

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Bimbingan Belajar**

Bimbingan dapat diartikan sebagai suatu bantuan yang diberikan kepada orang lain yang bermasalah, dengan harapan orang tersebut dapat menerima keadaannya sehingga dapat mengatasi masalahnya dan mengadakan penyesuaian terhadap diri pribadi, lingkungan keluarga, sekolah, maupun masyarakat dalam bidang akademik.[1]

#### **2.2 Sistem**

Sistem adalah suatu kerangka dari prosedur-prosedur yang saling berhubungan, yang disusun sesuai dengan skema yang menyeluruh untuk melaksanakan suatu kegiatan atau fungsi utama dari perusahaan yang dihasilkan oleh suatu proses tertentu yang bertujuan untuk menyediakan informasi. sistem adalah suatu rangkaian yang terdiri dari dua atau lebih komponen yang saling berhubungan dan saling berinteraksi satu sama lain untuk mencapai tujuan dimana sistem biasa nya terbagi dalam sub system yang lebih kecil yang mendukung system yang lebih besar. [2]

#### **2.3 Administrasi**

Administrasi dalam arti luas adalah seluruh proses kerja sama antara dua orang atau lebih dalam mencapai tujuan dengan memanfaatkan sarana prasarana tertentu secara berdaya guna dan berhasil guna. [3]

#### **2.4 Auli Learning Center**

Auli Learning Center atau yang sering sering disebut dengan ALC merupakan salah satu tempat bimbingan belajar di daerah Jl.Sariasih blok 6 no 29B Kota Bandung Provinsi Jawa Barat. Bimbingan belajar sudah berkembang dikalangan pelajar sudah cukup lama dengan tujuan untuk memberikan pelajaran tambahan di luar sekolah agar pelajar lebih mengerti pelajaran yang diajarkan di dalam sekolah. [4]

## 2.5 Framework dan CMS

Pemrograman adalah kumpulan kelas (*class*) dan fungsi (*function* atau *method*) yang disusun secara sistematis berdasarkan kegunaan atau fungsionalitas tertentu untuk mempermudah pembuatan atau pengembangan suatu aplikasi. Sebagian besar framework yang beredar saat ini dibangun berdasarkan konsep *Object-Oriented Programming*(OOP). Beberapa contoh umum kelas yang disediakan framework adalah *class session*, *class database*, dan *class email*. CMS(Content Management System) adalah aplikasi web yg berisikan template untuk mengelola isi halaman web. penggunaan content management system tidak memerlukan pengetahuan pemrograman web yg handal karena proses instalasi dan cara penggunaannya sudah user friendly.

*Framework* adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class* dan *function*) yang dapat membantu *developer/programmer* dalam menangani berbagai masalah-masalah dalam pemrograman seperti koneksi ke database, pemanggilan variabel, file,dll sehingga developer lebih fokus dan lebih cepat membangun aplikasi. Bisa juga dikatakan *Framework* adalah komponen pemrograman yang siap *re-use* kapansaja, sehingga programmer tidak harus membuat skrip yang sama untuk tugas yang sama. Itu adalah salah satu contoh kecil, selebihnya *Framework* jauh lebih luas dari itu. [5]

## 2.6 CodeIgniter

CI adalah *framework* pemrograman umum Web yang bisa dipakai untuk mengembangkan semua jenis aplikasi Web. Dikarenakan lebih ringan dan dilengkapi dengan mekanisme *caching* yang canggih, CI sangat cocok untuk pengembangan aplikasi dengan lalu lintas tinggi, seperti portal, forum, dll [6]. Komponennya antara lain :

### 2.6.1 Model

Biasanya berhubungan langsung dengan *database* untuk memanipulasi data (*insert*, *update*, *delete*, *search*), menangani validasi dari bagian *controller*, namun tidak dapat berhubungan langsung dengan bagian *view*.

### 2.6.2 View

Merupakan bagian yang menangani *presentation logic*. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML yang diatur oleh *controller*. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian *model*.

### 2.6.3 Controller

Merupakan bagian yang mengatur hubungan antara bagian model dan bagian *view*, *controller* berfungsi untuk menerima permintaan dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

## 2.7 PHP

PHP adalah sebuah bahasa pemrograman skrip yang dirancang untuk membangun aplikasi web. Ketika dipanggil dari *web browser*, program yang ditulis dengan PHP akan di-parsing di dalam web server oleh *interpreter* PHP dan diterjemahkan ke dalam dokumen HTML, yang selanjutnya akan ditampilkan kembali ke *web browser*. Karena pemrosesan program PHP dilakukan di lingkungan *web server*, PHP dikatakan sebagai bahasa sisi server (server-side). Oleh sebab itu, seperti yang telah dikemukakan sebelumnya, kode PHP tidak akan terlihat pada saat user memilih perintah “View Source” pada web browser yang mereka gunakan. Selain menggunakan PHP, aplikasi web juga dapat dibangun dengan Java (JSP – JavaServer Pager dan Servlet), Perl, maupun ASP (Active Server Pages).[7]

## 2.8 MySQL

MySQL (*My Structure Query Language*) adalah salah satu *Database Management System* (DBMS) dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL dan lainnya. MySQL berfungsi untuk mengolah database menggunakan bahasa SQL. MySQL bersifat *open source* sehingga kita bisa menggunakannya secara gratis. Pemrograman PHP juga sangat mendukung/support dengan database MySQL. [8]

MySQL dikembangkan, dipasarkan, dan disokong oleh sebuah perusahaan Swedia bernama MySQL AB. RDBMS ini berada dibawah bendera GNU GPL sehingga termasuk produk *open source* dan sekaligus memiliki lisensi komersial. Apabila

menggunakan MySQL sebagai basis data dalam suatu situs web, maka MySQL tersebut gratis. Akan tetapi, jika ingin membuat RDBMS baru dengan MySQL kemudian menjualnya, wajib bertemu muka dengan lisensi komersial.

## 2.9 UML (*Unified Modeling Language*)

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan tek-teks pendukung. [9]

UML hanya berfungsi untuk melakukan pemodelan. Jadi, penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek. UML menyediakan sembilan jenis diagram yang dapat dikelompokkan, yaitu :

### 1. Use Case Diagram

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah informasi dan siapa saja yang berhak menggunakan fungsi – fungsi itu. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*, yaitu :

- a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat
- b. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor

### 2. Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. *Class* memiliki apa yang disebut atribut dan metode atau operasi, yaitu :

- a. Atribut merupakan variabel - variabel yang dimiliki oleh suatu kelas
- b. Operasi atau metode adalah fungsi – fungsi yang dimiliki oleh suatu kelas

### 3. Sequence diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek atau *message* yang dikirimkan dan diterima antar objek. Untuk menggambar *sequence diagram*, harus mengetahui objek – objek yang terlibat dalam sebuah *use case* beserta metode – metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

### 4. Collaboration Diagram

*Collaboration diagram* menggambarkan kolaborasi dinamis seperti *sequence diagrams*. Dalam menunjukkan pertukaran pesan, *collaboration diagrams* menggambarkan object dan hubungannya (mengacu ke konteks). Jika penekannya pada waktu atau urutan gunakan *sequence diagrams*, tapi jika penekanannya pada konteks gunakan *collaboration diagram*.

### 5. Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah proses bisnis yang ada pada perangkat lunak. *Activity diagram* menggambarkan aktivitas yang ada di sistem bukan yang dilakukan oleh aktor. *Activity diagram* digunakan untuk mendefinisikan hal – hal berikut :

- a. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
- b. Urutan tampilan dari sistem atau user interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
- c. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
- d. Rancangan menu yang ditampilkan pada perangkat lunak

### 6. Statechart Diagram

Statechart diagram menggambarkan perubahan status dari sebuah sistem atau objek. Jika *sequence diagram* digunakan untuk interaksi antar objek maka *statechart* digunakan untuk interaksi didalam sebuah objek. *Statechart diagram* cocok digunakan untuk menggambarkan alur interaksi pengguna dengan sistem.

## 7. Component Diagram

Component diagram dibuat untuk menunjukkan organisasi dan ketergantungan diantar kumpulan komponen sebuah sistem. *Component diagram* saat digunakan untuk memodelkan hal – hal berikut :

- a. *Source code* program perangkat lunak
- b. Komponen *executable* yang dilepas ke pengguna
- c. Basis data secara fisik
- d. Sistem yang haru beradaptasi dengan sistem lain
- e. *Framework* sistem merupakan kerangka kerja yang dibuat untuk memudahkan pengembangan dan pemeliharaan aplikasi

## 8. Deployment Diagram

*Deployment diagram* menunjukan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* memodelkan hal – hal berikut :

- a. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device, node* dan *hardware*
- b. Sistem *client/ server*
- c. Sistem terdistribusi murni
- d. Rekayasa ulang aplikasi

## 9. Object diagram

Object diagram merupakan struktur sistem dari segi penamaan objek dan jalannya objek sistem. Pada *object diagram* harus dipastikan semua kelas yang sudah didefinisikan pada class diagram haru dipakai objeknya, jika tidak pendefinisian kelas itu tidak dapat dipertanggungjawabkan. *Object diagram* berfungsi mendefinisikan nilai atau isi dari atribut tiap kelas.

### 2.10 HTML (*Hypertext Markup Language*)

Menurut Sibero (2012:19) HTML merupakan bahasa yang digunakan untuk menulis halaman web. Biasanya mempunyai ekstensi. htm, .html, atau .shtml. HTML tersusun atas tag-tag, digunakan untuk menentukan tampilan dari dokumen HTML yang diterjemahkan oleh browser. Tag HTML tidak *case sensitive*, jadi bisa menggunakan <HTML> atau <html> keduanya menghasilkan output yang sama.

HTML berawal dari bahasa SGML (Standar Generalized Markup Language) yang penulisannya disederhanakan. HTML dapat dibaca oleh berbagai macam platform, HTML juga merupakan bahasa pemrograman yang fleksibel dapat digabungkan dengan bahasa pemrograman lain seperti PHP, ASP, JSP, Java Script dan lainnya. Jika ada kesalahan pada penulisan HTML maka browser tidak akan memperlihatkan syntax error tetapi hanya tidak menampilkannya. HTML terus berkembang seiring perkembangan browser. [10]

Keterangan komponen-komponennya:

- **HTML.** Setiap dokumen HTML, biasanya diawali dan ditutup dengan tag HTML yang memberi tahu *browser* bahwa yang berada di dalam kedua tag adalah dokumen HTML
- **HEAD.** Bagian header dari dokumen HTML, berada di antara tag HEAD. Di dalam bagian ini biasanya dimuat tag *TITLE* yang menampilkan judul halaman pada bagian title milik browser. *Header* juga memuat tag META yang biasa digunakan untuk menentukan informasi tertentu mengenai dokumen HTML.
- **BODY.** Dokumen BODY digunakan untuk menampilkan text, image link, dan semua yang akan ditampilkan pada halaman web.

## 2.11 CSS (*Cascading Style Sheets*)

*Cascading Style Sheets* (CSS) adalah salah satu bahasa pemrograman desain web (style sheet language) yang mengontrol format tampilan sebuah halaman web yang ditulis dengan menggunakan bahasa penanda (*markup language*). Biasanya CSS digunakan untuk mendesain halaman HTML dan XHTML, tetapi sekarang bahasa pemrograman CSS bisa diaplikasikan untuk segala dokumen XML, termasuk SVG dan XVL.

CSS merupakan salah satu bahasa style web yang sekarang sedang populer dipakai pada berbagai aplikasi web. CSS dibuat untuk membuat style pada web lebih bervariasi dan mudah digunakan. [11]

CSS dibuat untuk memisahkan konten utama dengan tampilan dokumen, yang meliputi layout, warna, dan font. Pemisahan ini dapat meningkatkan daya akses konten pada web, menyediakan lebih banyak fleksibilitas dan kontrol dalam spesifikasi dari

sebuah karakteristik dengan sebuah tampilan, memungkinkan untuk membagi banyak halaman untuk sebuah formatting dan mengurangi kerumitan dalam penulisan kode dan struktur dari konten.

### **2.12 SMS Gateway**

SMS Gateway merupakan pintu gerbang bagi penyebaran informasi dengan menggunakan SMS. Dapat menyebarkan pesan ke banyak nomor secara otomatis dan cepat yang langsung terhubung dengan database nomor-nomor ponsel saja, tanpa harus mengetik ratusan nomor dan pesan di ponsel. Karena semua nomor akan diambil secara otomatis dari database tersebut. [12]

### **2.13 Gammu**

Gammu merupakan salah satu tools untuk mengembangkan aplikasi SMS Gateway. Gammu bisa dikatakan sebagai "Aktor Utama", karena komponen inilah yang menjembatani pentransferan data-data SMS dari handphone atau mobile modem ke komputer atau sebaliknya. Kelebihan Gammu dari tool SMS gateway lainnya adalah [13]

- a. Gammu dapat dijalankan di Windows maupun Linux
- b. Banyak *device* atau ponsel yang kompatibel dengan gammu
- c. Gammu menggunakan database MySQL dan dapat menggunakan aplikasi desktop dan *interface web-based*
- d. Gammu dapat membantu menggunakan fitur-fitur yang ada pada ponsel dengan lebih efisien.