

**SISTEM INFORMASI MANAJEMEN *SUPPLIER* DAN STOK GUDANG
DI CV DUTA SQUARE BANDAR LAMPUNG**

(Skripsi)

Oleh

JANNATI ASRI SAFITRI



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2019**

ABSTRACT

MANAGEMENT INFORMATION SYSTEM OF SUPPLIER AND WAREHOUSE STOCK IN CV DUTA SQUARE BANDAR LAMPUNG

By

Jannati Asri Safitri

CV Duta Square is a one of companies in Lampung Province. CV Duta Square sells finished products (ready to use) consisting of Muslim women's clothing, men, children, and other items purchased from suppliers. The data record of suppliers and product in CV Duta Square is still done manually. A lot of data recorded manually will cause errors in writing, slipping, or even not being recorded. Restock of product will be sent by the supplier after the owner contact to purchase the item. However, some suppliers took self initiative to send product to Duta Square before the owner contacted. Delivery of product on the basis of the initiative will cause problems when the product sent have a lot of inventory because it will cause a buildup of stock of product. One solution to this problem is the creation of an information system that can record information of suppliers and product on CV Duta Square. Laravel is used as a framework in making this system. The Scrum method is used in the development stage and the system implementation stage. The system testing method in this study uses the Black-Box

Testing method. The information system suppliers and warehouse stock focuses on supplier data, item data, notifications to suppliers when their product are running low, and product purchase transactions from suppliers.

Kata Kunci: Supplier, stock of product, Scrum Method, Laravel.

ABSTRAK

SISTEM INFORMASI MANAJEMEN *SUPPLIER* DAN STOK GUDANG DI CV DUTA SQUARE BANDAR LAMPUNG

Oleh

Jannati Asri Safitri

CV Duta Square merupakan salah satu perusahaan dagang yang ada di Provinsi Lampung. CV Duta Square menjual produk (siap pakai) berupa pakaian muslim wanita, pria, anak-anak, dan barang lainnya yang dibeli dari *supplier*. Pendataan *supplier* dan barang di CV Duta Square masih dilakukan secara manual. Banyaknya data yang tercatat secara manual mudah untuk terjadi kesalahan dalam penulisan, terselip, atau bahkan tidak tercatat. *Restock* barang akan dikirimkan oleh *supplier* setelah *owner* menghubungi untuk melakukan pembelian barang. Namun, beberapa *supplier* berinisiatif sendiri mengirimkan barang ke Duta Square sebelum *owner* menghubungi. Pengiriman barang secara inisiatif akan menimbulkan masalah saat barang yang dikirimkan memiliki persediaan stok yang masih banyak karena akan menimbulkan penumpukan stok barang. Salah satu solusi untuk masalah tersebut yaitu dibuatnya sistem informasi yang dapat mendata *supplier* dan data barang di CV Duta Square. *Laravel* digunakan sebagai *framework* dalam pembuatan sistem ini. Metode *Scrum* digunakan dalam tahap

pengembangan dan tahap implementasi sistem. Metode pengujian sistem pada penelitian ini menggunakan metode *Black-Box Testing*. Sistem informasi *supplier* dan stok gudang berfokus pada data *supplier*, data barang, pemberitahuan untuk *supplier* saat barang mereka hampir habis, dan transaksi pembelian barang dari *supplier*.

Kata Kunci: *Supplier*, stok barang, Metode *Scrum*, *Laravel*.

**SISTEM INFORMASI MANAJEMEN *SUPPLIER* DAN STOK GUDANG
DI CV DUTA SQUARE BANDAR LAMPUNG**

Oleh

JANNATI ASRI SAFITRI

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar

SARJANA KOMPUTER

pada

Jurusan Ilmu Komputer

Fakultas Matematika dan Ilmu Pengetahuan Alam



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2019**

Judul Skripsi : **SISTEM INFORMASI MANAJEMEN
SUPPLIER DAN STOK GUDANG DI CV
DUTA SQUARE BANDAR LAMPUNG**

Nama Mahasiswa : **Jannati Asri Safitri**


Nomor Pokok Mahasiswa : **1517051184**

Jurusan : **Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

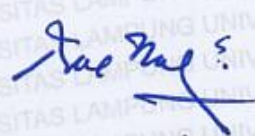
MENYETUJUI

1. **Komisi Pembimbing**



Astria Hijriani, S.Kom., M.Kom.
NIP. 19810308 200812 2 002

2. **Ketua Jurusan Ilmu Komputer**



Dr. Ir. Kurnia Muludi, M.S.Sc.
NIP. 19640616 198902 1 001

MENGESAHKAN

1. Tim Penguji

Ketua Tim Penguji : **Astria Hijriani, S.Kom., M.Kom.**

Penguji Pembahas : **Drs. Rd. Irwan Adi Pribadi, M.Kom.**

Penguji Pembahas : **Rico Andrian, S.Si., M.Kom.**

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Drs. Suratman, M.Sc.
NIP. 19640604 199003 1 002

Tanggal Lulus Ujian Skripsi : **26 November 2019**

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul **“Sistem Informasi Manajemen *Supplier* dan Stok Gudang di CV Duta Square Bandar Lampung”** merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti Kaidah Penulisan Karya Ilmiah Universitas Lampung. Apabila dikemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 26 November 2019



Jannati Asri Safitri
NPM. 1517051184

RIWAYAT HIDUP



Penulis dilahirkan pada tanggal 30 Januari 1998 di Bandar Lampung, sebagai anak pertama dari tiga bersaudara dengan Ayah bernama Sukaryanto dan Ibu Suharyati.

Penulis menyelesaikan pendidikan formal pertama kali di Taman Kanak-Kanak Beringin Raya pada tahun 2003. Pendidikan dasar di SDN 2 Sumberejo dan selesai pada tahun 2009. Pendidikan menengah pertama di SMPN 26 Bandar Lampung yang diselesaikan pada tahun 2012, kemudian melanjutkan ke pendidikan menengah atas di SMAN 7 Bandar Lampung dan lulus pada tahun 2015.

Pada tahun 2015 penulis terdaftar sebagai mahasiswa jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung dengan jalur SBMPTN. Selama menjadi mahasiswa beberapa kegiatan yang dilakukan penulis antara lain.

1. Menjadi anggota Abacus Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2015/2016.

2. Menjadi anggota Kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2016/2017.
3. Menjadi Asisten Dosen Jurusan Ilmu Komputer pada periode 2016/2017.
4. Pada bulan Januari 2018 penulis akan Kuliah Kerja Nyata (KKN) di Desa Harapan Jaya, Kecamatan Way Ratai, Kabupaten Pesawaran.
5. Pada bulan Juli 2018 penulis melaksanakan Kerja Praktik di PT. Jasa Raharja (Persero) Cabang Lampung.

PERSEMBAHAN

Puji dan syukur saya ucapkan kepada Allah SWT atas segala Rahmat-Nya sehingga saya dapat menyelesaikan skripsi ini.

Kupersembahkan karya ini kepada:

Teristimewa kedua orang tuaku, Papa dan Mama tercinta yang telah membesarkan, mendidik, memberikan doa, dan dukungannya. Trimakasih atas do'a dan segala yang telah diberikan untukku. Adik-adikku Bella dan Rayhan yang aku sayangi dan keluarga besar tercinta.

Teruntuk Dilar yang selalu memberikan dukungan padaku dalam menyelesaikan skripsi.

Teruntuk sahabat dan teman-temanku, terimakasih untuk canda tawa, tangis, dan perjuangan yang dilewati bersama-sama.

Keluarga Ilmu Komputer 2015,

Serta Almamater Tercinta, Universitas Lampung.

MOTTO

“Saat masalahmu terlalu berat untuk ditangani, jangan katakan “Wahai Allah, aku punya masalah besar” tapi katakan “Wahai masalah, aku punya Allah yang maha besar” (Ustadz Hanan Attaki)

“Keberhasilan yang paling manis adalah mencapai yang dikatakan oleh orang lain sebagai tidak mungkin” (Mario Teguh)

*“Bekerja keras adalah bagian dari fisikmu, bekerja cerdas merupakan bagian dari otakmu, sedangkan bekerja ikhlas adalah bagian dari hatimu”
(Abdullah Gymnastiar)*

“Jangan teralalu banyak mengeluh, syukuri hidupmu selagi kamu masih bisa bernafas” (Anonim)

SANWACANA

Puji syukur penulis panjatkan ke hadirat Allah SWT atas berkah, rahmat, dan hidayah-Nya penulis dapat menyelesaikan skripsi yang berjudul “**Sistem Informasi Manajemen *Supplier* dan Stok Gudang di CV Duta Square Bandar Lampung**” dengan baik dan lancar.

Penulis menyadari sepenuhnya bahwa terselesaikannya skripsi ini tidak terlepas dari bantuan berbagai pihak. Untuk itu penulis ingin mengucapkan terima kasih kepada semua pihak yang telah memberikan motivasi, bimbingan, dan saran yang sangat bermanfaat dari awal pelaksanaan sampai terselesaikannya laporan Skripsi ini. Dengan kerendahan hati, penulis mengucapkan terimakasih kepada:

1. Allah SWT yang telah memberi rahmat dan karunia-Nya sehingga penulis bisa menyelesaikan Skripsi dengan lancar.
2. Kedua orang tua tercinta, Bapak Sukaryanto dan Ibu Suharyati, adikku tercinta Aulia Bella Oktaviana dan Yusuf Rayhan Trianto, dan Keluarga besar yang selalu memberi doa, semangat, dukungan, dan kasih sayang yang tak terhingga.
3. Diar Rizky Dakasa Palambatra yang telah mendukung, menemani, dan menjadi penyemangat terbaik untuk penulis. Terima kasih untuk selalu

mengingatkan jangan banyak mengeluh dan selalu ikhlas.

4. Ibu Astria Hijriani, S.Kom., M.Kom. sebagai pembimbing utama yang telah membimbing penulis dan memberikan ide, kritik, serta saran sehingga penulisan skripsi ini dapat terselesaikan dengan baik. Selain itu juga sebagai pembimbing akademik penulis.
5. Bapak Drs. Rd. Irwan Adi Pribadi, M.Kom. sebagai Penguji Utama yang telah memberikan saran dan masukan yang bermanfaat dalam perbaikan skripsi ini.
6. Bapak Rico Andrian, S.Si., M.Kom. sebagai Penguji Kedua yang telah memberikan saran dan masukan yang bermanfaat dalam perbaikan skripsi ini.
7. Bapak Drs. Suratman, M.Sc. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
8. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
9. Bapak Didik Kurniawan, S.Si., M.T. selaku Sekretaris Jurusan Ilmu Komputer Universitas Lampung.
10. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung.
11. Ibu Ade Nora Maela, Bang Zainudin, Mas Noval, dan Mas Syam yang telah membantu segala urusan administrasi penulis di Jurusan Ilmu Komputer.
12. Tim seperjuangan Skripsi “Duta Square Team”, Fikri Ahmad Maulana sebagai ketua tim dan *scrum master*, Dina Munjiati, dan Elin Apriliana yang sama-sama berjuang dengan penulis dalam pengerjaan skripsi, terima kasih telah memberikan canda tawa, mengajarkan kerja sama, mengingatkan untuk selalu semangat, dan memberikan kisah yang akan dikenang dikemudian hari.

13. Sahabat-sahabatku Hanny Adelia Saputri, Dina Munjiati, Alinda Putri Chairunia, Ester Debora Priscila Silalahi, Wiwit Mudya Ningsih, Wenti Prisilia Novita dan Devi Maharani yang selalu menjadi *supporter* terbaik sepanjang menjalani perkuliahan. Kalian telah melewati canda tawa, tangis, dan mengukir kisah tentang perjuangan wal masuk perkuliahan hingga bisa mendapatkan gelar S.Kom yang tidak akan terlupakan oleh penulis.
14. Kak Dicky Kurniawan, Gandi Laksana Putra, dan M. Harun Ar Rasyid yang telah membantu penulis dalam pembuatan skripsi.
15. Keluarga KKN Desa Harapan Jaya kelompok 2, Agil Ratna Dila, Elisa Rahmawati Dewi, Hasril Mulya Budiman, Alvin Edo Lubis, Wildan Nuzwar, dan Firnando yang telah mengajari bagaimana hidup bermasyarakat dan selalu saling membantu dalam suka dan duka selama 40 hari tinggal bersama serta memberikan pengalaman tak terlupakan oleh penulis. Semoga ada kesempatan untuk kita bisa bertemu lagi.
16. Teman-teman Kerja Praktik Jasa Raharja Cabang Lampung Annisa (Universitas Teknokrat Indonesia), Ratih (Universitas Bandar Lampung), dan Kak Rani (Universitas Bandar Lampung) yang sama-sama menjalani tugas kerja praktik dengan baik.
17. Sahabat-sahabat sejak awal masuk SMA Desi Sukartini, Widya Agustina, dan Maya Annisa Nevanka yang selalu mendukung, memberi saran, masukan, do'a, dan semangat untuk penulis.
18. Sahabat-sahabat wag "ibu-ibu pejabat" Sefira Agista (Pido), Surya Cahya Annisa (Puyuh), Ayu Kartika Putri (Upil), dan Annisa Oktarina (Ipil) yang selalu menjadi penyemangat dan selalu memberikan canda tawa yang tak

pernah ada habisnya.

19. Teman-teman kelas C, Dinora Refiasari, Ayu Amalia, dan Novella Daria Utami.
20. Keluarga Ilmu Komputer Keluarga kasih atas bantuan dan motivasi yang telah diberikan.
21. Keluarga Ilmu Komputer 2015 yang tidak bisa penulis sebut satu persatu. Keluarga kedua penulis, rekan kelompok, rekan diskusi, rekan bercanda dan telah memberi arti dan warna serta pengalaman tak ternilai semasa duduk di bangku kuliah.
22. Seluruh kakak tingkat dan adik tingkat Ilmu Komputer yang tidak bisa disebutkan satu persatu.
23. Teman-teman Himakom yang sudah mengajari banyak hal dalam berorganisasi, memberikan banyak pengalaman, berjuang bersama memajukan Himakom, dan membawa nama baik jurusan Ilmu Komputer.
24. Teman-teman Asisten Dosen yang juga menjadi keluarga, mengajari banyak hal dalam berorganisasi, memberi banyak pengalaman, berjuang bersama memajukan Lab Ilmu Komputer.
25. Almamater Tercinta, Universitas Lampung yang telah memberikan Penulis kesempatan untuk menempuh pendidikan perkuliahan S1 dengan baik.

Bandar Lampung, 26 November 2019

Jannati Asri Safitri
NPM 1517051184

DAFTAR ISI

Halaman

DAFTAR TABEL	xx
---------------------------	-----------

DAFTAR GAMBAR.....	xxii
---------------------------	-------------

I. PENDAHULUAN	1
-----------------------------	----------

A. Latar Belakang.....	1
B. Rumusan Masalah	4
C. Batasan Masalah.....	4
D. Tujuan Penelitian.....	5
E. Manfaat Penelitian.....	5

II. TINJAUAN PUSTAKA.....	6
----------------------------------	----------

A. CV Duta Square.....	6
B. <i>Supplier</i>	7
C. Manajemen Gudang.....	7
D. <i>Safety Stock</i>	7
E. Sistem Informasi Manajemen (SIM)	8
F. Metode Pengembangan Sistem.....	8
G. Metode Pengujian Sistem	11
H. PHP	12
I. <i>Framework</i>	12
J. <i>Laravel</i>	12
K. <i>Composer</i>	13
L. <i>Database</i>	13
M. <i>MySQL</i>	13
N. <i>Unified Modeling Language (UML)</i>	14

III. METODE PENELITIAN	20
A. Tempat dan Waktu Penelitian	20
B. Alat dan Bahan	20
3.2.1. Metodologi Penelitian	22
1. Hubungan Antar Sistem.....	26
2. <i>User stories</i>	28
3. <i>Product Backlog</i>	29
C. Sumber Data	32
D. Metodologi Pengumpulan Data	32
E. Perancangan Sistem.....	33
1. <i>Use Case Diagram</i>	33
2. <i>Activity Diagram</i>	35
3. <i>Sequence Diagram</i>	55
4. <i>Class Diagram</i>	64
5. <i>Class Diagram Team</i>	65
6. <i>User interface</i>	67
7. Skenario Pengujian Sistem	74
IV. HASIL DAN PEMBAHASAN	81
A. Hasil.....	81
B. Pembahasan	83
1. Jadwal Pengerjaan Sistem.....	83
2. <i>Sprint Planning</i>	84
3. <i>Sprint</i>	91
C. Hasil Pengujian Sistem.....	120
V. KESIMPULAN DAN SARAN	129
A. Kesimpulan.....	129
B. Saran	130
DAFTAR PUSTAKA	131

DAFTAR TABEL

Tabel	Halaman
Tabel 1. Simbol-Simbol <i>Use Case Diagram</i>	15
Tabel 2. Simbol-Simbol <i>Activity Diagram</i>	16
Tabel 3. Simbol-Simbol <i>Class Diagram</i>	17
Tabel 4. Simbol-Simbol <i>Sequence Diagram</i>	18
Tabel 5. <i>Product Backlog</i>	29
Tabel 6. Daftar Pengujian Fungsional Sistem Informasi Manajemen <i>Supplier</i> dan Stok Gudang CV Duta Square untuk Peran <i>Admin</i> Gudang.	74
Tabel 7. Daftar Pengujian Fungsional Sistem Informasi Manajemen <i>Supplier</i> dan Stok Gudang CV Duta Square untuk Peran <i>Supplier</i>	80
Tabel 8. Daftar Jadwal Pengerjaan Sistem.	84
Tabel 9. <i>Sprint Planning</i> Sistem Informasi <i>Supplier</i> dan Stok Gudang	85
Tabel 10. Progres <i>Daily Scrum Sprint</i> ke-1	91
Tabel 11. Daftar Model Sistem Informasi SSG.	93
Tabel 12. Progres <i>Daily Scrum Sprint</i> ke-2	95
Tabel 13. Progres <i>Daily Scrum Sprint</i> ke-3	100
Tabel 14. Progres <i>Daily Scrum Sprint</i> ke-4	104
Tabel 15. Progres <i>Daily Scrum Sprint</i> ke-5	109
Tabel 16. Progres <i>Daily Scrum Sprint</i> ke-6	114

Tabel 17. Daftar Hasil Pengujian <i>Black-Box</i> untuk Peran <i>Admin</i>	120
Tabel 18. Daftar Hasil Pengujian <i>Black-Box</i> untuk Peran <i>Supplier</i>	127

DAFTAR GAMBAR

Gambar	Halaman
Gambar 1. Metode <i>Scrum</i>	9
Gambar 2. Alur Metode <i>Scrum</i>	25
Gambar 3. Hubungan Antar Sistem SSG, CRM, BID, dan SIA.	26
Gambar 4. <i>Use Case Diagram</i> Sistem <i>Supplier</i> dan Stok Gudang.	34
Gambar 5. <i>Activity Diagram</i> Mengakses <i>Supplier</i>	35
Gambar 6. <i>Activity Diagram</i> Mengubah <i>Supplier</i>	36
Gambar 7. <i>Activity Diagram</i> Menghapus <i>Supplier</i>	37
Gambar 8. <i>Activity Diagram</i> Menambah <i>Supplier</i>	38
Gambar 9. <i>Activity Diagram</i> Stok Gudang.	38
Gambar 10. <i>Activity Diagram</i> Mengakses Barang.	39
Gambar 11. <i>Activity Diagram</i> Mengubah Barang.	40
Gambar 12. <i>Activity Diagram</i> Menambah Barang.	40
Gambar 13. <i>Activity Diagram</i> Menghapus Barang.	41
Gambar 14. <i>Activity Diagram</i> Mengakses Rak Gudang.	42
Gambar 15. <i>Activity Diagram</i> Menambah Rak Gudang.	42
Gambar 16. <i>Activity Diagram</i> Mengubah Rak Gudang.	43
Gambar 17. <i>Activity Diagram</i> Menghapus Rak Gudang.	44
Gambar 18. <i>Activity Diagram</i> Mengakses Satuan Barang.	44

Gambar 19. <i>Activity Diagram</i> Menambah Satuan Barang.....	45
Gambar 20. <i>Activity Diagram</i> Menghapus Satuan Barang.	46
Gambar 21. <i>Activity Diagram</i> Mengakses Kategori Barang.....	46
Gambar 22. <i>Activity Diagram</i> Menambah Kategori Barang.....	47
Gambar 23. <i>Activity Diagram</i> Menghapus Kategori Barang.	48
Gambar 24. <i>Activity Diagram</i> Menambah Pengiriman Pembelian Barang.	49
Gambar 25. <i>Activity Diagram</i> Mengubah Pengiriman Pembelian Barang.	49
Gambar 26. <i>Activity Diagram</i> Menghapus Pengiriman Pembelian Barang.....	50
Gambar 27. <i>Activity Diagram</i> Menambah <i>Return</i> Barang.....	51
Gambar 28. <i>Activity Diagram</i> Mengubah <i>Return</i> Barang.....	52
Gambar 29. <i>Activity Diagram</i> Menghapus <i>Return</i> Barang.	53
Gambar 30. <i>Activity Diagram</i> Mengakses Notifikasi.	53
Gambar 31. <i>Activity Diagram</i> Menu <i>Department</i>	54
Gambar 32. <i>Activity Diagram</i> Menu <i>Pricing</i> Barang.	55
Gambar 33. <i>Sequence Diagram</i> <i>Supplier</i>	56
Gambar 34. <i>Sequence Diagram</i> Stok Gudang.	57
Gambar 35. <i>Sequence Diagram</i> Barang.....	58
Gambar 36. <i>Sequence Diagram</i> Rak Gudang.	59
Gambar 37. <i>Sequence Diagram</i> Satuan Barang.	60
Gambar 38. <i>Sequence Diagram</i> Kategori Barang.....	61
Gambar 39. <i>Sequence Diagram</i> Pengiriman Pembelian Barang.	62
Gambar 40. <i>Sequence Diagram</i> <i>Return</i> Barang.....	63
Gambar 41. <i>Sequence Diagram</i> Notifikasi.	64
Gambar 42. <i>Class Diagram</i>	65

Gambar 43. <i>Class Diagram Team</i>	66
Gambar 44. <i>User Interface Login</i>	67
Gambar 45. <i>User Interface Home</i>	68
Gambar 46. <i>User Interface Supplier</i>	68
Gambar 47. <i>User Interface Stok Gudang</i>	69
Gambar 48. <i>User Interface Barang</i>	69
Gambar 49. <i>User Interface Rak Gudang</i>	70
Gambar 50. <i>User Interface Satuan</i>	70
Gambar 51. <i>User Interface Kategori</i>	71
Gambar 52. <i>User Interface Pengiriman Pembelian Barang</i>	71
Gambar 53. <i>User Interface Return</i>	72
Gambar 54. <i>User Interface Home Supplier</i>	72
Gambar 55. <i>User Interface Barang- Supplier</i>	73
Gambar 56. <i>User Interface Notifikasi</i>	74
Gambar 57. Tampilan Halaman Menu <i>Supplier</i>	99
Gambar 58. Tampilan Halaman Jenis Kode Barang.....	102
Gambar 59. Tampilan Halaman Kategori Barang.....	102
Gambar 60. Tampilan Halaman Menu Barang.	103
Gambar 61. Tampilan Halaman Menu <i>Department</i>	108
Gambar 62. Tampilan Halaman Stok Gudang.	112
Gambar 63. Tampilan Halaman <i>Pricing</i> Barang.....	112
Gambar 64. Tampilan Halaman Satuan.	113
Gambar 65. Tampilan Halaman Rak Gudang.....	116
Gambar 67. Tampilan Halaman Faktur Pembelian Barang.	117

Gambar 68. Tampilan Halaman <i>Return</i> Barang.....	118
Gambar 69. Tampilan Halaman Menu Akun <i>Supplier</i>	118
Gambar 70. Tampilan Halaman Notifikasi.	119
Gambar 71. Tampilan Halaman Barang Tunda.	120

I. PENDAHULUAN

A. Latar Belakang

Indonesia memiliki perusahaan yang tersebar di berbagai provinsi mulai dari perusahaan kecil, menengah, hingga perusahaan besar. Badan Pusat Statistik Indonesia pada tahun 2015 mencatat sebanyak 3.695.195 perusahaan yang telah terdaftar di Indonesia (Badan Pusat Statistik, 2015). Data tersebut menunjukkan bahwa Indonesia memiliki banyak perusahaan. Jenis perusahaan yang ada di Indonesia yaitu perusahaan ekstraktif, industri atau manufaktur, agraris, jasa, dan dagang.

CV Duta Square merupakan salah satu perusahaan dagang di Provinsi Lampung yang menjual barang jadi (siap pakai) berupa pakaian muslim dewasa, anak-anak, dan barang lainnya yang dibeli dari *Supplier*. CV Duta Square memiliki kurang lebih 100 *supplier* yang informasinya masih didata secara manual. *Supplier* tersebut mengirimkan barang ke CV Duta Square. Barang yang telah dikirim *supplier*, biasanya dibayar oleh pemilik CV Duta Square kurang lebih per triwulan.

Barang yang telah sampai akan dicek oleh karyawan Duta Square. Barang yang sudah dicek selanjutnya didata secara manual di buku data barang. Banyaknya barang di Duta Square yang didata secara manual mudah untuk terjadi kesalahan penulisan, terselip, atau bahkan tidak terdata. Data barang yang tidak terdata atau terselip tersebut, menimbulkan kekeliruan pada stok barang. Pendataan stok barang di gudang yang tidak terdata dengan baik dapat memunculkan kesempatan bagi karyawan nakal untuk mengambil barang di gudang tanpa diketahui *owner*.

Persediaan stok barang yang habis akan dilakukan pembelian kembali oleh *owner*. *Restock* barang dikirimkan oleh *supplier* setelah *owner* menghubungi untuk melakukan pembelian barang. Pengiriman *restock* barang yang membutuhkan waktu beberapa hari perjalanan dapat merugikan perusahaan karena barang tersebut kosong untuk beberapa hari ke depan.

Masalah lainnya yaitu terdapat beberapa *supplier* yang berinisiatif mengirimkan barang ke Duta Square sebelum dihubungi oleh *owner*. *Supplier* tersebut berfikir, barang yang mereka kirimkan sebelumnya sudah habis di Duta Square. Masalah yang timbul karena hal tersebut yaitu akan terjadi penumpukan stok ketika barang yang dikirim masih banyak dan ditambah lagi dengan barang baru, padahal barang tersebut kurang diminati oleh pembeli.

Penelitian yang membahas tentang stok barang salah satunya yaitu “Sistem Informasi Inventori Gudang Untuk Mengontrol Persediaan Barang Pada Gudang Studi Kasus: PT Alaisys Sidoarjo”. Penelitian tersebut menjelaskan pencatatan barang PT Alaisys Sidoarjo masih secara manual menggunakan *Microsoft Office Excel*. Pencatatan secara manual menjadikan pihak kantor tidak dapat mengetahui data dari gudang dengan efisien dan akurat. Penelitian tersebut membuat dan menghasilkan sebuah sistem persediaan barang yang dapat mendata barang, transaksi, dan laporan barang. Sistem ini juga dapat membantu pihak perusahaan dalam mengawasi dan mengontrol barang di perusahaan tersebut (Agusvianto, 2017).

WebERP salah satu *software* yang datanya saling terintegrasi dan *open source*. WebERP memiliki pencatatan data *supplier*, *client*, keuangan, dan akuntansi. WebERP hanya memiliki akses untuk *admin* saja, sehingga *supplier* tidak dapat melihat jumlah stok barang mereka. Acosys merupakan contoh lain dari sistem yang datanya terintegrasi namun berbayar. Acosys memiliki pencatatan transaksi dan akuntansi yang cukup baik. Namun menurut pemilik CV Duta Square, Acosys belum memenuhi kebutuhan perusahaan secara spesifik sehingga tidak dapat diimplementasikan pada perusahaan.

Penelitian ini akan dilakukan pembuatan sistem informasi berbasis web untuk manajemen *supplier* dan stok gudang pada CV Duta Square. Sistem nantinya dapat membuat pendataan barang dan *supplier* lebih terdata dengan

baik sehingga CV Duta Square dapat mengontrol dan mengawasi data *supplier* dan data barang di gudang. Sistem ini juga akan memberitahukan *Supplier* saat stok barang mereka di Duta Square hampir habis, sehingga mereka bisa melakukan konfirmasi ke *owner* untuk melakukan *restock* barang atau tidak.

B. Rumusan Masalah

Rumusan masalah pada penelitian ini adalah bagaimana membangun sistem informasi yang dapat mendata *supplier*, mendata persediaan stok barang, mendata pengiriman barang dari *supplier* ke CV Duta Square, dan menginformasikan kepada *supplier* saat stok barang mereka hampir habis di Duta Square.

C. Batasan Masalah

Batasan Masalah dalam penelitian ini adalah sebagai berikut:

- a. sistem akan menangani manajemen *supplier* dan stok gudang,
- b. sistem akan menangani transaksi pengiriman pembelian barang,
- c. sistem akan memberikan notifikasi saat stok barang ke *supplier* saat stok barang mereka hampir habis,
- d. sistem akan terhubung dengan sistem informasi lainnya,
- e. sistem memakai *database* MySQL dan *framework* Laravel,
- f. memakai koneksi internet.

D. Tujuan Penelitian

Tujuan dari penelitian ini adalah terciptanya sistem informasi manajemen *supplier* dan stok gudang untuk CV Duta Square dengan menggunakan *framework* Laravel.

E. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

- a. *Supplier* CV Duta Square akan mendapatkan pemberitahuan saat stok barang mereka hampir habis,
- b. Perusahaan dapat mengontrol dan mengelola data *Supplier*,
- c. Perusahaan dapat mengelola data pengiriman barang dari *Supplier*,
- d. Perusahaan dapat mengontrol dan mengelola data barang.

II. TINJAUAN PUSTAKA

A. CV Duta Square

Duta square didirikan pada tahun 2004 oleh wirausahawan bernama Adrisman. Adrisman awalnya mendirikan *brand* Duta dengan menjual perhiasan dan busana muslim di CV Duta Square pertama yang berada di Simpur Center. Adrisman kemudian lebih fokus dalam bidang ini karena melihat pangsa pasar yang cukup besar dalam penjualan pakaian muslim.

Owner Duta pada tahun 2010 mencoba untuk membuka cabang di Bambu Kuning, dan tahun 2011 membuka cabang lagi di Jalan Wolter Monginsidi. *Owner* terus mengembangkan bisnisnya karena melihat respon pelanggan yang cukup baik dalam pembukaan CV Duta Square di luar *mall*. *Owner* pada tahun 2013 mencoba untuk membangun CV Duta Square di daerah Teuku Umar dengan *brand* Duta Square.

Owner Duta pada tahun 2016 mendaftarkan *brand* Duta Sqaure sebagai Perseroan Komanditor/ *Commanditaire Vennootschap* dengan nama CV Duta Square. Duta Square sekarang ini menyediakan pakaian muslim, kebutuhan *fashion* dan beberapa alat aksesoris *fashion*. CV Duta Square

memiliki 48 karyawan dalam pengoperasiannya dan berlokasi di Jalan Teuku Umar Bandar Lampung.

B. *Supplier*

Supplier adalah pihak yang menjadi penyedia barang yang dijual pada perusahaan/ toko, bisa disebut juga dengan distributor atau agen yang mendatangkan barang dagangan ke sebuah perusahaan/ toko. Perusahaan/ toko itu sendiri juga dapat bertindak sebagai *supplier*, misalnya saja ada sebagian produk yang dijual adalah produk rumahan buatan sendiri (Nugroho, 2014).

C. *Manajemen Gudang*

Gudang sebagai fasilitas khusus yang bersifat tetap, yang dirancang untuk mencapai target tingkat pelayanan dengan total biaya yang paling rendah. Manajemen pergudangan dirancang bertujuan untuk mengontrol kegiatan pergudangan yang diharapkan terjadi pengurangan biaya-biaya yang ada di dalam gudang, pengambilan dan pemasukan barang ke gudang yang efektif dan efisien, serta kemudahan dan keakuratan informasi stok barang di gudang (Richards, 2011).

D. *Safety Stock*

Persediaan pengaman (*safety stock*) adalah jumlah persediaan material minimum (*minimum stock*) yang harus dimiliki tiap periode oleh perusahaan

untuk menjaga kemungkinan keterlambatan datangnya material dan kemungkinan fluktuasi permintaan (Erlina, 2008).

E. Sistem Informasi Manajemen (SIM)

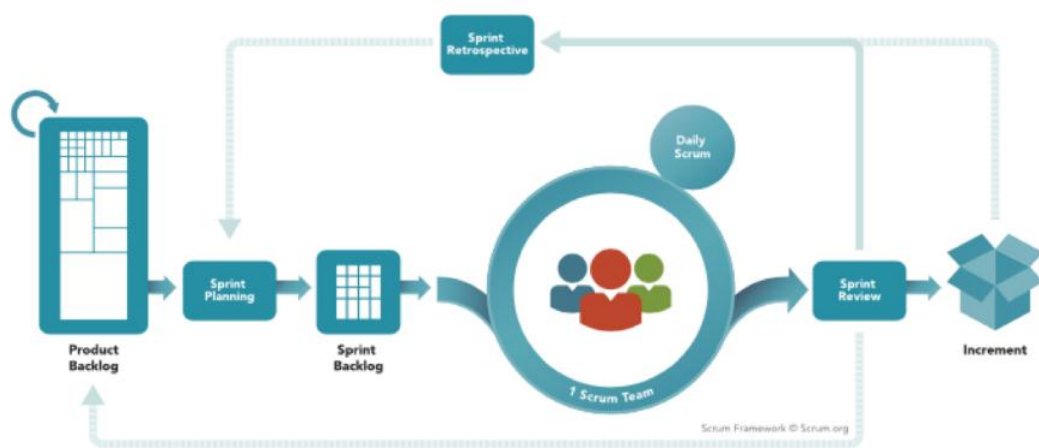
Sistem Informasi Manajemen (SIM) adalah suatu sistem berbasis komputer yang membuat informasi tersedia bagi para pengguna yang memiliki kebutuhan serupa. Pengguna SIM biasanya terdiri atas entitas-entitas organisasi formal-perusahaan atau sub unit anak perusahaannya. Informasi yang diberikan oleh SIM menjelaskan perusahaan atau salah satu sistem utamanya dilihat dari apa yang telah terjadi di masa lalu, apa yang sedang terjadi, dan apa yang kemungkinan akan terjadi di masa depan (Jr & Schell, 2009).

F. Metode Pengembangan Sistem

Scrum merupakan metode atau kerangka kerja yang diciptakan oleh Ken Schwaber and Jeff Sutherland. *Scrum* merupakan metode ataupun kerangka kerja untuk mengembangkan dan mengelola suatu produk yang kompleks. *Scrum* akan menghasilkan suatu produk dengan nilai setinggi mungkin secara produktif dan kreatif (Schwaber & Sutherland, 2017).

Scrum salah satu metode yang dapat dipakai dalam sebuah tim yang akan membangun sistem. *Scrum* dapat dipakai oleh sebuah tim kecil maupun tim besar. Tim yang memakai *Scrum* akan berkolaborasi dan saling berinteraksi melalui arsitektur pengembangan dan target rilis produk yang bagus. *Scrum*

telah banyak digunakan untuk mengembangkan perangkat lunak, perangkat keras, perangkat lunak terintegrasi, aplikasi dalam jaringan yang saling berinteraksi, dan sebagainya. *Scrum* cukup mampu dalam menghadapi kompleksitas dan efektif dalam transfer pengetahuan secara berkala serta berkelanjutan. Metode *Scrum* yang dipakai dalam penelitian ini, digambarkan pada Gambar 1.



Gambar 1. Metode *Scrum*.

Sumber: (<https://www.Scrum.org/resources/what-is-Scrum>).

Scrum memiliki beberapa tahapan-tahapan yang harus dilakukan, tahapan-tahapan tersebut yaitu:

1. *Product Backlog*

Product Backlog merupakan tahapan dimana seluruh tim membuat fitur yang akan digunakan dalam sistem masing-masing. Semua kebutuhan dianalisis pada tahapan ini mulai dari menganalisis kebutuhan *software* dan *hardware*, sampai dengan menganalisis jalannya sistem yang akan dipakai.

2. *Sprint Planning*

Sprint planning adalah tahapan ketika semua kebutuhan sudah dianalisis maka dilanjutkan dengan berdiskusi mengenai perencanaan dimana anggota *team* menentukan bagian-bagian yang akan mereka kerjakan dan menentukan berapa jam/ waktu yang dibutuhkan tiap anggota untuk melaksanakan pekerjaan pada tiap fiturnya.

3. *Sprint Backlog*

Sprint backlog merupakan tahapan yang dilakukan setelah anggota selesai berdiskusi mengenai tugas dari setiap anggota *scrum*. Tahap ini akan menampilkan seluruh target yang akan dicapai dalam satu *sprint*, yang nantinya akan dipantau peningkatannya.

4. *Daily Scrum*

Daily Scrum merupakan proses yang dilakukan oleh tim pengembang saat *sprint* berlangsung. Setiap harinya tim pengembang akan membahas pekerjaan yang telah dilakukan dan pekerjaan selanjutnya yang harus dikerjakan saat proses *sprint* berlangsung. Setiap pertemuan akan berfokus pada setiap *sprint goal* seperti hal yang dilakukan untuk membantu tim pengembang dalam penyelesaian *sprint goal* atau hal yang akan menjadi hambatan untuk tim pengembang atau diri sendiri dalam penyelesaian *sprint goal*.

5. *Sprint Review*

Sprint Review merupakan tahap untuk meninjau hasil yang telah dikerjakan pada tahapan *Sprint*. Pada tahap ini dilakukan diskusi bersama dan hasil dari diskusi selanjutnya akan ditentukan proses yang dilakukan selanjutnya untuk meningkatkan nilai bisnis.

6. *Increment*

Increment adalah hasil dari *Product Backlog* yang diselesaikan dalam setiap *sprint*. Saat masuk proses ini kondisi produk yang dikerjakan harus masuk kategori “selesai” bagi tim pengembang. Pada tahapan ini produk harus bisa digunakan walau belum dirilis oleh tim pengembang (Schwaber & Sutherland, 2017).

G. Metode Pengujian Sistem

Black-box testing adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi. Metode ujicoba *Black-box* memfokuskan pada keperluan fungsional dari *software*. *Black-Box testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program (Hidayat & Muttaqin, 2018).

H. PHP

PHP adalah bahasa *skript* yang dapat ditanamkan atau disisipkan kedalam HTML. PHP banyak dipakai untuk memprogram situs *web* dinamis. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya dan paling mudah karena memiliki referensi yang banyak (Hidayat & Muttaqin, 2018).

I. *Framework*

Framework adalah kumpulan perintah atau fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi *website*, kita harus mengikuti aturan dari *framework* tersebut (Gasan, Metandi, & Utomo, 2018)

J. *Laravel*

Laravel adalah sebuah MVC *web development framework* berbasis PHP. Metode pengembangan MVC (*Model*, *View*, *Controller*) menggunakan bahasa pemrograman PHP dan kerangka kerja *Laravel*. *Model* berisi representasi basis data, *view* berisi tampilan *website*, dan *controller* berisi proses data dari *view* diolah. Hasil implementasi berupa berkas *.php.

Laravel didesain untuk meningkatkan kualitas dari *software* dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman kerja dengan aplikasi yang berdasarkan *syntax*

murni dan satu set dari *function* yang akan menghemat waktu pengimplementasian. *Laravel* sangat menarik karena apapun bisa ditulis dalam dan bisa di kumpulkan menjadi sebuah *library* yang sederhana untuk seluruh *web* aplikasi (Gasani, Metandi, & Utomo, 2018).

K. *Composer*

Composer adalah package manager PHP yang digunakan untuk menginstal *Laravel* beserta semua *dependency*-nya. *Composer* terinspirasi dari NPM yaitu *Node.js package manager*, dan *Bundler* yang merupakan *package manager* Ruby (Basuki, 2016).

L. *Database*

Database adalah himpunan kelompok data (arsip) yang saling berhubungan dan diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah. *Database* dapat disebut juga kumpulan *file/ table/ arsip* yang saling berhubungan dan disimpan dalam media penyimpanan (Muhammad Aswadi, 2015).

M. *MySQL*

MySQL merupakan salah satu *relational database management system* bersifat *open source*. Struktur *database* disimpan dalam tabel-tabel yang saling berelasi. *MySQL* dapat dipergunakan dan didistribusikan baik untuk kepentingan individu maupun *corporate* secara gratis, tanpa memerlukan

lisensi dari pembuatnya. *MySQL* dapat dijalankan dalam berbagai *platform* sistem operasi antara lain *Windows*, *Linux*, *Unix*, *Sun OS*, dan lain-lain (Trisen & Putra, 2018).

N. *Unified Modeling Language (UML)*

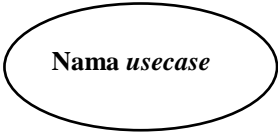



Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung (Sukamto & Shalahuddin, 2014).

UML memiliki beberapa diagram-diagram pemodelan yaitu:

1. *Use Case Diagram*

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Sukamto & Shalahuddin, 2014). Tabel 1 akan memperlihatkan simbol-simbol yang ada pada diagram *use case*.






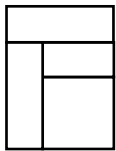
Tabel 1. Simbol-Simbol *Use Case Diagram*.

No	Simbol	Deskripsi
1.	<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor.
2.	Aktor/ <i>actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.
3.	Asosiasi/ <i>association</i> 	Komunikasi antara aktor & <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	Exstensi/ <i>extend</i> 	<i>Use case</i> yang ditambahkan dapat berdiri sendiri, mirip dengan prinsip inheritance pada pemrograman berorientasi objek.

2. *Activity Diagram*

Diagram aktivitas atau *Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Pembuatan diagram aktivitas harus diawali dengan status awal dan diakhiri oleh status akhir (Sukamto & Shalahuddin, 2014). Tabel 2 akan memperlihatkan simbol-simbol yang ada pada *Activity Diagram*.

Tabel 2. Simbol-Simbol *Activity Diagram*.

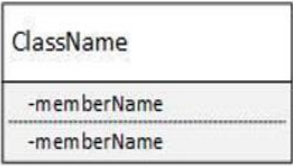






No	Simbol	Deskripsi
1.	Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status akhir 	Status akhir yang dilakukan oleh sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

3. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan method atau operasi. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas. Operasi atau method adalah fungsi-fungsi yang dimiliki oleh

suatu kelas (Sukamto & Shalahuddin, 2014). Tabel 3 akan memperlihatkan simbol-simbol yang ada pada *class diagram*.

Tabel 3. Simbol-Simbol *Class Diagram*.

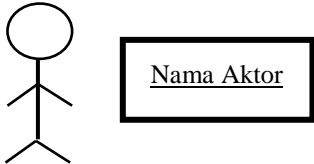

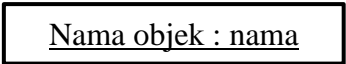

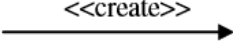
No	Simbol	Deskripsi
1.	Kelas/ Class 	Kelas pada struktur sistem.
2.	Antarmuka/ Interface 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	Asosiasi/ Association 	Relasi antar kelas dengan makna umum, biasanya juga disertai dengan <i>multiplicity</i> .
4.	Asosiasi berarah/ Directed Association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.	Kebergantungan/dependency 	Relasi antar kelas dengan makna kebergantungan antar kelas.
7.	Agregasi/ Aggregation 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

4. *Sequece Diagram*

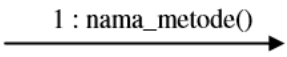
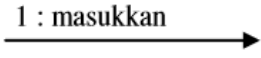
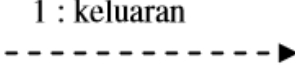
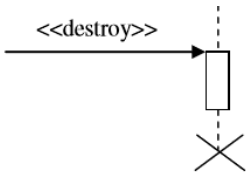
Sequence Diagram menggambarkan perilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan *message* yang dikirimkan dan diterima antar objek. *Sequence Diagram* harus mengetahui skenario dan objek-objek yang terlibat dalam sebuah *use*. Banyaknya *Sequence Diagram* yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang dimiliki (Sukanto & Shalahuddin, 2014).

Tabel 4 akan memperlihatkan simbol-simbol yang ada pada *Sequence Diagram*.

Tabel 4. Simbol-Simbol *Sequence Diagram*.

No	Simbol	Deskripsi
1.	Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.
2.	Garis hidup/ <i>lifeline</i> 	Menyatakan kehidupan suatu objek.
3.	Objek 	Menyatakan objek yang berinteraksi pesan.
4.	Waktu Aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi.
5.	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat

Tabel 4. Simbol-Simbol *Sequence Diagram* (Lanjutan).

No	Simbol	Deskripsi
6.	Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri.
7.	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data/ masukkan/ informasi ke objek lainnya.
8.	Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu.
9.	Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada <i>create</i> maka ada <i>destroy</i> .

III. METODE PENELITIAN

A. Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berlokasi di Jl. Soemantri Brojonegoro No.1, Gedung Meneng, Bandar Lampung dan di CV Duta Square yang berlokasi di Jl. Teuku Umar No.34, Pasir Gantung, Tanjung Karang Pusat, Kota Bandar Lampung. Penelitian ini mulai dilaksanakan pada Semester Ganjil Tahun Ajaran 2018/ 2019.

B. Alat dan Bahan

Penelitian ini dilakukan dengan menggunakan alat untuk mendukung dan menunjang pelaksanaan penelitian, antara lain:

1. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam pengembangan sistem informasi ini adalah satu unit *Printer* dan satu unit *Laptop* dengan spesifikasi:

- a. Processor: AMD A10-7300 Radeon R6, 10 Compute Cores 4C+6G
1.90 GHz,
- b. Display: 14 inci, FHD 1920 x 1080 pixels,

- c. VGA: AMD R7 M255 2GB,
- d. RAM: 4GB SSD, 8 eMMC,
- e. Storage: 1TB HDD,
- f. System type: Windows 10 Pro 64-bit (10.0, Build 17134).

2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam pengembangan sistem informasi ini yaitu:

- a. Sistem Operasi Windows 10,
- b. Beberapa aplikasi yaitu:
 - XAMPP versi 7.1.10 sebagai aplikasi local server untuk pengembangan sistem MYSQL dan PHP,
 - Web Browser (Google Chrome dan Microosoft Edge) sebagai aplikasi untuk menguji dan menjalankan sistem,
 - Balsamic Mockup untuk membuat desain *interface*,
 - Star UML sebagai aplikasi pembuat diagram model sistem informasi,
 - Atom Text Editor versi 1.25.1 sebagai aplikasi untuk menulis *code* program.
- c. Beberapa web aplikasi yaitu:
 - API sebagai pertukaran data antar sistem,
 - Draw.io sebagai aplikasi untuk membuat diagram.
- d. *Framework* yang digunakan yaitu:

- Laravel sebagai *framework* untuk mengatur logika dan kerja sistem dengan bahasa pemrograman PHP,
- Bootstrap sebagai *framework* berbasis *css* yang mengatur elemen *html* dan mempercantik tampilan *website*,
- Vue.js *framework* berbasis *javascript* untuk membuat aplikasi menjadi dinamis dan interaktif.

3.2.1. Metodologi Penelitian

Metode yang digunakan dalam Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square adalah Metode *Scrum*. Metode *Scrum* memiliki beberapa tahapan-tahapan dalam pengerjaannya yaitu:

1. *Product Backlog*

Product backlog akan dilakukan oleh tim dengan membuat daftar fitur yang diinginkan oleh *product owner*. Deskripsi *backlog* dapat dilihat pada Tabel 5.

2. *Sprint Planning*

Sprint planning dilakukan oleh seluruh tim, dan juga terkadang *product owner* ikut serta dalam *sprint planning*. *Team* kami akan melakukan *sprint planning* setiap 2 hari sekali.

3. *Sprint Backlog*

Sprint backlog dilakukan saat *sprint planning* telah dilaksanakan dan dipantau oleh *Scrum master*. Setiap *backlog* yang ingin dilaksanakan pada

tahap ini akan dimasukkan di kolom *To Do* di Tabel 5.

4. Daily Scrum

Daily Scrum pada tahapan ini, tim akan melaksanakan *backlog* yang telah direncanakan. Tim akan lebih fokus untuk mendapatkan *sprint goal*.

5. Sprint Review

Sprint review dilakukan oleh tim dengan meninjau hasil dari *sprint* yang dilakukan. Tahap *Sprint review* ini juga akan menentukan apakah *backlog* yang dikerjakan masuk ke *increment* atau kembali ke tahap *sprint planning*.

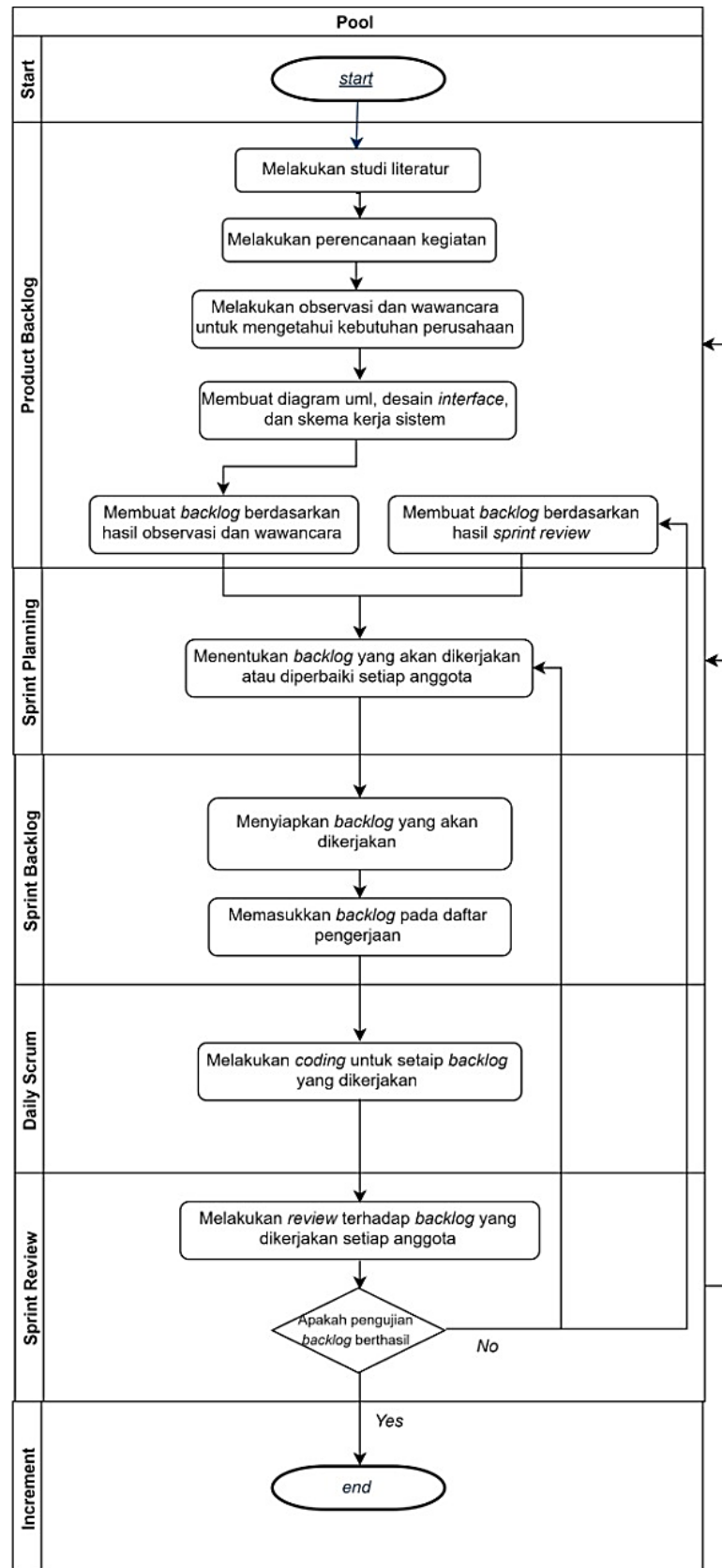
6. Increment

Proses *increment* memiliki *output* yang telah dikerjakan oleh tim dan disetujui oleh *Scrum master*. *Output* dari tahapan ini dapat dijadikan acuan untuk mengerjakan *product backlog* yang berkaitan.

Scrum merupakan salah satu metode yang dapat digunakan dalam sebuah tim yang akan membangun sistem baik tim kecil maupun tim besar. Gambar 2 akan memperlihatkan alur kerja Metode *Scrum* yang dilakukan oleh *Scrum Team*. *Scrum Team* memiliki pembagian tugas yaitu:

- a. Yumilia Harianti sebagai *Product Owner*,
- b. Fikri Ahmad Maulana sebagai *Scrum Master*,

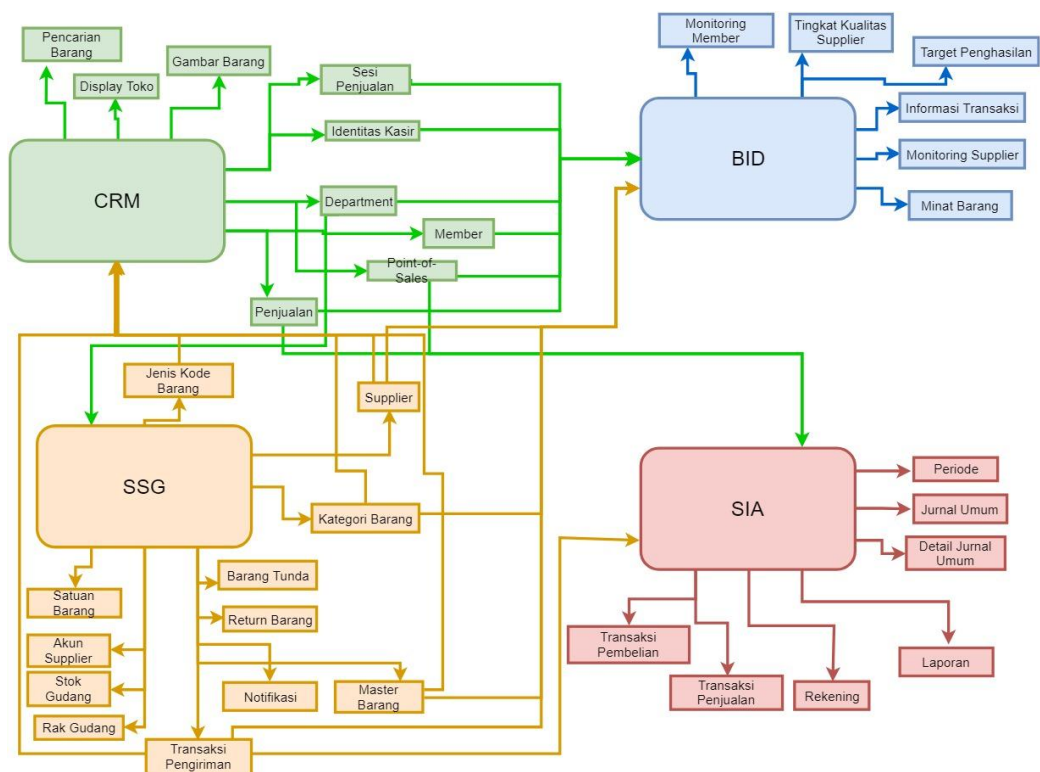
- c. Elin Aprilliana, Jannati Asri Safitri, dan Dina Munjiati sebagai *Scrum team*.



Gambar 2. Alur Metode Scrum.

1. Hubungan Antar Sistem

Sistem SSG (*Supplier dan Stok Gudang*) dalam pengembangannya akan terhubung ke Sistem CRM (*Customer Relationship Management*), Sistem BID (*Bussiness Intelegence Dashboard*), dan Sistem SIA (Sistem Informasi Akuntansi). Sistem- sistem tersebut saling terhubung karena terjadi proses menerima dan mengirim data antar sistem. Proses pengiriman dan penerimaan data tersebut dilakukan dengan menggunakan API (*Application Programming Interface*). API dapat digunakan atau diakses oleh semua sistem dengan cara mengakses *route* API yang telah dibuat dimasing-masing sistem. Hubungan antar sistem diperlihatkan pada Gambar 3.



Gambar 3. Hubungan Antar Sistem SSG, CRM, BID, dan SIA.

Gambar 3 menunjukkan hubungan antar sistem yang saling mengirim dan menerima data dari satu sistem ke sistem lainnya. Data yang dikirim dan diterima diataranya sebagai berikut.

1. Sistem Infomasi *Supplier* dan Stok Gudang (SSG)

- Mengirim data ke sistem CRM berupa data *supplier*, transaksi pengiriman, kode jenis barang, master barang, kategori, dan *return* barang.
- Mengirim data ke sistem BID berupa data *supplier*, transaksi pengiriman, kode jenis barang, master barang, kategori, dan *return* barang.
- Mengirim data ke sistem SIA berupa data transaksi pengiriman dan master barang.
- Menerima data *department* dari sistem CRM.

2. Sistem Informasi *Customer Relationship Management* (CRM)

- Mengirim data ke sistem BID berupa data *point of sale*, penjualan, *member*, *department*, identitas kasir, dan sesi penjualan.
- Mengirim data ke sistem SIA berupa data penjualan.
- Mengirim data ke sistem SSG berupa data *department*.
- Menerima data *supplier*, transaksi pengiriman, kode jenis barang, master barang, kategori, dan *return* barang dari sistem SSG.

3. Sistem Informasi *Bussiness Intelegence Dashboard* (BID)

- Menerima data *supplier*, transaksi pengiriman, kode jenis barang, master barang, kategori, dan *return* barang dari sistem SSG.

- Menerima data *point of sale*, penjualan, *member*, *department*, identitas kasir, dan sesi penjualan dari sistem CRM.

4. Sistem Informasi Akuntansi (SIA)

- Menerima data transaksi pengiriman dari sistem SSG.
- Menerima data penjualan dari sistem CRM.

2. *User stories*

User stories dalam metode *Scrum* digunakan untuk membuat *backlog* yang mana harus ditentukan oleh *Product Owner*. *User stories* berisi nama pengguna sistem, fitur-fitur yang menjadi kebutuhan sistem dan tujuan dari fitur yang direncanakan. *User stories* dibuat menggunakan bahasa yang mudah dimengerti secara umum. Salah satu hal yang paling penting dengan adanya *User stories* ini ialah agar pengembangan *software* dapat terealisasi sesuai dengan keinginan pengguna.

User Stories Owner

- a. Sebagai seorang *owner*, saya ingin adanya pendataan *Supplier* secara sistematis karena di perusahaan saya pendataan *Supplier* masih secara manual menggunakan buku.
- b. Sebagai seorang *owner*, saya ingin adanya pendataan barang secara sistematis mulai dari data kategori barang, satuan barang, rak barang, jenis barang, dan *pricing* barang.
- c. Sebagai seorang *owner*, saya ingin adanya pendataan transaksi pembelian barang dari *supplier* ke perusahaan karena di perusahaan

saya belum ada pendataan transaksi saat membeli barang dari *supplier*.

- d. Sebagai seorang *owner*, saya ingin adanya pendataan *return*/pengembalian barang dari perusahaan ke *supplier* karena di perusahaan saya belum ada pendataan barang yang di-*return*.
- e. Sebagai seorang *owner*, saya ingin *supplier* mengetahui saat stok barang mereka hanya tersisa sedikit di perusahaan agar stok kosong di perusahaan dapat dinimalisir.
- f. Sebagai seorang *owner*, saya ingin *supplier* mengetahui daftar barang mereka yang tidak perlu mereka kirimkan ke perusahaan.

3. Product Backlog

Pada pelaksanaannya *Scrum* memiliki daftar tugas atau *product backlog*. *Product Backlog* dibuat berdasarkan wawancara dari *Product Owner* yang sebelumnya telah didapatkan dari *user stories*. Rincian *backlog* diperlihatkan pada Tabel 5.

Tabel 5. *Product Backlog*

Pembuat	Backlog	To Do	In Progres	Testing	Done
Fikri <i>Customer Relationship Management (CRM)</i>	Konfigurasi Sistem A. <i>Upload Sistem</i> B. Konfigurasi VPN C. Pembuatan Data <i>Dummy</i> D. Pembuatan <i>Database</i> E. Pembuatan Model				

Tabel 5. *Product Backlog* (Lanjutan).

Pembuat	Backlog	To Do	In Progres	Testing	Done
	F. Pembuatan API Modul Sistem G. <i>Department</i> Identitas kasir H. <i>Point-of-Sales</i> I. Penjualan J. <i>Member</i> K. <i>Supplier</i> L. Master barang M. <i>Display</i> toko N. Kategori barang O. Gambar barang P. Pencaraian barang				
Elin <i>Business Intelligence Dashboard (BID)</i>	Konfigurasi Sistem A. <i>Upload</i> Sistem B. Konfigurasi VPN C. Pembuatan Data <i>Dummy</i> D. Pembuatan <i>Database</i> E. Pembuatan Model Modul Sistem F. Target Penghasilan G. Informasi Transaksi H. Tingkat kualitas <i>supplier</i> I. <i>Monitoring supplier</i> J. Minat Barang K. <i>Monitoring member</i> L. Data CRM M. Data SSG				
Jannati Supplier dan Stok Gudang (SSG)	Konfigurasi Sistem A. <i>Upload</i> Sistem B. Konfigurasi VPN C. Pembuatan Data				

Tabel 5. *Product Backlog* (Lanjutan).

Pembuat	Backlog	To do	In Progress	Testing	Done
	<i>Dummy</i> D. Pembuatan <i>Database</i> E. Pembuatan Model Modul Sistem F. <i>Supplier</i> G. Akun <i>Supplier</i> H. <i>Department</i> I. Stok Gudang J. Master Barang K. <i>Pricing</i> L. Jenis Kode Barang M. Rak Gudang N. Satuan Barang O. Kategori Barang P. Transaksi Pengiriman Q. <i>Return</i> Barang R. Notifikasi S. Barang Tunda				
Dina Sistem Informasi Akuntansi (SIA)	Konfigurasi Sistem A. <i>Upload</i> Sistem B. Konfigurasi VPN C. Pembuatan Data <i>Dummy</i> D. Pembuatan <i>Database</i> E. Pembuatan Model Modul Sistem F. Rekening G. Periode H. Transaksi Pembelian I. Transaksi Penjualan J. Jurnal Umum K. Detail Jurnal Umum L. Laporan				

C. Sumber Data

Sumber data yang akan digunakan dalam penelitian ini terbagi menjadi dua yaitu sumber data primer dan sumber data sekunder.

1. Sumber Data Primer

Sumber data primer merupakan sumber data yang diperoleh langsung dari sumber asli (tidak melalui media perantara). Sumber data primer yang didapat untuk melakukan penelitian ini adalah langsung dari pemilik CV Duta Square Bandar Lampung.

2. Sumber Data Sekunder

Data sekunder merupakan sumber data penelitian yang diperoleh secara tidak langsung melalui media perantara. Sumber data sekunder yang didapatkan untuk melakukan penelitian ini berupa catatan tentang *supplier* maupun bukti-bukti transaksi dan sebagainya.

D. Metodologi Pengumpulan Data

Metode pengumpulan data yang dipakai dalam penelitian ini yaitu wawancara dan observasi.

1. Wawancara

Wawancara merupakan salah satu metode pengumpulan data yang digunakan untuk memperoleh informasi dari narasumber. Wawancara dilakukan dengan berinteraksi atau memberikan pertanyaan kepada narasumber terkait dengan informasi yang dibutuhkan dalam melakukan

penelitian. Tim kami telah melakukan wawancara sebanyak 4 kali. Wawancara pertama mengenai sejarah Duta Square, wawancara kedua mengenai masalah yang terjadi pada Duta Square, wawancara ketiga tentang apa yang dibutuhkan oleh Duta Square, dan wawancara keempat melakukan validasi apa yang dibutuhkan oleh Duta Square.

2. Observasi

Observasi merupakan salah satu teknik pengumpulan data. Peneliti akan melakukan pengamatan secara langsung ke objek penelitian untuk melihat dari dekat kegiatan apa saja yang dilakukan dan mendapatkan informasi yang akan dibutuhkan untuk melancarkan penelitian tersebut. Tim kami telah melakukan observasi sebanyak 3 kali. Observasi pertama dilakukan untuk mengetahui proses bisnis perusahaan, observasi kedua dan ketiga dilakukan untuk mendapatkan *backlog* sistem.

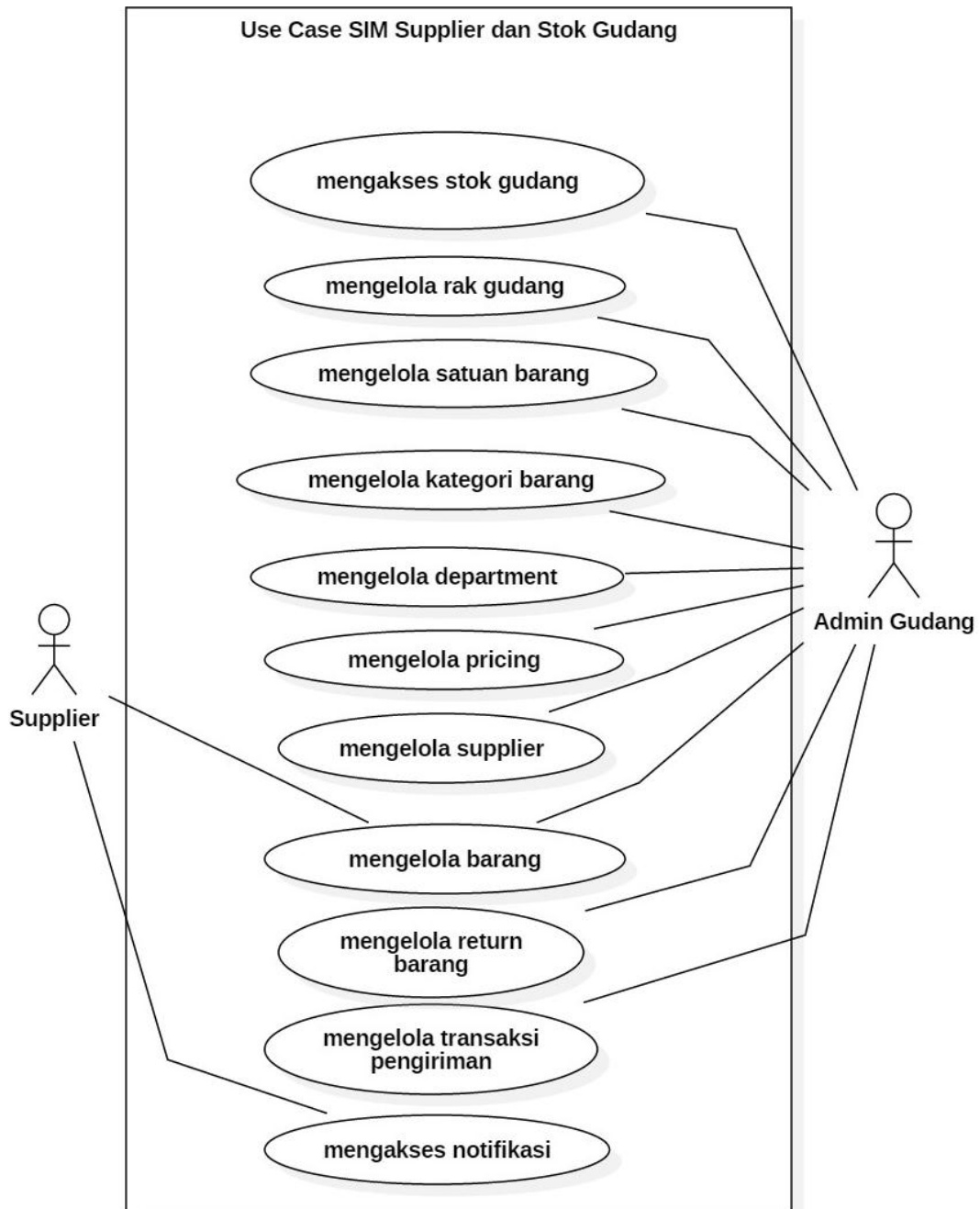
E. Perancangan Sistem

Perancangan sistem akan meliputi *Usecase Diagram*, *Activity Diagram*, *Class Diagram*, dan *Interface Diagram*.

1. Use Case Diagram

Gambar 4 memperlihatkan *use case diagram* yang menunjukkan bahwa terdapat dua level *user* yaitu *Admin Gudang* dan *supplier*. *Admin Gudang* dapat mengakses stok gudang, mengelola barang, rak gudang,

satuan barang, kategori barang, transaksi pengiriman pembelian barang, *return* barang, dan mengelola *supplier*. *Supplier* akan dapat mengakses barang dan mengakses notifikasi.

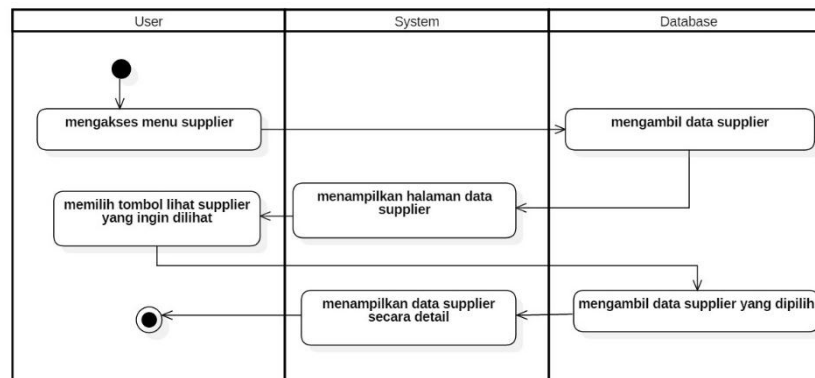


Gambar 4. *Use Case Diagram* Sistem *Supplier* dan Stok Gudang.

2. Activity Diagram

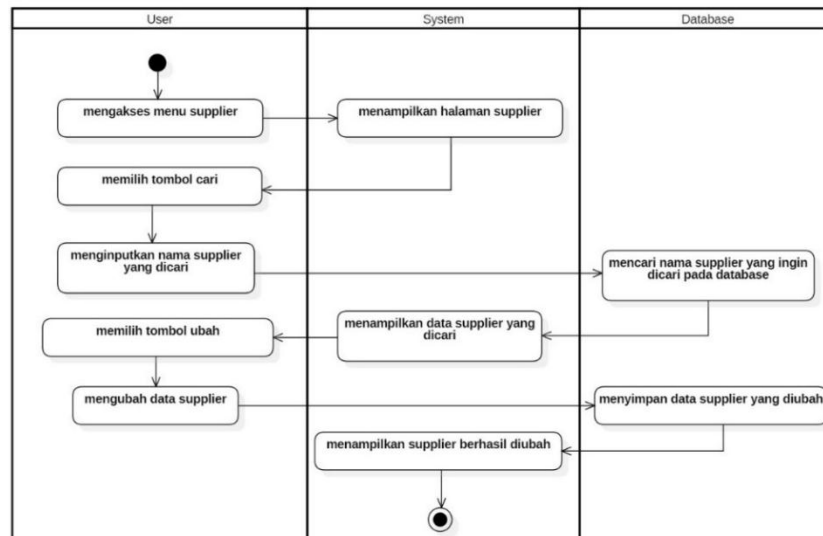
2.1. Activity Diagram Mengelola Supplier

Activity Diagram mengelola *supplier* terbagi menjadi 4 *Activity Diagram* yaitu menambah *supplier*, mengubah data *supplier*, menghapus *supplier*, dan mengakses data *supplier*. Gambar 5 menerangkan bahwa *user* dapat mengakses menu *supplier* dengan memilih menu *supplier*. Sistem kemudian menampilkan informasi *supplier* yang ada di *database supplier*.



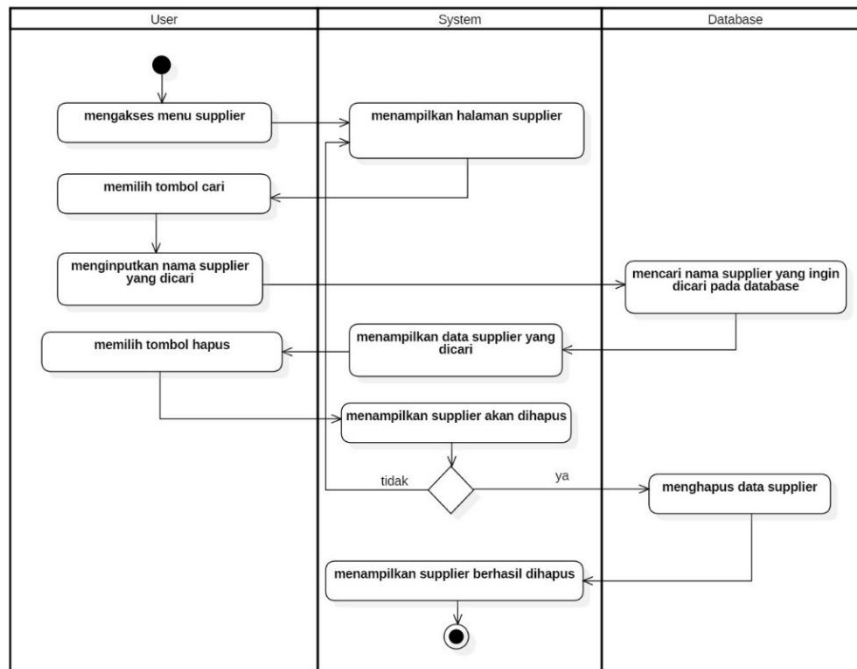
Gambar 5. Activity Diagram Mengakses Supplier.

Activity Diagram mengubah *supplier* akan diperlihatkan pada Gambar 6. *User* dapat mengubah data *supplier* dengan mencari nama *supplier* yang akan diubah datanya, selanjutnya sistem akan menampilkan data *supplier* yang dicari. *User* kemudian mengubah data *supplier* yang akan disimpan kedalam *database supplier*.



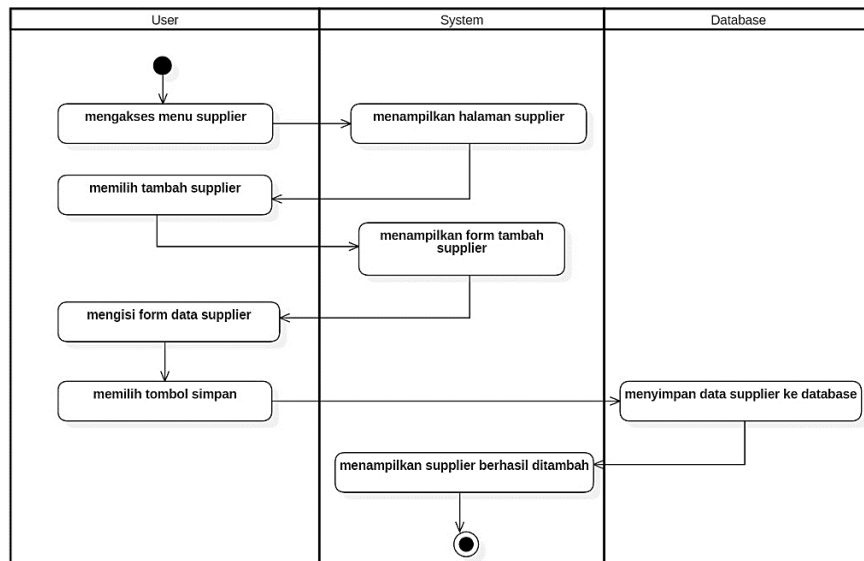
Gambar 6. *Activity Diagram Mengubah Supplier.*

Activity Diagram menghapus *supplier* akan diperlihatkan pada Gambar 7 yang menerangkan bahwa *user* dapat menghapus data *supplier* dengan mencari nama *supplier* yang akan dihapus datanya, selanjutnya sistem akan menampilkan data *supplier* yang akan dihapus. *User* kemudian akan memilih tombol hapus dan data *supplier* akan terhapus dari *database supplier*.



Gambar 7. Activity Diagram Menghapus Supplier.

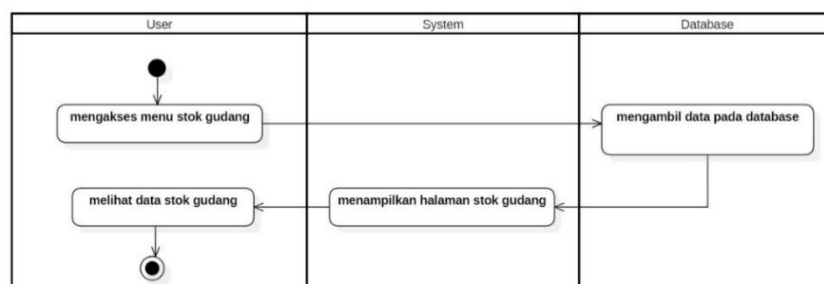
Gambar 8 memperlihatkan Activity Diagram menambah *supplier* bahwa *user* dapat menambah *supplier* baru dengan memilih tombol tambah *supplier*, selanjutnya sistem akan menampilkan halaman form tambah *supplier*. *User* kemudian akan mengisi data-data *supplier* yang akan disimpan kedalam *database supplier*.



Gambar 8. Activity Diagram Menambah Supplier.

2.2. Activity Diagram Mengakses Stok Gudang

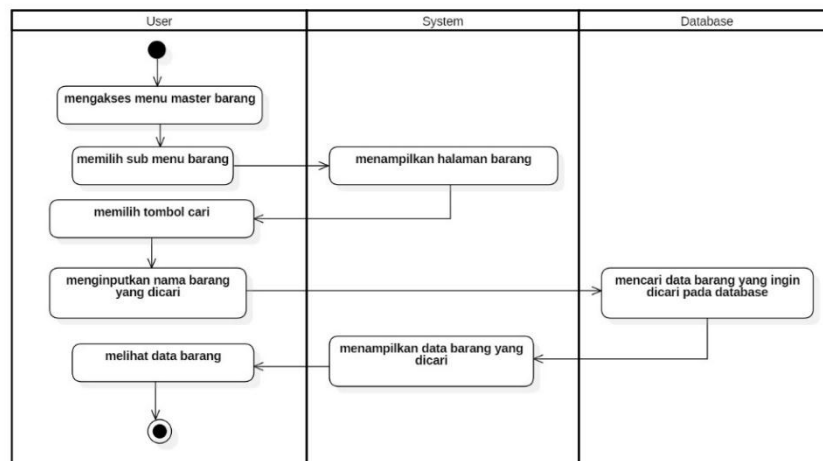
Activity Diagram mengelola stok gudang akan diperlihatkan pada Gambar 9, yang menerangkan bahwa *user* dapat mengakses menu stok gudang dengan memilih menu stok gudang. Sistem kemudian menampilkan informasi jumlah stok barang yang tersedia pada gudang yang ada di *database* stok gudang.



Gambar 9. Activity Diagram Stok Gudang.

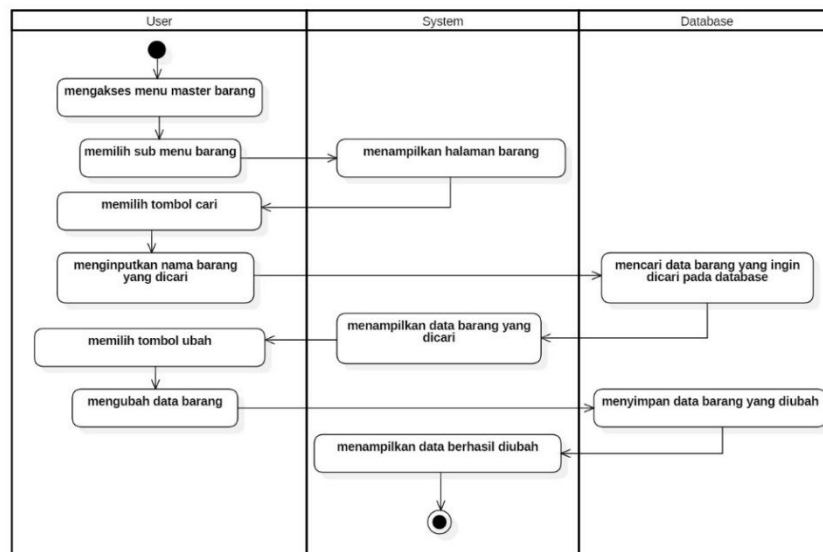
2.3. Activity Diagram Mengelola Barang

Activity Diagram mengelola barang terbagi menjadi 4 *Activity Diagram* yaitu mengakses barang, menambah barang, mengubah data barang, dan menghapus data barang. Gambar 10 memperlihatkan bahwa *user* dapat mengakses barang dengan memilih menu master barang lalu memilih sub menu barang. Sistem kemudian menampilkan informasi data barang yang ada di *database* barang.



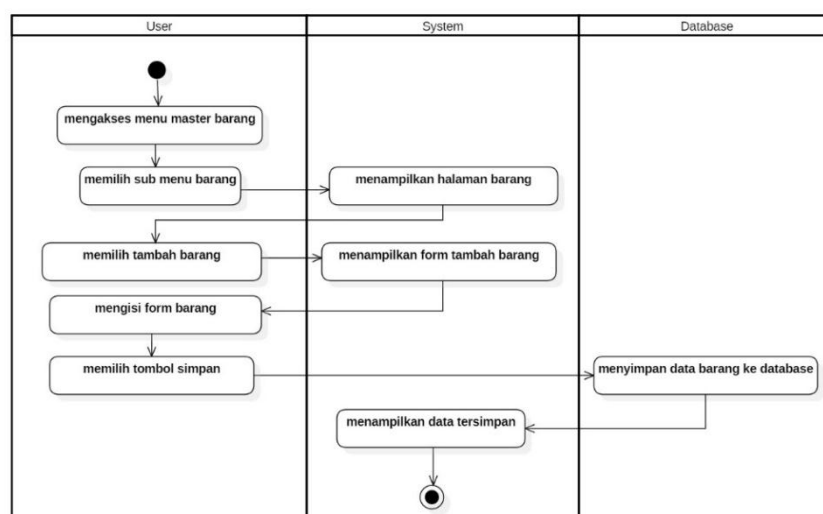
Gambar 10. *Activity Diagram* Mengakses Barang.

Activity Diagram mengubah barang diperlihatkan pada Gambar 11 yang menjelaskan *user* dapat mengubah data barang dengan mencari nama barang yang akan diubah datanya, selanjutnya sistem akan menampilkan data barang yang akan diubah datanya. *User* kemudian akan mengubah data barang dan akan disimpan kedalam *database* barang.



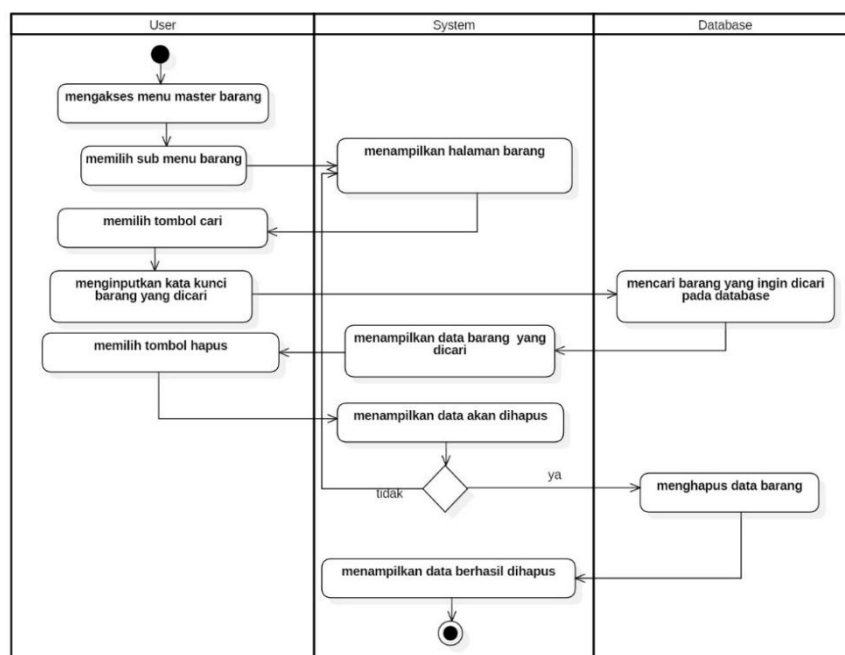
Gambar 11. *Activity Diagram Mengubah Barang.*

Activity Diagram menambah barang diperlihatkan pada Gambar 12 yang menjelaskan *user* dapat menambah data barang dengan dengan memilih tombol tambah barang, selanjutnya sistem akan menampilkan halaman form tambah barang. *User* kemudian akan mengisi data barang yang akan disimpan kedalam *database* barang.



Gambar 12. *Activity Diagram Menambah Barang.*

Activity Diagram menghapus barang diperlihatkan pada Gambar 13 yang menjelaskan *user* dapat menghapus data barang dengan mencari nama barang yang akan dihapus datanya, selanjutnya sistem akan menampilkan data barang yang akan dihapus. *User* kemudian akan memilih tombol hapus dan data barang akan terhapus dari *database* barang.

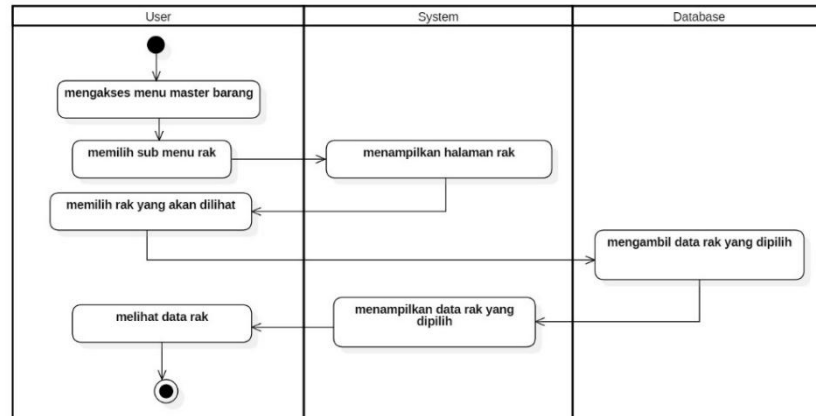


Gambar 13. *Activity Diagram* Menghapus Barang.

2.4. *Activity Diagram* Mengelola Rak Gudang

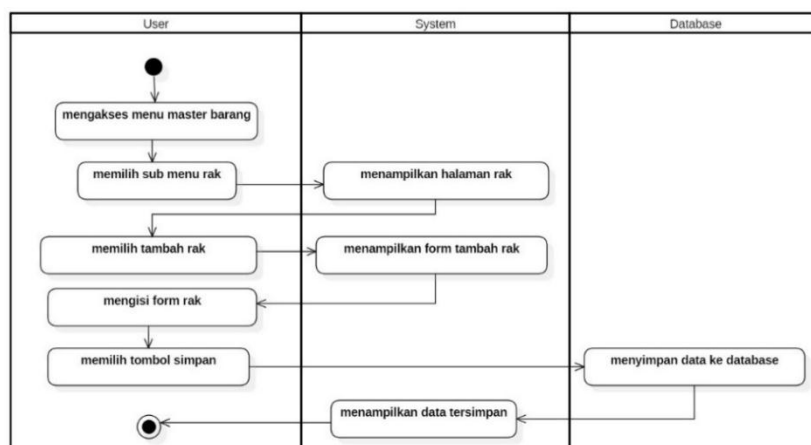
Activity Diagram mengelola rak gudang terbagi menjadi 4 *Activity Diagram* yaitu mengakses rak gudang, menambah rak gudang, mengubah data rak gudang, dan menghapus rak gudang. Gambar 14 memperlihatkan bahwa *user* dapat mengakses rak gudang dengan memilih menu master barang lalu memilih sub menu rak

gudang. Sistem kemudian menampilkan informasi data rak gudang yang ada di *database* rak gudang.



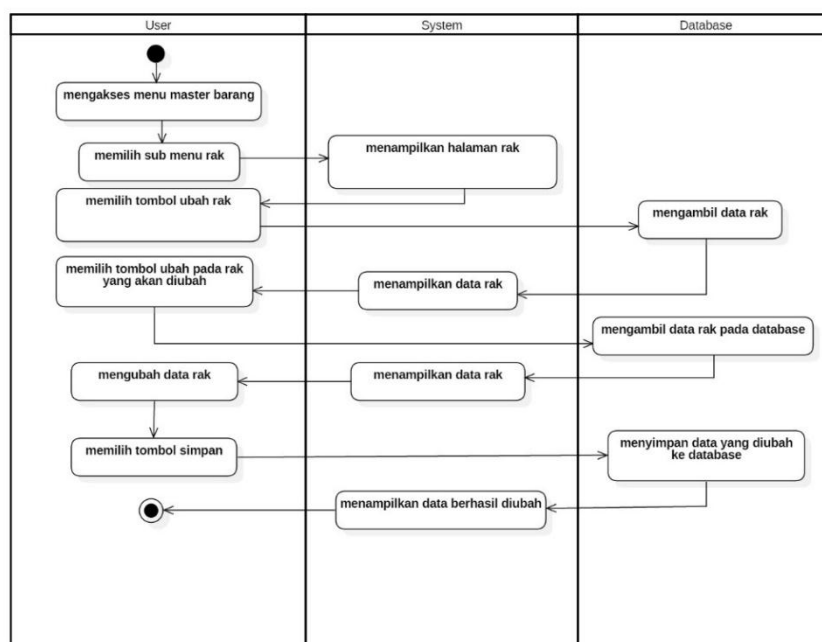
Gambar 14. *Activity Diagram* Mengakses Rak Gudang.

Activity Diagram menambah rak gudang diperlihatkan pada Gambar 15 yang menjelaskan *user* dapat menambah data rak gudang dengan dengan memilih tombol tambah rak gudang, selanjutnya sistem akan menampilkan halaman form tambah rak gudang. *User* kemudian akan mengisi data rak gudang yang akan disimpan kedalam *database* rak gudang.



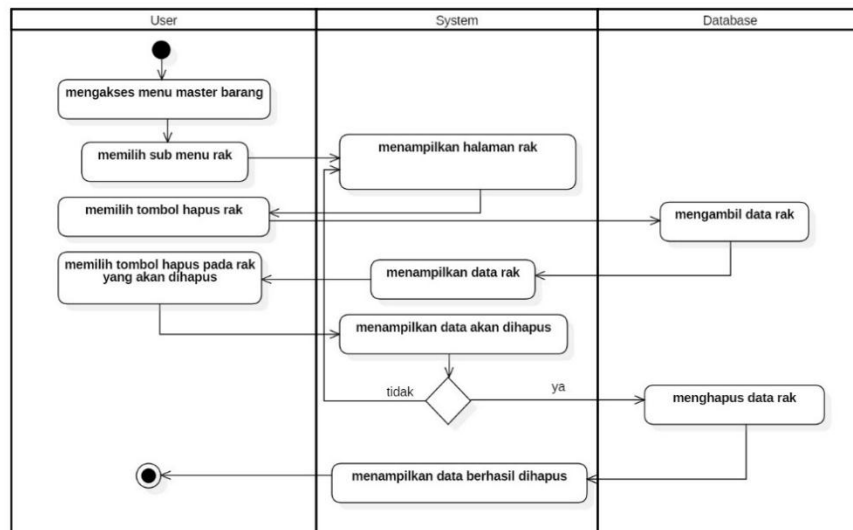
Gambar 15. *Activity Diagram* Menambah Rak Gudang.

Activity Diagram mengubah rak gudang diperlihatkan pada Gambar 16 yang menjelaskan, *user* dapat mengubah data rak gudang dengan mencari nama rak gudang yang akan diubah datanya selanjutnya, sistem akan menampilkan data rak gudang yang akan diubah datanya. *User* kemudian akan mengubah data rak gudang dan akan disimpan kedalam *database* rak gudang.



Gambar 16. Activity Diagram Mengubah Rak Gudang.

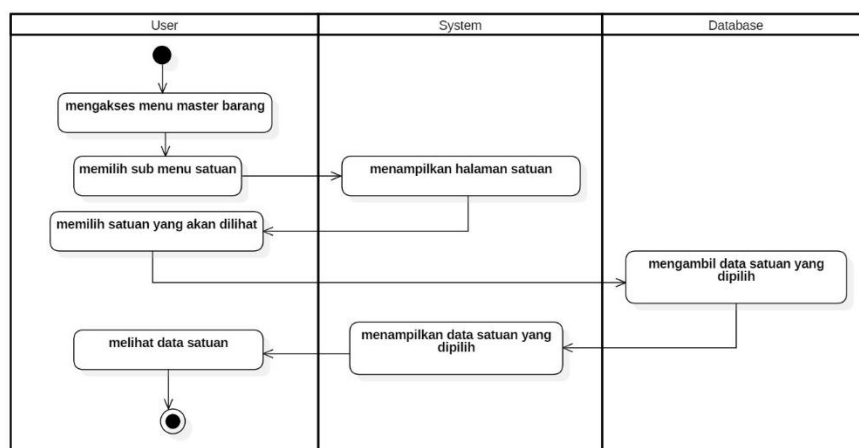
Activity Diagram menghapus rak gudang diperlihatkan pada Gambar 17 yang menjelaskan *user* dapat menghapus data rak gudang dengan mencari nama rak gudang yang akan dihapus datanya, selanjutnya sistem akan menampilkan data rak gudang yang akan dihapus. *User* kemudian akan memilih tombol hapus dan data rak gudang akan terhapus dari *database* rak gudang.



Gambar 17. *Activity Diagram* Menghapus Rak Gudang.

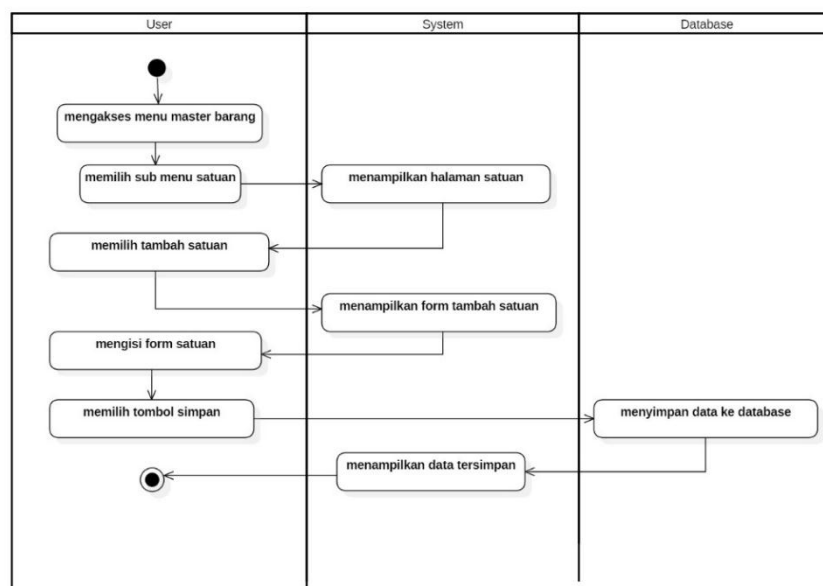
2.5. *Activity Diagram* Mengelola Satuan Barang

Gambar 18 memperlihatkan *Activity Diagram* mengakses satuan barang. *User* dapat mengakses satuan barang dengan memilih menu master barang lalu memilih sub menu satuan barang. Sistem kemudian menampilkan informasi data satuan barang yang ada di *database* satuan.



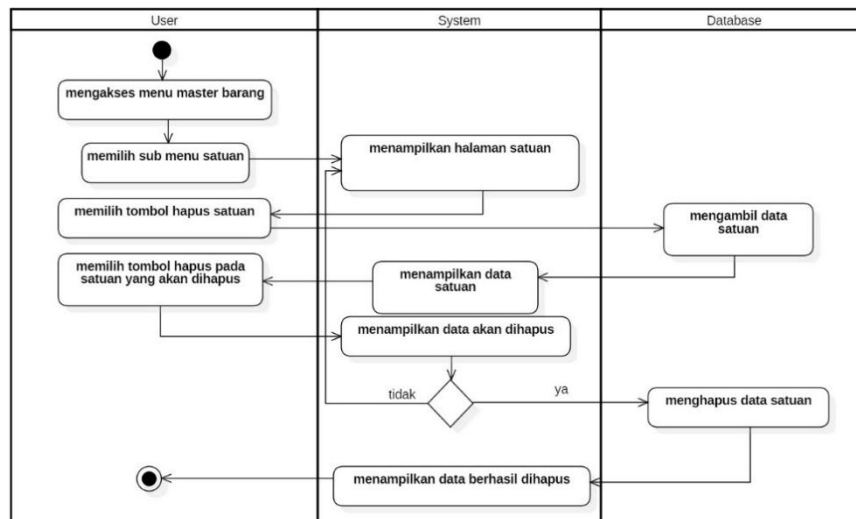
Gambar 18. *Activity Diagram* Mengakses Satuan Barang.

Activity Diagram menambah satuan barang diperlihatkan pada Gambar 19 yang menjelaskan *user* dapat menambah data satuan barang dengan dengan memilih tombol tambah satuan barang, selanjutnya sistem akan menampilkan halaman form tambah satuan barang. *User* kemudian akan mengisi data satuan barang yang akan disimpan kedalam *database* satuan.



Gambar 19. *Activity Diagram* Menambah Satuan Barang.

Activity Diagram menghapus satuan barang diperlihatkan pada Gambar 20 yang menjelaskan *user* dapat menghapus data satuan barang dengan mencari nama satuan barang yang akan dihapus datanya, selanjutnya sistem akan menampilkan data satuan barang yang akan dihapus. *User* kemudian akan memilih tombol hapus dan data satuan barang akan terhapus dari *database* satuan.

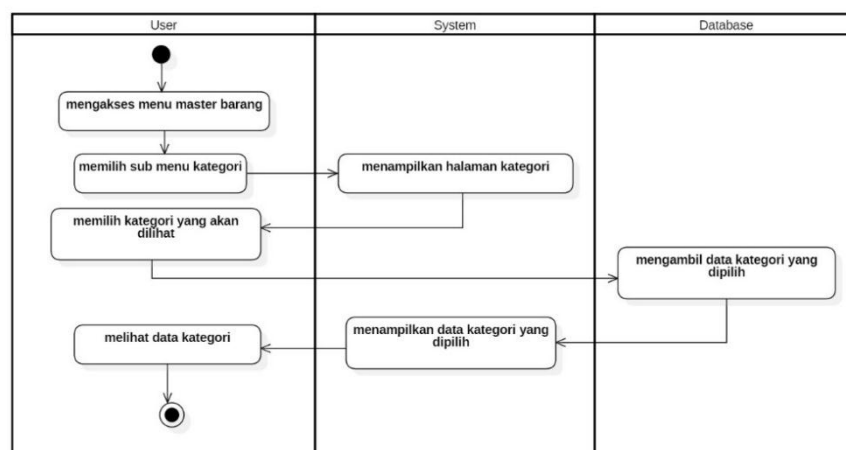


Gambar 20. *Activity Diagram* Menghapus Satuan Barang.

2.6. *Activity Diagram* Mengelola Kategori Barang

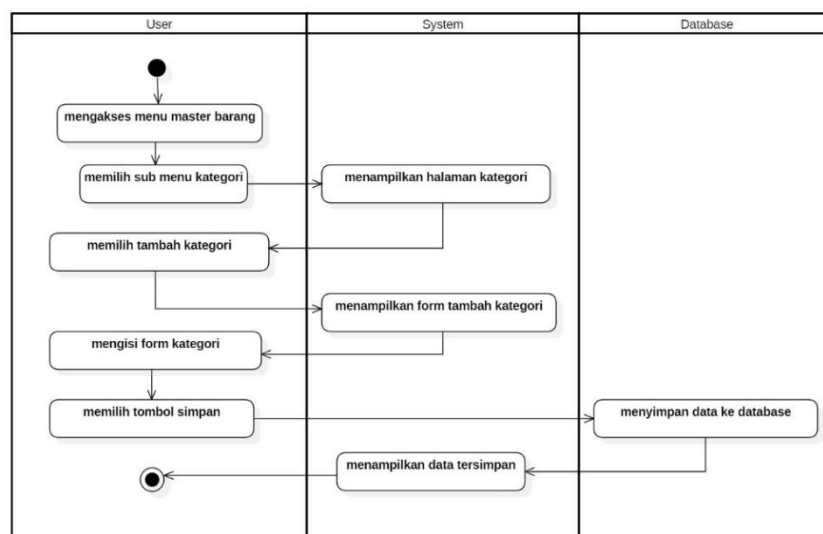
Gambar 21 memperlihatkan *Activity Diagram* mengakses *kategori*.

User dapat mengakses kategori barang dengan memilih menu master barang lalu memilih sub menu kategori barang. Sistem kemudian menampilkan informasi data kategori barang yang ada di *database* kategori barang.



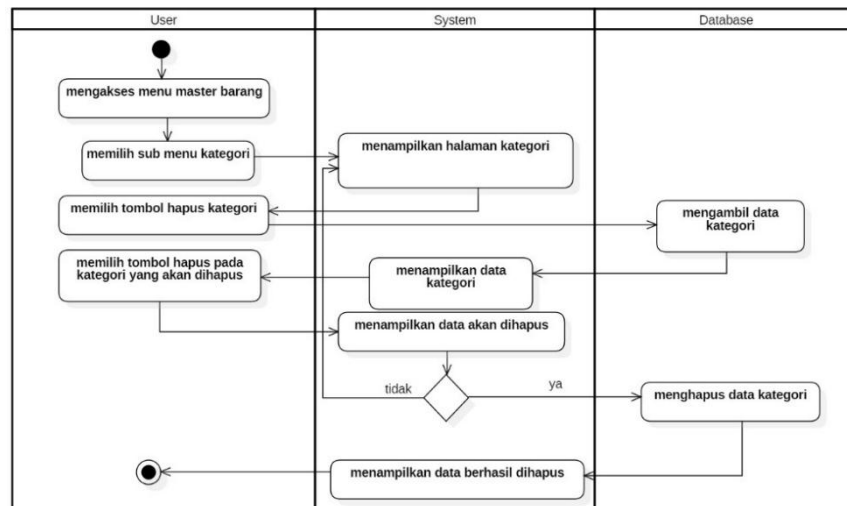
Gambar 21. *Activity Diagram* Mengakses Kategori Barang.

Activity Diagram menambah kategori barang diperlihatkan pada Gambar 22 yang menjelaskan *user* dapat menambah data kategori barang dengan dengan memilih tombol tambah kategori barang, selanjutnya sistem akan menampilkan halaman form tambah kategori barang. *User* kemudian akan mengisi data kategori barang yang akan disimpan kedalam *database* kategori.



Gambar 22. *Activity Diagram* Menambah Kategori Barang.

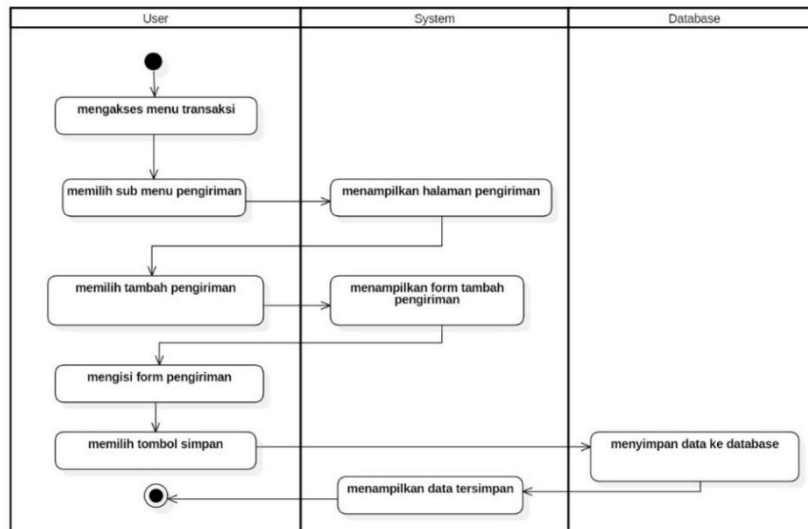
Activity Diagram menghapus kategori barang diperlihatkan pada Gambar 23 yang menjelaskan *user* dapat menghapus data kategori barang dengan mencari nama kategori barang yang akan dihapus datanya, selanjutnya sistem akan menampilkan data kategori barang yang akan dihapus. *User* kemudian akan memilih tombol hapus dan data kategori barang akan terhapus dari *database* kategori.



Gambar 23. *Activity Diagram* Menghapus Kategori Barang.

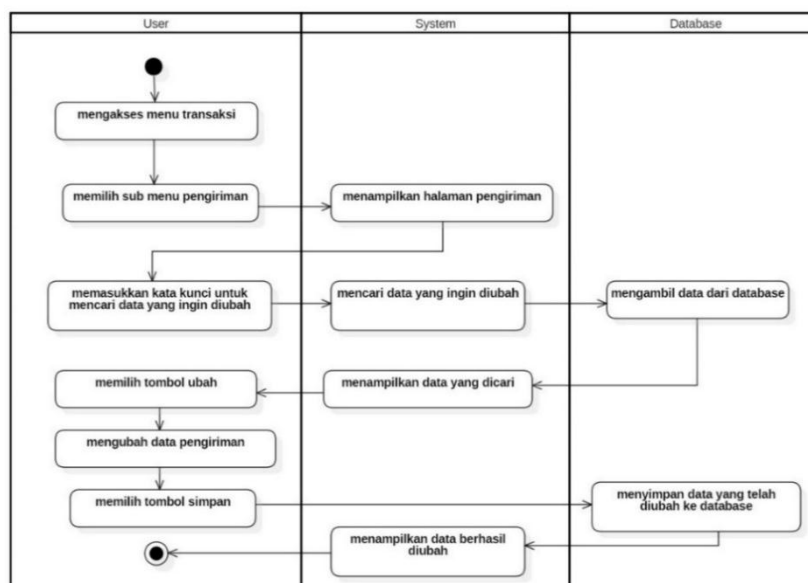
2.7. *Activity Diagram* Transaksi Pengiriman Pembelian Barang

Activity Diagram mengelola pengiriman pembelian barang terbagi menjadi 3 *Activity Diagram* yaitu menambah pengiriman, mengubah data pengiriman, dan menghapus data pengiriman. Gambar 24 memperlihatkan bahwa *user* dapat menambah data pengiriman dengan memilih tombol tambah pengiriman, selanjutnya sistem akan menampilkan halaman form tambah pengiriman. *User* kemudian akan mengisi data pengiriman yang akan disimpan kedalam *database* transaksi_pengiriman.



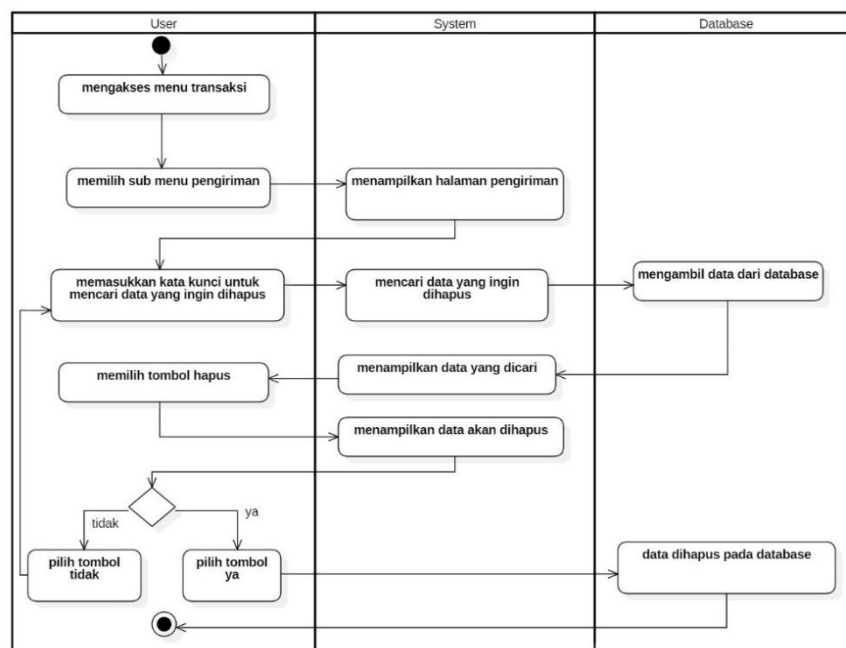
Gambar 24. *Activity Diagram* Menambah Pengiriman Pembelian Barang.

Gambar 25 yang menjelaskan *Activity Diagram* mengubah pengiriman. *User* mencari data ingin diubah, selanjutnya sistem menampilkan data pengiriman yang akan diubah. *User* mengubah data pengiriman dan disimpan ke *database* transaksi_pengiriman.



Gambar 25. *Activity Diagram* Mengubah Pengiriman Pembelian Barang.

Activity Diagram menghapus pengiriman diperlihatkan pada Gambar 26 yang menjelaskan *user* dapat menghapus data pengiriman dengan mencari nama pengiriman yang akan dihapus datanya, selanjutnya sistem akan menampilkan data pengiriman yang akan dihapus. *User* kemudian akan memilih tombol hapus dan data pengiriman akan terhapus dari *database* transaksi_pengiriman.

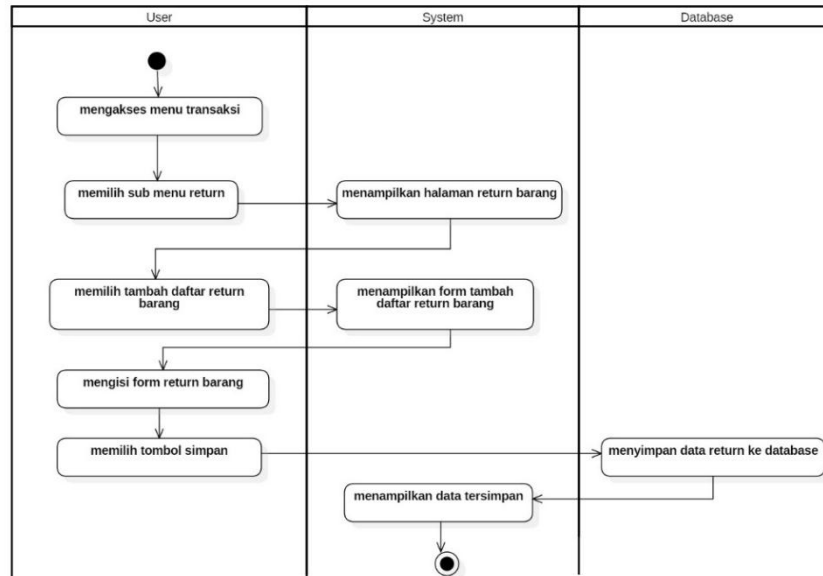


Gambar 26. *Activity Diagram* Menghapus Pengiriman Pembelian Barang.

2.8. *Activity Diagram* Mengelola *Return* Barang

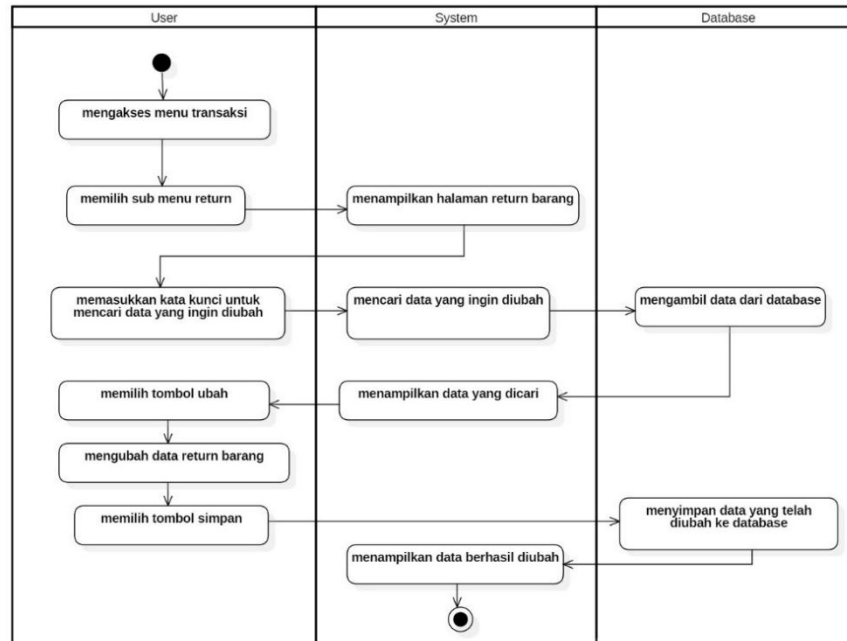
Activity Diagram mengelola *Return* barang terbagi menjadi 3 *Activity Diagram* yaitu menambah *Return*, mengubah data *Return*, dan menghapus data *Return*. Gambar 27 memperlihatkan bahwa *user* dapat menambah data *Return* dengan memilih tombol tambah *Return*, selanjutnya sistem akan menampilkan halaman form

tambah *Return*. *User* kemudian akan mengisi data *Return* yang akan disimpan kedalam *database Return*.



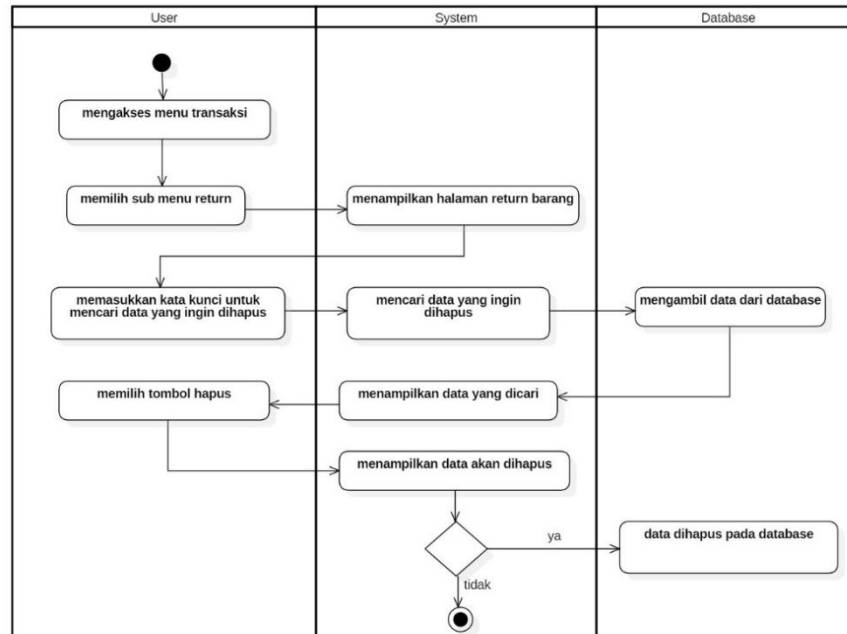
Gambar 27. *Activity Diagram* Menambah *Return* Barang.

Activity Diagram mengubah *Return* diperlihatkan pada Gambar 28 yang menjelaskan *user* dapat mengubah data *Return* dengan mencari nama *Return* yang akan diubah datanya, selanjutnya sistem akan menampilkan data *Return* yang akan diubah datanya. *User* kemudian akan mengubah data *Return* dan akan disimpan kedalam *database Return*.



Gambar 28. Activity Diagram Mengubah Return Barang.

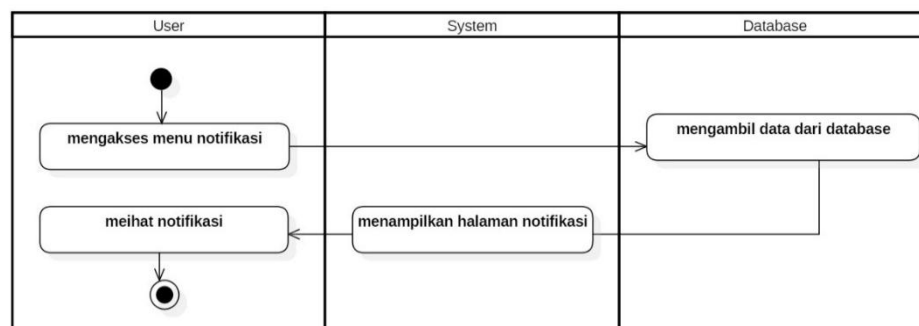
Activity Diagram menghapus Return diperlihatkan pada Gambar 29 yang menjelaskan user dapat menghapus data Return dengan mencari data Return yang akan dihapus datanya, selanjutnya sistem akan menampilkan data Return yang akan dihapus. User kemudian akan memilih tombol hapus dan data Return akan terhapus dari database Return.



Gambar 29. *Activity Diagram Menghapus Return Barang.*

2.9. Activity Diagram Notifikasi

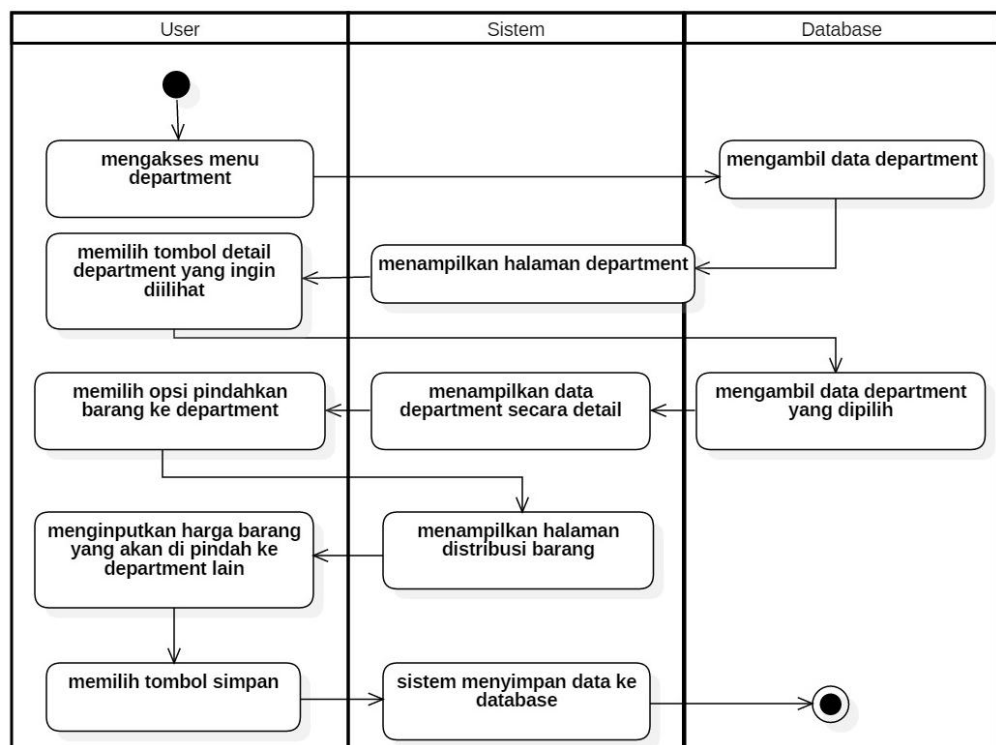
Activity Diagram mengelola notifikasi akan diperlihatkan pada Gambar 30 menerangkan bahwa *user* dapat mengakses menu notifikasi dengan memilih menu notifikasi. Sistem kemudian menampilkan informasi jumlah stok barang yang tersedia pada gudang yang ada di *database* notifikasi.



Gambar 30. *Activity Diagram Mengakses Notifikasi.*

2.10. Activity Diagram Department

Activity Diagram mengelola *department* akan diperlihatkan pada Gambar 31 menerangkan bahwa *user* dapat mengelola menu *department* untuk memindahkan barang dari *department* gudang pusat dengan gudang cabang. Sistem kemudian menampilkan informasi nama *department* yang dipilih. Halaman yang menampilkan detail *department* kemudian dapat memindahkan barang ke *department* lain.

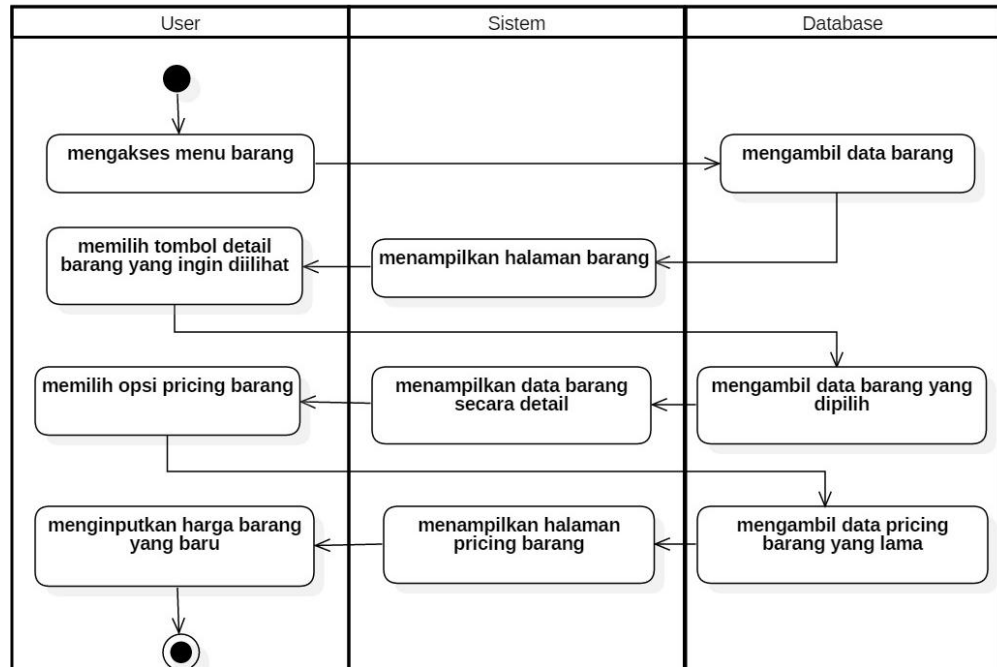


Gambar 31. Activity Diagram Menu Department.

2.11. Activity Diagram Pricing

Activity Diagram mengelola *pricing* akan diperlihatkan pada Gambar 32, menerangkan bahwa *admin* dapat mengelola *pricing*

barang untuk memberikan harga barang. *Admin* juga dapat mengubah harga barang lama menjadi harga barang yang baru.

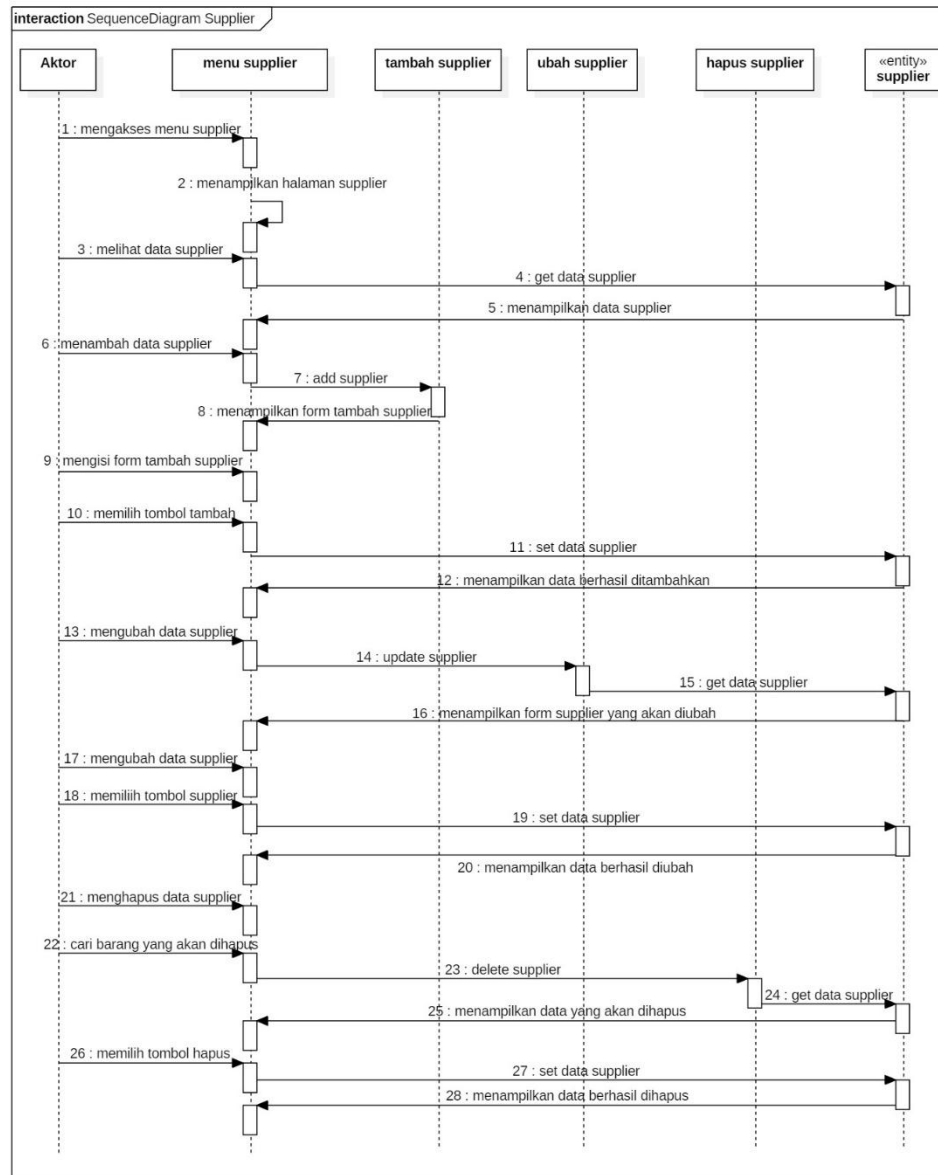


Gambar 32. Activity Diagram Menu Pricing Barang.

3. Sequence Diagram

3.1. Sequence Diagram Mengelola Supplier

Gambar 33 memperlihatkan *Sequence Diagram* yang didalamnya terdapat alur sistem yang akan dijalankan oleh *user*. *Sequence Diagram* ini memiliki *lifeline* aktor, menu *Supplier*, tambah *Supplier*, ubah *Supplier*, hapus *Supplier*, dan *database Supplier*.

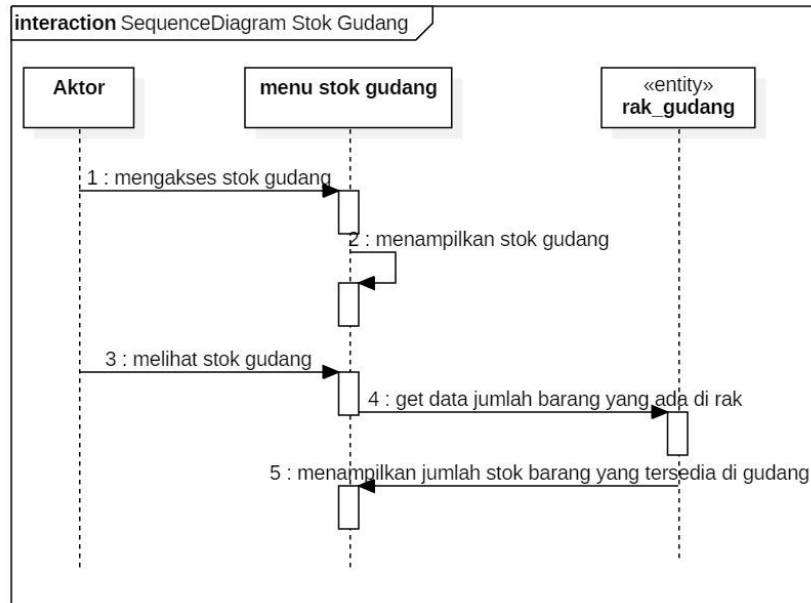


Gambar 33. *Sequence Diagram Supplier*.

3.2. *Sequence Diagram Mengakses Stok Gudang*

Gambar 34 memperlihatkan *Sequence Diagram* yang memiliki *lifeline* aktor yang merupakan *user*, menu stok gudang sebagai tampilan awal stok gudang, dan entity *rak_gudang* yang merupakan *database* pada menu tersebut. *Database* *rak_gudang* akan berfungsi

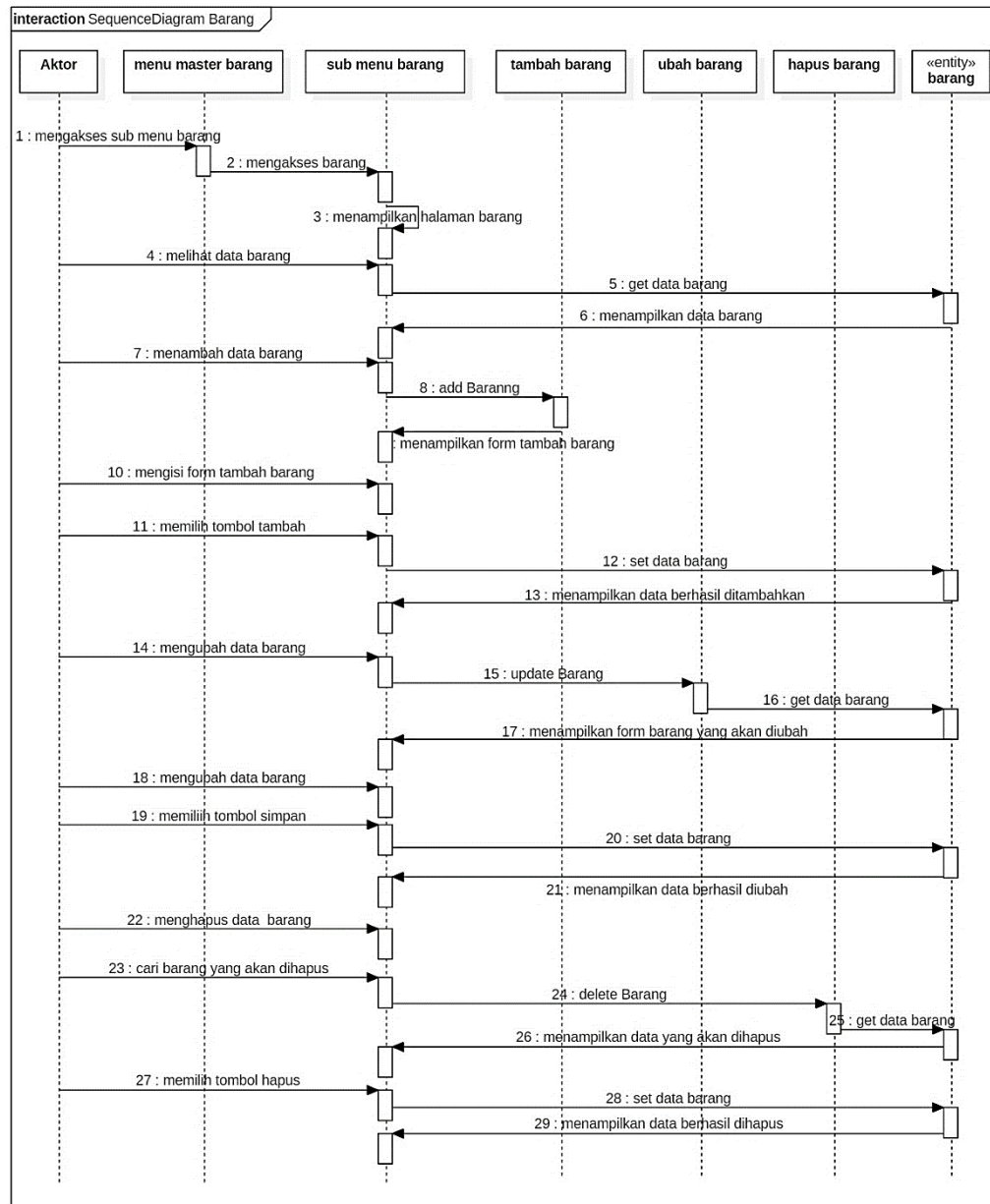
untuk mengetahui berapa banyak stok/jumlah barang yang ada pada gudang.



Gambar 34. *Sequence Diagram* Stok Gudang.

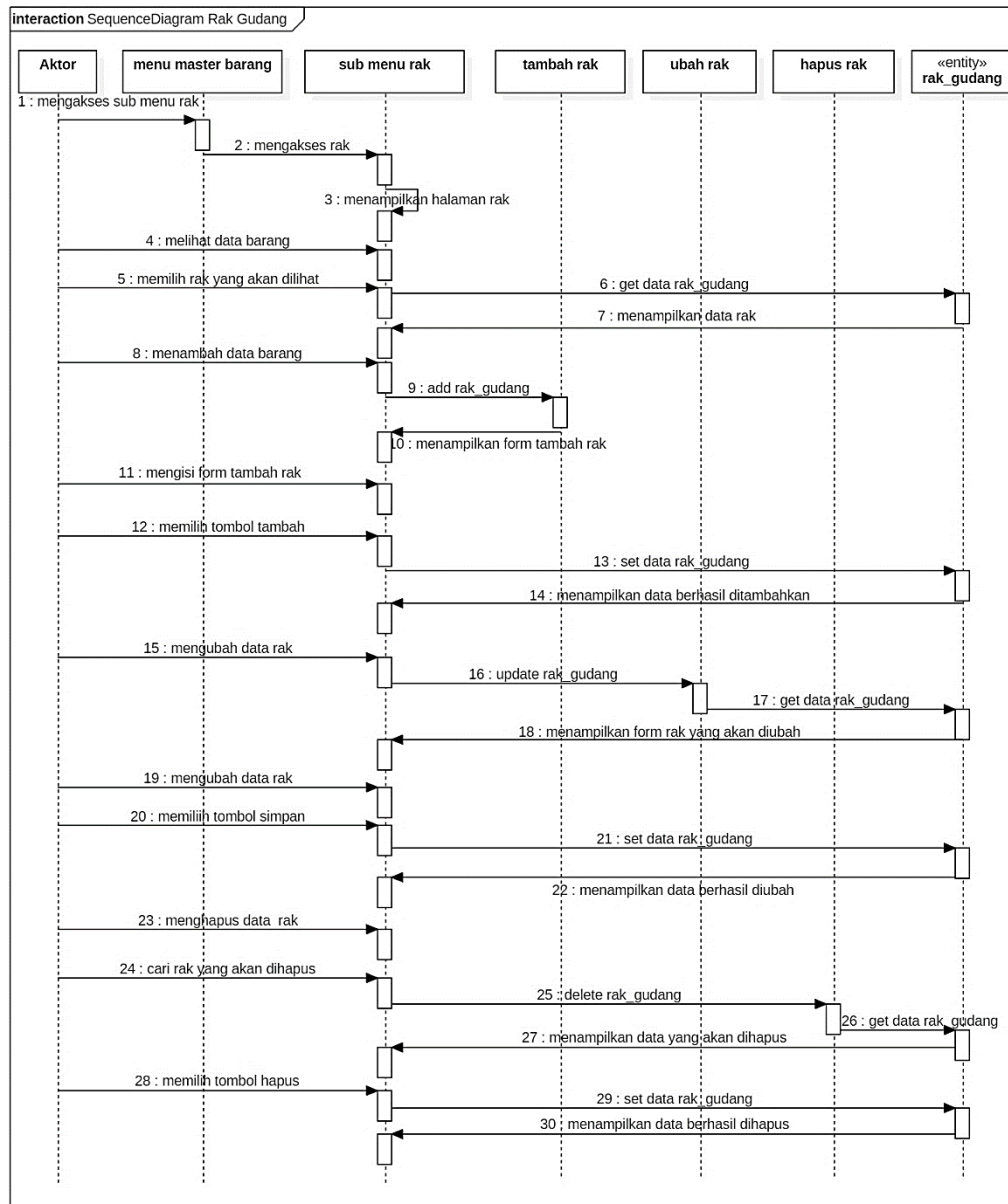
3.3. *Sequence Diagram* Mengelola Barang

Gambar 35 memperlihatkan *Sequence Diagram* yang memiliki aktor atau *user*, menu master barang, sub menu barang, tambah barang, ubah barang, hapus barang, dan *database* barang.

Gambar 35. *Sequence Diagram* Barang.

3.4. *Sequence Diagram* Mengelola Rak Gudang

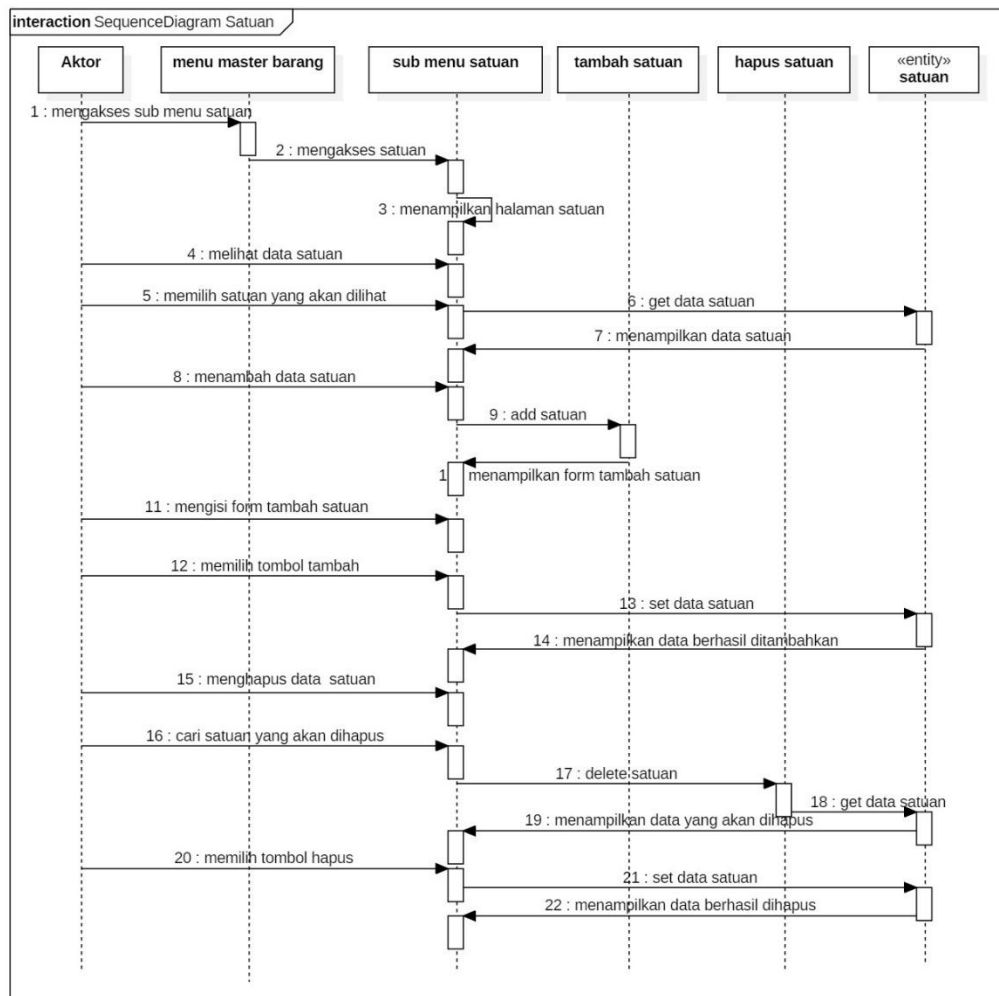
Sequence Diagram mengelola rak gudang diperlihatkan pada Gambar 36 yang memiliki aktor atau *user*, menu master barang, sub menu rak, tambah rak, ubah rak, hapus rak, dan *database* rak_gudang.



Gambar 36. *Sequence Diagram* Rak Gudang.

3.5. *Sequence Diagram* Mengelola Satuan Barang

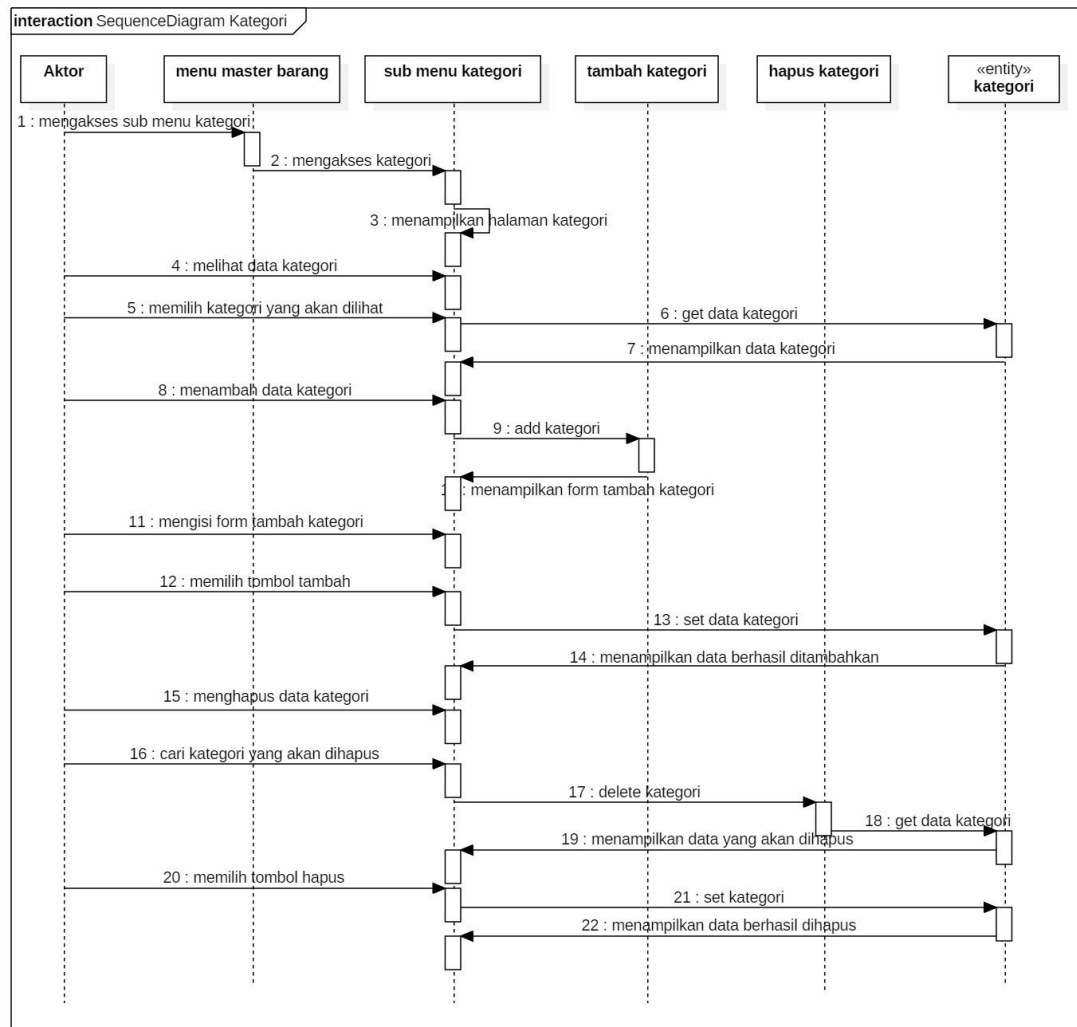
Sequence Diagram mengelola satuan barang diperlihatkan di Gambar 37 yang memiliki *lifeline* aktor atau *user*, menu master barang, sub menu satuan, tambah satuan, hapus satuan, dan *database* satuan.



Gambar 37. *Sequence Diagram* Satuan Barang.

3.6. *Sequence Diagram* Mengelola Kategori Barang

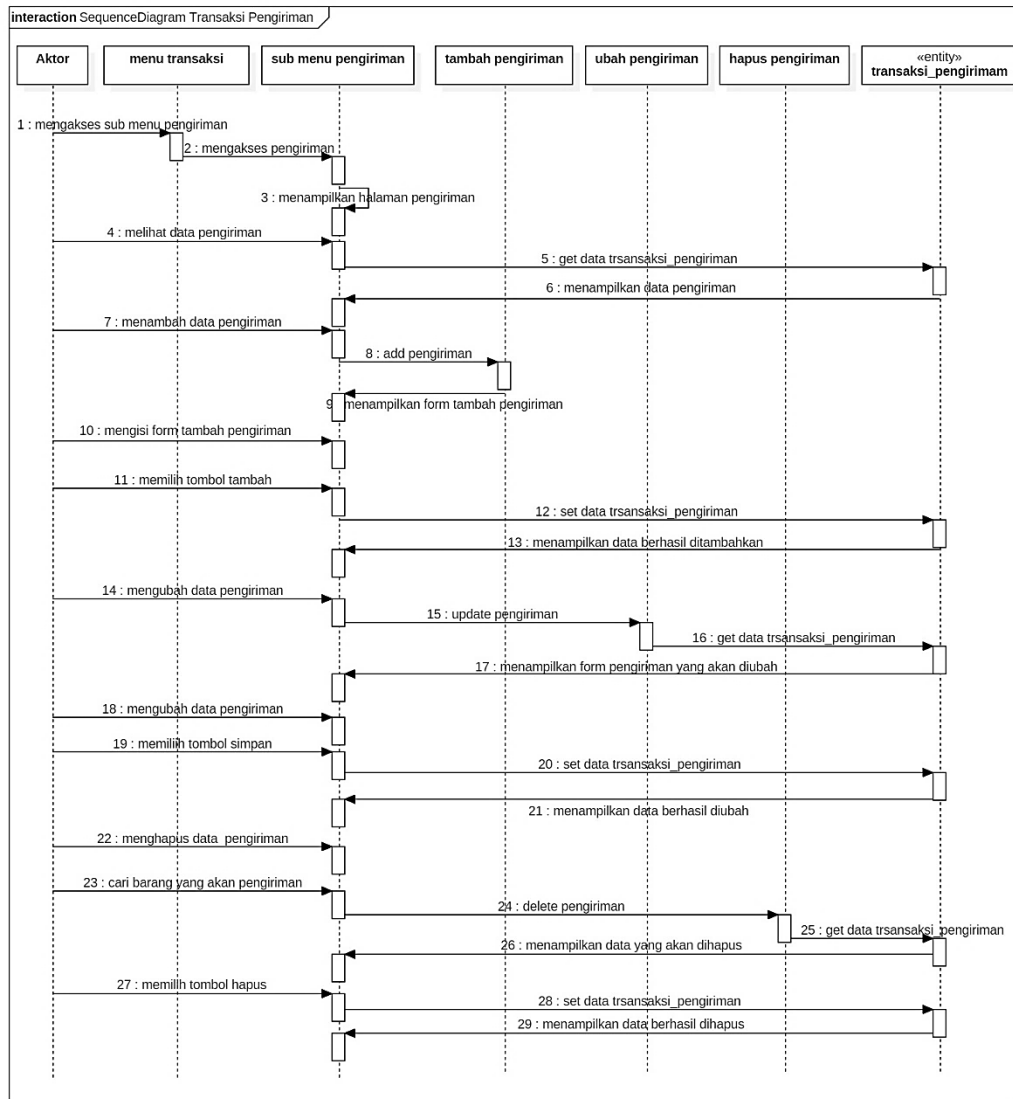
Sequence Diagram mengelola kategori barang diperlihatkan di Gambar 38 yang memiliki *lifeline* aktor atau *user*, menu master barang, sub menu kategori, tambah kategori, hapus kategori, dan *database* kategori.



Gambar 38. *Sequence Diagram* Kategori Barang.

3.7. *Sequence Diagram* Transaksi Pengiriman Pembelian Barang

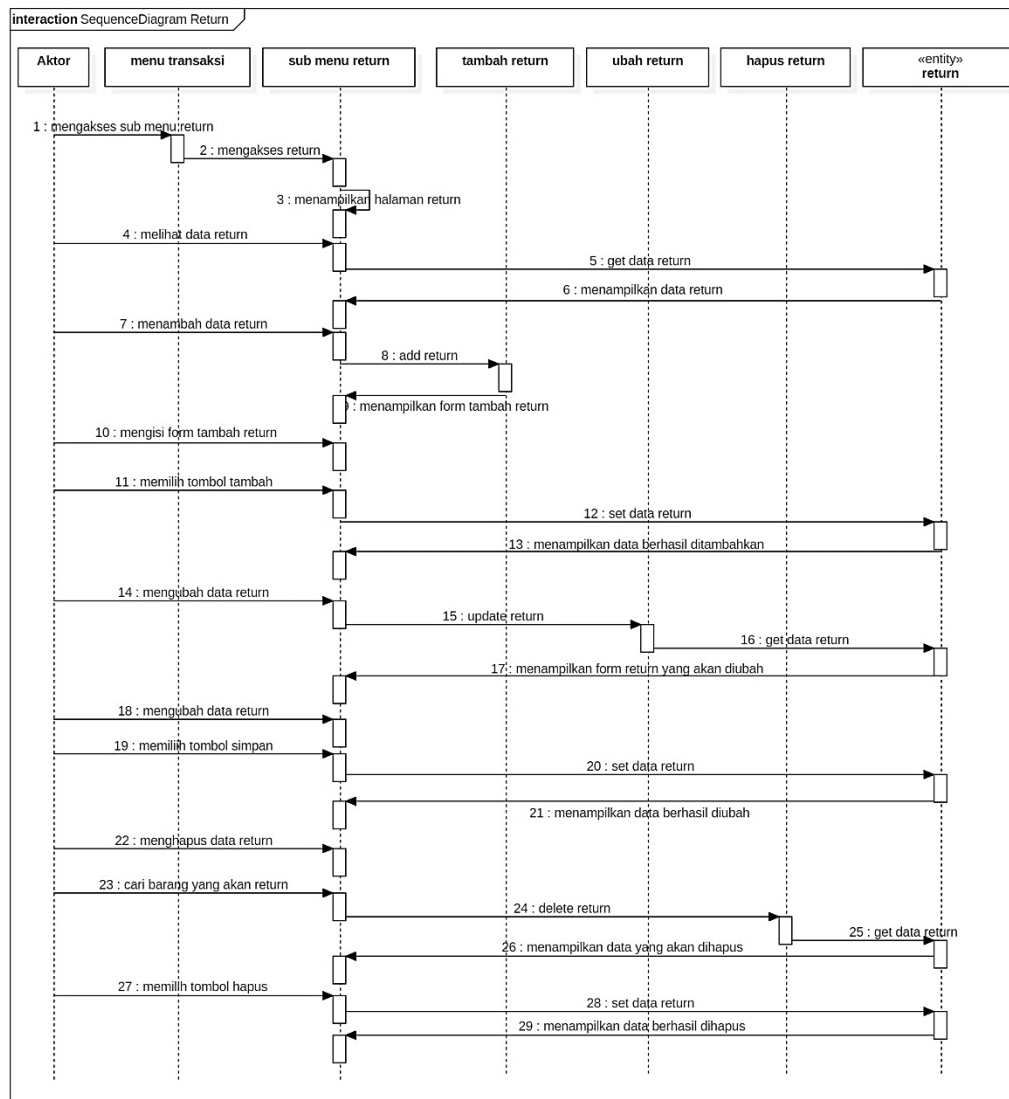
Sequence Diagram mengelola transaksi pengiriman pembelian barang diperlihatkan di Gambar 39 yang memiliki *lifeline* aktor atau *user*, menu transaksi, sub menu pengiriman, tambah pengiriman, ubah pengiriman, hapus pengiriman, dan *database* transaksi_pengiriman.



Gambar 39. *Sequence Diagram* Pengiriman Pembelian Barang.

3.8. *Sequence Diagram* Mengelola *Return* Barang

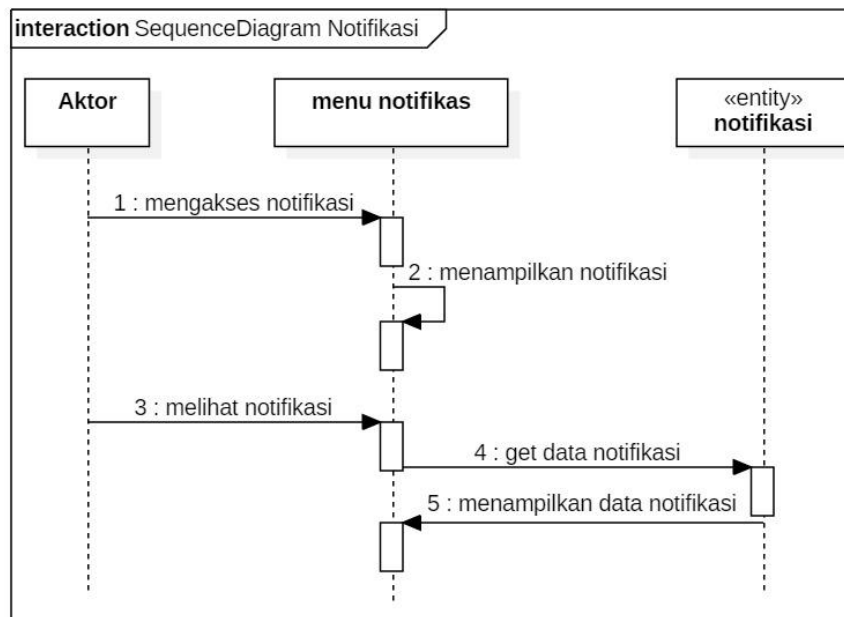
Sequence Diagram mengelola *Return* barang diperhatikan di Gambar 40 yang memiliki aktor atau *user*, menu transaksi, sub menu *Return*, tambah *Return*, ubah *Return*, hapus *Return*, dan *database Return*.



Gambar 40. *Sequence Diagram Return Barangs.*

3.9. Sequence Diagram Mengakses Notifikasi

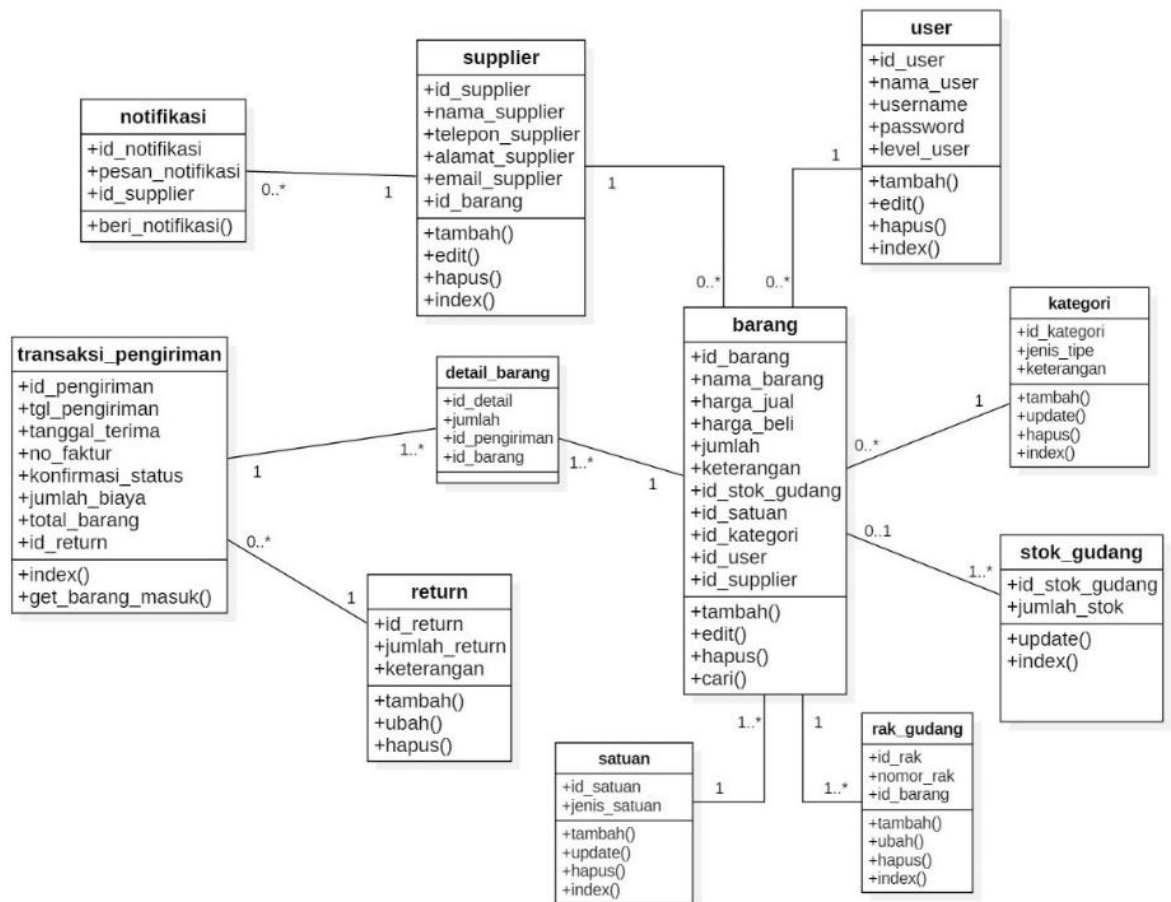
Sequence Diagram mengakses notifikasi diperlihatkan di Gambar 41 yang memiliki aktor atau *user*, menu notifikasi, dan *database* notifikasi.



Gambar 41. *Sequence Diagram* Notifikasi.

4. *Class Diagram*

Gambar 42 memperlihatkan hubungan antara *entity* yang akan diimplementasikan menjadi *table*-tabel dalam *database*. *Class* diagram ini memiliki beberapa *entity* yaitu barang sebagai master data barang, *supplier* sebagai master data *supplier*, *user* yang memiliki tugas untuk menginputkan data, notifikasi untuk pemberitahuan ke *supplier*, transaksi_pengiriman untuuk pendataan barang masuk, *return* sebagai data pengembalian barang ke *supplier*, detail_barang, kategori, satuan, dan rak_gudang.



Gambar 42. Class Diagram.

5. Class Diagram Team

Sistem Informasi Manajemen *Supplier* dan Stok Gudang ini akan terhubung dengan sistem CRM dan pendistribusian stok, sistem *Dashboard Business Intelligence*, dan sistem informasi akuntansi. Gambar 43 memperlihatkan hubungan arus data yang akan dikirim oleh sistem satu ke sistem lainnya. Sistem SSG akan mengirimkan data ke sistem CRM dan BID yaitu data *supplier*, barang, kategori barang, dan transaksi pengiriman pembelian barang. Sistem SSG akan mengirimkan data ke sistem SIA berupa data transaksi pengiriman.

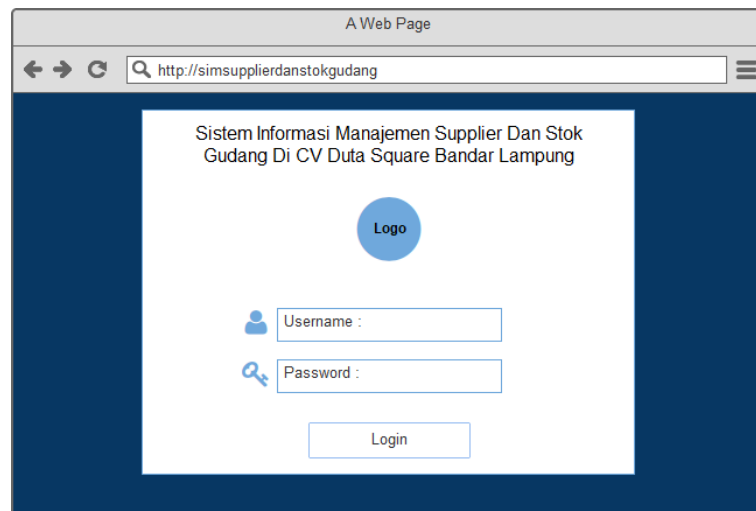
6. *User interface*

User interface ini akan terbagi untuk dua *user* yaitu *Admin* dan *Supplier*. *User interface* tersebut yaitu:

6.1. *User interface Login*

Gambar 44 memperlihatkan *user interface login* untuk semua *user*.

User harus masuk menggunakan *username* dan *password* jika ingin mengakses sistem. *User* yang datanya valid dengan *database*. Jika *username* dan *password* benar, maka *user* dapat mengakses sistem.



Gambar 44. *User interface Login*.

6.2. *User interface Home- Admin*

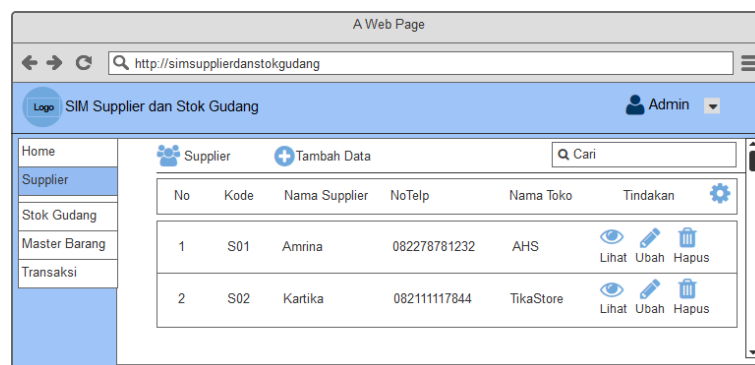
Gambar 45 memperlihatkan *user interface Home* untuk *Admin* yang merupakan *interface* halaman utama/ *Home* sebagai pembuka saat *admin* masuk kedalam sistem atau berhasil *Login*.



Gambar 45. *User interface Home.*

6.3. *User interface Supplier- Admin*

Gambar 46 menunjukkan bahwa *user interface* tersebut merupakan *interface Supplier*. *Interface* ini akan berisikan data *supplier* yang disimpan dalam *database supplier*.



Gambar 46. *User interface Supplier.*

6.4. *User interface Stok Gudang- Admin*

Gambar 47 memperlihatkan *user interface* menu stok gudang. Stok gudang merupakan menu yang di dalamnya terdapat informasi mengenai barang-barang yang telah dikirimkan oleh *supplier* dan jumlah stok barang yang tersedia di gudang.



Gambar 47. *User interface* Stok Gudang.

6.5. *User interface* Barang- Admin

Gambar 48 memperlihatkan *interface* sub menu barang. Barang merupakan sub menu yang dapat menginputkan informasi barang datang dari *supplier*.

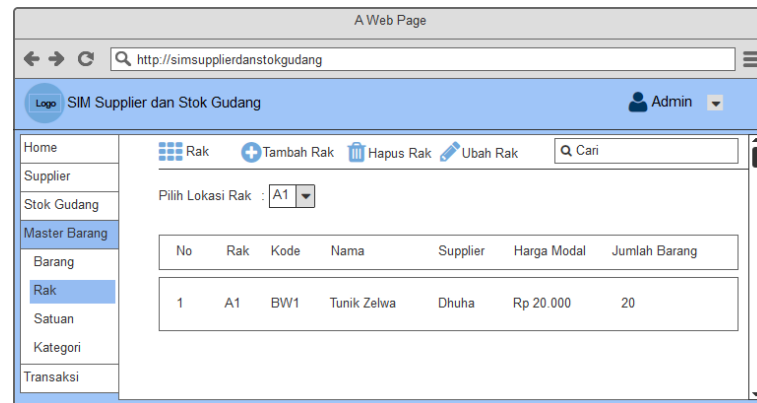


Gambar 48. *User interface* Barang.

6.6. *User interface* Rak Gudang- Admin

Gambar 49 memperlihatkan *user interface* rak gudang. Rak merupakan sub menu yang menampilkan informasi isi rak gudang,

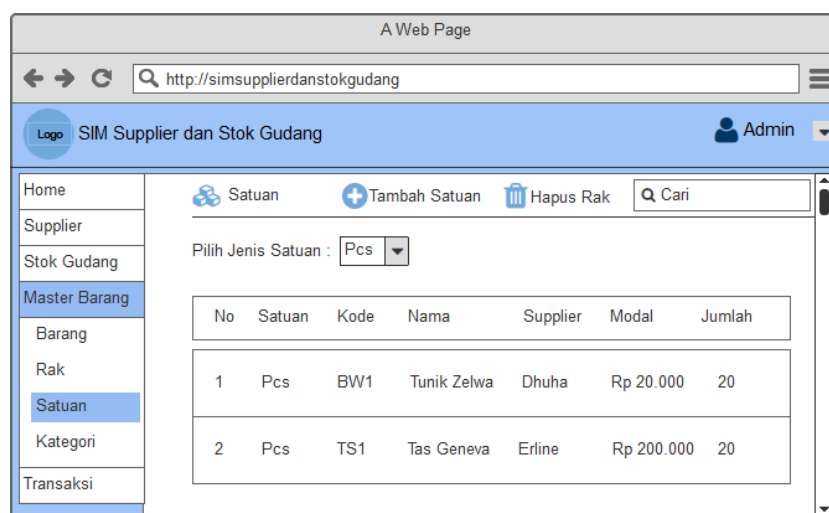
menambah, menghapus dan mengubah rak gudang. Rak gudang dibuat berdasarkan jenis barang.



Gambar 49. *User interface* Rak Gudang.

6.7. *User interface* Satuan- Admin

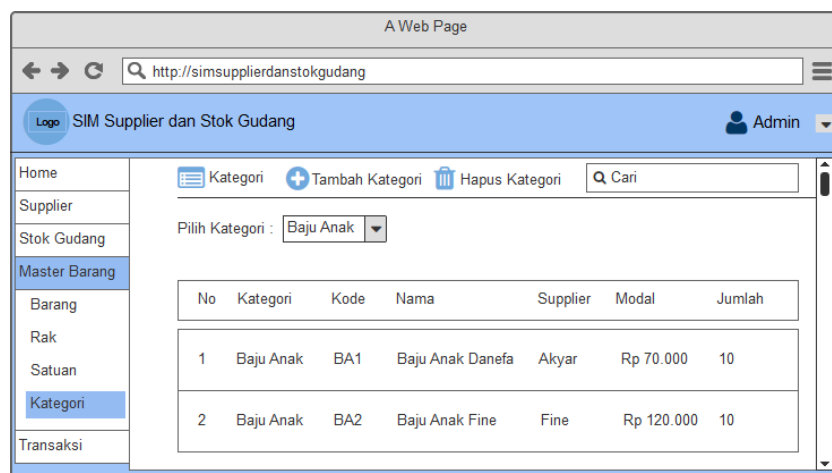
Gambar 50 memperlihatkan *interface* menu Satuan. Satuan merupakan menu yang dapat menampilkan informasi barang yang ada di gudang berdasarkan satuan.



Gambar 50. *User interface* Satuan.

6.8. User interface Kategori- Admin

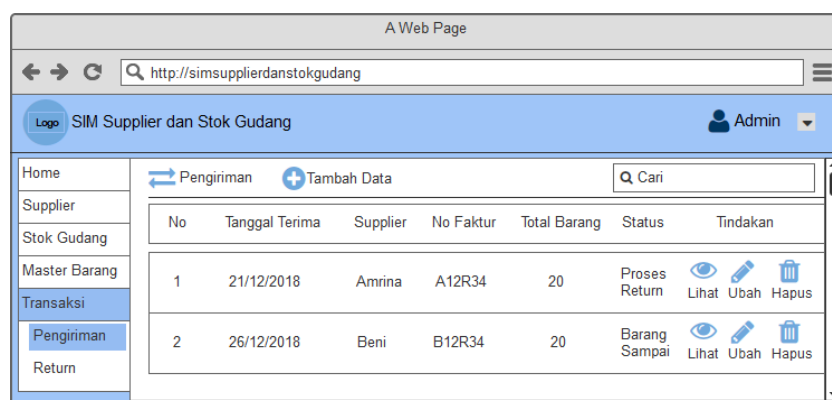
Gambar 51 memperlihatkan *interface* menu kategori. Kategori merupakan menu yang dapat menampilkan informasi barang berdasarkan kategori, contohnya yaitu kategori Baju Anak.



Gambar 51. User interface Kategori.

6.9. User interface Pengiriman Pembelian Barang - Admin

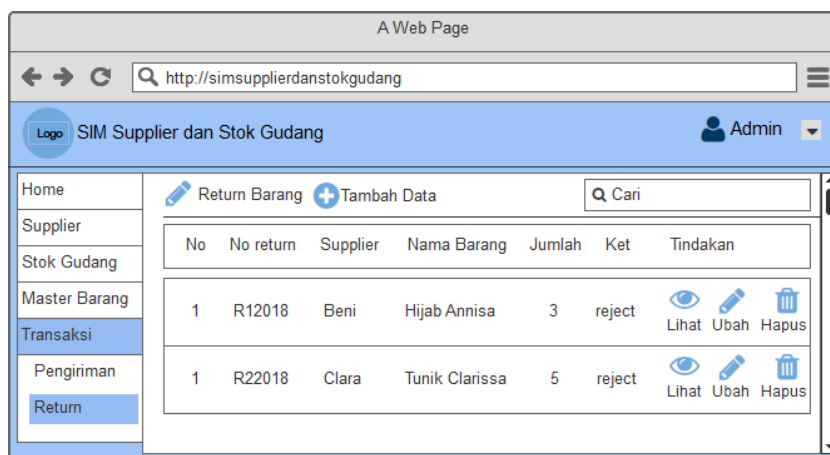
Gambar 52 memperlihatkan *interface* menu transaksi pengiriman. Transaksi pengiriman dapat menampilkan informasi tentang pengiriman pembelian barang dari *supplier* CV Duta Square.



Gambar 52. User interface Pengiriman Pembelian Barang.

6.10. User interface Return- Admin

Gambar 53 memperlihatkan *interface Return* yang berfungsi ketika terdapat barang yang rusak saat diterima oleh CV Duta Square dan akan dikembalikan ke *supplier*.



Gambar 53. User interface Return.

6.11. User interface Home- Supplier

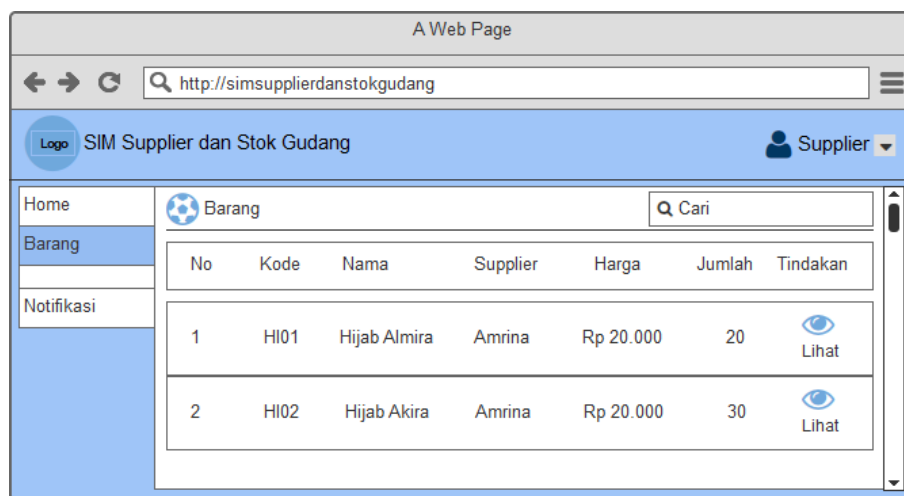
Gambar 54 memperlihatkan *interface Home* untuk hak akses *supplier*. *Interface Home* merupakan halaman utama sebagai pembuka saat *admin* masuk kedalam sistem atau berhasil *Login*.



Gambar 54. User interface Home Supplier.

6.12. User interface Barang- Supplier

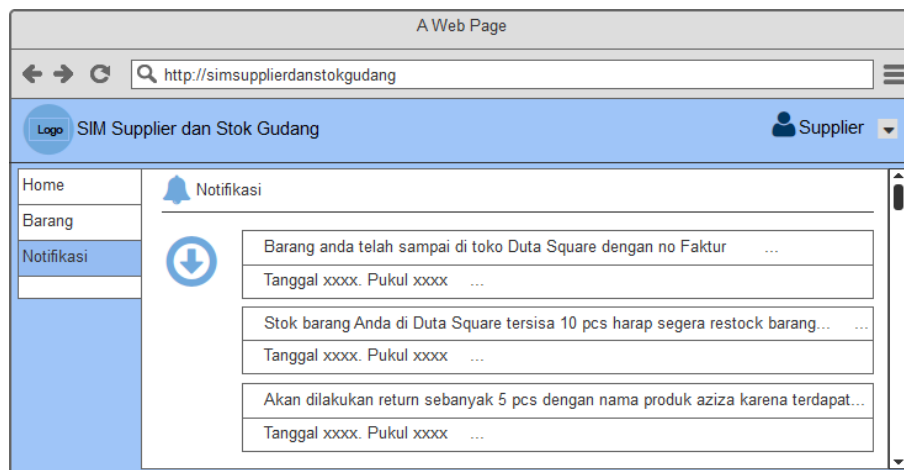
Gambar 55 memperlihatkan *interface* barang untuk hak akses *supplier*. *Interface* ini berguna agar *supplier* dapat melihat jumlah sisa barang mereka sehingga *supplier* mengetahui barang mereka laku atau tidak.



Gambar 55. User interface Barang- Supplier.

6.13. User interface Notifikasi- Supplier

Gambar 56 memperlihatkan *interface* notifikasi. *Interface* ini bertujuan agar *supplier* mengetahui informasi ketika stok barang mereka hampir habis di CV Duta Square.



Gambar 56. *User interface* Notifikasi.

7. Skenario Pengujian Sistem

Pengujian sistem yang dilakukan untuk menguji sistem ini menggunakan metode pengujian *Blackbox Testing*. *Black-box testing* adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi. Rancangan daftar pengujian dapat dilihat pada Tabel 6 untuk peran *admin* gudang dan Tabel 7 untuk peran *supplier*.

Tabel 6. Daftar Pengujian Fungsional Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square untuk peran *Admin* Gudang.

No.	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan
1.	<i>Login</i>	Pengujian pada halaman <i>login</i> .	Mengisi <i>username</i> = admin dan <i>password</i> =	Masuk ke Halaman <i>Dashboard</i> sebagai <i>Admin</i> Gudang.

Tabel 6. Daftar Pengujian Fungsional Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square untuk peran *Admin Gudang* (Lanjutan).

No.	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan
			admin kemudian klik “ <i>login</i> ”.	
			Mengisi <i>username</i> = admin dan <i>password</i> = 123 kemudian klik “ <i>login</i> ”.	Menampilkan peringatan bahwa <i>username</i> dan <i>password</i> salah dan <i>login</i> gagal.
2.	Halaman Menu Stok Gudang	Pengujian menampilkan halaman Stok Gudang	Klik pada menu Stok Gudang	Menampilkan informasi jumlah stok barang yang ada di gudang.
3.	Halaman Menu <i>Supplier</i>	Pengujian menampilkan halaman menu <i>Supplier</i>	Klik pada menu <i>Supplier</i>	Menampilkan informasi data semua <i>Supplier</i> dan tombol aksi setiap data <i>Supplier</i> .
		Pengujian pada tombol “Tambah Data”.	Klik pada tombol tambah data <i>Supplier</i>	Menampilkan form pengisian data <i>Supplier</i> baru
		Pengujian pada tombol “Ubah” pada data <i>Supplier</i>	Klik tombol ubah pada data <i>Supplier</i> yang akan diubah	Menampilkan data <i>Supplier</i> yang akan diubah dengan mengambil id <i>Supplier</i> dari <i>database Supplier</i> .
		Pengujian pada tombol “Hapus” yang ada pada setiap data <i>Supplier</i>	Klik tombol hapus pada data <i>Supplier</i> yang akan dihapus	Menampilkan kotak konfirmasi data akan dihapus. Jika ya data akan dihapus

Tabel 6. Daftar Pengujian Fungsional Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square untuk peran *Admin Gudang* (Lanjutan).

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan
.				dari <i>database</i> . Jika tidak akan kembali ke halaman menu <i>Supplier</i> .
4..	Halaman Sub Menu Barang	Pengujian menampilkan halaman sub menu barang.	Klik pada menu Master Barang, lalu klik sub menu barang.	Menampilkan halaman yang berisi data barang.
		Pengujian pada tombol “Tambah Data” yang ada di halaman sub menu barang.	Klik pada menu Master Barang, lalu klik sub menu barang, dan klik tombol “Tambah Data” yang ada pada halaman sub menu barang.	Menampilkan form pengisian data barang baru.
		Pengujian pada tombol “Ubah” yang ada pada setiap data barang.	Klik pada menu Master Barang, lalu klik sub menu barang, dan klik tombol “Ubah” yang ada pada tiap data barang.	Menampilkan data barang yang akan diubah dengan mengambil id barang dari <i>database</i> barang.
		Pengujian pada tombol “Hapus” yang ada pada setiap data barang	Klik pada menu Master Barang, lalu klik sub menu barang, dan klik tombol “Hapus” yang ada pada tiap data barang	Menampilkan kotak konfirmasi data akan dihapus atau tidak, Jika ya data akan dihapus dari <i>database</i> . Jika tidak akan kembali ke halaman sub menu barang.
5.	Halaman Sub Menu Rak	Pengujian menampilkan halaman sub menu rak dan data barang	Klik pada menu Master Barang, lalu klik sub menu rak, dan	Menampilkan halaman yang berisi data barang pada rak yang dipilih.

Tabel 6. Daftar Pengujian Fungsional Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square untuk peran *Admin Gudang* (Lanjutan).

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan
.		yang ada di rak.	pilih rak yang akan dilihat.	
		Pengujian pada tombol “Tambah Rak” yang ada di halaman sub menu rak.	Klik pada menu Master Barang, lalu klik sub menu rak, dan klik tombol tambah rak.	Menampilkan form pengisian tambah rak.
		Pengujian pada tombol “Ubah” yang ada di halaman sub menu rak.	Klik pada menu Master Barang, lalu klik sub menu rak, dan klik tombol ubah rak. Pilih rak yang akan diubah, dan klik tombol Ubah rak.	Menampilkan data rak yang akan diubah dengan mengambil id rak dari <i>database</i> rak_gudang.
		Pengujian pada tombol “Hapus” yang ada di halaman sub menu rak.	Klik pada menu Master Barang, lalu klik sub menu rak, dan klik tombol Hapus rak. Pilih rak yang akan dihapus, dan klik tombol hapus rak.	Menampilkan kotak konfirmasi data akan dihapus atau tidak, Jika ya data akan dihapus dari <i>database</i> rak_gudang. Jika tidak akan kembali ke halaman sub menu rak.
6.	Halaman Sub Menu Satuan	Pengujian menampilkan halaman sub menu satuan.	Klik pada menu Master Barang, lalu klik sub menu satuan.	Menampilkan halaman yang berisi data barang pada satuan yang dipilih.
		Pengujian pada tombol “Tambah Satuan” yang ada di halaman sub menu satuan.	Klik pada menu Master Barang, lalu klik sub menu satuan, dan klik tombol tambah satuan.	Menampilkan form pengisian untuk menambah satuan.

Tabel 6. Daftar Pengujian Fungsional Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square untuk peran *Admin Gudang* (Lanjutan).

No.	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan
		Pengujian pada tombol “Hapus” yang ada di halaman sub menu satuan.	Klik pada menu Master Barang, lalu klik sub menu satuan, dan klik tombol Hapus satuan. Pilih satuan yang akan dihapus, dan klik tombol hapus satuan.	Menampilkan data yang akan dihapus dan kotak konfirmasi data akan dihapus, Jika ya data akan dihapus dari <i>database</i> satuan.
7.	Halaman Sub Menu Kategori	Pengujian menampilkan halaman sub menu kategori.	Klik pada menu Master Barang, lalu klik sub menu kategori.	Menampilkan halaman yang berisi data barang pada kategori yang dipilih.
		Pengujian pada tombol “Tambah kategori” yang ada di halaman sub menu kategori.	Klik pada menu Master Barang, lalu klik sub menu kategori, dan klik tombol tambah kategori.	Menampilkan form pengisian untuk menambah kategori.
		Pengujian pada tombol “Hapus” yang ada di halaman sub menu kategori.	Klik pada menu Master Barang, lalu klik sub menu kategori, dan klik tombol Hapus kategori. Pilih kategori yang akan dihapus, dan klik tombol hapus kategori.	Menampilkan semua daftar kategori dan Menampilkan kotak konfirmasi data akan dihapus atau tidak, Jika ya data akan dihapus dari <i>database</i> kategori. Jika tidak akan kembali ke halaman sub menu kategori.
8.	Halaman Sub Menu Pengiriman Pembelian Barang.	Pengujian menampilkan halaman pengiriman	Klik pada transaksi pengiriman pembelian barang.	Menampilkan halaman yang berisi data pengiriman pembelian barang.

Tabel 6. Daftar Pengujian Fungsional Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square untuk peran *Admin Gudang* (Lanjutan).

No.	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan
		Pengujian pada tombol “Tambah Data” yang ada dihalaman pengiriman.	Klik Transaksi Pengiriman Pembelian, dan klik tombol “Tambah Data”	Menampilkan form pengisian data transaksi pengiriman pembelian baru.
		Pengujian pada Tombol “Ubah” yang ada pada setiap data pengiriman.	Klik Transaksi Pengiriman Pembelian dan klik tombol “Ubah” pada tiap data pengiriman.	Menampilkan data barang yang akan diubah dengan mengambil id pengiriman dari <i>database</i> .
		Pengujian pada tombol “Hapus” pada setiap data pengiriman.	Klik Transaksi Pengiriman Pembelian dan klik tombol “Hapus”	Menampilkan kotak konfirmasi data akan dihapus atau tidak.
9.	Halaman <i>Return</i>	Pengujian menampilkan halaman <i>Return</i> .	Klik pada menu Transaksi, dan klik <i>Return</i> .	Menampilkan halaman yang berisi data <i>Return</i> .
		Pengujian pada tombol “Tambah Data” pada <i>return</i> .	Klik pada menu Transaksi, klik halaman <i>Return</i> , dan klik tombol “Tambah Data”.	Menampilkan form pengisian data <i>Return</i> baru.
		Pengujian pada Tombol “Ubah” yang ada pada setiap data <i>Return</i>	Klik pada menu Transaksi, klik <i>Return</i> , dan klik tombol “Ubah”.	Menampilkan data <i>Return</i> yang akan diubah dengan mengambil id <i>Return</i> dari <i>database Return</i> .
		Pengujian pada tombol “Hapus” yang ada pada setiap data <i>Return</i>	Klik <i>Return</i> , dan klik tombol “Hapus” yang ada pada tiap data <i>Return</i> .	Menampilkan kotak konfirmasi data akan dihapus atau tidak.

Tabel 7. Daftar Pengujian Fungsional Sistem Informasi Manajemen *Supplier* dan Stok Gudang CV Duta Square untuk peran *Supplier*.

No.	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan
1.	<i>Login</i>	Pengujian pada halaman <i>login</i>	<p>Mengisi <i>username</i> = <i>Supplier</i> dan <i>password</i> = <i>Supplier</i> kemudian klik “<i>login</i>”.</p> <p>Mengisi <i>username</i> = <i>spplier</i> dan <i>password</i> = 123 kemudian klik “<i>login</i>”.</p>	<p>Masuk ke Halaman Dashboard sebagai <i>Supplier</i>.</p> <p>Menampilkan peringatan bahwa <i>username</i> dan <i>password</i> salah dan <i>login</i> gagal.</p>
2.	Halaman Menu Barang	Pengujian menampilkan halaman sub menu barang.	Klik pada menu Master Barang, lalu klik sub menu barang.	Menampilkan halaman yang berisi data barang <i>supplier</i> yang <i>login</i> .
3.	Halaman Notifikasi	Pengujian menampilkan halaman menu notifikasi.	Klik pada notifikasi	Menampilkan informasi notifikasi/ pemberitahuan untuk <i>Supplier</i> .
4.	Halaman Menu Transaksi Pengiriman Pembelian dan <i>Return</i>	Pengujian menampilkan halaman transaksi pengiriman pembelian dan <i>return</i>	Klik pada menu transaksi pengiriman pembelian barang dan <i>return</i>	Menampilkan halaman berisi data transaksi dan <i>return supplier</i> yang <i>login</i> .

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Adapaun kesimpulan yang didapat dari hasil penelitian ini adalah sebagai berikut:

1. Telah berhasil membangun Sistem *Supplier* dan Stok Gudang (SSG) berbasis web dengan menggunakan *framework* Laravel,
2. Sistem telah berhasil melakukan pendataan *supplier*, pendataan barang, dan menampilkan data transaksi pengiriman,
3. Sistem telah berhasil menampilkan data *Supplier*, data barang tunda, dan notifikasi pada saat *Supplier login*,
4. Sistem telah berhasil mengirimkan data ke Sistem *Business Intelligence Dashboard*, Sistem *Customer Relationship Management* dan Sistem Informasi Akuntansi menggunakan API (*Application Programming Interface*),
5. Sistem telah berhasil menerima data dari Sistem *Customer Relationship Management*.

B. Saran

Berdasarkan perancangan dan hasil dari implementasi sistem, terdapat beberapa saran yang perlu diperhatikan dalam mengembangkan sistem ini diantaranya:

1. Menambahkan fitur *live chat* antara *Owner* dan *Supplier* agar lebih mudah berdiskusi saat melakukan *restock* barang,
2. Menambahkan fitur laporan untuk data *supplier*, data barang, dan data transaksi pengiriman pembelian barang.

DAFTAR PUSTAKA

- Aswadi, M. (2015). *Database Dasar With XAMPP*. Surabaya: CV. Garuda Mas Sejahtera.
- Agusvianto, H. (2017). Sistem Informasi Inventori Gudang Untuk Mengontrol Persediaan Barang Pada Gudang Studi Kasus : PT.Alaisys Sidoarjo. *Journal Information Engineering and Educational Technology*, 1, 40–46.
- Badan Pusat Statistik. (2015). Jumlah Perusahaan Di Indonesia Menurut Provinsi Tahun 2015. Retrieved November 2, 2018, from <https://bps.go.id>
- Basuki, A. P. (2016). *Konsep dan Implementasi Pemrograman Laravel 5*. Yogyakarta: Lokomedia.
- Erlina. (2008). *Metodologi Penelitian Bisnis untuk Akuntansi dan Manajemen* (2nd ed.). Medan: USU Press.
- Gasam, E., Metandi, F., & Utomo, K. B. (2018). Rancang Bangun Aplikasi Penerimaan Anggota Himpunan Mahasiswa Di Jurusan Teknologi Informasi Menggunakan Framework Laravel. *Jurnal Sains Terapan Teknologi Informasi*, 10, 38–44.
- Hidayat, T., & Muttaqin, M. (2018). Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis. *Jurnal Teknik Informatika UNIS*, 6(1), 25–29.
- Jr, R. M., & Schell, G. P. (2009). Sistem Informasi Manajemen (10th ed.; N. Setyaningsih, Ed.). Jakarta: Salemba Empat.
- Muhammad Aswadi. (2015). *Database Dasar With XAMPP*. Surabaya: CV. Garuda Mas Sejahtera.

- Nugroho, B. (2014). *Program Toko Dengan Visual Basic Dan MySQL*. Yogyakarta: Gava Media.
- Richards, G. (2011). *Warehouse Management: A Complete Guide to Improving Efficiency and Minimizing Costs in the Modern Warehouse*. London: Kogan Page.
- Schwaber, K., & Sutherland, J. (2017). Scrum Guide. Retrieved January 13, 2019, from <https://www.scrumguides.org/>.
- Sukanto, R. A., & Shalahuddin, M. (2014). *Rekayasa Perangkat Lunak "Terstruktur dan Berorientasi Objek."* Bandung: Informatika.
- Trisen, D. W., & Putra, J. J. (2018). Perancangan Sistem Informasi Pencarian Lowongan Pekerjaan. *Jurnal Teknologi Informasi*, 6(1), 48–54.