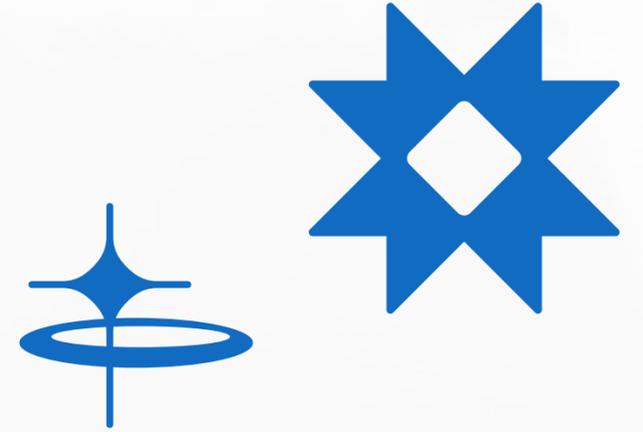
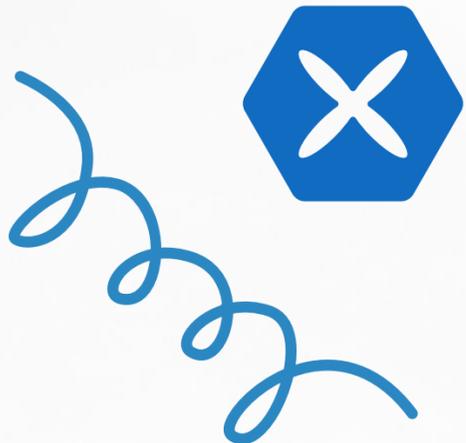


LARAVEL KNOWLEDGES



Laravel 11

THEORY 1



Yang Akan Dipelajari

- **Struktur Folder**
- **Konsep MVC**



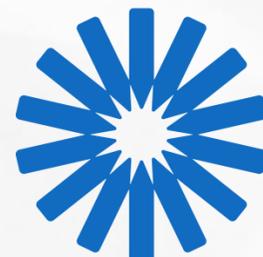
Struktur Folder

Struktur folder dari Laravel ini terbilang cukup rumit, Tapi kita bisa kelompokkan ke beberapa folder, misalnya kayak gini:

1. Folder App
2. Folder Database
3. Folder Public
4. Folder Resources
5. Folder Routes

Folder **App** ini ada beberapa folder, tapi yang biasa kita pake cuma folder **Http**, yang di dalamnya ada folder **Controllers**. Di folder ini, ada file default yang bernama **Controller.php**. Folder ini jarang dipake, tapi sering dipanggil di file controller lain.

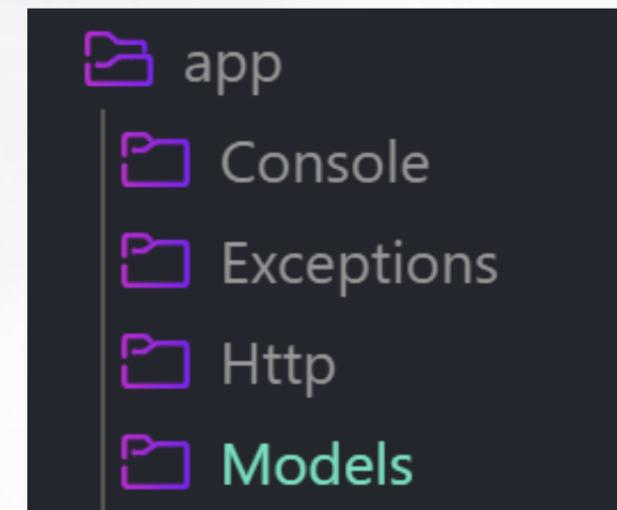
```
app
├── Http
│   └── Controllers
│       ├── Api
│       ├── AdminController.php
│       └── Controller.php
```



Struktur Folder

Di folder **App** ini juga, ada folder **Models**, yang digunakan untuk menampung file yang berhubungan dengan model. Nanti akan kita bahas di segmen konsep **MVC**.

Kira-kira begini nih struktur folder dari folder App hingga ke folder Models.



Struktur Folder

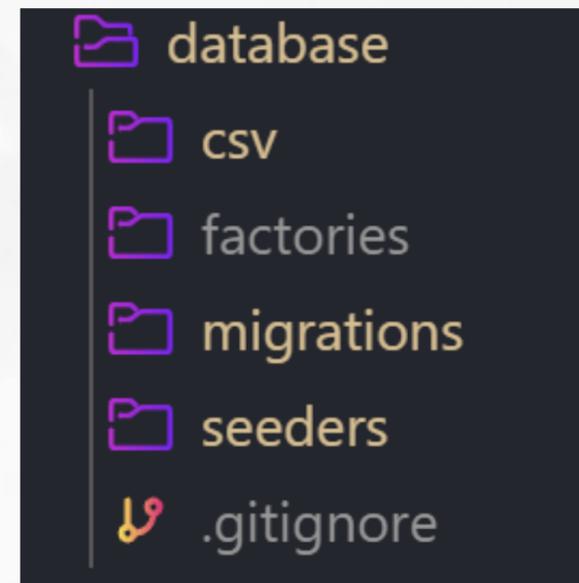
Selanjutnya, kita akan ke folder **Database**. Folder ini berisi tentang sub folder yang berkaitan dengan database, yaitu:

1. Factories
2. Migrations
3. Seeders

Nah, sebenarnya apa sih fungsi dari ketiga folder tersebut? Cus kita bahas!

Tapi sebelum itu, kita cek dulu gimana struktur awalnya.

*Abaikan folder csv, karena folder ini adalah tambahan jika diperlukan.



Struktur Folder

Folder **Factories**, adalah sebuah folder untuk menampung file yang berkaitan dengan factories / data yang bisa di-*generate* oleh Laravel dengan script dan fungsi tertentu. Contohnya seperti ini.

```
class UserFactory extends Factory
{
    protected static ?string $password;

    /**
     * Define the model's default state.
     *
     * @return array<string, mixed>
     */
    public function definition(): array
    {
        return [
            'name' => fake()->name(),
            'email' => fake()->unique()->safeEmail(),
            'email_verified_at' => now(),
            'password' => static::$password ??= Hash::make('password'),
            'remember_token' => Str::random(10),
        ];
    }
}
```



Struktur Folder

Script tersebut adalah script bawaan dari Laravel, dengan file yang bernama **UserFactory.php**

Disitu terlihat bahwa di fungsi definiton, ada beberapa key / kunci seperti **name**, **email**, **email_verified_at**, dan sebagainya, yang nantinya ketika dijalankan, akan membuat data *dummy* dengan *key / field* seperti itu, ke dalam database.

Tetapi, fungsi ini sangat jarang digunakan oleh developer. Jadi, biasanya gak terlalu dibahas banget.



Struktur Folder



Selanjutnya ada folder **Migrations**.

Folder ini adalah tempat untuk menampung file yang berkaitan dengan migrasi kolom-kolom dan tabel yang akan digunakan di dalam database, mulai dari create table sampai drop table.

Contoh file nya adalah seperti ini.

```
public function up(): void
{
    Schema::create('university', function (Blueprint $table) {
        $table->increments('university_id');
        $table->string('university_name');
        $table->string('university_acronym');
        $table->string('university_address');
        $table->string('university_acreditation');
        $table->string('university_website');
        $table->integer('user_id');
        $table->timestamps();
    });
}
```

Struktur Folder

Selanjutnya ada folder **Seeders**.

Folder ini adalah tempat untuk menampung file yang berkaitan dengan *seeder* / *seed* / penanaman data pada sebuah tabel. Sesuai dengan namanya, script di file yang ada di folder ini adalah untuk menanamkan data ke sebuah tabel, dengan cara menggunakan function `create()`;, yang disertai dengan model.



```

You, 3 months ago | 2 authors (fazallsaid and one other)
class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     */
    public function run(): void
    {
        $this->call(UniversitySeeder::class);
        $this->call(UserSeeder::class);
    }
}

```

Struktur Folder

Turun ke bawah lagi, kita akan menuju folder **Public**. Folder ini adalah untuk menampung semua aset aset kalian yang akan kalian gunakan di dalam project. Baik itu tentang css, js, gambar, dan file apapun yang bisa kalian gunakan di website. Tetapi ingat, jika file kalian bersifat rahasia / private, sebaiknya simpan di folder **Storage**, yang sejajar dengan folder **Database**.



Struktur Folder

Turun lagi, kita akan bertemu dengan folder **Resources**.

Folder ini berisikan beberapa folder seperti **css**, **js**, dan **views**. Nah, biasanya developer akan lebih fokus ke bagian **views**, karena di sinilah tempat file **Blade** itu berkumpul.

File Blade itu apa sih?

File Blade adalah file dengan ekstensi khusus dari laravel, guna untuk menampilkan tampilan yang akan atau sudah kita rancang di browser. Gampangnya, file blade ini isinya adalah *script* **html**, **css** dan **js**



Konsep MVC

Mungkin ada yang tanya, konsep MVC itu kayak gimana sih?

Sebelum itu, kita harus tahu dulu, MVC itu apa.

MVC adalah kependekan / singkatan dari Model View Controller.

MVC ini sangat diperlukan di dalam sebuah framework, untuk memetakan antara tampilan (View), logika program (Controller) dan Database / yang berhubungan dengan database (Model).



Konsep MVC



MVC ini digunakan dengan tujuan untuk memisahkan antara logika bisnis, tampilan dan database, dan juga buat mudahin developer dalam mengatur ketiga aspek utama tersebut.

Konsep MVC secara umum adalah kita ibaratkan aplikasi adalah sebagai sebuah restoran.

Customer resto tersebut (dalam hal ini adalah pengguna), pengen pesan makanan (data) ke kasir / pelayan, dengan segala request yang customer inginkan. Nah, si kasir / pelayan yang menerima pesanan tersebut, jika di dalam sistem adalah Controller, yang juga bertugas untuk menyampaikan pesanan ke koki. Koki di dalam konsep MVC adalah Model.

Konsep MVC



Koki tersebut akan memasak bahan-bahan (data mentah) yang sesuai dengan pesanan dan request dari customer tadi. Setelah makanan (data matang) sudah jadi, maka akan diantarkan kembali oleh pelayan (Controller) untuk disajikan kepada customer.

Lalu fungsi view di dalam contoh “restoran” tadi tuh apa?

Nah, view disini mengacu ke meja, kursi, dan aksesoris lainnya yang dibuat khusus untuk customer. Pasti kalian pernah menilai bangunan restoran dan ruangnya ketika pertama kali masuk, kan? Bentuk ruangan, bangunan, aksesoris, desain meja dan kursinya itulah, yang menjadi tanggung jawab View, untuk menampilkannya kepada customer (pengguna).

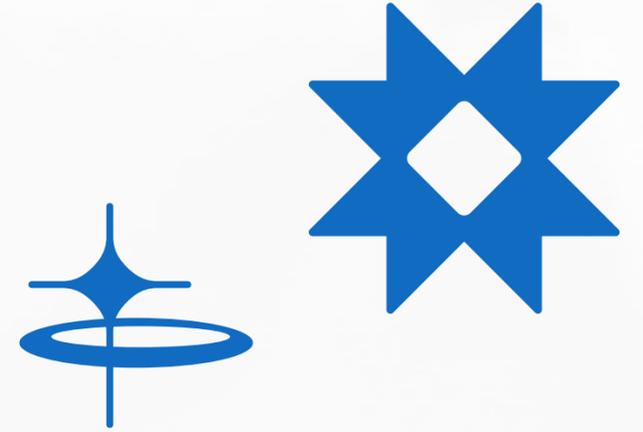
Tugas

Tugas pertama kali ini cuma simple aja, yaitu instal laravel 11 dengan benar, dan membuat tampilan html css sederhana, beserta dengan Model dan Controller nya.

Tugasnya disinkronisasi dengan Git, dan link repositori nya diupload di link ini:

<https://s.id/laraKnowTheory1>





Thank You!

