kolom yang ada kurang dan terjadi error. Untuk menjalankannya harus terlebih dahulu mengembalikan kolom password di UserModel. Halaman Browser (setelah UserModel ditambahkan kolom password)

7. Laporkan hasil Praktikum-1 ini dan commit perubahan pada git.

B. RETRIEVING SINGLE MODELS

Praktikum 2.1 – Retrieving Single Models

1. Buka file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

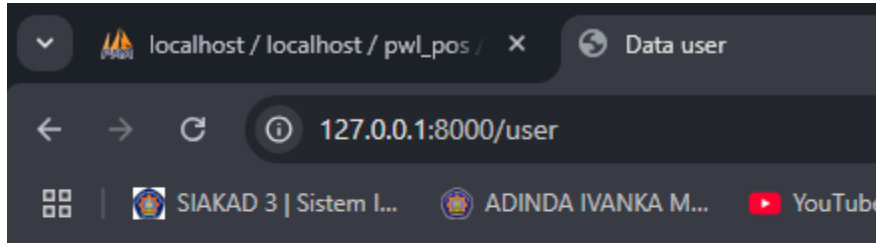
```
public function index()
{
    $user = UserModel::find(1);
    return view('user', ['data' => $user]);
}
```

2. Buka file view dengan nama user.blade.php dan ubah script seperti gambar di bawah ini

```
<tr>
    <td>{{ $data->user_id }}</td>
    <td>{{ $data->username }}</td>
    <td>{{ $data->nama }}</td>
```

```
<td>{{ $data->level_id }}</td>
</tr>
```

3. Simpan kode program Langkah 1 dan 2. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



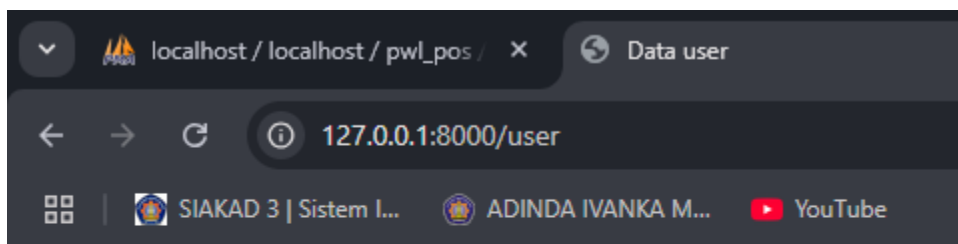
Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1

4. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::where('level_id', 1)->first();
    return view('user', ['data' => $user]);
}
```

5. Simpan kode program Langkah 4. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



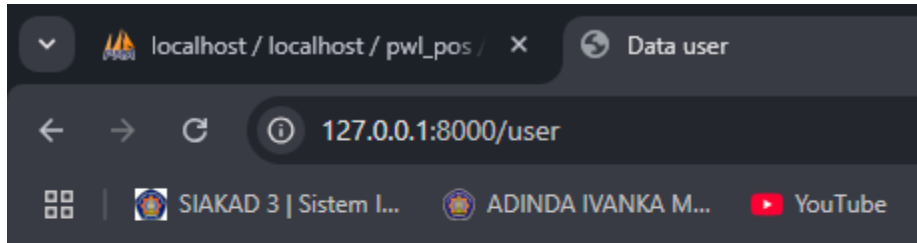
Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1

6. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $suser = UserModel::firstWhere('level_id', 1);
    return view('user', ['data' => $suser]);
}
```

7. Simpan kode program Langkah 6. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



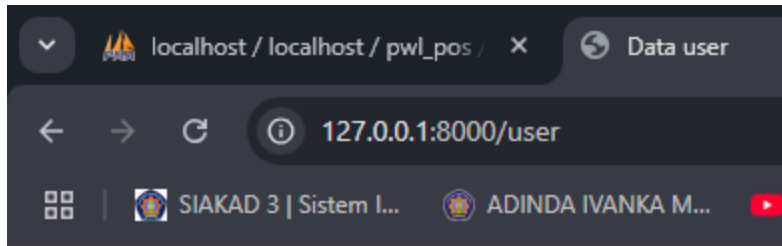
Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1

8. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $suser = UserModel::findOr(1, ['username', 'nama'], function () {
        abort(404);
    });
    return view('user', ['data' => $suser]);
}
```

9. Simpan kode program Langkah 8. Kemudian pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



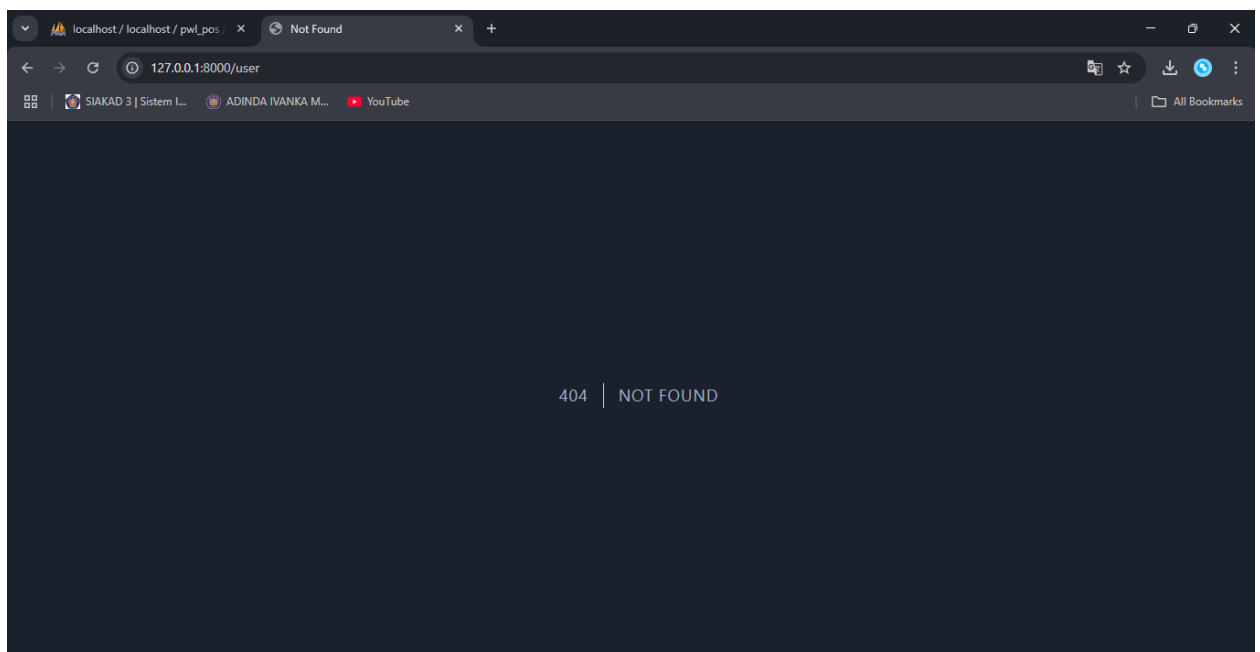
Data User

ID	Username	Nama	ID Level Pengguna
	admin	Administrator	

- Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::findOr(20, ['username', 'nama'], function () {
        abort(404);
    });
    return view('user', ['data' => $user]);
}
```

- Simpan kode program Langkah 10. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan

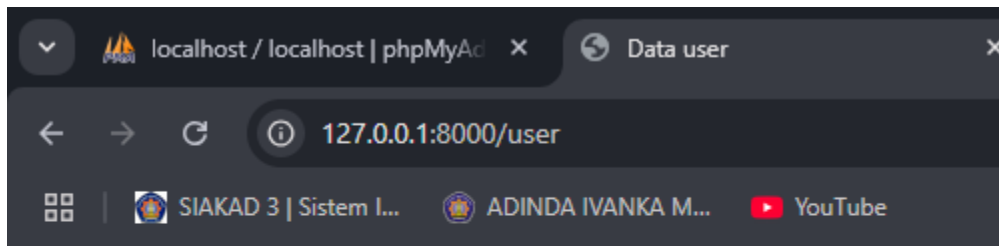


- Laporkan hasil Praktikum-2.1 ini dan commit perubahan pada git.

1. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::findOr(1);
    return view('user', ['data' => $user]);
}
```

2. Simpan kode program Langkah 1. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



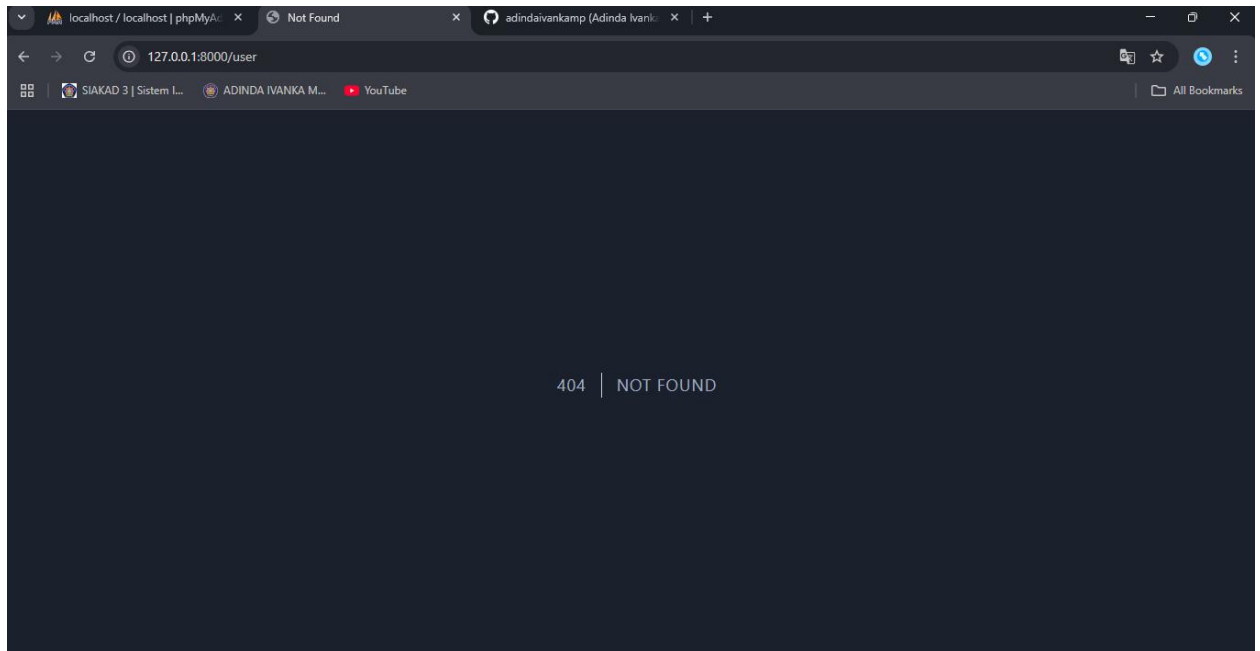
Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1

3. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::where('username', 'manager9')->firstOrFail();
    return view('user', ['data' => $user]);
}
```

4. Simpan kode program Langkah 3. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



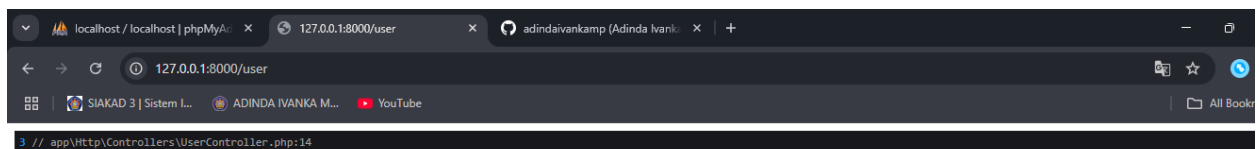
5. Laporkan hasil Praktikum-2.2 ini dan commit perubahan pada git.

Praktikum 2.3 – Retrieving Aggregates

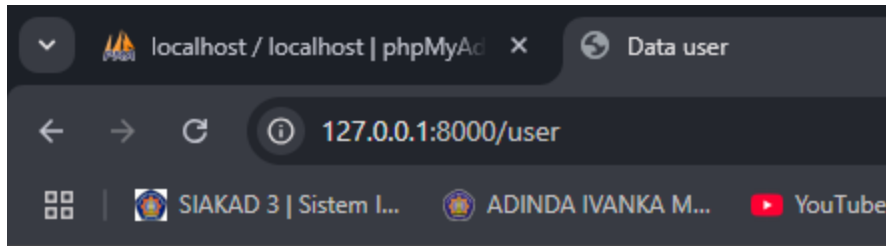
1. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::where('level_id', 2)->count(); dd($user);
    return view('user', ['data' => $user]);
}
```

2. Simpan kode program Langkah 1. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



3. Buat agar jumlah script pada langkah 1 bisa tampil pada halaman browser, sebagai contoh bisa lihat gambar di bawah ini dan ubah script pada file view supaya bisa muncul datanya



Data User

Jumlah Pengguna
3

4. Laporkan hasil Praktikum-2.3 ini dan commit perubahan pada git.

Praktikum 2.4 – Retrieving or Creating Models

1. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::firstOrCreate(
        [
            'username' => 'manager',
            'nama' => 'Manager',
        ]
    );
    return view('user', ['data' => $user]);
}
```

2. Ubah kembali file view dengan nama user.blade.php dan ubah script seperti gambar di bawah ini

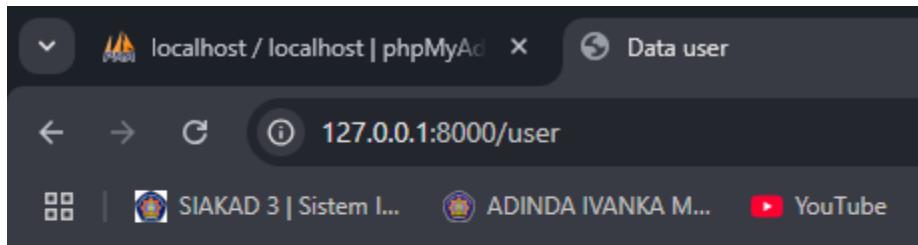
```
<tr>
    <th>ID</th>
    <th>Username</th>
    <th>Nama</th>
    <th>ID Level Pengguna</th>
</tr>
<tr>
    <td>{{ $data->user_id }}</td>
```

```

<td>{{ $data->username }}</td>
<td>{{ $data->nama }}</td>
<td>{{ $data->level_id }}</td>
</tr>

```

3. Simpan kode program Langkah 1 dan 2. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



Data User

ID	Username	Nama	ID Level Pengguna
2	manager	Manager	2

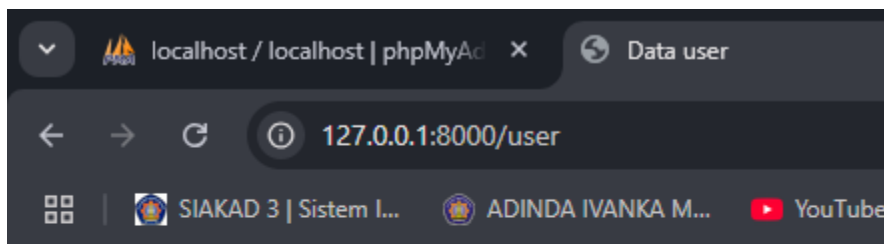
4. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```

'username' => 'manager22',
'nama' => 'Manager Dua Dua',
'password' => Hash::make('12345'),
'level_id' => 2

```

5. Simpan kode program Langkah 4. Kemudian jalankan pada browser dan amati apa yang terjadi dan cek juga pada phpMyAdmin pada tabel m_user serta beri penjelasan dalam laporan



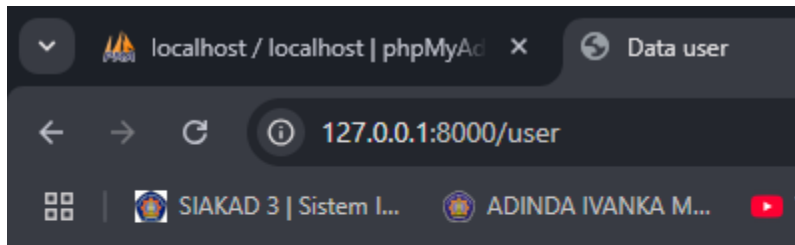
Data User

ID	Username	Nama	ID Level Pengguna
6	manager22	Manager Dua Dua	2

6. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::firstOrCreate(
        [
            'username' => 'manager',
            'nama' => 'Manager',
        ]
    );
    return view('user', ['data' => $user]);
}
```

7. Simpan kode program Langkah 6. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



Data User

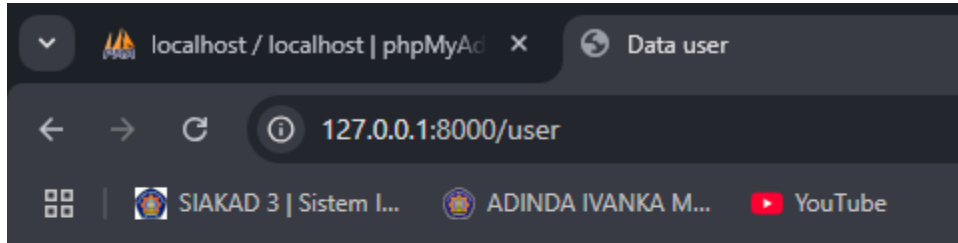
ID	Username	Nama	ID Level Pengguna
2	manager	Manager	2

8. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::firstOrCreate(
        [
            'username' => 'manager33',
            'nama' => 'Manager Tiga Tiga',
            'password' => Hash::make('12345'),
            'level_id' => 2
        ]
    );
}
```

```
return view('user', ['data' => $user]);
}
```

9. Simpan kode program Langkah 8. Kemudian jalankan pada browser dan amati apa yang terjadi dan cek juga pada phpMyAdmin pada tabel m_user serta beri penjelasan dalam laporan



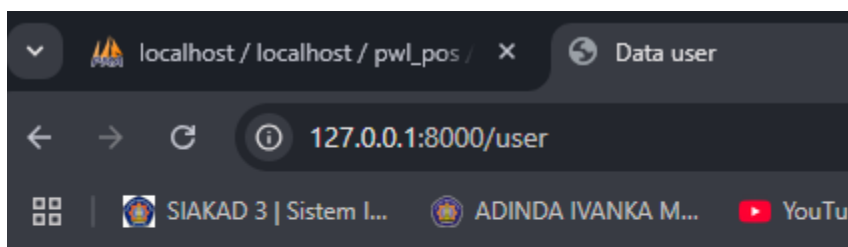
Data User

ID	Username	Nama	ID Level Pengguna
	manager33	Manager Tiga Tiga	2

	user_id	level_id	username	nama	password	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	1	admin	Administrator	\$2y\$12\$c/vqj8JijTSYq6rVPoTuR.JvkBVGE2PflVNRzRimX...	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	2	manager	Manager	\$2y\$12\$1bMr5lQQSjsJocRO5V1G2uixPGZCOM.cKnTMydXKJF...	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	3	staff	Staff/Kasir	\$2y\$12\$43j76XHZcyYFjDu1IKL4ReTDOftxbVJBVLQGYJLSV...	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	2	manager_dua	Manager 2	\$2y\$12\$SizU1aCTrkWeUp8YeefruBOObBwIURE0fBdMWoutUU...	2025-03-10 03:18:17	2025-03-10 03:18:17
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	2	manager_tiga	Manager 3	\$2y\$12\$IMjo4ERKFnHPT5p30fpuOilMURqEOpA5kQzEgMxw...	2025-03-10 03:43:22	2025-03-10 03:43:22
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	2	manager22	Manager Dua Dua	\$2y\$12\$VGTfjsPKNd.L2NmW8D8qIOYURTJr6U/HTSbqHxR8IUS...	2025-03-10 10:13:29	2025-03-10 10:13:29

10. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

11. Simpan kode program Langkah 9. Kemudian jalankan pada browser dan amati apa yang terjadi dan cek juga pada phpMyAdmin pada tabel m_user serta beri penjelasan dalam laporan



Data User

ID	Username	Nama	ID Level Pengguna
7	manager33	Manager Tiga Tiga	2

← T →		user_id	level_id	username	nama	password	created_at	updated_at
<input type="checkbox"/>	Edit Copy Delete	1	1	admin	Administrator	\$2y\$12\$c/vq8JijITSYq8rVPoTuR.JwkBVGE2PffVNRzRimX...	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	2	2	manager	Manager	\$2y\$12\$1bMr5/QQSjsJocRO5V1G2uXPGZCOM.oKnTMydXKJF...	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	3	3	staff	Staff/Kasir	\$2y\$12\$43j78XHZcyYPj0u1IkL4ReTDOfbxVJBvILQGyJLSV...	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	4	2	manager_dua	Manager 2	\$2y\$12\$SlzU1aCTrKweUpI8YeefruBOObBw/URE0fBdMW0utUU...	2025-03-10 03:18:17	2025-03-10 03:18:17
<input type="checkbox"/>	Edit Copy Delete	5	2	manager_tiga	Manager 3	\$2y\$12\$M.jo4ERKFhHPt5p30fpuOilMURqEOpA5kQzEgMxw...	2025-03-10 03:43:22	2025-03-10 03:43:22
<input type="checkbox"/>	Edit Copy Delete	6	2	manager22	Manager Dua Dua	\$2y\$12\$VGTfjsPKNd.L2NmM8D8qIOYURTJr6UIHTSbqHxR8IUS...	2025-03-10 10:13:29	2025-03-10 10:13:29
<input type="checkbox"/>	Edit Copy Delete	7	2	manager33	Manager Tiga Tiga	\$2y\$12\$E8uQ.kZNTn4Pankrfw5eOV4YkLpVWw7F3FP21YKEI/...	2025-03-10 10:20:25	2025-03-10 10:20:25

12. Laporkan hasil Praktikum-2.4 ini dan commit perubahan pada git.

Praktikum 2.5 – Attribute Changes

1. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```
public function index()
{
    $user = UserModel::create([
        'username' => 'manager55',
        'nama' => 'Manager55',
        'password' => Hash::make('12345'),
        'level_id' => 2,
    ]);

    $user->username = 'manager56';

    $user->isDirty(); // true
    $user->isDirty('username'); // true
    $user->isDirty('nama'); // false
    $user->isDirty(['nama', 'username']); // true

    $user->isClean(); // false
    $user->isClean('username'); // false
    $user->isClean('nama'); // true
    $user->isClean(['nama', 'username']); // false

    $user->save();

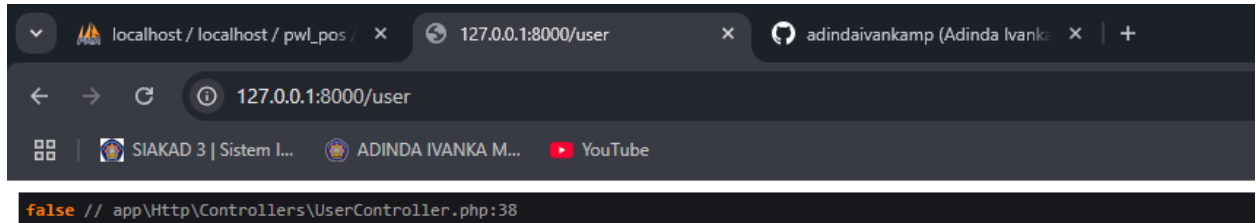
    $user->isDirty(); // false
    $user->isClean(); // true
}
```

```

        dd($user->isDirty());
    }

```

2. Simpan kode program Langkah 1. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



3. Ubah file controller dengan nama UserController.php dan ubah script seperti gambar di bawah ini

```

public function index()
{
    $user = UserModel::create([
        'username' => 'manager11',
        'nama' => 'Manager11',
        'password' => Hash::make('12345'),
        'level_id' => 2,
    ]);

    $user->username = 'manager12';

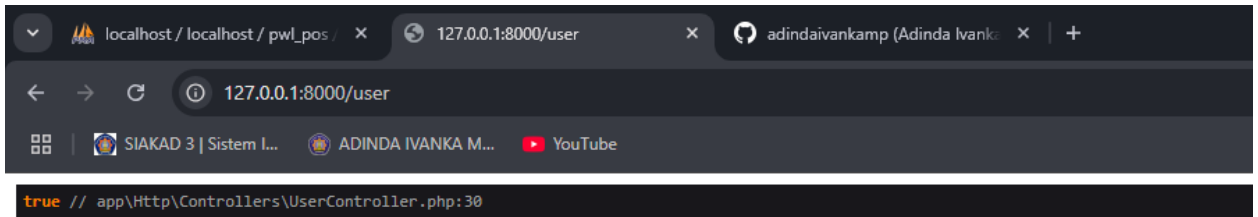
    $user->save();

    $user->wasChanged(); // true
    $user->wasChanged('username'); // true
    $user->wasChanged(['username', 'level_id']); // true
    $user->wasChanged('nama'); // false

    dd($user->wasChanged(['nama', 'username'])); // true
}

```

4. Simpan kode program Langkah 3. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



5. Laporkan hasil Praktikum-2.5 ini dan commit perubahan pada git.

Praktikum 2.6 – Create, Read, Update, Delete (CRUD)

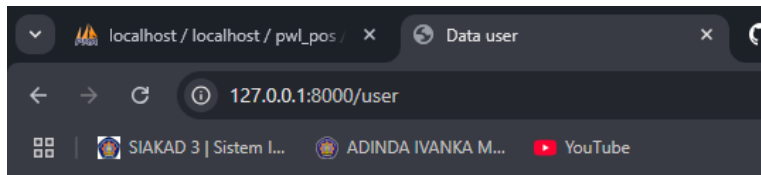
1. Buka file view pada user.blade.php dan buat scriptnya menjadi seperti di bawah ini

```
<body>
<h1>Data User</h1>
<a href="/user/tambah">Tambah User</a>
<table border="1" cellpadding="2" cellspacing="0">
  <tr>
    <td>ID</td>
    <td>Username</td>
    <td>Nama</td>
    <td>ID Level Pengguna</td>
    <td>Aksi</td>
  </tr>
  @foreach ($data as $d)
    <tr>
      <td>{{ $d->user_id }}</td>
      <td>{{ $d->username }}</td>
      <td>{{ $d->nama }}</td>
      <td>{{ $d->level_id }}</td>
      <td><a href="/user/ubah/{{ $d->user_id }}">Ubah</a> | <a href="/user/hapus/{{ $d->user_id
    }}">Hapus</a></td>
    </tr>
  @endforeach
</table>
</body>
```

2. Buka file controller pada UserController.php dan buat scriptnya untuk read menjadi seperti di bawah ini

```
public function index() {
    $user = UserModel::all();
    return view('user', ['data' => $user]);
}
```

3. Simpan kode program Langkah 1 dan 2. Kemudian jalankan pada browser dan amati apa yang terjadi dan beri penjelasan dalam laporan



Data User

[Tambah User](#)

ID	Username	Nama	ID Level Pengguna	Aksi
1	admin	Administrator	1	Ubah Hapus
2	manager	Manager	2	Ubah Hapus
3	staff	Staff/Kasir	3	Ubah Hapus
4	manager_dua	Manager 2	2	Ubah Hapus
5	manager_tiga	Manager 3	2	Ubah Hapus
6	manager22	Manager Dua Dua	2	Ubah Hapus
7	manager33	Manager Tiga Tiga	2	Ubah Hapus
9	manager56	Manager55	2	Ubah Hapus
10	manager12	Manager11	2	Ubah Hapus

4. Langkah berikutnya membuat create atau tambah data user dengan cara bikin file baru pada view dengan nama user_tambah.blade.php dan buat scriptnya menjadi seperti di bawah ini

```
PS C:\laragon\www\Pemrograman-Web-Lanjut\Minggu 4> php artisan make:view user_tambah
```

```
INFO View [C:\laragon\www\Pemrograman-Web-Lanjut\Minggu 4\resources\views\user_tambah.blade.php] created successfully.
```

```
<html>
<head>
    <title>Tambah User</title>
</head>
<body>
    <h1>Form Tambah Data User</h1>
    <form method="post" action="/user/tambah_simpan">
        {{ csrf_field() }}
        <label>Username</label>
        <input type="text" name="username" placeholder="Masukan Username">
        <br>
```



```

<label>Nama</label>
<input type="text" name="nama" placeholder="Masukan Nama">
<br>
<label>Password</label>
<input type="password" name="password" placeholder="Masukan Password">
<br>
<label>Level ID</label>
<input type="number" name="level_id" placeholder="Masukan ID Level">
<br><br>
<input type="submit" class="btn btn-success" value="Simpan">
</form>
</body>
</html>

```

5. Tambahkan script pada routes dengan nama file web.php. Tambahkan seperti gambar di bawah ini

```
Route::get('/user/tambah', [UserController::class, 'tambah']);
```

6. Tambahkan script pada controller dengan nama file UserController.php. Tambahkan script dalam class dan buat method baru dengan nama tambah dan diletakan di bawah method index seperti gambar di bawah ini

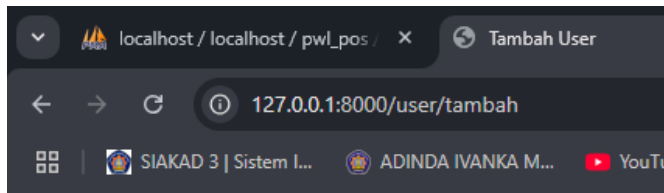
```

public function index() {
    $user = UserModel::all();
    return view('user', ['data' => $user]);
}

public function tambah(){
    return view('user_tambah');
}

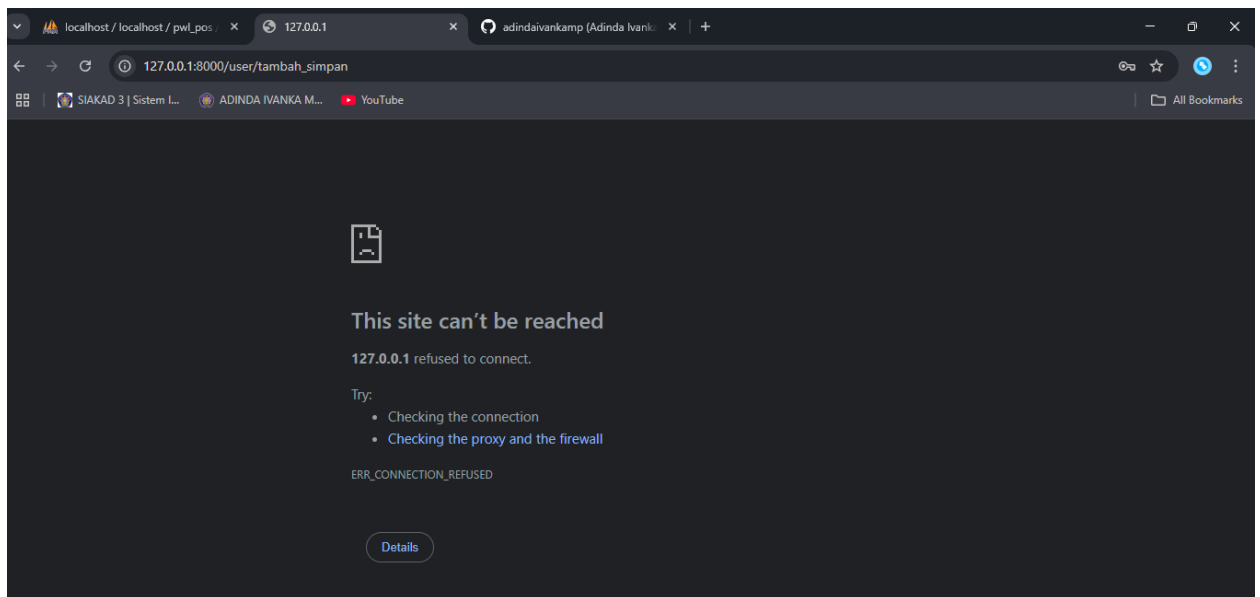
```

7. Simpan kode program Langkah 4 s/d 6. Kemudian jalankan pada browser dan klik link “+ Tambah User” amati apa yang terjadi dan beri penjelasan dalam laporan



Form Tambah Data User

Username
 Nama
 Password
 Level ID



8. Tambahkan script pada routes dengan nama file web.php. Tambahkan seperti gambar di bawah ini

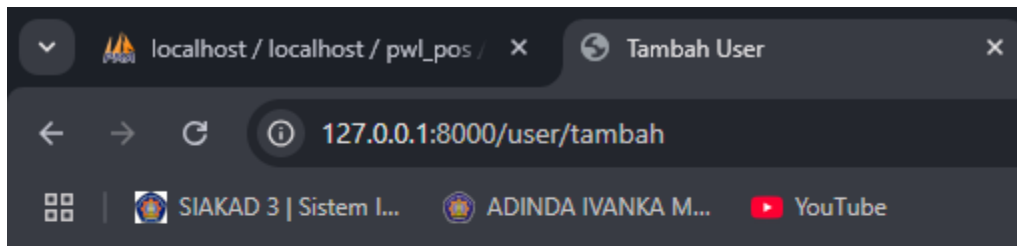
```
Route::post('/user/tambah_simpan', [UserController::class, 'tambah_simpan']);
```

9. Tambahkan script pada controller dengan nama file UserController.php. Tambahkan script dalam class dan buat method baru dengan nama tambah_simpan dan diletakan di bawah method tambah seperti gambar di bawah ini

```
public function tambah_simpan(Request $request)
{
    UserModel::create([
        'username' => $request->username,
        'nama' => $request->nama,
        'password' => Hash::make($request->password),
    ]);
}
```

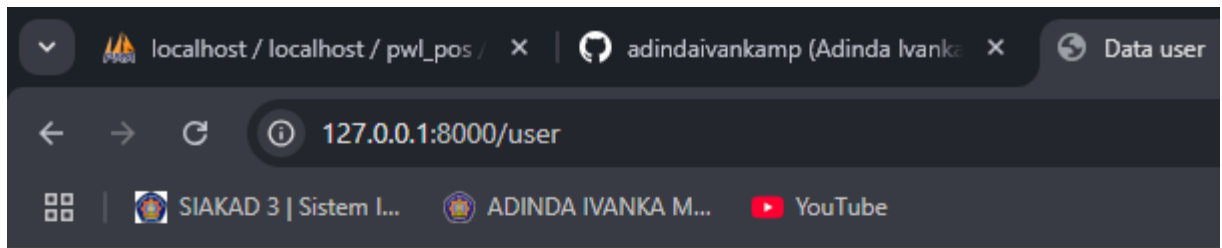
```
'level_id' => $request->level_id  
});  
  
return redirect('/user');  
}
```

10. Simpan kode program Langkah 8 dan 9. Kemudian jalankan link localhost:8000/user/tambah atau localhost/PWL_POS/public/user/tambah pada browser dan input formnya dan simpan, kemudian amati apa yang terjadi dan beri penjelasan dalam laporan



Form Tambah Data User

Username	<input type="text" value="Adin"/>
Nama	<input type="text" value="Adinda"/>
Password	<input type="password" value="....."/>
Level ID	<input type="text" value="4"/>
<input type="button" value="Simpan"/>	



Data User

Tambah User

ID	Username	Nama	ID Level Pengguna	Aksi
1	admin	Administrator	1	Ubah Hapus
2	manager	Manager	2	Ubah Hapus
3	staff	Staff/Kasir	3	Ubah Hapus
4	manager_dua	Manager 2	2	Ubah Hapus
5	manager_tiga	Manager 3	2	Ubah Hapus
6	manager22	Manager Dua Dua	2	Ubah Hapus
7	manager33	Manager Tiga Tiga	2	Ubah Hapus
9	manager56	Manager55	2	Ubah Hapus
10	manager12	Manager11	2	Ubah Hapus
19	Adin	Adinda	4	Ubah Hapus

	user_id	level_id	username	nama	password	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	1	admin	Administrator	\$2y\$12\$clvq8JijITSYq8vVPoTuR.JvkBVGE2PflVNRzRimX...	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	2	manager	Manager	\$2y\$12\$1bMr5lQQSjsJocRO5V1G2uXPGZCOM.cKnTMydXKJF...	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	3	staff	Staff/Kasir	\$2y\$12\$43j76XHZcyYfJ0u1kL4ReTDOfxbVJBvILQGyJLSM...	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	2	manager_dua	Manager 2	\$2y\$12\$SzlU1aCTrKweUpI8YeefruBOObBwIURE0BdMW0utUU...	2025-03-10 03:18:17	2025-03-10 03:18:17
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	2	manager_tiga	Manager 3	\$2y\$12\$RM.jo4ERKFNHPt5p30fpuOilMURqEOpA5kQzEgMxw...	2025-03-10 03:43:22	2025-03-10 03:43:22
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	2	manager22	Manager Dua Dua	\$2y\$12\$VGTfjsPKNd.L2NvM8D8qIOYURTJr6UIHTSBqHxR8IUS...	2025-03-10 10:13:29	2025-03-10 10:13:29
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	2	manager33	Manager Tiga Tiga	\$2y\$12\$E8uQ.kZNTn4Pankrfw5eOV4YkLpWWa7F3FP21YKEI/...	2025-03-10 10:20:25	2025-03-10 10:20:25
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	2	manager56	Manager55	\$2y\$12\$CY8cbBnR3ld5lYg3tCmeuOE5BeJPPvD8szPcSF91ivn...	2025-03-10 10:27:05	2025-03-10 10:27:05
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	2	manager12	Manager11	\$2y\$12\$vk3QVXpZyJ0sz6Z1g1YM0OslMfa.8igDUBDt9VG8JLP...	2025-03-10 10:29:40	2025-03-10 10:29:40
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	19	4	Adin	Adinda	\$2y\$12\$BbK8NkuD8Fp2lRnWUnB9ehb7O6Myt7r7BNg59bh/SZ...	2025-03-10 11:28:59	2025-03-10 11:28:59

11. Langkah berikutnya membuat update atau ubah data user dengan cara bikin file baru pada view dengan nama user_ubah.blade.php dan buat scriptnya menjadi seperti di bawah ini

```
PS C:\laragon\www\Pemrograman-Web-Lanjut\Minggu 4> php artisan make:view user_ubah
```

```
INFO View [C:\laragon\www\Pemrograman-Web-Lanjut\Minggu 4\resources\views\user_ubah.blade.php] created successfully.
```

```
<body>
```

```
<h1>Form Ubah Data User</h1>
```

```

<br><br>
<a href="/user">Kembali</a>
<form method="post" action="/user/ubah_simpan/{{ $data->user_id }}">
    {{ csrf_field() }}
    {{ method_field('PUT') }}
    <label>Username</label>
    <input type="text" name="username" placeholder="Masukan Username" value="{{ $data->username
}}">
    <br>
    <label>Nama</label>
    <input type="text" name="nama" placeholder="Masukan Nama" value="{{ $data->username }}">
    <br>
    <label>Password</label>
    <input type="password" name="password" placeholder="Masukan Password" value="{{ $data-
>password }}">
    <br>
    <label>Level ID</label>
    <input type="number" name="level_id" placeholder="Masukan ID Level" value="{{ $data->level_id
}}">
    <br><br>
    <input type="submit" class="btn btn-success" value="Ubah">
</form>
</body>

```

12. Tambahkan script pada routes dengan nama file web.php. Tambahkan seperti gambar di bawah ini

```
Route::get('/user/ubah/{id}', [UserController::class, 'ubah']);
```

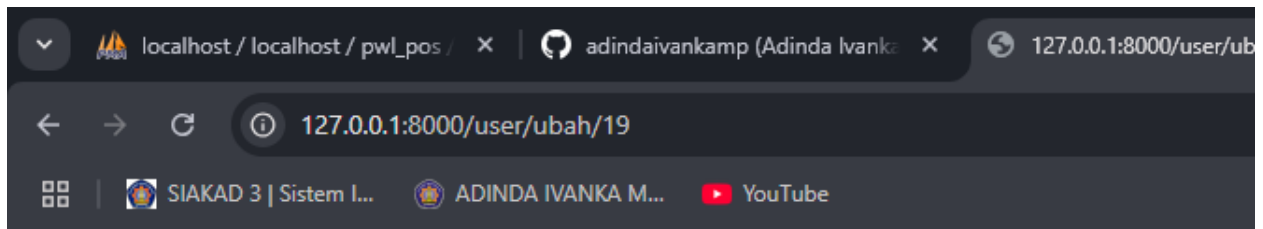
13. Tambahkan script pada controller dengan nama file UserController.php. Tambahkan script dalam class dan buat method baru dengan nama ubah dan diletakan di bawah method tambah_simpan seperti gambar di bawah ini

```

public function ubah($id)
{
    $user = UserModel::find($id);
    return view('user_ubah', ['data' => $user]);
}

```

14. Simpan kode program Langkah 11 sd 13. Kemudian jalankan pada browser dan klik link “Ubah”
amati apa yang terjadi dan beri penjelasan dalam laporan



Form Ubah Data User

[Kembali](#)

Username
Nama
Password
Level ID

15. Tambahkan script pada routes dengan nama file web.php. Tambahkan seperti gambar di bawah ini

```
Route::put('/user/ubah_simpan/{id}', [UserController::class, 'ubah_simpan']);
```

16. Tambahkan script pada controller dengan nama file UserController.php. Tambahkan script dalam class dan buat method baru dengan nama ubah_simpan dan diletakkan di bawah method ubah seperti gambar di bawah ini

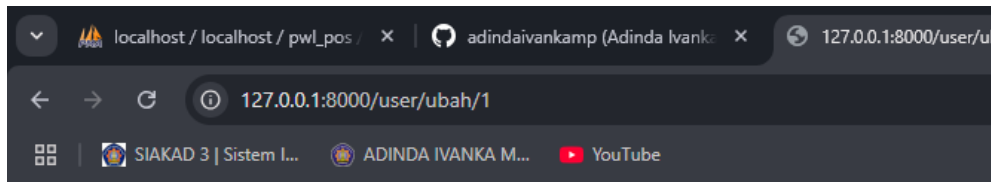
```
public function ubah_simpan($id, Request $request)
{
    $user = UserModel::find($id);

    $user->username = $request->username;
    $user->nama = $request->nama;
    $user->password = Hash::make($request->password);
    $user->level_id = $request->level_id;

    $user->save();

    return redirect('/user');
}
```

17. Simpan kode program Langkah 15 dan 16. Kemudian jalankan link localhost:8000/user/ubah/1 atau localhost/PWL_POS/public/user/ubah/1 pada browser dan ubah input formnya dan klik tombol ubah, kemudian amati apa yang terjadi dan beri penjelasan dalam laporan



Form Ubah Data User

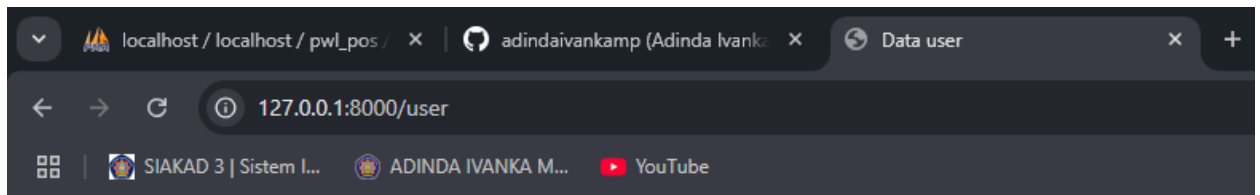
[Kembali](#)

Username

Nama

Password

Level ID



Data User

[Tambah User](#)

ID	Username	Nama	ID Level Pengguna	Aksi
1	Admin	Admin	1	Ubah Hapus
2	manager	Manager	2	Ubah Hapus
3	staff	Staff/Kasir	3	Ubah Hapus
4	manager_dua	Manager 2	2	Ubah Hapus
5	manager_tiga	Manager 3	2	Ubah Hapus
6	manager22	Manager Dua Dua	2	Ubah Hapus
7	manager33	Manager Tiga Tiga	2	Ubah Hapus
9	manager56	Manager55	2	Ubah Hapus
10	manager12	Manager11	2	Ubah Hapus
19	Adin	Adinda	4	Ubah Hapus

18. Berikut untuk langkah delete . Tambahkan script pada routes dengan nama file web.php. Tambahkan seperti gambar di bawah ini

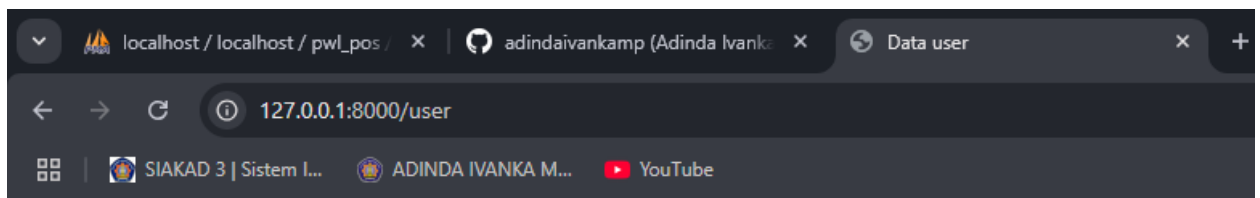
```
Route::get('/user/hapus/{id}', [UserController::class, 'hapus']);
```

19. Tambahkan script pada controller dengan nama file UserController.php. Tambahkan script dalam class dan buat method baru dengan nama hapus dan diletakkan di bawah method ubah_simpan seperti gambar di bawah ini

```
public function hapus($id)
{
    $user = UserModel::find($id);
    $user->delete();

    return redirect('/user');
}
```

20. Simpan kode program Langkah 18 dan 19. Kemudian jalankan pada browser dan klik tombol hapus, kemudian amati apa yang terjadi dan beri penjelasan dalam laporan



Data User

[Tambah User](#)

ID	Username	Nama	ID Level Pengguna	Aksi
1	Admin	Admin	1	Ubah Hapus
2	manager	Manager	2	Ubah Hapus
3	staff	Staff/Kasir	3	Ubah Hapus
4	manager_dua	Manager 2	2	Ubah Hapus
5	manager_tiga	Manager 3	2	Ubah Hapus
6	manager22	Manager Dua Dua	2	Ubah Hapus
7	manager33	Manager Tiga Tiga	2	Ubah Hapus
9	manager56	Manager55	2	Ubah Hapus
10	manager12	Manager11	2	Ubah Hapus

21. Laporkan hasil Praktikum-2.6 ini dan commit perubahan pada git.

Praktikum 2.7 – Relationships

1. Buka file model pada UserModel.php dan tambahkan scripnya menjadi seperti di bawah ini


```

public function level(): BelongsTo
{
    return $this->belongsTo(LevelModel::class, 'level id', 'level id');
}

```

2. Buka file controller pada UserController.php dan ubah method script menjadi seperti di bawah ini

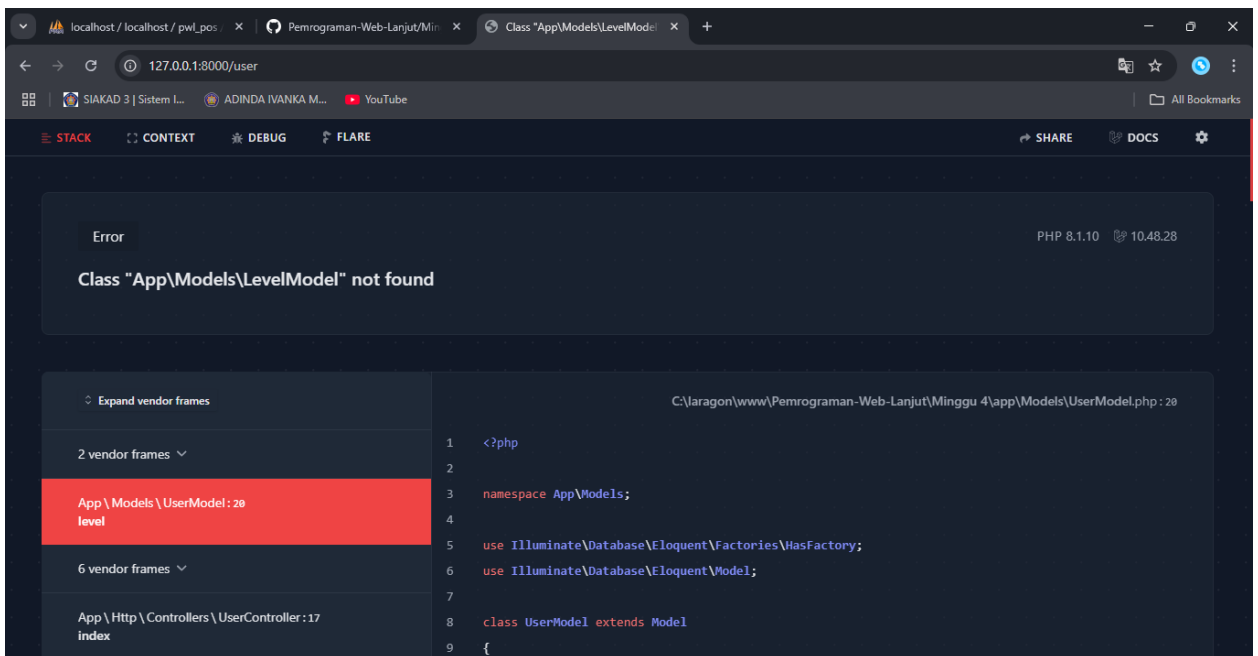
```

class UserController extends Controller
{
    // public function index() {
    //     $user = UserModel::all();
    //     return view('user', ['data' => $user]);
    // }

    public function index(){
        $user = UserModel::with('level')->get();
        dd($user);
    }
}

```

3. Simpan kode program Langkah 2. Kemudian jalankan link pada browser, kemudian amati apa yang terjadi dan beri penjelasan dalam laporan



4. Buka file controller pada UserController.php dan ubah method script menjadi seperti di bawah ini

```

public function index()
{
    $user = UserModel::with('level')->get();
    return view('user', ['data' => $user]);
}

```

```
}
```

5. Buka file view pada user.blade.php dan ubah script menjadi seperti di bawah ini

```
<h1>Data User</h1>

<a href="/user/tambah">Tambah User</a>

<table border="1" cellpadding="2" cellspacing="0">

  <tr>

    <td>ID</td>

    <td>Username</td>

    <td>Nama</td>

    <td>ID Level Pengguna</td>

    <td>Kode Level</td>

    <td>Nama Level</td>

    <td>Aksi</td>

  </tr>

  @foreach ($data as $d)

    <tr>

      <td>{{ $d->user_id }}</td>

      <td>{{ $d->username }}</td>

      <td>{{ $d->nama }}</td>

      <td>{{ $d->level_id }}</td>

      <td>{{ $d->level->level_kode }}</td>

      <td>{{ $d->level->level_nama }}</td>

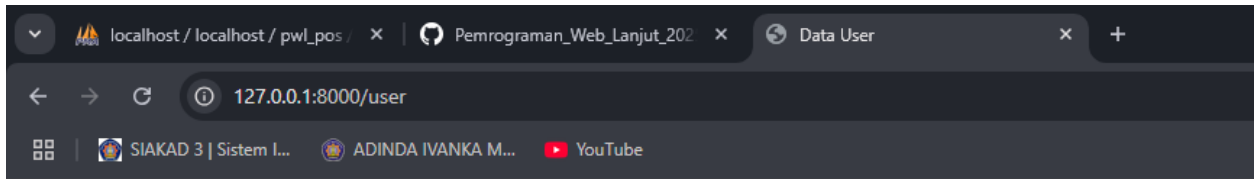
      <td><a href="/user/ubah/{{ $d->user_id }}">Ubah</a> | <a href="/user/hapus/{{ $d->user_id }}">Hapus</a></td>

    </tr>

  @endforeach

</table>
```

6. Simpan kode program Langkah 4 dan 5. Kemudian jalankan link pada browser, kemudian amati apa yang terjadi dan beri penjelasan dalam laporan



Data User

[Tambah User](#)

ID	Username	Nama	ID Level Pengguna	Kode Level	Nama Level	Aksi
1	Admin	Admin	1	ADM	Administrator	Ubah Hapus
2	manager	Manager	2	MNG	Manager	Ubah Hapus
3	staff	Staff/Kasir	3	STF	Staff/Kasir	Ubah Hapus
4	manager_dua	Manager 2	2	MNG	Manager	Ubah Hapus
5	manager_tiga	Manager 3	2	MNG	Manager	Ubah Hapus
6	manager22	Manager Dua Dua	2	MNG	Manager	Ubah Hapus
7	manager33	Manager Tiga Tiga	2	MNG	Manager	Ubah Hapus
9	manager56	Manager55	2	MNG	Manager	Ubah Hapus
10	manager12	Manager11	2	MNG	Manager	Ubah Hapus

Dari hasil pengamatan diatas merupakan proses join pada database atau menggabungkan dua tabel dan ditampilkan ke dalam halaman website. Disini yang perlu diperhatikan adalah relasi antar tabel sehingga pengimplementasian pada program bisa berjalan dan hasil nya bisa ditampilkan seperti pada gambar diatas.

7. Laporkan hasil Praktikum-2.7 ini dan commit perubahan pada git. Jobsheet 04 – PWL 2023/2024