NIM : H1D022006

Nama : Jehian Athaya Tsani Az Zuhry

Kasus : 6. Sistem Informasi Mahasiswa

1. **Tech Stack**

   Laravel 11, Breeze, MySQL, Inertia, Vue, SSR, Pest.

2. **Skema Database**

   a. [Tabel generate dari laravel dan breeze]

   b. Tabel program_studi (id, nama, timestamps)

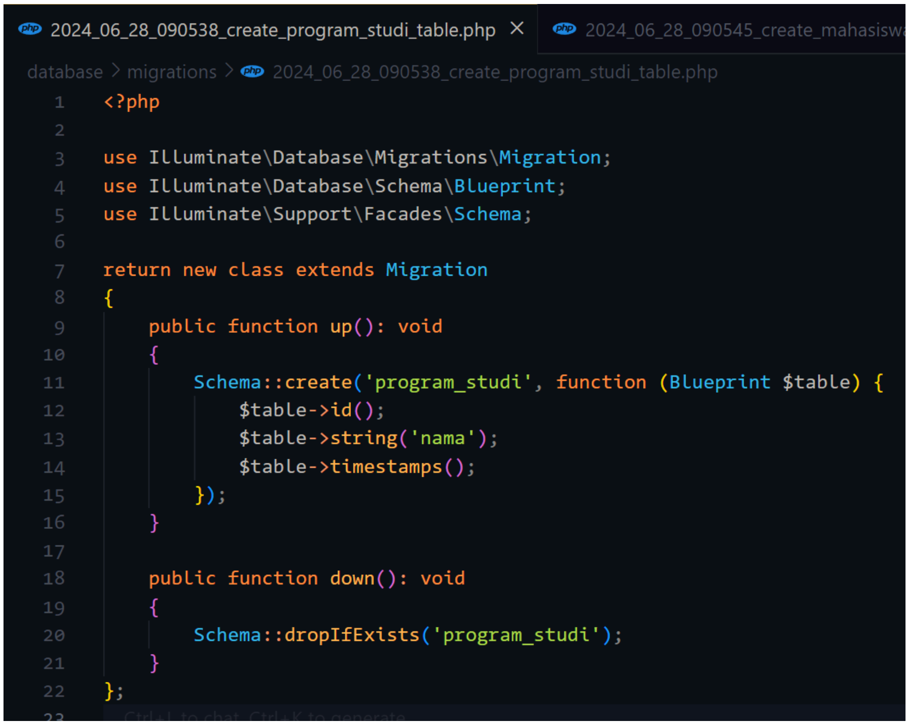
   c. Tabel mahasiswa (id, nama, nim, program_studi_id, timestamps)

   d. Tabel mata_kuliah (id, kode, nama, sks, timestamps)

   e. Tabel nilai (id, mahasiswa_id, mata_kuliah_id, nilai, timestamps)

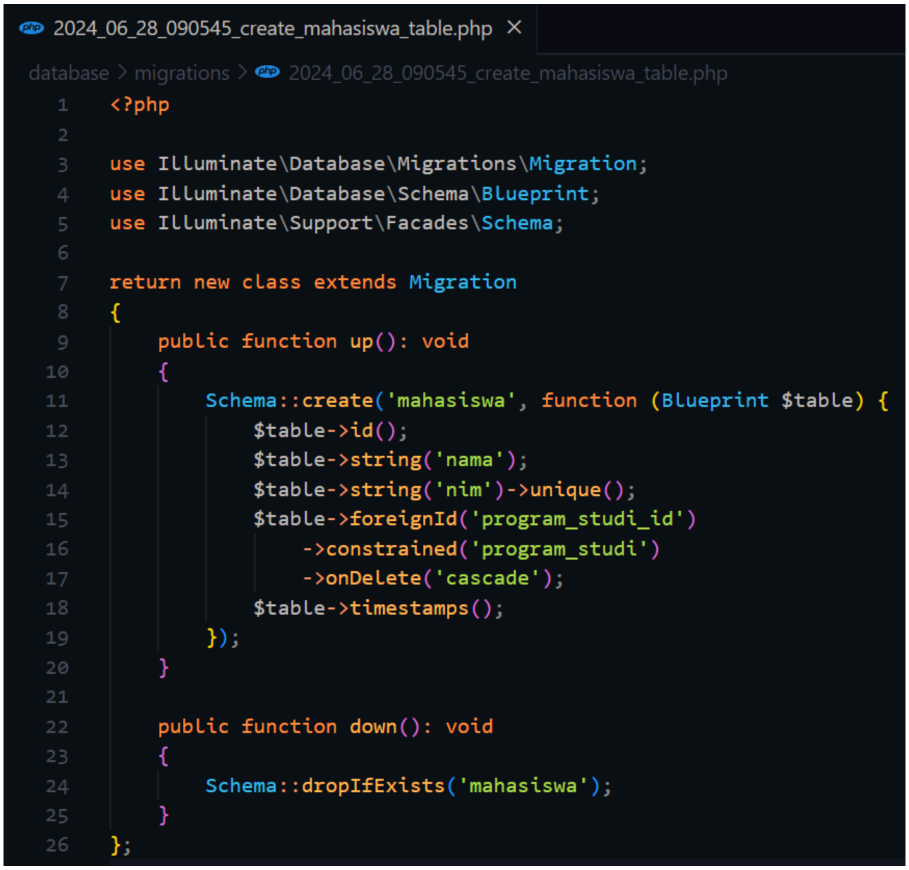
3. **Screenshot Kode Migrations**

   a. 2024_06_28_090538_create_program_studi_table.php


2024_06_28_090538_create_program_studi_table.php
2024_06_28_090545_create_mahasiswa
database > migrations > 2024_06_28_090538_create_program_studi_table.php
Ctrl+L to chat, Ctrl+K to generate

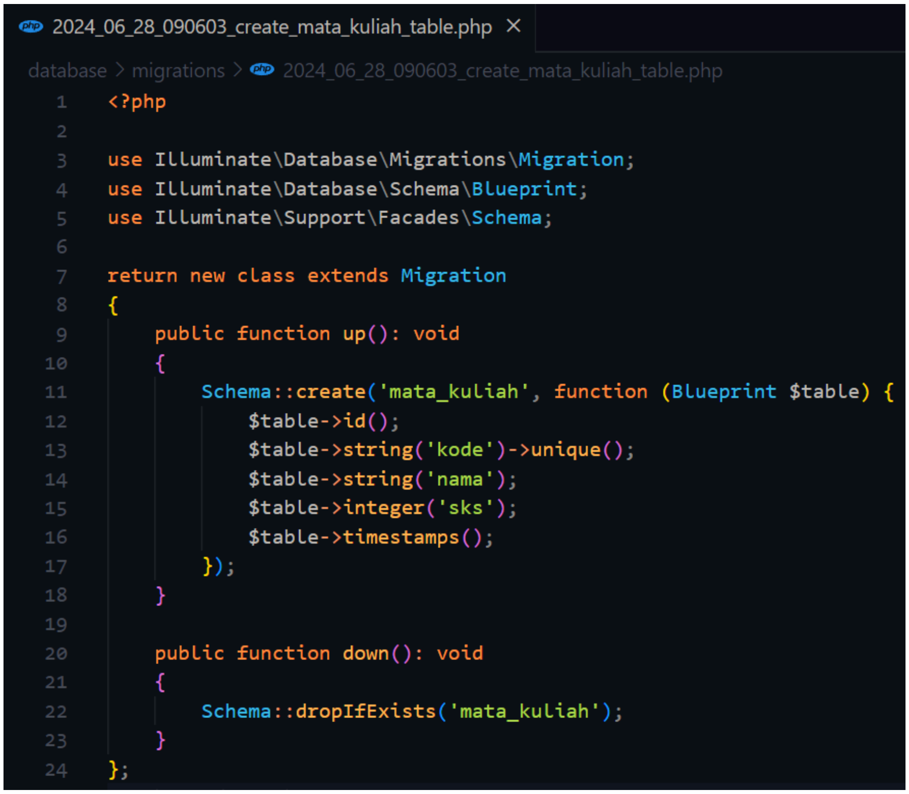
```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    public function up(): void
    {
        Schema::create('program_studi', function (Blueprint $table) {
            $table->id();
            $table->string('nama');
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('program_studi');
    }
};
```

   b. 2024_06_28_090545_create_mahasiswa_table.php


2024_06_28_090545_create_mahasiswa_table.php
database > migrations > 2024_06_28_090545_create_mahasiswa_table.php

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    public function up(): void
    {
        Schema::create('mahasiswa', function (Blueprint $table) {
            $table->id();
            $table->string('nama');
            $table->string('nim')->unique();
            $table->foreignId('program_studi_id')
                ->constrained('program_studi')
                ->onDelete('cascade');
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('mahasiswa');
    }
};
```

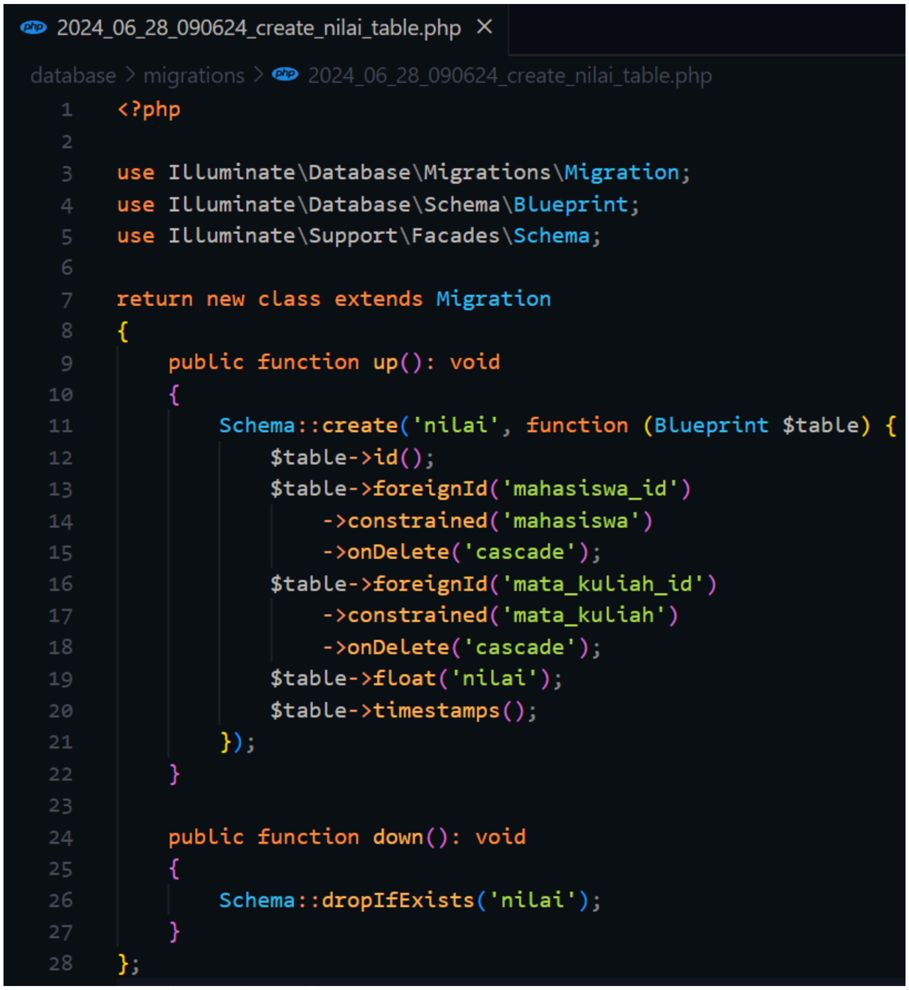
c. 2024_06_28_090603_create_mata_kuliah_table.php



2024_06_28_090603_create_mata_kuliah_table.php ×
database > migrations > 2024_06_28_090603_create_mata_kuliah_table.php

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    public function up(): void
    {
        Schema::create('mata_kuliah', function (Blueprint $table) {
            $table->id();
            $table->string('kode')->unique();
            $table->string('nama');
            $table->integer('sks');
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('mata_kuliah');
    }
};
```

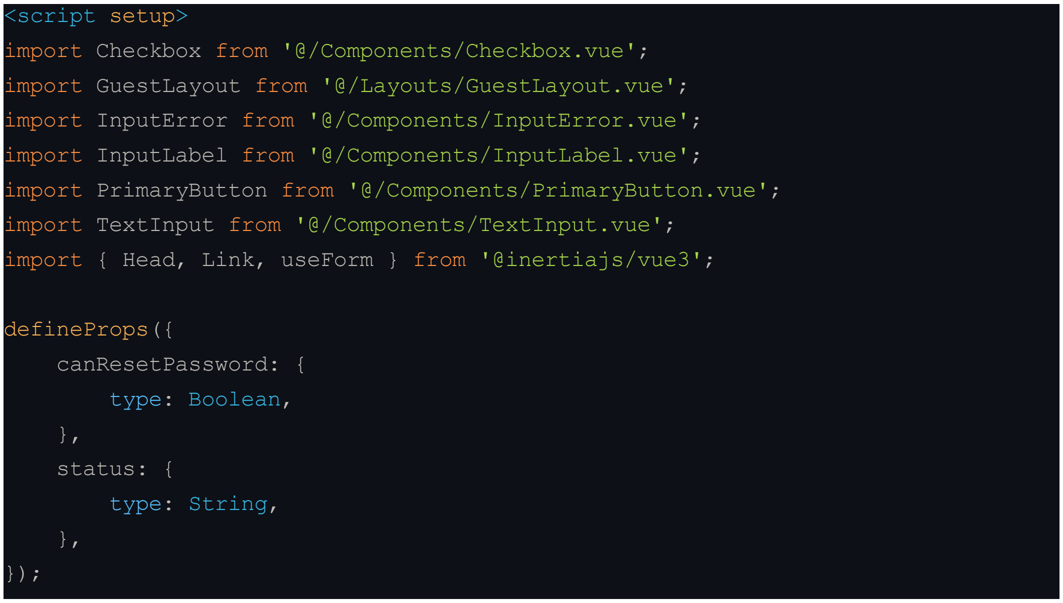
d. 2024_06_28_090624_create_nilai_table.php


2024_06_28_090624_create_nilai_table.php
database > migrations > 2024_06_28_090624_create_nilai_table.php

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    public function up(): void
    {
        Schema::create('nilai', function (Blueprint $table) {
            $table->id();
            $table->foreignId('mahasiswa_id')
                ->constrained('mahasiswa')
                ->onDelete('cascade');
            $table->foreignId('mata_kuliah_id')
                ->constrained('mata_kuliah')
                ->onDelete('cascade');
            $table->float('nilai');
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('nilai');
    }
};
```

4. Screenshot Kode CRUD

    a. resources/js/Pages/Auth/Login.vue



```
<script setup>
import Checkbox from '@/Components/Checkbox.vue';
import GuestLayout from '@/Layouts/GuestLayout.vue';
import InputError from '@/Components/InputError.vue';
import InputLabel from '@/Components/InputLabel.vue';
import PrimaryButton from '@/Components/PrimaryButton.vue';
import TextInput from '@/Components/TextInput.vue';
import { Head, Link, useForm } from '@inertiajs/vue3';

defineProps({
    canResetPassword: {
        type: Boolean,
    },
    status: {
        type: String,
    },
});
```

```
const form = useForm({
    email: '',
    password: '',
    remember: false,
});

const submit = () => {
    form.post(route('login'), {
        onFinish: () => form.reset('password'),
    });
};
</script>

<template>
    <GuestLayout>
        <Head title="Log in" />

        <div v-if="status" class="mb-4 font-medium text-sm text-green-600">
            {{ status }}
        </div>

        <form @submit.prevent="submit">
            <div>
                <InputLabel for="email" value="Email" />

                <TextInput
                    id="email"
                    type="email"
                    class="mt-1 block w-full"
                    v-model="form.email"
                    required
                    autofocus
                    autocomplete="username"
                />

                <InputError class="mt-2" :message="form.errors.email" />
            </div>

            <div class="mt-4">
                <InputLabel for="password" value="Password" />

                <TextInput
                    id="password"
                    type="password"
                    class="mt-1 block w-full"
                    v-model="form.password"
                    required
                    autocomplete="current-password"
                />

                <InputError class="mt-2" :message="form.errors.password" />
            </div>

            <div class="block mt-4">
                <label class="flex items-center">
                    <Checkbox name="remember" v-model:checked="form.remember"
/>
                    <span class="ms-2 text-sm text-gray-600">Remember me</span>
                </label>
            </div>

            <div class="flex items-center justify-end mt-4">
                <Link
                    v-if="canResetPassword"
                    :href="route('password.request')"
                    class="underline text-sm text-gray-600 hover:text-gray-900
rounded-md focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
                >
                    Forgot your password?
                </Link>

                <PrimaryButton class="ms-4" :class="{ 'opacity-25':
form.processing }" :disabled="form.processing">
                    Log in
                </PrimaryButton>
            </div>
        </form>
    </GuestLayout>
</template>
```

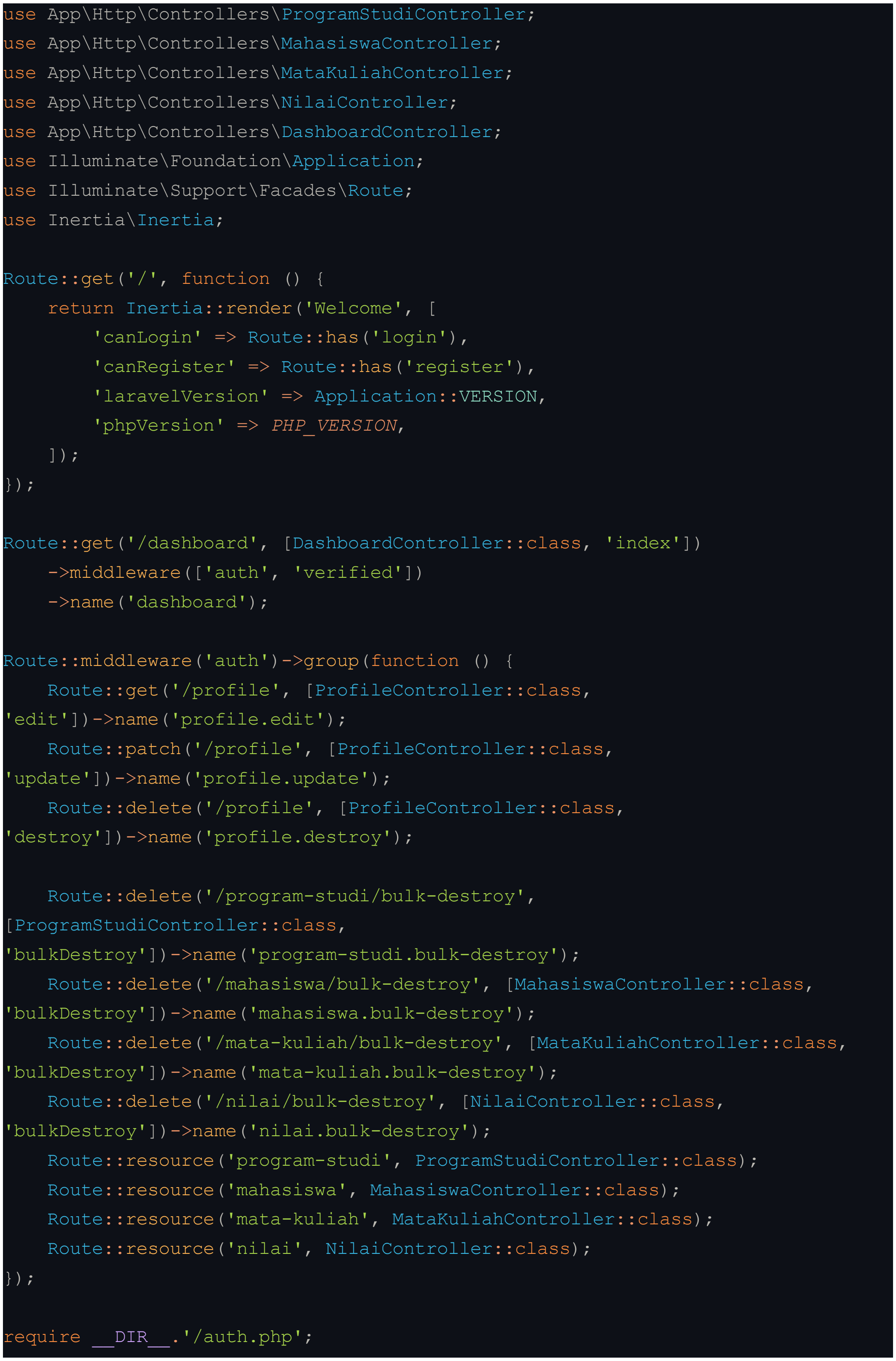
b. resources/js/Pages/Auth/Register.vue

```
<script setup>
import GuestLayout from '@/Layouts/GuestLayout.vue';
import InputError from '@/Components/InputError.vue';
import InputLabel from '@/Components/InputLabel.vue';
import PrimaryButton from '@/Components/PrimaryButton.vue';
import TextInput from '@/Components/TextInput.vue';
import { Head, Link, useForm } from '@inertiajs/vue3';

const form = useForm({
    name: '',
    email: '',
    password: '',
    password_confirmation: '',
});

```

```
const submit = () => {
    form.post(route('register'), {
        onFinish: () => form.reset('password', 'password_confirmation'),
    });
};
</script>

<template>
    <GuestLayout>
        <Head title="Register" />

        <form @submit.prevent="submit">
            <div>
                <InputLabel for="name" value="Name" />

                <TextInput
                    id="name"
                    type="text"
                    class="mt-1 block w-full"
                    v-model="form.name"
                    required
                    autofocus
                    autocomplete="name"
                />

                <InputError class="mt-2" :message="form.errors.name" />
            </div>

            <div class="mt-4">
                <InputLabel for="email" value="Email" />

                <TextInput
                    id="email"
                    type="email"
                    class="mt-1 block w-full"
                    v-model="form.email"
                    required
                    autocomplete="username"
                />

                <InputError class="mt-2" :message="form.errors.email" />
            </div>

            <div class="mt-4">
                <InputLabel for="password" value="Password" />

                <TextInput
                    id="password"
                    type="password"
                    class="mt-1 block w-full"
                    v-model="form.password"
                    required
                    autocomplete="new-password"
                />

                <InputError class="mt-2" :message="form.errors.password" />
            </div>

            <div class="mt-4">
                <InputLabel for="password_confirmation" value="Confirm
Password" />

                <TextInput
                    id="password_confirmation"
                    type="password"
                    class="mt-1 block w-full"
                    v-model="form.password_confirmation"
                    required
                    autocomplete="new-password"
                />

                <InputError class="mt-2"
:message="form.errors.password_confirmation" />
            </div>

            <div class="flex items-center justify-end mt-4">
                <Link
                    :href="route('login')"
                    class="underline text-sm text-gray-600 hover:text-gray-900
rounded-md focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
                >
                    Already registered?
                </Link>

                <PrimaryButton class="ms-4" :class="{ 'opacity-25':
form.processing }" :disabled="form.processing">
                    Register
                </PrimaryButton>
            </div>
        </form>
    </GuestLayout>
</template>
```

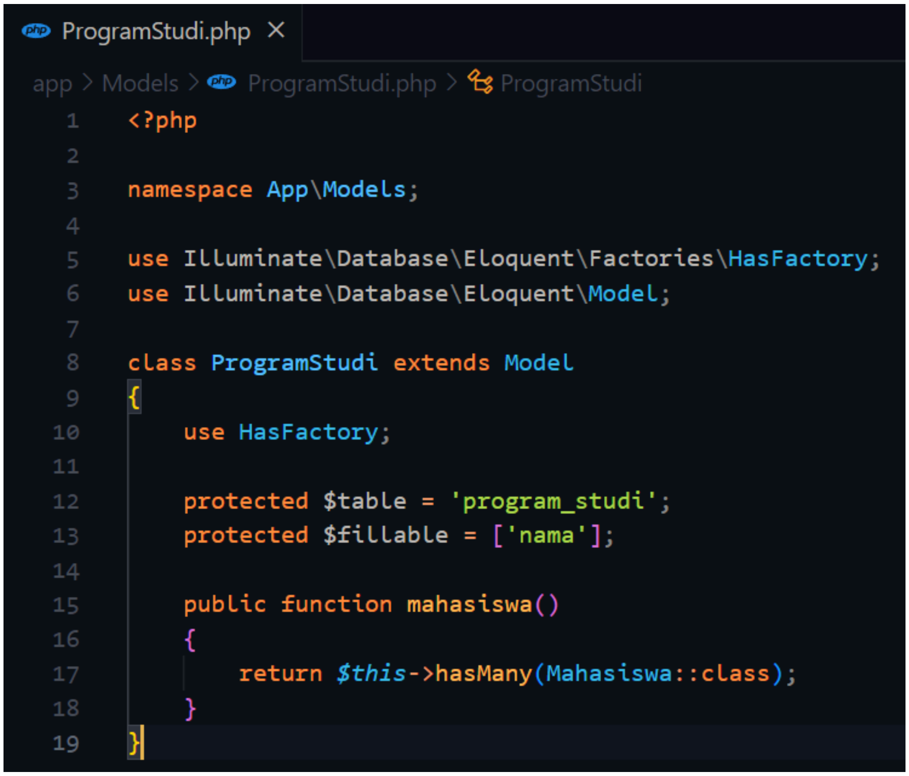
c. routes/web.php

```
<?php

use App\Http\Controllers\ProfileController;
```



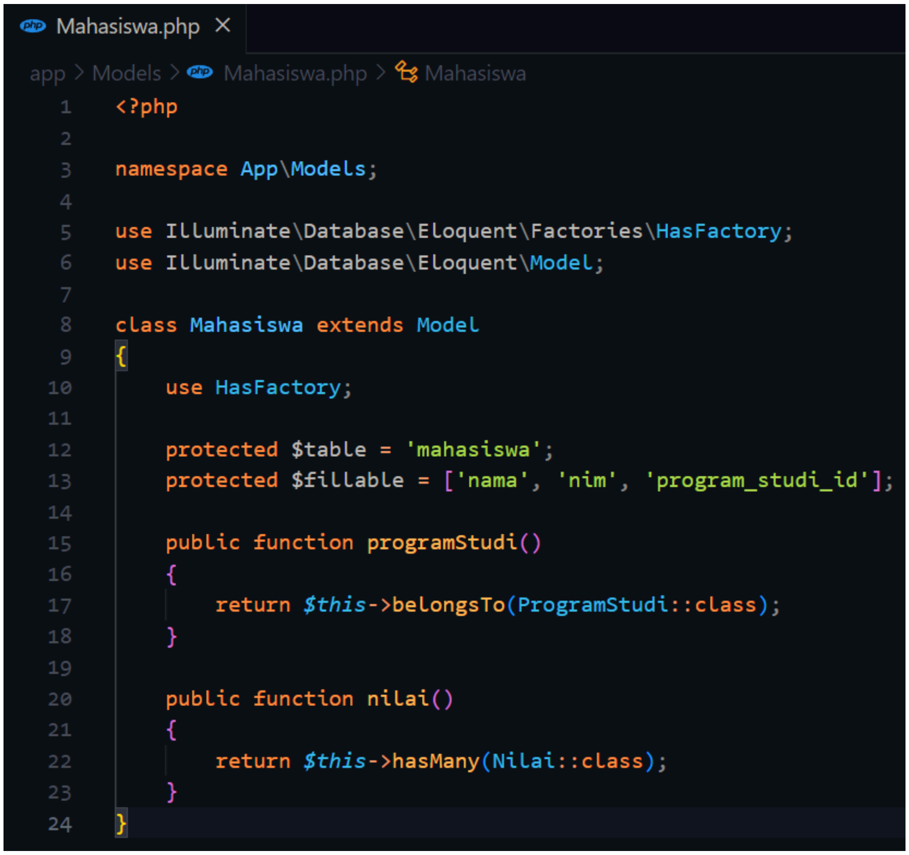
```
use App\Http\Controllers\ProgramStudiController;
use App\Http\Controllers\MahasiswaController;
use App\Http\Controllers\MataKuliahController;
use App\Http\Controllers\NilaiController;
use App\Http\Controllers\DashboardController;
use Illuminate\Foundation\Application;
use Illuminate\Support\Facades\Route;
use Inertia\Inertia;

Route::get('/', function () {
    return Inertia::render('Welcome', [
        'canLogin' => Route::has('login'),
        'canRegister' => Route::has('register'),
        'laravelVersion' => Application::VERSION,
        'phpVersion' => PHP_VERSION,
    ]);
});

Route::get('/dashboard', [DashboardController::class, 'index'])
    ->middleware(['auth', 'verified'])
    ->name('dashboard');

Route::middleware('auth')->group(function () {
    Route::get('/profile', [ProfileController::class,
'edit'])->name('profile.edit');
    Route::patch('/profile', [ProfileController::class,
'update'])->name('profile.update');
    Route::delete('/profile', [ProfileController::class,
'destroy'])->name('profile.destroy');

    Route::delete('/program-studi/bulk-destroy',
[ProgramStudiController::class,
'bulkDestroy'])->name('program-studi.bulk-destroy');
    Route::delete('/mahasiswa/bulk-destroy', [MahasiswaController::class,
'bulkDestroy'])->name('mahasiswa.bulk-destroy');
    Route::delete('/mata-kuliah/bulk-destroy', [MataKuliahController::class,
'bulkDestroy'])->name('mata-kuliah.bulk-destroy');
    Route::delete('/nilai/bulk-destroy', [NilaiController::class,
'bulkDestroy'])->name('nilai.bulk-destroy');
    Route::resource('program-studi', ProgramStudiController::class);
    Route::resource('mahasiswa', MahasiswaController::class);
    Route::resource('mata-kuliah', MataKuliahController::class);
    Route::resource('nilai', NilaiController::class);
});

require __DIR__.'/auth.php';
```

d. Models/ProgramStudi.php



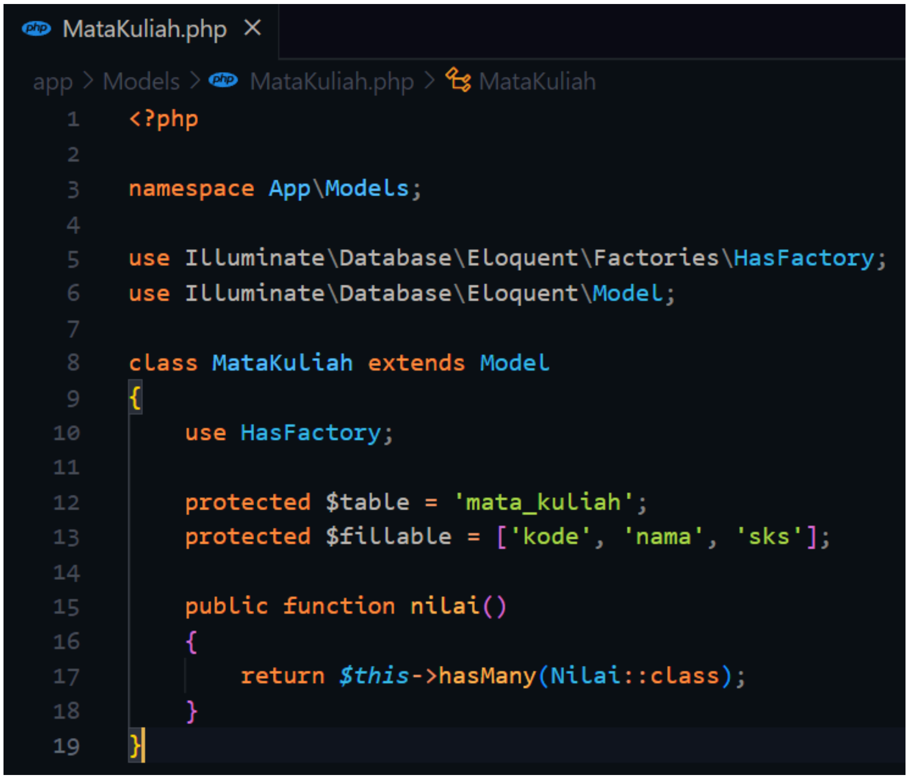
```php
<?php

namespace App\Models;

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

class ProgramStudi extends Model
{
    use HasFactory;

    protected $table = 'program_studi';
    protected $fillable = ['nama'];

    public function mahasiswa()
    {
        return $this->hasMany(Mahasiswa::class);
    }
}
```

e. Models/Mahasiswa.php



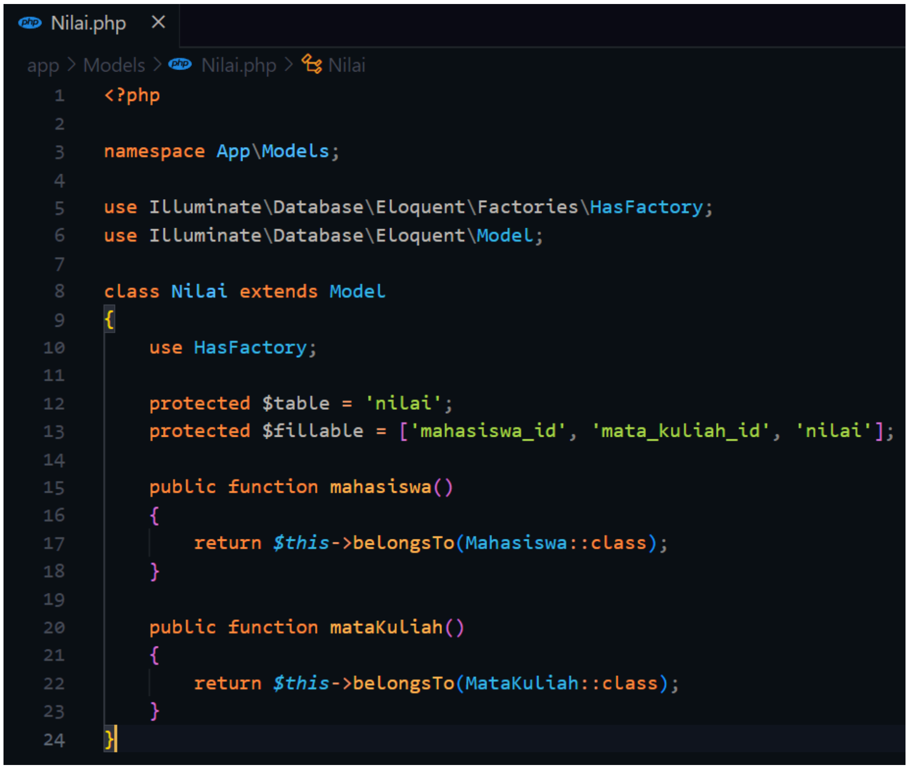
```php
<?php

namespace App\Models;

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

class Mahasiswa extends Model
{
    use HasFactory;

    protected $table = 'mahasiswa';
    protected $fillable = ['nama', 'nim', 'program_studi_id'];

    public function programStudi()
    {
        return $this->belongsTo(ProgramStudi::class);
    }

    public function nilai()
    {
        return $this->hasMany(Nilai::class);
    }
}
```

f. Models/MataKuliah.php



```php
<?php

namespace App\Models;

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

class MataKuliah extends Model
{
    use HasFactory;

    protected $table = 'mata_kuliah';
    protected $fillable = ['kode', 'nama', 'sks'];

    public function nilai()
    {
        return $this->hasMany(Nilai::class);
    }
}
```

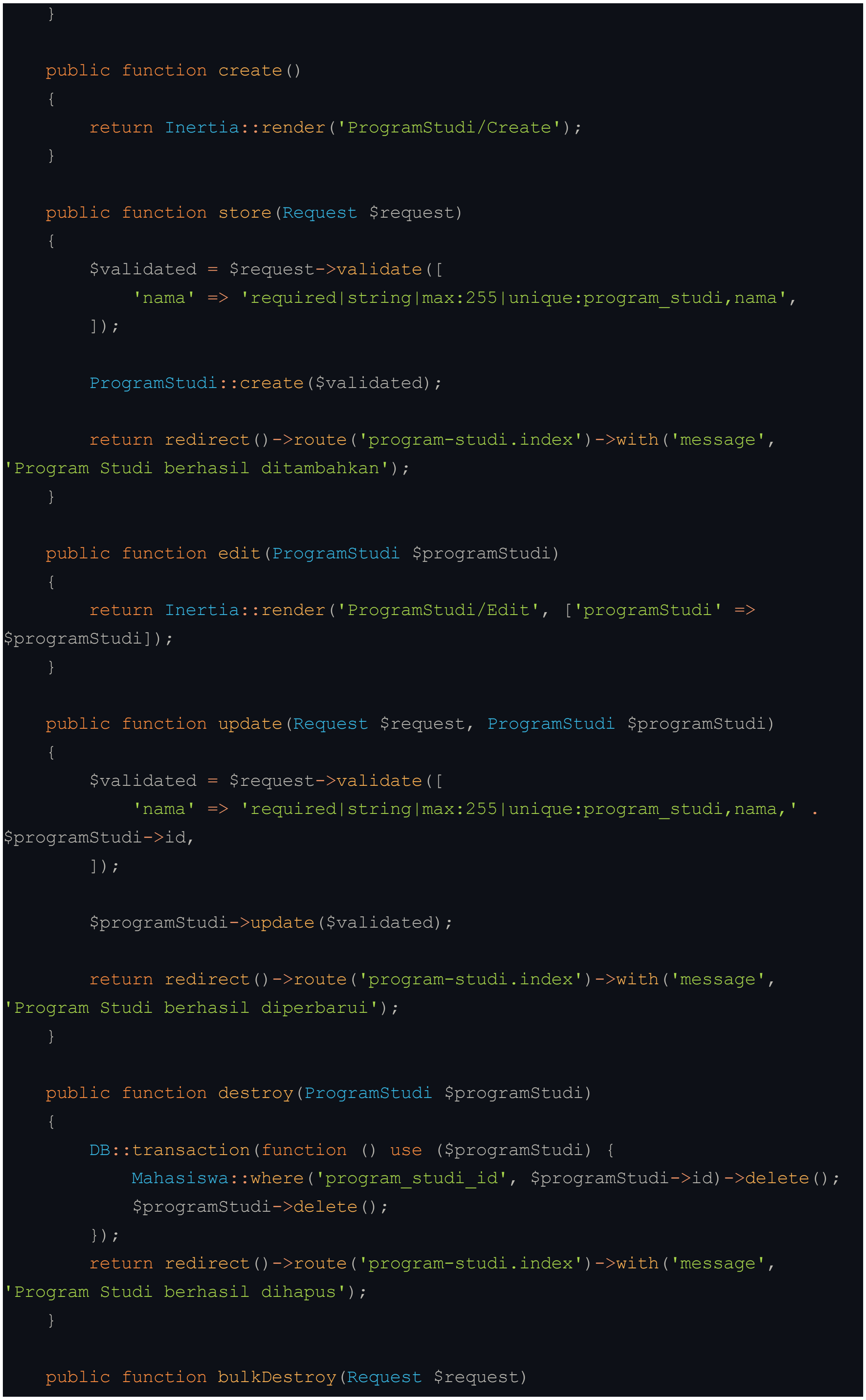
g. Models/Nilai.php



```php
<?php

namespace App\Models;

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

class Nilai extends Model
{
    use HasFactory;

    protected $table = 'nilai';
    protected $fillable = ['mahasiswa_id', 'mata_kuliah_id', 'nilai'];

    public function mahasiswa()
    {
        return $this->belongsTo(Mahasiswa::class);
    }

    public function mataKuliah()
    {
        return $this->belongsTo(MataKuliah::class);
    }
}
```

h. ProgramStudiController.php

```
<?php

namespace App\Http\Controllers;

use App\Models\ProgramStudi;
use App\Models\Mahasiswa;
use Illuminate\Http\Request;
use Inertia\Inertia;
use Illuminate\Support\Facades\DB;

class ProgramStudiController extends Controller
{
    public function index(Request $request)
    {
        $search = $request->input('search');
        $page = $request->input('page', 1);
        $sort = $request->input('sort');
        $direction = $request->input('direction', 'asc');

        $query = ProgramStudi::query();

        if ($search) {
            $query->where('nama', 'like', "%{$search}%");
        }

        if ($sort) {
            $query->orderBy($sort, $direction);
        }

        $programStudi = $query->paginate(10, ['*'], 'page',
$page)->withQueryString();

        return Inertia::render('ProgramStudi/Index', [
            'programStudi' => $programStudi->items(),
            'pagination' => [
                'current_page' => $programStudi->currentPage(),
                'last_page' => $programStudi->lastPage(),
                'per_page' => $programStudi->perPage(),
                'total' => $programStudi->total(),
                'next_page_url' => $programStudi->nextPageUrl(),
                'prev_page_url' => $programStudi->previousPageUrl(),
            ],
            'search' => $search,
            'sort' => [
                'column' => $sort,
                'direction' => $direction,
            ],
        ]);
```



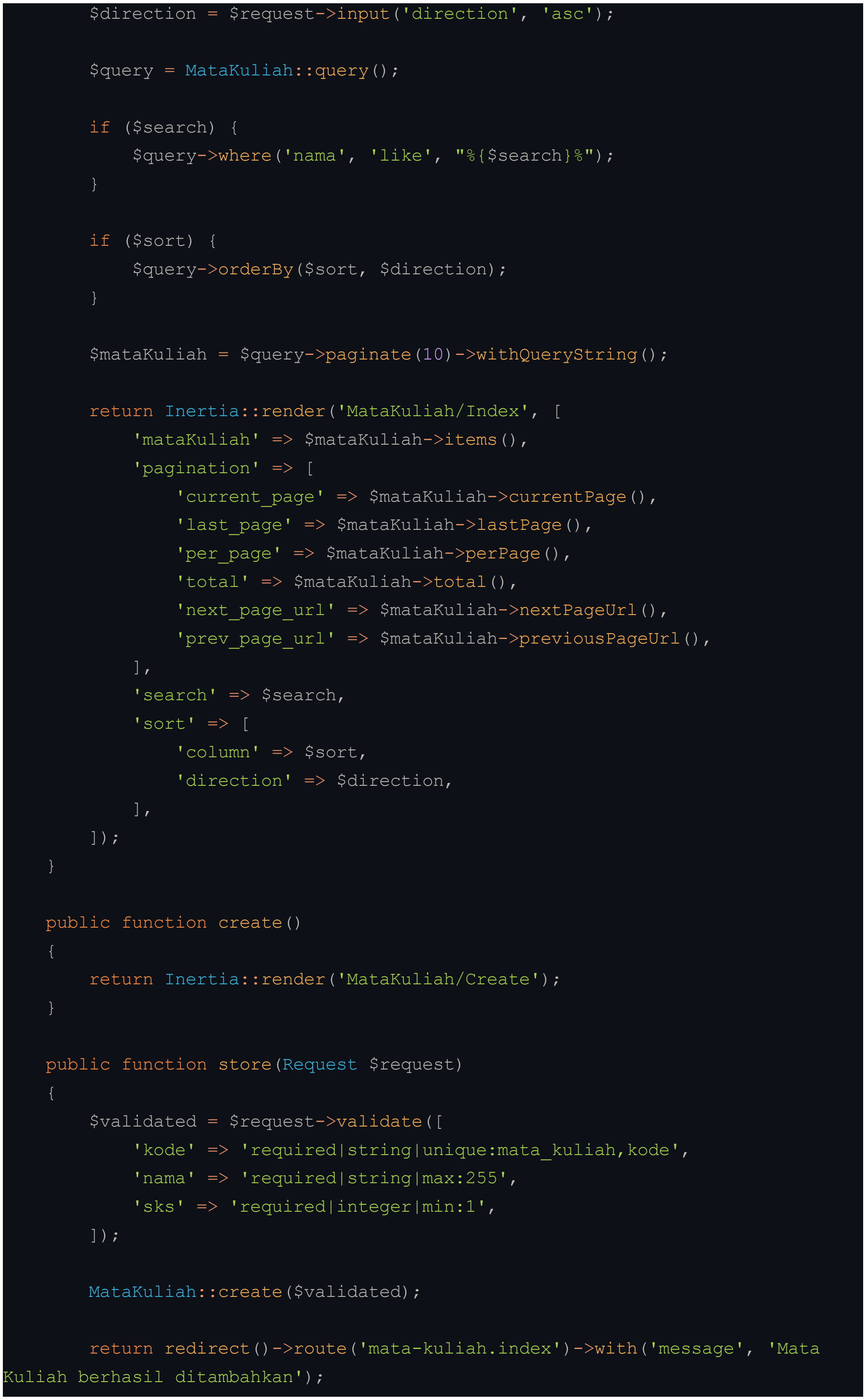
```
    }

    public function create()
    {
        return Inertia::render('ProgramStudi/Create');
    }

    public function store(Request $request)
    {
        $validated = $request->validate([
            'nama' => 'required|string|max:255|unique:program_studi,nama',
        ]);

        ProgramStudi::create($validated);

        return redirect()->route('program-studi.index')->with('message',
'Program Studi berhasil ditambahkan');
    }

    public function edit(ProgramStudi $programStudi)
    {
        return Inertia::render('ProgramStudi/Edit', ['programStudi' =>
$programStudi]);
    }

    public function update(Request $request, ProgramStudi $programStudi)
    {
        $validated = $request->validate([
            'nama' => 'required|string|max:255|unique:program_studi,nama,' .
$programStudi->id,
        ]);

        $programStudi->update($validated);

        return redirect()->route('program-studi.index')->with('message',
'Program Studi berhasil diperbarui');
    }

    public function destroy(ProgramStudi $programStudi)
    {
        DB::transaction(function () use ($programStudi) {
            Mahasiswa::where('program_studi_id', $programStudi->id)->delete();
            $programStudi->delete();
        });
        return redirect()->route('program-studi.index')->with('message',
'Program Studi berhasil dihapus');
    }

    public function bulkDestroy(Request $request)
    {
        $ids = $request->input('ids');
        DB::transaction(function () use ($ids) {
            Mahasiswa::whereIn('program_studi_id', $ids)->delete();
            ProgramStudi::whereIn('id', $ids)->delete();
        });
        return redirect()->route('program-studi.index')->with('message',
'Program Studi terpilih berhasil dihapus');
    }
}
```

i. MahasiswaController.php

```
<?php

namespace App\Http\Controllers;

use App\Models\Mahasiswa;
use App\Models\ProgramStudi;
use App\Models\Nilai;
use Illuminate\Http\Request;
use Inertia\Inertia;
use Illuminate\Support\Facades\DB;

class MahasiswaController extends Controller
{
    public function index(Request $request)
    {
        $search = $request->input('search');
        $page = $request->input('page', 1);
        $sort = $request->input('sort');
        $direction = $request->input('direction', 'asc');

        $query = Mahasiswa::with('programStudi');

        if ($search) {
            $query->where('nama', 'like', "%{$search}%");
        }

        if ($sort) {
            $query->orderBy($sort, $direction);
        }

        $mahasiswa = $query->paginate(10, ['*'], 'page',
$page)->withQueryString();

        return Inertia::render('Mahasiswa/Index', [
            'mahasiswa' => $mahasiswa->items(),
            'pagination' => [
                'current_page' => $mahasiswa->currentPage(),
                'last_page' => $mahasiswa->lastPage(),
                'per_page' => $mahasiswa->perPage(),
                'total' => $mahasiswa->total(),
                'next_page_url' => $mahasiswa->nextPageUrl(),
                'prev_page_url' => $mahasiswa->previousPageUrl(),
            ],
            'search' => $search,
            'sort' => [
                'column' => $sort,
                'direction' => $direction,
            ],
        ]);
    }

    public function create()
    {
        $programStudi = ProgramStudi::all();
        return Inertia::render('Mahasiswa/Create', ['programStudi' =>
$programStudi]);
    }

    public function store(Request $request)
    {
        $validated = $request->validate([
            'nama' => 'required|string|max:255',
            'nim' => 'required|string|unique:mahasiswa,nim',
            'program_studi_id' => 'required|exists:program_studi,id',
        ]);

        Mahasiswa::create($validated);

        return redirect()->route('mahasiswa.index')->with('message', 'Mahasiswa
berhasil ditambahkan');
    }

    public function edit(Mahasiswa $mahasiswa)
    {
        $programStudi = ProgramStudi::all();
        return Inertia::render('Mahasiswa/Edit', [
            'mahasiswa' => $mahasiswa,
            'programStudi' => $programStudi
        ]);
    }

    public function update(Request $request, Mahasiswa $mahasiswa)
    {
        $validated = $request->validate([
            'nama' => 'required|string|max:255',
            'nim' => 'required|string|unique:mahasiswa,nim,'.$mahasiswa->id,
            'program_studi_id' => 'required|exists:program_studi,id',
        ]);

        $mahasiswa->update($validated);

        return redirect()->route('mahasiswa.index')->with('message', 'Mahasiswa berhasil diperbarui');
    }

    public function destroy(Mahasiswa $mahasiswa)
    {
        DB::transaction(function () use ($mahasiswa) {
            Nilai::where('mahasiswa_id', $mahasiswa->id)->delete();
            $mahasiswa->delete();
        });
        return redirect()->route('mahasiswa.index')->with('message', 'Mahasiswa berhasil dihapus');
    }

    public function bulkDestroy(Request $request)
    {
        $ids = $request->input('ids');
        DB::transaction(function () use ($ids) {
            Nilai::whereIn('mahasiswa_id', $ids)->delete();
            Mahasiswa::whereIn('id', $ids)->delete();
        });
        return redirect()->route('mahasiswa.index')->with('message', 'Mahasiswa terpilih berhasil dihapus');
    }
}
```

j. MataKuliahController.php

```
<?php

namespace App\Http\Controllers;

use App\Models\MataKuliah;
use App\Models\Nilai;
use Illuminate\Http\Request;
use Inertia\Inertia;
use Illuminate\Support\Facades\DB;

class MataKuliahController extends Controller
{
    public function index(Request $request)
    {
        $search = $request->input('search');
        $page = $request->input('page', 1);
        $sort = $request->input('sort');
```

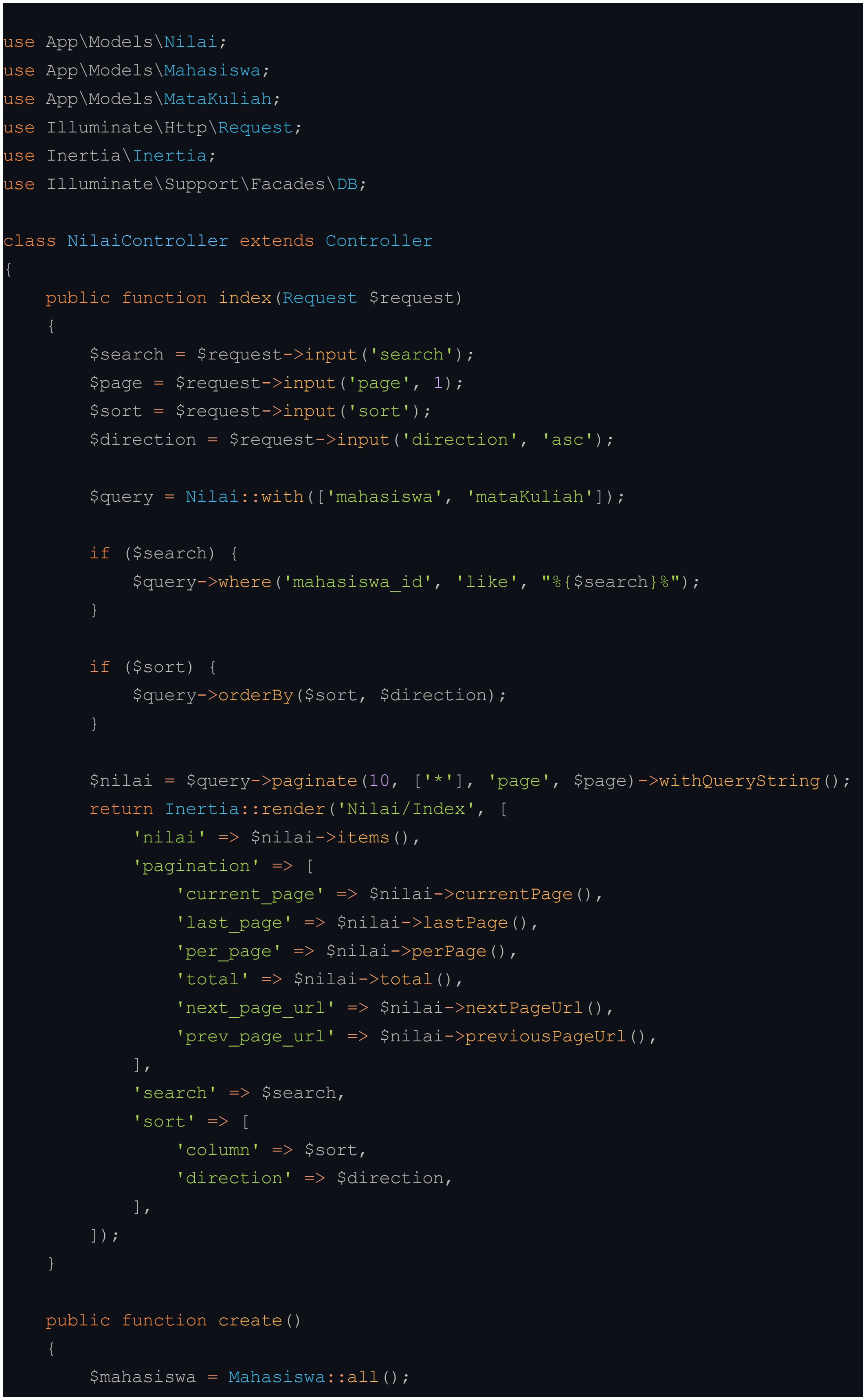


```
        $direction = $request->input('direction', 'asc');

        $query = MataKuliah::query();

        if ($search) {
            $query->where('nama', 'like', "%{$search}%");
        }

        if ($sort) {
            $query->orderBy($sort, $direction);
        }

        $mataKuliah = $query->paginate(10)->withQueryString();

        return Inertia::render('MataKuliah/Index', [
            'mataKuliah' => $mataKuliah->items(),
            'pagination' => [
                'current_page' => $mataKuliah->currentPage(),
                'last_page' => $mataKuliah->lastPage(),
                'per_page' => $mataKuliah->perPage(),
                'total' => $mataKuliah->total(),
                'next_page_url' => $mataKuliah->nextPageUrl(),
                'prev_page_url' => $mataKuliah->previousPageUrl(),
            ],
            'search' => $search,
            'sort' => [
                'column' => $sort,
                'direction' => $direction,
            ],
        ]);
    }

    public function create()
    {
        return Inertia::render('MataKuliah/Create');
    }

    public function store(Request $request)
    {
        $validated = $request->validate([
            'kode' => 'required|string|unique:mata_kuliah,kode',
            'nama' => 'required|string|max:255',
            'sks' => 'required|integer|min:1',
        ]);

        MataKuliah::create($validated);

        return redirect()->route('mata-kuliah.index')->with('message', 'Mata
Kuliah berhasil ditambahkan');
```

```
    }

    public function edit(MataKuliah $mataKuliah)
    {
        return Inertia::render('MataKuliah/Edit', ['mataKuliah' =>
$mataKuliah]);
    }

    public function update(Request $request, MataKuliah $mataKuliah)
    {
        $validated = $request->validate([
            'kode' =>
'required|string|unique:mata_kuliah,kode,'.$mataKuliah->id,
            'nama' => 'required|string|max:255',
            'sks' => 'required|integer|min:1',
        ]);

        $mataKuliah->update($validated);

        return redirect()->route('mata-kuliah.index')->with('message', 'Mata
Kuliah berhasil diperbarui');
    }

    public function destroy(MataKuliah $mataKuliah)
    {
        DB::transaction(function () use ($mataKuliah) {
            Nilai::where('mata_kuliah_id', $mataKuliah->id)->delete();
            $mataKuliah->delete();
        });
        return redirect()->route('mata-kuliah.index')->with('message', 'Mata
Kuliah berhasil dihapus');
    }

    public function bulkDestroy(Request $request)
    {
        $ids = $request->input('ids');
        DB::transaction(function () use ($ids) {
            Nilai::whereIn('mata_kuliah_id', $ids)->delete();
            MataKuliah::whereIn('id', $ids)->delete();
        });
        return redirect()->route('mata-kuliah.index')->with('message', 'Mata
Kuliah terpilih berhasil dihapus');
    }
}
```

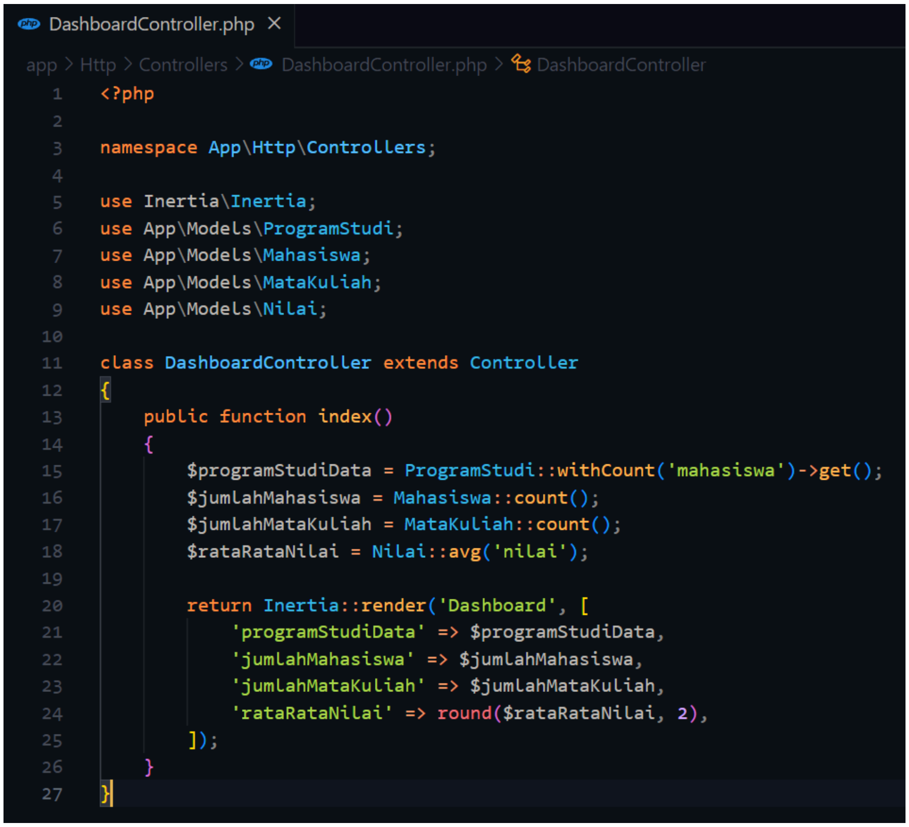
k. NilaiController.php

```
<?php

namespace App\Http\Controllers;
```



```php
use App\Models\Nilai;
use App\Models\Mahasiswa;
use App\Models\MataKuliah;
use Illuminate\Http\Request;
use Inertia\Inertia;
use Illuminate\Support\Facades\DB;

class NilaiController extends Controller
{
    public function index(Request $request)
    {
        $search = $request->input('search');
        $page = $request->input('page', 1);
        $sort = $request->input('sort');
        $direction = $request->input('direction', 'asc');

        $query = Nilai::with(['mahasiswa', 'mataKuliah']);

        if ($search) {
            $query->where('mahasiswa_id', 'like', "%{$search}%");
        }

        if ($sort) {
            $query->orderBy($sort, $direction);
        }

        $nilai = $query->paginate(10, ['*'], 'page', $page)->withQueryString();
        return Inertia::render('Nilai/Index', [
            'nilai' => $nilai->items(),
            'pagination' => [
                'current_page' => $nilai->currentPage(),
                'last_page' => $nilai->lastPage(),
                'per_page' => $nilai->perPage(),
                'total' => $nilai->total(),
                'next_page_url' => $nilai->nextPageUrl(),
                'prev_page_url' => $nilai->previousPageUrl(),
            ],
            'search' => $search,
            'sort' => [
                'column' => $sort,
                'direction' => $direction,
            ],
        ]);
    }

    public function create()
    {
        $mahasiswa = Mahasiswa::all();
        $mataKuliah = MataKuliah::all();
        return Inertia::render('Nilai/Create', [
            'mahasiswa' => $mahasiswa,
            'mataKuliah' => $mataKuliah
        ]);
    }

    public function store(Request $request)
    {
        $validated = $request->validate([
            'mahasiswa_id' => 'required|exists:mahasiswa,id',
            'mata_kuliah_id' => 'required|exists:mata_kuliah,id',
            'nilai' => 'required|numeric|min:0|max:100',
        ]);

        Nilai::create($validated);

        return redirect()->route('nilai.index')->with('message', 'Nilai
berhasil ditambahkan');
    }

    public function edit(Nilai $nilai)
    {
        $mahasiswa = Mahasiswa::all();
        $mataKuliah = MataKuliah::all();
        return Inertia::render('Nilai/Edit', [
            'nilai' => $nilai,
            'mahasiswa' => $mahasiswa,
            'mataKuliah' => $mataKuliah
        ]);
    }

    public function update(Request $request, Nilai $nilai)
    {
        $validated = $request->validate([
            'mahasiswa_id' => 'required|exists:mahasiswa,id',
            'mata_kuliah_id' => 'required|exists:mata_kuliah,id',
            'nilai' => 'required|numeric|min:0|max:100',
        ]);

        $nilai->update($validated);

        return redirect()->route('nilai.index')->with('message', 'Nilai
berhasil diperbarui');
    }

    public function destroy(Nilai $nilai)
    {
        DB::transaction(function () use ($nilai) {
            $nilai->delete();
        });
        return redirect()->route('nilai.index')->with('message', 'Nilai berhasil dihapus');
    }

    public function bulkDestroy(Request $request)
    {
        $ids = $request->input('ids');
        DB::transaction(function () use ($ids) {
            Nilai::whereIn('id', $ids)->delete();
        });
        return redirect()->route('nilai.index')->with('message', 'Nilai terpilih berhasil dihapus');
    }
}
```

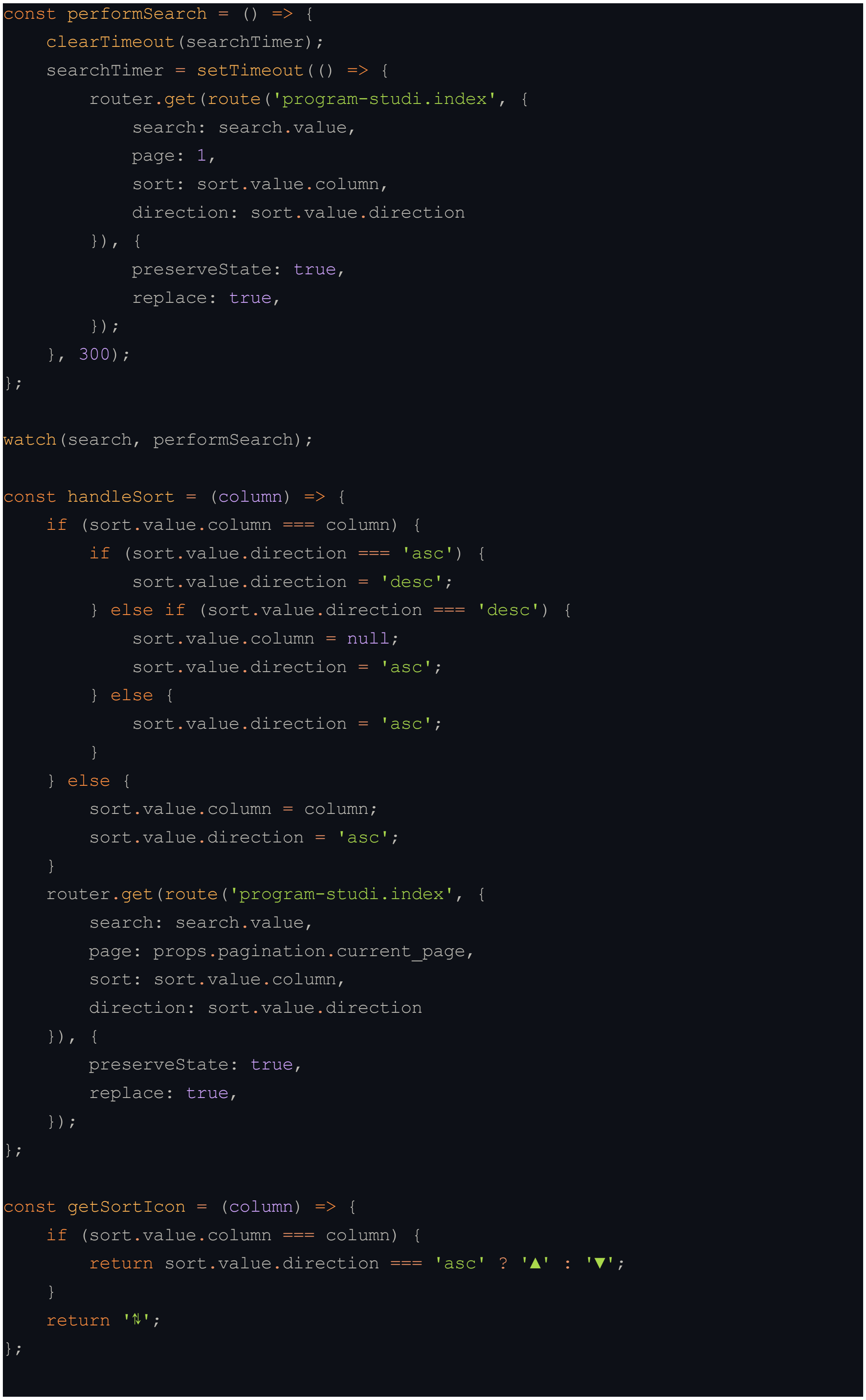
l. DashboardController.php (bukan crud, hanya untuk menampilkan kesimpulan aliran data)


DashboardController.php ×
app > Http > Controllers > DashboardController.php > DashboardController

```
<?php

namespace App\Http\Controllers;

use Inertia\Inertia;
use App\Models\ProgramStudi;
use App\Models\Mahasiswa;
use App\Models\MataKuliah;
use App\Models\Nilai;

class DashboardController extends Controller
{
    public function index()
    {
        $programStudiData = ProgramStudi::withCount('mahasiswa')->get();
        $jumlahMahasiswa = Mahasiswa::count();
        $jumlahMataKuliah = MataKuliah::count();
        $rataRataNilai = Nilai::avg('nilai');

        return Inertia::render('Dashboard', [
            'programStudiData' => $programStudiData,
            'jumlahMahasiswa' => $jumlahMahasiswa,
            'jumlahMataKuliah' => $jumlahMataKuliah,
            'rataRataNilai' => round($rataRataNilai, 2),
        ]);
    }
}
```

m. resources/js/Pages/ProgramStudi/Index.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, Link, useForm, router, usePage } from '@inertiajs/vue3';
import { ref, defineProps, computed, watch, onMounted } from 'vue';

const props = defineProps({
    programStudi: Array,
    pagination: {
        type: Object,
        required: true,
    },
    search: String,
    sort: Object,
});

const handleDelete = (id) => {
  if (window.confirm('Apakah Anda yakin ingin menghapus data ini?')) {
    router.delete(route('program-studi.destroy', { program_studi: id }));
  }
};

const selectedItems = ref([]);

const form = useForm({
    ids: [],
});

const handleBulkDelete = () => {
    if (window.confirm('Apakah Anda yakin ingin menghapus data yang dipilih?'))
{
        form.ids = selectedItems.value;
        form.delete(route('program-studi.bulk-destroy'), {
            preserveScroll: true,
            preserveState: true,
            onSuccess: () => {
                selectedItems.value = [];
            },
        });
    }
};

const toggleSelectAll = (event) => {
    if (event.target.checked) {
        selectedItems.value = props.programStudi.map(program_studi =>
program_studi.id);
    } else {
        selectedItems.value = [];
    }
};

const isAllSelected = computed(() => {
    return selectedItems.value.length === props.programStudi.length;
});

const selectedCount = computed(() => {
    return selectedItems.value.length;
});

const previousPage = () => {
    if (props.pagination.prev_page_url) {
        router.get(route('program-studi.index', { search: search.value, page: props.pagination.current_page - 1 }));
    }
};

const nextPage = () => {
    if (props.pagination.next_page_url) {
        router.get(route('program-studi.index', { search: search.value, page: props.pagination.current_page + 1 }));
    }
};

const goToPage = (page) => {
    router.get(route('program-studi.index', { search: search.value, page: page }));
};

const paginationRange = () => {
    const current = props.pagination.current_page;
    const last = props.pagination.last_page;
    const delta = 2;
    const range = [];
    const rangeWithDots = [];

    for (let i = Math.max(2, current - delta); i <= Math.min(last - 1, current + delta); i++) {
        range.push(i);
    }

    if (current - delta > 2) rangeWithDots.push('...');
    rangeWithDots.push(...range);
    if (current + delta < last - 1) rangeWithDots.push('...');

    return rangeWithDots;
};

const search = ref(props.search || '');
let searchTimer = null;

const sort = ref(props.sort || { column: null, direction: 'asc' });
```



```
const performSearch = () => {
    clearTimeout(searchTimer);
    searchTimer = setTimeout(() => {
        router.get(route('program-studi.index', {
            search: search.value,
            page: 1,
            sort: sort.value.column,
            direction: sort.value.direction
        }), {
            preserveState: true,
            replace: true,
        });
    }, 300);
};

watch(search, performSearch);

const handleSort = (column) => {
    if (sort.value.column === column) {
        if (sort.value.direction === 'asc') {
            sort.value.direction = 'desc';
        } else if (sort.value.direction === 'desc') {
            sort.value.column = null;
            sort.value.direction = 'asc';
        } else {
            sort.value.direction = 'asc';
        }
    } else {
        sort.value.column = column;
        sort.value.direction = 'asc';
    }
    router.get(route('program-studi.index', {
        search: search.value,
        page: props.pagination.current_page,
        sort: sort.value.column,
        direction: sort.value.direction
    }), {
        preserveState: true,
        replace: true,
    });
};

const getSortIcon = (column) => {
    if (sort.value.column === column) {
        return sort.value.direction === 'asc' ? '▲' : '▼';
    }
    return '⇅';
};
```

```
const page = usePage();
const message = ref('');

watch(() => page.props.flash.message, (newMessage) => {
    message.value = newMessage;
    if (newMessage) {
        setTimeout(() => {
            message.value = '';
        }, 5000);
    }
}, { immediate: true });
</script>

<template>
    <Head title="Program Studi" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800
leading-tight">Program Studi</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="message" class="mb-4 bg-green-100 border
border-green-400 text-green-700 px-4 py-3 rounded relative" role="alert">
                    <span class="block sm:inline">{{ message }}</span>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <div class="flex justify-between mb-2">
                            <div class="space-x-2">
                                <Link :href="route('program-studi.create')"
class="inline-flex items-center px-4 py-2 bg-gray-800 border border-transparent
rounded-md font-semibold text-xs text-white uppercase tracking-widest
hover:bg-gray-700 focus:bg-gray-700 active:bg-gray-900 focus:outline-none
focus:ring-2 focus:ring-indigo-500 focus:ring-offset-2 transition ease-in-out
duration-150">
                                    <span class="hidden sm:inline">Tambah
Program Studi</span>
                                    <span class="sm:hidden">+</span>
                                </Link>

                            </div>
                            <div class="relative">
                                <input
                                    v-model="search"
                                    type="text"
                                    placeholder="Cari program studi..."
                                    class="border rounded px-2 py-1
focus:outline-none focus:ring-2 focus:ring-blue-500"
                                />
                                <svg
                                    class="h-5 w-5 text-gray-400 absolute
right-2 top-1/2 transform -translate-y-1/2"
                                    xmlns="http://www.w3.org/2000/svg"
                                    fill="none"
                                    viewBox="0 0 24 24"
                                    stroke="currentColor"
                                >
                                    <path
                                        stroke-linecap="round"
                                        stroke-linejoin="round"
                                        stroke-width="2"
                                        d="M21 21l-6-6m2-5a7 7 0 11-14 0 7 7 0
0114 0z"
                                    />
                                </svg>
                            </div>
                        </div>
                        <div class="flex justify-between mb-2">
                            <span v-if="selectedCount > 0" class="text-sm
text-gray-600">
                                {{ selectedCount }} item terpilih
                            </span>
                            <button v-if="selectedCount > 0"
@click="handleBulkDelete" class="px-4 py-1 bg-red-600 text-white rounded-md
hover:bg-red-700">
                                Hapus Data
                            </button>
                        </div>

                        <div v-if="props.programStudi.length === 0"
class="text-center py-4 text-gray-600">
                            Tidak ada data yang sesuai.
                        </div>

                        <table v-else class="min-w-full divide-y
divide-gray-200">
                            <thead>
                                <tr>
                                    <th class="px-6 py-3 bg-gray-50 text-left">
                                        <input class="rounded-[4px]"
type="checkbox" @change="toggleSelectAll" :checked="isAllSelected">
                                    </th>
                                    <th @click="handleSort('nama')" class="px-6
py-3 bg-gray-50 text-left text-xs leading-4 font-medium text-gray-500 uppercase
tracking-wider cursor-pointer">
```

```
                                        Nama
                                        <span class="ml-1">{{
getSortIcon('nama') }}</span>
                                    </th>
                                    <th class="px-6 py-3 bg-gray-50 text-left
text-xs leading-4 font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
                                </tr>
                            </thead>
                            <tbody class="bg-white divide-y divide-gray-200">
                                <tr v-for="program_studi in props.programStudi"
:key="program_studi.id">
                                    <td class="px-6 py-4 whitespace-no-wrap">
                                        <input class="rounded-[4px]"
type="checkbox" v-model="selectedItems" :value="program_studi.id">
                                    </td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
program_studi.nama }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap
flex gap-2">
                                        <Link
:href="route('program-studi.edit', { program_studi: program_studi.id })"
class="text-yellow-600 hover:text-yellow-900">Edit</Link>
                                        <button
@click="handleDelete(program_studi.id)" class="text-red-600
hover:text-red-900">Hapus</button>
                                    </td>
                                </tr>
                            </tbody>
                        </table>

                        <div class="mt-4 flex justify-between items-center">
                            <div>
                                Tampil {{ (props.pagination.current_page - 1) *
props.pagination.per_page + 1 }}-{{ Math.min(props.pagination.current_page *
props.pagination.per_page, props.pagination.total) }} dari {{
props.pagination.total }} data
                            </div>
                            <div class="flex space-x-2">
                                <button @click="previousPage"
:disabled="!props.pagination.prev_page_url" class="px-4 py-2 bg-gray-800
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.prev_page_url }">
                                    <span class="hidden sm:inline">Prev</span>
                                    <span class="sm:hidden"><</span>
                                </button>
                                <button @click="goToPage(1)" class="hidden
sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === 1 }">
                                    1
                                </button>
                                <template v-for="page in paginationRange()"
:key="page">
                                    <span v-if="page === '...'" class="hidden
sm:inline px-4 py-2">...</span>
                                    <button v-else @click="goToPage(page)"
class="hidden sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{
'bg-blue-500': props.pagination.current_page === page }">
                                        {{ page }}
                                    </button>
                                </template>
                                <button v-if="props.pagination.last_page > 1"
@click="goToPage(props.pagination.last_page)" class="hidden sm:inline px-4 py-2
bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === props.pagination.last_page }">
                                    {{ props.pagination.last_page }}
                                </button>
                                <button @click="nextPage"
:disabled="!props.pagination.next_page_url" class="px-4 py-2 bg-gray-800
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.next_page_url }">
                                    <span class="hidden sm:inline">Next</span>
                                    <span class="sm:hidden">></span>
                                </button>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

n. resources/js/Pages/ProgramStudi/Create.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';

const form = useForm({
    nama: '',
});

const submit = () => {
    form.post(route('program-studi.store'));
};

const errors = computed(() => usePage().props.errors);
</script>
```

```
<template>
    <Head title="Tambah Program Studi" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800
leading-tight">Tambah Program Studi</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
                                <label for="nama" class="block font-medium
text-sm text-gray-700">Nama Program Studi</label>
                                <input id="nama" v-model="form.nama"
type="text" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.nama"
class="text-red-500 mt-2 text-sm">{{ form.errors.nama }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Simpan</button>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

o. resources/js/Pages/ProgramStudi/Edit.vue

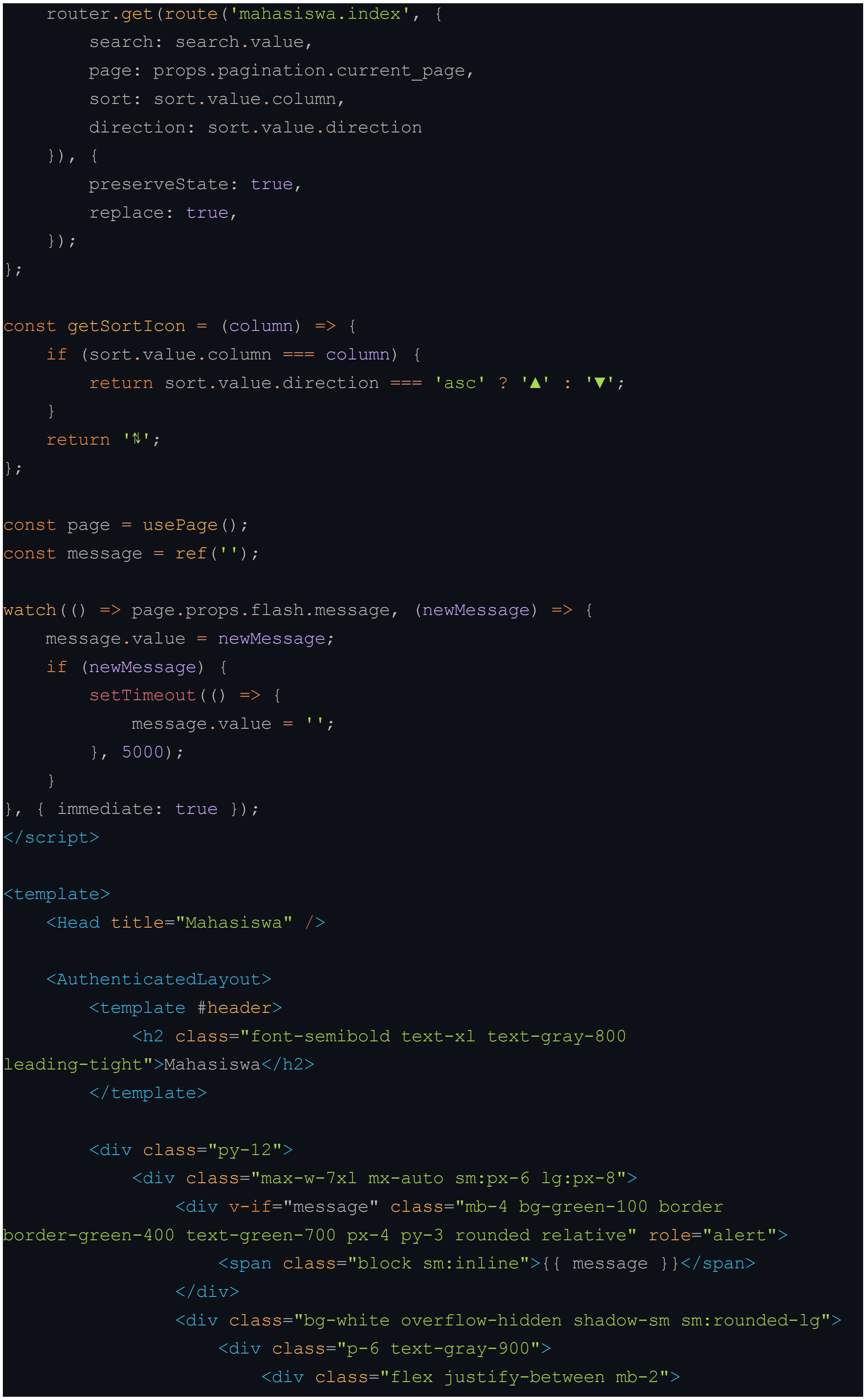
```
<script setup>
```

```
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';

const props = defineProps({
    programStudi: Object,
});

const form = useForm({
    nama: props.programStudi.nama,
});

const submit = () => {
    form.put(route('program-studi.update', props.programStudi.id));
};

const errors = computed(() => usePage().props.errors);
</script>

<template>
    <Head title="Edit Program Studi" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800 leading-tight">Edit
Program Studi</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
                                <label for="nama" class="block font-medium
text-sm text-gray-700">Nama Program Studi</label>
                                <input id="nama" v-model="form.nama"
type="text" class="mt-1 block w-full" required autofocus>
```

```
                                <div v-if="form.errors.nama"
class="text-red-500 mt-2 text-sm">{{ form.errors.nama }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Update</button>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

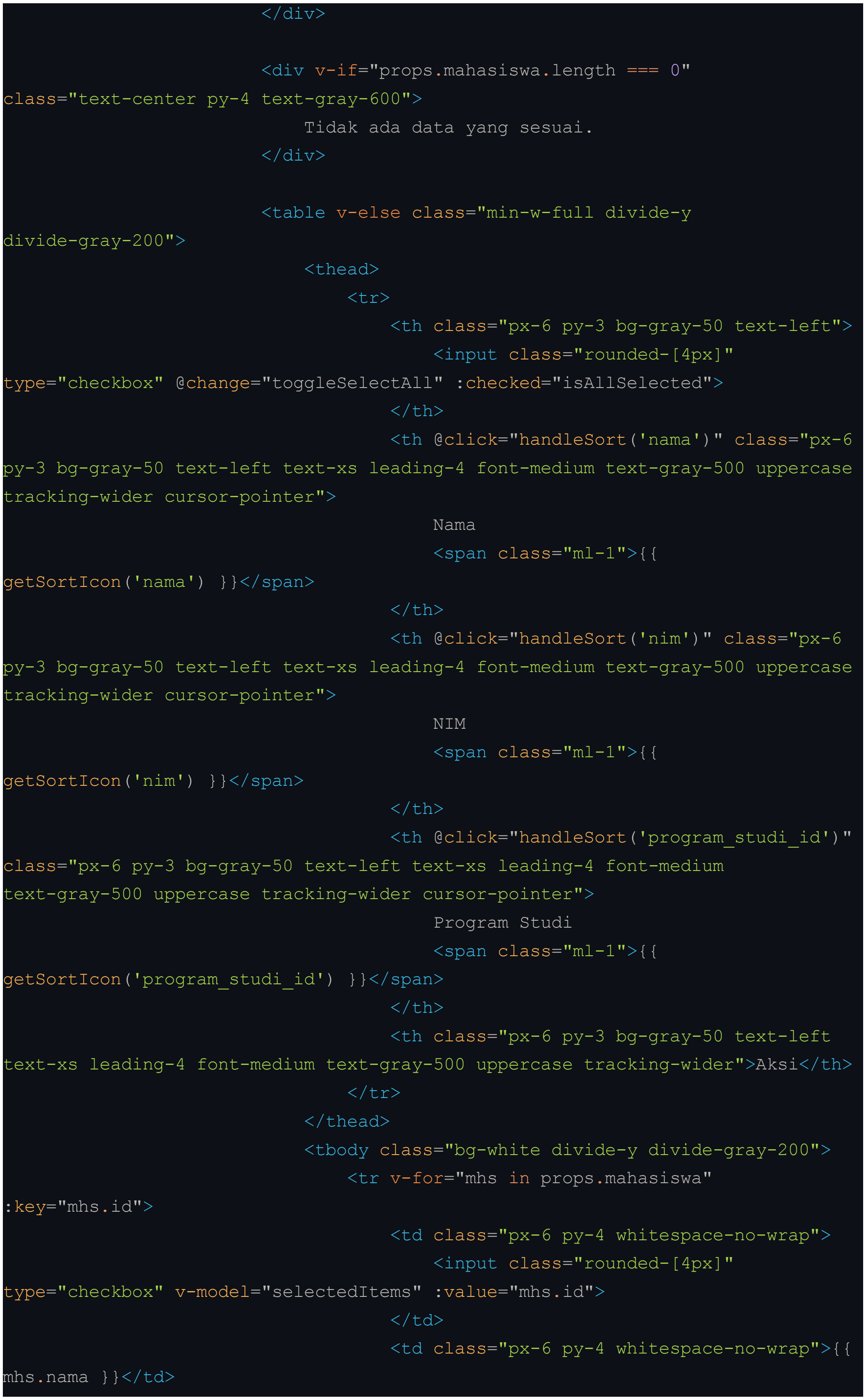
p. resources/js/Pages/Mahasiswa/Index.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, Link, useForm, router, usePage } from '@inertiajs/vue3';
import { ref, defineProps, computed, watch, onMounted } from 'vue';

const props = defineProps({
    mahasiswa: Array,
    pagination: {
        type: Object,
        required: true,
    },
    search: String,
    sort: Object,
});

const handleDelete = (id) => {
  if (window.confirm('Apakah Anda yakin ingin menghapus data ini?')) {
    router.delete(route('mahasiswa.destroy', { mahasiswa: id }));
  }
};

const selectedItems = ref([]);

const form = useForm({
    ids: [],
});

const handleBulkDelete = () => {
    if (window.confirm('Apakah Anda yakin ingin menghapus data yang dipilih?'))
{
        form.ids = selectedItems.value;
        form.delete(route('mahasiswa.bulk-destroy'), {
            preserveScroll: true,
            preserveState: true,
            onSuccess: () => {
                selectedItems.value = [];
            },
        });
    }
};

const toggleSelectAll = (event) => {
    if (event.target.checked) {
        selectedItems.value = props.mahasiswa.map(mhs => mhs.id);
    } else {
        selectedItems.value = [];
    }
};

const isAllSelected = computed(() => {
    return selectedItems.value.length === props.mahasiswa.length;
});

const selectedCount = computed(() => {
    return selectedItems.value.length;
});

const previousPage = () => {
    if (props.pagination.prev_page_url) {
        router.get(route('mahasiswa.index', { search: search.value, page:
props.pagination.current_page - 1 }));
    }
};

const nextPage = () => {
    if (props.pagination.next_page_url) {
        router.get(route('mahasiswa.index', { search: search.value, page:
props.pagination.current_page + 1 }));
    }
};

const goToPage = (page) => {
    router.get(route('mahasiswa.index', { search: search.value, page: page }));
};

const paginationRange = () => {
    const current = props.pagination.current_page;
    const last = props.pagination.last_page;
    const delta = 2;
    const range = [];
    const rangeWithDots = [];
    for (let i = Math.max(2, current - delta); i <= Math.min(last - 1, current
+ delta); i++) {
        range.push(i);
    }

    if (current - delta > 2) rangeWithDots.push('...');
    rangeWithDots.push(...range);
    if (current + delta < last - 1) rangeWithDots.push('...');

    return rangeWithDots;
};

const search = ref(props.search || '');
let searchTimer = null;

const sort = ref(props.sort || { column: null, direction: 'asc' });

const performSearch = () => {
    clearTimeout(searchTimer);
    searchTimer = setTimeout(() => {
        router.get(route('mahasiswa.index', {
            search: search.value,
            page: 1,
            sort: sort.value.column,
            direction: sort.value.direction
        }), {
            preserveState: true,
            replace: true,
        });
    }, 300);
};

watch(search, performSearch);

const handleSort = (column) => {
    if (sort.value.column === column) {
        if (sort.value.direction === 'asc') {
            sort.value.direction = 'desc';
        } else if (sort.value.direction === 'desc') {
            sort.value.column = null;
            sort.value.direction = 'asc';
        } else {
            sort.value.direction = 'asc';
        }
    } else {
        sort.value.column = column;
        sort.value.direction = 'asc';
    }
```

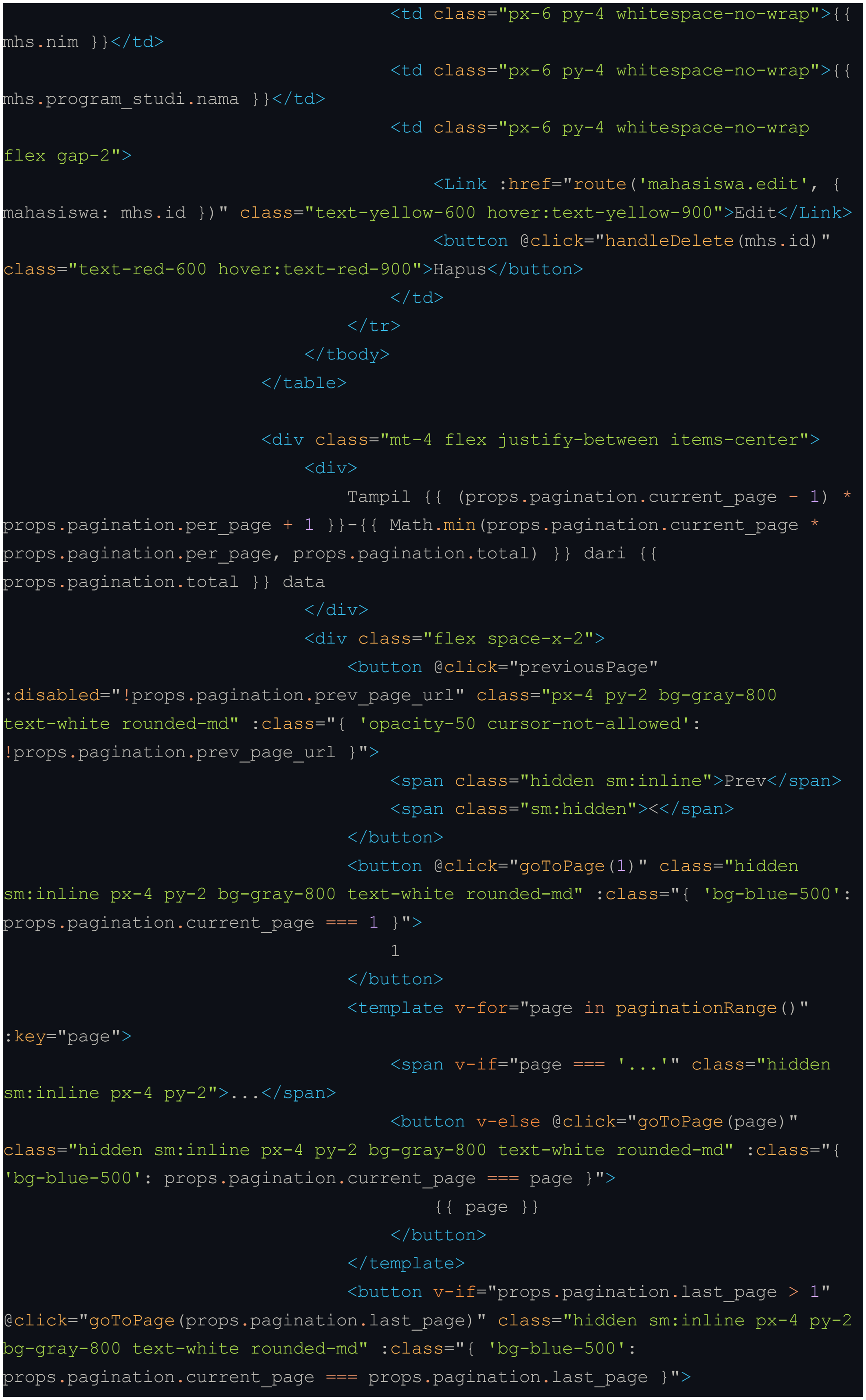


```
    router.get(route('mahasiswa.index', {
        search: search.value,
        page: props.pagination.current_page,
        sort: sort.value.column,
        direction: sort.value.direction
    }), {
        preserveState: true,
        replace: true,
    });
};

const getSortIcon = (column) => {
    if (sort.value.column === column) {
        return sort.value.direction === 'asc' ? '▲' : '▼';
    }
    return '⇅';
};

const page = usePage();
const message = ref('');

watch(() => page.props.flash.message, (newMessage) => {
    message.value = newMessage;
    if (newMessage) {
        setTimeout(() => {
            message.value = '';
        }, 5000);
    }
}, { immediate: true });
</script>

<template>
    <Head title="Mahasiswa" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800
leading-tight">Mahasiswa</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="message" class="mb-4 bg-green-100 border
border-green-400 text-green-700 px-4 py-3 rounded relative" role="alert">
                    <span class="block sm:inline">{{ message }}</span>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <div class="flex justify-between mb-2">
```

```
                            <div class="space-x-2">
                                <Link :href="route('mahasiswa.create')"
class="inline-flex items-center px-4 py-2 bg-gray-800 border border-transparent
rounded-md font-semibold text-xs text-white uppercase tracking-widest
hover:bg-gray-700 focus:bg-gray-700 active:bg-gray-900 focus:outline-none
focus:ring-2 focus:ring-indigo-500 focus:ring-offset-2 transition ease-in-out
duration-150">
                                    <span class="hidden sm:inline">Tambah
Mahasiswa</span>
                                    <span class="sm:hidden">+</span>
                                </Link>

                            </div>
                            <div class="relative">
                                <input
                                    v-model="search"
                                    type="text"
                                    placeholder="Cari mahasiswa..."
                                    class="border rounded px-2 py-1
focus:outline-none focus:ring-2 focus:ring-blue-500"
                                />
                                <svg
                                    class="h-5 w-5 text-gray-400 absolute
right-2 top-1/2 transform -translate-y-1/2"
                                    xmlns="http://www.w3.org/2000/svg"
                                    fill="none"
                                    viewBox="0 0 24 24"
                                    stroke="currentColor"
                                >
                                    <path
                                        stroke-linecap="round"
                                        stroke-linejoin="round"
                                        stroke-width="2"
                                        d="M21 21l-6-6m2-5a7 7 0 11-14 0 7 7 0
0114 0z"
                                    />
                                </svg>
                            </div>
                        </div>
                        <div class="flex justify-between mb-2">
                            <span v-if="selectedCount > 0" class="text-sm
text-gray-600">
                                {{ selectedCount }} item terpilih
                            </span>
                            <button v-if="selectedCount > 0"
@click="handleBulkDelete" class="px-4 py-1 bg-red-600 text-white rounded-md
hover:bg-red-700">
                                Hapus Data
                            </button>
```



```
                        </div>

                        <div v-if="props.mahasiswa.length === 0"
class="text-center py-4 text-gray-600">
                            Tidak ada data yang sesuai.
                        </div>

                        <table v-else class="min-w-full divide-y
divide-gray-200">
                            <thead>
                                <tr>
                                    <th class="px-6 py-3 bg-gray-50 text-left">
                                        <input class="rounded-[4px]"
type="checkbox" @change="toggleSelectAll" :checked="isAllSelected">
                                    </th>
                                    <th @click="handleSort('nama')" class="px-6
py-3 bg-gray-50 text-left text-xs leading-4 font-medium text-gray-500 uppercase
tracking-wider cursor-pointer">
                                        Nama
                                        <span class="ml-1">{{
getSortIcon('nama') }}</span>
                                    </th>
                                    <th @click="handleSort('nim')" class="px-6
py-3 bg-gray-50 text-left text-xs leading-4 font-medium text-gray-500 uppercase
tracking-wider cursor-pointer">
                                        NIM
                                        <span class="ml-1">{{
getSortIcon('nim') }}</span>
                                    </th>
                                    <th @click="handleSort('program_studi_id')"
class="px-6 py-3 bg-gray-50 text-left text-xs leading-4 font-medium
text-gray-500 uppercase tracking-wider cursor-pointer">
                                        Program Studi
                                        <span class="ml-1">{{
getSortIcon('program_studi_id') }}</span>
                                    </th>
                                    <th class="px-6 py-3 bg-gray-50 text-left
text-xs leading-4 font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
                                </tr>
                            </thead>
                            <tbody class="bg-white divide-y divide-gray-200">
                                <tr v-for="mhs in props.mahasiswa"
:key="mhs.id">
                                    <td class="px-6 py-4 whitespace-no-wrap">
                                        <input class="rounded-[4px]"
type="checkbox" v-model="selectedItems" :value="mhs.id">
                                    </td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
mhs.nama }}</td>
```

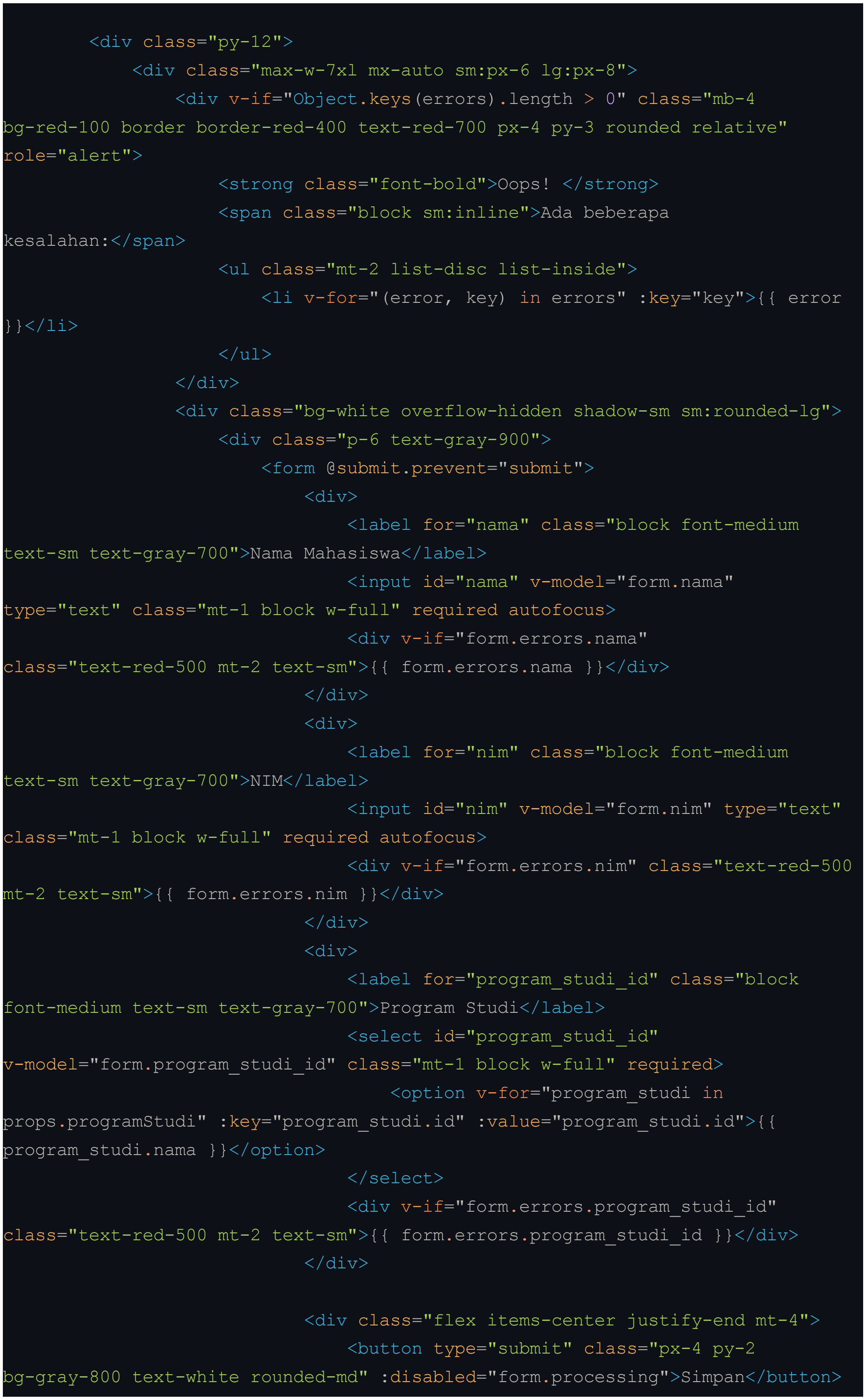


```
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
mhs.nim }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
mhs.program_studi.nama }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap
flex gap-2">
                                        <Link :href="route('mahasiswa.edit', {
mahasiswa: mhs.id })" class="text-yellow-600 hover:text-yellow-900">Edit</Link>
                                        <button @click="handleDelete(mhs.id)"
class="text-red-600 hover:text-red-900">Hapus</button>
                                    </td>
                                </tr>
                            </tbody>
                        </table>

                        <div class="mt-4 flex justify-between items-center">
                            <div>
                                Tampil {{ (props.pagination.current_page - 1) *
props.pagination.per_page + 1 }}-{{ Math.min(props.pagination.current_page *
props.pagination.per_page, props.pagination.total) }} dari {{
props.pagination.total }} data
                            </div>
                            <div class="flex space-x-2">
                                <button @click="previousPage"
:disabled="!props.pagination.prev_page_url" class="px-4 py-2 bg-gray-800
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.prev_page_url }">
                                    <span class="hidden sm:inline">Prev</span>
                                    <span class="sm:hidden"><</span>
                                </button>
                                <button @click="goToPage(1)" class="hidden
sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === 1 }">
                                    1
                                </button>
                                <template v-for="page in paginationRange()"
:key="page">
                                    <span v-if="page === '...'" class="hidden
sm:inline px-4 py-2">...</span>
                                    <button v-else @click="goToPage(page)"
class="hidden sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{
'bg-blue-500': props.pagination.current_page === page }">
                                        {{ page }}
                                    </button>
                                </template>
                                <button v-if="props.pagination.last_page > 1"
@click="goToPage(props.pagination.last_page)" class="hidden sm:inline px-4 py-2
bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === props.pagination.last_page }">
```

```
                                    {{ props.pagination.last_page }}
                                </button>
                                <button @click="nextPage"
:disabled="!props.pagination.next_page_url" class="px-4 py-2 bg-gray-800
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.next_page_url }">
                                    <span class="hidden sm:inline">Next</span>
                                    <span class="sm:hidden">></span>
                                </button>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

q. resources/js/Pages/Mahasiswa/Create.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';

const props = defineProps({
    programStudi: Array,
});

const form = useForm({
    nama: '',
    nim: '',
    program_studi_id: '',
});

const submit = () => {
    form.post(route('mahasiswa.store'));
};

const errors = computed(() => usePage().props.errors);
</script>

<template>
    <Head title="Tambah Mahasiswa" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800
leading-tight">Tambah Mahasiswa</h2>
        </template>
```



```
        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
                                <label for="nama" class="block font-medium
text-sm text-gray-700">Nama Mahasiswa</label>
                                <input id="nama" v-model="form.nama"
type="text" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.nama"
class="text-red-500 mt-2 text-sm">{{ form.errors.nama }}</div>
                            </div>
                            <div>
                                <label for="nim" class="block font-medium
text-sm text-gray-700">NIM</label>
                                <input id="nim" v-model="form.nim" type="text"
class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.nim" class="text-red-500
mt-2 text-sm">{{ form.errors.nim }}</div>
                            </div>
                            <div>
                                <label for="program_studi_id" class="block
font-medium text-sm text-gray-700">Program Studi</label>
                                <select id="program_studi_id"
v-model="form.program_studi_id" class="mt-1 block w-full" required>
                                    <option v-for="program_studi in
props.programStudi" :key="program_studi.id" :value="program_studi.id">{{
program_studi.nama }}</option>
                                </select>
                                <div v-if="form.errors.program_studi_id"
class="text-red-500 mt-2 text-sm">{{ form.errors.program_studi_id }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Simpan</button>
```

```
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

r. resources/js/Pages/Mahasiswa/Edit.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';

const props = defineProps({
    mahasiswa: Object,
    programStudi: Array,
});

const form = useForm({
    nama: props.mahasiswa.nama,
    nim: props.mahasiswa.nim,
    program_studi_id: props.mahasiswa.program_studi_id,
});

const submit = () => {
    form.put(route('mahasiswa.update', props.mahasiswa.id));
};

const errors = computed(() => usePage().props.errors);
</script>

<template>
    <Head title="Edit Mahasiswa" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800 leading-tight">Edit
Mahasiswa</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
                                <label for="nama" class="block font-medium
text-sm text-gray-700">Nama Mahasiswa</label>
                                <input id="nama" v-model="form.nama"
type="text" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.nama"
class="text-red-500 mt-2 text-sm">{{ form.errors.nama }}</div>
                            </div>
                            <div>
                                <label for="nim" class="block font-medium
text-sm text-gray-700">NIM</label>
                                <input id="nim" v-model="form.nim" type="text"
class="mt-1 block w-full" required>
                                <div v-if="form.errors.nim" class="text-red-500
mt-2 text-sm">{{ form.errors.nim }}</div>
                            </div>
                            <div>
                                <label for="program_studi_id" class="block
font-medium text-sm text-gray-700">Program Studi</label>
                                <select id="program_studi_id"
v-model="form.program_studi_id" class="mt-1 block w-full" required>
                                    <option v-for="program_studi in
props.programStudi" :key="program_studi.id" :value="program_studi.id">{{
program_studi.nama }}</option>
                                </select>
                                <div v-if="form.errors.program_studi_id"
class="text-red-500 mt-2 text-sm">{{ form.errors.program_studi_id }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Update</button>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
```

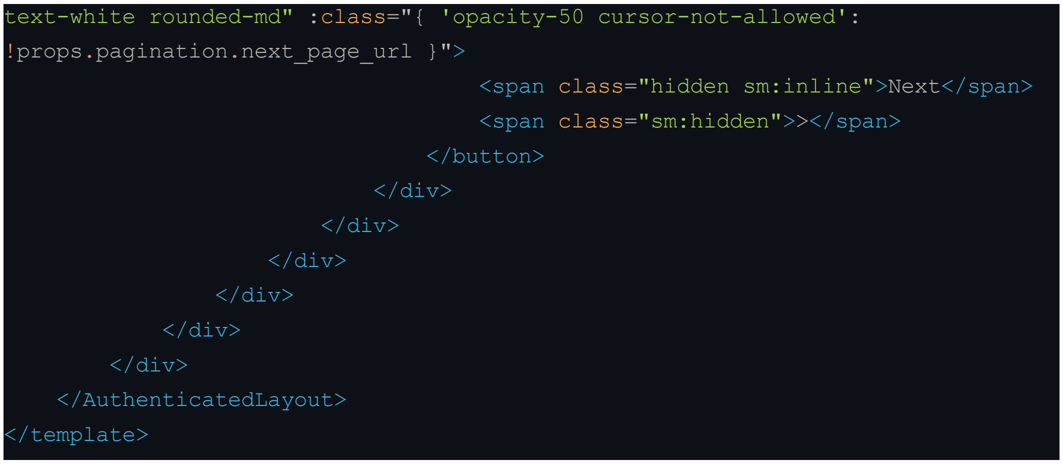
```
</template>
```

s. resources/js/Pages/MataKuliah/Index.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, Link, useForm, router, usePage } from '@inertiajs/vue3';
import { ref, defineProps, computed, watch, onMounted } from 'vue';

const props = defineProps({
    mataKuliah: Array,
    pagination: {
        type: Object,
        required: true,
    },
    search: String,
    sort: Object,
});

const handleDelete = (id) => {
  if (window.confirm('Apakah Anda yakin ingin menghapus data ini?')) {
    router.delete(route('mata-kuliah.destroy', { mata_kuliah: id }));
  }
};

const selectedItems = ref([]);

const form = useForm({
    ids: [],
});

const handleBulkDelete = () => {
    if (window.confirm('Apakah Anda yakin ingin menghapus data yang dipilih?'))
{
        form.ids = selectedItems.value;
        form.delete(route('mata-kuliah.bulk-destroy'), {
            preserveScroll: true,
            preserveState: true,
            onSuccess: () => {
                selectedItems.value = [];
            },
        });
    }
};

const toggleSelectAll = (event) => {
    if (event.target.checked) {
        selectedItems.value = props.mataKuliah.map(mata_kuliah =>
mata_kuliah.id);
    } else {
        selectedItems.value = [];
    }
};

const isAllSelected = computed(() => {
    return selectedItems.value.length === props.mataKuliah.length;
});

const selectedCount = computed(() => {
    return selectedItems.value.length;
});

const previousPage = () => {
    if (props.pagination.prev_page_url) {
        router.get(route('mata-kuliah.index', { search: search.value, page:
props.pagination.current_page - 1 }));
    }
};

const nextPage = () => {
    if (props.pagination.next_page_url) {
        router.get(route('mata-kuliah.index', { search: search.value, page:
props.pagination.current_page + 1 }));
    }
};

const goToPage = (page) => {
    router.get(route('mata-kuliah.index', {
        search: search.value,
        page: page,
        sort: sort.value.column,
        direction: sort.value.direction,
    }));
};

const paginationRange = () => {
    const current = props.pagination.current_page;
    const last = props.pagination.last_page;
    const delta = 2;
    const range = [];
    const rangeWithDots = [];

    for (let i = Math.max(2, current - delta); i <= Math.min(last - 1, current
+ delta); i++) {
        range.push(i);
    }

    if (current - delta > 2) rangeWithDots.push('...');
    rangeWithDots.push(...range);
    if (current + delta < last - 1) rangeWithDots.push('...');

    return rangeWithDots;
};

const search = ref(props.search || '');
let searchTimer = null;

const sort = ref(props.sort || { column: null, direction: 'asc' });

const performSearch = () => {
    clearTimeout(searchTimer);
    searchTimer = setTimeout(() => {
        router.get(route('mata-kuliah.index', {
            search: search.value,
            page: 1,
            sort: sort.value.column,
            direction: sort.value.direction
        }), {
            preserveState: true,
            replace: true,
        });
    }, 300);
};

watch(search, performSearch);

const handleSort = (column) => {
    if (sort.value.column === column) {
        if (sort.value.direction === 'asc') {
            sort.value.direction = 'desc';
        } else if (sort.value.direction === 'desc') {
            sort.value.column = null;
            sort.value.direction = 'asc';
        } else {
            sort.value.direction = 'asc';
        }
    } else {
        sort.value.column = column;
        sort.value.direction = 'asc';
    }
    router.get(route('mata-kuliah.index', {
        search: search.value,
        page: 1,
        sort: sort.value.column,
        direction: sort.value.direction
    }), {
        preserveState: true,
        replace: true,
    });
};

const getSortIcon = (column) => {
    if (sort.value.column === column) {
        return sort.value.direction === 'asc' ? '▲' : '▼';
    }
    return '⇅';
};

const page = usePage();
const message = ref('');

watch(() => page.props.flash.message, (newMessage) => {
    message.value = newMessage;
    if (newMessage) {
        setTimeout(() => {
            message.value = '';
        }, 5000);
    }
}, { immediate: true });
</script>

<template>
    <Head title="Mata Kuliah" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800 leading-tight">Mata
Kuliah</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="message" class="mb-4 bg-green-100 border
border-green-400 text-green-700 px-4 py-3 rounded relative" role="alert">
                    <span class="block sm:inline">{{ message }}</span>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <div class="flex justify-between mb-2">
                            <div class="space-x-2">
                                <Link :href="route('mata-kuliah.create')"
class="inline-flex items-center px-4 py-2 bg-gray-800 border border-transparent
rounded-md font-semibold text-xs text-white uppercase tracking-widest
hover:bg-gray-700 focus:bg-gray-700 active:bg-gray-900 focus:outline-none
focus:ring-2 focus:ring-indigo-500 focus:ring-offset-2 transition ease-in-out
duration-150">
```

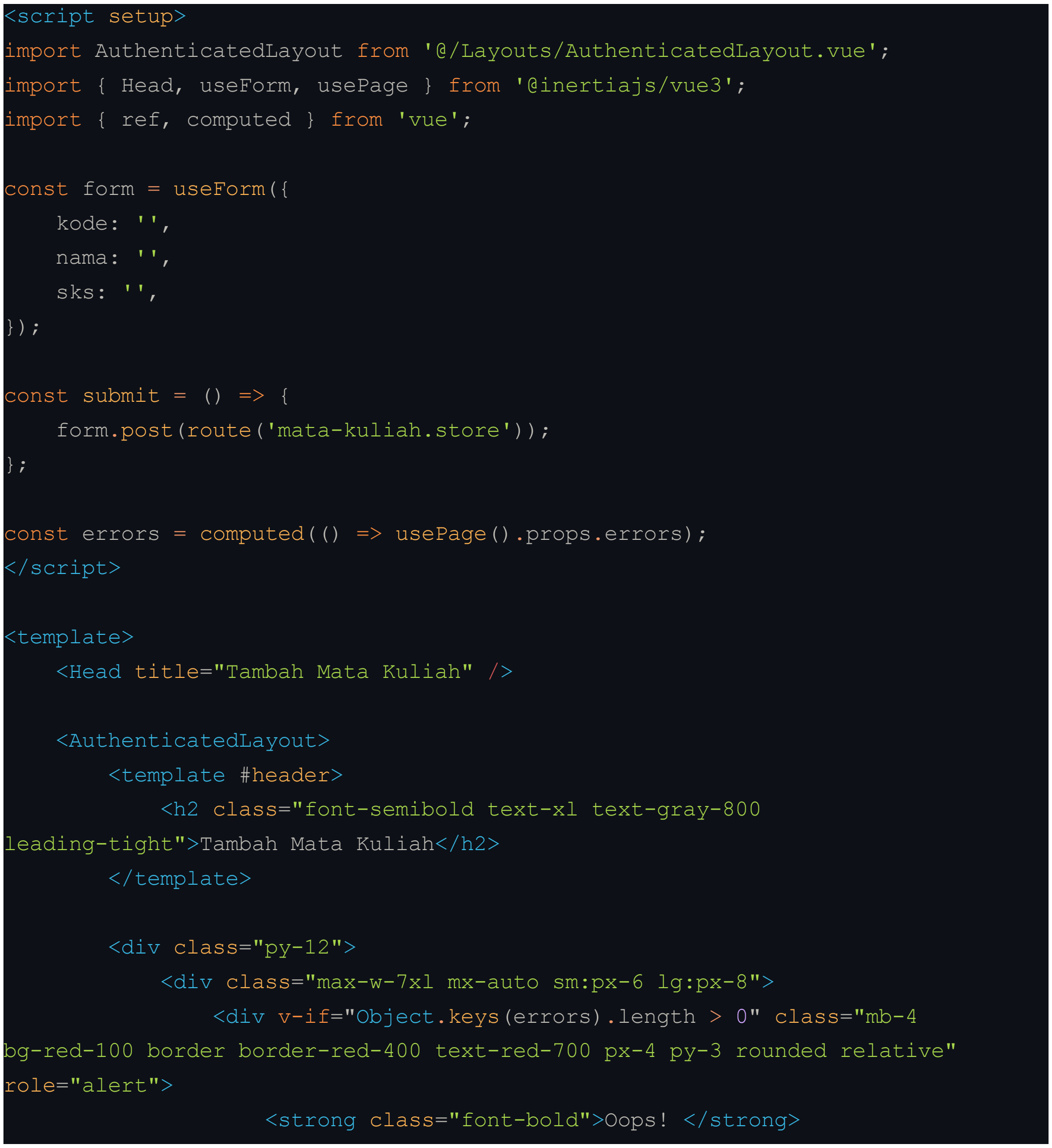
```
                                    <span class="hidden sm:inline">Tambah Mata
Kuliah</span>
                                    <span class="sm:hidden">+</span>
                                </Link>

                            </div>
                            <div class="relative">
                                <input
                                    v-model="search"
                                    type="text"
                                    placeholder="Cari mata kuliah..."
                                    class="border rounded px-2 py-1
focus:outline-none focus:ring-2 focus:ring-blue-500"
                                />
                                <svg
                                    class="h-5 w-5 text-gray-400 absolute
right-2 top-1/2 transform -translate-y-1/2"
                                    xmlns="http://www.w3.org/2000/svg"
                                    fill="none"
                                    viewBox="0 0 24 24"
                                    stroke="currentColor"
                                >
                                    <path
                                        stroke-linecap="round"
                                        stroke-linejoin="round"
                                        stroke-width="2"
                                        d="M21 21l-6-6m2-5a7 7 0 11-14 0 7 7 0
0114 0z"
                                    />
                                </svg>
                            </div>
                        </div>
                        <div class="flex justify-between mb-2">
                            <span v-if="selectedCount > 0" class="text-sm
text-gray-600">
                                {{ selectedCount }} item terpilih
                            </span>
                            <button v-if="selectedCount > 0"
@click="handleBulkDelete" class="px-4 py-1 bg-red-600 text-white rounded-md
hover:bg-red-700">
                                Hapus Data
                            </button>
                        </div>

                        <div v-if="props.mataKuliah.length === 0"
class="text-center py-4 text-gray-600">
                            Tidak ada data yang sesuai.
                        </div>
                        <table v-else class="min-w-full divide-y
divide-gray-200">
                            <thead>
                                <tr>
                                    <th class="px-6 py-3 bg-gray-50 text-left">
                                        <input class="rounded-[4px]"
type="checkbox" @change="toggleSelectAll" :checked="isAllSelected">
                                    </th>
                                    <th @click="handleSort('kode')" class="px-6
py-3 bg-gray-50 text-left text-xs leading-4 font-medium text-gray-500 uppercase
tracking-wider cursor-pointer">
                                        Kode
                                        <span class="ml-1">{{
getSortIcon('kode') }}</span>
                                    </th>
                                    <th @click="handleSort('nama')" class="px-6
py-3 bg-gray-50 text-left text-xs leading-4 font-medium text-gray-500 uppercase
tracking-wider cursor-pointer">
                                        Nama
                                        <span class="ml-1">{{
getSortIcon('nama') }}</span>
                                    </th>
                                    <th @click="handleSort('sks')" class="px-6
py-3 bg-gray-50 text-left text-xs leading-4 font-medium text-gray-500 uppercase
tracking-wider cursor-pointer">
                                        SKS
                                        <span class="ml-1">{{
getSortIcon('sks') }}</span>
                                    </th>
                                    <th class="px-6 py-3 bg-gray-50 text-left
text-xs leading-4 font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
                                </tr>
                            </thead>
                            <tbody class="bg-white divide-y divide-gray-200">
                                <tr v-for="mata_kuliah in props.mataKuliah"
:key="mata_kuliah.id">
                                    <td class="px-6 py-4 whitespace-no-wrap">
                                        <input class="rounded-[4px]"
type="checkbox" v-model="selectedItems" :value="mata_kuliah.id">
                                    </td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
mata_kuliah.kode }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
mata_kuliah.nama }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
mata_kuliah.sks }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap
flex gap-2">
```

```
                                        <Link :href="route('mata-kuliah.edit',
{ mata_kuliah: mata_kuliah.id })" class="text-yellow-600
hover:text-yellow-900">Edit</Link>
                                        <button
@click="handleDelete(mata_kuliah.id)" class="text-red-600
hover:text-red-900">Hapus</button>
                                    </td>
                                </tr>
                            </tbody>
                        </table>

                        <div class="mt-4 flex justify-between items-center">
                            <div>
                                Tampil {{ (props.pagination.current_page - 1) *
props.pagination.per_page + 1 }}-{{ Math.min(props.pagination.current_page *
props.pagination.per_page, props.pagination.total) }} dari {{
props.pagination.total }} data
                            </div>
                            <div class="flex space-x-2">
                                <button @click="previousPage"
:disabled="!props.pagination.prev_page_url" class="px-4 py-2 bg-gray-800
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.prev_page_url }">
                                    <span class="hidden sm:inline">Prev</span>
                                    <span class="sm:hidden"><</span>
                                </button>
                                <button @click="goToPage(1)" class="hidden
sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === 1 }">
                                    1
                                </button>
                                <template v-for="page in paginationRange()"
:key="page">
                                    <span v-if="page === '...'" class="hidden
sm:inline px-4 py-2">...</span>
                                    <button v-else @click="goToPage(page)"
class="hidden sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{
'bg-blue-500': props.pagination.current_page === page }">
                                        {{ page }}
                                    </button>
                                </template>
                                <button v-if="props.pagination.last_page > 1"
@click="goToPage(props.pagination.last_page)" class="hidden sm:inline px-4 py-2
bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === props.pagination.last_page }">
                                    {{ props.pagination.last_page }}
                                </button>
                                <button @click="nextPage"
:disabled="!props.pagination.next_page_url" class="px-4 py-2 bg-gray-800
```



```
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.next_page_url }">
                                    <span class="hidden sm:inline">Next</span>
                                    <span class="sm:hidden">></span>
                                </button>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

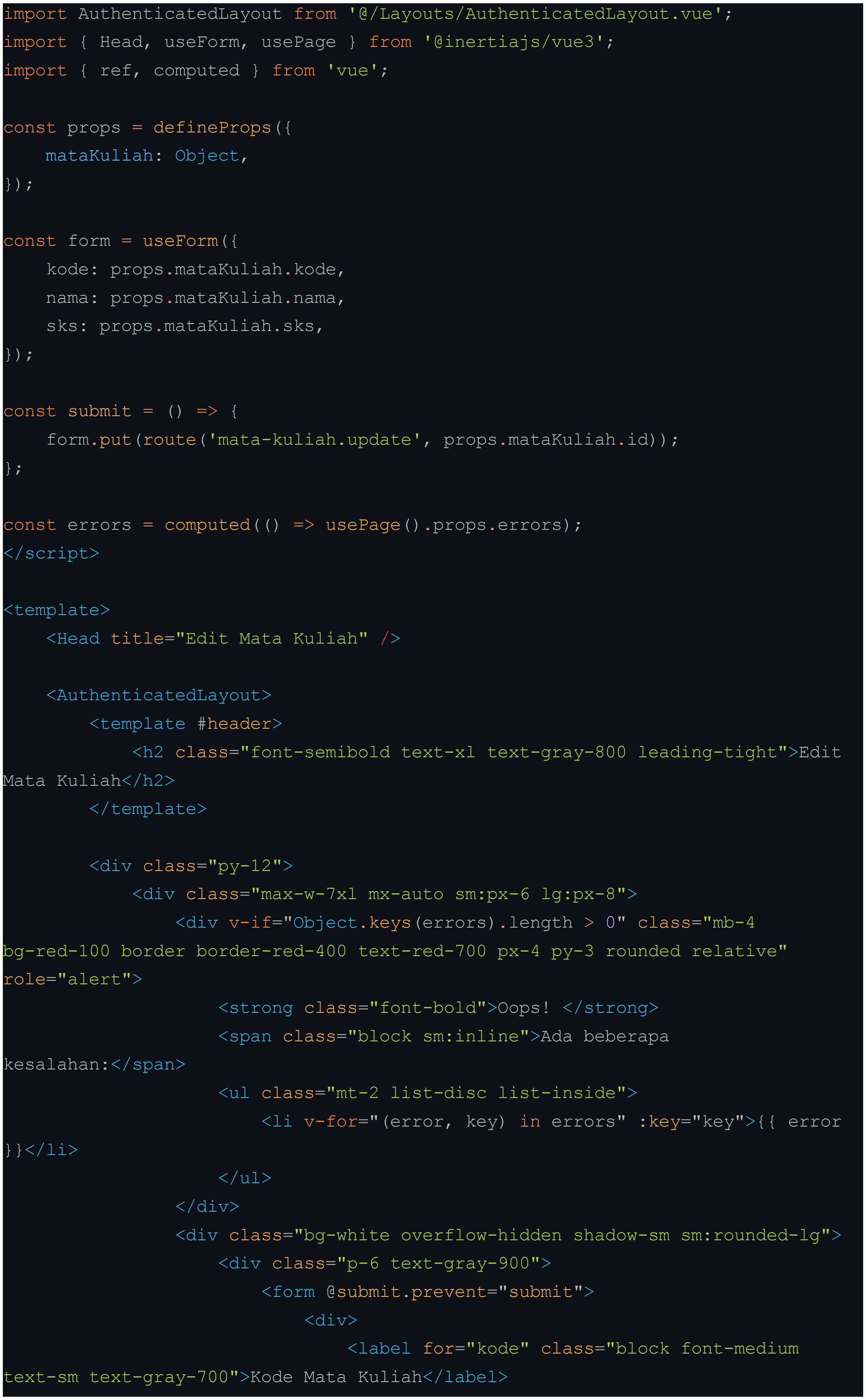
t. resources/js/Pages/MataKuliah/Create.vue



```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';

const form = useForm({
    kode: '',
    nama: '',
    sks: '',
});

const submit = () => {
    form.post(route('mata-kuliah.store'));
};

const errors = computed(() => usePage().props.errors);
</script>

<template>
    <Head title="Tambah Mata Kuliah" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800
leading-tight">Tambah Mata Kuliah</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
                                <label for="kode" class="block font-medium
text-sm text-gray-700">Kode Mata Kuliah</label>
                                <input id="kode" v-model="form.kode"
type="text" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.kode"
class="text-red-500 mt-2 text-sm">{{ form.errors.kode }}</div>
                            </div>
                            <div>
                                <label for="nama" class="block font-medium
text-sm text-gray-700">Nama Mata Kuliah</label>
                                <input id="nama" v-model="form.nama"
type="text" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.nama"
class="text-red-500 mt-2 text-sm">{{ form.errors.nama }}</div>
                            </div>
                            <div>
                                <label for="sks" class="block font-medium
text-sm text-gray-700">SKS</label>
                                <input id="sks" v-model="form.sks"
type="number" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.sks" class="text-red-500
mt-2 text-sm">{{ form.errors.sks }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Simpan</button>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

u. resources/js/Pages/MataKuliah/Edit.vue

```
<script setup>
```



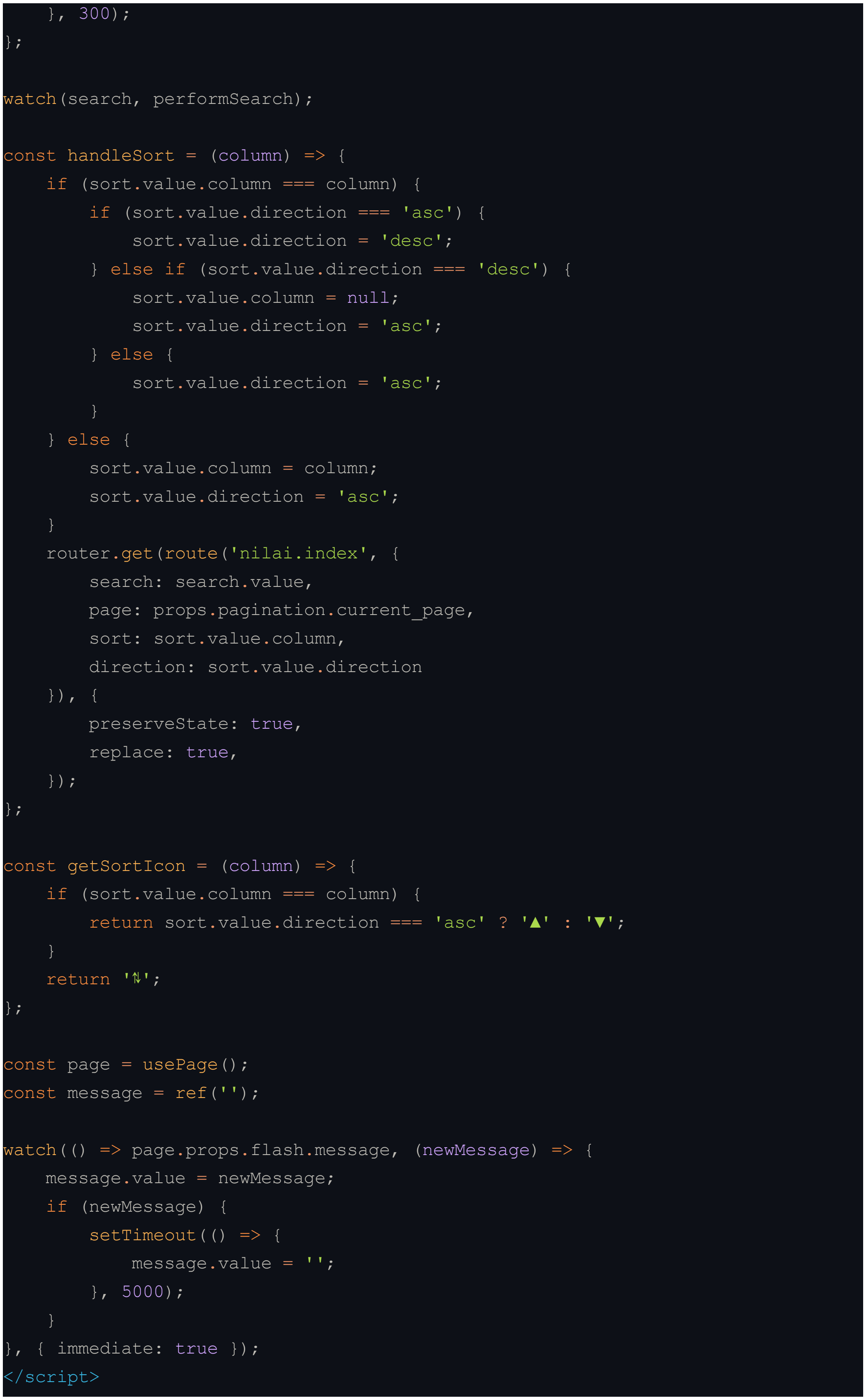
```
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';

const props = defineProps({
    mataKuliah: Object,
});

const form = useForm({
    kode: props.mataKuliah.kode,
    nama: props.mataKuliah.nama,
    sks: props.mataKuliah.sks,
});

const submit = () => {
    form.put(route('mata-kuliah.update', props.mataKuliah.id));
};

const errors = computed(() => usePage().props.errors);
</script>

<template>
    <Head title="Edit Mata Kuliah" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800 leading-tight">Edit
Mata Kuliah</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
                                <label for="kode" class="block font-medium
text-sm text-gray-700">Kode Mata Kuliah</label>
```

```
                                <input id="kode"
v-model="props.mataKuliah.kode" type="text" class="mt-1 block w-full" required
autofocus>
                                <div v-if="form.errors.kode"
class="text-red-500 mt-2 text-sm">{{ form.errors.kode }}</div>
                            </div>
                            <div>
                                <label for="nama" class="block font-medium
text-sm text-gray-700">Nama Mata Kuliah</label>
                                <input id="nama" v-model="form.nama"
type="text" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.nama"
class="text-red-500 mt-2 text-sm">{{ form.errors.nama }}</div>
                            </div>
                            <div>
                                <label for="sks" class="block font-medium
text-sm text-gray-700">SKS</label>
                                <input id="sks" v-model="form.sks"
type="number" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.sks" class="text-red-500
mt-2 text-sm">{{ form.errors.sks }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Update</button>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

v. resources/js/Pages/Nilai/Index.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, Link, useForm, router, usePage } from '@inertiajs/vue3';
import { ref, defineProps, computed, watch, onMounted } from 'vue';

const props = defineProps({
    nilai: Array,
    pagination: {
        type: Object,
        required: true,
    },
    search: String,
    sort: Object,
});

const handleDelete = (id) => {
  if (window.confirm('Apakah Anda yakin ingin menghapus data ini?')) {
    router.delete(route('nilai.destroy', { nilai: id }));
  }
};

const selectedItems = ref([]);

const form = useForm({
    ids: [],
});

const handleBulkDelete = () => {
    if (window.confirm('Apakah Anda yakin ingin menghapus data yang dipilih?'))
{
        form.ids = selectedItems.value;
        form.delete(route('nilai.bulk-destroy'), {
            preserveScroll: true,
            preserveState: true,
            onSuccess: () => {
                selectedItems.value = [];
            },
        });
    }
};

const toggleSelectAll = (event) => {
    if (event.target.checked) {
        selectedItems.value = props.nilai.map(nilai => nilai.id);
    } else {
        selectedItems.value = [];
    }
};

const isAllSelected = computed(() => {
    return selectedItems.value.length === props.nilai.length;
});

const selectedCount = computed(() => {
    return selectedItems.value.length;
});

const previousPage = () => {
    if (props.pagination.prev_page_url) {
        router.get(route('nilai.index', { search: search.value, page:
props.pagination.current_page - 1 }));
    }
```

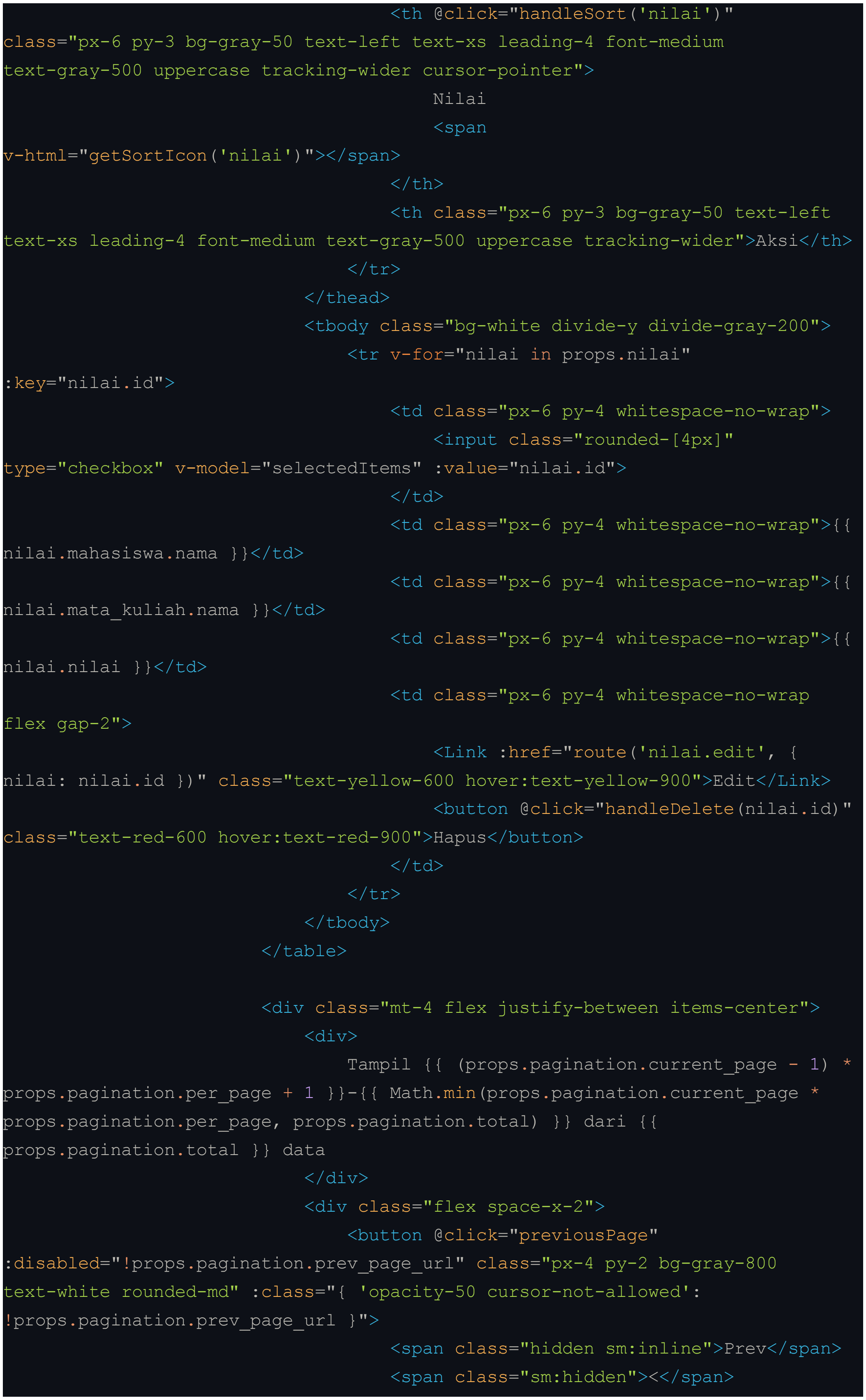
```
};

const nextPage = () => {
    if (props.pagination.next_page_url) {
        router.get(route('nilai.index', { search: search.value, page:
props.pagination.current_page + 1 }));
    }
};

const goToPage = (page) => {
    router.get(route('nilai.index', { search: search.value, page: page }));
};

const paginationRange = () => {
    const current = props.pagination.current_page;
    const last = props.pagination.last_page;
    const delta = 2;
    const range = [];
    const rangeWithDots = [];

    for (let i = Math.max(2, current - delta); i <= Math.min(last - 1, current
+ delta); i++) {
        range.push(i);
    }

    if (current - delta > 2) rangeWithDots.push('...');
    rangeWithDots.push(...range);
    if (current + delta < last - 1) rangeWithDots.push('...');

    return rangeWithDots;
};

const search = ref(props.search || '');
let searchTimer = null;

const sort = ref(props.sort || { column: null, direction: 'asc' });

const performSearch = () => {
    clearTimeout(searchTimer);
    searchTimer = setTimeout(() => {
        router.get(route('nilai.index', {
            search: search.value,
            page: 1,
            sort: sort.value.column,
            direction: sort.value.direction
        }), {
            preserveState: true,
            replace: true,
        });
```



```
    }, 300);
};

watch(search, performSearch);

const handleSort = (column) => {
    if (sort.value.column === column) {
        if (sort.value.direction === 'asc') {
            sort.value.direction = 'desc';
        } else if (sort.value.direction === 'desc') {
            sort.value.column = null;
            sort.value.direction = 'asc';
        } else {
            sort.value.direction = 'asc';
        }
    } else {
        sort.value.column = column;
        sort.value.direction = 'asc';
    }
    router.get(route('nilai.index', {
        search: search.value,
        page: props.pagination.current_page,
        sort: sort.value.column,
        direction: sort.value.direction
    }), {
        preserveState: true,
        replace: true,
    });
};

const getSortIcon = (column) => {
    if (sort.value.column === column) {
        return sort.value.direction === 'asc' ? '▲' : '▼';
    }
    return '⇅';
};

const page = usePage();
const message = ref('');

watch(() => page.props.flash.message, (newMessage) => {
    message.value = newMessage;
    if (newMessage) {
        setTimeout(() => {
            message.value = '';
        }, 5000);
    }
}, { immediate: true });
</script>
```

```
<template>
    <Head title="Nilai" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800
leading-tight">Nilai</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="message" class="mb-4 bg-green-100 border
border-green-400 text-green-700 px-4 py-3 rounded relative" role="alert">
                    <span class="block sm:inline">{{ message }}</span>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <div class="flex justify-between mb-2">
                            <div class="space-x-2">
                                <Link :href="route('nilai.create')"
class="inline-flex items-center px-4 py-2 bg-gray-800 border border-transparent
rounded-md font-semibold text-xs text-white uppercase tracking-widest
hover:bg-gray-700 focus:bg-gray-700 active:bg-gray-900 focus:outline-none
focus:ring-2 focus:ring-indigo-500 focus:ring-offset-2 transition ease-in-out
duration-150">
                                    <span class="hidden sm:inline">Tambah
Nilai</span>
                                    <span class="sm:hidden">+</span>
                                </Link>

                            </div>
                            <div class="relative">
                                <input
                                    v-model="search"
                                    type="text"
                                    placeholder="Cari nilai..."
                                    class="border rounded px-2 py-1
focus:outline-none focus:ring-2 focus:ring-blue-500"
                                />
                                <svg
                                    class="h-5 w-5 text-gray-400 absolute
right-2 top-1/2 transform -translate-y-1/2"
                                    xmlns="http://www.w3.org/2000/svg"
                                    fill="none"
                                    viewBox="0 0 24 24"
                                    stroke="currentColor"
                                >
                                    <path
```

```
                                        stroke-linecap="round"
                                        stroke-linejoin="round"
                                        stroke-width="2"
                                        d="M21 21l-6-6m2-5a7 7 0 11-14 0 7 7 0
0114 0z"
                                    />
                                </svg>
                            </div>
                        </div>
                        <div class="flex justify-between mb-2">
                            <span v-if="selectedCount > 0" class="text-sm
text-gray-600">
                                {{ selectedCount }} item terpilih
                            </span>
                            <button v-if="selectedCount > 0"
@click="handleBulkDelete" class="px-4 py-1 bg-red-600 text-white rounded-md
hover:bg-red-700">
                                Hapus Data
                            </button>
                        </div>

                        <div v-if="props.nilai.length === 0" class="text-center
py-4 text-gray-600">
                            Tidak ada data yang sesuai.
                        </div>

                        <table v-else class="min-w-full divide-y
divide-gray-200">
                            <thead>
                                <tr>
                                    <th class="px-6 py-3 bg-gray-50 text-left">
                                        <input class="rounded-[4px]"
type="checkbox" @change="toggleSelectAll" :checked="isAllSelected">
                                    </th>
                                    <th @click="handleSort('mahasiswa_id')"
class="px-6 py-3 bg-gray-50 text-left text-xs leading-4 font-medium
text-gray-500 uppercase tracking-wider cursor-pointer">
                                        Nama Mahasiswa
                                        <span
v-html="getSortIcon('mahasiswa_id')"></span>
                                    </th>
                                    <th @click="handleSort('mata_kuliah_id')"
class="px-6 py-3 bg-gray-50 text-left text-xs leading-4 font-medium
text-gray-500 uppercase tracking-wider cursor-pointer">
                                        Nama Mata Kuliah
                                        <span
v-html="getSortIcon('mata_kuliah_id')"></span>
                                    </th>
```



```
                                    <th @click="handleSort('nilai')"
class="px-6 py-3 bg-gray-50 text-left text-xs leading-4 font-medium
text-gray-500 uppercase tracking-wider cursor-pointer">
                                        Nilai
                                        <span
v-html="getSortIcon('nilai')"></span>
                                    </th>
                                    <th class="px-6 py-3 bg-gray-50 text-left
text-xs leading-4 font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
                                </tr>
                            </thead>
                            <tbody class="bg-white divide-y divide-gray-200">
                                <tr v-for="nilai in props.nilai"
:key="nilai.id">
                                    <td class="px-6 py-4 whitespace-no-wrap">
                                        <input class="rounded-[4px]"
type="checkbox" v-model="selectedItems" :value="nilai.id">
                                    </td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
nilai.mahasiswa.nama }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
nilai.mata_kuliah.nama }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap">{{
nilai.nilai }}</td>
                                    <td class="px-6 py-4 whitespace-no-wrap
flex gap-2">
                                        <Link :href="route('nilai.edit', {
nilai: nilai.id })" class="text-yellow-600 hover:text-yellow-900">Edit</Link>
                                        <button @click="handleDelete(nilai.id)"
class="text-red-600 hover:text-red-900">Hapus</button>
                                    </td>
                                </tr>
                            </tbody>
                        </table>

                        <div class="mt-4 flex justify-between items-center">
                            <div>
                                Tampil {{ (props.pagination.current_page - 1) *
props.pagination.per_page + 1 }}-{{ Math.min(props.pagination.current_page *
props.pagination.per_page, props.pagination.total) }} dari {{
props.pagination.total }} data
                            </div>
                            <div class="flex space-x-2">
                                <button @click="previousPage"
:disabled="!props.pagination.prev_page_url" class="px-4 py-2 bg-gray-800
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.prev_page_url }">
                                    <span class="hidden sm:inline">Prev</span>
                                    <span class="sm:hidden"><</span>
                                </button>
                                <button @click="goToPage(1)" class="hidden
sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === 1 }">
                                    1
                                </button>
                                <template v-for="page in paginationRange()"
:key="page">
                                    <span v-if="page === '...'" class="hidden
sm:inline px-4 py-2">...</span>
                                    <button v-else @click="goToPage(page)"
class="hidden sm:inline px-4 py-2 bg-gray-800 text-white rounded-md" :class="{
'bg-blue-500': props.pagination.current_page === page }">
                                        {{ page }}
                                    </button>
                                </template>
                                <button v-if="props.pagination.last_page > 1"
@click="goToPage(props.pagination.last_page)" class="hidden sm:inline px-4 py-2
bg-gray-800 text-white rounded-md" :class="{ 'bg-blue-500':
props.pagination.current_page === props.pagination.last_page }">
                                    {{ props.pagination.last_page }}
                                </button>
                                <button @click="nextPage"
:disabled="!props.pagination.next_page_url" class="px-4 py-2 bg-gray-800
text-white rounded-md" :class="{ 'opacity-50 cursor-not-allowed':
!props.pagination.next_page_url }">
                                    <span class="hidden sm:inline">Next</span>
                                    <span class="sm:hidden">></span>
                                </button>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

w. resources/js/Pages/Nilai/Create.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';
import SearchableSelect from '@/Components/SearchableSelect.vue';

const props = defineProps({
    mahasiswa: Array,
    mataKuliah: Array
});
```

```
const form = useForm({
    mahasiswa_id: '',
    mata_kuliah_id: '',
    nilai: '',
});

const submit = () => {
    form.post(route('nilai.store'));
};

const errors = computed(() => usePage().props.errors);

const mahasiswaOptions = computed(() =>
  props.mahasiswa.map(m => ({ value: m.id, label: `${m.nim} - ${m.nama}` }))
);

const mataKuliahOptions = computed(() =>
  props.mataKuliah.map(mk => ({ value: mk.id, label: `${mk.kode} - ${mk.nama}`
}))
);
</script>

<template>
    <Head title="Tambah Nilai" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800
leading-tight">Tambah Nilai</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
```

```
                                <SearchableSelect
                                    v-model="form.mahasiswa_id"
                                    :options="mahasiswaOptions"
                                    label="Mahasiswa"
                                    placeholder="Pilih mahasiswa"
                                />
                                <div v-if="form.errors.mahasiswa_id"
class="text-red-500 mt-2 text-sm">{{ form.errors.mahasiswa_id }}</div>
                            </div>
                            <div>
                                <SearchableSelect
                                    v-model="form.mata_kuliah_id"
                                    :options="mataKuliahOptions"
                                    label="Mata Kuliah"
                                    placeholder="Pilih mata kuliah"
                                />
                                <div v-if="form.errors.mata_kuliah_id"
class="text-red-500 mt-2 text-sm">{{ form.errors.mata_kuliah_id }}</div>
                            </div>
                            <div>
                                <label for="nilai" class="block font-medium
text-sm text-gray-700">Nilai</label>
                                <input id="nilai" v-model="form.nilai"
type="text" class="mt-1 block w-full" required autofocus>
                                <div v-if="form.errors.nilai"
class="text-red-500 mt-2 text-sm">{{ form.errors.nilai }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Simpan</button>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

x. resources/js/Pages/Nilai/Edit.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head, useForm, usePage } from '@inertiajs/vue3';
import { ref, computed } from 'vue';
import SearchableSelect from '@/Components/SearchableSelect.vue';

const props = defineProps({
    nilai: Object,
    mahasiswa: Array,
    mataKuliah: Array,
});

const form = useForm({
    mahasiswa_id: props.nilai.mahasiswa_id,
    mata_kuliah_id: props.nilai.mata_kuliah_id,
    nilai: props.nilai.nilai,
});

const submit = () => {
    form.put(route('nilai.update', props.nilai.id));
};

const errors = computed(() => usePage().props.errors);

const mahasiswaOptions = computed(() =>
  props.mahasiswa.map(m => ({ value: m.id, label: `${m.nim} - ${m.nama}` }))
);

const mataKuliahOptions = computed(() =>
  props.mataKuliah.map(mk => ({ value: mk.id, label: `${mk.kode} - ${mk.nama}`
}))
);
</script>

<template>
    <Head title="Edit Nilai" />

    <AuthenticatedLayout>
        <template #header>
            <h2 class="font-semibold text-xl text-gray-800 leading-tight">Edit
Nilai</h2>
        </template>

        <div class="py-12">
            <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
                <div v-if="Object.keys(errors).length > 0" class="mb-4
bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative"
role="alert">
                    <strong class="font-bold">Oops! </strong>
                    <span class="block sm:inline">Ada beberapa
kesalahan:</span>
                    <ul class="mt-2 list-disc list-inside">
                        <li v-for="(error, key) in errors" :key="key">{{ error
}}</li>
                    </ul>
                </div>
                <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
```

```
                    <div class="p-6 text-gray-900">
                        <form @submit.prevent="submit">
                            <div>
                                <SearchableSelect
                                    v-model="form.mahasiswa_id"
                                    :options="mahasiswaOptions"
                                    label="Mahasiswa"
                                    placeholder="Pilih mahasiswa"
                                />
                                <div v-if="form.errors.mahasiswa_id"
class="text-red-500 mt-2 text-sm">{{ form.errors.mahasiswa_id }}</div>
                            </div>
                            <div>
                                <SearchableSelect
                                    v-model="form.mata_kuliah_id"
                                    :options="mataKuliahOptions"
                                    label="Mata Kuliah"
                                    placeholder="Pilih mata kuliah"
                                />
                                <div v-if="form.errors.mata_kuliah_id"
class="text-red-500 mt-2 text-sm">{{ form.errors.mata_kuliah_id }}</div>
                            </div>
                            <div>
                                <label for="nilai" class="block font-medium
text-sm text-gray-700">Nilai</label>
                                <input id="nilai" v-model="form.nilai"
type="number" class="mt-1 block w-full" required>
                                <div v-if="form.errors.nilai"
class="text-red-500 mt-2 text-sm">{{ form.errors.nilai }}</div>
                            </div>

                            <div class="flex items-center justify-end mt-4">
                                <button type="submit" class="px-4 py-2
bg-gray-800 text-white rounded-md" :disabled="form.processing">Update</button>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </AuthenticatedLayout>
</template>
```

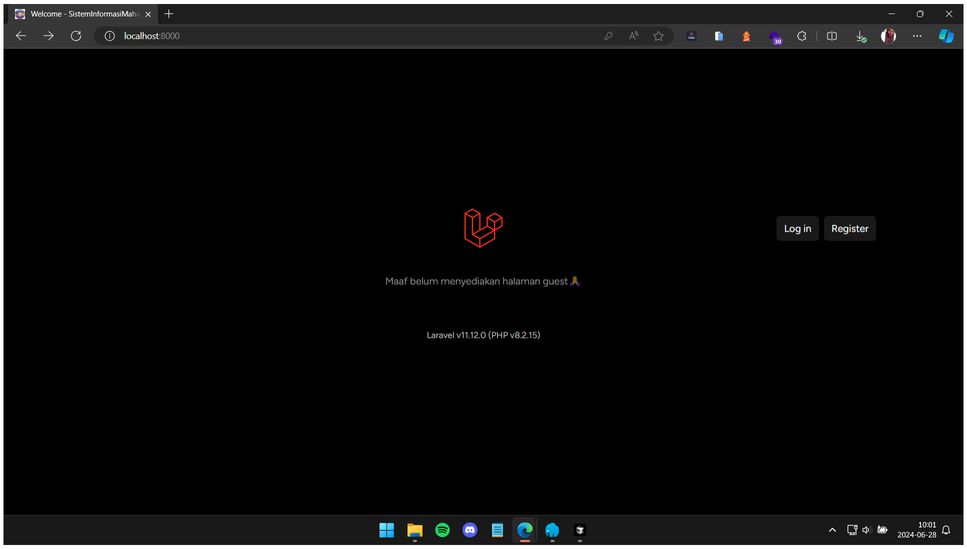
y. resources/js/Pages/Dashboard.vue

```
<script setup>
import AuthenticatedLayout from '@/Layouts/AuthenticatedLayout.vue';
import { Head } from '@inertiajs/vue3';
import PieChart from '@/Components/PieChart.vue';

```

```
const props = defineProps({
  programStudiData: Array,
  jumlahMahasiswa: Number,
  jumlahMataKuliah: Number,
  rataRataNilai: Number,
});
</script>

<template>
  <Head title="Dashboard" />

  <AuthenticatedLayout>
    <template #header>
      <h2 class="font-semibold text-xl text-gray-800 leading-tight">Dashboard</h2>
    </template>

    <div class="py-2">
      <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 text-gray-900">
            <h3 class="text-lg font-semibold mb-4">Selamat datang di Sistem Informasi Mahasiswa!</h3>

            <div class="grid grid-cols-1 md:grid-cols-3 gap-4 mb-8">
              <div class="bg-blue-100 p-4 rounded-lg">
                <h4 class="font-semibold">Jumlah Mahasiswa</h4>
                <p class="text-2xl">{{ jumlahMahasiswa }}</p>
              </div>
              <div class="bg-green-100 p-4 rounded-lg">
                <h4 class="font-semibold">Jumlah Mata Kuliah</h4>
                <p class="text-2xl">{{ jumlahMataKuliah }}</p>
              </div>
              <div class="bg-yellow-100 p-4 rounded-lg">
                <h4 class="font-semibold">Rata-rata Nilai</h4>
                <p class="text-2xl">{{ rataRataNilai }}</p>
              </div>
            </div>
            <div class="mb-8 flex flex-col justify-center items-center">
                <div class="max-w-xs">
                    <h4 class="font-semibold mb-4">Distribusi Mahasiswa per Program Studi</h4>
                    <PieChart :data="programStudiData" />
                </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </AuthenticatedLayout>
</template>
```

5. Screenshot Tampilan

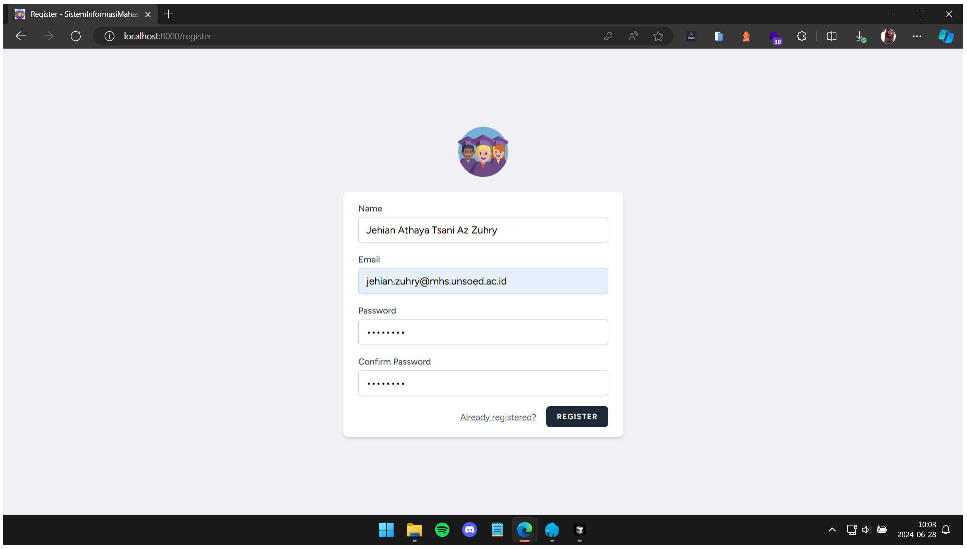
a. Welcome



Maaf belum menyediakan halaman guest 🙏

Laravel v11.12.0 (PHP v8.2.15)

b. Register



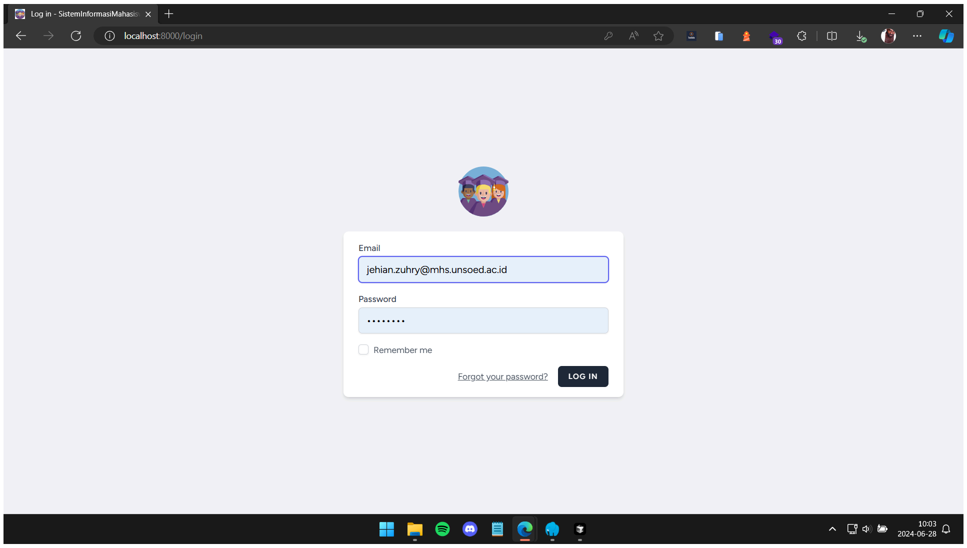
Name
Jehian Athaya Tsani Az Zuhry

Email
jehian.zuhry@mhs.unsoed.ac.id

Password
••••••••

Confirm Password
••••••••

c. Login



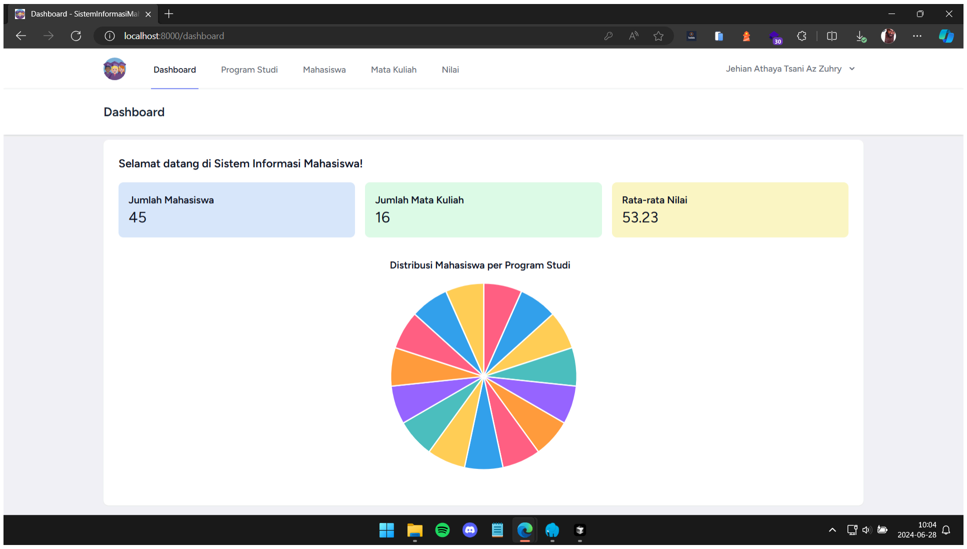
Email

jehian.zuhry@mhs.unsoed.ac.id

Password

••••••••

d. Dashboard



Dashboard

Selamat datang di Sistem Informasi Mahasiswa!

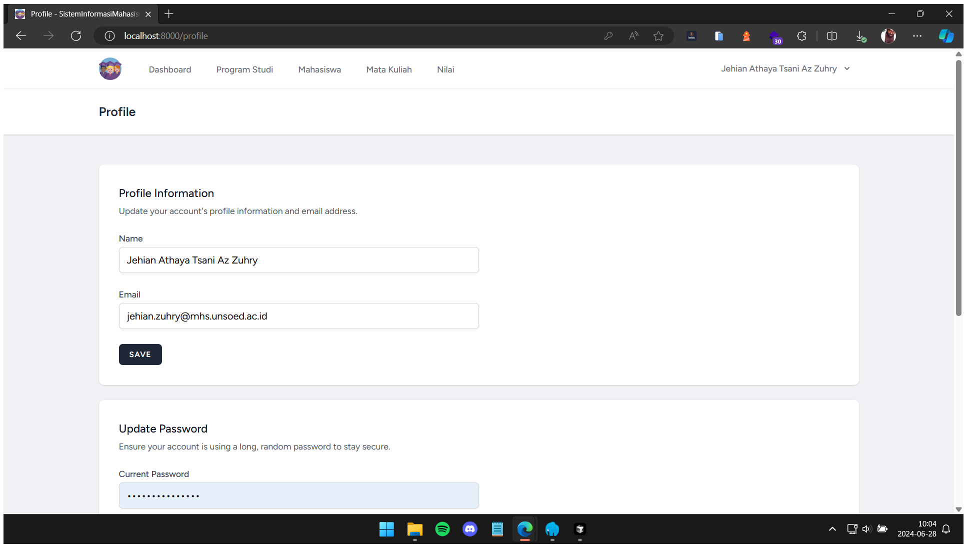
Jumlah Mahasiswa
45

Jumlah Mata Kuliah
16

Rata-rata Nilai
53.23

Distribusi Mahasiswa per Program Studi

e. Profile



Profile

Profile Information

Update your account's profile information and email address.

Name: Jehian Athaya Tsani Az Zuhry

Email: jehian.zuhry@mhs.unsoed.ac.id

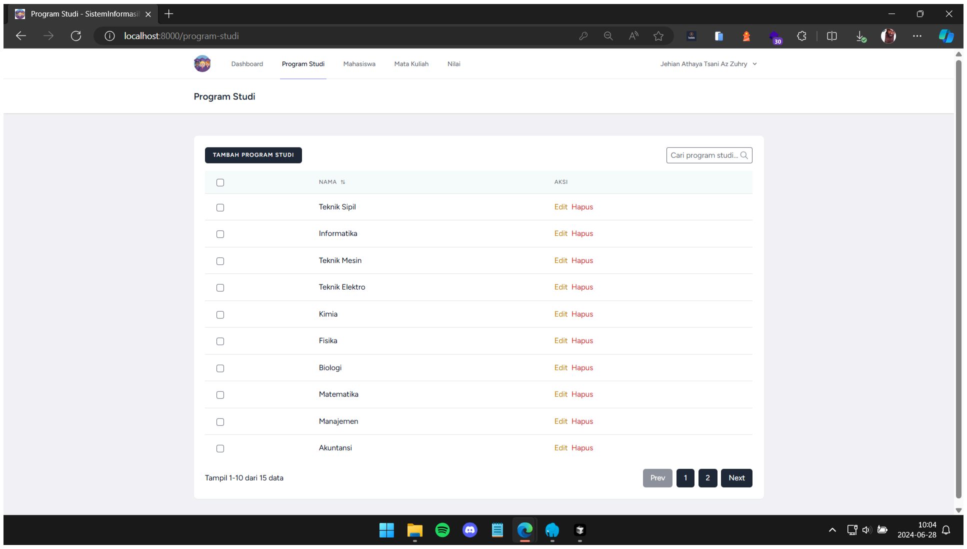
SAVE

Update Password

Ensure your account is using a long, random password to stay secure.

Current Password: •••••••••••••••

f. Program Studi - Index



Program Studi - SistemInformasi
localhost:8000/program-studi
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:04
2024-06-28

Program Studi

TAMBAH PROGRAM STUDI

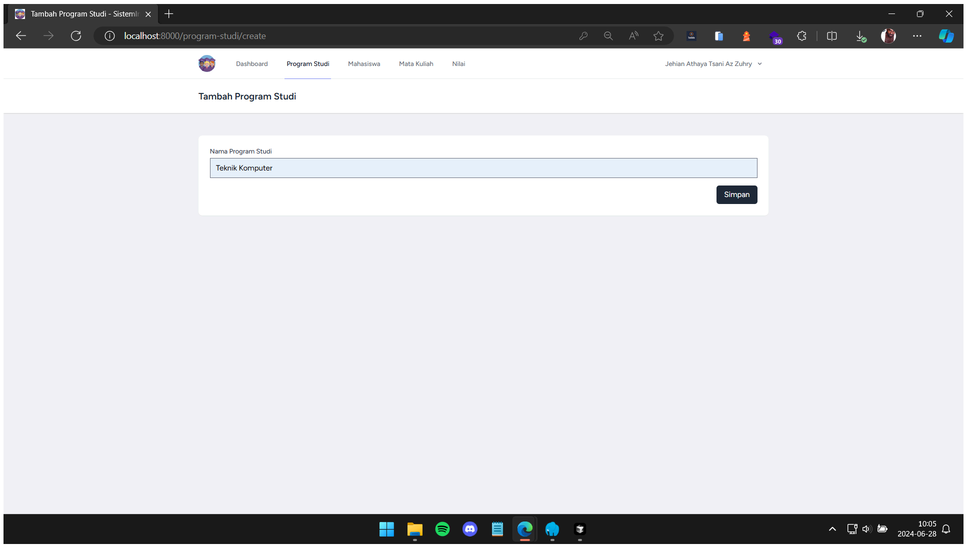
Cari program studi...

□	NAMA ⇅	AKSI
□	Teknik Sipil	Edit Hapus
□	Informatika	Edit Hapus
□	Teknik Mesin	Edit Hapus
□	Teknik Elektro	Edit Hapus
□	Kimia	Edit Hapus
□	Fisika	Edit Hapus
□	Biologi	Edit Hapus
□	Matematika	Edit Hapus
□	Manajemen	Edit Hapus
□	Akuntansi	Edit Hapus

Tampil 1-10 dari 15 data

Prev 1 2 Next

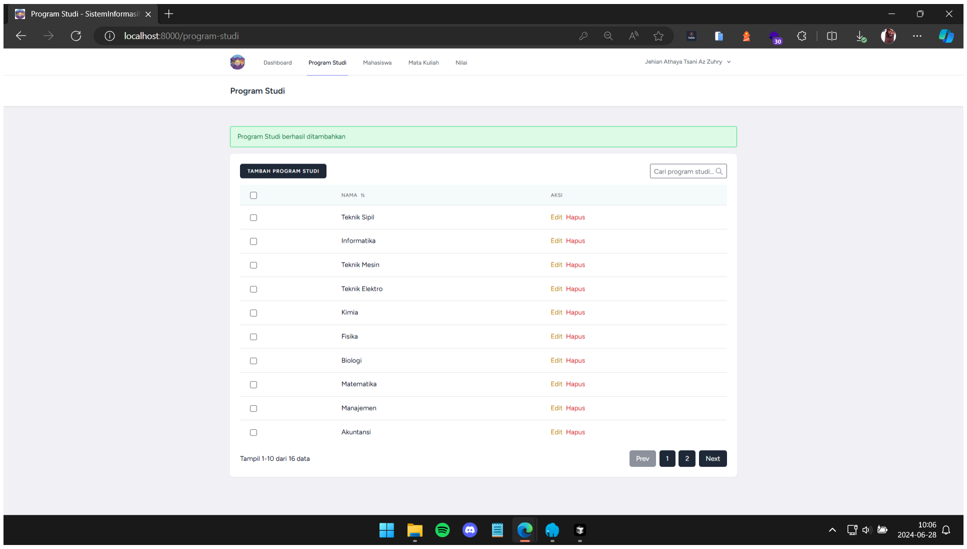
g. Program Studi - Create



Tambah Program Studi - SistemI
localhost:8000/program-studi/create
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry

Tambah Program Studi

Nama Program Studi: Teknik Komputer



Program Studi - SistemInformasi
localhost:8000/program-studi
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry

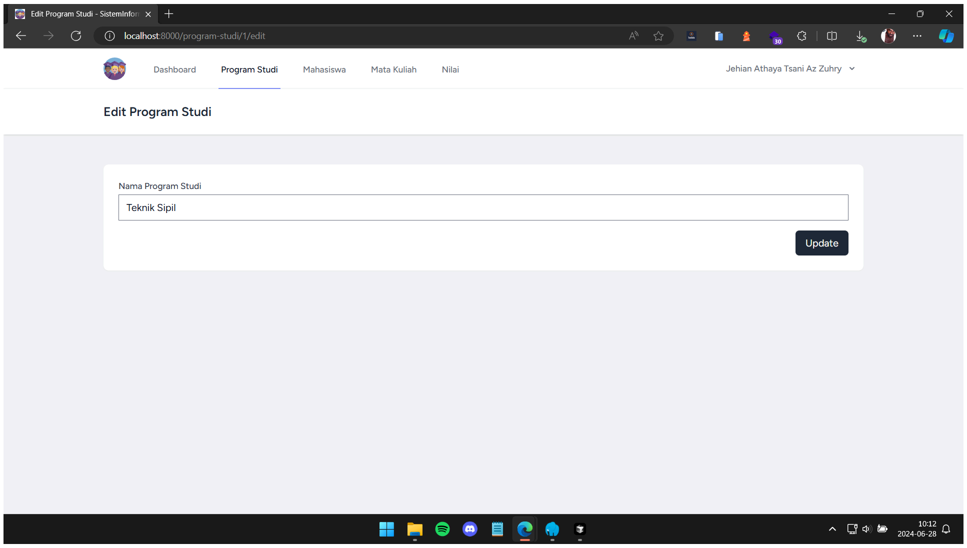
Program Studi

Program Studi berhasil ditambahkan

	NAMA ⇅	AKSI
□	Teknik Sipil	Edit Hapus
□	Informatika	Edit Hapus
□	Teknik Mesin	Edit Hapus
□	Teknik Elektro	Edit Hapus
□	Kimia	Edit Hapus
□	Fisika	Edit Hapus
□	Biologi	Edit Hapus
□	Matematika	Edit Hapus
□	Manajemen	Edit Hapus
□	Akuntansi	Edit Hapus

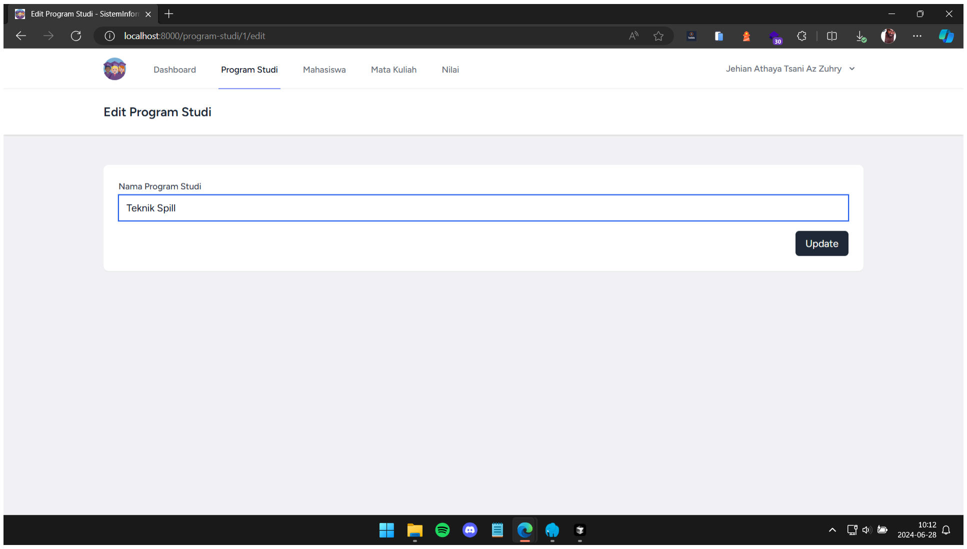
Tampil 1-10 dari 16 data

h. Program Studi - Edit


Edit Program Studi - SistemInfor
localhost:8000/program-studi/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
Update
10:12
2024-06-28

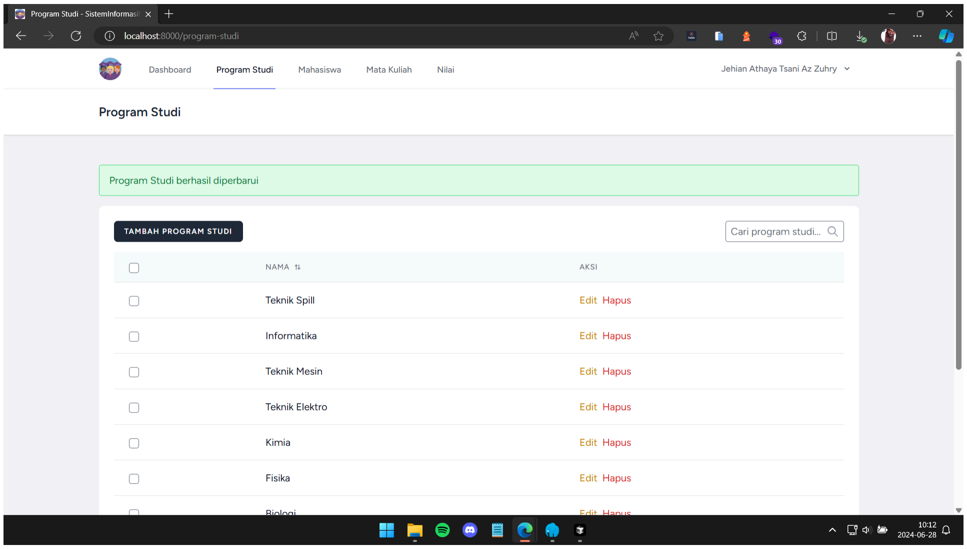
Edit Program Studi

Nama Program Studi: Teknik Sipil


Edit Program Studi - SistemInfor
localhost:8000/program-studi/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
Update
10:12
2024-06-28

Edit Program Studi

Nama Program Studi: Teknik Spill

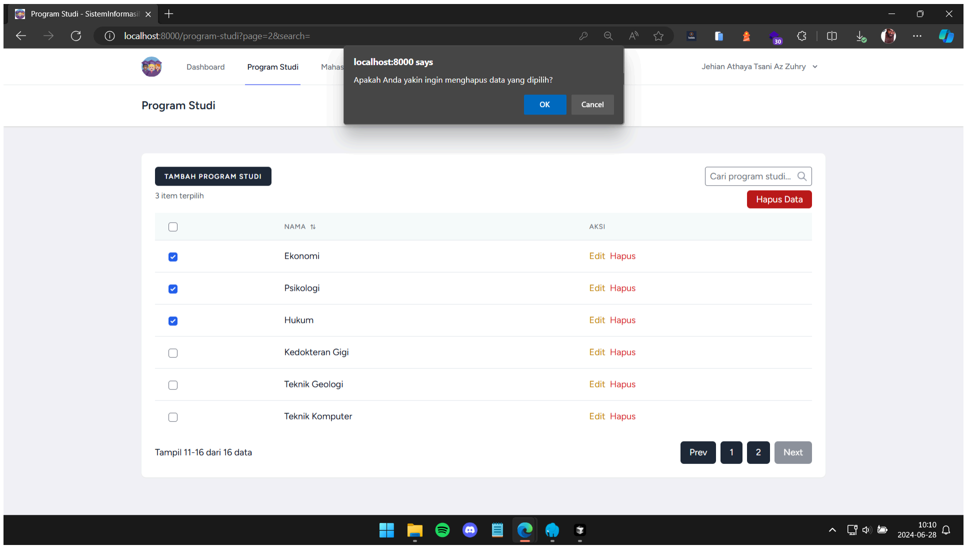


Program Studi

Program Studi berhasil diperbarui

□	NAMA	AKSI
□	Teknik Spill	Edit Hapus
□	Informatika	Edit Hapus
□	Teknik Mesin	Edit Hapus
□	Teknik Elektro	Edit Hapus
□	Kimia	Edit Hapus
□	Fisika	Edit Hapus
□	Biologi	Edit Hapus

i. Program Studi - Delete



localhost:8000 says

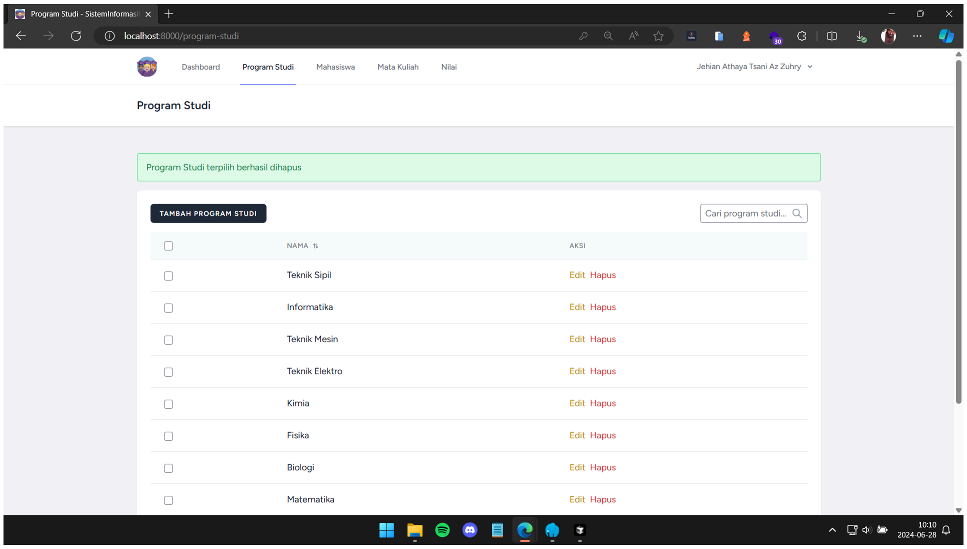
Apakah Anda yakin ingin menghapus data yang dipilih?

OK Cancel

3 item terpilih

	NAMA	AKSI
☑	Ekonomi	Edit Hapus
☑	Psikologi	Edit Hapus
☑	Hukum	Edit Hapus
□	Kedokteran Gigi	Edit Hapus
□	Teknik Geologi	Edit Hapus
□	Teknik Komputer	Edit Hapus

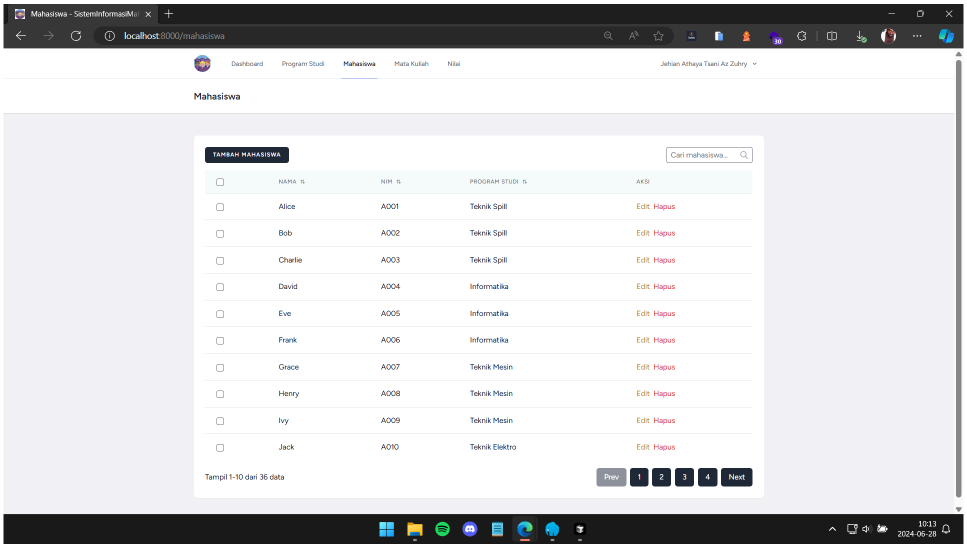
Tampil 11-16 dari 16 data



Program Studi terpilih berhasil dihapus

	NAMA	AKSI
□	Teknik Sipil	Edit Hapus
□	Informatika	Edit Hapus
□	Teknik Mesin	Edit Hapus
□	Teknik Elektro	Edit Hapus
□	Kimia	Edit Hapus
□	Fisika	Edit Hapus
□	Biologi	Edit Hapus
□	Matematika	Edit Hapus

j. Mahasiswa - Index



Mahasiswa

TAMBAH MAHASISWA

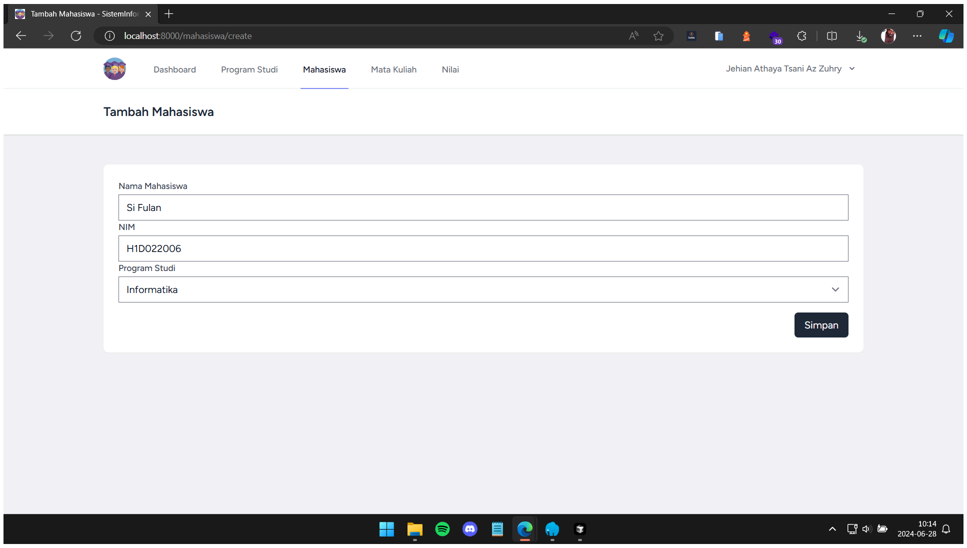
Cari mahasiswa...

□	NAMA ⇅	NIM ⇅	PROGRAM STUDI ⇅	AKSI
□	Alice	A001	Teknik Spill	Edit Hapus
□	Bob	A002	Teknik Spill	Edit Hapus
□	Charlie	A003	Teknik Spill	Edit Hapus
□	David	A004	Informatika	Edit Hapus
□	Eve	A005	Informatika	Edit Hapus
□	Frank	A006	Informatika	Edit Hapus
□	Grace	A007	Teknik Mesin	Edit Hapus
□	Henry	A008	Teknik Mesin	Edit Hapus
□	Ivy	A009	Teknik Mesin	Edit Hapus
□	Jack	A010	Teknik Elektro	Edit Hapus

Tampil 1-10 dari 36 data

Prev 1 2 3 4 Next

k. Mahasiswa - Create



Tambah Mahasiswa - SistemInfor
localhost:8000/mahasiswa/create
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:14
2024-06-28

Tambah Mahasiswa

Nama Mahasiswa

Si Fulan

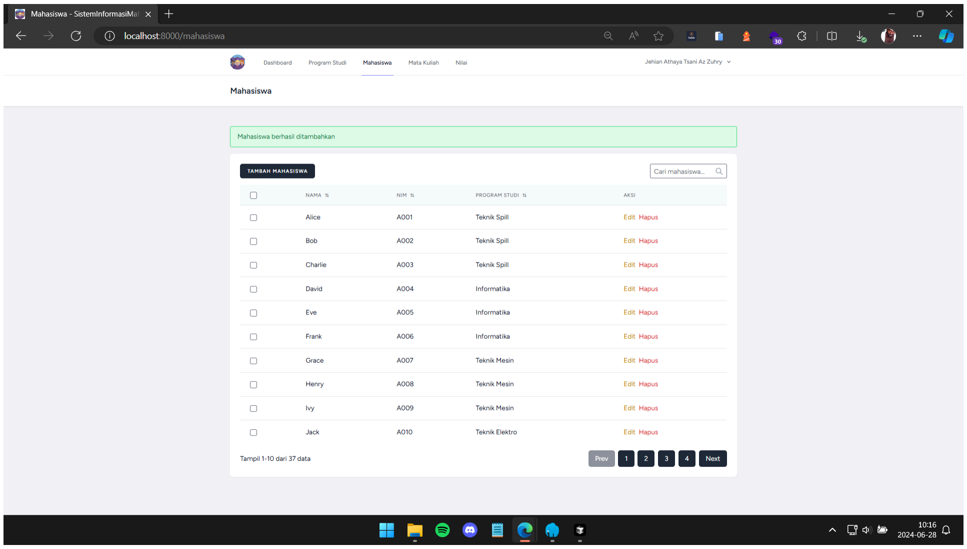
NIM

H1D022006

Program Studi

Informatika

Simpan



Mahasiswa

Mahasiswa berhasil ditambahkan

TAMBAH MAHASISWA

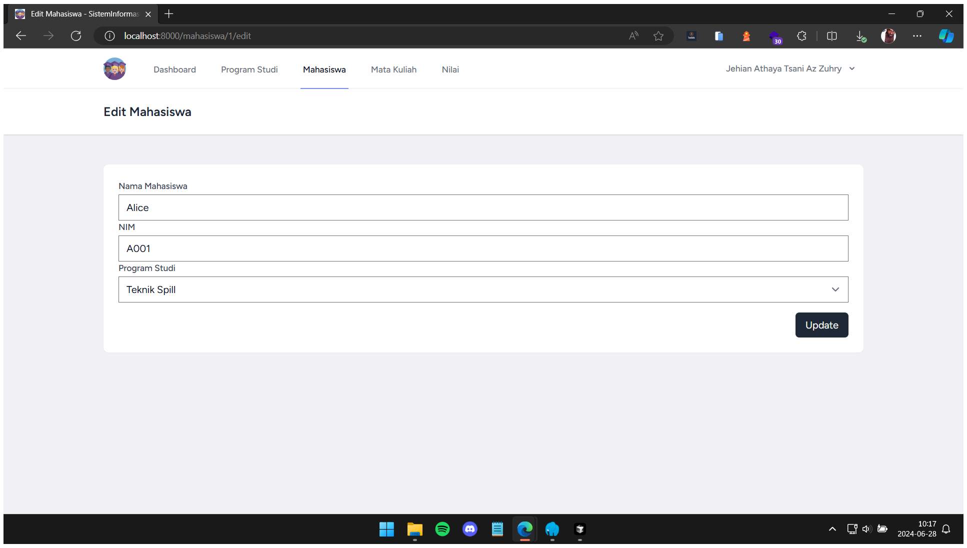
Cari mahasiswa...

□	NAMA ⇅	NIM ⇅	PROGRAM STUDI ⇅	AKSI
□	Alice	A001	Teknik Spill	Edit Hapus
□	Bob	A002	Teknik Spill	Edit Hapus
□	Charlie	A003	Teknik Spill	Edit Hapus
□	David	A004	Informatika	Edit Hapus
□	Eve	A005	Informatika	Edit Hapus
□	Frank	A006	Informatika	Edit Hapus
□	Grace	A007	Teknik Mesin	Edit Hapus
□	Henry	A008	Teknik Mesin	Edit Hapus
□	Ivy	A009	Teknik Mesin	Edit Hapus
□	Jack	A010	Teknik Elektro	Edit Hapus

Tampil 1-10 dari 37 data

Prev 1 2 3 4 Next

l. Mahasiswa - Edit

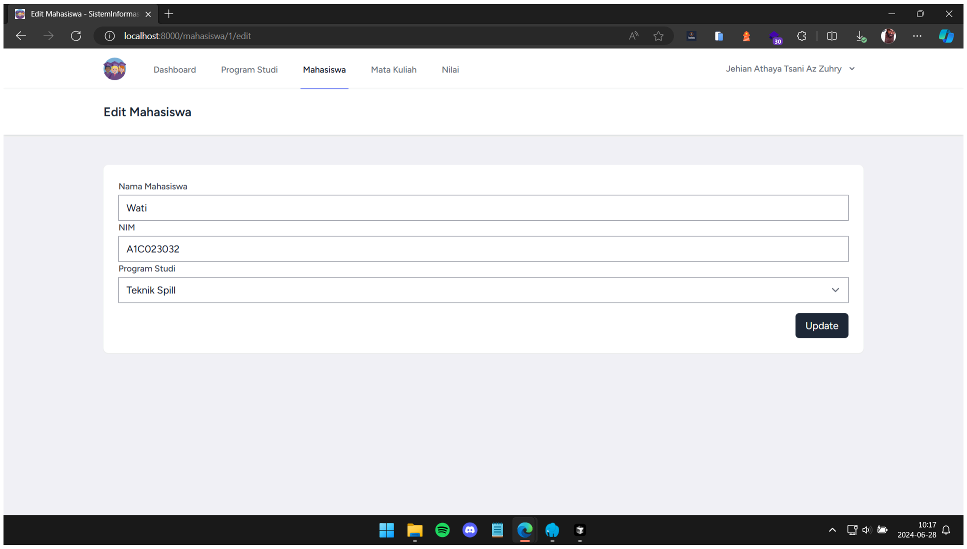

Edit Mahasiswa - SistemInformas
localhost:8000/mahasiswa/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry

Edit Mahasiswa

Nama Mahasiswa: Alice

NIM: A001

Program Studi: Teknik Spill

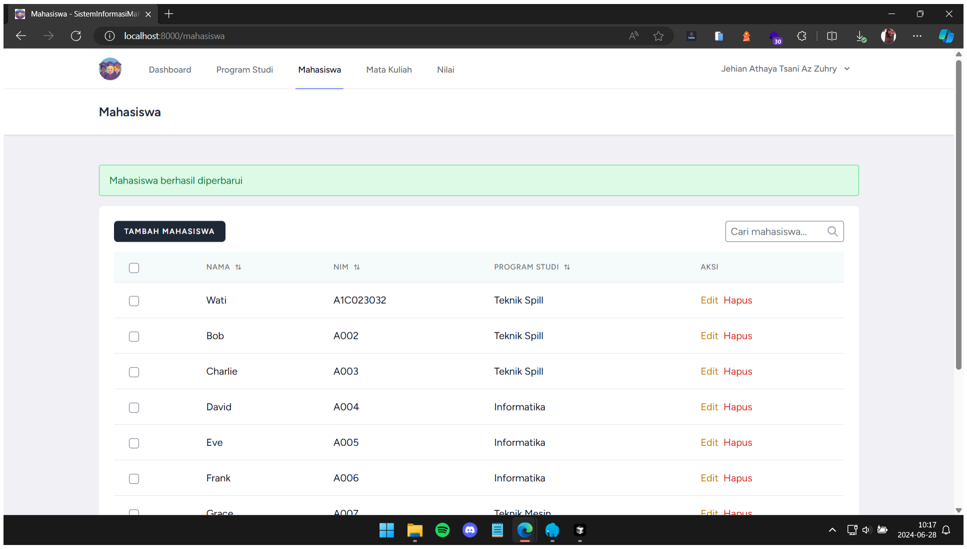

Edit Mahasiswa - SistemInformas
localhost:8000/mahasiswa/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry

Edit Mahasiswa

Nama Mahasiswa: Wati

NIM: A1C023032

Program Studi: Teknik Spill

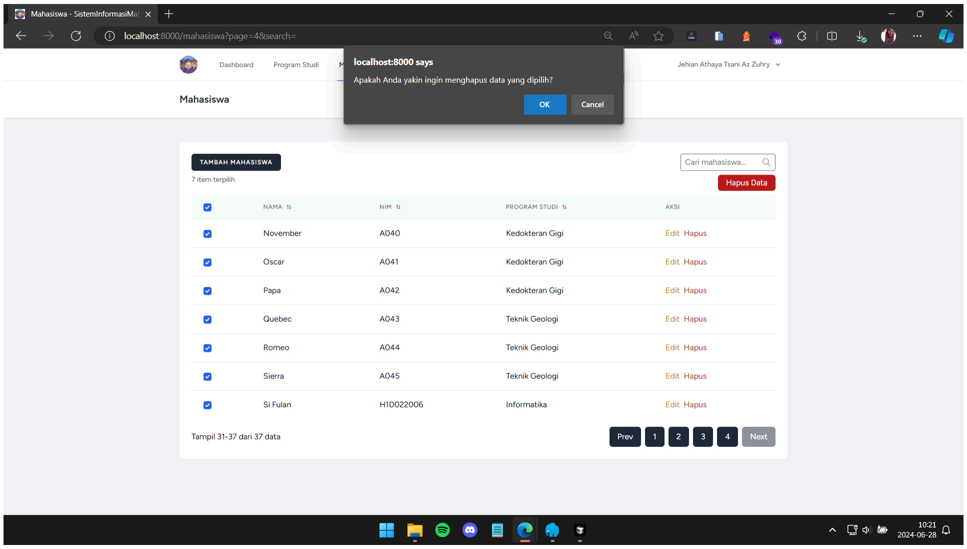


Mahasiswa

Mahasiswa berhasil diperbarui

□	NAMA ⇅	NIM ⇅	PROGRAM STUDI ⇅	AKSI
□	Wati	A1C023032	Teknik Spill	Edit Hapus
□	Bob	A002	Teknik Spill	Edit Hapus
□	Charlie	A003	Teknik Spill	Edit Hapus
□	David	A004	Informatika	Edit Hapus
□	Eve	A005	Informatika	Edit Hapus
□	Frank	A006	Informatika	Edit Hapus
□	Grace	A007	Teknik Mesin	Edit Hapus

m. Mahasiswa - Delete


Mahasiswa - SistemInformasiMal
localhost:8000/mahasiswa?page=4&search=
Dashboard
Program Studi
Jehian Athaya Tsani Az Zuhry

localhost:8000 says

Apakah Anda yakin ingin menghapus data yang dipilih?

OK Cancel

Mahasiswa

TAMBAH MAHASISWA

Cari mahasiswa...

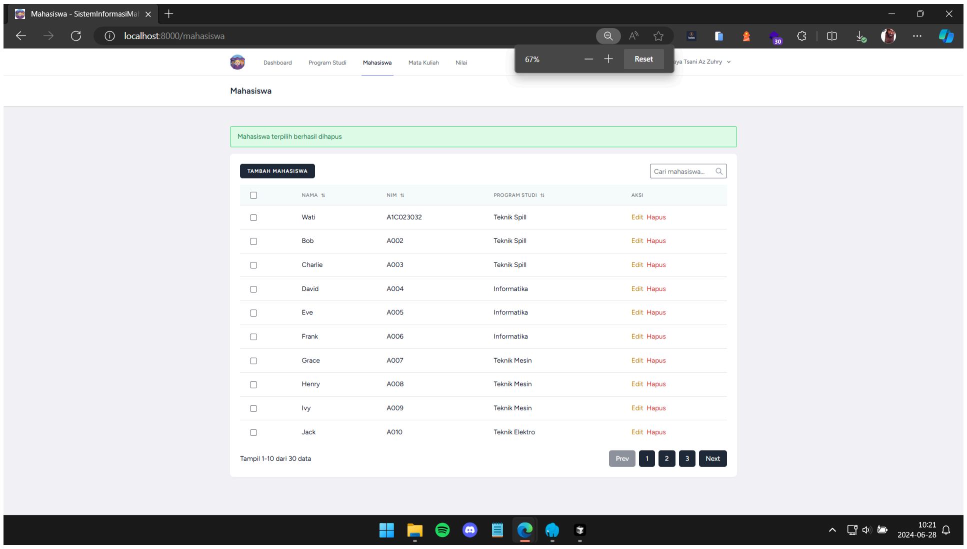
7 item terpilih

Hapus Data

☑	NAMA ⇅	NIM ⇅	PROGRAM STUDI ⇅	AKSI
☑	November	A040	Kedokteran Gigi	Edit Hapus
☑	Oscar	A041	Kedokteran Gigi	Edit Hapus
☑	Papa	A042	Kedokteran Gigi	Edit Hapus
☑	Quebec	A043	Teknik Geologi	Edit Hapus
☑	Romeo	A044	Teknik Geologi	Edit Hapus
☑	Sierra	A045	Teknik Geologi	Edit Hapus
☑	Si Fulan	H1D022006	Informatika	Edit Hapus

Tampil 31-37 dari 37 data

Prev 1 2 3 4 Next



Mahasiswa

Mahasiswa terpilih berhasil dihapus

TAMBAH MAHASISWA

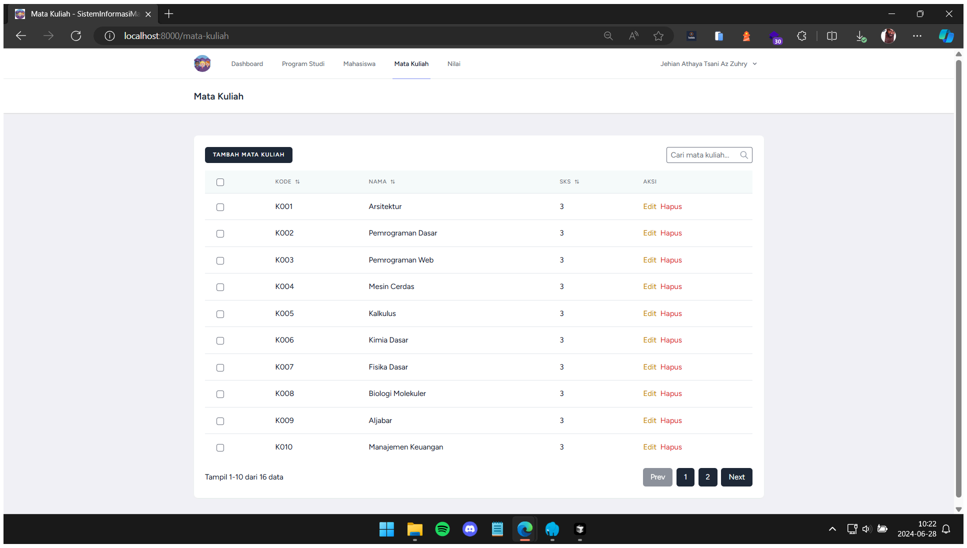
Cari mahasiswa...

□	NAMA ⇅	NIM ⇅	PROGRAM STUDI ⇅	AKSI
□	Wati	A1C023032	Teknik Spill	Edit Hapus
□	Bob	A002	Teknik Spill	Edit Hapus
□	Charlie	A003	Teknik Spill	Edit Hapus
□	David	A004	Informatika	Edit Hapus
□	Eve	A005	Informatika	Edit Hapus
□	Frank	A006	Informatika	Edit Hapus
□	Grace	A007	Teknik Mesin	Edit Hapus
□	Henry	A008	Teknik Mesin	Edit Hapus
□	Ivy	A009	Teknik Mesin	Edit Hapus
□	Jack	A010	Teknik Elektro	Edit Hapus

Tampil 1-10 dari 30 data

Prev 1 2 3 Next

n. Mata Kuliah - Index



Mata Kuliah - SistemInformasiMa
localhost:8000/mata-kuliah
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:22
2024-06-28

Mata Kuliah

TAMBAH MATA KULIAH

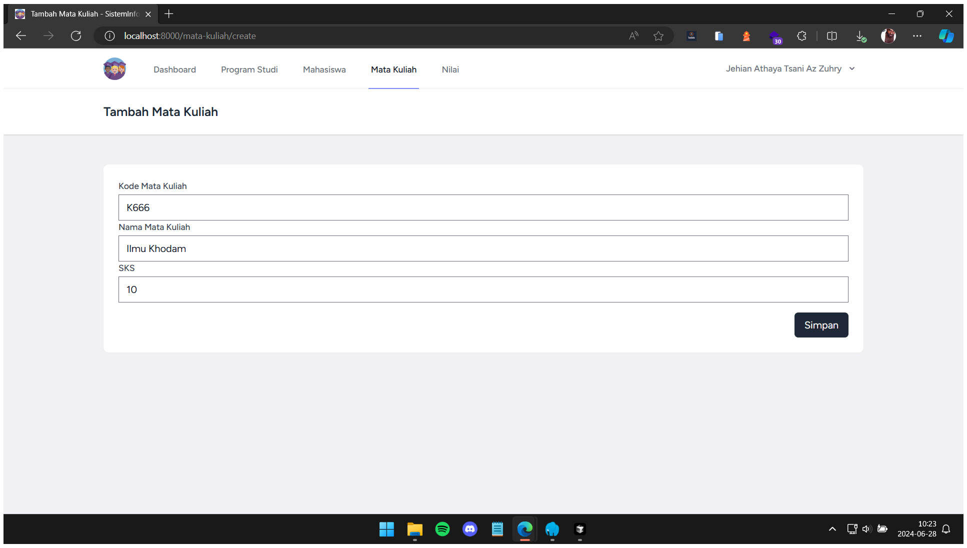
Cari mata kuliah...

□	KODE ⇅	NAMA ⇅	SKS ⇅	AKSI
□	K001	Arsitektur	3	Edit Hapus
□	K002	Pemrograman Dasar	3	Edit Hapus
□	K003	Pemrograman Web	3	Edit Hapus
□	K004	Mesin Cerdas	3	Edit Hapus
□	K005	Kalkulus	3	Edit Hapus
□	K006	Kimia Dasar	3	Edit Hapus
□	K007	Fisika Dasar	3	Edit Hapus
□	K008	Biologi Molekuler	3	Edit Hapus
□	K009	Aljabar	3	Edit Hapus
□	K010	Manajemen Keuangan	3	Edit Hapus

Tampil 1-10 dari 16 data

Prev 1 2 Next

o. Mata Kuliah - Create



Tambah Mata Kuliah - SistemInfo
localhost:8000/mata-kuliah/create
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:23
2024-06-28

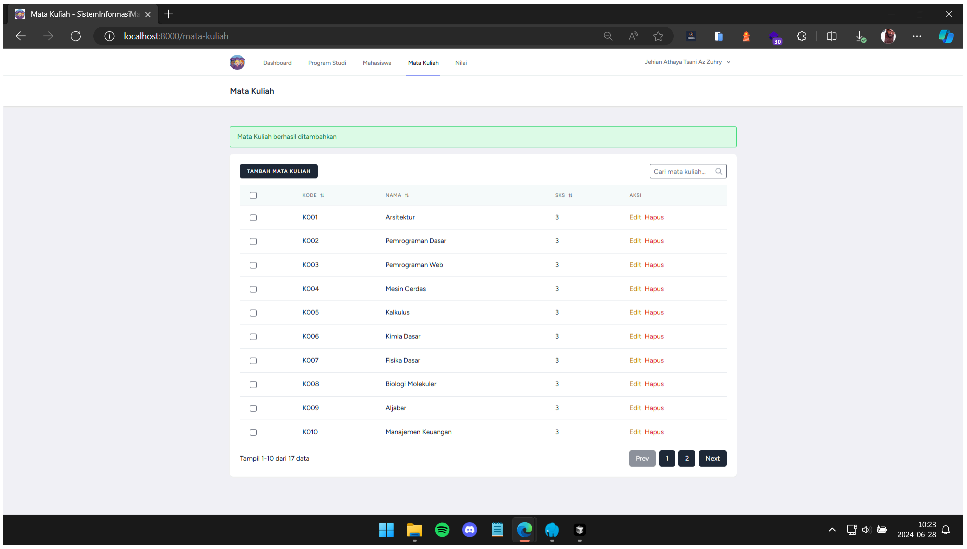
Tambah Mata Kuliah

Kode Mata Kuliah: K666

Nama Mata Kuliah: Ilmu Khodam

SKS: 10

Simpan



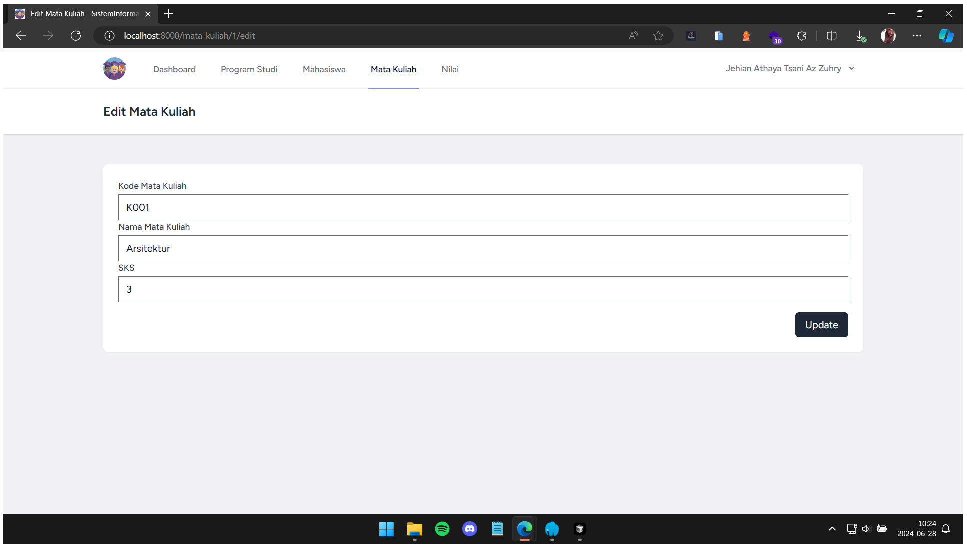
Mata Kuliah

Mata Kuliah berhasil ditambahkan

□	KODE ⇅	NAMA ⇅	SKS ⇅	AKSI
□	K001	Arsitektur	3	Edit Hapus
□	K002	Pemrograman Dasar	3	Edit Hapus
□	K003	Pemrograman Web	3	Edit Hapus
□	K004	Mesin Cerdas	3	Edit Hapus
□	K005	Kalkulus	3	Edit Hapus
□	K006	Kimia Dasar	3	Edit Hapus
□	K007	Fisika Dasar	3	Edit Hapus
□	K008	Biologi Molekuler	3	Edit Hapus
□	K009	Aljabar	3	Edit Hapus
□	K010	Manajemen Keuangan	3	Edit Hapus

Tampil 1-10 dari 17 data

p. Mata Kuliah - Edit


Edit Mata Kuliah - SistemInforma
localhost:8000/mata-kuliah/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:24
2024-06-28

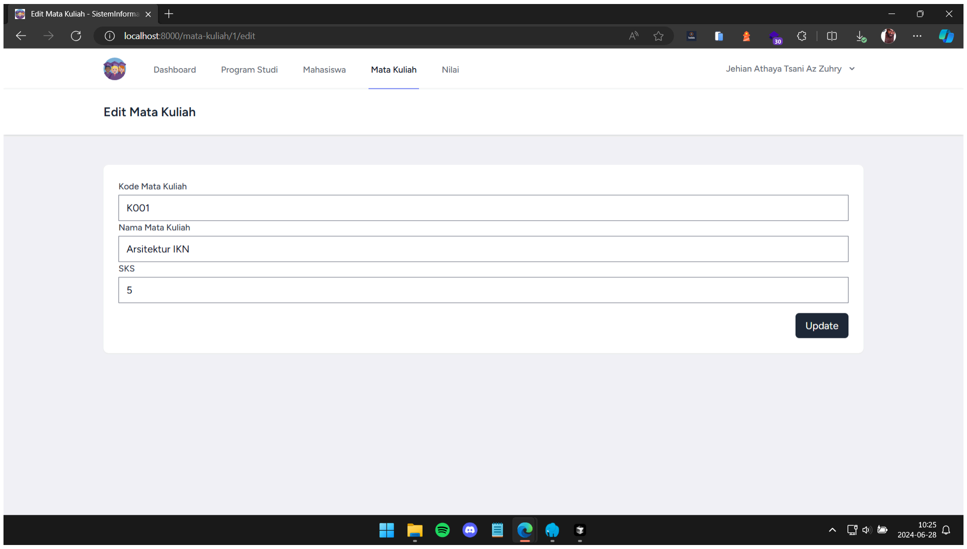
Edit Mata Kuliah

Kode Mata Kuliah
K001

Nama Mata Kuliah
Arsitektur

SKS
3

Update


Edit Mata Kuliah - SistemInforma
localhost:8000/mata-kuliah/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:25
2024-06-28

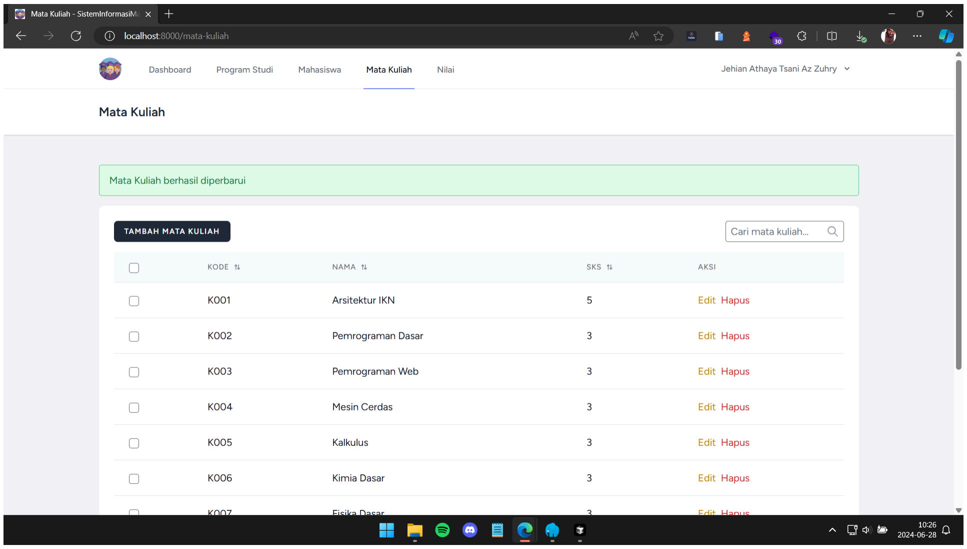
Edit Mata Kuliah

Kode Mata Kuliah
K001

Nama Mata Kuliah
Arsitektur IKN

SKS
5

Update

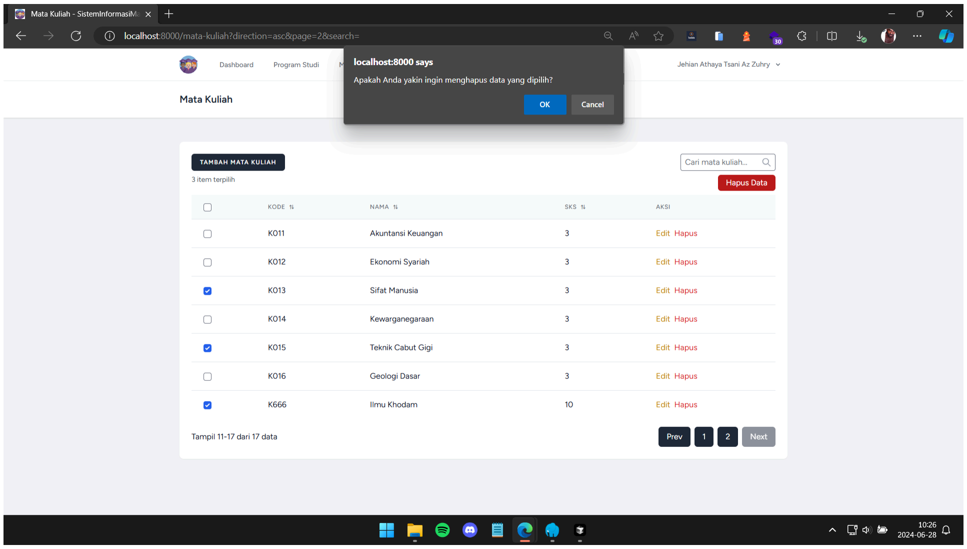


Mata Kuliah

Mata Kuliah berhasil diperbarui

□	KODE ⇅	NAMA ⇅	SKS ⇅	AKSI
□	K001	Arsitektur IKN	5	Edit Hapus
□	K002	Pemrograman Dasar	3	Edit Hapus
□	K003	Pemrograman Web	3	Edit Hapus
□	K004	Mesin Cerdas	3	Edit Hapus
□	K005	Kalkulus	3	Edit Hapus
□	K006	Kimia Dasar	3	Edit Hapus
□	K007	Fisika Dasar	3	Edit Hapus

q. Mata Kuliah - Delete



Mata Kuliah - SistemInformasiM
localhost:8000/mata-kuliah?direction=asc&page=2&search=
Dashboard
Program Studi
Jehian Athaya Tsani Az Zuhry

localhost:8000 says

Apakah Anda yakin ingin menghapus data yang dipilih?

OK Cancel

Mata Kuliah

TAMBAH MATA KULIAH

3 item terpilih

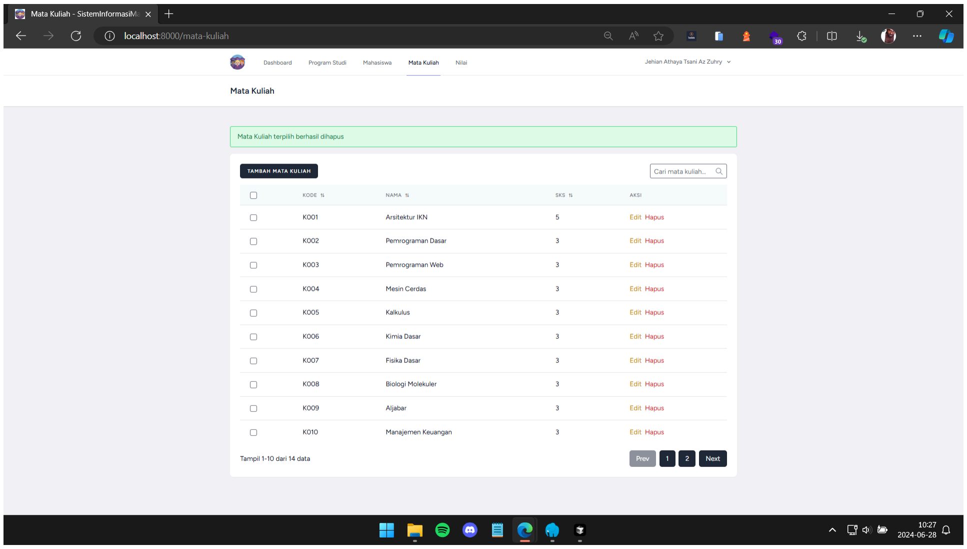
Cari mata kuliah...

Hapus Data

□	KODE ⇅	NAMA ⇅	SKS ⇅	AKSI
□	K011	Akuntansi Keuangan	3	Edit Hapus
□	K012	Ekonomi Syariah	3	Edit Hapus
☑	K013	Sifat Manusia	3	Edit Hapus
□	K014	Kewarganegaraan	3	Edit Hapus
☑	K015	Teknik Cabut Gigi	3	Edit Hapus
□	K016	Geologi Dasar	3	Edit Hapus
☑	K666	Ilmu Khodam	10	Edit Hapus

Tampil 11-17 dari 17 data

Prev 1 2 Next



Mata Kuliah - SistemInformasiM
localhost:8000/mata-kuliah
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry

Mata Kuliah

Mata Kuliah terpilih berhasil dihapus

TAMBAH MATA KULIAH

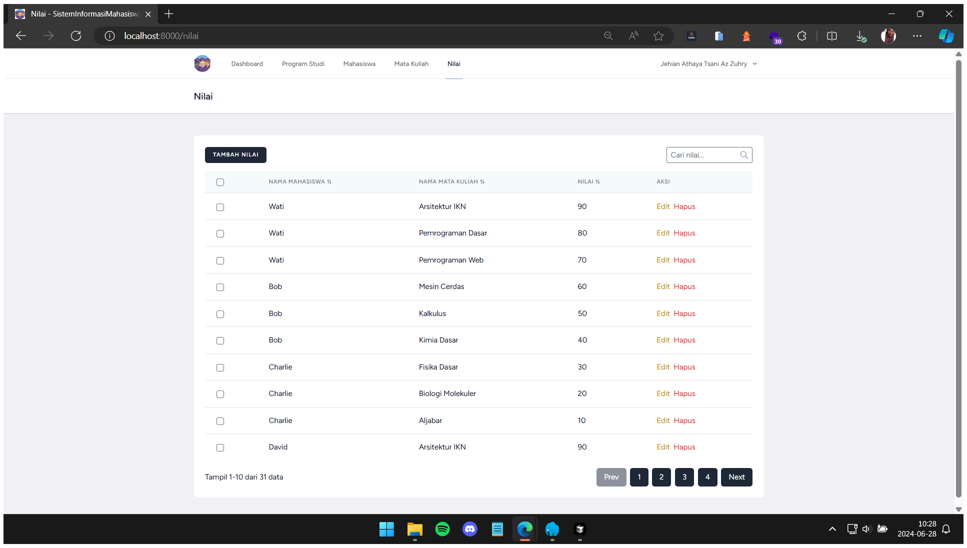
Cari mata kuliah...

□	KODE ⇅	NAMA ⇅	SKS ⇅	AKSI
□	K001	Arsitektur IKN	5	Edit Hapus
□	K002	Pemrograman Dasar	3	Edit Hapus
□	K003	Pemrograman Web	3	Edit Hapus
□	K004	Mesin Cerdas	3	Edit Hapus
□	K005	Kalkulus	3	Edit Hapus
□	K006	Kimia Dasar	3	Edit Hapus
□	K007	Fisika Dasar	3	Edit Hapus
□	K008	Biologi Molekuler	3	Edit Hapus
□	K009	Aljabar	3	Edit Hapus
□	K010	Manajemen Keuangan	3	Edit Hapus

Tampil 1-10 dari 14 data

Prev 1 2 Next

r. Nilai - Index

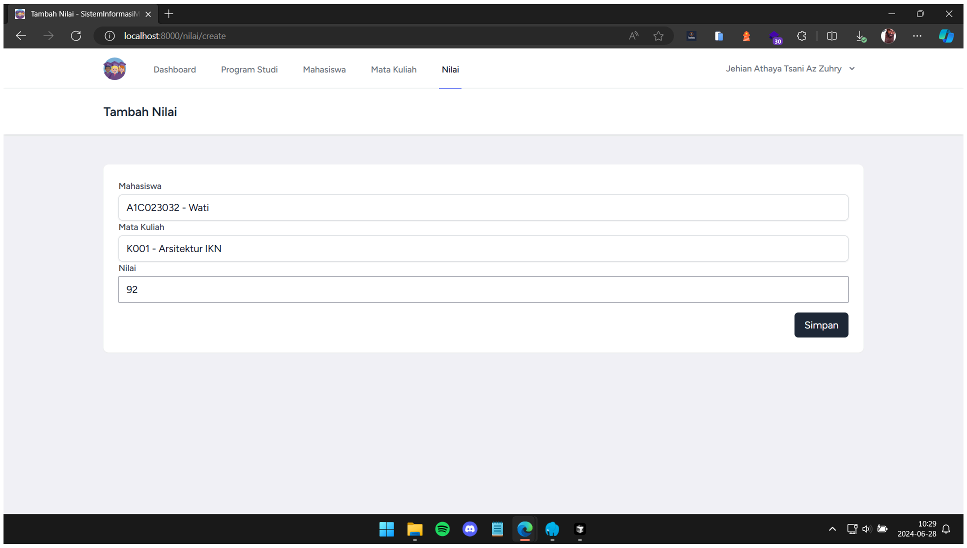


Nilai

□	NAMA MAHASISWA ⇅	NAMA MATA KULIAH ⇅	NILAI ⇅	AKSI
□	Wati	Arsitektur IKN	90	Edit Hapus
□	Wati	Pemrograman Dasar	80	Edit Hapus
□	Wati	Pemrograman Web	70	Edit Hapus
□	Bob	Mesin Cerdas	60	Edit Hapus
□	Bob	Kalkulus	50	Edit Hapus
□	Bob	Kimia Dasar	40	Edit Hapus
□	Charlie	Fisika Dasar	30	Edit Hapus
□	Charlie	Biologi Molekuler	20	Edit Hapus
□	Charlie	Aljabar	10	Edit Hapus
□	David	Arsitektur IKN	90	Edit Hapus

Tampil 1-10 dari 31 data

s. Nilai - Create



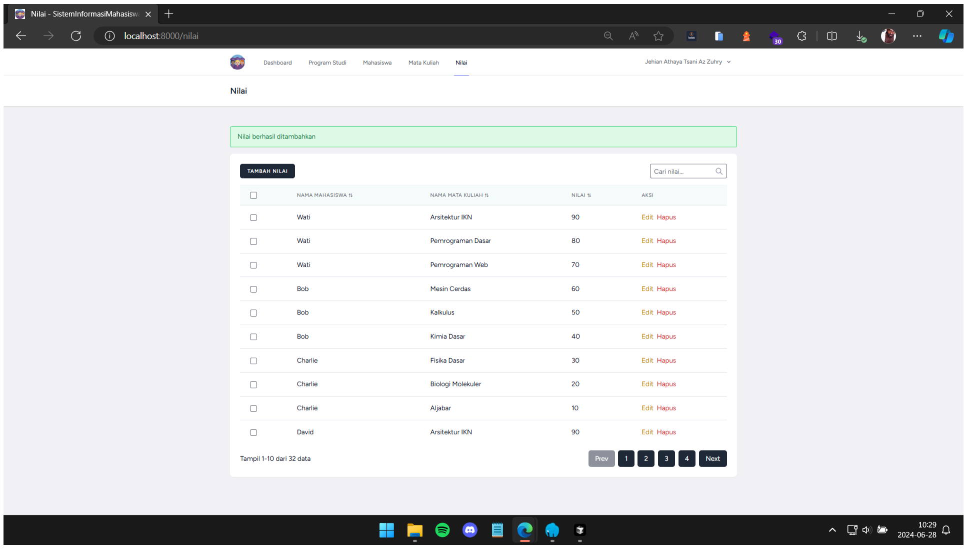
Tambah Nilai - SistemInformasiM
localhost:8000/nilai/create
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry

Tambah Nilai

Mahasiswa: A1C023032 - Wati

Mata Kuliah: K001 - Arsitektur IKN

Nilai: 92



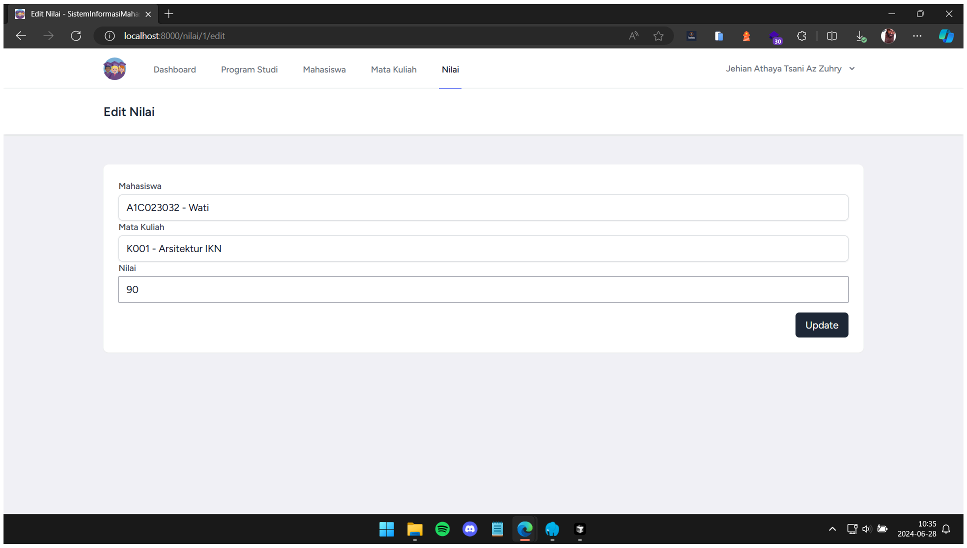
Nilai

Nilai berhasil ditambahkan

□	NAMA MAHASISWA ⇅	NAMA MATA KULIAH ⇅	NILAI ⇅	AKSI
□	Wati	Arsitektur IKN	90	Edit Hapus
□	Wati	Pemrograman Dasar	80	Edit Hapus
□	Wati	Pemrograman Web	70	Edit Hapus
□	Bob	Mesin Cerdas	60	Edit Hapus
□	Bob	Kalkulus	50	Edit Hapus
□	Bob	Kimia Dasar	40	Edit Hapus
□	Charlie	Fisika Dasar	30	Edit Hapus
□	Charlie	Biologi Molekuler	20	Edit Hapus
□	Charlie	Aljabar	10	Edit Hapus
□	David	Arsitektur IKN	90	Edit Hapus

Tampil 1-10 dari 32 data

t. Nilai - Edit


Edit Nilai - SistemInformasiMahas
localhost:8000/nilai/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:35
2024-06-28

Edit Nilai

Mahasiswa

A1C023032 - Wati

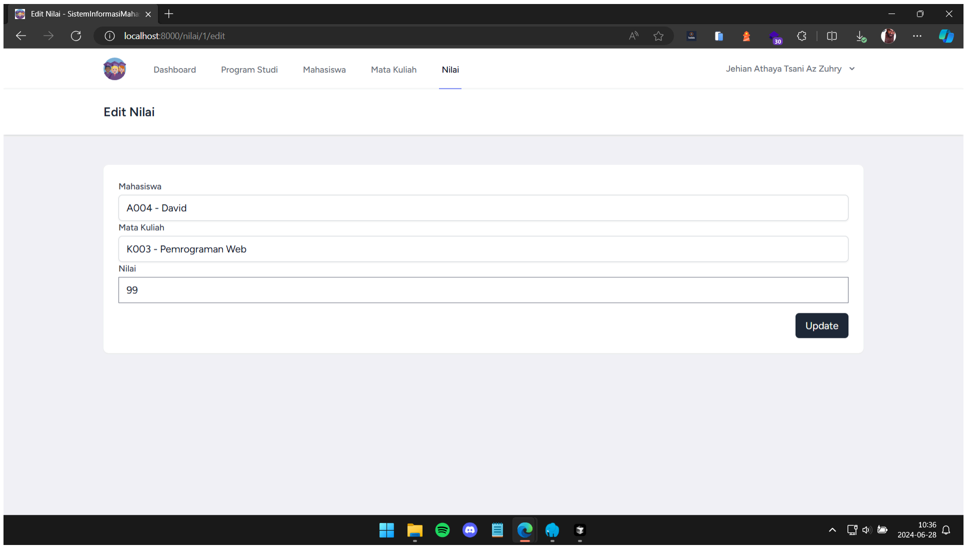
Mata Kuliah

K001 - Arsitektur IKN

Nilai

90

Update


Edit Nilai - SistemInformasiMahas
localhost:8000/nilai/1/edit
Dashboard
Program Studi
Mahasiswa
Mata Kuliah
Nilai
Jehian Athaya Tsani Az Zuhry
10:36
2024-06-28

Edit Nilai

Mahasiswa

A004 - David

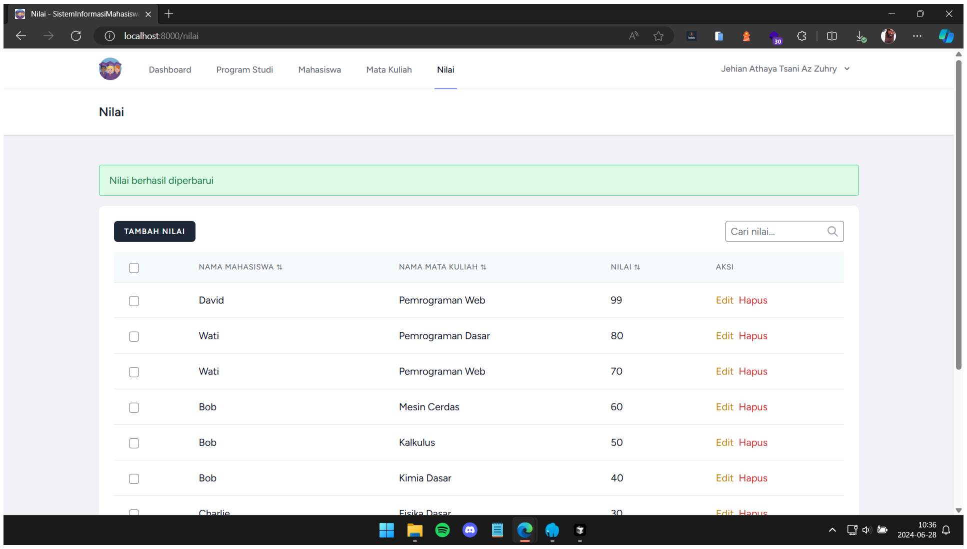
Mata Kuliah

K003 - Pemrograman Web

Nilai

99

Update

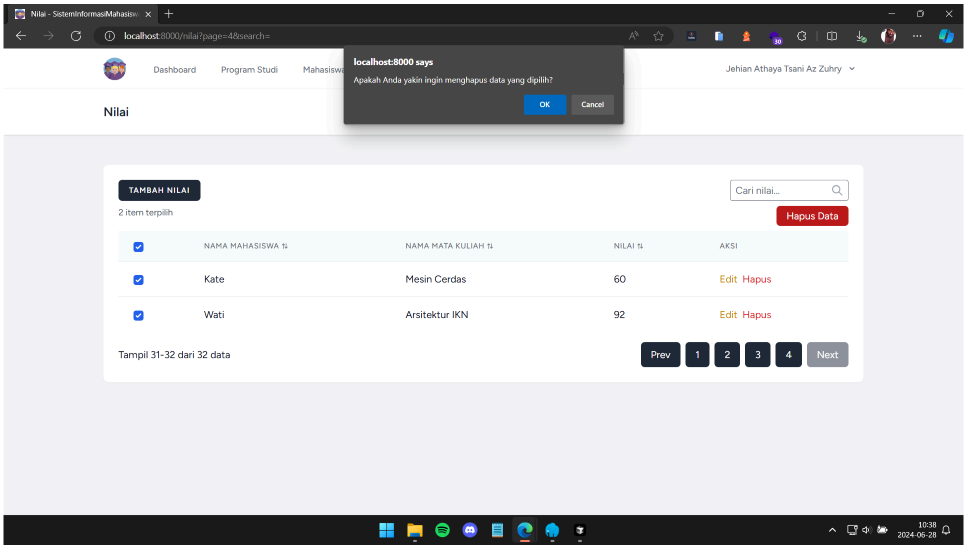


Nilai

Nilai berhasil diperbarui

☐	NAMA MAHASISWA ⇅	NAMA MATA KULIAH ⇅	NILAI ⇅	AKSI
☐	David	Pemrograman Web	99	Edit Hapus
☐	Wati	Pemrograman Dasar	80	Edit Hapus
☐	Wati	Pemrograman Web	70	Edit Hapus
☐	Bob	Mesin Cerdas	60	Edit Hapus
☐	Bob	Kalkulus	50	Edit Hapus
☐	Bob	Kimia Dasar	40	Edit Hapus
☐	Charlie	Fisika Dasar	30	Edit Hapus

u. Nilai - Delete



Nilai - SisteminformasiMahasisw
localhost:8000/nilai?page=4&search=
Dashboard
Program Studi
Mahasiswa
Jehian Athaya Tsani Az Zuhry

localhost:8000 says

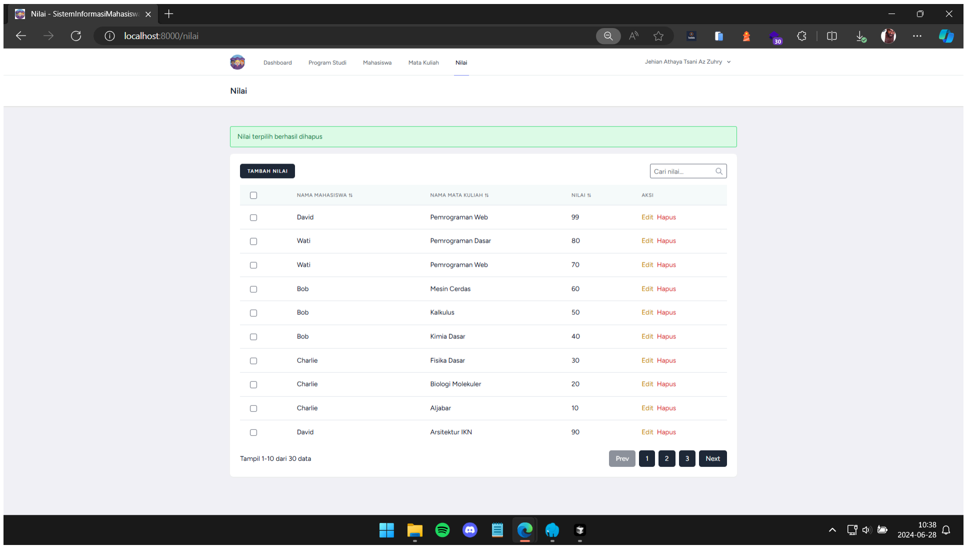
Apakah Anda yakin ingin menghapus data yang dipilih?

Nilai

2 item terpilih

☑	NAMA MAHASISWA ⇅	NAMA MATA KULIAH ⇅	NILAI ⇅	AKSI
☑	Kate	Mesin Cerdas	60	Edit Hapus
☑	Wati	Arsitektur IKN	92	Edit Hapus

Tampil 31-32 dari 32 data



Nilai

Nilai terpilih berhasil dihapus

☐	NAMA MAHASISWA ⇅	NAMA MATA KULIAH ⇅	NILAI ⇅	AKSI
☐	David	Pemrograman Web	99	Edit Hapus
☐	Wati	Pemrograman Dasar	80	Edit Hapus
☐	Wati	Pemrograman Web	70	Edit Hapus
☐	Bob	Mesin Cerdas	60	Edit Hapus
☐	Bob	Kalkulus	50	Edit Hapus
☐	Bob	Kimia Dasar	40	Edit Hapus
☐	Charlie	Fisika Dasar	30	Edit Hapus
☐	Charlie	Biologi Molekuler	20	Edit Hapus
☐	Charlie	Aljabar	10	Edit Hapus
☐	David	Arsitektur IKN	90	Edit Hapus

Tampil 1-10 dari 30 data