

**PENERAPAN METODE *MODEL VIEW CONTROLLER* (MVC) PADA
SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB**

Dikwan Moeis

Program Studi Sistem Informasi, STMIK Profesional

dikwan.moeis@gmail.com

Abstrak

Penelitian ini bertujuan untuk: (1) Membuat sistem informasi perpustakaan berbasis web yang digunakan untuk mendukung operasi manajemen perpustakaan agar segala informasi yang ada di perpustakaan dapat diakses dari mana saja tanpa dibatasi oleh jarak dan waktu sehingga informasi tersebut dapat dimanfaatkan secara optimal baik oleh pengelola perpustakaan maupun pengunjung dalam mendukung aktifitasnya.

Model pengembangan perangkat lunak yang digunakan pada sistem ini adalah menggunakan Iterative Waterfall Model yang memungkinkan developer untuk mendesain ulang dan merencanakan ulang pengembangan sistem dengan syarat perubahan tidak sampai mengubah tujuan utama dibuatnya sistem. Dalam pengembangannya diterapkan metode Model View Controller (MVC) agar dapat meningkatkan modularitas dan reusabilitas dari sistem, kompleksitas dari source code (kode sumber) dalam perangkat lunak dapat dikurangi secara signifikan dan juga dapat mempermudah perawatan di masa yang akan datang.

Hasil penelitian ini adalah sebuah perangkat lunak sistem informasi perpustakaan berbasis web yang dibangun dengan menerapkan metode Model View Controller (MVC). Sistem ini membantu pengelola perpustakaan dalam operasi manajemen perpustakaan.

Kata Kunci : MVC, Model View Controller, Perpustakaan, Framework CodeIgniter

A. PENDAHULUAN

Salah satu bagian di perguruan tinggi yang membutuhkan sistem informasi yang akurat dan cepat adalah pada bagian perpustakaan. Perpustakaan merupakan sarana sumber ilmu pengetahuan dari berbagai disiplin ilmu. Perpustakaan biasanya difungsikan oleh pengunjung sebagai media untuk mencari referensi dan memperoleh informasi.

Dalam prosesnya, perpustakaan membutuhkan suatu sistem untuk mengumpulkan data, mengolah data, menyimpan data, melihat kembali data dan menyalurkan informasi yang baik, salah satunya adalah memiliki keakuratan data yang tinggi. Untuk memenuhi kebutuhan tersebut, usaha yang harus dilakukan oleh pengelola perpustakaan adalah memanfaatkan teknologi informasi seperti komputer beserta perangkat lunak sistem

informasi perpustakaan disamping peningkatan sumber daya manusia dan peningkatan sistem.

Sistem informasi perpustakaan merupakan sebuah sistem yang terintegrasi untuk menyediakan informasi guna mendukung operasi manajemen dan fungsi pengambilan keputusan. Sistem informasi perpustakaan merupakan perangkat lunak yang didesain khusus untuk mempermudah pendataan koleksi perpustakaan, katalog buku, data anggota/peminjam, transaksi dan sirkulasi peminjaman/pengembalian buku. Keseluruhannya bekerja secara sistematis sehingga dapat memperbaiki administrasi dan operasional perpustakaan serta dapat menghasilkan bentuk-bentuk laporan yang efektif dan berguna bagi manajemen perpustakaan.

Web sebagai bagian dari teknologi informasi mempunyai peran dalam organisasi perpustakaan. Teknologi web dapat dimanfaatkan sebagai sarana untuk mengkomunikasikan kepada pengunjung tentang berbagai hal yang ada di perpustakaan, yaitu sebagai sistem informasi perpustakaan berbasis web. Penerapan sistem informasi perpustakaan berbasis web sangat diperlukan untuk mendukung tujuan perpustakaan agar segala informasi yang ada di perpustakaan dapat diakses dari mana saja tanpa dibatasi oleh jarak dan waktu sehingga informasi tersebut dapat dimanfaatkan secara optimal baik oleh pengelola perpustakaan maupun pengunjung dalam mendukung aktifitasnya.

Seiring dengan perkembangan teknologi dan tingkat fleksibilitas informasi yang berdampak pada semakin berkembangnya informasi melalui jaringan internet maka sistem informasi perpustakaan berbasis web tentu harus dapat beradaptasi dengan perkembangan tersebut, salah satu elemen yang perlu diperhatikan adalah sistem informasi tersebut dituntut untuk berjalan cepat, tepat dan mudah dalam pengembangan. Dengan alasan tersebut, maka perancangan dan pembangunan sistem informasi perpustakaan dirasakan perlu dikembangkan dengan berbasis web dan dengan menerapkan metode MVC (*Model View Controller*), dengan menggunakan metode ini pengembangan perangkat lunak yang besar mudah untuk dilakukan perawatan (perbaikan, penambahan atau pengurangan kode).

Model View Controller (MVC) merupakan arsitektur yang memisahkan dengan jelas antara data (*model*) dengan *user interface* (*view*), dan telah terbukti sangat efektif (Widiyanto, 2010). Penggunaan MVC memungkinkan penyusunan *source code* (kode sumber) aplikasi menjadi lebih rapi karena terjadi pemisahan dengan jelas antara *business logic* (logika bisnis) dengan *user interface* (antarmuka pengguna), sehingga dapat mengurangi kompleksitas *source code* (kode sumber), meningkatkan fleksibilitas dan

Dikwan Moeis – Penerapan Metode Model View Controller (MVC) modularitas perangkat lunak (Uyun dan Ma'arif, 2010), selain itu juga dapat mempermudah perawatan di masa yang akan datang (Utpatadevi, Sudana & Cahyawan, 2012). Dengan menggunakan MVC, alur aplikasi lebih mudah dibaca dan lebih mudah ditelusuri jika terjadi kesalahan (Martha, Harianto & Asfi, 2010).

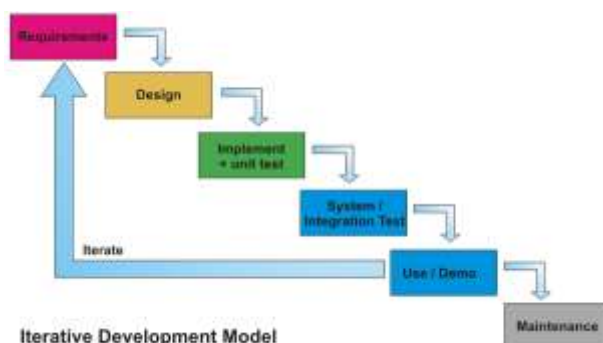
Berdasarkan dari uraian di atas, maka penulis menentukan rumusan masalah dalam penelitian ini adalah bagaimana membuat sistem informasi perpustakaan berbasis web dengan menerapkan metode *Model View Controller* (MVC)?

Untuk menjawab rumusan masalah tersebut, dalam penelitian ini akan dibuat sebuah sistem informasi perpustakaan berbasis web dengan menerapkan metode *Model View Controller* (MVC), dengan menerapkan metode ini pengembangan perangkat lunak di masa yang akan datang dapat lebih mudah dilakukan.

B. METODE PENELITIAN

Model Pengembangan Sistem

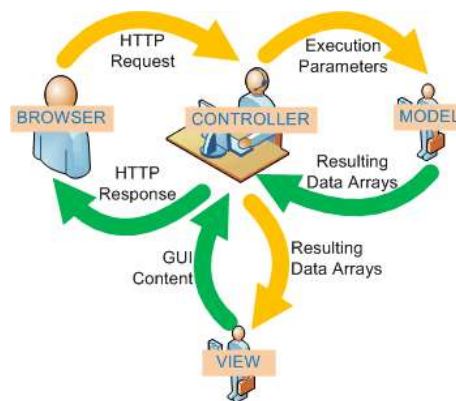
Model pengembangan yang digunakan pada sistem ini adalah menggunakan *Iterative Waterfall Model*. *Iterative Waterfall Model* adalah pengembangan dari *Waterfall Model*. *Waterfall Model* sendiri adalah alur pengembangan sistem dimulai dari *planning, analysis, design, implementation, operation & maintenance*. *Waterfall model* dikembangkan lagi menjadi *iterative waterfall model* yang memungkinkan *developer* untuk mendesain ulang dan merencanakan ulang pengembangan sistem dengan syarat perubahan tidak sampai mengubah tujuan utama dibuatnya sistem. Gambar dari *iterative waterfall model* dapat dilihat pada Gambar 1.



Gambar 1. Iterative Waterfall Model

Metode Model View Controller (MVC)

Metode dalam pengembangan sistem dibuat dengan arsitektur MVC (*Model View Controller*). *Model View Controller* adalah sebuah metode untuk pengembangan sebuah aplikasi web dengan memisahkan data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*). Dalam implementasinya kebanyakan *framework* (kerangka kerja) dalam aplikasi website adalah berbasis arsitektur MVC. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian yang menjadi kontrol dalam sebuah aplikasi web. Gambar dari arsitektur MVC dapat dilihat pada gambar 2.



Gambar 2. Arsitektur MVC

Jenis Data dan Teknik Pengumpulan Data

Jenis data yang akan digunakan dalam penelitian adalah sebagai berikut:

a. Data Primer

Pengambilan data primer dilakukan melalui cara observasi, yaitu penulis melakukan pengamatan langsung mengenai fungsi dan kegiatan mendasar pada sebuah perpustakaan.

b. Data Sekunder

Guna mendapatkan data, gambaran dan keterangan yang lebih lengkap penulis melakukan studi literatur dengan cara mempelajari laporan-laporan, makalah-makalah, jurnal-jurnal, dokumen-dokumen serta pengumpulan dan penelaahan hasil-hasil penelitian yang berhubungan dengan penelitian.

Pengumpulan data yang baik dalam sebuah penelitian dipengaruhi oleh cara memperoleh data, dan harus mengikuti metode atau teknik yang sesuai dengan permasalahan penelitian yang dibahas. Teknik pengumpulan data yang digunakan dalam penelitian ini adalah observasi dan studi kepustakaan.

C. HASIL

Spesifikasi Perangkat Keras dan Perangkat Lunak Untuk Menjalankan Aplikasi

1. Sistem operasi Microsoft Windows XP
2. Processor Intel Pentium Dual-Core 133 Mhz
3. RAM 1 GB DDR3
4. Space Disk 500 MB
5. Framework PHP CodeIgniter
6. Framework CMS Bootstrap
7. XAMPP 1.8.2
8. Browser

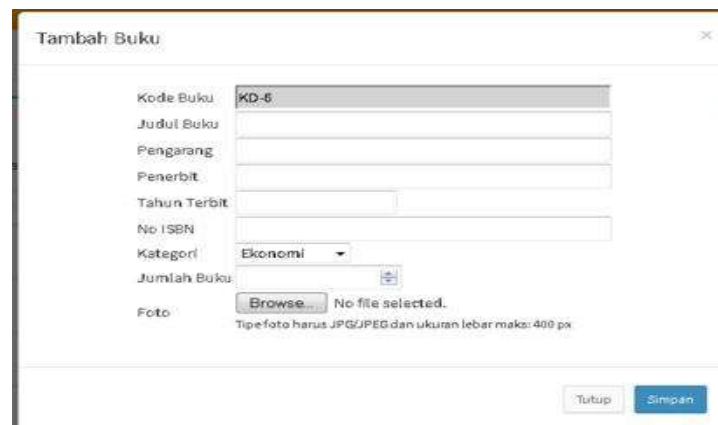
Hasil Pembuatan Aplikasi

Berikut beberapa tampilan form/halaman aplikasi:



Gambar 4. Halaman login

Form *login* digunakan oleh petugas perpustakaan untuk melakukan berbagai aktifitas mulai dari penambahan jumlah buku juga transaksi kegiatan peminjaman dan pengembalian buku. Untuk bisa masuk pada bagian administrator maka petugas perpustakaan harus mengisi *field* nama dan *password* yang sudah didaftarkan sebelumnya.



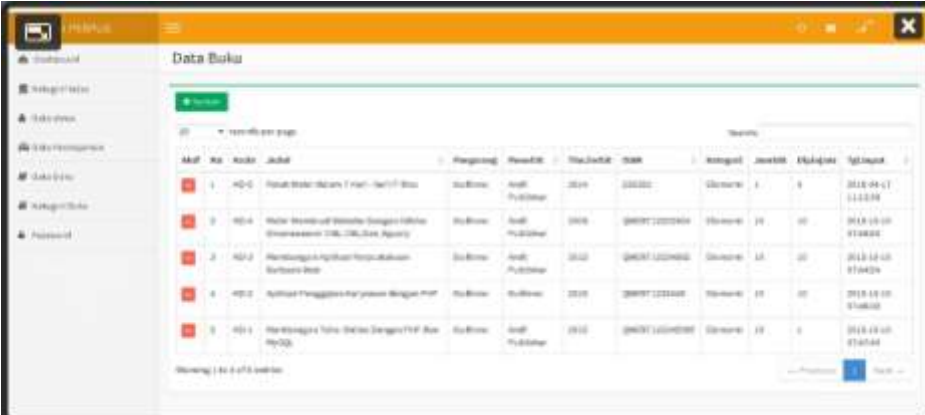
Gambar 5. Halaman *input*/tambah buku

Form *input*/tambah buku berfungsi untuk menambahkan stok buku dan judul-judul buku ke dalam database.



Gambar 6. Halaman *edit*/koreksi data buku

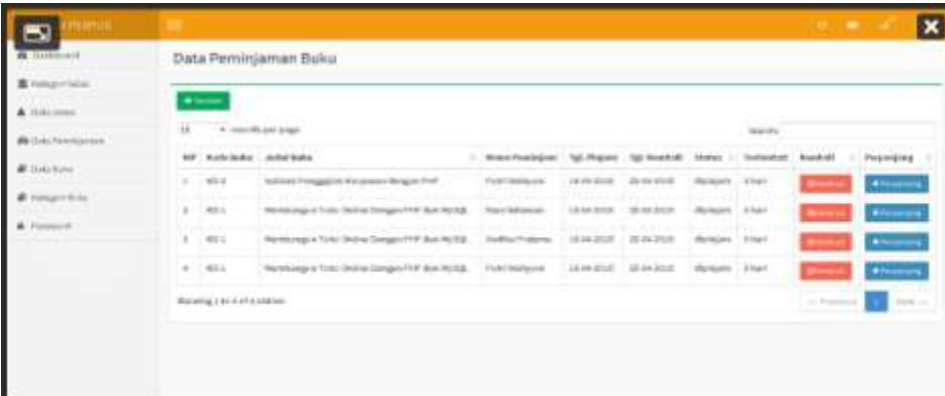
Form edit/koreksi buku berfungsi untuk memperbaiki data buku jika terjadi kesalahan pada saat penambahan data.



Aktif	No	Kode	Judul	Pengarang	Penerbit	Tahun Terbit	ISBN	Kategori	Jumlah	Stok	Tanggal
<input type="checkbox"/>	1	KD-5	Paket Mahir dalam 7 Hari - Seri IT Bin1	Budiono	Andi Publisher	2014	232323	Ekonomi	1	0	2018-04-17 11:13:34
<input type="checkbox"/>	2	KD-4	Water Resistant Wireless Design Inkless Smartpen (SR-100, 2.0k, 2.0k, 2.0k)	Budiono	Andi Publisher	2018	240371222949	Ekonomi	10	10	2018-10-10 15:44:22
<input type="checkbox"/>	3	KD-2	Perangkat Lunak Perawatan Kulit Elektronik	Budiono	Andi Publisher	2012	240371222949	Ekonomi	10	10	2018-10-10 15:44:29
<input type="checkbox"/>	4	KD-2	Perangkat Perawatan Kulit Elektronik	Budiono	Andi Publisher	2012	240371222949	Ekonomi	10	10	2018-10-10 15:44:30
<input type="checkbox"/>	5	KD-1	Perangkat Tahan Guncan Dengan MP3 dan MP3	Budiono	Andi Publisher	2012	240371222949	Ekonomi	10	0	2018-10-10 15:44:34

Gambar 7. Halaman lihat data buku

Form lihat data buku berfungsi untuk melihat data-data buku yang dimiliki oleh Perpustakaan.



Aktif	Kode Buku	Judul Buku	Nama Peminjam	Tgl. Peminjam	Tgl. Kembali	Status	Instansi	Peminjam
<input type="checkbox"/>	KD-2	Perangkat Perawatan Kulit Elektronik	Puri Hartono	18-04-2018	20-04-2018	Belum	IT-1	<input type="button" value="Hapus"/> <input type="button" value="Peminjam"/>
<input type="checkbox"/>	KD-1	Perangkat Tahan Guncan Dengan MP3 dan MP3	Puri Hartono	18-04-2018	20-04-2018	Belum	IT-1	<input type="button" value="Hapus"/> <input type="button" value="Peminjam"/>
<input type="checkbox"/>	KD-2	Perangkat Tahan Guncan Dengan MP3 dan MP3	Puri Hartono	18-04-2018	20-04-2018	Belum	IT-1	<input type="button" value="Hapus"/> <input type="button" value="Peminjam"/>
<input type="checkbox"/>	KD-1	Perangkat Tahan Guncan Dengan MP3 dan MP3	Puri Hartono	18-04-2018	20-04-2018	Belum	IT-1	<input type="button" value="Hapus"/> <input type="button" value="Peminjam"/>

Gambar 8. Halaman lihat data peminjaman buku

Form lihat data peminjaman buku berfungsi untuk melihat data-data buku yang terpinjam.

D. PEMBAHASAN

Instalasi CodeIgniter

Aplikasi yang dibangun dengan *framework* Codeigniter memerlukan paket *framework* Codeigniter. Sebelum melakukan instalasi yang perlu dilakukan pertama kali adalah mengunduh *file* codeigniter di <http://codeigniter.com/download.php>, kemudian di ekstrak di web server yang berlokasi di htdocs. dilanjutkan dengan konten untuk kosa kata

Konfigurasi Dasar CodeIgniter

Pada folder perpustakaan/application/config, terdapat file konfigurasi codeigniter. Dalam pembuatan aplikasi ini digunakan tiga file konfigurasi penting yaitu :

1. `konek_db.php`

File ini digunakan untuk pengaturan koneksi database, kode programnya sebagai berikut :

```
$db['default']['hostname'] = 'localhost';  
$db['default']['username'] = 'root';  
$db['default']['password'] = '';  
$db['default']['database'] = 'perpustakaan';  
$db['default']['dbdriver'] = 'mysql';
```

2. `config.php`

Digunakan untuk pengaturan *base* url dan konfigurasi url untuk *homepage*.

```
$config['base_url']='http://localhost/perpustakaan/';
```

3. `routes.php`

File ini digunakan untuk membaca default *controller*.

```
$route['default_controller'] = "main";
```

Pengelolaan Data Anggota

Daftar file pengelolaan data anggota terlihat pada tabel 1 berikut ini:

Tabel 1. Daftar File Pengelolaan Data Anggota

Controller	Model	View
anggota.php	anggota_m.php	ianggota_v.php eanggota_v.php tanggota_v.php

File anggota_m.php berfungsi untuk mengeksekusi perintah *query* di *database*.

Berikut potongan kode programnya :

```
function insert(){
$data = array("tipe" => $this->input->post("tipe"),
"passw" => base64_encode($this->input->post("passw")),
"nama" => $this->input->post("nama"),
"jk" => $this->input->post("jk"),
"kode" => $this->input->post("kode"),
```

File anggota.php merupakan *controller* yang menangani beberapa aksi yaitu *create*, *update*, *delete* dan *index*. Contoh aksi untuk membuat data anggota baru adalah membuat *function add* seperti pada potongan program berikut :

```
parent::__construct();
$this->load->model(array('anggota_m'));
$this->load->library(array('simpliparse','pquery',
'form_validation'));
```

Kode program yang ada pada fungsi diatas menggunakan model Anggota yang didefinisikan pada perintah baris kedua. Perintah ini akan membuat objek baru dan akan membuat data baru pada tabel anggota.

View yang digunakan untuk menangani tampilan aksi *create* ini adalah *view* ianggota_v.php. Berikut potongan kode programnya:

```
<?php echo form_open('admin/anggota/save','id="form"')?>
<fieldset id="personal">
<legend> INPUT DATA ANGGOTA</legend>
<label>Tipe</label>
<input name="tipe" type="radio"
```

Pengelolaan Data Buku

Daftar *file* pengelolaan data buku terlihat pada tabel 2 berikut ini:

Tabel 2. Daftar File Pengelolaan Data Buku

Controller	Model	View
buku.php	buku_m.php	ibuku_v.php ebuku_v.php tbuku_v.php

Model yang digunakan untuk pengelolaan data yaitu model buku_m.php yang merepresentasikan tabel Buku. Aksi yang ada pada *controller* yaitu aksi *create*, *update* dan *delete* buku. Kode programnya sebagai berikut :

```
parent::__construct();
$this->load->model(array('kategori_m','buku_m'));
$this->load->library(array('simpliparse','pquery',
'form_validation','pagination'));
```

```
$this->load->library('fungsi');  
$this->load->helper('html');
```

View yang digunakan untuk menangani tampilan aksi *create* ini adalah *view* `ibuku_v.php`. Berikut potongan kode programnya :

```
<h3 id="adduser">Tambah Buku</h3>  
<?php echo  
form_open_multipart("admin/buku/save", "id='form'");?>  
<fieldset id="personal">  
<legend> INPUT DATA BUKU</legend>  
<label>Kode Buku</label>  
<input name="kode" type="text" class="required"
```

Kode program di atas digunakan untuk menampilkan *form* pengisian data buku.

E. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan rumusah masalah dan hasil penelitian yang telah dilakukan, maka dapat ditarik kesimpulan sebagai berikut:

1. Dari penelitian ini dihasilkan sebuah perangkat lunak sistem informasi perpustakaan berbasis web. Dalam pengembangan sistem tersebut digunakan metode MVC (*Model View Controller*).
2. Penerapan arsitektur MVC (*Model View Controller*) dalam pengembangan sistem informasi perpustakaan berbasis web ini dapat meningkatkan modularitas dan reusabilitas dari sistem, kompleksitas dari *source code* (kode sumber) dalam perangkat lunak dapat dikurangi secara signifikan sehingga mudah disesuaikan dengan kebutuhan di masa yang akan datang. Jika ada perubahan pada tampilan antarmuka, maka tidak perlu mengubah kode sumber yang berhubungan dengan basis data.

Saran

Untuk perbaikan dan penelitian selanjutnya, beberapa saran yang dapat diberikan:

1. Meningkatkan keamanan data dengan menggunakan arsitektur *multitier*.
2. Penyempurnaan pada *view* atau *user interface* (antarmuka pengguna), sehingga lebih menarik dan dapat lebih baik dalam berinteraksi dengan pengguna.

F. DAFTAR PUSTAKA

- [1] Arief, M. Rudyanto. 2012. *Pemrograman Web Dinamis Menggunakan PHP dan MySql*. Andi Publisher. Yogyakarta.

- [2] Huda, M., & Komputer, B. 2009. *Membuat Aplikasi Rental dengan Java dan MySQL*. Elexmedia Komputindo. Jakarta.
- [3] Kadir, Abdul. 2008. *Dasar Pemrograman Web Dinamis Menggunakan PHP-Edisi Revisi*. Andi Publisher. Yogyakarta.
- [4] Martha, D., Harianto, C., & Asfi, M. 2010. *Metode MVC untuk Perancangan Sistem Berorientasi Objek pada Ujian Saringan Masuk Penerimaan Mahasiswa Baru di STMIK CIC Cirebon*. Jurnal Informatika, 145 - 160.
- [5] Nugroho, A. 2009. *Rekayasa Perangkat Lunak Menggunakan UML dan Java*. Penerbit Andi. Yogyakarta.
- [6] Pertiwi, Dini Hari. 2011. *Desain dan Implementasi Sistem Informasi Perpustakaan Berbasis Web Dengan MVC (Model View Controller)*. Jurnal Teknologi dan Informatika (TEKNOMATIKA) Vol. 1 No. 2 Mei 2011.
- [7] Prasetio, Adhi. 2012. *Buku Pintar Pemrograman Web*. Mediakita. Jakarta.
- [8] Pratama, ANW. 2010. *CodeIgniter: Cara Mudah Membangun Aplikasi PHP*. Mediakita. Jakarta.
- [9] Qureshi, M. R., & Sabir, F. 2013. *A Comparison of Model View Controller and Model View Presenter*. The Science International (Lahore), 7-9.
- [10] Simarmata, J. 2010. *Rekayasa Perangkat Lunak*. Penerbit Andi. Yogyakarta.
- [11] Suprianto, Wahyu dan Ahmad Muhsin. 2008. *Teknologi informasi perpustakaan*. Kanisius. Yogyakarta.
- [12] Supriyanto. 2010. *Pemrograman Database Menggunakan Java dan MySQL untuk Pemula*. Mediakita. Jakarta
- [13] Utomo, E. P. 2009. *Panduan Mudah Mengenal Bahasa Java*. Bandung: Yrama Widya.
- [14] Utpatadevi, N. L., Sudana, A. A., & Cahyawan, A. A. 2012. *Implementation of MVC (Model-View-Controller) Architectural to Academic Management Information System with Android Platform Base*. International Journal of Computer Applications, 1-6.
- [15] 'Uyun, S., & Ma'arif, M. R. 2010. *Implementation of Model View Controller (MVC) Architecture on Building Web-based Information System*. Seminar Nasional Aplikasi Teknologi Informasi 2010 (SNATI 2010), E-47 - E-50.
- [16] Wardana. 2010. *Menjadi Master PHP Dengan Framework Codeigniter*. Elexmedia komputindo. Jakarta.

- [17] Widiyanto, N. 2010. *Membangun Aplikasi Java Enterprise dengan Arsitektur Model View Controller (MVC)*. Penerbit Andi. Yogyakarta
- [18] Yulianti. 2013. *Implementasi Arsitektur Client-Server dan MVC (Model View Controller) Untuk Membangun Aplikasi Administrasi Sekolah (Studi Kasus: SMK Avenus Jakarta)*. Skripsi. Jakarta: Program Studi Teknik Informatika Fakultas Teknik Universitas Pamulang.
- [19] (<https://www.yohanli.com/konsep-mvc-model-view-controller.html>, diakses 05 April 2017).
- [20] (<https://blog.codinghorror.com/understanding-model-view-controller/>, diakses 12 April 2017).