

LAMPIRAN

Lampiran 1. Source Code

routes/web.php

```
<?php
```

```
Route::get('/', [
    'as' => 'guests.index',
    'uses' =>
    'GuestController@index'
]);
```

```
Auth::routes();
```

```
Route::get('auth/verify/{token}',
'Auth\RegisterController@verify');
```

```
Route::get('/home', 'HomeController@index')-
>name('home');
```

```
Route::get('settings/profile',
'SettingsController@profile');
```

```
Route::get('settings/profile/edit',
'SettingsController@editProfile');
```

```
Route::post('settings/profile',
'SettingsController@updateProfile');
```

```
Route::get('settings/edit-password',
'SettingsController@editPassword');
```

```
Route::post('settings/edit-password',
'SettingsController@updatePassword');
```

```
Route::group(['prefix'=>'admin',
'middleware'=>['auth', 'role:admin']], function () {
```

```
    Route::resource('jadwals',
'JadwalController');
```

```
    Route::resource('laporans',
'LaporanController');
```

```
    Route::resource('ruangs',
'RuangController');
```

```
    Route::resource('sisis',
'SesiController');
```

```
    Route::resource('members',
'MembersController');
```

```
    Route::get('export/jadwals', [
        'as' => 'export.jadwals',
        'uses' =>
        'JadwalController@export'
    ]);
```

```
    Route::post('export/jadwals', [
        'as' => 'export.jadwals.post',
        'uses' =>
        'JadwalController@exportPost'
    ]);
```

```
    Route::get('template/jadwals', [
        'as' => 'template.jadwals',
        'uses' =>
        'JadwalController@generateExcelTemplate'
    ]);
```

```
    Route::post('import/jadwals', [
        'as' => 'import.jadwals',
        'uses' =>
        'JadwalController@importExcel'
    ]);
```

```
    Route::get('export/laporans', [
        'as' => 'export.laporans',
        'uses' =>
        'LaporanController@export'
    ]);
```

```
    Route::post('export/laporans', [
```

```

        'as' => 'export.laporans.post',
        'uses' =>
'LaporanController@exportPost'
    );
});
Route::group(['prefix'=>'member',
'middleware'=>['auth', 'role:member']], function ()
{
    Route::resource('books',
'BookController');
});

```

Lampiran 2. Source Code Model

```

<?php
namespace App;
use Illuminate\Database\Eloquent\Model;
class Jadwal extends Model
{
    protected $fillable = ['ruang_id', 'tanggal',
'sesi_id', 'acara', 'user_id', 'pj_nama', 'pj_id',
'keterangan', 'alat', 'status'];
    public function ruang()
    {
        return $this->belongsTo('App\Ruang');
    }
    public function sesi()
    {
        return $this->belongsTo('App\Sesi');
    }
}

```

```

public function user()
{
    return $this->belongsTo('App\User');
}
public function scopeBook($query)
{
    return $query->where('status', 'Booking');
}
public function scopeConfirmed($query)
{
    return $query->where('status', 'Confirmed');
}
}
}

```

Lampiran 3. Source Code View

```

@extends('layouts.app')
@section('content')
<div class="container">
    <div class="row">
        <div class="col-md-12">
            <ul
                class="breadcrumb">
                    <li><a
href="{{ url('/home') }}">Dashboard</a></li>
                    <li
class="active">Data Jadwal</li>
                </ul>

```



```

                $('select[name="filter_stat"]').change(function() {

                    window.LaravelDataTables["dataTableBuilder"].ajax.reload();

                });

            });
        </script>

    @endsection

```

Lampiran 4. Source Code Controller

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Jadwal;

use Yajra\DataTables\Html\Builder;
use Yajra\DataTables\DataTables;

use Session;

use
Illuminate\Database\Eloquent\ModelNotFoundException;

use Illuminate\Support\Facades\Auth;

use Illuminate\Database\Eloquent\Model;

use App\Http\Requests\StoreJadwalRequest;

use App\Http\Requests\UpdateJadwalRequest;

use Excel;

```

```

use PDF;

use Validator;

use App\Ruang;

use App\Sesi;

use App\User;

use Illuminate\Support\Facades\Mail;

use Illuminate\Database\Eloquent\Scope;

class JadwalController extends Controller

{

    public function index(Request $request, Builder $htmlBuilder)

    {

        if ($request->ajax()) {

            $jadwals = Jadwal::with('ruang', 'sesi', 'user')->get();

            if ($request->get('stat') == 'booking')

                $jadwals = Jadwal::with('ruang', 'sesi', 'user')->book()->get();

            if ($request->get('stat') == 'confirmed')

                $jadwals = Jadwal::with('ruang', 'sesi', 'user')->confirmed()->get();

            return DataTables::of($jadwals)

                ->addColumn('action', function($jadwal){

                    return view('datatable._action', [

                        'model' => $jadwal,

                        'form_url' => route('jadwals.destroy', $jadwal->id),

```

```

        'edit_url' => route('jadwals.edit',
$jadwal->id),

        'confirm_message' => 'Yakin mau
menghapus jadwal ruang tanggal ' . $jadwal-
>tanggal . '?'

    ]);

    }->make(true);
}

$html = $htmlBuilder

->addColumn(['data' => 'ruang.nama',
'nama'=>'ruang.nama', 'title'=>'Ruang'])

->addColumn(['data' => 'tanggal',
'nama'=>'tanggal', 'title'=>'Tanggal'])

->addColumn(['data' => 'sesi.nama',
'nama'=>'sesi.nama', 'title'=>'Sesi'])

->addColumn(['data' => 'acara',
'nama'=>'acara', 'title'=>'Acara'])

->addColumn(['data' => 'user.name',
'nama'=>'user.name', 'title'=>'Pengguna'])

->addColumn(['data' => 'created_at',
'nama'=>'created_at', 'title'=>'Waktu Pengajuan'])

->addColumn(['data' => 'status' ,
'nama'=>'status', 'title'=>'Status',
'orderable'=>true, 'searchable'=>true])

->addColumn(['data' => 'action',
'nama'=>'action', 'title'=>'', 'orderable'=>false,
'searchable'=>false]);

return view('jadwals.index')-
>with(compact('html'));
}

/**
 * Show the form for creating a new resource.
 *
 * @return \Illuminate\Http\Response
 */

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

/**
 * Store a newly created resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @return \Illuminate\Http\Response
 */

public function store(StoreJadwalRequest
$request)
{
    $jadwal = Jadwal::create($request-
>except('foto_identitas'));

    // isi field foto jika ada cover yang diupload
    if ($request->hasFile('foto_identitas')) {

        // Mengambil file yang diupload
        $uploaded_cover = $request-
>file('foto_identitas');

        // mengambil extension file

```

```

        $extension = $uploaded_cover-
>getClientOriginalExtension();

        // membuat nama file random berikut
        extension

        $filename = md5(time()) . '.' .
        $extension;

        // menyimpan cover ke folder public/img

        $destinationPath = public_path() .
        DIRECTORY_SEPARATOR . 'img';

        $uploaded_cover-
>move($destinationPath, $filename);

        // mengisi field foto di jadwal dengan
        filename yang baru dibuat

        $jadwal->foto_identitas = $filename;

        $jadwal->save();

    }

    Session::flash("flash_notification", [

        "level"=>"success",

        "message"=>"Jadwal berhasil disimpan"

    ]);

    return redirect()->route('jadwals.index');

}

/**
 * Display the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

public function show($id)
{
    //
}

/**
 * Show the form for editing the specified
 * resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

public function edit($id)
{
    $jadwal = Jadwal::find($id);

    return view('jadwals.edit')-
>with(compact('jadwal'));

}

/**
 * Update the specified resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @param int $id
 * @return \Illuminate\Http\Response
 */

```

```

public function update(UpdateJadwalRequest
$request, $id)

{

    $jadwal = Jadwal::find($id);

    $jadwal->ruang_id      =      $request-
>get('ruang_id');

    $jadwal->tanggal = $request->get('tanggal');

    $jadwal->sesi_id = $request->get('sesi_id');

    $jadwal->acara = $request->get('acara');

    $jadwal->user_id = $request->get('user_id');

    $jadwal->pj_nama      =      $request-
>get('pj_nama');

    $jadwal->pj_id = $request->get('pj_id');

    if ($request->hasFile('foto_identitas')) {

        // mengambil cover yang diupload berikut
ekstensinya

        $filename = null;

        $uploaded_cover      =      $request-
>file('foto_identitas');

        $extension      =      $uploaded_cover-
>getClientOriginalExtension();

        // membuat nama file random dengan
extension

        $filename = md5(time()) . '.' . $extension;

        $destinationPath      =      public_path() .
DIRECTORY_SEPARATOR . 'img';

        // memindahkan file ke folder public/img

        $uploaded_cover-
>move($destinationPath, $filename);

        // hapus cover lama, jika ada

        if ($jadwal->foto_identitas) {

            $old_cover = $jadwal->foto_identitas;

            $filepath      =      public_path() .
DIRECTORY_SEPARATOR . 'img' .
DIRECTORY_SEPARATOR .
$jadwal->foto_identitas;

            try {

                File::delete($filepath);

            } catch (FileNotFoundException $e) {

                // File sudah dihapus/tidak ada

            }

        }

        // ganti field cover dengan cover yang baru

        $jadwal->foto_identitas = $filename;

    }

    $jadwal->keterangan      =      $request-
>get('keterangan');

    $jadwal->alat = $request->get('alat');

    if ($request->get('status') != null) {

        $jadwal->status = 'Confirmed';

        /* $users = User::find($jadwal->user_id);

        // kirim email

```



```

        Mail::send('jadwals.email',
compact('users'), function ($m) use ($users) {
            $m->to($users->email, $users->name)-
>subject('Ruang pesanan Anda telah
dikonfirmasi!'); }); */
    }
    else {
        $jadwal->status = 'Booking';
    }
    $jadwal->save();

    Session::flash("flash_notification", [
        "level"=>"success",
        "message"=>"Jadwal berhasil
diperbaharui."
    ]);
    return redirect()->route('jadwals.index');
}

/**
 * Remove the specified resource from storage.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function destroy($id)
{
    Jadwal::destroy($id);
    Session::flash("flash_notification", [
        "level"=>"success",
        "message"=>"Jadwal berhasil dihapus."
    ]);
    return redirect()->route('jadwals.index');
}

public function export()
{
    return view('jadwals.export');
}

public function exportPost(Request $request)
{
    // validasi
    $this->validate($request, [
        'type'=>'required|in:pdf,xls'
    ]);

    $jadwals = Jadwal::with('ruang', 'sesi', 'user')-
>get();

    $handler = 'export' . ucfirst($request-
>get('type'));

    return $this->$handler($jadwals);
}

private function exportXls($jadwals)
{
    Excel::create('Data Jadwal Ruang FT UMY',
function($excel) use ($jadwals) {

```

```

// Set property
$excel->setTitle('Data Jadwal Ruang FT
UMY')
->setCreator('Yusuf Hanafi Angkat');
$excel->sheet('Data          Jadwal',
function($sheet) use ($jadwals) {
    $row = 1;
    $sheet->row($row, [
        'Ruang',
        'Tanggal',
        'Sesi',
        'Acara',
        'Pengguna',
        'Nama Penanggungjawab',
        'ID Penanggungjawab',
        'Foto Identitas',
        'Keterangan',
        'Alat Dipinjam',
        'Status',
        'Waktu Pengajuan',
        'Terakhir Diperbaharui'
    ]);
foreach ($jadwals as $jadwal) {
    $sheet->row(++$row, [
        $jadwal->ruang->nama,
        $jadwal->tanggal,
        $jadwal->sesi->nama,
        $jadwal->acara,
        $jadwal->user->name,
        $jadwal->pj_nama,
        $jadwal->pj_id,
        $jadwal->foto_identitas,
        $jadwal->keterangan,
        $jadwal->alat,
        $jadwal->status,
        $jadwal->created_at,
        $jadwal->updated_at
    ]);
    }
    }->export('xls');
}

private function exportPdf($jadwals)
{
    $pdf = PDF::loadview('pdf.jadwals',
compact('jadwals'));
    return $pdf->download('jadwals.pdf');
}
}

```