

**Perancangan Sistem Informasi Manajemen Ruang dan Aset di  
LPMAI UKSW berbasis *Website***

**Artikel Ilmiah**

**Diajukan kepada  
Fakultas Teknologi Informasi  
Untuk memperoleh Gelar Sarjana Komputer**



**Oleh :**

**Tutus Widiyaning Tyas (672015258)**

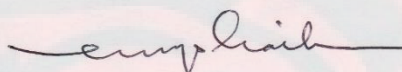
**Evangs Mailoa, S.Kom., M.Cs.**

**Program Studi Teknik Informatika  
Fakultas Teknologi Informasi  
Universitas Kristen Satya Wacana  
Salatiga  
2019**

## Lembar Pengesahan

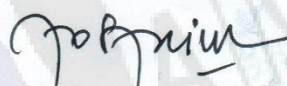
Judul Tugas Akhir : Perancangan Sistem Informasi Manajemen Ruang dan Aset di  
LPMAl UKSW berbasis Website  
Nama Mahasiswa : TUTUS WIDIYANING TYAS  
NIM : 672015258  
Program Studi : Teknik Informatika  
Fakultas : Teknologi Informasi

Menyetujui,

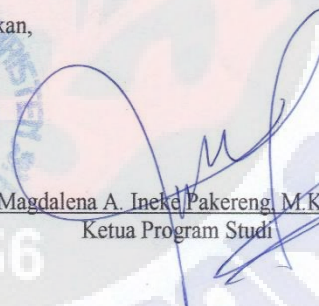


Evangs Mailoa, S.Kom., M.Cs.  
Pembimbing

Mengesahkan,



Dr. Wiwin Sulisty, ST., M.Kom.  
Dekan



Magdalena A. Inek, Pakereng, M.Kom.  
Ketua Program Studi

Dinyatakan Lulus Tanggal: 3 MEI 2019

Reviewer :

- Prof. Dr. Ir. Eko Sedyono, M.Kom.



**Perancangan Sistem Informasi Manajemen Ruang dan Aset di LPMAl  
UKSW berbasis Website**

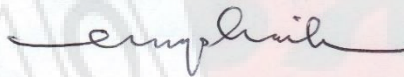
Oleh,

**TUTUS WIDIYANING TYAS**  
**672015258**

**ARTIKEL ILMIAH**

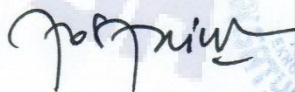
Diajukan Kepada Program Studi Teknik Informatika Guna Memenuhi Sebagian Dari  
Persyaratan Untuk Mencapai Gelar Sarjana Komputer

Disetujui oleh,

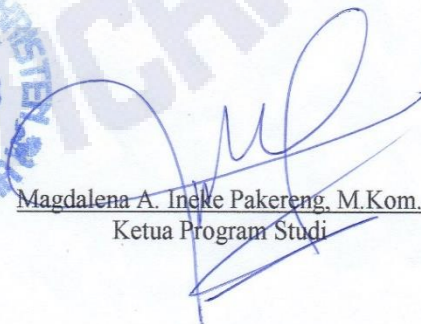


Evangs Mailoa, S.Kom., M.Cs.  
Pembimbing

Diketahui oleh,



Dr. Wiwin Sulisty, ST., M.Kom.  
Dekan



Magdalena A. Ineke Pakereng, M.Kom.  
Ketua Program Studi

**FAKULTAS TEKNOLOGI INFORMASI**  
**UNIVERSITAS KRISTEN SATYA WACANA**  
**SALATIGA**  
**2019**

## Pernyataan

Artikel Ilmiah berikut ini :

Judul : Perancangan Sistem Informasi Manajemen Ruang dan Aset di LPMAl UKSW berbasis *website*  
Pembimbing : Evangs Mailoa, S.Kom., M.Cs.

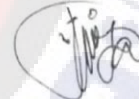
adalah benar hasil karya saya :

Nama : Tutus Widiyaning Tyas  
NIM : 672015258

Saya menyatakan tidak mengambil sebagian atau seluruhnya dari hasil karya orang lain kecuali sebagaimana yang tertulis pada daftar pustaka.

Pernyataan ini dibuat dengan sebenar-benarnya sesuai dengan ketentuan yang berlaku dalam penulisan karya ilmiah.

Salatiga, April 2019



Tutus Widiyaning Tyas



### PERNYATAAN PERSETUJUAN AKSES

Saya yang bertanda tangan di bawah ini:

Nama : TUTUS WIDIYAHINE TYAS  
NIM : 622015258 Email : tutus.widiyahine.tyas@gmail.com  
Fakultas : FTI Program Studi : TI  
Judul tugas akhir : PERANCANGAN SISTEM INFORMASI MANAJEMEN  
RUANG DAN ASET DI LAMPAI UKSW BERBASIS  
WEBSITE

Dengan ini saya menyerahkan hak *non-eksklusif*\* kepada Perpustakaan Universitas – Universitas Kristen Satya Wacana untuk menyimpan, mengatur akses serta melakukan pengelolaan terhadap karya saya ini dengan mengacu pada ketentuan akses tugas akhir elektronik sebagai berikut (beri tanda pada kotak yang sesuai):

- a. Saya mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA
- b. Saya tidak mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA\*\*

\* Hak yang tidak terbatas hanya bagi satu pihak saja. Pengajar, peneliti, dan mahasiswa yang menyerahkan hak non-eksklusif kepada Repositori Perpustakaan Universitas saat mengumpulkan hasil karya mereka masih memiliki hak copyright atas karya tersebut.  
\*\* Hanya akan menampilkan halaman judul dan abstrak. Pilihan ini harus dilampiri dengan penjelasan/ alasan tertulis dari pembimbing TA dan diketahui oleh pimpinan fakultas (dekan/prodi).

Demikian pernyataan ini saya buat dengan sebenarnya.

Salatiga, 0 MEI 2019

TUTUS WIDIYAHINE TYAS  
Tanda tangan & nama terang mahasiswa

Mengetahui,

  
Evany Mailen  
Tanda tangan & nama terang pembimbing I

Tanda tangan & nama terang pembimbing II



### PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama : TUTUS WIDIYANING TYAS  
NIM : 672015258 Email : tutuswidiyaningtyas@gmail.com  
Fakultas : FTI Program Studi : TI  
Judul tugas akhir : PERANCANGAN SISTEM INFORMASI MANAJEMEN  
RUANG DAN ASET DI LPMI UKSW BERBASIS  
WEBSITE  
Pembimbing : 1. EVANGS MAILLOA, S.KOM. M.Cs.  
2. \_\_\_\_\_

Dengan ini menyatakan bahwa:

1. Hasil karya yang saya serahkan ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar kesarjanaan baik di Universitas Kristen Satya Wacana maupun di institusi pendidikan lainnya.
2. Hasil karya saya ini bukan saduran/terjemahan melainkan merupakan gagasan, rumusan, dan hasil pelaksanaan penelitian/implementasi saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing akademik dan narasumber penelitian.
3. Hasil karya saya ini merupakan hasil revisi terakhir setelah diujikan yang telah diketahui dan disetujui oleh pembimbing.
4. Dalam karya saya ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali yang digunakan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari terbukti ada penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya saya ini, serta sanksi lain yang sesuai dengan ketentuan yang berlaku di Universitas Kristen Satya Wacana.

Salatiga, 8 Mei 2019  
  
6000  
Rp. 6000  
(Tutus Widiyaning Tyas)

## 1. Pendahuluan

Lembaga Penjaminan Mutu dan Audit Internal (LPMAI) merupakan salah satu lembaga yang berdiri di Universitas Kristen Satya Wacana. Sebagai lembaga penjaminan mutu di tingkat Universitas LPMAI mempunyai tanggung jawab untuk menyusun sistem penjaminan mutu dan terus – menerus meningkatkan serta mengembangkan sistem penjaminan mutu, baik itu pada tingkat universitas, fakultas maupun program studi.

Sistem informasi manajemen ruang rapat adalah salah satu contoh aplikasi yang akan diimplementasikan di LPMAI yang digunakan untuk mengelola waktu dan tempat dalam proses pemesanan ruang rapat. Rapat merupakan agenda yang paling umum di dunia pekerjaan. Sistem informasi yang terdapat di UKSW pun telah memiliki sistem informasi manajemen ruang untuk mengelola semua ruang kuliah dan ruang yang terdapat di beberapa gedung. Adapun perbedaan antara manajemen ruang tingkat universitas dan LPMAI yaitu, (1) aplikasi manajemen ruang pada LPMAI dirancang menggunakan bahasa pemrograman PHP sedangkan di Universitas menggunakan bahasa pemrograman .Net, dikarenakan bahasa pemrograman PHP memiliki performa dalam kekuatan aplikasi *website*. (2) aplikasi manajemen ruang LPMAI memperkirakan perhitungan anggaran guna peminjaman ruang rapat. (3) aplikasi manajemen ruang LPMAI terdapat sistem pelaporan atas penggunaan ruang rapat. Dikarenakan LPMAI mengelola ruang rapat sendiri yang tidak dibawah naungan Bagian Pelayanan Umum maka sistem yang berjalan untuk peminjaman ruang rapat masih sangat sederhana yang diharuskan untuk datang langsung ke tempat guna menanyakan ketersediaan ruang rapat tersebut. Minimnya informasi terkait ketersediaan ruang rapat harus segera dibenahi. Dalam segi pembukuan untuk peminjaman ruang rapat tersebut masih bersifat manual dengan cara menulis data peminjaman ruang rapat dibuku.

Aset merupakan barang atau benda yang terdiri dari benda yang bersifat bergerak dan benda yang bersifat tidak bergerak yang tercakup dalam kekayaan suatu instansi. Kebutuhan informasi mengenai data suatu aset sangatlah penting guna untuk memperbaiki kinerja atau efisiensi di dalam suatu instansi atau lembaga, dimana tiap aset memiliki umur dan cara perawatan yang berbeda-beda [1]. Aset juga mendukung kegiatan operasional suatu instansi setiap harinya, tidak adanya informasi yang tepat untuk mengelola aset dapat menghambat kegiatan operasional. Kebutuhan informasi mengenai data dan informasi suatu aset sangatlah penting guna untuk memperbaiki kinerja di dalam sebuah instansi.

Permasalahan di dalam penelitian ini adalah tentang apakah perancangan sistem informasi Manajemen Ruang dan Aset di LPMAI dapat memberikan informasi pengelolaan sesuai dengan yang diharapkan. Selain itu, penelitian ini juga diharapkan mampu melakukan monitoring ruang dan aset.

## 2. Kajian Pustaka

Pada penelitian yang berjudul “Sistem Informasi Manajemen Ruang (SIMERU) Kelas (Studi Kasus: FKTI Universitas Mulawarman)”, membahas mengenai sistem informasi manajemen ruang kelas yang membantu admin dalam

proses penjadwalan ruang perkuliahan sehingga meminimalisir adanya jadwal ruang kuliah yang berbenturan dan dapat membantu untuk pemesanan ruang kelas pengganti [2]. Penelitian ini memiliki persamaan tujuan dalam perancangan yaitu peminjaman ruang, namun juga memiliki perbedaan pada bahasa pemrograman yang digunakan, penelitian terdahulu menggunakan bahasa pemrograman JAVA sedangkan penelitian ini dibuat menggunakan PHP dengan kombinasi *javascript* agar sistem lebih interaktif.

Penelitian yang berjudul “Sistem Informasi Manajemen Pengelolaan Aset Daerah Berbasis Web di Kecamatan Simpang Mamplam Kab Bireuen”, bertujuan untuk merancang program aplikasi pengolahan data aset daerah yang dibuat dengan menggunakan PHP dan *My9Sql* sebagai *database*-nya, untuk membantu dan mempermudah dalam melakukan pengolahan data aset daerah di Kecamatan Simpang Mamplam [3]. Penelitian ini memiliki persamaan dalam perancangannya yaitu menggunakan *Mysql* sebagai *database* dan menggunakan UML sebagai perancangannya, namun juga memiliki perbedaan pada *feature* yang tersedia, penelitian terdahulu hanya melakukan pengolahan data asset sedangkan penelitian ini terdapat *feature* untuk peminjaman ruang.

Pada penelitian yang berjudul “Rancang Bangun Sistem Informasi Manajemen Aset Berbasis Web pada Rumah Sakit Labuang Baju Makassar”, bertujuan untuk merancang sistem informasi pada Rumah Sakit yang memudahkan dalam mengelola data dan membuat laporan. Aplikasi tersebut terfokus pada kegiatan manajemen aset rumah sakit dan aplikasi dibuat berbasis *website* [4]. Penelitian ini memiliki persamaan yaitu sama-sama mengelola laporan keuangan/asset setiap bulan dan berbasis *website*. Adapun perbedaannya ialah penelitian terdahulu hanya terfokus pada pengelolaan asset sedangkan penelitian ini terdapat pengelolaan peminjaman ruang.

Keuntungan menggunakan *CodeIgniter* saat membangun sebuah sistem aplikasi yaitu sebagai berikut :

1. Codeigniter adalah *framework* PHP yang bersifat *open-source*.
2. Memiliki ukuran yang kecil dibandingkan dengan *framework* lain.
3. Aplikasi yang dibuat menggunakan *CodeIgniter* bias berjalan cepat.
4. *CodeIgniter* menggunakan pola desain *Model-View-Controller* (MVC) sehingga satu file tidak terlalu berisi banyak kode yang menjadikan kode lebih mudah dibaca, dipahami dan dipelihara dikemudian hari.
5. *CodeIgniter* dapat diperluas sesuai dengan kebutuhan.
6. *CodeIgniter* terdokumentasi dengan baik informasi tentang pustaka kelas (*Library*) dan fungsi yang disediakan oleh *CodeIgniter* dapat diperoleh melalui dokumentasi yang disertakan di dalam paket distribusinya.
7. *CodeIgniter* memiliki security yang handal seperti *xss filtering*, *session encryption*, dll.
8. *CodeIgniter* bersifat tidak kaku, sehingga memberikan kebebasan kepada *developer* web untuk mengemnbangkan aplikasi berbasis web bahkan tanpa *framework*.
9. CodeIgniter mengizinkan pengembang web menggunakan *library* atau *helper* yang tidak disediakan oleh *codeigniter* seperti: Google Map API, Facebook API, *fpdf* dan lain sebagainya [5].



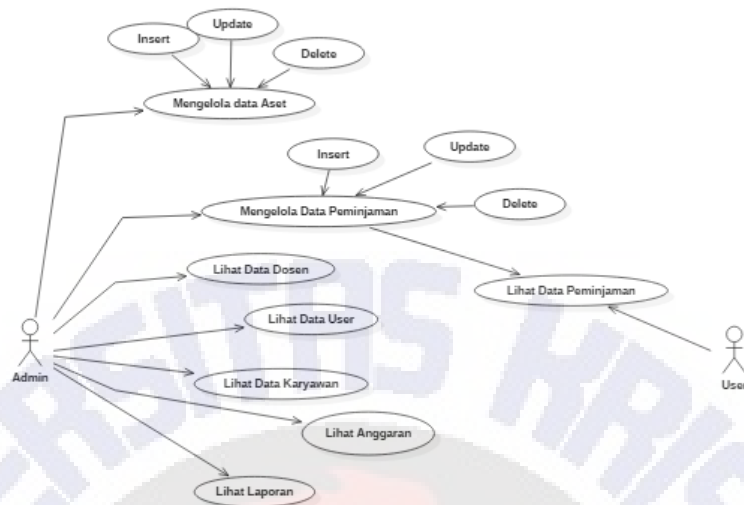
### 3. Tahapan Penelitian

Tahapan penelitian yang dilakukan dalam Perancangan Sistem Informasi Manajemen Ruang dan Aset di LPMAI menggunakan metode *waterfall*. Metode ini digunakan karena merupakan suatu metode yang praktis dan cukup menghemat biaya karena semua parameter-parameter yang dibutuhkan serta hasil yang diinginkan dapat langsung dimodelkan dan disimulasikan dengan menggunakan suatu program komputer (*Personal Computer*), dapat dilihat pada Gambar 1.



Gambar 1 Tahapan Penelitian

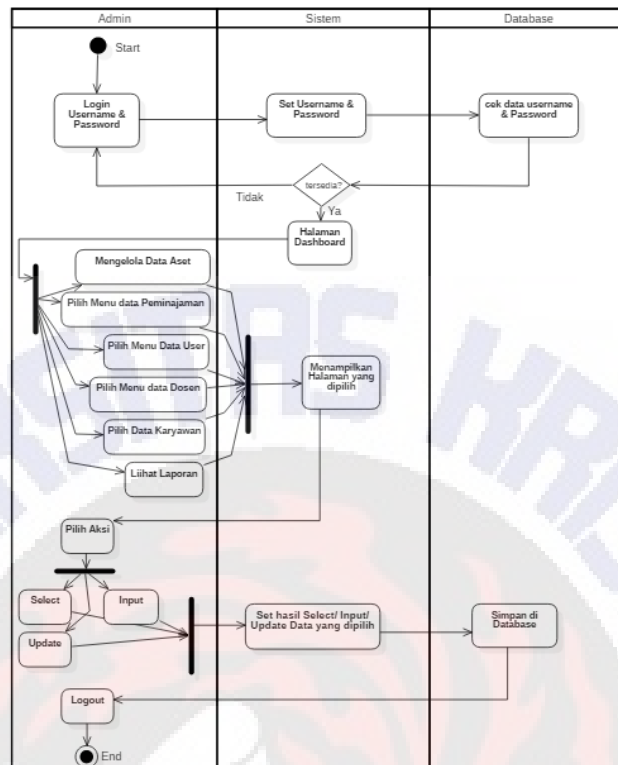
Tahapan penelitian pada Gambar 1 dijelaskan sebagai berikut : Analisis Kebutuhan, pada tahapan ini dilakukan proses pengumpulan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan user. Perancangan Aplikasi, pada tahap ini dilakukan pemodelan menggunakan UML untuk membuat alur program. UML yang telah dibuat akan diimplementasikan menjadi program pada tahap selanjutnya. Pembuatan Aplikasi, pada tahap ini dilakukan proses pembuatan aplikasi atau pembuatan kode program sesuai dengan UML yang sudah dibuat, proses pembuatan kode program ini menggunakan *framework CodeIgniter*. Setelah Pembuatan Kode Program selesai maka lanjut ke tahap Pengujian dimana tahap ini program yang sudah jadi akan dijalankan untuk diuji apakah dapat berjalan dengan baik dan benar untuk meminimalisir kesalahan (*Error*) dan memastikan *output* sesuai yang diinginkan. Jika terdapat kesalahan maka akan dilakukan perbaikan sampai program sesuai dengan kebutuhan. Tahap terakhir adalah Maintenance, pada tahap ini dilakukan proses pengembangan mulai dari analisis spesifikasi untuk perubahan program yang sudah ada.



**Gambar 2 Use Case Diagram**

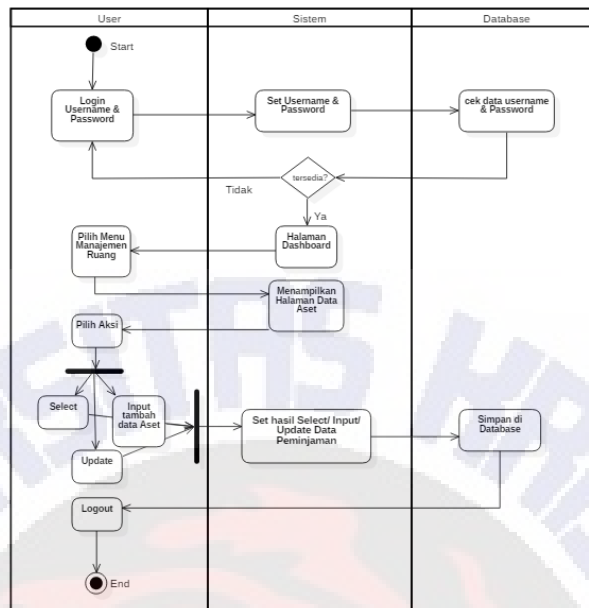
*Use Case Diagram* merupakan hal yang paling penting untuk memvisualisasikan, menspesifikasi dan mendokumentasikan kebutuhan perilaku sistem. *Use Case Diagram* biasa digunakan untuk menjelaskan kegiatan apa yang dapat dilakukan pengguna sistem yang sedang aktif [6]. Gambar 2 menunjukkan bahwa terdapat dua aktor dalam sistem yaitu admin dan *user*. Kedua aktor memiliki hak akses yang berbeda dalam mengelola aplikasi. *Admin* dapat mengakses seluruh aktifitas aplikasi seperti pengelolaan data asset LPMAI, pengelolaan peminjaman ruang rapat, input data dosen dan karyawan, lihat Laporan dan data Anggaran. Sedangkan *User* hanya dapat melakukan input peminjaman ruang rapat LPMAI dan melihat data peminjaman ruang rapat.

Alur aktifitas aktor dari awal hingga berakhirnya suatu *use case* digambarkan oleh *Activity Diagram*. *Activity Diagram* dari aplikasi ini merupakan aktifitas dari admin dan user dalam memasukkan data asset dan data peminjaman. *Activity Diagram* dapat dilihat pada gambar 3 dan gambar 4.



Gambar 3 Activity Diagram sebagai Admin

Gambar 3 digambarkan bahwa aktivitas aktor admin melakukan proses *login* terlebih dahulu di halaman *login* untuk dapat mengelola sistem informasi manajemen ruang dan asset di aplikasi. Aktor admin yang telah *login* maka dapat memilih fasilitas menu seperti menu data asset, data peminjaman, data *user*, data dosen, data karyawan serta lihat laporan. Admin dapat melakukan *insert*, *update*, *delete* pada fasilitas menu yang dipilih.



Gambar 4 Activity Diagram untuk User

Gambar 4 digambarkan bahwa *user* melakukan proses *login* terlebih dahulu di halaman *login* untuk dapat mengelola data peminjaman. Jika *username* dan *password* yang di inputkan tersedia di database maka akan langsung diarahkan ke Halaman Dashboard aplikasi dan memilih menu peminjaman untuk mengelola data peminjaman. Aktor *user* dapat melakukan *input* data peminjaman ruang rapat serta dapat mengubah data yang telah diinputkan sebelumnya.

The screenshot shows a web interface titled 'LPMAI' with a 'Logout' button in the top right. On the left, there is a vertical menu with 'Home', 'Menu 1', 'Menu 2', and 'Menu 3'. The main content area is titled 'Data Assets' and includes a 'Tambah' button above a table. The table has columns for 'ID', 'Nama', 'Kondisi', 'Lokasi', 'Biaya Main', and 'Aksi'. Each row in the table has two buttons: 'Edit' and 'Delete'.

Gambar 5 Design Interface Data Assets

Gambar 5 merupakan desain *Interface* pengelolaan manajemen data Aset. Pada bagian tengah terdapat *form table* untuk menampilkan data yang tersedia yang diambil dari *database*. Pada bagian samping kiri merupakan tombol menu untuk dapat berpindah ke menu – menu yang ada.

#### 4. Hasil dan Pembahasan

Penelitian ini menghasilkan suatu sistem dalam bentuk *website* dengan diimplementasikan menggunakan *browser*. Semua aktor akan mengakses aplikasi melalui *website*, berikut adalah hasil dan pembahasan sistem.

Form Peminjaman Ruang Rapat LPMAl-UKSW

Masukkan nama pemakai yang akan dicari

NO	Nama Peminjam	NIDN	Telepon	Status	Homebase	Jumlah	Keperluan	Judul Acara	Tanggal Pinjam	Tanggal Kembali	Jam	Biaya Pinjam	Aksi
1	MOZES KUNIAWAN	0602019002	08213328466	DOSEN	1002	30	rapat	rapat acara	2019-03-27	2019-03-27	13.42.42	Rp. 50000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
2	MUH. HARYANTO	0005125601	0816659063	DOSEN	1003	25	rapat	rapat kerja	2019-03-22	2019-03-22	13.00.00	Rp. 50000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

**Gambar 6** Tampilan Halaman Hasil Peminjaman

Pada gambar 6 merupakan tampilan untuk melihat data peminjaman ruang rapat yang dapat dilihat oleh admin maupun *user*. Proses peminjaman dilakukan dengan memilih tombol ‘Tambah’ diikuti dengan menginputkan data sesuai dengan yang diminta oleh sistem. Pada form peminjaman terdapat fungsi edit dan hapus dimana ketika user atau admin ingin mengedit data peminjaman dapat memilih aksi edit yang telah tersedia.

Tambah Data

Lokasi

Merk

Jenis Aset

Nama Aset

Kondisi

Warna

Harga

Biaya Maintenance

Keterangan

Tanggal Input

**Gambar 7** Tampilan Halaman *Insert Data*

Proses peminjaman ruang rapat LPMAl-UKSW dilakukan oleh admin dan juga *user*. Pada gambar 7 diarahkan untuk menginputkan data yang diminta oleh sistem yang akan tersimpan di dalam *database*. Pada form tambah data diperintahkan untuk mengisi Lokasi aset, Merk Aset, Jenis Aset, Nama Aset, Kondisi Aset, Warna, Harga, Biaya *Maintenance*, Keterangan dan tanggal input dengan benar dan tidak boleh kosong.

**Kode Program 1** Konsep *model get* Data Peminjaman

```

1. function get_data_list(){
2.     $this->db->select('*');
3.     $this->db->from('pinjam ');
4.     $this->db->order_by('id', 'ASC');
5.     $query = $this->db->get();
6.     return $query->result();
7. }

```

Kode program 1 menjelaskan tentang fungsi *get* data yang berfungsi untuk mengambil data yang akan ditampilkan pada halaman data peminjaman. Dijelaskan pada baris ke-1 yaitu membuat fungsi dengan nama *get\_data\_list* di dalam *class model* dimana fungsi tersebut akan dipanggil di *controller*. Baris ke-2 sampai ke-4 yaitu deklarasi pemanggilan semua data dari *table* pinjam berdasarkan *id* terkecil.

Baris ke-5 yaitu menjalankan *query* yang menghasilkan *select \* from pinjam order by id ASC*.

**Kode Program 2** Konsep *model insert* data Peminjaman

```
1. function insertData($data){
2.     $this->db->insert("pinjam", $data);
3. }
```

Kode Program 2 menjelaskan tentang fungsi *Insert* yang terdapat didalam folder *model* untuk dapat menambahkan data ke dalam *table* pinjam. Baris ke-2 merupakan deklarasi pemanggilan *table* pinjam agar data yang telah diinputkan dapat langsung masuk ke *table* pinjam.

**Kode Program 3** Konsep *model update* data Peminjaman

```
1. function editData($id,$data){
2.     $this->db->where('id',$id);
3.     return $this->db->update('pinjam',$data);
4. }
```

Kode program 3 menjelaskan tentang fungsi *update* di dalam konsep *model* untuk mengedit data di *table* pinjam. Baris ke-1 dijelaskan yaitu *function editData* yang terdapat di dalam *class model* dimana fungsi tersebut akan dipanggil di *controller* akan sistem dapat dijalankan. Baris ke-2 adalah deklarasi untuk menghasilkan *string* pembaharuan data ke dalam *table* pinjam berdasarkan *id* yang dipilih oleh user.

**Kode Program 4** Konsep *controller get* data peminjaman

```
1. public function index(){
2.     $data['isi'] = $this->M_datapeminjaman->get_data_list();
3.     $this->load->view('template/header');
4.     $this->load->view('template/menu_superadmin');
5.     $this->load->view('konten/datapeminjaman',$data);
6.     $this->load->view('template/footer');
7. }
```

Kode program 4 merupakan konsep *controller* untuk menampilkan data peminjaman di halaman manajemen peminjaman. Baris ke-2 adalah *variable* data untuk menampung isi yang di dalamnya memanggil data dari *model* datapeminjaman agar dapat ditampilkan di sistemnya. Baris ke-3 sampai ke-6 merupakan pemanggilan *view* dari template *header*, *menu\_superadmin* dan *footer* yang berada di dalam *folder template* serta *view* datapeminjaman yang telah dirancang di folder konten.

#### Kode Program 5 Konsep *controller* insert data peminjaman

```
1. public function insertData(){
2.     $data = array (
3.         "id" => $this->input->post('id'),
4.         "nama" => $this->input->post('nama'),
5.         "nidn" => $this->input->post('nidn'),
6.         "noTelp" => $this->input->post('noTelp'),
7.         "status" => $this->input->post('status'),
8.         "id_homebase" => $this->input->post('id_homebase'),
9.         "jmlorang" => $this->input->post('jmlorang'),
10.        "keperluan" => $this->input->post('keperluan'),
11.        "judulacara" => $this->input->post('judulacara'),
12.        "tglpinjam" => $this->input->post('tglpinjam'),
13.        "tglkembali" => $this->input->post('tglkembali'),
14.        "jam" => $this->input->post('jam'),
15.        "biaya" => $this->input->post('biaya'));
16.    $this->M_datapeminjaman->insertData($data);
17.    redirect('datapeminjaman');
18. }
```

Kode program 5 merupakan konsep *controller* untuk menambah data peminjaman. Pada baris ke-2 sampai ke-15 dijelaskan bahwa *array* yang diindeks dengan nama data akan menampung data id, nama, nidn, noTelp, status, id\_homebase, jmlorang, keperluan, judulacara, tglpinjam, tglkembali, jam dan biaya yang diambil dari *database table* pinjam. Baris ke-16 merupakan deklarasi untuk memanggil fungsi *insertData* dari *model* M\_datapeminjaman dimana data yang akan ditambahkan telah ditampung dalam *array* data.

#### Kode Program 6 Konsep *controller* update data peminjaman

```
1. public function editData(){
2.     $data = array (
3.         "id" => $this->input->post('id'),
4.         "nama" => $this->input->post('nama'),
5.         "nidn" => $this->input->post('nidn'),
6.         "noTelp" => $this->input->post('noTelp'),
7.         "status" => $this->input->post('status'),
8.         "id_homebase" => $this->input->post('id_homebase'),
9.         "jmlorang" => $this->input->post('jmlorang'),
10.        "keperluan" => $this->input->post('keperluan'),
11.        "judulacara" => $this->input->post('judulacara'),
12.        "tglpinjam" => $this->input->post('tglpinjam'),
13.        "tglkembali" => $this->input->post('tglkembali'),
14.        "jam" => $this->input->post('jam'),
15.        "biaya" => $this->input->post('biaya'));
16.    $this->M_datapeminjaman->editData($this->input->post('idedit'),$data);
17.    redirect('datapeminjaman');
18. }
```

Kode program 6 merupakan konsep *controller* untuk mengubah data pada halaman data peminjaman. Pada baris ke-2 sampai ke-15 dijelaskan bahwa *array* yang diindeks dengan nama data akan menampung data – data seperti : id, nama, nidn, noTelp, status, id\_homebase, jmlorang, keperluan, judulacara, tglpinjam, tglkembali, jam dan biaya yang diambil dari *database table* pinjam. Baris ke-16 merupakan deklarasi untuk memanggil fungsi *editData* dari *model* M\_datapeminjaman dimana data yang akan diubah telah ditampung terlebih dahulu di dalam *array* data berdasarkan idedit. Baris ke-17 dijelaskan bahwa akan dialihkan kembali ke halaman datapeminjaman.

Manajemen Data Aset

Berikut ini adalah daftar data Assets.

+Tambah Data Baru

Show 10 entries

Search:

No	Lokasi	Merk	Jenis Aset	Nama Aset	Kondisi	Warna	Harga	Biaya Maintenance	Keterangan	Tanggal Input	Aksi
1	Ruang Karyawan	Lenovo	Elektronik	Komputer All in One	Bagus	Putih	4700000	0	-	2019-03-12	Edit Hapus
2	Ruang Rapat	Sharp	Elektronik	AC	Bagus	Putih	2500000	500000	service	2019-03-12	Edit Hapus
3	Dapur	-	Peralatan Rumah Tangga	Kompor Gas	Bagus	Putih	123000	0		2019-03-14	Edit Hapus
4	Ruang Karyawan	Epson	Elektronik	Printer	Bagus	Hitam	1090000	120000	service	2019-03-14	Edit Hapus
5	Ruang Direktur	Sharp	Elektronik	AC	Bagus	Putih	2500000	0		2019-03-15	Edit Hapus
6	Ruang Karyawan	Canon	Elektronik	Mesin Foto Copy	Bagus	Putih	52000000	1500000	Biaya Maintenance Tahunan	2018-12-23	Edit Hapus
7	Ruang Karyawan	Samsung	Elektronik	Printer Samsung	Bagus	Hitam	2540000	460000	service	2019-03-04	Edit Hapus
8	Ruang Karyawan	Canon	Elektronik	Scanner	Bagus	Hitam	2300000	0		2019-03-15	Edit Hapus
9	Ruang Rapat	Tokai	Meuble	whiteboard	Bagus	putih	313000	0		2019-03-03	Edit Hapus
10	Ruang Rapat	Ergotec	Meuble	Furniture Kursi Kantor	Bagus	Hitam	345000	0	20 Kursi	2019-03-14	Edit Hapus

Showing 1 to 10 of 11 entries

Previous 1 2 Next

**Gambar 8** Tampilan Halaman Hasil data Aset

Gambar 8 merupakan tampilan dari hasil asset yang hanya dapat dilihat oleh user admin saja. Pada form manajemen data aset dilengkapi dengan harga awal aset dan biaya *maintenance* diikuti dengan tanggal *input* data aset. Proses pengelolaan *insert* data aset dilakukan dengan memilih tombol “Tambah Data Baru”. Pada form manajemen data aset terdapat fungsi edit dan hapus dimana admin dapat mengedit data aset yang sudah ada.

**Kode Program 7** Konsep *model get* Data Aset

```

1. function get_data_list(){
2.     $this->db->select('*');
3.     $this->db->from('input_aset');
4.     $this->db->order_by('no_aset','ASC');
5.     $query = $this->db->get();
6.     return $query->result();
7. }

```

Kode program 7 menjelaskan tentang fungsi *get* data atau mengambil data aset dari *database* untuk dapat ditampilkan dihalaman manajemen data aset. Dijelaskan pada baris ke-1 yaitu membuat fungsi dengan nama *get\_data\_list* didalam *class* model dimana fungsi tersebut akan dipanggil di *controller*. Baris ke-2 sampai ke-4 yaitu deklarasi pemanggilan semua data dari table *input\_aset* berdasarkan *no\_aset* terkecil. Baris ke-5 yaitu menjalankan query yang menghasilkan *select \* from input\_aset order by no\_aset ASC*.

**Kode Program 8** Konsep *model insert* Data Aset

```

1. function insertData($data){
2.     $this->db->insert("input_aset", $data);
3. }

```

Kode Program 8 menjelaskan tentang fungsi *insert* yang terdapat di dalam konsep *model insert* yang berfungsi untuk menambahkan data ke dalam *table* *input\_aset*. Baris ke-1 dijelaskan yaitu *function insertData* yang terdapat di dalam *class model* dimana fungsi tersebut akan dipanggil di *controller* untuk menjalankan



sistem nya. Baris ke-2 merupakan deklarasi pemanggilan *table* *input\_aset* agar data yang telah diinputkan dapat ditambahkan ke *table* *input\_aset* dan menjalankan *query*.

#### Kode Program 9 Konsep *model update* data Aset

```
1. function editData($id,$data){
2.     $this->db->where('no_aset',$id);
3.     return $this->db->update('input_aset',$data);
4. }
```

Kode program 9 menjelaskan tentang fungsi *update* yang terdapat di dalam konsep *model update* untuk mengubah data di *table* *input\_aset*. Baris ke-1 dijelaskan yaitu *function editData* yang terdapat di dalam *class model* dimana fungsi tersebut akan dipanggil di *controller* untuk menjalankan sistem nya. Baris ke-2 merupakan deklarasi untuk menghasilkan *string* pembaharuan data ke dalam *table* *input\_aset* berdasarkan *no\_aset* yang dipilih oleh *user*.

#### Kode Program 10 Konsep *controller get* data aset

```
1. public function index(){
2.     $data['isi'] = $this->M_manajemendataaset->get_data_list();
3.     $this->load->view('template/header');
4.     $this->load->view('template/menu_superadmin');
5.     $this->load->view('konten/manajemendataaset', $data);
6.     $this->load->view('template/footer');
7. }
```

Kode program 10 merupakan konsep *controller* untuk menampilkan data aset di halaman manajemen data Aset. Baris ke-2 adalah *variable* data untuk menampung isi yang di dalamnya berfungsi untuk memanggil data dari *model* manajemen dataset agar dapat ditampilkan di sistemnya. Baris ke-3 sampai ke-6 merupakan pemanggilan *view* dari *template header*, *menu\_superadmin* dan *footer* yang berada di dalam *folder template* serta *view* manajemen data aset yang telah dirancang di folder *konten* dengan memanggil *variable* data.

#### Kode Program 11 Konsep *controller insert* data aset

```
1. public function insertData(){
2.     $data = array (
3.         "no_aset" => $this->input->post('no_aset'),
4.         "area" => $this->input->post('area'),
5.         "merk" => $this->input->post('merk'),
6.         "jenis_aset" => $this->input->post('jenis_aset'),
7.         "nama_aset" => $this->input->post('nama_aset'),
8.         "kondisi" => $this->input->post('kondisi'),
9.         "warna" => $this->input->post('warna'),
10.        "Harga" => $this->input->post('Harga'),
11.        "biaya_main" => $this->input->post('biaya_main'),
12.        "ket" => $this->input->post('ket'),
13.        "tgl_input" => $this->input->post('tgl_input'));
14.     $this->M_manajemendataaset->insertData($data);
15.     redirect('manajemendataaset');
16. }
```

Kode program 11 merupakan konsep *controller* untuk menambah data aset ke dalam database yang akan ditampilkan di halaman manajemen data aset. Pada baris ke-2 sampai ke-13 dijelaskan bahwa *array* yang diindeks dengan nama data akan menampung data *no\_aset*, *area*, *merk*, *jenis\_aset*, *nama\_aset*, *kondisi*, *warna*,

Harga, biaya\_main, ket, dan tgl\_input yang diambil dari *database table input\_aset*. Baris ke-14 merupakan deklarasi untuk memanggil fungsi *insertData* dari *model M\_manajemendataaset* dimana data yang akan ditambahkan telah ditampung dalam *array data*.

**Kode Program 12** Konsep *controller update data aset*

```
1. public function editData(){
2.     $data = array (
3.         "no_aset" => $this->input->post('no_aset'),
4.         "area" => $this->input->post('area'),
5.         "merk" => $this->input->post('merk'),
6.         "jenis_aset" => $this->input->post('jenis_aset'),
7.         "nama_aset" => $this->input->post('nama_aset'),
8.         "kondisi" => $this->input->post('kondisi'),
9.         "warna" => $this->input->post('warna'),
10.        "Harga" => $this->input->post('Harga'),
11.        "biaya_main" => $this->input->post('biaya_main'),
12.        "ket" => $this->input->post('ket'),
13.        "tgl_input" => $this->input->post('tgl_input');
14.    $this->M_manajemendataaset->editData($this->input->post('idedit'),$data);
15.    redirect('manajemendataaset');
16.    }
```

Kode program 12 merupakan konsep *controller* untuk mengubah data pada halaman manajemen data aset. Pada baris ke-2 sampai ke-13 dijelaskan bahwa *array* yang diindeks dengan nama data akan menampung data – data yang akan diubah seperti : id, nama, nidn, noTelp, status, id\_homebase, jmlorang, keperluan, judulacara, tglpinjam, tglkembali, jam dan biaya yang diambil dari *database table input\_aset*. Baris ke-14 merupakan deklarasi untuk memanggil fungsi *editData* dari *model M\_manajemendataaset* dimana data yang akan diubah telah ditampung terlebih dahulu di dalam *array data* berdasarkan *idedit*. Baris ke-17 dijelaskan bahwa akan dialihkan kembali ke halaman *manajemendataaset*.

## Laporan data Aset LPMAI

Bulan :  
 1956

Output

**Gambar 9** Halaman tampilan laporan data aset

Pada gambar 9 merupakan tampilan halaman laporan data aset untuk menampilkan *form table* data aset perbulan. Output dari laporan data aset akan di tampilkan berdasarkan pilihan bulan dari januari hingga desember serta pilihan output yaitu PDF maupun WEB.

#### Kode Program 13 Konsep *Model* untuk laporan data aset

```
1. function get_report_aset($bulan){
2.     $this->db->select('*');
3.     $this->db->from('input_aset');
4.     $this->db->order_by('no_aset','ASC');
5.     $this->db->where('MONTH(tgl_input)', $bulan);
6.     $query = $this->db->get();
7.     return $query->result();
8. }
```

Kode program 13 merupakan konsep *model* untuk menampilkan laporan data aset. Baris ke-1 adalah *function get\_report\_aset* yang terdapat di dalam *class model* Manajemen Aset dimana fungsi tersebut akan di panggil di dalam *controller*. Baris ke-2 sampai ke-5 merupakan deklarasi pemanggilan dari fungsi *select* untuk menampilkan semua data dari *table* *input\_aset* berdasarkan *no\_aset* terkecil dan bulan yang telah dipilih dari *array* bulan. Baris ke-6 yaitu menjalankan query yang didapat dari baris ke-2 hingga ke-5.

#### Kode Program 14 *Controller* untuk Laporan Data Aset

```
1. public function report_Data(){
2.     $bulan = $this->input->post('bulan');
3.     $type = $this->input->post('type');
4.     if($this->input->post('type') == 'WEB'){
5.         $data['isi'] = $this->M_manajemendataaset-> get_report_aset($bulan);
6.         $this->load->view('template/header');
7.         $this->load->view('template/menu_superadmin');
8.         $this->load->view('konten/reportwebaset', $data);
9.         $this->load->view('template/footer');
10.    }else{
11.        $data['isi'] = $this->M_manajemendataaset->get_report_aset($bulan);
12.        $this->pdf->setPaper('A4','landscape');
13.        $this->pdf->filename='laporan aset.pdf';
14.        $this->pdf->load_view('konten/reportwebasetpdf', $data);
15.    }
16. }
```

Kode program 14 menjelaskan tentang fungsi yang terdapat di *controller* untuk laporan data aset dimana fungsi tersebut akan dipanggil di *model* untuk menghasilkan laporan dalam bentuk WEB atau PDF. Baris ke-4 dan ke-5 menjelaskan bahwa jika menginputkan dengan type WEB maka akan menampilkan data aset yang diambil dari model *M\_manajemendataaset* dalam fungsi *get\_report\_aset*. Baris ke-6 sampai ke-9 merupakan pemanggilan untuk tampilan dari *class view* dengan *template header*, *menu\_superadmin*, *footer* dan tampilan dari *reportwebaset*. Apabila input yang dipilih selain type WEB maka akan menampilkan data aset yang diambil dari model *M\_manajemendataaset* dalam fungsi *get\_report\_aset*. Baris ke-12 sampai ke-14 merupakan pengaturan untuk tampilan dihalaman PDF dari mengatur kertas menjadi ukuran A4 dengan *layout landscape*, memberi nama menjasi laporan aset.pdf serta tampilan didalam nya yang diambil berdasarkan konten *view*.

## Anggaran Dasar

Cetak

NO	Periode	Anggaran Pengajuan	Anggaran Valid
1	2018	Rp. 52000000	Rp. 1500000
2	2019	Rp. 18511000	Rp. 1380000

**Gambar 10** Tampilan halaman Anggaran dasar

Gambar 10 merupakan tampilan dari halaman anggaran dasar per tahun. Pada tampilan anggaran terdapat *feature* Cetak anggaran dalam bentuk PDF. Anggaran yang ditampilkan yaitu berupa anggaran pengajuan pertahun serta anggaran valid yang telah digunakan pertahun.

Berdasarkan hasil kuesioner yang telah dilakukan dengan staff LPMAI dapat dilakukan perhitungan untuk dapat diambil kesimpulan. Kuesioner ini terdiri dari 10 pertanyaan dengan menggunakan skala likert 1 sampai 4 dengan ketentuan skala untuk tiap pertanyaan sebagai berikut:

1. Tampilan sistem informasi manajemen ruang dan aset
2. Kemudahan pengoperasian sistem informasi manajemen ruang dan aset
3. Kecepatan akses sistem informasi manajemen ruang dan aset
4. Kelengkapan informasi yang disajikan pada sistem informasi
5. Fitur yang memadai pada sistem informasi manajemen ruang dan aset
6. Apakah aplikasi ini dapat memudahkan pengguna ?
7. Ketepatan fungsi tombol dengan tujuan menu yang diinginkan
8. Kesesuaian sistem informasi manajemen ruang dan aset dengan kebutuhan
9. Kemudahan dalam melakukan transaksi pemesanan pada sistem informasi
10. Kenyamanan menggunakan system informasi secara keseluruhan

Dari hasil kuesioner tersebut, dicari persentasi masing – masing jawaban dngan menggunakan rumus :

$$Y = \frac{P \times 100}{Q \times R}$$

Keterangan : P= Banyaknya jawaban responden, Q = Jumlah Responden, R = Nilai Persentase, Y= Nilai Persentase.

Berikut adalah hasil persentase masing – masing jawaban yang sudah dihitung nilainya dengan rumus diatas. Kuesioner ini diujikan kepada 10 orang.

$$Presentase = \frac{\text{Total jawaban responden} \times 100}{2 \times \text{Banyak Soal}}$$

**Table 1** Hasil Perhitungan Kuesioner

1	Pengujian Sistem Informasi									
	Responden	TB	KB	B	SB	Total	TB%	KB%	B%	SB%
	1	0	3	7	0	10	3%	33%	64%	0%
	2	0	4	6	0	10				
	3	0	3	7	0	10				
	4	0	5	5	0	10				
	5	0	0	10	0	10				
	6	0	1	9	0	10				
	7	0	4	6	0	10				
	8	1	4	5	0	10				
	9	0	4	6	0	10				
	10	2	5	3	0	10				
	<b>Jumlah</b>	<b>3</b>	<b>33</b>	<b>64</b>	<b>0</b>	<b>100</b>				

Tabel 1 adalah hasil perhitungan dari kuesioner yang telah dilakukan dan disebarkan kepada staff LPMAI. Pada hasil perhitungan dijelaskan terdapat 10 responden dengan mengisi 10 soal. Dari jumlah yang telah didapat dari 10 responden dapat disimpulkan bahwa persentase Baik lebih tinggi dibandingkan yang skala yang lain dengan begitu sistem informasi ini sudah dapat menghasilkan output yang diharapkan.

## 5. Kesimpulan dan Saran

Berdasarkan penelitian yang telah dilakukan maka dapat disimpulkan bahwa sistem informasi manajemen ruang dan aset di LPMAI dapat dikembangkan menggunakan *framework CodeIgniter* memenuhi kinerja di LPMAI. Aplikasi ini dirancang agar dapat memberikan sebuah rancangan sistem informasi pengelolaan data peminjaman dan data aset yang terdiri dari: peminjaman ruang rapat, penggunaan aset dan laporan – laporan yang terkait sehingga dapat membantu mengurangi permasalahan. Dari hasil pengujian yang telah dilakukan dengan menggunakan kuesioner, maka sistem informasi Manajemen Ruang dan Aset sudah dapat menghasilkan output yang diharapkan.

Dari hasil penerapan sistem informasi ini diharapkan agar menambahkan grafik untuk memberikan penggambaran visual data dari manajemen aset.

## 6. Daftar Pustaka

- [1] Sari, Patma ratna & Joni Devitra. 2017. “Analisis dan Perancangan Sistem Informasi Manajemen Aset pada Kantor BMKG Provinsi Jambi”. Stikom Dinamika Bangsa Jambi: Jurnal Manajemen Sistem Informasi. Vol.2, No.3
- [2] Khairina, Dyna Maris, dkk. 2018. “Sistem Informasi Manajemen Ruang (SIMERU) KELAS studi kasus: FKTI Universitas Mulawarman”. Jurnal Ilmiah Ilmu Komputer. Vol. 13, No. 1.

- [3] Zaiyuni, 2016. “Sistem Informasi Manajemen Pengelolaan Aset daerah berbasis web di kecamatan simpang mamplam kabupaten bireuen”. Jurnal TIKFA Fakultas Ilmu Komputer : Universitas Almuslim. Aceh. Vol. 3. No.2.
- [4] Qalby, Nurul. 2017. “Rancang Bangun Sistem Informasi Manajemen Aset berbasis web pada Rumah Sakit Labuang Baji Makassar”. UIN Alaudin Makasar: Makassar.
- [5] Setiadi, M Fikri. 2017. “Beberapa keunggulan *CodeIgniter*”. (<http://mfikri.com/artikel/Beberapa-keunggulan-Codeigniter.html> diakses pada 15 Maret 2019).
- [6] Poetro, Rotstaland Dito Soeharto. 2017. “Sistem Informasi Aset Berbasis WEB studi kasus SD NEGERI 1 Gatak Delanggu”. Universitas Muhammadiyah Surakarta: Surakarta.