

BAB II

TINJAUAN PUSTAKA

2.1 Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu, sistem juga dinyatakan abstrak atau fisis^[1]. Sistem yang abstrak yaitu sistem tersusun secara teratur dari gagasan-gagasan yang saling berhubungan atau saling bergantung. Sedangkan sistem yang bersifat fisis adalah bagaimana cara suatu sistem bekerjasama secara bersama-sama untuk mrncapai suatu tujuan.

2.2 Informasi

Berikut ini beberapa pendapat mengenai pengertian informasi:

- a. Informasi adalah sekumpulan fakta (data) yang diorganisasikan dengan cara tertentu sehingga mereka mempunyai arti bagi si penerima^[2].
- b. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna atau lebih berarti bagi yang menerimanya^[3].
- c. Informasi adalah data yang sudah mengalami pemrosesan sedemikian rupa sehingga dapat digunakan oleh penggunanya dalam membuat keputusan^[4].

Berdasarkan beberapa pendapat yang dikemukakan di atas, maka dapat disimpulkan bahwa Informasi adalah suatu data yang diorganisasikan dan juga diolah yang mengalami pemrosesan menjadi bentuk data yang lebih berguna untuk penerimanya dan digunakan untuk membuat suatu keputusan

2.3 Sistem Informasi

Sistem Informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan data transaksi harian, mendukung operasi yang bersifat manajerial dan kegiatan strategi dari suatu organisasi yang menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan^[5].

Sistem informasi adalah sistem yang dibuat oleh manusia yang digunakan sebagai penyedia informasi, menyimpan, mengelola data dalam suatu himpunan yang berbasis komputer komputer dan manual untuk ditujukan kepada pemakai. Dalam artian luas sistem informasi adalah cara berinteraksi antara pengguna, proses algoritmik, data, dan informasi dengan teknologi yang dapat mempermudah dalam kegiatan bisnis.

2.4 Sistem Informasi Akademik

Sistem Informasi Akademik adalah perangkat lunak yang digunakan untuk menyajikan informasi dan menata administrasi yang berhubungan dengan kegiatan akademis. Dengan menggunakan perangkat lunak seperti ini diharapkan kegiatan administrasi akademis dapat dikelola dengan baik dan informasi yang diperlukan dapat diperoleh dengan mudah dan cepat^[6]. Akademik adalah suatu bidang pembelajaran yang dikelola oleh suatu lembaga pendidikan untuk peningkatan pengetahuan dalam bidang pendidikan yang mempelajari kurikulum. Sedangkan Sistem informasi merupakan aplikasi komputer untuk mendukung operasi dari suatu organisasi: operasi, instalasi, perangkat lunak, data dan perawatan komputer^[7]. Komponen sistem informasi terdiri dari:

- a. *Hardware*, terdiri dari perangkat keras komputer dan perangkat pendukung lainnya.
- b. *Software*, merupakan kumpulan dari perintah atau fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.
- c. *Data*, merupakan komponen dasar yang menyediakan dan mempunyai nilai informasi.
- d. *Sumber Daya Manusia*, merupakan komponen lingkungan yang berusaha memecahkan masalah dan mengolah sistem informasi.

Jadi dapat disimpulkan bahwa sistem informasi akademik adalah sistem yang menangani hal dalam bidang akademik guna untuk mengetahui bagaimana pencapaian prestasi seorang siswa atau mahasiswa dalam proses pendidikan. Sistem akademik mulai dari pendaftaran, nilai, dan juga proses belajar-mengajar.

2.5 Akademik

Akademik adalah suatu bidang pembelajaran yang dikelola oleh suatu lembaga pendidikan untuk peningkatan pengetahuan dalam bidang pendidikan yang mempelajari kurikulum^[6].

Sedangkan mengajar adalah perbuatan yang dilakukan seseorang dengan tujuan untuk membantu atau memudahkan orang lain melakukan kegiatan belajar. Berdasarkan dari pendapat ahli tersebut, dapat penulis tarik benang merah bahwa proses kegiatan akademik baik itu belajar dan mengajar adalah memiliki suatu tujuan yang dapat membawa perubahan dalam pengetahuan, pemahaman, dan keterampilan pembelajar tersebut^[8].

2.6 Konsep Dasar Ujian

2.6.1 Pengertian Ujian

Ujian merupakan suatu cara dan juga alat ukur evaluasi untuk mengukur taraf pencapaian belajar dan pemahaman dari ilmu yang didapat oleh peserta didik. Ujian juga dijadikan sebagai evaluasi nilai dalam proses belajar-mengajar seberapa jauh ilmu yang dikuasai dan juga dapat mendorong wawasan dan pemikiran seseorang dalam kegiatan pembelajaran.

Dengan ujian kita dapat mengetahui seberapa jauh ilmu yang kita kuasai dan juga kita dapat selama ini. Dalam ujian kita dituntut untuk berfikir secara kritis, benar dan tepat.

2.6.2 Jenis-Jenis Ujian

Berikut adalah jenis-jenis ujian :

a. Ujian Konvensional

Ujian konvensional atau ujian manual adalah jenis ujian yang sudah ada dan diterapkan puluhan tahun yang lalu, ujian manual ini masih menggunakan alat tulis berupa kertas, pena, pensil dan alat tulis umum lainnya untuk pelaksanaan ujian dan dilakukan dengan tulis tangan baik untuk distribusi soal maupun jawaban, pengumpulan hasil ujian juga masih manual yaitu memberikan langsung kepada pengawan ujian ditempat^[9]

b. Ujian Modern

Ujian modern adalah ujian yang hampir sama dengan ujian konvensional atau ujian manual. Namun perbedaannya ujian modern sudah menggunakan alat ketik untuk penulisan soal dan untuk memperbanyak soal menggunakan mesin *fotocopy*. Pemeriksaan ujian juga sudah dipermudah dengan adanya alat yang dapat memeriksa hasil ujian secara komputerisasi. Bentuk ujian ini biasanya bersifat objektif atau pilihan ganda^[9].

c. Ujian Online

Ujian *online* merupakan jenis ujian dimana pendistribusian soal sampai pengoreksian hasil ujian semuanya dilakukan secara *online* menggunakan perangkat elektronik yang terkoneksi internet. Ujian *online* dapat membantu pengajar dalam proses pengoreksian sehingga tidak perlu pengumpulan lembar jawaban satu persatu. Dengan menggunakan ujian *online* (*online test*) ini dapat mengukur kemampuan seseorang dalam menggunakan komputer. Penilaian pada ujian *online* ini dapat digunakan untuk meminimalisir terjadinya kecurangan dan *human error*^[9].

2.7 Metode Kipling 5W+1H

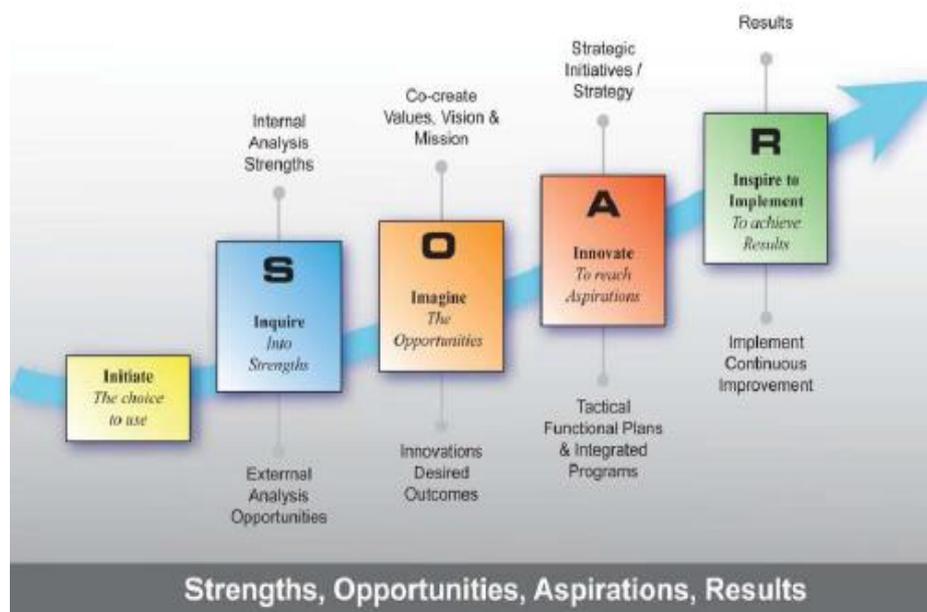
Pada tugas akhir ini dalam penulisan tahapan planning menggunakan metode analisis 5W+1H, untuk mendapatkan gambaran secara detail mengenai sistem informasi yang dirancang. Berikut penjelasan tentang 5W+1H sebagai metode analisis pada sistem yaitu sebagai berikut^[10]:

- a. *What* (Apa), Bentuk pertanyaan yang bertujuan untuk mencari tahu mengenai sesuatu perangkat lunak dan sistem informasi yang akan dirancang atau dikembangkan.
- b. *Who* (Siapa), Bentuk pertanyaan yang bertujuan mencari tahu setiap subjek atau aktor yang terlibat dalam penggunaan dan perancangan perangkat lunak dan sistem informasi.
- c. *Where* (Dimana), Bentuk pertanyaan yang bertujuan untuk menjelaskan tempat atau lokasi yang berhubungan dengan pemakaian perangkat lunak tersebut.

- d. *Where* (Kapan), Bentuk pertanyaan yang bertujuan mencari tahu tempat kejadian suatu peristiwa.
- e. *Why* (Mengapa), Bentuk pertanyaan yang bertujuan menjelaskan alasan dan sebab akibat dari adanya perancangan perangkat lunak dan sistem informasi.
- f. *How* (Bagaimana), Bentuk pertanyaan dengan tujuan mencari tahu proses peristiwa yang terjadi dalam perancangan perangkat lunak dan sistem informasi.

2.8 Metode SOAR

SOAR adalah kerangka perencanaan strategis dengan pendekatan yang berfokus pada kekuatan dan mencari untuk mengerti keseluruhan sistem dengan memasukkan suara dari stakeholders yang relevan. Pendekatan ini memungkinkan organisasi untuk membangun masa depan melalui kolaborasi, pemahaman bersama, dan komitmen untuk bertindak^[11].



Gambar 2.1 SOAR framework

(Sumber: Stavros , Hinrichs. 2009. *The Thin Book of Appreciative Inquiry 3rd Edition*. Thin Book Series. 1-4)

Berikut ini merupakan unsur-unsur dari analisis SOAR :

a. *Strength (S)*

Strength (S) merupakan segala hal yang menjadi kekuatan pada kemampuan terbesar yang dimiliki. Bentuk dari kekuatan tersebut dapat berupa aset yang berwujud maupun yang tidak berwujud. Kekuatan dapat terus dikembangkan agar terjadinya kemajuan.

b. *Opportunities (O)*

Opportunities (O) merupakan segala bentuk peluang yang ada pada lingkungan eksternal dari suatu usaha. Peluang tersebut harus dapat dianalisis dengan baik, kemampuan dalam memaksimalkan peluang akan mempengaruhi keberhasilan usaha.

c. *Aspiration (A)*

Keberhasilan juga memerlukan aspirasi yang dapat diperoleh dari berbagai pihak yang terlibat pada suatu usaha tersebut. Untuk memperoleh aspirasi dapat dilakukan dengan cara diskusi dan bertukar pendapat.

d. *Results (R)*

Results (R) merupakan hasil dan tujuan yang akan dicapai dalam suatu usaha, mengidentifikasi *results* di awal suatu proses dapat menjadikan acuan dan motivasi dalam mencapai tujuan yang telah .

2.9 UML Versi 2.0

Unified Modelling Language (UML) adalah bahasa standar untuk penulisan cetak biru (*blue print*) perangkat lunak. *UML* dapat digunakan untuk memvisualisasikan, menentukan, mengkonstruksi, dan mendokumentasikan artifak-artifak suatu sistem software. *UML 2.0* memberikan 13 diagram yang berbeda-beda untuk digunakan pada pemodelan perangkat lunak^[12].

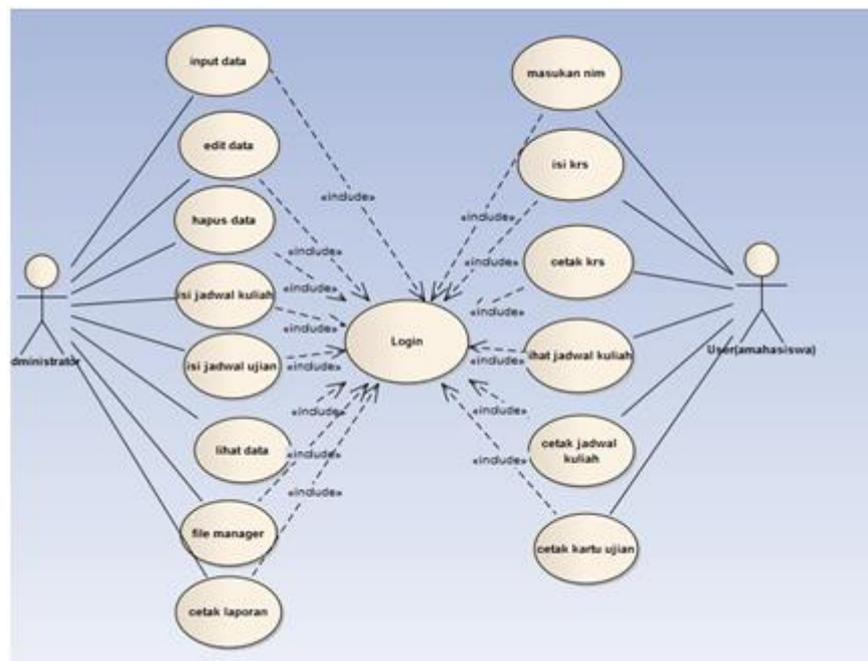
Pada tugas akhir ini penulis menggunakan *UML* Vversi 2.0 dan menggunakan 4 diagram. Keempat diagram tersebut adalah *usecase diagram*, *activity diagram*, *class diagram*, dan *sequence diagram*. Dengan menggunakan 4 diagram tersebut sudah cukup untuk menggambarkan keseluruhan dari perangkat

lunak dan sistem informasi yang akan dirancang. Berikut penjelasan dari 4 diagram tersebut :

a. *Usecase Diagrams*

Usecase menggambarkan bagaimana pengguna berinteraksi dengan sistem dengan cara menentukan langkah-langkah yang diperlukan untuk mencapai suatu tujuan tertentu. Tujuan utama dari *use case* terhadap proses pengembangan perangkat lunak adalah deskripsi tekstual dari masing-masing *usecase*.

Pada diagram *usecase*, *use case-use case* tersebut ditampilkan dengan bentuk oval. Aktor-aktor dihubungkan dengan garis-garis ke *usecase*. *Use case* ditempatkan pada kotak, namun aktor-aktor berada diluar kotak tersebut. Kotak tersebut merupakan pengingat visual tentang batasan-batasan antara sistem dan aktor^[12].



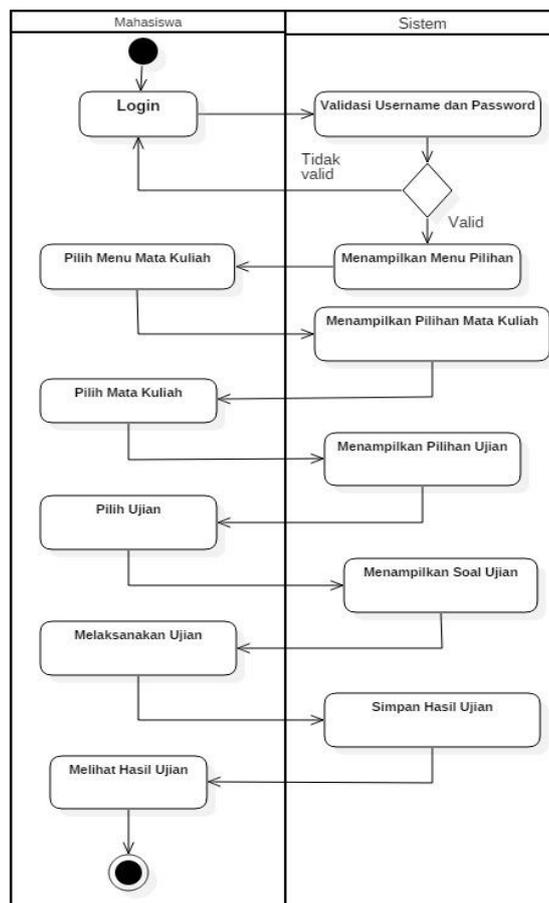
Gambar 2.2 Contoh *usecase diagram*

(Sumber : http://nurhayatiizz.blogspot.com/2012/08/use-case-sistem-informasi-pelaksanaan_13.html)

b. *Activity Diagrams*

Activity diagrams menggambarkan perilaku dinamis dari suatu sistem atau suatu komponen sistem melalui aliran kontrol atau alir kerja di antara aksi-aksi yang

dilakukan sistem yang berisi aktivitas^[12]. Komponen utama suatu diagram aktivitas adalah *node* aksi, direpresentasikan dengan suatu persegi panjang dengan sudut bulat, yang berhubungan dengan suatu tugas yang dilakukan oleh sistem perangkat lunak. Tanda panah ke *node* lain mengartikan bahwa ada aliran kerja aktivitas. Titik hitam pada awal adalah *initial node* sedangkan titik hitam dengan lingkaran luar menandakan *final node*.



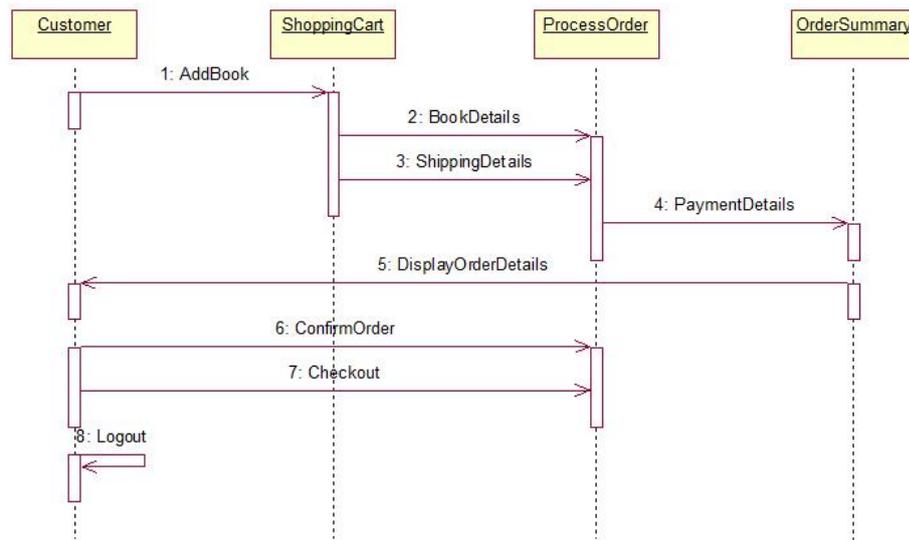
Gambar 2.3 Contoh *activity diagram*

(Sumber: <http://rasalibre.co/draw-a-uml-sequence-diagram-for-online-bookshop.html>)

c. *Sequence Diagrams*

Sequence diagram menggambarkan urutan sementara dimana pesan-pesan dikirim diantara objek-objek untuk menyelesaikan pekerjaan tersebut. Diagram urutan juga digunakan untuk menunjukkan interaksi-interaksi pada suatu *use case* atau pada suatu skenario dari suatu sistem perangkat lunak^[12]. Komponen utama

yang ada pada sequence diagram adalah *object*, *activation boxes*, *actors* dan *lifeline*. *Object* adalah komponen yang berbentuk kotak yang mendeskripsikan bagaimana sebuah *object* bertindak kepada sistem. *Activation boxes* adalah komponen persegi panjang yang menggambarkan waktu yang diperlukan *object* menyelesaikan tugas. *Actors* adalah yang mewakili seorang pengguna. Dan *lifeline* merupakan komponen berbentuk garis putus-putus yang menggambarkan aktifitas *object*.

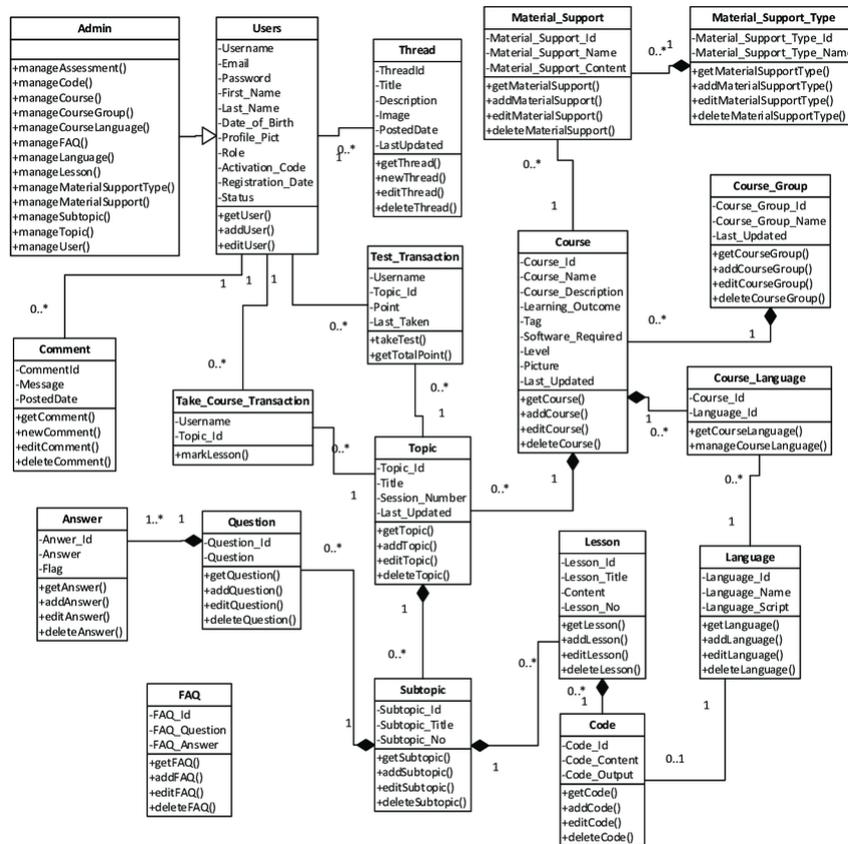


Gambar 2.4 Contoh *sequence diagram*

(Sumber: <http://rasalibre.co/draw-a-uml-sequence-diagram-for-online-bookshop.html>)

d. *Class Diagrams*

Class diagrams atau dapat disebut juga diagram kelas digunakan untuk memodelkan kelas-kelas, termasuk atribut-atributnya, operasi-operasi, dan hubungan serta asosiasinya dengan kelas-kelas lain^[11]. Terdapat 4 elemen pada class diagram yaitu *classes name*, *attributee*, *operations*, dan *relationships*.



Gambar 2.5 Contoh class diagram

(Sumber: https://www.researchgate.net/figure/Class-Diagram-of-Online-Learning-System_fig4_320211937)

2.10 Mobile

Kata *mobile* menunjuk kepada perangkat dengan teknologi tinggi yang dapat bergerak digunakan sebagai perangkat komunikasi tanpa menggunakan kabel. Aplikasi yang akan dirancang ini berupa perangkat lunak yang ada pada perangkat mobile^[13]. Dengan berkembangnya ajingan internet dengan jangkauan yang luas kini memberikan kemudahan mencari dan mendapatkan informasi juga berkomunikasi dengan orang lain yang berada jauh semakin mudah dan cepat. Untuk mengakses informasi serta berkomunikasi tersebut kita menggunakan sebuah perangkat *mobile* seperti handphone atau *smartphone*. Maka penggunaan *mobile* aplikasi dan *mobile web* tidak bisa kita hindari lagi.

a. *Mobile Web*

Mobile web adalah halaman HTML yang dapat diakses dengan menggunakan perangkat mobile (smartphone atau tablet) melalui jaringan internet 3G, 4G atau WiFi. *Mobile web* ini akan menampilkan konten-konten berupa text, gambar, video maupun audio.

Mobile web memiliki prinsip “*develop once run everywhere*”, yang berarti dapat beroperasi pada lintas *platform* dalam sekali pengembangan. Jadi, *mobile web* relatif tidak membutuhkan banyak biaya untuk pengembangan dibandingkan *mobile apps* karena dapat berjalan dengan baik di semua *browser* dan platform *mobile*. *Mobile web* dapat didistribusikan tanpa harus mendapatkan *approval* dari pihak tertentu dan pengguna tidak perlu harus mencari di toko aplikasi. *Mobile web* hanya tersedia hanya jika pengguna dalam keadaan *online*, performa relatif lambat, dan tidak semua perangkat memiliki fitur tertentu yang bisa dimiliki dalam *mobile web*^[13].

b. *Mobile Aplikasi*

Mobile aplikasi adalah aplikasi yang dirancang untuk digunakan pada perangkat smartphone yang dapat dibeli atau *download* melalui toko aplikasi seperti *Play Store*, *App Store*, maupun *Blackberry App* kemudian *install* untuk dapat dioperasikan.

Mobile apps memiliki performa lebih cepat, hanya memiliki 1 domain, dan lebih menarik dari segi visual. Pengguna memiliki akses penuh ke perangkat mobile, seperti kamera, gesture, pemberitahuan, *speaker*, dan fitur lainnya. Kualitas dan keamanan *mobile apps* terjamin karena dikontrol oleh vendor masing-masing. *Mobile apps* tidak selalu dapat dioperasikan ketika pengguna *online*, tetapi dalam keadaan *offline* juga bisa. Hanya saja, ada beberapa jenis aplikasi yang tidak digunakan di semua *platform* IOS lain. Biaya pengembangan dan *maintenance* aplikasi lebih mahal. *Mobile apps* juga mengharuskan adanya proses persetujuan dan pembatasan konten yang berlaku di toko aplikasi^[13].

2.11 Website

Website atau disingkat *web*, 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 internet. Lebih jelasnya, *website* merupakan halaman-halaman yang berisi informasi yang ditampilkan oleh *browser* seperti Mozilla Firefox, Google Chrome atau yang lainnya^[14].

Untuk menyediakan sebuah website, maka harus tersedia unsur-unsur penunjangnya, adalah sebagai berikut:

- a. Nama Domain (*domain name*) : Nama domain adalah alamat unik di dunia internet yang digunakan untuk mengidentifikasi sebuah *website*, atau dengan kata lain alamat yang digunakan untuk menemukan sebuah *website* pada dunia internet. Nama domain diperjualbelikan secara bebas di internet dengan status sewa tahunan.
- b. Rumah tempat website (*web hosting*) : *Web Hosting* dapat diartikan sebagai ruangan yang terdapat menyimpan berbagai data, *file-file*, gambar, video, data *email*, statistik, *database*, dan lain sebagainya yang akan ditampilkan di *website*. *Web Hosting* juga diperoleh dengan menyewa. Pengguna akan memperoleh kontrol panel yang terproteksi dengan *username* dan *password* untuk administrasi *websitenya*.
- c. Bahasa Program (*Scripts Program*) : Bahasa yang digunakan untuk menerjemahkan setiap perintah dalam website yang pada saat diakses. Jenis bahasa program sangat menentukan statis, dinamis, atau interaktifnya sebuah *website*. Semakin banyak bahasa program yang digunakan maka akan terlihat website semakin dinamis dan interaktif. Jenis-jenis bahasa program yang banyak dipakai antara lain: HTML, ASP, PHP, JSP, Java Scripts, Java Applets, XML, Ajax dan sebagainya.
- d. Desain Website : Setelah melakukan penyewaan *domain name* dan *web hosting* serta penguasaan bahasa program, unsur *website* yang penting dan utama adalah desain. Desain *web* menentukan kualitas dan keindahan sebuah *website*. Desain

akan berpengaruh kepada penilaian pengunjung akan bagus tidaknya sebuah *website*.

- e. Program transfer data ke pusat data : FTP (*File Transfer Protocol*) merupakan akses yang diberikan pada saat kita memesan *web hosting*, FTP berguna untuk memindahkan *file-file website* yang ada pada komputer kita ke pusat *web hosting* agar dapat terakses ke seluruh dunia.

2.12 RAD (*Rapid Application Development*)

Merupakan metode proses pengembangan perangkat lunak secara linear sequential yang menekankan pada siklus pengembangan yang sangat singkat. Jika kebutuhan dipahami dengan baik, proses RAD memungkinkan tim pengembangan menciptakan “sistem fungsional yang utuh” dalam periode waktu yang sangat pendek (kira-kira 60-90 hari). Pendekatan metode RAD menekankan cakupan^[15]:

a. Pemodelan bisnis (*Business Modelling*)

Aliran informasi diantara fungsi-fungsi bisnis dimodelkan dengan suatu cara untuk menjawab pertanyaan-pertanyaan berikut : Informasi apa yang mengendalikan proses bisnis ? Kemana informasi itu pergi? Siapa yang memrosesnya ?

b. Pemodelan data (*Data Modelling*)

Aliran informasi yang didefinisikan sebagai bagian dari fase pemodelan bisnis disaring ke dalam serangkaian objek data yang dibutuhkan untuk menopang bisnis tersebut. Karakteristik/atribut dari masing-masing objek diidentifikasi dan hubungan antara objek-objek tersebut didefinisikan.

c. Pemodelan proses (*Process Modelling*)

Aliran informasi yang didefinisikan dalam fase pemodelan data ditransformasikan untuk mencapai aliran informasi yang perlu bagi implementasi sebuah fungsi bisnis. Gambaran pemrosesan diciptakan untuk menambah, memodifikasi, menghapus atau mendapatkan kembali sebuah objek data.

d. Pembuatan aplikasi (*Application generation*)

Selain menciptakan perangkat lunak dengan menggunakan bahasa pemrograman generasi ketiga yang konvensional, RAD lebih banyak memproses kerja untuk memakai lagi komponen program yang telah ada atau menciptakan komponen yang bias dipakai lagi. Pada semua kasus, alat-alat Bantu otomatis dipakai untuk memfasilitasi konstruksi perangkat lunak.

e. Pengujian dan pergantian (*Testing and turnover*)

Karena proses RAD menekankan pada pemakaian kembali, banyak komponen yang telah diuji. Hal ini mengurangi keseluruhan waktu pengujian. Tapi komponen baru harus diuji.

2.13 PHP

PHP singkatan dari PHP *Hypertext Preprocessing*. Merupakan bahasa *scripting* untuk *web* yang cukup populer. Dengan PHP, kita bisa membuat *web* dinamis dimana kode PHP diselipkan di antara *script* kode-kode HTML yang merupakan bahasa markup standar untuk dunia web. PHP adalah bahasa *script*, artinya ditanamkan atau disisipkan ke dalam HTML. PHP adalah bahasa pemrograman yang populer dikalangan web programmer untuk membuat sebuah website yang dinamis dan interaktif. Untuk membedakan kode PHP dan kode HTML sebagai wadahnya, digunakan tag-tag PHP^[16].

2.14 MySQL

MySQL adalah *software database* yang paling banyak dipakai untuk membuat aplikasi dinamis. *Software* paketan AMP (Apache, MySQL, dan PHP) biasanya sudah mengandung MySQL. Indikasinya adalah dengan terlihatnya halaman PHPMyAdmin. MySQL adalah sebagai wadah untuk tempat penyimpanan text based data. Data berupa file, gambar, audio dan lain-lain diletakkan pada server dalam folder tanpa harus disimpan pada database, disimpan dengan referensi nama. MySQL ini adalah konstruksi database yang memungkinkan Apache dan PHP bekerja sama untuk mengakses dan menampilkan data dalam format yang terbaca di *browser*. MySQL adalah *server database* yang dapat diakses dengan

Structured Query Language. MySQL didesain untuk bisa menangani beban pemrosesan yang berat dan pemrosesan yang cepat^[16].

2.15 CSS

CSS (*Cascading Style Sheets*) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur *style* suatu dokumen. Pada umumnya CSS dipakai untuk memformat tampilan halaman *web* yang dibuat dengan bahasa HTML dan XHTML. CSS merupakan sekumpulan kode-kode program untuk mempercantik tampilan halaman HTML atau dengan kata lain CSS digunakan untuk memisahkan tampilan dari konte. CSS dapat mengubah besar kecilnya text, menjadi warna background pada sebuah halaman, atau dapat pula mengubah warna border pada tabel, dan masih banyak lagi hal yang dapat dilakukan oleh CSS. Singkatnya, CSS digunakan untuk mengatur susunan tampilan halaman HTML^[17].

2.16 XAMPP

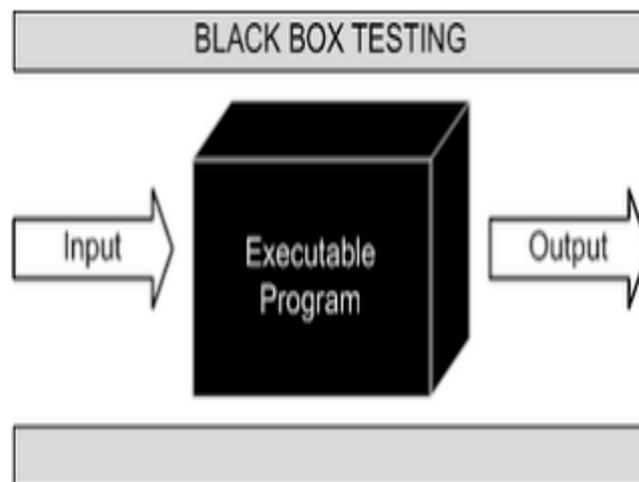
Salah satu paket *web server* yang praktis karena menginstal langsung Apache, PHP, dan MySQL adalah XAMPP. XAMPP adalah aplikasi pengembang yang berbasis PHP dan MySQL dan digunakan untuk mengembangkan sebuah *website*. XAMPP dilengkapi dengan fitur PHPMyAdmin sehingga pengembang dapat mengembangkan database dengan mudah. XAMPP tersedia untuk *multisystem* 19 operasi, seperti Windows dan Linux. Sebenarnya *server* untuk PHP tidak hanya XAMPP, ada banyak sekali *web server*^[16].

2.17 Codeigniter

Codeigniter dalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis php dibandingkan jika menulis semua kode program dari awal. *Codeigniter* menyediakan banyak *library* untuk mengerjakan tugas-tugas yang umumnya ada pada sebuah aplikasi berbasis *web*. Selain itu, struktur dan susunan logis dari *codeigniter* membuat aplikasi menjadi semakin teratur dan rapi^[18].

2.18 Metode Pengujian *Black-Box*

Black Box Testing adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan cek fungsional perangkat lunak. Jadi analogi seperti yang kita lihat kotak hitam, kita dapat melihat hanya penampilan luar, tanpa mengetahui apa di balik bungkus hitamnya. Sama seperti pengujian kotak hitam, mengevaluasi hanya penampilan *eksternal* (antarmuka), fungsional tanpa mengetahui apa yang sebenarnya terjadi dalam codingan^[19].



Gambar 2.6 Proses pengujian *black-box*
(Sumber : Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku Satu), (Roger S. Pressman, Ph.D. 2002 : 536)

***Black Box Testing* dapat menemukan *error* seperti:**

- a. Fungsi atau logika yang tidak benar
- b. *Error interface*
- c. *Error performance*
- d. Kesalahan dalam struktur data atau akses database eksternal

Tujuan *Black Box Testing* adalah untuk mencari kesalahan/kegagalan dalam operasi tingkat tinggi, yang mencakup kemampuan dari perangkat lunak, operasional/tata laksana, skenario pemakai. Fungsi dari pengujian ini berdasarkan kepada apa yang dapat dilakukan oleh sistem. Untuk melakukan pengujian perilaku seseorang harus mengerti lingkup dari aplikasi, solusi bisnis yang diberikan oleh aplikasi, dan tujuan sistem dibuat^[19].

2.19 *Netbeans*

NetBeans adalah *Integrated Development Environment* (IDE) berbasis *Java* dari *Sun Microsystems* yang berjalan di atas *Swing*. *Swing* adalah sebuah teknologi *Java* untuk pengembangan aplikasi *desktop* yang dapat berjalan di berbagai macam *platforms* seperti *Windows*, *Linux*, *Mac OS X* dan *Solaris* ^[20].

Suatu IDE adalah lingkup pemrograman yang diintegrasikan kedalam suatu aplikasi perangkat lunak yang menyediakan pembangun *Graphic User Interface* (GUI), suatu text atau kode editor, suatu *compiler* atau *interpreter* dan suatu *debugger*. *NetBeans* merupakan *software development* yang *Open Source*, dengan kata lain *software* ini di bawah pengembangan bersama dan bebas biaya. *NetBeans* IDE adalah sebuah lingkungan pengembangan sebuah *tools* untuk pemrogram menulis, mengompilasi, mencari kesalahan dan menyebarkan program. *NetBeans* IDE ditulis dalam *Java*, namun dapat mendukung bahasa pemrograman lain. Terdapat banyak modul untuk memperluas *Netbeans IDE* ^[20]



Gambar 2.7 Logo *netbeans*
(Sumber: <https://netbeans.org/>)

2.20 *Visual Studio Code*

Visual Studio merupakan salah satu perangkat lunak yang digunakan dalam pengembangan aplikasi. *Visual Studio Code* adalah salah satu teks editor yang dibuat *Microsoft* dapat digunakan pada sistem operasi *multi platform* seperti *Linux*, *Windows*, dan *Mac OS X*. *Visual Studio Code* mendukung bahasa pemrograman *Java Script*, *Type Script*, *Node.js* serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang pada marketplace *Visual Studio Code* ini seperti *C++*, *C#*, *Python*, *PHP*, *Go*, dan *Java* ^[21].

Visual Studio terdiri dari kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa *MSDN Library*). Kompiler yang dimasukkan ke dalam paket *Visual Studio* antara lain *Visual C++*,

Visual C#, Visual Basic, NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe. Visual Studio Code dalam perancangan ini digunakan untuk editor pembuatan aplikasi berbasis *mobile*^[21].



Gambar 2.8 Logo *visual studio code*

(Sumber : <https://medium.com/edgefund/c-development-with-visual-studio-code-b860cc71a5ec>)

2.21 *Bootstrap*

Bootstrap merupakan salah satu jenis *framework* untuk CSS (*Cascading Style Sheet*) yang sangat membantu bagi *programmer* untuk perancangan tampilan situs *website*. Penggunaan *Bootstrap* bisa untuk memilih fitur CSS dan *JavaScript* sesuai dengan kebutuhan atau menyesuaikan dengan kebutuhan *website*. Contohnya CSS pada *Bootstrap* juga menyediakan fitur *form*, tombol, navigasi dan komponen-komponen lainnya dan *JavaScript* yang membantu pembuatan antarmuka lebih mudah dan stabil. *Bootstrap* saat ini menjadi salah satu *front-end framework* yang banyak digunakan oleh *programmer* karena *bootstrap* juga menyediakan banyak *class-class* CSS dan *plug ini*, namun harus dikombinasikan dengan penggunaan *stylesheet*. *Stylesheet* digunakan sebagai pembeda tampilan *website* supaya tidak ada kemiripan antar tampilan *website*^[22].