

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Beberapa acuan yang digunakan dalam pengembangan sistem dan aplikasi ini di tampilkan pada tabel :

Tabel 2. 1 Tinjauan Pustaka

Parameter/ Penulisan	Judul	Framework/ Bahasa pemrograman	Interface
Windi Destria Damayanti (2016)	Aplikasi akademik pada SMK N 2 Magelang berbasis web	Php, Mysql.	SMK N 2 Magelang
Dony Prabowo (2015)	Website E- Comerce menggunakan MVC dengan framework CodeIgneter	CodeIgneter, php.	Toko Miniatur
Mariska Abrani (2011)	Pengembangan sistem informasi sekolah berbasis web	Php, Mysql.	MI an-Nizhimiyah depok
Vitalis Agung (2018)	Aplikasi penerimaan siswa baru	Twiter Bootstrap.	SMK SANTU PETRUS
Junior D Rumpaidus	Aplikasi pengelolaan data siswa berbasis web menggunakna framework laravel	Laravel, Bootstrap, php, Mysql.	SMA N 2 Biak Kota

Penelitian yang dilakukan oleh Winda Destria Damayanti (2016) dengan judul aplikasi akademik pada SMK N 2 Magelang berbasis website, penelitian ini bertujuan untuk dapat membantu pihak sekolah dalam pengelolaan data akademis dan memberikan informasi kepada siswa agar dapat melihat nilai dan presensi.

Penelitian yang dilakukan oleh Dony prabowo (2015) dengan judul Website E-commerce menggunakan (*Model View Controller*)MVC. Dengan Framework CodeIgneter (studi kasus : toko miniatur) penelitian ini memberikan informasi bahwa dengan adanya website *e-commerce* pada toko miniatur akan membantu pelanggan dalam melihat stok barang, memesan ataupun membeli hingga proses transaksi pada toko miniatur karena *E-commerce* menerapkan *Model View Controller* (MVC) yang merupakan metode untuk membuat sebuah aplikasi dengan memisahkan bagian data (Model), tampilan (View).

Penelitian yang dilakukan oleh Mariska Abrani (2011) dengan judul Pengembangan Sistem Informasi sekolah berbasis web (studi kasus : MI An-Nizhomiyah Depok) penelitian ini memberikan informasi untuk membantu kepala sekolah, guru, staf keuangan, staf TU, dalam menyajikan data sekolah sehingga dapat memenuhi kebutuhan dalam pencarian dan penyajian informasi yang berhubungan dengan sekolah itu sendiri dengan proses yang cepat dan tepat.

Pada penelitian yang dilakukan oleh Vitalis Agung (2018) dengan judul Aplikasi penerimaan siswa baru (studi kasus : SMK Santu Petrus) Penelitian ini memberikan informasi petunjuk teknis pelaksanaan persyaratan dan pendaftaran.

Sedangkan pada penelitian ini, Aplikasi pengelolaan data siswa di bangun agar dapat membantu staf tata usaha, dan guru di SMA N 2 untuk mengelola data nilai siswa. Aplikasi ini di bangun menggunakan *Framework Laravel* yang menggunakan pola *Model View Controller (MVC)*.

2.2 Dasar Teori

2.2.1 Framework

Framework adalah struktur konseptual dasar yang berisi kumpulan fungsi untuk tujuan tertentu yang sudah siap digunakan, sehingga pembuatan aplikasi dapat dilakukan dengan lebih cepat karena kode programnya tidak dibuat dari awal.

Beberapa alasan dari digunakan *framework* dalam membuat aplikasi adalah sebagai berikut.

1. Apliasi akan memiliki standar pemrogramanyang universal.
2. Menghindari *repetitive work*.
3. Memudahkan dalam *team work*.
4. Memudahkan dalam *maintenance* dan pengembangan aplikasi dimasa mendatang.
5. Hemat waktu dan biaya.

2.2.2 Laravel

Laravel adalah *framework* bahasa pemrograman *Hypertext Preprocessor (PHP)* yang di tujukan untuk pengembangan aplikasi berbasis web dengan menerapkan konsep *Model view Controller (MVC)*. *Framework* ini dibuat oleh

Taylor Otwell dan pertama kali dirilis pada tanggal 9 juli 2011. *Laravel* berlisensi *open source* yang artinya bebas digunakan tanpa harus melakukan pembayaran. Alamat *website* resmi dari *framework* laravel adalah <https://laravel.com>. Fitur-fitur modern *Laravel* yang sangat membantu *developer* dalam membuat aplikasi adalah *Bundles*, *Eloquent ORM (Object-Relational Mapping)*, *Query Builder*, *Application Logic*, *Reverse Routing*, *Resource Controller*, *Class Auto Loading*, *View Composers*, *Blade*, *IoC*, *Containers*, *Migration*, *Database Seeding*, *Unit Testing*, *Automatic Pagination*, *Form request*, *Middleware*.

Framework laravel juga memiliki beberapa keunggulan sebagai berikut.

1. Menggunakan *Command Line Interface (CLI) Artisan*.
2. Menggunakan *Package manager PHP Composer*.
3. Penulisan kode program lebih singkat, mudah dimengerti, dan ekspresif.

Kemudian untuk cara instalasi *framework Laravel* dapat dilakukan dengan 3 cara yaitu.

1. Melalui *Installer Laravel*.
2. Menggunakan *Composer* dengan mengetikan perintah *create-project*.
3. Download *source code Laravel* secara lengkap melalui *GitHub* dengan alamat <https://github.com/laravel/laravel/>.

Pada tanggal 4 september 2018 *framework Laravel* versi 5.7 resmi di rilis. Fitur *framework Laravel* yang ditekankan pada penelitian ini adalah *Blade*, *Migration*, *Eloquent ORM*, *Resource Controller*, dan *Middleware*. Berikut adalah penjelasan mengenai lima fitur tersebut.

1. Blade

Blade adalah *template engine*. Pada dasarnya *Blade* adalah *view* namun dengan menggunakan *Blade* akan mempermudah untuk mengatur tampilan *website* dan menampilkan data. Cara untuk membuat file *view Blade* adalah dengan menambahkan ekstensi `.blade.php` pada file *view*. Dan cara untuk memanggil file *Blade* sama dengan cara memanggil file *view* biasa.

2. Migration

Migration adalah fitur yang menyediakan cara baru untuk memnuat *database*. Dengan menggunakan *Migration* cara membuat *database* melalui *Command Line Interface (CLI) databse* atau dengan menggunakan aplikasi *database manager* digantikan dengan menggunakan *class*. Tahapan menggunakan *Migration* adalah membuat *class* kemudian melakukan perintah *migrate* melalui *Command Line Interface (CLI) artisan*.

Keuntungan menggunakan *Migration* adalah *class* yang dibuat bisa dipakai untuk membuat *database* pada berbagai macam *Relation Database Management System (RDBMS)* yang didukung oleh *Laravel*. Sebagai contoh misalnya aplikasi yang digunakan selama ini menggunakan *databse MYSQL*, kemudian karena alasan pengembangan aplikasi maka akan dilakukan penggantian *database* ke *PostgreSQL*. Dalam proses penggantian tersebut tidak perlumembuat *class* lagi, tinggal melakukan perintah *migrate* melalui *Command Line Inteface(CLI) artisan*.

Keuntungan lain dari menggunakan *Migration* adalah semua perubahan yang dilakukan pada *database* akan disimpan pada suatu tabel. Sehingga bisa

dilakukan pembatalan dengan cara memasukan perintah *rollback* pada *database* jika melakukan perubahan yang tidak benar.

3. Eloquent ORM

Eloquent ORM adalah implementasi dari *ActiveRecord* yang digunakan untuk mengatur relasi antar tabel di *database*. Pada *Eloquent ORM* tabel direpresenatasikan dalam bentuk kelas dan data yang tersimpan didalam tabel direpresentasikan dalam bentuk objek. Relasi yang dapat diatur menggunakan *Eloquent ORM* adalah sebagai berikut.

- a. *One-to-One* yaitu relasi satu ke satu. Pada relasi ini digunakan *method hasOne* dan *belongsTo*.
- b. *One-to-Many* yaitu relasi banyak ke satu. Pada relasi ini digunakan *method hasMany* dan *belongsTo*.
- c. *Many-to-One* yaitu relasi banyak ke satu. Pada relasi ini digunakan *method belongsTo* dan *hasMany*.
- d. *Many-to-Many* yaitu relasi banyak ke banyak. Pada relasi ini digunakan *method belongsToMany*.

4. Resource Controller

Resource Controller adalah fitur yang digunakan untuk mempercepat pembuatan *controller*. Sebagai contoh misalnya ada *controller* yang menangani semua *HTTP request* terhadap data dosen, untuk membuat *controller* tersebut hanya perlu mengetikan perintah berikut.

```
$ php artisan make:controller SiswaController
```

Perintah di atas akan menghasilkan *controller* SiswaController.php yang disimpan pada folder app/Http/Controllers.

Setelah membuat *controller* SiswaController.php hal yang harus dilakukan selanjutnya adalah memuat satu baris kode program pada *route*.

```
Route::resource('siswa', 'SiswaController');
```

Satu baris kode program *route* diatas akan menangani semua *route* diatas akan menangani semua *route* untuk melihat, menambah, mengedit, dan menghapus data siswa.

Jadi dapat disimpulkan dengan menggunakan fitur *Resource Controller* dapat mempercepat pembuatan *controller* serta dapat menyederhanakan *route* untuk *controller*.

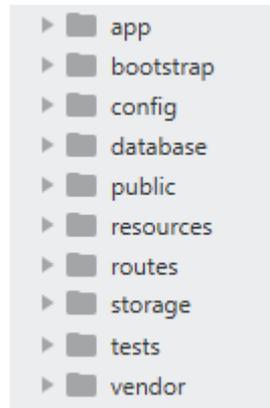
5. Middleware

Middleware adalah fitur yang menyediakan mekanisme untuk memfilter *HTTP request* yang masuk ke aplikasi. *Laravel* memiliki beberapa *Middleware* yaitu *Authenticate*, *EncryptCookies*, *RedirectIfAuthenticated*, dan *VerifyCsrfToken*.

Sebagai pembahasan akan dibahas *Middleware Authenticate*. *Middleware* tersebut akan memeriksa apakah *user* sudah *login* maka *request* akan dilanjutkan ke halaman yang dikehendaki oleh *user*. Tetapi jika *user* belum *login* maka *Middleware Authenticate* akan mengarahkan *user* ke halaman *login*.

Jika *Middleware* yang sudah ada pada *Laravel* kurang sesuai dengan kebutuhan ataupun tidak sesuai dengan kebutuhan maka dapat dibuat sendiri *Middleware* yang sesuai dengan kebutuhan.

Struktur folder dari *framework Laravel 5.7* yang masih default dapat dilihat pada gambar 2.1.



Gambar 2. 1 Struktur Folder Laravel 5.7

- a. Folder *app* adalah folder yang berisi kode program inti dari aplikasi yang akan dibuat. *Model* dan *Controller* tersimpan pada folder ini.
- b. Folder *bootstrap* adalah folder yang berisi kode konfigurasi *autoloading* dan terdapat juga folder *cache* yang menyimpan file-file yang dihasilkan secara otomatis oleh *Laravel* untuk mengoptimasi kinerja dari sistem yang dihasilkan.
- c. Folder *config* adalah folder yang berisi semua file konfigurasi aplikasi.
- d. Folder *database* adalah folder yang berisi file *database migration* dan *seeds*.
- e. Folder *public* adalah folder yang berisi file *index.php*. File tersebut digunakan sebagai *entry point* untuk menangani semua *request* yang masuk ke aplikasi. Pada folder ini juga dapat disimpan beberapa aset dari aplikasi seperti gambar, *JavaScript*, dan *CSS*.
- f. Folder *resources* adalah folder yang berisi file *view* dari aplikasi yang dibuat. Selain itu terdapat juga file *language* yang digunakan aplikasi.

- g. Folder *route* adalah folder yang berisi file yang digunakan untuk mendefinisikan semua *route* ke aplikasi. Secara *default* ada tiga file *route* yang disediakan *Laravel* yaitu *api.php*, *console.php*, dan *web.php*.
- h. Folder *storage* adalah folder yang berisi *template Blade* yang dikompilasi, file *session*, file *cache*, dan file lainnya yang dihasilkan secara otomatis oleh *Laravel*.
- i. Folder *test* adalah folder yang berisi semua file *test* yang dibuat untuk aplikasi.
- j. Folder *vendor* adalah folder yang menyimpan semua *library* yang digunakan.

2.2.3 Bootstrap

Bootstrap adalah *framework* bahasa pemrograman *Cascade Style Sheet (CSS)*, *Hyper Text Markup Language (HTML)*, dan *Javascript* yang ditujukan untuk membuat tampilan aplikasi berbasis web menjadi responsif. Maksud responsif adalah tampilan aplikasi web akan menyesuaikan dengan ukuran layar dari perangkat yang mengaksesnya. *Framework* ini dibuat oleh Mark Otto dan Jacob Thornton. *Bootstrap* pertama kali dirilis pada tanggal 19 agustus 2011 dan berlisensi *open source* yang artinya bebas digunakan tanpa harus melakukan pembayaran. Alamat *website* resmi dari *framework bootstrap* adalah <http://getbootstrap.com>. Untuk mengunduh *framework bootstrap* dapat melalui *Gituhup* dengan alamat <https://github.com/twbs/bootstrap/>.

2.2.4 MySQL

MySQL adalah *database server* yang digunakan untuk menyimpan dan memanajemen data, dalam bahasa inggris disebut *Database Management System (DBMS)*. *MySQL* merupakan implementasi dari sistem manajemen basis data

relasional, dalam bahasa inggris disebut *Relation Database Management System (RDBMS)*. Secara umum *Structured Query Language (SQL)* pada *MySQL* dibagi menjadi dua yaitu.

1. *Data Definition Language (DDL)* yang digunakan untuk membuat objek pada basis data seperti tabel, *indeks*, *sequence*, dan *view*. Yang termasuk dalam perintah *DDL* adalah *CREATE*, *ALTER*, dan *DROP*.
2. *Data Manipulation language (DLM)* yang digunakan untuk memanipulasi objek pada basis data. Yang termasuk dalam perintah *DLM* adalah *SELECT*, *INSERT*, *UPDATE*, dan *DELETE*.

Structured Query Language (SQL) yang dipakai pada aplikasi ini adalah *CREATE*, *DROP*, *SELECT*, *INSERT*, *UPDATE*, dan *DELETE*.