



**SISTEM INFORMASI MANAJEMEN INVENTARISASI BARANG
BERBASIS WEB PADA PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER**

SKRIPSI

Oleh

Hendro Agung Eko Wahyudi

NIM 092410101026

PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER

2015



**SISTEM INFORMASI MANAJEMEN INVENTARISASI BARANG
BERBASIS WEB PADA PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER**

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan pendidikan di Program Studi Sistem Informasi Universitas Jember dan mendapat gelar Sarjana Sistem Informasi

Oleh

Hendro Agung Eko Wahyudi

NIM 092410101026

PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER

2015

BAB 1. PENDAHULUAN

Bab ini merupakan langkah awal dari penulisan tugas akhir ini. Bab ini berisi latar belakang, perumusan masalah, tujuan dan manfaat, ruang lingkup studi, metodologi penelitian dan sistematika penulisan.

1.1 Latar Belakang

Perkembangan teknologi informasi saat ini sangat pesat. Hal ini karena teknologi informasi sudah menjadi bagian penting bagi manusia dalam kehidupan sehari-hari. Pada zaman yang maju saat ini, semua serba terkomputerisasi. Mulai pekerjaan perkantoran, perbankan, administrasi, perpustakaan, pendidikan, pemerintahan, dan banyak lagi sangat bergantung pada kecanggihan komputer dengan aplikasi di dalamnya. Tidak ada alasan lain mengapa semua pekerjaan mulai dilakukan dengan komputerisasi yaitu karena efektifitas dan efisiensinya. Dengan sistem yang dijalankan pada komputer, pekerjaan dapat dikerjakan dengan cepat, tepat, dan teliti. Dan semakin baik sistem dibuat, maka akan semakin mudah manusia menyelesaikan semua pekerjaan dan permasalahannya.

Berbagai sistem telah diterapkan dalam membantu proses pengarsipan secara digital dalam banyak lembaga agar data yang ada bisa tersimpan rapi, aman, dan mudah untuk ditemukan sewaktu dibutuhkan serta memperkecil kemungkinan terjadinya kerusakan data. Namun ditengah-tengah digitalisasi ini, masih ada beberapa lembaga yang memang melakukan pengarsipan data secara manual, yaitu menulisnya didalam buku besar. Beberapa juga sudah menggunakan komputer dengan memanfaatkan aplikasi perkantoran, namun berkas yang mereka buat setiap harinya akan bertambah dan tidak memiliki integrasi yang bagus. Jika akan menerbitkan laporan bulanan, harus membuka arsip berkas-berkas terdahulu yang sangat banyak.

Sebagai contoh dari kasus yang penulis ingin angkat dalam skripsi ini adalah proses pengarsipan barang yang dilakukan oleh kampus Program Studi Sistem Informasi. Kampus Program Studi Sistem Informasi masih menggunakan pencatatan

manual dengan buku. Ditinjau dari efektifitas dan efisiennya, hal ini sangat kurang. Dan yang perlu diperhatikan dari sistem inventarisasi barang yang sedang berjalan saat ini adalah belum terintegrasinya data-data barang inventaris sehingga keberadaan barang tidak teratur, tidak terdapat penyimpanan data berbasis *database* sehingga proses pengolahan data inventaris menjadi lama sehingga data atau informasi yang dihasilkan kurang akurat dan aktual.

Pengertian inventarisasi sendiri menurut (Sugono, Dendy, & et al, 2008) adalah pencatatan atau pendaftaran barang-barang milik kantor, (sekolah, rumah tangga, dan sebagainya) yang dipakai dalam melaksanakan tugas. Dasar hukum inventarisasi barang diatur jelas di dalam Peraturan Pemerintah No. 6 Tahun 2006 tentang Pengelolaan Barang Milik Negara/Daerah.

Bisa kita bayangkan betapa rumitnya jika kita ingin mengetahui barang apa saja yang dimiliki oleh suatu lembaga dalam waktu 10 atau 20 tahun yang lalu jika kita harus membuka berkas-berkas atau catatan mengenai barang tersebut dalam suatu buku, mungkin lebih dari satu tumpukan buku besar yang harus kita baca satu persatu dan akan memakan waktu yang cukup lama. Inventarisasi yang manual akan menyulitkan dalam pencarian data inventaris, atau bahkan kehilangan data tersebut.

Berdasarkan latar belakang dan permasalahan di atas, maka penulis mencoba mengangkat masalah ini menjadi judul laporan tugas akhir untuk memenuhi syarat kelulusan jenjang Strata Satu di Program Studi Sistem Informasi, yaitu “Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember”. Sistem yang berbasis *web* yang terintegrasi dengan segala kelebihannya diharapkan dapat mengatasi masalah-masalah yang timbul dalam pengerjaan secara manual sehingga dapat meningkatkan efisiensi dan data barang inventaris pada Program Studi Sistem Informasi menjadi lebih aman.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah, maka rumusan masalah pada penelitian ini adalah :

- a. Bagaimana membangun sistem informasi manajemen inventarisasi barang berbasis *web*.
- b. Bagaimana membangun sistem informasi yang mampu menghasilkan *output* berupa laporan inventarisasi barang.

1.3 Tujuan dan Manfaat

Tujuan dan manfaat berisi tentang tujuan dari penelitian pengembangan sistem informasi manajemen inventarisasi barang berbasis *web* pada Program Studi Sistem Informasi Universitas Jember. Sedangkan pada bagian manfaat berisi tentang manfaat apa yang akan diperoleh pada penelitian ini, baik bagi peneliti sendiri maupun bagi objek pada penelitian ini

1.3.1. Tujuan

Tujuan dari penelitian ini adalah sebagai berikut :

- a. Untuk membangun sistem informasi manajemen inventarisasi barang berbasis *web*.
- b. Untuk membangun sistem informasi yang mampu menghasilkan *output* berupa laporan inventarisasi barang.

1.3.2. Manfaat

Manfaat penelitian ini adalah :

- a. Manfaat bagi peneliti
 - 1) Mengetahui bagaimana proses pengembangan sistem informasi manajemen inventarisasi barang berbasis *web*.
 - 2) Membantu instansi dalam melakukan inventarisasi barang.
- b. Manfaat bagi objek penelitian
 - 1) Meningkatkan kinerja instansi dalam melakukan inventarisasi barang.

1.4 Ruang Lingkup Studi

Ruang lingkup studi dalam penelitian ini merupakan lingkup batasan-batasan masalah. Batasan masalah dalam penelitian ini diantaranya adalah sebagai berikut :

- a. Pengambilan jumlah sampel barang inventaris adalah semua data barang yang ada di Program Studi Sistem Informasi.
- b. Data diambil hanya dari Program Studi Sistem Informasi Universitas Jember.
- c. Penelitian ini terbatas pada kemudahan *input* data barang dan pembuatan laporan inventarisasi.
- d. Sistem informasi yang dikembangkan berbasis *web*.

1.5 Sistematika Penulisan

Sistematika penulisan dalam penyusunan tugas akhir ini adalah sebagai berikut:

- a. Pendahuluan
Bab ini terdiri dari latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah dan sistematika penulisan.
- b. Tinjauan Pustaka
Bab ini berisi tentang kajian materi, penelitian terdahulu dan informasi apa saja yang digunakan dalam penelitian ini. Dimulai dari kajian pustaka mengenai pengertian dari sistem sampai inventarisasi.
- c. Metodologi Penelitian
Bab ini menguraikan tentang metode apa yang dilakukan selama penelitian. Dimulai dari tahap pencarian permasalahan hingga pengujian aplikasi Sistem Informasi Manajemen Inventarisasi Barang (SIMBI) yang akan dibuat.
- d. Desain dan Perancangan Sistem
Bab ini berisi tentang gambaran dan kebutuhan sistem. Kebutuhan fungsional dan non-fungsional dan desain perancangan sistem, *usecase diagram*, *scenario*, *activity diagram*, *sequence diagram*, *class diagram* dan *entity relation diagram*.
- e. Hasil dan Pembahasan
Bab ini menjelaskan tentang hasil dan pembahasan dari penelitian yang telah dilakukan. Dengan menggambarkan dampak atau manfaat apa yang terjadi pada saat sebelum penggunaan sistem dan setelah penggunaan sistem.
- f. Penutup

Bab ini berisi kesimpulan dari penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.



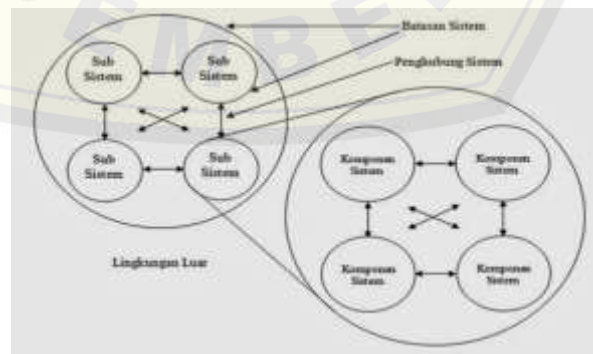
BAB 2. TINJAUAN PUSTAKA

Pada bab ini menguraikan secara detail mengenai teori yang mendasari penjelasan tentang isi dari bahan serta konsep yang akan dijadikan dalam kerangka pemikiran dalam penelitian.

2.1 Sistem Informasi

2.1.1. Sistem

Sistem adalah kumpulan elemen-elemen yang saling berhubungan dengan maksud yang sama untuk mencapai tujuan (McLeod, 2001). Sistem dapat didefinisikan dengan pendekatan prosedur dan pendekatan komponen yang didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu. Komponen sistem merupakan bagian kecil dari sebuah sub sistem. Beberapa komponen sistem bila menjadi satu akan menjadi sebuah sub sistem yang kemudian menjadi beberapa sub sistem dan sub sistem lainnya. Kumpulan dari beberapa sub sistem ini yang disebut dengan sistem dengan batasan dan penghubung di dalamnya (Mustakini, 2009). Penggambaran sistem menurut Mustakini (2009:34) disajikan dalam gambar 2.1



Gambar 2.1 Karakteristik Sistem
Sumber : mustakini, 2009

Sistem merupakan kumpulan atau *group* dari sub sistem atau bagian atau komponen apapun baik dari fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu (Susanto, 2004). Definisi lain sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu (Ladjamuddin. B, 2006). Berdasarkan definisi-definisi maka penulis dapat mengambil simpulan bahwa sistem merupakan kesatuan dari beberapa elemen yang saling berkaitan dan berhubungan untuk mencapai suatu tujuan tertentu.

2.1.2. Informasi

Informasi adalah data yang sudah diproses atau data yang memiliki arti (McLeod, 2001). Sedangkan Informasi adalah data yang telah dibentuk menjadi sesuatu yang berarti dan berguna bagi manusia (Laudon & Jane, 2000). Secara umum informasi dapat diartikan data yang telah diolah menjadi suatu bentuk yang penting bagi manusia dan mempunyai nilai yang nyata atau dapat dirasakan manfaatnya dalam keputusan atau kegiatan yang akan datang.

2.1.3. Sistem Informasi

Sistem Informasi adalah kombinasi dari manusia, fasilitas atau alat teknologi, media, prosedur dan pengendalian yang bermaksud menata jaringan komunikasi yang penting, proses atas transaksi-transaksi tertentu secara rutin, membantu manajemen dan pemakai intern dan ekstern dan menyediakan dasar pengambilan keputusan yang tepat (John, 1995). Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang sesuai dengan kebutuhan.

2.2 Pengertian Manajemen

Manajemen adalah proses perencanaan, pengorganisasian, pengarahan, dan pengawasan usaha-usaha para anggota organisasi dan penggunaan sumber daya-sumber daya organisasi lainnya agar mencapai tujuan organisasi yang telah ditetapkan (Handoko, 1997). Definisi lain manajemen adalah proses yang terdiri dari tindakan-tindakan, perencanaan, pengorganisasian, menggerakkan dan pengawasan, yang dilakukan untuk menentukan serta mencapai sasaran yang telah ditetapkan melalui pemanfaatan sumber daya manusia serta sumber-sumber yang lain (Terry, 2006). Dari definisi-definisi tersebut penulis simpulkan bahwa Manajemen adalah proses yang berupa tindakan-tindakan perencanaan, pengorganisasian, menggerakkan dan pengawasan serta pemanfaatan sumber daya untuk mencapai tujuan tertentu.

2.3 Pengertian Inventarisasi

Pergertian inventaris secara umum adalah daftar terperinci tentang barang-barang kantor, sekolah, perusahaan dan sebagainya. Inventaris adalah daftar yang memuat semua barang perusahaan dan sebagainya yang dipakai dalam melaksanakan tugas. Daftar yang dimaksud ialah berupa catatan tentang semua alat dan bahan yang disediakan untuk dipergunakan dalam pengolahan usaha yang dijalankan maupun sebagai peralatan operasional perusahaan (Ali, 2007). Inventarisasi sendiri adalah kegiatan untuk melakukan pendataan, pencatatan, dan pelaporan hasil pendataan Barang Milik Negara/Daerah (Republik Indonesia, 2014). Sedangkan menurut (Sugiana, 2009) inventarisasi ialah serangkaian kegiatan untuk melakukan pendataan, pencatatan, pelaporan hasil pendataan aset, dan mendokumentasikannya, baik aset berwujud maupun tidak berwujud pada suatu waktu tertentu. Dari beberapa pendapat diatas, dapat penulis simpulkan bahwa inventarisasi adalah proses yang dimulai dengan pengumpulan data, pencatatan, dan pelaporan seluruh aset atau barang beserta seluruh aspek legalnya. Inventarisasi digunakan untuk membuat sebuah daftar yang memuat semua aset atau barang yang dimiliki oleh organisasi, perusahaan, maupun instansi pemerintah baik itu yang berwujud ataupun tidak berwujud.

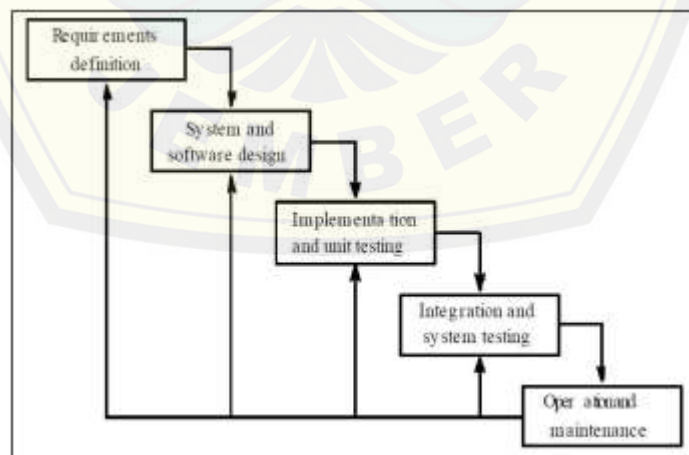
2.4 Model Waterfall

Menurut (Rosa & Salahuddin, 2013) Model SDCL air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

Teknik perancangan *waterfall* digunakan pada perancangan sistem informasi dikarenakan memiliki pendekatan yang lebih cocok di dalam pembuatan sistem baru. Secara terstruktur atau *Linear Sequential Model*. Dimana semua persyaratan didalam perancangan sudah dipahami mulai awal hingga akhir, seperti kebutuhan sistem, alokasi waktu, pengawasan dan lain-lain. Sehingga pengembangan perangkat lunak akan lebih mudah. Pada SDLC *waterfall* dipakai ketika pengerjakan proyek sistem baru, atau menghasilkan versi terbaru (*update, release* dan sebagainya).

Untuk melengkapi SDLC *waterfall*, digunakan OOAD (*Object Oriented Analyst and Design*). Dengan menerapkan berorientasi objek, sistem akan lebih tersruktur, mudah dalam *maintenance* dan mudah untuk dikembangkan.

Gambar ilustrasi untuk model *waterfall* dapat dilihat pada Gambar 2.2



Gambar 2.2 Model *Waterfall*
(Sumber: Sommerville, 2011)

Keterangan Gambar 2.2

- a. Analisis kebutuhan perangkat lunak
Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.
- b. Desain
Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.
- c. Pembuatan kode program
Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.
- d. Pengujian
Pengujian fokus pada perangkat lunak secara dari segi logik [*sic*] dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.
- e. Pendukung (*support*) atau pemeliharaan (*maintenance*)
Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke-*user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis

spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.



BAB 3. METODOLOGI PENELITIAN

Bab ini menguraikan tentang metode yang akan digunakan selama penelitian untuk membangun Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember.

3.1 Tujuan Penelitian

Penelitian ini dilakukan dengan tujuan untuk merancang dan membangun Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember.

3.2 Jenis Penelitian

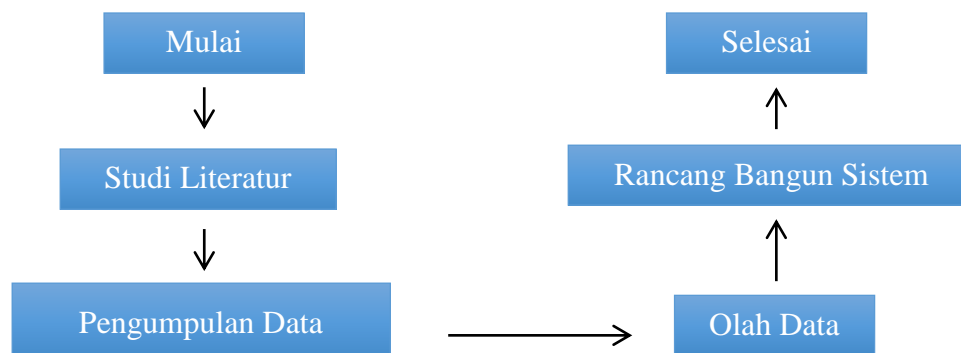
Penelitian kuantitatif merupakan sebuah penelitian yang berlangsung secara ilmiah dan sistematis dimana pengamatan yang dilakukan mencakup segala hal yang berhubungan dengan objek penelitian, fenomena serta korelasi yang ada diantaranya. Tujuan penelitian kuantitatif adalah untuk memperoleh penjelasan dari suatu teori dan hukum-hukum realitas. Penelitian kuantitatif dikembangkan dengan menggunakan model-model matematis, teori-teori dan atau hipotesis. Jenis penelitian yang dilakukan oleh penelitian ini masuk dalam jenis penelitian kuantitatif karena di dalam penelitian terdapat proses perhitungan.

3.3 Tempat dan Waktu Penelitian

Penelitian dilakukan di Program Studi Sistem Informasi Universitas Jember. Waktu penelitian dilaksanakan selama tiga bulan, Januari - Maret 2015.

3.4 Alur Penelitian

Alur penelitian menjelaskan urutan penelitian yang akan dilakukan mulai dari studi literatur, teknik pengumpulan data, dan perancangan sistem sampai dengan implementasi sistem yang terdapat pada Gambar 3.1.



Gambar 0.1 Diagram alir penelitian
Sumber Hasil Analisis, 2015

3.4.1. Studi Literatur

Studi literatur dibutuhkan untuk menunjang pemahaman dan pengetahuan penulis tentang materi, konsep, teori, dan metode apa yang diperlukan dalam proses pengerjaan tugas akhir ini. Studi literatur yang dilakukan meliputi penelitian-penelitian terdahulu di berbagai jurnal, buku, skripsi, dan *e-book*. Adapun hal-hal yang harus dipahami melalui studi literatur adalah beberapa penelitian terdahulu mengenai integrasi sistem manajemen data barang inventaris.

3.4.2. Tahap Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam mencapai tujuan penelitian. Data-data yang digunakan dalam penelitian ini berdasarkan sumber data meliputi data primer dan data sekunder :

1. Sumber Data Primer

Sumber data primer diperoleh langsung dari kantor bagian Tata Usaha pada Program Studi Sistem Informasi Universitas Jember. Meliputi data barang-barang dan kategori barang inventaris.

2. Sumber Data Sekunder

Dalam penelitian ini penulis memperoleh sumber data dari berbagai literature yang berkaitan dengan cara integrasi sistem pengarsipan data inventaris. Literatur ini diperoleh dari buku teks, jurnal yang diperoleh dari internet, dan penelitian sebelumnya.

3.4.3. Teknik Pengumpulan Data

Pengumpulan data diperlukan untuk mendapatkan data yang dibutuhkan dalam membangun Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember. Proses untuk mendapatkan data tersebut yaitu dengan cara:

1. Observasi

Metode observasi dilakukan oleh peneliti dengan mendatangi kantor bagian Tata Usaha pada Program Studi Sistem Informasi Universitas Jember yang bertujuan untuk mengamati secara langsung objek yang diteliti tersebut.

2. Wawancara

Observasi juga diikuti dengan melakukan wawancara langsung terhadap pihak yang terkait. Sehingga peneliti bisa mendapatkan data jenis barang-barang dan kategori barang sesuai dengan kebutuhan penelitian.

3.5 Metode Pengembangan Sistem

Metode yang digunakan dalam pembuatan Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember ini dengan menggunakan model *waterfall*. Model *waterfall* dibagi menjadi 4 tahapan, yaitu :

- 3.5.1. Analisis

Tahap analisis merupakan tahap pencarian dan analisa kebutuhan. Kebutuhan yang dimaksud, yaitu kebutuhan fungsional dan non-fungsional serta fitur yang akan dibangun pada sistem serta data yang dibutuhkan dalam membangun sistem.

- 3.5.2. Design

Sistem Informasi Manajemen Inventarisasi Barang dirancang menggunakan beberapa diagram yang dibuat untuk menentukan fitur-fitur dan kebutuhan fungsional sistem yang akan dibangun. Diagram tersebut antara lain:

1. *Usecase*
2. *Usecase Scenario*
3. *Sequence Diagram*

4. *Class Diagram*
5. *Entity Relationship Diagram (ERD)*
6. *Activity Diagram (ERD)*

3.5.3. *Coding*

Pada tahap implementasi ini, dimulai dengan pembuatan Sistem Informasi Manajemen Inventarisasi Barang. Dalam tahap implementasi, akan disesuaikan dengan desain sistem yang telah dikerjakan pada tahap perancangan sistem. Kegiatan yang dilakukan selama tahap implementasi antara lain :

1. Penulisan kode program (*coding*) menggunakan bahasa pemrograman PHP (*Page Hyper Text Pre-Processor*), CSS (*Cascading Style Sheet*), *Javascript* dan *framework CodeIgniter*.
2. Manajemen data menggunakan DBMS MySQL.

3.5.4. *Testing*

Tahap pengujian dilakukan apabila aplikasi yang dibuat telah selesai dan siap untuk digunakan pengguna. Pengujian yang dilakukan berguna untuk mengetahui kesalahan yang ada di dalam sistem dan mengetahui integrasi sistem. Tahapan pengujian dilakukan dengan mencari kesalahan-kesalahan yang mungkin terjadi, serta melakukan perbaikan untuk menyempurnakan sebuah sistem. Proses pengujian dilakukan dengan metode *whitebox* oleh pengembang dan *blackbox* oleh pengguna. Pengujian *whitebox* dilakukan untuk mengetahui apakah aplikasi yang dibangun dari segi desain dan program sesuai dengan kebutuhan. Sedangkan untuk pengujian *blackbox* dilakukan dengan memperhatikan masukan/keluaran (I/O) yang dihasilkan oleh sistem informasi manajemen inventarisasi barang tersebut. Dengan melakukan pengujian *blackbox* akan diketahui bahwa sistem sudah memiliki (I/O) yang sesuai dengan desain perancangan awal atau tidak.

BAB 4. DESAIN DAN PERANCANGAN SISTEM

Bab ini akan menguraikan tentang proses pendesainan dan perancangan sistem untuk mengimplementasi Sistem Informasi Manajemen Inventarisasi Barang Berbasis Web pada Program Studi Sistem Informasi Universitas Jember. Proses pendesainan dan perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, dilanjutkan dengan pembuatan *usecase diagram*, skenario, *activity diagram*, *sequence diagram*, *class diagram* dan *entity relation diagram* (ERD).

4.1 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak dalam penelitian ini yaitu dengan cara mengidentifikasi permasalahan yang ada untuk kemudian dicatat dan dijadikan bahan untuk mulai membangun aplikasi Sistem Informasi Manajemen Inventarisasi Barang Berbasis Web pada Program Studi Sistem Informasi Universitas Jember. Analisis kebutuhan yang dilakukan meliputi proses pengumpulan data kebutuhan fungsional, kebutuhan non-fungsional.

Kebutuhan fungsional sistem pada penelitian ini adalah sebagai berikut:

1. Sistem memiliki kemampuan dalam pengelompokan kategori barang dan ruangan.
2. Sistem dapat memberikan akses yang berbeda pada masing-masing jenis *user*.
3. Sistem dapat menambahkan data barang pada setiap kategori barang dan ruangan.
4. Sistem dapat menampilkan data barang berdasarkan kategori barang atau ruangan.
5. Sistem dapat menyimpan data laporan inventarisasi.
6. Sistem dapat mencetak laporan data barang dari hasil inventarisasi.

Kebutuhan non-fungsional sistem pada penelitian ini adalah :

1. Ketersediaan aplikasi, dalam hal ini sistem beroperasi 7 hari perminggu, 24 jam perhari (*Availability*).
2. Keandalan aplikasi, dalam hal ini sistem tidak boleh gagal, dapat digunakan secara bersamaan dengan komputer yang berbeda (*Reliability*).
3. Kemudahan pengoperasian sistem, dalam hal ini sistem bersifat fleksibel atau mampu dioperasikan pada beberapa sistem operasi, diantaranya Windows, Linux,

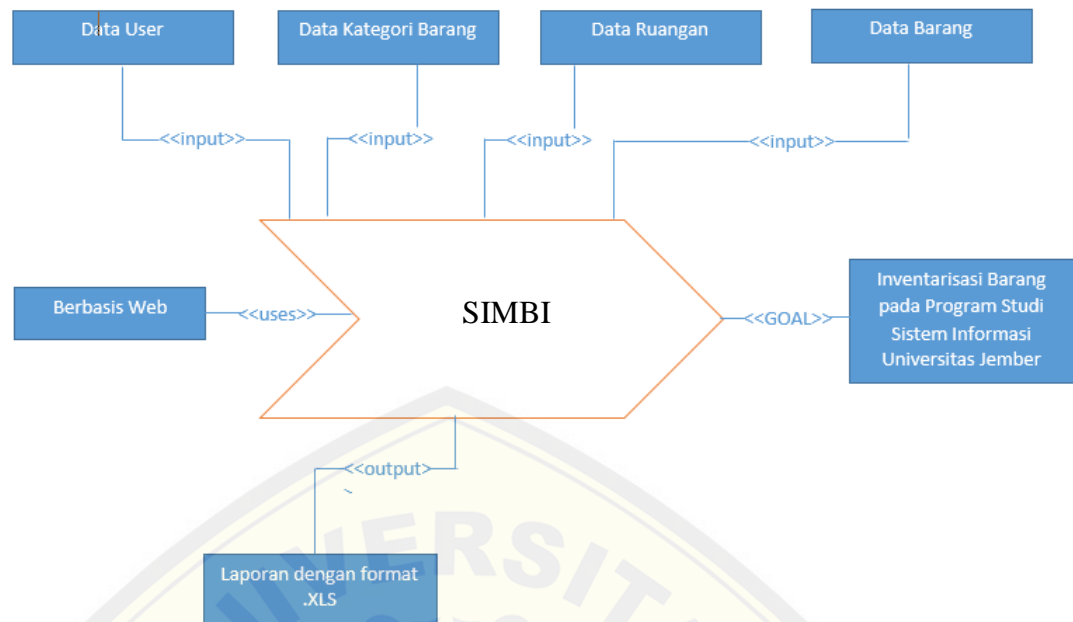
- Mac OS, Android, dan iOS (*Portability*).
4. Batasan waktu yang harus dipenuhi untuk *real time system*, dalam hal ini sistem harus mampu menampilkan hasil dalam waktu kurang dari 5 detik (*Respond Time*).
 5. Aspek keamanan yang harus dipenuhi, dalam hal ini sistem menggunakan *username* dan *password* sebagai syarat autentifikasi *user* untuk dapat mengakses sistem (*Security*).
 6. Bahasa komunikasi untuk sistem harus mudah dipahami oleh *user*, untuk menu atau fitur sistem menggunakan Bahasa Inggris atau Bahasa Indonesia, sedangkan bahasa untuk konten atau *list* menggunakan Bahasa Indonesia (*Communication*).

4.1.1 Deskripsi Umum Sistem

Sistem informasi manajemen inventarisasi barang berbasis web pada Program Studi Sistem Informasi Universitas Jember merupakan sistem pengolahan data barang inventaris yang menyajikan laporan penggunaan barang dan laporan data barang secara keseluruhan.

4.1.2 *Business Process*

Business Process Sistem Informasi Manajemen Inventarisasi Barang Berbasis Web pada Program Studi Sistem Informasi Universitas Jember pada gambar 4.1 menggambarkan sebuah proses sehingga menghasilkan *goal* yang dituju.

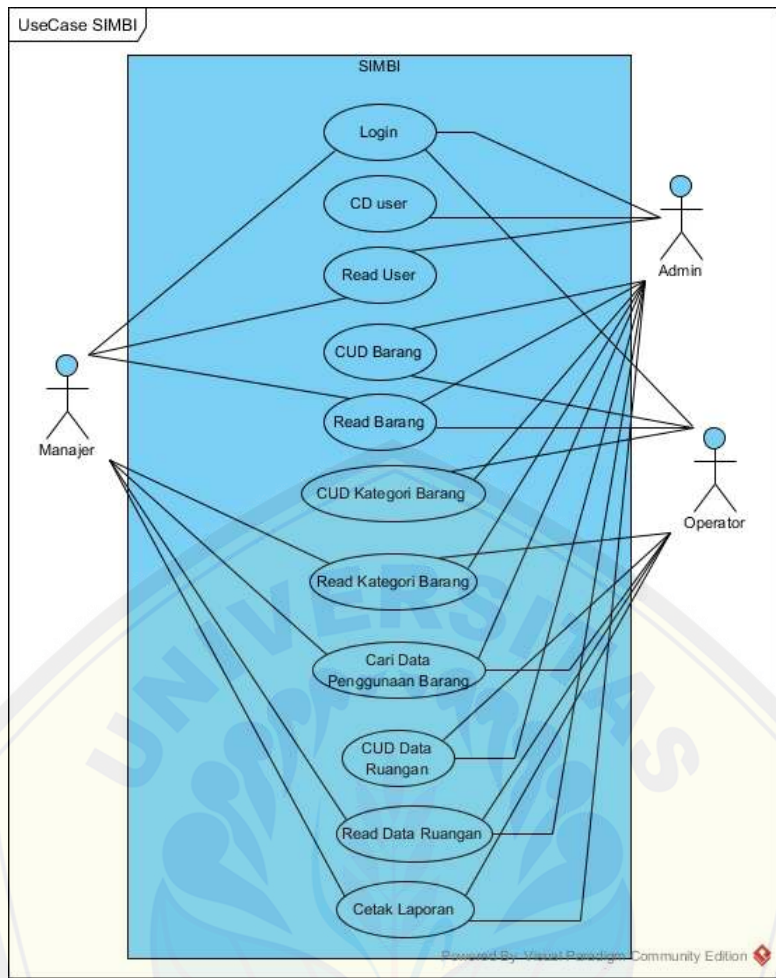


Gambar 4. 1 *Business Process* Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember

Sumber: Hasil Analisis, 2015

4.2 *Usecase Diagram*

Usecase diagram digunakan untuk menggambarkan fitur yang dibuat dalam sistem dan juga menggambarkan siapa saja yang dapat menggunakan fitur tersebut. *Usecase diagram* sistem informasi manajemen inventarisasi barang pada Program Studi Sistem Informasi Universitas Jember (SIMBI) dapat dilihat pada gambar 4.2:



Gambar 4. 2 Usecase SIMBI
 Sumber: Hasil Analisis, 2015

Pada *usecase diagram* tersebut terdapat tiga klasifikasi aktor yang dapat menggunakan sistem informasi inventarisasi barang. Tiga aktor tersebut adalah Admin (admin), Manajer (pimpinan), dan Operator (petugas inventarisasi). Deskripsi aktor pada *usecase* dijelaskan pada tabel 4.1 dan tabel 4.2 di bawah ini:

Tabel 4.1 Definisi Aktor Usecase

| No | Aktor | Definisi Tugas |
|----|-------|---|
| 1. | Admin | <i>Login, CD User, Read User, CUD Barang, Read Barang, CUD Kategori Barang, Read Kategori Barang, Cari Data Penggunaan Barang, CUD Data</i> |

| | | |
|----|----------|--|
| | | Ruangan, <i>Read</i> Data Ruangan, Cetak Laporan. |
| 2. | Operator | <i>Login</i> , CUD Barang, <i>Read</i> Barang, CUD Kategori Barang, <i>Read</i> Kategori Barang, Cari Data Penggunaan Barang, CUD Data Ruangan, <i>Read</i> Data Ruangan, Cetak Laporan. |
| 3. | Manajer | <i>Login</i> , <i>Read User</i> , <i>Read</i> Barang, <i>Read</i> Kategori Barang, Cari Data Penggunaan Barang, <i>Read</i> Data Ruangan, Cetak Laporan. |

Tabel 4.2 Definisi *Usecase*

| No. | <i>Usecase</i> | Deskripsi |
|-----|-----------------------------|--|
| 1. | <i>Login</i> | - Menampilkan portal <i>login</i> untuk mengisi <i>username</i> dan <i>password</i> |
| 2. | CD <i>User</i> | - Menambah <i>user</i> baru dengan memasukkan nama lengkap, NIP, <i>username</i> , <i>password</i> , jenis kelamin, <i>email</i> , dan level <i>user</i> . |
| 3. | <i>Read User</i> | - Menampilkan semua <i>user</i> yang terdaftar. |
| 4. | CUD Barang | - Menambah data barang dengan memasukkan nama kategori, nama barang, merk barang, jumlah, satuan, foto, ruangan. |
| 5. | <i>Read</i> Barang | - Menampilkan semua data barang yang telah diinput. |
| 6. | CUD Kategori Barang | - Menambah data kategori barang dengan memasukkan nama kategori barang. |
| 7. | <i>Read</i> Kategori Barang | - Menampilkan data kategori |

| | | |
|-----|-----------------------------|---|
| | | barang yang telah <i>diinput</i> . |
| 8. | Cari Data Penggunaan Barang | - Menampilkan penggunaan barang berdasarkan ruangan atau kategori barang. |
| 9. | CUD Data Ruangan | - Menambah data ruangan dengan memasukkan nama ruangan dan lokasi. |
| 10. | <i>Read</i> Data Ruangan | - Menampilkan data ruangan yang telah <i>diinput</i> . |
| 11. | Cetak Laporan | - Mencetak laporan hasil inventarisasi. |

4.3 Skenario

Menjelaskan alur cerita (skenario) pada setiap *usecase* yang telah digambarkan dalam *usecase* diagram secara lebih detail. *Usecase scenario* berisi id, nama *usecase*, aktor, *Pre Condition*, *Post Condition*, skenario normal, dan skenario alternatif.

4.3.1 Skenario Login

| | |
|--------------|-------------------------------|
| ID | USC001 |
| Nama | <i>Login</i> |
| Aktor | Manajer. |
| Prekondisi | Menampilkan Menu <i>Login</i> |
| Post Kondisi | Masuk SIMBI |

SKENARIO MANAJER

| Aksi Aktor | Reaksi Sistem |
|---|---------------------|
| Skenario Normal | |
| 1. <i>Owner</i> mengisi <i>field Username</i> dan <i>Password</i> | |
| 2. Klik tombol <i>Login</i> | |
| | 3. Cek <i>Login</i> |

| | |
|---|---|
| | <p>4. Sistem menampilkan halaman <i>home owner</i> dengan menu :</p> <ul style="list-style-type: none"> - Home - Data <i>User</i> - Kategori Barang - Ruangan - Data Barang - Data Penggunaan barang - Laporan |
| Skenario Alternatif Gagal <i>Login</i> | |
| | <p>3. Ketika <i>field Username</i> dan <i>Password</i> salah/ kosong, maka akan muncul pemberitahuan <i>Username/Password Salah</i></p> |
| 4. Klik Ok | |
| | <p>5. Kembali ke halaman <i>login</i></p> |
| ID | USC001 |
| Nama | <i>Login</i> |
| Aktor | Operator. |
| Prekondisi | Menampilkan Menu <i>Login</i> |
| Post Kondisi | Masuk SIMBI |
| SKENARIO OPERATOR | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Operator mengisi <i>field Username</i> dan <i>Password</i> | |
| 2. Klik tombol <i>Login</i> | |
| | 3. Cek <i>Login</i> |

| | |
|--|--|
| | <p>4. Sistem menampilkan halaman <i>home</i> operator dengan menu :</p> <ul style="list-style-type: none"> - Home - Kategori Barang - Ruangan - Data Barang - Data Penggunaan barang - Laporan |
| Skenario Alternatif Gagal <i>Login</i> | |
| | <p>4. Ketika <i>field Username</i> dan <i>Password</i> salah/ada yang kosong, maka akan muncul pemberitahuan <i>Username</i> atau <i>Password</i> Salah</p> |
| 5. Klik Ok | |
| | 6. Kembali ke halaman <i>login</i> |
| ID | USC001 |
| Nama | <i>Login</i> |
| Aktor | Admin. |
| Prekondisi | Menampilkan Menu <i>Login</i> |
| Post Kondisi | Masuk SIMBI |
| SKENARIO ADMIN | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Admin mengisi <i>field Username</i> dan <i>Password</i> | |
| 2. Klik tombol <i>Login</i> | |
| | 3. Cek <i>Login</i> |

| | |
|--|---|
| | <p>4. Sistem menampilkan halaman <i>home</i> admin dengan menu :</p> <ul style="list-style-type: none"> - Home - Data <i>User</i> - Kategori Barang - Ruangan - Data Barang - Data Penggunaan barang - Laporan |
| Skenario Alternatif Gagal <i>Login</i> | |
| | <p>4. Ketika <i>field Username</i> dan <i>Password</i> salah/ada yang kosong, maka akan muncul pemberitahuan <i>Username</i> atau <i>Password</i> Salah</p> |
| 5. Klik Ok | |
| | <p>6. Kembali ke halaman <i>login</i></p> |

4.3.2 Skenario CRUD *User*

| | |
|--------------|--|
| ID | USC002 |
| Nama | CD <i>User</i> |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | User telah bertambah |

SKENARIO *INPUT*

| Aksi Aktor | Reaksi Sistem |
|-------------------------------|---------------|
| Skenario Normal | |
| 1. Klik Menu Data <i>User</i> | |

| | |
|--|--|
| | <p>2. Sistem menampilkan halaman data <i>user</i> dengan menu :</p> <ul style="list-style-type: none"> - Tabel Data <i>User</i> - Tambah <i>User</i> - Delete <i>User</i> |
| 3. Klik menu Tambah <i>User</i> | |
| | 4. Sistem menampilkan <i>form</i> tambah <i>user</i> |
| 5. Mengisi <i>Form</i> | |
| 6. Klik Simpan | |
| | <p>7. Sistem menampilkan halaman data <i>user</i> dengan menu :</p> <ul style="list-style-type: none"> - Data <i>User</i> - Tambah <i>User</i> - <i>Update User</i> - Delete <i>User</i> |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan " <i>Please fill out this field</i> " |
| ID | USC002 |
| Nama | CD <i>User</i> |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | <i>User</i> telah berkurang |
| SKENARIO DELETE | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |

| | |
|--|--|
| 1. Klik Menu Data <i>User</i> | |
| | 2. Sistem menampilkan halaman data <i>user</i> dengan menu : <ul style="list-style-type: none"> - Data <i>User</i> - Tambah <i>User</i> - Delete <i>User</i> |
| 3. Klik <i>delete</i> | |
| | 4. Sistem menampilkan <i>popup</i> ‘Apakah anda yakin ingin menghapus?’ |
| 5. Klik OK | |
| | 6. Sistem menampilkan halaman data <i>user</i> dengan menu : <ul style="list-style-type: none"> - Data <i>User</i> - Tambah <i>User</i> - Delete <i>User</i> |
| Skenario Alternatif Data Batal Dihapus | |
| 3. Klik <i>Cancel</i> | |
| | 4. Sistem menampilkan halaman data <i>user</i> dengan menu : <ul style="list-style-type: none"> - Data <i>User</i> - Tambah <i>User</i> - Delete <i>User</i> |
| ID | USC003 |
| Nama | Read <i>User</i> |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |

| | |
|--------------|---|
| Post Kondisi | Menampilkan Halaman Data <i>User</i> |
|--------------|---|

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|-------------------------------|--|
| Skenario Normal | |
| 1. Klik Menu Data <i>User</i> | |
| | 2. Sistem menampilkan halaman data <i>user</i> dengan menu : - Data <i>User</i> - Tambah <i>User</i> - Delete <i>User</i> |

| | |
|--------------|--|
| ID | USC003 |
| Nama | Read <i>User</i> |
| Aktor | Manajer. |
| Prekondisi | Menampilkan Halaman <i>home</i> Manajer |
| Post Kondisi | Menampilkan Halaman Data <i>User</i> |

SKENARIO VIEW

| Aksi Aktor Manajer | Reaksi Sistem |
|-------------------------------|--|
| Skenario Normal | |
| 3. Klik Menu Data <i>User</i> | |
| | 4. Sistem menampilkan halaman data <i>user</i> dengan menu : - Data <i>User</i> |

4.3.3 Skenario CRUD Barang

| | |
|------|-------------------|
| ID | USC004 |
| Nama | CUD Barang |

| | |
|--|--|
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Barang telah bertambah |
| SKENARIO INPUT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |
| | 2. Sistem menampilkan halaman data barang dengan menu : - Data Barang - Tambah Barang - Edit Barang - Hapus Barang |
| 3. Klik menu Tambah Data | |
| | 4. Sistem menampilkan <i>form</i> tambah barang |
| 5. Isi <i>Form</i> | |
| 6. Klik Simpan | |
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil ditambahkan” |
| | 8. Sistem menampilkan halaman data barang dengan menu : - Data Barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| Skenario Alternatif Data Tidak Lengkap | |

| | |
|--------------------------|--|
| | 7. Sistem menampilkan pesan “Please fill out this field” |
| ID | USC004 |
| Nama | CUD Barang |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Barang telah bertambah |
| SKENARIO INPUT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | 2. Sistem menampilkan halaman data barang dengan menu : - Data Barang - Tambah Barang - Edit Barang - Hapus Barang |
| 3. Klik menu Tambah Data | 4. Sistem menampilkan <i>form</i> tambah barang |
| 5. Isi <i>Form</i> | |
| 6. Klik Simpan | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil ditambahkan” |
| | 8. Sistem menampilkan halaman data barang dengan menu : |

| | |
|---|---|
| | <ul style="list-style-type: none"> - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan “Please fill out this field” |
| ID | USC004 |
| Nama | CUD Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Barang telah teredit |
| SKENARIO EDIT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |
| | 2. Sistem menampilkan halaman data barang dengan menu : <ul style="list-style-type: none"> - Data Barang - Tambah Barang - Edit Barang - Hapus Barang |
| 3. Klik edit pada barang yang akan diedit | |
| | 4. Sistem menampilkan <i>form</i> yang terisi detail barang |
| 5. Edit <i>Form</i> | |
| 6. Klik Simpan | |

| | |
|--|--|
| | 7. Sistem menambahkan ke database dan Menampilkan <i>popup</i> “Data berhasil diedit” |
| | 8. Sistem menampilkan halaman data barang dengan menu : <ul style="list-style-type: none"> - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan <i>“Please fill out this field”</i> |
| ID | USC004 |
| Nama | CUD Barang |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Barang telah teredit |
| SKENARIO EDIT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |
| | 2. Sistem menampilkan halaman data barang dengan menu : <ul style="list-style-type: none"> - Data Barang - Tambah Barang - Edit Barang - Hapus Barang |

| | |
|---|--|
| 3. Klik edit pada barang yang akan diedit | |
| | 4. Sistem menampilkan <i>form</i> yang terisi detail barang |
| 5. Edit <i>Form</i> | |
| 6. Klik Simpan | |
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil diedit” |
| | 8. Sistem menampilkan halaman data barang dengan menu : <ul style="list-style-type: none"> - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan “ <i>Please fill out this field</i> ” |
| ID | USC004 |
| Nama | CUD Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | <i>User</i> telah berkurang |
| SKENARIO DELETE | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |

| | |
|--|--|
| | <p>2. Sistem menampilkan halaman data barang dengan menu :</p> <ul style="list-style-type: none"> - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| 3. Klik <i>delete</i> pada data yang ingin dihapus | |
| | <p>4. Sistem menampilkan <i>popup</i> ‘Apakah anda yakin ingin menghapus?’</p> |
| 5. Klik OK | |
| | <p>6. Sistem menampilkan halaman data barang dengan menu :</p> <ul style="list-style-type: none"> - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| Skenario Alternatif Data Batal Dihapus | |
| 4. Klik <i>Cancel</i> | |
| | <p>7. Sistem menampilkan halaman data barang dengan menu :</p> <ul style="list-style-type: none"> - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| ID | USC004 |
| Nama | CUD Barang |

| | |
|--|--|
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | <i>User</i> telah berkurang |
| SKENARIO DELETE | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |
| | 2. Sistem menampilkan halaman data barang dengan menu : - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| 3. Klik <i>delete</i> pada data yang ingin dihapus | |
| | 4. Sistem menampilkan <i>popup</i> ‘Apakah anda yakin ingin menghapus?’ |
| 5. Klik OK | |
| | 6. Sistem menampilkan halaman data barang dengan menu : - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
| Skenario Alternatif Data Batal Dihapus | |
| 5. Klik <i>Cancel</i> | |

| | |
|--|--|
| | <p>7. Sistem menampilkan halaman data barang dengan menu :</p> <ul style="list-style-type: none"> - Data barang - Tambah Barang - Edit Barang - <i>Delete</i> Barang |
|--|--|

| | |
|--------------|---------------------------------------|
| ID | USC005 |
| Nama | Read Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Menampilkan Halaman Data Barang |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|--------------------------|--|
| Skenario Normal | |
| 1. Klik Menu Data Barang | <p>2. Sistem menampilkan halaman data barang dengan menu :</p> <ul style="list-style-type: none"> - Data Barang - Tambah Barang - Edit Barang - Hapus Barang |

| | |
|------------|--|
| ID | USC005 |
| Nama | Read Barang |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |

| | |
|--------------|---------------------------------|
| Post Kondisi | Menampilkan Halaman Data Barang |
|--------------|---------------------------------|

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|------------|---------------|
|------------|---------------|

Skenario Normal

| | |
|--------------------------|--|
| 1. Klik Menu Data Barang | |
| | <p>2. Sistem menampilkan halaman data barang dengan menu :</p> <ul style="list-style-type: none"> - Data Barang - Tambah Barang - Edit Barang - Hapus Barang |

| | |
|--------------|---|
| ID | USC005 |
| Nama | Read Barang |
| Aktor | Manajer. |
| Prekondisi | Menampilkan Halaman <i>home</i> Manajer |
| Post Kondisi | Menampilkan Halaman Data Barang |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|------------|---------------|
|------------|---------------|

Skenario Normal

| | |
|--------------------------|--|
| 1. Klik Menu Data Barang | |
| | <p>2. Sistem menampilkan halaman data barang dengan menu :</p> <ul style="list-style-type: none"> - Data Barang |

4.3.4 Skenario CRUD Kategori Barang

| | |
|----|--------|
| ID | USC006 |
|----|--------|

| | |
|------------------------------|---|
| Nama | CUD Kategori Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Kategori barang telah bertambah |
| SKENARIO INPUT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Kategori Barang | |
| | 2. Sistem menampilkan halaman kategori barang dengan menu : - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| 3. Klik menu Tambah Kategori | |
| | 4. Sistem menampilkan <i>form</i> tambah kategori barang |
| 5. Isi <i>Form</i> | |
| 6. Klik Simpan | |
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil ditambahkan” |
| | 8. Sistem menampilkan halaman kategori barang dengan menu : - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |

| Skenario Alternatif Data Tidak Lengkap | |
|--|---|
| | 7. Sistem menampilkan pesan <i>"Please fill out this field"</i> |
| ID | USC006 |
| Nama | CUD Kategori Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Kategori barang telah bertambah |
| SKENARIO INPUT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Kategori Barang | 2. Sistem menampilkan halaman kategori barang dengan menu : - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| 3. Klik menu Tambah Kategori | 4. Sistem menampilkan <i>form</i> tambah kategori barang |
| 5. Isi <i>Form</i> | |
| 6. Klik Simpan | |
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> "Data berhasil ditambahkan" |

| | |
|--|--|
| | 8. Sistem menampilkan halaman kategori barang dengan menu : <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan " <i>Please fill out this field</i> " |
| ID | USC006 |
| Nama | CUD Kategori Barang |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Kategori barang telah bertambah |
| SKENARIO EDIT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario normal | |
| 1. Klik Menu Kategori Barang | |
| | 2. Sistem menampilkan halaman kategori barang dengan menu : <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| 3. Klik edit pada kategori yang ingin diedit | |

| | |
|--|---|
| | 4. Sistem menampilkan <i>form</i> yang berisi detail kategori barang |
| 5. Edit <i>Form</i> | |
| 6. Klik Simpan | |
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil diedit” |
| | 8. Sistem menampilkan halaman kategori barang dengan menu : <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan “ <i>Please fill out this field</i> ” |
| ID | USC006 |
| Nama | CUD Kategori Barang |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | <i>User</i> telah berkurang |
| SKENARIO DELETE | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Kategori Barang | |
| | 2. Sistem menampilkan halaman kategori barang dengan menu : |

| | |
|--|---|
| | <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| 3. Klik <i>delete</i> pada data yang ingin dihapus | |
| | 4. Sistem menampilkan <i>popup</i> ‘Apakah anda yakin ingin menghapus?’ |
| 5. Klik OK | |
| | 6. Data terhapus dan Sistem menampilkan halaman kategori dengan menu : <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| Skenario Alternatif Data Batal Dihapus | |
| 5. Klik <i>Cancel</i> | |
| | 6. Sistem menampilkan halaman kategori barang dengan menu : <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| ID | USC006 |
| Nama | CUD Kategori Barang |
| Aktor | Admin. |

| | |
|---|--|
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | <i>User</i> telah berkurang |
| SKENARIO DELETE | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Kategori Barang | |
| | 2. Sistem menampilkan halaman kategori barang dengan menu : - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| 3. Klik delete pada data yang ingin dihapus | |
| | 4. Sistem menampilkan <i>popup</i> ‘Apakah anda yakin ingin menghapus?’ |
| 5. Klik OK | |
| | 6. Data terhapus dan Sistem menampilkan halaman kategori dengan menu : - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| Skenario Alternatif Data Batal Dihapus | |
| 5. Klik <i>Cancel</i> | |
| | 6. Sistem menampilkan halaman kategori barang dengan menu : |

| | |
|--|--|
| | <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
|--|--|

| | |
|--------------|--|
| ID | USC007 |
| Nama | Read Kategori Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Menampilkan Halaman Data Kategori |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|--------------------------|--|
| Skenario Normal | |
| 1. Klik Menu Data Barang | 2. Sistem menampilkan kategori barang dengan menu : <ul style="list-style-type: none"> - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |

| | |
|--------------|---|
| ID | USC007 |
| Nama | Read Kategori Barang |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Menampilkan Halaman Data Kategori |

| SKENARIO VIEW | |
|--------------------------|---|
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |
| | 2. Sistem menampilkan halaman kategori barang dengan menu : - Data Kategori Barang - Tambah Kategori - Edit Kategori - <i>Delete</i> Kategori |
| ID | USC007 |
| Nama | Read Kategori Barang |
| Aktor | Manajer. |
| Prekondisi | Menampilkan Halaman <i>home</i> Manajer |
| Post Kondisi | Menampilkan Halaman Data Kategori |

| SKENARIO VIEW | |
|--------------------------|---|
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |
| | 2. Sistem menampilkan halaman data kategori dengan menu : - Data Kategori Barang |

4.3.5 Skenario CRUD Data Ruangan

| | |
|-------|-------------------------|
| ID | USC008 |
| Nama | CUD Data Ruangan |
| Aktor | Admin. |

| | |
|--|--|
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Ruangan telah bertambah |
| SKENARIO INPUT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Ruangan | |
| | 2. Sistem menampilkan halaman ruangan dengan menu : - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| 3. Klik menu Tambah Ruangan | |
| | 4. Sistem menampilkan <i>form</i> tambah ruangan |
| 5. Isi <i>Form</i> | |
| 6. Klik Simpan | |
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil ditambahkan” |
| | 8. Sistem menampilkan halaman ruangan dengan menu : - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| Skenario Alternatif Data Tidak Lengkap | |

| | |
|-----------------------------|--|
| | 7. Sistem menampilkan pesan “Please fill out this field” |
| ID | USC007 |
| Nama | CUD Ruangan |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Ruangan telah bertambah |
| SKENARIO INPUT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Ruangan | 2. Sistem menampilkan halaman ruangan dengan menu : - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| 3. Klik menu Tambah Ruangan | 4. Sistem menampilkan <i>form</i> tambah ruangan |
| 5. Isi <i>Form</i> | |
| 6. Klik Simpan | |
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil ditambahkan” |
| | 8. Sistem menampilkan halaman ruangan dengan menu : |

| | |
|---|---|
| | <ul style="list-style-type: none"> - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan "Please fill out this field" |
| ID | USC007 |
| Nama | CUD Ruangan |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Ruangan telah teredit |
| SKENARIO EDIT | |
| Aksi Aktor | Reaksi Sistem |
| Skenario normal | |
| 1. Klik Menu Kategori Barang | |
| | 2. Sistem menampilkan halaman ruangan dengan menu : <ul style="list-style-type: none"> - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| 3. Klik edit pada ruangan yang ingin diedit | |
| | 4. Sistem menampilkan <i>form</i> yang berisi detail ruangan |
| 5. Edit <i>Form</i> | |
| 6. Klik Simpan | |

| | |
|--|--|
| | 7. Sistem menambahkan ke <i>database</i> dan Menampilkan <i>popup</i> “Data berhasil diedit” |
| | 8. Sistem menampilkan halaman ruangan dengan menu : <ul style="list-style-type: none"> - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| Skenario Alternatif Data Tidak Lengkap | |
| | 7. Sistem menampilkan pesan “ <i>Please fill out this field</i> ” |
| ID | USC007 |
| Nama | CUD Ruangan |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Ruangan telah berkurang |
| SKENARIO DELETE | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Ruangan | |
| | 2. Sistem menampilkan halaman ruangan dengan menu : <ul style="list-style-type: none"> - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |

| | |
|---|--|
| 3. Klik delete pada data yang ingin dihapus | |
| | 4. Sistem menampilkan <i>popup</i> ‘Apakah anda yakin ingin menghapus?’ |
| 5. Klik OK | |
| | 6. Data terhapus dan Sistem menampilkan halaman ruangan dengan menu : <ul style="list-style-type: none"> - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| Skenario Alternatif Data Batal Dihapus | |
| 5. Klik <i>Cancel</i> | |
| | 6. Sistem menampilkan halaman ruangan dengan menu : <ul style="list-style-type: none"> - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| ID | USC006 |
| Nama | CUD Kategori Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Ruangan telah berkurang |
| SKENARIO DELETE | |
| Aksi Aktor | Reaksi Sistem |

| Skenario Normal | |
|---|--|
| 1. Klik Menu Kategori Barang | 2. Sistem menampilkan halaman ruangan dengan menu : - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| 3. Klik delete pada data yang ingin dihapus | 4. Sistem menampilkan <i>popup</i> ‘Apakah anda yakin ingin menghapus?’ |
| 5. Klik OK | 6. Data terhapus dan Sistem menampilkan halaman ruangan dengan menu : - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
| Skenario Alternatif Data Batal Dihapus | |
| 5. Klik <i>Cancel</i> | 6. Sistem menampilkan halaman ruangan dengan menu : - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |

| | |
|--------------|--|
| ID | USC007 |
| Nama | Read Ruangan |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Menampilkan Halaman Data ruangan |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|--------------------------|--|
| Skenario Normal | |
| 1. Klik Menu Data Barang | 2. Sistem menampilkan halaman ruangan dengan menu : - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |

| | |
|--------------|---|
| ID | USC007 |
| Nama | Read Ruangan |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Menampilkan Halaman Data Ruangan |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|--------------------------|---------------|
| Skenario Normal | |
| 1. Klik Menu Data Barang | |

| | |
|--|---|
| | 2. Sistem menampilkan halaman ruangan dengan menu : <ul style="list-style-type: none"> - Data Ruangan - Tambah Ruangan - Edit Ruangan - <i>Delete</i> Ruangan |
|--|---|

| | |
|--------------|---|
| ID | USC008 |
| Nama | Read Kategori Barang |
| Aktor | Manajer. |
| Prekondisi | Menampilkan Halaman <i>home</i> Manajer |
| Post Kondisi | Menampilkan Halaman Data Ruangan |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|--------------------------|---|
| Skenario Normal | |
| 1. Klik Menu Data Barang | 2. Sistem menampilkan halaman data ruangan dengan menu : <ul style="list-style-type: none"> - Data Kategori Barang |

4.3.6 Skenario Cari Data Barang

| | |
|--------------|---------------------------------------|
| ID | USC009 |
| Nama | Cari Data Barang |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Menampilkan Halaman Data Barang |

| Aksi Aktor | Reaksi Sistem |
|---------------------------|---|
| Skenario Normal | |
| 1. Klik Menu Data Barang | |
| | <p>2. Sistem menampilkan halaman data penggunaan barang dengan pilihan pencarian :</p> <ul style="list-style-type: none"> - Berdasarkan ruangan - Berdasarkan kategori barang <p>Menampilkan pilihan dari</p> |
| 3. Pilih tipe pencarian | |
| | 4. Menampilkan pilihan sesuai tipe pencarian |
| 5. Pilih detail pencarian | |
| 6. Klik Cari | |
| | 7. Sistem menampilkan data inventaris sesuai pencarian |
| ID | USC009 |
| Nama | Cari Data Barang |
| Aktor | Operator. |
| Prekondisi | Menampilkan Halaman <i>home</i> Operator |
| Post Kondisi | Menampilkan Halaman Data Barang |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Data Barang | |

| | |
|---------------------------|---|
| | <p>2. Sistem menampilkan halaman data penggunaan barang dengan pilihan pencarian :</p> <ul style="list-style-type: none"> - Berdasarkan ruangan - Berdasarkan kategori barang <p>Menampilkan pilihan dari</p> |
| 3. Pilih tipe pencarian | |
| | 4. Menampilkan pilihan sesuai tipe pencarian |
| 5. Pilih detail pencarian | |
| 6. Klik Cari | |
| | 7. Sistem menampilkan data inventaris sesuai pencarian |

4.3.7 Skenario Cetak Laporan

| | |
|--------------|--|
| ID | USC005 |
| Nama | Cetak Laporan |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Mencetak rekapan data inventaris dalam tipe file <i>.xls</i> |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|----------------------|---------------|
| Skenario Normal | |
| 1. Klik Menu Laporan | |

| | |
|---------------|--|
| | 2. Sistem menampilkan halaman laporan dengan menu : - Cetak |
| 3. Klik Cetak | |
| | 4. Sistem mengunduh file |

| | |
|--------------|--|
| ID | USC005 |
| Nama | Cetak Laporan |
| Aktor | Admin. |
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Mencetak rekapan data inventaris dalam tipe file <i>.xls</i> |

SKENARIO VIEW

| Aksi Aktor | Reaksi Sistem |
|----------------------|--|
| Skenario Normal | |
| 1. Klik Menu Laporan | |
| | 2. Sistem menampilkan halaman laporan dengan menu : - Cetak |
| 3. Klik Cetak | |
| | 4. Sistem mengunduh file |

| | |
|-------|----------------------|
| ID | USC005 |
| Nama | Cetak Laporan |
| Aktor | Admin. |

| | |
|----------------------|---|
| Prekondisi | Menampilkan Halaman <i>home</i> Admin |
| Post Kondisi | Mencetak rekapan data inventaris dalam tipe file <i>.xls</i> |
| SKENARIO VIEW | |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal | |
| 1. Klik Menu Laporan | |
| | 2. Sistem menampilkan halaman laporan dengan menu : - Cetak |
| 3. Klik Cetak | |
| | 4. Sistem mengunduh file |

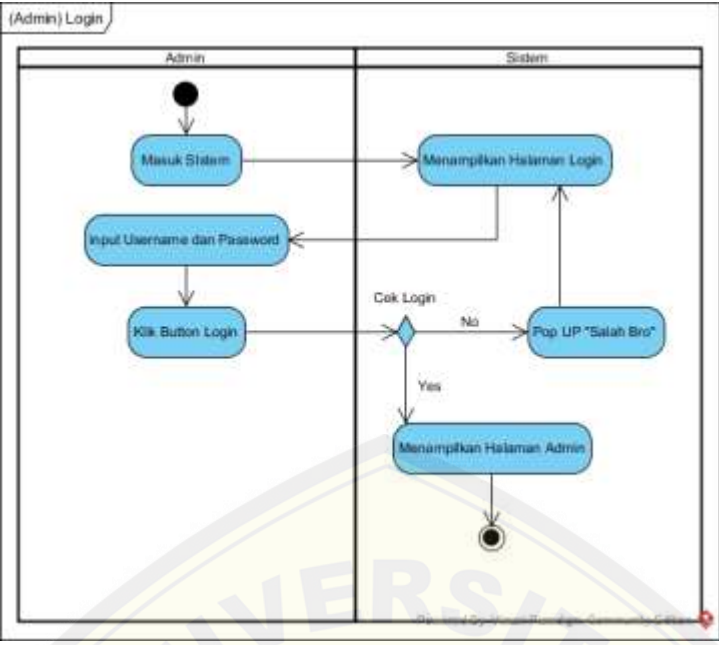
4.4 *Activity Diagram*

Activity Diagram menggambarkan aliran aktivitas dalam Sistem Informasi Manajemen Inventarisasi Barang yang akan dibangun.

4.4.1 *Activity Diagram Admin*

4.4.1.1 *Login*

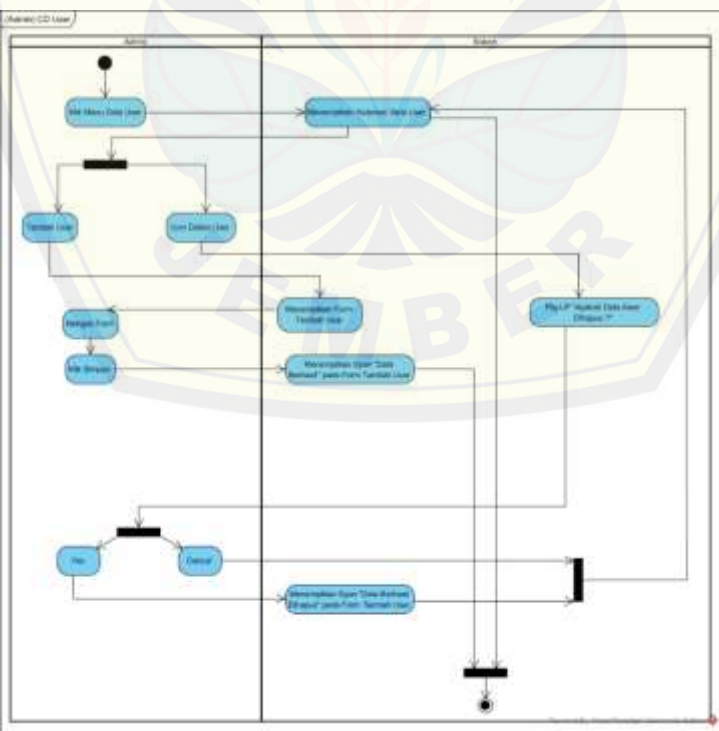
Penjelasan aliran aktifitas dalam fitur *Login* dijelaskan gambar 4.3.



Gambar 4. 3 Activity Diagram Admin Login
Sumber: Hasil Penelitian, 2015

4.4.1.2 CD User

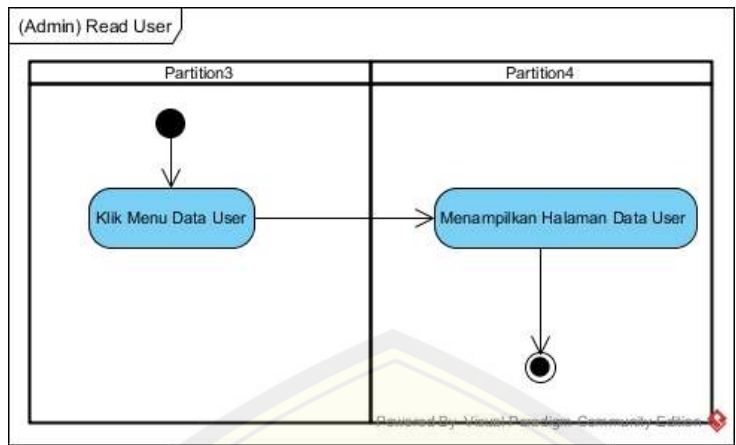
Penjelasan aliran aktifitas dalam fitur CD User dijelaskan gambar 4.4.



Gambar 4. 4 Activity Diagram Admin CD User
Sumber: Hasil Penelitian, 2015

4.4.1.3 Read User

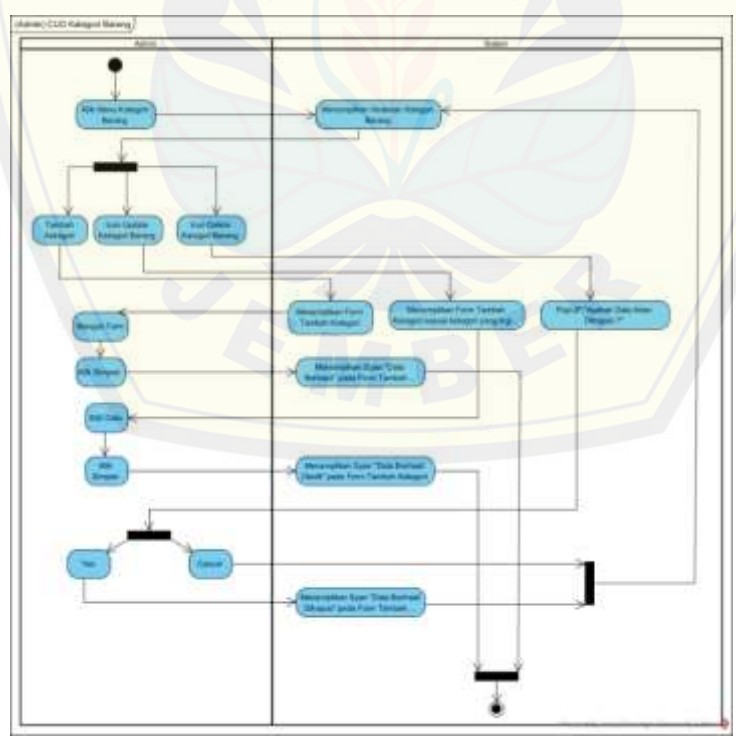
Penjelasan aliran aktifitas dalam fitur *Read User* dijelaskan gambar 4.5.



Gambar 4. 5 Activity Diagram Admin Read User
Sumber: Hasil Penelitian, 2015

4.4.1.4 CUD Kategori Barang

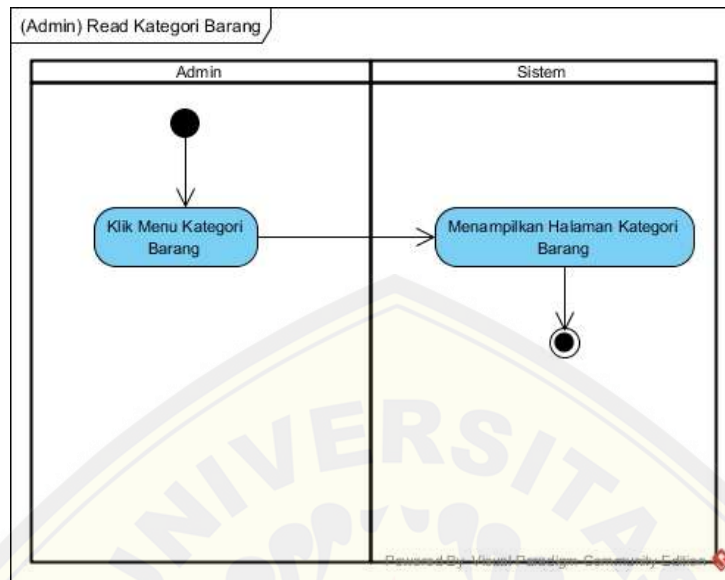
Penjelasan aliran aktifitas dalam fitur CUD Kategori Barang dijelaskan gambar 4.6.



Gambar 4. 6 Activity Diagram Admin CUD Kategori Barang
Sumber: Hasil Penelitian, 2015

4.4.1.5 *Read* Kategori Barang

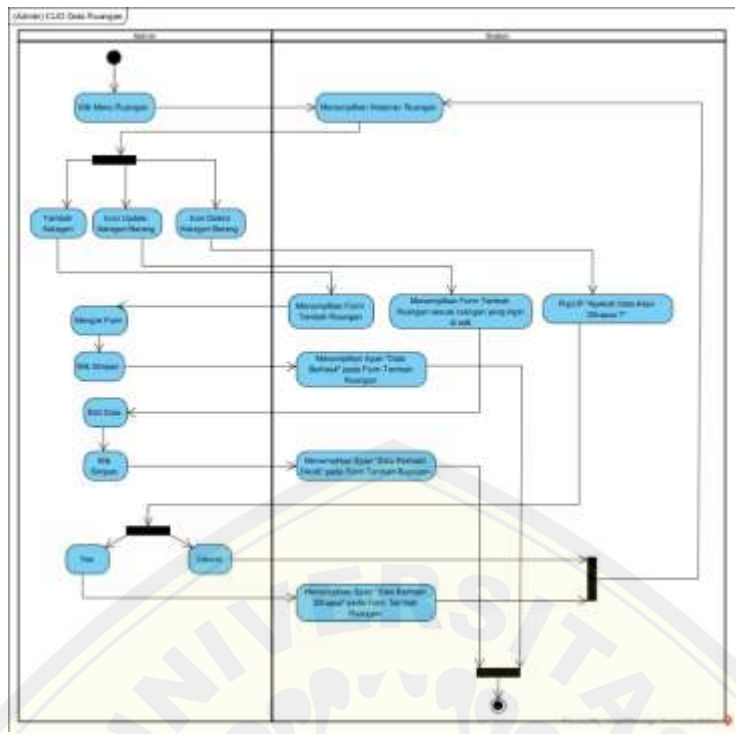
Penjelasan aliran aktifitas dalam fitur *Read* Kategori Barang dijelaskan gambar 4.7.



Gambar 4.7 *Activity Diagram* Admin *Read* Kategori Barang
Sumber: Hasil Penelitian, 2015

4.4.1.6 CUD Data Ruangan

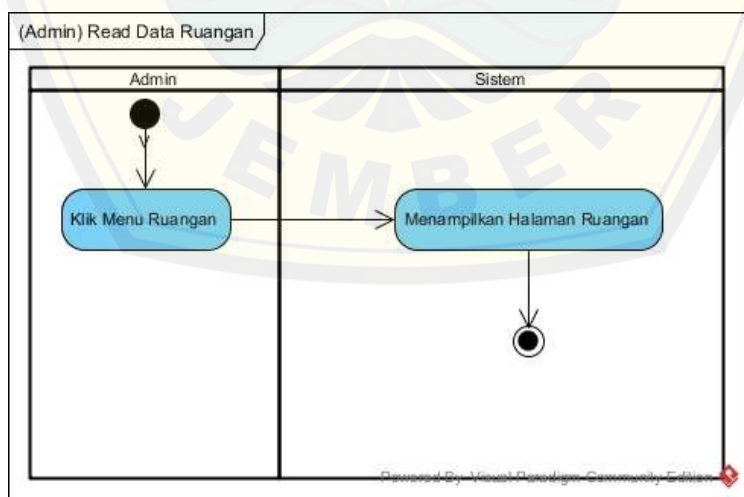
Penjelasan aliran aktifitas dalam fitur CUD Data Ruangan dijelaskan gambar 4.8.



Gambar 4. 8 Activity Diagram Admin Data Ruangan
Sumber: Hasil Penelitian, 2015

4.4.1.7 Read Data Ruangan

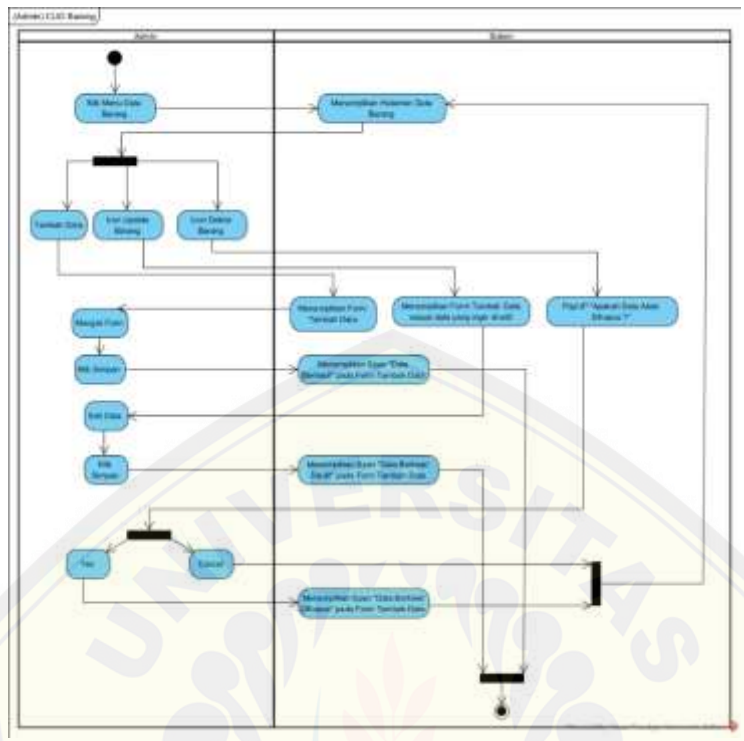
Penjelasan aliran aktifitas dalam fitur Read Data Ruangan dijelaskan gambar 4.9.



Gambar 4. 9 Activity Diagram Admin Read Data Ruangan
Sumber: Hasil Penelitian, 2015

4.4.1.8 CUD Barang

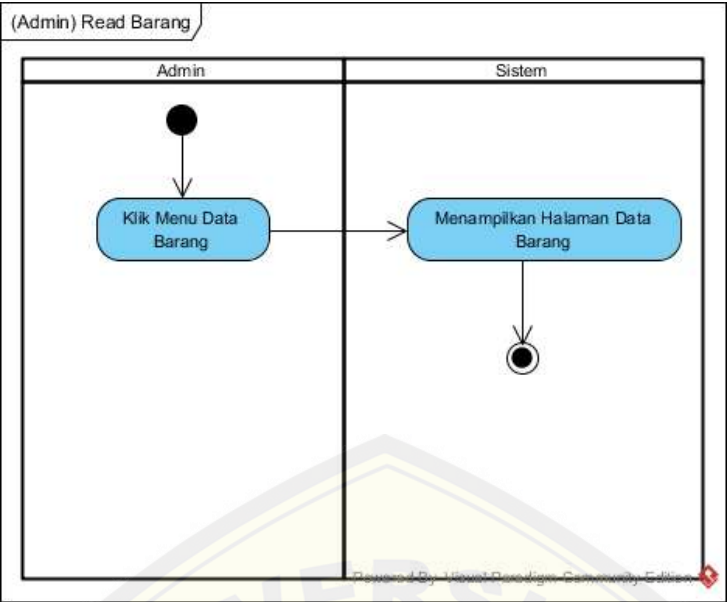
Penjelasan aliran aktifitas dalam fitur CUD Barang dijelaskan gambar 4.10.



Gambar 4. 10 Activity Diagram Admin CUD Barang
Sumber: Hasil Penelitian, 2015

4.4.1.9 Read Barang

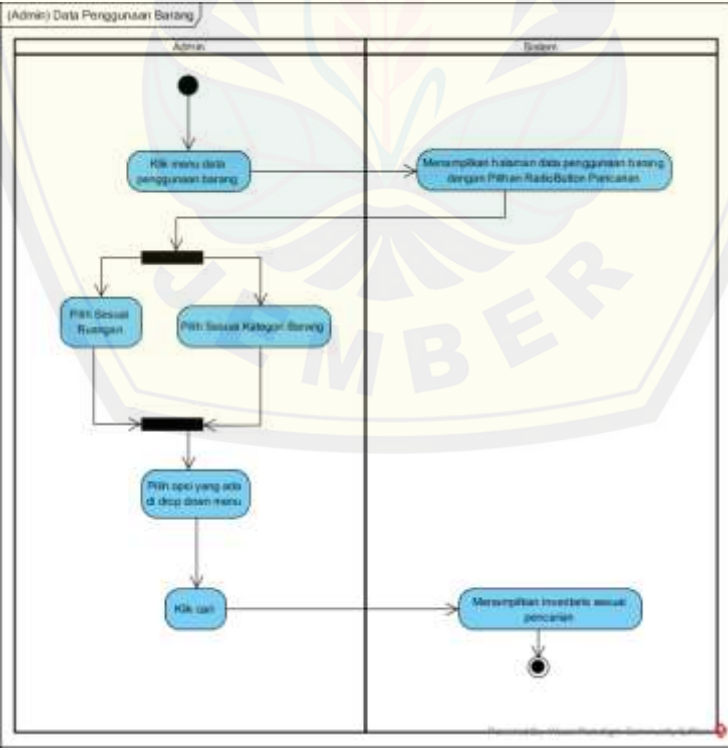
Penjelasan aliran aktifitas dalam fitur Read Barang dijelaskan gambar 4.11.



Gambar 4. 11 Activity Diagram Admin Read Barang (Sumber: Hasil Penelitian, 2015)

4.4.1.10 Data Penggunaan Barang

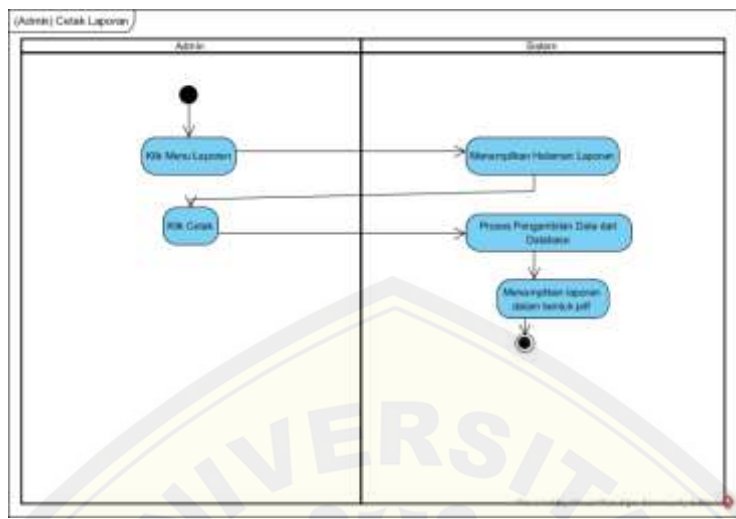
Penjelasan aliran aktifitas dalam fitur Penggunaan Data Barang dijelaskan gambar 4.12.



Gambar 4. 12 Activity Diagram Admin Data Penggunaan Barang Sumber: Hasil Penelitian, 2015

4.4.1.11 Cetak Laporan

Penjelasan aliran aktifitas dalam fitur Cetak Laporan dijelaskan gambar 4.13.

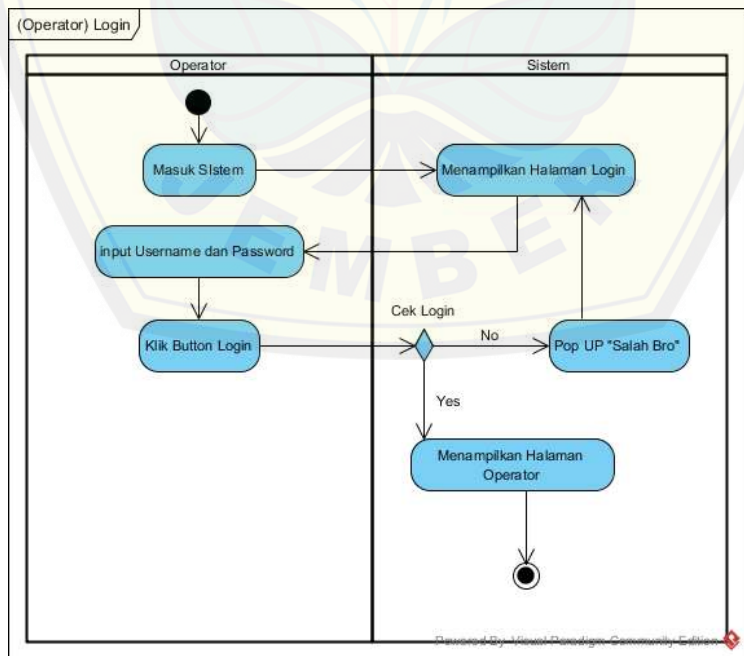


Gambar 4. 13 Acitivity Diagram Admin Cetak Laporan
Sumber: Hasil Penelitian, 2015

4.4.2 Activity Diagram Operator

4.4.2.1 Login

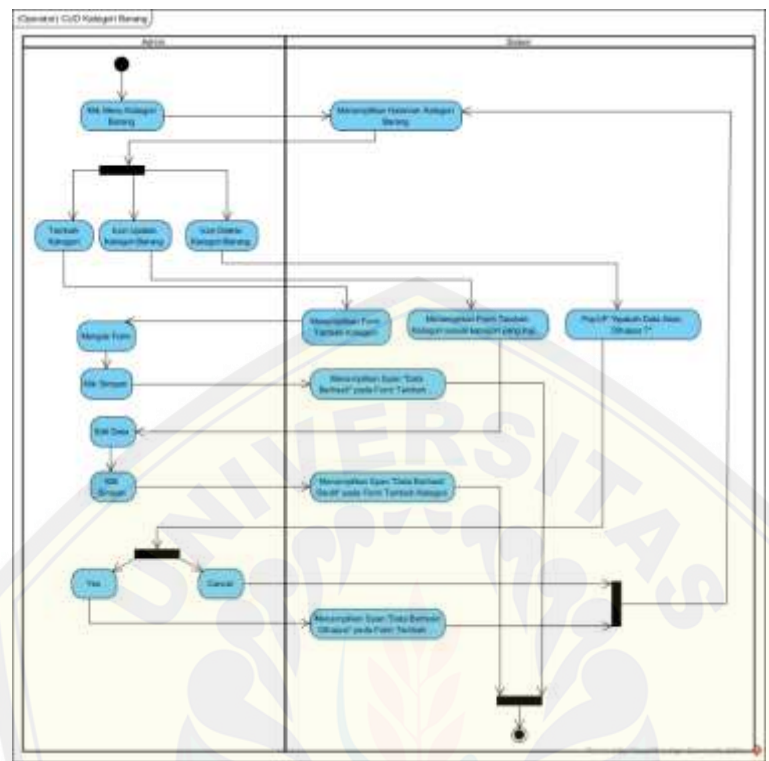
Penjelasan aliran aktifitas dalam fitur Login dijelaskan gambar 4.14.



Gambar 4. 14 Activity Diagram Operator Login
Sumber: Hasil Penelitian, 2015

4.4.2.2 CUD Kategori Barang

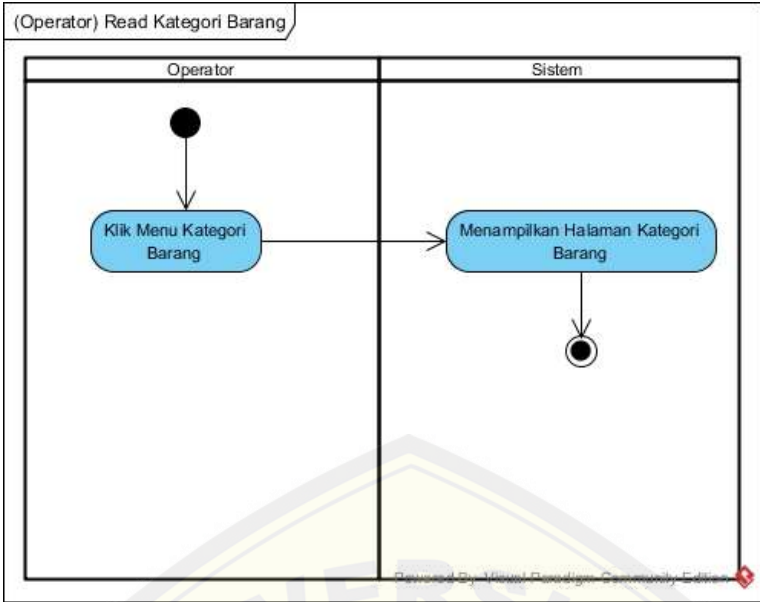
Penjelasan aliran aktifitas dalam fitur CUD Kategori Barang dijelaskan gambar 4.15.



Gambar 4. 15 Activity Diagram Operator CUD Kategori Barang
Sumber: Hasil Penelitian, 2015

4.4.2.3 Read Kategori Barang

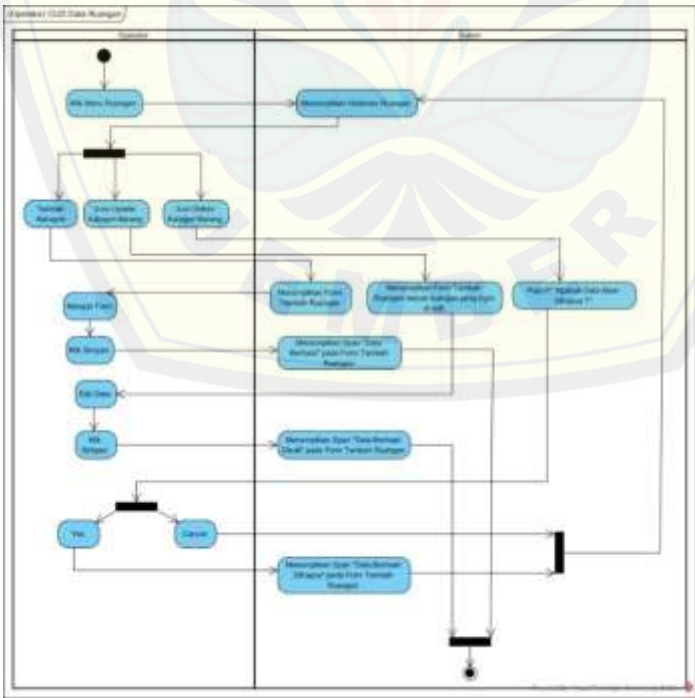
Penjelasan aliran aktifitas dalam fitur Read Kategori Barang dijelaskan gambar 4.16.



Gambar 4. 16 Activity Diagram Operator Read Kategori Barang
Sumber: Hasil Penelitian, 2015

4.4.2.4 CUD Data Ruangan

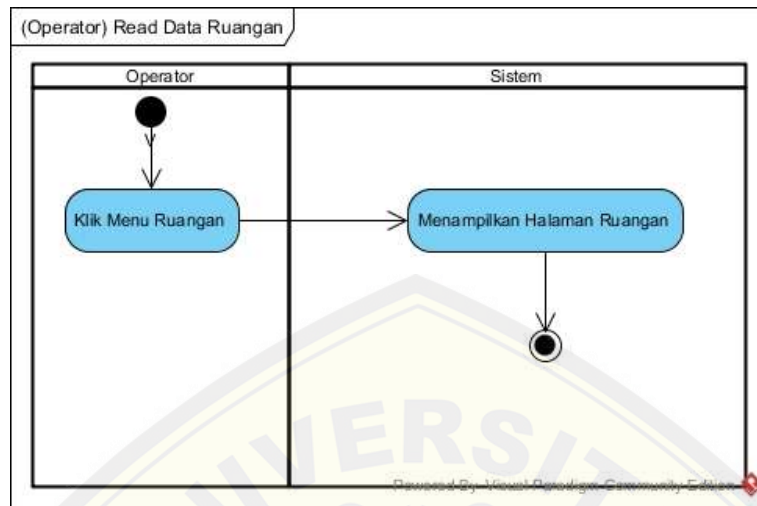
Penjelasan aliran aktifitas dalam fitur CUD Data Ruangan dijelaskan gambar 4.17.



Gambar 4. 17 Activity Diagram Operator CUD Data Ruangan
Sumber: Hasil Penelitian, 2015

4.4.2.5 Read Data Ruangan

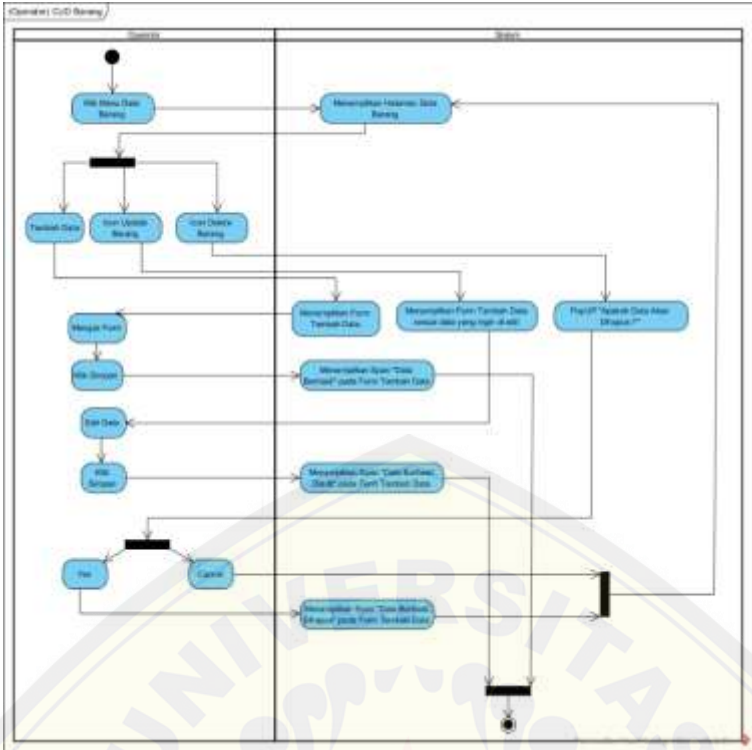
Penjelasan aliran aktifitas dalam fitur *Read Data Ruangan* dijelaskan gambar 4.18.



Gambar 4. 18 *Activity Diagram Operator Read Data Ruangan*
Sumber: Hasil Penelitian, 2015

4.4.2.6 CUD Barang

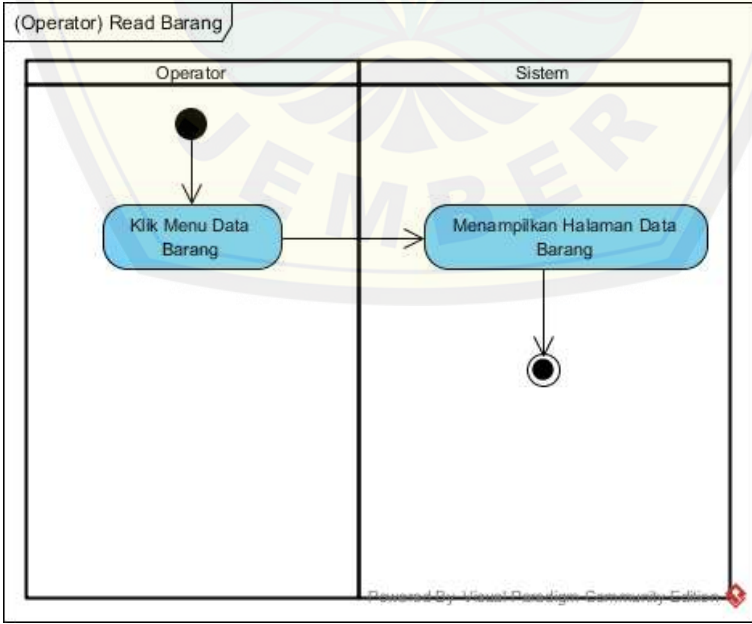
Penjelasan aliran aktifitas dalam fitur CUD Data Barang dijelaskan gambar 4.19.



Gambar 4. 19 *Activity Diagram* Operator CUD Barang
Sumber: Hasil Penelitian, 2015

4.4.2.7 *Read* Barang

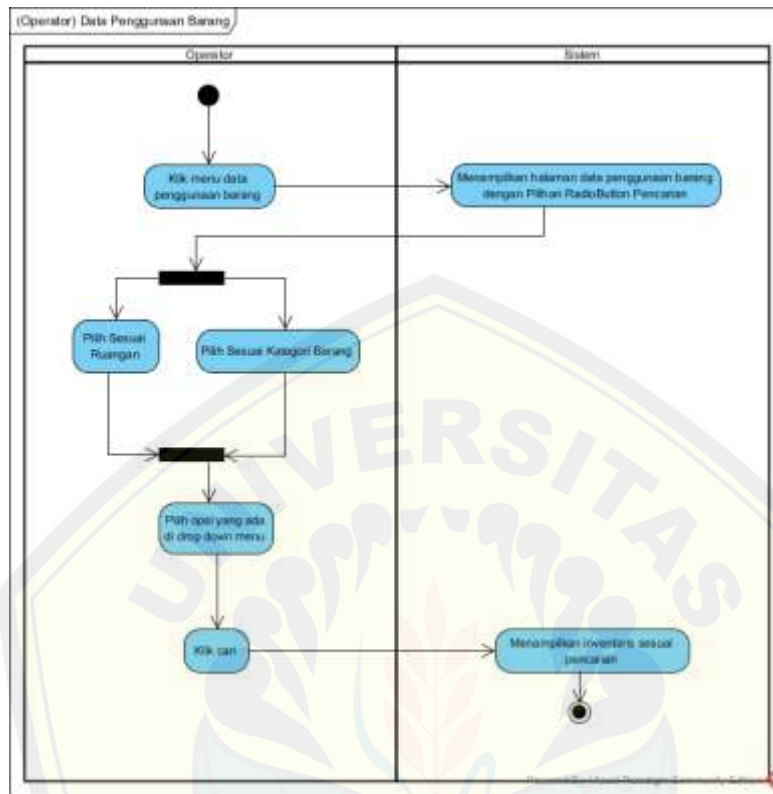
Penjelasan aliran aktifitas dalam fitur *Read* Barang dijelaskan gambar 4.20.



Gambar 4. 20 *Activity Diagram* Operator *Read* Barang
Sumber: Hasil Penelitian, 2015

4.4.2.8 Data Penggunaan Barang

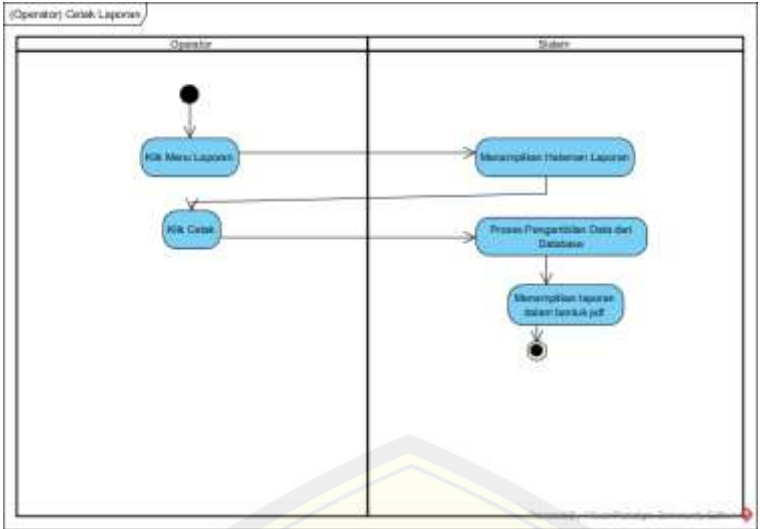
Penjelasan aliran aktifitas dalam fitur Data Penggunaan Barang dijelaskan gambar 4.21.



Gambar 4. 21 *Activity Diagram* Operator Data Penggunaan Barang
Sumber: Hasil Penelitian, 2015

4.4.2.9 Cetak Laporan

Penjelasan aliran aktifitas dalam fitur Cetak Laporan dijelaskan gambar 4.22.

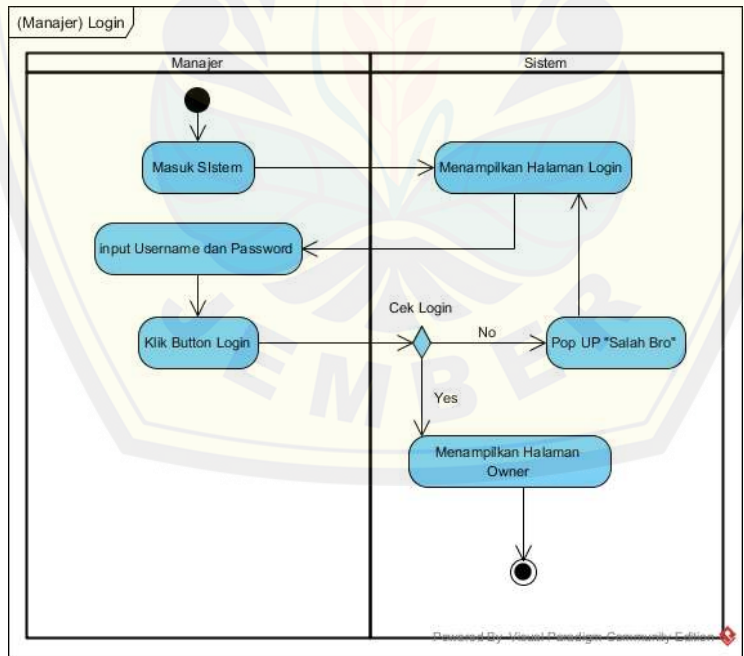


Gambar 4. 22 Activity Diagram Opertor Cetak Laporan
Sumber: Hasil Penelitian, 2015

4.4.3 Activity Diagram Manajer

4.4.3.1 Login

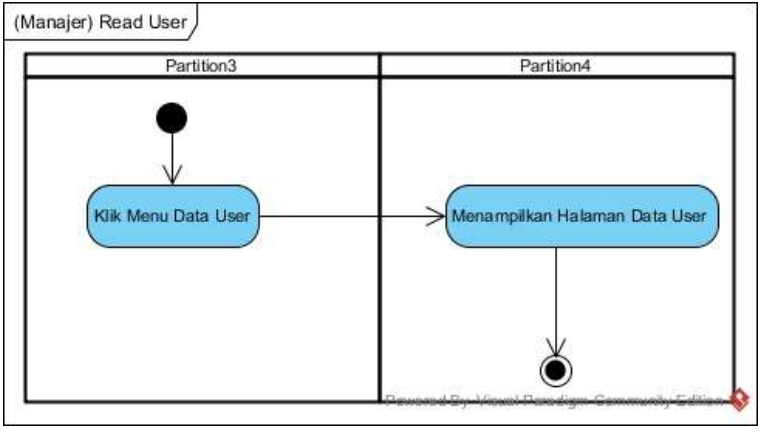
Penjelasan aliran aktifitas dalam fitur Login dijelaskan gambar 4.23.



Gambar 4. 23 Activity Diagram Manajer Login
Sumber: Hasil Penelitian, 2015

4.4.3.2 Read User

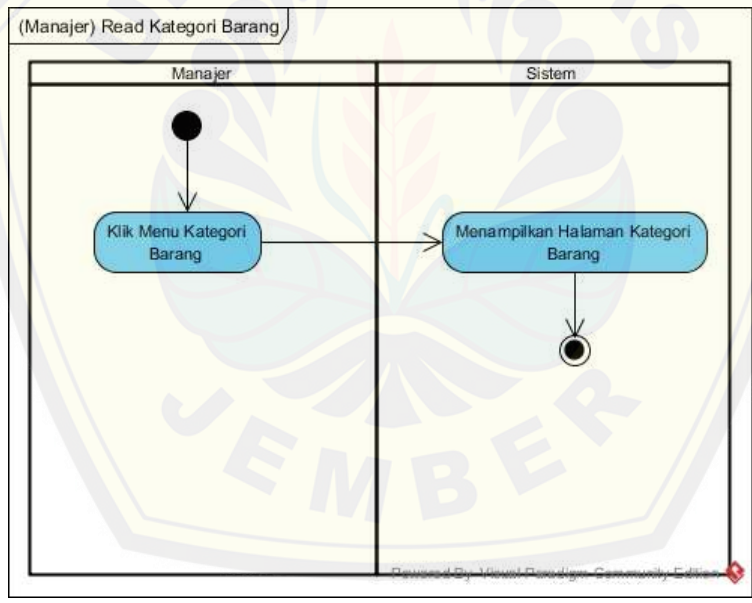
Penjelasan aliran aktifitas dalam fitur Read User dijelaskan gambar 4.24.



Gambar 4. 24 Activity Diagram Manajer Read User
Sumber: Hasil Penelitian, 2015

4.4.3.3 Read Kategori Barang

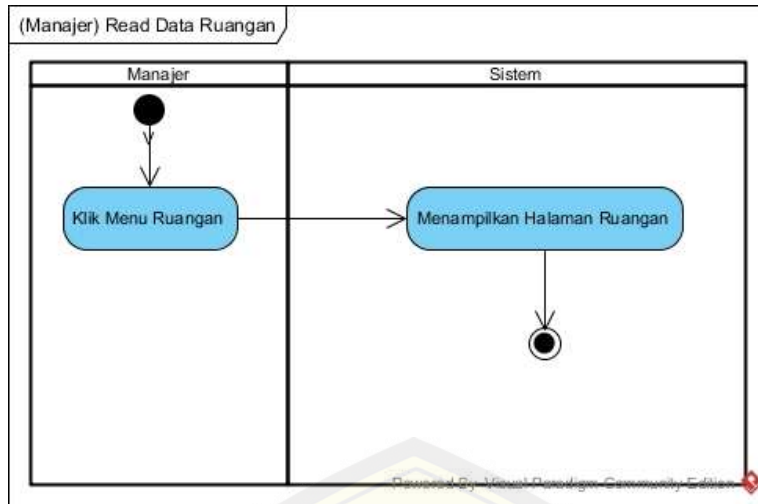
Penjelasan aliran aktifitas dalam fitur Read Kategori Barang dijelaskan gambar 4.25.



Gambar 4. 25 Activity Diagram Manajer Read Kategori Barang
Sumber: Hasil Penelitian, 2015

4.4.3.4 Read Data Ruangan

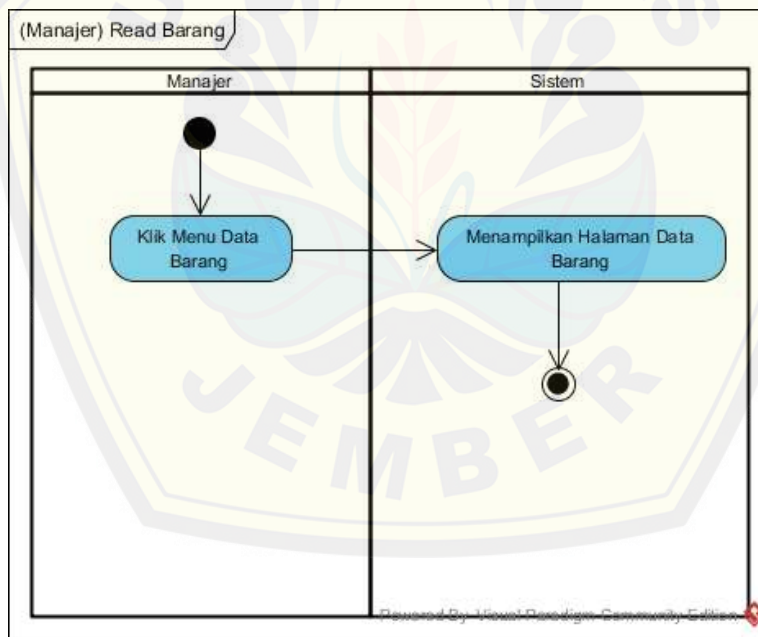
Penjelasan aliran aktifitas dalam fitur Read Data Ruangan dijelaskan gambar 4.26.



Gambar 4. 26 Activity Diagram Manajer Read Data Ruangan
 Sumber: Hasil Penelitian, 2015

4.4.3.5 Read Barang

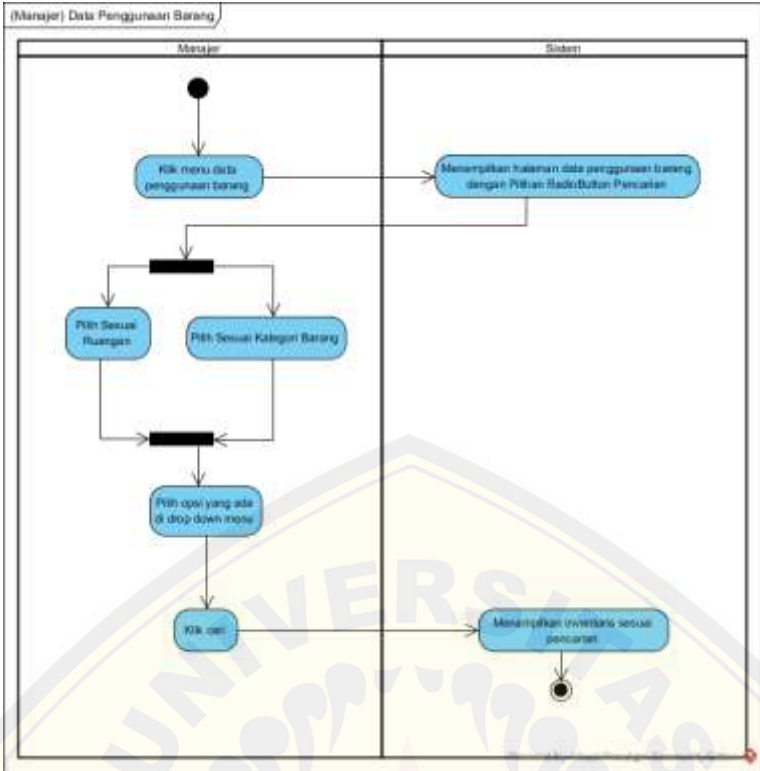
Penjelasan aliran aktifitas dalam fitur Read Barang dijelaskan gambar 4.27.



Gambar 4. 27 Activity Diagram Manajer Read Barang
 Sumber: Hasil Penelitian, 2015

4.4.3.6 Data Penggunaan Barang

Penjelasan aliran aktifitas dalam fitur Data Penggunaan Barang dijelaskan gambar 4.28.

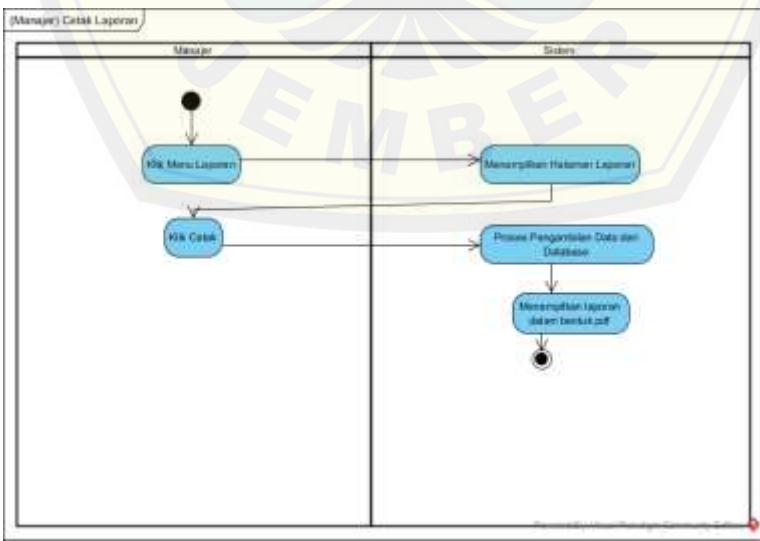


Gambar 4. 28 Activity Diagram Manajer Data Penggunaan Barang
Sumber: Hasil Penelitian, 2015

4.4.3.7 Cetak Laporan

Penjelasan aliran aktifitas dalam fitur Cetak Laporan dijelaskan gambar

4.29.



Gambar 4. 29 Activity Diagram Manajer Cetak Laporan
Sumber: Hasil Penelitian, 2015

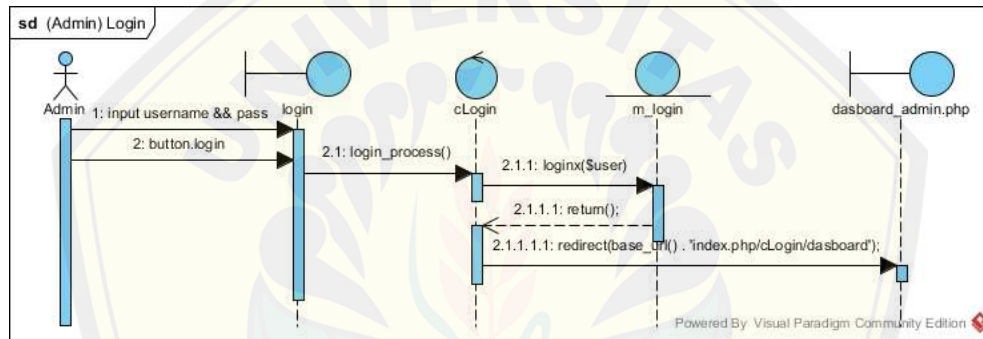
4.5 Sequence Diagram

Sequence diagram adalah dokumentasi suatu diagram terurut yang menampilkan interaksi - interaksi antar objek di dalam sistem. *Sequence diagram* digunakan untuk menggambarkan skenario dan memodelkan aliran logika dalam sistem dengan cara visual. *Sequence diagram* dari Sistem Informasi Manajemen Inventarisasi Barang adalah sebagai berikut:

4.5.1 Admin

4.5.1.1 Sequence Diagram Login

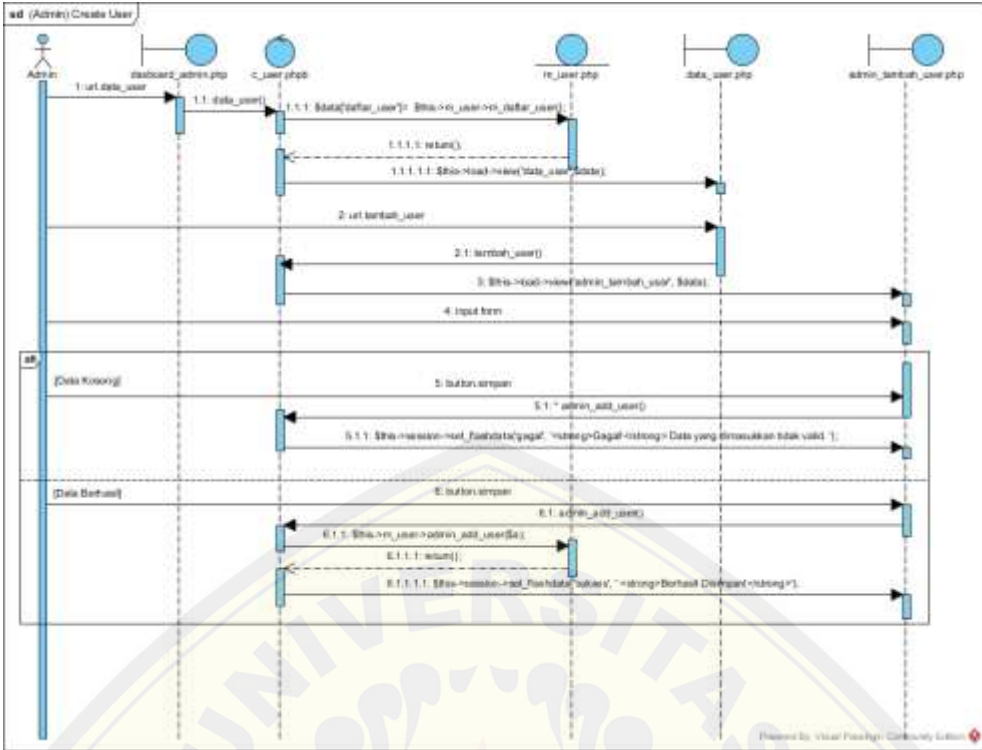
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Login*, dijelaskan pada gambar 4.30.



Gambar 4. 30 *Sequence Diagram Admin Login*
Sumber: Hasil Penelitian, 2015

4.5.1.2 Sequence Diagram Create User

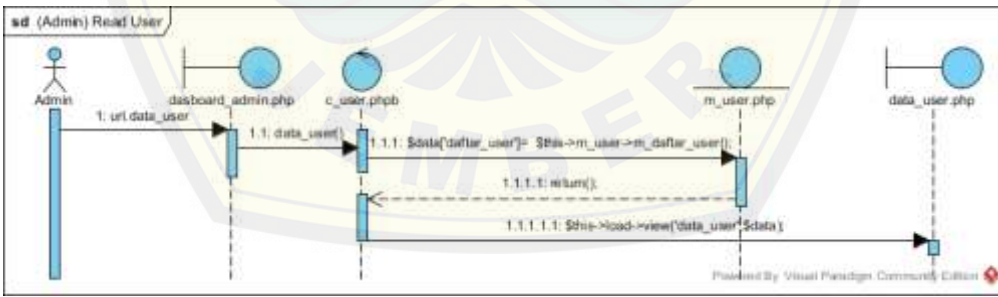
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Create User*, dijelaskan pada gambar 4.31.



Gambar 4. 31 Sequence Diagram Admin Crete User
Sumber: Hasil Penelitian, 2015

4.5.1.3 Sequence Diagram Read User

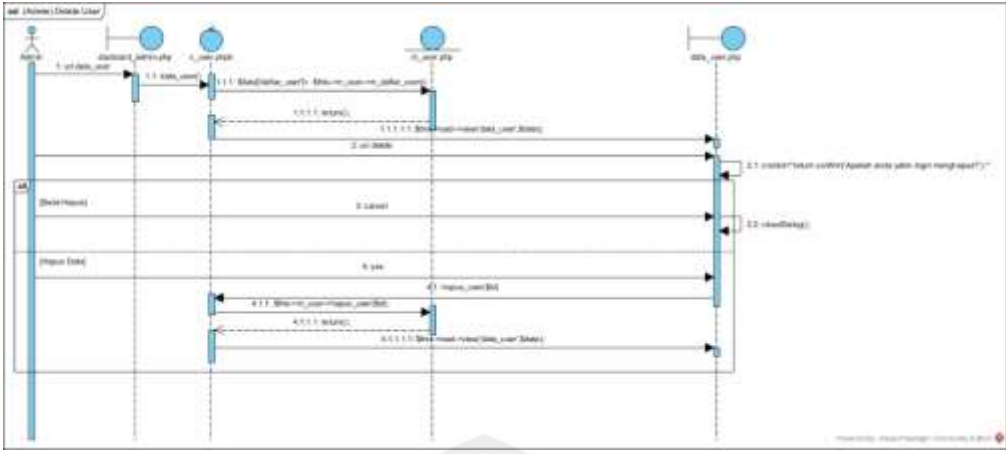
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Read User*, dijelaskan pada gambar 4.32.



Gambar 4. 32 Sequence Diagram Admin Read User
Sumber: Hasil Penelitian, 2015

4.5.1.4 Sequence Diagram Delete User

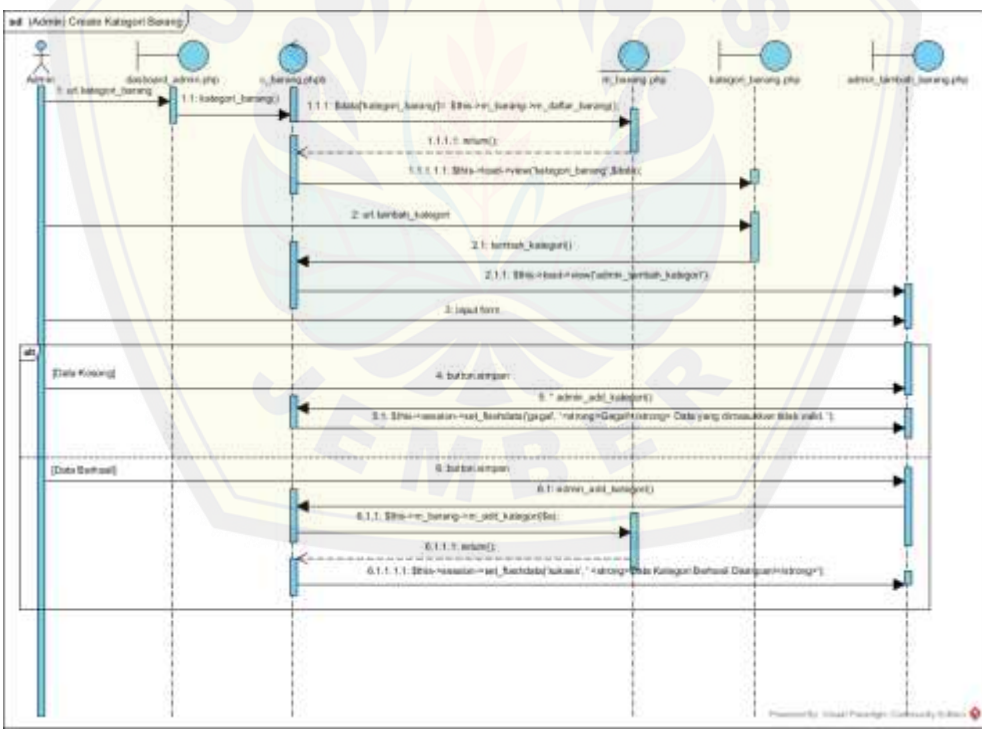
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Delete User*, dijelaskan pada gambar 4.33.



Gambar 4. 33 Sequence Diagram Admin Delete User
Sumber: Hasil Penelitian, 2015

4.5.1.5 Sequence Diagram Create Kategori Barang

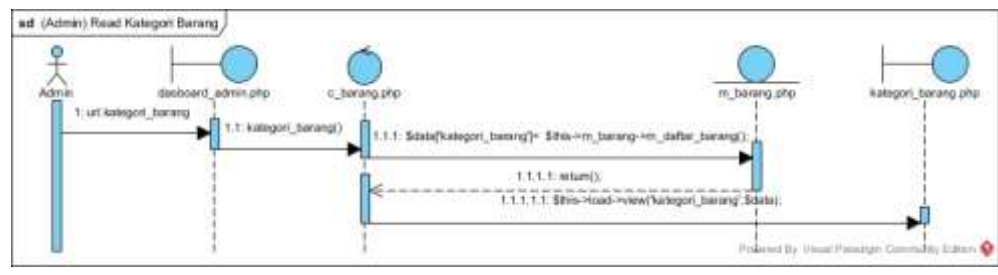
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Create Kategori Barang*, dijelaskan pada gambar 4.34.



Gambar 4. 34 Sequence Diagram Admin Create Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.1.6 Sequence Diagram Read Kategori Barang

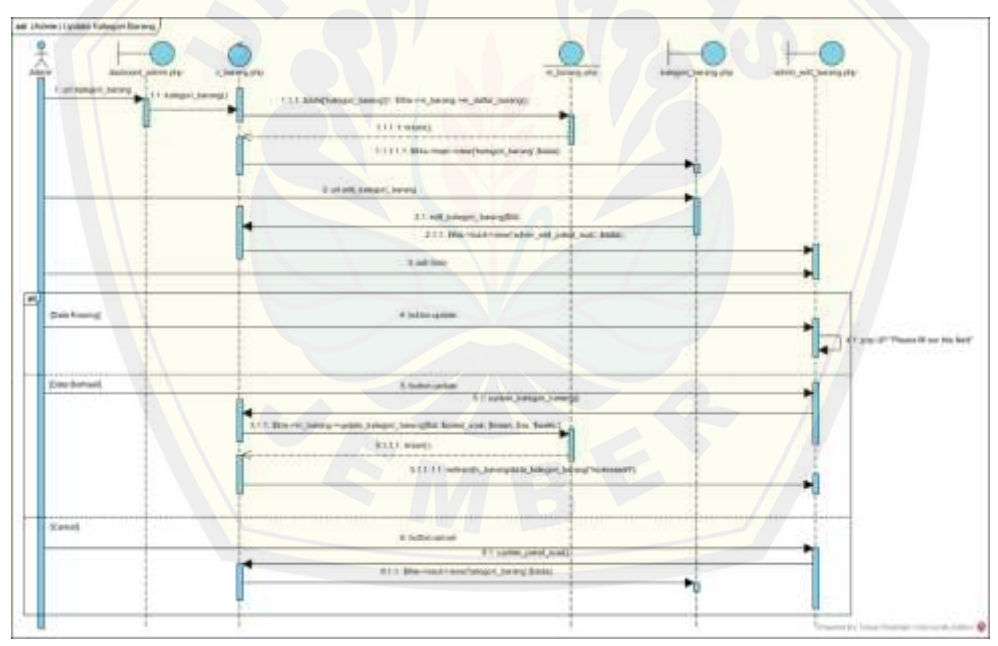
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram* Read Kategori Barang, dijelaskan pada gambar 4.35.



Gambar 4. 35 Sequence Diagram Admin Read Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.1.7 Sequence Diagram Update Kategori Barang

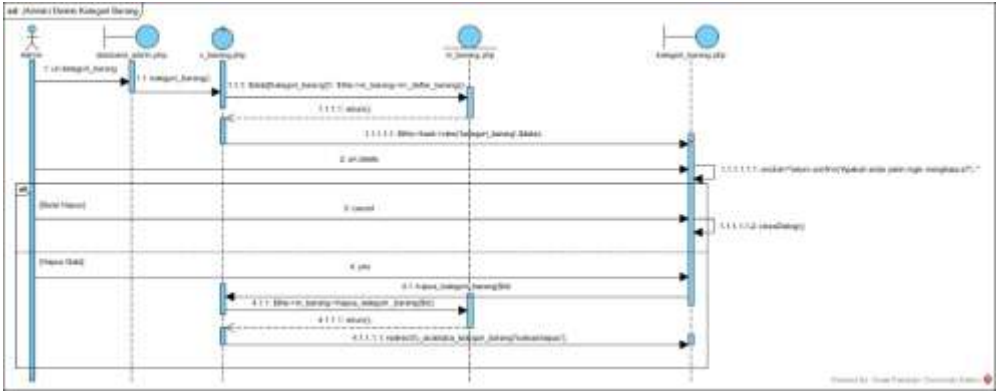
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram* Update Kategori Barang, dijelaskan pada gambar 4.36.



Gambar 4. 36 Sequence Diagram Admin Update Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.1.8 Sequence Diagram Delete Kategori Barang

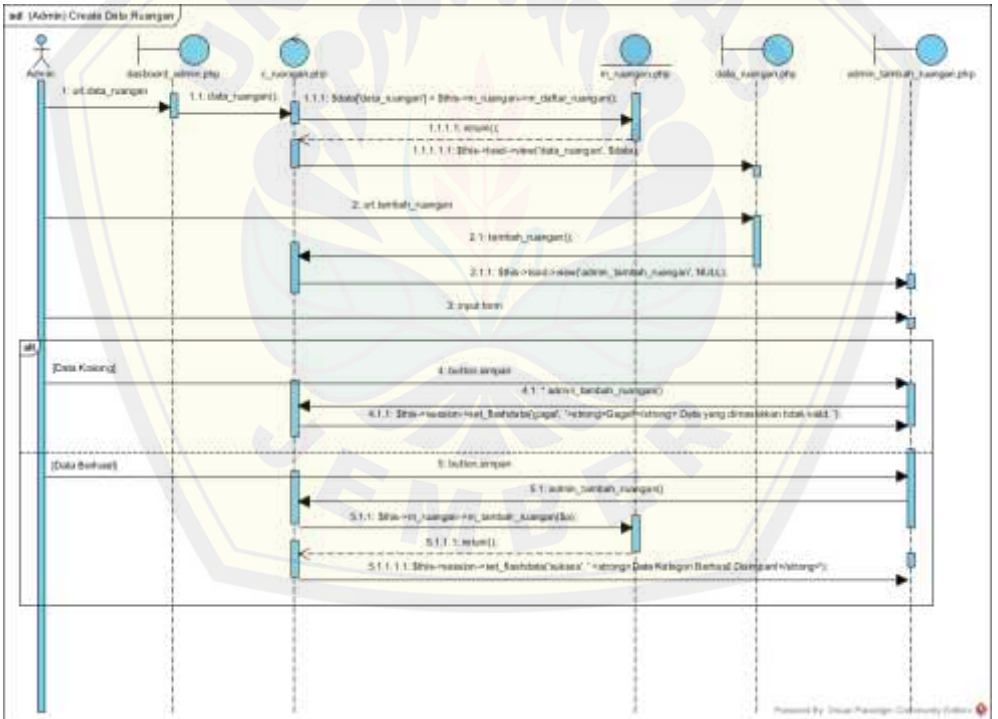
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram* Delete Kategori Barang, dijelaskan pada gambar 4.37.



Gambar 4. 37 Sequence Diagram Admin Delete Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.1.9 Sequence Diagram Create Data Ruang

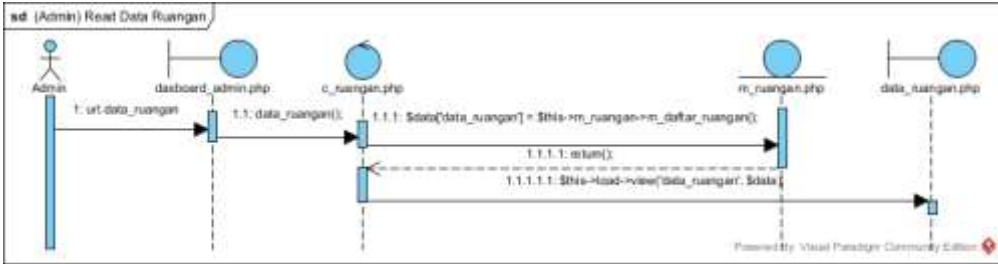
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Create Data Ruang*, dijelaskan pada gambar 4.38.



Gambar 4. 38 Sequence Diagram Admin Create Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.1.10 Sequence Diagram Read Data Ruang

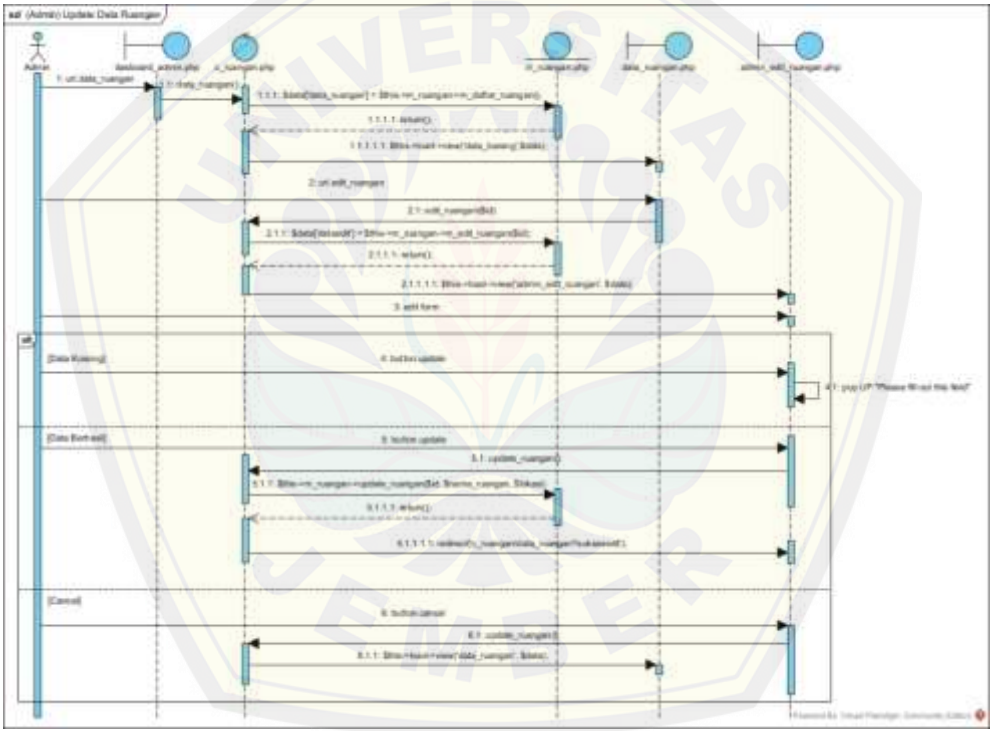
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Read Data Ruang*, dijelaskan pada gambar 4.39.



Gambar 4. 39 Sequence Diagram Admin Read Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.1.11 Sequence Diagram Update Data Ruang

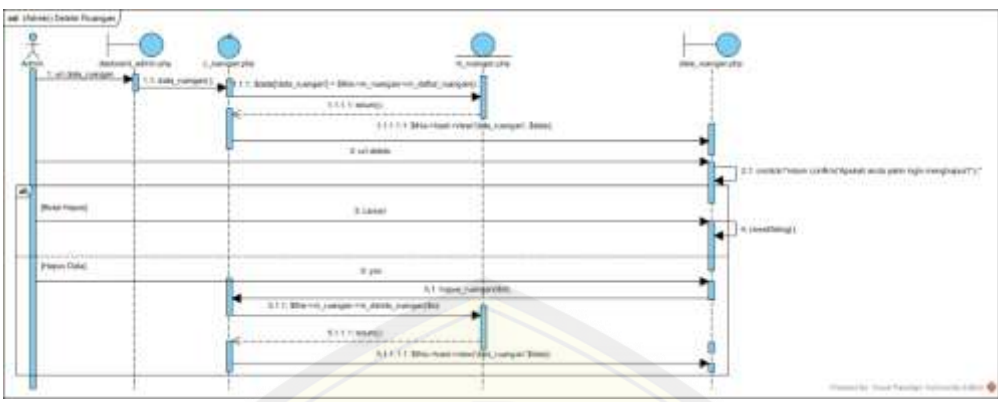
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Update Data Ruang, dijelaskan pada gambar 4.40.



Gambar 4. 40 Sequence Diagram Admin Update Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.1.12 Sequence Diagram Delete Data Ruang

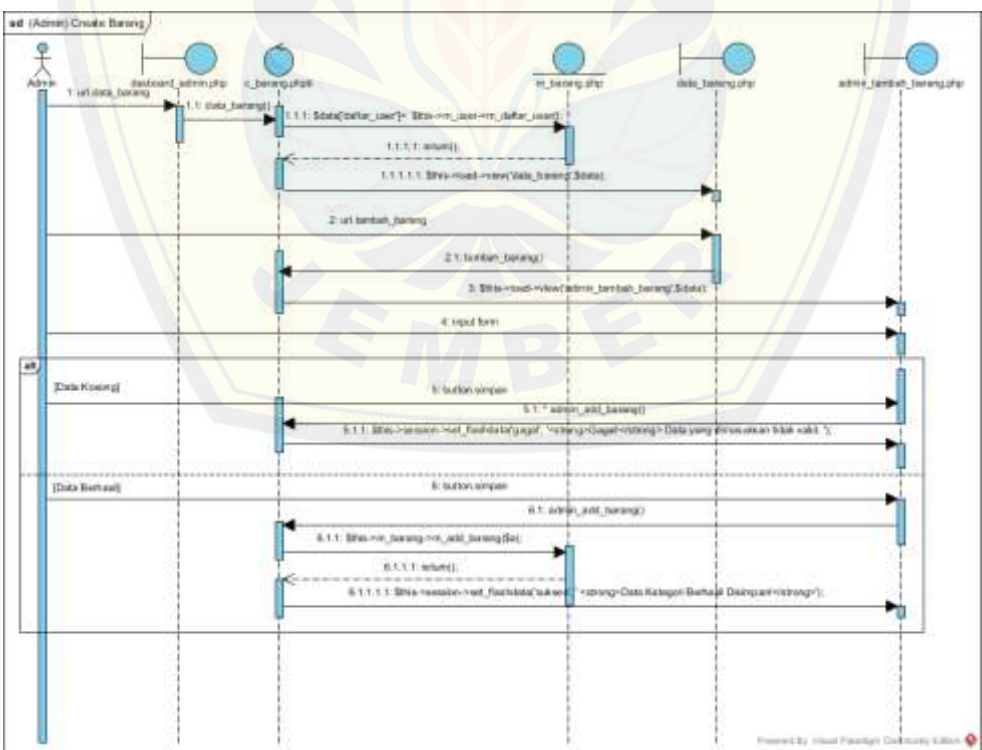
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Delete Data Ruang*, dijelaskan pada gambar 4.41.



Gambar 4. 41 Sequence Diagram Admin Delete Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.1.13 Sequence Diagram Create Barang

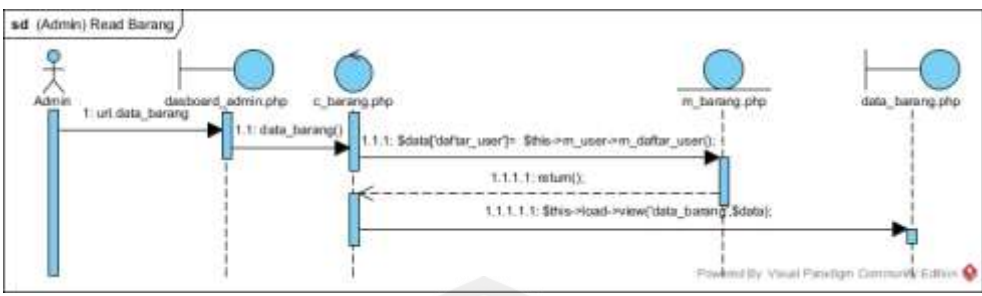
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Create Barang*, dijelaskan pada gambar 4.42.



Gambar 4. 42 Sequence Diagram Admin Create Barang
Sumber: Hasil Penelitian, 2015

4.5.1.14 Sequence Diagram Read Barang

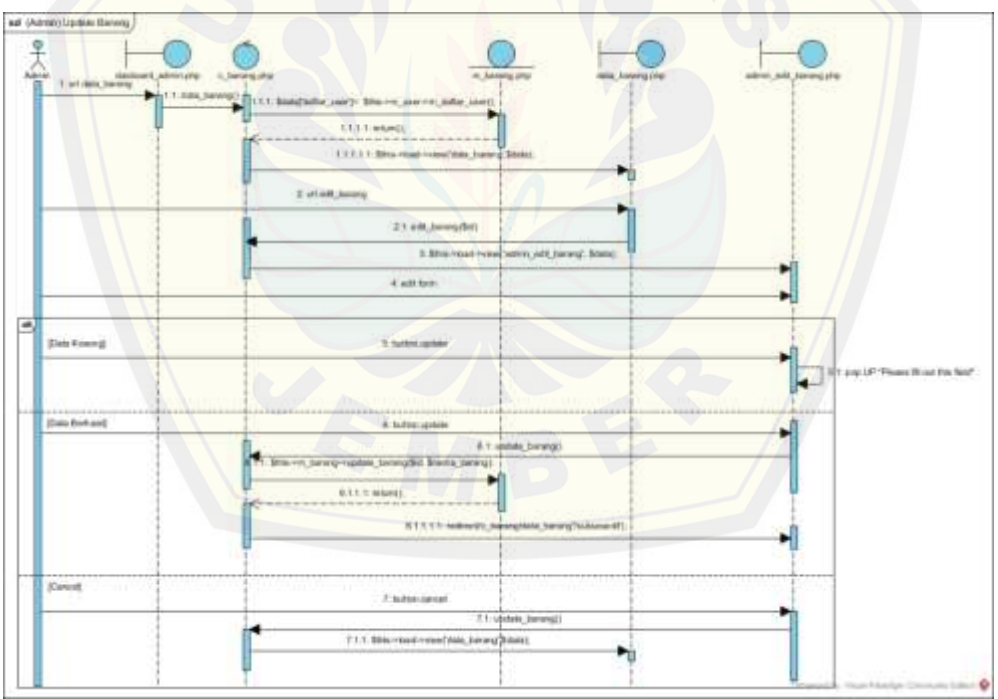
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Read Barang*, dijelaskan pada gambar 4.43.



Gambar 4. 43 Sequence Diagram Admin Read Barang
Sumber: Hasil Penelitian, 2015

4.5.1.15 Sequence Diagram Update Barang

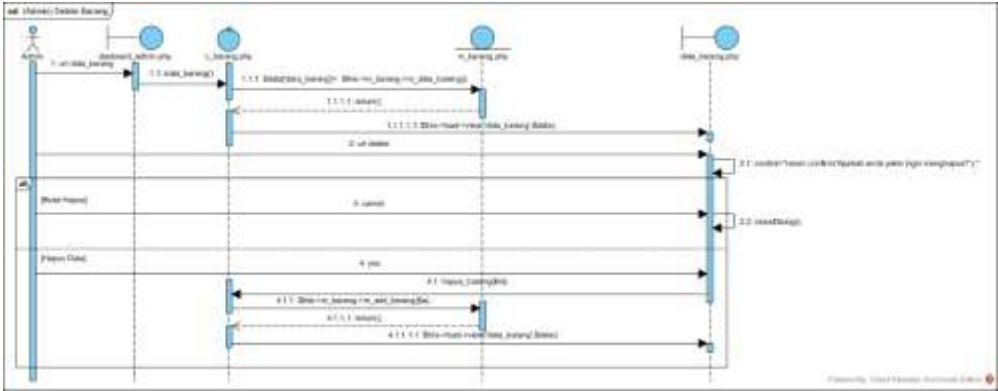
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Update Barang*, dijelaskan pada gambar 4.44.



Gambar 4. 44 Sequence Diagram Admin Update Barang
Sumber: Hasil Penelitian, 2015

4.5.1.16 Sequence Diagram Delete Barang

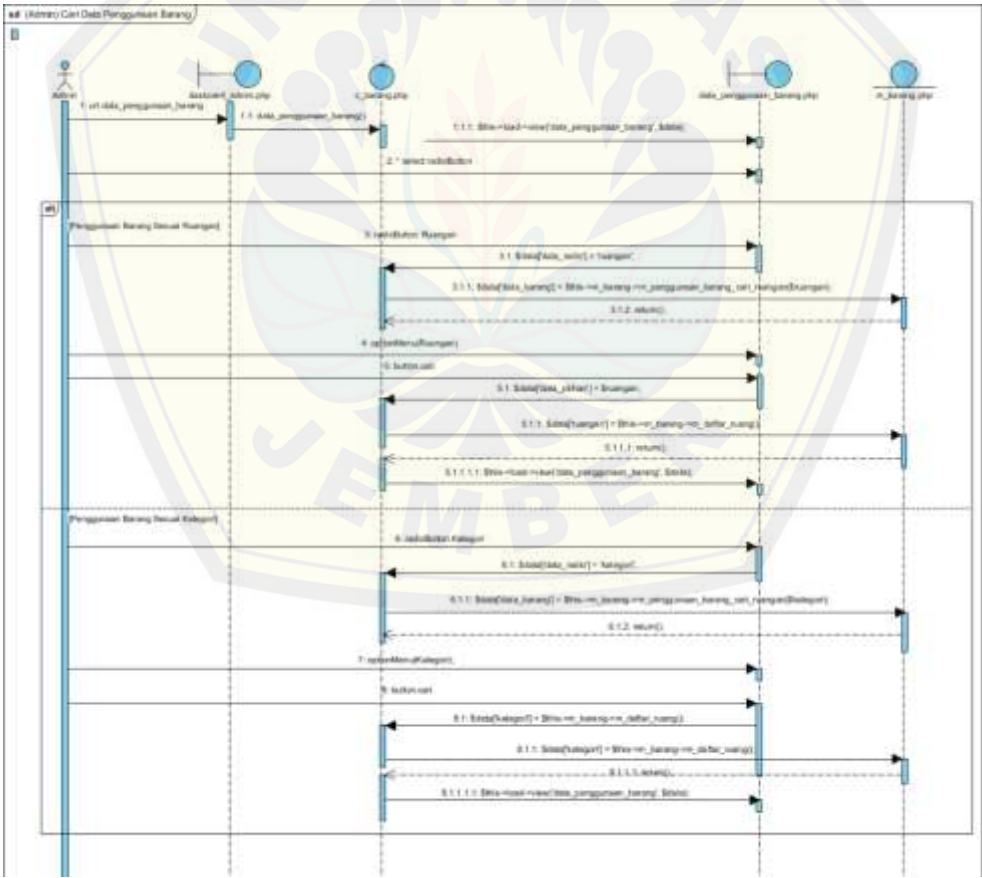
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Delete Barang*, dijelaskan pada gambar 4.45.



Gambar 4. 45 Sequence Diagram Admin Delete Barang
Sumber: Hasil Penelitian, 2015

4.5.1.17 Sequence Diagram Cari Data Penggunaan Barang

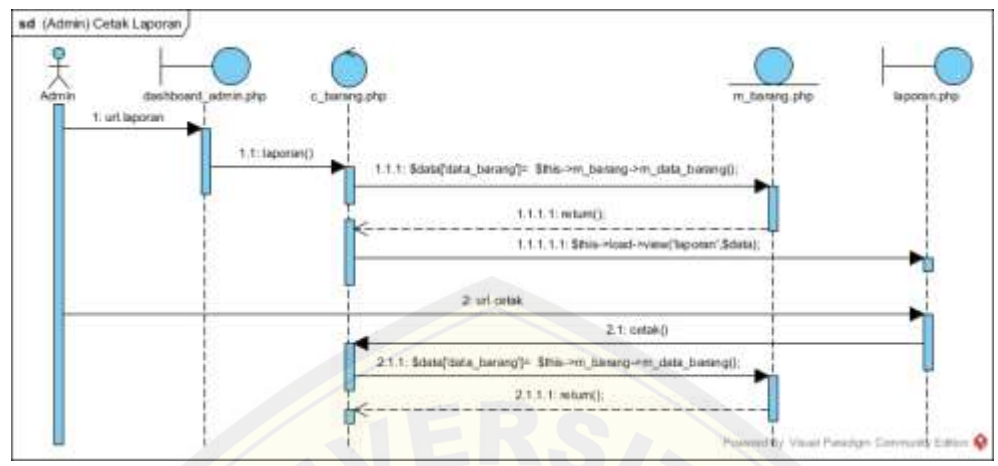
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram* Cari Data Penggunaan Barang, dijelaskan pada gambar 4.46.



Gambar 4. 46 Sequence Diagram Admin Cari Data Penggunaan Barang
Sumber: Hasil Penelitian, 2015

4.5.1.18 Sequence Diagram Cetak Laporan

Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram* Cetak Laporan, dijelaskan pada gambar 4.47.

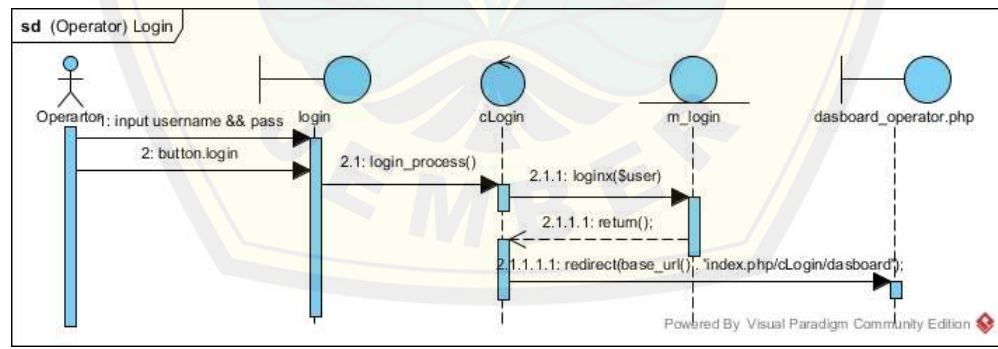


Gambar 4. 47 Sequence Diagram Admin Cetak Laporan
Sumber: Hasil Penelitian, 2015

4.5.2 Operator

4.5.2.1 Sequence Diagram Login

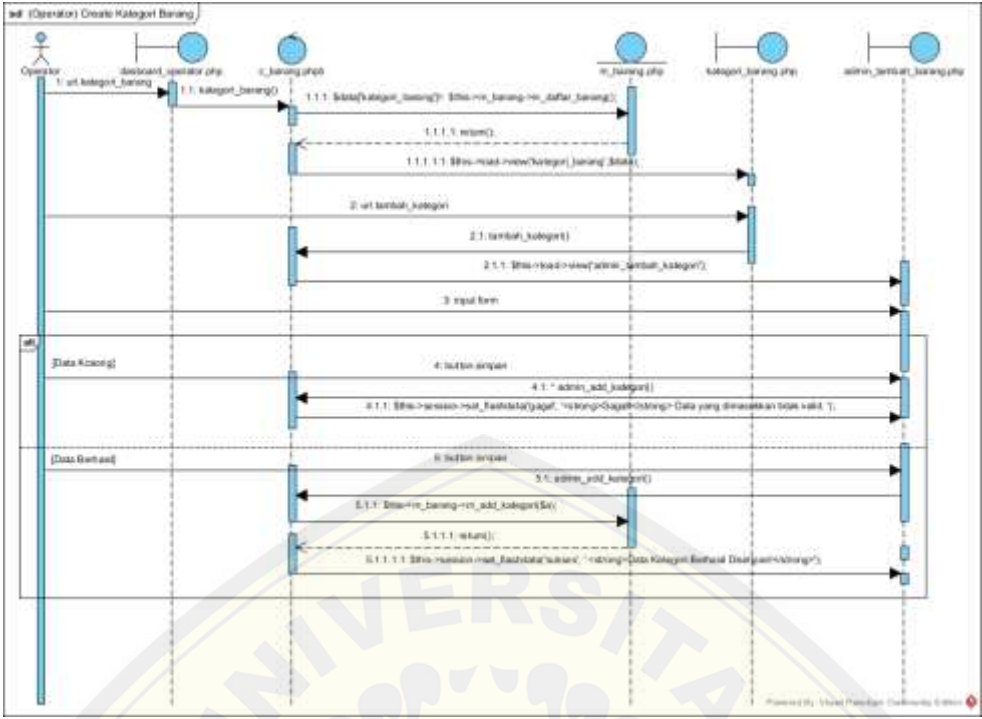
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram* Login, dijelaskan pada gambar 4.48.



Gambar 4. 48 Sequence Diagram Operator Login
Sumber: Hasil Penelitian, 2015

4.5.2.2 Sequence Diagram Create Kategori Barang

Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram* Create Kategori Barang, dijelaskan pada gambar 4.49.



Gambar 4. 49 Sequence Diagram Operator Create Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.2.3 Sequence Diagram Read Kategori Barang

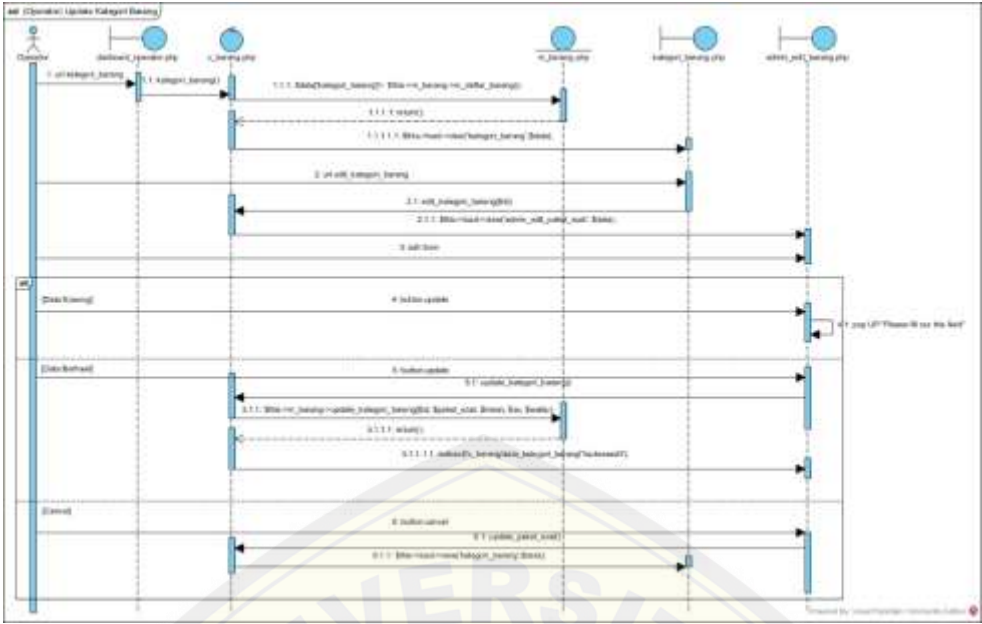
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Read Kategori Barang, dijelaskan pada gambar 4.50.



Gambar 4. 50 Sequence Diagram Operator Read Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.2.4 Sequence Diagram Update Kategori Barang

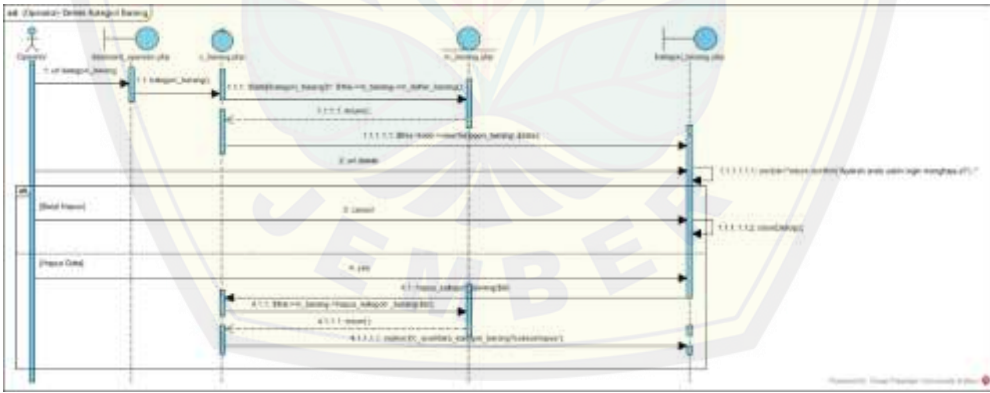
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Update Kategori Barang, dijelaskan pada gambar 4.51.



Gambar 4. 51 Sequence Diagram Operator Update Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.2.5 Sequence Diagram Delete Kategori Barang

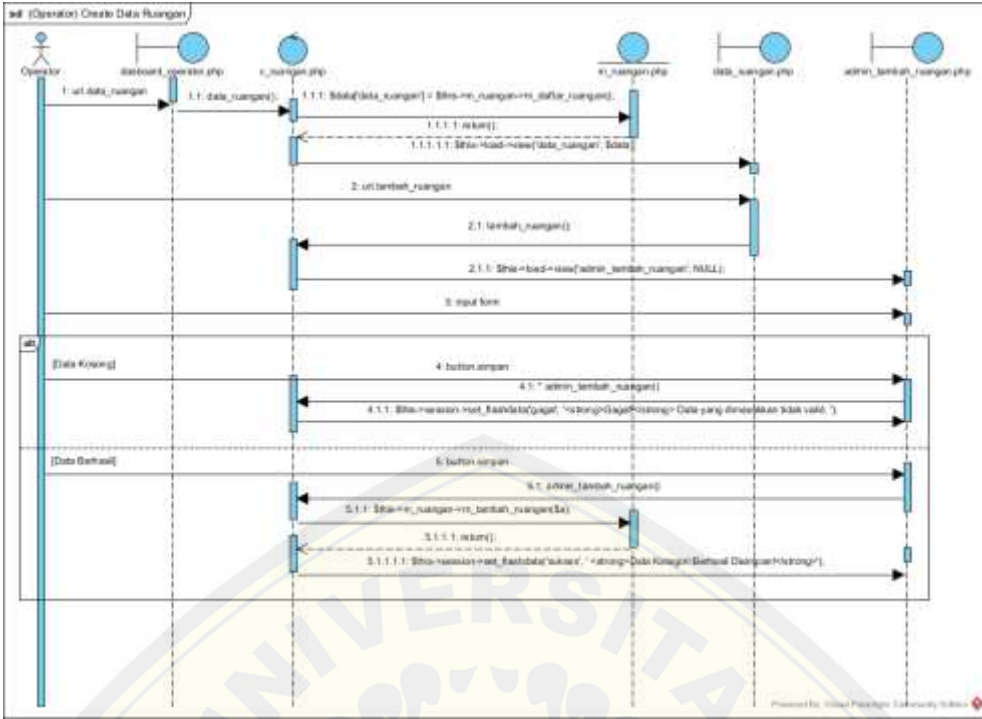
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Delete* Kategori Barang, dijelaskan pada gambar 4.52.



Gambar 4. 52 Sequence Diagram Operator Delete Kategori Barang
Sumber: Hasil Penelitian, 2015

4.5.2.6 Sequence Diagram Create Data Ruangan

Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Create Dta* Ruangan, dijelaskan pada gambar 4.53.



Gambar 4. 53 Sequence Diagram Operator Create Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.2.7 Sequence Diagram Read Data Ruang

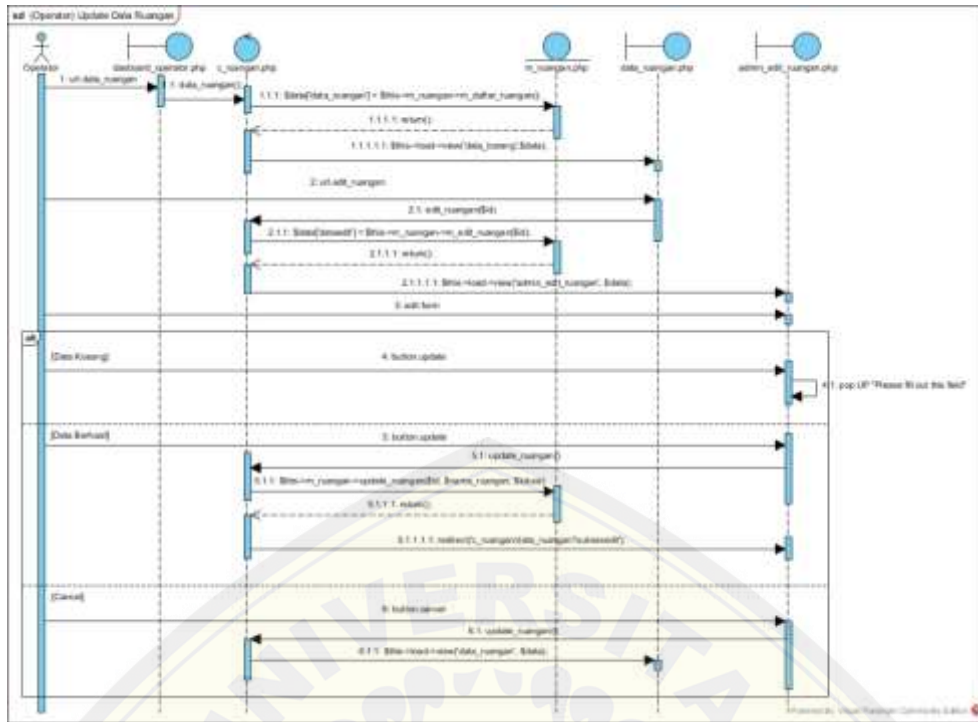
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Read Data Ruang , dijelaskan pada gambar 4.54.



Gambar 4. 54 Sequence Diagram Operator Read Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.2.8 Sequence Diagram Update Data Ruang

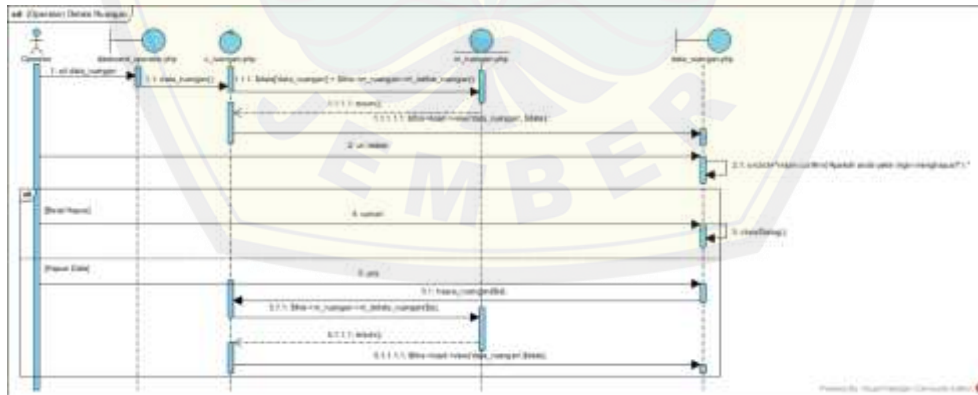
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Update Data Ruang, dijelaskan pada gambar 4.55.



Gambar 4. 55 Sequence Diagram Operator Update Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.2.9 Sequence Diagram Delete Data Ruang

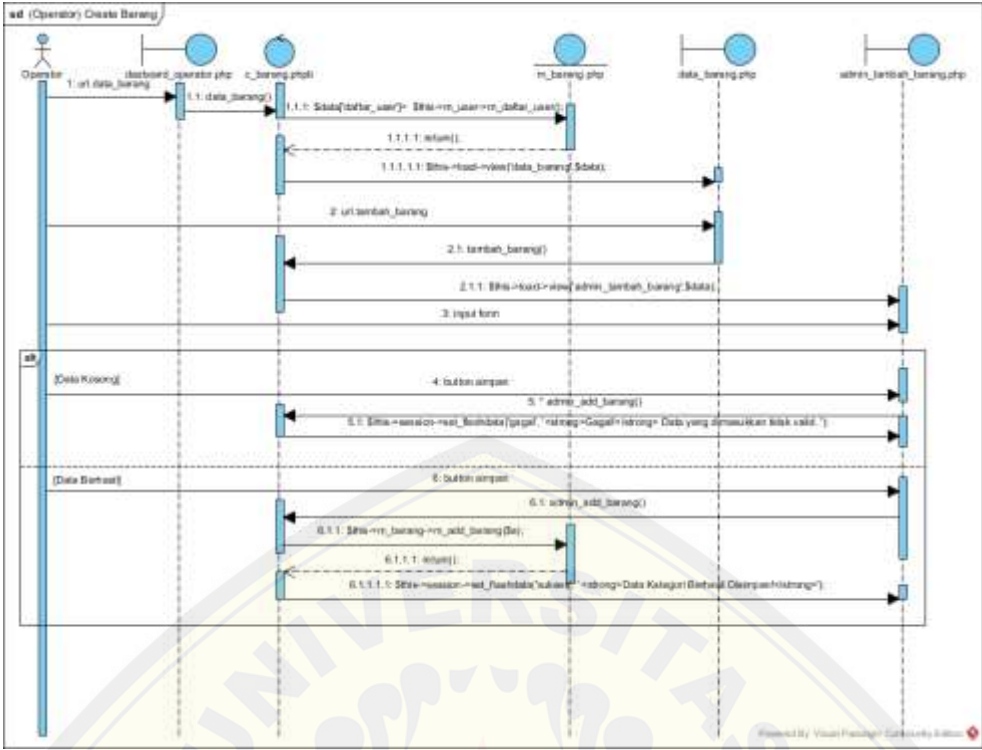
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Delete Data Ruang*, dijelaskan pada gambar 4.56.



Gambar 4. 56 Sequence Diagram Operator Delete Data Ruang
Sumber: Hasil Penelitian, 2015

4.5.2.10 Sequence Diagram Create Barang

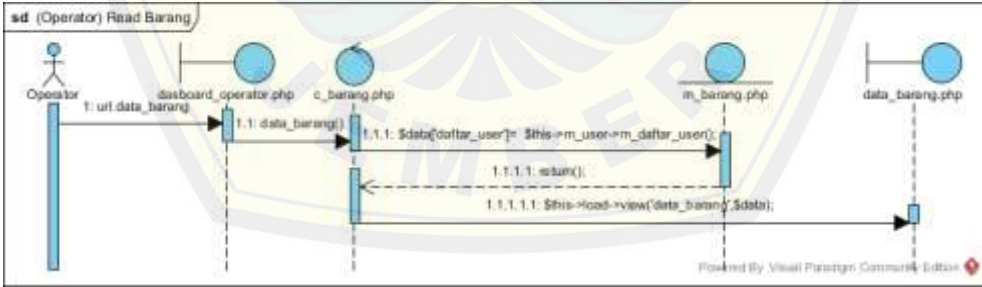
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Create Barang*, dijelaskan pada gambar 4.57.



Gambar 4. 57 Sequence Diagram Operator Create Barang
Sumber: Hasil Penelitian, 2015

4.5.2.11 Sequence Diagram Read Barang

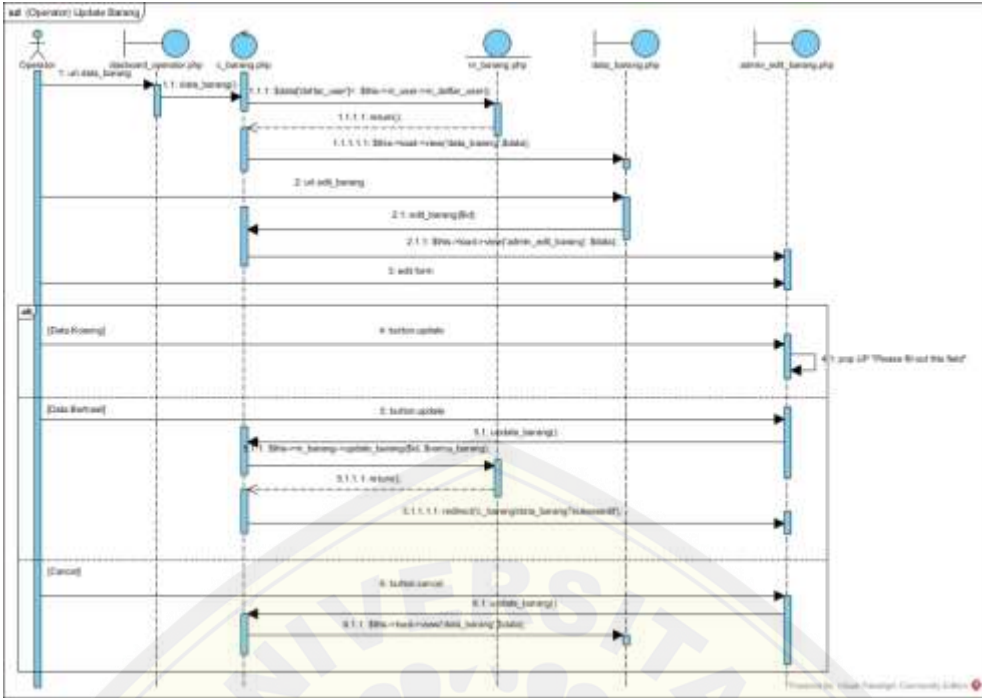
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Read Barang, dijelaskan pada gambar 4.58.



Gambar 4. 58 Sequence Diagram Operator Read Barang
Sumber: Hasil Penelitian, 2015

4.5.2.12 Sequence Diagram Update Barang

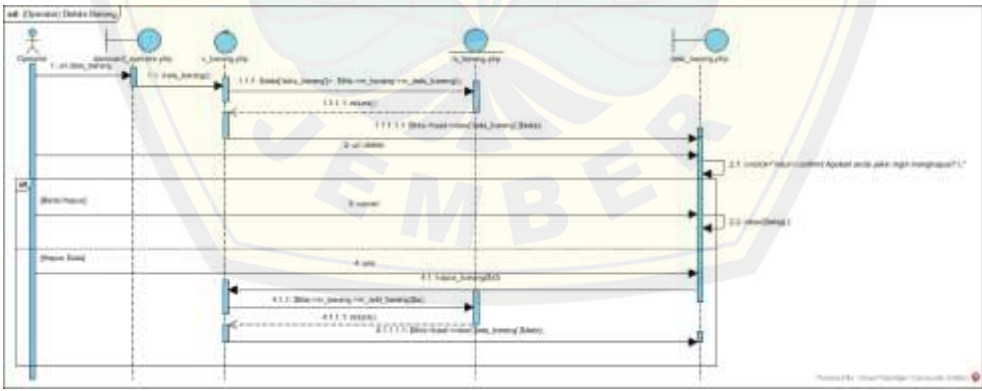
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Update Barang, dijelaskan pada gambar 4.59.



Gambar 4. 59 Sequence Diagram Operator Update Barang
Sumber: Hasil Penelitian, 2015

4.5.2.13 Sequence Diagram Delete Barang

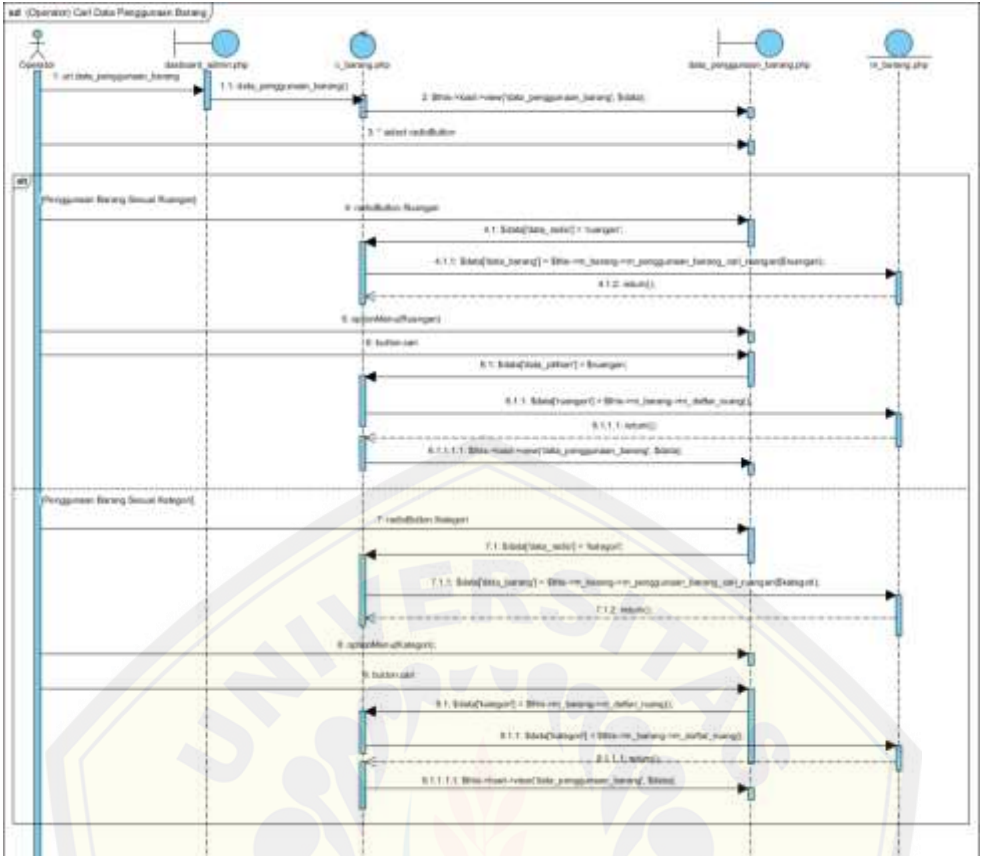
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Delete* Barang, dijelaskan pada gambar 4.60.



Gambar 4. 60 Sequence Diagram Operator Delete Barang
Sumber: Hasil Penelitian, 2015

4.5.2.14 Sequence Diagram Cari Data Penggunaan Barang

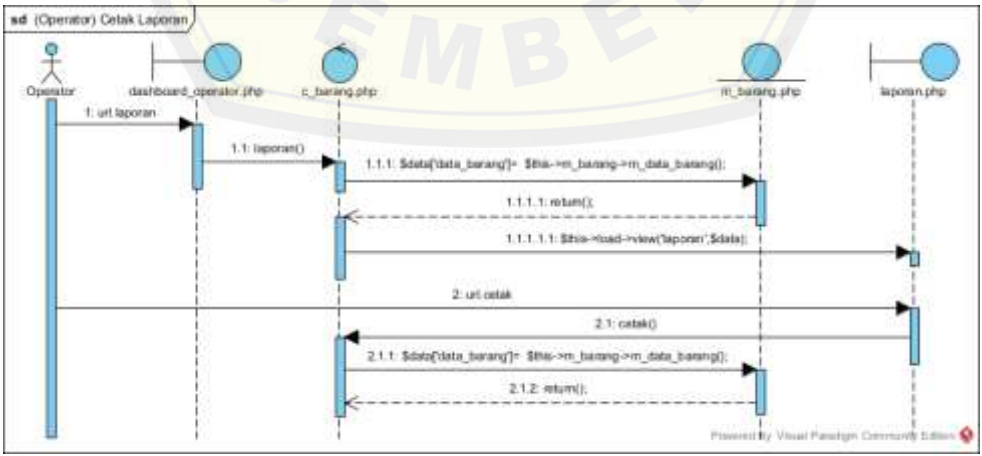
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Cari Data* Penggunaan Brang, dijelaskan pada gambar 4.61.



Gambar 4. 61 Sequence Diagram Operator Cari Data Penggunaan Brang
Sumber: Hasil Penelitian, 2015

4.5.2.15 Sequence Diagram Cetak Laporan

Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Cetak Laporan, dijelaskan pada gambar 4.62.

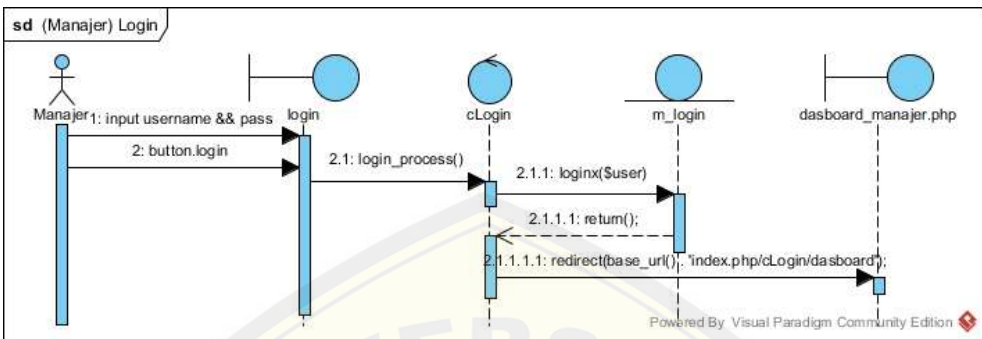


Gambar 4. 62 Sequence Diagram Operator Cetak Laporan
Sumber: Hasil Penelitian, 2015

4.5.3 Manajer

4.5.3.1 Sequence Diagram Login

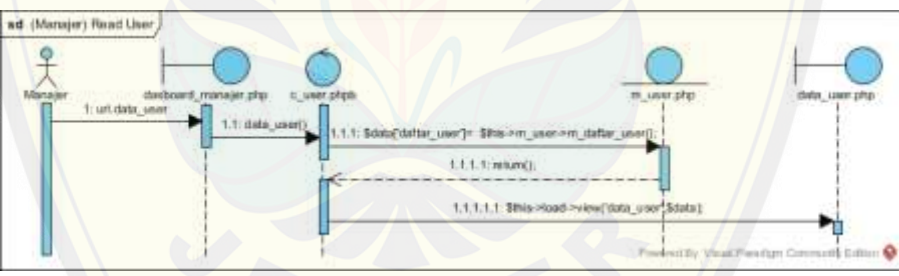
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Login*, dijelaskan pada gambar 4.63.



Gambar 4. 63 Sequence Diagram Manajer Login
Sumber: Hasil Penelitian, 2015

4.5.3.2 Sequence Diagram Read User

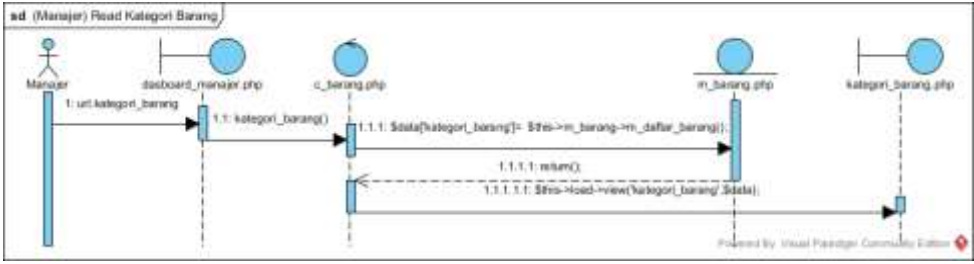
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Read User*, dijelaskan pada gambar 4.64.



Gambar 4. 64 Sequence Diagram Manajer Read User
Sumber: Hasil Penelitian, 2015

4.5.3.3 Sequence Diagram Read Kategori Barang

Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Read Kategori Barang*, dijelaskan pada gambar 4.65.



Gambar 4. 65 Sequence Diagram Manajer Read Kategori Barang
 Sumber: Hasil Penelitian, 2015

4.5.3.4 Sequence Diagram Read Data Ruangan

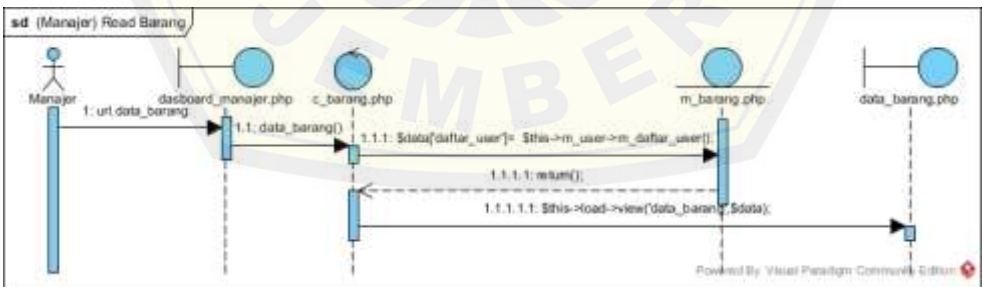
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Read Data Ruangan*, dijelaskan pada gambar 4.66.



Gambar 4. 66 Sequence Diagram Manajer Read Data Ruangan
 Sumber: Hasil Penelitian, 2015

4.5.3.5 Sequence Diagram Read Data Barang

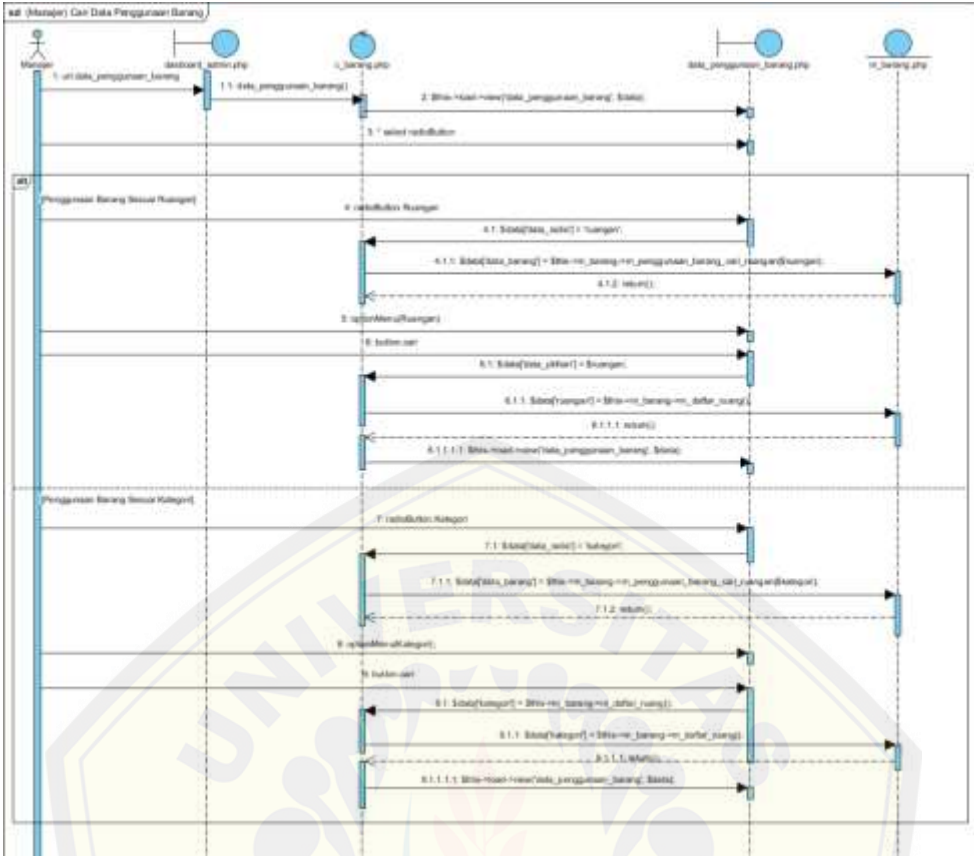
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Read Data Barang*, dijelaskan pada gambar 4.67.



Gambar 4. 67 Sequence Diagram Manajer Read Data Barang
 Sumber: Hasil Penelitian, 2015

4.5.3.6 Sequence Diagram Cari Data Penggunaan Barang

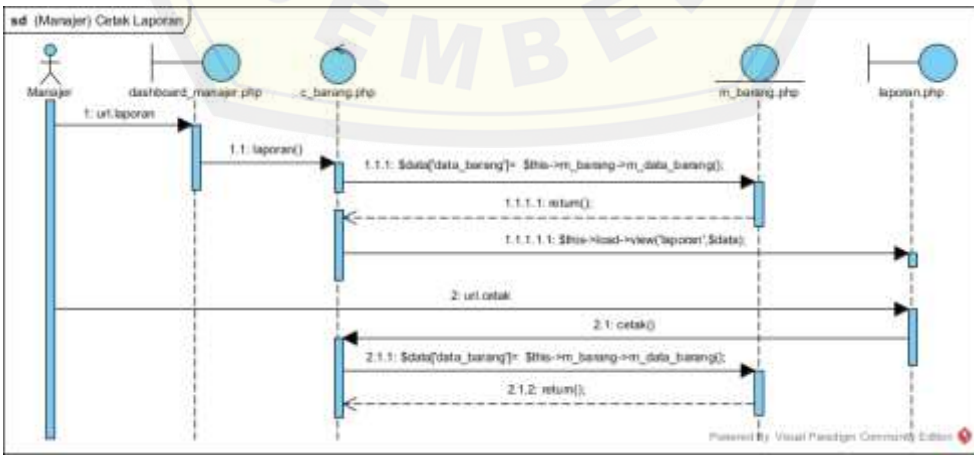
Penjelasan aliran SIMBI dapat dilihat dari *sequence diagram Cari Data Penggunaan Barang*, dijelaskan pada gambar 4.68.



Gambar 4. 68 Sequence Diagram Manajer Cari Data Penggunaan Barang
Sumber: Hasil Penelitian, 2015

4.5.3.7 Sequence Diagram Cetak Laporan

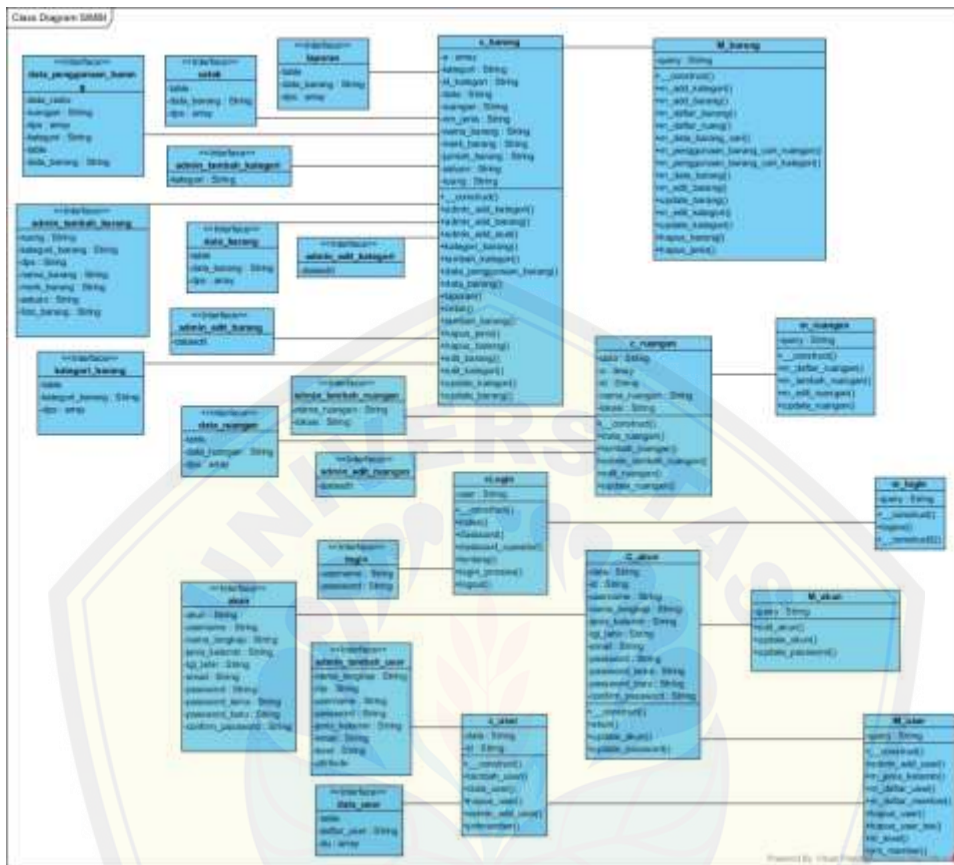
Penjelasan aliran SIMBI dapat dilihat dari sequence diagram Cetak Laporan, dijelaskan pada gambar 4.69.



Gambar 4. 69 Sequence Diagram Manajer Cetak Laporan
Sumber: Hasil Penelitian, 2015

4.6 Class Diagram

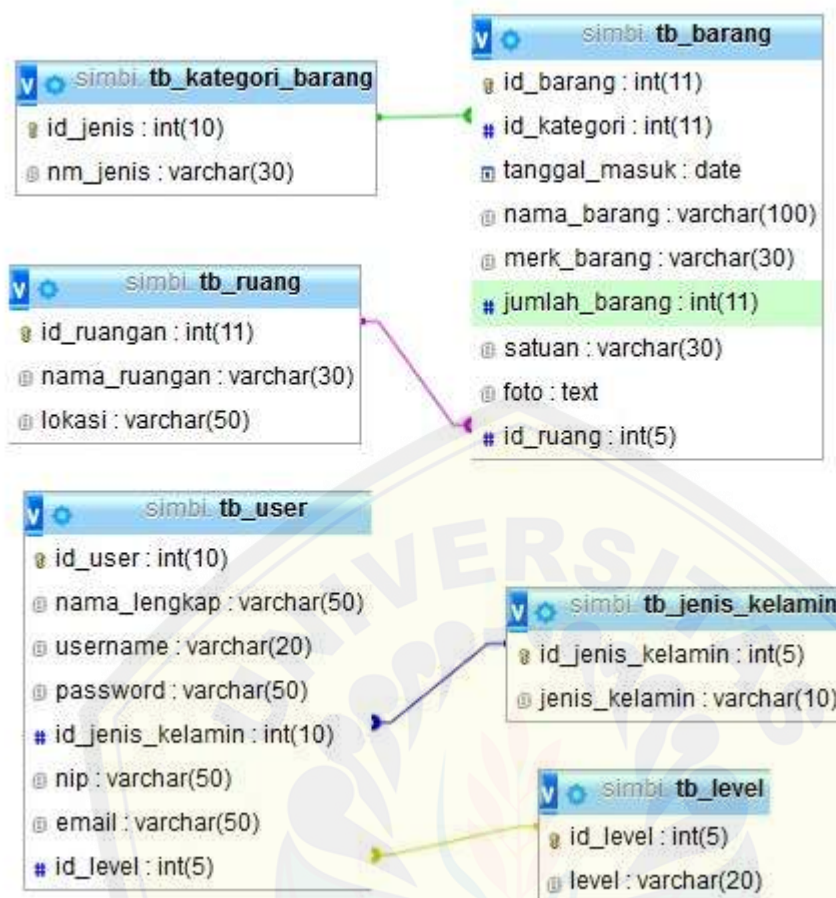
Setelah melalui tahap pembuatan desain dengan *sequence diagram*, tahap selanjutnya membuat desain perancangan *class diagram*, seperti pada gambar 4.70.



Gambar 4. 70 Class Diagram SIMBI
Sumber: Hasil Penelitian, 2015

4.7 Entity Relation Diagram

Setelah pembuatan *class diagram*, tahap perancangan selanjutnya yaitu membuat desain *database* sistem informasi manajemen inventarisasi barang, dapat dilihat gambar 4.71.



Gambar 4. 71 Entity Relation Diagram SIMBI
Sumber: Hasil Penelitian, 2015

4.8 Implementasi Perancangan

Setelah tahap desain perancangan selesai, tahap selanjutnya dalam penelitian ini yaitu tahap pengimplementasian desain perancangan ke dalam bahasa pemrograman. Bahasa pemrograman yang dipakai adalah bahasa pemrograman PHP (*Hypertext Preprocessor*) dan menggunakan *database* MySQL.

Dalam perancangan sistem informasi manajemen inventarisasi barang ini menggunakan *framework* Code Igniter untuk memudahkan dalam pengembangan dan penulisan *coding* di dalam sebuah pemrograman.

Pada tahap implementasi perancangan ini menjelaskan tentang fitur – fitur yang terdapat pada sistem informasi manajemen inventarisasi barang. Fitur-fitur tersebut

meliputi manajemen data *user*, manajemen kategori barang, manajemen ruangan, manajemen data barang dan lain-lainnya.



BAB 5. HASIL DAN PEMBAHASAN

Bab ini menjelaskan hasil analisis dan pembahasan tentang Sistem Informasi Manajemen Inventarisasi Barang Berbasis Web pada Program Studi Sistem Informasi Universitas Jember yang sudah dibuat.

5.1 Pengodean Program

Sistem informasi manajemen inventarisasi barang berbasis web pada Program Studi Sistem Informasi Universitas Jember ini dibuat dengan menggunakan metode pengembangan *waterfall* seperti yang sudah dijelaskan pada bab-bab sebelumnya. Tahapan yang dilakukan mulai dari pengumpulan data mengidentifikasi kebutuhan, perancangan sistem dengan menggunakan diagram UML dan selanjutnya pengodean sistem.

Tahap pengodean sistem ini menerjemahkan perancangan sistem yang sudah dibuat kedalam bahasa pemrograman dengan menggunakan bahasa pemrograman PHP dan menggunakan *framework codeigniter*.

Sistem informasi manajemen inventarisasi barang berbasis web pada Program Studi Sistem Informasi Universitas Jember dalam pengodean sistemnya terbagi menjadi tiga *pattern* yaitu *view*, *controller* dan *model*. Hasil dari pengodean sistem ini hanya menjelaskan kode-kode program pada bagian inti dari setiap fitur yaitu pada bagian *controller*.

5.1.1 *Login*

Controller pada fungsi *login* berisi tentang *statement* untuk masuk ke sistem sesuai hak masing-masing user berdasarkan *username* dan *password* yang diinputkan. *Controller* pada fitur *login* adalah *cLogin*. Kode programnya dapat dilihat pada tabel 5.1.

Tabel 5.1 Kode program *cLogin* pada fungsi *login*

Sumber: Hasil Analisis, 2015.

| |
|-------------------|
| <i>Controller</i> |
|-------------------|

```

<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class cLogin extends CI_Controller {

    function __construct() {
        parent::__construct();

        $this->load->model('m_login');
    }

    public function index() {
        $this->load->view('login');
    }

    public function dashboard() {
        $this->load->view('dashboard');
    }

    public function dashboard_operator() {
        $this->load->view('dashboard_operator');
    }

    public function tentang() {
        $this->load->view('tentang');
    }

    function login_process() {
        $user = array('username' => $this->input-
>post('username'),
        'password' => $this->input->post('password')
        );
        if (!$this->m_login->loginx($user)) {
            redirect(base_url() . '?salah');
        } else {
            if ($this->session->userdata('user_on')) {
                if ($this->session->userdata('admin')) {
                    redirect(base_url()
'index.php/cLogin/dashboard');
                } elseif ($this->session-
>userdata('operator')) {
                    redirect(base_url()
'index.php/cLogin/dashboard');
                } elseif ($this->session->userdata('manajer'))
{
                    redirect(base_url()
'index.php/cLogin/dashboard');
                } else {
            }
        } else {
            $this->logout();
        }
    }
}

```



```

    }
}

public function logout() {
    $this->session->sess_destroy();
    redirect(base_url());
}
}
?>

```

5.1.2 Manajemen Data User

Controller pada fungsi manajemen data *user* berisi tentang *statement* untuk memproses manajemen data *user*. *Controller* pada fitur manajemen data *user* adalah *c_user* yang memiliki *function* *tambah_user()*, *data_user()*, *hapus_user()*, dan *admin_add_user()*. Kode programnya dapat dilihat pada tabel 5.2.

Tabel 5.2 Kode program *c_user* pada fitur manajemen data *user*

Sumber: Hasil Analisis, 2015.

| <i>Controller</i> |
|--|
| <pre> <?php if (!defined('BASEPATH')) exit('No direct script access allowed'); class c_user extends CI_Controller { function __construct() { parent::__construct(); \$this->load->model('m_user'); } public function tambah_user() { \$data['jenis_kelamin'] = \$this->m_user- >m_jenis_kelamin(); \$data['level'] = \$this->m_user->m_level(); \$this->load->view('admin_tambah_user', \$data); } public function data_user() { \$data['daftar_user'] = \$this->m_user- >m_daftar_user(); \$this->load->view('data_user', \$data); } public function hapus_user(\$id) { if (\$id != 0) { \$this->m_user->hapus_user(\$id); \$this->session->set_flashdata('sukses', </pre> |

```

'<strong>Data Berhasil Dihapus</strong>');
        redirect(base_url()
'index.php/c_user/data_user');
    } else {

        $this->session->set_flashdata('gagal',
'<strong>Data Gagal Dihapus</strong>');
        redirect(base_url()
'index.php/c_user/data_user');
    }

    function admin_add_user() {
        $a = array
        (
            'nama_lengkap' => $_POST['nama_lengkap']
            , 'username' => $_POST['username']
            , 'password' => $_POST['password']
            , 'id_jenis_kelamin' =>
$_POST['id_jenis_kelamin']
            , 'email' => $_POST['email']
            , 'nip' => $_POST['nip']
            , 'id_level' => $_POST['id_level']
        );

        if (empty($a[nama_lengkap]) && empty($a[username])
&& empty($a[password]) && empty($a[id_jenis_kelamin]) &&
empty($a[email]) && empty($a[nip]) && empty($a[id_level])) {
            $this->session->set_flashdata('gagal',
'<strong>Gagal!</strong> Data yang dimasukkan tidak valid. ');
            redirect(base_url()
'index.php/c_user/tambah_user');
        } else {
            $this->m_user->admin_add_user($a);
            $this->session->set_flashdata('sukses',
'<strong>Berhasil Disimpan!</strong>');
            redirect(base_url()
'index.php/c_user/tambah_user');
        }
    }

}
?>

```

5.1.3 Manajemen Data Barang

Controller pada fungsi manajemen data barang berisi tentang *statement* untuk memproses data barang. *Controller* pada manajemen data barang adalah *c_barang* yang memiliki *function* *admin_add_kategori()*, *admin_add_barang()*, *kategori_barang()*, *tambah_kategori()*, *data_penggunaan_barang()*, *data_barang()*, *laporan()*, *cetak()*, *tambah_barang()*, *hapus_jenis()*, *hapus_barang()*, *edit_barang()*,

edit_kategori(), *update_kategori()*, dan *update_barang()*. Kode programnya dapat dilihat pada tabel 5.3.

Tabel 5.3 Kode program *c_barang* pada fitur manajemen data barang

Sumber: Hasil Analisis, 2015.

| <i>Controller</i> |
|--|
| <pre> <?php if (!defined('BASEPATH')) exit('No direct script access allowed'); class c_barang extends CI_Controller { function __construct() { parent::__construct(); \$this->load->model('m_barang'); } function admin_add_kategori() { \$a = array ('kategori' => \$_POST['kategori']); if (empty(\$a[kategori])) { \$this->session->set_flashdata('gagal', 'Gagal! Data yang dimasukkan tidak valid. '); redirect(base_url() 'index.php/c_soal/tambah_soal'); } else { \$this->m_barang->m_add_kategori(\$a); \$this->session->set_flashdata('sukses', 'Data Kategori Berhasil Disimpan!'); redirect(base_url() 'index.php/c_barang/kategori_barang'); } } function admin_add_barang() { \$a = array('id_kategori' => \$_POST['id_kategori'] , 'nama_barang' => \$_POST['nama_barang'] , 'merk_barang' => \$_POST['merk_barang'] , 'jumlah_barang' => \$_POST['jumlah_barang'] , 'satuan' => \$_POST['satuan'] , 'ruang' => \$_POST['ruang']); if (empty(\$a[id_kategori])) { \$this->session->set_flashdata('gagal', 'Gagal! Data yang dimasukkan tidak valid. '); redirect(base_url() 'index.php/c_soal/tambah_soal'); } } } </pre> |

```

    } else {
        $this->m_barang->m_add_barang($a);
    }
}

move_uploaded_file($_FILES['foto_barang']['tmp_name'], 'foto/' .
$_FILES['foto_barang']['name']);
$this->session->set_flashdata('sukses', '
<strong>Data Kategori Berhasil Disimpan!</strong>');
redirect(base_url()
'index.php/c_barang/data_barang');
}

}

public function kategori_barang() {
    $data['kategori_barang'] = $this->m_barang-
>m_daftar_barang();
    $this->load->view('kategori_barang', $data);
}

public function tambah_kategori() {
    $this->load->view('admin_tambah_kategori');
}

function data_penggunaan_barang() {
    $data['data_radio'] = 'ruangan';
    $data['data_pilihan'] = NULL;
    $data['data_barang'] = NULL;
    $ruangan = NULL;
    $kategori = NULL;
    //if (isset($_POST['caril'])) {
    error_reporting(0);

    if (isset($_POST)) {
        $ruangan = $_POST['ruangan'];
        $kategori = $_POST['kategori'];
        if ($ruangan != NULL) {
            $data['data_radio'] = 'ruangan';
            $data['data_pilihan'] = $ruangan;
            $data['data_barang'] = $this->m_barang-
>m_penggunaan_barang_cari_ruangan($ruangan);
        } else if ($kategori != NULL) {
            $data['data_radio'] = 'kategori';
            $data['data_pilihan'] = $kategori;
            $data['data_barang'] = $this->m_barang-
>m_penggunaan_barang_cari_kategori($kategori);
        }
    }

    $data['ruangan'] = $this->m_barang->m_daftar_ruang();
    $data['kategori'] = $this->m_barang-
>m_daftar_barang();
    $this->load->view('data_penggunaan_barang', $data);
}

public function data_barang() {
    if (isset($_POST['cari'])) {

```

```
        $data['data_barang'] = $this->m_barang-
>m_data_barang_cari($_POST['tanggal_mulai'],
$_POST['tanggal_ahir']);
    } else {
        $data['data_barang'] = $this->m_barang-
>m_data_barang();
        //$data['data_barang'] = NULL;
    }
    $this->load->view('data_barang', $data);
}

public function laporan() {
    if (isset($_POST['cari'])){
        $data['data_barang'] = $this->m_barang-
>m_data_barang_cari($_POST['tanggal_mulai'],
$_POST['tanggal_ahir']);
    } else {
        //$data['data_barang'] = $this->m_barang-
>m_data_barang();
        $data['data_barang'] = NULL;
    }
    $this->load->view('laporan', $data);
}

public function cetak() {
    $data['data_barang'] = $this->m_barang-
>m_data_barang();
    $this->load->view('cetak', $data);
}

public function tambah_barang() {
    $data['kategori_barang'] = $this->m_barang-
>m_daftar_barang();
    $data['ruang'] = $this->m_barang->m_daftar_ruang();
    $this->load->view('admin_tambah_barang', $data);
}

public function hapus_jenis($id) {
    $this->m_barang->hapus_jenis($id);

    redirect('c_barang/kategori_barang?sukseshapus');
}

public function hapus_barang($id) {
    $this->m_barang->hapus_barang($id);
    redirect('c_barang/data_barang?sukseshapus');
}

function edit_barang($id) {
    $data['dataedit'] = $this->m_barang-
>m_edit_barang($id);
    $this->load->view('admin_edit_barang', $data);
}

function edit_kategori($id) {
    $data['dataedit'] = $this->m_barang-
>m_edit_kategori($id);
```

```

        $this->load->view('admin_edit_kategori', $data);
    }

    function update_kategori() {
        $id = $this->input->post('id_jenis');
        $nm_jenis = $this->input->post('nm_jenis');

        if ($this->m_barang->update_kategori($id, $nm_jenis))
        {
            redirect('c_barang/kategori_barang?suksesedit');
        } else {
            redirect('c_barang/kategori_barang?gagaledit');
        }
    }

    function update_barang() {
        $id = $this->input->post('id_barang');
        $nama_barang = $this->input->post('nama_barang');
        $merk_barang = $this->input->post('merk_barang');
        $jumlah_barang = $this->input->post('jumlah_barang');
        $satuan = $this->input->post('satuan');
        $ruang = $this->input->post('ruang');

        $this->m_barang->update_barang($id, $nama_barang,
        $merk_barang, $jumlah_barang, $satuan, $ruang);
        if ($this->m_barang->update_barang($id, $nama_barang,
        $merk_barang, $jumlah_barang, $satuan, $ruang)) {
            redirect('c_barang/data_barang?suksesedit');
        } else {
            redirect('c_barang/data_barang?gagaledit');
        }
    }
}
?>

```

5.1.4 Manajemen Data Ruangan

Controller pada fungsi manajemen data ruangan berisi *statement* untuk memproses data ruangan. *Controller* pada manajemen data ruangan adalah *c_ruangan* yang memiliki *function* *data_ruangan()*, *tambah_ruangan()*, *admin_tambah_ruangan()*, *edit_ruangan()*, dan *update_ruangan()*. Kode programnya dapat dilihat pada tabel 5.4.

Tabel 5.4 Kode program *c_ruangan* pada fungsi manajemen data ruangan

Sumber: Hasil Analisis, 2015.

Controller

```

<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class c_ruangan extends CI_Controller {

    function __construct() {
        parent::__construct();
        $this->load->model('m_ruangan');
    }

    public function data_ruangan() {
        $data['data_ruangan'] = $this->m_ruangan-
>m_daftar_ruangan();
        $this->load->view('data_ruangan', $data);
    }

    public function tambah_ruangan() {
        $this->load->view('admin_tambah_ruangan', NULL);
    }

    public function admin_tambah_ruangan() {
        $a = array(
            'nama_ruangan' => $_POST['nama_ruangan']
            , 'lokasi' => $_POST['lokasi']
        );
        if (empty($a[nama_ruangan]) && empty($a[lokasi])) {
            $this->session->set_flashdata('gagal',
'<strong>Gagal!</strong> Data yang dimasukkan tidak valid. ');
            redirect(base_url()
'index.php/c_ruangan/data_ruangan');
        } else {
            $this->m_ruangan->m_tambah_ruangan($a);
            $this->session->set_flashdata('sukses',
<strong>Data Kategori Berhasil Disimpan!</strong>');
            redirect(base_url()
'index.php/c_ruangan/data_ruangan');
        }
    }

    public function edit_ruangan($id) {
        $data['dataedit'] = $this->m_ruangan-
>m_edit_ruangan($id);
        $this->load->view('admin_edit_ruangan', $data);
    }

    function update_ruangan() {
        $id = $this->input->post('id_ruangan');
        $nama_ruangan = $this->input->post('nama_ruangan');
        $lokasi = $this->input->post('lokasi');

        if ($this->m_ruangan->update_ruangan($id,
$nama_ruangan, $lokasi)) {
            redirect('c_ruangan/data_ruangan?suksesedit');
        } else {
            redirect('c ruangan/data ruangan?gagaledit');
        }
    }
}

```




5.2 Pengujian Program

5.2.1 Pengujian *White Box*

Pengujian *whitebox* testing terdiri dari *listing program*, *cyclomatic complexity*, diagram alir, jalur program independen dan *test case*. Pada tahap ini fitur yang diuji adalah Manajemen Data *User* sebagai berikut:

a) Listing program fitur Manajemen Data *User*

Listing program fitur Manajemen Data *User* dapat dilihat pada gambar 5.1 dan gambar 5.2

```

6  class c_user extends CI_Controller {
7
8  function __construct() {
9      parent::__construct();
10     $this->load->model('m_user');
11 }
12
13 public function tambah_user() {
14     $data['jenis_kelamin'] = $this->m_user->m_jenis_kelamin();
15     $data['level'] = $this->m_user->m_level();
16     $this->load->view('admin_tambah_user', $data);
17 }
18
19 public function data_user() {
20     $data['daftar_user'] = $this->m_user->m_daftar_user();
21     $this->load->view('data_user', $data);
22 }

```

Gambar 5. 1 *Listing* program fitur Manajemen Data *User* bagian 1
Sumber: Hasil Penelitian, 2015

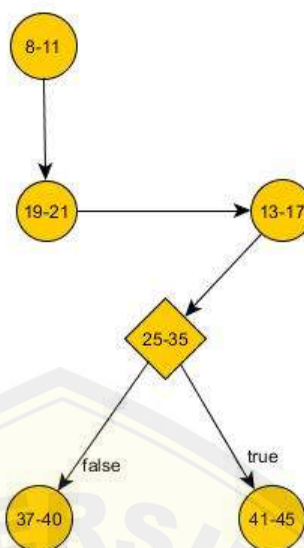
```

23 function admin_add_user() {
24     $e = array(
25         'name_lengkap' => $_POST['name_lengkap']
26         'username' => $_POST['username']
27         'password' => $_POST['password']
28         'id_jenis_kelamin' => $_POST['id_jenis_kelamin']
29         'email' => $_POST['email']
30         'nip' => $_POST['nip']
31         'id_level' => $_POST['id_level']
32     );
33
34     if (empty($e[name_lengkap]) || empty($e[username]) || empty($e[password]) || empty($e[id_jenis_kelamin])
35         || empty($e[email]) || empty($e[nip]) || empty($e[id_level])) {
36         $this->session->set_flashdata('gagal', '<strong>Gagal</strong> Data yang dimasukkan tidak valid. ');
37         redirect(base_url() . 'index.php/c_user/tambah_user');
38     } else {
39         $this->m_user->admin_add_user($e);
40         $this->session->set_flashdata('sukses', '<strong>Berhasil Disimpan</strong>');
41         redirect(base_url() . 'index.php/c_user/tambah_user');
42     }
43 }

```

Gambar 5. 2 *Listing* program fitur Manajemen Data *User* bagian 2
Sumber: Hasil Penelitian, 2015

b) Diagram alir fitur Manajemen Data *User*



Gambar 5. 3 Diagram alir fitur Manajemen Data *User*
 Sumber: Hasil Penelitian, 2015

c) Perhitungan *Cyclomatic Complexity* fitur Data *User*

Perhitungan diagram alir pada fitur *user* menggunakan *Cyclomatic Complexity* adalah sebagai berikut:

Class c_user : $V(G) = E - N + 2 = 6 - 5 + 2 = 3$

Kesimpulan: Kompleksitas dari alir diagram sederhana.

d) Pengujian jalur program fitur Data *User*

Pengujian jalur jalur program fitur *user* berdasarkan perhitungan *Cyclomatic Complexity* adalah sebagai berikut :

function data_user() : Jalur 1 : 8-11-19-21-13-17-25-35-37-40

Jalur 2 : 8-11-19-21-13-17-25-35-41-45

e) Test case fitur Data *User*

| <i>Test Case function data_user()</i> | |
|---------------------------------------|-----------------------------------|
| Jalur 1 | |
| <i>Test Case</i> | Jika <i>form user</i> tidak valid |

| | |
|------------------------|---|
| Target yang diharapkan | Mengeset <i>session</i> peringatan “GAGAL!” |
| Hasil pengujian | Benar |
| Path/Jalur | 8-11-19-21-13-17-25-35-37-40 |
| Jalur 2 | |
| <i>Test Case</i> | Jika <i>form user</i> valid |
| Target yang diharapkan | Menyimpan data <i>user</i> |
| Hasil pengujian | Benar |
| Path/Jalur | 8-11-19-21-13-17-25-35-41-45 |

5.2.2 Pengujian *Black Box*

Pengujian *blackbox* menitikberatkan pada fungsionalitas sistem. Pengujian ini tidak melihat kinerja internal dari sistem, jadi hanya berfokus pada kinerja sistem sesuai dengan spesifikasi dan kebutuhan yang dianalisis pada bab perancangan. Pengujian *blackbox* dilakukan oleh *developer* dan *reviewer* dari Program Studi Sistem Informasi Universitas Jember. Dokumen hasil pengujian dapat dilihat pada lampiran 1.

5.3 Pembahasan

Hasil penelitian ini menjelaskan tentang output dari sistem informasi inventarisasi barang beserta fitur-fiturnya. Sistem ini dapat diakses oleh 3 pihak yang berkepentingan, yaitu admin, operator, dan manajer.

Fitur-fitur yang terdapat di dalam sistem inventarisasi barang:

5.3.1 Halaman *Login*

Semua akun user harus melakukan login terlebih dahulu untuk masuk ke dalam sistem informasi inventarisasi barang. Admin memiliki keseluruhan akses pada semua fitur pada sistem inventarisasi barang. Admin menggunakan *username* admin dan *password* 1. Operator adalah akun untuk petugas inventarisasi. Operator menggunakan *username* operator dan *password* 2. Manajer adalah pimpinan pada

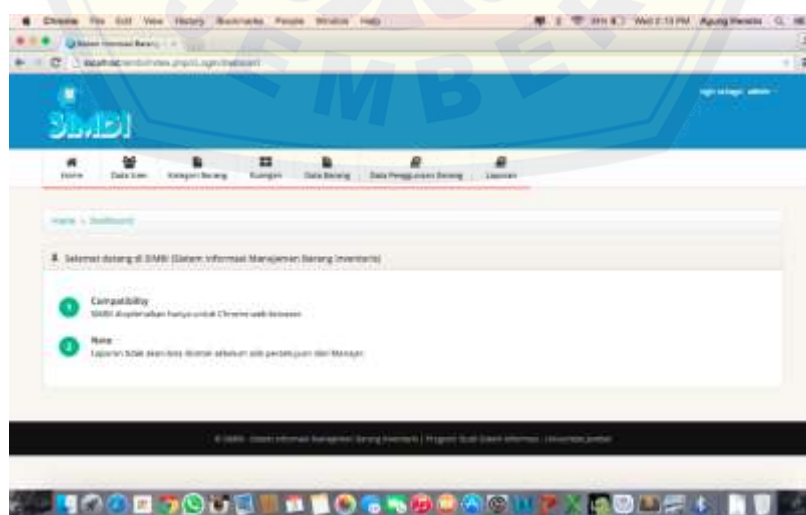
instansi terkait. Manajer menggunakan *username* manajer dan *password* 3. Dapat dilihat pada gambar 5.4.



Gambar 5. 4 Halaman *Login* SIMBI
Sumber: Hasil Penelitian, 2015

5.3.2 *Dashboard* Admin

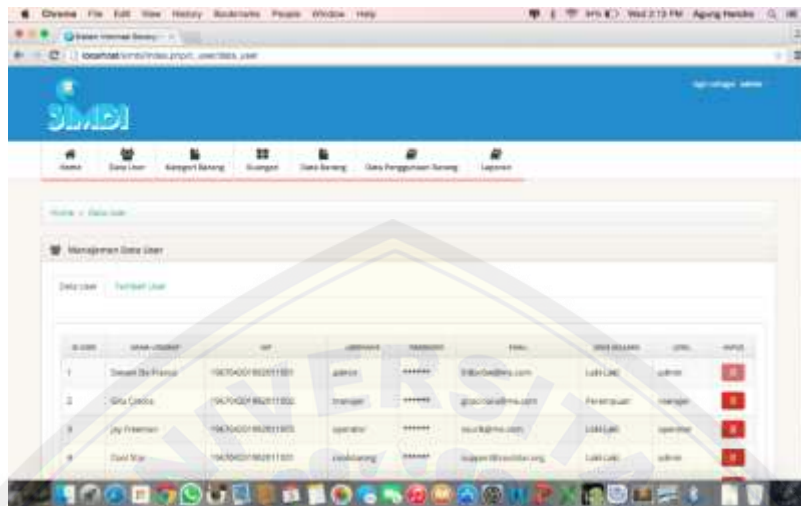
Setelah melakukan *login*, admin, operator, dan manajer akan masuk pada halaman *dashboard* admin. Terdapat beberapa keterangan mengenai sistem pada halaman ini dan fitur yang ada pada sistem inventarisasi barang yaitu Data *User*, Data Kategori, Data Ruang, Data Barang, Data Penggunaan Barang, dan Laporan. Khusus untuk akun operator tidak memiliki menu Data *User* sesuai dengan tugas dan fungsinya. Dapat dilihat pada gambar 5.5.



Gambar 5. 5 Halaman *Dashboard* Admin
Sumber: Hasil Penelitian, 2015

5.3.3 Data User

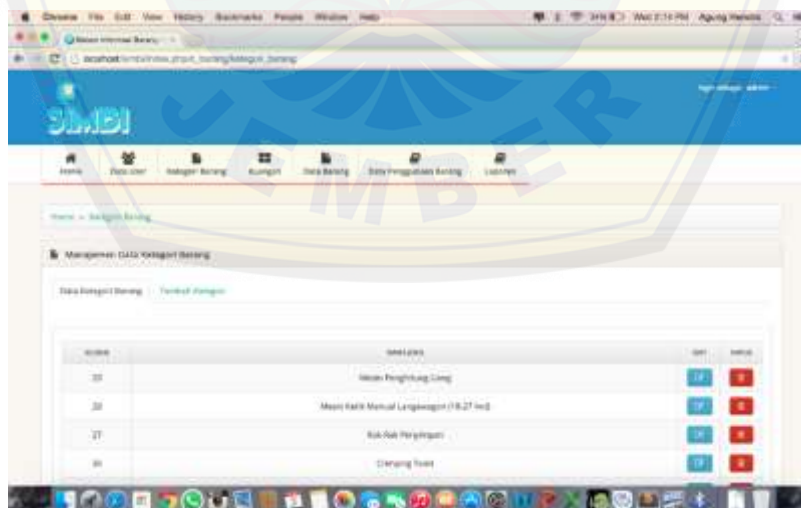
Menu data user merupakan fitur manajemen data yang meliputi tambah data *user* dan hapus data *user*. Menu ini hanya bisa diolah oleh akun admin. Manajer hanya bisa melihat siapa saja *user* yang terdaftar. Dapat dilihat pada gambar 5.6.



Gambar 5. 6 Halaman Data User
Sumber: Hasil Penelitian, 2015

5.3.4 Kategori Barang

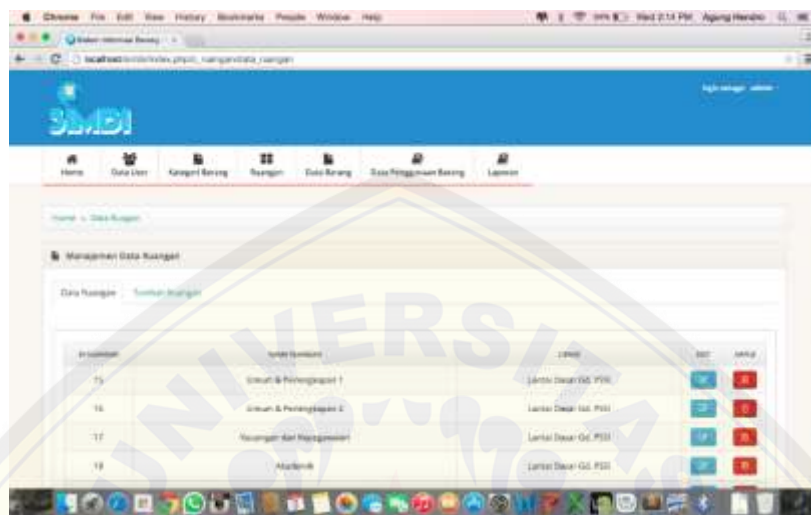
Menu kategori barang merupakan fitur manajemen data yang meliputi tambah kategori, edit kategori, dan hapus kategori. Dapat dilihat pada gambar 5.7.



Gambar 5. 7 Halaman Kategori Barang
Sumber: Hasil Penelitian, 2015

5.3.5 Ruang

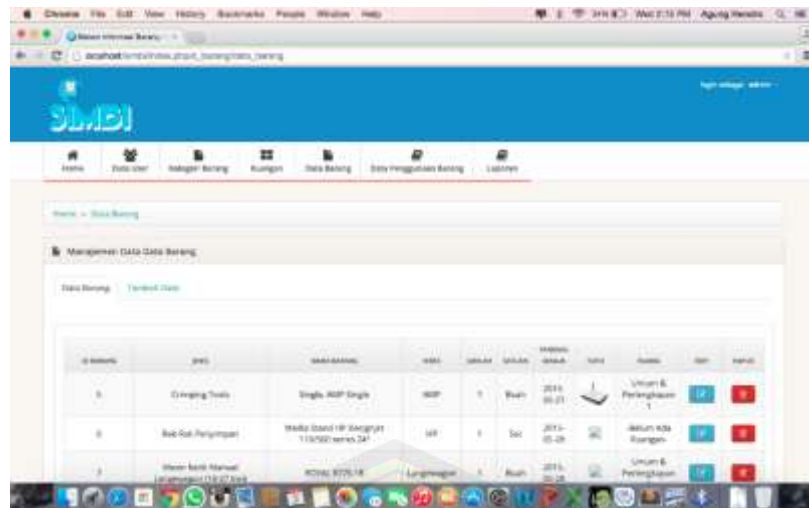
Menu ruangan merupakan fitur manajemen data ruangan yang meliputi tambah data ruangan, edit data ruangan, dan hapus data ruangan. Akun admin dan operator dapat mengolah menu ini, kecuali akun manajer yang hanya dapat melihat data ruangan yang sudah dimasukkan. Dapat dilihat pada gambar 5.8.



Gambar 5. 8 Halaman Ruang
Sumber: Hasil Penelitian, 2015

5.3.6 Data Barang

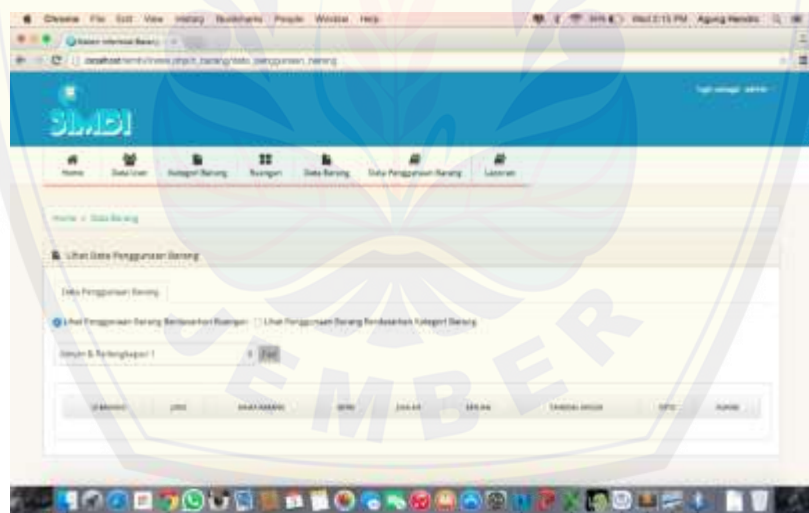
Menu data barang merupakan fitur manajemen data barang yang meliputi tambah data barang, edit data barang, dan hapus data barang. Akun admin dan operator dapat mengolah menu ini, kecuali akun manajer yang hanya dapat melihat data barang yang sudah dimasukkan. Dapat dilihat pada gambar 5.9.



Gambar 5. 9 Halaman Data Barang
Sumber: Hasil Penelitian, 2015

5.3.7 Data Penggunaan Barang

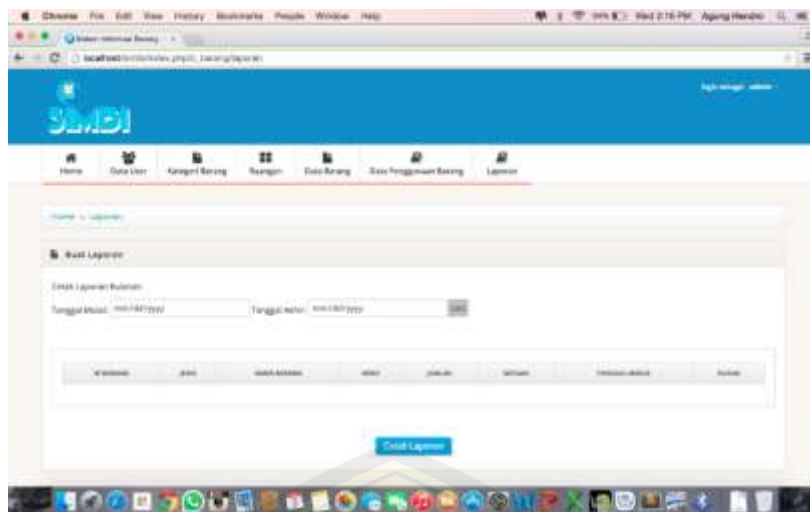
Menu data penggunaan barang merupakan fitur pencarian data penggunaan barang berdasarkan kategori dan ruangan. Dapat dilihat pada gambar 5.10.



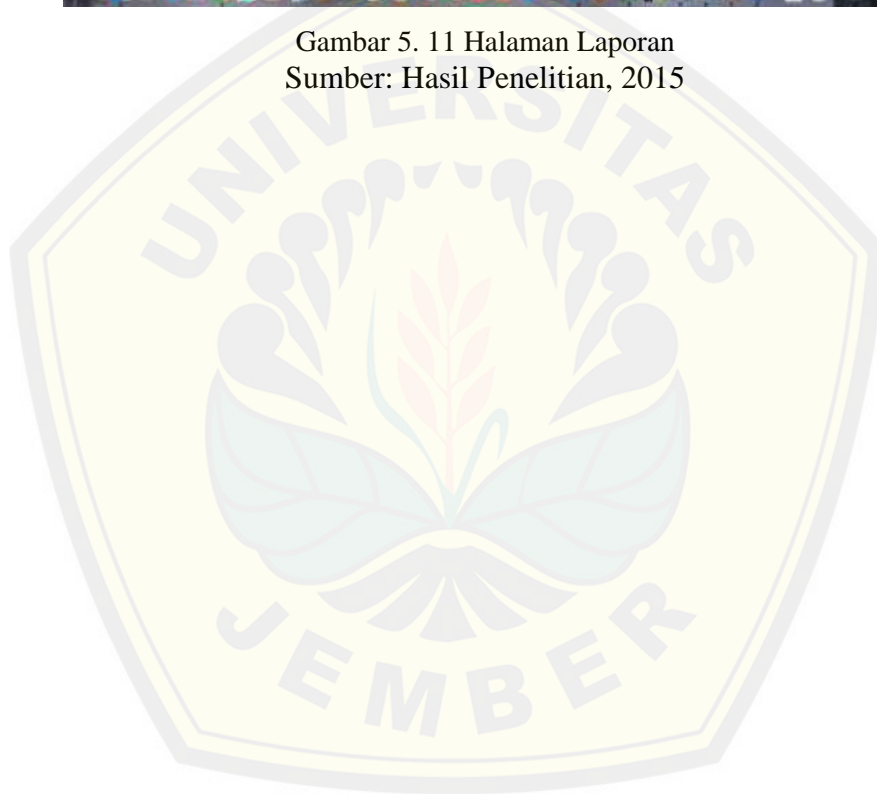
Gambar 5. 10 Halaman Data Penggunaan Barang
Sumber: Hasil Penelitian, 2015

5.3.8 Laporan

Menu laporan merupakan fitur untuk mengetahui hasil inventarisasi barang dan mencetaknya menjadi sebuah laporan yang dapat diolah lebih lanjut. Dapat dilihat pada gambar 5.11.



Gambar 5. 11 Halaman Laporan
Sumber: Hasil Penelitian, 2015



BAB 6. PENUTUP

6.1. Kesimpulan

Kesimpulan dari hasil penelitian yang telah dilakukan adalah sebagai berikut:

1. Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember ini dapat dirancang dan dibangun dengan menerapkan model *System Development Life Cycle* (SDLC), yakni *Waterfall Model*.
2. Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember ini dapat mempercepat kinerja petugas bagian sarana prasarana dalam menginventarisasi perlengkapan atau barang serta mempercepat proses pelaporan kepada pimpinan.
3. Sistem Informasi Manajemen Inventarisasi Barang Berbasis *Web* pada Program Studi Sistem Informasi Universitas Jember dibuat bersifat *intern* dan *multi user*, artinya pengguna sistem ini hanya kalangan tertentu yang memiliki hak akses terhadap sistem yaitu administrator, petugas inventarisasi, dan pimpinan.

6.2. Saran

Pengembangan sistem dapat dilakukan oleh peneliti selanjutnya dengan menambah fitur *capture image* supaya lebih aktual dalam mengisi kolom foto. Ditambah pengembangan untuk mampu berjalan diberbagai peramban yang ada pada sistem operasi *mobile* sehingga *user* mempunyai banyak pilihan dalam menggunakan peramban yang sesuai keinginan.

DAFTAR PUSTAKA

- Adeline, R. T. (2010). *Penggunaan Session Pada PHP*.
- Ali, M. (2007). *Inventarisasi Barang*. Jakarta: Bumi Aksara.
- Handoko, T. H. (1997). *Manajemen dan Sumber Daya Manusia*. Yogyakarta: Penerbit Liberty.
- Harianto, E. (2009). *Pengenalan UML, Suatu Bahasa Pemodelan Umum*.
- Henry, L. (1998). *Information System*.
- Indonesia. (2006). *Peraturan Pemerintah No. 6 Tahun 2006 Tentang Pengelolaan Barang Milik Negara/Daerah*. Jakarta: Direktorat Jendral Perbendaharaan.
- Indonesia. (2014). *Peraturan Pemerintah No. 27 Tahun 2014 Tentang Pengelolaan Barang Milik Negara/Daerah*. Jakarta: Direktorat Jendral Perbendaharaan.
- Jogiyanto. (1997). *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Penerbit ANDI.
- John, N. F. (1995). *Information System*.
- Kadir, A. (2003). *Pengenalan Sistem Informasi*. Yogyakarta: Penerbit ANDI.
- Kadir, A. (2009). *Dasar Perancangan dan Implementasi Database Relasional*. Yogyakarta: Penerbit ANDI.
- Kosasih, R. (2007). *Sistem Informasi Akuntansi*. Jakarta: Erlangga.
- Ladjamuddin, B, A.-B. (2006). *Rekayasa Perangkat Lunak*. Yogyakarta: GRAHA ILMU.
- Laudon, K. C., & Jane, P. L. (2000). *Manajemen Sistem Informasi, Organisasi, dan Teknologi pada Jaringan Enterprise*. Jakarta: Salemba Empat.
- McLeod, R. J. (2001). *Sistem Informasi*. Jakarta: Prehallindo.
- Mustakini, J. H. (2009). *Sistem Informasi Teknologi*. Yogyakarta: Andi Offset.
- Pressman. (2001). *Rekayasa Perangkat Lunak; Pendekatan Praktisi*. Boston: McGraw-Hill.
- Rosa, S. A., & Salahuddin, M. (2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Sugiana. (2009). *Inventarisasi BMN*. Yogyakarta: Penerbit ANDI.

Sugono, Dendy, & et al. (2008). *Kamus Besar Bahasa Indonesia*. Jakarta: Pusat Bahasa Departemen Pendidikan Nasional.

Susanto. (2004). *Sistem Informasi Manajemen Konsep dan Pengembanganya*.

Terry, G. R. (2006). *Prinsip-Prinsip Manajemen*. Jakarta: Bumi Aksara.

Universitas Jember. (2012). *Pedoman Penulisan Karya Tulis Ilmiah*. Jember: Jember University Press.



Lampiran 1. Form Pengujian

Pengujian sistem disesuaikan dengan kebutuhan yang telah didefinisikan pada tahap awal. Pengujian sistem dapat dilihat pada tabel di bawah ini:

| No | Fitur | Kasus | Hasil | Keterangan |
|----|--------|--|---|-------------------------------------|
| 1. | Login | <ul style="list-style-type: none"> Ketika <i>input username</i> dan <i>password benar</i> kemudian klik tombol “Login” <i>Login</i> sebagai admin : <i>Username</i> : admin <i>Password</i> : 1 <i>Login</i> sebagai manajer : <i>Username</i> : manajer <i>Password</i> : 2 <i>Login</i> sebagai operator: <i>Username</i> : operator <i>Password</i> : 3 | <ul style="list-style-type: none"> <i>Login</i> sukses dan masuk sesuai hak akses <i>Login</i> sukses dan masuk ke dashboard admin <i>Login</i> sukses dan masuk ke dashboard manajer <i>Login</i> sukses dan masuk ke dashboard operator | <p>[√] Berhasil [] Gagal</p> |
| | | <ul style="list-style-type: none"> Ketika <i>input username</i> dan <i>password salah</i> kemudian klik tombol “Login” Ketika <i>input username</i> dan <i>password kosong</i> kemudian klik tombol “Login” | <ul style="list-style-type: none"> Menampilkan pesan “Username/Password salah” Menampilkan Peringatan “field tidak boleh kosong” | <p>[√] Berhasil [] Gagal</p> |
| 2. | Logout | <ul style="list-style-type: none"> Klik menu <i>logout</i> | <ul style="list-style-type: none"> Menghapus session dan <i>logout</i> user | <p>[√] Berhasil [] Gagal</p> |
| 3. | User | <ul style="list-style-type: none"> Klik menu “Data User” | <ul style="list-style-type: none"> Menampilkan data user | |

| | | | | |
|----|--------|---|---|---|
| | | <ul style="list-style-type: none"> • Klik tab “Tambah User” • Ketika klik tombol “Simpan” dan data valid • Ketika klik tombol “Simpan” dan data kosong | <ul style="list-style-type: none"> • Menampilkan form tambah user • Menyimpan data <i>user</i> dan menampilkan halaman data <i>user</i> • Menampilkan peringatan “<i>fill the field</i>” | <p>[<input checked="" type="checkbox"/>] Berhasil [<input type="checkbox"/>] Gagal</p> <p>[<input checked="" type="checkbox"/>] Berhasil [<input type="checkbox"/>] Gagal</p> |
| | | <ul style="list-style-type: none"> • Klik <i>icon</i> hapus “sampah” pada barang yang akan dihapus • Klik “ok” • Klik “cancel” | <ul style="list-style-type: none"> • Menampilkan dialog “Hapus data?” • Menghapus user dan menampilkan data user • Kembali menampilkan data user | <p>[<input checked="" type="checkbox"/>] Berhasil [<input type="checkbox"/>] Gagal</p> |
| 4. | Barang | <ul style="list-style-type: none"> • Klik menu “Data Barang” • Klik tab “Tambah Barang” | <ul style="list-style-type: none"> • Menampilkan data barang • Menampilkan form tambah barang | <p>[<input checked="" type="checkbox"/>] Berhasil [<input type="checkbox"/>] Gagal</p> |
| | | <ul style="list-style-type: none"> • Klik <i>icon</i> ubah “pensil” pada barang yang akan diubah | <ul style="list-style-type: none"> • Menampilkan form yang berisi detail barang | <p>[<input checked="" type="checkbox"/>] Berhasil [<input type="checkbox"/>] Gagal</p> |
| | | <ul style="list-style-type: none"> • Ketika klik tombol “Simpan” dan data valid • Ketika klik tombol “Simpan” dan data kosong | <ul style="list-style-type: none"> • Menyimpan data barang dan menampilkan halaman data barang • Menampilkan peringatan “<i>fill the field</i>” | <p>[<input checked="" type="checkbox"/>] Berhasil [<input type="checkbox"/>] Gagal</p> |

| | | | | |
|----|-----------------|---|---|--|
| | | <ul style="list-style-type: none"> • Klik icon ubah “sampah” pada barang yang akan dihapus • Klik “ok” | <ul style="list-style-type: none"> • Menampilkan dialog “Hapus data?” • Menghapus barang dan menampilkan data barang | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Klik “cancel” | <ul style="list-style-type: none"> • Menampilkan halaman data barang | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| 4. | Kategori barang | <ul style="list-style-type: none"> • Klik menu “Kategori Barang” • Klik tab “Tambah Kategori” | <ul style="list-style-type: none"> • Menampilkan data kategori • Menampilkan form tambah kategori barang | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Klik icon ubah “pensil” pada kategori yang akan diubah | <ul style="list-style-type: none"> • Menampilkan form yang berisi detail kategori barang | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Ketika klik tombol “Simpan” dan data valid • Ketika klik tombol “Simpan” dan data kosong | <ul style="list-style-type: none"> • Menyimpan data kategori dan menampilkan halaman data kategori barang • Menampilkan peringatan “fill the field” | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Klik icon ubah “sampah” pada kategori yang akan dihapus • Klik “ok” | <ul style="list-style-type: none"> • Menampilkan dialog “Hapus data?” • Menghapus kategori dan menampilkan data kategori | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Klik “cancel” | <ul style="list-style-type: none"> • Menampilkan halaman data kategori barang | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |

| | | | | |
|--|---|---|---|--|
| 5. | Ruangan | <ul style="list-style-type: none"> • Klik menu “Ruangan” • Klik tab “Tambah Ruangan” | <ul style="list-style-type: none"> • Menampilkan data ruangan • Menampilkan form tambah ruangan | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Klik <i>icon</i> ubah “pensil” pada ruangan yang akan diubah | <ul style="list-style-type: none"> • Menampilkan form yang berisi detail ruangan | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Ketika klik tombol “Simpan” dan data valid • Ketika klik tombol “Simpan” dan data kosong | <ul style="list-style-type: none"> • Menyimpan data ruangan dan menampilkan halaman data ruangan • Menampilkan peringatan “<i>fill the field</i>” | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Klik <i>icon</i> ubah “sampah” pada ruangan yang akan dihapus • Klik “ok” | <ul style="list-style-type: none"> • Menampilkan dialog “Hapus data?” • Menghapus ruangan dan menampilkan data ruangan | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | <ul style="list-style-type: none"> • Klik “cancel” | <ul style="list-style-type: none"> • Menampilkan halaman data ruangan | <input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal |
| | | 6. | Cari Penggunaan Barang | <ul style="list-style-type: none"> • Klik menu “Data Penggunaan Barang” |
| <ul style="list-style-type: none"> • Pilih sesuai jenis pencarian | <ul style="list-style-type: none"> • Menampilkan daftar yang berisi data penggunaan barang | | | |

| | | | | |
|-----|---------|---|---|---|
| 77. | Laporan | <ul style="list-style-type: none"> • Klik menu “Laporan” | <ul style="list-style-type: none"> • Menampilkan halaman cetak laporan dengan pilihan tanggal awal akhir sesuai laporan yang ingin dicetak | |
| | | <ul style="list-style-type: none"> • Input tanggal awal dan akhir lalu klik “cari” • Klik “Cetak Laporan” | <ul style="list-style-type: none"> • Menampilkan daftar yang berisi data barang sesuai tanggal • Sistem mendownload file laporan berformat .xls | <p>[<input checked="" type="checkbox"/>] Berhasil</p> <p>[<input type="checkbox"/>] Gagal</p> |

