**LAPORAN PRAKTIKUM PEMROGRAMAN WEB**

**PRAKTIKUM 11 – MVC ROUTE**



Disusun oleh:

|  |  |  |
| --- | --- | --- |
| Nama | : | Alfath Roziq Widhayaka |
| Nim | : | L0122012 |
| Kelas | : | A |

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI DAN SAINS DATA**

**UNIVERSITAS SEBELAS MARET**

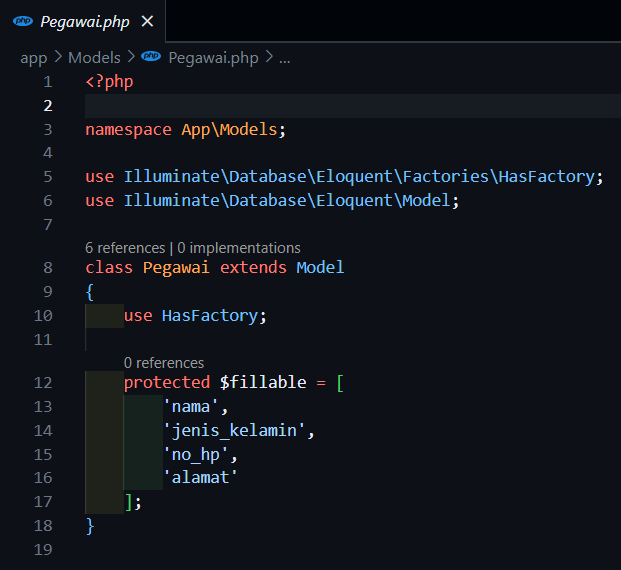
**2024**

**BAB I**

1. **Penjelasan Source Code**

Tugas praktikum Pemrograman Web week ke 11 ini saya membuat project Laravel dengan menerapkan materi MVC Route. MVC Route disini yang saya buat adalah data pegawai sederhana yang juga menerapkan sistem CRUD. Fungsi MVC Route disini sebagai untuk alur kerja aplikasi menjadi lebih terstruktur dan mudah untuk dikelola. Route dalam Laravel berfungsi untuk menentukan URL yang akan diakses oleh pengguna dan menghubungkannya dengan Controller yang sesuai. Cara untuk membuat MVC Route dengan mudah yaitu dengan mengetik pada terminal **php artisan make pegawai -**mrc. Berikut penjelasan untuk masing-masing pembuatan MVC Route nya.

1. **Model**

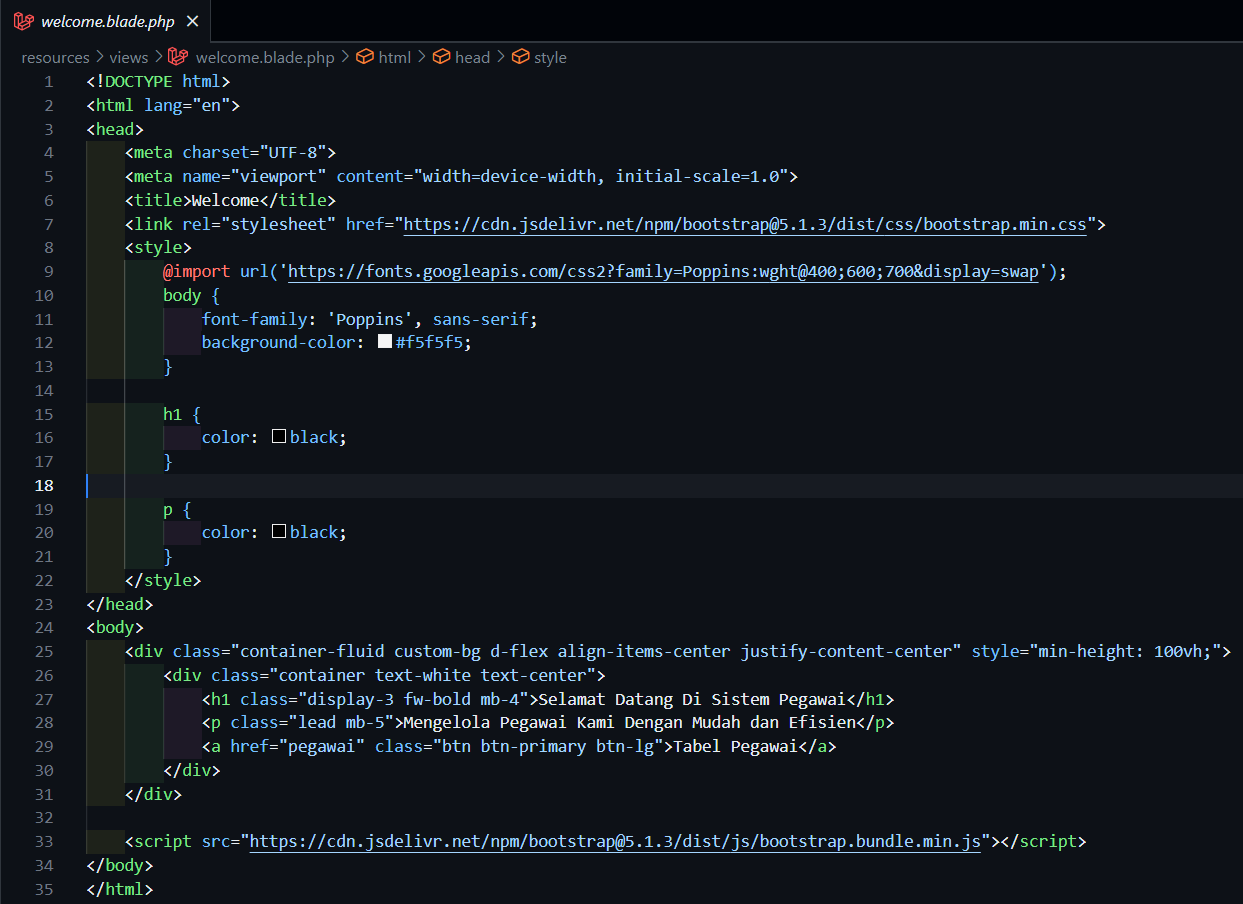
****

*Gambar 1 Pegawai.php*

**Model** Pegawai ini merupakan sebuah kelas yang digunakan untuk berinteraksi dengan tabel "**pegawai**" di database. Kelas ini mewarisi fitur-fitur dari kelas induk Model yang disediakan oleh **Eloquent**, sehingga dapat memanfaatkan kemampuan ORM yang kuat. Kelas ini menggunakan **trait HasFactory** untuk memungkinkan pembuatan data palsu selama pengujian. Properti **$fillable** mendefinisikan atribut-atribut yang dapat diisi secara massal saat membuat atau memperbarui data pegawai, seperti **nama, jenis\_kelamin, no\_hp,** dan **alamat**.

1. **View**

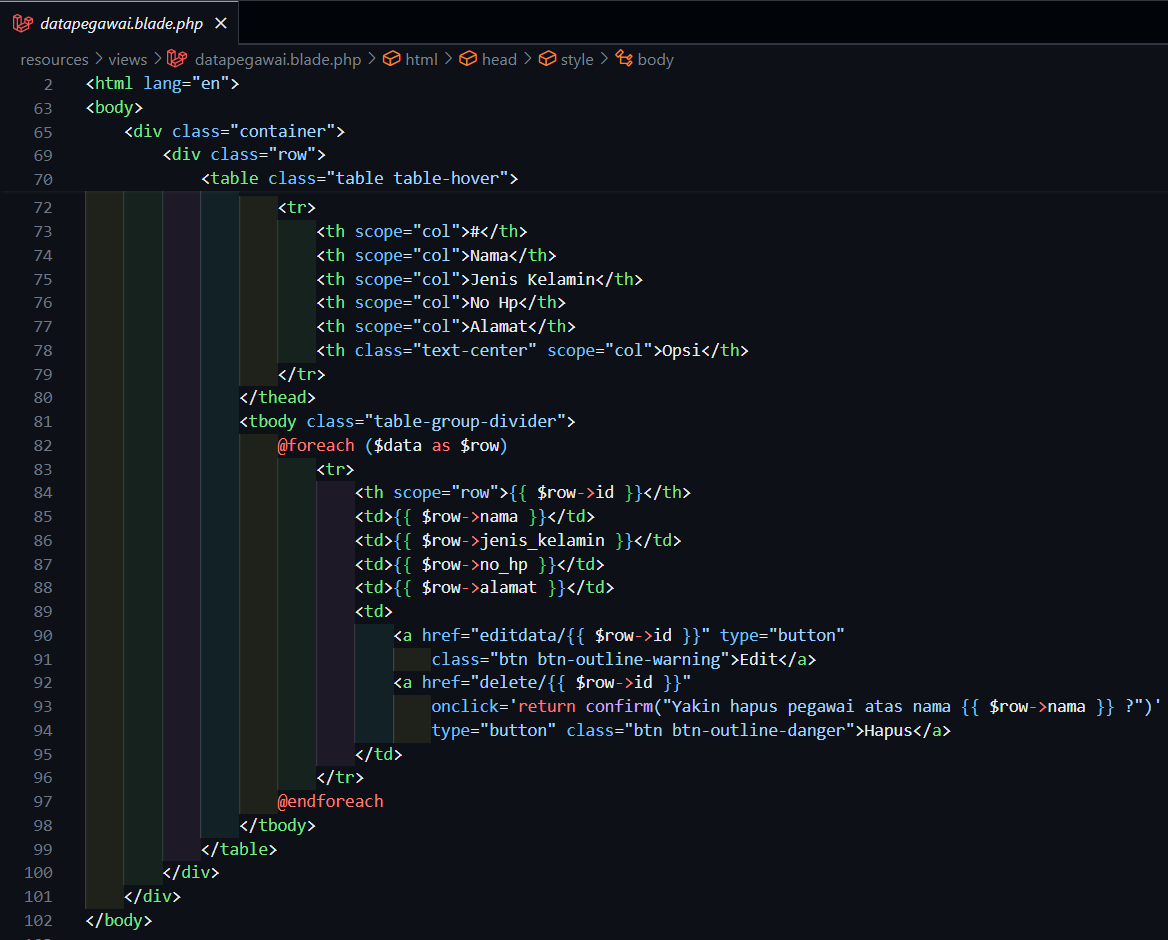
* **Welcome**

****

*Gambar 2 welcome.blade.php*

Kode View milik **welcome** diatas merupakan kode HTML yang isinya nanti dapat menampilkan halaman awal yang berisi judul dan tombol yang bilamana ditekan akan berpindah ke halaman **tabel data** pegawai milik **datapegawai.blade.php**.

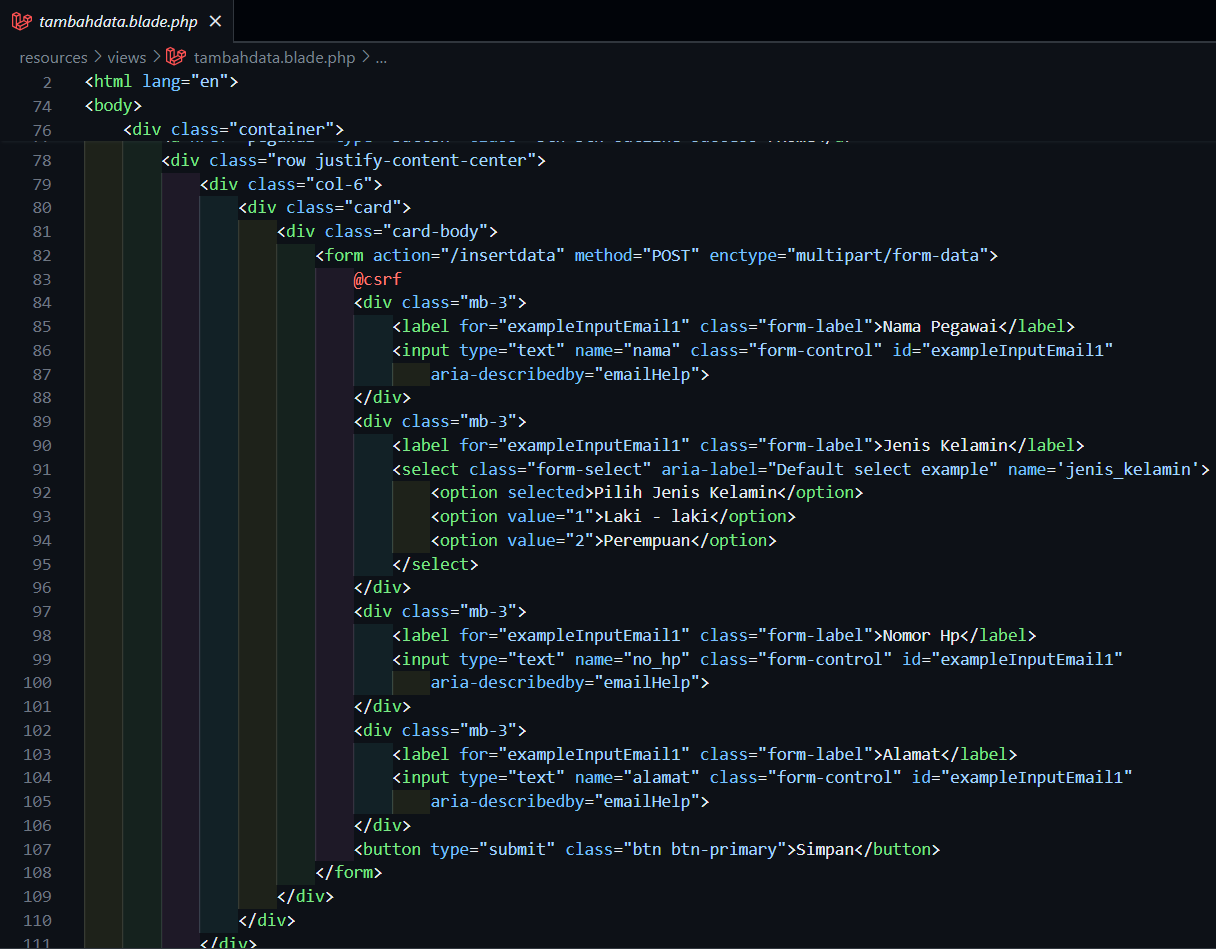
* **Tabel**

****

*Gambar 3 datapegawai.blade.php*

Kode HTML milik **datapegawai.blade.php** diatas merupakan kode yang nantinya akan menampilkan halaman yang berisi tabel data pegawai. Halaman ini pengguna dapat melihat daftar pegawai melalui tabel dan juga dapat melakukan menambah, mengedit, dan menghapus pegawai.

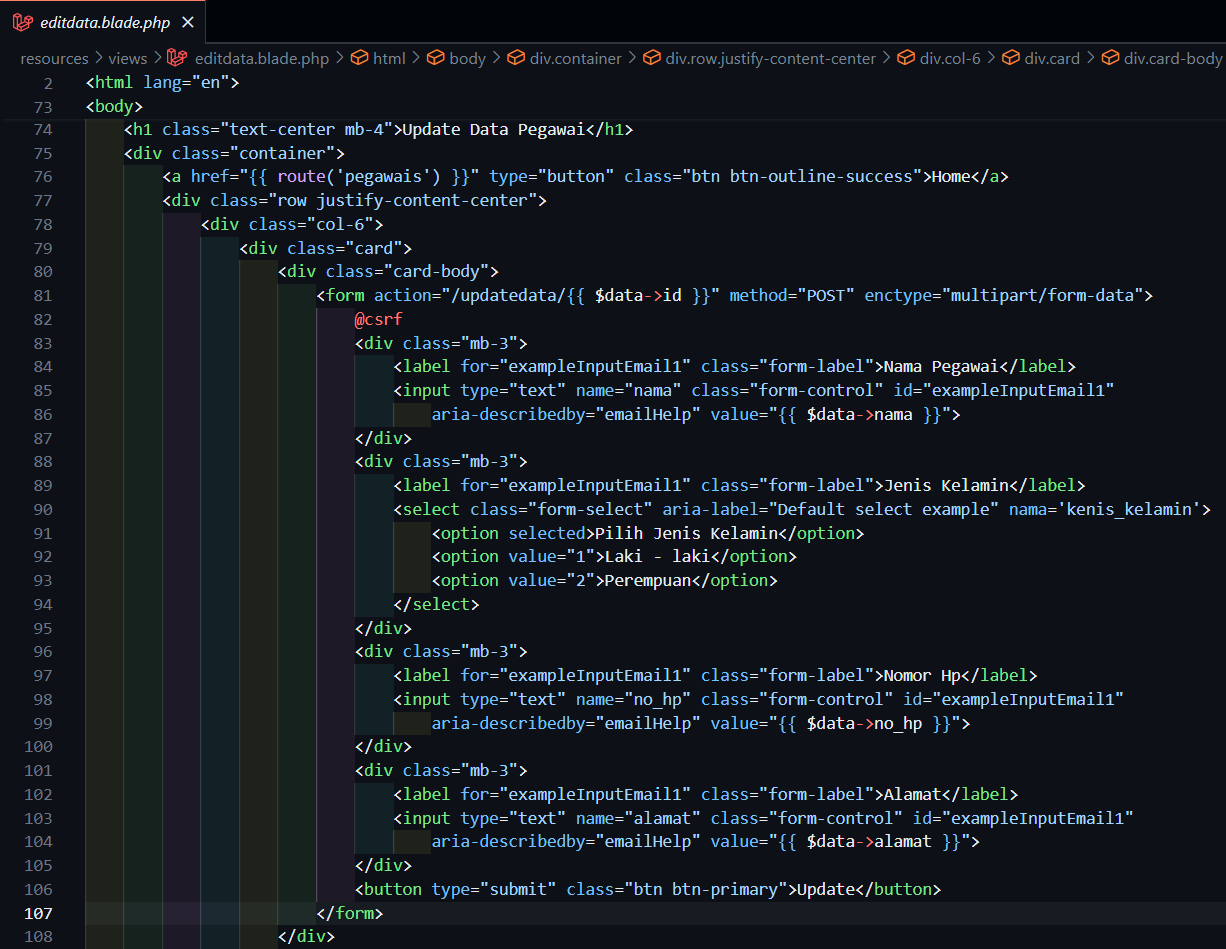
* **Tambah Data**

****

*Gambar 4 tambahdata.blade.php*

Kode HTML milik **tambahdata.blade.php** diatas merupakan kode yang nantinya akan menampilkan halaman form untuk pengguna dalam menambahkan pegawai.

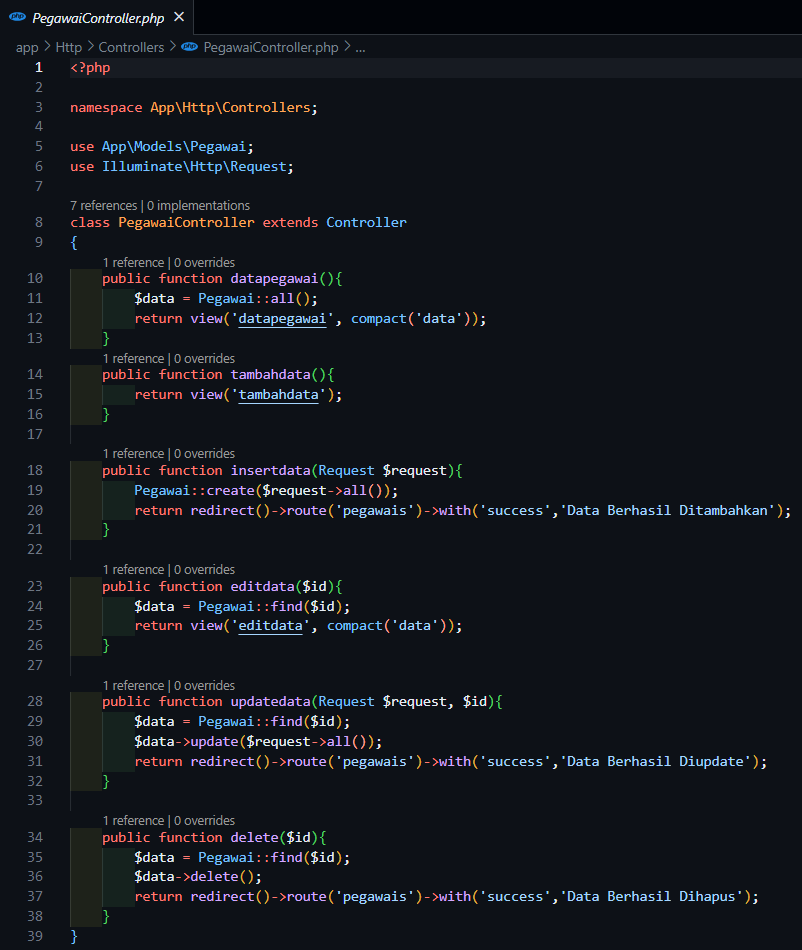
* **Edit Data**

****

*Gambar 5 editdata.blade.php*

Kode HTML milik **editdata.blade.php** diatas merupakan kode yang nantinya akan menampilkan halaman form untuk pengguna dalam mengedit pegawai.

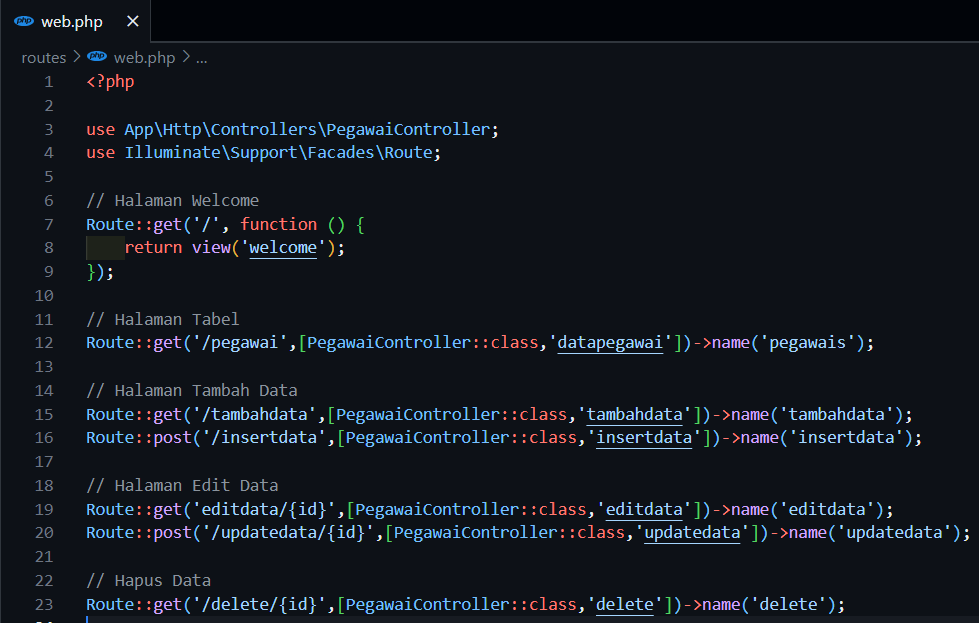
1. **Controller**

****

*Gambar 6 PegawaiControllerphp*

Kode ini merupakan sebuah Controller Laravel bernama **PegawaiController** yang bertanggung jawab untuk menangani logika terkait **pegawai**. Controller ini memiliki beberapa metode yang memungkinkan pengguna untuk melihat daftar pegawai (**datapegawai**), menambahkan pegawai baru (**tambahdata** dan **insertdata**), mengedit data pegawai (**editdata** dan **updatedata**), serta menghapus data pegawai (**delete**). Metode-metode ini menggunakan model Pegawai untuk berinteraksi dengan data di database, dan mengembalikan tampilan yang sesuai atau mengarahkan pengguna ke rute yang tepat. Secara keseluruhan, PegawaiController menyediakan fungsionalitas CRUD (Create, Read, Update, Delete) yang dibutuhkan untuk mengelola data pegawai dalam aplikasi Laravel.

1. **Route**



*Gambar 7 web.php*

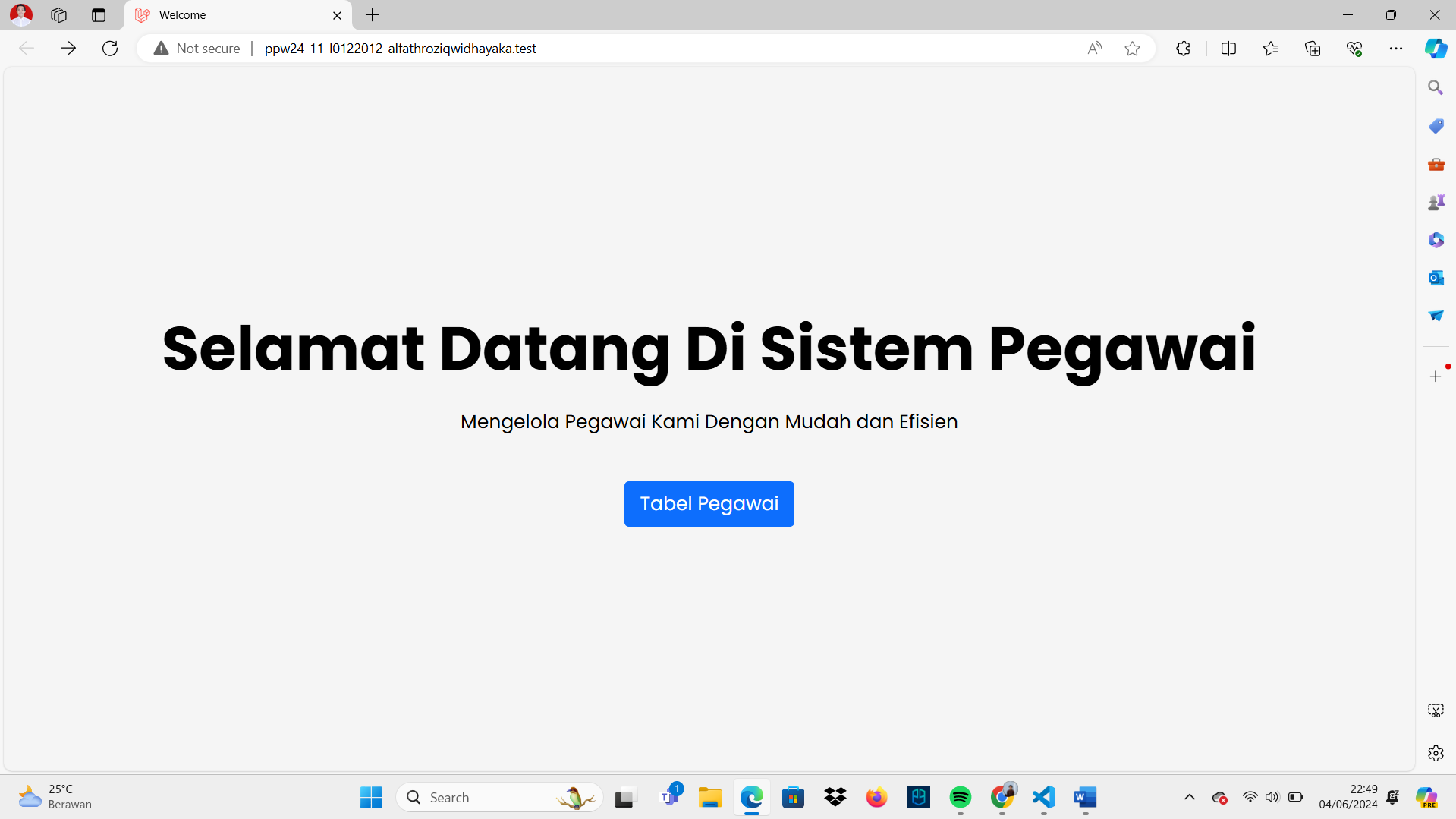
Kode ini merupakan definisi rute (**route**) dalam aplikasi Laravel. Rute-rute ini mengatur bagaimana aplikasi akan merespons permintaan HTTP dari pengguna. Pertama, terdapat rute untuk halaman welcome yang menampilkan tampilan awal aplikasi. Kemudian, ada rute untuk menampilkan daftar pegawai yang ditangani oleh metode **datapegawai** pada **PegawaiController**. Selanjutnya, ada rute untuk menampilkan halaman tambah data pegawai, yang terdiri dari dua rute yaitu satu untuk menampilkan formulirnya dan satu lagi untuk memproses data yang dikirimkan. Selain itu, terdapat rute untuk menampilkan halaman edit data pegawai, yang juga terdiri dari dua rute yaitu satu untuk menampilkan formulirnya dan satu lagi untuk memproses pembaruan data. Terakhir, ada rute untuk menghapus data pegawai. Keseluruhan rute-rute ini menggunakan nama (**name**) untuk memudahkan pengaksesan dan pengiriman URL.

**BAB II**

1. **Hasil Output**

Berikut merupakan hasil output dari project laravel yang saya buat menerapkan materi MVC Route.

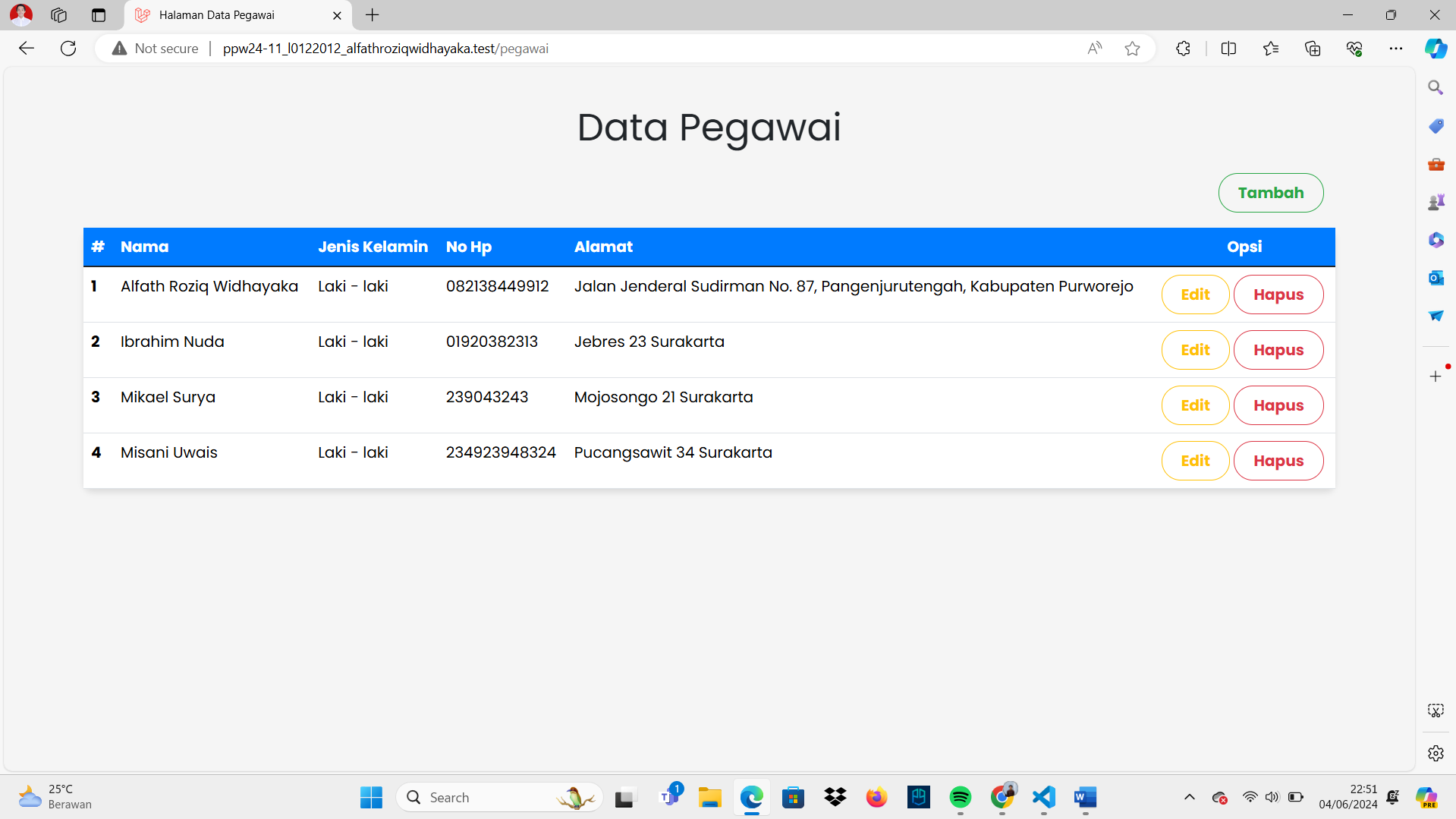
1. **Welcome**

****

*Gambar 8 welcome*

Gambar diatas merupakan hasil output ketika project laravel dijalankan. Terlihat ini merupakan halaman welcome yang menampilkan judul dan satu tombol yang nantinya akan mengarah ke tabel data pegawai.

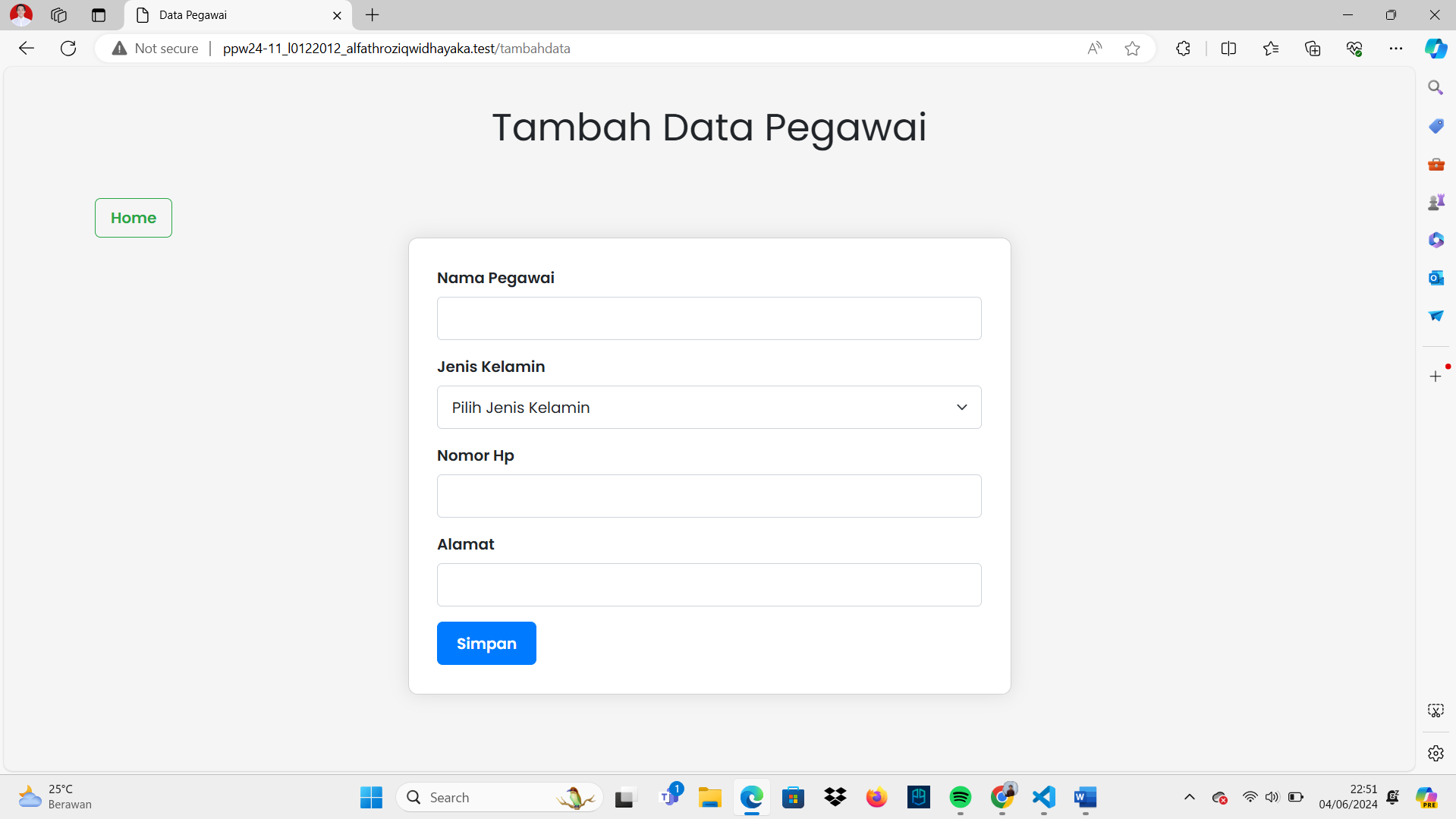
1. **Tabel Data**

****

*Gambar 8 tabel data*

Gambar diatas merupakan hasil output ketika pengguna telah menekan tombol **Tabel Pegawai** pada halaman welcome. Disini terlihat halaman menampilkan tabel dengan isinya yaitu daftar pegawai yang dapat ditambah, diedit, dan dihapus.

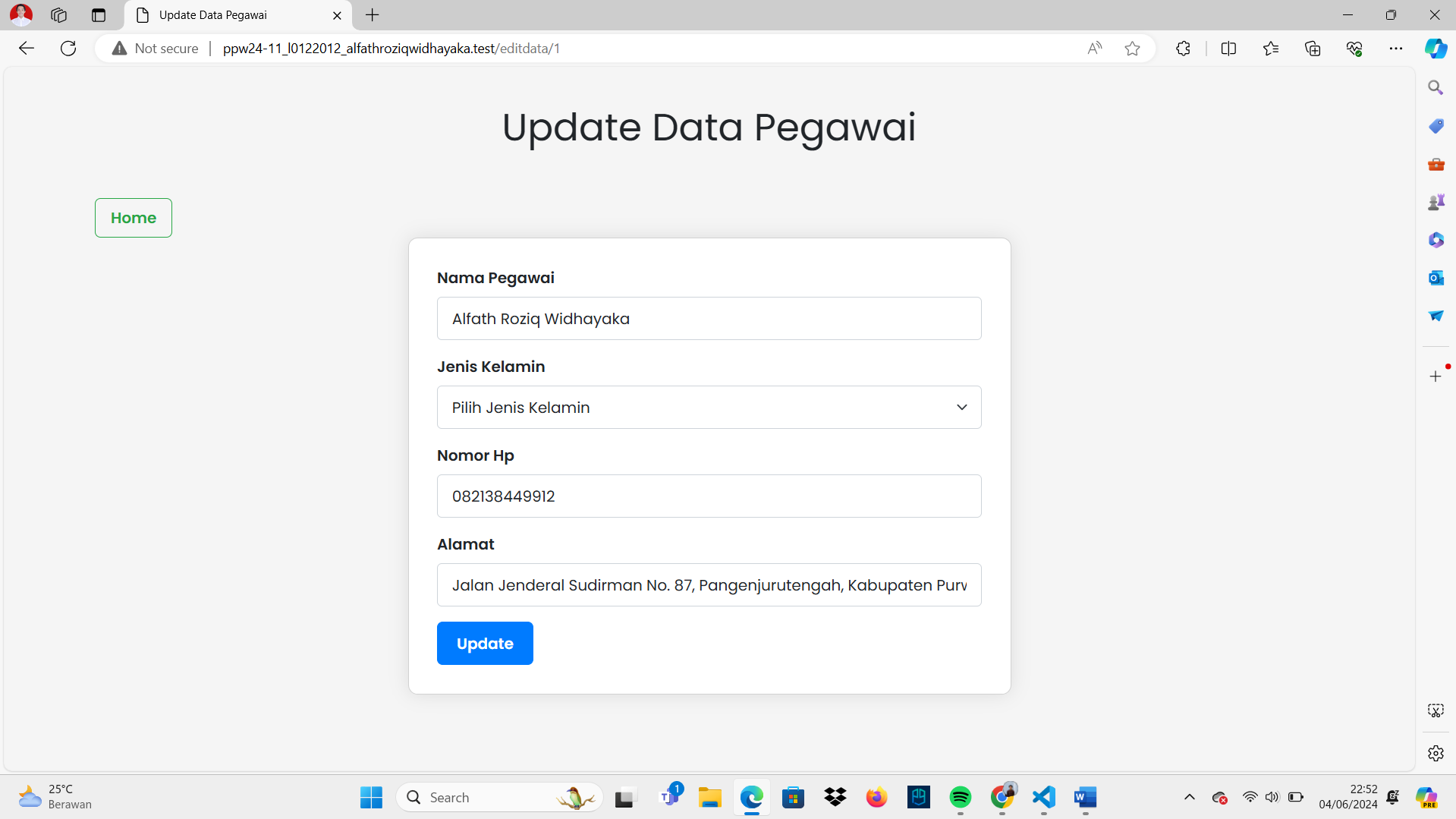
1. **Tambah Data**

****

*Gambar 9 tambah data*

Jika pengguna menekan tombol **Tambah** pada halaman tabel data, maka gambar diatas merupakan halaman yang diberikan kepada pengguna untuk menambahkan data pegawai yang ingin ditambah

1. **Edit Data**

****

*Gambar 10 edit data*

Ketika pengguna memilih tombol **Edit**, maka gambar diatas merupakan tampilan yang diberikan untuk pengguna mengubah atau memodifikasi isi data pada tabel.

**BAB III**

1. **Kesimpulan**

Untuk praktikum minggu ke 11 ini, saya telah membuat project Laravel yang mengimplementasikan **MVC Route**, yang mencakup sistem data pegawai sederhana dengan fungsionalitas **CRUD**. Proyek ini terdiri dari kelas **Model** untuk berinteraksi dengan database, **View** untuk menampilkan data, **Controller** untuk menangani logika website, dan **Route** untuk menentukan kemana arah jika salah satu tombol ditekan. Output dari project ini meliputi halaman welcome, tabel yang menampilkan data pegawai, dan formulir untuk menambah dan mengedit data pegawai.