

Wijdan Ardra Fulvian

6706223137

D3IF 46-04

## Praktikum 6

1. Screenshot Struktur tabel users, collections, transactions dan detail\_transactions
  - users

The screenshot shows the MySQL Workbench interface with the 'users' table selected. The 'Structure' tab is active, displaying the following table definition:

#	column_name	data_type	character_set	collation	is_nullable	column_default	extra	foreign_key	comment
1	id	bigint(20) unsigned	utf8mb4	utf8mb4_unicode_ci	NO	NULL	auto_increment	EMPTY	→ EMPTY
2	username	varchar(100)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
3	fullname	varchar(255)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
4	email	varchar(255)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
5	password	varchar(255)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
6	address	varchar(1000)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
7	birthdate	date	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
8	phone_number	varchar(20)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
9	remember_token	varchar(100)	utf8mb4	utf8mb4_unicode_ci	YES	NULL	EMPTY	EMPTY	→ EMPTY
10	created_at	timestamp	utf8mb4	utf8mb4_unicode_ci	YES	NULL	EMPTY	EMPTY	→ EMPTY
11	updated_at	timestamp	utf8mb4	utf8mb4_unicode_ci	YES	NULL	EMPTY	EMPTY	→ EMPTY

Below the table, an index section shows:

index_name	index_algorithm	is_unique	column_name
PRIMARY	BTREE	TRUE	id
users_username_unique	BTREE	TRUE	username

A preview window displays the data for user ID 6706223137:

File	Edit	View
6706223137	*	+ - ×
6706223137 Wijdan Ardra Fulvian 46-04		
Ln 1, Col 38 100% Windows (CRLF) UTF-8		

- collections

The screenshot shows the MySQL Workbench interface with the 'collections' table selected. The 'Structure' tab is active, displaying the following table definition:

#	column_name	data_type	character_set	collation	is_nullable	column_default	extra	foreign_key	comment
1	id	int(11)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	auto_increment	EMPTY	→ EMPTY
2	nama	varchar(100)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
3	jenis	tinyint(4)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
4	created_at	datetime	utf8mb4	utf8mb4_unicode_ci	NO	NULL	current_timestamp	EMPTY	→ EMPTY
5	jumlahAwal	int(11)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
6	jumlahSisa	int(11)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY
7	jumlahKeluar	int(11)	utf8mb4	utf8mb4_unicode_ci	NO	NULL	EMPTY	EMPTY	→ EMPTY

Below the table, an index section shows:

index_name	index_algorithm	is_unique	column_name
PRIMARY	BTREE	TRUE	id

A preview window displays the data for user ID 6706223137:

File	Edit	View
6706223137	*	+ - ×
6706223137 Wijdan Ardra Fulvian 46-04		
Ln 1, Col 38 100% Windows (CRLF) UTF-8		

- transactions

The screenshot shows the MySQL Workbench interface with the 'transactions' table selected. The table has 7 columns: id, userIdPetugas, userIdPeminjam, tanggalPinjam, tanggalSelesai, created\_at, and updated\_at. Primary key is 'id'. Foreign keys: 'userIdPetugas' references 'users(id)', 'userIdPeminjam' references 'users(id)'. Indexes: PRIMARY (id), transactions\_useridpetugas\_foreign (BTREE on userIdPetugas), transactions\_useridpeminjam\_foreign (BTREE on userIdPeminjam). A data editor window is open, showing a single row with values: 670622313, Wijdan Ardra Fulvian, 46-04.

#	column_name	data_type	character_set	collation	is_nullable	column_default	extra	foreign_key	comment
1	id	int(11)		NULL	NO	NULL	auto_increment	EMPTY	→ EMPTY
2	userIdPetugas	bigint(20) unsigned		NULL	NO	NULL	EMPTY	users(id)	→ EMPTY
3	userIdPeminjam	bigint(20) unsigned		NULL	NO	NULL	EMPTY	users(id)	→ EMPTY
4	tanggalPinjam	date		NULL	NO	NULL	EMPTY	EMPTY	→ EMPTY
5	tanggalSelesai	date		NULL	YES	NULL	EMPTY	EMPTY	→ EMPTY
6	created_at	datetime		NULL	YES	NULL	EMPTY	EMPTY	→ EMPTY
7	updated_at	datetime		NULL	YES	NULL	EMPTY	EMPTY	→ EMPTY

index_name	index_algorithm	is_unique	column_name
PRIMARY	BTREE	TRUE	id
transactions_useridpetugas_foreign	BTREE	FALSE	userIdPetugas
transactions_useridpeminjam_foreign	BTREE	FALSE	userIdPeminjam

- detail\_transactions

The screenshot shows the MySQL Workbench interface with the 'detail\_transactions' table selected. The table has 8 columns: id, transactionId, collectionId, tanggalKembali, status, keterangan, created\_at, and updated\_at. Primary key is 'id'. Foreign keys: detail\_transactions\_transactionid\_foreign (BTREE on transactionId), detail\_transactions\_collectionid\_foreign (BTREE on collectionId). Indexes: PRIMARY (id), detail\_transactions\_transactionid\_foreign (BTREE on transactionId). A data editor window is open, showing a single row with values: 670622313, Wijdan Ardra Fulvian, 46-04.

#	column_name	data_type	character_set	collation	is_nullable	column_default	extra	foreign_key	comment
1	id	int(11)		NULL	NO	NULL	auto_increment	EMPTY	→ EMPTY
2	transactionId	int(11)		NULL	NO	NULL	EMPTY	transactions(id)	→ EMPTY
3	collectionId	int(11)		NULL	NO	NULL	EMPTY	collections(id)	→ EMPTY
4	tanggalKembali	date		NULL	YES	NULL	EMPTY	EMPTY	→ EMPTY
5	status	tinyint(4)		NULL	NO	NULL	EMPTY	EMPTY	→ EMPTY
6	keterangan	varchar(1000)		utf8mb4	utf8mb4_unicode_ci	YES	NULL	EMPTY	→ EMPTY
7	created_at	datetime		NULL	NO	current_timestamp	EMPTY	EMPTY	→ EMPTY
8	updated_at	datetime		NULL	NO	current_timestamp	EMPTY	EMPTY	→ EMPTY

index_name	index_algorithm	is_unique	column_name
PRIMARY	BTREE	TRUE	id
detail_transactions_transactionid_foreign	BTREE	FALSE	transactionId
detail_transactions_collectionid_foreign	BTREE	FALSE	collectionId

## 2. File kelas routes/web.php

```

routes > web.php > ...
1  <?php
2
3 // 6706223137 Wijdan Ardra Fulvian 46-04
4
5 use Illuminate\Support\Facades\Route;
6 use App\Http\Controllers\UserController;
7 use App\Http\Controllers\CollectionController;
8 use App\Http\Controllers\TransactionController;
9 use App\Http\Controllers\DetailTransactionController;
10
11 /*
12 | -----
13 | Web Routes
14 | -----
15 |
16 | Here is where you can register web routes for your application. These
17 | routes are loaded by the RouteServiceProvider within a group which
18 | contains the "web" middleware group. Now create something great!
19 |
20 */
21
22 Route::get('/', function () {
23     return view('welcome');
24 });
25
26 Route::middleware(['auth', 'verified'])->group(function() {
27     // Routing ke dashboard
28     Route::get('/dashboard', function () {
29         return view('dashboard');
30     })->name('dashboard');
31
32     // Routing ke User Controller
33     Route::get('/user', [UserController::class, 'index'])->name('showUser');
34     Route::get('/userRegistration', [UserController::class, 'create'])->name('registerUser');
35     Route::post('/userStore', [UserController::class, 'store']);
36     Route::get('/userView/{user}', [UserController::class, 'show']);
37     Route::post('/userUpdate', [UserController::class, 'update']);
38
39     Route::get('/getAllUsers', [UserController::class, 'getAllUsers']);
40
41     // Routing ke CollectionController
42     Route::get('/koleksi', [CollectionController::class, 'index'])->name('showKoleksi');
43     Route::get('/koleksiTambah', [CollectionController::class, 'create'])->name('tambahKoleksi');
44     Route::post('/koleksiStore', [CollectionController::class, 'store']);
45     Route::get('/koleksiView/{collection}', [CollectionController::class, 'show']);
46     Route::post('/koleksiUpdate', [CollectionController::class, 'update']);
47
48     Route::get('/getAllCollections', [CollectionController::class, 'getAllCollections']);
49
50     // Routing ke TransactionController
51     Route::get('/transaksi', [TransactionController::class, 'index'])->name('transaksi');
52     Route::get('/transaksiTambah', [TransactionController::class, 'create'])->name('tambahTransaksi');
53     Route::post('/transaksiStore', [TransactionController::class, 'store']);
54     Route::get('/transaksiView/{transaction}', [TransactionController::class, 'show']);
55
56     Route::get('/getAllTransactions', [TransactionController::class, 'getAllTransactions']);
57
58     // Routing ke DetailTransactionController
59     Route::get('/detailTransactionKembalikan/{detailTransactionId}', [DetailTransactionController::class, 'detailTransactionKembalikan']);
60     Route::post('/detailTransactionUpdate', [DetailTransactionController::class, 'update']);
61
62     Route::get('/getAllDetailTransactions/{transactionId}', [DetailTransactionController::class, 'getAllDetailTransactions']);
63
64 });
65
66 });
67
68 require __DIR__ . '/auth.php';

```

### 3. Kelas controller

- UserController,

**Code Editor Screenshot 1:**

```

    <?php
    // 6706223137 Wijdan Andra Fulvian 46-04
    namespace App\Http\Controllers;

    use Illuminate\Http\Request;
    use App\Models\User;
    use Illuminate\Support\Facades\Hash;
    use Illuminate\Validation\Rules\Password;
    use Yajra\DataTables\Facades\DataTables;

    class UserController extends Controller
    {
        public function index()
        {
            return view('user.daftarPengguna');
        }

        public function create()
        {
            return view('user.registrasi');
        }

        public function store(Request $request)
        {
            $request->validate([
                'username' => ['required', 'string', 'max:255', 'unique:users'],
                'fullname' => ['required', 'string', 'max:255'],
                'email' => ['email'],
                'password' => ['required', 'confirmed', Password::defaults()],
                'address' => ['required', 'string'],
                'birthdate' => ['required', 'date', 'before:today'],
                'phoneNumber' => ['required']
            ]);

            $user = User::create([
                'username' => $request->username,
                'fullname' => $request->fullname,
                'email' => $request->email,
                'password' => Hash::make($request->password),
                'address' => $request->address,
                'birthdate' => $request->birthdate,
                'phoneNumber' => $request->phoneNumber,
            ]);

            return view('user.daftarPengguna');
        }

        public function show(User $user)
        {
            return view('user.infoPengguna', [
                'data' => compact(['user'])
            ]);
        }

        public function getAllUsers()
        {
            $users = DB::table('users')
                ->select(
                    'id as id',
                    'fullname as fullname',
                    'address as address',
                    'birthdate as birthdate',
                    'phoneNumber as phoneNumber'
                )
                ->orderBy('id', 'asc')
                ->get();
        }
    }

```

**Code Editor Screenshot 2:**

```

    <?php
    // 6706223137 Wijdan Andra Fulvian 46-04
    namespace App\Http\Controllers;

    use Illuminate\Http\Request;
    use App\Models\User;
    use Illuminate\Support\Facades\Hash;
    use Illuminate\Validation\Rules\Password;
    use Yajra\DataTables\Facades\DataTables;

    class UserController extends Controller
    {
        public function index()
        {
            return view('user.daftarPengguna');
        }

        public function create()
        {
            return view('user.registrasi');
        }

        public function store(Request $request)
        {
            $request->validate([
                'username.required' => 'Username harus diisi',
                'username.unique' => 'Username telah digunakan',
                'birthdate.before' => 'Tanggal lahir harus sebelum hari ini'
            ]);

            $user = User::create([
                'username' => $request->username,
                'fullname' => $request->fullname,
                'email' => $request->email,
                'password' => Hash::make($request->password),
                'address' => $request->address,
                'birthdate' => $request->birthdate,
                'phoneNumber' => $request->phoneNumber,
            ]);

            return view('user.daftarPengguna');
        }

        public function show(User $user)
        {
            return view('user.infoPengguna', [
                'data' => compact(['user'])
            ]);
        }

        public function getAllUsers()
        {
            $users = DB::table('users')
                ->select(
                    'id as id',
                    'fullname as fullname',
                    'address as address',
                    'birthdate as birthdate',
                    'phoneNumber as phoneNumber'
                )
                ->orderBy('id', 'asc')
                ->get();
        }
    }

```

The screenshot shows a code editor interface with the following details:

- File Path:** app > Http > Controllers > UserController.php
- Code Content (UserController.php):**

```
69     return DataTables::of($users)
70     ->addColumn('action', 'content: function($user) {
71         $html = '
72             <a class="btn btn-info" href="/userView/'.$user->id.'">Edit</a>
73             ';
74         return $html;
75     })
76     ->make(['dataSupport' true]);
77 }
78
79 // reference|0 overrides
80 public function update(Request $request)
81 {
82     $request->validate([
83         'fullname' => ['required', 'string', 'max:255'],
84         'password' => ['required'],
85         'address' => ['required', 'string'],
86         'phoneNumber' => ['required'],
87     ]);
88
89     $affected = DB::table('users')
90     ->where('column', 'id', 'operator', $request->id)
91     ->update(['values' [
92         'fullname' => $request->fullname,
93         'password' => Hash::make($value: $request->password),
94         'address' => $request->address,
95         'phoneNumber' => $request->phoneNumber,
96     ]]);
97
98     return view('view: user.daftarPengguna');
99 }
100 }
```

- Editor Status Bar:** Ln 4, Col 1 | Spaces: 4 | UTF-8 | LF | PHP 8.1 | Prettier

- CollectionController,

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz

EXPLORER ... web.php M UserController.php M CollectionController.php M

```

app > Http > Controllers > CollectionController.php > ...
1 <?php
2
3 // 6706223137 Wijdan Ardra Fulvian 46-04
4
5 namespace App\Http\Controllers;
6
7 use Illuminate\Http\Request;
8 use App\Models\Collection;
9 use Illuminate\Support\Facades\DB;
10 use Yajra\DataTables\Facades\DataTables;
11
12 class CollectionController extends Controller
13 {
14     /**
15      * Create a new controller instance.
16      */
17     public function index()
18     {
19         return view('view::koleksi.daftarkoleksi');
20     }
21
22     /**
23      * Show the form for creating a new collection.
24      */
25     public function show(Collection $collection)
26     {
27         return view('view::koleksi.infokoleksi', [
28             'data' => compact('var_name', 'collection')
29         ]);
30     }
31
32     /**
33      * Get all collections.
34      */
35     public function getAllCollections()
36     {
37         $collections = DB::table('collections')
38             ->select([
39                 'id' as id,
40                 'nama' as judul,
41                 DB::raw("CASE
42                     WHEN jenis='1' THEN 'Buku'
43                     WHEN jenis='2' THEN 'Majalah'
44                     WHEN jenis='3' THEN 'Cakram Digital'
45                     END) AS jenis
46             ],
47             'jumlahSisa' as jumlahSisa,
48             'jumlahAwal' as jumlahAwal,
49             'jumlahKeluarn' as jumlahKeluarn")
50             ->orderBy('nama', 'asc')
51             ->get();
52     }
53
54     /**
55      * Store a new collection.
56      */
57     public function store(Request $request)
58     {
59         $request->validate([
60             'nama' => ['required', 'string', 'max:255', 'unique:collections'],
61             'jenis' => ['required', 'gt:0'],
62             'jumlahAwal' => ['required', 'gt:0']
63         ],
64         [
65             'params[0]' => [
66                 'nama.unique' => 'Nama koleksi tersebut sudah ada'
67             ]
68         ]);
69
70         $koleksi = [
71             'nama' => $request->nama,
72             'jenis' => $request->jenis,
73             'jumlahAwal' => $request->jumlahAwal,
74             'jumlahSisa' => $request->jumlahSisa
75         ];
76
77         $collection = new Collection();
78         $collection->fill($koleksi);
79         $collection->save();
80
81         return DataTables::of($collections)
82             ->addColumn('action', 'content' => function($collection) {
83                 $html = '
84                     <a href="'.url("koleksiView")."/{$collection->id}"/>Edit</a>
85                 ';
86                 return $html;
87             })
88             ->make([
89                 'dataSupport' => true
90             ]);
91     }
92
93     /**
94      * Update the specified collection.
95      */
96     public function update(Request $request, Collection $collection)
97     {
98         $request->validate([
99             'nama' => ['required', 'string', 'max:255', 'unique:collections'],
100            'jenis' => ['required', 'gt:0'],
101            'jumlahAwal' => ['required', 'gt:0']
102        ],
103        [
104            'params[0]' => [
105                'nama.unique' => 'Nama koleksi tersebut sudah ada'
106            ]
107        ]);
108
109        $collection->fill($request->all());
110        $collection->save();
111
112        return redirect()->back();
113    }
114
115    /**
116     * Remove the specified collection.
117     */
118    public function destroy(Collection $collection)
119    {
120        $collection->delete();
121
122        return redirect()->back();
123    }
124}

```

Ln 3, Col 41 Spaces: 4 UTF-8 LF ↵ PHP 8.1 ⚡ Prettier

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz

EXPLORER ... web.php M UserController.php M CollectionController.php M

```

app > Http > Controllers > CollectionController.php > ...
1 <?php
2
3 // 6706223137 Wijdan Ardra Fulvian 46-04
4
5 namespace App\Http\Controllers;
6
7 use Illuminate\Http\Request;
8 use App\Models\Collection;
9 use Illuminate\Support\Facades\DB;
10 use Yajra\DataTables\Facades\DataTables;
11
12 class CollectionController extends Controller
13 {
14     /**
15      * Create a new controller instance.
16      */
17     public function index()
18     {
19         return view('view::koleksi.daftarkoleksi');
20     }
21
22     /**
23      * Show the form for creating a new collection.
24      */
25     public function show(Collection $collection)
26     {
27         return view('view::koleksi.infokoleksi', [
28             'data' => compact('var_name', 'collection')
29         ]);
30     }
31
32     /**
33      * Get all collections.
34      */
35     public function getAllCollections()
36     {
37         $collections = DB::table('collections')
38             ->select([
39                 'id' as id,
40                 'nama' as judul,
41                 DB::raw("CASE
42                     WHEN jenis='1' THEN 'Buku'
43                     WHEN jenis='2' THEN 'Majalah'
44                     WHEN jenis='3' THEN 'Cakram Digital'
45                     END) AS jenis
46             ],
47             'jumlahSisa' as jumlahSisa,
48             'jumlahAwal' as jumlahAwal,
49             'jumlahKeluarn' as jumlahKeluarn")
50             ->orderBy('nama', 'asc')
51             ->get();
52     }
53
54     /**
55      * Store a new collection.
56      */
57     public function store(Request $request)
58     {
59         $request->validate([
60             'nama' => ['required', 'string', 'max:255', 'unique:collections'],
61             'jenis' => ['required', 'gt:0'],
62             'jumlahAwal' => ['required', 'gt:0']
63         ],
64         [
65             'params[0]' => [
66                 'nama.unique' => 'Nama koleksi tersebut sudah ada'
67             ]
68         ]);
69
70         $koleksi = [
71             'nama' => $request->nama,
72             'jenis' => $request->jenis,
73             'jumlahAwal' => $request->jumlahAwal,
74             'jumlahSisa' => $request->jumlahSisa
75         ];
76
77         $collection = new Collection();
78         $collection->fill($koleksi);
79         $collection->save();
80
81         return DataTables::of($collections)
82             ->addColumn('action', 'content' => function($collection) {
83                 $html = '
84                     <a href="'.url("koleksiView")."/{$collection->id}"/>Edit</a>
85                 ';
86                 return $html;
87             })
88             ->make([
89                 'dataSupport' => true
90             ]);
91     }
92
93     /**
94      * Update the specified collection.
95      */
96     public function update(Request $request, Collection $collection)
97     {
98         $request->validate([
99             'nama' => ['required', 'string', 'max:255', 'unique:collections'],
100            'jenis' => ['required', 'gt:0'],
101            'jumlahAwal' => ['required', 'gt:0']
102        ],
103        [
104            'params[0]' => [
105                'nama.unique' => 'Nama koleksi tersebut sudah ada'
106            ]
107        ]);
108
109        $collection->fill($request->all());
110        $collection->save();
111
112        return redirect()->back();
113    }
114
115    /**
116     * Remove the specified collection.
117     */
118    public function destroy(Collection $collection)
119    {
120        $collection->delete();
121
122        return redirect()->back();
123    }
124}

```

Ln 3, Col 41 Spaces: 4 UTF-8 LF ↵ PHP 8.1 ⚡ Prettier

The screenshot shows a code editor interface with a dark theme. The left sidebar is titled 'EXPLORER' and lists several files and folders under 'WIZZ'. The main editor area displays the 'CollectionController.php' file. The code is a PHP script with syntax highlighting for variables, functions, and comments. It includes database queries using Eloquent ORM and validation rules for a 'collections' table.

```
    app > Http > Controllers > CollectionController.php > ...
    69     'jumlahSisa' => $request->jumlahAwal,
    70     'jumlahKeluar' => 0,
    71   ];
    72
    73   DB::table('collections')->insert(['values' => $oleksi]);
    74   return view('view' => 'koleksi.daftarkoleksi');
    75 }
    76
    1 reference | 0 overrides
  77 public function update(Request $request)
  78 {
  79     $request->validate([
  80         'jenis' => ['required', 'gt:0'],
  81         'jumlahSisa' => ['required', 'gt:0'],
  82         'jumlahKeluar' => ['required', 'gt:0'],
  83     ]);
  84
  85     $affected = DB::table('collections')
  86     ->where('column' => 'id', 'operator' => $request->id)
  87     ->update([
  88         'jenis' => $request->jenis,
  89         'jumlahSisa' => $request->jumlahSisa,
  90         'jumlahKeluar' => $request->jumlahKeluar,
  91     ]);
  92
  93     return view('view' => 'koleksi.daftarkoleksi');
  94
  95 }
  96 }
```

- TransactionController,

**Wizz**

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz

EXPLORER

- WIZZ
  - app
    - Console
    - DataTables
    - Exceptions
    - Http
      - Controllers
        - Auth
        - CollectionCo... M
        - Controller.php M
        - DetailTransac... U
        - ProfileController.php...
        - TransactionController.php U
    - Middleware
    - Requests
    - Kernel.php M
  - Models
  - Policies
  - Providers
  - View
    - bootstrap
    - config
    - database
    - lang
    - node\_modules
    - public
    - resources
    - routes
      - api.php M
      - auth.php M
- OUTLINE
- TIMELINE

web.php M UserController.php M CollectionController.php M TransactionController.php U

```

app > Http > Controllers > TransactionController.php > ...
1 <?php
2
3 // 6706223137 Wijdan Andra Fulvian 46-04
4
5 namespace App\Http\Controllers;
6
7 use Carbon\Carbon;
8 use App\Models\User;
9 use App\Models\Collection;
10 use App\Models\DetailTransaction;
11 use App\Models\Transaction;
12 use Illuminate\Http\Request;
13 use Illuminate\Support\Facades\DB;
14 use yajra\_datatables\Facades\datatables;
15
16 6 references | 0 implementations
17 class TransactionController extends Controller
18 {
19     1 reference | 0 overrides
20     public function index()
21     {
22         return view('views.transaction.daftarTransaksi');
23     }
24
25     1 reference | 0 overrides
26     public function create()
27     {
28         $users = User::get();
29         $collections = Collection::where('jumlahSisa', '>', 0)->get();
30         return view('views.transaction.transaksiTambah', [
31             'data' => compact(['collections', 'users'])
32         ]);
33     }
34
35     1 reference | 0 overrides
36     public function show(Transaction $transaction)
37     {
38         $transactions = DB::table('transactions')
39             ->select([
40                 'transactions.id as id',
41                 'u1.fullname as fullnamePeminjam',
42                 'u2.fullname as fullnamePetugas',
43                 'tanggaldiPinjam as tanggaldiPinjam',
44             ])
45
46     1 reference | 0 overrides
47     public function getAllTransactions()
48     {
49         $transactions = DB::table('transactions')
50             ->select([
51                 'transactions.id as id',
52                 'u1.fullname as peminjam',
53                 'u2.fullname as petugas',
54                 'tanggaldiPinjam as tanggaldiPinjam',
55                 'tanggalsSelesai as tanggalsSelesai',
56             ])
57
58         ->join('users as u1', 'first: userIdPeminjam', 'operator: '=', 'second: u1.id')
59         ->join('users as u2', 'first: userIdPetugas', 'operator: '=', 'second: u2.id')
60         ->where('column: transactions.id', 'operator: >', 'value: $transaction->id')
61         ->orderBy('column: tanggaldiPinjam', 'direction: asc')
62         ->first();
63
64         return view('views.transaction.transaksiView', [
65             'data' => compact(['transactions'])
66         ]);
67     }
68
69     1 reference | 0 overrides
70     public function store(Request $request)
71     {
72         $transaction = new Transaction();
73         $transaction->userIdPeminjam = $request->userIdPeminjam;
74         $transaction->userIdPetugas = $request->userIdPetugas;
75         $transaction->tanggaldiPinjam = Carbon::now();
76         $transaction->tanggalsSelesai = Carbon::parse($request->tanggalsSelesai);
77         $transaction->jumlahSisa = $request->jumlahSisa;
78         $transaction->save();
79
80         return redirect('transaction/transaksiView');
81     }
82
83     1 reference | 0 overrides
84     public function edit(Transaction $transaction)
85     {
86         $html = '';
87         $html .= 'Edit';
88         return $html;
89     }
90
91     1 reference | 0 overrides
92     public function update(Request $request, Transaction $transaction)
93     {
94         $transaction->userIdPeminjam = $request->userIdPeminjam;
95         $transaction->userIdPetugas = $request->userIdPetugas;
96         $transaction->tanggaldiPinjam = Carbon::now();
97         $transaction->tanggalsSelesai = Carbon::parse($request->tanggalsSelesai);
98         $transaction->jumlahSisa = $request->jumlahSisa;
99         $transaction->update();
100    }
101
102    1 reference | 0 overrides
103    public function destroy(Transaction $transaction)
104    {
105        $transaction->delete();
106    }
107}

```

Ln 3, Col 41 Spaces: 4 UTF-8 LF ↵ PHP 8.1 ⚡ Prettier

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz

EXPLORER

- WIZZ
  - app
    - Console
    - DataTables
    - Exceptions
    - Http
      - Controllers
        - Auth
        - CollectionCo... M
        - Controller.php M
        - DetailTransac... U
        - ProfileController.php...
        - TransactionController.php U
    - Middleware
    - Requests
    - Kernel.php M
  - Models
  - Policies
  - Providers
  - View
    - bootstrap
    - config
    - database
    - lang
    - node\_modules
    - public
    - resources
    - routes
      - api.php M
      - auth.php M
- OUTLINE
- TIMELINE

web.php M UserController.php M CollectionController.php M TransactionController.php U

```

app > Http > Controllers > TransactionController.php > ...
34
35     'tanggaldiPinjam as tanggaldiPinjam',
36     'tanggalsSelesai as tanggalsSelesai',
37 )
38 ->join('table: users as u1', 'first: 'userIdPeminjam', 'operator: '=', 'second: u1.id')
39 ->join('table: users as u2', 'first: 'userIdPetugas', 'operator: '=', 'second: u2.id')
40 ->where('column: transactions.id', 'operator: >', 'value: $transaction->id')
41 ->orderBy('column: tanggaldiPinjam', 'direction: asc')
42 ->first();
43
44     return view('views.transaction.transaksiView', [
45         'data' => compact(['transactions'])
46     ]);
47
48     1 reference | 0 overrides
49     public function getAllTransactions()
50     {
51         $transactions = DB::table('transactions')
52             ->select([
53                 'transactions.id as id',
54                 'u1.fullname as peminjam',
55                 'u2.fullname as petugas',
56                 'tanggaldiPinjam as tanggaldiPinjam',
57                 'tanggalsSelesai as tanggalsSelesai',
58             ])
59
60         ->join('users as u1', 'first: userIdPeminjam', 'operator: '=', 'second: u1.id')
61         ->join('users as u2', 'first: userIdPetugas', 'operator: '=', 'second: u2.id')
62         ->orderBy('column: tanggaldiPinjam', 'direction: asc')
63         ->get();
64
65         return DataTables::of($transactions)
66             ->addColumn('action', [
67                 'content' => function($transaction) {
68                     $html = '';
69                     $html .= 'Edit';
70                     return $html;
71                 }
72             ])
73             ->make(['dataSupport' => true]);
74
75     }
76
77     1 reference | 0 overrides
78     public function store(Request $request)
79     {
80         $transaction = new Transaction();
81         $transaction->userIdPeminjam = $request->userIdPeminjam;
82         $transaction->userIdPetugas = $request->userIdPetugas;
83         $transaction->tanggaldiPinjam = Carbon::now();
84         $transaction->tanggalsSelesai = Carbon::parse($request->tanggalsSelesai);
85         $transaction->jumlahSisa = $request->jumlahSisa;
86         $transaction->save();
87
88         return redirect('transaction/transaksiView');
89     }
90
91     1 reference | 0 overrides
92     public function edit(Transaction $transaction)
93     {
94         $html = '';
95         $html .= 'Edit';
96         return $html;
97     }
98
99     1 reference | 0 overrides
100    public function update(Request $request, Transaction $transaction)
101    {
102        $transaction->userIdPeminjam = $request->userIdPeminjam;
103        $transaction->userIdPetugas = $request->userIdPetugas;
104        $transaction->tanggaldiPinjam = Carbon::now();
105        $transaction->tanggalsSelesai = Carbon::parse($request->tanggalsSelesai);
106        $transaction->jumlahSisa = $request->jumlahSisa;
107        $transaction->update();
108    }
109
110    1 reference | 0 overrides
111    public function destroy(Transaction $transaction)
112    {
113        $transaction->delete();
114    }
115}

```

Ln 3, Col 41 Spaces: 4 UTF-8 LF ↵ PHP 8.1 ⚡ Prettier

Wizz

```

File Edit Selection View Go Run Terminal Help ← → ○ Wizz
EXPLORER ... web.php M UserController.php M CollectionController.php M TransactionController.php U X
app > Http > Controllers > TransactionController.php > ...
public function store(Request $request)
{
    $request->validate([
        'idPeminjam' => ['required', 'integer', 'gt:0'],
        'koleksi1' => ['required', 'integer', 'gt:0']
    ],
    $request->validated(['rules: [
        'idPeminjam' => 'Pilih satu Species',
        'koleksi1' => 'Pilih jenis item'
    ]);
    // membuat 1 object transaction dan simpan kedalam table transactions
    $transaction = new Transaction;
    $transaction->userIdPeminjam = $request->idPeminjam;
    $transaction->userIdPetugas = auth()->user()->id;
    $transaction->tanggalPinjam = Carbon::now()->toDateString();
    $transaction->save();
    // mengambil last transaction id untuk digunakan
    // pada proses insert detail transaction
    $lastTransactionIdStored = $transaction->id;
    // membuat object detail transaction dan simpan kedalam table detail_transactions
    $detailTransaksi1 = new DetailTransaction;
    $detailTransaksi1->transactionId = $lastTransactionIdStored;
    $detailTransaksi1->collectionId = $request->koleksi1;
    $detailTransaksi1->status = 1;
    $detailTransaksi1->save();
    // Mengurangi jumlah stok
    DB::table('collections')->where('id', '=', $request->koleksi1)->decrement('jumlahSisa');
    DB::table('collections')->where('id', '=', $request->koleksi1)->increment('jumlahKeluar');

    // Peminjaman koleksi 2
    if($request->koleksi2 > 0) {
        $detailTransaksi2 = new DetailTransaction;
        $detailTransaksi2->transactionId = $lastTransactionIdStored;
        ...
    }
}

```

Ln 3, Col 41 Spaces: 4 UTF-8 LF ↵ PHP 8.1 ⚡ Prettier

Wizz

```

File Edit Selection View Go Run Terminal Help ← → ○ Wizz
EXPLORER ... web.php M UserController.php M CollectionController.php M TransactionController.php U X
app > Http > Controllers > TransactionController.php > ...
public function store(Request $request)
{
    $detailTransaksi2->collectionId = $request->koleksi2;
    $detailTransaksi2->status = 1;
    $detailTransaksi2->save();
    // Mengurangi jumlah stok
    DB::table('collections')->where('id', '=', $request->koleksi2)->decrement('jumlahSisa');
    DB::table('collections')->where('id', '=', $request->koleksi2)->increment('jumlahKeluar');

    // Peminjaman koleksi 3
    if($request->koleksi3 > 0) {
        $detailTransaksi3 = new DetailTransaction;
        $detailTransaksi3->transactionId = $lastTransactionIdStored;
        $detailTransaksi3->collectionId = $request->koleksi3;
        $detailTransaksi3->status = 1;
        $detailTransaksi3->save();
        // Mengurangi jumlah stok
        DB::table('collections')->where('id', '=', $request->koleksi3)->decrement('jumlahSisa');
        DB::table('collections')->where('id', '=', $request->koleksi3)->increment('jumlahKeluar');
    }
}
return redirect()->route('transaksi')->with(['key: 'status', value: 'Peminjaman berhasil']);
}

references(0 overrides)
public function update(Request $request)
{
    $request->validate([
        'jenis' => ['required', 'gt:0'],
        'jumlahSisa' => ['required', 'gt:0'],
        'jumlahKeluar' => ['required', 'gt:0']
    ]);

    $affected = DB::table('transactions')
    ->where('id', '=', $request->id)
    ->update(['values: [
        'jenis' => $request->jenis,
        'jumlahSisa' => $request->jumlahSisa,
        'jumlahKeluar' => $request->jumlahKeluar,
    ]]);
}

return view('view: 'koleksi.daftarKoleksi');

}

```

Ln 129, Col 6 Spaces: 4 UTF-8 LF ↵ PHP 8.1 ⚡ Prettier

- DetailTransactionController

The image displays three vertically stacked screenshots of a code editor interface, likely PhpStorm, showing PHP code for a Laravel application named 'Wizz'.

**Top Editor:** Shows the file `DetailTransactionController.php`. The code defines a class `DetailTransactionController` extending `Controller`. It contains a method `getAllDetailTransactions` which performs a database query to select detail transactions based on their status ('1', '2', or '3'). The results are then processed by a `DataTables` facade.

```

class DetailTransactionController extends Controller
{
    public function getAllDetailTransactions($transactionId)
    {
        $detail_transactions = DB::table('detail_transactions as dt')
            ->select([
                'dt.id',
                'dt.tanggalKembali as tanggalKembali',
                'dt.tanggalPinjam as tanggalPinjam',
                'dt.status as statusType',
                DB::raw("value: (CASE WHEN dt.status='1' THEN 'Pinjam'
                    WHEN dt.status='2' THEN 'Kembalikan'
                    WHEN dt.status='3' THEN 'Hilang'
                    END) as status"),
                'c.nama as koleksi'
            ])
            ->join('collections as c', 'c.id', '=', 'dt.collectionId')
            ->join('transactions as t', 't.id', '=', 'dt.transactionId')
            ->where('c.transactionId', '=', $transactionId)
            ->get();
    }
}

```

**Middle Editor:** Shows the file `DetailTransactionController.php`. This version includes additional logic for handling transaction details. It includes methods for generating edit links for transactions with status '1', making the controller support data, and updating a transaction record.

```

if ($detail_transaction->statusType == "1") {
    $html = '<a class="btn btn-info" href="'.url(path: "detailTransactionKembalikan").'"/'.$detail_transaction->id.'>Edit</a>';
}
return $html;
}

public function detailTransactionKembalikan($detailTransactionId)
{
    $detailTransaction = DB::table('detail_transactions as dt')
        ->select([
            'dt.id as idTransaksi',
            'dt.id as id',
            'dt.tanggalKembali as tanggalKembali',
            'dt.tanggalPinjam as tanggalPinjam',
            'dt.status',
            'uPinjam.fullname as namaPeminjam',
            'uTugas.fullname as namaPetugas',
            'c.nama as koleksi',
            'c.id as idKoleksi'
        ])
        ->join('collections as c', 'c.id', '=', 'collectionId')
        ->join('transactions as t', 't.id', '=', 'transactionId')
        ->join('users as uPinjam', 'uPinjam.id', '=', 'dt.userIdPeminjam')
        ->join('users as uTugas', 'uTugas.id', '=', 'dt.userIdPetugas')
        ->where('dt.id', '=', $detailtransactionId)
        ->first();

    return view('detailtransaction.detailTransactionKembalikan', [
        'data' => compact(var_name: 'detailTransaction')
    ]);
}

public function update(Request $request)
{
    if ($request->status == 1) {
    } else {
    }
}

```

**Bottom Editor:** Shows the file `DetailTransactionController.php`. This version handles the update logic for a transaction. It updates the transaction record and then performs database operations on the 'collections' table to adjust item counts based on the transaction status.

```

$affected = DB::table('detail_transactions')
    ->where('id', '=', $request->idDetailTransaksi)
    ->update([
        'status' => $request->status,
        'tanggalKembali' => Carbon::now()->toDateString()
    ]);

if($request->status == 2) {
    // Kalau dikembalikan
    DB::table('collections')->where('id', '=', value: $request->idKoleksi)->increment(column: 'jumlahSisa');
    DB::table('collections')->where('id', '=', value: $request->idKoleksi)->decrement(column: 'jumlahKeluar');
} else {
    // Kalau hilang
    DB::table('collections')->increment(column: 'jumlahSisa');
}

$transaction = Transaction::where('id', '=', $request->idTransaksi)->first();
return redirect(to: 'transaksiView' . $request->idTransaksi)->with(key: 'transaction', value: $transaction);
}

```

## 4. Kelas model

- User,

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz
EXPLORER ... web.php M UserController.php M CollectionController.php M TransactionController.php U DetailTransactionController.php U User.php M
app > Models > User.php > ...
1 <?php
2
3 // 6706223137 Wijdan Andra Fulvian 46-04
4
5 namespace App\Models;
6
7 // use Illuminate\Contracts\Auth\MustVerifyEmail;
8 use Illuminate\Database\Eloquent\Factories\HasFactory;
9 use Illuminate\Foundation\Auth\User as Authenticatable;
10 use Illuminate\Notifications\Notifiable;
11 use Laravel\Sanctum\HasApiTokens;
12
13 46 references | 0 implementations
14 class User extends Authenticatable
15 {
16     use HasApiTokens, HasFactory, Notifiable;
17
18     /**
19      * The attributes that are mass assignable.
20      *
21      * @var array<int, string>
22     */
23     protected $fillable = [
24         'username',
25         'fullname',
26         'email',
27         'password',
28         'address',
29         'birthdate',
30         'phoneNumber',
31     ];
32
33     /**
34      * The attributes that should be hidden for serialization.
35      *
36      * @var array<int, string>
37     */
38     protected $hidden = [
39         'password',
40         'remember_token',
41     ];
42
43     /**
44      * The attributes that should be cast.
45      *
46      * @var array<string, string>
47     */
48     protected $casts = [
49         'email_verified_at' => 'datetime',
50     ];
51

```

Ln 3, Col 41 Spaces: 4 UTF-8 LF ⓘ PHP 8.1 ⚡ Prettier

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz
EXPLORER ... web.php M UserController.php M CollectionController.php M TransactionController.php U DetailTransactionController.php U User.php M
app > Models > User.php > ...
36 /**
37 0 references
38 protected $hidden = [
39     'password',
40     'remember_token',
41 ];
42
43 /**
44 0 references
45 * The attributes that should be cast.
46 *
47 * @var array<string, string>
48 */
49 0 references
50 protected $casts = [
51     'email_verified_at' => 'datetime',
52 ];
53

```

## • Collection,

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz
EXPLORER ... UserController.php M CollectionController.php M TransactionController.php U DetailTransactionController.php U User.php M Collection.php M
app > Models > Collection.php > ...
1 <?php
2
3 // 6706223137 Wijdan Andra Fulvian 46-04
4
5 namespace App\Models;
6
7 use Illuminate\Database\Eloquent\Factories\HasFactory;
8 use Illuminate\Database\Eloquent\Model;
9
10 15 references | 0 implementations
11 class Collection extends Model
12 {
13     use HasFactory;
14     0 references
15     protected $fillable = [
16         'id',
17         'nama',
18         'jenis',
19         'created_at',
20         'jumlahAwal',
21         'jumlahSisa',
22         'jumlahKeluan',
23     ];

```

Ln 3, Col 41 Spaces: 4 UTF-8 LF ⓘ PHP 8.1 ⚡ Prettier

## • Transaction,

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Transaction extends Model
{
    use HasFactory;

    protected $fillable = [
        'id',
        'userIdPertugas',
        'userIdPeminjam',
        'tanggalPinjam',
        'tanggalSelesai',
    ];
}

```

- DetailTransaction

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class DetailTransaction extends Model
{
    use HasFactory;

    protected $fillable = [
        'id',
        'transactionId',
        'collectionId',
        'tanggalKembali',
        'status',
        'keterangan',
    ];
}

```

## 5. Kelas blade View

- Transaksi
  - daftarTransaksi.blade.php

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz
EXPLORER ... roller.php U DetailTransactionController.php U User.php M Collection.php M Transaction.php U DetailTransaction.php U daftarTransaksi.blade.php U
resources > views > transaction > daftarTransaksi.blade.php > ...
1  {{-- 6706223137 Wijdan Andra Fulvian 46-04 --}}
2
3 <x-app-layout>
4   <x-slot name="header">
5     <h2 class="font-semibold text-xl text-gray-800 leading-tight">
6       {{ _('Daftar Transaksi') }}
7     </h2>
8   </x-slot>
9
10  <script type="text/javascript">
11    $(document).ready(function () {
12      $('#yaJra-datatable-transactions').DataTable([
13        ajax: '{ url(' getAllTransactions' ) }',
14        serverSide: false,
15        processing: true,
16        deferRender: true,
17        type: 'GET',
18        destroy:true,
19        columns: [
20          {data:'id', name: 'id'},
21          {data:'peminjam', name: 'peminjam'},
22          {data:'petugas', name: 'petugas'},
23          {data:'tanggallPinjam', name: 'tanggallPinjam'},
24          {data:'tanggalsSelesai', name: 'tanggalsSelesai'},
25          {data:'action', name: 'action', orderable: false, searchable: false}
26        ],
27      });
28    });
29  </script>
30  <div class="py-12">
31    <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
32      <div class="bg-white overflow-hidden sm:rounded-lg">
33        <div class="p-6 bg-white border-b border-gray-200">
34          <div class="container">
35            <table class="table table-bordered yaJra-datatable-transactions">
36              <thead>
37                <tr>
38                  <th>No</th>
39                  <th>Peminjam</th>
40                  <th>Petugas</th>
41                  <th>Tanggal Pinjam</th>
42                  <th>Tanggal Selesai</th>
43                  <th>Opsi</th>
44                </tr>
45              </thead>
46              <tbody>
47                </tbody>
48            </table>
49          </div>
50        </div>
51      </div>
52    </div>
53  </div>
54 </x-app-layout>
55

```

- transaksiTambah.blade.php

Screenshot of a code editor showing two files side-by-side:

**File 1 (Top): transaksiTambah.blade.php**

```

<div class="py-12">
    <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
        <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
            <div class="p-6 bg-white border-b border-gray-200">
                @if ($errors->any())
                    <div class="alert alert-success">
                        <div class="row form-inline" onclick="$(this).parent().remove();">
                            <ul>
                                @foreach ($errors->all as $error)
                                    <li>{{ $error }}</li>
                                @endforeach
                            </ul>
                            <span class="label"><strong>x</strong></span>
                        </div>
                    </div>
                @endif
                <form method="POST" action="{{ url('transaksiStore') }}">
                    @csrf
                    <!-- Peminjam -->
                    <div>
                        <x-input-label for="idPeminjam" :value="__('Peminjam')"/>
                        <select id="idPeminjam" name="idPeminjam" class="form-select" required>
                            <option>-Pilih dahulu--</option>
                            @foreach ($users as $user)
                                @if ($user->id == old('userPeminjam'))
                                    <option value="{{ $user->id }}" selected>{{ $user->fullname }}</option>
                                @else
                                    <option value="{{ $user->id }}>{{ $user->fullname }}</option>
                                @endif
                            @endforeach
                        </select>
                    </div>
                    <!-- Koleksi1 -->
                    <div>
                        <x-input-label for="koleksi1" :value="__('Koleksi1')"/>
                        <select id="koleksi1" name="koleksi1" class="form-select" required>
                            <option value="-1">-Pilih dahulu--</option>
                            @foreach ($collections as $collection)
                                @if ($collection->id == old('koleksi1'))
                                    <option value="{{ $collection->id }}" selected>{{ $collection->nama }}</option>
                                @else
                                    <option value="{{ $collection->id }}>{{ $collection->nama }}</option>
                                @endif
                            @endforeach
                        </select>
                    </div>
                    <!-- Koleksi2 -->
                    <div>
                        <x-input-label for="koleksi2" :value="__('Koleksi2')"/>
                        <select id="koleksi2" name="koleksi2" class="form-select" required>
                            <option value="-1">-Pilih dahulu--</option>
                            @foreach ($collections as $collection)
                                @if ($collection->id == old('koleksi2'))
                                    <option value="{{ $collection->id }}" selected>{{ $collection->nama }}</option>
                                @else

```

**File 2 (Bottom): transaksiTambah.blade.php**

```

38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74

```

```

<option value="{{ $collection->id }}"><{ $collection->nama }</option>
    @endif
    @endforeach
</select>
</div>
<!-- Koleksi3 -->
<div>
    <x-input-label for="koleksi3" :value="__('Koleksi3')"/>
    <select id="koleksi3" name="koleksi3" class="form-select" required>
        <option value="-1">-Pilih dahulu-</option>
        @foreach ($collections as $collection)
            @if ($collection->id == old('koleksi3'))
                <option value="{{ $collection->id }}" selected>{{ $collection->nama }}</option>
            @else
                <option value="{{ $collection->id }}"><{ $collection->nama }</option>
            @endif
        @endforeach
    </select>
    <x-input-error :messages="$errors->get('koleksi2')" class="mt-2"/>
</div>
<div class="flex items-center justify-end mt-4">
    <x-primary-button type="reset" class="ml-4">
        {{ __('Reset') }}
    </x-primary-button>
    <x-primary-button class="ml-4">
        {{ __('OK') }}
    </x-primary-button>
</div>
</div>

```

Ln 1, Col 43   Spaces: 4   UTF-8   LF   Blade   ⚙️   ⚙️   ⚙️

```

resources > views > transaction > transaksiTambah.blade.php > ...
    </div>
    </div>
    </div>
</x-app-layout>

```

- transaksiView.blade.php

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz
resources > views > transaction > transaksiView.blade.php ...
1  ((-- 6706223137 Wijdan Andra Fulvian 46-04 --))
2
3  <x-app-layout>
4      <x-slot name="header">
5          <h2 class="font-semibold text-xl text-gray-800 leading-tight">
6              {{ __('Detail Transaksi') }}
7          </h2>
8      </x-slot>
9
10     <script type="text/javascript">
11         $(document).ready(function () {
12             $('#yajra-datatable-transactions').DataTable([
13                 {
14                     ajax: ('{{ url('getAllDetailTransactions') }}' + '/' + ($transactions->id)),
15                     serverSide: false,
16                     processing: true,
17                     deferRender: true,
18                     type: 'GET',
19                     destroy: true,
20                     columns: [
21                         {data: 'id', name: 'id'},
22                         {data: 'koleksi', name: 'koleksi'},
23                         {data: 'tanggalkinjam', name: 'tanggalkinjam'},
24                         {data: 'tanggalkembalii', name: 'tanggalkembalii'},
25                         {data: 'status', name: 'status'},
26                         {data: 'action', name: 'action'}
27                     ],
28                 });
29             });
30         </script>
31         <div class="py-12">
32             <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
33                 <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
34                     <div class="p-6 bg-white border-b border-gray-200">
35                         <!-- <div>
36                             <x-input-label for="peminjam" :value="__('Peminjam')"/>
37
38                             <x-text-input id="peminjam" class="block mt-1 w-full" type="text" name="peminjam" :value="($transactions->fullnamePeminjam)" autocomplete="off" readonly/>
39                         </div>
40                         <x-input-error :messages="$errors->get('peminjam')" class="mt-2" />
41                     </div>
42                     <div>
43                         <x-input-label for="petugas" :value="__('Petugas')"/>
44
45                         <x-text-input id="petugas" class="block mt-1 w-full" type="text" name="petugas" :value="($transactions->fullnamePetugas)" autocomplete="off" readonly/>
46
47                         <x-input-error :messages="$errors->get('petugas')" class="mt-2" />
48                     </div> -->
49                     <div class="container">
50                         <div class="row form-group">
51                             <label for="peminjam" class="form-label">Peminjam</label>
52                             <input type="text" name="peminjam" id="peminjam" class="form-control" autocomplete="off" value="($transactions->fullnamePeminjam)" readonly>
53                         </div>
54                         <div class="row form-group">
55                             <label for="petugas" class="form-label">Petugas</label>
56                             <input type="text" name="petugas" id="petugas" class="form-control" autocomplete="off" value="($transactions->fullnamePetugas)" readonly>
57                         </div>
58                         <div class="row form-group">
59                             <table class="table table-bordered yajra-datatable-transactions">
60                                 <thead>
61                                     <tr>
62                                         <th>No</th>
63                                         <th>Koleksi</th>
64                                         <th>Tanggal Pinjam</th>
65                                         <th>Tanggal Kembali</th>
66                                         <th>Status</th>
67                                         <th>Opsi</th>
68                                     </tr>
69                                 </thead>
70                                 <tbody>
71                                     </tbody>
72                                 </table>
73                             </div>
74                         </div>
75                     </div>
76                 </div>
77             </div>
78         </div>
79     </x-app-layout>
80
81

```

- Detail Transaksi
  - detailTransactionKembalikan.blade.php

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz
in.php M Transaction.php U DetailTransaction.php U daftarTransaksi.blade.php U transaksiTambah.blade.php U transaksiView.blade.php U detailTransactionKembalikan.blade.php U
resources > views > detailtransaction > detailTransactionKembalikan.blade.php ...
1  {{-- 6706223137 Wijdan Andra Fulvian 46-04 --}}
2
3 <x-app-layout>
4   <x-slot name="header">
5     <h2 class="font-semibold text-xl text-gray-800 leading-tight">
6       {{ __('Edit Pengembalian Detil Transaksi') }}
7     </h2>
8   </x-slot>
9
10 <div class="py-12">
11   <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
12     <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
13       <div class="p-6 bg-white border-b border-gray-200">
14         @if ($errors->any())
15           <div class="alert alert-success">
16             <div class="row form-inline" onclick="$this.parent().remove();">
17               <ul>
18                 @foreach ($errors->all as $error)
19                   <li>{{ $error }}</li>
20                 @endforeach
21               </ul>
22               <span class="label"><strong>X</strong></span>
23             </div>
24           </div>
25         @endif
26         <form method="POST" action="{{ url('detailTransactionUpdate') }}>
27           @csrf
28           <div>
29             <x-input-label for="idTransaksi" :value="__('ID Transaksi')"/>
30
31             <x-text-input id="idTransaksi" class="block mt-1 w-full" type="text" name="idTransaksi" :value="old('idTransaksi')" autocomplete="off" value="{{ $detailTransaction->idTransaksi }}"/>
32
33             <x-input-error :messages="$errors->get('idTransaksi')" class="mt-2"/>
34           </div>
35
36           <div>
37             <x-input-label for="idDetailTransaksi" :value="__('ID Detail Transaksi')"/>

```

Live Share

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Wizz
in.php M Transaction.php U DetailTransaction.php U daftarTransaksi.blade.php U transaksiTambah.blade.php U transaksiView.blade.php U detailTransactionKembalikan.blade.php U
resources > views > detailTransaction > detailTransactionKembalikan.blade.php ...
38
39
40
41   <x-text-input id="idDetailTransaksi" class="block mt-1 w-full" type="text" name="idDetailTransaksi" :value="old('idDetailTransaksi')" autocomplete="off" value="{{ $detailTransaction->idDetailTransaksi }}"/>
42
43   <x-input-error :messages="$errors->get('idDetailTransaksi')" class="mt-2"/>
44
45
46   <div>
47     <x-input-label for="idPeminjam" :value="__('Peminjam')"/>
48
49     <x-text-input id="idPeminjam" class="block mt-1 w-full" type="text" name="idPeminjam" :value="old('idPeminjam')" autocomplete="off" value="{{ $detailTransaction->idPeminjam }}"/>
50
51     <x-input-error :messages="$errors->get('idPeminjam')" class="mt-2"/>
52
53
54   <div>
55     <x-input-label for="idPetugas" :value="__('Petugas')"/>
56
57     <x-text-input id="idPetugas" class="block mt-1 w-full" type="text" name="idPetugas" :value="old('idPetugas')" autocomplete="off" value="{{ $detailTransaction->idPetugas }}"/>
58
59     <x-input-error :messages="$errors->get('idPetugas')" class="mt-2"/>
60
61
62   <div>
63     <input type="hidden" name="idKoleksi" value="{{ $detailTransaction->idKoleksi }}>
64     <x-input-label for="koleksi" :value="__('Koleksi')"/>
65
66     <x-text-input id="koleksi" class="block mt-1 w-full" type="text" name="koleksi" :value="old('koleksi')" autocomplete="off" value="{{ $detailTransaction->koleksi }}"/>
67
68     <x-input-error :messages="$errors->get('koleksi')" class="mt-2"/>
69
70
71   <div>
72     <x-input-label for="status" :value="__('Status')"/>
73
74     <select id="status" name="status" class="form-select" required>
75       <option value="1" @if (old('status', $detailTransaction->status) == 1) selected @endif>Pinjam</option>
76       <option value="2" @if (old('status', $detailTransaction->status) == 2) selected @endif>Kembali</option>
77     </select>
78
79     <x-input-error :messages="$errors->get('status')" class="mt-2"/>
80
81
82   <div class="flex items-center justify-end mt-4">
83     <x-primary-button type="reset" class="ml-4">
84       {{ __('Reset') }}
85     </x-primary-button>
86
87     <x-primary-button class="ml-4">
88       {{ __('OK') }}
89     </x-primary-button>
90   </div>
91
92   </div>
93
94 </div>
95 </x-app-layout>

```