

BAB II

KAJIAN LITERATUR

Pada bab ini berisikan tentang pembuatan teori pengembangan sistem dengan menggunakan Laravel, teori dengan metode Scrum, dan teori atau definisi yang relevan serta penelitian terkait.

2.1 Tinjauan Pustaka

2.1.1 Laravel

Laravel merupakan *framework* PHP yang menekan pada kesederhanaan dan fleksibilitas pada desainnya. *Laravel* merupakan *framework* PHP yang selalu *up-to-date* karena bersifat *open-source* dan dikembangkan secara bersama. *Laravel* memberikan kemudahan untuk berinteraksi dengan *database* yang di sebut *migration*. Dengan adanya *migration*, programmer sangat mudah untuk melakukan modifikasi sebuah *database* secara independen karena implementasi skema *database* dipresentasikan dalam sebuah *class*. ada beberapa basis data yang mendukung *migration* di *laravel* seperti (*MySQL*, *PostgreSQL*, *MSSQL*, dan *SQLITE*) dan untuk menggunakan *framework laravel* sebelumnya programmer harus menguasai metode dalam membuat program dengan istilah OOP (*Object Oriented Programming*) [5]. Berapa fitur yang terdapat di *Laravel* :

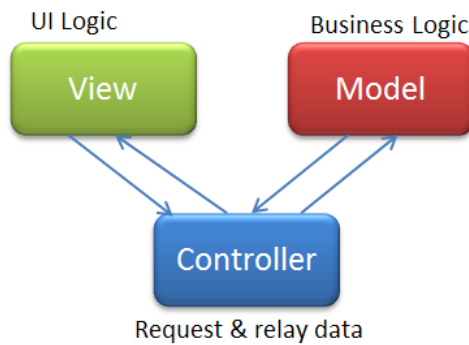
- *Bundles*, yaitu sebuah fitur dengan sistem pengemasan modular dan tersedia beragam di aplikasi,
- *Eloquent ORM*, merupakan penerapan PHP lanjutan menyediakan metode internal dari pola “*active record*” yang mengatasi masalah pada hubungan objek *database*.
- *Application Logic*, merupakan bagian dari aplikasi, menggunakan *controller* atau bagian *Route*.
- *Reverse Routing*, mendefinisikan relasi atau hubungan antara *Link* dan *Route*.

- *Restful Controllers*, memisahkan logika dalam melayani *HTTP GET* dan *POST*.
- *Class Auto Loading*, menyediakan loading otomatis untuk class PHP.
- *View Composer*, adalah kode *unit logical* yang dapat dieksekusi ketika *view* sedang *loading*
- *IoC Container*, memungkinkan objek baru dihasilkan dengan pembalikan *controller*.
- *Migration*, menyediakan sistem untuk skema database.
- *Unit Testing*, banyak tes untuk mendeteksi dan mencegah regresi.
- *Automatic pagination*, menyederhanakan tugas dari penerapan halaman.

Framework adalah kerangka kerja. *Framework* digunakan oleh *developer* untuk memudahkan pembangunan aplikasi *web* yang dapat berupa sekumpulan *library* yang berisi fungsi, *tools*, ataupun *class-class*, dan digunakan sebagai kerangka dalam pembangunan aplikasi *web*. Umumnya diidalam *framework* tersebut telah menyediakan solusi untuk akses *database*, *authetincation*, *templating*, *controls*, dan fungsi-fungsi lainnya. Penggunaan *framework* diharapkan membuat pengembangan aplikasi menjadi rapih dan bersih.

2.1.2 Model View Controller

Model View Controller merupakan suatu konsep yang cukup populer dalam pengembangan aplikasi *web*, berawal pada bahasa pemrograman *Small Talk*, MVC memisahkan pengembangan aplikasi berdasarkan komponen utaman ya membangun sebuah aplikasi seperti manipulasi data, *user interface*, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu MVC *pattern* dalam suatu aplikasi, yaitu :



Gambar 2.1 Model View Controller

- *Model* : Berfungsi untuk memanipulasi data beserta aturan bisnisnya seperti proses validasi dan relasi, serta mengimplementasi logika untuk data aplikasi. Dengan adanya model aplikasi yang di hasilkan akan lebih terstruktur dan data yang tersimpan akan lebih aman karena harus melalui validasi terlebih dahulu.
- *Views* : Merupakan komponen yang berisikan keseluruhan detail dari tampilan antar muka untuk pengguna aplikasi seperti User. Dengan *views* semua proses internal aplikasi akan ditampilkan kepada user dan menuntun alur interaksi user terhadap aplikasi.
- *Controller* : Terjadi proses peng-inputan dari pengguna dan menginstruksikan *model* dan *view* untuk melakukan aksi berdasarkan masukan tersebut. Sehingga bertanggung jawab atas penampungan event yang dilakukan oleh user.

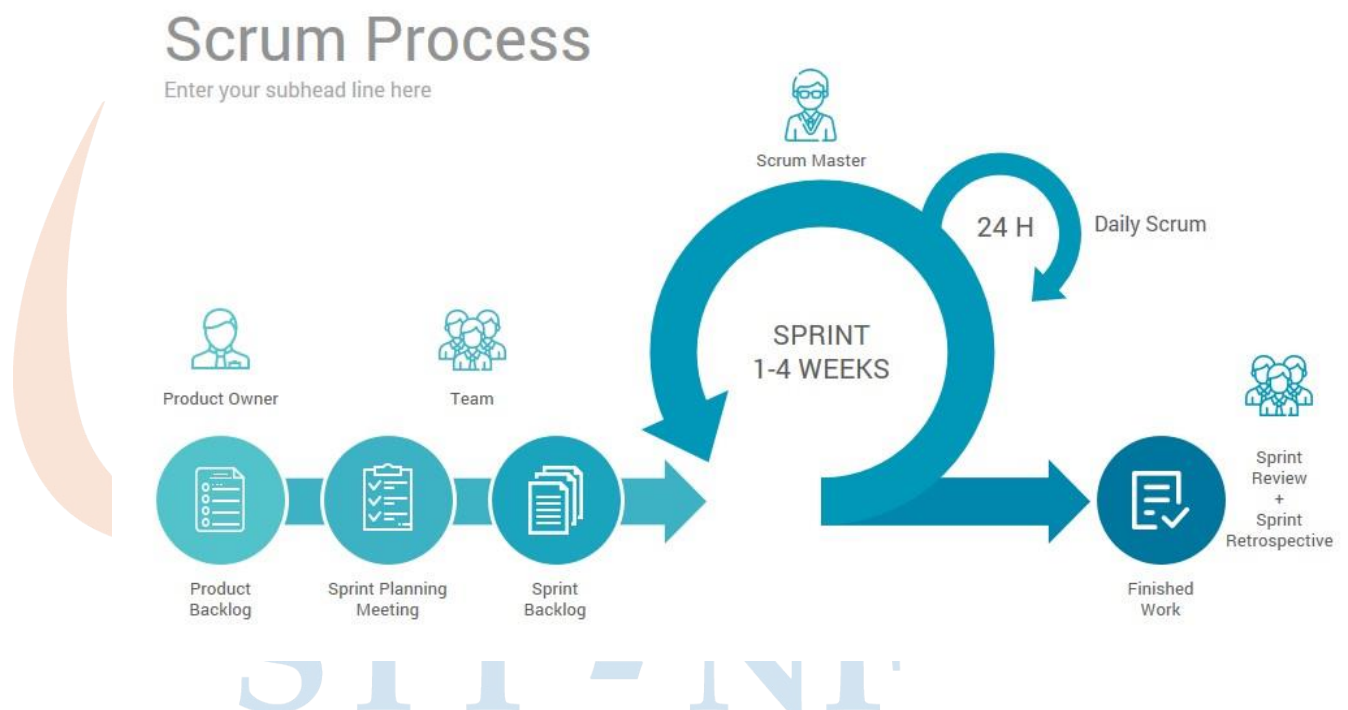
2.1.3 Elasticsearch

Elasticsearch merupakan sebuah *search engine* yang berorientasi pada *full-text* atau dokumen dan dapat diakses melewati *Restful API* yang bisa mengembalikan dokumen dengan ter-index dalam bentuk *JSON* saat diquery [6]. Poin terpenting dari *elasticsearch* yaitu dapat melakukan *full text search*, meng-handle *synonyms* dan menilai dokumen berdasarkan relevansinya, meng-generate

analytic dan *aggregation* dari data yang sama dan dapat melakukan proses pencarian secara *real-time* tanpa *big batch processing jobs*. *Elasticsearch* itu *Search Server* yang didasarkan pada *Apache Lucene Library* yang dikembangkan menggunakan Bahasa Pemrograman *JAVA* dan bersifat *Open Source* dibawah *the terms of the apache 2 license*.

Elasticsearch adalah mesin pencarian teks dan analitik yang bersifat *open source* yang sangat *scalable* serta memungkinkan penggunanya untuk menyimpan, mencari dan menganalisis data yang besar dengan cepat dan mendekati *real time*. *Elasticsearch* umumnya digunakan sebagai mesin/teknologi yang mampu mengoptimasi sebuah aplikasi yang memiliki fitur pencarian [7].

2.1.4 SCRUM



Gambar 2.2 Scrum

Scrum merupakan sebuah kerangka kerja untuk mengembangkan sebuah produk yang kompleks, dimana visi dari *Scrum* adalah produk yang bernilai tinggi secara kreativitas maupun produktivitas. Setiap komponen di dalam kerangka

kerja memiliki maksud tertentu dan peran penting demi keberhasilan penggunaan Scrum itu sendiri [8]. Scrum memiliki beberapa bagian seperti berikut :

- *Product Backlog*

Penulis menyusun dan mengumpulkan semua permintaan dan kebutuhan sistem, misalnya fitur-fitur yang dibutuhkan dan-atau kebutuhan non-fungsional sistem. Setelah tujuannya sudah ditetapkan, semua permintaan dan kebutuhan tersebut dibagi-bagi menjadi bagian kecil

- *Backlog Refinement*

Backlog harus di-maintenance dengan baik dan tepat oleh tim *scrum* untuk dilakukan perencanaan, sehingga Sprint dapat berjalan dengan lancar. Hal yang harus dilakukan dalam *me-maintenance backlog* antara lain adalah melakukan proses estimasi dan *breakdown* kebutuhan, hal tersebut dilakukan agar kondisi *Sprint* terpenuhi

- *Sprint*

Sebuah batasan waktu selama satu bulan atau kurang. Biasanya memiliki durasi yang konsisten sepanjang proses pengembangan produk. Sprint yang baru, langsung dimulai setelah *sprint* yang sebelumnya berakhir.

- *Sprint Backlog*

Sekumpulan Produk *Backlog* yang telah dipilih untuk dikerjakan di *sprint*. terdapat didalamnya juga rencana untuk mengembangkan potongan tambahan produk dan merealisasikan *Sprint Goal*.

- *Daily Scrum*

Kegiatan dengan batasan waktu maksimum selama 15 menit agar tim pengembangan dapat mensinkronisasikan pekerjaan mereka dan membuat perencanaan untuk 24 jam kedepan.

- *Sprint Review*

Diadakan di akhir Sprint untuk meninjau inkremen dan merubah Proyek Backlog bila diperlukan. Pada saat *Sprint Review*, tim *scrum* dan *stakeholder* berkolaborasi untuk membahas apa yang telah dikerjakan dalam Sprint yang baru usai.

2.1.5 Black Box Testing

Black Box Testing pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program untuk mengetahui apakah fungsi, *input* dan *output* dari aplikasi yang dibuat sesuai dengan spesifikasi yang diinginkan. Tester berfokus pada pengujian fungsi program terhadap spesifikasi tersebut. dengan pengujian *black-box*, tester memandang program sebagai kotak hitam dan benar-benar tidak peduli dengan struktur internal dari program atau sistem [9]. *Black Box Testing* cenderung untuk menemukan hal-hal berikut :

- a. Fungsi yang tidak benar atau tidak ada
- b. Kesalahan antar muka
- c. Kesalahan pada struktur data dan akses basis data
- d. Kesalahan Performa

2.1.6 User Acceptance Testing

User Acceptance Testing merupakan pengujian yang ditujukan di luar sistem yaitu user. Pengujian ditujukan untuk menghasilkan dokumen yang dijadikan bukti bahwa software yang telah dikembangkan dapat diterima oleh pengguna, apabila hasil pengujian testing sudah bisa dianggap memenuhi kebutuhan dari pengguna [10].

2.1.7 Skala Likert

Skala Likert adalah skala yang dapat dipergunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang suatu gejala atau fenomena sosial yang dimana jawaban setiap item instrumen mempunyai gradasi dari sangat positif sampai sangat negatif. *Skala Likert* adalah suatu skala psikometrik yang umum digunakan dalam kuisioner, dan

merupakan skala yang paling digunakan dalam riset berupa survei. Responden menentukan tingkat persetujuan mereka terhadap suatu pernyataan dengan memilih salah satu dari pilihan yang tersedia. Biasanya disediakan lima pilihan skala dengan format seperti :

- Sangat Setuju
- Setuju
- Tidak Setuju
- Sangat Tidak Setuju

2.1.8 MySQL

MySQL adalah *DBMS* yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi *MySQL* adalah *database server* yang gratis dengan lisensi *GNU General Public License (GPL)* sehingga dapat Anda pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada. *MySQL* masuk ke dalam jenis *RDBMS (Relational Database Management Sistem)*. Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada *MySQL*. Contohnya di dalam *MySQL* sebuah database terdapat satu atau beberapa tabel.

2.1.9 Postman

Postman adalah sebuah aplikasi berbasis *web* yang berfungsi sebagai *REST Client* untuk uji coba *REST API* yang tersedia dalam bentuk ekstensi pada *Google Chrome*. *Postman* mempunyai tampilan antarmuka untuk memudahkan *developer* dalam penggunaannya, serta *Postman* mempunyai fitur seperti *design, build, test,* dan *documentation API* [12].

2.1.10 UML

UML adalah sebuah bahasa pemodelan sebagai alat bantu untuk mevisualisasi, memspesifikasi, membangun dan pendokumentasian dari sebuah sistem pengembangan *Software* berbasis *Object Oriented*. Di sisi lain *UML* sebagai bahasa pemodelan juga memberikan standar penulisan sebuah cetak biru

(*Blueprint*) sebagai konsep bisnis proses, penulisan kelas-kelas dalam bahasa pemrograman yang spesifik, skema *database* dan komponen-komponen yang diperlukan dalam sistem software. UML menyediakan notasi-notasi yang sudah baku untuk pengembangan sistem yang memudahkan seorang analisis sistem untuk menggambarkan suatu permasalahan dalam bentuk model abstrak dengan detail kepada setiap orang yang terlibat di dalam proses pengembangan sistem tersebut, Beberapa model *tools* diagram yang digunakan sebagai dasar menggunakan UML antara lain:

a. *Diagram Use Case*

Diagram yang menggambarkan aktifitas aktordan *use case* yang dilakukan oleh sistem dari sudut pandang pengamatan seseorang. Model *use case* menggambarkan sistem sebagai sebuah kotak hitam dan interaksi antar aktor dan sistem dalam suatu bentuk teks, yang terdiri dari *input user* dan respon sistem. Dalam *use case* seseorang hanya dapat melihat aksi dari aktor dan respon dari sistem bagaimana sistem itu bekerja.

b. *Diagram Class*

Diagram yang menggambarkan hubungan antara *class-class* yang didalamnya terdapat atribut dan fungsi dari suatu objek. Class diagram mempunyai 3 relasi dalam penggunaannya yaitu : *Assosiation* (hubungan interaksi antar class), *Generalization* (hubungan antar class dari khusus ke umum), *Constraint* (hubungan antar class yang di batasi oleh sistem).

c. *Diagram Sequence*

Diagram yang menggambarkan *interaction* bagaimana suatu operasi dilakukan, apa yang dikirim dan kapan pelaksanaannya berdasarkan waktu. *Sequence* diagram membantu untuk mengidentifikasi setiap data masuk dan keluar dari sebuah sistem.

d. *Diagram Collaboration*

Diagram yang menggambarkan interaction sama halnya dengan diagram *sequence* akan tetapi lebih memusatkan atau memfokuskan pada kegiatan objek dari waktu pesan itu dikirimkan. Diagram *Sequence* mengacu terhadap konteks objek sedangkan diagram *Collaboration* mengacu terhadap waktu.

e. *Diagram State*

Diagram yang menggambarkan keadaan dari satu *state* ke *state* lainnya dalam masa transisi dari suatu objek pada sistem yang mengalami perubahan akibat dari respon yang di terima. Pada umumnya *stateclass* tidak dapat di gambarkan untuk semua *class* dan satu *class* dapat memiliki lebih dari satu *stateclass*

f. *Diagram Activity*

Diagram yang menggambarkan berbagai aliran dari aktivitas dalam sistem yang di rancang. Diagram *Activity* digunakan untuk mendeskripsikan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih.

g. *Diagram Deployment*

Diagram yang menggambarkan konfigurasi fisik perangkat lunak dan perangkat keras pada sistem serta hubungan antara *node-node*. Tujuan atau fungsi dari deployment diagram yaitu untuk menggambarkan secara umum proses yang terjadi pada suatu sistem/*software*.

STT - NF

2.2 Penelitian Terkait

Table 1. Penelitian Terkait

No	Nama dan Tahun	Studi Kasus	Tools	Metodologi	Kesimpulan	Platform
1	Vera Nopitasari, Intan oktaviani, Indah Nofikasari. STMIK Duta Bangsa Surakarta	Aplikasi Resep Masakan Tradisional	Ionic Serve, Dreamweaver	Multimedia Development Live Cycle (MDLC)	Aplikasi resep masakan tradisional berbasis mobile yang memfokuskan resep masakan pada daerah pulau jawa.	Android
2	Amira Salsabella. Teknik Elektro Fakultas Teknik Universitas Tanjungpura	Sistem Pendukung Keputusan penentuan Resep Masakan Berdasarkan Ketersediaan Bahan Makanan menggunakan Metode Simple Additive Weighting berbasis Web	SAW, Rank Order Centroid (ROC), MySQL	Simple Additive Weighting (SAW)	diperlukan sistem pendukung keputusan berbasis web yang dapat menentukan resep sesuai dengan kebutuhan pengguna. Pengguna akan memasukkan bahan makanan, bumbu, waktu memasak, jenis masakan, metode memasak dan tingkat kesulitan (pemula, menengah, ahli). Dari entri data pengguna, sistem akan menentukan resep yang	Website

					sesuai dengan keinginan pengguna.	
3	Pradiptya Setyahadi. universitas Islam Negeri Sunan Kalijaga.	Rancang Bangun Aplikasi Masakan Berbasis Mobile Web dengan Metode Case-Based Reasoning	Adobe Photoshop, notepad++, PHP, MySQL	Case-Based Reasoning	Aplikasi resep masakan ini menyajikan fitur pencarian dan rekomendasi resep masakan yang cocok berdasarkan bahan makan yang tersedia didapat user.	Mobile Website
4	Eric Agung Wibowo. Universitas Surabaya	Pembuatan Aplikasi Resep Masakan Berbasis Android	Android SDK, Android Studio	Tidak dijelaskan Menggunakan apa.	Membantu pengguna aplikasi untuk mengoleksi resep masakan favorit dan mengatur daftar pembelian bahan makanan.	Android

STT - NF

2.3 Posisi Penelitian

Table 2. Posisi Penelitian

No	Judul	Metode	Platform	Bahasa Pemrograman	Database	Testing
1	Aplikasi Resep Masakan Tradisional	Multimedia Development Live Cycle (MDLC)	Android	Assebly	Tidak dijelaskan Menggunakan apa.	Black Box Testing
2	Sistem Pendukung Keputusan penentuan Resep Masakan Berdasarkan Ketersediaan Bahan Makanan menggunakan Metode Simple Additive Weighting berbasis Web	Simple Additive Weighting (SAW)	Website	PHP, HTML	MySQL	Black Box Testing
3	Rancang Bangun Aplikasi Masakan Berbasis Mobile Web dengan Metode Case-Based Reasoning	Case-Based Reasoning	Mobile Website	PHP, HTML	MySQL	Tidak dijelaskan Menggunakan apa.
4	Pembuatan Aplikasi Resep Masakan Berbasis Android	Tidak dijelaskan Menggunakan apa.	Android	Tidak dijelaskan Menggunakan apa.	Tidak dijelaskan Menggunakan apa.	Tidak dijelaskan Menggunakan apa.

5	RANCANG BANGUN APLIKASI PENCARIAN RESEP MASAKAN MENGUNAKAN ELASTICSEARCH DAN LARAVEL	Agile Development Scrum	Website	PHP, HTML, Javascript	MySQL	Black Box Testing dan UAT
---	--	-------------------------------	---------	--------------------------	-------	---------------------------



STT - NF