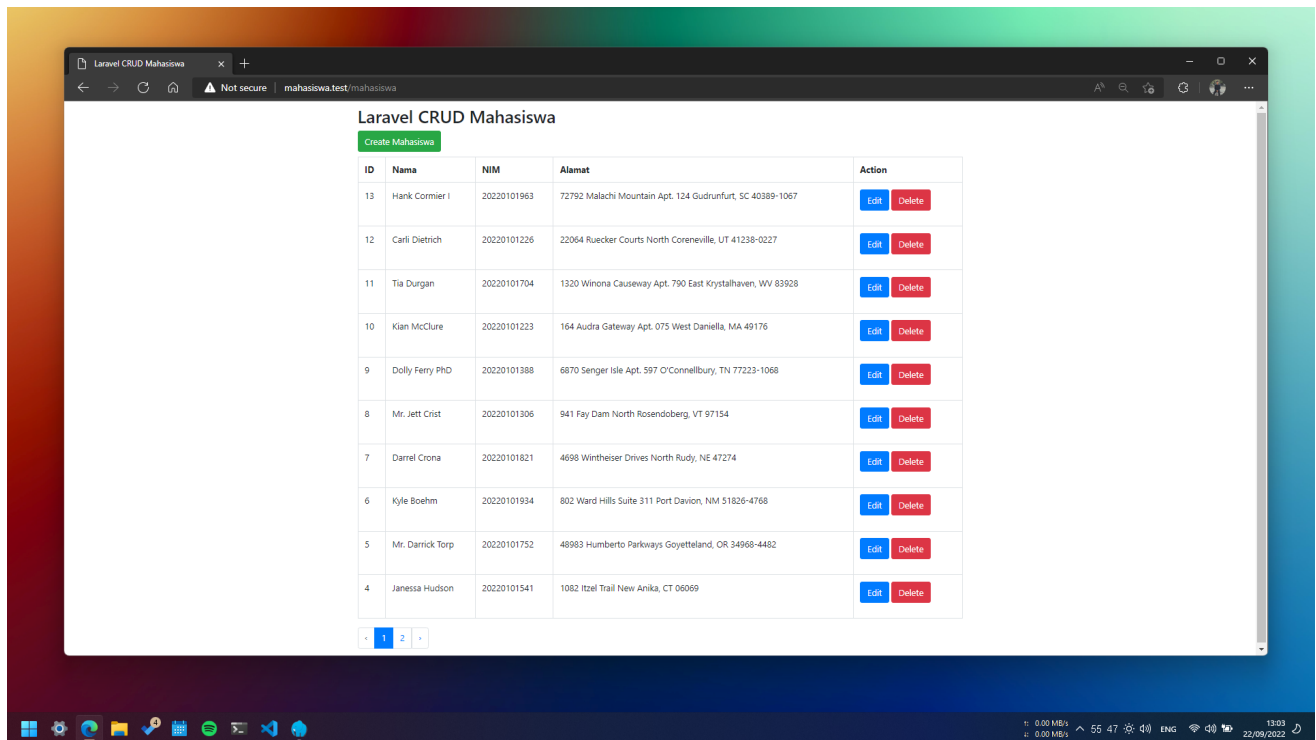


# Laravel 8 CRUD Mahasiswa



Membuat Laravel CRUD Mahasiswa tanpa Auth dan tanpa relasi DB

## Buat project baru

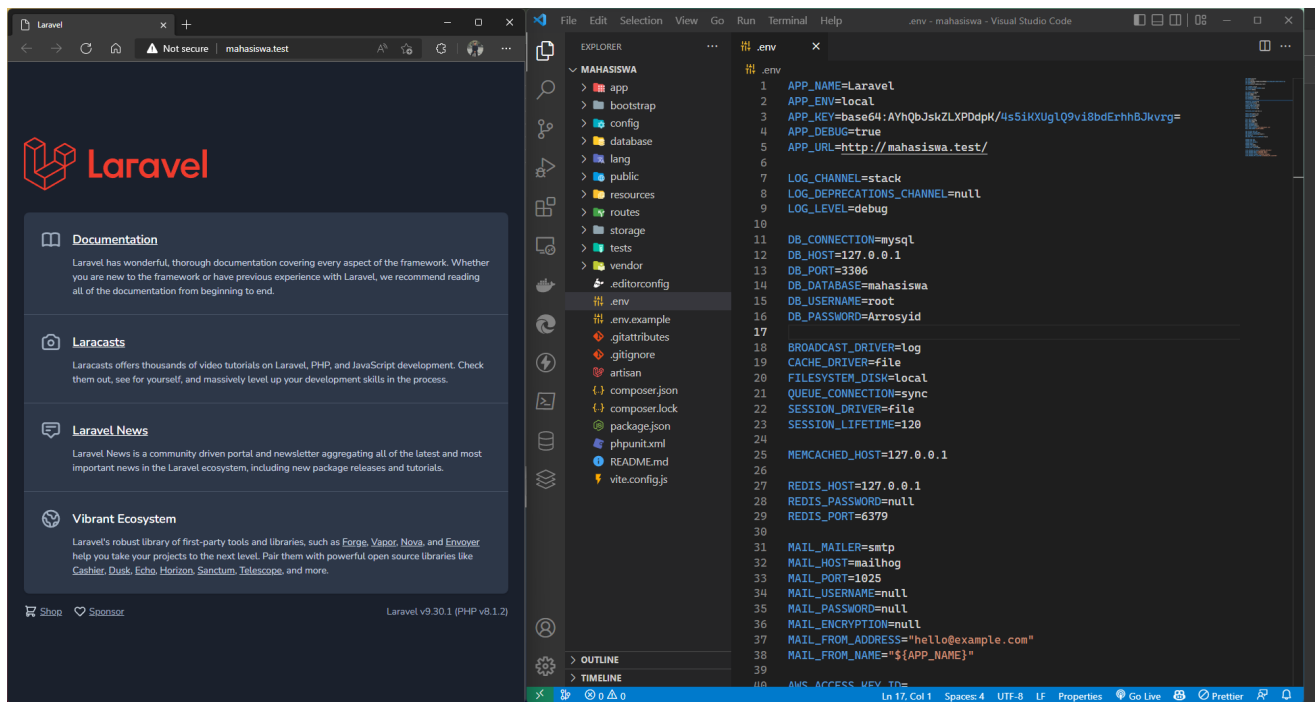
Buat project baru dengan nama project **mahasiswa** dan setup DB pada `.env` sebagai berikut

```
APP_URL=http://mahasiswa.test/
```

```
DB_DATABASE=mahasiswa
```

```
DB_USERNAME=root
```

```
DB_PASSWORD= #kosongkan kalau tidak ada password
```



# Membuat Model, Migration, Controller dan Factory

Membuat Model dan Migration dengan perintah berikut:

```
php artisan make:model Mahasiswa -m
```

 membuat model dan migration

```
php artisan make:controller MahasiswaController
```

 membuat controller

```
php artisan make:factory MahasiswaFactory
```

 membuat factory

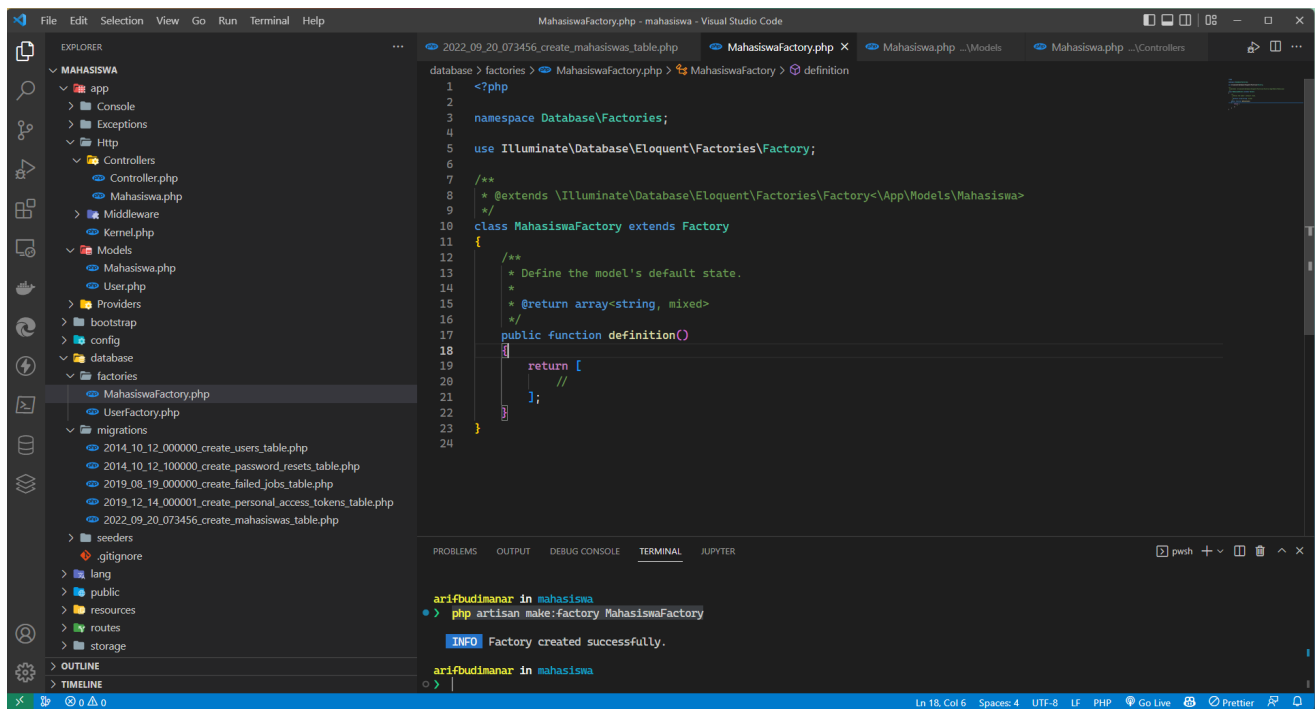
akan ada 4 file baru pada:

```
database\migrations\2022_09_20_073456_create_mahasiswas_table.php
```

```
app\Models\Mahasiswa.php
```

```
app\Http\Controllers\MahasiswaController.php
```

```
database\factories\MahasiswaFactory.php
```



# Edit Migration Mahasiswa

Buka file migration Mahasiswa dan ubah menjadi

```
<?php
```

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

```
return new class extends Migration
{
```

```
    /**
     * Run the migrations.
     *
     * @return void
     */
```

```
    public function up()
    {
```

```
        Schema::create('mahasiswas', function (Blueprint $table) {
            $table->id();
            $table->string('nim');
            $table->string('nama');
            $table->string('alamat');
            $table->timestamps();
        });
```

```

    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('mahasiswas');
    }
};

```

lakukan migration dengan `php artisan migrate` dan pastikan sudah ada tabel baru yaitu **mahasiswa**

## Edit Model Mahasiswa

```

<?php

namespace App\Models;

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

class Mahasiswa extends Model
{
    use HasFactory;
    protected $fillable = [
        'nim',
        'nama',
        'alamat',
    ];
}

```

## Edit Factory MahasiswaFactory

```

<?php

```

```

namespace Database\Factories;

use Illuminate\Database\Eloquent\Factories\Factory;

/**
 * @extends
 * \Illuminate\Database\Eloquent\Factories\Factory<\App\Models\Mahasiswa>
 */
class MahasiswaFactory extends Factory
{
    /**
     * Define the model's default state.
     *
     * @return array<string, mixed>
     */
    public function definition()
    {
        return [
            'nama' => fake()->name(),
            'nim' => fake()->unique()->numberBetween(20220101000,
20220101999),
            'alamat' => fake()->address(),
        ];
    }
}

```

## Edit DatabaseSeeder

```

<?php

namespace Database\Seeders;

// use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;

class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     *
     * @return void
     */
}

```

```

        */
    public function run()
    {
        // \App\Models\User::factory(10)->create();
        // highlight-start
        \App\Models\Mahasiswa::factory(100)->create();
        // highlight-end
        // \App\Models\User::factory()->create([
        //     'name' => 'Test User',
        //     'email' => 'test@example.com',
        // ]);
    }
}

```

lakukan perintah `php artisan db:seed` cek phpMyAdmin maka ada 100 data yang masuk, jika ingin menghapus semua data bisa menggunakan `php artisan migrate:fresh` kalau ingin melakukan hapus data sekaligus membuat data bisa menggunakan `php artisan migrate:fresh --seed`

## Edit Route

```

<?php

use App\Http\Controllers\MahasiswaController;
use App\Models\Mahasiswa;
use Illuminate\Support\Facades\Route;

/*
|-----
|
| Web Routes
|-----
|
| Here is where you can register web routes for your application.
These
| routes are loaded by the RouteServiceProvider within a group
which
| contains the "web" middleware group. Now create something great!
|

```

```

*/

Route::get('/', function () {
    return view('welcome');
});

Route::get('/mahasiswa', [MahasiswaController::class, 'index'])->
>name('mahasiswa.index'); // View Mahasiswa
Route::get('/mahasiswa/create', [MahasiswaController::class,
'create'])->>name('mahasiswa.create'); // Create Mahasiswa
Route::post('/mahasiswa', [MahasiswaController::class, 'store'])->
>name('mahasiswa.store'); // Create Mahasiswa
Route::get('/mahasiswa/{mahasiswa}/edit',
[MahasiswaController::class, 'edit'])->>name('mahasiswa.edit'); //
Edit Mahasiswa
Route::put('/mahasiswa/{mahasiswa}', [MahasiswaController::class,
'update'])->>name('mahasiswa.update'); // Edit Mahasiswa
Route::delete('/mahasiswa/{mahasiswa}delete',
[MahasiswaController::class, 'destroy'])->
>name('mahasiswa.destroy'); // Delete Mahasiswa

```

## Edit Controller MahasiswaController

```

<?php

namespace App\Http\Controllers;

use App\Models\Mahasiswa;
use Illuminate\Http\Request;

class MahasiswaController extends Controller
{
    // View Mahasiswa
    public function index()
    {
        $mahasiswa = Mahasiswa::orderBy('id', 'desc')-
>paginate(10);
        return view('mahasiswa.index', compact('mahasiswa'));
    }

    // Create Mahasiswa
    public function create()

```

```

{
    return view('mahasiswa.create');
}

public function store(Request $request)
{
    $request->validate([
        'nama' => 'required',
        'nim' => 'required|max:11',
        'alamat' => 'required',
    ]);

    Mahasiswa::create($request->post());

    return redirect()->route('mahasiswa.index')-
>with('success', 'Mahasiswa has been created successfully.');
```

}

// Edit Mahasiswa

```

public function edit(Mahasiswa $mahasiswa)
{
    return view('mahasiswa.edit', compact('mahasiswa'));
}

public function update(Request $request, Mahasiswa $mahasiswa)
{
    $request->validate([
        'nama' => 'required',
        'nim' => 'required|max:11',
        'alamat' => 'required',
    ]);

    $mahasiswa->fill($request->post())->save();

    return redirect()->route('mahasiswa.index')-
>with('success', 'Mahasiswa Has Been updated successfully');
```

}

// Delete Mahasiswa

```

public function destroy(Mahasiswa $mahasiswa)
{
    $mahasiswa->delete();
    return redirect()->route('mahasiswa.index')-
```



```
>with('success', 'Mahasiswa has been deleted successfully');  
    }  
}
```

# index.blade.php

Membuat index.blade.php di folder view/mahasiswa

```
</html>  
<!DOCTYPE html>  
<html lang="en">  
  
<head>  
    <meta charset="UTF-8">  
    <title>Laravel CRUD Mahasiswa</title>  
    <link rel="stylesheet"  
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootst  
trap.min.css">  
</head>  
  
<body>  
    <div class="container mt-2">  
        <div class="row">  
            <div class="col-lg-12 margin-tb">  
                <div class="pull-left">  
                    <h2>Laravel CRUD Mahasiswa</h2>  
                </div>  
                <div class="pull-right mb-2">  
                    <a class="btn btn-success" href="{{  
route('mahasiswa.create')}}"> Create Mahasiswa</a>  
                </div>  
            </div>  
        </div>  
        @if ($message = Session::get('success'))  
            <div class="alert alert-success">  
                <p>{{ $message }}</p>  
            </div>  
        @endif  
        <table class="table table-bordered">  
            <thead>  
                <tr>  
                    <th>ID</th>
```

```

        <th>Nama</th>
        <th>NIM</th>
        <th>Alamat</th>
        <th width="200px">Action</th>
    </tr>
</thead>
<tbody>
    @foreach ($mahasiswa as $data)
        <tr>
            <td>{{ $data->id }}</td>
            <td>{{ $data->nama }}</td>
            <td>{{ $data->nim }}</td>
            <td>{{ $data->alamat }}</td>
            <td>
                <form action="{{
route('mahasiswa.destroy', $data->id) }}" method="Post">
                    <a class="btn btn-primary" href="{{
route('mahasiswa.edit', $data->id) }}">Edit</a>
                    @csrf
                    @method('DELETE')
                    <button type="submit" class="btn btn-
danger">Delete</button>
                </form>
            </td>
        </tr>
    @endforeach
</tbody>
</table>
{!! $mahasiswa->links() !!}
</div>
</body>
</html>

```

## create.blade.php

Membuat create.blade.php di folder view/mahasiswa

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">

```

```

<title>Add Mahasiswa</title>
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootst
rap.min.css">
</head>

<body>
  <div class="container mt-2">
    <div class="row">
      <div class="col-lg-12 margin-tb">
        <div class="pull-left mb-2">
          <h2>Add Mahasiswa</h2>
        </div>
        <div class="pull-right">
          <a class="btn btn-primary" href="{{
route('mahasiswa.index')}}"> Back</a>
        </div>
      </div>
    </div>
    @if(session('status'))
    <div class="alert alert-success mb-1 mt-1">
      {{ session('status') }}
    </div>
    @endif
    <form action="{{ route('mahasiswa.store') }}"
method="POST" enctype="multipart/form-data">
      @csrf
      <div class="row">
        <div class="col-xs-12 col-sm-12 col-md-12">
          <div class="form-group">
            <strong>Nama</strong>
            <input type="text" name="nama"
class="form-control" placeholder="Nama">
            @error('nama')
            <div class="alert alert-danger mt-1 mb-1">
              {{ $message }}</div>
            @enderror
          </div>
        </div>
        <div class="col-xs-12 col-sm-12 col-md-12">
          <div class="form-group">
            <strong>NIM</strong>
            <input type="number" name="nim"

```

```

class="form-control" placeholder="NIM">
        @error('nim')
        <div class="alert alert-danger mt-1 mb-1">
{{ $message }}</div>
        @enderror
    </div>
</div>
<div class="col-xs-12 col-sm-12 col-md-12">
    <div class="form-group">
        <strong>Alamat</strong>
        <input type="text" name="alamat"
class="form-control" placeholder="Alamat">
        @error('alamat')
        <div class="alert alert-danger mt-1 mb-1">
{{ $message }}</div>
        @enderror
    </div>
</div>
<button type="submit" class="btn btn-primary ml-
3">Submit</button>
    </div>
</form>
</div>
</body>
</html>

```

## edit.blade.php

Membuat edit.blade.php di folder view/mahasiswa

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <title>Edit Company Form - Laravel 9 CRUD Tutorial</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/boots
trap.min.css">
</head>

<body>

```



```

{{ $message }}</div>
        @enderror
    </div>
</div>
<div class="col-xs-12 col-sm-12 col-md-12">
    <div class="form-group">
        <strong>Alamat</strong>
        <input type="text" name="alamat" value="{{
$mahasiswa->alamat }}" class="form-control"
            placeholder="Alamat">
        @error('alamat')
        <div class="alert alert-danger mt-1 mb-1">
{{ $message }}</div>
        @enderror
    </div>
</div>
<button type="submit" class="btn btn-primary ml-
3">Submit</button>
    </div>
</form>
</div>
</body>

</html>

```

## Edit AppServiceProvider

Mengedit AppServiceProvider agar bisa menggunakan paginate

```

<?php

namespace App\Providers;

use Illuminate\Pagination\Paginator;
use Illuminate\Support\ServiceProvider;

class AppServiceProvider extends ServiceProvider
{
    /**
     * Register any application services.
     *

```

```

    * @return void
    */
    public function register()
    {
        //
    }

    /**
     * Bootstrap any application services.
     *
     * @return void
     */
    public function boot()
    {
        Paginator::useBootstrap();
    }
}

```

## Hasil

jika tidak ada eror maka akan muncul pada <http://mahasiswa.test/mahasiswa>

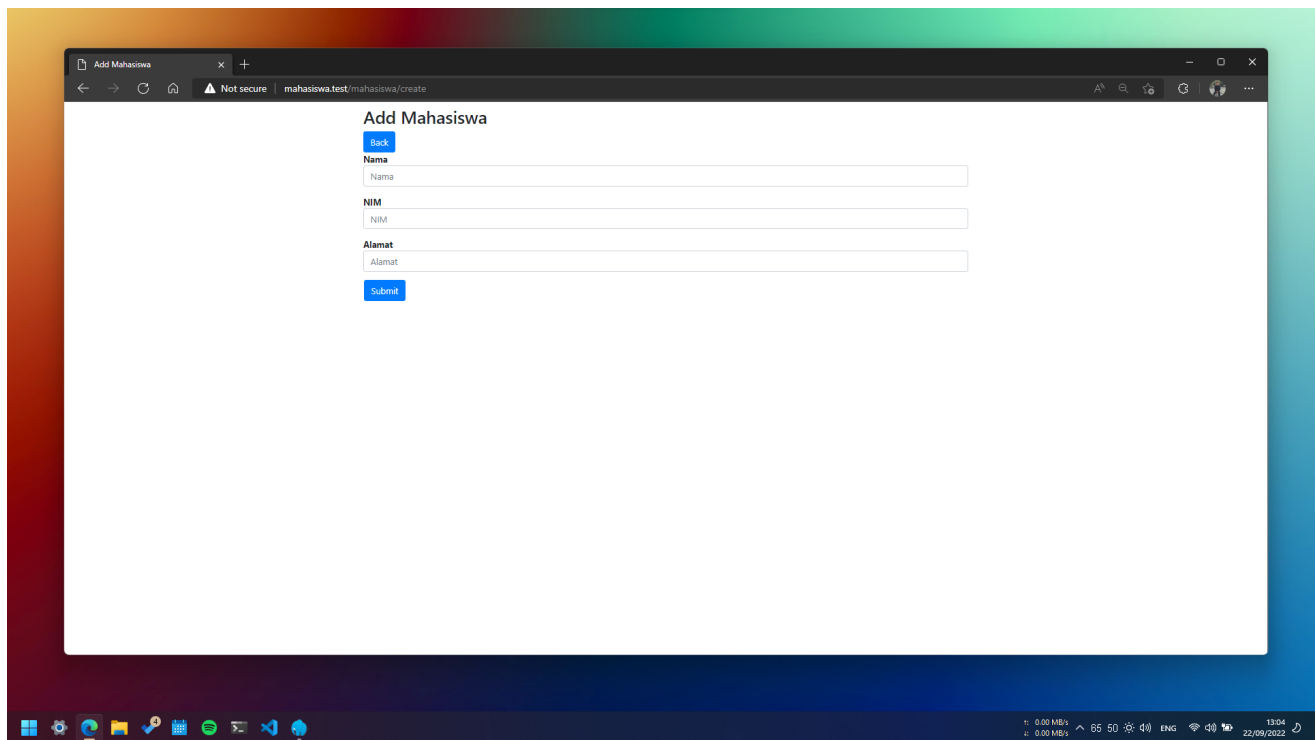
## Index

The screenshot shows a web application titled "Laravel CRUD Mahasiswa". It features a "Create Mahasiswa" button and a table listing students. The table has the following data:

ID	Nama	NIM	Alamat	Action
13	Hank Cormier i	20220101963	72792 Malachi Mountain Apt. 124 Gudrunfurt, SC 40389-1067	<a href="#">Edit</a> <a href="#">Delete</a>
12	Carli Dietrich	20220101226	22064 Ruecker Courts North Coreneville, UT 41238-0027	<a href="#">Edit</a> <a href="#">Delete</a>
11	Tia Durgan	20220101704	1320 Winona Causeway Apt. 790 East Krystallhaven, WV 83928	<a href="#">Edit</a> <a href="#">Delete</a>
10	Kian McClure	20220101223	164 Audra Gateway Apt. 075 West Daniella, MA 49176	<a href="#">Edit</a> <a href="#">Delete</a>
9	Dolly Ferry PhD	20220101388	6870 Senger Isle Apt. 597 O'Connellbury, TN 77223-1068	<a href="#">Edit</a> <a href="#">Delete</a>
8	Mr. Jett Crist	20220101306	941 Fay Dam North Rosendoberg, VT 97154	<a href="#">Edit</a> <a href="#">Delete</a>
7	Darnel Crona	20220101821	4698 Wintheiser Drives North Rudy, NE 47274	<a href="#">Edit</a> <a href="#">Delete</a>
6	Kyle Boehm	20220101934	802 Ward Hills Suite 311 Port Davion, NM 51826-4768	<a href="#">Edit</a> <a href="#">Delete</a>
5	Mr. Darnick Torp	20220101752	48983 Humberto Parkways Goyetteland, OR 34968-4482	<a href="#">Edit</a> <a href="#">Delete</a>
4	Janessa Hudson	20220101541	1082 Itzel Trail New Anika, CT 06069	<a href="#">Edit</a> <a href="#">Delete</a>

The browser's address bar shows "http://mahasiswa.test/mahasiswa". The page also includes a "Create Mahasiswa" button and a pagination bar at the bottom of the table.

# Create



The screenshot shows a web browser window with the address bar displaying 'mahasiwa.test/mahasiwa/create'. The page title is 'Add Mahasiswa'. It features a form with the following fields: 'Nama' (Name), 'NIM' (National Identification Number), and 'Alamat' (Address). Each field has a corresponding input box. Above the 'Nama' field is a blue 'Back' button, and below the 'Alamat' field is a blue 'Submit' button. The browser's status bar at the bottom shows system icons and the time '13:04' on '22/09/2022'.

Add Mahasiswa

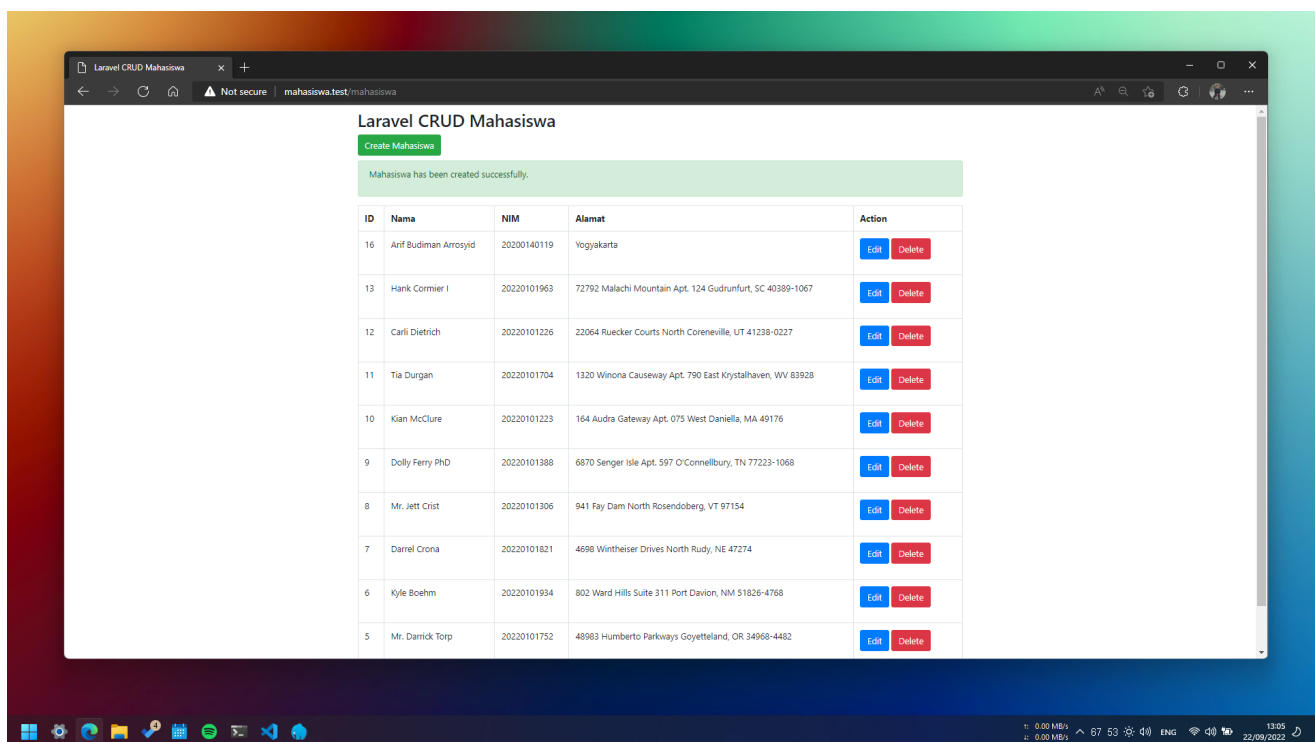
[Back](#)

Nama

NIM

Alamat

[Submit](#)



The screenshot shows a web browser window with the address bar displaying 'mahasiwa.test/mahasiwa'. The page title is 'Laravel CRUD Mahasiswa'. A green button labeled 'Create Mahasiswa' is at the top left. Below it, a green message box states 'Mahasiswa has been created successfully.' The main content is a table with columns: ID, Nama, NIM, Alamat, and Action. The table contains 10 rows of student data. Each row has 'Edit' and 'Delete' buttons in the Action column. The browser's status bar at the bottom shows system icons and the time '13:05' on '22/09/2022'.

Laravel CRUD Mahasiswa

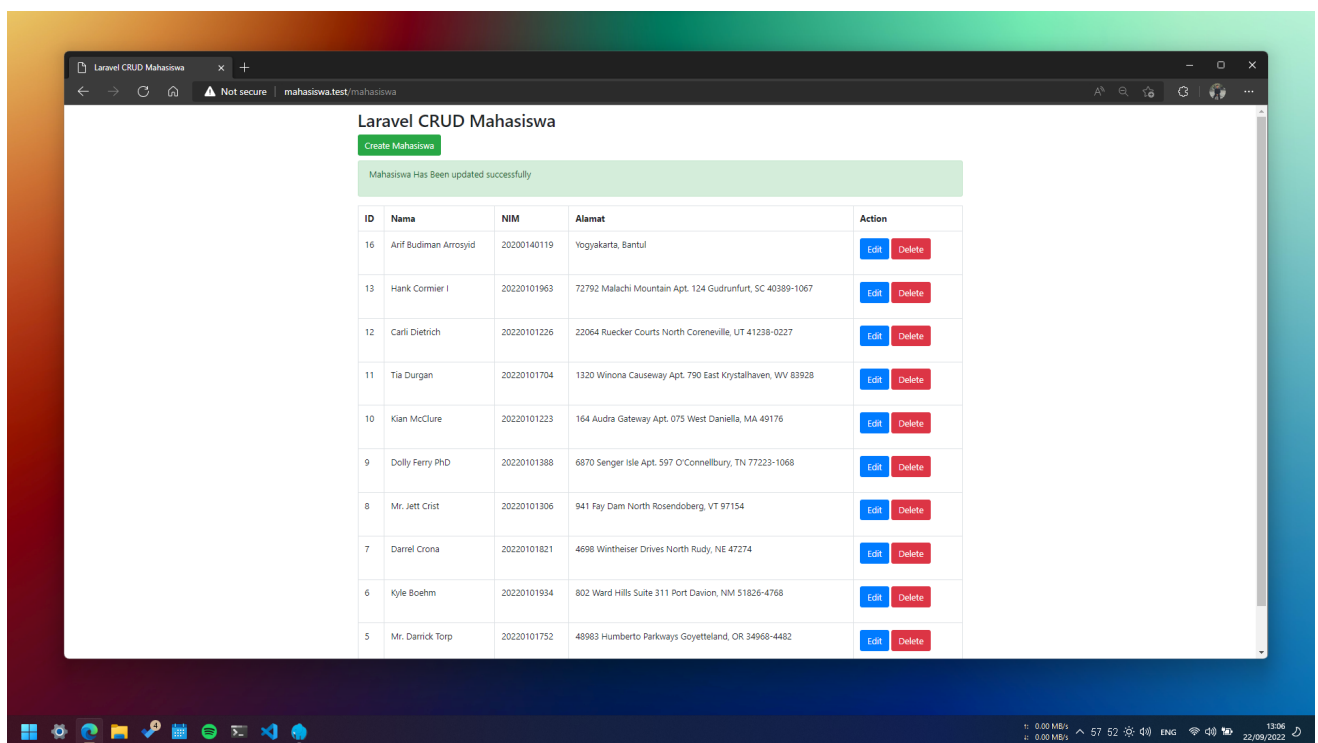
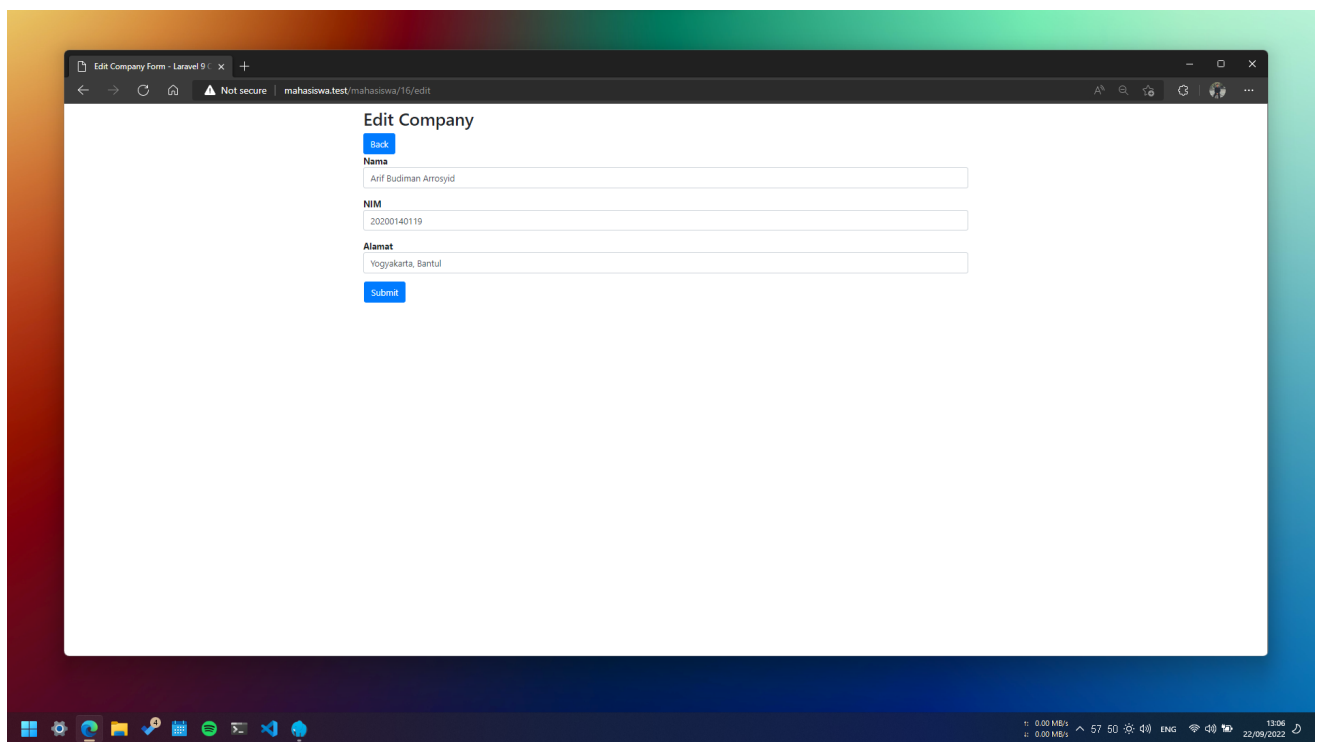
[Create Mahasiswa](#)

Mahasiswa has been created successfully.

ID	Nama	NIM	Alamat	Action
16	Aarif Budiman Amosyid	20200140119	Yogyakarta	<a href="#">Edit</a> <a href="#">Delete</a>
13	Hark Cormier I	20220101963	72792 Malachi Mountain Apt. 124 Gudrunfurt, SC 40389-1067	<a href="#">Edit</a> <a href="#">Delete</a>
12	Carli Dietrich	20220101226	22064 Ruecker Courts North Coreneville, UT 41238-0227	<a href="#">Edit</a> <a href="#">Delete</a>
11	Tia Durgan	20220101704	1320 Winona Causeway Apt. 790 East Krystalhaven, WV 83928	<a href="#">Edit</a> <a href="#">Delete</a>
10	Kian McClure	20220101223	164 Audra Gateway Apt. 075 West Daniella, MA 49176	<a href="#">Edit</a> <a href="#">Delete</a>
9	Dolly Ferry PhD	20220101388	6870 Senger Isle Apt. 597 O'Connellbury, TN 77223-1068	<a href="#">Edit</a> <a href="#">Delete</a>
8	Mr. Jett Crist	20220101306	941 Fay Dam North Rosendoberg, VT 97154	<a href="#">Edit</a> <a href="#">Delete</a>
7	Darnel Crona	20220101821	4698 Wintheiser Drives North Rudy, NE 47274	<a href="#">Edit</a> <a href="#">Delete</a>
6	Kyle Boehm	20220101934	802 Ward Hills Suite 311 Port Davion, NM 51826-4768	<a href="#">Edit</a> <a href="#">Delete</a>
5	Mr. Darrick Torp	20220101752	48983 Humberto Parkways Goyetteland, OR 34968-4482	<a href="#">Edit</a> <a href="#">Delete</a>

# Edit





Delete

Laravel CRUD Mahasiswa

Create Mahasiswa

Mahasiswa has been deleted successfully

ID	Nama	NIM	Alamat	Action
13	Hank Cormier I	20220101963	72792 Malachi Mountain Apt. 124 Gudrunfurt, SC 40389-1067	<a href="#">Edit</a> <a href="#">Delete</a>
12	Carli Dietrich	20220101226	22064 Ruecker Courts North Coreneville, UT 41238-0227	<a href="#">Edit</a> <a href="#">Delete</a>
11	Tia Durgan	20220101704	1320 Winona Causeway Apt. 790 East Krystalhaven, WV 83928	<a href="#">Edit</a> <a href="#">Delete</a>
10	Klan McClure	20220101223	164 Audra Gateway Apt. 075 West Daniella, MA 49176	<a href="#">Edit</a> <a href="#">Delete</a>
9	Dolly Ferry PhD	20220101388	6870 Senger Isle Apt. 597 O'Connellbury, TN 77223-1068	<a href="#">Edit</a> <a href="#">Delete</a>
8	Mr. Jett Crist	20220101306	941 Fay Dam North Rosendoberg, VT 97154	<a href="#">Edit</a> <a href="#">Delete</a>
7	Darrel Crona	20220101821	4698 Wintheiser Drives North Rudy, NE 47274	<a href="#">Edit</a> <a href="#">Delete</a>
6	Kyle Boehm	20220101934	802 Ward Hills Suite 311 Port Davion, NM 51826-4768	<a href="#">Edit</a> <a href="#">Delete</a>
5	Mr. Darrick Torp	20220101752	48983 Humberto Parkways Goyetteland, OR 34968-4482	<a href="#">Edit</a> <a href="#">Delete</a>
4	Janessa Hudson	20220101541	1082 Itzel Trail New Anika, CT 06069	<a href="#">Edit</a> <a href="#">Delete</a>

