

## Rancang Bangun Aplikasi Sistem Informasi Manajemen Laboratorium Biomedik Fakultas Kedokteran Universitas Brawijaya

Ahmad Leo Yudanto<sup>1</sup>, Herman Tolle<sup>2</sup>, Adam Hendra Brata<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>ahmadleoyudanto@gmail.com, <sup>2</sup>emang@ub.ac.id, <sup>3</sup>adam@ub.ac.id

### Abstrak

Laboratorium Biomedik adalah salah satu laboratorium yang terdapat pada fakultas kedokteran universitas brawijaya. Mengingat tingkat kebutuhan yang tinggi akan manajemen berbagai layanan di laboratorium biomedik yang terus berkembang serta kebutuhan untuk inventaris data, monitoring stok bahan dan peralatan, maka perlu dibangun ulang sistem informasi manajemen laboratorium yang dapat memfasilitasi semua kebutuhan laboratorium dengan tetap memberikan kemudahan dan keluwesan untuk pengembangan pada waktu-waktu mendatang. Sistem informasi manajemen laboratorium biomedik perlu dirancang ulang sesuai dengan kebutuhan pengguna berdasarkan tahapan analisis kebutuhan yang komprehensif dan perancangan yang mendetail agar nantinya dapat diimplementasi dengan lengkap dan sesuai dengan kebutuhan pengguna. Teknologi yang digunakan pada penelitian ini adalah framework laravel 5 yang dibangun menggunakan bahasa pemrograman PHP dan menggunakan server apache sebagai server lokal yang nantinya akan dijalankan pada beberapa komputer yang berada pada lingkungan laboratorium. Penelitian ini menghasilkan sistem informasi manajemen yang dapat melakukan proses inventaris data, monitoring stok bahan dan peralatan serta telah memenuhi proses bisnis dan kebutuhan fungsional laboratorium yang telah diuji dengan menggunakan metode pengujian perangkat lunak.

**Kata Kunci:** PHP, Framework, Laravel 5, Sistem Informasi Manajemen

### Abstract

*Biomedical Laboratory is one of laboratories at Brawijaya University Medical Faculty. Considering the eminent level of requirement for laboratory's various services management which continues to grow along with the need for a data inventory and monitoring of materials and equipment stock, it is necessary to rebuilt laboratory management information system to facilitate every laboratory requirements while providing simplicity and flexibility for future development. Biomedical laboratory management information system has to be redesigned according to user requirements based on comprehensive requirements analysis phase with detailed design so it can be completely implemented and in accordance with user requirements. This technologies are Laravel 5 framework which built by PHP programming language and Apache server as a local server which would be installed on several computers at laboratory. This research produces an information system which can perform data inventory process, stock monitoring of materials and tools and also informations system already meet laboratorium's business process and functionality requirement that already have been tested by software testing method.*

**Keywords:** PHP, Framework, Laravel 5, Management Information System.

### 1. PENDAHULUAN

Proses manajemen pengelolaan laboratorium atas berbagai layanan yang diberikan oleh Laboratorium Biomedik selama ini telah memanfaatkan teknologi informasi. Akan tetapi sistem informasi yang telah ada dan

digunakan saat ini dipandang sulit untuk dikembangkan mengikuti kebutuhan operasional yang terus berkembang pada laboratorium biomedik.

Mengingat tingkat kebutuhan yang tinggi akan manajemen pengelolaan berbagai layanan di laboratorium biomedik yang terus

berkembang serta kebutuhan untuk inventaris data, monitoring stok bahan dan peralatan, maka laboratorium biomedik bermaksud untuk membangun ulang suatu sistem informasi manajemen laboratorium yang dapat memfasilitasi semua kebutuhan laboratorium dengan tetap memberikan kemudahan dan keluwesan untuk pengembangan pada waktu-waktu mendatang.

Sistem informasi manajemen laboratorium biomedik perlu dirancang ulang sesuai dengan kebutuhan pengguna berdasarkan tahapan analisis kebutuhan yang komprehensif dan perancangan yang mendetail agar nantinya dapat diimplementasi dengan lengkap dan sesuai dengan kebutuhan pengguna. Teknologi aplikasi yang digunakan harus dibuat menggunakan teknologi yang memberikan kemudahan dalam implementasi, penggunaan, maupun kemudahan pengembangan lanjut di kemudian hari.

Untuk menjawab permasalahan diatas maka perlu dikembangkan suatu sistem informasi manajemen yang baru dengan pendekatan Aplikasi Web. Menurut Al-Fedaghi (2011), aplikasi web adalah aplikasi yang diakses menggunakan Web Browser melalui sebuah jaringan dan dikembangkan dengan menggunakan bahasa yang didukung oleh browser seperti HTML (Hyper Text Markup Language) dan Javascript. Pengembangan aplikasi web membutuhkan pendekatan yang komprehensif yang mencakup banyak aspek, seperti teknis, organisasi dan ukuran program yang sah. Oleh karena itu, metode memproses informasi, keahlian, dan alat telah di perluas untuk mendukung pengembangan aplikasi.

Perancangan proses bisnis baru yang tepat disertai dengan penggalan kebutuhan user menjadi kunci dari penelitian ini. Diagram yang digunakan, fungsi dan antarmuka yang dibutuhkan untuk mendukung proses bisnis tersebut dan bagaimana penerapannya dengan bahasa pemrograman PHP Framework Laravel akan dijelaskan secara rinci.

Sistem informasi manajemen laboratorium ini diharapkan akan mudah digunakan, fleksibel terhadap perubahan, mempunyai fitur-fitur yang lengkap dan proses pengolahan data yang baik. Laboran dan pelanggan dapat melakukan pekerjaan dan transaksi dengan mudah dan efisien.

## 2. LANDASAN KEPUSTAKAAN

### 2.1 Sistem Informasi

Sebuah Sistem Informasi adalah sebuah sistem kerja yang proses dan kegiatannya untuk memproses informasi, seperti menangkap, mengirimkan, menyimpan, mendapatkan kembali, mengubah dan menampilkan informasi. Sebuah sistem kerja itu sendiri adalah sebuah sistem di mana partisipasi manusia dan/atau mesin melakukan pekerjaan (proses dan aktivitas) menggunakan informasi, teknologi, dan sumber daya lainnya untuk memproduksi produk yang spesifik (Alter, 2008).

### 2.2 Framework

*Framework* adalah sebuah arsitektur yang terbuka yang dibuat berdasarkan pada standar pengembangan perangkat lunak yang diterima secara umum (Shan & Hua, 2006).

Menurut Supaartagorn (2011), Penggunaan *Framework* secara signifikan mengurangi penggunaan waktu, usaha dan sumber daya yang dibutuhkan untuk mengembangkan dan *maintenance* aplikasi web.

### 2.3 Konsep MVC (Model View Controller)

Sebuah *framework* aplikasi web biasanya mengimplementasikan pola desain yang disebut *Model*, *View* dan *Controller* atau yang biasa disebut sebagai MVC.

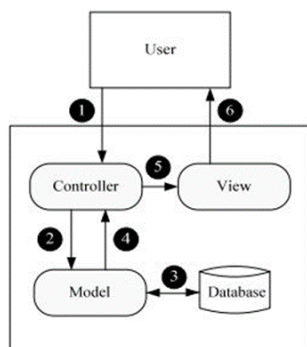
Modul *model* memuat kelas-kelas yang mewakili tabel pada database yang mempunyai *instances* yang digunakan untuk memanipulasi database. *Model* biasanya digunakan sebagai penghubung antara modul *controller* dengan database ketika *controller* ingin mengambil dan menggunakan data di *database*.

Modul *controller* adalah kelas-kelas yang dibuat oleh *programmer* untuk menangani logika program dan *user events*. Pada aplikasi yang menggunakan pola MVC *controller* bertindak sebagai otak dari sistem, menjembatani hubungan antara *model* dan *view*. *Controller* juga berfungsi menerima *request* dari *user* dan kemudian memprosesnya.

Modul *view* berfungsi untuk menerima dan menampilkan data yang dikirim oleh *controller*. *View* dalam aplikasi berbasis *website* biasanya berbentuk kumpulan halaman HTML.

MVC membantu mengurangi kompleksitas dari pembuatan desain dan menambah fleksibilitas dan pemakaian kembali (re-use) kode (Cui et al, 2009). Berikut adalah ilustrasi dari konsep MVC seperti yang ditunjukkan pada

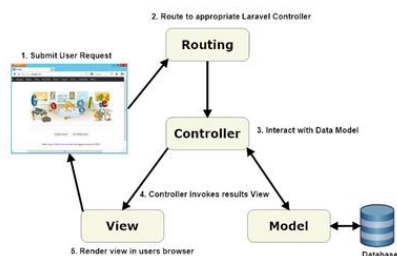
Gambar 1.



Gambar 1. Ilustrasi MVC

## 2.4 Laravel

Laravel adalah sebuah *framework* web berbasis PHP yang *open-source* dan tidak berbayar, diciptakan oleh Taylor Otwell dan diperuntukkan untuk pengembangan aplikasi web yang menggunakan pola MVC. Struktur pola MVC pada laravel sedikit berbeda pada struktur pola MVC pada umumnya. Di laravel terdapat *routing* yang menjembatani antara *request* dari *user* dan *controller*. Jadi *controller* tidak langsung menerima *request* tersebut. Berikut adalah ilustrasi dari konsep MVC pada laravel yang ditunjukkan pada Gambar 2.



Gambar 2. Ilustrasi MVC Laravel

## 3. METODOLOGI

Penelitian ini bertipe implementatif pembangunan, karena penelitian ini mengimplementasikan rancangan yang sudah dibuat menjadi sebuah sistem informasi. Penelitian ini dimulai pada tahap analisis kebutuhan, dilanjutkan dengan perancangan dan kemudian implementasi sistem.

Selanjutnya sistem yang telah final, di instalasi dan diuji pada lingkungan penggunaan sebenarnya yaitu di laboratorium biomedik FKUB. Untuk lebih jelasnya dapat dilihat pada Gambar 3.



Gambar 3. Metodologi penelitian

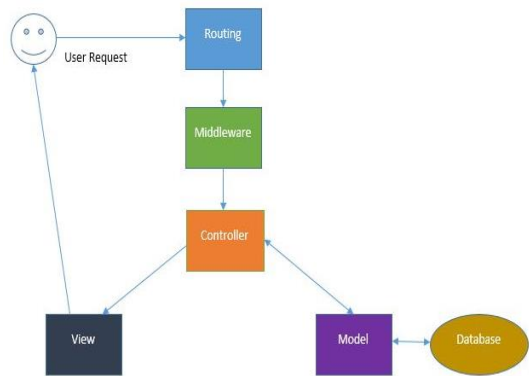
## 4. PERANCANGAN

### 4.1 Deskripsi Singkat Perangkat Lunak

Sistem informasi manajemen laboratorium biomedik FKUB adalah perangkat lunak yang dikembangkan dengan tujuan untuk memenuhi kebutuhan dan proses bisnis baru dari laboratorium biomedik FKUB dengan bantuan teknologi informasi. Adapun beberapa kebutuhan tersebut antara lain adalah kelola data *item* seperti bahan dan alat, kelola data pelanggan, kelola data transaksi peneliti dan pasien serta monitoring laporan buku kas pemasukan dan pengeluaran laboratorium. Sistem informasi memiliki 3 tipe pengguna, yaitu administrator, laboran dan operator logistik. Sistem informasi akan di implementasikan di lingkungan laboratorium biomedik FKUB. Berikut adalah gambaran dari arsitektur perangkat lunak seperti yang ditunjukkan pada Gambar 4.

4.2 Identifikasi Aktor

Pada tahap ini akan dilakukan identifikasi aktor beserta tanggung jawabnya saat menggunakan sistem informasi manajemen laboratorium biomedik FKUB. Detail identifikasi aktor dapat dilihat pada Tabel 1.



Gambar 4. Arsitektur perangkat lunak

Tabel 1. Identifikasi aktor

No.	Aktor	Tanggung Jawab
1.	Administrator	Aktor yang bertugas untuk mengelola data pelanggan, data laboran, data transaksi peneliti dan pasien dan monitoring kas pemasukan dan pengeluaran laboratorium.
2.	Operator Logistik	Aktor yang bertugas untuk mengelola data item, data pembelian bahan, data supplier, penggunaan item bahan pada transaksi peneliti dan pasien.
3.	Laboran	Aktor yang bertugas untuk mengelola penggunaan item alat dan jasa pada transaksi peneliti dan pasien.

4.3 Analisis data

Tahapan analisis data dibutuhkan untuk menjelaskan data-data utama yang digunakan oleh sistem informasi manajemen laboratorium biomedik FKUB. Terdapat 5 data utama yang direpresentasikan menjadi beberapa tabel. Detail analisis data dapat dilihat pada Tabel 2 berikut.

Tabel 2. Analisis data

No.	Tabel	Deskripsi
1.	Data Pelanggan	Berisi data diri dari pelanggan yang telah terdaftar.

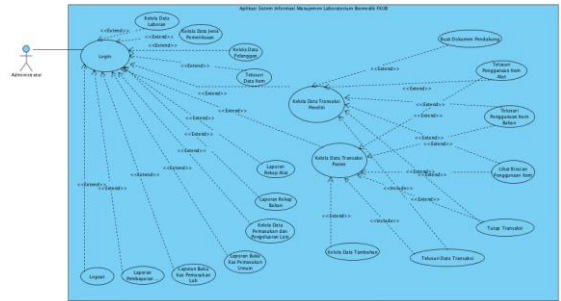
2.	Data Transaksi Peneliti	Berisi data transaksi pelanggan yang mendaftar sebagai peneliti.
3.	Data Transaksi Pasien	Berisi data transaksi pelanggan yang mendaftar sebagai pasien.
4.	Data Laboran	Berisi data laboran yang digunakan untuk masuk kedalam sistem.
5.	Data Item	Berisi data item bahan, alat dan jasa.

4.4 Kebutuhan fungsional

Yang termasuk dalam kebutuhan fungsional adalah fitur-fitur yang disediakan oleh perangkat lunak yang dapat diakses oleh pengguna secara langsung melalui antar muka pengguna yang disediakan oleh sistem. Dalam sistem yang dikembangkan nantinya fitur-fitur tersebut akan dibagi berdasarkan aktor yang bertanggung jawab menggunakan fitur tersebut, yaitu administrator, laboran dan operator logistik yang setelah ini hanya disebutkan logistik saja. Fitur yang dapat digunakan oleh beberapa aktor juga akan dipisah dari fitur yang hanya dapat digunakan oleh satu aktor.

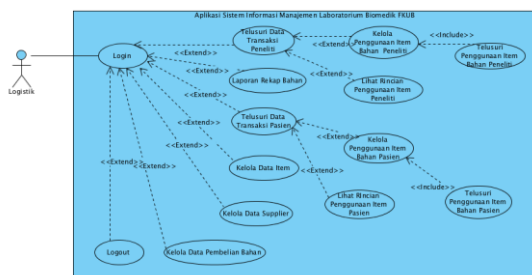
Karena jumlah kebutuhan fungsional yang banyak maka tidak mungkin jika dimasukkan kedalam jurnal ini. Oleh karena itu pada jurnal ini hanya dilampirkan diagram *use case* seperti yang ditunjukkan pada Gambar 5-7.

Untuk aktor administrator terdapat 14 kebutuhan fungsional, laboran mempunyai 2 kebutuhan fungsional dan logistik mempunyai 5 kebutuhan fungsional dan kebutuhan fungsional yang dapat dilakukan oleh beberapa aktor berjumlah 14 kebutuhan fungsional.

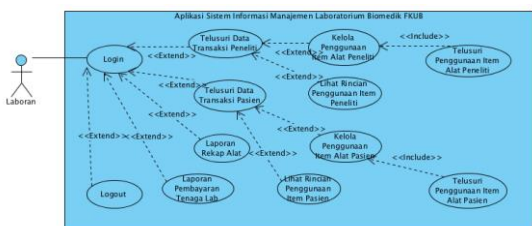


Gambar 5. Diagram use case administrator





**Gambar 6.** Diagram *use case* logistic



**Gambar 7.** Diagram *use case* laboran

Untuk aktor administrator terdapat 14 kebutuhan fungsional, laboran mempunyai 2 kebutuhan fungsional dan logistik mempunyai 5 kebutuhan fungsional dan kebutuhan fungsional yang dapat dilakukan oleh beberapa aktor berjumlah 14 kebutuhan fungsional.

#### 4.5 Kebutuhan Non-fungsional

Yang termasuk dalam kebutuhan non-fungsional adalah kebutuhan-kebutuhan yang terkait dengan kualitas dan batasan sistem. Dalam sistem yang dikembangkan nantinya kebutuhan non-fungsional yang harus dipenuhi ditunjukkan pada Tabel 3 berikut.

**Tabel 3. Kebutuhan Non-Fungsional**

No.	Kebutuhan	Deskripsi
1	Security	Mampu membatasi hak akses pengguna sesuai dengan tanggung jawab masing-masing pengguna.
2	Usability	Aplikasi mudah digunakan oleh pengguna secara umum.
3	reliability	Aplikasi mampu memberikan informasi secara <i>real-time</i> dan tersedia setiap saat.

## 5. IMPLEMENTASI

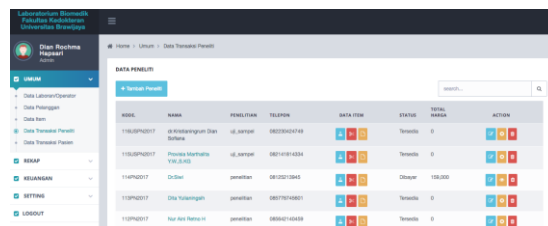
### 5.1 Implementasi algoritma

Sistem informasi manajemen laboratorium biomedik FKUB mempunyai fitur-fitur yang menunjang proses bisnis yang dimiliki oleh laboratorium. Implementasi algoritma

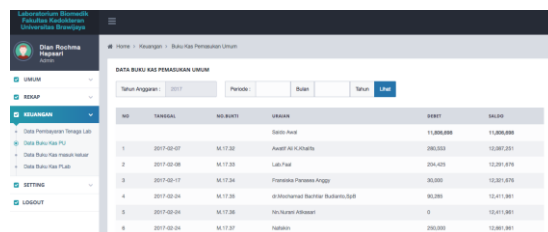
mengimplementasikan fitur-fitur tersebut sebagai *method* dan kelas dalam bentuk barisan kode program.

## 5.2 Implementasi Antarmuka

Implementasi antarmuka memberikan gambaran akhir dari antarmuka aplikasi yang dikembangkan. Berikut adalah contoh implementasi antarmuka dari aplikasi seperti yang ditunjukkan pada Gambar 8-10.



**Gambar 8.** Antarmuka data transaksi peneliti



**Gambar 9.** Antarmuka buku kas pemasukan umum



**Gambar 10.** Antarmuka cetak laporan buku kas pemasukan umum

## 6. PENGUJIAN DAN ANALISIS

Proses pengujian mempunyai 3 tahapan yaitu pengujian unit, pengujian integrasi dan pengujian validasi.

Proses pengujian unit dan integrasi menggunakan teknik *white-box testing* yang diimplementasikan dengan metode basis *path*. Metode basis *path* akan memodelkan fungsi yang akan diuji kedalam sebuah bentuk *flow graph*. Dari *flow graph*, dapat ditentukan nilai dari *cyclomatic complexity*, menentukan jumlah jalur independen berdasarkan nilai *cyclomatic complexity*, kemudian penentuan kasus uji berdasarkan jumlah jalur independen yang telah didapat.

### 6.1 Pengujian Unit

Pada proses pengujian unit, pengujian dilakukan dengan menguji suatu fungsi dasar yang dibuat oleh peneliti didalam sistem.

### 6.2 Pengujian Integrasi

Pada proses pengujian integrasi, pengujian dilakukan dengan menguji suatu fungsi yang lebih kompleks yang dibuat oleh peneliti didalam sistem.

### 6.3 Pengujian Validasi

Pengujian validasi dilakukan untuk mengetahui apakah fungsi-fungsi didalam sistem yang dibangun sudah berjalan dengan baik. Pengujian validasi menggunakan teknik pengujian *black-box*.

### 6.4 Analisis

Proses analisis dilakukan untuk mendapatkan hasil analisis dan kesimpulan dari setiap tahapan pengujian aplikasi sistem informasi manajemen laboratorium biomedik FKUB yang telah dilakukan. Dikarenakan proses analisis mengacu pada tahapan pengujian aplikasi, maka proses analisis mempunyai tahapan yang sama dengan tahapan pengujian yaitu analisis hasil pengujian unit, analisis hasil pengujian integrasi dan analisis hasil pengujian validasi.

Proses analisis hasil pengujian unit dilakukan dengan mencocokkan jumlah kemungkinan *flow* yang terdapat pada *source code* dengan jumlah jalur independen yang ditentukan dari hasil perhitungan *cyclomatic complexity*. Hasil yang diperoleh dari pengujian unit menyatakan bahwa semua unit yang diuji mempunyai jumlah kemungkinan *flow* yang sama dengan jumlah jalur independen.

Unit-unit yang diuji pada pengujian unit sebenarnya bukan unit terkecil pada sistem ini, karena pada *framework* laravel, unit-unit terkecil sudah disediakan oleh *framework* dan *programmer* hanya perlu menggunakannya saja. Unit-unit terkecil inilah yang akan digunakan di setiap unit-unit yang dibuat oleh peneliti.

Proses analisis hasil pengujian integrasi dilakukan dengan mencocokkan jumlah kemungkinan *flow* yang terdapat pada *source code* dengan jumlah jalur independen yang ditentukan dari hasil perhitungan *cyclomatic complexity*. Hasil yang diperoleh dari pengujian integrasi menyatakan bahwa semua integrasi yang diuji mempunyai jumlah kemungkinan *flow* yang sama dengan jumlah jalur independen.

Unit-unit yang diuji pada pengujian integrasi tidak menggunakan unit-unit pada pengujian unit, yang membedakan unit-unit yang diuji pada pengujian unit dengan unit-unit yang diuji pada pengujian integrasi adalah unit-unit yang diuji pada pengujian integrasi mempunyai *scope* yang lebih besar dan lebih kompleks daripada unit-unit yang diuji pada pengujian unit.

Proses analisis hasil pengujian validasi dilakukan dengan menguji beberapa fungsi-fungsi yang terdapat pada kebutuhan fungsional, jika saat diuji fungsi tersebut berjalan dengan baik, maka hasil pengujian validasi dari fungsi tersebut adalah valid.

Berdasarkan pengujian validasi yang telah dilakukan, semua fungsi yang diuji menghasilkan nilai valid, yang artinya semua fungsi yang diuji telah berjalan dengan baik.

## 7. KESIMPULAN

Berdasarkan penelitian rancang bangun aplikasi sistem informasi manajemen laboratorium biomedik FKUB yang telah dilakukan setelah melalui tahapan analisis kebutuhan, perancangan, implementasi dan pengujian. Maka dapat diambil kesimpulan bahwa sistem informasi ini mengadopsi proses bisnis laboratorium biomedik FKUB yang telah diperbaharui. Sistem informasi ini mengadopsi konsep MVC dan dibangun dengan bahasa pemrograman PHP menggunakan *framework* laravel. Sistem informasi ini memiliki 3 aktor, 35 kebutuhan fungsional dan 23 tabel data. Sistem informasi ini memiliki 43 kelas yang terdiri dari kelas *routing*, *middleware*, *view*, *controller* dan *model*.

Sistem ini dirancang menggunakan diagram UML, yaitu diagram *use-case*, *activity*, *entity-relationship*, kelas dan *sequence*. Sistem ini diuji dengan 3 tahapan yang berbeda yaitu pengujian unit, pengujian integrasi dan pengujian validasi. Pengujian unit dan integrasi menggunakan teknik pengujian *white-box* sedangkan pengujian validasi menggunakan teknik pengujian *black-box*.

Dikarenakan sistem informasi ini melibatkan *stakeholder* selain peneliti dan pembimbing peneliti yaitu peneliti dan laboran yang bekerja di laboratorium biomedik FKUB, maka disarankan untuk berkomunikasi dengan *stakeholder* dengan baik dan jelas, misalnya terdapat tambahan kebutuhan fungsional baru

pastikan agar kebutuhan fungsional tersebut tidak keluar dari *scope project* yang telah ditentukan. Sistem informasi ini masih dijalankan pada *server* lokal disekitar laboratorium biomedik saja, namun jika kedepannya sistem informasi ini dikembangkan untuk memenuhi kebutuhan seluruh laboratorium yang terdapat pada FKUB, maka dapat dipertimbangkan untuk menggunakan *server* universitas brawijaya yaitu ub.ac.id.

Conference on e-Business Engineering (ICEBE'06).

Supaartagorn, Chanchai., 2011. *PHP Framework For Database Management Based On MVC Pattern*. Department of Mathematics Statistics and Computer, Faculty of Science, Ubon Ratchathani University, Thailand.

## DAFTAR PUSTAKA

- Alter, Steven., 2008. *Defining Information System as Work Systems: Implications for the IS field*. University of San Fransisco.
- Al-Fedaghi, Sabah., 2011. *Developing Web Applications*. International Journal of Software Engineering and Its Application Vol. 5 No. 2.
- Bizagi., 2014. *BPMN by Example*. [online] Tersedia di: <<http://resources.bizagi.com/docs/BPMNByExampleENG.pdf>> [Diakses 1 Januari 2017].
- Cui, Wei., Huang, Lin., Liang, Lijing., & Li, Jing., 2009. *The Research of PHP Development Framework Based on MVC Pattern*. Fourth International Conference on Computer Sciences and Convergence Information Technology (ICCIT), pp.947-949.
- Ehsan, Max. *Architecture of Laravel Applications*. [online] Tersedia di: <<http://laravelbook.com/laravel-architecture/>> [Diakses 15 November 2016]
- Lawrence, Jeremy., 2015. *Mengenal Framework Laravel*. [online] Tersedia di: <<http://laravel.id/mengenal-framework-laravel>> [Diakses 06 September 2016]
- McLaughlin, Emily., 2014. *Business process outsourcing (BPO)*. [online] Tersedia di: <<http://searchcio.techtarget.com/definition/business-process-outsourcing>> [Diakses 06 September 2016]
- Shan, Tony C., Hua, Winnie W., 2006. *Taxonomy of Java Web Application Frameworks*. IEEE International