

APLIKASI PENGELOLAAN KING MUAYTHAI BANDUNG BERBASIS WEB

Widia Rismadewi¹, Wawa Wikusna², Heru Nugroho³,
123 Program Studi D3 Manajemen Informatika, Fakultas Ilmu Terapan, Universitas Telkom
rismadewi.widia@gmail.com

Abstrak

Muay Thai atau *Tinju Thai* adalah seni bela diri kerajaan *Thai*. *King Muaythai* Bandung adalah sebuah camp seni bela diri *muaythai* camp ini berdiri sejak tahun 2011. *King Muaythai* memiliki 100 *member* aktif, dan sangat sulit untuk mengatur ketika banyak *member* ingin melakukan pendaftaran *member* baru, melakukan pembayaran kuota latihan perbulannya, karena masih menggunakan sistem manual atau kertas dan sering kali kertas tersebut hilang atau rusak akibatnya sering sekali di temukan duplikasi data. Berdasarkan masalah tersebut, dibutuhkan aplikasi pengelolaan data yang sesuai dengan kebutuhan organisasi. Aplikasi tersebut di beri nama "Aplikasi Pengelolaan King Muaythai Bandung Berbasis Web" Aplikasi yang akan dibangun dapat membantu petugas administrasi dalam pengolahan data dan aplikasi tersebut akan memfasilitasi *member* dalam melakukan seluruh kegiatan yang bisa dilakukan secara *online*. Metode pengembangan yang digunakan adalah model *Waterfall*. Secara garis besar model *Waterfall* mempunyai langkah-langkah seperti *requirements definition, system and software design, implementation and unit testing, integration and system testing, operation and maintenance* Harapannya agar *member* dapat mendaftar secara *online*, mengetahui *history* latihan dan melakukan pembayaran. Admin dapat mengelola data *member*, mengelola jadwal latihan, mengelola presensi kehadiran, mengelola pembayaran dan mengelola data trainer. Trainer dapat melihat jadwal dan melihat *history* latihan.

Kata Kunci: King Muaythai, Aplikasi Pengelolaan, Web Based, Waterfall, pengelolaan data

Abstract

Based on the problems found in the camp, an application to manage the data that fits with the need of the organisation. The Application is named as " Web Based Application Management King Muaythai Bandung ". The Application that shall be created may help the administration workers in the data management, and the application shall facilitate members in managing their activities online. The method of the development that is used is Waterfall mode. Essentially, the waterfall model has the steps such as; requirements definition, system and software design, implementation and unit testing, integration and system testing, operation and maintenance. It is expected that members could do the registration online, know the practice history and could do the payment. Admin could manage the data of the members, manage members' attendance, manage the payment and manage the data of the trainers. Trainers could also see the training schedule online.

1. Pendahuluan

1.1 Latar Belakang

Muay Thai atau Tinju Thai adalah seni bela diri kerajaan Thai. Muay Thai adalah olahraga nasional Kerajaan Thai dan turunan dari bela diri kuno Muay Boran, Muay Thai adalah seni bela diri tangan kosong yang berasal dari Thailand. Olahraga ini sedang sangat diminati oleh berbagai kalangan karena sangat banyak manfaatnya diantaranya menjaga diri dari tindak kejahatan, sangat baik untuk menjaga berat badan dan menjaga kesehatan. King Muaythai Bandung yang berdiri sejak 2011 yang berlokasi di Jl. Terusan Buah Batu No.54, Batununggal, Bandung Kidul, Bandung, Jawa Barat 40266 telah memiliki *member* aktif saat ini adalah 100 orang. Berdasarkan hasil wawancara dengan Bambang sebagai pemilik camp, di temukan permasalahan yaitu data *member* tercatat hanya pada form pendaftaran yang dikumpulkan dalam map, sehingga pengelola tidak memiliki data *member* yang terorganisi, backup data, form

pendaftaran sering hilang atau rusak dan untuk update data *member* harus ganti form baru. Pada pencatatan pembayaran *member* masih dilakukan secara manual dengan menggunakan media buku dan *member* menerima kwitansi, masalah muncul ketika buku hilang atau rusak sehingga tidak dapat diketahui *history* pembayaran *member*. Duplikasi dan kehilangan data *member* sering terjadi dikarenakan adanya *member* lama yang ingin mengikuti latihan kembali membuat admin harus mencari data lama *member* yang memakan waktu lebih lama dan sering kali data *member* hilang membuat admin harus membuat data *member* kembali, itu yang menyebabkan terjadinya duplikasi data *member*. Berdasarkan uraian permasalahan tersebut, maka dibutuhkan aplikasi pengelolaan data yang sesuai dengan kebutuhan organisasi. Aplikasi tersebut di beri nama Aplikasi Pengelolaan King Muaythai Bandung Berbasis Web. Harapannya agar *member* dapat mendaftar secara *online*, mengetahui *history* latihan dan melakukan

pembayaran. Admin dapat mengelola data member, mengelola jadwal latihan, mengelola presensi kehadiran, mengelola pembayaran dan mengelola data trainer. Trainer dapat melihat jadwal dan melihat history latihan.

1.2 Batasan Masalah

Adapun batasan masalah dalam pembuatan proyek akhir ini yaitu:

1. Aplikasi tidak mengelola gaji *trainer*.
2. Presensi *member* dan *trainer* dilakukan secara *onsite*.
3. Pengelolaan, pemeliharaan sarana dan prasarana tidak dibahas.

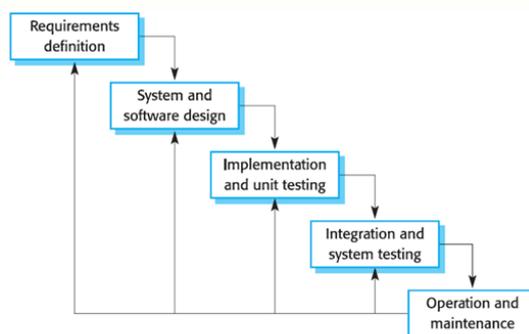
1.5 Definisi Operasional

Aplikasi Pengelolaan King *Muaythai* ini adalah Aplikasi Berbasis Web berfungsi untuk mempermudah calon *member* melakukan pendaftaran secara *online*, aplikasi ini dapat membantu petugas administrasi dalam melakukan konfirmasi pembayaran, mencatat data *member*, daftar hadir *member*, pencatatan pembayaran, membantu *member* melihat jadwal latihan, dan juga dapat mengetahui sisa kuota latihan *member*, sistem informasi ini dapat membantu admin mencatat presensi *member*.

Aplikasi ini dapat digunakan secara umum. Dengan menggunakan aplikasi ini calon *member* dengan mudah mengetahui informasi tentang King *Muaythai* Bandung.

1.6 Metode Pengerjaan

Dalam pengembangan aplikasi perlu digunakan metode *Software Development Life Cycle* (SDLC) sebagai pedoman bagaimana dan apa yang harus dilakukan selama melaksanakan pengembangan sistem. Adapun metode pengembangan yang digunakan adalah model *Waterfall*. Secara garis besar model *Waterfall* mempunyai langkah-langkah seperti *requirements definition*, *system and software design*, *implementation and unit testing*, *integration and system testing*, *operation and maintenance*.



Gambar 1. 1
Model *Waterfall* [2]

1. Requirements definition

Analisis kebutuhan yang akan dilakukan yaitu dengan mengumpulkan data dan informasi serta melakukan wawancara terhadap pemilik camp, *trainer*, petugas administrasi serta *member* di King *Muaythai* Bandung untuk mengetahui kebutuhan sistem yang akan dibangun.

Tahapan ini akan menghasilkan dokumen *user requirement* atau dapat dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Hasil dari tahapan ini akan dijadikan dasar untuk ke tahapan selanjutnya.

2. System and software design

Tahap ini adalah tahap mendesain sistem baru agar dapat digunakan oleh petugas administrasi, *member*, *trainer* dan pemilik camp sesuai dengan kebutuhan dan diharapkan dapat mengatasi masalah-masalah yang ada dengan cara perancangan menggunakan *flowmap*, *use case diagram*, *class diagram*, *sequence diagram*, *entity relationship diagram* dan *user acceptance testing*

3. Implementation and unit testing

Tahapan pengkodean merupakan implementasi dari analisis kebutuhan dan desain sesuai kebutuhan *user*. Dalam penerapannya akan menggunakan *CodeIgneter*.

4. Integration and system testing

Setelah program selesai dibuat tahapan terakhir adalah dengan dilakukan *testing* terhadap program tersebut. Pengujian akan menggunakan *black box testing* dan *user acceptance testing* untuk mengetahui sebuah kesalahan yang mungkin terjadi dalam sistem.

5. Operation and maintenance

Tahapan ini tidak dilakukan.

2. Tinjau Pustaka

Dalam pembahasan pada bab ini akan menjelaskan mengenai sekolah King *Muaythai* Bandung sebagai objek dari aplikasi ini dengan mengambil tinjauan pustaka berdasarkan hasil kajian yang diambil dari sumber - sumber atau literasi dari kajian pustaka yang diperoleh terkait pembahasan pada materi ini.

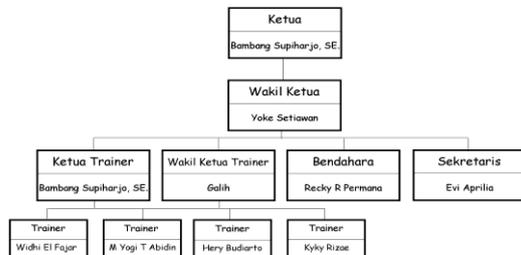
2.1 King *Muaythai* Bandung

King *Muaythai* Bandung adalah salah satu sasana *muaythai* yang berlokasi di daerah sangat strategis yaitu terletak di keramaian kota Bandung di jl. Terusan Buah Batu No.54 Bandung yang berdiri di tahun 2011. King *Muaythai* Bandung di pimpin oleh *head coach* bernama Bambang Supriharjo SE yang sudah memiliki sertifikat resmi dari Asosiasi *Muaythai* Thailand, dan telah

mengikuti pelatihan bersertifikat nasional maupun internasional.

2.2 Struktur Organisasi

Berdasarkan wawancara yang dilakukan kepada pemilik, *King Muaythai* Bandung memiliki badan Struktur Organisasi. Di bawah ini merupakan badan Struktur Organisasi *King Muaythai* Bandung :



Struktur Organisasi *King Muaythai* Bandung

2.2.1 Kelas Latihan dan Biaya

Reguler Class adalah pengajaran yang memiliki tingkat kelas yaitu *Basic, Advance, Professional*. *Basic class* merupakan pengajaran *muaythai* pada tahap dasar, pengajaran akan di lakukan dengan metode permainan dan simulasi yang membuat latihan jadi menyenangkan untuk biaya *regular basic class* 4 kali pertemuan Rp 200.000 untuk 8 kali pertemuan Rp 350.000 untuk 12 kali pertemuan Rp 550.000 . *Advance class* yaitu untuk tingkat kelas mahir *muaythai* dengan kemampuan gerakan yang sudah stabil untuk biaya *regular advance class* 4 kali pertemuan Rp 250.000 untuk 8 kali pertemuan Rp 400.000 untuk 12 kali pertemuan Rp 600.000 .

2.2.2 Tata Usaha

Ditinjau dari sudut asal usul kata (etimologis), maka ADMINISTRASI berasal dari Bahasa Latin yaitu Ad+Ministrare. Ad berarti intensif, sedangkan Ministrare berarti melayani, membantu, dan memenuhi atau menyediakan, tenaga tata usaha memiliki tiga peranan pokok yaitu [3]:

1. Melayani pelaksanaan pekerjaan-pekerjaan operatif untuk mencapai tujuan dari suatu organisasi
2. Menyediakan keterangan-keterangan bagi pucuk pimpinan organisasi itu untuk membuat keputusan atau melakukan tindakan yang tepat
3. Membantu kelancaran perkembangan organisasi sebagai suatu keseluruhan.

Berdasarkan pendapat The Lian Gie di atas, maka peranan tenaga administrasi sekolah sesungguhnya hanya satu yaitu sebagai administrator karena ketiga peranan yang diungkapkan di atas yaitu melayani,

menyediakan dan membatu sama dengan administrasi.

2.2.3 Pendaftaran dan Pembayaran

Tata cara pendaftaran di *King Muaythai* Bandung adalah *member* datang ke camp mengisi formulir pendaftaran dan data pribadi setelah itu memilih kelas yang di minati dan membayar sesuai dengan yang sudah di tentukan, jika sudah melakukan pembayaran petugas administrasi akan memberikan kwitansi sebagai bukti sah pembayaran.

2.2 Aplikasi

Aplikasi adalah penerapan dari rancangan *sistem* untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna [4].

Aplikasi adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pertanyaan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output* [5].

2.3 Web

Web adalah ruang informasi di dalam ruang internet yang menggunakan teknologi *hypertext*. Informasi dapat berupa teks, gambar, suara, animasi, dan data multimedia lainnya. Situs atau *web* dapat dikategorikan menjadi dua, yaitu *web* statis dan *web* dinamis [6].

Web Statis adalah *web* yang berisi atau menampilkan informasi-informasi yang sifatnya statis atau tetap. Disebut statis karena pengguna tidak dapat berinteraksi dengan *web* tersebut. *Web* dinamis merupakan *web* yang menampilkan informasi serta dapat berinteraksi dengan pengguna. *Web* dinamis memungkinkan pengguna untuk berinteraksi menggunakan *form* sehingga dapat mengolah informasi yang ditampilkan[6].

2.4 PHP Hypertext Preprocessor

PHP (*PHP Hypertext Preprocessor*) adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Karena PHP merupakan *server-side scripting* maka dieksekusi di *server* kemudian hasilnya dikirimkan ke *browser* dalam format HTML, sehingga keamanan halaman *web* lebih terjamin [10].

PHP sangat mudah untuk digunakan karena memiliki beberapa keunggulan, yaitu:

1. *Open source* dan gratis. Teknologi semakin lama semakin berkembang dan semakin canggih sehingga lebih terjangkau, murah, mudah dan aman.

2. *Life Cycle* karena PHP selalu *up to date* mengikuti perkembangan teknologi.
3. *Cross Platform* yaitu dapat dipakai di semua *webserver* dan dijalankan di sistem operasi (Linux, Windows, FreeBSD).
4. Mendukung ke banyak *database* baik gratis maupun komersil.

Kemudahan dalam mendapatkan dokumentasi di internet dan banyaknya aplikasi berbasis *web* yang dibuat dengan menggunakan PHP [11].

2.5 CodeIgniter

CodeIgniter adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi berbasis PHP dibandingkan jika menulis semua kode program dari awal. *Framework* merupakan kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat kodenya dari awal [10].

2.6 Web Server

Web server adalah sebuah software yang memberikan layanan berbasis data dan berfungsi menerima permintaan dari HTTP atau HTTPS pada klien yang dikenal dan biasanya kita kenal dengan nama web browser (Mozilla Firefox, Google Chrome) dan untuk mengirimkan kembali yang hasilnya dalam bentuk beberapa halaman web dan pada umumnya akan berbentuk dokumen HTML.

Apache (*Apache Web Server – The HTTP Web Server*) merupakan web server yang paling banyak dipergunakan di Internet. Program ini pertama kali didesain untuk sistem operasi lingkungan UNIX. Apache mempunyai program pendukung yang cukup banyak. Hal ini memberikan layanan yang cukup lengkap bagi penggunaanya.

2.7 MySQL

Pengaksesan data dalam *database* dapat dilakukan dengan mudah melalui *Structured Query Language (SQL)*. Data dalam *database* bisa diakses melalui aplikasi non-*web* (misalnya *Visual Basic*) maupun aplikasi *web* (misalnya dengan PHP). MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*database management system*), *database* ini *multithread*, *multi-user*. MySQL adalah *relational database management system (RDBMS)* yang didistribusikan secara gratis di bawah lisensi *general public license (GPL)* yang berarti aplikasi ini bebas untuk menggunakannya [7].

2.8 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk permodelan basis data relasional [6].

2.9 FlowMap

Flow map merupakan metode untuk menggambarkan tahap-tahap penyelesaian masalah (prosedur) beserta aliran data dengan simbol-simbol standar yang mudah dipahami. Tujuan utama penggunaan *flow map* adalah untuk menyederhanakan rangkaian proses atau prosedur untuk memudahkan pemahaman pengguna terhadap informasi tersebut [8].

2.10 Use Case Diagram

Use case adalah diagram yang bersifat statis dan menunjukkan himpunan *use case* dan aktor (suatu jenis khusus dari kelas). *Use case* menggambarkan kebutuhan sistem dari sudut pandang *user*. Diagram *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut [9].

2.11 Class Diagram

CodeIgniter adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis PHP dibandingkan jika menulis semua kode program dari awal. *Framework* merupakan kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat kodenya dari awal [9].

Model ini bertujuan untuk memisahkan proses bisnis dari pertimbangan antarmuka *user* agar para pengembang bisa lebih mudah mengubah setiap bagian tanpa memengaruhi yang lain. Dalam MVC, model menggambarkan informasi (data) dan proses bisnis, *view* (tampilan) berisi elemen antarmuka *user* seperti teks, formulir masukan, sementara *controller* mengatur komunikasi antarmodel dan *view*. Penjelasan jenis-jenis komponen MVC yaitu :

Model, merupakan kelas yang mendasari logika proses dalam aplikasi perangkat lunak dan kelas yang terkait dengannya. Model adalah suatu objek yang tidak mengandung informasi tentang *user interface*. Model juga suatu kelas yang berisi metode/fungsi yang merupakan kumpulan dan proses-proses. Model digunakan untuk menyimpan data dan aturan bisnis yang relevan.

View, Merupakan dari kelas yang mewakili unsur-unsur dalam antarmuka *user*. Dalam *view* terdapat nama yang dipakai untuk mengidentifikasi *file* skrip tampilan saat dipanggil lewat fungsi *render*. Beberapa bagian *view* seperti; *Layout*, *widget*, dan *view* sistem.

Controller, merupakan kelas yang menghubungkan model dan *view*, dan digunakan untuk berkomunikasi antara kelas dalam model dan *view*. *Controller* memiliki aksi yang standar, dan memiliki beberapa bagian seperti; *Route*, *instan controller*, *action*, dan *action parameter binding*.

2.11 Black Box Testing

Black-box testing adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi / struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Uji kasus dibangun di sekitar spesifikasi dan persyaratan, yakni, aplikasi apa yang seharusnya dilakukan. Menggunakan deskripsi eksternal perangkat lunak, termasuk spesifikasi, persyaratan, dan desain untuk menurunkan uji kasus. Tes ini dapat menjadi fungsional atau non-fungsional, meskipun biasanya fungsional. Perancang uji memilih input yang valid dan tidak valid dan menentukan output yang benar. Tidak ada pengetahuan tentang struktur internal benda uji itu. Metode uji dapat diterapkan pada semua tingkat pengujian perangkat lunak yaitu unit, integrasi, fungsional, sistem dan penerimaan. Ini biasanya terdiri dari kebanyakan jika tidak semua pengujian pada tingkat yang lebih tinggi, tetapi juga bisa mendominasi unit testing juga. Metode ujicoba blackbox memfokuskan pada keperluan fungsional dari software. Karna itu ujicoba blackbox memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program. Ujicoba blackbox bukan merupakan alternatif dari ujicoba whitebox, tetapi merupakan pendekatan yang melengkapi untuk menemukan kesalahan lainnya, selain menggunakan metode whitebox. Ujicoba blackbox berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya[10]:

1. Fungsi-fungsi yang salah atau hilang.
2. Kesalahan interface.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan performa.
5. Kesalahan inisialisasi dan terminasi.

2.12 User Acceptance Test

User Acceptance Test (UAT) atau Uji Penerimaan Pengguna adalah suatu proses pengujian oleh pengguna yang dimaksudkan untuk menghasilkan dokumen yang dijadikan bukti bahwa software yang telah dikembangkan telah dapat diterima oleh pengguna, apabila hasil pengujian (testing) sudah bisa dianggap memenuhi kebutuhan dari pengguna. Proses UAT didasarkan pada dokumen requirement yang disepakati bersama. Dokumen requirement adalah dokumen yang berisi lingkup pekerjaan software yang harus dikembangkan, dengan demikian maka dokumen ini semestinya menjadi acuan untuk pengujian[11].

3. Analisis dan Perancangan

Proses yang terjadi pada sistem berjalan di *King Muaythai* Bandung meliputi proses pendaftaran, proses pembayaran iuran, presensi kehadiran *member*, presensi kehadiran *trainer* yang masih di lakukan secara *manual* dengan menggunakan media buku.

King Muaythai Bandung merupakan *camp* yang bergerak di bidang olah raga seni bela diri *muaythai*. Proses bisnis yang berjalan adalah proses pendaftaran *member* baru, pembuatan *form* presensi *member*, proses pendaftaran ulang *member*, proses pencatatan *presensi member* dan *trainer*, pembuatan jadwal *trainer*, pemeriksaan atau pemberitahuan kuota latihan *member*, pembuatan rekapitulasi honor *trainer*. Para calon *member* mendaftarkan secara *onsite* datang ke *camp*.

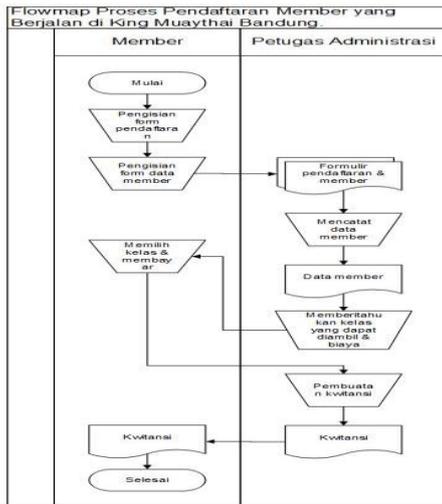
3.1 Proses Pendaftaran Member Baru

Sistem yang berjalan saat ini untuk penilaian resiko pengembangan perangkat lunak memiliki dua *user* yaitu koordinator IT dan responden. Koordinator IT berperan untuk mengelola sistem. Untuk pengisian kuisisioner dan perhitungan hasil kuisisioner masih manual dengan menggunakan *microsoft excel* sehingga masih kurang efektif karena tidak ada tempat penyimpanan data untuk hasil perhitungan seperti database. Perhitungan dimulai dari perhitungan *risk factor*, *risk element*, *development phases*, dan *software risk management activities*. Setelah semua perhitungan selesai, akan menghasilkan *total product risk*. Setelah di dapatkan semua nilai dari setiap faktor, maka hasil akhirnya adalah sebuah laporan yang berisi hasil perhitungan.

Gambar 3-1 Flowmap berjalan

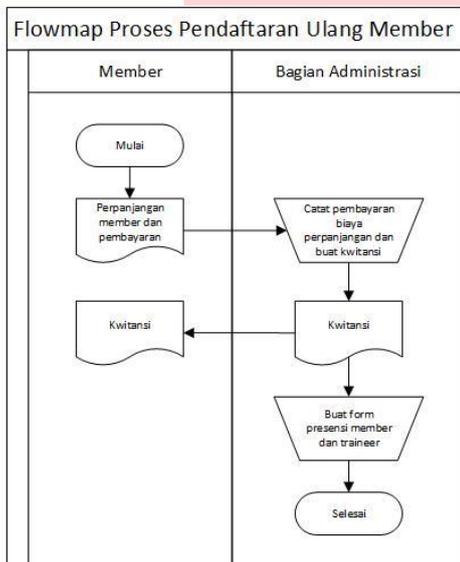
3.2 Analisis Kebutuhan Sistem(atau Produk)

Gambar 3-1 menunjukkan bahwa proses *member* baru dilaksanakan meliputi *form* pendaftaran *member* lalu bagian administrasi akan menentukan kelas, jadwal dan harga yang harus di bayar *member*, lalu jika sudah melakukan pembayaran akan di berikan kwitansi pembayaran, dan akan di buat *form* presensi *member*.



Gambar 3 - 1 Flowmap Proses Berjalan Sistem Pendaftaran Member

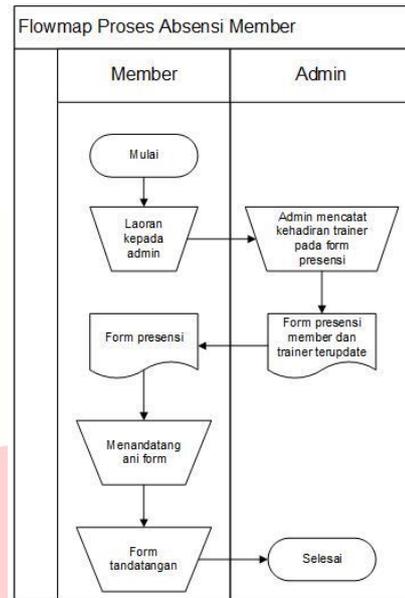
3.3 Proses Pendaftaran Ulang Member



Gambar 3 - 2 Flowmap Proses Pendaftar Ulang Member

Gambar 3-2 menunjukkan bahwa proses pendafran ulang *member* di lakukan karena kuota latihan *member* sudah jatuh tempo dengan melanjutkan pembayaran agar kuota latihan *member* bertambah, lalu jika sudah melakukan pembayaran bagian administrasi akan membuat *form* presensi *member* yang baru.

3.4 Pencatatan Presensi Member



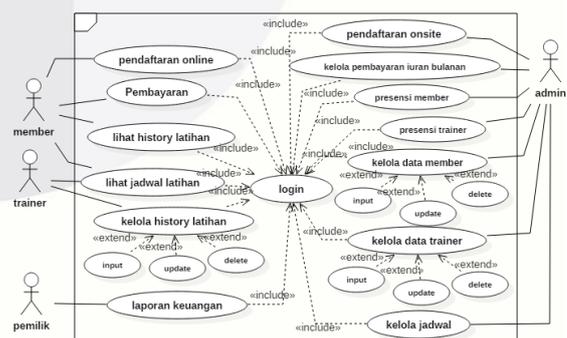
Gambar 3 - 3 Flowmap Proses Absensi Member

Gambar 3-3 menunjukkan bahwa proses pencatatan presensi *member* dilakukan dengan cara melapor kepada *admin*, *admin* mencatat kehadiran.

3.2.3. Use Case Diagram

Proses bisnis yang diusulkan dalam Sistem Informasi King Muaythai Bandung Berbasis Web ini memiliki empat pengguna (*user/actor*) yang berperan sesuai hak akses nya masing-masing, yaitu *admin*, pemilik *muaythai* bandung, *member* dan *Trainer*.

Berikut gambaran proses bisnis usulan dalam bentuk *use case* :

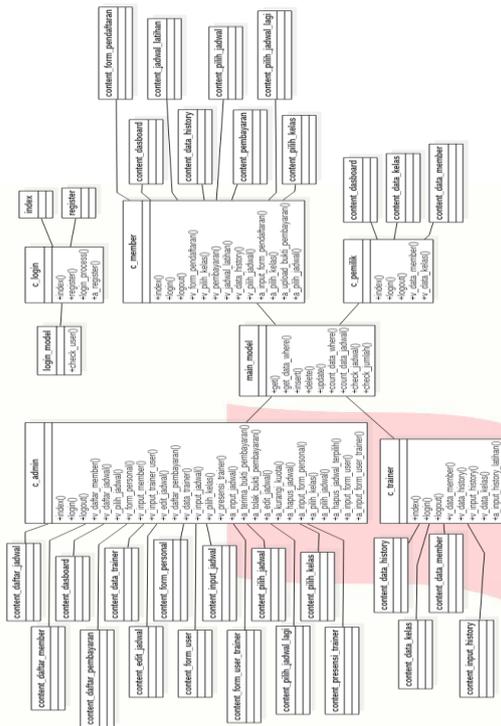


Gambar 3-3 Use Case

3.2.4 Diagram Kelas

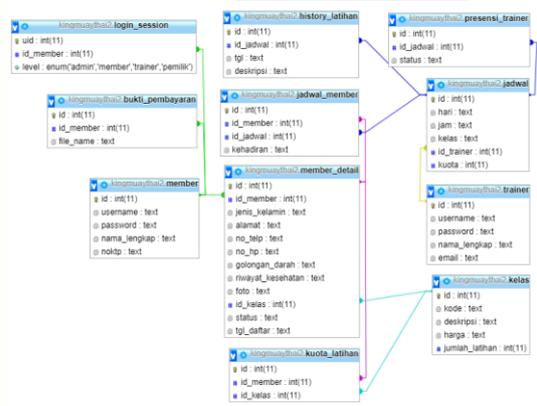
Class diagram mendeskripsikan jenis-jenis objek dalam sebuah sistem dan hubungan antar kelas dalam sebuah sistem yang sedang dibangun. Berikut

merupakan gambar kelas diagram dari aplikasi



Gambar 3-6 Diagram Kelas

3.2.7 Skema Relasi



Gambar 3-7 Skema Relasi

3.7 Analisis Kebutuhan Perangkat Keras dan Perangkat Lunak

3.7.1. Kebutuhan Perangkat Keras

Adapun kebutuhan perangkat lunak untuk membuat aplikasi penilaian resiko pembangunan perangkat lunak di perguruan tinggi berdasarkan *software engineering risk management* adalah sebagai berikut.

Tabel 3-1 Perangkat Keras

No	Jenis Hardware	Spesifikasi yang digunakan
----	----------------	----------------------------

1.	RAM	4 GB
2.	Harddisk	1 TERA
3.	Processor	Intel(R) Core™3 Duo Kecepatan 3.0 GHZ
4.	Screen Resolution	480 854 pixels

3.7.2 Perangkat Lunak

Adapun kebutuhan perangkat lunak untuk membuat aplikasi penilaian resiko pembangunan perangkat lunak di perguruan tinggi berdasarkan *software engineering risk management* adalah sebagai berikut. Tabel 3-2 Perangkat Lunak

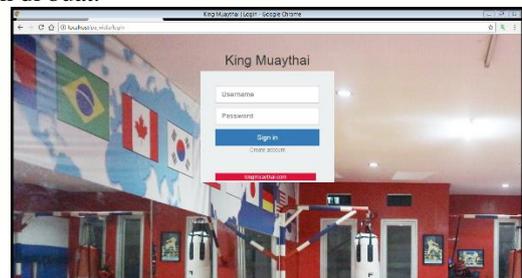
No	Jenis software	Spesifikasi yang digunakan
1.	Operating sistem	Windows 10
2.	Database	MySQL
3.	Script editor	Seblime Text 3
4.	Web server	Apache, MySQL, PHP, and Xampp 3.2.1 CodeIgniter 2.2.2
5.	Web browser	Mozilla Firefox, Google Chrome
7.	Dokumentasi	Ms. Word 2016, Ms. Visio 2016, StarUML

4. Implementasi Perangkat Lunak

Implementasi antarmuka aplikasi merupakan tahap dari perancangan sistem yang telah dibahas di BAB sebelumnya.

4.1 Perancangan Antar Muka

Pada tahapan ini akan dijelaskan mengenai perancangan tampilan pada pembuatan aplikasi, berikut merupakan *mockup* dari tampilan yang akan di buat.



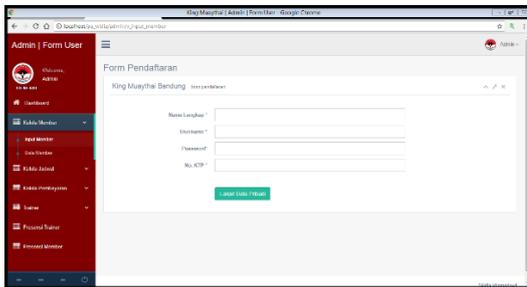
Gambar 4 - 1 Tampilan Login

Tampilan *login* admin merupakan tampilan awal saat aplikasi dibuka. Didalam tampilan login, admin harus mengisi *username* dan *password*. Jika

username dan password salah maka akan ada pesan *invalid username or password*.

tagihan, status, bukti pembayaran dan dapat melakukan edit dan menghapus data.

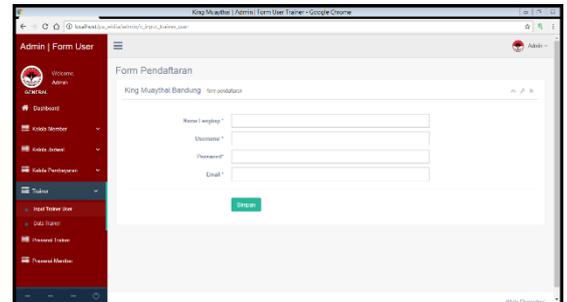
1. Tampilan Kelola Member



Gambar 4 - 2 Tampilan Kelola Member

Tampilan untuk kelola *member* berfungsi untuk pendaftaran *onsite member* yang berisi nama lengkap, *username*, *password*, noktp. Jika sudah di isi klik *button* lanjut pilih kelas

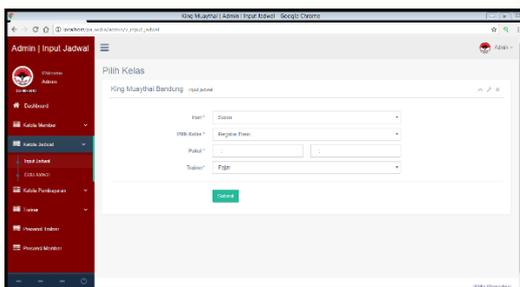
5. Tampilan *Form* Pendaftaran *Trainer*



Gambar 4 - 5 Tampilan form pendaftaran *trainer*

Tampilan untuk kelola *member* berfungsi untuk pendaftaran *onsite member* yang berisi nama lengkap, *username*, *password*, gmail. Jika sudah di isi klik *button* lanjut pilih kelas

2. Tampilan Kelola Jadwal



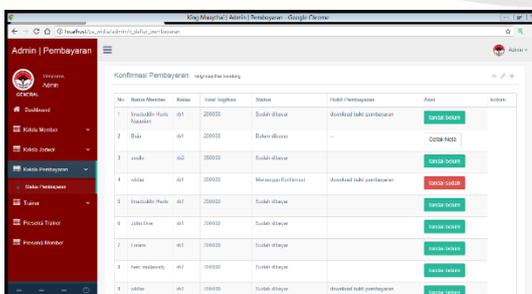
Gambar 4 - 3 Tampilan Kelola Jadwal

Tampilan kelola *member* meliputi *input* jadwal dan data jadwal. Pada tampilan *input* jadwal terdapat hari, pilih kelas, waktu, *trainer* lalu klik *button* submit. Pada tampilan data jadwal terdapat kode kelas, hari, waktu, *edit* dan hapus

Implementasi Antarmuka Bagian *Member*

Proses implementasi antarmuka pada *user member* dibagi menjadi lima bagian yaitu *login*, pendaftaran, pembayaran, melihat jadwal latihan, dan melihat *history* latihan. Berikut penjelasan implementasi antarmuka pada *user member*.

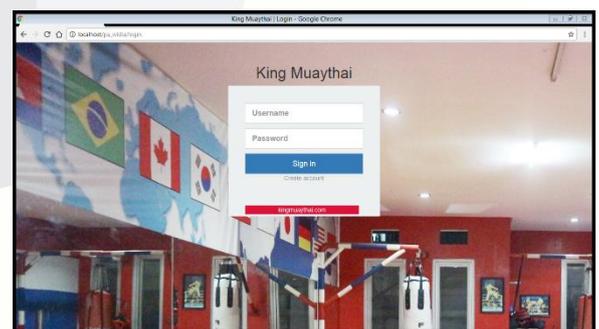
3. Tampilan Kelola Pembayaran



Gambar 4 - 4 Tampilan Kelola Pembayaran

Tampilan kelola pembayaran terdapat status pembayaran meliputi nama member, kelas, total

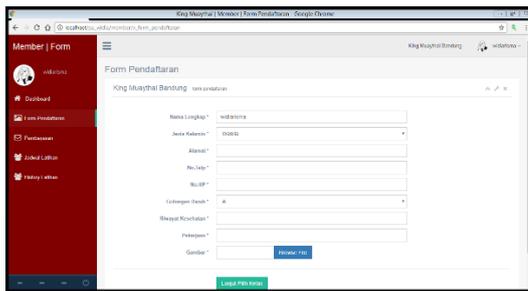
1. Tampilan *Login*



Gambar 4 - 6 Tampilan *Login*

Tampilan *login* member merupakan tampilan awal saat aplikasi dibuka. Didalam tampilan *login*, member harus mengisikan *username* dan *password*. Jika belum memiliki account maka klik *create account* dan mengisikan nama lengkap, *username*, *password* dan no ktp lalu klik *button* register.

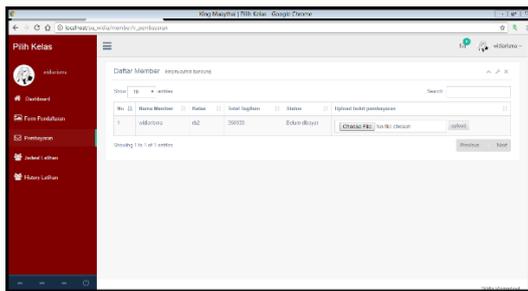
2. Tampilan *Form* Pendaftaran



Gambar 4 - 7 Tampilan *Form* Pendaftaran

Tampilan *form* pendaftaran berisi nama lengkap, jenis kelamin, alamat, no telepon, no hp, golongan darah, riwayat kesehatan, pekerjaan dan foto lalu klik *button* lanjut pilih kelas. Di dalam tampilan pilih kelas terdapat beberapa kelas yang dapat di pilih lalu setelah selesai maka klik *button* lanjut pembayaran.

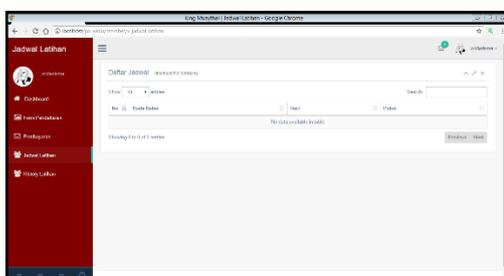
3. Tampilan Pembayaran



Gambar 4 - 8 Pembayaran

Tampilan pembayaran berisi nama *member*, kelas, total tagihan, status, *upload* bukti pembayaran. Pada menu *upload* bukti pembayaran data yang akan di *upload* berupa foto.

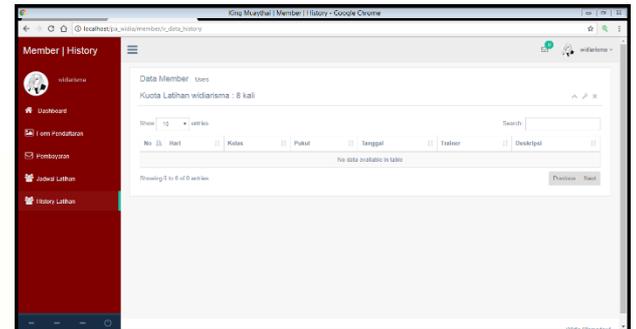
4. Tampilan Melihat Jadwal Latihan



Gambar 4 - 9 Melihat Jadwal Latihan

Tampilan melihat jadwal latihan berisi kode kelas, hari dan waktu.

5. Tampilan Melihat *History* Latihan



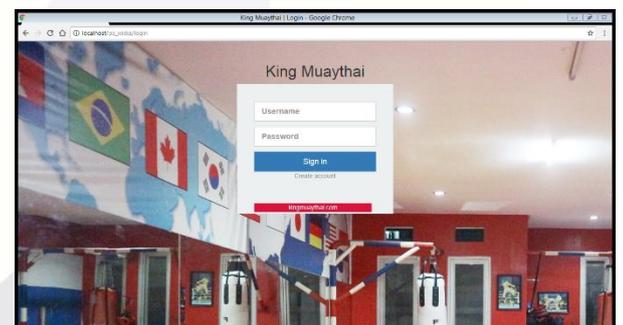
Gambar 4 - 10 Tampilan *History* Latihan

Tampilan melihat *history* latihan berisi tanggal, waktu mulai, waktu selesai, tanda tangan *member*, nama *trainer* dan keterangan latihan .

4.2 Implementasi Antarmuka Bagian Trainer

Proses implementasi antarmuka pada *user trainer* dibagi menjadi tiga bagian yaitu *login*, *view* data member dan *view* data kelas. Berikut penjelasan implementasi antarmuka pada *user trainer*.

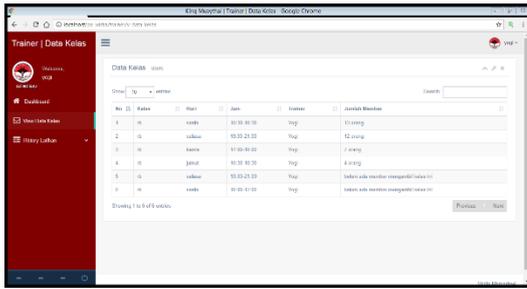
1. Tampilan *Login*



Gambar 4 - 11 Tampilan *Login*

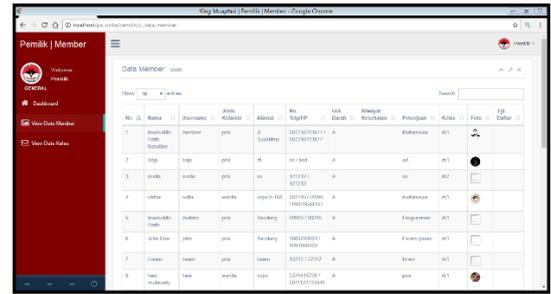
Tampilan *login trainer* merupakan tampilan awal saat aplikasi dibuka. Didalam tampilan *login*, trainer harus mengisi *username* dan *password*. Lalu klik *button* sign in.

2. Tampilan *View* Data kelas



Gambar 4 - 12 Tampilan View Data Kelas

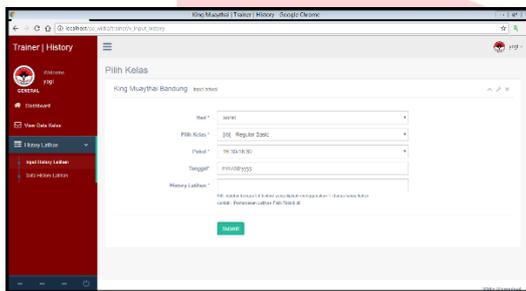
Tampilan view data member berisi kelas, hari, trainer, jumlah member.



Gambar 4 - 14 Tampilan View Data Member

Tampilan view data member berisi nama, user name, jenis kelamin, alamat, no telepon, golongan darah, riwayat kesehatan, pekerjaan, kelas dan foto.

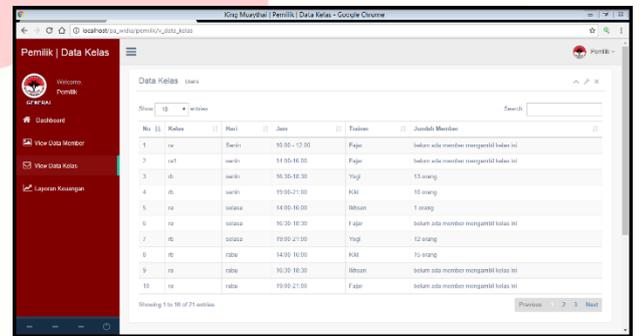
3. History Latihan



Gambar 4 - 12 Tampilan History latihan

Tampilan history latihan berisi hari, pilih kelas, pukul, tanggal, history latihan.

3. Tampilan View Data Kelas



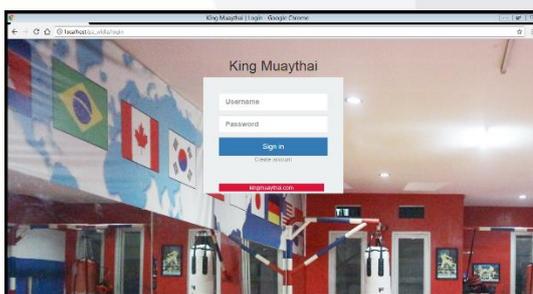
Gambar 4 - 15 Tampilan View Data Kelas

Tampilan view data kelas berisi hari, kelas, jam, trainer, jumlah member.

4.3 Implementasi Antarmuka Bagian Pemilik

Proses implementasi antarmuka pada user pemilik dibagi menjadi tiga bagian yaitu login, view data member dan view data kelas. Berikut penjelasan implementasi antarmuka pada user pemilik.

1. Tampilan Login

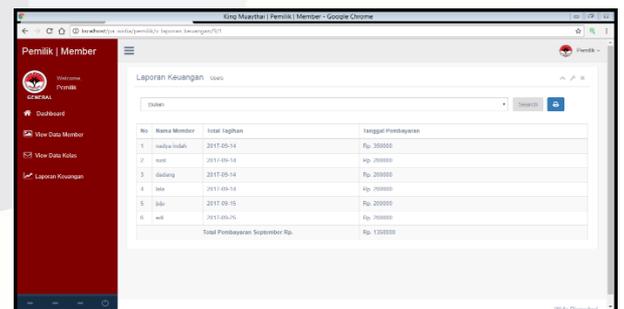


Gambar 4 - 13 Tampilan Login

Tampilan login trainer merupakan tampilan awal saat aplikasi dibuka. Didalam tampilan login, trainer harus mengisi username dan password. Lalu klik button sign in.

2. Tampilan View Data Member

4. Tampilan Laporan Keuangan



Gambar 4 - 16 Tampilan Laporan Keuangan

Tampilan laporan keuangan berisi bulan, nama member, total tagihan, tanggal pembayaran.

5. Penutup

BAB 5 PENUTUP

5.1 Kesimpulan

Dari hasil pengujian data sebenarnya pada uji Aplikasi Pengelolaan *King Muaythai*, dapat disimpulkan bahwa:

1. Aplikasi ini memfasilitasi *member* dalam melakukan pendaftaran secara *online*
2. Aplikasi ini membantu petugas administrasi dalam pengelolaan data agar tidak terjadi duplikasi data *member* atau kehilangan data *member*
3. Aplikasi ini memfasilitasi penjadwalan latihan
4. Aplikasi ini membangun aplikasi yang dapat membantu petugas administrasi dalam mencatat pembayaran iuran bulanan

5.2 Saran

Untuk meningkatkan kinerja dan kelengkapan fungsi dari Aplikasi Pengelolaan *King Muaythai* ini, maka disarankan untuk menambah fungsionalitas untuk sistem penggajian *trainer*.

Daftar Pustaka:

- [1] A.S Rosa, Salahuddin M, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika ,2014.
- [2] Ar Rafi.(2016, Maret 13)Sejarah Singkat [online].Available :<http://www.arrafibandung.com/?pancadewa=page&do=read&id=29>.
- [3] Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 62 Tahun 2004.
- [4] Indonesia D.P.d.K.R, *Kamus Besar Bahasa Indonesia*.Jakarta: Pustaka Amani, 1998.
- [5] Hartono J, *Pengenalan Komputer*. Yogyakarta: ANDI, 1999.
- [6] Kadir, Abdul, *Dasar Pemrograman Web Dengan Menggunakan PHP dan MySQL*. Yogyakarta: Andi, 2008.
- [7] A.S Rosa, Modul Pembelajaran Rekayasa Perangkat Lunak. Bandung: Modula, 2011.
- [8] Komputer Wahana, *Menguasai Pemograman Web Dengan PHP 5*. Semarang: Andi Publisher, 2008.
- [9] Basuki AP, *Membangun Web Berbasis PHP dengan Framework Condeigniter*. Yogyakarta: Lokomedia, 2010.
- [10] M.Rudyanto Arief, *Pemrograman Web Dinamis Menggunakan PHP & MySQL*. Yogyakarta: Penerbit ANDI, 2011.
- [11] Faisal, S.Si, *Aplikasi Berbasis Web dengan PHP dan MySQL*, Yogyakarta: Ram Media, 2011.
- [12] Bunafit Nugroho, *Aplikasi Sistem Pakar dengan PHP & Editor Dreamweaver*. Yogyakarta: Gava Media, 2014.