

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Pengendalian Persediaan (*Inventory Control*)**

##### **2.1.1 Pengertian pengendalian (*Control*)**

Pengendalian persediaan merupakan menjaga biaya keseluruhan yang terkait persediaan dengan memiliki persediaan sedikit mungkin tanpa menimbulkan masalah. *Inventory control* disebut juga *stock control*. Pengendalian persediaan merupakan bagian yang penting dalam bisnis. Mengendalikan persediaan dengan baik adalah menjaga keseimbangan setiap waktu dengan memiliki persediaan yang optimal untuk memaksimalkan keuntungan. Jumlah persediaan yang besar akan menimbulkan biaya yaitu biaya penyimpanan persediaan sehingga keuntungan perusahaan berkurang. Selain itu, barang juga dapat rusak jika tidak disimpan dengan baik.

##### **2.1.2 Pengertian Persediaan (*Inventory*)**

Persediaan merupakan salah satu unsur yang paling aktif dalam operasi perusahaan kontinue diperoleh, dirubah, yang kemudian dijual kembali. Sebagian besar dari sumber-sumber perusahaan juga sering dikaitkan didalam persediaan yang akan digunakan dalam perusahaan pabrik. Nilai pada persediaan harus dicatat dan digolong-golongkan menurut jenisnya yang kemudian dibuatkan perincian dari masing-masing barangnya dalam suatu periode yang bersangkutan (Assauri, 1999).

Setiap perusahaan apakah itu perusahaan perdagangan atau pabrik serta perusahaan jasa selalu mengadakan persediaan, karena itu persediaan sangat

penting, tanpa adanya persediaan para pengusaha yang mempunyai perusahaan—perusahaan tersebut akan dihadapkan pada resiko-resiko yang dihadapi, misalnya; pada sewaktu-waktu perusahaan tidak dapat memenuhi keinginan pelanggan yang memerlukan atau meminta barang atau jasa yang dihasilkan. Hal tersebut dapat terjadi karena disetiap perusahaan tidak selamanya barang-barang atau jasa-jasa tersedia setiap saat, yang berarti pengusaha akan kehilangan kesempatan memperoleh keuntungan yang seharusnya didapatkan.

Begitu pentingnya persediaan sehingga merupakan elemen utama terbesar dari modal kerja yang merupakan aktiva yang selalu dalam keadaan berputar dimana secara terus-menerus mengalami perubahan. Persediaan merupakan suatu aktiva yang meliputi barang-barang milik perusahaan yang dimaksud untuk dijual dalam satu periode usaha yang normal atau persediaan barang baku yang menunggu penggunaannya dalam suatu proses produksi (Rangkuti, 2004).

### **2.1.3 Jenis-jenis persediaan**

Menurut Siagian (2005) ada 5 jenis-jenis persediaan antara lain, yaitu :

1. Persediaan bahan mentah (*Raw Material*), persediaan barang-barang berwujud seperti besi, kayu serta komponen-komponen lainnya yang digunakan dalam proses produksi barang-barang dari jenis ini dapat diperoleh dari sumber-sumber alam atau dibeli dari perusahaan yang menghasilkan bahan baku.
2. Persediaan komponen-komponen rakitan (*Purchased Parts*), persediaan barang-barang yang terdiri dari komponen-komponen yang diperoleh dari perusahaan lain yang secara langsung dapat dirakit menjadi suatu produk.

3. Persediaan bahan pembantu atau penolong (*Supplies*), persediaan barang-barang (bahan pembantu) yang diperlukan dalam proses produksi tetapi tidak merupakan bagian atau komponen barang jadi.
4. Persediaan barang dalam proses (*Working Process*), persediaan barang-barang yang merupakan keluaran dari tiap-tiap bagian dalam proses produksi atau yang telah diolah menjadi suatu bentuk, tetapi masih perlu diproses lebih lanjut menjadi barang jadi.
5. Persediaan barang jadi (*Finished Goods*), persediaan barang-barang yang telah selesai dipolah dalam pabrik dan siap dijual atau dikirim kepada pelanggan.

#### 2.1.4 Fungsi dan Tujuan Persediaan

Setiap perusahaan dagang atau manufaktur sepakat bahwa persediaan memiliki fungsi yang sangat membantu dalam setiap kegiatan usaha. Seperti yang dijelaskan sebelumnya bahwa persediaan merupakan suatu hal vital dalam suatu perusahaan. Fungsi persediaan menurut Rangkuti (2004) terdiri dari:

1. Fungsi *Decoupling* adalah persediaan yang memungkinkan perusahaan dapat memenuhi permintaan pelanggan tanpa tergantung pada *supplier*. Persediaan bahan mentah diadakan agar perusahaan tidak akan sepenuhnya tergantung pada pengadaannya dalam hal kuantitas dan waktu pengiriman. Persediaan barang dalam proses diadakan agar departemen dan proses-proses individual perusahaan terjaga kebebasannya. Persediaan barang jadi diperlukan untuk memenuhi permintaan produk yang tidak pasti dari para pelanggan.
2. Fungsi *Economic Lot Sizing*. Persediaan *lot size* ini perlu mempertimbangkan penghematan-penghematan atau potongan pembelian, biaya pengangkutan

per unit menjadi lebih murah dan sebagainya. Hal ini disebabkan karena perusahaan melakukan pembelian dalam kuantitas yang lebih besar, dibandingkan dengan biaya-biaya yang timbul karena besarnya persediaan (biaya sewa gudang, investasi, resiko, dan sebagainya).

3. Fungsi *Antisipasi* permintaan yaitu yang dapat apabila perusahaan diperkirakan menghadapi dan diramalkan fluktuasi berdasarkan pengalaman atau data-data masa lalu, yaitu permintaan musiman. Dalam hal ini perusahaan dapat mengadakan persediaan musiman (*seasonal inventories*).

### 2.1.5 Metode Penilaian Persediaan

Persediaan barang dagangan bisa dihitung dengan menggunakan beberapa metode penilaian persediaan diantaranya adalah Metode FIFO (*First In First Out*), Metode LIFO (*Last In First Out*), Metode Rata-Rata (*Average*).

Metode FIFO adalah metode yang mendahulukan bahan yang dibeli pertama kali untuk dikeluarkan pertama juga (Muljono dan Wicaksono, 2009). Metode FIFO mengasumsikan bahwa unit-unit yang pertama memasuki proses produksi adalah unit-unit yang pertama kali diselesaikan dan ditransfer keluar.

Metode FIFO banyak digunakan oleh perusahaan-perusahaan, karena:

1. Perhitungan dan pelaksanaan sederhana
2. Nilai persediaan akhir pada neraca sesuai dengan harga yang berlaku sekarang
3. Dapat menghindari kerusakan dan keusangan persediaan terkait produk yang dihasilkan berupa produk makanan dan minuman

## 1.2 Penjadwalan Produksi

Penjadwalan (*scheduling*) merupakan tindakan penentuan periode waktu untuk masing-masing pekerjaan dalam proses produksi. Penjadwalan produksi (*production schedule*) merupakan rencana penentuan waktu dalam volume pekerjaan-pekerjaan produksi. Penjadwalan adalah suatu hal yang bermanfaat karena akan menentukan jumlah produksi yang diharapkan yang seharusnya dapat dicapai disetiap stasiun kerja selama satu hari atau satu minggu (Madura, 2007).

Penjadwalan merupakan pengaturan waktu dari suatu kegiatan operasi, yang mencakup kegiatan pengalokasikan fasilitas, peralatan maupun tenaga kerja dan penentuan aturan pelaksanaan bagi suatu kegiatan operasi (Herjanto, 2007). Dalam suatu perusahaan industri, penjadwalan diperlukan antara lain dalam pengalokasian tenaga operator, mesin dan peralatan produksi, urutan proses, jenis produk, dan pembelian material.

Penjadwalan bertujuan meminimalkan waktu proses, waktu tunggu langganan, dan tingkat persediaan, serta penggunaan yang efisien dari fasilitas, tenaga kerja dan peralatan. Penjadwalan yang baik akan memberikan dampak positif yaitu rendahnya biaya operasi dan waktu pengiriman, yang pada akhirnya dapat meningkatkan kepuasan pelanggan.

Penjadwalan diperlukan ketika beberapa pekerjaan harus diproses pada suatu mesin tertentu yang tidak bisa memproses lebih dari satu pekerjaan pada saat yang sama. Penjadwalan yang baik akan memaksimalkan efektivitas pemanfaatan setiap sumber daya yang ada, sehingga penjadwalan merupakan kegiatan yang penting dalam perencanaan dan pengendalian produksi. Tahap perencanaan dan tahap implementasi dari kegiatan penjadwalan merupakan masalah yang kompleks.

### 1.3 Master Production Schedule (MPS)

Jadwal Induk Produksi (*Master Production Schedule*, MPS) adalah gambaran atas periode perencanaan dari suatu permintaan, termasuk peramalan, *backlog*, rencana suplai/penawaran, persediaan akhir, serta kualitas yang dijanjikan tersedia (*available to promise*). MPS disusun berdasarkan perencanaan produksi agraget, dan merupakankunci penghubung dalam rantai perencanaan dan pengendalian produksi. MPS berkaitan dengan pemesanan, rencana distribusi, perencanaan produksi dan perencanaan kapasitas. MPS harus dibuat secara realistis, dengan mempertimbangkan kemampuan, kapasitas produksi, tenaga kerja, dan subkontraktor.

Ketepatan MPS bervariasi berdasarkan jangka waktu perencanaannya. Perencanaan jangka pendek harus lebih akurat, mengingat biasanya berisi pesanan yang sudah pasti (*fixed order*), kebutuhan distribusi pergudangan, dan kebutuhan suku cadang. Semakin jauh jangka waktu perencanaan ketepatan MPS biasanya semakin berkurang (Herjanto, 2007).

### 1.4 Sistem Informasi

Menurut McLeod (2006) sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Informasi adalah data yang telah diproses, atau data yang telah memiliki arti. Data itu sendiri berarti fakta-fakta dan angka-angka yang relatif tidak berarti bagi pemakai. Sebagai contoh, data dapat berupa jumlah jam kerja tiap pegawai dalam perusahaan. Jika jam kerja tiap pegawai dikalikan dengan upah per jam maka hasilnya adalah pendapatan kotor. Jika angka pendapatan kotor tiap pekerja dijumlahkan, penjumlahan tersebut adalah total biaya gaji bagi seluruh

perusahaan. Jumlah biaya gaji dapat menjadi informasi bagi pemilik perusahaan. Maka, sistem informasi itu berarti kumpulan dari elemen-elemen yang saling berhubungan untuk satu tujuan dan menghasilkan informasi bagi perusahaan.

Maka dapat disimpulkan sistem informasi merupakan kumpulan elemen-elemen atau komponen-komponen yang saling berhubungan, yang mempunyai satu tujuan dan digunakan untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi yang berguna bagi perusahaan.

### **1.5 System Development Life Cycle (SDLC)**

Pengembangan sistem *System Development Life Cycle* (SDLC) diperlukan untuk menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang ada hal ini dikarenakan adanya permasalahan pada sistem lama, pertumbuhan organisasi, meraih kesempatan, adanya instruksi.

Dengan telah dikembangkannya sistem yang baru, maka diharapkan akan terjadi peningkatan-peningkatan di sistem yang baru. Peningkatan-peningkatan ini berhubungan yaitu sebagai berikut:

1. *Performance* (kinerja), peningkatan terhadap kinerja (hasil kerja) sistem yang baru sehingga menjadi lebih efektif. Kinerja dapat diukur dari *throughput* dan *response time*. *Throughput* adalah jumlah dari pekerjaan yang dapat dilakukan suatu saat tertentu. *Response time* adalah rata-rata waktu yang tertunda diantara dua transaksi atau pekerjaan ditambah dengan waktu response untuk menanggapi pekerjaan tersebut.

2. *Information* (informasi), peningkatan terhadap kualitas informasi yang disajikan. *Economy* (ekonomis), peningkatan terhadap manfaat-manfaat atau keuntungan-keuntungan atau penurunan-penurunan biaya yang terjadi.
3. *Control* (pengendalian), peningkatan terhadap pengendalian untuk mendeteksi dan memperbaiki kesalahan-kesalahan serta kecurangan-kecurangan yang dan akan terjadi.
4. *Efficiency* (efisiensi), peningkatan terhadap efisiensi operasi. Efisiensi berbeda dengan ekonomis. Bila ekonomis berhubungan dengan jumlah sumber daya yang digunakan, efisiensi berhubungan dengan bagaimana sumber daya tersebut.

SDLC adalah pendekatan bertahap untuk melakukan analisis dan membangun rancangan sistem dengan menggunakan siklus yang spesifik terhadap kegiatan pengguna dalam membangun sistem informasi (Kendall & Kendall, 2006). Langkah yang digunakan meliputi: melakukan survei dan menilai kelayakan proyek pengembangan sistem informasi, mempelajari dan menganalisis sistem informasi yang sedang berjalan, menentukan permintaan pemakai sistem informasi, memilih solusi atau pemecahan masalah yang paling baik, menentukan perangkat keras (*hardware*) dan perangkat lunak (*software*), merancang sistem informasi baru, membangun sistem informasi baru, mengkomunikasikan dan mengimplementasikan sistem informasi baru, memelihara dan melakukan perbaikan/peningkatan sistem informasi baru bila diperlukan.



Dalam sebuah siklus SDLC, terdapat lima langkah. Jumlah langkah SDLC pada referensi lain mungkin berbeda, namun secara umum adalah sama. Langkah tersebut adalah sebagai berikut :

1. Identifikasi masalah

Seluruh kebutuhan *software* harus bisa didapatkan dalam fase ini, termasuk didalamnya kegunaan *software* yang diharapkan pengguna dan batasan *software*. Informasi ini biasanya dapat diperoleh melalui wawancara, *survey* atau diskusi. Informasi tersebut dianalisis untuk mendapatkan dokumentasi kebutuhan pengguna untuk digunakan pada tahap selanjutnya.

2. Analisis sistem

Analisis sistem dapat didefinisikan sebagai pengguna dari suatu sistem informasi yang utuh kedalam bagian-bagian komponen dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan, sehingga dapat diusulkan perbaikan-perbaikan. Tahap ini merupakan tahap yang kritis dan penting karena kesalahan pada tahap ini akan menyebabkan kesalahan pada tahap berikutnya. Langkah-langkah dasar yang harus dilaksanakan oleh analisis sistem, yaitu:

- a. Memahami sistem dan membatasinya.
- b. Alternatif-alternatif apa saja yang ada untuk mencapai sasaran dan untuk memodifikasi atau mengubah sistem, yaitu: pilih satu dari alternatif yang telah diidentifikasi pada tahap sebelumnya, implementasikan alternatif yang dipilih, dan evaluasi masalah dari perubahan yang kita buat dalam sistem.

### 3. Rancangan sistem

Alternatif yang telah dipilih dalam langkah analisis sistem merupakan dasar dari rancangan sistem. Rancangan sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang harus diselesaikan. Tahap ini menyangkut konfigurasi dari komponen-komponen perangkat keras dan perangkat lunak sistem sehingga setelah menginstalasi sistem akan benar-benar akan memuaskan spesifikasi sistem yang telah ditetapkan pada akhir analisis sistem.

### 4. Implementasi sistem

Tahap dari implementasi sistem adalah membangun dan menguji jaringan *database*, membangun dan menguji program, instalasi dan menguji sistem yang baru, penyerahan sistem yang telah dibuat, dan perawatan dan pengembangan sistem.

Diperlukan adanya kegiatan tambahan setelah sistem yang baru dijalankan, seperti merawat menjaga agar sistem tetap berjalan sesuai dengan apa yang dikehendaki. Perlu juga diperhatikan akibat adanya kebijaksanaan yang baru yaitu perubahan-perubahan prosedur, agar sistem tetap menjalankan fungsinya sehingga pengembangan sistem diperlukan.

## 1.6 MySQL

MySQL adalah *Relational Database Management Sistem* (RDBMS) yang didistribusikan secara gratis di bawah lisensi *General Public License* (GPL). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial (Prasetyo, 2006).

Berikut ini beberapa keistimewaan yang dimiliki oleh MySQL, yaitu:

1. *Portability* : dapat berjalan stabil pada berbagai sistem operasi (windows, Linux, dan Mac OS).
2. *Open Source*: didistribusikan secara gratis, di bawah lisensi GPL.
3. *Multiuser* : dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. *Performance Tuning*: memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana.
5. *Column Types*: memiliki tipe kolom yang sangat kompleks, seperti *signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year, set* dan *enum*.
6. *Command dan Functions*: memiliki operator dan fungsi secara penuh yang mendukung perintah *SELECT* dan *WHERE* dalam query.
7. *Security*: memiliki beberapa lapisan sekuritas seperti level *subnetmark*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta *password* terenkripsi.
8. *Scalability dan Limits*: mampu menangani *database* dalam skala besar, dengan jumlah records lebih dari 50 juta dan 60 ribu *table* serta 5 miliar baris. Batas *index database* MySQL yang dapat ditampung mencapai 32 *indeks* pada tiap tabelnya.
9. *Connectivity* : dapat melakukan koneksi dengan *client* menggunakan protokol TCP/IP, *Unix socket (Unix)*, atau *Named Pipes (NT)*.
10. *Localisation*: dapat mendeteksi pesan kesalahan (*error code*) pada *client* dengan menggunakan lebih dari dua puluh bahasa.

11. *Interface*: memiliki interface terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi *Application Programming Interface* (API).
12. *Lients* dan *Tools*: dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi *database*, dan pada setiap *tools* yang ada disertakan petunjuk online.
13. Struktur tabel: memiliki struktur tabel yang lebih lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan database lainnya semacam Postgre SQL ataupun Oracle.

### **1.7 Hypertext Preprocessor (PHP)**

PHP merupakan singkatan dari *Hypertext Preprocessor*. PHP merupakan bahasa pemrograman yang bersifat *server-side scripting*. PHP memungkinkan Anda untuk membuat halaman web yang bersifat dinamis. PHP merupakan *software open source* yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya <http://www.php.net>.

PHP dapat dijalankan pada berbagai macam sistem operasi misalnya Windows, LINUX, dan Mac OS. Selain Apache, PHP juga mendukung beberapa web server lain, misalnya Microsoft IIS, Caudium, PWS, dan lain-lain.

Hingga kini, PHP sudah berkembang hingga versi 5. PHP 5 mendukung penuh *Object-Oriented Programming* (OOP), integrasi XML, mendukung semua ekstensi terbaru MySQL, pengembangan web services dengan SOAP dan REST, serta ratusan peningkatan lainnya dibandingkan versi sebelumnya (Ramadhan, 2006).

Kelebihan dari PHP itu sendiri, yaitu :

- a. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, nginx, hingga Xitami dengan konfigurasi lebih mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa *script* yang paling mudah karena memiliki referensi yang banyak.
- e. PHP adalah bahasa *open source* yang dapat digunakan di beberapa mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

### 1.8 Cascading Style Sheets (CSS)

*Cascading Style Sheets* (CSS) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa markup. Penggunaan yang paling umum dari CSS adalah untuk memformat halaman web yang ditulis dengan HTML dan XHTML. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen XML termasuk SVG dan XUL. Spesifikasi CSS diatur oleh *World Wide Web Consortium* (W3C).

CSS diperkenalkan untuk pengembangan website pada tahun 1996. Nama CSS didapat dari fakta bahwa setiap deklarasi *style* yang berbeda dapat diletakkan secara beruntun, yang kemudian akan membentuk hubungan *parent-child* pada setiap *style*.

Setelah CSS distandarisasikan, Internet Explorer dan Netscape melepas browser terbaru mereka yang telah sesuai atau paling tidak hampir mendekati dengan standar CSS. CSS adalah sebuah dokumen yang berdiri sendiri dan dapat dimasukkan dalam kode HTML atau sekedar menjadi rujukan oleh HTML dalam pendefinisian style. CSS menggunakan kode-kode yang tersusun untuk menetapkan style pada elemen HTML atau dapat juga digunakan untuk membuat style baru yang biasa disebut class (Sulistiyawan, Rubianto, dan Saleh, 2008).

### **1.9 Java Script**

*Java script* diperkenalkan pertama kali oleh Netscape pada tahun 1995. Pada awalnya bahasa ini dinamakan “ *LiveScript* ” yang berfungsi sebagai bahasa sederhana untuk browser *Netscape Navigator 2*. *Javascript* adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengizinkan pengekseskuan perintah perintah disisi *user*, yang artinya disisi browser bukan disisi server web. *Javascript* bergantung kepada browser (*navigator*) yang memanggil halaman web yang berisi skrip-skrip dari *Javascript* dan tentu saja terselip didalam dokumen HTML (Hernita, Sri dan Amanda, 2010).

### **1.10 CodeIgniter**

Menurut Hakim (2010:8) *CodeIgniter* adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi web berbasis PHP dibanding jika menulis semua kode program dari awal. *CodeIgniter*

pertama kali dibuat oleh Rick Ellis, CEO Ellislab, nc. (<http://ellislab.com>), sebuah perusahaan yang memproduksi *Content Management System* (CMS) yang cukup handal, yaitu *Expression Engine*. Saat ini, *CodeIgniter* dikembangkan dan dimaintain oleh *Expression Engine Development Team*.

Adapun beberapa keuntungan menggunakan *CodeIgniter*, diantaranya:

1. Gratis

*CodeIgniter* berlisensi dibawah *Apache/BSD open source*.

2. Ditulis menggunakan PHP 4

Meskipun *CodeIgniter* dapat berjalan pada PHP 5, namun sampai saat ini kode program *CodeIgniter* masih dibuat dengan menggunakan PHP 4.

3. Berukuran Kecil

Ukuran *CodeIgniter* yang kecil merupakan keunggulan tersendiri dibanding dengan *framework* lain yang berukuran besar.

4. Menggunakan Konsep MVC

*CodeIgniter* menggunakan konsep MVC yang memungkinkan pemisahan *layer application - logic dan presentation*.

5. URL yang Sederhana

Secara default, URL yang dihasilkan *CodeIgniter* sangat bersih dan *Search Engine Friendly* (SEF).

6. Memiliki Paket *Library* yang Lengkap

*CodeIgniter* mempunyai *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh sebuah aplikasi berbasis web, misalnya mengakses *database*, mengirim *email*, memvalidasi *form*, dan menangani *session*.

## 7. Extensible

Sistem dapat dikembangkan dengan mudah menggunakan *plugin* dan *helper*, atau dengan menggunakan *hooks*.

## 8. Tidak Memerlukan *Template Engine*

Meskipun *CodeIgniter* dilengkapi dengan *template* parser sederhana yang dapat digunakan, tetapi hal ini tidak mengharuskan untuk menggunakannya.

## 9. Dokumentasi Lengkap dan Jelas

Dari sekian banyak *framework*, *CodeIgniter* adalah satu-satunya *framework* dengan dokumentasi yang lengkap dan jelas.

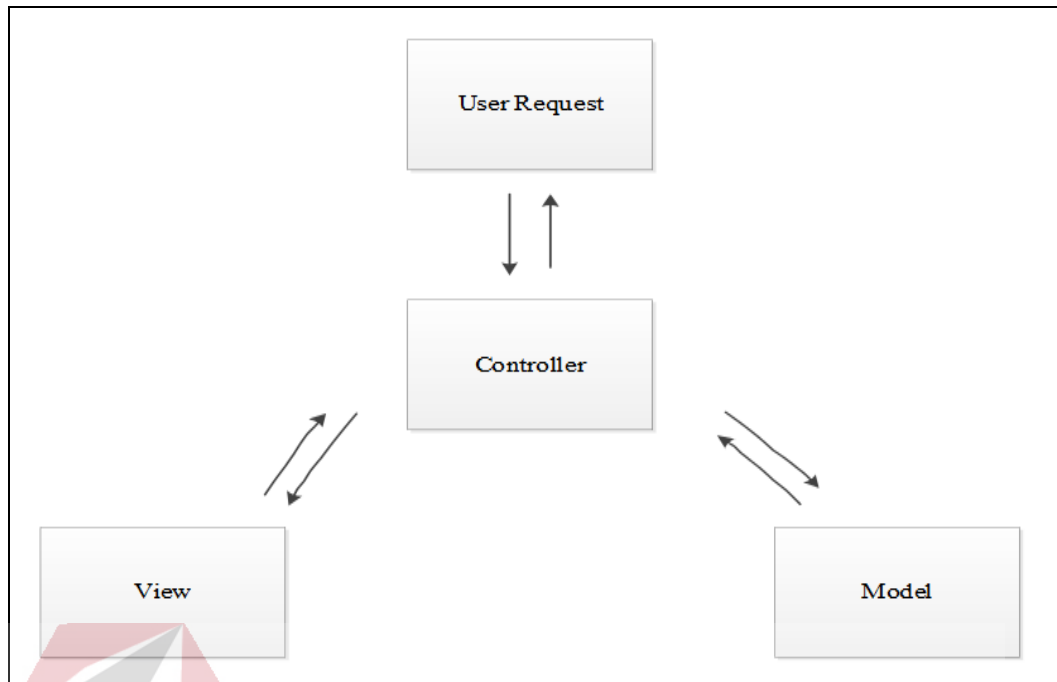
## 10. Komunitas

Komunitas *CodeIgniter* saat ini berkembang pesat. Salah satu komunitasnya bisa dilihat pada (<http://codeigniter.com/forum/>).

### 1.11 Model-View-Controller

*CodeIgniter* adalah *framework* PHP yang dibuat berdasarkan kaidah *model-view-controller* (MVC). Dengan MVC, maka memungkinkan pemisahan antara layer *application-logic* dan *presentation*. Sehingga, dalam sebuah pengembangan web, seorang programmer bisa berkonsentrasi pada *core-system*, sedangkan web designer bisa berkonsentrasi pada tampilan web. Menariknya, *skrip* PHP, *query* MySQL, *Javascript* dan CSS bisa saling terpisah, tidak dibuat dalam satu skrip berukuran besar yang membutuhkan *resource* besar pula untuk mengesekusinya (Hakim, 2010).





**Gambar 2.1** *Model-View-Controller*

Adapun alur program aplikasi berbasis *framework Codeigniter* dapat dilihat pada Gambar 2.1. Gambar di atas menerangkan bahwa ketika datang sebuah *user request*, maka akan ditangani oleh *controller*, kemudian *controller* akan memanggil model jika memang diperlukan operasi *database*. Hasil dari *query* oleh model kemudian akan dikembalikan ke-*controller*. Selanjutnya *controller* akan memanggil *view* yang tepat dan mengkombinasikannya dengan hasil *query* model. Hasil akhir dari operasi ini akan ditampilkan web dibrowser.

Dalam konteks *CodeIgniter* dan aplikasi berbasis web, maka penerapan konsep MVC mengakibatkan kode program dapat dibagi menjadi tiga kategori, yaitu :

1. *Model*

Kode program (berupa OOP *class*) yang digunakan untuk memanipulasi *database*.

2. *View*

Berupa template html/xml atau php untuk menampilkan data pada *browser*.

3. *Controller*

Kode program (berupa OOP *class*) yang digunakan untuk mengontrol aliran aplikasi (sebagai pengontrol *model* dan *View*).

