

# IMPLEMENTASI PEMROGRAMAN PHP BERBASIS MODEL VIEW CONTROLLER (MVC) PADA WEBSITE E-COMMERCE ( Studi Kasus: [www.indominiatur.com](http://www.indominiatur.com) )

Donni Prabowo S.Kom<sup>1)</sup>, Beti Wulan Sari S.Kom<sup>2)</sup>, Pendi Ventri Hendika S.Kom<sup>3)</sup>

<sup>1), 2), 3)</sup> Teknik Informatika STMIK AMIKOM Yogyakarta

Jl Ring road Utara, Condongcatur, Sleman, Yogyakarta 55281

Email : [donni.official@gmail.com](mailto:donni.official@gmail.com)<sup>1)</sup>, [sariwulanbety@gmail.com](mailto:sariwulanbety@gmail.com)<sup>2)</sup>, [ven3day@gmail.com](mailto:ven3day@gmail.com)<sup>3)</sup>

## Abstrak

*Model-View-Controller (MVC) merupakan suatu metode untuk membuat sebuah aplikasi atau sistem dengan memisahkan beberapa bagian penting dari aplikasi tersebut diantaranya memisahkan bagian data (Model) dari tampilan (View) dan cara bagaimana memprosesnya (Controller). Salah satu keuntungan dari MVC itu sendiri adalah kemudahan maintenance dan kemudahan pengembangan aplikasi. Penerapan konsep Model-View-Controller (MVC) akan membuat alur program menjadi lebih jelas sehingga mudah dipahami oleh pengembang selanjutnya.*

*Pada penelitian ini konsep Model-View-Controller (MVC) akan diimplementasikan dengan menggunakan Framework PHP yaitu Framework Codeigniter. Studi kasus yang digunakan yaitu Website yang menjual miniatur-miniatur seperti miniatur kereta , becak , kapal , pesawat , sepeda , motor , mobil dan lain sebagainya . Disini pemilik toko hanya berperan sebagai distributor saja , jadi pemilik tidak mempunyai toko miniatur secara fisik. Oleh karena itu seluruh media promosi akan dilakukan melalui media internet.*

*Selain menggunakan konsep Model-View-Controller (MVC), Website E-commerce ini juga memiliki dua fitur yang cukup menarik yaitu Push Notification dan Aplikasi Mobile. Dengan adanya dua fitur tersebut, diharapkan dapat memudahkan pengguna.*

**Kata kunci:** Website, E-Commerce, MVC, Model, View, Controller, Push Notification

## 1. Pendahuluan

Indominiatur merupakan toko produk miniatur dan kerajinan tangan yang berada di jalan Parangtritis km 13, Yogyakarta. Saat ini produk-produk miniatur yang ada dalam toko Indominiatur masih dipasarkan secara *offline*, sedangkan sebagai toko yang selalu ingin bergerak maju dibutuhkan media periklanan sekaligus perdagangan yang baik. Karena alasan tersebut penulis tertarik untuk mengembangkan sistem yang masih *offline* ini menjadi *online* agar pasar yang dapat diperoleh menjadi jauh lebih luas.

*Electronic Commerce (E-commerce) merupakan salah satu metode baru dalam berbisnis melalui internet. E-commerce dapat menghubungkan penjual dan pembeli yang berbeda tempat untuk melakukan transaksi. Calon pembeli dapat mengetahui info mengenai produk seperti harga, model, dan warna melalui sebuah website.*

Dari bahasan di atas, penulis menawarkan sebuah solusi untuk memasarkan produk-produk pada toko Indominiatur secara lebih luas dengan judul "Implementasi Pemrograman PHP berbasis Model-View-Controller (MVC) pada website E-commerce (Studi Kasus : [www.indominiatur.com](http://www.indominiatur.com))". Dengan adanya website *e-commerce* ini diharapkan pasar yang diperoleh toko Indominiatur dapat lebih luas sehingga penghasilan yang dapat diperoleh juga menjadi lebih besar.

Penelitian ini bertujuan untuk mendapatkan sistem baru yang dapat memasarkan produk-produk dari toko Indominiatur secara lebih luas. Selain itu, dapat sebagai landasan atau dasar pemikiran pengembangan sistem *e-commerce* pada toko Indominiatur dimasa yang akan datang. Dan juga sebagai salah satu cara untuk memperoleh peningkatan citra toko di kalangan pengguna jasa *internet*.

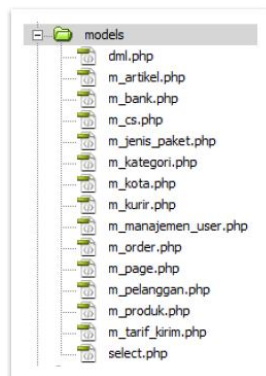
## 2. Pembahasan

Dalam buku M. Suyanto, 2003, *Electronic Commerce (E-commerce)* merupakan konsep baru yang bisa digambarkan sebagai proses jual beli barang atau jasa pada *World Wide Web Internet (Shin, dkk, 2000)* atau proses jual beli produk atau pertukaran produk, jasa, dan informasi melalui jaringan informasi termasuk *Internet (Turban, dkk, 2000)* <sup>[1]</sup>.

*MVC* atau *Model View Controller* adalah sebuah konsep yang memisahkan antara aplikasi logika dengan presentasi pada halaman web. Sehingga hal ini akan menyebabkan halaman *web* akan mengandung kode yang sedikit karena sudah terjadi pemisahan antara tampilan dan pemrograman <sup>[2]</sup>.

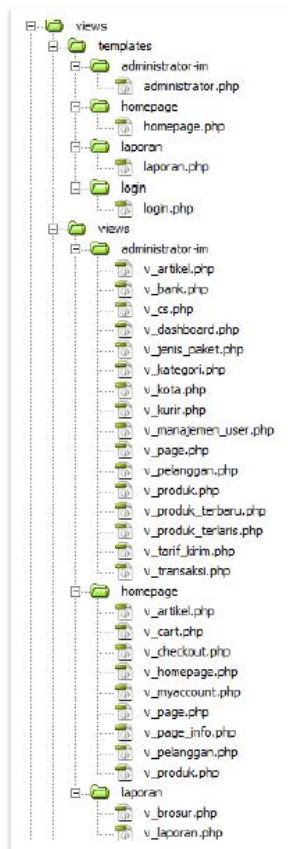


Berikut ini merupakan hasil implementasi bagian *model* pada sistem *e-commerce* toko indominiatur.



Gambar 2.5 Stuktur File pada bagian Model

Selanjutnya adalah implementasi bagian *view* pada sistem *e-commerce* toko indominiatur seperti berikut.



Gambar 2.6 Stuktur File pada bagian View

### 2.1.2 Konfigurasi Framework Codeigniter pada Toko Indominiatur

Berikut ini merupakan konfigurasi file *config.php* dan *database.php* pada Framework Codeigniter. File *config.php* dan *database.php* terletak di dalam folder *config*. Berikut ini adalah settingan file *config.php*.

```
$url_path = "http://";
$url_path .= $_SERVER['SERVER_NAME']. "";
$url_path .= str_replace("index.php", "", $_SERVER['SCRIPT_NAME']);

$config['base_url'] = $url_path;
```

Script `$config['base_url']` digunakan untuk menginialisasikan variable “*base\_url*” yang berisi *path server* dimana aplikasi ini diinstall. Sedangkan untuk file *database.php* settingannya sebagai berikut.

```
$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'root';
$db['default']['password'] = '';
$db['default']['database'] = 'indominiatur_db';
$db['default']['dbdriver'] = 'mysql';
$db['default']['dbprefix'] = '';
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = '';
$db['default']['char_set'] = 'utf8';
$db['default']['dbcollat'] = 'utf8_general_ci';
$db['default']['swap_pre'] = '';
$db['default']['autoinit'] = TRUE;
$db['default']['stricton'] = FALSE;
```

### 2.1.3 Pembahasan Listing Program pada Website

#### 1. Login Administrator

Script program yang digunakan untuk verifikasi data ketika *login*, apakah *username* tersebut *valid* atau tidak. Berikut ini adalah *script* proses *login* administrator pada bagian *controller*.

```
function proses_login(){
    $table = 't_user_admin';
    // Ambil Data dari Form

    $dtField = array(
        'username' => $this->input->post('username'),
        'pass' => $this->input->post('password')
    );

    // Select Database
    $datauser = $this->Select->get_data_query($table,$dtField);

    if($datauser->num_rows() > 0)
    {
        // Ambil Baris , untuk ambil nama User
        $datauser = $datauser->row();

        $data = array(
            'nama' => $datauser->nama,
            'login' => TRUE,
            'hak' => $datauser->hak
        );

        $this->session->set_userdata($data);
        redirect('administrator-im/dashboard');
    }
    else
    {
        $this->session->set_flashdata('message', 'Maaf, username
        redirect('administrator-im/login');
```

#### 2. Add to Cart

*Add to Cart* merupakan fitur menambahkan barang ke keranjang belanja calon pembeli. Pada proses ini, penulis menggunakan teknologi *AJAX* untuk membuat produk dapat masuk ke dalam keranjang belanja tanpa harus *refresh* halaman lagi.

```
function ajaxCartAdd(sidProduk) {
    $rowID = "";
    $produk = $this->m_produk->get_produk_id($sidProduk);

    $produkDipesan = $this->m_order->cek_produk_dipesan($sidProduk);

    if(count($this->cart->contents()) > 0)
    {
        foreach($this->cart->contents() as $scart)
        {
            $totalproduktersedia = ($produk[0]['stok'] - $produkDipesan);

            if($scart['id'] == $sidProduk)
            {
                if(($scart['qty'] + 1) > $totalproduktersedia)
                {
                    $this->session->set_flashdata('message', 'Maaf un-
                    nya tersedia '.$scart['qty'].' item. ');
                    exit;
                }
            }

            $rowID = $scart['rowid'];
            $data = array(
                'rowid' => $rowID,
                'qty' => $scart['qty'] + 1
            );

            $this->cart->update($data);

            break;
        }
    }
}
```

### 3. Push Notification

Push notification merupakan fitur bagi admin yang sangat berguna untuk mengetahui apabila pelanggan melakukan aktifitas pembelian, konfirmasi pembayaran, kirim testimonial, maupun komentar artikel. Fitur ini hampir sama dengan fitur notifikasi pada beberapa jejaring sosial pada umumnya. Jadi dengan demikian admin tidak perlu melakukan refresh halaman untuk mengetahui apakah ada pelanggan yang membeli produknya atau tidak. Fitur push notification ini dibuat dengan mengombinasikan javascript framework jquery dan prototype. Berikut adalah potongan script dari fitur ini.

```
$filename = realpath('./asset/push/notif-log/data.txt');
$msg = '<a href="'.base_url().'index.php/administrat

file_put_contents($filename,$msg);
```

Script diatas digunakan untuk mengubah isi dari file data.txt yang berisi notifikasi dari pelanggan, sedangkan dibawah ini merupakan script untuk mendeteksi perubahan file tersebut dan menampilkannya sebagai notifikasi secara real-time.

```
handleResponse: function(response)
{
    if(stat != 0)
    {
        notif(response['msg']);
        notifTestiProduk();
        notifPesanan();
        notifKonfirmasi();
        notifkomentar();
    }
    stat = 1;
    //alert(response['msg']);
},

doRequest: function(request)
{
    new Ajax.Request(this.url, {
        method: 'get',
        parameters: { 'msg' : request }
    });
}
```

## 2.1.3 Pembahasan Listing Program pada Aplikasi Mobile

### 1. Login

Teknologi Phonegap saat ini belum mampu membaca file php, oleh karena itu penulis memanfaatkan fungsi-fungsi javascript dan JSON (JavaScript Object Notation) untuk memperoleh data dari database Toko Indominitur dan mencocokkannya dengan username yang dimasukkan oleh pelanggan. Berikut ini adalah fungsi-fungsi javascript yang digunakan.

```
<!-- JS -->
<script src="import/js/jquery.js"> </script>
<script src="import/cordova-1.5.0.js"></script>
<script src="import/js/jquery.mobile-1.0.1.js"> </script>
<script src="import/js/heartcode-canvasloader-min-0.9.1.js"></script>
<!-- END JS -->
```

```
function prosesLogin(){
    var a= baseurl + "ajaxLogin/prosesLogin" + "/" + $("#username")
    $.ajax ({
        url: a,
        dataType: 'json',
        timeout: 10000,
        success: function(data, status){
            if(data[0].datalog == 1)
            {
                localStorage.setItem("id_pel", data[0].id_pel);
                localStorage.setItem("nama", data[0].nama);
                localStorage.setItem("login_pel", data[0].datalog);

                window.location='main.html';
            }
            else
            {
                $("#errorMes").html(strAlertWrongPass);
            }
        }
    });
}
```

### 2. Produk

Untuk menampilkan produk dari database Toko Indominitur caranya hampir sama dengan proses login. Penulis masih menggunakan fungsi javascript dan JSON (JavaScript Object Notation). Berikut ini adalah script untuk menampilkan produk katalog.

```
function getProduk(){
    var a= baseurl + "ajaxData/getDataProduk";
    var dataProduk;
    $.ajax ({
        url: a,
        dataType: 'json',
        timeout: 10000,
        success: function(data, status){
            $("#product").html(data[0].dataprod);
        },
        error: function(){
            //error loading data
            $("#product").html(strAlertNoConnection);
        }
    });
}

function getDataProduk(){
    $dataProduk = '';
    $data = $this->m_produk->get_produk();
    $n = count($data);

    if($n > 0){
        for($i=0;$i<$n;$i++){
            $fotoproduk = $this->m_produk->get_fotoproc

            $file = "./uploads/produk/".$data[$i]['id_p

            if(count($fotoproduk) > 0 && file_exists($f

        }

        $json = json_encode($dataAll);

        echo isset($_GET['callback'])
        ? "{$_GET['callback']}($json)"
        : $json;
    }
}
```

### 3. Add to Cart

Pada fitur ini penulis memanfaatkan teknologi *HTML5* yaitu *local storage*. Jadi barang yang di masukkan ke keranjang belanja pada aplikasi *mobile* akan di simpan dengan menggunakan fungsi *local storage*. Selain untuk menyimpan data produk yang ada dalam keranjang fitur ini juga dipakai untuk menyimpan data pelanggan ketika sudah *login*. Berikut ini adalah potongan *script* pada fitur *add to cart*.

```

$.ajax ({
    url: reqStok,
    dataType: 'jsonp',
    timeout: 10000,
    success: function (data, status) {
        stokTersedia = data[0].datastok ;

        if (stokTersedia > 0)
        {
            var reqDetail= baseUrl + "ajaxCart/addtocart/" + idPr;
            $.ajax ({
                url: reqDetail,
                dataType: 'jsonp',
                timeout: 10000,
                success: function (data, status) {

                    localStorage.setItem("noCart",parseFloat(noCa
                    noCart = window.localStorage.getItem("noCart")
                    localStorage.setItem("item-"+noCart, JSON.str
                    //alert (noCart);
                    for (i=1;i<= (parseFloat (noCart)-1);i++)
                    {
                        //alert (i);

                        var dat = JSON.parse(window.localStorage.

                        insert = i;
                        if (insert == (parseFloat (noCart)-1))
                        {
                            insert = parseFloat (noCart) + 1;
                        }

                        error: function () {
                            //error loading data
                            alert ('Error Update');
                        }
                    });
                }
            });

            //alert (insert);
            if (insert > (parseFloat (noCart)) || insert == 0)
            {
                cekLogin ();
            }
        },
        error: function () {
            //error loading data
            alert ('Error Cek Stok');
        }
    });
}
},
error: function () {
    //error loading data
    alert ('Error Cek Stok');
}
}

```

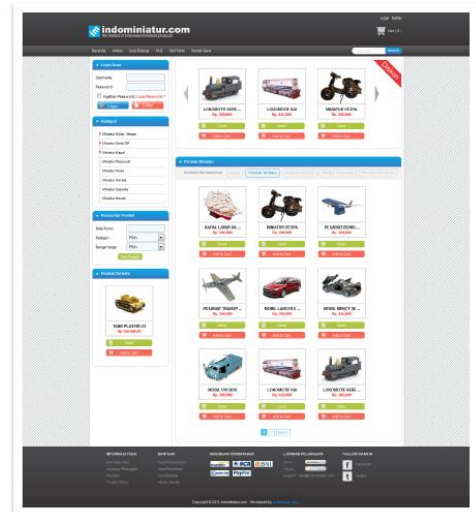
## 2.2 Hasil Implementasi

### 2.2.1 Aplikasi Website

#### 2.2.1.1 Halaman Publik

##### 1. Beranda

Halaman beranda berisi informasi terbaru yang ada di Toko Indominiatur.



Gambar 2.7 Halaman Beranda

### 2. Menu My Cart

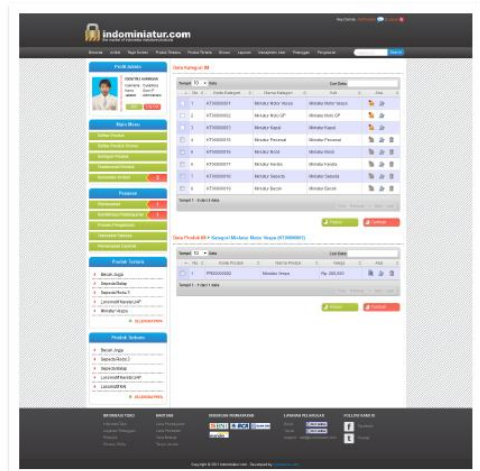
Menu *my cart* merupakan menu untuk menampilkan keranjang belanja.



Gambar 2.8 Menu My Cart

#### 2.1.1.1 Halaman Administrator

##### 1. Beranda



Gambar 2.9 Beranda Administrator

## 2. Fitur Notifikasi



Gambar 2.10 Fitur Notifikasi

merupakan solusi yang tepat untuk membuat seorang administrator atau karyawan dapat memperoleh informasi secara cepat dan tepat waktu.

Saran yang dapat diusulkan penulis terhadap penelitian selanjutnya adalah fitur dari aplikasi *mobile* sistem ini masih dapat di kembangkan lagi agar lebih baik, misalnya dengan menambahkan fitur *barcode scanner*. Selain itu, aplikasi *mobile* pada sistem ini dapat dikembangkan sehingga dapat diimplementasikan pada 6 *platform mobile* berbeda yang masuk dalam dukungan *Phonegap*, yaitu *IOS*, *Android*, *Blackberry*, *Windows Phone*, *WebOS*, dan *Symbian*.

### 2.1.2 Aplikasi Mobile

#### 2.1.2.1 Halaman Utama



Gambar 2.11 Halaman Utama

### Daftar Pustaka

- [1] Suyanto, M. 2003. Strategi Periklanan pada e-Commerce Perusahaan Top Dunia. Yogyakarta : ANDI.
- [2] Wismakarma, Komang. 2010. 9 Langkah Menjadi Master Framework Codeigniter. Yogyakarta: Lokomedia.
- [3] Hakim, Lukmanul. 2010. Bikin Website Super Keren Dengan PHP dan jQuery. Yogyakarta : Lokomedia.
- [4] Basuki, Awan Pribadi. 2010. Membangun Web Berbasis php dengan Framework Codeigniter. Yogyakarta: Lokomedia.
- [5] Nugroho, Adi. 2005. Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek. Bandung : Informatika
- [6] Purbo, Ono W. 1998. Membangun Situs E-Commerce. Jakarta : Elex Media Komputindo.

#### 2.1.2.2 Keranjang Belanja



Gambar 2.12 Keranjang Belanja

### Biodata Penulis

**Donni Prabowo**, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Sistem Informasi STMIK AMIKOM Yogyakarta, lulus tahun 2012. Saat ini sedang menempuh Program Pasca Sarjana Magister Teknik Informatika STMIK AMIKOM Yogyakarta.

**Beti Wulan Sari**, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Sistem Informasi STMIK AMIKOM Yogyakarta, lulus tahun 2012. Saat ini sedang menempuh Program Pasca Sarjana Magister Teknik Informatika STMIK AMIKOM Yogyakarta.

**Pendi Ventri Hendika**, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Sistem Informasi STMIK AMIKOM Yogyakarta, lulus tahun 2012. Saat ini sedang menempuh Program Pasca Sarjana Magister Teknik Informatika STMIK AMIKOM Yogyakarta.

## 3. Kesimpulan

Beberapa kesimpulan yang dapat ditarik dari penelitian ini adalah dengan sistem berarsitektur *MVC (Model View Controller)*, pengembangan sistem menjadi sistem yang lebih besar menjadi lebih mudah dilakukan karena pemisahan antara *model*, *view*, dan *controller*. Aplikasi *mobile* yang diimplementasikan dengan menggunakan *Framework Phonegap* pada sistem ini merupakan sebuah solusi yang tepat untuk mempermudah pengguna perangkat *mobile* memperoleh informasi mengenai produk-produk terbaru. Selain itu, dengan *Framework Phonegap* aplikasi menjadi mudah dikembangkan ke *platform* lain. Dan fitur *push notification* pada sistem ini