

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. *State of the Art***

Penelitian terdahulu sebagai bentuk standarisasi dalam mencari perbandingan dan memperkuat teori yang digunakan dalam penelitian yang dilakukan. Beberapa penelitian terdahulu yang judul tidak memiliki kesamaan, akan tetapi pada bahan kajian pada penelitian yang memiliki teori yang sama untuk memperkuat teori yang berhubungan. Dalam penelitian ini disertakan 3 jurnal penelitian sebelumnya yang berhubungan dengan konsep disposisi. Jurnal Ghazi dan Irfan (2018) menceritakan tentang aplikasi berbasis website yang berfungsi manajemen arsip dan disposisi di BPN Kota Padang karena proses yang masih kurang efektif dan efisien sehingga dapat memperlambat proses pengerjaan dan membuang banyak waktu. Pada aplikasi ini nantinya di dalam sistem aplikasi ini sebelum user masuk ke sistem, user terlebih dahulu login. Kepala urusan bertugas mengelola arsip data surat masuk dan surat keluar. Admin mengelola data semua user yang terdaftar. Sistem akan mendisposisikan surat masuk ke kepala seksi melalui kepala kantor. Disini masing-masing kepala seksi memiliki 5 bidang keahlian. Pada prosedur surat keluar caraka bertugas mengkonfirmasi serta mengirim surat ke masyarakat atau instansi lain melalui jasa kurir, Admin dapat mengakses data user yang diberi hak untuk akses ke sistem. Kepala urusan dapat mengelola data arsip surat masuk dan keluar. Kepala kantor bertugas dalam melakukan pendisposisian, sehingga muncul notifikasi disposisi masuk ke akun Kepala Seksi. Sedangkan Caraka dapat mengkonfirmasi apakah surat sudah terkirim atau belum dengan menggunakan fitur approval, kesimpulan yang dapat diambil salah satunya aplikasi ini dapat menghasilkan sistem yang dapat memberikan kemudahan dalam melakukan proses disposisi surat sehingga penerima disposisi tidak perlu bertatap muka dengan pimpinan. selain itu Irawan (2018) pada penelitiannya menceritakan tentang aplikasi disposisi berbasis web mobile pada STMIK MUSIRAWAS Lubuk Linggau, penelitian ini membahas tentang fitur yang menyesuaikan bagi pengguna mobile yang sangat berguna jika tidak menggunakan laptop dan lain-lain, Proses aplikasi ini mencakup penggunaan

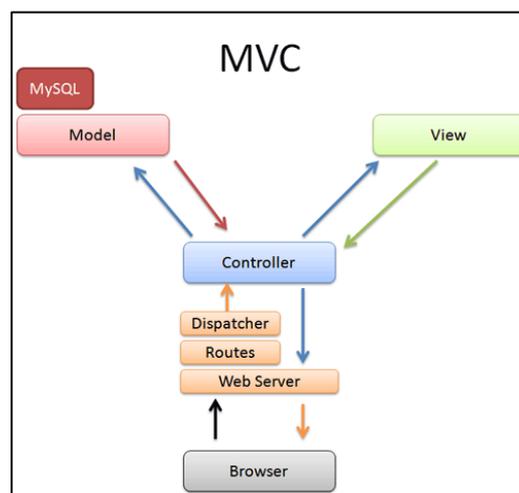
database yaitu mysql, dan bahasa pemrograman php versi 7, pada aplikasi ini dapat diakses staff dan siswa, alur kerja aplikasi ini adalah seluruh user akan diarahkan ke halaman login terlebih dahulu, setelah login maka dilanjutkan ke halaman utama, pada halaman ini staff dapat melakukan pengolahan data surat masuk dan surat keluar serta disposisi dari surat masuk yang ada, dengan memilih menu pengolahan data di sub menu sistem. Apabila memilih menu input data jenis surat maka di Halaman Input Data Jenis Surat akan tampil setelah memilih sub menu jenis surat serta memilih input jenis surat, pada halaman ini dapat digunakan user untuk melakukan pengolahan data jenis-jenis surat yang dapat dibuat untuk menjadi surat keluar dari STMIK Musirawas, selain halaman input data jenis surat terdapat halaman input data surat Halaman Input Data Buat surat digunakan untuk melakukan input data buat surat dimana nantinya data dari buat surat dapat digunakan untuk menjadi data surat keluar yang akan dikirim oleh STMIK Musirawas, selanjutnya terdapat Halaman Input Data Surat Keluar, halaman input data surat diawali dengan melakukan pencarian nomor surat keluar yang telah dibuat pada halaman buat surat. Setelah mendapatkan surat yang dicari, akan dilakukan proses input data surat keluar pada form input data surat keluar. Dan nanti akan di arahkan ke halaman hasil pencarian no surat yang menampilkan Halaman Output disposisi surat Disposisi Surat Masuk dan Pembuatan surat Keluar di STMIK Musirawas Lubuklinggau Berbasis Web Mobile yang di bangun ini menggunakan bahasa pemograman php dan database mysql dengan interface aplikasi berbasis Web Mobile sedangkan untuk menjalankan aplikasi ini menggunakan web browser seperti mozilla firefox, opera, safari dan lain lain pada semua *device platform*. Selain itu pada penelitian Haryati (2018) yang menceritakan tentang aplikasi disposisi yang didukung dengan teknologi SMS yang dipergunakan untuk sistem disposisi di perusahaan X. Aplikasi ini bertujuan sebagai solusi proses disposisi terhenti/terhambat apabila pelaku tidak ditempat. Pemanfaatan teknologi SMS sebagai salah satu solusi disposisi online tetap bisa dijalankan tanpa terkendala kuota/ internet. Tabel yang digunakan pada aplikasi ini diantaranya Tabel Departemen, Disposisi, Draft, Inbox, Jabatan, Jenis Surat, Karyawan, Lampiran Surat, Mitra, Notifikasi, Pesan, Surat dan user level. Pemanfaatan SMS gateway

dalam perancangan sistem ini menyesuaikan dengan kebutuhan lapangan yaitu perusahaan X dengan mengabaikan kuota/ jaringan internet untuk akses informasi. Penelitian ini diawali dengan identifikasi sistem disposisi perusahaan X.

## 2.2. Sistem

Sistem adalah sekelompok komponen dan elemen yang digabungkan menjadi satu untuk mencapai tujuan tertentu. Sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*systema*) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan. (Simangunsong, A.,2018). Sistem itu sendiri berasal dari bahasa latin yaitu *systema* atau bahasa yunani *systema* yang berarti suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Sistem juga merupakan sebuah kesatuan bagian-bagian yang saling memiliki hubungan yang berbeda dalam suatu wilayah, serta memiliki item-item sebagai penggerak.

## 2.3. Arsitektur Sistem MVC



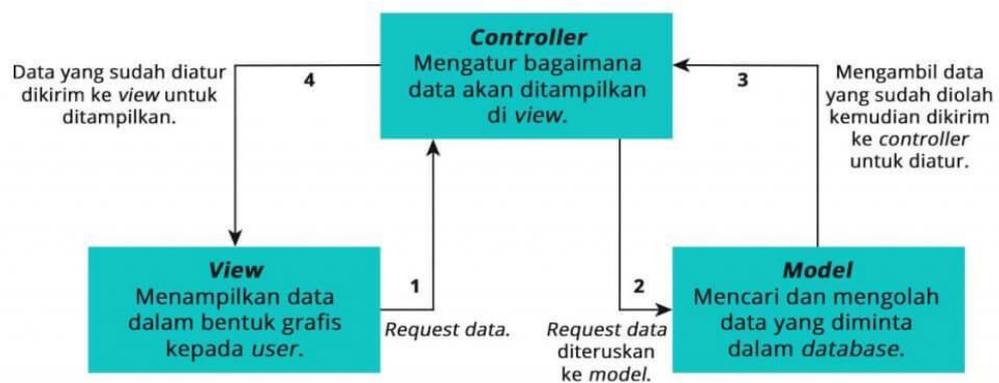
Gambar 2.1. MVC

Menurut Setiawan (2021), Model View Controller atau yang dapat disingkat MVC adalah sebuah pola arsitektur dalam membuat sebuah aplikasi dengan cara memisahkan kode menjadi tiga bagian yang terdiri dari :

- a. Model merupakan bagian yang bertugas untuk menyiapkan, mengatur, memanipulasi, dan mengorganisasikan data yang ada di database.
- b. *View* merupakan bagian yang bertugas untuk menampilkan informasi dalam bentuk *Graphical User Interface* (GUI).
- c. Controller merupakan bagian yang bertugas untuk menghubungkan serta mengatur model dan *view* agar dapat saling terhubung.

Setelah penjelasan mengenai MVC dan komponennya, terdapat alur model proses dari MVC itu sendiri yaitu :

- a. Proses pertama adalah *view* akan meminta data untuk ditampilkan dalam bentuk grafis kepada pengguna.
- b. Permintaan tersebut diterima oleh *controller* dan diteruskan ke model untuk diproses.
- c. Model akan mencari dan mengolah data yang diminta di dalam database
- d. Setelah data ditemukan dan diolah, model akan mengirimkan data tersebut kepada *controller* untuk ditampilkan di *view*.
- e. *Controller* akan mengambil data hasil pengolahan model dan mengaturnya di bagian *view* untuk ditampilkan kepada pengguna.



Gambar 2.2. Proses MVC

Alasan dari penggunaan penggunaan MVC untuk aplikasi disposisi ini adalah sebagai berikut :

a. Proses pengembangan aplikasi menjadi lebih efisien

Penggunaan MVC dapat mempercepat pengembangan aplikasi karena kode dapat dikerjakan oleh beberapa developer. Contohnya dalam kasus pengembangan aplikasi web, bagian model dan controller dapat dikerjakan oleh back-end developer sedangkan bagian view dapat dilakukan oleh front-end developer.

b. Dapat melakukan testing dengan lebih mudah

Untuk memastikan seluruh aplikasi bekerja sesuai dengan rencana maka langkah testing atau uji coba wajib dilakukan. Dengan menggunakan model view controller ini, maka proses uji coba dapat dilakukan pada setiap bagian.

c. Perbaikan bug atau error lebih cepat untuk diselesaikan

Penggunaan MVC dapat memudahkan developer untuk memperbaiki error atau bug yang terjadi. Developer dapat fokus untuk menemukan dan memperbaiki masalah yang terjadi karena kode dituliskan pada bagian-bagian terpisah.

d. Mempermudah pemeliharaan

Konsep MVC ini dapat mempermudah pemeliharaan aplikasi, karena script atau kode yang lebih rapi dan terstruktur sehingga mempermudah developer dalam proses pemeliharaan aplikasi.

Kesimpulannya MVC adalah metode dalam membuat aplikasi dengan cara memisahkan kode menjadi tiga bagian, yaitu Model, View, dan Controller. Karena memiliki tiga bagian yang saling berkaitan satu sama lain model view controller ini jadi memiliki banyak keuntungan untuk pengembangan dan pemeliharaan aplikasi.

#### **2.4. SIMORMAWA**

SIMORMAWA atau Sistem Informasi Organisasi Mahasiswa adalah salah satu aplikasi berbasis website dari server Polstri. Adapun kegunaannya adalah untuk mendigitalisasi pekerjaan yang ada di di bidang kemahasiswaan seperti mengurus disposisi, proposal, dan surat izin kegiatan lainnya. Pada hal ini akan dibahas

mengenai sistem disposisi pada aplikasi SIMORMAWA.

## **2.5. Disposisi**

Pengertian disposisi menurut Edward III dalam Widodo(2010:104) dikatakan sebagai “kemauan, keinginan dan kecenderungan para perlaku kebijakan untuk melaksanakan kebijakan tadi secara sungguh sungguh sehingga apa yang menjadi tujuan kebijakan dapat diwujudkan”. Edward III dalam Widodo (2010:104-105) mengatakan bahwa : jika implementasi kebijakan ingin berhasil secara efektif dan efisien, para pelaksana (implementors) tidak hanya mengetahui apa yang harus dilakukan dan mempunyai kemampuan untuk melakukan kebijakan tersebut, tetapi mereka juga harus mempunyai kamauan untuk melaksanakan kebijakan tersebut.

Faktor-faktor yang menjadi perhatian Edward III dalam Agustino (2006:159-160) mengenai disposisi dalam implementasi kebijakan terdiri dari: 1) Pengangkatan birokrasi. Disposisi atau sikap pelaksana akan menimbulkan hambatan-hambatan yang nyata terhadap implementasi kebijakan bila personel yang ada tidak melaksanakan kebijakan yang diinginkan oleh pejabat-pejabat yang lebih atas. Karena itu, pengangkatan dan pemilihan personel pelaksana kebijakan haruslah orang-orang yang memiliki dedikasi pada kebijakan yang telah ditetapkan, lebih khusus lagi pada kepentingan warga masyarakat. 2) Insentif merupakan salah-satu teknik yang disarankan untuk mengatasi masalah sikap para pelaksana kebijakan dengan memanipulasi insentif. Pada dasarnya orang bergerak berdasarkan kepentingan dirinya sendiri, maka memanipulasi insentif oleh para pembuat kebijakan mempengaruhi tindakan para pelaksana kebijakan. Dengan cara menambah keuntungan atau biaya tertentu mungkin akan menjadi faktor pendorong yang membuat para pelaksana menjalankan perintah dengan baik. Hal ini dilakukan sebagai upaya memenuhi kepentingan pribadi atau organisasi.

## **2.6. Aplikasi**

Aplikasi berasal dari kata application yaitu bentuk benda dari kata kerja to apply yang dalam bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu subkelas perangkat lunak komputer yang menggunakan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan

pemakai. Contoh utama perangkat lunak aplikasi adalah program pengolah kata, lembar kerja, dan pemutar media. Kumpulan aplikasi komputer yang digabung menjadi suatu paket biasanya disebut paket atau suite aplikasi (*application suite*).<sup>6</sup> Pengertian aplikasi secara umum adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya aplikasi merupakan suatu perangkat komputer yang siap pakai bagi user.

Pengertian aplikasi menurut para ahli adalah sebagai berikut :

a. Menurut Jogiyanto (1999:12) adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output.

b. Menurut Kamus Kamus Besar Bahasa Indonesia (1998 : 52) adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna.

c. Menurut Rachmad Hakim S, Aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (*game*), dan sebagainya. d. Menurut Harip Santoso, Aplikasi adalah suatu kelompok file (*form, class, report*) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi payroll, aplikasi fixed asset, dan lain-lain.

## **2.7. HyperText Markup Language (HTML)**

HTML merupakan sebuah bahasa pemrograman terstruktur yang dikembangkan untuk membuat halaman *website* yang dapat diakses atau ditampilkan menggunakan *web browser*. HTML sendiri secara resmi lahir pada tahun 1989 oleh tim Berners Lee dan dikembangkan oleh *World Wide Web Consortium (W3C)*, yang kemudian pada tahun 2004 dibentuklah *Web Hypertext Application Technology Working Group (WHATG)* yang hingga kini bertanggung jawab akan perkembangan bahasa HTML (Setiawan, 2017:16).

HTML merupakan singkatan dari *Hypertext Markup Language* yaitu bahasa standar *website* yang dikelola penggunaannya oleh W3C (*World Wide Web*

*Consortium*) berupa *tag-tag* yang menyusun setiap elemen dari *website*. HTML berperan sebagai penyusun struktur halaman *website* yang menempatkan setiap elemen *website* sesuai *layout* yang diinginkan. HTML biasanya disimpan dalam sebuah *file* berekstensi *.html*. Untuk mengetikkan *script* HTML dapat menggunakan *text editor* seperti *Notepad* sebagai paling bentuk sederhana atau *text editor* khusus yang dapat mengenali setiap unsur *script* HTML dan menampilkannya dengan warna yang berbeda sehingga mudah dibaca, seperti *Notepad++*, *Sublime Text*, dan masih banyak lagi aplikasi lain yang sejenis (Abdulloh, 2018:7).

HTML singkatan dari *Hyper Text Markup Language*, yaitu *script* yang berupa *tag-tag* untuk membuat dan mengatur struktur *website*. Beberapa tugas utama HTML dalam membangun *website* diantaranya sebagai berikut:

- a) Menentukan *layout website*.
- b) Memformat *text* dasar seperti pengaturan paragraf, dan format *font*.
- c) Membuat *list*.
- d) Membuat tabel.
- e) Menyisipkan gambar, video, dan audio.
- f) Membuat *link*.
- g) Membuat formulir (Josi, 2017:51-52).

## **2.8. Cascading Style Sheets (CSS)**

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen *website* yang berfungsi mengatur elemen HTML dengan berbagai *property* yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan. Sebagian orang menganggap CSS bukan termasuk salah satu bahasa pemrograman karena memang strukturnya sederhana, hanya berupa kumpulan-kumpulan aturan yang mengatur *style* elemen HTML.

Cara kerja CSS dalam memodifikasi HTML dengan memilih elemen HTML yang akan diatur kemudian memberikan *property* yang sesuai dengan tampilan yang diinginkan. Dalam memberikan aturan pada elemen HTML, *script* CSS terdiri dari atas 3 bagian yaitu *selector* untuk memilih elemen yang akan diberi aturan, *property* yang merupakan aturan yang diberikan dan *value* sebagai nilai dari aturan

yang diberikan (Abdulloh, 2018:45).

Selain itu, CSS juga menjelaskan bagaimana elemen-elemen HTML ditampilkan pada *layer* dengan lebih mulus. CSS dapat menghemat banyak pekerjaan dan mengontrol tata letak beberapa halaman *web* sekaligus. CSS dapat ditambahkan ke elemen HTML dalam 3 cara, yaitu:

1. *Inline*, yaitu dengan menggunakan atribut *style* dari elemen HTML.
2. *Internal*, yaitu dengan menggunakan elemen `<style>` yang ditempatkan pada area `<head>`.
3. *External*, yaitu dengan menggunakan *file* CSS eksternal (Saputra, 2019:47).

## 2.9. JavaScript (JS)

*JavaScript* merupakan bahasa pemrograman *website* yang pemrosesnya dilakukan di sisi *client*. Karena berjalan disisi *client*, *JavaScript* dapat dijalankan hanya dengan menggunakan *browser*. Berbeda dengan *PHP* yang bekerja disisi *server*, untuk menjalankan *script JavaScript* tidak memerlukan *refresh* pada *browser*. *JavaScript* biasanya dijalankan ketika ada *event* tertentu yang terjadi pada halaman *website*. Baik *event* yang dilakukan oleh *user*, maupun *event* yang terjadi karena adanya perubahan pada halaman *website* (Abdulloh, 2018:193).

## 2.10. JQuery

*JQuery* merupakan salah satu dari sekian banyak *JavaScript library*, yaitu kumpulan fungsi *JavaScript* yang siap pakai, sehingga mempermudah dan mempercepat dalam membuat kode *JavaScript* (Abdulloh, 2018:233).

*JQuery* adalah kumpulan fungsi-fungsi *JavaScript* yang sudah dibentuk sebagai suatu objek. Sehingga penggunaan *jQuery* ini bisa dikategorikan sebagai suatu *library* yang nantinya kita hanya perlu menggunakan fungsi-fungsi di dalam *library* tersebut.

*JQuery* pertama kali dirilis oleh John Resig pada tahun 2006. Pada perkembangannya *jQuery* tidak sekedar sebagai *library javascript*, namun memiliki keandalan dan kelebihan yang cukup banyak. Hal tersebut menyebabkan banyak *developer web* menggunakannya. *jQuery* dikenal dengan slogan “*Write less, do more*” artinya penulisan kode yang sedikit tetapi memiliki beberapa aksi (*action*) (Hidayatullah, 2017:399).

### 2.11. Perl Hypertext Preprocessor (PHP)

Menurut Josi (2017:51) PHP berasal dari kata *Perl Hypertext Preprocessor*, yaitu bahasa pemrograman yang bersifat *universal* dan dapat digunakan bersamaan dengan bahasa pemrograman HTML. PHP digunakan untuk melakukan penanganan, pembuatan dan pengembangan sebuah situs *web*.

*Perl Hypertext Preprocessor* atau disingkat PHP adalah suatu bahasa *scripting* khususnya digunakan untuk *web development*. Karena sifatnya yang *server side scripting*, maka untuk menjalankan PHP harus menggunakan *web server*. PHP juga dapat diintegrasikan dengan HTML, *JavaScript*, *JQuery*, *Ajax*. Namun, pada umumnya PHP lebih banyak digunakan bersamaan dengan *file* bertipe HTML. Dengan menggunakan PHP anda bisa membuat *website* yang dinamis dengan disertai manajemen *database*-nya (Hidayatullah, 2017:231).

PHP merupakan kependekan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *website* yang dapat disisipkan dalam *script* HTML dan bekerja disisi *server*. Tujuan dari bahasa ini adalah membantu para pengembang *website* untuk membuat *website* dinamis dengan cepat.

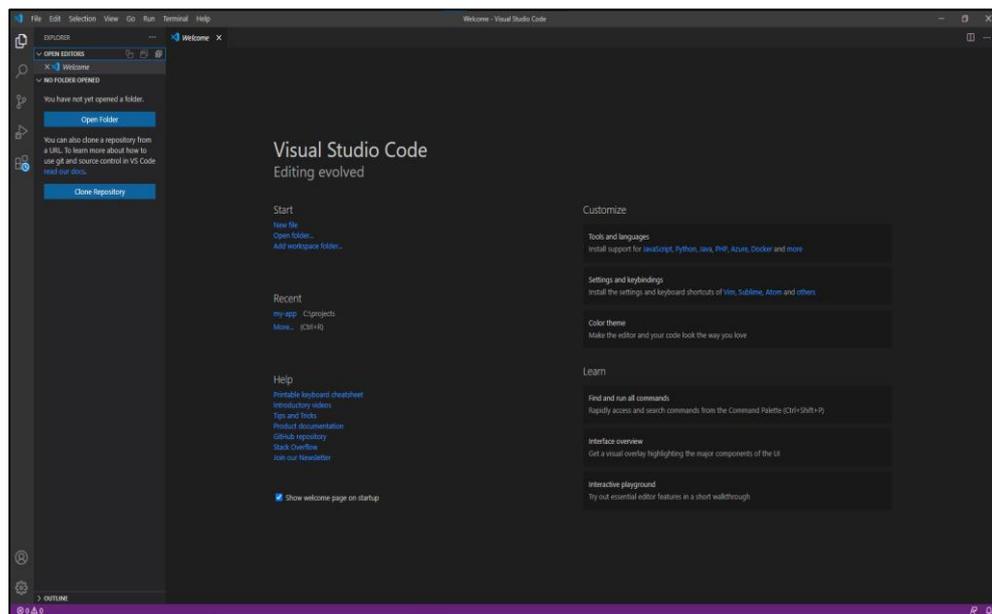
Agar dapat menjalankan PHP harus menyediakan perangkat lunak berikut sebagai :

1. *Web server (Apache, IIS, Personal Web Server/PWS)*.
2. *PHP server*.
3. *Database server (MySQL, Interbase, MS SQL, dan lain-lain)*.

Aplikasi diatas tidak perlu diinstal satu per-satu, karena biasanya telah tersedia paket aplikasi yang telah menyediakan ketiganya dalam satu paket aplikasi seperti XAMPP atau *Appserv* (Abdulloh, 2018:127).

### 2.12. Visual Studio Code

*Visual Studio Code (VS Code)* adalah kode editor yang bersifat *open-source* untuk pengembangan dan *debugging* aplikasi *cloud* dan situs *web modern* yang tersedia secara gratis di Linux, OS X dan Windows. *VS Code* mendukung lebih dari 30 bahasa pemrograman, *markup* dan *database* yang berbeda, beberapa di antaranya adalah PHP, HTML, CSS dan SQL (Gamma, 2016:5).



Gambar 2.3. Tampilan Visual Studio Code

(Sumber: Dokumen Pribadi)

### 2.13. Framework

*Framework* merupakan kumpulan fungsi–fungsi/prosedur–prosedur dan *class-class* untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang *programmer*, tanpa harus membuat fungsi atau *class* dari awal (Supardi dan Hermawan, 2018:2).

### 2.14. Bootstrap

*Bootstrap* merupakan salah satu *framework* CSS yang paling populer dari sekian banyak *framework* CSS yang ada. *Bootstrap* memungkinkan desain sebuah *website* menjadi *responsive* sehingga dapat dilihat dari berbagai macam ukuran *device* dengan tampilan tetap menarik. *Bootstrap* juga membuat proses pengaturan desain menjadi lebih cepat karena tidak perlu lagi banyak menulis CSS, bahkan hampir tidak perlu, kecuali memerlukan pengaturan desain yang berbeda dengan *style bootstrap*. *Bootstrap* telah didukung oleh hampir semua *browser* baik pada *desktop* maupun *mobile* (Abdullah, 2018:261).

### 2.15. Laravel

Laravel merupakan satu-satunya *framework* yang membantu Anda untuk memaksimalkan penggunaan PHP di dalam proses pengembangan *website*. PHP

menjadi bahasa pemrograman yang sangat dinamis, tapi semenjak adanya Laravel, dia menjadi lebih powerful, cepat, aman, dan simpel. Setiap rilis versi terbaru, Laravel selalu memunculkan teknologi baru di antara framework PHP lainnya. Laravel menawarkan beberapa keuntungan ketika Anda mengembangkan website menggunakan dasar framework ini.

Pertama, website menjadi lebih scalable (mudah dikembangkan).

Kedua, terdapat namespace dan tampilan yang membantu Anda untuk mengorganisir dan mengatur sumber daya website.

Ketiga, proses pengembangan menjadi lebih cepat sehingga menghemat waktu karena Laravel dapat dikombinasikan dengan beberapa komponen dari framework lain untuk mengembangkan website. Selain itu, ada dua tools Laravel yang jarang dimiliki oleh framework lain (kecuali Symfony), yaitu Composer dan Artisan.

#### 1. Composer

Composer merupakan *tool* yang di dalamnya terdapat *dependencies* dan kumpulan *library*. Seluruh *dependencies* disimpan menggunakan format file `composer.json` sehingga dapat ditempatkan di dalam folder utama *website*. Inilah mengapa composer terkadang dikenal dengan *dependencies* management.

Composer membantu dalam proses untuk menginstall *library* yang dibutuhkan oleh *library* Google reCaptcha. Jadi jika menggunakan composer Anda tinggal menginstall *library* Google reCaptcha dan secara otomatis *library* lain akan terinstall.

Begitu pun ketika ingin memperbarui *library*, Anda cukup menggunakan perintah “`$ composer update`” dan satu per satu *library* akan diperbarui secara otomatis.

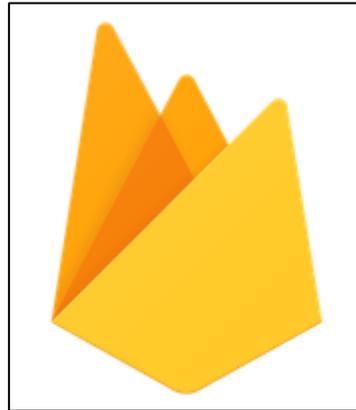
#### 2. Artisan

Artisan merupakan *command line interface* yang dimiliki oleh Laravel. Artisan mencakup sekumpulan perintah yang membantu Anda untuk membangun sebuah website atau aplikasi web.

Kumpulan perintah Artisan juga termasuk penggabungan dengan framework Symfony yang menghasilkan fitur add-on di Laravel 5.1 (sekarang sudah masuk

ke versi Laravel 5.8). Dengan adanya fitur add-on, dan nanti bisa menambahkan berbagai macam fitur baru ke Laravel.

### 2.16. Firebase



Gambar 2.4. Logo Firebase

Firebase memiliki produk utama, yaitu menyediakan database realtime dan backend sebagai layanan (Backend as a Service). Layanan ini menyediakan pengembang aplikasi API yang memungkinkan aplikasi data yang akan disinkronisasi di klien dan disimpan di cloud Firebase ini. Firebase menyediakan library untuk berbagai client platform yang memungkinkan integrasi dengan Android, iOS, JavaScript, Java, Objective-C dan Node aplikasi Js dan dapat juga disebut sebagai layanan DbaaS (Database as a Service) dengan konsep realtime. 8 Firebase digunakan untuk mempermudah dalam penambahan fitur-fitur yang akan dibangun oleh developer.

Semua data Firebase Realtime Database disimpan sebagai objek JSON. Bisa dianggap basis data sebagai JSON tree yang di-host di awan. Tidak seperti basis data SQL, tidak ada tabel atau rekaman. Ketika ditambahkan ke JSON tree, data akan menjadi simpul dalam struktur JSON yang ada. Meskipun basis data menggunakan JSON tree, data yang tersimpan dalam basis data bisa diwakili sebagai tipe bawaan tertentu yang sesuai dengan tipe JSON yang tersedia untuk membantu Anda menulis lebih banyak kode yang bisa dipertahankan, kemudian ada empat metode untuk menulis data ke Firebase Realtime Database seperti pada tabel 2.1 berikut :

Tabel 2.1. Metode Menulis Data ke Firebase

Metode	Penggunaan Umum
setValue()	Menulis atau mengganti data ke jalur yang didefinisikan, seperti users/<user-id>/<username>
push()	Tambahkan ke daftar data. Setiap kali Anda memanggil push(), Firebase akan menghasilkan ID unik, seperti user-posts/<user-id>/<unique-post-id>
updateChildren()	Memperbarui beberapa kunci untuk jalur yang didefinisikan tanpa mengganti semua data.
runTransaction()	Memperbarui data kompleks yang bisa rusak karena pembaruan bersamaan.

Untuk operasi tulis dasar, Anda bisa menggunakan setValue() untuk menyimpan data ke referensi yang ditetapkan, menggantikan data yang ada di jalur tersebut. Fungsi dalam pengambilan data melalui Firebase dapat dilihat pada table 2.2 berikut ini:

Tabel 2.2. Callback Kejadian dalam Pengambilan Data Firebase

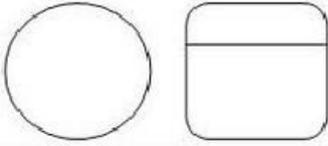
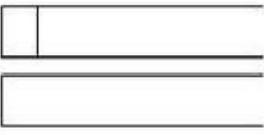
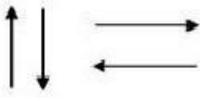
Listener	Callback kejadian	Penggunaan biasa
ValueEventListener	onDataChange()	Membaca dan mendengarkan perubahan untuk seluruh konten jalur.
ChildEventListener	onChildAdded()	Mengambil daftar item atau mendengarkan penambahan daftar item. Disarankan untuk digunakan dengan onChildChanged() dan onChildRemoved() untuk memantau perubahan daftar.
	onChildAdded()	Mendengarkan perubahan pada item dalam daftar. Gunakan dengan onChildAdded() dan onChildRemoved() untuk memantau perubahan daftar.
	onChildRemoved()	Mendengarkan item yang dibuang dari daftar. Gunakan dengan onChildAdded() dan onChildChanged() untuk memantau perubahan daftar.
	onChildMoved()	Gunakan dengan data diurutkan untuk mendengarkan perubahan dalam prioritas item.

Untuk menambahkan listener kejadian, gunakan metode `addValueEventListener()` atau `addListenerForSingleValueEvent()`. Untuk menambahkan listener kejadian anak, gunakan metode `addChildEventListener()`. Metode `onDataChange()` untuk membaca cuplikan statis konten pada jalur tertentu, seperti yang telah ada pada saat kejadian. Metode ini terpicu satu kali 10 ketika listener terpasang dan terpicu lagi setiap kali terjadi perubahan data, termasuk anaknya. Callback kejadian meneruskan cuplikan yang berisi semua data di lokasi tersebut, termasuk data anak. Jika tidak ada data, cuplikan yang dikembalikan adalah null. Metode `onDataChange()` dipanggil setiap kali terjadi perubahan data pada referensi database yang ditetapkan, termasuk perubahan ke anaknya. (Firebase, 2015).

### **2.17. Data Flow Diagram**

Menurut Saputra (2018:11), Data Flow Diagram merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau ke entitas. Data Flow Diagram juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dan transformasi yang digunakan sebagai perjalanan data dari input atau masukan menuju keluaran atau output. Berikut ini adalah simbol Data Flow Diagram menurut Saputra, yang dapat dilihat pada tabel 2.1 :

Tabel 2.3. Tabel Data Flow Diagram

Simbol	Keterangan
	<i>External Entity</i> , merupakan kesatuan di lingkungan luar sistem yang bisa berupa orang, organisasi atau sistem lain.
	<i>Process</i> , merupakan proses seperti perhitungan aritmatik penulisan suatu formula atau pembuatan laporan
	<i>Data Store (Simpan Data)</i> , dapat berupa suatu file atau database pada sistem komputer atau catatan manual
	<i>Data Flow ( arus data)</i> , arus data ini mengalir diantara proses, simpan data dan kesatuan luar

### 2.18. Use Case Diagram

Rosa dan M. Shalahudin (2015:155), use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah simbol- simbol yang ada pada diagram use case yang dapat dilihat pada Tabel 2.4 :

Tabel 2.4. Simbol Use Case Diagram

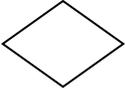
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>Sistem</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya ( <i>sinergi</i> ).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

## 2.19. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas berfungsi untuk menggambarkan aktivitas sistem bukan yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2018:161).

Berikut pada Tabel 2.5 adalah simbol-simbol yang digunakan dalam *activity diagram* disertai dengan keterangan fungsinya:

Tabel 2.5. Simbol Activity Diagram

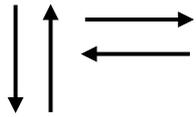
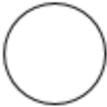
No.	Simbol	Keterangan
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan / <i>Decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan / <i>Join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

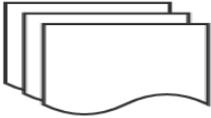
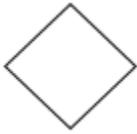
(Sumber: Rosa dan Shalahuddin, 2018:162-163)

## 2.20. Flowchart

Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan flowchart akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu flowchart juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek. Flowchart membantu memahami urutan-urutan logika yang rumit dan panjang. Flowchart membantu mengkomunikasikan jalannya program ke orang lain (bukan pemrogram) akan lebih mudah (Santoso, 2017:86).

Tabel 2.6. Simbol Flowchart

NO	Simbol	Keterangan
1.	Flow Direction Symbol 	Untuk menghubungkan antara simbol yang satu dengan simbol yang lain atau menyatakan jalannya arus dalam suatu proses.
2.	Terminal (mulai atau berhenti) 	Simbol ini digunakan untuk menunjukkan awal kegiatan ( <i>start</i> ) atau akhir dari suatu kegiatan ( <i>stop</i> ).
3.	Input dan Output 	Untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
4.	Proses (Pengolahan) 	Untuk menunjukkan pengolahan yang dilakukan oleh komputer atau PC.
5.	Connector 	Simbol suatu keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang sama.
6.	Offline Connector 	Simbol untuk keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang berbeda.
7.	Document 	Untuk menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas.

8.	<p>Manual Input</p> 	<p>Berfungsi untuk pemasukan data secara manual <i>on-line keyboard</i>.</p>
9.	<p>Preparation</p> 	<p>Berfungsi untuk mempersiapkan penyimpanan yang sedang/ akan digunakan sebagai tempat pengolahan didalam <i>storage</i>.</p>
10.	<p>Manual Operation</p> 	<p>Berfungsi untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer atau PC.</p>
11.	<p>Multiple Document</p> 	<p>Sama seperti symbol document, hanya saja dokumen yang digunakan lebih dari satu dalam simbol ini.</p>
12.	<p>Predefined</p> 	<p>Untuk pelaksanaan suatu bagian (subprogram) / prosedur.</p>
13.	<p>Decision (Keputusan)</p> 	<p>Menunjukkan suatu perbandingan yang harus dibuat bila hasilnya “ya”, maka alir data menunjukkan ke suatu tempat, bila “tidak” maka akan menuju ke tempat lain.</p>
14	<p>Display</p> 	<p>Simbol yang digunakan untuk menyatakan perangkat output yang digunakan.</p>

15.	<p data-bbox="491 309 600 338">Database</p> 	<p data-bbox="715 309 1278 389">Simbol yang digunakan untuk menyatakan data disimpan dalam database.</p>
16.	<p data-bbox="443 510 647 539">On-line stograge</p> 	<p data-bbox="715 510 1350 591">Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.</p>