Nama :

NIS :

MODUL 9

1. Tujuan
2. Dapat menjelaskan CRUD dan API
3. Dapat menjelaskan stuktur CRUD dan API
4. Dapat menjelaskan konsep CRUD dan API
5. Dapat mengenal dan menerapkan blade framework Laravel
6. Dapat mengenal dan menerapkan CRUD framework Laravel
7. Dapat mengenal dan menerapkan API framework Laravel
8. Dasar teori

CRUD adalah singkatan dari Create, Read, Update, dan Delete. Keempat hal tersebut merupakan perintah dengan peran yang esensial dalam sebuah aplikasi database yang sifatnya relational.

Seperti namanya, keempat fungsi ini merupakan fondasi kerja sebuah database. Bayangkan saja kinerja sebuah database itu sebagai sebuah perpustakaan. Agar operasional perpustakaan dapat berjalan, kamu perlu sebuah sistem pendataan buku-buku yang ada di dalamnya. Dengan analogi tersebut, berikut adalah penjelasan tentang fungsi-fungsi dalam CRUD:

* **Create**

Fungsi Create berguna untuk membuat entri atau record baru dalam sebuah tabel di dalam database. Di dalam aplikasi berbasis SQL, fungsi create seringkali disebut dengan insert.

* **Read**

Fungsi read ini berguna untuk membaca entri data yang sudah terdaftar dalam sebuah database. Namun, fungsi ini tidak akan memberikan akses untuk mengubah data yang telah tersimpan di dalam database.

* **Update**

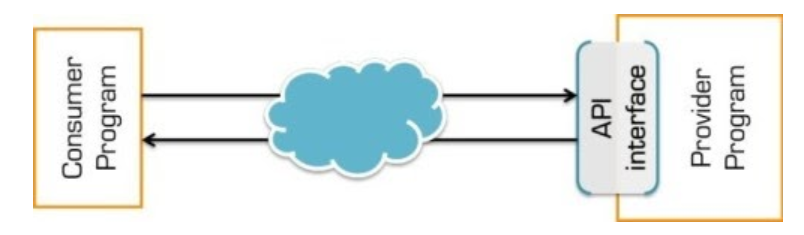
Fungsi update ini berperan penting untuk memperbarui entri data ketika informasi di dalamnya memerlukan perubahan. Dengan fungsi ini, kamu dapat mengubah detail dari sebuah entri di database.

* **Delete**

Seperti namanya, fungsi delete ini berguna untuk menghapus entri data yang sudah tidak diperlukan dalam sebuah database. Ketika menggunakan fungsi ini, kamu akan mengakses detail terkait sebuah entri dan kemudian memberikan perintah kepada sistem untuk menghilangkannya dari database.

API adalah singkatan dari Application Programming Interface. API sendiri merupakan interface yang dapat menghubungkan satu aplikasi dengan aplikasi lainnya. Dengan kata lain, peran API adalah sebagai perantara antar berbagai aplikasi berbeda, baik dalam satu platform yang sama atau pun lintas platform.

Perumpamaan yang bisa digunakan untuk menjelaskan API adalah seorang pelayan di restoran. Tugas pelayan tersebut adalah menghubungkan tamu restoran dengan juru masak. Jadi, tamu cukup memesan makanan sesuai daftar menu yang ada dan pelayan memberitahukannya ke juru masak. Nantinya, pelayan akan kembali ke tamu tadi dengan masakan yang sudah siap sesuai pesanan.



Pada penggunaannya, terdapat empat jenis API sesuai dengan hak aksesnya. Apa saja?

1. Public API

Public API sering disebut dengan Open API. Sesuai dengan namanya, Public API adalah jenis API yang bisa digunakan siapa saja dalam lintas platform. Selain itu, jenis API ini paling mudah digunakan. Anda cukup mendaftar atau langsung memanfaatkannya di aplikasi Anda. Contohnya, API untuk Google Maps dan API untuk data Covid Indonesia.

1. Private API

Berbeda dengan public API, Private API adalah jenis API yang tidak terbuka untuk penggunaan umum. Biasanya, API jenis ini dibuat untuk keperluan internal dalam pengembangan aplikasi tertentu. Misalnya, API dari back end yang digunakan untuk mengakses front end dari sebuah website. Atau, aplikasi untuk pengembangan aplikasi mobile.

1. Partner API

Partner API bisa digunakan untuk kepentingan umum tapi sebatas pihak yang sudah memiliki izin penggunaannya. Seperti halnya public API, Anda perlu mendaftar kepada penyedia API terlebih dahulu. Kemudian, menggunakannya hanya di aplikasi tertentu sesuai perjanjian. Contohnya, API Pinterest.

1. Composite API

Composite API merupakan API yang menyimpan data dari berbagai server atau hosting dalam satu tempat. Tentunya, hal tersebut sangat menghemat waktu bagi pengguna. Itu karena pengguna bisa mendapatkan berbagai jenis data hanya dalam sekali akses.

Lalu, apa sebenarnya manfaat API dalam pengembangan website atau aplikasi?

1. Memudahkan Membangun Aplikasi yang Fungsional

Dengan menggunakan API, akan lebih mudah untuk membuat aplikasi yang fungsional dan kompleks. Tanpa perlu menambahkan data secara manual, aplikasi yang dikembangkan akan memiliki fitur dari aplikasi tujuan.

Sebagai contoh, pada aplikasi Gojek. Sebagai sebuah platform layanan transportasi, peran peta sangatlah penting. Namun, Gojek tidak perlu mengembangkan aplikasi peta sendiri. Dengan API, aplikasi tersebut cukup mengambil data dari Google Maps.

Penggunaan API ini cukup membantu membuat platform Gojek semakin besar. Alasannya, developer cukup mengembangkan layanan lain karena penggunaan peta sebagai elemen utama dipastikan berjalan dengan baik.

1. Pengembangan Aplikasi Menjadi Lebih Efisien

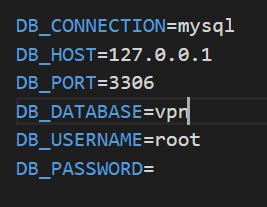
Dengan adanya API, Anda tidak perlu melakukan komunikasi langsung dengan aplikasi lain yang ingin dihubungkan. Cukup dengan komunikasi melalui API. Hal ini sangat membantu, terutama jika Anda ingin membangun aplikasi lintas platform dengan berbagai layanan sekaligus.

Sebagai contoh, Anda membangun website pemesanan tiket online untuk berbagai maskapai di dunia. Dengan bantuan API, Anda cukup melakukan integrasi untuk masing-masing layanan maskapai tersebut. Jadi, tidak perlu lagi melakukan komunikasi manual berupa update harga atau tersedianya tempat duduk.

1. Meringankan Beban Server

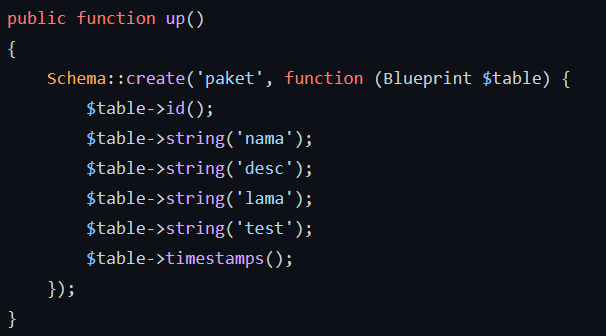
Dengan menggunakan API, Anda tidak perlu menyimpan semua data yang dibutuhkan di server Anda sendiri. Cukup meminta API untuk mendapatkan data terbaru dari server aplikasi asal. Dengan kondisi ini, server Anda tidak akan terbebani. Pada akhirnya, mengurangi resiko website tidak dapat diakses karena server down.

1. Langkah pengerjaan
2. Buat project baru
3. Buat project baru di dalam folder Laravel dengan nama project **crud-laravel** dengan mengetikan command berikut di CMD folder “composer create-project laravel/laravel crud-laravel”
4. Setelah selesai mendownload Laravel selanjutnya masuk ke directory project yang kita buat dengan mengetikan command berikut “**cd crud-laravel**”
5. Selanjutnya untuk membuka code editor ketikan command berikut di CMD “**code .**”
6. Selanjutnya jalankan xampp dengan service apache dan mysql
7. Selanjutnya jalankan Laravel dengan mengetikan command berikut di CMD “**php artisan serv**”
8. Setelah berjalan cari file dengan nama **.env** dan rubah configurasi database seperti gambar di bawah

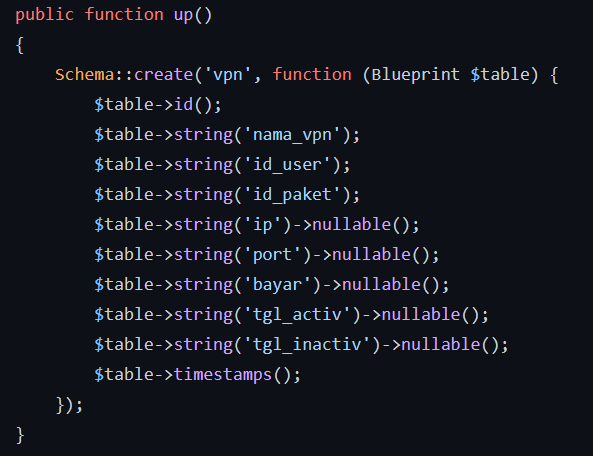


1. Selanjutnya buat database dengan nama **vpn** di phpMyAdmin
2. Membuat buat migration
3. Di CMD jalankan command berikut untuk membuat migration “php artisan make:migration create\_paket\_table” dan “php artisan make:migration create\_vpn\_table”
4. Selanjutnya buka file migration yang kita buat sebelumnya dan buat seperti gambar di bawah

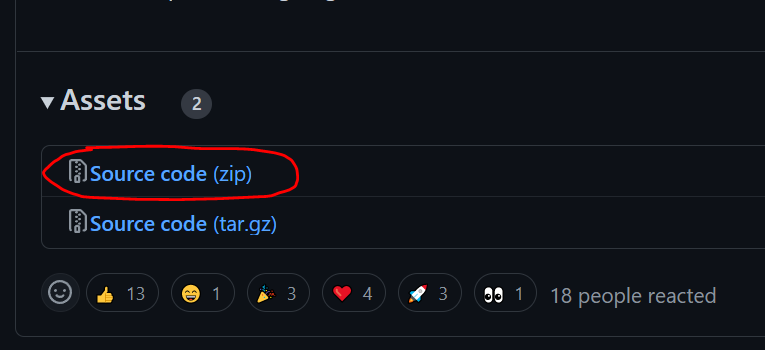
Paket



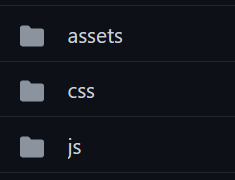
Vpn



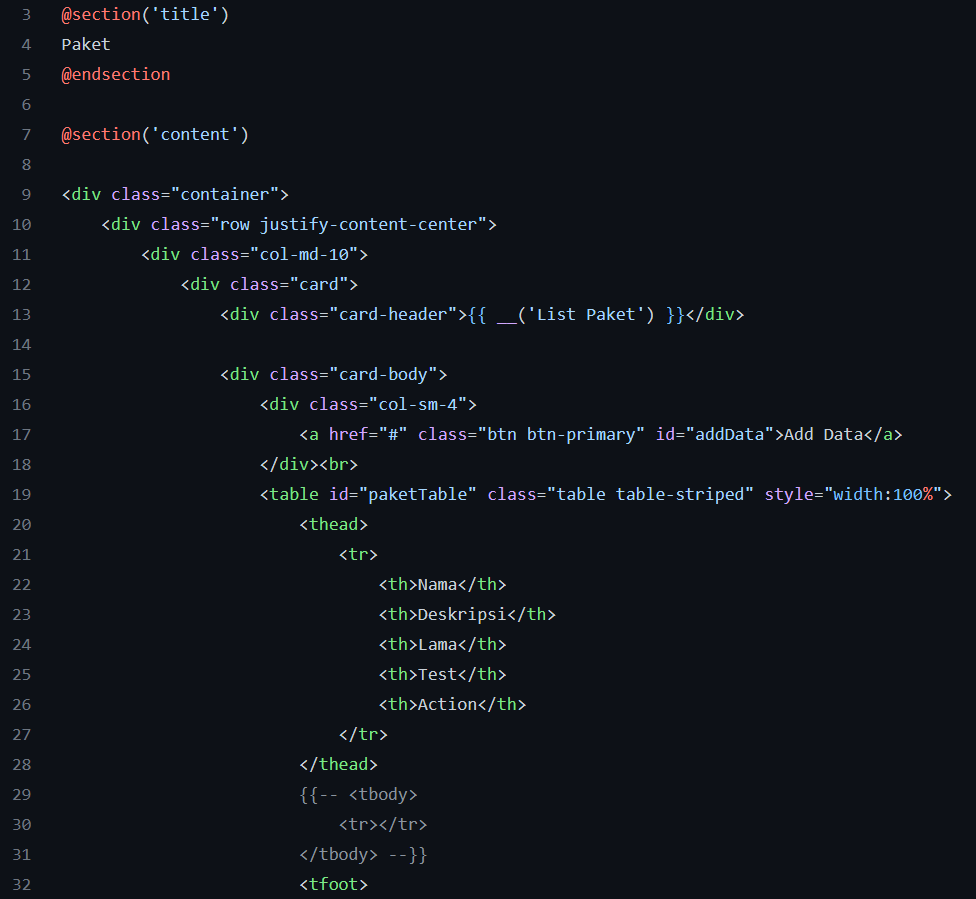
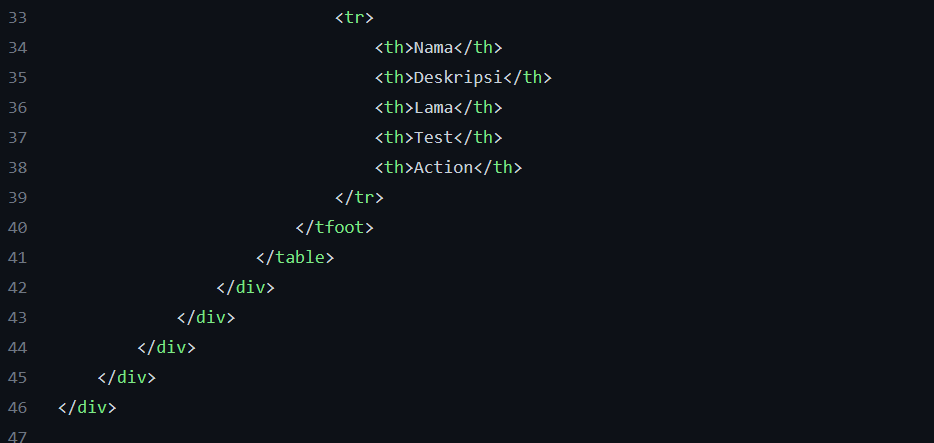
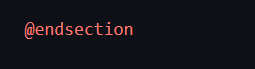
1. Setelah selesai membuat semua migration selanjutnya jalankan command berikut untuk membuat migration database “php artisan migrate”
2. Templating blade
3. Pertama download terlebih dahulu templete HTML di link berikut <https://github.com/ColorlibHQ/AdminLTE/releases/tag/v2.4.18> click yang dilingkari merah



1. Selanjutnya extrak file yang didownload sebelumnya
2. Selanjutnya copy folder assets,css dan js ke folder public laravel



1. Selanjutnya buka file index.html dengan vs code dari folder templete HTML dan copy isi coding tersebut ke file **app.blade.php** yang terdapat di folder **resource/views/layouts**
2. Selanjutnya demontrasi guru untuk lanjutannya
3. Membuat CRUD Read
4. Buka file welcome.blade.php dan ganti dengan codingan dibawah ini

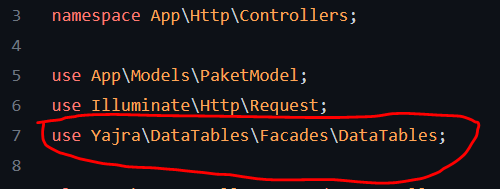
1. Selanjutnya install datatable dengan cara menjalankan command berikut “composer require yajra/laravel-datatables-oracle:"^10.0"”
2. Selanajutnya proses installasi akan berjalan dan tunggu sampai selesai
3. Selanjutnya configurasi datatable dengan cara menjalankan command berikut “php artisan vendor:publish --tag=datatables”
4. Karena proses installasi datatable sudah selesai lanjutkan dengan membuat model paket dengan cara menjalankan commad berikut “php artisan make:model PaketModel”
5. Selanjutnya buka file model yang dibuat sebelumnya yang terdapat pada folder **/app/Models** dan ikuti coding dibawah ini



1. Selanjutnya buat controller dengan cara menjalankan command berikut “php artisan make:controller PaketController -r”
2. Selanjutnya ikuti coding di bawah untuk function show()



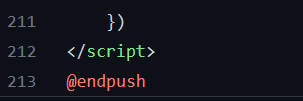
Perlu diingat jangan sampai lupa menambahkan library datatable di atas seperti gambar di bawah yang dilingkari merah



1. Selanjutnya rubah codingan route seperti gambar dibawah, yang terdapat pada folder **/routes/**



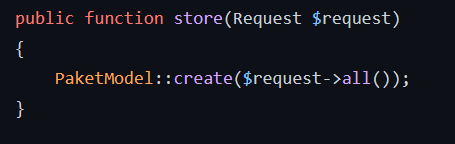
1. Selanjutnya kembali lagi ke file welcome.blade.php dan tambahkan codingan berikut dibawah section

1. Dan lihat dihome maka akan tampil sebuah table
2. Membuat CRUD Create
3. Pertama buka file welcome.blade.php dan tambahkan codingan berikut di atas @endsection

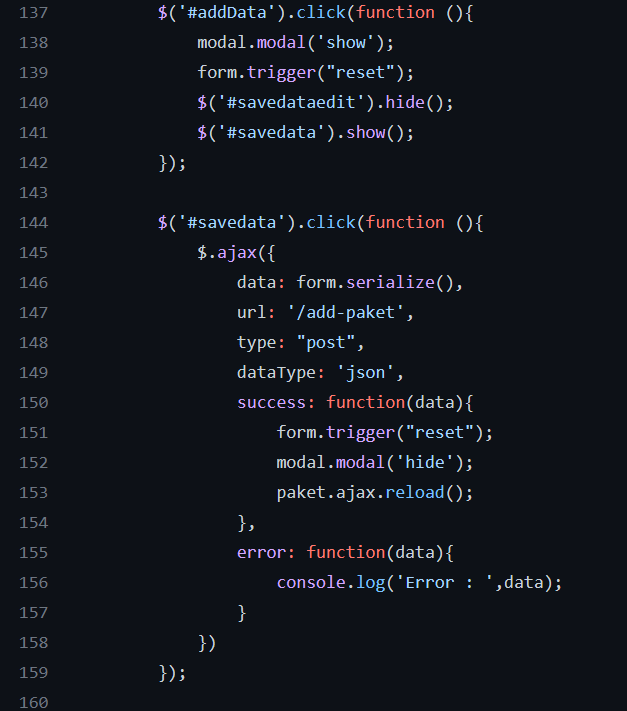
1. Selanjutnya di controller tambahkan codingan berikut



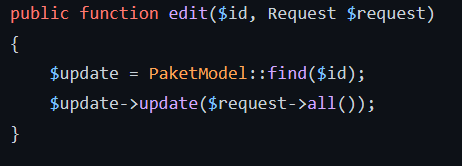
1. Selanjutnya di route tambahkan codingan berikut



1. Selanjutnya tambahkan codingan berikut di dalam script



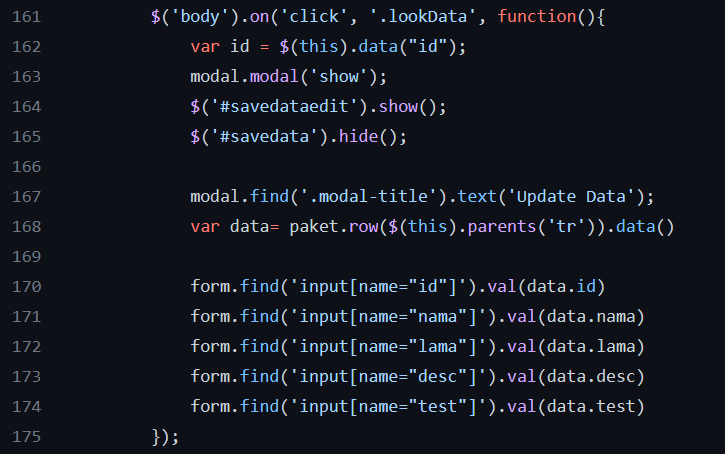
1. Selanjutnya jalankan Laravel dan click tombol add yang sekarang sudah bisa add data
2. Membuat CRUD Update
3. Untuk update karena beberapa fungsi telah ditambahkan di create dan read maka sekarang bisa lanjutkan ke controller dan tambahkan codingan berikut



1. Selanjutnya tambahkan route baru seperti codingan di bawah ini



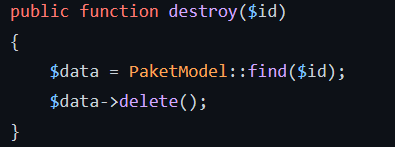
1. Selanjutnya tambahkan codingan berikut di welcome.blade.php



Dan



1. Selanjutnya jalankan Laravel dan click edit yang sekarang sudah bisa untuk mngedit data
2. Membuat CRUD Delete
3. Sama halnya dengan update beberapa sudah dibuat dan selanjutnya bisa buat coding berikut di controller



1. Selanjutnya tambahkan route berikut di route

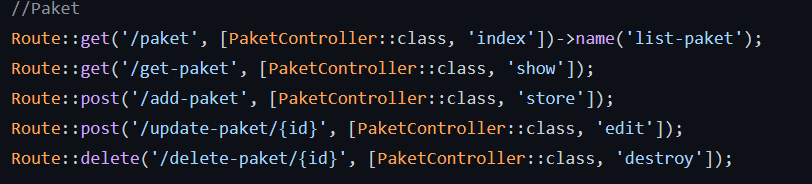


1. Selanjutnya buka file welcome.blade.php dan tambahkan codingan berikut



1. Selanjutnya jalankan program dan click tombol delete sekarang sudah bisa hapus data
2. Konversi ke API

Untuk membuat API cukup mendaftarkan controller yang telah kita buat ke api.php yang terdapat di folder route berikut codingannya



Dan untuk sedikit arahan ikuti demontrasi dari guru

1. Latihan

Bentuhlah kelompok atau tim yang terdiri dari dua orang untuk membuat simple CRUD dengan framework Lravel yang di push ke github dengan nama repository ditentukan masing-masing kelompok