BAB II

LANDASAN TEORI

2.1. Pengertian Aplikasi, Media, Pembelajaran, dan Latihan Ujian

Dalam hal ini, penulis mendefinisikan arti kata dari judul skripsi agar penulis lebih mengerti dan memahami maksud penelitian ini.

2.1.1. Pengertian Aplikasi

Aplikasi merupakan kumpulan atau rangkaian instruksi komputer (program komputer) yang bila kita eksekusi atau jalankan akan menghasilkan performansi dan fungsi yang kita kehendaki.**[2,p. 3]**

2.1.2. Pengertian Media

Media secara umum adalah alat bantu proses belajar mengajar atau segala sesuatu yang dapat dipergunakan untuk merangsang pikiran, perasaan, perhatian, dan kemampuan atau keterampilan pembelajar sehingga dapat mendorong terjadinya proses belajar.**[3]**

2.1.3. Pengertian Pembelajaran

Modul pembelajaran merupakan bahan ajar yang disusun secara sistematis dan menarik yang mencakup isi materi, metode dan evaluasi yang dapat digunakan secara mandiri untuk mencapai kompetensi yang diharapkan.**[4]**

2.1.3.1. Pembelajaran Aktif

Pembelajaran aktif (*active learning*) adalah suatu pembelajaran yang mengajak peserta didik untuk belajar secara aktif. Belajar aktif meliputi berbagai cara untuk membuat mahasiswa aktif sejak awal melakukan aktivitas-aktivitas yang

membangun kerja kelompok dalam waktu yang singkat membuat mereka berpikir tentang materi pelajaran.**[5]**

Pembelajaran aktif merupakan suatu bentuk pembelajaran yang memungkinkan mahasiswa berperan aktif dalam proses pembelajaran baik dalam bentuk interaksi sesama mahasiswa maupun mahasiswa dengan pengajar pada proses pembelajaran aktif tersebut. **[5]**

2.1.3.2. Karakteristik Pembelajaran Aktif

Pembelajaran aktif memiliki karakteristik sebagai berikut **[5]** :

1. Penekanan proses pembelajaran bukan pada penyampaian informasi oleh pengajar melainkan pada pengembangan keterampilan pemikiran analistis dan kritis terhadap topik atau permasalahan yang dibahas.
2. Siswa tidak mendengarkan pelajaran secara pasif, tetapi mengerjakan sesuatu yang berkaitan dengan materi pelajaran.
3. Penekanan pada eksplorasi nilai-nilai dan sikap-sikap berkenaan dengan materi pelajaran.
4. Siswa lebih banyak dituntut berpikir kritis menganalisa dan melakukan evaluasi.
5. Umpan balik yang lebih cepat akan terjadi pada proses pembelajaran.

2.1.3.3. Keuntungan Pembelajaran Aktif

Pembelajaran aktif selain mengoptimalkan segi keaktifan dalam pembelajaran juga banyak memberikan keuntungan lain yang mendukung kegiatan pembelajaran. Keuntungan pembelajaran aktif diantaranya sebagai berikut **[5]** :

1. Siswa akan lebih termotivasi karena akan lebih mudah belajar di saat enjoy.

2. Berlangsung dalam lingkungan yang tenang, karena percobaan dan kegagalan diterima.
3. Adanya partisipasi dari semua kelompok
4. Tiap orang bertanggung jawab atas pembelajarannya masing-masing.
5. Fleksibel dan relevan.
6. Sesuatu menyatakan pemikirannya.
7. Masing-masing memberikan koreksi jika ada kesalahan.

2.1.3.4. Pembelajaran Kooperatif

Pembelajaran kooperatif adalah salah satu bentuk pembelajaran yang berdasarkan paham konstruktivis. Pembelajaran kooperatif merupakan strategi belajar dengan sejumlah mahasiswa sebagai anggota kelompok kecil yang tingkat kemampuannya berbeda. Dalam menyelesaikan tugas kelompoknya, setiap mahasiswa anggota kelompok harus saling bekerja sama dan saling membantu untuk memahami materi pelajaran. Dalam Pembelajaran kooperatif, belajar dikatakan belum selesai jika salah satu teman dalam kelompok belum menguasai bahan pelajaran. **[6]**

Tujuan pembelajaran kooperatif berbeda dengan kelompok tradisional yang menerapkan sistem kompetisi, di mana keberhasilan individu diorientasikan pada kegagalan orang lain. Sedangkan tujuan dari Pembelajaran kooperatif adalah menciptakan situasi di mana keberhasilan individu ditentukan atau dipengaruhi oleh keberhasilan kelompoknya

2.1.4. Pengertian Latihan Ujian

Pada umumnya semua siswa yang akan menghadapi ujian nasional (UN) dan ujian akhir semester (UAS) di wajibkan untuk mempersiapkan diri terlebih dahulu agar mampu mengukur sejauh mana tingkat kesiapan siswa dalam menghadapi ujian. *Tryout*, latihan ujian, quiz dan berbagai jenis ujian lainnya dapat didefinisikan sebagai wahana latihan ujian siswa, untuk menghadapi ujian nasional atau ujian akhir semester (UAS). *Tryout* dan jenis ujian lainnya sudah merupakan suatu kegiatan rutin yang dilaksanakan oleh kelas 7, 8, 9, 10, 11 dan 12 bukan hanya terfokus kepada ujian semester saja. Akan tetapi untuk saat ini *tryout* dan berbagai jenis ujian itu lebih ditekankan untuk menghadapi ujian nasional. Hal yang membedakannya adalah bahwa *tryout* untuk menghadapi ujian nasional kelas 9 dan 12 itu waktunya sudah ditentukan oleh Dinas pendidikan kabupaten, sedangkan untuk ujian semester itu biasanya untuk kelas 7, 8, 10, dan 11 waktunya juga tergantung sepenuhnya kepada sekolah masing-masing.

Tujuan dilaksanakannya *tryout* dan berbagai ujian itu adalah sebagai wahana pembiasaan siswa terhadap penyelesaian soal-soal terutama yang sifatnya ujian bersama, baik yang diselenggarakan oleh dinas, gugus kecamatan, maupun pusat yang berarti mempersiapkan siswa untuk menghadapi ujian nasional.**[10]**

2.2. Pendekatan Berorientasi Objek

Pendekatan beorientasi objek merupakan paradigma pemograman yang berorientasi kepada objek. Semua data dan fungsi di dalam paradigma dibungkus dalam kelas-kelas atau objek-objek. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya.

2.2.1 Metodologi Berorientasi Objek

Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang merorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan poreasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metode berorientasi objek meliputi rangkaian aktivitas analisis orientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek.**[9,p. 82]**

Keuntungan menggunakan metodologi berorientasi objek adalah sebagai berikut:

1. Meningkatkan Produktivitas

   Karena kelas dan objek yang ditemukan dalam suatu masalah masih dapat dipakai ulang untuk masalah lainnya yang melibatkan objek tersebut *(reusable).*

2. Kecepatan Pengembangan

   Karena sistem yang dibangun dengan baik dan benar pada saat analisis dan perancangan akan menyebabkan berkurangnya kesalahan pada saat pengkodean.

3. Kemudahan Pemeliharaan

   Karena dengan model objek, pola-pola yang cenderung tetap dan stabil dapat dipisahkan dan pola-pola yang mungkin sering berubah-ubah.

4. Adanya Konsistensi

Karena sifat pewarisan dan penggunaan notasi yang sama pada saat analisis, perancangan, maupun pengkodean.

5. Meningkatkan Kualitas Perangkat Lunak

   Karena pendekatan pengembangan lebih dekat dengan dunia nyata dan adanya konsistensi pada saat pengembangannya, perangkat lunak yang dihasilkan akan mampu memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan.

2.2.2. *Object Oriented Programming* (OOP)

Object oriented programming merupakan suatu cara mengorganisasi program dengan memodelkan objek-objek dunia nyata seperti benda, sifat, sistem, dan lainnya ke dalam sebuah bahasa pemograman.

2.2.2.1. *Use Case*

Diagram ini memperlihatkan himpunan *use case* dan *actor-actor*. Diagram ini penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna.**[2,p. 201-202]**

2.2.2.2. Skenario *Use Case*

Skenario *Use Case* digunakan untuk memudakan dalam menganalisa skenario yang akan digunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut.

2.2.2.3. *Activity Diagram*

Diagram ini memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem. Diagram ini penting dalam pemodelan fungsi-fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antar objek.**[2,p. 205-206]**

2.2.2.4. *Sequence Diagram*

Diagram ini menggambarkan alur kerja dari fungsi-fungsi dalam sistem dengan *usecase* dimana didalamnya terdapat *actor. Actor* merupakan orang atau sistem yang menerima atau memberikan informasi dari sistem. Diagram ini sangat memperhatikan waktu atau terurut berdasarkan kejadian.**[2,p. 202-203]**

2.2.2.5. *Class Diagram*

Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, dan relasi-relasi. Diagram ini umum ditemui pada pemodelan sistem berorientasi objek. Meski sifatnya statis, sering pula memuat kelas-kelas aktif.**[2,p. 201]**

2.2.2.6. *Object Diagram*

Diagram ini memperlihatkan objek-objek serta relasi-relasi antar objek. Diagram objek memperlihatkan instantiasi statis dari segala sesuatu yang dijumpai dari pada diagram kelas.**[2,p. 201]**

2.2.2.7. *Deployment Diagram*

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*), memuat simpul-simpul atau node beserta komponen-komponen yang ada didalamnya.**[2,p. 207-208]**

2.2.3. UML (*Unified Modelling Language*)

UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak, *artifact* tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada

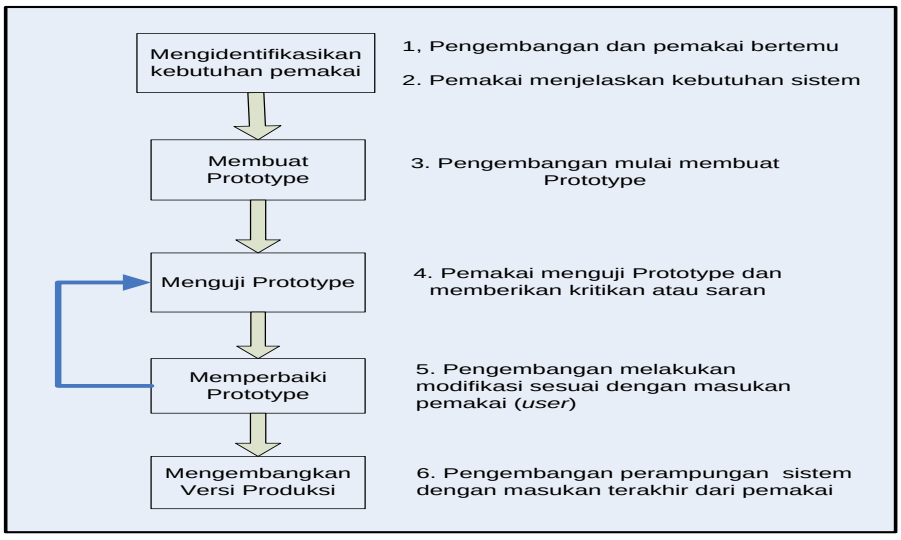
pemodelan bisnis dan sistem non perangkat lunak lainnya. Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi objek.**[2,p. 192-193]**

UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hamper dalam semua bidang yang membutuhkan pemodelan.

2.3. Pengembangan Sistem *Prototype*

Prototype merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat suatu program dengan cepat dan bertahap sehingga segera dapat dievaluasi oleh pemakai**.** Selain itu *prototype* juga membuat proses pengembangan sistem informasi menjadi lebih cepat dan lebih mudah.

Berikut ini merupakan mekanisme pengembangan sistem dengan menggunakan *prototype*:



Gambar 2.1. Mekanisme Pengembangan Sistem dengan *Prototype*

(Sumber: Pengenalan Sistem Informasi [10,p. 417])

Tahapan-tahapan *prototype* antara lain:

1. Identifikasi Kebutuhan Pemakai

   Pada tahap ini merupakan tahap awal dalam membangun sebuah aplikasi, di mana antara pemakai sistem (*users*) dan pengembang sistem bertemu. *Users* menjelaskan tentang kebutuhan sistem yang akan dibangun oleh pengembang sistem. Pada tahapan ini, penulis melakukan observasi dan wawancara untuk menentukan konten yang dibutuhkan pemakai dan merancang alur sistem.

2. Pembuatan *Prototype*

   Setelah menganalisa sistem yang akan dikembangkan serta kebutuhan-kebutuhan sistem untuk sistem yang akan dibangun, pengembang sistem mulai membuat *prototype*. Pembuatan ini meliputi: perancangan sistem yang akan dibangun, dan kemudian diimplementasikan dengan pembuatan *coding*, yaitu menerjemahkan hasil rancangan ke dalam bentuk bahasa pemrograman yang akan menjadi sebuah aplikasi yang diharapkan oleh *Users*. Pada tahap ini, penulis mulai membuat aplikasi sesuai dengan alur yang dibuat pada tahap sebelumnya.

3. Pengujian *Prototype*

   Setelah tahap pembuatan *prototype* selesai, kemudian pengembang sistem dan *Users* melakukan pengujian program agar program dapat digunakan

sesuai dengan kebutuhan, dan *users* memberikan saran atau masukan bila terdapat kekurangan pada program.

4. Perbaikan *Prototype*

   Pada tahap ini, pengembang sistem melakukan perbaikan dan modifikasi sesuai dengan masukan atau saran dari *user*.

5. Mengembangkan Versi Produksi

   Pada tahap ini, pengembang sistem menyelesaikan sistem yang telah dibuatnya sesuai dengan masukan atau saran terakhir dari pemakai sistem.

2.4. Pengertian Jaringan Komputer

Jaringan komputer merupakan gabungan antara teknologi komputer dan teknologi telekomunikasi. Gabungan ini menghasilkan pengolahan data yang didistribusikan, mencakup pemakaian *database, software* aplikasi dan peralatan *hardware* secara bersamaan.

2.4.1 Tipe-Tipe Jaringan Komputer

Jaringan komputer dapat dibedakan berdasarkan beberapa criteria, seperti luas area, media transmisi, pola operasi, dan sebagainya **[11,p. 108].** Berdasarkan luas areanya, maka jaringan komputer dapat dibedakan menjadi :

1. PAN *(Personal Area Network)*

   Jaringan komputer yang digunakan untuk komunikasi antara komputer perangkat (termasuk telepon dan asisten pribadi digital) dekat dari satu orang.

2. LAN *(Local Area Network)*

Suatu jaringan area lokal nirkabel yang menggunakan gelombang radio sebagai media transmisinya. Link terakhir yang digunakan adalah nirkabel, untuk memberi sebuah koneksi jaringan ke seluruh pengguna dalam area sekitar.

3. MAN *(Metropolitan Area Network)*

   Menggunakan metode yang sama dengan LAN, tetapi daerah cakupannya lebih luas. Daerah cakupan MAN dapat meliputi satu kampus, bahkan satu provinsi. Dapat dikatakan, MAN merupakan pengembangan dari LAN.

4. WAN *(Wide Area Network)*

   Merupakan jaringan komputer yang mencakup area yang besar. Sebagai contoh, yaitu jaringan komputer antarwilayah bahkan negara, atau dapat didefinisikan juga sebagai jaringan komputer yang membutuhkan *router* dan saluran komunikasi publik.

2.4.2. Topologi Jaringan

Topologi jaringan adalah hal yang menjelaskan hubungan geometris antara unsur-unsur dasar penyusun jaringan, yaitu *node, link*, dan *station*. Macam topologi jaringan ada 5 macam yaitu topologi bintang, cincin, bus, mesh, pohon.

1. Topologi Bintang

   Topologi bintang merupakan bentuk topologi jaringan yang berupa konvergensi dari node tengah ke setiap node atau pengguna.

2. Topologi Cincin

   Sesuai dengan namanya, jaringan yang menggunakan topologi ini dapat dikenali dari kabel *backbone* yang membentuk cincin. Setiap kali komputer

terhubung dengan kabel *backbone*. Setelah sampai pada komputer terakhir maka ujung kabel akan kembali dihubungkan dengan komputer pertama.

3. Topologi *Bus*

   Topologi *bus* banyak digunakan diawal penggunaan jaringan komputer dan dapat dikatakan sebagai topologi yang paling sederhana.

4. Topologi *Mesh*

   Topologi *mesh* dapat dikenali dengan hubungan *point to point* atau satu-satu ke setiap komputer. Setiap komputer terhubung ke komputer lain melalui kabel, dapat menggunakan kabel *coaxial, twisted pair*, bahkan serat optik. Topologi *mesh*sangat jarang diimplementasikan. Selain sangat rumit, juga sangat boros. Apabila jaringan komputer semakin banyak, maka isntalasi kabel jaringan akan semakin rumit juga.

5. Topologi Pohon

   Topologi pohondisebut juga topologi bintang-bus atau bintang/bus hybrid. Topologi pohon merupakan gabungan beberapa topologi bintang yang dihubungkan dengan topologi bus. Topologi pohon digunakan untuk menghubungkan beberapa LAN dengan LAN lain. Hubungan antar-LAN dilakukan via *hub*. Masing-masing *hub* dapat dianggap sebagai akar (*root*) dari masing-masing pohon (*tree*). Topologi *tree* dapat mengatasi kekurangan topologi *bus* yang disebabkan persoalan *broadcast traffic* dan kekurangan topologi *star* yang disebabkan oleh keterbatasan kapasitas *port hub*.

2.4.3. *Client Server*

Client server adalah jaringan komputer yang salah satu (boleh lebih) komputer difungsikan sebagai *server* atau induk bagi komputer lain. *Server* melayani komputer lain yang disebut *client*. Layanan yang diberikan dapat berupa akses *Web, e-mail, file*, atau yang lain. *Client server* banyak sekali dipakai pada *Internet*. Namun LAN atau jaringan lain pun dapat mengimplementasikan *client server*. Hal ini sangat bergantung pada kebutuhan masing-masing.

2.5. Perangkat Lunak Pendukung

Untuk membuat aplikasi *E-learning* tentu memerlukan perangkat lunak yang berfungsi sebagai pendukung pembuatan aplikasi *E-learning* tersebut. Dibawah ini akan dijelaskan perangkat lunak pendukung yang digunakan penulis.

2.5.1. Apache

Apache adalah sebuah nama *web server* yang bertanggung jawab pada request-response HTTP dan logging informasi secara detail (kegunaan dasarnya). Selain itu, Apache juga diartikan sebagai suatu *web server* yang kompak, modular, mengikuti standar protokol HTTP, dan tentu saja sangat digemari.

2.5.2. MySQL

MySQL merupakan *database* yang paling digemari dikalangan programer *web*, dengan alasan bahwa program ini merupakan *database* yang sangat kuat dan cukup stabil untuk digunakan sebagai media penyimpanan data. Sebagai *database* server yang mampu untuk memanajemen *database* dengan baik, MySQL terhitung merupakan *database* yang paling digemari dan paling banyak digunakan dibandingkan *database* lainnya.

Kemampuan lain yang dimiliki MySQL adalah mendukung Relasional Database Managemen Sistem (RDBMS), sehingga dengan kemampuannya ini MySQL akan dapat mampu menangani data-data sebuah perusahaan yang berukuran sangat besar hingga berukuran Giga Byte.**[13,p. 133-144]**

2.5.3. PHP

PHP (*Personal Home Page*) adalah sebuah bahasa pemograman yang berbentuk *scripting*. Sistem kerja dari program ini adalah sebagai *Interpreter* bukan sebagai *Compiler*.**[13,p. 201]**

2.5.4 PHPMyAdmin

PHPMyAdmin merupakan sebuah program bebas berbasis *web* yang dibuat menggunakan aplikasi PHP. Tujuan dibuatnya program ini adalah untuk mengakses *database* MySQL, intinya adalah untuk menjadi administator dari server MySQL. Dengan adanya program ini akan mempermudah dan mempersingkat kinerja kita, dengan kelebihan yang ada mengkibatkan para pengguna awam tidak harus mampu mengetahui sintaks-sintaks SQL dalam pembuatan database.**[13,p. 171]**

2.5.5. XAMPP

XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server* Apache, PHP dan MySQL secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi.

2.5.6 Codeigniter

Codeigniter adalah sebuah *web application framework* yang bersifat *open source* digunakan untuk membngun aplikasi php dinamis. Tujuan utama pengembangan codeigniter adalah untuk membantu developer untuk mengerjakan aplikasi lebih cepat daripada menulis semua code dari awal. Codeigniter menyediakan berbagai macam *library* yang dapat mempermudah dalam pengembangan. Codeigniter diperkenalkan kepada publik pada tanggal 28 februari 2006. Codeigniter itu sendiri dibangun menggunakan konsep *Model-View-Controller development pattern.* Codeigniter sendiri merupakan salah satu framework tercepat dibandingkan dengan framework lainnya. **[17]**