BAB II TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian pertama yang dilakukan oleh Rahayu, Nurhaeni, & Rohmah (2015)

yang berjudul "Sistem Persediaan Alat Tulis Kantor Sebagai Penunjang Pengambilan Keputusan Bagian Logistik di Perguruan Tinggi Raharja" bertujuan untuk menganalisa dan merancang sistem persediaan alat tulis kantor yang dapat menyajikan laporan persediaan dengan akurat dan cepat. Perancangan sistem ini berfungsi untuk menggantikan sistem lama yang masih menggunakan sistem semi komputerisasi yaitu penggunaan aplikasi *spreadsheet* untuk menyimpan datanya sehingga menyebabkan adanya penumpukan stok barang yang jarang digunakan atau tidak tersedianya barang yang dibutuhkan. Dengan penerapan sistem persediaan alat tulis kantor dapat memberikan informasi persediaan yang akurat, cepat dan sesuai dengan kebutuhan pengguna.

Penelitian selanjutnya oleh Zahruddin, Handayani, & Santika (2018) dengan judul "Merancang Sistem Pengelolahan ATK Berbasis Web pada PT. Arthaasia Finance" bertujuan untuk merancang sistem pengolahan ATK yang dapat mempermudah pengguna dalam mengakses informasi, mempermudah karyawan dalam melakukan pekerjaan dengan cepat dan efisien, serta dapat memudahkan staff dalam melihat data ATK. Metode SDLC Waterfall merupakan acuan dari metode pengembangan yang digunakan sistem. Dalam sistem ini juga terdapat pencetakan

laporan berdasarkan pada data yang ada sehingga dapat mengurangi kesalahan dalam pencatatan yang dilakukan jika menggunakan sistem manual.

Penelitian lainnya Hermawati (2018) yang berjudul "Aplikasi Sistem Perpustakaan Terintegrasi dengan Notifikasi SMS dan *Email Reminder*" bertujuan untuk mempermudah dalam pencarian informasi bahan pustaka, mempermudah proses peminjaman bahan pustaka, dan memberikan informasi ketersediaan bahan pustaka dengan lebih cepat dan efektif, serta memberikan fitur notifikasi informasi perpustakaan terintegrasi dengan sms dan *email reminder*. Dengan diterapkannya fitur *email reminder* dalam sistem dapat membantu meningkatkan efektivitas pengunjung perpustakaan dalam mengembalikan bahan pustaka dibandingkan dengan diingatkan lewat orang lain.

Penelitian berikutnya oleh Dinata, Widodo, & Sukmaaji (2016) yang berjudul "Rancang Bangun Aplikasi *Workflow* Persetujuan Permintaan Kebutuhan *Workshop* pada Departemen HSE PT. Bangun Sarana Baja" bertujuan agar pengguna aplikasi dapat melakukan proses persetujuan terhadap pengajuan permintaan kebutuhan *workshop* secara digitalisasi melewati akses internet sehingga pengguna dapat memberikan persetujuan permintaan barang dari tempat manapun. Selain itu, perancangan aplikasi juga dapat membantu pengelolahan data dan menampilkan laporan dokumen sesuai dengan data yang ada.

Penelitian terakhir oleh Nawang et al. (2017) dengan berjudul "Rancang Bangun Sistem Informasi Pengolahan Data Persediaan Barang Berbasis Dekstop Dengan Model *Waterfall*" bertujuan membangun suatu sistem yang menerapkan teknologi informasi agar dapat menghasilkan informasi yang informatif dan akurat.

Tabel 2.1 Tinjuan Pustaka

Peneliti	Tahun	Penerapan Penelitian
Rahayu,	2015	Menerapkan pengembangan dari kerangka konsep
Nurhaeni, &		proses perancangan aplikasi inventory alat tulis kantor
Rohmah		(ATK).
Zahruddin,	2018	Menerapkan metode penelitian didasarkan pada metode
Handayani, &		SDLC Waterfall dalam merancang sistem inventory.
Santika		
Hermawati	2018	Menerapkan fitur email reminder dalam sistem
		inventory sehingga dapat memberikan informasi kepada
~ V		pengguna jika barang telah dikonfirmasi oleh admin.
Dinata,	2016	Menerapkan workflow request barang sederhana
Widodo, &		sehinga pengguna sistem dapat meminta barang melalui
Sukmaaji		internet secara langsung tanpa diperlukan komunikasi
Sec. 1		secara tetap muka.
Nawang,	2017	Menerapkan sistem persediaan barang yang dapat
Kurniawati, &		mencatat proses barang masuk dan barang keluar yang
Duta		baik sehingga informasi yang dihasilkan cepat, tepat,
		dan akurat.

Kesimpulan yang didapatkan oleh penulis berdasarkan dari hasil penelitianpenelitian tersebut berupa pengembangan aplikasi sistem informasi *inventory* bertujuan untuk membantu meningkatkan kecepatan proses pendataan barang sehingga data yang diperoleh dapat lebih akurat, cepat dan efisien sehingga laporan hasil rekapitulasi dari data-data tersebut dapat berisi informasi yang berguna untuk pengguna melalui akses internet. Selanjutnya, konsep alur perancangan sistem *inventory* ATK diterapkan dan dikembangkan berdasarkan konsep dari Rahayu, Nurhaeni, & Rohmah (2015) dengan menerapkan metode SDLC Waterfall dari penelitain Zahruddin, Handayani, & Santika (2018) disertai dengan penambahan fitur-fitur berupa pengiriman notifikasi bahwa permintan barang telah disetujui oleh admin melalui email berdasarkan penelitian Hermawati (2018) dan konsep kerja request barang dari pengguna kepada admin

berdasarkan penelitian Dinata, Widodo, & Sukmaaji (2016). Terakhir sistem *inventory* ATK juga menerapkan penelitian Nawang et al. (2017) dalam perancangan sistem persediaan barang agar dapat membangun suatu sistem yang menerapkan teknologi informasi agar dapat menghasilkan informasi yang informatif dan akurat.

2.2 Landasan Teori

2.2.1 Sistem Informasi

Menurut penelitian Soedijono, Fatta, Road, & Catur (2017), pengertian dari sistem informasi merupakan suatu sistem yang digunakan di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian sehingga dapat mendukung fungsi manajerial operasional organisasi dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kebutuhan pihak luar tertentu dengan berbagai laporan yang dibutuhkan. Adapun beberapa manfaat adanya sistem informasi bagi suatu organisasi yaitu dapat menyajikan informasi guna mendukung pengambilan keputusan dan mendukung operasi harian serta dapat menyajikan informasi berhubungan dengan kepengurusan.

2.2.2 Sistem *Inventory*

Menurut Fahrisal, Pohan, & Nasution (2018), sistem *inventory* adalah suatu proses utuk mengatur dan mendatakan barang mentah yang terdapat dalam proses produksi perusahaan dan barang hasil jadi yang disediakan untuk memenuhi

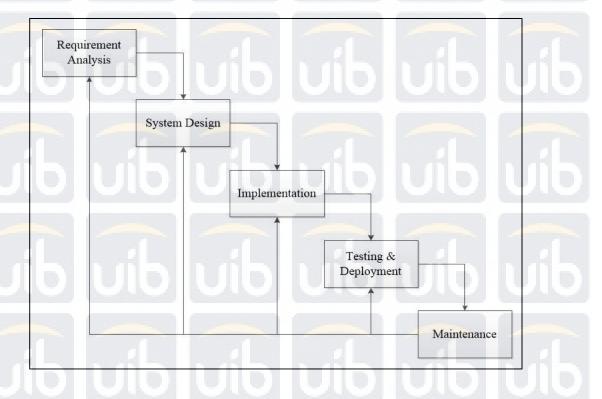
permintaan dari konsumen setiap waktu yang disimpan dan dirawat menurut aturan tertentu dalam keadaan siap pakai dan tersimpan dalam database.

Contoh penerapan sistem *inventory* dapat dilihat dari penelitian yang dilakukan oleh Karlena & Sudarmadi (2015) pada Otoritas Jasa Keuangan, lembaga pengawasan jasa keuangan, sebelum pencatatan inventory masih dilakukan secara manual. Pada pengunaan pencatatan manual terjadinya informasi yang salah dapat mengakibatkan proses pembuatan laporan rencana pemakaian barang yang lambat. Terjadinya ketidaksesuaian jumlah stok barang yang ada di dalam gudang dengan jumlah stok barang yang tercatat secara manual menjadi sebuah masalah pada sistem lama sehingga dengan adanya sebuah sistem *inventory* diharapkan bisa mengatasi permasalahan yang terjadi. Setelah penerapan sistem *inventory* ATK, terjadi peningkatan kecepatan, akurasi, dan data yang ada dapat dipertanggungjawabkan, serta dapat memudahkan user dalam mengakses dan mengambil data yang ada dalam bentuk laporan sehingga dapat membantu perusahaan dalam mengambil keputusan.

2.2.3 Metode Waterfall

Menurut Junaedy & S (2017), Winston Royce memperkenalkan suatu model pengembangan perangkat lunak yang memiliki aliran sistem linier yaitu model klasik waterfall. Metode Waterfall merupakan metode pengembangan sistem yang memiliki urutan yang jelas, yang memiliki cara kerja air terjun sehingga diterapkan pada proes pengembangan terus mengalir dari atas ke bawah melewati berbagai fase seperti

analisis kebutuhan, desain sistem, implementasi, pengujian dan pemeliharaan. Menurut Karlena & Sudarmadi (2015) langkah-langkah pada metode *waterfall* berupa:



Gambar 2.1 Metode Waterfall

1 Requirement Analysis

Pada tahapan analisis kebutuhan akan melakukan suatu analisis terhadap kebutuhan sistem. Beberapa cara pengumpulan data yang dilakukan dengan penelitian, wawancara atau studi literatur. Informasi akan dikumpulkan dan dicatat oleh sistem analis bedasarkan pengumpulan data dari pengguna sehingga dapat membentuk suatu gagasan dasar yang dapat menyelesaikan solusi dari permasalahan pengguna. Hasil yang didapatkan dari tahapan ini berupa dokumen informasi (*user requirement*) yang berisi harapan dari pengguna dalam perancangan sistem baru. Kemudian dokumen *user*

requirement ini akan menjadi acuan gagasan dasar untuk sistem analisis dalam perancangan sistem yang dapat menyelesaikan solusi pengguna.

2 System Design

Pada tahapan desain sistem akan melakukan suatu penerjemahan dari syarat kebutuhan menjadi suatu tampilan dasar antarmuka perangkat lunak sebelum melakukan tahap penulisan kode pemrograman. Titik fokus pada tahapan ini berupa representasi *interface*, arsitektur *software*, dan *data structure*, serta algoritma prosedural. Hasil yang diharapkan dari tahapan ini disebut dengan *software* requirement. Dokumen softwaren requirement kemudian akan digunakan oleh programmer untuk merancang sistem yang akan dibuat.

Implementation

Pada tahapan ini dilakukannya penerjemahan dari design menjadi sistem yang dilakukan oleh seorang programmer dengan bahasa-bahasa pemrograman. Hasil yang didapatkan dalam tahap ini yaitu sebuah aplikasi yang telah dilengkapi dengan solusi-solusi dari permasalahan yang dihadapi oleh pengguna.

4 Testing & Deployment

Pada tahapan ini dilakukannya suatu analisa secara *logic* dan fungsional terhadap kinerja sistem secara menyeluruh agar sistem yang dihasilkan dapat bekerja dengan baik tanpa terjadi permasalahan yang fatal. Peluncuran aplikasi akan dilakukan apabila sistem yang dirancang telah melewati tahap pengujian.

Maintenance

5

Pada tahapan ini merupakan suatu tahapan dalam pemeliharaan sistem agar sistem dapat bekerja sesuai dengan harapan awal perancangan sistem. Pemeliharaan sistem diperlukan karena kadang kala sistem yang dirancang terjadi perubahan sehingga menciptakan suatu masalah yang dapat berakar dari penyesuaian sistem dengan lingkungan peripheral baru maupun sistem operasi baru ataupun dikarenakan dibutuhkannya suatu perkembangan fungsional oleh pengguna.

2.2.4 System Development Life Cycle (SDLC)

Menurut Aswati, Ramadhan, Firmansyah, & Anwar (2017), metode SDLC adalah metode klasik yang sering diterapkan dalam perancangan, pengembangan dan pemeliharaan sistem informasi. Metode yang sering digunakan merupakan pendekatan air terjun (*waterfall approach*), yang dalam pengembangan sistem memiliki beberapa tahapan yang akan digunakan. Berikut merupakan tahapan-tahapan yang digunakan dalam SDLC menurut Aswati et al. (2017):

System planning atau tahap perencanaan mendefinisikan berbagai perencanaan kebutuhan sumber daya dan merupakan tahapan awal dari pengembangan sistem. Adapun beberapa contoh dari sumber daya seperti perangkat fisik, teknik atau metode, manusia, dan anggaran yang sifatnya masih umum.



- . System analysis atau tahap analisis sistem merupakan suatu tahapan untuk melakukan suatu penelitian terhadap sistem yang telah ada yang bertujuan untuk melakukan perancangan sistem yang baru atau mengembangkan sistem yang lama.
- 3. System design atau tahap desain sistem merupakan suatu tahapan yang dilakukan setelah tahapan analisis sistem yang menentukan proses dan data yang diperlukan untuk perancangan sistem baru maupun pengembangan sistem lama. Desain sistem dapat dibedakan menjadi desain sistem terinci dan desain sistem umum.
- 4. System implementation atau tahap implementasi merupakan tahapan yang membuat desain sistem yang telah dilakukan pada tahapan sebelumnya menjadi suatu aplikasi yang siap pakai.
- 5. *Maintenance System* atau tahap perawatan sistem merupakan tahapan yang dilakukan setelah tahap implementasi. Tahap ini meliputi pengajaan sistem, audit sistem, penggunaan sistem, perbaikan sistem dan peningkatan sistem.

2.2.5 FlowChart

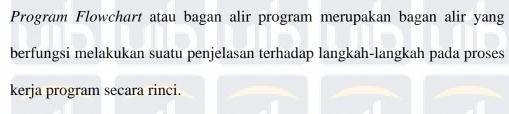
Menurut Setiawan, Sutardi, & Tajidun (2017), pengertian dari *flowchart* merupakan suatu penggambaran mengenai logika dan proses dari seluruh kegiatan pengumpulan data secara sistematis atau penyajian urutan *procedural* suatu program secara grafik. Analis dan programmer dapat memecahkan masalah secara detail bedasarkan berbagai segmen kecil melalui *flowchart*. Berbagai alternative yang

mungkin terjadi dalam pengoperasian sistem juga dianalisis oleh *flowchart*. Selain itu, sistem flowchart juga merupakan suatu urutan proses yang menunjukkan media *input*, media *output* dan media *storage* dalam proses pengelolaan data. Disisi lain, *flowchart* juga berarti suatu bagan yang berisi berbagai simbol khusus yang dapat menjelaskan urutan dari proses secara jelas beserta hubungan dari masing-masing proses yang terdapat dalam suatu program.

Berikut merupakan lima jenis flowchart menurut penelitian Sumarlinda (2016):

- System Flowchart atau bagan alir sistem merupakan bagan yang menunjukkan arus proses kerja secara menyeluruh. Prosedur yang ada di sebuah sistem akan dijelaskan pada bagan ini.
- 2. *Document Flowchart* atau bagan alir dokumen merupakan bagan alir formulir yang menunjukkan arus dari sebuah laporan beserta formulir dan termasuk dengan berbagai tembusannya.
 - Schematic Flowchart atau bagan alir skematik merupakan bagan alir yang memiliki fungsi yang mirip dengan bagan alir sistem yang menggambarkan prosedur-prosedur yang ada di dalam sistem. Yang menjadi perbedaan dari mereka berdua adalah pada bagan alir sistem hanya menggunakan simbol-simbol, sedangkan bagan alir skematik tidak hanya simbol saja, tetapi terdapat juga gambar-gambar berupa peralatan komputer yang digunakan.





Proccess Flowchart atau bagan alir proses merupakan bagan alir yang menunjukkan jarak kegiatan antara suatu kegiatan dengan kegiatan lainya dan pengunaan simpanan dalam suatu prosedur serta dapat menunjukkan waktu yang diperlukan oleh suatu kegiatan.

2.2.6 Activity Diagram

5.

Berdasarkan penelitian dari Dalal & Hooda (2017), alur *logic* sistem dapat diwakilkan dengan sederhana dan mudah menggunakan *activity diagram*. Activity diagram berisi penjelasan dari tahapan-tahapan dari aktivitas dan penerapan berbagai notasi yang terinspirasi dari flowchart.

Menurut Hendini (2016), activity diagram memiliki fungsi untuk menampilkan gambaran terhadap aliran kerja sistem ataupun aktivitas dari sebuah sistem.

 Tabel 2.2 Simbol Activity Diagram

Gambar	Keterangan
	Start Point merupakan awal aktivitas
	yang diletakkan pada pojok kiri atas.

ub uio uib	End Point merupakan akhir dari aktivitas
6 (450)	Activities menggambarkan suatu proses kegiatan bisnis
	Decision Points menggambarkan pilihan
	untuk pengambilan keputusan true atau
	false
	Swimlane merupakan pembagian activity
طنب طنب طنب	diagram untuk menunjukkan siapa yang melakukan apa.
	Fork digunakan untuk menunjukkan
	kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan
	paralel menjadi satu
	Join digunakan untuk menunjukkan
	adanya dekomposisi
	حانيما حانيما حانيا

2.2.7 Sequence Diagram

Berdasarkan penelitian dari Isa & Hartawan (2017), *sequence diagram* menjelaskan interaksi atau hubungan yang ada antara berbagai objek pada satu kumpulan dalam kurun waktu yang berurut yang memiliki fungsi untuk menjelaskan

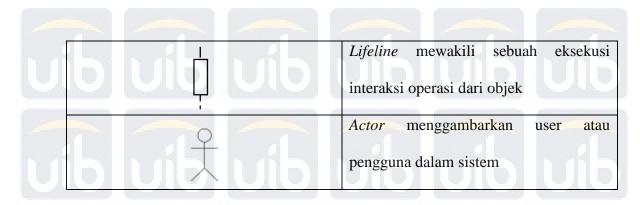
pengiriman serangkaian pesan antar objek beserta penjelasan interaksi antar objek yang terjadi pada titik tertentu dalam sistem.

Menurut Hendini (2016), fungsi dari *sequence diagram* adalah suatu penggambaran aktivitas objek pada *use case* dengan menjelaskan waktu aktif objek tersebut beserta pesan yang akan dikirimkan dan diterima antar objek.

Tabel 2.3 Simbol Sequence Digram

Gambar	Keterangan
	Entity Class merupakan bagian dari
	sistem yang berisi kumpulan kelas
	berupa entitas-entitas yang membentuk
	gambaran awal sistem dan menjadi
	landasan untuk menyusun basis data
	Boundary Class berisi kumpulan kelas
	yang menjadi interaksi antara satu atau
	lebih aktor dengan sistem
	Control Class mrupakan suatu objek
i b lu G uib l	yang berisi aplikasi yang tidak memiliki
	tanggung jawab kepada entitas
	Message merupakan simbol untuk
	mengirim pesan antar kelas

diu diu diu diu diu dit



2.2.8 Use Case Diagram

Berdasarkan penelitian dari Isa & Hartawan (2017), fungsi dari *use case diagram* yaitu untuk menjelaskan sebuah aplikasi dari sudut pandang pengguna. Fungsionalitas pada sistem yang menggambarkan interaksi antara sistem dengan aktor lebih dipentingkan dalam perancangan *use case diagram*.

Tabel 2.4 Simbol Use Case

	Gamba	r	Keterangan
6		S 6	UseCase menggambarkan fungsionalitas yang disediakan sistem sebagai unit yang
Ь			bertugas untuk menukar pesan antara actor dengan unit yang dinyatakan dengan menggunakan kata kerja
6	uis	uib	Aktor adalah orang atau sistem lain yang mengaktifkan fungsi dari target sistem

115	ا كان ما كان ما	Asosiasi antara actor dengan use case
UU		yang mengindikasikan siapa atau apa
		yang meminta interaksi secara langsung
luib.		dan bukan mengindikasikan data
		Asosiasi antara actor dengan use case
		yang mendindikasikan bila actor
UD		berinteraksi secara pasif dengan sistem
		Include berfungsi untuk memanggil
.1		sebuah fungsi program tertentu
(OID	<< include >>	
		Extends merupakan perluasan dari use
ub	<< extends >>	case lain jika kondisi atau syarat
		terpenuhi

2.2.9 Entity Relationship Diagram (ERD)

Menurut Sari & Muhartini (2017), *entity relationship diagram* berfungsi untuk membuat suatu model sistem yang akan dikembangkan dalam perancangan basis data. Beberapa komponen yang ada dalam ERD akan dijelaskan pada Tabel 2.5.



Tabel 2.5 Simbol	Entity	Relationship	Diagram
------------------	--------	--------------	---------

	Gambar	Keterangan
Ь	uib Julb	Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai
Ь		Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berbeda
<u>ь</u>		Atribut berfungsi mendeskripsikan karakter entitas
L L		Garis sebagai penghubung antara relasi dengan entitas atau relasi dengan entitas dengan atribut.

2.2.10 Basis Data

Menurut Hendini (2016), pengertian dari basis data adalah kumpulan dari beberapa item data yang saling berkaitan dan disusun berdasarkan sebuah skema tertentu dimana nanti bisa digunakan kembali dengan gampang dan cepat. Jenjangan data yang terdapat pada pembuatan basis data adalah sebagai berikut:



1. Characters diartikan sebagai karakter huruf, karakter angka dan karakter special yang membantuk suatu field. 2. Field diartikan sebagai kumpulan dari karakter yang membentuk suatu arti kata. 3. Record diartikan sebagai kumpulan dari field yang membentuk sebuah unit data tunggal tertentu. 4. File diartikan sebagai kumpulan dari berbagai record yang memiliki kesatuan data yang sama. Database diartikan sebagai kumpulan dari file yang membentuk suatu basis 5. data. database file record Field data/item characters Gambar 2.2 Jenjang Data

2.3 Aplikasi yang Digunakan

2.3.1 PHP

Menurut Isa & Hartawan (2017), PHP merupakan penyatuan dari bahasa pemrograman web *server side* dengan HTML agar dapat menghasilkan sebuah website yang dinamis. Beberapa fitur dari PHP berupa pertama penggunaan bisa pada *multiplatform* dan kedua dapat digunakan di berbagai *web server* dan ketiga PHP tidak memerlukan kompilasi. Beberapa kelebihan PHP berupa sifatnya yang *open source* sehingga memungkinkan pencarian solusi yang mudah dengan bantuan dari komunitas.

2.3.2 MySQL (My Structure Query Language)

Berdasarkan penelitian dari Isa & Hartawan (2017), MySQL merupakan program perancangan database yang memiliki sifat *open source*. Beberapa kelebihan MySQL berupa struktur penggunaan bahasa SQL yang teratur dan terstandarisasi bagi banyak program untuk mengakses database. Disisi lain, MySQL memiliki fitur-fitur berupa kemudahannya untuk memasang, memakai dan menjalankan MySQL pada banyak sistem operasi serta sifatnya yang *open source* sehingga berbagai dokumentasi solusi permasalahan dapat dicari dengan mudah menggunakan forum terbuka dengan bantuan dari komunitas.

2.3.3 Javascript

Bedasarkan penelitian dari Ruhimat, Hernawati, & Siswanto (2017), Javascript disebutkan sebagai suatu bahasa pemrograman yang memiliki fungsi dalam

membuat tampilan website pada browser dapat menjadi indah dan interaktif sehingga dapat memberikan suatu fungsionalitas ke halaman web yang diakses. Javascript adalah bahasa scripting yang disajikan selaras dengan dokumen HTML sehingga dapat dijalankan pada browser klien. Javascript adalah suatu bahasa dari hasil modifikasi bahasa C++ yang disertai dengan pola penulisan yang lebih sederhana. Bahasa Javascript memiliki fleksibilitas yang tinggi dalam penambahan komponen baru dalam Javascript sehingga dapat sesuai dengan arus perkembangan teknologi. Hal ini dikarenakan sifat dinamis yang dimiliki oleh Javascript. Keunggulan lainnya berupa Javascript cukup memerlukan interpreter dan tidak memerlukan proses kompilasi.

2.3.4 Ajax

Berdasarkan penelitian dari Ruhimat et al., (2017) Asynchronous Javascript and XML merupakan kepanjangan dari AJAX. Pada dasarnya AJAX menggunakan XMLHttpRequest Object Javascript untuk request ke server secara asynchronous yang memiliki arti tanpa melakukan refresh halaman website dengan syarat membutuhkan Javascript di enable pada browser. Sedangkan menurut Anisya & Wandyra (2016), AJAX merupakan suatu teknik pemrograman berbasis web yang menciptakan suatu aplikasi web yang interaktif dengan tujuan untuk memindahkan sebagian besar interaksi ke komputer web pengguna. Selain itu, fitur yang dimiliki AJAX berupa loading halaman web dapat dilakukan tanpa refresh dan dapat menvalidasi form sehingga dapat melakukan pertukaran data di belakang layer yang dapat meningkatkan kecepatan dan usability. Selain itu AJAX juga memiliki keunggulan dalam mengurangi

beban jaringan dan *hit server* dan dapat meningkatkan kecepatan serta meningkatkan *user experience* pengguna.

2.3.5 JQuery

Berdasarkan penelitian dari Ruhimat et al., (2017) interaksi antara HTML dengan Javascript dapat diatur oleh sebuah *framework* Javascript berupa JQuery. JQuery merupakan *library* Javascript yang bersifat *open source*. Fitur-fitur yang dimiliki oleh JQuery berupa terdapat pembuatan animasi, peluncuran efek-efek, adanya penguraian JSON dan events serta dapat memanipulasi DOM yang didasarkan pada selektor CSS yang menggunakan atribut dan nama dari elemen yang digunakan pada HTML yakni Id dan Class yang ditujukan sebagai salah satu kriteria seleksi dari simpul DOM. JQuery memiliki beberapa kelebihan berupa pertama terdapatnya suatu pemisahan antara Javascript dan HTML sehingga *coding* yang dihasilkan lebih jelas dan singkat, kedua mengatasi masalah kecocokan antara satu *browser* dengan *browser* lainnya sehingga masalah inkonsistensi antar *browser* dapat diatasi dan ketiga adanya *plugin* penambahan berbagai *function* yang tidak ada pada distribusi JQuery standar.

2.3.6 XAMPP

Berdasarkan penelitian dari Ruhimat et al., (2017) pengertian dari Xampp merupakan suatu paket perangkat lunak server web yang terdiri atas program Apache, PHP, MySQL dan phpMyAdmin. Para web developer dapat dengan mudah mengembangkan dan merancang website di komputer lokal dengan menggunakan

XAMPP sehingga pada proses perancangan website akan lebih cepat dan lebih aman dibandingkan dengan proses perancangan melewati *online server*. Penggunaan webserver XAMPP juga dapat memudahkan penggunaan pelayanan tampilan halaman website yang dinamis.

2.3.7 Sublime Text

Berdasarkan penelitian dari Soelistio, Wibowo, & Permana (2015) pengertian dari Sublime Text merupakan sebuah *text editor* yang digunakan untuk melakukan pengolahan *plugin*, kode pemrograman dan *markup* serta digunakan dalam pengetikan prosa normal beserta penulisan artikel. Beberapa keunggulan Sublime Text terdapat pada kuantitas dan kualitas fiturnya yakni terdapat duplikasi kusor banyak, blok multi tempat dan pengolahan *split* serta bersifat *multi platform* sehingga bisa digunakan pada berbagai sistem operasi.

2.3.8 Larayel

Menurut Fernando, Tanaamah, & WIjaya (2017), Laravel merupakan sebuah framework PHP yang fokus pada fleksibilitas dan kesederhanaan pada desainnya. Keunggulan dari Laravel terletak pada beberapa point berikut yaitu adanya Eloquent ORM yang menyediakan suatu metode internal agar dapat mengatasi masalah relasi antar objek pada database, selanjutnya ada reverse routing yang mendefinisikan hubungan antara link dengan route sehingga jika terjadi perubahan pada route akan tersambung dengan link yang relevan secara otomatis, berikutnya ada restful

controllers yang menyediakan opsi untuk melakukan pemisahana antara logika dalam melayani HTTP GET dengan permintaan POST, dan adanya *auto loading class* yang memberikan fungsi *loading* secara otomatis untuk berbagai *class* PHP tanpa dibutuhkan pemeriksaan secara manual terhadap jalur masuknnya guna untuk mencegah terjadinya *loading* yang tidak diperlukan.

