BAB II

KAJIAN PUSTAKA

A. Deskripsi Teori

1. Pengertian Pengembangan Sistem

Pengembangan sistem didefinisikan sebagai aktivitas untuk menghasilkan sistem informasi berbasis komputer untuk menyelesaikan persoalan organisasi atau memanfaatkan kesempatan yang timbul (Jogiyanto H.M, 2010). Menurut Bin Ladjamudin (2005) mengatakan bahwa pengembangan sistem adalah menyusun dan membuat suatu sistem yang baru untuk menggantikan sistem yang lama. Pengemabngan sistem bertujuan untuk memenuhi kebutuhan pengguna sistem, memberikan gambaran kepada para pengembang dan ahli-ahli teknik dalam membangun suatu sistem.

2. Pengertian Sistem Informasi

Sistem dapat didefinisikan secara sederhana sebagai gabungan dari beberapa elemen yang saling berhubungan membentuk satu kesatuan. Dalam mendefinisikan sistem terdapat dua kelompok pendekatan yaitu pendekatan pada prosedur dimana setiap sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama, untuk melakukan suatu kegiatan atau untuk tujuan tertentu. Pendekatan pada komponen-komponen atau elemen-elemen, pendekatan pada komponen dianggap lebih mudah dalam mempelajari sistem untuk tujuan dan perancangan sistem (Jogiyanto, 2014). Menurut Syahrina (2015) sistem merupakan gabungan dari beberapa elemen fungsi yang semuanya saling berinteraksi untuk melakukan tugas tertentu (Syahrina Ramadhina, 2015). Sistem komputer yang diciptakan merupakan hasil

dari tujuan sistem fisik. Sistem fisik merupakan kumpulan elemen-elemen/unsur-unsur yang saling berinteraksi satu sama lain secara fisik (Ondi Saondi, 2014).

Sebuah sistem bisa terbentuk tidak terlepas dari informasi yang berkulitas dan tepat. Hal yang paling penting dalam menghasilkan informasi yang berkualitas adalah pengolahan data secara baik dan benar sehingga menjadi sebuah informasi yang bisa digunakan untuk menentukan keputusan yang dapat dirasakan bersama pada saat mendatang (Edhy Sutanta, 2004). Menurut Jogiyanto(1999) ada tiga hal yang sangat penting dalam menentukan kualitas sebuah informasi yaitu (Dani Eko, 2014):

a. Akurat, berarti informasi harus sesuai dengan keadaan di lapangan dan terhindar dari unsur-unsur kesalahan.
b. Tepat waktu, informasi yang disampaikan kepada penerima harus tepat waktu sesuai dengan keadaan di lapangan.
c. Relevan, informasi harus memiliki manfaat positif kepada penerima dan bisa dijadikan sebagai acuan dalam membuat keputusan.

Sistem dan informasi saling melengkapi dalam pengembangan sebuah sistem informasi. Budi Sutedjo (2002) mengatakan bahwa sistem informasi adalah aliran informasi yang diciptakan dan dibentuk dari gabungan elemen-elemen yang terdiri dari orang, prosedur, perangkat lunak, perangkat keras dan basis data yang saling berinteraksi secara sistematis dan teratur untuk mendukung pembuatan keputusan (Viviliana Siang, dkk, 2012). Pendapat lain dari Leitch Davis (1983) bahwa sistem informasi adalah sistem yang berada didalam suatu organisasi yang mempertemukan kebutuhan pengolahan data transaksi harian dan bersifat manajerial dan kegiatan strategis yang diperlukan oleh pihak tertentu (dalam Minarni & Fazril Hadi Saputra, 2011). Blok bangunan (*building-block*) yang terdiri

dari komponen *input*, komponen *input*, komponen model, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol tergabung menjadi satu kesatuan yang disebut dengan sistem informasi (Dessy Irmawati & Yuniar Indrihapsari, 2014).

3. Sistem Informasi Perpustakaan

a. Pengertian

Menurut Lutfian (2009), untuk menjalankan administrasi perpustakaan dan operasional perpustakaan secara sistematis dibutuhkan sistem yang dapat bekerja secara efektif yang berguna bagi manajemen perpustakaan dalam menyusun dan menghasilkan laporan, sistem tersebut adalah sistem informasi perpustakaan. (Lutfian, 2009). Pelayanan perpustakaan seperti transaksi peminjaman dan pengembalian maupun operasi yang bersifat manajerial akan dijalankan dan didukung oleh organisasi pelayanan publik yang disebut dengan sistem informasi perpustakaan (Beiling Siregar, 2007).

b. Konsep

Konsep dasar dari perpustakaan yang memanfaatkan teknologi informasi merupakan alat atau media yang digunakan untuk mengolah data, menyusun menghasilkan, menyimpan, memanipulasi data dalam menghasilkan informasi yang berkualitas (Drs. Hartono, 2017). Sistem informasi perpustakaan menghubungkan data dengan menggunakan *relational database*. *Relational database* akan mengaitkan pangkalan data apabila terjadi transaksi, seperti peminjaman buku, pengembalian buku dan perpanjang masa pinjam buku. Kebanyakan sistem informasi perpustakaan yang dibuat memisahkan fungsi ke dalam program tersendiri yang disebut modul. Sistem informasi perpustakaan

harus memiliki standar minimal modul yang dimiliki yaitu katalogisasi, sirkulasi dan *Online Public Access Catalog* (OPAC) (Drs. Hartono, 2017).

c. Karakteristik Sistem Informasi Perpustakaan

Perpustakaan yang memanfaatkan teknologi informasi akan memudahkan pekerjaan pustakawan dan layanan perpustakaan. Pekerjaan dan layanan perpustakaan yang dimudahkan diantaranya (Drs. Hartono, 2017).

1) Memudahkan dalam membuat katalog

Rangkaian kegiatan dalam membuat katalog buku secara manual banyak membuang waktu dan tenaga. Penerapan komputer akan dapat menghemat segalanya. Proses pembuatan katalog akan lebih cepat, mudah dan efisien.

2) Memudahkan dalam layanan sirkulasi

Dengan komputer pkerjaan peminjaman, pengembalian dan perpanjangan buku dapat dilakukan dengan cepat dan mudah yaitu dengan memilih buku yang akan di kembalikan dan pilih opsi kembalikan. Kegiatan sirkulasi bisa dilakukan dengan mandiri oleh peminjam tanpa melibatkan pustakawan. Transaksi ini dinamakan dengan " *self service* ".

3) Memudahkan dalam penelusuran melalui katalog

Peminjam atau pustakawan dapat menelusuri buku dengan menginputkan kata kunci seperti judul buku atau pengarang.

4) Komponen Sistem Informasi Perpustakaan

a) Pengguna (*users*)

Unsur utama dalam sebuah sistem informasi perpustakaan adalah pengguna. Pengguna yang nantinya yang menentukan sistem seperti apa yang akan dibuat. Pengguna disini meliputi kepala perpustakaan, pustakawan dan staf perpustakaan. Konsultasi dengan pengguna sebelum sistem dibangun sangat

penting untuk menentukan kebutuhan-kebutuhan mereka. Sistem informasi yang dikatakan baik apabila bisa memenuhi kebutuhan pengguna.

b) Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan adalah komputer. Komputer adalah mesin yang dapat mengolah data menjadi informasi secara cepat dan tepat. Kemampuan komputer sangat tergantung pada manusia yang mengoperasikannya dan software yang digunakan. Komputer yang nantiinya akan menjalankan sistem informasi perpustakaan.

c) Network/Jaringan

Jaringan komputer digunakan untuk menghubungan komputer satu dengan yang lain agar bisa berbagi data dan lebih mudah dalam proses pengelolaan datanya. Hal yang perlu diperhatikan dalam membangun jaringan komputer antara lain (a) jumlah komputer, (b) lokasi dari hardware, (c) protokol komunikasi yang digunakan.

d) Data

Data tersusun mulai dari bits, bytes, fields, records, file dan database. Kemudian sistem infromasi perpustakaan menerima masukan data dan instruksi, mengolah data tersebut sesuai instruksi, dan mengeluarkan hasilnya

4. Rekayasa Perangkat Lunak Berbasis Web

Rosa A.S & M. Shalahudin (2014) mengatakan kebutuhan, model, desain dan penggunaan adalah dokumentasi perangkat lunak yang berhubungan langsung dengan program komputer yang menghasilkan sebuah perangkkat lunak (Rosa A. S & M. Shalahudin, 2014). Sedangkan rekayasa perangkat lunak merupakan suatu kumpulan berbagai metode dan perintah-perintah yang digunakan oleh pengembang perangkat lunak komputer (Pressman, 2012). Rekayasa perangkat

lunak memiliki peran penting dalam hal ini karena dengan kumpulan metode-metode yang ada memudahkan profesional membangun sistem/perangkat lunak secara sistematis dalam jangka waktu tertentu.

Perangkat lunak yang dikembangakan dalam penelitian ini adalah perangkat lunak berbasis *web*. Perangkat lunak berbasis *web* merupakan sekumpulan file *hypertext* yang saling terhubung untuk menunjukan informasi-informasi melalui grafis-grafis yang sifatnya terbatas (Pressman, 2012). Semakin berkembangnya zaman, saat ini perangkat lunak berbasis *web* sudah bisa diintegrasikan dengan basis data sehingga dapat mengolah informasi berukuran besarAdapun sifat-sifat yang sering ditemui dalam sebuah perangkat lunak berbasis *web* diantaranya sebagai berikut.

a. Kepadatan Jaringan

Aplikasi-aplikasi *web* pada umumnya berada pada suatu jaringan komputer dan harus bisa melayani kebutuhan komunitas klien yang beragam.

b. Keserempakan

Sejumlah besar pengguna mungkin akan mengakses aplikasi-aplikasi web secara serempak.

c. Kinerja

Jika seseorang pengguna aplikasi-aplikasi web harus menunggu lama untuk mengakses web, pengguna bisa memutuskan pergi ke tempat lain.

d. Digerakan oleh Data

Fungsi utama dari banyak perangkat lunak berbasis *web* adalah menggunakan *hypermedia* untuk menyajikan isi teks, suara, grafis,dan video kepada pengguna.

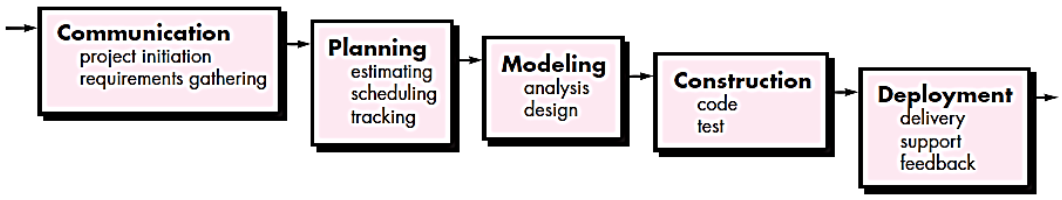
e. Kemanan

Perangkat lunak berbasis *web* tersedia yang melalui akses jaringan komputer maka kemanan juga sangat penting. Tindakan pencegahan terhadap penyusupan dan pengambilan data secara ilegal oleh pihak yang tidak bertanggung jawab perlu diterapkan melalui infrastuktur jaringan komputer.

5. Model Pengembangan

a. Pengembangan Perangkat Lunak (*Software Enginering*)

Menurut Pressman (2010), untuk mengembangkan sebuah perangkat lunak (*software enginering*) dibutuhkan beberapa metode dan perkakas pendukung seperti *compiler* bahasa pemrograman dalam menghasilkan perangkat lunak komputer berkualitas tinggi (Pressman, 2010). Ada tahapan-tahapan yang harus dijalankan salah satunya adalah *Software Development Life Cycle* (SDLC). SDLC merupakan tahapan proses dalam mengembangkan perangkat lunak dengan menggunakan metodologi yang telah digunakan dalam pengembangan sistem-sistem perangkat lunak sebelumnya (Rosa A. S & M. Shalahudin, 2014). SDLC memiliki beberapa varian model yang salah satunya adalah model air terjun (*waterfall)*. Alur tahapan model *waterfall* dapat dilihat pada gambar 1 berikut.



Gambar 1. *Waterfall* Model (Pressman, 2010)

Alur pengembangan model *Waterfall* sistematis dan beruurutan (Pressman, 2010). Tahapan model air terjun (*The Waterfall Model*) adalah sebagai berikut:

1) Komunikasi (*Communication*)

Komunikasi bertujuan untuk mengumpulkan informasi kebutuhan dari pengguna yang harus dibuat dalam aplikasi. Hasil dari informasi yang didapatkan dari pengguna akan membantu menentukan kebutuhan apa saja yang diperlukan yang akan diterapkan pada perangkat lunak.

2) Perencanaan (*Planning*)

Perencanaan bertujuan untuk menentukan tugas-tugas yang harus dilakukan, lama waktu pengerjaan, dan *progress* yang bisa dicapai setiap waktunya.

3) Pemodelan (*Modelling*)

Pemodelan bertujuan untuk menentukan desain *User Interface* dan *User Experience* sesuai dengan analisis kebutuhan.

4) Konstruksi (*Construction*)

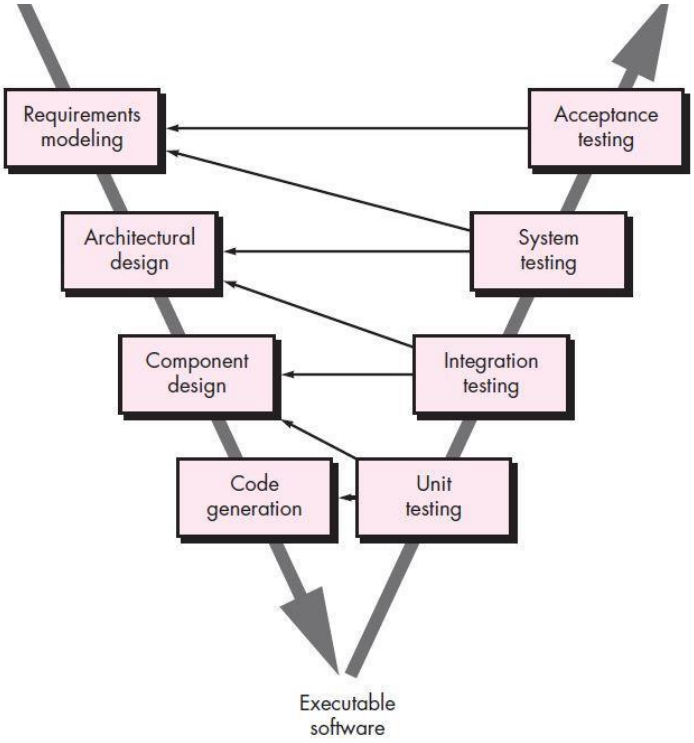
Tahapan ini membangun perangkat lunak berdasarkan analisis kebutuhan pengguna dan pemodelan. Kemudaian akan digabungkan dalam satu perangkat lunak dengan menggunakan code program. Pengujian juga akan dilakukan dalam tahap ini mulai dari pengujian dari pengembang dan para ahli untuk menguji perangkat lunak layak atau tidak sebelum diserahkan kepada pengguna.

5) Penyerahan perangkat lunak ke pelanggan/pengguna (*Deployment*)

Perangkat lunak akan di implementasikan dan disajikan kepada pelanggan/pengguna untuk selanjutnya akan dilakukan penilaian oleh pengguna untuk kemudian hasilnya dijadikan bahan perbaikan oleh pengembang.

Penelitian ini menggunakan model pengembangan air terjun variasi model-V. Tahapan ini memudahkan pengembang dalam mengembangkan aplikasi bersifat *Object Iriented Programming* (OOP). Dalam pemodelan model-V proses pengujian

lebih kompleks karena dibagi menjadi beberapa bagian yang lebih detail (Windi Eka Y & dkk, 2013). Alur tahapan model-V dapat dilihat pada gambar 2 berikut:



Gambar 2. *Waterfall* Model-V (Pressman, 2010)

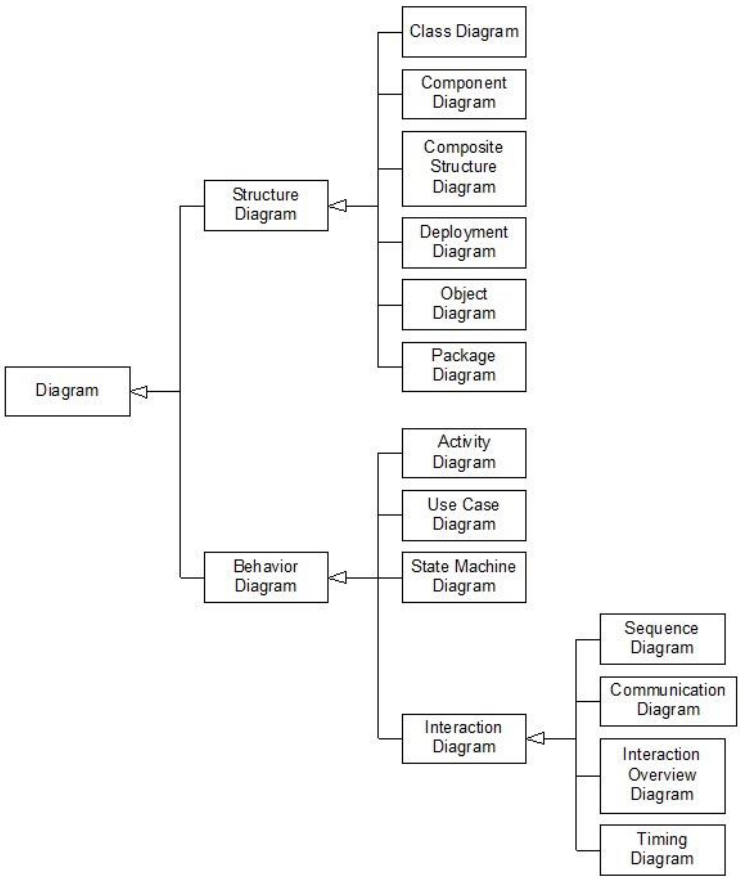
Model-V memilki kelebihan yaitu 1) proses pengembangan produk lebih teliti karena setiap langkah harus diverifikasi sebelum melangkah maju, 2) setiap tahap pengembangan dalam model-V terhubung satu sama lainnya, dan 3) tahapan model-V tidak dirancang dalam sumbu linear sehingga tahapan bisa kembali ke atas setelah pengkodean selesai (S.Balaji & Dr. M. Sundararajan, 2012).

b. Desain *Unfied Modeling Language* (UML)

Standar bahasa yang biasanya digunakan oleh profesional membangun perangkat lunak adalah UML. UML digunakan untuk menyusun kebutuhan sistem mulai dari perancangan, desain, arsitektur pemrograman berorientasi objek (Rosa

A. S & M. Shalahudin, 2014). Penggambaran dan pemahaman terhadap perangkat lunak yang akan dibuat dipermudah dengan adanya UML. Pendapat lain mengatakan bahwa UML adalah bahasa pemodelan (*modeling language*) yang kosakata dan aturannya difokuskan pada representasi dari suatu sistem secara konseptual dan fisik (Irma Yunita & Joni Devitra, 2017).

Diagram UML memiliki 13 macam diagram yang kemudian dikelompokan menjadi 3 kategori diagram. Pembagian diagram UML dapat dilihat pada gambar 3 berikut.



Gambar 3. Diagram UML

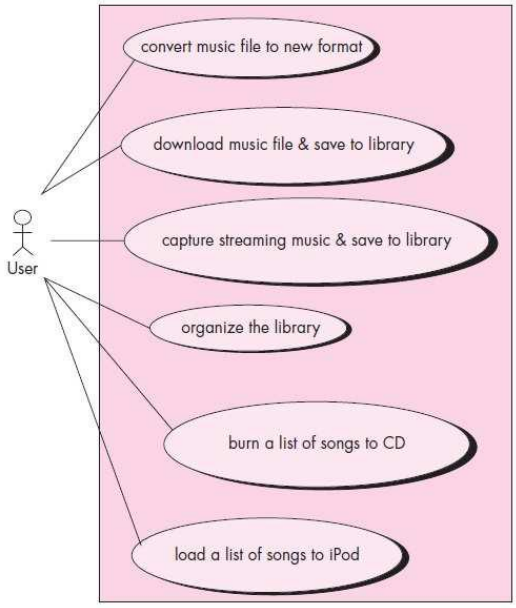
Dijelaskan dalam Rosa A. S & M. Shalahudin (2014) bahwa 1) *structure diagram* yaitu sekumpulan diagram yang memiliki kegunaan untuk menjelaskan bagaimana struktur sebuah sistem dimodelkan, 2) *behaviour diagram* yaitu sekumpulan diagram yang digunakan untuk menggambarkan kerja sistem, dan 3) *interaction diagram* yaitu sekumpulan diagram yang menggambarkan interkasi sistem dengan sistem lain.

Diagram UML yang digunakan dalam penelitian ini yaitu *use case diagram, activity diagram,* dan *sequence diagram.* Dibawah ini berikut penjelasan diagram UML yang digunakan dalam penelitian (Indra Griha. I & George Pri. H, 2017) :

1) *Use Case Diagram*

Use Case diagram digunakan untuk menggambarkan hubungan aktor (pengguna) dengan sistem. Use case diagram menunjukan fungsionalitas yang dapat dijalankan oleh pengguna terhadap sistem yang dibuat.

Pressman (2010) mengatakan bahwa *Use Case diagram* dapat membantu pengembang dalam menentukan fungsi dan fitur dari sebuah perangkat lunak. *Use case* menjelaskan bagaimana pengguna berinteraksi dengan sistem dengan cara mendefinisikan langkah-langkah yang diperlukan untuk mencapai tujuan tertentu.

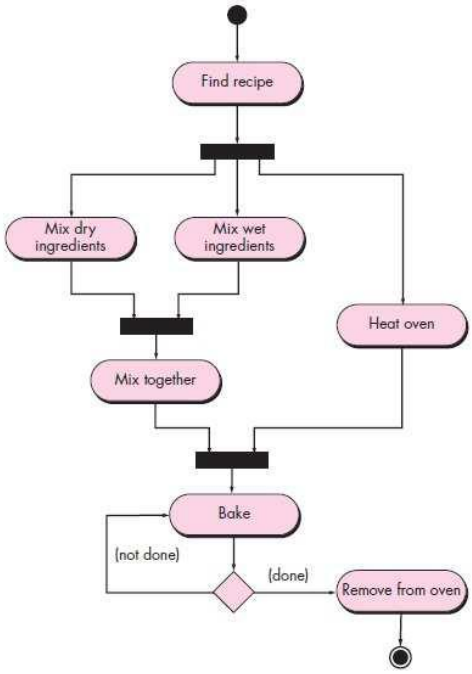


Gambar 4. Contoh *Use Case Diagram* (Pressman, 2010)

2) *Activity Diagram*

Activity diagram menggambarkan rangkaian aliran dari sistem, yang berhubungan langsung dengan alur kerja sistem. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari suatu aktifitas ke aktifitas yang lainnya. *Activity diagram* juga digunakan untuk menggambarkan interaksi antara beberapa *use case*. Aktivitas berjalannya sebuah sistem akan digambarkan pada diagram aktivitas (*activity diagram*). Diagram aktivitas menjelaskan bagaimana sebuah sistem harus berjalan dan apa yang akan dihasilkan dari aksi dari sistem itu sendiri (Rosa dan M. Shalahuddin, 2013). Komponen-komponen *activity diagram* terdiri dari *status awal* yang digambarkan dengan lingkaran berwarna hitam, *aktivitas* yang digambarkan dengan bulat persegi panjang, *decision* yang digambarkan dengan belah ketupat, *join* digambarkan dengan garis, *status akhir* digambarkan dengan lingkaran berwarna hitam dengan *bold* berwarna putih.

Status awal menggambarkan awal proses activity dimulai. Kemudian akan dihubungkan dengan *aktivitas* melalui *join*. *Activity diagram* diakhiri dengan *status akhir*.

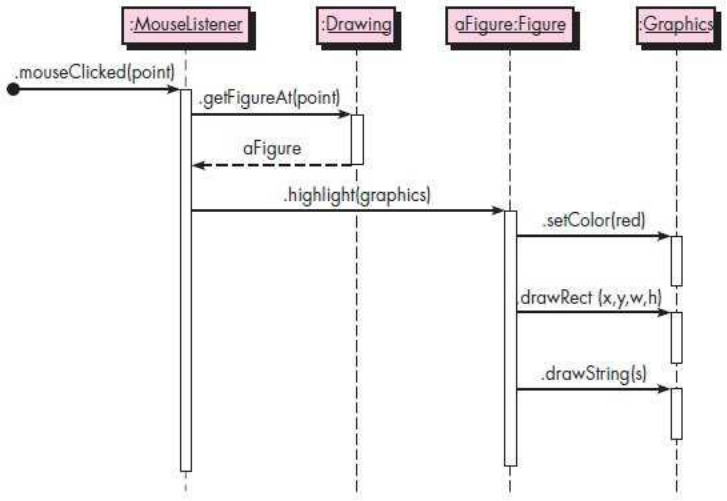


Gambar 5. Contoh *Activity Diagram* (Pressman, 2010)

3) *Sequence Diagram*

Sequence diagram menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukan rangkaian pesan yang dikirim antara objek dan interaksi antara objek yang terjadi pada titik tertentu dalam eksekusi sistem.

Sequence diagram menampilkan metode panggilan menggunakan panah horizontal dari pelaku ke target pelaku, diberi label dengan nama metode dan termasuk parameter, jenis dan jenis timbal balik.



Gambar 6. Contoh *Sequence Diagram* (Pressman, 2010)

c. Kerangka Kerja (*Framework*) *Codeigniter*

Framework adalah kerangka kerja yang tersusun dari kumpulan script terutama *class* dan *function* yang dapat membantu *developer/programer* dalam menangani pemrograman seperti koneksi ke *database,* pemanggilan *variable,* dan *file*(Dewi Rosmalia, dkk, 2011). Menurut Basuki (2010) bahwa *framework* diartikan sebagai kumpulan potongan-potongan program yang disusun dan diorganisasikan

sedemikian rupa, sehingga memudahkan dalam membuat sebuah aplikasi tanpa harus membuat semua kodenya dari awal dalam (D Tri Octafian, 2015). Saat ini ada banyak *framework,* diantaranya: *Zend, Cake PHP, Trax, Symfony, Bootstrap, Codeigniter*, dan sebagainya.

Codeiginter merupakan salah satu dari sekian banyak *framework* PHP yang ada. *Codeigniter* dikembangkan oleh Rick Ellis (Aris Rakhmadi, Eko Listiyanto, 2008). *Codeigniter* adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi berbasis website tanpa harus membuat semua kodenya dari awal (D Tri Oktafian, 2015). *Codeigniter* menyediakan banyak *library* untuk mengerjakan tugas-tugas yang umumnya ada pada aplikasi berbasis website. *Codeigniter* memakai sistem Model View Control (MVC) yang komponen utamanya antara lain (Ibnu Daqiqil, 2011):

1) Model

Model berhubungan dengan data dan interaksi ke *database*. Model bertugas untuk mempresentasikan struktur data dari aplikasi yang bisa berupa data maupun data lain, misalnya file teks, file XML maupun webservice. Model berisi *class* dan *fungsi* untuk mengambil, melakukan update dan menghapus data website. Aplikasi *web* biasanya menggunakan *basis data* dalam menyimpan data, maka pada model biasanya akan berhubungan dengan perintah-perintah query SQL.

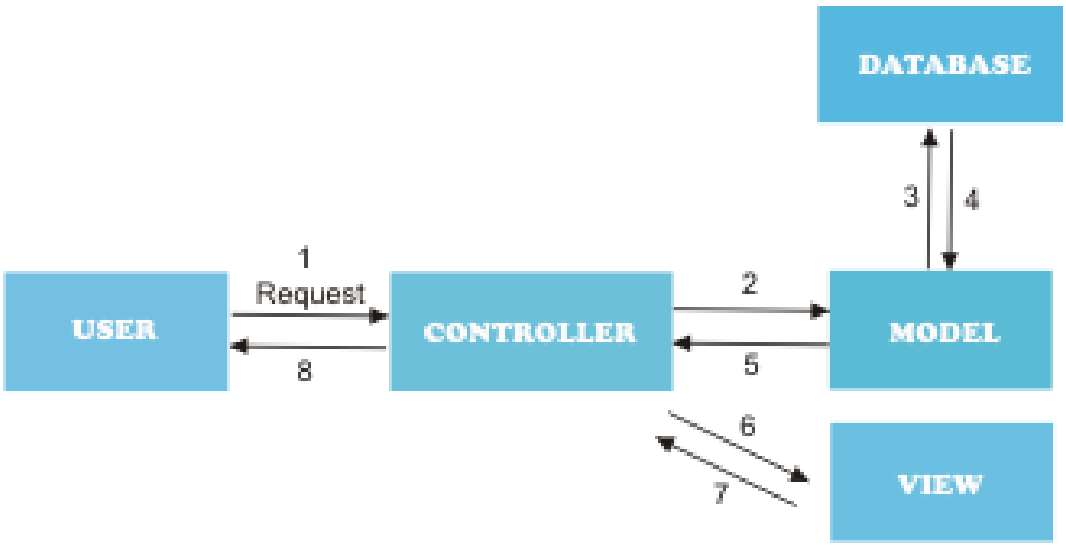
2) View

View berhubungan dengan segala sesuatu yang akan ditampilkan ke *user*. Bisa berupa halaman *web*, rss, javascript dan lain-lain. Di dalam view hanya berisi variabel-variabel yang berisi data yang siaap ditampilkan. View dapat dikatakan sebagai halaman website yang dibuat dengan menggunakan HTML dan bantuan

CSS atau javascript. View dikhususkan untuk menampilkan data-data hasil dari model dan controller.

3) Controller

Controller bertindak sebagai penghubung data dan view. Di dalam controller, terdapat *class-class* dan *fungsi-fungsi* yang memproses permintaan dari view ke struktur data yang ada di dalam model. Tugas controller adalah menyediakan berbagai variabel yang akan ditampilkan di view, memanggil model untuk melakukan akses ke basis data, menyediakan penanganan kesalahan/error, mengerjakan proses logika dari aplikasi serta melakukan validasi atau cek input. Gambaran sederahana tentang MVC yang diterapkan Codeigniter ditunjukan pada gambar 7 berikut.



Gambar 7. Konsep *MVC* (Awan Pribadi Basuki, 2016)

Penjelasan gambar:

- No. 1: user me-request suatu halaman kepada aplikasi dengan melalui *Controller* terlebih dahulu.

- No. 2: tugas *Controller* adalah melakukan pengecekan skrip yang ada di *Model* dan menjalankan perintahnya sesuai permintaan yang ada di *Model*.
- No. 3, No. 4, & No. 5: jika diperlukan *query* ke *database*, maka akan dilakukan oleh *Model* ke *database*. Hasil *query* dan pengolahan data di *Model* dikembalikan ke *Controller*.
- No. 6: untuk menampilkan data yang dihasilkan *Model*, *Controller* akan memanggil *View*.
- No.7 & No. 8: hasil dari penampilan data di *View*, akan dikembalikan lagi ke *User*.

Penerapan MVC pada *Codeigniter* tidak seketat seperti di *framework* lainnya karena kita bisa membuat seluruh kode di *Controller* atau hanya memanfaatkan *Controller* dan *View* saja. Tapi itu kurang tepat karena *framework* dibuat agar kode program lebih terorganisir dan rapi. Tidak bercampur antara kode yang memproses data dengan kode yang memproses penampilan data. Codeigniter memiliki beberapa keuntungan diantaranya gratis, menggunakan PHP 4, dan berukuran kecil (D Tri Oktafian, 2015).

6. Kualitas Perangkat Lunak

Untuk menghasilkan perangkat lunak sesuai dengan yang diharapkan, kualitas perangkat lunak harus terjamin. Pressman (2010) mengatakan bahwa jaminan kualitas perangkat lunak merupakan aktivitas yang penting yang harus diterapkan dalam pengembangan perangkat lunak, sehingga pengujian perangkat lunak yang mengacu pada standar tertentu harus dilakukan. Pengujian dilakukan untuk setiap modul dan dilanjutkan dengan pengujian untuk semua modul yang telah dirangkai (Didik Hariyanto, 2008).

Standar internasional pengujian perangkat lunak yang bisa digunakan adalah *ISO/IEC 25010*. Alasan menggunakan Standar *ISO/IEC 25010* adalah untuk mengikuti perkembangan zaman dan agar lebih relevan dengan teknologi yang dikembangkan saat ini (Jose P.M & David. M, 2014). Faktor kualitas *ISO/IEC 25010* menurut ISO (2011) meliputi delapan karakteristik kualitas yaitu, (*functional suitability, reliability, performance efficiency, usability, security, compatibility, maintainability and portability*). Delapan karakteristik dijabarkan secara mendetail pada gambar 8 berikut:

Gambar 8. Model Kualitas Produk ISO/EC 25010

Tahap paling awal dalam pengujian perangkat lunak adalah menguji bagian yang terlihat oleh pengguna yaitu apakah fungsi-fungsi dan antarmuka yang sudah dibangun bisa berjalan sesuai dengan yang diharapkan (Pressman, 2012). Menurut olsina dan rekan-rekan dalam (Pressman, 2012) standar kualitas web dinilai dari lima aspek yaitu fungsionalitas, kemudahan pengguna, keandalan, efisiensi, dan kemudahan pemeliharaan (Heru Steiawan & Handaru Jati, 2017). Pengujian sistem informasi perpustakaan dilakukan pada 4 aspek yaitu fungsionalitas, kemudahan pengguna, efisiensi, dan kemudahan pemeliharaan. Jika disesuaikan dengan karakteristik dalam *ISO/EC 25010* yaitu *functional suitability*, *reliability*, *maintainability*, *usability* . Pengujian *performance efficiency*

tidak dilakukan karena sistem informasi perpustakaan *web* yang dibuat pengembang hanya bersifat lokal dan *offline*.

a. *Functional Suitability*

Pengujian ini menggunakan kuesioner yang berisi fungsi dari sistem informasi yang dijabarkan sesuai dengan analisis kebutuhan fungsional. Pengujian ini biasanya dilakukan oleh responden ahli di bidang sistem informasi. Karakteristik *functional suitability* ini terdiri dari 3 sub karakteristik yaitu sebagai berikut.

1) Functional completeness, yakni sejauh mana fungsi yang ada dalam sistem sesuai dengan apa yang dibutuhkan dan diinginkn oleh pelanggan .

2) Functional correctness, yakni sejauh mana produk atau sistem informasi menyediakan hasil yang benar sesuai dengan kebutuhan.

3) Functional appropriateness, yakni sejauh mana fungsi yang disediakan mampu menyelesaikan tugas dan bisa mencapai tujuan tertentu.

b. *Reliability*

Dalam karakteristik ini sistem akan diuji sejauh mana komponen dapat menjalankan fungsi tertentu dibawah kondisi tertentu dalam waktu yang sudah ditetapkan. Karakteristik *reliability* ini terdiri dari 4 sub karakteristik yaitu sebagai berikut.

1) Maturity, yakni kemampuan perangkat lunak untuk menghindari kegagalan sebagai akibat dari kesalahan dalam perangkat lunak .

2) Availability, yakni sejauh mana waktu yang dibutuhkan oleh perangkat lunak menjalankan fungsi dan akses nya.

3) Fault tolerance, yakni kemampuan perangkat lunak untuk mempertahankan kinerjanya jika terjadi kesalahan perangkat lunak .

4) *Recoverability,* yakni kemampuan perangkat lunak untuk membangun kembali tingkat kinerja ketika terjadi kesalahan/kegagalan sistem, termasuk data dan koneksi jaringan.

Untuk melakukan pengujian ini dibutuhkan sebuah *tools* yaitu WAPT yang dapat mengukur *performance* dan melakukan melakukan *stress testing* (Kundu, 2012).

c. *Maintainability*

Karakteristik ini menguji sejauh mana tingkat efektivitas dan efisiensi perangkat lunak untuk dimodifikasi oleh pengembang. *Maintanability* diperlukan untuk pemeliharaan perangkat lunak dimana setelah dikembangkan dan diimplementasikan terdapat beberapa hal yang perlu diperbaiki berdasarkan hasil ujicoba maupun evaluasi. Suatu perangkat lunak yang baik dikatakan *maintainability* yaitu dapat dengan mudah direvisi apabila diperlukan.

d. *Usability*

Karakteristik ini melihat dan menilai seberapa jauh produk atau sistem dapat digunakan oleh *user* untuk mencapai tujuan yang ingin dicapainya. *User* menjadi subyek utama dalam pengujian ini karena *user* berinteraksi langsung dengan produk atau sistem. Hasil dari pengujian karakteristik ini adalah *user* memberikan *feedback* berupa penilaian kepuasan terhadap produk yang sudah digunakan. *Usability testing* adalah pengujian untuk memastikan keefisienan, keefektifan, dan kepuasan penguna aplikasi (Ben David, 2011). Karakteristik *usability* terdiri dari 7 sub karakteristik yaitu sebagai berikut.

1) *Appropriatness recognizability,* yakni sejauh pengguna dapat mengetahui apakah sistem atau produk sesuai kebutuhan mereka.

2) *Learnability,* yakni sejauh mana produk atau sistem dapat digunakan oleh pengguna untuk mencapai tujuan tertentu yang belajar menggunakan sistem atau produk dengan efisien, efektif, kebebasan dari resiko dan kepuasan dalam konteks tertentu.
3) *Operability,* yakni sejauh mana produk dapat mudah digunakan pengguna tanpa harus berfikir untuk mengoperasikannya.
4) *User error protection,* yakni sejauh mana produk atau sistem memberikan kesan mudah tanpa haru membuat kesalahan.
5) *User interface aesthetics,* yakni sejauh mana antarmuka bisa manarik minat dari pengguna untuk menggunakannya.
6) *Accessibility,* yakni semua kalangan bisa menggunakan produk atau sistem.

Untuk mengetahui bagaiman hasil pengujian *usabilty* pada sistem informasi perpustakaan adalah dengan menggunakan kuesioner. Kuesioner yang dibagikan pada pengguna bertujuan untuk mengetahui *feedback* apa yang dihasilkan dari *user* untuk mengetahui produk atau sistem layak digunakan atau tidak. Kuesioner yang digunakan adalah *USE Questionnarie* oleh Arnold M. Lund(2001).

Bentuk dari *USE Questionnarie* tersaji pada tabel 1 berikut.

Tabel 1. *USE Questionnarie*

USE Questionnaire: Usefulness, Satisfaction, and Ease of use

Based on: Lund, A.M. (2001) *Measuring Usability with the USE Questionnaire.* STC Usability SIG Newsletter, 8:2. [Abstract] | About quest.cgi

Please rate your agreement with these statements.

- Try to respond to all the items.
- For items that are not applicable, use: **NA**
- Make sure these fields are filled in: **System: Email to:**
- Add a comment about an item by clicking on its icon, or add comment fields for all items by clicking on **Comment All**.
- To mail in your results, click on: **Mail Data**

System: **Email to:**
Optionally provide comments and your email address in the box.

Mail Data | Comment All

		1	2	3	4	5	6	7		NA
USEFULNESS										
1. It helps me be more effective.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
2. It helps me be more productive.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
3. It is useful.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
4. It gives me more control over the activities in my life.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
5. It makes the things I want to accomplish easier to get done.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
6. It saves me time when I use it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
7. It meets my needs.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
8. It does everything I would expect it to do.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
EASE OF USE		**1**	**2**	**3**	**4**	**5**	**6**	**7**		**NA**
9. It is easy to use.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
10. It is simple to use.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
11. It is user friendly.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
12. It requires the fewest steps possible to accomplish what I want to do with it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
13. It is flexible.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
14. Using it is effortless.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
15. I can use it without written instructions.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
16. I don't notice any inconsistencies as I use it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
17. Both occasional and regular users would like it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
18. I can recover from mistakes quickly and easily.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
19. I can use it successfully every time.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
EASE OF LEARNING		**1**	**2**	**3**	**4**	**5**	**6**	**7**		**NA**
20. I learned to use it quickly.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
21. I easily remember how to use it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
22. It is easy to learn to use it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
23. I quickly became skillful with it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
SATISFACTION		**1**	**2**	**3**	**4**	**5**	**6**	**7**		**NA**
24. I am satisfied with it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
25. I would recommend it to a friend.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
26. It is fun to use.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
27. It works the way I want it to work.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
28. It is wonderful.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
29. I feel I need to have it.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
30. It is pleasant to use.	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
		1	**2**	**3**	**4**	**5**	**6**	**7**		**NA**

B. Penelitian Yanga Relevan

Beberapa penelitian yang relevan dengan penelitian ini antara lain:

1. Pengembangan Sistem Informasi Perpustakaan Berbasis Web Di SMK Muhammadiyah 1 Yogyakarta oleh Husin Nanda Perwira, Universitas Negeri Yogyakarta. Tujuan dari penelitian ini adalah membuat sistem informasi perpustakaan sekolah dan mengetahui tingkat kelayakan dari sisi *functionality*, *security*, *usability*, *maintainability*, *portability* dan *efficiency*. Hasil yang didapat dalam penelitian ini dapat disimpulkan bahwa, menghasilkan perangkat lunak yang berkualitas baik dari sisi functionality, security, usability, maintainability, portability dan efficiency. (Husin Nanda Perwira, 2015). Kekurangan aplikasi yang disampaikan dalam penelitian ini adalah belum adanya fitur barcode untuk membaca identitas buku dan juga fitur baca buku secara langsung agar *user* dapat mereview buku yang ada di perpustakaan terlebih dahulu.

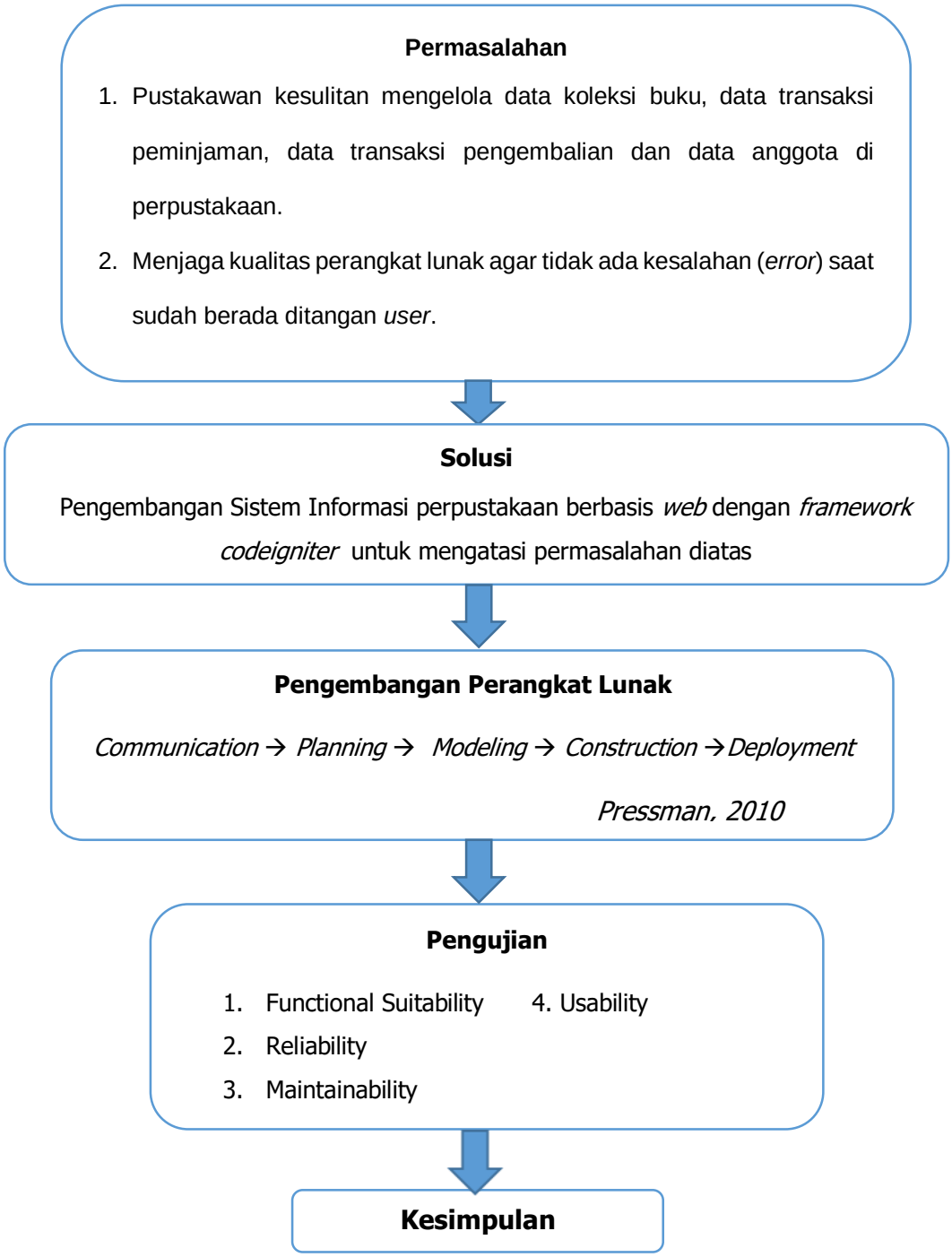
2. Analisis Dan Pengembangan Sistem Informasi Alumni Sekolah Berbasis Web Di SMA 2 Wates oleh Azis Amirulbahar, Universitas Negeri Yogyakarta. Tujuan dari penelitian ini adalah mengembangkan sistem informasi alumni berbasis web di SMA 2 Wates dan mengetahui kualitas sistem informasi yang dikembangkan dari aspek *functionality, reliability, usability, efficiency, maintainbility, portability* (Azis Amirulbahar, 2015). Kekurangan dari aplikasi ini adalah belum adanya fitur forum diskusi online antar alumni.
3. Sistem Informasi Perpustakaan Sekolah Berbasis *Visual Basic* di SMK Muhammadiyah 2 Moyudan oleh Ahi Sholihin Saragih, Universitas Negeri Yogyakarta. Tujuan dari penelitian ini adalah membangun sistem informasi perpustakaan yang dapat mengurangi kesalahan pendataan di perpustakaan SMK Moyudan 2 serta mengurangi tingkat kerumitan pengisian data di

perpustakaan SMK Muhammadiyah 2 Moyudan (Ahi Sholihin Saragih, 2017). Kekurangan aplikasi yang disampaikan adalah belum dapat mengimpor anggota atau buku menggunakan database yang sudah ada. Sistem juga belum bisa menambahkan foto untuk anggota.

C. Kerangka Berfikir

Permasalahan yang terjadi di perpustakaan sekolah SMK Negeri 1 Banjar yaitu siswa kesulitan mencari informasi buku yang tersedia di perpustakaan sekolah. Ini terjadi karena belum adanya informasi terpasang seperti komputer. Sehingga menyulitkan siswa dalam mendapatkan informasi buku yang tersedia di perpustakaan. Untuk mengetahui buku yang tersedia, pustakawan harus membuka buku transaksi peminjaman dan katalog buku.

Permasalahan lainnya adalah pustakawan membutuhkan sistem informasi perpustakaan yang berkualitas sesuai standar perangkat lunak agar fitur dan fungsionalitas dapat berjalan dengan baik sesuai dengan yang diharapkan.



Gambar 9. Kerangka Berfikir

D. Pertanyaan Penelitian

1. Bagaimana mengembangkan sistem informasi perpustakaan berbasis *web* untuk mengatasi kesulitan pustakawan mengelola data perpustakaan sekolah di SMK Negeri 1 Banjar ?
2. Apakah sistem informasi perpustakaan berbasis *web* di SMK Negeri 1 Banjar yang dikembangkan memenuhi aspek *functional suitability* ?
3. Apakah sistem informasi perpustakaan berbasis *web* di SMK Negeri 1 Banjar yang dikembangkan memenuhi aspek *reliability* ?
4. Apakah sistem informasi perpustakaan berbasis *web* di SMK Negeri 1 Banjar yang dikembangkan memenuhi aspek *maintainability* ?
5. Apakah sistem informasi perpustakaan berbasis *web* di SMK Negeri 1 Banjar yang dikembangkan memenuhi aspek *usability* ?