

BAB II

LANDASAN TEORI

Pada bab ini akan dijelaskan konsep-konsep dan teori-teori secara singkat yang berkaitan dengan inti perancangan perangkat. Konsep-konsep dan teori-teori yang akan dijelaskan adalah pengertian tes potensi akademik, basis data, dan teori berkaitan dengan perangkat yang digunakan seperti *mySQL*, *PHP*, dan *CodeIgniter*.

II.1 Prosedur Mengikuti Tes Potensi Akademik

Prosedur mengikuti Tes Potensi Akademik adalah sebuah aturan yang harus dilakukan sebelum melakukan Tes Potensi Akademik (TPA) yaitu melakukan pendaftaran *online* adapun tata caranya antara lain:

1. Calon peserta melakukan pendaftaran secara *online* (melalui Internet) dengan mengunjungi alamat yang telah ditentukan dan memilih menu Pendaftaran.
2. Calon peserta mengisi borang (formulir) pendaftaran *online* sesuai dengan petunjuk yang ada secara benar. Semua informasi yang diisikan dalam borang ini harus benar. Kesalahan/kecurangan dalam pengisian borang ini berakibat pembatalan penerimaan di SMA yang dituju.

II.2 Tes Potensi Akademik

Tes Potensi Akademik adalah sebuah tes yang bertujuan untuk mengukur kemampuan seseorang dibidang akademik umum. Tes ini juga sering diidentikkan dengan tes kecerdasan seseorang (Taufiqurrohman, 1998). Tes ini

diperuntukkan mengukur kemungkinan keberhasilan seseorang apabila yang bersangkutan melanjutkan ke dunia akademik yang lebih tinggi.

Adapun, Tes Potensi Akademik ini umumnya memiliki empat jenis soal. Yaitu, tes *verbal* atau bahasa, tes *numerik* atau angka, tes logika, dan tes *spasial* atau gambar.

Tes verbal berfungsi untuk mengukur kemampuan seseorang di bidang kata dan bahasa. Tes ini meliputi tes *sinonim* (persamaan kata), tes *antonim* (lawan kata), tes padanan hubungan kata, dan tes pengelompokan kata.

Tes angka berfungsi mengukur kemampuan seseorang di bidang angka, dalam rangka berpikir terstruktur dan logis matematis. Tes ini meliputi tes aritmetik (hitungan), tes seri angka, tes seri huruf, tes logika angka dan tes angka dalam cerita.

Tes logika berfungsi mengukur kemampuan seseorang dalam penalaran dan pemecahan persoalan secara logis atau masuk akal. Tes logika ini meliputi tes logika umum, tes analisa pernyataan dan kesimpulan (silogisme), tes logika cerita dan tes logika diagram.

Sedangkan tes spasial atau tes gambar, berfungsi mengukur daya logika ruang yang dimiliki seseorang. Tes ini meliputi antara lain tes padanan hubungan gambar, tes seri gambar, tes pengelompokan gambar, tes bayangan gambar dan tes identifikasi gambar.

Adapun standar yang diharapkan dapat dihasilkan melalui TPA atau akselerasi (Depdiknas, 2003) adalah siswa yang memiliki kemampuan-kemampuan unggul, yaitu:

- a. Kualifikasi kognitif: daya tangkap cepat, mudah dan cepat memecahkan masalah, dan kritis.

- b. Kualifikasi kreatif: rasa ingin tahu, imajinatif, tertantang, berani ambil resiko.
- c. Kualifikasi kecerdasan emosi: pemahaman diri sendiri, pemahaman terhadap orang lain, pengendalian diri, penyesuaian diri, harkat diri, dan berbudi pekerti luhur.

II.3 Profil Dinas Pendidikan Pemuda dan Olahraga Denpasar

Dinas Pendidikan Pemuda dan Olahraga Kota Denpasar pada waktu pemerintahan Orde Baru, Dinas ini bernama Dinas Pendidikan dan Kebudayaan Kota Madya Dati II Denpasar, seiring dengan jatuhnya pemerintahan Orde Baru hingga dengan munculnya pemerintahan Orde Reformasi maka seluruh tata kerja pemerintahan berubah termasuk Dinas ini berubah nama menjadi Dinas Pendidikan Dasar karena hanya mengurus bidang pendidikan dasar saja yaitu SD dan yang lainnya menjadi Kanin (Kantor Infeksi) Pendidikan dan Kebudayaan. Setelah berlangsung sekian dekade, maka Dinas Pendidikan Dasar dan Kanin Pendidikan Kebudayaan digabung menjadi Dinas Pendidikan dan Kebudayaan Kota Denpasar yang mengurus masalah pendidikan mulai dari TK, SD, SMP dan SMA/SMK.

Menindaklanjuti peraturan Pemerintah No. 41 Tahun 2007 tentang Organisasi Perangkat Daerah maka dibentuklah Organisasi dan tata Kerja daerah termasuk Dinas ini dengan nama Dinas Pendidikan dan Kebudayaan Kota Denpasar, sehingga Sub Dinas Kebudayaan yang dulunya bergabung dengan Dinas ini menjadi Dinas tersendiri yaitu Dinas Kebudayaan Kota Denpasar.

Berdasarkan peraturan daerah Kota Denpasar Kenpasar Nomor 7 tahun 2008 secara definitif berdirilah Dinas Pendidikan Pemuda dan Olah Raga Kota Denpasar.

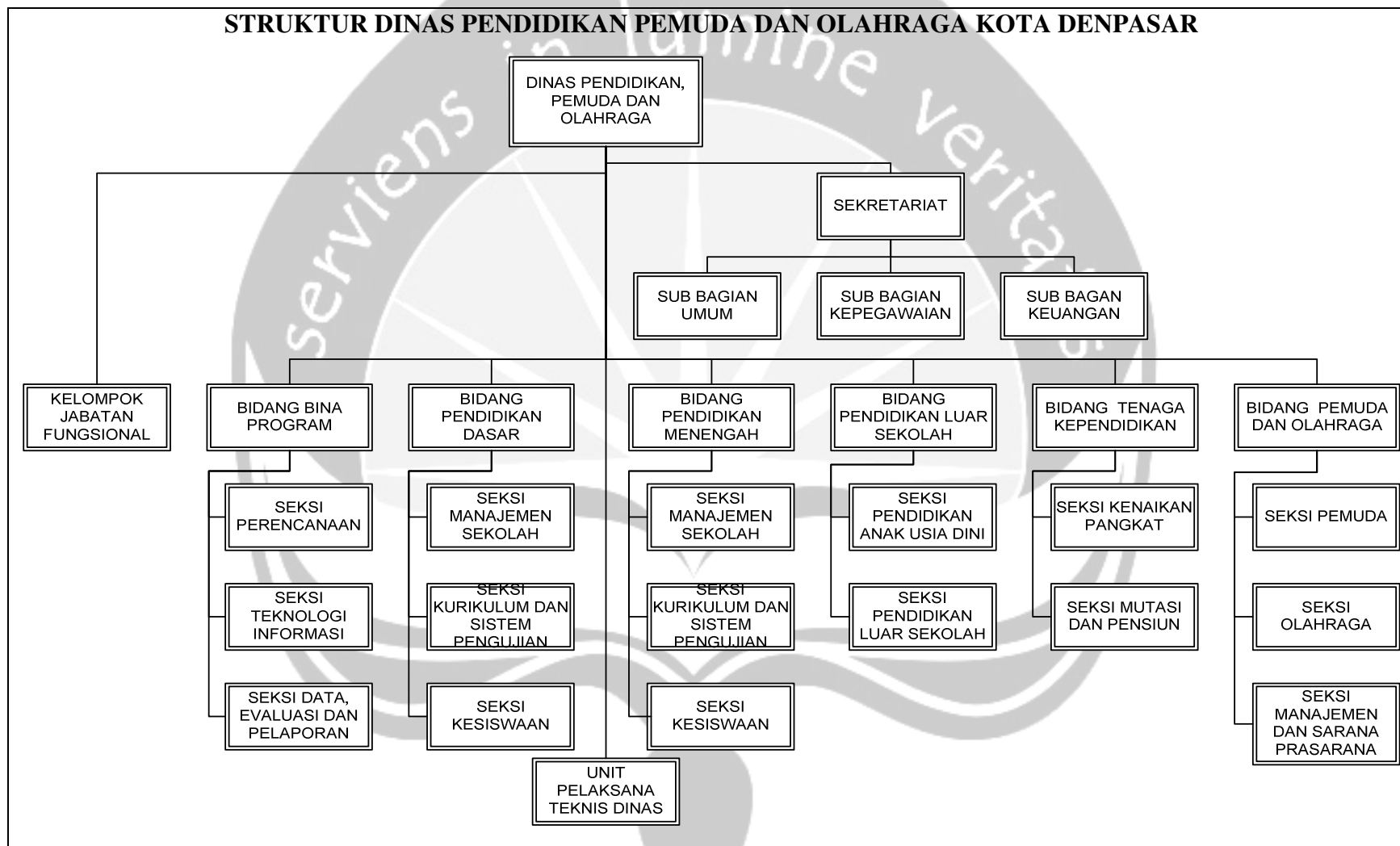
Khusus untuk pendirinya tidak tertera, namun Kepala Dinas mulai dari:

1. Drs. I Made Dasne,
2. Drs. I Gusti Ngurah Yadnya, BA
3. Drs. I gusti Lanang Jelantik M.Si.

II.3.1 Struktur Organisasi Dinas Pendidikan Pemuda dan Olahraga Denpasar

Struktur Organisasi dibentuk untuk menggambarkan hubungan antara bagian didalam organisasi. Dengan demikian setiap bagian mengetahui tugas, wewenang dan tanggung jawabnya dalam mencapai tujuan organisasi.

Pada Dinas Pendidikan Pemuda dan Olahraga Denpasar, struktur organisasinya berbentuk struktur garis, yang bercirikan rantai vertical antara berbagai tingkat organisasi. Hubungan antara bagian tercermin dalam bagan Struktur Dinas pendidikan Pemuda dan Olahraga Kota Denpasar.



Gambar 2.1 Struktur Dinas Pendidikan Pemuda dan Olahraga Denpasar

II.3.2 Visi dan Misi

1. Visi:

Terwujudnya peningkatan mutu pendidikan yang berakar pada budaya bagsa.

2. Misi:

1. Menciptakan Pemerataan Peningkatan akses Pendidikan.
2. Menciptakan Peningkatan Mutu Pendidikan.
3. Menciptakan efisiensi dan efektivitas pendidikan
4. Menciptakan pendidikan yang berkesesuaian (relevansi pendidikan).
5. Menciptakan peningkatan kapasitas manajemen pendidikan.
6. Menciptakan Peningkatan Pelestarian dan Pengembangan Pemuda, Olahraga.

II.3.3 Fungsi dan Tugas

Dinas pendidikan, Pemuda dan Olahraga Kota Denpasar merupakan salah satu lembaga pemerintahan yang berfungsi sebagai:

1. Meningkatkan kualitas pendidikan melalui jalur sekolah (SD/MI, SMP/MTs, SMA/MA dan SMK).
2. Meningkatkan kualitas pendidikan melalui pendidikan melalui jalur Formal dan Non Formal.
3. Meningkatkan pembinaan pemuda, olahraga dan kebudayaan.
4. Meningkatkan pelayanan dibidang pendidikan.

Melestarikan & pengembangan sekolah sebagai pusat budaya.

II.3.4 Tempat Pelaksanaan Penelitian

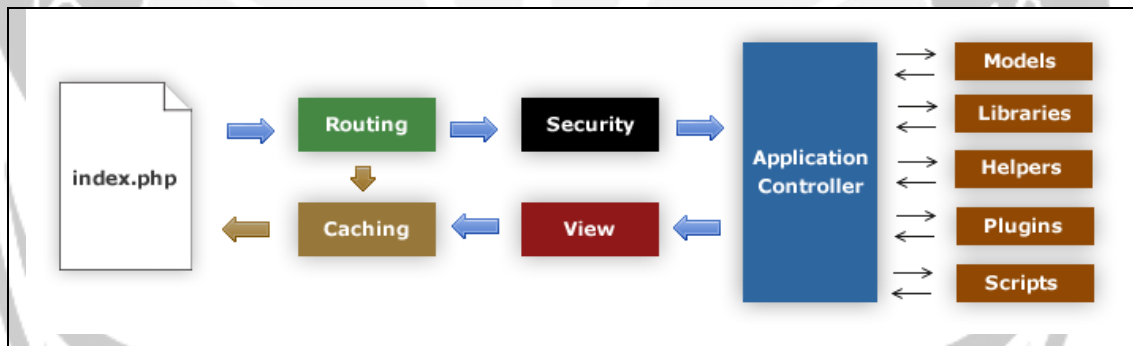
Penelitian ini dilaksanakan di kantor Bidang Bina Program Dinas Pendidikan Pemuda dan Olahraga Kota Denpasar lantai dua jln. Mawar no.6 denpasar. Kantor Bidang ini di kepalai oleh Dra. Nyoman Helsy Pujiastuti,MM. yang memiliki bawahan delapan orang yang memiliki tugas berbeda. Adapun seksi pada kantor bidang ini meliputi:

1. Seksi Perencanaan
2. Seksi teknologi informasi
3. Seksi data evaluasi dan pelaporan

II.4 Framework CodeIgniter

CodeIgniter merupakan salah satu *open source framework* yang digunakan oleh *script* pemrograman web PHP (*PHP Hypertext Preprocessor*) dalam mengembangkan aplikasi web dinamis. *CodeIgniter* dapat mempercepat *developer* untuk membuat sebuah aplikasi web. Ada banyak *library* dan *helper* yang berguna didalamnya dan tentunya mempermudah proses *development*. *Codeigniter* pada awalnya ditulis oleh mas Rick Ellis, pendiri dan CEO EllisLab.com, perusahaan yang mengembangkan *codeigniter* (Upton, 2007). Saat ini, *codeigniter* dikembangkan oleh komunitas dan disebarkan ke seluruh dunia dengan lisensi bebas. Metode yang digunakan oleh *framework CodeIgniter* disebut *Model - View - Controller* atau yang disingkat dengan sebutan MVC.

MVC memisahkan antara logika pemrograman dengan presentasi. Hal ini dapat terlihat dari adanya minimalisir *script* presentasi (*HTML*, *CSS*, *JavaScript*, dsb) yang dipisahkan dari *PHP script*. Didalam folder *CodeIgniter*, MVC dapat kita temukan dalam folder *application*. Selain *framework CodeIgniter*, masih banyak *framework* lain yang juga menggunakan *PHP*, misalnya *CakePHP*, *Symphony Project*, *Zend*, *Seagull*, *Prado*, dan lain sebagainya. *CodeIgniter* dirilis pertama kali pada 28 Februari 2006. Versi yang sudah mulai stabil dirilis pada 26 Juni 2008, yaitu versi 1.6.3. Sekarang versi terakhir yang telah dikeluarkan adalah 1.7.2.



Gambar 2.2 Konsep Framework CodeIgniter

II.4.1 Fitur-fitur Codeigniter

Berikut fitur-fitur yang didukung oleh *Codeigniter*:

1. Sistem berbasis *Model View Controller*
2. Berinteraksi dengan database apapun dengan satu bahasa tunggal
3. Memanajemen *session* dan *cookies*
4. Melakukan *validasi user input*

5. Membangun html seperti tabel, *form*, *link*, dan lainnya dengan kode minimal
6. Berkomunikasi dengan *xmlrpc*, *ftp*, *captcha*, *rss* dan teknologi lainnya
7. Sistem berbasis *Model View Controller*
8. Kompatibel dengan PHP versi 4.
9. Ringan dan Cepat.
10. Terdapat dukungan untuk berbagai basis data.
11. Mendukung *Active Record Database*.
12. Mendukung form dan validasi data masukan.
13. Keamanan dan *XSS filtering*.
14. Tersedia pengaturan *session*.
15. Tersedia *class* untuk mengirim email.
16. Tersedia *class* untuk manipulasi gambar (*cropping*, *resizing*, *rotate* dan lain-lain).
17. Tersedia *class* untuk upload file.
18. Tersedia *class* yang mendukung transfer via FTP.
19. Mendukung lokalisasi bahasa.
20. Tersedia *class* untuk melakukan *pagination* (membuat tampilan perhalaman).
21. Mendukung *enkripsi* data.
22. Mendukung *benchmarking*.
23. Mendukung *caching*.
24. Pencatatan *error* yang terjadi.
25. Tersedia *class* untuk membuat kalender.
26. Tersedia *class* untuk mengetahui *user agent*, misalnya tipe browser dan sistem operasi yang digunakan pengunjung.
27. Tersedia *class* untuk pembuatan *template website*.

28. Tersedia *class* untuk membuat *trackback*.
29. Tersedia pustaka untuk bekerja dengan XMP-RPC.
30. Menghasilkan clean URL.
31. URI *routing* yang *felksibel*.
32. Mendukung *hooks*, *ekstensi class* dan *plugin*.
33. Memiliki *helper* yang sangat banyak jumlahnya.

II.4.2 Keuntungan-keuntungan yang diperoleh programmer web dari penggunaan CodeIgniter antara lain :

1. Kompatibel dengan akses *hosting* yang standar dan dapat berjalan dengan variasi versi PHP apapun dengan konfigurasi yang bagaimanapun
2. *Open source framework* yang ringan dan *powerfull*
3. Alur pemrograman lebih terstruktur
4. Kinerja program yang baik
5. Proses *setup* yang cepat
6. Logika proses akan lebih dimengerti oleh *programmer web*
7. Tidak meminta konfigurasi apapun.
8. Fokus untuk menghasilkan solusi yang mudah
9. Tidak meminta untuk akses menggunakan CLI (seperti: *command line*, *console*, atau *terminal*).
10. Fleksibel terhadap peraturan *coding*
11. Dokumentasi yang lengkap
12. Tidak meminta *dependency* apapun, seperti PEAR (Upton, 2007).

II.5 Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, yang tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data yang terdapat di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas sehingga mudah untuk digunakan atau ditampilkan kembali serta dapat digunakan oleh satu atau lebih program aplikasi secara optimal.

Dalam melakukan desain dari *database* terdapat dua kata penting, yaitu data dan informasi (Rob Peter, Carlos Coronel, 2006). Data adalah fakta mentah atau fakta yang belum diproses untuk menghasilkan arti. Data-data tersebut dapat diolah dengan menggunakan grafik, sehingga dari data tersebut dapat diperoleh sesuatu yang berguna, yaitu informasi. Terdapat beberapa titik kunci yang menyatakan hubungan antara data dan informasi:

1. Data merupakan bangunan blok-blok informasi.
2. Informasi dihasilkan dengan memproses data.
3. Informasi digunakan untuk menyatakan arti data.
4. Informasi yang baik, relevan dan sesuai waktu adalah kunci untuk pembuatan keputusan yang baik.
5. Pembuatan keputusan yang baik adalah kunci terhadap pertahanan organisasi dalam lingkungan global.

Sehingga dari titik kunci tersebut dapat disimpulkan informasi yang sesuai waktu dan berguna membutuhkan data

yang baik. Basis data relasional selalu menggunakan field kunci dalam membangun relasi antar tabel dan merupakan model basis data yang populer karena kemudahan dalam penerapan dan kemampuannya dalam mengakomodasi berbagai kebutuhan pengelolaan basis data. Semakin banyak tabel yang ada dalam suatu basis data, semakin banyak pula relasi yang diperlukan untuk menghubungkan semua tabel.

Perkembangan teknologi basis data sendiri tidak terlepas dari perkembangan perangkat keras dan perangkat lunak komputer yang diikuti dengan perkembangan jaringan komputer dan komunikasi data yang kemudian melahirkan sistem basis data terdistribusi.

II.6 MySQL

MySQL adalah sebuah sistem manajemen database relasi (*relational database management system*) yang bersifat *open source* (Arbie, 2004). MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah *lisensi GPL* (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja *optimizer*-nya dalam

melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query* MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase. Ada beberapa pertimbangan mengapa memilih MySQL yaitu (Arbie, 2004):

Kecepatan. Berdasarkan hasil pengujian, MySQL memiliki kecepatan paling baik dibanding database server lainnya. Hasil pengujiannya ini dapat dilihat di <http://www.mysql.com/information/benchmarks.html>.

Kemudian, dengan adanya fitur tambahan *Query Caching* pada MySQL rilis 4.0 kinerja query secara umum akan naik rata-rata 200% dari kinerja biasanya.

Mudah digunakan. Perintah-perintah dan aturan-aturan pada MySQL maupun proses instalasinya relatif mudah digunakan.

Open Source. Dengan konsep ini siapapun dapat berpartisipasi untuk mengembangkan MySQL dan hasil pengembangan itu diserahkan kepada umum atau kepada komunitas *Open Source*.

Kapabilitas. MySQL telah digunakan untuk mengelola database dengan jumlah 50 juta *record*. Nahkan sanggup untuk mengelola 60.000 tabel dengan jumlah baris 5.000.000.000. Mendukung penggunaan index hingga 32 buah index per-tabelnya, sedangkan setiap indexnya terdiri dari 1 hingga 16 kolom criteria. Informasi ini diperoleh

dari manual MySQL yang dapat di-download dari situs MySQL.

Replikasi Data. Dengan adanya fasilitas replikasi data ini, kita dapat mempunyai beberapa database bayangan pada beberapa server 'anak' lainnya yang berasal dari satu database induk sehingga akan meningkatkan kinerja dan kecepatan MySQL.

Konektifitas dan keamanan. MySQL mendukung dan menerapkan sistem keamanan dan izin akses tingkat lanjut (*advanced permissions and security system*), termasuk dukungan pengamanan dengan cara pengacakan lapisan data (SSL transport layer encryption). Bahkan, pada MySQL rilis 4.0 dapat juga diterapkan pembatasan penggunaan sumber daya server berdasarkan penggunaan per user. Adanya tingkatan user dan jenis akses yang beragam dapat memberikan batasan akses berdasarkan lokasi (mesin) pengakses, dan terdapatnya sistem pengacakan password (encrypted password) akan memberikan jaminan keamanan pada MySQL.

Fleksibilitas/Portabilitas. MySQL mendukung perintah-perintah ANSI SQL 99 dan beberapa perintah database alternative lainnya sehingga memudahkan untuk beralih dari database MySQL.

Lintas platform sistem operasi. MySQL dapat dijalankan pada beberapa sistem operasi yang berbeda, seperti Linux, Microsoft Windows, FreeBSD, Sun Solaris, IBM's AIX, Mac OS X, HP-UX, AIX, QNX, Novell NetWare, SCO Open Unix, SGI Irix, dan Dec OSF.

Selain itu MySQL juga memiliki beberapa keistimewaan, antara lain :

1. *Multiuser*

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

2. *Performance tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

3. *Column types*

MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.

4. *Command dan functions*

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam *query*.

5. *Security*

MySQL memiliki beberapa lapisan sekuritas seperti *level subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta *password* terenkripsi.

6. *Scalability dan limits*

MySQL mampu menangani database dalam skala besar, dengan jumlah records lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

7. *Connectivity*

MySQL dapat melakukan koneksi dengan *client* menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).

8. *Localisation*

MySQL dapat mendeteksi pesan kesalahan pada *client* dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk didalamnya.

9. *Interface*

MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

10. *Clients dan tools*

MySQL dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi *database*, dan pada setiap *tool* yang ada disertakan petunjuk *online*.

11. *Struktur table*

MySQL memiliki struktur tabel yang lebih *fleksibel* dalam menangani *ALTER TABLE*, dibandingkan *database* lainnya semacam *PostgreSQL* ataupun *Oracle*.

II.7 Konsep Pembuatan Web

Untuk membuat suatu situs web yang menarik dan interaktif yang tidak hanya sekedar menampilkan informasi, tetapi dapat berinteraksi dengan user sehingga suatu situs tidak bersifat statis tetapi dapat bersifat dinamis, membutuhkan kemampuan pemrogram web.

Membuat aplikasi berbasis web berarti:

1. Memperkaya fungsi web server dengan cara menambahkan program pada dokumen web yang akan dieksekusi oleh server ketika file dokumen web tersebut diakses oleh web server.

Misalnya, program yang mengambil data ke basis data untuk ditampilkan ke web browser

2. Memperkaya inter aktivitas dokumen dengan cara menambahkan program pada dokumen web yang akan dieksekusi oleh web browser ketika file dokumen tersebut ditampilkan oleh web browser.

Misalnya, program yang memvalidasi data masukan pada form sebelum disubmit ke web server.

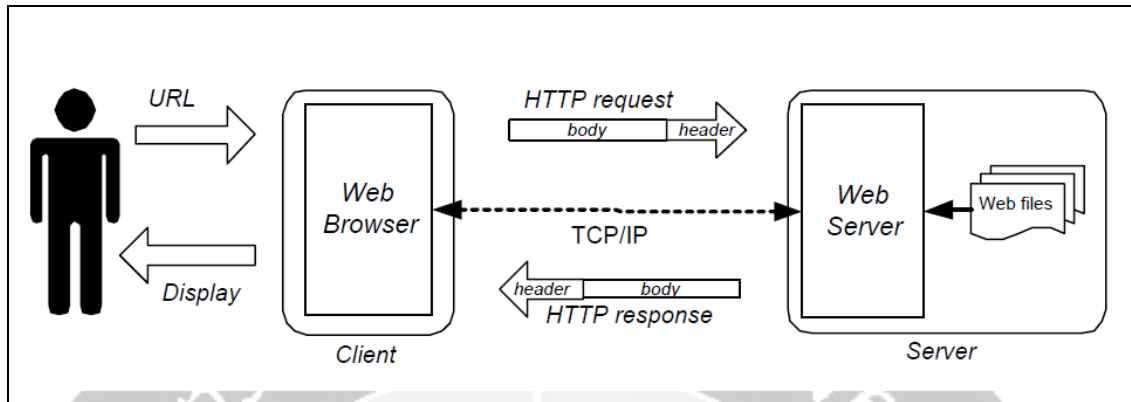
Kelebihan dan Kekurangan Aplikasi Berbasis Web

Kelebihan:

- a. Dapat diakses kapanpun dan dari manapun selama ada internet
- b. Dapat diakses hanya dengan menggunakan web browser (umumnya sudah tersedia di PC, PDA, dan handphone terbaru), tidak perlu menginstall aplikasi client khusus.

Kekurangan:

- a. Antarmuka yang dapat dibuat terbatas sesuai spesifikasi standar untuk membuat dokumen web dan keterbatasan kemampuan web browser untuk menampilkannya.
- b. Terbatasnya kecepatan internet mungkin membuat respon aplikasi menjadi lambat.



Gambar 2.3 Konsep Arsitektur Web

II.8 Web browser dan server web

Penjelajah web (bahasa inggris: *web browser*), disebut juga sebagai perambah atau peramban, adalah perangkat lunak yang berfungsi menampilkan dan melakukan interaksi dengan dokumen-dokumen yang disediakan oleh web server. Penjelajah yang populer adalah Microsoft internet Explorer dan Mozilla Firefox. (<http://id.wikipedia.org/wiki/Browser>).

Server web adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP atau HTTPS dari *client* yang dikenal dengan *browser web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman web yang umumnya berbentuk dokumen HTML. Server web yang terkenal diantaranya adalah *apache* dan *Microsoft Internet Information Service (IIS)*. *Apache* merupakan server web antar-platrom, sedangkan IIS hanya dapat beroperasi pada system operasi windows (http://id.wikipedia.org/wiki/server_web).

II.9 HTTP

HTTP adalah *protocol* yang digunakan untuk layanan *www*(world wide web) di jaringan TCP/IP. *Protocol* HTTP bersifat *request-response*, yaitu dalam *protocol* ini client menyampaikan pesan request ke server dan server kemudian memberikan *response* yang sesuai dengan *request* tersebut (RJB Wahju Agung et.al.,2002).

Hubungan HTTP yang paling sederhana terdiri atas hubungan langsung antara *user agent* dengan server asal. Hubungan HTTP tidak selalu seperti ini karena spesifikasi HTTP mengenal adanya beberapa komponen yang dapat terlibat dalam membentuk sebuah hubungan HTTP, yaitu *client* , *user agent*, server asal , *proxy*, *gateway*, dan *tunnel*.

Pada *protocol* HTTP terdapat tiga jenis hubungan dengan perantara, yaitu *proxy*, *gateway*, dan *tunnel*. *Proxy* bertindak sebagai agen penerus, menerima *request* dalam bentuk *Uniform Resource Identifier (URI) absolute*, mengubah *format request*, dan mengirimkan *request* ke server yang ditujukan oleh URI. *Gateway* bertindak sebagai agen penerima dan menerjemahkan *request* ke *protocol* server yang dilayani. *Tunnel* bertindak sebagai titik relay antara dua hubungan HTTP tanpa mengubah *request* dan *response* HTTP.

II.10 Server side dan Client side

Teknologi yang digunakan dalam pemrograman web dibagi menjadi dua yaitu *server side* dan *client side*. *Server* adalah komputer yang berjalan terus menerus tanpa

berhenti dan sebagai tempat dimana file/program dari aplikasi web diletakkan. Sedangkan *client* adalah *user* yang hendak mengakses program ke server dengan menggunakan alamat yang unik (tidak boleh kembar). Secara umum, untuk dapat mengakses aplikasi web di butuhkan *web browser*.

Pada *server side*, perintah-perintah program dijalankan di server dan dikirimkan ke browser sudah dalam bentuk HTML. Sedangkan *client side*, proses akan dilakukan di *web browser*. Biasanya *client side* digunakan untuk hal-hal yang membutuhkan interaksi user tetapi data yang ditampilkan tetap seragam.

Server side dan *client side* seakan-akan merupakan hal yang kontradiksi atau berlawanan. Namun *server side* dan *client side* tidak bersifat kotradiksi melainkan saling melengkapi. Penggunaan *server side* dan *client side* ini harus disesuaikan dengan keperluan dari aplikasi web itu sendiri.

Aplikasi web berjalan pada *protocol* HTTP, dan semua protokol di internet selalu melibatkan antara server dan client. Ketika seseorang mengetikkan suatu alamat di browser. Maka browser akan mengirimkan perintah tersebut ke web server. Jika yang diminta oleh *client* adalah *file* yang mengandung perintah *server side* maka *server web* akan menjalankan dahulu program tersebut lalu mengirimkannya kembali ke browser dalam bentuk HTML sehingga dapat diterjemahkan oleh browser. Sedangkan jika yang diminta oleh *client* adalah file yang mengandung file *client side*

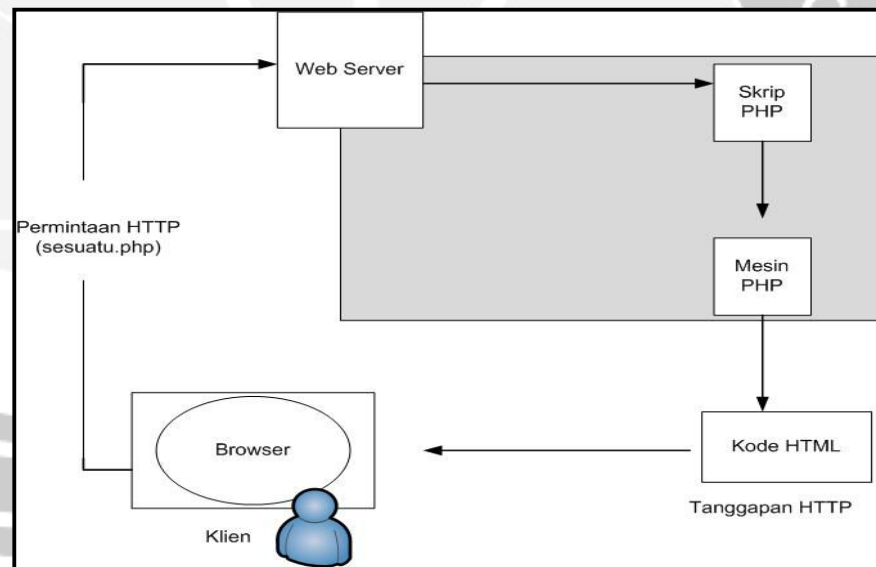
maka oleh server file tersebut akan langsung dikirimkan ke browser.

II.11 PHP (*PHP Hypertext Preprocessor*)

PHP merupakan singkatan dari "*PHP: Hypertext Preprocessor*". PHP merupakan salah satu bahasa script yang terbilang baru dan tersedia secara bebas dan masih memungkinkan untuk dikembangkan lebih lanjut. PHP dapat diintegrasikan (*embedded*) ke dalam *web server*, atau dapat berperan sebagai program CGI yang terpisah. PHP adalah bahasa pemrograman yang memungkinkan para *web developer* untuk membuat aplikasi web yang dinamis dengan cepat. Karakteristik yang paling unggul dan paling kuat dalam PHP adalah lapisan integrasi database (*database integration layer*). Database yang didukung PHP adalah: *Oracle, Adabas-D, Sybase, FilePro, mSQL, Velocis, MySQL, Informix, Solid, dBase, ODBC, Unix dbm*, dan *PostgreSQL*.

PHP dikatakan sebagai sebuah *server-side embedded script language* artinya sintaks-sintaks dan perintah yang kita berikan akan sepenuhnya dijalankan oleh server tetapi disertakan pada halaman HTML biasa. Aplikasi-aplikasi yang dibangun oleh PHP pada umumnya akan memberikan hasil pada web browser, tetapi prosesnya secara keseluruhan dijalankan di server. Konsep kerja PHP diawali dengan permintaan suatu halaman web oleh browser. Berdasarkan URL (*Uniform Resource Locator*) atau dikenal dengan sebutan alamat internet, browser mendapatkan alamat dari web server, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang

dibutuhkan oleh web server. Selanjutnya, web browser akan mencari berkas yang diminta dan memberikan isinya ke browser. Browser yang mendapatkan isinya segera melakukan proses penerjemahan ke kode HTML dan menampilkannya ke layar pemakai. Jika yang diminta adalah halaman PHP, maka ketika berkas PHP yang diminta didapatkan oleh web server, isinya segera dikirimkan ke mesin PHP dan mesin inilah yang memproses dan memberikan hasilnya (berupa kode HTML) ke web server. Selanjutnya web server menyampaikan ke klien.



Gambar 2.4 Konsep Kerja PHP (Kadir, 2003)

II.11.1 Kelebihan PHP

Kelebihan PHP dari bahasa pemrograman lain :

1. *Server-side script*

PHP dijalankan pada web server, kemudian barulah hasilnya ditampilkan ke *client (browser)* dalam bentuk bahasa HTML, sehingga source code dari script PHP tersebut tidak akan bisa dilihat dari browser.

2. *Cross-platform*

Script-script PHP dapat dijalankan pada berbagai sistem operasi, sehingga PHP ini dapat dijalankan pada komputer yang menggunakan Windows maupun yang menggunakan Linux.

3. *HTML-embedded*

Source code PHP dapat dituliskan dengan penggabungan instruksi-instruksi PHP dan tag-tag HTML.

4. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.

5. Web Server yang mendukung PHP dapat ditemukan dimana-mana dari mulai IIS sampai dengan *apache*, dengan *configurasi* yang relatif mudah.

6. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu dalam pengembangan.

7. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena referensi yang banyak.

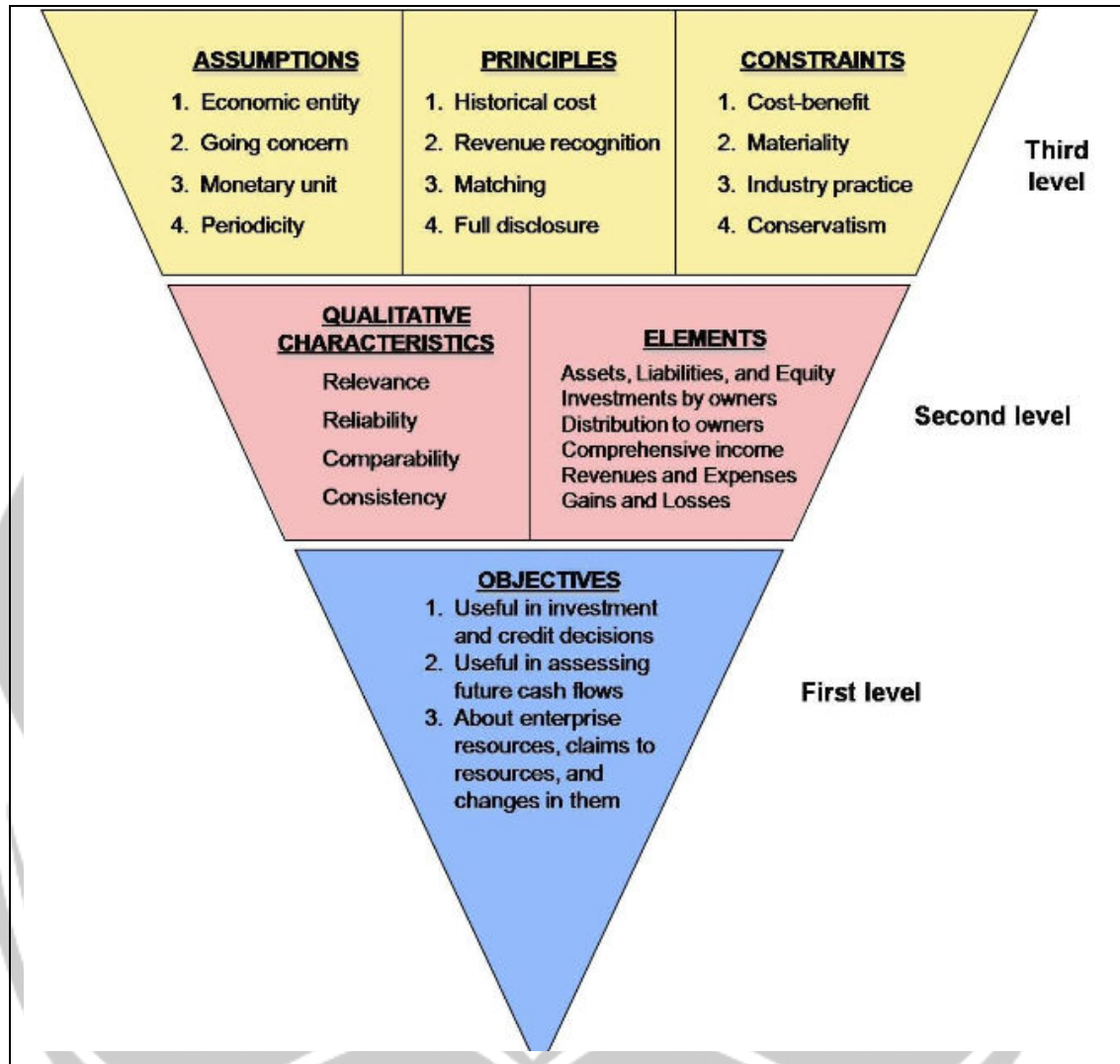
8. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (*linux, unix, windows*) dan dapat

dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

9. PHP mampu berkomunikasi dengan berbagai database terkenal. Beberapa diantaranya adalah (Kadir, 2003):
DBM, FirePro, Informix, Ingres, InterBase, Microsoft Access, MySQL, MSOL, Oracle, Postgre SQL, Sybase, dBase.

II.12 Framework

Framework adalah sekumpulan fungsi, *class*, dan aturan-aturan. Berbeda dengan *library* yang sifatnya untuk tujuan tertentu saja, *framework* bersifat menyeluruh mengatur bagaimana kita membangun aplikasi. Framework memungkinkan membangun aplikasi dengan lebih cepat. Karena framework telah menyediakan hal-hal penunjang lainnya seperti koneksi database, *form validation*, GUI, dan *security*.



Gambar 2.5 Konsep Framework

II.13 Framework PHP

Framework PHP adalah sebuah bahasa pemrograman yang memungkinkan seorang developer (*programer* atau *system analyst*) membuat aplikasi berbasis web yang *powerful* sekaligus mampu mengampu *database* berskala besar. Dalam perkembangannya, seorang programer PHP seringkali diuntut untuk menciptakan sebuah dasar aplikasi yang bisa

di kembangkan menjadi aplikasi lain dengan skala yang lebih besar dengan melibatkan banyak anggota tim, Framework sendiri terdiri dari file-file pustaka berupa kelas-kelas PHP yang bisa di gunakan dan di kolaborasikan dalam aplikasi yang sedang anda kembangkan.

II.14 MVC (Model, View, Controller)

Design MVC (*Model, View, Controller*) akan memisahkan *business logic* dari *presentation*, tetapi meletakkan "sesuatu" (*controller*) diantara *business logic* dan *presentation* sehingga mereka masih bisa berhubungan dan *business logic* dapat berdiri sendiri dan tidak perlu tahu apapun tentang *presentation*. Model adalah kombinasi dari *business data* dan *metod*.

MVC terdiri dari:

1. Model

Model bertanggung jawab atas *business logic* dan data. Model mengetahui bagaimana cara untuk mengolah data. Model adalah bagian dari *system* yang berhubungan dengan *database*.

2. View

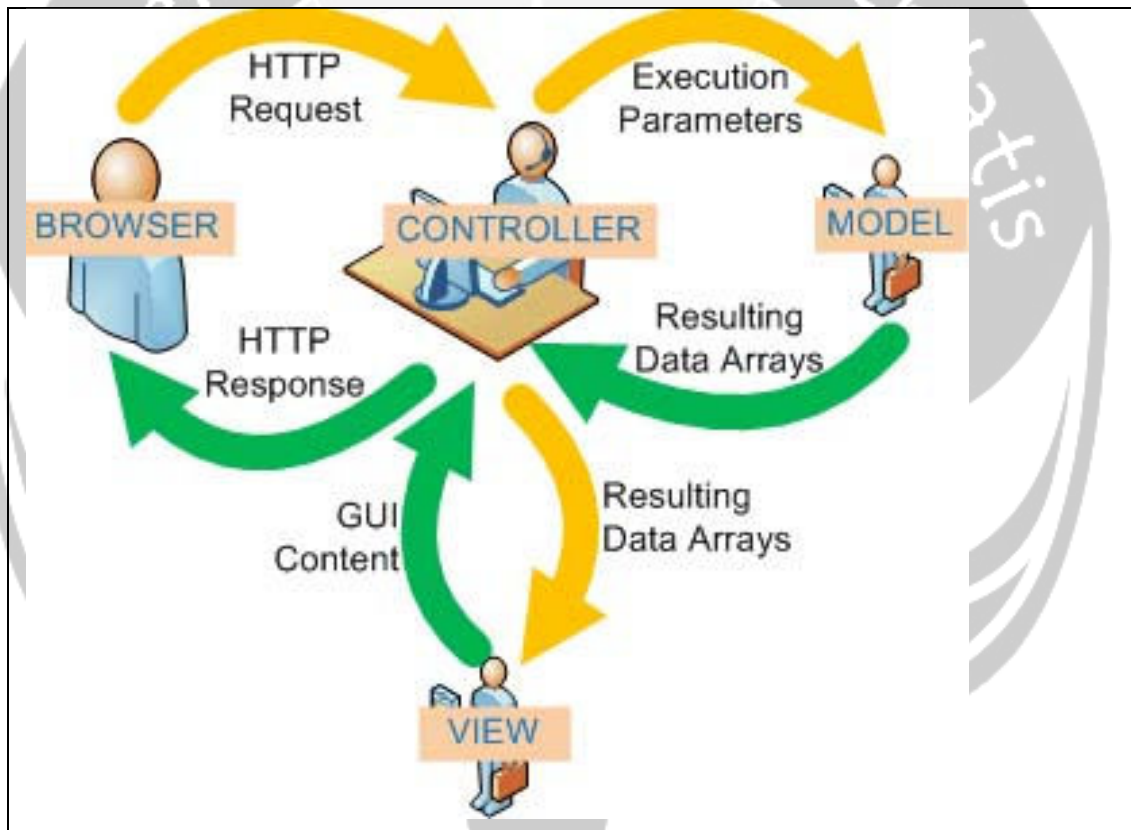
View bertanggung jawab atas *presentation* (tampilan yang dapat dilihat oleh user). *View* mendapatkan data dari Model melalui *Controller*.

3. Controller

Controller menerima input dari user dan melemparkannya ke Model. *Controller* akan meminta Model untuk mengelola data tersebut dan memberikan

hasil data yang sesuai untuk diberikan pada user melalui view.

Dengan model MVC akan memudahkan jika suatu saat terdapat perubahan baik di business logic maupun presentation karena masing-masing sudah ditangani oleh komponen yang berbeda dan tidak akan mempengaruhi semua komponen (Basham et al., 2004).



Gambar 2.6 Konsep MVC (Model, View, Controller)

Pada bab ini telah dijelaskan konsep-konsep dan teori-teori secara singkat yang berkaitan dengan inti perancangan perangkat. Adapun yang telah dijelaskan adalah pengertian tes potensi akademik, basis data, dan

teori berkaitan dengan perangkat yang digunakan seperti *mySQL*, PHP, dan *CodeIgniter*. Selanjutnya pada bab III akan menjelaskan analisis dan perancangan sistem dengan bahasa pemrograman PHP dengan framework *CodeIgniter* yang memiliki Login, register, Change Password, Kelola setting, Kelola Soal, Pilih Jawaban, edit Profile, Empty Field, serta Pengumuman.

