BAB 2
TINJAUAN PUSTAKA

2.1. Tinjauan Sekolah

Pada sub bab ini membahas peninjauan terhadap tempat penelitian yaitu Sekolah Peradaban Al-Kamil Cianjur.

2.1.1. Sejarah Sekolah

Sebelum berdiri Sekolah Peradaban Al-Kamil ini, anak asuh Yayasan Mizan Amanah yang tinggal di asrama menimba ilmu pendidikan atau bersekolah di luar. Dan ini sering menjadi kendala dalam pembentukan akhlak, kreativitas, serta kemampuan anak itu sendiri, sebab anak–anak banyak terpengaruh lingkungan luar. Maka untuk mewujudkan generasi unggul, Sekolah Peradaban Al-Kamil bertekad untuk mendirikan sebuah lembaga pendidikan yang kelak akan mencetak generasi yang penuh manfaat.

Dalam perjalanan waktu dan rahmat serta karunia Allah SWT, Alhamdulillah lewat tangan para dermawan, pada tahun 2011 terbebaskannya lahan di daerah Ciranjang, Cianjur, Jawa Barat. Penerimaan murid mulai dilaksanakan pada 2012. Hingga 2015 kini, 131 murid yang terdiri dari anak-anak yatim, dhu'afa, dan berprestasi sudah berkesempatan belajar di Sekolah Peradaban Al-Kamil ini. Di lahan tersebut Insya Allah sedang didirikan Sekolah Peradaban Al-Kamil, yaitu sekolah bagi anak–anak yatim, kaum dhu’afa dan anak-anak yang berprestasi dengan beasiswa penuh dari Yayasan Mizan Amanah. Sekolah yang menargetkan muridnya memiliki pondasi iman yang kuat, adab yang baik, bekal tahfidz Qur'an dan memiliki keilmuan yang cakap sesuai dengan kompetensi peradaban.

2.1.2. Visi dan Misi Sekolah

Visi Sekolah

“Menjadi lembaga pendidikan terbaik bertaraf internasional dalam membentuk generasi unggul dan mewujudkan manusia yang hakiki”

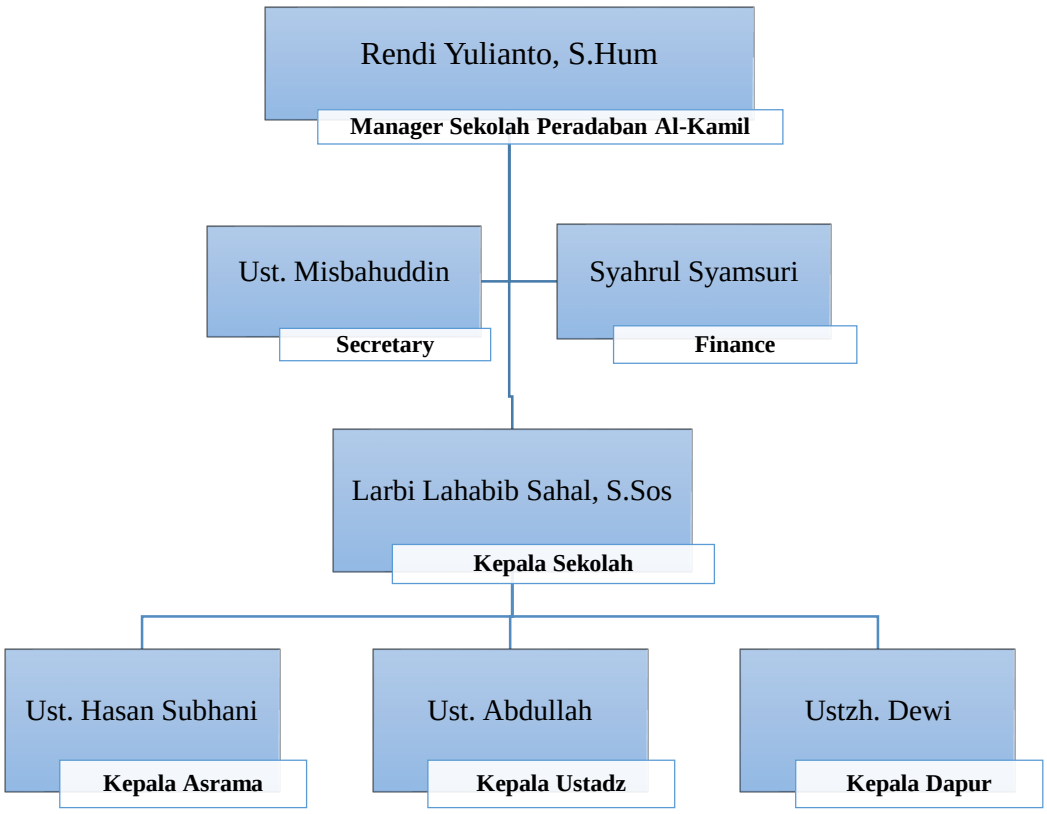
Misi Sekolah

a. Membangun sistem pendidikan yang terpadu, profesional dan dinamis.
b. Menyediakan sarana dan prasarana pendidikan yang bermutu dan berkualitas.
c. Mencetak lulusan yang memiliki kompetensi dan berprestasi di bidangnya.
d. Menciptakan lingkungan yang bertauhid dan syar'i.
e. Mengembangkan kapabilitas pendidik, tenaga kependidikan dan tim manajemen secara berkesinambungan.

Motto

"Amanah, Cerdas, Mandiri, Berakhlak Mulia"

2.1.3. Struktur Organisasi Sekolah



Gambar 2.1 Struktur Organisasi Sekolah Peradaban Al-Kamil

Tugas dan fungsi dari struktur organisasi pada Sekolah Peradaban Al-Kamil terdapat pada Tabel 2.1 berikut ini

Tabel 2.1 *Job Description*

Manager	Memutuskan dan menentukan peraturan dan kebijakan tertinggi sekolah Mengangkat dan memberhentikan ustadz atau staf sekolah Memberi Beasiswa atau *reward* untuk santri berprestasi
Secretary	Mengatur aktivitas sekolah Menjadi mediator manager dengan ustadz atau staf Menyiapkan arsip-arsip penting sekolah Menyiapkan dokumen dan persentasi
Finance	Membuat laporan pertanggung jawaban keuangan sekolah Mencatat seluruh transaksi keuangan yang berkaitan dengan sekolah Membuat pembukuan keuangan
Kepala Sekolah	Bertanggung jawab dalam memimpin dan menjalankan sekolah Mengawasi tugas ustadz dan staf Menyetujui anggaran sekolah Menyampaikan laporan kepada *Manager*
Kepala Asrama	Bertanggung jawab dalam semua urusan asrama di sekolah Menyampaikan laporan mengenai asrama kepada Kepala Sekolah
Kepala Ustadz	Bertanggung jawab atas semua ustadz dan pembelajaran Menyampaikan laporan mengenai ustadz dan pembelajaran kepada Kepala Sekolah
Kepala Dapur	Bertanggung jawab dalam semua urusan dapur di sekolah Menyampaikan laporan mengenai dapur kepada Kepala Sekolah

2.2. Landasan Teori

2.2.1. Pengertian Al-Qur'an

Al-Qur'an menurut bahasa berarti bacaan atau yang dibaca. Menurut istilah, Al-Qur'an adalah wahyu Allah *Subhanahu wa ta'ala* yang diturunkan kepada nabi Muhammad *Shallallahu 'alaihi wasallam* melalui malaikat Jibril *'Alaihissalam* sebagai petunjuk bagi umat manusia. Al-Qur'an diturunkan untuk menjadi pegangan bagi mereka yang ingin mencapai kebahagian dunia dan akhirat. Al-Qur'an menggunakan bahasa Arab dan merupakan mukjizat bagi rasul. Sebagian besar ayat-ayat Al-Qur'an diturunkan di kota Mekah dan kota Madinah. Isi yang terkandung dalam Al-Qur'an terdapat 6236 ayat 114 surat dan 30 juz. [2] Perlu diketahui bahwa kebanyakan metode menghafal Al-Qur'an yang diterapkan adalah membaca dan mendengarkan secara berulang-ulang (*muraja'ah*), namun hal tersebut banyak menemui kendala karena terbatasnya peralatan yang disediakan berupa mushaf Al-Qur'an. [3]

Dalam menghafal Al-Qur'an banyak metode yang dikembangkan, namun setiap metode harus disesuaikan dengan kondisi dan situasi. Metode juga bisa memberikan bantuan kepada para penghafal untuk mengurangi kesusahannya dalam menghafal Al-Qur'an. Setiap kesukaran dan kesusahan yang akan dihadapi oleh penghafal merupakan suatu tantangan yang wajib dilalui agar terdorong lebih giat dan bersungguh-sungguh dalam menghafalnya. Walaupun banyak halangan dan rintangan yang dialami oleh penghafal, pada dasarnya telah ada metode-metode menghafal Al-Qur'an sebagaimana yang pernah diterapkan Rasulullah kepada para sahabatnya. Salah satu metode yang diajarkan Rasulullah kepada para sahabat adalah mengulang-ulang doa atau ayat-ayat Allah di hadapan Rasulullah Saw sementara beliau menyimak bacaan para sahabat. [4]

Dengan dibuatnya Aplikasi Qur'an *Memorization Tool* (QMT) di Sekolah Peradaban Al-Kamil Cianjur ini diharapkan akan mempermudah menghafal Al-Qur'an dengan adanya metode-metode menghafal yang akan membantu.

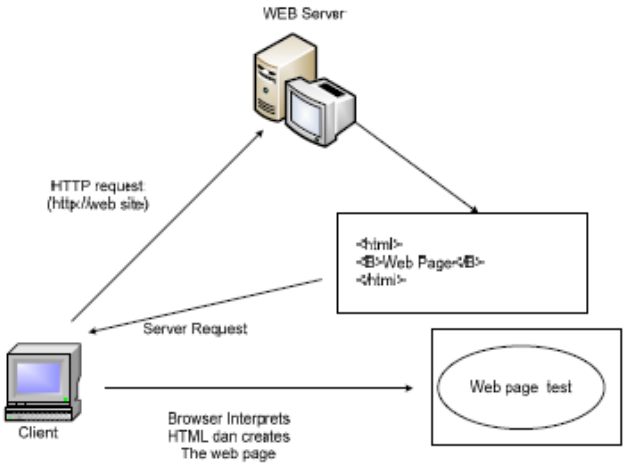
2.2.2. Pengertian *Web Application*

Web application dikenal sebagai aplikasi yang diakses melalui *web browser* dan melalui jaringan seperti internet atau intranet. Kemampuan untuk memperbarui dan memelihara aplikasi web tanpa harus mendistribusikan dan menginstal perangkat lunak pada kemungkinan ribuan komputer klien merupakan keunggulan teknologi ini, selain juga untuk *cross-platform compatibility*. Termasuk aplikasi *web common webmail*, penjualan ritel online, online pelelangan, wiki dan banyak fungsi lainnya. [5]

Banyak situs internet yang memiliki halaman dengan sifat statis seperti profil perusahaan, artikel, dan keterangan-keterangan lain. Situs ini mempunyai dokumen dengan teks yang sederhana, image dan *hyperlinks* ke dokumen yang dimilikinya. Untuk mengembangkan situs yang bersifat statis, kita menggunakan teknologi client side. HTML dan *Cascading Style Sheet* (CSS) dapat digunakan untuk mengatur struktur dan menampilkan halaman isi. Seandainya ingin diperindah dapat ditambahkan *script* yang sifatnya *client side*, seperti JavaScript, Jscript ataupun VBScript.

Dengan berkembangnya internet, situs yang ada di internet tidak hanya berfungsi untuk mempresentasikan content tetapi cenderung berupa aplikasi yang kebanyakan terhubung ke suatu basis data. Pada tahapan ini situs akan bersifat dinamis, karena content yang dipresentasikan akan bervariasi dan berubah-ubah sesuai dengan data yang diminta dan action dari user. Untuk mengembangkan situs yang dinamis diperlukan teknologi *server side* seperti PHP, ASP, Perl dan CGI yang lain. Dengan teknologi *server side* kita dapat mengembangkan suatu aplikasi berbasis internet yang dapat mengahsilkan dan menampilkan content secara dinamis. [6]

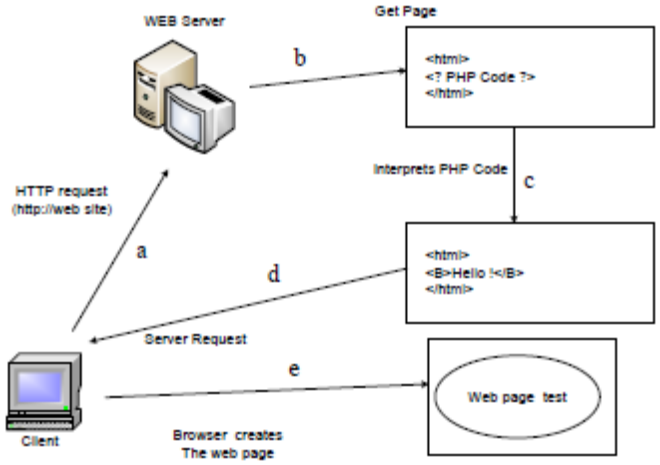
Pada Gambar 2.2 diperlihatkan cara kerja situs dengan content yang statis. Pada saat terjadi suatu *request* dari *browser*, *server web* selanjutnya membaca *request* yang dikirim oleh *browser*, mencari dan menemukan halaman di *server*, dan mengirim halaman yang diminta melalui internet ke *browser*. Selanjutnya halaman tersebut ditampilkan di *browser*.



Gambar 2.2 Web yang Statis

Pada Gambar 2.3 diperlihatkan cara kerja situs dengan *content* yang dinamis. Kronologis gambar adalah sebagai berikut : [6]

a. *Browser* (*client*) melakukan *request* halaman web ke WEB Server.
b. WEB Server kemudian membaca request yang dikirim dari *browser*, mencari dan menemukan halaman di *server*.
c. Menterjemahkan perintah yang diberikan oleh bahasa program *server-side* menjadi halaman HTML.
d. Setelah konversi ke halaman HTML selesai kemudian mengirim halaman yang diminta ke *browser*.
e. Setelah halaman HTML yang dikirim dari WEB server sampai, selanjutnya *browser client* akan menampilkan halaman web.



Gambar 2.3 Web yang Dinamis

Dengan demikian aplikasi ini akan dibuat berbasis web agar mempermudah penggunaan, karena tidak memerlukan instalasi, mempermudah pengaksesan bagi ustadz tamu dan *manager* dari luar sekolah.

2.2.3. Internet

Dalam menjalankan aplikasi ini dibutuhkan jaringan internet sebagai koneksinya. Berikut ini penjelasan dari jaringan internet :

2.2.3.1. Pengertian Internet

Definisi internet adalah suatu jaringan komputer global yang terbentuk dari jaringan-jaringan komputer lokal dan regional yang memungkinkan komunikasi data antar komputer yang terhubung ke jaringan tersebut. [7] Menurut Salahuddien, internet adalah berbagai jaringan komputer di seluruh dunia yang saling terhubung tanpa mengenal batas teritorial, hokum dan budaya. Secara fisik dianalogikan sebagai jaring laba-laba (*The Web*) yang menyelimuti bola dunia dan terdiri dari titik-titik (*node*) yang saling berhubungan.

Lebih jauh Salahuddien menjelaskan *node* bisa berupa komputer, jaringan lokal atau peralatan komunikasi, sedangkan garis penghubung antar simpul disebut sebagai tulang punggung (*backbone*) yaitu media komunikasi terestrial (kabel, serat optik, *microwave*, radio link) maupun satelit. *Node* terdiri dari pusat informasi dan *database*, peralatan komputer dan perangkat interkoneksi jaringan serta peralatan

yang dipakai pengguna untuk mencari, menempatkan dan atau bertukar informasi di Internet.

2.2.3.2. Sejarah Perkembangan Internet

Internet awalnya merupakan suatu rencana dari Departemen Pertahanan Amerika Serikat (US *Departement of Defense*) pada sekitar tahun 1960. Dimulai dari suatu proyek yang dinamakan ARPANET atau *Advanced Research Project Agency Network*. Beberapa universitas di Amerika Serikat diantaranya UCLA, Stanford, UC Santa Barbara dan University of Utah, diminta bantuan dalam mengerjakan proyek ini dan awalnya telah berhasil menghubungkan empat komputer di lokasi universitas yang berbeda tersebut. [7] [8]

Perkembangan ARPANET ini cukup pesat jika dilihat perkembangan komputer pada saat itu. Sebagai gambarannya pada tahun 1977, ARPANET telah menghubungkan lebih dari 100 *mainframe* komputer dan saat ini terdapat sekitar 4 juta *host* jaringan yang terhubung pada jaringan ini. Karena perkembangannya sangat pesat, jaringan komputer ini tidak dapat lagi disebut sebagai ARPANET karena semakin banyak komputer dan jaringan-jaringan regional yang terhubung. Konsep ini yang kemudian berkembang dan dikenal sebagai konsep *Internet working* (jaringan antar jaringan). Oleh karena itu istilah Internet menjadi semakin popular, dan orang menyebut jaringan besar komputer tersebut dengan istilah Internet. [7]

Pada era 2000, perkembangan Internet dan jaringannya naik secara eksponensial. Dalam waktu kurang dari 10 tahun, tingkat pertumbuhan Internet melebihi densitas teknologi telekomunikasi (telepon, radio dan TV). Lambat laun *content* dari internet mengintegrasikan teknologi telepon (VOIP – *Voice Over* IP, telepon berbasis jaringan Internet) dan WebTV. Sehingga *backbone* dan teknologi Internet saat ini menjadi penting untuk dikembangkan karena dipandang sebagai masa depan telekomunikasi dunia. [7]

Berdasarkan survei AC Nielsen di Indonesia pada Juni 1999 diketahui jumlah pengguna Internet telah mencapai 800 ribu orang dan oleh Priyatmo (Kompas 12 Maret 2000) diprediksikan tumbuh 20% per tahun. APJII (Asosiasi

Penyelenggara Jasa Internet Indonesia) memprediksikan angka 1,5 juta pada tahun 2000 dan 15 juta pada 2005. 25% diantaranya merupakan pelanggan personal ISP (*Internet Service Provider*) sedang sisanya akan mengakses Internet dari Warnet atau Sekolah dan Kantor. Hal ini sesuai dengan kenyataan terjadinya *booming* bisnis Warnet sejak tahun 1999. [7]

Dengan demikian aplikasi ini membutuhkan internet, karena aplikasi ini harus dapat diakses dari luar sekolah dan aplikasi ini juga memerlukan internet untuk melakukan *Video Call.*

2.2.4. *Video Call*

Dalam aplikasi ini dibutuhkan *Video Call* sebagai media untuk melakulan ujian, setoran dan *muraja'ah*. Berikut ini penjelasan dari *Video Call* :

2.2.4.1. Pengertian *Video Call*

Video Call adalah telepon dengan layar *video* dan mampu menangkap *video* (gambar) sekaligus suara yang ditransmisikan. Fungsi *video call* sebagai alat komunikasi antara satu orang dengan orang yang lainnya secara waktu nyata (*real-time*). Saat ini *video call* sangat berguna bagi orang tuli dan bisu, karena melalui *video call*, komunikasi bisa dilakukan dengan menggunakan bahasa isyarat melalui layanan video tersebut. Begitu juga untuk orang-orang yang berada di tempat lain yang jauh dan ingin berkomunikasi dengan orang yang berada ditempat lain yang jauh pula. *Video call* dapat digunakan sebagai alat yang dapat menyalurkan gambar serta suara dalam bentuk *video* sehingga terlihat seperti nyata. [9]

2.2.4.2. Sejarah *Video Call*

Sekitar dua tahun setelah telepon pertama kali diciptakan oleh Alexander Graham Bell dan dipatenkan di Amerika serikat merupakan konsep awal dari gabungan *videophone* yang disebut *telephonescope*. Hal ini juga disebutkan dalam berbagai awal karya-karya fiksi ilmiah seperti *Le Vingtième Siecle: La vie electrigue* (*The 20th Century*: The Electrical Hidup) dan karya-karya lain yang ditulis oleh Albert Robida, dan juga membuat sketsa di berbagai kartun oleh George du Maurier sebagai fiksi penemuan Thomas Edision. Salah satu sketsa tersebut

diterbitkan pada 9 Desember 1878 di majalah Punch. Diantara fungsi-fungsi lain, *videophone* memungkinkan pedagang untuk mengirim gambar dagangan mereka kepada pelanggan mereka. Dalam era sebelum munculnya lembaga penyiaran publik, listrik dipandang sebagai perangkat tambahan untuk pesawat telepon, sehingga menciptakan konsep *videophone*. [8]

Videophone perlahan-lahan masuk ke masyarakat umum setelah 1950, meskipun ‘telepon video’ mungkin memasuki masa setelah ‘video’ diciptakan pada tahun 1935. Sebelum waktu itu tampaknya tidak ada istilah-istilah standar untuk ‘video telepon’, dengan ekspresi seperti ‘melihat sound sistem televisi’, ‘visual radio’ dan hampir 20 orang lain yang menggunakannya untuk menggambarkan penggabungan telegraf dan telepon, Televisi dan radio teknologi yang digunakan pada awal percobaan. Satu teknologi pendahulu *videophone* adalah mesin *teleostereograph* dikembangkan oleh AT & T Bell Labs pada tahun 1920, yang merupakan pendahulu dari *today’s faks* (faksimili) mesin. Di tahun 1927 AT & T telah menciptakan elektro-mekanis *videophone* yang beroperasi pada 18 frame per detik dan menempati setengah ruangan yang penuh dengan lemari peralatan. [8]

Seiring dengan perkembangannya, nama *videophone* sudah tidak dipakai lagi, nama *video call* menjadi lebih populer setelah telepon menjadi bagian yang tidak perlu terintegrasi secara fisik dalam satu perangkat. [8]

Dengan demikian *Video Call* akan diterapkan pada proses ujian, setoran dan *muraja’ah* agar dapat membantu ustadz.

2.2.5. Perangkat Lunak Pembangun Sistem

Dalam membangun aplikasi ini menggunakan PHP sebagai bahasa pemrograman untuk mengimplementasikan fungsi-fungsi yang dirancang, serta dibantu dengan MYSQL yang digunakan sebagai penyimpanan data atau database dari aplikasi yang dibangun, *Codeigniter* sebagai *framework* PHP, XAMPP sebagai server untuk menjalanan website dan *Sublime Text* sebagai editor yang digunakan untuk pengkodean PHP. Berikut ini penjelasan dari perangkat lunak yang digunakan:

2.2.5.1. Pengertian PHP

PHP adalah bahasa pemrograman *script* yang paling banyak dipakai saat ini. PHP banyak dipakai untuk memrogram situs web dinamis, walaupun tidak tertutup kemungkinam untuk digunakan untuk pemakaian lain. PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP bernama FI (*Form Interpreted*). Pada saat tersebut PHP adalah sekumpulan *script* yang digunakan untuk mengolah data *form* dari web. Perkembangan selanjutnya adalah Rasmus melepaskan kode sumber tersebut dan menamakannya PHP/ FI, pada saat tersebut kepanjangan dari PHP/ FI adalah *Personal Home Page*/ *Form Interpreter*. Dengan pelepasan kode sumber ini menjadi *open source*, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP. Pada November 1997, dirilis PHP/ FI 2.0. Pada rilis ini interpreter sudah diimplementasikan dalam C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/ FI secara signifikan. [6]

Pada tahun 1997, sebuah perusahaan bernama Zend, menulis ulang interpreter PHP menjadi lebih bersih, lebih baik dan lebih cepat. Kemudian pada Juni 1998 perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan nama rilis tersebut menjadi PHP 3.0.Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai. Versi ini banyak dipakai sebab versi ini mampu dipakai untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan proses dan stabilitas yang tinggi.Pata Juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi mutakhir dari PHP. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Dalam versi ini juga dikenalkan model pemrograman berorientasi objek baru untuk menjawab perkembangan bahasa pemrograman kearah pemrograman berorientasi objek. [6]

PHP merupakan bahasa pemrograman web, penulisan *script* PHP dapat dijadikan satu dengan kode html dan disimpan dalam satu *file* yang sama ataupun berdiri sendiri dengan nama *file* berekstensi *.php. Ciri khas PHP diawali dengan tanda “<?php“ dan diakhiri dengan “?>”. PHP memiliki fasilitas untuk melakukan

koneksi ke *database*. Script program PHP disimpan dan dijalankan di sisi *server* sehingga keamanan data lebih terjamin. [6]

Dengan demikian PHP digunakan dalam aplikasi ini karena PHP paling banyak digunakan dan tidak sulit untuk penerapan pada *server.*

2.2.5.2. Pengertian MySQL

MySQL merupakan aplikasi *database server*. Perkembangannya disebut SQL yang merupakan kepanjangan dari *Structured Query Language*. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah *database*. MySQL dapat digunakan untuk membuat dan mengelola database beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah, dan menghapus data yang berada dalam database. [6]

MySQL adalah sebuah sistem manajemen *database* relasi (*relation database management system*) yang bersifat *open source.* MySql merupakan buah pikiran dari Michael "Monty" Widenius, Daveid axmark dan Allan Larson yang dimulai tahun 1995.Mereka bertiga kemudian mendirikan perusahaan bernama MySQL AB di Swedia. [6]

MySQL versi 1.0 rilis pada Mei 1996 dan penggunaannya hanya terbatas di kalangan perusahaan saja. Barulah pada bulan Oktober 1996, MySQL versi 3.11.0 di rilis ke masyarakat luas. MySQL menggunakan bahasa standar SQL (*Structure Query Languange*) sebagai bahasa interaktif dalam mengelola data. MySQL memiliki kinerja, kecepatan proses dan ketangguhan yang tidak kalah dibanding *database-database* besar lainnya yang komersil seperti ORACLE, Sysbase, Unify dan sebagainya. MySQL dapat berjalan di atas banyak sistem operasi Linux, Windows, Solaris, FreeBSD, Mac OS X, dan lain sebagainya. [6]

Dengan demikian MySQL sebagai database yang digunakan dalam aplikasi ini karena sangat cocok diterapkan pada PHP.

2.2.5.3. Pengertian XAMPP

XAMPP adalah *software* web *server apache* yang di dalamnya tertanam *server* MySQL yang didukung dengan bahasa pemrograman PHP untuk membuat website yang dinamis. XAMPP sendiri mendukung dua system operasi yaitu

windows dan Linux. Untuk linux dalam proses penginstalanny menggunakan *command line* sedangkan untuk windows dalam proses penginstalannya menggunakan interface grafis sehingga lebih mudah dalam penggunaaan XAMPP di Windows di banding dengan Linux.

Apache merupakan web *server* yang digunakan untuk menampilkan website di internet seperti menggunakan Mozilla fire fox, Google Crome, IE, Safari, dan lain-lain berdasarkan kode-kode yang di tulis di dalam website tersebut baik menggunakan bahasa pemrograman HTML maupun PHP yang mengambil suatu *database* yang dibangun di MySQL, sehingga terbentuklah sebuah website yang dapat di lihat di Mozilla fire fox dan kawan-kawannya. *Apache* sendiri bersifat *open source* sehingga dapat digunakan oleh siapa saja dan dikembangkan oleh siapa saja tentunya bagi yang mampu mengembagkannya.

Dengan demikian XAMPP dipilih menjadi server untuk menjalankan dan mengembangkan aplikasi ini, karena lengkap dan memenuhi semua perangkat lunak yang digunakan.

2.2.5.4. Pengertian *Codeigniter*

Codeigniter adalah 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 yang dibuat menjadi semakin teratur dan rapi. Dengan demikian *developer* dapat fokus pada fitur-fitur apa yang dibutuhkan aplikasi dengan membuat kode program seminimal mungkin.

Codeigniter pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc. sebuah perusahaan yang memproduksi sebuah CMS (*content management system*) yang cukup handal, yaitu *expression engine*. Saat ini, *codeigniter* dikembangkan dan dimaintain oleh *expression engine development team*.

Dengan demikian *Codeigniter* digunakan sebagai *framework* untuk membuat aplikasi ini, karena dengan menggunakan *codeigniter* tidak perlu lagi

membuat struktur dan fungsi-fungsi kewangka awal, sehingga akan mempercepat dalam proses pembuatan aplikasi ini.

2.2.5.5. Pengertian *Sublime Text*

Sublime Text itu adalah salah satu *text editor* yang biasa digunakan oleh para *programmer*, khususnya *Web Developer*. Bisa diibaratkan sebagai senjata-nya *Web Developer*. Setiap *web developer* wajib untuk menggunakan *Sublime Text* sebagai senjata hariannya. Dengan banyaknya fitur-fitur yang bagus menjadikan *Sublime Text* pilihan para Web Developer. Beberapa fiturnya sebagai berikut :

1. *Multiple Selection,* fitur untuk merubah banyak text dengan satu kali ketik.
2. *Find in Project,* fitur untuk mencari file dalam project
3. *Go to line,* fitur untuk berpindah baris dengan cepat
4. *Package Control dan Themes,* fitur untuk menambahkan fitur dan mengganti tema

Dengan demikian *Sublime Text* digunakan sebagai *editor* untuk membuat aplikasi ini, karena dengan beberapa fitur yang sangat membantu untuk mempercepat dalam pengkodean.

2.2.6. Perancangan Sistem

Dalam membangun aplikasi ini perancangan yang digunakan adalah perancangan berorientasi objek dengan menggunakan pemodelan *Unified Modelling Language* (UML), serta pemrograman berorientasi objek. Berikut ini penjelasan dari perancangan sistem yang digunakan:

2.2.6.1. Pengertian *Unified Modelling Language* (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman

apapun. Tetapi karena UML juga menggunakan class dan operation dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C.

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML syntax mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari tiga notasi yang telah ada sebelumnya: Grady Booch OOD (*Object-Oriented Design*), Jim Rumbaugh OMT (*Object Modeling Technique*), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*). [10]

UML ini menyediakan 10 macam diagram untuk memodelkan aplikasi berorientasi objek, yaitu : [11]

1. *Use Case Diagram* untuk memodelkan proses bisnis.
2. *Conceptual Diagram* untuk memodelkan konsep-konsep yang ada di dalam aplikasi.
3. *Sequence Diagram* untuk memodelkan pengiriman pesan (*message*) antar objek.
4. *Collaboration Diagram* untuk memodelkan interaksi antar objek.
5. *State Diagram* untuk memodelkan perilaku objek di dalam sistem.
6. *Activity Diagram* untuk memodelkan perilaku userdan objek di dalam sistem.
7. *Class Diagram* untuk memodelkan struktur kelas.
8. *Objek Diagram* untuk memodelkan struktur objek.
9. *Component Diagram* untuk memodelkan komponen objek.
10. *Deployment Diagram* untuk memodelkan distribusi aplikasi.

Berikut akan dijelaskan 6 macam diagram yang digunakan dalam perancangan aplikasi ini, yaitu *use case diagram, activity diagram, class diagram, sequence diagram, deployment diagram,* dan *component diagram.*

1. *Use case diagram*

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas *diagram* untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi.

Use case merepresentasikan operasi-operasi yang dilakukan oleh *actor*.*Use case* digambarkan berbentuk elips dengan nama operasi dituliskan didalamnya. *Actor* yang melakukan operasi dihubungkan dengan garis lurus ke *use case.* [11]

2. *Activity Diagram*

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state* diagram khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan *behaviour* internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. [11]

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan *behaviour* pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. *Activity diagram* dapat dibagi menjadi beberapa object swimlane untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu. [11]

3. *Class Diagram*

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi*). Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. [11]

Class memiliki tiga area pokok : [11]

1. Nama (dan *stereotype*)
2. Atribut
3. Metoda

Atribut dan metoda dapat memiliki salah satu sifat berikut : [11]

1. *Private* , tidak dapat dipanggil dari luar class yang bersangkutan
2. *Protected* , hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya
3. *Public* , dapat dipanggil oleh siapa saja

Class dapat merupakan implementasi dari sebuah *interface*, yaitu *class* abstrak yang hanya memiliki metoda. *Interface* tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah *class*. Dengan demikian *interface* mendukung resolusi metoda pada saat *run-time*. Sesuai dengan perkembangan class model, class dapat dikelompokkan menjadi package. [11]

4. *Sequence Diagram*

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai *respons* dari sebuah *event* untuk menghasilkan *output* tertentu.

Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada *fase* desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah *message*. [11]

5. *Deployment Diagram*

Deployment diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisikal Sebuah node adalah server, *workstation,* atau piranti keras lain yang digunakan untuk men- *deploy* komponen dalam lingkungan sebenarnya. Hubungan antar node (misalnya TCP/IP) dan *requirement* dapat juga didefinisikan dalam diagram ini. [11]

6. *Component Diagram*

Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi *code*, baik berisi *source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time*, *link time*, maupun *run time*. Umumnya komponen terbentuk dari beberapa *class* dan/atau *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain. [11]

2.2.6.2. Pemrograman Berorientasi Objek

Analisis dan desain berorientasi objek adalah cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dab perilaku dalam suatu entitas. Model berorientasi objek bermanfaat

untuk memahami masalah, komunikasi dengan ahli aplikasi, pemodelan suatu organisasi, meyiapkan dokumentasi serta perancangan program dan basis data. Pertama-tama suatu model analisis dibuat untuk menggambarkan aspek dasar dari domain aplikasi, dimana model tersebut berisi objek yang terdapat dalam domain aplikasi termasuk deskripsi dari keterangan objek dan perilakunya. [11]

Secara spesifik, pengertian berorientai objek berarti bahwa mengorganisasi perangkat lunak sebagai kumpulan dari objek tertentu yang memiliki struktur data dan perilakunya. Hal ini yang membedakan dengan pemograman konvensional dimana struktur data dan perilaku hanya berhubungan secara terpisah. Terdapat beberapa cara untuk menentukan karateristik dalam pendekatan berorientasi objek, tetapi secara umum mencakup empat hal, yaitu identifikasi, klasifikasi, *polymorphism* (polimorfisme) dan *inheritance* (pewarisan).

Metodologi pengembangan sistem berorientasi objek mempunyai tiga karateristik utama, yaitu : [11]

a. Encapsulation

Encapsulation (pengkapsulan) merupakan dasar untuk pembatasan ruang lingkup program terhadap data yang diproses. Data dan prosedur atau fungsi dikemas dalam bersama-sama dalam suatu objek, sehingga prosedur atau fungsi lain dari luar tidak dapat mengaksesnya. Data terlindung dari prosedur atau objek lain kecuali prosedur yang berada dalam objek itu sendiri.

b. Inheritance

Inheritance (pewarisan) adalah teknik yang menyatakan bahwa anak dari objek akan mewarisi atribut dan metoda dari induknya langsung. Atribut dan metoda dari objek induk diturunkan kepada anak objek, demikian seterusnya. Pendefinisian objek dipergunakan untuk membangun suatu hirarki dari objek turunannya, sehingga tidak perlu membuat atribut dan metoda lagi pada anaknya, karena telah mewarisi sifat induknya.

c. Polymorphism

Polymorphism (polimorfisme) yaitu konsep yang menyatakan bahwa sesuatu yang sama dapat mempunyai bentuk dan perilaku berbeda. Polimorfisme mempunyai arti bahwa operasi yang sama mungkin mempunyai perbedaan dalam kelas yang berbeda.