

PURCHASE REQUISITION BERBASIS WEB MENGGUNAKAN METODE MODEL VIEW CONTROLLER DENGAN FRAMEWORK CODEIGNITER

Studi Kasus : PT. Batamindo Investment Cakrawala

Vivin Andryan Daya¹, Eka Suswaini, ST, MT² dan Martaleli Bettiza, S.Si., M.Sc³
Teknik Perangkat Lunak, Fakultas Teknik, Universitas Maritim Raja Ali Haji
Jl. Politeknik Senggarang, Tanjungpinang
E-mail: vivinandryandaya@gmail.com

Abstrak— Proses permintaan pembelian barang di PT Batamindo masih dilakukan secara manual, dimana karyawan harus mengisi borang dan mengantarkan borang tsb ke bagian pembelian. Cara ini kurang efisien dalam penggunaan waktu dan biaya. Dalam penelitian ini dibuat suatu aplikasi permintaan barang (*Purchase Requisition*) berbasis web menggunakan bahasa pemrograman PHP dengan metode MVC dalam kerangka *CodeIgniter*, untuk memudahkan pengadaan atau pembelian barang. Metode MVC adalah metode pengembangan web yang mengembangkan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi yaitu manipulasi data, *user interface*, dan *control* aplikasi. *Codeigniter* adalah sebuah *framework* yang digunakan dalam membuat sebuah aplikasi berbasis web yang disusun dengan menggunakan bahasa PHP. Penggunaan metode MVC pada pembuatan aplikasi *Purchase Requisition* ini menjadikan program lebih terstruktur sehingga lebih mudah untuk dikembangkan. Aplikasi *Purchase Requisition* yang dibuat menjadikan proses permintaan pembelian barang di PT Batamindo menjadi lebih efisien dari segi waktu dan biaya.

Kata kunci : *purchase Requisition*, , *model*, *view*, *controller*, *framework codeigniter*,

I. PENDAHULUAN

A. Latar Belakang

Manajemen pekerjaan, sangatlah diperlukan di zaman globalisasi yg penuh dengan kemajuan teknologi pada saat ini, berbagai fasilitas telah dikembangkan oleh ilmu pengetahuan yg mempermudah segala pekerjaan dan aktifitas yg dilakukan, sering kali kebanyakan orang mengabaikan kemudahan ini karena sudah terbiasa menggunakan proses manual. Bercermin pada permasalahan ini maka diperlukan inisiatif untuk mempermudah segala pekerjaan dengan menggunakan teknologi yang ada di zaman globalisasi sekarang . Dalam melaksanakan Kerja Praktek (KP) di PT. Batamindo investement cakrawala, proses permintaan pembelian barang (*PURCHASE REQUISITION*) di Perusahaan tersebut masih menggunakan kertas atau masih menggunakan proses manual yang memiliki beberapa kerugian. Karena masih menggunakan proses manual yaitu menggunakan borang maka sering kehilangan data permintaan, susah nya mencari data pemesanan yang telah ada, diperlukan tempat penyimpanan kertas yang lumayan besar serta efisiensi waktu yang masih kurang.

Sementara di zaman yang modern seperti saat ini kita melihat hampir semua sistem informasi telah bisa diselesaikan secara komputerisasi sehingga memudahkan dalam pencarian data serta meminimalisir kerugian waktu dan biaya.

B. Tujuan

Berdasarkan latar belakang dan rumusan masalah yang telah dipaparkan diatas maka dapat disusun tujuan penelitian yaitu:

1. Membangun sistem perangkat lunak *multiuser* berbasis web yang dapat mempermudah proses pembelian barang dalam sebuah manajemen pekerjaan dengan penerapan metode *model*, *view* dan *controller* (MVC).
2. Serta membantu perusahaan dalam efisiensi anggaran biaya yang sebelumnya masih menggunakan borang atau kertas yang membutuhkan biaya yg besar.

C. Batasan Masalah

Batasan masalah yang diambil dalam penulisan tugas akhir ini meliputi:

1. Penelitian akan dilakukan di PT Batamindo Investment Cakrawala
2. Aplikasi yang dibuat berfungsi untuk menangani *Purchase Requisition* (pembelian barang)
3. Hanya menangani *Expenditure Control*

II. LANDASAN TEORI

A. PHP (*hypertext Preprocessor*)

PHP merupakan bahasa pemrograman yang digunakan untuk membuat program website dimana kode program yang telah dibuat dikompilasi dan dijalankan pada sisi server untuk menghasilkan halaman website yang dinamis. PHP dibuat pada tahun 1994 oleh Rasmus Lerdorf (Adhi Prasetyo, 2011). Penulisan bahasa pemrograman PHP ini menyatu dengan HTML (*HyperText Markup Language*).

B. MVC

Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman *Small Talk*, MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, *user interface*, dan bagian yang menjadi kontrol aplikasi. Dengan

menggunakan konsep *Model, View dan Controller (MVC)* ini akan menyebabkan pada halaman *web* mengandung kode yang sedikit, dikarenakan pada konsep MVC struktur tampilan dan logika telah di pisahkan sehingga memungkinkan pengerjaan web lebih terstruktur dan mudah dikembangkan serta pengerjaan dapat dilakukan secara fokus. MVC juga meningkatkan fleksibilitas aplikasi untuk pengembangan aplikasi kedepan (Gupta, 2010).

C. Purchase

Istilah *purchasing* atau pembelian sinonim dengan *procurement* atau pengadaan barang. Berikut adalah definisi *procurement* menurut Bodnar dan Hopwood (2001:323), yaitu: “*Procurement is the business process of selecting a source, ordering, and acquiring goods or services.*” Pendapat tersebut kurang lebih mempunyai arti: bahwa pengadaan barang adalah proses bisnis dalam memilih sumber daya-sumber daya, pemesanan dan perolehan barang atau jasa. Brown dkk. (2001:132) mengatakan bahwa secara umum pembelian bisa didefinisikan sebagai: “*managing the inputs into the organization’s transformation (production process).*” Pendapat tersebut kurang lebih mempunyai arti bahwa pembelian merupakan pengelolaan masukan ke dalam proses produksi organisasi. Berikut adalah pendapat Galloway dkk. (2000:31) mengenai fungsi pembelian, yaitu: “*The role of purchasing function is to make materials and parts of the right quality, and quantity available for use by operations at the right time and at the right place.*” Pendapat tersebut kurang lebih mempunyai arti bahwa peran fungsi pembelian adalah untuk mengadakan material dan part pada kualitas yang tepat dan kuantitas yang tersedia untuk digunakan dalam operasi pada waktu yang tepat dan tempat yang tepat.

D. Requisition

Permintaan / demand adalah kegiatan manusia/pembeli yang menunjukkan keinginan untuk membeli sejumlah barang tertentu sesuai dengan tingkat harga di waktu tertentu. Faktor - faktor yang mempengaruhi permintaan :

- Harga barang itu sendiri
- Adanya barang pengganti / barang substitusi
- Pendapatan rumah tangga dan pendapatan rata-rata masyarakat
- Corak distribusi pendapatan dalam masyarakat
- Cita rasa / selera masyarakat
- Jumlah penduduk
- Ramalan masa datang,

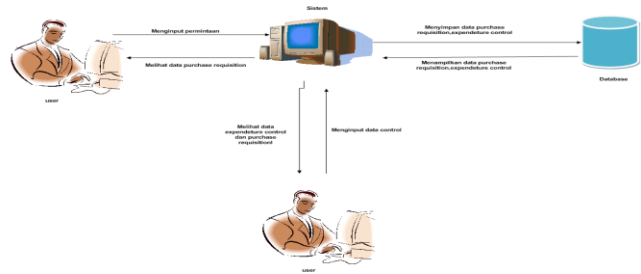
E. Codeigniter

CodeIgniter adalah sebuah *framework* yang digunakan untuk membuat sebuah aplikasi berbasis web yang disusun dengan menggunakan bahasa PHP. Di dalam CI terdapat beberapa macam kelas (*class*) yang berbentuk *library* dan helper. Keduanya berfungsi untuk membantu pemrogram (*programmer*) dalam mengembangkan aplikasinya (Riyanto, 2010).

III. PEMODELAN DAN PERANCANGAN APLIKASI

A. Pemodelan Sistem

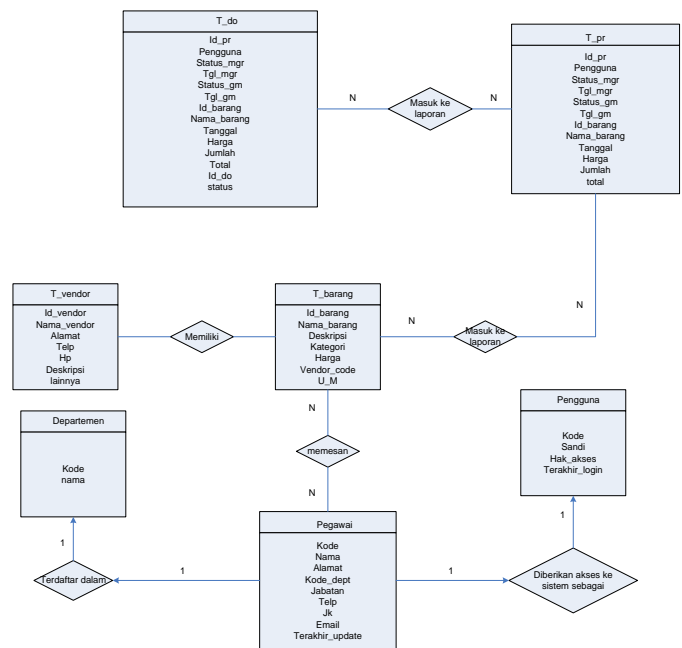
Pemodelan sistem aplikasi yang akan dibangun dalam penelitian ini dapat di lihat pada gambar di bawah ini:



Gambar 1 Permodelan Sistem

B. Entity Relationship Diagram

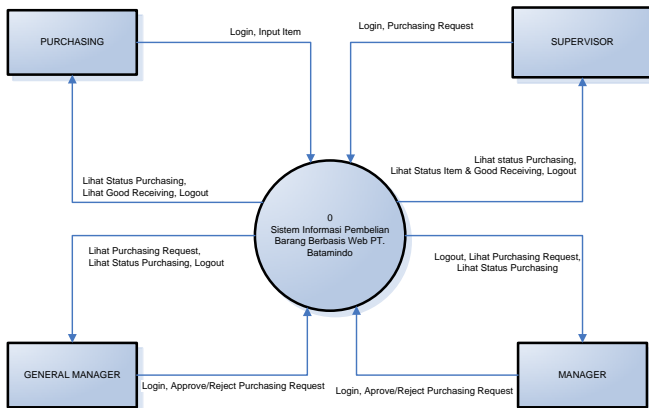
Berikut gambar *Entity Relationship Diagram*



Gambar 2 Entity Relationship Diagram

C. Data Flow Diagram (DFD)

Berikut adalah gambar data flow diagram dari aplikasi ini.



Gambar 3 Data Flow Diagram

IV. IMPLEMENTASI DAN PENGUJIAN

Berikut merupakan hasil pembahasan dari implementasi metode *model*, *View* dan *controller* (MVC) pada sistem aplikasi monitoring pekerjaan, MVC merupakan arsitektur yang menjadi pondasi dasar dalam merancang dan membangun aplikasi monitoring pekerjaan yang berbasis *web* ini.

Berdasarkan hasil perancangan yang telah dijelaskan pada bab sebelumnya bahwasanya sistem aplikasi *purchasing request* ini dibangun dengan mengimplementasikan arsitektur *model*, *View*, *controller* (MVC). Penerapan arsitektur MVC pada framework CodeIgniter akan membagi kode program menjadi 3 komponen yaitu *model*, *view*, dan *controller*, yang mana akan membagi penulisan kode program akan menjadi lebih terstruktur karena pemisahan kode program dan memiliki dokumentasi yang baik karena direktori penyimpanan yang telah dibedakan. Sebagai contoh proses pembuatan proses manajemen pekerjaan pada sistem ini yang diimplementasikan pada framework Codeigniter.

Berikut adalah cuplikan kode program *file model* yang merupakan interpretasi dari basis data yang telah dirancang. File model ini berfungsi sebagai penanganan validasi dari inputan yang dimasukkan pengguna dan sebagai file yang mengatur proses penyimpanan, perubahan data terhadap tabel database. Dibawah ini adalah cuplikan kode program file model *Requisition.php*

```
<?php
class purchasingmodel extends CI_Model {
    function _construct()
    {
        parent::_construct();
    }
}
function
save_item($kdv,$kd,$nm,$des,$ktg,$hrgr,$u_m
)
{
    $this->db->where('Id_barang',$kd);
    $this->db->limit(1);
    $q = $this->db->get('t_barang');
```

```
if($q->num_rows() == 0){
    $data = array('Id_barang' => $kd,
        'nama_barang' => $nm,
        'deskripsi' => $des,
        'kategori' => $ktg,
        'harga' => $hrgr,
        'vendor_code' => $kdv,
        'u_m' => $u_m
    );
    $this->db->insert('t_barang',$data);
    if($this->db->affected_rows()
    {
        return true;
    }else{
        return false;
    }
}
return false;
}
```

Function pada *file model* yang telah dijelaskan di atas merupakan sebuah *function* yang akan mengatur proses penyimpanan data proses *Purchasing Request* kedalam *table database* *t_pr*. Pada *framework codeigniter* *file model* memiliki beberapa *function* yang akan mengatur alur penyimpanan data dan penayangan data kedalam halaman tampilan antarmuka. *Function* yang telah dijelaskan di atas adalah *function* yang memiliki fungsi untuk proses penyimpanan data kedalam *database*. Setelah membuat *file model*, maka langkah selanjutnya adalah membuat *file controller* yang berfungsi sebagai pengatur dan penghubung antara *file model* dan *view*. Sama halnya pada setiap aplikasi yang menerapkan metode *model*, *view*, *controller*, pada *framework codeigniter* penerapan arsitektur ini pada *framework* ini juga tidak berbeda jauh. Di bawah ini adalah cuplikan pada program *file controller purch_order.php*.

```
class purc_order extends CI_Controller {
    function simpan()
    {
        $this->load->model('ordermodel');
        if($this->input->post('jm') == '' ||
        $this->input->post('jm') == '0')
        {
            $this->session->set_flashdata('sess_submitpr','Jumlah tidak boleh bernilai 0 atau kosong');
            redirect('/purc_order/edit/');
        }else{
            if($this->ordermodel->save_data($this->input->post('kd'),$this->input->post('nm'),
            $this->input->post('hrgr'),$this->input->post('jm'),$this->input->post('ttl'))
            {
```

```

        $this->session-
>set_flashdata('sess_submitpr','Data
Pekerjaan Berhasil Disimpan');
        redirect('/purc_order/cari/');
    }else{
        $this->session-
>set_flashdata('sess_submitpr','Data
Pekerjaan Gagal Disimpan');
        redirect('/purc_order/edit/');
    }
}

function cari(){
    $this->load->model('ordermodel');
    $data['datapr'] = $this->ordermodel-
>list_datapr();
    $this->load-
>view('src_purc_order',$data);
}

```

Function baru pada *file controller* yang menggunakan *framework codeigniter* di atas memiliki tujuan untuk mengatur dan mengarahkan pengguna ke *function cari* dan *function cari* akan mengarahkan pengguna ke halaman *src_purc_order.php*. Halaman *view purc_order.php* adalah halaman yang berfungsi sebagai halaman yang digunakan oleh pengguna level *supervisor* untuk melihat setatus permintaan barang yang akan disetujui oleh *level manager* dan *general manager*. Setelah disetujui oleh *manager* dan *general manager* kemudian *system* akan mengarahkan permintaan barang tersebut ke bagian *purchasing* dan *purchasing* akan membeli barang tersebut. Setelah pembuatan *file model* dan *controller* telah selesai dilakukan tahap selanjutnya adalah pembuatan *file view* sebagai komponen yang akan menjadi halaman antar muka yang akan mempermudah pengguna dalam menggunakan aplikasi ini. Di bawah ini adalah contoh cuplikan kode program halaman tampilan antarmuka *src_purc_order.php*.

```

<?php $this->load->view('navigasi');?>
<div id="content">
<br/><br/>
<h2>Purchasing Request</h2>
<?php
if($this->session-
>flashdata('sess_noticepekerjaan') != ""){
    echo("<div class=\"message
info\"><p>" . $this->session-
>flashdata('sess_noticepekerjaan')
."</p></div>");
}
?>
<table cellpadding="0" cellspacing="0"
border="0" class="display" id="table-
pekerjaan" style="width:100%">
<thead>
<tr>
<th>Kode PR</th>
<th>Status
Manager</th>
<th>Tanggal Action</th>
<th>Status General
Manager</th>

```

```

<th>Tanggal Action</th>
<th>Kode
Barang</th>
<th>Nama Barang</th>
<th>Harga</th>
<th>Jumlah</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<?php
$linkproses = "-";
foreach($datapr->result() as
$rows)
{
    echo("<tr>
<td>$rows->id_pr</td>
<td>$rows->status_mgr</td>
<td>$rows->tgl_mgr</td>
<td>$rows->status_gm</td>
<td>$rows->tgl_gm</td>
<td>$rows->id_barang</td>
<td>$rows-
>nama_barang</td>
<td>$rows->harga</td>
<td>$rows->jumlah</td>
<td>$rows->total</td>
</tr>");
}
?>
</tbody>
</table>
</div>
<?php $this->load-
>view('script_footer');?>

```

File view di atas akan menjadi *landing page* ketika pengguna ingin mengetahui status dari permintaan barang mereka. Pemanfaatan *framework codeigniter* untuk membangun aplikasi web adalah keuntungannya yang memanfaatkan arsitektur MVC sehingga membuat kode program menjadi lebih tertata. Keuntungan lain ketika terjadi kesalahan pada salah satu komponen maka pelacakan pesan kesalahan menjadi lebih mudah dan akan mempercepat proses pembaharuan kode program sehingga program menjadi siap untuk digunakan.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Implementasi metode *model, view, controller* pada pengembangan *system Purchase Requisition* dapat disimpulkan bahwa, aplikasi yang menerapkan MVC sebagai metode dasar pembuatan aplikasinya, mempermudah pengelolaan dan *maintenance* aplikasi. Aplikasi *purchase Requisition* berbasis *web* pada PT. Batamindo Investment Cakrawala mengimplementasikan metode MVC yang menggunakan *codeigniter* sebagai *framework* memudahkan karyawan dalam proses rencana pembelian barang (*Purchase Requisition*) dan membantu penyimpanan data barang kedalam *database* dan penerapan metode MVC ini dalam

aplikasi dapat meminimalisir waktu apabila terjadi kesalahan program, dan apabila satu komponen dikembangkan tidak akan mengganggu komponen lain. Karena pemisahan antara file *model*, *view*, dan *controller*.

B. Saran

Dalam pengembangan aplikasi *Purchase requisition* di sarankan agar aplikasi ini dapat dikembangkan ke *Mobile application*. Dan arsitektur MVC dapat dipertimbangkan untuk mengurangi biaya *maintenance* aplikasi besar.

UCAPAN TERIMA KASIH

Penulis Vivin Andryan Daya Mengucapkan terimakasih kepada pembimbing tugas akhir ini yang telah membimbing dengan baik dan menyempatkan waktunya dan kepada teman-teman yang membantu dalam pembuatan tugas akhir ini sehingga tugas ini dapat diselesaikan pada waktunya dan seluruh dosen dan staf universitas maritim raja ali haji atas dukungannya.

DAFTAR PUSTAKA

- [1] Glenn E. Krasner & Stephen T. Pope (1988), "A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System". *ParcPlace Systems, Inc. 1550 Plymouth Street Mountain View, CA 94043 glenn@ParcPlace.com Copyright © 1988 ParcPlace Systems. All Rights Reserved.*
- [2] E. Althammer and W. Pree (2001), "Design And Implementation Of A Mvc-Based Architecture For E-Commerce Applications". *C. Doppler Lab for Software Research University of Constance.*
- [3] Bahrawi (2010), "Perancangan Aplikasi Message Board Menggunakan Framework Berbasis MVC". *Jurnal Penelitian Komunikasi dan Opini Publik.*
- [4] AdhiPrasetio 2011, *Menjadi Master PHP*, Mediakita, Jakarta Selatan
- [5] Bodnar dan Hopwood (2001:323), yaitu: "Procurement is the business process of selecting a source, ordering, and acquiring goods or services."
- [6] Riyanto 2011, *Membuat Sendiri Aplikasi E-commerce dengan PHP dan MySQL Menggunakan Codeigniter dan JQuery*. Andi, Yogyakarta.