

BAB II

LANDASAN TEORI

2.1 Kajian Pustaka

Pembuatan sistem tes masuk mahasiswa baru berbasis *website* di Universitas Sahid Surakarta ini menggunakan beberapa penelitian yang terkait sebagai dasar yaitu sebagai berikut :

Penelitian dari Dhamayanti (2015) dengan judul sistem pelayanan ujian Universitas Sahid Surakarta membahas tentang membuat sebuah sistem guna mendukung proses pengolahan data sehingga pekerjaan dapat berjalan secara sistematis, efektif dan efisien. Selama ini staf Direktorat Akademik Universitas Sahid Surakarta dalam proses pelayanan ujian terlebih jika data yang diproses jumlahnya cukup banyak dan kompleks atau beragam, masih terdapat kendala dengan penggandaan data. Hal ini disebabkan antara lain karena dalam proses pengolahan data ujian masih menggunakan perangkat lunak microsoft excel dan microsoft word yang tidak ada validasi jika terdapat data ganda. Hal ini sangatlah sulit karena kedua perangkat lunak ini memiliki keterbatasan fitur, serta belum adanya sistem aplikasi yang khusus dalam proses pengolahan data khususnya data pelayanan ujian. Aplikasi yang digunakan dalam pembuatan sistem ini adalah adobe dreamweaver dan *database* yang digunakan adalah MySQL, bahasa pemrograman yang dipakai adalah PHP (*Hypertext Preprocessor*) serta metode pengembangan menggunakan metode *waterfall*, perancangan sistem informasi ini menggunakan model UML (*Unified Modeling Language*.)

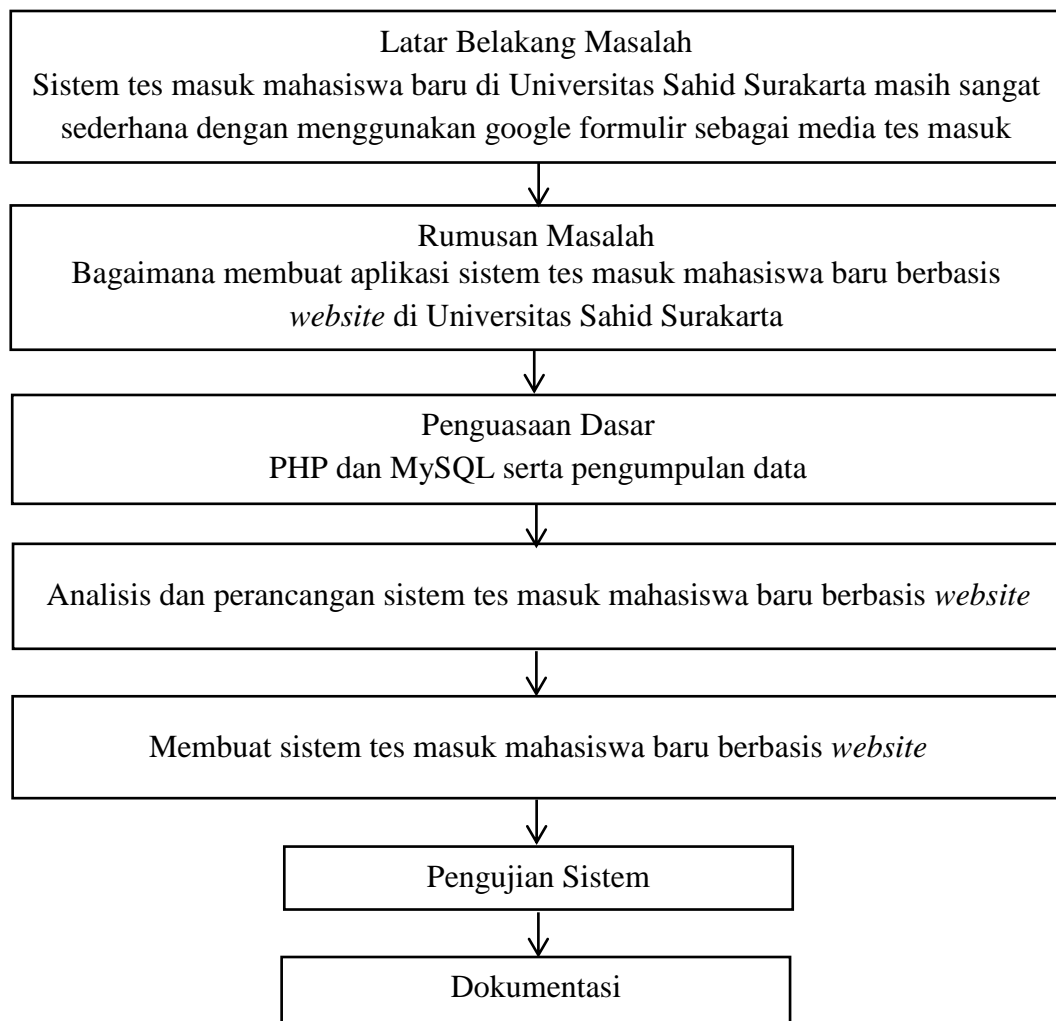
Penelitian dari Safitri (2016) dengan judul sistem pelayanan pendaftaran mahasiswa baru Universitas Sahid Surakarta membahas tentang membuat sebuah sistem guna membantu proses pengelolaan data mahasiswa baru di Universitas Sahid Surakarta. Proses pengelolaan data mahasiswa baru masih menggunakan cara manual sehingga sehingga membutuhkan kecermatan yang tinggi dan waktu yang cukup lama. Kegiatan pendaftaran mahasiswa baru dilakukan dengan 2 (dua) cara yaitu dengan *online* dan mahasiswa mendaftar langsung ke Universitas Sahid Surakarta tapi kenyataanya calon mahasiswa lebih memilih untuk mendaftar

langsung ke Universitas Sahid Surakarta. *Database* yang digunakan adalah MySQL, bahasa pemrograman yang dipakai adalah PHP (*Hypertext Preprocessor*) serta metode pengembangan menggunakan metode *waterfall* atau model sekuensial linier, metode pengujian yang dipakai adalah metode *WebQual*, perancangan sistem informasi ini menggunakan model UML (*Unified Modeling Language*.)

Penelitian dari Febrianto (2016) dengan judul pengembangan sistem ujian *online* berbasis *web* pada mata pelajaran teknik listrik di sekolah menengah kejuruan Yogyakarta membahas tentang sekolah menengah sebagai salah satu lembaga pendidikan yang berintiskan nasional, dimana siswa-siswi melakukan ujian menggunakan metode ujian konvensional. Masalah yang ditemukan saat menggunakan ujian dengan cara konvensional yaitu menggunakan media kertas dan alat tulis sehingga banyak memakan biaya dan waktu, panjangnya waktu pendistribusian soal dari pembuat soal hingga sampai ke siswa dapat beresiko terjadinya kebocoran soal ujian, siswa dan orangtua tidak dapat langsung mengetahui hasil ujian, pengarsipan hasil ujian konvensional memakan banyak tempat, belum terdapat sistem yang menampilkan hasil ujian siswa berdasarkan tingkat kelulusan kompetensinya secara *detail*. Untuk dapat memudahkan para siswa dalam melakukan ujian dan mendapatkan informasi hasil ujian yang cepat dibutlah aplikasi ujian *online*. *Database* yang digunakan adalah MySQL, bahasa pemrograman yang dipakai adalah PHP (*Hypertext Preprocessor*), HTML (*Hypertext Markup Language*), dan CSS (*Cascading Style Sheets*). Serta metode pengembangan menggunakan metode *waterfall* atau model sekuensial linier, metode pengujian yang dipakai adalah metode ISO 9126 memiliki 5 karakteristik yaitu *functionality*, *reliability*, *usability*, *efficiency*, dan *portability*. Perancangan sistem informasi ini menggunakan model UML (*Unified Modeling Language*.)

2.2 Kerangka Pemikiran

Berikut ini adalah tahapan kerangka pemikiran dalam membuat sistem tes masuk mahasiswa baru berbasis *website* di Universitas Sahid Surakarta. Pada Gambar 2.1 berikut.



Gambar 2.1 Kerangka Pemikiran

Penjelasan dari kerangka pemikiran tersebut adalah :

1. Latar Belakang Masalah

Penggunaan teknologi dan informasi untuk penilaian hasil pembelajaran atau untuk menyeleksi pendaftar yang masuk ke instansi pendidikan maka diperlukan sebuah sistem yang akan mengelola serta memberikan laporan secara sistematis, cepat dan efisien. Ujian *online* berbasis *website* merupakan sebuah jawaban yang dihadirkan dalam dunia pendidikan. Proses tes masuk mahasiswa baru di Universitas Sahid Surakarta masih terlalu sederhana, pada tes ujian masuk penerimaan mahasiswa baru tahun 2018 Universitas Sahid Surakarta masih menggunakan google formulir yang antarmuka dan fiturnya sangat sederhana

sehingga terdapat keterbatasan fitur yang bisa dan seharusnya digunakan dalam ujian tes masuk Universitas Sahid Surakarta.

2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan maka rumusan masalahnya adalah bagaimana membuat aplikasi sistem tes masuk mahasiswa baru berbasis *website* di Universitas Sahid Surakarta.

3. Penguasaan dasar PHP dan MySQL

Penguasaan dasar PHP dan MySQL merupakan suatu kunci sistem tes masuk mahasiswa baru berbasis *website* tersebut, karena pembuatan aplikasi ini menggunakan bahasa pemrograman PHP dan menggunakan database MySQL sebagai media penyimpan datanya, sehingga semakin mengerti tentang dasar PHP dan MySQL pembuatan sistem menjadi lebih cepat.

4. Pengumpulan data

Tahap pengumpulan data pada penelitian ini melalui observasi, wawancara, dan studi literatur. Observasi dilakukan di sekitar kampus Universitas Sahid Surakarta, wawancara akan dilakukan kepada staf informasi, mahasiswa tahun 2016, 2017, dan 2018 sebagai bahan perbandingan. Pengumpulan data ini bertujuan untuk mengetahui permasalahan dan kebutuhan informasi mengenai sistem tes masuk mahasiswa baru berbasis *website* di Universitas Sahid Surakarta.

5. Analisis dan perancangan sistem

Analisis dan perancangan sistem tes masuk mahasiswa baru berbasis *website* bertujuan untuk mengetahui kesiapan data dan rancangan dari aplikasi yang diharapkan dapat memberikan suatu informasi dengan efektif. Pemodelan dan perancangan sistem ini menggunakan metode perancangan sistem berbasis OOAD (*Object Oriented Analysis and Design*) melalui tahap *use case diagram*, *class diagram*, *activity diagram*, *sequence diagram*, *component diagram*, dan *deployment diagram*, desain perancangan sistem menggunakan aplikasi StarUML.

6. Rancang Bangun Sistem

Membuat sistem tes masuk mahasiswa baru berbasis *website* menggunakan bahasa pemrograman PHP dan *database* MySQL sebagai media penyimpanan datanya.

7. Pengujian Sistem

Pengujian sistem merupakan tahap setelah sistem berhasil dibuat. Pengujian sistem bertujuan untuk mengetahui kelebihan, kelemahan serta mengetahui kelayakan suatu sistem untuk digunakan. Pengujian sistem ini dibagi menjadi dua, pengujian fungsional aplikasi ini menggunakan metode *black box* dan pengujian pengguna aplikasi menggunakan metode kuesioner.

8. Dokumentasi

Dokumentasi merupakan catatan peristiwa yang sudah berlalu. Dokumentasi bisa berbentuk tulisan, gambar, atau karya monumental dari seseorang.

2.3 Landasan Teori

2.3.1 Sistem

Pengertian sistem menurut Romney dan Steinbart (2015), sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Sebagian besar sistem terdiri dari sub sistem yang lebih kecil yang mendukung sistem yang lebih besar.

Definisi sistem menurut Mulyadi (2016), sistem adalah suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan.

Menurut Susanto (2013), sistem adalah kumpulan atau grup dari sub sistem/bagian/komponen apapun baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu. Berdasarkan pengertian di atas dapat disimpulkan bahwa sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai tujuan dalam melaksanakan suatu kegiatan pokok perusahaan.

2.3.2 Informasi

Pengertian menurut Krismaji (2015), informasi adalah data yang telah diorganisasi dan telah memiliki kegunaan dan manfaat. Sementara menurut Romney dan Steinbart (2015), informasi adalah data yang telah dikelola dan

diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan. Sebagaimana perannya, pengguna membuat keputusan yang lebih baik sebagai kuantitas dan kualitas dari peningkatan informasi.

Berdasarkan pengertian di atas dapat disimpulkan bahwa pengertian informasi adalah data yang diolah agar bermanfaat dalam pengambilan keputusan bagi penggunanya.

2.3.3 Sistem Informasi

Pengertian menurut Kadir (2014), sistem informasi adalah sebuah rangkaian prosedur formal dimana data dikelompokkan, diproses menjadi informasi, dan didistribusikan kepada pemakai.

Pengertian menurut Krismaji (2015), sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, dan mengolah serta menyimpan data, dan cara-cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan.

Jadi berdasarkan pengertian di atas dapat disimpulkan bahwa sistem informasi adalah kumpulan data yang terintegritasi dan saling melengkapi dengan menghasilkan output yang baik guna untuk memecahkan masalah dan pengambilan keputusan.

2.3.4 Konsep Dasar Web

Menurut Ardhana (2012), *web* adalah salah satu layanan sajian informasi yang menggunakan konsep *hyperlink* (tautan), yang memudahkan *surfer* (sebutan pemakai komputer yang melakukan *browsing* atau penelusuran informasi melalui internet).

Menurut Pamungkas, dkk (2013), *web* adalah salah satu fitur dari internet di samping fitur-fitur lain seperti *e-mail*, *remote* (SSH), dan FTP, berdasarkan uraian di atas dapat disimpulkan *web* sebagai salah satu fitur dari internet dan layanan sajian informasi yang menggunakan konsep *hyperlink* (tautan), yang

memudahkan *surfer* (sebutan pemakai komputer yang melakukan *browsing* atau penelusuran informasi melalui (internet).

2.3.5 Website

Menurut Abdullah (2015), *website* dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa *text*, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet.

Menurut Hikmah, dkk (2015), *website* atau situs dapat diartikan sebagai kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan/atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Berdasarkan pengertian para ahli di atas dapat disimpulkan bahwa *website* adalah kumpulan dari halaman-halaman *web* yang berisi sebuah data atau informasi baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman yang diakses dan dibaca melalui *browser* seperti netscape navigator, internet explorer, mozilla firefox, google chrome dan aplikasi *browser* lainnya .

2.3.6 Bahasa Pemrograman

1. PHP (*Hypertext Preprocessor*)

Menurut Hidayatullah dan Kawistara (2017), PHP atau disingkat *Hypertext Preprocessor* ini adalah sebuah 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 bisa membuat *website powerful*

yang dinamis dengan disertai manajemen *database*-nya. Selain itu juga penggunaan PHP yang sebagian besar dapat jalan dibanyak *platform*.

2. HTML (*Hyperlink Text Markup Language*)

Menurut Sidik dan Husni (2005), HTML kependekan dari *Hyperlink Text Markup Language*. Dokumen HTML adalah file teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *web page*. Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*. Dokumen ini umumnya berisi informasi atau *interface* aplikasi di dalam internet.

Menurut Sibero (2013), *Hyperlink Text Markup Language* (HTML) adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*.

Berdasarkan uraian di atas maka kesimpulannya *Hyperlink Text Markup Language* (HTML) adalah bahasa yang digunakan untuk menulis halaman *web* sebagai pertukaran dokumen *web* dan dokumen ini umumnya berisi informasi atau *interface* aplikasi di dalam internet .

2.3.7 *Web Server*

Menurut Supono dan Putratama (2016), *web-server* adalah perangkat lunak *server* yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui HTTP atau HTTPS dari klien itu, yang dikenal sebagai *web browser* dan mengirimkan kembali (reaksi) hasil dalam bentuk situs yang biasanya merupakan dokumen HTML.

Menurut Sibero (2013), *web server* adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak.

Berdasarkan pendapat di atas dapat disimpulkan bahwa *web server* adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak server yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui *web browser* dan mengirimkan kembali (reaksi) hasil dalam bentuk sistus yang biasanya merupakan dokumen HTML.

2.3.8 Web Browser

Menurut Sibero (2012), *web browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi *web*.

Menurut Winarno (2013), *web browser* merupakan suatu program yang dirancang untuk mengambil informasi-informasi dari suatu *server* komputer pada jaringan internet. Jadi untuk mengakses *web* diperlukan suatu program yaitu *web browser* atau disebut *browser* saja.

Berdasarkan pengertian di atas dapat disimpulkan bahwa *browser* adalah sebuah perangkat lunak yang dirancang untuk menampilkan dan mengakses informasi atau halaman-halaman yang tersedia di *web server*.

2.3.9 Internet

Menurut Ahmadi dan Hermawan (2013), internet adalah komunikasi global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem operasi dan mesin.

Sedangkan Menurut Sibero (2013), internet (*interconnected network*) adalah jaringan komputer yang menghubungkan antar jaringan secara global, internet dapat juga disebut jaringan dalam suatu jaringan yang luas.

Berdasarkan kesimpulan di atas internet adalah sistem jaringan komunikasi secara global yang menghubungkan seluruh komputer di dunia dengan waktu dan wilayah tak terbatas sehingga semua orang dapat selalu terhubung di waktu siang maupun malam dengan platform apapun.

2.3.10 Definisi Basis Data

Definisi menurut Hidayatullah dan Kawistara (2017), basis data dapat didefinisikan sebagai himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah. Prinsip utamanya adalah pengaturan data. Tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data. Secara lebih lengkap pemanfaatan data dilakukan untuk memenuhi kecepatan dan kemudahan (*speed*), efisiensi ruang penyimpanan (*space*), keakuratan (*accuracy*),

ketersediaan (*availability*), kelengkapan (*completeness*), keamanan (*security*), pemakaian bersama (*sharability*).

2.3.11 MySQL

Menurut Sibero (2013), MySQL atau dibaca “My Sekuel” dengan suatu RDBMS (*Relational Database Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data.

Menurut Hidayatullah dan Jauhari (2015), MySQL adalah salah satu aplikasi DBMS yang sudah banyak oleh para pemogram aplikasi *web*. Contoh DBMS lainnya adalah : PostgreSQL (*freeware*), SQL Server, MS Access dari Microsoft, DB2 dari IBM, Oracle dan Oracle Corp, Dbase, FoxPro, dsb.

Berdasarkan penjelasan di atas dapat disimpulkan bahwa MySQL adalah perangkat lunak sistem manajemen basis data SQL atau DBMS (*Database Management System*) yang *multithread* dan *multi-user* dan menjalankan fungsi pengolahan data untuk membangun sebuah aplikasi berbasis *website* maupun desktop.

2.3.12 Metode Angket atau Kuesioner

Menurut Sugyiono (2014), kuesioner merupakan teknik pengumpulan data yang efisien apabila peneliti tahu dengan siapa variabel akan diukur dan tahu apa yang bisa diharapkan dari responden. Kuesioner dapat berupa pertanyaan-pertanyaan tertutup atau terbuka, dapat diberikan kepada responden secara langsung atau dikirim melalui pos atau internet. Angket keterampilan menjelaskan dan bertanya, untuk mengetahui persepsi siswa 42 mengenai keterampilan mengajar guru di kelas. Dalam angket tersebut terdapat komponen-komponen yang dapat dinilai oleh siswa terhadap kemampuan guru dalam keterampilan menjelaskan dan bertanya selama pembelajaran berlangsung. Untuk angket data yang diperoleh angket berupa nilai skor. Untuk menentukan skor pilihan jawaban angket menggunakan skala *Likert*. Dikemukakan Sugiyono (2014) bahwa skala *Likert* digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Kriteria pemberian skor untuk

alternatif jawaban untuk setiap item sebagai berikut : (1) Skor 4 untuk jawaban sangat setuju, (2) Skor 3 untuk jawaban setuju, (3) Skor 2 untuk jawaban tidak setuju, (4) skor 1 untuk jawaban sangat tidak setuju.

2.3.13 Black Box Testing

Pengujian menggunakan sekumpulan aktifitas validasi, dengan pendekatan *black box testing*. Menurut Shalahuddin dan Sukamto (2011), *black box testing* adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai.

2.3.14 Framework

Framework menurut Hidayatullah dan Kawistara (2017), *framework* dapat diartikan sekumpulan perintah/program dasar di mana perintah dasar tersebut dapat digunakan lagi untuk menyelesaikan masalah yang lebih kompleks sehingga dapat digunakan untuk membantu membuat aplikasi baru atau aplikasi kompleks tanpa harus membuat program dari awal.

2.3.15 Aplikasi Pendukung

1. PHPMyAdmin

Nugroho (2004), PhpMyAdmin adalah suatu program *open source* yang berbasis *web* yang dibuat menggunakan aplikasi PHP. Program ini digunakan untuk mengakses *database* MySQL dan untuk mempermudah dan mempersingkat kerja kita. Dengan kelebihanannya, para pengguna awam tidak harus paham syntax-syntax SQL dalam pembuatan *database* dan tabel.

Hidayatullah dan Kawistara (2014) menyatakan, *PHPMyAdmin* adalah *tool open source* yang ditulis dalam bahasa PHP untuk menangani administrasi MySQL berbasis *world wide web*.

2. XAMPP

Menurut Purbadian (2016), berpendapat bahwa, XAMPP merupakan suatu *software* yang bersifat *open source* yang merupakan pengembangan dari LAMP (Linux, Apache, MySQL, PHP dan Perl).

Menurut Kartini (2013), Xampp merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket.

Sedangkan menurut Riyanto (2010), Xampp merupakan paket PHP dan MySQL berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP.

3. Sublime Text

Menurut Supono (2016), sublime text merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-*edit* suatu aplikasi. Sublime text mempunyai fitur *plugin* tambahan yang memudahkan programmer. Selain itu, sublime text terkesan elegan untuk sebuah *syntax editor*. Selain ringan, IDE (*Integrated Development Environment*) ini memiliki kecepatan proses simpan dan buka file. Tidak heran kalau IDE ini paling banyak digunakan terutama dikalangan programmer berbasis *web*.

4. Codeigniter

Pengertian codeigniter menurut Hidayatullah dan Kawistara (2017), codeigniter adalah salah satu *framework* PHP bahkan *framework* PHP yang paling *powerful* saat ini karena di dalamnya terdapat fitur lengkap aplikasi *web* di mana fitur-fitur tersebut sudah dikemas menjadi satu. Selain itu codeigniter juga saat ini banyak digunakan khususnya bagi *developer web* untuk mengembangkan aplikasi berbasis *webnya* tersebut.

5. Bootstrap

Menurut Husein (2013), bootstrap merupakan *framework* ataupun *tools* untuk membuat aplikasi *web* ataupun *web responsive* secara tepat, mudah, dan gratis. Bootstrap terdiri dari CSS (*Cascading Style Sheet*) dan HTML (*Hypertext Markup Language*) untuk menghasilkan *grid*, *layout*, *typography*, *table*, *form*, *navigation*, dan lain-lain. Di dalam bootstrap juga sudah terdapat *JQuery plugins* untuk menghasilkan komponen UI yang menarik.

2.3.16 Pengertian UML

Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu (Sukamto dan Shalahudin, 2014).

Sukamto dan Shalahudin (2014), pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Berikut ini penjelasan singkat dari pembagian kategori tersebut.




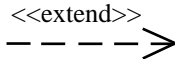
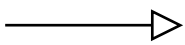
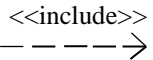
1. *Structure diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. *Structure diagram* terdiri dari *class diagram*, *object diagram*, *component diagram*, *composite structure diagram*, *package diagram* dan *deployment diagram*.
2. *Behavior diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. *Behavior diagram* terdiri dari *use case diagram*, *activity diagram*, *state machine system*.
3. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar *sub* sistem pada suatu sistem. *Interaction diagram* terdiri dari *sequence diagram*, *communication diagram*, *timing diagram*, *interaction overview diagram*.

2.3.17 Pengertian Use Case Diagram

Sukamto dan Shalahudin (2014), *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. Simbol *use case diagram* disajikan pada Tabel 2.1.

Tabel 2.1 Simbol *Use Case Diagram* (Sukamto dan Shalahuddin, 2013).

NO	SIMBOL	NAMA	KETERANGAN
1		<i>Use case</i>	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan kata kerja di awal di awal frase nama <i>use case</i> .
2		<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3		<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5		<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6		<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.


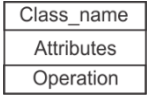



2.3.18 Pengertian *Class Diagram*

Sukamto dan Shalahudin (2014), diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan method atau operasi. Berikut penjelasan atribut dan method :

1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau method adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Simbol *class diagram* disajikan pada Tabel 2.2.







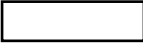
Tabel 2.2 Simbol *Class Diagram* (Sukamto dan Shalahuddin, 2013).

NO	GAMBAR	NAMA	KETERANGAN
1		<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>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.3.19 Pengertian *Activity Diagram*

Sukamto dan Shalahudin (2014), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Simbol *activity diagram* disajikan pada Tabel 2.3.

Tabel 2.3 Simbol *Activity Diagram* (Sukamto dan Shalahuddin, 2013).



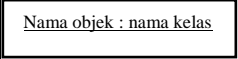

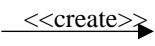
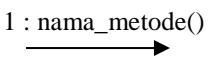
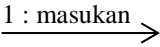
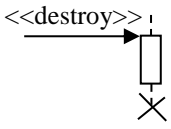
No	Gambar	Nama	Keterangan
1		<i>Start Point</i>	Diletakkan pada objek kiri atas dan merupakan awal aktivitas
2		<i>End Point</i>	Akhir aktivitas
3		<i>Activities</i>	Menggambarkan suatu proses atau kegiatan bisnis
4		<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau digunakan untuk menggabungkan dua kegiatan menjadi satu
5		<i>Join</i>	Digunakan untuk menunjukkan dekomposisi
6		<i>Decision Points</i>	Menggambarkan pilihan untuk pengambilan keputusan true atau false
7		<i>Swimlane</i>	Pembagian <i>activity diagram</i> dalam sebuah urutan yang sama

2.3.20 Pengertian *Sequence Diagram*

Sukamto dan Shalahudin (2014), diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan

message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. *Sequence diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.4.

Tabel 2.4 Simbol *Sequence Diagram* (Sukamto dan Shalahuddin, 2016).

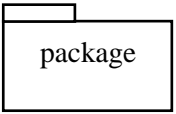
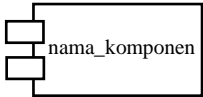



NO	SIMBOL	NAMA	KETERANGAN
1		Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
2		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
3		Objek	Menyatakan objek yang berinteraksi pesan.
4		Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi.
5		Pesan tipe create	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6		Pesan tipe call	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.
7		Pesan tipe send	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8		Pesan tipe <i>destroy</i>	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

2.3.21 Pengertian *Component Diagram*

Component diagram menurut Sukamto dan Shalahudin (2014), diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada didalam sistem.

Hubungan antara *component* dan *class*, *component* adalah implementasi *software* dan sebuah *class*. *Class* mewakili abstraksi dari serangkaian *attribute* dan *operation*. Hal terpenting yang perlu diingat tentang *class* dan *component* adalah sebuah *component* bisa jadi merupakan implementasi dari lebih dari sebuah *class*. *Component diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.5.

Tabel 2.5 Simbol *Component Diagram* (Sukamto dan Shalahudin, 2014).

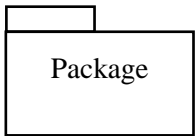
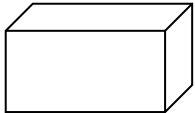
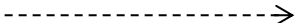

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Package</i>	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih komponen
2		<i>Component</i>	Komponen sistem
3		<i>Dependency</i>	Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai
4		<i>Interface</i>	Sama dengan konsep <i>interface</i> pada pemrograman berorientasi objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen
5		<i>Link</i>	Relasi antar komponen

2.3.22 Pengertian *Deployment Diagram*

Menurut Sukamto dan Shalahuddin (2014), mengatakan bahwa diagram *deployment* atau *deployment diagram* menunjukan konfigurasi komponen dalam proses eksekusi aplikasi. Secara umum pengertian diagram komponen adalah menggambarkan struktur fisik dari sebuah sistem dan digunakan untuk

mengilustrasikan bagaimana kode program dibagi menjadi beberapa komponen, dan mendeskripsikan hubungan antar komponen. Diagram komponen memberikan gambaran tingkat tinggi dan sederhana dari sebuah sistem. Mengklasifikasikan sekelompok kelas menjadi komponen akan membantu pertukaran dan penggunaan kembali dari sebuah kode. Tujuan dari diagram komponen adalah menunjukkan hubungan struktural antar komponen dalam sebuah sistem. *Deployment diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.6.

Tabel 2.6 Simbol *Deployment Diagram* (Sukamto dan Shalahudin, 2014).

SIMBOL	NAMA	KETERANGAN
	<i>Package</i>	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih komponen.
	<i>Node</i>	<i>Node</i> biasa mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang telah didefinisikan sebelumnya pada diagram komponen.
	<i>Dependency</i>	Ketergantungan atau <i>dependency</i> atau kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
	<i>Link</i>	Menggambarkan relasi antar <i>node</i> .